

User Defined Field
Oracle FLEXCUBE Corporate Lending 12.1.0.0.0
[April] [2016]

Part No. E74823-01





User Defined Field
[April] [2016]
Version 12.1.0.0.0

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:
Phone: +91 22 6718 3000
Fax: +91 22 6718 3001
www.oracle.com/financialservices/

Copyright ©[2005], [2016], Oracle and/or its affiliates. All rights reserved.

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. 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 failsafe, 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.

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.

This software or hardware and documentation may provide access to or information on 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. 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.



Table of Contents

1.	ABOUT THIS MANUAL.....	1-1
1.1	INTRODUCTION.....	1-1
1.1.1	<i>Organization.....</i>	<i>1-1</i>
1.1.2	<i>Conventions Used in this Manual.....</i>	<i>1-1</i>
1.1.3	<i>Related Documents.....</i>	<i>1-1</i>
2.	CREATING CUSTOM FIELDS IN ORACLE FLEXCUBE.....	2-1
2.1	INTRODUCTION.....	2-1
2.1.1	<i>Identifying a Field that you create.....</i>	<i>2-1</i>
2.1.2	<i>Defining a Numeric field.....</i>	<i>2-2</i>
2.1.3	<i>Defining a Text field.....</i>	<i>2-3</i>
2.1.4	<i>Defining a Date field.....</i>	<i>2-4</i>
2.1.5	<i>A Cube entity.....</i>	<i>2-5</i>
2.1.6	<i>Specifying the Default Value.....</i>	<i>2-5</i>
2.1.7	<i>Factory shipped fields.....</i>	<i>2-6</i>
2.1.8	<i>Indicating a procedure for populating the values of a field.....</i>	<i>2-6</i>
2.1.9	<i>Defining additional Validation Rules.....</i>	<i>2-6</i>
2.2	MAKING A FIELD APPLICABLE TO A PRODUCT.....	2-7
2.3	POPULATING THE VALUES OF UDF.....	2-8
2.3.1	<i>Population of UDF values at the contract level.....</i>	<i>2-8</i>
2.3.2	<i>Population of UDF values at the Function Id level.....</i>	<i>2-10</i>
2.3.3	<i>Example of creating a new user defined field.....</i>	<i>2-10</i>
2.3.4	<i>Using the value of the UDF for creating a customer account.....</i>	<i>2-11</i>
2.4	UPLOADING USER DEFINED FIELDS.....	2-12

1. About this Manual

1.1 Introduction

This manual is intended as a guide to help you define User Defined Fields (UDFs) and associate the UDFs with products (and therefore on contracts involving the products) and on specific Functions in FCC. Besides this User Manual, you can find answers to specific features and procedures in the Online Help, which can be invoked, by choosing 'Help Contents' from the *Help* Menu of the software. You can further obtain information specific to a particular field by placing the cursor on the relevant field and striking <F1> on the keyboard.

1.1.1 Organization

This manual contains a single chapter **Creating Custom Fields in Oracle FLEXCUBE**, which explains how UDFs can be defined and used. It details the procedure involved in defining UDFs and associating them with Products, so that the UDF is associated with contracts associated with all contracts involving the product. It also explains how to define UDFs and associating them with specific functions in FCC.

1.1.2 Conventions Used in this Manual

Important information is preceded with the  symbol.

1.1.3 Related Documents

For further information on procedures discussed in the manual, refer to the Oracle FLEXCUBE manuals on:

- Core Entities
- Core Services
- Common Procedures
- Products

2. Creating Custom fields in Oracle FLEXCUBE

2.1 Introduction

While working with Oracle FLEXCUBE, there are additional fields that you would like to use either for your convenience or to suit the requirements of your bank. Adding to its flexibility, Oracle FLEXCUBE now provides you the option to add fields based on your specifications to meet your needs.

Based on your requirement and the nature of the field, you can specify default values and validations for the field. Oracle FLEXCUBE will validate all entries made to the field against the validations you define for a field.

The screenshot displays the 'Function User Defined Fields' window. The 'Function' field is set to '100'. The 'Field Name' is 'SDFSDF' and the 'Field Description' is 'IBLC VALIDATIONS'. The 'Field Type' is 'Number' and the 'Validation Type' is 'None'. Under 'Field Attributes', 'Fixed Length' is unchecked, 'Date Format' is empty, 'Min. Length' and 'Max. Length' are empty, 'Min Value' and 'Max Value' are empty, 'Mask' is empty, and 'Cube Entity Type' is empty. Under 'Other Attributes', 'Mandatory', 'Shipped', and 'Global Field' are unchecked, 'Intermediate Field' is checked, and 'Unique' is unchecked. The 'Derivation Rule' section has 'Force Derivation' checked. The bottom table shows the user 'KRISH' who input the field on '31/10/2003 09:31:34'. The 'Mod No' section has 'Open' checked and 'Authorised' unchecked.

A field that you have created will become operational in Oracle FLEXCUBE only after it is authorized. A user bearing a different Login ID can authorize a field definition record that you have created.

2.1.1 Identifying a Field that you create

Specifying a field name and description

You can identify a field that you create with a unique identifier, and a brief description.

Each field that you define in Oracle FLEXCUBE should be assigned a unique code. You can briefly describe the field in Description field. The description is for your information only. It will not be printed on any customer correspondence.

Indicating the field type

The type of field that you can create in Oracle FLEXCUBE can be of the following formats:

- Number — Chose this option to create a Numeric field
- Text — Chose this option to create a Text field
- Date— Chose this option to create a Date field
- Cube Entity —Should you need to reuse any of the existing fields of Oracle FLEXCUBE, to enter additional details you can indicate the Field Type as Cube Entity. The Cube Entity can be any of the existing fields in Oracle FLEXCUBE like customer, currency, account, etc

Marking a field as mandatory

You can make entry to a field mandatory. To do so, check against the Mandatory option. A user of the field will be forced to make an entry to the field. Leave it unchecked to indicate that the field is not mandatory.

2.1.2 Defining a Numeric field

To define a numeric field choose the number option at the Field Type field. You can set up validation rules for a numeric field. The validation types applicable to a numeric field are:

- Range
- Length
- LOV
- None

You can indicate your preference at the Validation type field of this screen. Choose *None* to indicate that no validation should be performed on entries made to this field.

Validation Type — Range

You can specify the range validation type only for Numeric fields. In this case the entry to the field should be within a permissible range.

On choosing this validation type, you should indicate either the maximum or minimum values or both values allowed for the field. Any valid entry to the field should be within the range that you specify.

Validation Type — Length


For a numeric field you can indicate that the entry should be of a certain length. You have the option to indicate,

- A fixed length or
- The maximum and minimum length for the field

Depending on the option you select, indicate the fixed field length or indicate the maximum and minimum length for valid entries to the field.

Validation Type — LOV (List of Values)

Choose LOV to indicate that the entry to this field can be chosen from a predefined list. On choosing this option you can define the items that should be displayed on this list.

Click  and define the list of values and their description. The items that you define for the list are displayed whenever the field is used in Oracle FLEXCUBE.

2.1.3 Defining a Text field

To define a text field choose the text option at the Field Type field. A text field can contain alphabets of the English language or a combination of alphabets and numeric values.

You can specify validation rules for a text field. The validation types applicable to a text field include:

- Length
- Mask
- LOV
- None

You can indicate your preference at the Validation type field of this screen. Choose *None* to indicate that no validation should be performed on entries made to this field.

Validation Type — Length

For a text field you can indicate that a valid entry to the field should be of a certain length. You have the option to indicate,

- A fixed length or
- The maximum and minimum length for the field

Depending on the option that you select, indicate the fixed field length or indicate the maximum and minimum length of entries made to the field.

Validation Type — Mask

To indicate a field as a masked field, choose Mask as the field type. This option allows you to define a broad field structure to which all entries to the field should conform.

The mask structure can consist solely of 'a' or 'n' or a combination of these. An "a" would indicate an alphabet of the English language and "n" a numeric value.

All entries made to the field will be validated against the format that you specify for the mask.

Validation Type — LOV (List of Values)

Choose LOV to indicate that the entry to this field can be chosen from a predefined list. On choosing this option you can define the items to be displayed on this list.

Click  and define the list of values and their description. The items that you define for the list are displayed whenever the field is used in Oracle FLEXCUBE.

2.1.4 Defining a Date field

To define a date field, choose Date as the Field Type. For a date field, you can indicate validations like whether back and future dates can be entered into the field and the back or future period applicable to the field.

2.1.4.1 Validations that you can specify for date fields

Indicating the Validation type

You can specify the validation type for a date field. The validation types applicable to a date field include:

- LOV
- None
- Date format

If you indicate *LOV* then an entry to the field can be made only from the predefined list that you maintain for the field. If you are defining a date type field you have to indicate the format in which the date is to be captured. The options available are:

- YYYYMMDD
- DDMMYYYY
- DD-MON-YYYY

The format of the date entered as the value of the UDF will be validated against this format.

Choose *None* to indicate that no validation should be made for the field.

2.1.5 **A Cube entity**

If the field that you are creating is of the type Cube Entity, indicate the entity that is applicable to the field. You can select a Cube Entity from the option list positioned next to the Cube Entity Type field. This contains the following Cube Entities:

- Currency
- GL
- Branch
- Customer
- Account
- Contract Ref No
- User Ref No
- Liability ID

For instance,

You need to capture details of an additional customer for a transaction. You can choose the Cube Entity field type and select the Customer field. The list of customers maintained for your bank will be made available at the field as in any other Customer field in Oracle FLEXCUBE.



At the time of installation, the implementer will write the PL/SQL code for the derivation rule, validation rule and also the query for fetching a cube entity according to the requirements of the bank.


2.1.6 **Specifying the Default Value**

You can specify a default value for a user-defined field. The option list positioned next to this field will be enabled only if the field is a cube entity. Else the option list will be disabled.

Example

Assume that your bank wants to capture an additional currency field for processing certain transactions and the default value for this additional field has to be USD. Therefore, your selections will be as follows:

- Field Type –Cube Entity
- Cube Entity Type- Currency

Click on  in the 'Default Value' field. All the currencies maintained at your bank will be displayed. Select USD. Consequently, in the UDF screen of the Contract Input or Function Id screen, USD will be displayed as the default currency.



If you have specified a default value for a field and also the derivation rule, the value obtained from the derivation rule will take precedence over the default value at the Contract/Function Id level.

2.1.7 Factory shipped fields


Apart from the fields that you have created, there are a set of fields that are sent by default as part of Oracle FLEXCUBE. These fields are referred to as factory shipped fields. You cannot define or change validations specified for factory shipped fields.

2.1.8 Indicating a procedure for populating the values of a field

When you are processing a transaction that would use the user defined field, the value of the UDF can be populated in the 'User Defined Field/Field Name to Value Definition' screen that is invoked from the Contract Input screen or the Function Id screen. The value for the field is populated based on certain conditions, which have to be defined as statements of code. Check against the Force Derivation field to specify that the value of a field has to be populated based on certain conditions.

2.1.9 Defining additional Validation Rules

Apart from specifying the validation type for a field, you can specify additional validation rules to meet the requirements of the bank. To indicate that additional validations are required for a particular user defined field, check against 'Validation Allowed'.

This will allow you to write a code to validate the value in the new field. Click on  to write the validation code.

WIN_RULE

Product Category: PRIN Field Number: 1

Validation Rule	Condition	Return Type	Result	Err Msg
1	@value = 0	TRUE	Error	Check digit is wrong

During contract processing the system validates the check-digit against each of these validations.

Example


At the time of processing a contract for a customer, Grindlay's Bank wants to add a new field to enter the first nominee that the customer has specified in his account. The names of the nominees of a customer are maintained in the 'Customer Accounts Maintenance' screen. The bank wants the system to check that the name of the nominee that is entered in the new field and the name of the first nominee that is maintained in the Customer Accounts Maintenance screen are same.

To do this, you need to:

- Create a new user defined field of type 'Text',
- Select Length as the 'Validation Type', and
- Write a code to validate the value entered in the new field. In the validation rule, you write a code to check that the value of the first nominee specified in the Customer Accounts Maintenance screen for the customer for whom you are processing a contract and the value entered at the time of processing a contract are same.

2.2 Making a field applicable to a product

The fields that you define in the User Defined Fields screen can be made applicable to the products (and thereby to the contracts) that you create for the front-end modules of Oracle FLEXCUBE depending on your selection in the Function ID field.

At the time of creating a product, click  from the Product Definition screen. From this list of all the fields that you created, you can choose specific fields and make them applicable to the product.

SEQ No	Field Name
1	COLLATERAL
2	GOODY1
3	GOODY2
4	PRODTEST

In this screen, you can link the user-defined fields to the product that you are defining. Oracle FLEXCUBE allows you to link up to fifty fields to a product.

To link a user-defined field, click on . A list of all the user-defined fields for which the Usage Allowed is 'Product' will be displayed. Select the fields that have to be included in the Contract Input screen for the contracts processed under the product that is being defined.

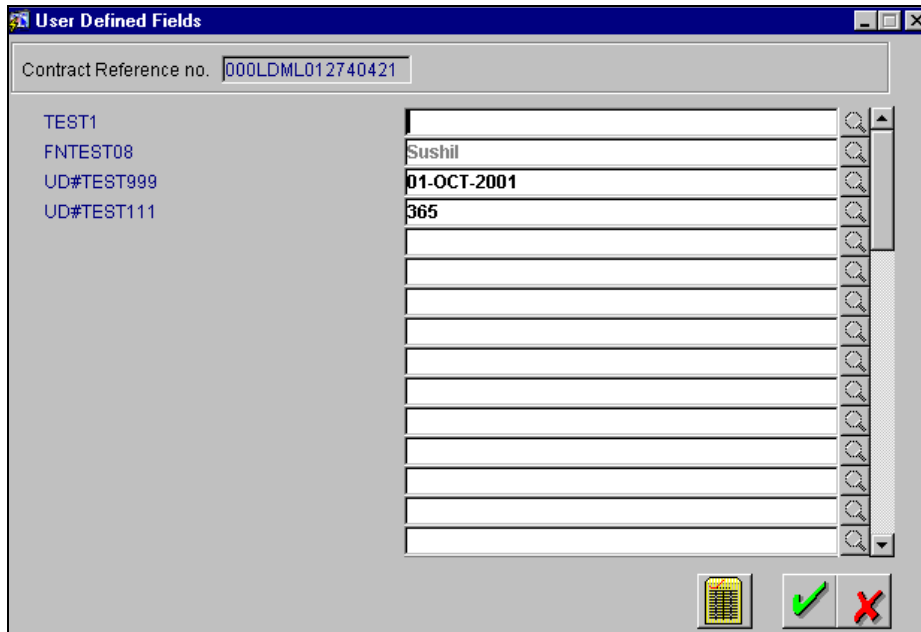
You can also choose not to specify values for a UDF, at the product level. But, when a contract is processed under that product, the fields associated with the product will be available in the Contract Input screen and you will have to capture the value of the UDF on a mandatory basis. If you choose to modify the value associated with the UDF, the system will display the old as well as the new values during authorization.

2.3 Populating the values of UDF

The defaulted or derived values of the UDF linked at the product level will be populated in the Contract Input screen at the time of processing contracts under a particular product. Similarly, the values of the UDF's linked to a function Id will be populated in the function Id screen.

2.3.1 Population of UDF values at the contract level

The UDF's linked to a particular product will be available when a contract is processed under that product. In the Contract Input screen, click on to invoke the 'User Defined Fields' screen.



In this screen, all the user-defined fields associated with the product under which you are processing the contract will be displayed.

According to your specifications at the time of defining the user-defined fields, system may derive the values of the UDF from:

- Derivation Logic

When you click on **F..**, system derives and displays the values of the fields for which derivation rules are written.

- Static List of Values

If the validation type of the UDF that is associated with the contract is 'List of Values' (LoV), system will display the static values. If you have indicated a default value for the LoV, the default value will be displayed.

- Method of derivation for cube entity fields

If a UDF is of type Cube Entity, system populates the values in a LoV. The values will be derived from the specified Oracle FLEXCUBE table (if the cube entity type is 'Static') or from the query written by the user (if the cube entity type is 'Dynamic')

You will not be allowed to change the values of the fields for which you have not allowed amendment (if the option 'Amendable' is unchecked at the time of defining an UDF) and derivation is allowed (Option 'Derivation Allowed' is checked).

Enter the values for the fields, for which the values are not derived by the system.


You can change the value of a UDF after the system has derived the value from the derivation logic. But it might so happen that the derivation logic of another UDF might use the value of the UDF that you have changed. Consequently, if you change the value of the UDF whose value is used in another UDF, the value of that UDF will also change.

For example


Let us assume that you have defined a user-defined field UDF1 to display the Euro equivalent of the contract currency.

The derivation logic of another user-defined field (UDF2) utilizes the values of UDF1 to arrive at the value of UDF2. Assume that in the derivation logic written for UDF2, you have specified that the value of UDF2= 2 UDF1.

At the contract level, system converts the contract currency into Euro equivalent and displays the value of UDF1, as 220.00. According to this value of UDF2 will be 444.00. However, if you change the value of UDF1 to 250.00, it will effect on the value of UDF2 also (it becomes 500.00).


After changing the value of the UDF, if you want to see if it has effected the values of any other UDF's, click on . If the change in the UDF value has effected any other UDF, system will display an override informing you about the UDF whose value will be changed. Select 'Ok' if you want the system to re-calculate the value of the UDF based on the modified UDF value.

2.3.2 Population of UDF values at the Function Id level

The values of the UDF will be populated in the 'Field Name to Value Definition' screen of the Function Id screen. In any of the Function Id screens (Maintenance screens), click on  to invoke the 'Field Name to Value Definition' screen.


In this screen, all the user-defined fields associated with the Function Id of that particular screen will be displayed.

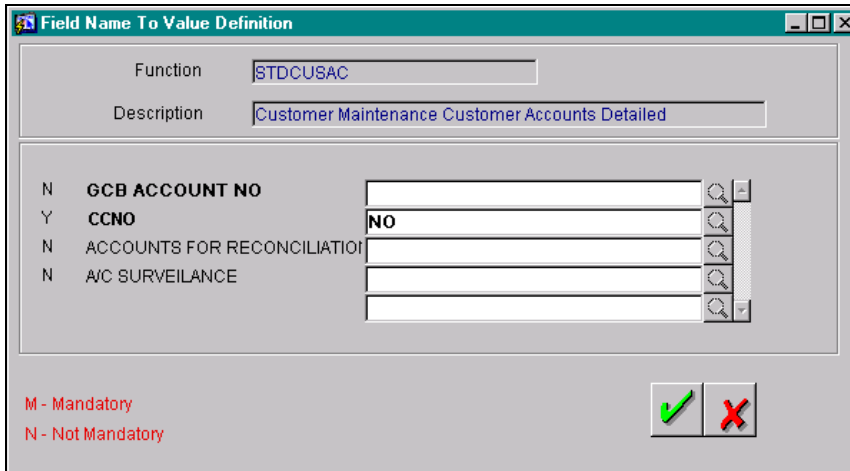
The method by which the system derives the UDF values is similar to the population of UDF values at the contract level, which has been discussed in the previous section.

You can change the values of the UDF's derived by the system. Click on  to view if the change in the UDF value has effected the value of any other UDF.

2.3.3 Example of creating a new user defined field

To include the value of a user defined field (UDF) in the customer account number, you need to create a user defined field for the function Id STDCIF. Function Id STDCIF indicates the 'Customer Information Maintenance' screen. Since the value of the UDF will be included in an account number, it is mandatory for you enter the value of this UDF (value of the UDF is entered at the time of creating a customer). Therefore, remember to check against 'Mandatory' while defining the user defined field. If you choose this option, system ensures that you enter a value for the UDF.

While capturing the details of a new customer in the 'Customer Information Maintenance' screen, click on  to invoke the 'Field Name to Value Definition' screen. In this screen, you will notice that the user defined fields that you had linked to the function Id STDCIF will be displayed. Enter the value for the user defined field, which has to be included in the customer account (since it is mandatory). This value will be displayed at the time of defining a new customer account on selection of the customer.



2.3.4 Using the value of the UDF for creating a customer account

You might have linked several user defined fields to the function Id STDCIF. Out of these, you want the value of a particular UDF to be included in the account number. Therefore, your bank should inform the implementer of Oracle FLEXCUBE, the UDF whose value has to be picked up and included in the customer account number.


The implementer will indicate the UDF whose value has to be picked up and included in the account number at the back end. Consequently, at the time of creating new customer accounts, Oracle FLEXCUBE automatically picks up the value of that particular user defined field. This will be displayed on selection of the customer.


Example

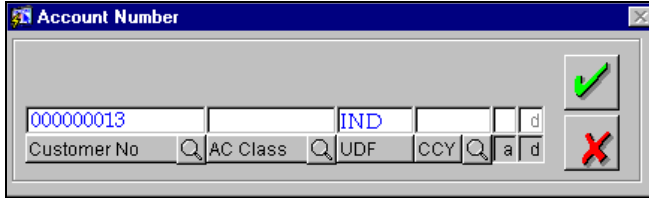
Let us assume that your bank wants to include the value of the UDF CUST_COUNTRY_CODE created to include the country code of the customers in customer account numbers. Therefore, your bank has to inform the implementer that the value of the UDF CUST_COUNTRY_CODE has to be included in the account number. The implementer will indicate at the back end that the value of the user defined field CUST_COUNTRY_CODE should be included in the customer account number.

Let us assume that you have entered the value IND for the UDF CUST_COUNTRY_CODE for the customer Ms.Tanya Agnihotri (CIF Id 000000013).

At the time of creating a new account for Ms.Tanya Agnihotri, system will automatically display the value IND on selection of CIF Id 000000013.

In Oracle FLEXCUBE, customer accounts are defined in the 'Customer Accounts Maintenance' screen. To create a new account for a customer, click on . You will be prompted to select the account number according to the account mask.

If you have specified the account mask as CCCCCCLLLLLUUU\$\$\$d, system will allow you to select the customer, account class, currency. Click on  to select these parameters. When you select the customer, system automatically picks up and displays the value of the UDF maintained for that customer.



2.4 Uploading user defined fields

Your bank may use Oracle FLEXCUBE to process certain transactions (say LD and FT transactions) and use another application to process other transactions (say MM and FX transactions).

At some point of time, your bank may want to use the values from the external system in Oracle FLEXCUBE. For this purpose, Oracle FLEXCUBE stores the values of the various fields from the external system (an application other than Oracle FLEXCUBE) in a database table. If your bank wants to use the value of the fields from the external system in Oracle FLEXCUBE, the values have to be brought into Oracle FLEXCUBE database. This is done through a process called 'Upload'.

2.4.1.1 Uploading a list of pre-defined values for the User Defined Fields

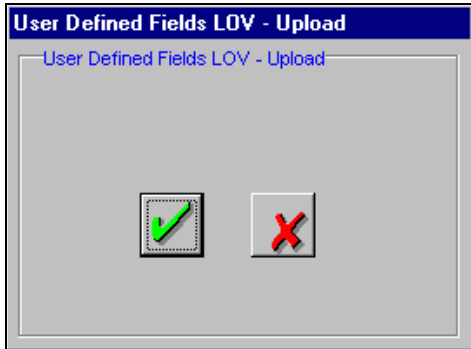
In Oracle FLEXCUBE, you have the option of uploading the list of values, which, should be populated for User Defined Fields. The list of values that get uploaded will either have drop down or option lists positioned next to them. For instance options like Check Lost, Check Damaged may be uploaded for the Reason Code field in the Stop Payments screen.

Prior to executing the upload process you will have to upload the list of values from a predefined oracle table in which you will maintain the required data in the following manner:

Field	Description
LOV_NAME	This is the name of the option/drop down list for the specified UDF.
LANGUAGE_CODE	This is the language in which the list of values has been maintained.
KEY_VALUE	This is the name of one of the allowed value.
VALUE_DESC	This is the description of the Allowed value.
UPLOAD_STATUS	This indicates whether the record has been successfully uploaded or whether it is yet to be uploaded. The mandatory value for the upload to

Field	Description
	be successful is U.

After having maintained the required details you can trigger the upload process using the User Defined Fields LOV Upload screen available in the Application Browser.



The upload process will not be successful under the following conditions:

- If the LOV_NAME, KEY_VALUE, and LANGUAGE_CODE combination is duplicate.
- If you have failed to maintain the Language Code specified in the oracle table as a valid value in FCC.

Also note that the exceptions encountered during the upload process will be recorded in the Exception Log table.