

Oracle® Quoting

API Reference Guide

Release 11*i*

Part No. B10591-01

April 2003

Oracle Quoting API Reference Guide, Release 11i

Part No. B10591-01

Copyright © 2002, 2003 Oracle Corporation. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information of Oracle Corporation; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent and other intellectual and industrial property laws. Reverse engineering, disassembly or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. Oracle Corporation does not warrant that this document is error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Oracle Corporation.

If the Programs are delivered to the U.S. Government or anyone licensing or using the programs on behalf of the U.S. Government, the following notice is applicable:

Restricted Rights Notice Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the Programs, including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication, and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and Oracle Corporation disclaims liability for any damages caused by such use of the Programs.

Oracle is a registered trademark, and Oracle 8, Oracle 8i, Oracle Metalink, PL/SQL, Oracle SQL*Plus, and Oracle Store are registered trademarks of Oracle Corporation. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments	xv
Preface	xvii
Audience for This Guide	xvii
How To Use This Guide	xvii
Typographic Conventions	xviii
Documentation Accessibility	xviii
Other Information Sources	xix
Do Not Use Database Tools to Modify Oracle Applications Data	xxiii
About Oracle	xxiv
1 oracle.apps.qot.core	
1.1 Package oracle.apps.qot.core	1-1
1.2 Class Approval	1-3
1.2.1 Fields for Class Approval	1-3
1.2.2 Constructors for Class Approval	1-4
1.2.3 Methods for Class Approval	1-4
1.3 Class ApprovalException.....	1-9
1.3.1 Fields for Class ApprovalException.....	1-10
1.3.2 Constructors for Class ApprovalException.....	1-10
1.4 Class Attachment.....	1-13
1.4.1 Fields for Class Attachment.....	1-14
1.4.2 Constructors for Class Attachment	1-14
1.4.3 Methods for Class Attachment.....	1-16

1.5	Class ConfigLaunchInfo	1-20
1.5.1	Fields for Class ConfigLaunchInfo	1-21
1.5.2	Constructors for Class ConfigLaunchInfo	1-21
1.5.3	Methods for Class ConfigLaunchInfo	1-21
1.6	Class Configurator.....	1-24
1.6.1	Fields for Class Configurator.....	1-24
1.6.2	Constructors for Class Configurator	1-25
1.6.3	Methods for Class Configurator.....	1-25
1.7	Class Contract	1-30
1.7.1	Fields for Class Contract	1-30
1.7.2	Constructors for Class Contract	1-33
1.7.3	Methods for Class Contract	1-33
1.8	Class ContractException.....	1-37
1.8.1	Fields for Class ContractException.....	1-38
1.8.2	Constructors for Class ContractException.....	1-39
1.9	Class ContractTemplate.....	1-41
1.9.1	Fields for Class ContractTemplate.....	1-42
1.9.2	Constructors for Class ContractTemplate.....	1-43
1.9.3	Methods for Class ContractTemplate.....	1-43
1.10	Class InstallBaseExtAttr.....	1-45
1.10.1	Fields for Class InstallBaseExtAttr.....	1-45
1.10.2	Constructors for Class InstallBaseExtAttr	1-45
1.10.3	Methods for Class InstallBaseExtAttr.....	1-46
1.11	Class InstallBaseItem.....	1-48
1.11.1	Fields for Class InstallBaseItem.....	1-48
1.11.2	Constructors for Class InstallBaseItem	1-52
1.11.3	Methods for Class InstallBaseItem.....	1-53
1.12	Class NotificationDetail.....	1-71
1.12.1	Fields for Class NotificationDetail.....	1-72
1.12.2	Constructors for Class NotificationDetail.....	1-73
1.12.3	Methods for Class NotificationDetail.....	1-74
1.13	Class PendingOrderLine	1-74
1.13.1	Fields for Class PendingOrderLine	1-74
1.13.2	Constructors for Class PendingOrderLine	1-76
1.13.3	Methods for Class PendingOrderLine	1-76

1.14	Class Proposal.....	1-83
1.14.1	Fields for Class Proposal.....	1-83
1.14.2	Constructors for Class Proposal.....	1-84
1.14.3	Methods for Class Proposal.....	1-85
1.15	Class Quote.....	1-87
1.15.1	Fields for Class Quote.....	1-87
1.15.2	Constructors for Class Quote	1-89
1.15.3	Methods for Class Quote.....	1-90
1.16	Class QuoteConstant.....	1-130
1.16.1	Fields for Class QuoteConstant.....	1-130
1.16.2	Constructors for Class QuoteConstant	1-135
1.17	Class QuoteException	1-135
1.17.1	Fields for Class QuoteException	1-136
1.17.2	Constructors for Class QuoteException.....	1-136
1.18	Class QuoteHeader	1-139
1.18.1	Fields for Class QuoteHeader	1-139
1.18.2	Constructors for Class QuoteHeader	1-142
1.18.3	Methods for Class QuoteHeader	1-142
1.19	Class QuoteLine.....	1-156
1.19.1	Fields for Class QuoteLine.....	1-157
1.19.2	Constructors for Class QuoteLine	1-160
1.19.3	Methods for Class QuoteLine.....	1-160
1.20	Class QuoteSecurityMgr	1-181
1.20.1	Fields for Class QuoteSecurityManager	1-181
1.20.2	Constructors for Class QuoteSecurityManager	1-181
1.20.3	Methods for Class QuoteSecurityManager	1-182
1.21	Class QuoteWarningException	1-185
1.21.1	Fields for Class QuoteWarningException	1-186
1.21.2	Constructors for Class QuoteWarningException	1-186
1.22	Class Service.....	1-189
1.22.1	Fields for Class Service.....	1-189
1.22.2	Constructors for Class Service.....	1-190
1.22.3	Methods for Class Service.....	1-190

2 oracle.apps.qot.core.util

2.1	Package oracle.apps.qot.core.util	2-1
2.2	Class CCPayment	2-2
2.2.1	Fields for Class CCPayment	2-3
2.2.2	Constructors for Class CCPayment	2-4
2.2.3	Methods for Class CCPayment	2-4
2.3	Class ConcurrentRequest	2-7
2.3.1	Fields for ConcurrentRequest.....	2-7
2.3.2	Constructors for ConcurrentRequest	2-7
2.3.3	Methods for ConcurrentRequest.....	2-7
2.4	Class Currency	2-13
2.4.1	Fields for Class Currency	2-14
2.4.2	Constructors for Class Currency	2-14
2.4.3	Methods for Class Currency	2-15
2.5	Class InstallBaseRelType.....	2-18
2.5.1	Fields for Class InstallBaseRelType	2-19
2.5.2	Constructors for Class InstallBaseRelType.....	2-19
2.5.3	Methods for Class InstallBaseRelType.....	2-19
2.6	Class MarketingSource	2-21
2.6.1	Fields for Class MarketingSource	2-22
2.6.2	Constructors for Class MarketingSource.....	2-22
2.6.3	Methods for Class MarketingSource	2-23
2.7	Class Opportunity	2-26
2.7.1	Fields for Class Opportunity	2-27
2.7.2	Constructors for Class Opportunity	2-28
2.7.3	Methods for Class Opportunity	2-28
2.8	Class PaymentTerm	2-41
2.8.1	Fields for Class PaymentTerm.....	2-41
2.8.2	Constructors for Class PaymentTerm	2-41
2.8.3	Methods for Class PaymentTerm.....	2-42
2.9	Class PriceAgreement.....	2-46
2.9.1	Fields for Class PriceAgreement	2-46
2.9.2	Constructors for Class PriceAgreement.....	2-46
2.9.3	Methods for Class PriceAgreement.....	2-47
2.10	Class PriceList	2-50

2.10.1	Fields for Class PriceList	2-51
2.10.2	Constructors for Class PriceList	2-51
2.10.3	Methods for Class PriceList	2-51
2.11	Class PriceModifier	2-55
2.11.1	Fields for Class PriceModifier	2-56
2.11.2	Constructors for Class PriceModifier	2-56
2.11.3	Methods for Class PriceModifier	2-57
2.12	Class QuoteStatus.....	2-59
2.12.1	Fields for Class QuoteStatus.....	2-60
2.12.2	Constructors for Class QuoteStatus.....	2-60
2.12.3	Methods for Class QuoteStatus.....	2-60
2.13	Class Resource	2-64
2.13.1	Fields for Class Resource	2-64
2.13.2	Constructors for Class Resource	2-64
2.13.3	Methods for Class Resource	2-65
2.14	Class ResourceGroup.....	2-72
2.14.1	Fields for Class ResourceGroup.....	2-73
2.14.2	Constructors for Class ResourceGroup.....	2-73
2.14.3	Methods for Class ResourceGroup.....	2-73
2.15	Class SalesCreditType	2-78
2.15.1	Fields for Class SalesCreditType	2-79
2.15.2	Constructors for Class SalesCreditType	2-79
2.15.3	Methods for Class SalesCreditType	2-79
2.16	Class Territory	2-82
2.16.1	Fields for Class Territory.....	2-82
2.16.2	Constructors for Class Territory	2-82
2.16.3	Methods for Class Territory.....	2-83
2.17	Class TransactionType.....	2-85
2.17.1	Fields for Class Transaction Type.....	2-86
2.17.2	Constructors for Class Transaction Type	2-86
2.17.3	Methods for Class TransactionType.....	2-87

3 oracle.apps.qot.inventory

3.1	Package oracle.apps.qot.inventory	3-1
3.2	Class InventoryException	3-2

3.2.1	Fields for Class InventoryException.....	3-3
3.2.2	Constructors for Class InventoryException	3-3
3.3	Class InventoryItem	3-8
3.3.1	Fields for Class InventoryItem	3-8
3.3.2	Constructors for Class InventoryItem.....	3-9
3.3.3	Methods for Class InventoryItem	3-11
3.4	Class InventoryUtil	3-27
3.4.1	Fields for Class InventoryUtil.....	3-28
3.4.2	Constructors for Class InventoryUtil	3-28
3.4.3	Methods for Class InventoryUtil.....	3-29
3.5	Class ItemCategory	3-30
3.5.1	Fields for Class ItemCategory	3-31
3.5.2	Constructors for Class ItemCategory	3-31
3.5.3	Methods for Class ItemCategory	3-31
3.6	Class ShippingMethod.....	3-34
3.6.1	Fields for Class ShippingMethod.....	3-34
3.6.2	Constructors for Class ShippingMethod	3-34
3.6.3	Methods for Class ShippingMethod.....	3-34

4 oracle.apps.qot.perzquery

4.1	Package oracle.apps.qot.perzquery	4-1
4.2	Class ColumnDefinition	4-2
4.2.1	Fields for Class ColumnDefinition	4-3
4.2.2	Constructors for Class ColumnDefinition	4-3
4.2.3	Methods for Class ColumnDefinition	4-5
4.3	Class ConditionParameter.....	4-10
4.3.1	Fields for Class ConditionParameter.....	4-10
4.3.2	Constructors for Class ConditionParameter	4-10
4.3.3	Methods for Class ConditionParameter.....	4-11
4.4	Class DataFormat	4-13
4.4.1	Fields for Class DataFormat.....	4-13
4.4.2	Constructors for Class DataFormat	4-14
4.5	Class DisplayColumn	4-15
4.5.1	Fields for Class DisplayColumn.....	4-15
4.5.2	Constructors for Class DisplayColumn	4-15

4.5.3	Methods for Class DisplayColumn	4-16
4.6	Class ListQuery.....	4-16
4.6.1	Fields for Class ListQuery.....	4-17
4.6.2	Constructors for Class ListQuery	4-18
4.6.3	Methods for Class ListQuery.....	4-18
4.7	Class OrderByParameter	4-25
4.7.1	Fields for Class OrderByParameter	4-25
4.7.2	Constructors for Class OrderByParameter.....	4-25
4.7.3	Methods for Class OrderByParameter	4-27
4.8	Class QueryBuilder	4-28
4.8.1	Fields for Class QueryBuilder	4-29
4.8.2	Constructors for Class QueryBuilder.....	4-29
4.8.3	Methods for Class QueryBuilder	4-29
4.9	Class QueryBuilderException	4-42
4.9.1	Fields for Class QueryBuilderException.....	4-43
4.9.2	Constructors for Class QueryBuilderException	4-44
4.10	Class SavedSearch	4-47
4.10.1	Fields for Class SavedSearch	4-48
4.10.2	Constructors for Class SavedSearch.....	4-48
4.10.3	Methods for Class SavedSearch	4-49
4.11	Class SavedSearchException	4-57
4.11.1	Fields for Class SavedSearchException.....	4-59
4.11.2	Constructors for Class SavedSearchException	4-59
4.12	Class SearchInfo.....	4-62
4.12.1	Fields for Class SearchInfo.....	4-63
4.12.2	Constructors for Class SearchInfo	4-63
4.12.3	Methods for Class SearchInfo.....	4-63

5 oracle.apps.aso.quote

5.1	Package oracle.apps.aso.quote	5-1
5.2	Class ApprovalInstanceRec	5-3
5.2.1	Fields for Class ApprovalInstanceRec	5-3
5.2.2	Constructors for Class ApprovalInstanceRec	5-5
5.2.3	Methods for Class ApprovalInstanceRec	5-5
5.3	Class ApproversListRec	5-6

5.3.1	Fields for Class ApproversListRec.....	5-6
5.3.2	Constructors for Class ApproversListRec	5-7
5.3.3	Methods for Class ApproversListRec.....	5-8
5.4	Class AtpRecord	5-8
5.4.1	Fields for Class AtpRecord	5-8
5.4.2	Constructors for Class AtpRecord	5-11
5.4.3	Methods for Class AtpRecord	5-11
5.5	Class AvailServiceRec	5-13
5.5.1	Fields for Class AvailServiceRec	5-13
5.5.2	Constructors for Class AvailServiceRec.....	5-14
5.5.3	Methods for Class AvailServiceRec	5-14
5.6	Class ControlRecord.....	5-15
5.6.1	Fields for Class ControlRecord.....	5-15
5.6.2	Constructors for Class ControlRecord	5-18
5.6.3	Methods for Class ControlRecord.....	5-19
5.7	Class CopyQuoteControlRecord	5-19
5.7.1	Fields for Class CopyQuoteControlRecord	5-20
5.7.2	Constructors for Class CopyQuoteControlRecord.....	5-21
5.7.3	Methods for Class CopyQuoteControlRecord	5-22
5.8	Class CopyQuoteHeaderRecord	5-23
5.8.1	Fields for CopyQuoteHeaderRecord.....	5-23
5.8.2	Constructors for Class CopyQuoteHeaderRecord	5-24
5.8.3	Methods for Class CopyQuoteHeaderRecord	5-25
5.9	Class HeaderRecord	5-25
5.9.1	Fields for Class HeaderRecord	5-25
5.9.2	Constructors for Class HeaderRecord.....	5-45
5.9.3	Methods for Class HeaderRecord.....	5-45
5.10	Class InstanceRecord	5-46
5.10.1	Fields for Class InstanceRecord.....	5-46
5.10.2	Constructors for Class InstanceRecord	5-46
5.10.3	Methods for Class InstanceRecord	5-47
5.11	Class LineDetailRecord.....	5-47
5.11.1	Fields for Class LineDetailRecord.....	5-48
5.11.2	Constructors for Class LineDetailRecord	5-58
5.11.3	Methods for Class LineDetailRecord.....	5-58

5.12	Class LineRecord	5-58
5.12.1	Fields for Class LineRecord	5-59
5.12.2	Constructors for Class LineRecord	5-71
5.12.3	Methods for Class LineRecord	5-72
5.13	Class LineRelationshipRecord	5-72
5.13.1	Fields for Class LineRelationshipRecord	5-72
5.13.2	Constructors for Class LineRelationshipRecord	5-75
5.13.3	Methods for Class LineRelationshipRecord	5-75
5.14	Class OppQteInRec	5-75
5.14.1	Fields for Class OppQteInRec	5-76
5.14.2	Constructors for Class OppQteInRec	5-78
5.14.3	Methods for Class OppQteInRec	5-79
5.15	Class OppQteOutRec	5-80
5.15.1	Fields for Class OppQteOutRec	5-80
5.15.2	Constructors for Class OppQteRecOut	5-81
5.15.3	Methods for Class OppQteRecOut	5-81
5.16	Class OrderHeaderRecord	5-81
5.16.1	Fields for Class OrderHeaderRecord	5-82
5.16.2	Constructors for Class OrderHeaderRecord	5-83
5.16.3	Methods for Class OrderHeaderRecord	5-83
5.17	Class OrderServiceRec	5-83
5.17.1	Fields for OrderServiceRec	5-84
5.17.2	Constructors for Class OrderServiceRec	5-84
5.17.3	Methods for Class OrderServiceRec	5-84
5.18	Class PaymentRecord	5-85
5.18.1	Fields for Class PaymentRecord	5-85
5.18.2	Constructors for Class PaymentRecord	5-90
5.18.3	Methods for Class PaymentRecord	5-90
5.19	Class PriceAdjustmentRecord	5-91
5.19.1	Fields for Class PriceAdjustmentRecord	5-91
5.19.2	Constructors for Class PriceAdjustmentRecord	5-103
5.19.3	Methods for Class PriceAdjustmentRecord	5-104
5.20	Class PriceAttributeRecord	5-104
5.20.1	Fields for Class PriceAttributeRecord	5-105
5.20.2	Constructors for Class PriceAttributeRecord	5-122

5.20.3	Methods for Class PriceAttributeRecord.....	5-122
5.21	Class QuoteAccessRecord	5-123
5.21.1	Fields for Class QuoteAccessRecord	5-123
5.21.2	Constructors for Class QuoteAccessRecord	5-128
5.21.3	Methods for Class QuoteAccessRecord	5-129
5.22	Class RelatedObjectRecord	5-129
5.22.1	Fields for Class RelatedObjectRecord.....	5-129
5.22.2	Constructors for Class RelatedObjectRecord	5-132
5.22.3	Methods for Class RelatedObjectRecord	5-132
5.23	Class RulesListRec.....	5-133
5.23.1	Fields for Class RulesListRec	5-133
5.23.2	Constructors for Class RulesListRec.....	5-134
5.23.3	Methods for Class RulesListRec.....	5-134
5.24	Class SalesCreditRecord	5-135
5.24.1	Fields for Class SalesCreditRecord	5-135
5.24.2	Constructors for Class SalesCreditRecord.....	5-140
5.24.3	Methods for Class SalesCreditRecord	5-140
5.25	Class ShipmentRecord	5-141
5.25.1	Fields for Class ShipmentRecord	5-141
5.25.2	Constructors for Class ShipmentRecord.....	5-150
5.25.3	Methods for Class ShipmentRecord	5-150
5.26	Class SubmitControlRecord.....	5-151
5.26.1	Fields for Class SubmitControlRecord	5-151
5.26.2	Constructors for Class SubmitControlRecord.....	5-152
5.26.3	Methods for Class SubmitControlRecord.....	5-152
5.27	Class TaxDetailRecord	5-153
5.27.1	Fields for Class TaxDetailRecord	5-153
5.27.2	Constructors for Class TaxDetailRecord.....	5-158
5.27.3	Methods for Class TaxDetailRecord	5-158
5.28	Class WarrantyRec	5-160
5.28.1	Fields for Class WarrantyRec	5-160
5.28.2	Constructors for Class WarrantyRec	5-161
5.28.3	Methods for Class WarrantyRec	5-161

6 oracle.apps.qot.util

6.1	Package oracle.apps.qot.util	6-1
6.2	Class FlexMapper	6-2
6.2.1	Fields for Class FlexMapper	6-2
6.2.2	Constructors for Class FlexMapper	6-2
6.2.3	Methods for Class FlexMapper	6-3
6.3	Class OperatingUnit.....	6-4
6.3.1	Fields for Class OperatingUnit.....	6-4
6.3.2	Constructors for Class OperatingUnit	6-5
6.3.3	Methods for Class OperatingUnit.....	6-5
6.4	Class LookupValuesKey.....	6-8
6.4.1	Fields for Class LookupValuesKey.....	6-9
6.4.2	Constructors for Class LookupValuesKey	6-9
6.4.3	Methods for Class LookupValuesKey.....	6-10
6.5	Class QotLookup	6-12
6.5.1	Fields for Class QotLookup	6-13
6.5.2	Constructors for Class QotLookup.....	6-16
6.5.3	Methods for Class QotLookup	6-17
6.6	Class QotRegionManager.....	6-25
6.6.1	Fields for Class QotRegionManager.....	6-25
6.6.2	Methods for Class QotRegionManager.....	6-26
6.7	Class QotUtil	6-28
6.7.1	Fields for Class QotUtil	6-29
6.7.2	Constructors for Class QotUtil.....	6-30
6.7.3	Methods for Class QotUtil	6-30
6.8	Class QueryResultSet.....	6-39
6.8.1	Fields for Class QueryResultSet.....	6-40
6.8.2	Constructors for Class QueryResultSet.....	6-40
6.8.3	Methods for Class QueryResultSet.....	6-40
6.9	Class RequestCtx	6-42
6.9.1	Fields for Class RequestCtx	6-43
6.9.2	Constructors for Class RequestCtx	6-44
6.9.3	Methods for Class RequestCtx	6-44

7 oracle.apps.qot.salesupp

7.1	Package oracle.apps.qot.salesupp	7-1
7.2	Class Component.....	7-2
7.2.1	Fields for Class Component.....	7-2
7.2.2	Constructors for Class Component	7-4
7.2.3	Methods for Class Component.....	7-4
7.3	Class InstanceResponseValue.....	7-9
7.3.1	Fields for Class InstanceResponseValue	7-9
7.3.2	Constructors for Class InstanceResponseValue.....	7-10
7.3.3	Methods for Class InstanceResponseValue.....	7-10
7.4	Class Response.....	7-12
7.4.1	Fields for Class Response	7-12
7.4.2	Constructors for Class Response.....	7-13
7.4.3	Methods for Class Response	7-13
7.5	Class SaleSuppException.....	7-16
7.5.1	Fields for Class SaleSuppException.....	7-17
7.5.2	Constructors for Class SaleSuppException	7-17
7.6	Class SectComponent.....	7-20
7.6.1	Fields for Class SectComponent.....	7-20
7.6.2	Constructors for Class SectComponent	7-22
7.6.3	Methods for Class SectComponent.....	7-22
7.7	Class Section.....	7-23
7.7.1	Fields for Class Section	7-24
7.7.2	Constructors for Class Section.....	7-24
7.7.3	Methods for Class Section	7-24
7.8	Class Template	7-27
7.8.1	Fields for Class Template	7-27
7.8.2	Constructors for Class Template.....	7-28
7.8.3	Methods for Class Template	7-28
7.9	Class TemplInstance.....	7-31
7.9.1	Fields for Class TemplInstance.....	7-32
7.9.2	Constructors for Class TemplInstance	7-32
7.9.3	Methods for Class TemplInstance.....	7-32

Send Us Your Comments

Oracle Quoting API Reference Guide, Release 11*i*

Part No. B10591-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: eccontent_us@oracle.com
- FAX: 650-654-6208 Attn: Oracle Quoting Documentation
- Postal service:

Oracle Corporation
Oracle Quoting Documentation
500 Oracle Parkway, 60p4
Redwood Shores, CA 94065
USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

Audience for This Guide

Welcome to Release 11*i* of the *Oracle Quoting API Reference Guide*.

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.
- Oracle Quoting
- The Oracle Applications graphical user interface.

To learn more about the Oracle Applications graphical user interface, read the *Oracle Applications User's Guide*.

See **Other Information Sources** for more information about Oracle Applications product information.

How To Use This Guide

This guide contains the information you need to understand Oracle Quoting APIs.

- Chapter 1 provides information about the APIs in package `oracle.apps.qot.core`.
- Chapter 2 provides information about the APIs in package `oracle.apps.qot.core.util`.
- Chapter 3 provides information about the APIs in package `oracle.apps.qot.inventory`.
- Chapter 4 provides information about the APIs in package `oracle.apps.qot.perzquery`.

- Chapter 5 provides information about the APIs in package oracle.apps.aso.quote.
- Chapter 6 provides information about the APIs in package oracle.apps.qot.util.
- Chapter 7 provides information about the APIs in package oracle.apps.qot.salesupp.

Typographic Conventions

The following table lists the typographic conventions used in this manual:

Convention	Meaning
<i>italic text</i>	Book titles
Courier text	User commands and file content examples
UPPERCASE	Structured Query Language (SQL) commands, initialization parameters, profile options, responsibilities, or environment variables
boldface text	Menu, button, keyboard, form options, and profile options Emphasized words or phrases
< >	Angle brackets enclose user-supplied names. Note: Do not type the angle brackets.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle Corporation is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

Accessibility of Code Examples in Documentation JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an

otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation This documentation may contain links to Web sites of other companies or organizations that Oracle Corporation does not own or control. Oracle Corporation neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Other Information Sources

You can choose from many sources of information, including online documentation, training, and support services, to increase your knowledge and understanding of Oracle Quoting.

If this guide refers you to other Oracle Applications documentation, use only the Release 11*i* versions of those guides.

Online Documentation

All Oracle Applications documentation is available online (HTML or PDF). Online help patches are available on MetaLink.

Related Documentation

Oracle Quoting shares business and setup information with other Oracle Applications products. Therefore, you may want to refer to other product documentation when you set up and use Oracle Quoting.

You can read the documents online by choosing Library from the expandable menu on your HTML help window, by reading from the Oracle Applications Document Library CD included in your media pack, or by using a Web browser with a URL that your system administrator provides.

If you require printed guides, you can purchase them from the Oracle Store at <http://oraclestore.oracle.com>.

Documents Related to All Products

Oracle Applications User's Guide

This guide explains how to enter data, query, run reports, and navigate using the graphical user interface (GUI) available with this release of Oracle Quoting (and any other Oracle Applications products). This guide also includes information on

setting user profiles, as well as running and reviewing reports and concurrent processes.

You can access this user's guide online by choosing "Getting Started with Oracle Applications" from any Oracle Applications help file.

Documents Related to This Product

Oracle Quoting User Guide

This document provides users with information on the general principles and procedures of creating and managing quotes in Oracle Quoting.

Oracle Quoting Implementation Guide

This document provides information and instructions to help you successfully implement Oracle Quoting.

Installation and System Administration

Oracle Applications Concepts

This guide provides an introduction to the concepts, features, technology stack, architecture, and terminology for Oracle Applications Release 11*i*. It provides a useful first book to read before an installation of Oracle Applications. This guide also introduces the concepts behind Applications-wide features such as Business Intelligence (BIS), languages and character sets, and Self-Service Web Applications.

Installing Oracle Applications

This guide provides instructions for managing the installation of Oracle Applications products. In Release 11*i*, much of the installation process is handled using Oracle Rapid Install, which minimizes the time to install Oracle Applications, the Oracle8 technology stack, and the Oracle8*i* Server technology stack by automating many of the required steps. This guide contains instructions for using Oracle Rapid Install and lists the tasks you need to perform to finish your installation. You should use this guide in conjunction with individual product user's guides and implementation guides.

Upgrading Oracle Applications

Refer to this guide if you are upgrading your Oracle Applications Release 10.7 or Release 11.0 products to Release 11*i*. This guide describes the upgrade process and lists database and product-specific upgrade tasks. You must be either at Release 10.7

(NCA, SmartClient, or character mode) or Release 11.0, to upgrade to Release 11*i*. You cannot upgrade to Release 11*i* directly from releases prior to 10.7.

Maintaining Oracle Applications

Use this guide to help you run the various AD utilities, such as AutoUpgrade, AutoPatch, AD Administration, AD Controller, AD Relink, License Manager, and others. It contains how-to steps, screenshots, and other information that you need to run the AD utilities. This guide also provides information on maintaining the Oracle applications file system and database.

Oracle Applications System Administrator's Guide

This guide provides planning and reference information for the Oracle Applications System Administrator. It contains information on how to define security, customize menus and online help, and manage concurrent processing.

Oracle Applications Developer's Guide

This guide contains the coding standards followed by the Oracle Applications development staff. It describes the Oracle Application Object Library components needed to implement the Oracle Applications user interface described in the *Oracle Applications User Interface Standards for Forms-Based Products*. It also provides information to help you build your custom Oracle Forms Developer 6*i* forms so that they integrate with Oracle Applications.

Oracle Applications User Interface Standards for Forms-Based Products

This guide contains the user interface (UI) standards followed by the Oracle Applications development staff. It describes the UI for the Oracle Applications products and how to apply this UI to the design of an application built by using Oracle Forms.

Other Implementation Documentation

Oracle Applications Flexfields Guide

This guide provides flexfields planning, setup and reference information for the Oracle Quoting implementation team, as well as for users responsible for the ongoing maintenance of Oracle Applications product data. This manual also provides information on creating custom reports on flexfields data.

Oracle eTechnical Reference Manuals

Each eTechnical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for a specific Oracle Applications product. This information helps you convert data from your existing applications, integrate Oracle Applications data with non-Oracle applications, and write custom reports for Oracle Applications products. Oracle eTRM is available on Metalink

Oracle Applications Message Reference Manual

This manual describes Oracle Applications messages. This manual is available in HTML format on the documentation CD-ROM for Release 11*i*.

Oracle CRM Application Foundation Implementation Guide

Many CRM products use components from CRM Application Foundation. Use this guide to correctly implement CRM Application Foundation.

Training and Support

Training

Oracle offers training courses to help you and your staff master Oracle Quoting and reach full productivity quickly. You have a choice of educational environments. You can attend courses offered by Oracle University at any one of our many Education Centers, you can arrange for our trainers to teach at your facility, or you can use Oracle Learning Network (OLN), Oracle University's online education utility. In addition, Oracle training professionals can tailor standard courses or develop custom courses to meet your needs. For example, you may want to use your organization's structure, terminology, and data as examples in a customized training session delivered at your own facility.

Support

From on-site support to central support, our team of experienced professionals provides the help and information you need to keep Oracle Quoting working for you. This team includes your Technical Representative, Account Manager, and Oracle's large staff of consultants and support specialists with expertise in your business area, managing an Oracle8*i* server, and your hardware and software environment.

OracleMetaLink

OracleMetaLink is your self-service support connection with web, telephone menu, and e-mail alternatives. Oracle supplies these technologies for your convenience, available 24 hours a day, 7 days a week. With OracleMetaLink, you can obtain information and advice from technical libraries and forums, download patches, download the latest documentation, look at bug details, and create or update TARs. To use MetaLink, register at (<http://metalink.oracle.com>).

Alerts: You should check OracleMetaLink alerts before you begin to install or upgrade any of your Oracle Applications. Navigate to the Alerts page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade/Alerts.

Self-Service Toolkit: You may also find information by navigating to the Self-Service Toolkit page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade.

Do Not Use Database Tools to Modify Oracle Applications Data

Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle Applications data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle Applications data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle Applications tables are interrelated, any change you make using Oracle Applications can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

When you use Oracle Applications to modify your data, Oracle Applications automatically checks that your changes are valid. Oracle Applications also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

About Oracle

Oracle Corporation develops and markets an integrated line of software products for database management, applications development, decision support, and office automation, as well as Oracle Applications, an integrated suite of more than 160 software modules for financial management, supply chain management, manufacturing, project systems, human resources and customer relationship management.

Oracle products are available for mainframes, minicomputers, personal computers, network computers and personal digital assistants, allowing organizations to integrate different computers, different operating systems, different networks, and even different database management systems, into a single, unified computing and information resource.

Oracle is the world's leading supplier of software for information management, and the world's second largest software company. Oracle offers its database, tools, and applications products, along with related consulting, education, and support services, in over 145 countries around the world.

oracle.apps.qot.core

This section lists the Oracle Quoting Java APIs in the package oracle.apps.qot.core.

1.1 Package oracle.apps.qot.core

The package oracle.apps.qot.core contains the APIs for Oracle Quoting core procedures. The table below lists a description for each class:

Table 1–1 *Class Summary for oracle.apps.qot.core*

Class	Description
Class Approval	Approval provides the methods for starting/cancelling the approval process and retrieving information regarding quote approvals.
Class Attachment	The Attachment object contains the information for an attachment that is associated with a quote.
Class ConfigLaunchInfo	ConfigLaunchInfo object contains information which is needed to launch the configurator: ICX session ticket, database ID, whether the inventory item is configurable, servlet URL for the Configurator, database system date.
Class Configurator	Configurator Class handles the integration between Quoting and the Oracle Configurator product.
Class Contract	Contract contains methods retrieve contract information, create contract, and update contract.
Class ContractTemplate	ContractTemplate contains methods to retrieve the default contract template and retrieve the list of available contract templates.

Table 1–1 Class Summary for oracle.apps.qot.core

Class	Description
Class InstallBaseExtAttr	The InstallBaseExtAttr object is modeled to provide details of the extended attributes associated with the install base item.
Class InstallBaseItem	InstallBaseItem contains methods to retrieve information for an install base instance and to search for install base instances.
Class NotificationDetail	NotificationDetail contains the following information for approval notification details: recipient role name, activity display name, begin date, end date, due date, activity status, result.
Class PendingOrderLine	PendingOrderLine contains methods to retrieve information for a pending order line and to search for pending order lines.
Class Proposal	The Proposal object contains the information for a proposal that is associated with a quote.
Class Quote	Quote object is used to model an entire quote.
Class QuoteConstant	QuoteConstant contains constants that are shared by the classes in the package oracle.apps.qot.core
Class QuoteHeader	QuoteHeader object is used to model a quote header and its additional attributes, which include payments, price attributes, price adjustments, sales credits, shipments, and tax details.
Class QuoteLine	QuoteLine object is used to model a quote line and its additional attributes, which include line details, payments, shipments, price attributes, price adjustments, sales credits, shipments, and tax details.
Class QuoteSecurityMgr	The QuoteSecurityMgr contains the methods used to implement quote security.
Class Service	The Service object contains the basic information for a service, such as inventory item part number, inventory item description, service start date, service duration, service period code, whether the service is an included warranty.
Exceptions	Description
Class ApprovalException	ApprovalException is thrown when an application error occurs in an Approval method.

Table 1–1 Class Summary for oracle.apps.qot.core

Class	Description
Class ContractException	ContractException is thrown when an application error occurs in a Contract method.
Class QuoteException	QuoteException is thrown when an application error occurs in a method in package oracle.apps.qot.core or oracle.apps.qot.core.util.

1.2 Class Approval

```
java.lang.Object
|
+--oracle.apps.qot.core.Approval
public class Approval
```

Approval provides the methods for starting/cancelling the approval process and retrieving information regarding quote approvals.

Table 1–2 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(long, int), wait(long, int), wait(long, int)

1.2.1 Fields for Class Approval

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to oracle.apps.fnd.common.VersionInfo.recordClassVersion.

1.2.2 Constructors for Class Approval

Approval()

```
public Approval()
```

1.2.3 Methods for Class Approval

The following table is an index of the Class Approval methods:

Table 1–3 Methods for Class Approval

Method	Description
<code>cancelApproval(BigDecimal, BigDecimal)</code>	<p>Cancels an approval process.</p> <pre>public static void cancelApproval(java.math.BigDecimal quoteHdrId, java.math.BigDecimal approvalInstance Id) throws FrameworkException, SQLExcepti on, ApprovalException</pre>
<code>getAllApprovalInstances(BigDecimal)</code>	<p>Returns all the approval instances for a given quote number.</p> <pre>public static oracle.apps.aso.quote.ApprovalInstanc eRec[] getAllApprovalInstances(java.math.Big Decimal quoteNumber) throws FrameworkException, SQLExcepti on</pre>
<code>getApprovalInstance(BigDecimal)</code>	<p>Returns a approval instance given object approval ID.</p> <pre>public static oracle.apps.aso.quote.ApprovalInstanc eRec getApprovalInstance(java.math.BigDeci mal objApvlId) throws FrameworkException, SQLExcepti on</pre>

Table 1–3 Methods for Class Approval

Method	Description
<code>getApproverComments(BigDecimal)</code>	Returns approver comments for a given approval. <pre>public static java.lang.String getApproverComments(java.math.BigDeci mal approvalDetId) throws FrameworkException, SQLExcepti on</pre>
<code>getApprovers(BigDecimal)</code>	Returns approvers for a given approval instance. <pre>public static oracle.apps.aso.quote.ApproversListRe c[] getApprovers(java.math.BigDecimal objApvlId) throws FrameworkException, SQLExcepti on</pre>
<code>getNotificationHistory(BigDecimal)</code>	Returns notification history for a given approval. <pre>public static oracle.apps.qot.core.NotificationDeta il[] getNotificationHistory(java.math.BigD ecimal approvalDetId) throws FrameworkException, SQLExcepti on</pre>
<code>getQuoteApprovers(BigDecimal, ApproversListRec[][], RulesListRec[][])</code>	Returns quote approvers for a given quote. <pre>public static void getQuoteApprovers(java.math.BigDecima l qotHdrId, oracle.apps.aso.quote.ApproversListRe c[][] approverList, oracle.apps.aso.quote.RulesListRec[] [] ruleList) throws FrameworkException, SQLExcepti on, ApprovalException</pre>
<code>getRules(BigDecimal)</code>	Returns rules for a given approval instance. <pre>public static oracle.apps.aso.quote.RulesListRec[] getRules(java.math.BigDecimal objApvlId) throws FrameworkException, SQLExcepti on</pre>

Table 1–3 Methods for Class Approval

Method	Description
<code>startApproval(BigDecimal, BigDecimal, String)</code>	<p>Starts approval process.</p> <pre>public static void startApproval(java.math.BigDecimal gotHdrId, java.math.BigDecimal startSequence, java.lang.String comments) throws FrameworkException, SQLException, ApprovalException</pre>

cancelApproval(BigDecimal, BigDecimal)

```
public static void cancelApproval(java.math.BigDecimal gotHdrId,
java.math.BigDecimal approvalInstanceId)
throws FrameworkException, SQLException, ApprovalException
```

Cancels an approval process. This API should be called within a transaction block.

Parameters:

`gotHdrId` - Quote header ID.

`approvalInstanceId` - Approval instance ID.

Throws:

`ApprovalException` - If an application error occurs.

`java.sql.SQLException` - If a database error occurs.

`oracle.apps.jtf.base.resources.FrameworkException` - If an error occurs while getting connection.

getAllApprovalInstances(BigDecimal)

```
public static oracle.apps.aso.quote.ApprovalInstanceRec[]
getAllApprovalInstances(java.math.BigDecimal quoteNumber)
throws FrameworkException, SQLException
```

Returns all the approval instances for a given quote number.

Parameters: `quoteNumber` - Quote number.

Returns: Array of `ApprovalInstanceRec`.

Throws:

`java.sql.SQLException` - If a database error occurs.

`oracle.apps.jtf.base.resources.FrameworkException` - If an error occurs while getting connection.

getApprovalInstance(BigDecimal)

```
public static oracle.apps.aso.quote.ApprovalInstanceRec  
getApprovalInstance(java.math.BigDecimal objApvId)  
throws FrameworkException, SQLException
```

Returns a approval instance given object approval ID.

Parameters: objApvId - Object approval ID.

Returns: ApprovalInstanceRec with the approval instance information.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

getApproverComments(BigDecimal)

```
public static java.lang.String getApproverComments(java.math.BigDecimal  
approvalDetId)  
throws FrameworkException, SQLException
```

Returns approver comments for a given approval.

Parameters: approvalDetId - Approval detail ID.

Returns: String containing approver comments.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

getApprovers(BigDecimal)

```
public static oracle.apps.aso.quote.ApproversListRec[]  
getApprovers(java.math.BigDecimal objApvId)  
throws FrameworkException, SQLException
```

Returns approvers for a given approval instance.

Parameters: objApvId - Object approval ID.

Returns: Array of ApproversListRec.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

getNotificationHistory(BigDecimal)

```
public static oracle.apps.qot.core.NotificationDetail[]  
getNotificationHistory(java.math.BigDecimal approvalDetId)  
throws FrameworkException, SQLException
```

Returns notification history for a given approval.

Parameters: approvalDetId - Approval detail ID.

Returns: Array of NotificationDetail containing notification history information.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

getQuoteApprovers(BigDecimal, ApproversListRec[][], RulesListRec[][])

```
public static void getQuoteApprovers(java.math.BigDecimal qotHdrId,  
oracle.apps.aso.quote.ApproversListRec[][] approverList,  
oracle.apps.aso.quote.RulesListRec[][] ruleList)  
throws FrameworkException, SQLException, ApprovalException
```

Returns quote approvers for a given quote.

Parameters:

qotHdrId - Quote header ID.

approverList - The list of approvers required for an approval. approverList should be initialized to new ApproversListRec[1][] before it is passed as input. After the method returns, approverList[0] will be populated with the list of approvers.

ruleList - The list of rules which determines the list of approvers required. ruleList should be initialized to new RulesListRec[1][] before it is passed as input. After the method returns, rulesList[0] will be populated with the list of rules.

Throws:

ApprovalException - If an application error occurs.

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

getRules(BigDecimal)

```
public static oracle.apps.aso.quote.RulesListRec[]  
getRules(java.math.BigDecimal objApvlId)  
throws FrameworkException, SQLException
```


Returns rules for a given approval instance.

Parameters: objApvId - Object approval ID.

Returns: Array of RulesListRec.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

startApproval(BigDecimal, BigDecimal, String)

```
public static void startApproval(java.math.BigDecimal qotHdrId,
    java.math.BigDecimal startSequence, java.lang.String comments)
    throws FrameworkException, SQLException, ApprovalException
```

Starts the approval process. This API should be called within a transaction block.

Parameters:

qotHdrId - Quote header ID.

startSequence - Start sequence for approval process.

comments - Comments for starting approval process.

Throws:

ApprovalException - If an application error occurs.

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

1.3 Class ApprovalException

```
java.lang.Object
|
+--java.lang.Throwable
    |
    +--java.lang.Exception
        |
        +--oracle.apps.jtf.base.resources.FrameworkException
            |
            +--oracle.apps.qot.core.ApprovalException
```

All Implemented Interfaces: java.io.Serializable

```
public class ApprovalException extends
    oracle.apps.jtf.base.resources.FrameworkException
```

ApprovalException is thrown when an application error occurs in an Approval method.

Table 1–4 Inherited Member Summary

Fields inherited from interface FrameworkException

DEBUG, ERROR, FATAL, INFORMATION, NONE, WARNING, defaultMsgMgr

Methods inherited from interface FrameworkException

addException(Exception), convertException(Exception), getAllInfo(),
 getCurrentMessageManager(), getExceptionStack(), getExceptionStackRec(),
 getExternException(), getKey(), getMessage(), getMessageManager(), getMessageStack(),
 getParameters(), getParentExcep(), getRootException(), getRootExternExcept(),
 getSeverity(), getThrowerInfo(), getWholeStack(), printAllInfo(PrintStream),
 printAllInfo(PrintStream), printMesg(PrintStream), printMesg(PrintStream),
 printMessageStack(PrintWriter), printMessageStack(PrintWriter),
 printStackTrace(PrintWriter), printStackTrace(PrintWriter),
 printThrowerInfo(PrintStream), printThrowerInfo(PrintStream),
 printWholeStack(PrintStream), printWholeStack(PrintStream), setCurrents(),
 setStackTrace(String)

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long,
 int), wait(long, int)

Methods inherited from class Throwable

fillInStackTrace(), getLocalizedMessage(), printStackTrace(), toString()

1.3.1 Fields for Class ApprovalException

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.3.2 Constructors for Class ApprovalException

ApprovalException(Exception, String)

```
public ApprovalException(java.lang.Exception e, java.lang.String errorKey)
```

Constructs an exception with the given exception and errorKey.

Parameters:

e - The parent exception.

errorKey - Error key.

ApprovalException(Exception, String, Object[])

```
public ApprovalException(java.lang.Exception e, java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs an exception with the given exception, errorKey, and parameters.

Parameters:

e - The parent exception.

errorKey - Error key.

params - An array of tokens for errorKey.

ApprovalException(Exception, String, String)

```
public ApprovalException(java.lang.Exception e, java.lang.String errorKey,  
java.lang.String param)
```

Constructs an exception with the given exception, errorKey, and parameter.

Parameters:

e - The parent exception.

errorKey - Error key.

param - Token for the error key.

ApprovalException(int, String)

```
public ApprovalException(int err_msg_count, java.lang.String errorKey)  
throws FrameworkException
```

Constructs an exception with the message count and error key. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

ApprovalException(int, String, Object[])

```
public ApprovalException(int err_msg_count, java.lang.String errorKey,  
java.lang.Object[] params)
```

throws FrameworkException

Constructs an exception with the message count, error key, and parameter tokens. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

params - An array of tokens for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

ApprovalException(int, String, String)

```
public ApprovalException(int err_msg_count, java.lang.String errorKey,  
java.lang.String param)
```

throws FrameworkException

Constructs an exception with the message count, error key, and parameter token. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

param - A token for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

ApprovalException(String)

```
public ApprovalException(java.lang.String errorKey)
```

Constructs an exception with the errorKey.

Parameters: errorKey - Error key.

ApprovalException(String, Object[])

```
public ApprovalException(java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs an exception with the errorKey and parameters.

Parameters:

errorKey - Error key.

params - An array of tokens for errorKey.

ApprovalException(String, String)

```
public ApprovalException(java.lang.String err_msg,  
java.lang.String errorKey)
```

Constructs an exception with the error message and error key.

Parameters:

err_msg - Error message.

errorKey - Error key.

ApprovalException(String, String, Object[])

```
public ApprovalException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.Object[] params)
```

Constructs an exception with the error message, error key, and parameter tokens.

Parameters:

err_msg - Error message.

errorKey - Error key.

params - An array of tokens for errorKey.

ApprovalException(String, String, String)

```
public ApprovalException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.String param)
```

Constructs an exception with the error message, error key, and parameter token.

Parameters:

err_msg - Error message.

errorKey - Error key.

param - Token for the error key.

1.4 Class Attachment

```
java.lang.Object  
|  
+--oracle.apps.qot.core.Attachment
```

```
public class Attachment
```

The Attachment object contains the information for an attachment that is associated with a quote. It contains the following attributes: quote attachment ID, quote header ID, quote line ID, description, and the attached file name.

This class provides the ability to list attachments associated with a quote, create attachments associated to a quote, and delete attachments associated with a quote.

Table 1–5 Inherited Member Summary

Methods inherited from class Object

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

1.4.1 Fields for Class Attachment

FILE

```
public static final int FILE
```

Constant indicating that the attachment type is a file.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.4.2 Constructors for Class Attachment

Attachment()

```
public Attachment()
```

Default constructor.

Attachment(BigDecimal, BigDecimal, BigDecimal, String, String, BigDecimal, BigDecimal, String, Timestamp)

```
public Attachment(java.math.BigDecimal quoteAttachmentId,  
java.math.BigDecimal quoteHeaderId, java.math.BigDecimal quoteLineId,  
java.lang.String description, java.lang.String fileName,  
java.math.BigDecimal fileId, java.math.BigDecimal createdById,  
java.lang.String createdByName, java.sql.Timestamp createDate)
```

Constructor.

Parameters:

`quoteAttachmentId` - Quote attachment ID.

`quoteHeaderId` - Quote header ID.

quoteLineId - Quote line ID.

description - Description of the attachment.

fileName - File name of the attachment.

fileId - File ID of the attached document.

createdById - ID of the user who created the document.

createdByName - Name of the user who created the document.

createDate - Creation date of the document.

1.4.3 Methods for Class Attachment

The following table is an index of the Class Attachment methods:

Table 1–6 Methods for Class Attachment

Method	Description
<code>add(BigDecimal, BigDecimal, int, Object, String, String)</code>	<p>Adds an attachment to the specified quote. This API should be called within a transaction block.</p> <pre>public static void add(java.math.BigDecimal quoteHeaderId, java.math.BigDecimal quoteLineId, int attachType, java.lang.Object attachContent, java.lang.String attachDesc, java.lang.String docCtgId)</pre> <p>throws <code>FrameworkException</code>, <code>SQLException</code>, <code>QuoteException</code></p>
<code>delete(BigDecimal, BigDecimal, BigDecimal[])</code>	<p>Deletes a list of attachments associated with the quote. The specified attachments will be deleted from the specified quote. This API should be called within a transaction block.</p> <pre>public static void delete(java.math.BigDecimal quoteHeaderId, java.math.BigDecimal quoteLineId, java.math.BigDecimal[] quoteAttachmentIds)</pre> <p>throws <code>FrameworkException</code>, <code>SQLException</code>, <code>QuoteException</code></p>
<code>getCreatedById()</code>	<p>Returns ID of the user who created the attachment.</p> <pre>public java.math.BigDecimal getCreatedById()</pre>
<code>getCreatedByName()</code>	<p>Returns name of the user who created the attachment.</p> <pre>public java.lang.String getCreatedByName()</pre>
<code>getCreatedDate()</code>	<p>Returns attachment creation date.</p> <pre>public java.sql.Timestamp getCreatedDate()</pre>
<code>getDescription()</code>	<p>Returns attachment description.</p> <pre>public java.lang.String getDescription()</pre>
<code>getFileId()</code>	<p>Returns file ID.</p> <pre>public java.math.BigDecimal getFileId()</pre>
<code>getFileName()</code>	<p>Returns attached file name.</p> <pre>public java.lang.String getFileName()</pre>

Table 1–6 Methods for Class Attachment

Method	Description
<code>getQuoteAttachmentId()</code>	Returns quote attachment ID. <pre>public java.math.BigDecimal getQuoteAttachmentId()</pre>
<code>getQuoteHeaderId()</code>	Returns quote header ID. <pre>public java.math.BigDecimal getQuoteHeaderId()</pre>
<code>getQuoteLineId()</code>	Returns quote line ID. <pre>public java.math.BigDecimal getQuoteLineId()</pre>
<code>list(BigDecimal, BigDecimal, int, int, int)</code>	Lists the attachments associated with the quote header passed in as parameter. Attachments at quote line level are currently not supported. <pre>public static oracle.apps.qot.util.QueryResultSet list(java.math.BigDecimal quoteHeaderId, java.math.BigDecimal quoteLineId, int batchSize, int startIndex, int resultCount) throws FrameworkException, SQLException</pre>
<code>toString()</code>	Returns a String representation of the Attachment object. <pre>public java.lang.String toString()</pre>

add(BigDecimal, BigDecimal, int, Object, String, String)

```
public static void add(java.math.BigDecimal quoteHeaderId,
java.math.BigDecimal quoteLineId, int attachType,
java.lang.Object attachContent, java.lang.String attachDesc,
java.lang.String docCtgId)
throws FrameworkException, SQLException, QuoteException
```

Adds an attachment to the specified quote. This API should be called within a transaction block.

Parameters:

`quoteHeaderId` - Quote header ID.

`quoteLineId` - Quote line ID. Currently not supported.

`attachType` - Attachment type. Currently, only FILE is supported.

`attachContent` - Attachment content.

`attachDesc` - Attachment description.

docCtgId - Document category ID.

Throws:

QuoteException - If an application error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

delete(BigDecimal, BigDecimal, BigDecimal[])

```
public static void delete(java.math.BigDecimal quoteHeaderId,  
java.math.BigDecimal quoteLineId,  
java.math.BigDecimal[] quoteAttachmentIds)  
throws FrameworkException, SQLException, QuoteException
```

Deletes a list of attachments associated with the quote. The specified attachments will be deleted from the specified quote. This API should be called within a transaction block.

Parameters:

quoteHeaderId - Quote Header ID.

quoteLineId - Quote line ID.

quoteAttachmentIds - Quote attachment IDs.

Throws:

QuoteException - If an application error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

getCreatedById()

```
public java.math.BigDecimal getCreatedById()  
Returns ID of the user who created the attachment.
```

Returns: ID of the user who created the attachment.

getCreatedByName()

```
public java.lang.String getCreatedByName()  
Returns name of the user who created the attachment.
```

Returns: Name of the user who created the attachment.

getCreatedDate()

```
public java.sql.Timestamp getCreatedDate()  
Returns attachment creation date.
```

Returns: Attachment creation date.

getDescription()

```
public java.lang.String getDescription()
```

Returns attachment description.

Returns: Attachment description.

getFileId()

```
public java.math.BigDecimal getFileId()
```

Returns file ID.

Returns: Attachment file ID.

getFileName()

```
public java.lang.String getFileName()
```

Returns attached file name.

Returns: Attached file name.

getQuoteAttachmentId()

```
public java.math.BigDecimal getQuoteAttachmentId()
```

Returns quote attachment ID.

Returns: Quote attachment ID.

getQuoteHeaderId()

```
public java.math.BigDecimal getQuoteHeaderId()
```

Returns quote header ID.

Returns: Quote header ID.

getQuoteLineId()

```
public java.math.BigDecimal getQuoteLineId()
```

Returns quote line ID.

Returns: Quote line ID.

list(BigDecimal, BigDecimal, int, int, int)

```
public static oracle.apps.qot.util.QueryResultSet  
list(java.math.BigDecimal quoteHeaderId,  
java.math.BigDecimal quoteLineId, int batchSize, int startIndex,  
int resultCount)  
throws FrameworkException, SQLException
```

Lists the attachments associated with the quote header passed in as parameter. Attachments at quote line level are currently not supported.

Parameters:

quoteHeaderId - Quote header ID.

quoteLineId - Quote Line ID, currently not supported.

batchSize - Batch size for the query.

startIndex - Start index indicating which row number to start retrieving data. -1 indicates that last page.

resultCount - The total number of objects to return. If -1, the count will be queried from the database.

Returns: QueryResultSet containing Attachment[]

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the Attachment object.

Overrides: toString in class Object

Returns: A String representation of the attachment object.

1.5 Class ConfigLaunchInfo

```
java.lang.Object
|
+--oracle.apps.qot.core.ConfigLaunchInfo
```

public class ConfigLaunchInfo

ConfigLaunchInfo object contains information which is needed to launch the configurator: ICX session ticket, database ID, whether the inventory item is configurable, servlet URL for the Configurator, database system date.

Table 1-7 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

1.5.1 Fields for Class ConfigLaunchInfo

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.5.2 Constructors for Class ConfigLaunchInfo

ConfigLaunchInfo(String, String, boolean, String, String)

```
public ConfigLaunchInfo(java.lang.String icxSessionTicket,
    java.lang.String dbId, boolean isConfigurable, java.lang.String servletUrl,
    java.lang.String sysdate)
```

Creates ConfigLaunchInfo with the parameters passed in as parameter.

Parameters:

icxSessionTicket - ICX session ticket.

dbId - Database ID.

isConfigurable - Whether the item is configurable.

servletUrl - URL for Configurator servlet.

sysdate - System date from the database.

1.5.3 Methods for Class ConfigLaunchInfo

The following table is an index of the Class ConfigLaunchInfo methods:

Table 1–8 *Methods for Class ConfigLaunchInfo*

Method	Description
getDbId()	Returns database ID. public java.lang.String getDbId()
getIcxSessionTicket()	Returns ICX session ticket. public java.lang.String getIcxSessionTicket()
getServletUrl()	Returns the Configurator servlet URL. public java.lang.String getServletUrl()

Table 1–8 Methods for Class ConfigLaunchInfo

Method	Description
getSysdate()	Returns database system date. <code>public java.lang.String getSysdate()</code>
isConfigurable()	Indicates whether item is configurable. <code>public boolean isConfigurable()</code>
setConfigurable(boolean)	Sets whether the item is configurable. <code>public void setConfigurable(boolean isConfigurable)</code>
setDbId(String)	Sets database ID. <code>public void setDbId(java.lang.String dbId)</code>
setIcxSessionTicket(String)	Sets ICX session ticket. <code>public void setIcxSessionTicket(java.lang.String icxSessionTicket)</code>
setServletUrl(String)	Sets the Configurator Servlet. <code>public void setServletUrl(java.lang.String servletUrl)</code>
setSysdate(String)	Sets the database system date. <code>public void setSysdate(java.lang.String sysdate)</code>
toString()	Returns a String representation of the ConfigLaunchInfo object. <code>public java.lang.String toString()</code>

getDbId()

```
public java.lang.String getDbId()
```

Returns the database ID.

Returns: The database ID.

getIcxSessionTicket()

```
public java.lang.String getIcxSessionTicket()
```

Returns the ICX session ticket.

Returns: The ICX session ticket.

getServletUrl()

```
public java.lang.String getServletUrl()
```

Returns the URL of the Configurator servlet.

Returns: The URL of the Configurator servlet.

getSysdate()

```
public java.lang.String getSysdate()
```

Returns the database system date.

Returns: The database system date.

isConfigurable()

```
public boolean isConfigurable()
```

Returns whether the item is configurable.

Returns: A boolean value indicating whether the item is configurable.

setConfigurable(boolean)

```
public void setConfigurable(boolean isConfigurable)
```

Sets whether the item is configurable.

Parameters: isConfigurable - A boolean value indicating whether the item is configurable

setDbId(String)

```
public void setDbId(java.lang.String dbId)
```

Sets the database ID.

Parameters: dbId - The database ID.

setIcxSessionTicket(String)

```
public void setIcxSessionTicket(java.lang.String icxSessionTicket)
```

Sets the ICX session ticket.

Parameters: icxSessionTicket - The ICX session ticket.

setServletUrl(String)

```
public void setServletUrl(java.lang.String servletUrl)
```

Sets the URL of the Configurator servlet.

Parameters: servletUrl - The URL of the Configurator servlet.

setSysdate(String)

```
public void setSysdate(java.lang.String sysdate)
```

Sets the database system date.

Parameters: sysdate - The database system date.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the ConfigLaunchInfo object.

Overrides: toString in class Object

Returns: A String representation of the ConfigLaunchInfo object.

1.6 Class Configurator

```
java.lang.Object  
|  
+--oracle.apps.qot.core.Configurator
```

public class Configurator

Configurator Class handles the integration between Quoting and the Oracle Configurator product. It contains methods to retrieve information needed to launch the Configurator, add a configuration to quote, and utility methods for constructing/processing the Configurator initialization/termination messages.

Table 1–9 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(long, int), wait(long, int), wait(long, int)

1.6.1 Fields for Class Configurator

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```


1.6.2 Constructors for Class Configurator

Configurator()

```
public Configurator()
```

1.6.3 Methods for Class Configurator

The following table is an index of the Class Configurator methods:

Table 1–10 *Methods for Class Configurator*

Method	Description
addConfigToQuote(ControlRecord, BigDecimal, BigDecimal, BigDecimal, BigDecimal, String, String)	<p>Adds the child items of a Configured product to the quote.</p> <pre>public static void addConfigToQuote(oracle.apps.aso.quote.ControlRecord controlRec, java.math.BigDecimal quoteHdrId, java.math.BigDecimal quoteLnId, java.math.BigDecimal configHdrId, java.math.BigDecimal configRevNum, java.lang.String validCfg, java.lang.String completeCfg) throws FrameworkException, SQLException, QuoteException</pre>
addConfigToQuote(ControlRecord, QuoteHeader, BigDecimal, BigDecimal, BigDecimal, String, String)	<p>Adds the child items of a configured model item to a quote. This API should be called within a transaction block.</p> <pre>public static void addConfigToQuote(oracle.apps.aso.quote.ControlRecord controlRec, oracle.apps.qot.core.QuoteHeader quoteHdr, java.math.BigDecimal quoteLnId, java.math.BigDecimal configHdrId, java.math.BigDecimal configRevNum, java.lang.String validCfg, java.lang.String completeCfg) throws FrameworkException, SQLException, QuoteException</pre>

Table 1–10 Methods for Class Configurator

Method	Description
<code>getInitTag(String, String)</code>	<p>Helper function to extract values from an initialize message. Finds the given tag and returns the value for that tag in the XML message string.</p> <pre>public static java.lang.String getInitTag(java.lang.String xmlMsg, java.lang.String tagName)</pre>
<code>getLaunchInfo(BigDecimal, BigDecimal)</code>	<p>Returns details from the database for different parameters of the initialize message to launch the Configurator.</p> <pre>public static oracle.apps.qot.core.ConfigLaunchInfo getLaunchInfo(java.math.BigDecimal invItemId, java.math.BigDecimal invOrgId)</pre> <p>throws <code>SQLException</code>, <code>FrameworkException</code>, <code>QuoteException</code></p>
<code>getRootPath(String)</code>	<p>Helper function to get the base directory path before the file name, but after the port number. Expects output of <code>request.getRequestURI</code> as input.</p> <pre>public static java.lang.String getRootPath(java.lang.String rootPath)</pre>
<code>getRootUrl(String)</code>	<p>Helper function to get the base URL from the server specification retrieved from <code>getLaunchDetails</code>.</p> <pre>public static java.lang.String getRootUrl(java.lang.String servletUrl)</pre>
<code>getTerminateTag(String, String)</code>	<p>Helper function to extract values from a terminate message. Finds the given tag and returns the value for that tag in the XML message string.</p> <pre>public static java.lang.String getTerminateTag(java.lang.String xmlMsg, java.lang.String tagName)</pre>

addConfigToQuote(ControlRecord, BigDecimal, BigDecimal, BigDecimal, BigDecimal, String, String)

```
public static void addConfigToQuote(oracle.apps.aso.quote.ControlRecord
controlRec, java.math.BigDecimal quoteHdrId,
java.math.BigDecimal quoteLnId, java.math.BigDecimal configHdrId,
java.math.BigDecimal configRevNum, java.lang.String validCfg,
java.lang.String completeCfg)
throws FrameworkException, SQLException, QuoteException
```

Adds the child items of a configured product to the quote. This API should be called within a transaction block.

Parameters:

controlRec - The standard control record for updating a quote.

quoteHdrId - Quote header ID.

quoteLnId - Quote line ID.

configHdrId - Configuration header ID. Returned from Configurator terminate message.

ConfigRevNum - Configuration revision number. Returned from Configurator terminate message.

validCfg - Whether the configuration is valid. Returned from Configurator terminate message.

completeCfg - Whether the configuration is valid. Returned from Configurator terminate message.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteException - If an expected error occurs.

addConfigToQuote(ControlRecord, QuoteHeader, BigDecimal, BigDecimal, BigDecimal, String, String)

```
public static void addConfigToQuote(oracle.apps.aso.quote.ControlRecord
controlRec, oracle.apps.qot.core.QuoteHeader quoteHdr,
java.math.BigDecimal quoteLnId, java.math.BigDecimal configHdrId,
java.math.BigDecimal configRevNum, java.lang.String validCfg,
java.lang.String completeCfg)
throws FrameworkException, SQLException, QuoteException
```

Adds the child items of a configured model item to a quote. This API should be called within a transaction block.

Parameters:

controlRec - The standard control record for updating a quote.

quoteHdr - The header information of the quote, where the configuration items will be added.

quoteLnId - The quote line ID of the configured model item.

configHdrId - The configuration header ID. This value may be retrieved from the Configurator termination message.

`configRevNum` - The configuration revision number. This value may be retrieved from the Configurator termination message.

`validCfg` - Indicating whether the configuration is valid. true if the configuration is valid, false otherwise. This value may be retrieved from the Configurator termination message.

`completeCfg` - Indicating whether the configuration is complete. True if the configuration is complete, false otherwise. This value may be retrieved from the Configurator termination message.

Throws:

`java.sql.SQLException` - If a database error occurs.

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`QuoteException` - If an application error occurs.

getInitTag(String, String)

```
public static java.lang.String getInitTag(java.lang.String xmlMsg,
java.lang.String tagName)
```

Extracts the value for the given tag from a Configurator initialization message in XML format. This is a utility method to find the given XML tag and return the value for that tag in the XML message string.

Parameters:

`xmlMsg` - Entire initialize message string in XML format; received on successful return from the Configurator.

`tagName` - Name of the tag to return a value for.

Returns: A String value for the tag in the XML initialize message.

getLaunchInfo(BigDecimal, BigDecimal)

```
public static oracle.apps.got.core.ConfigLaunchInfo
getLaunchInfo(java.math.BigDecimal invItemId, java.math.BigDecimal invOrgId)
throws SQLException, FrameworkException, QuoteException
```

Returns a `ConfigLaunchInfo` object containing detailed Configurator launching information retrieved from the database, given the inventory item information. The Configurator launching information may be used to construct the initialization message for a configuration session.

Parameters:

`invItemId` - The inventory item ID of the model item to be configured.

`invOrgId` - The inventory organization ID of the model item to be configured.

Returns: A `ConfigLaunchInfo` object containing the information required to launch the Configurator.

Throws:

`java.sql.SQLException` - If a database error occurs.

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`QuoteException` - If an application error occurs.

getRootPath(String)

```
public static java.lang.String getRootPath(java.lang.String rootPath)
```

Extracts the base directory string from the given URI path string. This is a utility method to retrieve the base directory path, which is located before the file name but after the port number, from a URI path string. It expects the output of `request.getRequestURI()` as input.

For example:

Input string: `/html/US/file.jsp`

Output string: `/html/US/`

Parameters: `rootPath` - A URI path string, e.g., the output of `request.getRequestURI()`.

Returns: The base directory path in a URI path string.

getRootUrl(String)

```
public static java.lang.String getRootUrl(java.lang.String servletUrl)
```

Extracts the base URL from the given URL. This is a utility method to retrieve the base URL string from a full URL string. It expects the servlet URL of the Configurator, which may be obtained from the `ConfigLaunchInfo` object returned from `API getLaunchInfo()`, as input.

For example:

Input string:

`http://machine.domain.com:12345/configurator/oracle.apps.cz.servlet.UiServlet`

Output string: `http://machine.domain.com:12345/`.

Parameters: `servletUrl` - The full URL string, e.g., the servlet URL of the Configurator.

Returns: The base URL string which specifies protocol, machine name, domain and port number, e.g., `http://machine.domain.com:12345/`.

getTerminateTag(String, String)

```
public static java.lang.String getTerminateTag(java.lang.String xmlMsg,  
java.lang.String tagName)
```

Extracts the value for the given tag from a Configurator termination message in XML format. This is a utility method to find the given XML tag and return the value for that tag in the given XML message string.

Parameters:

xmlMsg - The entire termination message in XML format received in return from the Configurator after a configuration session terminates.

tagName - The name of the tag for which to return the value.

Returns: The value for the given tag in the Configurator termination message in XML format.

1.7 Class Contract

```
java.lang.Object  
|  
+--oracle.apps.got.core.Contract
```

public class Contract

Contract contains methods retrieve contract information, create contract, and update contract.

Table 1–11 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

1.7.1 Fields for Class Contract

ACTIVE

```
public static final int ACTIVE
```

Indicates contract state is **Active**.

APPROVED

```
public static final int APPROVED
```

Indicates contract state is **Approved**.

CANCELLED

```
public static final int CANCELLED
```

Indicates contract state is **Cancelled**.

comments

```
public java.lang.String comments
```

Contract comments.

contractId

```
public java.math.BigDecimal contractId
```

Contract ID.

contractNumber

```
public java.lang.String contractNumber
```

Contract number.

contractNumberModifier

```
public java.lang.String contractNumberModifier
```

Contract number modifier.

dateApproved

```
public java.sql.Timestamp dateApproved
```

Date contract was approved.

dateSigned

```
public java.sql.Timestamp dateSigned
```

Date contract was signed.

dateTerminated

```
public java.sql.Timestamp dateTerminated
```

Date contract was terminated.

DEEP

```
public static final int DEEP
```

Indicates deep load of the contract should be performed. The following contract information should be loaded: `contractId`, `contractNumber`, `contractNumberModifier`, `state`, `shortDescription`, `dateApproved`, `dateSigned`, `dateTerminated`, `startDate`, `endDate`, `description`, `comments`.

description

public java.lang.String **description**
Contract description.

endDate

public java.sql.Timestamp **endDate**
Contract end date.

ENTERED

public static final int **ENTERED**
Indicates contract state is **Entered**.

EXPIRED

public static final int **EXPIRED**
Indicates contract state is **Expired**.

HOLD

public static final int **HOLD**
Indicates contract state is **Hold**.

quoteld

public java.math.BigDecimal **quoteId**
Quote header ID.

RCS_ID

public static final java.lang.String **RCS_ID**
Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

public static final boolean **RCS_ID_RECORDED**
Standard public final static boolean which is initialized by a call to oracle.apps.fnd.common.VersionInfo.recordClassVersion.

SHALLOW

public static final int **SHALLOW**
Indicates shallow load of the contract should be performed. The following contract information should be loaded: contractId, contractNumber.

shortDescription

```
public java.lang.String shortDescription
```

Contract short description.

SIGNED

```
public static final int SIGNED
```

Indicates contract state is **Signed**.

startDate

```
public java.sql.Timestamp startDate
```

Contract start date.

state

```
public int state
```

Contract state. Possible values are: **Entered, Approved, Signed, Active, Cancelled, Terminated, Hold, Expired**.

TERMINATED

```
public static final int TERMINATED
```

Indicates contract status is **Terminated**.

1.7.2 Constructors for Class Contract

Contract()

```
public Contract()
```

1.7.3 Methods for Class Contract

The following table is an index of the Class Contract methods:

Table 1–12 Methods for Class Contract

Method	Description
<code>createContract(BigDecimal, BigDecimal, BigDecimal, String, String, String, BigDecimal, BigDecimal)</code>	<p>Creates a contract using the specified quote. This API should be called within a transaction block.</p> <pre>public static oracle.apps.qot.core.Contract createContract(java.math.BigDecimal quoteId, java.math.BigDecimal templateId, java.math.BigDecimal templateVersion, java.lang.String relationType, java.lang.String interactionSubject, java.lang.String interactionBody, java.math.BigDecimal partyId, java.math.BigDecimal resourceId)</pre> <p>throws <code>FrameworkException</code>, <code>SQLException</code>, <code>ContractException</code></p>
<code>getContract(BigDecimal, int)</code>	<p>Returns the contract associated with this quote. If no contract is associated, then it returns null.</p> <pre>public static oracle.apps.qot.core.Contract getContract(java.math.BigDecimal quoteId, int mode)</pre> <p>throws <code>FrameworkException</code>, <code>SQLException</code></p>
<code>getContractText(BigDecimal, BigDecimal)</code>	<p>Returns the text of the articles contained in the specified contract. For performance reasons, a reader stream (one for each article) is returned. After use, each reader stream should be closed by the calling application.</p> <pre>public static java.io.Reader getContractText(java.math.BigDecimal contractId, java.math.BigDecimal version)</pre> <p>throws <code>FrameworkException</code>, <code>SQLException</code>, <code>ContractException</code></p>
<code>toString()</code>	<p>Returns String representation of the contract object.</p> <pre>public java.lang.String toString()</pre>

Table 1–12 Methods for Class Contract

Method	Description
updateContract(BigDecimal, String, String, BigDecimal, BigDecimal)	<p>Updates a contract. This API should be called within a transaction block.</p> <pre>public static void updateContract(java.math.BigDecimal quoteId, java.lang.String interactionSubject, java.lang.String interactionBody, java.math.BigDecimal partyId, java.math.BigDecimal resourceId) throws FrameworkException, SQLException, ContractException</pre>

createContract(BigDecimal, BigDecimal, BigDecimal, String, String, String, BigDecimal, BigDecimal)

```
public static oracle.apps.got.core.Contract
createContract( java.math.BigDecimal quoteId,
java.math.BigDecimal templateId, java.math.BigDecimal templateVersion,
java.lang.String relationType, java.lang.String interactionSubject,
java.lang.String interactionBody, java.math.BigDecimal partyId,
java.math.BigDecimal resourceId)
throws FrameworkException, SQLException, ContractException
```

Creates a contract using the specified quote. This API should be called within a transaction block.

Parameters:

quoteId - Quote header ID.

TemplateId - Contract template ID.

templateVersion - Contract template version.

relationType - Relationship type.

interactionSubject - Interaction subject.

interactionBody - Interaction body.

partyId - Party ID.

resourceId - Resource ID.

Returns: Contract object with contract ID and contract number of the created contract.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

ContractException - If an error occurs while creating the contract.

getContract(BigDecimal, int)

```
public static oracle.apps.qot.core.Contract getContract(java.math.BigDecimal  
quoteId, int mode)
```

throws FrameworkException, SQLException

Returns the contract associated with this quote. If no contract is associated, then it returns null.

Parameters:

quoteId - Quote header ID.

mode - Level indicating amount of contract information to retrieve. Possible values: **SHALLOW, DEEP**.

Returns: Contract associated with the quote header ID passed in as parameter.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException -If an error occurs while getting connection.

getContractText(BigDecimal, BigDecimal)

```
public static java.io.Reader getContractText(java.math.BigDecimal  
contractId, java.math.BigDecimal version)
```

throws FrameworkException, SQLException, ContractException

Returns the text of the articles contained in the specified contract. For performance reasons, a reader stream (one for each article) is returned. After use, each reader stream should be closed by the calling application.

Parameters:

contractId - Contract ID.

version - Contract version.

Returns: Reader stream containing text of the contract articles.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

ContractException - If an expected application error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the contract object.

Overrides: toString in class Object

Returns: A String representation of the contract object.

updateContract(BigDecimal, String, String, BigDecimal, BigDecimal)

```
public static void updateContract(java.math.BigDecimal quoteId,
    java.lang.String interactionSubject, java.lang.String interactionBody,
    java.math.BigDecimal partyId, java.math.BigDecimal resourceId)
    throws FrameworkException, SQLException, ContractException
```

Updates a contract. This API should be called within a transaction block.

Parameters:

quoteId - Quote header ID.

interactionSubject - Interaction subject.

interactionBody - Interaction body.

partyId - Party ID.

resourceId - Resource ID.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

ContractException - If an application error occurs.

1.8 Class ContractException

```
java.lang.Object
|
+--java.lang.Throwable
    |
    +--java.lang.Exception
        |
        +--oracle.apps.jtf.base.resources.FrameworkException
            |
            +--oracle.apps.qot.core.ContractException
```

All Implemented Interfaces: java.io.Serializable

```
public class ContractException extends
    oracle.apps.jtf.base.resources.FrameworkException
```

ContractException is thrown when an application error occurs in a Contract method.

Table 1–13 Inherited Member Summary

Fields inherited from interface FrameworkException

DEBUG, ERROR, FATAL, INFORMATION, NONE, WARNING, defaultMsgMgr

Methods inherited from interface FrameworkException

addException(Exception), convertException(Exception), getAllInfo(),
getCurrentMessageManager(), getExceptionStack(), getExceptionStackRec(),
getExternException(), getKey(), getMessage(), getMessageManager(), getMessageStack(),
getParameters(), getParentExcep(), getRootException(), getRootExternExcept(),
getSeverity(), getThrowerInfo(), getWholeStack(), printAllInfo(PrintStream),
printAllInfo(PrintStream), printMesg(PrintStream), printMesg(PrintStream),
printMessageStack(PrintWriter), printMessageStack(PrintWriter),
printStackTrace(PrintWriter), printStackTrace(PrintWriter),
printThrowerInfo(PrintStream), printThrowerInfo(PrintStream),
printWholeStack(PrintStream), printWholeStack(PrintStream), setCurrents(),
setStackTrace(String)

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long,
int), wait(long, int)

Methods inherited from class Throwable

fillInStackTrace(), getLocalizedMessage(), printStackTrace(), toString()

1.8.1 Fields for Class ContractException

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.8.2 Constructors for Class ContractException

ContractException(Exception, String)

```
public ContractException(java.lang.Exception e, java.lang.String errorKey)
```

Construct an Exception with the given exception and errorKey

Parameters:

e - The parent exception.

errorKey - Error key.

ContractException(Exception, String, Object[])

```
public ContractException(java.lang.Exception e, java.lang.String errorKey,  
java.lang.Object[] params)
```

Construct an Exception with the given exception, errorKey, and parameters

Parameters:

e - The parent exception.

errorKey - Error key.

params - An array of tokens for errorKey.

ContractException(Exception, String, String)

```
public ContractException(java.lang.Exception e, java.lang.String errorKey,  
java.lang.String param)
```

Construct an Exception with the given exception, errorKey, and parameter

Parameters:

e - The parent exception

errorKey - Error key.

param - Token for the error key.

ContractException(int, String)

```
public ContractException(int err_msg_count, java.lang.String errorKey)  
throws FrameworkException
```

Construct an Exception with the message count and error key. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

ContractException(int, String, Object[])

```
public ContractException(int err_msg_count, java.lang.String errorKey,  
java.lang.Object[] params)  
throws FrameworkException
```

Construct an Exception with the message count, error key, and parameter tokens. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

params - An array of tokens for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

ContractException(int, String, String)

```
public ContractException(int err_msg_count, java.lang.String errorKey,  
java.lang.String param)  
throws FrameworkException
```

Construct an Exception with the message count, error key, and parameter token. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

param - A token for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

ContractException(String)

```
public ContractException(java.lang.String errorKey)
```

Construct an exception with the errorKey.

Parameters: errorKey - Error key.

ContractException(String, Object[])

```
public ContractException(java.lang.String errorKey,  
java.lang.Object[] params)
```


Construct an exception with the errorKey and parameters.

Parameters:

errorKey - Error key.

params - An array of tokens for errorKey.

ContractException(String, String)

```
public ContractException(java.lang.String err_msg,
    java.lang.String errorKey)
```

Construct an Exception with the error message and error key.

Parameters:

err_msg - Error message.

errorKey - Error key.

ContractException(String, String, Object[])

```
public ContractException(java.lang.String err_msg,
    java.lang.String errorKey, java.lang.Object[] params)
```

Construct an Exception with the error message, error key, and parameter tokens.

Parameters:

err_msg - Error message.

errorKey - Error key.

params - An array of tokens for errorKey.

ContractException(String, String, String)

```
public ContractException(java.lang.String err_msg,
    java.lang.String errorKey, java.lang.String param)
```

Construct an Exception with the error message, error key, and parameter token.

Parameters:

err_msg - Error message.

errorKey - Error key.

param - Token for the error key.

1.9 Class ContractTemplate

```
java.lang.Object
|
+--oracle.apps.qot.core.ContractTemplate
```

public class ContractTemplate

ContractTemplate contains methods to retrieve the default contract template and retrieve the list of available contract templates. This object contains the following information for a contract template: ID, short description, contract number, contract number modifier, major version

Table 1–14 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

1.9.1 Fields for Class ContractTemplate

contractNumber

public java.lang.String **contractNumber**
Contract template number.

contractNumberModifier

public java.lang.String **contractNumberModifier**
Contract template number modifier.

id

public java.math.BigDecimal **id**
Contract template ID.

majorVersion

public java.math.BigDecimal **majorVersion**
Contract template major version.

RCS_ID

public static final java.lang.String **RCS_ID**
Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

public static final boolean **RCS_ID_RECORDED**
Standard public final static boolean which is initialized by a call to oracle.apps.fnd.common.VersionInfo.recordClassVersion.

shortDescription

```
public java.lang.String shortDescription
Contract template short description.
```

1.9.2 Constructors for Class ContractTemplate**ContractTemplate()**

```
public ContractTemplate()
Default constructor.
```

ContractTemplate(BigDecimal, String, String, String, BigDecimal)

```
public ContractTemplate(java.math.BigDecimal _id, java.lang.String _
shortDescription, java.lang.String _contractNumber, java.lang.String _
contractNumberModifier, java.math.BigDecimal _majorVersion)
Constructor.
```

Parameters:

_id - Contract template ID.

_shortDescription - Contract template short description.

_contractNumber - Contract template number.

_contractNumberModifier - Contract template number modifier.

_majorVersion - Contract template major version.

1.9.3 Methods for Class ContractTemplate

The following table is an index of the Class ContractTemplate methods:

Table 1–15 Methods for Class ContractTemplate

Method	Description
getDefault()	Returns the default contract template. <pre>public static oracle.apps.qot.core.ContractTemplate getDefault()throws FrameworkException, SQLExce ption</pre>

Table 1–15 Methods for Class ContractTemplate

Method	Description
<code>getTemplates()</code>	Returns a list of available contract templates. <pre>public static oracle.apps.got.core.ContractTemplate[] getTemplates() throws FrameworkException, SQLException</pre>
<code>toString()</code>	Returns a String representation of the contract template object. <pre>public java.lang.String toString()</pre>

getDefault()

```
public static oracle.apps.got.core.ContractTemplate getDefault()
throws FrameworkException, SQLException
```

Returns the default contract template.

Returns: The default contract template.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

getTemplates()

```
public static oracle.apps.got.core.ContractTemplate[] getTemplates()
throws FrameworkException, SQLException
```

Returns a list of available contract templates.

Returns: Array of ContractTemplate object.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting a connection.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the contract template object.

Overrides: toString in class Object

Returns: A String representation of the contract template object.

1.10 Class InstallBaseExtAttr

```

java.lang.Object
|
+--oracle.apps.qot.core.InstallBaseExtAttr

public class InstallBaseExtAttr

```

The InstallBaseExtAttr object is modeled to provide details of the extended attributes associated with the install base item.

The details for the attributes of the item are: Attribute ID, Attribute name.

Table 1–16 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

1.10.1 Fields for Class InstallBaseExtAttr

ATTRIBUTE_NAME

```
public static final java.lang.String ATTRIBUTE_NAME
```

Criteria to search Install Base item attributes.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.10.2 Constructors for Class InstallBaseExtAttr

InstallBaseExtAttr()

```
public InstallBaseExtAttr()
```

Default constructor.

1.10.3 Methods for Class InstallBaseExtAttr

Table 1–17 *Methods for Class InstallBaseExtAttr*

Method	Description
getAccessLevel()	Returns access level of the Install Base item attribute. <code>public java.lang.String getAccessLevel()</code>
getAccessValue()	Returns access value of the Install Base item attribute. <code>public java.lang.String getAccessValue()</code>
getAttributeId()	Returns the attribute ID of the Install Base item attribute. <code>public java.math.BigDecimal getAttributeId()</code>
getAttributeName()	Returns attribute name of the Install Base item attribute. <code>public java.lang.String getAttributeName()</code>
list(HashMap, int, int, int)	Lists extended attributes defined for Install Base items based on the search criteria passed in as parameter. Extended attributes are defined at levels - Global, Category, Instance, Item . Returns attribute ID, attribute name, access level and access name of the attribute. <code>public static oracle.apps.qot.util.QueryResultSet list(com.sun.java.util.collections.HashMap searchCriteria, int batchSize, int startIndex, int resCount) throws FrameworkException, SQLException</code>
toString()	Returns a String representation of the InstallBaseExtAttr object. <code>public java.lang.String toString()</code>

getAccessLevel()

```
public java.lang.String getAccessLevel()
```

Returns access level of the Install Base item attribute.

Returns: Access level of the Install Base item attribute.

getAccessValue()

```
public java.lang.String getAccessValue()
```

Returns access value of the Install Base item attribute.

Returns: Access value of the Install Base item attribute.

getAttributeId()

```
public java.math.BigDecimal getAttributeId()
```

Returns the attribute ID of the Install Base item attribute.

Returns: Attribute ID of the Install Base item attribute.

getAttributeName()

```
public java.lang.String getAttributeName()
```

Returns attribute name of the Install Base item attribute.

Returns: Attribute name of the Install Base item attribute.

list(HashMap, int, int, int)

```
public static oracle.apps.qot.util.QueryResultSet
```

```
list(com.sun.java.util.collections.HashMap searchCriteria, int batchSize,  
int startIndex, int resCount)
```

```
throws FrameworkException, SQLException
```

Lists extended attributes defined for Install Base items based on the search criteria passed in as parameter.

Extended attributes are defined at levels - **Global, Category, Instance, Item**. Returns attribute ID, attribute name, access level and access name of the attribute.

Parameters:

searchCriteria - Criteria for search. HashMap having: key = ATTRIBUTE_NAME, value =attribute name.

batchSize - Batch size for the query.

startIndex - Start index indicating which row number to start retrieving data from. -1 indicates the last rows.

resCount - The total number of objects to return. If -1, then count will be queried from the database.

Returns: QueryResultSet containing an array of InstallBaseExtAttr objects for search criteria passed in.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the InstallBaseExtAttr object.

Overrides: toString in class Object

Returns: A String representation of InstallBaseExtAttr object.

1.11 Class InstallBaseItem

```
java.lang.Object
|
+--oracle.apps.qot.core.InstallBaseItem
```

public class **InstallBaseItem**

InstallBaseItem contains methods to retrieve information for an install base instance and to search for install base instances.

The InstallBaseItem object is used to model the instance of an ordered inventory item in customer's Install Base. It holds the following information about the item: unique instance ID, primary unit of measure code, uom, serial number, quantity, system name, item ID, inventory org ID, description, part number, BOM item type, ordered date, installed at address.

Table 1–18 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

1.11.1 Fields for Class InstallBaseItem

ADD_SERVICE

```
public static final int ADD_SERVICE
```

Indicates search for Install Base instances for which to purchase services. Used as input to InstallBaseItem.list() method.

ADD_TO_CONTAINER

```
public static final int ADD_TO_CONTAINER
```

Indicates search for Install Base instances of models that can be added to container model in a quote. Used as input to InstallBaseItem.list() method.

ATTRIBUTE_ID

```
public static final java.lang.String ATTRIBUTE_ID
```


Indicates search for Install Base instances with the attribute ID. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

ATTRIBUTE_VALUE

```
public static final java.lang.String ATTRIBUTE_VALUE
```

Indicates search for Install Base instances with the attribute value. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

CATEGORY

```
public static final java.lang.String CATEGORY
```

Indicates search in an inventory category. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

CATEGORY_SET

```
public static final java.lang.String CATEGORY_SET
```

Indicates search in an inventory category set. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

COMPONENT_OF

```
public static final java.lang.String COMPONENT_OF
```

Indicates Component-Of instance relationship when retrieving Install Base instance relationships.

CONNECTED_TO

```
public static final java.lang.String CONNECTED_TO
```

Indicates Connected-To instance relationship when retrieving Install Base instance relationships.

CONTAINER_MODEL_ID

```
public static final java.lang.String CONTAINER_MODEL_ID
```

Indicates search for Install Base instance models that can be added to the container model. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

CUST_ACCT_ID

```
public static final java.lang.String CUST_ACCT_ID
```

Indicates search for Install Base instances for a customer account. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

INSTALL_CITY

```
public static final java.lang.String INSTALL_CITY
```

Indicates search for Install Base instances installed in the city. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

INSTALL_COUNTRY

```
public static final java.lang.String INSTALL_COUNTRY
```

Indicates search for Install Base instances installed in the country. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

INSTANCE_NAME

```
public static final java.lang.String INSTANCE_NAME
```

Indicates search for Install Base instances for an instance name. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

INSTANCE_NUMBER

```
public static final java.lang.String INSTANCE_NUMBER
```

Indicates search for Install Base instances for an instance number. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

ITEM_DESC

```
public static final java.lang.String ITEM_DESC
```

Indicates search for an inventory item description. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

ITEM_PART_NUM

```
public static final java.lang.String ITEM_PART_NUM
```

Indicates search for an inventory item part number. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

ORDER_DATE_FROM

```
public static final java.lang.String ORDER_DATE_FROM
```

Indicates search for orders placed on or after a specific order date. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

ORDER_DATE_TO

```
public static final java.lang.String ORDER_DATE_TO
```

Indicates search for orders placed on or before a specific order date. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

ORDER_NUMBER

```
public static final java.lang.String ORDER_NUMBER
```

Indicates search for Install Base instances created in this order. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

RECONFIGURE

```
public static final int RECONFIGURE
```

Indicates search for Install Base instances of models that can be reconfigured. Used as input to `InstallBaseItem.list()` method.

SERIAL_NUM

```
public static final java.lang.String SERIAL_NUM
```

Indicates search for an Install Base serial number. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

SYSTEM

```
public static final java.lang.String SYSTEM
```

Indicates search for an Install Base system name. Used as the key for an entry in the HashMap for the Install Base search criteria input in method `InstallBaseItem.list()`.

TRADE_IN

```
public static final int TRADE_IN
```

Indicates search for Install Base instances to trade-in (return). Used as input to InstallBaseItem.list() method.

1.11.2 Constructors for Class InstallBaseltem

InstallBaseltem()

```
public InstallBaseItem()
```

Default constructor.

InstallBaseltem(BigDecimal)

```
public InstallBaseItem(java.math.BigDecimal instanceId)
```

Constructor for an InstallbaseItem for given instance ID.

Parameters:

instanceId - Instance ID of the Install Base item.

InstallBaseltem(BigDecimal, String, BigDecimal, String, String, String, BigDecimal, BigDecimal, String, String, int, Timestamp, String, String, String, String, String, String, String, String)

```
public InstallBaseItem(java.math.BigDecimal instanceId,  
java.lang.String serialNumber, java.math.BigDecimal quantity,  
java.lang.String uomCode, java.lang.String uom, java.lang.String systemName,  
java.math.BigDecimal itemId, java.math.BigDecimal invOrgId,  
java.lang.String description, java.lang.String partNumber, int bomItemType,  
java.sql.Timestamp orderedDate, java.lang.String installAddress1,  
java.lang.String installAddress2, java.lang.String installAddress3,  
java.lang.String installAddress4, java.lang.String installCity,  
java.lang.String installCounty, java.lang.String installPostalCode,  
java.lang.String installState, java.lang.String installProvince,  
java.lang.String installCountry)
```

Constructor for Install Base item.

Parameters:

instanceId - Instance ID of the Install Base item.

serialNumber - Serial number of the Install Base item.

quantity - Quantity of the Install Base item.

uomCode - UOM code of the Install Base item.

uom - UOM of the Install Base item.

systemName - System name of the Install Base item.

itemId - Item ID of the Install Base item.
 invOrgId - Inventory org ID of the Install Base item.
 description - Description of the Install Base item.
 partNumber - Part number of the Install Base item.
 bomItemType - BOM item type of the Install Base item.
 orderedDate - Ordered date of the Install Base item.
 installAddress1 - Installed at Address1 of the Install Base item.
 installAddress2 - Installed at Address2 of the Install Base item.
 installAddress3 - Installed at Address3 of the Install Base item.
 installAddress4 - Installed at Address4 of the Install Base item.
 installCity - Installed at city of the Install Base item.
 installCountry - Installed at county of the Install Base item.
 installPostalCode - Installed at postal code of the Install Base item.
 installState - Installed at state of the Install Base item.
 installProvince - Installed at province of the Install Base item.
 installCountry - Installed at country of the Install Base item.

1.11.3 Methods for Class InstallBaseItem

The following table is an index of the Class InstallBaseItem methods:

Table 1–19 Methods for Class InstallBaseItem

Method	Description
getBomItemType()	Returns BOM item type of Install Base item. <code>public int getBomItemType()</code>
getDepth()	Returns the depth of the Install Base item in an Install Base item tree. <code>public int getDepth()</code>
getDescription()	Returns description of Install Base item. <code>public java.lang.String getDescription()</code>

Table 1–19 Methods for Class InstallBaseItem

Method	Description
<code>getExtendedAttributes()</code>	Returns the extended attributes of the instance in Install Base. Instance ID should be set in the InstallBaseItem before calling this API. <pre>public com.sun.java.util.collections.HashMap getExtendedAttributes() throws FrameworkException, SQLException, FrameworkException</pre>
<code>getInstallAddress1()</code>	Returns installed at address1 for install base item. <pre>public java.lang.String getInstallAddress1()</pre>
<code>getInstallAddress2()</code>	Returns installed at address2 for install base item. <pre>public java.lang.String getInstallAddress2()</pre>
<code>getInstallAddress3()</code>	Returns installed at address3 for install base item. <pre>public java.lang.String getInstallAddress3()</pre>
<code>getInstallAddress4()</code>	Returns installed at address4 for install base item. <pre>public java.lang.String getInstallAddress4()</pre>
<code>getInstallAtAddr()</code>	Returns a String concatenation of installed at address fields. <pre>public java.lang.String getInstallAtAddr()</pre>
<code>getInstallCity()</code>	Returns installed at city for install base item. <pre>public java.lang.String getInstallCity()</pre>
<code>getInstallCountry()</code>	Returns installed at country for install base item. <pre>public java.lang.String getInstallCountry()</pre>
<code>getInstallCounty()</code>	Returns installed at county for install base item. <pre>public java.lang.String getInstallCounty()</pre>
<code>getInstallPostalCode</code>	Returns installed at PostalCode for install base item. <pre>public java.lang.String getInstallPostalCode()</pre>

Table 1–19 Methods for Class InstallBaseItem

Method	Description
<code>getInstallProvince()</code>	Returns installed at province for Install Base item. <code>public java.lang.String getInstallProvince()</code>
<code>getInstallState()</code>	Returns installed at state for Install Base item. <code>public java.lang.String getInstallState()</code>
<code>getInstanceId()</code>	Returns the instance ID of the particular Install Base item. <code>public java.math.BigDecimal getInstanceId()</code>
<code>getInstanceName()</code>	Returns the instance name of the Install Base item. <code>public java.lang.String getInstanceName()</code>
<code>getInstanceNumber()</code>	Returns the instance number of the Install Base item. <code>public java.lang.String getInstanceNumber()</code>
<code>getInvOrgId()</code>	Returns the inventory org ID of the Install Base item. <code>public java.math.BigDecimal getInvOrgId()</code>
<code>getItemId()</code>	Returns the item ID of the Install Base item. <code>public java.math.BigDecimal getItemId()</code>
<code>getNodeType</code>	Returns the node type of the Install Base item in an Install Base item tree <code>public int getNodeType()</code>
<code>getOrderedDate()</code>	Returns ordered date of Install Base item. <code>public java.sql.Timestamp getOrderedDate()</code>
<code>getOrderNumber</code>	Returns the order number of the Install Base Item. <code>public java.math.BigDecimal getOrderNumber()</code>
<code>getParentId</code>	Returns the parent ID of the Install Base item. <code>public java.math.BigDecimal getParentId()</code>
<code>getPartNumber()</code>	Returns Install Base item part number. <code>public java.lang.String getPartNumber()</code>

Table 1–19 Methods for Class InstallBaseItem

Method	Description
<code>getQuantity()</code>	Returns the quantity of the Install Base item. public java.math.BigDecimal getQuantity()
<code>getRelatedInstances(String, boolean)</code>	Lists related Install Base items for relationship type and depth specified. public oracle.apps.qot.core.InstallBaseItem[] getRelatedInstances (java.lang.String relationshipType, boolean fullDepth) throws FrameworkException, SQLException
<code>getRelatedInstances(String, HashMap, int, int)</code>	Lists Install Base items that are related to the Install Base item in an Install Base item tree, given the relationship type, the list of items to be expanded in the tree. The returning array of Install Base items will contain the item itself (the root of the tree). public oracle.apps.qot.core.InstallBaseItem[] getRelatedInstances (java.lang.String relationshipType, com.sun.java.util.collections.HashMap expa ndList, int loadDepth, int loadStructure) throws FrameworkException, SQLException
<code>getSerialNumber()</code>	Returns serial number of the item in InstallBase. public java.lang.String getSerialNumber()
<code>getSystemName()</code>	Returns the system name associated with the Install Base item. public java.lang.String getSystemName()
<code>getUom()</code>	Returns primary unit of measure for the InstallBase item. public java.lang.String getUom()
<code>getUomCode()</code>	Returns primary unit of measure code for the InstallBase item. public java.lang.String getUomCode()

Table 1–19 Methods for Class InstallBaseItem

Method	Description
<code>list(int, BigDecimal, HashMap, int, int, int)</code>	<p>Lists Install Base items based on search criteria passed in as parameter.</p> <pre>public static oracle.apps.qot.util.QueryResultSet list(int type, java.math.BigDecimal invOrgId, com.sun.java.util.collections.HashMap searchCriteria, int batchSize, int startIndex, int resCount) throws FrameworkException, SQLException</pre>
<code>load(BigDecimal)</code>	<p>Returns InstallBaseItem object corresponding to the instance ID passed in as parameter.</p> <pre>public static oracle.apps.qot.core.InstallBaseItem load(java.math.BigDecimal instanceId) throws FrameworkException, SQLException</pre>
<code>setBomItemType(int)</code>	<p>Sets the value for BOM item type.</p> <pre>public void setBomItemType(int bomItemType)</pre>
<code>setDepth(int)</code>	<p>Sets node depth for the Install Base item to be displayed in a tree.</p> <pre>public void setDepth(int depth)</pre>
<code>setDescription(String)</code>	<p>Sets the value for description.</p> <pre>public void setDescription(java.lang.String description)</pre>
<code>setInstallAddress1(String)</code>	<p>Sets the value for install at Address1.</p> <pre>public void setInstallAddress1(java.lang.String installAddress1)</pre>
<code>setInstallAddress2(String)</code>	<p>Sets the value for install at Address2.</p> <pre>public void setInstallAddress2(java.lang.String installAddress2)</pre>

Table 1–19 Methods for Class InstallBaseltem

Method	Description
setInstallAddress3(String)	Sets the value for install at Address3. public void setInstallAddress3 (java.lang.String installAddress3)
setInstallAddress4(String)	Sets the value for install at Address4. public void setInstallAddress4 (java.lang.String installAddress4)
setInstallCity(String)	Sets the value for install at city. public void setInstallCity (java.lang.String installCity)
setInstallCountry(String)	Sets the value for install at country. public void setInstallCountry (java.lang.String installCountry)
setInstallCounty(String)	Sets the value for install at county. public void setInstallCounty (java.lang.String installCounty)
setInstallPostalCode(String)	Sets the value for install at postal code. public void setInstallPostalCode (java.lang.String installPostalCode)
setInstallProvince(String)	Sets the value for install at province. public void setInstallProvince (java.lang.String installProvince)
setInstallState(String)	Sets the value for install at state. public void setInstallState (java.lang.String installState)
setInstanceId(BigDecimal)	Sets the value for the instance ID. public void setInstanceId (java.math.BigDecimal instanceId)

Table 1–19 Methods for Class InstallBaseItem

Method	Description
setInstanceName(String)	Sets the value for instance name. public void setInstanceName (java.lang.String instanceName)
setInstanceNumber(String)	Sets the value for instance number. public void setInstanceNumber (java.lang.String instanceNumber)
setInvOrgId(BigDecimal)	Sets the value for the inventory organization ID. public void setInvOrgId (java.math.BigDecimal invOrgId)
setItemId(BigDecimal)	Sets the value for the item ID. public void setItemId (java.math.BigDecimal itemId)
setNodeType(int)	Sets the node type for the Install Base item to be displayed in a tree. public void setNodeType (int nodeType)
setOrderedDate(Timestamp)	Sets the value for ordered date of item. public void setOrderedDate (java.sql.Timestamp orderedDate)
setOrderNumber(BigDecimal)	Sets the value for order number of item. public void setOrderNumber (java.math.BigDecimal orderNumber)
setParentId(BigDecimal)	Sets the parent instance ID for this Install Base item public void setParentId (java.math.BigDecimal parentId)
setPartNumber(String)	Sets the value for part number. public void setPartNumber (java.lang.String partNumber)
setQuantity(BigDecimal)	Sets the value for the quantity. public void setQuantity (java.math.BigDecimal quantity)

Table 1–19 Methods for Class InstallBaseltem

Method	Description
setSerialNumber(String)	Sets the value for serial number. public void setSerialNumber (java.lang.String serialNumber)
setSystemName(String)	Sets the value for system name. public void setSystemName (java.lang.String systemName)
setUom(String)	Sets the value for primary uom. public void setUom (java.lang.String uom)
setUomCode(String)	Sets the value for primary uom code. public void setUomCode (java.lang.String uomCode)
toString()	Returns a String representation of Install Base instance. public java.lang.String toString ()

getBomItemType()public int **getBomItemType**()

Returns BOM item type of the Install Base item.

Returns: BOM item type of the Install Base item.**getDepth()**public int **getDepth**()

Returns the depth of the Install Base item in an Install Base item tree.

Returns: The depth of the Install Base item in an Install Base item tree.**getDescription()**public java.lang.String **getDescription**()

Returns description of the Install Base item.

Returns: Description of the Install Base item.**getExtendedAttributes()**public com.sun.java.util.collections.HashMap **getExtendedAttributes**()
throws FrameworkException, SQLException

Returns the extended attributes of the instance in Install Base. Instance ID should be set in the InstallBaseItem before calling this API.

Returns: Extended attributes of the instance in Install Base.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

getInstallAddress1()

```
public java.lang.String getInstallAddress1()
```

Returns installed at address1 of the Install Base item.

Returns: Installed at address1 of the Install Base item.

getInstallAddress2()

```
public java.lang.String getInstallAddress2()
```

Returns installed at address2 of the Install Base item.

Returns: Installed at address2 of the Install Base item.

getInstallAddress3()

```
public java.lang.String getInstallAddress3()
```

Returns installed at address3 of the Install Base item.

Returns: Installed at address3 of the Install Base item.

getInstallAddress4()

```
public java.lang.String getInstallAddress4()
```

Returns installed at address4 of the Install Base item.

Returns: Installed at address4 of the Install Base item.

getInstallAtAddr()

```
public java.lang.String getInstallAtAddr()
```

Returns a concatenation of installed at address fields.

Returns: A concatenation of installed at address fields.

getInstallCity()

```
public java.lang.String getInstallCity()
```

Returns installed at city of the Install Base item.

Returns: Installed at City of the Install Base item.

getInstallCountry()

public java.lang.String **getInstallCountry()**

Returns installed at country of the Install Base item.

Returns: Installed at country of the Install Base item.

getInstallCounty()

public java.lang.String **getInstallCounty()**

Returns installed at county of the Install Base item.

Returns: Installed at county of the Install Base item.

getInstallPostalCode()

public java.lang.String **getInstallPostalCode()**

Returns installed at postal code of the Install Base item.

Returns: Installed at postal code of the Install Base item.

getInstallProvince()

public java.lang.String **getInstallProvince()**

Returns installed at province of the Install Base item.

Returns: Installed at province of the Install Base item.

getInstallState()

public java.lang.String **getInstallState()**

Returns installed at state of the Install Base item.

Returns: Installed at state of the Install Base item.

getInstanceld()

public java.math.BigDecimal **getInstanceId()**

Returns the instance ID of the Install Base item.

Returns: Instance ID of the Install Base item.

getInstanceName()

public java.lang.String **getInstanceName()**

Returns the instance name of the Install Base item.

Returns: Instance name of the Install Base item.

getInstanceNumber()

public java.lang.String **getInstanceNumber()**

Returns the instance number of the Install Base item.

Returns: Instance number of the Install Base item.

getInvOrgId()

```
public java.math.BigDecimal getInvOrgId()
```

Returns the inventory organization ID of the Install Base item.

Returns: Inventory organization ID of the Install Base item.

getItemId()

```
public java.math.BigDecimal getItemId()
```

Returns the inventory item ID of the Install Base item.

Returns: Inventory item ID of the Install Base item.

getNodeType()

```
public int getNodeTypeId()
```

Returns the node type of the Install Base item in an Install Base item tree.

Returns: The node type of the Install Base item in an Install Base item tree.

getOrderedDate()

```
public java.sql.Timestamp getOrderedDate()
```

Returns the ordered date of the Install Base item.

Returns: Ordered date of the Install Base item.

getOrderNumber()

```
public java.math.BigDecimal getOrderNumber()
```

Returns the order number of the Install Base Item.

Returns: Order number of the Install Base Item.

getParentId()

```
public java.math.BigDecimal getParentId()
```

Returns the parent ID of the Install Base item.

Returns: The parent ID of the Install Base item.

getPartNumber()

```
public java.lang.String getPartNumber()
```

Returns the part number of the Install Base item.

Returns: Part number of the Install Base item.

getQuantity()

```
public java.math.BigDecimal getQuantity()
```

Returns the quantity of the Install Base item.

Returns: Quantity of the Install Base item.

getRelatedInstances(String, boolean)

```
public oracle.apps.qot.core.InstallBaseItem[]  
getRelatedInstances(java.lang.String relationshipType, boolean fullDepth)  
throws FrameworkException, SQLException
```

Lists related Install Base Items for relationship type and depth specified. The returning array of Install Base items will not contain the install base item itself.

Parameters:

relationshipType - The type of relationship among the items in the Install Base. Possible values are:

- InstallBaseItem.COMPONENT_OF
- InstallBaseItem.CONNECTED_TO

fullDepth - If true, returns all related instances that are descendents of the install base item. Otherwise, returns the immediate descendents of the Install Base item.

This parameter may not be true if the parameter relationshipType is a bi-directional instance relationship, such as CONNECTED_TO, which might end up with an infinite loop of related instances.

Returns: An array of Install Base items that are related to this install base item based on the given relationship type. If no related instances are obtained, returns an empty array.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

getRelatedInstances(String, HashMap, int, int)

```
public oracle.apps.qot.core.InstallBaseItem[]  
getRelatedInstances(java.lang.String relationshipType,  
com.sun.java.util.collections.HashMap expandList, int loadDepth,  
int loadStructure)  
throws FrameworkException, SQLException
```

Lists Install Base items that are related to the Install Base item in an Install Base item tree, given the relationship type, the list of items to be expanded in the tree. The returning array of Install Base items will contain the item itself (the root of the tree).

Parameters:

relationshipType - The relationship type code. Possible values are:

- InstallBaseItem.COMPONENT_OF
- InstallBaseItem.CONNECTED_TO

expandList - The list of Install Base items in the Install Base items tree of which the root is the Install Base item. This is in the form of a hash table, with the key being the Install Base item (root), the value being an array containing a list of Install Base items in the tree to be expanded.

loadDepth - Determines when instances are loaded from database, whether all nodes in an install- base item tree will be loaded, or only the first level children of the items in the tree specified in the expand list will be loaded. Possible values are:

- QuoteConstant.COMPLETE_DEPTH
- QuoteConstant.PARTIAL_DEPTH

loadStructure - Determines how the results (the instances in an Install Base tree) should be returned. Possible values are:

- QuoteConstant.TREE_STRUCTURE
- QuoteConstant.FLATTEN_STRUCTURE

Returns: An array of Install Base items loaded that are related to this Install Base item based on the given relationship type.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

Since: 11.5.9

getSerialNumber()

```
public java.lang.String getSerialNumber()
```

Returns serial number of the Install Base item.

Returns: Serial number of the Install Base item.

getSystemName()

```
public java.lang.String getSystemName()
```

Returns the system name of the Install Base item.

Returns: System name of the Install Base item.

getUom()

```
public java.lang.String getUom()
```

Returns primary unit of measure of the Install Base item.

Returns: Primary unit of measure of the Install Base item.

getUomCode()

```
public java.lang.String getUomCode()
```

Returns primary unit of measure code of the Install Base item.

Returns: Primary unit of measure code of the Install Base item.

list(int, BigDecimal, HashMap, int, int, int)

```
public static oracle.apps.got.util.QueryResultSet list(int type,
java.math.BigDecimal invOrgId,
com.sun.java.util.collections.HashMap searchCriteria, int batchSize,
int startIndex, int resCount)throws FrameworkException, SQLException
Lists Install Base items based on search criteria passed in as parameter.
```

Parameters:

type - Type of search to be conducted. Possible values:

- InstallBaseItem.ADD_SERVICE
- InstallBaseItem.TRADE_IN
- InstallBaseItem.RECONFIGURE
- InstallBaseItem.ADD_TO_CONTAINER

invOrgId - Inventory organization ID.

searchCriteria - Criteria for search. HashMap having:

- key = CATEGORY, value = categoryId
- key = CATEGORY_SET, value = categorySetId
- key = ORDER_NUMBER, value =orderNumber
- key = ORDER_DATE_FROM, value = order date from
- key = ORDER_DATE_TO, value = order date to
- key = ITEM_PART_NUM, value = item part number
- key = ITEM_DESC, value =item description
- key = CUST_ACCT_ID, value = customer account ID

- key = SERIAL_NUM, value =serial number
- key = SYSTEM_NAME, value =system name
- key = INSTANCE_NAME, value =instance name
- key = INSTANCE_NUMBER, value =instance number
- key = INSTALL_CITY, value =install city
- key = INSTALL_COUNTRY, value =install country code
- key = ATTRIBUTE_ID, value = extended attribute ID
- key = ATTRIBUTE_VALUE, value =extended attribute value
- key = CONTAINER_MODEL_ID, value = inventory item ID of container model for search type ADD_TO_CONTAINER

batchSize - Batch size for the query.

startIndex - Start index indicating which row number to start retrieving data from. -1 indicates the last rows.

resCount - The total number of objects to return. If -1, then count will be queried from the database.

Returns: A QueryResultSet object containing an array of InstallBaseItem objects for search criteria and type passed in.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.core.InstallBaseItem load(java.math.BigDecimal  
instanceId)
```

```
throws FrameworkException, SQLException
```

Returns an InstallBaseItem object corresponding to the instance ID passed in as paramter.

Parameters: instanceId - Instance ID of the item in Install Base.

Returns:

InstallBaseItem object for the instance ID passed in.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

setBomItemType(int)

```
public void setBomItemType(int bomItemType)
```

Sets the value for bom item type.

Parameters: bomItemType - BOM item type of the Install Base item.

setDepth(int)

```
public void setDepth(int depth)
```

Sets node depth for the Install Base item to be displayed in a tree.

Parameters: depth - The depth of the Install Base item.

setDescription(String)

```
public void setDescription(java.lang.String description)
```

Sets the value for description.

Parameters: description - Description of the Install Base item.

setInstallAddress1(String)

```
public void setInstallAddress1(java.lang.String installAddress1)
```

Sets the value for install at address1.

Parameters: installAddress1 - Installed at address1 of the Install Base item.

setInstallAddress2(String)

```
public void setInstallAddress2(java.lang.String installAddress2)
```

Sets the value for install at address2.

Parameters: installAddress2 - Installed at address2 of the Install Base item.

setInstallAddress3(String)

```
public void setInstallAddress3(java.lang.String installAddress3)
```

Sets the value for install at address3.

Parameters: installAddress3 - Installed at address3 of the Install Base item.

setInstallAddress4(String)

```
public void setInstallAddress4(java.lang.String installAddress4)
```

Sets the value for install at address4.

Parameters: installAddress4 - Installed at address4 of the Install Base item.

setInstallCity(String)

```
public void setInstallCity(java.lang.String installCity)
```

Sets the value for install at city.

Parameters: InstallCity - installed at city of the Install Base item.

setInstallCountry(String)

```
public void setInstallCountry(java.lang.String installCountry)
```

Sets the value for install at Country.

Parameters: installCountry - Installed at country of the Install Base item.

setInstallCounty(String)

```
public void setInstallCounty(java.lang.String installCounty)
```

Sets the value for install at county.

Parameters: installCounty - Installed at country of the Install Base item.

setInstallPostalCode(String)

```
public void setInstallPostalCode(java.lang.String installPostalCode)
```

Sets the value for install at postal code.

Parameters: installPostalCode - Installed at postal code of the Install Base item.

setInstallProvince(String)

```
public void setInstallProvince(java.lang.String installProvince)
```

Sets the value for install at Province.

Parameters: installProvince - Installed at province of the Install Base item.

setInstallState(String)

```
public void setInstallState(java.lang.String installState)
```

Sets the value for install at state.

Parameters: installState - Installed at state of the Install Base item.

setInstanceId(BigDecimal)

```
public void setInstanceId(java.math.BigDecimal instanceId)
```

Sets the value for the Instance ID.

Parameters: instanceId - Instance ID of the Install Base item.

setInstanceName(String)

```
public void setInstanceName(java.lang.String instanceName)
```

Sets the value for Instance name.

Parameters: instanceName - Instance name of the Install Base item.

setInstanceNumber(String)

```
public void setInstanceNumber(java.lang.String instanceNumber)
```

Sets the value for instance number.

Parameters: instanceNumber - Instance number of the Install Base item.

setInvOrgId(BigDecimal)

```
public void setInvOrgId(java.math.BigDecimal invOrgId)
```

Sets the value for the inventory organization ID.

Parameters: invOrgId - Inventory organization ID of the Install Base item.

setItemId(BigDecimal)

```
public void setItemId(java.math.BigDecimal itemId)
```

Sets the value for the item ID.

Parameters: itemId - Item ID of the Install Base item.

setNodeType(int)

```
public void setNodeType(int nodeType)
```

Sets the node type for the Install Base item to be displayed in a tree.

Parameters: nodeType - The node type of the Install Base item.

setOrderedDate(Timestamp)

```
public void setOrderedDate(java.sql.Timestamp orderedDate)
```

Sets the value for ordered date of item.

Parameters: orderedDate - Ordered on date of the Install Base item.

setOrderNumber(BigDecimal)

```
public void setOrderNumber(java.math.BigDecimal orderNumber)
```

Sets the value for order number of item.

Parameters: orderNumber - Order number of the Install Base item.

setParentId(BigDecimal)

```
public void setParentId(java.math.BigDecimal parentId)
```

Sets the parent instance ID for this Install Base item.

Parameters: parentId - The parent ID of the Install Base item.

setPartNumber(String)

```
public void setPartNumber(java.lang.String partNumber)
```

Sets the value for part number.

Parameters: partNumber - Part number of the Install Base item.

setQuantity(BigDecimal)

```
public void setQuantity(java.math.BigDecimal quantity)
```

Sets the value for the quantity.

Parameters: quantity - Quantity of the Install Base item.

setSerialNumber(String)

```
public void setSerialNumber(java.lang.String serialNumber)
```

Sets the value for serial number.

Parameters: serialNumber - Serial number of the Install Base item.

setSystemName(String)

```
public void setSystemName(java.lang.String systemName)
```

Sets the value for system name.

Parameters: systemName - System name of the Install Base item.

setUom(String)

```
public void setUom(java.lang.String uom)
```

Sets the value for unit of measure.

Parameters: uom - Unit of measure of the Install Base item.

setUomCode(String)

```
public void setUomCode(java.lang.String uomCode)
```

Sets the value for primary uom code.

Parameters: uomCode - UOM code of the Install Base item.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the InstallBaseItem object.

Overrides: toString in class Object

Returns: A String representation of the InstallBaseItem object.

1.12 Class NotificationDetail

```
java.lang.Object
```

```
|  
+--oracle.apps.got.core.NotificationDetail
```

public class **NotificationDetail**

NotificationDetail contains the following information for approval notification details: recipient role name, activity display name, begin date, end date, due date, activity status, result.

Table 1–20 Inherited Member Summary

Methods inherited from class Object

```
equals(Object), getClass(), hashCode(), notify(), notifyAll(),  
wait(long, int), wait(long, int), wait(long, int)
```

1.12.1 Fields for Class NotificationDetail

activityDisplayName

```
public java.lang.String activityDisplayName  
Activity display name for the approval notification.
```

activityStatus

```
public java.lang.String activityStatus  
Activity status for the approval notification.
```

beginDate

```
public java.sql.Timestamp beginDate  
Begin date for the approval notification.
```

dueDate

```
public java.sql.Timestamp dueDate  
Due date for the approval notification.
```

endDate

```
public java.sql.Timestamp endDate  
End date for the approval notification.
```

RCS_ID

```
public static final java.lang.String RCS_ID
```


recipientRoleName

```
public java.lang.String recipientRoleName
```

Recipient role name for the approval notification.

result

```
public java.lang.String result
```

Result for the approval notification.

1.12.2 Constructors for Class NotificationDetail

NotificationDetail()

```
public NotificationDetail()
```

Default constructor.

NotificationDetail(String, String, Timestamp, Timestamp, Timestamp, String, String)

```
public NotificationDetail(java.lang.String recipientRoleName,  
java.lang.String activityDisplayName, java.sql.Timestamp beginDate,  
java.sql.Timestamp endDate, java.sql.Timestamp dueDate,  
java.lang.String activityStatus, java.lang.String result)
```

Construct notification detail with recipient role name, activity display name, begin date, end date, due date, activity status, and result.

Parameters:

recipientRoleName - Recipient role name.

activityDisplayName - Activity display name.

beginDate - Begin date.

endDate - End date.

activityStatus - Activity status.

result - Result.

1.12.3 Methods for Class NotificationDetail

The following table is an index of the Class NotificationDetail methods:

Table 1–21 *Methods for Class NotificationDetail*

Method	Description
toString	Returns a String representation of Notification Detail. public java.lang.String toString()

toString()

```
public java.lang.String toString()
```

Returns a String representation of Notification Detail.

Overrides: toString in class Object

Returns: A String representation of notification detail.

1.13 Class PendingOrderLine

```
java.lang.Object
|
+--oracle.apps.got.core.PendingOrderLine
```

```
public class PendingOrderLine
```

PendingOrderLine contains methods to retrieve information for a pending order line and to search for pending order lines.

Table 1–22 *Inherited Member Summary*

Methods inherited from class Object
equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

1.13.1 Fields for Class PendingOrderLine

CUST_ACCT_ID

```
public static final java.lang.String CUST_ACCT_ID
```

Indicates search for pending order lines created for a customer account. Used as the key for an entry in the HashMap for the pending order line search criteria input in method PendingOrderLine.list().

ITEM_DESC

```
public static final java.lang.String ITEM_DESC
```

Indicates search for pending order lines with an inventory item description. Used as the key for an entry in the HashMap for the pending order line search criteria input in method PendingOrderLine.list().

ITEM_PART_NUM

```
public static final java.lang.String ITEM_PART_NUM
```

Indicates search for pending order lines with an inventory item part number. Used as the key for an entry in the HashMap for the pending order line search criteria input in method PendingOrderLine.list().

ORDER_DATE_FROM

```
public static final java.lang.String ORDER_DATE_FROM
```

Indicates search for pending orders placed on or after a specific order date. Used as the key for an entry in the HashMap for the pending order line search criteria input in method PendingOrderLine.list().

ORDER_DATE_TO

```
public static final java.lang.String ORDER_DATE_TO
```

Indicates search for pending orders placed on or before a specific order date. Used as the key for an entry in the HashMap for the pending order line search criteria input in method PendingOrderLine.list().

ORDER_NUMBER

```
public static final java.lang.String ORDER_NUMBER
```

Indicates search for pending orders with a specific order number. Used as the key for an entry in the HashMap for the pending order line search criteria input in method PendingOrderLine.list().

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.13.2 Constructors for Class PendingOrderLine

PendingOrderLine()

```
public PendingOrderLine()
```

Constructor.

PendingOrderLine(BigDecimal, Timestamp, BigDecimal, BigDecimal, BigDecimal, String, BigDecimal, String, String)

```
public PendingOrderLine(java.math.BigDecimal _orderNumber,  
java.sql.Timestamp _orderDate, java.math.BigDecimal _orderLineId,  
java.math.BigDecimal _orderLineNumber, java.math.BigDecimal _quantity,  
java.lang.String _uomCode, java.math.BigDecimal _itemId,  
java.lang.String _partNumber, java.lang.String _description)  
throws FrameworkException, SQLException
```

Construct pending order line with order number, order date, order line ID, order line number, quantity, unit of measure code, inventory item ID, inventory item part number, and inventory item description.

Parameters:

`_orderNumber` - Order number.

`_orderDate` - Order date.

`_orderLineId` - Order line ID.

`_quantity` - Quantity.

`_uomCode` - Unit of measure code.

`_itemId` - Inventory item ID.

`_partNumber` - Inventory item part number.

`_description` - Inventory item description.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

1.13.3 Methods for Class PendingOrderLine

The following table is an index of the Class PendingOrderLine methods:

Table 1–23 Methods for Class PendingOrderLine

Method	Description
getDescription()	Returns the description from the PendingOrderLine Object. public java.lang.String getDescription()
getItemId()	Returns the item ID from the PendingOrderLine Object. public java.math.BigDecimal getItemId()
getOrderDate()	Returns the OrderDate from the PendingOrderLine Object. public java.sql.Timestamp getOrderDate()
getOrderLineId()	Returns the OrderLineId from the PendingOrderLine Object. public java.math.BigDecimal getOrderLineId()
getOrderLineNumber()	Returns the OrderLineNumber from the PendingOrderLine Object. public java.math.BigDecimal getOrderLineNumber()
getOrderNumber()	Returns the OrderNumber from the PendingOrderLine Object. public java.math.BigDecimal getOrderNumber()
getPartNumber()	Returns the partNumber from the PendingOrderLine Object. public java.lang.String getPartNumber()
getQuantity()	Returns the Quantity from the PendingOrderLine Object. public java.math.BigDecimal getQuantity()
getUom()	Returns the Translated UOM Description from the PendingOrderLine Object. public java.lang.String getUom()
getUomCode()	Returns the UomCode from the PendingOrderLine Object. public java.lang.String getUomCode()
list(BigDecimal, HashMap, int, int, int)	Lists the pending order lines for the search criteria, batch size, start index, and result count passed in as parameter. public static oracle.apps.qot.util.QueryResultSet list (java.math.BigDecimal invOrgId, com.sun.java.util.collections.HashMap searchCr iteria, int batchSize, int startIndex, int resCount) throws FrameworkException, SQLException

Table 1–23 Methods for Class PendingOrderLine

Method	Description
load(BigDecimal, BigDecimal)	<p>Loads pending order line for the order line ID passed in as parameter.</p> <pre>public static oracle.apps.qot.core.PendingOrderLine load(java.math.BigDecimal invOrgId, java.math.BigDecimal orderLineId) throws FrameworkException, SQLException</pre>
setDescription(String)	<p>Sets the Description for the PendingOrderLine Object.</p> <pre>public void setDescription(java.lang.String _ description)</pre>
setItemId(BigDecimal)	<p>Sets the Item ID for the PendingOrderLine Object.</p> <pre>public void setItemId(java.math.BigDecimal _ itemId)</pre>
setOrderDate(Timestamp)	<p>Sets the OrderDate for the PendingOrderLine Object.</p> <pre>public void setOrderDate(java.sql.Timestamp _ orderDate)</pre>
setOrderLineId(BigDecimal)	<p>Sets the OrderLineId for the PendingOrderLine Object.</p> <pre>public void setOrderLineId(java.math.BigDecimal _ orderLineId)</pre>
setOrderLineNumber(BigDecimal)	<p>Sets OrderLineNumber for the PendingOrderLine Object.</p> <pre>public void setOrderLineNumber(java.math.BigDecimal _ orderLineNumber)</pre>
setOrderNumber(BigDecimal)	<p>Sets the OrderNumber for the PendingOrderLine Object.</p> <pre>public void setOrderNumber(java.math.BigDecimal _ orderNumber)</pre>
setPartNumber(String)	<p>Sets the PartNumber for the PendingOrderLine Object.</p> <pre>public void setPartNumber(java.lang.String _ partNumber)</pre>
setQuantity(BigDecimal)	<p>Sets the Quantity for the PendingOrderLine Object.</p> <pre>public void setQuantity(java.math.BigDecimal _ quantity)</pre>

Table 1–23 Methods for Class PendingOrderLine

Method	Description
setUom(String)	Sets the Translated Uom Description from the PendingOrderLine Object. public java.lang.String setUom (java.lang.String _uom)
setUomCode(String)	Sets the UomCode for the PendingOrderLine Object. public void setUomCode (java.lang.String _uomCode)
toString()	Returns a String representation of PendingOrderLine object. public java.lang.String toString ()

getDescription()

```
public java.lang.String getDescription()
```

Returns the description from the PendingOrderLine object.

Returns: Description.

getItemId()

```
public java.math.BigDecimal getItemId()
```

Returns the item ID from the PendingOrderLine object.

Returns: ItemId.

getOrderDate()

```
public java.sql.Timestamp getOrderDate()
```

Returns the order date from the PendingOrderLine object.

Returns: orderDate.

getOrderLineId()

```
public java.math.BigDecimal getOrderLineId()
```

Returns the order line ID from the PendingOrderLine object.

Returns: Order number.

getOrderLineNumber()

```
public java.math.BigDecimal getOrderLineNumber()
```

Returns the order line number from the PendingOrderLine object.

Returns: Order line number.

getOrderNumber()

public java.math.BigDecimal **getOrderNumber()**

Returns the order number from the PendingOrderLine object.

Returns: OrderNumber.

getPartNumber()

public java.lang.String **getPartNumber()**

Returns the part number from the PendingOrderLine object.

Returns: PartNumber.

getQuantity()

public java.math.BigDecimal **getQuantity()**

Returns the quantity from the PendingOrderLine object.

Returns: Quantity.

getUom()

public java.lang.String **getUom()**

Returns the translated UOM description from the PendingOrderLine object.

Returns: Translated unit of measure.

getUomCode()

public java.lang.String **getUomCode()**

Returns the UOM code from the PendingOrderLine object.

Returns: The UOM code.

list(BigDecimal, HashMap, int, int, int)

public static oracle.apps.qot.util.QueryResultSet

list(java.math.BigDecimal invOrgId,
com.sun.java.util.collections.HashMap searchCriteria, int batchSize,
int startIndex, int resCount)

throws FrameworkException, SQLException

Lists the pending order lines for the search criteria, batch size, start index, and result count passed in as parameter.

Parameters:

invOrgId - Inventory org ID.

searchCriteria - Search Criteria.

- key = ORDER_NUMBER, value = order number

- key = ORDER_DATE_FROM, value = order date from
- key = ORDER_DATE_TO, value = order date to
- key = ITEM_PART_NUM, value = item part number
- key = ITEM_DESC, value = item description
- key = CUST_ACCT_ID, value = custAccountId

batchsSize - Batch size for the query.

startIndex - Start index indicating which row number to start retrieving data. -1 indicates the last rows.

resCount - The total number of objects to return. If -1, the count will be queried from the database.

Returns: QueryResultSet containing array of pending order lines.

Throws:

java.sql.SQLException - If database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If error occurs while getting connection.

load(BigDecimal, BigDecimal)

```
public static oracle.apps.qot.core.PendingOrderLine  
load(java.math.BigDecimal invOrgId, java.math.BigDecimal orderLineId)  
throws FrameworkException, SQLException
```

Loads pending order line for the order line ID passed in as parameter.

Parameters:

invOrgId - Inventory org ID.

orderLineId - Order line ID.

Returns: PendingOrderLine object.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If an error occurs while getting connection.

setDescription(String)

```
public void setDescription(java.lang.String _description)  
Sets the description for the PendingOrderLine object.
```

Parameters: _description - Inventory item description.

setItemId(BigDecimal)

public void **setItemId**(java.math.BigDecimal _itemId)
set the item ID for the PendingOrderLine object.

Parameters: _itemId - Inventory item ID.

setOrderDate(Timestamp)

public void **setOrderDate**(java.sql.Timestamp _orderDate)
Sets the orderdate for the PendingOrderLine object.

Parameters: _orderDate - Order date.

setOrderLineId(BigDecimal)

public void **setOrderLineId**(java.math.BigDecimal _orderLineId)
Sets the order line ID for the PendingOrderLine object.

Parameters: _orderLineId - Order line ID.

setOrderLineNumber(BigDecimal)

public void **setOrderLineNumber**(java.math.BigDecimal _orderLineNumber)
Sets the order line number for the PendingOrderLine object.

Parameters: _orderLineNumber - Order line number.

setOrderNumber(BigDecimal)

public void **setOrderNumber**(java.math.BigDecimal _orderNumber)
Sets the order number for the PendingOrderLine object.

Parameters: _orderNumber - Order number.

setPartNumber(String)

public void **setPartNumber**(java.lang.String _partNumber)
Sets the part number for the PendingOrderLine object.

Parameters: _partNumber - Inventory item part number.

setQuantity(BigDecimal)

public void **setQuantity**(java.math.BigDecimal _quantity)
Sets the quantity for the PendingOrderLine object.

Parameters: _quantity - Quantity.

setUom(String)

public java.lang.String **setUom**(java.lang.String _uom)

set the translated UOM description from the PendingOrderLine object.

Parameters: `_uom` - Translated unit of measure.

setUomCode(String)

```
public void setUomCode(java.lang.String _uomCode)
```

Sets the UOM code for the PendingOrderLine object.

Parameters: `_uomCode` - Unit of measure code.

toString()

```
public java.lang.String toString()
```

Returns a String representation of PendingOrderLine object.

Overrides: `toString` in class `Object`

Returns: A String representation of PendingOrderLine object.

1.14 Class Proposal

```
java.lang.Object
|
+--oracle.apps.qot.core.Proposal
public class Proposal
```

The Proposal object contains the information for a proposal that is associated with a quote. It contains the following attributes: quote proposal ID, description, proposal name, proposal due date and proposal status.

This class provides the ability to list proposals associated with a quote.

Table 1–24 Inherited Member Summary

Methods inherited from class `Object`

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

1.14.1 Fields for Class Proposal

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.14.2 Constructors for Class Proposal

Proposal()

```
public Proposal()
```

Default constructor.

1.14.3 Methods for Class Proposal

The following table is an index of the Class Proposal methods:

Table 1–25 Methods for Class Proposal

Method	Description
<code>getCreatedByName()</code>	Returns name of the user who created the proposal. <pre>public java.lang.String getCreatedByName()</pre>
<code>getDescription()</code>	Returns proposal description. <pre>public java.lang.String getDescription()</pre>
<code>getDueDate()</code>	Returns proposal due date. <pre>public java.sql.Timestamp getDueDate()</pre>
<code>getProposalName()</code>	Returns proposal name. <pre>public java.lang.String getProposalName()</pre>
<code>getProposalStatusCode()</code>	Returns status of the proposal status code. <pre>public java.lang.String getProposalStatusCode()</pre>
<code>getQuoteProposalId()</code>	Returns quote proposal ID. <pre>public java.math.BigDecimal getQuoteProposalId()</pre>
<code>getStatus()</code>	Returns status of the proposal. <pre>public java.lang.String getStatus()</pre>
<code>list(BigDecimal, int, int, int)</code>	Lists the proposals associated with the quote header passed in as parameter. <pre>public static oracle.apps.qot.util.QueryResultSet list(java.math.BigDecimal quoteHeaderId, int batchSize, int startIndex, int resultCount) throws FrameworkException, SQLException</pre>
<code>toString()</code>	Returns a String representation of the proposal object. <pre>public java.lang.String toString()</pre>

getCreatedByName()

```
public java.lang.String getCreatedByName()
```

Returns name of the user who created the proposal.

Returns: Name of the user who created the proposal.

getDescription()

```
public java.lang.String getDescription()
```

Returns proposal description.

Returns: Proposal description.

getDueDate()

```
public java.sql.Timestamp getDueDate()
```

Returns proposal due date.

Returns: Proposal due date.

getProposalName()

```
public java.lang.String getProposalName()
```

Returns Proposal name.

Returns: Proposal name.

getProposalStatusCode()

```
public java.lang.String getProposalStatusCode()
```

Returns status of the proposal status code.

Returns: Status of the proposal status code.

getQuoteProposalId()

```
public java.math.BigDecimal getQuoteProposalId()
```

Returns quote Proposal ID.

Returns: Quote proposal ID.

getStatus()

```
public java.lang.String getStatus()
```

Returns status of the proposal.

Returns: Status of the proposal.

list(BigDecimal, int, int, int)

```
public static oracle.apps.got.util.QueryResultSet list(java.math.BigDecimal  
quoteHeaderId, int batchSize, int startIndex, int resultCount)  
throws FrameworkException, SQLException
```

Lists the proposals associated with the quote number passed in as parameter.

Parameters:

quoteNumber - Quote number.

batchSize - Batch size for the query.

startIndex - Start index indicating which row number to start retrieving data. -1 indicates that last page.

resultCount - The total number of objects to return. If -1, the count will be queried from the database.

Returns: QueryResultSet containing an array of Proposal objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the Proposal object.

Overrides: toString in class Object

Returns: A String representation of the Proposal object.

1.15 Class Quote

```
java.lang.Object
|
+--oracle.apps.qot.core.Quote
```

```
public class Quote
```

Quote object is used to model an entire quote. It is composed of a QuoteHeader and multiple QuoteLines. Quote contains the basic methods to create, update, delete, and retrieve quotes from the database.

1.15.1 Fields for Class Quote

appliedTaxes

```
public oracle.apps.aso.quote.TaxDetailRecord[] appliedTaxes
```

An array of tax detail records representing the taxes which have been applied to the quote. For each tax code, calculates the sum of the tax amount for all lines in the

quote. The following fields will be populated in the tax detail records: tax_code, tax_amount, tax_name. This member variable is used only for retrieving information from the database. This member variable will be populated after calling the method loadAppliedTaxes().

DESCRIPTION

public static final java.lang.String **DESCRIPTION**

Indicates search column quote description. Used as input to the method Quote.listQuoteTemplates().

HDR_OBJ_TYPE

public static final java.lang.String **HDR_OBJ_TYPE**

Indicates quote header object type. Used to populate quote_object_type_code in RelatedObjectRecord when constructing the input parameter for Quote.updateObjectRelationship().

lineReIns

public oracle.apps.aso.quote.LineRelationshipRecord[] **lineReIns**

Quote line relationships corresponding to rows in ASO_LINE_RELATIONSHIPS. This member variable is used to retrieve and update information in the database. This member variable will be populated after calling the method loadLineRelationships().

NAME

public static final java.lang.String **NAME**

Indicates that search column quote name. Used as input to the method Quote.listQuoteTemplates().

OPP_OBJ_TYPE

public static final java.lang.String **OPP_OBJ_TYPE**

Indicates opportunity related object type. Used to populate object_type_code in RelatedObjectRecord when constructing the input parameter for Quote.updateObjectRelationship().

OPP_QUOTE_REL_TYPE

public static final java.lang.String **OPP_QUOTE_REL_TYPE**

Indicates opportunity and quote relationship type. Used to populate relationship_type_code in RelatedObjectRecord when constructing the input parameter for Quote.updateObjectRelationship().

prcAdjPerModifierLine

```
public oracle.apps.aso.quote.PriceAdjustmentRecord[] prcAdjPerModifierLine
```

An array of price adjustment records representing the price adjustments per modifier line. For each price modifier line, the sum of the adjusted amount will be determined. The following fields will be populated in the price adjustment records: `modifier_line_id`, `modifier_header_id`, `modifier_header_name`, `modifier_level_code`, `modifier_level`, `modifier_line_type_code`, `modifier_line_type`, `charge_type_code`, `charge_type`, `charge_subtype_code`, `charge_subtype`, `adjusted_amount`. This member variable is used only for retrieving information from the database. This member variable will be populated after calling the method `loadPrcAdjPerModifierLine()`.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.15.2 Constructors for Class Quote

Quote()

```
public Quote()
```

Default constructor.

Quote(BigDecimal)

```
public Quote(java.math.BigDecimal headerId)
```

Constructs Quote object. QuoteHeader will be populated with quote header ID passed in as parameter.

Parameters: headerId - Quote header ID.

1.15.3 Methods for Class Quote

The following table is an index of the Class Quote methods:

Table 1–26 Methods for Class Quote

Method	Description
addQuoteLines(ControlRecord, BigDecimal[], BigDecimal[], String[], String[], String[], BigDecimal[])	<p>Add lines to the quote. This API should be called within a transaction block.</p> <pre>public void addQuoteLines(oracle.apps.aso.quote.ControlRecord controlRec, java.math.BigDecimal[] itemIds, java.math.BigDecimal[] invOrgIds, java.lang.String[] uomCodes, java.lang.String[] qtyStr, java.lang.String[] itemTypes, java.math.BigDecimal[] priceListIds) throws FrameworkException, SQLException, QuoteException</pre>
addTemplatesToQuote(ControlRecord, BigDecimal[])	<p>Adds quote templates to a quote. This API should be called within a transaction block.</p> <pre>public void addTemplatesToQuote(oracle.apps.aso.quote.ControlRecord controlRec, java.math.BigDecimal[] templateIds) throws FrameworkException, SQLException, QuoteException</pre>
cancelPricingBatchRequest()	<p>Cancels the current pricing batch request for the quote. Quote header information, such as <code>quote_header_id</code>, <code>pricing_status_indicator</code>, <code>tax_status_indicator</code>, should be set in the quote object. This method should be called within a transaction block.</p> <pre>public void cancelPricingBatchRequest() throws FrameworkException, QuoteException, SQLException</pre>
checkAvailability(AtpRecord[])	<p>Checks the ATP availability information of each quote line in the quote. This API should be called within a transaction block.</p> <pre>public void checkAvailability(oracle.apps.aso.quote.AtpRecord[] atpRecords) throws FrameworkException, SQLException, QuoteException</pre>

Table 1–26 Methods for Class Quote

Method	Description
checkCredit()	<p>Performs credit check on the customer: If credit check returns NULL result, method throws a FrameworkException with “No credit check performed” error message; If credit check returns with FAIL, method throws a QuoteException with message got from PL/SQL OUT parameter X_CC_HOLD_COMMENT; If credit check returns PASS, method returns without throwing any exceptions. This API should be called within a transaction block.</p> <pre>public void checkCredit()throws FrameworkException, SQLException, QuoteException</pre>
checkCustomerAccounts()	<p>Checks whether there are any missing accounts in the system for the sold-to, ship-to, or invoice-to customers specified in the quote. If there are no missing accounts, the method will return without error. If there are missing accounts, a QuoteException will be thrown. Quote header ID should be populated before calling this method.</p> <pre>public void checkCustomerAccounts() throws SQLException, FrameworkException, QuoteException</pre>
copy(CopyQuoteControlRecord, QuoteHeader)	<p>Creates a copy of an existing quote. The new quote will not be repriced. This API should be called within a transaction block.</p> <pre>public static oracle.apps.qot.core.Quote copy(oracle.apps.aso.quote.CopyQuoteControlRecord copyControlRec, oracle.apps.qot.core.QuoteHeader copyQuoteHdr) throws FrameworkException, SQLException, QuoteException</pre>
create(ControlRecord)	<p>Creates a quote with the attributes populated in this quote object. Quote header ID, last update date, and quote number will be populated in the quote header after the quote is created. This API should be called within a transaction block.</p> <pre>public void create(oracle.apps.aso.quote.ControlRecord controlRec)throws FrameworkException, SQLException, QuoteException</pre>

Table 1–26 Methods for Class Quote

Method	Description
createFromOpportunity(ControlRecord, String, OppQteInRec)	<p>Creates a new quote based on the information populated in the OppQteInRec record structure. A quote will be created which contains the lines that are in the opportunity. This API should be called within a transaction block.</p> <pre>public static oracle.apps.qot.core.Quote createFromOpportunity(oracle.apps.aso.quote.ControlRecord controlRec, java.lang.String sourceCode, oracle.apps.aso.quote.OppQteInRec oppQteInRec)throws FrameworkException, SQLException, QuoteException</pre>
createNewVersion(CopyQuoteControlRecord, BigDecimal)	<p>Creates a new version of an existing quote. The new version of the quote will be repriced. This API should be called within a transaction block.</p> <pre>public static oracle.apps.qot.core.Quote createNewVersion(oracle.apps.aso.quote.CopyQuoteControlRecord copyControlRec, java.math.BigDecimal quoteHdrId)throws FrameworkException, SQLException, QuoteException</pre>
createNewVersion(CopyQuoteControlRecord, QuoteHeader)	<p>Creates a new version of an existing quote. The new version of the quote will not be repriced. This API should be called within a transaction block.</p> <pre>public static oracle.apps.qot.core.Quote createNewVersion(oracle.apps.aso.quote.CopyQuoteControlRecord copyControlRec, oracle.apps.qot.core.QuoteHeader quoteHdr) throws FrameworkException, SQLException, QuoteException</pre>

Table 1–26 Methods for Class Quote

Method	Description
<code>deactivate(ControlRecord, BigDecimal[])</code>	<p>Deactivates configuration components from the quote. If <code>deactivate_flag</code> in control record is T, all configuration components which have not been modified will be deactivated. Otherwise, the quote line ids specified will be deactivated (removed) from quote. This API should be called within a transaction block.</p> <pre>public void deactivate(oracle.apps.aso.quote.ControlRecord controlRec, java.math.BigDecimal[] quoteLineIds) throws SQLException, FrameworkException, QuoteException</pre>
<code>deleteQuotes(BigDecimal[])</code>	<p>Deletes quotes, given an array of Quote Header ID. This API should be called within a transaction block.</p> <pre>public static void deleteQuotes(java.math.BigDecimal[] quoteHeaderIds) throws FrameworkException, SQLException, QuoteException</pre>
<code>determineExpDate()</code>	<p>Determines the default quote expiration date.</p> <pre>public static java.sql.Timestamp determineExpDate() throws SQLException, QuoteException, FrameworkException</pre>
<code>freezePrice()</code>	<p>Freezes the price of the quote by setting the <code>price_frozen_date</code> to the current database system date. The quote will be repriced. This API should be called within a transaction block.</p> <pre>public void freezePrice() throws FrameworkException, SQLException, QuoteException</pre>
<code>freezePrice(ControlRecord)</code>	<p>Freezes the price of the quote by setting the <code>price_frozen_date</code> to the current database system date. This API should be called within a transaction block.</p> <pre>public void freezePrice(oracle.apps.aso.quote.ControlRecord controlRec) throws FrameworkException, SQLException, QuoteException</pre>

Table 1–26 Methods for Class Quote

Method	Description
<code>getNumberOfLines(BigDecimal)</code>	Returns the number of lines for the specified quote header ID. <pre>public static java.math.BigDecimal getNumberOfLines(java.math.BigDecimal quoteHdrId) throws FrameworkException, SQLException</pre>
<code>getPriceAdjustments()</code>	Returns a HashMap containing the total price adjustments for the quote header and each quote line. Key in the HashMap is BigDecimal(-1) for the quote header. Key is quote line ID for each quote line. Value in the HashMap is containing the sum of the adjusted amount. <code>loadPriceAdjustments()</code> should be called before calling this API. <pre>public com.sun.java.util.collections.HashMap getPriceAdjustments()</pre>
<code>getPropertyMap()</code>	Returns the property map. <pre>public com.sun.java.util.collections.HashMap getPropertyMap()</pre>
<code>getQuoteHeader()</code>	Returns the quote header. <pre>public oracle.apps.qot.core.QuoteHeader getQuoteHeader()</pre>
<code>getQuoteLines()</code>	Returns the quote lines. <pre>public oracle.apps.qot.core.QuoteLine[] getQuoteLines()</pre>
<code>getTaxDetails()</code>	Returns a HashMap containing the tax details for the quote header and each quote line. Key in the HashMap is BigDecimal(-1) for the quote header. Key is quote line ID for each quote line. Value in the HashMap is TaxDetailRecord[] containing tax detail records. <code>loadTaxDetails()</code> should be called before calling this API. <pre>public com.sun.java.util.collections.HashMap getTaxDetails()</pre>
<code>hasLineLevelBilling()</code>	Returns whether the quote has line level billing <pre>public boolean hasLineLevelBilling() throws FrameworkException, SQLException</pre>

Table 1–26 Methods for Class Quote

Method	Description
<code>list(SavedSearch, BigDecimal, int, int)</code>	<p>Lists the quotes based on the search criteria, <code>resourceId</code>, <code>startIndex</code> and <code>resultCount</code> passed in as parameters.</p> <pre>public static oracle.apps.qot.util.QueryResultSet list(oracle.apps.qot.perzquery.SavedSearch savedSearch, java.math.BigDecimal resourceId, int startIndex, int resCount) throws FrameworkException, QueryBuilderExcept ion</pre>
<code>listQuoteTemplates(String, String, int, int, int)</code>	<p>Obtains a list of existing quote templates.</p> <pre>public static oracle.apps.qot.util.QueryResultSet listQuoteTemplates(java.lang.String searchColumn, java.lang.String searchString, int batchSize, int startIndex, int resultCount) throws FrameworkException, SQLException</pre>
<code>listVersions(BigDecimal, int, int, int)</code>	<p>Lists all versions for a particular quote number. Returns the <code>QueryResultSet</code> containing the <code>Quote[]</code> of the applicable quotes. <code>QuoteHeader</code> information will be populated in each <code>Quote</code>.</p> <pre>public static oracle.apps.qot.util.QueryResultSet listVersions(java.math.BigDecimal quoteNumber, int batchSize, int startIndex, int resCount) throws FrameworkException, SQLException</pre>
<code>load(BigDecimal, int)</code>	<p>Loads quote header information from the database based on quote header ID and load level passed in as parameter.</p> <pre>public static oracle.apps.qot.core.Quote load(java.math.BigDecimal quoteHdrId, int level) throws FrameworkException, SQLException</pre>

Table 1–26 Methods for Class Quote

Method	Description
<code>loadAllLines(int, boolean)</code>	<p>Loads all the lines in the quote based on load level and whether to load line relationships. Line details for each quote line will be populated.</p> <pre>public void loadAllLines(int level, boolean loadLineRel)throws FrameworkException , SQLException</pre>
<code>loadAppliedTaxes()</code>	<p>Loads the taxes that have been applied to the quote as a whole (sum of taxes applied to all lines for each tax code). After calling this API, <code>quote.appliedTaxes</code> will be populated with <code>TaxDetailRecord[]</code> containing the information for the applied taxes. The following information will be populated in the tax detail records: <code>tax_code</code>, <code>tax_amount</code>, <code>tax_name</code>.</p> <pre>public void loadAppliedTaxes()throws FrameworkException, SQLException</pre>
<code>loadHighestVersion(BigDecimal, int)</code>	<p>Loads quote header information from the database for the highest version of the quote number passed in as parameter. Load level passed in as parameter will determine the quote header information which will be loaded.</p> <pre>public static oracle.apps.qot.core.Quote loadHighestVersion(java.math.BigDecimal quoteNumber, int level)throws FrameworkException, SQLException</pre>
<code>loadLineRelationships()</code>	<p>Loads the line relationships for the quote. After calling this API, <code>quote.lineRelns</code> will be populated with <code>LineRelationshipRecord[]</code> containing the line relationship information.</p> <pre>public void loadLineRelationships()throws FrameworkException, SQLException</pre>

Table 1–26 Methods for Class Quote

Method	Description
<code>loadLines(int,HashMap,int,int)</code>	<p>Loads lines of a quote from database and structures the lines loaded based on line load level. Which lines to load is determined by load depth and the expand list. The loaded lines can be structured into a tree data structure or a flattened array of quote line objects. Service lines whose parents are quote lines will be loaded, but will not be put into the array of quote line objects. Instead they will be put into the array of service lines of their parent lines. Quote line details for each quote line will be populated.</p> <pre>public void loadLines(int level, com.sun.java.util.collections.HashMap expandList, int loadDepth, int loadStructure) throws FrameworkException, SQLException</pre>
<code>loadPrcAdjPerModifierLines(int,Int)</code>	<p>Loads total price adjustments for the whole quote per price modifier line. For each price modifier line, the sum of the adjusted amount will be determined. After price adjustments are loaded, the member variable <code>prcAdjPerModifierLine</code> will be populated with the following price adjustment information: <code>modifier_line_id</code>, <code>modifier_header_id</code>, <code>modifier_header_name</code>, <code>modifier_level_code</code>, <code>modifier_level</code>, <code>modifier_line_type_code</code>, <code>modifier_line_type</code>, <code>charge_type_code</code>, <code>charge_type</code>, <code>charge_subtype_code</code>, <code>charge_subtype</code>, <code>adjusted_amount</code>.</p> <pre>public void loadPrcAdjPerModifierLine(int prcAdjFlags, int prcAdjTypes) throws FrameworkException, SQLException</pre>
<code>loadPriceAdjustments(int)</code>	<p>Loads total price adjustments for the whole quote. For the quote header and each quote line, the sum of the adjusted amount will be determined. After price adjustments are loaded, use the API <code>getPriceAdjustments()</code> to retrieve a <code>HashMap</code> containing the price adjustments. In the <code>HashMap</code>, the key is quote line ID and the value is total adjusted amount. For quote header, the key is <code>BigDecimal(-1)</code>.</p> <pre>public void loadPriceAdjustments(int prcAdjFlags) throws FrameworkException, SQLException</pre>

Table 1–26 Methods for Class Quote

Method	Description
<code>loadTaxDetails()</code>	<p>Loads tax details for the whole quote. For the quote header and each quote line, the tax details will be loaded. After tax details are loaded, use the API <code>getTaxDetails()</code> to retrieve a <code>HashMap</code> containing the tax details. In the <code>HashMap</code>, the key is quote line ID, the value is <code>TaxDetailRecord[]</code> containing the tax details for the quote line. For the quote header, the key is <code>BigDecimal(-1)</code>.</p> <pre>public void loadTaxDetails() throws FrameworkException, SQ LException</pre>
<code>publish(BigDecimal, String, String, String, String, String)</code>	<p>Publishes a quote. This API should be called within a transaction block.</p> <pre>public void publish(java.math.BigDecimal emailId, java.lang.String comment, java.lang.String storeName, java.lang.String storeURL, java.lang.String storeWebSite, java.lang.String fndUserName) throws FrameworkException, SQLException, Quot eException</pre>
<code>publish(ControlRecord, BigDecimal, String, String, String, String, String)</code>	<p>Publishes a quote. This API should be called within a transaction block.</p> <pre>publish(ControlRecord, BigDecimal, String, String, String, String, String) public void publish(oracle.apps.aso.quote.ControlRecord controlRec, java.math.BigDecimal emailId, java.lang.String comment, java.lang.String storeName, java.lang.String storeURL, java.lang.String storeWebSite, java.lang.String fndUserName) throws FrameworkException, SQLException, Quot eException</pre>

Table 1–26 Methods for Class Quote

Method	Description
reconfigure(ControlRecord, InstanceRecord)	<p>Adds the Install Base items to be reconfigured, to quote, given the instance records. This API should be called within a transaction block.</p> <pre>public void reconfigure(oracle.apps.aso.quote.ControlRecord controlRec, oracle.apps.aso.quote.InstanceRecord[] instanceRec) throws SQLException, FrameworkException, QuoteException</pre>
reprice()	<p>Reprices the quote. This API should be called within a transaction block.</p> <pre>public void reprice()throws FrameworkException, SQLException, QuoteException</pre>
save(ControlRecord, int)	<p>Saves quote information to the database based on the data set in the quote object and the level indicating the information which should be saved. This API should be called within a transaction block.</p> <pre>public void save(oracle.apps.aso.quote.ControlRecord controlRec, int level)throws FrameworkException, SQLException, QuoteException</pre>
setPriceAdjustments(HashMap)	<p>Populates the HashMap containing the total price adjustments for the quote header and each quote line. Key is quote line ID for each quote line. Value in the HashMap contains the sum of the adjusted amount.</p> <pre>public void setPriceAdjustments(com.sun.java.util.collections.HashMap priceAdjs)</pre>
setPropertyMap(HashMap)	<p>Sets the property map.</p> <pre>public void setPropertyMap(com.sun.java.util.collections.HashMap map)</pre>
setQuoteHeader(QuoteHeader)	<p>Sets the quote header.</p> <pre>public void setQuoteHeader(oracle.apps.qot.core.QuoteHeader header)</pre>

Table 1–26 Methods for Class Quote

Method	Description
<code>setQuoteLines(QuoteLine[])</code>	<p>Sets the quote lines.</p> <pre>public void setQuoteLines(oracle.apps.got.core.QuoteLine[] lines)</pre>
<code>setTaxDetails(HashMap)</code>	<p>Populates the HashMap containing the tax details for the quote header and each quote line. Key is quote line ID for each quote line. Value in the HashMap is TaxDetailRecord[] containing tax details records.</p> <pre>public void setTaxDetails(com.sun.java.util.collections.HashMap taxDetails)</pre>
<code>submit(SubmitControlRecord)</code>	<p>Submits a quote to convert it into an order. This API should be called within a transaction block.</p> <pre>public oracle.apps.aso.quote.OrderHeaderRecord submit(oracle.apps.aso.quote.SubmitControlRecord submitControlRec)</pre> <p>throws FrameworkException, SQLException, QuoteException</p>
<code>submitPricingBatchRequest(ControlRecord)</code>	<p>Submits a pricing batch request for the quote. Quote header information, such as quote_header_id, pricing_status_indicator, tax_status_indicator, may be set in the quote object. The price_request_id will be populated in the quoteHeader object after the pricing batch request is submitted. This method should be called within a transaction block.</p> <pre>public void submitPricingBatchRequest(oracle.apps.aso.quote.ControlRecord controlRec)</pre> <p>throws FrameworkException, QuoteException, SQLException</p>
<code>toString()</code>	<p>Returns a String representation of the quote.</p> <pre>public java.lang.String toString()</pre>

Table 1–26 Methods for Class Quote

Method	Description
tradeIn(ControlRecord, InstanceRecord[])	<p>Adds the to-tradein installbase items to quote, given the instance records. This API should be called within a transaction block.</p> <pre>public void tradeIn(oracle.apps.aso.quote.ControlRecord controlRec, oracle.apps.aso.quote.InstanceRecord[] instan ceRec)throws SQLException, QuoteException, Fr ameworkException</pre>
unfreezePrice()	<p>Unfreezes the price of the quote by setting the price_frozen_date to null. The quote will be repriced. This API should be called within a transaction block.</p> <pre>public void unfreezePrice()throws FrameworkException, SQL Exception, QuoteException</pre>
unfreezePrice(ControlRecord)	<p>Unfreezes the price of the quote by setting the price_frozen_date to null. This API should be called within a transaction block.</p> <pre>public void unfreezePrice(oracle.apps.aso.quote.ControlRe cord controlRec) throws FrameworkException, SQLException, Quot eException</pre>
unpublish()	<p>Unpublishes a quote. This API should be called within a transaction block.</p> <pre>public void unpublish()throws FrameworkException, SQLExce ption, QuoteException</pre>
unpublish(ControlRecord)	<p>Unpublishes a quote. This API should be called within a transaction block.</p> <pre>public void unpublish(oracle.apps.aso.quote.ControlRecord controlRec) throws FrameworkException, SQLException, Quot eException</pre>

Table 1–26 Methods for Class Quote

Method	Description
updateObjectRelationship(RelatedObjectRecord)	<p>Updates object relationship based on the RelatedObjectRecord object passed in as parameter.</p> <pre>public static void updateObjectRelationship(oracle.apps.aso.quote. RelatedObjectRecord relatedObjectRec)throws FrameworkException, SQLException, QuoteException</pre>

addQuoteLines(ControlRecord, BigDecimal[], BigDecimal[], String[], String[], String[], BigDecimal[])

```
public void addQuoteLines(oracle.apps.aso.quote.ControlRecord controlRec,
java.math.BigDecimal[] itemIds, java.math.BigDecimal[] invOrgIds,
java.lang.String[] uomCodes, java.lang.String[] qtyStr,
java.lang.String[] itemTypes, java.math.BigDecimal[] priceListIds)
throws FrameworkException, SQLException, QuoteException
```

Adds lines to the quote. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

itemIds - Inventory item IDs for the lines to be added to the quote.

invOrgIds - Inventory organization IDs for the lines to be added to the quote.

uomCodes - Unit of measure codes for the lines to be added to the quote.

qtyStr - Quantities for the lines to be added to the quote.

itemTypes - Item type codes for the lines to be added to the quote. Possible values:

- QuoteLine.CONFIG_ITEM_TYPE - Indicates a configuration component.
- QuoteLine.MODEL_ITEM_TYPE - Indicates a model item.
- QuoteLine.SERVICE_ITEM_TYPE - Indicates a service item.
- QuoteLine.SERVICEABLE_ITEM_TYPE - Indicates a serviceable item.
- QuoteLine.STANDARD_ITEM_TYPE - Indicates a standard item.

priceListIds - Price list IDs for the lines to be added to the quote.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

addTemplatesToQuote(ControlRecord, BigDecimal[])

```
public void addTemplatesToQuote(oracle.apps.aso.quote.ControlRecord controlRec,  
java.math.BigDecimal[] templateIds)
```

throws FrameworkException, SQLException, QuoteException

Adds quote templates to a quote. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

templateIds - An array of IDs of the quote templates to be added to a quote.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

cancelPricingBatchRequest()

```
public void cancelPricingBatchRequest()
```

throws FrameworkException, QuoteException, SQLException

Cancels the current pricing batch request for the quote. Quote header information, such as quote_header_id, pricing_status_indicator, tax_status_indicator, should be set in the quote object. This method should be called within a transaction block.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteException - If an application error occurs.

java.sql.SQLException - If a database error occurs.

checkAvailability(AtpRecord[])

```
public void checkAvailability(oracle.apps.aso.quote.AtpRecord[] atpRecords)
```

throws FrameworkException, SQLException, QuoteException

Checks the ATP availability information for each quote line in the quote. This API should be called within a transaction block.

The input parameter atpRecords is an array of AtpRecord objects, each element of which would contain ATP information of each quote line in the quote respectively. So before calling this method, one should call getQuoteLines() to retrieve all quote lines in the quote, and then loop through them to initialize an AtpRecord object for each quote line. When initializing AtpRecord object for a quote line, if the need-by date of the line item is not the current date, one should set the requested_ship_date

of the `AtpRecord` object to the actual need-by date. Other ATP information in the `atpRecords` will be populated by the method.

After the method returns, availability information for all quote lines can be retrieved from `atpRecords`.

Parameters: `atpRecords` - An array of `AtpRecord` objects, each of which contains ATP information for each quote line in the quote respectively. The `requested_ship_date` should be populated in each `AtpRecord` in the array for the corresponding quote line if the need-by date of the line is not the current system date.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

`QuoteException` - If an application error occurs.

checkCredit()

```
public void checkCredit()
```

```
throws FrameworkException, SQLException, QuoteException, QuoteWarningException
```

Performs credit check on the quote for the customer.

The credit check process is performed by a PL/SQL procedure:

- If the procedure returns a **NULL** result, method throws a `FrameworkException` with “No credit check performed” error message;
- If the procedure returns **FAIL**, method throws a `QuoteWarningException` with messages returned by the procedure indicating the failure reason;
- If the procedure returns **PASS**, method returns without throwing any exceptions.

This API should be called within a transaction block.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

`QuoteException` - If an application error occurs.

`QuoteWarningException` - If a credit check returns **FAIL**.

Since: 11.5.8

checkCustomerAccounts()

```
public void checkCustomerAccounts()
```

```
throws SQLException, FrameworkException, QuoteException
```


Checks whether there are any missing accounts in the system for the sold-to, ship-to, or invoice-to customers specified in the quote. If there are no missing accounts, the method will return without error. If there are missing accounts, a `QuoteException` will be thrown. Quote header ID should be populated before calling this method.

Throws:

`java.sql.SQLException` - If a database error occurs.

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`QuoteException` - If a customer specified in the quote does not have an account in the system.

copy(CopyQuoteControlRecord, QuoteHeader)

```
public static oracle.apps.qot.core.Quote  
copy(oracle.apps.aso.quote.CopyQuoteControlRecord copyControlRec,  
oracle.apps.qot.core.QuoteHeader copyQuoteHdr)  
throws FrameworkException, SQLException, QuoteException
```

Creates a copy of an existing quote. The new quote will not be repriced. This API should be called within a transaction block.

Parameters:

`copyControlRec` - Control record indicating the quote information which should be copied.

`copyQuoteHdr` - Quote header containing the header information which should be passed to the copy quote API. Quote header ID of the quote to be copied should be passed in through this parameter.

Returns: A `Quote` object representing the new quote which has been created. Quote header ID and quote number will be populated.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

`QuoteException` - If an application error occurs.

create(ControlRecord)

```
public void create(oracle.apps.aso.quote.ControlRecord controlRec)  
throws FrameworkException, SQLException, QuoteException
```

Creates a quote with the attributes populated in this quote object. Quote header ID, last update date, and quote number will be populated in the quote header after the quote is created. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

createFromOpportunity(ControlRecord, String, OppQteInRec)

```
public static oracle.apps.got.core.Quote  
createFromOpportunity(oracle.apps.aso.quote.ControlRecord controlRec,  
java.lang.String sourceCode, oracle.apps.aso.quote.OppQteInRec oppQteInRec)  
throws FrameworkException, SQLException, QuoteException
```

Creates a new quote from an opportunity. The oppQteInRec object passed in should contain the opportunity information as well as the quote header information of the new quote. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

sourceCode - Source code for creating the quote.

oppQteInRec - Opportunity to quote input record.

Returns:

A Quote object with the following information populated in quoteHeader: opportunity_id, quote_header_id, quote_number, cust_account_id, party_id, currency_code.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

createNewVersion(CopyQuoteControlRecord, BigDecimal)

```
public static oracle.apps.got.core.Quote  
createNewVersion(oracle.apps.aso.quote.CopyQuoteControlRecord copyControlRec,  
java.math.BigDecimal quoteHdrId)  
throws FrameworkException, SQLException, QuoteException
```

Creates a new version of an existing quote. The new version of the quote will not be repriced. This API should be called within a transaction block.

Parameters:

copyControlRec - Control record indicating the quote information which should be copied.

quoteHdrId - Quote header ID of the quote to be copied.

Returns: A Quote object representing the new quote which has been created. Quote header ID and quote number will be populated.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

createNewVersion(CopyQuoteControlRecord, QuoteHeader)

```
public static oracle.apps.qot.core.Quote  
createNewVersion(oracle.apps.aso.quote.CopyQuoteControlRecord copyControlRec,  
oracle.apps.qot.core.QuoteHeader quoteHdr)  
throws FrameworkException, SQLException, QuoteException
```

Creates a new version of an existing quote. The new version of the quote will not be repriced. This API should be called within a transaction block.

Parameters:

copyControlRec - Control record indicating the quote information which should be copied.

quoteHdr - Quote header containing the header information which should be passed to the copy quote API. Quote header ID of the quote to be copied should be passed in through this parameter.

Returns: A Quote object representing the new quote which has been created. Quote header ID and quote number will be populated.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

deactivate(ControlRecord, BigDecimal[])

```
public void deactivate(oracle.apps.aso.quote.ControlRecord controlRec,  
java.math.BigDecimal[] quoteLineIds)  
throws SQLException, FrameworkException, QuoteException
```

Deactivates configuration components from the quote. If deactivate_flag in control record is T, all configuration components which have not been modified will be

deactivated. Otherwise, the quote line ids specified will be deactivated(removed) from quote. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

quoteLineIds - Quote line IDs that need to be deactivated from quote.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteException - If an application error occurs.

deleteQuotes(BigDecimal[])

```
public static void deleteQuotes(java.math.BigDecimal[] quoteHeaderIds)
```

```
throws FrameworkException, SQLException, QuoteException
```

Deletes quotes, given an array of quote header IDs. This API should be called within a transaction block.

Parameters: quoteHeaderIds - An array of quote header IDs of the quotes to delete.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

determineExpDate()

```
public static java.sql.Timestamp determineExpDate()
```

```
throws SQLException, QuoteException, FrameworkException
```

Determines the default quote expiration date.

Returns: The default quote expiration date.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

freezePrice()

```
public void freezePrice()
```

```
throws FrameworkException, SQLException, QuoteException
```

Freezes the price of the quote by setting the `price_frozen_date` to the current database system date. The quote will be repriced. This API should be called within a transaction block.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

`QuoteException` - If an application error occurs.

freezePrice(ControlRecord)

```
public void freezePrice(oracle.apps.aso.quote.ControlRecord controlRec)  
throws FrameworkException, SQLException, QuoteException
```

Freezes the price of the quote by setting the `price_frozen_date` to the current database system date. This API should be called within a transaction block.

Parameters: `controlRec` - Standard control record for updating a quote.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

`QuoteException` - If an application error occurs.

getNumberOfLines(BigDecimal)

```
public static java.math.BigDecimal getNumberOfLines(java.math.BigDecimal  
quoteHdrId)  
throws FrameworkException, SQLException
```

Returns the number of lines for the specified quote header ID.

Parameters: `quoteHdrId` - quote header ID.

Returns: The number of lines in the specified quote.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

getPriceAdjustments()

```
public com.sun.java.util.collections.HashMap getPriceAdjustments()
```

Returns a `HashMap` containing the total price adjustments for the quote header and each quote line. Key in the `HashMap` is `BigDecimal(-1)` for the quote header. Key is quote line ID for each quote line. Value in the `HashMap` contains the sum of the adjusted amount. `loadPriceAdjustments()` should be called before calling this API.

Returns: HashMap containing the total price adjustments for the quote header and each quote line.

getPropertyMap()

```
public com.sun.java.util.collections.HashMap getPropertyMap()
```

Returns the property map.

Returns: HashMap property map.

getQuoteHeader()

```
public oracle.apps.qot.core.QuoteHeader getQuoteHeader()
```

Returns the quote header.

Returns: Quote header object.

getQuoteLines()

```
public oracle.apps.qot.core.QuoteLine[] getQuoteLines()
```

Returns the quote lines.

Returns: Array of quote lines.

getTaxDetails()

```
public com.sun.java.util.collections.HashMap getTaxDetails()
```

Returns a HashMap containing the tax details for the quote header and each quote line. Key in the HashMap is BigDecimal(-1) for the quote header. Key is quote line ID for each quote line. Value in the HashMap is TaxDetailRecord[] containing tax detail records. loadTaxDetails() should be called before calling this API.

Returns: HashMap containing the tax details for the quote header and each quote line.

hasLineLevelBilling()

```
public boolean hasLineLevelBilling()
```

```
throws FrameworkException, SQLException
```

Returns whether the quote has line level billing.

Returns: Whether the quote has line level billing.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

list(SavedSearch, BigDecimal, int, int)

```
public static oracle.apps.qot.util.QueryResultSet
list(oracle.apps.qot.perzquery.SavedSearch savedSearch,
java.math.BigDecimal resourceId, int startIndex, int resCount)
throws FrameworkException, QueryBuilderException
```

Lists existing quotes or templates based on the search criteria, resourceId, start index and result count passed in as parameters.

The search criteria is provided in `savedSearch` object. Batch size for the query is also provided in `savedSearch` object.

Search parameters that can be used by consumers of `Quote.list()` API are:

Table 1–27 Parameters for Quote.list API

Parameter	Type	Meaning
quoteHeaderId	BigDecimal	Quote header ID
orderId	BigDecimal	Order ID of the quote
lastUpdateDate	Timestamp	Last updated date of the quote
lastUpdatedBy	String	Name of FND User who's last update the quote
creationDate	Timestamp	Quote creation date
createdBy	String	Name of FND User who's created quote
expirationDate	String	Quote expiration date
quoteName	String	Quote Name
quoteDescription	String	Quote Description
quoteNumber	BigDecimal	QuoteNumber
quoteVersion	BigDecimal	QuoteVersion
quoteAmount	BigDecimal	Total price of the quote
currencyCode	BigDecimal	Currency code of the quote
quoteStatusCode	String	Status code of the quote
quoteStatus	String	Translated quote status meaning
publishFlag	String	Web published status of the quote

Table 1–27 Parameters for Quote.list API

Parameter	Type	Meaning
publishMeaning	String	Lookup meaning of publish flag
opptyId	BigDecimal	Opportunity ID
opptyName	String	Opportunity Name
groupId	BigDecimal	Sales Group ID
groupName	String	Sales Group Name
resourceId	BigDecimal	ResourceId
resourceName	String	Resource Name
salesChannelCode	String	Sales Channel Code
salesChannel	String	Sales Channel Code Meaning
mktgSourceId	BigDecimal	Marketing Source ID
mktgSourceName	String	Marketing Source Name
partyId	BigDecimal	Party ID
partyType	String	Party Type
custAccountId	BigDecimal	Customer Account ID
custAccountNumber	String	Customer Account Number
customerPartyId	BigDecimal	Customer Party ID
customerName	String	Customer Name
customerType	String	Type of Customer
contactPartyId	BigDecimal	Contact Party ID
contactName	String	Contact Name (Search is done on First Name or Last Name)
contactFirstName	String	Contact First Name
contactLastName	String	Contact Last Name
partyReInId	BigDecimal	Party Relationship ID
party ReInType	String	Party Relationship Type

Table 1–27 Parameters for Quote.list API

Parameter	Type	Meaning
primaryResourceFlag	String	Primary Sales Resource. Indicates whether salesperson and salesgroup are to be primary.
quoteType	String	Type of quote. Indicates Template or Quote.
quoteSource	String	Quote source. Indicates whether sales representative, consumer or all quotes to be listed.

The following parameters are available as display columns, which can be set to savedSearch object:

quoteHeaderId, orderId, lastUpdateDate, lastUpdatedBy, creationDate, createdBy, expirationDate, quoteName, quoteNumber, quoteVersion, quoteAmount, currencyCode, quoteDescription, quoteType, quoteStatusCode, quoteStatus, publishFlag, publishMeaning, opptyId, opptyName, groupId, groupName, resourceId, resourceName, salesChannelCode, salesChannel, mktgSourceId, mktgSourceName, partyId, custAccountId, custAccountNumber, customerPartyId, customerName, customerType, contactPartyId, contactName, contactFirstName, contactLastName, partyReInId, partyReInType.

The following parameters are available as search columns, which can be set to savedSearch object:

quoteHeaderId, orderId, lastUpdateDate, lastUpdatedBy, creationDate, createdBy, expirationDate, quoteName, quoteNumber, quoteVersion, quoteDescription, quoteAmount, currencyCode, quoteStatusCode, publishFlag, opptyId, opptyName, groupId, resourceId, salesChannelCode, mktgSourceId, partyId, custAccountId, custAccountNumber, customerPartyId, customerName, customerType, contactPartyId, contactName, contactFirstName, contactLastName, partyReInId, partyReInType.

Additional search information like primary salesperson & sales group, type of quote, source of quote is provided by setting the following parameters to savedSearch object:

primaryResourceFlag, quoteType and quoteSource.

primaryResourceFlag: **Y, N** or **null**.

quoteType: **T, Q** or **null**.

quoteSource: **SALESREP, CONSUMER, ALL** or **null**.

The following parameters are available as sort columns, which can be set to savedSearch object:

quoteHeaderId, orderId, lastUpdateDate, lastUpdatedBy, creationDate, createdBy, expirationDate, quoteName, quoteNumber, quoteVersion, quoteAmount, currencyCode, quoteStatusCode, quoteDescription, quoteStatus, publishFlag, publishMeaning, opptyName, groupName, resourceName, salesChannelCode, salesChannel, customerName, mktgSourceId, mktgSourceName, partyId, custAccountId, custAccountNumber, customerPartyId, customerType, contactPartyId, contactName, contactFirstName, contactLastName, partyRelnId, partyRelnType.

Each quote header information that is retrieved from the database is stored as a property map in the quote object. The property map is a key-value combination. Key of the property map is any of the column mentioned above that is eligible for display. Value of the property map is the corresponding quote header value.

A simple example on how to retrieve quotes using Quote.list() API and to list the details of each quote is given below:

```
SavedSearch savedSearchObj = new SavedSearch();
String[] displayColumns = new String[] { "quoteName", "quoteNumber",
    "opptyName", "customerName" };
// set the columns (display) to be retrieved from the database
savedSearchObject.setDispColumnNameNames(displayColumns);
// set the batch size for the query
int batchSize = RequestCtx.getBatchSize();
savedSearchObject.setDisplayRows(batchSize);
// add the search conditions
savedSearchObject.addConditionParameter("quoteType", "Q");
savedSearchObject.addConditionParameter("quoteSource", "SALESREP");
savedSearchObject.addConditionParameter("quoteName", "LIKE", "testQ%");
savedSearchObject.addConditionParameter("opptyId", "1234");
savedSearchObject.addConditionParameter("quoteStatus", "APPROVED");
BigDecimal resourceId = RequestCtx.getResourceId();
int qotRsltCurrIndex = 1;
```

```
int qotRsltResCount = -1;
// conduct the quote search
QueryResultSet qrs = Quote.list(savedSearchObject, resourceId, qotRsltCurrIndex,
qotRsltResCount);
//list the quotes
Quotes[] quotes = (Quote[]) qrs.getQueryResult()
for (int i=0; i < quotes.length; i++) {
Quote quote = quotes[i];
//get the quote header detail property map
HashMap propertyMap = quote.getPropertyMap();
String quoteName = (String)propertyMap.get("quoteName");
String customerName = (String)propertyMap.get("customerName");
.....
.....
}
```

Parameters:

savedSearch - SavedSearch object containing search criteria and the batch size.

resourceId - Resource ID of the logged-in user. Pass null, if the logged-in user is not a valid resource.

startIndex - The start index indicating which row number to start retrieving data. -1 indicates the last set of rows.

resCount - The total number of objects to return. If -1, the count will be queried from the database.

Returns:

QueryResultSet object containing array of Quote objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

oracle.apps.qot.perzquery.QueryBuilderException - If an error occurs while executing the query.

listQuoteTemplates(String, String, int, int, int)

```
public static oracle.apps.qot.util.QueryResultSet  
listQuoteTemplates(java.lang.String searchColumn, java.lang.String searchString,  
int batchSize, int startIndex,  
int resCount)throws FrameworkException, SQLException
```

Obtains a list of existing quote templates.

Parameters:

searchColumn - The search criteria, possible values are:

- Quote.NAME - search by template name;
- Quote.DESCRPTION - search by template description.

searchString - The search string input by user.

batchSize - The batch size.

startIndex - The start index.

resCount - The result count.

Returns: A QueryResultSet object containing an array of quote templates satisfied the given search criteria.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

listVersions(BigDecimal, int, int, int)

```
public static oracle.apps.qot.util.QueryResultSet  
listVersions(java.math.BigDecimal quoteNumber, int batchSize, int startIndex,  
int resCount)throws FrameworkException, SQLException
```

Lists all versions for a particular quote number. Returns a QueryResultSet object containing the an array of Quote objects for the applicable quotes. In each Quote object returned with the QueryResultSet object, the quote header information will be populated.

Parameters:

quoteNumber - Quote number.

batchSize - The batch size for the query.

startIndex - The start index indicating which row number to start retrieving data. -1 indicates the last rows.

resCount - The total number of objects to return. If -1, the count will be queried from the database.

Returns: A QueryResultSet object containing an array of Quote objects of the versions for the quote number passed in as parameter. In each Quote object returned with the QueryResultSet object, the quote header information will be populated.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal, int)

```
public static oracle.apps.qot.core.Quote load(java.math.BigDecimal quoteHdrId,  
int level)throws FrameworkException, SQLException
```

Loads quote header information from the database based on quote header ID and load level passed in as parameter.

Parameters:

quoteHdrId - Quote header ID.

level - Load level which determines the quote header information which will be retrieved. Possible values:

- QuoteHeader.LEVEL10
- QuoteHeader.LEVEL20
- QuoteHeader.LEVEL90
- QuoteHeader.LEVEL100

Returns: Quote object with the appropriate fields in quoteHeader populated.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

loadAllLines(int, boolean)

```
public void loadAllLines(int level, boolean loadLineRel)  
throws FrameworkException, SQLException
```

Loads all the lines in the quote based on load level and whether to load line relationships. Line details for each quote line will be populated.

Parameters:

level - Load level indicating which line information will be loaded. Possible values are:

- QuoteLine.LEVEL10

- QuoteLine.LEVEL20
- QuoteLine.LEVEL90
- QuoteLine.LEVEL100

loadLineRel - Indicates whether or not to load line relationships. Possible values are:

- **true:**
 - Line relationships will be loaded;
 - The list of quote lines will contain top level products and service products. Configurations will be populated in a tree structure under the top level model;
 - Rollup list price, quote price and adjusted amount will be calculated for top level models and their configuration components. Rollup prices and adjusted amounts will reflect the total prices, taking quantities into account.
- **false:**
 - Line relationships will not be loaded;
 - The list of quote lines will be flat;
 - Rollup prices and adjusted amounts will not be calculated.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadAppliedTaxes()

```
public void loadAppliedTaxes()  
throws FrameworkException, SQLException
```

Loads the taxes that have been applied to the quote as a whole (sum of taxes applied to all lines for each tax code). After calling this API, quote.appliedTaxes will be populated with TaxDetailRecord[] containing the information for the applied taxes. The following information will be populated in the tax detail records: tax_code, tax_amount, tax_name.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadHighestVersion(BigDecimal, int)

```
public static oracle.apps.qot.core.Quote loadHighestVersion(java.math.BigDecimal  
quoteNumber, int level) throws FrameworkException, SQLException
```

Loads quote header information from the database for the highest version of the quote number passed in as parameter. Load level passed in as parameter will determine the quote header information which will be loaded.

Parameters:

quoteNumber - Quote number.

level - Load level which determines the quote header information which will be retrieved. Possible values:

- QuoteHeader.LEVEL10
- QuoteHeader.LEVEL20
- QuoteHeader.LEVEL90
- QuoteHeader.LEVEL100

Returns:

Quote object with the appropriate fields in the quote header populated.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

loadLineRelationships()

```
public void loadLineRelationships()  
throws FrameworkException, SQLException
```

Loads the line relationships for the quote. After calling this API, quote.lineRelns will be populated with LineRelationshipRecord[] containing the line relationship information.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadLines(int, HashMap, int, int)

```
public void loadLines(int level,  
com.sun.java.util.collections.HashMap expandList, int loadDepth,  
int loadStructure) throws FrameworkException, SQLException
```

Loads lines of a quote from database and structures the lines loaded based on line load level. Which lines to load is determined by load depth and the expand list. The loaded lines can be structured into a tree data structure or a flattened array of quote line objects. Service lines whose parents are quote lines will be loaded, but will not be put into the array of quote line objects. Instead they will be put into the array of service lines of their parent lines. Quote line details for each quote line will be populated.

Parameters:

level - The line load level indicating which line information should be loaded.

Possible values are:

- QuoteLine.LEVEL10
- QuoteLine.LEVEL20
- QuoteLine.LEVEL90
- QuoteLine.LEVEL100

expandList - A list of quote line IDs corresponding to the quote line objects for which we need to load the children. It is in the form of a hash table, the key being the quote line ID of the root of tree to be displayed, the value being an array containing the quote line IDs of the expanded nodes under the root. The first value in this array should be the root itself.

loadDepth - Signifies whether the entire tree of quote lines or only the first level children for the quote lines, whose quote line IDs are in the expandList, should be loaded. Possible values are:

- QuoteConstant.COMPLETE_DEPTH: the entire tree of quote lines should be loaded.
- QuoteConstant.PARTIAL_DEPTH: the first level children for the quote lines, whose quote line IDs are in the expandList, should be loaded.

loadStructure - Determines in which format the result of loaded lines should be returned. Possible values are:

- QuoteConstant.TREE_STRUCTURE: the quote lines will be structured into a tree data structure and returned.
- QuoteConstant.FLATTEN_STRUCTURE: the quote lines will be put into an array, ordered based on the tree structure (parent followed by its children) and returned.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

Since: 11.5.9

loadPrcAdjPerModifierLine(int, int)

```
public void loadPrcAdjPerModifierLine(int prcAdjFlags, int prcAdjTypes)
throws FrameworkException, SQLException
```

Loads total price adjustments for the whole quote per price modifier line. For each price modifier line, the sum of the adjusted amount will be determined. After price adjustments are loaded, the member variable `prcAdjPerModifierLine` will be populated with the following price adjustment information: `modifier_line_id`, `modifier_header_id`, `modifier_header_name`, `modifier_level_code`, `modifier_level`, `modifier_line_type_code`, `modifier_line_type`, `charge_type_code`, `charge_type`, `charge_subtype_code`, `charge_subtype`, `adjusted_amount`.

Parameters:

`prcAdjFlags` - flags indicating the conditions which must be met by the price adjustments. This parameter should be constructed using bitwise OR of the following possible values:

- `QuoteConstant.PRC_ADJ_ALL` - indicates all price adjustments
- `QuoteConstant.PRC_ADJ_APPLIED` - indicates applied price adjustments
- `QuoteConstant.PRC_ADJ_NON_AUTOMATIC` - indicates non-automatic price adjustments
- `QuoteConstant.PRC_ADJ_UPDATE_ALLOWED` - indicates updateable price adjustments. For example, `prcAdjFlags = QuoteConstant.PRC_ADJ_APPLIED | QuoteConstant.PRC_ADJ_NON_AUTOMATIC` indicates that the price adjustments should be applied and non-automatic

`prcAdjType` - flags indicating the type of price adjustments which should be included. The parameter should be constructed using bitwise OR of the following possible values:

- `QuoteConstant.PRC_ADJ_ALL` - indicates all price adjustments
- `QuoteConstant.PRC_ADJ_FREIGHT` - indicates freight/special charge price adjustments
- `QuoteConstant.PRC_ADJ_DIS` - indicates discount price adjustments
- `QuoteConstant.PRC_ADJ_OID` - indicates other item discount price adjustments
- `QuoteConstant.PRC_ADJ_PBH` - indicates price break header price adjustments

- `QuoteConstant.PRC_ADJ_SUR` - indicates surcharge price adjustment. For example, `prcAdjType = QuoteConstant.PRC_ADJ_FREIGHT | QuoteConstant.PRC_ADJ_DIS` indicates that price adjustments of type freight or type discount will be included

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

loadPriceAdjustments(int)

```
public void loadPriceAdjustments(int prcAdjFlags)
throws FrameworkException, SQLException
```

Loads total price adjustments for the whole quote. For the quote header and each quote line, the sum of the adjusted amount will be determined. After price adjustments are loaded, use the API `getPriceAdjustments()` to retrieve a `HashMap` containing the price adjustments. In the `HashMap`, the key is quote line ID and the value is total adjusted amount. For quote header, the key is `BigDecimal(-1)`.

Parameters:

`prcAdjFlags` - flags indicating the conditions which must be met by the price adjustments. This parameter should be constructed using bitwise OR of the following possible values:

- `QuoteConstant.PRC_ADJ_ALL` - Indicates all price adjustments.
- `QuoteConstant.PRC_ADJ_APPLIED` - Indicates applied price adjustments.
- `QuoteConstant.PRC_ADJ_NON_AUTOMATIC` - Indicates non-automatic price adjustments.
- `QuoteConstant.PRC_ADJ_UPDATE_ALLOWED` - Indicates updateable price adjustments. For example, `prcAdjFlags = QuoteConstant.PRC_ADJ_APPLIED | QuoteConstant.PRC_ADJ_NON_AUTOMATIC` indicates that the price adjustments should be applied and non-automatic

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

loadTaxDetails()

```
public void loadTaxDetails()
throws FrameworkException, SQLException
```

Loads tax details for the whole quote. For the quote header and each quote line, the tax details will be loaded. After tax details are loaded, use the API `getTaxDetails()` to

retrieve a HashMap containing the tax details. In the HashMap, the key is quote line ID, the value is TaxDetailRecord[] containing the tax details for the quote line. For the quote header, the key is BigDecimal(-1).

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

publish(BigDecimal, String, String, String, String, String)

```
public void publish(java.math.BigDecimal emailId, java.lang.String comment,  
java.lang.String storeName, java.lang.String storeURL,  
java.lang.String storeWebSite, java.lang.String fndUserName)  
throws FrameworkException, SQLException, QuoteException
```

Publishes a quote. The control record used for updating the quote will be defaulted to ControlRecord.OFF. This API should be called within a transaction block.

Parameters:

emailId - Email ID.

comment - Comment for publishing the quote.

storeName - Name of store where quote will be published.

storeURL - URL of store where quote will be published.

storeWebSite - Website of store where quote will be published.

fndUserName - User to whom the quote will be published.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

publish(ControlRecord, BigDecimal, String, String, String, String, String)

```
public void publish(oracle.apps.aso.quote.ControlRecord controlRec,  
java.math.BigDecimal emailId, java.lang.String comment,  
java.lang.String storeName, java.lang.String storeURL,  
java.lang.String storeWebSite, java.lang.String fndUserName)  
throws FrameworkException, SQLException, QuoteException
```

Publishes a quote. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

emailId - Email ID.

comment - Comment for publishing the quote.

storeName - Name of store where quote will be published.

storeURL - URL of store where quote will be published.

storeWebSite - Website of store where quote will be published.

findUserName - User to whom the quote will be published.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

reconfigure(ControlRecord, InstanceRecord[])

```
public void reconfigure(oracle.apps.aso.quote.ControlRecord controlRec,
```

```
oracle.apps.aso.quote.InstanceRecord[] instanceRec)
```

```
throws SQLException, FrameworkException, QuoteException
```

Adds the Install Base items to be reconfigured, to quote, given the instance records.

This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

instanceRec - Instance records containing instance ids and price list ids for the Install Base items to be reconfigured.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteException - If an application error occurs.

reprice()

```
public void reprice()
```

```
throws FrameworkException, SQLException, QuoteException
```

Reprices the quote. This API should be called within a transaction block.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

save(ControlRecord, int)

```
public void save(oracle.apps.aso.quote.ControlRecord controlRec, int level)  
throws FrameworkException, SQLException, QuoteException
```

Saves quote information to the database based on the data set in the quote object and the level indicating the information which should be saved. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

- level - Level indicating the quote information which should be saved. This parameter should be constructed using bitwise OR of the following possible values:
 - QuoteConstant.SAVE_HEADER - Indicates that basic quote header information should be saved.
 - QuoteConstnat.SAVE_LINE - Indicates that basic quote line information should be saved.
 - QuoteConstant.SAVE_LINE_DETAIL - Indicates that line detail information should be saved.
 - QuoteConstant.SAVE_LINE_REL - Indicates that line relationship information should be saved.
 - QuoteConstant.SAVE_PRC_ADJ - Indicates that price adjustment information should be saved.
 - QuoteConstant.SAVE_PRC_ATTR - Indicates that price attribute information should be saved.
 - QuoteConstant.SAVE_TAX - Indicates that tax information should be saved.
 - QuoteConstant.SAVE_PAYMENT - Indicates that payment information should be saved.
 - QuoteConstant.SAVE_SHIPMENT - Indicates that shipment information should be saved.
 - QuoteConstant.SAVE_SALES_CREDIT - Indicates that sales credit information should be saved. For example, level = QuoteConstant.SAVE_HEADER | QuoteConstant.SAVE_SHIPMENT | QuoteConstant.SAVE_PAYMENT indicates that quote header, header shipment and header payment information should be saved. For example, level = QuoteConstant.SAVE_HEADER | QuoteConstant.SAVE_LINE | QuoteConstant.SAVE_SHIPMENT | QuoteConstant.SAVE_PAYMENT indicates that quote header, header shipment,

header payment, quote lines, line shipment, and line payment information should be saved.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

setPriceAdjustments(HashMap)

public void **setPriceAdjustments**(com.sun.java.util.collections.HashMap priceAdjs)
Populates the HashMap containing the total price adjustments for the quote header and each quote line. Key is quote line ID for each quote line. Value in the HashMap contains the sum of the adjusted amount.

Parameters: priceAdjs - A HashMap containing the total price adjustments for the quote header and each quote line.

setPropertyMap(HashMap)

public void **setPropertyMap**(com.sun.java.util.collections.HashMap map)
Sets the property map.

Parameters: map - Property map.

setQuoteHeader(QuoteHeader)

public void **setQuoteHeader**(oracle.apps.qot.core.QuoteHeader header)
Sets the quote header.

Parameters: header - Quote header.

setQuoteLines(QuoteLine[])

public void **setQuoteLines**(oracle.apps.qot.core.QuoteLine[] lines)
Sets the quote lines.

Parameters: lines - Array of quote lines.

setTaxDetails(HashMap)

public void **setTaxDetails**(com.sun.java.util.collections.HashMap taxDetails)
Populates the HashMap containing the tax details for the quote header and each quote line. Key is quote line ID for each quote line. Value in the HashMap is TaxDetailRecord[] containing tax details records.

Parameters: taxDetails - A HashMap containing the tax details for the quote header and each quote line.

submit(SubmitControlRecord)

```
public oracle.apps.aso.quote.OrderHeaderRecord  
submit(oracle.apps.aso.quote.SubmitControlRecord submitControlRec)  
throws FrameworkException, SQLException, QuoteException
```

Submits a quote to convert it into an order. This API should be called within a transaction block.

Parameters: submitControlRec - Submits control information.

Returns: OrderHeaderRecord containing order header information: order number, order header ID, order request ID, order contract ID, order status.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

submitPricingBatchRequest(ControlRecord)

```
public void submitPricingBatchRequest(oracle.apps.aso.quote.ControlRecord  
controlRec) throws FrameworkException, QuoteException, SQLException  
Submits a pricing batch request for the quote. Quote header information, such as  
quote_header_id, pricing_status_indicator, tax_status_indicator,  
may be set in the quote object. The price_request_id will be populated in the  
quoteHeader object after the pricing batch request is submitted. This method  
should be called within a transaction block.
```

Parameters: controlRec - The standard control record for updating a quote.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteException - If an application error occurs.

java.sql.SQLException - If a database error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the quote.

Overrides: toString in class Object

Returns: A String representation of the quote.

tradeIn(ControlRecord, InstanceRecord[])

```
public void tradeIn(oracle.apps.aso.quote.ControlRecord controlRec,  
oracle.apps.aso.quote.InstanceRecord[] instanceRec)
```

throws SQLException, FrameworkException, QuoteException

Adds trade-in Install Base items to the quote, given the instance records. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

instanceRec - Instance records containing instance ids and price list ids for the Install Base items to be traded in.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteException - If an application error occurs.

unfreezePrice()

public void unfreezePrice()

throws FrameworkException, SQLException, QuoteException

Unfreezes the price of the quote by setting the price_frozen_date to null. The quote will be repriced. This API should be called within a transaction block.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

unfreezePrice(ControlRecord)

public void unfreezePrice(oracle.apps.aso.quote.ControlRecord controlRec)

throws FrameworkException, SQLException, QuoteException

Unfreezes the price of the quote by setting the price_frozen_date to null. This API should be called within a transaction block.

Parameters: controlRec - Standard control record for updating a quote.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

unpublish()

public void unpublish()throws FrameworkException, SQLException, QuoteException

Unpublishes a quote. The control record used for updating the quote will be defaulted to `ControlRecord.OFF`. This API should be called within a transaction block.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

`QuoteException` - If an application error occurs.

unpublish(ControlRecord)

```
public void unpublish(oracle.apps.aso.quote.ControlRecord controlRec)
throws FrameworkException, SQLException, QuoteException
```

Unpublishes a quote. This API should be called within a transaction block.

Parameters: `controlRec` - Standard control record for updating a quote.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

`QuoteException` - If an application error occurs.

updateObjectRelationship(RelatedObjectRecord)

```
public static void
updateObjectRelationship(oracle.apps.aso.quote.RelatedObjectRecord
relatedObjectRec) throws FrameworkException, SQLException, QuoteException
```

Updates object relationship based on the `RelatedObjectRecord` object passed in as parameter. Field `operation_code` in `RelatedObjectRecord` will be used to determine which type of operation will be performed on object relationship. Possible values of operation types are:

- `QuoteConstant.CREATE_OPCODE`: Create object relationship
- `QuoteConstant.UPDATE_OPCODE`: Update object relationship
- `QuoteConstant.DELETE_OPCODE`: Delete object relationship

If operation type is `QuoteConstant.CREATE_OPCODE`, then the `RelatedObjectRecord` object will be populated with related object ID. This API should be called within a transaction block.

Parameters: `relatedObjectRec` - Related object record.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

Since: 11.5.8

1.16 Class QuoteConstant

```
java.lang.Object
|
+--oracle.apps.qot.core.QuoteConstant
public class QuoteConstant
```

QuoteConstant contains constants that are shared by the classes in the package oracle.apps.qot.core

1.16.1 Fields for Class QuoteConstant

CASH_PAYMENT

```
public static final java.lang.String CASH_PAYMENT
```

Cash payment type.

CC_PAYMENT

```
public static final java.lang.String CC_PAYMENT
```

Credit Card payment type.

CHECK_PAYMENT

```
public static final java.lang.String CHECK_PAYMENT
```

Check payment type.

COMPLETE_DEPTH

```
public static final int COMPLETE_DEPTH
```

Indicates that the entire tree structure should be loaded in the display of a tree structure.

COMPLETE_PRC_TAX

```
public static final java.lang.String COMPLETE_PRC_TAX
```

Indicates that the pricing/tax status is complete.

CREATE_OPCODE

```
public static final java.lang.String CREATE_OPCODE
```

CREATE operation code.

CURRENT_QUOTE

public static final int **CURRENT_QUOTE**
Indicates service for current quote.

DELETE_OPCODE

public static final java.lang.String **DELETE_OPCODE**
DELETE operation code.

FLATTEN_STRUCTURE

public static final int **FLATTEN_STRUCTURE**
Indicates when a tree structure of objects is loaded, the results (structured tree objects) are returned in the form of an array of objects, ordered based on the tree structure (parent followed by its children).

INCOMPLETE_PRC_TAX

public static final java.lang.String **INCOMPLETE_PRC_TAX**
Indicates that the pricing/tax status is incomplete.

INSTALL_BASE

public static final int **INSTALL_BASE**
Indicates service for install base item.

INTERMEDIATE_NODE

public static final int **INTERMEDIATE_NODE**
Indicates that a node in the display of a tree structure is a intermediate node (having children nodes).

PARTIAL_DEPTH

public static final int **PARTIAL_DEPTH**
Indicates that only the first level children of the node to be expanded should be loaded in the display of a tree structure.

PENDING_ORDER

public static final int **PENDING_ORDER**
Indicates service for pending order line.

PRC_ADJ_ALL

public static final int **PRC_ADJ_ALL**

Indicates that all price adjustments should be loaded.

PRC_ADJ_APPLIED

public static final int **PRC_ADJ_APPLIED**

Indicates that applied price adjustments should be loaded.

PRC_ADJ_DIS

public static final int **PRC_ADJ_DIS**

Indicates that price adjustments with modifier line type code **DIS** (Discount) should be loaded.

PRC_ADJ_FREIGHT

public static final int **PRC_ADJ_FREIGHT**

Indicates that price adjustments with modifier line type code **FREIGHT_CHARGE** (Freight/Special Charge) should be loaded.

PRC_ADJ_NO_FREIGHT

public static final int **PRC_ADJ_NO_FREIGHT**

Indicates that price adjustments of all modifier line type codes, except **FREIGHT**.

PRC_ADJ_NON_AUTOMATIC

public static final int **PRC_ADJ_NON_AUTOMATIC**

Indicates that non-automatic price adjustments should be loaded.

PRC_ADJ_OID

public static final int **PRC_ADJ_OID**

Indicates that price adjustments with modifier line type code **OID** (Other Item Discount) should be loaded.

PRC_ADJ_PBH

public static final int **PRC_ADJ_PBH**

Indicates that price adjustments with modifier line type code **PBH** (Price Break Header) should be loaded.

PRC_ADJ_SUR

public static final int **PRC_ADJ_SUR**

Indicates that price adjustments with modifier line type code **SUR** (Surcharge) should be loaded.

PRC_ADJ_UPDATE_ALLOWED

```
public static final int PRC_ADJ_UPDATE_ALLOWED
```

Indicates that updateable price adjustments should be loaded.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

SAVE_HEADER

```
public static final int SAVE_HEADER
```

Indicates that basic header information should be saved. This value will be used for input to Quote.save() API.

SAVE_LINE

```
public static final int SAVE_LINE
```

Indicates that line information should be saved along with the basic quote header information. This value will be used to construct the input to Quote.save() API.

SAVE_LINE_DETAIL

```
public static final int SAVE_LINE_DETAIL
```

Indicates that line detail information should be saved along with the basic quote header information. This value will be used to construct the input to Quote.save() API.

SAVE_LINE_REL

```
public static final int SAVE_LINE_REL
```

Indicates that line relationship information should be saved along with the basic quote header information. This value will be used to construct the input to Quote.save() API.

SAVE_PAYMENT

```
public static final int SAVE_PAYMENT
```

Indicates that payment information should be saved along with the basic quote header information. This value will be used to construct the input to Quote.save() API.

SAVE_PRC_ADJ

```
public static final int SAVE_PRC_ADJ
```

Indicates that price adjustment information should be saved along with the basic quote header information. This value will be used to construct the input to Quote.save() API.

SAVE_PRC_ATTR

```
public static final int SAVE_PRC_ATTR
```

Indicates that price attribute information should be saved along with the basic quote header information. This value will be used to construct the input to Quote.save() API.

SAVE_SALES_CREDIT

```
public static final int SAVE_SALES_CREDIT
```

Indicates that sales credit information should be saved along with the basic quote header information. This value will be used to construct the input to Quote.save() API.

SAVE_SHIPMENT

```
public static final int SAVE_SHIPMENT
```

Indicates that shipment information should be saved along with the basic quote header information. This value will be used to construct the input to Quote.save() API.

SAVE_TAX

```
public static final int SAVE_TAX
```

Indicates that tax detail information should be saved along with the basic quote header information. This value will be used to construct the input to Quote.save() API.

TAX_EXEMPT

```
public static final java.lang.String TAX_EXEMPT
```

Tax Exempt status.

TAX_REQUIRE

```
public static final java.lang.String TAX_REQUIRE
```

Tax Require status.

TAX_STANDARD

```
public static final java.lang.String TAX_STANDARD
```

Tax Standard status.

TERMINAL_NODE

```
public static final int TERMINAL_NODE
```

Indicates that a node in the display of a tree structure is a terminal node (leaf node).

TREE_STRUCTURE

```
public static final int TREE_STRUCTURE
```

Indicates when a tree structure of objects is loaded, the results (structured tree objects) are returned in the form of a tree data structure.

UPDATE_OPCODE

```
public static final java.lang.String UPDATE_OPCODE
```

UPDATE operation code.

1.16.2 Constructors for Class QuoteConstant

QuoteConstant()

```
public QuoteConstant()
```

1.17 Class QuoteException

```
java.lang.Object
|
+--java.lang.Throwable
    |
    +--java.lang.Exception
        |
        +--oracle.apps.jtf.base.resources.FrameworkException
            |
            +--oracle.apps.qot.core.QuoteException
```

All Implemented Interfaces: java.io.Serializable

```
public class QuoteException extends
oracle.apps.jtf.base.resources.FrameworkException
```

QuoteException is thrown when an application error occurs in a method in package oracle.apps.qot.core or oracle.apps.qot.core.util.

1.17.1 Fields for Class QuoteException

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.17.2 Constructors for Class QuoteException

QuoteException(Exception, String)

```
public QuoteException(java.lang.Exception e, java.lang.String errorKey)
```

Constructs an exception with the given exception and errorKey.

Parameters:

e - The parent exception.

errorKey - Error key.

QuoteException(Exception, String, Hashtable)

```
public QuoteException(java.lang.Exception e, java.lang.String errorKey,  
java.util.Hashtable params)
```

Constructs an exception with the given exception, errorKey, and parameters.

Parameters:

e - The parent exception.

errorKey - Error key.

params - Hashtable of tokens for the errorKey.

QuoteException(Exception, String, Object[])

```
public QuoteException(java.lang.Exception e, java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs an exception with the given exception, errorKey, and parameters.

Parameters:

e - The parent exception.

errorKey - Error key.

params - An array of tokens for the error key.

QuoteException(Exception, String, String)

```
public QuoteException(java.lang.Exception e, java.lang.String errorKey,  
java.lang.String param)
```

Constructs an exception with the given exception, errorKey, and parameter.

Parameters:

e - The parent exception.

errorKey - Error key.

param - Token for the error key.

QuoteException(int, String)

```
public QuoteException(int err_msg_count, java.lang.String errorKey)  
throws FrameworkException
```

Constructs an exception with the message count and error key. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteException(int, String, Object[])

```
public QuoteException(int err_msg_count, java.lang.String errorKey,  
java.lang.Object[] params)  
throws FrameworkException
```

Constructs an exception with the message count, error key, and parameter tokens. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

params - An array of tokens for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteException(int, String, String)

```
public QuoteException(int err_msg_count, java.lang.String errorKey,  
java.lang.String param)  
throws FrameworkException
```

Construct an Exception with the message count, error key, and parameter token. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

param - A token for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteException(String)

```
public QuoteException(java.lang.String errorKey)
```

Constructs an exception with the errorKey.

Parameters: errorKey - error key corresponding to the error message

QuoteException(String, Object[])

```
public QuoteException(java.lang.String errorKey, java.lang.Object[] params)
```

Constructs an exception with the errorKey and parameters.

Parameters:

errorKey - Error key.

params - An array of tokens for the error key.

QuoteException(String, String)

```
public QuoteException(java.lang.String err_msg, java.lang.String errorKey)
```

Constructs an exception with the error message and error key.

Parameters:

err_msg - Error message.

errorKey - Error key.

QuoteException(String, String, Object[])

```
public QuoteException(java.lang.String err_msg, java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs an exception with the given error message, error key, and parameters.

Parameters:

err_msg - Error message.

errorKey - Error key.

params - An array of tokens for the error key.

QuoteException(String, String, String)

```
public QuoteException(java.lang.String errMsg, java.lang.String errorKey,
java.lang.String param)
```

Constructs an exception with the given error message, error key, and parameter.

Parameters:

errMsg - Error message.

errorKey - Error key.

param - Token for the error key.

1.18 Class QuoteHeader

```
java.lang.Object
|
+--oracle.apps.aso.quote.HeaderRecord
|
+--oracle.apps.qot.core.QuoteHeader
```

public class **QuoteHeader** extends oracle.apps.aso.quote.HeaderRecord

QuoteHeader object is used to model a quote header and its additional attributes, which include payments, price attributes, price adjustments, sales credits, shipments, and tax details. QuoteHeader contains the basic methods to retrieve and update quote headers in the database.

1.18.1 Fields for Class QuoteHeader

LEVEL10

```
public static final int LEVEL10
```

Level indicating that the following quote header information should be loaded: columns of aso_quote_headers_all, account_number, party_type, party_name, agreement_name, customer_id, customer_name, customer_type, sold_to_relationship_id, sold_to_relationship_code, contact_id, contact_name, opportunity_id, opportunity_name.

LEVEL100

```
public static final int LEVEL100
```

Level indicating that the following quote header information should be loaded: columns of aso_quote_headers_all, account_number, party_type, party_name,

agreement_name, customer_id, customer_name, customer_type, sold_to_relationship_id, sold_to_relationship_code, contact_id, contact_name, opportunity_id, opportunity_name, contract_template_number, contract_template_modifier, contract_template_name, order_number, primary_salesrep_name, marketing_source_name, invoice_to_account_number, invoice_to_relationship_id, invoice_to_relationship_code, invoice_to_contact_id, invoice_to_customer_id, invoice_to_customer_name, invoice_to_customer_type, invoice_to_customer_id, invoice_to_customer_name, invoice_to_customer_type, sold_to_addr_party_type, invoice_to_addr_party_type, is_sold_to_contact_addr, is_invoice_to_contact_addr, sold_to_address1, sold_to_address2, sold_to_address3, sold_to_address4, sold_to_city, sold_to_postal_code, sold_to_state, sold_to_county, sold_to_province, sold_to_country_code, sold_to_country, invoice_to_address1, invoice_to_address2, invoice_to_address3, invoice_to_address4, invoice_to_city, invoice_to_postal_code, invoice_to_state, invoice_to_county, invoice_to_province, invoice_to_country_code, invoice_to_country.

LEVEL20

```
public static final int LEVEL20
```

Level indicating that the following quote header information should be loaded: columns of aso_quote_headers_all, account_number, party_type, party_name, agreement_name, customer_id, customer_name, customer_type, sold_to_relationship_id, sold_to_relationship_code, contact_id, contact_name, opportunity_id, opportunity_name, contract_template_number, contract_template_modifier, contract_template_name.

LEVEL90

```
public static final int LEVEL90
```

Level indicating that the following quote header information should be loaded: columns of aso_quote_headers_all, account_number, party_type, party_name, agreement_name, customer_id, customer_name, customer_type, sold_to_relationship_id, sold_to_relationship_code, contact_id, contact_name, opportunity_id, opportunity_name, contract_template_number, contract_template_modifier, contract_template_name, order_number, primary_salesrep_name, marketing_source_name, invoice_to_account_number, invoice_to_relationship_id, invoice_to_relationship_code, invoice_to_contact_id, invoice_to_customer_id, invoice_to_customer_name, invoice_to_customer_type, invoice_to_customer_id, invoice_to_customer_name, invoice_to_customer_type, sold_to_addr_party_type, invoice_to_addr_party_type, is_sold_to_contact_addr, is_invoice_to_contact_addr.

payments

```
public oracle.apps.aso.quote.PaymentRecord[] payments
```

An array of payments for the quote header, corresponding to rows in ASO_PAYMENTS. The payments are used for retrieving and updating information in the

database. The member variable is populated after calling `loadPayments()` on a `QuoteHeader` instance.

priceAdjs

```
public oracle.apps.aso.quote.PriceAdjustmentRecord[] priceAdjs
```

An array of price adjustments for the quote header, corresponding to rows in `ASO_PRICE_ADJUSTMENTS`. The price adjustments are used for retrieving and updating information in the database. This member variable is populated after calling `loadPriceAdjustments()` on a `QuoteHeader` instance.

priceAttrs

```
public oracle.apps.aso.quote.PriceAttributeRecord[] priceAttrs
```

An array of price attributes for the quote header, corresponding to rows in `ASO_PRICE_ATTRIBUTES`. The price attributes are used for retrieving and updating information in the database. This member variable is populated after calling `loadPriceAttributes()` or `loadPromotions()` on a `QuoteHeader` instance.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

salesCredits

```
public oracle.apps.aso.quote.SalesCreditRecord[] salesCredits
```

An array of sales credits for the quote header, corresponding to rows in `ASO_SALES_CREDITS`. The sales credits are used for retrieving and updating information in the database. The member variable is populated after calling `loadSalesCredits()` on a `QuoteHeader` instance.

shipments

```
public oracle.apps.aso.quote.ShipmentRecord[] shipments
```

An array of shipments for the quote header, corresponding to rows in `ASO_SHIPMENTS`. The shipments are used for retrieving and updating information in the database. The member variable is populated after calling `loadShipments()` on a `QuoteHeader` instance.

taxDetails

```
public oracle.apps.aso.quote.TaxDetailRecord[] taxDetails
```

An array of tax details for the quote header, corresponding to rows in ASO_TAX_DETAILS. Tax detail for a quote header stores information regarding tax exemptions. The tax details are used for retrieving and updating information in the database. The member variable is populated after calling loadTaxDetails() on a QuoteHeader instance.

1.18.2 Constructors for Class QuoteHeader

QuoteHeader()

```
public QuoteHeader()
```

Default constructor.

QuoteHeader(BigDecimal)

```
public QuoteHeader(java.math.BigDecimal quoteHdrId)
```

Constructor which sets quote header ID.

Parameters: quoteHdrId - Quote header ID.

1.18.3 Methods for Class QuoteHeader

The following table is an index of the Class QuoteHeader methods:

Table 1–28 *Methods for Class QuoteHeader*

Methods	Description
getPriceAdjustment(int)	Returns a price adjustment record which satisfies the conditions indicated by the prcAdjFlags passed in as parameter. If no price adjustments meet the requirements, returns null. If multiple price adjustments meet the requirements, returns an applied price adjustment. If there are no applied price adjustments which meet the conditions, returns one of the price adjustments which satisfy the criteria. public oracle.apps.aso.quote.PriceAdjustmentRecord getPriceAdjustment (int prcAdjFlags) throws FrameworkException, SQLException

Table 1–28 Methods for Class QuoteHeader

Methods	Description
<code>getTotalPrcAdjOperand(int t)</code>	<p>Returns operand total for the price adjustments which satisfy the conditions indicated by the <code>prcAdjFlags</code> passed in as parameter. If no price adjustments meet the requirements, the value 0 is returned.</p> <pre>public java.math.BigDecimal getTotalPrcAdjOperand(int prcAdjFlags) throws FrameworkException, SQLException</pre>
<code>isHighestVersion()</code>	<p>Returns whether the quote header is the highest version.</p> <pre>public boolean isHighestVersion()</pre>
<code>load(BigDecimal, int)</code>	<p>Loads quote header information from the database based on quote header ID and load level passed in as parameter.</p> <pre>public static oracle.apps.qot.core.QuoteHeader load(java.math.BigDecimal quoteHdrId, int level) throws FrameworkException, SQLException</pre>
<code>loadHighestVersion(BigDecimal, int)</code>	<p>Loads quote header information from the database for the highest version of the quote number passed in as parameter. Load level passed in as parameter will determine the quote header information which will be loaded.</p> <pre>public static oracle.apps.qot.core.QuoteHeader loadHighestVersion(java.math.BigDecimal quoteNumber, int level) throws FrameworkException, SQLException</pre>
<code>loadPayments()</code>	<p>Loads the payment information from the database for the quote header and populates payments in the quote header object.</p> <pre>public void loadPayments() throws FrameworkException, SQLException</pre>
<code>loadPriceAdjustments(int)</code>	<p>Loads the price adjustment information from the database for price adjustments which satisfy the conditions indicated by <code>prcAdjFlags</code> passed in as parameter. Populates priceAdjs in the quote header object.</p> <pre>public void loadPriceAdjustments(int prcAdjFlags) throws FrameworkException, SQLException</pre>

Table 1–28 Methods for Class QuoteHeader

Methods	Description
loadPriceAttributes()	<p>Loads the price attribute information from the database for the quote header and populates priceAttrs in the quote header object.</p> <pre>public void loadPriceAttributes()throws FrameworkException , SQLException</pre>
loadPromotions()	<p>Loads the promotion information from the database for the quote header and populates priceAttrs in the quote header object.</p> <pre>public void loadPromotions()throws FrameworkException, SQL Exception</pre>
loadSalesCredits()	<p>Loads the sales credit information from the database for the quote header and populates salesCredits in the quote header object.</p> <pre>public void loadSalesCredits()throws FrameworkException, S QLException</pre>
loadShipments()	<p>Loads the shipment information from the database for the quote header and populates shipments in the quote header object.</p> <pre>public void loadShipments()throws FrameworkException, SQL Exception</pre>
loadTaxDetails()	<p>Loads the tax detail information from the database for the quote header and populates taxDetails in the quote header object.</p> <pre>public void loadTaxDetails()throws FrameworkException, SQL Exception</pre>
save(ControlRecord, int)	<p>Saves quote header information to the database. This API should be called within a transaction block.</p> <pre>public void save(oracle.apps.aso.quote.ControlRecord controlRec, int level) throws FrameworkException, SQLException, Quote Exception</pre>

Table 1–28 Methods for Class QuoteHeader

Methods	Description
setCashPayment(BigDecimal, BigDecimal, BigDecimal, BigDecimal, String)	<p>Sets the appropriate payment information in the quote header object for a cash payment.</p> <pre>public void setCashPayment (java.math.BigDecimal paymentId, java.math.BigDecimal billToCustAccId, java.math.BigDecimal billToContactPartyId, java.math.BigDecimal billToPartySiteId, java.math.BigDecimal paymentTermId, java.lang.String poNumber) throws QuoteException</pre>
setCCPayment(BigDecimal, BigDecimal, BigDecimal, String, CCPayment)	<p>Sets the appropriate payment information in the quote header object for a credit card payment.</p> <pre>public void setCCPayment (java.math.BigDecimal paymentId, java.math.BigDecimal billtoCustomerAccountId, java.math.BigDecimal billtoContactPartyId, java.math.BigDecimal billtoPartySiteId, java.math.BigDecimal paymentTermId, java.lang.String poNumber, oracle.apps.qot.core.util.CCPayment cc) throws QuoteException</pre>
setCheckPayment(BigDecimal, BigDecimal, BigDecimal, String, String)	<p>Sets the appropriate payment information in the quote header object for a check payment.</p> <pre>public void setCheckPayment (java.math.BigDecimal paymentId, java.math.BigDecimal billtoCustomerAccountId, java.math.BigDecimal billtoContactPartyId, java.math.BigDecimal billtoPartySiteId, java.math.BigDecimal paymentTermId, java.lang.String poNumber, java.lang.String checkNumber) throws QuoteException</pre>

Table 1–28 Methods for Class QuoteHeader

Methods	Description
setPayment(BigDecimal, BigDecimal, BigDecimal, String, BigDecimal, String, String, CCPayment)	Sets the appropriate payment information in the quote header object. <pre>public void setPayment(java.math.BigDecimal paymentId, java.math.BigDecimal billtoCustomerAccountId, java.math.BigDecimal billtoContactPartyId, java.math.BigDecimal billtoPartySiteId, java.lang.String paymentType, java.math.BigDecimal paymentTermId, java.lang.String poNumber, java.lang.String checkNumber, oracle.apps.qot.core.util.CCPayment cc) throws QuoteException</pre>
setShipment(BigDecimal, BigDecimal, BigDecimal, String, Timestamp, String, String)	Sets the appropriate shipment information in the quote header object. <pre>public void setShipment(java.math.BigDecimal shipmentId, java.math.BigDecimal shipToCustAcctId, java.math.BigDecimal shipToContactPartyId, java.math.BigDecimal shipToPartySiteId, java.lang.String shippingMethod, java.sql.Timestamp requestedDeliveryDate, java.lang.String shippingInstructions, java.lang.String packingInstructions) throws QuoteException</pre>
setTaxDetail(BigDecimal, String, String, String)	Sets the appropriate tax detail information in the quote header object. <pre>public void setTaxDetail(java.math.BigDecimal taxDetailId, java.lang.String taxExemptStatus, java.lang.String taxExemptReason, java.lang.String taxExemptNumber) throws QuoteException</pre>
toString()	Returns a String representation of the quote header. <pre>public java.lang.String toString()</pre>

getPriceAdjustment(int)

```
public oracle.apps.aso.quote.PriceAdjustmentRecord getPriceAdjustment(int
prcAdjFlags)
throws FrameworkException, SQLException
```

Returns a price adjustment record which satisfies the conditions indicated by the `prcAdjFlags` passed in as parameter. If no price adjustments meet the requirements, returns null. If multiple price adjustments meet the requirements, returns an applied price adjustment. If there are no applied price adjustments which meet the conditions, returns one of the price adjustments which satisfy the criteria.

Parameters: `prcAdjFlags` - flags indicating the conditions which must be met by the price adjustments. This parameter should be constructed using bitwise OR of the following possible values:

- `QuoteConstant.PRC_ADJ_ALL` - Indicates all price adjustments.
- `QuoteConstant.PRC_ADJ_APPLIED` - Indicates applied price adjustments.
- `QuoteConstant.PRC_ADJ_NON_AUTOMATIC` - Indicates non-automatic price adjustments.
- `QuoteConstant.PRC_ADJ_UPDATE_ALLOWED` - Indicates updateable price adjustments.

For example, `prcAdjFlags = QuoteConstant.PRC_ADJ_APPLIED | QuoteConstant.PRC_ADJ_NON_AUTOMATIC` indicates the price adjustments should be applied and non-automatic.

Returns: `PriceAdjustmentRecord` which satisfies the specified conditions.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

getTotalPrcAdjOperand(int)

```
public java.math.BigDecimal getTotalPrcAdjOperand(int prcAdjFlags)
throws FrameworkException, SQLException
```

Returns operand total for the price adjustments which satisfy the conditions indicated by the `prcAdjFlags` passed in as parameter. If no price adjustments meet the requirements, value 0 is returned

Parameters: `prcAdjFlags` - flags indicating the conditions which must be met by the price adjustments. This parameter should be constructed using bitwise OR of the following possible values:

- `QuoteConstant.PRC_ADJ_ALL` - Indicates all price adjustments.
- `QuoteConstant.PRC_ADJ_APPLIED` - Indicates applied price adjustments.
- `QuoteConstant.PRC_ADJ_NON_AUTOMATIC` - Indicates non-automatic price adjustments.

- `QuoteConstant.PRC_ADJ_UPDATE_ALLOWED` - Indicates updateable price adjustments.

For example, `prcAdjFlags = QuoteConstant.PRC_ADJ_APPLIED | QuoteConstant.PRC_ADJ_NON_AUTOMATIC` indicates that the price adjustments should be applied and non-automatic

Returns: Contains the sum of operands for the price adjustments satisfying the specified conditions.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - if a database error occurs.

isHighestVersion()

```
public boolean isHighestVersion()
```

Returns whether the quote header is the highest version.

Returns: Whether the quote header is the highest version.

load(BigDecimal, int)

```
public static oracle.apps.qot.core.QuoteHeader load(java.math.BigDecimal quoteHdrId, int level)
```

```
throws FrameworkException, SQLException
```

Loads quote header information from the database based on quote header ID and load level passed in as parameter.

Parameters: `quoteHdrId` - Quote header ID.

`level` - Load level which determines the quote header information which will be retrieved. Possible values:

- `QuoteHeader.LEVEL10`
- `QuoteHeader.LEVEL20`
- `QuoteHeader.LEVEL90`
- `QuoteHeader.LEVEL100`

Returns: `QuoteHeader` object with the appropriate fields populated.

Throws:

`java.sql.SQLException` - If a database error occurs.

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

loadHighestVersion(BigDecimal, int)

```
public static oracle.apps.qot.core.QuoteHeader  
loadHighestVersion(java.math.BigDecimal quoteNumber, int level)  
throws FrameworkException, SQLException
```

Loads quote header information from the database for the highest version of the quote number passed in as parameter. Load level passed in as parameter will determine the quote header information which will be loaded.

Parameters:

quoteNumber - Quote number.

level - Load level which determines the quote header information which will be retrieved. Possible values:

- QuoteHeader.LEVEL10
- QuoteHeader.LEVEL20
- QuoteHeader.LEVEL90
- QuoteHeader.LEVEL100

Returns: QuoteHeader object with the appropriate fields populated.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

loadPayments()

```
public void loadPayments()  
throws FrameworkException, SQLException
```

Loads the payment information from the database for the quote header and populates payments in the quote header object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadPriceAdjustments(int)

```
public void loadPriceAdjustments(int prcAdjFlags)  
throws FrameworkException, SQLException
```

Loads the price adjustment information from the database for price adjustments which satisfy the conditions indicated by prcAdjFlags passed in as parameter. Populates priceAdjs in the quote header object.

Parameters:

prcAdjFlags - Flags indicating the conditions which must be met by the price adjustments. This parameter should be constructed using bitwise OR of the following possible values:

- QuoteConstant.PRC_ADJ_ALL - Indicates all price adjustments.
- QuoteConstant.PRC_ADJ_APPLIED - Indicates applied price adjustments.
- QuoteConstant.PRC_ADJ_NON_AUTOMATIC - Indicates non-automatic price adjustments.
- QuoteConstant.PRC_ADJ_UPDATE_ALLOWED - Indicates updateable price adjustments.

For example, prcAdjFlags = QuoteConstant.PRC_ADJ_APPLIED | QuoteConstant.PRC_ADJ_NON_AUTOMATIC indicates the price adjustments should be applied and non-automatic.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadPriceAttributes()

```
public void loadPriceAttributes()  
throws FrameworkException, SQLException
```

Loads the price attribute information from the database for the quote header and populates priceAttrs in the quote header object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadPromotions()

```
public void loadPromotions()  
throws FrameworkException, SQLException
```

Loads the promotion information from the database for the quote header and populates priceAttrs in the quote header object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadSalesCredits()

```
public void loadSalesCredits()
```

throws `FrameworkException`, `SQLException`

Loads the sales credit information from the database for the quote header and populates salesCredits in the quote header object.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

loadShipments()

public void `loadShipments()`

throws `FrameworkException`, `SQLException`

Loads the shipment information from the database for the quote header and populates shipments in the quote header object.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

loadTaxDetails()

public void `loadTaxDetails()`

throws `FrameworkException`, `SQLException`

Loads the tax detail information from the database for the quote header and populates taxDetails in the quote header object.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

save(ControlRecord, int)

public void `save`(`oracle.apps.aso.quote.ControlRecord controlRec`, `int level`)

throws `FrameworkException`, `SQLException`, `QuoteException`

Saves quote header information to the database. This API should be called within a transaction block.

Parameters:

`controlRec` - Standard control record for updating a quote.

`level` - Level indicating the quote header information which should be saved. This parameter should be constructed using bitwise OR of the following possible values:

- `QuoteConstant.SAVE_HEADER` - Indicates that basic quote header information should be saved.

- `QuoteConstant.SAVE_PRC_ADJ` - Indicates that price adjustment information populated in the quote header object should be saved.
- `QuoteConstant.SAVE_PRC_ATTR` - Indicates that price attribute information populated in the quote header object should be saved.
- `QuoteConstant.SAVE_TAX` - Indicates that tax information populated in the quote header object should be saved.
- `QuoteConstant.SAVE_PAYMENT` - Indicates that payment information populated in the quote header object should be saved.
- `QuoteConstant.SAVE_SHIPMENT` - Indicates that shipment information populated in the quote header object should be saved.
- `QuoteConstant.SAVE_SALES_CREDIT` - Indicates that sales credit information populated in the quote header object should be saved. For example, `level = QuoteConstant.SAVE_SHIPMENT | QuoteConstant.SAVE_PAYMENT` indicates that shipment and payment information should be saved along with quote header basic information.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

`QuoteException` - If an application error occurs.

setCashPayment(BigDecimal, BigDecimal, BigDecimal, BigDecimal, BigDecimal, String)

```
public void setCashPayment(java.math.BigDecimal paymentId,  
java.math.BigDecimal billToCustAccId,  
java.math.BigDecimal billToContactPartyId,  
java.math.BigDecimal billToPartySiteId,  
java.math.BigDecimal paymentTermId, java.lang.String poNumber)  
throws QuoteException
```

Sets the appropriate payment information in the quote header object for a cash payment.

Parameters:

`paymentId` - Payment ID, the primary key in `ASO_PAYMENTS`, if there is an existing payment for the quote header.

`billToCustAccId` - Bill To customer account ID.

billToContactPartyId - Bill To contact party ID corresponding to the party ID of the contact relationship (a row in HZ_PARTIES with party_type PARTY_RELATIONSHIP) if there is a Bill To contact.

billToPartySiteId - Bill To party site ID for the bill-to address.

paymentTermId - Payment term ID.

poNumber - Purchase order number.

Throws: QuoteException - If an application error occurs.

setCCPayment(BigDecimal, BigDecimal, BigDecimal, BigDecimal, BigDecimal, String, CCPayment)

```
public void setCCPayment(java.math.BigDecimal paymentId,
    java.math.BigDecimal billtoCustomerAccountId,
    java.math.BigDecimal billtoContactPartyId,
    java.math.BigDecimal billtoPartySiteId,
    java.math.BigDecimal paymentTermId, java.lang.String poNumber,
    oracle.apps.qot.core.util.CCPayment cc)
    throws QuoteException
```

Sets the appropriate payment information in the quote header object for a credit card payment.

Parameters:

paymentId - Payment ID, the primary key in ASO_PAYMENTS, if there is an existing payment for the quote header.

billtoCustomerAccountId - Bill To customer account ID.

billToContactPartyId - Bill To contact party ID corresponding to the party ID of the contact relationship (a row in HZ_PARTIES with party_type PARTY_RELATIONSHIP) if there is a bill-to contact.

billToPartySiteId - Bill To party site ID for the bill-to address.

paymentTermId - Payment term ID.

poNumber - Purchase order number.

cc - Credit card payment information.

Throws: QuoteException - If an application error occurs.

setCheckPayment(BigDecimal, BigDecimal, BigDecimal, BigDecimal, BigDecimal, String, String)

```
public void setCheckPayment(java.math.BigDecimal paymentId,
    java.math.BigDecimal billtoCustomerAccountId,
    java.math.BigDecimal billtoContactPartyId,
```

```
java.math.BigDecimal billtoPartySiteId,  
java.math.BigDecimal paymentTermId, java.lang.String poNumber,  
java.lang.String checkNumber)  
throws QuoteException
```

Sets the appropriate payment information in the quote header object for a check payment.

Parameters:

paymentId - Payment ID, the primary key in ASO_PAYMENTS, if there is an existing payment for the quote header.

billtoCustomerAccountId - Bill T customer account ID.

billtoContactPartyId - Bill To contact party ID corresponding to the party ID of the contact relationship (a row in HZ_PARTIES with party_type PARTY_RELATIONSHIP) if there is a bill-to contact.

billtoPartySiteId - Bill To party site ID for the bill-to address.

paymentTermId - Payment Term ID.

poNumber - Purchase order number.

checkNumber - Check number.

Throws: QuoteException - If an application error occurs.

setPayment(BigDecimal, BigDecimal, BigDecimal, BigDecimal, String, BigDecimal, String, String, CCPayment)

```
public void setPayment(java.math.BigDecimal paymentId,  
java.math.BigDecimal billtoCustomerAccountId,  
java.math.BigDecimal billtoContactPartyId,  
java.math.BigDecimal billtoPartySiteId, java.lang.String paymentType,  
java.math.BigDecimal paymentTermId, java.lang.String poNumber,  
java.lang.String checkNumber, oracle.apps.got.core.util.CCPayment cc)  
throws QuoteException
```

Sets the appropriate payment information in the quote header object.

Parameters:

paymentId - Payment ID, the primary key in ASO_PAYMENTS, if there is an existing payment for the quote header.

billtoCustomerAccountId - Bill To customer account ID.

billtoContactPartyId - Bill-to contact party ID corresponding to the party ID of the contact relationship (a row in HZ_PARTIES with party_type PARTY_RELATIONSHIP) if there is a bill-to contact.

billtoPartySiteId - Bill To party site ID for the bill To address.

paymentType - Payment type. Possible values:

- QuoteConstant.CHECK_PAYMENT - Indicates check payment.
- QuoteConstant.CASH_PAYMENT - Indicates cash payment.
- QuoteConstant.CC_PAYMENT - Indicates credit card payment.
- null - Indicates invoice payment.

paymentTermId - Payment term ID.

poNumber - Purchase order number.

checkNumber - Check number if the payment type is **CHECK_PAYMENT**.

ccPayment - Credit card payment information, including credit card type, credit card number, card holder name, expiration date, if the payment type is **CC_PAYMENT**.

Throws: QuoteException - If an application error occurs.

setShipment(BigDecimal, BigDecimal, BigDecimal, BigDecimal, String, Timestamp, String, String)

```
public void setShipment(java.math.BigDecimal shipmentId,
    java.math.BigDecimal shipToCustAcctId,
    java.math.BigDecimal shipToContactPartyId,
    java.math.BigDecimal shipToPartySiteId, java.lang.String shippingMethod,
    java.sql.Timestamp requestedDeliveryDate,
    java.lang.String shippingInstructions,
    java.lang.String packingInstructions)
    throws QuoteException
```

Sets the appropriate shipment information in the quote header object.

Parameters:

shipmentId - Shipment ID, the primary key in ASO_SHIPMENTS, if there is an existing shipment for the quote header

shipToCustomerAcctId - Ship To customer account ID.

shipToContactPartyId - Ship To contact party ID corresponding to the party ID of the contact relationship (a row in HZ_PARTIES with party_type PARTY_RELATIONSHIP) if there is a ship-to contact.

shipToPartySiteId - Ship To party site ID for the ship-to address.

shippingMethod - Shipping method.

requestedDeliveryDate - Requested delivery date.

shippingInstructions - Shipping instructions.

packingInstructions - Packing instructions.

Throws: QuoteException - If there is an application error.

setTaxDetail(BigDecimal, String, String, String)

```
public void setTaxDetail(java.math.BigDecimal taxDetailId,  
java.lang.String taxExemptStatus, java.lang.String taxExemptReason,  
java.lang.String taxExemptNumber)  
throws QuoteException
```

Sets the appropriate tax detail information in the quote header object.

Parameters:

taxDetailId Tax Detail ID, the primary key in ASO_TAX_DETAILS, if there is an existing tax detail record for the quote header.

taxExemptStatus - Tax Exempt Status. Possible values:

- QuoteConstant.TAX_EXEMPT
- QuoteConstant.TAX_REQUIRE
- QuoteConstant.TAX_STANDARD

taxExemptReason - Tax Exempt Reason if the tax exempt status is **TAX_EXEMPT**.

taxExemptNumber - Tax Exempt Number if the tax exempt status is **TAX_EXEMPT**.

Throws: QuoteException - If an application error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the quote header.

Overrides: toString in class HeaderRecord

Returns: A String representation of the quote header.

1.19 Class QuoteLine

```
java.lang.Object  
|  
+--oracle.apps.aso.quote.LineRecord  
|  
+--oracle.apps.qot.core.QuoteLine
```

```
public class QuoteLine extends oracle.apps.aso.quote.LineRecord
```

QuoteLine object is used to model a quote line and its additional attributes, which include line details, payments, shipments, price attributes, price adjustments, sales

credits, shipments, and tax details. QuoteLine contains the basic methods to retrieve and update quote lines in the database.

1.19.1 Fields for Class QuoteLine

CONFIG_ITEM_TYPE

```
public static final java.lang.String CONFIG_ITEM_TYPE
```

Configuration item type.

LEVEL10

```
public static final int LEVEL10
```

Level indicating that the following quote line information should be loaded: columns of `aso_quote_lines_all`, `item_part_number`, `item_description`, `is_serviceable`, `is_service_item`.

LEVEL100

```
public static final int LEVEL100
```

Level indicating that the following quote line information should be loaded: columns of `aso_quote_lines_all`, `item_part_number`, `item_description`, `is_serviceable`, `is_service_item`, `ship_quote_price`, `agreement_name`, `commitment_number`, `invoice_to_account_number`, `invoice_to_relationship_id`, `invoice_to_relationship_code`, `invoice_to_contact_id`, `invoice_to_contact_name`, `invoice_to_customer_id`, `invoice_to_customer_name`, `invoice_to_customer_type`, `invoice_to_addr_party_type`, `is_invoice_to_contact_addr`, `invoice_to_address1`, `invoice_to_address2`, `invoice_to_address3`, `invoice_to_address4`, `invoice_to_city`, `invoice_to_postal_code`, `invoice_to_state`, `invoice_to_county`, `invoice_to_province`, `invoice_to_country_code`, `invoice_to_country`.

LEVEL20

```
public static final int LEVEL20
```

Level indicating that the following quote line information should be loaded: columns of `aso_quote_lines_all`, `item_part_number`, `item_description`, `is_serviceable`, `is_service_item`, `ship_quote_price`.

LEVEL90

```
public static final int LEVEL90
```

Level indicating that the following quote line information should be loaded: columns of `aso_quote_lines_all`, `item_part_number`, `item_description`, `is_serviceable`, `is_service_item`, `ship_quote_price`, `agreement_name`, `commitment_number`, `invoice_to_account_number`, `invoice_to_relationship_id`, `invoice_to_relationship_code`, `invoice_to_contact_id`, `invoice_to_contact_name`, `invoice_to_customer_id`, `invoice_to`

`to_customer_name`, `invoice_to_customer_type`, `invoice_to_addr_party_type`, `is_invoice_to_contact_addr`.

lineDetails

```
public oracle.apps.aso.quote.LineDetailRecord[] lineDetails
```

An array of line details for the quote line, corresponding to rows in ASO_QUOTE_LINE_DETAILS. The line details are used for retrieving and updating information in the database. This member variable is populated after calling `loadLineDetails()` on a `QuoteLine` instance. Line details are also populated in `QuoteLine` objects by the `loadLines()` API in `Quote`.

lineReIns

```
public oracle.apps.aso.quote.LineRelationshipRecord[] lineReIns
```

An array of line relationships for the quote line, corresponding to rows in ASO_LINE_RELATIONSHIPS. This member variable is currently not supported.

MODEL_ITEM_TYPE

```
public static final java.lang.String MODEL_ITEM_TYPE
```

Model item type.

payments

```
public oracle.apps.aso.quote.PaymentRecord[] payments
```

An array of payments for the quote line, corresponding to rows in ASO_PAYMENTS. The payments are used for retrieving and updating information in the database. The member variable is populated after calling `loadPayments()` on a `QuoteLine` instance.

priceAdjs

```
public oracle.apps.aso.quote.PriceAdjustmentRecord[] priceAdjs
```

An array of price adjustments for the quote line, corresponding to rows in ASO_PRICE_ADJUSTMENTS. The price adjustments are used for retrieving and updating information in the database. This member variable is populated after calling `loadPriceAdjustments()` on a `QuoteLine` instance.

priceAttrrs

```
public oracle.apps.aso.quote.PriceAttributeRecord[] priceAttrrs
```

An array of price attributes for the quote line, corresponding to rows in ASO_PRICE_ATTRIBUTES. The price attributes are used for retrieving and updating information in the database. This member variable is populated after calling `loadPriceAttributes()` or `loadPromotions()` on a `QuoteLine` instance.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

salesCredits

```
public oracle.apps.aso.quote.SalesCreditRecord[] salesCredits
```

An array of sales credits for the quote line, corresponding to rows in ASO_SALES_CREDITS. The sales credits are used for retrieving and updating information in the database. The member variable is populated after calling loadSalesCredits() on a QuoteLine instance.

SERVICE_ITEM_TYPE

```
public static final java.lang.String SERVICE_ITEM_TYPE
```

Service item type.

SERVICEABLE_ITEM_TYPE

```
public static final java.lang.String SERVICEABLE_ITEM_TYPE
```

Serviceable item type.

shipments

```
public oracle.apps.aso.quote.ShipmentRecord[] shipments
```

An array of shipments for the quote line, corresponding to rows in ASO_SHIPMENTS. The shipments are used for retrieving and updating information in the database. The member variable is populated after calling loadShipments() on a QuoteLine instance.

STANDARD_ITEM_TYPE

```
public static final java.lang.String STANDARD_ITEM_TYPE
```

Standard item type.

taxDetails

```
public oracle.apps.aso.quote.TaxDetailRecord[] taxDetails
```

An array of tax details for the quote line, corresponding to rows in ASO_TAX_DETAILS. The tax details for a quote line contains information regarding taxes applied to the quote line. The tax details are used only for retrieving information in the database. The member variable is populated after calling loadTaxDetails() on a QuoteLine instance.

1.19.2 Constructors for Class QuoteLine

QuoteLine()

`public QuoteLine()`
Default constructor.

QuoteLine(BigDecimal, BigDecimal)

`public QuoteLine(java.math.BigDecimal quoteHdrId,
java.math.BigDecimal quoteLineId)`

Construct a quote line with the quote header ID and quote line ID passed in as parameter.

Parameters:

quoteHdrId - Quote header ID.

quoteLineId - Quote line ID.

1.19.3 Methods for Class QuoteLine

The following table is an index of the Class QuoteLine methods:

Table 1–29 *Methods for Class QuoteLine*

Method	Description
addServices	<p>Adds services for this quote line. The following attributes should be set in the quote line before calling this API: quote_header_id, quote_line_id, quantity. The quantity for the service lines will be defaulted from the quote line.</p> <pre>public void addServices(oracle.apps.aso.quote.ControlRecord controlRec, java.math.BigDecimal[] srvcItemIds, java.math.BigDecimal[] invOrgIds, java.lang.String[] uomCodes, java.sql.Timestamp[] startDate, java.math.BigDecimal[] srvcDuration, java.lang.String[] srvcPeriodCode, java.math.BigDecimal[] priceListIds) throws FrameworkException, SQLException, QuoteException</pre>

Table 1–29 Methods for Class QuoteLine

Method	Description
<code>addToContainer(ControlRecord, InstanceRecord[])</code>	<p>Adds the Install Base item instances to the model container in the quote. This API should be called within a transaction block.</p> <pre>public void addToContainer(oracle.apps.aso.quote.ControlRecord controlRec, oracle.apps.aso.quote.InstanceRecord[] instanceRec) throws SQLException, QuoteException, FrameworkException</pre>
<code>copy(CopyQuoteControlRecord, ControlRecord)</code>	<p>Copies the quote line into a new quote line within the same quote with the a CopyQuoteControlRecord object passed in as parameter.</p> <p>If the quote line is a configured model line, the entire configuration is copied together with the model line. The configuration structure is preserved in the newly created model.</p> <p>If services are associated to the quote line or to one or more component items in the configuration (if the quote line is a configured model), then services are copied as well. The newly created service items are associated to the corresponding serviceable items.</p> <p>Other line-level attributes (e.g. Price List, Shipping and Billing, Flexfields etc.) should be copied as well.</p> <p>The following attributes should be set in the QuoteLine object before calling this method: <code>quote_line_id</code>, <code>QuoteHeader.quote_header_id</code>, <code>QuoteHeader.last_update_date</code>.</p> <p>This API should be called within a transaction block.</p> <pre>public oracle.apps.qot.core.QuoteLine copy(oracle.apps.aso.quote.CopyQuoteControlRecord copyQuoteControlRecord, oracle.apps.aso.quote.ControlRecord controlRecord) throws FrameworkException, SQLException, QuoteException</pre>

Table 1–29 Methods for Class QuoteLine

Method	Description
<code>delete(ControlRecord)</code>	<p>Deletes this quote line from the quote.</p> <pre>public void delete(oracle.apps.aso.quote.ControlRecord controlRec) throws FrameworkException, SQLException, Quote Exception</pre>
<code>getConfigurationLines()</code>	<p>Returns the immediate child configuration lines of the quote line. The configuration lines are used only for retrieving information from the database. They are populated by the <code>loadLines()</code> and <code>loadAllLines()</code> method in <code>Quote</code>.</p> <pre>public oracle.apps.qot.core.QuoteLine[] getConfigurationLines()</pre>
<code>getDepth()</code>	<p>Returns the node depth of the quote line when all the quote lines in a quote are displayed as a tree structure.</p> <pre>public int getDepth()</pre>
<code>getNodeType</code>	<p>Returns the node type of the quote line when all the quote lines in a quote are displayed as a tree structure. Possible values are:</p> <p><code>QuoteConstant.INTERMEDIATE</code></p> <p><code>QuoteConstant.TERMINAL</code></p> <pre>public int getNodeType()</pre>
<code>getParentLine()</code>	<p>Returns the parent line for the line. If the quote line is a service line, the parent line would reference the serviceable parent line. If the quote line is a configuration component, the parent line would reference the parent configuration line. The parent quote line is used only for retrieving information from the database. It is populated by the <code>loadLines()</code> and <code>loadAllLines()</code> method in <code>Quote</code>.</p> <pre>public oracle.apps.qot.core.QuoteLine getParentLine()</pre>
<code>getQuoteHeader()</code>	<p>Returns quote header for the line.</p> <pre>public oracle.apps.qot.core.QuoteHeader getQuoteHeader()</pre>
<code>getServiceLines()</code>	<p>Returns the child service lines of the line.</p> <pre>public oracle.apps.qot.core.QuoteLine[] getServiceLines()</pre>

Table 1–29 Methods for Class QuoteLine

Method	Description
load(BigDecimal, int)	<p>Loads quote line information from the database based on the quote line ID and load level passed in as parameter.</p> <pre>public static oracle.apps.qot.core.QuoteLine load(java.math.BigDecimal quoteLineId, int level)</pre> <p>throws FrameworkException, SQLException</p>
loadLineDetails()	<p>Loads the line detail information from the database for the quote line and populates lineDetails in the quote line object.</p> <pre>public void loadLineDetails()</pre> <p>throws FrameworkException, SQLException</p>
loadPayments()	<p>Loads the payment information from the database for the quote line and populates payments in the quote line object.</p> <pre>public void loadPayments()</pre> <p>throws FrameworkException, SQLException</p>
loadPriceAdjustments(boolean, int)	<p>Loads price adjustments for the quote line. If the quote header is not set, the quote header will be loaded using the quote header ID in the quote line. The quote header is needed to calculate the adjustment percentage.</p> <pre>public void loadPriceAdjustments(boolean loadHdr, int prcAdjFlags)</pre> <p>throws FrameworkException, SQLException</p>
loadPriceAdjustments(boolean, int, int)	<p>Loads price adjustments for the quote line. If the quote header is not set, the quote header will be loaded using the quote header ID in the quote line. The quote header is needed to calculate the adjustment percentage.</p> <pre>public void loadPriceAdjustments(boolean loadHdr, int prcAdjFlags, int prcAdjTypes)</pre> <p>throws FrameworkException, SQLException</p>
loadPriceAttributes()	<p>Loads the price attribute information from the database for the quote line and populates priceAttrs in the quote line object.</p> <pre>public void loadPriceAttributes()</pre> <p>throws FrameworkException, SQLException</p>

Table 1–29 Methods for Class QuoteLine

Method	Description
loadPromotions()	<p>Loads the promotion information from the database for the quote line and populates priceAttrs in the quote line object.</p> <pre>public void loadPromotions() throws FrameworkException, SQLException</pre>
loadSalesCredits()	<p>Loads the sales credit information from the database for the quote line and populates salesCredits in the quote line object.</p> <pre>public void loadSalesCredits() throws FrameworkException, SQLException</pre>
loadShipments()	<p>Loads the shipment information from the database for the quote line and populates shipments in the quote line object.</p> <pre>public void loadShipments() throws FrameworkException, SQLException</pre>
loadTaxDetails()	<p>Loads the tax detail information from the database for the quote line and populates taxDetails in the quote line object.</p> <pre>public void loadTaxDetails() throws FrameworkException, SQLException</pre>
save(ControlRecord, int)	<p>Saves quote line information to the database. If the caller wants to explicitly pass header last updated date when saving the quote line, the quoteHeader should be set in the quote line. If the quote header is not populated, a quote header will be created using quote_header_id in the QuoteLine object. After the save operation, the quote header will be populated with the new last updated date.</p> <pre>public void save(oracle.apps.aso.quote.ControlRecord controlRec, int level) throws FrameworkException, SQLException, Quote Exception</pre>
setConfigurationLines(QuoteLine[])	<p>Sets the child configuration lines of the line.</p> <pre>public void setConfigurationLines(oracle.apps.qot.core.QuoteLine[] configLines)</pre>
setDepth(int)	<p>Sets the node depth of the quote line when all the quote lines in a quote are displayed as a tree structure.</p> <pre>public void setDepth(int depth)</pre>

Table 1–29 Methods for Class QuoteLine

Method	Description
setNodeType(int)	<p>Sets the node type of the quote line when all the quote lines in a quote are displayed as a tree structure. Possible values are:</p> <p>QuoteConstant.INTERMEDIATE</p> <p>QuoteConstant.TERMINAL</p> <pre>public void setNodeType(int nodeType)</pre>
setParentLine(QuoteLine)	<p>Sets the parent line for the line.</p> <pre>public void setParentLine(oracle.apps.qot.core.QuoteLine parent)</pre>
setQuoteHeader(QuoteHeader)	<p>Sets the quote header for the line.</p> <pre>public void setQuoteHeader(oracle.apps.qot.core.QuoteHeader header)</pre>
setServiceLines(QuoteLine[])	<p>Sets the child service lines of the line.</p> <pre>public void setServiceLines(oracle.apps.qot.core.QuoteLine [] services)</pre>
setShipment(BigDecimal, BigDecimal, BigDecimal, String, Timestamp, String, String)	<p>Sets the appropriate shipment information in the quote line object.</p> <pre>public void setShipment(java.math.BigDecimal shipmentId, java.math.BigDecimal shipToCustAcctId, java.math.BigDecimal shipToContactPartyId, java.math.BigDecimal shipToPartySiteId, java.lang.String shippingMethod, java.sql.Timestamp requestedDeliveryDate, java.lang.String shippingInstructions, java.lang.String packingInstructions)</pre> <p>throws QuoteException</p>

Table 1–29 Methods for Class QuoteLine

Method	Description
<code>split(ControlRecord,String[])</code>	<p>Splits the quote line into multiple lines using the quantities passed in as parameter. The attributes of the quote line will be copied to the new lines created in the database. The following attributes should be set in the quote line before calling this method:</p> <p>quote_line_id inventory_item_id organization_id uom_code quantity quoteHeader.quote_header_id quoteHeader.last_update_date</p> <p>The sum of the quantities passed in as parameter must be less than or equal to the quantity of the quote line being split. If the sum of the quantities passed in as parameter is less than the quantity of the quote line being split, an additional quote line will be created with the difference. This API should be called within a transaction block.</p> <pre>public oracle.apps.got.core.QuoteLine[] split(oracle.apps.aso.quote.ControlRecord controlRecord, java.lang.String[] splitQtyStr) throws FrameworkException, SQLException, Quote Exception</pre>
<code>split(String[])</code>	<p>Splits the quote line into multiple lines using the quantities passed in as parameter. The attributes of the quote line will be copied to the new lines created in the database. The following attributes should be set in the quote line before calling this method: quote_line_id, inventory_item_id, organization_id, uom_code, quantity. The sum of the quantities passed in as parameter must be less than or equal to the quantity of the quote line being split. If the sum of the quantities passed in as parameter is less than the quantity of the quote line being split, an additional quote line will be created with the difference.</p> <pre>public oracle.apps.got.core.QuoteLine[] split(java.lang.String[] splitQtyStr) throws FrameworkException, SQLException, Quote Exception</pre>

Table 1–29 Methods for Class QuoteLine

Method	Description
toString()	Returns a String representation of the quote line. public java.lang.String toString ()
updateService(ControlRecord, BigDecimal, BigDecimal, String, Timestamp, BigDecimal, BigDecimal, String)	Updates the service details for a service line. The following information should be populated in the service quote line before calling this method: quote_line_id, quote_header_id. public void updateService (oracle.apps.aso.quote.ControlRecord controlRecord, java.math.BigDecimal srvcItemId, java.math.BigDecimal invOrgId, java.lang.String uomCode, java.sql.Timestamp startDate, java.math.BigDecimal lineDetailId, java.math.BigDecimal srvcDuration, java.lang.String srvcPeriodCode) throws FrameworkException, SQLException, QuoteException

addServices

```
public void addServices(oracle.apps.aso.quote.ControlRecord controlRec,
java.math.BigDecimal[] srvcItemIds, java.math.BigDecimal[] invOrgIds,
java.lang.String[] uomCodes, java.sql.Timestamp[] startDate,
java.math.BigDecimal[] srvcDuration, java.lang.String[] srvcPeriodCode,
java.math.BigDecimal[] priceListIds)
throws FrameworkException, SQLException, QuoteException
```

Adds services for this quote line. The following attributes should be set in the quote line before calling this API: quote_header_id, quote_line_id, quantity. The quantity for the service lines will be defaulted from the quote line. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

srvcItemIds - Array of inventory item IDs for the service items.

invOrgIds - Array of inventory organization ID for the service items.

uomCodes - Array of unit of measure codes for the service items.

startDate - Array of start dates for the service items.

srvcDuration - Array of service durations for the service items.

srvcPeriodCode - Array of service period codes for the service items.

priceListIds - Array of price list IDs for the service items.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

addToContainer(ControlRecord, InstanceRecord[])

```
public void addToContainer(oracle.apps.aso.quote.ControlRecord controlRec,  
oracle.apps.aso.quote.InstanceRecord[] instanceRec)  
throws SQLException, QuoteException, FrameworkException
```

Adds the Install Base item instances to the model container in the quote. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

instanceRec - Instance records containing instance ids and pricelist IDs of the install base items that need to be added to the container.

Throws:

java.sql.SQLException - If a database error occurs.

QuoteException - If an error occurs while adding IB items to container.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

copy(CopyQuoteControlRecord, ControlRecord)

```
public oracle.apps.qot.core.QuoteLine  
copy(oracle.apps.aso.quote.CopyQuoteControlRecord copyQuoteControlRecord,  
oracle.apps.aso.quote.ControlRecord controlRecord)  
throws FrameworkException, SQLException, QuoteException
```

Copies the quote line into a new quote line within the same quote with the a CopyQuoteControlRecord object passed in as parameter.

If the quote line is a configured model line, the entire configuration is copied together with the model line. The configuration structure is preserved in the newly created model.

If services are associated to the quote line or to one or more component items in the configuration (if the quote line is a configured model), then services are copied as well. The newly created service items are associated to the corresponding serviceable items.

Other line-level attributes (e.g. Price List, Shipping and Billing, Flexfields etc.) should be copied as well.

The following attributes should be set in the QuoteLine object before calling this method: quote_line_id, QuoteHeader.quote_header_id, QuoteHeader.last_update_date.

This API should be called within a transaction block.

Parameters:

copyQuoteControlRecord - The control record indicating which attributes should be copied in the quote line.

controlRecord - The standard control record for updating a quote.

Returns: The QuoteLine object with quote line ID populated for the new quote line created.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

delete(ControlRecord)

```
public void delete(oracle.apps.aso.quote.ControlRecord controlRec)  
throws FrameworkException, SQLException, QuoteException
```

Deletes this quote line from the quote. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - if an application error occurs.

getConfigurationLines()

```
public oracle.apps.qot.core.QuoteLine[] getConfigurationLines()
```

Returns the immediate child configuration lines of the quote line. The configuration lines are used only for retrieving information from the database. They are populated by the loadLines() and loadAllLines() method in Quote.

Returns: An array of immediate child configuration lines.

getDepth()

```
public int getDepth()
```

Returns the node depth of the quote line when all the quote lines in a quote are displayed as a tree structure.

Returns: The node depth.

getNodeType()

```
public int getNodeTypes()
```

Returns the node type of the quote line when all the quote lines in a quote are displayed as a tree structure. Possible values are:

- QuoteConstant.INTERMEDIATE
- QuoteConstant.TERMINAL

Returns: The node type.

getParentLine()

```
public oracle.apps.qot.core.QuoteLine getParentLine()
```

Returns the parent line for the line. If the quote line is a service line, the parent line would reference the serviceable parent line. If the quote line is a configuration component, the parent line would reference the parent configuration line. The parent quote line is used only for retrieving information from the database. It is populated by the loadLines() and loadAllLines() method in Quote.

Returns: The parent of the line.

getQuoteHeader()

```
public oracle.apps.qot.core.QuoteHeader getQuoteHeader()
```

Returns quote header for the line.

Returns: quote header for the line.

getServiceLines()

```
public oracle.apps.qot.core.QuoteLine[] getServiceLines()
```

Returns the child service lines of the line.

Returns: The child service lines of the line.

load(BigDecimal, int)

```
public static oracle.apps.qot.core.QuoteLine load(java.math.BigDecimal  
quoteLineId, int level)
```

throws FrameworkException, SQLException

Loads quote line information from the database based on the quote line ID and load level passed in as parameter.

Parameters:

quoteLineId - quote line ID

level - load level which determines the quote line information which will be retrieved. Possible values:

- QuoteLine.LEVEL10
- QuoteLine.LEVEL20
- QuoteLine.LEVEL90
- QuoteLine.LEVEL100

Returns: QuoteLine object with the appropriate fields populated.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

loadLineDetails()

```
public void loadLineDetails()  
throws FrameworkException, SQLException
```

Loads the line detail information from the database for the quote line and populates lineDetails in the quote line object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadPayments()

```
public void loadPayments()  
throws FrameworkException, SQLException
```

Loads the payment information from the database for the quote line and populates payments in the quote line object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadPriceAdjustments(boolean, int)

```
public void loadPriceAdjustments(boolean loadHdr, int prcAdjFlags)
throws FrameworkException, SQLException
```

Loads price adjustments for the quote line. If the quote header is not set, the quote header will be loaded using the quote header ID in the quote line. The quote header is needed to calculate the adjustment percentage.

Parameters:

loadHdr - Whether to load header level price adjustments and apply them to the quote line.

prcAdjFlags - Indicates which price adjustments to load. Input should be created using the bitwise OR of the following values:

- QuoteConstant.PRC_ADJ_ALL - All price adjustments will be loaded.
- QuoteConstant.PRC_ADJ_APPLIED - Only loads price adjustments that are applied.
- QuoteConstant.PRC_ADJ_NON_AUTOMATIC - Only loads price adjustments that are not automatic.
- QuoteConstant.PRC_ADJ_UPDATE_ALLOWED - Only loads price adjustments that are allowed to be updated.

If `prcAdjFlags = (QuoteConstant.PRC_ADJ_APPLIED | QuoteConstant.PRC_ADJ_NON_AUTOMATIC)`, only loads price adjustments in which `applied_flag = Y` and `automatic_flag = N`.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

loadPriceAdjustments(boolean, int, int)

```
public void loadPriceAdjustments(boolean loadHdr, int prcAdjFlags,
int prcAdjTypes)
throws FrameworkException, SQLException
```

Loads price adjustments for the quote line. If the quote header is not set, the quote header will be loaded using the quote header ID in the quote line. The quote header is needed to calculate the adjustment percentage.

Parameters:

loadHdr - Whether to load header level price adjustments and apply them to the quote line.

prcAdjFlags - Indicates which price adjustments to load. Input should be created using the bitwise OR of the following values:

- QuoteConstant.PRC_ADJ_ALL - All price adjustments will be loaded.
- QuoteConstant.PRC_ADJ_APPLIED - Only loads price adjustments that are applied.
- QuoteConstant.PRC_ADJ_NON_AUTOMATIC - Only loads price adjustments that are not automatic.
- QuoteConstant.PRC_ADJ_UPDATE_ALLOWED - Only loads price adjustments that are allowed to be updated.

If `prcAdjFlags = (QuoteConstant.PRC_ADJ_APPLIED | QuoteConstant.PRC_ADJ_NON_AUTOMATIC)`, only loads price adjustments in which `applied_flag = Y` and `automatic_flag = N`.

`prcAdjType` - Indicates the type of price adjustments which should be loaded. The parameter should be constructed using bitwise OR of the following possible values:

- QuoteConstant.PRC_ADJ_ALL - Indicates all price adjustments.
- QuoteConstant.PRC_ADJ_FREIGHT - Indicates freight/special charge price adjustments.
- QuoteConstant.PRC_ADJ_DIS - Indicates discount price adjustments.
- QuoteConstant.PRC_ADJ_OID - Indicates other item discount price adjustments.
- QuoteConstant.PRC_ADJ_PBH - Indicates price break header price adjustments.
- QuoteConstant.PRC_ADJ_SUR - Indicates surcharge price adjustments. For example, `prcAdjType = QuoteConstant.PRC_ADJ_FREIGHT | QuoteConstant.PRC_ADJ_DIS` indicates that price adjustments of type freight or type discount will be loaded.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

loadPriceAttributes()

```
public void loadPriceAttributes()
throws FrameworkException, SQLException
```

Loads the price attribute information from the database for the quote line and populates `priceAttrs` in the quote line object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadPromotions()

```
public void loadPromotions()
```

```
throws FrameworkException, SQLException
```

Loads the promotion information from the database for the quote line and populates priceAttrs in the quote line object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadSalesCredits()

```
public void loadSalesCredits()
```

```
throws FrameworkException, SQLException
```

Loads the sales credit information from the database for the quote line and populates salesCredits in the quote line object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadShipments()

```
public void loadShipments()
```

```
throws FrameworkException, SQLException
```

Loads the shipment information from the database for the quote line and populates shipments in the quote line object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadTaxDetails()

```
public void loadTaxDetails()
```

```
throws FrameworkException, SQLException
```

Loads the tax detail information from the database for the quote line and populates taxDetails in the quote line object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

save(ControlRecord, int)

```
public void save(oracle.apps.aso.quote.ControlRecord controlRec, int level)  
throws FrameworkException, SQLException, QuoteException
```

Saves quote line information to the database. If the caller wants to explicitly pass header last updated date when saving the quote line, the quoteHeader should be set in the quote line. If the quote header is not populated, a quote header will be created using quote_header_id in the QuoteLine object. After the save operation, the quote header will be populated with the new last updated date. This API should be called within a transaction block.

Parameters:

controlRec - Standard control record for updating a quote.

level - Level indicating the quote line information which should be saved. This parameter should be constructed using bitwise OR of the following possible values:

- QuoteConstant.SAVE_HEADER - Indicates that basic quote header information should be saved.
- QuoteConstnat.SAVE_LINE - Indicates that basic quote line information should be saved.
- QuoteConstant.SAVE_LINE_DETAIL - Indicates that line detail information should be saved.
- QuoteConstant.SAVE_PRC_ADJ - Indicates that price adjustment information populated in the quote line object should be saved.
- QuoteConstant.SAVE_PRC_ATTR - Indicates that price attribute information populated in the quote line object should be saved.
- QuoteConstant.SAVE_TAX - Indicates that tax information populated in the quote line object should be saved.
- QuoteConstant.SAVE_PAYMENT - Indicates that payment information populated in the quote line object should be saved.
- QuoteConstant.SAVE_SHIPMENT - Indicates that shipment information populated in the quote line object should be saved.
- QuoteConstant.SAVE_SALES_CREDIT - Indicates that sales credit information populated in the quote line object should be saved. For example, level = QuoteConstant.SAVE_SHIPMENT | QuoteConstant.SAVE_PAYMENT indicates that shipment and payment information should be saved along with quote line basic information.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

setConfigurationLines(QuoteLine[])

public void **setConfigurationLines**(oracle.apps.qot.core.QuoteLine[] configLines)

Sets the child configuration lines of the line.

Parameters: configLines - An array of immediate child configuration lines.

setDepth(int)

public void **setDepth**(int depth)

Sets the node depth of the quote line when all the quote lines in a quote are displayed as a tree structure.

Parameters: depth - The node depth.

setNodeType(int)

public void **setNodeType**(int nodeType)

Sets the node type of the quote line when all the quote lines in a quote are displayed as a tree structure. Possible values are:

- QuoteConstant.INTERMEDIATE
- QuoteConstant.TERMINAL

Parameters: nodeType - The node type.

setParentLine(QuoteLine)

public void **setParentLine**(oracle.apps.qot.core.QuoteLine parent)

Sets the parent line for the line.

Parameters: parent - The parent line for the line.

setQuoteHeader(QuoteHeader)

public void **setQuoteHeader**(oracle.apps.qot.core.QuoteHeader header)

Sets the quote header for the line.

Parameters: header - The quote header for the line

setServiceLines(QuoteLine[])

public void **setServiceLines**(oracle.apps.qot.core.QuoteLine[] services)

Sets the child service lines of the line.

Parameters: services - The child service lines of the line.

setShipment(BigDecimal, BigDecimal, BigDecimal, BigDecimal, String, Timestamp, String, String)

```
public void setShipment(java.math.BigDecimal shipmentId,  
java.math.BigDecimal shipToCustAcctId,  
java.math.BigDecimal shipToContactPartyId,  
java.math.BigDecimal shipToPartySiteId, java.lang.String shippingMethod,  
java.sql.Timestamp requestedDeliveryDate, java.lang.String shippingInstructions,  
java.lang.String packingInstructions)  
throws QuoteException
```

Sets the appropriate shipment information in the quote line object.

Parameters:

shipmentId - Shipment ID, the primary key in ASO_SHIPMENTS, if there is an existing shipment for the quote header.

shipToCustAcctId - Ship To customer account ID.

shipToContactPartyId - Ship To contact party ID corresponding to the party ID of the contact relationship (a row in HZ_PARTIES with party_type PARTY_RELATIONSHIP) if there is a ship-to contact.

shipToPartySiteId - Ship To party site ID for the Ship To address.

shippingMethod - Shipping method.

requestedDeliveryDate - Requested delivery date.

shippingInstructions - Shipping instructions.

packingInstructions - Packing instructions.

Throws:

QuoteException - If an application error occurs.

split(ControlRecord, String[])

```
public oracle.apps.qot.core.QuoteLine[]  
split(oracle.apps.aso.quote.ControlRecord controlRecord,  
java.lang.String[] splitQtyStr)  
throws FrameworkException, SQLException, QuoteException
```

Splits the quote line into multiple lines using the quantities passed in as parameter. The attributes of the quote line will be copied to the new lines created in the database. The following attributes should be set in the quote line before calling this method:

- quote_line_id
- inventory_item_id
- organization_id
- uom_code
- quantity
- quoteHeader.quote_header_id
- quoteHeader.last_update_date

The sum of the quantities passed in as parameter must be less than or equal to the quantity of the quote line being split. If the sum of the quantities passed in as parameter is less than the quantity of the quote line being split, an additional quote line will be created with the difference. This API should be called within a transaction block.

Parameters:

controlRecord - The standard control record for updating a quote.

splitQtyStr - An array of String objects containing the new quantities.

Returns: An array of QuoteLine objects for newly split quote lines with quote_line_id populated for each quote line.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

split(String[])

public oracle.apps.got.core.QuoteLine[] **split**(java.lang.String[] splitQtyStr)
throws FrameworkException, SQLException, QuoteException

Splits the quote line into multiple lines using the quantities passed in as parameter and reprises the quote. The attributes of the quote line will be copied to the new lines created in the database. The following attributes should be set in the quote line before calling this method:

- quote_line_id
- inventory_item_id
- organization_id
- uom_code

- quantity

The sum of the quantities passed in as parameter must be less than or equal to the quantity of the quote line being split. If the sum of the quantities passed in as parameter is less than the quantity of the quote line being split, an additional quote line will be created with the difference. This API should be called within a transaction block.

Parameters: splitQtyStr - An array of String objects containing the new quantities

Returns: An array of QuoteLine objects for newly split quote lines with quote_line_id populated for each quote line.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the quote line.

Overrides: toString in class LineRecord

Returns: String representation of the quote line.

updateService(ControlRecord, BigDecimal, BigDecimal, String, Timestamp, BigDecimal, BigDecimal, String)

```
public void updateService(oracle.apps.aso.quote.ControlRecord controlRecord,  
java.math.BigDecimal srvcItemId, java.math.BigDecimal invOrgId,  
java.lang.String uomCode, java.sql.Timestamp startDate,  
java.math.BigDecimal lineDetailId, java.math.BigDecimal srvcDuration,  
java.lang.String srvcPeriodCode)  
throws FrameworkException, SQLException, QuoteException
```

Updates the service details for a service line. The following information should be populated in the service quote line before calling this method: quote_line_id, quote_header_id. This API should be called within a transaction block.

Parameters:

controlRecord - Standard control record for updating a quote.

srvcItemId - Inventory item ID of the service item.

invOrgId - Inventory organization ID of the service item.

uomCode - Unit of measure code of the service item.

startDate - Start date for the service line.

lineDetailId - Line detail ID, primary key in ASO_QUOTE_LINE_DETATILS, for the service line.

srvcDuration - Service duration for the service line.

srvcPeriodCode - Service period code for the service line.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If expected application error occurs.

1.20 Class QuoteSecurityMgr

```
java.lang.Object
|
+--oracle.apps.qot.core.QuoteSecurityMgr
```

public class QuoteSecurityMgr

The QuoteSecurityMgr contains the methods used to implement quote security. It provides the methods to retrieve quote security related information, such as quote access level, quote sales team, and account sales team information. It provides the methods to update the quote sales team.

1.20.1 Fields for Class QuoteSecurityManager

NONE

```
public static final java.lang.String NONE
```

Indicates no access to a quote.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

READ

```
public static final java.lang.String READ
```

Indicates read-only access to a quote.

UPDATE

```
public static final java.lang.String UPDATE
```

Indicates update access to a quote.

1.20.2 Constructors for Class QuoteSecurityManager

QuoteSecurityMgr()

```
public QuoteSecurityMgr()
```

Default constructor.

1.20.3 Methods for Class QuoteSecurityManager

The following table is an index of the Class QuoteSecurityManager methods:

Table 1–30 Methods for Class QuoteSecurityManager

Method	Description
deleteResource(QuoteAccessRecord[])	<p>Deletes resources from the quote sales team. This API should be called with in a transaction block.</p> <pre>public static void deleteResource(oracle.apps.aso.quote.QuoteAccessRecord[] quoteAccessRecord) throws FrameworkException, SQLException, QuoteException</pre>
getAccountAccess(BigDecimal, BigDecimal)	<p>Returns the account sales team access level for the given resource ID and customer partyId.</p> <pre>public static java.lang.String getAccountAccess(java.math.BigDecimal resourceId, java.math.BigDecimal partyId) throws FrameworkException, SQLException</pre>
getQuoteAccess(BigDecimal, BigDecimal)	<p>Returns the Quote Access Level information for the given quote number and resource ID.</p> <pre>public static java.lang.String getQuoteAccess(java.math.BigDecimal resourceId, java.math.BigDecimal quoteNumber) throws FrameworkException, SQLException, QuoteException</pre>
getQuoteAccessRecord(BigDecimal)	<p>Returns the quote sales team information for the specified Quote Number.</p> <pre>public static oracle.apps.aso.quote.QuoteAccessRecord[] getQuoteAccessRecord(java.math.BigDecimal quoteNumber) throws FrameworkException, SQLException</pre>

Table 1–30 Methods for Class QuoteSecurityManager

Method	Description
updateResource(QuoteAccessRecord[])	<p>Adds or updates resources on the quote sales team for the quote access records passed in as parameter. This API should be called with in a transaction block.</p> <pre>public static void updateResource(oracle.apps.aso.quote.QuoteAccessRecord[] quoteAccessRecord) throws FrameworkException, SQLException, QuoteException</pre>

deleteResource(QuoteAccessRecord[])

```
public static void deleteResource(oracle.apps.aso.quote.QuoteAccessRecord[]
quoteAccessRecord)
throws FrameworkException, SQLException, QuoteException
```

Deletes resources from the quote sales team. This API should be called with in a transaction block.

Parameters: quoteAccessRecord - Array of quote access records with access_id populated for the rows to be deleted from the quote sales team.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

getAccountAccess(BigDecimal, BigDecimal)

```
public static java.lang.String getAccountAccess(java.math.BigDecimal
resourceId, java.math.BigDecimal partyId)
throws FrameworkException, SQLException
```

Returns the account sales team access level for the given resource ID and customer partyId

Parameters:

resourceId - Resource ID.

partyId - Customer party ID.

Returns:

The account sales team access level for the given resource ID and customer party ID. Possible values:

- **Y** - The resource is allowed to create a quote for the customer.

- N - The resource is not allowed to create a quote for the customer.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

getQuoteAccess(BigDecimal, BigDecimal)

```
public static java.lang.String getQuoteAccess(java.math.BigDecimal  
resourceId, java.math.BigDecimal quoteNumber)
```

```
throws FrameworkException, SQLException, QuoteException
```

Returns the Quote Access Level information for the given quote number and resource ID.

Parameters:

resourceId - Resource ID.

quoteNumber - Quote Number.

Returns:

The quote access level for the specified quote number and resource ID. Possible values:

- UPDATE - The resource has update access to the quote.
- READ - The resource has read-only access to the quote.
- NONE - The resource has no access to the quote.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

getQuoteAccessRecord(BigDecimal)

```
public static oracle.apps.aso.quote.QuoteAccessRecord[]
```

```
getQuoteAccessRecord(java.math.BigDecimal quoteNumber)
```

```
throws FrameworkException, SQLException
```

Returns the quote sales team information for the specified Quote Number.

Parameters: quoteNumber - Quote number.

Returns: Array of QuoteAccessRecord containing the quote sales team information.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

updateResource(QuoteAccessRecord[])

```
public static void updateResource(oracle.apps.aso.quote.QuoteAccessRecord[]
quoteAccessRecord)
```

throws FrameworkException, SQLException, QuoteException

Adds or updates resources on the quote sales team for the quote access records passed in as parameter. This API should be called with in a transaction block.

Parameters: quoteAccessRecord - Quote access records containing the quote access information to be updated or added. To update details of an existing record on the quote sales team, populate access_id in addition to the information to be updated. To add a new resource to a quote sales team, populate the access information to be added.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

QuoteException - If an application error occurs.

1.21 Class QuoteWarningException

```
java.lang.Object
|
+--java.lang.Throwable
|
+--java.lang.Exception
|
+--oracle.apps.jtf.base.resources.FrameworkException
|
+--oracle.apps.qot.core.QuoteException
|
+--oracle.apps.qot.core.QuoteWarningException
```

```
public class QuoteWarningException extends QuoteException
```

All Implemented Interfaces: java.io.Serializable

QuoteWarningException is thrown when a warning occurs in a method in package oracle.apps.qot.core or oracle.app.qot.core.util.

1.21.1 Fields for Class QuoteWarningException

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.21.2 Constructors for Class QuoteWarningException

QuoteWarningException(Exception, String)

```
public QuoteWarningException(java.lang.Exception e, java.lang.String errorKey)
```

Constructs an exception with the given exception and the error key.

Parameters:

e - The parent exception.

errorKey - The error key.

QuoteWarningException(Exception, String, Hashtable)

```
public QuoteWarningException(java.lang.Exception e, java.lang.String errorKey,  
java.util.Hashtable params)
```

Constructs an exception with the given exception, the error key, and parameters.

Parameters:

e - The parent exception.

errorKey - The error key.

params - A hash table of tokens for the error key.

QuoteWarningException(Exception, String, Object[])

```
public QuoteWarningException(java.lang.Exception e, java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs an exception with the given exception, the error key, and parameters.

Parameters:

e - The parent exception.

errorKey - The error key.

params - An array of tokens for the error key.

QuoteWarningException(Exception, String, String)

```
public QuoteWarningException(java.lang.Exception e, java.lang.String errorKey,  
java.lang.String param)
```

Constructs an exception with the given exception, the error key, and the parameter.

Parameters:

e - The parent exception.

errorKey - The error key.

param - The token for the error key.

QuoteWarningException(int, String)

```
public QuoteWarningException(int err_msg_count, java.lang.String errorKey)
```

```
throws FrameworkException
```

Constructs an exception with the message count and error key. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the PL/SQL error stack.

errorKey - The error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteWarningException(int, String, Object[])

```
public QuoteWarningException(int err_msg_count, java.lang.String errorKey,  
java.lang.Object[] params)
```

```
throws FrameworkException
```

Constructs an exception with the message count, error key, and parameter tokens. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the PL/SQL error stack.

errorKey - The error key.

params - An array of tokens for the errorKey

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QuoteWarningException(int, String, String)

```
public QuoteWarningException(int err_msg_count, java.lang.String errorKey,  
java.lang.String param)  
throws FrameworkException
```

Constructs an exception with the message count, error key, and parameter token. Errors at the PL/SQL level will be retrieved.

Parameters:

`err_msg_count` - The number of messages to be returned from the PL/SQL error stack.

`errorKey` - The error key.

`param` - A token for the error key.

Throws: `oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

QuoteWarningException(String)

```
public QuoteWarningException(java.lang.String errorKey)
```

Constructs an exception with the error key.

Parameters: `errorKey` - The error key.

QuoteWarningException(String, Object[])

```
public QuoteWarningException(java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs an exception with the error key and parameters.

Parameters:

`errorKey` - The error key.

`params` - An array of tokens for the error key.

QuoteWarningException(String, String)

```
public QuoteWarningException(java.lang.String err_msg,  
java.lang.String errorKey)
```

Constructs an exception with the error message and error key.

Parameters:

`err_msg` - The error message.

`errorKey` - The error key.

QuoteWarningException(String, String, Object[])

```
public QuoteWarningException(java.lang.String err_msg,
    java.lang.String errorKey, java.lang.Object[] params)
```

Constructs an exception with the error message, the error key, and parameters.

Parameters:

err_msg - The error message.

errorKey - The error key.

params - An array of tokens for the error key.

QuoteWarningException(String, String, String)

```
public QuoteWarningException(java.lang.String err_msg,
    java.lang.String errorKey, java.lang.String param)
```

Constructs an exception with the error message, the error key, and the parameter.

Parameters:

err_msg - The error message.

errorKey - The error key.

param - The parameter.

1.22 Class Service

```
java.lang.Object
|
+--oracle.apps.qot.core.Service
```

public class Service

The Service object contains the basic information for a service, such as inventory item part number, inventory item description, service start date, service duration, service period code, whether the service is an included warranty. The Service object provides a method to list the services for a quote line, a pending order line, or and install base instance.

1.22.1 Fields for Class Service

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

1.22.2 Constructors for Class Service

Service()

```
public Service()
```

Default constructor.

Service(String, String, Timestamp, BigDecimal, String, String)

```
public Service(java.lang.String _itemNumber, java.lang.String _itemDescription, java.sql.Timestamp _startDate, java.math.BigDecimal _duration, java.lang.String _periodCode, java.lang.String _warranty) throws FrameworkException, SQLException
```

Constructor.

Parameters:

`_itemNumber` - Inventory item part number.

`_itemDescription` - Inventory item description.

`_startDate` - Service start date.

`_duration` - Service duration.

`_periodCode` - Service period code.

`_warranty` - Flag indicating whether the service is an included warranty “Y” if the service is an included warranty.

Throws:

`java.sql.SQLException` - If a database error occurs.

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

1.22.3 Methods for Class Service

The following table is an index of the Class Service methods:

Table 1–31 *Methods for Class Service*

Method	Description
<code>getDuration()</code>	Returns the duration from the Service object. <pre>public java.math.BigDecimal getDuration()</pre>

Table 1–31 Methods for Class Service

Method	Description
<code>getIsIncluded()</code>	Returns the <code>isIncluded</code> flag from the Service Object. <code>public boolean getIsIncluded()</code>
<code>getItemDescription()</code>	Returns the Inventory Item Description for the Service Object. <code>public java.lang.String getItemDescription()</code>
<code>getItemNumber()</code>	Returns the Inventory Item Part Number for the Service Object. <code>public java.lang.String getItemNumber()</code>
<code>getPeriod()</code>	Returns the translated meaning of the service period from the Service object. <code>public java.lang.String getPeriod()</code>
<code>getPeriodCode()</code>	Returns the period code from the Service object. <code>public java.lang.String getPeriodCode()</code>
<code>getStartDate()</code>	Returns the start Date for the Service object. <code>public java.sql.Timestamp getStartDate()</code>
<code>listServices(int, BigDecimal)</code>	Lists the selected services and included warranties for the source and <code>sourceId</code> passed in as parameter. If source is <code>INSTALL_BASE</code> , services will be selected from installed base. If source is <code>PENDING_ORDER</code> , services will be selected from the order. If source is <code>CURRENT_QUOTE</code> , services will be selected from the quote. <code>public static oracle.apps.qot.core.Service[] listServices(int source, java.math.BigDecimal sourceId)</code> throws <code>QuoteException</code> , <code>FrameworkException</code> , <code>SQLException</code>
<code>setDuration(BigDecimal)</code>	Sets the duration for the service object. <code>public void setDuration(java.math.BigDecimal _duration)</code>
<code>setIsIncluded(boolean)</code>	Sets whether the service is an included warranty. <code>public void setIsIncluded(boolean _isIncluded)</code>
<code>setItemDescription(String)</code>	Sets the inventory item description. <code>public void setItemDescription(java.lang.String _itemDescription)</code>

Table 1–31 Methods for Class Service

Method	Description
<code>setItemNumber(String)</code>	Sets the item number. <code>public void setItemNumber(java.lang.String _ itemNumber)</code>
<code>setPeriod(String)</code>	Sets the translated meaning of the service period for the Service object. <code>public void setPeriod(java.lang.String _ period)</code>
<code>setPeriodCode(String)</code>	Sets the period code for the Service object. <code>public void setPeriodCode(java.lang.String _ periodCode)</code>
<code>setStartDate(Timestamp)</code>	Sets the Start Date for the Service object. <code>public void setStartDate(java.sql.Timestamp _ startDate)</code>
<code>toString()</code>	Returns a String representation of the Service object. <code>public java.lang.String toString()</code>

getDuration()

```
public java.math.BigDecimal getDuration()
```

Returns the duration from the service object.

Returns: Service duration.

getIsIncluded()

```
public boolean getIsIncluded()
```

Returns the `isIncluded` flag from the service object.

Returns: whether the service is an included warranty.

getItemDescription()

```
public java.lang.String getItemDescription()
```

Returns the inventory item description for the service object.

Returns: Inventory item description.

getItemNumber()

```
public java.lang.String getItemNumber()
```

Returns the inventory item part number for the service object.

Returns: Inventory item part number for the service object.

getPeriod()

```
public java.lang.String getPeriod()
```

Returns the translated meaning of the service period from the service object.

Returns: Translated meaning of the service period.

getPeriodCode()

```
public java.lang.String getPeriodCode()
```

Returns the period code from the service object.

Returns: PeriodCode.

getStartDate()

```
public java.sql.Timestamp getStartDate()
```

Returns the start date for the service object.

Returns: Service start date.

listServices(int, BigDecimal)

```
public static oracle.apps.qot.core.Service[] listServices(int source,  
java.math.BigDecimal sourceId)
```

```
throws QuoteException, FrameworkException, SQLException
```

Lists the selected services and included warranties for the source and sourceId passed in as parameter. If source is `INSTALL_BASE`, services will be selected from installed base. If source is `PENDING_ORDER`, services will be selected from the order. If source is `CURRENT_QUOTE`, services will be selected from the quote

Parameters:

source - Source for retrieving the services. Possible values:

- `QuoteConstant.CURRENT_QUOTE`
- `QuoteConstant.INSTALL_BASE`
- `QuoteConstant.PENDING_ORDER`

sourceId - If source is `CURRENT_QUOTE`, quote line ID. If source is `INSTALL_BASE`, instance ID. If source is `PENDING_ORDER`, order line ID.

Returns: Array of Service Objects for the services which have already been selected

Throws:

`QuoteException` - If an application error occurs.

`java.sql.SQLException` - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setDuration(BigDecimal)

```
public void setDuration(java.math.BigDecimal _duration)
```

Sets the duration for the service object.

Parameters: _duration - Service duration.

setIsIncluded(boolean)

```
public void setIsIncluded(boolean _isIncluded)
```

Sets whether the service is an included warranty

Parameters: _isIncluded - Whether the service is an included warranty.

setItemDescription(String)

```
public void setItemDescription(java.lang.String _itemDescription)
```

Sets the inventory item description.

Parameters: _itemDescription - Inventory item description.

setItemNumber(String)

```
public void setItemNumber(java.lang.String _itemNumber)
```

Sets the item number.

Parameters: _itemNumber - Inventory item part number.

setPeriod(String)

```
public void setPeriod(java.lang.String _period)
```

Sets the translated meaning of the service period for the service object.

Returns: Translated meaning of the service period.

setPeriodCode(String)

```
public void setPeriodCode(java.lang.String _periodCode)
```

Sets the period code for the service object.

Parameters: _periodCode - Service period code.

setStartDate(Timestamp)

```
public void setStartDate(java.sql.Timestamp _startDate)
```

Sets the start date for the service object.

Parameters: _startDate - Service start date.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the service object

Overrides: toString in class Object

Returns: A String representation of the service object.

oracle.apps.qot.core.util

This section lists the Oracle Quoting Java APIs in the package oracle.apps.qot.core.util.

2.1 Package oracle.apps.qot.core.util

The following table lists each class in this package:

Table 2–1 Class Summary for oracle.apps.qot.core.util

Class	Description
Class CCPayment	CCPayment handles Credit Cards as a payment instrument.
Class ConcurrentRequest	The ConcurrentRequest object contains the information of a concurrent request, as well as the convenient methods to retrieve the information of a concurrent request.
Class Currency	The Currency object contains the following information for a currency: currency code, currency name, currency symbol, and available price list IDs for the currency.
Class InstallBaseRelType	The InstallBaseRelType object contains the following information for an install base instance relationship type: relationship type code, and relationship direction. It provides methods for loading a specific install base instance relationship type.
Class MarketingSource	The MarketingSource object represents a marketing campaign which can be applied to a quote.
Class Opportunity	Opportunity provides the methods to retrieve a specified opportunity and retrieve the list of open opportunities.

Table 2–1 Class Summary for oracle.apps.qot.core.util

Class	Description
Class PaymentTerm	The PaymentTerm object contains information for payment terms. This class provides methods to retrieve a specified payment term, retrieve a list of payment terms, and determine the default payment term.
Class PriceAgreement	The PriceAgreement object contains the following information for a price agreement: agreement ID, agreement name, price list ID. The class provides methods to retrieve a specific price agreement and retrieve the list of available price agreements for a customer.
Class PriceList	The PriceList object contains the following information for a price list: header ID, name, currency code.
Class PriceModifier	The PriceModifier object contains the following information for a price modifier: header ID, code, and name.
Class QuoteStatus	The QuoteStatus object contains the following information for a quote status: quote status ID, status code, status meaning, whether update is allowed.
Class Resource	The Resource object contains the following information for a resource: Resource ID, sales rep ID, sales credit type ID, name, resource category, job title, email, phone, city, state, postal code, and country.
Class ResourceGroup	The ResourceGroup object contains the following information for a resource group: group ID, group name.
Class SalesCreditType	The SalesCreditType object contains the following information for a sales credit type: sales credit type ID, name, quota flag.
Class Territory	The Territory object contains the following information for a territory: territory code, name, address style.
Class TransactionType	The TransactionType object contains the following information for a transaction type: transaction type ID, transaction type code, name, and price list ID.

2.2 Class CCPayment

```

java.lang.Object
|
+--oracle.apps.qot.core.util.CCPayment

```

public class CCPayment

CCPayment handles Credit Cards as a payment instrument. It provides methods to validate credit card information, mask a portion of the credit card number, and remove filler characters from a credit card number.

Table 2–2 Inherited Member Summary**Methods inherited from class Object**

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

2.2.1 Fields for Class CCPayment**credit_card_holder_name**

```
public java.lang.String credit_card_holder_name
```

Name of the holder of the credit card.

credit_card_id

```
public java.math.BigDecimal credit_card_id
```

Credit card ID. Foreign key to AP_BANK_ACCOUNTS_ALL.

credit_card_num

```
public java.lang.String credit_card_num
```

Credit card number.

credit_card_type_code

```
public java.lang.String credit_card_type_code
```

Type of the credit card (like Visa, MC, etc.).

currency_code

```
public java.lang.String currency_code
```

Currency code.

DEF_FILLER_CHARS

```
public static char[] DEF_FILLER_CHARS
```

Default filler characters in a credit card #- i.e. characters which are not part of the number itself, but can be safely ignored.

exp_date

```
public java.sql.Timestamp exp_date
```

Credit card expiration date.

filler_chars

```
public char[] filler_chars
```

Filler characters that can safely be ignored.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.2.2 Constructors for Class CCPayment

CCPayment()

```
public CCPayment()
```

Constructor. Sets filler_chars to the default filler characters.

2.2.3 Methods for Class CCPayment

The following table is an index of the Class CCPayment methods:

Table 2–3 *Methods for Class CCPayment*

Method	Description
maskCCNum(String, int)	Mask out the first n digits of a credit card number. <pre>public static java.lang.String maskCCNum(java.lang.String cc_number, int num_show)</pre>

Table 2–3 Methods for Class CCPayment

<code>stripNumber(String, char[])</code>	<p>Removes all non digit characters out of the credit card string, skipping “acceptable” characters. Essentially cleans up and checks for unacceptable characters at the same time.</p> <pre>public static java.lang.String stripNumber(java.lang.String cc_num, char[] fill_chars)</pre>
<code>toString()</code>	<p>Returns a String representation of CCPayment object.</p> <pre>public java.lang.String toString()</pre>
<code>validateCreditCard()</code>	<p>Validates a Credit Card.</p> <pre>public void validateCreditCard()</pre> <p>throws QuoteException, SQLException, FrameworkException</p>

maskCCNum(String, int)

```
public static java.lang.String maskCCNum(java.lang.String cc_number,
int num_show)
```

Masks out the 1st n digits of a credit card number.

Parameters:

`cc_number` - The credit card number.

`num_show` - How many of the “last” digits to show.

Returns: A masked credit card number.

stripNumber(String, char[])

```
public static java.lang.String stripNumber(java.lang.String cc_num,
char[] fill_chars)
```

Removes all non digit characters out of the credit card string, skipping “acceptable” characters. Essentially cleans up and checks for unacceptable characters at the same time.

Parameters:

`cc_num` - A credit card number, possibly filled with non-digit filler characters.

`fill_chars` - An array of non-digit characters to be ignored. If this parameter is null then all non-digits are ignored, allowing for maximum leniency.

Returns: A credit card number composed entirely of digits, or null if it contains invalid characters.

toString()

```
public java.lang.String toString()
```

Returns a String representation of CCPayment object.

Overrides: toString in class Object

Returns: A String representation of CCPayment object.

validateCreditCard()

```
public void validateCreditCard()
```

throws `QuoteException`, `SQLException`, `FrameworkException`

Validates a credit card.

Performs the following validations:

1. Checks for valid characters (allowing for filler characters).
2. That the type matches the number.
3. That the expiration date is beyond the database sysdate.
4. Verify the digit checksum. Before calling this API, the following instance variables should be set in the CCPayment object:
 - * `cc_num`: Credit card number.
 - * `cc_type_code`: Code of the type of credit card (ex/ Visa, MC, etc).
 - * `exp_date`: Expiration date of the credit card.
 - * `filler_chars`: Array of characters to safely ignore in the validation. If a value is not set, uses default filler chars.

Throws:

`java.sql.SQLException` - If a database error occurs.

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`oracle.apps.qot.core.QuoteException` - If the credit card information is invalid.

2.3 Class ConcurrentRequest

```
java.lang.Object
|
+--oracle.apps.qot.core.util.ConcurrentRequest
```

public class **ConcurrentRequest**

The ConcurrentRequest object contains the information of a concurrent request, as well as the convenient methods to retrieve the information of a concurrent request.

Table 2–4 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long,int), wait(long,int), wait(long,int)

2.3.1 Fields for ConcurrentRequest

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.3.2 Constructors for ConcurrentRequest

ConcurrentRequest()

```
public ConcurrentRequest()
```

Default constructor.

2.3.3 Methods for ConcurrentRequest

The following table is an index of the Class ConcurrentRequest methods:

Table 2–5 Methods for Class ConcurrentRequest

Method	Description
--------	-------------

Table 2–5 Methods for Class ConcurrentRequest

<code>getActualStartDate()</code>	Returns the actual start date of the concurrent request. <code>public java.sql.Timestamp getActualStartDate()</code>
<code>getName()</code>	Returns the name of the concurrent request. <code>public java.lang.String getName()</code>
<code>getPhase()</code>	Returns the phase name of the concurrent request. <code>public java.lang.String getPhase()</code>
<code>getPhaseCode()</code>	Returns the phase code of the concurrent request. <code>public java.lang.String getPhaseCode()</code>
<code>getRequestDate()</code>	Returns the request date of the concurrent request. <code>public java.sql.Timestamp getRequestDate()</code>
<code>getRequestedBy()</code>	Returns the user name of the person who requested the concurrent request. <code>public java.lang.String getRequestedBy()</code>
<code>getRequestId()</code>	Returns the request ID of the concurrent request. <code>public java.math.BigDecimal getRequestId()</code>
<code>getStatus()</code>	Returns the status name of the concurrent request. <code>public java.lang.String getStatus()</code>
<code>getStatusCode()</code>	Returns the status code of the concurrent request. <code>public java.lang.String getStatusCode()</code>
<code>load(BigDecimal)</code>	Returns concurrent request information from the database, given the specified concurrent request ID. <code>public static oracle.apps.qot.core.util.ConcurrentRequest load(java.math.BigDecimal requestId)</code> throws <code>SQLException</code> , <code>FrameworkException</code>

Table 2–5 Methods for Class ConcurrentRequest

setActualStartDate(Timestamp)	Sets the actual start date of the concurrent request. public void setActualStartDate (java.sql.Timestamp actualStartDate)
setName(String)	Sets the request name. public void setName (java.lang.String name)
setPhase(String)	Sets the phase name of the concurrent request. public void setPhase (java.lang.String phase)
setPhaseCode(String)	Sets the phase code of the concurrent request. public void setPhaseCode (java.lang.String phaseCode)
setRequestDate(Timestamp)	Sets the request date of the concurrent request. public void setRequestDate (java.sql.Timestamp requestDate)
setRequestedBy(String)	Sets the user name of the person who requested the concurrent request. public void setRequestedBy (java.lang.String requestedBy)
setRequestId(BigDecimal)	Sets the request ID of the concurrent request. public void setRequestId (java.math.BigDecimal requestId)
setStatus(String)	Sets the status name of the concurrent request. public void setStatus (java.lang.String status)
setStatusCode(String)	Sets the status code of the concurrent request. public void setStatusCode (java.lang.String statusCode)

Table 2–5 Methods for Class ConcurrentRequest

<code>toString()</code>	Returns a String representation of the ConcurrentRequest object. <code>public java.lang.String toString()</code>
-------------------------	---

getActualStartDate()

```
public java.sql.Timestamp getActualStartDate()
```

Returns the actual start date of the concurrent request.

Returns: The actual start date.

getName()

```
public java.lang.String getName()
```

Returns the name of the concurrent request.

Returns: The request name.

getPhase()

```
public java.lang.String getPhase()
```

Returns the phase name of the concurrent request.

Returns: The phase name.

getPhaseCode()

```
public java.lang.String getPhaseCode()
```

Returns the phase code of the concurrent request.

Returns: The phase code.

getRequestDate()

```
public java.sql.Timestamp getRequestDate()
```

Returns the request date of the concurrent request.

Returns: The request date.

getRequestedBy()

```
public java.lang.String getRequestedBy()
```

Returns the user name of the person who requested the concurrent request.

Returns: The user name of the person who requested the concurrent request.

getRequestId()

```
public java.math.BigDecimal getRequestId()
```

Returns the request ID of the concurrent request.

Returns: The request ID.

getStatus()

```
public java.lang.String getStatus()
```

Returns the status name of the concurrent request.

Returns: The status name.

getStatusCode()

```
public java.lang.String getStatusCode()
```

Returns the status code of the concurrent request.

Returns: The status code.

load(BigDecimal)

```
public static oracle.apps.qot.core.util.ConcurrentRequest  
load(java.math.BigDecimal requestId)  
throws SQLException, FrameworkException
```

Returns concurrent request information from the database, given the specified concurrent request ID.

Parameters: requestId - The concurrent request ID.

Returns: A ConcurrentRequest object with the appropriate fields populated.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setActualStartDate(Timestamp)

```
public void setActualStartDate(java.sql.Timestamp actualStartDate)
```

Sets the actual start date of the concurrent request.

Parameters: actualStartDate - The actual start date.

setName(String)

```
public void setName(java.lang.String name)
```

Sets the request name.

Parameters: name - The request name.

setPhase(String)

```
public void setPhase(java.lang.String phase)
```

Sets the phase name of the concurrent request.

Parameters: phase - The phase name.

setPhaseCode(String)

```
public void setPhaseCode(java.lang.String phaseCode)
```

Sets the phase code of the concurrent request.

Parameters: phaseCode - The phase code.

setRequestDate(Timestamp)

```
public void setRequestDate(java.sql.Timestamp requestDate)
```

Sets the request date of the concurrent request.

Parameters: requestDate - The request date.

setRequestedBy(String)

```
public void setRequestedBy(java.lang.String requestedBy)
```

Sets the user name of the person who requested the concurrent request.

Parameters: requestedBy - The user name of the person who requested the request .

setRequestId(BigDecimal)

```
public void setRequestId(java.math.BigDecimal requestId)
```

Sets the request ID of the concurrent request.

Parameters: requestId - The request ID.

setStatus(String)

public void **setStatus**(java.lang.String status)
Sets the status name of the concurrent request.

Parameters: status - The status name.

setStatusCode(String)

public void **setStatusCode**(java.lang.String statusCode)
Sets the status code of the concurrent request.

Parameters: statusCode - The status code.

toString()

public java.lang.String **toString**()
Returns a String representation of the ConcurrentRequest object.

Overrides: toString in class Object

Returns: A String representation of the ConcurrentRequest object.

2.4 Class Currency

```
java.lang.Object
|
+--oracle.apps.qot.core.util.Currency
public class Currency implements java.io.Serializable
```

All Implemented Interfaces: java.io.Serializable

The currency object contains the following information for a currency: currency code, currency name, currency symbol, and available price list IDs for the currency. Currency provides methods to retrieve a list of currencies and retrieve a specific currency.

Table 2–6 *Inherited Member Summary*

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.4.1 Fields for Class Currency

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.4.2 Constructors for Class Currency

Currency()

```
public Currency()
```

Default constructor.

Currency(String, String, HashMap)

```
public Currency(java.lang.String code, java.lang.String symbol,  
com.sun.java.util.collections.HashMap langNameMap)
```

Constructor.

Parameters:

code - The currency code.

symbol - The currency symbol.

langNameMap - The currency name and language mapping.

Currency(String, String, HashMap, BigDecimal[])

```
public Currency(java.lang.String code, java.lang.String symbol,  
com.sun.java.util.collections.HashMap langNameMap,  
java.math.BigDecimal[] priceListIds)
```

Constructor.

Parameters:

code - Currency code.

symbol - Currency symbol.

langNameMap - Translated name for the currency.

priceListIds - IDs of available price lists in this currency.

2.4.3 Methods for Class Currency

The following table is an index of the Class Currency methods:

Table 2–7 Methods for Class Currency

Methods	Description
getCurrencyCode()	Returns the currency code. <code>public java.lang.String getCurrencyCode()</code>
getFormatPattern()	Returns the currency format pattern. <code>public java.lang.String getFormatPattern()</code>
getLangNameMap()	Returns the language and name mapping. <code>public com.sun.java.util.collections.HashMap getLangNameMap()</code>
getName()	Returns the currency name in the current language. <code>public java.lang.String getName()</code>
getPriceListIds()	Returns the price list IDs for price lists available in this currency. <code>public java.math.BigDecimal[] getPriceListIds()</code>
getSymbol()	Returns the currency symbols. <code>public java.lang.String getSymbol()</code>
list()	Lists the available currencies (currencies which are enabled, active, and have price lists defined). <code>public static oracle.apps.qot.core.util.Currency[] list()</code> throws FrameworkException
load(String)	Returns a Currency object corresponding to currencyCode passed in as parameter. <code>public static oracle.apps.qot.core.util.Currency load(java.lang.String currencyCode)</code> throws FrameworkException

Table 2–7 Methods for Class Currency

<code>setPriceListIds(BigDecimal[] l())</code>	Sets the price list IDs. <pre>public void setPriceListIds(java.math.BigDecimal[] priceListIds)</pre>
<code>toString()</code>	Returns a String representation of the Currency object.. <pre>public java.lang.String toString()</pre>

getCurrencyCode()

```
public java.lang.String getCurrencyCode()
```

Returns the currency code.

Returns: The currency code.

getFormatPattern()

```
public java.lang.String getFormatPattern()
```

Returns the currency format pattern.

Returns: A String representing the currency format pattern.

getLangNameMap()

```
public com.sun.java.util.collections.HashMap getLangNameMap()
```

Returns the language and name mapping.

Returns: a HashMap containing the language and currency name mapping.

getName()

```
public java.lang.String getName()
```

Return currency name in the current language.

Returns: The currency name.

getPriceListIds()

```
public java.math.BigDecimal[] getPriceListIds()
```

Returns the price list IDs for price lists available in this currency.

Returns: Price list IDs for price lists available in this currency.

getSymbol()

```
public java.lang.String getSymbol()
```

Returns the currency symbol.

Returns: The currency symbol.

list()

```
public static oracle.apps.qot.core.util.Currency[] list()
```

```
throws FrameworkException
```

Lists the available currencies (currencies which are enabled, active, and have price lists defined)

Returns: A list of available currencies.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(String)

```
public static oracle.apps.qot.core.util.Currency
```

```
load(java.lang.String currencyCode)
```

```
throws FrameworkException
```

Returns a Currency object corresponding to the currency code passed in as parameter.

Parameters: currency Code - The currency code.

Returns: A Currency object corresponding to the currency code passed in as parameter.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setPriceListIds(BigDecimal[])

```
public void setPriceListIds(java.math.BigDecimal[] priceListIds)
```

Sets the price list IDs.

Parameters: priceListIds - An array of price list ID.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the Currency object.

Overrides: toString in class Object

Returns: A String representation of the Currency object.

2.5 Class InstallBaseRelType

```
java.lang.Object
|
+--oracle.apps.got.core.util.InstallBaseRelType
```

public class **InstallBaseRelType** implements java.io.Serializable

The InstallBaseRelType object contains the following information for an install base instance relationship type: relationship type code, and relationship direction. It provides methods for loading a specific install base instance relationship type.

Table 2–8 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.5.1 Fields for Class InstallBaseRelType

BI_DIR

```
public static final java.lang.String BI_DIR
```

Constant indicating bi-directional instance relationship.

OBJ_TO_SBJ_DIR

```
public static final java.lang.String OBJ_TO_SBJ_DIR
```

Constant indicating object-to-subject instance relationship.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

SBJ_TO_OBJ_DIR

```
public static final java.lang.String SBJ_TO_OBJ_DIR
```

Constant indicating subject-to-object instance relationship.

2.5.2 Constructors for Class InstallBaseRelType

InstallBaseRelType()

```
public InstallBaseRelType()
```

Default constructor.

2.5.3 Methods for Class InstallBaseRelType

The following table is an index of the Class InstallBaseRelType methods:

Table 2–9 *Methods for Class InstallBaseRelType*

Methods	Description
getDirection()	Returns the install base instance relationship direction. <pre>public java.lang.String getDirection()</pre>

Table 2–9 Methods for Class InstallBaseRelType

<code>getRelationshipTypeCode()</code>	Returns the install base instance relationship type code. <code>public java.lang.String getRelationshipTypeCode()</code>
<code>load(String)</code>	Returns an <code>InstallBaseRelType</code> object corresponding to the install base instance relationship type code passed in as parameter. <code>public static oracle.apps.qot.core.util.InstallBaseRelType load(java.lang.String relTypeCode) throws FrameworkException</code>
<code>toString()</code>	Returns a <code>String</code> representation of the object. <code>public java.lang.String toString()</code>

getDirection()

```
public java.lang.String getDirection()  
Returns the install base instance relationship direction.
```

Returns: The instance relationship direction.

getRelationshipTypeCode()

```
public java.lang.String getRelationshipTypeCode()  
Returns the install base instance relationship type code.
```

Returns: The relationship type code.

load(String)

```
public static oracle.apps.qot.core.util.InstallBaseRelType  
load(java.lang.String relTypeCode)  
throws FrameworkException  
Returns an InstallBaseRelType object corresponding to the install base instance  
relationship type code passed in as parameter.
```

Parameters: `relTypeCode` - Install base instance relationship type code.

Returns: An `InstallBaseRelType` object corresponding to the relationship type code passed in as parameter.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

toString()

public java.lang.String **toString()**
Returns a String representation of the object.

Overrides: toString in class Object

Returns: A String representation of the object.

2.6 Class MarketingSource

```
java.lang.Object
|
+--oracle.apps.qot.core.util.MarketingSource
```

public class **MarketingSource**

The MarketingSource object represents a marketing campaign which can be applied to a quote. This class provides methods to retrieve a specified marketing source and retrieve a list of marketing sources based on some search criteria.

Table 2–10 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.6.1 Fields for Class MarketingSource

CODE

```
public static final java.lang.String CODE
```

Indicates the search column marketing source code. Used as input to MarketingSource.list() method.

NAME

```
public static final java.lang.String NAME
```

Indicates the search column marketing source name. Used as input to MarketingSource.list() method.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.6.2 Constructors for Class MarketingSource

MarketingSource()

```
public MarketingSource()
```

Constructor.

MarketingSource(BigDecimal, String, BigDecimal, String, String, String)

```
public MarketingSource(java.math.BigDecimal _sourceCodeId,  
java.lang.String _sourceCode, java.math.BigDecimal _objectId,  
java.lang.String _sourceType, java.lang.String _typeMeaning,  
java.lang.String _name)
```

Constructor.

Parameters:

_sourceCodeId - Marketing source code ID.

_sourceCode - Marketing source code.

`_objectId` - Object ID of the object with this marketing source. For example, campaign ID if the marketing source is of type campaign.

`_sourceType` - Marketing source type.

`_typeMeaning` - Translated meaning for the marketing source type.

`_name` - Marketing source name.

2.6.3 Methods for Class MarketingSource

The following table is an index of the Class MarketingSource methods:

Table 2–11 *Methods for Class MarketingSource*

Methods	Description
<code>getName()</code>	Returns Translated marketing source name. <code>public java.lang.String getName()</code>
<code>getObjectId()</code>	Returns Object ID for the marketing source. For example, if source type is Campaign, this API will return Campaign ID. <code>public java.math.BigDecimal getObjectId()</code>
<code>getSourceCode()</code>	Returns marketing Source code. <code>public java.lang.String getSourceCode()</code>
<code>getSourceCodeId()</code>	Returns marketing Source code ID. <code>public java.math.BigDecimal getSourceCodeId()</code>
<code>getTypeCode()</code>	Returns marketing Source Type code. <code>public java.lang.String getTypeCode()</code>
<code>getTypeMeaning()</code>	Returns translated meaning of the marketing Source Type. <code>public java.lang.String getTypeMeaning()</code>

Table 2–11 Methods for Class MarketingSource

<code>list(String, int, int, int)</code>	<p>Lists marketing sources based on search criteria passed in as parameter.</p> <pre>public static oracle.apps.qot.util.QueryResultSet list(java.lang.String searchColumn, java.lang.String searchString, int batchSize, int startIndex, int resCount) throws FrameworkException, SQLException</pre>
<code>load(BigDecimal)</code>	<p>Loads marketing source based on source code ID passed in as parameter.</p> <pre>public static oracle.apps.qot.core.util.Marketingsource load(java.math.BigDecimal sourceCodeId) throws FrameworkException, SQLException</pre>
<code>toString()</code>	<p>Returns a String representation of the MarketingSource object.</p> <pre>public java.lang.String toString()</pre>

getName()

```
public java.lang.String getName()
```

Returns the translated marketing source name.

Returns: The translated marketing source name.

getObjectId()

```
public java.math.BigDecimal getObjectId()
```

Returns Object ID for the marketing source. For example, if source type is Campaign, this API will return Campaign ID.

Returns: Object ID for the marketing source.

getSourceCode()

```
public java.lang.String getSourceCode()
```

Returns the marketing source code.

Returns: The marketing source code.

getSourceCodeId()

```
public java.math.BigDecimal getSourceCodeId()
```

Returns the marketing source code ID.

Returns: The marketing source code ID.

getTypeCode()

```
public java.lang.String getTypeCode()
```

Returns the marketing source type code.

Returns: The marketing source type code.

getTypeMeaning()

```
public java.lang.String getTypeMeaning()
```

Returns the translated meaning of the marketing source type.

Returns: The translated meaning of the marketing source type.

list(String, String, int, int, int)

```
public static oracle.apps.qot.util.QueryResultSet
```

```
list(java.lang.String searchColumn, java.lang.String searchString,  
int batchSize, int startIndex, int resCount)
```

```
throws FrameworkException, SQLException
```

Lists marketing sources based on search criteria passed in as parameter.

Parameters:

searchColumn - Search column. Possible values:

- NAME - Marketing source name.
- CODE - Marketing source code.

searchString - Search string.

batchSize - The batch size displaying a search result page.

startIndex - start index indicating which row number to start retrieving data. -1 indicates the last page.

resultCount - the total number of objects to return. If -1, the count will be queried from the database.

Returns: QueryResultSet containing an array of MarketingSource objects.

Throws:

java.sql.SQLException - if a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.core.util.Marketingsource  
load ( java.math.BigDecimal sourceCodeId)  
throws FrameworkException, SQLException
```

Loads the marketing source based on source code ID passed in as parameter.

Parameters: sourceCodeId - Marketing source code ID.

Returns: MarketingSource object with the appropriate information populated.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the MarketingSource object.

Overrides: toString in class Object

Returns: A String representation of the MarketingSource object.

2.7 Class Opportunity

```
java.lang.Object  
|  
+--oracle.apps.qot.core.util.Opportunity
```

public class **Opportunity**

Opportunity provides the methods to retrieve a specified opportunity and retrieve the list of open opportunities.

Table 2–12 Inherited Member Summary**Methods inherited from class Object**

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

2.7.1 Fields for Class Opportunity

CUST_PARTY_ID

```
public static final java.lang.String CUST_PARTY_ID
```

Constant indicating that the search of opportunities is conducted on the customary party ID. This may be passed into API `list()` as a key to the `HashMap` parameter `SearchCriteria`.

OPPTY_NAME

```
public static final java.lang.String OPPTY_NAME
```

Constant indicating that the search of opportunities is conducted on the opportunity name. This may be passed into API `list()` as a key to the `HashMap` parameter `searchCriteria`.

OPPTY_NUMBER

```
public static final java.lang.String OPPTY_NUMBER
```

Constant indicating that the search of opportunities is conducted on the opportunity number. This may be passed into API `list()` as a key to the `HashMap` parameter `searchCriteria`.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.7.2 Constructors for Class Opportunity

Opportunity()

`public Opportunity()`
Default constructor.

2.7.3 Methods for Class Opportunity

The following table is an index of the Class Opportunity methods:

Table 2–13 *Methods for Class Opportunity*

Methods	Description
<code>getAddress1()</code>	Returns the first line of the address. <code>public java.lang.String getAddress1()</code>
<code>getAddress2()</code>	Returns the second line of the address. <code>public java.lang.String getAddress2()</code>
<code>getAddress3()</code>	Returns the third line of the address. <code>public java.lang.String getAddress3()</code>
<code>getAddress4()</code>	Returns the fourth line of the address. <code>public java.lang.String getAddress4()</code>
<code>getAddressId()</code>	Returns the address ID. <code>public java.math.BigDecimal getAddressId()</code>
<code>getCity()</code>	Returns the city in the address. <code>public java.lang.String getCity()</code>
<code>getContactId()</code>	Returns the party ID of the contact person. <code>public java.math.BigDecimal getContactId()</code>
<code>getContactName()</code>	Returns the name of the contact person. <code>public java.lang.String getContactName()</code>
<code>getCountry()</code>	Returns the translated country name in the address. <code>public java.lang.String getCountry()</code>
<code>getCountryCode()</code>	Returns the country code in the address. <code>public java.lang.String getCountryCode()</code>

Table 2–13 Methods for Class Opportunity

<code>getCounty()</code>	Returns the county in the address. <code>public java.lang.String getCounty()</code>
<code>getCurrencyCode()</code>	Returns the currency code. <code>public java.lang.String getCurrencyCode()</code>
<code>getCustomerId()</code>	Returns the customer party ID. <code>public java.math.BigDecimal getCustomerId()</code>
<code>getCustomerName()</code>	Returns the customer name. <code>public java.lang.String getCustomerName()</code>
<code>getCustomerPartyType()</code>	Returns the customer party type. <code>public java.lang.String getCustomerPartyType()</code>
<code>getDecisionDate()</code>	Returns the opportunity decision date. <code>public java.sql.Timestamp getDecisionDate()</code>
<code>getMarketingSourceId()</code>	Returns the marketing source ID. <code>public java.math.BigDecimal getMarketingSourceId()</code>
<code>getMarketingSourceName() ()</code>	Returns the marketing source name. <code>public java.lang.String getMarketingSourceName()</code>
<code>getName()</code>	Returns the opportunity name. <code>public java.lang.String getName()</code>
<code>getOpportunityId()</code>	Returns the opportunity ID. <code>public java.math.BigDecimal getOpportunityId()</code>
<code>getOpportunityNumber()</code>	Returns the opportunity number. <code>public java.lang.String getOpportunityNumber()</code>
<code>getPostalCode()</code>	Returns the postal code in the address. <code>public java.lang.String getPostalCode()</code>

Table 2–13 Methods for Class Opportunity

<code>getPrimaryContactId()</code>	Returns the party ID of the primary contact relationship. <code>public java.math.BigDecimal getPrimaryContactId()</code>
<code>getProvince()</code>	Returns the province in the address. <code>public java.lang.String getProvince()</code>
<code>getRelationshipCode()</code>	Returns the party relationship code. <code>public java.lang.String getRelationshipCode()</code>
<code>getRelationshipId()</code>	Returns the party relationship ID. <code>public java.math.BigDecimal getRelationshipId()</code>
<code>getSalesChannelCode()</code>	Returns the sales channel code. <code>public java.lang.String getSalesChannelCode()</code>
<code>getSalesStage()</code>	Returns the sales stage. <code>public java.lang.String getSalesStage()</code>
<code>getSalesStageId()</code>	Returns the sales stage ID. <code>public java.math.BigDecimal getSalesStageId()</code>
<code>getState()</code>	Returns the address state. <code>public java.lang.String getState()</code>
<code>getStatus()</code>	Returns the translated opportunity state name. <code>public java.lang.String getStatus()</code>
<code>getStatusCode()</code>	Returns the opportunity status code. <code>public java.lang.String getStatusCode()</code>
<code>getTotalAmount()</code>	Returns the total amount of the opportunity. <code>public java.math.BigDecimal getTotalAmount()</code>
<code>getWinProbability()</code>	Returns the win probability. <code>public java.math.BigDecimal getWinProbability()</code>

Table 2–13 Methods for Class Opportunity

<code>isContactAddress()</code>	Returns whether the address belongs to the contact. <code>public boolean isContactAddress()</code>
<code>list(HashMap, int, int, int)</code>	Lists the opportunities based on the search criteria passed in as parameter. Only the opportunities with the status OPEN will be listed. <code>public static oracle.apps.qot.util.QueryResultSet list(com.sun.java.util.collections.HashMap searchCriteria, int batchSize, int startIndex, int resultCount) throws FrameworkException, SQLException</code>
<code>load(BigDecimal)</code>	Loads opportunity information from the database for the opportunity ID passed in as parameter. <code>public static oracle.apps.qot.core.util.Opportunity load(java.math.BigDecimal opportunityId) throws FrameworkException, SQLException</code>
<code>setCustomerId(BigDecimal)</code>	Sets the customer party ID. <code>public void setCustomerId(java.math.BigDecimal customerId)</code>
<code>setDecisionDate(Timestamp)</code>	Sets the opportunity decision date. <code>public void setDecisionDate(java.sql.Timestamp decisionDate)</code>
<code>setName(String)</code>	Sets the opportunity name. <code>public void setName(java.lang.String name)</code>
<code>setOpportunityId(BigDecimal)</code>	Sets the opportunity ID. <code>public void setOpportunityId(java.math.BigDecimal opptyId)</code>
<code>setOpportunityNumber(String)</code>	Sets the opportunity number. <code>public void setOpportunityNumber(java.lang.String opptyNumber)</code>

Table 2–13 Methods for Class Opportunity

<code>setSalesStage(String)</code>	Sets the sales stage. <code>public void setSalesStage(java.lang.String salesStage)</code>
<code>setSalesStageId(BigDecimal)</code>	Sets the sales stage ID. <code>public void setSalesStageId(java.math.BigDecimal salesStageId)</code>
<code>setStatus(String)</code>	Sets the translated opportunity name status. <code>public void setStatus(java.lang.String status)</code>
<code>setStatusCode(String)</code>	Sets the opportunity status code. <code>public void setStatusCode(java.lang.String statusCode)</code>
<code>setTotalAmount(BigDecimal)</code>	Sets the total amount of the opportunity. <code>public void setTotalAmount(java.math.BigDecimal totalAmount)</code>
<code>setWinProbability(BigDecimal)</code>	Sets the win probability. <code>public void setWinProbability(java.math.BigDecimal winProbability)</code>
<code>toString()</code>	Returns a String representation of the opportunity object. <code>public java.lang.String toString()</code>

getAddress1()

```
public java.lang.String getAddress1()
```

Returns the first line of the address.

Returns: The first line of the address.

getAddress2()

```
public java.lang.String getAddress2()
```

Returns the second line of the address.

Returns: The second line of the address.

getAddress3()

```
public java.lang.String getAddress3()
```

Returns the third line of the address.

Returns: The third line of the address.

getAddress4()

```
public java.lang.String getAddress4()
```

Returns the fourth line of the address.

Returns: The fourth line of the address.

getAddressId()

```
public java.math.BigDecimal getAddressId()
```

Returns the address ID.

Returns: The address ID.

getCity()

```
public java.lang.String getCity()
```

Returns the city in the address.

Returns: The city in the address.

getContactId()

```
public java.math.BigDecimal getContactId()
```

Returns the party ID of the contact person.

Returns: The party ID of the contact person.

getContactName()

```
public java.lang.String getContactName()
```

Returns the name of the contact person.

Returns: The name of the contact person.

getCountry()

```
public java.lang.String getCountry()
```

Returns the translated country name in the address.

Returns: The translated country name in the address.

getCountryCode()

```
public java.lang.String getCountryCode()
```

Returns the country code in the address.

Returns: The country code in the address.

getCounty()

```
public java.lang.String getCounty()
```

Returns the county in the address.

Returns: The county in the address.

getCurrencyCode()

```
public java.lang.String getCurrencyCode()
```

Returns the currency code.

Returns: The currency code.

getCustomerId()

```
public java.math.BigDecimal getCustomerId()
```

Returns the customer party ID.

Returns: The customer party ID.

getCustomerName()

```
public java.lang.String getCustomerName()
```

Returns the customer name.

Returns: The customer name.

getCustomerPartyType()

```
public java.lang.String getCustomerPartyType()
```

Returns the customer party type.

Returns: The customer party type.

getDecisionDate()

```
public java.sql.Timestamp getDecisionDate()
```

Returns the opportunity decision date.

Returns: The decision date of the opportunity.

getMarketingSourceId()

```
public java.math.BigDecimal getMarketingSourceId()
```

Returns the marketing source ID.

Returns: The marketing source ID.

getMarketingSourceName()

```
public java.lang.String getMarketingSourceName()
```

Returns the marketing source name.

Returns: The marketing source name.

getName()

```
public java.lang.String getName()
```

Returns the opportunity name.

Returns: The opportunity name.

getOpportunityId()

```
public java.math.BigDecimal getOpportunityId()
```

Returns the opportunity ID.

Returns: The opportunity ID.

getOpportunityNumber()

```
public java.lang.String getOpportunityNumber()
```

Returns the opportunity number.

Returns: The opportunity number.

getPostalCode()

```
public java.lang.String getPostalCode()
```

Returns the postal code in the address.

Returns: The postal code in the address.

getPrimaryContactId()

```
public java.math.BigDecimal getPrimaryContactId()
```

Returns the party ID of the primary contact relationship.

Returns: The party ID of the primary contact relationship.

getProvince()

```
public java.lang.String getProvince()
```

Returns the province in the address.

Returns: The province in the address.

getRelationshipCode()

```
public java.lang.String getRelationshipCode()
```

Returns the party relationship code.

Returns: The party relationship code.

getRelationshipId()

```
public java.math.BigDecimal getRelationshipId()
```

Returns the party relationship ID.

Returns: The party relationship ID.

getSalesChannelCode()

```
public java.lang.String getSalesChannelCode()
```

Returns the sales channel code.

Returns: The sales channel code.

getSalesStage()

```
public java.lang.String getSalesStage()
```

Returns the sales stage.

Returns: The sales stage of the opportunity.

getSalesStageId()

```
public java.math.BigDecimal getSalesStageId()
```

Returns the sales stage ID.

Returns: The sales stage ID of the opportunity.

getState()

```
public java.lang.String getState()
```

Returns the state in the address.

Returns: The state in the address.

getStatus()

```
public java.lang.String getStatus()
```

Returns the translated opportunity status name.

Returns: The status name of the opportunity.

getStatusCode()

```
public java.lang.String getStatusCode()
```

Returns the opportunity status code.

Returns: The status code of the opportunity.

getTotalAmount()

```
public java.math.BigDecimal getTotalAmount()
```

Returns the total amount of the opportunity.

Returns: The total amount of the opportunity.

getWinProbability()

```
public java.math.BigDecimal getWinProbability()
```

Returns the win probability.

Returns: The win probability of the opportunity.

isContactAddress()

```
public boolean isContactAddress()
```

Returns whether the address belongs to the contact.

Returns: whether the address belongs to the contact.

list(HashMap, int, int, int)

```
public static oracle.apps.got.util.QueryResultSet  
list(com.sun.java.util.collections.HashMap searchCriteria,  
int batchSize, int startIndex, int resultCount)  
throws FrameworkException, SQLException
```

List opportunities based on search criteria passed in as parameter. Only the opportunities with the status **Open** will be listed.

Parameters:

searchCriteria - A HashMap object containing the search criteria. Possible key values are:

- OPPTY_NAME: A value indicating opportunity name.
- CUST_PARTY_ID: A value indicating sold-to customer party ID.
- OPPTY_NUMBER: A String indicating opportunity number.

batchSize - Batch size for the query.

startIndex - Start index indicating which row number to start retrieving data. -1 indicates the last rows.

resultCount - The total number of objects to return. If -1, the count will be queried from the database.

Returns: QueryResultSet object containing an array of Opportunity objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal)

```
public static oracle.apps.got.core.util.Opportunity  
load(java.math.BigDecimal opportunityId)  
throws FrameworkException, SQLException
```

Loads opportunity information from the database for the opportunity ID passed in as parameter.

Parameters: opportunityId - Opportunity ID corresponding to lead ID in AS_LEADS_ALL.

Returns: Opportunity object with the appropriate information populated.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

setCustomerId(BigDecimal)

```
public void setCustomerId(java.math.BigDecimal customerId)
```

Sets the customer party ID.

Parameters: customerId - The customer party ID.

setDecisionDate(Timestamp)

```
public void setDecisionDate(java.sql.Timestamp decisionDate)
```

Sets the opportunity decision date.

Parameters: decisionDate - The decision date.

setName(String)

```
public void setName(java.lang.String name)
```

Sets the opportunity name.

Parameters: name - The opportunity name.

setOpportunityId(BigDecimal)

```
public void setOpportunityId(java.math.BigDecimal opptyId)
```

Sets the opportunity ID.

Parameters: opptyId - The opportunity ID.

setOpportunityNumber(String)

```
public void setOpportunityNumber(java.lang.String opptyNumber)
```

Sets the opportunity number.

Parameters: opptyNumber - The opportunity number.

setSalesStage(String)

```
public void setSalesStage(java.lang.String salesStage)
```

Sets the sales stage.

Parameters: salesStage - The sales stage.

setSalesStageId(BigDecimal)

```
public void setSalesStageId(java.math.BigDecimal salesStageId)
```

Sets the sales stage ID.

Parameters: salesStageId - The sales stage ID.

setStatus(String)

```
public void setStatus(java.lang.String status)
```

Sets the translated opportunity status name.

Parameters: status - The translated opportunity status name.

setStatusCode(String)

```
public void setStatusCode(java.lang.String statusCode)
```

Sets the opportunity status code.

Parameters: statusCode - Opportunity status code.

setTotalAmount(BigDecimal)

```
public void setTotalAmount(java.math.BigDecimal totalAmount)
```

Sets the total amount of the opportunity.

Parameters: totalAmount - The total amount of the opportunity.

setWinProbability(BigDecimal)

```
public void setWinProbability(java.math.BigDecimal winProbability)
```

Sets the win probability.

Parameters: winProbability - The win probability.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the opportunity object.

Overrides: toString in class Object

Returns: A String representation of the opportunity object.

2.8 Class PaymentTerm

```
java.lang.Object
|
+--oracle.apps.qot.core.util.PaymentTerm
```

public class **PaymentTerm**

All Implemented Interfaces: java.lang.Cloneable

The PaymentTerm object contains information for payment terms. This class provides methods to retrieve a specified payment term, retrieve a list of payment terms, and determine the default payment term.

Table 2–14 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.8.1 Fields for Class PaymentTerm

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.8.2 Constructors for Class PaymentTerm

PaymentTerm()

```
public PaymentTerm()
```

Default constructor.

2.8.3 Methods for Class PaymentTerm

The following table is an index of the Class PaymentTerm methods:

Table 2–15 *Methods for Class PaymentTerm*

Methods	Description
determinePaymentTerm (BigDecimal, BigDecimal)	<p>Determines the default payment term based on the price agreement ID and customer account ID.</p> <pre>public static oracle.apps.got.core.util.PaymentTerm determinePaymentTerm(java.math.BigDecimal agreementId, java.math.BigDecimal custAcctId) throws FrameworkException, SQLException</pre>
getDescription()	<p>Returns translated payment term description in the current session language.</p> <pre>public java.lang.String getDescription()</pre>
getDescriptionMap()	<p>Returns HashMap containing translated payment term description for each language.</p> <pre>public com.sun.java.util.collections.HashMap getDescriptionMap()</pre>
getIndex()	<p>Returns index specifying the order of the payment term in a list</p> <pre>public int getIndex()</pre>
getName()	<p>Returns translated payment term name in the current session language.</p> <pre>public java.lang.String getName()</pre>
getNameMap()	<p>Returns HashMap containing translated payment term name for each language.</p> <pre>public com.sun.java.util.collections.HashMap getNameMap()</pre>
getTermId()	<p>Returns the term ID of the PaymentTerm..</p> <pre>public java.math.BigDecimal getTermId()</pre>
isEnabled()	<p>Returns whether payment term is enabled.</p> <pre>public boolean isEnabled()</pre>

Table 2–15 Methods for Class PaymentTerm

<code>list()</code>	<p>Lists all the enabled payment terms.</p> <p>Returns an array containing enabled payment terms.</p> <pre>public static oracle.apps.qot.core.util.PaymentTerm[] list() throws FrameworkException</pre>
<code>load(BigDecimal)</code>	<p>Returns a PaymentTerm object corresponding to the term ID passed in as parameter.</p> <pre>public static oracle.apps.qot.core.util.PaymentTerm load(java.math.BigDecimal paymentTermId) throws FrameworkException</pre>
<code>toString()</code>	<p>Returns a String representation of the PaymentTerm object.</p> <pre>public java.lang.String toString()</pre>

determinePaymentTerm(BigDecimal, BigDecimal)

```
public static oracle.apps.qot.core.util.PaymentTerm
determinePaymentTerm(java.math.BigDecimal agreementId,
java.math.BigDecimal custAcctId)
throws FrameworkException, SQLException
```

Determines the default payment term based on the price agreement ID and customer account ID.

The following defaulting sequence will be used:

1. Active payment term derived from the price agreement.
2. Active payment term derived from the customer account.

Parameters:

`agreementId` - Price agreement ID.

`custAcctId` - Sold-to customer's account ID.

Returns: A PaymentTerm object based on the price agreement ID and customer account ID.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

getDescription()

```
public java.lang.String getDescription()
```

Returns translated payment term description in the current session language.

Returns: Translated payment term description in the current session language.

getDescriptionMap()

```
public com.sun.java.util.collections.HashMap getDescriptionMap()
```

Returns HashMap containing translated payment term description for each language.

Returns: Translated payment term name for each language.

getIndex()

```
public int getIndex()
```

Returns index specifying the order of the payment term in a list

Returns: The order of payment term in a list.

getName()

```
public java.lang.String getName()
```

Returns translated payment term name in the current session language.

Returns: Translated payment term name in the current session language.

getNameMap()

```
public com.sun.java.util.collections.HashMap getNameMap()
```

Returns HashMap containing translated payment term name for each language.

Returns: Translated payment term name for each language.

getTermId()

```
public java.math.BigDecimal getTermId()
```

Returns the term ID of the PaymentTerm.

Returns: The term ID of PaymentTerm.

isEnabled()

```
public boolean isEnabled()
```

Returns whether payment term is enabled.

Returns: Whether payment term is enabled.

list()

```
public static oracle.apps.qot.core.util.PaymentTerm[] list()  
throws FrameworkException
```

Lists all the enabled payment terms.

Returns an array containing enabled payment terms.

Returns: An array of enabled PaymentTerm objects.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.core.util.PaymentTerm  
load(java.math.BigDecimal paymentTermId)  
throws FrameworkException
```

Returns a PaymentTerm object corresponding to the term ID passed in as parameter.

Checks if required payment term is present in cache. If present, returns the object from cache. Otherwise loads the payment term from the database.

Parameters: paymentTermId - PaymentTerm ID of the Payment term.

Returns: PaymentTerm object corresponding to the term ID passed in.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the PaymentTerm object.

Overrides: toString in class Object

Returns: A String representation of the PaymentTerm object.

2.9 Class PriceAgreement

```
java.lang.Object
|
+--oracle.apps.qot.core.util.PriceAgreement
```

public class **PriceAgreement**

PriceAgreement object contains the following information for a price agreement: agreement ID, agreement name, price list ID. The class provides methods to retrieve a specific price agreement and retrieve the list of available price agreements for a customer.

Table 2–16 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.9.1 Fields for Class PriceAgreement

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.9.2 Constructors for Class PriceAgreement

PriceAgreement()

```
public PriceAgreement()
Default constructor.
```

PriceAgreement(BigDecimal, String, BigDecimal)

```
public PriceAgreement(java.math.BigDecimal agreementId,
    java.lang.String agreementName, java.math.BigDecimal priceListId)
Constructor.
```

Parameters:

agreementID - Agreement ID.

agreementName - Translated name for the agreement.

priceListID - Price list ID.

2.9.3 Methods for Class PriceAgreement

The following table is an index of the Class PriceAgreement methods:

Table 2–17 Methods for Class PriceAgreement

Methods	Description
getAgreementId()	Returns price agreement ID. <code>public java.math.BigDecimal getAgreementId()</code>
getAgreementName()	Returns price agreement name. <code>public java.lang.String getAgreementName()</code>
getPriceListId()	Returns price list ID. <code>public java.math.BigDecimal getPriceListId()</code>
getTermId()	Returns payment term ID. <code>public java.math.BigDecimal getTermId()</code>
isActive()	Returns whether the price agreement is active. <code>public boolean isActive()</code>
list(BigDecimal, String)	List the price agreements that are available for the specified customer account and currency code. <code>public static oracle.apps.qot.core.util.PriceAgreement[] list(java.math.BigDecimal custAccountId, java.lang.String currencyCode)</code> throws FrameworkException, SQLException

Table 2–17 Methods for Class PriceAgreement

<code>load(BigDecimal)</code>	Returns PriceAgreement object corresponding to agreementId passed in as a parameter. <pre>public static oracle.apps.got.core.util.PriceAgreement load(java.math.BigDecimal agreementId) throws FrameworkException, SQLException</pre>
<code>setTermId(BigDecimal)</code>	Sets the payment term ID. <pre>public void setTermId(java.math.BigDecimal termId)</pre>
<code>toString()</code>	Returns a String representation of the PriceAgreement object. <pre>public java.lang.String toString()</pre>

getAgreementId()

```
public java.math.BigDecimal getAgreementId()
```

Returns the price agreement ID.

Returns: The price agreement ID.**getAgreementName()**

```
public java.lang.String getAgreementName()
```

Returns the price agreement name.

Returns: The agreement name.**getPriceListId()**

```
public java.math.BigDecimal getPriceListId()
```

Returns the price list ID.

Returns: The price list ID.**getTermId()**

```
public java.math.BigDecimal getTermId()
```

Returns the payment term ID.

Returns: The payment term ID.

isActive()

```
public boolean isActive()
```

Returns whether the price agreement is active.

Returns: Whether the price agreement is active.

list(BigDecimal, String)

```
public static oracle.apps.qot.core.util.PriceAgreement[]  
list(java.math.BigDecimal custAccountId,  
      java.lang.String currencyCode)  
throws FrameworkException, SQLException
```

List the price agreements that are available for the specified customer account and currency code.

Parameters:

custAccountId - Customer account ID.

currencyCode - Currency code.

Returns: PriceAgreement objects containing the price agreements that are available for the specified customer account and currency code.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.core.util.PriceAgreement  
load(java.math.BigDecimal agreementId)  
throws FrameworkException, SQLException
```

Returns PriceAgreement object corresponding to agreementId passed in as a parameter.

Parameters: agreementId - Price agreement ID.

Returns: PriceAgreement object corresponding to agreementId passed in as parameter.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

setTermId(BigDecimal)

public void **setTermId**(java.math.BigDecimal termId)

Sets the payment term ID.

Parameters: termId - The term ID of the payment term.

toString()

public java.lang.String **toString**()

Returns a String representation of the PriceAgreement object.

Overrides: toString in class Object

Returns: A String representation of the PriceAgreement object.

2.10 Class PriceList

```
java.lang.Object
|
+--oracle.apps.qot.core.util.PriceList
```

public class **PriceList** implements java.io.Serializable

All Implemented Interfaces: java.io.Serializable

The PriceList object contains the following information for a price list, e.g., header ID, name, currency code, etc. It provides methods to retrieve a specific price list and retrieve the price lists available for a currency.

Table 2–18 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.10.1 Fields for Class PriceList

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.10.2 Constructors for Class PriceList

PriceList()

```
public PriceList()
Default constructor.
```

PriceList(BigDecimal, HashMap, String)

```
public PriceList(java.math.BigDecimal priceListId,
com.sun.java.util.collections.HashMap langNameMap,
java.lang.String currencyCode)
```

Constructs an active price list of type **PRL** by setting price list ID, language-name mapping, and currency code.

Parameters:

priceListId - Price list header ID.

langNameMap - Translated price list names.

currencyCode - Price list currency code.

2.10.3 Methods for Class PriceList

The following table is an index of the Class PriceList methods:

Table 2–19 *Methods for Class PriceList*

Methods	Description
---------	-------------

Table 2–19 Methods for Class PriceList

determinePriceList (String, boolean, boolean, BigDecimal, BigDecimal, BigDecimal)	Returns the price list ID based on current selection in currency, price list, price agreement, order type and customer. <pre>public static oracle.apps.qot.core.util.PriceList determinePriceList(java.lang.String currencyCode, boolean isCurrencyReadOnly, boolean isCurrencyChanged, java.math.BigDecimal priceListId, java.math.BigDecimal priceAgrmtPriceListId, java.math.BigDecimal accountPriceListId, java.math.BigDecimal orderTypePriceListId) throws FrameworkException, SQLException</pre>
getCurrencyCode ()	Returns currency code for the price list. <pre>public java.lang.String getCurrencyCode()</pre>
getLangNameMap ()	Returns a price list language and name mapping. <pre>public com.sun.java.util.collections.HashMap getLangNameMap()</pre>
getListTypeCode ()	Returns the price list type code. <pre>public java.lang.String getListTypeCode()</pre>
getName ()	Returns price list name in the current language. <pre>public java.lang.String getName()</pre>
getPriceListId ()	Returns the price list ID. <pre>public java.math.BigDecimal getPriceListId()</pre>
isActive ()	Returns whether the price list is active or not. <pre>public boolean isActive()</pre>
list (String)	Lists price lists available for the currency code passed in as parameter. <pre>public static oracle.apps.qot.core.util.PriceList[] list(java.lang.String currencyCode) throws FrameworkException, SQLException</pre>

Table 2–19 Methods for Class PriceList

<code>load(BigDecimal)</code>	<p>Returns a PriceList object corresponding to price list ID passed in as parameter.</p> <pre>public static oracle.apps.qot.core.util.PriceList load(java.math.BigDecimal priceListId) throws FrameworkException</pre>
<code>toString()</code>	<p>Returns a String representation of PriceList object.</p> <pre>public java.lang.String toString()</pre>

determinePriceList(String, boolean, boolean, BigDecimal, BigDecimal, BigDecimal, BigDecimal)

```
public static oracle.apps.qot.core.util.PriceList
determinePriceList(java.lang.String currencyCode,
boolean isCurrencyReadOnly, boolean isCurrencyChanged,
java.math.BigDecimal priceListId,
java.math.BigDecimal priceAgrmtPriceListId,
java.math.BigDecimal accountPriceListId,
java.math.BigDecimal orderTypePriceListId)
throws FrameworkException, SQLException
```

Return price list ID based on current selection in currency, price list, price agreement, order type and customer.

Parameters:

`currencyCode` - The currency code.

`isCurrencyReadOnly` - True if the currency is read only.

`isCurrencyChanged` - True if the currency is changed.

`priceListId` - The price list ID.

`priceAgrmtPriceListId` - The default price list ID associated to the price agreement.

`accountPriceListId` - The default price list ID associated to the customer.

`orderTypePriceListId` - The default price list ID associated to the order type.

Returns: Price list object for the price list. The price list ID will be null if no price list is selected.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

getCurrencyCode()

```
public java.lang.String getCurrencyCode()
```

Returns the currency code for the price list.

Returns: The currency code for the price list.

getLangNameMap()

```
public com.sun.java.util.collections.HashMap getLangNameMap()
```

Returns price list language and name mapping.

Returns: A mapping between language and name.

getListTypeCode()

```
public java.lang.String getListTypeCode()
```

Returns the price list type code.

Returns: The price list type code.

getName()

```
public java.lang.String getName()
```

Returns the price list name in the current language.

Returns: The price list name in the current language.

getPriceListId()

```
public java.math.BigDecimal getPriceListId()
```

Returns the price list header ID.

Returns: The price list ID.

isActive()

```
public boolean isActive()
```

Returns whether the price list is active or not.

Returns: A boolean value indicating whether the price list is active.

list(String)

```
public static oracle.apps.qot.core.util.PriceList[]
```

```
list(java.lang.String currencyCode)
```

```
throws FrameworkException, SQLException
```

Lists price lists available for the currency code passed in as parameter.

Parameters: currencyCode - The currency code.

Returns: An array of PriceList objects containing the price lists available for the currency code passed in as parameter.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.core.util.PriceList
```

```
load(java.math.BigDecimal priceListId)
```

```
throws FrameworkException
```

Return PriceList object corresponding to price list ID passed in as parameter.

Parameters: priceListId - The price list header ID.

Returns: PriceList object corresponding to price list ID passed in as parameter.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of PriceList object.

Overrides: toString in class Object

Returns: A String representation of PriceList object.

2.11 Class PriceModifier

```
java.lang.Object
```

```
|
```

```
+--oracle.apps.qot.core.util.PriceModifier
```

public class PriceModifier

The PriceModifier object contains the following information for a price modifier, e.g., header ID, code, name, etc. The class provides the method to retrieve a list of promotions.

Table 2–20 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.11.1 Fields for Class PriceModifier

CODE

```
public static final java.lang.String CODE
```

Indicates that the search of price modifiers is conducted on price modifier code. This value will be passed as the `searchColumn` input parameter to `list()` API.

NAME

```
public static final java.lang.String NAME
```

Indicates that the search of price modifiers is conducted on price modifier name. This value will be passed as the `searchColumn` input to `list()` API.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.11.2 Constructors for Class PriceModifier

PriceModifier()

```
public PriceModifier()
```

Default constructor.

PriceModifier(BigDecimal, String, String)

```
public PriceModifier(java.math.BigDecimal headerId,
    java.lang.String code, java.lang.String name)
```

Constructor.

Parameters:

headerId - Price modifier header ID.

code - Price modifier code.

name - Price modifier name.

2.11.3 Methods for Class PriceModifier

The following table is an index of the Class PriceModifier methods:

Table 2–21 *Methods for Class PriceModifier*

Methods	Description
getCode()	Returns price modifier code. <code>public java.lang.String getCode()</code>
getHeaderId()	Returns list header ID. <code>public java.math.BigDecimal getHeaderId()</code>
getName()	Returns price modifier name. <code>public java.lang.String getName()</code>
listPromotion(String, String, int, int, int)	Lists promotions based on search criteria passed in as parameter. <code>public static oracle.apps.qot.util.QueryResultSet listPromotions(java.lang.String searchColumn, java.lang.String searchString, java.lang.String currencyCode, int batchSize, int startIndex, int resultCount)</code> <code>throws FrameworkException, SQLException</code>

Table 2–21 Methods for Class PriceModifier

<code>toString()</code>	Returns a String representation of PriceModifier object. <code>public java.lang.String toString()</code>
-------------------------	---

getCode()

```
public java.lang.String getCode()
```

Returns the price modifier code.

Returns: The price modifier code.

getHeaderId()

```
public java.math.BigDecimal getHeaderId()
```

Returns the list header ID.

Returns: The list header ID.

getName()

```
public java.lang.String getName()
```

Returns the price modifier name.

Returns: The price modifier name.

listPromotions(String, String, String, int, int, int)

```
public static oracle.apps.got.util.QueryResultSet  
listPromotions(java.lang.String searchColumn,  
java.lang.String searchString, java.lang.String currencyCode,  
int batchSize, int startIndex, int resultCount)  
throws FrameworkException, SQLException
```

Lists promotions based on search criteria passed in as parameter.

Parameters:

`searchColumn` - Search column. Possible values:

- `NAME` - Price modifier name.
- `CODE` - Price modifier code.

`searchString` - Search string.

currencyCode - Currency code.

batchSize - The batch size displaying a search result page.

startIndex - Start index indicating which row number to start retrieving data. -1 indicates the last page.

resultCount - The total number of objects to return. If -1, the count will be queried from the database.

Returns: A QueryResultSet containing an array of PriceModifier objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

toString()

```
public java.lang.String toString()
Returns a String representation of PriceModifier object.
```

Overrides: toString in class Object

Returns: A String representation of PriceModifier object.

2.12 Class QuoteStatus

```
java.lang.Object
|
+--oracle.apps.qot.core.util.QuoteStatus
```

public class **QuoteStatus** implements java.io.Serializable

All Implemented Interfaces: java.io.Serializable

QuoteStatus contains the following information for a quote status: quote status ID, status code, status meaning, whether update is allowed. QuoteStatus provides methods to determine if a transition is allowed to a specific status, to retrieve the list of LOV transitions that are allowed from the current status, and to list all quote statuses.

Table 2–22 Inherited Member Summary

Methods inherited from class Object

Table 2–22 Inherited Member Summary

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`,
`wait(long, int)`, `wait(long, int)`

2.12.1 Fields for Class QuoteStatus

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.12.2 Constructors for Class QuoteStatus

QuoteStatus()

```
public QuoteStatus()
```

Default constructor.

2.12.3 Methods for Class QuoteStatus

The following table is an index of the Class QuoteStatus methods:

Table 2–23 Methods for Class QuoteStatus

Methods	Description
<code>getLovTransitions()</code>	Returns an array of quote statuses to which this quote status has an LOV transition. <pre>public oracle.apps.got.core.util.QuoteStatus[] getLovTransitions() throws FrameworkException</pre>
<code>getMeaning()</code>	Returns translated quote status meaning based on current language in the session. <pre>public java.lang.String getMeaning()</pre>
<code>getQuoteStatusId()</code>	Returns quote status ID. <pre>public java.math.BigDecimal getQuoteStatusId()</pre>

Table 2–23 Methods for Class QuoteStatus

<code>getStatusCode()</code>	Returns quote status code. <code>public java.lang.String getStatusCode()</code>
<code>isTransitionAllowed(String)</code>	Returns whether a transition is allowed from this quote status to the quote status passed in as parameter. <code>public boolean isTransitionAllowed(java.lang.String toStatusCode)</code>
<code>isUpdateAllowed()</code>	Returns whether a quote in this status may be updated. <code>public boolean isUpdateAllowed()</code>
<code>list()</code>	Returns all quote statuses. <code>public static oracle.apps.qot.core.util.QuoteStatus[] list()</code> throws <code>FrameworkException</code>
<code>load(BigDecimal)</code>	Loads quote status information based on the quote status ID passed in as parameter. <code>public static oracle.apps.qot.core.util.QuoteStatus load(java.math.BigDecimal quoteStatusId)</code> throws <code>FrameworkException</code>
<code>load(String)</code>	Loads quote status information based on quote status code passed in as parameter. <code>public static oracle.apps.qot.core.util.QuoteStatus load(java.lang.String statusCode)</code> throws <code>FrameworkException</code>
<code>toString()</code>	Returns a String representation of the quote status object. <code>public java.lang.String toString()</code>

getLovTransitions()

```
public oracle.apps.qot.core.util.QuoteStatus[] getLovTransitions()
throws FrameworkException
```

Returns an array of quote statuses to which this quote status has an LOV transition.

Returns: An array of quote statuses to which this quote status has an LOV transition.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

getMeaning()

```
public java.lang.String getMeaning()
```

Returns the translated quote status meaning based on current language in the session.

Returns: The translated quote status meaning.

getQuoteStatusId()

```
public java.math.BigDecimal getQuoteStatusId()
```

Returns the quote status ID.

Returns: The quote status ID.

getStatusCode()

```
public java.lang.String getStatusCode()
```

Returns the quote status code.

Returns: The quote status code.

isTransitionAllowed(String)

```
public boolean isTransitionAllowed(java.lang.String toStatusCode)
```

Returns whether a transition is allowed from this quote status to the quote status passed in as parameter.

Parameters: toStatusCode - destination quote status code.

Returns: whether a transition is allowed from this quote status to the specified quote status

isUpdateAllowed()

```
public boolean isUpdateAllowed()
```

Returns whether a quote in this status may be updated.

Returns: Whether a quote in this status may be updated.

list()

```
public static oracle.apps.qot.core.util.QuoteStatus[] list()  
throws FrameworkException
```

Returns all quote statuses.

Returns: An array of quote status objects.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.core.util.QuoteStatus  
load(java.math.BigDecimal quoteStatusId)  
throws FrameworkException
```

Loads quote status information based on the quote status ID passed in as parameter.

Parameters: quoteStatusId - Quote status ID.

Returns: QuoteStatus object with the appropriate information populated.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(String)

```
public static oracle.apps.qot.core.util.QuoteStatus  
load(java.lang.String statusCode)  
throws FrameworkException
```

Loads quote status information based on quote status code passed in as parameter.

Parameters: statusCode - Quote status code.

Returns: QuoteStatus object with the appropriate information populated.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the quote status object.

Overrides: toString in class Object

Returns: A String representation of the quote status object.

2.13 Class Resource

```
java.lang.Object
|
+--oracle.apps.qot.core.util.Resource
```

public class **Resource**

The Resource object contains the following information for a resource, e.g., resource ID, sales rep ID, sales credit type ID, name, resource category, job title, email, phone, city, state, postal code, and country. It provides methods for retrieving a specific resource, listing available resources, and determining whether a resource is a sales representative.

Table 2–24 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.13.1 Fields for Class Resource

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.13.2 Constructors for Class Resource

Resource()

```
public Resource()
Default constructor.
```

Resource(BigDecimal, BigDecimal, BigDecimal, String, String, String,

String, String, String, String, String, String)

```
public Resource(java.math.BigDecimal resourceId,
java.math.BigDecimal salesRepId,
java.math.BigDecimal salesCreditTypeId, java.lang.String name,
java.lang.String resourceCategory, java.lang.String jobTitle,
java.lang.String email, java.lang.String phone,
java.lang.String city, java.lang.String state,
java.lang.String postalCode, java.lang.String country)
Constructor.
```

Parameters:

resourceId - Resource ID.

salesRepId - Sales Rep ID.

name - Resource Name.

resourceCategory - Resource Category.

jobTitle - Resource Job Title.

email - Resource Email.

phone - Resource Phone.

city - Resource City.

state - Resource State.

postalCode - Resource Postal Code.

country - Resource Country.

2.13.3 Methods for Class Resource

The following table is an index of the Class Resource methods:

Table 2–25 *Methods for Class Resource*

Methods	Description
---------	-------------

Table 2–25 Methods for Class Resource

determineSalesperson(BigDecimal)	Return resource ID for the default primary salesperson. This function should only be applied during the initialization of page Create Quote. <pre>public static java.math.BigDecimal determinesalesperson(java.math.BigDecimal loggedInUserResourceId) throws FrameworkException, SQLException</pre>
getCategory()	Returns the resource category code. <pre>public java.lang.String getCategory()</pre>
getCity()	Returns the resource city. <pre>public java.lang.String getCity()</pre>
getCountry()	Returns the resource country. <pre>public java.lang.String getCountry()</pre>
getEmail()	Returns the resource email. <pre>public java.lang.String getEmail()</pre>
getJobTitle()	Returns the resource job title. <pre>public java.lang.String getJobTitle()</pre>
getName()	Returns the resource name. <pre>public java.lang.String getName()</pre>
getPhone()	Returns the resource phone. <pre>public java.lang.String getPhone()</pre>
getPostalCode()	Returns the resource postal code. <pre>public java.lang.String getPostalCode()</pre>
getResourceId()	Returns the resource ID. <pre>public java.math.BigDecimal getResourceId()</pre>
getSalesCreditTypeId()	Returns the sales credit type ID. <pre>public java.math.BigDecimal getSalesCreditTypeId()</pre>
getSalesrepId()	Returns the sales rep ID. null if the resource is not a sales rep. <pre>public java.math.BigDecimal getSalesRepId()</pre>

Table 2–25 Methods for Class Resource

<code>getState()</code>	Returns the resource state. <code>public java.lang.String getState()</code>
<code>isSalesrep()</code>	Returns whether the resource is a sales rep. <code>public boolean isSalesRep()</code>
<code>List(Boolean, String, String, int, int, int)</code>	List resources based on search criteria passed in as parameter. If <code>onlySalesreps</code> is true, lists only sales reps which meet the criteria. Otherwise, lists all resources which meet the criteria. <code>public static oracle.apps.qot.util.QueryResultSet list(boolean onlySalesReps, java.lang.String searchString, java.lang.String categoryCode, int batchSize, int startIndex, int resultCount) throws FrameworkException, SQLException</code>
<code>Load(BigDecimal)</code>	Returns a Resource object corresponding to the resource ID passed in as parameter. <code>public static oracle.apps.qot.core.util.Resource load(java.math.BigDecimal resourceId) throws FrameworkException, SQLException</code>
<code>Load(String)</code>	Returns a Resource object corresponding to the user name passed in as parameter. <code>public static oracle.apps.qot.core.util.Resource load(java.lang.String findUserName) throws FrameworkException, SQLException</code>
<code>toString()</code>	Returns a String representation of the Resource object. <code>public java.lang.String toString()</code>

determineSalesperson(BigDecimal)

```
public static java.math.BigDecimal
determineSalesperson(java.math.BigDecimal loggedInUserResourceId)
throws FrameworkException, SQLException
```

Return resource ID for the default primary salesperson. This function should only be applied during the initialization of page Create Quote.

Parameters: `loggedInUserId` - Resource ID of the logged in user. Null if not a resource.

Returns: The resource ID for the default primary salesperson if any.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

getCategory()

```
public java.lang.String getCategory()
```

Returns the resource category code.

Returns: The resource category code.

getCity()

```
public java.lang.String getCity()
```

Returns the resource city.

Returns: The resource city.

getCountry()

```
public java.lang.String getCountry()
```

Returns the resource country.

Returns: The resource country.

getEmail()

```
public java.lang.String getEmail()
```

Returns the resource email.

Returns: The resource email.

getJobTitle()

```
public java.lang.String getJobTitle()
```

Returns the resource job title.

Returns: The resource job title.

getName()

```
public java.lang.String getName()
```

Returns the resource name.

Returns: The resource name.

getPhone()

```
public java.lang.String getPhone()
```

Returns the resource phone number.

Returns: The resource phone number.

getPostalCode()

```
public java.lang.String getPostalCode()
```

Return the resource postal code.

Returns: The resource postal code.

getResourceId()

```
public java.math.BigDecimal getResourceId()
```

Returns the resource ID.

Returns: The resource ID.

getSalesCreditTypeId()

```
public java.math.BigDecimal getSalesCreditTypeId()
```

Returns the sales credit type ID.

Returns: The sales credit type ID.

getSalesRepId()

```
public java.math.BigDecimal getSalesRepId()
```

Returns the sales rep ID.null if the resource is not a sales rep.

Returns: The sales rep ID.null if the resource is not a sales rep.

getState()

```
public java.lang.String getState()
```

Returns the resource state.

Returns: The resource state.

isSalesRep()

```
public boolean isSalesRep()
```

Returns whether the resource is a sales rep.

Returns: A boolean value indicating whether the resource is a sales rep. `true` if the resource is a sales rep; `false` otherwise.

list(boolean, String, String, int, int, int)

```
public static oracle.apps.got.util.QueryResultSet list(boolean  
onlySalesReps, java.lang.String searchString,  
java.lang.String categoryCode, int batchSize, int startIndex,  
int resultCount)  
throws FrameworkException, SQLException
```

List resources based on search criteria passed in as parameter. If `onlySalesReps` is `true`, lists only sales reps which meet the criteria. Otherwise, lists all resources which meet the criteria.

Parameters:

`onlySalesReps` - Whether to limit the search for sales reps.

`searchString` - Search string.

`categoryCode` - Resource category code.

`batchSize` - Batch size for the query.

`startIndex` - Start index indicating which row number to start retrieving data. `-1` indicates the last rows.

`resultCount` - The total number of objects to return. If `-1`, the count will be queried from the database.

Returns: A `QueryResultSet` containing an array of `Resource` objects.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.core.util.Resource  
load(java.math.BigDecimal resourceId)  
throws FrameworkException, SQLException
```

Returns a Resource object corresponding to the resource ID passed in as parameter.

Parameters: resourceId - Resource ID.

Returns: A Resource object corresponding to the resource ID passed in as parameter.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If database error occurs.

load(String)

```
public static oracle.apps.qot.core.util.Resource  
load(java.lang.String fndUserName)  
throws FrameworkException, SQLException
```

Returns a Resource object corresponding to the user name passed in as parameter.

Parameters: fndUserName - User name.

Returns: A Resource object corresponding to the user name passed in as parameter.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

toString()

```
public java.lang.String toString()  
Returns a String representation of the Resource object.
```

Overrides: toString in class Object

Returns: A String representation of the Resource object.

2.14 Class ResourceGroup

```
java.lang.Object
|
+--oracle.apps.qot.core.util.ResourceGroup
```

public class **ResourceGroup** implements java.io.Serializable

All Implemented Interfaces: java.io.Serializable

The ResourceGroup object contains the following information for a resource group, e.g., group ID, group name, etc. It provides methods for retrieving a specific resource group, listing resource groups to which a resource belongs, and listing resource groups based on search criteria.

Table 2–26 *Inherited Member Summary*

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.14.1 Fields for Class ResourceGroup

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.14.2 Constructors for Class ResourceGroup

ResourceGroup()

```
public ResourceGroup()
```

Default constructor.

ResourceGroup(BigDecimal, HashMap)

```
public ResourceGroup(java.math.BigDecimal groupId,
com.sun.java.util.collections.HashMap langNameMap)
```

Constructor.

Parameters:

groupId - Group ID.

langNameMap - Group language and name mapping.

2.14.3 Methods for Class ResourceGroup

The following table is an index of the Class ResourceGroup methods:

Table 2–27 *Methods for Class ResourceGroup*

Methods	Description
---------	-------------

Table 2–27 Methods for Class ResourceGroup

determineResourceGroup(BigDecimal, BigDecimal, BigDecimal)	Returns the resource group ID for the default primary sales group. This function should only be applied to page Create Quote. <pre>public static java.math.BigDecimal determineResourceGroup(java.math.BigDecimal loggedInUserResourceId, java.math.BigDecimal resourceId, java.math.BigDecimal opportunityId) throws FrameworkException, SQLException</pre>
getGroupId()	Returns the resource group ID. <pre>public java.math.BigDecimal getGroupId()</pre>
getGroupName()	Returns the resource group name. <pre>public java.lang.String getGroupName()</pre>
getLangNameMap()	Returns resource group language and name mapping. <pre>public com.sun.java.util.collections.HashMap getLangNameMap()</pre>
list()	Lists the available resource groups. <pre>public static oracle.apps.qot.core.util.ResourceGroup[] list() throws FrameworkException</pre>
list(BigDecimal)	Returns a list of resource groups to which the specified resource belongs. <pre>public static oracle.apps.qot.core.util.ResourceGroup[] list(java.math.BigDecimal resourceId) throws FrameworkException, SQLException</pre>
list(BigDecimal[])	Returns a HashMap object, in which key is resource ID and value is an array of ResourceGroup objects containing the list of resource groups to which the specified resources belong. <pre>public static com.sun.java.util.collections.HashMap list(java.math.BigDecimal[] resourceIds) throws FrameworkException, SQLException</pre>

Table 2–27 Methods for Class ResourceGroup

<code>list(String, int, int, int)</code>	<p>Lists resource groups based on search criteria passed in as parameter.</p> <pre>public static oracle.apps.qot.util.QueryResultSet list(java.lang.String searchString, int batchSize, int startIndex, int resultCount) throws FrameworkException, SQLException</pre>
<code>load(BigDecimal)</code>	<p>Returns a ResourceGroup object corresponding to the group ID passed in as parameter.</p> <pre>public static oracle.apps.qot.core.util.ResourceGroup load(java.math.BigDecimal groupId) throws FrameworkException, SQLException</pre>
<code>toString()</code>	<p>Returns a String representation of the ResourceGroup object.</p> <pre>public java.lang.String toString()</pre>

determineResourceGroup(BigDecimal, BigDecimal, BigDecimal)

```
public static java.math.BigDecimal
determineResourceGroup(java.math.BigDecimal loggedInUserResourceId,
java.math.BigDecimal resourceId, java.math.BigDecimal opportunityId)
throws FrameworkException, SQLException
```

Returns resource group ID for the default primary sales group. This function should only be applied to page Create Quote.

Parameters:

`loggedInUserResourceId` - Resource ID of the logged in user.null if not a resource.

`resourceId` - Resource ID of the primary salesperson.

`opportunityId` - ID of the opportunity where the quote is created from.

Returns: Resource group ID for the default primary salesperson if any.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

getGroupId()

```
public java.math.BigDecimal getGroupId()
```

Returns the resource group ID.

Returns: The resource group ID.

getGroupName()

```
public java.lang.String getGroupName()
```

Returns the resource group name.

Returns: The resource group name.

getLangNameMap()

```
public com.sun.java.util.collections.HashMap getLangNameMap()
```

Returns the resource group language and name mapping.

Returns: The resource group language and name mapping.

list()

```
public static oracle.apps.got.core.util.ResourceGroup[] list()  
throws FrameworkException
```

Lists the available resource groups.

Returns: A list of available resource groups.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

list(BigDecimal)

```
public static oracle.apps.got.core.util.ResourceGroup[]  
list(java.math.BigDecimal resourceId)  
throws FrameworkException, SQLException
```

Returns a list of resource groups to which the specified resource belongs.

Parameters: resourceId - Resource ID.

Returns: A list of resource groups to which the specified resource belongs.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

list(BigDecimal[])

```
public static com.sun.java.util.collections.HashMap
```

```
list(java.math.BigDecimal[] resourceIds)
```

```
throws FrameworkException, SQLException
```

Returns a `HashMap` object, in which key is resource ID and value is an array of `ResourceGroup` objects containing the list of resource groups to which the specified resources belong

Parameters: resourceIds - An array of resource IDs

Returns: `HashMap` A key value pair, in which the key is resource ID and the value is an array of `ResourceGroup` objects containing the list of resource groups to which the specified resources belong.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

list(String, int, int, int)

```
public static oracle.apps.qot.util.QueryResultSet
```

```
list(java.lang.String searchString, int batchSize, int startIndex,  
int resultCount)
```

```
throws FrameworkException, SQLException
```

Lists resource groups based on search criteria passed in as parameter.

Parameters:

searchString - Search string.

batchSize - The batch size displaying a search result page.

startIndex - start index indicating which row number to start retrieving data. -1 indicates the last page.

resultCount - the total number of objects to return. If -1, the count will be queried from the database.

Returns: A `QueryResultSet` object containing an array of `ResourceGroup` objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.core.util.ResourceGroup
```

```
load(java.math.BigDecimal groupId)
```

```
throws FrameworkException, SQLException
```

Return ResourceGroup object corresponding to the group ID passed in as parameter.

Parameters: groupId - Resource group ID.

Returns: A ResourceGroup corresponding to the group ID passed in as parameter.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the ResourceGroup object.

Overrides: toString in class Object

Returns: A String representation of the ResourceGroup object.

2.15 Class SalesCreditType

```
java.lang.Object
```

```
|
```

```
+- oracle.apps.qot.core.util.SalesCreditType
```

```
public class SalesCreditType implements java.io.Serializable
```

All Implemented Interfaces: java.io.Serializable

The SalesCreditType object contains the following information for a sales credit type: sales credit type ID, name, quota flag. It provides methods for loading a specific sales credit type and listing all sales credit types for a specific quota flag.

Table 2–28 Inherited Member Summary**Methods inherited from class Object**

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.15.1 Fields for Class SalesCreditType

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.15.2 Constructors for Class SalesCreditType

SalesCreditType()

```
public SalesCreditType()
```

Default constructor.

SalesCreditType(BigDecimal, String, String, int)

```
public SalesCreditType(java.math.BigDecimal salesCreditTypeId,
    java.lang.String name, java.lang.String quotaFlag, int index)
```

Constructor.

Parameters:

salesCreditTypeId - Sales credit type ID.

name - Sales credit type name.

quotaFlag - Indicates whether the sales credit type is revenue or non-revenue.

index - Index used for sorting.

2.15.3 Methods for Class SalesCreditType

The following table is an index of the Class SalesCreditType methods:

Table 2–29 *Methods for Class SalesCreditType*

Methods	Description
getName()	Returns the sales credit type name. <code>public java.lang.String getName()</code>
getQuotaFlag()	Returns the sales credit type quota flag. <code>public java.lang.String getQuotaFlag()</code>
getSalesCreditTypeId()	Returns the sales credit type ID. <code>public java.math.BigDecimal getSalesCreditTypeId()</code>
list(boolean)	Returns a list of sales credit types. If quotaCredits is true, only quota (i.e. revenue) credit types are returned. Otherwise, non-quota (i.e. non-revenue) credit types are returned. <code>public static oracle.apps.qot.core.util.SalesCreditType[] list(boolean quotaCredits) throws FrameworkException</code>
load(BigDecimal)	Return SalesCreditType object corresponding to the sales credit type ID passed in as parameter. <code>public static oracle.apps.qot.core.util.SalesCreditType load(java.math.BigDecimal salesCreditTypeId) throws FrameworkException</code>
toString()	Returns a String representation of the SalesCreditType object. <code>public java.lang.String toString()</code>

getName()

```
public java.lang.String getName()
```

Returns the sales credit type name.

Returns: The sales credit type name.

getQuotaFlag()

```
public java.lang.String getQuotaFlag()
```

Returns the sales credit type quota flag.

Returns: The sales credit type quote flag.

getSalesCreditTypeId()

```
public java.math.BigDecimal getSalesCreditTypeId()
```

Returns the sales credit type ID.

Returns: The sales credit type ID.

list(boolean)

```
public static oracle.apps.qot.core.util.SalesCreditType[]
```

```
list(boolean quotaCredits)
```

```
throws FrameworkException
```

Returns a list of sales credit types. If `quotaCredits` is `true`, only quota (i.e. revenue) credit types are returned. Otherwise, non-quota (i.e. non-revenue) credit types are returned.

Parameters: `quotaCredits` - whether to return quota credit types or non-quota credit types.

Returns: An array of sales credit types.

Throws: `oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.core.util.SalesCreditType
```

```
load(java.math.BigDecimal salesCreditTypeId)
```

```
throws FrameworkException
```

Returns a `SalesCreditType` object corresponding to the sales credit type ID passed in as parameter.

Parameters: `SaleCreditTypeId` - Sales credit type ID.

Returns: A `SalesCreditType` corresponding to the sales credit type ID passed in as parameter.

Throws: `oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the SalesCreditType object.

Overrides: toString in class Object

Returns: A String representation of the SalesCreditType object.

2.16 Class Territory

```
java.lang.Object
|
+--oracle.apps.got.core.util.Territory
```

public class **Territory** implements java.io.Serializable

All Implemented Interfaces: java.io.Serializable

The Territory object contains the following information for a territory: territory code, name, address style. It provides a method for loading a specific territory.

Table 2–30 *Inherited Member Summary*

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

2.16.1 Fields for Class Territory

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.16.2 Constructors for Class Territory

Territory()

```
public Territory()
```

Default constructor.

2.16.3 Methods for Class Territory

The following table is an index of the Class Territory methods:

Table 2–31 *Methods for Class Territory*

Methods	Description
<code>addName(String, String)</code>	<p>Adds the translated name for the corresponding language.</p> <pre>public void addName(java.lang.String lang, java.lang.String name)</pre>
<code>getAddressStyle()</code>	<p>Returns address style.</p> <pre>public java.lang.String getAddressStyle()</pre>
<code>getName()</code>	<p>Returns territory name.</p> <pre>public java.lang.String getName()</pre>
<code>getTerritoryCode()</code>	<p>Returns territory code.</p> <pre>public java.lang.String getTerritoryCode()</pre>
<code>load(String)</code>	<p>Return Territory object corresponding to the territory code passed in as parameter.</p> <pre>public static oracle.apps.qot.core.util.Territory load(java.lang.String territoryCode) throws FrameworkException</pre>
<code>setAddressStyle(String)</code>	<p>Sets the address style.</p> <pre>public void setAddressStyle(java.lang.String addressStyle)</pre>
<code>setTerritoryCode(String)</code>	<p>Sets the territory code.</p> <pre>public void setTerritoryCode(java.lang.String territoryCode)</pre>
<code>toString()</code>	<p>Returns a String representation of the territory object.</p> <pre>public java.lang.String toString()</pre>

addName(String, String)

```
public void addName(java.lang.String lang, java.lang.String name)
```

Adds the translated name for the corresponding language

Parameters: lang - Language code.

name - Translated name in the specified language.

getAddressStyle()

```
public java.lang.String getAddressStyle()
```

Returns the address style.

Returns: The address style.

getName()

```
public java.lang.String getName()
```

Returns the territory name.

Returns: The territory name.

getTerritoryCode()

```
public java.lang.String getTerritoryCode()
```

Returns the territory code.

Returns: The territory code.

load(String)

```
public static oracle.apps.qot.core.util.Territory  
load(java.lang.String territoryCode)  
throws FrameworkException
```

Return Territory object corresponding to the territory code passed in as parameter.

Parameters: territoryCode - Territory code.

Returns: Territory corresponding to the territory code passed in as parameter.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setAddressStyle(String)

```
public void setAddressStyle(java.lang.String addressStyle)
```

Sets the address style.

Parameters: addressStyle - Address style.

setTerritoryCode(String)

```
public void setTerritoryCode(java.lang.String territoryCode)
```

Sets the territory code.

Parameters: territoryCode - Territory code.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the territory object.

Overrides: toString in class Object

Returns: A String representation of the territory object.

2.17 Class TransactionType

```
java.lang.Object  
|  
+--oracle.apps.qot.core.util.TransactionType
```

public class **TransactionType** implements java.io.Serializable

All Implemented Interfaces: java.io.Serializable

The TransactionType object contains the following information for a transaction type, e.g., transaction type ID, transaction type code, name, price list ID, price list name, etc.

If a transaction type is an order type, transaction type ID, transaction type code, name, price list ID, and price list name will be populated.

If a transaction type is a line type, transaction type ID, transaction type code and name will be populated.

Table 2–32 Inherited Member Summary**Methods inherited from class Object**

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

2.17.1 Fields for Class Transaction Type

LINE

```
public static final java.lang.String LINE
```

Indicates transaction type is a line type.

ORDER

```
public static final java.lang.String ORDER
```

Indicates transaction type is an order type.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

2.17.2 Constructors for Class Transaction Type

TransactionType()

```
public TransactionType()
```

Default constructor.

TransactionType(BigDecimal, String, HashMap, BigDecimal)

```
public TransactionType(java.math.BigDecimal transactionTypeId,  
java.lang.String type,  
com.sun.java.util.collections.HashMap langNameMap,  
java.math.BigDecimal priceListId)
```

Constructor.

Parameters:

transactionTypeId - Transaction type ID.

type - Transaction type code. Possible values are ORDER or LINE.

langNameMap - Transaction type language and name mapping.

priceListId - Price list ID.

2.17.3 Methods for Class TransactionType

The following table is an index of the Class TransactionType methods:

Table 2–33 Methods for Class TransactionType

Methods	Description
getLangMapName()	Returns the transaction type language and name mapping. public com.sun.java.util.collections.HashMap getLangNameMap()
getName()	Returns the transaction type name. public java.lang.String getName()
getPriceListId()	Returns the price list ID for the transaction type. public java.math.BigDecimal getPriceListId()
getTransactionTypeId()	Returns the transaction type ID. public java.math.BigDecimal getTransactionTypeId()
getType()	Returns the transaction type indicating whether the transaction is a order type or line type. public java.lang.String getType()
listLineTypes(String, BigDecimal)	Lists the available line types for the specified line category code and order type. public static oracle.apps.qot.core.util.TransactionType[] listLineTypes (java.lang.String lineCategoryCode, java.math.BigDecimal orderTypeId) throws FrameworkException, SQLException

Table 2–33 Methods for Class TransactionType

<code>listOrderTypes()</code>	Lists the available order types. <pre>public static oracle.apps.qot.core.util.TransactionType[] listOrderTypes()</pre> throws FrameworkException, SQLException
<code>load(BigDecimal)</code>	Loads transaction type information based on the transaction type ID passed in as parameter. <pre>public static oracle.apps.qot.core.util.TransactionType load(java.math.BigDecimal typeId)</pre> throws FrameworkException, SQLException
<code>toString()</code>	Returns a String representation of the TransactionType object. <pre>public java.lang.String toString()</pre>

getLangNameMap()

```
public com.sun.java.util.collections.HashMap getLangNameMap()
```

Returns the transaction type language name mapping.

Returns: The transaction type language and name mapping.

getName()

```
public java.lang.String getName()
```

Returns the transaction type name.

Returns: The transaction type name.

getPriceListId()

```
public java.math.BigDecimal getPriceListId()
```

Returns the price list ID for the transaction type.

Returns: The price list ID for the transaction type.

getTransactionTypeId()

```
public java.math.BigDecimal getTransactionTypeId()
```

Returns the transaction type ID.

Returns: The transaction type ID.

getType()

```
public java.lang.String getType()
```

Returns the transaction type indicating whether the transaction is a order type or line type.

Returns: The transaction type. Possible values are TransactionType.LINE and TransactionType.ORDER.

listLineTypes(String, BigDecimal)

```
public static oracle.apps.qot.core.util.TransactionType[]
```

```
listLineTypes(java.lang.String lineCategoryCode,
```

```
java.math.BigDecimal orderTypeId)
```

```
throws FrameworkException, SQLException
```

Lists the available line types for the specified line category code and order type.

Parameters:

lineCategoryCode - Line category code.

orderTypeId - Order type ID.

Returns: A list of line types. Transaction type ID, transaction type code, and name will be populated.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

listOrderTypes()

```
public static oracle.apps.qot.core.util.TransactionType[]
```

```
listOrderTypes()
```

```
throws FrameworkException, SQLException
```

Lists the available order types.

Returns: A list of order types. Transaction type ID, transaction type code, name, and price list ID will be populated.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal)

```
public static oracle.apps.got.core.util.TransactionType
```

```
load(java.math.BigDecimal typeId)
```

```
throws FrameworkException, SQLException
```

Loads transaction type information based on the transaction type ID passed in as parameter.

Parameters: typeId - transaction type ID

Returns: A TransactionType object corresponding to the transaction type ID passed in as parameter.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the TransactionType object.

Overrides: toString in class Object

Returns: A String representation of the TransactionType object.

oracle.apps.qot.inventory

This section lists the Oracle Quoting Java APIs in the package `oracle.apps.qot.inventory`.

3.1 Package `oracle.apps.qot.inventory`

The package `oracle.apps.aso.qot.inventory` contains the APIs for Oracle Quoting inventory procedures. The table below lists a description for each class:

Table 3–1 *Class Summary for `oracle.apps.qot.inventory`*

Class	Description
Class <code>InventoryItem</code>	<code>InventoryItem</code> is used to model the basic attributes of an inventory item.
Class <code>InventoryUtil</code>	<code>InventoryUtil</code> provides inventory-related utility methods.
Class <code>ItemCategory</code>	<code>ItemCategory</code> is used to model an inventory item category and provides the method for retrieving the list of inventory item categories.
Class <code>ShippingMethod</code>	<code>ShippingMethod</code> is used to model a shipping method and provides the methods to list available shipping methods and retrieve basic information for shipping methods.
Exceptions	
Class <code>InventoryException</code>	<code>InventoryException</code> is thrown when an application error occurs in a method in package <code>oracle.apps.qot.inventory</code> .

3.2 Class InventoryException

```

java.lang.Object
|
+--java.lang.Throwable
|
+--java.lang.Exception
|
+--oracle.apps.jtf.base.resources.FrameworkException
|
+--oracle.apps.qot.inventory.InventoryException

```

public class **InventoryException** extends
oracle.apps.jtf.base.resources.FrameworkException

All Implemented Interfaces: java.io.Serializable

InventoryException is thrown when an application error occurs in a method in package oracle.apps.qot.inventory.

Table 3–2 Inherited Member Summary

Fields inherited from interface FrameworkException

DEBUG, ERROR, FATAL, INFORMATION, NONE, WARNING, defaultMsgMgr

Methods inherited from interface FrameworkException

addException(Exception), convertException(Exception), getAllInfo(),
getCurrentMessageManager(), getExceptionStack(), getExceptionStackRec(),
getExternException(), getKey(), getMessage(), getMessageManager(), getMessageStack(),
getParameters(), getParentExcept(), getRootException(), getRootExternExcept(),
getSeverity(), getThrowerInfo(), getWholeStack(), printAllInfo(PrintStream),
printAllInfo(PrintStream), printMesg(PrintStream), printMesg(PrintStream),
printMessageStack(PrintWriter), printMessageStack(PrintWriter),
printStackTrace(PrintWriter), printStackTrace(PrintWriter),
printThrowerInfo(PrintStream), printThrowerInfo(PrintStream),
printWholeStack(PrintStream), printWholeStack(PrintStream), setCurrents(),
setStackTrace(String)

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long,
int), wait(long, int)

Methods inherited from class Throwable

fillInStackTrace(), getLocalizedMessage(), printStackTrace(), toString()

3.2.1 Fields for Class InventoryException

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

3.2.2 Constructors for Class InventoryException

InventoryException()

```
public InventoryException()
```

Default constructor.

InventoryException(BigDecimal, String, String)

```
public InventoryException(java.math.BigDecimal err_msg_count,  
java.lang.String errorKey, java.lang.String param)
```

```
throws FrameworkException
```

Constructs an exception with the message count, error key, and parameter. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

param - A token for the error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

InventoryException(Exception, String)

```
public InventoryException(java.lang.Exception e,  
java.lang.String errorKey)
```

Constructs an exception with the given exception and error key and parameters.

Parameters:

e - The parent exception.

errorKey - Error key.

InventoryException(Exception, String, Hashtable)

```
public InventoryException(java.lang.Exception e,  
java.lang.String errorKey, java.util.Hashtable params)
```

Constructs an exception with the given exception, error key and parameters.

Parameters:

e - The parent exception.

errorKey - Error key.

params - Hashtable of tokens for the error key.

InventoryException(Exception, String, Object[])

```
public InventoryException(java.lang.Exception e,  
java.lang.String errorKey, java.util.Hashtable params)
```

Constructs an exception with the given exception, error key, and parameters.

Parameters:

e - The parent exception.

errorKey - Error key.

params - An array of tokens for the error key.

InventoryException(Exception, String, String)

```
public InventoryException(java.lang.Exception e,  
java.lang.String errorKey, java.lang.Object[] params)
```

Constructs an exception with the given exception, error key, and parameters.

Parameters:

e - The parent exception.

errorKey - Error key.

params - An array of tokens for the error key.

InventoryException(int, String)

```
public InventoryException(int err_msg_count,  
java.lang.String errorKey)  
throws FrameworkException
```

Constructs an exception with the message count and error key. Errors at the PL/SQL level will be retrieved.

Parameters:

`err_msg_count` - The number of messages to be returned from the pl/sql error stack.

`errorKey` - Error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

InventoryException(int, String, Object[])

```
public InventoryException(int err_msg_count,  
java.lang.String errorKey, java.lang.Object[] params)  
throws FrameworkException
```

Constructs an exception with the message count, error key, and parameter tokens. Errors at the PL/SQL level will be retrieved.

Parameters:

`err_msg_count` - The number of messages to be returned from the pl/sql error stack.

`errorKey` - Error key.

`params` - An array of tokens for the error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

InventoryException(int, String, String)

```
public InventoryException(int err_msg_count,  
java.lang.String errorKey, java.lang.String param)  
throws FrameworkException
```

Constructs an exception with the message count, error key, and parameter. Errors at the PL/SQL level will be retrieved.

Parameters:

`err_msg_count` - The number of messages to be returned from the pl/sql error stack.

`errorKey` - Error key.

`param` - A token for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

InventoryException(String)

```
public InventoryException(java.lang.String errorKey)
```

Constructs an exception with the error key.

Parameters: errorKey - Error key.

InventoryException(String, Object[])

```
public InventoryException(java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs an exception with the error key and parameters.

Parameters:

errorKey - Error key.

params - An array of tokens for the error key.

InventoryException(String, String)

```
public InventoryException(java.lang.String err_msg,  
java.lang.String errorKey)
```

Constructs an exception with the error message and error key.

Parameters:

err_msg - Error message.

errorKey - Error key.

InventoryException(String, String, Object[])

```
public InventoryException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.Object[] params)
```

Constructs an exception with the given error message, error key, and parameters

Parameters:

err_msg - Error message.

errorKey - Error key.

params - An array of tokens for error key.

InventoryException(String, String, String)

```
public InventoryException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.String param)
```

Constructs an exception with the given error message, error key, and parameter

Parameters:

err_msg - Error message.

errorKey - Error key.

param - Token for the error key.

3.3 Class InventoryItem

```
java.lang.Object
|
+--oracle.apps.qot.inventory.InventoryItem
```

public class **InventoryItem**

InventoryItem is used to model the basic attributes of an inventory item. It provides the methods to retrieve the following information for inventory items: basic attributes, included warranties, available services, and related items. The Inventory item object contains the following information for an inventory item: inventory organization ID, Inventory item ID, part number, description, primary unit of measure code, service item flag, serviceable flag, bom item type, returnable flag, and shippable flag.

Table 3–3 *Inherited Member Summary*

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

3.3.1 Fields for Class InventoryItem

MODEL

```
public static final int MODEL
```

Indicates BOM item type model.

OPTION_CLASS

```
public static final int OPTION_CLASS
```

Indicates BOM item type option class.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

3.3.2 Constructors for Class InventoryItem

InventoryItem()

```
public InventoryItem()
```

Default constructor.

InventoryItem(BigDecimal, BigDecimal, String, String, String, String, String, int, String, String, String, BigDecimal, String, BigDecimal, String)

```
public InventoryItem(java.math.BigDecimal invOrgId,  
java.math.BigDecimal invItemId, java.lang.String concatSegments,  
java.lang.String description, java.lang.String primaryUomCode,  
java.lang.String servProdFlag, java.lang.String servItemFlag,  
int bomItemType, java.lang.String bomItemTypeMeaning,  
java.lang.String itemTypeCode, java.lang.String itemTypeMeaning,  
java.math.BigDecimal durationQty,  
java.lang.String durationPeriodCode, java.math.BigDecimal relnTypeId,  
java.lang.String relnTypeMeaning)
```

Constructor.

Parameters:

invOrgId - Inventory organization ID.

invItemId - Inventory item ID.

concatSegments - Concatenated segments representing item part number.

description - Inventory item description.

primaryUomCode - Primary unit of measure code.

servProdFlag - Whether the item is serviceable. Possible values:

- Y - The item is serviceable.
- N - The item is not serviceable.

servItemFlag - Whether the item is a service. Possible values:

- Y - Item is a service.
- N - Item is not a service.

bomItemType - BOM item type.

bomItemTypeMeaning - Translated meaning for BOM item type.

itemTypeCode - Item type code.

itemTypeMeaning - Item type meaning.

durationQty - Service duration.

durationPeriodCode - Service duration period code.

relnTypeId - Relationship type ID.

relnTypeMeaning - Translated meaning for relationship type.

InventoryItem(BigDecimal, BigDecimal, String, String, String, String, String, int, String, String, String, BigDecimal, String, String, String)

```
public InventoryItem(java.math.BigDecimal invOrgId,  
java.math.BigDecimal invItemId, java.lang.String concatSegments,  
java.lang.String description, java.lang.String primaryUomCode,  
java.lang.String servProdFlag, java.lang.String servItemFlag,  
int bomItemType, java.lang.String bomItemTypeMeaning,  
java.lang.String itemTypeCode, java.lang.String itemTypeMeaning,  
java.math.BigDecimal durationQty,  
java.lang.String durationPeriodCode,  
java.lang.String shippableItemFlag, java.lang.String returnableFlag)  
Constructor.
```

Parameters:

invOrgId - Inventory organization ID.

invItemId - Inventory item ID.

concatSegments - Concatenated segments representing item part number.

description - Inventory item description.

primaryUomCode - Primary unit of measure code.

servProdFlag - Whether the item is serviceable. Possible values:

- Y - The item is serviceable.
- N - The item is not serviceable.

servItemFlag - Whether the item is a service. Possible values:

- Y - Item is a service.
- N - Item is not a service.

bomItemType - BOM item type.

bomItemTypeMeaning - Translated meaning for BOM item type.

itemTypeCode - Item type code.

itemTypeMeaning - Item type meaning.

durationQty - Service duration.

durationPeriodCode - Service duration period code.

shippableItemFlag - Whether the item is shippable. Possible values:

- Y - The item is shippable.
- N - The item is not shippable.

returnableItemFlag - Whether the item is returnable. Possible values:

- Y - The item is returnable.
- N - The item is not returnable.

3.3.3 Methods for Class InventoryItem

The following table is an index of the Class InventoryItem methods:

Table 3–4 Methods for Class InventoryItem

Method	Description
getBomType()	Returns the translated meaning of BOM item type. <code>public java.lang.String getBomType()</code>
getBomTypeCode()	Returns the BOM Item Type. <code>public int getBomTypeCode()</code>
getContainedModelIds()	Returns an array of inventory item IDs for all models in the given container. <code>public java.math.BigDecimal[] getContainedModelIds()</code> throws <code>SQLException</code> , <code>FrameworkException</code>
getInvOrgId()	Returns the Inventory organization ID. <code>public java.math.BigDecimal getInvOrgId()</code>
getItemId()	Returns the Inventory item ID. <code>public java.math.BigDecimal getItemId()</code>

Table 3–4 Methods for Class InventoryItem

Method	Description
<code>getItemType()</code>	Returns the translated meaning for the user-defined item type. <code>public java.lang.String getItemType()</code>
<code>getItemTypeCode()</code>	Returns the user-defined item type code. <code>public java.lang.String getItemTypeCode()</code>
<code>getName()</code>	Returns the translated inventory item name. <code>public java.lang.String getName()</code>
<code>getPartNumber()</code>	Returns the inventory part number. <code>public java.lang.String getPartNumber()</code>
<code>getPrimaryUom()</code>	Returns the translated name for the primary unit of measure. <code>public java.lang.String getPrimaryUom()</code>
<code>getPrimaryUomCode()</code>	Returns the primary unit of measure code for the inventory item. <code>public java.lang.String getPrimaryUomCode()</code>
<code>getReInTypeId()</code>	Returns the relationship type ID if the inventory item was retrieved by calling <code>listRelatedItems()</code> . Otherwise, returns null. <code>public java.math.BigDecimal getReInTypeId()</code>
<code>getReInTypeMeaning()</code>	Returns the translated relationship type meaning if the inventory item was retrieved by calling <code>listRelatedItem()</code> . Otherwise, returns null. <code>public java.lang.String getReInTypeMeaning()</code>
<code>getSrvDuration()</code>	Returns the service duration. <code>public java.math.BigDecimal getSrvDuration()</code>
<code>getSrvDurationPeriod()</code>	Returns the translated meaning of service duration period. <code>public java.lang.String getSrvDurationPeriod()</code>
<code>getSrvDurationPeriodCode()</code>	Returns the service duration period code. <code>public java.lang.String getSrvDurationPeriodCode()</code>

Table 3–4 Methods for Class InventoryItem

Method	Description
getUomCodes()	Returns the UOM codes for the inventory item if the UOM codes have not been loaded into the item. Otherwise, returns null. public java.lang.String[] getUomCodes()
isContainer	Returns whether the given model item is a container. public boolean isContainer() throws SQLException, FrameworkException
isReturnable()	Returns whether the item is returnable. public boolean isReturnable()
isServiceable()	Returns whether the inventory item is serviceable. public boolean isServiceable()
isServiceItem()	Returns whether the inventory item is a service. public boolean isServiceItem()
isShippable()	Returns whether the inventory item is shippable. public boolean isShippable()
list(String, BigDecimal, BigDecimal, BigDecimal, int, int, int)	Lists inventory items based on search criteria passed in as parameter. public static oracle.apps.qot.util.QueryResultSet list (java.lang.String searchString, java.math.BigDecimal invOrgId, java.math.BigDecimal categorySetId, java.math.BigDecimal categoryId, java.math.BigDecimal priceListId, int batchSize, int startIndex, int resCount) throws FrameworkException, SQLException

Table 3–4 Methods for Class InventoryItem

Method	Description
listAvailableServices(BigDecimal, BigDecimal, BigDecimal, String, Timestamp)	<p>Lists the available services for the specified inventory item, striped by customer, item revision, and request date.</p> <pre>public static oracle.apps.qot.inventory.InventoryItem[] listAvailableServices(java.math.BigDecimal invOrgId, java.math.BigDecimal itemId, java.math.BigDecimal custAccountId, java.lang.String itemRevision, java.sql.Timestamp requestDate) throws FrameworkException, SQLException, Inv entoryException</pre>
listIncludedWarranties(BigDecimal, BigDecimal)	<p>Lists the included warranties for the specified inventory item.</p> <pre>public static oracle.apps.qot.inventory.InventoryItem[] listIncludedWarranties(java.math.BigDecimal invOrgId, java.math.BigDecimal itemId) throws FrameworkException, SQLException, Inv entoryException</pre>
listRelatedItems(BigDecimal, BigDecimal, int, int)	<p>Lists the inventory items for relationship type ID passed on as parameter. If relnTypeId is null, list related inventory items for all relationship types. Relationship type ID will be populated in the inventory items returned.</p> <pre>public static oracle.apps.qot.util.QueryResultSet listRelatedItems(java.math.BigDecimal invOrgId, java.math.BigDecimal itemId, java.math.BigDecimal relnTypeId, int batchSize, int startIndex, int resCount) throws FrameworkException, SQLException</pre>
load(BigDecimal, BigDecimal)	<p>Returns the InventoryItem object corresponding to the inventory organization ID and inventory item ID passed in as parameter.</p> <pre>public static oracle.apps.qot.inventory.InventoryItem load(java.math.BigDecimal invOrgId, java.math.BigDecimal itemId) throws FrameworkException, SQLException</pre>

Table 3–4 Methods for Class InventoryItem

Method	Description
load(BigDecimal, BigDecimal[])	<p>Return InventoryItem objects corresponding to the inventory organization ID and inventory item IDs passed in as parameter.</p> <pre>public static oracle.apps.qot.inventory.InventoryItem[] load(java.math.BigDecimal invOrgId, java.math.BigDecimal[] itemIds) throws FrameworkException, SQLException</pre>
loadUomCodes(InventoryItem[])	<p>For each inventory item passed in as parameter, loads the list of available unit of measure codes into the item. After calling this API, use the API getUomCodes() on each inventory item to retrieve the list of available unit of measure codes.</p> <pre>public static void loadUomCodes(oracle.apps.qot.inventory.Inven toryItem[] items) throws FrameworkException, SQLException</pre>
setBomItemType(String)	<p>Sets the value for BOM item type meaning.</p> <pre>public void setBomItemType(java.lang.String bomItemTypeMeaning)</pre>
setBomTypeCode(int)	<p>Sets the value for BOM item type code.</p> <pre>public void setBomTypeCode(int bomItemTypeCode)</pre>
setInvOrgId(BigDecimal)	<p>Sets the value for the inventory organization ID.</p> <pre>public void setInvOrgId(java.math.BigDecimal invOrgId)</pre>
setItemID(BigDecimal)	<p>Sets the value for the inventory item ID.</p> <pre>public void setItemId(java.math.BigDecimal itemId)</pre>
setItemType(String)	<p>Sets the value for the item type meaning.</p> <pre>public void setItemType(java.lang.String itemTypeMeaning)</pre>
setItemTypeCode(String)	<p>Sets the value for the item type code.</p> <pre>public void setItemTypeCode(java.lang.String itemTypeCode)</pre>

Table 3–4 Methods for Class InventoryItem

Method	Description
setName(String)	Sets the value for the inventory item name. <code>public void setName(java.lang.String name)</code>
setPartNumber(String)	Sets the value for the part number. <code>public void setPartNumber(java.lang.String partNumber)</code>
setPrimaryUom(String)	Sets the value for primary UOM name. <code>public void setPrimaryUom(java.lang.String primaryUomName)</code>
setPrimaryUomCode(String)	Sets the value for the primary UOM code. <code>public void setPrimaryUomCode(java.lang.String primaryUomCode)</code>
setReInTypeId(BigDecimal)	Sets the value for the relationship type ID. <code>public void setReInTypeId(java.math.BigDecimal reInTypeId)</code>
setReInTypeMeaning(String)	Sets the value for relationship type meaning. <code>public void setReInTypeMeaning(java.lang.String reInTypeMeaning)</code>
setReturnable(boolean)	Sets the value for whether the item is returnable. <code>public void setReturnable(boolean isReturnable)</code>
setServiceable(boolean)	Sets the value for whether the item is serviceable. <code>public void setServiceable(boolean isServiceable)</code>
setServiceItem(boolean)	Sets the value for whether the item is a service item. <code>public void setServiceItem(boolean isServiceItem)</code>
setShippable(boolean)	Sets the value for whether the item is shippable. <code>public void setShippable(boolean isShippable)</code>

Table 3–4 Methods for Class InventoryItem

Method	Description
setSrvDuration(BigDecimal)	Sets the value for service duration. public void setSrvDuration (java.math.BigDecimal durationQty)
setSrvDurationPeriod(String)	Sets the value for service duration period meaning. public void setSrvDurationPeriod (java.lang.String durationPeriodMeaning)
setSrvDurationPeriodCode(String)	Sets the value for service duration period code. public void setSrvDurationPeriodCode (java.lang.String durationPeriodCode)
setUomCodes(String[])	Sets the value for the UOM codes. public void setUomCodes (java.lang.String[] uomCodes)
toString()	Returns a String representation of the inventory item object. public java.lang.String toString ()

getBomType()

```
public java.lang.String getBomType()
```

Returns the translated meaning of BOM item type.

Returns: The translated meaning of BOM item type.

getBomTypeCode()

```
public int getBomTypeCode()
```

Returns the Bom item type.

Returns: The BOM item type.

getContainedModelIds()

```
public java.math.BigDecimal[] getContainedModelIds()  
throws SQLException, FrameworkException
```

Returns an array of inventory item IDs for all models in the given container.

Returns: An array of inventory item IDs for models in the container.

Throws: java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

getInvOrgId()

```
public java.math.BigDecimal getInvOrgId()
```

Returns the inventory organization ID.

Returns: The inventory organization ID.

getItemId()

```
public java.math.BigDecimal getItemId()
```

Returns the inventory item ID.

Returns: Inventory item ID.

getItemType()

```
public java.lang.String getItemType()
```

Returns the translated meaning for the user defined item type.

Returns: The translated meaning for the user defined item type.

getItemTypeCode()

```
public java.lang.String getItemTypeCode()
```

Returns the user defined item type code.

Returns: The user defined item type code.

getName()

```
public java.lang.String getName()
```

Returns the translated inventory item name.

Returns: The translated inventory item name.

getPartNumber()

```
public java.lang.String getPartNumber()
```

Returns the inventory item part number.

Returns: The inventory item part number.

getPrimaryUom()

```
public java.lang.String getPrimaryUom()
```

Returns the translated name for the primary unit of measure.

Returns: The translated name for the primary unit of measure.

getPrimaryUomCode()

```
public java.lang.String getPrimaryUomCode()
```

Returns the primary unit of measure code for the inventory item.

Returns: The primary unit of measure code for the inventory item.

getReInTypeId()

```
public java.math.BigDecimal getReInTypeId()
```

Returns relationship type ID if the Inventory Item was retrieved by calling `listRelatedItems()`. Otherwise, returns null.

Returns: The relationship type ID if the inventory item was retrieved by calling `listRelatedItems()`. Otherwise, returns null.

getReInTypeMeaning()

```
public java.lang.String getReInTypeMeaning()
```

Returns the translated relationship type meaning if the inventory item was retrieved by calling `listRelatedItems()`. Otherwise, returns null.

Returns: The translated relationship type meaning if the inventory item was retrieved by calling `listRelatedItems()`. Otherwise, returns null.

getSrvDuration()

```
public java.math.BigDecimal getSrvDuration()
```

Returns the service duration.

Returns: The service duration.

getSrvDurationPeriod()

```
public java.lang.String getSrvDurationPeriod()
```

Returns the translated meaning of service duration period.

Returns: The translated meaning of service duration period.

getSrvDurationPeriodCode()

```
public java.lang.String getSrvDurationPeriodCode()
```

Returns the service duration period code.

Returns: The service duration period code.

getUomCodes()

```
public java.lang.String[] getUomCodes()
```

Returns UOM codes for the inventory item if UOM codes have been loaded into the item. Otherwise returns null.

Returns: UOM codes if the UOM codes have been loaded into the item.

isContainer()

```
public boolean isContainer()
```

```
throws SQLException, FrameworkException
```

Returns whether the given model item is a container.

Returns: True if model item is a container. False otherwise.

Throws: java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

isReturnable()

```
public boolean isReturnable()
```

Returns whether the item is returnable.

Returns: Whether the item is returnable.

isServiceable()

```
public boolean isServiceable()
```

Returns whether the inventory item is serviceable.

Returns: Whether the inventory item is serviceable.

isServiceItem()

```
public boolean isServiceItem()
```

Returns whether the inventory item is a service item.

Returns: Whether the inventory item is service item.

isShippable()

```
public boolean isShippable()
```

Returns whether the item is shippable.

Returns: whether the item is shippable.

list(String, BigDecimal, BigDecimal, BigDecimal, BigDecimal, int, int, int)

```
public static oracle.apps.qot.util.ResultSet
list(java.lang.String searchString, java.math.BigDecimal invOrgId,
java.math.BigDecimal categorySetId, java.math.BigDecimal categoryId,
java.math.BigDecimal priceListId, int batchSize, int startIndex,
int resCount) throws FrameworkException, SQLException
```

List inventory items based on search criteria passed in as parameter.

Parameters:

searchString - Search string.

categorySetId - Inventory category set ID.

categoryId - Inventory category ID.

priceListId - Price list ID. This parameter is currently not supported.

batchSize - Batch size for the query.

startIndex - Start index indicating which row number to start retrieving data. -1 indicates the last rows.

resCount - The total number of objects to return. If -1, the count will be queried from the database.

Returns: ResultSet containing an array of InventoryItem objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

listAvailableServices(BigDecimal, BigDecimal, BigDecimal, String, Timestamp)

```
public static oracle.apps.qot.inventory.InventoryItem[]
listAvailableServices(java.math.BigDecimal invOrgId,
java.math.BigDecimal itemId, java.math.BigDecimal custAccountId,
java.lang.String itemRevision, java.sql.Timestamp requestDate)
throws FrameworkException, SQLException, InventoryException
```

Lists the available services for the specified inventory item, striped by customer, item revision, and request date

Parameters:

invOrgId - Inventory organization ID. This parameter is currently not supported.

itemId - Inventory Item ID.

custAccountId - Customer Account ID.

itemRevision - Item Revision.

requestDate - Request Date.

Returns: An array of InventoryItem objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

InventoryException - If an application error occurs.

listIncludedWarranties(BigDecimal, BigDecimal)

```
public static oracle.apps.qot.inventory.InventoryItem[]
listIncludedWarranties(java.math.BigDecimal invOrgId,
java.math.BigDecimal itemId)
throws FrameworkException, SQLException, InventoryException
Lists the included warranties for the specified inventory item.
```

Parameters:

invOrgId - Inventory organization ID.

itemId - Inventory item ID.

Returns: An array of InventoryItem objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

InventoryException - If an application error occurs.

listRelatedItems(BigDecimal, BigDecimal, BigDecimal, int, int, int)

```
public static oracle.apps.qot.util.QueryResultSet
listRelatedItems(java.math.BigDecimal invOrgId,
java.math.BigDecimal itemId, java.math.BigDecimal relnTypeId,
int batchSize, int startIndex, int resCount)
throws FrameworkException, SQLException
List the related inventory items for relationship type ID passed in as parameter. If
relnTypeId is null, list related inventory items for all relationship types.
Relationship type ID will be populated in the inventory items returned.
```

Parameters:

invOrgId - Inventory Organization ID.

itemId - Inventory Item ID.

relnTypeId - Relationship Item ID.

batchSize - Batch size for the query.

startIndex - Start index indicating which row number to start retrieving data. -1 indicates the last rows.

resCount - The total number of objects to return. If -1, the count will be queried from the database.

Returns: QueryResultSet containing an array of InventoryItem objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs

java.sql.SQLException - If a database error occurs.

load(BigDecimal, BigDecimal)

```
public static oracle.apps.qot.inventory.InventoryItem  
load(java.math.BigDecimal invOrgId, java.math.BigDecimal itemId)  
throws FrameworkException, SQLException
```

Return InventoryItem object corresponding to the inventory organization ID and inventory item ID passed in as parameter

Parameters:

invOrgId - Inventory organization ID.

itemId - Inventory item ID.

Returns: InventoryItem object with the appropriate information populated.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal, BigDecimal[])

```
public static oracle.apps.qot.inventory.InventoryItem[]  
load(java.math.BigDecimal invOrgId, java.math.BigDecimal[] itemIds)  
throws FrameworkException, SQLException
```

Return InventoryItem objects corresponding to the inventory organization ID and inventory item IDs passed in as parameter.

Parameters:

invOrgId - Inventory organization ID.

itemId - An array of inventory item IDs.

Returns: An array of InventoryItem objects with the appropriate information populated.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

loadUomCodes(InventoryItem[])

```
public static void
loadUomCodes(oracle.apps.qot.inventory.InventoryItem[] items)
throws FrameworkException, SQLException
```

For each inventory item passed in as parameter, loads the list of available unit of measure codes into the item. After calling this API, use the API `getUomCodes()` on each inventory item to retrieve the list of available unit of measure codes.

Parameters: items - An array of InventoryItem objects.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

setBomItemType(String)

```
public void setBomItemType(java.lang.String bomItemTypeMeaning)
```

Sets the value for bom item type meaning.

Parameters: bomItemTypeMeaning - BOM item type meaning.

setBomTypeCode(int)

```
public void setBomTypeCode(int bomItemTypeCode)
```

Sets the value for bom item type code.

Parameters: bomItemTypeCode - BOM item type code.

setInvOrgId(BigDecimal)

```
public void setInvOrgId(java.math.BigDecimal invOrgId)
```

Sets the value for the inventory organization ID

Parameters: invOrgId - Inventory organization ID.

setItemId(BigDecimal)

```
public void setItemId(java.math.BigDecimal itemId)
```

Sets the value for the item ID.

Parameters: itemId - Inventory item ID.

setItemType(String)

```
public void setItemType(java.lang.String itemTypeMeaning)
```

Sets the value for item type meaning.

Parameters: itemTypeMeaning - The translated meaning for item type.

setItemTypeCode(String)

```
public void setItemTypeCode(java.lang.String itemTypeCode)
```

Sets the value for item type code.

Parameters: itemTypeCode - Item type code.

setName(String)

```
public void setName(java.lang.String name)
```

Sets the value for name.

Parameters: name - Inventory item name (description).

setPartNumber(String)

```
public void setPartNumber(java.lang.String partNumber)
```

Sets the value for part number.

Parameters: partNumber - Inventory item part number (concatenated segments).

setPrimaryUom(String)

```
public void setPrimaryUom(java.lang.String primaryUomName)
```

Sets the value for primary UOM name.

Parameters: primaryUomName - Translated name for primary unit of measure.

setPrimaryUomCode(String)

```
public void setPrimaryUomCode(java.lang.String primaryUomCode)
```

Sets the value for primary UOM code.

Parameters: primaryUomCode - Primary unit of measure code.

setReInTypeId(BigDecimal)

```
public void setReInTypeId(java.math.BigDecimal relnTypeId)
```

Sets the value for relationship type ID.

Parameters: relnTypeId - Relationship type ID.

setReInTypeMeaning(String)

```
public void setReInTypeMeaning(java.lang.String relnTypeMeaning)
```

Sets the value for relationship type meaning.

Parameters: relnTypeMeaning - Translated meaning for relationship type.

setReturnable(boolean)

```
public void setReturnable(boolean isReturnable)
```

Sets the value for whether the item is returnable.

Parameters: isReturnable - Whether the item is returnable.

setServiceable(boolean)

```
public void setServiceable(boolean isServiceable)
```

Sets the value for whether the item is serviceable.

Parameters: isServiceable - Whether the item is serviceable.

setServiceItem(boolean)

```
public void setServiceItem(boolean isServiceItem)
```

Sets the value for whether the item is a service item.

Parameters: isServiceItem - Whether the item is a service item.

setShippable(boolean)

```
public void setShippable(boolean isShippable)
```

Sets the value for whether the item is shippable.

Parameters: isShippable - Whether the item is shippable.

setSrvDuration(BigDecimal)

```
public void setSrvDuration(java.math.BigDecimal durationQty)
```

Sets the value for service duration.

Parameters: durationQty - Service duration.

setSrvDurationPeriod(String)

```
public void setSrvDurationPeriod(java.lang.String  
durationPeriodMeaning)
```

Sets the value for service duration period meaning.

Parameters: durationPeriodMeaning - Translated meaning for service duration period.

setSrvDurationPeriodCode(String)

```
public void setSrvDurationPeriodCode(java.lang.String  
durationPeriodCode)
```

Sets the value for service duration period code.

Parameters: durationPeriodCode - Service duration period code.

setUomCodes(String[])

```
public void setUomCodes(java.lang.String[] uomCodes)
```

Sets the value for the UOM codes.

Parameters: uomCodes - An array of available unit of measure codes.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the inventory item object.

Overrides: toString in class Object

Returns: A String representation of the inventory item object.

3.4 Class InventoryUtil

```
java.lang.Object  
|  
+--oracle.apps.qot.inventory.InventoryUtil
```

```
public class InventoryUtil
```

InventoryUtil provides inventory-related utility methods. It includes methods to retrieve the default category set, translate unit of measure code, and validate quantities for inventory items.

Table 3–5 Inherited Member Summary

Methods inherited from class Object

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `toString()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

3.4.1 Fields for Class InventoryUtil

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

3.4.2 Constructors for Class InventoryUtil

InventoryUtil()

```
public InventoryUtil()  
Default constructor.
```

3.4.3 Methods for Class InventoryUtil

The following table is an index of the Class InventoryUtil methods:

Table 3–6 Methods for Class InventoryUtil

Method	Description
getDefaultCategorySet()	Returns the default category set ID. public static java.math.BigDecimal getDefaultCategorySet() throws FrameworkException, SQLException
getTranslatedUom(String)	Returns the translated unit of measure name in the current language for the uomCode passed in as parameter. public static java.lang.String getTranslatedUom (java.lang.String uomCode) throws FrameworkException, SQLException
validateQuantity(BigDecimal[], BigDecimal[], String[], String[])	Validate quantities for the inventory items, quantities, and unit of measure codes passed in as parameter. Throws an InventoryItem exception if the quantity validation fails. public static void validateQuantity (java.math.BigDecimal[] itemIds, java.math.BigDecimal[] invOrgIds, java.lang.String[] inputQty, java.lang.String[] uomCodes) throws FrameworkException, SQLException, InventoryException

getDefaultCategorySet()

public static java.math.BigDecimal **getDefaultCategorySet()**
throws FrameworkException, SQLException

Returns the default category set ID.

Returns: default category set ID.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

getTranslatedUom(String)

public static java.lang.String **getTranslatedUom**(java.lang.String uomCode) throws FrameworkException, SQLException

Returns the translated unit of measure name in the current language for the uomCode passed in as parameter.

Parameters: uomCode - Unit of Measure Code.

Throws:

oracle.apps.jtf.base.resources.FrameworkException -If a system error occurs.

java.sql.SQLException - If a database error occurs.

validateQuantity(BigDecimal[], BigDecimal[], String[], String[])

```
public static void validateQuantity(java.math.BigDecimal[] itemIds,  
java.math.BigDecimal[] invOrgIds, java.lang.String[] inputQty,  
java.lang.String[] uomCodes)
```

```
throws FrameworkException, SQLException, InventoryException
```

Validate quantities for the inventory items, quantities, and unit of measure codes passed in as parameter. Throws an InventoryItem exception if the quantity validation fails.

Parameters:

itemIds - Array of inventory Item IDs.

invOrgIds - Array of Inventory Organization IDs.

inputQty - Array of input quantities.

uomCodes - Array of Unit of Measure codes.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

InventoryException - If quantity validation fails or an application error occurs.

3.5 Class ItemCategory

```
java.lang.Object  
|  
+--oracle.apps.qot.inventory.ItemCategory
```

public class **ItemCategory**

ItemCategory is used to model an inventory item category and provides the method for retrieving the list of inventory item categories. The ItemCategory object contains

the following information for an inventory item category: category ID, category name.

Table 3-7 Inherited Member Summary

Methods inherited from class Object

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `toString()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

3.5.1 Fields for Class ItemCategory

RCS_ID

`public static final java.lang.String RCS_ID`

RCS_ID_RECORDED

`public static final boolean RCS_ID_RECORDED`

3.5.2 Constructors for Class ItemCategory

ItemCategory()

`public ItemCategory()`

Default constructor.

ItemCategory(BigDecimal, HashMap)

`public ItemCategory(java.math.BigDecimal _categoryId, com.sun.java.util.collections.HashMap _name)`

Constructor.

Parameters:

`_categoryId` - Item category ID.

`_name` - HashMap for the translated item category name. The key in the HashMap is language code, the value is translated item category name.

3.5.3 Methods for Class ItemCategory

The following table is an index of the Class ItemCategory methods:

Table 3–8 Methods for Class ItemCategory

Method	Description
<code>getCategoryId()</code>	Returns the category ID. <pre>public java.math.BigDecimal getCategoryId()</pre>
<code>getCatName()</code>	Returns the HashMap containing the translated item category names. The key in the HashMap is language code. The value is translated item category name. <pre>public com.sun.java.util.collections.HashMap getCatName()</pre>
<code>getName()</code>	Returns the translated category name for the current language. <pre>public java.lang.String getName()</pre>
<code>list(BigDecimal)</code>	Lists the categories for the category set ID passed in as parameter. <pre>public static oracle.apps.qot.inventory.ItemCategory[] list(java.math.BigDecimal categoryId) throws FrameworkException</pre>
<code>setCategoryId(BigDecimal)</code>	Sets the category ID. <pre>public void setCategoryId(java.math.BigDecimal _ categoryId)</pre>
<code>setName(HashMap)</code>	Sets the HashMap containing the translated item category names. The key in the HashMap is language code. The value is translated item category name. <pre>public void setName(com.sun.java.util.collections.HashMap _name)</pre>

getCategoryId()

```
public java.math.BigDecimal getCategoryId()
```

Returns the category ID.

Returns: Category ID.**getCatName()**

```
public com.sun.java.util.collections.HashMap getCatName()
```

Returns the HashMap containing the translated item category names. The key in the HashMap is language code. The value is translated item category name.

Returns: HashMap containing translated item category names.

getName()

```
public java.lang.String getName()
```

Returns the translated category name for the current language.

Returns: The translated item category name in the current language.

list(BigDecimal)

```
public static oracle.apps.qot.inventory.ItemCategory[]  
list(java.math.BigDecimal categorySetId) throws FrameworkException
```

Lists the categories for the category set ID passed in as parameter.

Parameters: categorySetId - Category Set ID.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setCategoryId(BigDecimal)

```
public void setCategoryId(java.math.BigDecimal _categoryId)
```

Sets the category ID.

Parameters: _categoryId - Item category ID.

setName(HashMap)

```
public void setName(com.sun.java.util.collections.HashMap _name)
```

Sets the HashMap containing the translated item category names. The key in the HashMap is language code. The value is translated item category name.

Parameters: _name - HashMap containing translated item category names. The key in the HashMap is language code. The value is translated item category name.

3.6 Class ShippingMethod

```
java.lang.Object
|
+--oracle.apps.qot.inventory.ShippingMethod
public class ShippingMethod
```

ShippingMethod is used to model a shipping method and provides the methods to list available shipping methods and retrieve basic information for shipping methods. The ShippingMethod object contains the following information for a shipping method: shipping method, name.

Table 3–9 Inherited Member Summary

Methods inherited from class Object
<code>equals(Object)</code> , <code>getClass()</code> , <code>hashCode()</code> , <code>notify()</code> , <code>notifyAll()</code> , <code>toString()</code> , <code>wait(long, int)</code> , <code>wait(long, int)</code> , <code>wait(long, int)</code>

3.6.1 Fields for Class ShippingMethod

```
RCS_ID
public static final java.lang.String RCS_ID

RCS_ID_RECORDED
public static final boolean RCS_ID_RECORDED
```

3.6.2 Constructors for Class ShippingMethod

```
ShippingMethod()
public ShippingMethod()
Constructor.
```

3.6.3 Methods for Class ShippingMethod

The following table is an index of the Class ShippingMethod methods:

Table 3–10 Methods for Class ShippingMethod

Method	Description
getCode()	Returns shipping method code. public java.lang.String getCode()
getName()	Returns the translated shipping method name for the current language. public java.lang.String getName()
getShipName()	Return the HashMap containing translated shipping method name. The key in the HashMap is language code, the value is translated shipping method name. public com.sun.java.util.collections.HashMap getShipName()
list(BigDecimal)	Lists the available shipping methods for the inventory organization ID passed in as parameter. public static oracle.apps.qot.inventory.ShippingMethod[] list (java.math.BigDecimal invOrgId) throws FrameworkException
load(BigDecimal, String)	Returns the ShippingMethod object corresponding to the inventory organization ID and shipping method code passed in as parameter. public static oracle.apps.qot.inventory.ShippingMethod load (java.math.BigDecimal invOrgId, java.lang.String shipMethodCode) throws FrameworkException
setName(HashMap)	Sets the HashMap containing translated shipping method name. public void setName (com.sun.java.util.collections.HashMap _name)
setShipMethodCode(String)	Sets the shipping method code. public void setShipMethodCode (java.lang.String _ shipMethodCode)

Table 3–10 Methods for Class ShippingMethod

Method	Description
toString()	Returns a String representation of the ShippingMethod object <code>public java.lang.String toString()</code>

getCode()

```
public java.lang.String getCode()
```

Returns the shipping method code.

Returns: The shipping method code.

getName()

```
public java.lang.String getName()
```

Returns the translated shipping method name for the current language.

Returns: The translated shipping method name for the current language.

getShipName()

```
public com.sun.java.util.collections.HashMap getShipName()
```

Return the HashMap containing translated shipping method name. The key in the HashMap is language code, the value is translated shipping method name.

Returns: HashMap containing translated shipping method name. The key in the HashMap is language code, the value is translated shipping method name.

list(BigDecimal)

```
public static oracle.apps.qot.inventory.ShippingMethod[]
```

```
list(java.math.BigDecimal invOrgId)
```

```
throws FrameworkException
```

Lists the available shipping methods for the inventory organization ID passed in as parameter.

Parameters: invOrgId - Inventory organization ID.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(BigDecimal, String)

```
public static oracle.apps.qot.inventory.ShippingMethod
```

```
load(java.math.BigDecimal invOrgId, java.lang.String shipMethodCode)
```

throws `FrameworkException`

Returns the `ShippingMethod` object corresponding to the inventory organization ID and shipping method code passed in as parameter.

Parameters:

`invOrgId` - Inventory organization ID.

`shipMethodCode` - Shipping method code.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

setName(HashMap)

public void **setName**(com.sun.java.util.collections.HashMap _name)
Sets the HashMap containing translated shipping method name

Parameters: `_name` - HashMap containing translated shipping method name. The key in the HashMap is language code, the value is translated shipping method name.

setShipMethodCode(String)

public void **setShipMethodCode**(java.lang.String _shipMethodCode)
Sets the shipping method code.

Parameters: `shipMethodCode` - Shipping method code.

toString()

public java.lang.String **toString**()
Returns a String representation of the `ShippingMethod` object.

Overrides: `toString` in class `Object`

Returns: A String representation of the `ShippingMethod` object.

oracle.apps.qot.perzquery

This section lists the Oracle Quoting Java APIs in the package oracle.apps.qot.perzquery.

4.1 Package oracle.apps.qot.perzquery

The table below lists a description for each class in this package:

Table 4–1 Class Summary for oracle.apps.qot.perzquery

Class	Definition
Class ColumnDefinition	The ColumnDefinition class represents the mapping information of a search column. The information consists of, search column(parameter) name, corresponding database column name, data type, length of the column, flag to indicate whether this column can be directly included in SQL WHERE clause, flag to indicate whether the search on this column is to be conducted in upper case letters and the position of the column in the SQL SELECT clause or resultset.
Class ConditionParameter	The ConditionParameter class represents the filter/search information of a search column. The mandatory information required in order to use this object is, search parameter(column) name, search parameter value(s) and search condition operator.
Class DataFormat	The DataFormat class provides various SQL data types as integer constants.
Class DisplayColumn	The DisplayColumn class represents the column/parameter that is to be retrieved from the database.

Table 4–1 Class Summary for oracle.apps.qot.perzquery

Class	Definition
Class ListQuery	The ListQuery is an abstract class that supports the ability to generically construct SQL query statements, specifying select columns, setting hints, specifying the from clause, adding filter conditions, adding WHERE , ORDER BY and GROUP BY clauses, and specifying range for the result set.
Class OrderByParameter	The OrderByParameter class represents the sort information of a column.
Class QueryBuilder	The QueryBuilder object is a utility class to build and execute the Listing SQL query using the various SQL clauses that are provided to this clause.
Class SavedSearch	SavedSearch class represents the search information that is stored in the database and is used to save, delete and load the search information from database.
Class SearchInfo	The SearchInfo class represents all the filter (search) information entered by a user for one search page.
Exceptions	
Class QueryBuilderException	The QueryBuilderException is thrown if any errors encountered while building or executing the SQL query statement.
Class SavedSearchException	The SavedSearchException is thrown if any errors encountered while loading, saving or deleting the search criteria from database.

4.2 Class ColumnDefinition

```
java.lang.Object
|
+--oracle.apps.qot.perzquery.ColumnDefinition
```

public class **ColumnDefinition**

The ColumnDefinition class represents the mapping information of a search column. The information consists of, search column(parameter) name, corresponding database column name, data type, length of the column, flag to indicate whether this column can be directly included in SQL WHERE clause, flag to indicate whether the search on this column is to be conducted in upper case letters and the position of the column in the SQL SELECT clause or resultset. This

object is used while preparing the SQL statement and also while iterating the returned resultset.

Table 4–2 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

4.2.1 Fields for Class ColumnDefinition

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to oracle.apps.fnd.common.VersionInfo.recordClassVersion.

4.2.2 Constructors for Class ColumnDefinition

ColumnDefinition(String, String, int)

```
public ColumnDefinition(java.lang.String appColName,  
java.lang.String dbColName, int type)
```

Constructs a ColumnDefinition object with search column information.

Parameters:

appColName - Application search column name.

dbColName - Database column name.

type - Data type of column.

ColumnDefinition(String, String, int, boolean)

```
public ColumnDefinition(java.lang.String appColName,  
java.lang.String dbColName, int type, boolean whereColumn)
```

Constructs a ColumnDefinition object with search column information.

Parameters:

appColName - Application search column name.

dbColName - Database column name.

type - Data type of column.

whereColumn - Indicates whether this column should be included directly in SQL WHERE clause. Default value is true.

ColumnDefinition(String, String, int, boolean, boolean)

```
public ColumnDefinition(java.lang.String appColName,  
java.lang.String dbColName, int type, boolean upperColumn,  
boolean whereColumn)
```

Constructs a ColumnDefinition object with search column information.

Parameters:

appColName - Application search column name.

dbColName - Database column name.

type - Data type of column.

upperColumn - Indicates whether this column should be filtered in upper case letters. Default value is false.

whereColumn - Indicates whether this column should be included directly in SQL WHERE clause. Default value is true.

ColumnDefinition(String, String, int, int)

```
public ColumnDefinition(java.lang.String appColName,  
java.lang.String dbColName, int type, int length)
```

Constructs a ColumnDefinition object with search column information.

Parameters:

appColName - Application search column name.

dbColName - Database column name.

type - Data type of column.

length - Length of column.

ColumnDefinition(String, String, int, int, boolean)

```
public ColumnDefinition(java.lang.String appColName,  
java.lang.String dbColName, int type, int length, boolean upperColumn)
```

Constructs a ColumnDefinition object with search column information.

Parameters:

appColName - Application search column name.

dbColName - Database column name.

type - Data type of column.

length - Length of column.

upperColumn - Indicates whether this column should be filtered in upper case letters. Default value is false.

ColumnDefinition(String, String, int, int, boolean, boolean)

```
public ColumnDefinition(java.lang.String appColName,  
java.lang.String dbColName, int type, int length, boolean upperColumn,  
boolean whereColumn)
```

Constructs a ColumnDefinition object with search column information.

Parameters:

appColName - Application search column name.

dbColName - Database column name.

type - Data type of column.

length - Length of column.

upperColumn - Indicates whether this column should be filtered in upper case letters. Default value is false.

whereColumn - Indicates whether this column should be included directly in SQL WHERE clause. Default value is true.

4.2.3 Methods for Class ColumnDefinition

The following table is an index of the Class ColumnDefinition methods:

Table 4–3 Methods for Class ColumnDefinition

Method	Description
<code>getAppColumnName()</code>	Returns the application search column name. <code>public java.lang.String getAppColumnName()</code>
<code>getDbColumnName()</code>	Returns the database column name. <code>public java.lang.String getDbColumnName()</code>
<code>getLength()</code>	Returns the length of database column. <code>public int getLength()</code>
<code>getResultSetIndex()</code>	Returns the position of database column in SELECT clause. <code>public int getResultSetIndex()</code>
<code>getType()</code>	Returns the data type of database column. <code>public int getType()</code>
<code>getUpperColumn()</code>	Returns a boolean to indicate whether this column is filtered in upper or lower case letters. <code>public boolean getUpperColumn()</code>
<code>getWhereColumn()</code>	Returns a boolean to indicate whether the database column is directly included in SQL WHERE clause. <code>public boolean getWhereColumn()</code>
<code>setAppColumnName(String)</code>	Sets the application search column name. <code>public void setAppColumnName(java.lang.String appColumnName)</code>
<code>setDbColumnName(String)</code>	Sets the database column name. <code>public void setDbColumnName(java.lang.String dbColumnName)</code>
<code>setLength(int)</code>	Sets the length of database column. <code>public void setLength(int length)</code>
<code>setResultSetIndex(int)</code>	Sets the position of database column in SELECT clause. <code>public void setResultSetIndex(int resultSetIndex)</code>
<code>setType(int)</code>	Sets the data type of database column. <code>public void setType(int type)</code>

Table 4–3 Methods for Class ColumnDefinition

Method	Description
setUpperColumn(boolean)	Sets whether this column should be filtered in upper case letters. public void setUpperColumn (boolean upperColumn)
setWhereColumn(boolean)	Sets whether the database column should be directly included in SQL WHERE clause. public void setWhereColumn (boolean whereColumn)
toString()	Returns a String representation of ColumnDefintion object. public java.lang.String toString ()

getAppColumnName()

```
public java.lang.String getAppColumnName()
```

Returns the application search column name.

Returns: Application search column name.

getDbColumnName()

```
public java.lang.String getDbColumnName()
```

Returns the database column name.

Returns: Database column name.

getLength()

```
public int getLength()
```

Returns the length of database column.

Returns: Length of database column.

getResultSetIndex()

```
public int getResultSetIndex()
```

Returns the position of database column in SELECT clause.

Returns: Position of column in resultset.

getType()

public int `getType()`
Returns the data type of database column.

Returns: Data type of database column.

getUpperColumn()

public boolean `getUpperColumn()`
Returns a boolean to indicate whether this column is filtered in upper or lower case letters.

Returns: The boolean value to indicate whether this column is filtered in upper case letters.

getWhereColumn()

public boolean `getWhereColumn()`
Returns a boolean to indicate whether the database column is directly included in SQL WHERE clause.

Returns: The boolean value to indicate whether this column is directly included in SQL WHERE clause.

setAppColumnName(String)

public void `setAppColumnName(java.lang.String appColumnName)`
Sets the application search column name.

Parameters: appColumnName - Application search column name.

setDbColumnName(String)

public void `setDbColumnName(java.lang.String dbColumnName)`
Sets the database column name.

Parameters: dbColumnName - Database column name.

setLength(int)

public void `setLength(int length)`
Sets the length of database column.

Parameters: length - Length of database column.

setResultSetIndex(int)

public void `setResultSetIndex(int resultSetIndex)`

Sets the position of database column in SELECT clause.

Parameters: resultSetIndex - Position of column in resultset.

setType(int)

```
public void setType(int type)
```

Sets the data type of database column.

Parameters: type - Data type of database column.

setUpperColumn(boolean)

```
public void setUpperColumn(boolean upperColumn)
```

Sets whether this column should be filtered in upper case letters.

Parameters: upperColumn - Indicates whether this column should be filtered in upper case letters. Default value is false.

setWhereColumn(boolean)

```
public void setWhereColumn(boolean whereColumn)
```

Sets whether the database column should be directly included in SQL WHERE clause.

Parameters: whereColumn - Indicates whether this column should be directly included in SQL WHERE clause. Default value is true.

toString()

```
public java.lang.String toString()
```

Returns a String representation of ColumnDefintion object.

Overrides: toString in class Object

Returns: A String representation of ColumnDefinition object.

4.3 Class ConditionParameter

```
java.lang.Object
|
+--oracle.apps.qot.perzquery.ConditionParameter
```

public class **ConditionParameter**

The ConditionParameter class represents the filter/search information of a search column. The mandatory information required to use this object is: search parameter(column) name, search parameter value(s,) and search condition operator. This object is used while preparing the SQL statement's WHERE clause and also while saving the search criteria to the database.

Table 4–4 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

4.3.1 Fields for Class ConditionParameter

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to oracle.apps.fnd.common.VersionInfo.recordClassVersion.

4.3.2 Constructors for Class ConditionParameter

ConditionParameter()

```
public ConditionParameter()
```

Default Constructor.

ConditionParameter(String, ArrayList, String)

```
public ConditionParameter(java.lang.String name,
    com.sun.java.util.collections.ArrayList values,
    java.lang.String condition)
```

Constructs ConditionParameter object with filter/search information.

Parameters:

name - Name of the search parameter.

values - Values of the search parameter. Search value(s) must be of type String. Date search values must be in Timestamp string format (Timestamp.toString()).

condition - Filter operator of the search parameter. Default value is =.

4.3.3 Methods for Class ConditionParameter

The following table is an index of the Class ConditionParameter methods:

Table 4–5 Methods for Class ConditionParameter

Method	Description
getCondition()	Returns the filter condition/operator of the search parameter. public java.lang.String getCondition()
getName()	Returns the name of the search parameter. public java.lang.String getName()
getValues()	Returns the filter values of the search parameter. public com.sun.java.util.collections.ArrayList getValues()
setCondition(string)	Sets the filter condition/operator for the search parameter. public void setCondition (java.lang.String condition)
setName(string)	Sets the name of the search parameter. public void setName (java.lang.String name)
setValues(ArrayList)	Sets the filter values of the search parameter. public void setValues (com.sun.java.util.collections.ArrayL ist values)

Table 4–5 Methods for Class ConditionParameter

Method	Description
toString()	Converts the ConditionParameter object into string representation. <code>public java.lang.String toString()</code>

getCondition()

```
public java.lang.String getCondition()
```

Returns the filter condition/operator of the search parameter.

Returns: Filter condition/operator of the search parameter.

getName()

```
public java.lang.String getName()
```

Returns the name of the search parameter.

Returns: Name of the search parameter.

getValues()

```
public com.sun.java.util.collections.ArrayList getValues()
```

Returns the filter values of the search parameter.

Returns: Filter values of the search parameter.

setCondition(String)

```
public void setCondition(java.lang.String condition)
```

Sets the filter condition/operator for the search parameter.

Parameters: condition - Filter condition/operator for the search parameter. Default value is “=”.

setName(String)

```
public void setName(java.lang.String name)
```

Sets the name of the search parameter.

Parameters: name - Name of the search parameter.

setValues(ArrayList)

```
public void setValues(com.sun.java.util.collections.ArrayList values)
```

Sets the filter values of the search parameter.

Parameters: values - Filter values of the search parameter.

toString()

```
public java.lang.String toString()
```

Converts the ConditionParameter object into a String representation.

Overrides: toString in class Object

Returns: A String representation of this object.

4.4 Class DataFormat

```
java.lang.Object
|
+--oracle.apps.qot.perzquery.DataFormat
```

public class DataFormat

The DataFormat class provides various SQL data types as integer constants. Typical data types that are currently supported are: VARCHAR, TIMESTAMP, and BIGDECIMAL. These constants are used while setting the search parameter/column information and also while preparing the SQL statement.

Table 4–6 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(long, int), wait(long, int), wait(long, int)

4.4.1 Fields for Class DataFormat

BIGDECIMAL

```
public static final int BIGDECIMAL
```

Represents java.sql.Types.NUMERIC.

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to `oracle.apps.fnd.common.VersionInfo.recordClassVersion`.

STRING

```
public static final int STRING
```

Represents `java.sql.Types.VARCHAR`.

TIMESTAMP

```
public static final int TIMESTAMP
```

Represents `java.sql.Types.TIMESTAMP`.

4.4.2 Constructors for Class DataFormat

DataFormat()

```
public DataFormat()
```

4.5 Class DisplayColumn

```
java.lang.Object
|
+--oracle.apps.qot.perzquery.DisplayColumn
public class DisplayColumn
```

The DisplayColumn class represents the column/parameter that is to be retrieved from the database. This object is used while preparing the SQL statement, iterating the resultset, and while saving the search criteria to the database.

Table 4-7 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

4.5.1 Fields for Class DisplayColumn

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to oracle.apps.fnd.common.VersionInfo.recordClassVersion.

4.5.2 Constructors for Class DisplayColumn

DisplayColumn()

```
public DisplayColumn()
```

Default Constructor.

DisplayColumn(String)

```
public DisplayColumn(java.lang.String name)
```

Constructs the DisplayColumn object with the column name.

Parameters: name - Name of the column to be retrieved from the database.

4.5.3 Methods for Class DisplayColumn

The following table is an index of the Class DisplayColumn methods:

Table 4–8 *Methods for Class DisplayColumn*

Method	Description
getName()	Returns the name of the column that is to be retrieved from database. <code>public java.lang.String getName()</code>
setName(String)	Sets the name of the column that is to be retrieved from database. <code>public void setName(java.lang.String name)</code>
toString()	Converts the DisplayColumn object into a String representation. <code>public java.lang.String toString()</code>

getName()

```
public java.lang.String getName()
```

Returns the name of the column that is to be retrieved from database.

Returns: Name of the column that is to be retrieved from database.

setName(String)

```
public void setName(java.lang.String name)
```

Sets the name of the column that is to be retrieved from database.

Parameters: name - Name of the column that is to be retrieved from database.

toString()

```
public java.lang.String toString()
```

Converts the DisplayColumn object into a String representation.

Overrides: toString in class Object

Returns: A String representation of DisplayColumn object.

4.6 Class ListQuery

```
java.lang.Object
```

```
|
```

```
+--oracle.apps.qot.perzquery.ListQuery
```

public abstract class **ListQuery**

ListQuery is an abstract class that supports the ability to generically construct SQL query statements by specifying select columns, setting hints, specifying the from clause, adding filter conditions, adding “WHERE”, “ORDER BY” and “GROUP BY” clauses, and specifying range for the result set. Individual implementations need to extend the ListQuery and provide the mapping information of search parameters/columns with their corresponding database column names and their datatypes.

Customizations to the methods for constructing the query and post-query processing can be performed by overriding the appropriate methods provided in this class. Implementation code has been provided in some methods for preparing the SQL query with the search values from SavedSearch object. If any of the methods of this class are overridden in the deriving class, care should be taken to provide the equivalent functionality in the deriving class.

Table 4–9 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(long, int), wait(long, int), wait(long, int)

4.6.1 Fields for Class ListQuery

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to oracle.apps.fnd.common.VersionInfo.recordClassVersion.

4.6.2 Constructors for Class ListQuery

ListQuery()

```
public ListQuery()
```

4.6.3 Methods for Class ListQuery

The following table is an index of the Class DisplayColumn methods:

Table 4–10 *Methods for Class DisplayColumn*

Method	Description
<code>addFilter(String, String, ArrayList)</code>	<p>Adds a column against which search is conducted. This column should be mapped to the database column before this method is invoked. This column is added to the SQL WHERE clause. These columns are typically set from SavedSearch object. The values from the SavedSearch are parsed and converted into the appropriate type (eg. BigDecimal, Timestamp etc.) based on the type specified when the column was mapped. The filter information is stored in filterColumns, filterOperators and filterValues. The column definition of the search column is used to determine the datatype of the search value.</p> <pre>public void addFilter(java.lang.String colName, java.lang.String op, com.sun.java.util.collections.ArrayList values)</pre>
<code>addFilter(String, String, Object)</code>	<p>Adds a column against which search is conducted. This column should be mapped to the database column before this method is invoked. This column is added to the SQL WHERE clause. These columns are typically set from SavedSearch object. The values from the SavedSearch are parsed and converted into the appropriate type (eg. BigDecimal, Timestamp etc.) based on the type specified when the column was mapped. The filter information is stored in filterColumns, filterOperators and filterValues. The column definition of the search column is used to determine the datatype of the search value.</p> <pre>public void addFilter(java.lang.String colName, java.lang.String op, java.lang.Object value)</pre>

Table 4–10 Methods for Class DisplayColumn

Method	Description
<code>addFilters(ConditionParameter[])</code>	<p>Adds columns against which search is to be conducted. These columns should be mapped to the database column before this method is invoked. These columns are added to the SQL WHERE clause. These columns are typically set from SavedSearch object. The values from the SavedSearch are parsed and converted into the appropriate type (eg. BigDecimal, Timestamp etc.) based on the type specified when the column was mapped. The filter information is stored in filterColumns, filterOperators and filterValues. The column definition of the search column is used to determine the datatype of the search value.</p> <pre>public void addFilters(oracle.apps.qot.perzquery.ConditionParameter[] conditionParameters)</pre>
<code>addOrderBy(OrderByParameter[])</code>	<p>Adds columns on which sorting is to be done. These column should be mapped to the database column before this method is invoked. These column is added to the SQL ORDER BY clause. These columns are typically set from SavedSearch object. The sort information is stored in orderByColumns and orderByQualifiers.</p> <pre>public void addOrderBy(oracle.apps.qot.perzquery.OrderByParameter[] orderByParameters)</pre>
<code>addOrderBy(String, String)</code>	<p>Adds a column on which sorting is to be done. This column should be mapped to the database column before this method is invoked. This column is added to the SQL ORDER BY clause. These columns are typically set from SavedSearch object. The sort information is stored in orderByColumns and orderByQualifiers.</p> <pre>public void addOrderBy(java.lang.String colName, java.lang.String qualifier)</pre>
<code>addSelectColumn(String)</code>	<p>Adds a column that is to be retrieved from the database. This column should be mapped to the database column before this method is invoked. This column is added to selectColumns and the columns that are not mapped will not be stored in selectColumns. This column is used for building the SQL SELECT clause and iterating ResultSet. These columns are typically set from SavedSearch object.</p> <pre>public void addSelectColumn(java.lang.String colName)</pre>

Table 4–10 Methods for Class DisplayColumn

Method	Description
<code>addSelectColumns(String[])</code>	<p>Adds a column that is to be retrieved from the database. This column should be mapped to the database column before this method is invoked. This column is used for building the SQL SELECT clause and iterating ResultSet. These columns are typically set from SavedSearch object.</p> <pre>public void addSelectColumns(java.lang.String[] colNames)</pre>
<code>clearColumnMapping()</code>	<p>Clears the column mapping information. This method is called from the implementation class, once before initiating new column mapping.</p> <pre>public void clearColumnMapping()</pre>
<code>closeStatement()</code>	<p>Closes PreparedStatement object of QueryBuilder. Is typically invoked from the implementing class of ListQuery to close the PreparedStatement.</p> <pre>public void closeStatement()</pre>
<code>getBatchSize()</code>	<p>Returns the batch size of the query.</p> <pre>public int getBatchSize()</pre>
<code>getResultCount()</code>	<p>Returns the total number of rows returned from the query.</p> <pre>public int getResultCount()</pre>
<code>getStartIndex()</code>	<p>Returns the start index (row number to start retrieving data).</p> <pre>public int getStartIndex()</pre>
<code>initColumnMapping()</code>	<p>Method to provide mapping information for the search columns(parameters) with their corresponding database column names. Deriving class MUST provide implementation. Provide mapping information for those search parameters that have the probability(chance) of getting directly included in the SELECT, WHERE (not in JOIN, but BIND) and ORDER BY clauses of SQL statement. appColumnName should not be duplicate. There may be some search parameters which do not associate directly to any data base column, but are still part of search criteria. For these search parameters, specify dbColumnName as “NULL” and whereColumn as false.</p> <pre>public abstract void initColumnMapping()</pre>
<code>setBatchSize(int)</code>	<p>Sets the batch size for the query. Typically set from SavedSearch object or the implementing class of ListQuery.</p> <pre>public void setBatchSize(int batchSize)</pre>

Table 4–10 Methods for Class DisplayColumn

Method	Description
setResultCount(int)	Sets the total number of rows returned from the query. Typically set from SavedSearch object or the implementing class of ListQuery. public void setResultCount (int resultCount)
setStartIndex(int)	Sets the start index (row number to start retrieving data). Typically set from SavedSearch object or the implementing class of ListQuery. public void setStartIndex (int startIndex)

addFilter(String, String, ArrayList)

```
public void addFilter(java.lang.String colName, java.lang.String op,
com.sun.java.util.collections.ArrayList values)
```

Adds a column against which search is conducted. This column should be mapped to the database column before this method is invoked. This column is added to the SQL WHERE clause. These columns are typically set from SavedSearch object. The values from the SavedSearch are parsed and converted into the appropriate type (eg. BigDecimal, Timestamp etc.) based on the type specified when the column was mapped. The filter information is stored in `filterColumns`, `filterOperators` and `filterValues`. The column definition of the search column is used to determine the datatype of the search value.

Parameters:

`colName` - Application search column name.

`op` - Condition/Operator for the search column.

`values` - Search values of type String that are to be bind in WHERE clause.

addFilter(String, String, Object)

```
public void addFilter(java.lang.String colName, java.lang.String op,
java.lang.Object value)
```

Adds a column against which search is conducted. This column should be mapped to the database column before this method is invoked. This column is added to the SQL WHERE clause. These columns are typically set from SavedSearch object. The values from the SavedSearch are parsed and converted into the appropriate type (eg. BigDecimal, Timestamp etc.) based on the type specified when the column was mapped. The filter information is stored in `filterColumns`, `filterOperators` and

filterValues. The column definition of the search column is used to determine the datatype of the search value.

Parameters:

colName - Application search column name.

op - Operator/Condition for the search column.

values - Search values of type String that are to be bind in WHERE clause.

addFilters(ConditionParameter[])

```
public void addFilters(oracle.apps.qot.perzquery.ConditionParameter[]  
conditionParameters)
```

Adds columns against which search is to be conducted. These columns should be mapped to the database column before this method is invoked. These columns are added to the SQL WHERE clause. These columns are typically set from SavedSearch object. The values from the SavedSearch are parsed and converted into the appropriate type (eg. BigDecimal, Timestamp etc.) based on the type specified when the column was mapped. The filter information is stored in filterColumns, filterOperators and filterValues. The column definition of the search column is used to determine the datatype of the search value.

Parameters: conditionParameters - ConditionParameter objects with search column name, value and operator information.

addOrderBy(OrderByParameter[])

```
public void addOrderBy(oracle.apps.qot.perzquery.OrderByParameter[]  
orderByParameters)
```

Adds columns on which sorting is to be done. These column should be mapped to the database column before this method is invoked. These column is added to the SQL ORDER BY clause. These columns are typically set from SavedSearch object. The sort information is stored in orderbyColumns and orderbyQualifiers.

Parameters: orderByParameters - Array of OrderByParameters objects.

addOrderBy(String, String)

```
public void addOrderBy(java.lang.String colName,  
java.lang.String qualifier)
```

Adds a column on which sorting is to be done. This column should be mapped to the database column before this method is invoked. This column is added to the SQL ORDER BY clause. These columns are typically set from SavedSearch object. The sort information is stored in orderbyColumns and orderbyQualifiers.

Parameters:

colName - Application sort column name.

qualifier - Order by Qualifier for sort column (“ASC” or “DESC”).

addSelectColumn(String)

```
public void addSelectColumn(java.lang.String colName)
```

Adds a column that is to be retrieved from the database. This column should be mapped to the database column before this method is invoked. This column is added to `selectColumns` and the columns that are not mapped will not be stored in `selectColumns`. This column is used for building the SQL SELECT clause and iterating `ResultSet`. These columns are typically set from `SavedSearch` object.

Parameters: colName - Application search column name that is to be retrieved from the database.

addSelectColumns(String[])

```
public void addSelectColumns(java.lang.String[] colNames)
```

Adds a column that is to be retrieved from the database. This column should be mapped to the database column before this method is invoked. This column is used for building the SQL SELECT clause and iterating `ResultSet`. These columns are typically set from `SavedSearch` object.

Parameters: colNames[] - Application search column names that are to be retrieved from the database.

clearColumnMapping()

```
public void clearColumnMapping()
```

Clears the column mapping information. This method is called from the implementation class, once before initiating new column mapping.

closeStatement()

```
public void closeStatement()
```

Closes `PreparedStatement` object of `QueryBuilder`. Is typically invoked from the implementing class of `ListQuery` to close the `PreparedStatement`.

getBatchSize()

```
public int getBatchSize()
```

Returns the batch size of the query.

Returns: Batch size of the query.

getResultCount()

```
public int getResultCount()
```

Returns the total number of rows returned from the query.

Returns: Total number of rows returned.

getStartIndex()

```
public int getStartIndex()
```

Returns the start index (row number to start retrieving data).

Returns: Start index.

initColumnMapping()

```
public abstract void initColumnMapping()
```

Method to provide mapping information for the search columns(parameters) with their corresponding database column names. Deriving class MUST provide implementation. Provide mapping information for those search parameters that have the probability(chance) of getting directly included in the SELECT, WHERE (not in JOIN, but BIND) and ORDER BY clauses of SQL statement.

appColumnName should not be duplicate. There may be some search parameters which do not associate directly to any data base column, but are still part of search criteria. For these search parameters, specify dbColumnName as "NULL" and whereColumn as false.

setBatchSize(int)

```
public void setBatchSize(int batchSize)
```

Sets the batch size for the query. Typically set from SavedSearch object or the implementing class of ListQuery.

Parameters: batchSize - Batch size for the query.

setResultCount(int)

```
public void setResultCount(int resultCount)
```

Sets the total number of rows returned from the query. Typically set from SavedSearch object or the implementing class of ListQuery.

Parameters: resultCount - Total number of rows returned. -1 indicates that the count should be queried from database.

setStartIndex(int)

```
public void setStartIndex(int startIndex)
```

Sets the start index (row number to start retrieving data). Typically set from SavedSearch object or the implementing class of ListQuery.

Parameters: startIndex - Indicates which row number to start retrieving data. -1 indicates the last rows.

4.7 Class OrderByParameter

```
java.lang.Object
|
+--oracle.apps.qot.perzquery.OrderByParameter
public class OrderByParameter
```

The OrderByParameter class represents the sort information of a column. The information consists of, sort column(parameter) name and sort qualifier. This object is used while preparing the SQL WHERE-JOIN and ORDER BY clauses.

Table 4–11 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

4.7.1 Fields for Class OrderByParameter

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to oracle.apps.fnd.common.VersionInfo.recordClassVersion.

4.7.2 Constructors for Class OrderByParameter

OrderByParameter()

```
public OrderByParameter()
```

Constructs OrderByParameter object.

OrderByParameter(String, Boolean)

```
public OrderByParameter(java.lang.String name, java.lang.Boolean ascdng)
```

Constructs OrderByParameter object with sort information.

Parameters:

name - Name of the column to sort.

ascdng - Sort Order. Boolean.TRUE for ascending order and Boolean.FALSE for descending order. Default value is Boolean.TRUE.

4.7.3 Methods for Class OrderByParameter

The following table is an index of the Class OrderByParameter methods:

Table 4–12 *Methods for Class OrderByParameter*

Method	Description
<code>getAscng()</code>	Returns the sort order of the column to be sorted. <code>public java.lang.Boolean getAscng()</code>
<code>getName()</code>	Returns the name of the column to be sorted. <code>public java.lang.String getName()</code>
<code>setAscng(Boolean)</code>	Sets sort order for the column to be sorted. <code>public void setAscng(java.lang.Boolean ascng)</code>
<code>setName(String)</code>	Sets the name of the column to be sorted. <code>public void setName(java.lang.String name)</code>
<code>toString()</code>	Converts OrderByParameter object into a String representation. <code>public java.lang.String toString()</code>

getAscng()

```
public java.lang.Boolean getAscng()
```

Returns the sort order of the column to be sorted.

Returns: Sort order of the column to be sorted. Boolean.TRUE for ascending and Boolean.FALSE for descending order. Default value is Boolean.TRUE.

getName()

```
public java.lang.String getName()
```

Returns the name of the column to be sorted.

Returns: Name of the column to be sorted.

setAscng(Boolean)

```
public void setAscng(java.lang.Boolean ascng)
```

Sets sort order for the column to be sorted.

Parameters: ascndng - Sort Order for the column to be sorted. Boolean.TRUE for ascending and Boolean.FALSE for descending order. Default value is Boolean.TRUE.

setName(String)

```
public void setName(java.lang.String name)
```

Sets the name of the column to be sorted.

Parameters: name - Name of the column to be sorted.

toString()

```
public java.lang.String toString()
```

Converts OrderByParameter object into a String representation.

Overrides: toString in class Object

Returns: A String representation of OrderByParameter object.

4.8 Class QueryBuilder

```
java.lang.Object  
|  
+--oracle.apps.got.perzquery.QueryBuilder
```

public class QueryBuilder

The QueryBuilder object is a utility class to build and execute the Listing SQL query using the various SQL clauses that are provided to this class. The minimum information that is required for QueryBuilder class to execute any listing SQL query is SELECT clause, FROM clause, resultset start index, result count and batch size. It implements methods to add different parts of the query (SELECT clause, FROM clause, WHERE clause etc) to the sql query. It also defines the column types, binds the variables before executing the SQL query.

Table 4–13 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(long, int), wait(long, int), wait(long, int)

4.8.1 Fields for Class QueryBuilder

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to `oracle.apps.fnd.common.VersionInfo.recordClassVersion`.

4.8.2 Constructors for Class QueryBuilder

QueryBuilder()

```
public QueryBuilder()
```

Constructs the QueryBuilder object.

4.8.3 Methods for Class QueryBuilder

The following table is an index of the Class QueryBuilder methods:

Table 4–14 *Methods for Class OrderByParameter*

Method	Description
<code>addSelectClause(String)</code>	<p>Adds individual SELECT attributes. This method is called multiple times, once for each column that is to be retrieved from the database.</p> <pre>public void addSelectClause(java.lang.String attrClause)</pre>
<code>addSelectClause(String, ArrayList)</code>	<p>Adds individual SELECT attributes along with bind variables. This method is called multiple times, once for each column that is to be retrieved from the database.</p> <pre>public void addSelectClause(java.lang.String attrClause, com.sun.java.util.collections.ArrayList bi ndVars)</pre>

Table 4–14 Methods for Class OrderByParameter

Method	Description
addSelectClauseBindVars(Array List)	<p>Sets the bind variables for the SELECT clause. Can be called multiple times. Null values are not allowed for binding.</p> <pre>public void addSelectClauseBindVars(com.sun.java.util. collections.ArrayList bindVars)</pre>
addWhereClause(String, String, Object)	<p>Adds a SQL WHERE clause to the query along with bind variables. Null values are not allowed as bind variables. Binding is done based on the object type of the bind variable value passed. The query will be constructed using “?” and the passed value is bound. This method can be called multiple times and all where clauses are ANDed together.</p> <pre>public void addWhereClause(java.lang.String attrClause, java.lang.String operator, java.lang.Object bindValue)</pre>
addWhereClause(String, String, String)	<p>Adds a SQL WHERE clause to the query. This can be invoked multiple times, and all the WHERE clauses are ANDed together.</p> <pre>public void addWhereClause(java.lang.String attrClause, java.lang.String operator, java.lang.String valueClause)</pre>
addWhereClause(String, String, String, ArrayList)	<p>Adds a SQL WHERE clause to the query along with bind variables. Null values are not allowed as bind variables. Binding is done based on the object type of the bind variable value passed. This method can be called multiple times and all where clauses are ANDed together.</p> <pre>public void addWhereClause(java.lang.String attrClause, java.lang.String operator, java.lang.String valueClause, com.sun.java.util.collections.ArrayList bi ndVars)</pre>

Table 4–14 Methods for Class OrderByParameter

Method	Description
<code>buildDebugQueryStatement()</code>	<p>Debugging Method that constructs the entire query using the different SQL clauses. The query is constructed in the following way: SELECT clause + FROM clause + WHERE clause + extra WHERE clause + GROUP BY clause + HAVING clause + ORDER BY clause. Only used for de-bugging and testing purposes.</p> <pre>public java.lang.String buildDebugQueryStatement() throws Exception</pre>
<code>closeStatement()</code>	<p>Closes PreparedStatement object of QueryBuilder class. Typically invoked from the containing classes of QueryBuilder object to close/release the PreparedStatement.</p> <pre>public void closeStatement()</pre>
<code>executeQuery(OracleConnection)</code>	<p>Creates a SQL PreparedStatement, binds variables to it and executes the SQL query.</p> <pre>public com.sun.java.util.collections.HashMap executeQuery(oracle.jdbc.driver.OracleConnection conn) throws QueryBuilderException, SQLException</pre>
<code>getBatchSize()</code>	<p>Returns the batch size of the query.</p> <pre>public int getBatchSize()</pre>
<code>getDebugQueryStatement()</code>	<p>Returns the entire constructed SQL statement along with the bind variables. Is mainly used for debugging/logging purposes.</p> <pre>public java.lang.String getDebugQueryStatement()</pre>
<code>getQueryStatement()</code>	<p>Returns the entire constructed SQL statement.</p> <pre>public java.lang.String getQueryStatement()</pre>
<code>getResultCount()</code>	<p>Returns the total number of rows returned from the query.</p> <pre>public int getResultCount()</pre>
<code>getStartIndex()</code>	<p>Returns the start index of the query.</p> <pre>public int getStartIndex()</pre>

Table 4–14 *Methods for Class OrderByParameter*

Method	Description
setBatchSize(int)	Sets the batch size for the query. public void setBatchSize (int batchSize)
setExtraWhereSize(String)	Sets an extra SQL WHERE clause to the query. Used for setting WHERE clause involving ORs. This clause is ANDed to the existing WHERE clause if it is present. Multiple calls replaces existing clause. public void setExtraWhereClause (java.lang.String extraWhereClause)
setExtraWhereClause(String, ArrayList)	Sets an extra SQL WHERE clause to the query. Used for setting where clause involving ORs. This clause is ANDed to the existing WHERE clause if it is present. Multiple calls replaces the existing clause. Null values are not allowed as bind variables. Binding is done based on the object type of the bind variable value passed. public void setExtraWhereClause (java.lang.String extraWhereClause, com.sun.java.util.collections.ArrayList bindVars)
setFromClause(String)	Sets the SQL FROM clause for the query. Multiple calls to this method replaces the existing FROM clause. public void setFromClause (java.lang.String fromClause)
setFromClause(String, ArrayList)	Sets the SQL FROM clause for the query along with the bind variables. Multiple calls to this method replaces the existing FROM clause. public void setFromClause (java.lang.String fromClause, com.sun.java.util.collections.ArrayList bindVars)
setGroupByClause(String)	Sets the GROUP BY clause for the SQL query. Multiple calls to this method replaces the existing clause. public void setGroupByClause (java.lang.String groupByClause)

Table 4–14 Methods for Class OrderByParameter

Method	Description
setHavingClause(String)	<p>Sets the HAVING clause for the SQL query. This clause is appended immediately after the GROUP BY clause and only GROUP BY clause exists. Multiple calls to this method replaces the existing clause.</p> <pre>public void setHavingClause(java.lang.String havingClause)</pre>
setHavingClause(String, ArrayList)	<p>Sets the HAVING clause for the SQL query. This clause is appended immediately after the GROUP BY clause and only GROUP BY clause exists. Multiple calls to this method replaces the existing clause.</p> <pre>public void setHavingClause(java.lang.String havingClause, com.sun.java.util.collections.ArrayList bi ndVars)</pre>
setOrderByClause(String)	<p>Sets the ORDER BY clause for the SQL query. Multiple calls to this method replaces the existing clause.</p> <pre>public void setOrderByClause(java.lang.String orderByClause)</pre>
setQueryStatement(String)	<p>Sets the entire SQL query statement.</p> <pre>public void setQueryStatement(java.lang.String queryStatement)</pre>
setQueryStatement(String, ArrayList)	<p>Sets the entire SQL query statement and the bind variables for it.</p> <pre>public void setQueryStatement(java.lang.String queryStatement, com.sun.java.util.collections.ArrayList bi ndVars)</pre>
setResultCount(int)	<p>Sets the total number of rows returned from the query.</p> <pre>public void setResultCount(int resultCount)</pre>

Table 4–14 Methods for Class OrderByParameter

Method	Description
setSelectClause(String)	Sets the complete SQL SELECT clause. This method is called only once, multiple calls replaces the existing SELECT clause. <pre>public void setSelectClause(java.lang.String selectClause)</pre>
setSelectClause(String, ArrayList)	Sets the complete SQL SELECT clause along with the bind variables. This method is called only once, multiple calls replaces the existing SELECT clause. Null values are not allowed as bind variables. <pre>public void setSelectClause(java.lang.String selectClause, com.sun.java.util.collections.ArrayList bi ndVars)</pre>
setSelectColumns(ArrayList)	Sets the ColumnDefinition objects of columns that are to be retrieved from database. This will be used for defining the column types while building the prepared statement. <pre>public void setSelectColumns(com.sun.java.util.collect ions.ArrayList selectCols)</pre>
setStartIndex(int)	Sets the start index of the query. <pre>public void setStartIndex(int startIndex)</pre>

addSelectClause(String)

```
public void addSelectClause(java.lang.String attrClause)
```

Adds individual SELECT attributes. This method is called multiple times, once for each column that is to be retrieved from the database. Each attribute typically is in the following format:

ALA.lead_id “leadID”

Here ALA.lead_id is the database column name and leadID is the associated application search column name. Each new attribute will be appended to the previous SELECT clause.

Parameters: attrClause - SELECT attribute.

addSelectClause(String, ArrayList)

```
public void addSelectClause(java.lang.String attrClause,  
com.sun.java.util.collections.ArrayList bindVars)
```

Adds individual SELECT attributes along with bind variables. This method is called multiple times, once for each column that is to be retrieved from the database. Each attribute typically is in the below format:

ALA.lead_id “leadID”

Here ALA.lead_id is the database column name and leadID is the associated application search column name. Each new attribute will be appended to the previous SELECT clause. Null values are not allowed as bind variables.

Parameters:

attrClause - SELECT attribute.

bindVars - Bind variables for the SELECT clause.

addSelectClauseBindVars(ArrayList)

```
public void  
addSelectClauseBindVars(com.sun.java.util.collections.ArrayList bindVars)
```

Sets the bind variables for the SELECT clause. Can be called multiple times. Null values are not allowed for binding.

Parameters: bindVars - Bind variables for the SELECT clause.

addWhereClause(String, String, Object)

```
public void addWhereClause(java.lang.String attrClause,  
java.lang.String operator, java.lang.Object bindValue)
```

Adds a SQL WHERE clause to the query along with bind variables. Null values are not allowed as bind variables. Binding is done based on the object type of the bind variable value passed. The query will be constructed using “?” and the passed value is bound. This method can be called multiple times and all where clauses are ANDed together.

Parameters:

attrClause - Attribute (left side) of the WHERE clause.

operator - Operator for the WHERE clause.

bindValue - Bind value for the WHERE clause.

addWhereClause(String, String, String)

```
public void addWhereClause(java.lang.String attrClause,  
java.lang.String operator, java.lang.String valueClause)
```

Adds a SQL WHERE clause to the query. This can be invoked multiple times, and all the WHERE clauses are ANDed together.

Parameters:

attrClause - Attribute (left side) of the WHERE clause.

operator - Operator for the WHERE clause.

valueClause - Value clause (right side) for the WHERE clause.

addWhereClause(String, String, String, ArrayList)

```
public void addWhereClause(java.lang.String attrClause,  
java.lang.String operator, java.lang.String valueClause,  
com.sun.java.util.collections.ArrayList bindVars)
```

Adds a SQL WHERE clause to the query along with bind variables. Null values are not allowed as bind variables. Binding is done based on the object type of the bind variable value passed. This method can be called multiple times and all where clauses are ANDed together.

Parameters:

attrClause - Attribute (left side) of the WHERE clause.

operator - Operator for the WHERE clause.

valueClause - Value clause (right side) for the WHERE clause. Should be "?".

bindVars - Bind variables for the WHERE clause.

buildDebugQueryStatement()

```
public java.lang.String buildDebugQueryStatement()  
throws Exception
```

Debugging Method that constructs the entire query using the different SQL clauses. The query is constructed in the following way: SELECT clause + FROM clause + WHERE clause + extra WHERE clause + GROUP BY clause + HAVING clause + ORDER BY clause. Only used for de-bugging and testing purposes.

Returns: queryStatement Complete query string and bind variables.

Throws: java.lang.Exception

closeStatement()

```
public void closeStatement()
```

Closes PreparedStatement object of QueryBuilder class. Typically invoked from the containing classes of QueryBuilder object to close/release the PreparedStatement.

executeQuery(OracleConnection)

```
public com.sun.java.util.collections.HashMap
```

```
executeQuery(oracle.jdbc.driver.OracleConnection conn)
```

```
throws QueryBuilderException, SQLException
```

Creates a SQL PreparedStatement, binds variables to it and executes the SQL query.

Parameters: conn - OracleConnection object.

Returns:

HashMap that contains ResultSet, Result Count and Start Index values: ResltSet key
—> RSET

Result Count key —> RCOUNT

Start Index key —> SINDEK

Throws: java.sql.SQLException, QueryBuilderException

getBatchSize()

```
public int getBatchSize()
```

Returns the batch size of the query.

Returns: Batch size of the query. Maximum number of rows retrieved from the query each time.

getDebugQueryStatement()

```
public java.lang.String getDebugQueryStatement()
```

Returns the entire constructed SQL statement along with the bind variables. Is mainly used for debugging/logging purposes.

Returns: Entire SQL statement plus bind variables as a String.

getQueryStatement()

```
public java.lang.String getQueryStatement()
```

Returns the entire constructed SQL statement.

Returns: Entire SQL statement.

getResultCount()

```
public int getResultCount()
```

Returns the total number of rows returned from the query.

Parameters: resultCount - Total number of rows returned for the query.

getStartIndex()

```
public int getStartIndex()
```

Returns the start index of the query.

Returns: Current start index of the query.

setBatchSize(int)

```
public void setBatchSize(int batchSize)
```

Sets the batch size for the query.

Parameters: batchSize - Batch size for the query. Maximum number of rows retrieved from the query each time.

setExtraWhereClause(String)

```
public void setExtraWhereClause(java.lang.String extraWhereClause)
```

Sets an extra SQL WHERE clause to the query. Used for setting WHERE clause involving ORs. This clause is ANDed to the existing WHERE clause if it is present. Multiple calls replaces existing clause. WHERE clause typically is in the below format:

Eg:- "(AQSV.STATUS_CODE = ? OR AQSV.STATUS_CODE = ?)"

Parameters: extraWhereClause - Complete extra WHERE clause.

setExtraWhereClause(String, ArrayList)

```
public void setExtraWhereClause(java.lang.String extraWhereClause,  
com.sun.java.util.collections.ArrayList bindVars)
```

Sets an extra SQL WHERE clause to the query. Used for setting where clause involving ORs. This clause is ANDed to the existing WHERE clause if it is present. Multiple calls replaces the existing clause. Null values are not allowed as bind variables. Binding is done based on the object type of the bind variable value passed.

Parameters:

extraWhereClause - Complete extra WHERE clause.

bindVars - Bind variables for the WHERE clause.

setFromClause(String)

```
public void setFromClause(java.lang.String fromClause)
```

Sets the SQL FROM clause for the query. Multiple calls to this method replaces the existing FROM clause. FROM clause is typically in one complete string of the form:

```
ASO_QUOTE_HEADERS_ALL AQH1, ASO_QUOTE_STATUSES_VL AQSV, HZ_
CUST_ACCOUNTS HCA1
```

Parameters: fromClause - Complete FROM clause.

setFromClause(String, ArrayList)

```
public void setFromClause(java.lang.String fromClause,
com.sun.java.util.collections.ArrayList bindVars)
```

Sets the SQL FROM clause for the query along with the bind variables. Multiple calls to this method replaces the existing FROM clause. FROM clause is typically in one complete string of the form:

```
ASO_QUOTE_HEADERS_ALL AQH1, ASO_QUOTE_STATUSES_VL AQSV, HZ_
CUST_ACCOUNTS HCA1
```

Null values are not allowed as bind variables.

Parameters:

fromClause - Complete FROM clause.

bindVars - Bind variables for the FROM clause.

setGroupByClause(String)

```
public void setGroupByClause(java.lang.String groupByClause)
```

Sets the GROUP BY clause for the SQL query. Multiple calls to this method replaces the existing clause. GROUP BY clause typically is in the below format:

Eg:- "AQH1.TOTAL_QUOTE_PRICE, AQH1.QUOTE_NAME, AQH1.QUOTE_ID"

Parameters: groupByClause - Complete GROUP BY clause.

setHavingClause(String)

```
public void setHavingClause(java.lang.String havingClause)
```

Sets the HAVING clause for the SQL query. This clause is appended immediately after the GROUP BY clause and only GROUP BY clause exists. Multiple calls to this method replaces the existing clause.

Parameters: havingClause - Complete HAVING clause.

setHavingClause(String, ArrayList)

```
public void setHavingClause(java.lang.String havingClause,  
com.sun.java.util.collections.ArrayList bindVars)
```

Sets the HAVING clause for the SQL query. This clause is appended immediately after the GROUP BY clause and only GROUP BY clause exists. Multiple calls to this method replaces the existing clause. Eg:- “AQH1.TOTAL_QUOTE_PRICE > ?”

Null values are not allowed as bind variables. Binding is done based on the object type of the bind variable value passed.

Parameters:

havingClause - Complete HAVING clause as a String.

bindVars - Bind variables in ArrayList.

setOrderByClause(String)

```
public void setOrderByClause(java.lang.String orderByClause)
```

Sets the ORDER BY clause for the SQL query. Multiple calls to this method replaces the existing clause. ORDER BY clause typically is in the below format:

Eg:- “AQH1.TOTAL_QUOTE_NAME ASC, AQH1.QUOTE_ID DESC”

Parameters: orderByClause - Complete ORDER BY clause.

setQueryStatement(String)

```
public void setQueryStatement(java.lang.String queryStatement)
```

Sets the entire SQL query statement.

Parameters: queryStatement - Entire SQL query statement.

setQueryStatement(String, ArrayList)

```
public void setQueryStatement(java.lang.String queryStatement,  
com.sun.java.util.collections.ArrayList bindVars)
```

Sets the entire SQL query statement and the bind variables for it.

Parameters:

queryStatement - Entire SQL query statement.

bindVars - Bind variables for the SQL statement.

setResultCount(int)

```
public void setResultCount(int resultCount)
```

Sets the total number of rows returned from the query.

Parameters: resultCount - Total number of rows returned from the query. -1 indicates that count should be queried from database.

setSelectClause(String)

```
public void setSelectClause(java.lang.String selectClause)
```

Sets the complete SQL SELECT clause. This method is called only once, multiple calls replaces the existing SELECT clause. SELECT clause typically is in the below format:

AQH1.quote_header_id "quoteID", AQHA.quote_name "quoteName" where AQH1.quote_header_id is the database column name and quoteID is the associated application search column name.

Parameters: selectClause - Complete SQL SELECT clause.

setSelectClause(String, ArrayList)

```
public void setSelectClause(java.lang.String selectClause,  
com.sun.java.util.collections.ArrayList bindVars)
```

Sets the complete SQL SELECT clause along with the bind variables. This method is called only once, multiple calls replaces the existing SELECT clause. Null values are not allowed as bind variables. SELECT clause typically is in the below format:

AQH1.quote_header_id "quoteID", AQHA.quote_name "quoteName", where AQH1.quote_header_id is the database column name and quoteID is the associated application search column name.

Parameters:

selectClause - Complete SQL SELECT clause.

bindVars - Bind variables for the SELECT clause.

setSelectColumns(ArrayList)

```
public void setSelectColumns(com.sun.java.util.collections.ArrayList  
selectCols)
```

Sets the ColumnDefinition objects of columns that are to be retrieved from database. This will be used for defining the column types while building the prepared statement.

Parameters: selectCols - ColumnDefinition objects of columns that are to be selected.

setStartIndex(int)

```
public void setStartIndex(int startIndex)
```

Sets the start index of the query.

Parameters: startIndex - Indicates which row number to start retrieving data. -1 indicates the last rows.

4.9 Class QueryBuilderException

```

java.lang.Object
|
+--java.lang.Throwable
    |
    +--java.lang.Exception
        |
        +--oracle.apps.jtf.base.resources.FrameworkException
            |
            +--oracle.apps.qot.perzquery.QueryBuilderException
  
```

All Implemented Interfaces: java.io.Serializable

```
public class QueryBuilderException extends
oracle.apps.jtf.base.resources.FrameworkException
```

The QueryBuilderException is thrown if any errors encountered while building or executing the SQL query statement. The probable causes for this exception being thrown are:

- If any bind parameter of the query encounters null as a bind value.
- Incorrect database column information for defining the query column types.
- Invalid column operator-value combination, which causes the query to fail. Eg: Quote_Amount = 'sodas'.
- Incorrect SQL clauses or invalid SQL clause combinations, which causes the query to fail.

Table 4–15 Inherited Member Summary

Fields inherited from interface FrameworkException

DEBUG, ERROR, FATAL, INFORMATION, NONE, WARNING, defaultMsgMgr

Methods inherited from interface FrameworkException

Table 4–15 Inherited Member Summary

```

addException(Exception), convertException(Exception), getAllInfo(),
getCurrentMessageManager(), getExceptionStack(), getExceptionStackRec(),
getExternException(), getKey(), getMessage(), getMessageManager(), getMessageStack(),
getParameters(), getParentExcep(), getRootException(), getRootExternExcept(),
getSeverity(), getThrowerInfo(), getWholeStack(), printAllInfo(PrintStream),
printAllInfo(PrintStream), printMesg(PrintStream), printMesg(PrintStream),
printMessageStack(PrintWriter), printMessageStack(PrintWriter),
printStackTrace(PrintWriter), printStackTrace(PrintWriter),
printThrowerInfo(PrintStream), printThrowerInfo(PrintStream),
printWholeStack(PrintStream), printWholeStack(PrintStream), setCurrents(),
setStackTrace(String)

```

Methods inherited from class Object

```

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long,
int), wait(long, int)

```

Methods inherited from class Throwable

```

fillInStackTrace(), getLocalizedMessage(), printStackTrace(), toString()

```

4.9.1 Fields for Class QueryBuilderException

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to `oracle.apps.fnd.common.VersionInfo.recordClassVersion`

4.9.2 Constructors for Class QueryBuilderException

QueryBuilderException(Exception, String)

```
public QueryBuilderException(java.lang.Exception e,  
java.lang.String errorKey)
```

Constructs QueryBuilderException with the given exception and an error key.

Parameters:

e - Parent exception.

errorKey - Error key.

QueryBuilderException(Exception, String, Object[])

```
public QueryBuilderException(java.lang.Exception e,  
java.lang.String errorKey, java.lang.Object[] params)
```

Constructs QueryBuilderException with the given exception, errorKey and parameter tokens.

Parameters:

e - Parent exception.

errorKey - Error key.

params - An array of tokens for errorKey.

QueryBuilderException(Exception, String, String)

```
public QueryBuilderException(java.lang.Exception e,  
java.lang.String errorKey, java.lang.String param)
```

Constructs QueryBuilderException with the given exception, errorKey and parameter.

Parameters:

e - Parent exception.

errorKey - Error key.

param - Parameter for errorKey.

QueryBuilderException(int, String)

```
public QueryBuilderException(int err_msg_count,  
java.lang.String errorKey)  
throws FrameworkException
```


Constructs QueryBuilderException with the message count and an error key. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QueryBuilderException(int, String, Object[])

```
public QueryBuilderException(int err_msg_count,  
    java.lang.String errorKey, java.lang.Object[] params)  
    throws FrameworkException
```

Constructs QueryBuilderException with the message count, error key and parameter tokens. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

params - An array of tokens for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QueryBuilderException(int, String, String)

```
public QueryBuilderException(int err_msg_count,  
    java.lang.String errorKey, java.lang.String param)  
    throws FrameworkException
```

Constructs QueryBuilderException with the message count, error key and a parameter token. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

param - A Token for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

QueryBuilderException(String)

```
public QueryBuilderException(java.lang.String errorKey)
```

Constructs QueryBuilderException with the errorKey.

Parameters: errorKey - Error key.

QueryBuilderException(String, Object[])

```
public QueryBuilderException(java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs QueryBuilderException with the errorKey and parameter tokens.

Parameters:

errorKey - Error key.

params - An array of tokens for errorKey.

QueryBuilderException(String, String)

```
public QueryBuilderException(java.lang.String err_msg,  
java.lang.String errorKey)
```

Constructs QueryBuilderException with the error message and error key.

Parameters:

err_msg - Error message.

errorKey - Error key.

QueryBuilderException(String, String, Object[])

```
public QueryBuilderException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.Object[] params)
```

Constructs QueryBuilderException with the error message, error key and parameter tokens.

Parameters:

err_msg - Error message.

errorKey - Error key.

params - An array of tokens for errorKey.

QueryBuilderException(String, String, String)

```
public QueryBuilderException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.String param)
```

Constructs QueryBuilderException with the error message, error key and parameter tokens.

Parameters:

err_msg - Error message.

errorKey - Error key.

param - Token for errorKey.

4.10 Class SavedSearch

```

java.lang.Object
|
+--oracle.apps.qot.perzquery.SearchInfo
|
+--oracle.apps.qot.perzquery.SavedSearch

```

public class **SavedSearch** extends SearchInfo

SavedSearch class represents the search information that is stored in the database and is used to save, delete and load the search information from database.

SavedSearch class can also be used as temporary object to hold search information. Typically the search information consists of display columns, search conditions, and sort criteria etc. SavedSearch class inherits the search attributes from SearchInfo class. It also contains personalized search information like search ID, search name, search type, search description and application ID.

Table 4–16 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

Methods inherited from class SearchInfo

Table 4–16 Inherited Member Summary

```
addConditionParameter(String, String, String[]), addConditionParameter(String, String,
String[]), addConditionParameter(String, String, String[]),
addConditionParameter(String, String, String[]), addConditionParameter(String, String,
String, String, String[]), addConditionParameter(String, String,
String[]), addConditionParameter(String, String, String[]), addDisplayColumn(String),
addDisplayColumn(String), addDisplayColumn(String), addOrderByParameter(String,
Boolean), addOrderByParameter(String, Boolean), addOrderByParameter(String,
Boolean), addOrderByParameter(String, Boolean), getCondParamOperator(String),
getCondParamValue(String, String), getCondParamValue(String, String),
getCondParamValues(String, String[]), getCondParamValues(String, String[]),
getCondParamValues(String, String[]), getConditionParameter(String),
getConditionParameters(), getDispColNamesList(), getDispColumnNames(),
getDisplayColumns(), getDisplayRows(), getOrdByParamNames(),
getOrdByParamSequence(String), getOrderByParameters(),
setConditionParameters(ConditionParameter[]), setDispColumnNames(String[]),
setDisplayColumns(DisplayColumn[]), setDisplayRows(int),
setOrdByParamNames(String[]), setOrderByParameters(OrderByParameter[])
```

4.10.1 Fields for Class SavedSearch

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to `oracle.apps.fnd.common.VersionInfo.recordClassVersion`.

4.10.2 Constructors for Class SavedSearch

SavedSearch()

```
public SavedSearch()
```

Constructs SavedSearch object.

4.10.3 Methods for Class SavedSearch

The following table is an index of the Class SavedSearch methods:

Table 4–17 Methods for Class SavedSearch

Method	Description
delete(String, String)	<p>Deletes the search of the user for the given search ID. Mandatory information required to delete the search are: Application Profile Name, User Profile Name, Application ID, Search Type, Search ID Search Name.</p> <pre>public void delete(java.lang.String seededProfileName, java.lang.String userProfileName)</pre> <p>throws FrameworkException, SQLException, SavedSearchException</p>
getAppId()	<p>Returns the application ID of the search.</p> <pre>public java.math.BigDecimal getAppId()</pre>
getQueryDesc()	<p>Returns the description of saved search.</p> <pre>public java.lang.String getQueryDesc()</pre>
getQueryId()	<p>Returns the ID of the saved search.</p> <pre>public java.math.BigDecimal getQueryId()</pre>
getQueryName()	<p>Returns the name of the saved search.</p> <pre>public java.lang.String getQueryName()</pre>
getQueryType()	<p>Returns the type of saved search.</p> <pre>public java.lang.String getQueryType()</pre>
isDefaultQuery(String)	<p>Checks whether the search is set as default for the given user. Mandatory information required for this check to be done successfully are: Application Profile Name, User Profile Name, Application ID, Search Type and Search ID.</p> <pre>public boolean isDefaultQuery(java.lang.String seededProfileName, java.lang.String userProfileName)</pre> <p>throws FrameworkException, SQLException</p>
isSeededQuery()	<p>Returns the boolean flag indicating whether the search is application seeded.</p> <pre>public boolean isSeededQuery()</pre>

Table 4–17 Methods for Class SavedSearch

Method	Description
<code>list(BigDecimal, String, String, String)</code>	Returns all the user searches for the given search type and also the application seeded searches. <pre>public static oracle.apps.qot.util.QueryResultSet list(java.math.BigDecimal appId, java.lang.String seededProfileName, java.lang.String userProfileName, java.lang.String queryType) throws FrameworkException, SQLException</pre>
<code>list(BigDecimal, String, String, String, int, int, int)</code>	Returns all the user saved searches for the given search type and also the application seeded searches. Used to retrieve a set (batch size) of rows at a time. <pre>public static oracle.apps.qot.util.QueryResultSet list(java.math.BigDecimal appId, java.lang.String seededProfileName, java.lang.String userProfileName, java.lang.String queryType, int batchSize, int startIndex, int resultCount) throws FrameworkException, SQLException</pre>
<code>load(BigDecimal, String, BigDecimal)</code>	Loads a SavedSearch object with search information for a given search ID. <pre>public static oracle.apps.qot.perzquery.SavedSearch load(java.math.BigDecimal appId, java.lang.String seededProfileName, java.math.BigDecimal queryId) throws FrameworkException, SQLException, S avedSearchException</pre>
<code>loadDefaultSavedSearch(BigDecimal, String, String, String)</code>	Loads the SavedSearch object for default search of the user. <pre>public static oracle.apps.qot.perzquery.SavedSearch loadDefaultSavedSearch(java.math.BigDecima l appId, java.lang.String seededProfileName, java.lang.String userProfileName, java.lang.String queryType) throws FrameworkException, SQLException, S avedSearchException</pre>

Table 4–17 Methods for Class SavedSearch

Method	Description
save(String, String, Boolean)	<p>Saves the SavedSearch object with all the search information of the user. Profile name of the application, Profile name of the user, Application ID, Search Name, Search Type and Search ID(for new search) are the mandatory information required to save the search successfully. A search name is treated as duplicate, if the same search name exists for combination of Application ID, Profile and Search Type. Application seeded searches can not be updated, but can be set as default for the user.</p> <pre>public void save(java.lang.String seededProfileName, java.lang.String userProfileName, java.lang.Boolean saveAsDefaultQuery)</pre> <p>throws SavedSearchException, FrameworkException, SQLException</p>
setAppId(BigDecimal)	<p>Sets the application ID of the search.</p> <pre>public void setAppId(java.math.BigDecimal appId)</pre>
setQueryDesc(String)	<p>Sets the description of saved search.</p> <pre>public void setQueryDesc(java.lang.String queryDesc)</pre>
setQueryId(BigDecimal)	<p>Sets the ID of the saved search.</p> <pre>public void setQueryId(java.math.BigDecimal queryId)</pre>
setQueryName(String)	<p>Sets the name of the saved search.</p> <pre>public void setQueryName(java.lang.String queryName)</pre>
setQueryType(String)	<p>Sets the type of saved search.</p> <pre>public void setQueryType(java.lang.String queryType)</pre>
toString()	<p>Converts SavedSearch object into a String representation.</p> <pre>public java.lang.String toString()</pre>

delete(String, String)

```
public void delete(java.lang.String seededProfileName, java.lang.String userProfileName)
```

throws `FrameworkException`, `SQLException`, `SavedSearchException`

Deletes the search of the user for the given search ID. Mandatory information required to delete the search are: Application Profile Name, User Profile Name, Application ID, Search Type, Search ID, and Search Name.

Parameters:

`seededProfileName` - Profile name of the application.

`userProfileName` - Profile name of the user.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

`SavedSearchException` - If an error occurs while deleting `SavedSearch` object.

getAppId()

```
public java.math.BigDecimal getAppId()
```

Returns the application ID of the search.

Returns: Application ID of the search

getQueryDesc()

```
public java.lang.String getQueryDesc()
```

Returns the description of saved search.

Returns: Description of saved search

getQueryId()

```
public java.math.BigDecimal getQueryId()
```

Returns the ID of the saved search.

Returns: ID of the saved search.

getQueryName()

```
public java.lang.String getQueryName()
```

Returns the name of the saved search.

Returns: Name of saved search.

getQueryType()

```
public java.lang.String getQueryType()
```


Returns the type of saved search.

Returns: Type of saved search.

isDefaultQuery(String, String)

```
public boolean isDefaultQuery(java.lang.String seededProfileName,  
java.lang.String userProfileName)
```

throws FrameworkException, SQLException

Checks whether the search is set as default for the given user. Mandatory information required for this check to be done successfully are: Application Profile Name, User Profile Name, Application ID, Search Type and Search ID.

Parameters:

seededProfileName - Profile name of the application.

userProfileName - Profile name of the user.

Returns: Indicates whether the search is set as user's default search.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

isSeededQuery()

```
public boolean isSeededQuery()
```

Returns the boolean flag indicating whether the search is application seeded.

Returns: Indicates whether the search is application seeded.

list(BigDecimal, String, String, String)

```
public static oracle.apps.qot.util.QueryResultSet
```

```
list(java.math.BigDecimal appId, java.lang.String seededProfileName,  
java.lang.String userProfileName, java.lang.String queryType)
```

throws FrameworkException, SQLException

Returns all the user searches for the given search type and also the application seeded searches.

Parameters:

appId - Application ID of the search.

seededProfileName - Profile name of the application.

userProfileName - Profile name of the user.

queryType - Type of query (Saved Search).

Returns: QueryResultSet object containing array of SavedSearch objects. Only search ID and search name is populated in each SavedSearch object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

list(BigDecimal, String, String, String, int, int, int)

```
public static oracle.apps.got.util.QueryResultSet  
list(java.math.BigDecimal appId, java.lang.String seededProfileName,  
java.lang.String userProfileName, java.lang.String queryType,  
int batchSize, int startIndex, int resultCount)  
throws FrameworkException, SQLException
```

Returns all the user saved searches for the given search type and also the application seeded searches. Used to retrieve a set (batch size) of rows at a time.

Parameters:

appId - Application ID of the search.

seededProfileName - Profile name of the application.

userProfileName - Profile name of the user.

queryType - Type of query (saved search).

batchSize - Batch size for the query.

startIndex - Indicates which row number to start retrieving data. -1 indicates the last rows.

resultCount - Total number of rows returned from the query. If -1, the count will be queried from the database.

Returns: QueryResultSet object containing array of SavedSearch objects. Only header information like search ID, search name, search description etc., is populated in each SavedSearch object.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(BigDecimal, String, BigDecimal)

```
public static oracle.apps.qot.perzquery.SavedSearch  
load(java.math.BigDecimal appId, java.lang.String seededProfileName,  
java.math.BigDecimal queryId)
```

throws FrameworkException, SQLException, SavedSearchException

Loads a SavedSearch object with search information for a given search ID.

Parameters:

appId - Application ID associated with the search

seededProfileName - Profile name of the application.

queryId - ID of search.

Returns: SavedSearch object with all the search information.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

SavedSearchException - If an error occurs while loading SavedSearch object.

loadDefaultSavedSearch(BigDecimal, String, String, String)

```
public static oracle.apps.qot.perzquery.SavedSearch  
loadDefaultSavedSearch(java.math.BigDecimal appId,  
java.lang.String seededProfileName, java.lang.String userProfileName,  
java.lang.String queryType)
```

throws FrameworkException, SQLException, SavedSearchException

Loads the SavedSearch object for default search of the user.

Parameters:

appId - Application ID of the search.

seededProfileName - Profile name of the application.

userProfileName - Profile name of the user.

queryType - Type of the query.

Returns: SavedSearch object with all the search information.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

SavedSearchException - Error while loading SavedSearch object.

save(String, String, Boolean)

```
public void save(java.lang.String seededProfileName,  
java.lang.String userProfileName, java.lang.Boolean saveAsDefaultQuery)  
throws SavedSearchException, FrameworkException, SQLException
```

Saves the SavedSearch object with all the search information of the user. Profile name of the application, Profile name of the user, Application ID, Search Name, Search Type and Search ID (for new search) are the mandatory information required to save the search successfully. A search name is treated as duplicate, if the same search name exists for combination of Application ID, Profile and Search Type. Application seeded searches can not be updated, but can be set as default for the user.

Parameters:

seededProfileName - Profile name of the application.

userProfileName - Profile name of the user.

saveAsDefaultQuery - Indicates whether this search is to be set as default search for the user. Expected values are:

- Boolean.TRUE ---> Sets the search as default.
- Boolean.FALSE ---> Unsets the default search.

null ---> Status-Quo is maintained.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database occurs.

SavedSearchException - If an error occurs while saving SavedSearch object.

setAppId(BigDecimal)

```
public void setAppId(java.math.BigDecimal appId)
```

Sets the application ID of the search.

Parameters: id - Application ID of the search.

setQueryDesc(String)

```
public void setQueryDesc(java.lang.String queryDesc)
```

Sets the description of saved search.

Parameters: queryDesc - Description of saved search.

setQueryId(BigDecimal)

```
public void setQueryId(java.math.BigDecimal queryId)
```

Sets the ID of the saved search.

Parameters: queryId - ID of saved search.

setQueryName(String)

```
public void setQueryName(java.lang.String queryName)
```

Sets the name of the saved search.

Parameters: queryName - Name of saved search.

setQueryType(String)

```
public void setQueryType(java.lang.String queryType)
```

Sets the type of saved search.

Parameters: queryType - Type of saved search.

toString()

```
public java.lang.String toString()
```

Converts SavedSearch object into a String representation.

Overrides: toString in class SearchInfo.

Returns: A String representation of SavedSearch object.

4.11 Class SavedSearchException

```

java.lang.Object
|
+--java.lang.Throwable
    |
    +--java.lang.Exception
        |
        +--oracle.apps.jtf.base.resources.FrameworkException
            |
            +--oracle.apps.qot.perzquery.SavedSearchException

```

All Implemented Interfaces: java.io.Serializable

```
public class SavedSearchException extends
oracle.apps.jtf.base.resources.FrameworkException
```

The SavedSearchException is thrown if any errors encountered while loading, saving or deleting the search criteria from database. Some probable causes for this exception being thrown are:

- Invalid search parameter values (E.g., too long value).
- Invalid user profile or application profile is passed to saved search object.
- Create/Update a search with the duplicate name for the same user, application and search type.

Table 4–18 Inherited Member Summary

Fields inherited from interface FrameworkException

DEBUG, ERROR, FATAL, INFORMATION, NONE, WARNING, defaultMsgMgr

Methods inherited from interface FrameworkException

addException(Exception), convertException(Exception), getAllInfo(),
getCurrentMessageManager(), getExceptionStack(), getExceptionStackRec(),
getExternException(), getKey(), getMessage(), getMessageManager(), getMessageStack(),
getParameters(), getParentExcep(), getRootException(), getRootExternExcept(),
getSeverity(), getThrowerInfo(), getWholeStack(), printAllInfo(PrintStream),
printAllInfo(PrintStream), printMesg(PrintStream), printMesg(PrintStream),
printMessageStack(PrintWriter), printMessageStack(PrintWriter),
printStackTrace(PrintWriter), printStackTrace(PrintWriter),
printThrowerInfo(PrintStream), printThrowerInfo(PrintStream),
printWholeStack(PrintStream), printWholeStack(PrintStream), setCurrents(),
setStackTrace(String)

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long,
int), wait(long, int)

Methods inherited from class Throwable

fillInStackTrace(), getLocalizedMessage(), printStackTrace(), toString()

4.11.1 Fields for Class SavedSearchException

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to `oracle.apps.fnd.common.VersionInfo.recordClassVersion`.

4.11.2 Constructors for Class SavedSearchException

SavedSearchException(Exception, String)

```
public SavedSearchException(java.lang.Exception e,  
java.lang.String errorKey)
```

Constructs SavedSearchException with the given exception and error key.

Parameters:

e - Parent exception.

errorKey - Error key.

SavedSearchException(Exception, String, Object[])

```
public SavedSearchException(java.lang.Exception e,  
java.lang.String errorKey, java.lang.Object[] params)
```

Constructs SavedSearchException with the given exception, error key and token parameters.

Parameters:

e - Parent exception.

errorKey - Error key.

params - An array of tokens for errorKey.

SavedSearchException(Exception, String, String)

```
public SavedSearchException(java.lang.Exception e,  
java.lang.String errorKey, java.lang.String param)
```

Constructs SavedSearchException with the given exception, error key and parameter token.

Parameters:

e - Parent exception.

errorKey - Error key.

param - Token for errorKey.

SavedSearchException(int, String)

```
public SavedSearchException(int err_msg_count, java.lang.String errorKey)
throws FrameworkException
```

Constructs SavedSearchException with the message count and error key. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

SavedSearchException(int, String, Object[])

```
public SavedSearchException(int err_msg_count, java.lang.String errorKey,
java.lang.Object[] params)
throws FrameworkException
```

Constructs SavedSearchException with the message count, error key and parameter tokens. Errors at the PL/SQL level will be retrieved.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack

errorKey - Error key.

params - An array of tokens for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

SavedSearchException(int, String, String)

```
public SavedSearchException(int err_msg_count, java.lang.String errorKey,
java.lang.String param)
```


throws `FrameworkException`

Constructs `SavedSearchException` with the message count, error key and parameter token. Errors at the PL/SQL level will be retrieved.

Parameters:

`err_msg_count` - The number of messages to be returned from the pl/sql error stack.

`errorKey` - Error key.

`param` - A token for the `errorKey`.

Throws: `oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

SavedSearchException(String)

```
public SavedSearchException(java.lang.String errorKey)
```

Constructs `SavedSearchException` with the `errorKey`.

Parameters: `errorKey` - Error key.

SavedSearchException(String, Object[])

```
public SavedSearchException(java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs `SavedSearchException` with the `errorKey`.

Parameters:

`errorKey` - Error key.

`params` - An array of tokens for `errorKey`.

SavedSearchException(String, String)

```
public SavedSearchException(java.lang.String err_msg,  
java.lang.String errorKey)
```

Constructs `SavedSearchException` with the error message and error key.

Parameters:

`err_msg` - Error message.

`errorKey` - Error key.

SavedSearchException(String, String, Object[])

```
public SavedSearchException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.Object[] params)
```

Constructs SavedSearchException with the error message, error key and token parameters.

Parameters:

err_msg - Error message.

errorKey - Error key.

params - An array of tokens for errorKey.

SavedSearchException(String, String, String)

```
public SavedSearchException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.String param)
```

Constructs an Exception with the error message, error key and token parameter.

Parameters:

err_msg - Error message.

errorKey - Error key.

param - Token for errorKey.

4.12 Class SearchInfo

```
java.lang.Object  
|  
+--oracle.apps.got.perzquery.SearchInfo  
Direct Known Subclasses: SavedSearch  
public class SearchInfo
```

The SearchInfo class represents all the filter (search) information entered by a user for one search page. SavedSearch class uses this information to load and save searches. This class has various accessory and mutator methods to set and get the search information like displayRows, displayColumns, conditionParameters and orderByParameters. The search information is stored in DisplayColumn, ConditionParameter and OrderByParameter objects.

Table 4–19 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

4.12.1 Fields for Class SearchInfo

RCS_ID

```
public static final java.lang.String RCS_ID
```

Standard public final static String which is initialized with the usual RCS header used by ARCS.

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

Standard public final static boolean which is initialized by a call to `oracle.apps.fnd.common.VersionInfo.recordClassVersion`.

4.12.2 Constructors for Class SearchInfo

SearchInfo()

```
public SearchInfo()
```

Constructs SearchInfo object.

4.12.3 Methods for Class SearchInfo

The following table is an index of the Class SearchInfo methods:

Table 4–20 *Methods for Class Component*

Method	Description
<code>addConditionParameter(ConditionParameter)</code>	Adds a new ConditionParameter object. public void addConditionParameter (oracle.apps.got.perzquery.ConditionParameter condParam)
<code>addConditionParameter(ConditionParameter[])</code>	Adds new ConditionParameter objects. public void addConditionParameter (oracle.apps.got.perzquery.ConditionParameter[] condParams)
<code>addConditionParameter(String, String)</code>	Adds a new ConditionParameter object. Condition operator is defaulted to “=”. public void addConditionParameter (java.lang.String name, java.lang.String value)

Table 4–20 Methods for Class Component

Method	Description
<code>addConditionParameter(String, String[])</code>	Adds a new ConditionParameter object. public void addConditionParameter (java.lang.String name, java.lang.String[] values)
<code>addConditionParameter(String, String, String)</code>	Adds a new ConditionParameter object. public void addConditionParameter (java.lang.String name, java.lang.String operator, java.lang.String value)
<code>addConditionParameter(String, String, String[])</code>	Adds a new ConditionParameter object. public void addConditionParameter (java.lang.String name, java.lang.String operator, java.lang.String[] values)
<code>addDisplayColumn(DisplayColumn)</code>	Adds a new DisplayColumn object. public void addDisplayColumn (oracle.apps.got.perzquery.DisplayColumn dcol)
<code>addDisplayColumn(DisplayColumn[])</code>	Adds new DisplayColumn objects. public void addDisplayColumn (oracle.apps.got.perzquery.DisplayColumn[] dcols)
<code>addDisplayColumn(String)</code>	Adds a new DisplayColumn. public void addDisplayColumn (java.lang.String name)
<code>addOrderByParameter(OrderByParameter)</code>	Adds a new OrderByParameter object. public void addOrderByParameter (oracle.apps.got.perzquery.OrderByParameter obp)
<code>addOrderByParameter(OrderByParameter[])</code>	Adds new OrderByParameter objects. public void addOrderByParameter (oracle.apps.got.perzquery.OrderByParameter[] obps)

Table 4–20 Methods for Class Component

Method	Description
<code>addOrderByParameter(String)</code>	Adds a new <code>OrderByParameter</code> object. Order by sequence is defaulted to true (ascending). public void addOrderByParameter (java.lang.String name)
<code>addOrderByParameter(String, Boolean)</code>	Adds a new <code>OrderByParameter</code> object. public void addOrderByParameter (java.lang.String name, java.lang.Boolean sequence)
<code>getConditionParameter(String)</code>	Returns all the matching <code>ConditionParameter</code> objects for the given search parameter name. public oracle.apps.qot.perzquery.ConditionParameter[] getConditionParameter (java.lang.String name)
<code>getConditionParameters()</code>	Returns all the <code>ConditionParameter</code> objects. public oracle.apps.qot.perzquery.ConditionParameter[] getConditionParameters ()
<code>getCondParamOperator(String)</code>	Returns the operator of the <code>ConditionParameter</code> for the given search parameter name. public java.lang.String getCondParamOperator (java.lang.String name)
<code>getCondParamValue(String)</code>	Returns the value of the <code>ConditionParameter</code> for the given search parameter name. public java.lang.String getCondParamValue (java.lang.String name)
<code>getCondParamValue(String, String)</code>	Returns the value of the <code>ConditionParameter</code> for the given search parameter name and operator. public java.lang.String getCondParamValue (java.lang.String name, java.lang.String cond)
<code>getCondParamValues(String)</code>	Returns the value of the <code>ConditionParameter</code> for the given search parameter name. public java.lang.String[] getCondParamValues (java.lang.String name)

Table 4–20 Methods for Class Component

Method	Description
<code>getCondParamValues(String, String)</code>	Returns the value of the ConditionParameter for the given search parameter name and operator. <pre>public java.lang.String[] getCondParamValues(java.lang.String name, java.lang.String cond)</pre>
<code>getCondParamValues(String, String[])</code>	Returns the value of the ConditionParameter for the given search parameter name and operators. <pre>public java.lang.String[] getCondParamValues(java.lang.String name, java.lang.String[] conds)</pre>
<code>getDispColNamesList()</code>	Returns all the parameter names of the DisplayColumn objects. <pre>public com.sun.java.util.collections.ArrayList getDispColNamesList()</pre>
<code>getDispColumnNames()</code>	Returns all the parameter names of the DisplayColumn objects. <pre>public java.lang.String[] getDispColumnNames()</pre>
<code>getDisplayColumns()</code>	Returns all the DisplayColumn objects. <pre>public oracle.apps.got.perzquery.DisplayColumn[] getDisplayColumns()</pre>
<code>getDisplayRows()</code>	Returns the number of rows to display (batch size for the query). Maximum number of rows returned from a query at a time. <pre>public int getDisplayRows()</pre>
<code>getOrdByParamNames()</code>	Returns all the names of the OrderByParameter objects. <pre>public java.lang.String[] getOrdByParamNames()</pre>
<code>getOrdByParamSequence(String)</code>	Returns order by sequence (Ascending or Descending) for the given sort parameter name <pre>public java.lang.Boolean getOrdByParamSequence(java.lang.String name)</pre>

Table 4–20 Methods for Class Component

Method	Description
<code>getOrderByParameters()</code>	Returns all the <code>OrderByParameter</code> objects. public oracle.apps.qot.perzquery.OrderByParameter [] getOrderByParameters()
<code>setConditionParameters(ConditionParameter[])</code>	Sets all the <code>ConditionParameter</code> objects. public void setConditionParameters (oracle.apps.qot.perzquery.ConditionParameter[] c)
<code>setDisplayColumnNames(String)</code>	Sets names of all the <code>DisplayColumn</code> objects. public void setDisplayColumnNames (java.lang.String[] cols)
<code>setDisplayColumns(DisplayColumn[])</code>	Sets all the <code>DisplayColumn</code> objects. public void setDisplayColumns (oracle.apps.qot.perzquery.DisplayColumn[] dcols)
<code>setDisplayRows(int)</code>	Sets the number of rows to display (batch size for the query). Maximum number of rows returned from a query at a time. public void setDisplayRows (int rows)
<code>setOrdByParamNames(String[])</code>	Sets all the names of the <code>OrderByParameter</code> objects. Order by sequence is defaulted to true (ascending). public void setOrdByParamNames (java.lang.String[] obParamNames)
<code>setOrderByParameters(OrderByParameters[])</code>	Sets all the <code>OrderByParameter</code> objects. public void setOrderByParameters (oracle.apps.qot.perzquery.OrderByParameter[] obp)
<code>toString()</code>	Converts <code>SearchInfo</code> object into a <code>String</code> representation. public java.lang.String toString()

addConditionParameter(ConditionParameter)

```
public void  
addConditionParameter(oracle.apps.got.perzquery.ConditionParameter  
condParam)
```

Adds a new ConditionParameter object.

Parameters: condParam - ConditionParameter object.

addConditionParameter(ConditionParameter[])

```
public void  
addConditionParameter(oracle.apps.got.perzquery.ConditionParameter[]  
condParams)
```

Adds new ConditionParameter objects.

Parameters: condParams - Array of ConditionParameter objects.

addConditionParameter(String, String)

```
public void addConditionParameter(java.lang.String name,  
java.lang.String value)
```

Adds a new ConditionParameter object. Condition operator is defaulted to “=”.

Parameters:

name - Name of search parameter.

value - Value of search parameter.

addConditionParameter(String, String[])

```
public void addConditionParameter(java.lang.String name,  
java.lang.String[] values)
```

Adds a new ConditionParameter object.

Parameters:

name - Name of search parameter.

values - Value of search parameter.

addConditionParameter(String, String, String)

```
public void addConditionParameter(java.lang.String name,  
java.lang.String operator, java.lang.String value)
```

Adds a new ConditionParameter object.

Parameters:

name - Name of search parameter.

operator - Operator of search parameter.

value - Value of search parameter.

addConditionParameter(String, String, String[])

```
public void addConditionParameter(java.lang.String name,  
java.lang.String operator, java.lang.String[] values)
```

Adds a new ConditionParameter object.

Parameters:

name - Name of search parameter.

operator - Operator of search parameter.

values - Values of search parameter.

addDisplayColumn(DisplayColumn)

```
public void addDisplayColumn(oracle.apps.qot.perzquery.DisplayColumn  
dcol)
```

Adds a new DisplayColumn object.

Parameters: dcol - DisplayColumn to be added.

addDisplayColumn(DisplayColumn[])

```
public void addDisplayColumn(oracle.apps.qot.perzquery.DisplayColumn[]  
dcols)
```

Adds new DisplayColumn objects.

Parameters: dcols - Array of DisplayColumn objects.

addDisplayColumn(String)

```
public void addDisplayColumn(java.lang.String name)
```

Adds a new DisplayColumn.

Parameters: name - Name of DisplayColumn to be added.

addOrderByParameter(OrderByParameter)

```
public void  
addOrderByParameter(oracle.apps.qot.perzquery.OrderByParameter obp)
```

Adds a new OrderByParameter object.

Parameters: obp - OrderByParameter object.

addOrderByParameter(OrderByParameter[])

```
public void  
addOrderByParameter(oracle.apps.got.perzquery.OrderByParameter[] obps)
```

Adds new OrderByParameter objects.

Parameters: obps - Array of OrderByParameter objects

addOrderByParameter(String)

```
public void addOrderByParameter(java.lang.String name)
```

Adds a new OrderByParameter object. Order by sequence is defaulted to true (ascending).

Parameters: name - Name of sort parameter.

addOrderByParameter(String, Boolean)

```
public void addOrderByParameter(java.lang.String name,  
java.lang.Boolean sequence)
```

Adds a new OrderByParameter object.

Parameters:

name - Name of sort parameter.

sequence - Order by sequence of sort parameter.

getConditionParameter(String)

```
public oracle.apps.got.perzquery.ConditionParameter[]  
getConditionParameter(java.lang.String name)
```

Returns all the matching ConditionParameter objects for the given search parameter name.

Parameters: name - Name of ConditionParameter to look for.

Returns: Array of matching ConditionParameter objects.

getConditionParameters()

```
public oracle.apps.got.perzquery.ConditionParameter[]  
getConditionParameters()
```

Returns all the ConditionParameter objects.

Returns: Array of ConditionParameter objects

getCondParamOperator(String)

```
public java.lang.String getCondParamOperator(java.lang.String name)
```

Returns the operator of the ConditionParameter for the given search parameter name.

Parameters: name - Name of the ConditionParameter to look for.

Returns: Operator of the first matching ConditionParameter.

getCondParamValue(String)

```
public java.lang.String getCondParamValue(java.lang.String name)
```

Returns the value of the ConditionParameter for the given search parameter name.

Parameters: name - Name of the ConditionParameter to look for.

Returns: Value of the first matching ConditionParameter.

getCondParamValue(String, String)

```
public java.lang.String getCondParamValue(java.lang.String name,  
java.lang.String cond)
```

Returns the value of the ConditionParameter for the given search parameter name and operator.

Parameters:

name - Name of the ConditionParameter to look for.

cond - Operator of the ConditionParameter to look for.

Returns: Value of the first matching ConditionParameter.

getCondParamValues(String)

```
public java.lang.String[] getCondParamValues(java.lang.String name)
```

Returns the value of the ConditionParameter for the given search parameter name.

Parameters: name - Name of the ConditionParameter to look for.

Returns: Values of the first matching ConditionParameter.

getCondParamValues(String, String)

```
public java.lang.String[] getCondParamValues(java.lang.String name,  
java.lang.String cond)
```

Returns the value of the ConditionParameter for the given search parameter name and operator.

Parameters:

name - Name of the ConditionParameter to look for.

cond - Operator of the ConditionParameter to look for.

Returns: Values of the first matching ConditionParameter.

getCondParamValues(String, String[])

```
public java.lang.String[] getCondParamValues(java.lang.String name,  
java.lang.String[] conds)
```

Returns the value of the ConditionParameter for the given search parameter name and operators.

Parameters: name - Name of the ConditionParameter to look for.

conds - Operators of the ConditionParameter to look for.

Returns: Values of the first matching ConditionParameter.

getDispColNamesList()

```
public com.sun.java.util.collections.ArrayList  
getDispColNamesList()
```

Returns all the parameter names of the DisplayColumn objects.

Returns: ArrayList of names of all the parameters that are to be displayed.

getDispColumnNames()

```
public java.lang.String[] getDispColumnNames()
```

Returns all the parameter names of the DisplayColumn objects.

Returns: Array of names of all the parameters that are to be displayed

getDisplayColumns()

```
public oracle.apps.qot.perzquery.DisplayColumn[] getDisplayColumns()
```

Returns all the DisplayColumn objects.

Returns: Array of DisplayColumn objects.

getDisplayRows()

```
public int getDisplayRows()
```

Returns the number of rows to display (batch size for the query). Maximum number of rows returned from a query at a time.

Returns: Number of rows to display (batch size for the query).

getOrdByParamNames()

```
public java.lang.String[] getOrdByParamNames()
```

Returns all the names of the OrderByParameter objects.

Returns: Array of all sort parameter names.

getOrdByParamSequence(String)

```
public java.lang.Boolean getOrdByParamSequence(java.lang.String name)
```

Returns order by sequence (Ascending or Descending) for the given sort parameter name.

Parameters: name - Name of the sort parameter to look for.

Returns: Order by sequence of the first matching OrderByParameter.

getOrderByParameters()

```
public oracle.apps.qot.perzquery.OrderByParameter[]  
getOrderByParameters()
```

Returns all the OrderByParameter objects.

Returns: Array of OrderByParameter objects

setConditionParameters(ConditionParameter[])

```
public void  
setConditionParameters(oracle.apps.qot.perzquery.ConditionParameter[] c)
```

Sets all the ConditionParameter objects.

Parameters: c - Array of ConditionParameter objects.

setDispColumnNames(String[])

```
public void setDispColumnNames(java.lang.String[] cols)
```

Sets names of all the DisplayColumn objects.

Parameters: cols - Array of parameter names that are to be displayed.

setDisplayColumns(DisplayColumn[])

```
public void setDisplayColumns(oracle.apps.qot.perzquery.DisplayColumn[]  
dcols)
```

Sets all the DisplayColumn objects.

Parameters: dcols - Array of DisplayColumn objects.

setDisplayRows(int)

```
public void setDisplayRows(int rows)
```

Sets the number of rows to display (batch size for the query). Maximum number of rows returned from a query at a time.

Parameters: rows - Number of rows to display (batch size for the query).

setOrdByParamNames(String[])

```
public void setOrdByParamNames(java.lang.String[] obParamNames)
```

Sets all the names of the OrderByParameter objects. Order by sequence is defaulted to true (ascending).

Parameters: obParamNames - Array of sort parameter names.

setOrderByParameters(OrderByParameter[])

```
public void
```

```
setOrderByParameters(oracle.apps.got.perzquery.OrderByParameter[] obp)
```

Sets all the OrderByParameter objects.

Parameters: obp - Array of OrderByParameter objects.

toString()

```
public java.lang.String toString()
```

Converts SearchInfo object into a String representation.

Overrides: toString in class Object

Returns: A String representation of SearchInfo object.

oracle.apps.aso.quote

This section lists the Oracle Quoting Java APIs in the package oracle.apps.aso.quote.

5.1 Package oracle.apps.aso.quote

The package oracle.apps.aso.quote contains the APIs for Oracle Quoting quote procedures. The table below lists a description for each class:

Table 5–1 Class Summary

Class	Description
Class ApprovalInstanceRec	ApprovalInstanceRec is used to model an instance of a quote approval.
Class ApproversListRec	ApproversListRec is used to model the details regarding an approver and the approver's response.
Class AtpRecord	AtpRecord class is used to store ATP (Available-To-Promise) related information.
Class AvailServiceRec	AvailServiceRec is used as an input parameter when retrieving the available services for an inventory item.
Class ControlRecord	ControlRecord is used as an input when creating and updating a quote.
Class CopyQuoteControlRecord	CopyQuoteControlRecord is used as an input when copying a quote.
Class CopyQuoteHeaderRecord	CopyQuoteHeaderRecord is used as an input when copying a quote.
Class HeaderRecord	HeaderRecord contains the columns in ASO_QUOTE_HEADERS_ALL, which stores quote header information.
Class InstanceRecord	InstanceRecord is used for Install Base integration.

Table 5–1 Class Summary

Class	Description
Class LineDetailRecord	LineDetailRecord contains the columns in ASO_QUOTE_LINE_DETAILS, which stores quote line detail information.
Class LineRecord	LineRecord contains the columns in ASO_QUOTE_LINES_ALL, which stores quote line information.
Class LineRelationshipRecord	LineRelationshipRecord contains the columns in ASO_LINE_RELATIONSHIPS, which stores information for relationships between quote lines.
Class OppQteInRec	OppQteInRec is used as an input parameter when creating a quote from opportunity.
Class OppQteOutRec	OppQteOutRec is used as an output parameter when creating a quote from an opportunity.
Class OrderHeaderRecord	OrderHeaderRecord contains information for an order.
Class OrderServiceRec	OrderServiceRec is used as an output parameter when retrieving the list of available services for an inventory item.
Class PaymentRecord	PaymentRecord contains the columns in ASO_PAYMENTS, which stores payment information for quote headers and quote lines.
Class PriceAdjustmentRecord	PriceAdjustmentRecord contains the columns in ASO_PRICE_ADJUSTMENTS, which stores price adjustment information for quote headers and quote lines.
Class PriceAttributeRecord	PriceAttributeRecord contains the columns in ASO_PRICE_ATTRIBUTES, which stores price attribute information for quote headers and quote lines.
Class QuoteAccessRecord	QuoteAccessRecord contains the columns in ASO_QUOTE_ACCESSES, as well as additional information for quote sales team.
Class RelatedObjectRecord	RelatedObjectRecord is used to model relationships between a quote and another object, such as opportunity, order, or contract.
Class RulesListRec	RulesListRec contains the information regarding the rules which determined the approvers needed for a quote.
Class SalesCreditRecord	SalesCreditRecord contains the columns in ASO_SALES_CREDITS, which stores sales credit information for quote headers and quote lines.

Table 5–1 Class Summary

Class	Description
Class ShipmentRecord	ShipmentRecord contains the columns in ASO_SHIPMENTS, which stores shipment information for quote headers and quote lines.
Class SubmitControlRecord	SubmitControlRecord is used as an input parameter when submitting a quote to convert it into an order.
Class TaxDetailRecord	TaxDetailRecord contains the columns in ASO_TAX_DETAILS, which stores tax detail information for quote headers and quote lines.
Class WarrantyRec	WarrantyRec is used as an output parameter when retrieving the included warranties for an inventory item.

5.2 Class ApprovalInstanceRec

```

java.lang.Object
|
+--oracle.apps.aso.quote.ApprovalInstanceRec

```

public class **ApprovalInstanceRec**

ApprovalInstanceRec is used to model an instance of a quote approval.

Table 5–2 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

5.2.1 Fields for Class ApprovalInstanceRec

approval_instance_id

public java.math.BigDecimal **approval_instance_id**

Approval instance ID.

approval_status

public java.lang.String **approval_status**

Approval status.

end_date

public java.sql.Timestamp **end_date**
End date of the approval.

object_approval_id

public java.math.BigDecimal **object_approval_id**
Object approval ID.

object_id

public java.math.BigDecimal **object_id**
Object ID for which the approval was initiated. For example, quote header ID in the case of quote approval.

object_type

public java.lang.String **object_type**
Object type of the object for which the approval was initiated.

quote_version

public java.math.BigDecimal **quote_version**
Version of the quote for which the approval was initiated.

RCS_ID

public static final java.lang.String **RCS_ID**

requester_comments

public java.lang.String **requester_comments**
Comments entered by the person who requested the approval.

requester_name

public java.lang.String **requester_name**
Name of the person who requested the approval.

requester_userid

public java.math.BigDecimal **requester_userid**
User ID of the person who requested the approval.

start_date

public java.sql.Timestamp **start_date**

Start date of the approval.

5.2.2 Constructors for Class ApprovalInstanceRec

ApprovalInstanceRec()

```
public ApprovalInstanceRec()
```

Default constructor.

ApprovalInstanceRec(boolean)

```
public ApprovalInstanceRec(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: `__RosettaUseGMISSValues` - Whether to default all values to Rosetta G_MISS values.

5.2.3 Methods for Class ApprovalInstanceRec

The following table is an index of the Class ApprovalInstanceRec methods:

Table 5–3 Methods for Class ApprovalInstanceRec

Method	Description
<code>toString()</code>	Returns a String representation of an ApprovalInstanceRec. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns a String representation of an ApprovalInstanceRec.

Overrides: `toString` in class `Object`

Returns: A String representation of an ApprovalInstanceRec.

5.3 Class ApproversListRec

```
java.lang.Object
|
+--oracle.apps.aso.quote.ApproversListRec
```

public class ApproversListRec

ApproversListRec is used to model the details regarding an approver and the approver's response.

Table 5–4 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

5.3.1 Fields for Class ApproversListRec

approval_comments

```
public java.lang.String approval_comments
```

Comments entered by the approver.

approval_det_id

```
public java.math.BigDecimal approval_det_id
```

Approval detail ID.

approver_name

```
public java.lang.String approver_name
```

Name of the approver.

approver_person_id

```
public java.math.BigDecimal approver_person_id
```

Person ID of the approver.

approver_sequence

```
public java.math.BigDecimal approver_sequence
```

Sequence of the approver in the list of approvers.

approver_status

```
public java.lang.String approver_status
```

Status for the approver.

approver_user_id

```
public java.math.BigDecimal approver_user_id
```

User ID of the approver.

date_recieved

```
public java.sql.Timestamp date_received
```

Date the approval notification response was received.

date_sent

```
public java.sql.Timestamp date_sent
```

Date the approval notification was sent.

notification_id

```
public java.math.BigDecimal notification_id
```

ID of the approval notification.

object_approval_id

```
public java.math.BigDecimal object_approval_id
```

Object approval ID.

RCS_ID

```
public static final java.lang.String RCS_ID
```

5.3.2 Constructors for Class ApproversListRec

ApproversListRec()

```
public ApproversListRec()
```

Default Constructor.

ApproversListRec(boolean)

```
public ApproversListRec(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: `__RosettaUseGMISSValues` - Whether to default all values to Rosetta G_MISS values.

5.3.3 Methods for Class `ApproversListRec`

The following table is an index of the Class `ApproversListRec` methods:

Table 5–5 *Methods for Class `ApproversListRec`*

Method	Description
<code>toString()</code>	Returns a String representation of <code>ApproversListRec</code> object. <code>public java.lang.String toString()</code>

toString()

```
public java.lang.String toString()
```

Returns a String representation of `ApproversListRec` object.

Overrides: `toString` in class `Object`

Returns: A String representation of `ApproversListRec` object.

5.4 Class `AtpRecord`

```
java.lang.Object
|
+--oracle.apps.aso.quote.AtpRecord
```

```
public class AtpRecord
```

`AtpRecord` class is used to store ATP (Available-To-Promise) related information. It is passed as an output parameter for check availability API `AsoAtpInt.checkAtp()`.

5.4.1 Fields for Class `AtpRecord`

ATP_ITEM_AVAILABLE

```
public static final int ATP_ITEM_AVAILABLE
```

Constant indicating that the item is currently available, i.e. in stock, with the requested item quantity.

ATP_ITEM_NOT_APPLICABLE

```
public static final int ATP_ITEM_NOT_APPLICABLE
```

Constant indicating that the ATP information for this item is not applicable, e.g. ATP information is not set up in the system.

ATP_ITEM_UNAVAILABLE

```
public static final int ATP_ITEM_UNAVAILABLE
```

Constant indicating that the item is currently not available with the requested item quantity.

available_quantity

```
public java.math.BigDecimal available_quantity
```

Indicates the quantity available on the actual ship date. This field should be populated by the check availability API after the API returns.

available_status

```
public int available_status
```

Availability status based on the status value returned by check availability process.

customer_id

```
public java.math.BigDecimal customer_id
```

Indicates the customer party ID. This field should be populated by the check availability API after the API returns.

customer_site_id

```
public java.math.BigDecimal customer_site_id
```

Indicates the customer party site ID. This field should be populated by the check availability API after the API returns.

error_code

```
public java.math.BigDecimal error_code
```

Indicates the ATP check status code. This field should be populated by the check availability API after the API returns. Use **API `getAvailableStatus()`** to obtain an interpretation of the status code indicated by this field.

identifier

```
public java.math.BigDecimal identifier
```

Indicates the identifier. This field should be populated by the check availability API after the API returns.

inventory_item_id

```
public java.math.BigDecimal inventory_item_id
```

Indicates the inventory item ID. This field should be populated by the check availability API after the API returns.

inventory_item_name

```
public java.lang.String inventory_item_name
```

Indicates the inventory item name or part number. This field should be populated by the check availability API. after the API returns.

message

```
public java.lang.String message
```

Indicates the ATP check status message. This field should be populated by the check availability API after the API returns.

quantity_ordered

```
public java.math.BigDecimal quantity_ordered
```

Indicates the requested item quantity. This field should be populated by the check availability API after the API returns.

quantity_uom

```
public java.lang.String quantity_uom
```

Indicates the unit of measure. This field should be populated by the check availability API after the API returns.

RCS_ID

```
public static final java.lang.String RCS_ID
```

request_date_quantity

```
public java.math.BigDecimal request_date_quantity
```

Indicates the item quantity on the requested ship date. This field should be populated by the check availability API after the API returns.

requested_ship_date

```
public java.sql.Timestamp requested_ship_date
```

Indicates the requested ship date, defaulted to system date. This field should be populated by the check availability API after the API returns.

ship_date

```
public java.sql.Timestamp ship_date
```

Indicates the actual ship date. If the item is currently not available, this field indicates the earliest date that item can be shipped. This field should be populated by the check availability API after the API returns.

source_organization_code

```
public java.lang.String source_organization_code
```

Indicates the source organization ID. This field should be populated by the check availability API after the API returns.

source_organization_id

```
public java.math.BigDecimal source_organization_id
```

Indicates the source organization ID. This field should be populated by the check availability API after the API returns.

5.4.2 Constructors for Class AtpRecord

AtpRecord()

```
public AtpRecord()
```

Default constructor.

AtpRecord(boolean)

```
public AtpRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.4.3 Methods for Class AtpRecord

The following table is an index of the Class AtpRecord methods:

Table 5–6 Methods for Class AtpRecord

Method	Description
getAvailableStatus()	Returns the ATP availability status based on the status value entered by the check availability process.
	<pre>public int getAvailableStatus()</pre>

Table 5–6 Methods for Class AtpRecord

Method	Description
toString()	Returns string representation of the ATP record object. <code>public java.lang.String toString()</code>

getAvailableStatus()

```
public int getAvailableStatus()
```

Returns the ATP availability status based on the status value returned by check availability process.

Returns: The ATP availability status. Possible values are:

- **ATP_ITEM_NOT_APPLICABLE:** ATP information is not applicable for the item.
- **ATP_ITEM_AVAILABLE:** The item is currently in-stock with the requested quantity.
- **ATP_ITEM_UNAVAILABLE:** The item is currently not in-stock with the requested quantity. If this is the case, the earliest ship date would be stored in `ship_date` field, and the currently available quantity would be stored in `request_date_quantity` field.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the AtpRecord object.

Overrides: toString in class Object

Returns: A String representation of the AtpRecord object.

5.5 Class AvailServiceRec

```
java.lang.Object
|
+--oracle.apps.aso.quote.AvailServiceRec
```

public class AvailServiceRec

AvailServiceRec is used as an input parameter when retrieving the available services for an inventory item.

5.5.1 Fields for Class AvailServiceRec

customer_id

```
public java.math.BigDecimal customer_id
```

Customer account ID of the customer account for which available services will be retrieved.

product_item_id

```
public java.math.BigDecimal product_item_id
```

Inventory item ID of the item for which available services will be retrieved.

product_revision

```
public java.lang.String product_revision
```

Inventory item revision of the item for which available services will be retrieved.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

request_date

```
public java.sql.Timestamp request_date
```

Request date for the service.

5.5.2 Constructors for Class AvailServiceRec

AvailServiceRec()

```
public AvailServiceRec()  
Default constructor.
```

AvailServiceRec(boolean)

```
public AvailServiceRec(boolean __RosettaUseGMISSValues)  
Constructor.
```

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.5.3 Methods for Class AvailServiceRec

The following table is an index of the Class AvailServiceRec methods:

Table 5–7 *Methods for Class AvailServiceRec*

Method	Description
toString()	String representations of the AvailServiceRec object. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the AvailServiceRec object.

Overrides: toString in class Object

Returns: A String representation of the AvailServiceRec object.

5.6 Class ControlRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.ControlRecord
```

public class ControlRecord

ControlRecord is used as an input when creating and updating a quote. The control record specifies whether certain operations should be performed when updating the quote, such as whether to calculate price, tax, and freight.

5.6.1 Fields for Class ControlRecord

auto_version_flag

```
public java.lang.String auto_version_flag
```

Whether to automatically increment the version number of the quote.

BATCH

```
public static final java.lang.String BATCH
```

Constant indicating the pricing event used to batch calculate the quote price.

calculate_freight_charge_flag

```
public java.lang.String calculate_freight_charge_flag
```

Whether to calculate freight charge.

calculate_tax_flag

```
public java.lang.String calculate_tax_flag
```

Whether to calculate tax.

copy_att_flag

```
public java.lang.String copy_att_flag
```

Whether to copy attachments.

copy_notes_flag

```
public java.lang.String copy_notes_flag
```

Whether to copy notes.

copy_task_flag

public java.lang.String **copy_task_flag**
Whether to copy tasks.

deactivate_all

public java.lang.String **deactivate_all**
Indicates whether all unchanged components of a configuration should be deactivated from the quote. This flag will only be used when calling the API which deactivates components in a configuration. In other words, the flag will be ignored when creating and updating a quote.

DEACTIVATE_ALL

public static final int **DEACTIVATE_ALL**
Constant indicating that the deactivate all flag should be set to on in the control record. Used as input to convenience constructor ControlRecord(int).

ENTIRE_QUOTE

public static final java.lang.String **ENTIRE_QUOTE**
Constant indicating that the entire quote will be priced.

FREIGHT

public static final int **FREIGHT**
Used to construct input to convenience constructor ControlRecord(int). Indicates that freight should be calculated.

functionality_code

public java.lang.String **functionality_code**
Functionality code.

header_pricing_event

public java.lang.String **header_pricing_event**
Header pricing event.

last_update_date

public java.sql.Timestamp **last_update_date**
Last update date.

line_pricing_event

```
public java.lang.String line_pricing_event
```

Line pricing event.

LIST_PRICE

```
public static final java.lang.String LIST_PRICE
```

Constant indicating the pricing event used to calculate the list price.

OFF

```
public static final int OFF
```

Used as input to convenience constructor ControlRecord(int). Indicates that price, tax, and freight should not be calculated.

PRC

```
public static final int PRC
```

Used to construct input to convenience constructor ControlRecord(int). Indicates that price should be calculated.

PRC_TAX_FREIGHT

```
public static final int PRC_TAX_FREIGHT
```

Used as input to convenience constructor ControlRecord(int). Indicates that price, tax, and freight should be calculated.

price_mode

```
public java.lang.String price_mode
```

Indicates the price mode. Possible values are:

- ENTIRE_QUOTE: Pricing will be called for the entire quote.
- QUOTE_LINE: Pricing will be called for the specified quote lines.

pricing_request_type

```
public java.lang.String pricing_request_type
```

Pricing request type.

QUOTE_LINE

```
public static final java.lang.String QUOTE_LINE
```

Constant indicating that pricing will be called only for specified quote lines.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

TAX

```
public static final int TAX
```

Constant indicating that calculate tax flag should be set to on in the control record. Used to construct input to convenience constructor ControlRecord(int).

5.6.2 Constructors for Class ControlRecord

ControlRecord()

```
public ControlRecord()
```

Default constructor.

ControlRecord(boolean)

```
public ControlRecord(boolean __RosettaUseGMISSValues)
```

Constructor. price_request_type will be set to **ASO**, price_mode will be set to **ENTIRE_QUOTE**. The remaining values will be set to Rosetta G_MISS values or to null, based on input passed in as parameter.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

ControlRecord(int)

```
public ControlRecord(int flags)
```

Constructor. Sets values in the control record for price, tax, and freight calculation based on the flags specified in the input parameter. Price_mode will be set to **ENTIRE_QUOTE**. Pricing_request_type will be set to **ASO**. auto_version_flag will be set **N**. copy_task_flag, copy_notes_flag, and copy_att_flag will be set to **Y**. last_update_date, line_pricing_event, and functionality_code will be set to G_MISS values.

Parameters:

flags - Flags indicating the values to be set in the control record for price, tax, and freight calculation. This parameter should be constructed using bitwise OR of the following possible values:

ControlRecord.OFF - Indicates that tax and freight should not be calculated. header_pricing_event should be set to null.

ControlRecord.PRC - Indicates that header_pricing_event should be set to **BATCH**.

ControlRecord.TAX - Indicates that tax should be calculated.

ControlRecord.FREIGHT - Indicates that freight should be calculated.

ControlRecord.PRC_TAX_FREIGHT - Indicates that tax and freight should be calculated. header_pricing_event should be set to **BATCH**. For example, flags = **PRC | TAX** indicates that tax should be calculated and header_pricing_event should be **BATCH**.

5.6.3 Methods for Class ControlRecord

The following table is an index of the Class ControlRecord methods:

Table 5–8 *Methods for Class ControlRecord*

Method	Description
toString()	String representations of the ControlRecord object. <code>public java.lang.String toString()</code>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the ControlRecord object.

Overrides: toString in class Object

Returns: A String representation of the ControlRecord object.

5.7 Class CopyQuoteControlRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.CopyQuoteControlRecord
public class CopyQuoteControlRecord
```

CopyQuoteControlRecord is used as an input when copying a quote. The copy quote control record specifies whether the copy of the quote is a new version of the copied quote and it specifies the information which should be copied.

5.7.1 Fields for Class CopyQuoteControlRecord

ATTCH

`public static final int ATTCH`

Used to construct input to convenience constructor `CopyQuoteControlRecord(int)`. Indicates that attachments should be copied.

copy_attachment

`public java.lang.String copy_attachment`

Whether to copy attachments.

copy_header_only

`public java.lang.String copy_header_only`

Whether to only copy quote header information. If set to `F`, quote lines will be copied.

copy_note

`public java.lang.String copy_note`

Whether to copy notes.

copy_task

`public java.lang.String copy_task`

Whether to copy tasks.

LINE

`public static final int LINE`

Used to construct input to convenience constructor `CopyQuoteControlRecord(int)`. Indicates that lines should be copied.

LINE_NOTE_TASK_ATTCH

`public static final int LINE_NOTE_TASK_ATTCH`

Used as input to convenience constructor `CopyQuoteControlRecord(int)`. Indicates that lines, notes, tasks, attachments should be copied.

new_version

`public java.lang.String new_version`

Whether the copy of the quote is a new version of the copied quote.

NOTE

```
public static final int NOTE
```

Used to construct input to convenience constructor.

CopyQuoteControlRecord(int). Indicates that notes should be copied.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

TASK

```
public static final int TASK
```

Used to construct input to convenience constructor.

CopyQuoteControlRecord(int). Indicates that tasks should be copied.

5.7.2 Constructors for Class CopyQuoteControlRecord

CopyQuoteControlRecord()

```
public CopyQuoteControlRecord()
```

Default constructor.

CopyQuoteControlRecord(boolean)

```
public CopyQuoteControlRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

CopyQuoteControlRecord(int)

```
public CopyQuoteControlRecord(int flags)
```

Constructor. Sets values in the copy quote control record for whether to copy quote lines, notes, tasks, and attachments based on the flags specified in the input parameter. new_version will be set to F.

Parameters:

flags - Flags indicating the values to be set in the copy quote control record. This parameter should be constructed using bitwise OR of the following possible values:

- CopyQuoteControlRecord.LINE - Indicates that quote lines should be copied.
- ControlRecord.NOTES - Indicates that notes should be copied.
- ControlRecord.TASK - Indicates that tasks should be copied.
- ControlRecord.ATTCH - Indicates that attachments should be copied.
- ControlRecord.NOTES_TASK_ATTCH - Indicates that notes, tasks, and attachments should be copied to the new quote. For example, flags = LINE | TASK indicates that quote lines and tasks should be copied.

5.7.3 Methods for Class CopyQuoteControlRecord

The following table is an index of the Class CopyQuoteControlRecord methods:

Table 5–9 *Methods for Class CopyQuoteControlRecord*

Method	Description
toString()	String representations of the CopyQuoteControlRecord object. <code>public java.lang.String toString()</code>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the CopyQuoteControlRecord object.

Overrides: toString in class Object

Returns: A String representation of the CopyQuoteControlRecord object.

5.8 Class CopyQuoteHeaderRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.CopyQuoteHeaderRecord
```

public class CopyQuoteHeaderRecord

CopyQuoteHeaderRecord is used as an input when copying a quote. The copy quote header record contains information for the quote to be copied as well as information which should be populated in the copy of the quote.

5.8.1 Fields for CopyQuoteHeaderRecord

price_updated_date

```
public java.sql.Timestamp price_updated_date
```

Price updated date for the copy of the quote.

pricing_status_indicator

```
public java.lang.String pricing_status_indicator
```

Pricing status indicator for the copy of the quote.

quote_expiration_date

```
public java.sql.Timestamp quote_expiration_date
```

Quote expiration date for the copy of the quote.

quote_header_id

```
public java.math.BigDecimal quote_header_id
```

Quote header ID of the quote to be copied.

quote_name

```
public java.lang.String quote_name
```

Quote name for the copy of the quote.

quote_number

```
public java.math.BigDecimal quote_number
```

Quote number for the copy of the quote.

quote_source_code

public java.lang.String **quote_source_code**
Quote source code for the copy of the quote.

RCS_ID

public static final java.lang.String **RCS_ID**

RCS_ID_RECORDED

public static final boolean **RCS_ID_RECORDED**

resource_grp_id

public java.math.BigDecimal **resource_grp_id**
Resource group ID of the primary salesperson for the copy of the quote.

resource_id

public java.math.BigDecimal **resource_id**
Resource ID of the primary salesperson for the copy of the quote.

tax_status_indicator

public java.lang.String **tax_status_indicator**
Tax status indicator for the copy of the quote.

tax_updated_date

public java.sql.Timestamp **tax_updated_date**
Tax updated date for the copy of the quote.

5.8.2 Constructors for Class CopyQuoteHeaderRecord

CopyQuoteHeaderRecord()

public **CopyQuoteHeaderRecord**()
Default constructor.

CopyQuoteHeaderRecord(boolean)

public **CopyQuoteHeaderRecord**(boolean __RosettaUseGMISSValues)
Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.8.3 Methods for Class CopyQuoteHeaderRecord

The following table is an index of the Class CopyQuoteHeaderRecord methods:

Table 5–10 *Methods for Class CopyQuoteHeaderRecord*

Method	Description
toString()	String representations of the CopyQuoteHeaderRecord object. public java.lang.String toString()

toString()

```
public java.lang.String toString()
```

Returns a String representation of the CopyQuoteHeaderRecord object.

Overrides: toString in class Object

Returns: A String representation of the CopyQuoteHeaderRecord object.

5.9 Class HeaderRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.HeaderRecord
public class HeaderRecord
```

Direct Known Subclasses: oracle.apps.qot.core.QuoteHeader

HeaderRecord contains the columns in ASO_QUOTE_HEADERS_ALL, which stores quote header information. It also contains member variables for additional quote header information. When saving a quote, only the member variables which correspond to a column in ASO_QUOTE_HEADERS_ALL will be updated in the database.

5.9.1 Fields for Class HeaderRecord

account_number

```
public java.lang.String account_number
```

Account number of the customer account.

accounting_rule_id

```
public java.math.BigDecimal accounting_rule_id
```

Accounting rule ID. Corresponds to ASO_QUOTE_HEADERS_ALL.accounting_rule_id.

agreement_name

public java.lang.String **agreement_name**
Agreement name.

attribute_category

public java.lang.String **attribute_category**
Attribute category which defines the descriptive flexfield structure. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute_category.

attribute1

public java.lang.String **attribute1**
Descriptive flexfield attribute 1. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute1.

attribute10

public java.lang.String **attribute10**
Descriptive flexfield attribute 10. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute10.

attribute11

public java.lang.String **attribute11**
Descriptive flexfield attribute 11. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute11.

attribute12

public java.lang.String **attribute12**
Descriptive flexfield attribute 12. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute12.

attribute13

public java.lang.String **attribute13**
Descriptive flexfield attribute 13. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute13.

attribute14

public java.lang.String **attribute14**

Descriptive flexfield attribute 14. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute14.

attribute15

```
public java.lang.String attribute15
```

Descriptive flexfield attribute 15. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute15.

attribute2

```
public java.lang.String attribute2
```

Descriptive flexfield attribute 2. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute2.

attribute3

```
public java.lang.String attribute3
```

Descriptive flexfield attribute 3. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute3.

attribute4

```
public java.lang.String attribute4
```

Descriptive flexfield attribute 4. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute4.

attribute5

```
public java.lang.String attribute5
```

Descriptive flexfield attribute 5. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute5.

attribute6

```
public java.lang.String attribute6
```

Descriptive flexfield attribute 6. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute6.

attribute7

```
public java.lang.String attribute7
```

Descriptive flexfield attribute 7. Corresponds to ASO_QUOTE_HEADERS_ALL.attribute7.

attribute8

public java.lang.String **attribute8**
Descriptive flexfield attribute 8. Corresponds to ASO_QUOTE_HEADERS_
ALL.attribute8.

attribute9

public java.lang.String **attribute9**
Descriptive flexfield attribute 9. Corresponds to ASO_QUOTE_HEADERS_
ALL.attribute9.

contact_id

public java.math.BigDecimal **contact_id**
Party ID of the Sold To contact person if there is a Sold To contact.

contact_name

public java.lang.String **contact_name**
Party name of the Sold To contact person if there is a Sold To contact.

contract_approval_level

public java.lang.String **contract_approval_level**
Contract approval level. Corresponds to ASO_QUOTE_HEADERS_ALL.contract_
approval_level.

contract_id

public java.math.BigDecimal **contract_id**
Contract ID which stores price agreement ID. Corresponds to ASO_QUOTE_
HEADERS_ALL.contract_id.

contract_number

public java.lang.String **contract_number**
Number of the contract created for the quote.

contract_requester_id

public java.math.BigDecimal **contract_requester_id**
User ID of the contract requester. Corresponds to ASO_QUOTE_HEADERS_
ALL.contract_requester_id.

contract_template_id

```
public java.math.BigDecimal contract_template_id
```

Contract template ID used when creating a contract for the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.contract_template_id.

contract_template_major_ver

```
public java.math.BigDecimal contract_template_major_ver
```

Major version of the contract template used when creating a contract for the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.contract_template_major_ver.

contract_template_modifier

```
public java.lang.String contract_template_modifier
```

Modifier of the contract template used when creating a contract for the quote.

contract_template_name

```
public java.lang.String contract_template_name
```

Name of the contract template used when creating a contract for the quote.

contract_template_number

```
public java.lang.String contract_template_number
```

Number of the contract template used when creating a contract for the quote.

created_by

```
public java.math.BigDecimal created_by
```

User ID of the quote creator. Corresponds to ASO_QUOTE_HEADERS_ALL.created_by.

created_by_name

```
public java.lang.String created_by_name
```

User name of the user who created the quote.

creation_date

```
public java.sql.Timestamp creation_date
```

Quote creation date. Corresponds to ASO_QUOTE_HEADERS_ALL.creation_date.

currency_code

```
public java.lang.String currency_code
```

Currency code. Corresponds to ASO_QUOTE_HEADERS_ALL.currency_code.

cust_account_id

public java.math.BigDecimal **cust_account_id**
Customer account ID. Corresponds to ASO_QUOTE_HEADERS_ALL.cust_account_id.

cust_party_id

public java.math.BigDecimal **cust_party_id**
Party ID of the Sold To customer. Corresponds to ASO_QUOTE_HEADERS_ALL.cust_party_id.

customer_id

public java.math.BigDecimal **customer_id**
Party ID of the Sold To customer of type person or organization.

customer_name

public java.lang.String **customer_name**
Party name of the Sold To customer of type person or organization.

customer_type

public java.lang.String **customer_type**
Party type of the Sold To customer of type person or organization.

display_arithmetic_operator

public java.lang.String **display_arithmetic_operator**
Display arithmetic operator. Corresponds to ASO_QUOTE_HEADERS_ALL.display_arithmetic_operator.

employee_person_id

public java.math.BigDecimal **employee_person_id**
Employee person ID. Corresponds to ASO_QUOTE_HEADERS_ALL.employee_person_id.

exchange_rate

public java.math.BigDecimal **exchange_rate**
Exchange rate. Corresponds to ASO_QUOTE_HEADERS_ALL.exchange_rate.

exchange_rate_date

public java.sql.Timestamp **exchange_rate_date**

Exchange rate date. Corresponds to ASO_QUOTE_HEADERS_ALL.exchange_rate_date.

exchange_type_code

```
public java.lang.String exchange_type_code
```

Exchange type code. Corresponds to ASO_QUOTE_HEADERS_ALL.exchange_type_code.

ffm_request_id

```
public java.math.BigDecimal ffm_request_id
```

This member variable is currently not supported.

invoice_to_account_number

```
public java.lang.String invoice_to_account_number
```

Account number of the Invoice To customer account.

invoice_to_addr_party_type

```
public java.lang.String invoice_to_addr_party_type
```

Party type of the party to which the Invoice To address belongs.

invoice_to_address1

```
public java.lang.String invoice_to_address1
```

First line of the Invoice To address.

invoice_to_address2

```
public java.lang.String invoice_to_address2
```

Second line of the Invoice To address.

invoice_to_address3

```
public java.lang.String invoice_to_address3
```

Third line of the Invoice To address.

invoice_to_address4

```
public java.lang.String invoice_to_address4
```

Fourth line of the Invoice To address.

invoice_to_city

```
public java.lang.String invoice_to_city
```

City of the Invoice To address.

invoice_to_contact_first_name

public java.lang.String **invoice_to_contact_first_name**

This member variable is currently not supported.

invoice_to_contact_id

public java.math.BigDecimal **invoice_to_contact_id**

Party ID of the Invoice To contact person.

invoice_to_contact_last_name

public java.lang.String **invoice_to_contact_last_name**

This member variable is currently not supported.

invoice_to_contact_middle_name

public java.lang.String **invoice_to_contact_middle_name**

This member variable is currently not supported.

invoice_to_contact_name

public java.lang.String **invoice_to_contact_name**

Party name of the Invoice To contact person.

invoice_to_country

public java.lang.String **invoice_to_country**

Translated country name of the Invoice To address.

invoice_to_country_code

public java.lang.String **invoice_to_country_code**

Country code of the Invoice To address.

invoice_to_county

public java.lang.String **invoice_to_county**

County of the Invoice To address.

invoice_to_cust_account_id

public java.math.BigDecimal **invoice_to_cust_account_id**

Invoice To customer account ID. Corresponds to ASO_QUOTE_HEADERS_
ALL.invoice_to_cust_account_id.

invoice_to_cust_party_id

```
public java.math.BigDecimal invoice_to_cust_party_id
```

Party ID of the Invoice To customer. Corresponds to ASO_QUOTE_HEADERS_ALL.invoice_to_cust_party_id.

invoice_to_customer_id

```
public java.math.BigDecimal invoice_to_customer_id
```

Party ID of the Invoice To customer of type person or organization.

invoice_to_customer_name

```
public java.lang.String invoice_to_customer_name
```

Party name of the Invoice To customer of type person or organization.

invoice_to_customer_type

```
public java.lang.String invoice_to_customer_type
```

Party type of the Invoice To customer. The party type is person or organization.

invoice_to_party_id

```
public java.math.BigDecimal invoice_to_party_id
```

Invoice To party ID is populated when there is an Invoice To contact. The party is of type party relationship. Corresponds to ASO_QUOTE_HEADERS_ALL.invoice_to_party_id.

invoice_to_party_name

```
public java.lang.String invoice_to_party_name
```

This member variable is currently not supported.

invoice_to_party_site_id

```
public java.math.BigDecimal invoice_to_party_site_id
```

Invoice To party site ID for the Invoice To address. Corresponds to ASO_QUOTE_HEADERS_ALL.invoice_to_party_site_id.

invoice_to_postal_code

```
public java.lang.String invoice_to_postal_code
```

Postal code of the Invoice To address.

invoice_to_province

```
public java.lang.String invoice_to_province
```

Province of the Invoice To address.

invoice_to_relationship_code

public java.lang.String **invoice_to_relationship_code**
Party relationship code if there is an Invoice To contact.

invoice_to_relationship_id

public java.math.BigDecimal **invoice_to_relationship_id**
Party relationship ID if there is an Invoice To contact.

invoice_to_state

public java.lang.String **invoice_to_state**
State of the Invoice To address.

invoicing_rule_id

public java.math.BigDecimal **invoicing_rule_id**
Invoicing rule ID. Corresponds to ASO_QUOTE_HEADERS_ALL.invoicing_rule_id.

is_invoice_to_contact_addr

public boolean **is_invoice_to_contact_addr**
Whether the Invoice To address belongs to a contact.

is_sold_to_contact_addr

public boolean **is_sold_to_contact_addr**
Whether the Sold To address belongs to the contact.

last_update_date

public java.sql.Timestamp **last_update_date**
Date the quote was last updated. Corresponds to ASO_QUOTE_HEADERS_ALL.last_update_date.

last_update_login

public java.math.BigDecimal **last_update_login**
Login ID of the user who last updated the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.last_update_login.

last_updated_by

```
public java.math.BigDecimal last_updated_by
```

User ID of the user who last updated the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.last_updated_by.

last_updated_by_name

```
public java.lang.String last_updated_by_name
```

User name of the user who last updated the quote.

marketing_source_code

```
public java.lang.String marketing_source_code
```

This member variable is currently not supported.

marketing_source_code_for_id

```
public java.math.BigDecimal marketing_source_code_for_id
```

ID of the object with the marketing source code in the quote.

marketing_source_code_id

```
public java.math.BigDecimal marketing_source_code_id
```

Marketing source code for the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.marketing_source_code_id.

marketing_source_name

```
public java.lang.String marketing_source_name
```

Marketing source code name.

marketing_source_type

```
public java.lang.String marketing_source_type
```

Marketing source type.

max_version_flag

```
public java.lang.String max_version_flag
```

Indicates whether the quote is the maximum version. Corresponds to ASO_QUOTE_HEADERS_ALL.max_version_flag.

minisite_id

```
public java.math.BigDecimal minisite_id
```

Store minisite ID to which the quote was published. Corresponds to ASO_QUOTE_HEADERS_ALL.minisite_id.

opportunity_id

public java.math.BigDecimal **opportunity_id**
Lead ID of the opportunity associated with the quote.

opportunity_name

public java.lang.String **opportunity_name**
Name of the opportunity associated with the quote.

oppty_rel_obj_id

public java.math.BigDecimal **oppty_rel_obj_id**
Related object ID of the opportunity associated with the quote. Corresponds to ASO_QUOTE_RELATED_OBJECTS.related_object_id.

order_id

public java.math.BigDecimal **order_id**
Order header ID for a quote converted into an order. Corresponds to ASO_QUOTE_HEADERS_ALL.order_id.

order_number

public java.math.BigDecimal **order_number**
Order number for a quote converted into an order.

order_type_id

public java.math.BigDecimal **order_type_id**
Order type ID. Corresponds to ASO_QUOTE_HEADERS_ALL.order_type_id.

order_type_name

public java.lang.String **order_type_name**
This member variable is currently not supported.

ordered_date

public java.sql.Timestamp **ordered_date**
Date the quote was converted to an order. Corresponds to ASO_QUOTE_HEADERS_ALL.ordered_date.

org_contact_id

public java.math.BigDecimal **org_contact_id**
Organization contact ID. Corresponds to ASO_QUOTE_HEADERS_ALL.org_contact_id.

org_id

public java.math.BigDecimal **org_id**
Operating unit ID. Corresponds to ASO_QUOTE_HEADERS_ALL.org_id.

orig_mktg_source_code_id

public java.math.BigDecimal **orig_mktg_source_code_id**
Marketing source code from which the quote originated. Corresponds to ASO_QUOTE_HEADERS_ALL.orig_mktg_source_code_id.

original_system_reference

public java.lang.String **original_system_reference**
Original system reference. Corresponds to ASO_QUOTE_HEADERS_ALL.original_system_reference.

party_id

public java.math.BigDecimal **party_id**
Party ID of the Sold To party, which can be of type person, organization, or party relationship. The party is of type party relationship when there is a contact for the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.party_id.

party_name

public java.lang.String **party_name**
Party name of the Sold To party.

party_type

public java.lang.String **party_type**
Party type of the Sold To party.

payment_amount

public java.math.BigDecimal **payment_amount**
Payment amount. Corresponds to ASO_QUOTE_HEADERS_ALL.payment_amount.

person_first_name

public java.lang.String **person_first_name**
This member variable is currently not supported.

person_last_name

public java.lang.String **person_last_name**
This member variable is currently not supported.

person_middle_name

public java.lang.String **person_middle_name**
This member variable is currently not supported.

phone_id

public java.math.BigDecimal **phone_id**
Phone ID. Corresponds to ASO_QUOTE_HEADERS_ALL.phone_id.

price_frozen_date

public java.sql.Timestamp **price_frozen_date**
Date the quote price was frozen. Corresponds to ASO_QUOTE_HEADERS_ALL.price_frozen_date.

price_list_id

public java.math.BigDecimal **price_list_id**
Price list ID used to price the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.price_list_id.

price_list_name

public java.lang.String **price_list_name**
This member variable is currently not supported.

price_request_id

public java.math.BigDecimal **price_request_id**
Price request ID if a batch request has been submitted to price the quote and the price request has not completed. Corresponds to ASO_QUOTE_HEADERS_ALL.price_request_id.

price_updated_date

public java.sql.Timestamp **price_updated_date**

The date the price was last calculated for the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.price_updated_date.

pricing_status_indicator

```
public java.lang.String pricing_status_indicator
```

Indicates whether the pricing status of the quote is complete or incomplete. Corresponds to ASO_QUOTE_HEADERS_ALL.pricing_status_indicator.

primary_salesrep_name

```
public java.lang.String primary_salesrep_name
```

Primary sales representative name.

program_application_id

```
public java.math.BigDecimal program_application_id
```

Program application ID of last concurrent program to update the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.program_application_id.

program_id

```
public java.math.BigDecimal program_id
```

Program ID of the last concurrent program to update the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.program_id.

program_update_date

```
public java.sql.Timestamp program_update_date
```

Date the last concurrent program updated the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.program_update_date.

publish_flag

```
public java.lang.String publish_flag
```

Indicates whether the quote is published. Corresponds to ASO_QUOTE_HEADERS_ALL.publish_flag.

publish_flag_meaning

```
public java.lang.String publish_flag_meaning
```

This member variable is currently not supported.

qte_contract_id

```
public java.math.BigDecimal qte_contract_id
```

This member variable is currently not supported.

quote_category_code

`public java.lang.String quote_category_code`

Quote category code. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_category_code.

quote_description

`public java.lang.String quote_description`

Quote description. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_description.

quote_expiration_date

`public java.sql.Timestamp quote_expiration_date`

Date the quote expires. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_expiration_date.

quote_header_id

`public java.math.BigDecimal quote_header_id`

Quote header ID. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_header_id.

quote_name

`public java.lang.String quote_name`

Quote name. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_name.

quote_number

`public java.math.BigDecimal quote_number`

Quote number. Quote number and quote version uniquely identify a quote. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_number.

quote_password

`public java.lang.String quote_password`

Quote password. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_password.

quote_source_code

`public java.lang.String quote_source_code`

Source of the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_source_code.

quote_status

```
public java.lang.String quote_status
```

This member variable is currently not supported.

quote_status_code

```
public java.lang.String quote_status_code
```

This member variable is currently not supported.

quote_status_id

```
public java.math.BigDecimal quote_status_id
```

Quote status ID. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_status_id.

quote_type

```
public java.lang.String quote_type
```

Quote type. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_type.

quote_version

```
public java.math.BigDecimal quote_version
```

Quote version. Quote number and quote version uniquely identify a quote. Corresponds to ASO_QUOTE_HEADERS_ALL.quote_version.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

recalculate_flag

```
public java.lang.String recalculate_flag
```

Recalculate flag. Corresponds to ASO_QUOTE_HEADERS_ALL.recalculate_flag.

request_id

```
public java.math.BigDecimal request_id
```

Request ID. Corresponds to ASO_QUOTE_HEADERS_ALL.request_id.

resource_group_name

```
public java.lang.String resource_group_name
```

This member variable is currently not supported.

resource_grp_id

public java.math.BigDecimal **resource_grp_id**
Resource group ID of the primary sales representative. Corresponds to ASO_QUOTE_HEADERS_ALL.resource_grp_id.

resource_id

public java.math.BigDecimal **resource_id**
Resource ID of the primary sales representative. Corresponds to ASO_QUOTE_HEADERS_ALL.resouce_id.

sales_channel_code

public java.lang.String **sales_channel_code**
Sales channel code. Corresponds to ASO_QUOTE_HEADERS_ALL.sales_channel_code.

sales_channel_name

public java.lang.String **sales_channel_name**
This member variable is currently not supported.

salesrep_first_name

public java.lang.String **salesrep_first_name**
This member variable is currently not supported.

salesrep_last_name

public java.lang.String **salesrep_last_name**
This member variable is currently not supported.

sold_to_addr_party_type

public java.lang.String **sold_to_addr_party_type**
Party type of the party to which the Sold To address belongs.

sold_to_address1

public java.lang.String **sold_to_address1**
First line of the Sold To address.

sold_to_address2

public java.lang.String **sold_to_address2**
Second line of the Sold To address.

sold_to_address3

```
public java.lang.String sold_to_address3
```

Third line of the Sold To address.

sold_to_address4

```
public java.lang.String sold_to_address4
```

Fourth line of the Sold To address.

sold_to_city

```
public java.lang.String sold_to_city
```

City of the Sold To address.

sold_to_country

```
public java.lang.String sold_to_country
```

Translated country name of the Sold To address.

sold_to_country_code

```
public java.lang.String sold_to_country_code
```

Country code of the Sold To address.

sold_to_county

```
public java.lang.String sold_to_county
```

County of the Sold To address.

sold_to_party_site_id

```
public java.math.BigDecimal sold_to_party_site_id
```

Sold To party site ID of the Sold To address. Corresponds to ASO_QUOTE_HEADERS_ALL.sold_to_party_site_id.

sold_to_postal_code

```
public java.lang.String sold_to_postal_code
```

Postal code of the Sold To address.

sold_to_province

```
public java.lang.String sold_to_province
```

Province of the Sold To address.

sold_to_relationship_code

public java.lang.String **sold_to_relationship_code**
Party relationship code if there is a Sold To contact.

sold_to_relationship_id

public java.math.BigDecimal **sold_to_relationship_id**
Party relationship ID if there is a Sold To contact.

sold_to_state

public java.lang.String **sold_to_state**
State of the Sold To address.

surcharge

public java.math.BigDecimal **surcharge**
Surcharge. Corresponds to ASO_QUOTE_HEADERS_ALL.surcharge.

tax_status_indicator

public java.lang.String **tax_status_indicator**
Indicates whether the tax status of the quote is complete or incomplete.
Corresponds to ASO_QUOTE_HEADERS_ALL.tax_status_indicator.

tax_updated_date

public java.sql.Timestamp **tax_updated_date**
The date the tax was last calculated for the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.tax_updated_date.

total_adjusted_amount

public java.math.BigDecimal **total_adjusted_amount**
Total adjusted amount. Corresponds to ASO_QUOTE_HEADERS_ALL.total_adjusted_amount.

total_adjusted_percent

public java.math.BigDecimal **total_adjusted_percent**
Total adjusted percent. Corresponds to ASO_QUOTE_HEADERS_ALL.total_adjusted_percent.

total_list_price

public java.math.BigDecimal **total_list_price**

Total list price of the quote. Corresponds to ASO_QUOTE_HEADERS_ALL.total_list_price.

total_quote_price

```
public java.math.BigDecimal total_quote_price
```

Total quote price. Corresponds to ASO_QUOTE_HEADERS_ALL.total_quote_price.

total_shipping_charge

```
public java.math.BigDecimal total_shipping_charge
```

Total shipping charge. Corresponds to ASO_QUOTE_HEADERS_ALL.total_shipping_charge.

total_tax

```
public java.math.BigDecimal total_tax.
```

Total tax. Corresponds to ASO_QUOTE_HEADERS_ALL.total_tax

5.9.2 Constructors for Class HeaderRecord

HeaderRecord()

```
public HeaderRecord()
```

Default constructor.

HeaderRecord(boolean)

```
public HeaderRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.9.3 Methods for Class HeaderRecord

The following table is an index of the Class HeaderRecord methods:

Table 5–11 Methods for Class HeaderRecord

Method	Description
toString()	String representations of the HeaderRecord object. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns String representation of the HeaderRecord object.

Overrides: toString in class Object

Returns: A String representation of the HeaderRecord object.

5.10 Class InstanceRecord

```
java.lang.Object  
|  
+--oracle.apps.aso.quote.InstanceRecord
```

```
public class InstanceRecord
```

InstanceRecord is used for Install Base integration. It contains instance ID and price list ID.

5.10.1 Fields for Class InstanceRecord

instance_id

```
public java.math.BigDecimal instance_id
```

Instance ID.

price_list_id

```
public java.math.BigDecimal price_list_id
```

Price list ID.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

5.10.2 Constructors for Class InstanceRecord

InstanceRecord()

```
public InstanceRecord()
```

Default constructor

InstanceRecord(boolean)

```
public InstanceRecord(boolean __RosettaUseGMISSValues)
Constructor.
```

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.10.3 Methods for Class InstanceRecord

The following table is an index of the Class InstanceRecord methods:

Table 5–12 *Methods for Class InstanceRecord*

Method	Description
toString	String representations of the InstanceRecord object. public java.lang.String toString()

toString()

```
public java.lang.String toString()
```

Returns a String representation of the InstanceRecord object.

Overrides: toString in class Object

Returns: A String representation of the InstanceRecord object.

5.11 Class LineDetailRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.LineDetailRecord
```

```
public class LineDetailRecord
```

LineDetailRecord contains the columns in ASO_QUOTE_LINE_DETAILS, which stores quote line detail information.

5.11.1 Fields for Class LineDetailRecord

attribute_category

public java.lang.String **attribute_category**

Attribute category which defines the descriptive flexfield structure. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute_category.

attribute1

public java.lang.String **attribute1**

Descriptive flexfield attribute 1. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute1.

attribute10

public java.lang.String **attribute10**

Descriptive flexfield attribute 10. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute10.

attribute11

public java.lang.String **attribute11**

Descriptive flexfield attribute 11. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute11.

attribute12

public java.lang.String **attribute12**

Descriptive flexfield attribute 12. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute12.

attribute13

public java.lang.String **attribute13**

Descriptive flexfield attribute 13. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute13.

attribute14

public java.lang.String **attribute14**

Descriptive flexfield attribute 14. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute14.

attribute15

public java.lang.String **attribute15**
Descriptive flexfield attribute 15. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute15.

attribute2

public java.lang.String **attribute2**
Descriptive flexfield attribute 2. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute2.

attribute3

public java.lang.String **attribute3**
Descriptive flexfield attribute 3. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute3.

attribute4

public java.lang.String **attribute4**
Descriptive flexfield attribute 4. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute4.

attribute5

public java.lang.String **attribute5**
Descriptive flexfield attribute 5. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute5.

attribute6

public java.lang.String **attribute6**
Descriptive flexfield attribute 6. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute6.

attribute7

public java.lang.String **attribute7**
Descriptive flexfield attribute 7. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute7.

attribute8

public java.lang.String **attribute8**
Descriptive flexfield attribute 8. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute8.

attribute9

public java.lang.String **attribute9**
Descriptive flexfield attribute 9. Corresponds to ASO_QUOTE_LINE_DETAILS.attribute9.

bom_sort_order

public java.lang.String **bom_sort_order**
Bill of Materials sorting order for configurations. Corresponds to ASO_QUOTE_LINE_DETAILS.bom_sort_order.

change_reason_code

public java.lang.String **change_reason_code**
Reason for change. Corresponds to ASO_QUOTE_LINE_DETAILS.change_reason_code.

complete_configuration_flag

public java.lang.String **complete_configuration_flag**
Indicates whether the configuration is complete. Corresponds to ASO_QUOTE_LINE_DETAILS.complete_configuration_flag.

component_code

public java.lang.String **component_code**
Concatenation of the component item IDs in a configuration. Corresponds to ASO_QUOTE_LINE_DETAILS.component_code.

config_delta

public java.math.BigDecimal **config_delta**
Configuration delta indicating changes to the configuration. Corresponds to ASO_QUOTE_LINE_DETAILS.config_delta.

config_header_id

public java.math.BigDecimal **config_header_id**
Configuration header ID. Corresponds to ASO_QUOTE_LINE_DETAILS.config_header_id.

config_instance_name

public java.lang.String **config_instance_name**
Configuration instance name. Corresponds to ASO_QUOTE_LINE_DETAILS.config_instance_name.

config_item_id

```
public java.math.BigDecimal config_item_id
```

Configuration item ID. Corresponds to ASO_QUOTE_LINE_DETAILS.config_item_id.

config_revision_num

```
public java.math.BigDecimal config_revision_num
```

Configuration revision number. Corresponds to ASO_QUOTE_LINE_DETAILS.config_revision_num.

created_by

```
public java.math.BigDecimal created_by
```

User ID of the user who created the quote line detail. Corresponds to ASO_QUOTE_LINE_DETAILS.created_by.

creation_date

```
public java.sql.Timestamp creation_date
```

Quote line detail creation date. Corresponds to ASO_QUOTE_LINE_DETAILS.creation_date.

instance_id

```
public java.math.BigDecimal instance_id
```

Instance ID of the Install Base instance which was traded-in. Corresponds to ASO_QUOTE_LINE_DETAILS.instance_id.

last_update_date

```
public java.sql.Timestamp last_update_date
```

Date the quote line detail was last updated. Corresponds to ASO_QUOTE_LINE_DETAILS.last_update_date.

last_update_login

```
public java.math.BigDecimal last_update_login
```

Login ID of the user who last updated the quote line detail. Corresponds to ASO_QUOTE_LINE_DETAILS.last_update_login.

last_updated_by

```
public java.math.BigDecimal last_updated_by
```

User ID of the user who last updated the quote line detail. Corresponds to ASO_QUOTE_LINE_DETAILS.last_updated_by.

operation_code

```
public java.lang.String operation_code
```

Operation code used when saving line details. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

program_application_id

```
public java.math.BigDecimal program_application_id
```

Program application ID of the last concurrent program to update the quote line detail. Corresponds to ASO_QUOTE_LINE_DETAILS.program_application_id

program_id

```
public java.math.BigDecimal program_id
```

Program ID of the last concurrent program to update the quote line detail. Corresponds to ASO_QUOTE_LINE_DETAILS.program_id.

program_update_date

```
public java.sql.Timestamp program_update_date
```

Date the last concurrent program to updated the quote line detail. Corresponds to ASO_QUOTE_LINE_DETAILS.program_update_date.

qte_line_index

```
public java.math.BigDecimal qte_line_index
```

Quote line index.

quote_line_detail_id

```
public java.math.BigDecimal quote_line_detail_id
```

Quote line detail ID. Corresponds to ASO_QUOTE_LINE_DETAILS.quote_line_detail_id.

quote_line_id

```
public java.math.BigDecimal quote_line_id
```

Quote line ID. Corresponds to ASO_QUOTE_LINE_DETAILS.quote_line_id.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

ref_line_id

```
public java.math.BigDecimal ref_line_id
```

ID of the quote line reference. Corresponds to ASO_QUOTE_LINE_DETAILS.ref_line_id.

ref_type_code

```
public java.lang.String ref_type_code
```

Type of quote line reference. Corresponds to ASO_QUOTE_LINE_DETAILS.ref_type_code.

request_id

```
public java.math.BigDecimal request_id
```

Request ID. Corresponds to ASO_QUOTE_LINE_DETAILS.request_id.

return_attribute_category

```
public java.lang.String return_attribute_category
```

Attribute category which defines the descriptive flexfield structure for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute_category.

return_attribute1

```
public java.lang.String return_attribute1
```

Descriptive flexfield attribute 1 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute1.

return_attribute10

```
public java.lang.String return_attribute10
```

Descriptive flexfield attribute 10 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute10.

return_attribute11

```
public java.lang.String return_attribute11
```

Descriptive flexfield attribute 11 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute11.

return_attribute12

```
public java.lang.String return_attribute12
```

Descriptive flexfield attribute 12 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute12.

return_attribute13

public java.lang.String **return_attribute13**

Descriptive flexfield attribute 13 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute13.

return_attribute14

public java.lang.String **return_attribute14**

Descriptive flexfield attribute 14 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute14.

return_attribute15

public java.lang.String **return_attribute15**

Descriptive flexfield attribute 15 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute15.

return_attribute2

public java.lang.String **return_attribute2**

Descriptive flexfield attribute 2 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute2.

return_attribute3

public java.lang.String **return_attribute3**

Descriptive flexfield attribute 3 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute3.

return_attribute4

public java.lang.String **return_attribute4**

Descriptive flexfield attribute 4 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute4.

return_attribute5

public java.lang.String **return_attribute5**

Descriptive flexfield attribute 5 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute5.

return_attribute6

public java.lang.String **return_attribute6**

Descriptive flexfield attribute 6 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute6.

return_attribute7

```
public java.lang.String return_attribute7
```

Descriptive flexfield attribute 7 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute7.

return_attribute8

```
public java.lang.String return_attribute8
```

Descriptive flexfield attribute 8 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute8.

return_attribute9

```
public java.lang.String return_attribute9
```

Descriptive flexfield attribute 9 for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_attribute9.

return_reason_code

```
public java.lang.String return_reason_code
```

Reason for return. Corresponds to ASO_QUOTE_LINE_DETAILS.return_reason_code.

return_ref_header_id

```
public java.math.BigDecimal return_ref_header_id
```

Header ID of the return reference. Corresponds to ASO_QUOTE_LINE_DETAILS.return_ref_header_id.

return_ref_line_id

```
public java.math.BigDecimal return_ref_line_id
```

Line ID of the return reference. Corresponds to ASO_QUOTE_LINE_DETAILS.return_ref_line_id.

return_ref_type

```
public java.lang.String return_ref_type
```

Reference type for returns. Corresponds to ASO_QUOTE_LINE_DETAILS.return_ref_type.

service_coterminate_flag

```
public java.lang.String service_coterminate_flag
```

Cotermination indicator for service order lines. Corresponds to ASO_QUOTE_LINE_DETAILS.service_coterminate_flag.

service_duration

public java.math.BigDecimal **service_duration**
Service duration. Corresponds to ASO_QUOTE_LINE_DETAILS.service_duration.

service_number

public java.math.BigDecimal **service_number**
Service number. Corresponds to ASO_QUOTE_LINE_DETAILS.service_number.

service_period

public java.lang.String **service_period**
Service period. Corresponds to ASO_QUOTE_LINE_DETAILS.service_period.

service_ref_line_id

public java.math.BigDecimal **service_ref_line_id**
Line ID which the service references. If service reference type is QUOTE, quote line ID. If service reference type is ORDER, order line ID. If service reference type is CUSTOMER_PRODUCT, instance ID. Corresponds to ASO_QUOTE_LINE_DETAILS.service_ref_line_id.

service_ref_line_number

public java.math.BigDecimal **service_ref_line_number**
Order line number if the service references an order line. Corresponds to ASO_QUOTE_LINE_DETAILS.service_ref_line_number.

service_ref_option_num

public java.math.BigDecimal **service_ref_option_num**
Option number which the service references. Corresponds to ASO_QUOTE_LINE_DETAILS.service_ref_option_num.

service_ref_order_number

public java.math.BigDecimal **service_ref_order_number**
Order number if the service references an order line. Corresponds to ASO_QUOTE_LINE_DETAILS.service_ref_order_number.

service_ref_qte_line_index

public java.math.BigDecimal **service_ref_qte_line_index**
Quote line index which the service references.

service_ref_shipment_numb

```
public java.math.BigDecimal service_ref_shipment_numb
```

Shipment number which the service references. Corresponds to ASO_QUOTE_LINE_DETAILS.service_ref_shipment_numb.

service_ref_system_id

```
public java.math.BigDecimal service_ref_system_id
```

System ID which the service references. Corresponds to ASO_QUOTE_LINE_DETAILS.service_ref_system_id.

service_ref_type_code

```
public java.lang.String service_ref_type_code
```

Type of object the service references. Corresponds to ASO_QUOTE_LINE_DETAILS.service_ref_type_code.

service_unit_list_percent

```
public java.math.BigDecimal service_unit_list_percent
```

Price list percent for the service when percent-based pricing is used. Corresponds to ASO_QUOTE_LINE_DETAILS.service_unit_list_percent.

service_unit_selling_percent

```
public java.math.BigDecimal service_unit_selling_percent
```

Percent charged for the service when percent-based pricing is used. Corresponds to ASO_QUOTE_LINE_DETAILS.service_unit_selling_percent.

unit_percent_base_price

```
public java.math.BigDecimal unit_percent_base_price
```

Base price used for percent-based pricing. Corresponds to ASO_QUOTE_LINE_DETAILS.unit_percent_base_price.

valid_configuration_flag

```
public java.lang.String valid_configuration_flag
```

Indicates whether the configuration is valid. Corresponds to ASO_QUOTE_LINE_DETAILS.valid_configuration_flag.

5.11.2 Constructors for Class LineDetailRecord

LineDetailRecord()

`public LineDetailRecord()`
Default constructor.

LineDetailRecord(boolean)

`public LineDetailRecord(boolean __RosettaUseGMISSValues)`
Constructor.

Parameters: `__RosettaUseGMISSValues` - Whether to default all values to Rosetta G_MISS values.

5.11.3 Methods for Class LineDetailRecord

The following table is an index of the Class LineDetailRecord methods:

Table 5–13 *Methods for Class LineDetailRecord*

Method	Description
<code>toString</code>	String representations of the LineDetailRecord object. <code>public java.lang.String toString()</code>

toString()

`public java.lang.String toString()`
Returns String representation of the LineDetailRecord object.

Overrides: `toString` in class `Object`

Returns: String representation of the LineDetailRecord object.

5.12 Class LineRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.LineRecord
```

`public class LineRecord`

Direct Known Subclasses: `oracle.apps.qot.core.QuoteLine`

LineRecord contains the columns in ASO_QUOTE_LINES_ALL, which stores quote line information. It also contains member variables for additional quote line information. When saving a quote line, only the member variables which correspond to a column in ASO_QUOTE_LINES_ALL will be updated in the database.

5.12.1 Fields for Class LineRecord

accounting_rule_id

```
public java.math.BigDecimal accounting_rule_id
```

Accounting rule ID. Corresponds to ASO_QUOTE_LINES_ALL.accounting_rule_id.

agreement_id

```
public java.math.BigDecimal agreement_id
```

Agreement ID. Corresponds to ASO_QUOTE_LINES_ALL.agreement_id.

agreement_name

```
public java.lang.String agreement_name
```

Agreement name.

attribute_category

```
public java.lang.String attribute_category
```

Attribute category which defines the descriptive flexfield structure. Corresponds to ASO_QUOTE_LINES_ALL.attribute_category.

attribute1

```
public java.lang.String attribute1
```

Descriptive flexfield attribute 1. Corresponds to ASO_QUOTE_LINES_ALL.attribute1.

attribute10

```
public java.lang.String attribute10
```

Descriptive flexfield attribute 10. Corresponds to ASO_QUOTE_LINES_ALL.attribute10.

attribute11

```
public java.lang.String attribute11
```

Descriptive flexfield attribute 11. Corresponds to ASO_QUOTE_LINES_ALL.attribute11.

attribute12

```
public java.lang.String attribute12
```

Descriptive flexfield attribute 12. Corresponds to ASO_QUOTE_LINES_ALL.attribute12.

attribute13

```
public java.lang.String attribute13
```

Descriptive flexfield attribute 13. Corresponds to ASO_QUOTE_LINES_ALL.attribute13.

attribute14

```
public java.lang.String attribute14
```

Descriptive flexfield attribute 14. Corresponds to ASO_QUOTE_LINES_ALL.attribute14.

attribute15

```
public java.lang.String attribute15
```

Descriptive flexfield attribute 15. Corresponds to ASO_QUOTE_LINES_ALL.attribute15.

attribute2

```
public java.lang.String attribute2
```

Descriptive flexfield attribute 2. Corresponds to ASO_QUOTE_LINES_ALL.attribute2.

attribute3

```
public java.lang.String attribute3
```

Descriptive flexfield attribute 3. Corresponds to ASO_QUOTE_LINES_ALL.attribute3.

attribute4

```
public java.lang.String attribute4
```

Descriptive flexfield attribute 4. Corresponds to ASO_QUOTE_LINES_ALL.attribute4.

attribute5

public java.lang.String **attribute5**
Descriptive flexfield attribute 5. Corresponds to ASO_QUOTE_LINES_
ALL.attribute5.

attribute6

public java.lang.String **attribute6**
Descriptive flexfield attribute 6. Corresponds to ASO_QUOTE_LINES_
ALL.attribute6.

attribute7

public java.lang.String **attribute7**
Descriptive flexfield attribute 7. Corresponds to ASO_QUOTE_LINES_
ALL.attribute7.

attribute8

public java.lang.String **attribute8**
Descriptive flexfield attribute 8. Corresponds to ASO_QUOTE_LINES_
ALL.attribute8.

attribute9

public java.lang.String **attribute9**
Descriptive flexfield attribute 9. Corresponds to ASO_QUOTE_LINES_
ALL.attribute9.

backorder_flag

public java.lang.String **backorder_flag**
Whether the quote line is backordered. Corresponds to ASO_QUOTE_LINES_
ALL.backorder_flag.

commitment_id

public java.math.BigDecimal **commitment_id**
Commitment ID. Corresponds to ASO_QUOTE_LINES_ALL.commitment_id.

commitment_number

public java.lang.String **commitment_number**
Commitment number.

created_by

public java.math.BigDecimal **created_by**

User ID of the user who created the quote line. Corresponds to ASO_QUOTE_LINES_ALL.created_by.

creation_date

public java.sql.Timestamp **creation_date**

Date the quote line was created. Corresponds to ASO_QUOTE_LINES_ALL.creation_date.

currency_code

public java.lang.String **currency_code**

Currency code. Corresponds to ASO_QUOTE_LINES_ALL.currency_code.

display_arithmetic_operator

public java.lang.String **display_arithmetic_operator**

Display arithmetic operator. Corresponds to ASO_QUOTE_LINES_ALL.display_arithmetic_operator.

end_date_active

public java.sql.Timestamp **end_date_active**

Effective end date. Corresponds to ASO_QUOTE_LINES_ALL.end_date_active.

ffm_content_name

public java.lang.String **ffm_content_name**

This member variable is currently not supported.

ffm_content_type

public java.lang.String **ffm_content_type**

This member variable is currently not supported.

ffm_document_type

public java.lang.String **ffm_document_type**

This member variable is currently not supported.

ffm_media_id

public java.lang.String **ffm_media_id**

This member variable is currently not supported.

ffm_media_type

```
public java.lang.String ffm_media_type
```

This member variable is currently not supported.

ffm_user_note

```
public java.lang.String ffm_user_note
```

This member variable is currently not supported.

inventory_item_id

```
public java.math.BigDecimal inventory_item_id
```

Inventory item ID. Corresponds to ASO_QUOTE_LINES_ALL.inventory_item_id.

invoice_to_account_number

```
public java.lang.String invoice_to_account_number
```

Account number of the Invoice To customer account.

invoice_to_addr_party_type

```
public java.lang.String invoice_to_addr_party_type
```

Party type of the party to which the Invoice To address belongs.

invoice_to_address1

```
public java.lang.String invoice_to_address1
```

First line of the Invoice To address.

invoice_to_address2

```
public java.lang.String invoice_to_address2
```

Second line of the Invoice To address.

invoice_to_address3

```
public java.lang.String invoice_to_address3
```

Third line of the Invoice To address.

invoice_to_address4

```
public java.lang.String invoice_to_address4
```

Fourth line of the Invoice To address.

invoice_to_city

```
public java.lang.String invoice_to_city
```

City of the Invoice To address.

invoice_to_contact_id

public java.math.BigDecimal **invoice_to_contact_id**
Party ID of the Invoice To contact person.

invoice_to_contact_name

public java.lang.String **invoice_to_contact_name**
Party name of the Invoice To contact person.

invoice_to_country

public java.lang.String **invoice_to_country**
Translated country name of the Invoice To address.

invoice_to_country_code

public java.lang.String **invoice_to_country_code**
Country code of the Invoice To address.

invoice_to_county

public java.lang.String **invoice_to_county**
County of the Invoice To address.

invoice_to_cust_account_id

public java.math.BigDecimal **invoice_to_cust_account_id**
Invoice To customer account ID. Corresponds to ASO_QUOTE_LINES_
ALL.invoice_to_cust_account_id.

invoice_to_cust_party_id

public java.math.BigDecimal **invoice_to_cust_party_id**
Party ID of the invoice-to customer. Corresponds to ASO_QUOTE_LINES_
ALL.invoice_to_cust_party_id.

invoice_to_customer_id

public java.math.BigDecimal **invoice_to_customer_id**
Party ID of the Invoice To customer of type person or organization.

invoice_to_customer_name

public java.lang.String **invoice_to_customer_name**

Party name of the Invoice To customer of type person or organization.

invoice_to_customer_type

```
public java.lang.String invoice_to_customer_type
```

Party type of the Invoice To customer of type person or organization.

invoice_to_party_id

```
public java.math.BigDecimal invoice_to_party_id
```

Invoice To party ID is populated when there is an Invoice To contact. The party is of type party relationship. Corresponds to ASO_QUOTE_LINES_ALL.invoice_to_party_id

invoice_to_party_site_id

```
public java.math.BigDecimal invoice_to_party_site_id
```

Invoice To party site ID for the Invoice To address. Corresponds to ASO_QUOTE_LINES_ALL.invoice_to_party_site_id.

invoice_to_postal_code

```
public java.lang.String invoice_to_postal_code
```

Postal code of the Invoice To address.

invoice_to_province

```
public java.lang.String invoice_to_province
```

Province of the Invoice To address.

invoice_to_relationship_code

```
public java.lang.String invoice_to_relationship_code
```

Party relationship code if there is an Invoice To contact.

invoice_to_relationship_id

```
public java.math.BigDecimal invoice_to_relationship_id
```

Party relationship ID if there is an Invoice To contact.

invoice_to_state

```
public java.lang.String invoice_to_state
```

State of the Invoice To address.

invoicing_rule_id

public java.math.BigDecimal **invoicing_rule_id**
Invoicing rule ID. Corresponds to ASO_QUOTE_LINES_ALL.invoicing_rule_id.

is_invoice_to_contact_addr

public boolean **is_invoice_to_contact_addr**
Whether the Invoice To address belongs to a contact.

is_service_item

public boolean **is_service_item**
Whether the inventory item in the quote line is a service item.

is_serviceable

public boolean **is_serviceable**
Whether is the inventory item in the quote line is serviceable.

item_description

public java.lang.String **item_description**
Inventory item description.

item_part_number

public java.lang.String **item_part_number**
Inventory item part number.

item_relationship_type

public java.lang.String **item_relationship_type**
Type of relationship between the related item and the quoted item. Corresponds to ASO_QUOTE_LINES_ALL.item_relationship_type.

item_type_code

public java.lang.String **item_type_code**
Item type code. Corresponds to ASO_QUOTE_LINES_ALL.item_type_code.

last_update_date

public java.sql.Timestamp **last_update_date**
Date the quote line was last updated. Corresponds to ASO_QUOTE_LINES_ALL.last_update_date.

last_update_login

```
public java.math.BigDecimal last_update_login
```

Login ID of the user who last updated the quote line. Corresponds to ASO_QUOTE_LINES_ALL.last_update_login.

last_updated_by

```
public java.math.BigDecimal last_updated_by
```

User ID of the user who last updated the quote line. Corresponds to ASO_QUOTE_LINES_ALL.last_updated_by.

line_adjusted_amount

```
public java.math.BigDecimal line_adjusted_amount
```

Adjustment amount applied to the quote line. Corresponds to ASO_QUOTE_LINES_ALL.line_adjusted_amount.

line_adjusted_percent

```
public java.math.BigDecimal line_adjusted_percent
```

Adjustment percent applied to the quote line. Corresponds to ASO_QUOTE_LINES_ALL.line_adjusted_percent.

line_category_code

```
public java.lang.String line_category_code
```

Line category code. Corresponds to ASO_QUOTE_LINES_ALL.line_category_code.

line_list_price

```
public java.math.BigDecimal line_list_price
```

Line list price. Corresponds to ASO_QUOTE_LINES_ALL.line_list_price.

line_number

```
public java.math.BigDecimal line_number
```

Line number. Corresponds to ASO_QUOTE_LINES_ALL.line_number.

line_quote_price

```
public java.math.BigDecimal line_quote_price
```

Selling price of the quote line. Corresponds to ASO_QUOTE_LINES_ALL.line_quote_price.

line_type_source_flag

public java.lang.String **line_type_source_flag**

Flag indicating the source of the line type. Corresponds to ASO_QUOTE_LINES_ALL.line_type_source_flag.

marketing_source_code_id

public java.math.BigDecimal **marketing_source_code_id**

Marketing source code. Corresponds to ASO_QUOTE_LINES_ALL.marketing_source_code_id.

minisite_id

public java.math.BigDecimal **minisite_id**

Minisite ID. Corresponds to ASO_QUOTE_LINES_ALL.minisite_id.

operation_code

public java.lang.String **operation_code**

Operation code used when saving a quote line. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

order_line_type_id

public java.math.BigDecimal **order_line_type_id**

Order line type ID. Corresponds to ASO_QUOTE_LINES_ALL.order_line_type_id.

org_id

public java.math.BigDecimal **org_id**

Operating unit ID. Corresponds to ASO_QUOTE_LINES_ALL.org_id.

organization_id

public java.math.BigDecimal **organization_id**

Inventory organization ID. Corresponds to ASO_QUOTE_LINES_ALL.organization_id.

price_list_id

public java.math.BigDecimal **price_list_id**

Price list ID. Corresponds to ASO_QUOTE_LINES_ALL.price_list_id.

price_list_line_id

public java.math.BigDecimal **price_list_line_id**

Price list line ID. Corresponds to ASO_QUOTE_LINES_ALL.price_list_line_id.

priced_price_list_id

```
public java.math.BigDecimal priced_price_list_id
```

Priced price list ID. Corresponds to ASO_QUOTE_LINES_ALL.priced_price_list_id.

pricing_quantity_uom

```
public java.lang.String pricing_quantity_uom
```

This member variable is currently not supported.

program_application_id

```
public java.math.BigDecimal program_application_id
```

Program application ID of the last concurrent program to update the quote line. Corresponds to ASO_QUOTE_LINES_ALL.program_application_id.

program_id

```
public java.math.BigDecimal program_id
```

Program ID of the last concurrent program to update the quote line. Corresponds to ASO_QUOTE_LINES_ALL.program_id.

program_update_date

```
public java.sql.Timestamp program_update_date
```

Date the last concurrent program updates the quote line. Corresponds to ASO_QUOTE_LINES_ALL.program_update_date.

quantity

```
public java.math.BigDecimal quantity
```

Quantity. Corresponds to ASO_QUOTE_LINES_ALL.quantity.

quote_header_id

```
public java.math.BigDecimal quote_header_id
```

Quote header ID. Corresponds to ASO_QUOTE_LINES_ALL.quote_header_id.

quote_line_id

```
public java.math.BigDecimal quote_line_id
```

Quote line ID. Corresponds to ASO_QUOTE_LINES_ALL.quote_line_id.

RCS_ID

public static final java.lang.String **RCS_ID**

RCS_ID_RECORDED

public static final boolean **RCS_ID_RECORDED**

recalculate_flag

public java.lang.String **recalculate_flag**

Whether to recalculate the price of the quote line.

related_item_id

public java.math.BigDecimal **related_item_id**

Related item ID. Corresponds to ASO_QUOTE_LINES_ALL.related_item_id.

request_id

public java.math.BigDecimal **request_id**

Request ID. Corresponds to ASO_QUOTE_LINES_ALL.request_id.

rollup_adjusted_amount

public java.math.BigDecimal **rollup_adjusted_amount**

Sum of the adjusted amount for the line and child configuration lines (multiplied by quantity).

rollup_list_price

public java.math.BigDecimal **rollup_list_price**

Sum of the list price for the line and child configuration lines (multiplied by quantity).

rollup_quote_price

public java.math.BigDecimal **rollup_quote_price**

Sum of the quote price for the line and child configuration lines (multiplied by quantity).

section_id

public java.math.BigDecimal **section_id**

Section ID. Corresponds to ASO_QUOTE_LINES_ALL.secton_id.

selling_price_change

```
public java.lang.String selling_price_change
```

Whether the selling price has been changed in the quote line.

ship_quote_price

```
public java.math.BigDecimal ship_quote_price
```

Shipping charge price for the quote line.

split_shipment_flag

```
public java.lang.String split_shipment_flag
```

Whether the quote line is split into multiple shipments. Corresponds to ASO_QUOTE_LINES_ALL.split_shipment_flag.

start_date_active

```
public java.sql.Timestamp start_date_active
```

Effective start date. Corresponds to ASO_QUOTE_LINES_ALL.start_date_active.

ui_line_number

```
public java.lang.String ui_line_number
```

Line number displayed in the UI.

uom_code

```
public java.lang.String uom_code
```

Unit of measure code. Corresponds to ASO_QUOTE_LINES_ALL.uom_code.

5.12.2 Constructors for Class LineRecord

LineRecord()

```
public LineRecord()
```

Default constructor.

LineRecord(boolean)

```
public LineRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.12.3 Methods for Class LineRecord

The following table is an index of the Class LineRecord methods:

Table 5–14 *Methods for Class LineRecord*

Method	Description
toString	String representations of the LineRecord object. <code>public java.lang.String toString()</code>

toString()

```
public java.lang.String toString()
```

Returns String representation of the LineRecord object.

Overrides: toString in class Object

Returns: A String representation of the LineRecord object.

5.13 Class LineRelationshipRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.LineRelationshipRecord
```

```
public class LineRelationshipRecord
```

LineRelationshipRecord contains the columns in ASO_LINE_RELATIONSHIPS, which stores information for relationships between quote lines.

5.13.1 Fields for Class LineRelationshipRecord

created_by

```
public java.math.BigDecimal created_by
```

User ID of the user who created the line relationship. Corresponds to ASO_LINE_RELATIONSHIPS.created_by.

creation_date

```
public java.sql.Timestamp creation_date
```

Date the line relationship was created. Corresponds to ASO_LINE_RELATIONSHIPS.creation_date.

last_update_date

```
public java.sql.Timestamp last_update_date
```

Date the line relationship was last updated. Corresponds to ASO_LINE_RELATIONSHIPS.last_update_date.

last_update_login

```
public java.math.BigDecimal last_update_login
```

Login ID of the user who last updated the line relationship. Corresponds to ASO_LINE_RELATIONSHIPS.last_update_login.

last_updated_by

```
public java.math.BigDecimal last_updated_by
```

User ID of the user who last updated the line relationship. Corresponds to ASO_LINE_RELATIONSHIPS.last_updated_by.

line_relationship_id

```
public java.math.BigDecimal line_relationship_id
```

Line relationship ID. Corresponds to ASO_LINE_RELATIONSHIPS.line_relationship_id.

operation_code

```
public java.lang.String operation_code
```

Operation code used when saving line relationships. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

program_application_id

```
public java.math.BigDecimal program_application_id
```

Program application ID of the last concurrent program to update the line relationship. Corresponds to ASO_LINE_RELATIONSHIPS.program_application_id.

program_id

```
public java.math.BigDecimal program_id
```

Program ID of the last concurrent program to update the line relationship. Corresponds to ASO_LINE_RELATIONSHIPS.program_id.

program_update_date

```
public java.sql.Timestamp program_update_date
```

Date the last concurrent program updates the line relationship. Corresponds to ASO_LINE_RELATIONSHIPS.program_update_date.

qte_line_index

```
public java.math.BigDecimal qte_line_index  
Quote line index.
```

quote_line_id

```
public java.math.BigDecimal quote_line_id  
Quote line ID. Corresponds to ASO_LINE_RELATIONSHIPS.quote_line_id.
```

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

reciprocal_flag

```
public java.lang.String reciprocal_flag  
Indicates whether the relationship is reciprocal. Corresponds to ASO_LINE_RELATIONSHIPS.reciprocal_flag.
```

related_qte_line_index

```
public java.math.BigDecimal related_qte_line_index  
Related quote line index.
```

related_quote_line_id

```
public java.math.BigDecimal related_quote_line_id  
Related quote line ID. Corresponds to ASO_LINE_RELATIONSHIPS.related_quote_line_id.
```

relationship_type_code

```
public java.lang.String relationship_type_code  
Relationship type code. Corresponds to ASO_LINE_RELATIONSHIPS.relationship_type_code.
```

request_id

```
public java.math.BigDecimal request_id
```


Request ID. Corresponds to ASO_LINE_RELATIONSHIPS.request_id.

5.13.2 Constructors for Class LineRelationshipRecord

LineRelationshipRecord()

```
public LineRelationshipRecord()
Default constructor.
```

LineRelationshipRecord(boolean)

```
public LineRelationshipRecord(boolean __RosettaUseGMISSValues)
Constructor.
```

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.13.3 Methods for Class LineRelationshipRecord

The following table is an index of the Class LineRelationshipRecord methods:

Table 5–15 Methods for Class LineRelationshipRecord

Method	Description
toString	String representations of the LineRelationshipRecord object. public java.lang.String toString()

toString()

```
public java.lang.String toString()
Returns String representation of the LineRelationshipRecord object
```

Overrides: toString in class Object

Returns: String representation of the LineRelationshipRecord object.

5.14 Class OppQteInRec

```
java.lang.Object
|
+--oracle.apps.aso.quote.OppQteInRec
public class OppQteInRec
```

OppQteInRec is used as an input parameter when creating a quote from opportunity. It contains the opportunity ID of the opportunity from which the quote will be created, as well as information which should be populated in the new quote.

5.14.1 Fields for Class OppQteInRec

agreement_id

`public java.math.BigDecimal agreement_id`
Agreement ID for the new quote.

channel_code

`public java.lang.String channel_code`
Sales channel code for the new quote.

contract_template_id

`public java.math.BigDecimal contract_template_id`
Contract template ID for the new quote.

contract_template_major_ver

`public java.math.BigDecimal contract_template_major_ver`
Contract template major version for the new quote.

currency_code

`public java.lang.String currency_code`
Currency code for the new quote.

cust_account_id

`public java.math.BigDecimal cust_account_id`
Customer account ID for the new quote.

cust_party_id

`public java.math.BigDecimal cust_party_id`
Party ID of the Sold To customer.

marketing_source_code_id

`public java.math.BigDecimal marketing_source_code_id`
Marketing source code for the new quote.

opportunity_id

```
public java.math.BigDecimal opportunity_id
```

Lead ID of the opportunity from which the quote will be created.

order_type_id

```
public java.math.BigDecimal order_type_id
```

Order type ID for the new quote.

price_list_id

```
public java.math.BigDecimal price_list_id
```

Price list ID for the new quote.

price_updated_date

```
public java.sql.Timestamp price_updated_date
```

The date the price was last calculated for the quote.

pricing_status_indicator

```
public java.lang.String pricing_status_indicator
```

Indicates whether the pricing status of the quote is complete or incomplete.

quote_expiration_date

```
public java.sql.Timestamp quote_expiration_date
```

Quote expiration date for the new quote.

quote_name

```
public java.lang.String quote_name
```

Quote name for the new quote.

quote_number

```
public java.math.BigDecimal quote_number
```

Quote number for the new quote.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

resource_grp_id

public java.math.BigDecimal **resource_grp_id**
Resource group ID of the primary sales representative for the new quote.

resource_id

public java.math.BigDecimal **resource_id**
Resource ID of the primary sales representative for the new quote.

sold_to_contact_id

public java.math.BigDecimal **sold_to_contact_id**
Party ID of the Sold To contact for the new quote.

sold_to_party_site_id

public java.math.BigDecimal **sold_to_party_site_id**
Party site ID of the Sold To address for the new quote.

tax_status_indicator

public java.lang.String **tax_status_indicator**
Indicates whether the tax status of the quote is complete or incomplete.

tax_updated_date

public java.sql.Timestamp **tax_updated_date**
The date the tax was last calculated for the quote.

5.14.2 Constructors for Class OppQteInRec

OppQteInRec()

public **OppQteInRec**()
Default constructor.

OppQteInRec(boolean)

public **OppQteInRec**(boolean __RosettaUseGMISSValues)
Constructor

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.14.3 Methods for Class OppQteInRec

The following table is an index of the Class OppQteInRec methods:

Table 5–16 Methods for Class OppQteInRec

Method	Description
toString	String representations of the OppQteInRec object. <code>public java.lang.String toString()</code>

toString()

```
public java.lang.String toString()
```

Returns String representation of the OppQteInRec object.

Overrides: toString in class Object

Returns: A String representation of the OppQteInRec object.

5.15 Class OppQteOutRec

```
java.lang.Object
|
+--oracle.apps.aso.quote.OppQteOutRec
```

public class OppQteOutRec

OppQteOutRec is used as an output parameter when creating a quote from an opportunity. It contains information for the newly created quote.

5.15.1 Fields for Class OppQteOutRec

currency_code

```
public java.lang.String currency_code
```

Currency code of the new quote.

cust_account_id

```
public java.math.BigDecimal cust_account_id
```

Customer account ID of the new quote.

party_id

```
public java.math.BigDecimal party_id
```

Party ID of the new quote.

quote_header_id

```
public java.math.BigDecimal quote_header_id
```

Quote header ID of the new quote.

quote_number

```
public java.math.BigDecimal quote_number
```

Quote number of the new quote.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

related_object_id

public java.math.BigDecimal **related_object_id**
 Related object ID created which associates the opportunity to the quote.
 Corresponds to ASO_QUOTE_RELATED_OBJECTS.related_object_id.

5.15.2 Constructors for Class OppQteRecOut

OppQteOutRec()

public **OppQteOutRec**()
 Default constructor.

OppQteOutRec(boolean)

public **OppQteOutRec**(boolean __RosettaUseGMISSValues)
 Constructor

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.15.3 Methods for Class OppQteRecOut

The following table is an index of the Class OppQteRecOut methods:

Table 5–17 Methods for Class OppQteRecOut

Method	Description
toString	String representations of the OppQteRecOut object. public java.lang.String toString ()

toString()

public java.lang.String **toString**()
 Returns String representation of the OppQteOutRec object.

Overrides: toString in class Object

Returns: String representation of the OppQteOutRec object.

5.16 Class OrderHeaderRecord

java.lang.Object
 |

```
+++oracle.apps.aso.quote.OrderHeaderRecord
```

public class **OrderHeaderRecord**

OrderHeaderRecord contains information for an order. It is used as an output parameter when a quote is submitted into an order.

5.16.1 Fields for Class OrderHeaderRecord

contract_id

```
public java.math.BigDecimal contract_id  
Contract ID.
```

order_header_id

```
public java.math.BigDecimal order_header_id  
Order header ID.
```

order_number

```
public java.math.BigDecimal order_number  
Order number.
```

order_request_id

```
public java.math.BigDecimal order_request_id  
Order request ID.
```

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

status

```
public java.lang.String status  
Order status.
```


5.16.2 Constructors for Class OrderHeaderRecord

OrderHeaderRecord()

```
public OrderHeaderRecord()
```

Default constructor.

OrderHeaderRecord(boolean)

```
public OrderHeaderRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: `__RosettaUseGMISSValues` - Whether to default all values to Rosetta G_MISS values.

5.16.3 Methods for Class OrderHeaderRecord

The following table is an index of the Class OrderHeaderRecord methods:

Table 5–18 *Methods for Class OrderHeaderRecord*

Method	Description
toString	String representations of the OrderHeaderRecord object. <code>public java.lang.String toString()</code>

toString()

```
public java.lang.String toString()
```

Returns String representation of the OrderHeaderRecord object

Overrides: toString in class Object

Returns: String representation of the OrderHeaderRecord object.

5.17 Class OrderServiceRec

```
java.lang.Object
|
+--oracle.apps.aso.quote.OrderServiceRec
```

```
public class OrderServiceRec
```

OrderServiceRec is used as an output parameter when retrieving the list of available services for an inventory item. It contains the inventory item ID of the available service.

5.17.1 Fields for OrderServiceRec

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

service_item_id

```
public java.math.BigDecimal service_item_id
```

Inventory item ID of the service item.

5.17.2 Constructors for Class OrderServiceRec

OrderServiceRec()

```
public OrderServiceRec()
```

Default constructor.

OrderServiceRec(boolean)

```
public OrderServiceRec(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.17.3 Methods for Class OrderServiceRec

The following table is an index of the Class OrderServiceRec methods:

Table 5–19 *Methods for Class OrderServiceRec*

Method	Description
toString	String representations of the OrderServiceRec object. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the OrderServiceRec object.

Overrides: toString in class Object

Returns: String representation of the OrderServiceRec object.

5.18 Class PaymentRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.PaymentRecord
public class PaymentRecord
```

PaymentRecord contains the columns in ASO_PAYMENTS, which stores payment information for quote headers and quote lines.

5.18.1 Fields for Class PaymentRecord

attribute_category

```
public java.lang.String attribute_category
```

Attribute category which defines the descriptive flexfield structure. Corresponds to ASO_PAYMENTS.attribute_category

attribute1

```
public java.lang.String attribute1
```

Descriptive flexfield attribute 1. Corresponds to ASO_PAYMENTS.attribute1.

attribute10

```
public java.lang.String attribute10
```

Descriptive flexfield attribute 10. Corresponds to ASO_PAYMENTS.attribute10.

attribute11

```
public java.lang.String attribute11
```

Descriptive flexfield attribute 11. Corresponds to ASO_PAYMENTS.attribute11.

attribute12

```
public java.lang.String attribute12
```

Descriptive flexfield attribute 12. Corresponds to ASO_PAYMENTS.attribute12.

attribute13

```
public java.lang.String attribute13
```

Descriptive flexfield attribute 13. Corresponds to ASO_PAYMENTS.attribute13.

attribute14

```
public java.lang.String attribute14
```

Descriptive flexfield attribute 14. Corresponds to ASO_PAYMENTS.attribute14.

attribute15

```
public java.lang.String attribute15
```

Descriptive flexfield attribute 15. Corresponds to ASO_PAYMENTS.attribute15.

attribute2

```
public java.lang.String attribute2
```

Descriptive flexfield attribute 2. Corresponds to ASO_PAYMENTS.attribute2.

attribute3

```
public java.lang.String attribute3
```

Descriptive flexfield attribute 3. Corresponds to ASO_PAYMENTS.attribute3.

attribute4

```
public java.lang.String attribute4
```

Descriptive flexfield attribute 4. Corresponds to ASO_PAYMENTS.attribute4.

attribute5

```
public java.lang.String attribute5
```

Descriptive flexfield attribute 5. Corresponds to ASO_PAYMENTS.attribute5.

attribute6

```
public java.lang.String attribute6
```

Descriptive flexfield attribute 6. Corresponds to ASO_PAYMENTS.attribute6.

attribute7

```
public java.lang.String attribute7
```

Descriptive flexfield attribute 7. Corresponds to ASO_PAYMENTS.attribute7.

attribute8

```
public java.lang.String attribute8
```

Descriptive flexfield attribute 8. Corresponds to ASO_PAYMENTS.attribute8.

attribute9

```
public java.lang.String attribute9
```

Descriptive flexfield attribute 9. Corresponds to ASO_PAYMENTS.attribute9.

created_by

```
public java.math.BigDecimal created_by
```

User ID of the user who created the payment. Corresponds to ASO_PAYMENTS.created_by.

creation_date

```
public java.sql.Timestamp creation_date
```

Date the payment was created. Corresponds to ASO_PAYMENTS.creation_date.

credit_card_approval_code

```
public java.lang.String credit_card_approval_code
```

Credit card approval code. Corresponds to ASO_PAYMENTS.credit_card_approval_code.

credit_card_approval_date

```
public java.sql.Timestamp credit_card_approval_date
```

Credit card approval date. Corresponds to ASO_PAYMENTS.credit_card_approval_date.

credit_card_code

```
public java.lang.String credit_card_code
```

Credit card code. Corresponds to ASO_PAYMENTS.credit_card_code.

credit_card_expiration_date

```
public java.sql.Timestamp credit_card_expiration_date
```

Credit card expiration date. Corresponds to ASO_PAYMENTS.credit_card_expiration_date.

credit_card_holder_name

```
public java.lang.String credit_card_holder_name
```

Credit card holder name. Corresponds to ASO_PAYMENTS.credit_card_holder_name.

cust_po_number

public java.lang.String **cust_po_number**
Customer purchase order number. Corresponds to ASO_PAYMENTS.cust_po_number.

last_update_date

public java.sql.Timestamp **last_update_date**
Date the payment was last updated. Corresponds to ASO_PAYMENTS.last_update_date.

last_update_login

public java.math.BigDecimal **last_update_login**
Login ID of the user who last updated the payment. Corresponds to ASO_PAYMENTS.last_update_login.

last_updated_by

public java.math.BigDecimal **last_updated_by**
User ID of the user who last updated the payment. Corresponds to ASO_PAYMENTS.last_updated_by.

operation_code

public java.lang.String **operation_code**
Operation code used when saving payments. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

payment_amount

public java.math.BigDecimal **payment_amount**
Payment amount. Corresponds to ASO_PAYMENTS.payment_amount.

payment_id

public java.math.BigDecimal **payment_id**
Payment ID. Corresponds to ASO_PAYMENTS.payment_id.

payment_option

public java.lang.String **payment_option**
Payment option. Corresponds to ASO_PAYMENTS.payment_option.

payment_ref_number

```
public java.lang.String payment_ref_number
```

Reference number for the payment. For example, if payment type is CREDIT_CARD, this member variable stores the credit card number. Corresponds to ASO_PAYMENTS.payment_ref_number.

payment_term_id

```
public java.math.BigDecimal payment_term_id
```

Payment term ID. Corresponds to ASO_PAYMENTS.payment_term_id.

payment_type_code

```
public java.lang.String payment_type_code
```

Payment type code. Corresponds to ASO_PAYMENTS.payment_type_code.

program_application_id

```
public java.math.BigDecimal program_application_id
```

Program application ID of the last concurrent program to update the payment. Corresponds to ASO_PAYMENTS.program_application_id.

program_id

```
public java.math.BigDecimal program_id
```

Program ID of the last concurrent program to update the payment. Corresponds to ASO_PAYMENTS.program_id.

program_update_date

```
public java.sql.Timestamp program_update_date
```

Date the last concurrent program updates the payment. Corresponds to ASO_PAYMENTS.program_update_date.

qte_line_index

```
public java.math.BigDecimal qte_line_index
```

Quote line index.

quote_header_id

```
public java.math.BigDecimal quote_header_id
```

Quote header ID. Corresponds to ASO_PAYMENTS.quote_header_id.

quote_line_id

`public java.math.BigDecimal quote_line_id`
Quote line ID. Corresponds to ASO_PAYMENTS.quote_line_id.

quote_shipment_id

`public java.math.BigDecimal quote_shipment_id`
Quote shipment ID. Corresponds to ASO_PAYMENTS.quote_shipment_id.

RCS_ID

`public static final java.lang.String RCS_ID`

RCS_ID_RECORDED

`public static final boolean RCS_ID_RECORDED`

request_id

`public java.math.BigDecimal request_id`
Request ID. Corresponds to ASO_PAYMENTS.request_id.

shipment_index

`public java.math.BigDecimal shipment_index`
Shipment index.

5.18.2 Constructors for Class PaymentRecord

PaymentRecord()

`public PaymentRecord()`
Default constructor.

PaymentRecord(boolean)

`public PaymentRecord(boolean __RosettaUseGMISSValues)`
Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.18.3 Methods for Class PaymentRecord

The following table is an index of the Class PaymentRecord methods:

Table 5–20 Methods for Class PaymentRecord

Method	Description
toString	String representations of the PaymentRecord object. public java.lang.String toString()

toString()

```
public java.lang.String toString()
```

Returns a String representation of the PaymentRecord object.

Overrides: toString in class Object

Returns: A String representation of the PaymentRecord object.

5.19 Class PriceAdjustmentRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.PriceAdjustmentRecord
```

public class PriceAdjustmentRecord

PriceAdjustmentRecord contains the columns in ASO_PRICE_ADJUSTMENTS, which stores price adjustment information for quote headers and quote lines. It also contains member variables for additional price adjustment information. When saving price adjustments, only the member variables which correspond to columns in ASO_PRICE_ADJUSTMENTS will be updated in the database.

5.19.1 Fields for Class PriceAdjustmentRecord

accrual_conversion_rate

```
public java.math.BigDecimal accrual_conversion_rate
```

Accrual conversion rate. Corresponds to ASO_PRICE_ADJUSTMENTS.accrual_conversion_rate.

accrual_flag

```
public java.lang.String accrual_flag
```

Whether the adjustment is accrued. Corresponds to ASO_PRICE_ADJUSTMENTS.accrual_flag.

adjusted_amount

public java.math.BigDecimal **adjusted_amount**

Adjusted amount. Corresponds to ASO_PRICE_ADJUSTMENTS.adjusted_amount.

applied_flag

public java.lang.String **applied_flag**

Whether the adjustment has been applied. Corresponds to ASO_PRICE_ADJUSTMENTS.applied_flag.

arithmetic_operator

public java.lang.String **arithmetic_operator**

Indicates the type of adjustment, such as percent, amount, or new price. The corresponding numeric value for the adjustment is in operand. Corresponds to ASO_PRICE_ADJUSTMENTS.arithmetic_operator.

attribute_category

public java.lang.String **attribute_category**

Attribute category which defines the descriptive flexfield structure. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute_category.

attribute1

public java.lang.String **attribute1**

Descriptive flexfield attribute 1. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute1.

attribute10

public java.lang.String **attribute10**

Descriptive flexfield attribute 10. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute10.

attribute11

public java.lang.String **attribute11**

Descriptive flexfield attribute 11. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute11.

attribute12

public java.lang.String **attribute12**

Descriptive flexfield attribute 12. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute12.

attribute13

```
public java.lang.String attribute13
```

Descriptive flexfield attribute 13. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute13.

attribute14

```
public java.lang.String attribute14
```

Descriptive flexfield attribute 14. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute14.

attribute15

```
public java.lang.String attribute15
```

Descriptive flexfield attribute 15. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute15.

attribute2

```
public java.lang.String attribute2
```

Descriptive flexfield attribute 2. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute2.

attribute3

```
public java.lang.String attribute3
```

Descriptive flexfield attribute 3. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute3.

attribute4

```
public java.lang.String attribute4
```

Descriptive flexfield attribute 4. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute4.

attribute5

```
public java.lang.String attribute5
```

Descriptive flexfield attribute 5. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute5.

attribute6

```
public java.lang.String attribute6
```

Descriptive flexfield attribute 6. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute6.

attribute7

public java.lang.String **attribute7**
Descriptive flexfield attribute 7. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute7.

attribute8

public java.lang.String **attribute8**
Descriptive flexfield attribute 8. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute8.

attribute9

public java.lang.String **attribute9**
Descriptive flexfield attribute 9. Corresponds to ASO_PRICE_ADJUSTMENTS.attribute9.

automatic_flag

public java.lang.String **automatic_flag**
Whether the adjustment is applied automatically. Corresponds to ASO_PRICE_ADJUSTMENTS.automatic_flag.

benefit_qty

public java.math.BigDecimal **benefit_qty**
Quantity accrued. Corresponds to ASO_PRICE_ADJUSTMENTS.benefit_qty.

benefit_uom_code

public java.lang.String **benefit_uom_code**
Unit of measure for quantity accrued. Corresponds to ASO_PRICE_ADJUSTMENTS.benefit_uom_code.

change_reason_code

public java.lang.String **change_reason_code**
Reason code for changing the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.change_reason_code.

change_reason_text

public java.lang.String **change_reason_text**
Reason text for changing the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.change_reason_text.

change_sequence

```
public java.lang.String change_sequence
```

Change sequence. Corresponds to ASO_PRICE_ADJUSTMENTS.change_sequence.

charge_subtype

```
public java.lang.String charge_subtype
```

Translated meaning of the charge subtype.

charge_subtype_code

```
public java.lang.String charge_subtype_code
```

Charge subtype code. Corresponds to ASO_PRICE_ADJUSTMENTS.charge_subtype_code.

charge_type

```
public java.lang.String charge_type
```

Translated meaning of the charge type.

charge_type_code

```
public java.lang.String charge_type_code
```

Charge type code. Corresponds to ASO_PRICE_ADJUSTMENTS.charge_type_code.

cost_id

```
public java.math.BigDecimal cost_id
```

Cost ID. Corresponds to ASO_PRICE_ADJUSTMENTS.cost_id.

created_by

```
public java.math.BigDecimal created_by
```

User ID of the user who created the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.created_by.

creation_date

```
public java.sql.Timestamp creation_date
```

Date the price adjustment was created. Corresponds to ASO_PRICE_ADJUSTMENTS.creation_date.

credit_or_charge_flag

```
public java.lang.String credit_or_charge_flag
```

Credit or charge flag. Corresponds to ASO_PRICE_ADJUSTMENTS.credit_or_charge_flag.

estimated_flag

public java.lang.String **estimated_flag**

Indicates whether the charge is estimated. Corresponds to ASO_PRICE_ADJUSTMENTS.estimated_flag.

expiration_date

public java.sql.Timestamp **expiration_date**

Expiration date of the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.expiration_date.

inc_in_sales_performance

public java.lang.String **inc_in_sales_performance**

Indicates whether the charge should be included in sales performance calculation. Corresponds to ASO_PRICE_ADJUSTMENTS.inc_in_sales_performance.

include_on_returns_flag

public java.lang.String **include_on_returns_flag**

Whether to include the adjustment on returns. Corresponds to ASO_PRICE_ADJUSTMENTS.include_on_returns_flag.

invoiced_flag

public java.lang.String **invoiced_flag**

Indicates whether the charge is invoiced. Corresponds to ASO_PRICE_ADJUSTMENTS.invoiced_flag.

last_update_date

public java.sql.Timestamp **last_update_date**

Date the price adjustment was last updated. Corresponds to ASO_PRICE_ADJUSTMENTS.last_update_date.

last_update_login

public java.math.BigDecimal **last_update_login**

Login ID of the user who last updated the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.last_update_login.

last_updated_by

```
public java.math.BigDecimal last_updated_by
```

User ID of the user who last updated the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.last_updated_by.

list_header_id

```
public java.math.BigDecimal list_header_id
```

List header ID. Corresponds to ASO_PRICE_ADJUSTMENTS.list_header_id.

list_line_id

```
public java.math.BigDecimal list_line_id
```

List line ID. Corresponds to ASO_PRICE_ADJUSTMENTS.list_header_id.

list_line_no

```
public java.lang.String list_line_no
```

List line number. Corresponds to ASO_PRICE_ADJUSTMENTS.list_line_no.

list_line_type_code

```
public java.lang.String list_line_type_code
```

List line type code. Corresponds to ASO_PRICE_ADJUSTMENTS.list_line_type_code.

modified_from

```
public java.math.BigDecimal modified_from
```

List line ID of the other item discount or price break header. Corresponds to ASO_PRICE_ADJUSTMENTS.modified_from.

modified_to

```
public java.math.BigDecimal modified_to
```

List line ID of the list line which is linked to the modified_from. If modified_from is the list line ID of a price break header, then modified_to will be the list line ID of the price break line. Corresponds to ASO_PRICE_ADJUSTMENTS.modified_from.

modifier_header_description

```
public java.lang.String modifier_header_description
```

Modifier header description.

modifier_header_id

public java.math.BigDecimal **modifier_header_id**
Modifier header ID. Corresponds to ASO_PRICE_ADJUSTMENTS.modifier_header_id.

modifier_header_name

public java.lang.String **modifier_header_name**
Modifier header name.

modifier_level

public java.lang.String **modifier_level**
Translated meaning of the modifier level.

modifier_level_code

public java.lang.String **modifier_level_code**
Modifier level code. Corresponds to ASO_PRICE_ADJUSTMENTS.modifier_level_code.

modifier_line_id

public java.math.BigDecimal **modifier_line_id**
Modifier line ID. Corresponds to ASO_PRICE_ADJUSTMENTS.quote_line_id.

modifier_line_type

public java.lang.String **modifier_line_type**
Translated meaning of the modifier line type.

modifier_line_type_code

public java.lang.String **modifier_line_type_code**
Modifier line type code. Corresponds to ASO_PRICE_ADJUSTMENTS.modifier_line_type_code.

modifier_mechanism_type_code

public java.lang.String **modifier_mechanism_type_code**
Modifier mechanism type code. Corresponds to ASO_PRICE_ADJUSTMENTS.modifier_mechanism_type_code.

on_invoice_flag

public java.lang.String **on_invoice_flag**

Whether the adjustment is invoiced. Corresponds to ASO_PRICE_ADJUSTMENTS.on_invoice_flag.

operand

```
public java.math.BigDecimal operand
```

Positive or negative numeric value for the adjustment. If arithmetic operator is percent, specifies the percentage adjustment. If arithmetic operator is amount, specifies the amount of the adjustment. If arithmetic operator is new price, specifies the new price. Corresponds to ASO_PRICE_ADJUSTMENTS.operand.

operation_code

```
public java.lang.String operation_code
```

Operation code used when saving price adjustments. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

orig_sys_discount_ref

```
public java.lang.String orig_sys_discount_ref
```

Original system discount reference. Corresponds to ASO_PRICE_ADJUSTMENTS.orig_sys_discount_ref.

parent_adjustment_id

```
public java.math.BigDecimal parent_adjustment_id
```

Adjustment ID for the parent record in the case of Tax on charge. Corresponds to ASO_PRICE_ADJUSTMENTS.parent_adjustment_id.

percentage

```
public java.math.BigDecimal percentage
```

Adjustment percentage.

price_adjustment_id

```
public java.math.BigDecimal price_adjustment_id
```

Price adjustment ID. Corresponds to ASO_PRICE_ADJUSTMENTS.price_adjustment_id.

price_break_type_code

```
public java.lang.String price_break_type_code
```

Price break type code. Corresponds to ASO_PRICE_ADJUSTMENTS.price_break_type_code.

pricing_group_sequence

public java.math.BigDecimal **pricing_group_sequence**
Pricing group sequence. Corresponds to ASO_PRICE_ADJUSTMENTS.pricing_group_sequence.

pricing_phase_id

public java.math.BigDecimal **pricing_phase_id**
Pricing phase ID. Corresponds to ASO_PRICE_ADJUSTMENTS.pricing_phase_id.

print_on_invoice_flag

public java.lang.String **print_on_invoice_flag**
Whether to print the adjustment on the invoice. Corresponds to ASO_PRICE_ADJUSTMENTS.print_on_invoice_flag.

program_application_id

public java.math.BigDecimal **program_application_id**
Program application ID of the last concurrent program to update the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.program_application_id.

program_id

public java.math.BigDecimal **program_id**
Program ID of the last concurrent program to update the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.program_id.

program_update_date

public java.sql.Timestamp **program_update_date**
Date the last concurrent program updated the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.program_update_date.

proration_type_code

public java.lang.String **proration_type_code**
Proration type code. Corresponds to ASO_PRICE_ADJUSTMENTS.proration_type_code.

qte_line_index

public java.math.BigDecimal **qte_line_index**
Quote line index.

quote_header_id

```
public java.math.BigDecimal quote_header_id
```

Quote header ID. Corresponds to ASO_PRICE_ADJUSTMENTS.quote_header_id.

quote_line_id

```
public java.math.BigDecimal quote_line_id
```

Quote line ID. Corresponds to ASO_PRICE_ADJUSTMENTS.quote_line_id.

quote_shipment_id

```
public java.math.BigDecimal quote_shipment_id
```

Quote shipment ID. Corresponds to ASO_PRICE_ADJUSTMENTS.quote_shipment_id.

range_break_quantity

```
public java.math.BigDecimal range_break_quantity
```

Breakup of line quantity, used when the price break is of type range. Corresponds to ASO_PRICE_ADJUSTMENTS.range_break_quantity.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

rebate_payment_system_code

```
public java.lang.String rebate_payment_system_code
```

Rebate payment system code. Corresponds to ASO_PRICE_ADJUSTMENTS.rebate_payment_system_code.

rebate_transaction_reference

```
public java.lang.String rebate_transaction_reference
```

Rebate transaction reference. Corresponds to ASO_PRICE_ADJUSTMENTS.rebate_transaction_reference.

rebate_transaction_type_code

```
public java.lang.String rebate_transaction_type_code
```

Rebate transaction type code. Corresponds to ASO_PRICE_ADJUSTMENTS.rebate_transaction_type_code.

redeemed_date

public java.sql.Timestamp **redeemed_date**
Date the accrual was redeemed. Corresponds to ASO_PRICE_ADJUSTMENTS.redeemed_date.

redeemed_flag

public java.lang.String **redeemed_flag**
Whether the accrual was redeemed. Corresponds to ASO_PRICE_ADJUSTMENTS.redeemed_flag.

request_id

public java.math.BigDecimal **request_id**
Request ID. Corresponds to ASO_PRICE_ADJUSTMENTS.request_id.

shipment_index

public java.math.BigDecimal **shipment_index**
Shipment index.

source_system_code

public java.lang.String **source_system_code**
Source system which generated the price adjustment. Corresponds to ASO_PRICE_ADJUSTMENTS.source_system_code.

split_action_code

public java.lang.String **split_action_code**
Action required to split the charge amount in the case of a split in the line. Corresponds to ASO_PRICE_ADJUSTMENTS.split_action_code.

substitution_attribute

public java.lang.String **substitution_attribute**
Substitution attribute. Corresponds to ASO_PRICE_ADJUSTMENTS.substitution_attribute.

tax_code

public java.lang.String **tax_code**
Tax code defined for charges. Corresponds to ASO_PRICE_ADJUSTMENTS.tax_code.

tax_exempt_flag

```
public java.lang.String tax_exempt_flag
```

Indicates whether the price adjustment is exempt from tax. Corresponds to ASO_PRICE_ADJUSTMENTS.tax_exempt_flag.

tax_exempt_number

```
public java.lang.String tax_exempt_number
```

Tax exemption number. Corresponds to ASO_PRICE_ADJUSTMENTS.tax_exempt_number.

tax_exempt_reason_code

```
public java.lang.String tax_exempt_reason_code
```

Tax exemption reason code. Corresponds to ASO_PRICE_ADJUSTMENTS.tax_exempt_reason_code.

update_allowable_flag

```
public java.lang.String update_allowable_flag
```

Whether updating the adjustment is allowable. Corresponds to ASO_PRICE_ADJUSTMENTS.update_allowable_flag.

update_allowed

```
public java.lang.String update_allowed
```

Whether updating the price adjustment is allowed. Corresponds to ASO_PRICE_ADJUSTMENTS.update_allowed.

updated_flag

```
public java.lang.String updated_flag
```

Whether the adjustment has been updated. Corresponds to ASO_PRICE_ADJUSTMENTS.updated_flag.

5.19.2 Constructors for Class PriceAdjustmentRecord

PriceAdjustmentRecord()

```
public PriceAdjustmentRecord()
```

Default constructor.

PriceAdjustmentRecord(boolean)

```
public PriceAdjustmentRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.19.3 Methods for Class PriceAdjustmentRecord

The following table is an index of the Class PriceAdjustmentRecord methods:

Table 5–21 *Methods for Class PriceAdjustmentRecord*

Method	Description
toString()	String representations of the PriceAdjustmentRecord object. <code>public java.lang.String toString()</code>

toString()

`public java.lang.String toString()`

Returns String representation of the PriceAdjustmentRecord object.

Overrides: toString in class Object

Returns: A String representation of the PriceAdjustmentRecord object.

5.20 Class PriceAttributeRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.PriceAttributeRecord
```

public class PriceAttributeRecord

PriceAttributeRecord contains the columns in ASO_PRICE_ATTRIBUTES, which stores price attribute information for quote headers and quote lines. It also contains member variables for additional price attribute information. When saving price attributes, only the member variables which correspond to columns in ASO_PRICE_ATTRIBUTES will be updated in the database.

5.20.1 Fields for Class PriceAttributeRecord

attribute1

public java.lang.String **attribute1**
Descriptive flexfield attribute 1. Corresponds to ASO_PRICE_ATTRIBUTES.attribute1.

attribute10

public java.lang.String **attribute10**
Descriptive flexfield attribute 10. Corresponds to ASO_PRICE_ATTRIBUTES.attribute10.

attribute11

public java.lang.String **attribute11**
Descriptive flexfield attribute 11. Corresponds to ASO_PRICE_ATTRIBUTES.attribute11.

attribute12

public java.lang.String **attribute12**
Descriptive flexfield attribute 12. Corresponds to ASO_PRICE_ATTRIBUTES.attribute12.

attribute13

public java.lang.String **attribute13**
Descriptive flexfield attribute 13. Corresponds to ASO_PRICE_ATTRIBUTES.attribute13.

attribute14

public java.lang.String **attribute14**
Descriptive flexfield attribute 14. Corresponds to ASO_PRICE_ATTRIBUTES.attribute14.

attribute15

public java.lang.String **attribute15**
Descriptive flexfield attribute 15. Corresponds to ASO_PRICE_ATTRIBUTES.attribute15.

attribute2

```
public java.lang.String attribute2
```

Descriptive flexfield attribute 2. Corresponds to ASO_PRICE_ATTRIBUTES.attribute2.

attribute3

```
public java.lang.String attribute3
```

Descriptive flexfield attribute 3. Corresponds to ASO_PRICE_ATTRIBUTES.attribute3.

attribute4

```
public java.lang.String attribute4
```

Descriptive flexfield attribute 4. Corresponds to ASO_PRICE_ATTRIBUTES.attribute4.

attribute5

```
public java.lang.String attribute5
```

Descriptive flexfield attribute 5. Corresponds to ASO_PRICE_ATTRIBUTES.attribute5.

attribute6

```
public java.lang.String attribute6
```

Descriptive flexfield attribute 6. Corresponds to ASO_PRICE_ATTRIBUTES.attribute6.

attribute7

```
public java.lang.String attribute7
```

Descriptive flexfield attribute 7. Corresponds to ASO_PRICE_ATTRIBUTES.attribute7.

attribute8

```
public java.lang.String attribute8
```

Descriptive flexfield attribute 8. Corresponds to ASO_PRICE_ATTRIBUTES.attribute8.

attribute9

```
public java.lang.String attribute9
```

Descriptive flexfield attribute 9. Corresponds to ASO_PRICE_ATTRIBUTES.attribute9.

context

```
public java.lang.String context
```

Context which defines the descriptive flexfield structure. Corresponds to ASO_PRICE_ATTRIBUTES.context.

created_by

```
public java.math.BigDecimal created_by
```

User ID of the user who created the price attribute. Corresponds to ASO_PRICE_ATTRIBUTES.created_by.

creation_date

```
public java.sql.Timestamp creation_date
```

Date the price attribute was created. Corresponds to ASO_PRICE_ATTRIBUTES.creation_date.

flex_title

```
public java.lang.String flex_title
```

Flexfield title. Corresponds to ASO_PRICE_ATTRIBUTES.flex_title.

last_update_date

```
public java.sql.Timestamp last_update_date
```

Date the price attribute was last updated. Corresponds to ASO_PRICE_ATTRIBUTES.last_update_date.

last_update_login

```
public java.math.BigDecimal last_update_login
```

Login ID of the user who last updated by the price attribute. Corresponds to ASO_PRICE_ATTRIBUTES.login_id.

last_updated_by

```
public java.math.BigDecimal last_updated_by
```

User ID of the user who last updated the price attribute. Corresponds to ASO_PRICE_ATTRIBUTES.last_updated_by.

operation_code

```
public java.lang.String operation_code
```

Operation code used when saving price attributes. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

price_attribute_id

public java.math.BigDecimal **price_attribute_id**
Price attribute ID. Corresponds to ASO_PRICE_ATTRIBUTES.price_attribute_id.

pricing_attribute1

public java.lang.String **pricing_attribute1**
Pricing descriptive flexfield attribute 1. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute1.

pricing_attribute10

public java.lang.String **pricing_attribute10**
Pricing descriptive flexfield attribute 10. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute10.

pricing_attribute100

public java.lang.String **pricing_attribute100**
Pricing descriptive flexfield attribute 100. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute100.

pricing_attribute11

public java.lang.String **pricing_attribute11**
Pricing descriptive flexfield attribute 11. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute11.

pricing_attribute12

public java.lang.String **pricing_attribute12**
Pricing descriptive flexfield attribute 12. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute12.

pricing_attribute13

public java.lang.String **pricing_attribute13**
Pricing descriptive flexfield attribute 13. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute13.

pricing_attribute14

public java.lang.String **pricing_attribute14**
Pricing descriptive flexfield attribute 14. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute14.

pricing_attribute15

```
public java.lang.String pricing_attribute15
```

Pricing descriptive flexfield attribute 15. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute15.

pricing_attribute16

```
public java.lang.String pricing_attribute16
```

Pricing descriptive flexfield attribute 16. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute16.

pricing_attribute17

```
public java.lang.String pricing_attribute17
```

Pricing descriptive flexfield attribute 17. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute17.

pricing_attribute18

```
public java.lang.String pricing_attribute18
```

Pricing descriptive flexfield attribute 18. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute18.

pricing_attribute19

```
public java.lang.String pricing_attribute19
```

Pricing descriptive flexfield attribute 19. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute19.

pricing_attribute2

```
public java.lang.String pricing_attribute2
```

Pricing descriptive flexfield attribute 2. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute2.

pricing_attribute20

```
public java.lang.String pricing_attribute20
```

Pricing descriptive flexfield attribute 20. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute20.

pricing_attribute21

```
public java.lang.String pricing_attribute21
```

Pricing descriptive flexfield attribute 21. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute21.

pricing_attribute22

public java.lang.String **pricing_attribute22**
Pricing descriptive flexfield attribute 22. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute22.

pricing_attribute23

public java.lang.String **pricing_attribute23**
Pricing descriptive flexfield attribute 23. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute23.

pricing_attribute24

public java.lang.String **pricing_attribute24**
Pricing descriptive flexfield attribute 24. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute24.

pricing_attribute25

public java.lang.String **pricing_attribute25**
Pricing descriptive flexfield attribute 25. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute25.

pricing_attribute26

public java.lang.String **pricing_attribute26**
Pricing descriptive flexfield attribute 26. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute26.

pricing_attribute27

public java.lang.String **pricing_attribute27**
Pricing descriptive flexfield attribute 27. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute27.

pricing_attribute28

public java.lang.String **pricing_attribute28**
Pricing descriptive flexfield attribute 28. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute28.

pricing_attribute29

public java.lang.String **pricing_attribute29**
Pricing descriptive flexfield attribute 29. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute29.

pricing_attribute3

```
public java.lang.String pricing_attribute3
```

Pricing descriptive flexfield attribute 3. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute3.

pricing_attribute30

```
public java.lang.String pricing_attribute30
```

Pricing descriptive flexfield attribute 30. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute30.

pricing_attribute31

```
public java.lang.String pricing_attribute31
```

Pricing descriptive flexfield attribute 31. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute31.

pricing_attribute32

```
public java.lang.String pricing_attribute32
```

Pricing descriptive flexfield attribute 32. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute32.

pricing_attribute33

```
public java.lang.String pricing_attribute33
```

Pricing descriptive flexfield attribute 33. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute33.

pricing_attribute34

```
public java.lang.String pricing_attribute34
```

Pricing descriptive flexfield attribute 34. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute34.

pricing_attribute35

```
public java.lang.String pricing_attribute35
```

Pricing descriptive flexfield attribute 35. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute35.

pricing_attribute36

```
public java.lang.String pricing_attribute36
```

Pricing descriptive flexfield attribute 36. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute36.

pricing_attribute37

public java.lang.String **pricing_attribute37**
Pricing descriptive flexfield attribute 37. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute37.

pricing_attribute38

public java.lang.String **pricing_attribute38**
Pricing descriptive flexfield attribute 38. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute38.

pricing_attribute39

public java.lang.String **pricing_attribute39**
Pricing descriptive flexfield attribute 39. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute39.

pricing_attribute4

public java.lang.String **pricing_attribute4**
Pricing descriptive flexfield attribute 4. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute4.

pricing_attribute40

public java.lang.String **pricing_attribute40**
Pricing descriptive flexfield attribute 40. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute40.

pricing_attribute41

public java.lang.String **pricing_attribute41**
Pricing descriptive flexfield attribute 41. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute41.

pricing_attribute42

public java.lang.String **pricing_attribute42**
Pricing descriptive flexfield attribute 42. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute42.

pricing_attribute43

public java.lang.String **pricing_attribute43**
Pricing descriptive flexfield attribute 43. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute43.

pricing_attribute44

```
public java.lang.String pricing_attribute44
```

Pricing descriptive flexfield attribute 44. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute44.

pricing_attribute45

```
public java.lang.String pricing_attribute45
```

Pricing descriptive flexfield attribute 45. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute45.

pricing_attribute46

```
public java.lang.String pricing_attribute46
```

Pricing descriptive flexfield attribute 46. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute46.

pricing_attribute47

```
public java.lang.String pricing_attribute47
```

Pricing descriptive flexfield attribute 47. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute47.

pricing_attribute48

```
public java.lang.String pricing_attribute48
```

Pricing descriptive flexfield attribute 48. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute48.

pricing_attribute49

```
public java.lang.String pricing_attribute49
```

Pricing descriptive flexfield attribute 49. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute49.

pricing_attribute5

```
public java.lang.String pricing_attribute5
```

Pricing descriptive flexfield attribute 5. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute5.

pricing_attribute50

```
public java.lang.String pricing_attribute50
```

Pricing descriptive flexfield attribute 50. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute50.

pricing_attribute51

public java.lang.String **pricing_attribute51**
Pricing descriptive flexfield attribute 51. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute51.

pricing_attribute52

public java.lang.String **pricing_attribute52**
Pricing descriptive flexfield attribute 52. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute52.

pricing_attribute53

public java.lang.String **pricing_attribute53**
Pricing descriptive flexfield attribute 53. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute53.

pricing_attribute54

public java.lang.String **pricing_attribute54**
Pricing descriptive flexfield attribute 54. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute54.

pricing_attribute55

public java.lang.String **pricing_attribute55**
Pricing descriptive flexfield attribute 55. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute55.

pricing_attribute56

public java.lang.String **pricing_attribute56**
Pricing descriptive flexfield attribute 56. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute56.

pricing_attribute57

public java.lang.String **pricing_attribute57**
Pricing descriptive flexfield attribute 57. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute57.

pricing_attribute58

public java.lang.String **pricing_attribute58**
Pricing descriptive flexfield attribute 58. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute58.

pricing_attribute59

```
public java.lang.String pricing_attribute59
```

Pricing descriptive flexfield attribute 59. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute59.

pricing_attribute6

```
public java.lang.String pricing_attribute6
```

Pricing descriptive flexfield attribute 6. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute6.

pricing_attribute60

```
public java.lang.String pricing_attribute60
```

Pricing descriptive flexfield attribute 60. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute60.

pricing_attribute61

```
public java.lang.String pricing_attribute61
```

Pricing descriptive flexfield attribute 61. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute61.

pricing_attribute62

```
public java.lang.String pricing_attribute62
```

Pricing descriptive flexfield attribute 62. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute62.

pricing_attribute63

```
public java.lang.String pricing_attribute63
```

Pricing descriptive flexfield attribute 63. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute63.

pricing_attribute64

```
public java.lang.String pricing_attribute64
```

Pricing descriptive flexfield attribute 64. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute64.

pricing_attribute65

```
public java.lang.String pricing_attribute65
```

Pricing descriptive flexfield attribute 65. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute65.

pricing_attribute66

public java.lang.String **pricing_attribute66**
Pricing descriptive flexfield attribute 66. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute66.

pricing_attribute67

public java.lang.String **pricing_attribute67**
Pricing descriptive flexfield attribute 67. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute67.

pricing_attribute68

public java.lang.String **pricing_attribute68**
Pricing descriptive flexfield attribute 68. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute68.

pricing_attribute69

public java.lang.String **pricing_attribute69**
Pricing descriptive flexfield attribute 69. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute69.

pricing_attribute7

public java.lang.String **pricing_attribute7**
Pricing descriptive flexfield attribute 7. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute7.

pricing_attribute70

public java.lang.String **pricing_attribute70**
Pricing descriptive flexfield attribute 70. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute70.

pricing_attribute71

public java.lang.String **pricing_attribute71**
Pricing descriptive flexfield attribute 71. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute71.

pricing_attribute72

public java.lang.String **pricing_attribute72**
Pricing descriptive flexfield attribute 72. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute72.

pricing_attribute73

```
public java.lang.String pricing_attribute73
```

Pricing descriptive flexfield attribute 73. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute73.

pricing_attribute74

```
public java.lang.String pricing_attribute74
```

Pricing descriptive flexfield attribute 74. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute74.

pricing_attribute75

```
public java.lang.String pricing_attribute75
```

Pricing descriptive flexfield attribute 75. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute75.

pricing_attribute76

```
public java.lang.String pricing_attribute76
```

Pricing descriptive flexfield attribute 76. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute76.

pricing_attribute77

```
public java.lang.String pricing_attribute77
```

Pricing descriptive flexfield attribute 77. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute77.

pricing_attribute78

```
public java.lang.String pricing_attribute78
```

Pricing descriptive flexfield attribute 78. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute78.

pricing_attribute79

```
public java.lang.String pricing_attribute79
```

Pricing descriptive flexfield attribute 79. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute79.

pricing_attribute8

```
public java.lang.String pricing_attribute8
```

Pricing descriptive flexfield attribute 8. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute8.

pricing_attribute80

public java.lang.String **pricing_attribute80**
Pricing descriptive flexfield attribute 80. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute80.

pricing_attribute81

public java.lang.String **pricing_attribute81**
Pricing descriptive flexfield attribute 81. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute81.

pricing_attribute82

public java.lang.String **pricing_attribute82**
Pricing descriptive flexfield attribute 82. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute82.

pricing_attribute83

public java.lang.String **pricing_attribute83**
Pricing descriptive flexfield attribute 83. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute83.

pricing_attribute84

public java.lang.String **pricing_attribute84**
Pricing descriptive flexfield attribute 84. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute84.

pricing_attribute85

public java.lang.String **pricing_attribute85**
Pricing descriptive flexfield attribute 85. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute85.

pricing_attribute86

public java.lang.String **pricing_attribute86**
Pricing descriptive flexfield attribute 86. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute86.

pricing_attribute87

public java.lang.String **pricing_attribute87**
Pricing descriptive flexfield attribute 87. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute87.

pricing_attribute88

public java.lang.String **pricing_attribute88**
Pricing descriptive flexfield attribute 88. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute88.

pricing_attribute89

public java.lang.String **pricing_attribute89**
Pricing descriptive flexfield attribute 89. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute89.

pricing_attribute9

public java.lang.String **pricing_attribute9**
Pricing descriptive flexfield attribute 9. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute9.

pricing_attribute90

public java.lang.String **pricing_attribute90**
Pricing descriptive flexfield attribute 90. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute90.

pricing_attribute91

public java.lang.String **pricing_attribute91**
Pricing descriptive flexfield attribute 91. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute91.

pricing_attribute92

public java.lang.String **pricing_attribute92**
Pricing descriptive flexfield attribute 92. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute92.

pricing_attribute93

public java.lang.String **pricing_attribute93**
Pricing descriptive flexfield attribute 93. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute93.

pricing_attribute94

public java.lang.String **pricing_attribute94**
Pricing descriptive flexfield attribute 94. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute94.

pricing_attribute95

public java.lang.String **pricing_attribute95**
Pricing descriptive flexfield attribute 95. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute95.

pricing_attribute96

public java.lang.String **pricing_attribute96**
Pricing descriptive flexfield attribute 96. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute96.

pricing_attribute97

public java.lang.String **pricing_attribute97**
Pricing descriptive flexfield attribute 97. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute97.

pricing_attribute98

public java.lang.String **pricing_attribute98**
Pricing descriptive flexfield attribute 98. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute98.

pricing_attribute99

public java.lang.String **pricing_attribute99**
Pricing descriptive flexfield attribute 99. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_attribute99.

pricing_context

public java.lang.String **pricing_context**
Pricing context which defines the pricing descriptive flexfield structure. Corresponds to ASO_PRICE_ATTRIBUTES.pricing_context

program_application_id

public java.math.BigDecimal **program_application_id**
Program application ID of the last concurrent program to update the price attribute. Corresponds to ASO_PRICE_ATTRIBUTES.program_application_id

program_id

public java.math.BigDecimal **program_id**
Program ID of the last concurrent program to update the price attribute. Corresponds to ASO_PRICE_ATTRIBUTES.program_id.

program_update_date

```
public java.sql.Timestamp program_update_date
```

Date the last concurrent program to updated the price attribute. Corresponds to ASO_PRICE_ATTRIBUTES.program_update_date.

promotion_code

```
public java.lang.String promotion_code
```

Promotion code.

promotion_name

```
public java.lang.String promotion_name
```

Promotion name.

qte_line_index

```
public java.math.BigDecimal qte_line_index
```

Quote line index.

quote_header_id

```
public java.math.BigDecimal quote_header_id
```

Quote header ID. Corresponds to ASO_PRICE_ATTRIBUTES.quote_header_id.

quote_line_id

```
public java.math.BigDecimal quote_line_id
```

Quote line ID. Corresponds to ASO_PRICE_ATTRIBUTES.quote_line_id.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

request_id

```
public java.math.BigDecimal request_id
```

Request ID. Corresponds to ASO_PRICE_ATTRIBUTES.request_id.

5.20.2 Constructors for Class PriceAttributeRecord

PriceAttributeRecord()

`public PriceAttributeRecord()`
Default constructor.

PriceAttributeRecord(boolean)

`public PriceAttributeRecord(boolean __RosettaUseGMISSValues)`
Constructor.

Parameters: `__RosettaUseGMISSValues` - Whether to default all values to Rosetta G_MISS values.

5.20.3 Methods for Class PriceAttributeRecord

The following table is an index of the Class PriceAttributeRecord methods:

Table 5–22 *Methods for Class PriceAttributeRecord*

Method	Description
<code>toString()</code>	String representations of the PriceAttributeRecord object. <code>public java.lang.String toString()</code>

toString()

`public java.lang.String toString()`
Returns a String representation of the PriceAttributeRecord object.

Overrides: `toString` in class `Object`

Returns: A String representation of the PriceAttributeRecord object.

5.21 Class QuoteAccessRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.QuoteAccessRecord
```

public class **QuoteAccessRecord**

QuoteAccessRecord contains the columns in ASO_QUOTE_ACCESSES, as well as additional information for quote sales team. When saving quote sales team, only the member variables which correspond to columns in ASO_QUOTE_ACCESSES will be updated in the database.

5.21.1 Fields for Class QuoteAccessRecord

access_id

```
public java.math.BigDecimal access_id
```

Access ID. Corresponds to ASO_QUOTE_ACCESSES.access_id.

attribute_category

```
public java.lang.String attribute_category
```

Attribute category which defines the descriptive flexfield structure. Corresponds to ASO_QUOTE_ACCESSES.attribute_category.

attribute1

```
public java.lang.String attribute1
```

Descriptive flexfield attribute 1. Corresponds to ASO_QUOTE_ACCESSES.attribute1.

attribute10

```
public java.lang.String attribute10
```

Descriptive flexfield attribute 10. Corresponds to ASO_QUOTE_ACCESSES.attribute10.

attribute11

```
public java.lang.String attribute11
```

Descriptive flexfield attribute 11. Corresponds to ASO_QUOTE_ACCESSES.attribute11.

attribute12

public java.lang.String **attribute12**
Descriptive flexfield attribute 12. Corresponds to ASO_QUOTE_ACCESSSES.attribute12.

attribute13

public java.lang.String **attribute13**
Descriptive flexfield attribute 13. Corresponds to ASO_QUOTE_ACCESSSES.attribute13.

attribute14

public java.lang.String **attribute14**
Descriptive flexfield attribute 14. Corresponds to ASO_QUOTE_ACCESSSES.attribute14.

attribute15

public java.lang.String **attribute15**
Descriptive flexfield attribute 15. Corresponds to ASO_QUOTE_ACCESSSES.attribute15.

attribute2

public java.lang.String **attribute2**
Descriptive flexfield attribute 2. Corresponds to ASO_QUOTE_ACCESSSES.attribute2.

attribute3

public java.lang.String **attribute3**
Descriptive flexfield attribute 3. Corresponds to ASO_QUOTE_ACCESSSES.attribute3.

attribute4

public java.lang.String **attribute4**
Descriptive flexfield attribute 4. Corresponds to ASO_QUOTE_ACCESSSES.attribute4.

attribute5

public java.lang.String **attribute5**
Descriptive flexfield attribute 5. Corresponds to ASO_QUOTE_ACCESSSES.attribute5.

attribute6

public java.lang.String **attribute6**
Descriptive flexfield attribute 6. Corresponds to ASO_QUOTE_ACCESSSES.attribute6.

attribute7

public java.lang.String **attribute7**
Descriptive flexfield attribute 7. Corresponds to ASO_QUOTE_ACCESSSES.attribute7.

attribute8

public java.lang.String **attribute8**
Descriptive flexfield attribute 8. Corresponds to ASO_QUOTE_ACCESSSES.attribute8.

attribute9

public java.lang.String **attribute9**
Descriptive flexfield attribute 9. Corresponds to ASO_QUOTE_ACCESSSES.attribute9.

created_by

public java.math.BigDecimal **created_by**
User ID of the user who created the quote access record. Corresponds to ASO_QUOTE_ACCESSSES.created_by.

created_by_tap_flag

public java.lang.String **created_by_tap_flag**
Whether the record was created by the Territory Manager Assignment Program. Corresponds to ASO_QUOTE_ACCESSSES.created_by_tap_flag.

creation_date

public java.sql.Timestamp **creation_date**
Date the quote access record was created. Corresponds to ASO_QUOTE_ACCESSSES.creation_date.

keep_flag

public java.lang.String **keep_flag**
Whether the record will be kept by the Territory Manager Assignment Program. Corresponds to ASO_QUOTE_ACCESSSES.keep_flag.

last_update_date

public java.sql.Timestamp **last_update_date**

Date the quote access record was last updated. Corresponds to ASO_QUOTE_ACCESSSES.last_update_date.

last_update_login

public java.math.BigDecimal **last_update_login**

Login ID of the user who last updated the quote access record. Corresponds to ASO_QUOTE_ACCESSSES.last_update_login.

last_updated_by

public java.math.BigDecimal **last_updated_by**

User ID of the user who last updated the quote access record. Corresponds to ASO_QUOTE_ACCESSSES.last_updated_by.

prm_keep_flag

public java.lang.String **prm_keep_flag**

Whether the resource is the primary sales representative.

program_application_id

public java.math.BigDecimal **program_application_id**

Program application ID of the last concurrent program to update the quote access record. Corresponds to ASO_QUOTE_ACCESSSES.program_application_id.

program_id

public java.math.BigDecimal **program_id**

Program ID of the last concurrent program to update the quote access record. Corresponds to ASO_QUOTE_ACCESSSES.program_id.

program_update_date

public java.sql.Timestamp **program_update_date**

Date the last concurrent program updated the quote access record. Corresponds to ASO_QUOTE_ACCESSSES.program_update_date.

quote_number

public java.math.BigDecimal **quote_number**

Quote number. Corresponds to ASO_QUOTE_ACCESSSES.quote_number.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

request_id

```
public java.math.BigDecimal request_id
```

Request ID. Corresponds to ASO_QUOTE_ACCESSSES.request_id.

resource_city

```
public java.lang.String resource_city
```

City of the resource's address.

resource_country

```
public java.lang.String resource_country
```

Country of the resource's address.

resource_email

```
public java.lang.String resource_email
```

Resource email.

resource_grp_id

```
public java.math.BigDecimal resource_grp_id
```

Resource group ID for the resource on the quote sales team. Corresponds to ASO_QUOTE_ACCESSSES.resource_grp_id.

resource_id

```
public java.math.BigDecimal resource_id
```

Resource ID on the quote sales team. Corresponds to ASO_QUOTE_ACCESSSES.resource_id.

resource_job_title

```
public java.lang.String resource_job_title
```

Resource job title.

resource_name

```
public java.lang.String resource_name
```

Resource name

resource_phone_no

public java.lang.String **resource_phone_no**
Resource phone number

resource_postal_code

public java.lang.String **resource_postal_code**
Postal code of the resource's address.

resource_state

public java.lang.String **resource_state**
State of the resource's address.

update_access_flag

public java.lang.String **update_access_flag**
Whether the resource has update access to the quote. Corresponds to ASO_ QUOTE_ACCESES.update_access_flag.

5.21.2 Constructors for Class QuoteAccessRecord

QuoteAccessRecord()

public **QuoteAccessRecord**()
Default constructor.

QuoteAccessRecord(boolean)

public **QuoteAccessRecord**(boolean __RosettaUseGMISSValues)
Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.21.3 Methods for Class QuoteAccessRecord

The following table is an index of the Class QuoteAccessRecord methods:

Table 5–23 *Methods for Class QuoteAccessRecord*

Method	Description
toString()	String representations of the QuoteAccessRecord object. public java.lang.String toString()

toString()

```
public java.lang.String toString()
```

Returns String representation of the QuoteAccessRecord object.

Overrides: toString in class Object

Returns: String representation of the QuoteAccessRecord object.

5.22 Class RelatedObjectRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.RelatedObjectRecord
```

```
public class RelatedObjectRecord
```

RelatedObjectRecord is used to model relationships between a quote and another object, such as opportunity, order, or contract. It contains the columns in ASO_QUOTE_RELATED_OBJECTS and an additional member variable used when saving quote relationships.

5.22.1 Fields for Class RelatedObjectRecord

created_by

```
public java.math.BigDecimal created_by
```

User ID of the user who created the quote relationship. Corresponds to ASO_QUOTE_RELATED_OBJECTS.created_by.

creation_date

```
public java.sql.Timestamp creation_date
```

Date the quote relationship was created. Corresponds to ASO_QUOTE_RELATED_OBJECTS.creation_date.

last_update_date

public java.sql.Timestamp **last_update_date**

Date the quote relationship was last updated. Corresponds to ASO_QUOTE_RELATED_OBJECTS.last_update_date.

last_update_login

public java.math.BigDecimal **last_update_login**

Login ID of the user who last updated the quote relationship. Corresponds to ASO_QUOTE_RELATED_OBJECTS.last_update_login.

last_updated_by

public java.math.BigDecimal **last_updated_by**

User ID of the user who last updated the quote relationship. Corresponds to ASO_QUOTE_RELATED_OBJECTS.last_updated_by.

object_id

public java.math.BigDecimal **object_id**

Object ID of the object related to the quote. Corresponds to ASO_QUOTE_RELATED_OBJECTS.object_id.

object_type_code

public java.lang.String **object_type_code**

Object type code of the object related to the quote. Corresponds to ASO_QUOTE_RELATED_OBJECTS.object_type_code.

operation_code

public java.lang.String **operation_code**

Operation code used when saving related object information. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

program_application_id

public java.math.BigDecimal **program_application_id**

Program application ID of the last concurrent program to update the quote relationship. Corresponds to ASO_QUOTE_RELATED_OBJECTS.program_application_id.

program_id

```
public java.math.BigDecimal program_id
```

Program ID of the last concurrent program to update the quote relationship. Corresponds to ASO_QUOTE_RELATED_OBJECTS.program_id.

program_update_date

```
public java.sql.Timestamp program_update_date
```

Date the last concurrent program updated the quote relationship. Corresponds to ASO_QUOTE_RELATED_OBJECTS.program_update_date.

quote_object_code

```
public java.math.BigDecimal quote_object_code
```

This member variable is currently not supported.

quote_object_id

```
public java.math.BigDecimal quote_object_id
```

Quote header ID or quote line ID, depending on the quote object type code. Corresponds to ASO_QUOTE_RELATED_OBJECTS.quote_object_id.

quote_object_type_code

```
public java.lang.String quote_object_type_code
```

Quote object type code specifying if the relationship is for a quote header or quote line. Corresponds to ASO_QUOTE_RELATED_OBJECTS.quote_object_type_code.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

reciprocal_flag

```
public java.lang.String reciprocal_flag
```

Whether the relationship is reciprocal. Corresponds to ASO_QUOTE_RELATED_OBJECTS.reciprocal_flag.

related_object_id

```
public java.math.BigDecimal related_object_id
```

Related object ID. Corresponds to ASO_QUOTE_RELATED_OBJECTS.related_object_id.

relationship_type_code

`public java.lang.String relationship_type_code`
Relationship type code. Corresponds to ASO_QUOTE_RELATED_OBJECTS.relationship_type_code.

request_id

`public java.math.BigDecimal request_id`
Request ID. Corresponds to ASO_QUOTE_RELATED_OBJECTS.request_id.

5.22.2 Constructors for Class RelatedObjectRecord

RelatedObjectRecord()

`public RelatedObjectRecord()`
Default constructor.

RelatedObjectRecord(boolean)

`public RelatedObjectRecord(boolean __RosettaUseGMISSValues)`
Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.22.3 Methods for Class RelatedObjectRecord

The following table is an index of the Class RelatedObjectRecord methods:

Table 5–24 *Methods for Class RelatedObjectRecord*

Method	Description
<code>toString()</code>	String representations of the RelatedObjectRecord object. <code>public java.lang.String toString()</code>

toString()

`public java.lang.String toString()`
Returns String representation of the RelatedObjectRecord object.

Overrides: toString in class Object

Returns: String representation of the RelatedObjectRecord object.

5.23 Class RulesListRec

```
java.lang.Object
|
+--oracle.apps.aso.quote.RulesListRec
```

public class **RulesListRec**

RulesListRec contains the information regarding the rules which determined the approvers needed for a quote.

5.23.1 Fields for Class RulesListRec

approval_level

```
public java.lang.String approval_level
```

Approval level.

object_approval_id

```
public java.math.BigDecimal object_approval_id
```

Object approval ID for which the rules were applied.

RCS_ID

```
public static final java.lang.String RCS_ID
```

rule_action_id

```
public java.math.BigDecimal rule_action_id
```

Rule action ID.

rule_description

```
public java.lang.String rule_description
```

Rule description.

rule_id

```
public java.math.BigDecimal rule_id
```

Rule ID.

5.23.2 Constructors for Class RulesListRec

RulesListRec()

```
public RulesListRec()
```

Default constructor.

RulesListRec(boolean)

```
public RulesListRec(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: `__RosettaUseGMISSValues` - Whether to default all values to Rosetta G_MISS values.

5.23.3 Methods for Class RulesListRec

The following table is an index of the Class RulesListRec methods:

Table 5–25 *Methods for Class RulesListRec*

Method	Description
<code>toString()</code>	String representations of the RulesListRec object. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the RulesListRec object.

Overrides: `toString` in class `Object`

Returns: A String representation of the RulesListRec object.

5.24 Class SalesCreditRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.SalesCreditRecord
```

public class SalesCreditRecord

SalesCreditRecord contains the columns in ASO_SALES_CREDITS, which stores sales credit information for quote headers and quote lines. It also contains member variables for additional sales credit information. When saving sales credits, only the member variables which correspond to columns in ASO_SALES_CREDITS will be updated in the database.

5.24.1 Fields for Class SalesCreditRecord

attribute_category_code

```
public java.lang.String attribute_category_code
```

Attribute category which defines the descriptive flexfield structure. Corresponds to ASO_SALES_CREDITS.attribute_category_code.

attribute1

```
public java.lang.String attribute1
```

Descriptive flexfield attribute 1. Corresponds to ASO_SALES_CREDITS.attribute1.

attribute10

```
public java.lang.String attribute10
```

Descriptive flexfield attribute 10. Corresponds to ASO_SALES_CREDITS.attribute10.

attribute11

```
public java.lang.String attribute11
```

Descriptive flexfield attribute 11. Corresponds to ASO_SALES_CREDITS.attribute11.

attribute12

```
public java.lang.String attribute12
```

Descriptive flexfield attribute 12. Corresponds to ASO_SALES_CREDITS.attribute12.

attribute13

public java.lang.String **attribute13**
Descriptive flexfield attribute 13. Corresponds to ASO_SALES_CREDITS.attribute13.

attribute14

public java.lang.String **attribute14**
Descriptive flexfield attribute 14. Corresponds to ASO_SALES_CREDITS.attribute14.

attribute15

public java.lang.String **attribute15**
Descriptive flexfield attribute 15. Corresponds to ASO_SALES_CREDITS.attribute15.

attribute2

public java.lang.String **attribute2**
Descriptive flexfield attribute 2. Corresponds to ASO_SALES_CREDITS.attribute2.

attribute3

public java.lang.String **attribute3**
Descriptive flexfield attribute 3. Corresponds to ASO_SALES_CREDITS.attribute3.

attribute4

public java.lang.String **attribute4**
Descriptive flexfield attribute 4. Corresponds to ASO_SALES_CREDITS.attribute4.

attribute5

public java.lang.String **attribute5**
Descriptive flexfield attribute 5. Corresponds to ASO_SALES_CREDITS.attribute5.

attribute6

public java.lang.String **attribute6**
Descriptive flexfield attribute 6. Corresponds to ASO_SALES_CREDITS.attribute6.

attribute7

public java.lang.String **attribute7**
Descriptive flexfield attribute 7. Corresponds to ASO_SALES_CREDITS.attribute7.

attribute8

```
public java.lang.String attribute8
```

Descriptive flexfield attribute 8. Corresponds to ASO_SALES_CREDITS.attribute8.

attribute9

```
public java.lang.String attribute9
```

Descriptive flexfield attribute 9. Corresponds to ASO_SALES_CREDITS.attribute9.

created_by

```
public java.math.BigDecimal created_by
```

User ID of the user who created the sales credit. Corresponds to ASO_SALES_CREDITS.sales_credit_id.

creation_date

```
public java.sql.Timestamp creation_date
```

Date the sales credit was created. Corresponds to ASO_SALES_CREDITS.creation_date.

employee_person_id

```
public java.math.BigDecimal employee_person_id
```

Employee person ID. Corresponds to ASO_SALES_CREDITS.employee_person_id.

first_name

```
public java.lang.String first_name
```

This member variable is currently not supported.

last_name

```
public java.lang.String last_name
```

This member variable is currently not supported.

last_update_date

```
public java.sql.Timestamp last_update_date
```

Date the sales credit was last updated. Corresponds to ASO_SALES_CREDITS.last_update_date.

last_update_login

```
public java.math.BigDecimal last_update_login
```

Login ID of the user who last updated the sales credit. Corresponds to ASO_SALES_CREDITS.last_update_login.

last_updated_by

public java.math.BigDecimal **last_updated_by**

User ID of the user who last updated the sales credit. Corresponds to ASO_SALES_CREDITS.last_updated_by.

operation_code

public java.lang.String **operation_code**

Operation code used when saving sales credits. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

percent

public java.math.BigDecimal **percent**

Sales credit percent. Corresponds to ASO_SALES_CREDITS.percent.

program_application_id

public java.math.BigDecimal **program_application_id**

Program application ID of the last concurrent program to update the sales credit. Corresponds to ASO_SALES_CREDITS.program_application_id.

program_id

public java.math.BigDecimal **program_id**

Program ID of the last concurrent program to update the sales credit. Corresponds to ASO_SALES_CREDITS.program_id.

program_update_date

public java.sql.Timestamp **program_update_date**

Date the last concurrent program updates the sales credit. Corresponds to ASO_SALES_CREDITS.program_update_date.

qte_line_index

public java.math.BigDecimal **qte_line_index**

Quote line index.

quota_flag

public java.lang.String **quota_flag**

Whether the sales credit type is revenue-based.

quote_header_id

```
public java.math.BigDecimal quote_header_id
```

Quote Header ID. Corresponds to ASO_SALES_CREDITS.quote_header_id.

quote_line_id

```
public java.math.BigDecimal quote_line_id
```

Quote line ID. Corresponds to ASO_SALES_CREDITS.quote_line_id.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

request_id

```
public java.math.BigDecimal request_id
```

Request ID. Corresponds to ASO_SALES_CREDITS.request_id.

resource_ctg

```
public java.lang.String resource_ctg
```

Resource category of the resource receiving the sales credit.

resource_group_id

```
public java.math.BigDecimal resource_group_id
```

Resource group ID for the resource receiving the sales credit. Corresponds to ASO_SALES_CREDITS.resource_id.

resource_id

```
public java.math.BigDecimal resource_id
```

Resource ID receiving the sales credit. Corresponds to ASO_SALES_CREDITS.resource_id.

resource_name

```
public java.lang.String resource_name
```

Resource name of the resource receiving the sales credit.

sales_credit_id

```
public java.math.BigDecimal sales_credit_id
```

Sales credit ID. Corresponds to ASO_SALES_CREDITS.sales_credit_id.

sales_credit_type

```
public java.lang.String sales_credit_type
```

Translated meaning for Sales credit type.

sales_credit_type_id

```
public java.math.BigDecimal sales_credit_type_id
```

Sales credit type ID. Corresponds to ASO_SALES_CREDITS.sales_credit_type_id.

source_id

```
public java.math.BigDecimal source_id
```

Source ID of the resource receiving the sales credit.

5.24.2 Constructors for Class SalesCreditRecord

SalesCreditRecord()

```
public SalesCreditRecord()
```

Default constructor.

SalesCreditRecord(boolean)

```
public SalesCreditRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.24.3 Methods for Class SalesCreditRecord

The following table is an index of the Class SalesCreditRecord methods:

Table 5–26 *Methods for Class SalesCreditRecord*

Method	Description
toString()	String representations of the SalesCreditRecord object. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the SalesCreditRecord object.

Overrides: toString in class Object

Returns: A String representation of the SalesCreditRecord object.

5.25 Class ShipmentRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.ShipmentRecord
```

public class **ShipmentRecord**

ShipmentRecord contains the columns in ASO_SHIPMENTS, which stores shipment information for quote headers and quote lines. It also contains member variables for additional shipment information. When saving shipments, only the member variables which correspond to columns in ASO_SHIPMENTS will be updated in the database.

5.25.1 Fields for Class ShipmentRecord

attribute_category

```
public java.lang.String attribute_category
```

Attribute category which defines the descriptive flexfield structure. Corresponds to ASO_SHIPMENTS.attribute_category.

attribute1

```
public java.lang.String attribute1
```

Descriptive flexfield attribute 1. Corresponds to ASO_SHIPMENTS.attribute1.

attribute10

```
public java.lang.String attribute10
```

Descriptive flexfield attribute 10. Corresponds to ASO_SHIPMENTS.attribute10.

attribute11

```
public java.lang.String attribute11
```

Descriptive flexfield attribute 11. Corresponds to ASO_SHIPMENTS.attribute11.

attribute12

public java.lang.String **attribute12**

Descriptive flexfield attribute 12. Corresponds to ASO_SHIPMENTS.attribute12.

attribute13

public java.lang.String **attribute13**

Descriptive flexfield attribute 13. Corresponds to ASO_SHIPMENTS.attribute13.

attribute14

public java.lang.String **attribute14**

Descriptive flexfield attribute 14. Corresponds to ASO_SHIPMENTS.attribute14.

attribute15

public java.lang.String **attribute15**

Descriptive flexfield attribute 15. Corresponds to ASO_SHIPMENTS.attribute15.

attribute2

public java.lang.String **attribute2**

Descriptive flexfield attribute 2. Corresponds to ASO_SHIPMENTS.attribute2.

attribute3

public java.lang.String **attribute3**

Descriptive flexfield attribute 3. Corresponds to ASO_SHIPMENTS.attribute3.

attribute4

public java.lang.String **attribute4**

Descriptive flexfield attribute 4. Corresponds to ASO_SHIPMENTS.attribute4.

attribute5

public java.lang.String **attribute5**

Descriptive flexfield attribute 5. Corresponds to ASO_SHIPMENTS.attribute5.

attribute6

public java.lang.String **attribute6**

Descriptive flexfield attribute 6. Corresponds to ASO_SHIPMENTS.attribute6.

attribute7

public java.lang.String **attribute7**

Descriptive flexfield attribute 7. Corresponds to ASO_SHIPMENTS.attribute7.

attribute8

```
public java.lang.String attribute8
```

Descriptive flexfield attribute 8. Corresponds to ASO_SHIPMENTS.attribute8.

attribute9

```
public java.lang.String attribute9
```

Descriptive flexfield attribute 9. Corresponds to ASO_SHIPMENTS.attribute9.

created_by

```
public java.math.BigDecimal created_by
```

User ID of the user who created the shipment. Corresponds to ASO_SHIPMENTS.created_by.

creation_date

```
public java.sql.Timestamp creation_date
```

Date the shipment was created. Corresponds to ASO_SHIPMENTS.creation_date.

fob_code

```
public java.lang.String fob_code
```

Point of ownership transfer. Corresponds to ASO_SHIPMENTS.fob_code.

freight_carrier_code

```
public java.lang.String freight_carrier_code
```

Freight carrier code. Corresponds to ASO_SHIPMENTS.freight_carrier_code.

freight_terms_code

```
public java.lang.String freight_terms_code
```

Freight terms code. Corresponds to ASO_SHIPMENTS.freight_terms_code.

is_ship_to_contact_addr

```
public boolean is_ship_to_contact_addr
```

Whether the ship-to address belongs to the contact.

last_update_date

```
public java.sql.Timestamp last_update_date
```

Date the shipment was last updated. Corresponds to ASO_SHIPMENTS.last_update_date.

last_update_login

`public java.math.BigDecimal last_update_login`

Login ID of the user who last updated the shipment. Corresponds to ASO_SHIPMENTS.last_update_login.

last_updated_by

`public java.math.BigDecimal last_updated_by`

User ID of the user who last updated the shipment. Corresponds to ASO_SHIPMENTS.last_updated_by.

operation_code

`public java.lang.String operation_code`

Operation code used when saving shipments. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

order_line_id

`public java.math.BigDecimal order_line_id`

Order line ID. Corresponds to ASO_SHIPMENTS.order_line_id.

packing_instructions

`public java.lang.String packing_instructions`

Packing instructions. Corresponds to ASO_SHIPMENTS.packing_instructions.

pricing_quantity

`public java.math.BigDecimal pricing_quantity`

This member variable is currently not supported.

program_application_id

`public java.math.BigDecimal program_application_id`

Program application ID of the last concurrent program to update the shipment. Corresponds to ASO_SHIPMENTS.program_application_id.

program_id

`public java.math.BigDecimal program_id`

Program ID of the last concurrent program to update the shipment. Corresponds to ASO_SHIPMENTS.program_id.

program_update_date

```
public java.sql.Timestamp program_update_date
```

Date the last concurrent program updates the shipment. Corresponds to ASO_SHIPMENTS.program_update_date.

promise_date

```
public java.sql.Timestamp promise_date
```

In stock promise date. Corresponds to ASO_SHIPMENTS.promise_date.

qte_line_index

```
public java.math.BigDecimal qte_line_index
```

Quote line index.

quantity

```
public java.math.BigDecimal quantity
```

Quantity Corresponds to ASO_SHIPMENTS.quantity.

quote_header_id

```
public java.math.BigDecimal quote_header_id
```

Quote header ID. Corresponds to ASO_SHIPMENTS.quote_header_id.

quote_line_id

```
public java.math.BigDecimal quote_line_id
```

Quote line ID. Corresponds to ASO_SHIPMENTS.quote_line_id.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

request_date

```
public java.sql.Timestamp request_date
```

Requested shipment date. Corresponds to ASO_SHIPMENTS.request_date.

request_id

```
public java.math.BigDecimal request_id
```

Request ID. Corresponds to ASO_SHIPMENTS.request_id.

reservation_id

public java.math.BigDecimal **reservation_id**
Reservation ID. Corresponds to ASO_SHIPMENTS.reservation_id.

reserved_quantity

public java.math.BigDecimal **reserved_quantity**
Reserved quantity. Corresponds to ASO_SHIPMENTS.reserved_quantity.

schedule_ship_date

public java.sql.Timestamp **schedule_ship_date**
Date the shipment is scheduled to ship. Corresponds to ASO_SHIPMENTS.schedule_ship_date.

ship_from_org_id

public java.math.BigDecimal **ship_from_org_id**
Inventory organization from which to ship. Corresponds to ASO_SHIPMENTS.ship_from_org_id.

ship_method_code

public java.lang.String **ship_method_code**
Shipping method code. Corresponds to ASO_SHIPMENTS.shipping_method_code.

ship_partial_flag

public java.lang.String **ship_partial_flag**
Whether items can be shipped separately. Corresponds to ASO_SHIPMENTS.ship_partial_flag.

ship_quote_price

public java.math.BigDecimal **ship_quote_price**
Unit shipment charges. Corresponds to ASO_SHIPMENTS.ship_quote_price.

ship_set_id

public java.math.BigDecimal **ship_set_id**
Ship set ID. Corresponds to ASO_SHIPMENTS.ship_set_id.

ship_to_account_number

public java.lang.String **ship_to_account_number**
Ship to customer account number.

ship_to_addr_party_type

```
public java.lang.String ship_to_addr_party_type
```

Party type of the party to which the ship-to address belongs.

ship_to_address1

```
public java.lang.String ship_to_address1
```

This member variable is currently not supported.

ship_to_address2

```
public java.lang.String ship_to_address2
```

This member variable is currently not supported.

ship_to_address3

```
public java.lang.String ship_to_address3
```

This member variable is currently not supported.

ship_to_address4

```
public java.lang.String ship_to_address4
```

This member variable is currently not supported.

ship_to_city

```
public java.lang.String ship_to_city
```

This member variable is currently not supported.

ship_to_contact_first_name

```
public java.lang.String ship_to_contact_first_name
```

This member variable is currently not supported.

ship_to_contact_id

```
public java.math.BigDecimal ship_to_contact_id
```

Party ID of the ship-to contact person if there is a ship-to contact.

ship_to_contact_last_name

```
public java.lang.String ship_to_contact_last_name
```

This member variable is currently not supported.

ship_to_contact_middle_name

```
public java.lang.String ship_to_contact_middle_name
```

This member variable is currently not supported.

ship_to_contact_name

public java.lang.String **ship_to_contact_name**

Party name of the ship-to contact person if there is a ship-to contact.

ship_to_country

public java.lang.String **ship_to_country**

This member variable is currently not supported.

ship_to_country_code

public java.lang.String **ship_to_country_code**

This member variable is currently not supported.

ship_to_county

public java.lang.String **ship_to_county**

This member variable is currently not supported.

ship_to_cust_account_id

public java.math.BigDecimal **ship_to_cust_account_id**

Ship To customer account ID. Corresponds to ASO_SHIPMENTS.ship_to_cust_account_id.

ship_to_cust_party_id

public java.math.BigDecimal **ship_to_cust_party_id**

Party ID of the ship-to customer. Corresponds to ASO_SHIPMENTS.ship_to_cust_party_id.

ship_to_customer_id

public java.math.BigDecimal **ship_to_customer_id**

Party ID of the ship-to customer of type person or organization.

ship_to_customer_name

public java.lang.String **ship_to_customer_name**

Party name of the ship-to customer of type person or organization.

ship_to_customer_type

public java.lang.String **ship_to_customer_type**

Party type of the ship-to customer of type person or organization.

ship_to_party_id

```
public java.math.BigDecimal ship_to_party_id
```

Ship-to party ID is populated when there is a ship-to contact. The party is of type party relationship. Corresponds to ASO_SHIPMENTS.ship_to_party_id.

ship_to_party_name

```
public java.lang.String ship_to_party_name
```

This member variable is currently not supported.

ship_to_party_site_id

```
public java.math.BigDecimal ship_to_party_site_id
```

Party site ID of the ship-to address. Corresponds to ASO_SHIPMENTS.ship_to_party_site_id.

ship_to_postal_code

```
public java.lang.String ship_to_postal_code
```

This member variable is currently not supported.

ship_to_province

```
public java.lang.String ship_to_province
```

This member variable is currently not supported.

ship_to_relationship_code

```
public java.lang.String ship_to_relationship_code
```

Party relationship code if there is a ship-to contact.

ship_to_relationship_id

```
public java.math.BigDecimal ship_to_relationship_id
```

Party relationship ID if there is a ship-to contact.

ship_to_state

```
public java.lang.String ship_to_state
```

This member variable is currently not supported.

shipment_id

```
public java.math.BigDecimal shipment_id
```

Shipment ID. Corresponds to ASO_SHIPMENTS.shipment_id.

shipment_priority_code

```
public java.lang.String shipment_priority_code
```

Shipment priority code. Corresponds to ASO_SHIPMENTS.shipment_priority_code.

shipping_instructions

```
public java.lang.String shipping_instructions
```

Shipping instructions. Corresponds to ASO_SHIPMENTS.shipping_instructions.

5.25.2 Constructors for Class ShipmentRecord

ShipmentRecord()

```
public ShipmentRecord()
```

Default constructor.

ShipmentRecord(boolean)

```
public ShipmentRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values.

5.25.3 Methods for Class ShipmentRecord

The following table is an index of the Class ShipmentRecord methods:

Table 5–27 *Methods for Class ShipmentRecord*

Method	Description
toString()	String representations of the ShipmentRecord object. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the ShipmentRecord object.

Overrides: toString in class Object

Returns: A String representation of the ShipmentRecord object.

5.26 Class SubmitControlRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.SubmitControlRecord
```

public class **SubmitControlRecord**

SubmitControlRecord is used as an input parameter when submitting a quote to convert it into an order.

5.26.1 Fields for Class SubmitControlRecord

book_flag

```
public java.lang.String book_flag
```

Whether to place the order in booked status.

calculate_price

```
public java.lang.String calculate_price
```

Whether to recalculate price when submitting the order.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

reserve_flag

```
public java.lang.String reserve_flag
```

Whether to reserve.

server_id

```
public java.math.BigDecimal server_id
```

Server ID.

5.26.2 Constructors for Class SubmitControlRecord

SubmitControlRecord()

```
public SubmitControlRecord()
```

Default constructor. If profile ASO_DEFAULT_ORDER_STATE is set to **BOOKED**, sets book_flag = **T**. Otherwise, sets book_flag = **F**. Sets calculate_price flag = **F**. Sets reserve_flag and server_id to **G_MISS values**.

SubmitControlRecord(boolean)

```
public SubmitControlRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values

5.26.3 Methods for Class SubmitControlRecord

The following table is an index of the Class SubmitControlRecord methods:

Table 5–28 *Methods for Class SubmitControlRecord*

Method	Description
toString()	String representations of the SubmitControlRecord object. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the SubmitControlRecord object.

Overrides: toString in class Object

Returns: A String representation of the SubmitControlRecord object.

5.27 Class TaxDetailRecord

```
java.lang.Object
|
+--oracle.apps.aso.quote.TaxDetailRecord
```

public class **TaxDetailRecord**

TaxDetailRecord contains the columns in ASO_TAX_DETAILS, which stores tax detail information for quote headers and quote lines. It also contains member variables for additional tax detail information. When saving tax details, only the member variables which correspond to columns in ASO_TAX_DETAILS will be updated in the database.

5.27.1 Fields for Class TaxDetailRecord

attribute_category

```
public java.lang.String attribute_category
```

Attribute category which defines the descriptive flexfield structure. Corresponds to ASO_TAX_DETAILS.attribute_category.

attribute1

```
public java.lang.String attribute1
```

Descriptive flexfield attribute 1. Corresponds to ASO_TAX_DETAILS.attribute1.

attribute10

```
public java.lang.String attribute10
```

Descriptive flexfield attribute 10. Corresponds to ASO_TAX_DETAILS.attribute10.

attribute11

```
public java.lang.String attribute11
```

Descriptive flexfield attribute 11. Corresponds to ASO_TAX_DETAILS.attribute11.

attribute12

```
public java.lang.String attribute12
```

Descriptive flexfield attribute 12. Corresponds to ASO_TAX_DETAILS.attribute12.

attribute13

```
public java.lang.String attribute13
```

Descriptive flexfield attribute 13. Corresponds to ASO_TAX_DETAILS.attribute13.

attribute14

```
public java.lang.String attribute14
```

Descriptive flexfield attribute 14. Corresponds to ASO_TAX_DETAILS.attribute14.

attribute15

```
public java.lang.String attribute15
```

Descriptive flexfield attribute 15. Corresponds to ASO_TAX_DETAILS.attribute15.

attribute2

```
public java.lang.String attribute2
```

Descriptive flexfield attribute 2. Corresponds to ASO_TAX_DETAILS.attribute2.

attribute3

```
public java.lang.String attribute3
```

Descriptive flexfield attribute 3. Corresponds to ASO_TAX_DETAILS.attribute3.

attribute4

```
public java.lang.String attribute4
```

Descriptive flexfield attribute 4. Corresponds to ASO_TAX_DETAILS.attribute4.

attribute5

```
public java.lang.String attribute5
```

Descriptive flexfield attribute 5. Corresponds to ASO_TAX_DETAILS.attribute5.

attribute6

```
public java.lang.String attribute6
```

Descriptive flexfield attribute 6. Corresponds to ASO_TAX_DETAILS.attribute6.

attribute7

```
public java.lang.String attribute7
```

Descriptive flexfield attribute 7. Corresponds to ASO_TAX_DETAILS.attribute7.

attribute8

```
public java.lang.String attribute8
```

Descriptive flexfield attribute 8. Corresponds to ASO_TAX_DETAILS.attribute8.

attribute9

```
public java.lang.String attribute9
```

Descriptive flexfield attribute 9. Corresponds to ASO_TAX_DETAILS.attribute9.

created_by

```
public java.math.BigDecimal created_by
```

User ID of the user who created the tax detail. Corresponds to ASO_TAX_DETAILS.created_by.

creation_date

```
public java.sql.Timestamp creation_date
```

Date the tax detail was created. Corresponds to ASO_TAX_DETAILS.creation_date.

last_update_date

```
public java.sql.Timestamp last_update_date
```

Date the tax detail was last updated. Corresponds to ASO_TAX_DETAILS.last_update_date.

last_update_login

```
public java.math.BigDecimal last_update_login
```

Login ID of the user who last updated the tax detail. Corresponds to ASO_TAX_DETAILS.last_update_login.

last_updated_by

```
public java.math.BigDecimal last_updated_by
```

User ID of the user who last updated the tax detail. Corresponds to ASO_TAX_DETAILS.last_updated_by.

operation_code

```
public java.lang.String operation_code
```

Operation code used when saving tax details. Used to indicate if the operation is CREATE, UPDATE, or DELETE.

orig_tax_code

```
public java.lang.String orig_tax_code
```

Tax group code. Corresponds to ASO_TAX_DETAILS.orig_tax_code.

program_application_id

public java.math.BigDecimal **program_application_id**
Program application ID of the last concurrent program to update the tax detail. Corresponds to ASO_TAX_DETAILS.program_application_id.

program_id

public java.math.BigDecimal **program_id**
Program ID of the last concurrent program to update the tax detail. Corresponds to ASO_TAX_DETAILS.program_id.

program_update_date

public java.sql.Timestamp **program_update_date**
Date the last concurrent program updates the tax detail. Corresponds to ASO_TAX_DETAILS.program_update_date.

qte_line_index

public java.math.BigDecimal **qte_line_index**
Quote line index.

quote_header_id

public java.math.BigDecimal **quote_header_id**
Quote header ID. Corresponds to ASO_TAX_DETAILS.quote_header_id.

quote_line_id

public java.math.BigDecimal **quote_line_id**
Quote line ID. Corresponds to ASO_TAX_DETAILS.quote_line_id.

quote_shipment_id

public java.math.BigDecimal **quote_shipment_id**
Quote shipment ID. Corresponds to ASO_TAX_DETAILS.quote_shipment_id.

RCS_ID

public static final java.lang.String **RCS_ID**

RCS_ID_RECORDED

public static final boolean **RCS_ID_RECORDED**

request_id

```
public java.math.BigDecimal request_id
```

Request ID. Corresponds to ASO_TAX_DETAILS.request_id.

shipment_index

```
public java.math.BigDecimal shipment_index
```

Quote shipment index.

tax_amount

```
public java.math.BigDecimal tax_amount
```

Tax amount. Corresponds to ASO_TAX_DETAILS.tax_amount.

tax_code

```
public java.lang.String tax_code
```

Tax code. Corresponds to ASO_TAX_DETAILS.tax_code.

tax_date

```
public java.sql.Timestamp tax_date
```

Tax date. Corresponds to ASO_TAX_DETAILS.tax_date.

tax_detail_id

```
public java.math.BigDecimal tax_detail_id
```

Tax detail ID. Corresponds to ASO_TAX_DETAILS.tax_detail_id.

tax_exempt_flag

```
public java.lang.String tax_exempt_flag
```

Tax exemption status (standard, require, or exempt). Corresponds to ASO_TAX_DETAILS.tax_exempt_flag.

tax_exempt_number

```
public java.lang.String tax_exempt_number
```

Tax exemption number. Corresponds to ASO_TAX_DETAILS.tax_exempt_number.

tax_exempt_reason_code

```
public java.lang.String tax_exempt_reason_code
```

Tax exemption reason code. Corresponds to ASO_TAX_DETAILS.tax_exempt_reason_code.

tax_inclusive_flag

```
public java.lang.String tax_inclusive_flag
```

Whether the tax is included in the price of the product. Corresponds to ASO_TAX_DETAILS.tax_inclusive_flag.

tax_name

```
public java.lang.String tax_name
```

Translated tax name.

tax_rate

```
public java.math.BigDecimal tax_rate
```

Tax rate. Corresponds to ASO_TAX_DETAILS.tax_rate.

5.27.2 Constructors for Class TaxDetailRecord

TaxDetailRecord()

```
public TaxDetailRecord()
```

Default constructor.

TaxDetailRecord(boolean)

```
public TaxDetailRecord(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values

5.27.3 Methods for Class TaxDetailRecord

The following table is an index of the Class TaxDetailRecord methods:

Table 5–29 *Methods for Class TaxDetailRecord*

Method	Description
toString()	String representations of the TaxDetailRecord object. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the TaxDetailRecord object.

Overrides: toString in class Object

Returns: A String representation of the TaxDetailRecord object.

5.28 Class WarrantyRec

```
java.lang.Object
|
+--oracle.apps.aso.quote.WarrantyRec
```

public class WarrantyRec

WarrantyRec is used as an output parameter when retrieving the included warranties for an inventory item.

5.28.1 Fields for Class WarrantyRec

coverage_schedule_id

```
public java.math.BigDecimal coverage_schedule_id
```

Coverage schedule ID.

duration_period

```
public java.lang.String duration_period
```

Service duration period of the included service warranty.

duration_quantity

```
public java.math.BigDecimal duration_quantity
```

Service duration of the included service warranty.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

service_description

```
public java.lang.String service_description
```

Inventory item description of the included service warranty.

service_item_id

```
public java.math.BigDecimal service_item_id
```

Inventory item ID of the included service warranty.

service_name

```
public java.lang.String service_name
```

Inventory item part number of the included service warranty.

warranty_end_date

```
public java.sql.Timestamp warranty_end_date
```

End date of the included service warranty.

warranty_start_date

```
public java.sql.Timestamp warranty_start_date
```

Start date of the included service warranty.

5.28.2 Constructors for Class WarrantyRec

WarrantyRec()

```
public WarrantyRec()
```

Default constructor.

WarrantyRec(boolean)

```
public WarrantyRec(boolean __RosettaUseGMISSValues)
```

Constructor.

Parameters: __RosettaUseGMISSValues - Whether to default all values to Rosetta G_MISS values

5.28.3 Methods for Class WarrantyRec

The following table is an index of the Class WarrantyRec methods:

Table 5–30 *Methods for Class WarrantyRec*

Method	Description
toString()	String representations of the WarrantyRec is object. <pre>public java.lang.String toString()</pre>

toString()

```
public java.lang.String toString()
```

Returns a String representation of the WarrantyRec object.

Overrides: toString in class Object

Returns: A String representation of the WarrantyRec object.

oracle.apps.qot.util

This section lists the Oracle Quoting Java APIs in the package oracle.apps.qot.util.

6.1 Package oracle.apps.qot.util

The table below lists a description for each class in this package:

Table 6–1 Class Summary

Class	Description
Class FlexMapper	FlexMapper provides the methods to map the fields from an object to a Hashtable and to update flexfield information in the database for an object.
Class LookupValuesKey	LookupValuesKey is the key for the HTML Quoting lookup values cache.
Class OperatingUnit	OperatingUnit contains the following information for an operating unit: operating unit ID, master inventory organization ID, default territory code, bill to territory codes, and ship to territory codes.
Class QotLookup	QotLookup is a generic container class for the lookups (dropdowns) used in HTML Quoting.
Class QotRegionManager	QotRegionManager provides utility methods for retrieving information regarding regions.
Class QotUtil	QotUtil provides utility methods used in Quoting.
Class QueryResultSet	QueryResultSet is implemented to hold the result of an SQL query.
Class RequestCtx	RequestCtx holds the session information for an HTTP request.

6.2 Class FlexMapper

```
java.lang.Object
|
+--oracle.apps.qot.util.FlexMapper
```

public class **FlexMapper**

FlexMapper provides the methods to map the fields from an object to a Hashtable and to update flexfield information in the database for an object.

Table 6–2 *Inherited Member Summary*

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(long, int), wait(long, int), wait(long, int)

6.2.1 Fields for Class FlexMapper

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

6.2.2 Constructors for Class FlexMapper

FlexMapper(Object, Flexfield)

```
public FlexMapper(java.lang.Object src,  
oracle.apps.jtf.jflex.Flexfield flexfield)
```

Constructor

Parameters:

src - Source object.

flexfield - Flexfield object.

6.2.3 Methods for Class FlexMapper

The following table is an index of the Class FlexMapper methods:

Table 6–3 *Methods for Class FlexMapper*

Method	Description
mapValuesFromDb(Object)	Loads all the field names and values for a given object. <pre>public static java.util.Hashtable mapValuesFromDb(java.lang.Object data) throws IllegalAccessException, FlexException</pre>
mapValuesToDb(Object, String[])	Updates flexfield information in the database for the object. After comparing the columns of the flexfield with the object, new values will be inserted in the database. <pre>public static boolean mapValuesToDb(java.lang.Object data, java.lang.String[][] segValues) throws IllegalAccessException, IllegalArgumentException</pre>

mapValuesFromDb(Object)

```
public static java.util.Hashtable mapValuesFromDb(java.lang.Object data)
throws IllegalAccessException, FlexException
```

Loads all the field names and values for a given object.

Parameters: data - an Object to be read

Returns: Hashtable containing field names and values. The key in the Hashtable is field name, the value is field value.

Throws:

java.lang.IllegalAccessException - If an executing method does not have access to a class, field, method, or constructor while reflectively creating an instance, setting or getting a field, or invoking a method.

FlexException - If a flexfield internal error occurs.

oracle.apps.fnd.flexj.FlexException

mapValuesToDb(Object, String[][])

```
public static boolean mapValuesToDb(java.lang.Object data,
java.lang.String[][] segValues)
throws IllegalAccessException, IllegalArgumentException
```

Updates flexfield information in the database for the object. After comparing the columns of the flexfield with the object, new values will be inserted in the database.

Parameters:

data - An object to be updated.

segValues - Name of the flexfield segments and values.

Returns: Whether the operation was successful.

Throws:

java.lang.IllegalAccessException - If an executing method does not have access to a class, field, method, or constructor while reflectively creating an instance, setting or getting a field, or invoking a method.

java.lang.IllegalArgumentException - If an argument is illegal or inappropriate.

6.3 Class OperatingUnit

```
java.lang.Object
```

```
|
```

```
+--oracle.apps.got.util.OperatingUnit
```

All Implemented Interfaces: java.io.Serializable

public class **OperatingUnit** implements java.io.Serializable

OperatingUnit contains the following information for an operating unit: operating unit ID, master inventory organization ID, default territory code, bill to territory codes, and ship to territory codes. It provides a method to retrieve an OperatingUnit for a specified operating unit ID.

Table 6–4 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

6.3.1 Fields for Class OperatingUnit

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

6.3.2 Constructors for Class OperatingUnit

OperatingUnit(BigDecimal)

```
public OperatingUnit(java.math.BigDecimal operatingUnitId)
```

Constructs operating unit with operatingUnitId passed in as parameter

Parameters: operatingUnitId - operating unit ID.

6.3.3 Methods for Class OperatingUnit

The following table is an index of the Class OperatingUnit methods:

Table 6–5 *Methods for Class OperatingUnit*

Method	Description
getBillToTerritoryCodes()	Returns bill to territory codes. public java.lang.String[] getBillToTerritoryCodes()
getDefaultTerritoryCode()	Returns default territory code. public java.lang.String getDefaultTerritoryCode()
getMasterOrganizationId()	Returns master organization ID used for inventory item validation. public java.math.BigDecimal getMasterOrganizationId()
getOperatingUnitId()	Returns operating unit ID. public java.math.BigDecimal getOperatingUnitId()
getShipToTerritoryCodes()	Returns ship to territory codes. public java.lang.String[] getShipToTerritoryCodes()

Table 6–5 Methods for Class OperatingUnit

Method	Description
<code>load(BigDecimal)</code>	Returns an <code>OperatingUnit</code> object corresponding to the operating unit ID passed in as parameter. <pre>public static oracle.apps.got.util.OperatingUnit load (java.math.BigDecimal operatingUnitId) throws FrameworkException</pre>
<code>setBillToTerritoryCodes(String[])</code>	Sets bill to territory codes. <pre>public void setBillToTerritoryCodes (java.lang.String [] billToTerrCodes)</pre>
<code>setDefaultTerritoryCode(String)</code>	Sets default territory code. <pre>public void setDefaultTerritoryCode (java.lang.String dfltTerrCode)</pre>
<code>setMasterOrganizationId(BigDecimal)</code>	Sets master organization ID used for inventory item validation. <pre>public void setMasterOrganizationId (java.math.BigDec imal masterOrgId)</pre>
<code>setOperatingUnitId(BigDecimal)</code>	Sets operating unit ID. <pre>public void setOperatingUnitId (java.math.BigDecimal operatingUnitId)</pre>
<code>setShipToTerritoryCodes(String[])</code>	Sets ship to territory codes. <pre>public void setShipToTerritoryCodes (java.lang.String [] shipToTerrCodes)</pre>
<code>toString()</code>	Returns String representation of the <code>OperatingUnit</code> object. <pre>public java.lang.String toString()</pre>

getBillToTerritoryCodes()

```
public java.lang.String[] getBillToTerritoryCodes()
```

Returns bill to territory codes.

Returns: An array of bill to territory codes.

getDefaultTerritoryCode()

```
public java.lang.String getDefaultTerritoryCode()
```

Returns default territory code.

Returns: default territory code.

getMasterOrganizationId()

```
public java.math.BigDecimal getMasterOrganizationId()
```

Returns master organization ID used for inventory item validation.

Returns: master organization ID used for inventory item validation.

getOperatingUnitId()

```
public java.math.BigDecimal getOperatingUnitId()
```

Returns operating unit ID.

Returns: operating unit ID.

getShipToTerritoryCodes()

```
public java.lang.String[] getShipToTerritoryCodes()
```

Returns ship to territory codes.

Returns: An array of ship to territory codes.

load(BigDecimal)

```
public static oracle.apps.qot.util.OperatingUnit  
load(java.math.BigDecimal operatingUnitId)  
throws FrameworkException
```

Returns an OperatingUnit object corresponding to the operating unit ID passed in as parameter.

Parameters: operatingUnitId - operating unit ID.

Returns: OperatingUnit object with the appropriate information populated.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setBillToTerritoryCodes(String[])

```
public void setBillToTerritoryCodes(java.lang.String[] billToTerrCodes)
```

Sets bill to territory codes.

Parameters: billToTerrCodes - An array of bill to territory codes.

setDefaultTerritoryCode(String)

```
public void setDefaultTerritoryCode(java.lang.String dfltTerrCode)
```

Sets default territory code.

Parameters: dfltTerrCode - default territory code.

setMasterOrganizationId(BigDecimal)

```
public void setMasterOrganizationId(java.math.BigDecimal masterOrgId)
```

Sets master organization ID used for inventory item validation.

Parameters: masterOrgId - master organization ID used for inventory item validation.

setOperatingUnitId(BigDecimal)

```
public void setOperatingUnitId(java.math.BigDecimal operatingUnitId)
```

Sets operating unit ID.

Parameters: operatingUnitId - operating unit ID.

setShipToTerritoryCodes(String[])

```
public void setShipToTerritoryCodes(java.lang.String[] shipToTerrCodes)
```

Sets ship to territory codes.

Parameters: shipToTerrCodes - An array of ship to territory codes.

toString()

```
public java.lang.String toString()
```

Returns String representation of the OperatingUnit object.

Overrides: toString in class Object

Returns: String representation of the OperatingUnit object.

6.4 Class LookupValuesKey

```
java.lang.Object
|
+--oracle.apps.got.util.LookupValuesKey
public class LookupValuesKey
```


LookupValuesKey is the key for the HTML Quoting lookup values cache. The lookup key is composed of a lookup type and a view application ID.

Table 6–6 Inherited Member Summary

Methods inherited from class Object

getClass(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

6.4.1 Fields for Class LookupValuesKey

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

6.4.2 Constructors for Class LookupValuesKey

LookupValuesKey()

```
public LookupValuesKey()
```

Default constructor.

LookupValuesKey(String, BigDecimal)

```
public LookupValuesKey(java.lang.String lookupType,  
java.math.BigDecimal viewApplicationId)
```

Constructs a LookupValuesKey object by specifying the lookup type and the view application ID.

Parameters:

lookupType - The type of the lookup.

viewApplicationId - The ID of the application where the lookup is defined.

6.4.3 Methods for Class LookupValuesKey

Table 6–7 *Methods for Class LookupValuesKey*

Method	Description
<code>equals(Object)</code>	<p>Compares this object to the specified object passed in as parameter. Two objects are considered equal if and only if:</p> <ul style="list-style-type: none"> the input object is not null; and the input object is an instance of <code>LookupValuesKey</code> class; and the input object and this object have equal return values for method <code>getLookupType()</code>; and the input object and this object have equal return values for method <code>getViewApplicationId()</code>. <pre>public boolean equals(java.lang.Object anObject)</pre>
<code>getLookupType()</code>	<p>Returns the lookup type of the lookup key.</p> <pre>public java.lang.String getLookupType()</pre>
<code>getViewApplicationId()</code>	<p>Returns the view application ID of the lookup key.</p> <pre>public java.math.BigDecimal getViewApplicationId()</pre>
<code>hashCode()</code>	<p>Returns a hash code for this object. The hash code for a <code>LookupValuesKey</code> object is computed as the addition of <code>lookupType.hashCode()*P1</code> and <code>viewApplicationId.hashCode()*P2</code>, where <code>P1</code> and <code>P2</code> are two prime numbers. The hash code of an empty object is 0.</p> <pre>public int hashCode()</pre>
<code>setLookupType(String)</code>	<p>Sets the lookup type of the lookup key.</p> <pre>public void setLookupType(java.lang.String lookupType)</pre>
<code>setViewApplicationId(BigDecimal)</code>	<p>Sets the view application ID of the lookup key.</p> <pre>public void setViewApplicationId(java.math.BigDec imal viewApplicationId)</pre>

Table 6–7 Methods for Class LookupValuesKey

Method	Description
toString()	Returns a String representation of the LookupValuesKey object. public java.lang.String toString()

equals(Object)

```
public boolean equals(java.lang.Object anObject)
```

Compares this object to the specified object passed in as parameter. Two objects are considered equal if and only if:

- The input object is not null.
- The input object is an instance of LookupValuesKey class.
- The input object and this object have equal return values for method getLookupType().
- The input object and this object have equal return values for method getViewApplicationId().

Overrides: equals in class Object

Parameters: anObject - The object to compare this object against.

Returns: True if two objects are equal; false otherwise.

getLookupType()

```
public java.lang.String getLookupType()
```

Returns the lookup type of the lookup key.

Returns: The lookup type of the lookup key.

getViewApplicationId()

```
public java.math.BigDecimal getViewApplicationId()
```

Returns the view application ID of the lookup key.

Returns: The view application ID of the lookup key.

hashCode()

```
public int hashCode()
```

Returns a hash code for this object. The hash code for a LookupValuesKey object is computed as the addition of lookupType.hashCode()*P1 and

`viewApplicationId.hashCode()*P2`, where P1 and P2 are two prime numbers. The hash code of an empty object is 0.

Overrides: `hashCode` in class `Object`

Returns: A hash code value for this object.

setLookupType(String)

```
public void setLookupType(java.lang.String lookupType)
```

Sets the lookup type of the lookup key.

Parameters: `lookupType` - The lookup type.

setViewApplicationId(BigDecimal)

```
public void setViewApplicationId(java.math.BigDecimal viewApplicationId)
```

Sets the view application ID of the lookup key.

Parameters: `viewApplicationId` - The view application ID.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the `LookupValuesKey` object.

Overrides: `toString` in class `Object`

Returns: A String representation of the `LookupValuesKey` object.

6.5 Class QotLookup

```
java.lang.Object
```

```
|
```

```
+--oracle.apps.qot.util.QotLookup
```

All Implemented Interfaces: `java.io.Serializable`

```
public class QotLookup implements java.io.Serializable
```

`QotLookup` is a generic container class for the lookups (dropdowns) used in HTML Quoting. The lookup is composed of a key, translated names, and an index for ordering. This class provides methods for retrieving a specific `QotLookup` object and listing all the `QotLookup` objects of a specific type.

Table 6–8 Inherited Member Summary

Methods inherited from class `Object`

Table 6–8 Inherited Member Summary

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

6.5.1 Fields for Class QotLookup

ASO_PQUOTE_OUTPUT_FMT

`public static final int ASO_PQUOTE_OUTPUT_FMT`

Indicates print quote output format lookup type. Used as input to `load()` and `list()` methods.

ASO_PQUOTE_TEMPLATES

`public static final int ASO_PQUOTE_TEMPLATES`

Indicates print quote templates lookup type. Used as input to `load()` and `list()` methods.

ATP_STATUS

`public static final int ATP_STATUS`

Indicates ATP status lookup type. Used as input to `load()` and `list()` methods.

BOM_ITEM_TYPE

`public static final int BOM_ITEM_TYPE`

Indicates Bill of Materials item type lookup type. Used as input to `load()` and `list()` methods.

CONC_REQ_PHASE

`public static final int CONC_REQ_PHASE`

Indicates concurrent request phase lookup type. Used as input to `load()` and `list()` methods.

CONC_REQ_STATUS

`public static final int CONC_REQ_STATUS`

Indicates concurrent request status lookup type. Used as input to `load()` and `list()` methods.

CREDIT_CARD_TYPE

`public static final int CREDIT_CARD_TYPE`

Indicates credit card lookup type. Used as input to `load()` and `list()` methods.

FREIGHT_CHARGES_TYPE

public static final int **FREIGHT_CHARGES_TYPE**

Indicates freight charges type lookup type. Used as input to load() and list() methods.

FREIGHT_COST_TYPE

public static final int **FREIGHT_COST_TYPE**

Indicates freight cost type lookup type. Used as input to load() and list() methods.

INSTALLBASE_ACTION

public static final int **INSTALLBASE_ACTION**

Indicates install base action lookup type. Used as input to load() and list() methods.

INSTALLBASE_RELATIONSHIP

public static final int **INSTALLBASE_RELATIONSHIP**

Indicates install base relationship lookup type. Used as input to load() and list() methods.

ITEM_TYPE

public static final int **ITEM_TYPE**

Indicates inventory item type lookup type. Used as input to load() and list() methods.

LINE_ACTION

public static final int **LINE_ACTION**

Indicates line action lookup type. Used as input to load() and list() methods.

LINE_ITEM_ACTION

public static final int **LINE_ITEM_ACTION**

Indicates line item action lookup type. Used as input to load() and list() methods.

MKTG_SRC_TYPE

public static final int **MKTG_SRC_TYPE**

Indicates marketing source lookup type. Used as input to load() and list() methods.

MODIFIER_LEVEL

public static final int **MODIFIER_LEVEL**

Indicates modifier level lookup type. Used as input to load() and list() methods.

MODIFIER_LINE_TYPE

```
public static final int MODIFIER_LINE_TYPE
```

Indicates modifier line type lookup type. Used as input to load() and list() methods.

PAYMENT_METHOD

```
public static final int PAYMENT_METHOD
```

Indicates payment method. Used as input to load() and list() methods.

PAYMENT_TERM

```
public static final int PAYMENT_TERM
```

Indicates payment term lookup type. Used as input to load() and list() methods.

PRC_TAX_STATUS

```
public static final int PRC_TAX_STATUS
```

Indicates price and tax status lookup type. Used as input to load() and list() methods.

PRODUCT_SOURCE

```
public static final int PRODUCT_SOURCE
```

Indicates product source lookup type. Used as input to load() and list() methods.

PROPOSAL_STATUS

```
public static final int PROPOSAL_STATUS
```

Indicates proposal status lookup type. Used as input to load() and list() methods.

QUOTE_OVERVIEW_ACTION

```
public static final int QUOTE_OVERVIEW_ACTION
```

Indicates quote overview action lookup type. Used as input to load() and list() methods.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

RELATIONSHIP_TYPE

```
public static final int RELATIONSHIP_TYPE
```

Indicates inventory relationship lookup type. Used as input to load() and list() methods.

RESOURCE_CTG

```
public static final int RESOURCE_CTG
```

Indicates resource category lookup type. Used as input to load() and list() methods.

SALES_CHANNEL

```
public static final int SALES_CHANNEL
```

Indicates sales channel lookup type. Used as input to load() and list() methods.

SAVED_SEARCH

```
public static final int SAVED_SEARCH
```

Indicates saved search lookup type. Used as input to load() and list() methods.

TAX_EXEMPT_REASON

```
public static final int TAX_EXEMPT_REASON
```

Indicates tax exempt reason lookup type. Used as input to load() and list() methods.

TAX_EXEMPT_STATUS

```
public static final int TAX_EXEMPT_STATUS
```

Indicates tax exempt status lookup type. Used as input to load() and list() methods.

TRXN_LINE_CTG

```
public static final int TRXN_LINE_CTG
```

Indicates transaction line category lookup type. Used as input to load() and list() methods.

YES_NO

```
public static final int YES_NO
```

Indicates yes-no lookup type. Used as input to load() and list() methods.

6.5.2 Constructors for Class QotLookup

QotLookup()

```
public QotLookup()
```

Default constructor.

QotLookup(Object, HashMap, int)

```
public QotLookup(java.lang.Object key,
com.sun.java.util.collections.HashMap name, int index)
```

Constructs QotLookup object with key, translated names, and index passed in as parameter.

Parameters:

key - Lookup key.

name - Hashmap containing translated name. Key in the hashmap is language code, value is translated name.

index - Index used for ordering.

6.5.3 Methods for Class QotLookup

The following table is an index of the Class QotLookup methods:

Table 6–9 Methods for Class QotLookup

Method	Description
addDescription (String,String)	Adds the translated description for the specified language code public void addDescription (java.lang.String lang, java.lang.String desc)
addName(String, String)	Add translated name for the specified language code. public void addName (java.lang.String lang, java.lang.String name)
getDescription()	Returns the translated description in the current language public java.lang.String getDescription ()
getIndex()	Returns the lookup index. public int getIndex ()
getKey()	Returns the key of the lookup object. public java.lang.Object getKey ()
getName()	Return translated name in the current language. public java.lang.String getName ()
isEnabled()	Returns whether the lookup is enabled. public boolean isEnabled ()

Table 6–9 Methods for Class QotLookup

Method	Description
<code>list(int)</code>	List the lookups of the type passed in as parameter. <pre>public static oracle.apps.got.util.QotLookup[] list(int type) throws FrameworkException</pre>
<code>load(int, object)</code>	Returns the lookup object corresponding to the type and key passed in as parameter. <pre>public static oracle.apps.got.util.QotLookup load(int type, java.lang.Object key) throws FrameworkException</pre>
<code>load(LookupValuesKey, Object)</code>	Returns the lookup object corresponding to the <code>LookupValuesKey</code> and key passed in as parameter. The lookup object value will be retrieved from Quoting generic lookup values cache. <pre>public static oracle.apps.got.util.QotLookup load(oracle.apps.got.util.LookupValuesKey lookupKey, java.lang.Object key) throws FrameworkException</pre>
<code>setEnabled(boolean)</code>	Sets whether the lookup is enabled. <pre>public void setEnabled(boolean isEnabled)</pre>
<code>setIndex(int)</code>	Set index used for ordering. <pre>public void setIndex(int index)</pre>
<code>setKey(object)</code>	Set lookup key. <pre>public void setKey(java.lang.Object key)</pre>
<code>setType(int)</code>	Set lookup type. <pre>public void setType(int type)</pre>
<code>toString()</code>	Return String representation of the lookup. <pre>public java.lang.String toString()</pre>

addDescription(String, String)

```
public void addDescription(java.lang.String lang, java.lang.String desc)
```

Adds the translated description for the specified language code

Parameters:

`lang` - Language code.

desc - Translated description for the specified language code.

addName(String, String)

```
public void addName(java.lang.String lang, java.lang.String name)
```

Add translated name for the specified language code

Parameters:

lang - Language code.

name - Translated name for the specified language code.

getDescription()

```
public java.lang.String getDescription()
```

Returns the translated description in the current language.

Returns: The translated description in the current language.

getIndex()

```
public int getIndex()
```

Returns the lookup index.

Returns: The lookup index.

getKey()

```
public java.lang.Object getKey()
```

Return key of the lookup object

Returns: Lookup key.

getName()

```
public java.lang.String getName()
```

Return translated name in the current language

Returns: Translated name in the current language.

isEnabled()

```
public boolean isEnabled()
```

Returns whether the lookup is enabled

Returns: True if the lookup is enabled. False otherwise.

list(int)

```
public static oracle.apps.got.util.QotLookup[] list(int type)
throws FrameworkException
```

List the lookups of the type passed in as parameter.

Parameters:

type - Lookup type. Possible values:

- CREDIT_CARD_TYPE
- PAYMENT_TERM
- PAYMENT_METHOD
- RESOURCE_CTG
- SALES_CHANNEL
- TAX_EXEMPT_REASON
- TRXN_LINE_CTG
- TAX_EXEMPT_STATUS
- RELATIONSHIP_TYPE
- YES_NO
- BOM_ITEM_TYPE
- ITEM_TYPE
- ASO_PQUOTE_OUTPUT_FMT
- ASO_PQUOTE_TEMPLATES
- MKTG_SRC_TYPE
- ATP_STATUS
- MODIFIER_LINE_TYPE
- MODIFIER_LEVEL
- FREIGHT_CHARGES_TYPE
- FREIGHT_COST_TYPE
- INSTALLBASE_ACTION
- INSTALLBASE_RELATIONSHIP
- PRODUCT_SOURCE

- LINE_ACTION
- LINE_ITEM_ACTION
- QUOTE_OVERVIEW_ACTION
- CONC_REQ_PHASE
- CONC_REQ_STATUS
- PRC_TAX_STATUS
- PROPOSAL_STATUS
- SAVED_SEARCH

Returns: An array of QotLookup objects of the type passed in as parameter.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(int, Object)

```
public static oracle.apps.qot.util.QotLookup load(int type,  
java.lang.Object key)throws FrameworkException
```

Returns the lookup object corresponding to the type and key passed in as parameter.

Parameters:

type - Lookup type. Possible values:

- CREDIT_CARD_TYPE
- PAYMENT_TERM
- PAYMENT_METHOD
- RESOURCE_CTG
- SALES_CHANNEL
- TAX_EXEMPT_REASON
- TRXN_LINE_CTG
- TAX_EXEMPT_STATUS
- RELATIONSHIP_TYPE
- YES_NO
- BOM_ITEM_TYPE

- ITEM_TYPE
- ASO_PQUOTE_OUTPUT_FMT
- ASO_PQUOTE_TEMPLATES
- MKTG_SRC_TYPE
- ATP_STATUS
- MODIFIER_LINE_TYPE
- MODIFIER_LEVEL
- FREIGHT_CHARGES_TYPE
- FREIGHT_COST_TYPE
- INSTALLBASE_ACTION
- INSTALLBASE_RELATIONSHIP
- PRODUCT_SOURCE
- LINE_ACTION
- LINE_ITEM_ACTION
- QUOTE_OVERVIEW_ACTION
- CONC_REQ_PHASE
- CONC_REQ_STATUS
- PRC_TAX_STATUS
- PROPOSAL_STATUS
- SAVED_SEARCH

key - Lookup key.

Returns: QotLookup object with the appropriate information populated.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(LookupValuesKey, Object)

```
public static oracle.apps.qot.util.QotLookup  
load(oracle.apps.qot.util.LookupValuesKey lookupKey,  
java.lang.Object key)throws FrameworkException
```

Returns the lookup object corresponding to the LookupValuesKey and key passed in as parameter. The lookup object value will be retrieved from Quoting generic lookup values cache.

Parameters:

lookupKey - A LookupValuesKey object containing the lookup type and the view application ID.

key - An object specifying the lookup key.

Returns: A QotLookup object containing the lookup key as well as the translated lookup value.

Throws: oracle.apps.jtf.base.resources.FrameworkException- If a system error occurs.

setEnabled(boolean)

```
public void setEnabled(boolean isEnabled)
```

Sets whether the lookup is enabled.

Parameters: isEnabled - Whether the lookup is enabled.

setIndex(int)

```
public void setIndex(int index)
```

Set index used for ordering.

Parameters: index - Index used for ordering.

setKey(Object)

```
public void setKey(java.lang.Object key)
```

Set lookup key

Parameters: key - Lookup key.

setType(int)

```
public void setType(int type)
```

Set lookup type.

Parameters:

type - Lookup type. Possible values:

- CREDIT_CARD_TYPE
- PAYMENT_TERM

- PAYMENT_METHOD
- RESOURCE_CTG
- SALES_CHANNEL
- TAX_EXEMPT_REASON
- TRXN_LINE_CTG
- TAX_EXEMPT_STATUS
- RELATIONSHIP_TYPE
- YES_NO
- BOM_ITEM_TYPE
- ITEM_TYPE
- ASO_PQUOTE_OUTPUT_FMT
- ASO_PQUOTE_TEMPLATES
- MKTG_SRC_TYPE
- ATP_STATUS
- MODIFIER_LINE_TYPE
- MODIFIER_LEVEL
- FREIGHT_CHARGES_TYPE
- FREIGHT_COST_TYPE
- INSTALLBASE_ACTION
- INSTALLBASE_RELATIONSHIP
- PRODUCT_SOURCE
- LINE_ACTION
- LINE_ITEM_ACTION
- QUOTE_OVERVIEW_ACTION
- CONC_REQ_STATUS
- CONC_REQ_PHASE
- PRC_TAX_STATUS
- SAVED_SEARCH

toString()

```
public java.lang.String toString()
```

Return String representation of the lookup

Overrides: toString in class Object

Returns: A String representation of the QotLookup object.

6.6 Class QotRegionManager

```
java.lang.Object
|
+--oracle.apps.qot.util.QotRegionManager
```

```
public class QotRegionManager
```

QotRegionManager provides utility methods for retrieving information regarding regions.

Table 6–10 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(long, int), wait(long, int), wait(long, int)

6.6.1 Fields for Class QotRegionManager

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

6.6.2 Methods for Class QotRegionManager

The following table is an index of the Class QotRegionManager methods:

Table 6–11 Methods for Class QotRegionManager

Method	Description
getDescription()	Returns the region description. <code>public java.lang.String getDescription()</code>
getMaxLength(String)	Returns the display length for this region item. <code>public int getMaxLength(java.lang.String attributeName)</code>
getPrompt(String)	Returns the user language prompt for this region item. <code>public java.lang.String getPrompt(java.lang.String attributeName)</code> throws FrameworkException, SQLException
is4thLevelDisabled()	Returns whether the 4th level menu is disabled. <code>public boolean is4thLevelDisabled()</code>
isDisplayable(String)	Returns whether the region item is displayable. <code>public boolean isDisplayable(java.lang.String attributeName)</code> throws FrameworkException, SQLException
load(String)	Loads the Region Object as a HashMap. This API only loads the regions defined in application QOT. This API will get the responsibility ID and Language Code from cookie. <code>public static oracle.apps.qot.util.QotRegionManager load(java.lang.String regionCode)</code> throws FrameworkException, SQLException
load(String, int, int, String)	Loads the Region Object as a HashMap. <code>public static oracle.apps.qot.util.QotRegionManager load(java.lang.String regionCode, int respId, int appsId, java.lang.String langCode)</code> throws FrameworkException, SQLException

getDescription()

```
public java.lang.String getDescription()
```

Returns the region description.

Returns: Region description.

getMaxLength(String)

```
public int getMaxLength(java.lang.String attributeName)
```

Returns the display length for this region item.

Parameters: attribute - Attribute Name/Code.

Returns: Maximum display length for the region item.

getPrompt(String)

```
public java.lang.String getPrompt(java.lang.String attributeName)
```

```
throws FrameworkException, SQLException
```

Returns the user-language prompt for this region item.

Parameters: attribute - Attribute Name/Code.

Returns: Translated prompt.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If system error occurs.

java.sql.SQLException - If database error occurs.

is4thLevelDisabled()

```
public boolean is4thLevelDisabled()
```

Returns whether the 4th level menu is disabled.

Returns: True if the 4th level is disabled. False otherwise.

isDisplayable(String)

```
public boolean isDisplayable(java.lang.String attributeName)
```

```
throws FrameworkException, SQLException
```

Returns whether the region item is displayable.

Parameters: attributeName - Attribute name of region item.

Returns: True if the region item is displayable. False otherwise.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If system error occurs.

java.sql.SQLException - If database error occurs.

load(String)

```
public static oracle.apps.qot.util.QotRegionManager load(java.lang.String  
regionCode)
```

```
throws FrameworkException, SQLException
```

Loads the region object as a HashMap. This API only loads the regions defined in application QOT. This API will get the responsibility ID and Language Code from cookie.

Parameters: regionCode - Internal name of region.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

load(String, int, int, String)

```
public static oracle.apps.qot.util.QotRegionManager load(java.lang.String  
regionCode, int respId, int appsId, java.lang.String langCode)
```

```
throws FrameworkException, SQLException
```

To Load the Region Object as a HashMap.

Parameters:

regionCode - Internal name of Region.

respId - Responsibility Identifier.

appsId - Application Identifier.

langCode - Language Code.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If system error occurs.

java.sql.SQLException - If a database error occurs.

6.7 Class QotUtil

```
java.lang.Object  
|  
+--oracle.apps.qot.util.QotUtil
```

```
public class QotUtil
```

QotUtil provides utility methods used in Quoting. This includes methods which format numbers, retrieve a currency symbol, generate a password, retrieve a String

representation of the error message stack, retrieve a String representation of the stack trace, retrieve a profile value, write messages to the log file, and validate an email address.

Table 6–12 Inherited Member Summary

Methods inherited from class Object

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `toString()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

6.7.1 Fields for Class QotUtil

PERZ_APP_PROFILE_NAME

`public static final java.lang.String PERZ_APP_PROFILE_NAME`
Quoting personalized search profile name.

QOT_APPS_CODE

`public static final java.lang.String QOT_APPS_CODE`
Quoting application code.

QOT_APPS_ID

`public static final java.math.BigDecimal QOT_APPS_ID`
Quoting application ID.

QOT_LOG_APPL_CODE

`public static final java.lang.String QOT_LOG_APPL_CODE`
Attribute name of Quoting log application code set in the environment.

QOT_LOG_USER_FLAG

`public static final java.lang.String QOT_LOG_USER_FLAG`
Attribute name of Quoting log user flag set in the environment.

QOT_LOG_USER_NAME

`public static final java.lang.String QOT_LOG_USER_NAME`
Attribute name of Quoting log username set in the environment.

RCS_ID

`public static final java.lang.String RCS_ID`

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

6.7.2 Constructors for Class QotUtil**QotUtil()**

```
public QotUtil()
```

6.7.3 Methods for Class QotUtil

The following table is an index of the Class QotUtil methods:

Table 6–13 *Methods for Class QotUtil*

Method	Description
formatNumber(String, BigDecimal)	<p>Formats number passed in as parameter based on the currency code. Prepends currency symbol to the front of the number, adds the appropriate decimal and grouping separators. If currencyCode is null, uses Java's default formatting. If number is null, returns "" (an empty string).</p> <pre>public static java.lang.String formatNumber(java.lang.String currencyCode, java.math.BigDecimal number) throws FrameworkException, SQLException</pre>
formatNumber(String, BigDecimal, boolean)	<p>Formats number passed in as parameter based on the currency code. Based on the prependCurrencySymbol boolean parameter value, prepends currency symbol to the front of the number. Adds the appropriate decimal and grouping separators. If currencyCode is null, uses Java's default formatting. If number is null, returns "" (an empty string).</p> <pre>public static java.lang.String formatNumber(java.lang.String currencyCode, java.math.BigDecimal number, boolean prependCurrencySymbol) throws FrameworkException, SQLException</pre>

Table 6–13 Methods for Class QotUtil

Method	Description
<code>formatNumber(String, double)</code>	<p>Formats number passed in as parameter based on the currency code. Prepends currency symbol to the front of the number, adds the appropriate decimal and grouping separators</p> <pre>public static java.lang.String formatNumber(java.lang.String currencyCode, double number)</pre> <p>throws <code>FrameworkException</code>, <code>SQLException</code></p>
<code>generatePassword(int)</code>	<p>Generates a random password.</p> <pre>public static java.lang.String generatePassword(int length)</pre>
<code>getAppsContext()</code>	<p>Returns application context.</p> <pre>public static oracle.apps.jtf.base.session.FWAppsContext getAppsContext()</pre>
<code>getMasterOrg()</code>	<p>Returns master inventory organization ID for the current operating unit.</p> <pre>public static java.math.BigDecimal getMasterOrg()</pre> <p>throws <code>SQLException</code>, <code>FrameworkException</code></p>
<code>getMessageStack(FrameworkException)</code>	<p>Returns message stack from the framework exception passed in as parameter.</p> <pre>public static java.lang.String getMessageStack(oracle.apps.jtf.base.resources.F rameworkException e)</pre>
<code>getProfile(String)</code>	<p>Returns profile value using resolution of values stored at user, responsibility, application, and site level.</p> <pre>public static java.lang.String getProfile(java.lang.String profile)</pre>
<code>getSpecificProfile(String, String, String)</code>	<p>Returns profile value based on the specified user, responsibility, and application.</p> <pre>public static java.lang.String getSpecificProfile(java.lang.String profile, java.lang.String userIdStr, java.lang.String respIdStr, java.lang.String applIdStr)</pre>

Table 6–13 Methods for Class QotUtil

Method	Description
<code>getStackTrace(Exception)</code>	Returns stack trace from the exception passed in as parameter. <pre>public static java.lang.String getStackTrace(java.lang.Exception e)</pre>
<code>getSysdate()</code>	Returns database system date. <pre>public static java.sql.Timestamp getSysdate() throws FrameworkException, SQLException</pre>
<code>getSysdate(BigDecimal)</code>	Returns timestamp representing database system date plus the duration passed in as parameter. <pre>public static java.sql.Timestamp getSysdate(java.math.BigDecimal duration) throws FrameworkException, SQLException</pre>
<code>isLogEnabled</code>	Returns whether logging is enabled. <pre>public static boolean isLogEnabled()</pre>
<code>isNull(Object)</code>	Returns whether the input object is an empty or blank string or null. <pre>public static boolean isNull(java.lang.Object obj)</pre>
<code>log(String, String, int)</code>	Writes a message into the log file for diagnostic purpose. Should be called from a JSP. <pre>public static void log(java.lang.String jspPageName, java.lang.String message, int severity)</pre>
<code>log(String, String, String)</code>	Writes a message into the log file for diagnostic purpose. Should be called from a java method. <pre>public static void log(java.lang.String className, java.lang.String methodName, java.lang.String message)</pre>
<code>log(String, String, String, int)</code>	Writes a message into the log file for diagnostic purpose. Should be called from a java method. <pre>public static void log(java.lang.String className, java.lang.String methodName, java.lang.String message, int severity)</pre>

Table 6–13 Methods for Class QotUtil

Method	Description
<code>nonNull(Object)</code>	Returns a String representation of the object passed in as parameter. Does not handle <code>g_miss</code> values. <pre>public static java.lang.String nonNull(java.lang.Object obj)</pre>
<code>stringToBigDecimal(String)</code>	Converts a String into a object. <pre>public static java.math.BigDecimal stringToBigDecimal(java.lang.String str)</pre>
<code>trim(String)</code>	Returns null if input parameter <code>s</code> is null or empty; otherwise trims it. <pre>public static java.lang.String trim(java.lang.String s)</pre>
<code>useFeature(String)</code>	Checks if a particular functionality is turned on or off. The information is stored as an application level profile. <pre>public static boolean useFeature(java.lang.String feature)</pre>
<code>validateEmail(String)</code>	Validates an email string. <pre>public static java.lang.String validateEmail(java.lang.String email)</pre>

formatNumber(String, BigDecimal)

```
public static java.lang.String formatNumber(java.lang.String
currencyCode, java.math.BigDecimal number)
throws FrameworkException, SQLException
```

Formats number passed in as parameter based on the currency code. Prepends currency symbol to the front of the number, adds the appropriate decimal and grouping separators. If `currencyCode` is null, uses Java's default formatting. If number is null, returns "" (an empty string).

Parameters:

`currencyCode` - Currency code. For example, **USD**.

`number` - Number to be formatted.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

formatNumber(String, BigDecimal, boolean)

```
public static java.lang.String formatNumber(java.lang.String  
currencyCode, java.math.BigDecimal number, boolean prependCurrencySymbol)  
throws FrameworkException, SQLException
```

Formats number passed in as parameter based on the currency code. Based on the `prependCurrencySymbol` boolean parameter value, prepends currency symbol to the front of the number. Adds the appropriate decimal and grouping separators. If `currencyCode` is null, uses Java's default formatting. If `number` is null, returns "" (an empty string).

Parameters:

`currencyCode` - Currency code, such as USD.

`number` - Number to be formatted.

`prependCurrencySymbol` - Indicates whether currency symbol to be prefixed to the formatted number.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

formatNumber(String, double)

```
public static java.lang.String formatNumber(java.lang.String  
currencyCode, double number)  
throws FrameworkException, SQLException
```

Formats number passed in as parameter based on the currency code. Prepends currency symbol to the front of the number, adds the appropriate decimal and grouping separators.

Parameters:

`currencyCode` - Currency code like USD.

`number` - Number to be formatted.

Throws:

`oracle.apps.jtf.base.resources.FrameworkException` - If a system error occurs.

`java.sql.SQLException` - If a database error occurs.

generatePassword(int)

```
public static java.lang.String generatePassword(int length)
```

Generates a random password.

Parameters: length - Length of the password.

Returns: A random password.

getAppsContext()

```
public static oracle.apps.jtf.base.session.FWAppsContext getAppsContext()
```

Returns the application context.

Returns: the application context.

getMasterOrg()

```
public static java.math.BigDecimal getMasterOrg()
```

```
throws SQLException, FrameworkException
```

Return master inventory organization ID for the current operating unit.

Returns: master inventory organization ID for the current operating unit.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If system error occurs.

java.sql.SQLException - If database error occurs.

getMessageStack(FrameworkException)

```
public static java.lang.String
```

```
getMessageStack(oracle.apps.jtf.base.resources.FrameworkException e)
```

Returns message stack from the framework exception passed in as parameter.

Parameters: e - Framework exception.

Returns: String representation of the exception stack trace.

getProfile(String)

```
public static java.lang.String getProfile(java.lang.String profile)
```

Returns profile value using resolution of values stored at user, responsibility, application, and site level.

Parameters: profile - The profile name.

Returns: The profile value.

getSpecificProfile(String, String, String, String)

```
public static java.lang.String getSpecificProfile(java.lang.String profile, java.lang.String userIdStr, java.lang.String respIdStr, java.lang.String applIdStr)
```

Returns profile value based on the specified user, responsibility, and application.

Parameters:

profile - The profile name.

userIdStr - The fnd user ID.

respIdStr - The responsibility ID.

applIdStr - The application ID.

Returns: The profile value.

getStackTrace(Exception)

```
public static java.lang.String getStackTrace(java.lang.Exception e)
```

Returns a String representation of the stack trace from the exception passed in as parameter.

Parameters: e - Exception.

Returns: A String representation of the exception stack trace.

getSysdate()

```
public static java.sql.Timestamp getSysdate()  
throws FrameworkException, SQLException
```

Returns the database system date.

Returns: the database system date.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If system error occurs.

java.sql.SQLException - If database error occurs.

getSysdate(BigDecimal)

```
public static java.sql.Timestamp getSysdate(java.math.BigDecimal duration)  
throws FrameworkException, SQLException
```

Returns a timestamp representing database system date plus the duration passed in as parameter.

Parameters: duration - Duration to added to the database system date.

Returns: Timestamp representing database system date plus the duration passed in as parameter.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

isLogEnabled()

```
public static boolean isLogEnabled()
```

Returns whether logging is enabled.

Returns: True if logging is enabled. False otherwise.

isNull(Object)

```
public static boolean isNull(java.lang.Object obj)
```

Returns whether the input object is an empty or blank string or null

Parameters: obj - Object input parameter.

Returns: Whether the object is null or empty string.

log(String, String, int)

```
public static void log(java.lang.String jspPageName,  
java.lang.String message, int severity)
```

Writes a message into the log file for diagnostic purpose. Should be called from a JSP.

Parameters:

jspPageName - Name of the JSP from which this method is called.

message - The message to be logged.

severity - Severity of the message.

log(String, String, String)

```
public static void log(java.lang.String className,  
java.lang.String methodName, java.lang.String message)
```

Writes a message into the log file for diagnostic purpose. Should be called from a java method.

Parameters:

`className` - Name of the java class from which this method is called.

`methodName` - Name of the method from which this method is called.

`message` - The message to be logged.

log(String, String, String, int)

```
public static void log(java.lang.String className,  
java.lang.String methodName, java.lang.String message, int severity)
```

Writes a message into the log file for diagnostic purpose. Should be called from a java method.

Parameters:

`className` - Name of the java class from which this method is called.

`methodName` - Name of the method from which this method is called.

`message` - The message to be logged.

`severity` - Severity of the message.

nonNull(Object)

```
public static java.lang.String nonNull(java.lang.Object obj)
```

Returns a String representation of the object passed in as parameter. Does not handle `g_miss` values.

Parameters: `obj` - Object input parameter.

Returns: String representation of the object passed in as parameter with leading and trailing white spaces trimmed. If the object is null, returns an empty string.

stringToBigDecimal(String)

```
public static java.math.BigDecimal stringToBigDecimal(java.lang.String  
str)
```

Converts a String into a object.

Parameters: `str` - String object to be converted into a object.

Returns: object converted from the input string.

trim(String)

```
public static java.lang.String trim(java.lang.String s)
```

Returns null if input parameter `s` is null or empty; otherwise trim it

Parameters: `s` - String input parameter.

Returns: Trimmed input String parameter. Returns null if the input is null or empty.

useFeature(String)

```
public static boolean useFeature(java.lang.String feature)
```

Checks if a particular functionality is turned on or off. The information is stored as an application level profile.

Parameters: feature - The profile name. For example, ASO_USE_CONTRACTS.

Returns: Whether the functionality is enabled.

validateEmail(String)

```
public static java.lang.String validateEmail(java.lang.String email)
```

Validates an email string.

Returns: Null if valid; message code if invalid.

6.8 Class QueryResultSet

```
java.lang.Object
|
+--oracle.apps.qot.util.QueryResultSet
```

```
public class QueryResultSet
```

QueryResultSet is implemented to hold the result of an SQL query.

Table 6–14 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(long, int), wait(long, int), wait(long, int)

6.8.1 Fields for Class QueryResultSet

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

6.8.2 Constructors for Class QueryResultSet

QueryResultSet()

```
public QueryResultSet()
```

Constructor.

6.8.3 Methods for Class QueryResultSet

The following table is an index of the Class QueryResultSet methods:

Table 6–15 *Methods for Class QueryResultSet*

Method	Description
<code>getBatchSize()</code>	Returns the batch size (number of the rows retrieved from the query each time). <pre>public int getBatchSize()</pre>
<code>getQueryResult()</code>	Returns the array of rows returned from the SQL query. <pre>public java.lang.Object[] getQueryResult()</pre>
<code>getQueryResultCount()</code>	Returns the number of rows returned from the SQL query. Total number of objects held by QueryResult. <pre>public int getQueryResultCount()</pre>
<code>getRowCount()</code>	Returns the number of total rows returned from the SQL query. <pre>public int getRowCount()</pre>
<code>getStartIndex()</code>	Returns the start index (row number to start retrieving data). <pre>public int getStartIndex()</pre>
<code>setBatchSize(int)</code>	Set the batch size (number of the rows retrieved from the query each time). <pre>public void setBatchSize(int newBatchSize)</pre>

Table 6–15 Methods for Class QueryResultSet

Method	Description
setQueryResult(Object[])	Sets array of rows returned from the SQL query. public void setQueryResult (java.lang.Object[] newQueryResult)
setRowCount(int)	Sets the number of total rows returned from the SQL query. public void setRowCount (int newRowCount)
setStartIndex(int)	Sets the start index (row number to start retrieving data). public void setStartIndex (int newStartIndex)

getBatchSize()

```
public int getBatchSize()
```

Returns the batch size (number of the rows retrieved from the query each time).

Returns: The batch size.

getQueryResult()

```
public java.lang.Object[] getQueryResult()
```

Returns the array of rows returned from the SQL query.

Returns: An array of objects for the rows returned from the SQL query.

getQueryResultCount()

```
public int getQueryResultCount()
```

Returns the number of rows returned from the SQL query. Total number of objects held by QueryResultSet.

Returns: The number of the rows.

getRowCount()

```
public int getRowCount()
```

Returns the number of total rows returned from the SQL query.

Returns: The number of rows returned from the SQL query.

getStartIndex()

```
public int getStartIndex()
```

Returns the start index (row number to start retrieving data).

Returns: The start index.

setBatchSize(int)

```
public void setBatchSize(int newBatchSize)
```

Sets the batch size (number of the rows retrieved from the query each time).

Parameters: newBatchSize - The number of rows retrieved each time.

setQueryResult(Object[])

```
public void setQueryResult(java.lang.Object[] newQueryResult)
```

Sets the array of objects for the rows returned from the SQL query

Parameters: newQueryResult - An array of objects for the rows returned from the SQL query.

setRowCount(int)

```
public void setRowCount(int newRowCount)
```

Sets the number of total rows returned from the SQL query.

Parameters: newRowCount - The number of rows returned from the SQL query.

setStartIndex(int)

```
public void setStartIndex(int newStartIndex)
```

Sets the start index (row number to start retrieving data).

Parameters: newStartIndex - The row number to start retrieving data.

6.9 Class RequestCtx

```
java.lang.Object  
|  
+--oracle.apps.got.util.RequestCtx
```

```
public class RequestCtx
```

RequestCtx holds the session information for an HTTP request.

Table 6–16 *Inherited Member Summary*

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(long, int), wait(long, int), wait(long, int)

6.9.1 Fields for Class RequestCtx

CONFIRMATION

public static final java.lang.String **CONFIRMATION**
Indicates confirmation message.

CUSTOMER_ACCOUNT_ID

public static final java.lang.String **CUSTOMER_ACCOUNT_ID**
Cookie name for customer account ID.

CUSTOMER_TYPE

public static final java.lang.String **CUSTOMER_TYPE**
Cookie name for customer type.

DEBUG_FLAG

public static final java.lang.String **DEBUG_FLAG**
Cookie name for debug flag.

ERROR

public static final java.lang.String **ERROR**
Indicates error message.

INFORMATION

public static final java.lang.String **INFORMATION**
Indicates information message.

OPERATING_UNIT

public static final java.lang.String **OPERATING_UNIT**
Cookie name for operating unit.

QUOTE_ACCESS

public static final java.lang.String **QUOTE_ACCESS**
Cookie name for quote access level.

QUOTEHEADER_ID

public static final java.lang.String **QUOTEHEADER_ID**
Cookie name for quote header ID.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

RESOURCE_ID

```
public static final java.lang.String RESOURCE_ID
```

Cookie name for resource ID.

WARNING

```
public static final java.lang.String WARNING
```

Indicates warning message.

6.9.2 Constructors for Class RequestCtx

RequestCtx()

```
public RequestCtx()
```

6.9.3 Methods for Class RequestCtx

The following table is an index of the Class RequestCtr methods:

Table 6–17 *Methods for Class RequestCtx*

Methods	Description
<code>convertToTimestamp(String, String)</code>	<p>Converts a String into a Timestamp based on the date format passed in as parameter.</p> <pre>public static java.sql.Timestamp convertToTimestamp(java.lang.String dateString, java.lang.String dateFormat)</pre> <p>throws ParseException</p>
<code>formatTimestamp(Timestamp, String)</code>	<p>Format timestamp based on date format passed in as parameter.</p> <pre>public static java.lang.String formatTimeStamp(java.sql.Timestamp pTimeStam, java.lang.String pDateFormat)</pre>

Table 6–17 *Methods for Class RequestCtx*

Methods	Description
<code>getApplicationCode()</code>	Return the application code (eg: QOT for Quoting). <code>public static java.lang.String getApplicationCode()</code>
<code>getApplicationId()</code>	Return current application ID. <code>public static java.math.BigDecimal getApplicationId()</code>
<code>getBatchSize()</code>	Returns batch size/ number of rows to displayed. <code>public static int getBatchSize()</code>
<code>getCustomerAccountId()</code>	Return the customer account ID. <code>public static java.math.BigDecimal getCustomerAccountId()</code>
<code>getCustomerType()</code>	Return the customer type. <code>public static java.lang.String getCustomerType()</code>
<code>getDateFormat()</code>	Return user's date format. <code>public static java.lang.String getDateFormat()</code>
<code>getDefaultLocale()</code>	Returns the default java locale based on application language code. <code>public static java.util.Locale getDefaultLocale()</code>
<code>getLanguageCode()</code>	Returns user's language code. <code>public static java.lang.String getLanguageCode()</code>
<code>getNumericCharacters()</code>	Return user's numeric characters. <code>public static java.lang.String getNumericCharacters()</code>
<code>getOperatingUnit()</code>	Returns the operating unit ID. <code>public static java.math.BigDecimal getOperatingUnit()</code>

Table 6–17 *Methods for Class RequestCtx*

Methods	Description
<code>getParameter(HttpServletRequestRequest, PageContext, String)</code>	Returns the value of parameter. The parameter value in page context will get priority over the one in request. <pre>public static java.lang.String getParameter(javax.servlet.http.HttpServletRequest request, javax.servlet.jsp.PageContext pageContext, java.lang.String paramName)</pre>
<code>getQuoteAccess()</code>	Returns the quote Access level. <pre>public static java.lang.String getQuoteAccess()</pre>
<code>getQuoteHeaderId()</code>	Returns the quote header ID. <pre>public static java.math.BigDecimal getQuoteHeaderId()</pre>
<code>getResourceId()</code>	Returns the resource ID when the user is a sales rep. <pre>public static java.math.BigDecimal getResourceId()</pre>
<code>getResponsibilityId()</code>	Returns the current responsibility of the user. <pre>public static java.math.BigDecimal getResponsibilityId()</pre>
<code>getServletPath(HttpServletRequestRequest)</code>	Returns the path and jsp page name of the current request. <pre>public static java.lang.String getServletPath(javax.servlet.http.HttpServletRequest request)</pre>
<code>getUserId()</code>	Returns user ID (this maps to user_id in fnd_user table). <pre>public static java.math.BigDecimal getUserId()</pre>
<code>getUserLogFlag()</code>	Returns the user log flag. <pre>public static java.lang.String getUserLogFlag()</pre>
<code>getUserName()</code>	Returns user's login name (this maps to user_name in fnd_user table). <pre>public static java.lang.String getUserName()</pre>
<code>htmlEncode(String)</code>	Overload of <code>htmlEncode()</code> , new line characters are not converted to tags. <pre>public static java.lang.String htmlEncode(java.lang.String s)</pre>

Table 6–17 *Methods for Class RequestCtx*

Methods	Description
htmlEncode(String, boolean)	<p>Takes in a String, moves it to a char[] and then moves each character to a StringBuffer. special HTML characters are replaced with standard HTML tokens. If newLine is true, new line characters are replaced with
 tags.</p> <pre>public static java.lang.String htmlEncode(java.lang.String s, boolean newLine)</pre>
jsEncode(String)	<p>Overload of jsEncode(), new line characters are not converted to
 tags.</p> <pre>public static java.lang.String jsEncode(java.lang.String s)</pre>
jsEncode(String, boolean)	<p>Takes in a String, moves it to a char[] and then moves each character to a StringBuffer. special HTML characters are replaced with standard HTML tokens. if newLine is true, new line characters are replaced with
 tags.</p> <pre>public static java.lang.String jsEncode(java.lang.String s, boolean newLine)</pre>
log(HttpServletRequest, String, String)	<p>Writes a message into the log file for diagnostic purpose. Should be called from a jsp page.</p> <pre>public static void log(javax.servlet.http.HttpServletRequest request, java.lang.String jspPageName, java.lang.String message)</pre>
log(String,String)	<p>Writes a message into the log file for diagnostic purpose. Should be called from a jsp page.</p> <pre>public static void log(java.lang.String jspPageName, java.lang.String message)</pre>
nbsp(Object)	<p>Returns “&nbsp;” if input parameter s is null or “”.</p> <pre>public static java.lang.String nbsp(java.lang.Object o)</pre>
putMessage(PageContext, String, FrameworkException)	<p>Stores the message in the ArrayList and set it in the page context.</p> <pre>public static void putMessage(javax.servlet.jsp.PageContext pageContext, java.lang.String pMsgType, oracle.apps.jtf.base.resources.FrameworkExcep tion e)</pre>

Table 6–17 *Methods for Class RequestCtx*

Methods	Description
<code>putMessage(PageContext, String, String)</code>	Stores the message in the ArrayList and set it in the page context. <pre>public static void putMessage(javax.servlet.jsp.PageContext pageContext, java.lang.String pMsgType, java.lang.String pMsgDesc)</pre>
<code>setApplicationCode(String)</code>	Sets a application code name-value pair in the cookie. <pre>public static void setApplicationCode(java.lang.String value)</pre>
<code>setApplicationId(String)</code>	Sets application ID name-value pair in the cookie. <pre>public static void setApplicationId(java.lang.String value)</pre>
<code>setCustomerAccountId(String)</code>	Sets customer account ID name-value pair in the cookie. <pre>public static void setCustomerAccountId(java.lang.String value)</pre>
<code>setCustomerType(String)</code>	Sets customer type name-value pair in the cookie. <pre>public static void setCustomerType(java.lang.String value)</pre>
<code>setDateFormat(String)</code>	Sets the date format name-value pair in the cookie. <pre>public static void setDateFormat(java.lang.String value)</pre>
<code>setLanguageCode(String)</code>	Sets the language code name-value pair in the cookie. <pre>public static void setLanguageCode(java.lang.String value)</pre>
<code>setNumericCharacters(String)</code>	Sets the numeric characters name-value pair in the cookie. <pre>public static void setNumericCharacters(java.lang.String value)</pre>
<code>setOperatingUnit(String)</code>	Sets the operation unit ID name-value pair in the cookie. <pre>public static void setOperatingUnit(java.lang.String value)</pre>
<code>setQuoteAccess(String)</code>	Sets the quote access level name-value pair in the cookie. <pre>public static void setQuoteAccess(java.lang.String value)</pre>
<code>setQuoteHeaderId(String)</code>	Sets the quote header ID name-value pair in the cookie. <pre>public static void setQuoteHeaderId(java.lang.String value)</pre>

Table 6–17 Methods for Class RequestCtx

Methods	Description
setResourceId(String)	Sets the resource ID name-value pair in the cookie. public static void setResourceId (java.lang.String value)
setResponsibilityId(String)	Sets the responsibility ID name-value pair in the cookie. public static void setResponsibilityId (java.lang.String value)
setUserId(String)	Sets the user ID name-value pair in the cookie. public static void setUserId (java.lang.String value)
setUserLogFlag(String)	Sets the user log flag in the cookie. public static void setUserLogFlag (java.lang.String value)
setUserName(String)	Sets a username name-value pair in the cookie. public static void setUserName (java.lang.String value)
validateSrchStr(boolean, int, String)	Validates the search string. public static java.lang.String validateSrchStr (boolean allowBlankSrch, int minNumOfLetters, java.lang.String srchStr) throws FrameworkException

convertToTimestamp(String, String)

```
public static java.sql.Timestamp convertToTimestamp(java.lang.String
dateString, java.lang.String dateFormat)
throws ParseException
```

Converts a String into a Timestamp based on the date format passed in as parameter.

Parameters:

dateString - Date string to be converted to a timestamp.

dateFormat - Date format to use when converting the string to a timestamp.

Returns: Converted timestamp.

Throws: java.text.ParseException - If parsing error occurs while converting the String to a Timestamp.

formatTimeStamp(Timestamp, String)

```
public static java.lang.String formatTimeStamp(java.sql.Timestamp  
pTimeStamp, java.lang.String pDateFormat)
```

Formats a timestamp based on date format passed in as parameter.

Parameters:

pTimeStamp - Input timestamp to be formatted.

pDateFormat - Date format to use when formatting the timestamp.

Returns: Formatted timestamp based on the specified date format.

getApplicationCode()

```
public static java.lang.String getApplicationCode()
```

Returns the application code (eg: QOT for Quoting).

Returns: Application code.

getApplicationId()

```
public static java.math.BigDecimal getApplicationId()
```

Returns the current application ID.

Returns: Current application ID.

getBatchSize()

```
public static int getBatchSize()
```

getCustomerAccountId()

```
public static java.math.BigDecimal getCustomerAccountId()
```

Return the Customer account ID.

Returns: Customer account ID.

getCustomerType()

```
public static java.lang.String getCustomerType()
```

Return the customer type.

Returns: Customer type.

getDateFormat()

```
public static java.lang.String getDateFormat()
```

Returns user's date format.

Returns: User's date format.

getDefaultLocale()

```
public static java.util.Locale getDefaultLocale()
```

Returns default java locale based on application language code.

Returns: Locale Object - Default locale.

getLanguageCode()

```
public static java.lang.String getLanguageCode()
```

Return user's language code

Returns: User's language code.

getNumericCharacters()

```
public static java.lang.String getNumericCharacters()
```

Returns user's numeric characters.

Returns: user's numeric characters

getOperatingUnit()

```
public static java.math.BigDecimal getOperatingUnit()
```

Returns the operating unit ID.

Returns: operation unit ID.

getParameter(HttpServletRequest, PageContext, String)

```
public static java.lang.String
```

```
getParameter(javax.servlet.http.HttpServletRequest request,  
             javax.servlet.jsp.PageContext pageContext, java.lang.String paramName)
```

Returns the value of parameter. The parameter value in page context will get priority over the one in request.

Parameters:

request - Request.

pageContext - PageContext.

paramName - Name of the Parameter.

Returns: Value of the parameter.

getQuoteAccess()

public static java.lang.String **getQuoteAccess()**

Returns quote Access level.

Returns: Quote access level.

getQuoteHeaderId()

public static java.math.BigDecimal **getQuoteHeaderId()**

Returns the Quote Header ID.

Returns: Quote header ID.

getResourceId()

public static java.math.BigDecimal **getResourceId()**

Returns the resource ID when the user is a sales rep.

Returns: Resource ID.

getResponsibilityId()

public static java.math.BigDecimal **getResponsibilityId()**

Returns current responsibility of the user.

Returns: Current responsibility of the user.

getServletPath(HttpServletRequest)

public static java.lang.String

getServletPath(javax.servlet.http.HttpServletRequest request)

Returns the path and jsp page name of the current request.

Parameters: request - Request.

Returns: Servlet path and jsp name of the current request.

getUserId()

public static java.math.BigDecimal **getUserId()**

Returns user ID (this maps to user_id in fnd_user table).

Returns: User ID.

getUserLogFlag()

public static java.lang.String **getUserLogFlag()**

Returns the user log flag.

Returns: User log flag.

getUserName()

```
public static java.lang.String getUserName()
```

Returns the user's login name (this maps to user_name in fnd_user table).

Returns: User's login name.

htmlEncode(String)

```
public static java.lang.String htmlEncode(java.lang.String s)
```

Overload of htmlEncode(), new line characters are not converted to
 tags.

Parameters: s - String Value to be encoded.

Returns: Encoded HTML string.

htmlEncode(String, boolean)

```
public static java.lang.String htmlEncode(java.lang.String s,  
boolean newLine)
```

Takes in a String, moves it to a char[] and then moves each character to a StringBuffer. special HTML characters are replaced with standard HTML tokens. If newLine is true, new line characters are replaced with
 tags.

Parameters:

s - String value to be encoded.

newLine - Whether to replace newline characters with
 tags.

Returns: Encoded HTML string.

jsEncode(String)

```
public static java.lang.String jsEncode(java.lang.String s)
```

Overloads of jsEncode(), new line characters are not converted to
 tags.

Parameters: s - String Value to be encoded.

Returns: Encoded string.

jsEncode(String, boolean)

```
public static java.lang.String jsEncode(java.lang.String s,  
boolean newLine)
```

Takes in a String, moves it to a char[] and then moves each character to a StringBuffer. special HTML characters are replaced with standard HTML tokens. if newLine is true, new line characters are replaced with
 tags.

Parameters:

s - String value to be encoded.

newLine - Whether to replace newline characters with
 tags.

Returns: Encoded string.

log(HttpServletRequest, String, String)

```
public static void log(javax.servlet.http.HttpServletRequest request,  
java.lang.String jspPageName, java.lang.String message)
```

Writes a message into the log file for diagnostic purpose. Should be called from a jsp page.

Parameters:

request - HTTP servlet request.

jspPageName - Name of the jsp file from which the method is called.

message - The message to be logged.

log(String, String)

```
public static void log(java.lang.String jspPageName,  
java.lang.String message)
```

Writes a message into the log file for diagnostic purpose. Should be called from a jsp page.

Parameters:

jspPageName - Name of the jsp file from which the method is called.

message - The message to be logged.

nbsp(Object)

```
public static java.lang.String nbsp(java.lang.Object o)
```

Returns " " if input parameter s is null or "".

Parameters: o - Object as input parameter.

Returns: Returns " " if the input is null or empty string. Otherwise, trims the String representation of the object.

putMessage(PageContext, String, FrameworkException)

```
public static void putMessage(javax.servlet.jsp.PageContext pageContext,  
    java.lang.String pMsgType,  
    oracle.apps.jtf.base.resources.FrameworkException e)
```

Stores the Message in the ArrayList and set it in the page context.

Parameters:

pageContext - PageContext.

pMsgType - Message type.

e - FrameworkException containing message stack.

putMessage(PageContext, String, String)

```
public static void putMessage(javax.servlet.jsp.PageContext pageContext,  
    java.lang.String pMsgType, java.lang.String pMsgDesc)
```

Stores the Message in the ArrayList and set it in the page context.

Parameters:

pageContext - PageContext.

pMsgType - Message type.

pMsgDesc - Translated Message Description.

setApplicationCode(String)

```
public static void setApplicationCode(java.lang.String value)
```

Sets application code name-value pair in the cookie.

Parameters: value - Application code value to be set in the cookie.

setApplicationId(String)

```
public static void setApplicationId(java.lang.String value)
```

Sets application ID name-value pair in the cookie

Parameters: value - Application ID value to be set in the cookie.

setCustomerId(String)

```
public static void setCustomerId(java.lang.String value)
```

Sets customer account ID name-value pair in the cookie.

Parameters: value - Customer account ID value to be set in the cookie.

setCustomerType(String)

public static void **setCustomerType**(java.lang.String value)
Sets customer type name-value pair in the cookie.

Parameters: value - Customer type value to be set in the cookie.

setDateFormat(String)

public static void **setDateFormat**(java.lang.String value)
Sets date format name-value pair in the cookie.

Parameters: value - Date format value to be set in the cookie.

setLanguageCode(String)

public static void **setLanguageCode**(java.lang.String value)
Sets language code name-value pair in the cookie.

Parameters: value - Language code value to be set in the cookie.

setNumericCharacters(String)

public static void **setNumericCharacters**(java.lang.String value)
Sets numeric characters name-value pair in the cookie.

Parameters: value - Numeric characters value to be set in the cookie.

setOperatingUnit(String)

public static void **setOperatingUnit**(java.lang.String value)
Sets operation unit ID name-value pair in the cookie.

Parameters: value - Value of the operating unit ID to be set in the cookie.

setQuoteAccess(String)

public static void **setQuoteAccess**(java.lang.String value)
Sets quote access level name-value pair in the cookie.

Parameters: value - Quote access level value to be set in the cookie.

setQuoteHeaderId(String)

public static void **setQuoteHeaderId**(java.lang.String value)
Sets quote header ID name-value pair in the cookie.

Parameters: value - Quote header ID value to be set in the cookie.

setResourceId(String)

```
public static void setResourceId(java.lang.String value)  
Sets resource ID name-value pair in the cookie.
```

Parameters: value - Resource ID value to be set in the cookie.

setResponsibilityId(String)

```
public static void setResponsibilityId(java.lang.String value)  
Sets responsibility ID name-value pair in the cookie.
```

Parameters: value - Responsibility ID value to be set in the cookie.

setUserId(String)

```
public static void setUserId(java.lang.String value)  
Sets user ID name-value pair in the cookie.
```

Parameters: value - User ID value to be set in the cookie.

setUserLogFlag(String)

```
public static void setUserLogFlag(java.lang.String value)  
Sets the user log flag in the cookie.
```

Parameters: value - Value of the user log flag to be set in the cookie.

setUserName(String)

```
public static void setUserName(java.lang.String value)  
Sets a username name-value pair in the cookie.
```

Parameters: value - Username value to be set in the cookie.

validateSrchStr(boolean, int, String)

```
public static java.lang.String validateSrchStr(boolean allowBlankSrch,  
int minNumOfLetters, java.lang.String srchStr)  
throws FrameworkException  
Validates search string.
```

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If system error occurs.

oracle.apps.qot.salesupp

This section lists the Oracle Quoting Java APIs in the package oracle.apps.qot.salesupp.

7.1 Package oracle.apps.qot.salesupp

The package oracle.apps.qot.salesupp contains the APIs for Oracle Quoting sales supplement procedures. The table below lists a description for each class:

Table 7–1 Class Summary

Class	Description
Class Component	Component object is used to model a sales supplement component.
Class InstanceResponseValue	InstanceResponseValue contains the following information for responses entered for sales supplement: section component map ID, response value, response IDs, and whether there are multiple responses.
Class Response	Response contains the following information for a sales supplement response: response ID, response name.
Class SectComponent	SectComponent maintains the relationship between a section and its child components/ subsections, as well as the instance responses for section components.
Class Section	Section contains the following information for a sales supplement section: section ID and translated section name.
Class Template	Template contains the following information for a sales supplement template: template ID, template name, and section IDs of sections in the template.

Table 7-1 Class Summary

Class	Description
Class TemplInstance	TemplInstance contains the following information for a sales supplement template instance: template ID used to create the template instance, template instance ID.
Exceptions	
Class SaleSuppException	SaleSuppException is thrown when an application error occurs in a method in package oracle.apps.qot.salesupp.

7.2 Class Component

```
java.lang.Object
|
+--oracle.apps.qot.salesupp.Component
```

public class **Component**

Component object is used to model a sales supplement component. The class provides methods to retrieve a specific sales supplement component and list all sales supplement components.

Table 7-2 Inherited Member Summary

Methods inherited from class Object
equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

7.2.1 Fields for Class Component

componentDescription

```
public com.sun.java.util.collections.HashMap componentDescription
```

HashMap containing the translated component description. The key in the HashMap is language code, the value is component description.

componentId

```
public java.math.BigDecimal componentId
```

Component ID.

componentInstruction

```
public com.sun.java.util.collections.HashMap componentInstruction
```

HashMap containing the translated component instruction. The key in the HashMap is language code, the value is component instruction.

componentName

```
public com.sun.java.util.collections.HashMap componentName
```

HashMap containing the translated component name. The key in the HashMap is language code, the value is component name.

componentType

```
public java.lang.String componentType
```

Component type.

mandFlag

```
public java.lang.String mandFlag
```

Whether a response for the component is mandatory.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

responseIds

```
public com.sun.java.util.collections.ArrayList responseIds
```

Possible responses for the component if the response type is a predefined list of values.

respType

```
public java.lang.String respType
```

Response type.

7.2.2 Constructors for Class Component

Component()

```
public Component()  
Constructor.
```

7.2.3 Methods for Class Component

The following table is an index of the Class Component methods:

Table 7–3 *Methods for Class Component*

Method	Description
<code>getComponent()</code>	Returns HashMap containing the translated component name. <code>public com.sun.java.util.collections.HashMap getComponent()</code>
<code>getComponentDesc()</code>	Returns translated component description for the current language. <code>public java.lang.String getComponentDesc()</code>
<code>getComponentId()</code>	Returns the component ID. <code>public java.math.BigDecimal getComponentId()</code>
<code>getComponentInst()</code>	Returns the translated component instruction for the current language. <code>public java.lang.String getComponentInst()</code>
<code>getComponentName()</code>	Returns translated component name for the current language. <code>public java.lang.String getComponentName()</code>
<code>getDesc()</code>	Returns a HashMap containing the translated component description. <code>public com.sun.java.util.collections.HashMap getDesc()</code>

Table 7–3 Methods for Class Component

Method	Description
getInst()	Returns the component instruction. public com.sun.java.util.collections.HashMap getInst()
getResponseIds()	Returns the response IDs for the component. public com.sun.java.util.collections.ArrayList getResponseIds()
list()	Lists the available components. public static oracle.apps.qot.salesupp.Component[] list() throws FrameworkException
load(BigDecimal)	Returns the component object corresponding to the component ID passed in as parameter. public static oracle.apps.qot.salesupp.Component load (java.math.BigDecimal componentId) throws FrameworkException
setComponentId(BigDecimal)	set the component ID. public void setComponentId (java.math.BigDecimal _componentId)
setDesc(HashMap)	Sets the component description. public void setDesc (com.sun.java.util.collections.HashMap _desc)
setInst(HashMap)	Set the component instruction. public void setInst (com.sun.java.util.collections.HashMap _inst)
setName(HashMap)	Sets the name. public void setName (com.sun.java.util.collections.HashMap _name)

Table 7–3 Methods for Class Component

Method	Description
<code>setResponseIds(ArrayList)</code>	Sets the response IDs for the component. <code>public void setResponseIds(com.sun.java.util.collections. ArrayList _responseIds)</code>
<code>toString()</code>	Returns a String representation of the Component object. <code>public java.lang.String toString()</code>

getComponent()

```
public com.sun.java.util.collections.HashMap getComponent()  
Returns HashMap containing the translated component name.
```

Returns: HashMap containing the translated component name. The key in the HashMap is language code, the value is translated component name.

getComponentDesc()

```
public java.lang.String getComponentDesc()  
Returns the translated component description for the current language.
```

Returns: The translated component description for the current language.

getComponentId()

```
public java.math.BigDecimal getComponentId()  
Returns the component ID.
```

Returns: The component ID.

getComponentInst()

```
public java.lang.String getComponentInst()  
Returns the translated component instruction for the current language.
```

Returns: The translated component instruction for the current language

getComponentName()

```
public java.lang.String getComponentName()  
Returns the translated component name for the current language.
```

Returns: The translated component name for the current language

getDesc()

```
public com.sun.java.util.collections.HashMap getDesc()
```

Returns a HashMap containing the translated component description.

Returns: HashMap containing the translated component description. The key in the HashMap is language code, the value is translated component description.

getInst()

```
public com.sun.java.util.collections.HashMap getInst()
```

Returns the component instruction.

Returns: HashMap containing the translated component instruction. The key in the HashMap is language code, the value is translated component instruction.

getResponseIds()

```
public com.sun.java.util.collections.ArrayList getResponseIds()
```

Returns the response IDs for the component.

Returns: An ArrayList of possible response IDs for the component.

list()

```
public static oracle.apps.qot.salesupp.Component[] list()  
throws FrameworkException
```

Lists the available components.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(BigDecimal)

```
public static oracle.apps.qot.salesupp.Component  
load(java.math.BigDecimal componentId)  
throws FrameworkException
```

Returns a component object corresponding to the component ID passed in as parameter.

Parameters: componentId - Component ID

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setComponentId(BigDecimal)

```
public void setComponentId(java.math.BigDecimal _componentId)
```

Sets the component ID.

Parameters: `componentId` - Component ID.

setDesc(HashMap)

```
public void setDesc(com.sun.java.util.collections.HashMap _desc)
```

Sets the component description.

Parameters: `_desc` - HashMap containing the translated component description. The key in the HashMap is language code, the value is translated component description.

setInst(HashMap)

```
public void setInst(com.sun.java.util.collections.HashMap _inst)
```

Sets the component instruction.

Parameters: `_inst` - HashMap containing the translated component instruction. The key in the HashMap is language code, the value is translated component instruction.

setName(HashMap)

```
public void setName(com.sun.java.util.collections.HashMap _name)
```

Set the name.

Parameters: `_name` - HashMap containing the translated component name. The key in the HashMap is language code, the value is translated component name.

setResponseIds(ArrayList)

```
public void setResponseIds(com.sun.java.util.collections.ArrayList _responseIds)
```

Set the response IDs for the component.

Parameters: `_responseIds` - possible response IDs for the component.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the Component object.

Overrides: `toString` in class `Object`

Returns: A String representation of the Component object.

7.3 Class InstanceResponseValue

```
java.lang.Object
|
+--oracle.apps.qot.salesupp.InstanceResponseValue
```

public class InstanceResponseValue

InstanceResponseValue contains the following information for responses entered for sales supplement: section component map ID, response value, response IDs, and whether there are multiple responses.

Table 7–4 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

7.3.1 Fields for Class InstanceResponseValue

compSectMapId

```
public java.math.BigDecimal compSectMapId
```

Section component map ID.

multAnsFlag

```
public java.lang.String multAnsFlag
```

Whether there are multiple responses.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

responseId

```
public com.sun.java.util.collections.ArrayList responseId
```

Selected response IDs in the case of components with a response type of a predefined list of values.

responseValue

```
public java.lang.String responseValue  
Response value.
```

7.3.2 Constructors for Class InstanceResponseValue

InstanceResponseValue()

```
public InstanceResponseValue()  
Constructor.
```

InstanceResponseValue(BigDecimal, String, ArrayList)

```
public InstanceResponseValue(java.math.BigDecimal _sectMapId,  
java.lang.String _responseValue,  
com.sun.java.util.collections.ArrayList _responseId)  
Constructor.
```

Parameters:

_sectMapId - Section component map ID.

_responseValue - Response value.

_responseId - An ArrayList of response IDs for radio button and checkbox responses.

7.3.3 Methods for Class InstanceResponseValue

The following table is an index of the Class InstanceResponseValue methods:

Table 7–5 *Methods for Class InstanceResponseValue*

Method	Description
getInstanceValues(ArrayList st, BigDecimal)	Returns the Instance Response Values for a specified template instance and section component map IDs. <pre>public static java.util.Hashtable getInstanceValues(com.sun.java.util.collection s.ArrayList alSectCompMapId, java.math.BigDecimal templateInstanceId) throws FrameworkException, SQLException</pre>

Table 7–5 Methods for Class InstanceResponseValue

Method	Description
setResponseId(ArrayList)	Set the Response IDs in the InstanceResponseValue object. public void setResponseIds (com.sun.java.util.collections.ArrayList _responseId)
toString()	Returns a String representation of the InstanceResponseValue object. public java.lang.String toString ()

getInstanceValues(ArrayList, BigDecimal)

```
public static java.util.Hashtable
getInstanceValues(com.sun.java.util.collections.ArrayList
alSectCompMapId, java.math.BigDecimal templateInstanceId)
throws FrameworkException, SQLException
```

Returns the InstanceResponseValues objects for a specified template instance and section component map IDs.

Parameters:

alSectCompMapId - An ArrayList of section component map IDs.

templateInstanceId - Template instance ID.

Returns: Hashtable containing the instance response values for the specified template instance and section component map IDs. The key in the Hashtable is section component map ID, the value is InstanceResponseValue.

Throws:

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

java.sql.SQLException - If a database error occurs.

setResponseIds(ArrayList)

```
public void setResponseIds(com.sun.java.util.collections.ArrayList _
responseId)
```

Set the Response IDs in the InstanceResponseValue object.

Parameters: _responseId - An ArrayList of response IDs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the InstanceResponseValue object.

Overrides: toString in class Object

Returns: A String representation of the InstanceResponseValue object

7.4 Class Response

```
java.lang.Object
|
+--oracle.apps.got.salesupp.Response
```

public class **Response**

Response contains the following information for a sales supplement response: response ID, response name.

Table 7-6 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

7.4.1 Fields for Class Response

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

responseId

```
public java.math.BigDecimal responseId
```

Response ID.

responseName

```
public com.sun.java.util.collections.HashMap responseName
```

HashMap containing translated response name. The key in the HashMap is language code, the value is translated response name.

7.4.2 Constructors for Class Response

Response()

```
public Response()
```

Constructor.

Response(BigDecimal, HashMap)

```
public Response(java.math.BigDecimal _responseId,
```

```
com.sun.java.util.collections.HashMap _responseName)
```

Constructor.

Parameters:

`_responseId` - Response ID.

`_responseName` - HashMap containing translated response name. The key in the HashMap is language code, the value is translated response name.

7.4.3 Methods for Class Response

The following table is an index of the Class Response methods:

Table 7-7 Methods for Class Response

Method	Description
<code>getResponse()</code>	Returns a HashMap containing the translated response name. <pre>public com.sun.java.util.collections.HashMap getResponse()</pre>
<code>getResponseId()</code>	Returns the response ID. <pre>public java.math.BigDecimal getResponseId()</pre>
<code>getResponseName()</code>	Returns the translated response name for the current language. <pre>public java.lang.String getResponseName()</pre>
<code>load(ArrayList)</code>	Lists the responses corresponding to the response IDs passed in as parameter. <pre>public static oracle.apps.qot.salesupp.Response[] load(com.sun.java.util.collections.ArrayList alresponseId)</pre> throws <code>FrameworkException</code>

Table 7–7 Methods for Class Response

Method	Description
<code>load(BigDecimal)</code>	Returns a Response object corresponding to the response ID passed in as parameter. <pre>public static oracle.apps.got.salesupp.Response load(java.math.BigDecimal responseId) throws FrameworkException</pre>
<code>setName(HaspMap)</code>	Sets the response name. <pre>public void setName(com.sun.java.util.collections.HashMap _name)</pre>
<code>setResponse(BigDecimal)</code>	Sets the response ID. <pre>public void setResponseId(java.math.BigDecimal _responseId)</pre>
<code>toString()</code>	Returns a String representation of the Response object. <pre>public java.lang.String toString()</pre>

getResponse()

```
public com.sun.java.util.collections.HashMap getResponse()
```

Returns a HashMap containing the translated response name.

Returns: HashMap containing the translated response name. The key in the HashMap is language code, the value is translated response name.

getResponseId()

```
public java.math.BigDecimal getResponseId()
```

Returns a response ID.

Returns: A response ID.

getResponseName()

```
public java.lang.String getResponseName()
```

Returns the translated response name for the current language.

Returns: The translated response name for the current language.

load(ArrayList)

```
public static oracle.apps.got.salesupp.Response[]  
load(com.sun.java.util.collections.ArrayList alresponseId)  
throws FrameworkException
```

Lists the responses corresponding to the response IDs passed in as parameter.

Parameters: alresponseId - An ArrayList containing the list of responses.

Returns: An array of response objects corresponding to the specified response IDs.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(BigDecimal)

```
public static oracle.apps.got.salesupp.Response  
load(java.math.BigDecimal responseId)  
throws FrameworkException
```

Returns a response object corresponding to the response ID passed in as parameter.

Parameters: responseId - Response ID.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setName(HashMap)

```
public void setName(com.sun.java.util.collections.HashMap _name)  
Sets the response name.
```

Parameters: _name - HashMap containing the translated response name. The key in the HashMap is language code, the value is translated response name.

setResponseId(BigDecimal)

```
public void setResponseId(java.math.BigDecimal _responseId)  
Sets the response ID.
```

Parameters: response_id - Response ID.

toString()

```
public java.lang.String toString()  
Returns a String representation of the Response object.
```

Overrides: toString in class Object

Returns: A String representation of the Response object.

7.5 Class SaleSupException

```

java.lang.Object
|
+--java.lang.Throwable
    |
    +--java.lang.Exception
        |
        +--oracle.apps.jtf.base.resources.FrameworkException
            |
            +--oracle.apps.qot.salesupp.SaleSupException
  
```

public class **SaleSupException** extends
oracle.apps.jtf.base.resources.FrameworkException

All Implemented Interfaces: java.io.Serializable

SaleSupException is thrown when an application error occurs in a method in package oracle.apps.qot.salesupp.

Table 7–8 Inherited Member Summary

Fields inherited from interface FrameworkException

DEBUG, ERROR, FATAL, INFORMATION, NONE, WARNING, defaultMsgMgr

Methods inherited from interface FrameworkException

addException(Exception), convertException(Exception), getAllInfo(),
getCurrentMessageManager(), getExceptionStack(), getExceptionStackRec(),
getExternException(), getKey(), getMessage(), getMessageManager(), getMessageStack(),
getParameters(), getParentExcep(), getRootException(), getRootExternExcept(),
getSeverity(), getThrowerInfo(), getWholeStack(), printAllInfo(PrintStream),
printAllInfo(PrintStream), printMesg(PrintStream), printMesg(PrintStream),
printMessageStack(PrintWriter), printMessageStack(PrintWriter),
printStackTrace(PrintWriter), printStackTrace(PrintWriter),
printThrowerInfo(PrintStream), printThrowerInfo(PrintStream),
printWholeStack(PrintStream), printWholeStack(PrintStream), setCurrents(),
setStackTrace(String)

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long,
int), wait(long, int)

Methods inherited from class Throwable

fillInStackTrace(), getLocalizedMessage(), printStackTrace(), toString()

7.5.1 Fields for Class SaleSuppException

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

7.5.2 Constructors for Class SaleSuppException

SaleSuppException(Exception, String)

```
public SaleSuppException(java.lang.Exception e,  
java.lang.String errorKey)
```

Constructs an exception with the given exception and error key.

Parameters:

e - The parent exception.

errorKey - Error key.

SaleSuppException(Exception, String, Object[])

```
public SaleSuppException(java.lang.Exception e,  
java.lang.String errorKey, java.lang.Object[] params)
```

Constructs an exception with the given exception, error key and parameter.

Parameters:

e - The parent exception.

errorKey - Error key.

params - An array of tokens for errorKey.

SaleSuppException(Exception, String, String)

```
public SaleSuppException(java.lang.Exception e,  
java.lang.String errorKey, java.lang.String param)
```

Constructs an exception with the given exception, error key, and parameter.

Parameters:

e - The parent exception.

errorKey - Error key.

param - Parameter for the error key.

SaleSuppException(int, String)

```
public SaleSuppException(int err_msg_count,  
java.lang.String errorKey)  
throws FrameworkException
```

Constructs an exception with the message count and error key.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

SaleSuppException(int, String, Object[])

```
public SaleSuppException(int err_msg_count,  
java.lang.String errorKey, java.lang.Object[] params)  
throws FrameworkException
```

Constructs an exception with the message count, error key, and parameters.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

params - An array of tokens for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

SaleSuppException(int, String, String)

```
public SaleSuppException(int err_msg_count,  
java.lang.String errorKey, java.lang.String param)  
throws FrameworkException
```

Constructs an exception with the message count, error key, and parameter.

Parameters:

err_msg_count - The number of messages to be returned from the pl/sql error stack.

errorKey - Error key.

params - Token for the errorKey.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

SaleSuppException(String)

```
public SaleSuppException(java.lang.String errorKey)
```

Constructs an exception with the error key.

Parameters: errorKey - Error key.

SaleSuppException(String, Object[])

```
public SaleSuppException(java.lang.String errorKey,  
java.lang.Object[] params)
```

Constructs an exception with the error key and parameters.

Parameters:

errorKey - Error key.

params - An array of tokens for error key.

SaleSuppException(String, String)

```
public SaleSuppException(java.lang.String err_msg,  
java.lang.String errorKey)
```

Constructs an exception with the error key and error message.

Parameters:

err_msg - Error message.

errorKey - Error key.

SaleSuppException(String, String, Object[])

```
public SaleSuppException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.Object[] params)
```

Constructs an exception with the error key, error message, and parameters.

Parameters:

err_msg - Error message.

errorKey - Error key.

params - An array of tokens for errorKey.

SaleSuppException(String, String, String)

```
public SaleSuppException(java.lang.String err_msg,  
java.lang.String errorKey, java.lang.String param)
```

Constructs an exception with the error message, error key, and parameter.

Parameters:

`err_msg` - Error message.

`errorKey` - Error key.

`params` - A token for `errorKey`.

7.6 Class SectComponent

```
java.lang.Object  
|  
+--oracle.apps.qot.salesupp.SectComponent
```

public class SectComponent

SectComponent maintains the relationship between a section and its child components/ subsections, as well as the instance responses for section components. It provides the method to retrieve a sorted list of section components for a specified section and template instance.

Table 7–9 Inherited Member Summary

Methods inherited from class Object

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

7.6.1 Fields for Class SectComponent

comp

```
public oracle.apps.qot.salesupp.Component comp  
Child component.
```

compId

```
public java.math.BigDecimal compId  
Child component ID.
```

defResponseId

```
public java.math.BigDecimal defResponseId
```

Default response ID for the section component.

displaySequence

```
public java.math.BigDecimal displaySequence
```

Display sequence of the subsection or component within the section.

instRespValue

```
public oracle.apps.got.salesupp.InstanceResponseValue instRespValue
```

InstanceResponseValue for the section-component mapping.

level

```
public java.math.BigDecimal level
```

Level used for ordering.

presentStyle

```
public java.lang.String presentStyle
```

Presentation style.

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

sectCompMapId

```
public java.math.BigDecimal sectCompMapId
```

Section component mapID.

sectCompType

```
public java.lang.String sectCompType
```

Section component type.

sectId

```
public java.math.BigDecimal sectId
```

Section ID.

subSectId

```
public java.math.BigDecimal subSectId
```

Subsection ID.

7.6.2 Constructors for Class SectComponent

SectComponent()

```
public SectComponent()
```

Constructor.

7.6.3 Methods for Class SectComponent

The following table is an index of the Class SectComponent methods:

Table 7–10 Methods for Class SectComponent

Method	Description
<code>getSectComp(BigDecimal, BigDecimal)</code>	Returns the list of section components for the specified section and template instance. <pre>public static oracle.apps.got.salesupp.SectComponent[] getSectComp(java.math.BigDecimal sectionId, java.math.BigDecimal templInstId) throws FrameworkException, SQLException</pre>
<code>toString()</code>	Returns a String representation of the SectComponent object. <pre>public java.lang.String toString()</pre>

getSectComp(BigDecimal, BigDecimal)

```
public static oracle.apps.got.salesupp.SectComponent[]  
getSectComp(java.math.BigDecimal sectionId,  
java.math.BigDecimal templInstId)  
throws FrameworkException, SQLException
```

Returns the list of section components for the specified section and template instance.

Parameters:

`sectionId` - Section ID.

`templInstId` - Template instance ID.

Throws:

java.sql.SQLException - if a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the `SectComponent` object.

Overrides: `toString` in class `Object`

Returns: A String representation of the `SectComponent` object.

7.7 Class Section

```
java.lang.Object
|
+--oracle.apps.qot.salesupp.Section
```

```
public class Section
```

Section contains the following information for a sales supplement section: section ID and translated section name. It provides methods to retrieve sections based on section ID(s) passed on as parameter.

Table 7–11 Inherited Member Summary

Methods inherited from class `Object`

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

7.7.1 Fields for Class Section

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

sectionId

```
public java.math.BigDecimal sectionId  
Section ID
```

sectionName

```
public com.sun.java.util.collections.HashMap sectionName  
HashMap containing translated section name. The key in the HashMap is language  
code, the value is the translated section name.
```

7.7.2 Constructors for Class Section

Section()

```
public Section()  
Constructor.
```

Section(BigDecimal, HashMap)

```
public Section(java.math.BigDecimal _sectionId,  
com.sun.java.util.collections.HashMap _sectionName)  
Constructor.
```

Parameters:

_sectionId - Section ID.

_sectionName - HashMap containing translated section name. The key in the HashMap is language code, the value is translated section name.

7.7.3 Methods for Class Section

The following table is an index of the Class Section methods:

Table 7–12 Methods for Class Section

Method	Description
<code>getSectionId()</code>	Returns the section ID. <code>public java.math.BigDecimal getSectionId()</code>
<code>getSectionName()</code>	Returns the translated section name for the current language. <code>public java.lang.String getSectionName()</code>
<code>load(BigDecimal)</code>	Returns the array of Section objects corresponding to the section IDs passed in as parameter. <code>public static oracle.apps.qot.salesupp.Section load(java.math.BigDecimal sectionId)</code> throws <code>FrameworkException</code>
<code>load(BigDecimal[])</code>	Returns the sections corresponding to the section IDs passed in as parameter. <code>public static oracle.apps.qot.salesupp.Section[] load(java.math.BigDecimal[] sectionIds)</code> throws <code>FrameworkException</code>
<code>setName(HashMap)</code>	Sets the section name. <code>public void setName(com.sun.java.util.collections.HashMap _name)</code>
<code>setSectionId(BigDecimal)</code>	Sets the section ID. <code>public void setSectionId(java.math.BigDecimal _sectionId)</code>
<code>toString()</code>	Returns a String representation of the Section object. <code>public java.lang.String toString()</code>

getSectionId()

```
public java.math.BigDecimal getSectionId()
```

Returns the section ID.

Returns: The section ID.

getSectionName()

```
public java.lang.String getSectionName()
```

Returns the translated section name for the current language.

Returns: The translated section name for the current language.

load(BigDecimal)

```
public static oracle.apps.got.salesupp.Section  
load(java.math.BigDecimal sectionId)  
throws FrameworkException
```

Returns Section object corresponding to the Section ID passed in as parameter.

Parameters: sectionId - section ID.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(BigDecimal[])

```
public static oracle.apps.got.salesupp.Section[]  
load(java.math.BigDecimal[] sectionIds)  
throws FrameworkException
```

Returns the sections corresponding to the section IDs passed in as parameter.

Parameters: sectionIds - An array of section IDs.

Returns: An array of sections corresponding to the section IDs passed in as parameter.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setName(HashMap)

```
public void setName(com.sun.java.util.collections.HashMap _name)  
Sets the section name.
```

Parameters: _name - HashMap containing translated section name. The key in the HashMap is language code, the value is translated section name.

setSectionId(BigDecimal)

```
public void setSectionId(java.math.BigDecimal _sectionId)  
Sets the section ID.
```

Parameters: _sectionId - Section ID.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the Section object.

Overrides: toString in class Object

Returns: A String representation of the Section object.

7.8 Class Template

```
java.lang.Object
|
+--oracle.apps.qot.salesupp.Template
```

public class **Template**

Template contains the following information for a sales supplement template: template ID, template name, and section IDs of the section in the template. It provides methods to retrieve a specific template and list all available templates.

Table 7–13 Inherited Member Summary

Methods inherited from class Object

equals(Object), getClass(), hashCode(), notify(), notifyAll(), wait(long, int), wait(long, int), wait(long, int)

7.8.1 Fields for Class Template

RCS_ID

public static final java.lang.String **RCS_ID**

RCS_ID_RECORDED

public static final boolean **RCS_ID_RECORDED**

sectionIds

public java.math.BigDecimal[] **sectionIds**
Section IDs for the sections which are in the template.

templateId

public java.math.BigDecimal **templateId**
Template ID.

templateName

public com.sun.java.util.collections.HashMap **templateName**

HashMap containing translated template name. The key in the HashMap is language code, the value is translated template name.

7.8.2 Constructors for Class Template

Template()

```
public Template()
```

Constructor.

Template(BigDecimal, HashMap, BigDecimal[])

```
public Template(java.math.BigDecimal _templateId,  
com.sun.java.util.collections.HashMap _templateName,  
java.math.BigDecimal[] _sectionIds)
```

Constructor.

Parameters:

`_templateId` - Template ID.

`_templateName` - HashMap containing translated template name. The key in the HashMap is language code, the value is translated template name.

`_sectionIds` - An array of section IDs for sections which are in the template.

7.8.3 Methods for Class Template

The following table is an index of the Class Template methods:

Table 7–14 Method for Class Template

Method	Description
<code>getSectionIds()</code>	Returns section IDs from the template object. <pre>public java.math.BigDecimal[] getSectionIds()</pre>
<code>getTemplate()</code>	Returns a HashMap containing translated template name. <pre>public com.sun.java.util.collections.HashMap getTemplate()</pre>
<code>getTemplateId()</code>	Returns the template ID. <pre>public java.math.BigDecimal getTemplateId()</pre>

Table 7–14 Method for Class Template

Method	Description
<code>getTemplateName()</code>	Returns the translated template name for the current language. <code>public java.lang.String getTemplateName()</code>
<code>list()</code>	Lists the available templates. <code>public static oracle.apps.qot.salesupp.Template[] list() throws FrameworkException</code>
<code>load(BigDecimal)</code>	Returns the Template object corresponding to the template ID passed in as parameter. <code>public static oracle.apps.qot.salesupp.Template load(java.math.BigDecimal templateId) throws FrameworkException</code>
<code>setName(HashMap)</code>	Sets the name. <code>public void setName(com.sun.java.util.collections.HashMa p _name)</code>
<code>setSectionIds(BigDecimal[])</code>	Sets the section IDs in the Template object. <code>public void setSectionIds(java.math.BigDecimal[] _ sectionIds)</code>
<code>setTemplateId(BigDecimal)</code>	Sets the template ID. <code>public void setTemplateId(java.math.BigDecimal _ templateId)</code>
<code>toString()</code>	Returns a String representation of the Template object. <code>public java.lang.String toString()</code>

getSectionIds()

```
public java.math.BigDecimal[] getSectionIds()
```

Returns the section IDs from the template object.

Returns: An array of section IDs for sections which are in the template.

getTemplate()

```
public com.sun.java.util.collections.HashMap getTemplate()
```

Returns a HashMap containing the translated template name.

Returns: A HashMap containing translated template name. The key in the HashMap is language code, the value is translated template name.

getTemplateId()

```
public java.math.BigDecimal getTemplateId()
```

Returns the template ID.

Returns: The template ID.

getTemplateName()

```
public java.lang.String getTemplateName()
```

Returns the translated template name for the current language.

Returns: The translated template name for the current language.

list()

```
public static oracle.apps.got.salesupp.Template[] list()  
throws FrameworkException
```

Lists the available templates.

Returns: An array of templates.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

load(BigDecimal)

```
public static oracle.apps.got.salesupp.Template
```

```
load(java.math.BigDecimal templateId)
```

```
throws FrameworkException
```

Returns the template object corresponding to the template ID passed in as parameter.

Parameters: templateId - Template ID.

Throws: oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

setName(HashMap)

```
public void setName(com.sun.java.util.collections.HashMap _name)
```


Sets the name.

Parameters: `_name` - HashMap containing translated template name. The key in the HashMap is language code, the value is translated template name.

setSectionIds(BigDecimal[])

public void **setSectionIds**(java.math.BigDecimal[] `_sectionIds`)
Set the section IDs in the template object.

Parameters: `_sectionIds` - Section IDs for sections which are in the template.

setTemplateId(BigDecimal)

public void **setTemplateId**(java.math.BigDecimal `_templateId`)
Sets the template ID.

Parameters: `_templateId` - Template ID.

toString()

public java.lang.String **toString**()
Returns a String representation of the Template object.

Overrides: `toString` in class `Object`

Returns: A String representation of the Template object.

7.9 Class TemplInstance

```
java.lang.Object
|
+--oracle.apps.qot.salesupp.TemplInstance
```

public class TemplInstance

TemplInstance contains the following information for a template instance. This object creates the template instance, updates the responses for the given template instance ID, and loads the TemplInstance object.

Table 7–15 Inherited Member Summary

Methods inherited from class Object

`equals(Object)`, `getClass()`, `hashCode()`, `notify()`, `notifyAll()`, `wait(long, int)`, `wait(long, int)`, `wait(long, int)`

7.9.1 Fields for Class TemplInstance

RCS_ID

```
public static final java.lang.String RCS_ID
```

RCS_ID_RECORDED

```
public static final boolean RCS_ID_RECORDED
```

templateId

```
public java.math.BigDecimal templateId  
Template ID.
```

templateInstanceId

```
public java.math.BigDecimal templateInstanceId  
Template instance ID.
```

7.9.2 Constructors for Class TemplInstance

TemplInstance()

```
public TemplInstance()  
Constructor.
```

7.9.3 Methods for Class TemplInstance

The following table is an index of the Class TemplInstance methods:

Table 7–16 Methods for Class TemplInstance

Method	Description
createTemplInstance(BigDecimal, ArrayList, String, BigDecimal)	<p>Creates a template instance with the specified template and instance response values. The template instance will be associated with the object specified by the ownerTableName and ownerTableId specified. Template instance ID of the new template will be returned. This method should be called within a transaction block.</p> <pre>public static java.math.BigDecimal createTemplInst(java.math.BigDecimal templId, com.sun.java.util.collections.ArrayList instRe sVal, java.lang.String ownerTableName, java.math.BigDecimal ownerTableId) throws FrameworkException, SQLException, Sales uppException</pre>
getTemplateId()	<p>Returns the template ID.</p> <pre>public java.math.BigDecimal getTemplateId()</pre>
getTemplateInstanceId()	<p>Returns the template instance ID.</p> <pre>public java.math.BigDecimal getTemplateInstanceId()</pre>
load(String, BigDecimal)	<p>Returns the TemplInstance objects for the given ownerTableName and ownerTableId.</p> <pre>public static oracle.apps.got.salesupp.TemplInstance[] load(java.lang.String ownerTableName, java.math.BigDecimal ownerTableId) throws FrameworkException, SQLException</pre>
toString()	<p>Returns a String representation of the TemplInstance object.</p> <pre>public java.lang.String toString()</pre>

Table 7–16 Methods for Class TemplInstance

Method	Description
updateResponse(BigDecimal, ArrayList)	<p>Updates the existing the value for the given template instance ID, and ArrayList of InstanceResponseValue objects. This method should be called within a transaction block.</p> <pre>public static void updateResponse (java.math.BigDecimal templateInstanceId, com.sun.java.util.collections.ArrayList instResVal) throws FrameworkException, SQLException, SalesuppException</pre>

createTemplInst(BigDecimal, ArrayList, String, BigDecimal)

```
public static java.math.BigDecimal
createTemplInst( java.math.BigDecimal templId,
com.sun.java.util.collections.ArrayList instResVal,
java.lang.String ownerTableName, java.math.BigDecimal ownerTableId)
throws FrameworkException, SQLException, SaleSuppException
```

Creates a template instance with the specified template and instance response values. The template instance will be associated with the object specified by the ownerTableName and ownerTableId specified. Template instance ID of the new template instance will be returned. This method should be called within a transaction block.

Parameters:

templId - The template ID.

instResVal - An ArrayList of InstanceResponseValue objects.

ownerTableName - The table name to which this instance should be associated. For example, if the template instance should be associated with a quote header, ownerTableName should be ASO_QUOTE_HEADERS_ALL.

ownerTableId - The primary key in the table specified. For example, if the template instance should be associated with a quote header, ownerTableId should be the quote_header_id.

Returns: Template instance ID of the new template instance.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

SaleSuppException - If an application error occurs.

getTemplateId()

```
public java.math.BigDecimal getTemplateId()
```

Returns the template ID.

Returns: Template ID.

getTemplateInstanceId()

```
public java.math.BigDecimal getTemplateInstanceId()
```

Returns the template instance ID.

Returns: Template instance ID.

load(String, BigDecimal)

```
public static oracle.apps.qot.salesupp.TemplInstance[]  
load(java.lang.String ownerTableName,  
java.math.BigDecimal ownerTableId)  
throws FrameworkException, SQLException
```

Returns the TemplInstance objects for the given ownerTableName and ownerTableId.

Parameters:

ownerTableName - The table name to which this instance should be associated. For example, if the template instance should be associated with a quote header, ownerTableName should be ASO_QUOTE_HEADERS_ALL.

ownerTableId - Primary key in the table specified. For example, if the template instance should be associated with a quote header, ownerTableId should be the quote_header_id.

Returns: An array of TemplInstance objects.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

toString()

```
public java.lang.String toString()
```

Returns a String representation of the TemplInstance object.

Overrides: toString in class Object

Returns: A String representation of the TemplInstance object.

updateResponse(BigDecimal, ArrayList)

```
public static void updateResponse(java.math.BigDecimal  
templateInstanceId,  
com.sun.java.util.collections.ArrayList instResVal)  
throws FrameworkException, SQLException, SaleSuppException
```

Updates the existing values for the given template instance ID, and ArrayList of InstanceResponseValue objects. This method should be called within a transaction block.

Parameters:

templateInstance - Template instance ID.

instResVal - An array of InstanceResponseValue objects.

Throws:

java.sql.SQLException - If a database error occurs.

oracle.apps.jtf.base.resources.FrameworkException - If a system error occurs.

SaleSuppException - If an application error occurs.