Oracle FLEXCUBE Direct Banking

Development Workbench for Direct and Mobile Banking User Manual
Release 12.0.2.0.0

Part No. E50108-01

September 2013

Development Workbench for Direct and Mobile Banking User Manual
September 2013

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 © 2008, 2013, 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.

Contents

1. Preface

1.1. Intended Audience

This document intended for the following audience:

- Customers
- Partners

1.2. Documentation Accessibility

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

1.3. Access to OFSS Support

https://flexsupp.oracle.com/

1.4. Structure

This manual is organized into the following categories:

Preface gives information on the intended audience. It also describes the overall structure of the User Manual

Overview provides brief information on the overall functionality covered in the User Manual

Chapters post overview are dedicated to individual transactions and its details, covered in the User Manual

1.5. Related Information Sources

For more information on Oracle FLEXCUBE Direct Banking Release 12.0.2.0.0, refer to the following documents:

- Oracle FLEXCUBE Direct Banking Licensing Guide
- Oracle FLEXCUBE Direct Banking Installation Manuals.

2. Overview

This Tool is a Swing based User Interface for designing screen based on Light Weight Extensions and Application Programming (LEAP) framework. Earlier user used to make the database entries for creating the new screens in LEAP Framework. Now this workbench gives a platform to user to design a new screen and modify existing screens without directly accessing the database tables. User can preview the SQL scripts using this tool.

3. Prerequisites

Following are the prerequisites to run this tool

1. To use this tool one should have “JDK 6” or higher version of java on their machine.
2. Current version of LEAP.jar
3. Following four jars are also required to be copied in the following sub-directories of the main directory where LEAP.jar is present.

    1. ext/lib/kernel
        - FCDB_kernel.jar
    2. ext/lib
        - xalan.jar
        - xercesImpl.jar
        - serializer.jar
        - log4j-1.2.17.jar
        - xml-apis.jar
        - yuicompressor-2.4.7.jar
        - commons-codec.jar
        - ojdbc6.jar
        - servlet-api.jar

4. How to Run this Tool

This tool can be run in following two ways:

To run this tool double click on LEAP.jar, if JDK 6 is available in the recommended programs to execute jar files.

Or

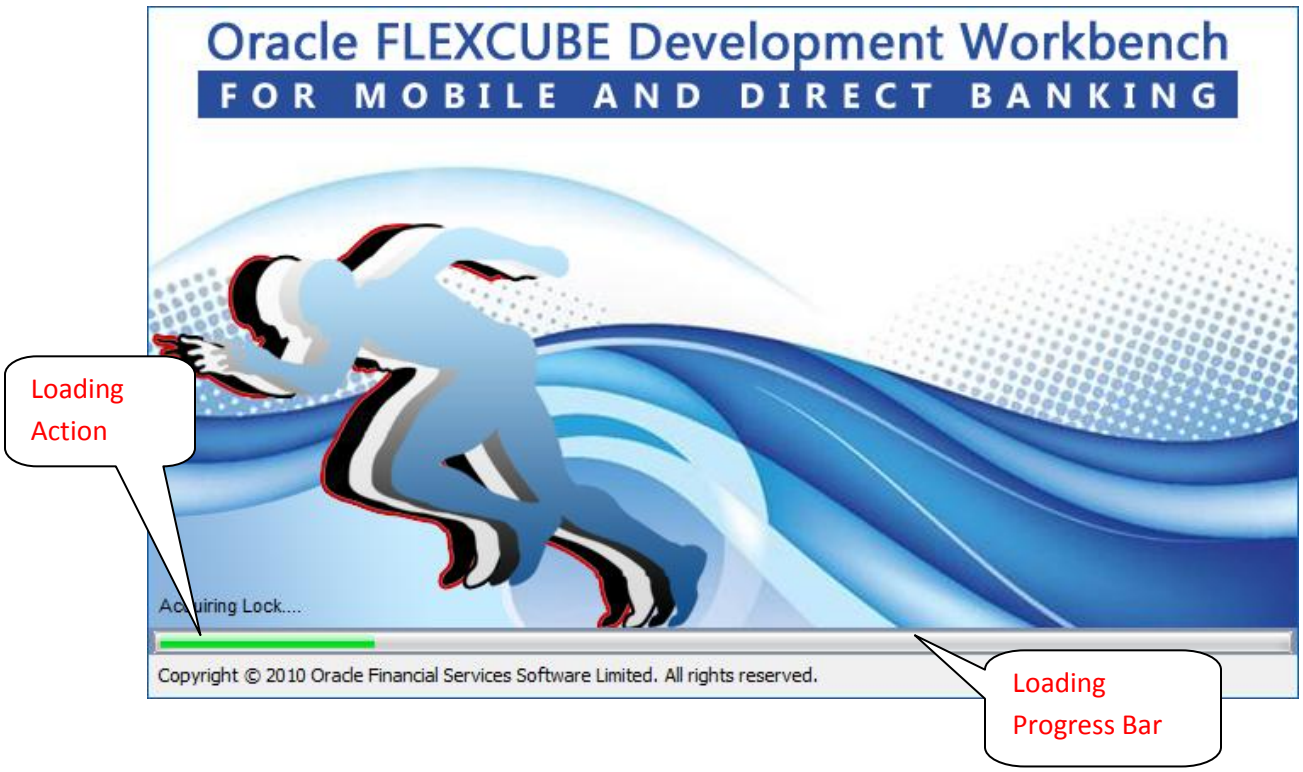
Use command java –jar LEAP.jar to execute the jar file.

In order to increase heap memory size for leap tool use the following command

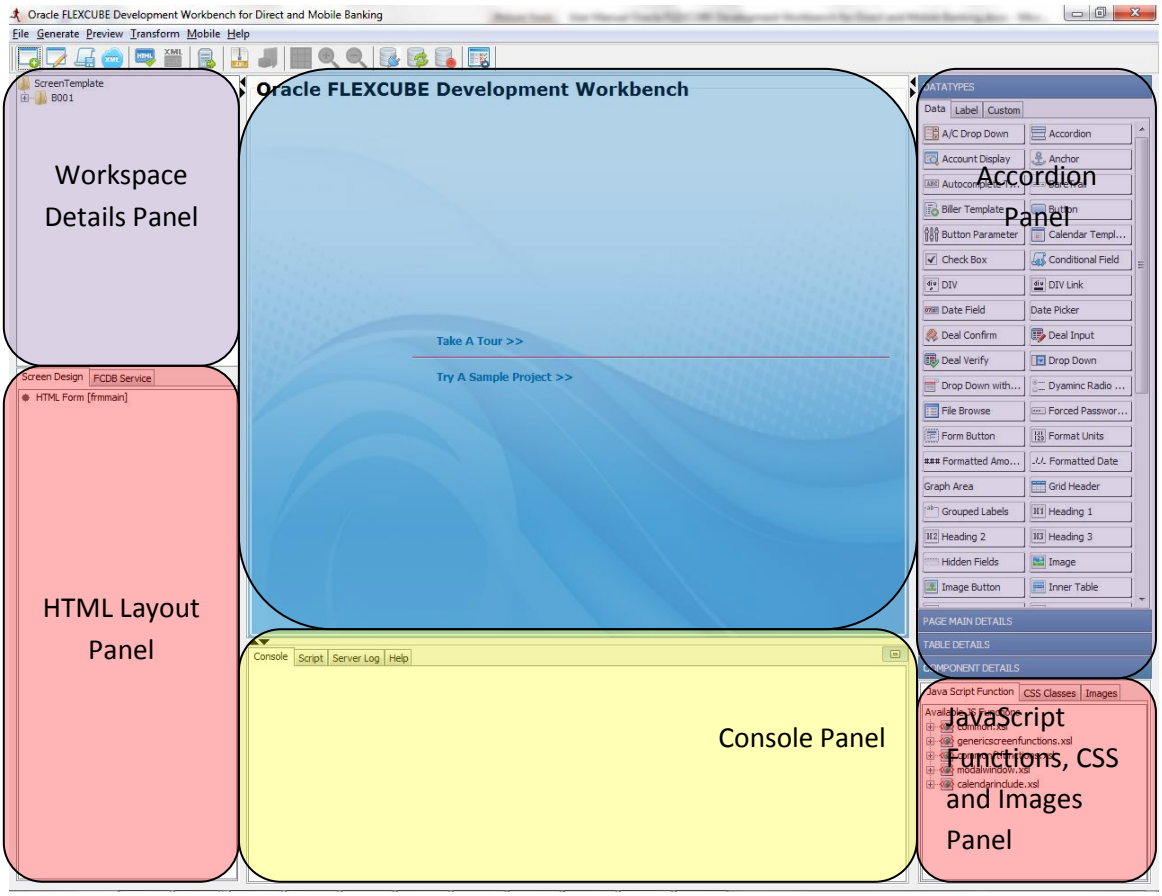
Java –Xms512m –jar LEAP.jar

5. Understanding this Tool

On executing LEAP.jar a splash screen opens with a progress bar in the bottom as shown below,



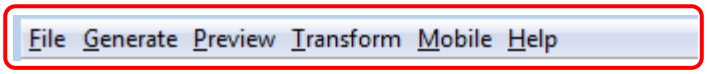
Once the progress bar completes, splash screen goes off the screen and the main working platform appears. The following screenshot displays the main working platform along with the description of the different panels.



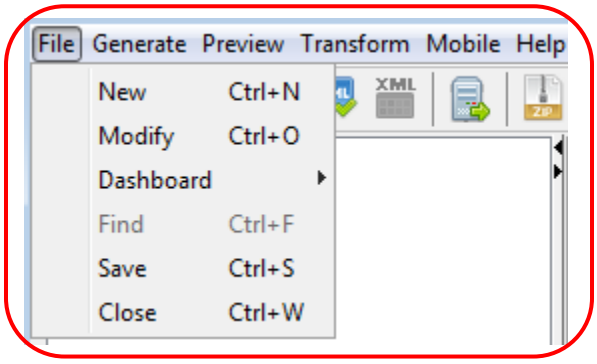
In the following section, details of each of the highlighted components of this tool is discussed.

6. Menu Bar

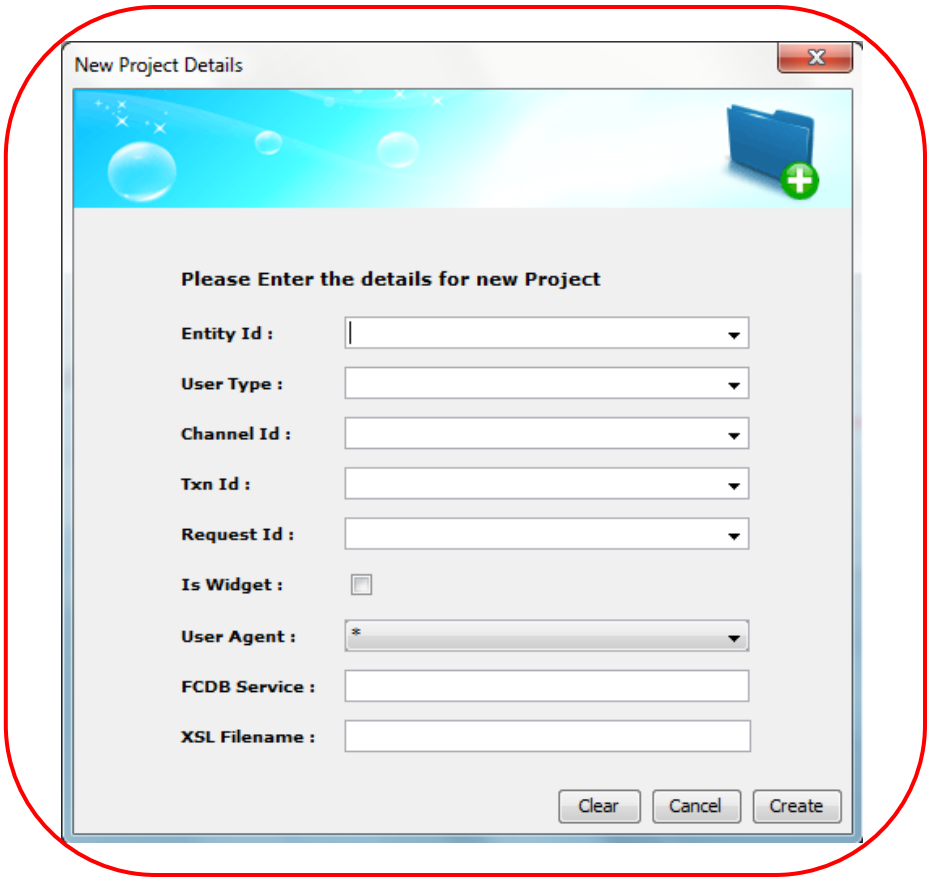
Menu Bar of this tool has following five menus



1. File: File menu has following five sub-menus

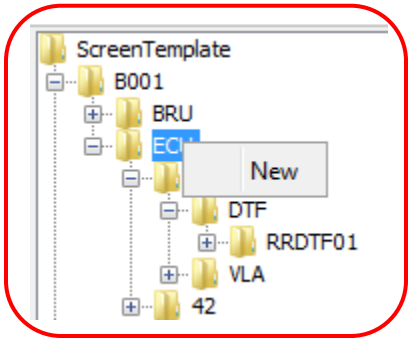


   a. New : This sub-menu allows a user to create a new screentemplate project in the workspace of the tool. Clicking on "New" opens a dialog window as shown below

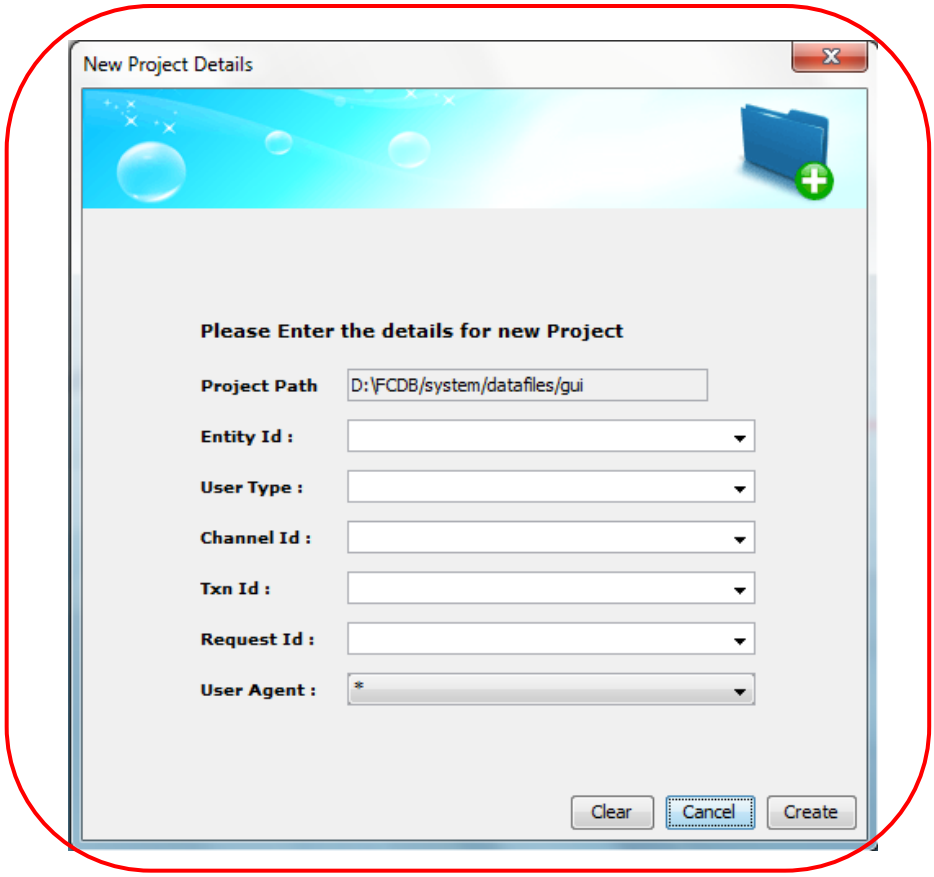


User needs to enter the entity id, usertype, channel id, transaction id, request id, user agent, fcdb service and XSL file name of the screen to be designed. Click on "Create" to create a new project in the workspace. Click on "Cancel" to close this dialog. Click on "Clear" to clear the input fields.

A new screentemplate project can also be created by right clicking on the file tree node and selecting "New" option as shown below:

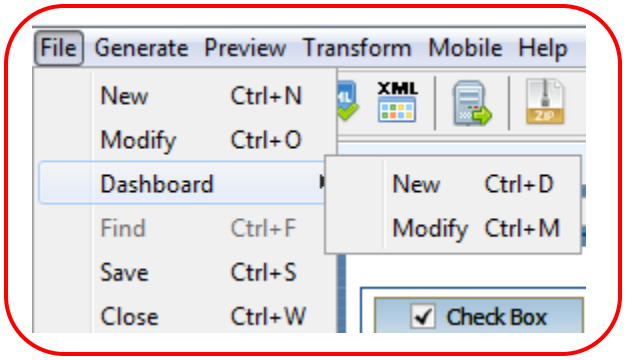


b. Modify: This sub-menu allows a user to modify the existing screens in FCDB application. Using this sub-menu user can load the screentemplatexml from FCDB datafiles folder into the workspace of this tool and modify the screen design. Clicking on "Modify" opens a dialog window as shown below.



Project Path consists of the default FCDB home folder path where gui/services are kept. User needs to enter the entity id, usertype, channel id, transaction id, request id, user agent and FCDB service name of the screen to be modified. Click on "Create", on the basis of the input field's tool will copy the screen template xml and js file from FCDB project into its own workspace. Click on "Cancel" to close this dialog. Click on "Clear" to clear the input fields.

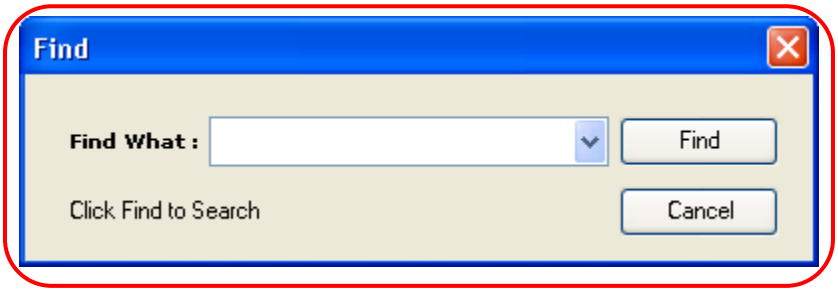
c. Dashboard: This sub-menu allows user to create/modify Dashboard Projects.



i. New: Clicking this sub-menu, user can create a new Dashboard Project.

ii. Modify: Clicking this sub_menu, user can modify an existing Dashboard Project.

d. Find: This sub-menu allows user to find any words in the JavaScript file loaded in the JavaScript editor panel. Clicking on "Find" opens a dialog window as shown below.

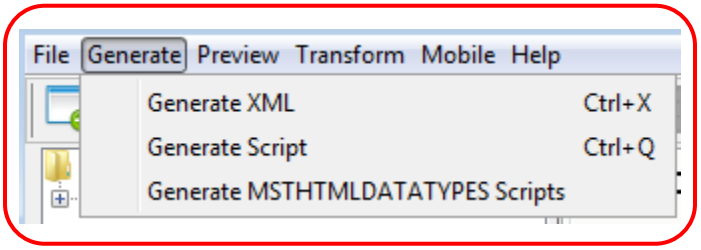


Enter a text and click on "Find" to find the text. Click on "Cancel" to close this dialog window.

e. Save: Click on save to save the idrequest.js file.

f. Close: Click on "Close" to close the tool.

2. Generate: Generate menu has following three sub-menus.

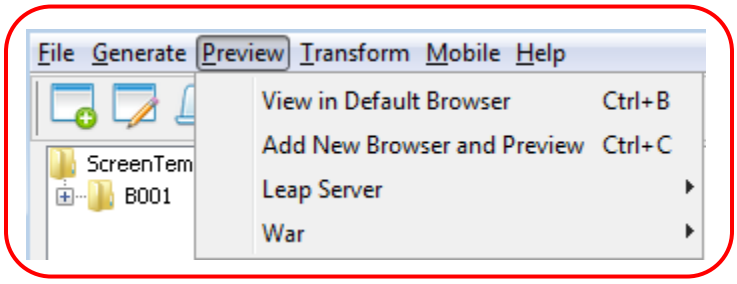


   a. Generate XML: This sub-menu allows user to regenerate the screentemplate xml after changing the screen design. It can be considered as an action to save the screen design.

   b. Generate Script: This menu allows user to generate the complete insert script of the screen design. Clicking this menu will clear all the update/insert/delete script corresponding to an idrequest and provide the user with a fresh complete insert script corresponding to that idrequest.

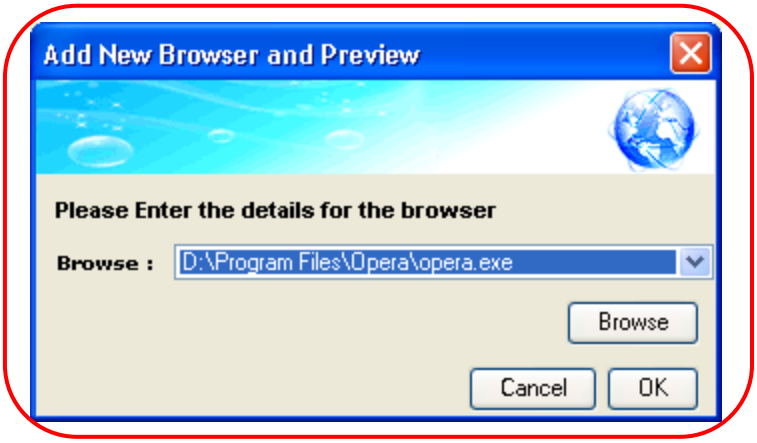
   c. Generate MSTHTMLDATATYPES Scripts: This sub-menu allows user to generate the MSTHTMLDATATYPES insert script of the screen design. Clicking this menu will generate script along with Html Data Type Id and its description.

3. Preview: Preview menu has following two sub-menus.



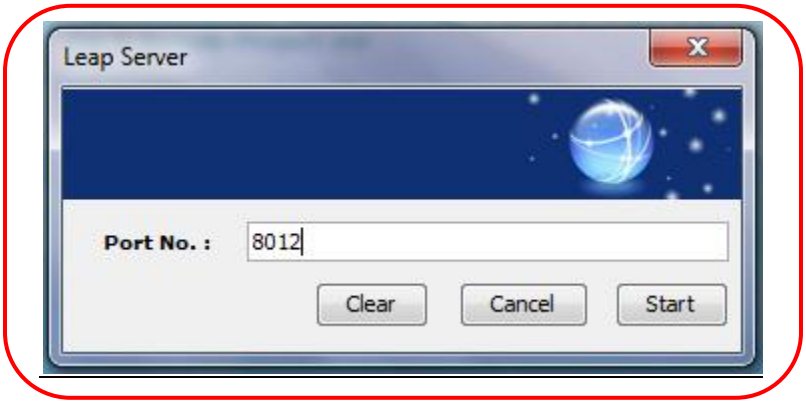
   a. View in Default Browser: Clicking this sub-menu will launch the system's default browser with the output html of the current screen design project. User can preview the screen design done by them in this output html.

   b. Add New Browser and Preview: This menu allows a user to launch a selected browser. Clicking this menu will opens a dialog window as shown below.



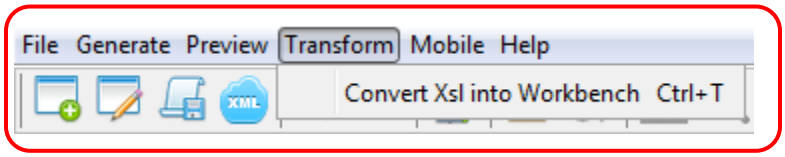
Click on "Browse" to browse the browser path, Click "OK" to launch the output HTML in selected browser, Click "Cancel" to close this dialog.

c. Leap Server: This menu allows the user to start/stop leap server. On clicking start the following dialog pops up,

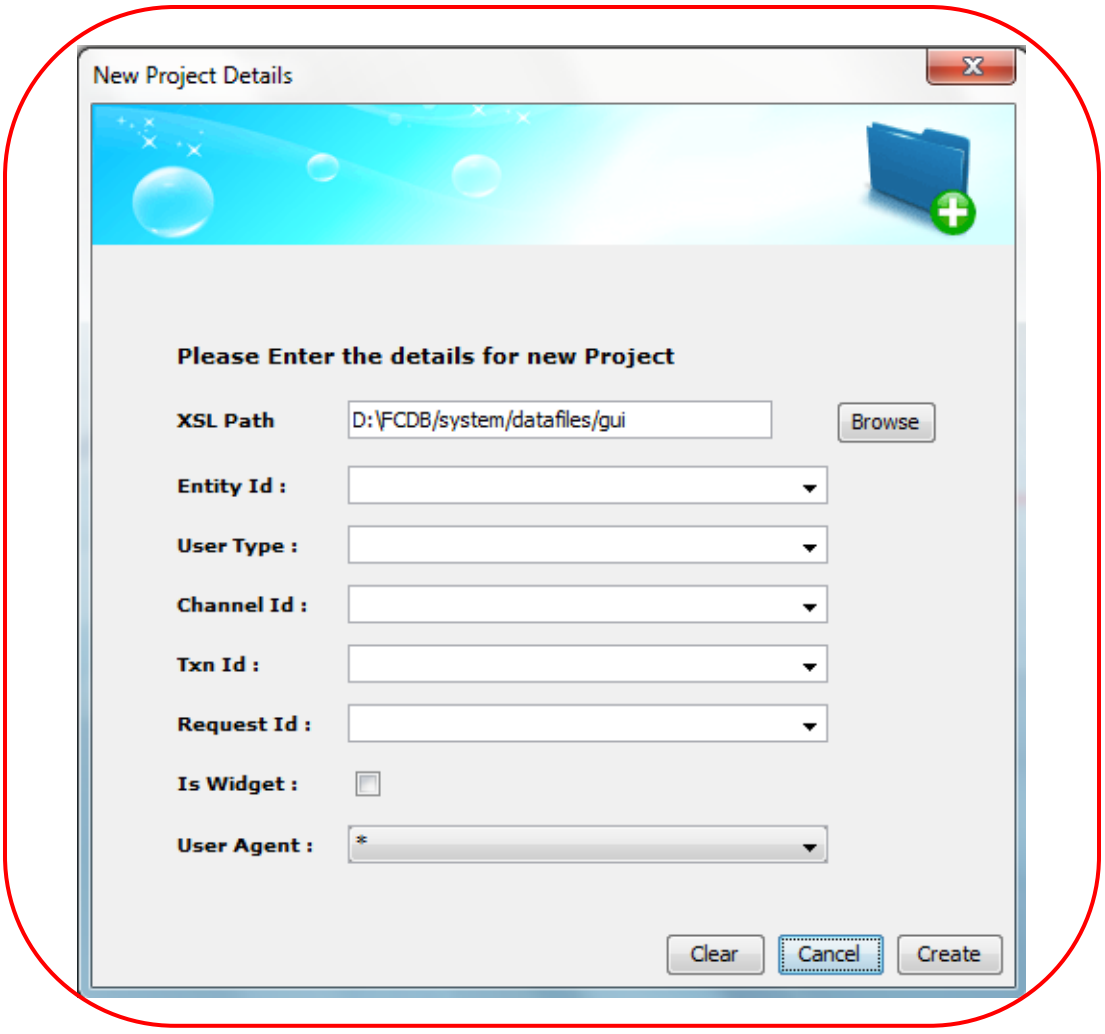


Enter the port number of the server. Click on "Start" to start the leap server. Click "Cancel" to close this dialog. Click on "Clear" to clear port no.

4. Transform: Transform menu has following sub-menu.



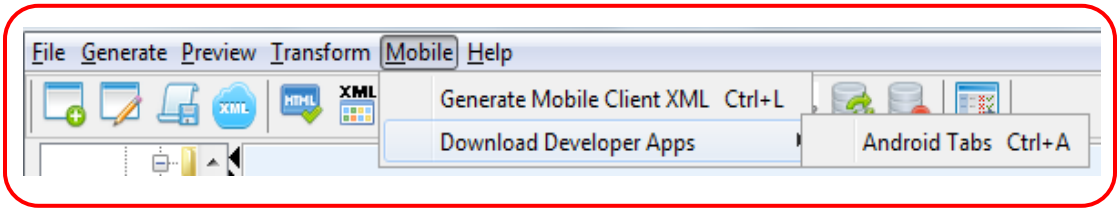
a. Convert Xsl into Workbench: This sub-menu allows a user to convert existing FCDB screen which has been designed using XSL file to Channel Workbench framework. Clicking this sub-menu opens a dialog window as shown below.



User needs to browse the XSL file path, enter the entity id, usertype, channel id, transaction id, request id and user agent of the screen to be converted to Channel Workbench framework. Click on “Create”, on the basis of the input field’s tool will convert the XSL file into screentemplate xml and js file and load these files into its own workspace. Click on “Cancel” to close this dialog. Click on “Clear” to clear the input fields.

While transforming XSL to Channel Workbench, tool will generate some self-explanatory messages in the output log file available in the directory **logs** as **LEAPOut.log**.

5. Mobile: This Menu Bar item is enables only for Mobile Application Channel (Channel 43).



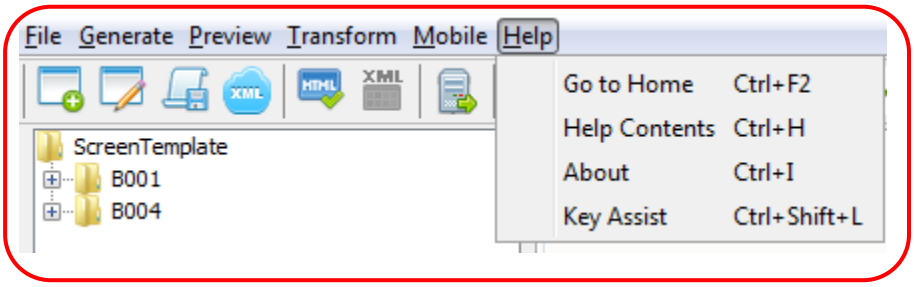
It Consists of the following two options:

   a. Generate Mobile Client XML: Clicking this sub- menu generates the mobile client XML.

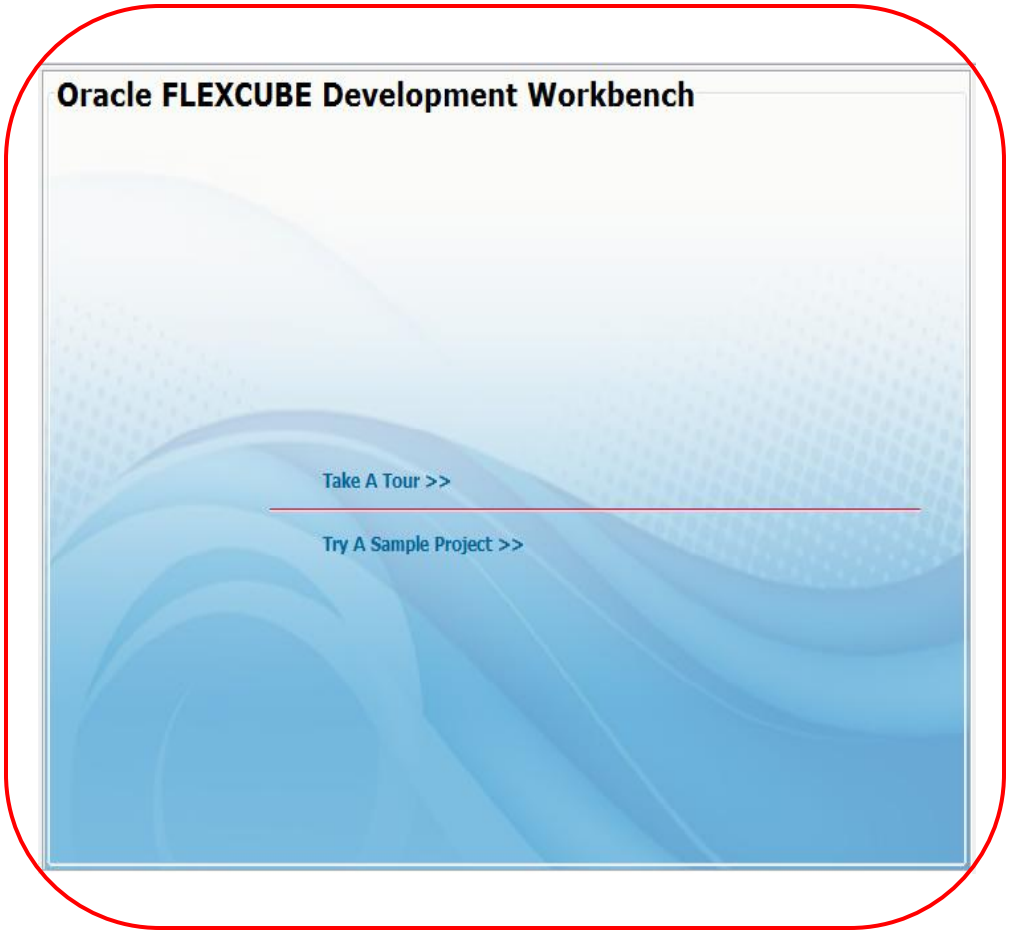
   b. Download Developer Apps: Clicking on the submenu opens another submenu containing application of various user agents.

      i. Android Tabs: Clicking this submenu downloads the FCDB application (.apk file) supported by Android Tablets.

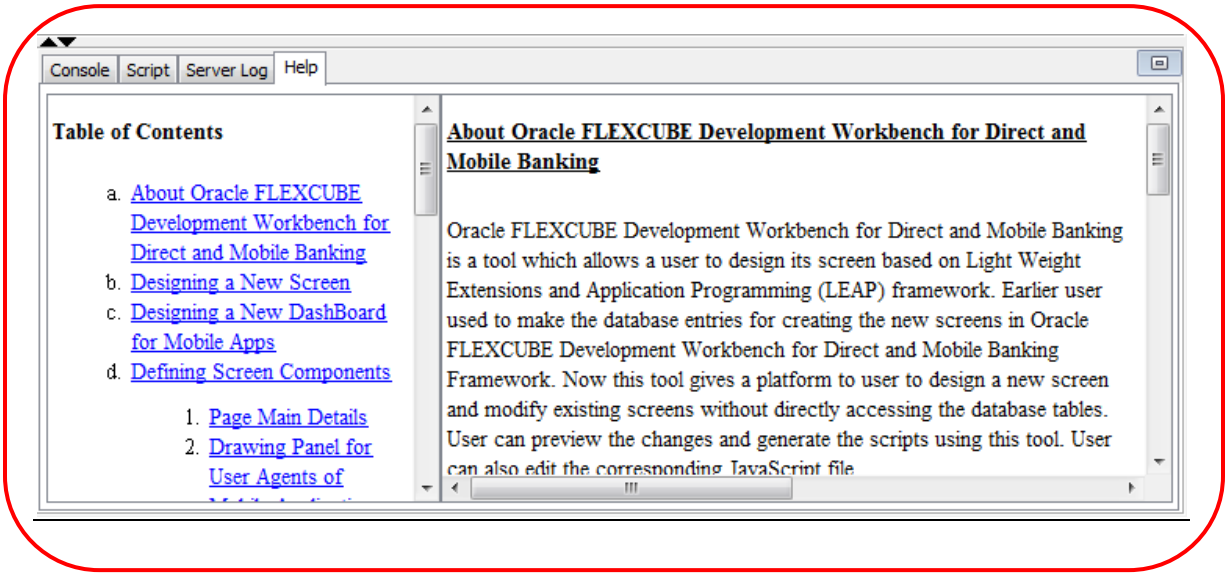
6. Help: Help menu has following four sub-menus.



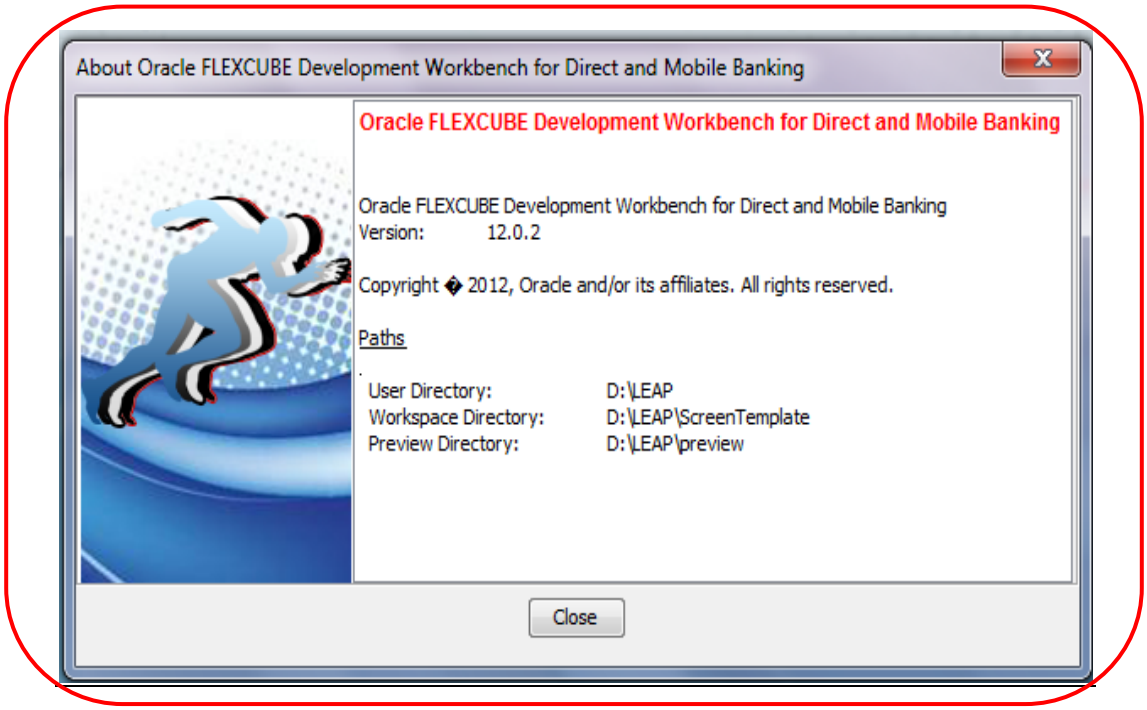
   a. Go To Home: Clicking this sub- menu opens the welcome panel in the main work area panel, as shown below.



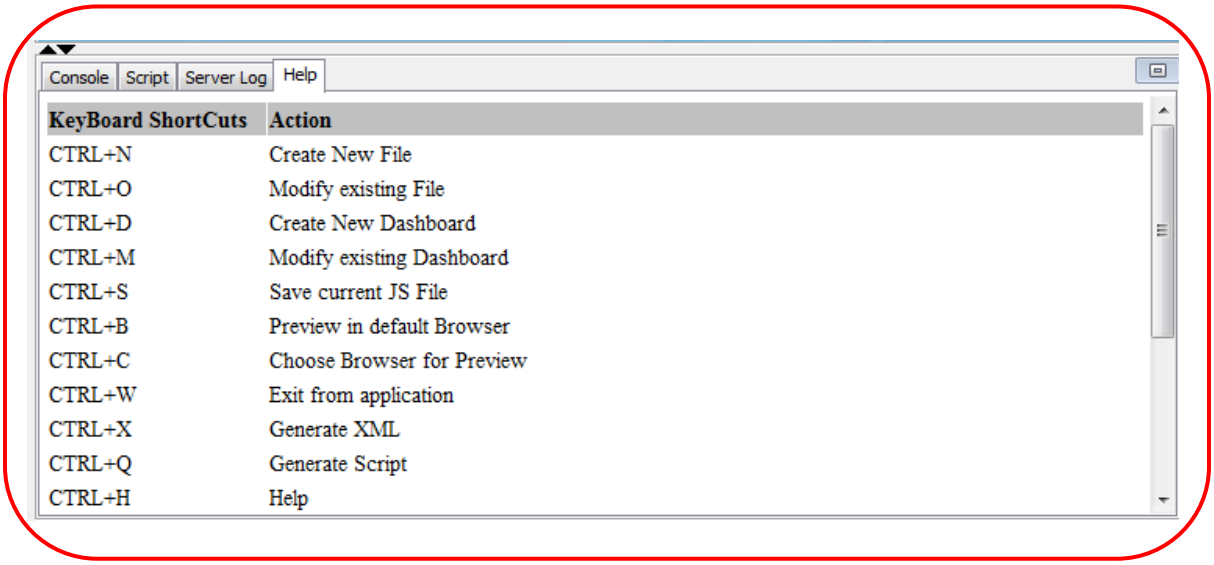
b. Help Contents: Clicking this sub-menu opens the complete HTML help for this tool in the bottom console pane under the help tab.



c. About: Clicking this sub-menu opens following dialog window which contains the details associated with the tool version and working directories.

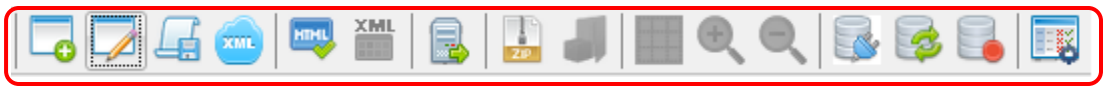


d. Key Assist: Clicking this sub-menu opens the keyboard shortcuts HTML page in the bottom console pane under the help tab.



7. Tool Bar

Tool Bar of this tool has the following icons.



a.  New: This icon allows a user to create a new screentemplate project in the workspace of the tool.

b.  Modify**:** This icon allows a user to modify the existing screens in FCDB application. Using this icon user can load the screentemplate xml from FCDB datafiles folder into the workspace of this tool and modify the screen design.

c.  Save: Click on this icon to save the idrequest.js file.

d.  Generate XML**:** This icon allows user to regenerate the screentemplate xml after changing the screen design. It can be considered as an action to save the screen design.

e.  Preview**:** This icon allow user to preview the complete screen layout at any point of change done by them. Previewing launches the browser with the output html. One can preview either in system default browser or select the alternative browser .exe path and launch the alternative browser. Previewing automatically saves your screen definition.

f.  Generate Mobile Client XML**:** This icon allows user to generate mobile client XML for channel id 43.

g.  Server Start/Stop: This icon allows the user to start/stop leap server.

h.  Synchronize war : This icon allows user to synchronize latest war in to the workplace.

i.  Package war**:** This icon allows user to repackage the war with updated CSS and image files. Package War will be enabled only if any Web component like CSS or images has been edited by the user.

j.  Show Canvas: This icon shows Drawing Canvas for user Agents of Channel 43 or Wireframe Canvas for channels other than Mobile Channel.

k.  Zoom in**:** This icon allows user to zoom into the canvas

l.  Zoom Out: This icon allows user zoom out of the canvas.

m. Database Connectivity**:**

i.  Inactive Connection: This icon depicts that tool is not connected to the database. Click on this icon to configure database connection.

ii.  Active Connection: This Icon signifies that you are connected to Database. Click this icon to modify database connection.

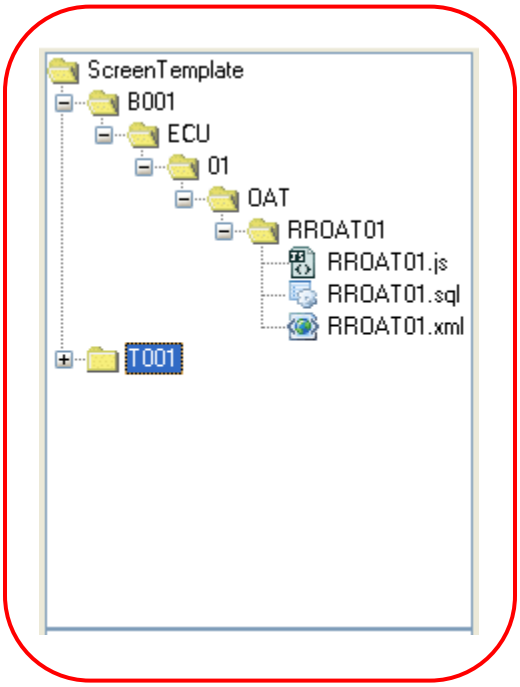
iii.  Database Refresh: Click this icon to refresh the database connection and reload the changes done in database.

iv.  Disconnect Database Connection: Click this icon to close the database connection.

e.  FCDB Setup Details**:** FCDB Properties dialog will allow user to enter the path of the FCDB folder and the language.

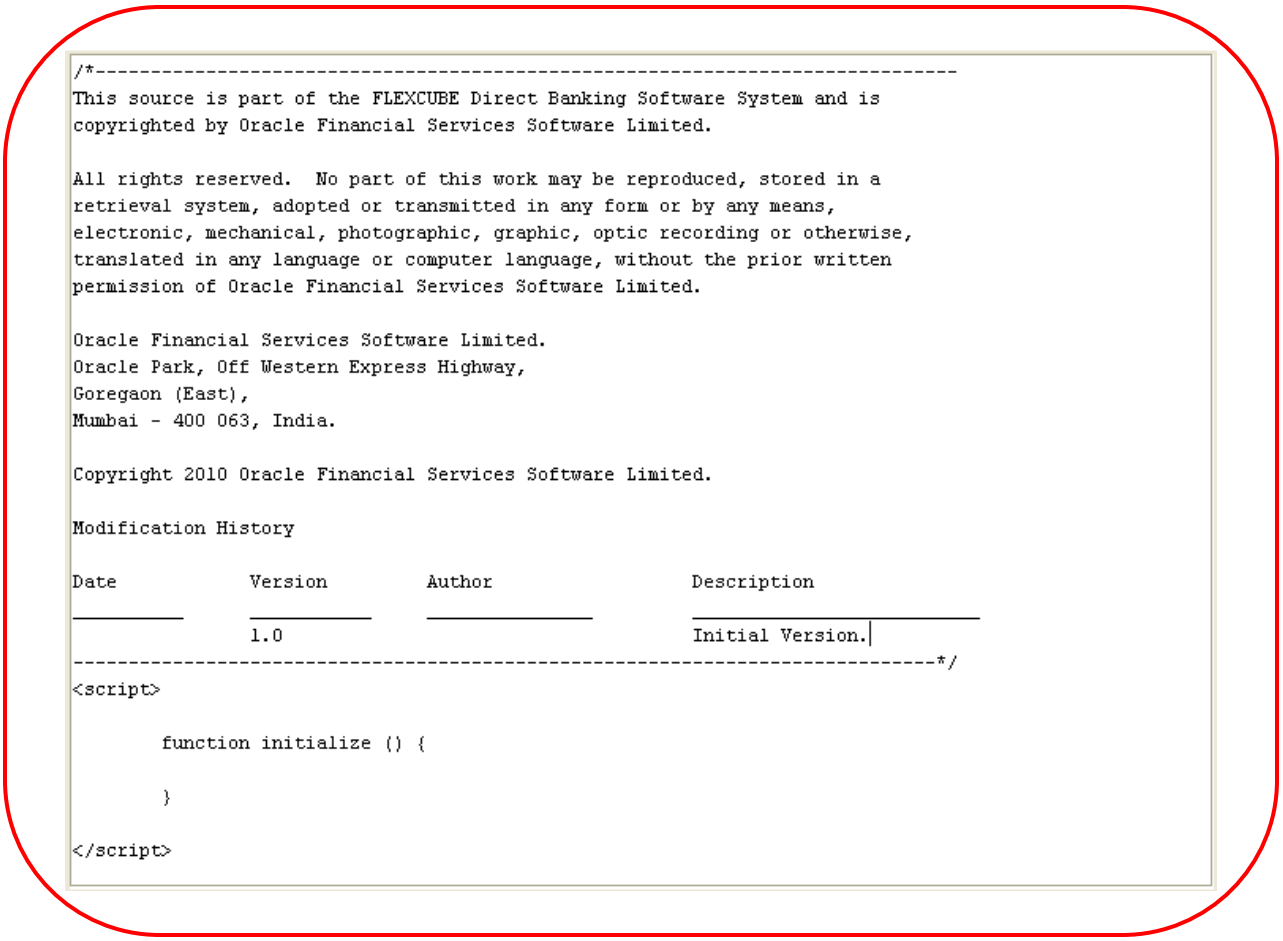
8. Workspace Details Panel

The topmost left panel of this tool as shown below contains the details of all the screentemplate projects created in the workspace, in a tree layout format. This Tool creates a folder "*ScreenTemplate*" in the user's current directory (where LEAP.jar is located), this folder is the workspace for this tool. This tree layout represents the actual directory structure created inside the workspace folder.



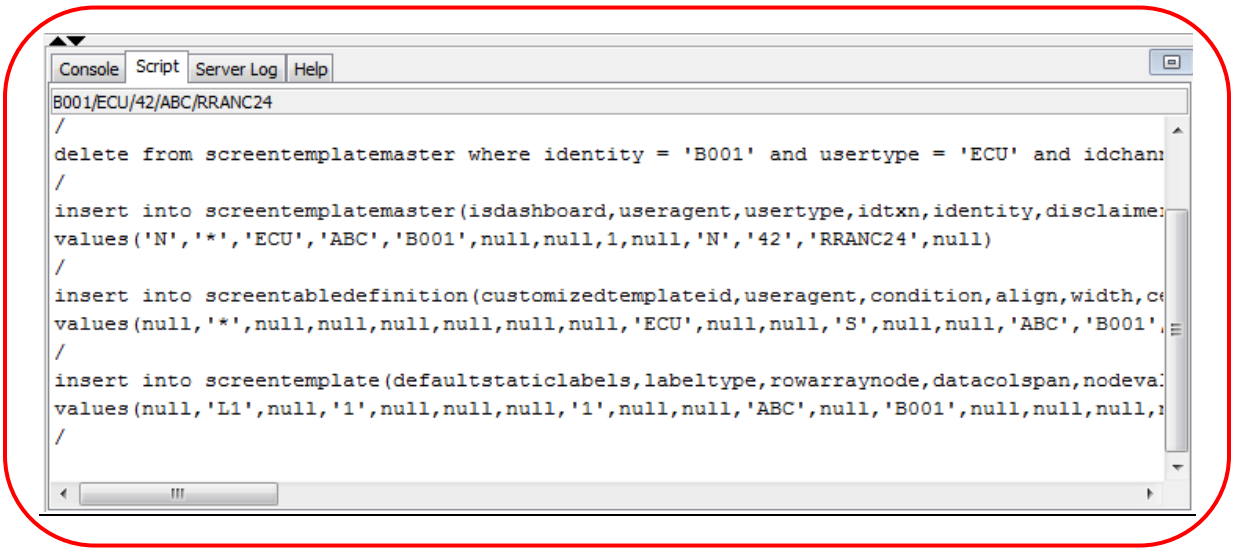
For each screentemplate project there are three files loaded in the workspace

1. <Idrequest>.js: Clicking on this tree node will open an editor panel in the center main work area panel as shown below. The content of the .js files are loaded in this editor. User can use this editor panel to write/edit the content of the JavaScript file. To save the changes user needs to click the submenu save (Under File Menu) or use "Ctrl+S" as keyboard shortcut.



```
/*-----------------------------------------------------------------------------
This source is part of the FLEXCUBE Direct Banking Software System and is
copyrighted by Oracle Financial Services Software Limited.

All rights reserved.  No part of this work may be reproduced, stored in a
retrieval system, adopted or transmitted in any form or by any means,
electronic, mechanical, photographic, graphic, optic recording or otherwise,
translated in any language or computer language, without the prior written
permission of Oracle Financial Services Software Limited.

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

Copyright 2010 Oracle Financial Services Software Limited.

Modification History

Date            Version         Author                  Description
_____           _____           _____                   _____
                1.0                                     Initial Version.
-----------------------------------------------------------------------------*/
<script>

        function initialize () {

        }

</script>
```

2. < Idrequest >.sql: Clicking on this tree node menu opens the script text panel in the bottom console pane under the script tab as shown below. All the scripts generated for a particular screentemplate project are displayed in this text panel.

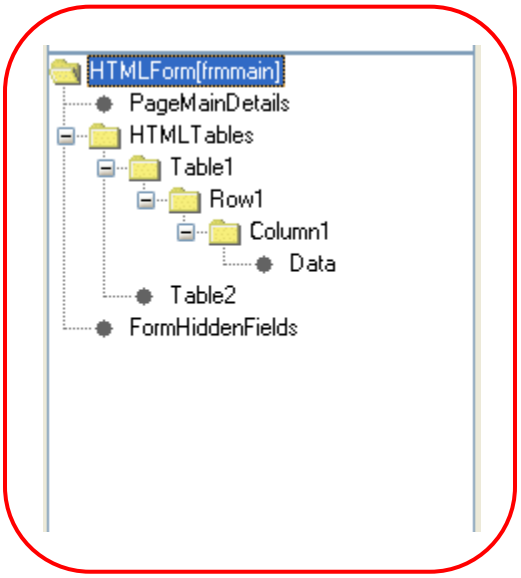


Console | Script | Server Log | Help

B001/ECU/42/ABC/RRANC24

```
/
delete from screentemplatemaster where identity = 'B001' and usertype = 'ECU' and idchan
/
insert into screentemplatemaster(isdashboard,useragent,usertype,idtxn,identity,disclaime
values('N','*','ECU','ABC','B001',null,null,1,null,'N','42','RRANC24',null)
/
insert into screentabledefinition(customizedtemplateid,useragent,condition,align,width,c
values(null,'*',null,null,null,null,null,null,'ECU',null,null,'S',null,null,'ABC','B001'
/
insert into screentemplate(defaultstaticlabels,labeltype,rowarraynode,datacolspan,nodeva
values(null,'L1',null,'1',null,null,null,'1',null,null,'ABC',null,'B001',null,null,null,
/
```

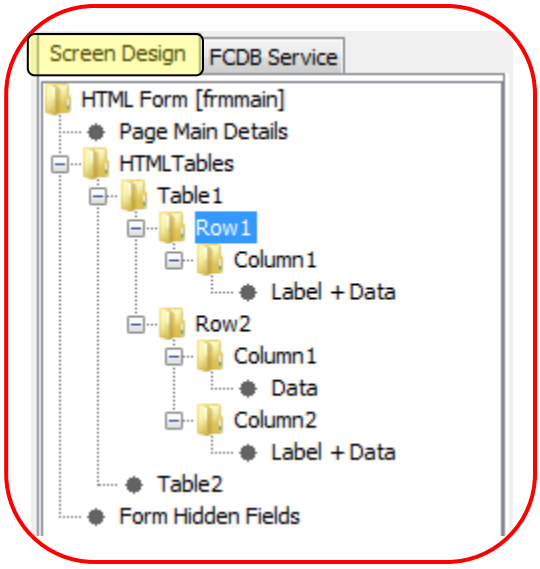
3. < Idrequest >.xml: Clicking on this tree node loads the HTML Layout Panel, in the bottom left of this tool, with the HTML specific details of the screentemplate as shown below.



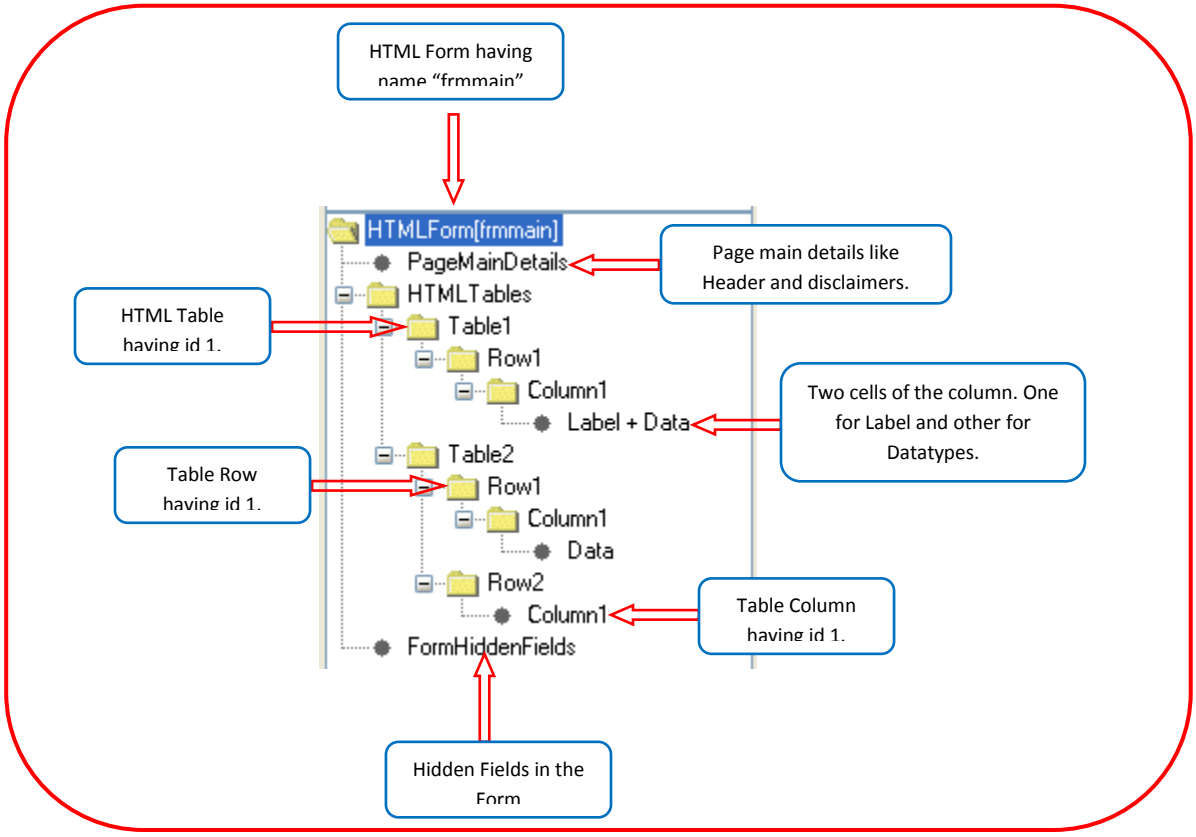
These details are loaded in the form of a tree structure where events are attached to the nodes of the tree. Details of this tree structure and their events have been discussed in the next section.

9. HTML Layout Panel

The HTML Layout Panel opens up on clicking the Screen Design Tab at the bottom-left of this tool as shown below.

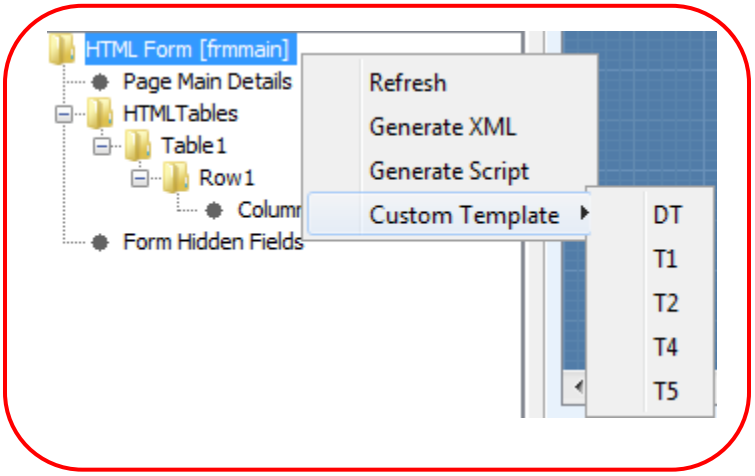


This panel contains the HTML specific details of the screentemplate. This panel loads the screen design in a tree layout format, where each node represents certain HTML component of the screen design as shown below.



Certain nodes shown in the screen shot above have right-click options, user can use these options to add child nodes or delete these nodes. Details of the right-click options available on the nodes has been discussed below.

1. HTMLForm: User can right-click on this node and will get the following option as shown below.



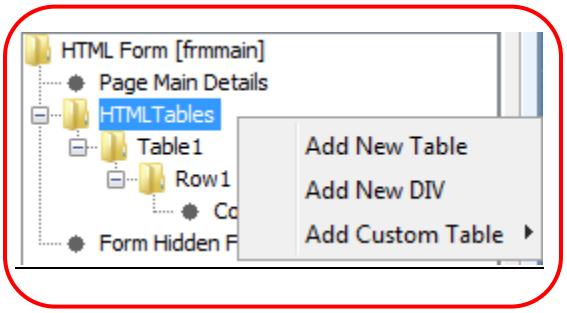
i. Refresh: Click on this option it refresh the Layout tree. It means Layout tree is render again and if any table order is changed it reflect after the refresh.

ii. Generate XML: Click on this option generate XML corresponding to current state of DTO.

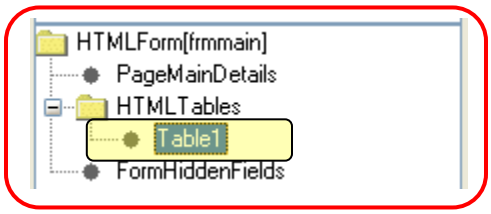
iii. Generate Script: Click on this option generate SQL script corresponding to current state of DTO.

iv. Custom Template: Click on these option display custom templates which are available. This feature developed for mobile banking because many of their screens has same layout.

2. HTMLTables: User can right-click on this node and will get the following option as shown below.

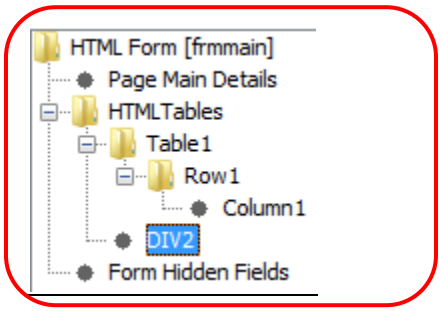


i. Add New Table: Click on this option to add a new HTML Table to your screen design. A new Table with a numeric id will automatically get added under the "HTMLTables" node as shown below.

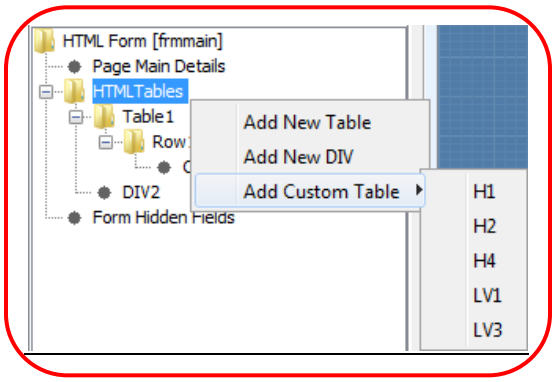


Double-Click this node or Press "F2" to change the numeric id of this newly added table.

ii. Add New DIV: Click on this option to add a new HTML DIV to your screen design. A new DIV with a numeric id will automatically get added under the "HTMLTables" node as shown below.

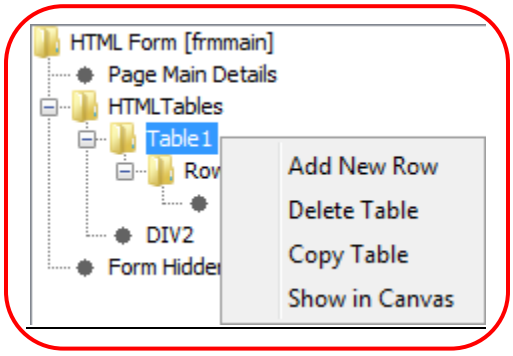


iii. Add Custom Table: Click on this option to add a new predefined HTML Table such as H1, H2, H4, LV1, and LV3 to your screen design. A predefined table with a numeric id will automatically get added under the "HTMLTables" node as shown below. This feature is available for mobile banking only.

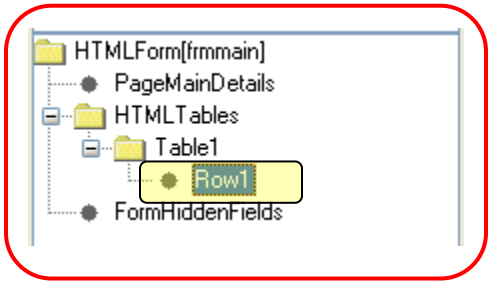


Double-Click this node or Press "F2" to change the numeric id of this newly added table.

3. Table<id>: User can right-click on this node and will get the following options as shown below



i. Add New Row: Click on this option to add a new row to your HTML Table. A new row with numeric id will automatically get added under the current HTML Table as shown below.



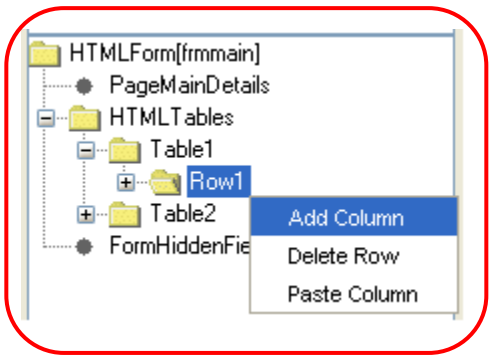
Double-Click this node or Press "F2" to change the numeric id of this newly added table row.

ii. Delete Table: Click on this option to delete the selected HTML table. The current HTML Table will get deleted along with its child nodes, if any.

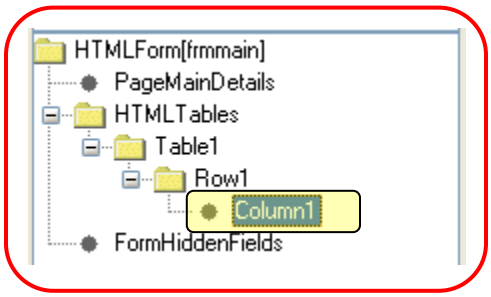
iii. Copy Table: Click on this option to copy this HTML Table. Once copied user will get the right click option to paste the copied table in "HTMLTables" node.

iv. Show in Canvas: Click on this option to show the selected HTML Table on Drawing Canvas.

4. Row<id>: User can right-click on this node and will get the following options as shown below.



i. Add Column: Click on this option to add a new column to your HTML Table Row. A new column with numeric id will automatically get added under the current HTML Table Row as shown below.

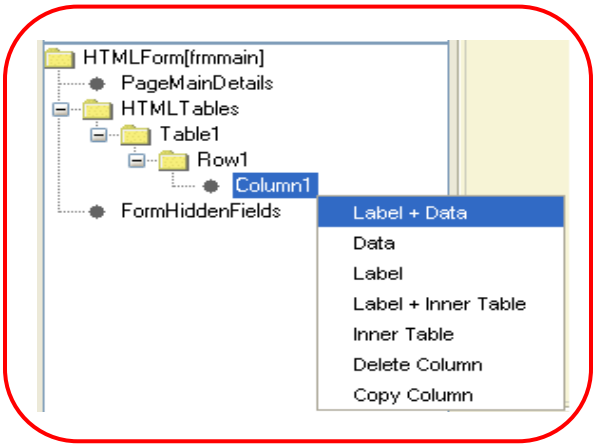


Double-Click this node or Press "F2" to change the numeric id of this newly added column.

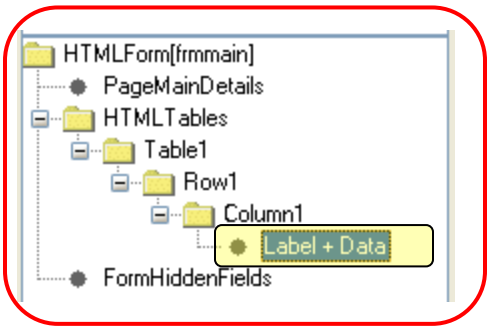
ii. Delete Row: Click on this option to delete the selected row. The current row will get deleted along with its child nodes, if any.

iii. Paste Column: Click on this option to add a copied column to your screen design. This option is available only if user has copied any column. Copying a column has been discussed in following section.

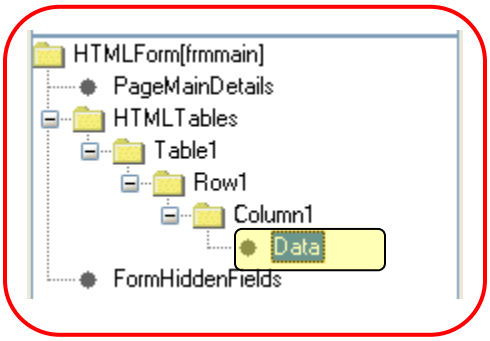
5. Column<id>: User can right-click on this node and will get the following options as shown below.



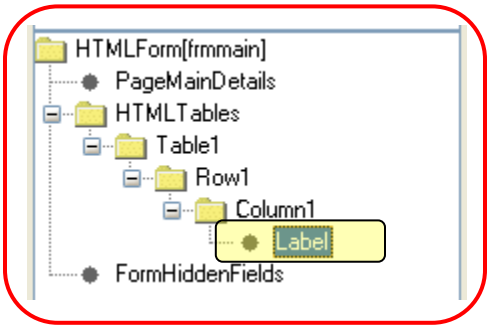
i. Label + Data: Click on this option to add two cells to the selected column, one cell for the label and one for the HTMLDatatypes (for e.g. textbox). A new node will get added representing those two cells, as shown below.



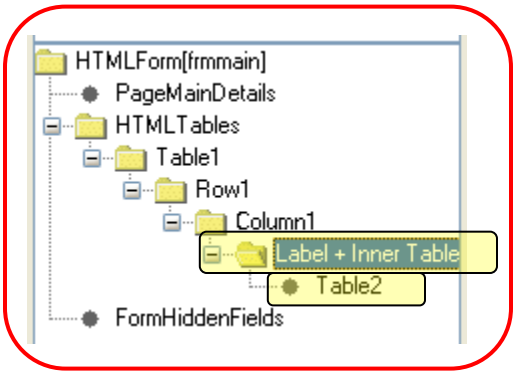
ii. Data: Click on this option to add a single cell in the column to place HTMLDatatypes in the cell. A new node will get added, as shown below, representing a single cell meant for HTMLDatatypes only.



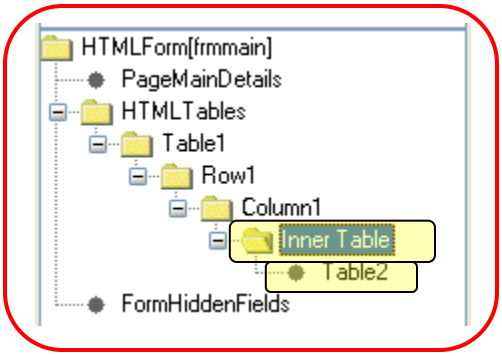
iii. Label: Click on this option to add a single cell in the column to place label in the cell. A new node will get added, as shown below, representing a single cell meant for labels only.



iv. Label + Inner Table: Click on this option to add two cells to the selected column, one cell for adding a label and one cell for adding an inner HTML Table. A new node will get added representing those two cells, as shown below. A new Table<id> node will also get added as a child of this new node representing the inner HTML table.



v. Inner Table: Click on this option to add a single cell in the column to place an inner HTML table in the cell. A new node will get added, as shown below, representing a single cell meant for inner HTML table only. A new Table<id> node will also get added as a child of this new node representing the inner HTML table.

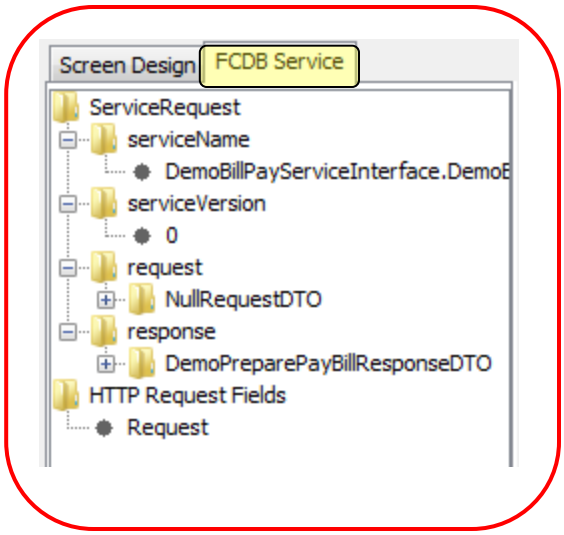


vi. Delete Column: Click on this option to delete the selected column. The current column will get deleted along with its child nodes, if any.

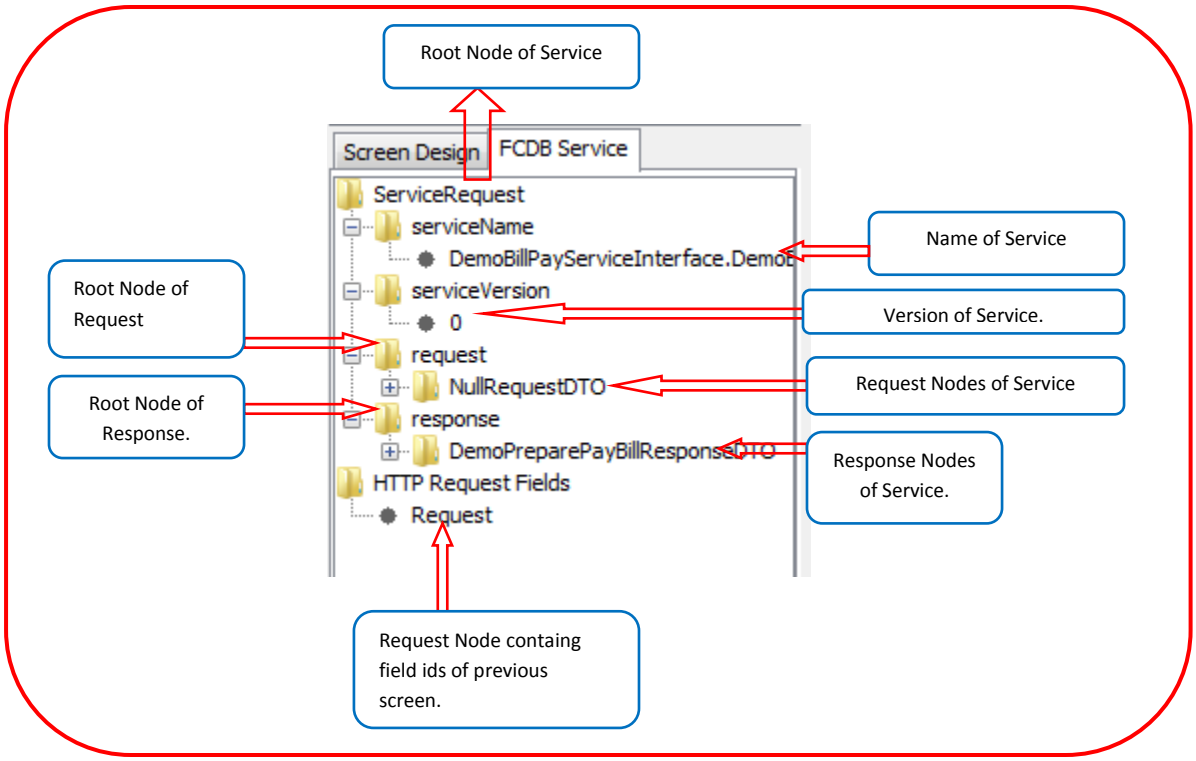
vii. Copy Column: Click on this option to copy this Table column. Once copied user will get the right click option to paste the copied column in table's row node.

10. FCDB Service Panel

The FCDB Service Panel opens up on clicking the FCDB Service Tab at the bottom-left of this tool as shown below.

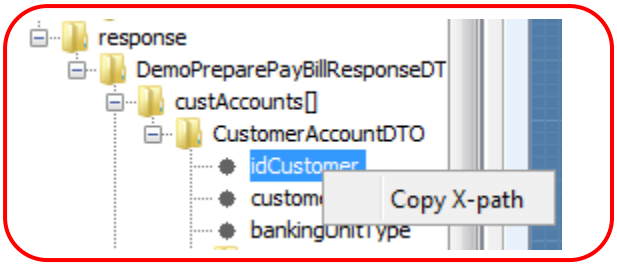


This panel contains the FCDB Service specific details that the user has selected while creating the Project. This panel loads the FCDB Service in a tree layout format, where each node represents certain element of the service as shown below.



Copying X-Path:

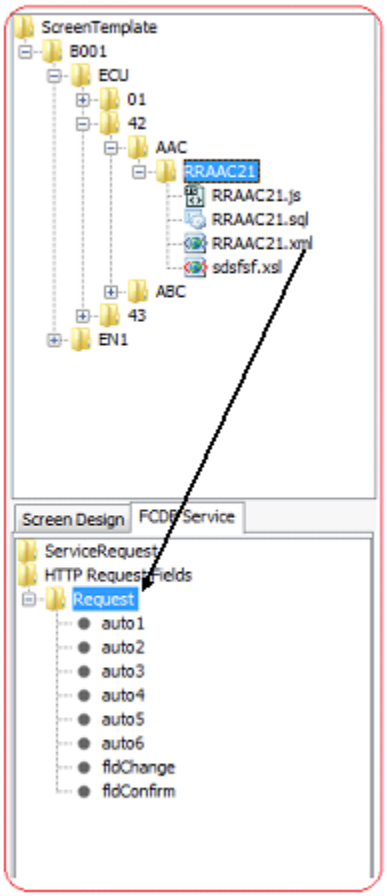
User can Copy X-path of any node from response of service by right-clicking on the node and then selecting Copy X-path as shown below.



Copied X-path:

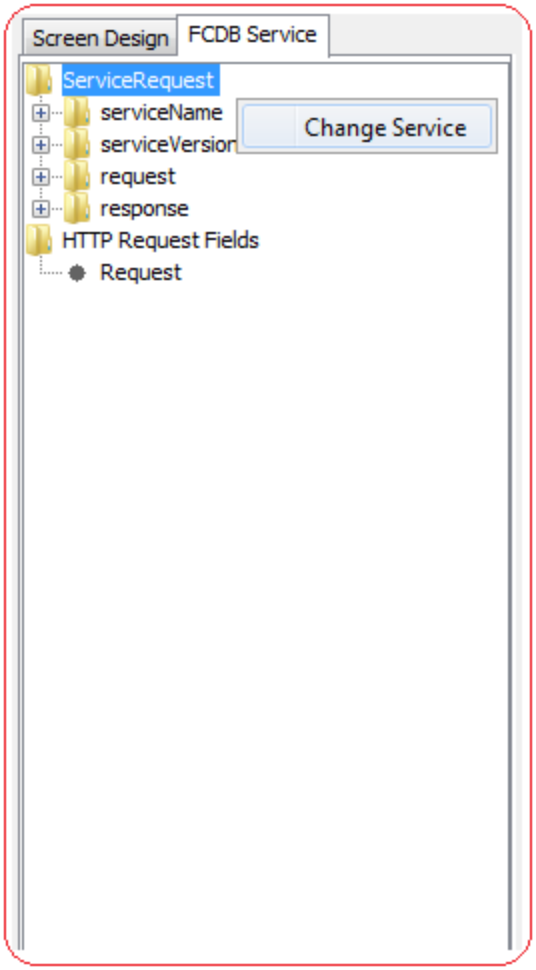
/faml/response/demopreparepaybillresponsedto/custaccounts[]/customeraccountdto/idcustomer

Field Names of another screen project can be added in the current FCDB Service tree by dragging the project's ".xml" file to the "Request" node of "HTTP Request Fields" in current FCDB Service tree as shown below:

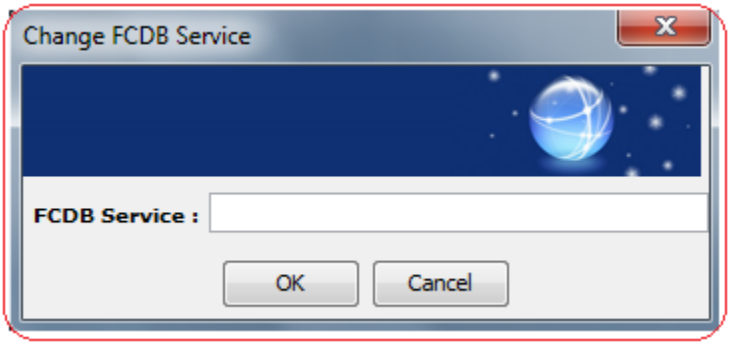


Changing FCDB Service:

FCDB Service can be changed by right clicking mouse on the root node “ServiceRequest" of FCDB Service tree as shown below:



A new dialog will appear providing the user with the list of FCDB Services to be selected as shown below:

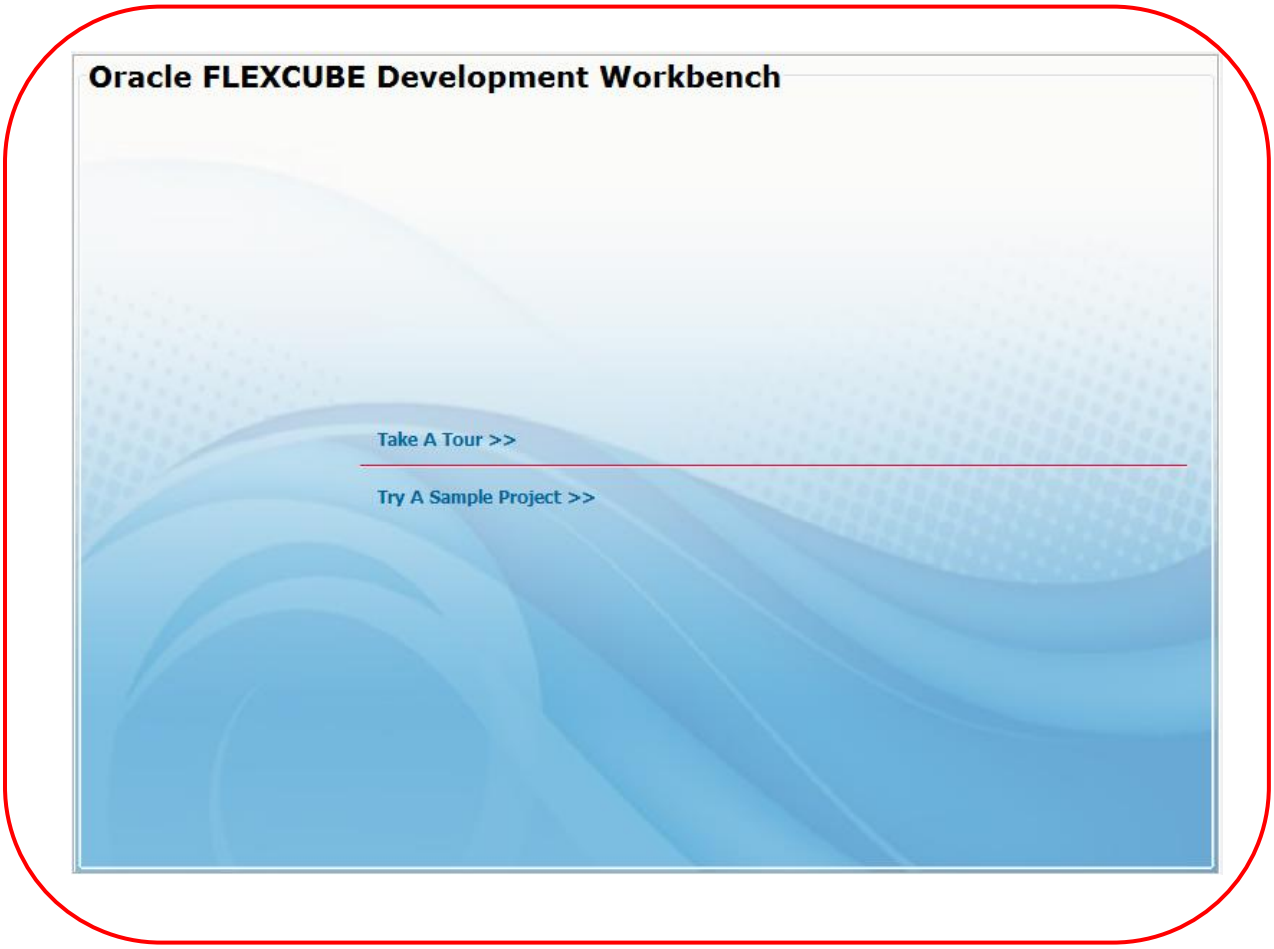


11. Main Work Area Panel

This panel appears at the center of this tool. This panel consists of several sub-panels which appear depending upon what component of screen design is being currently worked on by the user. Following are the sub-panels which appear inside this panel.

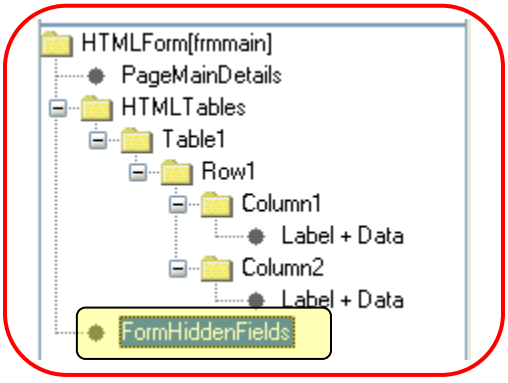
11.1. Welcome Panel

This is the default panel which will appear in the main work area panel, as shown below, when the tool is launched. Once the user starts working on any screentemplate project, this panel will be replaced by some other sub-panel depending upon what component of screen design user is configuring. User can come back to this panel by using “Go to Home” sub-menu under the “Help” menu.

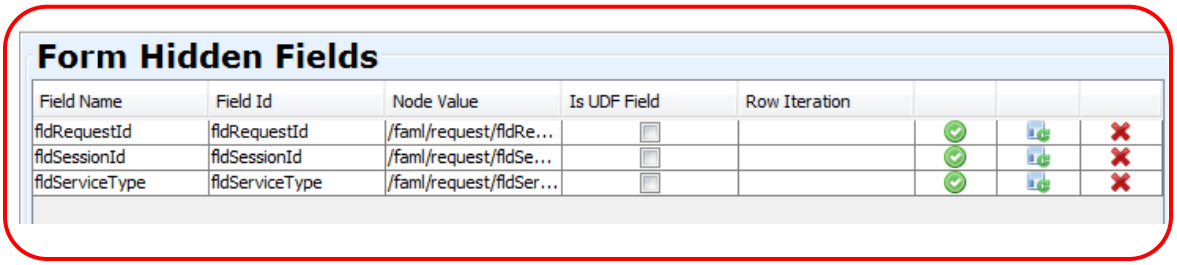


11.2. Form Hidden Fields Panel

This panel appears when the user clicks on “FormHiddenFields” node of HTML Layout panel, as shown below.



Clicking this node will display “Form Hidden Fields” panel, as shown below. In this panel user will be able to configure the HTML Form Hidden Fields. User can add, delete and modify hidden fields.



As shown in the screen shot above the hidden fields are displayed in a table grid format with following columns

i. Field Name: User has to enter a field name. It is a mandatory field and cannot be left blank. It is always recommended to use “fld” as prefix with the field name, for e.g. fldbeneficiary.

ii. Field Id: User has to enter a field id. It is a mandatory field and cannot be left blank. This field id has to be unique, user cannot use any field id which has been used while defining the screen. It is always recommended to use “fld” as prefix with the field id, for e.g. fldbeneficiaryid.

iii. Node Value: Node value is the value assigned to this field. It can be either a XPATH or a constant value. For constant value user has to use the syntax string(‘<value>’).

iv. Is UDF Field: This is a check-box. If checked, a name-value pair of this field can be assigned to UDFDTO and passed to the service.

v. Row Iteration: In certain scenario's, one need to create array of hidden data using an xsl:for-each loop on a particular xpath. To create such an array one can use this column. In this column user needs to enter the xpath on which the iteration has to be done.

vi. Condition Field: This feature is only supported in mobile banking. It's a XSL expression which return either true or false on basis of this application decides that this hidden field will render or not.

Three icons are there against each row of hidden field detail

i. : Click on this to apply the changes done on the corresponding row of the hidden field's grid table.

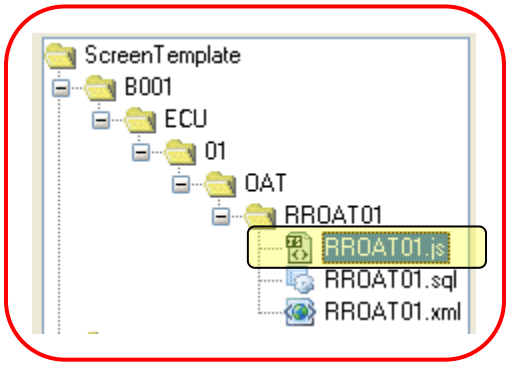
ii. : If the user has not clicked on "" and wants to go back to last applied changes, they can click on this icon.

iii. : Click on this to delete the corresponding row. Deleting the row will remove the hidden field from the HTML form.

To add a new row to this grid table, select the last column of the last row and click tab, a new row will get added where user can define a new hidden field.

11.3. Java Script Editor Panel

This panel appears when the user clicks on the "<idrequest>.js" file node of a particular project in Work Space Details Panel, as shown below



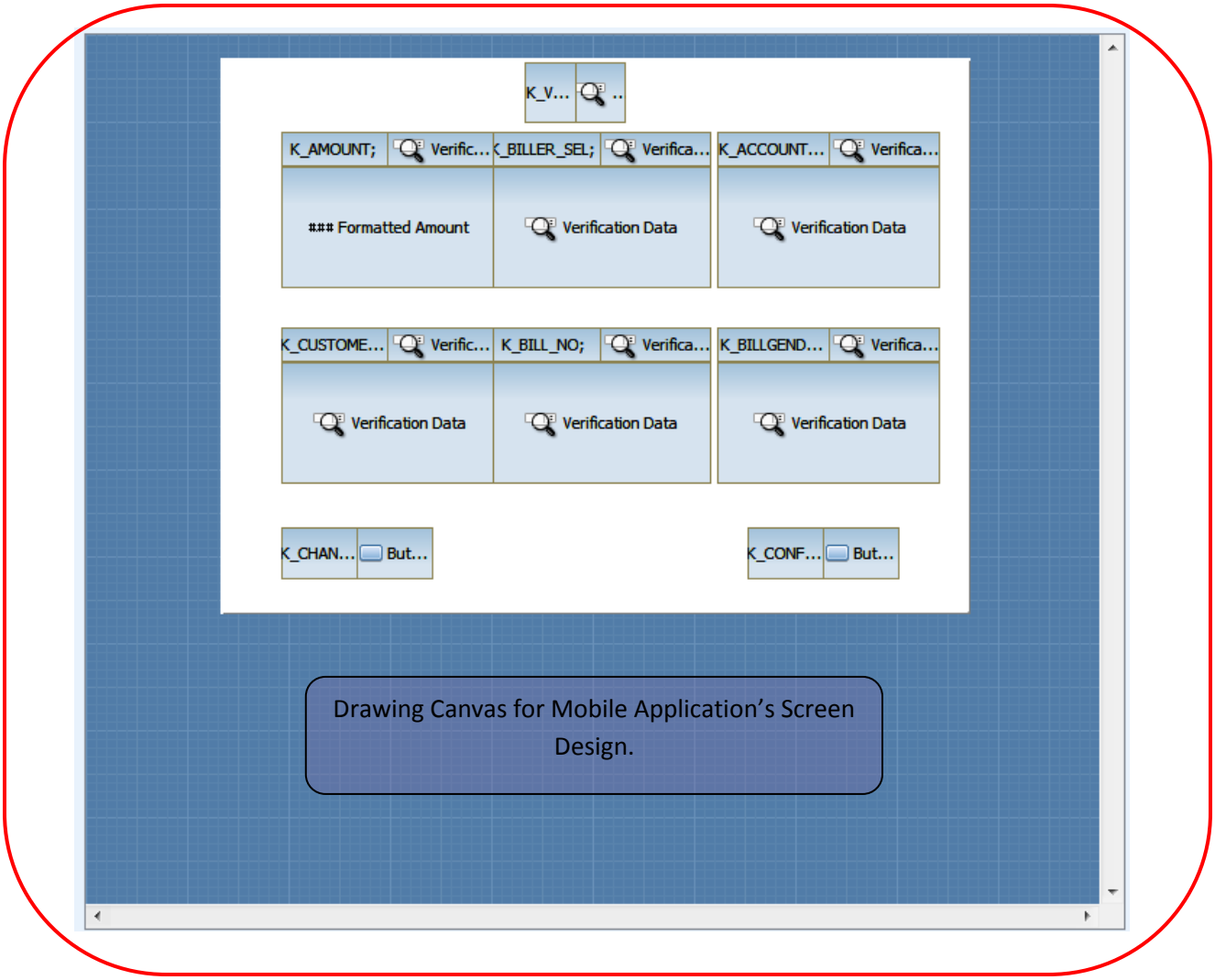
Clicking this node will open an editor panel in the center main work area panel as shown below. The content of the .js files are loaded in this editor. User can use this editor panel to write/edit the content of the JavaScript file. To save the changes user needs to click the submenu "Save" (Under File Menu) or use "Ctrl+S" as keyboard shortcut.

```
<script language="JavaScript" type="text/JavaScript">
//--------------------------------------------------------------------------------
var table1,
        table2,
        table3,
        table4,
        table5,
        table6,
        transAmt,
        dealBox;
//--------------------------------------------------------------------------------
function fnCreateTableObject () {
        table1          = eval('document.getElementById("tablediv1")');
        table2          = eval('document.getElementById("tablediv2")');
        table3          = eval('document.getElementById("tablediv3")');
        table4          = eval('document.getElementById("tablediv4")');
        table5          = eval('document.getElementById("tablediv5")');
        table6          = eval('document.getElementById("tablediv6")');
        transAmt        = eval('document.getElementById("fldtransferamt")');
        dealBox         = eval('document.getElementById("flddealtemplate")');
}
//--------------------------------------------------------------------------------
function initialize () {
        fnSetDataId ();
        fnCreateTableObject ();
        fnIsDealAllowed (document.frmmain.fldflagdealallowed.value,dealBox);
        fnResetRequestValues ();
        fnOnPaymentToClick ('request');
        fnHideShowTables ();
```

This editor panel has the potential to highlight the selected text as shown in the screen shot above. The current cursor is on text “fnCreateTableObject”, clicking on that text automatically highlight the same text in the entire document with a grey background color. User can also click the submenu “Find” (Under File Menu) or use “Ctrl+F” as keyboard shortcut to find any text in the JavaScript file.

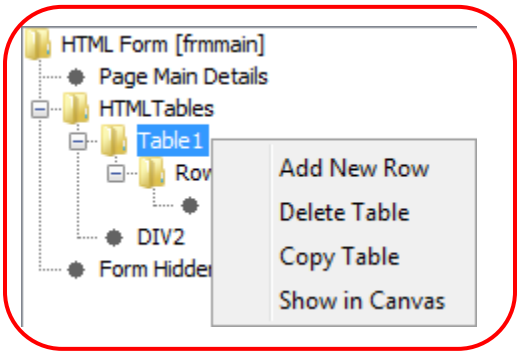
11.4. Drawing Canvas

Drawing Canvas is available for user agents(Ipad, Iphone, Android Phone and Android tab) of Mobile Application Channel. The Canvas represents the screen which allows the user to draw table and table components at the required position on the screen as shown below.

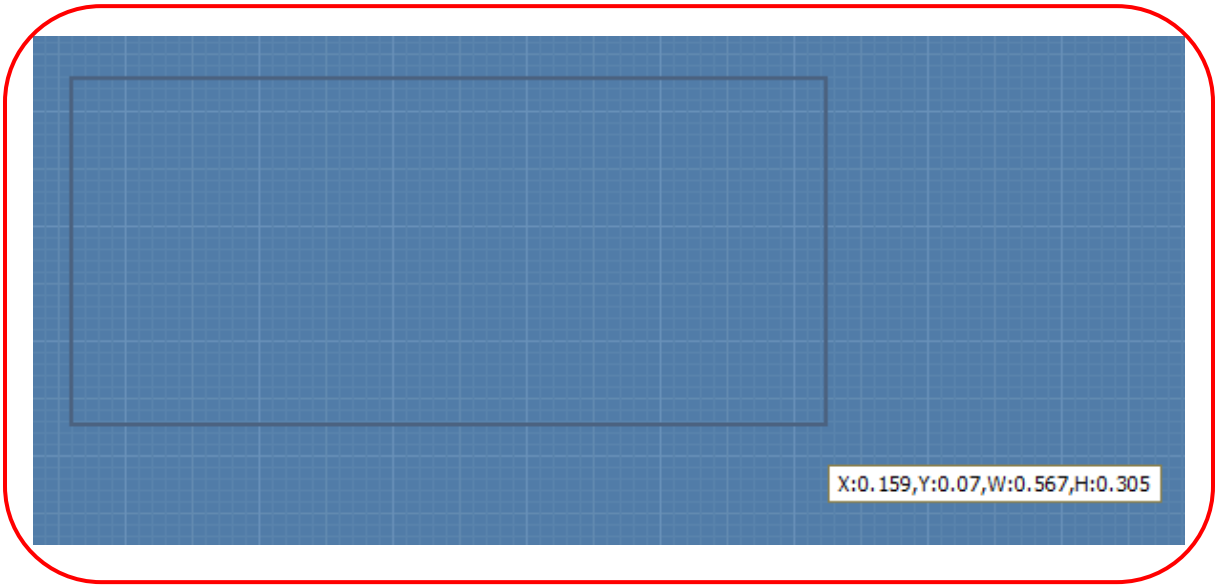


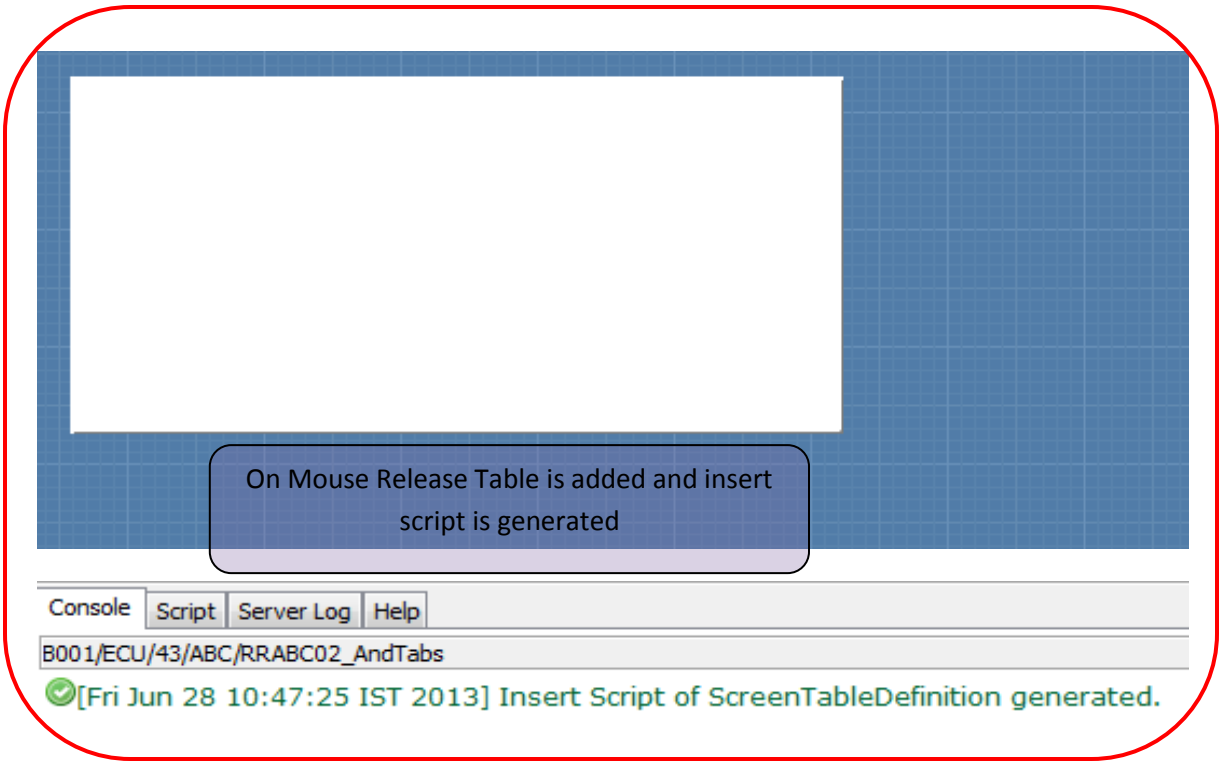
Table's and Component's Position X, Position Y, Width and Height will be calculated as % of Screen i.e. it is relative to the total width and height of the screen. Features such as drag/stretch table and components from one position to another and resize of table and components are provided. User can drag data types, label types and custom types from html data types pane to table and components on Drawing Panel and their corresponding scripts will be generated and their corresponding DTO's will be updated.

The Canvas only shows tables having display type as “Default”. In order to view tables with default display type other than “default” i.e. (Hidden, Modal, Popup), user has to explicitly select the table from the screen layout tree and click on “Show in canvas option” as shown below,

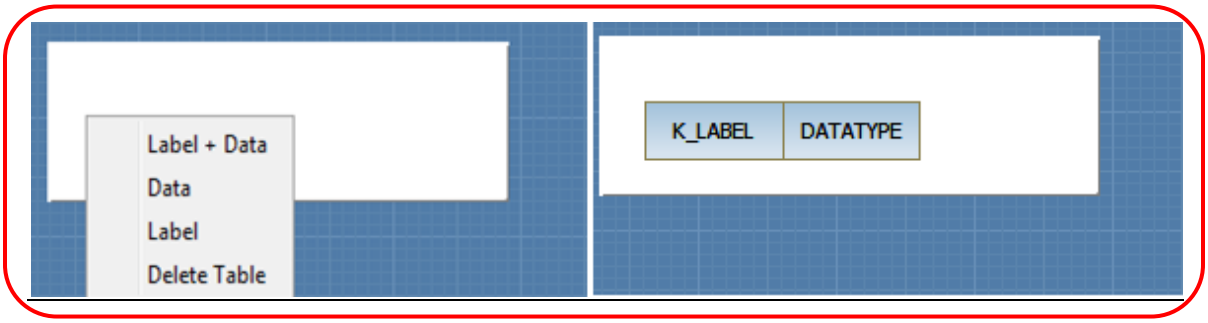


1) Adding a Table: To add a table in drawing canvas user has to press and drag the mouse at the required position in screen. Table current Relative Position X, Y, Width and Height will be shown while dragging. On releasing the mouse button, table will be added to the canvas and the corresponding scripts will be generated as shown below.



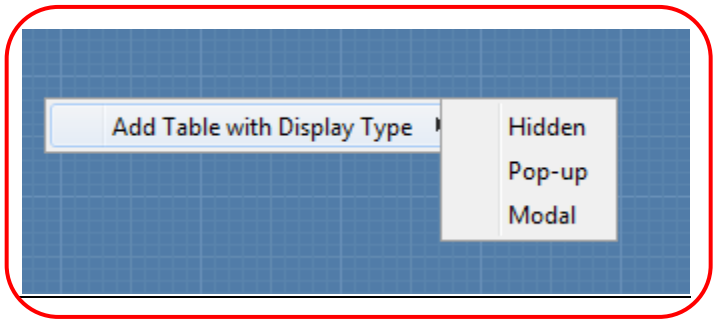


2) Adding a Component: To add a component to the canvas, user first has to right-click on the table and then click on the type of the component to be added. The Component with the desired type get added to the corresponding table as shown below.



3) Dragging/Resizing Table/Component: User can also drag/stretch table/Component in order to change the position of the table/Component on screen or resize the table/Component respectively.

4) Adding a table with Display Type: User can add a table with default display type other than "Default" (i.e. Hidden/Modal/Popup) by right clicking on the drawing canvas as shown below.

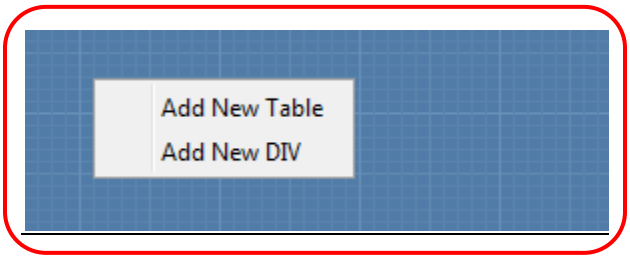


11.5. WireFrame Panel

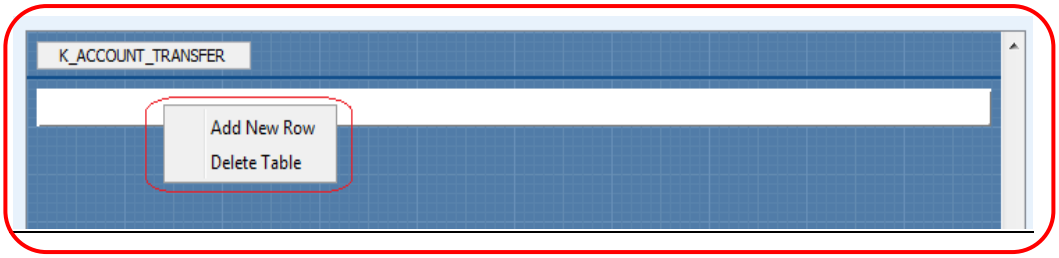
Wireframe Panel provides a wire frame of the tables and components added on the screen for channels other than the user agents (Iphone, Ipad, Android Phone, Android Tab) of Mobile Application as shown below.

Creating a screen component can achieved in just few easy step.

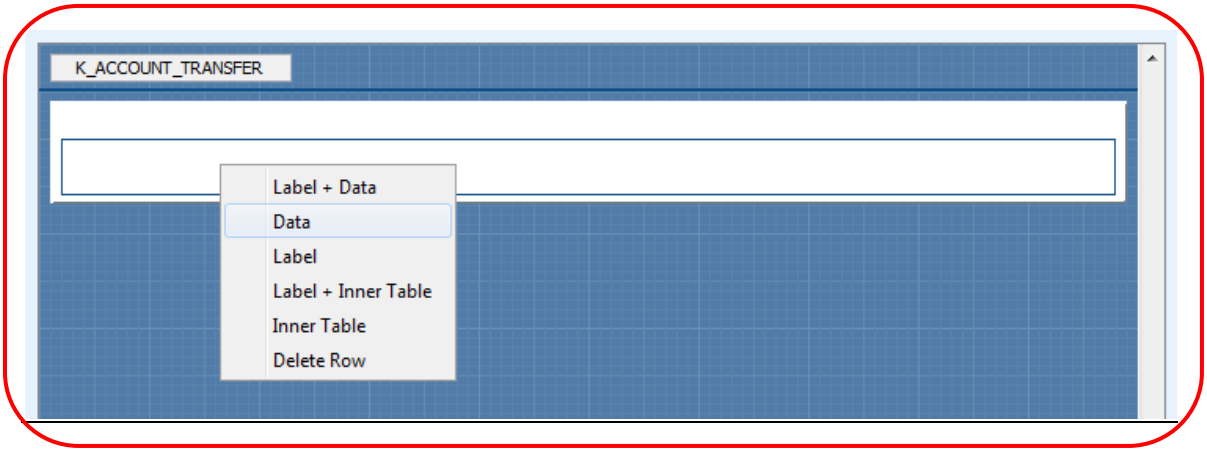
1) Adding table/DIV: User can directly add a table/DIV from the wire frame panel by clicking on the wire frame panel as shown below,



2) Adding Row: User can directly add a row from the wire frame panel by right clicking on a table/DIV as shown below,

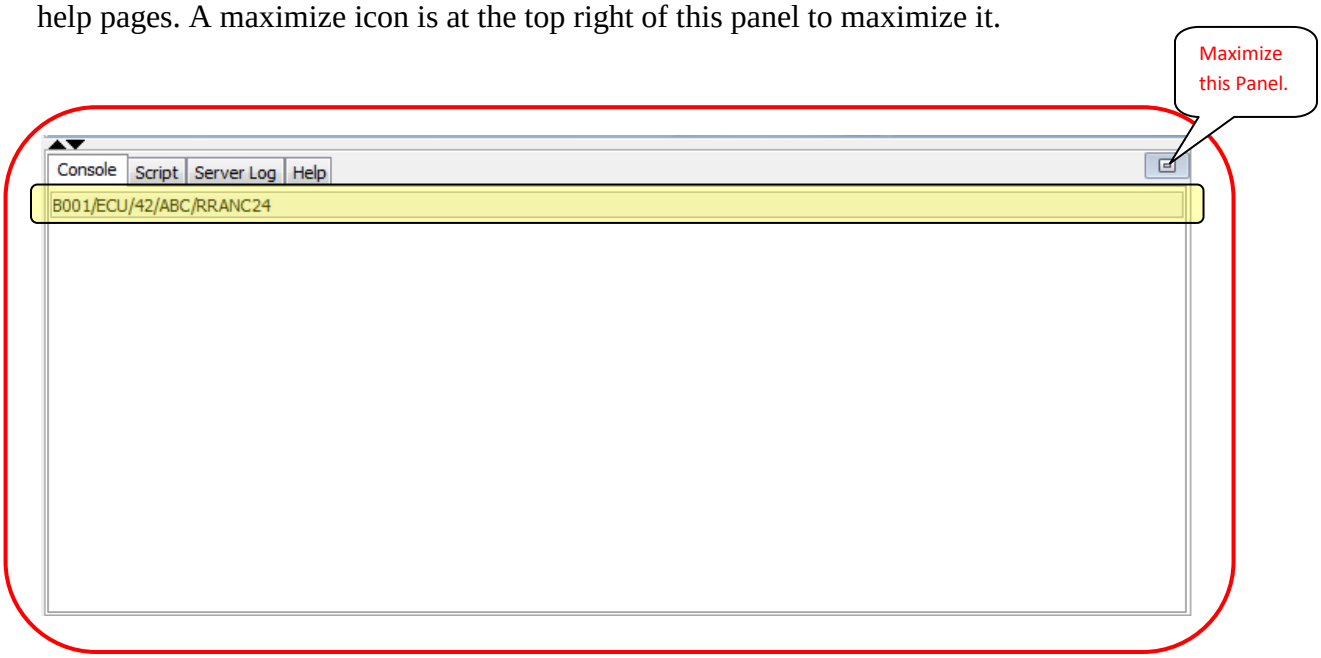


3) Adding Component: User can also directly add a component from the wire frame panel by right clicking on a row as shown below,



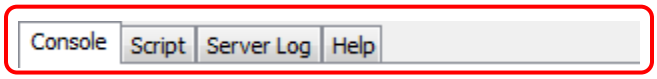
12. Console Panel

This panel is at the center-bottom of this tool. This panel, as shown below, is meant to display the messages generated by the tool, the scripts generated by the tool and the tool help pages. A maximize icon is at the top right of this panel to maximize it.

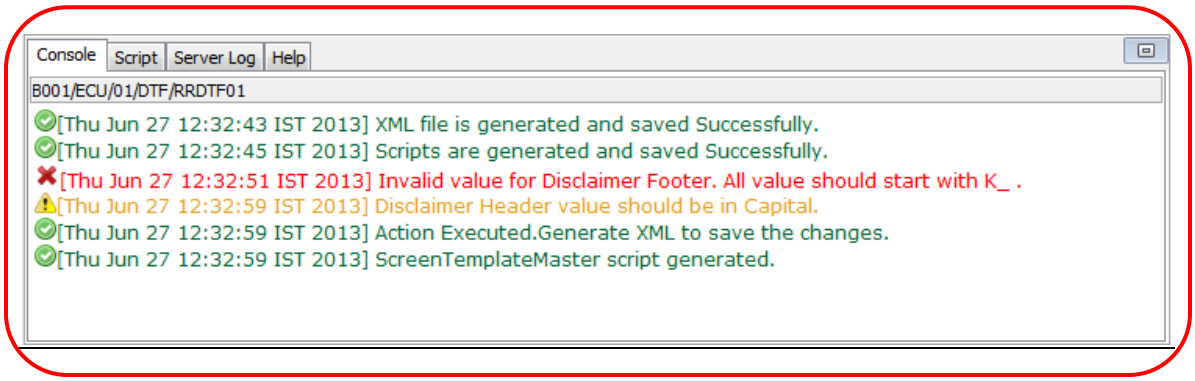


As highlighted in the screenshot above the current project details is also displayed, so that a user can identify and distinguish the console messages and scripts for different screentemplate projects.

This console panel has following four tabs



1. Console: Clicking this tab opens the text panel, as shown below, with the messages generated by the tool while working on a particular screentemplate project.



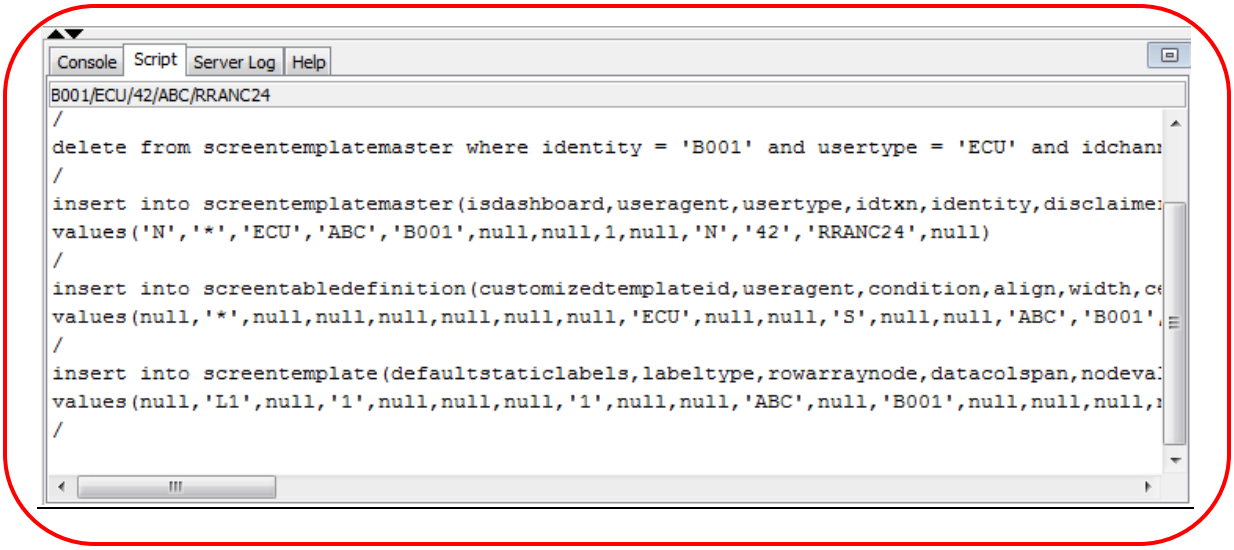
The messages displayed in the console can be of three types

a. Success: For successful events/changes done by user, the message generated by the tool is displayed as text in green color with icon

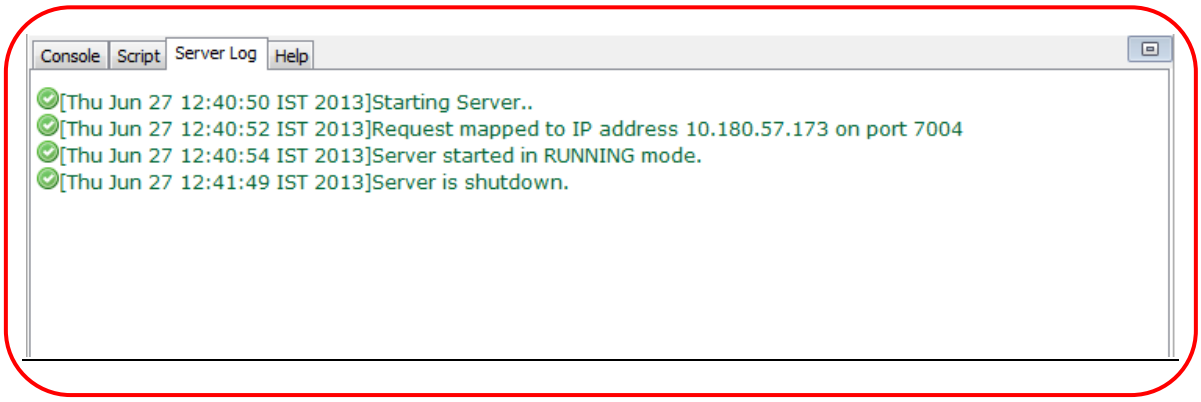
b. Error: For events/changes done by user, which are not correct, the message generated by the tool is displayed as text in red color with icon

c. Warning: For events/changes done by user, which are correct but are not as per standards, the message generated by the tool is displayed as text in yellow color with icon

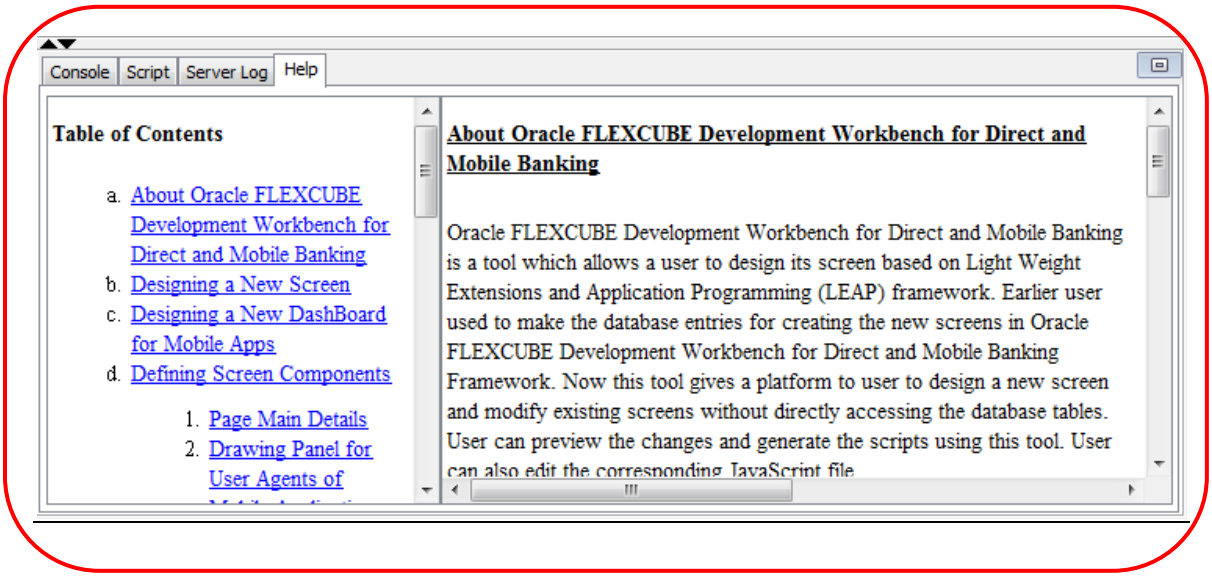
Script: Clicking this tab opens the text panel, as shown below. All the scripts generated for a particular screentemplate project are displayed in this text panel.



2. Server Log: Cliking this tab opens the text panel, as shown below, with the messages generated by the tool while working with the server.



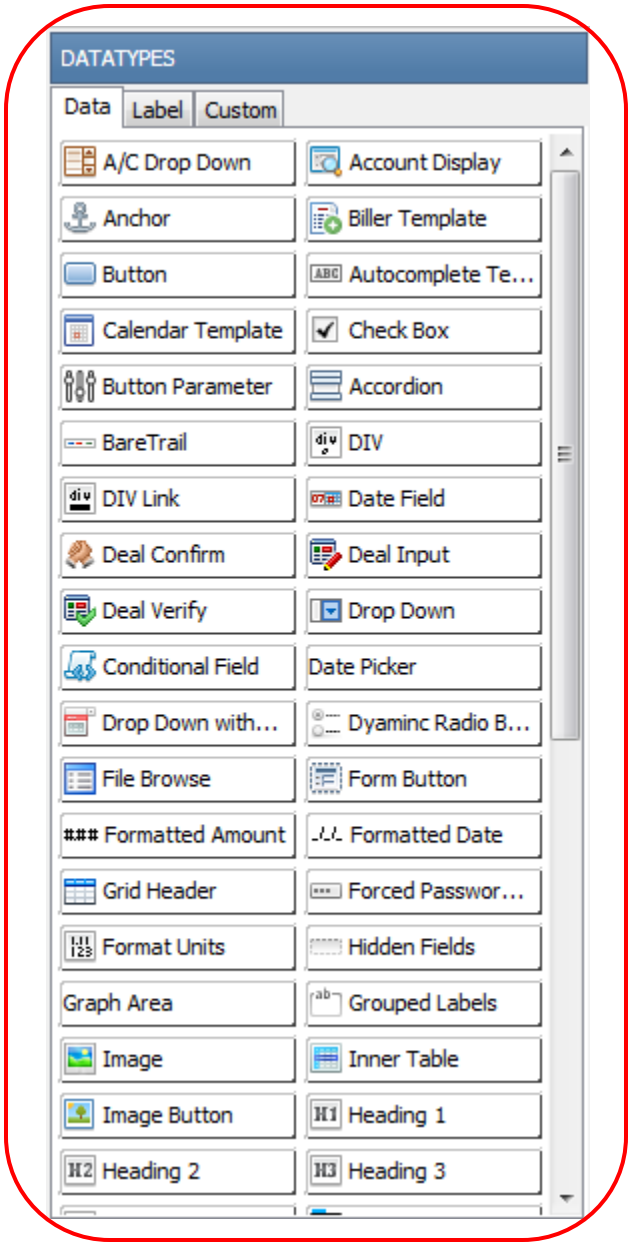
3. Help: Clicking this tab opens the help pages of this tool, as shown below.



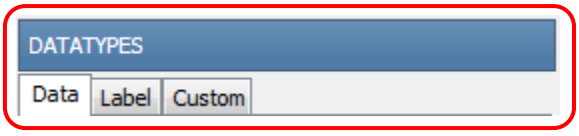
13. Accordion Panel

13.1. HTML DataTypes Panel

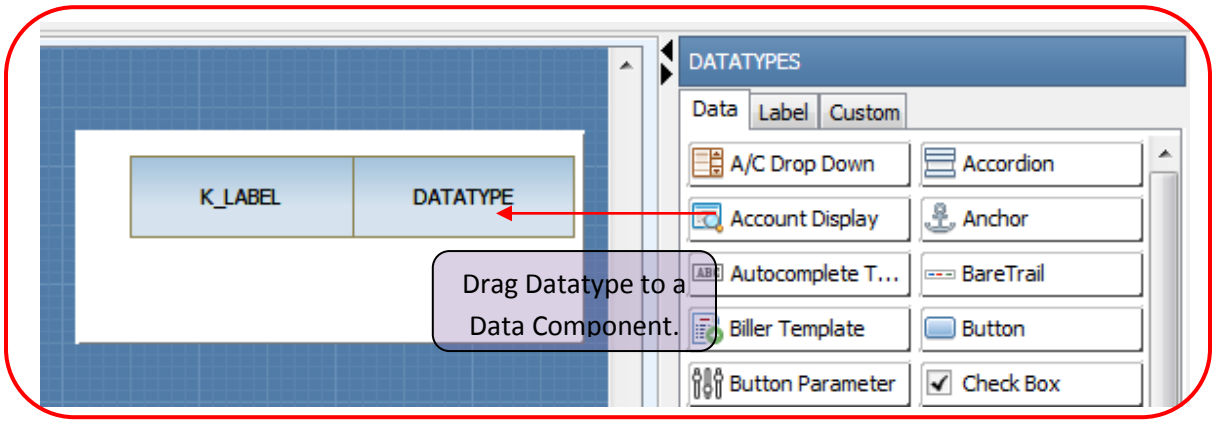
HTML DataTypes Panel This panel is at the top-right of this tool. This panel contains all the HTMLDatatypes currently supported in LEAP framework in the form of button's, as shown below. User can click on these buttons to use these HTMLDatatypes to design the screen.



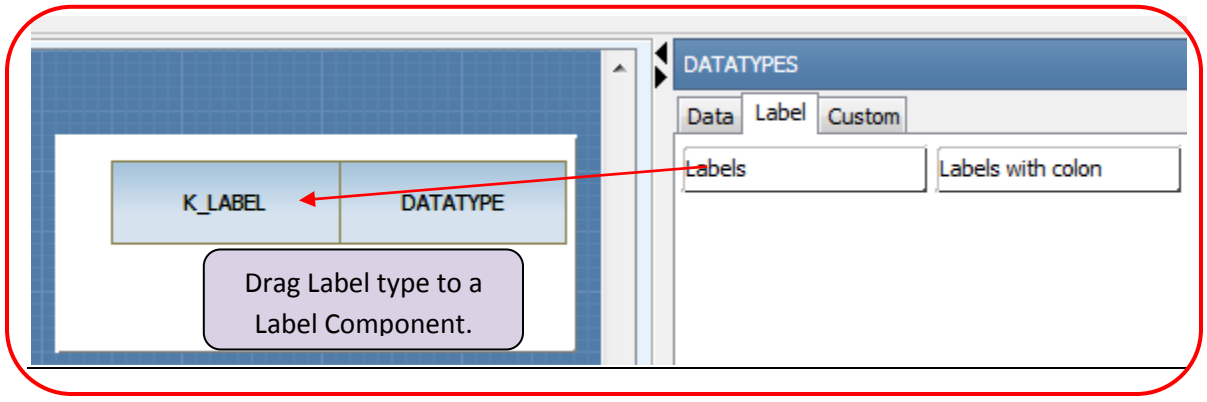
These HTMLDatatypes has been classified into three groups



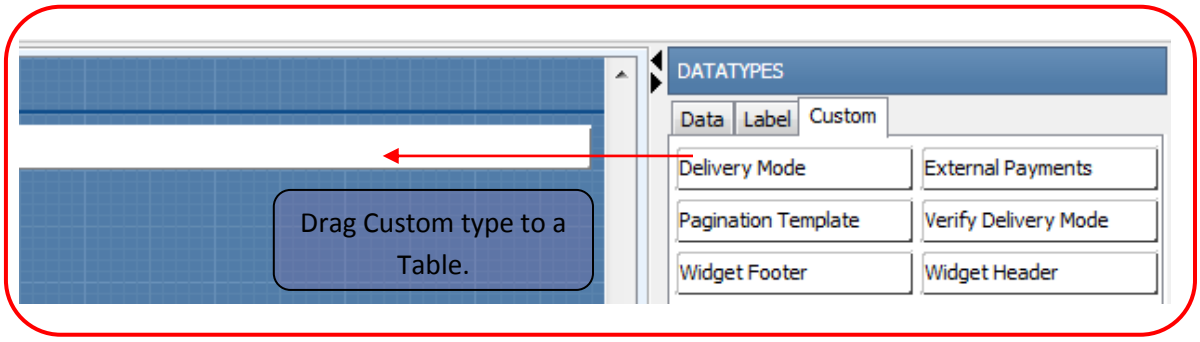
1. Data: These datatypes are those datatypes which helps user to place certain HTML components like textbox, dropdown inside a particular column of a particular row of particular HTML Table.



2. Label: These datatypes help user to configure the labels in the form. Currently there are two configurations available. One is "Labels" which can be used for simple label text with mandatory icon, if any and other is "Labels with colon" which can be used for simple label text with mandatory icon, if any and a colon appended at the end.

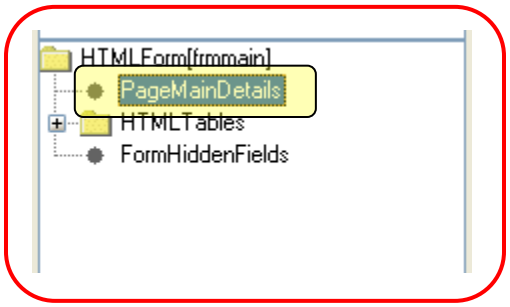


3. Custom: These datatypes are nothing but a predefined layout which one can directly place at certain position of the screen. By predefined layout it means that rather than defining each component of the screen one can call these datatype templates to place an html layout at a particular table sequence. As these layouts can be used at a particular table sequence which means these datatypes can be used while defining the HTML Table attributes. Clicking these datatypes sets the value in "Customized Template ID" drop down of "HTML Table Details" work area panel.

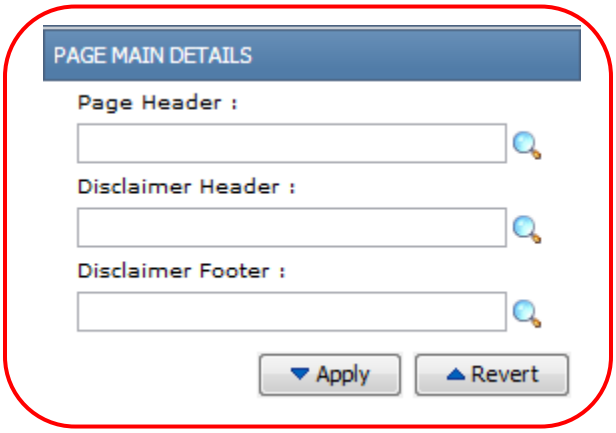


13.2. Page Main Details Panel

This panel appears when the user clicks on "PageMainDetails" node of HTML Layout panel, as shown below.



Clicking this node will display the Page Main Details Panel as shown below.



As shown in the screenshot above user can configure following three screen design parameters

a. Page Header: User needs to enter XSL keyword (starting with K_) in the text box against this field. The text entered here will appear on the top of the screen design as its page header. A lookup icon is available beside the textbox, clicking this lookup image will open a dialog box where user can search available XSL keywords for a particular text, for more details please refer to section "*XSL Keywords Lookup*".

b. Disclaimer Header: User needs to enter XSL keyword (starting with K_) in the text box against this field. The text entered here will appear on the top of the screen design as disclaimer header. A lookup icon is available beside the textbox, clicking this lookup image will open a dialog box where user can search available XSL keywords for a particular text, for more details please refer to section "*XSL Keywords Lookup*".

c. Disclaimer Footer: User needs to enter XSL keyword (starting with K_) in the text box against this field. The text entered here will appear on the bottom of the page as disclaimer footer. A lookup icon is available beside the textbox, clicking this lookup image will open a dialog box where user can search available XSL keywords for a particular text, for more details please refer to section "*XSL Keywords Lookup*".

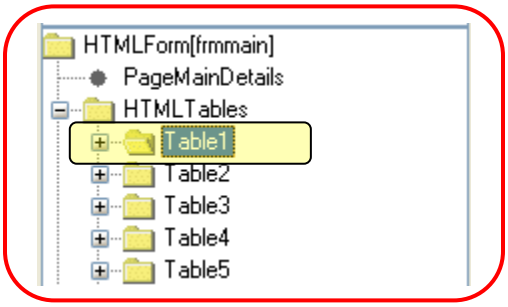
d. Total Step Count: User can set the number of steps that are there in this screen. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

Two buttons are there in this panel

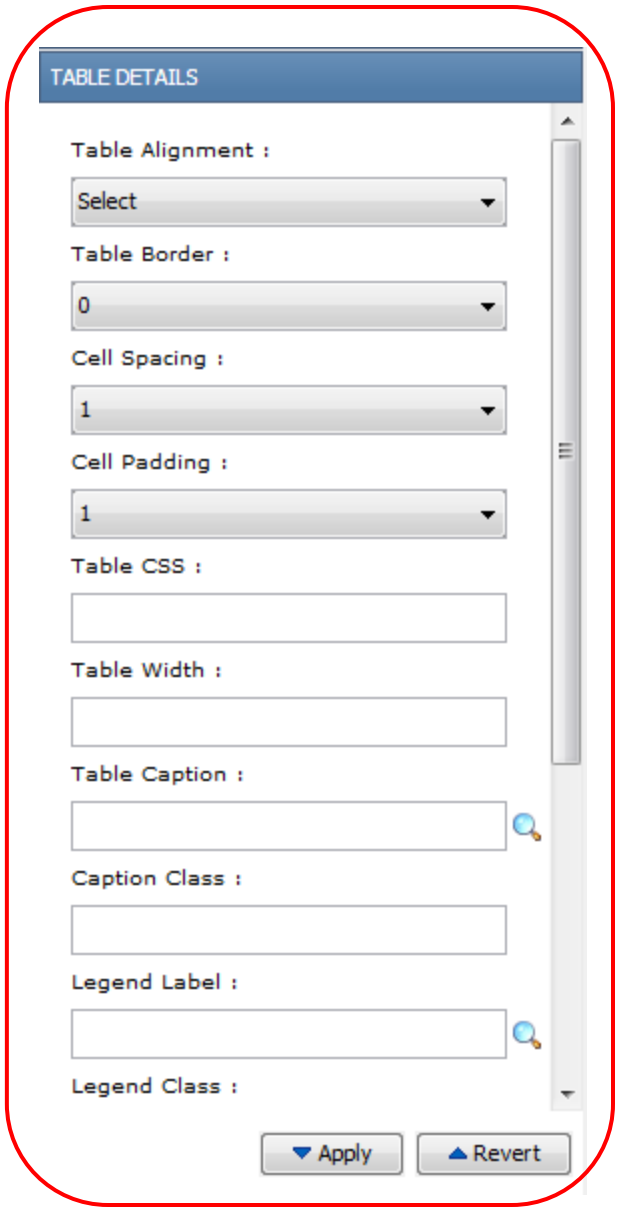
a. Apply: Once the user has entered the above details, they need to click on "Apply" button to apply the changes done. If not clicked the changes will not get reflected on revisiting this panel.
b. Revert: If the user has not clicked on "Apply" and wants to go back to last applied changes, they can click on "Revert" button.

13.3. Table Details Panel

This panel appears when user clicks on any HTML “Table” node of HTML Layout Panel, as shown below



Clicking this node will display “HTML Table Details” panel, as shown below. In this panel user will get the fields to configure the HTML properties of the selected HTML table. User can configure properties like table border, cellspacing, cellpadding and several other properties listed below.



As shown in the screenshot above user can configure following HTML properties of the selected table

i. Table Alignment: User can set the alignment property of the HTML table by selecting the text available in this drop down list against this field. Possible values that can be selected are:

   a. left
   b. right
   c. center

   Tool will not allow the user to apply any value other than the above three values.

ii. Table Border: User can set the border property of the HTML table by selecting the text available in this drop down list against this field. User can select only numeric values >= 0. Tool will not allow the user to apply any other values. Recommended value for border is between 0 to 5.

iii. Cell Spacing: User can set the cell spacing property of the HTML table by selecting the text available in this drop down list against this field. User can select only numeric values >= 0. Tool will not allow the user to apply any other values. Recommended value for cell spacing is 1.

iv. Cell Padding: User can set the cell padding property of the HTML table by selecting the text available in this drop down list against this field. User can select only numeric values >= 0. Tool will not allow the user to apply any other values. Recommended value for cell padding is 1.

v. Table CSS: User can set the class property of the HTML Table by entering the text in the text box against this field. User can only enter the CSS classes which are available in this tool. List of all CSS classes is loaded in the bottom-right panel of this tool under "CSS Classes" tab. User can click on the desired class and the value will be automatically populated in this text box. For more details refer to section "*CSS Classes*" under the heading "*JavaScript Functions and CSS Panel*". Tool will not allow the user to apply any other CSS class name other than those available in this tool. Recommended CSS classes for HTML tables are

   a. formtable: Use this class when the table contains certain input fields.
   b. formtablereadonly: Use this class to display the table on verification screen where no input fields are there.
   c. infotable: Use this class to display certain info inside a table.
   d. standardtable:
   e. gridtable: Use this class for table in grid format
   f. uidownload: Use this class for calling UI download framework of FCDB.
   g. buttonarea: Use this class for making form submission buttons.
   h. tabpanel: Use this class for making set of tabs inside a table.

vi. Table Width: User can set the width property of the HTML table by entering the text in the text box against this field. User can enter numeric value with "px" or "%" as postfix. Tool will not allow the user to apply any other values. Recommended value for width is, it can be left blank or 100%.

vii. Table Relative Width: User can set the width of the table relative to the width of entire screen i.e. Width of the Table will be calculated as % of the total width of the screen. User can enter only decimal values and value

1.0 represents the entire screen width. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

viii. Table Relative Height: User can set the height of the table relative to the height entire screen i.e. Height of the Table will be calculated as % of the total height of the screen. User can enter only decimal values and value 1.0 represents the entire screen height. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

ix. Table Relative Position X: User can set the X-Coordinate of the table relative to the entire screen i.e. X-Coordinate of the Table will be calculated as % of the total width of the screen. User can enter only decimal values. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

x. Table Relative Position Y: User can set the Y-Coordinate of the table relative to the entire screen i.e. Y-Coordinate of the Table will be calculated as % of the total height of the screen. User can enter only decimal values. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

xi. Table Caption: User can set the caption of the HTML table by entering the text in the text box against this field. User can enter only XSL keyword (starting with K_) in this text box. Tool will not allow the user to apply values not in the format of XSL keyword. A lookup icon is available beside the textbox, clicking this lookup image will open a dialog box where user can search available XSL keywords for a particular text, for more details please refer to section "*XSL Keywords Lookup*".

xii. Caption Class: User can set the class property of the Table Caption. User can only enter the CSS classes which are available in this tool. List of all CSS classes is loaded in the bottom-right panel of this tool under "CSS Classes" tab. User can click on the desired class and the value will be automatically populated in this text box. For more details refer to section "*CSS Classes*" under the heading "*JavaScript Functions and CSS Panel*". Tool will not allow the user to apply any other CSS class name other than those available in this tool. There is no recommended value for this field, it can be left blank.

xiii. Legend Label: User can create a legend for the HTML Table by entering the text in the text box against this field. User can enter only XSL keyword (starting with K_) in this text box. Tool will not allow the user to apply values not in the format of XSL keyword. Legends are

recommended only for inner HTML tables. A lookup icon is available beside the textbox, clicking this lookup image will open a dialog box where user can search available XSL keywords for a particular text, for more details please refer to section "*XSL Keywords Lookup*".

xiv. Legend Class: User can set the class property of the table legend. User can only enter the CSS classes which are available in this tool. List of all CSS classes is loaded in the bottom-right panel of this tool under "CSS Classes" tab. User can click on the desired class and the value will be automatically populated in this text box. For more details refer to section "*CSS Classes*" under the heading "*JavaScript Functions and CSS Panel*". Tool will not allow the user to apply any other CSS class name other than those available in this tool. Recommended value for this field is "labeltext" or it can be left blank.

xv. Default Hidden: This check box can be checked if the user wants to make this HTML table default hidden when the screen output html page loads for the first time. This feature is recommended where certain HTML table in the form needs to be kept hidden and displayed only on some JavaScript event.

xvi. Table Sequence: User can set the sequence number of the HTML table by entering a numeric value in the text box against this field. User can enter only numeric value > 0. This property helps the user to rearrange the order in which the HTML Tables will appear in the form. For e.g., if the current sequence number of the table is 3, it will appear as third HTML table in the form.

xvii. Customized Template ID: User can use already existing custom templates available as custom datatypes in this panel. User can select any custom template available in the dropdown list. Use this feature only when you want to use an existing HTML Table layout available, else leave this field blank. User can refer to the data types available in the HTMLDatatypes panel grouped under the "Custom" tab. User can click on the data type button and value will get selected in the drop down against this field. Refer to "*Custom*" section under heading "*HTML DataTypes Panel*" for more details on the usage of Customized Template ID.

xviii. Condition Field: User can enter the text in this text box against this field. This feature is only supported in mobile banking. It's a XSL expression which return either true or false on basis of this application decides that this table will render or not.

Two buttons are there in this panel,

i. Apply: Once the user has entered the above details, they need to click on "Apply" button to apply the changes done. If not clicked the changes will not get reflected on revisiting this panel.

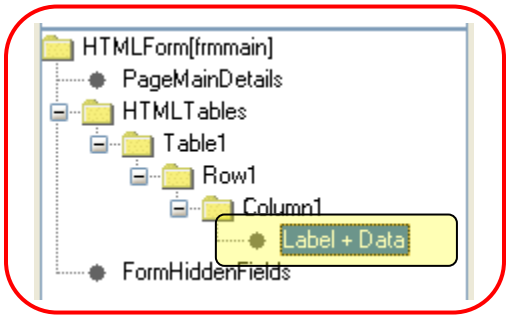
ii. Revert: If the user has not clicked on "Apply" and wants to go back to last applied changes, they can click on "Revert" button.

13.4. Component Details Panel

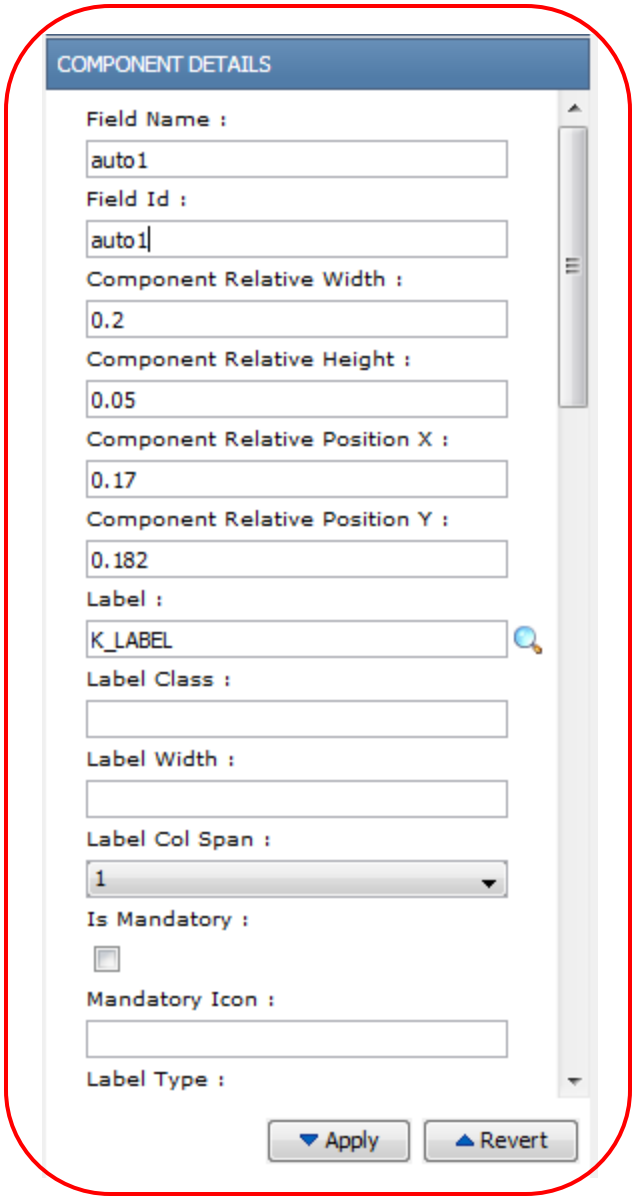
The user can select any of the following nodes

a. Label+Data
b. Data
c. Label
d. Label+Inner Table
e. Inner Table

of HTML Layout Panel, as shown below.



Clicking this node will display “HTML Table Data Details” panel, as shown below. In this panel user will get the fields to configure the HTML components of the particular table column. User can configure properties like fieldname, fieldid, labels and their style, HTMLDataType and several other properties listed below.



As shown in the screenshot above user can configure following properties of the column components

i. Field Name: User has to enter a field name. It is a mandatory field and cannot be left blank. It is always recommended to use "fld" as prefix with the field name, for e.g. fldbeneficiary. User can enter an alphanumeric value for Name.

ii. Field Id: User has to enter a field id. It is a mandatory field and cannot be left blank. This field id has to be unique, user cannot use any field id which has been used while defining the screen. It is always recommended to use "fld" as prefix with the field id, for e.g. fldbeneficiaryid. User can enter an alphanumeric value for Id.

iii. Component Relative Width: User can set the width of the component's relative to the width of entire screen i.e. Width of the component will be calculated as % of the total width of the screen. User can enter only decimal values and value 1.0 represents the entire screen width. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

iv. Component Relative Height: User can set the height of the Component relative to the height entire screen i.e. Height of the Component will be calculated as % of the total height of the screen. User can enter only decimal values and value 1.0 represents the entire screen height. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

v. Component Relative Position X: User can set the X-Coordinate of the Component relative to the table in which it is enclosed i.e. X-Coordinate of the Component will be calculated as % of the total width of the table. User can enter only decimal values. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

vi. Component Relative Position Y: User can set the Y-Coordinate of the Component relative to the table in which it is enclosed i.e. Y-Coordinate of the Component will be calculated as % of the total height of the table. User can enter only decimal values. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

vii. Label: User can enter XSL keyword (starting with K_) in the text box against this field. This text will appear as a label in the first cell of the column. Tool will not allow the user to apply values not in the format of XSL keyword. A lookup icon is available beside the textbox, clicking this lookup image will open a dialog box where user can search available XSL keywords for a particular text, for more details please refer to section "*XSL Keywords Lookup*".

viii. Label Class: User can enter the css class to be applied on the text entered against label. User can only enter the CSS classes which are available in this tool. List of all CSS classes is loaded in the bottom-right panel of this tool under "CSS Classes" tab. User can click on the desired class and the value will be automatically populated in this text box. For more details refer to section "*CSS Classes*" under the heading "*JavaScript Functions and CSS Panel*". Tool will not allow the user to apply any other CSS class name other than those available in this tool. Recommended CSS classes for labels are

a. labeltext: Use this class to keep the label text right aligned.
b. col1: Use this class for giving a width of 20% to the cell containing the label.
c. col2: Use this class for giving a width of 30% to the cell containing the label.

d. col3: Use this class for giving a width of 20% to the cell containing the label.
e. col4: Use this class for giving a width of 30% to the cell containing the label.
f. col80: Use this class for giving a width of 80% to the cell containing the label.

One can use a combination of these classes by giving a space between the classes, for e.g. "labeltext col1" will make the label right aligned with a width of 20%.

ix. Label Width: User can set the width property of the cell containing the label text by entering the text in the text box against this field. User can enter numeric value with "px" or "%" as postfix. Tool will not allow the user to apply any other values. It is recommended to use the CSS classes, as discussed above, to handle the width and leave this field blank. If the above recommended classes do not match your width requirement, then use this field to set the width.

x. Label Col Span: User can select a numeric value (>0) available in this drop down list against this field. This value will determine the number of columns this cell, containing the label, will span.

xi. Is Mandatory: User has to check this check-box, if they want to append the mandatory icon, discussed below, to the label of this field.

xii. Mandatory Icon: User can enter the key-board special characters like "**" to mark this field as mandatory/or can append this icon to the label for any disclaimer note.

xiii. Label Type: User can select the label types available in this drop down list. User can refer to the label types available in the HTMLDatatypes panel grouped under the "Label" tab. User can click on the data type button and value will get selected in the drop down against this field. For more details refer to "*Label*" section under heading "*HTML DataTypes Panel*".

xiv. Label Row Span: User can select a numeric value (>0) available in this drop down list against this field. This value will determine the number of rows this cell, containing the label, will span.

xv. Data Class: For details refer to Label Class. This CSS class will be applied on the column cell containing the HTMLDataType like textbox, hence no need to use labeltext CSS class.

xvi. Data Width: For details refer to Label Width. This width will be applied on the column cell containing the HTMLDataType like textbox.

xvii. Data Col Span: User can select a numeric value (>0) available in this drop down list against this field. This value will determine the number of columns this cell, containing the HTMLDataType, will span.

xviii. Data Type: User can select the data type available in this drop down list. User can select the HTMLDataType to be displayed in this cell of the column. User can refer to the data types available in the HTMLDatatypes panel grouped under the "Data" tab. User can click on the data type button and value will get selected in the drop down against this field. For more details refer to "*Data*" section under heading "*HTML DataTypes Panel*". For Inner table component this field is read only.

xix. Node Value: User can use text box against this field to populate data in different HTMLDataType. For e.g. user can enter a XPATH or a constant value (string'<value>') in textbox HTMLDataType to populate the textbox with a default value. Similarly user can use this field to populate list of values in drop down. Refer to "*Oracle_FLEXCUBE_Direct_Banking_LEAP_Framework_DeveloperGuide.docx* " for more details on the format of nodevalue for different HTMLDatatypes. For Inner table component this field is read only and contains the table id of child table.

xx. Function Name: User can give the JavaScript function name which they want to attach to the main event of the HTMLComponent, for e.g. giving a function name for dropdown will assign this function to onchange event of the dropdown. The logic of the function has to be written in the corresponding idrequest.js file.

xxi. Function Arguments: User can give comma separated arguments to the JavaScript function, discussed above. User can pass only constant values using this field. To pass dynamic values as an argument using XPATH's, please refer to "Dynamic Function Arguments" below.

xxii. Input Size: This field is used for different datatypes with different purposes.

a. Textbox: Size of the textbox.
b. Textarea: Rows value of the textarea.
c. Dropdowns: Making a dropdown into a list.
d. Radio Button: Number of radio buttons.
e. Tabs: Number of tabs.
f. Image: Height of the image.
g. Buttons: Size of the button.

xxiii. Input Maximum Length: This field is used for different datatypes with different purposes.

a. Textbox: Maximum Length of the Textbox.

b. Textarea: Columns value of textarea.
c. Image: Width of the image.
d. Buttons: Maximum Length of the buttons.

xxiv. IS UDF Field: This is a check-box. If checked, a name-value pair of this field can be assigned to UDFDTO and passed to the service.

xxv. Alternate Text: User can enter XSL keyword (starting with K_) in the text box against this field. This text will appear as a tooltip for the HTMLDataType. Tool will not allow the user to apply values not in the format of XSL keyword. A lookup icon is available beside the textbox, clicking this lookup image will open a dialog box where user can search available XSL keywords for a particular text, for more details please refer to section "*XSL Keywords Lookup*".

For dynamic tooltip user can enter XPATH in this field. For combination of static and dynamic tooltip one should use "^" as the delimiter, so for an input like "*K_ACCOUNT^K_BLANK^K_IS^K_BLANK^/faml/response/preparerepaymentiqresponsedto/custloanaccounts/customeraccountdto/accounts/accountnodto[position()=$rowiteration]/nbraccount*", tooltip will appear as "Account is (evaluated account no)".

xxvi. Image URL: User can enter relative path of the image file to be displayed. Recommended to be used when one is using "Image" or "Text Box with Lookup" as HTMLDataType. Relative path of commonly used images are

a. images/eng/calander.gif: Use this path for a calendar image.
b. images/lookup.gif: Use this path for lookup image.

xxvii. Default Static Label: User can enter XSL keyword (starting with K_) in the text box against this field. This field is used for different datatypes with different purposes.

a. Drop Downs: First static option K_SELECT in the dropdown list, if required.
b. Static drop downs: "~" separated static options to be displayed in the dropdown list.
c. Radio buttons: "~" separated values to be assigned as label to each radio button.
d. Image: Defines the value of alt attribute of image.
e. Buttons: Label to be rendered on button.
f. Anchors: Text to be rendered as link.

A lookup icon is available beside the textbox, clicking this lookup image will open a dialog box where user can search available XSL keywords for a particular text, for more details please refer to section "*XSL Keywords Lookup*".

xxviii. Is Readonly: User can check this check-box to make the field read-only. For e.g. a read-only text box.

xxix. Tab Index: User can enter a numeric value (>0) as tab index for this field.

xxx. Default Value: This field is used for different datatypes with different purposes.

a. Textarea: Textarea wrap attribute value.
b. Dropdowns: Default selected value of the dropdown. One can give XPATH or a constant value like string('<value>').
c. Radio Buttons: This value will determine which radio button will be default selected.
d. Check Box: Check box "checked" attribute value.
e. Image: Image align attribute value.
f. Anchor: Anchor target attribute value.

xxxi. Data Row Span: User can select a numeric value (>0) available in this drop down list against this field. This value will determine the number of rows this cell, containing the HTMLDataType, will span.

xxxii. Dynamic Function Arguments: User can give comma separated XPATH's as arguments to the JavaScript function. These XPATH will be evaluated by the framework and their value will always come after the constant function arguments, discussed above in "Function Arguments". For e.g. if function name is "*fnViewLoanDetails*", function arguments are "*'LDT', '01'*" and dynamic function arguments are "*//faml/response/prepareloandetailsresponsedto/custaccounts/customeraccountdto[1]/accounts/accountnodto[position()=$rowiteration]/nbraccount,//faml/response/prepareloandetailsresponsedto/custaccounts/customeraccountdto[1]/accounts/accountnodto[position()=$rowiteration]/nbrbranch*", then function fnViewLoanDetails will have arguments as ('LDT', '01', 'evaluated account no', 'evaluated branch').

xxxiii. Condition Field: User can enter the text in this text box against condition field. This feature is only supported in mobile banking. It's a XSL expression which return either true or false on basis of this application decides that this html component will render or not.

xxxiv. Step Number: User can set the step number of the screen on which this component has to be shown. Currently this field is used for user agents of mobile

applications. This field is only available for user agents (Ipad, Iphone, Android Tab and Android Phone) of Mobile Application channel.

xxxv. Token 1-5: These fields i.e. Token1, Token2, Token3, Token4, Token5 are configuration fields that are used for different datatypes for different purposes.

xxxvi. Row Iteration: User can enter the XPATH on which they can iterate to create multiple rows dynamically or to create an array of Hidden fields. For more details refer to "*Oracle_FLEXCUBE_Direct_Banking_LEAP_Framework_DeveloperGuide.docx*".

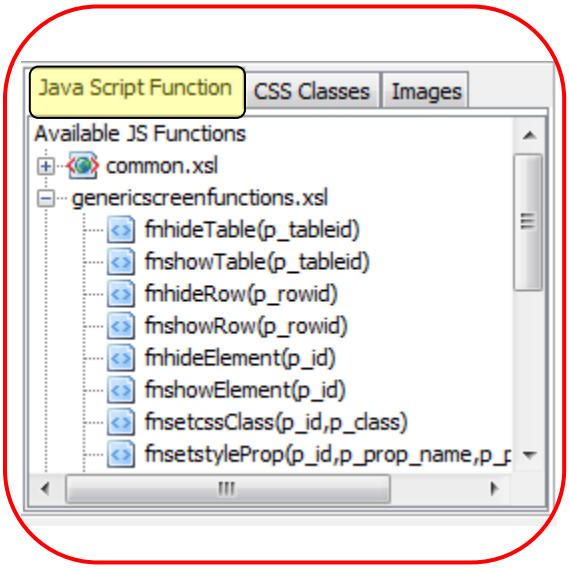
Two buttons are there in this panel.

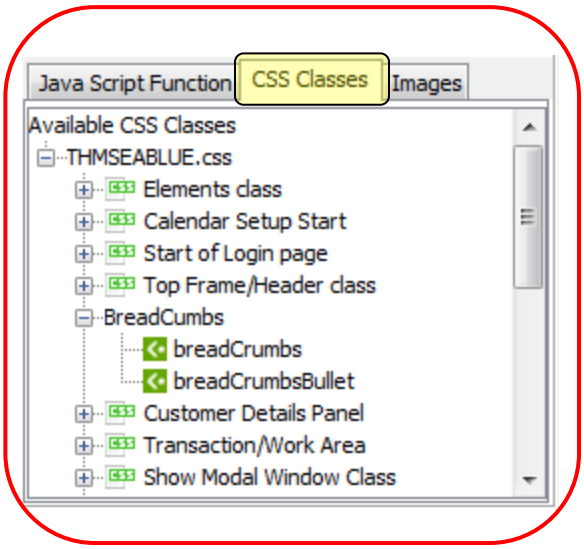
i. Apply: Once the user has entered the above details, they need to click on "Apply" button to apply the changes done. If not clicked the changes will not get reflected on revisiting this panel.

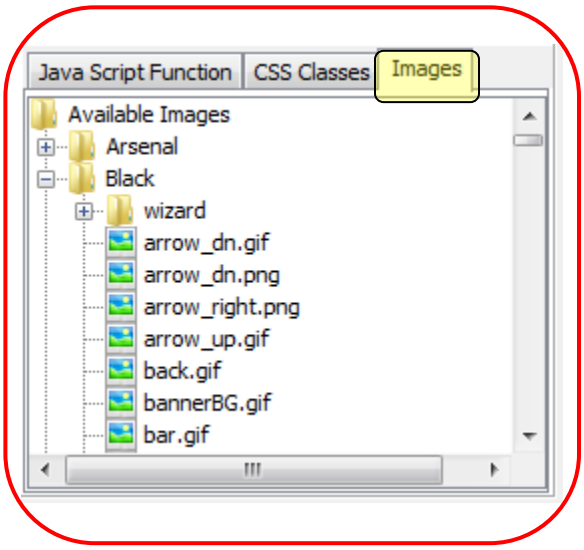
ii. Revert: If the user has not clicked on "Apply" and wants to go back to last applied changes, they can click on "Revert" button.

14. JavaScript Functions, CSS and Images Panel

This panel is at bottom-right of this tool. This panel contains the common JavaScript functions and the list of all CSS classes and images available in LEAP framework as shown below.

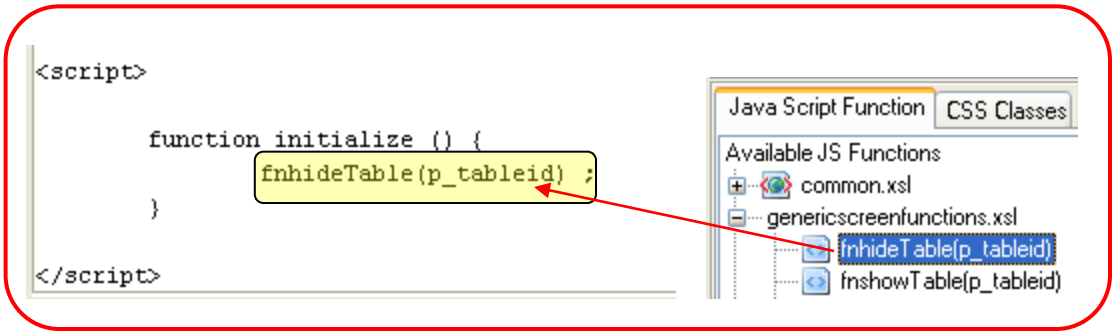




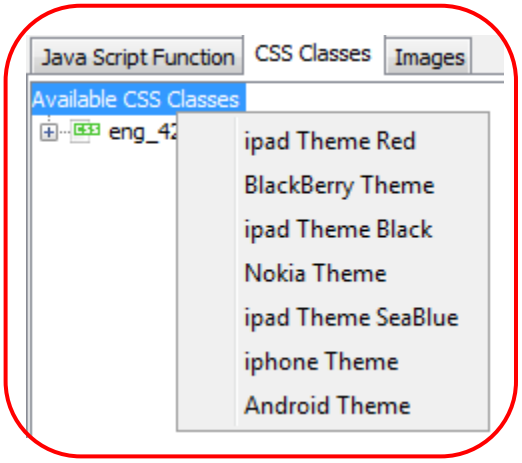


This panel has three tabs as highlighted above

1. Java Script Function: Clicking this tab loads the panel with xsl's which are part of LEAP framework and the common JavaScript function's defined in these xsl's in a tree layout format. User can call these functions directly in their corresponding idrequest.js file to achieve certain functionality. For e.g. user can use the function "fnhideTable" defined in "genericscreenfunctions.xsl" to hide any particular table by calling this function from their corresponding idrequest.js file. User just needs to click this function and the function will be targeted to current cursor position in the idrequest.js file.



2. CSS Classes: Clicking this tab loads the panel with all the CSS classes available in a particular theme corresponding to the selected project, in a tree layout format. Themes can be switched by right clicking on the "Available CSS Classes". Based on the selected project the available themes will be displayed in the popup menu as shown below:



Each theme folder has its own <themename>.css file. For e.g. "THMRED" has "THMRED.css" file. Currently in CSS file of the respective themes the classes have been grouped using a delimiter "/*--" to identify their usage.

This tool extracts the preview folder from "LEAP.jar" in the user's current directory and loads the theme name from the "\preview\css\default.css" file. If a user wants to change the theme name they can edit this file and change the relative url of a different CSS file to be applied. For e.g. if a user prefer to work on screen design using CSS classes of theme blue, they need to change the following import statement of "\preview\css\default.css" and restart the tool.

@import url(THMRED/THMRED.css');

to

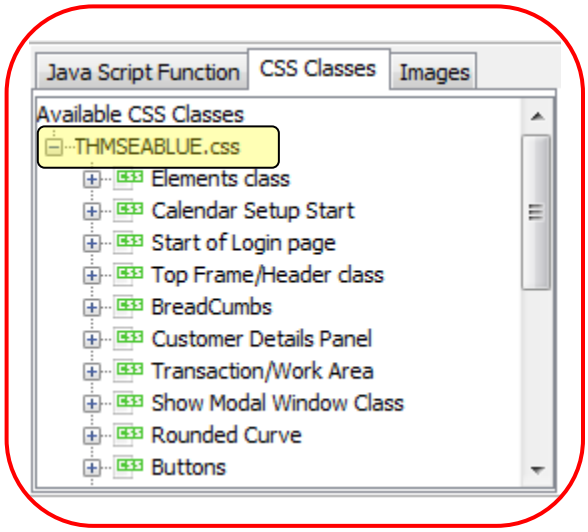
@import url(THMSEABLUE/THMSEABLUE.css');

@import url(THMRED/jquery-ui.css');

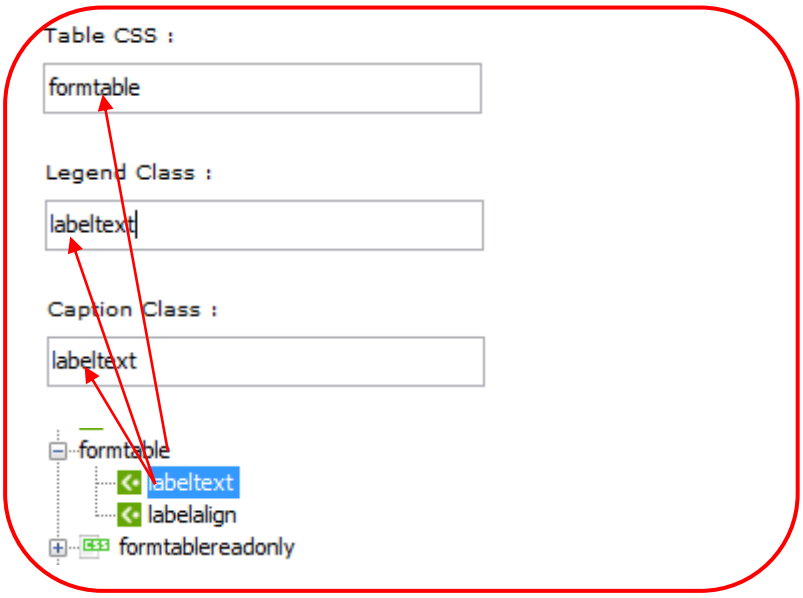
to

@import url(THMSEABLUE/jquery-ui.css');

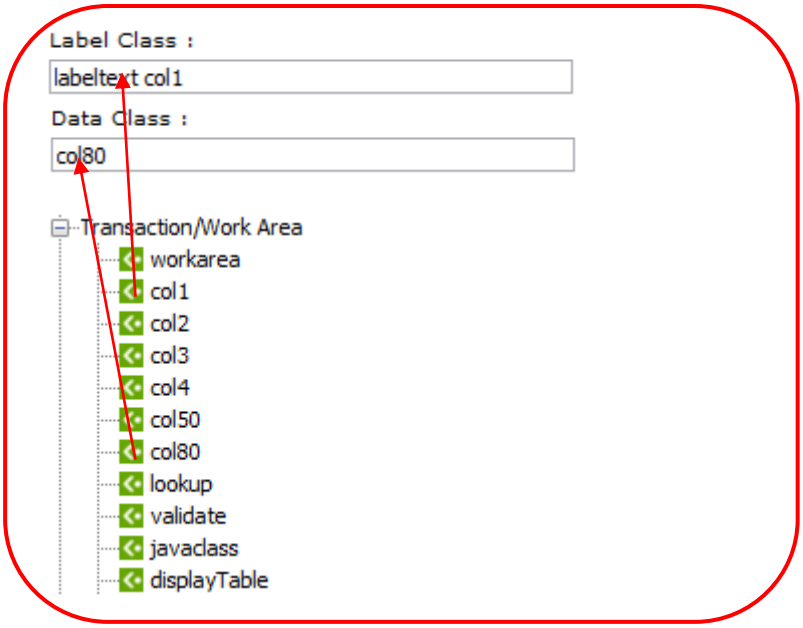
After restarting the tool the CSS file name will get changed as shown below.



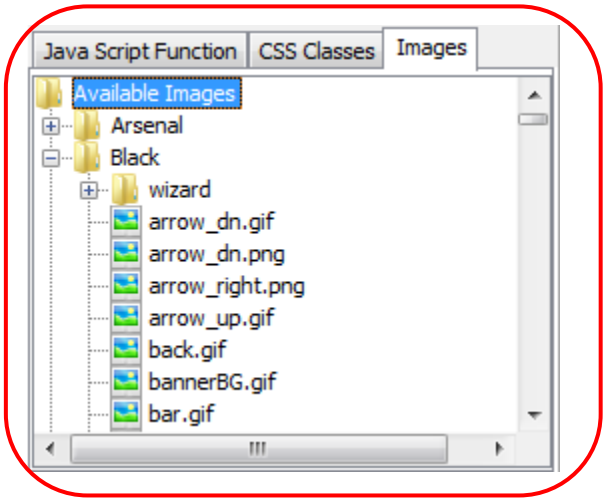
User can use these CSS classes by just clicking them. These CSS classes are linked to all those textboxes in the work area panel where user is supposed to use CSS class name. For e.g. while defining the HTML Table details user can click on these CSS classes to set the value of following fields.



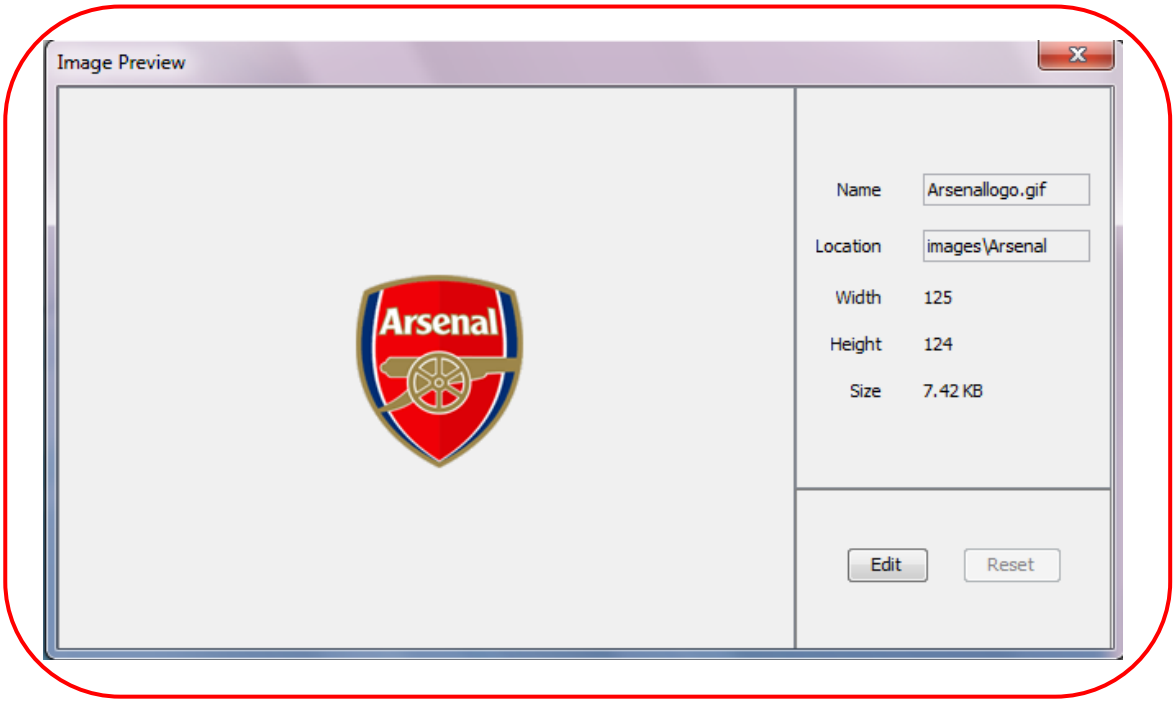
Similarly these CSS classes can be used while defining the HTML column components, user can click on these CSS classes to set the value of following fields.



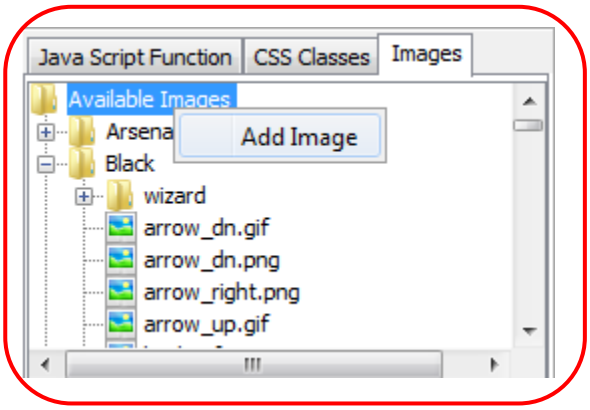
3. Images: Clicking this tab loads the panel with all the available images in a tree layout format as shown below:



Double clicking on the image nodes opens a dialog as shown below, where the user can view the image and its properties and edit the image.



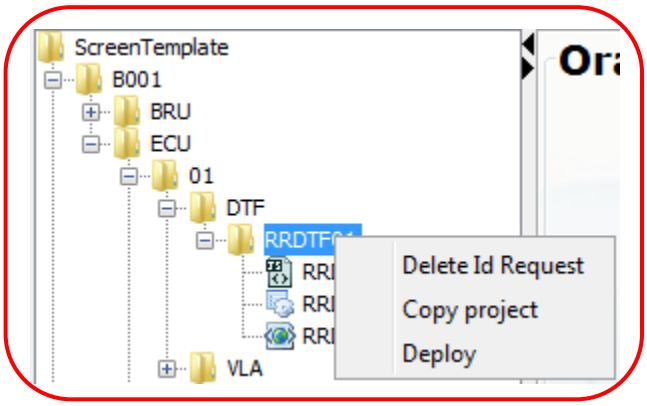
Images can be added by right clicking on the “Available Images”. Based on the selected project the available themes will be displayed in the popup menu as shown below:



15. Deleting/Copying/Deploying a Project

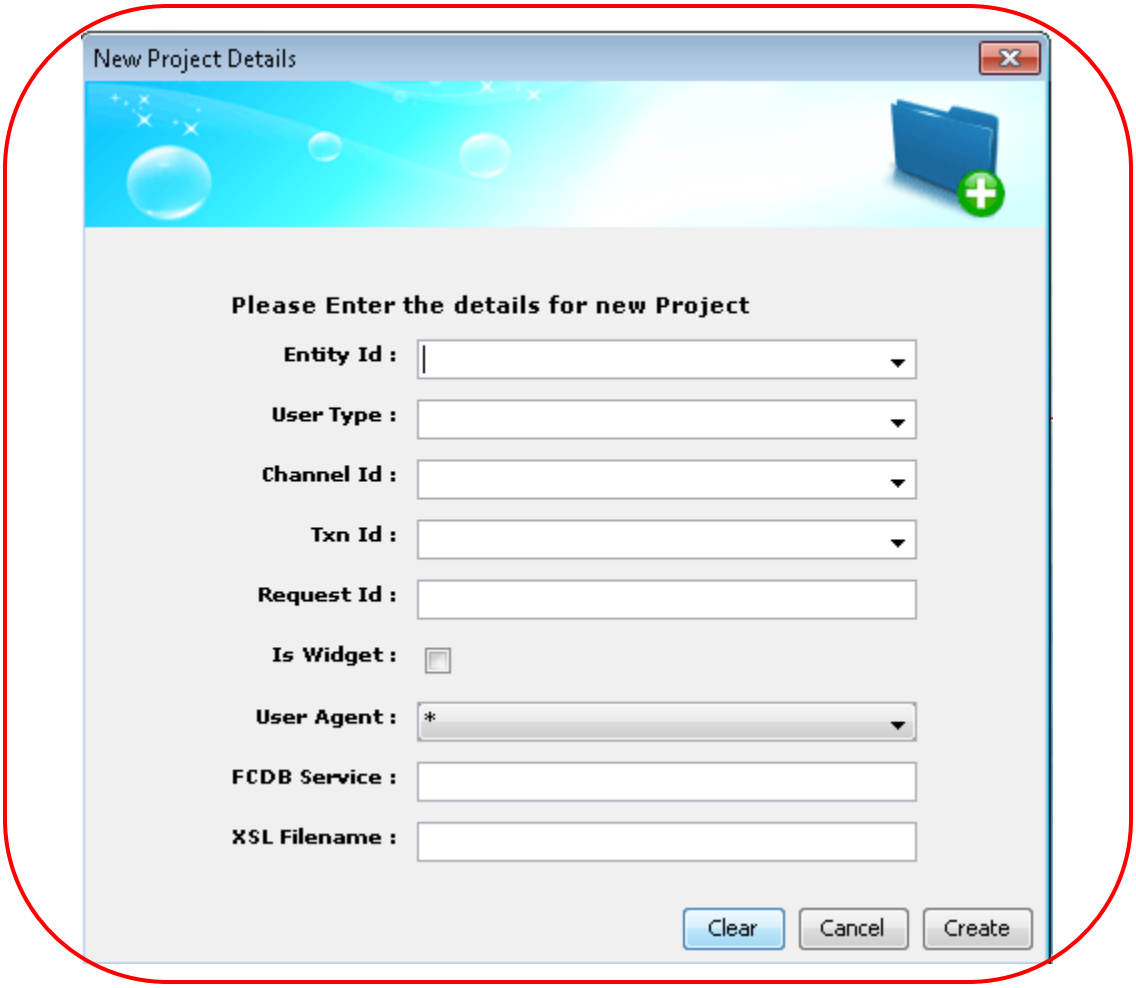
For deleting/copying/deploying a project from the workspace, right click on that idrequest node. A menu appears with the options

1. Delete IdRequest
2. Copy project
3. Deploy

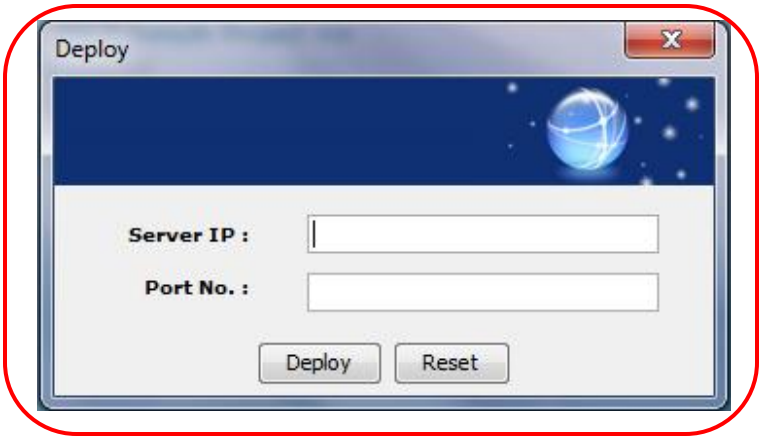


Clicking the “Delete Id Request” option will delete the selected IdRequest along with its three files (i.e. .js, .sql, .xml) from Txn Id tree node.

Clicking the “Copy project” option will open a dialog (as shown below) which will allow the user to enter the project details. On clicking the “Create” button the selected project will be copied.

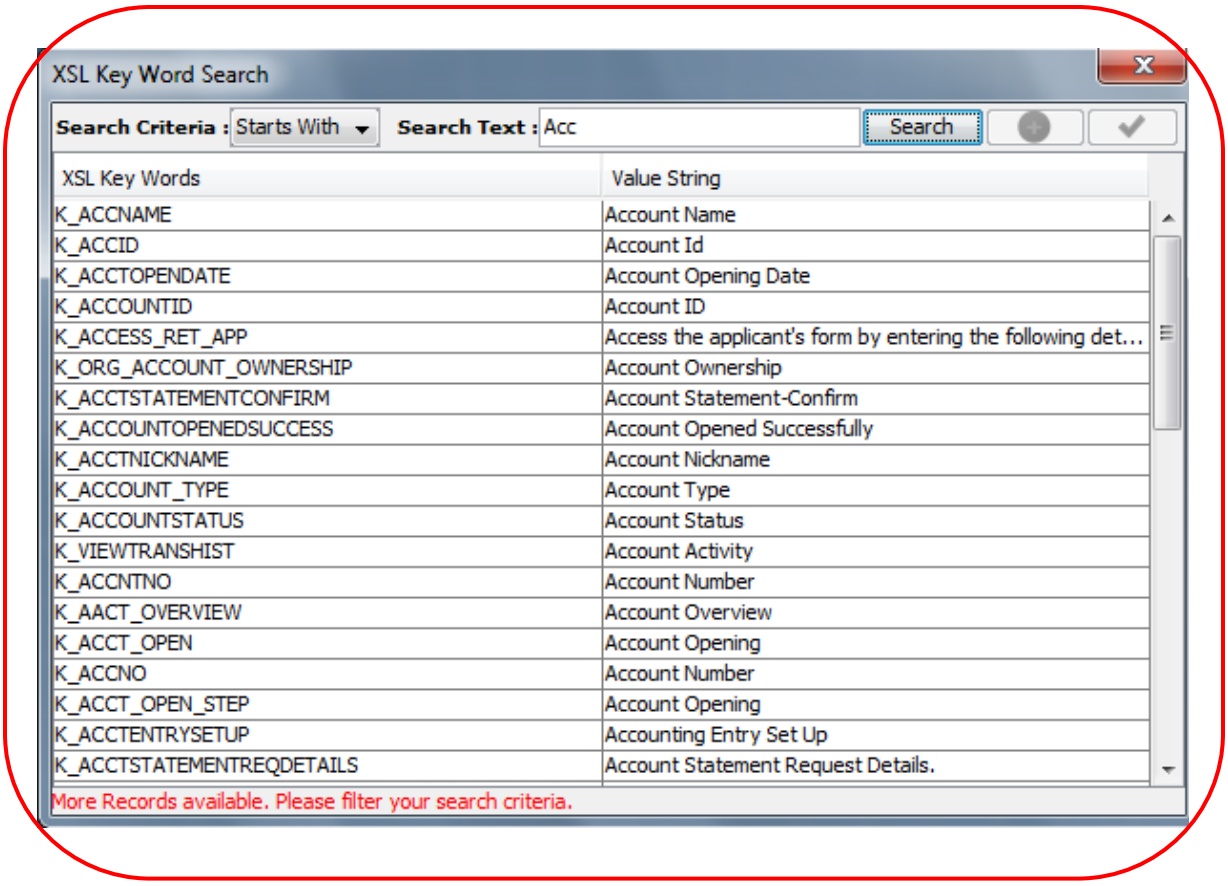


Clicking the “Deploy” option will open a dialog as shown below. Enter the Server IP and Port No. and click on “Deploy” to deploy the project on the given server.



16. XSL Keywords Lookup

User User can search the available XSL keywords for a particular text using the lookup icons available against the fields in the work area panels. These lookup icons are available only against those fields where user is supposed to use XSL keyword. Clicking on the lookup icon will open a search dialog as shown below.



This dialog box has following fields:

1. Search Criteria: This dropdown has following three values

   a. Starts With: Use this search criteria to search all XSL key word for text starting with the entered search text.

   b. Ends With: Use this search criteria to search all XSL key word for text ending with the entered search text.

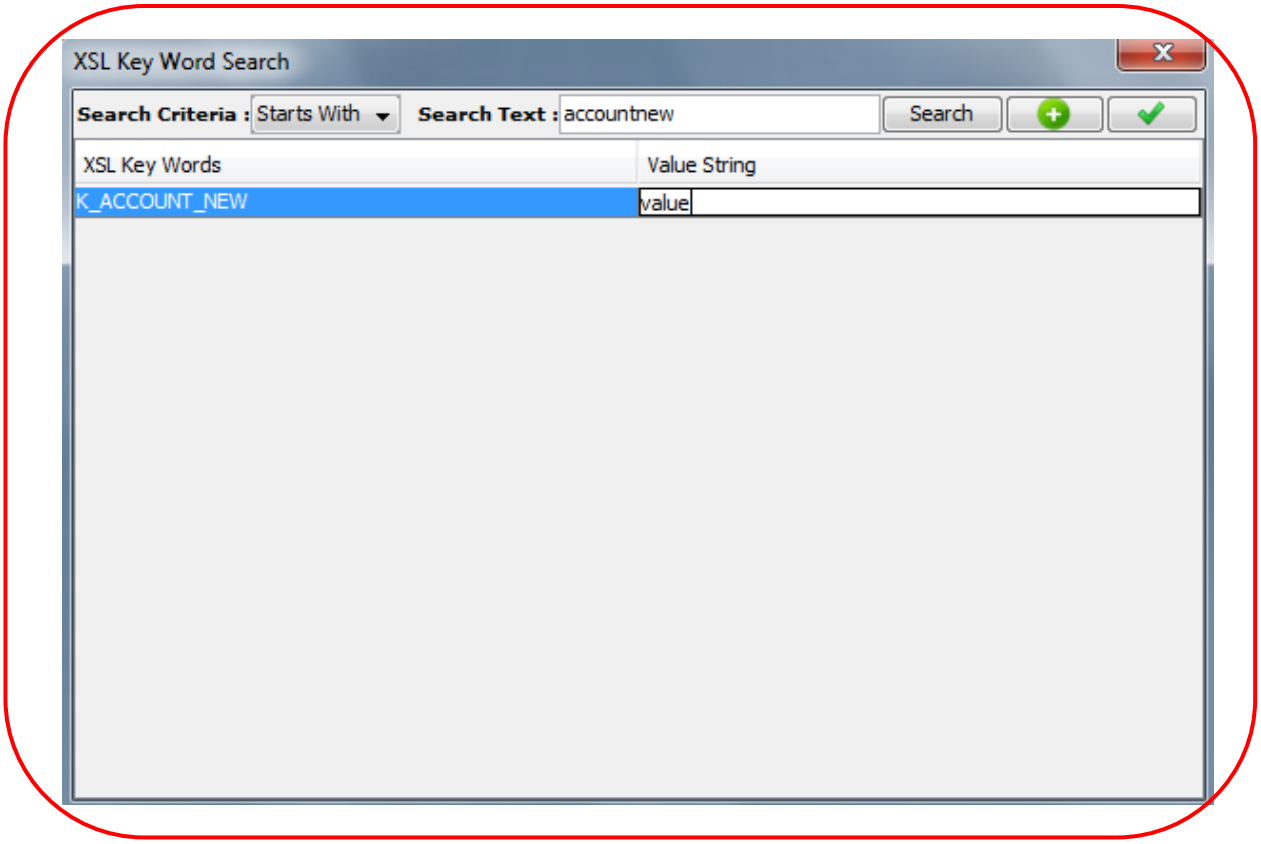
   c. Contains: Use this search criteria to search all XSL key word for text containing the entered search text.

2. Search Text: Use this text box to enter the text for which XSL key word is to be fetched.

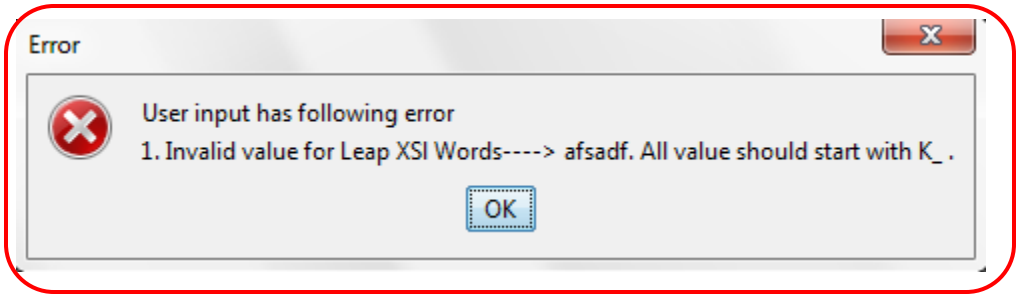
User needs to click on search button to search the XSL keywords. The search results will be available to user in the form of grid table. Currently the maximum search output is 50, if the search output results exceeds limit of 50 records, a message will be displayed in the bottom of the dialog, asking user to further filter the search criteria. User can select any XSL key word from the search output and double click on that to automatically target that XSL keyword into the desired text box field from where the lookup dialog has been opened.

16.1.1. Add New XSL Keyword

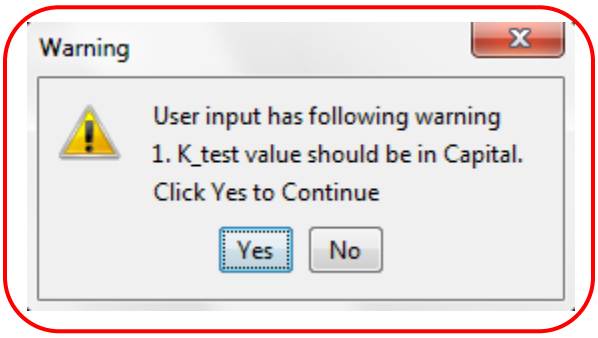
User can add new XSL keywords by clicking on the add button. Initially the add button is disabled. When the user searches for the XSL keyword which is not available, then the button gets enabled. Click on the button to add new keyword. Give the valid values for XSL keyword and Value String columns as shown below.



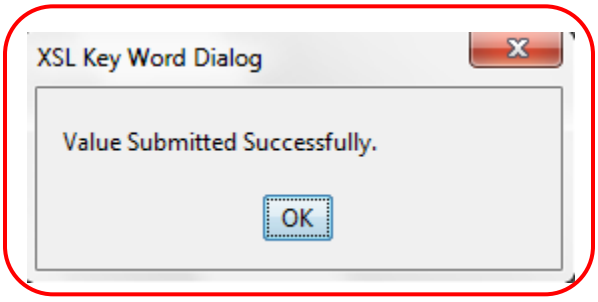
The XSL keyword should start with "K_" else it gives the following error.



If the XSL keyword is not in capital letters, then it gives a warning as shown below



If the new key word is successfully added , then the following dialog box appears. All the corresponding scripts are generated.

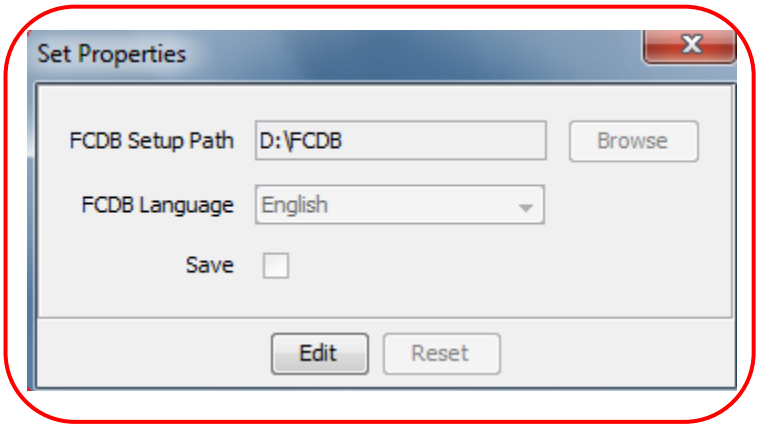


17. FCDB Properties Dialog

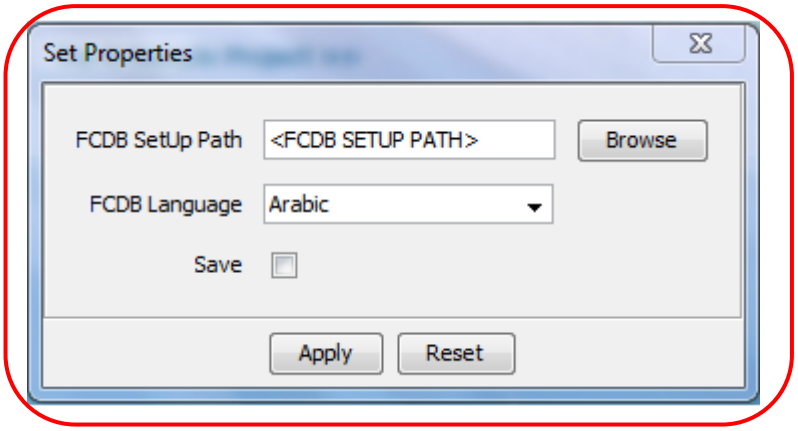
The FCDB properties can be edited or viewed by clicking on the following icon in the tool bar.

FCDB Properties dialog will allow you to enter the path of the FCDB folder and the language

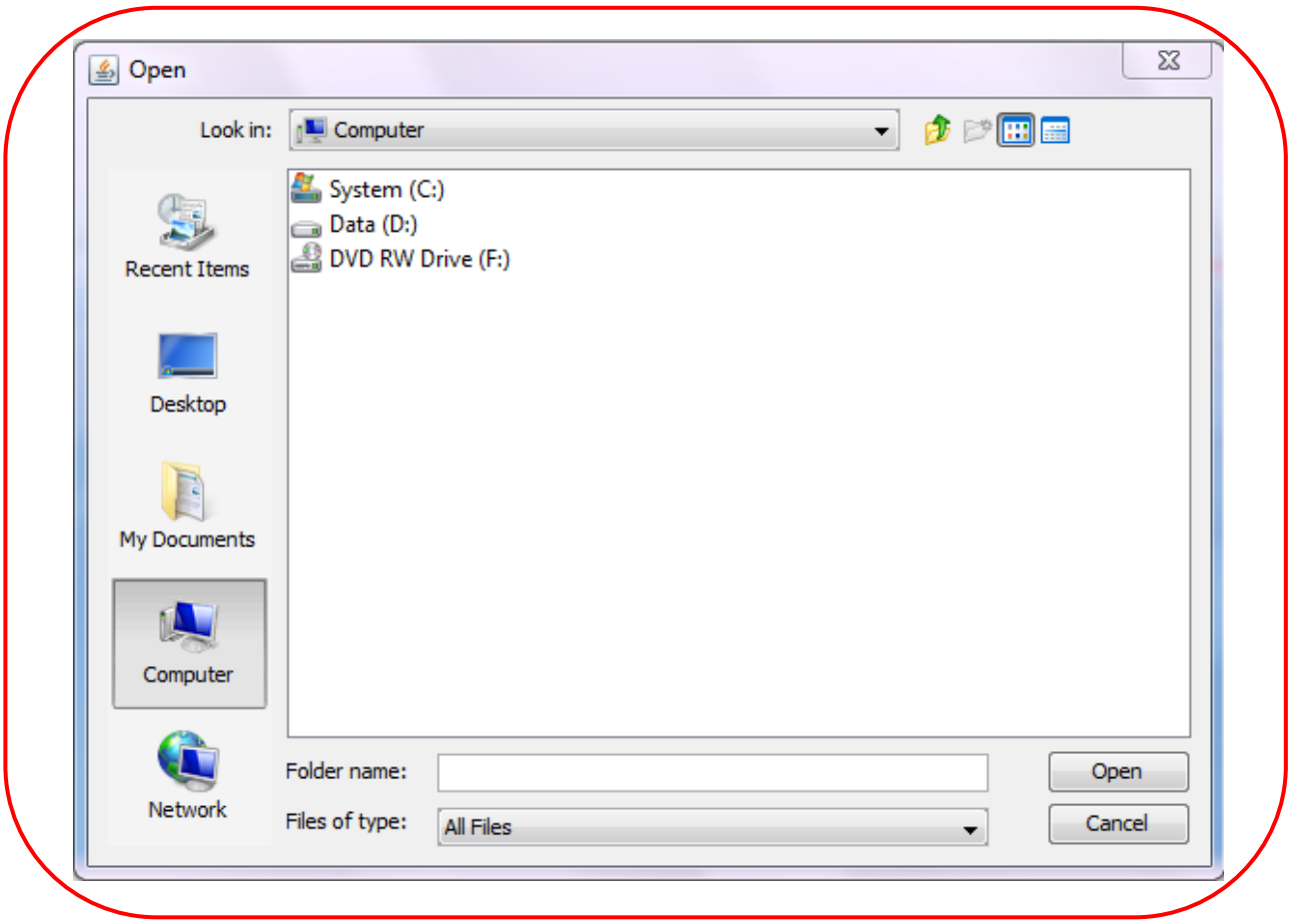
When you run the LEAP tool you will get a pop up at home screen asking for the FCDB Properties as shown below:



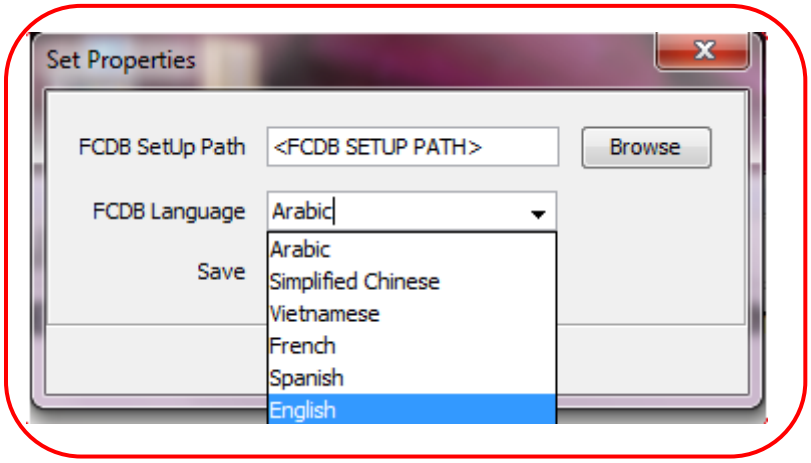
Click on the Edit button and user will be able to enter the details:



The FCDB home directory path can be browsed using browse button.



The language can be chosen from the available languages displayed in the drop down or by typing the first 3 letters of the language.



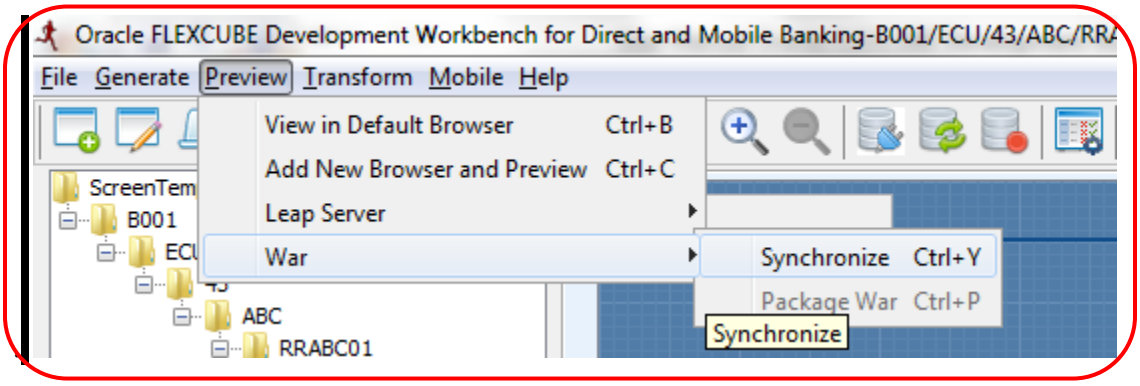
18. Synchronize/Package War

18.1.1. Creating WAR file using the updated CSS and image files

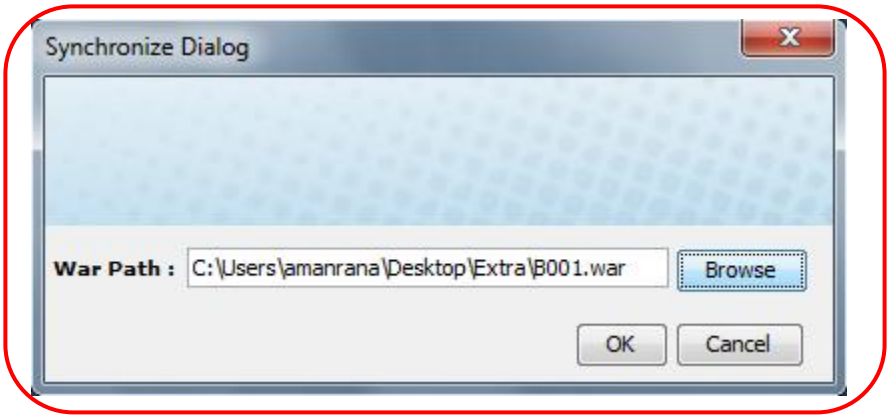
1. Synchronize WAR:

   To ensure that one have the latest CSS classes and Images loaded in LEAP tool before editing such web component one can synchronize the latest war in your worspace using this option.

   Select War option from the Preview Menu. A submenu will append, as shown below:



   Clicking on the synchronize a dialog box will open where we will enter the path of the war file to be synchronized as shown below:

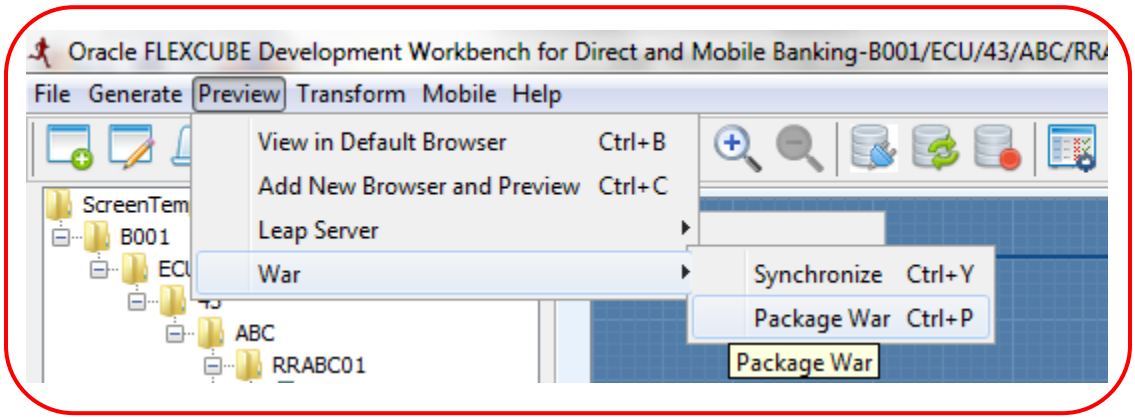


   Clicking the ok button will synchronize the selected war file into the workspace .

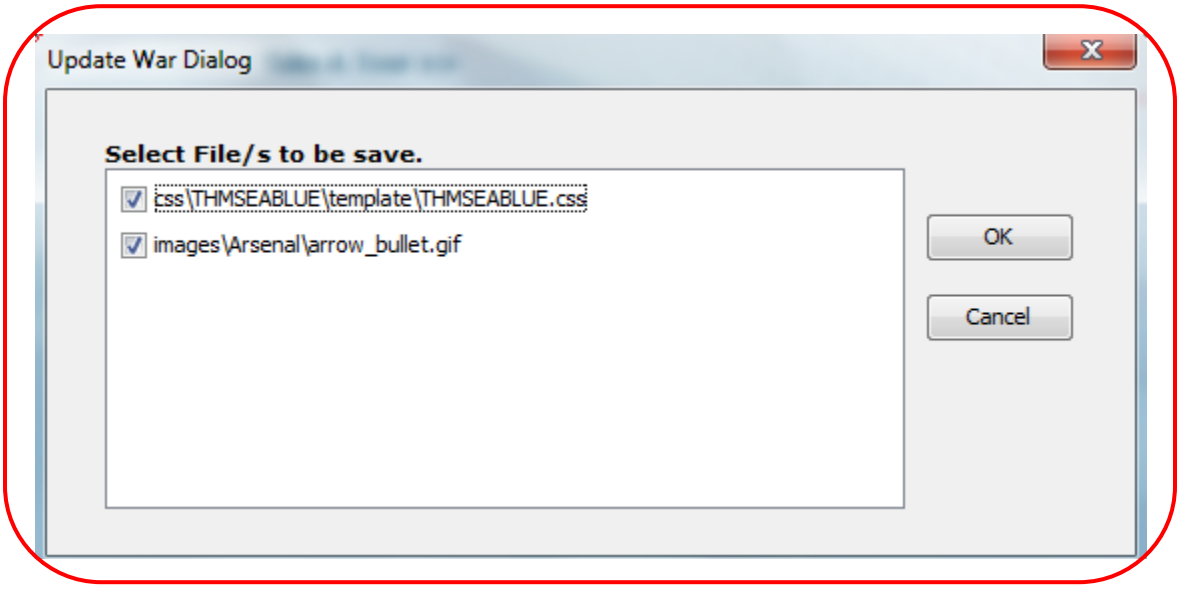
2. Package War:

Package war will help us to repackage the war with updated CSS and image files. Package War will be enabled only if any Web component like CSS or images has been edited by the user.

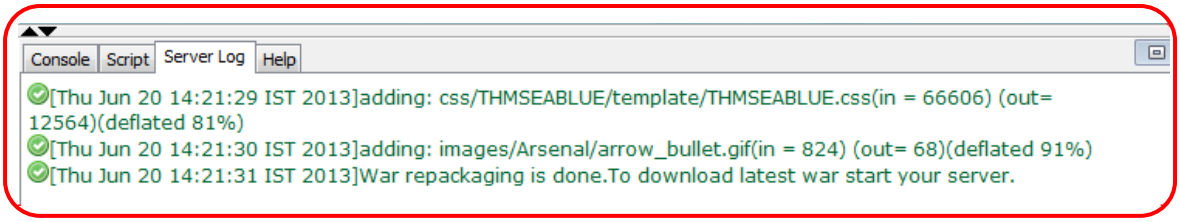
Select War option from the Preview Menu. A submenu will append, as shown below



Clicking on the package war a dialog with the updated flies will open which will let us choose the changes to be made in the war as shown below:

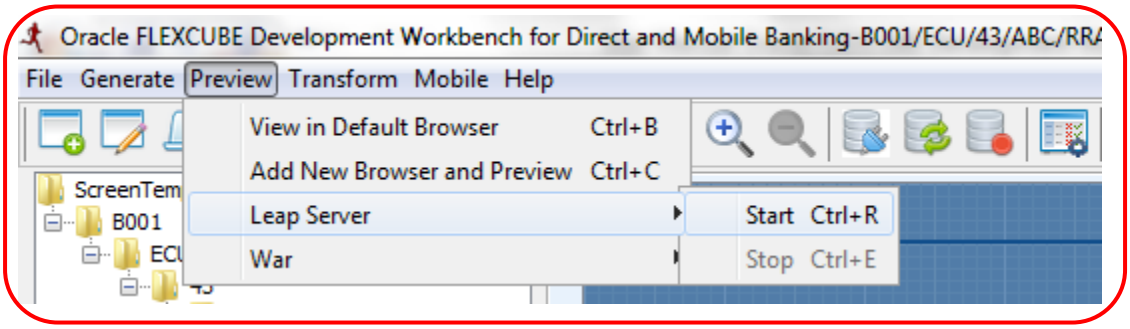


Clicking the ok button will package the war with the selected changes and logs will be generated for the same in server logs as shown below:

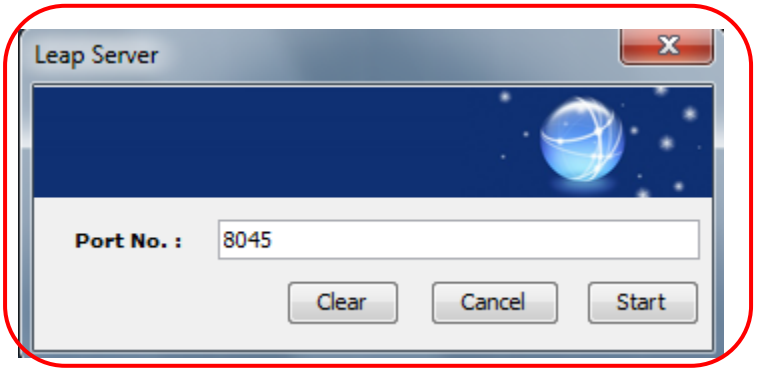


3. Download Latest WAR

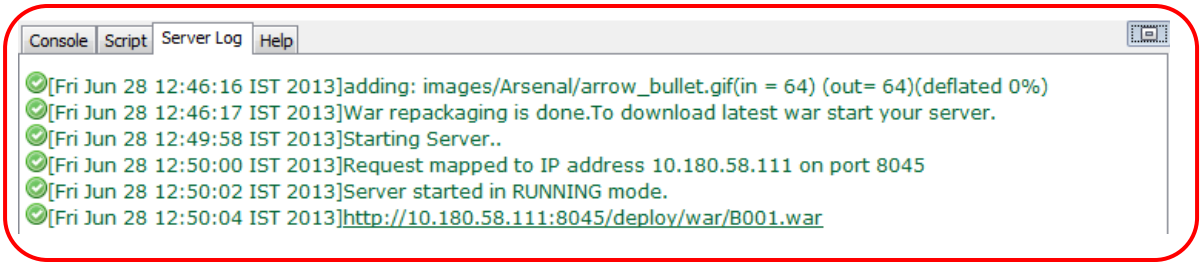
To download the latest packaged war.Start the server by selecting Leap Server option from Preview Menu. A submenu will append,as shown below:



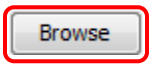
Clicking on start a dialog will open. Enter a valid port no.(i.e., it should be a 4 digited numeric value) as shown below:

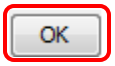


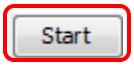
Clicking on the start button the server will start and a link will be provided in the server logs to download the packaged war as shown below:

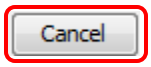


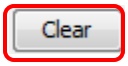
Description of Buttons

1. 

: Click this button and select browser .war location path.
2. 

: Click this button to synchronize the war.
3. 

: Click this button to start the server.
4. 

: Clcik this button to close the server dialog.
5. 

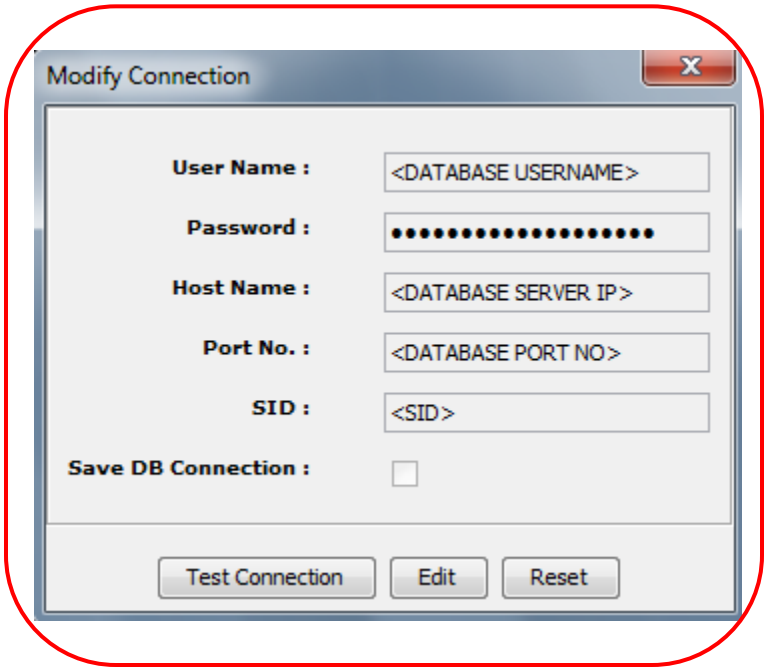
: Click this button to clear input fields.

19. Database Connectivity

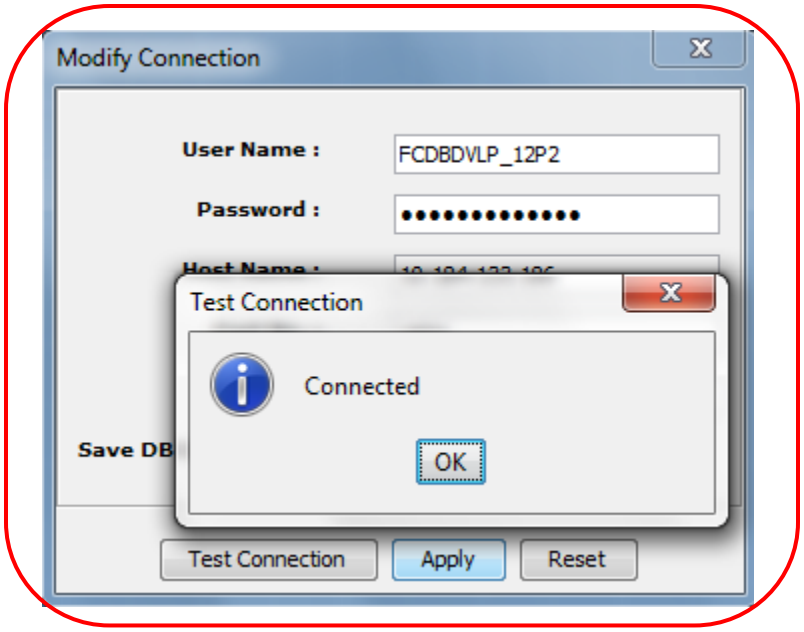
Database Connection provides a feature to connect the tool to a database schema,so as to give a option to the user to work while connected to the database schema or to work offline.

1. Connecting to the Database:

   When you run the LEAP tool you will get a pop up at home screen asking for the Database Connection details as shown below:



Click on the Edit button and user will be enabled to enter the details:



2. Description of Icons in the toolbar

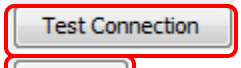
a.  **:** This icon depicts that tool is not connected to the database.Click this icon to start the database connection.

b.  : Click this icon to refresh the database connection and reload the changes done in database.

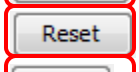
c.  **:** This icon depicts that tool is connected to the database.Click this icon to modify the database connection if required.

d.  **:** Click this icon to close the database connection.

3. Description of Buttons

a.  **:** Click this button to test the database connection.

b.  **:** Click this button to create a databse connection.

c.  **:** Click this button to reset the text fields.

d.  **:** Click this button to enter the value in text fields.