Informatica® Cloud MDM - Customer 360 for Salesforce

Winter 2022 Version 7.221

API Guide

Informatica Cloud MDM - Customer 360 for Salesforce API Guide
Winter 2022 Version 7.221
December 2021

Publication Date: 2021-12-24

Table of Contents

Preface

Refer to the *Informatica® Cloud MDM - Customer 360 for Salesforce API Guide* for information about the Cloud MDM - Customer 360 for Salesforce APIs. This guide is written for administrators who are responsible for implementing the APIs.

Informatica Resources

Informatica Network

Informatica Network hosts Informatica Global Customer Support, the Informatica Knowledge Base, and other product resources. To access Informatica Network, visit https://network.informatica.com.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to search Informatica Network for product resources such as documentation, how-to articles, best practices, and PAMs.

To access the Knowledge Base, visit https://kb.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

To get the latest documentation for your product, browse the Informatica Knowledge Base at https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx.

If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com.

Informatica Product Availability Matrixes

Product Availability Matrixes (PAMs) indicate the versions of operating systems, databases, and other types of data sources and targets that a product release supports. If you are an Informatica Network member, you can access PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions.

If you are an Informatica Network member, you can access Informatica Velocity resources at http://velocity.informatica.com.

If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that augment, extend, or enhance your Informatica implementations. By leveraging any of the hundreds of solutions from Informatica developers and partners, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through Online Support on Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
http://www.informatica.com/us/services-and-training/support-services/global-support-centers.

If you are an Informatica Network member, you can use Online Support at http://network.informatica.com.

CHAPTER 1

API Reference

This chapter includes the following topics:

API_AccountTriggerHandler Class Reference

Provides methods to control the execution of the Cloud MDM - Customer 360 for Salesforce (CC360) Account trigger.

You can:

- Enable or disable the trigger.
- Check the status of the trigger.
- Initialize and execute the trigger.

- Invoke the trigger logic for insert, update, and delete operations.
- Skip the account trigger processing.
- Skip the bean trigger processing.

Note: If you use the methods in this API, disable the CC360 Account trigger to ensure that the trigger is not executed twice.

Package Methods

- global API_AccountTriggerHandler ()
  The constructor for the Account Trigger Handler class.
- global void disableCloudMDMTrigger ()
  Disables the CC360 account trigger.
- global void enableCloudMDMTrigger ()
  Enables the CC360 account trigger.
- global Boolean getCloudMDMTriggerStatus ()
  Checks if the CC360 account trigger is active.
- global void initTrigger ()
  Initializes the CC360 account trigger.
- global void executeTrigger ()
  Executes the CC360 account trigger.
- global void Trigger_InsertBefore_Handler ()
  Invokes the trigger logic that must be executed before a new account is inserted.
- global void Trigger_InsertAfter_Handler ()
  Invokes the trigger logic that must be executed after a new account is inserted.
- global void Trigger_UpdateBefore_Handler ()
  Invokes the trigger logic that must be executed before an account is updated.
- global void Trigger_UpdateAfter_Handler ()
  Invokes the trigger logic that must be executed after an account is updated.
- global void Trigger_DeleteBefore_Handler ()
  Invokes the trigger logic that must be executed before an account is deleted.
- global void Trigger_DeleteAfter_Handler ()
  Invokes the trigger logic that must be executed after an account is deleted.

Static Package Functions

- global static void setForceSkipAccountTrigger (boolean state)
  Allows you to skip the execution of the CC360 account trigger.
- global static void setForceSkipBeanTrigger (boolean state)
  Allows you to skip the execution of the bean trigger.

Constructor & Destructor Documentation

global API_AccountTriggerHandler ()

The constructor for the Account Trigger Handler class.

Method Documentation

global void disableCloudMDMTrigger ()

Disables the CC360 account trigger.

global void enableCloudMDMTrigger ()

Enables the CC360 account trigger.

global void executeTrigger ()

Executes the CC360 account trigger.

This is a single wrapper method that executes the trigger logic for operations: insert, update, and delete (before and after).

global Boolean getCloudMDMTriggerStatus ()

Checks if the CC360 account trigger is active.

Returns:

TRUE if the account trigger is active.

global void initTrigger ()

Initializes the CC360 account trigger.

Exceptions:

DS_CloudMDMSkipTriggerException	If you skip the execution of the account trigger using the setForceSkipAccountTrigger method.

global static void setForceSkipAccountTrigger (boolean state)

Allows you to skip the execution of the CC360 account trigger.

Parameters:

state

Specify True to skip the account trigger.

global static void setForceSkipBeanTrigger (boolean state)

Allows you to skip the execution of the bean trigger.

Parameters:

state

Specify True to skip the account trigger.

global void Trigger_DeleteAfter_Handler ()

Invokes the trigger logic that must be executed after an account is deleted.

global void Trigger_DeleteBefore_Handler ()

Invokes the trigger logic that must be executed before an account is deleted.

global void Trigger_InsertAfter_Handler ()

Invokes the trigger logic that must be executed after a new account is inserted.

global void Trigger_InsertBefore_Handler ()

Invokes the trigger logic that must be executed before a new account is inserted.

global void Trigger_UpdateAfter_Handler ()

Invokes the trigger logic that must be executed after an account is updated.

global void Trigger_UpdateBefore_Handler ()

Invokes the trigger logic that must be executed before an account is updated.

API_CleansingStandardization Class Reference

Provides methods to retrieve the country name based on the ISO code or the ISO code based on the country name.

Package Methods

- global static String getIsoCodeFromCountry (string inCountry)
  Gets the ISO code for the specified country.
- global static String getCountryFromIsoCode (string isoCode)
  Gets the country name for the specified ISO code.

Method Documentation

global static String getIsoCodeFromCountry (String inCountry)

Gets the ISO code for the specified country.

Parameters:

inCountry

The country name for which you want the ISO code.

Returns:

The ISO code for the specified country.

global static String getCountryFromIsoCode (String isoCode)

Gets the country name for the specified ISO code.

Parameters:

isoCode

The ISO code for which you want the country name.

Returns:

The country name for the specified ISO code.

API_AdapterTestServices Class Reference

You can implement your adapter by using the sample implementation provided for integration with AdderssDoctor. However, to perform a unit test of the adapter you implement, you need to register the adapter with CC360. Use the method in this API class to register the adapter.

Package Method

- global static setExternalAdapterClassName (string classNamel)
  Allows you to register your adapter with CC360.

Method Documentation

global static String setExternalAdaptrClassName(String className)

Allows you to register your adapter with CC360.

Parameters:

className

The name of the class you want to register.

API_UserExit Class Reference

Provides methods to register user exit classes for accounts, contacts, and leads. User exit classes must be registered to invoke the implementer class. The class name specified as a parameter value must contain the namespace prefix <<NameSpacePrefix.ClassName>> if the implementing class is in some other managed package. If the implementing class is part of an inner class that is not part of a managed package, then the format is <<OuterClassName.InnerClassName>>. If the implementing class is part of an outer class, then the format is <<ClassName>>.

For example, during the merge process, if you want the users to have visibility over the details of the records before and after merge, you can use the API_UserExit class to create a utility that provides access to the merged record for a user who had access to the deleted record.

Package Methods

- global static setUserExitMergeAccountHandler (string className, Boolean isPartial)
  Sets up the user exit custom setting for merge of accounts.
- global static getUserExitMergeAccountHandler
  Gets the user exit custom setting for merge of accounts.
- global static setUserExitMergeLeadHandler(string className, Boolean isPartial)
  Sets up the user exit custom setting for merge of leads.
- global static getUserExitMergeLeadHandler
  Gets the user exit custom setting for merge of leads.
- global static setUserExitMergeContactHandler(string className, Boolean isPartial)
  Sets up the user exit custom setting for merge of contacts.
- global static getUserExitMergeContactHandler
  Gets the user exit custom setting for merge of contacts.

Method Documentation

global static setUserExitMergeAccountHandler (string className, Boolean isPartial)

Sets up the user exit custom setting for merge of accounts.

Parameters:

className

The name of the handler class that implements DS_Classes.ICloudMDMUserExit interface.

isPartial

Indicates whether the transaction must fail on partial completion.

global static getUserExitMergeAccountHandler

Gets the user exit custom setting for merge of accounts.

Returns:

A string with the name of the handler class and isPartial value separated by a colon in the format <<HandlerClass>>:<<Ispartial>>.

global static setUserExitMergeLeadHandler (string className, Boolean isPartial)

Sets up the user exit custom setting for merge of leads.

Parameters:

className

The name of the handler class that implements DS_Classes.ICloudMDMUserExit interface.

isPartial

Indicates whether the transaction must fail on partial completion.

global static getUserExitMergeLeadHandler

Gets the user exit custom setting for merge of leads.

Returns:

A string with the name of the handler class and isPartial value separated by a colon in the format <<HandlerClass>>:<<Ispartial>>.

global static setUserExitMergeContatcHandler (string className, Boolean isPartial)

Sets up the user exit custom setting for merge of contacts.

Parameters:

className

The name of the handler class that implements DS_Classes.ICloudMDMUserExit interface.

isPartial

Indicates whether the transaction must fail on partial completion.

global static getUserExitMergeContactHandler

Gets the user exit custom setting for merge of contacts.

Returns:

A string with the name of the handler class and isPartial value separated by a colon in the format <<HandlerClass>>:<<Ispartial>>.

API_MultiOrgImpl Class Reference

If an external bean for a Spoke record is created in the Hub Org, then you can use the method in this API class to get the enriched master bean from the Hub Org to the Spoke Org. You need to invoke this API from the Spoke Org. This REST API can be accessed via the following URL:

```
https://<server>.salesforce.com/services/apexrest/DSE/MasterData/
```

Package Methods

- global static doPost ()
  Sends a request from a Spoke Org to the Hub Org to get the updated master bean from the Hub Org.

Method Documentation

global static doPost()

While making a request to this API, Request object must be set in the HTTP request body.

```
public class GetMasterRequest
{
public Set<String> sObjectIds;
public String spokeOrgName;
public Datetime hubLastChangeTime;
public String sObjectType;
}
```

Parameters:

sObjectIds

The list of record IDs for which you want to get master data.

spokeOrgName

The Spoke Org name as defined in the data source on the Hub Org.

hubLastChangeTime

The time when the record was previously synchronizd with the Hub Org. This is used when a real-time callout is made from the Spoke Org user interface.

sObjectType

This must always be set to “Account”.

The method returns success or error response from the Hub Org. If the master beans are found, then the response is a list of GetMasterResponse Object for each input record with 200 as response status code.

```
public class GetMasterResponse
{
public String spokeSObjectId;
public Map<String,String> fieldNameValuePair;
public String responseMsg;
public DateTime lastUpdateTimeFromHub;
public String errorMsg;
```

Parameters:

spokeSObjectId

Spoke Org record ID.

fieldNameValuePair

Map that contains the list of enriched fields as keys and their respective values.

responseMsg

Contains the response status. The following are the response codes:

- CMDM_RESPONSE_MB_FOUND = 1, if MB is found for the Spoke's external bean.

- CMDM_RESPONSE_MB_NOT_FOUND= '2', if MB is not found, this can be because the external bean for the spoke is not created by Informatica Cloud Services or de-duplication Jobs are not run yet.
- CMDM_RESPONSE_MB_NO_UPDATE_FOUND = '3', if there has been not update found on the MB since the last update time.
- CMDM_RESPONSE_MB_DELETED= '4', if the master bean was existing and later found to be deleted on the Hub Org.

lastUpdateTimeFromHub

Provides the last update time stamp of the master bean.

API_Compare Class Reference

Provides methods to compare bean to bean, master bean to master bean, bean to master bean, or master bean to Bean and returns the following:

- Fuzzy comparison score
- Status that indicates whether duplicates found or not
- Status that indicates whether match found or not

Package Methods

- global static Decimal getScore (Id sObjectId1, Id sObjectId2, Boolean isPartial , Decimal threshold, String settingName)
  Compares a bean to bean, master bean to master bean, bean to master bean, or master bean to bean and returns the fuzzy comparison score.
- global static Decimal getScore (Id sObjectId1, Id sObjectId2, String settingName)
  Compares a bean to bean, master bean to master bean, bean to master bean, or master bean to bean and returns the fuzzy comparison score.
- global static boolean isDuplicate (Id sObjectId1, Id sObjectId2, String settingName)
  Compares a bean to bean, master bean to master bean, bean to master bean, or master bean to bean and returns True if objects are duplicates.
- global static boolean isMatch (Id sObjectId1, Id sObjectId2, String settingName)
  Compares a bean to bean, master bean to master bean, bean to master bean, or master bean to bean and returns True if objects match.

Method Documentation

global static Decimal getScore (Id sObjectId1, Id sObjectId2, Boolean isPartial , Decimal threshold, String settingName)

Compares a bean to bean, master bean to master bean, bean to master bean, or master bean to bean and returns the fuzzy comparison score. The match score is calculated using the specified match setting. If any

field with weightage > 100 does not match, score is returned as zero. Otherwise, a score is returned regardless of the value in the Duplicate Check Threshold and High Matching Threshold.

Parameters:

sObjectId1

The first Salesforce object ID.

sObjectId2

The second Salesforce object ID.

isPartial

Default is true. isPartial causes an early exit from the comparison when the negative score exceeds threshold.

threshold

The threshold score. If you do not specify a value, the parameter uses the Duplicate Check Threshold from the match setting.

settingName

The name of the match setting. For example, account.

Returns:

The fuzzy comparison score.

global static Decimal getScore (Id sObjectId1, Id sObjectId2, String settingName)

Compares a bean to bean, master bean to master bean, bean to master bean, or master bean to bean and returns the fuzzy comparison score. The match score is calculated using the specified match setting. If any field with weightage > 100 does not match, score is returned as zero. Otherwise, a score is returned regardless of the value in the Duplicate Check Threshold and High Matching Threshold.

Parameters:

sObjectId1

The first Salesforce object ID.

sObjectId2

The second Salesforce object ID.

settingName

The name of the match setting. For example, account.

Returns:

The fuzzy comparison score.

global static Boolean isDuplicate (Id sObjectId1, Id sObjectId2, String settingName)

Compares a bean to bean, master bean to master bean, bean to master bean, or master bean to bean and returns True if objects compared are duplicates. The score is calculated using the specified match setting. If the Total Score >= Duplicate Check Threshold of the match setting, the pair is considered a duplicate.

Parameters:

sObjectId1

The first Salesforce object ID.

sObjectId2

The second Salesforce object ID.

settingName

The name of the match setting.

Returns:

True or False. True indicates that the records are duplicate.

global static Boolean isMatch (Id sObjectId1, Id sObjectId2, String settingName)

Compares a bean to bean, master bean to master bean, bean to master bean, or master bean to bean and returns True if objects compared match each other. The score is calculated using the specified match setting. If the Total Score >= High Matching Threshold from the match setting, the pair is considered a match.

Parameters:

sObjectId1

The first Salesforce object ID.

sObjectId2

The second Salesforce object ID.

settingName

The name of the match setting.

Returns:

True or False. True indicates that the records match.

API_ContactTriggerHandler Class Reference

Provides methods to control the execution of the Cloud MDM - Customer 360 for Salesforce (CC360) Contact trigger.

You can:

- Enable or disable the trigger.
- Check the status of the trigger.

- Initialize and execute the trigger.
- Invoke the trigger logic for insert, update, and delete operations.

Note: If you use the methods in this API, disable the CC360 Contact trigger to ensure that the trigger is not executed twice.

Package Methods

- global API_ContactTriggerHandler ()
  The constructor for the Contact Trigger Handler class.
- global void disableCloudMDMTrigger ()
  Disables the CC360 contact trigger.
- global void enableCloudMDMTrigger ()
  Enables the CC360 contact trigger.
- global Boolean getCloudMDMTriggerStatus ()
  Checks if the CC360 contact trigger is active.
- global void initTrigger ()
  Initializes the CC360 contact trigger.
- global void executeTrigger ()
  Executes the CC360 contact trigger.
- global void Trigger_InsertBefore_Handler ()
  Invokes the trigger logic that must be executed before a new contact is inserted.
- global void Trigger_InsertAfter_Handler ()
  Invokes the trigger logic that must be executed after a new contact is inserted.
- global void Trigger_UpdateBefore_Handler ()
  Invokes the trigger logic that must be executed before a contact is updated.
- global void Trigger_UpdateAfter_Handler ()
  Invokes the trigger logic that must be executed after a contact is updated.
- global void Trigger_DeleteBefore_Handler ()
  Invokes the trigger logic that must be executed before a contact is deleted.
- global void Trigger_DeleteAfter_Handler ()
  Invokes the trigger logic that must be executed after a contact is deleted.

Constructor & Destructor Documentation

global API_ContactTriggerHandler ()

The constructor for the Contact Trigger Handler class.

Method Documentation

global void disableCloudMDMTrigger ()

Disables the CC360 contact trigger.

global void enableCloudMDMTrigger ()

Enables the CC360 contact trigger.

global void executeTrigger ()

Executes the CC360 contact trigger.

This is a single wrapper method that executes the trigger logic for operations: insert, update, and delete (before and after).

global Boolean getCloudMDMTriggerStatus ()

Checks if the CC360 contact trigger is active.

Returns:

TRUE if the contact trigger is active.

global void initTrigger ()

Initializes the CC360 contact trigger.

Exceptions:

DS_CloudMDMSkipTriggerException	when contact trigger logic is skipped.

global void Trigger_DeleteAfter_Handler ()

Invokes the trigger logic that must be executed after a contact is deleted.

global void Trigger_DeleteBefore_Handler ()

Invokes the trigger logic that must be executed before a contact is deleted.

global void Trigger_InsertAfter_Handler ()

Invokes the trigger logic that must be executed after a new contact is inserted.

global void Trigger_InsertBefore_Handler ()

Invokes the trigger logic that must be executed before a new contact is inserted.

global void Trigger_UpdateAfter_Handler ()

Invokes the trigger logic that must be executed after a contact is updated.

global void Trigger_UpdateBefore_Handler ()

Invokes the trigger logic that must be executed before a contact is updated.

API_CustomScoutClass Class Reference

Use this API to synchronize the custom objects.

The custom trigger of the respective custom object invokes this class.

Package Methods

- global API_CustomScoutClass (string aObjName)
  The constructor for the Custom Scout Class.
- global void triggerSynchronize (List< SObject > newObjects, List< SObject > oldObjects, boolean isInsert, boolean isUpdate, boolean isDelete, boolean isUnDelete, boolean isBefore, boolean isAfter)
  Triggers the synchronization process that creates beans for the custom objects based on the trigger state. CC360 does not trigger the synchronization process if you do not define the synchronization settings for the custom object and do not select the Synchronize option in the synchronization settings of the custom object.

Constructor & Destructor Documentation

global API_CustomScoutClass (stringaObjName)

The constructor for the Custom Scout Class.

Parameters:

aObjName

The name of the custom object to be synchronized.

Method Documentation

public void triggerSynchronize (List<SObject> newObjects, List<SObject> oldObjects, boolean isInsert, boolean isUpdate, boolean isDelete, boolean isUnDelete, boolean isBefore, boolean isAfter)

Triggers the synchronization process that creates beans for the custom objects based on the trigger state. The triggerSynchronize() method triggers the synchronization process only if you define the synchronization settings for the custom object and select the Synchronize option in the synchronization settings of the custom object.

Parameters:

newObjects

List of new versions of custom objects.

oldObjects

List of old versions of custom objects.

isInsert

Indicates whether to trigger the synchronization process for the insert operation.

Specify true to trigger the synchronization process, or specify false to not trigger the process.

isUpdate

Indicates whether to trigger the synchronization process for the update operation.

Specify true to trigger the synchronization process, or specify false to not trigger the process.

isDelete

Indicates whether to trigger the synchronization process for the delete operation.

Specify true to trigger the synchronization process, or specify false to not trigger the process.

isUnDelete

Indicates whether to trigger the synchronization process if the record is recovered from the Recycle Bin.

Specify true to trigger the synchronization process, or specify false to not trigger the process.

isBefore

Indicates whether to trigger the synchronization process before you save a record.

Specify true to trigger the synchronization process, or specify false to not trigger the process.

isAfter

Indicates whether to trigger the synchronization process after you save a record.

Specify true to trigger the synchronization process, or specify false to not trigger the process.

API_DataManager Class Reference

Provides methods to retrieve beans and master beans for a list of accounts, contacts, and leads.

Static Package Functions

- global static list< DS_Bean__c > getBeans (list< Account > accounts)
  Gets the beans for the list of specified accounts.
- global static list< DS_Bean__c > getBeans (list< Contact > contacts)
  Gets the beans for the list of specified contacts.
- global static list< DS_Bean__c > getBeans (list< Lead > leads)
  Gets the beans for the list of specified leads.
- global static list< DS_Bean__c > getBeans (list< DS_Master_Bean__c > masterBeans)
  Gets the beans for the list of specified master beans.
- global static list< DS_Master_Bean__c > getMasterBeans (list< Account > accounts)
  Gets the master beans for the list of specified accounts.
- global static list< DS_Master_Bean__c > getMasterBeans (list< Contact > contacts)
  Gets the master beans for the list of specified contacts.
- global static list< DS_Master_Bean__c > getMasterBeans (list< Lead > leads)
  Gets the master beans for the list of specified leads.

Method Documentation

global static list<DS_Bean__c> getBeans (list< Account > accounts)

Gets the beans for the list of specified accounts.

Parameters:

accounts

The list of accounts for which you want to get the beans.

Returns:

The beans for the specified accounts.

global static list<DS_Bean__c> getBeans (list< Contact > contacts)

Gets the beans for the list of specified contacts.

Parameters:

contacts

The list of contacts for which you want to get the beans.

Returns:

The beans for the specified contacts.

global static list<DS_Bean__c> getBeans (list< DS_Master_Bean__c > masterBeans)

Gets the beans for the list of specified master beans.

Parameters:

masterBeans

the list of master beans for which you want to the beans.

Returns:

The beans for the specified master beans.

global static list<DS_Bean__c> getBeans (list< Lead > leads)

Gets the beans for the list of specified leads.

Parameters:

leads

The list of leads for which you want to get the beans.

Returns:

The beans for the specified leads.

global static list<DS_Master_Bean__c> getMasterBeans (list< Lead > leads)

Gets the master beans for the list of specified leads.

Parameters:

leads

The list of leads for which you want to get the master beans.

Returns:

The master beans for the specified leads.

global static list<DS_Master_Bean__c> getMasterBeans (list< Account > accounts)

Gets the master beans for the list of specified accounts.

Parameters:

accounts

The list of accounts for which you want to get the master beans.

Returns:

The master beans for the specified accounts.

global static list<DS_Master_Bean__c> getMasterBeans (list< Contact > contacts)

Gets the master beans for the list of specified contacts.

Parameters:

contacts

The list of contacts for which you want to get the master beans.

Returns:

The master beans for the specified contacts.

API_DisableDaaSTrigger Class Reference

Disables or enables Data as a Service (DaaS) for standard and custom Salesforce objects.

Static Package Functions

- global static DisableDaaSTrigger_(Integer sObjectKeyPrefix, Integer Value)
  Disables or enables DaaS for an object with the specified key prefix.
- global static DisableDaaSTrigger_(String sObjectName, Integer Value)
  Disables or enables DaaS for a specified object.

Method Documentation

global static DisableDaaSTrigger_(Integer sObjectKeyPrefix, Integer Value)

Disables or enables DaaS for an object with the specified key prefix.

Parameters:

sObjectKeyPrefix

Indicates the key prefix of the object.

Value

Specify 1 to disable DaaS, and specify 0 to enable DaaS.

global static DisableDaaSTrigger_(String sObjectName, Integer Value)

Disables or enables DaaS for a specified object.

Parameters:

sObjectName

Indicates the name of the object.

Value

Specify 1 to disable DaaS, and specify 0 to enable DaaS.

API_DuplicateSearch Class Reference

Provides methods to search for duplicates by using the basic search or fuzzy search capability.

Static Package Functions

- global static list< DS_Bean__c > basicSearch (sObject obj, String searchMode, Set< String > searchSources, String sortOrder)
  Matches the fields in the object passed through the method directly with the fields in the database and returns the duplicate records.
- global static list< DS_Bean__c > fuzzySearch (sObject obj, Set< String > searchSources, String sortOrder)
  Uses the CC360 pattern matching algorithms to match the fields in the object passed through this method with the fields in the database and returns the duplicate records.

Method Documentation

global static list<DS_Bean__c> basicSearch (sObject obj, String searchMode, Set< String > searchSources, String sortOrder)

Matches the fields in the object passed through the method directly with the fields in the database and returns the duplicate records.

Parameters:

obj

The salesforce sObject to be searched for duplicates. You can use any salesforce sObject in the method. You can pass the object to the method without casting the object as an sObject type. The object to be searched need not exist in the database.

searchMode

Specifies whether the search matches all the fields or any one of the fields. Specify AND to return records by matching all the fields. Specify OR to return records by matching any one field. When you specify OR, the search might return a large number of records.

searchSources

The salesforce object to be compared with the sObject. You can compare the sObject passed through the method with any other type of salesforce object that is synchronized with CC360. You must define the object in CC360 data sources as part of custom settings. To search for duplicates, CC360 uses the match settings defined as part of custom settings.

sortOrder

Controls the sort order of the beans returned by the search. The recommended value is null.

Returns:

The list of duplicate records for the specified sObject.

global static list<DS_Bean__c> fuzzySearch (sObjectobj, Set< String > searchSources, String sortOrder)

Uses the CC360 pattern matching algorithms to match the fields in the object passed through this method with the fields in the database and returns the duplicate records.

Parameters:

obj

The salesforce sObject to be searched for duplicates. You can use any salesforce sObject in the method. You can pass the object to the method without casting the object as an sObject type. The object to be searched need not exist in the database.

searchSources

The salesforce object to be compared with the sObject. You can compare the sObject passed through the method with any other type of salesforce object that is synchronized with CC360. You must define the object in CC360 data sources as part of custom settings. CC360 uses the match settings defined as part of custom settings to search for duplicates.

sortOrder

Controls the sort order of the beans returned by the search. The recommended value is null.

Returns:

The list of duplicate records for the specified sObject.

API_EventQueueManager Class Reference

Provides a method to create data cleansing requests for historical accounts, contacts, and leads.

Static Package Functions

global static void loadDataQualityEvents(String sObjectName, List<Sobject> sobjRecords)

Creates transaction entries in the Event Queue table for a list of accounts, contacts, and leads. When the External Cleansing batch job is run, the batch job picks up records from the Event Queue table and sends the records to AddressDoctor® for cleansing addresses.

Method Documentation

global static void loadDataQualityEvents(String sObjectName, List<Sobject> sobjRecords)

Creates transaction entries in the Event Queue table for a list of accounts, contacts, and leads. When the External Cleansing batch job is run, the batch job picks up records from the Event Queue table and sends the records to AddressDoctor for cleansing addresses.

Parameters:

sObjectName

The salesforce object type; account, contact, or lead.

sobjRecords

The list of records corresponding to the Salesforce object type. For example, if you specify account as sObjectName, then you must specify a list of accounts for this parameter.

Note: Include all the relevant address fields for records that you pass as sobjRecords.

API_HierarchyHandler Class Reference

Provides methods to lock and unlock hierarchies based on root bean ID and account name.

When Cloud MDM - Customer 360 for Salesforce (CC360) performs an asynchronous operation on a hierarchy, the hierarchy must be locked to prevent users from modifying it.

Static Package Functions

- global static void LockCloudMDMHierarchy (Id rootNodeId)
  Locks the hierarchy for the specified root bean ID.
- global static void LockCloudMDMHierarchy (Set< Id > beanIds)
  Locks the hierarchies for the specified root bean IDs.
- global static void LockCloudMDMHierarchy (Account account)
  Locks the hierarchy that contains the specified account.

- global static void UnLockCloudMDMHierarchy (Id rootNodeId)
  Unlocks the hierarchy for the specified root bean ID.
- global static void UnLockCloudMDMHierarchy (Set< Id > beanIds)
  Unlocks the hierarchies for the specified root bean IDs.
- global static void UnLockCloudMDMHierarchy (Account account)
  Unlocks the hierarchy that contains the specified account.

Method Documentation

global static void LockCloudMDMHierarchy (Id rootNodeId)

Locks the hierarchy for the specified root bean ID.

Parameters:

rootNodeId

The root bean ID of the hierarchy to be locked.

Exceptions:

DS_CloudMDM_HierarchyAPIException	

global static void LockCloudMDMHierarchy (Set< Id > beanIds)

Locks the hierarchies for the specified root bean IDs.

Parameters:

beanIDs

The root bean IDs of the hierarchies to be locked.

Exceptions:

DS_CloudMDM_HierarchyAPIException	

global static void LockCloudMDMHierarchy (Account account)

Locks the hierarchy that contains the specified account.

Parameters:

account

The name of the account to be used to lock a hierarchy.

Exceptions:

DS_CloudMDM_HierarchyAPIException	

global static void UnLockCloudMDMHierarchy (Id rootNodeId)

Unlocks the hierarchy for the specified root bean ID.

Parameters:

rootNodeId

The root bean ID of the hierarchy to be unlocked.

Exceptions:

DS_CloudMDM_HierarchyAPIException	

global static void UnLockCloudMDMHierarchy (Set< Id > beanIds)

Unlocks the hierarchies for the specified root bean IDs.

Parameters:

beanIDs

The root bean IDs of the hierarchies to be unlocked.

Exceptions:

DS_CloudMDM_HierarchyAPIException	

global static void UnLockCloudMDMHierarchy (Account account)

Unlocks the hierarchy that contains the specified account.

Parameters:

account

The name of the account to be used to unlock a hierarchy.

Exceptions:

DS_CloudMDM_HierarchyAPIException	

API_JobManager Class Reference

Use this API to manage the different Cloud MDM - Customer 360 for Salesforce (CC360) batch jobs.

You can:

- Run a batch job.
- Run a batch job concurrently.
- Abort a batch job.

In addition, you can instantiate the following batch classes:

- Duplicate check and duplicate bean check

- Matching
- Consolidation
- Conversion
- Merge account, contact, and lead
- Migration

Static Package Functions

- global static boolean runJob (String jobName)
  Runs the specified CC360 batch job.
- global static boolean runConcurrentJob (String jobName, Integer concurrentBatchJobs)
  Runs the specified CC360 batch job.
- global static boolean abortJob (String jobName)
  Aborts the specified CC360 batch job.
- global static DS_DuplicateBatchClass instantiateDuplicateBatchClass (set< Id > jobIdSet, List< DS_Master_Bean__c > selectedMBeans, List< DS_Master_Bean__c > selectedCompMBeans)
  Creates an instance of the duplicate check batch class.
- global static DS_DuplicateBeanBatchClass instantiateDuplicateBeanBatchClass (set< Id > jobIdSet, List< DS_Bean__c > selectedBeans, List< DS_Bean__c > selectedCompBeans)
  Creates an instance of the duplicate bean check batch class.
- global static DS_MatchingBatchClass instantiateMatchingBatchClass (set< Id > jobIdSet, List< DS_Bean__c > selectedBeans, List< DS_Master_Bean__c > selectedMBeans)
  Creates an instance of the matching batch class to match beans with master beans.
- global static DS_ConsolidationBatchClass instantiateConsolidationBatchClass (set< Id > jobIdSet, List< DS_Bean__c > selectedBeans, List< DS_Master_Bean__c > selectedMBeans)
  Creates an instance of the consolidation batch class.
- global static DS_BeanConversionBatchClass instantiateBeanConversionBatchClass (set< Id > jobIdSet, List< DS_Bean__c > selectedBeans, List< DS_Bean__c > selectedCompBeans)
  Creates an instance of the bean conversion batch class job to convert beans into accounts, contacts, or leads.
- global static DS_MergeAccountBatchClass instantiateMergeAccountBatchClass (List< DS_Duplicates__c > selectedDupes)
  Creates an instance of the merge accounts batch class to merge duplicate accounts.
- global static DS_MergeContactBatchClass instantiateMergeContactBatchClass (List< DS_DuplicateBean__c > selectedDupes)
  Creates an instance of the merge contacts batch class to merge duplicate contacts.
- global static DS_MergeLeadBatchClass instantiateMergeLeadBatchClass (List< DS_DuplicateBean__c > selectedDupes)
  Creates an instance of the merge leads batch class to merge duplicate leads.
- global static DS_MigrationBatchClass instantiateMigrationBatchClass (set< Id > jobIdSet, String objName)
  Creates an instance of the migration batch class to migrate accounts, contacts, or leads.

Method Documentation

global static boolean abortJob (String jobName)

Aborts the specified CC360 batch job.

Parameters:

jobName

The name of the job you want to abort.

Returns:

True if the job is successfully aborted.

global static DS_BeanConversionBatchClass instantiateBeanConversionBatchClass (set< Id > jobIdSet, List< DS_Bean__c > selectedBeans, List< DS_Bean__c > selectedCompBeans)

Creates an instance of the bean conversion batch class job to convert beans into accounts, contacts, or leads.

Parameters:

jobIdSet

The list of bean IDs.

selectedBeans

The list of beans to be converted through the conversion job. The list is compared with the list of beans specified for the selectetCompBeans parameter. The list consists of beans for which the DS_Conversion_Status_c flag is set to To Be Converted.

selectedCompBeans

The list of beans to be used for comparison.

Returns:

The instance of the class.

global static DS_ConsolidationBatchClass instantiateConsolidationBatchClass (set< Id > jobIdSet, List< DS_Bean__c > selectedBeans, List< DS_Master_Bean__c > selectedMBeans)

Creates an instance of the consolidation batch class.

Parameters:

jobIdSet

The list of bean IDs.

selectedBeans

The list of beans to be consolidated through the consolidation batch job. The list consists of beans for which the DS_Consolidated_c flag is disabled.

selectedMBeans

The list of master beans to be consolidated.

Returns:

The instance of the class.

global static DS_DuplicateBatchClass instantiateDuplicateBatchClass (set< Id > jobIdSet, List< DS_Master_Bean__c > selectedMBeans, List< DS_Master_Bean__c > selectedCompMBeans)

Creates an instance of the duplicate check batch class.

Parameters:

jobIdSet

The list of master bean IDs.

selectedMBeans

The list of master beans to be compared with the list of master beans specified for the selectedCompMBeans parameter. The list consists of master beans for which the DS_Duplicate_Checked_c flag is disabled.

selectedCompMBeans

The list of master beans to be used for comparison.

Returns:

The instance of the class.

global static DS_DuplicateBeanBatchClass instantiateDuplicateBeanBatchClass (set< Id > jobIdSet, List< DS_Bean__c > selectedBeans, List< DS_Bean__c > selectedCompBeans)

Creates an instance of the duplicate bean check batch class.

Parameters:

jobIdSet

The list of bean IDs.

selectedBeans

The list of beans to be compared with the list of beans specified for the selectedCompBeans parameter. The list consists of beans for which the DS_Duplicate_Checked_c flag is disabled.

selectedCompBeans

The list of beans to be used for comparison.

Returns:

The instance of the class.

global static DS_MatchingBatchClass instantiateMatchingBatchClass (set< Id > jobIdSet, List< DS_Bean__c > selectedBeans, List< DS_Master_Bean__c > selectedMBeans)

Creates an instance of the matching batch class to match beans with master beans.

Parameters:

jobIdSet

The list of bean IDs.

selectedBeans

The list of beans to be matched through the matching batch job. The list is matched with the list of master beans specified for the selectedMBeans parameter.

selectedMBeans

The list of master beans to be used for matching.

Returns:

The instance of the class.

global static DS_MergeAccountBatchClass instantiateMergeAccountBatchClass (List< DS_Duplicates__c > selectedDupes)

Creates an instance of the merge accounts batch class to merge duplicate accounts.

Parameters:

selectedDupes

The list of duplicate accounts to be merged.

Returns:

The instance of the class.

global static DS_MergeContactBatchClass instantiateMergeContactBatchClass (List< DS_DuplicateBean__c > selectedDupes)

Creates an instance of the merge contacts batch class to merge duplicate contacts.

Parameters:

selectedDupes

The list of duplicate contacts to be merged.

Returns:

The instance of the class.

global static DS_MergeLeadBatchClass instantiateMergeLeadBatchClass (List< DS_DuplicateBean__c > selectedDupes)

Creates an instance of the merge leads batch class to merge duplicate leads.

Parameters:

selectedDupes

The list of duplicate leads to be merged.

Returns:

The instance of the class.

global static DS_MigrationBatchClass instantiateMigrationBatchClass (set< Id > jobIdSet, StringobjName)

Creates an instance of the migration batch class to migrate accounts, contacts, or leads.

Parameters:

jobIdSet

The list of sObjects to be migrated.

objName

sObject type for which beans and master beans have to be created.

Returns:

The instance of the class.

global static boolean runConcurrentJob (String jobName, Integer concurrentBatchJobs)

Runs the specified CC360 batch job.

Parameters:

jobName

The name of the job you want to run.

concurrentBatchJobs

The number of batch jobs that you want to run concurrently.

Returns:

True if the job runs successfully.

global static boolean runJob (String jobName)

Runs the specified CC360 batch job.

Parameters:

jobName

The name of the job you want to run. Use one of the following values for the job name:

- Matching
- DuplicateCheck
- DuplicateBeanCheck
- Consolidation
- BeanConversion
- MergeAccount
- MergeLead
- MergeContact
- MigrateAccount
- MigrateContact

- MigrateLead
- Maintenance
- SyncReportingData
- Data as a Service - <Template Name>

Returns:

True if the job runs successfully.

API_LeadTriggerHandler Class Reference

Provides methods to control the execution of the Cloud MDM - Customer 360 for Salesforce (CC360) Lead trigger.

You can:

- Enable or disable the trigger.
- Check the status of the trigger.
- Initialize and execute the trigger.
- Invoke the trigger logic for insert, update, and delete operations.

Note: If you use the methods in this API, disable the CC360 Lead trigger to ensure that the trigger is not executed twice.

Package Methods

- global API_LeadTriggerHandler ()
  The constructor for the Lead Trigger Handler class.
- global void disableCloudMDMTrigger ()
  Disables the CC360 lead trigger.
- global void enableCloudMDMTrigger ()
  Enables the CC360 lead trigger.
- global Boolean getCloudMDMTriggerStatus ()
  Checks if the CC360 lead trigger is active.
- global void initTrigger ()
  Initializes tha CC360 lead trigger.
- global void executeTrigger ()
  Executes the CC360 lead trigger.
- global void Trigger_InsertBefore_Handler ()
  Invokes the trigger logic that must be executed before a new lead is inserted.
- global void Trigger_InsertAfter_Handler ()
  Invokes the trigger logic that must be executed after a new lead is inserted.
- global void Trigger_UpdateBefore_Handler ()
  Invokes the trigger logic that must be executed before a lead is updated.
- global void Trigger_UpdateAfter_Handler ()
  Invokes the trigger logic that must be executed after a lead is updated.

- global void Trigger_DeleteBefore_Handler ()
  Invokes the trigger logic that must be executed before a lead is deleted.
- global void Trigger_DeleteAfter_Handler ()
  Invokes the trigger logic that must be executed after a lead is deleted.

Constructor & Destructor Documentation

global API_LeadTriggerHandler ()

The constructor for the Lead Trigger Handler class.

Method Documentation

global void disableCloudMDMTrigger ()

Disables the CC360 lead trigger.

global void enableCloudMDMTrigger ()

Enables the CC360 lead trigger.

global void executeTrigger ()

Executes the CC360 lead trigger.

This is a single wrapper method that executes the trigger logic for operations: insert, update, and delete (before and after).

global Boolean getCloudMDMTriggerStatus ()

Checks if the CC360 lead trigger is active.

Returns:

TRUE if the lead trigger is active.

global void initTrigger ()

Initializes tha CC360 lead trigger.

Exceptions:

DS_CloudMDMSkipTriggerException	when lead trigger logic is skipped.

global void Trigger_DeleteAfter_Handler ()

Invokes the trigger logic that must be executed after a lead is deleted.

global void Trigger_DeleteBefore_Handler ()

Invokes the trigger logic that must be executed before a lead is deleted.

global void Trigger_InsertAfter_Handler ()

Invokes the trigger logic that must be executed after a new lead is inserted.

global void Trigger_InsertBefore_Handler ()

Invokes the trigger logic that must be executed before a new lead is inserted.

global void Trigger_UpdateAfter_Handler ()

Invokes the trigger logic that must be executed after a lead is updated.

global void Trigger_UpdateBefore_Handler ()

Invokes the trigger logic that must be executed before a lead is updated.

API_LicenseManager Class Reference

Provides methods to enable a Cloud MDM - Customer 360 for Salesforce (CC360) feature based on certain parameters.

Static Package Functions

- global static void setFeature (String featureName, boolean enabled, String value, Date expirationDate)
  Enables a CC360 feature based on certain parameters.
- global static void setFeature (String featureName, boolean enabled)
  Enables the specified CC360 feature.
- global static void setFeature (String featureName, boolean enabled, String value, Date expirationDate, string password)
  Enables a CC360 feature based on certain parameters.
- global static void setFeature (String featureName, boolean enabled, string password)
  Enables a CC360 feature based on certain parameters.
- global static boolean isFeatureEnabled (String featureName)
  Checks if a CC360 feature is enabled or not and returns the status.

Method Documentation

global static boolean isFeatureEnabled (String featureName)

Checks if a CC360 feature is enabled or not and returns the status.

Parameters:

featureName

The name of the feature.

Returns:

The status of the feature (enabled or disabled).

global static void setFeature (String featureName, boolean enabled)

Enables the specified CC360 feature.

Parameters:

featureName

The name of the feature to be enabled.

enabled

Specify True to enable the feature.

global static void setFeature (String featureName, boolean enabled, string password)

Enables a CC360 feature based on certain parameters.

Parameters:

featureName

The name of the feature to be enabled.

enabled

Specify True to enable the feature.

password

The password to access the feature.

global static void setFeature (String featureName, boolean enabled, String value, Date expirationDate, String password)

Enables a CC360 feature based on certain parameters.

Parameters:

featureName

The name of the feature to be enabled.

enabled

Specify True to enable the feature.

value

The value of the feature.

expirationDate

The date on which the feature will expire.

password

The password to access the feature.

global static void setFeature (String featureName, boolean enabled, String value, Date expirationDate)

Enables a CC360 feature based on certain parameters.

Parameters:

featureName

The name of the feature to be enabled.

enabled

Specify True to enable the feature.

value

The value of the feature.

expirationDate

The date on which the feature will expire.

API_QueueManager Class Reference

Provideds methods to integrate an external apex batch into Cloud MDM - Customer 360 for Salesforce (CC360) Batch Queue.

The Salesforce platform limits the number of apex batches to 5. To overcome this limit, you can integrate apex batches into the CC360 Batch Queue. This class contains two parts:

- Integrate or Insert external apex batch job into CC360 Job Queue.
- Static function that must be invoked from the external apex batch job code to be integrated.

To integrate an external apex batch job into the CC360 batch queue:

- Instantiate the CC360 Queue Manager class (API_QueueManager qManager = new API_QueueManager ('CustomBatch', 'ETL_CustomApexBatchJob');)
- Set optional quick parameters (if required).
- Set optional extended parameters (if required).
- Submit the external apex job to the CC360 Batch Queue

Static Public Attributes

- static final integer MAX_JOBNAME_SZ = 25

Package Methods

- global API_QueueManager (String jobName, String jobClassName)
  The constructor for the CC360 Queue Manager class.
- global void setQuickParameters (List< String > quickParameters)
  Sets quick parameters for the external apex job to be integrated.
- global void pushExtendedParameters (String parameterName, List< String > parameterValues)
  Pushes extended parameters for the external apex job to be integrated.
- global Id submitJob ()
  Submits the external apex batch job to the CC360 Batch Queue.
- global Id submitJob (Integer batchSize)
  Submits the external apex batch job to the CC360 Batch Queue with information about the size of the batch job.
- global Id submitJob (Boolean allowConcurrentExecution)
  Submits the external apex batch job to the CC360 Batch Queue with the flag to enable concurrent execution.
- global Id submitJob (Boolean allowConcurrentExecution, Id dependsOnQueueEntry)
  Submits the external apex batch job to the CC360 Batch Queue with the flag to enable concurrent exection and the queue ID of an existing job in the CC360 Batch Queue.
- global void setBatchSize (Integer batchSize)
  Sets the batch size for the external apex batch job to be integrated into CC360 Batch Queue.

Static Package Functions

- global static Id start (Database.BatchableContext BC)
  Gets the queue ID from the CC360 Batch Queue using the batch ID of the apex job.
- global static void updateExecutionProgress (Id batchQueueId, Integer totalRecords, Integer itemProcessed)
  Updates the batch status with the total items available in a submitted apex job and the number of items processed by the job.
- global static void finish (Id batchQueueId, BATCH_STATUS batchStatus)
  Triggers the next job in the queue When one submitted apex job is completed successfully.
- global static Id finish (Id batchQueueId)
  Overloaded Finish method to be used when you want to loop to next batch job.
- global static void setQuickParameters (Id batchQueueId, String phase, List< String > quickParameters)
  Sets the quick parameters for an apex job to be submitted to the CC360 Batch Queue.
- global static List< String > getQuickParameters (Id batchQueueId)
  Gets the quick parameters for the apex batch job submitted to the CC360 Batch Queue.
- global static void setExtendedParameters (Id batchQueueId, String parameterName, List< String > parameterValues)
  Updates the values of the extended parameter for the apex job submitted to the CC360 Batch Queue.
- global static void setExtendedParameters (Id batchQueueId, String parameterName, List< String > parameterValues, Boolean retainOld)
  Updates the values of the extended parameter for the apex job submitted to the CC360 Batch Queue.
- global static List< String > getExtendedParameters (Id batchQueueId, String parameterName)
  Gets the list of values of the extended parameter for the apex job submitted to the CC360 Batch Queue.

- global static Id registerJobLoop (Id batchQueueId)
  Registers a batch job loop.
- global static void submitJob (Id batchQueueId)
  Submits the specified to a batch job loop.
- global static Id registerJobLoop (Id batchQueueId, List< String > quickParameters)
  Registers a batch job loop along with quick parameters.
- global static Id registerJobLoop (Id batchQueueId, List< String > quickParameters, Boolean retainExtendedParams, Boolean skipLoopLogic)
  Registers a batch job loop along with quick parameters.
- global static void UpdateStatus (Id batchQueueId, String phase, Integer itemProcessed)
  Updates the batch status with the phase of the submitted apex job and the total items processed by the job.
- global static void UpdateStatus (Id batchQueueId, String phase, Integer itemProcessed, Boolean isItemProcessedDelta)
  Updates the batch status with the phase of the submitted apex job and the total items processed by the job.
- global static void UpdateStatus (Id batchQueueId, String phase, BATCH_STATUS batchStatus, Integer totalRecords, Integer itemProcessed)
  Updates the batch status with:
- global static void UpdateStatus (Id batchQueueId, String phase, BATCH_STATUS batchStatus, Integer totalRecords, Integer itemProcessed, Boolean isItemProcessedDelta)
  Updates the batch status with:
- global static String getStatus (Id batchQueueId)
  Gets the status of the submitted apex batch job from the CC360 Batch Queue.
- global static void refreshQueueManager ()
  Refreshes the CC360 Batch Queue.

Constructor & Destructor Documentation

global API_QueueManager (String jobName, String jobClassName)

The constructor for the CC360 Queue Manager class.

Parameters:

jobName

The name of the external apex batch job to be integrated into the CC360 Batch Queue.

jobClassName

The name of the batch job class. The name must include the namespace prefix if the class is part of a managed package.

Exceptions:

DS_QueueManagerException	

Method Documentation

global static void finish (Id batchQueueId, BATCH_STATUS batchStatus)

Triggers the next job in the queue When one submitted apex job is completed successfully.

Note: This method must be invoked from the external batch finish method along with the status of the apex batch.

Parameters:

batchQueueId

The queue ID in the CC360 Batch Queue.

batchStatus

The status of the batch job (SUCCESS, ABORTED, OR FAILURE).

global static Id finish (Id batchQueueId)

Overloaded Finish method to be used when you want to loop to next batch job.

Note: This method must be invoked from external batch finish method to loop to next batch job.

Parameters:

batchQueueId

The queue ID in the CC360 Batch Queue.

Returns:

The CC360 Batch Queue ID.

global static List<String> getExtendedParameters (Id batchQueueId, String parameterName)

Gets the list of values of the extended parameter for the apex job submitted to the CC360 Batch Queue.

Parameters:

batchQueueId

The queue ID in the CC360 Batch Queue.

parameterName

The name of the extended parameter.

Returns:

The list of values of the extended parameter.

global static List<String> getQuickParameters (Id batchQueueId)

Gets the quick parameters for the apex batch job submitted to the CC360 Batch Queue.

Parameters:

batchQueueID

The queue ID in the CC360 Batch Queue.

Returns:

The list of quick parameters.

global static String getStatus (Id batchQueueId)

Gets the status of the submitted apex batch job from the CC360 Batch Queue.

The function may return any of the following values:

- Registered - Internal Status
- Waiting - Internal Status
- Looping - Internal Status
- Queued - Intenal Status
- Running - Internal Status
- Completed - BATCH_STATUS.SUCCESS
- Failed - BATCH_STATUS.FAILURE
- Aborted - BATCH_STATUS.ABORTED

Parameters:

batchQueueId

The queue ID of the submitted apex job in the CC360 Batch Queue.

Returns:

The current status of the job.

global void pushExtendedParameters (String parameterName, List< String > parameterValues)

Pushes extended parameters for the external apex job to be integrated.

Each parameter is stored as a key value (list of string) pair.

Note: This function must be invoked multiple times to add multiple parameters.

Parameters:

parameterName

The name of the extended parameter to be pushed.

parametervalues

The list of values of the extended parameter.

global static void refreshQueueManager ()

Refreshes the CC360 Batch Queue.

Invoke this function for manually refreshing the queue.

global static Id registerJobLoop (Id batchQueueId)

Registers a batch job loop.

Parameters:

batchQueueId

The queue ID of a current batch job in the CC360 Batch Queue.

Returns:

The CC360 Batch queue ID.

global static Id registerJobLoop (Id batchQueueId, List< String > quickParameters)

Registers a batch job loop along with quick parameters.

Parameters:

batchQueueId

The queue ID of a current batch job in the CC360 Batch Queue.

quickParameters

The list of quick parameters to be set. You can set a maximum of 9 quick parameters.

Returns:

The CC360 Batch queue ID.

global static Id registerJobLoop (Id batchQueueId, List< String > quickParameters, Boolean retainExtendedParams, Boolean skipLoopLogic)

Registers a batch job loop along with quick parameters.

In addition, you can retain extended parameters and skip loop logic.

Parameters:

batchQueueId

The ID of a current batch job in the CC360 Batch Queue.

quickParameters

Note: The list of quick parameters to be set. No validation for quick parameters. If no quick parameters to pass, you can pass null.

retainExtendedParams

Flag to retain previously added extended parameters. If false, old parameters are deleted and added again.

skipLoopLogic

Required mainly for concurrent jobs. TRUE for non-concurrent jobs.

Returns:

The CC360 batch queue ID.

global void setBatchSize (Integer batchSize)

Sets the batch size for the external apex batch job to be integrated into CC360 Batch Queue.

This function must be invoked before the submitJob functions.

Parameters:

batchSize

The batch size of the external apex batch job.

Exceptions:

DS_QueueManagerException	

global static void setExtendedParameters (Id batchQueueId, String parameterName, List< String > parameterValues)

Updates the values of the extended parameter for the apex job submitted to the CC360 Batch Queue.

Parameters:

batchQueueId

The queue ID in the CC360 Batch Queue.

parameterName

The name of the extended parameter.

parameterValues

The list of values to be updated.

global static void setExtendedParameters (Id batchQueueId, String parameterName, List< String > parameterValues, Boolean retainOld)

Updates the values of the extended parameter for the apex job submitted to the CC360 Batch Queue.

Parameters:

batchQueueId

The queue ID in the CC360 Batch Queue.

parameterName

The name of the extended parameter.

parameterValues

The list of values to be updated.

retainOld

Flag to retain old parameter. If false, the old parameter is deleted and a new parameter added.

global static void setQuickParameters (Id batchQueueId, String phase, List< String > quickParameters)

Sets the quick parameters for an apex job to be submitted to the CC360 Batch Queue.

You can set a maximum of 9 quick parameters.

Parameters:

batchQueueId

The queue ID in the CC360 Batch Queue.

phase

The phase (Start, Execute, or Finish) of the submitted Job.

quickParameters

The list of quick parameters to be set.

global void setQuickParameters (List< String > quickParameters)

Sets quick parameters for the external apex job to be integrated.

Parameters:

quickParameters

The list of quick parameters to be set. You can set a maximum of 9 quick parameters.

Exceptions:

DS_QueueManagerException	

global static Id start (Database.BatchableContext BC)

Gets the queue ID from the CC360 Batch Queue using the batch ID of the apex job.

Note: This function must be invoked from the external batch start method.

Parameters:

BC

The external apex batch job context.

Returns:

Id The CC360 batch queue ID.

global Id submitJob (Integer batchSize)

Submits the external apex batch job to the CC360 Batch Queue with information about the size of the batch job.

Parameters:

batchSize

The size of the batch. The input batch size overrides the batch size configured as part of Performance Settings in CC360.

Returns:

The job ID in the CC360 Batch Queue.

Exceptions:

DS_QueueManagerException	

global static void submitJob (Id batchQueueId)

Submits the specified to a batch job loop.

Note: This function must be invoked immediately after 'registerJobLoop'.

Parameters:

batchQueueId

The queue ID in the CC360 Batch Queue.

global Id submitJob (Boolean allowConcurrentExecution, Id dependsOnQueueEntry)

Submits the external apex batch job to the CC360 Batch Queue with the flag to enable concurrent exection and the queue ID of an existing job in the CC360 Batch Queue.

The concurrent execution depends on the queue ID of the existing job.

Parameters:

allowConcurrentExecution

If TRUE, the job is enabled for concurrent execution.

dependsOnQueueEntry

The queue ID of an existing job in the CC360 Batch Queue.

Returns:

The job ID in the CC360 Batch Queue.

Exceptions:

DS_QueueManagerException	

global Id submitJob (Boolean allowConcurrentExecution)

Submits the external apex batch job to the CC360 Batch Queue with the flag to enable concurrent execution.

Parameters:

allowConcurrentExecution

If TRUE, the job is enabled for concurrent execution.

Returns:

The job ID in the CC360 Batch Queue.

Exceptions:

DS_QueueManagerException	

global Id submitJob ()

Submits the external apex batch job to the CC360 Batch Queue.

Returns:

The job ID in the CC360 Batch Queue.

global static void updateExecutionProgress (Id batchQueueId, Integer totalRecords, Integer itemProcessed)

Updates the batch status with the total items available in a submitted apex job and the number of items processed by the job.

Note: This function must be invoked from the external batch execute method.

Parameters:

batchQueueId

The queue ID in the CC360 Batch Queue.

totalRecords

The total records present in the batch job.

itemProcessed

The number of records processed by the job.

global static void UpdateStatus (Id batchQueueId, String phase, BATCH_STATUS batchStatus, Integer totalRecords, Integer itemProcessed)

Updates the batch status with:

The phase and status of the submitted apex job.

The total number of items in the apex job and the number of items processed by the job.

Parameters:

batchQueueId

The queue ID of the submitted apex job in the CC360 Batch Queue.

phase

The phase of the job (Start, Execute, or Finish).

batchStatus

The status of the job (SUCCESS, ABORTED, OR FAILURE).

totalRecords

The total number of items in the job.

itemProcessed

The Total number of items processed by job.

global static void UpdateStatus (Id batchQueueId, String phase, Integer itemProcessed, Boolean isItemProcessedDelta)

Updates the batch status with the phase of the submitted apex job and the total items processed by the job.

This function also checks if the items processed are delta items.

Parameters:

batchQueueId

The queue ID of the submitted apex job in the CC360 Batch Queue.

phase

The phase of the job (Start, Execute, or Finish).

itemProcessed

The Total number of items processed by job.

isItemProcessedDelta

If True, the items are added to the total number of items processed for the itemProcessed parameter.

global static void UpdateStatus (Id batchQueueId, String phase, BATCH_STATUS batchStatus, Integer totalRecords, Integer itemProcessed, Boolean isItemProcessedDelta)

Updates the batch status with:

The phase and status of the submitted apex job.

The total number of items in the apex job and the number of items processed by the job. In addition, this function also checks if the items processed are delta items.

Parameters:

batchQueueId

The queue ID of the submitted apex job in the CC360 Batch Queue.

phase

The phase of the job (Start, Execute, or Finish).

batchStatus

The status of the job (SUCCESS, ABORTED, OR FAILURE).

totalRecords

The total number of items in the job.

itemProcessed

The Total number of items processed by job.

isItemProcessedDelta

If True, the items are added to the total number of items processed for the itemProcessed parameter.

global static void UpdateStatus (Id batchQueueId, String phase, Integer itemProcessed)

Updates the batch status with the phase of the submitted apex job and the total items processed by the job.

Parameters:

batchQueueId

The queue ID of the submitted job in the CC360 Batch Queue.

phase

The phase of the job (Start, Execute, or Finish).

itemProcessed

The Total number of items processed by the submitted apex job.

Member Data Documentation

final integer MAX_JOBNAME_SZ = 25 [static]

API_SettingsManager Class Reference

Provides methods to manage the different custom settings in Cloud MDM - Customer 360 for Salesforce (CC360).

You can retrieve the following settings:

- CC360 settings.
- Account synchronization settings.
- Contact synchronization settings.
- Lead synchronization settings.
- Account enrichment settings.
- Synchronization settings.
- Matching settings.
- Source settings.
- Performance settings.

In addition, you can perform the following tasks:

- Update a custom setting with a value or long value.
- Enrich the hierarchy of an account.
- Enable the comparison of a leads and contacts associated with person accounts.
- Set master response size.
- Enable default job priority.

Static Package Functions

- global static DS_Settings__c getDataScoutSettings ()
- global static DS_Settings__c getCloudMDMSettings ()
  Gets the CC360 settings.
- global static DS_SF_Synchronization_Settings__c getAccountSyncSettings ()
  Gets the synchronization settings for accounts.
- global static DS_SF_Synchronization_Settings__c getAccountSyncSettings (String recType)
  Gets the synchronization settings of the specified account record type.
- global static DS_SF_Synchronization_Settings__c getPersonAccountSyncSettings ()
  Gets the synchronization settings of person accounts.
- global static DS_SF_Synchronization_Settings__c getPersonAccountSyncSettings (String recType)
  Gets the synchronization settings of the specified person account record type.

- global static DS_SF_Synchronization_Settings__c getContactSyncSettings ()
  Gets the synchronization settings for contacts.
- global static DS_SF_Synchronization_Settings__c getContactSyncSettings (String recType)
  Gets the synchronization settings of the specified contact record type.
- global static DS_SF_Synchronization_Settings__c getLeadSyncSettings ()
  Gets the synchronization settings for leads.
- global static DS_SF_Synchronization_Settings__c getLeadSyncSettings (String recType)
  Gets the synchronization settings of the specified lead record type.
- global static DS_SF_Enrichment_Settings__c getAccountEnrichmentSettings ()
  Gets the enrichment settings for accounts.
- global static DS_SF_Enrichment_Settings__c getAccountEnrichmentSettings (String recType)
  Gets the enrichment settings of the specified account record type.
- global static DS_SF_Synchronization_Settings__c getSyncSettings (String objectName)
  Gets the synchronization settings for the specified object.
- global static DS_SF_Synchronization_Settings__c getSyncSettings (String objectName, String recType)
  Gets the synchronization settings of the record type of the specified object.
- global static DS_Matching__c getMatchingSettings (String objectClass, String recordTypeName)
  Gets the matching settings for the specified object class and record type.
- global static DS_Sources__c getSourceSettings (String sourceName)
  Gets the source settings for the specified source.
- global static DS_Sources__c getSourceSettings (String sourceName, String recTypeName)
  Gets the source settings of the record type of the specified source.
- global static DS_Performance__c getPerformanceSettings (String performanceSettingName)
  Gets the performance settings for the specified setting name.
- global static void updateInternalSetting (String internalSettingName, String value)
  Updates an internal setting with the specified value.
- global static void updateInternalSetting (String internalSettingName, long value)
  Updates a custom setting setting with the specified long value.
- global static void enableLeadPersonAccountComparison (boolean value)
  If this setting is set to True, the duplicate bean check job identifies duplicates for contacts and leads associated with a person account.
- global static void enableHierarchyAccountEnrichment (boolean value)
  If this setting is set to True, any update to bean because of a hierarchy operation is cascaded to accounts also.
- global static void enableDefaultPriorityForRunAutomatic (boolean value)
  Enables the default priority for batch jobs that are flagged to run automatically.
- global static void setMasterDataResponseSize (Integer value)
  Allows you to limit the number of responses from the Hub Org to a Spoke Org.
- global static void enableQueueMonitorAutomaticRefresh (Integer value)
  Allows you to enable or disable the automatic refresh capability of the Queue Monitor.
- global static void resetExecutionManager ()
  Allows you to reset the execution manager.
- global static Map<String,String> getInternalSettings ()
  Returns values for all internal settings. If any internal setting does not have a value, then the default value is returned.
- global static void enableExtractCountryFromPicklist (boolean value)

Allows you to enable or disable extraction of Country Text from Country Picklists for Standard Address fields.

- global static void resetDefaultRecordTypesValues ()
  Resets the default record types that you explicitly set in the Default Record Type For Account and Default Record Type For Person Account internal settings.
- global static boolean resetInternalSetting (String settingName)
  Resets the value of the specified internal setting.
- global static String getInternalSettingValue (String settingName)
  Gets the value of the specified internal setting.
- global static object getConfig (String configName, List<object> params)
  Gets the settings for the specified configuration and passes a list of parameters to the configuration.

Method Documentation

global static void enableHierarchyAccountEnrichment (boolean value)

If this setting is set to True, any update to bean because of a hierarchy operation is cascaded to accounts also.

This is set to True by default.

Parameters:

value

Specify False to disable updates to the account from the beans during a hierarchy operation. If you set the value to False, the account is not updated with information from the beans in the hierarchy. In this case, run the Synchronize Reporting Data batch job to ensure there is no inconsistency between the account and the corresponding beans. You can run this job from the CC360 Queue Manager console.

global static void enableLeadPersonAccountComparison (boolean value)

If this setting is set to True, the duplicate bean check job identifies duplicates for contacts and leads asscociated with a person account.

This is set to False by default.

Parameters:

value

Specify True to check for duplicate contacts and leads associated with person accounts.

global static DS_SF_Enrichment_Settings__c getAccountEnrichmentSettings ()

Gets the enrichment settings for accounts.

Returns:

Account enrichment settings.

global static DS_SF_Enrichment_Settings__c getAccountEnrichmentSettings (String recType)

Gets the enrichment settings of the specified account record type.

Parameters:

recType

Name of the account record type for which you want to retrieve the enrichment settings.

Returns:

Enrichment settings of the specified account record type.

global static DS_SF_Synchronization_Settings__c getAccountSyncSettings ()

Gets the synchronization settings for accounts.

Returns:

Account synchronization settings.

global static DS_SF_Synchronization_Settings__c getAccountSyncSettings (String recType)

Gets the synchronization settings of the specified account record type.

Parameters:

recType

Name of the account record type for which you want to retrieve the synchronization settings.

Returns:

Synchronization settings of the specified account record type.

global static DS_SF_Synchronization_Settings__c getPersonAccountSyncSettings ()

Gets the synchronization settings of person accounts.

Returns:

Person account synchronization settings.

global static DS_SF_Synchronization_Settings__c getPersonAccountSyncSettings (String recType)

Gets the synchronization settings of the specified person account record type.

Parameters:

recType

Name of the person account record type for which you want to retrieve the synchronization settings.

Returns:

Synchronization settings of the specified person account record type.

global static DS_Settings__c getCloudMDMSettings ()

Gets the CC360 settings.

Returns:

CC360 settings.

global static DS_SF_Synchronization_Settings__c getContactSyncSettings ()

Gets the synchronization settings for contacts.

Returns:

Contact synchronization settings.

global static DS_SF_Synchronization_Settings__c getContactSyncSettings (String recType)

Gets the synchronization settings of the specified contact record type.

Parameters:

recType

Name of the contact record type for which you want to retrieve the synchronization settings.

Returns:

Synchronization settings of the specified contact record type.

global static DS_Settings__c getDataScoutSettings ()

Deprecated:

Use the getCloudMDMSettings () method instead.

global static DS_SF_Synchronization_Settings__c getLeadSyncSettings ()

Gets the synchronization settings for leads.

Returns:

Lead synchronization settings.

global static DS_SF_Synchronization_Settings__c getLeadSyncSettings(String recType)

Gets the synchronization settings of the specified lead record type.

Parameters:

recType

Name of the lead record type for which you want to retrieve the synchronization settings.

Returns:

Synchronization settings of the specified lead record type.

global static DS_Matching__c getMatchingSettings (String objectClass, String recordTypeName)

Gets the matching settings for the specified object class.

Parameters:

objectClass

The name of the object class, such as account, contact, or lead.

recordTypeName

Record type associated with the specified object class.

Returns:

Matching settings.

global static DS_Performance__c getPerformanceSettings (String jobName)

Gets the performance settings for the specified batch job.

Parameters:

jobName

The name of the batch job.

Returns:

Performance settings.

global static DS_Sources__c getSourceSettings (String sourceName)

Gets the source settings for the specified source.

Parameters:

sourcename

The name of the source for which you want to get the settings.

Returns:

Source settings.

global static DS_Sources__c getSourceSettings (String sourceName, String recTypeName)

Gets the source settings of the record type of the specified source.

Parameters:

sourceName

Name of the source to which the specified record type belongs.

recTypeName

Record type name of the specified source for which you want to retrieve the source settings.

Returns:

Source settings of the record type of the specified source.

global static DS_SF_Synchronization_Settings__c getSyncSettings (String objectName)

Gets the synchronization settings for the specified object type.

Parameters:

objectname

The name of the object, such as account, contact, or lead.

Returns:

Synchronization settings.

global static DS_SF_Synchronization_Settings__c getSyncSettings (String objectName, String recType)

Gets the synchronization settings of the record type of the specified object.

Parameters:

objectname

Name of the object to which the specified record type belongs.

recType

Record type name of the specified object for which you want to retrieve the synchronization settings.

Returns:

Synchronization settings of the record type of the specified object.

global static void updateInternalSetting (String internalSettingName, String value)

Updates an internal setting with the specified value.

Parameters:

internalSettingName

The name of the internal setting to be updated.

value

The value that will updated for the internal setting.

global static void updateInternalSetting (String internalSettingName, long value)

Updates an internal setting with the specified long value.

Parameters:

internalSettingName

The name of the setting to be updated.

value

The long value that will updated for the internal setting.

global static void enableDefaultJobPriorityForRunAutomatic (boolean value)

Enables the default priority for batch jobs that are flagged to run automatically.

Parameters:

value

Set to True to run the batch jobs automatically with default priorities that CC360 assigns.

global static void setMasterDataResponseSize (Integer value)

Allows you to limit the number of responses from the Hub Org to Spoke Orgs.

Parameters:

value

The number of responses that will be sent from the Hub Org to Spoke Orgs.

global static void enableQueueManagerAutomaticRefresh (boolean value)

Allows you to enable or disable the automatic refresh capability of the Queue Monitor.

Parameters:

value

Set to True to enable automatic refresh.

global static void resetExecutionManager ()

Allows you to reset the execution manager.

global static Map<String, String> getInternalSettings ()

Returns values for all internal settings. If any internal setting does not have a value, then the default value is returned.

global static void enableExtractCountryFromPicklist (boolean value)

Allows you to enable or disable extraction of Country Text from Country picklists for standard address fields. This should be enabled only if Country picklists have been enabled in the Org.

Parameters:

value

Set to True to enable extraction of Country Text from Country Picklists for Standard address fields.

global static void resetDefaultRecordTypesValues ()

Resets the default record types that you explicitly set in the Default Record Type For Account and Default Record Type For Person Account internal settings.

You can use the updateInternalSetting(String, String) method to explicitly set the default record types for business or person accounts.

global static boolean resetInternalSetting (String settingName)

Resets the value of the specified internal setting.

Parameters:

settingName

Name of the internal setting that you want to reset.

Returns:

True if the internal setting is reset or false if the internal setting is not reset.

global static String getInternalSettingValue(String settingName)

Gets the value of the specified internal setting. If the value of the internal setting is not set, this method returns the default value of the internal setting.

Parameters:

settingName

Name of the internal setting for which you want to retrieve the value.

global static object getConfig (String configName, List<object> params)

Gets the settings for the specified configuration and passes a list of parameters to the configuration.

Parameters:

configName

Name of the configuration for which you want to retrieve the settings.

params

List of parameters that you want to pass to the specified configuration.

Deprecated List

Member API_SettingsManager.getDataScoutSettings ()

Use the getCloudMDMSettings () method instead.