Controller Extension in OAF

Set the values of following profiles to enable Personalization Page link in OAF Pages

Profile Name	Value
FND: Personalization Region Link Enabled	Yes
Personalize Self-Service Defn	Yes
Disable Self-Service Personal	No

Oracle does not recommend that customers extend controller objects associated with regions or webbeans in shipped E-Business Suite product pages.

Controller class (oracle.apps.fnd.framework.webui.OAControllerImpl) methods should effectively be considered private, since their implementation is subject to change. Controller extensions are therefore not considered to be durable between upgrades.

If it is absolutely essential to handle custom form submit events on a shipped product page, processFormRequest() is the only method that should be overriden in a controller class, although the risks outlined above still apply.

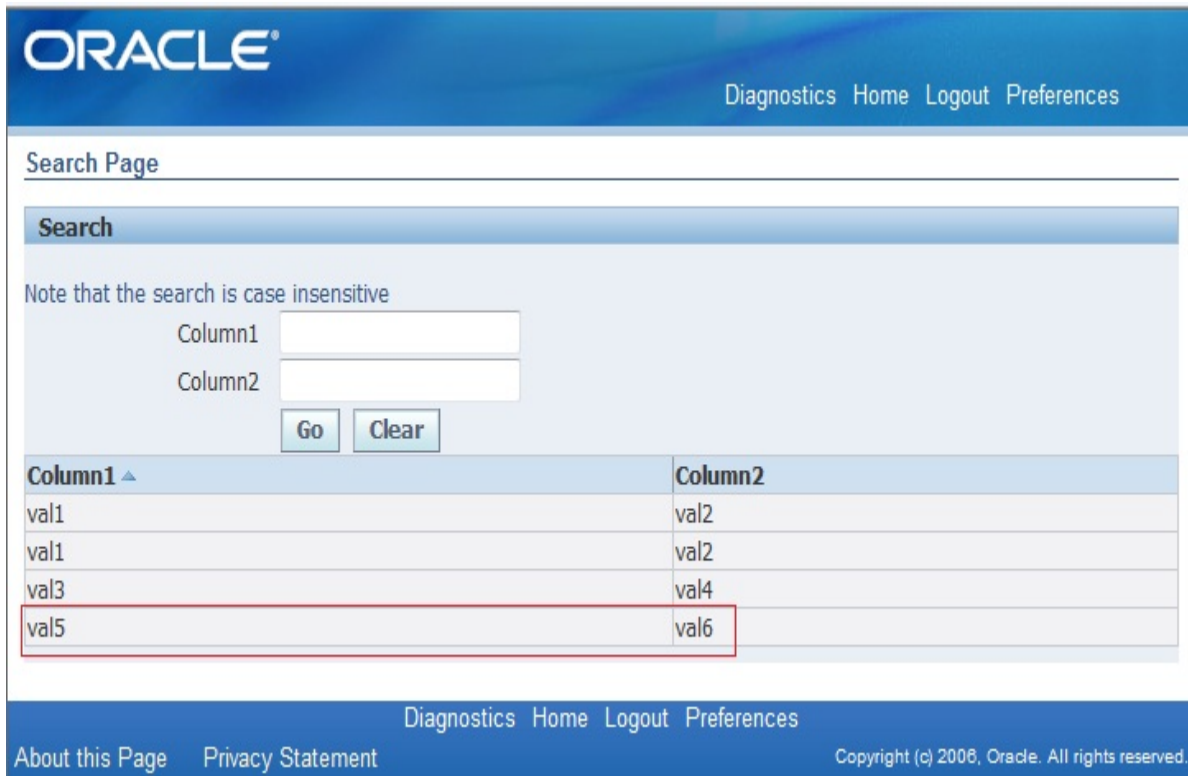**Let us try to Extend Controller in OAF Page –**

In this exercise we are going to extend CO of SearchPG.

Now we will extend this newly created CO under this exercise.

The purpose of this exercise is to modify the VO query of results table.

Now when we click on **Go** button all the records are displaying in the results table and our OBJECTIVE is to bind the VO query of results table in such a way that in result **Column1** value**val5** and **Column2** value **val6** should not come as result on click Go button

Now for knowing which controller to extend we click on "About This Page" Link and select Expand All. Here we can see the Name of the controller that we need to extend

Page Definition

Expand All | Collapse All

Focus Name	Controller	Application Module	View Object	View Attribute
⊟ pageLayout: Search Page Window	SearchCO	SearchAM		
rawText: <noscript>JavaScript enabled browser re...				
rawText: <noscript>JavaScript enabled browser re...				
rawText: <noscript>JavaScript enabled browser re...				
⊟ stackLayout				
⊟ stackLayout: (QueryRN)				
⊟ stackLayout				
⊟ header: Search				
⊟ stackLayout				
⊟ tableLayout				
⊟ rowLayout				
⊟ cellFormat				
⊟ tableLayout				
⊟ rowLayout				
styledText: Note that the search is case insensitive				
⊟ messageComponentLayout: (ResultTable_mclrbs_queryRN)				
messageTextInput: Column1				
messageTextInput: Column2				
⊟ messageLayout				
submitButton: Go				
submitButton: Clear				
⊟ table: (ResultTable)			SearchVO1	
messageStyledText: Column1			SearchVO1	Column1
messageStyledText: Column2			SearchVO1	Column2
sortableHeader: (OASH__44)				
image: Oracle Logo				
link: About this Page				

1. Create a New Workspace and Project

File > New > General > Workspace Configured for Oracle Applications

File Name – MahiCOExtensionDemo

Automatically a new OA Project will also be created

Project Name -- COExtensionDemo

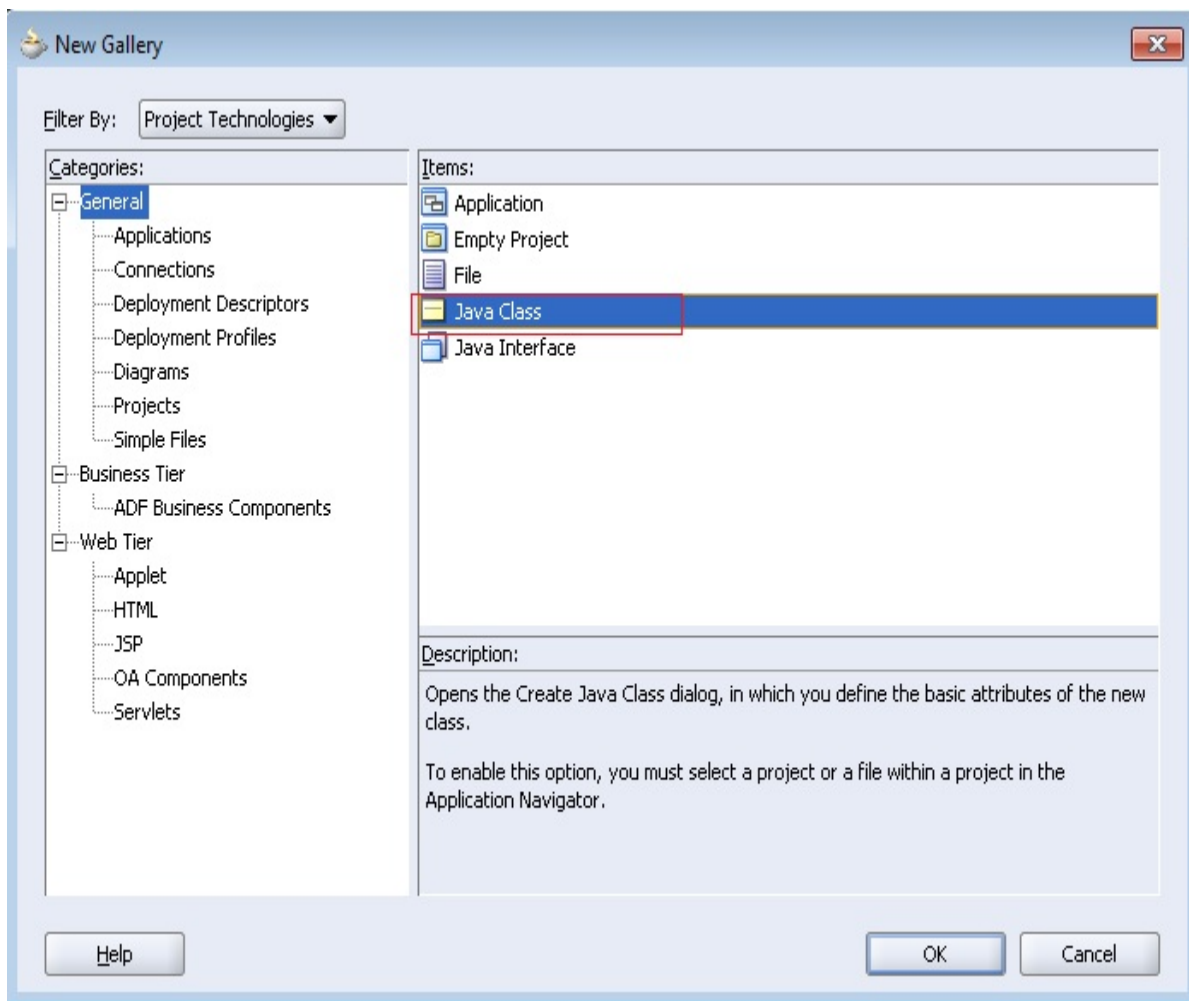Default Package -- Mahi.oracle.apps.fnd.coextensiondemo.webui

2. Create a New Java Class

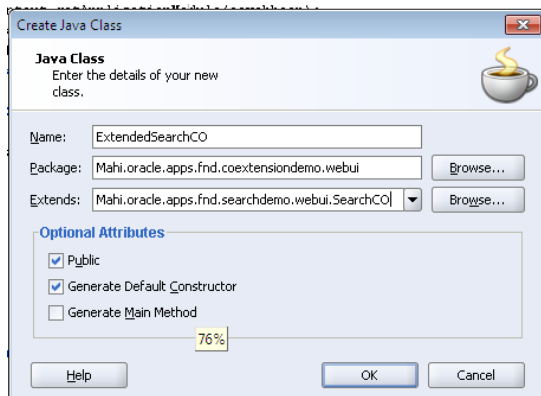Right Click on COExtensionDemo > New > General > Java Class

Name -- ExtendedSearchCO

Package -- Mahi.oracle.apps.fnd.coextensiondemo.webui

Extends -- Mahi.oracle.apps.fnd.searchdemo.webui.SearchCO

Note -- Give the Name of your Extended Class give its package path and in the extends property select base class

3. Write below logic in ExtendedSearchCO Java Class

package Mahi.oracle.apps.fnd.coextensiondemo.webui;

import Mahi.oracle.apps.fnd.searchdemo.webui.SearchCO;

import oracle.apps.fnd.framework.webui.OAPageContext;
import oracle.apps.fnd.framework.webui.beans.OAWebBean;
import oracle.apps.fnd.framework.OAApplicationModule;
import oracle.apps.fnd.framework.webui.beans.layout.OAQueryBean;
import Mahi.oracle.apps.fnd.searchdemo.server.SearchVOImpl;

public class ExtendedSearchCO extends SearchCO
{
 public ExtendedSearchCO ()
 {
 }

 public void processFormRequest(OAPageContext pageContext, OAWebBean webBean)
 {
 super.processFormRequest(pageContext, webBean);
 OAApplicationModule am = pageContext.getApplicationModule(webBean);
 OAQueryBean queryBean = (OAQueryBean)webBean.findChildRecursive("QueryRN");
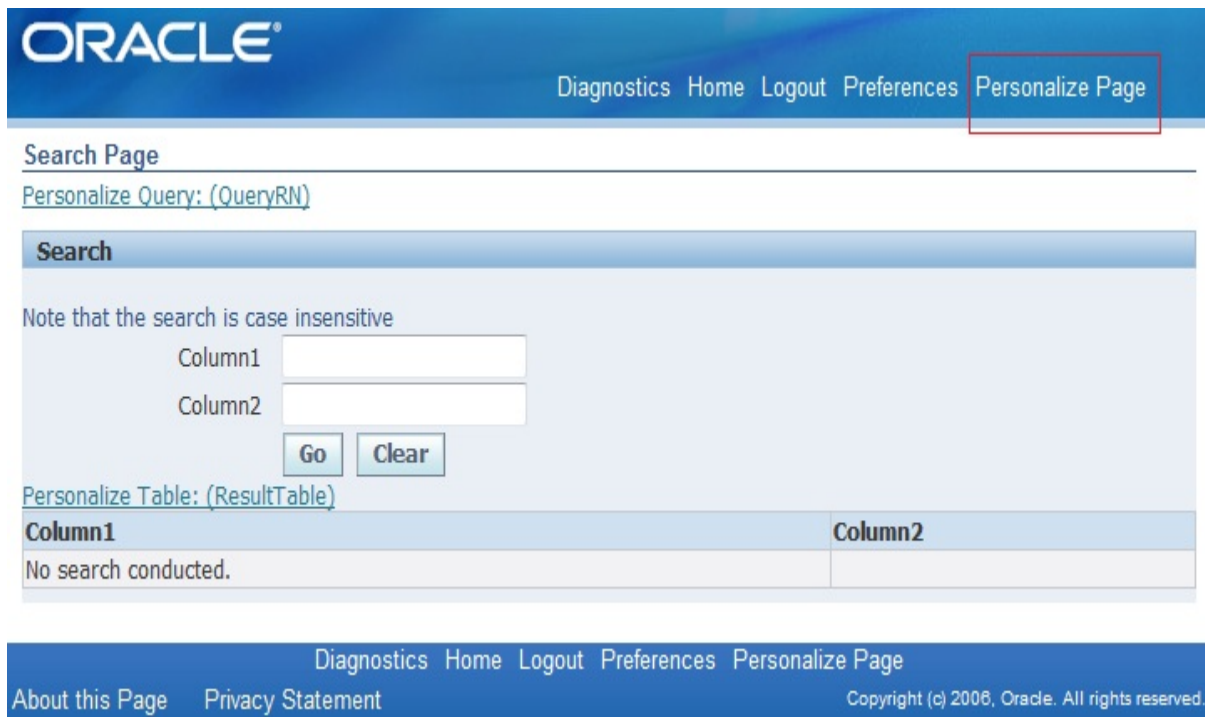
 //Capturing Go Button ID
 String go = queryBean.getGoButtonName();

 //If its Not NULL which mean user has pressed "Go" Button
 if(pageContext.getParameter(go)!=null)
 {
 // Setting whereClause at Runtime to restrict the query
 SearchVOImpl vo = (SearchVOImpl)am.findViewObject("SearchVO1");
 vo.setWhereClause(null);
 vo.setWhereClause("Column1 <>:1 AND Column2 <>:2");
 vo.setWhereClauseParam(0,"val5");

```
   vo.setWhereClauseParam(1,"val6");
 }
}
}
```

4. Attach new controller to SearchPG through personalization

Click on Personalize Page link on top right hand side of your page

Click on Complete View -> Expand All -> Click on personalize icon next to Page Layout

Personalization Structure

○ Simple View ● Complete View

Expand All | Collapse All

Focus	Name	Shown	User Personalizable	Personalize	Reorder	Create Item	Update Item	Delete Item	Seeded User Views
	⊟ Page Layout: Search Page Window	Yes		✏		📄			
⊕	⊟ Query: (QueryRN)	Yes		✏		📄			
⊕	⊟ Table: (ResultTable)	Yes	Yes	✏	🔁	📄			∞
	Message Styled Text: Column1	Yes	Yes	✏					
	Message Styled Text: Column2	Yes	Yes	✏					
⊕	⊟ Corporate Branding								
	Image: (corporateBrandingImage)	Yes		✏					

Now at site level give the path of extended controller as we are extending the controller at **SITE LEVEL**

Mahi.oracle.apps.fnd.coextensiondemo.webui.ExtendedSearchCO

 By defaylt it will come as Inherit

Click Apply -> Return to Application

ORACLE®

Diagnostics Home Logout Preferences Personalize Page

Search Page
Personalize Query: (QueryRN)

Search

Note that the search is case insensitve

Column1 [_____]

Column2 [_____]

[Go] [Clear]

Personalize Table: (ResultTable)

Column1	Column2
No search conducted.	

Diagnostics Home Logout Preferences Personalize Page

About this Page   Privacy Statement   Copyright (c) 2006, Oracle. All rights reserved.

Click Go

Note – Record with **Column1** value **val5** and **Column2** value **val6** is not coming in result