Deploying Custom page in Oracle Apps

Step 1: Move Files to Server

Step 2: Import the files to server

Step 3 : Creating Function

Step 4: Creating Menu

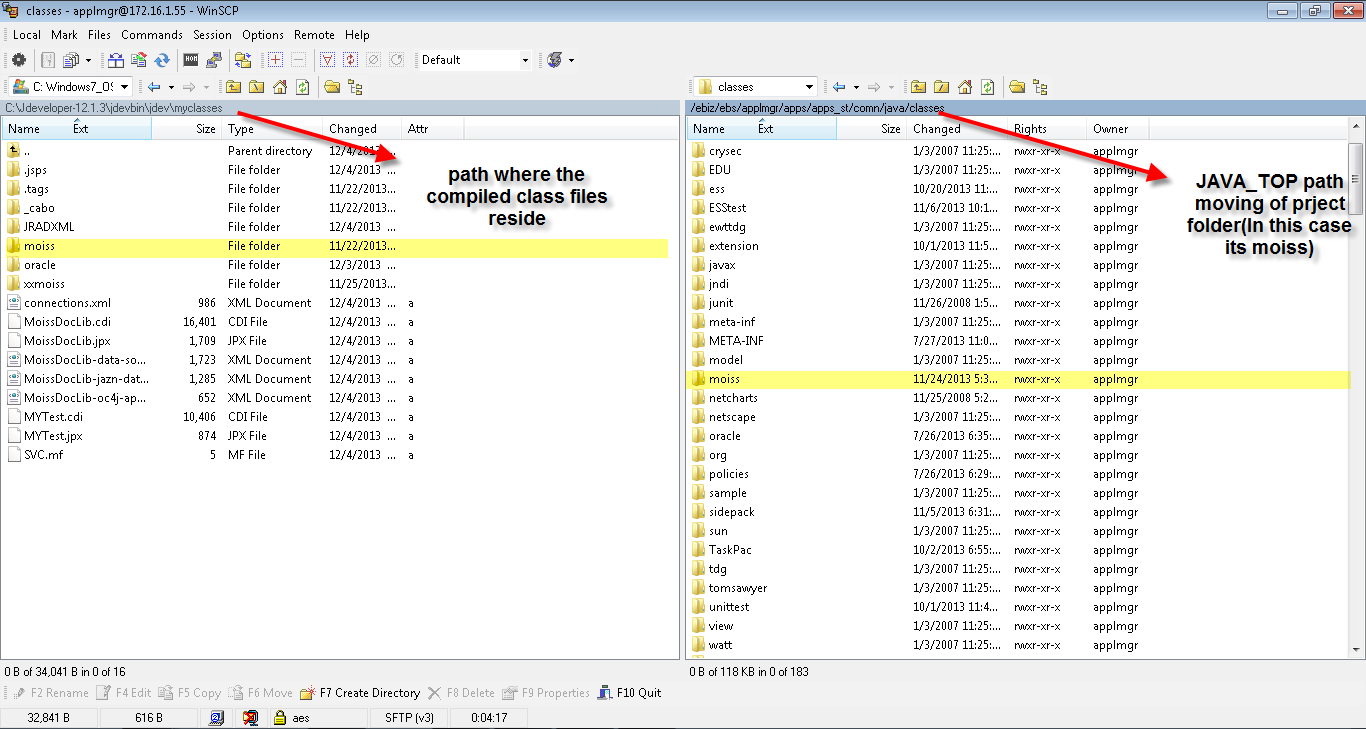
Step 5: Attaching Menu to Responsibility

Step 6: Attaching responsibility to User

Step 1: Move Files to Server

After Developing your OAF Page we have to deploy it to server. For this first we have to move the files to the server. Open your path and go to Jdeveloper-> jdevbin -> jdev -> myclasses -> moiss

Use a tool for file transfer like filezilla or winscp



Note : The source code resides in myprojects folder in Jdeveloper. But for deployment you have to go inside myclasses as the complied class files resides here.

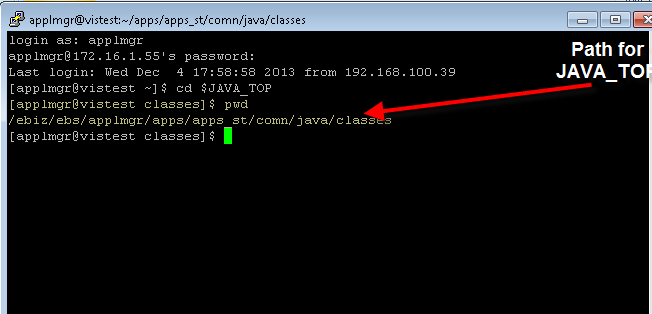
Below is a screenshot of file transfer in Winscp tool

You have to move you files to JAVA\_TOP in your server

You can find it by logging into server like a tool called putty or any other which has terminal. After logging to your server type

$cd JAVA\_TOP 🡪 which tells to navigate to JAVA\_TOP

classes $ pwd 🡪 which tells to print the directory



Step 2: Import the files to server

At the stage of moving your files your page and region just reside in your server. In order to make your pages work you have to import the files. Importing means saving the path of page files and region files in the database. As oracle is a database it needs to get registered that where the pages and the region reside. For importing you have to run the below script. You have to include the page path and region path for importing.

Here are the scripts for which I have my pages and regions

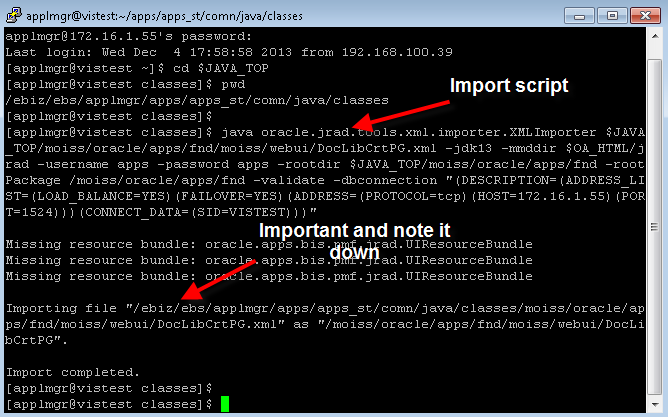
java oracle.jrad.tools.xml.importer.XMLImporter $JAVA\_TOP/moiss/oracle/apps/fnd/moiss/webui/DocLibCrtPG.xml -jdk13 -mmddir $OA\_HTML/jrad -username apps -password apps -rootdir $JAVA\_TOP/moiss/oracle/apps/fnd -rootPackage /moiss/oracle/apps/fnd -validate -dbconnection "(DESCRIPTION=(ADDRESS\_LIST=(LOAD\_BALANCE=YES)(FAILOVER=YES)(ADDRESS=(PROTOCOL=tcp)(HOST=172.16.1.55)(PORT=1524)))(CONNECT\_DATA=(SID=VISTEST)))"

java oracle.jrad.tools.xml.importer.XMLImporter $JAVA\_TOP/moiss/oracle/apps/fnd/moiss/webui/DocLibPG.xml -jdk13 -mmddir $OA\_HTML/jrad -username apps -password apps -rootdir $JAVA\_TOP/moiss/oracle/apps/fnd -rootPackage /moiss/oracle/apps/fnd -validate -dbconnection "(DESCRIPTION=(ADDRESS\_LIST=(LOAD\_BALANCE=YES)(FAILOVER=YES)(ADDRESS=(PROTOCOL=tcp)(HOST=172.16.1.55)(PORT=1524)))(CONNECT\_DATA=(SID=VISTEST)))"

java oracle.jrad.tools.xml.importer.XMLImporter $JAVA\_TOP/moiss/oracle/apps/fnd/moiss/webui/MoissCommRN.xml -jdk13 -mmddir $OA\_HTML/jrad -username apps -password apps -rootdir $JAVA\_TOP/moiss/oracle/apps/fnd -rootPackage /moiss/oracle/apps/fnd -validate -dbconnection "(DESCRIPTION=(ADDRESS\_LIST=(LOAD\_BALANCE=YES)(FAILOVER=YES)(ADDRESS=(PROTOCOL=tcp)(HOST=172.16.1.55)(PORT=1524)))(CONNECT\_DATA=(SID=VISTEST)))"

java oracle.jrad.tools.xml.importer.XMLImporter $JAVA\_TOP/moiss/oracle/apps/fnd/moiss/webui/SearchPGRN.xml -jdk13 -mmddir $OA\_HTML/jrad -username apps -password apps -rootdir $JAVA\_TOP/moiss/oracle/apps/fnd -rootPackage /moiss/oracle/apps/fnd -validate -dbconnection "(DESCRIPTION=(ADDRESS\_LIST=(LOAD\_BALANCE=YES)(FAILOVER=YES)(ADDRESS=(PROTOCOL=tcp)(HOST=172.16.1.55)(PORT=1524)))(CONNECT\_DATA=(SID=VISTEST)))"

java oracle.jrad.tools.xml.importer.XMLImporter $JAVA\_TOP/moiss/oracle/apps/fnd/moiss/webui/UpdPGRN.xml -jdk13 -mmddir $OA\_HTML/jrad -username apps -password apps -rootdir $JAVA\_TOP/moiss/oracle/apps/fnd -rootPackage /moiss/oracle/apps/fnd -validate -dbconnection "(DESCRIPTION=(ADDRESS\_LIST=(LOAD\_BALANCE=YES)(FAILOVER=YES)(ADDRESS=(PROTOCOL=tcp)(HOST=172.16.1.55)(PORT=1524)))(CONNECT\_DATA=(SID=VISTEST)))"



Note : when you have many regions and pages you can have a bulk import. For bulk import just join the two scripts by && symbol as shown in below example. You can join as many import script as possible.

java oracle.jrad.tools.xml.importer.XMLImporter $JAVA\_TOP/moiss/oracle/apps/fnd/moiss/webui/DocLibCrtPG.xml -jdk13 -mmddir $OA\_HTML/jrad -username apps -password apps -rootdir $JAVA\_TOP/moiss/oracle/apps/fnd -rootPackage /moiss/oracle/apps/fnd -validate -dbconnection "(DESCRIPTION=(ADDRESS\_LIST=(LOAD\_BALANCE=YES)(FAILOVER=YES)(ADDRESS=(PROTOCOL=tcp)(HOST=172.16.1.55)(PORT=1524)))(CONNECT\_DATA=(SID=VISTEST)))"**&&** java oracle.jrad.tools.xml.importer.XMLImporter $JAVA\_TOP/moiss/oracle/apps/fnd/moiss/webui/DocLibPG.xml -jdk13 -mmddir $OA\_HTML/jrad -username apps -password apps -rootdir $JAVA\_TOP/moiss/oracle/apps/fnd -rootPackage /moiss/oracle/apps/fnd -validate -dbconnection "(DESCRIPTION=(ADDRESS\_LIST=(LOAD\_BALANCE=YES)(FAILOVER=YES)(ADDRESS=(PROTOCOL=tcp)(HOST=172.16.1.55)(PORT=1524)))(CONNECT\_DATA=(SID=VISTEST)))"

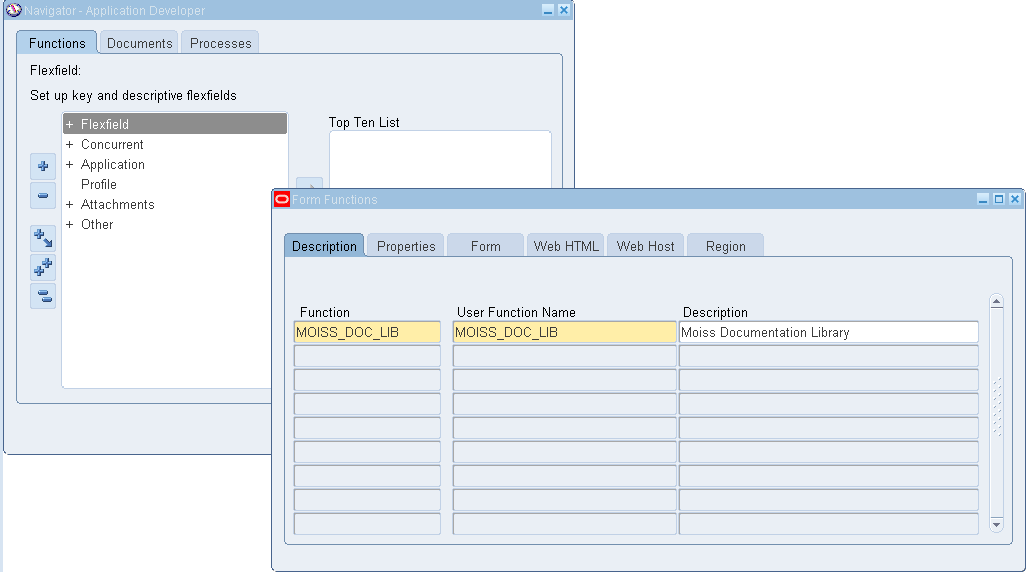
Note : You have make a note of the import file where it gets resided as shown in above screenshot . It is important for debugging and a good practice.

Step 3 : Creating Function

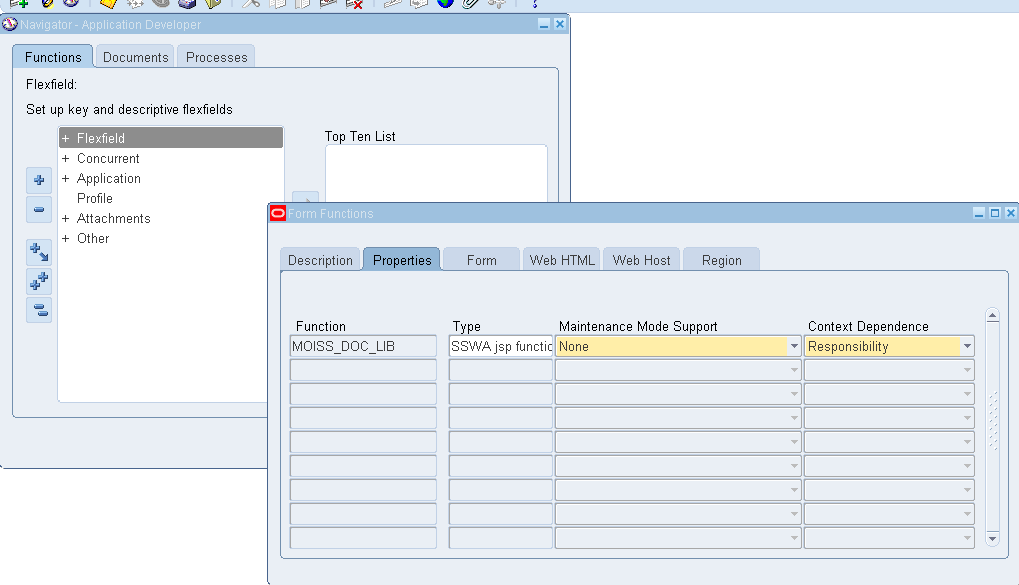
Till now our project has been moved to server. Now we have to define who should use it and where it can be found. For this you have to create function. We create function for setting the page url which is a type of register to oracle apps

Navigate to : Application Developer Responsibility 🡪 Application 🡪 Function

Create your function and make a note of the function name as shown below.



After creating a function go to properties tab and set type as SSWA Jsp Function this defines that it is a OAF page.



Now click on HTML call tab and give the script as below

The script should have your corresponding path for your page.

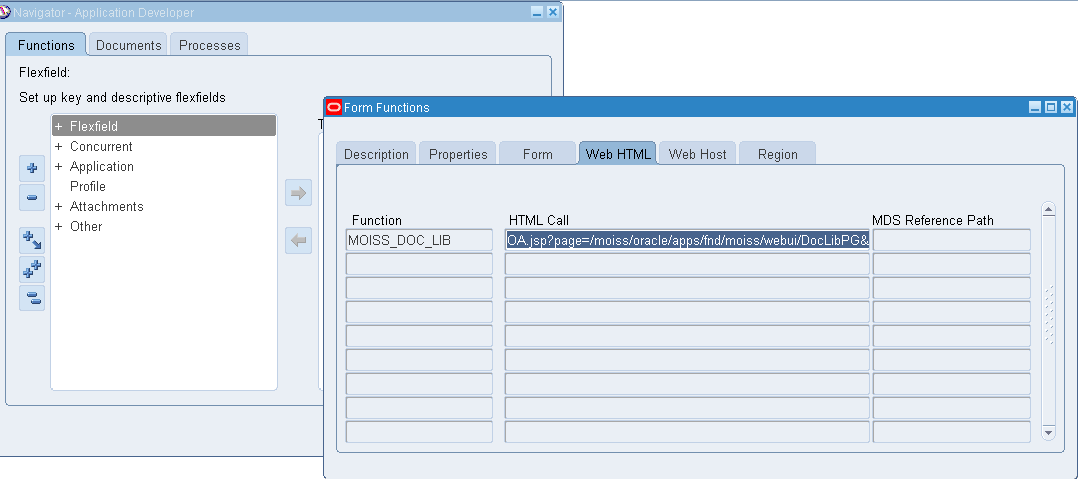
OA.jsp?page=/moiss/oracle/apps/fnd/moiss/webui/DocLibPG&akRegionApplicationId=0&OAHP=OA\_TEST\_INSTALL\_APPLICATION&OASF=OA\_TEST\_INSTALL\_LKP\_TYPE

Here After the “OA.jsp?page=” my page’s path name comes

OA.jsp?page=/moiss/oracle/apps/fnd/moiss/webui/DocLibPG&param=ADMIN&akRegionApplicationId=0&OAHP=OA\_TEST\_INSTALL\_APPLICATION&OASF=OA\_TEST\_INSTALL\_LKP\_TYPE

Can you observe a difference between two scripts above.

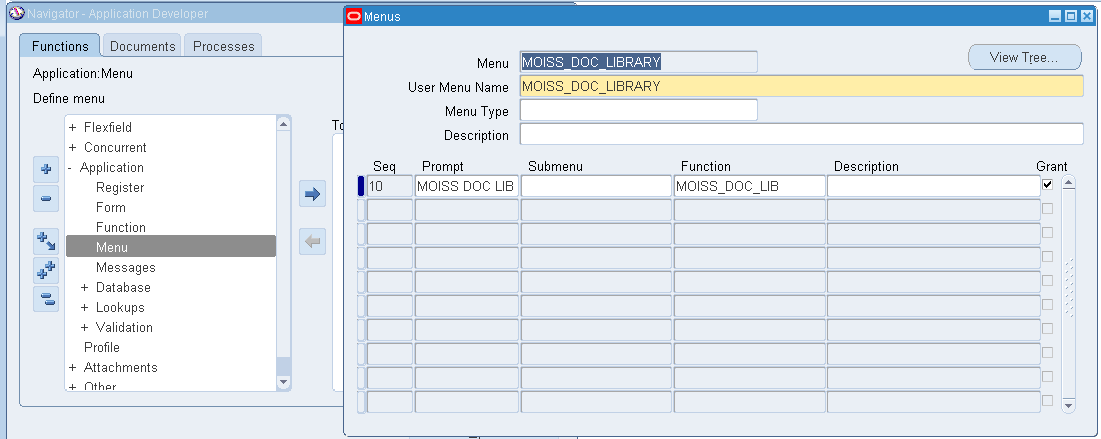
In the second script there is something extra that is **&param=ADMIN.** This is the parameter that im passing to the page. So you can also pass parameter on deployment.



Step 4: Creating Menu

Navigate to : Application Developer Responsibility 🡪 Application 🡪 Menu

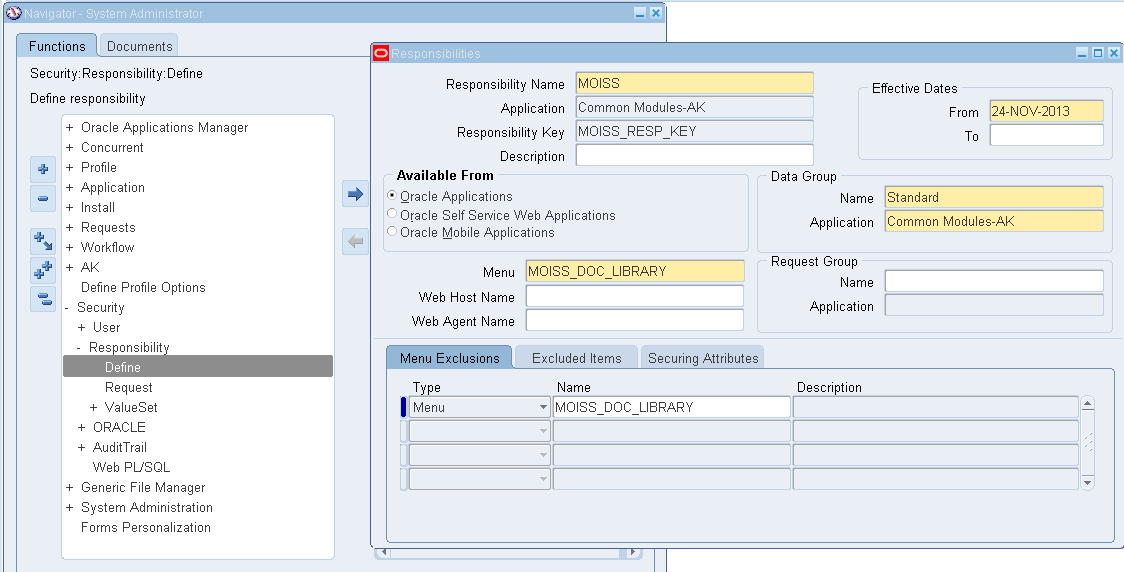
Now you have to create Menu attach the function you have created to this menu. This menu appears when you click the responsiblity



Step 5: Attaching Menu to Responsibility

Navigate to System Administrator Responsibility 🡪 Security 🡪 Responsibility 🡪Define

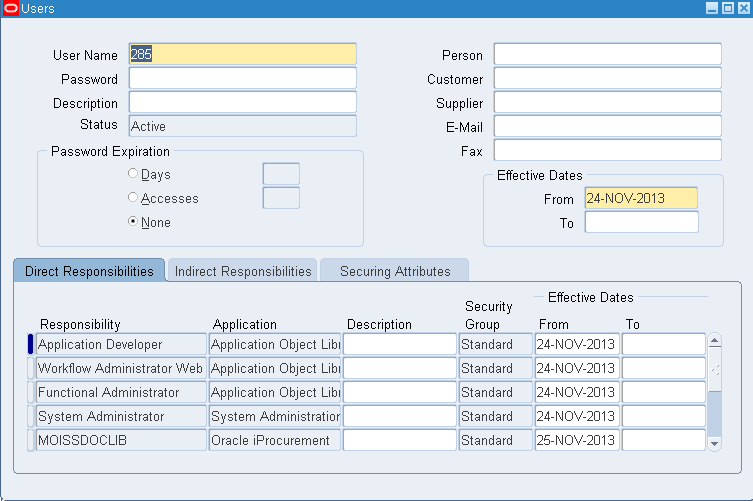
After you have created the Menu you to attach it to Responsibility or you can create your own responsibility, so that the users can view it from the responsibility



Step 6: Attaching responsibility to User

Navigate to System Administrator Responsibility 🡪 Security 🡪 User 🡪Define

Now we come the final of Deployment. After creating responsibility you have to attach the responsibility to the user



Now login to your Application and have a great day.

If you have any problems just redo the process where it went wrong.

If you were not able to see the responsibility or any problem after checking it twice you can bounce the server or you can also clear the cache.

**For Bouncing the server**

1) login to your instance with putty or any other software

2) cd $ADMIN\_SCRIPTS\_HOME

3) adapcctl.sh stop

4) adoacorectl.sh stop

5) adapcctl.sh start

6) adoacorectl.sh start

**For clearing the cache**

Navigate to: Functional Administrator Responsibility 🡪 Core Services 🡪 Caching Framework 🡪 Global configuration

You can use Both the scripts below for cross checking

BEGIN

jdr\_utils.ListDocuments('/moiss/oracle/apps/fnd/moiss/webui/DocLibPG',TRUE);

END;

BEGIN

jdr\_utils.PrintDocument('/moiss/oracle/apps/fnd/moiss/webui/DocLibPG',1000);

END;

For an extra information you can delete a page by below script. This is for your information it is not connected with this topic

Deleting the documents

*---------------------------------------*

BEGIN

jdr\_utils.deletedocument(p\_document => '/oracle/apps/per/irc/candidateSelfService/server/customizations/site/0/VisitorAM');

END;