Oracle® Cloud

Oracle Analytics Publisher Developer's Guide for Oracle Analytics Cloud

G28329-04
November 2026

Oracle Cloud Oracle Analytics Publisher Developer's Guide for Oracle Analytics Cloud,

G28329-04

Copyright © 2025, 2026, Oracle and/or its affiliates.

Contributing Authors: Oracle Analytics development, product management, and quality assurance teams

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Contents

1 Introduction to Publisher Web Services

About Publisher Web Services 1
Authenticating Web Services 2
Accessing WSDLs 2
About In-Session Methods 2

2 Data Types in Publisher Web Services

Base Data Types 1
XML-to-Java Data Type Mappings 1
Complex Data Types 2
- ArrayOf_xsd_string 3
- ArrayOfEMailDeliveryOption 3
- ArrayOfFaxDeliveryOption 4
- ArrayOfFTPDeliveryOption 4
- ArrayOfWCCDeliveryOption 4
- ArrayOfItemData 4
- ArrayOfJobInfo 4
- ArrayOfJobOutput 5
- ArrayOfJobOutputDelivery 5
- ArrayOfLocalDeliveryOption 5
- ArrayOfMetaData 5
- ArrayOfObjectStorageDeliveryOption 5
- ArrayOfParamNameValue 6
- ArrayOfPrintDeliveryOption 6
- ArrayOfString 6
- ArrayOfTemplateFormatLabelValue 6
- ArrayOfTemplateFormatLabelValues 6
- ArrayOfWCCDeliveryOption 7
- ArrayOfWebDAVDeliveryOption 7
- BIPDataSource 7
- CatalogContents 8
- CatalogObjectInfo 8
- DeliveryChannels 8

Copyright © 2025, 2026, Oracle and/or its affiliates.

DeliveryRequest 9
DeliveryServiceDefinition 9
EMailDeliveryOption 10
FaxDeliveryOption 11
FileDataSource 11
FTPDeliveryOption 11
WCCDeliveryOption 12
ItemData 12
JDBCDataSource 13
JobDetail 13
JobFilterProperties 15
JobInfo 15
JobInfoList 16
JobOutput 16
JobOutputDelivery 17
JobOutputDeliverysList 18
JobOutputsList 18
JobStatus 18
LocalDeliveryOption 19
MetaData 19
MetaDataList 19
ObjectStorageDeliveryOption 19
ParamNameValue 20
ParamNameValues 21
PrintDeliveryOption 21
ReportDataChunk 21
ReportDefinition 22
ReportRequest 23
Values for attributeFormat 24
ReportResponse 25
ScheduleRequest 25
TemplateFormatLabelValue 27
TemplateFormatsLabelValues 27
WCCDeliveryOption 28

3 Publisher ScheduleService

cancelSchedule() Method 2
cancelScheduleInSession() Method 2
deleteJobHistory() Method 3
deleteJobHistoryInSession() Method 3
deleteSchedule() Method 4

Copyright © 2025, 2026, Oracle and/or its affiliates.

deleteScheduleInSession() Method 4
deliveryService() Method 4
deliveryServiceInSession() Method 5
downloadDocumentData() Method 5
downloadDocumentDataInSession() Method 6
downloadXMLData() Method 6
downloadXMLDataInSession() Method 6
getAllJobInstanceIDs() Method 7
getAllScheduledReportInfo() Method 7
getAllScheduledReportHistory() Method 7
getAllScheduledReportHistoryInSession() Method 8
getAllScheduledReportInSession() Method 8
getDeliveryServiceDefinition() Method 9
getDeliveryServiceDefinitionInSession() Method 9
getDocumentData() Method 10
getDocumentDataInSession() Method 10
getScheduledJobInfo() Method 11
getScheduledJobInfoInSession() Method 11
getScheduledReportDeliveryInfo() Method 12
getScheduledReportDeliveryInfoInSession() Method 12
getScheduledReportOutputInfo() Method 13
getScheduledReportOutputInfoInSession() Method 13
getScheduledReportStatus() Method 13
getScheduledReportStatusInSession() Method 14
getXMLData() Method 14
getXMLDataInSession() Method 15
purgeJobHistory() Method 15
purgeJobHistoryInSession() Method 15
resendScheduledReport() Method 16
resendScheduledReportInSession() Method 16
resumeSchedule() Method 17
resumeScheduleInSession() Method 17
scheduleReport() Method 18
scheduleReportInSession() Method 18
suspendSchedule() Method 19
suspendScheduleInSession() Method 19

4 Publisher ReportService

createReportInSession() Method 2
downloadReportDataChunk() Method 3
downloadReportDataChunkInSession() Method 4

Copyright © 2025, 2026, Oracle and/or its affiliates.

getReportDefinition() Method 4
createReport() Method 5
getListOfSubjectArea() Method 5
getListOfSubjectAreaInSession() Method 6
getReportDefinitionInSession() Method 6
getReportParameters() Method 7
getReportParametersInSession() Method 7
getReportSampleData() Method 8
getReportSampleDataInSession() Method 8
getTemplate() Method 9
getTemplateInSession() Method 9
getTemplateAvailableLocales() Method 9
getTemplateAvailableLocalesInSession() Method 10
getTemplateAvailableXLIFFs() Method 10
getTemplateAvailableXLIFFsInSession() Method 11
getTemplateParameters() Method 11
getTemplateParameterInSession() Method 11
getXDOSchema() Method 12
getXDOSchemaInSession() Method 12
removeObjectFromCache() Method 13
removeObjectFromCacheInSession() Method 13
removeTemplateForReportInSession() Method 13
runDataModel() Method 14
runDataModelInSession() Method 14
runReport() Method 14
removeTemplateForReport() Method 15
runReportInSession() Method 15
updateReportDefinition() Method 16
updateReportDefinitionInSession() Method 16
updateTemplateForReport() Method 17
updateTemplateForReportInSession() Method 17
updateXLIFFForReport() Method 18
updateXLIFFForReportInSession() Method 18
uploadReportDataChunk() Method 19
uploadReportDataChunkInSession() Method 19
uploadTemplateForReport() Method 20
uploadTemplateForReportInSession() Method 21
uploadXLIFFForReport() Method 21
uploadXLIFFForReportInSession() Method 22

Copyright © 2025, 2026, Oracle and/or its affiliates.

5 Publisher SecurityService

assignRolesToUser() Method 2
checkObjectPermissions() Method 2
checkObjectPermissionsInSession() Method 3
createRole() Method 3
createUser() Method 3
deleteRole() Method 4
deleteUser() Method 4
getBIPHTTPSessionInterval() Method 4
getObjectSecurityXML() Method 5
getSecurityModel() Method 5
impersonate() Method 5
isAdmin() Method 6
isDataModelDeveloper() Method 6
isExcelAnalyzer() Method 6
isOnlineAnalyzer() Method 6
isReportDeveloper() Method 7
isScheduler() Method 7
isUserExists() Method 7
login() Method 8
logout() Method 8
notifyBIEEPreferencesUpdated() Method 8
notifyBIEEPreferencesUpdatedWithString() Method 9
removeRolesFromUser() Method 9
updateRole() Method 9
updateUser() Method 10
validateLogin() Method 10

6 Publisher CatalogService

copyObject() Method 2
copyObjectInSession() Method 2
createFolder() Method 2
createFolderInSession() Method 3
createObject() Method 3
createObjectInSession() Method 4
deleteObject() Method 5
deleteObjectInSession() Method 5
downloadObject() Method 5
downloadObjectInSession() Method 6
downloadXLIFF() Method 6

Copyright © 2025, 2026, Oracle and/or its affiliates.

downloadXLIFFInSession() Method 7
getFolderContents() Method 7
getFolderContentsInSession() Method 7
getObject() Method 8
getObjectInfo() Method 8
getObjectInfoInSession() Method 9
getObjectInSession() Method 9
objectExist() Method 10
objectExistInSession() Method 10
renameObject() Method 10
renameObjectInSession() Method 11
updateObject() Method 11
updateObjectInSession() Method 12
uploadObject() Method 12
uploadObjectInSession() Method 13
uploadXLIFF() Method 13
uploadXLIFFInSession() Method 14

Copyright © 2025, 2026, Oracle and/or its affiliates.

Preface

This guide contains information about developing applications and integrations between Oracle Analytics Publisher and other Oracle systems.

Audience

This document is intended for application developers, data service providers, and middle tier administrators who want to programatically access and use the Oracle Analytics components to create applications or integrations with other components.

Related Documents

For a full list of guides, refer to the Books tab on Oracle Analytics Cloud Help Center.

http://docs.oracle.com/en/cloud/paas/analytics-cloud/books.html

Conventions

Conventions used in this document are described in this topic.

Text Conventions

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Videos and Images

Your company can use skins and styles to customize the look of the application, dashboards, reports, and other objects. It is possible that the videos and images included in the product documentation look different than the skins and styles your company uses.

Even if your skins and styles are different than those shown in the videos and images, the product behavior and techniques shown and demonstrated are the same.

Copyright © 2025, 2026, Oracle and/or its affiliates.

1

Introduction to Publisher Web Services

This chapter provides an introduction to the Publisher web services.

Topics:

- About Publisher Web Services
- Accessing WSDLs
- About In-Session Methods

About Publisher Web Services

Publisher web services provide data types and services to programatically access and use the Publisher components to create applications or integrations with other components.

Data types include base data types and complex data types. Publisher web services also support XML-to-Java data type mappings. For more information on supported data types and mappings, see Data Types in Publisher Web Services.

Publisher provides the following public web services:

- Publisher ScheduleService: Provides methods for executing scheduler tasks such as to schedule report jobs, retrieve report outputs, and manage report history. See Publisher ScheduleService.
- Publisher ReportService: Provides methods to interact with Publisher Report object, such as to run reports, get information about reports, define and modify reports, and upload report templates. See Publisher ReportService.
- Publisher SecurityService: Provides methods for security operations, such as authentication, impersonation, login, logout, and account management. See Publisher SecurityService.
- Publisher CatalogService: Provides methods for managing report catalog objects such as folders, reports, data models, style templates, and sub-templates. Managing methods include creating, deleting, uploading and downloading report catalog objects. See Publisher CatalogService.

Many of the methods in the Publisher services are provided in pairs for stateless operation and "in session" operation. The methods for stateless operation require login credentials. The methods for "in session" operation use an existing user's session ID, which is obtained through the login() or impersonate() method of the Publisher SecurityService. For more information, see About In-Session Methods.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Authenticating Web Services

In environments that are enabled for single sign-on, the Oracle Analytics Publisher web services use OAuth authentication.

OAuth Authentication

OAuth authentication enables a client application to receive an access token from Oracle Identity Cloud Service, Identity Provider and Authorization server to access secured resources without using username and password credentials. See Authenticate in Oracle Analytics Cloud.

After you get the bearer token, send the token in the header of the web service as follows:

- Header key - Authorization.
- Header value - Bearer <*token*>. For example, Bearer `eibcccundbccikehkenekhnbhihdctbfnhjinbubgkti`.

Accessing WSDLs

After you have installed or deployed Publisher, there is a unique URL associated with Publisher's web services. Enter the URL for the Publisher server and append "/services" as follows:

http://<host>:<port>/ui/xmlpserver/services/

Use the following Web Services Description Language files (WSDLs):

- ScheduleService - v2/ScheduleService

  WSDL: http://<host>:<port>/ui/xmlpserver/services/v2/ScheduleService?wsdl
- ReportService- v2/ReportService

  WSDL: http://<host>:<port>/ui/xmlpserver/services/v2/ReportService?wsdl
- SecurityService - v2/SecurityService

  WSDL: http://<host>:<port>/ui/xmlpserver/services/v2/SecurityService?wsdl
- CatalogService - v2/CatalogService

  WSDL: http://<host>:<port>/ui/xmlpserver/services/v2/CatalogService?wsdl

About In-Session Methods

Publisher web services provide many "in session" methods such as the deliveryServiceInSession() Method, createReportInSession() Method, and copyObjectInSession() Method. In-session methods enable your applications to perform a variety of operations for active user sessions. These methods rely on the bipSessionToken string. This session acts as a proprietary token that is generated during the user login.

To leverage in-session methods, the user must log in through the Publisher SecurityService login() Method or impersonate() Method. Upon successful user authentication from SecurityService, Publisher generates a bipSessionToken string. Use the bipSessionToken string to perform all in-session operations.

Copyright © 2025, 2026, Oracle and/or its affiliates.

2

Data Types in Publisher Web Services

This topic provides details on the data types that Publisher Web services use or define.

Topics:

- Base Data Types
- XML-to-Java Data Type Mappings
- Complex Data Types

Base Data Types

Publisher Web services use the following base data types:

Table 2-1 Base Data Types

Base Type	Description	Example
xsd:boolean	Boolean	true, false
xsd:dateTime	Date and Time	2023-10-26T21:32:52
xsd:int	Integer	23
xsd:string	String	Home/Shared/HR Reports/Salary Report
xsd:base64Binary	64-bit binary	A document, such as a PDF or HTML report

XML-to-Java Data Type Mappings

The mapping between Web service XML schema data types and Java data types depends on the SOAP development environment.

Publisher Web Services use document/literal formats. The following table shows mappings for the Oracle JDeveloper environment:

Table 2-2 XML-to-Java Data Type Mappings

Base Type	Oracle JDeveloper
xsd:boolean	java.lang.Boolean
xsd:dateTime	java.util.Date
xsd:int	java.lang.Integer
xsd:string	java.lang.String
xsd:base64Binary	java.lang.Byte

Copyright © 2025, 2026, Oracle and/or its affiliates.

Complex Data Types

Publisher Web services define and use the following complex data types:

- ArrayOf_xsd_string
- ArrayOfEMailDeliveryOption
- ArrayOfFaxDeliveryOption
- ArrayOfFTPDeliveryOption
- ArrayOfItemData
- ArrayOfJobInfo
- ArrayOfJobOutput
- ArrayOfJobOutputDelivery
- ArrayOfLocalDeliveryOption
- ArrayOfMetaData
- ArrayOfObjectStorageDeliveryOption
- ArrayOfParamNameValue
- ArrayOfPrintDeliveryOption
- ArrayOfString
- ArrayOfTemplateFormatLabelValue
- ArrayOfTemplateFormatLabelValues
- BIPDataSource
- CatalogContents
- CatalogObjectInfo
- DeliveryChannels
- DeliveryRequest
- DeliveryServiceDefinition
- EMailDeliveryOption
- FaxDeliveryOption
- FileDataSource
- FTPDeliveryOption
- ItemData
- JDBCDataSource
- JobDetail
- JobFilterProperties
- JobInfo
- JobInfoList
- JobOutput
- JobOutputDelivery

Copyright © 2025, 2026, Oracle and/or its affiliates.

- JobOutputDeliverysList
- JobOutputsList
- JobStatus
- LocalDeliveryOption
- MetaData
- MetaDataList
- ObjectStorageDeliveryOption
- ParamNameValue
- ParamNameValues
- PrintDeliveryOption
- ReportDataChunk
- ReportDefinition
- ReportRequest
- ReportResponse
- ScheduleRequest
- TemplateFormatLabelValue
- TemplateFormatsLabelValues

ArrayOf_xsd_string

Use the ArrayOf_xsd_string data type to hold an array of strings, such as for objects contained in the catalog.

Fields of ArrayOf_xsd_string

Field	Description
String[] item	An array of strings.

ArrayOfEMailDeliveryOption

Use the ArrayOfEMailDeliveryOption data type to hold an array of EMailDeliveryOption objects.

Fields of ArrayOfEMailDeliveryOption

Field	Description
EMailDeliveryOption[] item	See EMailDeliveryOption.

Copyright © 2025, 2026, Oracle and/or its affiliates.

ArrayOfFaxDeliveryOption

Use the ArrayOfFaxDeliveryOption data type to hold an array of FaxDeliveryOption objects.

Fields of ArrayOfFaxDeliveryOption

Field	Description
FaxDeliveryOption[] item	See FaxDeliveryOption.

ArrayOfFTPDeliveryOption

Use the ArrayOfFTPDeliveryOption data type to hold an array of FTPDeliveryOption objects.

Fields Provided by ArrayOfFTPDeliveryOption

Field	Description
FTPDeliveryOption[] item	See FTPDeliveryOption.

ArrayOfWCCDeliveryOption

Use the ArrayOfWCCDeliveryOption data type to hold an array of WCCDeliveryOption objects.

Fields Provided by ArrayOfWCCDeliveryOption

Field	Description
WCCDeliveryOption [] item	See WCCDeliveryOption.

ArrayOfItemData

Use the ArrayOfItemData data type to hold an array of objects contained in the catalog.

Fields Provided by ArrayOfItemData

Field	Description
ItemData[] item	See ItemData.

ArrayOfJobInfo

Use the ArrayOfJobInfo data type to hold an array of JobInfo objects.

Fields Provided by ArrayOfJobInfo

Field	Description
JobInfo[] item	See JobInfo.

Copyright © 2025, 2026, Oracle and/or its affiliates.

ArrayOfJobOutput

Use the ArrayOfJobOutput data type to hold an array of JobOutput objects.

Fields Provided by ArrayOfJobOutput

Field	Description
JobOutput[] item	See JobOutput.

ArrayOfJobOutputDelivery

Use the ArrayOfJobOutputDelivery data type to hold an array of JobOutputDelivery objects.

Fields Provided by ArrayOfJobOutputDelivery

Field	Description
JobOutputDelivery[] item	See JobOutputDelivery.

ArrayOfLocalDeliveryOption

Use the ArrayOfLocalDeliveryOption data type to hold an array of LocalDeliveryOption objects.

Fields Provided by ArrayOfLocalDeliveryOption

Field	Description
LocalDeliveryOption[] item	See LocalDeliveryOption.

ArrayOfMetaData

Use the ArrayOfMetaData data type to hold an array of MetaData objects.

Fields Provided by ArrayOfMetaData

Field	Description
MetaData[] item	See MetaData.

ArrayOfObjectStorageDeliveryOption

Use the ArrayOfObjectStorageDeliveryOption data type to hold an array of ObjectStorageDeliveryOption objects.

Fields Provided by ArrayOfObjectStorageDeliveryOption

Field	Description
ObjectStorageDeliveryOption[] item	See ObjectStorageDeliveryOption.

Copyright © 2025, 2026, Oracle and/or its affiliates.

ArrayOfParamNameValue

Use the ArrayOfParamNameValue data type to hold an array of ParamNameValue objects (field name-value pairs).

Fields Provided by ArrayOfParamNameValue

Field	Description
ParamNameValue[] item	See ParamNameValue.

ArrayOfPrintDeliveryOption

Use the ArrayOfPrintDeliveryOption data type to hold an array of PrintDeliveryOption objects.

Fields Provided by ArrayOfPrintDeliveryOption

Field	Description
PrintDeliveryOption[] item	See PrintDeliveryOption.

ArrayOfString

Use the ArrayOfString data type to hold an array of strings.

Fields Provided by ArrayOfString

Field	Description
String[] item	An array of strings.

ArrayOfTemplateFormatLabelValue

Use the ArrayOfTemplateFormatLabelValue data type to hold an array of TemplateFormatLabelValue objects (template label-value pairs).

Fields Provided by ArrayOfTemplateFormatLabelValue

Field	Description
TemplateFormatLabelValue[] item	See TemplateFormatLabelValue.

ArrayOfTemplateFormatLabelValues

Use the ArrayOfTemplateFormatLabelValues data type to hold an array of TemplateFormatsLabelValues objects.

ArrayOfTemplateFormatsLabelValues is included in the ReportDefinition complex data type to contain the specific fields to describe the available template formats.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Fields Provided by ArrayOfTemplateFormatLabelValue

Field	Description
TemplateFormatsLabelValues[ ] item	See TemplateFormatsLabelValues.

ArrayOfWCCDeliveryOption

Use the ArrayOfWCCDeliveryOption data type to hold an array of WCCDeliveryOption objects.

Fields Provided by ArrayOfWCCDeliveryOption

Field	Description
WCCDeliveryOption [] item	See WCCDeliveryOption.

ArrayOfWebDAVDeliveryOption

Use the ArrayOfWebDAVDeliveryOption data type to hold an array of WebDAVDeliveryOption objects.

Fields Provided by ArrayOfWebDAVDeliveryOption

Field	Description
WebDAVDeliveryOption[] item	See WebDAVDeliveryOption.

BIPDataSource

Use the BIPDataSource data type to dynamically specify a data source when using the runReport() method.

BIPDataSource is used by the ReportRequest complex data type.

Fields Provided by BIPDataSource

Field	Description
JDBCDataSource JDBCDataSource	Contains the elements to specify a JDBC data source. See JDBCDataSource.
FileDataSource fileDataSource	Contains the elements to specify a file data source. See FileDataSource.

Copyright © 2025, 2026, Oracle and/or its affiliates.

CatalogContents

Use the CatalogContents data type to hold objects contained in the catalog.

Fields Provided by CatalogContents

Field	Description
ArrayOfItemData catalogContents	See ArrayOfItemData.

CatalogObjectInfo

Use the CatalogObjectInfo data type to return information about an object in the catalog.

This data type is returned by getObjectInfo() or getObjectInfoInSession().

Fields Provided by CatalogObjectInfo

Field	Description
String accessPermissions	The permissions that are required to access the catalog object.
String[] availableLocales	The array of locales available to the catalog object.
long creationDate	The creation date of the catalog object.
String description	The description of the catalog object.
String displayName	The display name for the catalog object.
long lastModifiedDate	The date the catalog object was last modified.
String objectAbsolutePath	The absolute path to the catalog object.
String objectName	The name of the catalog object.
String objectSubType	The subtype of the catalog object. For Folder object, object type and subtype are Folder. For Report object, object type is ReportItem and subtype is xdo. For Data Model object, object Type is ReportItem and subtype is xdm. For Style Template object, object type is StyleTemplate and subtype is RTF or XSL. For Sub Template object, object Type is SubTemplate and subtype is RTF or XSL.
String objectType	The type of catalog object. Valid values are: `xdm` (data model) `xdo` (report) `xsb` (sub-template) `xss` (style template)
String owner	The owner of the catalog object.

DeliveryChannels

Use the DeliveryChannels data type to define the specifications to deliver a report to multiple destinations.

Copyright © 2025, 2026, Oracle and/or its affiliates.

This data type is used by the DeliveryRequest and ScheduleRequest complex data types.

Fields Provided by DeliveryChannels

Field	Description
ArrayOfEMailDeliveryOption emailOptions	See ArrayOfEMailDeliveryOption.
ArrayOfFaxDeliveryOption faxOptions	See ArrayOfFaxDeliveryOption.
ArrayOfFTPDeliveryOption ftpOptions	See ArrayOfFTPDeliveryOption.
ArrayOfLocalDeliveryOption localOptions	See ArrayOfLocalDeliveryOption.
ArrayOfObjectStorageDeliveryOption objectStorageOptions	See ArrayOfObjectStorageDeliveryOption.
ArrayOfPrintDeliveryOption printOptions	See ArrayOfPrintDeliveryOption.
ArrayOfWCCDeliveryOption wccOptions	See ArrayOfWCCDeliveryOption.

DeliveryRequest

Use the DeliveryRequest data type to define the specifications to deliver a report to multiple destinations.

Fields Provided by DeliveryRequest

Field	Description
String contentType	The content type of the generated document. Possible values are: "text/html;charset=UTF-8" "text/plain;charset=UTF-8" "application/pdf" "application/vnd.ms-powerpoint" "application/vnd.ms-powerpoint" "application/vnd.ms-excel" "application/msword" "application/x-shockwave-flash" "text/xml" "message/rfc822"
DeliveryChannels deliveryChannels	See DeliveryChannels.
byte[] documentData	The output document.
BIPDataSource dynamicDataSource	See BIPDataSource.

DeliveryServiceDefinition

Use the DeliveryServiceDefinition data type to return data in the response for the getDeliveryServiceDefinition() Method.

Fields Provided by DeliveryServiceDefinition

Use this method to obtain information about the delivery servers set up for Publisher.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
ArrayOf_xsd_String EMailServerNames	The list of e-mail server names returned in the ArrayOf_xsd_string data type. See ArrayOf_xsd_string.
ArrayOf_xsd_String FTPServerNames	The list of FTP server names returned in the ArrayOf_xsd_stringdata type. See ArrayOf_xsd_string.
ArrayOf_xsd_String HTTPServerNames	The list of HTTP server names returned in the ArrayOf_xsd_string data type. See ArrayOf_xsd_string.
ArrayOf_xsd_String SFTPServerNames	The list of SFTP server names returned in the ArrayOf_xsd_string data type. See ArrayOf_xsd_string.
ArrayOf_xsd_String defaultServerNames	The list of the default server names for each defined type, returned in the ArrayOf_xsd_string data type. See ArrayOf_xsd_string.
ArrayOf_xsd_String faxServerNames	The list of fax server names returned in the ArrayOf_xsd_string data type. See ArrayOf_xsd_string.
ArrayOf_xsd_String printerNames	The list of printer names returned in the ArrayOf_xsd_stringdata type. See ArrayOf_xsd_string.

EMailDeliveryOption

Use the EMailDeliveryOption data type to define the specifications to deliver a report through e-mail.

This data type is used by ArrayOfEMailDeliveryOption.

Fields Provided by EMailDeliveryOption

Field	Description
String emailAttachmentName	The name of the attachment file.
String emailBCC	The e-mail addresses to receive blind copies of the e-mail.
String emailBody	A text string that will appear as the body of the e-mail.
String emailCC	The e-mail addresses to receive copies of the e-mail.
String emailFrom	Required. The e-mail address that will appear as the From address. If this field is empty, a SOAP fault is thrown with the following message: `Sender (From) email address is not specified.`
String emailReplyTo	The e-mail address to appear in the Reply-to field.
String emailServerName	The e-mail server name, for example: "Oracle Mail".
String emailSubject	The subject line of the e-mail.
String emailTo	Required. The addresses to which to send the e-mail. If this field is empty, a SOAP fault is thrown with the following message: `Recipient (TO) email address is not specified.`
Boolean emailDeliveryStatusNotification	Set to "true" to send email delivery status notification.
Boolean emailReturnReceipt	Set to "true" to send email return receipt.

Copyright © 2025, 2026, Oracle and/or its affiliates.

FaxDeliveryOption

Use the FaxDeliveryOption data type to define the options to set for facsimile (fax) delivery of a report.

Fields Provided by FaxDeliveryOption

This type is used by ArrayOfFaxDeliveryOption.

Field	Description
String faxNumber	Required. The number to which to send the fax (for example, `916505069560`). If this field is empty, a SOAP fault is thrown with the following message: `Fax Number is not specified.`
String faxServer	Required. The fax server (defined on the Publisher Administration page) to which to send the fax (for example, `ipp://mycupsserver:631/printers/fax2`). If this field is empty, a SOAP fault is thrown with the following message: `Fax server is not specified.`

FileDataSource

Use the FileDataSource data type to dynamically create a connection to a file data source when you run a report. You can specify a direct path to a location on your server, or indicate that the file is in the temporary directory.

This data type is used by BIPDataSource.

Fields Provided by FileDataSource

Field	Description
String dynamicDataSourcePath	To specify a path to a data source that resides on an available server, specify the full path to the data source and set temporaryDataSource to "false". For example: "D:\BI\OracleBI\xmlp\XMLP\DemoFiles\Balance.xml")" If the file is located in the system temporary directory, set temporaryDataSource to true, and specify the file name here. For example: "Balance.xml".
boolean temporaryDataSource	Set to "true" when the file data source is in the system temporary directory. Set to "false" when dynamicDataSourcePath specifies the full path.

FTPDeliveryOption

Use the FTPDeliveryOption data type to define the options to set for FTP delivery of a report. The FTP server must be set up in the Publisher Administration pages first.

Fields Provided by FTPDeliveryOption

The FTPDeliveryOption data type is used by ArrayOfFTPDeliveryOption.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
String ftpServerName	Required. The FTP Server Name as defined in the Publisher Administration page. If this field is empty, a SOAP fault is thrown with the following message: `FTP server is not specified.`
String ftpUserName	A user name for the FTP server; required only for SFTP.
String ftpUserPassword	The password for the user entered; required only for SFTP.
String remoteFile	The name to assign the file on the server. For example: report.pdf.
boolean sftpOption	Set to "true" to use secure FTP (SFTP). Set to "false" to use FTP.

WCCDeliveryOption

Use the WCCDeliveryOption data type to define the options to set for WCC delivery of a report. The administrator must first set up the WCC server.

Fields Provided by WCCDeliveryOption

This type is used in the ArrayOfWCCDeliveryOption complex data type.

Field	Description
String WCCAccount	Optional. Select an account from the WebContent Server.
String WCCAuthor	Optional. Enter the name of the author. If you don't specify an author, then the value defaults to the login name of the user.
String WCCComments	Optional. Enter comments to include with the document on the WebContent Server.
String WCCFileName	Required. Enter the name to assign to the file on the server. For example: report.pdf .
Boolean WCCIncludeMetadata	Required. Specify `True` to allow custom metadata to be sent with the document. Custom metadata is defined in the data model.
String WCCSecurityGroup	Required. Select the security group on the WebContent Server to assign to the report.
String WCCServerName	Required. Enter the name of the WebContent Server as defined in thePublisher Administration page.
String WCCTitle	Optional. Enter a title for the report. If you don't enter a title, then the Layout name is used as the title.

ItemData

Use the ItemData data type to return object metadata of an object stored in the catalog.

Fields Provided by ItemData

Field	Description
String absolutePath	The path to the catalog object. For example: /HR Manager/ HR Reports/ Employee Listing.xdo
dateTime creationDate	The creation date of the catalog object.
String displayName	The display name for the catalog object. For example: Employee Listing
String fileName	The file name for the catalog object (for example, Employee Listing.xdo).

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
dateTime lastModified	The last modified date for the catalog object.
String lastModifier	The user name of the last person to modify the object.
String owner	The user name of the owner of the object.
String parentAbsolutePath	The absolute path of the parent folder. For example, "/HR Manager/HR Reports" is the parentAbsolutePath for the report having the absolute path "/HR Manager/HR Reports/Employee Listing.xdo".
String type	The item type. Possible values are: "report" or "folder".
String description	The description of the object. For example: "This report lists all employees by department and includes parameter selections for manager and salary range."

JDBCDataSource

Use the JDBCDataSource data type to dynamically create a connection to a JDBC data source when you run a report.

This data type is used by BIPDataSource.

Fields Provided by JDBCDataSource

Field	Description
String JDBCDriverClass	The JDBC driver class for the data source (for example, oracle.jdbc.OracleDriver).
String JDBCDriverType	The driver type as String can be either jdbc or jndi.
String JDBCPassword	The password for the data source as defined in the Publisher data source definition page.
String JDBCURL	The connection string for the data source (for example, jdbc:oracle:thin:@mydatabase.foobar.com.example:1521: orcl).
String JDBCUserName	The user name for the data source as defined in the Publisher data source definition page.
String dataSourceName	The Data Source Name assigned to the data source in the Publisher data source definition page (for example, Oracle).

JobDetail

The collection of information about a job request.

Fields Provided by JobDetail

Field	Description
boolean bursting	The value `true` indicates the bursting option is enabled for the job.
String burstingParameters	The parameters for the bursting engine.
dateTime created	The date the job was created.
String dataLocator	When the storageType is DB, dataLocator is the primary key for retrieving the data from database.
boolean deleted	The value `true` indicates the job was deleted.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
String deliveryDescription	The description for the job delivery.
String deliveryParameters	The parameters for the delivery channels.
dateTime endDate	The date the job is scheduled to end.
int instanceId	The numeric identification for the job instance.
String issuer	The issuer of the job.
String jobGroup	The group to which the job belongs.
int jobId	The numeric identification for the job instance.
int jobSetId	Inactive. Do not use.
String jobType	The type of job.
dateTime lastUpdated	The date and time the job was last updated.
String locale	The locale to which the job belongs.
String notificationParameters	The notification parameters for the job.
String owner	The name of the job owner.
int parentJobId	The numeric identifier for the parent of the job.
boolean public	Whether the job is viewable by other users (true) or not (false).
String reportParameters	The parameters for the report.
String reportUrl	The URL to the job output.
String runType	The type of job (either Single or Recurring).
String scheduleContext	The context for the job schedule when external applications submit the report.
String scheduleDescription	The description of the job schedule.
String scheduleParameters	The parameters of the job schedule.
String scheduleSource	The source of the job schedule, which is used with the scheduleContext to provide information on external applications.
dateTime startDate	The date the job is scheduled to start or started.
String status	The status of the job.
String statusDetail	The details of the job status.
String storageType	The storage type for the job. Supported value is DB.
String userDescription	A user-assigned description for the job.
String userJobName	The user-assigned named for the job.
boolean xmlDataAvailable	Whether XML data for the job is available (true) or not (false).
boolean xmlDataCompressed	A value of true indicates the XML data for the job is or will be compressed.
String xmlDataContentType	The content type of the XML data.
String xschurl	The URL of the report.

Copyright © 2025, 2026, Oracle and/or its affiliates.

JobFilterProperties

The JobFilterProperties data type is used to define the filter criteria for a specific report job.

Fields Provided by JobFilterProperties

This data type is used by the following methods to identify the specific report job or jobs about which you want information returned:

- getAllScheduledReportInfo() Method
- getAllScheduledReportHistory() Method
- getAllScheduledReportHistoryInSession() Method
- getAllScheduledReportInSession() Method

Field	Description
String endTime	The time the job is scheduled to end or ended.
String endTimeOperator	The operator for endTime. Valid values are "Equals or Earlier Than", "Equals" or "Earlier Than".
long jobID	The numeric identification assigned by Publisher for the job request.
String jobName	The user-assigned job name. Valid values are "Contains" or "Equals".
String jobNameOperator	The operator of the jobName.
String owner	The name of the job's owner.
String ownerOperator	The operator for owner. Valid values are "Contains" or "Equals".
String reportName	The user-assigned named for the job.
String reportNameOperator	The operator for reportName. Valid values are "Contains" or "Equals".
String scope	The scope of the job. Valid values are "All", "Private" or "Public".
String startTime	The time the job is scheduled to start or started.
String startTimeOperator	The operator for startTime. Valid values are "Equals or Later Than", "Equals" or "Later Than".
String status	The current status for the job.

JobInfo

The collection of information about a job request.

Table 2-3 Fields Provided by JobInfo

Field	Description
boolean burstingJob	A value of `true` indicates the bursting option is enabled.
dateTime created	The date the job was created.
boolean deleted	The value of `true` indicates the job was deleted.
dateTime endDate	The date the job is scheduled to end or ended.
long instanceId	The numeric identification for the scheduled job request.
long jobID	The numeric identification assigned by Publisher to the job request.
String jobType	The type of job.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Table 2-3 (Cont.) Fields Provided by JobInfo

Field	Description
dateTime lastUpdated	The date and time the job was last updated.
String owner	The owner of the job.
long parentJobId	The numeric identification for the parent of the scheduled job request.
boolean public	True indicates the report is a member of a report set. In the current implementation this will always return false.
String reportUrl	The report absolute path URL, for example: /HR Manager/Employee Reports/Employee Salary Report.xdo.
dateTime startDate	The date the job is scheduled to start or started.
String status	The status of the scheduled job request. Valid values are: Canceled", "Done", "Scheduled", "Suspended", or "Unknown".
String statusDetail	Additional details for the status.
String userJobName	The username of the user submitting the job request.

JobInfoList

Use the JobInfoList data type to return an array of JobInfo objects.

Fields Provided by JobInfoList

Field	Description
ArrayOfJobInfo jobInfoList	An array of JobInfo objects. See ArrayOfJobInfo.

JobOutput

Use the JobOutput data type to return a description of job outputs.

Fields Provided by JobOutput

Field	Description
String burstKey	The key used to split the data for each bursted job.
dateTime created	The date the report job was created.
boolean deleted	The value "true" indicates the job output was deleted.
boolean documentDataAvailable	True indicates that the user selected the "Save Output" option when the report was scheduled.
boolean documentDataCompressed	True indicates that the document data is compressed.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
String documentDataContentType	The content type of the generated document. Possible values are: "text/html;charset=UTF-8" "text/plain;charset=UTF-8" "application/pdf" "application/vnd.ms-powerpoint" "application/vnd.ms-excel" "application/msword" "application/x-shockwave-flash" "text/xml" "message/rfc822"
long jobID	The identification number assigned to the job byPublisher.
String jobName	The user-assigned job name.
dateTime lastUpdated	The date and time the job was last updated.
long outputID	The identification of the report in history. One scheduled JobID can be associated with multiple outputIDs or historyIDs. This is because one scheduled report can be executed or republished multiple times.
String outputName	The name assigned to the output.
long parentOutputId	The output ID of the parent request.
String status	JobOutput status values are: • STATUS_RUNNING = 'R'; • STATUS_SUCCESS = 'S'; • STATUS_FAILED = 'F'; • STATUS_CANCELLING = 'G'; • STATUS_CANCELED = 'C'; • STATUS_WITH_DELIVERY_ERROR = 'D'; • STATUS_SKIPPED = 'K'; • STATUS_WARNING = 'I'; • STATUS_UNKNOWN = 'X';
String statusDetail	Detailed status information from Publisher.

JobOutputDelivery

Use the JobOutputDelivery data type to return a description of job output.

Fields Provided by JobOutputDelivery

Field	Description
dateTime created	The date the report job was created.
long deliveryID	The primary key to identify the job delivery.
byte[] deliveryParameters	The parameters for the deliverychannels.
dateTime lastUpdated	The date and time the job was last updated.
long outputID	The identification of the report in history. One scheduled JobID can be associated with multiple outputIDs or historyIDs. This is because one scheduled report can be executed or republished multiple times.
long parentDeliveryID	The delivery ID of the parent request.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
String status	JobOutputDelivery status values are: • STATUS_RUNNING = 'R'; • STATUS_SUCCESS = 'S'; • STATUS_FAILED = 'F'; • STATUS_WARNING = 'I'; • STATUS_UNKNOWN = 'X';
String statusDetail	Detailed status information from Publisher.

JobOutputDeliverysList

The JobOutputDeliverysList data type is a wrapper class implemented to return an array of JobOutputDelivery objects.

Fields Provided by JobOutputDeliverysList

Field	Description
ArrayOfJobOutputDelivery jobOutputDeliveryList	See ArrayOfJobOutputDelivery.

JobOutputsList

Use the JobOutputsList data type to return an array of JobOutput objects.

Fields Provided by JobOutputsList

Field	Description
ArrayOfJobOutput jobOutputList	See ArrayOfJobOutput.

JobStatus

Use the JobStatus data type to return the status of a job request.

Fields Provided by JobStatus

Field	Description
String jobID	The numeric identification assigned by Publisher to the job request.
String jobStatus	The current status of the job.
String message	Details on the job status.

Copyright © 2025, 2026, Oracle and/or its affiliates.

LocalDeliveryOption

The options to set for delivery of a report to the Publisher repository. The LocalDeliveryOption data type is used in the ArrayOfLocalDeliveryOption complex data type.

Fields Provided by LocalDeliveryOption

Field	Description
String destination	Required. The file path to the Publisher repository on the local server. This field supports concatenation of the directory path and the file path. If empty, a SOAP fault is thrown with the following message: `Local destination is not specified.`

MetaData

Use the MetaData data type to set the name-value pair for a MetaData object.

Fields Provided by MetaData

Field	Description
String metaDataName	The name of the metadata.
String metaDataValue	The metadata content.

MetaDataList

Use the MetaDataList data type to return a list of MetaData objects.

Fields Provided by MetaDataList

Field	Description
ArrayOfMetaData metaDataList	See ArrayOfMetaData.

ObjectStorageDeliveryOption

The options to set for delivery of reports to Object Storage. This type is used in the ArrayOfObjectStorageDeliveryOption complex data type.

Fields Provided by ObjectStorageDeliveryOption

Field	Description
String serverName	Name of the Object Storage server.
String filePrefix	String to prefix the file names.
String fileName	File name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

ParamNameValue

The ParamNameValue data type describes parameters defined for a reports and templates in Publisher.

The ParamNameValue data type is used in the ArrayOfParamNameValue, which is included in the ReportRequest, ReportDefinition, and JobInfo data types. ParamNameValue is also returned by the getTemplateParameters() Method and getReportParameters() Method.

Fields Provided by ParamNameValue

Field	Description
String UIType	The type of parameter as defined in the Publisher data model user interface. Valid values include: • Date • Hidden • Menu • Search • Text
String dataType	Valid values include: • Boolean • Date • Float • Integer • String
String dateFormatString	If UIType is "Date", this specifies the Date Format String. The date format string must be a Java date format (for example, MM-DD-YYYY).
String dateFrom	If UIType is "Date", this specifies the begin value of the date.
String dateTo	If UIType is "Date", this specifies the end value of the date.
String defaultValue	Specifies the default value of the parameter.
String fieldSize	For parameter types "Text" and "Date", specifies the text field size for the parameter.
String label	For all parameter types except "Hidden", specifies the display label for the parameter.
ArrayOfString lovLabels	If the parameter type is "Menu", specifies the values displayed in the list of values to the user.
boolean multiValuesAllowed	`True` indicates that a parameter may contain multiple values.
String name	The parameter name.
boolean refreshParamOnChange	For parameter types "Text" and "Menu", a value of `true` for this parameter indicates that other defined parameters should be refreshed when a selection is made for this parameter.
boolean selectAll	For parameter type "Menu", a value of `true` indicates that all values can be selected for the LOV.
boolean templateParam	A value of `true` indicates the parameter is defined in the RTF template.
boolean useNullForAll	For parameter type "Menu", a value of `true` indicates that a null will be passed if all values are selected for the parameter.
ArrayOfString values	See ArrayOfString.

Copyright © 2025, 2026, Oracle and/or its affiliates.

ParamNameValues

The ParamNameValues data type is a wrapper class that returns an array of ParamNameValue objects.

Fields Provided by ParamNameValues

Field	Description
ArrayOfParamNameValue listOfParamNameValues	See ArrayOfParamNameValue.

PrintDeliveryOption

Use the PrintDeliveryOption data type to set the options for printer delivery of a report. This type is used in the ArrayofPrintDeliveryOption complex data type.

Fields Provided by PrintDeliveryOption

Field	Description
String printNumberOfCopy	The number of copies to print.
String printOrientation	Valid values are "portrait" or "landscape".
String printRange	The range of pages to print. For example, "3" will print page 3; "2-5" will print pages 2-5; "1,3-5" will print pages 1 and pages 3-5.
String printSide	Valid values are: • "d_single_sided" for single-sided • "d_double_sided_l" for duplex/long edge • "d_double_sided_s" for tumble/short edge If the parameter is not specified, single-sided is used.
String printTray	Valid values are: • "t1" for "Tray 1" • "t2" for "Tray 2" • "t3" for "Tray 3" If not specified, the printer default is used.
String printerName	Required. The printer Server Name (as set up in the Publisher Administration page) to which to send the report. If empty, a SOAP fault is thrown with the following message: `Print server is not specified.`

ReportDataChunk

Use the ReportDataChunk data type to handle large report data sets, or to upload and download report data in smaller data chunks.

Fields Provided by ReportDataChunk

Field	Description
String fileID	The identifier for the data file of the report in Publisher.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
byte[] reportDataChunk	Byte[] array representing binary report data transported between the Publisher client and server.
String reportRawDataChunk	If raw XML data is used for the report, this element contains the XML data.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

ReportDefinition

Use the ReportDefinition data type to define a report object. This is the object returned by the getReportDefinition() method.

Fields Provided by ReportDefinition

Field	Description
boolean autoRun	True indicates that the report property Auto Run is turned on.
boolean cacheDocument	True indicates that the report property Enable document cache is turned on.
boolean controledByExtApp	Whether the report definition is controlled by an external application (true) or not (false).
String dataModelURL	The .xdm location from where to get the Data Model definition.
String defaultOutputFormat	The default output format. Valid values include: • `csv` (CSV) • `data` (Data) • `eText` (eText template) • `excel` (Microsoft Excel) • `excel2000` (Microsoft Excel 2000) • `flash` (Adobe Flash) • `html` (HTML) • `mhtml` (MIME HTML) • `pdf` (Adobe PDF) • `ppt` (Microsoft PowerPoint) • `rtf` (Rich Text Format)
String defaultTemplateId	The default template identified for the report.
boolean diagnostics	True indicates that diagnostics have been turned on for the report.
String ESSJobName	The ESS job name.
String ESSPackageName	The ESS package name. Used in conjunction with ESSJobName.
ArrayOfTemplateFormatsLabelValues listOfTemplateFormatsLabelValues	Passes the list of template format labels through the ArrayOfTemplateFormatsLabelValues data type. See ArrayOfTemplateFormatLabelValues.
boolean onLine	True indicates the property "Run report online" is turned on for the report.
boolean openLinkInNewWindow	True indicates the property "Open Links in New Window" is turned on for the report.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
integer parameterColumns	The value of the report property "Parameters per line."
ArrayOfString parameterNames	Passes the parameter names defined for the report through the ArrayOfString data type. See ArrayOfString.
String reportDefnTitle	Inactive. Do not use.
String reportDescription	The user-assigned description of the report.
String reportName	The user-assigned name for the report.
ArrayOfParamNameValue reportParameterNameValues	Passes the report name-value pairs through the ArrayOfParamNameValue data type. See ArrayOfParamNameValue.
String reportType	Inactive. Do not use.
boolean showControls	True indicates the report property "Show controls" has been turned on.
boolean showReportLinks	True indicates the report property "Show report links" has been turned on.
ArrayOfString templateIds	Passes the layout names of the report templates through the ArrayOfString data type. See ArrayOfString.

ReportRequest

Use the ReportRequest data type to define the settings needed to run a report. Note that allowable values for `attributeFormat` will vary according to the type of template used (for example, PDF templates can only generate PDF output.)

Fields Provided by ReportRequest

Field	Description
String attributeCalendar	The formatting calendar to use for the report request. Valid values are: "Gregorian", "Arabic Hijrah", "English Hijrah", "Japanese Imperial", "Thai Buddha", and "ROC Official".
String attributeFormat	The output format of the requested report. See Values for attributeFormat.
String attributeLocale	The locale selection for the report. Example: fr-FR
String attributeUILocale	The locale selection for the user-interface. Example: fr-FR. Required when parameter input from the user-interface is not English.
String attributeTemplate	The template to apply to the report. For example: Employeelisting.rtf.
String attributeTimeZone	Specifies the time zone to use for the request, using a supported Java time zone ID. For example: "America/Los_Angeles".
boolean byPassCache	True indicates to bypass document cache.
BIPDataSource dynamicDataSource	If the data source for the report is not defined, you can dynamically define it. See BIPDataSource.
boolean flattenXML	True indicates that the XML is to be flattened. This flag is used for the Analyzer for Microsoft Excel because Excel requires XML data type to be flattened.
ParamNameValues parameterNameValues	The parameter name-value pairs to be used in the submission of this report request, passed through the ParamNameValues data type. See ParamNameValues.
String reportAbsolutePath	The absolute path to the report in the Publisher repository. For example: /HR Manager/HR Reports/Employee Listing.xdo.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
byte[] reportData	If you are providing the data directly for the report use this element to pass the data.
String reportOutputPath	Specifies the output path for the generated report.
String reportRawData	If raw XML data is used for the report, this element contains the XML data.
integer sizeOfDataChunkDownload	If you set flattenXML to true, or if you do not want to chunk the data, set this parameter to -1 to return all data back to the client.
MetaDataList XDOPropertyList	See MetaDataList.

Values for attributeFormat

The following table lists the valid values for attributeFormat and the template types that support each output format type.

Table 2-4 Valid Values for attributeFormat

Output Format	Value to Enter for attributeFormat Field	Template Types That Can Generate This Output Format
Interactive	Interactive	BI Publisher
HTML	html	BI Publisher, RTF, XSL Stylesheet (FO)
PDF	pdf	BI Publisher, RTF, PDF, Flash, XSL Stylesheet (FO)
RTF	rtf	BI Publisher, RTF, XSL Stylesheet (FO)
Excel (mhtml)	excel	BI Publisher, RTF, Excel, XSL Stylesheet (FO)
Excel (html)	excel2000	BI Publisher, RTF, Excel, XSL Stylesheet (FO)
Excel (*.xlsx)	xlsx	BI Publisher, RTF, XSL Stylesheet (FO)
PowerPoint (mhtml)	ppt	BI Publisher, RTF, XSL Stylesheet (FO)
PowerPoint (.*pptx)	pptx	BI Publisher, RTF, XSL Stylesheet (FO)
MHTML	mhtml	BI Publisher, RTF, Flash, XSL Stylesheet (FO)
PDF/A	pdfa	BI Publisher, RTF, XSL Stylesheet (FO)
PDF/X	pdfx	BI Publisher, RTF, XSL Stylesheet (FO)
Zipped PDFs	pdfz	BI Publisher, RTF, PDF, XSL Stylesheet (FO)
FO Formatted XML	xslfo	BI Publisher, RTF, XSL Stylesheet (FO)
Data (XML)	xml	BI Publisher, RTF, PDF, Excel, Flash, XSL Stylesheet (FO), Etext, XSL Stylesheet (HTML XML/Text)
Data (CSV)	csv	BI Publisher, RTF, PDF, Excel, Flash, XSL Stylesheet (FO), XSL Stylesheet (HTML XML/Text), Etext
XML	xml	XSL Stylesheet (HTML XML/Text)
Text	text	XSL Stylesheet (HTML XML/Text), Etext
Flash	flash	Flash

Copyright © 2025, 2026, Oracle and/or its affiliates.

ReportResponse

ReportResponse is the object returned from runReport.

Fields Provided by ReportResponse

Field	Description
MetaDataList metaDataList	See MetaDataList.
byte[] reportBytes	The report binary data output.
String reportContentType	The report content type. Possible values include: "text/html;charset=UTF-8" "text/plain;charset=UTF-8" "application/pdf" "application/vnd.ms-powerpoint" "application/vnd.ms-excel" "application/msword" "application/x-shockwave-flash" "text/xml" "message/rfc822"
String reportFileID	The numeric identification for the report file.
String reportLocale	The locale selected for the report (for example, `fr_FR`).

ScheduleRequest

The options to schedule a report.

Fields Provided by ScheduleRequest

Field	Description
boolean bookBindingOutputOption	Whether the book binding output is enabled (true) or not (false).
boolean compressDeliveryOutputOption	Whether the compress delivery output is enabled (true) or not (false).
String dataModelUrl	The location of the .xdm file from which to obtain the Data Model definition.
DeliveryChannels deliveryChannels	See DeliveryChannels.
boolean enableConsolidatedJobDiagnostic	Whether the consolidated job diagnostic is enabled (true) or not (false).
boolean enableDataEngineDiagnostic	Whether the data engine diagnostic is enabled (true) or not (false).
boolean enableReportProcessorDiagnostic	Whether the report processing diagnostic is enabled (true) or not (false).
boolean enableSQLExplainPlan	Whether the SQL Explain Plan generation is enabled (true) or not (false).

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
boolean enableXmlPruning	Whether the XML pruning is enabled (true) or not (false).
String endDate	The end date of the schedule.
String jobLocale	The locale to use for the scheduled requests. Example: fr-FR. Required when parameter values submitted for the scheduled job are not English.
String jobTZ	The time zone to use for the scheduled requests.
boolean mergeOutputOption	Whether the merge output option is enabled (true) or not (false).
String notificationPassword	The HTTP notification server password when scheduling notification through an HTTP server.
String notificationServer	The name of the HTTP server used for notification.
String notificationTo	E-mail addresses to which to send notifications.
String notificationUserName	The user name for the HTTP server used for notification.
boolean notifyHttpWhenFailed	True indicates to send a notification when the job request fails.
boolean notifyHttpWhenSkipped	Optional. Specify true to send an HTTP notification when a job is skipped. Default is false.
boolean notifyHttpWhenSuccess	True indicates to send a notification when the job request succeeds.
boolean notifyHttpWhenWarning	True indicates to send a notification when the job completes with a warning.
boolean notifyWhenFailed	True indicates to send a notification when the job request fails.
boolean notifyWhenSkipped	Optional. Specify true to send an email notification when a job is skipped. Default is false.
boolean notifyWhenSuccess	True indicates to send a notification when the job request succeeds.
boolean notifyWhenWarning	True indicates to send a notification when the job completes with a warning.
String recurrenceExpression	The expression that defines a recurring schedule.
String recurrenceExpressionType	The type of expression defined for a recurring schedule. Valid value is `cron`.
integer repeatCount	The number of times to repeat the schedule. For the recursive scheduling of a report, startDate must not be null, and repeatCount repeatInterval should be greater than 0 for any meaningful schedule. The endDate can be null.
integer repeatInterval	The interval between two scheduled jobs in seconds.
ReportRequest reportRequest	Information about the request included through the ReportRequest data type. See ReportRequest.
boolean saveDataOption	True indicates that the report data from the scheduled request run will be saved.
boolean saveOutputOption	Optional. Specify true to save the report output generated by the scheduled request to a database. Default is false.
boolean scheduleBurstingOption	Optional. Specify true to enable bursting for the scheduled job. Default is false.
boolean scheduleChunkingOption	Whether the data chunking is enabled (true) or not (false).
boolean schedulePublicOption	True indicates that the scheduled requests are to be made public.
String startDate	The date on which the schedule starts.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Field	Description
boolean useUTF8Option	True indicates that the Use UTF8 option is enabled.
String userJobDesc	The user-entered description for the scheduled job.
String userJobName	The user-entered name for the scheduled job.

TemplateFormatLabelValue

To specify the template format labels and values for a report. TemplateFormatLabelValue is included in the ArrayOfTemplateFormatLabelValue complex data type.

Fields Provided by TemplateFormatLabelValue

Field	Description
String templateFormatLabel	The label that displays for a template format. Examples include: "HTML" "PDF" "Excel"
String templateFormatValue	The template format value that corresponds to the label. Examples include: "html" "pdf" "excel"

TemplateFormatsLabelValues

Provides detailed information about template formats stored in the Publisher repository. TemplateFormatsLabelValues is included in the ArrayOfTemplateFormatLabelValues complex data type.

Fields Provided by TemplateFormatsLabelValues

Field	Description
boolean active	Whether the template is active (true) or not (false).
boolean applyStyleTemplate	Whether to apply the style template (true) or not (false).
boolean default	Whether the template is the default template (true) or not (false).
ArrayOfTemplateFormatLabelValue listOfTemplateFormatLabelValue	Contains the TemplateFormatLabelValue label-value pairs. See ArrayOfTemplateFormatLabelValue.
ArrayOfString templateAvailableLocales	The available locale options defined for a template passed in the ArrayOfString data type. See ArrayOfString.
String templateBaseLocale	The base locale options defined for a template.
String templateDefaultLocale	The default locale options defined for a template.
String templateID	The name assigned to the template in Publisher for example: "Employee Listing".
String templateType	The type of template, for example: "rtf" or "pdf".
String templateURL	The template file name in the Publisher repository, for example: "Employee Listing.rtf".
boolean viewOnline	Whether the template can be viewed online (true) or not (false).

Copyright © 2025, 2026, Oracle and/or its affiliates.

WCCDeliveryOption

Use the WCCDeliveryOption data type to define the options to set for WCC delivery of a report. The administrator must first set up the WCC server.

Fields Provided by WCCDeliveryOption

This type is used in the ArrayOfWCCDeliveryOption complex data type.

Field	Description
String WCCAccount	Optional. Select an account from the WebContent Server.
String WCCAuthor	Optional. Enter the name of the author. If you don't specify an author, then the value defaults to the login name of the user.
String WCCComments	Optional. Enter comments to include with the document on the WebContent Server.
String WCCFileName	Required. Enter the name to assign to the file on the server. For example: report.pdf .
Boolean WCCIncludeMetadata	Required. Specify `True` to allow custom metadata to be sent with the document. Custom metadata is defined in the data model.
String WCCSecurityGroup	Required. Select the security group on the WebContent Server to assign to the report.
String WCCServerName	Required. Enter the name of the WebContent Server as defined in thePublisher Administration page.
String WCCTitle	Optional. Enter a title for the report. If you don't enter a title, then the Layout name is used as the title.

Copyright © 2025, 2026, Oracle and/or its affiliates.

3

Publisher ScheduleService

This topic provides details on the ScheduleService methods that you can use to interact with the Publisher scheduler. This includes methods for scheduling report jobs, retrieving report outputs, and managing report histories.

Topics:

- cancelSchedule() Method
- cancelScheduleInSession() Method
- deleteJobHistory() Method
- deleteJobHistoryInSession() Method
- deleteSchedule() Method
- deleteScheduleInSession() Method
- deliveryService() Method
- deliveryServiceInSession() Method
- downloadDocumentData() Method
- downloadDocumentDataInSession() Method
- downloadXMLData() Method
- downloadXMLDataInSession() Method
- getAllJobInstanceIDs() Method
- getAllScheduledReportInfo() Method
- getAllScheduledReportHistory() Method
- getAllScheduledReportHistoryInSession() Method
- getAllScheduledReportInSession() Method
- getDeliveryServiceDefinition() Method
- getDeliveryServiceDefinitionInSession() Method
- getDocumentData() Method
- getDocumentDataInSession() Method
- getScheduledJobInfo() Method
- getScheduledJobInfoInSession() Method
- getScheduledReportDeliveryInfo() Method
- getScheduledReportDeliveryInfoInSession() Method
- getScheduledReportOutputInfo() Method
- getScheduledReportOutputInfoInSession() Method
- getScheduledReportStatus() Method
- getScheduledReportStatusInSession() Method

Copyright © 2025, 2026, Oracle and/or its affiliates.

- getXMLData() Method
- getXMLDataInSession() Method
- purgeJobHistory() Method
- purgeJobHistoryInSession() Method
- resendScheduledReport() Method
- resendScheduledReportInSession() Method
- resumeSchedule() Method
- resumeScheduleInSession() Method
- scheduleReport() Method
- scheduleReportInSession() Method
- suspendSchedule() Method
- suspendScheduleInSession() Method

cancelSchedule() Method

In ScheduleService, use the cancelSchedule() method to cancel a currently running scheduled job.

Signature

boolean cancelSchedule(String jobInstanceID, String userID, String password);

Parameters for cancelSchedule() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job to be canceled. The jobInstanceID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

cancelScheduleInSession() Method

In ScheduleService, use the cancelScheduleInSession() method to cancel the schedule associated with the bipSessionToken string for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

String cancelScheduleInSession(String jobInstanceID, String bipSessionToken);

Parameters for cancelScheduleInSession() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job that generated the output. The jobInstanceID is a string of integers.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

deleteJobHistory() Method

In ScheduleService, use the deleteJobHistory() method to perform a "soft" delete of the historical information of a report job, as opposed to the purgeJobHistory() method, which performs a "hard" (permanent) deletion. The deleteJobHistory() method must precede the purgeJobHistory() method.

Signature

boolean deleteJobHistory(String instanceJobID, String userID, String password);

Parameters for deleteJobHistory() Method

Parameter	Description
String instanceJobID	The ID assigned to the instance of the job that generated the output. The instanceJobID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

deleteJobHistoryInSession() Method

In ScheduleService, use the deleteJobHistoryInSession() method to delete the job history associated with the bipSessionToken string for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean deleteJobHistoryInSession(String jobInstanceID, String bipSessionToken);

Parameters for deleteJobHistoryInSession() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job that generated the output. The jobInstanceID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

deleteSchedule() Method

In ScheduleService, use the deleteSchedule() method to delete a scheduled job from the scheduler queue.

Signature

boolean deleteSchedule(String jobInstanceID, String userID, String password);

Parameters for deleteSchedule() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job to be deleted. The jobInstanceID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

deleteScheduleInSession() Method

In ScheduleService, use the deleteScheduleInSession() method to delete the schedule associated with the bipSessionToken string for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean deleteScheduleInSession(String jobInstanceID, String bipSessionToken);

Parameters for deleteScheduleInSession() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job that generated the output. The jobInstanceID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

deliveryService() Method

In ScheduleService, use the deliveryService() method to deliver a document from source to destination through the specified delivery channel.

Signature

String deliveryService(DeliveryRequest deliveryRequest, String userID, String password);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for deliveryService() Method

Parameter	Description
DeliveryRequest deliveryRequest	The DeliveryRequest object. See DeliveryRequest.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

deliveryServiceInSession() Method

In ScheduleService, use the deliveryServiceInSession() method to deliver a document associated with the deliveryRequest and bipSessionToken string for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

String deliveryServiceInSession(String jobInstanceID, String bipSessionToken);

Parameters for deliveryServiceInSession() Method

Parameter	Description
DeliveryRequest deliveryRequest	The DeliveryRequest object. See DeliveryRequest.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

downloadDocumentData() Method

In ScheduleService, use the downloadDocumentData() method to save a report document into the local temporary directory, and return the fileID of the user for later downloads. This implementation is for performance concern in case that report data size is significant.

Signature

String downloadDocumentData(String JobOutputID, String userID, String password);

Parameters for downloadDocumentData() Method

Parameter	Description
String jobOutputID	The ID assigned to the output. The jobOutputID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

downloadDocumentDataInSession() Method

In ScheduleService, use the downloadDocumentDataInSession() method to download the document report associated with the jobInstanceID and bipSessionToken string for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

String downloadDocumentDataInSession(String jobInstanceID, String bipSessionToken);

Parameters for downloadDocumentDataInSession() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job that generated the output. The jobInstanceID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

downloadXMLData() Method

In ScheduleService, use the downloadXMLData() method to return XML data used to generate a report document. It returns raw data in XML format.

Signature

downloadXMLData(String jobInstanceID, String userID, String password);

Parameters for downloadXMLData() Method

Parameter	Description
String jobInstanceID	The ID assigned to the XML data. The jobInstanceID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

downloadXMLDataInSession() Method

In ScheduleService, use the downloadXMLDataInSession() method to download the XML data for a document report associated with the bipSessionToken string for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

String downloadXMLDataInSession(String jobInstanceID, String bipSessionToken);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for downloadXMLDataInSession() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job that generated the output. The jobInstanceID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getAllJobInstanceIDs() Method

In ScheduleService, use the getAllJobInstanceIDs() method to get all the instance IDs of a submitted job.

Signature

public String[] getAllJobInstanceIDs(String submittedJobId);

Parameters for getAllJobInstanceIDs() Method

Parameter	Description
String submittedJobId	The job ID of the submitted job.

getAllScheduledReportInfo() Method

In String, use the getAllScheduledReportInfo() method to return information about all scheduled report jobs that match filter criteria passed through the JobFilterProperties object.

Signature

JobInfosList getAllScheduledReportInfo(JobFilterProperties filter, int beginIdx, String userID, String password);

Parameters for getAllScheduledReportInfo() Method

Parameter	Description
JobFilterProperties filter	The JobFilterProperties object specifies the specific criteria for the report jobs you want to return information about. See JobFilterProperties.
int beginIdx	The starting point of the index (default is 1).
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getAllScheduledReportHistory() Method

In ScheduleService, use the getAllScheduledReportHistory() method to return information about scheduled report histories that match filter criteria passed through the JobFilterProperties object.

Copyright © 2025, 2026, Oracle and/or its affiliates.

The getAllScheduledReportHistory method has a limit of 50 records per call. To return more than 50 records you must write a loop.

Signature

JobInfosList getAllScheduledReportHistory(JobFilterProperties filter, int beginIdx, String userID, String password);

Parameters for getAllScheduledReportHistory() Method

Parameter	Description
JobFilterProperties filter	The JobFilterProperties object specifies the specific criteria for the report jobs you want to return information about. See JobFilterProperties.
int beginIdx	The starting point of the index (default is 1).
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getAllScheduledReportHistoryInSession() Method

In ScheduleService, use the getAllScheduledReportHistoryInSession() method to return information about scheduled report histories that match filter criteria passed through the JobFilterProperties object and that are based on the bipSessionToken string for a given user.

The getAllScheduledReportHistoryInSession method has a limit of 50 records per call. To return more than 50 records you must write a loop.

For more information about in-session methods, see About In-Session Methods.

Signature

JobInfosList getAllScheduledReportHistoryInSession(JobFilterProperties filter, int beginIdx, String bipSessionToken);

Parameters for getAllScheduledReportHistoryInSession() Method

Parameter	Description
JobFilterProperties filter	The JobFilterProperties object specifies the specific criteria for the report jobs you want to return information about. See JobFilterProperties.
int beginIdx	The starting point of the index (default is 1).
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getAllScheduledReportInSession() Method

In ScheduleService, use the getAllScheduledReportInSession() method to return information about all scheduled reports that match filter criteria passed through the JobFilterProperties object and that are based on the bipSessionToken string for a given user.

For more information about in-session methods, see About In-Session Methods.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Signature

JobInfosList getAllScheduledReportInSession(JobFilterProperties filter, int beginIdx, String bipSessionToken);

Parameters for getAllScheduledReportInSession() Method

Parameter	Description
JobFilterProperties filter	The JobFilterProperties object specifies the specific criteria for the report jobs you want to return information about. See JobFilterProperties.
int beginIdx	The starting point of the index (default is 1).
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getDeliveryServiceDefinition() Method

In ScheduleService, use the getDeliveryServiceDefinition() method to get the delivery service definition for a given userID and password.

See deliveryService() Method.

Signature

DeliveryServiceDefinition getDeliveryServiceDefinition(String userID, String password);

Parameters for getDeliveryServiceDefinition() Method

Parameter	Description
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getDeliveryServiceDefinitionInSession() Method

In ScheduleService, use the getDeliveryServiceDefinitionInSession() method to get the delivery service definition based on the bipSessionToken of a given user.

See deliveryService() Method.

For more information about in-session methods, see About In-Session Methods.

Signature

DeliveryServiceDefinition getDeliveryServiceDefinitionInSession(String bipSessionToken);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for getDeliveryServiceDefinitionInSession() Method

Parameter	Description
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getDocumentData() Method

In ScheduleService, use the getDocumentData() method to return a document generated by a scheduled job. The getDocumentData() method takes the jobOutputID as a parameter to return the appropriate document.

To get the jobOutputID: The scheduleReport() method returns jobID. Call getAllScheduledReportHistory() using this parent jobID to get a list of its children (schedule jobs). Use the appropriate child jobID to call getScheduledOutputInfo() to get a list of the outputIDs for this particular job. Use the outputID of the desired job as input (JobOutputID parameter) to the getDocumentData() method to retrieve the document data.

Note that the getDocumentData() method returns the byte[] of a report document, while the downloadDocumentData() Method saves the report document as a local file. The latter method returns the file ID, enabling the user to download the report document later through the Delivery Service. This is for performance concerns in cases where a report document size is large.

Signature

byte[] getDocumentData(String jobOutputID, String userID, String password);

Parameters for getDocumentData() Method

Parameter	Description
String jobOutputID	Job output assigned to the output. The output ID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getDocumentDataInSession() Method

In ScheduleService, use the getDocumentDataInSession() method to return the byte[] of a report document based on the JobOutputID and bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

byte[] getDocumentDataInSession(String jobOutputID, String bipSessionToken);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for getDocumentDataInSession() Method

Parameter	Description
String jobOutputID	The ID assigned to the output. The jobOutputID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getScheduledJobInfo() Method

In ScheduleService, use the getScheduledJobInfo() method to return a JobDetail object that provides the details about a submitted job, including report parameters and other properties. This method retrieves all information stored in the database for a given jobInstanceID, userID, and password.

Signature

JobDetail getScheduledJobInfo(int jobInstanceID, String userID, String password);

Parameters for getScheduledJobInfo() Method

Parameter	Description
int jobInstanceID	The ID of the job for which to return job information.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getScheduledJobInfoInSession() Method

In ScheduleService, use the getScheduledJobInfoInSession() method to return a JobDetail object that provides the details for the job that's associated with a given jobInstanceID and the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

JobDetail getScheduledJobInfoInSession(String jobInstanceID, String bipSessionToken);

Parameters for getScheduledJobInfoInSession() Method

Parameter	Description
String jobInstanceID	The ID assigned to the job instance. The jobInstanceID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

getScheduledReportDeliveryInfo() Method

In ScheduleService, use the getScheduledReportdeliveryInfo() method to retrieve information about the delivery of a scheduled job output. For each scheduled Job, it could have multiple outputIDs.

For each outputID, there could be multiple delivery info. See JobOutputDeliverysList.

Signature

JobOutputDeliverysList getScheduledReportDeliveryInfo(String jobOutputID, String userID, String password);

Parameters for getScheduledReportDeliveryInfo() Method

Parameter	Description
String jobOutputID	The ID assigned to the output of the job for which you want information. The jobOutputID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getScheduledReportDeliveryInfoInSession() Method

In ScheduleService, use the getScheduledReportDeliveryInfoInSession() method to return a JobOutputDeliverysList object that provides the details for a given jobOutputID and the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

JobOutputDeliverysList getScheduledReportDeliveryInfoInSession(String jobOutputID, String bipSessionToken);

Parameters for getScheduledReportDeliveryInfoInSession() Method

Parameter	Description
String jobOutputID	The ID assigned to the job output. The jobOutputID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

getScheduledReportOutputInfo() Method

In ScheduleService, use the getScheduledReportOutputInfo() method to return information about a specific scheduled report output.

Signature

JobOutputsList getScheduledReportOutputInfo(String jobInstanceID, String userID, String password);

Parameters for getScheduledReportOutputInfo() Method

Parameter	Description
String jobInstanceID	The ID of the job for which to return job information.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getScheduledReportOutputInfoInSession() Method

In ScheduleService, use the getScheduledReportOutputInfo() method to return information about a specific scheduled report output based on its jobInstanceID and the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

JobOutputsList getScheduledReportOutputInfoInSession(String jobInstanceID, String bipTokenSession);

Parameters for getScheduledReportOutputInfoInSession() Method

Parameter	Description
String jobInstanceID	The ID of the job for which to return job information.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getScheduledReportStatus() Method

In ScheduleService, use the getScheduledReportStatus() method to get the status of the scheduled job.

Signature

public JobStatus getScheduledReportStatus(String scheduledJobID, String userID, String password);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for getScheduledReportStatus() Method

Parameter	Description
String scheduledJobID	Specifies the job ID of the submitted job.
String userID	Specifies the user ID of the user.
String password	Specifies the password of the user.

getScheduledReportStatusInSession() Method

In ScheduleService, use the getScheduledReportStatusInSession() method to get the status of the scheduled job.

Signature

JobStatus getScheduledReportStatusInSession(java.lang.String scheduledJobID, java.lang.String bipSessionToken);

Parameters for getScheduledReportStatusInSession() Method

Parameter	Description
String scheduledJobID	The job ID of the submitted job.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getXMLData() Method

In ScheduleService, use the getXMLData() method to return, in XML format, the data document generated by a scheduled job. You can use the JobOutputID returned from the scheduleReport() method to retrieve the generated XML document.

Note, the getXMLData() method returns the byte[] of a report document, while the downloadXMLData() Method saves the XML data as a local file. The latter method returns the file ID, enabling the user to download the XML-based document later through the Delivery Service. This is for performance concerns in cases where a report document size is quite large.

Signature

byte[] getXMLData(String JobInstanceID, String userID, String password);

Parameters for getXMLData() Method

Parameter	Description
String JobInstanceID	The ID assigned to the instance of the job that generated the output. The JobInstanceID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

getXMLDataInSession() Method

In ScheduleService, use the getXMLDataInSession() method to return the byte[] of XML data based on the jobInstanceID and bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

byte[] getXMLDataInSession(String jobInstanceID, String bipSessionToken);

Parameters for getXMLDataInSession() Method

Parameter	Description
String jobInstanceID	The ID assigned to the job instance. The jobInstanceID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

purgeJobHistory() Method

In ScheduleService, use the purgeHistory() method to perform a "hard" delete of the historical information about a report job. That is, to permanently purge the information from the database.

You must precede the purgeJobHistory() method with the deleteJobHistory() method, otherwise the following SOAP fault is thrown:

```
purgeJobHistory failed due to job is not deleted. You have to delete JobHistory
first prior to purge.
```

Signature

boolean purgeJobHistory(String instanceJobID, String userID, String password);

Parameters for purgeJobHistory() Method

Parameter	Description
String instanceJobID	The ID assigned to the instance of the job that generated the output. The instanceJobID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

purgeJobHistoryInSession() Method

In ScheduleService, use the purgeJobHistoryInSession() method to permanently purge the job history from the database for the given instanceJobID and bipSessionToken of the given user. This action must be preceded by a deleteJobHistoryInSession.

For more information about in-session methods, see About In-Session Methods.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Signature

boolean purgeJobHistoryInSession(String instanceJobID, String bipSessionToken);

Parameters for purgeJobHistoryInSession() Method

Parameter	Description
String instanceJobID	The ID assigned to the job instance. The instanceJobID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

resendScheduledReport() Method

In ScheduleService, use the resendScheduledReport() method to resend a previously-defined scheduled report. The resend action is respective to the outputJobID. There's no need to define any delivery channels options, as the previously-defined delivery parameters are used to perform the resend action.

Signature

boolean resendScheduledReport(String outputJobID, String userID, String password);

Parameters for resendScheduledReport() Method

Parameter	Description
String outputJobID	The outputId generated when you run the job. To get the outputId of a specific job, run the getScheduledReportOutputInfo() method of ScheduleService.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

resendScheduledReportInSession() Method

In ScheduleService, use the resendScheduledReportInSession() method to resend a previously-defined scheduled report based on its associated outputJobID and the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean resendScheduledReportInSession(String outputJobID, String bipSessionToken);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for resendScheduledReportInSession() Method

Parameter	Description
String outputJobID	The outputId generated when you run the job. To get the outputId of a specific job, run the getScheduledReportOutputInfo() method of ScheduleService.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

resumeSchedule() Method

In ScheduleService, use the resumeSchedule() method to resume a schedule job that has been suspended.

Signature

boolean resumeSchedule(String jobInstanceID, String userID, String password);

Parameters for resumeSchedule() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job to be resumed. The jobInstanceID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

resumeScheduleInSession() Method

In In ScheduleService, use the resumeScheduleInSession() method resumes a scheduled job that was previously suspended based on its jobInstanceID and the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean resumeScheduleInSession(String outputJobID, String bipSessionToken);

Parameters for resumeScheduleInSession() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job to be resumed. The jobInstanceID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the server session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

scheduleReport() Method

In In ScheduleService, use the scheduleReport() method to schedule the reports that are in the catalog. You can submit and run the reports immediately or create a job to schedule the reports to run. When you schedule reports you can also deliver reports to any of the delivery destinations that are set up in your instance.

The method returns a JobID of the scheduled job.

Signature

String scheduleReport(ScheduleRequest scheduleRequest, String userID, String password);

Parameters for scheduleReport() Method

Parameter	Description
ScheduleRequest scheduleRequest	Specifies a ScheduleRequest object for the report that you want to run. See ScheduleRequest.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

scheduleReportInSession() Method

In ScheduleService, use the scheduleReportInSession() method to schedule a report based on the schedule request, delivery channel, and bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

String scheduleReportInSession(ScheduleRequest scheduleRequest, DeliveryChannel deliveryChannel, String bipSessionToken);

Parameters for scheduleReportInSession() Method

Parameter	Description
ScheduleRequest scheduleRequest	Specifies a ScheduleRequest object for the report that you want to run. See ScheduleRequest..
DeliveryChannel deliveryChannel	Specifies the delivery channels through which the report will be delivered. See DeliveryChannels.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

suspendSchedule() Method

In ScheduleService, use the suspendSchedule() method to suspend a schedule job.

Signature

boolean suspendschedule(String jobInstanceID, String userID, String password);

Parameters for suspendSchedule() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job to be suspended. The jobInstanceID is a string of integers.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

suspendScheduleInSession() Method

In ScheduleService, use the suspendScheduleInSession() method to suspend a scheduled report based on its associated jobInstanceID and the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean suspendScheduleInSession(String jobInstanceID, String bipSessionToken);

Parameters for suspendScheduleInSession() Method

Parameter	Description
String jobInstanceID	The ID assigned to the instance of the job to be suspended. The jobInstanceID is a string of integers.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

4

Publisher ReportService

Use ReportService methods to interact with Publisher Report objects. These methods enable you to design and define reports, report templates, run-time operations, and parameters.

Topics:

- createReport() Method
- createReportInSession() Method
- downloadReportDataChunk() Method
- downloadReportDataChunkInSession() Method
- getListOfSubjectArea() Method
- getListOfSubjectAreaInSession() Method
- getReportDefinition() Method
- getReportDefinitionInSession() Method
- getReportParameters() Method
- getReportParametersInSession() Method
- getReportSampleData() Method
- getReportSampleDataInSession() Method
- getTemplate() Method
- getTemplateInSession() Method
- getTemplateAvailableLocales() Method
- getTemplateAvailableLocalesInSession() Method
- getTemplateAvailableXLIFFs() Method
- getTemplateAvailableXLIFFsInSession() Method
- getTemplateParameters() Method
- getTemplateParameterInSession() Method
- getXDOSchema() Method
- getXDOSchemaInSession() Method
- removeObjectFromCache() Method
- removeObjectFromCacheInSession() Method
- removeTemplateForReport() Method
- removeTemplateForReportInSession() Method
- runDataModel() Method
- runDataModelInSession() Method
- runReport() Method
- runReportInSession() Method

Copyright © 2025, 2026, Oracle and/or its affiliates.

- updateReportDefinition() Method
- updateReportDefinitionInSession() Method
- updateTemplateForReport() Method
- updateTemplateForReportInSession() Method
- updateXLIFFForReport() Method
- updateXLIFFForReportInSession() Method
- uploadReportDataChunk() Method
- uploadReportDataChunkInSession() Method
- uploadTemplateForReport() Method
- uploadTemplateForReportInSession() Method
- uploadXLIFFForReport() Method
- uploadXLIFFForReportInSession() Method

createReportInSession() Method

In ReportService, use the createReportInSession() method to create a report in the catalog based on the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

String createReport(String reportName, String folderAbsolutePathURL, String dataModelURL, String templateFileName, byte[] templateData, String XLIFFFileName, byte[] XLIFFData, boolean updateFlag, String bipSessionToken);

Parameters for createReportInSession() Method

Parameter	Description
String reportName	The report name to create with the suffix ".xdo". For example, "myreport.xdo".
String folderAbsolutePathURL	The path to the folder in which to place the created report. For example: xmlp/Reports/financials.
String dataModelURL	The path to the data model that will be used as the data source for this report. For example: xmlp/Reports/financials/Data Models/my data model.xdm.
String templateFileName	The file name of the template to add the report definition.
byte[] templateData,	The template file.
String XLIFFFileName	The file name of the XLIFF file. You must append the locale to the XLIFF file name as follows: `template_<language code>_<country code>.xlf` where `<language_code>` is the two-letter ISO 639 language code `<country_code>` is the two-letter ISO 639 language code For example: template_en_us.xlf.
byte[] XLIFFData	The XLIFF file.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
boolean updateFlag	If `true`, overwrites existing report. If `false`, throws error if the report exists.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform operations.

downloadReportDataChunk() Method

In ReportService, use the downloadReportDataChunk() method to download very large documents, so that the caller calls this method multiple times until the entire data is downloaded.

Each call to this method downloads one chunk of the document, where the beginIdx parameter refers to the file download starting point. The client code can check if the value of ReportDataOffset is -1 and exit the loop. If it is -1, it means it finished reading all bytes from the file.

See ReportDataChunk.

> **Note**
>
> When using uploadReportDataChunk() or downloadReportDataChunk() in a clustered environment, you must set the System Temporary Directory to be a shared directory accessible to all servers within the cluster. To set the System Temporary Directory:
>
> 1. Sign in to Publisher with Administrator privileges.
> 2. Click the **Administration** link.
> 3. Under **System Maintenance**, click **Server Configuration**.
> 4. Under **General Properties** in the **System Temporary Directory** property, enter the absolute path to a directory accessible to all servers in the cluster.
>
>   For example, the directory can exist under `${xdo.server.config.dir}/temp` but you must enter the absolute path, such as `/net/subfoldera/scratch/subfolderb/bicat/temp`
>
> Repeat this procedure for all servers in the cluster, entering the same value for **System Temporary Directory**.

Signature

ReportDataChunk downloadReportDataChunk(String fileID, int beginIdx, int size);

Parameters for downloadReportDataChunk() Method

Parameter	Description
String fileID	fileID is returned inside ReportResponse, which is returned when calling runReport() Method.
int beginIdx	The starting point of the index (default is `0`).

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
int size	The size of the file to download (in bytes).

downloadReportDataChunkInSession() Method

In ReportService, use the downloadReportDataChunk() method to download very large documents using the bipSessionToken of a given user.

The caller calls this method multiple times until all document content is downloaded. Each call to this method downloads one chunk of the document, where the beginIdx parameter refers to the file download starting point.

For more information about in-session methods, see About In-Session Methods.

Signature

ReportDataChunk downloadReportDataChunkInSession(String fileID, int beginIdx, int size, String bipSessionToken);

Parameters for downloadReportDataChunkInSession() Method

Parameter	Description
String fileID	fileID is returned inside ReportRequest, which is returned when calling runReport() Method.
int beginIdx	The starting point of the index (default is 1).
int size	The size of the file to download (in kilobytes).
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getReportDefinition() Method

In ReportService, use the getReportDefinition() method to get information about a report, such as the default template, output type, and a list of template IDs

With the list of template IDs, you can generate a report with a template other than the default. See ReportDefinition.

Signature

ReportDefinition getReportDefinition(String reportAbsolutePath, String userID, String password);

Parameters for getReportDefinition() Method

Parameter	Description
String reportAbsolutePath	The path to the report for which to retrieve the report definition. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String userID	Specifies the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
String password	Specifies the password for the user name.

createReport() Method

In ReportService, use the createReport() method to create a report in the catalog.

The method enables you to set the path to the data model and supply template files and translation (XLIFF) files to the report definition.

Signature

String createReport(String reportName, String folderAbsolutePathURL, String dataModelURL, String templateFileName, byte[] templateData, String XLIFFFileName, byte[] XLIFFData, boolean updateFlag, String userID, String password);

Parameters for createReport() Method

Parameter	Description
String reportName	The report name to create with the suffix ".xdo". For example, "myreport.xdo".
String folderAbsolutePathURL	The path to the folder in which to place the created report. For example: `xmlp/Reports/financials`
String dataModelURL	The path to the data model that will be used as the data source for this report. For example: `xmlp/Reports/financials/Data Models/my data model.xdm`
String templateFileName	The file name of the template to add the report definition.
byte[] templateData	The template file.
String XLIFFFileName	The file name of the XLIFF file. You must append the locale to the XLIFF file name as follows: `template_<language code>_<country code>.xlf` where `<language_code>` is the two-letter ISO 639 language code `<country_code>` is the two-letter ISO 639 language code For example: template_en_us.xlf
byte[] XLIFFData	The XLIFF file.
boolean updateFlag	If `true`, overwrites existing report. If `false`, throws error if the report exists.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getListOfSubjectArea() Method

In ReportService, use the getListOfSubjectArea() method to get the list of Subject Areas in semantic layer.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Signature

public ArrayOfString getListOfSubjectArea(String userID, String password, String connectionName);

Parameters for getListOfSubjectArea() Method

Parameter	Description
String userID	Specifies the user ID of the user.
String password	Specifies the password of the user.
String connectionName	Specifies a valid connection name.

getListOfSubjectAreaInSession() Method

In ReportService, use the getListOfSubjectAreaInSession() method to get the list of Subject Areas in semantic layer.

Signature

public ArrayOfString getListOfSubjectAreaInSession(String bipSessionToken, String connectionName);

Parameters for getListOfSubjectAreaInSession() Method

Parameter	Description
String bipSessionToken	The session token retrieved from login.
String connectionName	A valid connection name.

getReportDefinitionInSession() Method

In ReportService, use the getReportDefinitionInSession() method to obtain information about a report using the bipSessionToken of a given user.

This method returns report details such as the default template, output type, and a list of template IDs. With the list of template IDs, you can generate a report with a template other than the default.

See ReportDefinition.

For more information about in-session methods, see About In-Session Methods.

Signature

ReportDefinition getReportDefinitionInSession(String reportAbsolutePath, String bipSessionToken);

Parameters for getReportDefinitionInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report for which to retrieve the report definition. For example: /HR Manager/Employee Reports/Employee Listing.xdo

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getReportParameters() Method

In ReportService, use the getReportParameters() method to get an array of report parameters and their default values.

With the list of parameters, you can set parameter values before running or scheduling a report.

See ParamNameValues.

Signature

ParamNameValues getReportParameters(ReportRequest reportRequest, String userID, String password);

Parameters for getReportParameters() Method

Parameter	Description
ReportRequest reportRequest	See ReportRequest.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getReportParametersInSession() Method

In ReportService, use the getReportParameters() method to get an array of report parameters and their default values based on the bipSessionToken of a given user.

With the list of parameters, you can set parameter values before running or scheduling a report.

See ParamNameValues.

For more information about in-session methods, see About In-Session Methods.

Signature

ParamNameValues getReportParametersInSession(ReportRequest reportRequest, String bipSessionToken);

Parameters for getReportParametersInSession() Method

Parameter	Description
ReportRequest reportRequest	See ReportRequest.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getReportSampleData() Method

In ReportService, use the getReportSampleData() method to retrieve the sample data file stored with the report data model.

Signature

byte[] getReportSampleData(String reportAbsolutePath, String userID, String password);

Parameters for getReportSampleData() Method

Parameter	Description
String reportAbsolutePath	The path to the report for which to retrieve the report data model sample data. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getReportSampleDataInSession() Method

In ReportService, use the getReportSampleDataInSession() method to retrieve the sample data file stored with the report data model based on the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

byte[] getReportSampleData(String reportAbsolutePath, String bipSessionToken);

Parameters for getReportSampleDataInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report for which to retrieve the report data model sample data. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

getTemplate() Method

In ReportService, use the getTemplate() method to retrieve a template from a report definition in the catalog.

Signature

byte[] getTemplate(String reportAbsolutePath, String templateID, String locale, String userID, String password);

Parameters for getTemplate() Method

Parameter	Description
String reportAbsolutePath	The path to the report to which the template is associated. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateID	The ID of the template (for example, `Chart Layout`).
String locale	The locale of the template to retrieve (for example, `en_US`).
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getTemplateInSession() Method

In ReportService, use the getTemplateInSession() method to retrieve a template from a report definition in the catalog based on the bipTokenSession of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

byte[] getTemplateInSession(String reportAbsolutePath, String templateID, String locale, String bipSessionToken);

Parameters for getTemplateInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report to which the template is associated. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateID	The ID of the template (for example, `Chart Layout`).
String locale	The locale of the template to retrieve (for example, `en_US`).
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getTemplateAvailableLocales() Method

In ReportService, use the getTemplateAvailableLocales() method to get the locales associated with the template.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Signature

public ArrayOfXsdString getTemplateAvailableLocales(String reportAbsolutePath, String templateID, String userID, String password);

Parameters for getTemplateAvailableLocales() Method

Parameter	Description
String reportAbsolutePath	The absolute path to the report in catalog.
String templateID	The ID of the template used by the report.
String userID	The user ID of the user.
password	Password of the user.

getTemplateAvailableLocalesInSession() Method

In ReportService, use the getTemplateAvailableLocalesInSession() method to get the locales associated with the template.

Signature

public ArrayOfXsdString getTemplateAvailableLocalesInSession(String reportAbsolutePath, String templateID, String bipSessionToken);

Parameters for getTemplateAvailableLocalesInSession() Method

Parameter	Description
String reportAbsolutePath	The absolute path to the report in catalog.
String templateID	The ID of the template used by the report.
String bipSessionToken	The session token retrieved from login.

getTemplateAvailableXLIFFs() Method

In ReportService, use the getTemplateAvailableXLIFFs() method to get the XLIFFs (XML Localization Interchange File Format) associated with the template.

Signature

public ArrayOfXsdString getTemplateAvailableXLIFFs(String reportAbsolutePath, String templateID, String userID, String password);

Parameters for getTemplateAvailableXLIFFs() Method

Parameter	Description
String reportAbsolutePath	The absolute path to the report in catalog.
String templateID	The ID of the template used by the report.
String userID	The user ID of the user.
String password	The password of the user.

Copyright © 2025, 2026, Oracle and/or its affiliates.

getTemplateAvailableXLIFFsInSession() Method

In ReportService, use the getTemplateAvailableXLIFFsInSession() method to get the XLIFFs (XML Localization Interchange File Format) associated with the template.

Signature

public ArrayOfXsdString getTemplateAvailableXLIFFsInSession(String reportAbsolutePath, String templateID, String bipSessionToken);

Parameters for getTemplateAvailableXLIFFsInSession() Method

Parameter	Description
String reportAbsolutePath	The absolute path to the report in catalog.
String templateID	The ID of the template used by the report.
String bipSessionToken	The session token retrieved from login.

getTemplateParameters() Method

In ReportService, use the getTemplateParameters() method to get the parameters for a template.

See ParamNameValue.

Signature

ParamNameValue[] getTemplateParameters(String reportAbsolutePath, String templateID, String userID, String password);

Parameters for getTemplateParameters() Method

Parameter	Description
String reportAbsolutePath	The path to the report for which to retrieve the report definition. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateID	The ID assigned to the template, for example: "Chart Layout".
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getTemplateParameterInSession() Method

In ReportService, use the getReportParametersInSession() method to get the parameters for a template.

See ParamNameValue.

For more information about in-session methods, see About In-Session Methods.

Signature

ParamNameValue[] getTemplateParameters(String reportAbsolutePath, String templateID, String bipSessionToken);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for getTemplateParameterInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report for which to retrieve the report definition. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateID	The ID assigned to the template, for example: "Chart Layout".
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getXDOSchema() Method

In ReportService, use the getXDOSchema() method to retrieve the XDO schema for a report definition in the catalog.

Signature

byte[] getXDOSchema(String reportAbsolutePath, String locale, String userID, String password);

Parameters for getXDOSchema() Method

Parameter	Description
String reportAbsolutePath	The path to the report from which to retrieve the XDO schema. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String locale	The locale of the template to retrieve (for example, en_US).
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getXDOSchemaInSession() Method

In ReportService, use the getXDOSchemaInSession() method to retrieve the XDO schema for a report definition in the catalog based on a bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

byte[] getXDOSchemaInSession(String reportAbsolutePath, String locale, String bipSessionToken);

Parameters for getXDOSchemaInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report from which to retrieve the XDO schema. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String locale	The locale of the template to retrieve (for example, en_US).

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

removeObjectFromCache() Method

In ReportService, use the removeObjectFromCache() method to remove the object from the cache.

Signature

public boolean removeObjectFromCache (String objectPath, String userID, String password);

Parameters for removeObjectFromCache() Method

Parameter	Description
String objectPath	The path to the object in catalog.
String userID	The user ID of the user.
String password	The password of the user.

removeObjectFromCacheInSession() Method

In ReportService, use the removeObjectFromCacheInSession() method to remove the object from the cache.

Signature

public boolean removeObjectFromCacheInSession(String objectPath, String bipSessionToken);

Parameters for removeObjectFromCacheInSession() Method

Parameter	Description
String objectPath	The path to the object in catalog.
String bipSessionToken	The session token retrieved from login.

removeTemplateForReportInSession() Method

In ReportService, use the removeTemplateForReportInSession() method to remove a template from a report definition in the catalog based on the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean removeTemplateForReportInSession(String reportAbsolutePath, String templateFileName, String bipSessionToken);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for removeTemplateForReportInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report from which to remove the template. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateFileName	The file name of the template to remove.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

runDataModel() Method

In ReportService, use the runDataModel() method to run the data model and get the XML data.

Signature

public ReportResponse runDataModel(ReportRequest reportRequest, String userID, String password);

Parameters for runDataModel() Method

Parameter	Description
ReportRequest reportRequest	The path to the report and other metadata of the report.
String userID	The user ID of the user.
String password	The password of the user.

runDataModelInSession() Method

In ReportService, use the runDataModelInSession() method to run the data model and get the XML data.

Signature

public ReportResponse runDataModelInSession(ReportRequest reportRequest, String bipSessionToken);

Parameters for runDataModelInSession() Method

Parameter	Description
ReportReques reportRequest	The path to the report and other metadata of the report.
String bipSessionToken	The session token retrieved from login.

runReport() Method

In ReportService, use the runReport() method to send a request to the Publisher server to run a specific report.

See ReportRequest and ReportResponse.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Signature

ReportResponse runReport(ReportRequest reportRequest, String userID, String password);

Parameters for runReport() Method

Parameter	Description
ReportRequest reportRequest	See ReportRequest.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

removeTemplateForReport() Method

In ReportService, use the removeTemplateForReport() method to remove a template from a report definition in the catalog.

Signature

boolean removeTemplateForReport(String reportAbsolutePath, String templateFileName, String userID, String password);

Parameters for removeTemplateForReport() Method

Parameter	Description
String reportAbsolutePath	The path to the report from which to remove the template. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateFileName	The file name of the template to remove.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

runReportInSession() Method

In ReportService, use the runReportInSession() method to send a request to the server to run a specific report based on the bipSessionToken of a given user.

See ReportRequest and ReportResponse.

For more information about in-session methods, see About In-Session Methods.

Signature

ReportResponse runReportInSession(ReportRequest reportRequest, String bipSessionToken);

Parameters for runReportInSession() Method

Parameter	Description
ReportRequest reportRequest	See ReportRequest.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

updateReportDefinition() Method

In ReportService, use the updateReportDefinition() to update attributes of the report definition file (.xdo) and write the file back to the catalog.

Signature

boolean updateReportDefinition(String reportAbsPath, ReportDefinition newReportDefn, String userID, String password);

Parameters for updateReportDefinition() Method

Parameter	Description
String reportAbsPath	The path to the report for which to update the report definition. For example: /HR Manager/Employee Reports/Employee Listing.xdo
ReportDefinition newReportDefn	See ReportDefinition.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

updateReportDefinitionInSession() Method

In ReportService, use the updateReportDefinitionInSession() to update attributes of the report definition file (.xdo) based on the bipTokenSession of a given user, and then to write the file back to the catalog.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean updateReportDefinitionInSession(String reportAbsPath, ReportDefinition newReportDefn, String bipSessionToken);

Parameters for updateReportDefinitionInSession() Method

Parameter	Description
String reportAbsPath	The path to the report for which to update the report definition. For example: /HR Manager/Employee Reports/Employee Listing.xdo
ReportDefinition newReportDefn	See ReportDefinition.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

updateTemplateForReport() Method

In ReportService, use the updateTemplateForReport() method to update a template for a specific report in the catalog.

Signature

boolean updateTemplateForReport(String reportAbsolutePath, String templateName, String locale, byte[] templateData, String userID, String password);

Parameters for updateTemplateForReport() Method

Parameter	Description
String reportAbsolutePath	The path to the report that contains the template to update. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateName	The name of the template to update (for example, `Chart Layout`).
String locale	The locale of the template to update (for example, `en_US`).
byte[] templateData	The template file.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

updateTemplateForReportInSession() Method

In ReportService, use the updateTemplateForReportInSession() method to update a template for a specific report in the catalog based on the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean updateTemplateForReportInSession(String reportAbsolutePath, String templateName, String locale, byte[] templateData, String bipSessionToken);

Parameters for updateTemplateForReportInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report that contains the template to update. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateName	The name of the template to update (for example, `Chart Layout`).
String locale	The locale of the template to update (for example, `en_US`).
byte[] templateData	The template file.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

updateXLIFFForReport() Method

In ReportService, use the updateXLIFFForReport() method to update a translation file (XLIFF) associated with a layout definition in the catalog.

Signature

boolean updateXLIFFForReport(String reportAbsolutePath, byte[] xliffData, String layoutFileName, String locale, String userID, String password);

Parameters for updateXLIFFForReport() Method

Parameter	Description
String reportAbsolutePath	The path to the report to that contains the XLIFF file to update. For example: /HR Manager/Employee Reports/Employee Listing.xdo
byte[] xliffData	The XLIFF fie to upload.
String layoutFileName	The file name of the layout for which the XLIFF file is to be updated. For example: employee_listing.rtf.
String locale	The locale to assign to the XLIFF (for example, en_US).
String userID	Specifies the user name.
String password	Specifies the password for the user name.

updateXLIFFForReportInSession() Method

In ReportService, use the updateXLIFFForReportInSession() method to update a translation file (XLIFF) associated with a layout definition in the catalog based on the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean updateXLIFFForReportInSession(String reportAbsolutePath, byte[] xliffData, String layoutFileName, String locale, String bipSessionToken);

Parameters for updateXLIFFForReportInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report to that contains the XLIFF file to update. For example: /HR Manager/Employee Reports/Employee Listing.xdo
byte[] xliffData	The XLIFF fie to upload.
String layoutFileName	The file name of the layout for which the XLIFF file is to be updated. For example: employee_listing.rtf.
String locale	The locale to assign to the XLIFF (for example, en_US).
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

uploadReportDataChunk() Method

In ReportService, use the uploadReportDataChunk() method to upload a report data chunk.

When using uploadReportDataChunk() or downloadReportDataChunk() in a clustered environment, you must set the System Temporary Directory to be a shared directory accessible to all servers within the cluster. To set the System Temporary Directory, repeat this procedure for all servers in the cluster, entering the same value for **System Temporary Directory**:

1. Sign in with Administrator privileges.
2. Click the **Administration** link.
3. Under **System Maintenance**, click **Server Configuration**.
4. Under **General Properties** in the **System Temporary Directory** property, enter the absolute path to a directory accessible to all servers in the cluster.

   For example, the directory can exist under `${xdo.server.config.dir}/temp` but you must enter the absolute path, such as `/net/subfoldera/scratch/subfolderb/bicat/temp`

Signature

uploadReportDataChunk(String fileID, byte[] reportDataChunk, String reportRawDataChunk, String userID, String password);

Parameters for uploadReportDataChunk() Method

Parameter	Description
String fileID	In the first call, you do not need to provide the fileID, after the successful uploading of the first chunk of XML data, it will return a fileID, for example: filename. On your subsequent calls, you can supply the same fileID to append the subsequent data chunks to the same file.
byte[] reportDataChunk	The XML data to upload.
String reportRawDataChunk	String representation of XML data, presenting as reportRawDataChunk. This is an alternative to reportDataChunk byte[].
String userID	Specifies the user name.
String password	Specifies the password for the user name.

uploadReportDataChunkInSession() Method

In ReportService, use the uploadReportDataChunkInSession() method to upload a report data chunk based on the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

uploadReportDataChunkInSession(String fileID, byte[] reportDataChunk, String reportRawDataChunk, String bipSessionToken);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for uploadReportDataChunkInSession() Method

Parameter	Description
byte[] reportDataChunk	The XML data to upload.
String reportRawDataChunk	String representation of XML data, presenting as reportRawDataChunk. This is an alternative to reportDataChunk byte[].
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

uploadTemplateForReport() Method

In ReportService, use the uploadTemplateForReport() method to upload a template to a report definition in the catalog.

Signature

boolean uploadTemplateForReport(String reportAbsolutePath, String templateName, String templateType, String locale, byte[] templateData, String userID, String password);

Parameters for uploadTemplateForReport() Method

Parameter	Description
String reportAbsolutePath	The path to the report to which to upload the template. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateName	The file name of the template to upload.
String templateType	The template type. Valid values are: • `csv` (CSV) • `eText` (eText template) • `excel` (Microsoft Excel) • `excel2000` (Microsoft Excel 2000) • `flash` (Adobe Flash) • `html` (HTML) • `mhtml` (MIME HTML) • `pdf` (Adobe PDF) • `pdfz` (eBook) • `ppt` (Microsoft PowerPoint) • `rtf` (Rich Text Format) • `text` (Text) • `txml` (Transformed XML) • `xml` (XML) • `xpa` (Analyzer template) • `xpt` (BI Publisher template) • `xslfo` (XSL-FO style sheet)
String locale	The locale to assign to the template (for example, `en_US`).
byte[] templateData	The contents of the template file to upload.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

uploadTemplateForReportInSession() Method

In ReportService, use the uploadTemplateForReportInSession() method to upload a template to a report definition in the catalog based on the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean uploadTemplateForReportInSession(String reportAbsolutePath, String templateName, String templateType, String locale, byte[] templateData, String bipSessionToken);

Parameters for uploadTemplateForReportInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report to which to upload the template. For example: /HR Manager/Employee Reports/Employee Listing.xdo
String templateFileName	The file name of the template to upload.
String templateName	The name of the template to upload.
String locale	The locale to assign to the template (for example, en_US).
byte[] templateData	The contents of the template file to upload.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

uploadXLIFFForReport() Method

In ReportService, use the uploadXLIFFForReport() method to upload a translation file (XLIFF) to a layout definition in the catalog.

Signature

boolean uploadXLIFFForReport(String reportAbsolutePath, byte[] xliffData, String layoutFileName, String locale, String userID, String password);

Parameters for uploadXLIFFForReport() Method

Parameter	Description
String reportAbsolutePath	The path to the report to which to upload the XLIFF. For example: /HR Manager/Employee Reports/Employee Listing.xdo
byte[] xliffData	The XLIFF fie to upload.
String layoutFileName	The file name of the layout to which to associate the XLIFF file. For example: employee_listing.rtf.
String locale	The locale to assign to the XLIFF (for example, en_US).
String userID	Specifies the user name.
String password	Specifies the password for the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

uploadXLIFFForReportInSession() Method

In ReportService, use the uploadXLIFFForReport() method to upload a translation file (XLIFF) to a layout definition in the catalog based on the bipSessionToken of a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean uploadXLIFFForReportInSession(String reportAbsolutePath, byte[] xliffData, String layoutFileName, String locale, String bipSessionToken);

Parameters for uploadXLIFFForReportInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the report to which to upload the XLIFF. For example: /HR Manager/Employee Reports/Employee Listing.xdo
byte[] xliffData	The XLIFF fie to upload.
String layoutFileName	The file name of the layout to which to associate the XLIFF file. For example: employee_listing.rtf.
String locale	The locale to assign to the XLIFF (for example, en_US).
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

5

Publisher SecurityService

Use the SecurityService methods to manage Publisher security operations, such as authentication, impersonation, login, logout, and account management.

Topics:

- assignRolesToUser() Method
- checkObjectPermissions() Method
- checkObjectPermissionsInSession() Method
- createRole() Method
- createUser() Method
- deleteRole() Method
- deleteUser() Method
- getBIPHTTPSessionInterval() Method
- getObjectSecurityXML() Method
- getSecurityModel() Method
- impersonate() Method
- isAdmin() Method
- isDataModelDeveloper() Method
- isExcelAnalyzer() Method
- isOnlineAnalyzer() Method
- isReportDeveloper() Method
- isScheduler() Method
- isUserExists() Method
- login() Method
- logout() Method
- notifyBIEEPreferencesUpdated() Method
- notifyBIEEPreferencesUpdatedWithString() Method
- removeRolesFromUser() Method
- updateRole() Method
- updateUser() Method
- validateLogin() Method

Copyright © 2025, 2026, Oracle and/or its affiliates.

> **Note**
>
> SecurityService is available to the Publisher Security Model only. If your Publisher deployment uses another security model (for example, LDAP, Oracle E-Business Suite, or Oracle Fusion Cloud Applications Suite), you can't use the SecurityService API.

assignRolesToUser() Method

In SecurityService, use the assignRolesToUser() method to assign new roles to a user in Publisher.

Signature

String[] assignRolesToUser(String userName, String[] roleNames, String adminUser, String adminPassword);

Parameters for assignRolesToUser() Method

Parameter	Description
String userName	The user to which to add the role or roles.
String[] roleNames	The name of the role to add to the user. For example, "Financial Users".
String adminUser	The user name for a user with administration privileges.
String adminPassword	The password for the administration user name.

checkObjectPermissions() Method

In SecurityService, use the checkObjectPermissions() method to validate the user's permission to the specified object.

Signature

public Boolean[] checkObjectPermissions(String objectAbsolutePath, ArrayOfString privileges, String userID, String password);

Parameters for checkObjectPermissions() Method

Parameter	Description
String objectAbsolutePath	The absolute path to the object.
ArrayOfString privileges	A list of privilege names.
String userID	The user ID of the user.
String password	The password of the user.

Copyright © 2025, 2026, Oracle and/or its affiliates.

checkObjectPermissionsInSession() Method

In SecurityService, use the checkObjectPermissionsInSession() method to validate the user's permission to the specified object.

Signature

public Boolean[] checkObjectPermissionsInSession(String objectAbsolutePath, ArrayOfString privileges, String bipSessionToken);

Parameters for checkObjectPermissionsInSession() Method

Parameter	Description
String objectAbsolutePath	The absolute path to the object.
ArrayOfString privileges	A list of privilege names.
String bipSessionToken	The session token retrieved from login.

createRole() Method

In SecurityService, use the createRole() method to create a role in Publisher.

Signature

boolean createRole(String roleName, String description, String adminUser, String adminPassword);

Parameters for createRole() Method

Parameter	Description
String roleName	The name of the role to create. For example, Financial Users.
String description	The description of the role.
String adminUser	The user name for a user with administration privileges.
String adminPassword	The password for the administration user name.

createUser() Method

In SecurityService, use the createUser() method to create a user in Publisher.

Signature

boolean createUser(String userName, String password, String adminUser, String adminPassword);

Parameters for createUser() Method

Parameter	Description
String userName	The user name to create.
String password	The password for the newly created user.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
String adminUser	The user name for a user with administration privileges.
String adminPassword	The password for the administration user name.

deleteRole() Method

In SecurityService, use the deleteRole() method to delete a role from Publisher.

Signature

boolean deleteRole(String roleName, String adminUser, String adminPassword);

Parameters for deleteRole() Method

Parameter	Description
String roleName	The user name to delete.
String adminUser	The user name for a user with administration privileges.
String adminPassword	The password for the administration user name.

deleteUser() Method

In SecurityService, use the deleteUser() method to delete a user from Publisher.

Signature

boolean deleteUser(String userName, String adminUser, String adminPassword);

Parameters for deleteUser() Method

Parameter	Description
String userName	The user name to delete.
String adminUser	The user name for a user with administration privileges.
String adminPassword	The password for the administration user name.

getBIPHTTPSessionInterval() Method

In SecurityService, use the getBIPHTTPSessionInterval() method returns the number of seconds of an HTTP session interval.

Signature

int getBIPHTTPSessionInterval(void);

Copyright © 2025, 2026, Oracle and/or its affiliates.

getObjectSecurityXML() Method

In SecurityService, use the getObjectSecurityXML() method to extract the permissions for a catalog object.

Signature

byte[] getObjectSecurityXML(String adminUsername, String adminPassword, String objectAbsolutePath, boolean isRecursive);

Parameters for getObjectSecurityXML() Method

Parameter	Description
String adminUsername	The user name for a user with administrator privileges.
String adminPassword	The password associated with the adminUserName.
String objectAbsolutePath	The absolute path to the catalog object for which to retrieve the permissions description.
boolean isRecursive	Whether or not objectAbsolutePath is recursive.

getSecurityModel() Method

In SecurityService, use the getSecurityModel() method to return Publisher security model in place.

Signature

String getSecurityModel(void);

impersonate() Method

In SecurityService, use the impersonate() method to enable an admin account to act on the behalf of a user account.

This method is very useful if the user doesn't have a known password to be authenticated. This method logs in using the admin account privilege, then switches the owner of the session to the passed-in username. Therefore, bipSession token later is verified by the passed-in username. All further operations are performed through given username.

Signature

String impersonate(String adminUsername, String adminPassword,String username);

Parameters for impersonate() Method

Parameter	Description
String adminUserName	The user name for a user with administration privileges
String adminPassword	The password for the administration user name.
String username	The username of the user account that's granted administrator privileges.

Copyright © 2025, 2026, Oracle and/or its affiliates.

isAdmin() Method

In SecurityService, use the isAdmin() method to check if the user has administrator privileges.

Signature

public boolean isAdmin(String userID, String password);

Parameters for isAdmin() Method

Parameter	Description
String userID	The user ID of the user.
String password	The password of the user.

isDataModelDeveloper() Method

In SecurityService, use the isDataModelDeveloper() method to check if the user has the DEVELOP_MODEL privilege.

Signature

public boolean isDataModelDeveloper(String userID, String password);

Parameters for isDataModelDeveloper() Method

Parameter	Description
String userID	The user ID of the user.
String password	The password of the user.

isExcelAnalyzer() Method

In SecurityService, use the isExcelAnalyzer() method to check if the user has the EXCEL_ANALYZER privilege.

Signature

public boolean isExcelAnalyzer(String userID, String password);

Parameters for isExcelAnalyzer() Method

Parameter	Description
String userID	The user ID of the user.
String password	The password of the user.

isOnlineAnalyzer() Method

In SecurityService, use the isOnlineAnalyzer() method to check if the user has the ONLINE_ANALYZER privilege.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Signature

public boolean isOnlineAnalyzer(String userID, String password);

Parameters for isOnlineAnalyzer() Method

Parameter	Description
String userID	The user ID of the user.
String password	The password of the user.

isReportDeveloper() Method

In SecurityService, use the isReportDeveloper() method to check if the user has the DEVELOP_REPORT privilege.

Signature

public boolean isReportDeveloper(String userID, String password);

Parameters for isReportDeveloper() Method

Parameter	Description
String userID	The user ID of the user.
String password	The password of the user.

isScheduler() Method

In SecurityService, use the isScheduler() method to check if the user has the SCHEDULE_REPORT privilege.

Signature

public boolean isScheduler(String userID, String password);

Parameters for isScheduler() Method

Parameter	Description
String userID	The user ID of the user.
String password	The password of the user.

isUserExists() Method

In SecurityService, use the isUserExists() method to test if a user name exists in the Publisher security model.

Signature

boolean isUserExists(String userName, String adminUser, String adminPassword);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for isUserExists() Method

Parameter	Description
String userName	The user name to test.
String adminUser	The user name for a user with administration privileges.
String adminPassword	The password for the administration user name.

login() Method

In SecurityService, use the login() method to log in to Publisher and perform other actions using Web Services.

The login() method returns a String, which will become the Publisher session ID.

Signature

String login(String userID, String password);

Parameters for login() Method

Parameter	Description
String userID	The user name.
String password	The password for the user name.

logout() Method

In SecurityService, use the logout() method, to effectively log the specified user out of the system by invalidating the user's bipSessionToken.

After successful logout, the bipSessionToken string is no longer valid.

Signature

boolean logout(String bipSessionToken);

Parameters for logout() Method

Parameter	Description
String bipSessionToken	The session ID.

notifyBIEEPreferencesUpdated() Method

In SecurityService, use the notifyBIEEPreferencesUpdated() method for user preference integration purposes only.

Signature

boolean notifyBIEEPreferencesupdated(bieeSessionID);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for notifyBIEEPreferencesUpdated() Method

Parameter	Description
String bieeSessionID	The session ID for the integration.

notifyBIEEPreferencesUpdatedWithString() Method

In SecurityService, use the notifyBIEEPreferencesUpdatedWithString() method for user preference integration purposes only.

Signature

boolean notifyBIEEPreferencesUpdatedWithString(String bieeSessionID, String userPrefesXML);

Parameters for notifyBIEEPreferencesUpdatedWithString() Method

Parameter	Description
String bieeSessionID	The session ID for integration.
String userPrefesXML	The XML data that contain user preferences.

removeRolesFromUser() Method

In SecurityService, use the removeRolesFromUser() method to remove roles from a user in Publisher.

Signature

String[] removeRolesFromUser(String userName, String[] roleNames, String adminUser, String adminPassword);

Parameters for removeRolesFromUser() Method

Parameter	Description
String userName	The user from which to delete the role or roles.
String[] roleNames	The name of the role to delete from the user. For example, "Financial Users".
String adminUser	Specifies a user name for a user with administration privileges.
String adminPassword	Specifies the password for the administration user name.

updateRole() Method

In SecurityService, use the updateRole() method to update the description of a role that currently exists in Publisher.

Signature

boolean updateRole(String currentRoleName, String newDescription, String adminUser, String adminPassword);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for updateRole() Method

Parameter	Description
String currentRoleName	The name of the role to update.
String newDescription	The updated description of the role to apply.
String adminUser	The user name for a user with administration privileges.
String adminPassword	The password for the administration user name.

updateUser() Method

In SecurityService, use the updateUser() method to update a user's password in Publisher.

This method returns a boolean value of the success of the method.

Signature

boolean updateUser(String currentUsername, String newPassword, String adminUser, String adminPassword);

Parameters for updateUser() Method

Parameter	Description
String currentUserName	The user name to update.
String newPassword	The new password to assign to the user name.
String adminUser	The user name for a user with administration privileges.
String adminPassword	The password for the administration user name.

validateLogin() Method

In SecurityService, use the validateLogin() method to validate that a UserID and Password have the privilege to access Publisher.

Signature

boolean validateLogin(String userID, String password);

Parameters for validateLogin() Method

Parameter	Description
String userID	The user name.
String password	The password for the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

6

Publisher CatalogService

Use the CatalogService methods to interact with the top-level catalog in Publisher.

CatalogService manages all report objects, including folders, reports, data models, style templates, and sub-templates, and provides methods for common operations, such as create, delete, copy, and rename.

Topics:

- copyObject() Method
- copyObjectInSession() Method
- createFolder() Method
- createFolderInSession() Method
- createObject() Method
- createObjectInSession() Method
- deleteObject() Method
- deleteObjectInSession() Method
- downloadObject() Method
- downloadObjectInSession() Method
- downloadXLIFF() Method
- downloadXLIFFInSession() Method
- getFolderContents() Method
- getFolderContentsInSession() Method
- getObject() Method
- getObjectInfo() Method
- getObjectInfoInSession() Method
- getObjectInSession() Method
- objectExist() Method
- objectExistInSession() Method
- renameObject() Method
- renameObjectInSession() Method
- updateObject() Method
- updateObjectInSession() Method
- uploadObject() Method
- uploadObjectInSession() Method
- uploadXLIFF() Method
- uploadXLIFFInSession() Method

Copyright © 2025, 2026, Oracle and/or its affiliates.

copyObject() Method

In CatalogService, use the copyObject() method to copy an object in the catalog.

Signature

boolean copyObject(String srcOjectAbsolutePath, String destFolderAbsolutePath, String newName, String userID, String password);

Parameters for copyObject() Method

Parameter	Description
String srcOjectAbsolutePath	The path to the catalog object to copy.
String destFolderAbsolutePath	The path to the folder in the catalog to which to copy the object.
String newName	The name to assign the new object.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

copyObjectInSession() Method

In CatalogService, use the copyObjectInSession() method to copy the report object referenced by srcObjectAbsolutePath to a destination folder specified by destFolderAbsolutePath.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean copyObjectInSession(String srcObjectAbsolutePath, String destFolderAbsolutePath, String newName, String bipSessionToken);

Parameters for copyObjectInSession() Method

Parameter	Description
String srcOjectAbsolutePath	The path to the catalog object to copy.
String destFolderAbsolutePath	The path to the folder in the catalog to which to copy the object.
String newName	The name to assign the new object.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

createFolder() Method

In CatalogService, use the createFolder() method to create a folder in the catalog.

Signature

String createFolder(String folderAbsolutePath, String userID, String password);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for createFolder() Method

Parameter	Description
String folderAbsolutePath	The path to the folder that you want to create. For example: /HR Manager/Employee Reports/.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

createFolderInSession() Method

In CatalogService, use the createFolderInSession() method to create a folder in the Publisher catalog for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

String createFolderInSession(String folderAbsolutePath, String bipSessionToken);

Parameter	Description
String folderAbsolutePath	The path to the folder that you want to create. For example: /HR Manager/Employee Reports/.
String bipSessionToken	The proprietary token string generated for the user by Publisher. With the bipSessionToken string, the user no longer needs to provide user credentials. Publisher can validate this token string and restore the Publisher session to perform needed operation.

createObject() Method

In CatalogService, use the createObject() method to create a Publisher object in the catalog.

> **Tip**
>
> If you have extensively used the createObject() or createObjectInSession() methods to dynamically create catalog objects, after the catalog objects are no longer required, clean up the catalog using deleteObject() or deleteObjectInSession() methods. Otherwise, the accumulation of the dynamically created catalog objects might cause severe catalog performance degradation issues.

Signature

String createObject(String folderAbsolutePathURL, String objectName, String objectType, String objectDescription, byte[] objectData, String userID, String password);

Parameters for createObject() Method

Parameter	Description
String folderAbsolutePathURL	The absolute path to the folder in the catalog in which to place the new object.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
String objectName	The name of the new object.
String objectType	The type of catalog object. Valid values are: `xdm` (data model) `xdo` (report) `xsb` (sub-template) `xss` (style template)
String objectDescription	Specifies the description of the new object.
byte[] objectData	Publisher object in bytes.
String userID	Specifies a user name.
String password	Specifies the password for the user name.

createObjectInSession() Method

In CatalogService, use the createObjectInSession() method to create a Publisher object in catalog for a given user.

> **Tip**
>
> If you have extensively used the createObject() or createObjectInSession() methods to dynamically create catalog objects, after the catalog objects are no longer required, clean up the catalog using deleteObject() or deleteObjectInSession() methods. Otherwise, the accumulation of the dynamically created catalog objects might cause severe catalog performance degradation issues.

For more information about in-session methods, see About In-Session Methods.

Signature

String createObjectInSession(String folderAbsolutePathURL, String objectName, String objectType, String objectDescription, byte[] objectData, String bipSessionToken);

Parameters for createObjectInSession() Method

Parameter	Description
String folderAbsolutePathURL	The absolute path to the folder in the catalog in which to place the new object.
String objectName	The name of the new object.
String objectType	The type of catalog object. Valid values are: `xdm` (data model) `xdo` (report) `xsb` (sub-template) `xss` (style template)
String objectDescription	Specifies the description of the new object.
byte[] objectData	Publisher object in bytes.

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameter	Description
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

deleteObject() Method

In CatalogService, use the deleteObject() method to delete an object from the catalog.

Signature

boolean deleteObject(String objectAbsolutePath, String userID, String password);

Parameters for deleteObject() Method

Parameter	Description
String objectAbsolutePath	The path to the object in the catalog to delete.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

deleteObjectInSession() Method

In CatalogService, use the deleteObjectInSession() method to delete an object from the catalog for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean deleteObjectInSession(String objectAbsolutePath, String bipSessionToken);

Parameters for deleteObjectInSession() Method

Parameter	Description
String objectAbsolutePath	The path to the object in the catalog to delete.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

downloadObject() Method

In CatalogService, use the downloadObject() method to download an object from the catalog. This method returns the requested object in binary.

Signature

byte[] downloadObject(String reportAbsolutePath, String userID, String password);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for downloadObject() Method

Parameter	Description
String reportAbsolutePath	The path to the object in the catalog to download.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

downloadObjectInSession() Method

In CatalogService, use the downloadObjectInSession() method to download an object in binary format from the catalog for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

byte[] downloadObjectInSession(String reportAbsolutePath, String bipSessionToken);

Parameters for downloadObjectInSession() Method

Parameter	Description
String reportAbsolutePath	The path to the object in the catalog to download.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the server session to perform needed operation.

downloadXLIFF() Method

In CatalogService, use the downloadXLIFF() method to download a translation file (XLIFF) in binary format from the catalog.

Signature

byte[] downloadXLIFF(String objectAbsolutePath, String locale, String userID, String password);

Parameters for downloadXLIFF() Method

Parameter	Description
String objectAbsolutePath	The path to the XLIFF object to download.
String locale	The locale of the XLIFF object (for example, en_US).
String userID	Specifies the user name.
String password	Specifies the password for the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

downloadXLIFFInSession() Method

In CatalogService, use the Use downloadXLIFFInSession() method to download a translation file (XLIFF) in binary format from the catalog downloadXLIFFInSession for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

byte[] downloadXLIFFInSession(String objectAbsolutePath, String locale, String bipSessionToken);

Parameters for downloadXLIFFInSession() Method

Parameter	Description
String objectAbsolutePath	The path to the XLIFF object to download.
String locale	The locale of the XLIFF object (for example, en_US).
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getFolderContents() Method

In CatalogService, use the getFolderContents() method to get all of the items in a folder.

This method returns the reports and folders contained in the specified folder. You can then use these items to determine what reports you might want to execute or what folders you may want to further search to identify a report.

See CatalogContents for a description of the return object.

Signature

CatalogContents getFolderContents(String folderAbsolutePath, String userID, String password);

Parameters for getFolderContents() Method

Parameter	Description
String folderAbsolutePath	The path to the folder for which to retrieve the contents. For example: /HR Manager/Employee Reports/
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getFolderContentsInSession() Method

In CatalogService, use the getFolderContentsInSession() to get all of the items in a folder for a given user.

Copyright © 2025, 2026, Oracle and/or its affiliates.

This will return all the reports and folders contained in the specified folder. You can then use these items to determine what reports you might want to execute or what folders you may want to further search to identify a report.

See CatalogContents for a description of the return object.

For more information about in-session methods, see About In-Session Methods.

Signature

CatalogContents getFolderContentsInSession(String folderAbsolutePath, String bipSessionToken);

Parameters for getFolderContentsInSession() Method

Parameter	Description
String folderAbsolutePath	The path to the folder for which to retrieve the contents. For example: /HR Manager/Employee Reports/.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getObject() Method

In CatalogService, use the getObject() method to download an object in binary format from the catalog.

Signature

byte[] getObject(String objectAbsolutePath, String locale, String userID, String password);

Parameters for getObject() Method

Parameter	Description
String objectAbsolutePath	The path to the object to download.
String locale	The locale of the object to get.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getObjectInfo() Method

In CatalogService, use the getObjectInfo() method to get information about an object in the catalog.

This method returns the CatalogObjectInfo object. See CatalogObjectInfo.

Signature

CatalogObjectInfo getObjectInfo(String objectAbsolutePath, String userID, String password);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for getObjectInfo() Method

Parameter	Description
String objectAbsolutePath	The path to the object about which to get information.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

getObjectInfoInSession() Method

In CatalogService, use the getObjectInfoInSession() method to get information about an object in the catalog for a given user.

This method returns the CatalogObjectInfo object. See CatalogObjectInfo.

For more information about in-session methods, see About In-Session Methods.

Signature

CatalogObjectInfo getObjectInfoInSession(String objectAbsolutePath, String bipSessionToken);

Parameters for getObjectInfoInSession() Method

Parameter	Description
String objectAbsolutePath	The path to the object about which to get information.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

getObjectInSession() Method

In CatalogService, use the getObjectInSession() method to download an object from the catalog for a given user.

This method returns the requested object file in binary.

For more information about in-session methods, see About In-Session Methods.

Signature

byte[] getObjectInSession(String objectAbsolutePath, String bipSessionToken);

Parameters for getObjectInSession() Method

Parameter	Description
String objectAbsolutePath	The path to the object to retrieve.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.

objectExist() Method

In CatalogService, use the objectExist() method to determine whether an object exists in the catalog.

Signature

boolean objectExist(String reportObjectAbsolutePath, String userID, String password);

Parameters for objectExist() Method

Parameter	Description
String reportObjectAbsolutePath	The path to the object to test for in the catalog. In the online report viewer or when you edit the report, if the path in the URL is /HR Manager/Employee%20Reports/Employee%20Data%20Model.xdm, replace the encoded characters with the corresponding ASCII character to get the correct absolute path of the report. For example: /HR Manager/Employee Reports/Employee Data Model.xdm.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

objectExistInSession() Method

In CatalogService, use the objectExistInSession() method to determine if an object exists in the catalog for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean objectExistInSession(String reportObjectAbsolutePath, String bipSessionToken);

Parameters for objectExistInSession() Method

Parameter	Description
String reportObjectAbsolutePath	The path to the object to test for in the catalog. For example: /HR Manager/Employee Reports/Employee Data Model.xdm
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

renameObject() Method

In CatalogService, use the renameObject() method to rename an object in the catalog.

Signature

boolean renameObject(String objectAbsolutePath, String newName, String userID, String password);

Copyright © 2025, 2026, Oracle and/or its affiliates.

Parameters for renameObject() Method

Parameter	Description
String objectAbsolutePath	The path to the object in the catalog to rename.
String newName	The new name to assign the object.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

renameObjectInSession() Method

In CatalogService, use the renameObjectInSession() method to rename an object in the catalog for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean renameObjectInSession(String objectAbsolutePath, String newName, String bipSessionToken);

Parameters for renameObjectInSession() Method

Parameter	Description
String objectAbsolutePath	The path to the object in the catalog to rename.
String newName	The new name for the object.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

updateObject() Method

In CatalogService, use the updateObject() method to update an object in the catalog.

Signature

boolean updateObject(String objectAbsolutePath, byte[] objectData, String userID, String password);

Parameters for updateObject() Method

Parameter	Description
String objectAbsolutePath	The path to the object to update in the catalog. For example: /HR Manager/Employee Reports/Employee Data Model.xdm
byte[] objectData	The data with which to update the object.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

updateObjectInSession() Method

In CatalogService, use the updateObjectInSession() method to update an object in the catalog for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean updateObjectInSession(String objectAbsolutePath, byte[] objectData, String bipSessionToken);

Parameters for updateObjectInSession() Method

Parameter	Description
String objectAbsolutePath	The path to the object to update in the catalog. For example: /HR Manager/Employee Reports/Employee Data Model.xdm
byte[] objectData	The data with which to update the object.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

uploadObject() Method

In CatalogService, use the uploadObject() method to upload a new object to the catalog.

Signature

String uploadObject(String reportObjectAbsolutePathURL, String objectType, byte[] objectZippedData, String userID, String password);

Parameters for uploadObject() Method

Parameter	Description
String reportObjectAbsolutePathURL	The path to the object in the catalog.
String objectType	The type of object to upload. Valid values are: `xdm` (data model) `xdo` (report) `xsb` (sub-template) `xss` (style template)
byte[] objectZippedData	The object to upload in zipped format.
String userID	Specifies the user name.
String password	Specifies the password for the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

uploadObjectInSession() Method

In CatalogService, use the uploadObjectInSession() method to upload a new object to the catalog for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

String uploadObject(String reportObjectAbsolutePathURL, String objectType, byte[] objectZippedData, String bipSessionToken);

Parameters for uploadObjectInSession() Method

Parameter	Description
String reportObjectAbsolutePathURL	The path to the object in the catalog.
String objectType	The type of object to upload. Valid values are: `xdm` (data model) `xdo` (report) `xsb` (sub-template) `xss` (style template)
byte[] objectZippedData	The object to upload in zipped format.
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

uploadXLIFF() Method

In CatalogService, use the uploadXLIFF() method to upload a translation file (XLIFF) to the catalog.

Signature

boolean uploadXLIFF(String objectAbsolutePath, byte[] xliffData, String locale, String userID, String password);

Parameters for uploadXLIFF() Method

Parameter	Description
String objectAbsolutePath	The path to the XLIFF object to upload.
byte[] xliffData	The XLIFF fie to upload.
String locale	The locale to assign to the XLIFF (for example, `en_US`).
String userID	Specifies the user name.
String password	Specifies the password for the user name.

Copyright © 2025, 2026, Oracle and/or its affiliates.

uploadXLIFFInSession() Method

In CatalogService, use the uploadXLIFFInSession() method to upload a translation file (XLIFF) to the catalog for a given user.

For more information about in-session methods, see About In-Session Methods.

Signature

boolean uploadXLIFF(String objectAbsolutePath, byte[] xliffData, String locale, String bipSessionToken);

Parameters for uploadXLIFFInSession() Method

Parameter	Description
String objectAbsolutePath	The path to the XLIFF object to upload.
byte[] xliffData	The XLIFF fie to upload.
String locale	The locale to assign to the XLIFF (for example, en_US).
String bipSessionToken	The proprietary token string generated for the user. With the bipSessionToken string, the user no longer needs to provide user credentials. The server can validate this token string and restore the session to perform needed operation.

Copyright © 2025, 2026, Oracle and/or its affiliates.