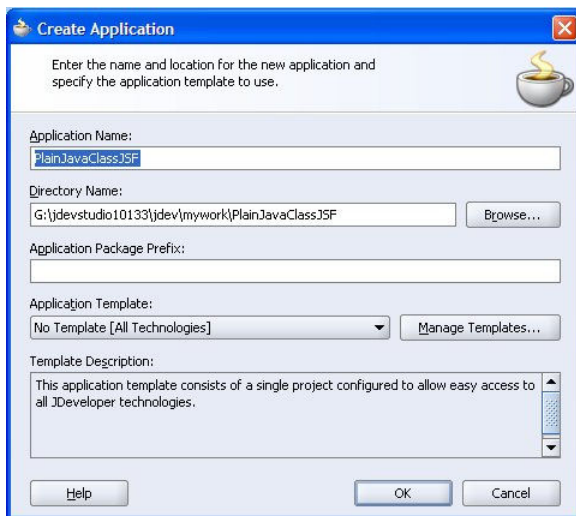
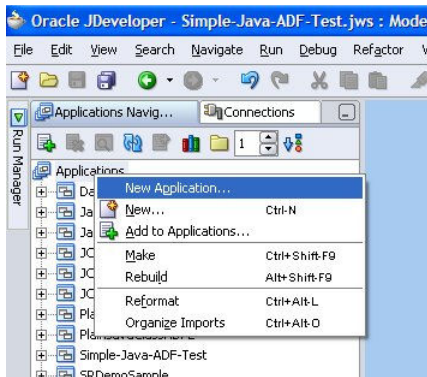


## TUTORIAL: Using Plain Java Class as business component for JSF page

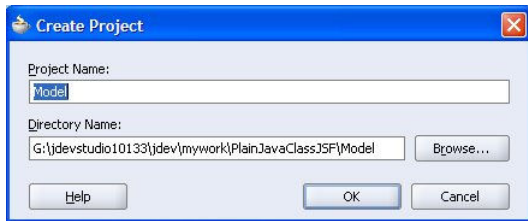
By: Dr.Ahmad Taufik Jamil, Hospital UKM

This tutorial will show how to use plain java class as business component for a JSF page. ADF data control/binding will be used.

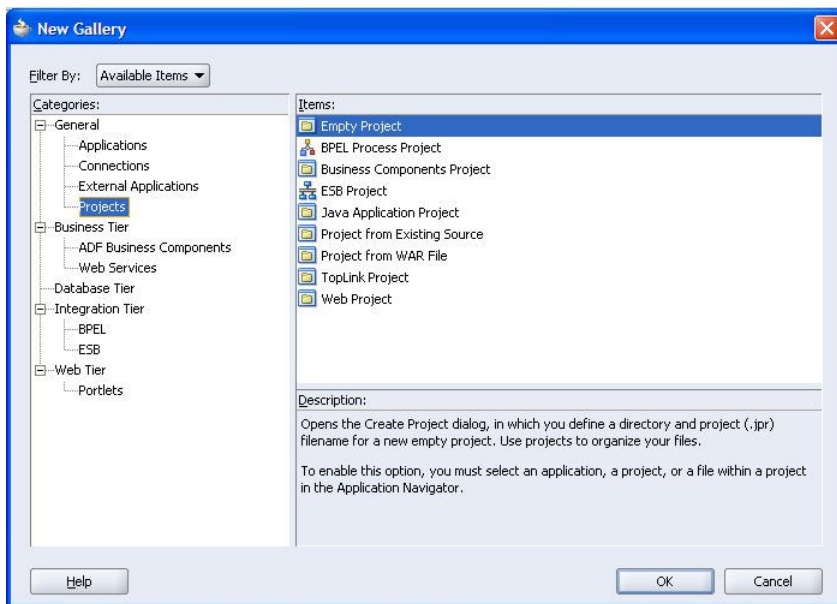
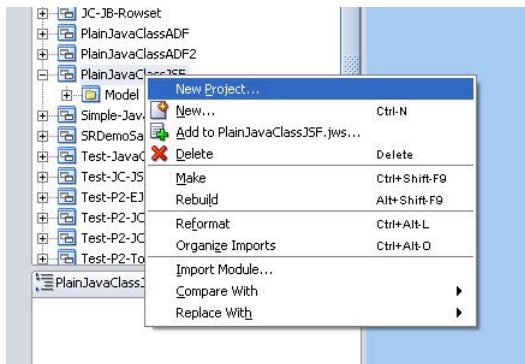
1. Creating workspace and project folder
  - a. Launch JDeveloper 10g
  - b. Create New Workspace, enter **PlainJavaClassJSF** for *Application Name*; and **No Template [All Technologies]** for *Application Template*; click **OK**.

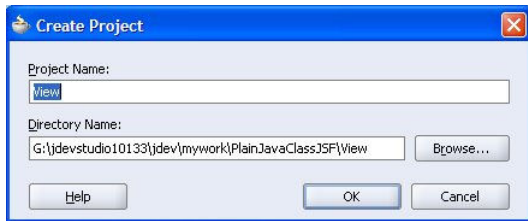


- c. Create a project folder. Enter **Model** for *Project Name*; then click **OK**.

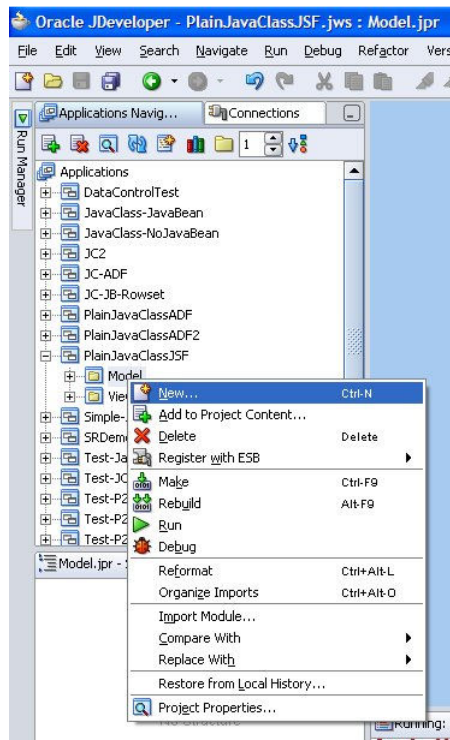


- d. Right click **PlainJavaClassJSF** workspace, click *New Project...*, in the *New Gallery* dialog box, click *Project (General)*, click *Empty Project* in column *Items:*, click *OK*, then, enter **View** for *Project Name:* and click *OK*.

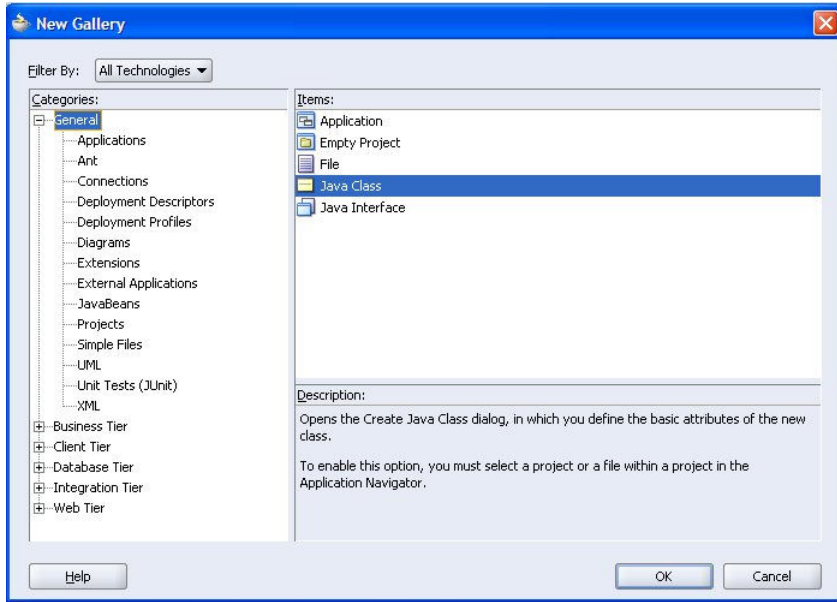




2. Creating java class
  - a. Right click **Model** project folder, click *New...*

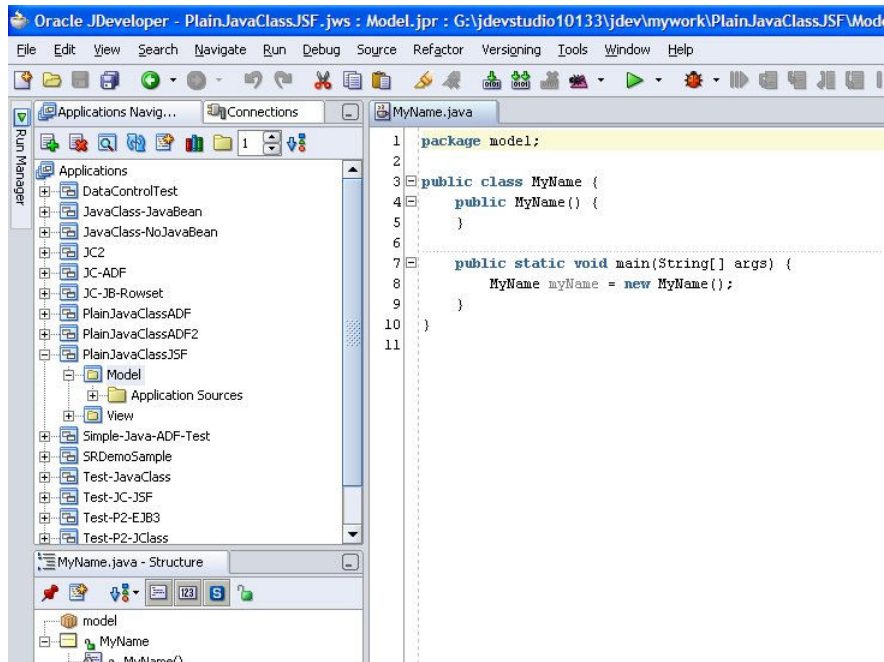


- b. In the *New Gallery* dialog box, click *General*, click *Java Class* inside *Items:* column, click *OK*.

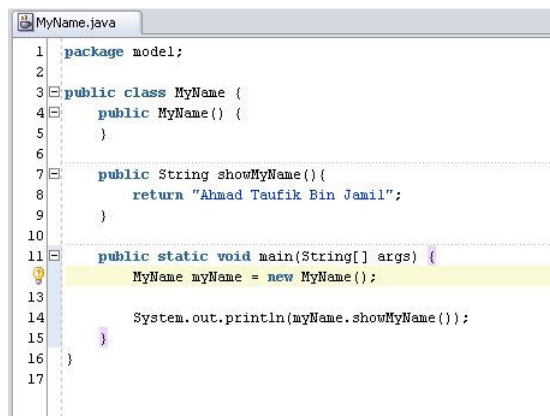


c. Enter **MyName** for column *Name*; check *Generate Main Method*, click *OK*.

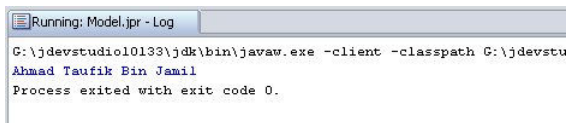
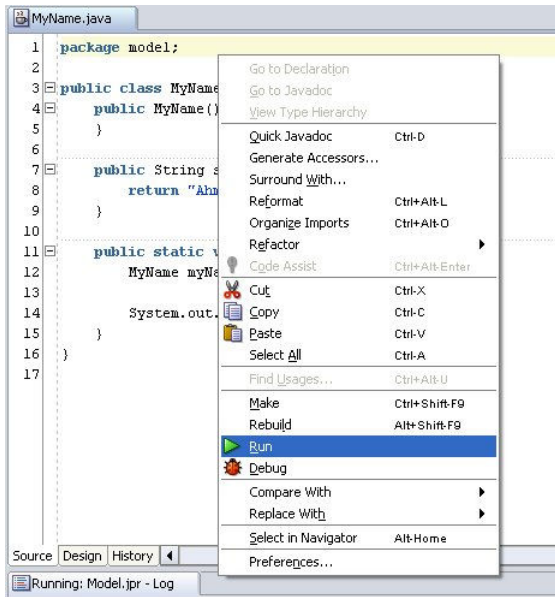




d. Enter source code below.

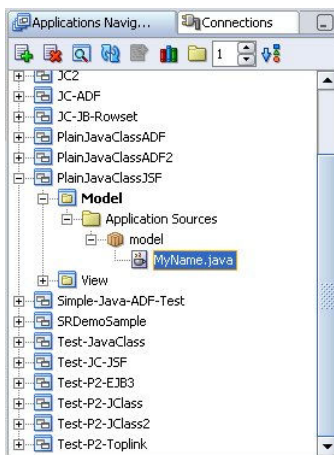


e. Run the **MyName.java** and see the output.



### 3. Creating data control

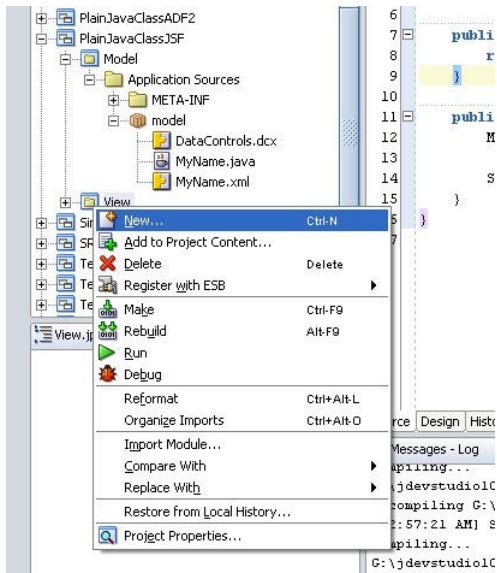
- a. Expand **Model** project folder until you see **MyName.java**



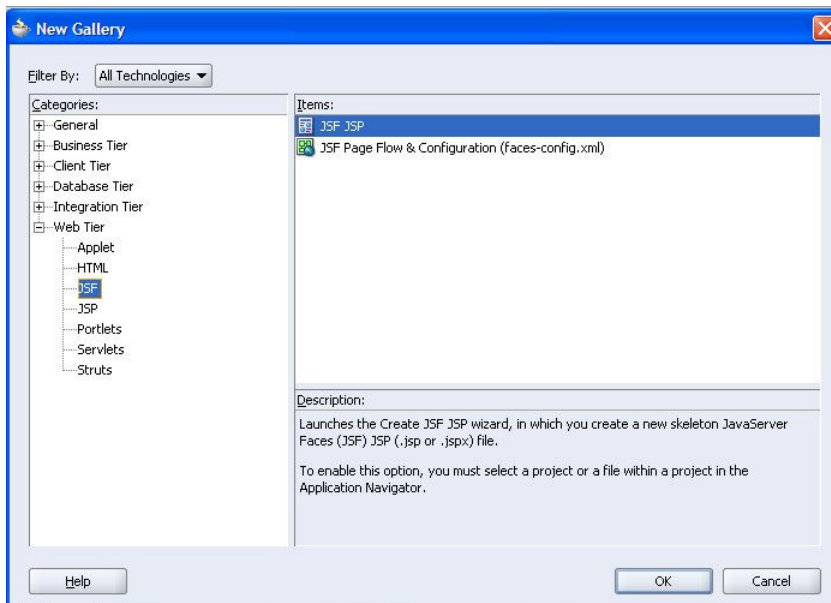
- b. Right click **MyName.java** and click *Create Data Control*.



a. Right click **View** project folder, click *New...*



b. In the *New Gallery* dialog box, expand *Web Tier*, click *JSF*, inside *Items:* column, click *JSF JSP*, click *OK*.

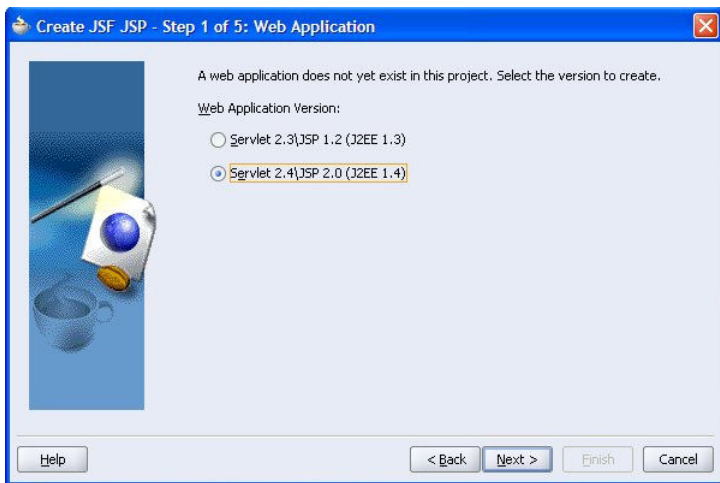


c. In the *Create JSF JSP* wizard, click *Next*.

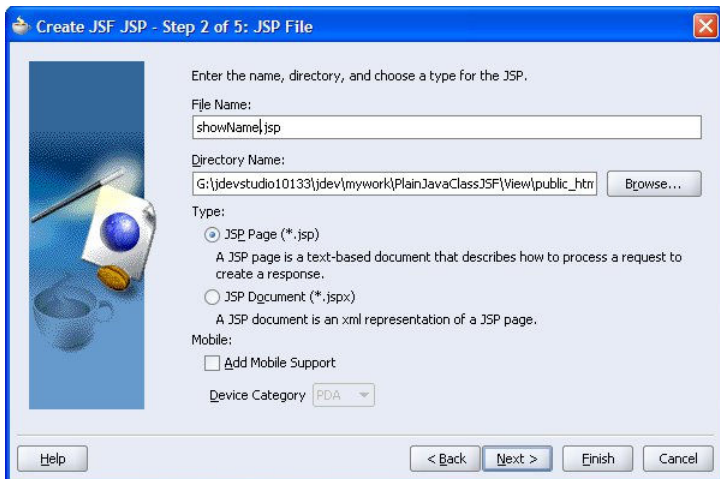




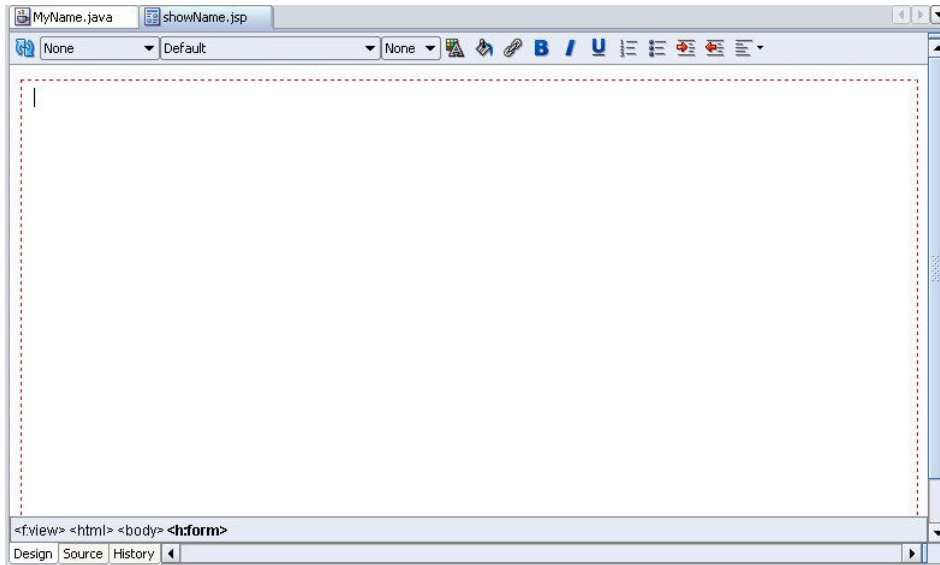
d. Choose *Servlet 2.4*, click *Next*.



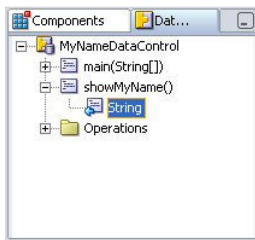
e. Enter **showName.jsp** for *File Name:* and click *Finish*.



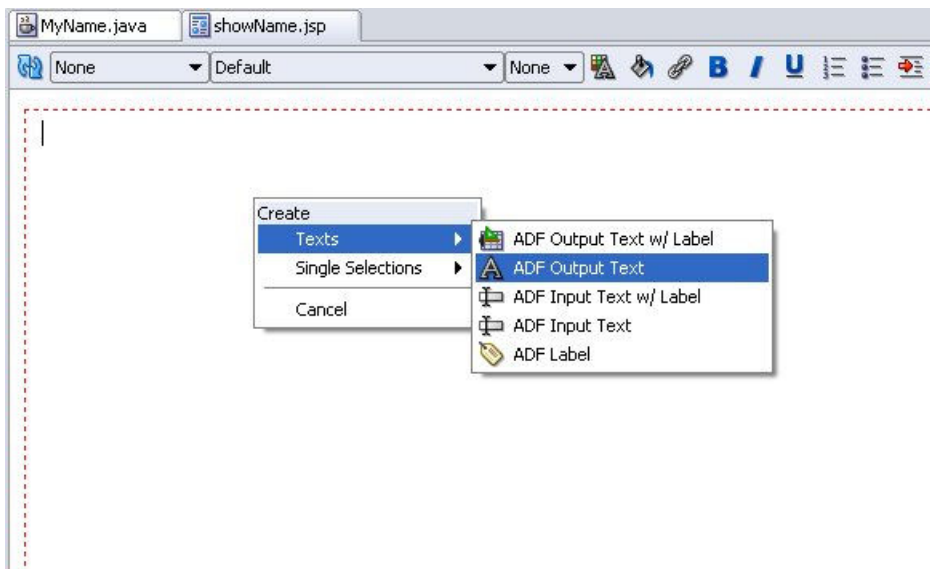
f. See the outcome.



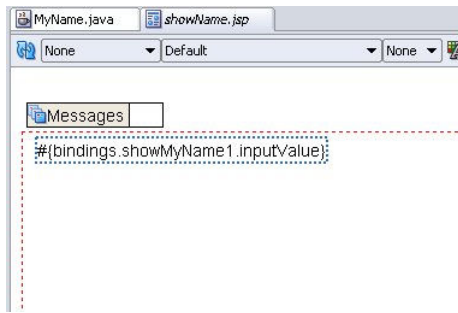
g. Inside *Data Control Palette*, expand **showMyName()**, you will see **String**.



h. Choose **String**, drag & drop onto **showName.jsp** design view. Choose *Text*, click *ADF Output Text*.

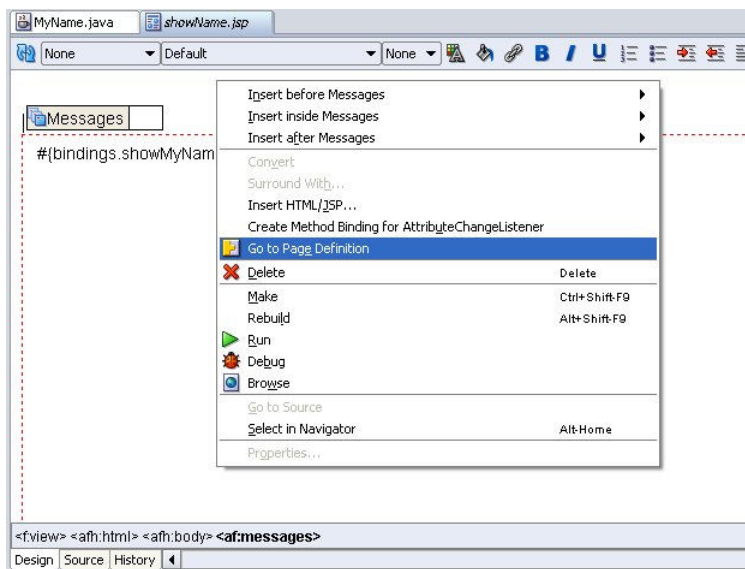


i. See the outcome.



## 5. Editing page definition file

- a. Right click on **showName.jsp**, click *Go To Page Definition*.



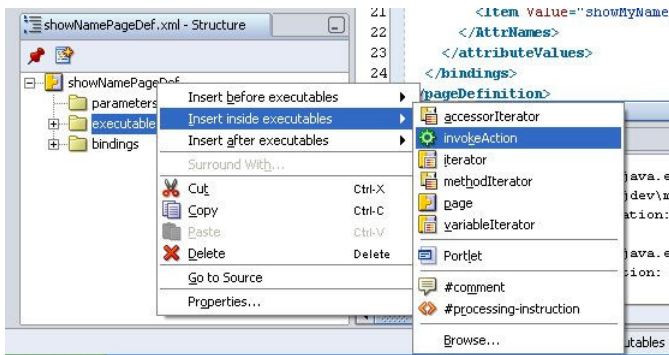
- b. See the outcome.

```

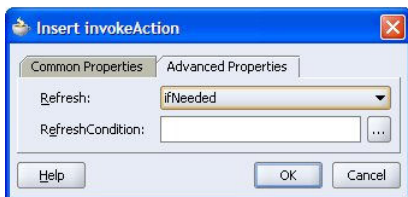
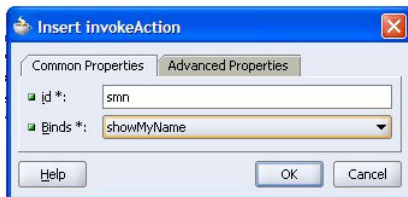
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <pageDefinition xmlns="http://xmlns.oracle.com/adfm/timodel"
3 version="10.1.3.41.57" id="showNamePageDef"
4 Package="view.pageDefs">
5
6 <parameters/>
7 <executables>
8 <variableIterator id="variables">
9 <variable Type="java.lang.String" Name="showMyName_return"
10 IsQueryable="false" IsUpdateable="0"
11 DefaultValue="{bindings.showMyName.result}"/>
12 </variableIterator>
13 </executables>
14 <bindings>
15 <methodAction id="showMyName" InstanceName="MyNameDataControl.dataProvider"
16 DataControl="MyNameDataControl" MethodName="showMyName"
17 RequiresUpdateModel="true" Action="999"
18 IsViewObjectMethod="false"
19 ReturnName="MyNameDataControl.methodResults.MyNameDataControl_dataProvider_showMyName" />
20 <attributeValues id="showMyName1" IterBinding="variables">
21 <AttrNames>
22 <Item Value="showMyName_return"/>
23 </AttrNames>
24 </attributeValues>
25 </bindings>
26 </pageDefinition>

```

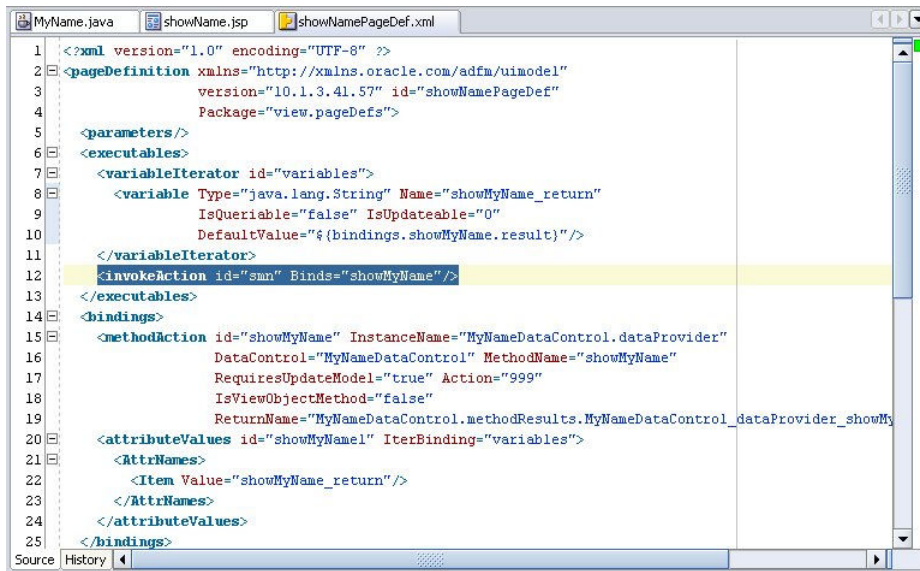
c. On the **showNamePageDef.xml** structure, right click *executables*, *Insert Inside executables*, click *invokeAction*.



d. Inside *Insert invokeAction* dialog box, enter **smn** for *id\**: and **showMyName** for *binds\**:, click OK.



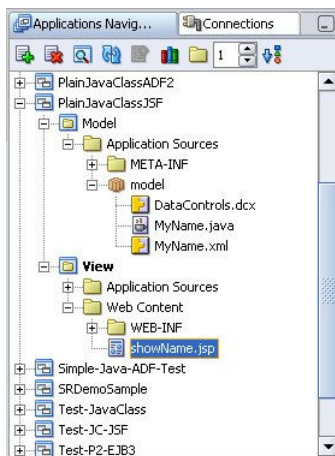
e. See the outcome.



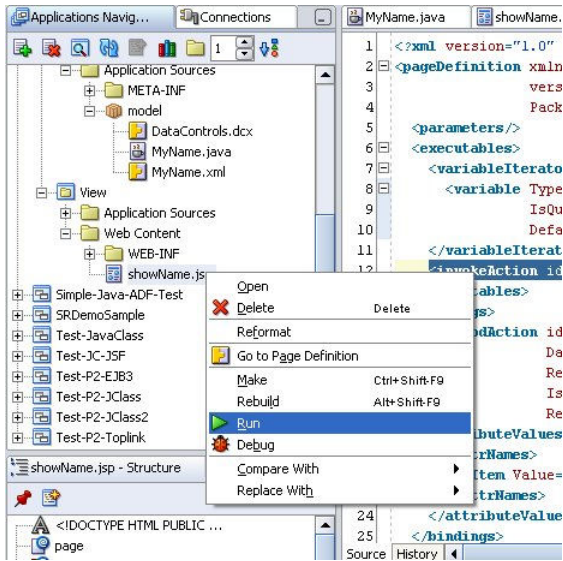
```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <pageDefinition xmlns="http://xmlns.oracle.com/adfm/uimodel"
3 version="10.1.3.41.57" id="showNamePageDef"
4 Package="view.pageDefs">
5 <parameters/>
6 <executables>
7 <variableIterator id="variables">
8 <variable Type="java.lang.String" Name="showMyName_return"
9 IsQueryable="false" IsUpdateable="0"
10 DefaultValue="{bindings.showMyName.result}"/>
11 </variableIterator>
12 <invokeAction id="smn" Binds="showMyName" />
13 </executables>
14 <bindings>
15 <methodAction id="showMyName" InstanceName="MyNameDataControl.dataProvider"
16 DataControl="MyNameDataControl" MethodName="showMyName"
17 RequiresUpdateModel="true" Action="999"
18 IsViewObjectMethod="false"
19 ReturnName="MyNameDataControl.methodResults.MyNameDataControl_dataProvider_showMy
20 <attributeValues id="showMyName1" IterBinding="variables">
21 <AttrNames>
22 <Item Value="showMyName_return" />
23 </AttrNames>
24 </attributeValues>
25 </bindings>
```

6. Test and run JSF file.

a. Find file **showName.jsp** inside **View** project folder.



b. Right click **showName.jsp**, and click *Run*.



c. See the outcome.

