Oracle Application Express Workshop I

Acticity Guide – Volume II

D79653GC10
Edition 1.0
March 2013
D81481

Authors

Anupama Mandya, Marcie Young

Technical Contributors and Reviewers

Anthony Rayner, Bryan Roberts, Chaitanya Koratamaddi, David Peake, Hilary Farrell, Maarc Sewtz, Patrick Wolf, Sathish Kumar, Shakeeb Rahman, Wayne Abbott, Klaus Husermann, Salome Clement, Nancy Greenberg, Maria Billings, Diganta Choudhury, Joel Kallman, Yi Lu, Lakshmi Narapareddi, Swarnapriya Shridhar, Jason Straub

This book was published using: Oracle Tutor

Table of Contents

Practices for Lesson 12: Adding Shared Components That Aid Navigation

Chapter 12

Practices for Lesson 12: Overview

Practices Overview

There are five practices for this lesson. You create a list and list region, create and edit tabs, manage help text, add a navigation bar entry, and add a breadcrumb.

Practice 12-1: Creating Lists and List Regions

Overview

Create a list of links that aid navigation. Using these links, a user will be able to navigate to the Customers, Products, and Orders pages.

Assumptions

You have performed the previous practices or imported the `LAB112_SOLN.sql` file.

Tasks

a. In the Order Management application, create a list called Home Page List with the following list entries:
   - View Customers (linked to the Customers page)
   - View Products (linked to the Products page)
   - View Orders (linked to the Orders page)

b. Create a list region called Tasks in the Home region.

c. Create a Dynamic List showing all the Products on the Products page. Create the List Region on the Products page by using the same Create List Wizard.

Practice 12-2: Creating and Editing Standard Tabs

Overview

In this practice, you create tabs for the application and assign pages to the tabs. You also create a Help page and a Help tab to add to the existing tab set.

Assumptions

You have performed the previous practices.

Tasks

a. Create a tab called Orders for the Order Management application. It should be the current tab for the List of Orders page. Sequence the Orders tab before the Products tab.
b. Assign the Master Detail page to the Orders tab. Run the application and confirm that the tab is created successfully.
c. Create a page called Help, and then add a new tab called Help, to the existing tab set. The new Help tab will link to the new Help page.

Practice 12-3: Adding Navigation Bar Entries

Overview

In this practice, you add a navigation bar entry that, when clicked, will show the page-level help. You also modify the name of an existing navigation bar entry and add another navigation bar entry that is displayed on all pages except the page that it navigates to.

Assumptions

You have performed the previous practices.

Tasks

a. Create a navigation bar entry called Help that navigates to the Help page. Run the page and click the Help navigation bar entry.
b. Change the name of the navigation bar entry for Logout to Sign Off.
c. Add a navigation bar entry called Home. Display the navigation bar entry on all pages except the Home page.

Practice 12-4: Adding Breadcrumbs and Tabs to an Existing Page

Overview

In this practice, you create a breadcrumb and add a tab set to an existing page.

Assumptions

You have performed the previous practices.

Tasks

a. Edit the existing breadcrumb for the Order Management application and create a breadcrumb entry for the Customers page. Run the customers page. Is the breadcrumb displayed on the page?
b. Create a breadcrumb region on the Customers page and run the page.
c. On the Customer Details page, create a breadcrumb called Customer Details whose parent entry is Customers (page 3).

Solution 12-1: Creating Lists and List Regions

Overview

Create a list of links that aid navigation. Using these links, a user will be able to navigate to the Customers, Products, and Orders pages.

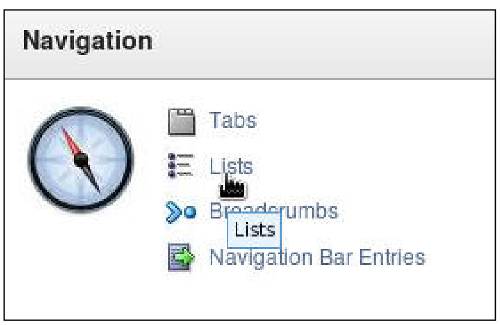
Steps

a. In the Order Management application, create a list called Home Page List with the following list entries:

- View Customers (linked to the Customers page)
- View Products (linked to the Products page)
- View Orders (linked to the Orders page)

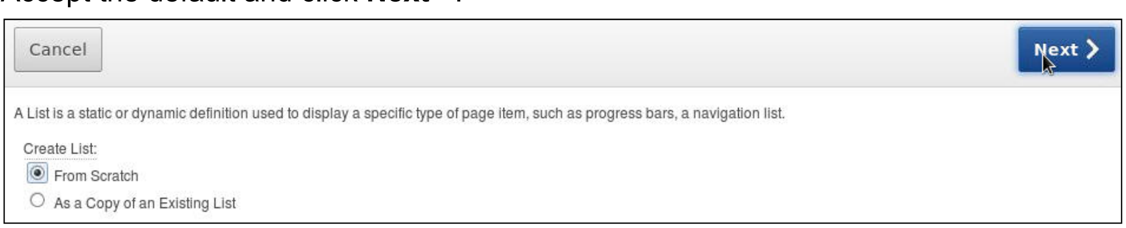
1) Navigate to your **Order Management** application home page and select **Shared Components**.

2) In the Navigation section, click **Lists**.

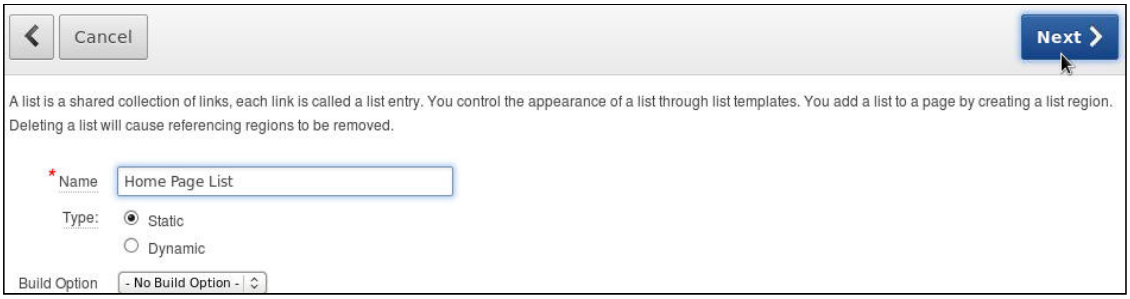


3) Click **Create >**.

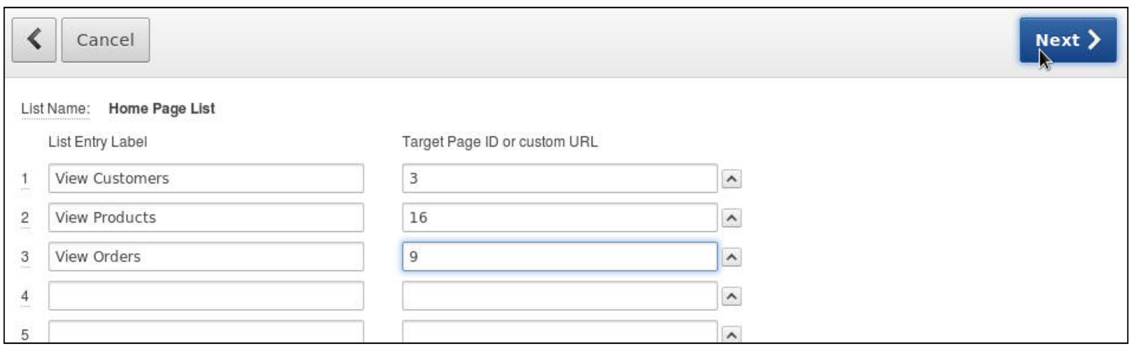
4) Accept the default and click **Next >**.



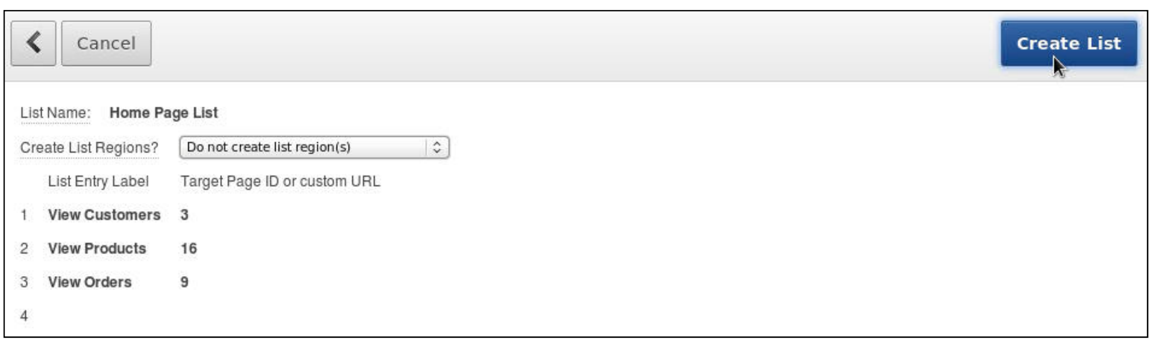
5) For Name, enter `Home Page List` and click **Next >**.



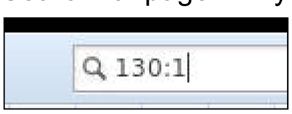
6) Enter **View Customers** in the first List Entry Label field. Then click the corresponding up button and select the Customers page for Target ID. Similarly, enter **View Products** and **View Orders** in the second and third List Entry Label fields. Then select the Products and List of Orders pages respectively by clicking the up arrow. Click **Next >**.



7) Accept the default and click **Create List**.

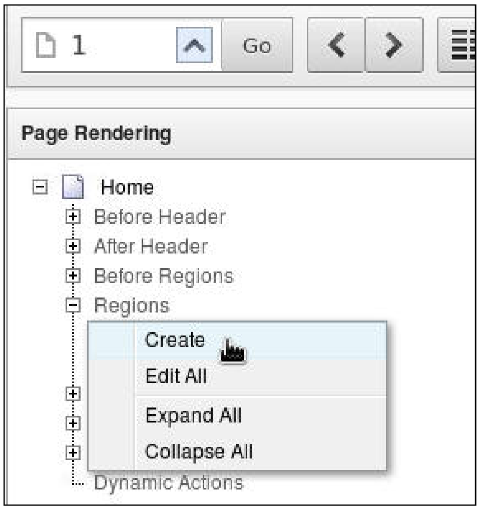


8) Search for page 1 in your Order Management application.

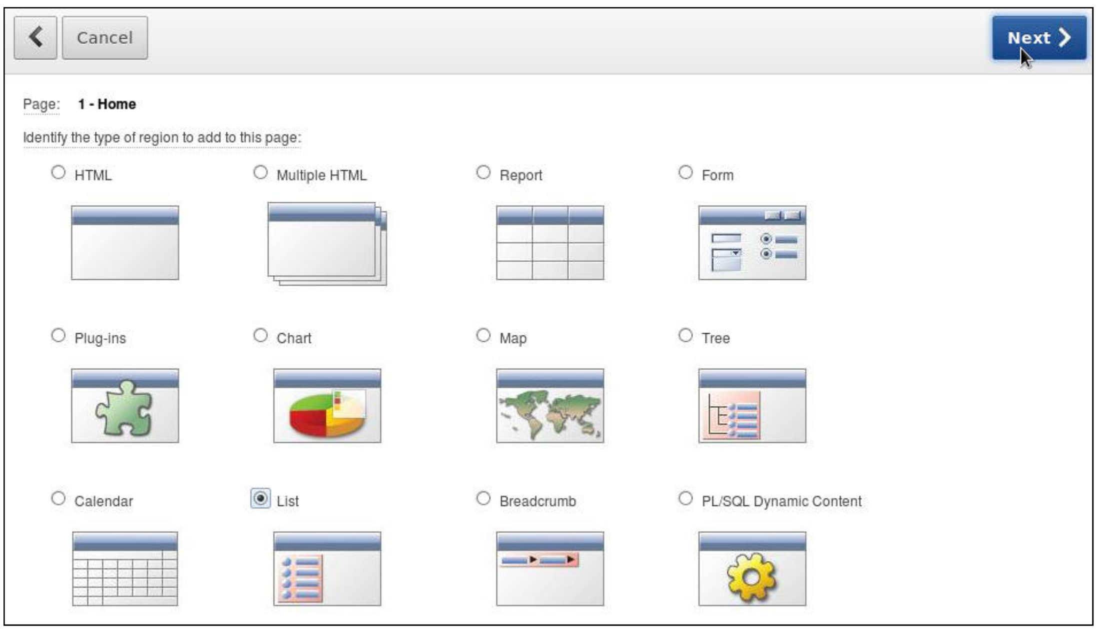


b. Create a list region called Tasks on the Home page.

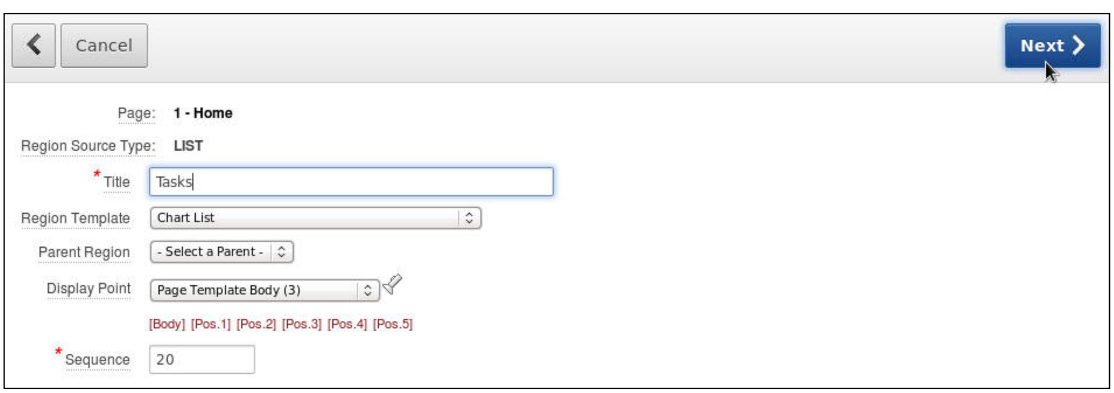
1) From the Home page definition, right-click the **Regions** and select **Create**.



2) Select the **List** option, and click **Next >**.

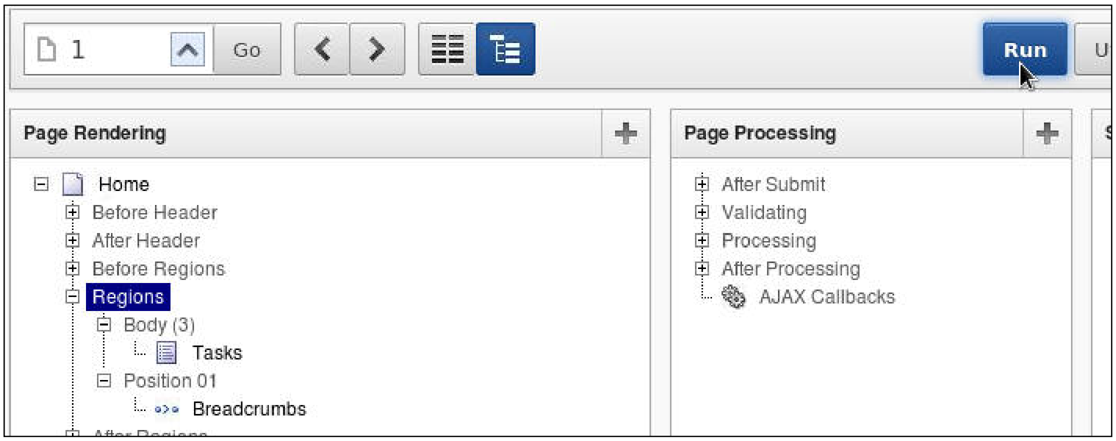


3) For Title, enter `Tasks` and click **Next >**.

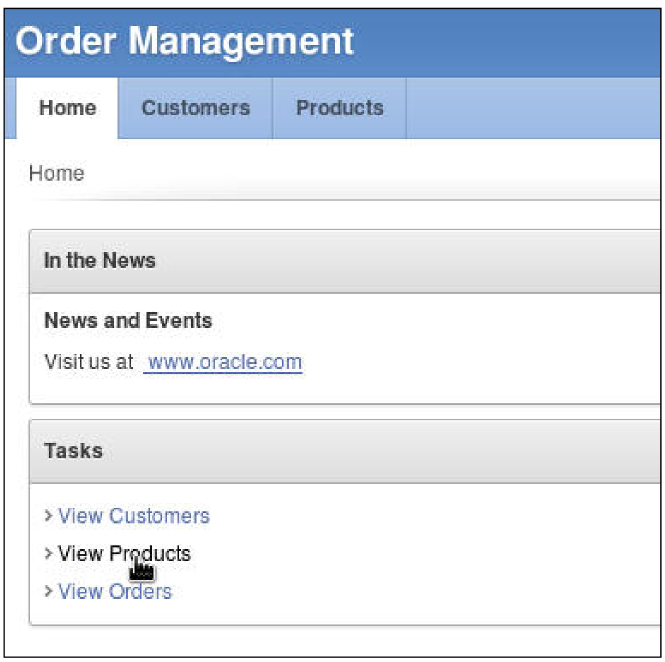


4) From the List drop-down list, select **Home Page List** and click **Create List Region**.

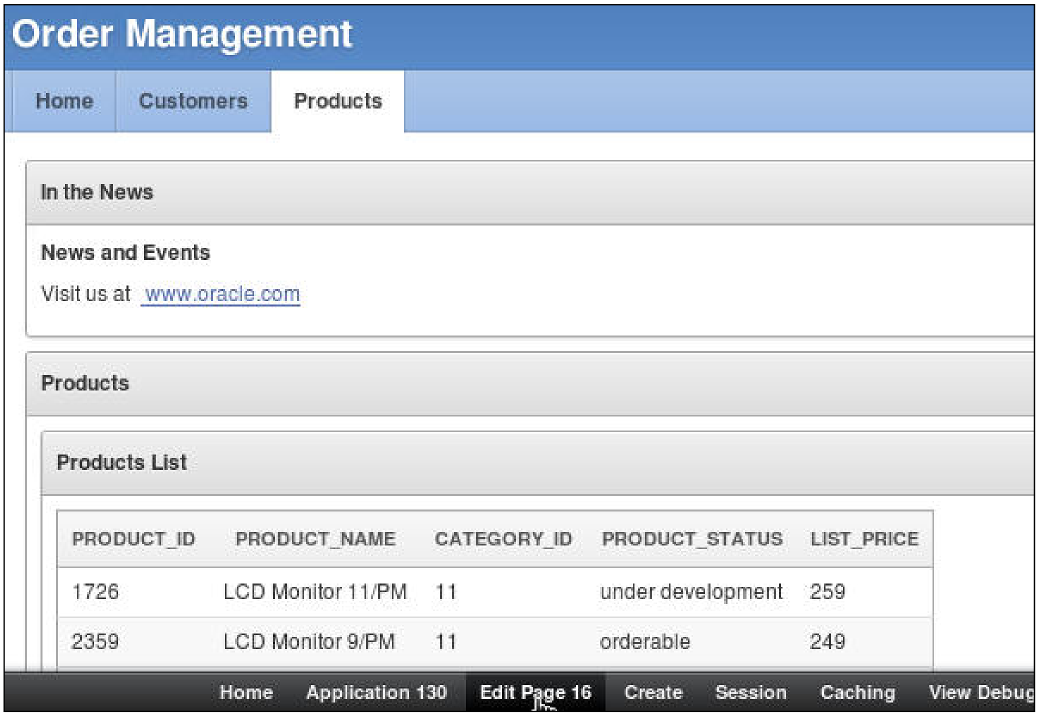
5) To view the list, run the page. Click **Run**.



6) The list is displayed. Click the **View Products** link.

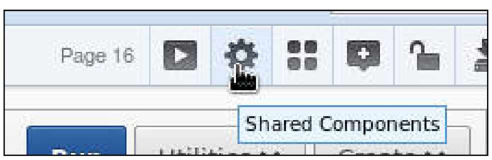


7) The Products page is displayed. Click the **Edit Page** link on the Developer toolbar.



c. Create a Dynamic List showing all the Products on the Products page. Create the List Region on the Products page using the same Create List Wizard.

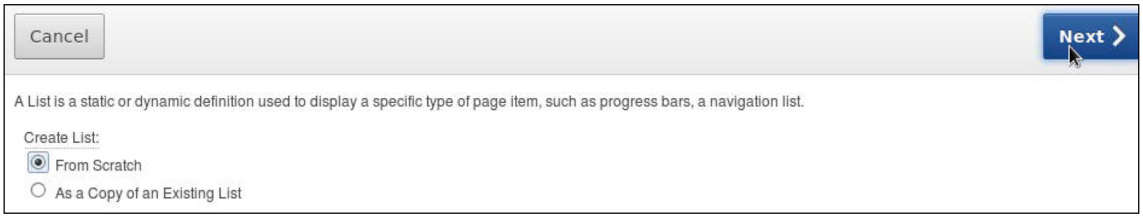
1) On the **Products** page, click the **Shared Components** icon.



2) Click **Lists** under Navigation.

3) Click **Create >**.

4) Accept the default and click **Next >**.

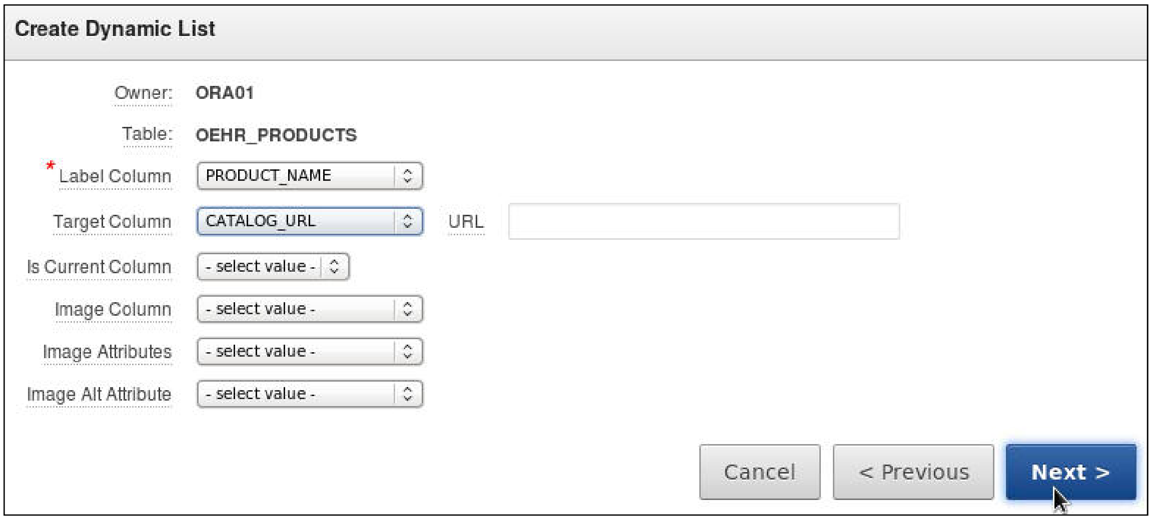


5) Enter **List of Products** for Name, select **Dynamic** for Type, and click **Next >**.

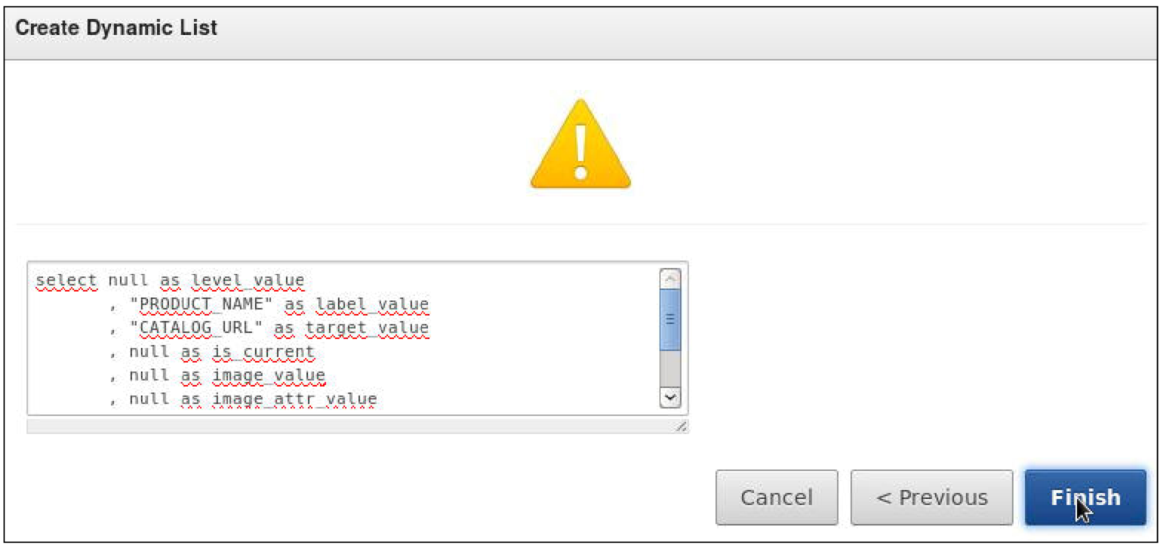
6) Click the **Create Dynamic List** link.

7) Select `OEHR_PRODUCTS` for Table or View and click **Next >**.

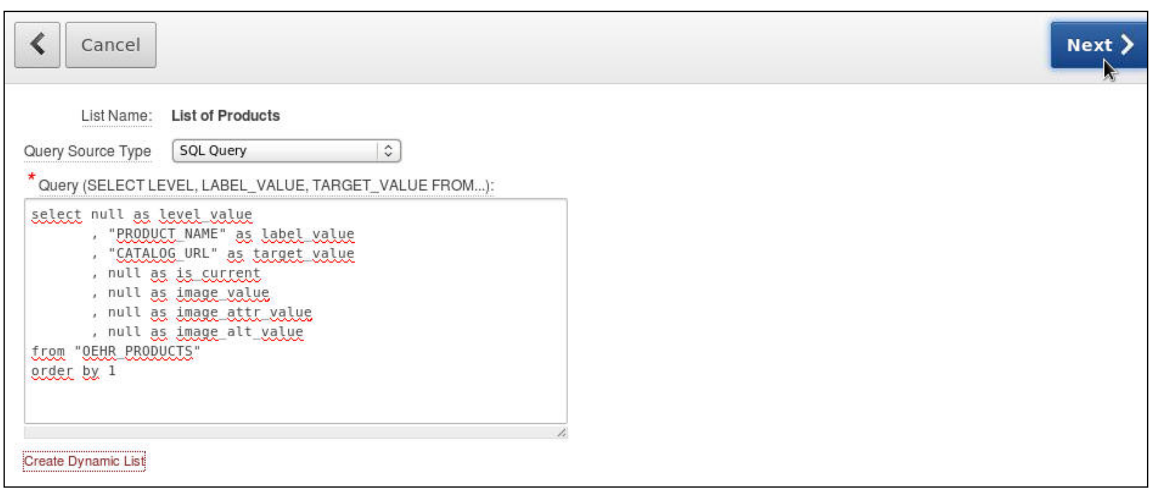
8) Select **PRODUCT_NAME** for Label Column and **CATALOG_URL** for Target Column. Click **Next >**.



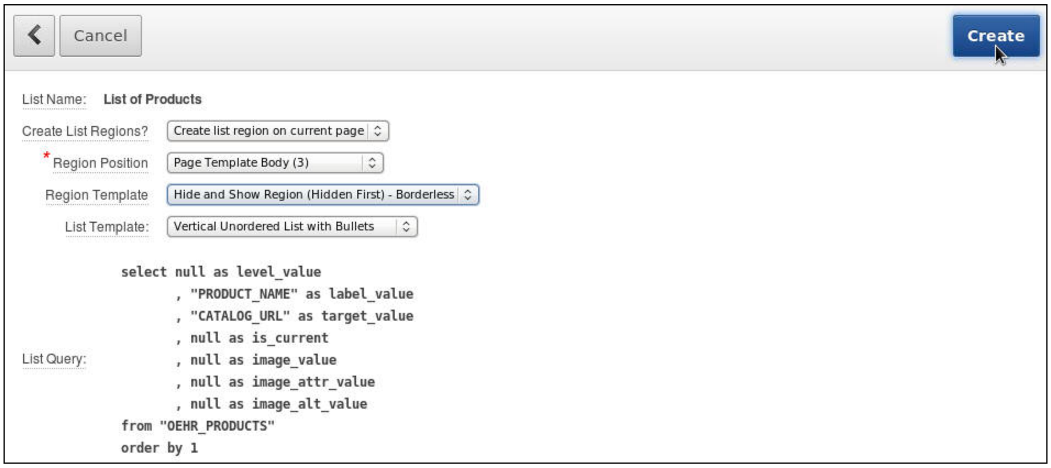
9) Click **Finish**.



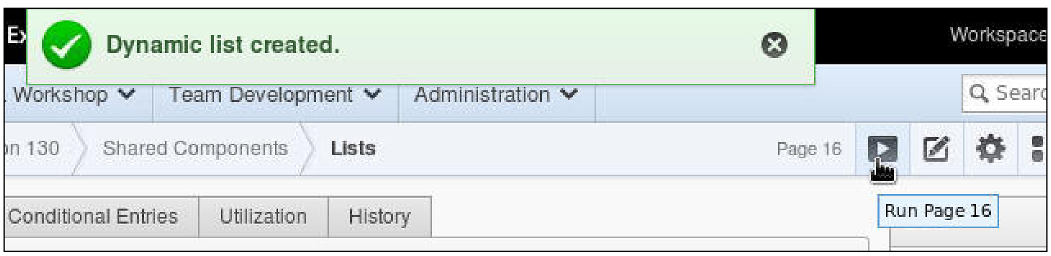
10) Click **Next >**.



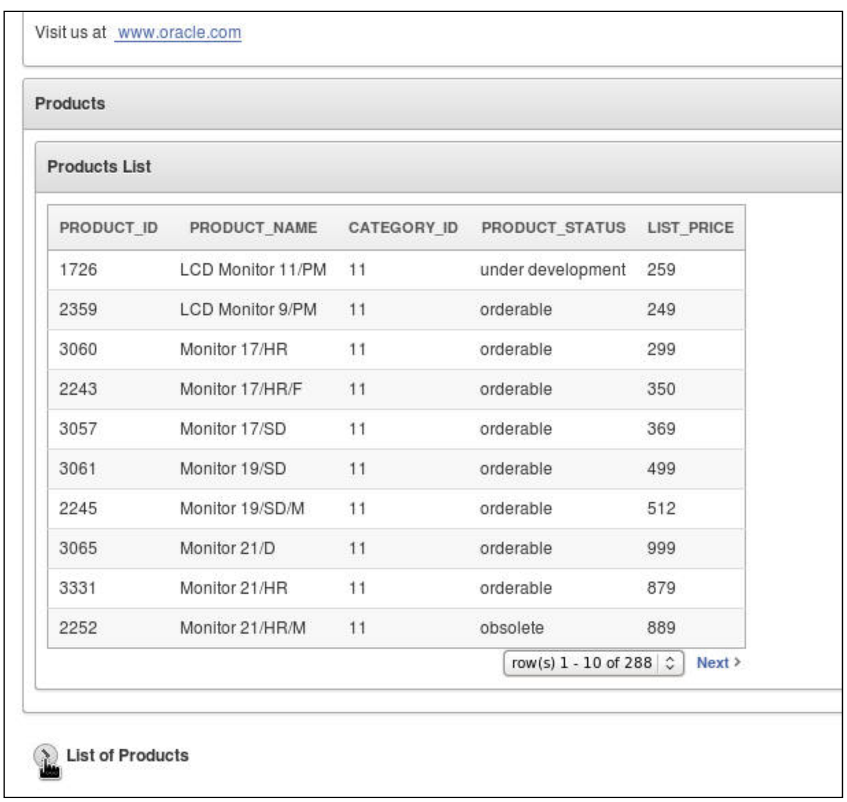
11) Select **Create list region on current page** for “Create List Regions?”, select **Page Template Body (3)** for Region Position, and select **Hide and Show Region (Hidden first) – Borderless** for Region Template. Click **Create**.



12) The Dynamic list is created. Click the **Run Page** icon.



13) Scroll to the bottom of the page. You had selected the region as a hide and show region. Click the right arrow to show the list.



14) The list is displayed. Click the **Application <*n*>** link on the Developer toolbar.

Solution 12-2: Creating and Editing Standard Tabs

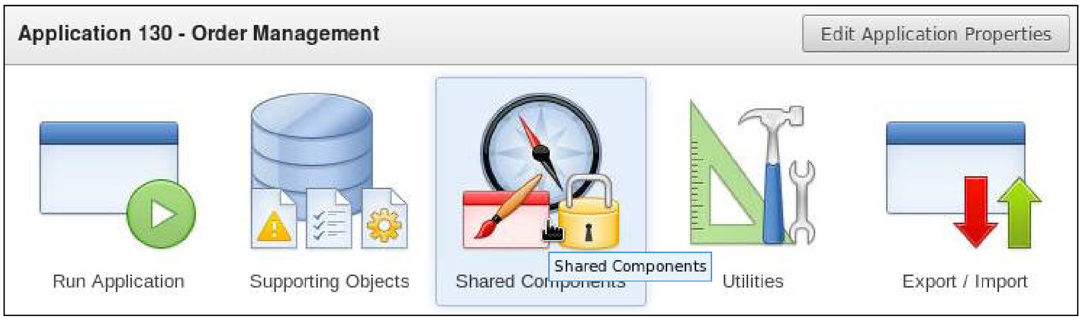
Overview

In this practice, you create tabs for the application and assign pages to the tabs. You also create a Help page and a Help tab to add to the existing tab set.

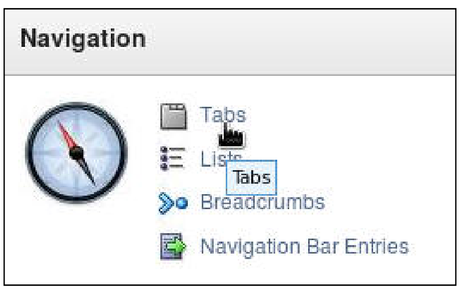
Tasks

a. Create a tab called Orders for the Order Management application. It should be the current tab for the List of Orders page. Sequence the Orders tab before the Products tab.

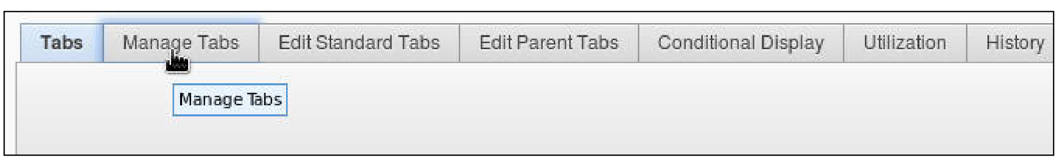
1) Click the **Shared Components** icon.



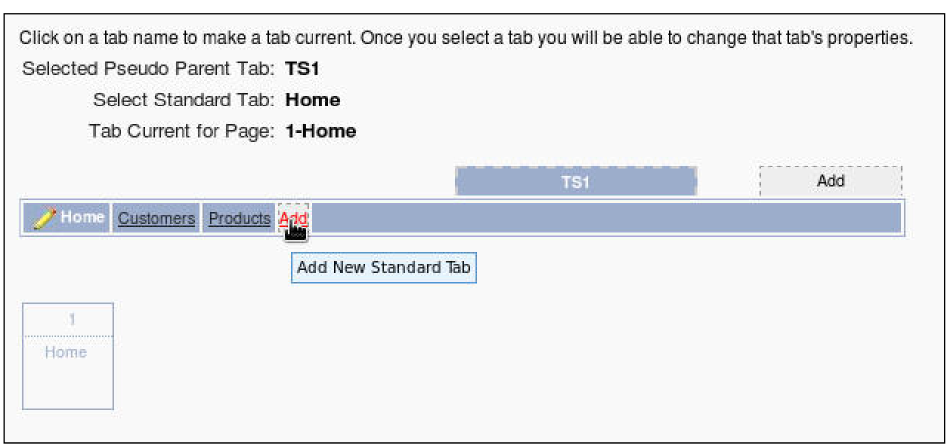
2) In the Navigation region, click **Tabs**.



3) Click the **Manage Tabs** tab.

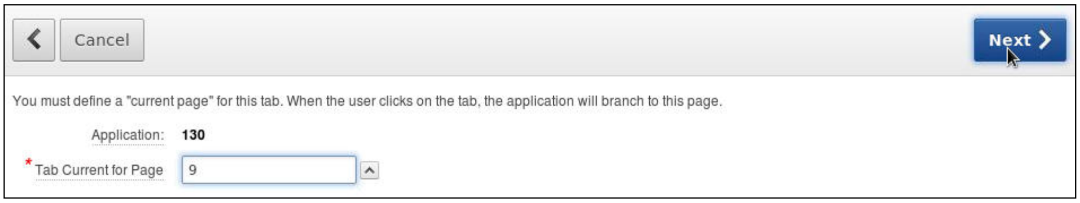


4) Click the **Add** link next to Products to add a New Standard tab.



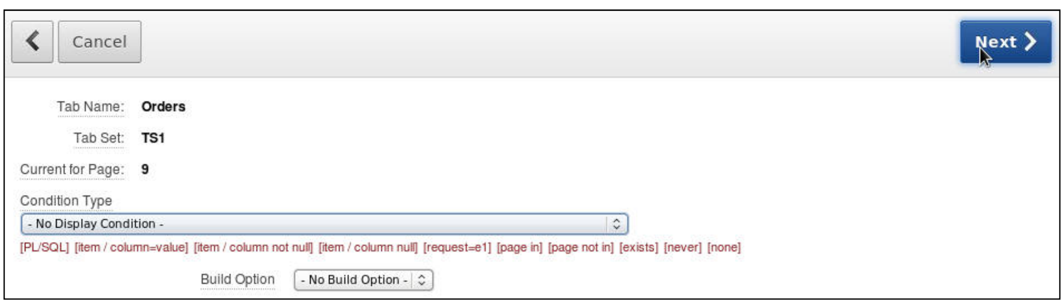
5) Enter **Orders** for Tab Label and click **Next >**.

6) For "Tab Current for Page," click the up arrow and select the **List of Orders** page link. Click **Next >**.



7) Enter **25** for sequence number and click **Next >**.

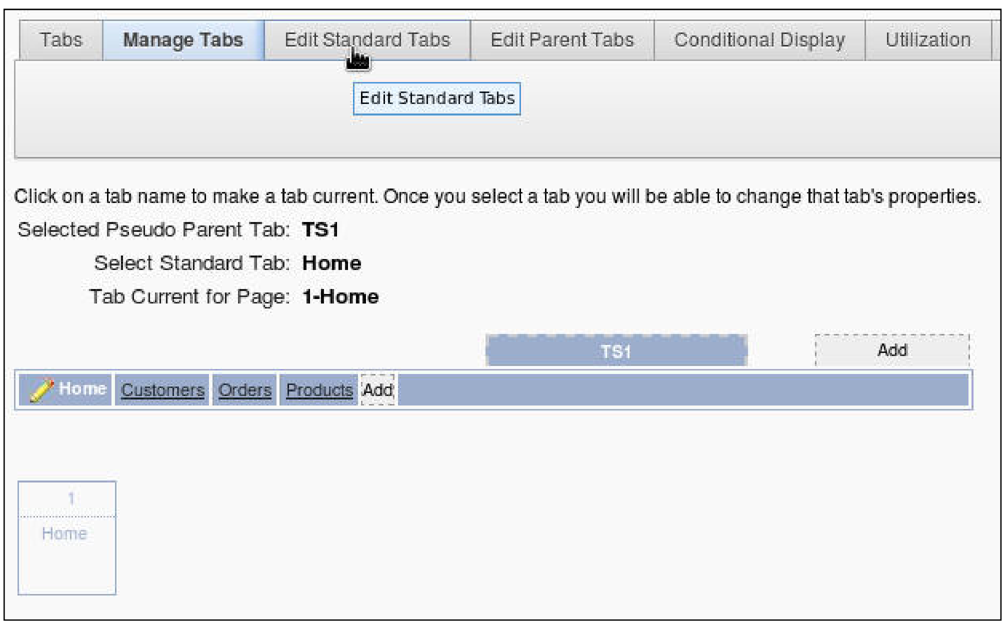
8) Accept the defaults and click **Next >**.



9) Review the details and click **Create Tab**.

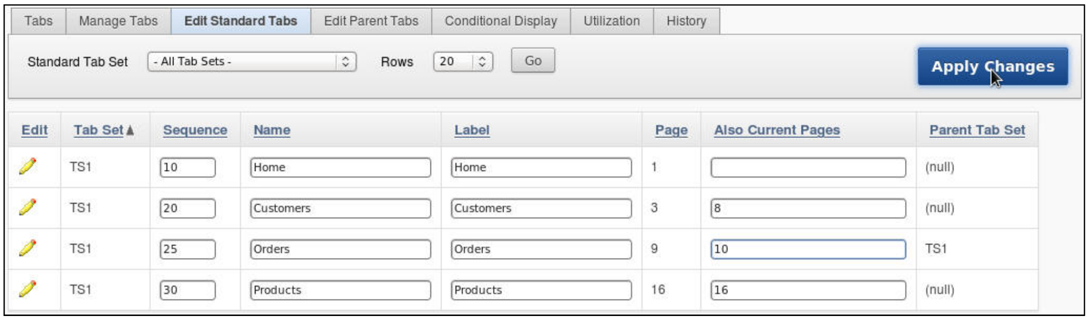
b. Assign the Master Detail page to the Orders tab. Run the application and confirm that the tab is created successfully.

1) Click the **Edit Standard Tabs** tab.

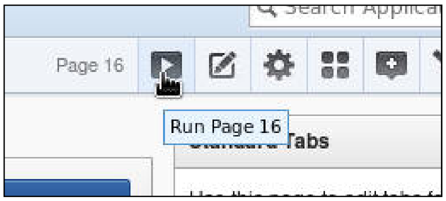


2) Make the following changes (in bold) and click **Apply Changes**.

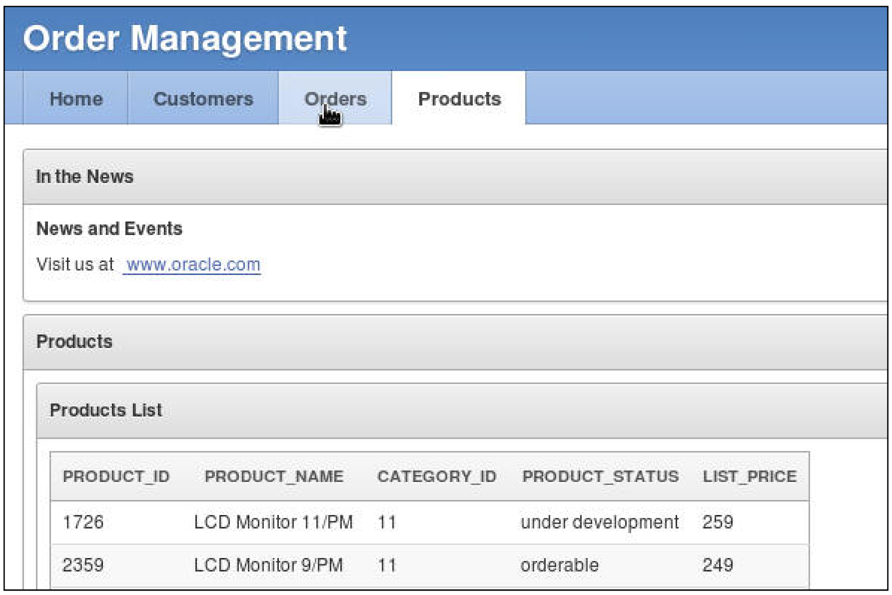
Name	Also Current Pages
Home	
Customers	**8**
Orders	**10**
Products	16



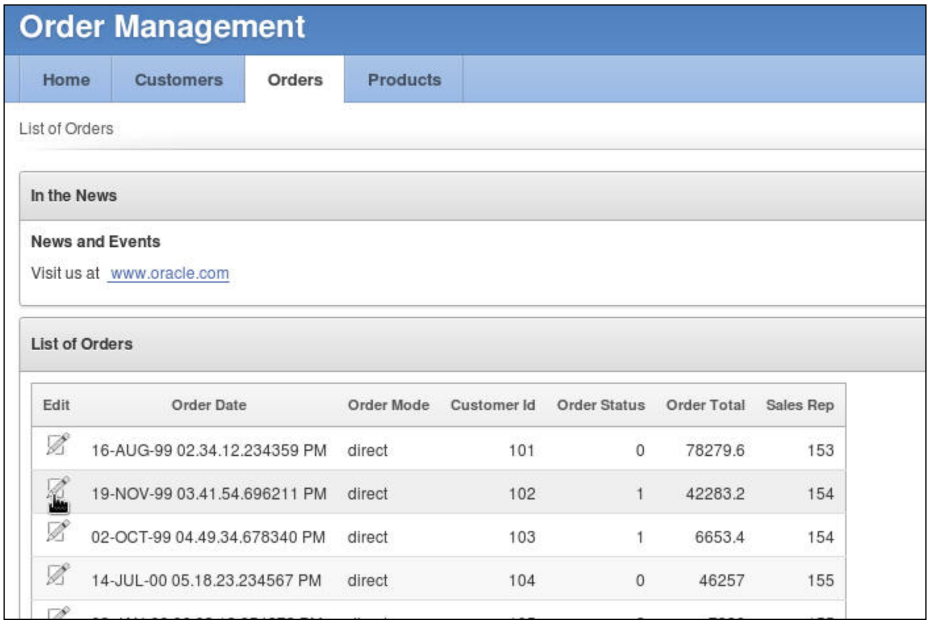
3) Click the **Run Page** icon.



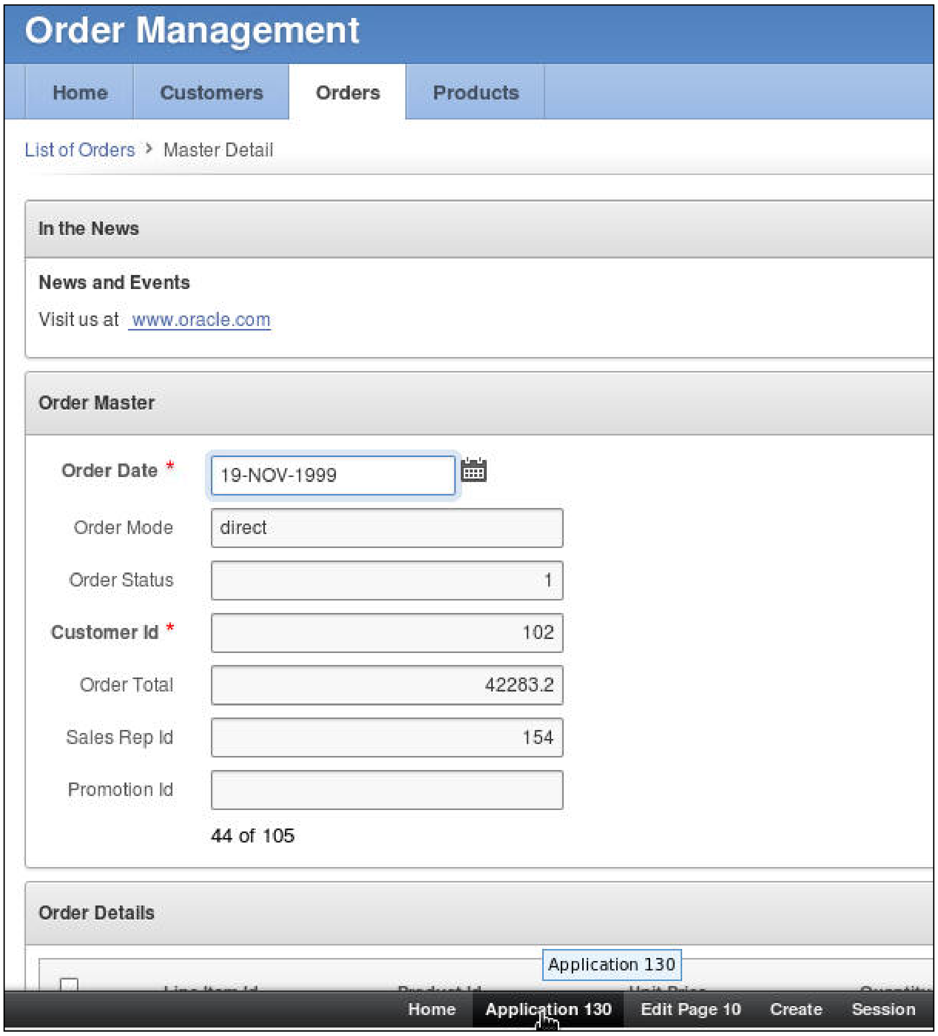
4) The Orders tab is displayed. Click the **Orders** tab.



5) The List of Orders page is displayed. Click the **Edit** icon for one of the order entries.

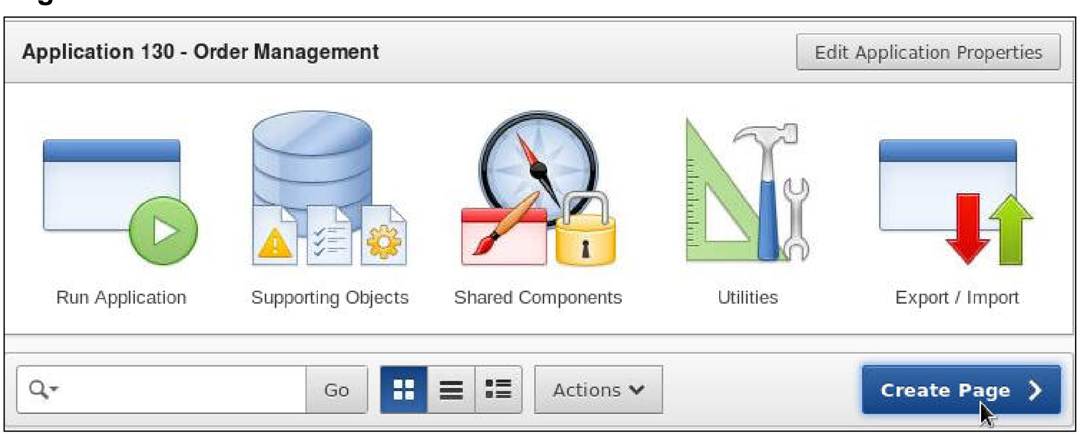


6) The Master Detail page (10) is displayed. Notice that the Orders tab is still active. Click the **Application <*n*>** link on the Developer toolbar.

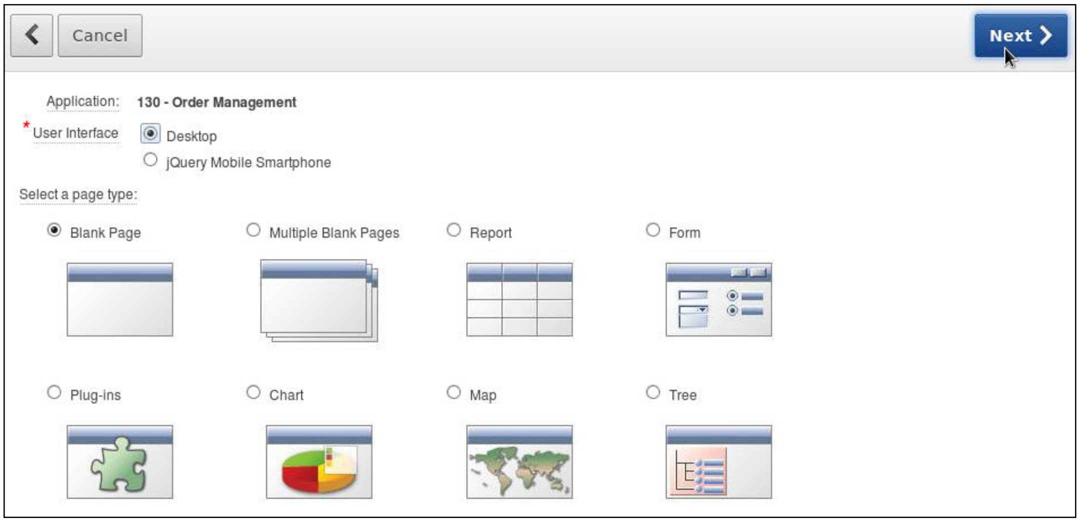


c. Create a page called Help, and then add a new tab called Help, to the existing tab set. The new Help tab will link to the new Help page.

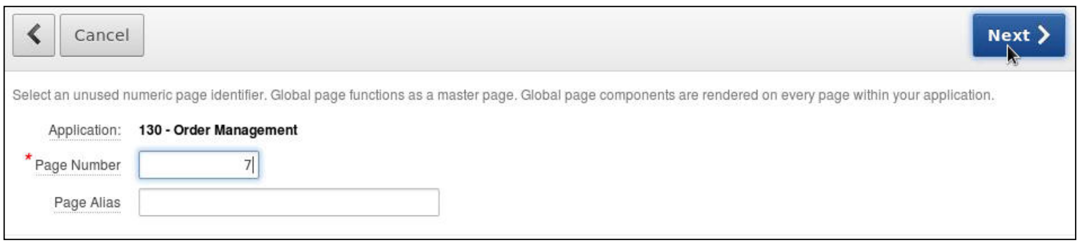
1) You want to create a new page that will contain the help information. Click **Create Page >**.



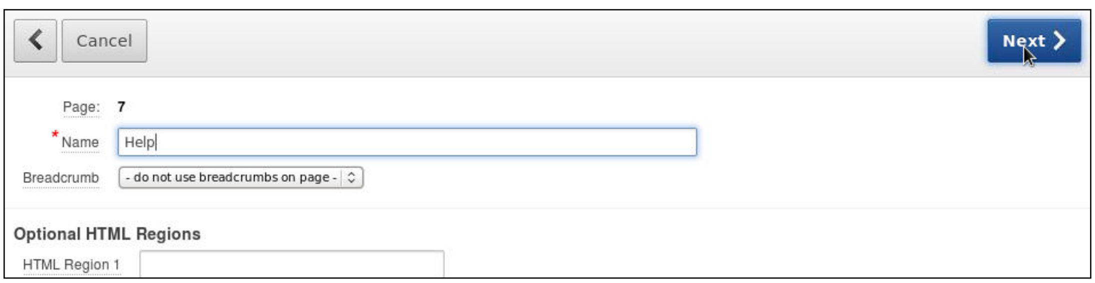
2) Select **Blank page** and click **Next >**.



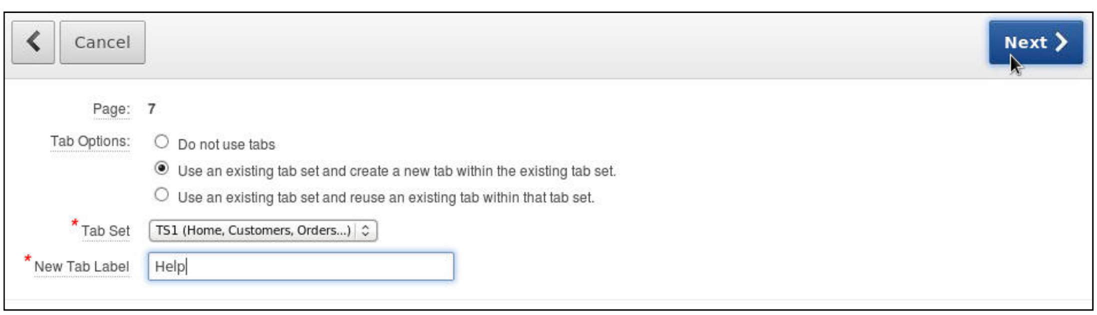
3) Accept the default page number and click **Next >**.



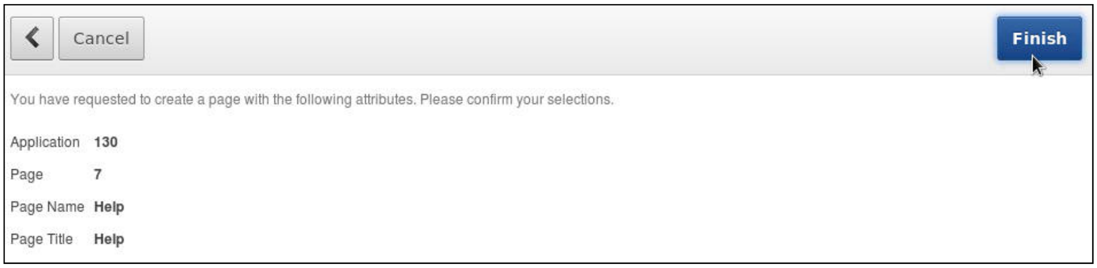
4) For Name, enter **Help** and click **Next >**.



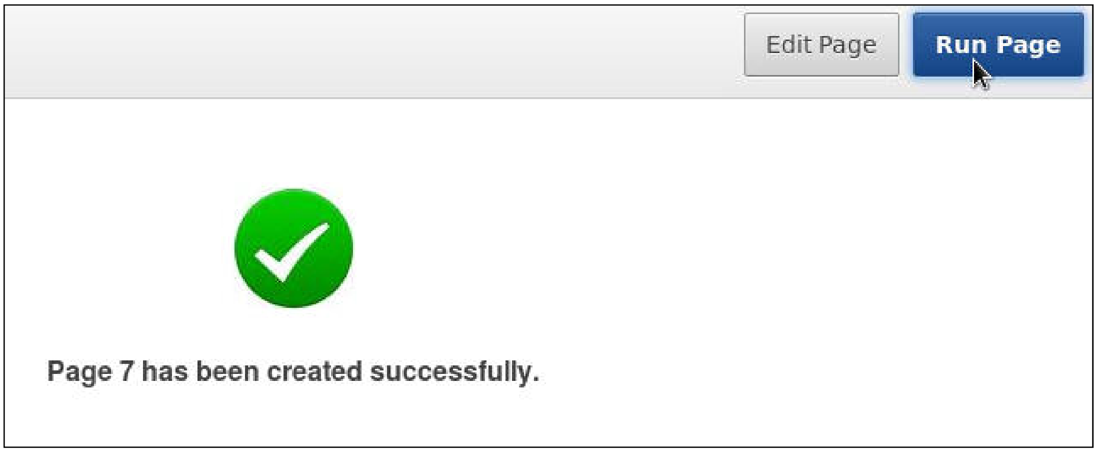
5) Select the **Use an existing tab set and create a new tab within the existing tab set** option. Enter `Help` for New Tab Label, and then click **Next >**.



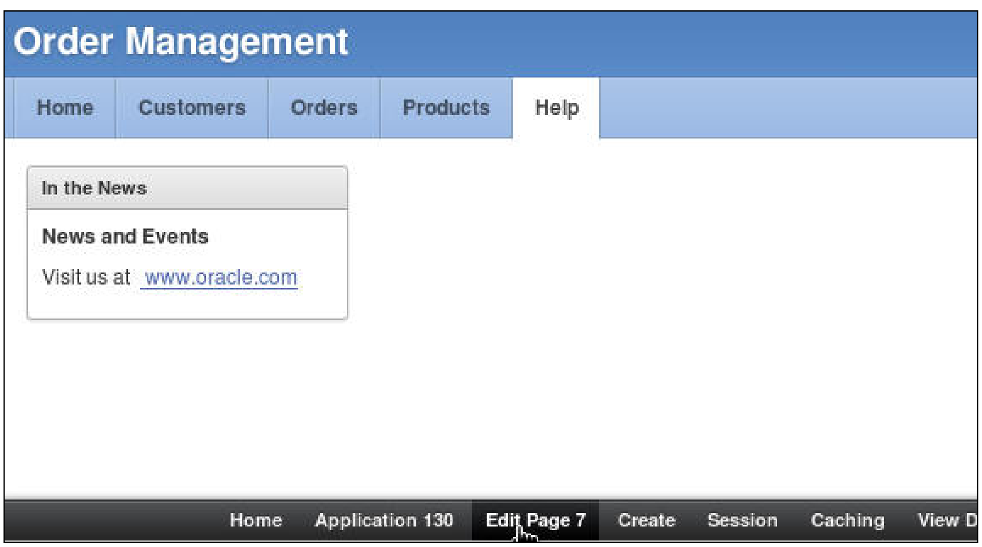
6) Click **Finish**.



7) Run the page to see the new tab. Click **Run Page**.



8) The tab is displayed. In the next practice, you add a region to the Help page and add it as a navigation bar entry. Click the **Edit Page** link on the Developer toolbar.



Solution 12-3: Adding Navigation Bar Entries

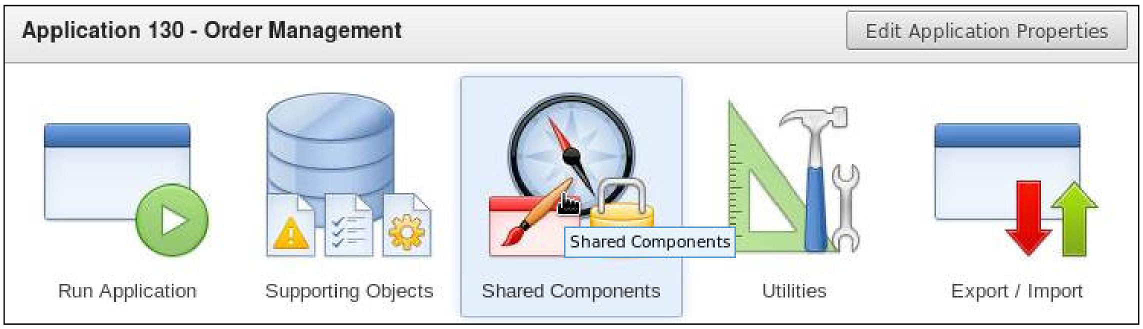
Overview

In this practice, you add a navigation bar entry that, when clicked, will show the page-level help. You also modify the name of an existing navigation bar entry and add another navigation bar entry that is displayed on all pages except the page that it navigates to.

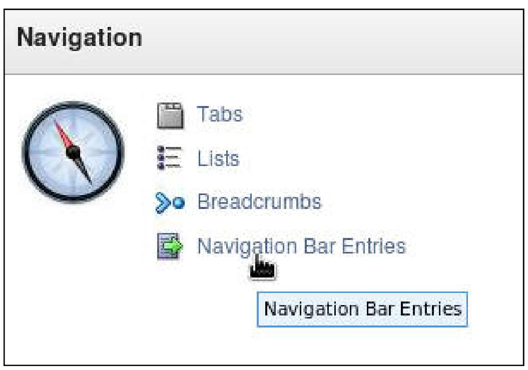
Tasks

a. Create a navigation bar entry called Help that navigates to the help page. Run the page and click the Help navigation bar entry.

1) Click the **Shared Components** icon.



2) Under Navigation, click **Navigation Bar Entries**.

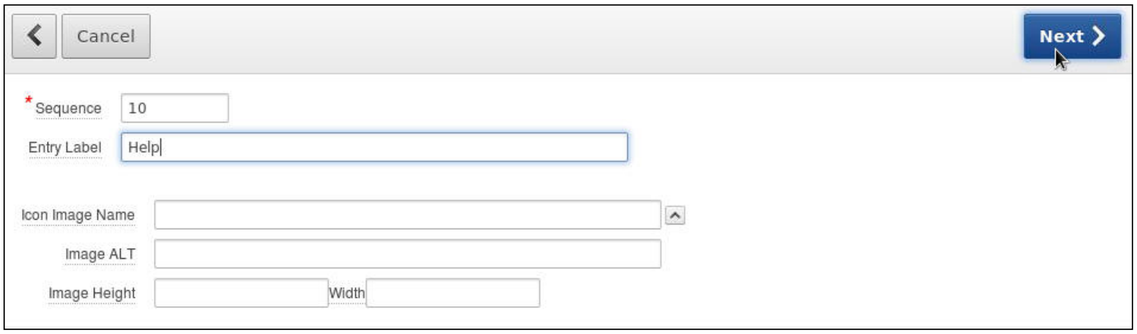


3) Click **Create >**.

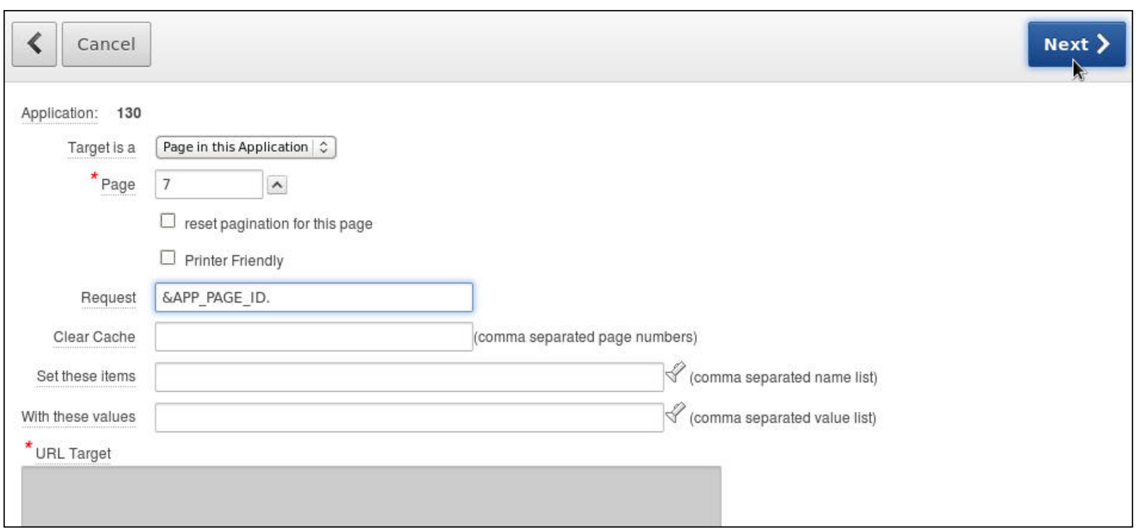
4) Make sure that **From Scratch** is selected and click **Next >**.

5) Select **Navigation to URL** and click **Next >**.

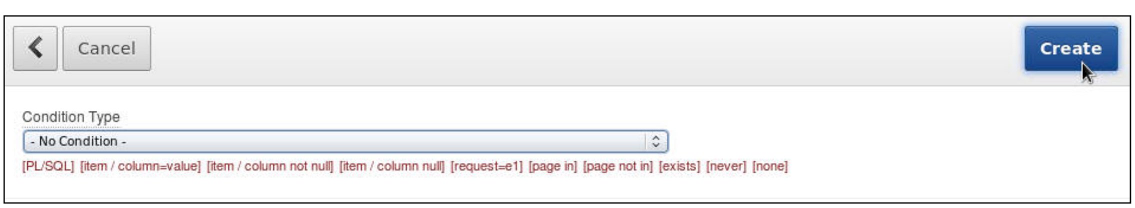
6) For Entry Label, enter **Help** and click **Next >**.



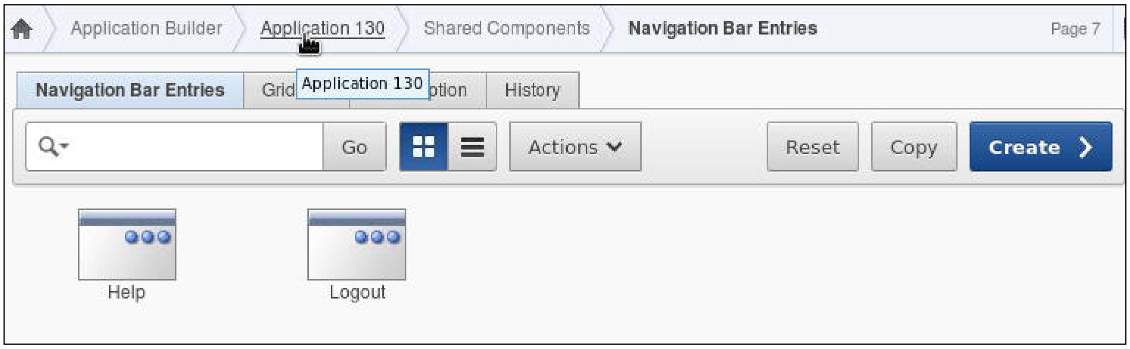
7) Click the icon to the right of the Page field and click the Help page link. In Request, enter **`&APP_PAGE_ID.`** (including the period) and click **Next**.
Note: By specifying the `&APP_PAGE_ID.` bind variable as the request, you are instructing the Oracle Application Express engine to display the help text for the current page when the user clicks the navigation bar entry link.



8) Click **Create**.

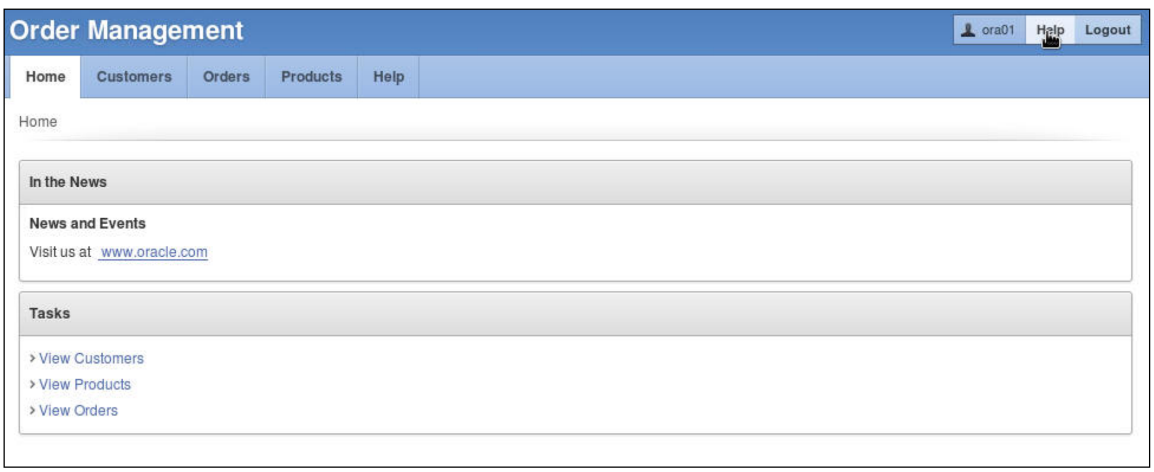


9) Click the **Application <n>** breadcrumb.

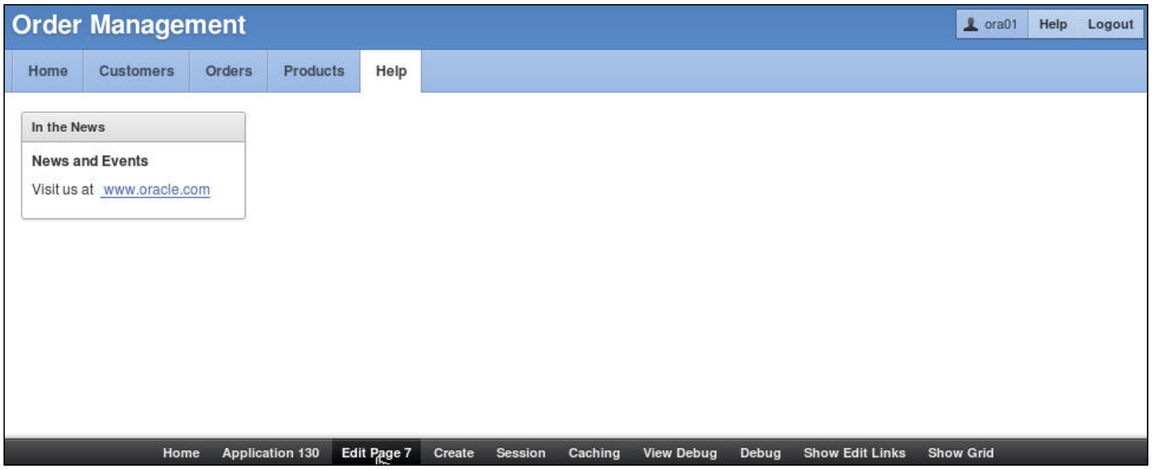


10) Click **Run Application**.

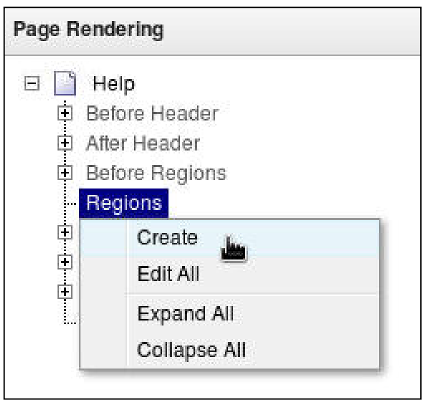
11) Click the **Help** navigation bar entry.



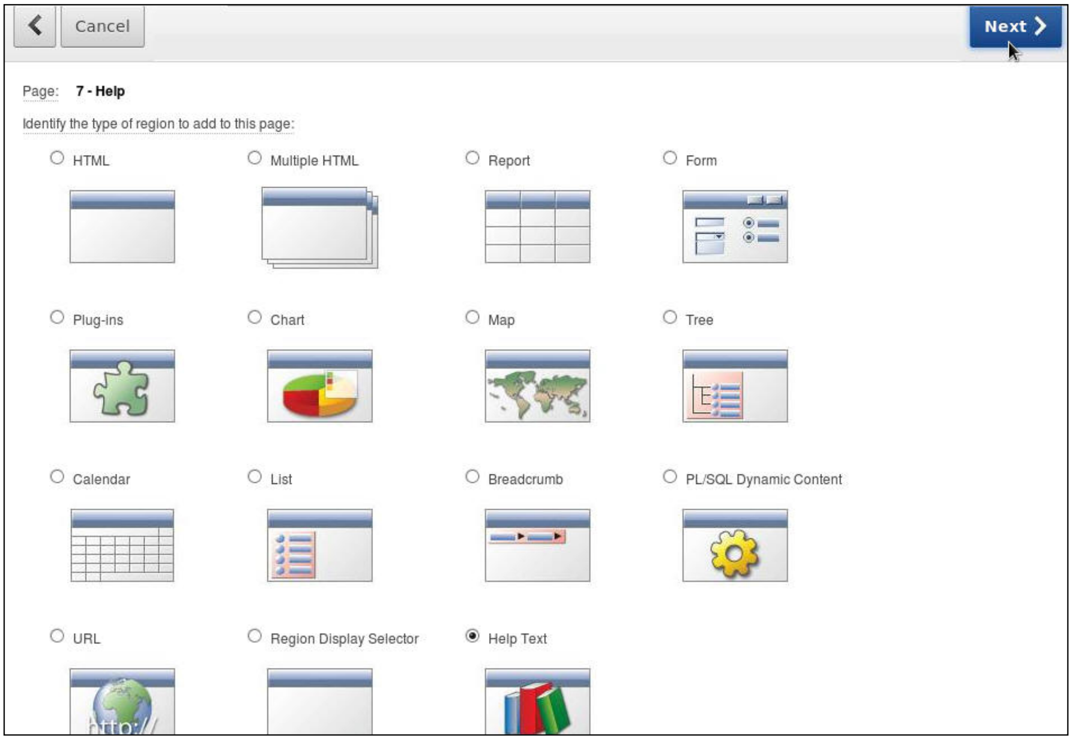
12) The page help page is displayed. You want to add a help text region on the Help page so that the help for the active page will be displayed. Click the **Edit Page** link on the Developer toolbar.



13) Right-click **Regions** and select **Create**.

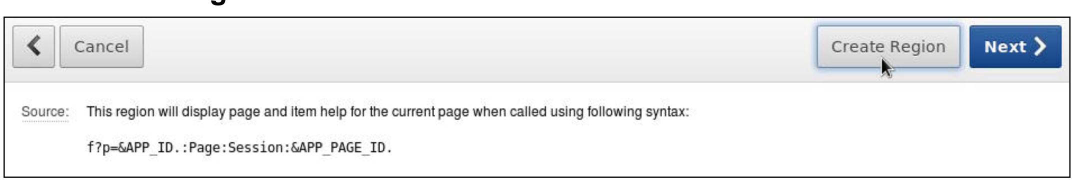


14) Select the Region Type **Help Text** and click **Next >**.



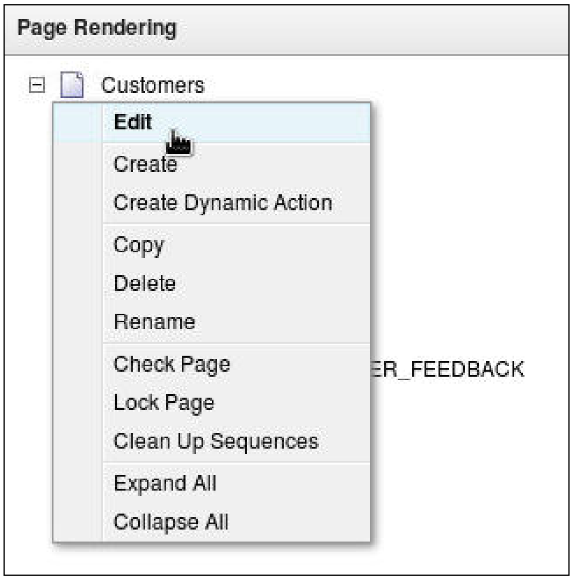
15) Enter **Help** for Title and click **Next >**.

16) Click **Create Region**.



17) You need to enter some help text for a page. Enter **3** (or the page number for the Customers page) in the Page area and click **Go.**

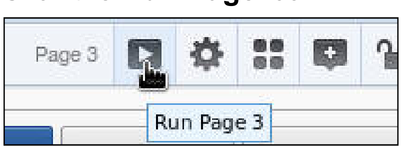
18) Right-click **Customers** page node and select **Edit**.



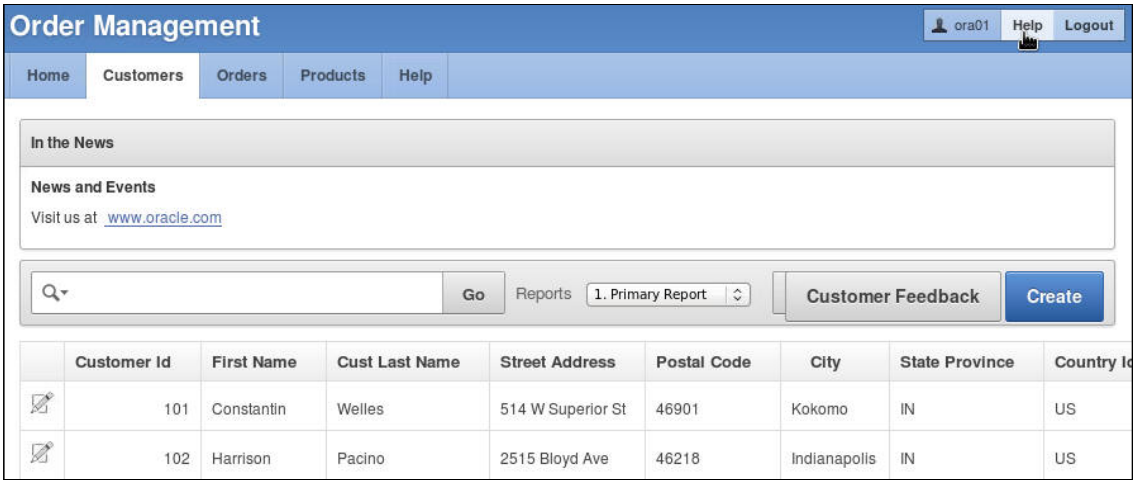
19) Click the **Help** subtab.

20) Enter something in the Help Text area and click **Apply Changes**.

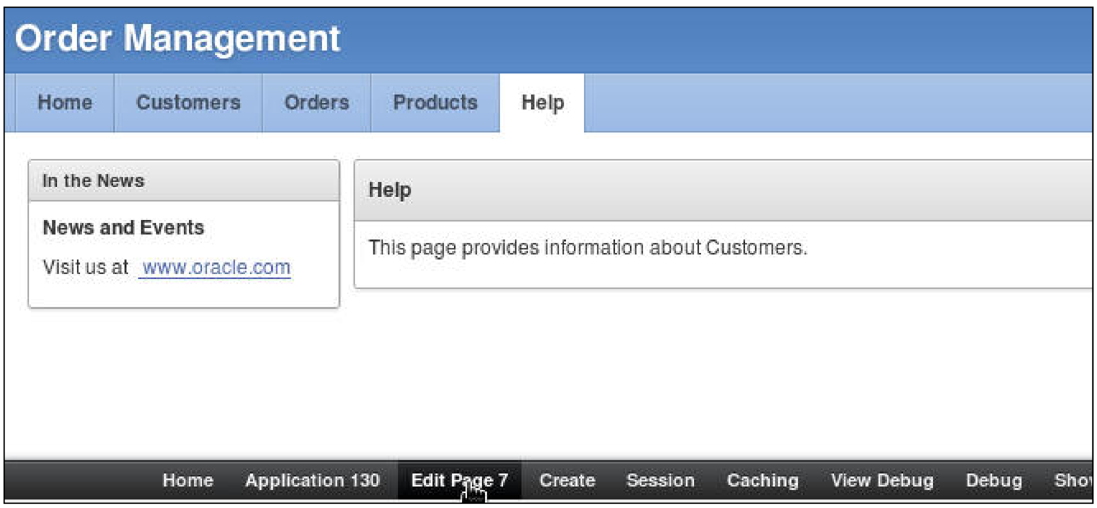
21) Click the **Run Page** icon.



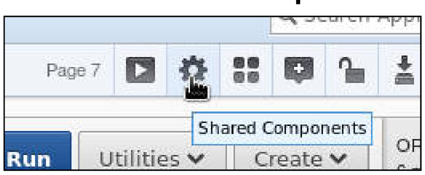
22) On the Customers page, select the **Help** Navigation Bar entry.



23) Notice that the Help text from the page definition is displayed. Click the **Edit Page** link on the Developer toolbar.

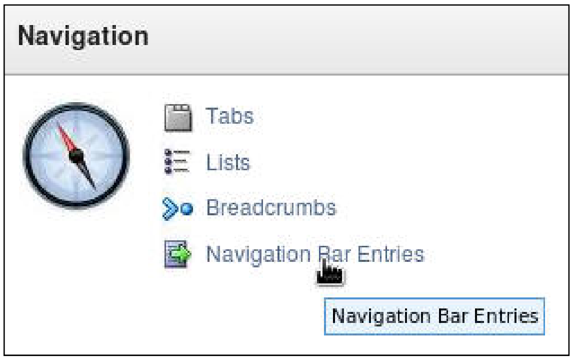


24) Click the **Shared Components** icon.

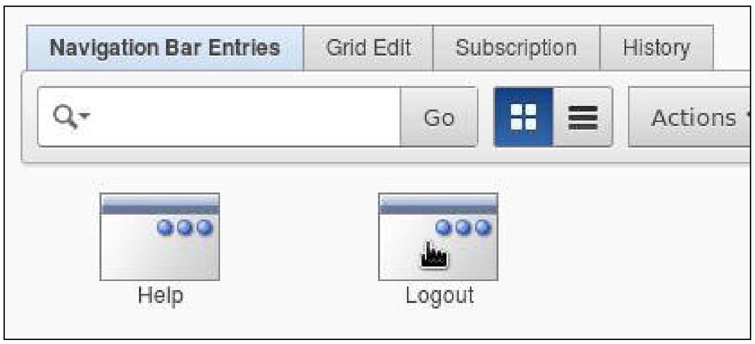


b. Change the name of the navigation bar entry for Logout to Sign Off.

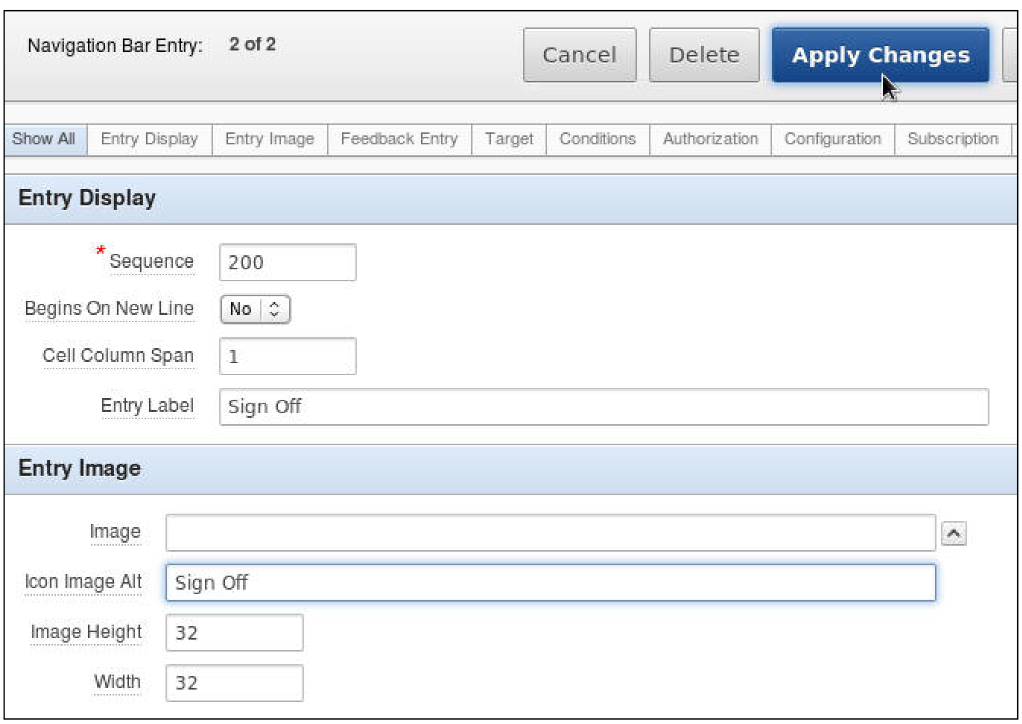
1) In the Navigation region, click the **Navigation Bar Entries** link.



2) Click the **Logout** icon.



3) In the Image Attributes region, change the Entry Label and Icon Image Alt text to **Sign Off** and click **Apply Changes**.



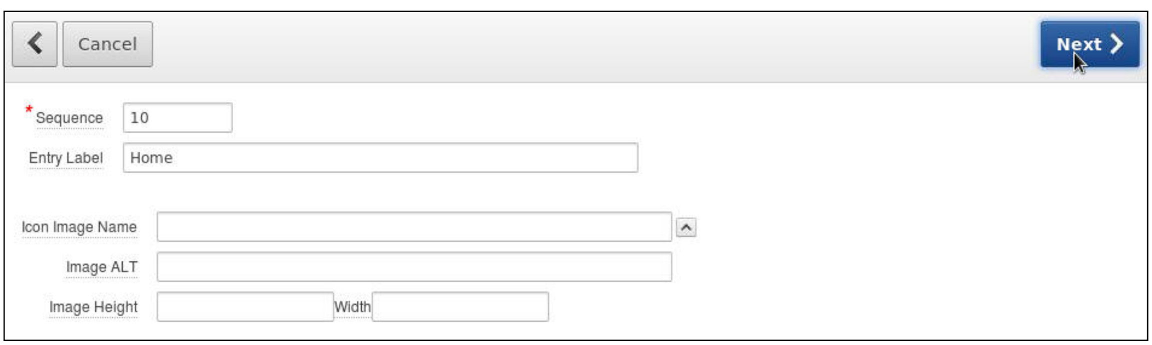
c. Add a navigation bar entry called Home. Display the navigation bar entry on all pages except the Home page.

1) Click **Create >**.

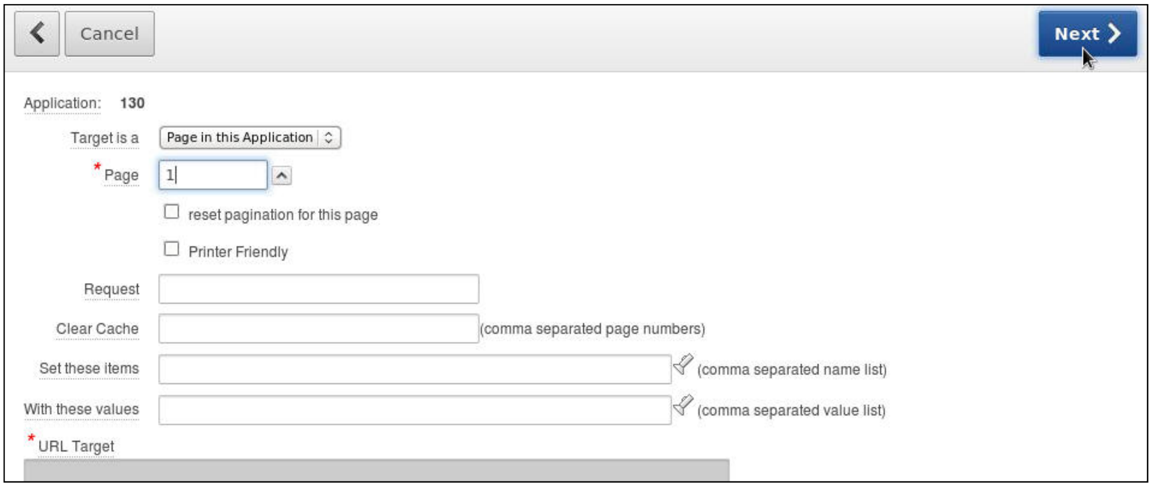
2) Ensure that the **From Scratch** option is selected. Click **Next >**.

3) Select **Navigation to URL** and click **Next >**.

4) For Entry Label and enter **Home**. Then click **Next >**.



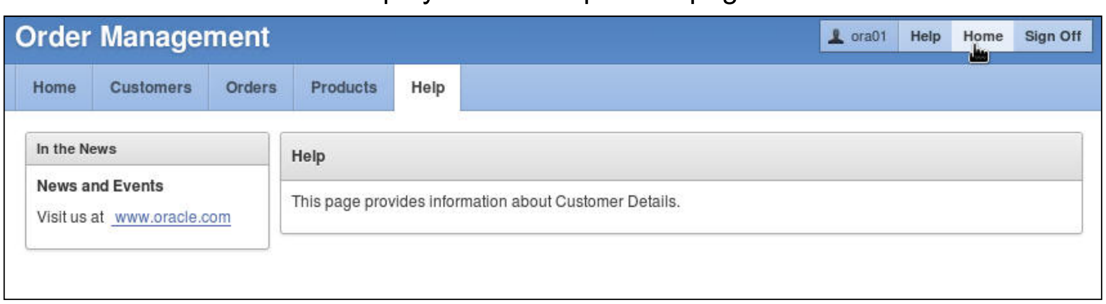
5) Click the icon next to Page and select **Home**. Click **Next >**.



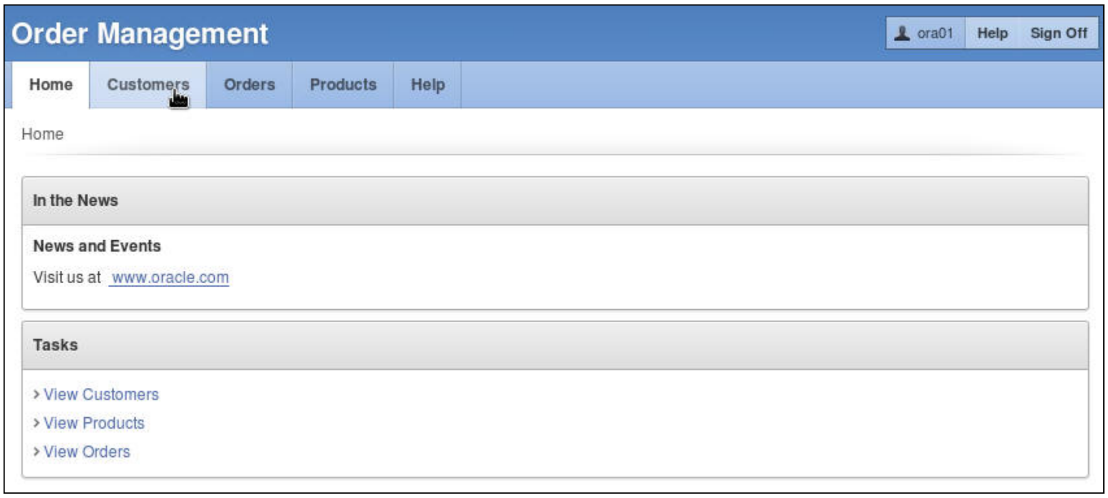
6) From the list of links below the Condition Type drop-down list, click **[page not in]**. In Expression 1, enter `1` and click **Create**.

7) Click the **Run Page** icon.

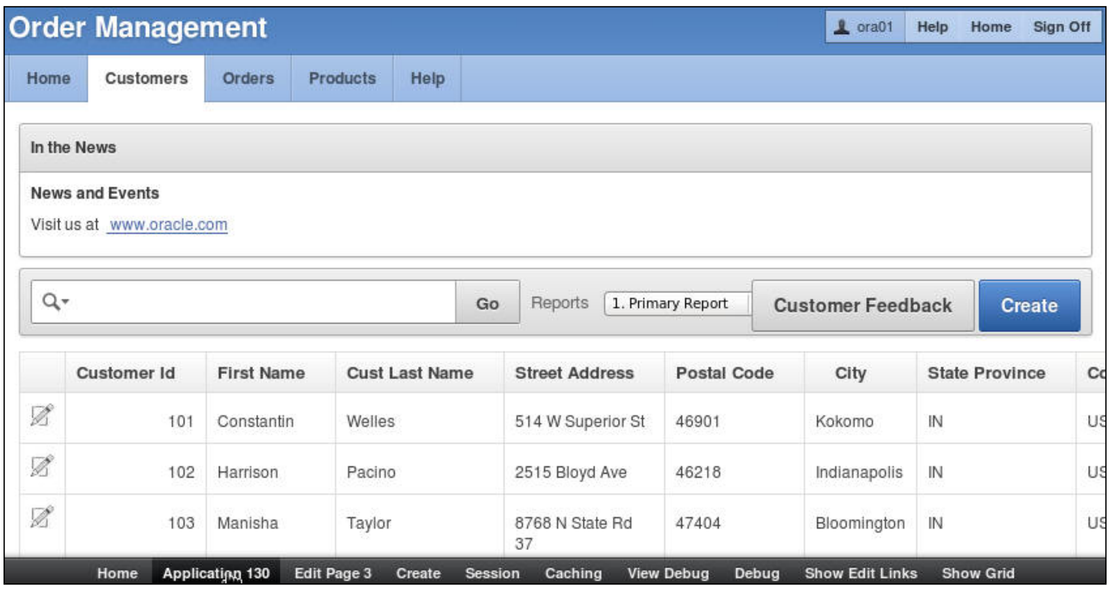
8) Note that the Home link is displayed at the top of the page. Click the **Home** tab.



9) You see that the Home link is no longer displayed in the Navigation bar list. Click the **Customers** tab.



10) Notice that the Home navigation bar entry is displayed again.



Solution 12-4: Adding Breadcrumbs and Tabs to an Existing Page

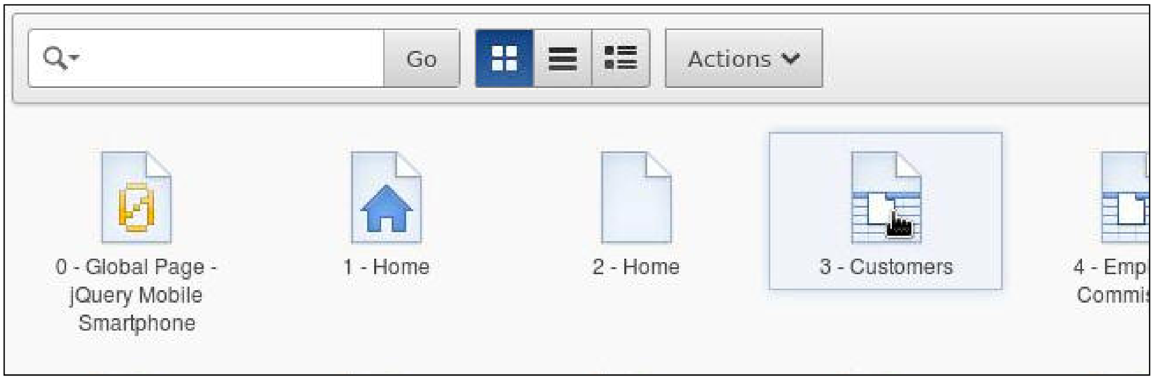
Overview

In this practice, you create a breadcrumb and add a tab set to an existing page.

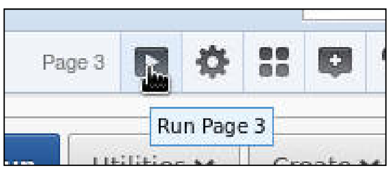
Tasks

a. Edit the existing breadcrumb for the Order Management application and create a breadcrumb entry for the Customers page. Run the customers page. Is the breadcrumb displayed on the page?

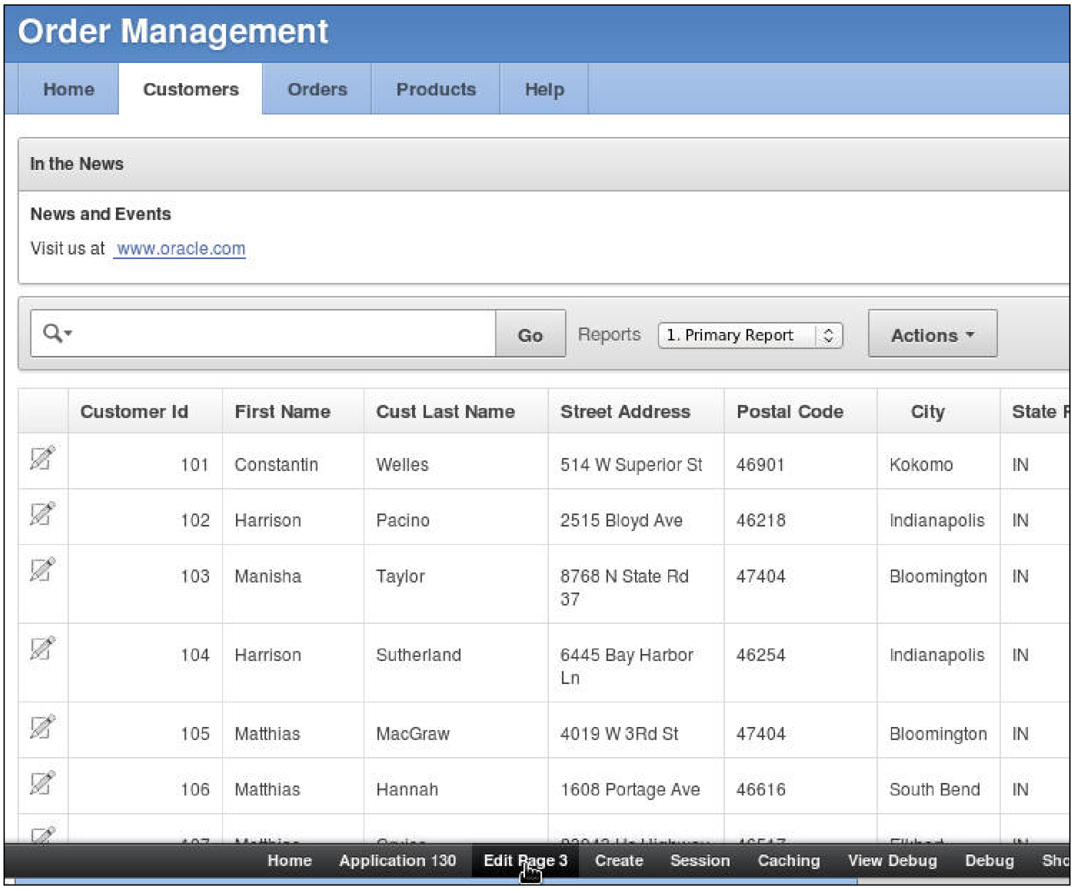
1) From the Order Management application page, select the **Customers** page.



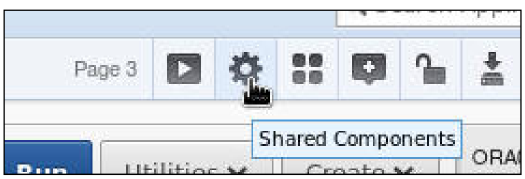
2) Click the **Run Page** icon.



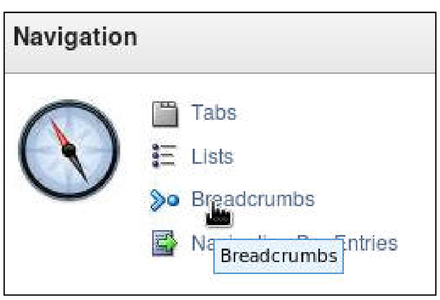
3) The Customers page is displayed. Notice there is no Breadcrumb on this page. Click the **Edit Page** link on the Developer toolbar.



4) Click the **Shared Components** icon.



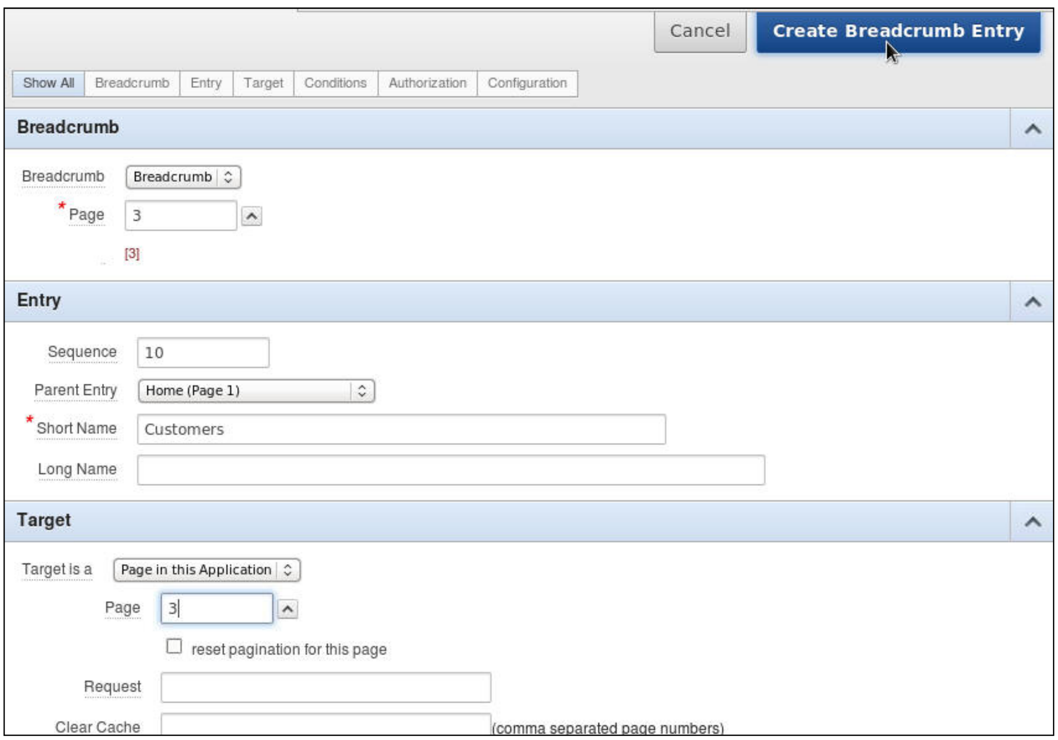
5) Click the **Breadcrumbs** link.



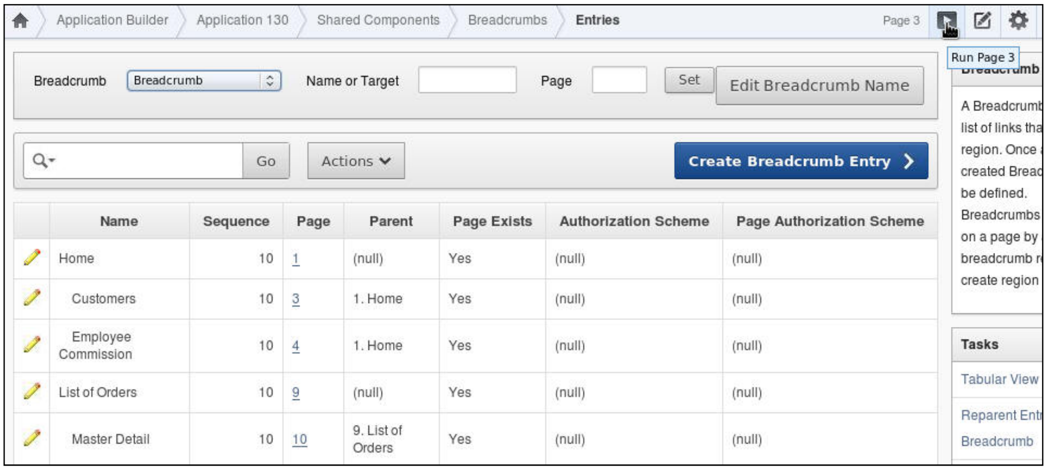
6) Click the **Breadcrumb** icon.

7) Click **Create Breadcrumb Entry >**.

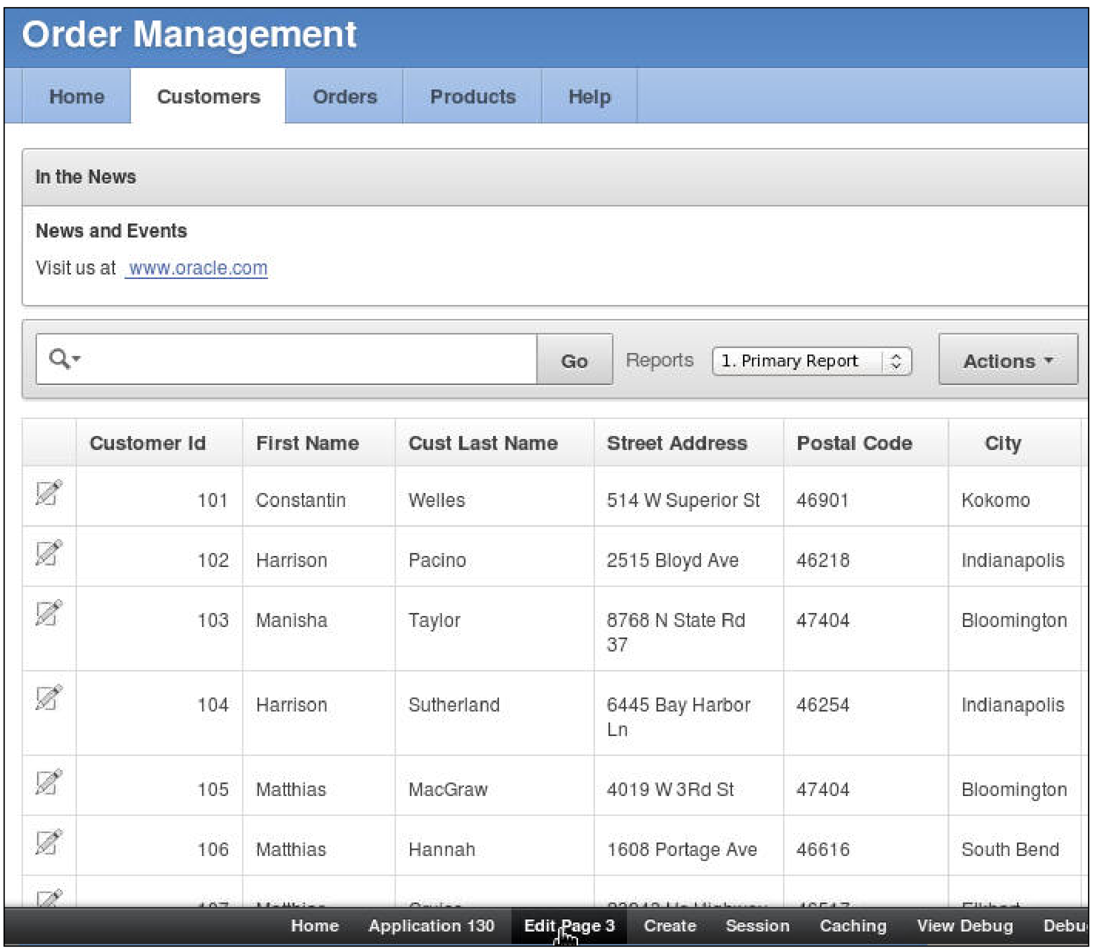
8) Click the **[3]** quick pick for Page on the Breadcrumb tab. Select **Home (Page 1)** for Parent Entry and enter **Customers** for Short Name under Entry. Enter **3** for Page under Target and click **Create Breadcrumb Entry**.



9) Click the **Run Page** icon.

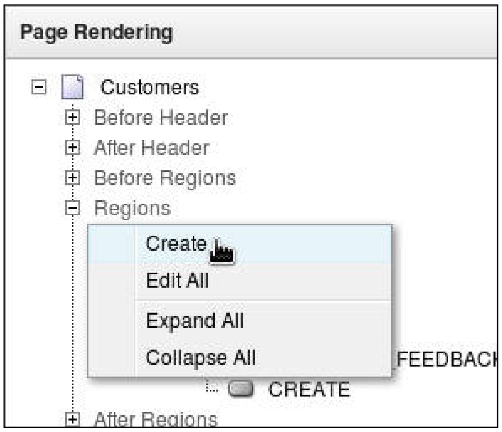


10) Note that the breadcrumb you created is not reflected. This is because there is no breadcrumb region on this page. Click the **Edit Page** link on the Developer toolbar.

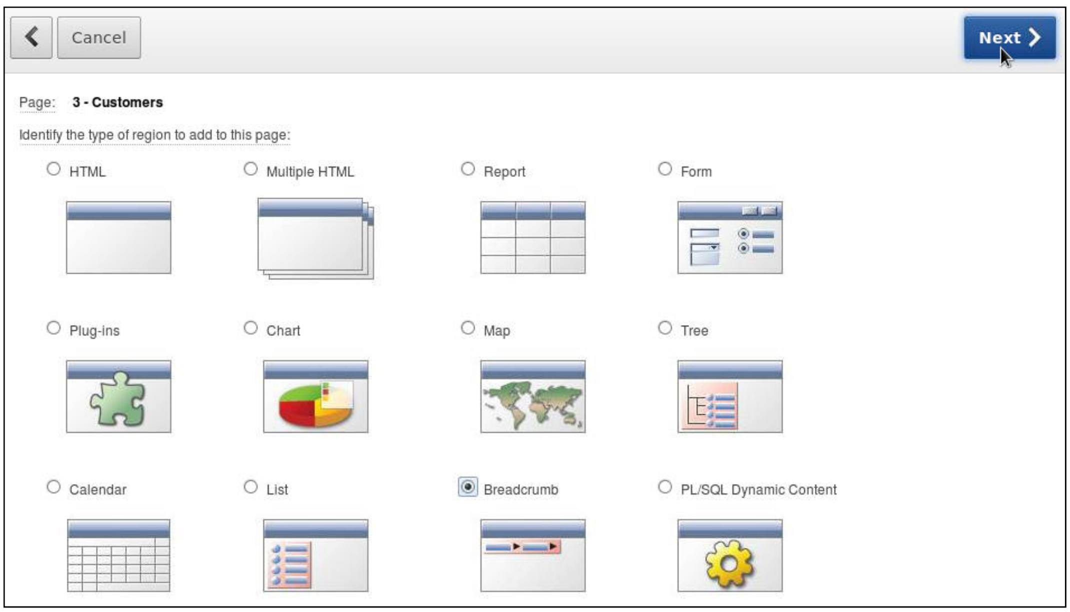


b. Create a breadcrumb region on the Customers page and run the page.

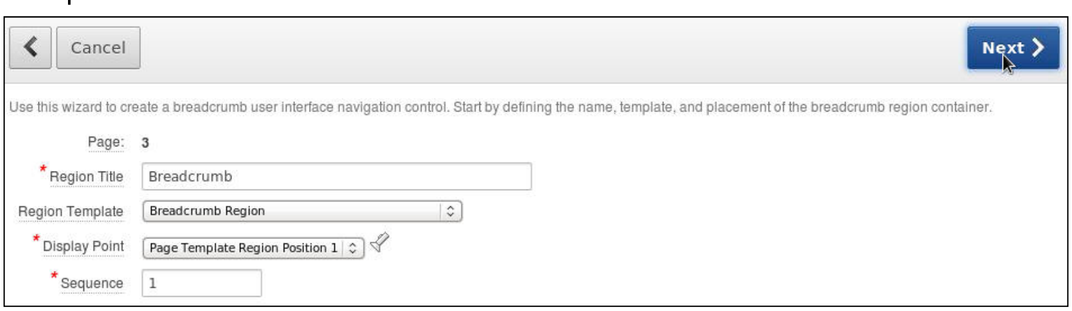
1) Right-click Regions and select **Create**.



2) Select the **Breadcrumb** option and click **Next >**.



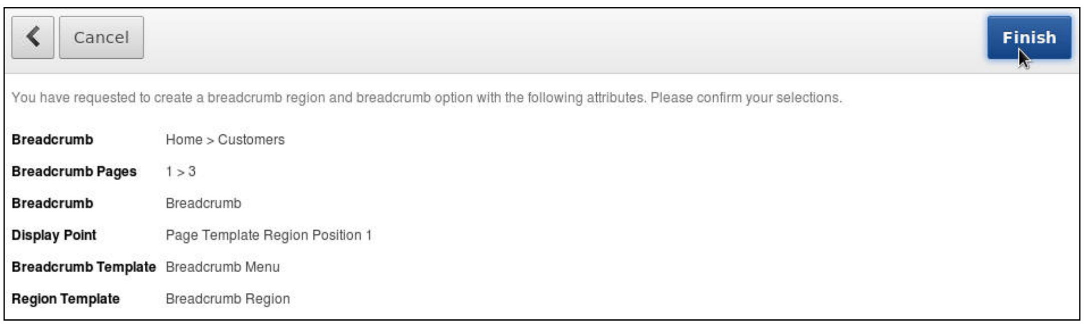
3) Accept the defaults and click **Next >**.



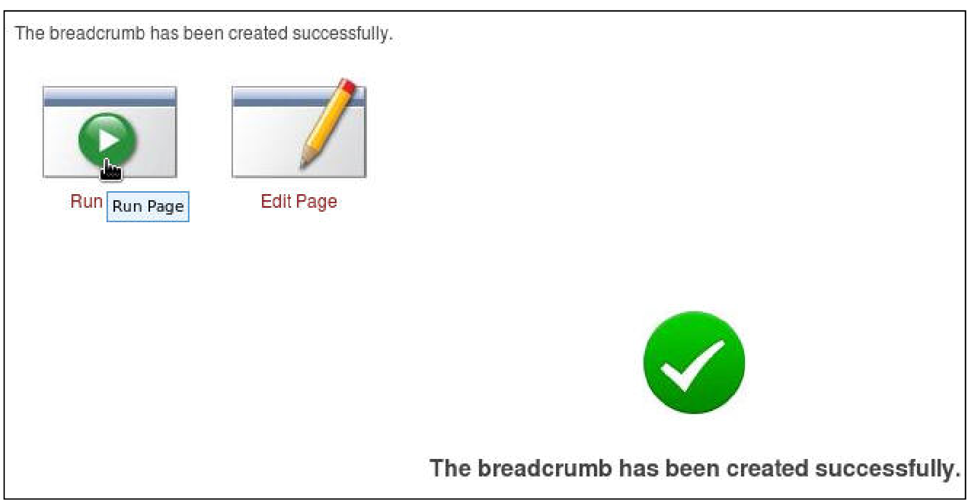
4) Accept the defaults and click **Next >**.

5) Review the details and click **Next >**.

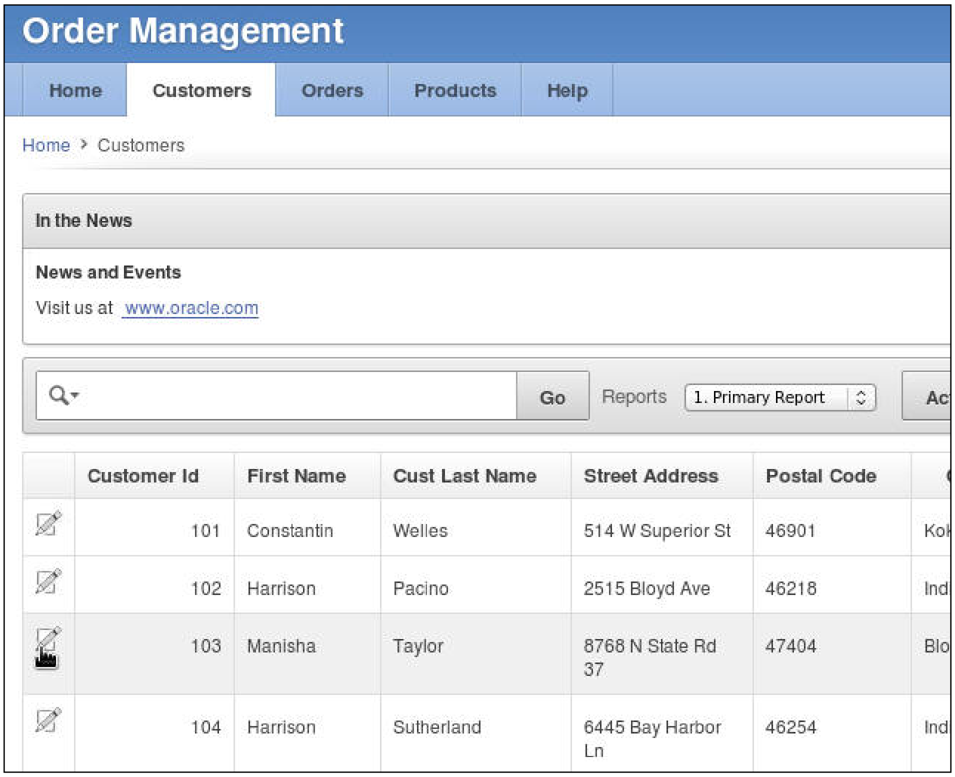
6) Click **Finish**.



7) Click **Run Page** to view the page.

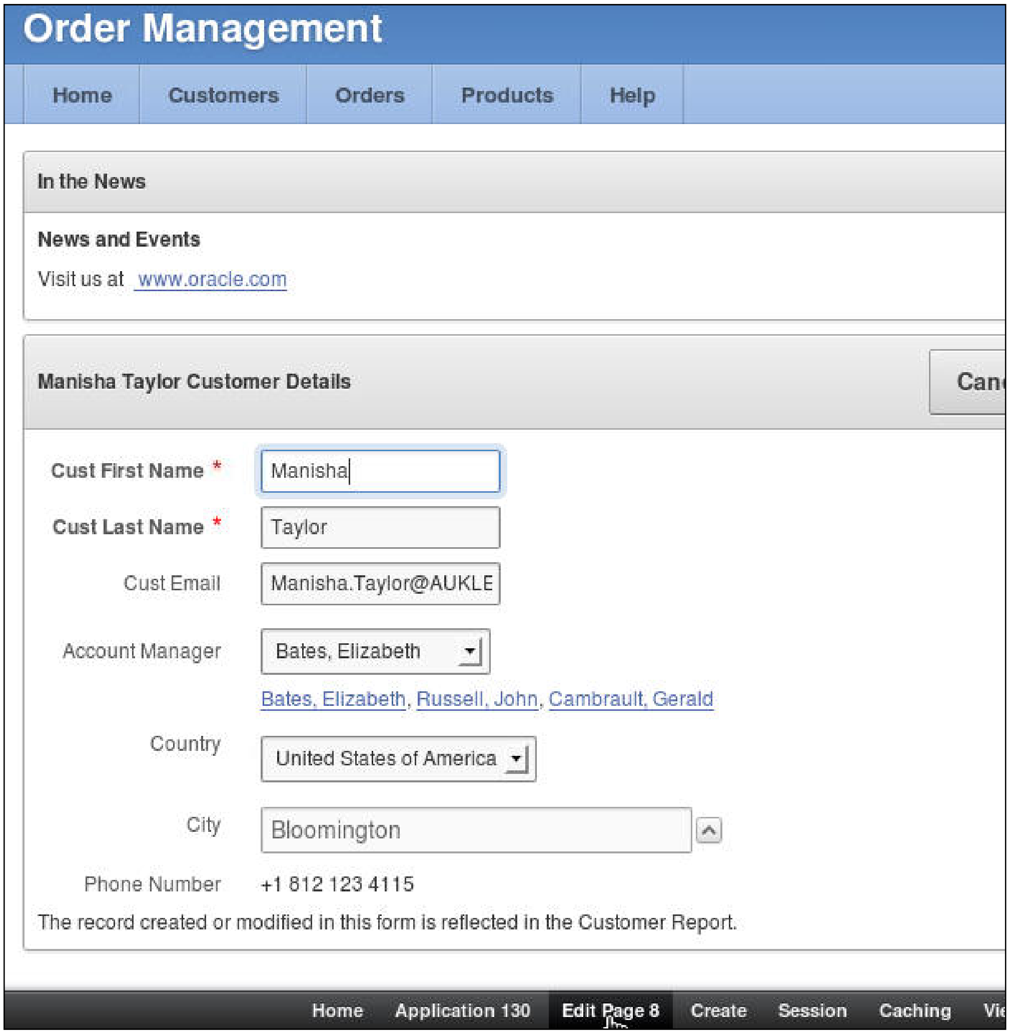


8) Note that now the breadcrumb is seen. Click the edit icon next to any row in the report.

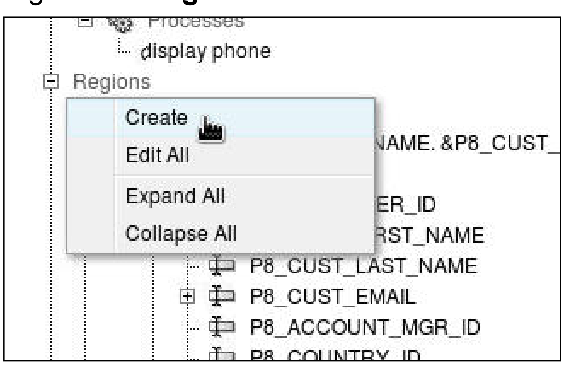


c. On the Customer Details page, create a breadcrumb called Customer Details whose parent entry is Customers (page 3).

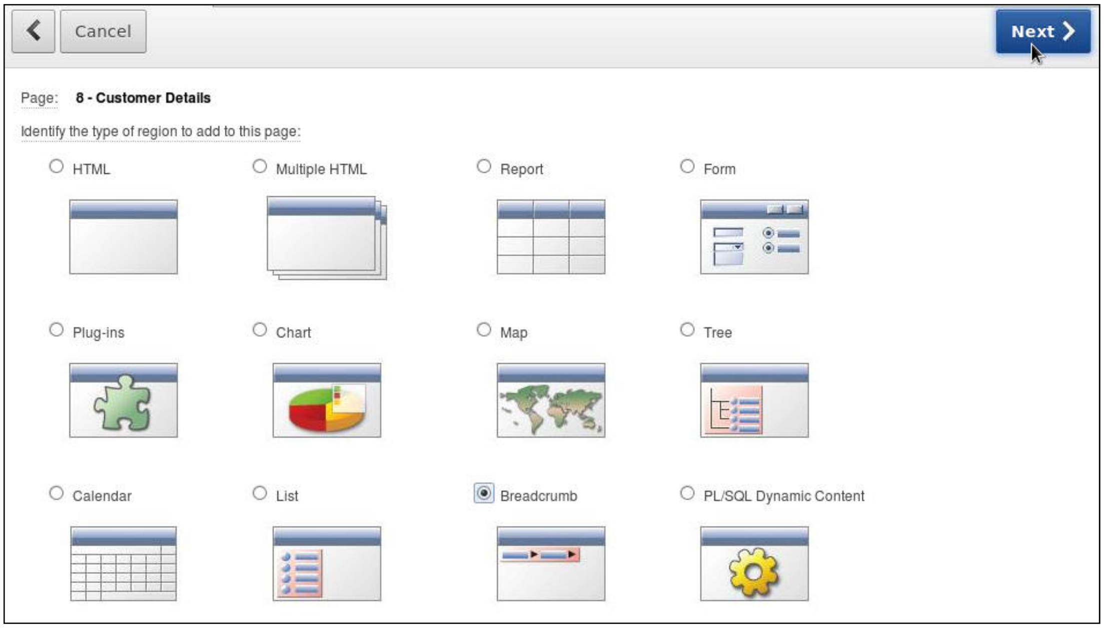
1) The Customer Details page does not have any breadcrumbs. You want to add one that has a parent entry to the Customers report on page 3. Click the **Edit Page** link on the Developer toolbar.



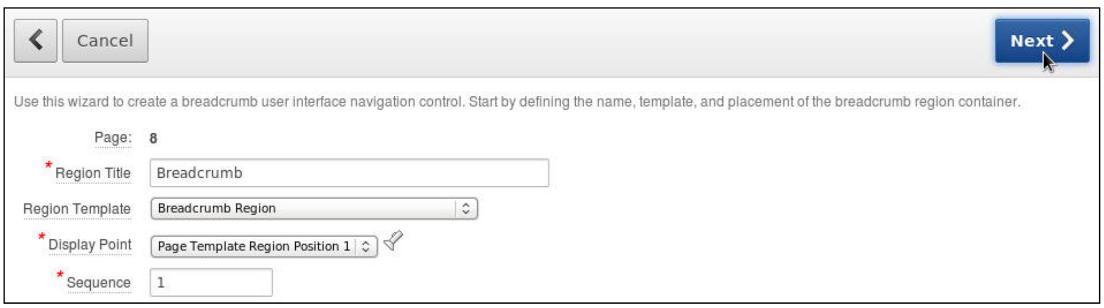
2) Right-click **Regions** and select **Create**.



3) Select **Breadcrumb** for type and click **Next >**.

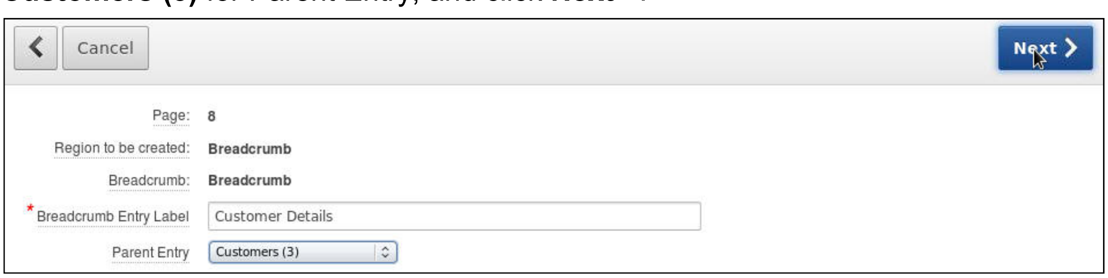


4) Accept the defaults and click **Next >**.



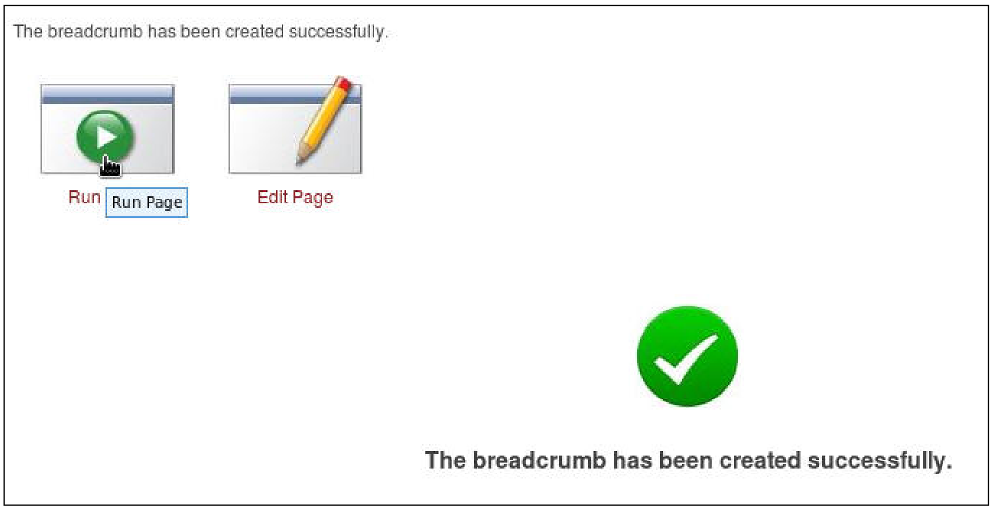
5) Accept the defaults and click **Next >**.

6) Ensure that **Customer Details** is entered for Breadcrumb Entry Label, select **Customers (3)** for Parent Entry, and click **Next >**.

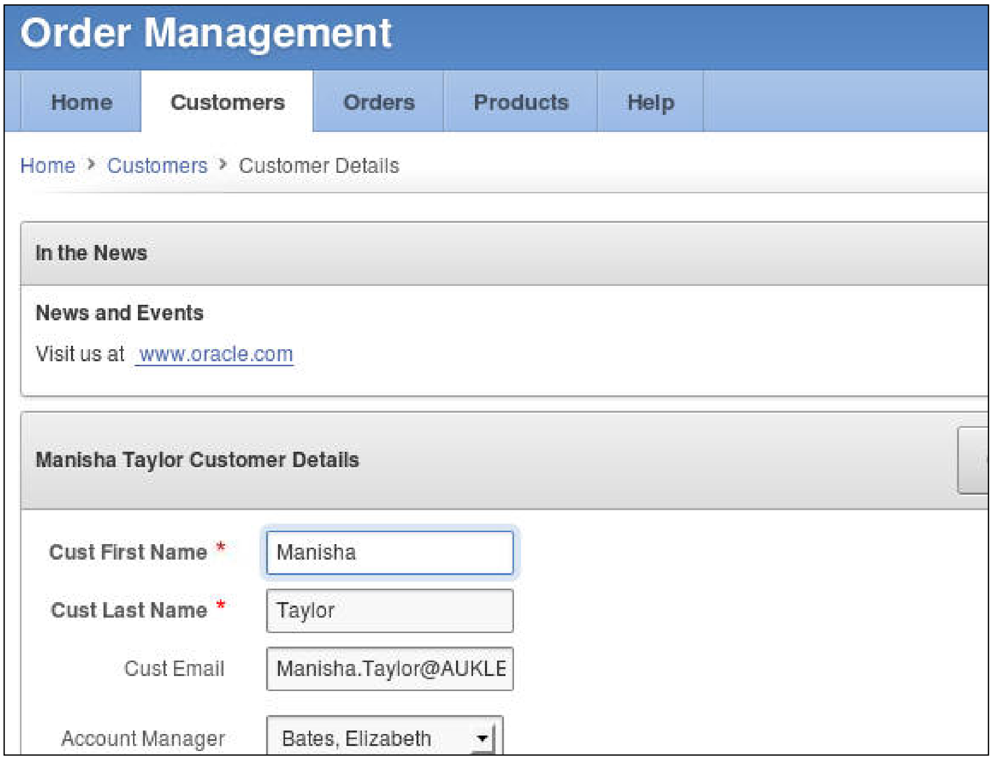


7) Click **Finish**.

8) Click **Run Page**.



9) Notice that the breadcrumb is now displayed. Click the **Application <*n*>** link on the Developer toolbar.



Practices for Lesson 13: Working with Themes, Templates, and Files

Chapter 13

Practices for Lesson 13: Overview

Practices Overview

There is one practice for this lesson. In this practice, you examine how to work with various themes, templates, and files.

Practice 13-1: Working with Themes, Templates, and Files

Overview

In this practice, you switch to a different theme for the Order Management application and upload an image and add it as an application logo. You change an existing region to use a different template. You copy an existing region template, make a change to the custom template, and associate it with a region on your page. Finally, you upload a cascading style sheet, and change the custom template to use the class from the style sheet and also add the class to what is displayed for help text.

Assumptions

You have performed the previous practices.

Tasks

a. Create a new theme (Theme 23) for your application. After creating the theme, switch the theme of your application to that theme.

b. Upload the `logo.png` image from your `/home/oracle/labs/files` directory and add it as a logo for your application.

c. On your Home page (Page 1), change the template for the Tasks to use the Navigation Region template in Region Position 2 and override the List Template to use the Vertical Sidebar List. In addition, change the Page Template so that it has a left sidebar.

d. Copy the “Region without Buttons and Titles” template to the “Region without Buttons and Titles – Custom” template.

e. Add the following line (located in `/home/oracle/labs/files/lab13_1.txt`) right after the last line in the template definition.

```
<div>Oracle Confidential - Internal Only</div>
```

f. Change the template used for the Customer Address List Region on the Customer Address List page so that it uses the custom template that you just modified.

g. Upload the `apexstyle.css` file from the `/home/oracle/labs/files` directory.

h. Add the cascading style sheet to the One Level Tabs – No Sidebar Page Template on the Customer Address List page. The line of code is as follows (located in `/home/oracle/labs/files/lab13_2.txt`):

```
#WORKSPACE_IMAGES#apexstyle.css
```

i. Change the custom region template to use the new style, bigblue, for the text you added at the bottom of the template so that it looks like the following (located in `/home/oracle/labs/files/lab13_3.txt`):

```
<div class=”bigblue”>Oracle Confidential - Internal Only</div>
```

j. Change the Page Template on the Help page and add the <span> tag to the help text (located in `/home/oracle/labs/files/lab13_4.txt`) on the Customers page.

```
<span class=”bigblue”>This page provides information about
Customers.</span>
```

Solution 13-1: Working with Themes, Templates, and Files

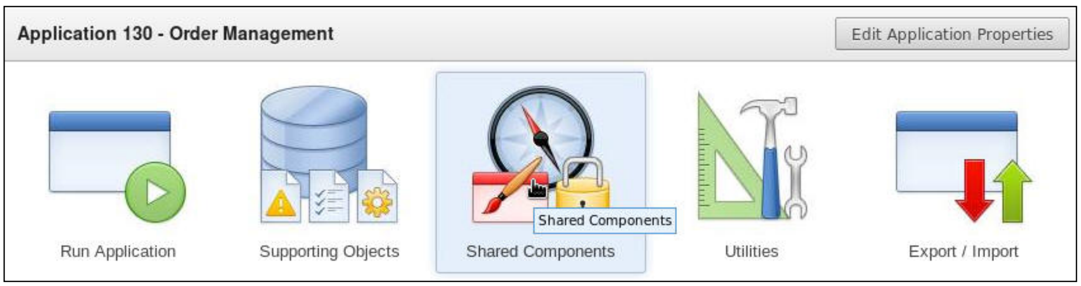
Overview

In this practice, you switch to a different theme for the Order Management application and upload an image and add it as an application logo. You change an existing region to use a different template. You copy an existing region template, make a change to the custom template, and associate it with a region on your page. Finally, you upload a cascading style sheet, and change the custom template to use the class from the style sheet and also add the class to what is displayed for help text.

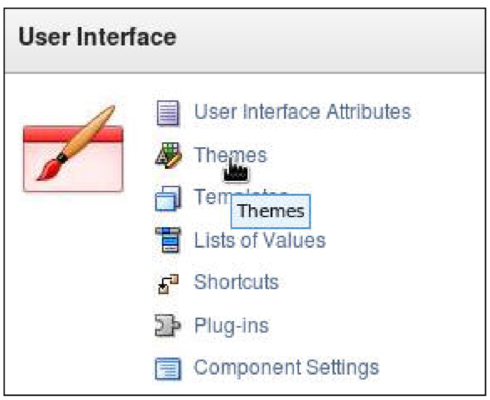
Tasks

a. Create a new theme (Theme 23) for your application. After creating the theme, switch the theme of your application to that theme.

1) Navigate to the **Order Management** application home page and click **Shared Components**.

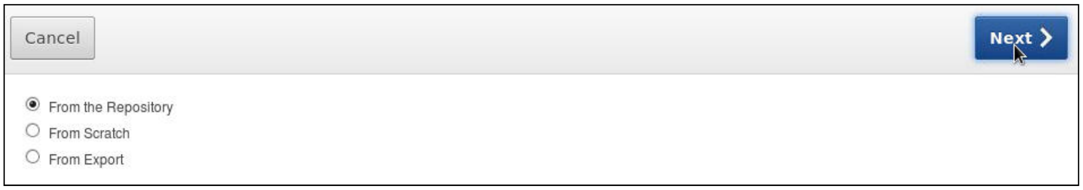


2) In the User Interface region, click **Themes**.



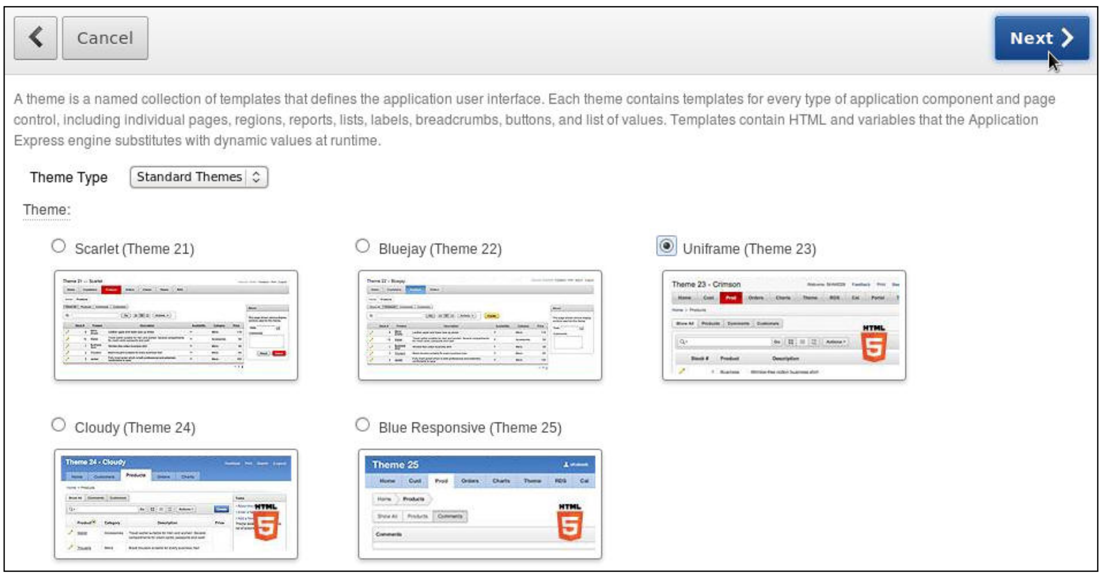
3) Click **Create >**.

4) Ensure that **From the Repository** is selected and click **Next >**.



5) Select User Interface as **Desktop** and click **Next >**.

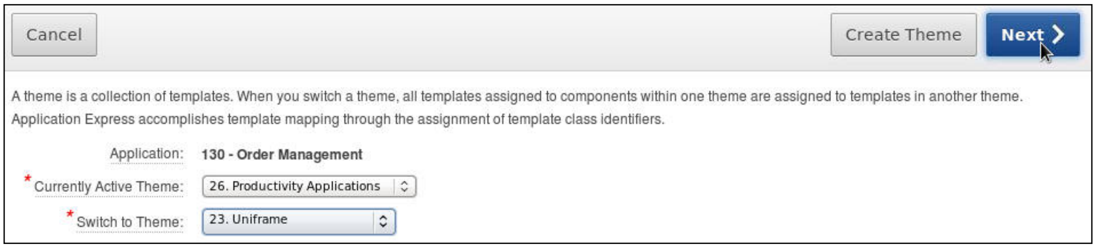
6) Select **Uniframe (Theme 23)** and click **Next >**.



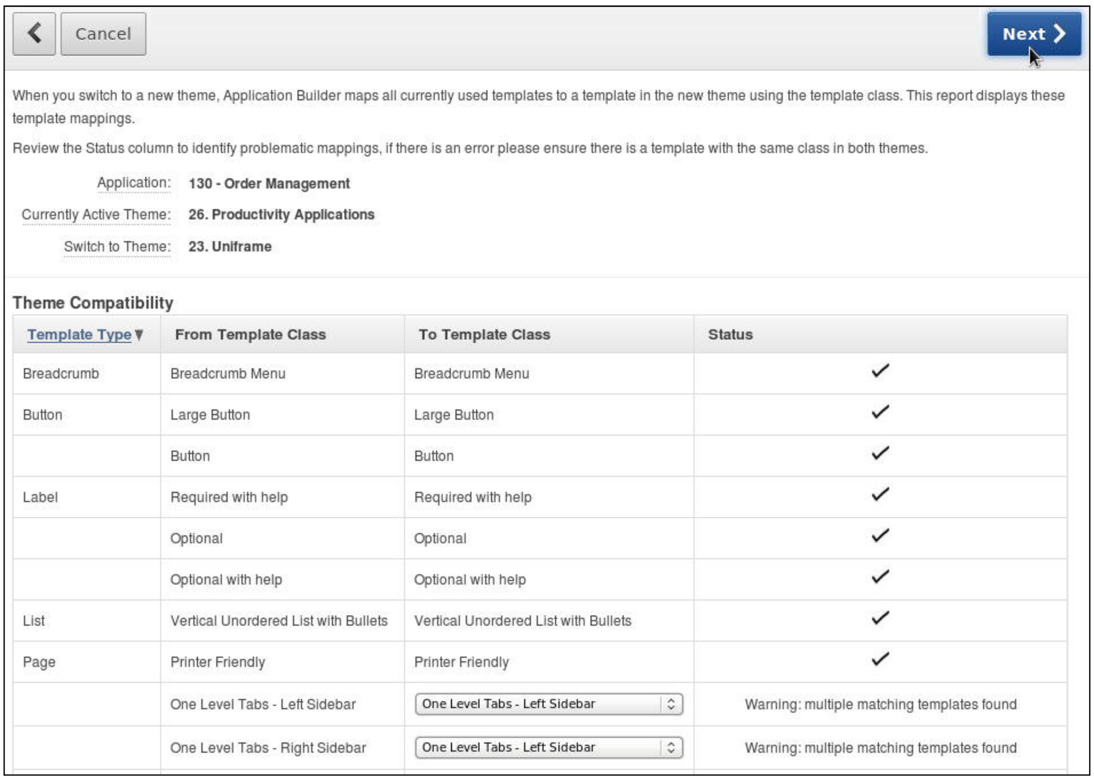
7) Click **Create**.

8) To use the theme that you just created in your application, you must switch to that theme. Click **Switch Theme**.

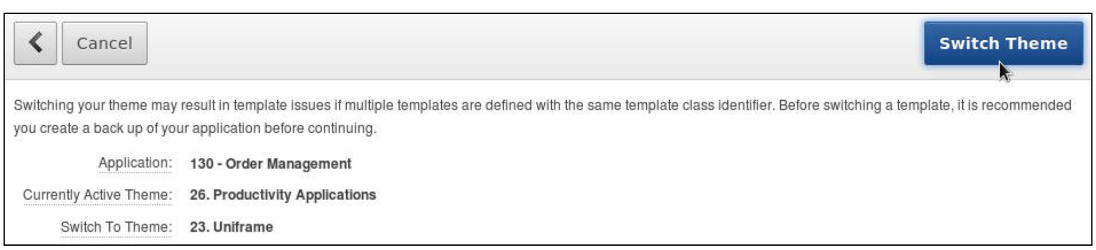
9) Select Currently Active Theme from the drop-down list, in this case, **26. Productivity Applications**. Make sure **23. Uniframe** is selected for Switch to Theme and click **Next >**.



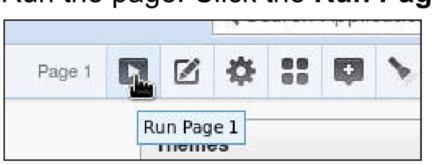
10) Note that the page shows the compatibility between the old and new templates. Click **Next >**.



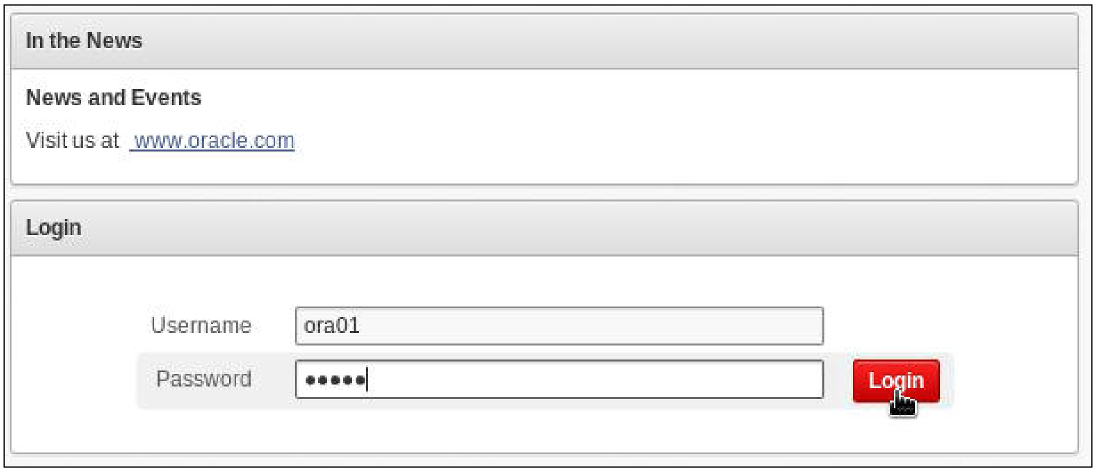
11) Click **Switch Theme**.



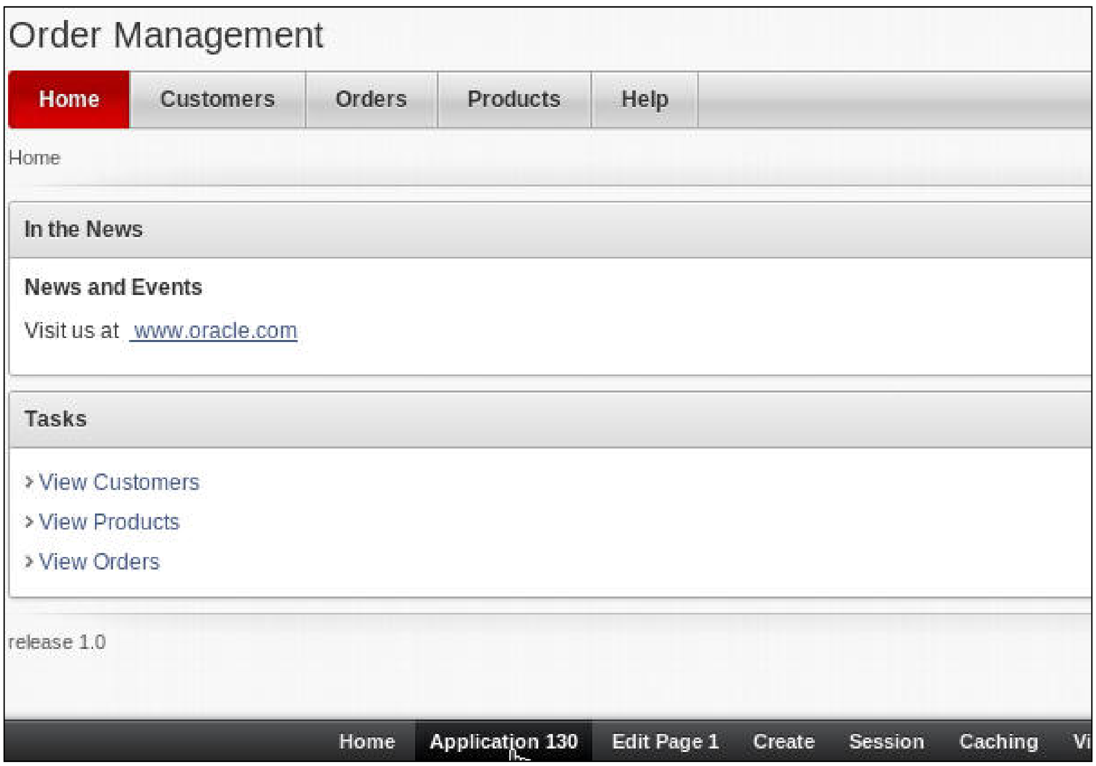
12) Run the page. Click the **Run Page** icon.



13) Note that the Login window uses a different theme. Enter your login information and click **Login**.

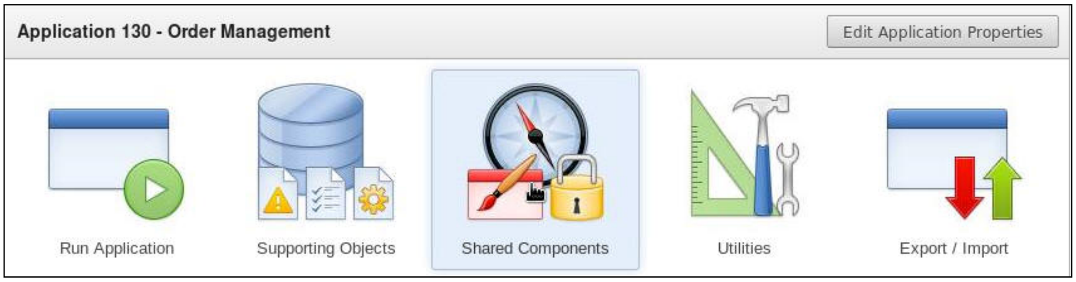


14) Notice that the new theme is used. You want to add a logo to the application. Click the **Application <*n*>** link on the Developer toolbar.



b. Upload the `logo.png` image from your `/home/oracle/labs/files` directory and add it as a logo for your application.

1) You must upload the image to the image repository. Click **Shared Components**.



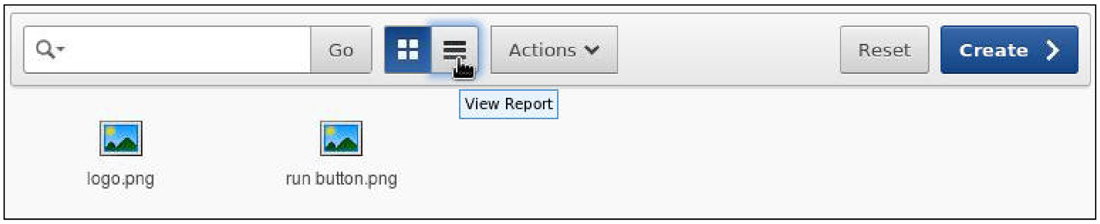
2) Under Files, click **Images**.

3) Click **Create >**.

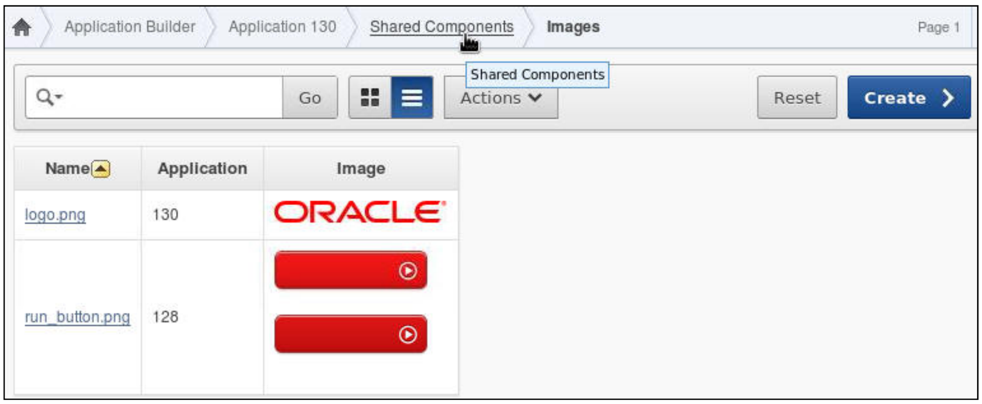
4) Make sure your current application is selected for Application and click **Browse**.

5) Select the `logo.png` file in your `/home/oracle/labs/files` folder. Click **Upload**.

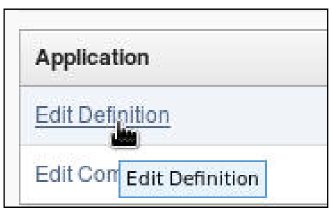
6) You can see what the images look like in the Review view. Click the **View Report** icon.



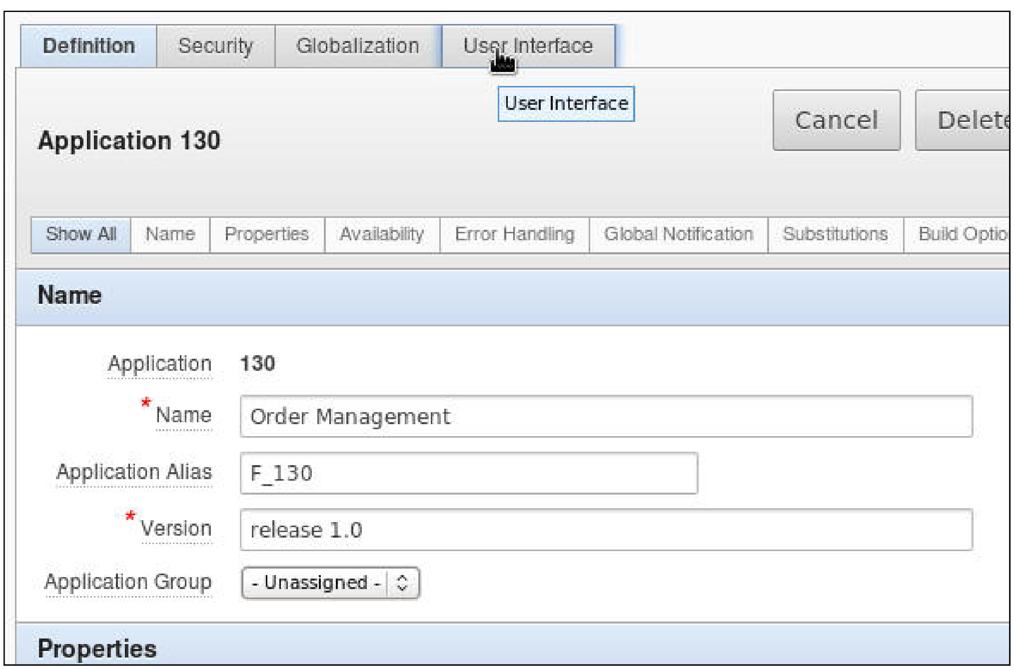
7) You can add the logo to your application. Click the **Shared Components** breadcrumb.



8) Click **Edit Definition** in the Application section.

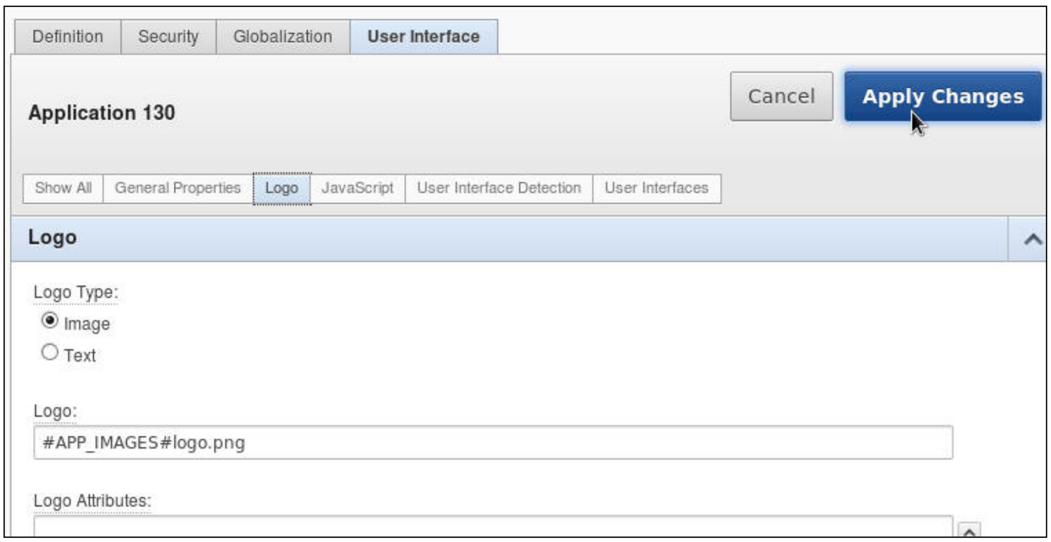


9) In the list of links displayed at the top of the page, click **User Interface**.

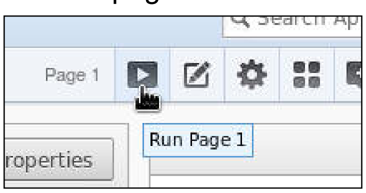


10) Click the **Logo** link.

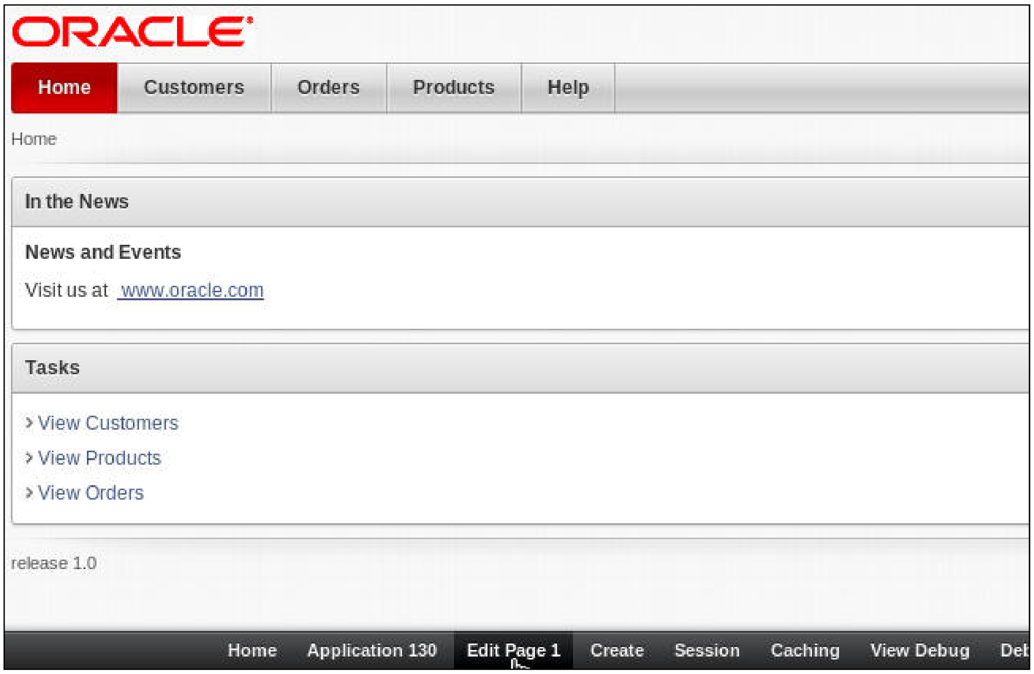
11) Click the Image option. Enter `#APP_IMAGES#logo.png` in the Logo field. Click **Apply Changes**.



12) Run the page. Click the **Run Page** icon.

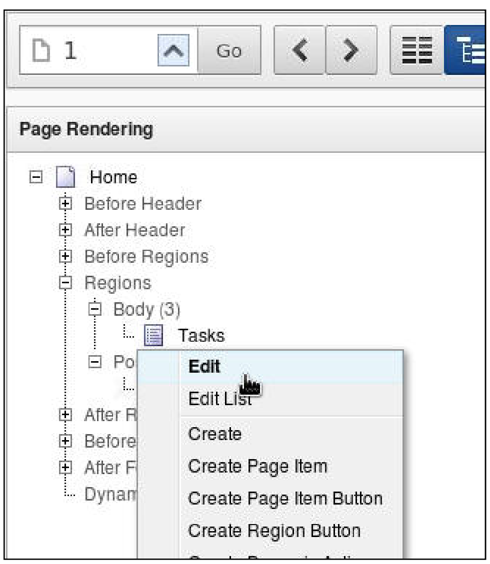


13) Note that all your pages display the logo. You want to change the templates used on this page. Click the **Edit Page** link on the Developer toolbar.

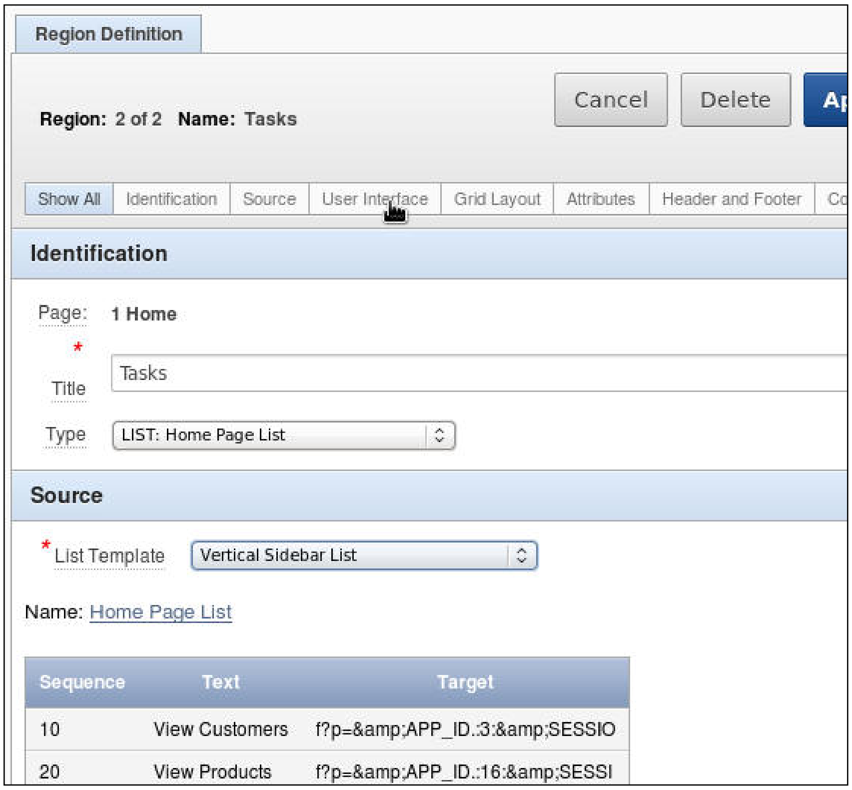


c. On your Home page (Page 1), change the template for the Tasks to use the Navigation Region template in Region Position 2 and override the List Template to use the Vertical Sidebar List. In addition, change the Page Template so that it has a left sidebar.

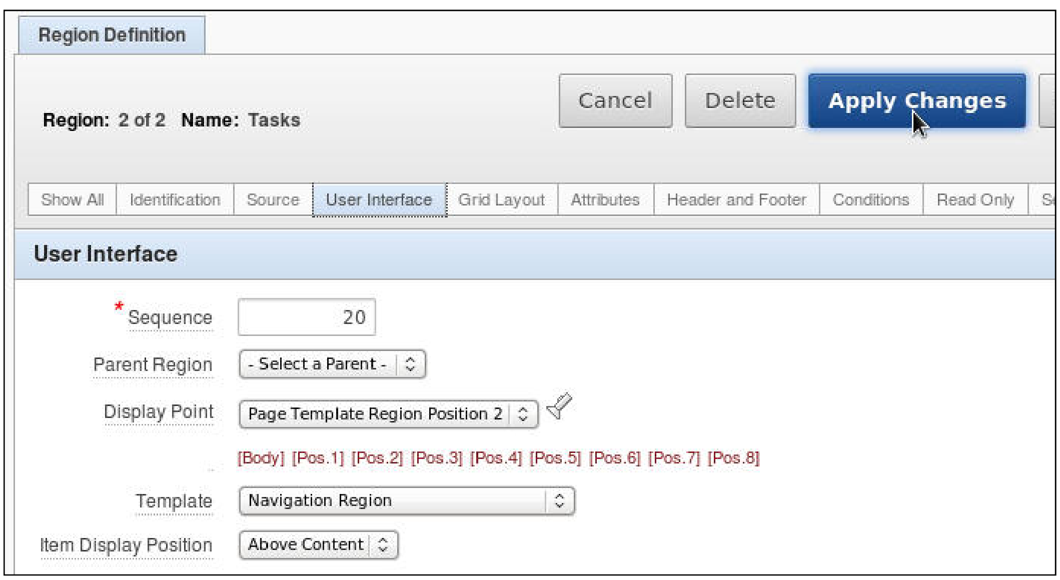
1) You want to modify the templates used in the Tasks Region. Right-click **Tasks** and select **Edit**.



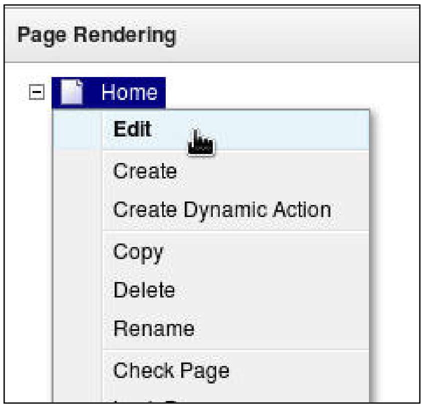
2) Under Source, select **Vertical Sidebar List** for List Template and click the **User Interface** subtab.



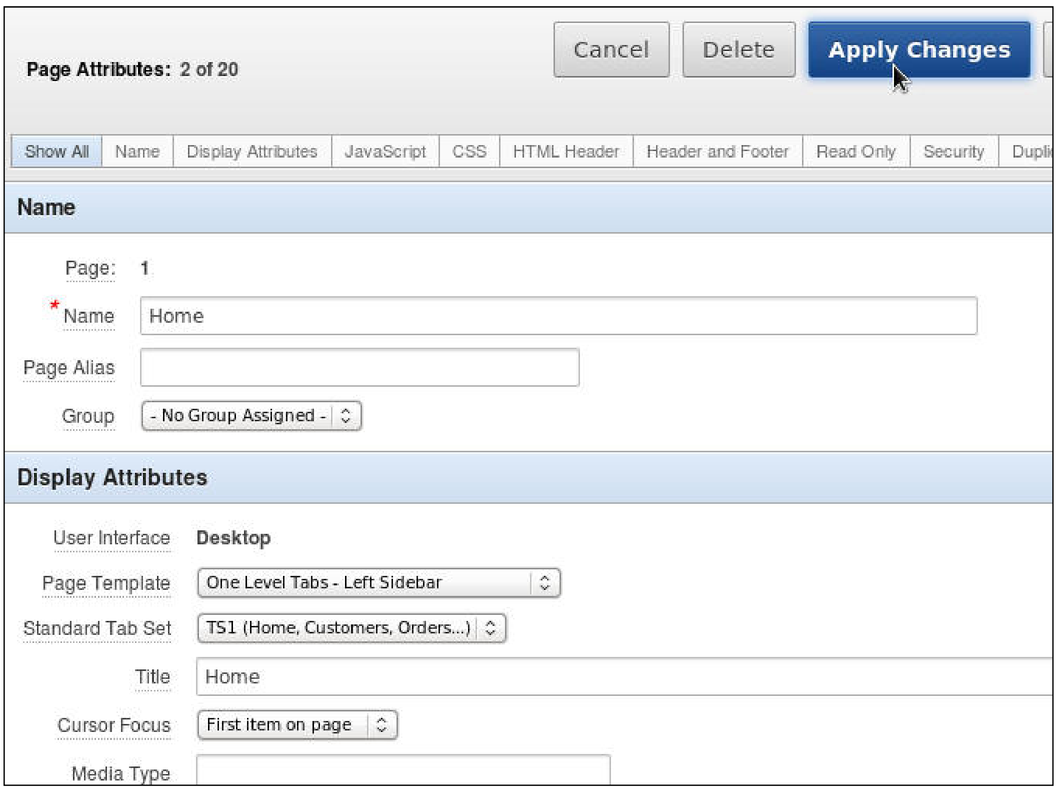
3) Select **Page Template Region Position 2** for Display Point and select **Navigation Region** for Template. Then click **Apply Changes**.



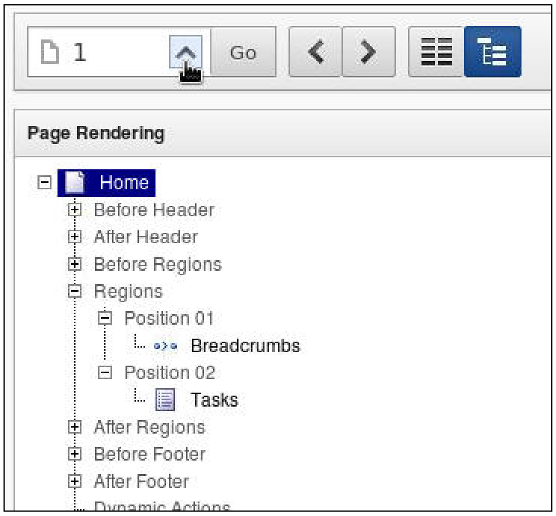
4) The default page template does not have a sidebar. To add a sidebar, you need to change the page template. Right-click **Home** and select **Edit**.



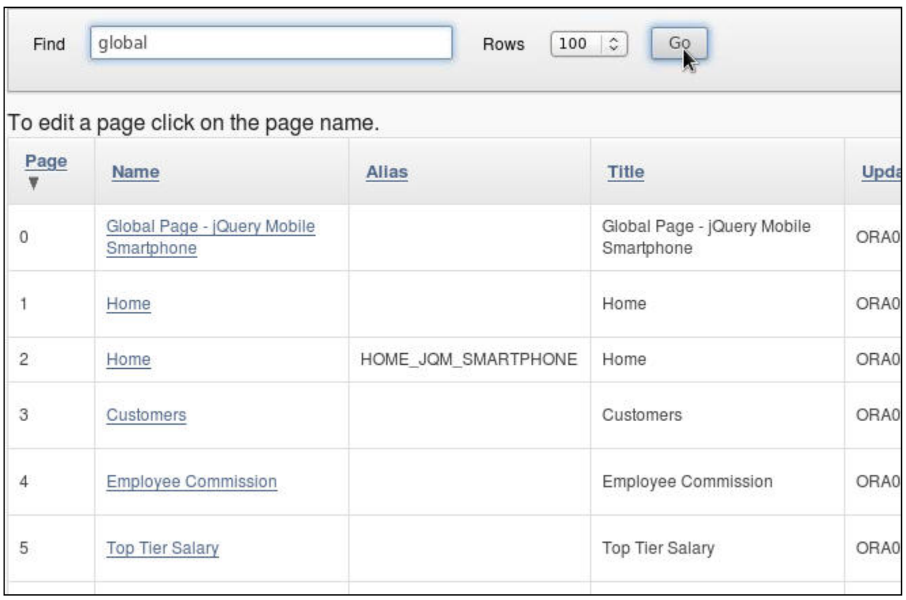
5) Change the Page Template to **One Level Tabs – Left Sidebar** and click **Apply Changes**.



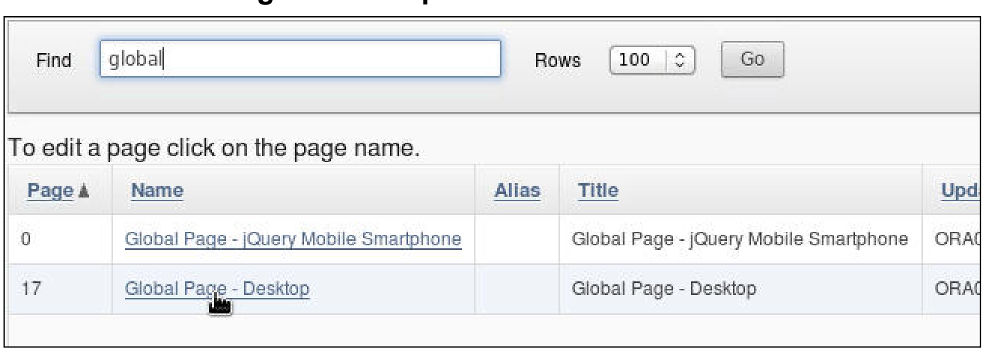
6) Before you run the page, you want to change the place where the "In the News Region" is displayed on the Global Page. Click the up arrow to find the Global Page for the desktop.



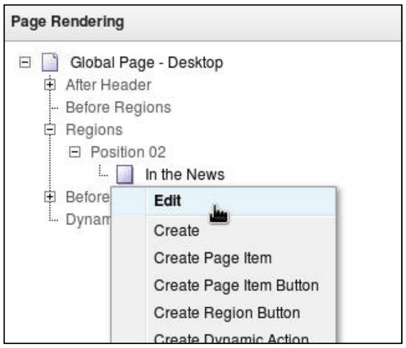
7) Enter **global** in the Find area and click **Go**.



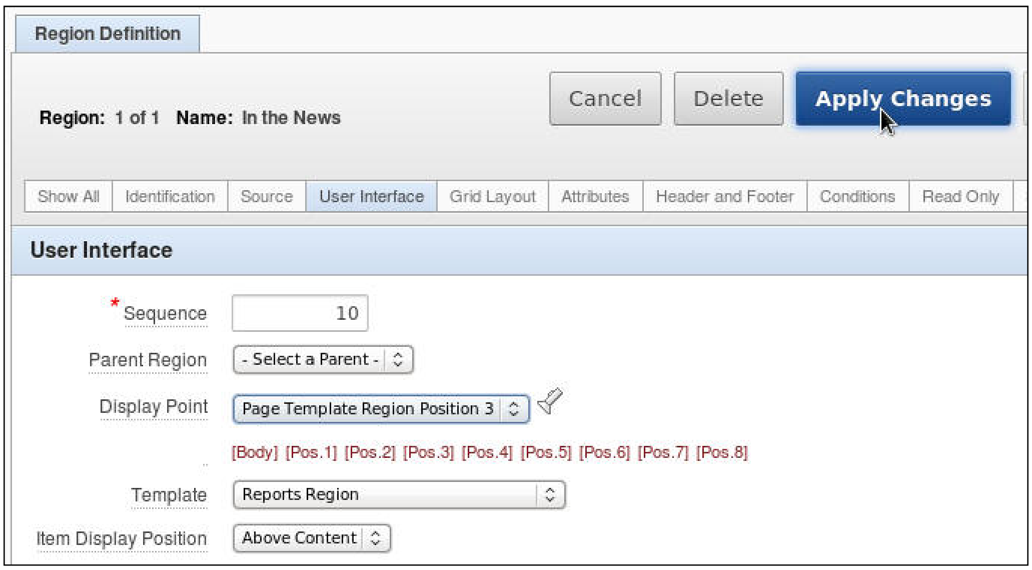
8) Click the **Global Page – Desktop** link.



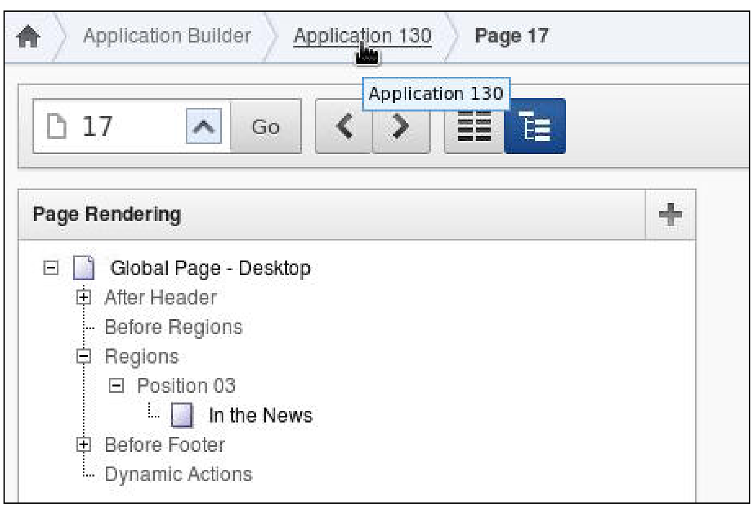
9) Note that the “In the News” region is also in Position 2. You want to change the display point. Right-click **In the News** and select **Edit**.



10) Select **Page Template Region Position 3** for Display Point and click **Apply Changes**.



11) Click the **Application <*n*>** breadcrumb.



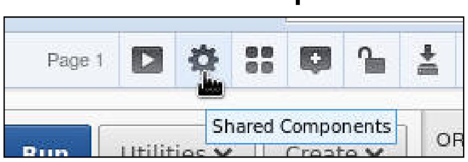
12) Click **Run Application**.

13) The new template is set for the list. If you place your cursor over one of the options, notice that the vertical bar next to the option turns red, indicating the active region.

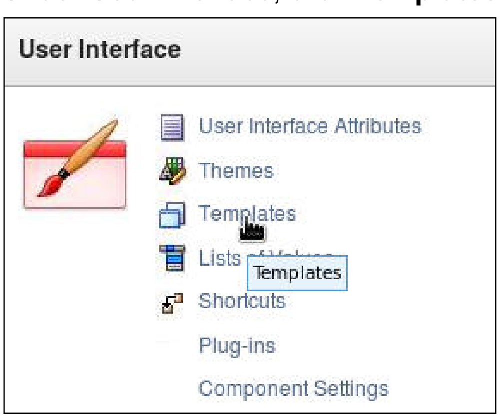
14) You want to make a change to one of the templates. Click the **Edit Page** link on the Developer toolbar.

d. Copy the "Region without Buttons and Titles" template to the "Region without Buttons and Titles – Custom" template.

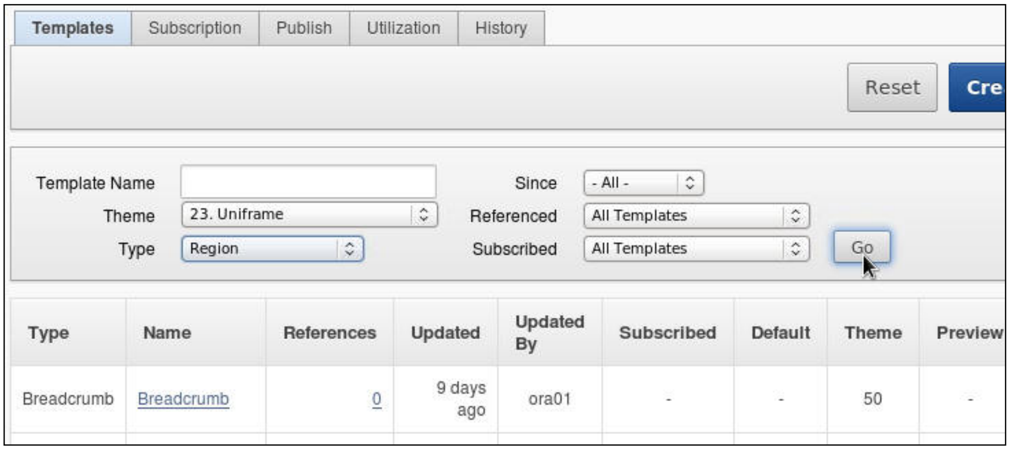
1) Click the **Shared Components** icon.



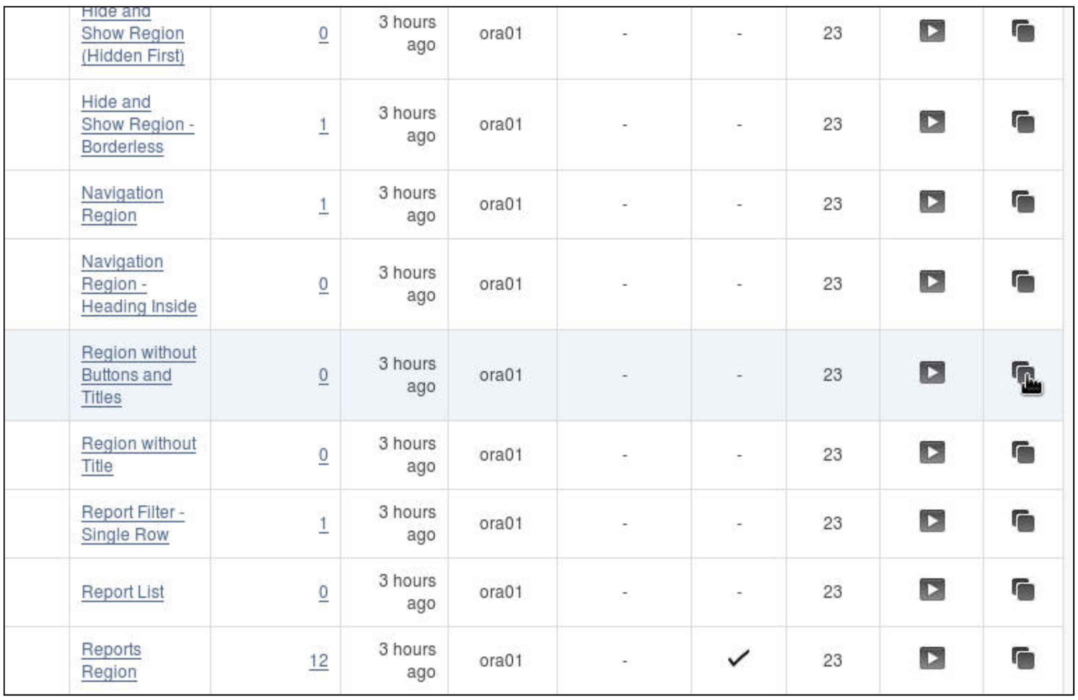
2) Under User Interface, click **Templates**.



3) Select **23. Uniframe** for Theme and **Region** for Type and click **Go**.



4) You want to make a copy of the **Region without Buttons and Titles** template. Click the **Copy** icon for this template.



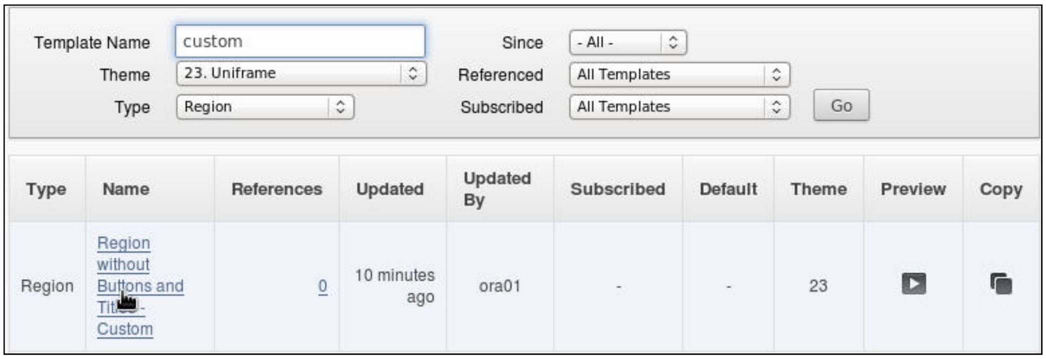
5) Enter **Region without Buttons and Titles – Custom** for New Template Name, and click **Copy**.

6) Leave the Theme and Type selection as is and enter **Custom** for Template Name and click **Go**.

e. Add the following line (located in `/home/oracle/labs/files/lab13_1.txt`) right after the last line in the template definition:

```
<div>Oracle Confidential – Internal Only</div>
```

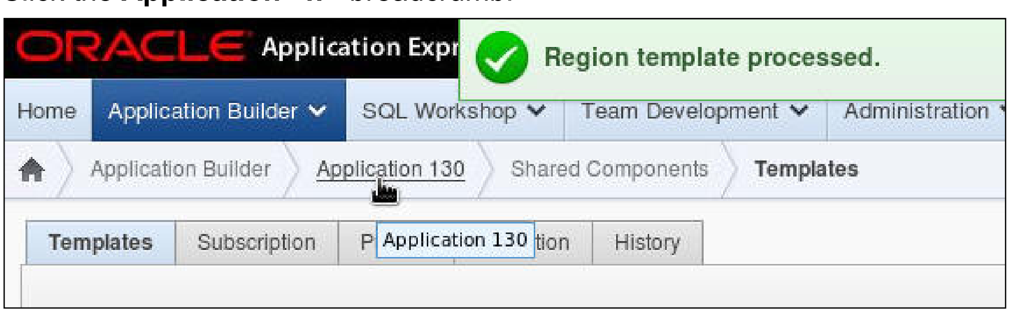
1) Click the **Region without Buttons and Titles – Custom** link.



2) Add the following line (located in `/home/oracle/labs/files/lab13_1.txt`) to the template definition right after the last line </div> and click **Apply Changes**:

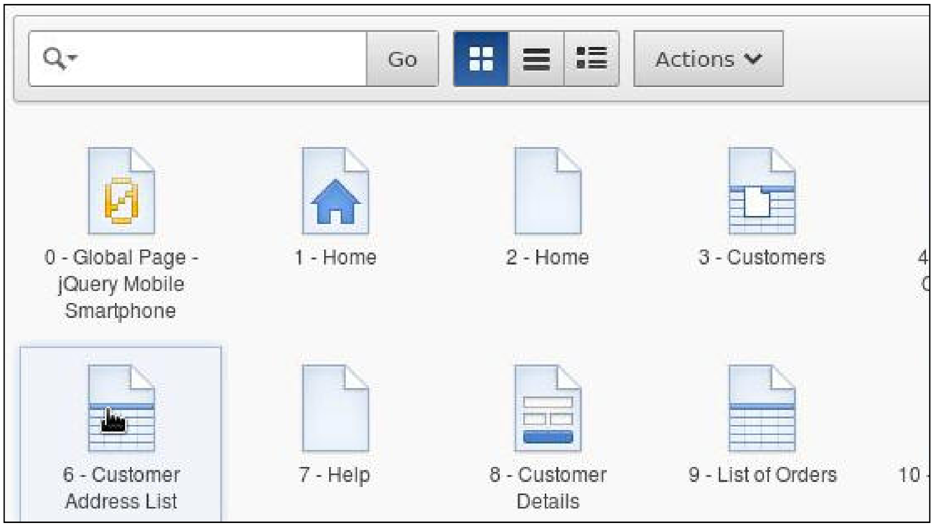
```
<div>Oracle Confidential – Internal Only</div>
```

3) Click the **Application <*n*>** breadcrumb.

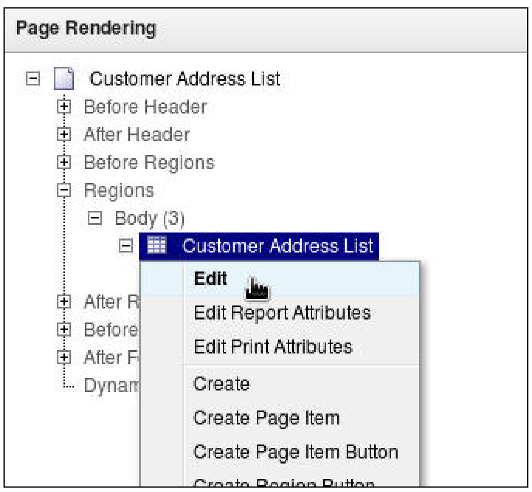


f. Change the template used for the Customer Address List Region on the Customer Address List page so that it uses the custom template that you just modified.

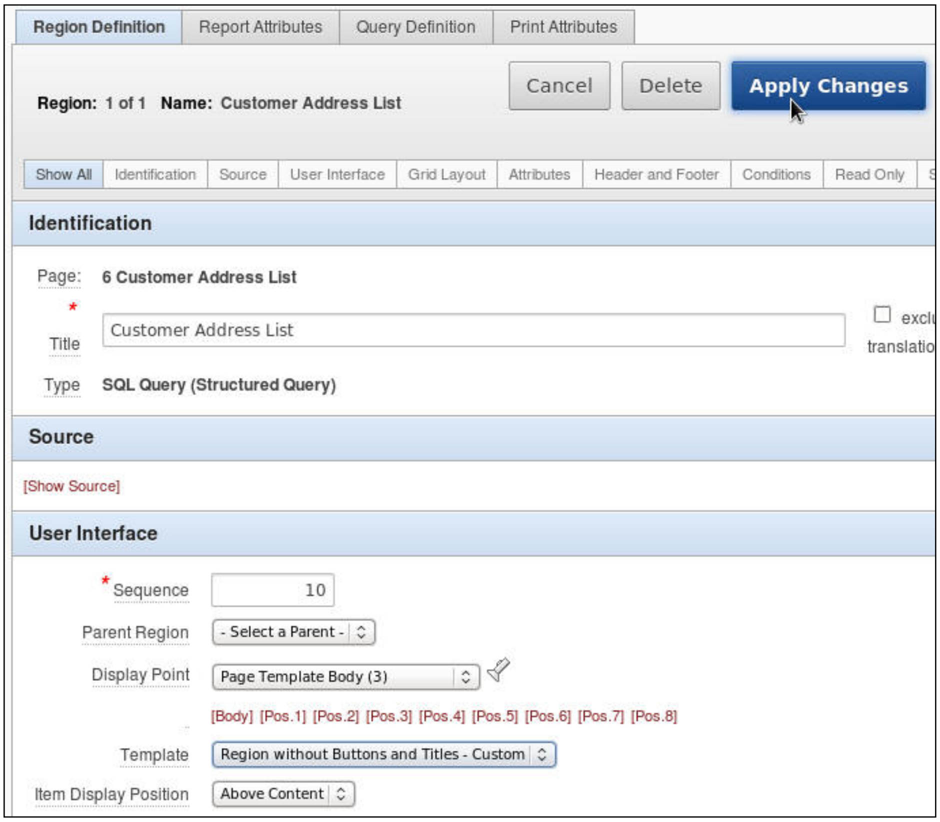
1) You must change the template used for a region to the template that you just modified. Navigate to the Order Management application and click the **Customer Address List** page.



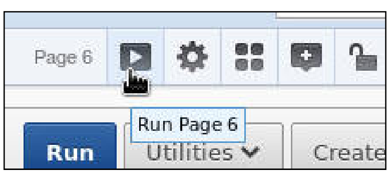
2) Under regions, right-click the **Customer Address List** region and select **Edit**.



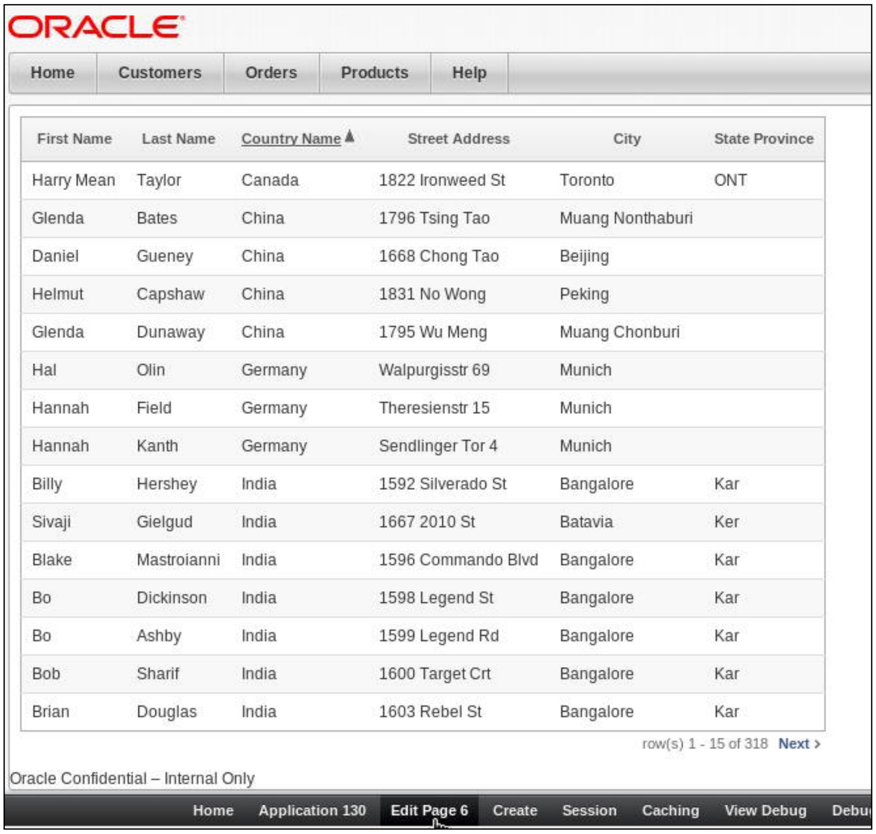
3) Under User Interface, select **Region without Buttons and Titles – Custom** from the list of templates and click **Apply Changes**.



4) Click the **Run Page** icon.

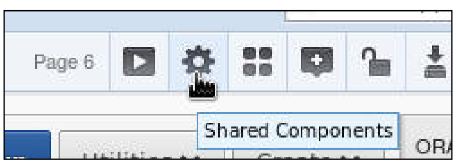


5) Notice that your report contains the Oracle Confidential message at the bottom of the page. You want to change the style of this text to use a class in a custom style sheet. Click the **Edit Page** link on the Developer toolbar.

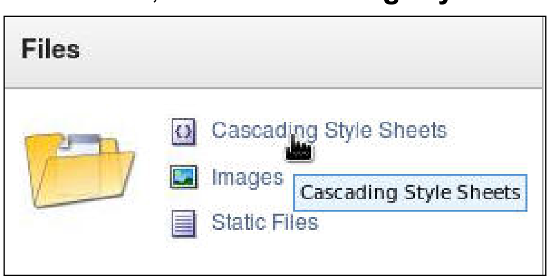


g. Upload the `apexstyle.css` file from the `/home/oracle/labs/files` directory.

1) Click the **Shared Components** icon.

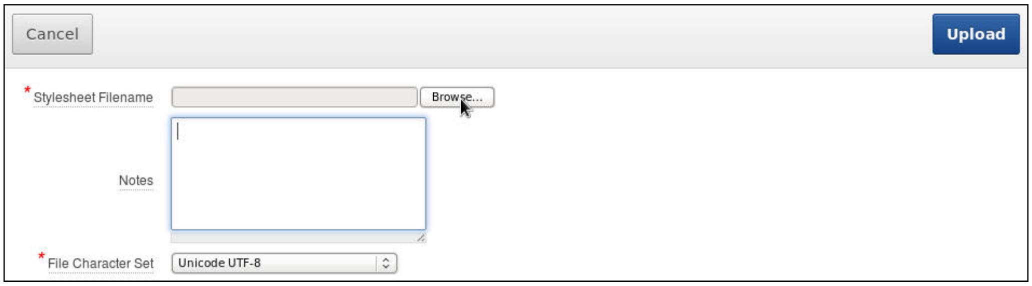


2) Under Files, select **Cascading Style Sheets**.

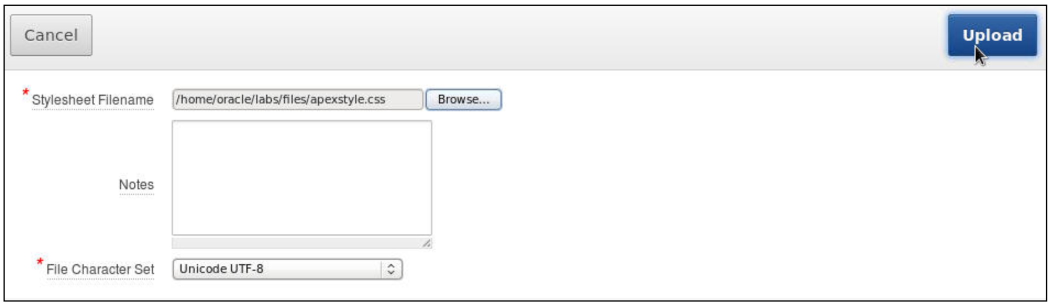


3) Click **Create >**.

4) Click **Browse**.

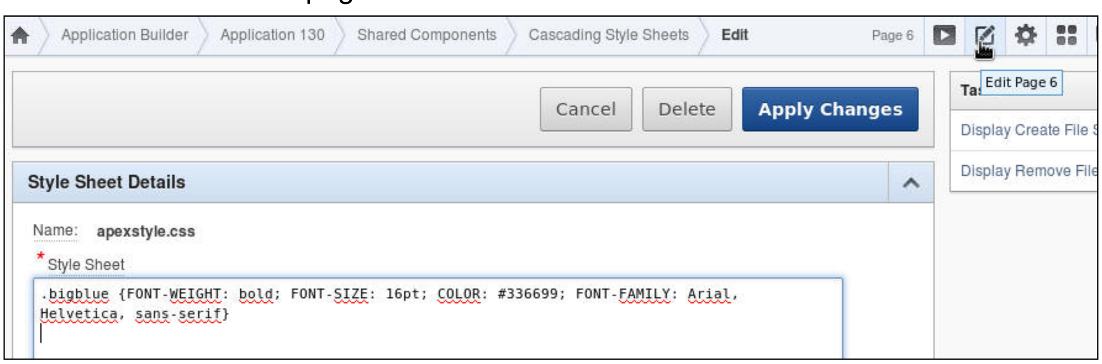


5) Select the **`apexstyle.css`** file from the **`/home/oracle/labs/files`** directory and click **Open**. Click **Upload**.



6) To view the style sheet, select **`apexstyle.css`**.

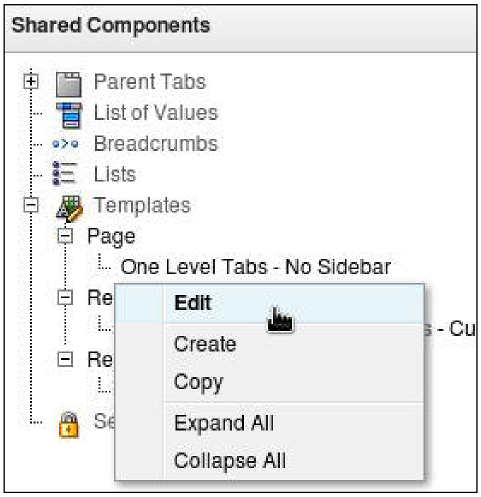
7) Notice that the name of the style is bigblue. Click the **Edit Page** icon for the Customer Address List page.



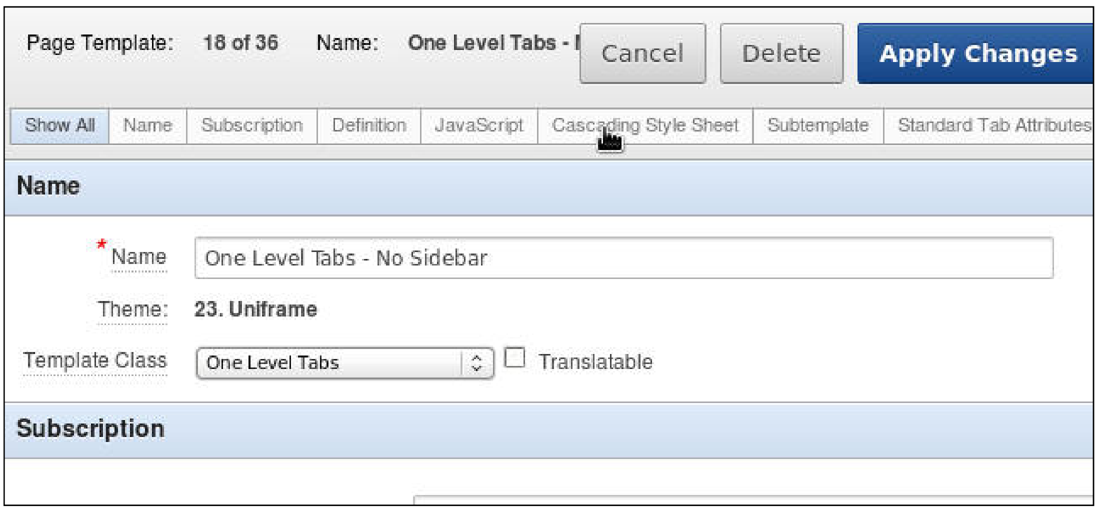
h. Add the cascading style sheet to the One Level Tabs – No Sidebar Page Template on the Customer Address List page. The line of code is as follows (located in `/home/oracle/labs/files/lab13_2.txt`):

```
#WORKSPACE_IMAGES#apexstyle.css
```

1) You first need to add the style sheet to the Page template. Under Shared Components, expand **Templates**. Right-click **One Level Tabs – No Sidebar** under Page and select **Edit**.

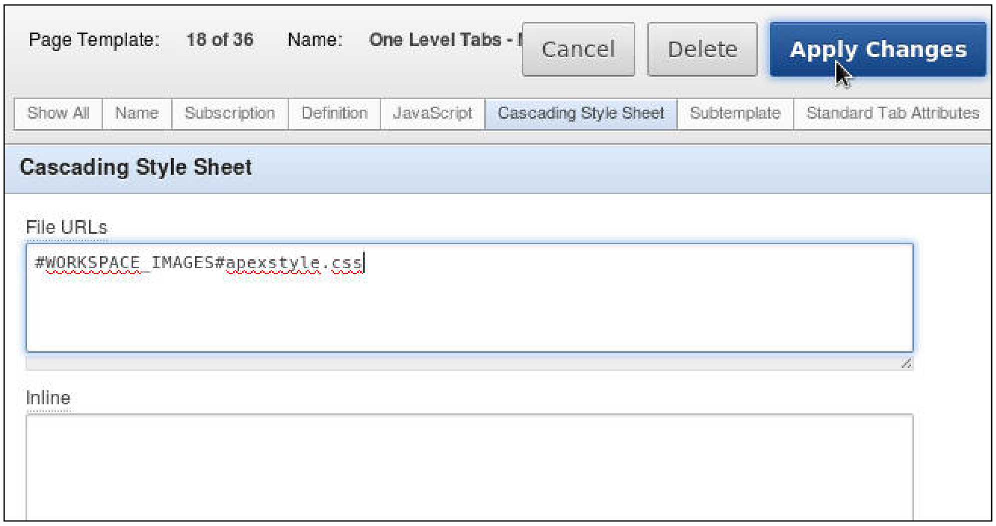


2) Click the **Cascading Style Sheet** subtab.



3) Enter the following line of code (located in `/home/oracle/labs/files/lab13_2.txt`) in the File URLs area. Then click **Apply Changes**.

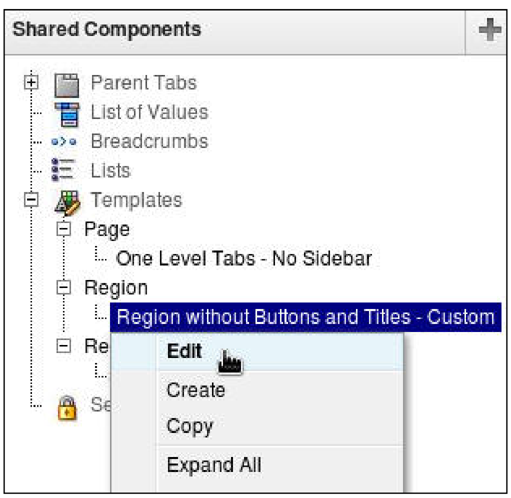
```
#WORKSPACE_IMAGES#apexstyle.css
```



i. Change the custom region template to use the new style, bigblue, for the text you added at the bottom of the template so that it looks like the following (located in `/home/oracle/labs/files/lab13_3.txt`):

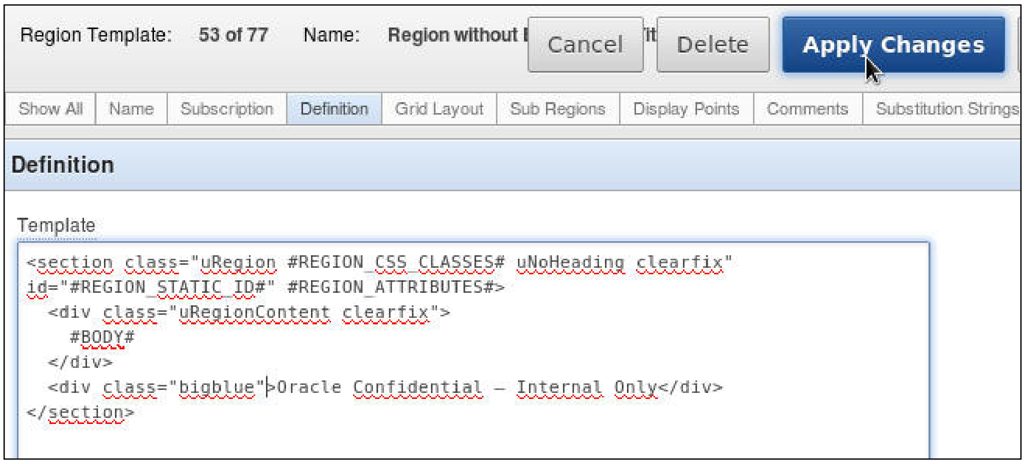
```
<div class="bigblue">Oracle Confidential – Internal Only</div>
```

1) Under Templates > Region, right-click **Region without Buttons and Titles – Custom** and select **Edit**.

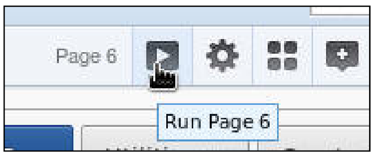


2) Add **`class="bigblue"`** to the line that you added previously (located in `/home/oracle/labs/files/lab13_3.txt`) so that it reads as follows (changes in bold). Then click **Apply Changes**.

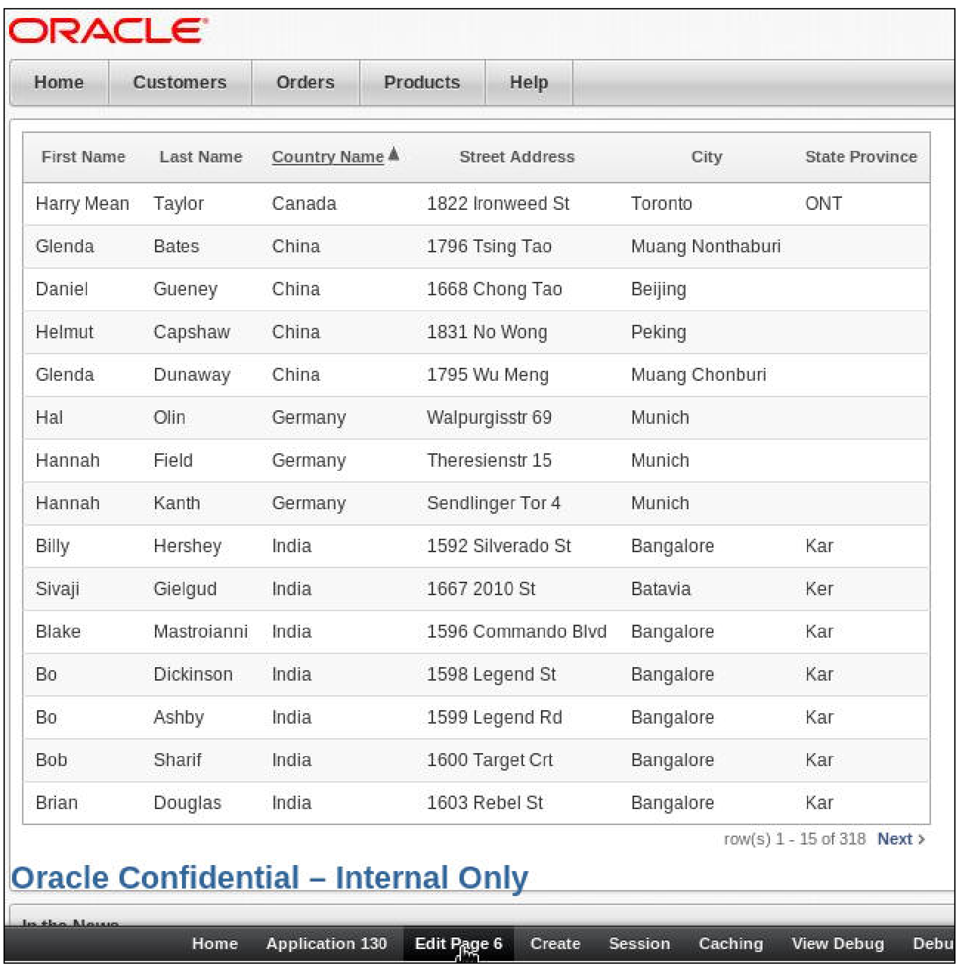
```
<div class="bigblue">Oracle Confidential – Internal
Only</div>
```



3) Click the **Run Page** icon.



4) Notice that the bigblue style was applied. You can also add the style to a section on your page. Click the **Edit Page** link on the Developer toolbar.



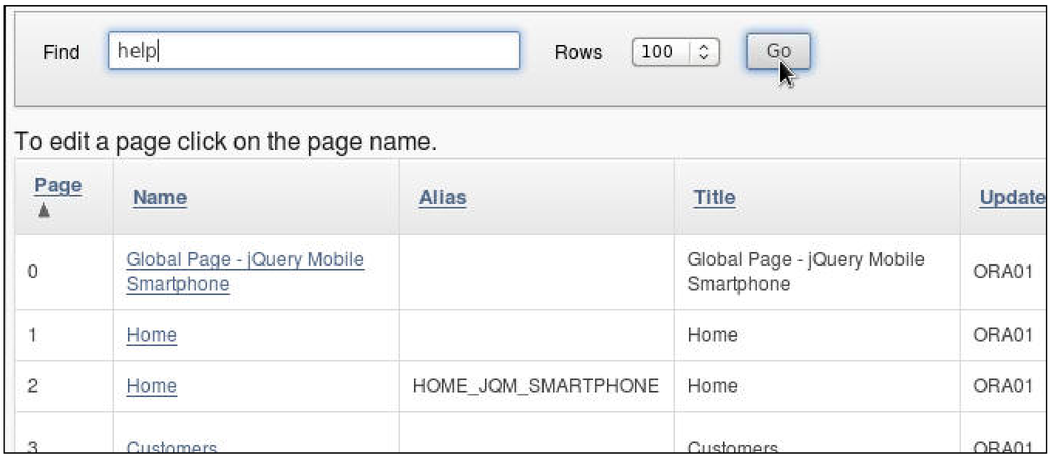
First Name	Last Name	Country Name	Street Address	City	State Province
Harry Mean	Taylor	Canada	1822 Ironweed St	Toronto	ONT
Glenda	Bates	China	1796 Tsing Tao	Muang Nonthaburi	
Daniel	Gueney	China	1668 Chong Tao	Beijing	
Helmut	Capshaw	China	1831 No Wong	Peking	
Glenda	Dunaway	China	1795 Wu Meng	Muang Chonburi	
Hal	Olin	Germany	Walpurgisstr 69	Munich	
Hannah	Field	Germany	Theresienstr 15	Munich	
Hannah	Kanth	Germany	Sendlinger Tor 4	Munich	
Billy	Hershey	India	1592 Silverado St	Bangalore	Kar
Sivaji	Gielgud	India	1667 2010 St	Batavia	Ker
Blake	Mastroianni	India	1596 Commando Blvd	Bangalore	Kar
Bo	Dickinson	India	1598 Legend St	Bangalore	Kar
Bo	Ashby	India	1599 Legend Rd	Bangalore	Kar
Bob	Sharif	India	1600 Target Crt	Bangalore	Kar
Brian	Douglas	India	1603 Rebel St	Bangalore	Kar

j. Change the Page Template on the Help page and add the <span> tag to the help text (located in `/home/oracle/labs/files/lab13_4.txt`) on the Customers page.

```
<span class="bigblue">This page provides information about
Customers.</span>
```

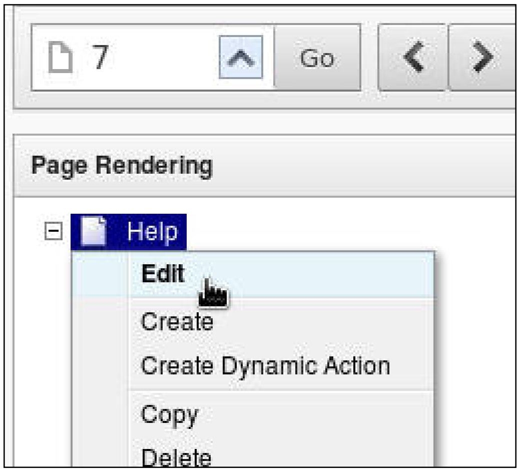
1) Click the up arrow to find a page.

2) Enter **help** in the Find area and click **Go**.

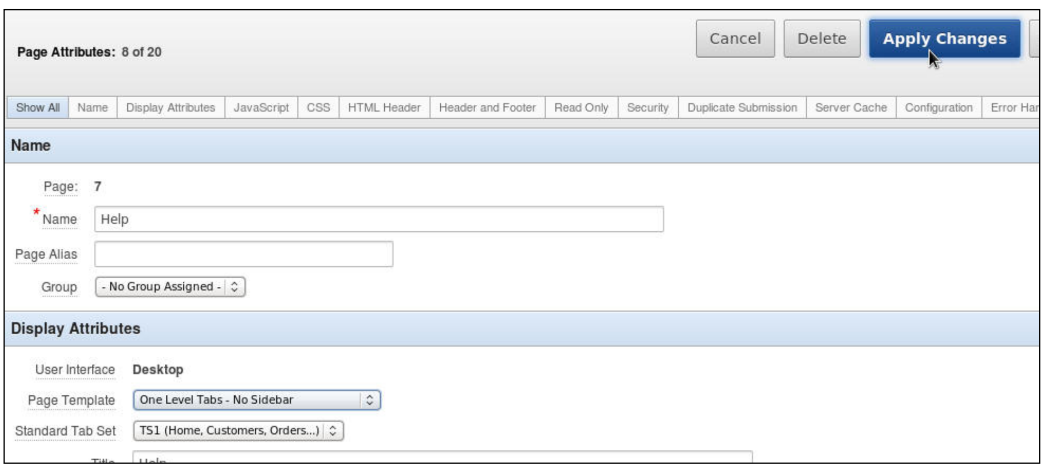


3) Click the **Help** link to open the page definition.

4) Change the Page template on the Help page to the same template you added the style sheet to in the previous steps. Right-click the **Help** page definition node and select **Edit**.

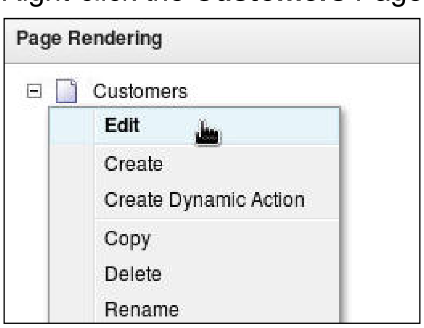


5) Select **One Level Tabs – No Sidebar** for Page Template and click **Apply Changes**.

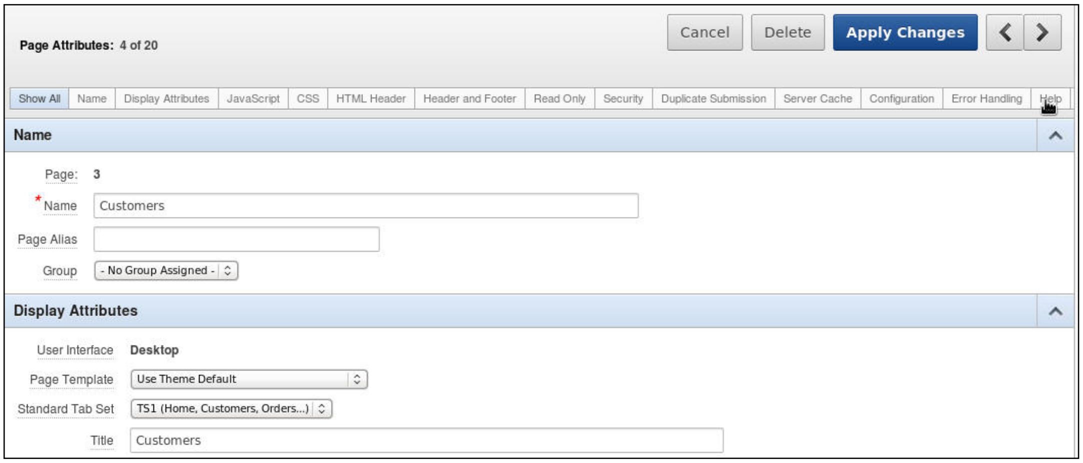


6) Go to the Customers page. Enter **3** in the page area and click **Go**.

7) Right-click the **Customers** Page Rendering node and select **Edit**.



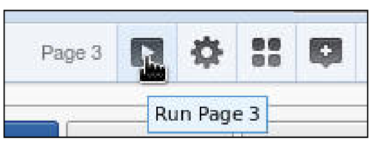
8) Click the **Help** tab.



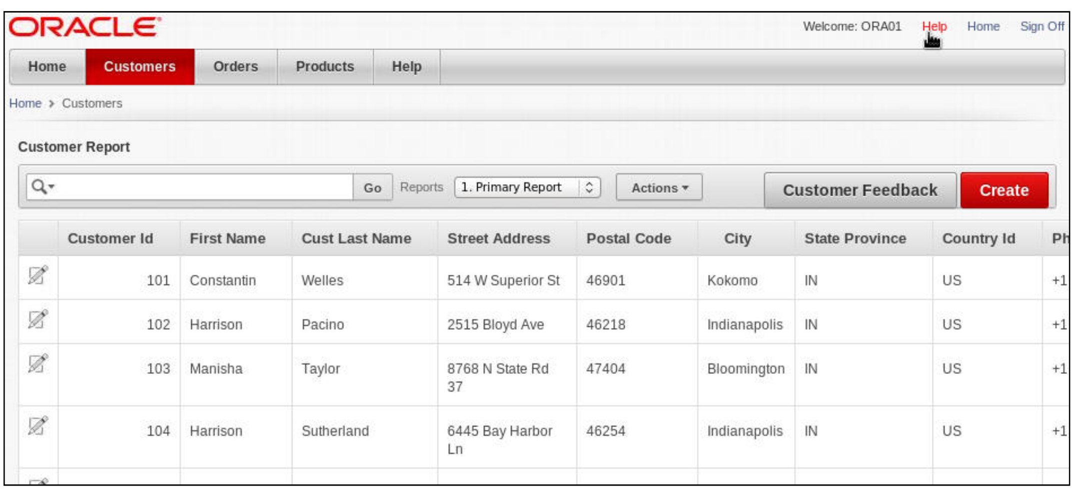
9) Change the help text to the following (located in `/home/oracle/labs/scripts/lab13_4.txt`) and click **Apply Changes**.

```
<span class="bigblue">This page provides information about
Customers.</span>
```

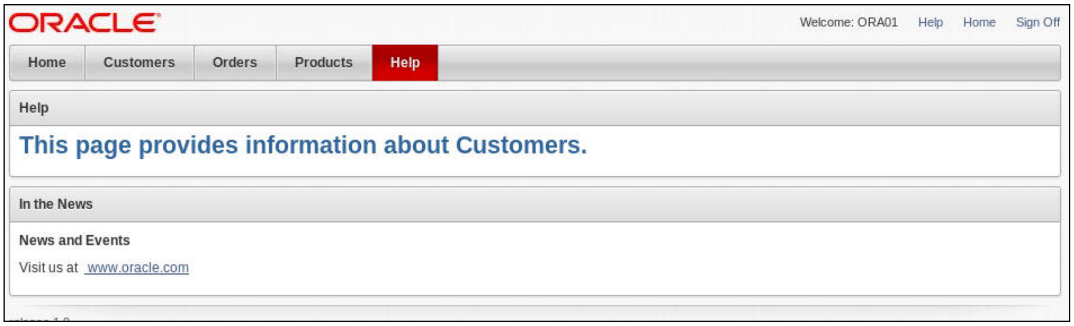
10) Click the **Run Page** icon.



11) Click the **Help** navigation bar entry to show the page-level help.



12) Notice that the bigblue class has been applied to the help text.



Practices for Lesson 14: Implementing Security

Chapter 14

Practices for Lesson 14: Overview

Practices Overview

There are three practices for this lesson. In these practices, you examine how to implement security within your application.

Practice 14-1: Creating an Authentication Scheme

Overview

In this practice, you create an authentication scheme, and then make it the current default.

Assumptions

You have performed the previous practices.

Tasks

a. Create an authentication scheme called OpenDoor based on the Open Door credentials.
b. Switch the current authentication schema to Application Express.

Practice 14-2: Restricting Users by Using Access Control

Overview

In this practice, you create an access control page, set the application mode to restrict access, create the access control list, and assign application components to an authorization scheme.

Assumptions

You have performed the previous practices

Tasks

a. Create the following users to add to the access control list.

Name	User Type
Brad.Knight	Developer
Susie.Parker	Workspace Administrator
John.Bell	End User

b. Create an access control page for the Order Management application.

c. Set the application mode to Restricted access. Only users defined in the access control list are allowed.

d. Add users to the access control list:

- `john.bell` can only view the information in the application; he cannot make any changes to application data.
- `brad.knight` should be allowed to edit application data, but he cannot change application administration settings (application mode and the access control list).
- `susie.parker` is the administrator of the application, so she can change anything in addition to changing the user privileges.
- `ora<n>` is also an administrator of the application.

e. Define and apply the authorization schemes to each application component. This will restrict access to application pages and components.

- Users with the View privilege can review customer information, but cannot change or create information.
- Users with the Edit privilege can make changes to customer information but cannot make changes to the application mode and the access control list.
- Users with the Administrator privilege can make any changes, including administering the Order Management application.

f. Run the application and log in as a different user. What do you observe?

Practice 14-3: Using Session State Protection

Overview

In this practice, you enable Session State Protection, set the Session State Protection for a particular item, and review the various options for page access protection so that you can examine their differences.

Assumptions

You have performed the previous practices.

Tasks

a. Set the item session state protection for the P<n>_ORDER_ID on the Master Detail page to Checksum Required – Session Level. Run the page to see the error message.
b. Set the Page Access Protection for the Master Detail page to Arguments Must Have Checksum. Run the page, and edit an order to see the checksum that is passed to the Master Detail page.
c. Set the Page Access Protection for the Employee Commission page to No URL Access.
d. Add a Commission button to the Top Tier Salary page that redirects to the Employee Commission page. Run the page to see that there is an error because of the branch.
e. Edit the Commission button to submit the page and create a Branch to Page branch that will go to the Employee Commission page without passing the URL.

Solution 14-1: Creating an Authentication Scheme

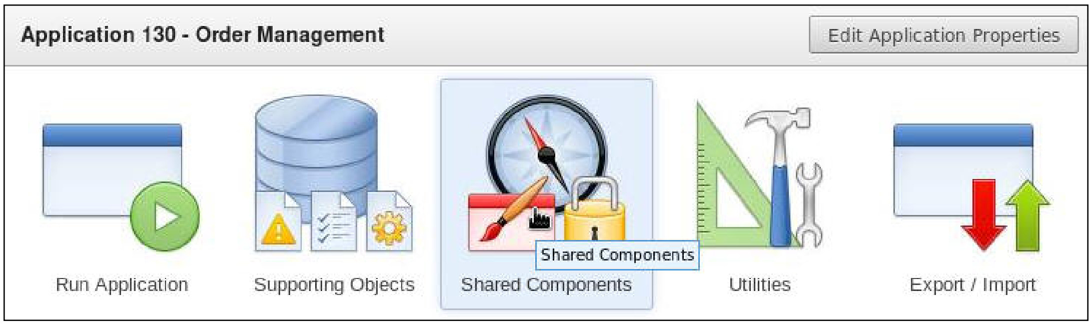
Overview

In this practice, you create an authentication scheme, and then make it the current default.

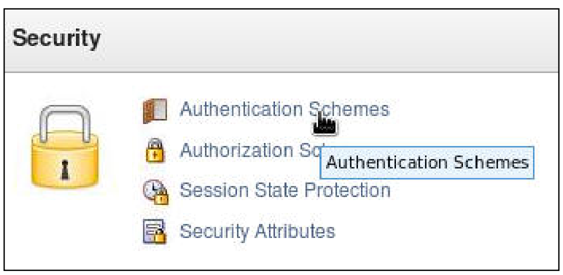
Tasks

a. Create an authentication scheme called OpenDoor based on the Open Door credentials.

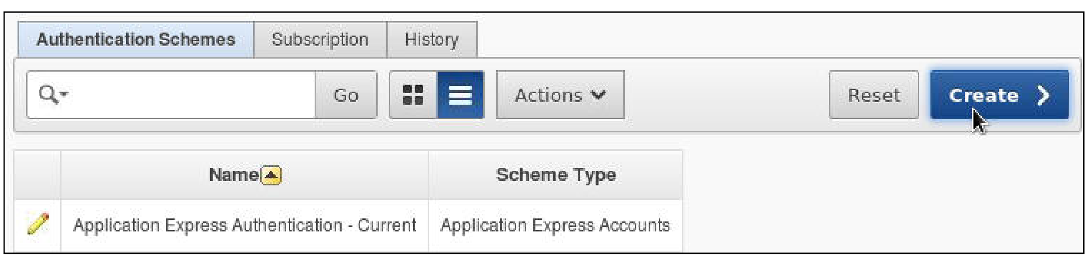
1) On the application home page, click **Shared Components**.



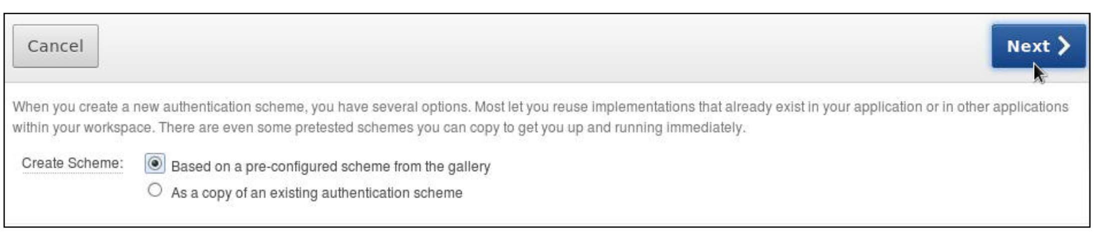
2) In the Security region, click the **Authentication Schemes** link.



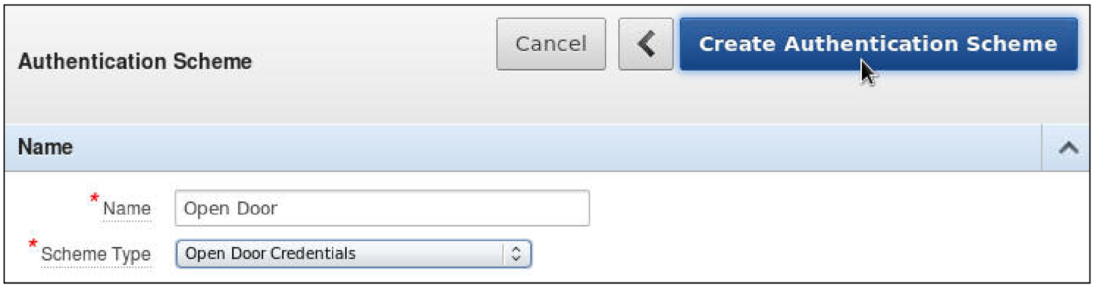
3) Click **Create >**.



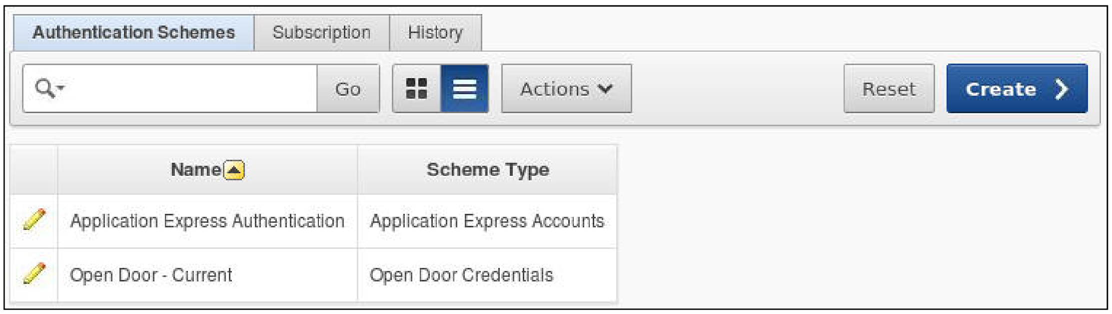
4) Leave the default value of "Based on a pre-configured scheme from the gallery" and click **Next >**.



5) For Name, enter **OpenDoor** and select the **Open Door Credentials** option from the Scheme Type list. Click **Create Authentication Scheme**.

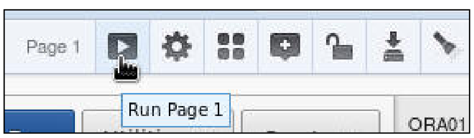


6) The scheme is created and made the current authentication scheme for the Order Management application.

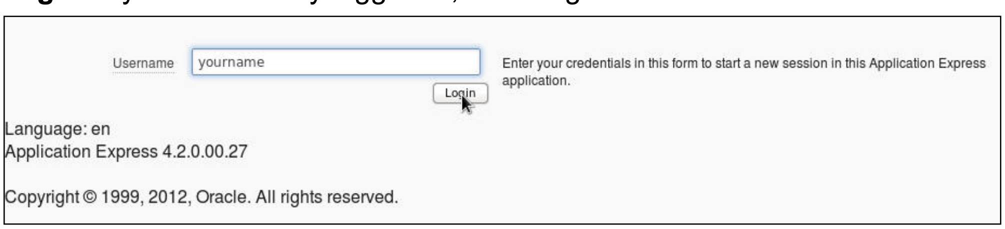


7) Search for the Home page of the Order Management application.

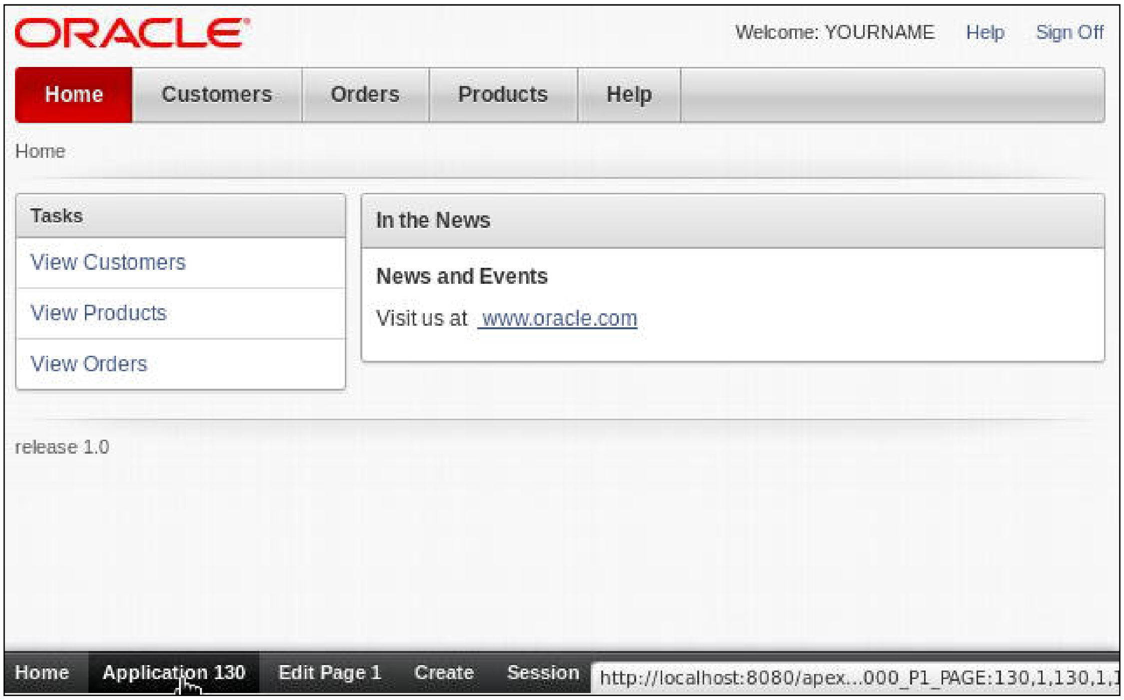
8) Click the **Run page** icon.



9) To log in to the system, enter your name in the Username field, and then click **Login**. If you are already logged in, click Logout.

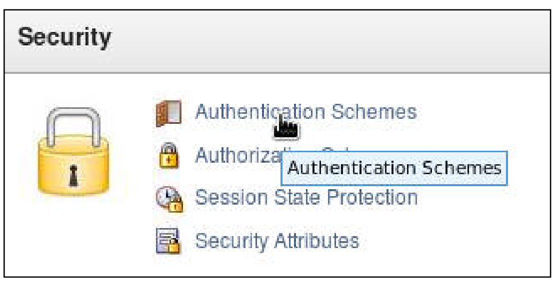


10) The Home page is displayed. Notice that the name that you entered appears in the Welcome area. In the next practice, you will use access control to set up authorization to certain pages and buttons. Click the **Application <*n*>** link on the Developer toolbar.

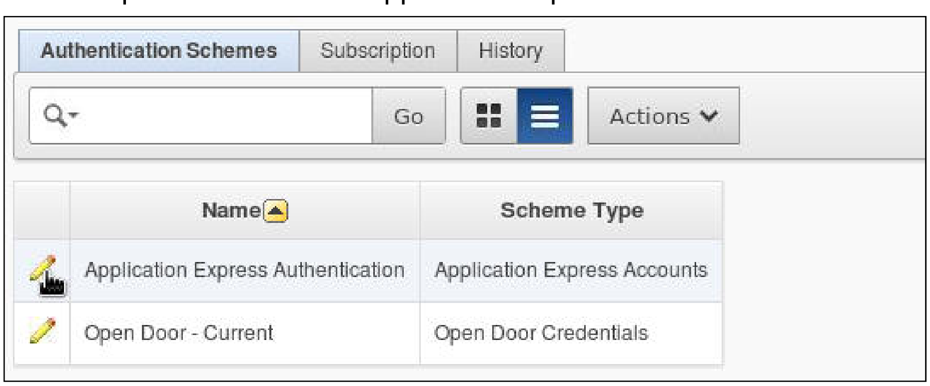


b. Switch the current authentication scheme to Application Express Authentication.

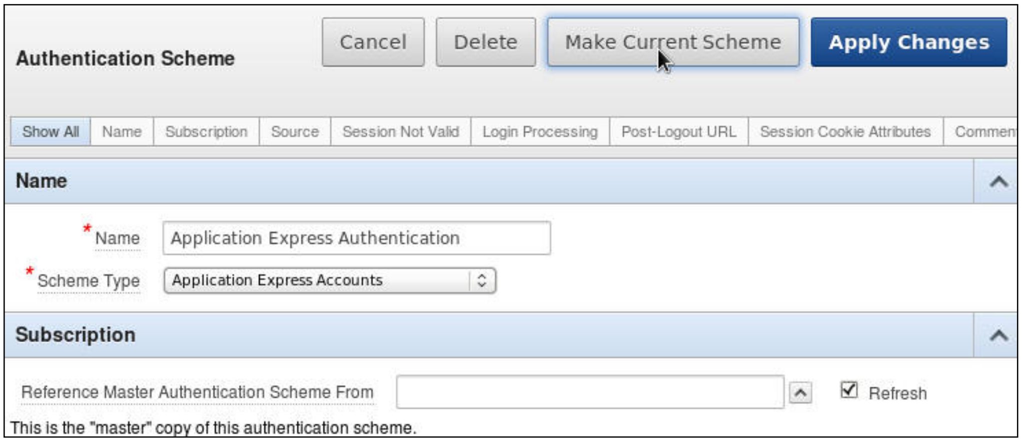
1) Navigate to **Shared Components** > **Authentication Schemes**.



2) Click the pencil icon next to Application Express Authentication.

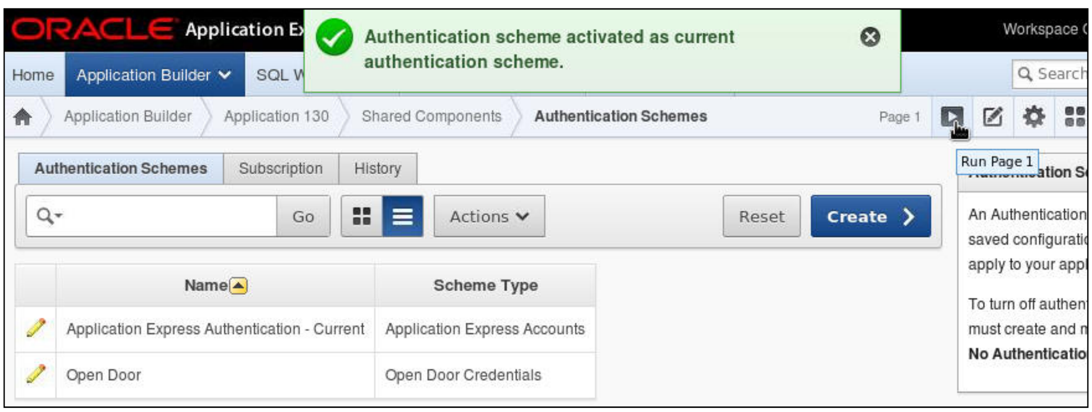


3) Click **Make Current Scheme**.

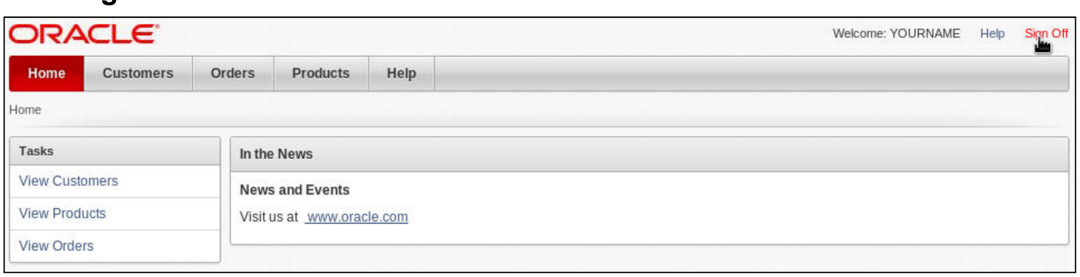


4) Click **OK**.

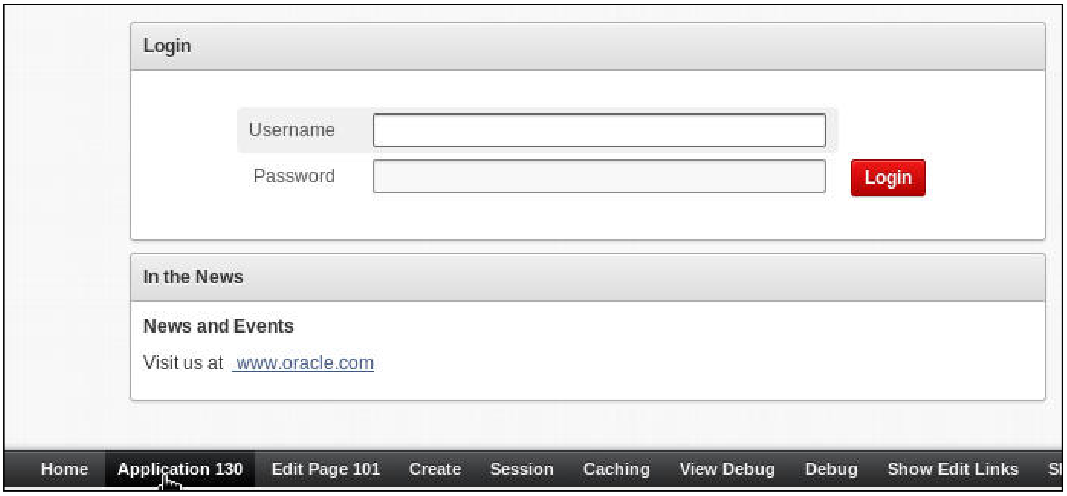
5) Application Express Authentication is now the current scheme. Click the **Run Page** icon again.



6) Click **Sign Off**.



7) Notice that the Login page uses the Application Express authentication login now instead of the OpenDoor login. Click the **Application <*n*>** link on the Developer toolbar.



Solution 14-2: Restricting Users by Using Access Control

Overview

In this practice, you create an access control page, set the application mode to restrict access, create the access control list, and assign application components to an authorization scheme.

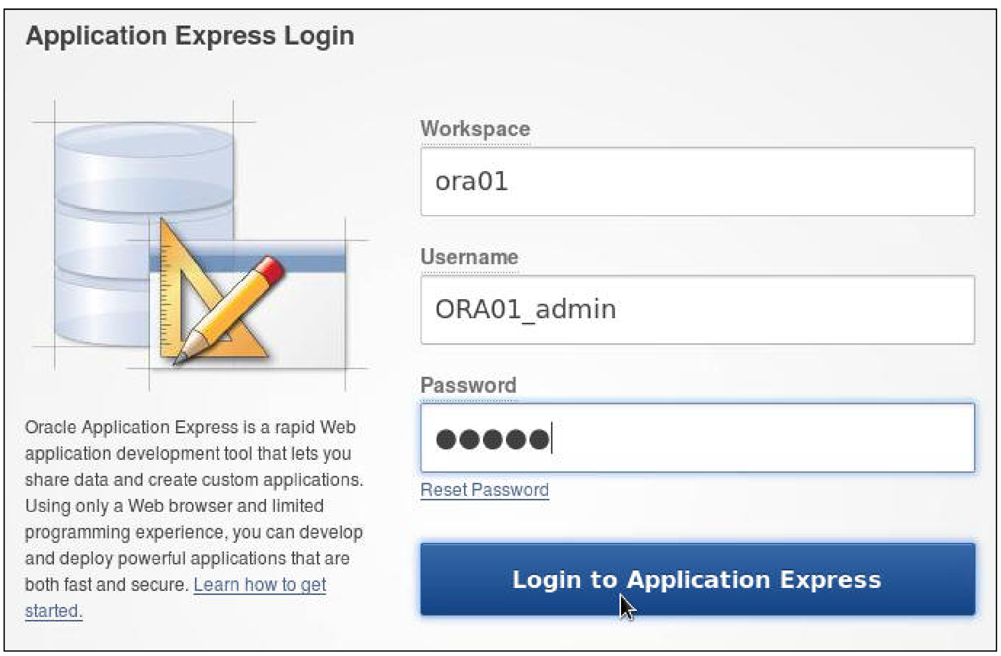
Tasks

a. Create the following users to add to the access control list.

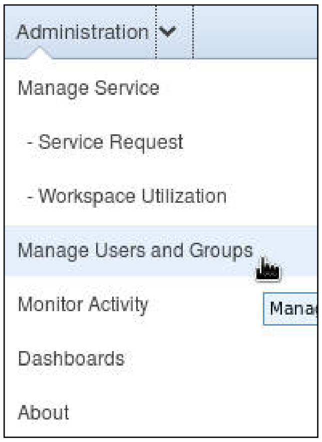
Name	User Type
Brad.Knight	Developer
Susie.Parker	Workspace Administrator
John.Bell	End User

1) Log out of Application Express. Click **Logout**.

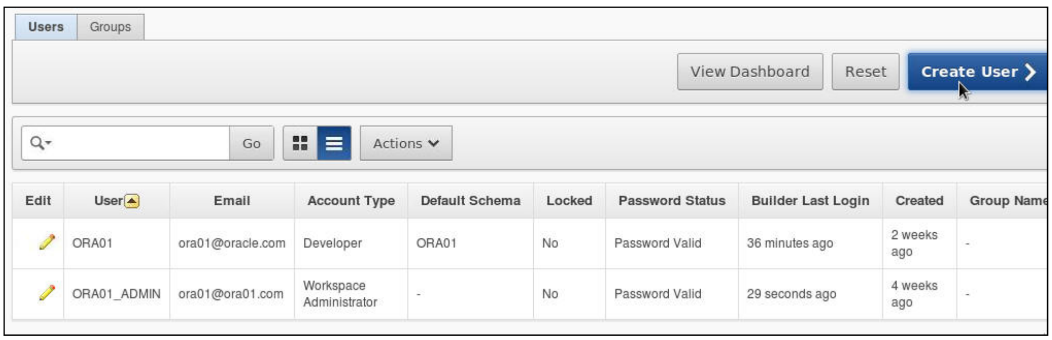
2) Log in as a workspace administrator. You might be prompted to change the password and log in again.



3) Click the down arrow next to **Administration** and select **Manage Users and Groups**.



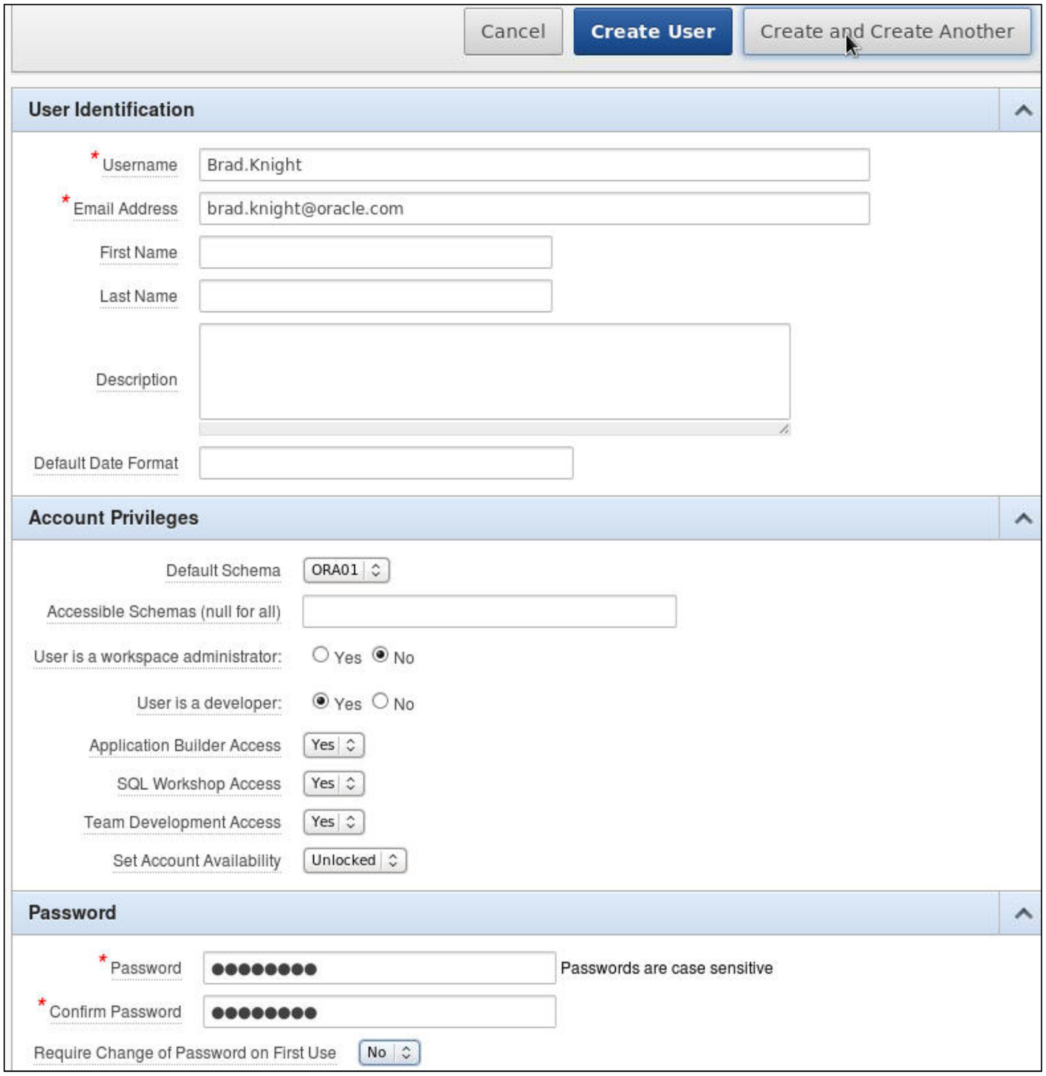
4) Click **Create User >**.



5) Enter the following information and click **Create and Create Another**.

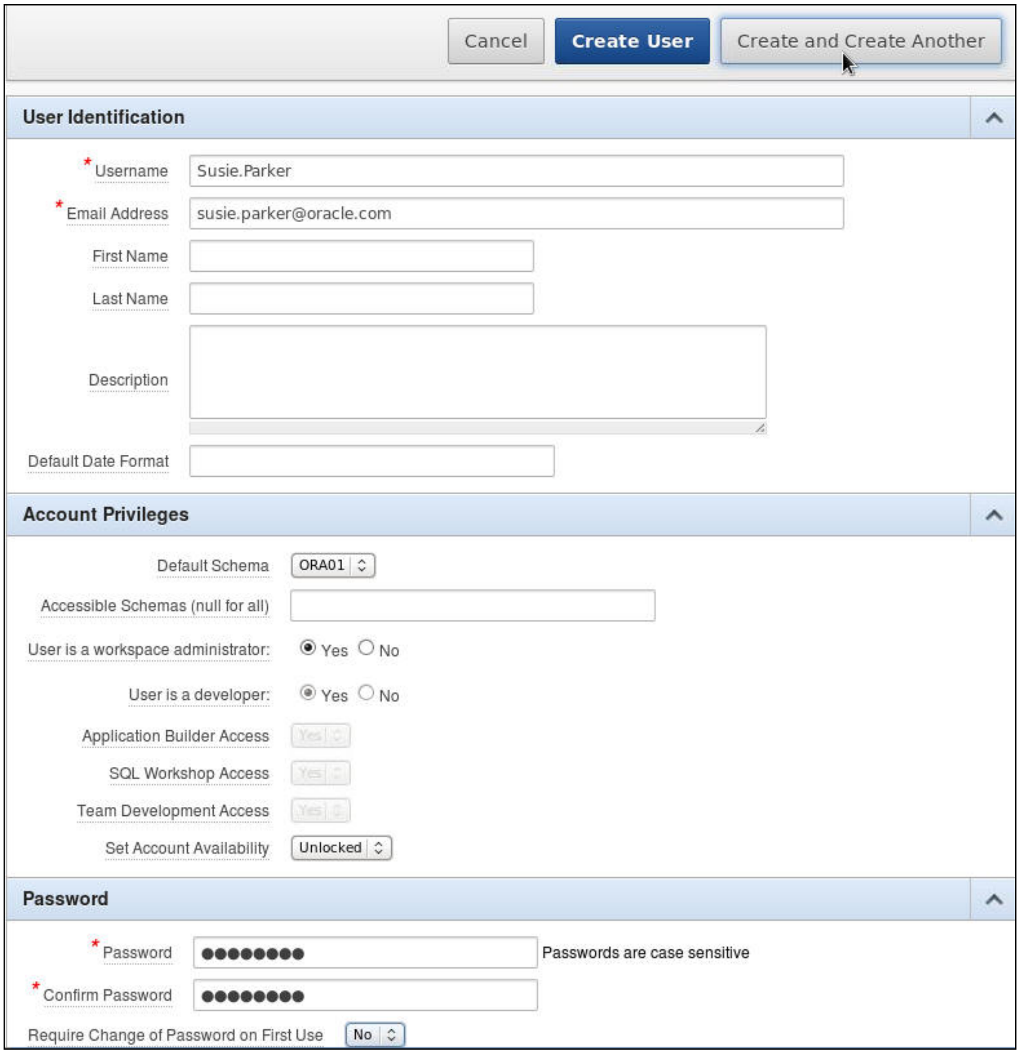
- User Name: **Brad.Knight**
- Email address: **brad.knight@oracle.com**
- Select **No** for the **User is a workspace administrator** option.
- Select **Yes** for the **User is a developer** option.
- Password: `welcome1`

- Confirm Password: **`welcome1`**
- Select **No** for **Require Change of Password on First Use**.



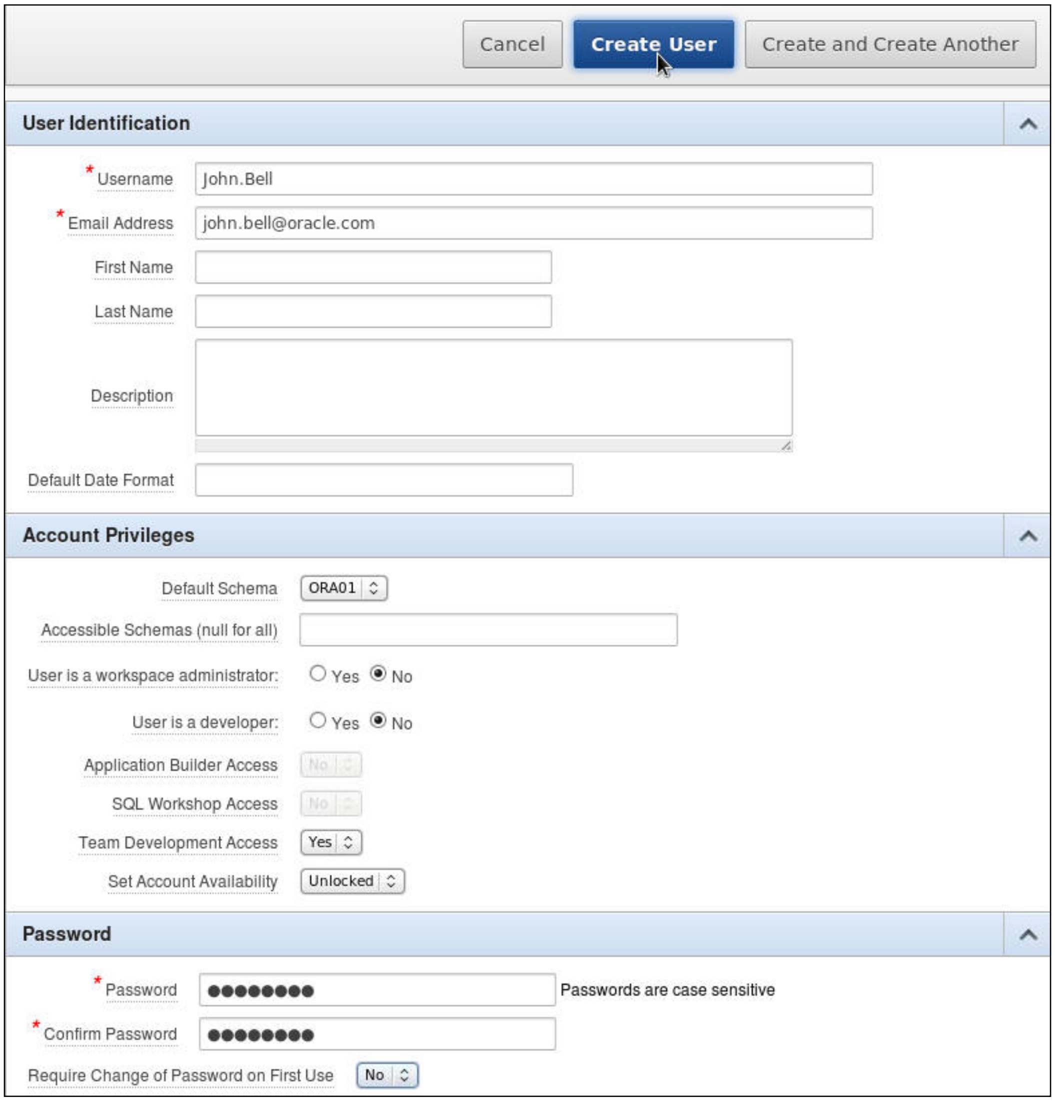
6) Enter the following information and click **Create and Create Another**.
- User Name: `Susie.Parker`
- Email address: **susie.parker@oracle.com**
- Select **Yes** for the **User is a workspace administrator** option.
- Password: **`welcome1`**
- Confirm Password: **`welcome1`**

- Select **No** for **Require Change of Password on First Use**.

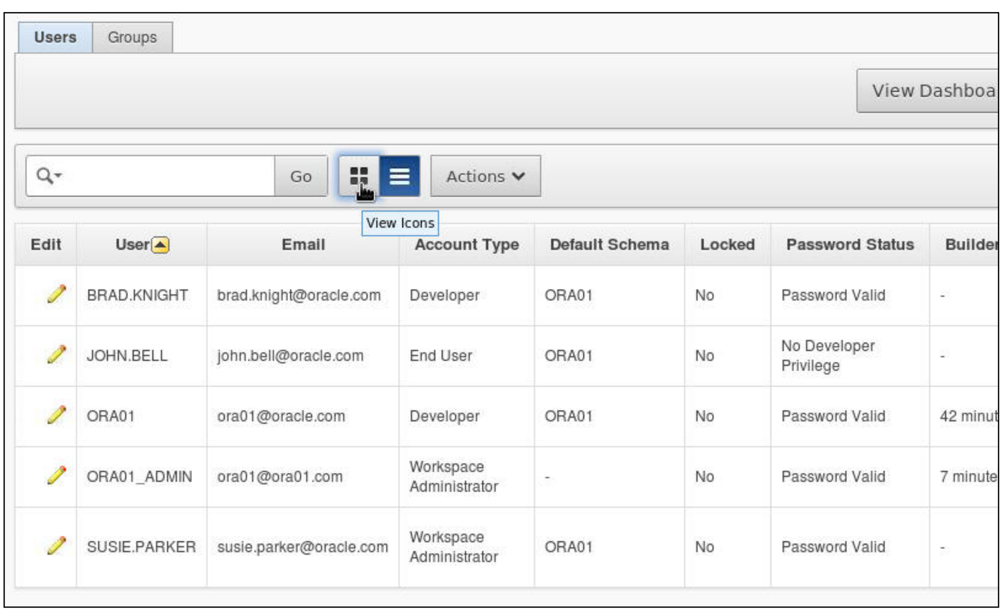


7) Enter the following information and click **Create User**.

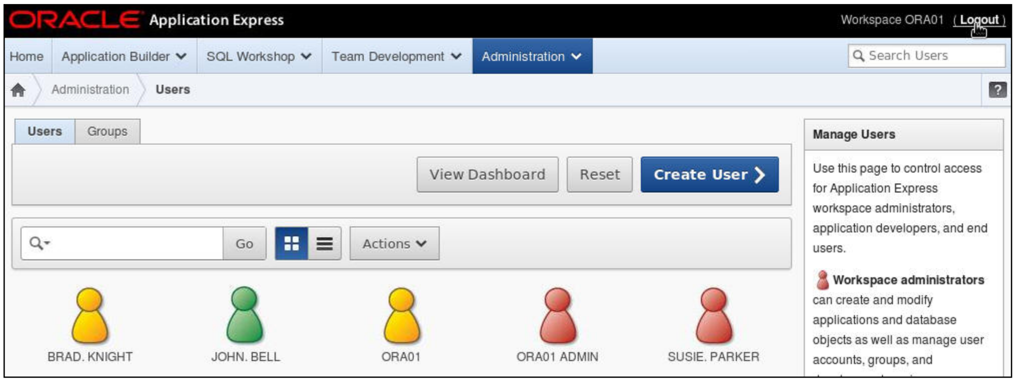
- User Name: `John.Bell`
- Email address: john.bell@oracle.com
- Select **No** for the **User is a workspace administrator** option.
- Select **No** for the **User is a developer** option.
- Password: **`welcome1`**
- Confirm Password: **`welcome1`**
- Select **No** for **Require Change of Password on First Use**.



8) The new users are now in the list. Click the **View icons** button.

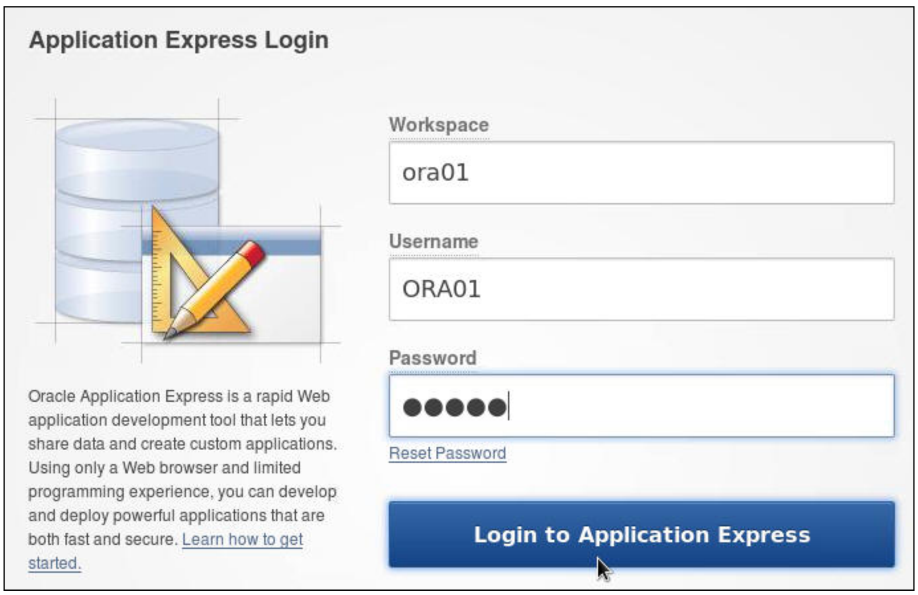


9) Click **Logout**.

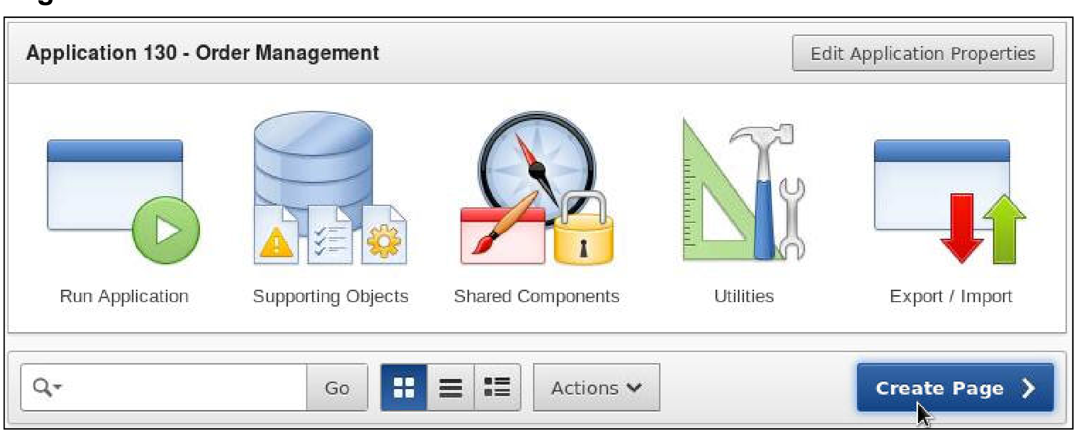


b. Create an access control page for the Order Management application.

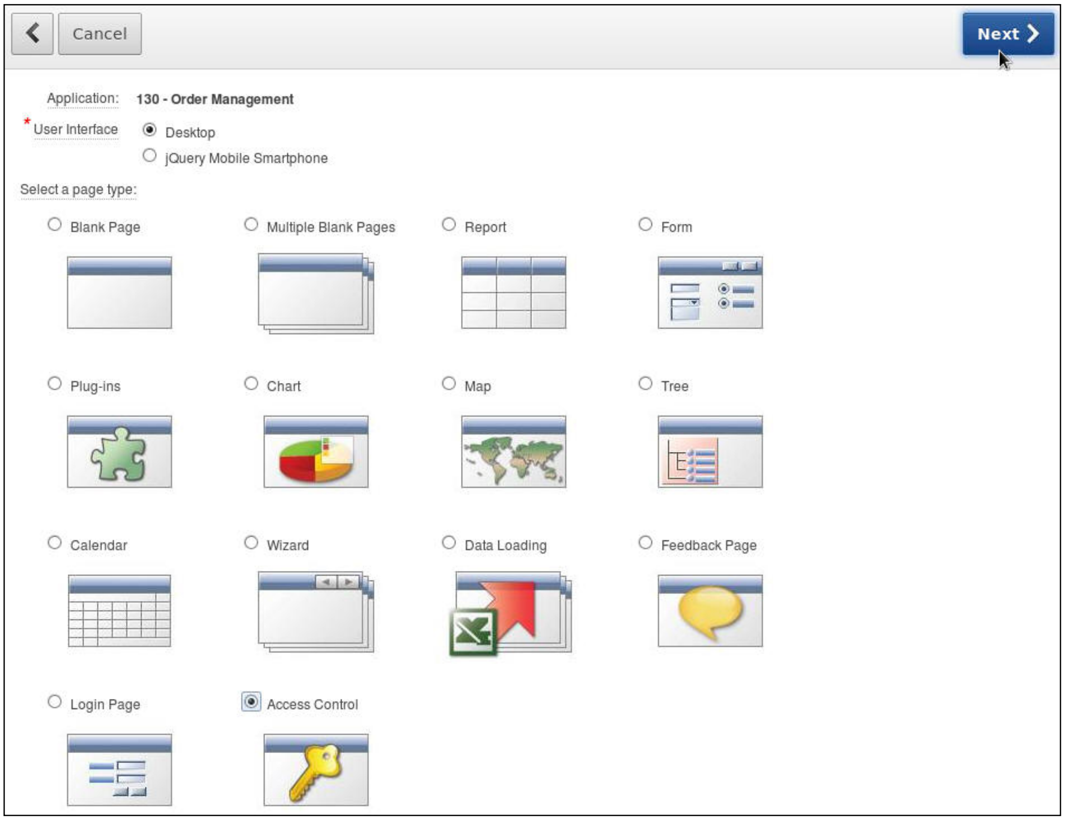
1) Log in to the workspace as the `ora<n>` developer.



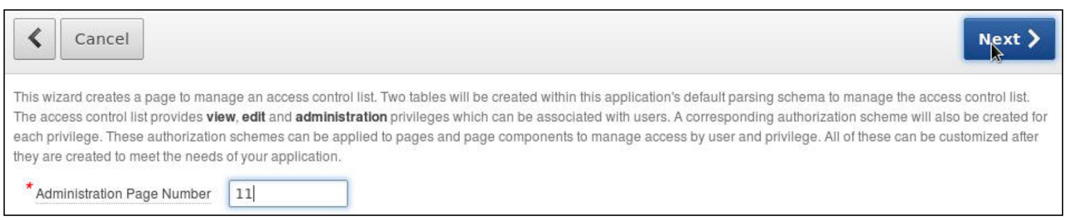
2) Navigate to your **Order Management** application home page and click **Create Page >**.



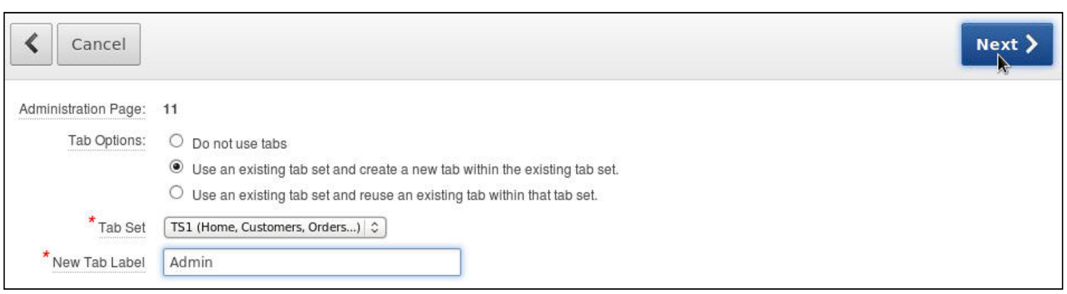
3) Select **Access Control** and click **Next >**.



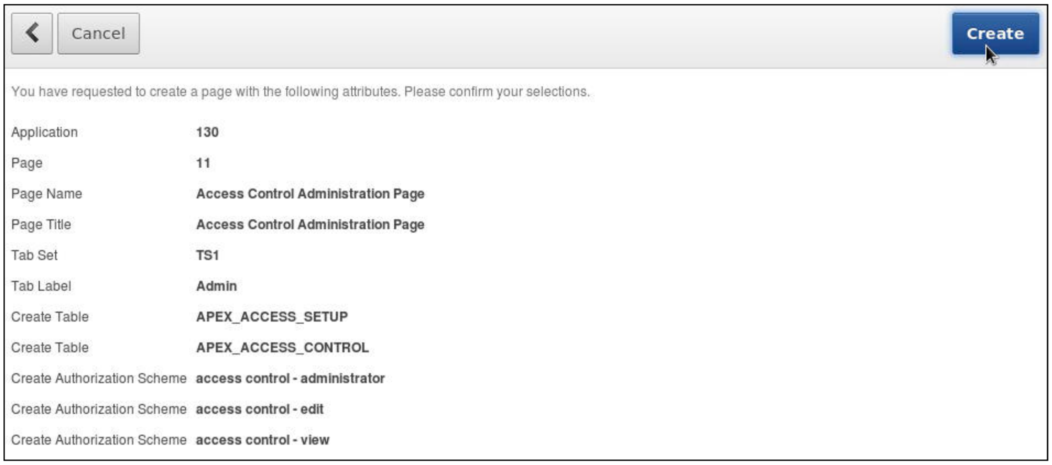
4) Accept the default page value and click **Next >**.



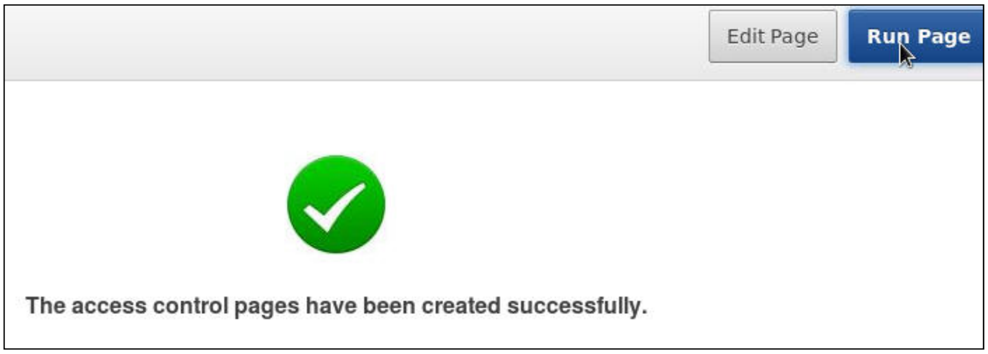
5) For Tab Options, select **Use an existing tab set and create a new tab within the existing tab set**, ensure **TS1** is selected for the Tab Set, and enter **Admin** for the New Tab Label. Then click **Next >**.



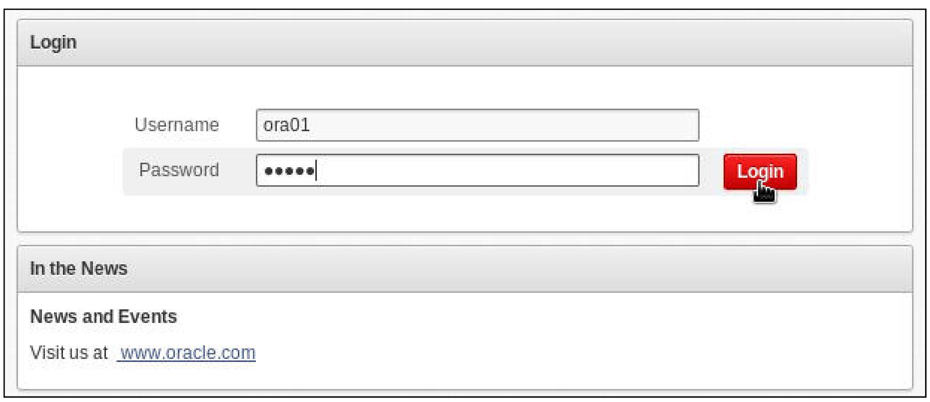
6) Click **Create**.



7) Click **Run Page**.

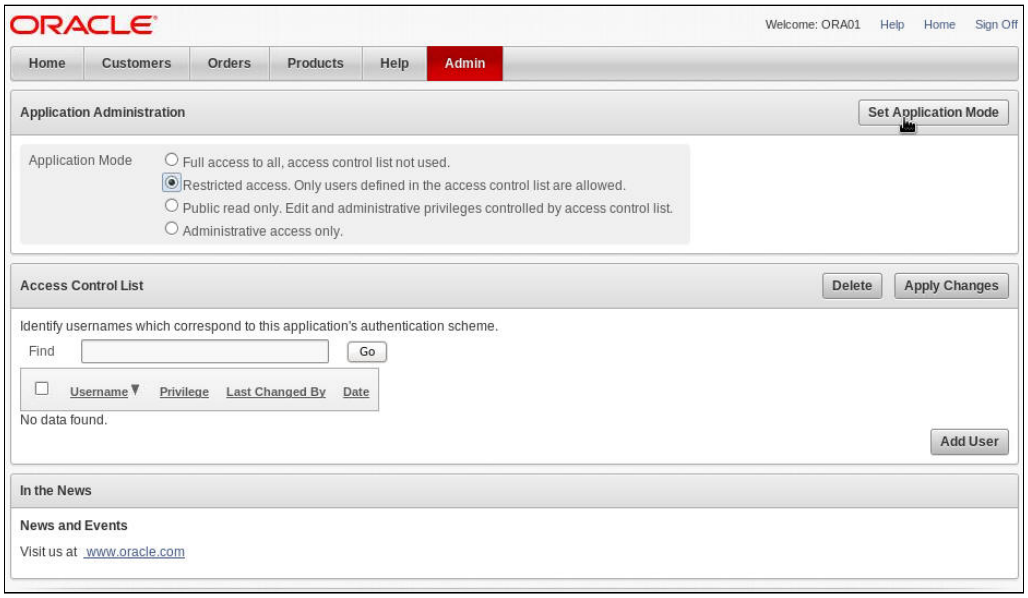


8) Log in as the **ora<*n*>** user.



c. Set the application mode to Restricted access. Only users defined in the access control list are allowed.

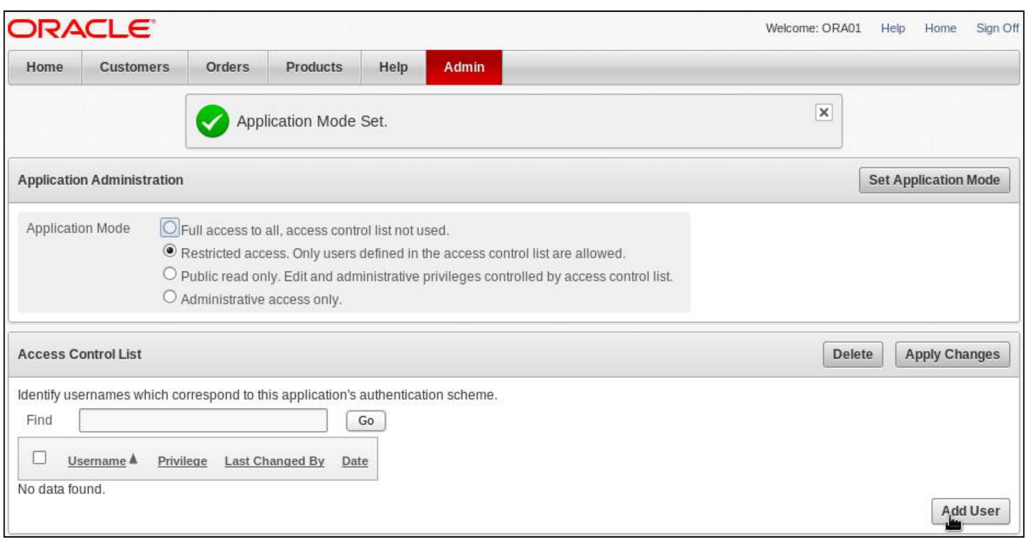
1) The access control page is displayed. Here is where you set the application mode and add users to the access control list. For Application Mode, select **Restricted access, Only users defined in the access control list are allowed**, and click **Set Application Mode**.



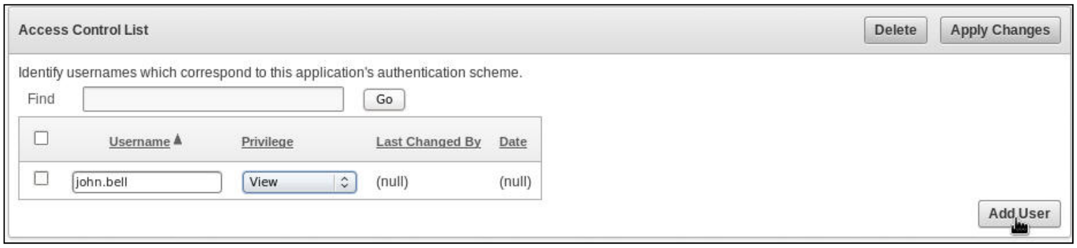
b. Add users to the access control list.

- `john.bell` can only view the information in the application; he cannot make any changes to application data.
- `brad.knight` should be allowed to edit application data, but he cannot change application administration settings (application mode and the access control list).
- `susie.parker` is the administrator of the application, so she can change anything in addition to changing the user privileges.
- `ora<n>` is also an administrator of the application.

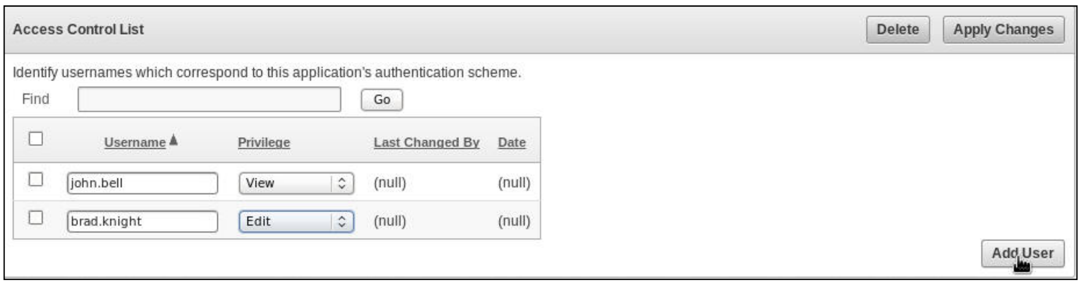
1) The application mode has been set. Now, users whose credentials are defined in the access control list can access your Order Management application. To add your privileged users, click **Add User**.



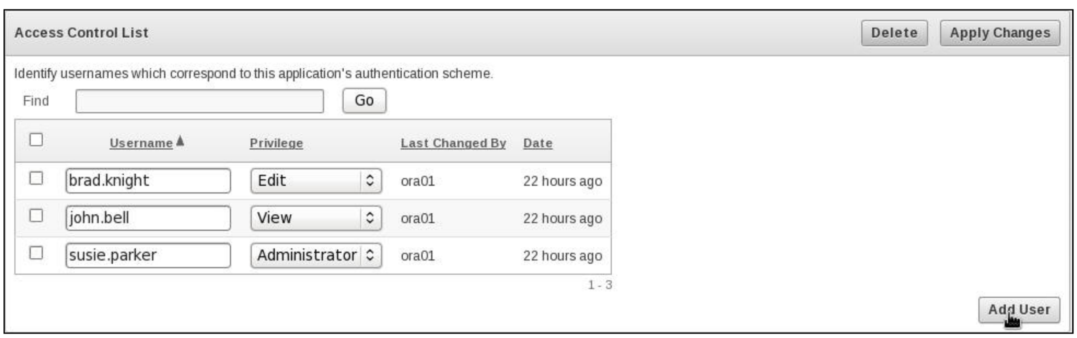
2) You want John Bell to only be able to view the information in your application. He cannot make any changes to application data. Enter `john.bell` as the username and select **View** from the list of privileges. Then click **Add User**.



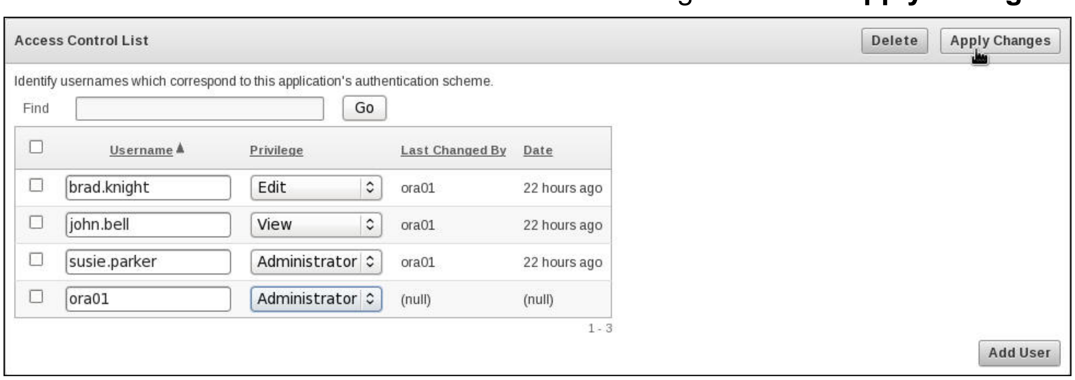
3) You want Brad Knight to be able to edit the information in your application, but not be able to change any of the administration options. Enter `brad.knight` as the username and select **Edit** from the Privilege list. Click **Add User**.



4) You want Susie Parker to be able to edit the information in your application as well as update the administration access control list. Enter `susie.parker` as the username and select **Administrator** from the Privilege list. Click **Add User**.



5) Lastly, you want `ora<n>` to have administrator access. Enter `ora<n>` as the username and select **Administrator** from the Privilege list. Click **Apply Changes**.

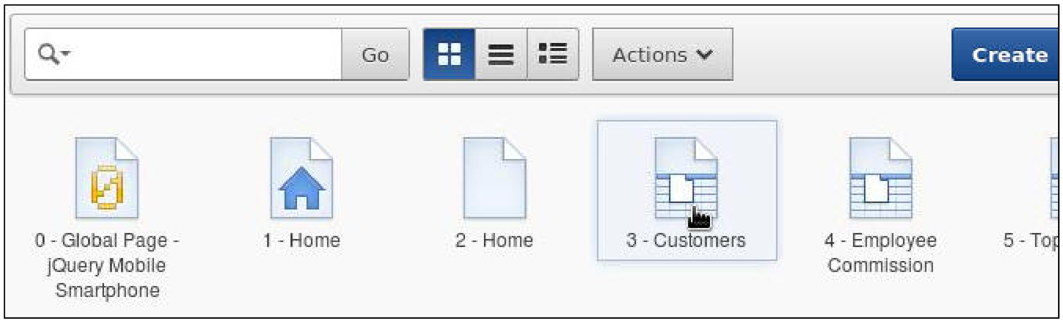


c. Define and apply the authorization schemes to each application component. This will restrict access to application pages and components.

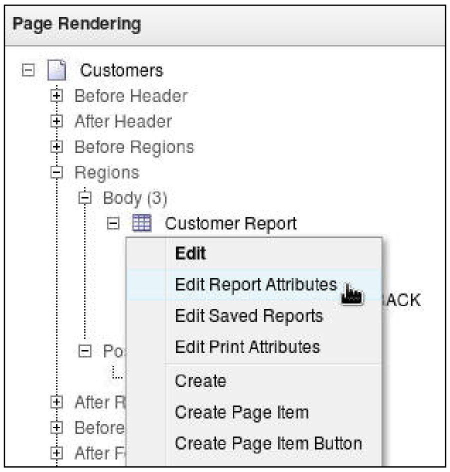
- Users with the View privilege can review customer information, but cannot change or create information.
- Users with the Edit privilege can make changes to customer information but cannot make changes to the application mode and the access control list.
- Users with the Administrator privilege can make any changes, including administering the Order Management application.

1) You can define which areas of the application are restricted. Click the **Application <*n*>** link on the Developer toolbar.

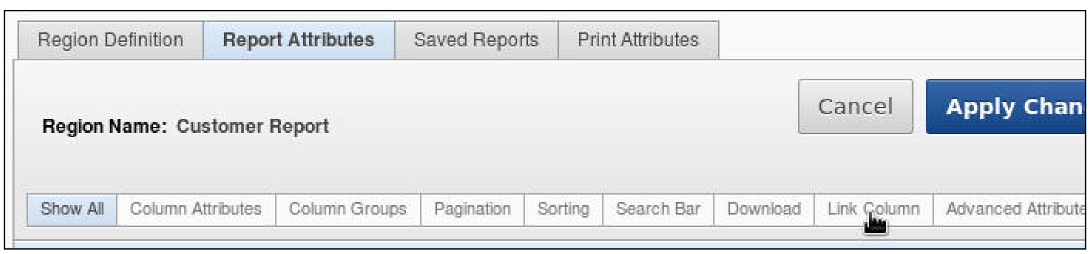
2) You can restrict access for users with the View privilege to customer information. On the **Order Management** application home page, click the **Customers** page.



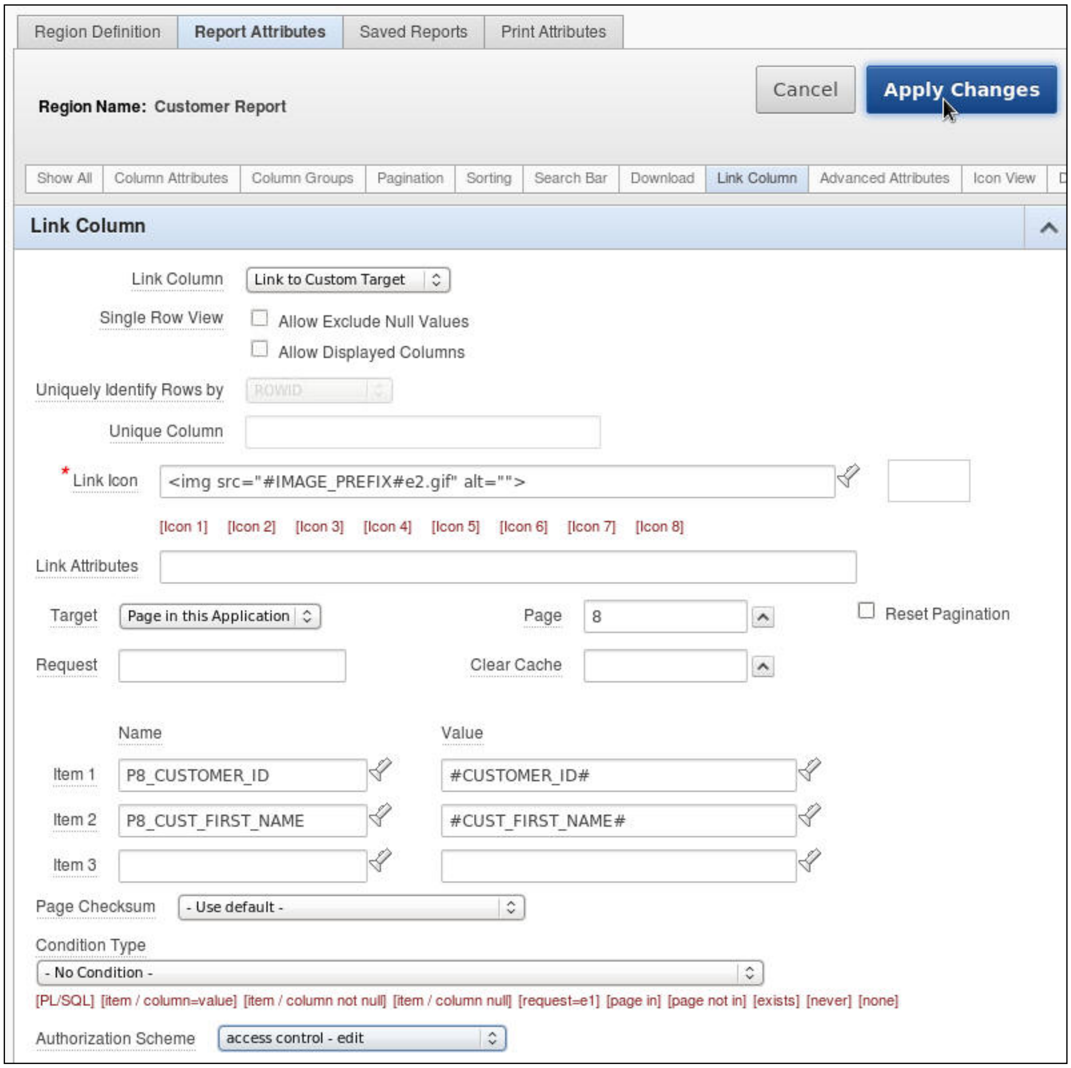
3) Right-click the **Customer Report** region and select **Edit Report Attributes**.



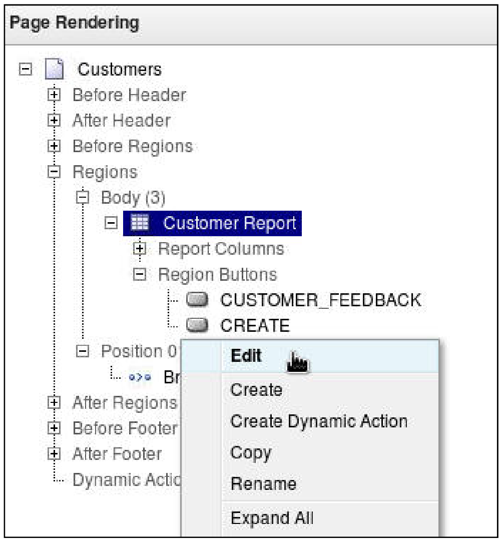
4) Click the **Link Column** tab.
Note: If the report were a classic report, you would modify the `CUSTOMER_ID` attribute and set the authorization in the column attribute definition instead of at the report level (as in an interactive report).



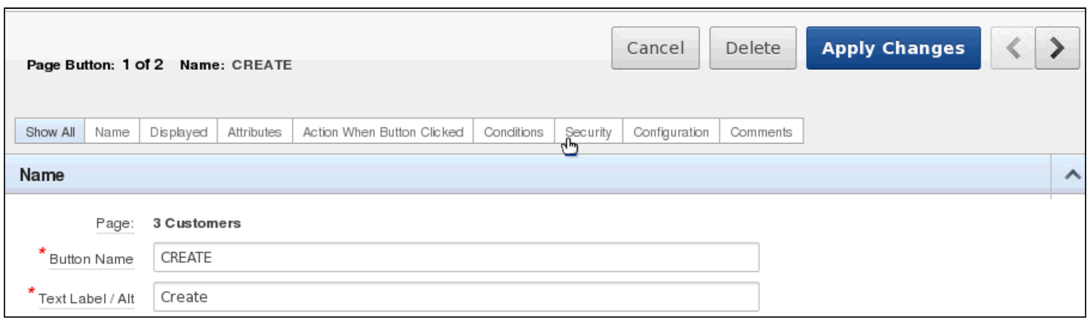
5) Select **access control - edit** for Authorization Scheme and click **Apply Changes**.



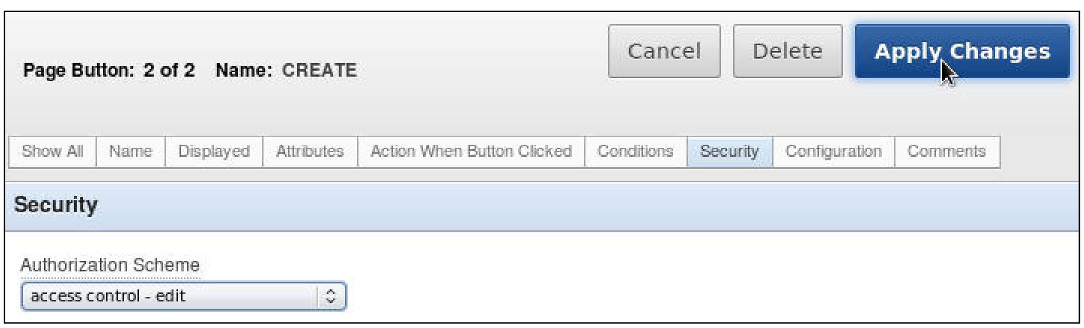
6) Because you want the Create Button to appear only if the user has the Edit or Administrator privilege, you must set the authorization scheme accordingly. Under Region Buttons for the Customer Report region, right-click **CREATE** and select **Edit**.



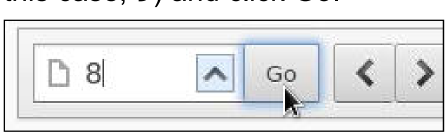
7) Click the **Security** tab.



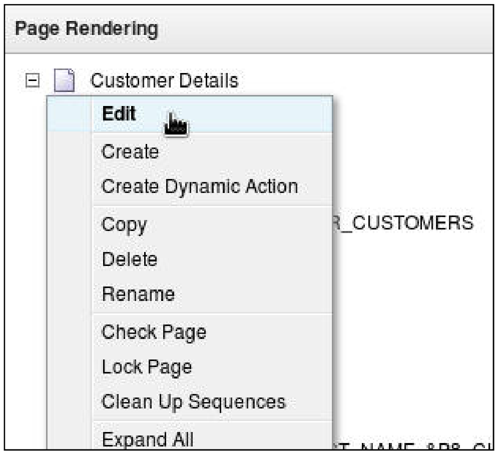
8) Select the **access control - edit** authorization scheme and click **Apply Changes**.



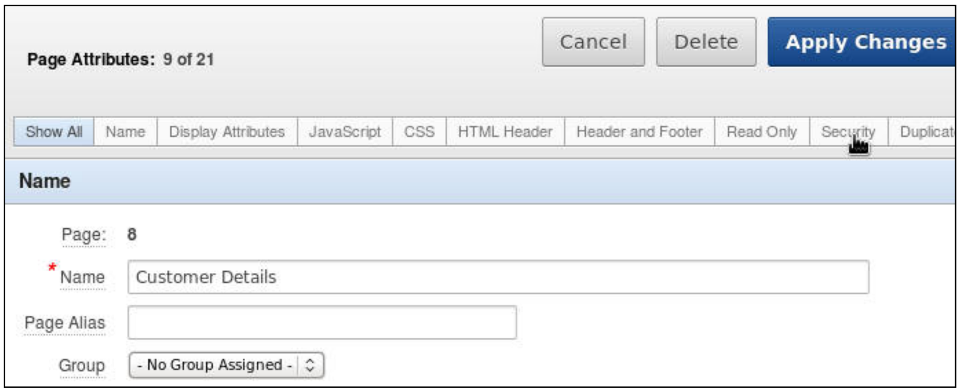
9) You also want to protect against direct access to a page. Even though you restricted a user who did not have the Edit privilege from editing or creating users on the Customers page, he or she can still access Customer Details if the correct URL is entered. To prevent this from happening, you must restrict Customer Details to only users with the Edit privilege. Enter the page number of the **Customer Details** page (in this case, `9`) and click Go.



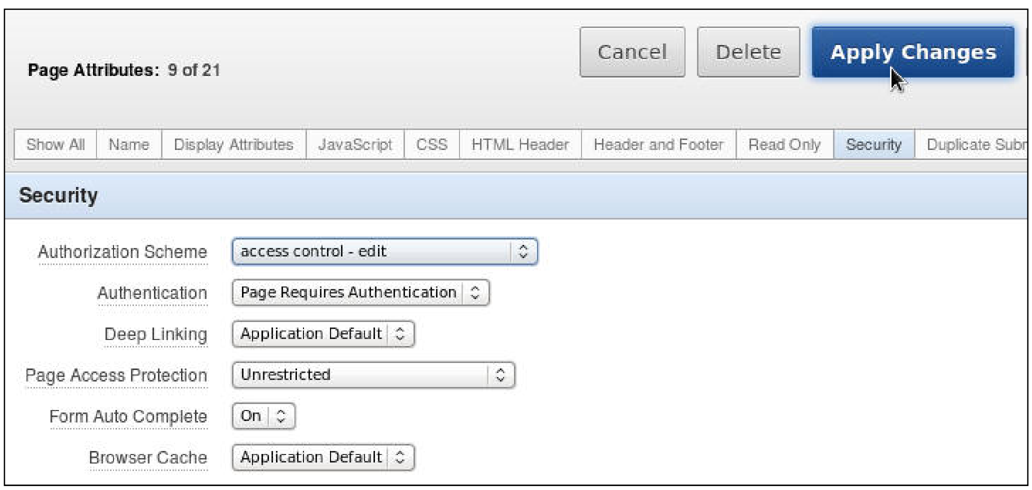
10) Right-click the **Customer Details** page and select **Edit**.



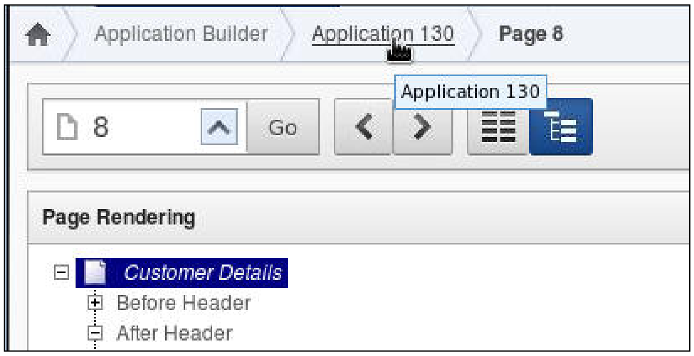
11) Click the **Security** tab.



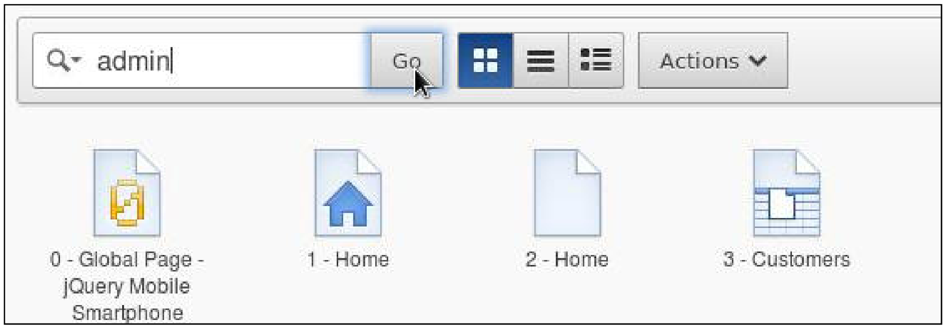
12) For Authorization Scheme, select **access control - edit**. Click **Apply Changes**.



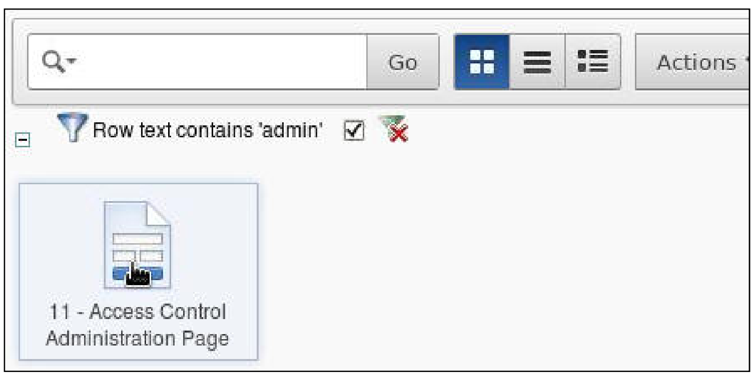
13) Click the **Application<*n*>** breadcrumb to return to the Order Management application home page.



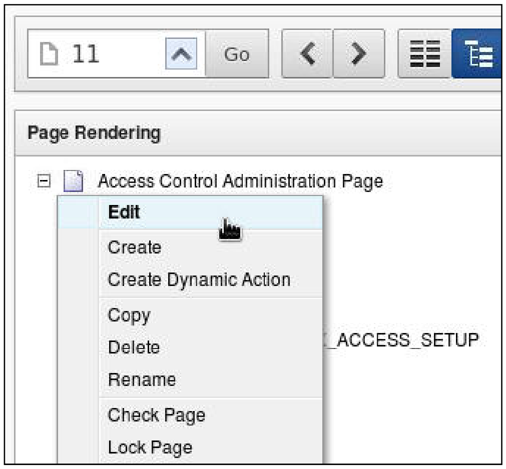
14) Enter **admin** in the search area and click **Go**.



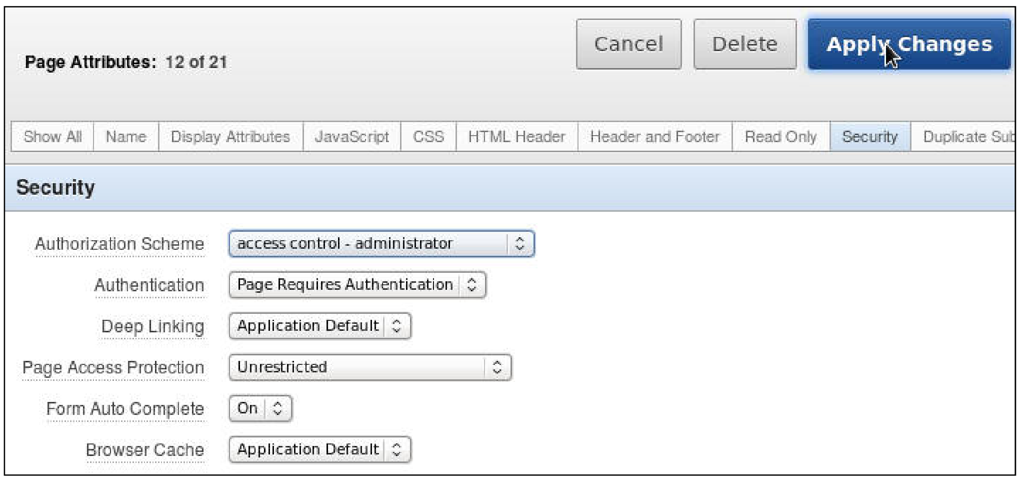
15) Because users with the Administrator privilege are only allowed to make changes to the access control list, you must set the authorization scheme for the Access Control Administration page. Click **Access Control Administration Page**. Note that you may have to advance to the second page of icons to see this page.



16) Right-click **Access Control Administration Page** and select **Edit**.



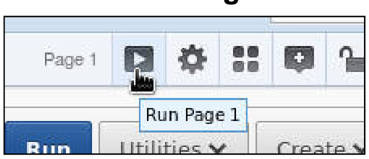
17) Select **access control - administrator** for the Authorization Scheme and click **Apply Changes**.



d. Run the application and log in as a different user. What do you observe?

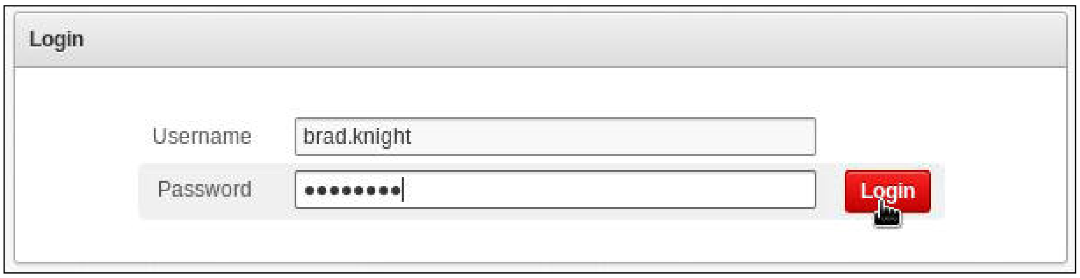
1) Now you are ready to run the application. Enter `1` for Page and click **Go**.

2) Click the **Run Page** icon.

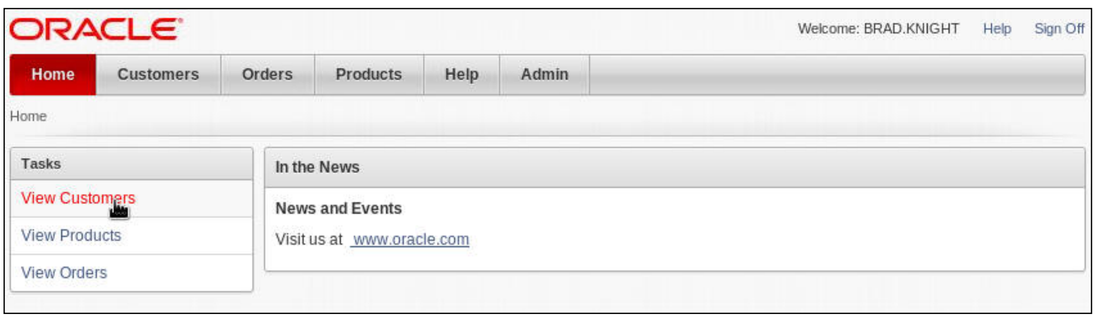


3) If you are already logged in, click **Sign Off**.

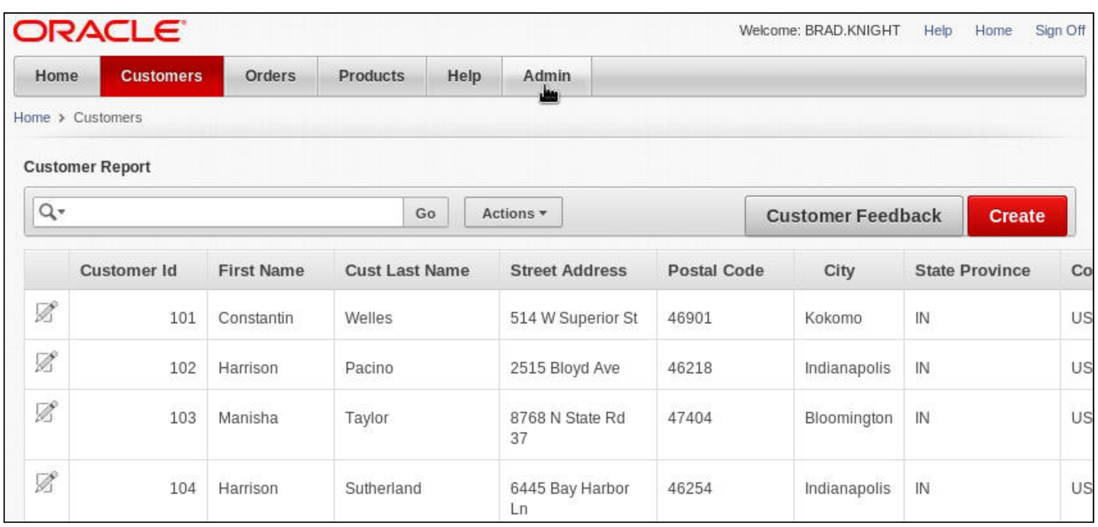
4) Enter **`brad.knight`** and **`welcome1`** for the username and password, respectively. Then click **Login**.



5) Click **View Customers**.



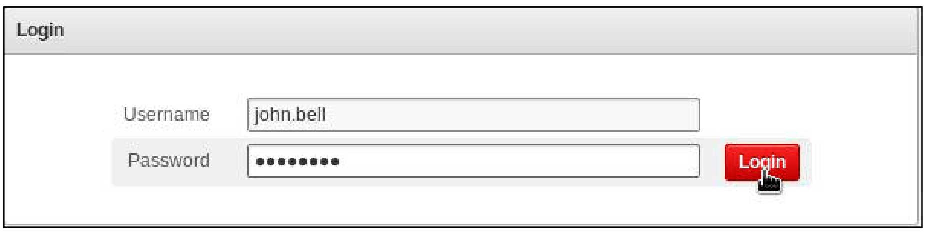
6) Notice that Brad can edit customer information, and he has the Create button. Click the **Admin** tab.



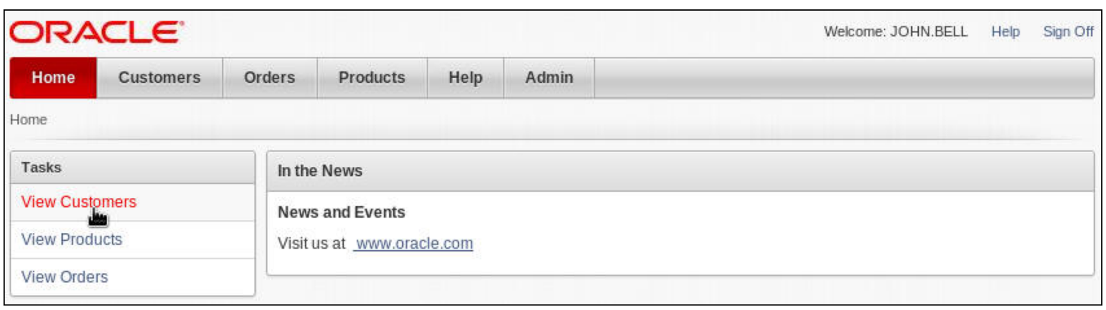
7) You receive an error because you need the Administration privilege to access this page. Click **OK**.

8) Click **Sign Off**.

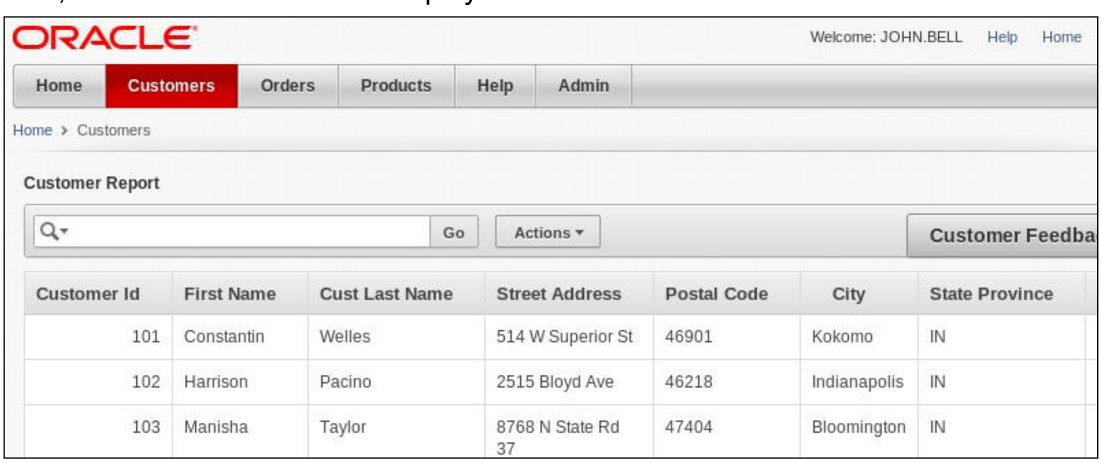
9) Enter `john.bell` and `welcome1` as the username and password, respectively. Then click **Login**.



10) Click **View Customers**.



11) John has the View privilege only and, therefore, cannot edit customer information. Also, the Create button is not displayed for him.



12) Change the page number in your URL to try to access the Customer Details page. Press the **Enter** key:

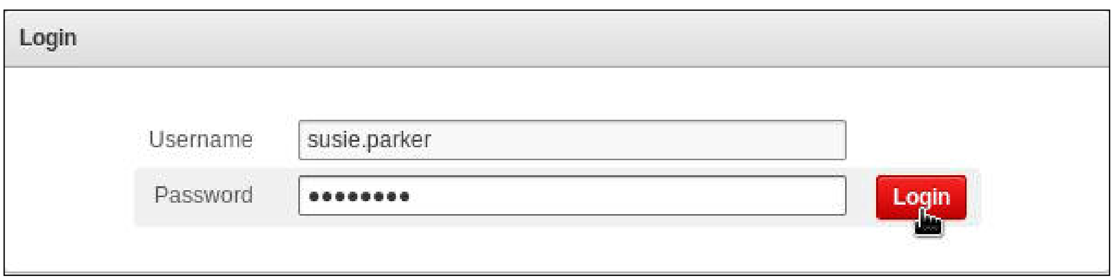
Example url …/f?p=104:8:2101953412249296357::NO

Change to …/f?p=104:**<customer details page number>**:2101953412249296357::NO

13) Notice that you receive a message denying access to the page, because access to the Customer Details page is restricted to the Edit privilege users. Click **OK.**

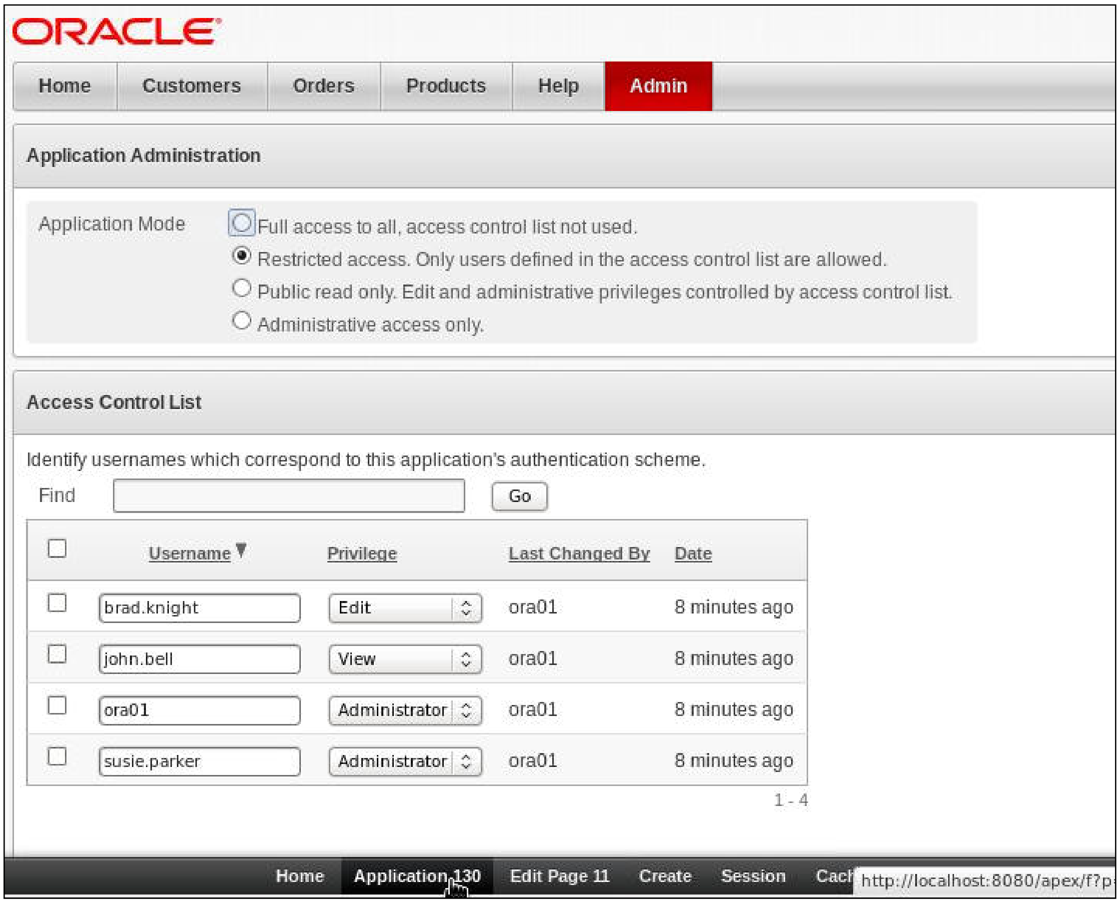
14) Lastly, you want to log in as Susie Parker who is an administrator. Click **Sign Off**.

15) Enter **susie.parker** for Username and **welcome1** for Password and click **Login**.



16) Click the **Admin** tab.

17) Notice that because Susie is an Administrator, you can access this page. Click the **Application <*n*>** link on the Developer toolbar.



Solution 14-3: Using Session State Protection

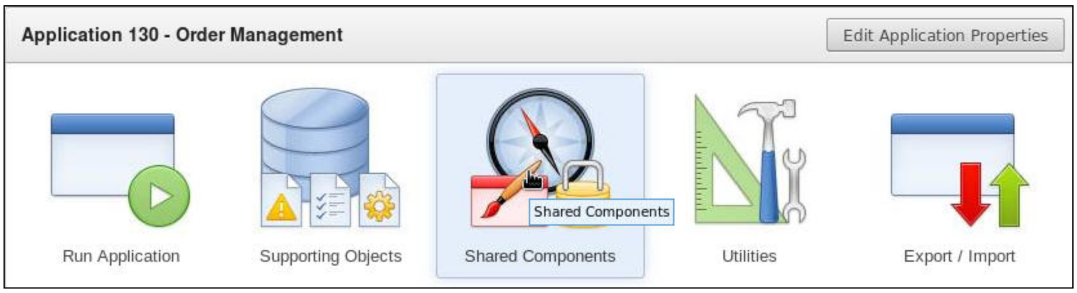
Overview

In this practice, you enable Session State Protection, set the Session State Protection for a particular item, and review the various options for page access protection so that you can examine their differences.

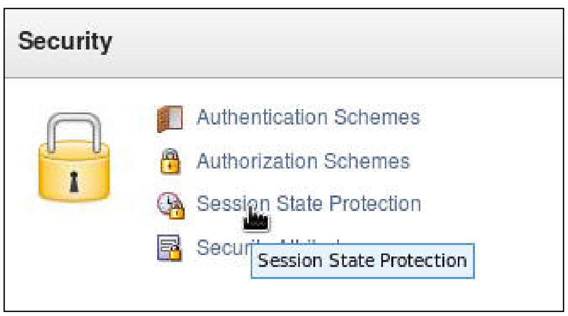
Tasks

a. Set the item session state protection for the P<n>_ORDER_ID on the Master Detail page to be Checksum Required – Session Level. Run the page to see the error message.

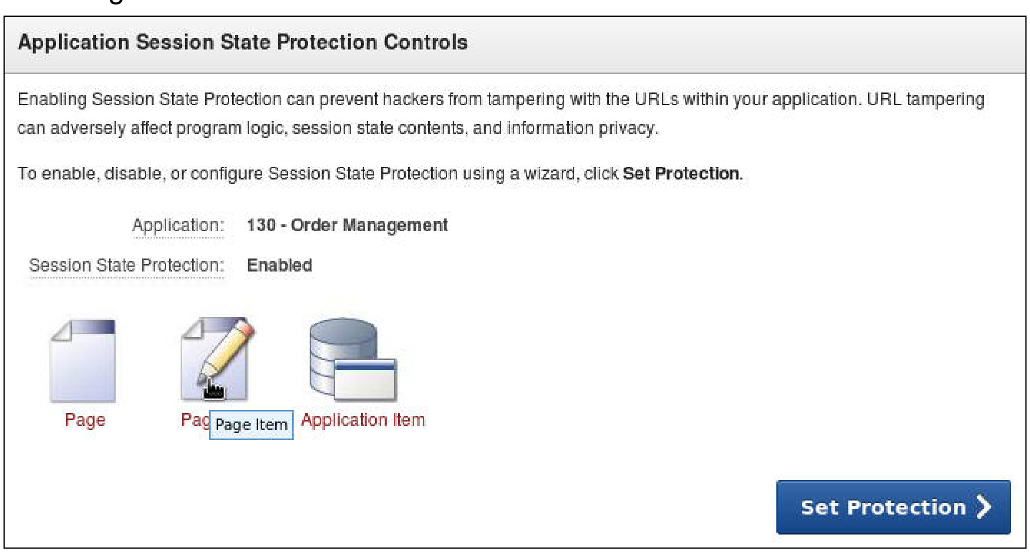
1) On your Order Management application home page, select **Shared Components**.



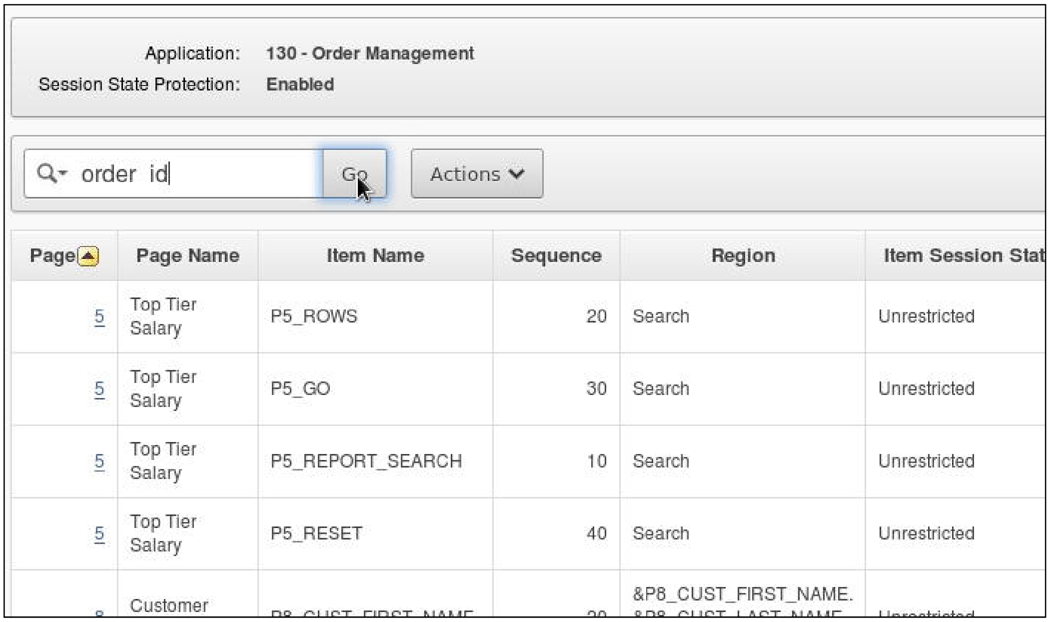
2) Under Security, select **Session State Protection**.



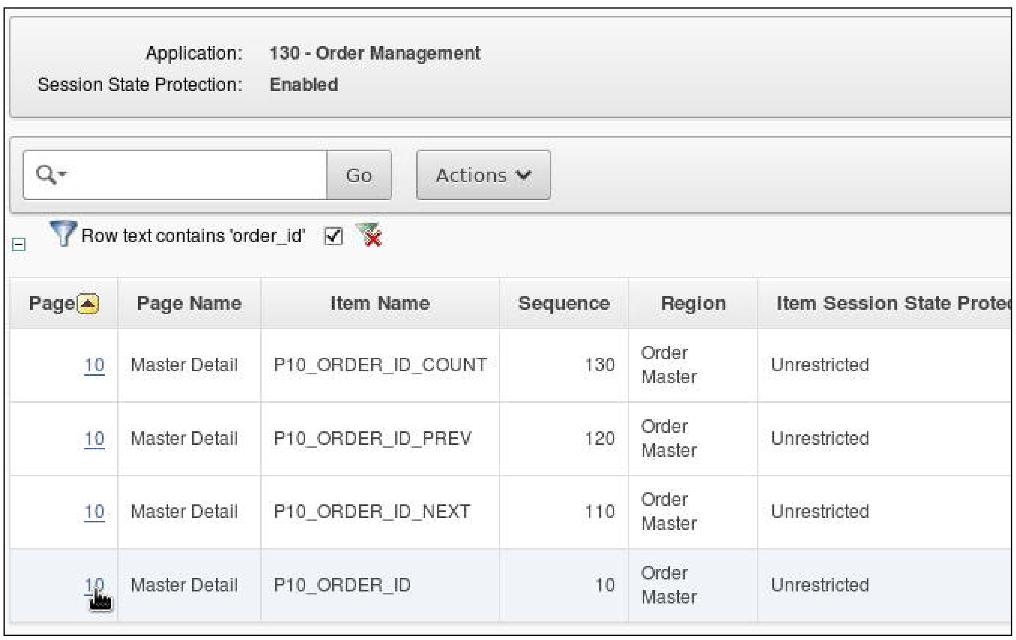
3) Notice that Session Protection is enabled. On the Master Details page, you want to set the item security for P<n>_ORDER_ID to Checksum required – Session Level. Click Page Item.



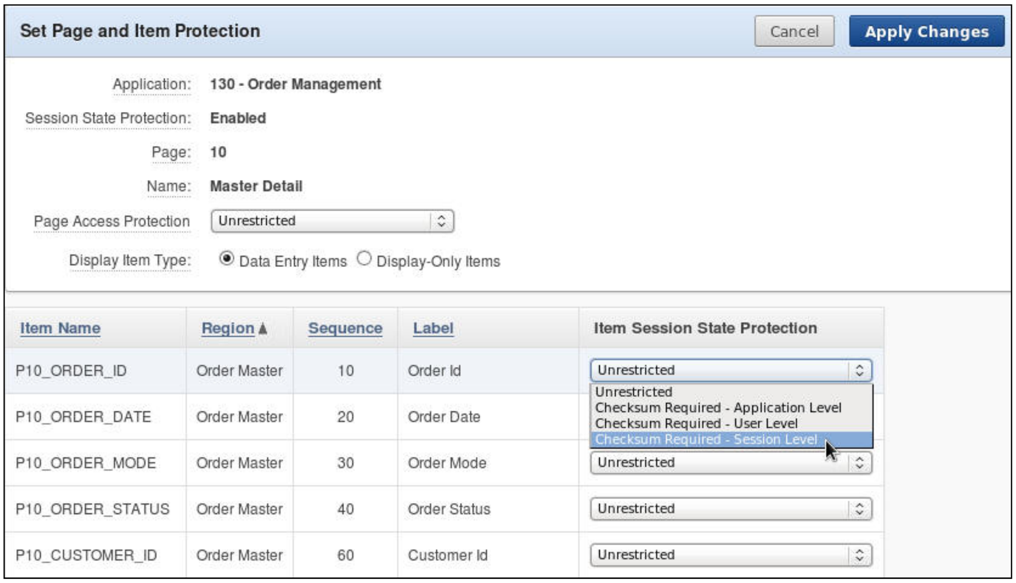
4) Enter **order_id** in the search area and click **Go**.



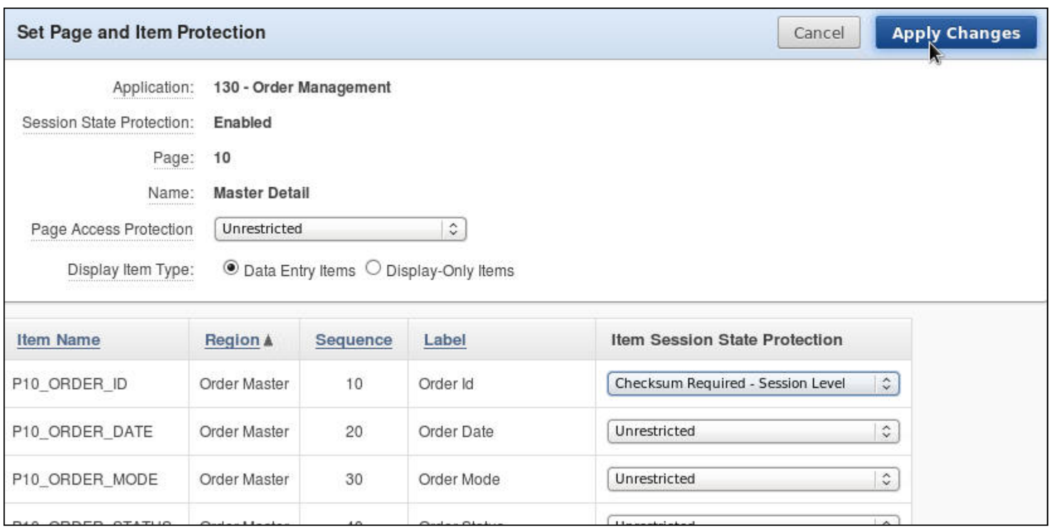
5) Click the Page link for the P<n>_ORDER_ID item.



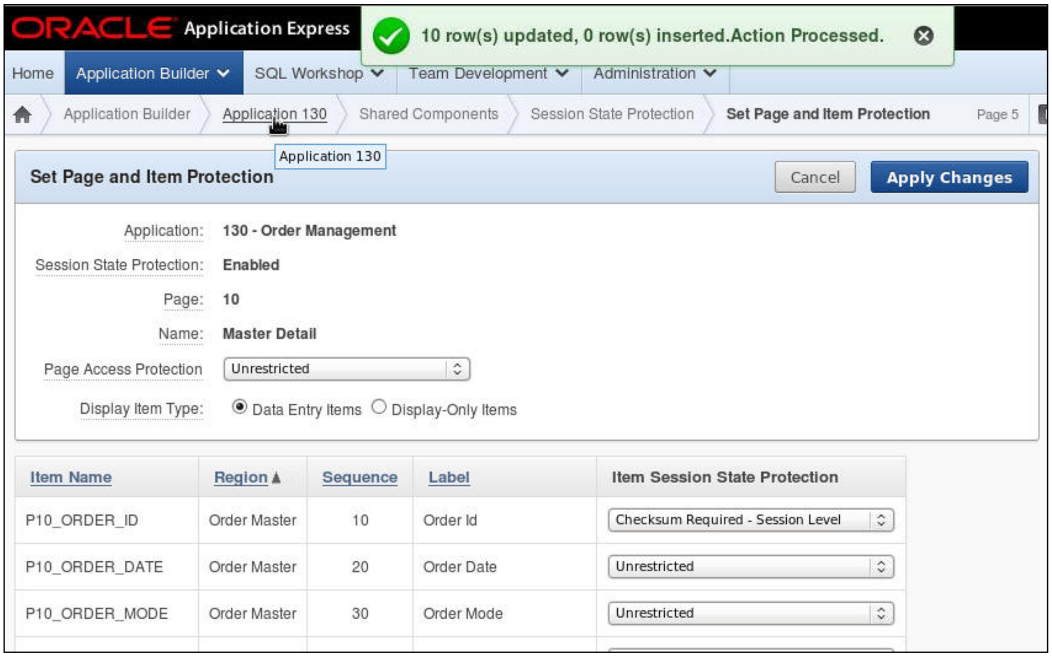
6) Select **Checksum Required – Session Level** for Item Session State Protection for P<n>_ORDER_ID.



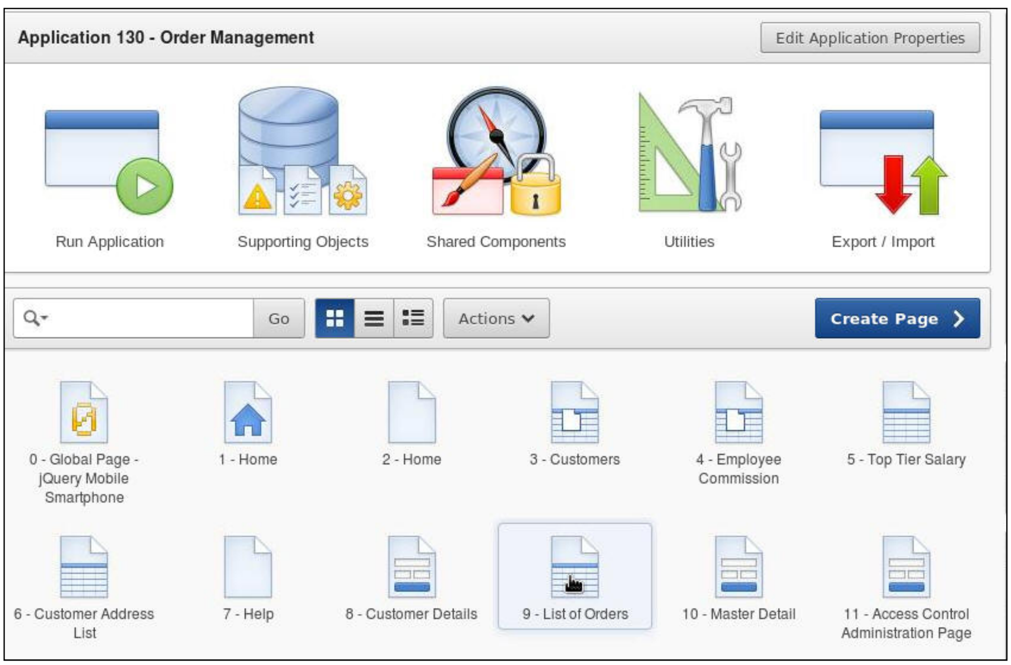
7) Click **Apply Changes**.



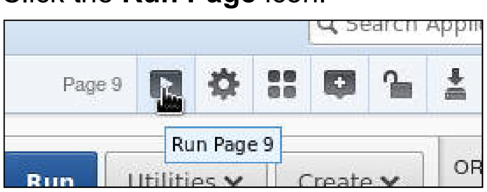
8) Click the **Application <n>** breadcrumb.



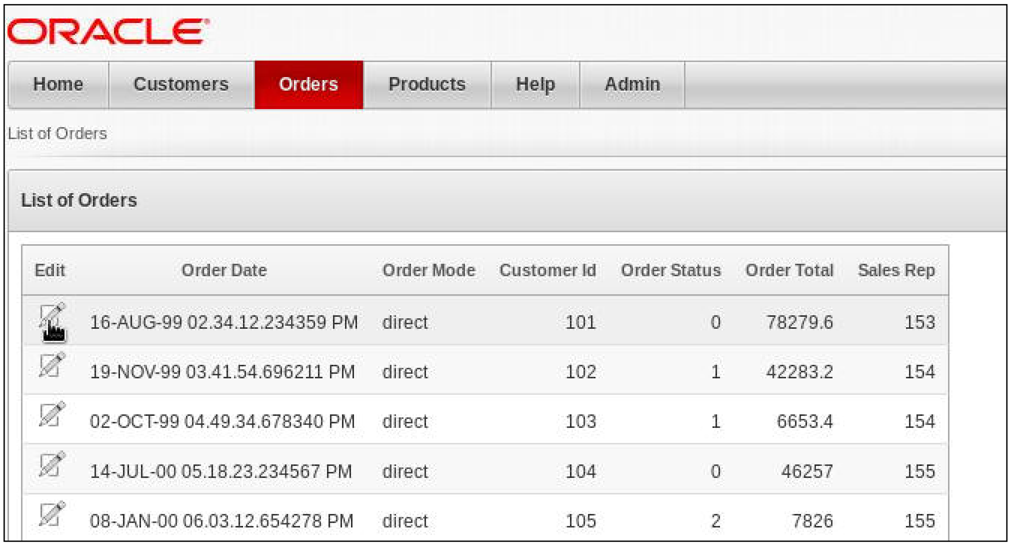
9) Select the **List of Orders** page.



10) Click the **Run Page** icon.



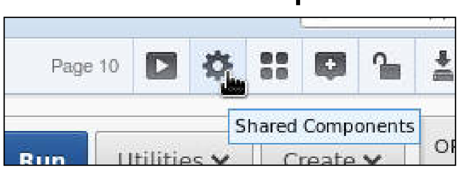
11) The List of Orders page is displayed. Click the Edit icon next to one of the Orders.



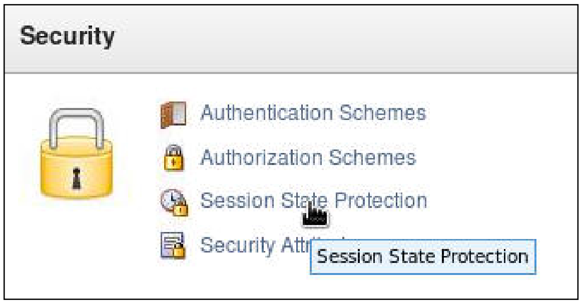
12) Notice that you receive an error indicating that the item has Internal Only protection. This is because it did not have a checksum. Click the **Edit Page** link on the Developer toolbar.

b. Set the Page Access Protection for the Master Detail page to **Arguments Must Have Checksum**. Run the page, edit an order to see the checksum that is passed to the Master Detail page.

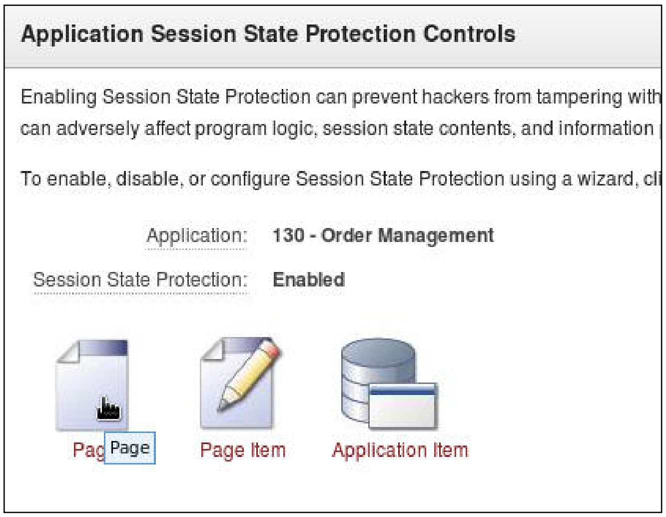
1) Click the **Shared Components** icon.



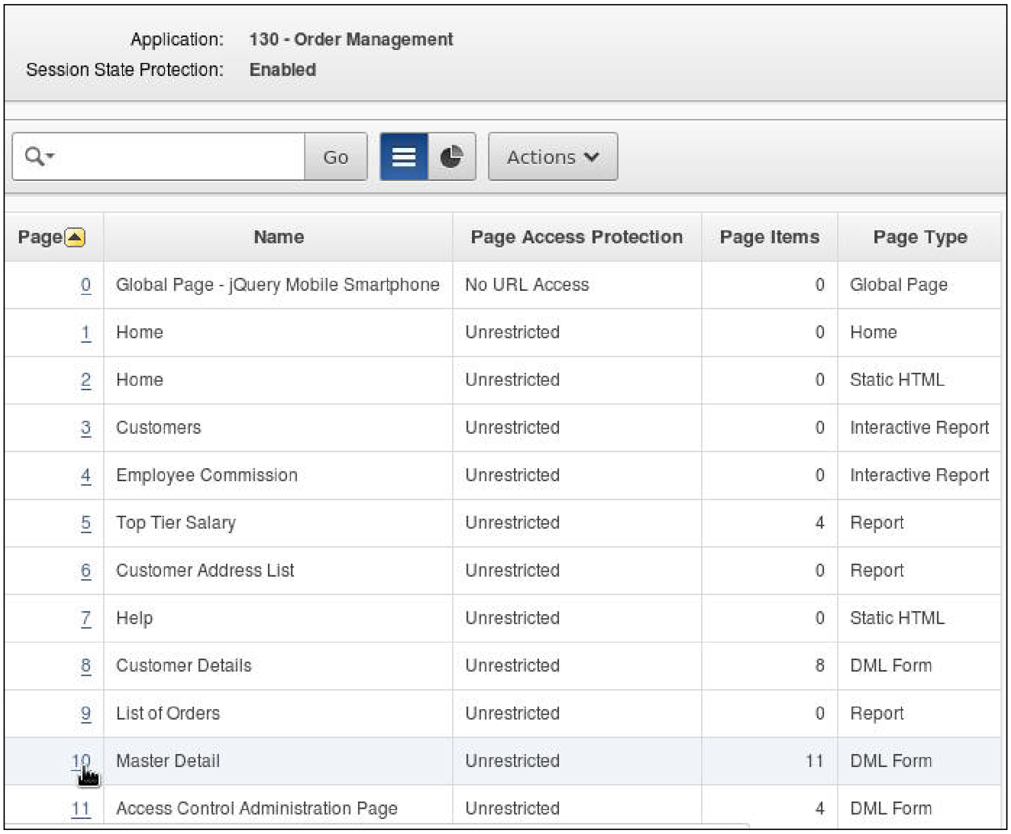
2) Under Security, select **Session State Protection**.



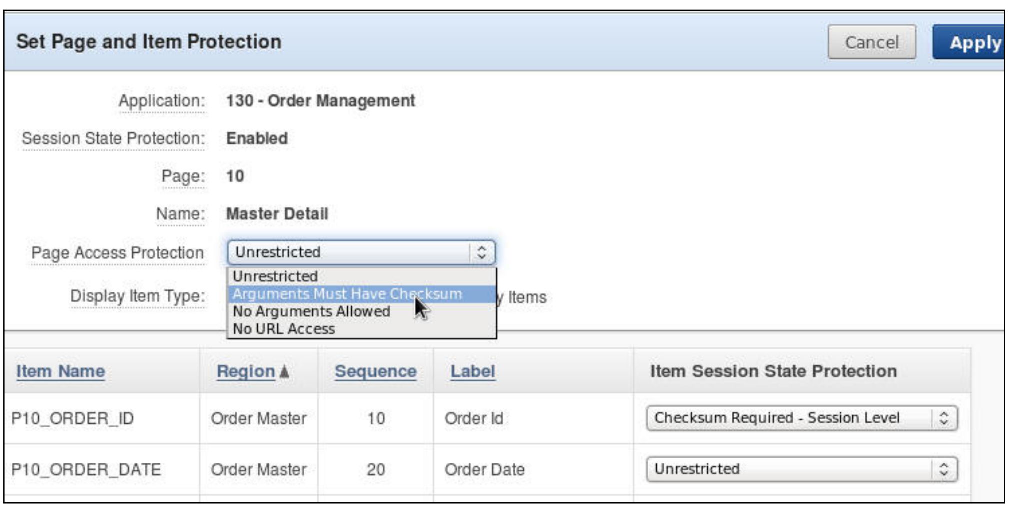
3) You want to set the Page Access Protection for the Master Detail page. Click **Page**.



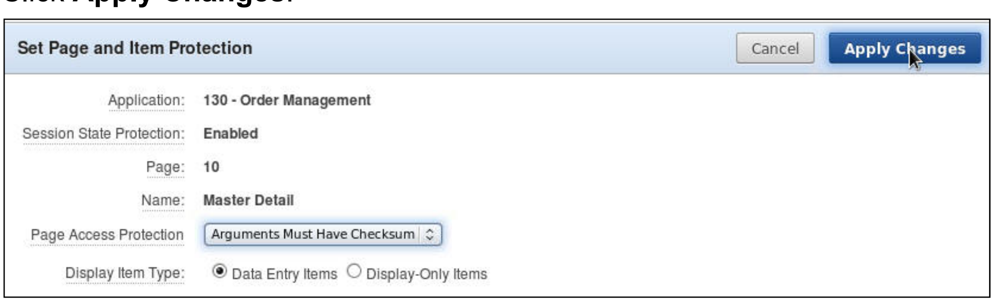
4) Click the Page link for the Master Detail page.



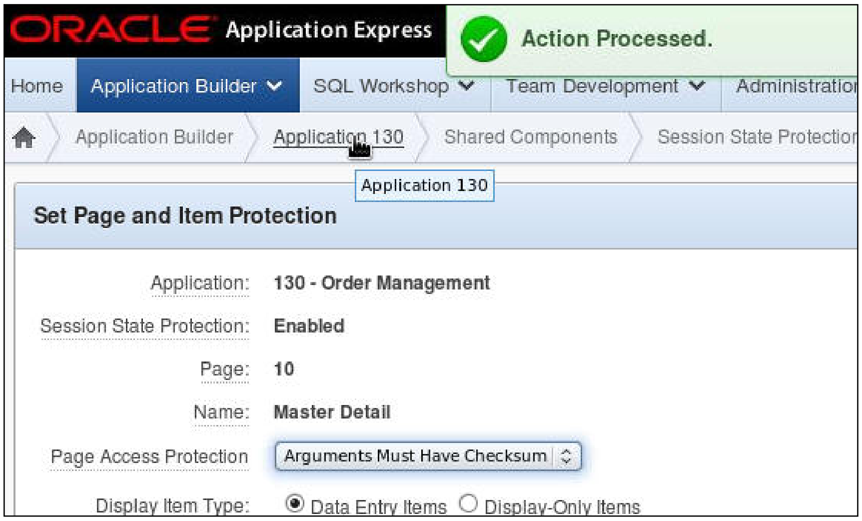
5) Select **Arguments Must Have Checksum** for Page Access Protection.



6) Click **Apply Changes**.



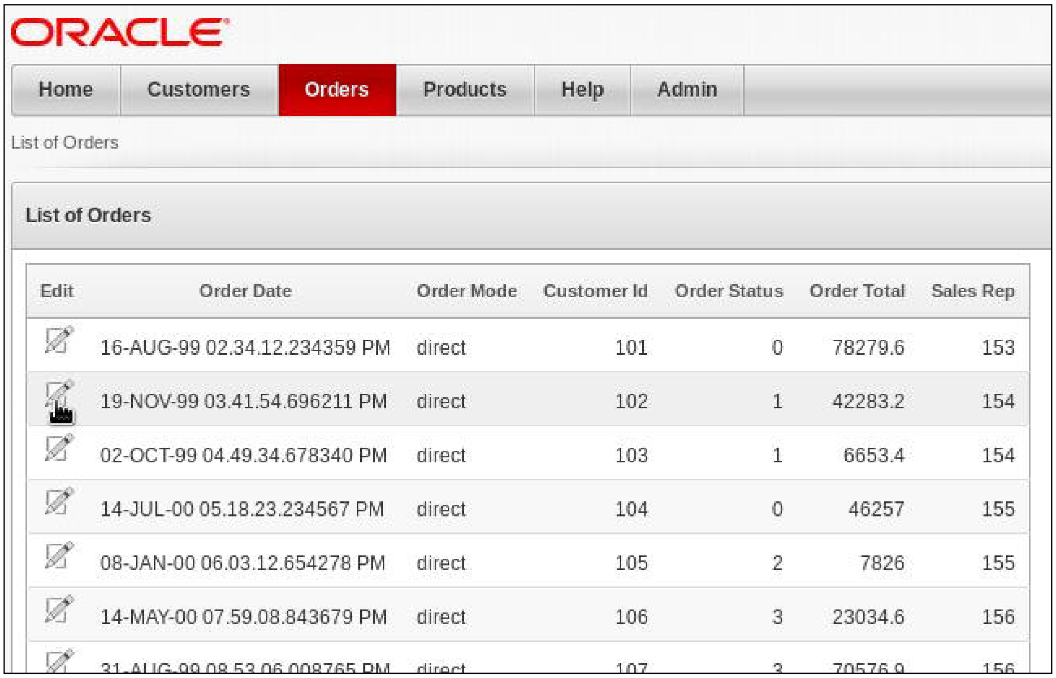
7) Click the **Application <*n*>** breadcrumb.



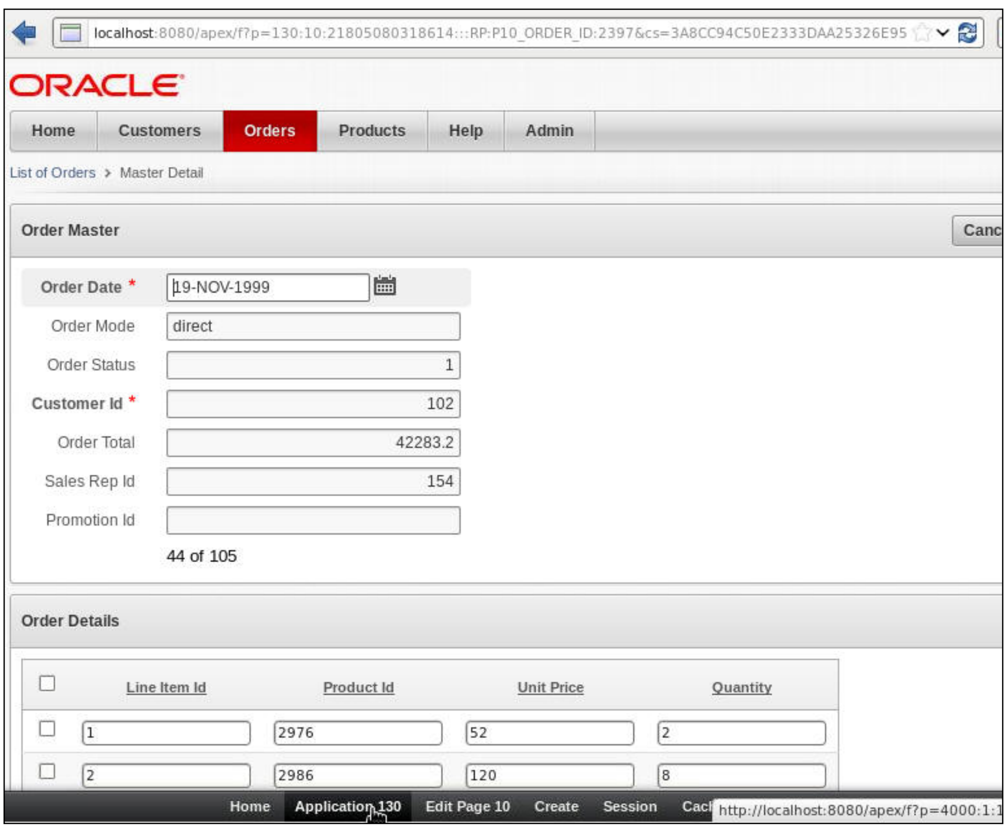
8) Click **Run Application**.

9) From the Home page, select **View Orders**.

10) Click the Edit icon next to one of the Orders.

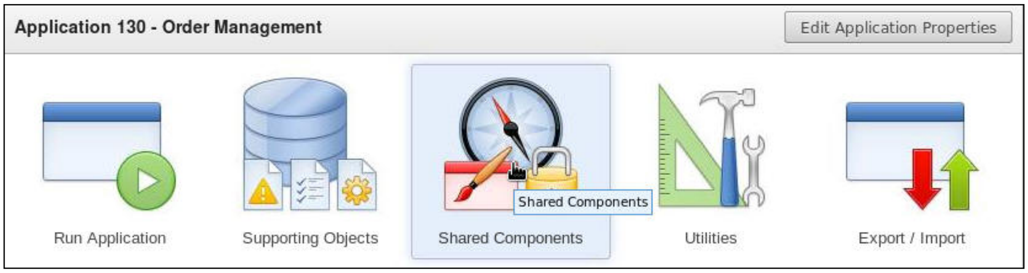


11) Notice that the Order is now displayed because there is a checksum in the URL. Click the **Application <*n*>** link on the Developer toolbar.

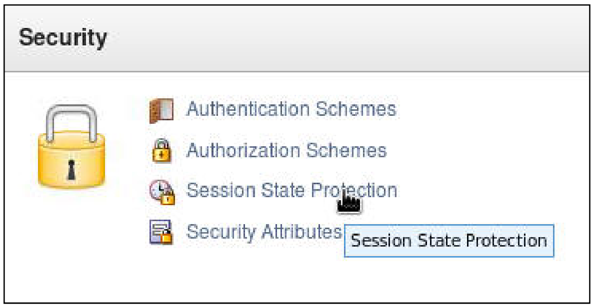


c. Set the Page Access Protection for the Employee Commission page to No URL Access.

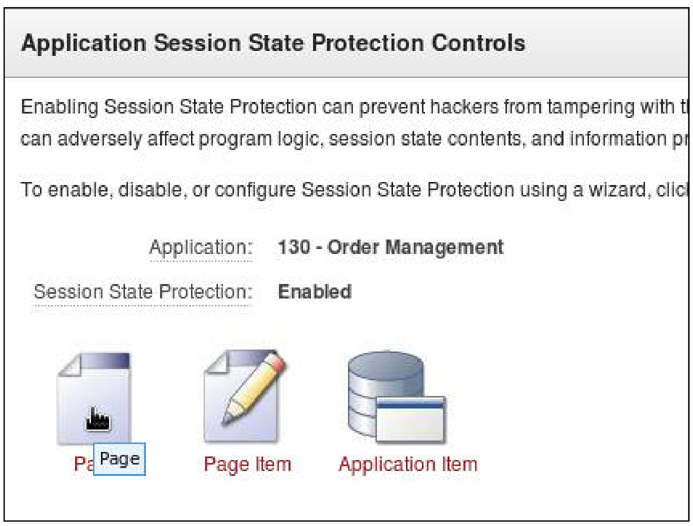
1) Click **Shared Components**.



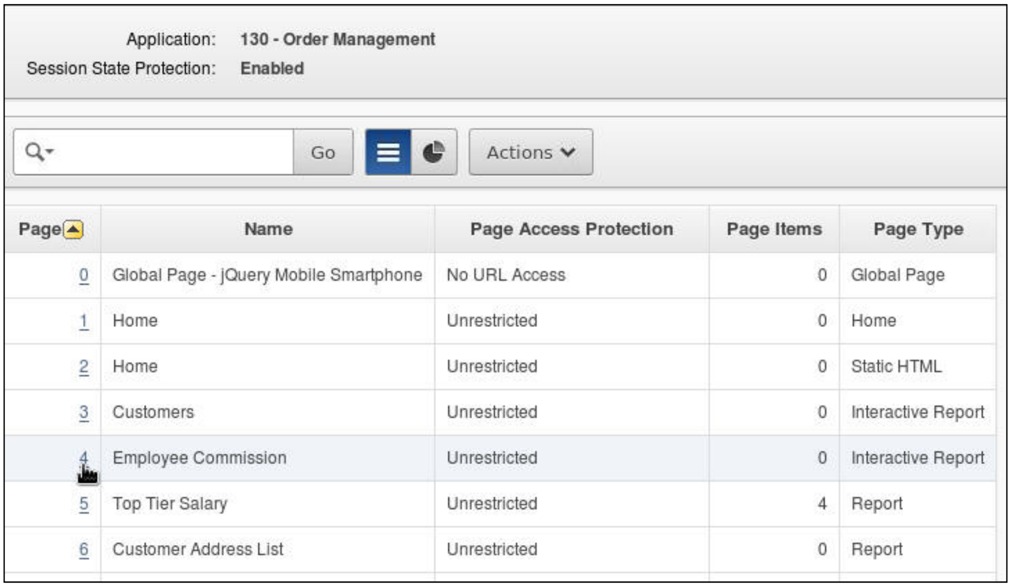
2) Under Security, select **Session State Protection**.



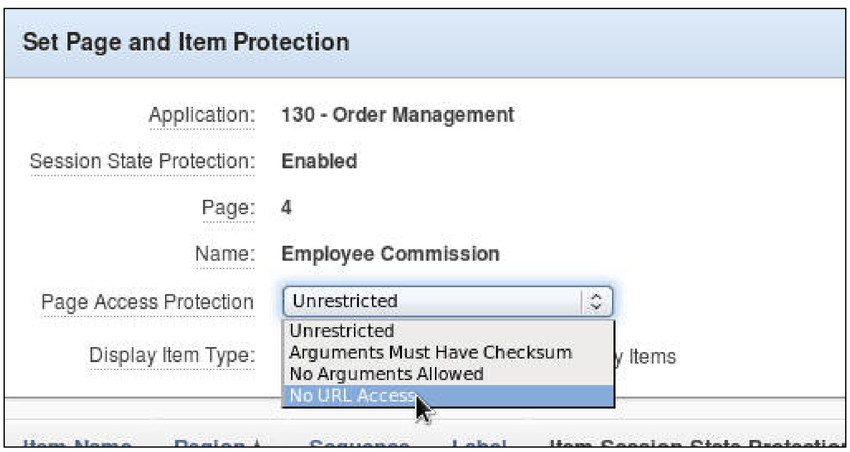
3) Click **Page**.



4) Click the Page link for the Employee Commission page.



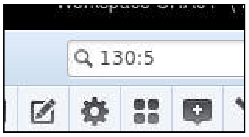
5) Select No URL Access for **Page Access Protection**.



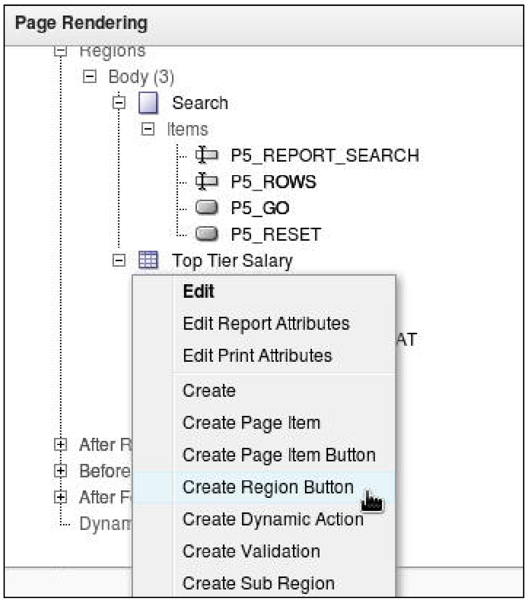
6) Click **Apply Changes**.

d. Add a Commission button to the Top Tier Salary page that redirects to the Employee Commission page. Run the page to see that there is an error because of the branch.

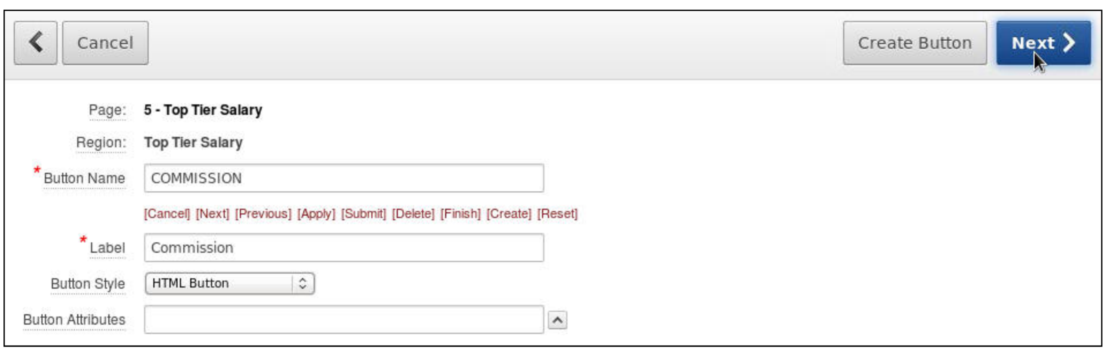
1) Search for the Employee Commission page in your Order Management application.



2) Right-click the **Top Tier Salary** Region and select **Create Region Button**.

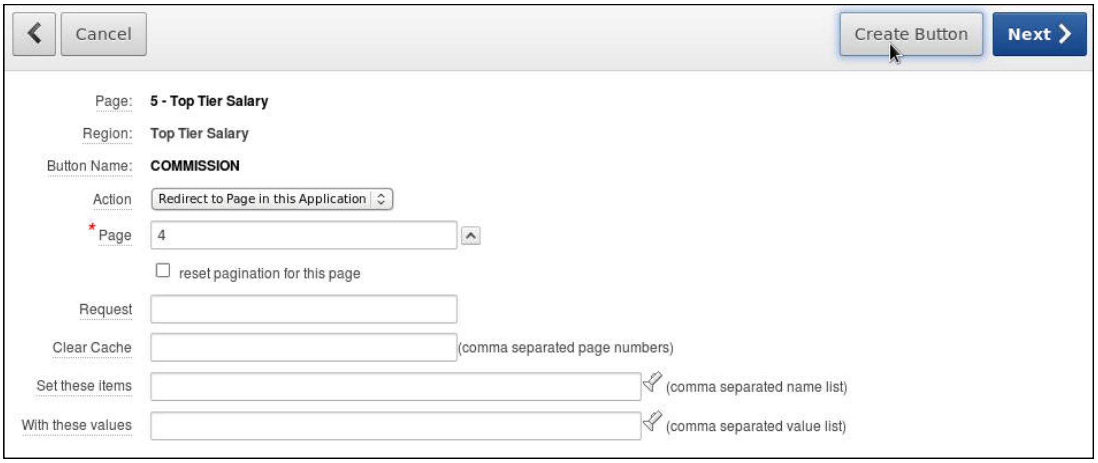


3) Enter **Commission** for Button Name and click **Next >**.

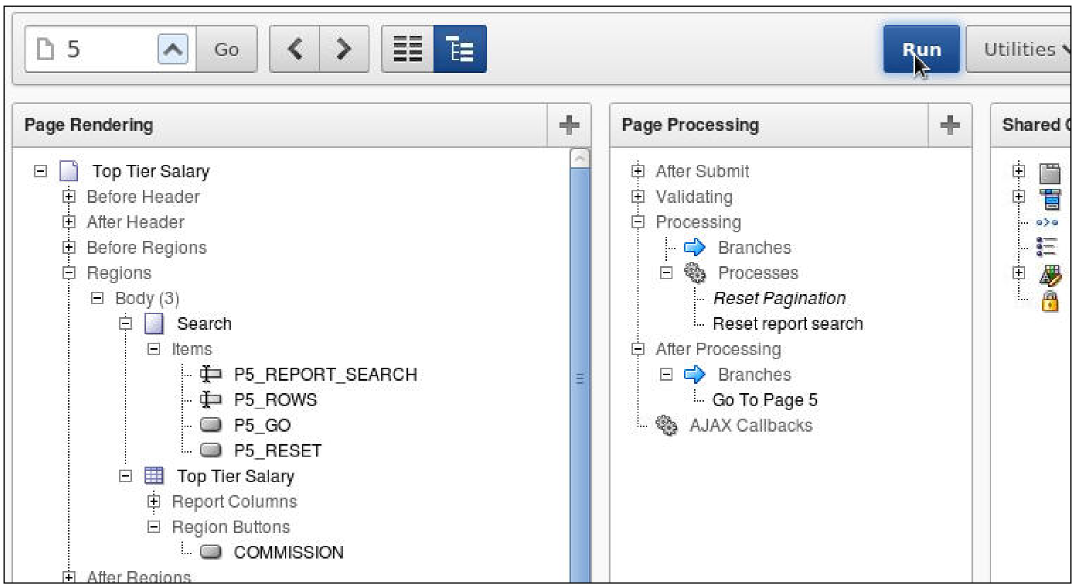


4) Accept the defaults and click **Next >**.

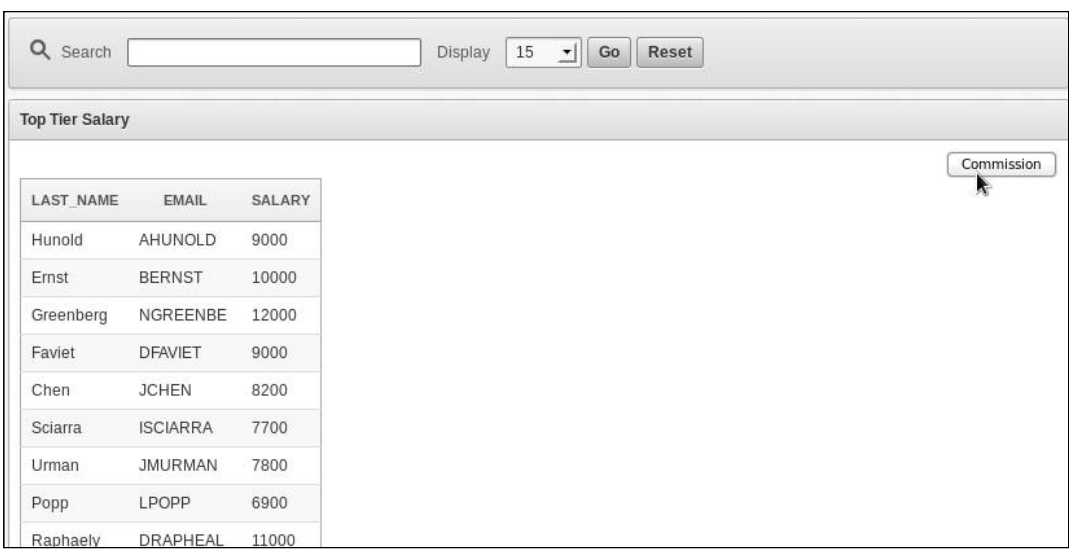
5) Select **Redirect to Page in this Application** for Action and enter the Employee Commission page for Page and click **Create Button**.



6) Click **Run**.



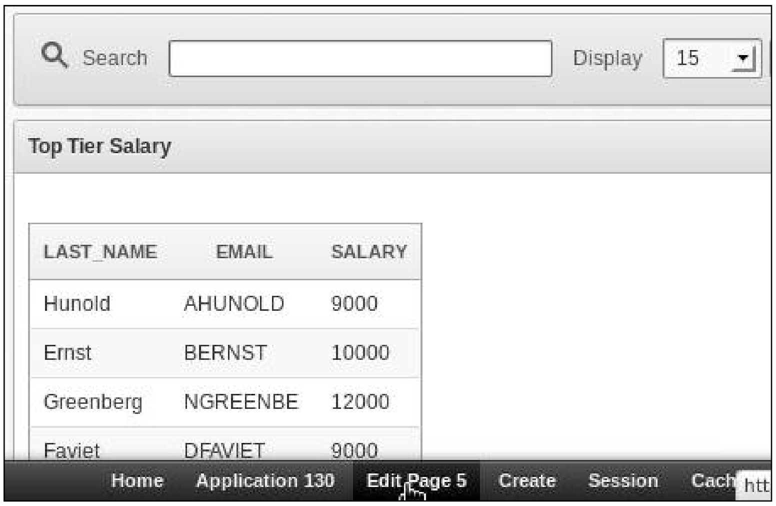
7) Click **Commission**.



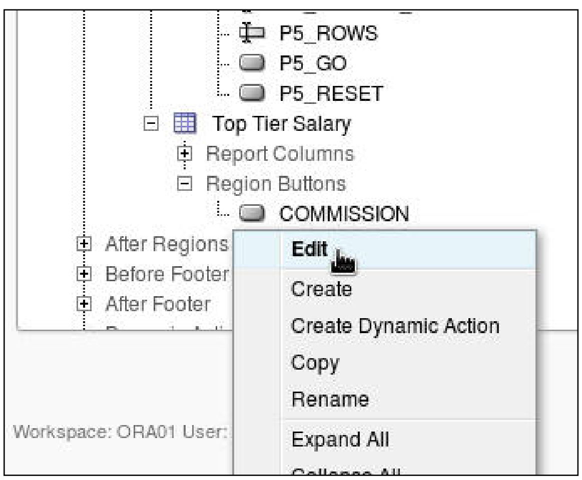
8) Notice you receive an error indicating that you must create a Branch to Page branch that generates a page without a URL. Click **OK**.

e. Edit the Commission button to submit the page and create a Branch to Page branch that will go to the Employee Commission page without passing the URL.

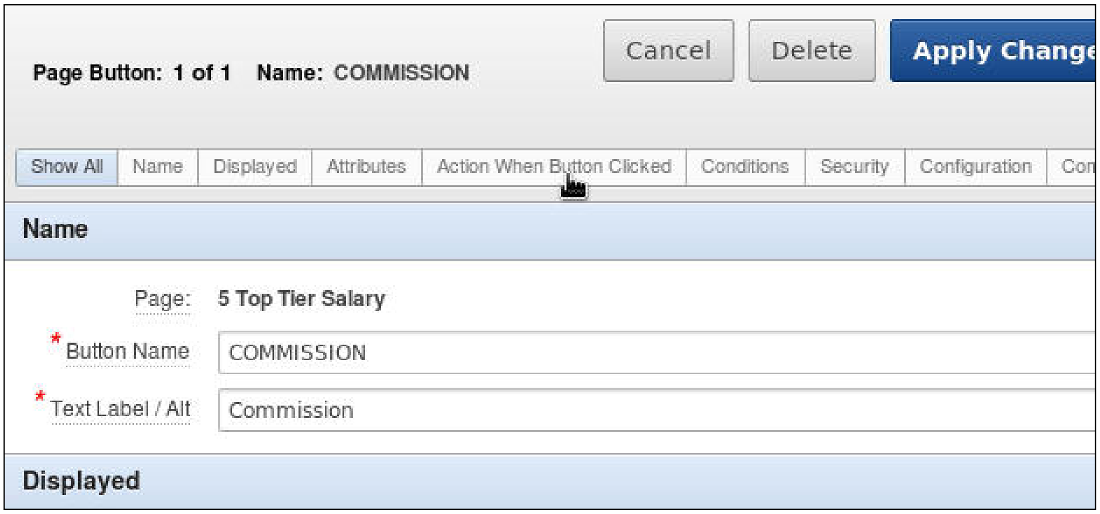
1) Click the **Edit Page** link on the Developer toolbar.



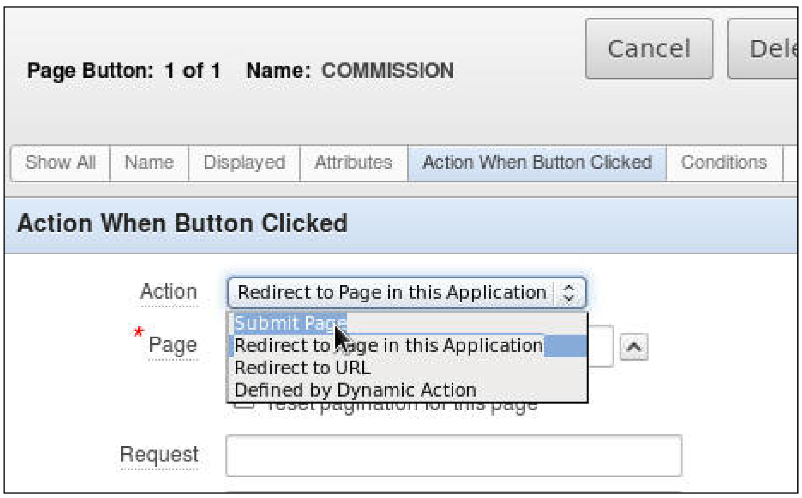
2) Right-click the Region Button **Commission** and select **Edit**.



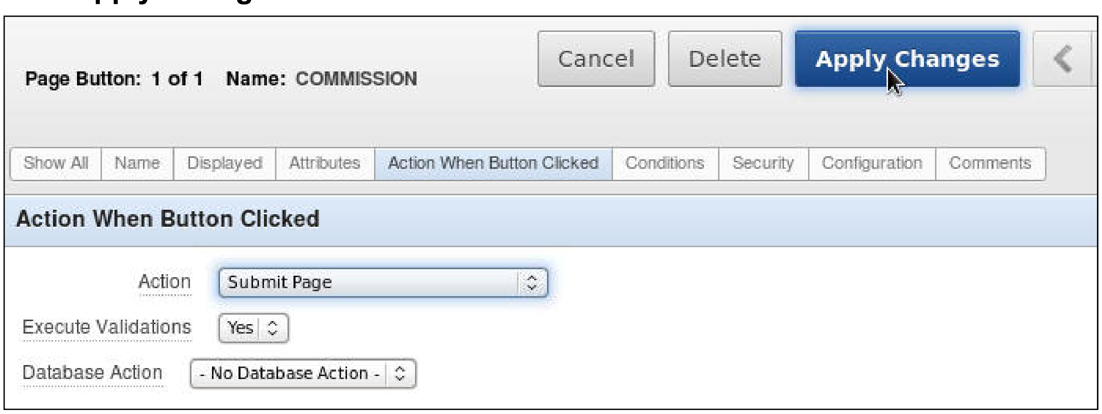
3) Click the **Action When Button Clicked** subtab.



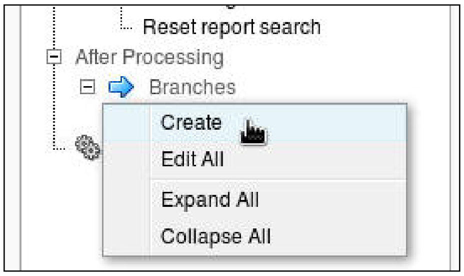
4) Select **Submit Page** for Action.



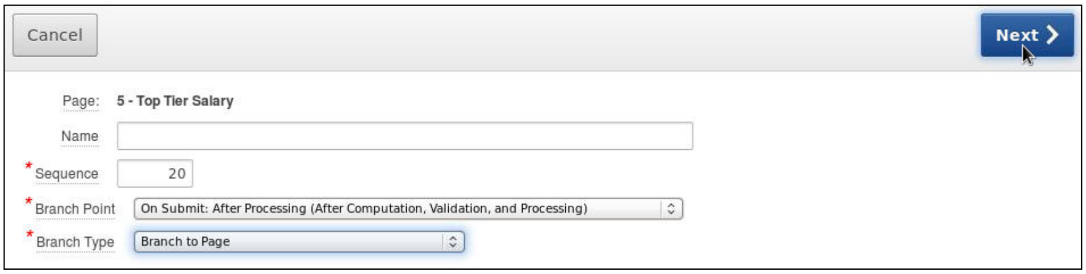
5) Click **Apply Changes**.



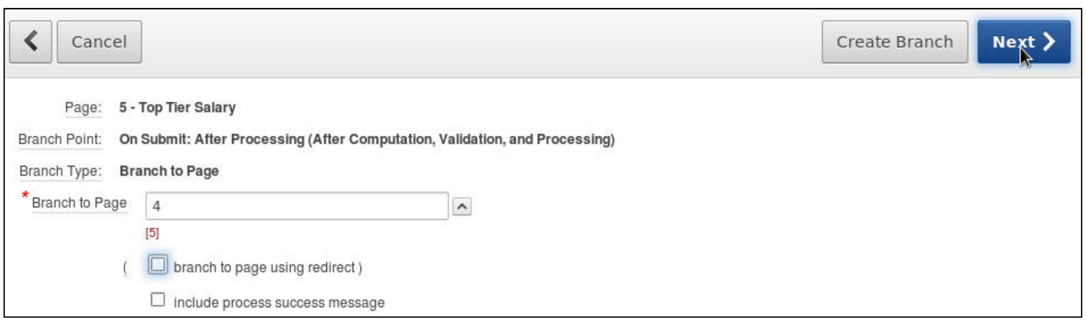
6) Now you need to create a Branch to Page branch. Under After Processing, right-click **Branches** and select **Create**.



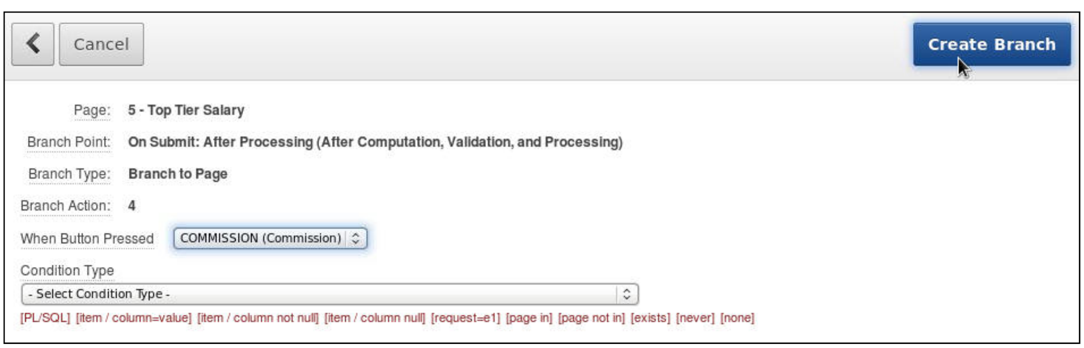
7) Select **Branch to Page** for Branch Type and click **Next >**.



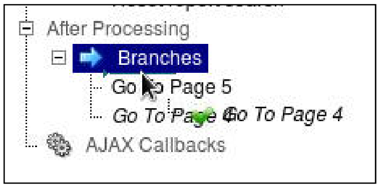
8) Select the Employee Commission page for Branch to Page and deselect the **branch to page using redirect** check box. Then click **Next >**.



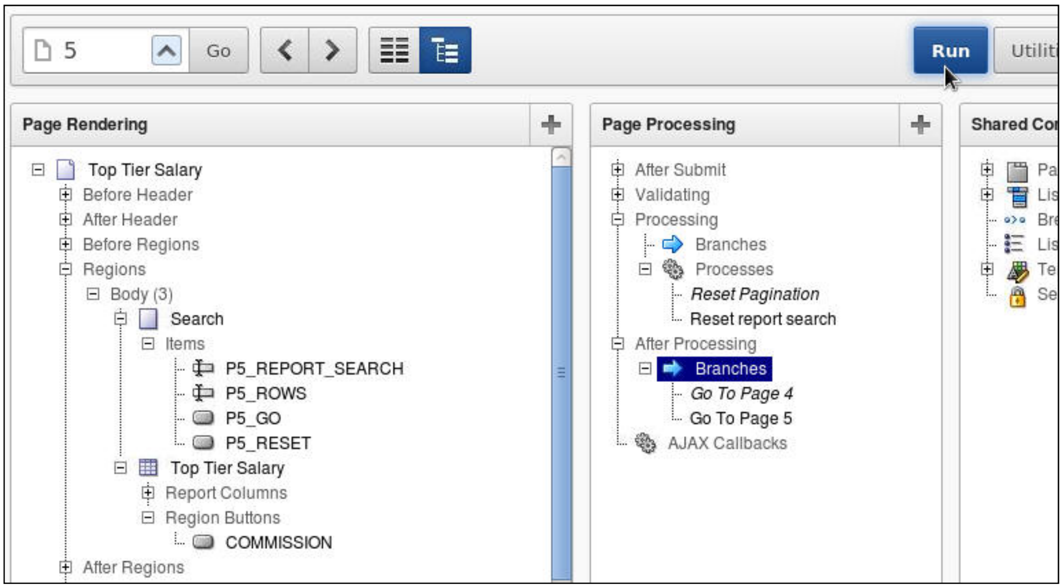
9) Select the **Commission** button for When Button Pressed and click **Create Branch**.



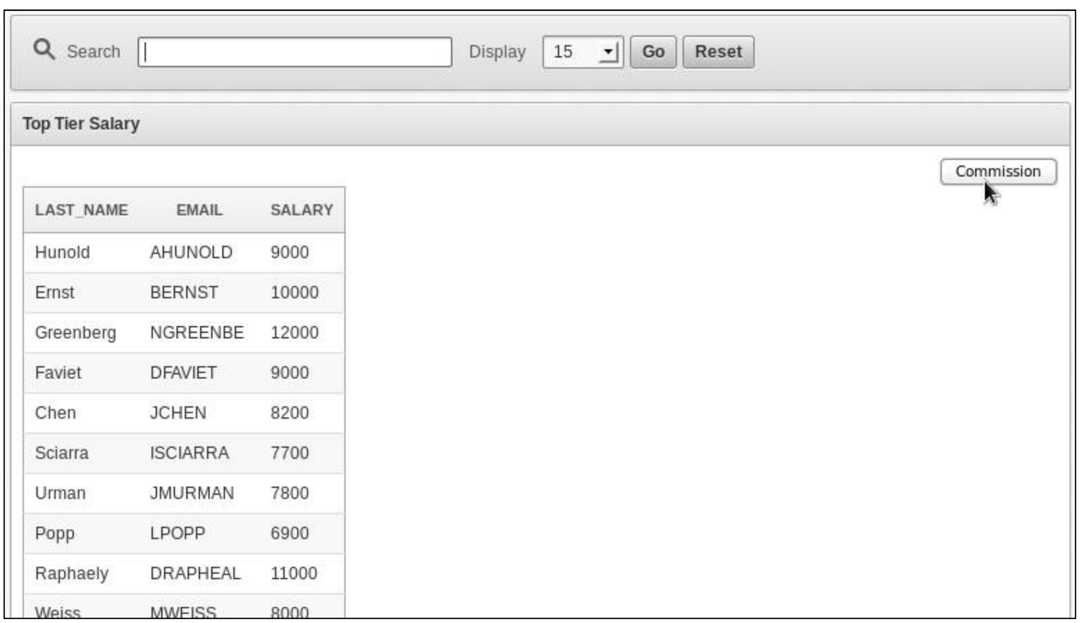
10) You want the branch you just created to be run before the one with no condition. Move the branch you just created before the Go to Page branch without a condition.



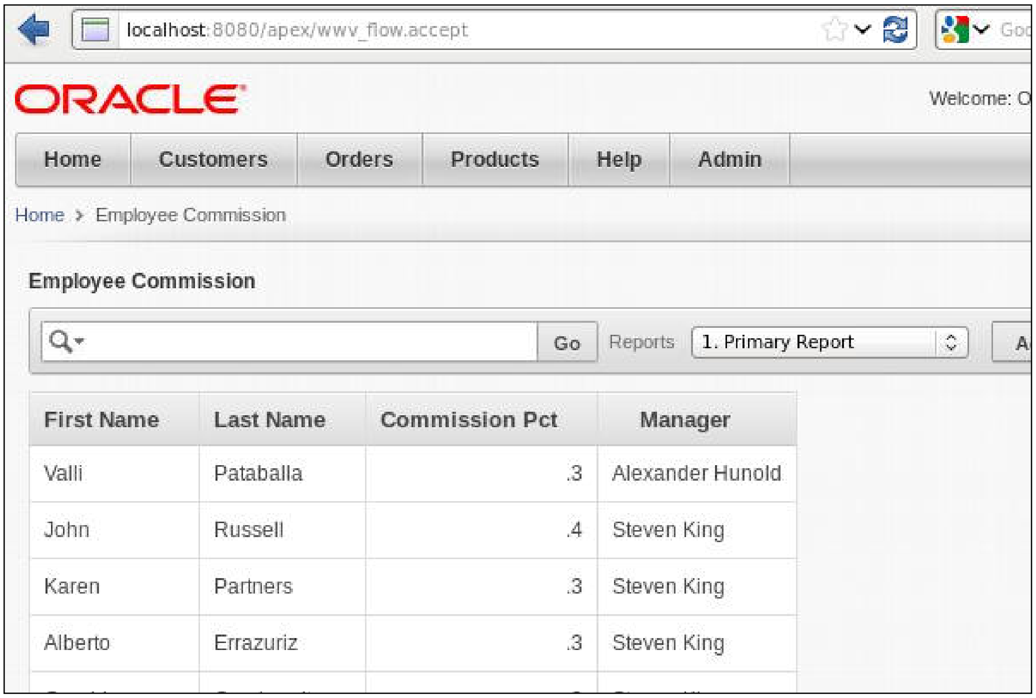
11) Click **Run**.



12) Click **Commission**.



13) The Employee Commission page is displayed. Review the URL. Notice that it no longer passes the f?p= in the URL and it is rendered successfully.



Practices for Lesson 15: Managing Application Navigation

Chapter 15

Practices for Lesson 15: Overview

Practices Overview

In these practices, you build a hierarchical list with images, a database-driven report, and a site map, and enforce authorization on the site map.

Practice 15-1: Building a Hierarchical List with Images

Overview

In this practice, you upload the images to the Home Page List that you have created in the previous practices, assign the Pull-down menu with the images template to the List, change the attributes of the Tasks region, and associate the region with each Parent List Entry.

Assumptions

You have performed the previous practices. You can import the `LAB142_SOLN.sql` packaged application.

Tasks

a. Load the images that your Hierarchical List will use into the APEX Repository. Upload the following images from your `/home/oracle/labs/files` directory:

File Name
customers.png
products.png
orders.png

b. Change the Home Page List to the following list structure.

Parent List Entry	Sequence	List Entry Label	Page Number
	10	Manage Customers	
Manage Customers	12	View Customers	3
	20	Manage Products	
Manage Products	22	View Products	17
	30	Manage Orders	
Manage Orders	32	View Orders	11

c. In your Navigation region on the Home page, change the template to No Template and the List Template to Pull Down Menu with Image**.**

d. Run the page to see the Home Page List. Click the arrow next to each image to show the sublist entries.

Practice 15-2: Building a Database-Driven Report

Overview

In this practice, you build a report based on the data in a table, and then navigate to the detail. You also change the template to the Navigation Region template.

Assumptions

You have performed the previous practices.

Tasks

a. Create a new page with a Classic Report called **List of Customers**.

b. Hide the `CUSTOMER_ID` column in the report and turn off pagination. If you want to display the Customer Name column heading in blue, edit the column definition and enter the following for Column Heading.

```
<font color="blue">CUSTOMERS</font>
```

c. Create a link to the Customer Details page on the `NAME` column. Pass the **`P<n>`**`_CUSTOMER_ID` item from the Customer Details page with the value `#CUSTOMER_ID#` from this page.

d. Run the page to see the database-driven report.

Practice 15-3: Building a Site Map

Overview

In this practice, you build a site map page that will be added as a navigation bar entry.

Assumptions

You have performed the previous practices.

Tasks

a. Assign each page that you want in the site map to the same page group. Create a Page Group called Site Map and assign the following pages to the Site Map page group.

Customers
Products
List of Orders
Customer Feedback
Access Control Administration Page

b. Create a new page called Site Map with a Classic Report that shows only the pages in the Site Map page group.

c. Change the Pagination Scheme to None and do not show the `PAGE_ID` column. Link the `PAGE_NAME` column to the `#PAGE_ID#` so that it opens the corresponding page when clicked.

d. Add this page to your navigation bar and call it Site Map.

e. Run the page and click one of the page names to view the corresponding page.

Practice 15-4: Enforcing Authorization in Site Map

Overview

In this practice, you add a function that determines whether you are authorized to see a particular page in your site map. You also change the SQL Report query for the Site Map to make sure that the page is selected only if you are authorized to see it.

Assumptions

You have performed the previous practices.

Tasks

a. Navigate to **SQL Workshop > SQL Commands**. Copy and paste the following code (located in the `/home/oracle/labs/files/lab15_4_1.txt` file) to create the `authorization_check` function. Click Run.

```
create or replace function authorization_check(
  p_scheme in varchar2)
return varchar2
is
begin
    if apex_util.public_check_authorization(p_scheme) then
        return 'true';
    else
        return 'false';
    end if;
end;
```

b. For the Site Map, change the Source for your Classic Report to the following code (located in the `/home/oracle/labs/files/lab15_4_2.txt` file), so that only the pages that are authorized will be shown based on the user logged in.

```
select distinct PAGE_ID, PAGE_NAME
from APEX_APPLICATION_PAGES p, apex_application_authorization a
where p.APPLICATION_ID = :APP_ID
and a.application_id = p.application_id
and (p.authorization_scheme is null or
  (a.authorization_scheme_id = p.authorization_scheme_id and
     authorization_check(a.authorization_scheme_name) = 'true'))
and PAGE_GROUP = 'Site Map'
```

c. You do not want end users or developers to see the Access Control Administration page. Make sure that the authorization schemes for this page is set to **access control - Administrator**. Run the page to see if only an authorized user can view it.

Solution 15-1: Building a Hierarchical List with Images

Overview

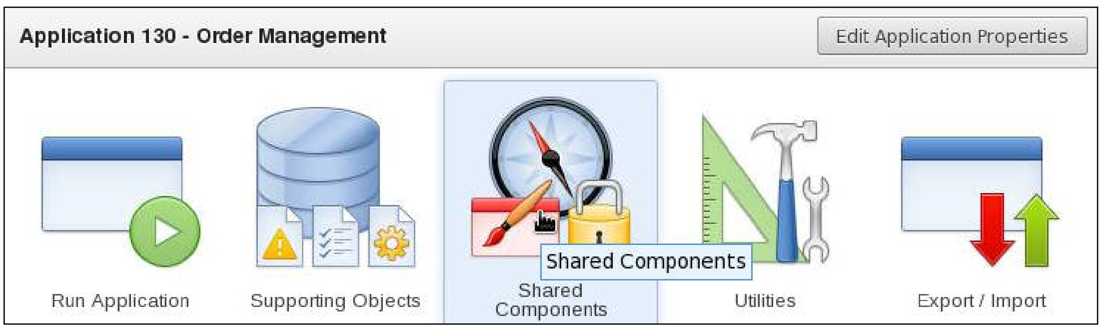
In this solution, you upload the images to the Home Page List that you have created in the previous practices, assign the Pull-down menu with the images template to the List, change the attributes of the Tasks region, and associate the region with each Parent List Entry.

Tasks

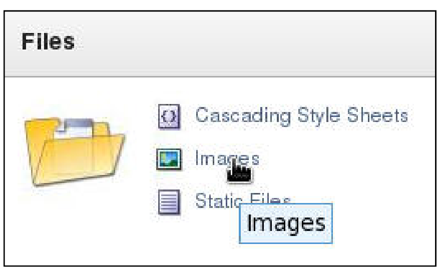
a. Load the images that your Hierarchical List will use into the APEX Repository. Upload the following images from your `/home/oracle/labs/files` directory:

File Name
`customers.png`
`products.png`
`orders.png`

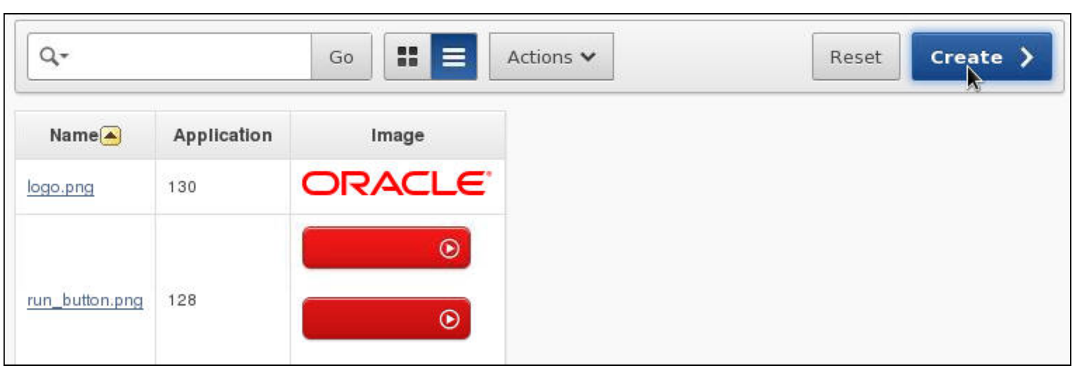
1) Navigate to **Shared Components** for Order Management application.



2) Select **Files > Images**.



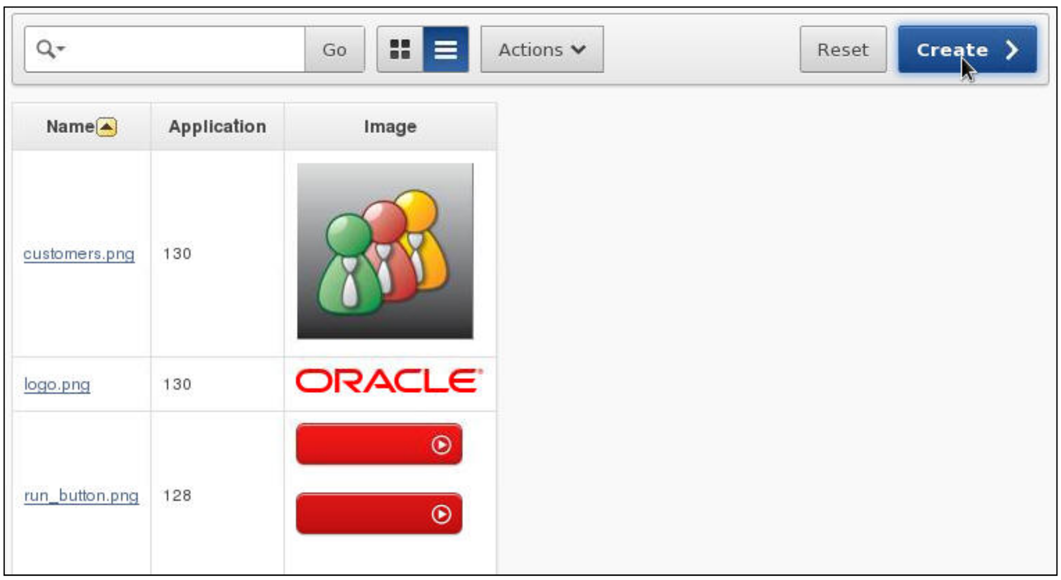
3) Click **Create >**.



4) Select your application name for **Application** and click **Browse**.

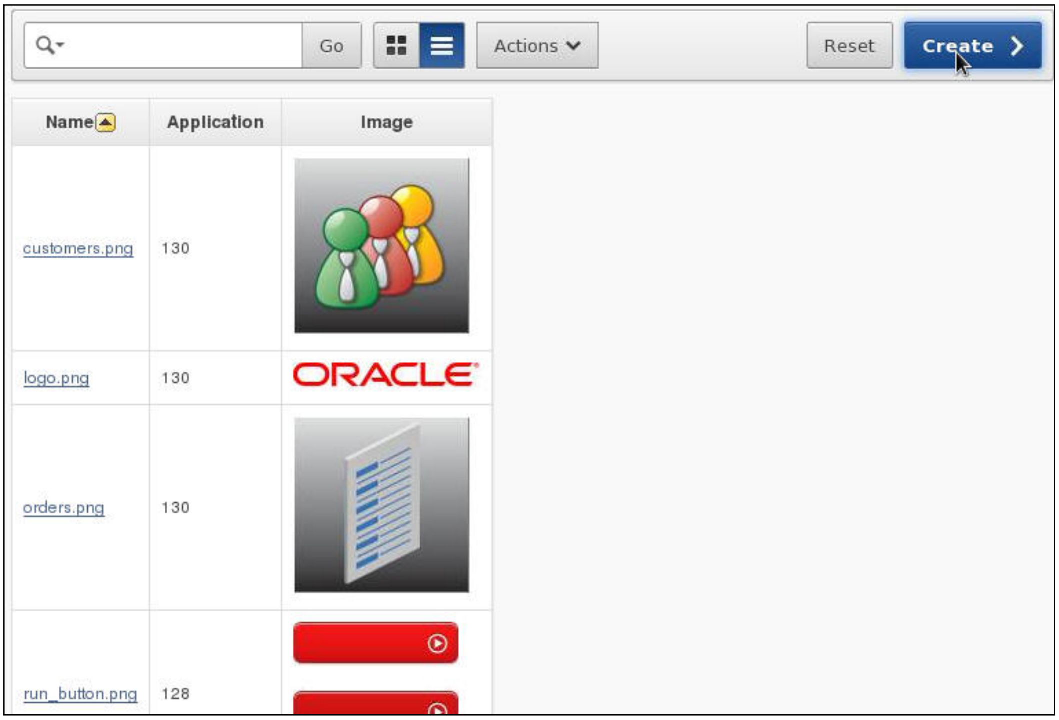
5) Select **customers.png** from the **`home/oracle/labs/files`** folder and click **Open**. Click **Upload**.

6) Click **Create**.



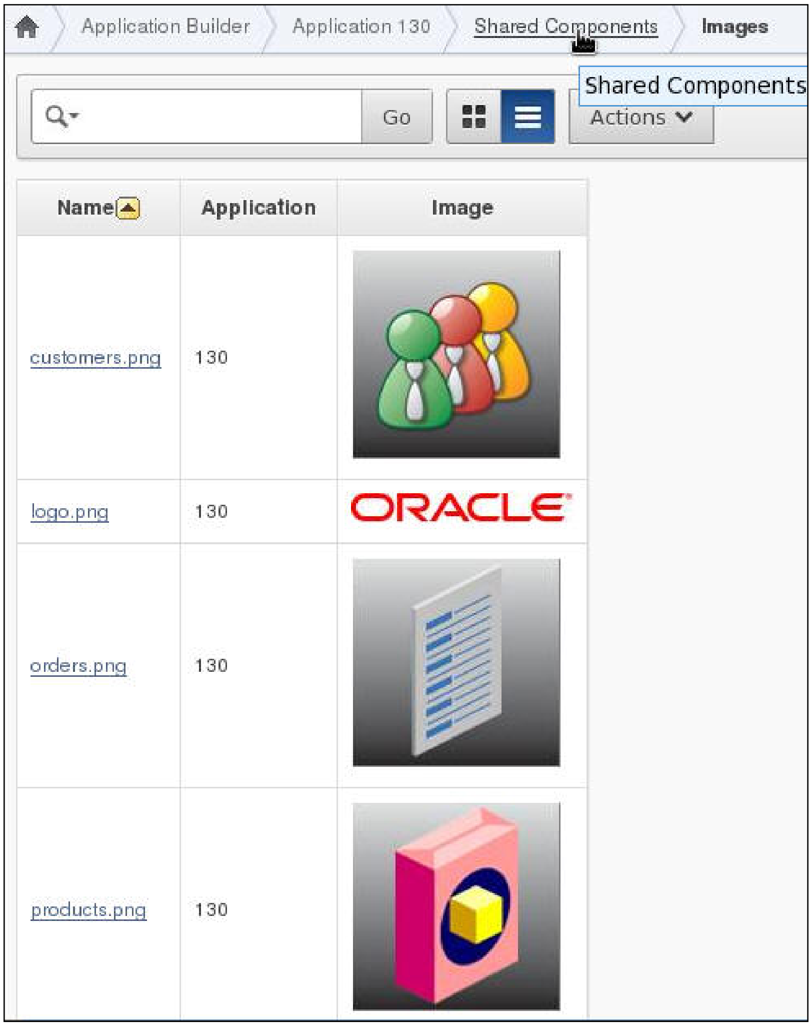
Select your application name for **Application** and click **Browse**. Select **orders.png** from the **`home/oracle/labs/files`** folder and click **Open**. Click **Upload**.

7) Click **Create >**.



8) Select your application name for **Application** and click **Browse**. Select **products.png** from the `home/oracle/labs/files` folder and click **Open**. Click **Upload**.

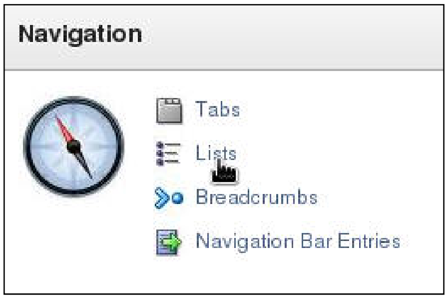
9) You want to add the images to the Home Page List. Click the **Shared Components** breadcrumb.



b. Change the Home Page List to the following list structure.

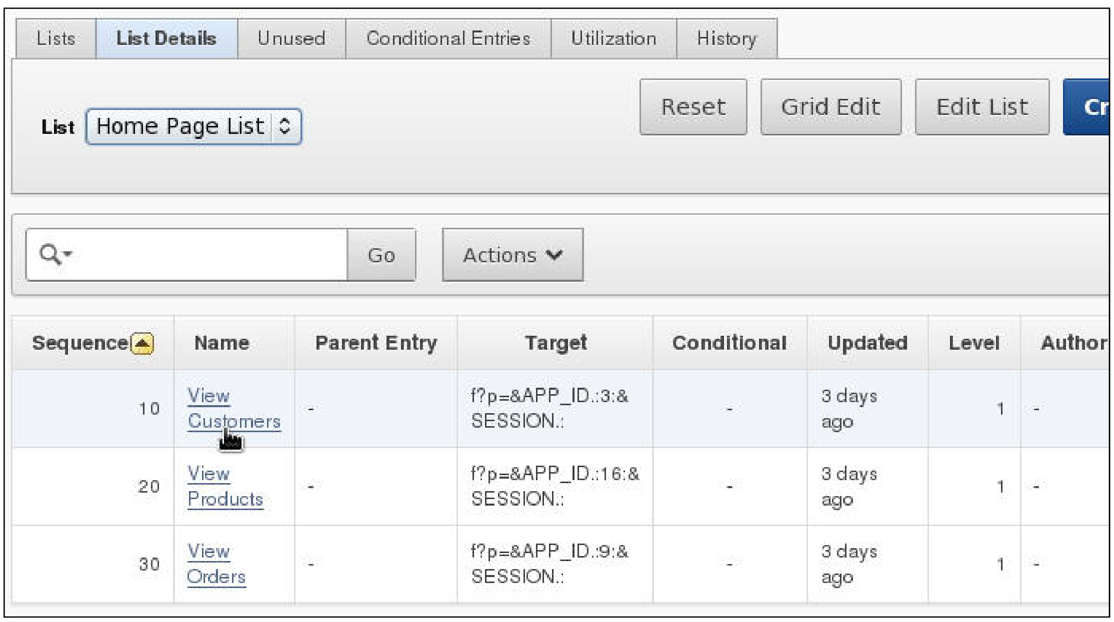
Parent List Entry	Sequence	List Entry Label	Page Number
	10	Manage Customers	
Manage Customers	12	View Customers	3
	20	Manage Products	
Manage Products	22	View Products	17
	30	Manage Orders	
Manage Orders	32	View Orders	11

1) Select **Navigation > Lists**.

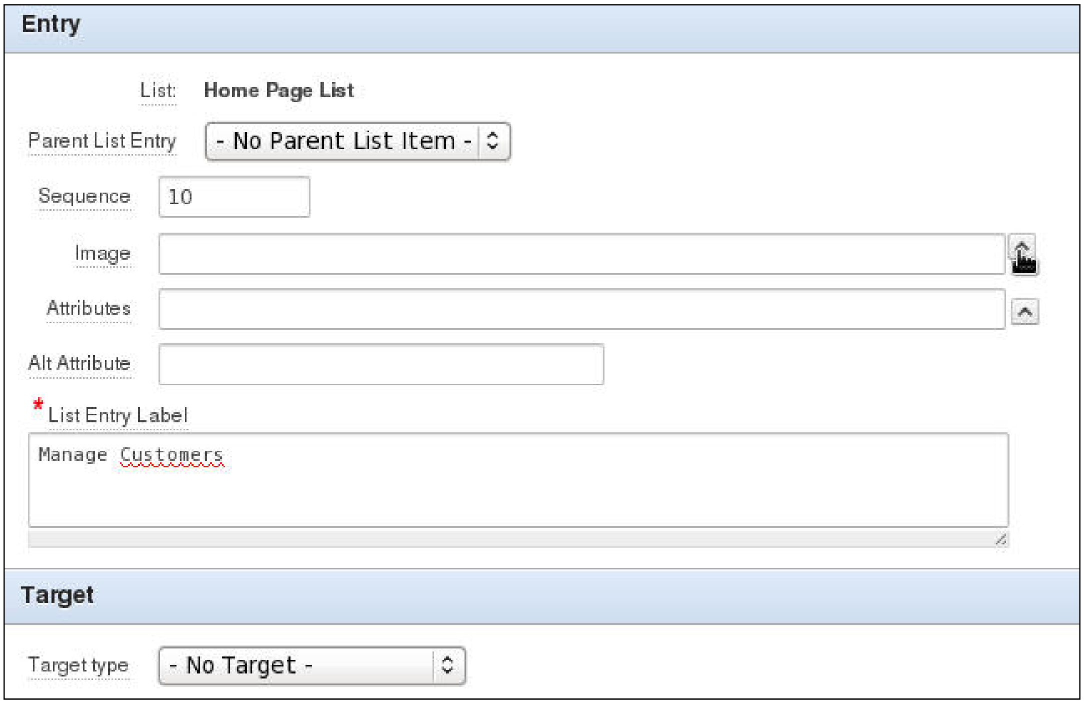


2) Select **Home Page List**.

3) Click **View Customers** to edit this List Entry detail.



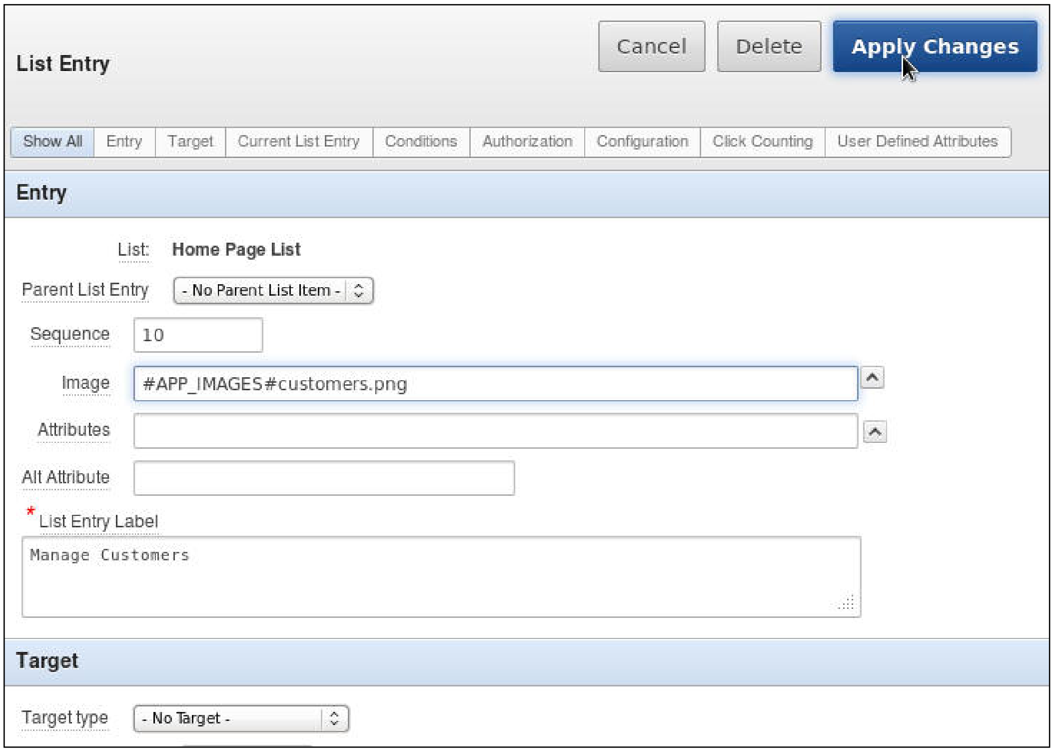
4) In the Entry section, change View Customers to **Manage Customers** for List Entry Label. In the Target section, change the Target type to **No Target** and click the up arrow for Image.



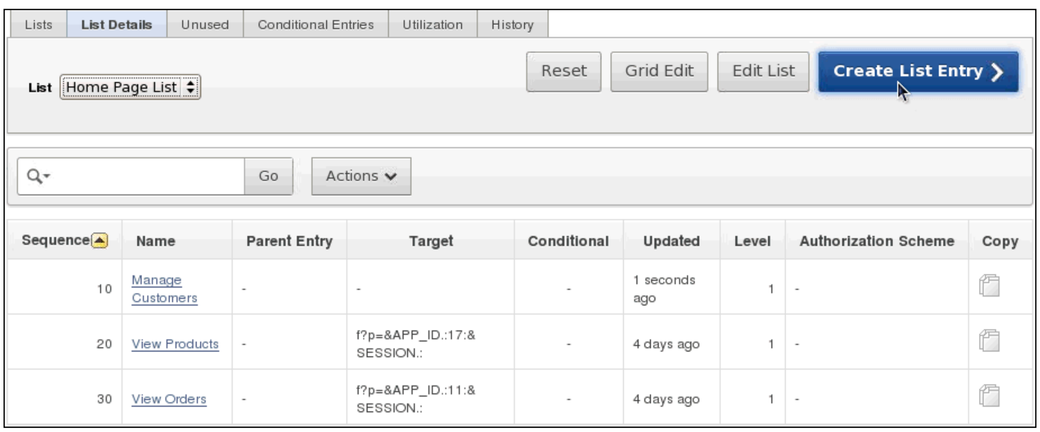
5) Select **Application Images** from the Show drop-down list and click **Go**.

6) Select the first image in the list.

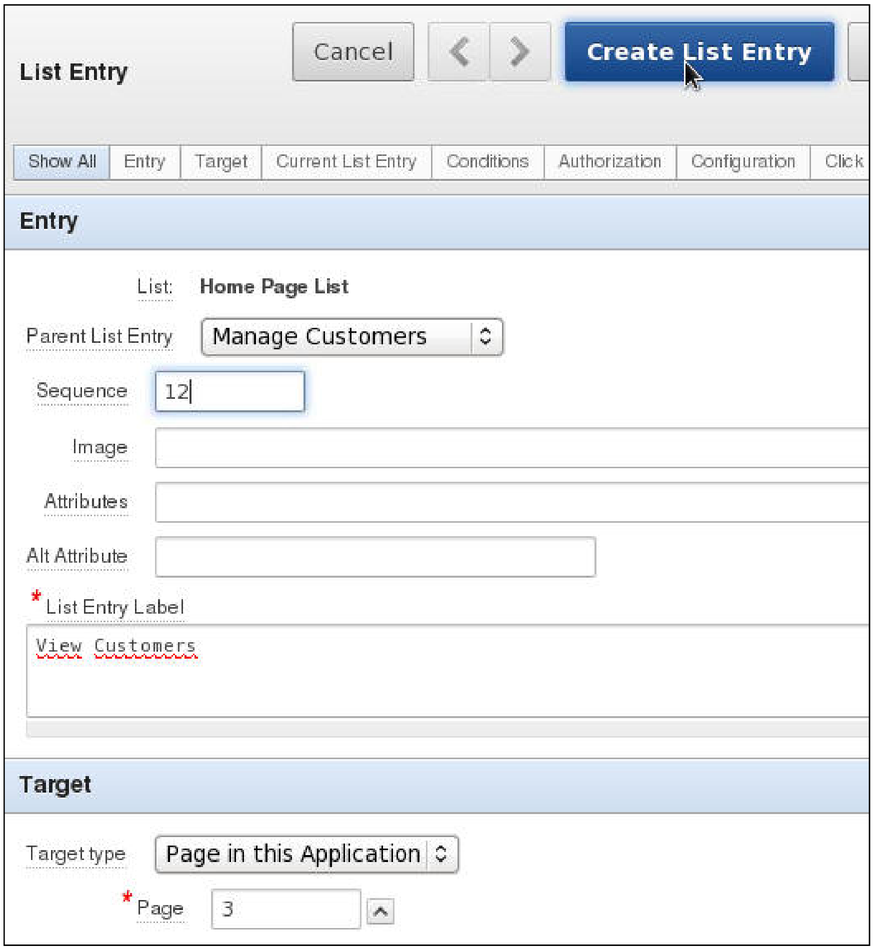
7) Click Apply Changes.



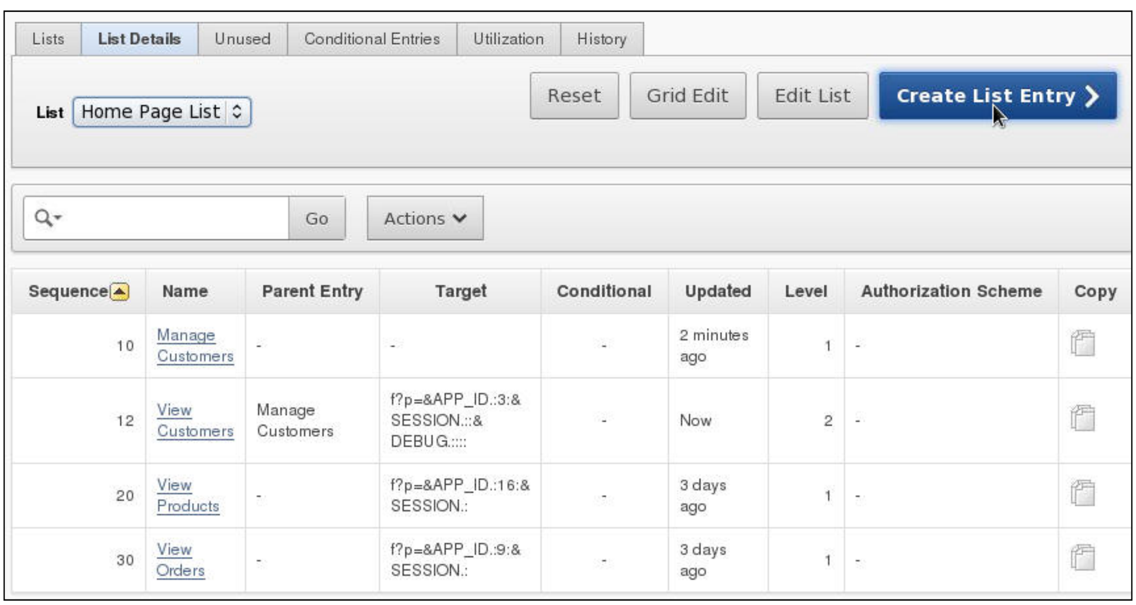
8) Click **Create List Entry**.



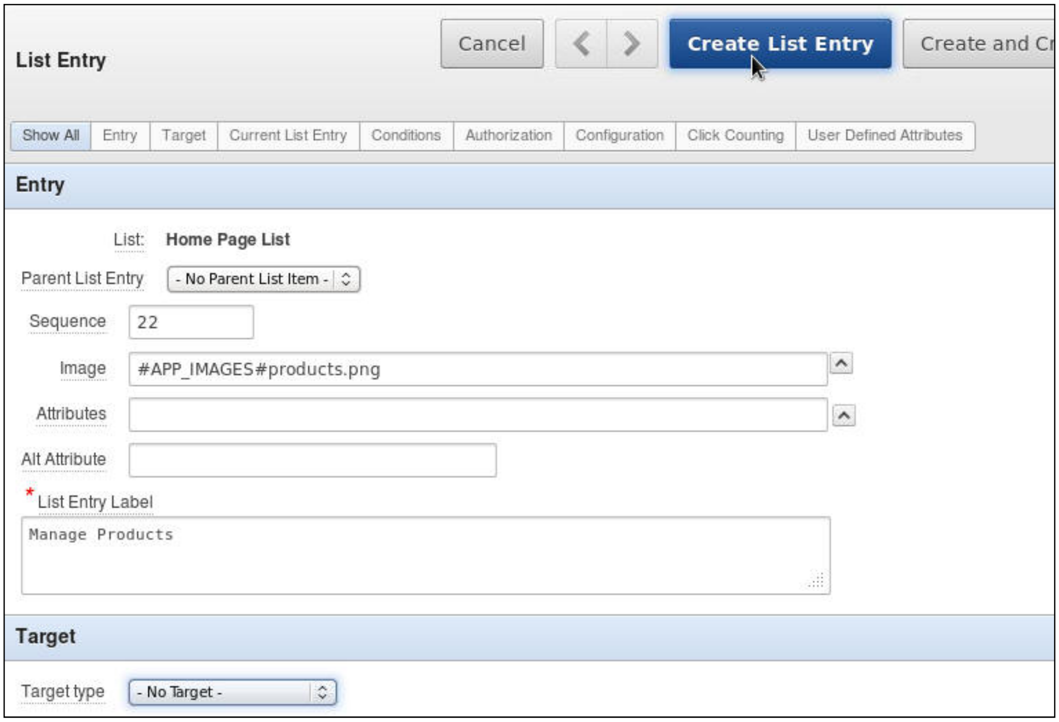
9) Select **Manage Customers** for Parent List Entry, enter **12** for Sequence, and enter **View Customers** for List Entry Label. In the Target section, make sure that Target Type is set to **Page in this Application** and the Page is **3** (the page number corresponding to the Customers page). Then click **Create List Entry**.



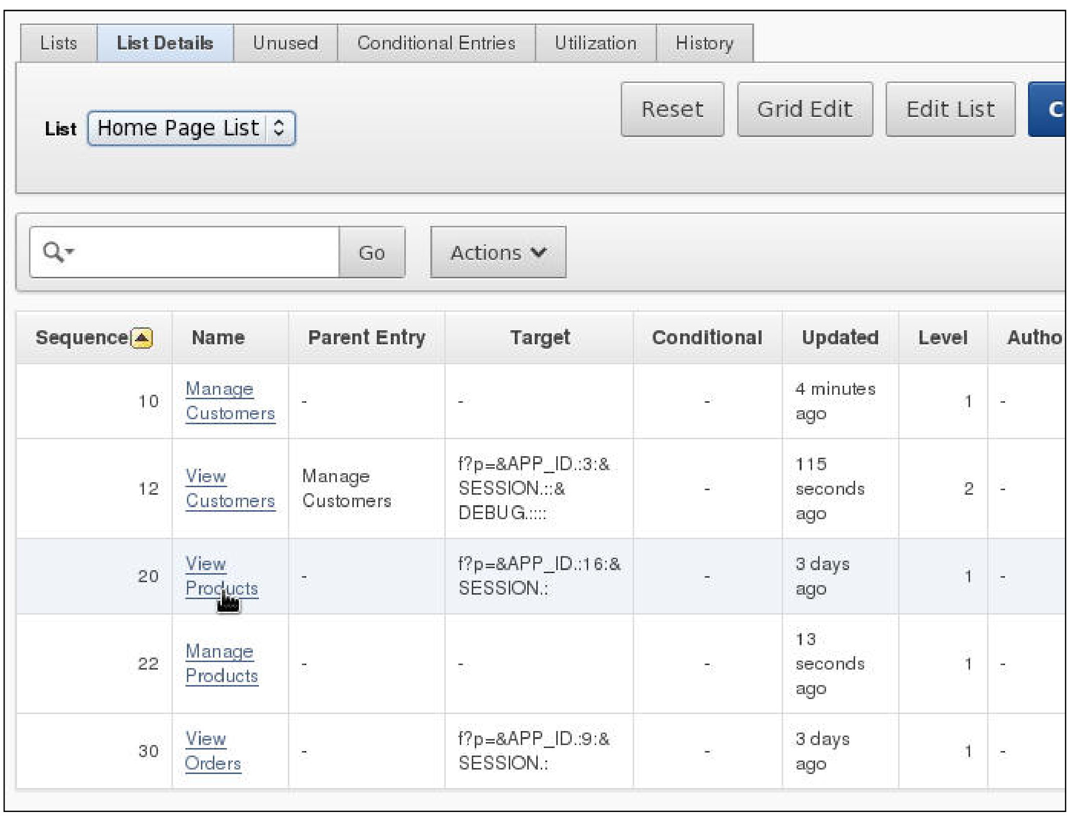
10) There is an easier way to modify your entries. Click **Create List Entry**.



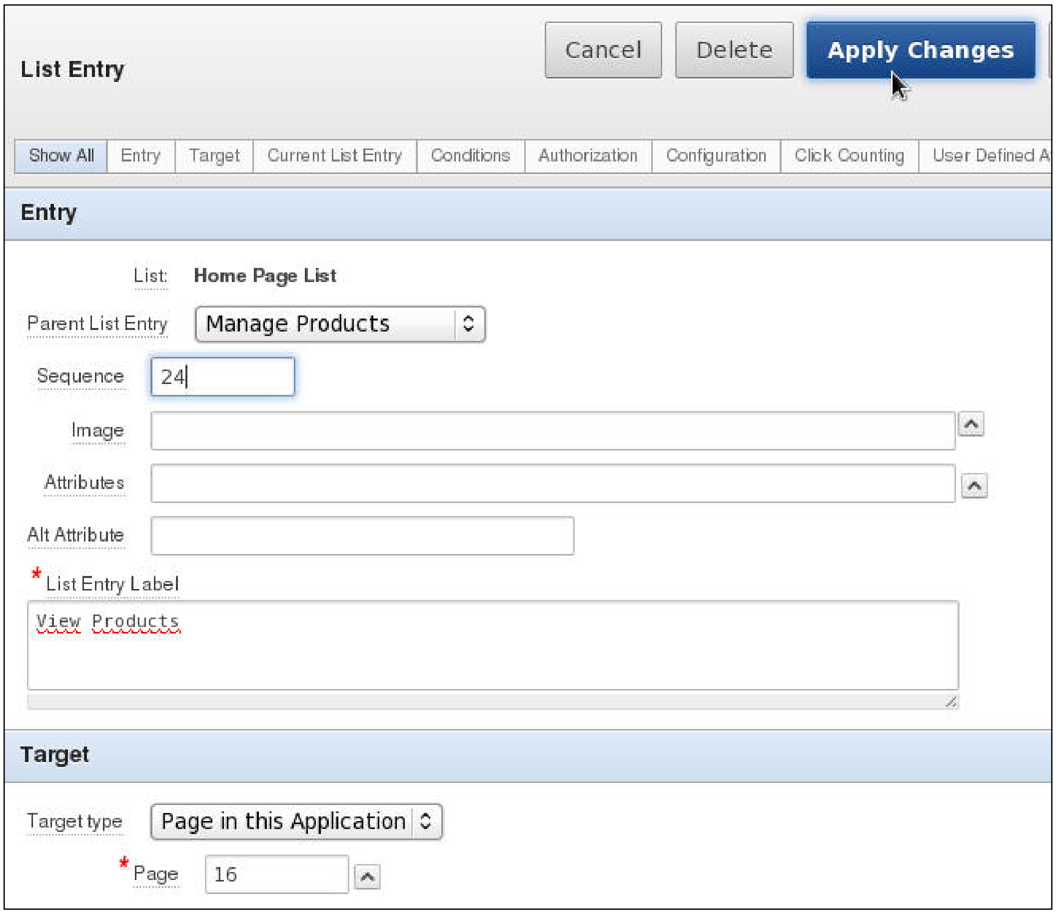
11) In the Entry section, enter **22** for Sequence, and enter **Manage Products** for List Entry Label. Select products.png from the Image pop-up list or enter `#APP_IMAGES#products.png` in the Image area. In the **Target** section, change the Target type to **No Target** and click **Create List Entry**.



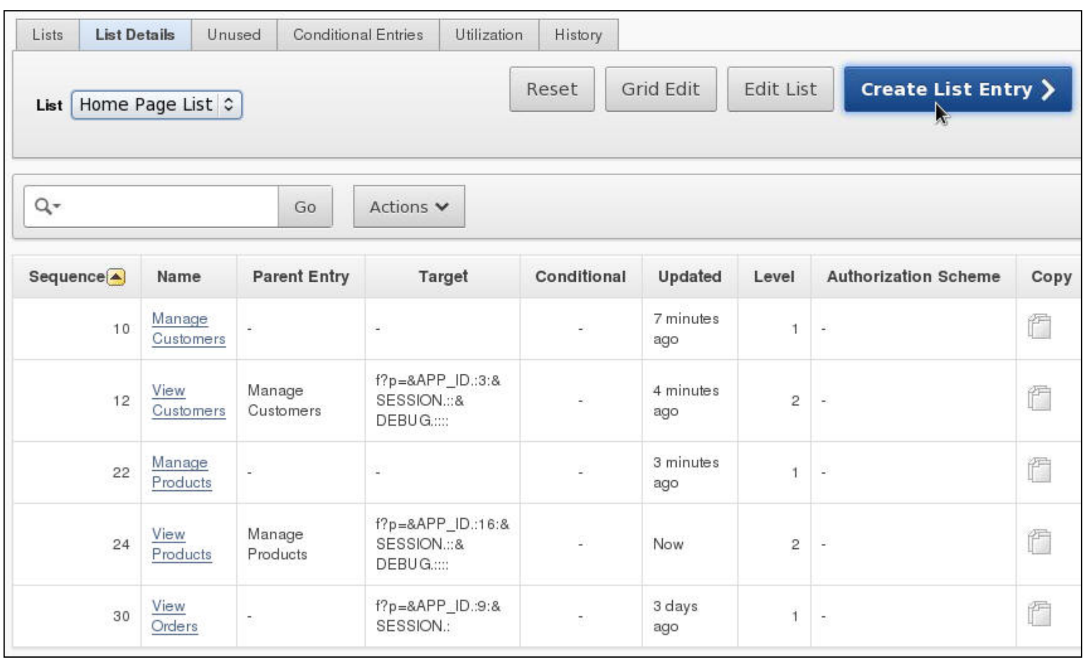
12) You want to change view Products so that it appears in the Manage Products drop-down list. Click the **View Products** link.



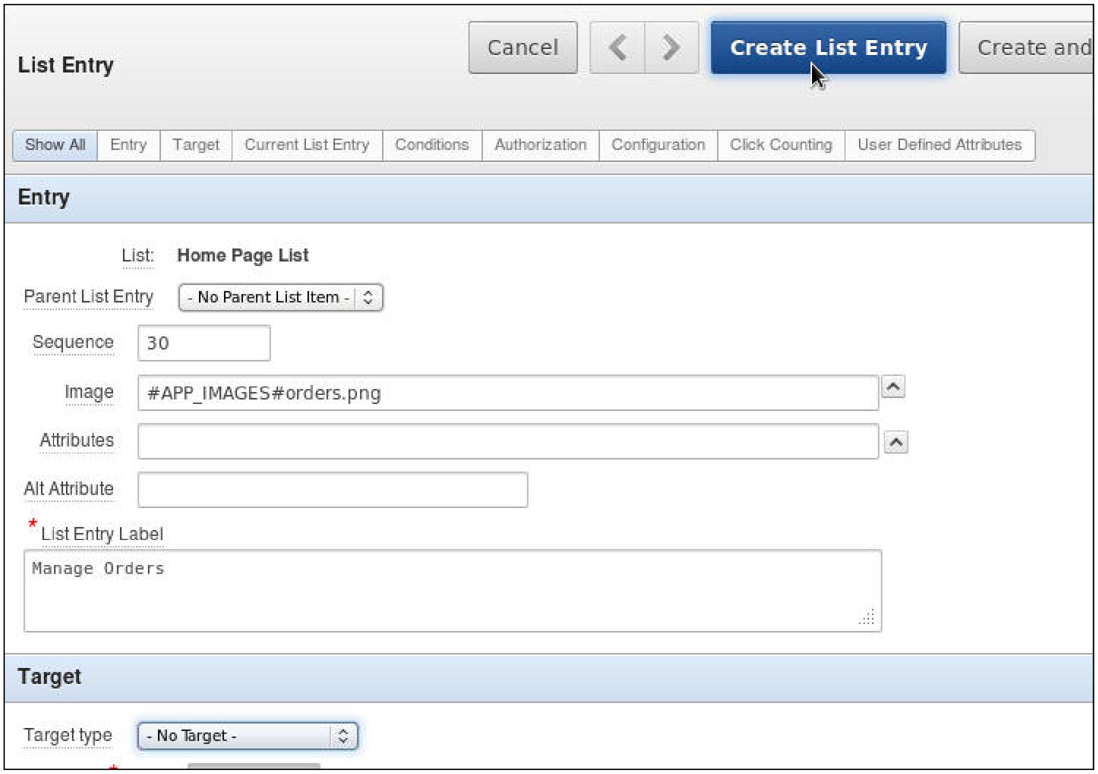
13) Select **Manage Products** for Parent List Entry, enter **24** for Sequence, and click **Apply Changes**.



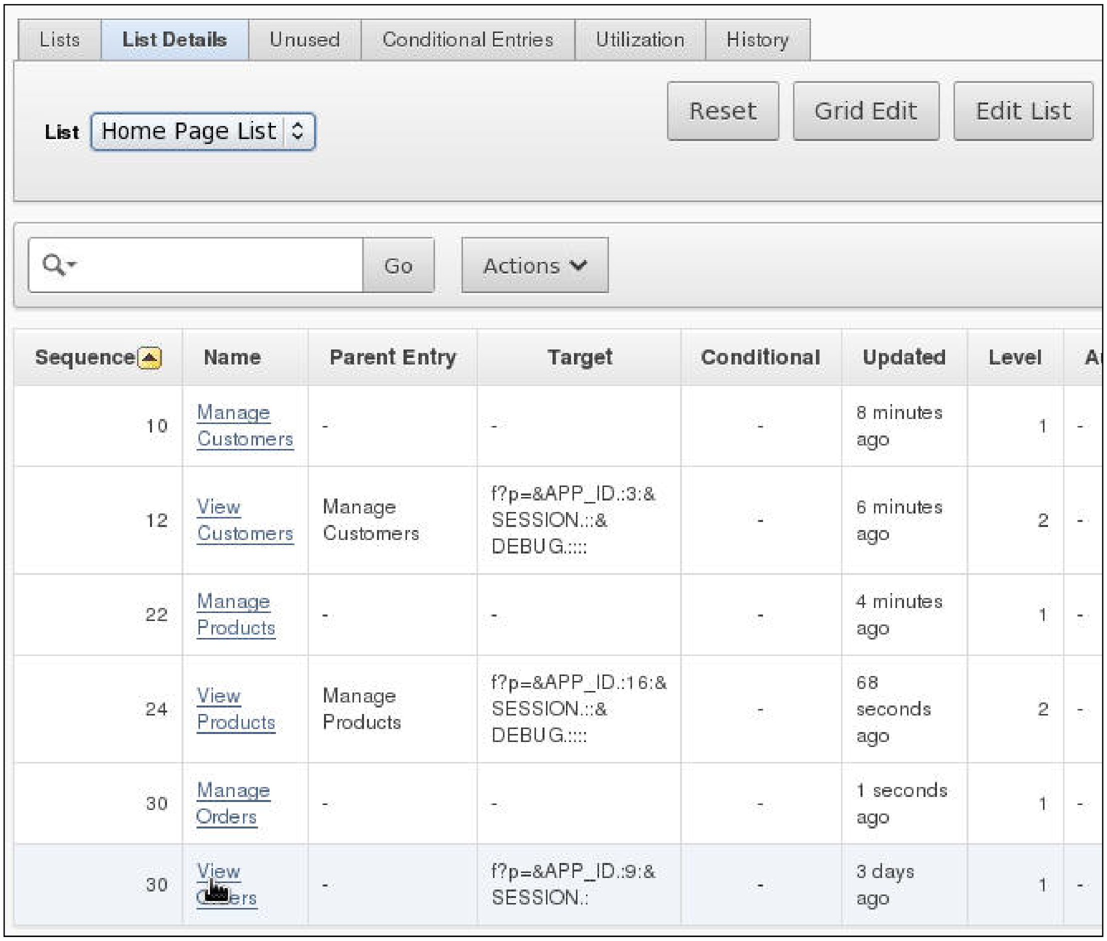
14) Click **Create List Entry**.



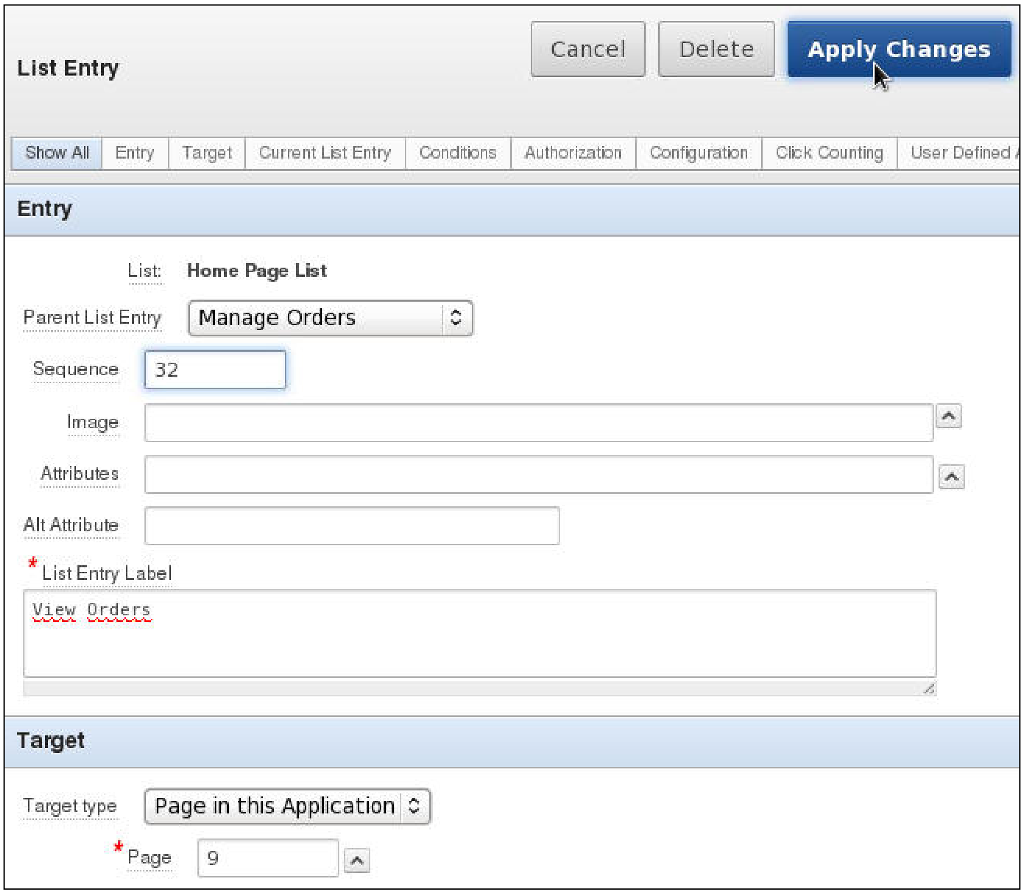
15) In the Entry section, enter **30** for Sequence, enter **`#APP_IMAGES#orders.png`** for Images (or select the image from the pop-up list, and enter **Manage Orders** for List Entry Label. In the Target section, change the Target type to **No Target** and click **Create List Entry**.



16) Click the **View Orders** link.

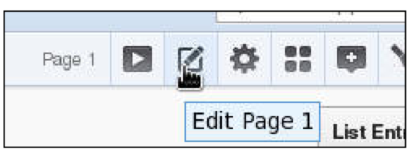


17) Select **Manage Orders** for Parent List Entry, and enter **32** for Sequence. Then click **Apply Changes**.

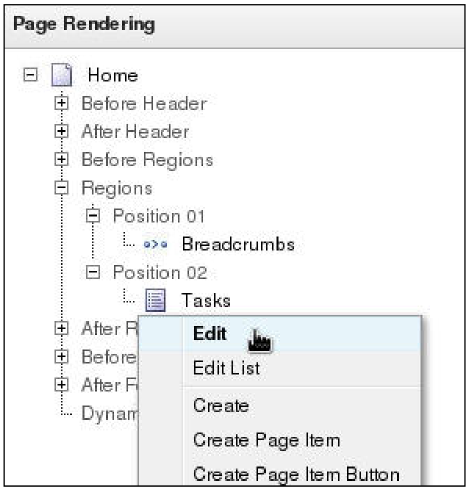


c. In your Navigation region on the Home page, change the template to **No Template** and the List Template to **Pull Down Menu with Image.**

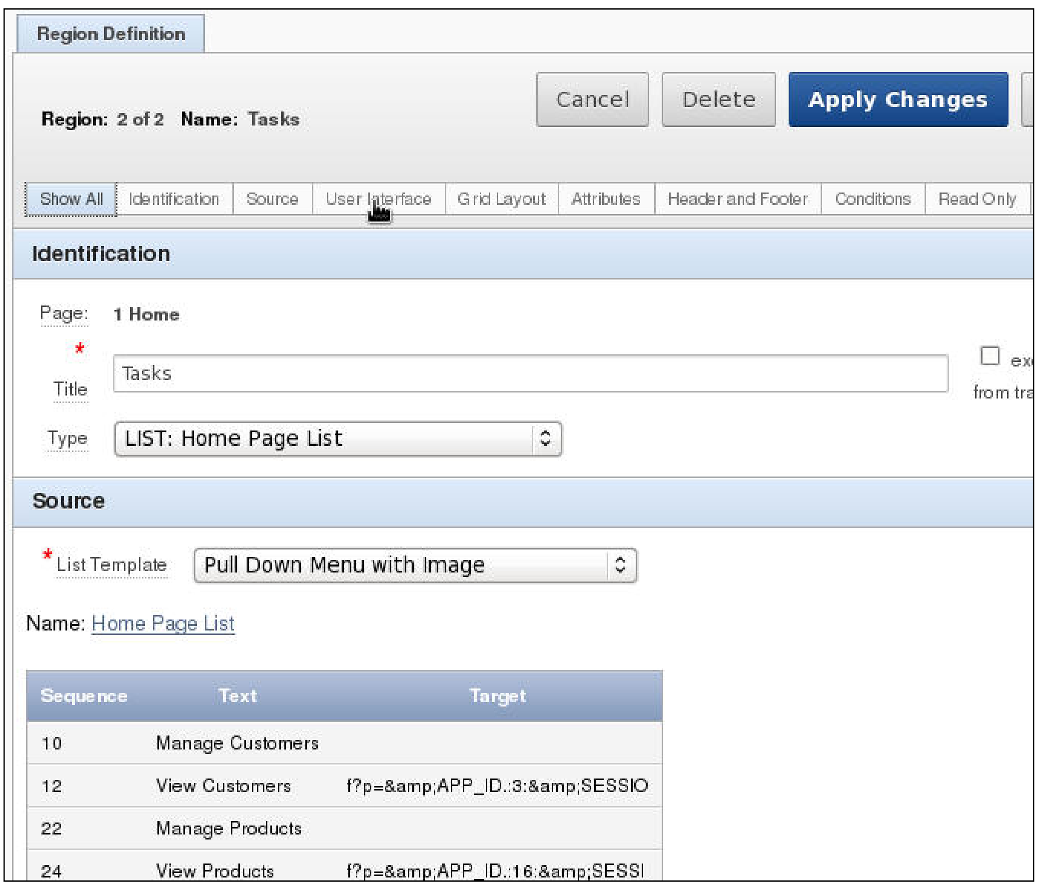
1) Navigate to your **Home** page in the application.



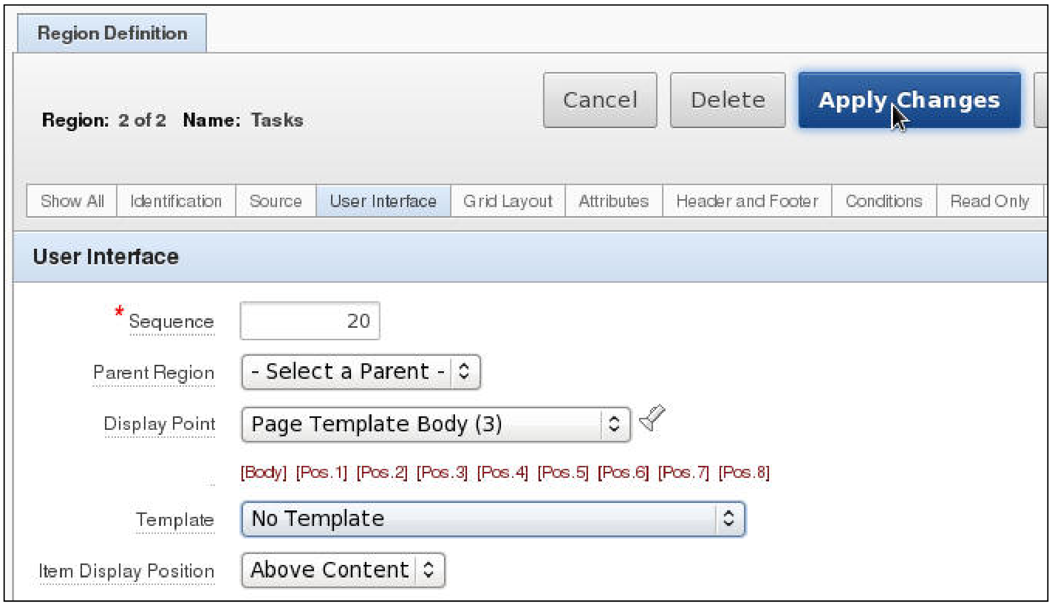
2) Right-click the **Tasks** Region and select **Edit**.



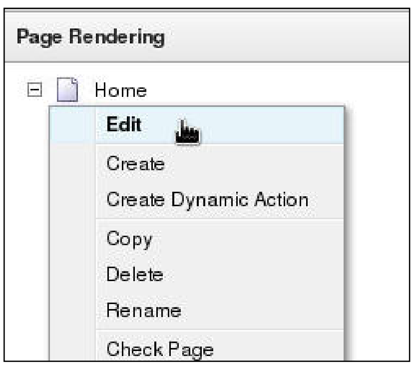
3) Under Source, change the List Template to **Pull Down Menu with Image**. Click the User Interface subtab.



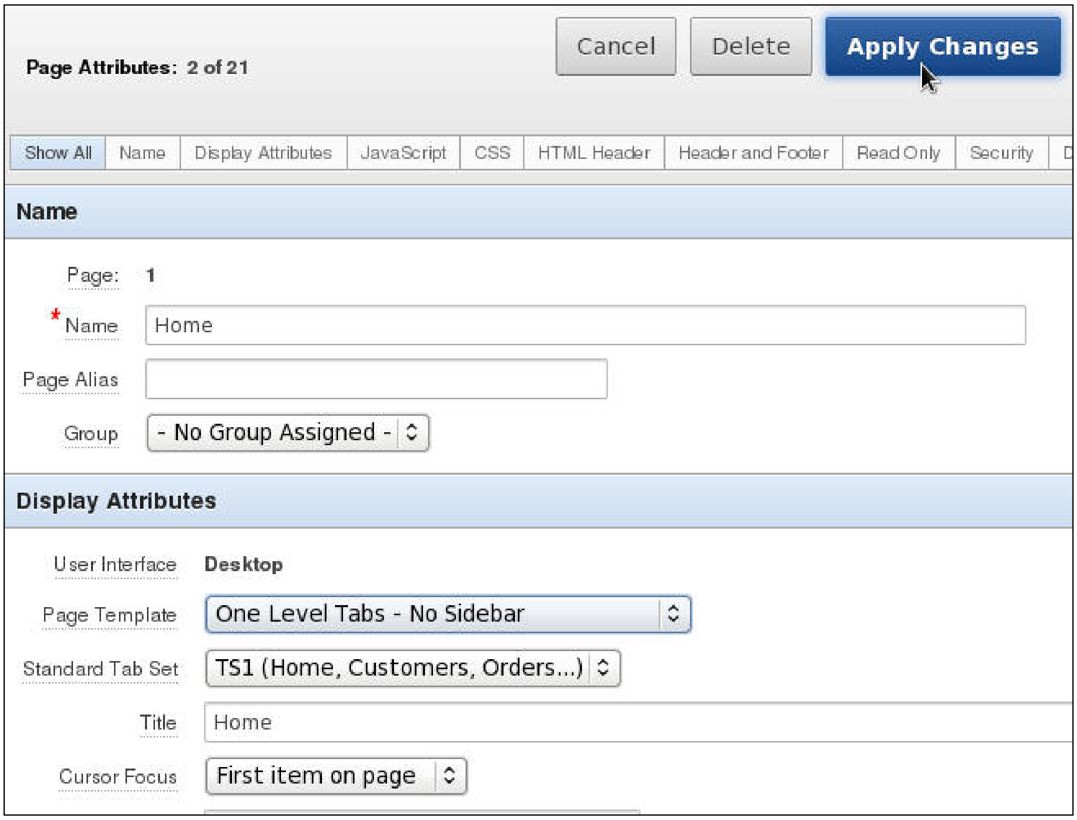
4) Under User Interface, change the Display Point to **Page Template Body (3)** and Template to **No Template**, and click **Apply Changes**.



5) You want to also change the Page Template to have a sidebar. Right-click the **Home** page node and select **Edit**.

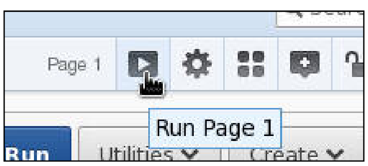


6) Change the Page Template to **One Level Tabs – No Sidebar** and click **Apply Changes**.

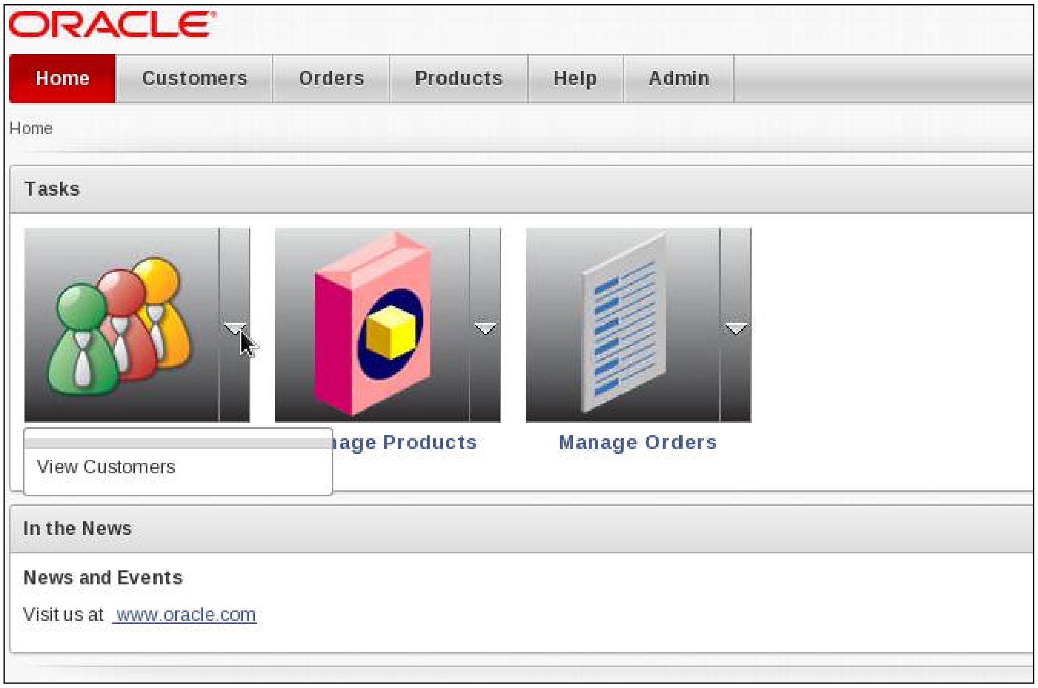


d. Run the page to see the Home Page List. Click the arrow next to each image to show the sublist entries.

1) Click the **Run Page** icon.



2) The page is displayed. Click the down arrow for the first image to see the sublist.



Solution 15-2: Building a Database-Driven Navigation Report

Overview

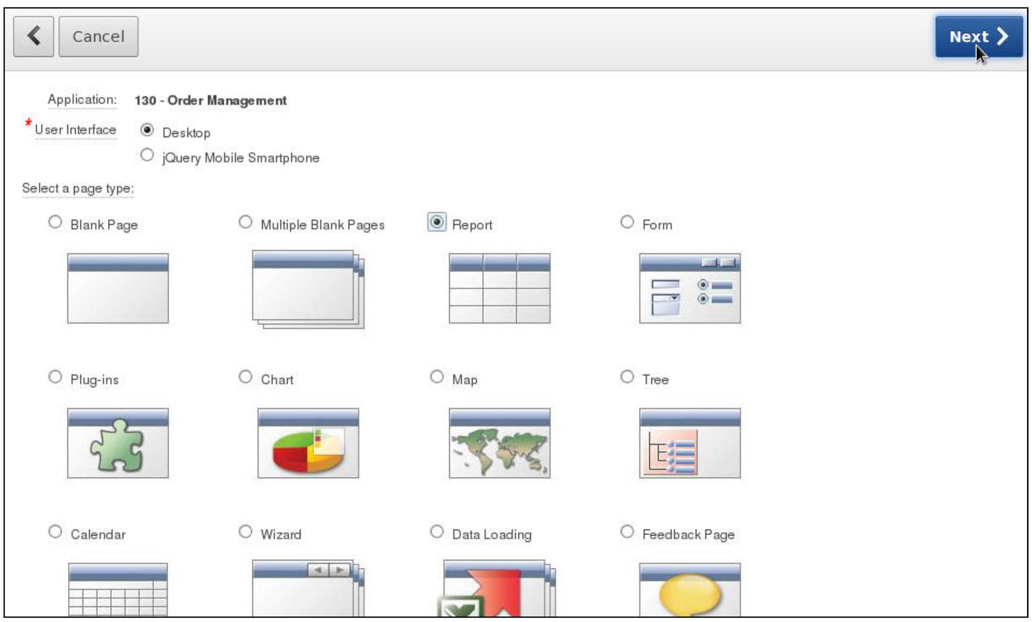
In this practice, you build a report based on the data in a table and then navigate to the detail. You also change the template to the Navigation Region template.

Tasks

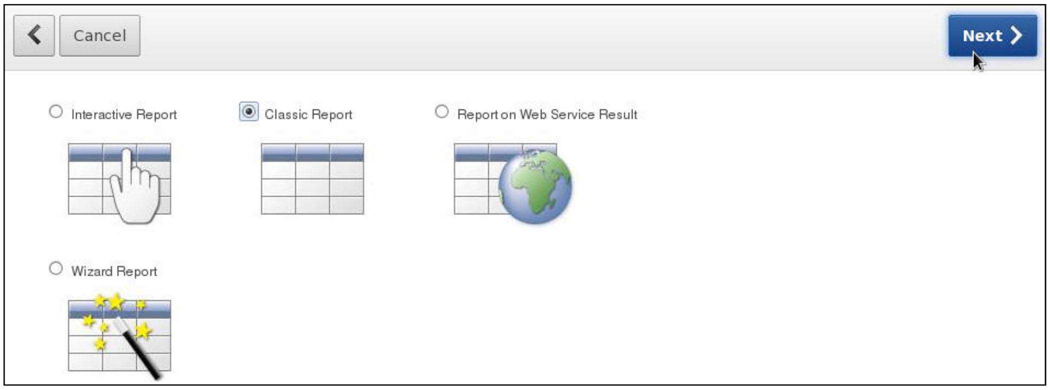
a. Create a new page with a Classic Report called **List of Customers**.

1) Navigate to the Order Management application and click **Create Page**.

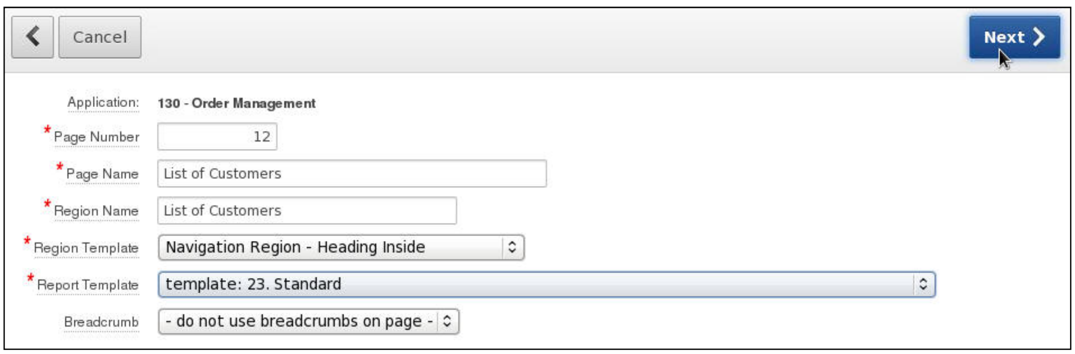
2) Select **Report** and click **Next >**.



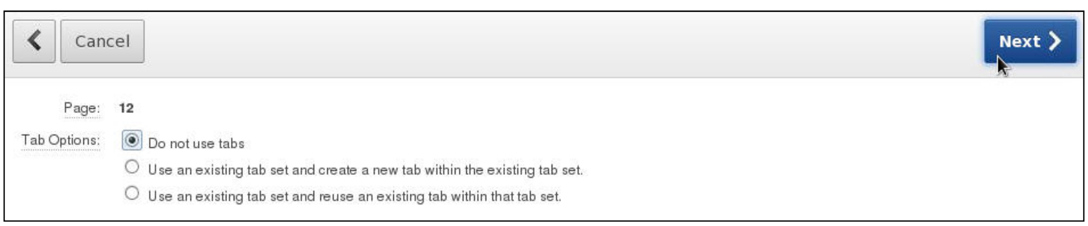
3) Select **Classic Report** and click **Next**.



4) Enter **List of Customers** for Page Name and Region Name. Select **Navigation Region, Heading inside** for Region template and **template 23: Standard** for Report Template. Click **Next >**.

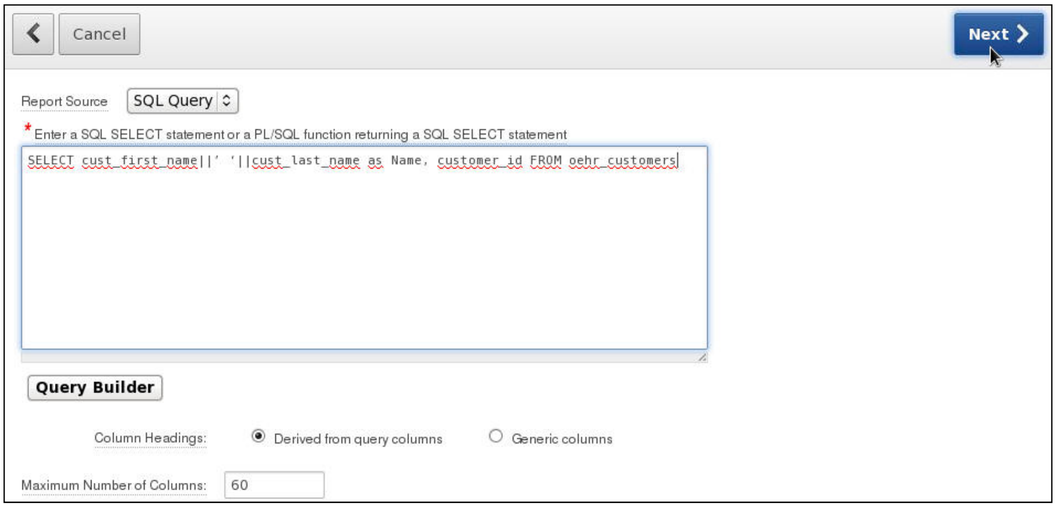


5) Accept the defaults and click **Next >**.

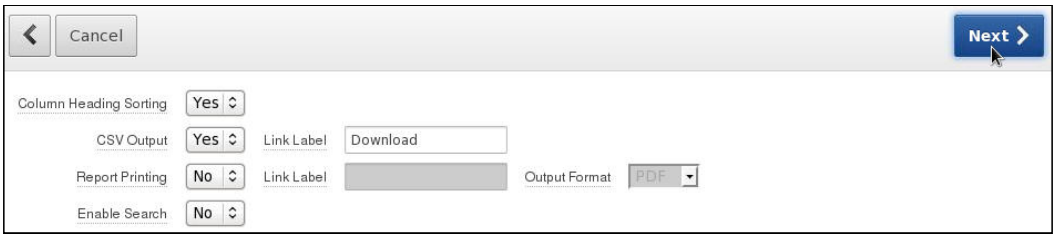


6) For Report Source, select **SQL Query** and enter the following query (located in the `home/oracle/labs/files/lab15_2_1.txt` file). Click **Next >**.

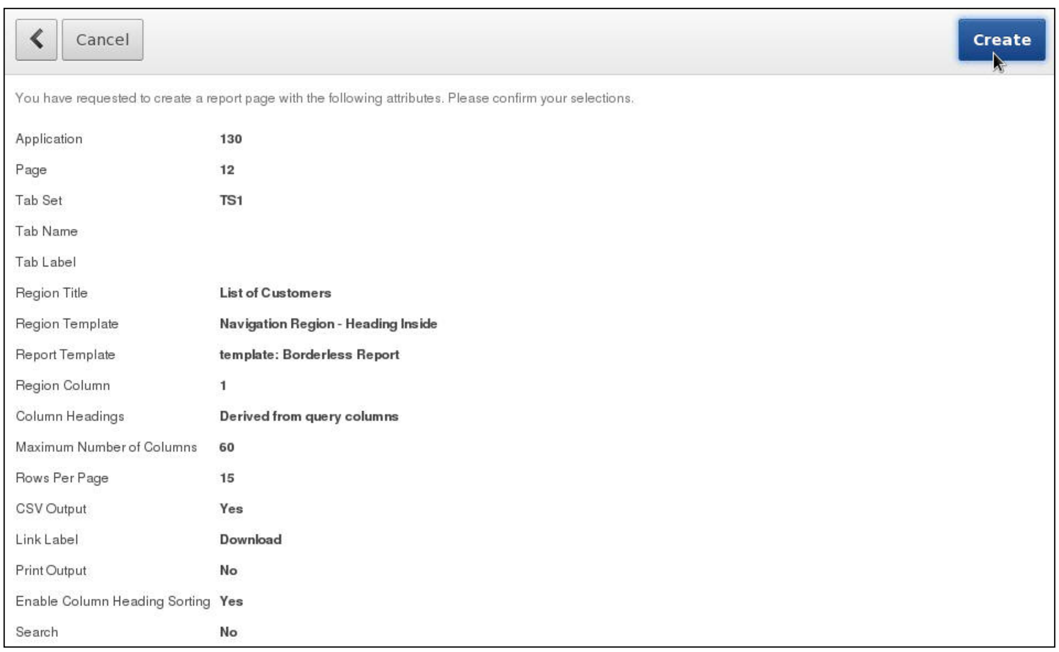
```
SELECT cust_first_name||' '||cust_last_name as Name, customer_id FROM
oehr_customers
```



7) Click **Next >**.



8) Click **Create**.

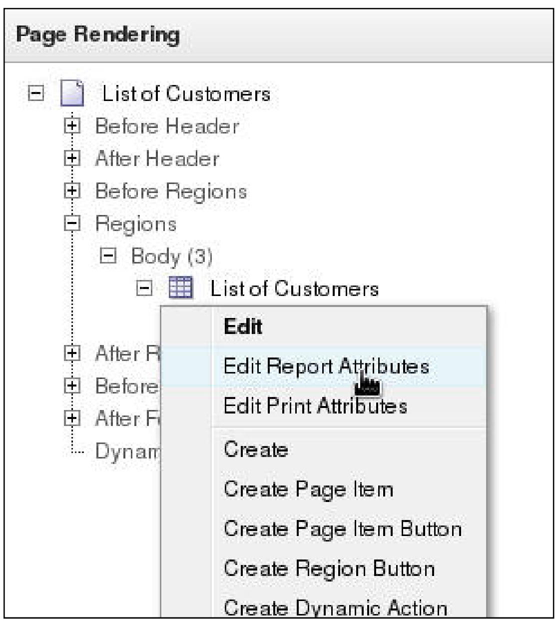


b. Hide the `CUSTOMER_ID` column in the report and turn off pagination. If you want to display the Customer Name column heading in blue, edit the column definition and enter the following for Column Heading.

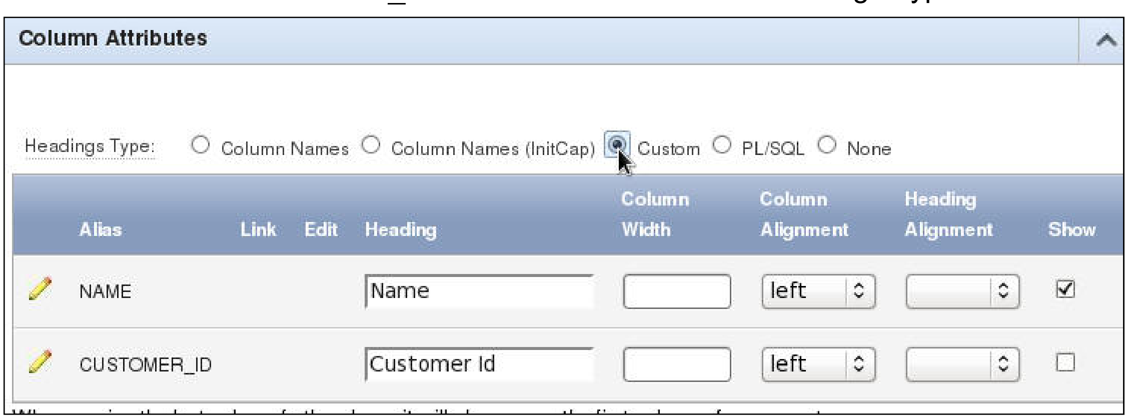
```
<font color="blue">CUSTOMERS</font>
```

1) Click **Edit Page**.

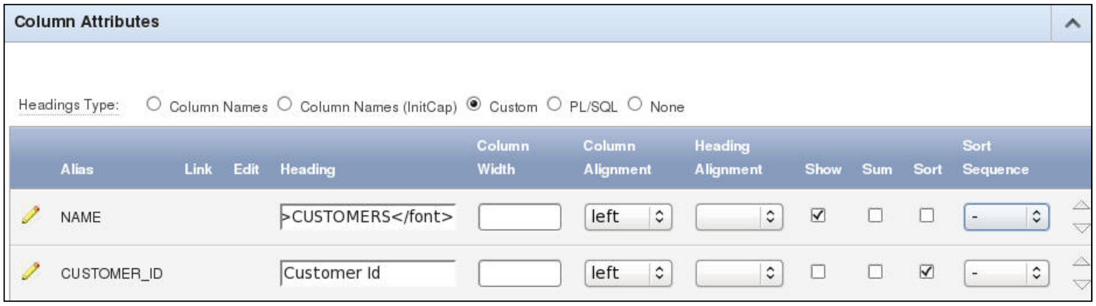
2) Under Regions, right-click **List of Customers** and select **Edit Report Attributes**.



3) Deselect Show for **`CUSTOMER_ID`** and select **Custom** for Headings Type.

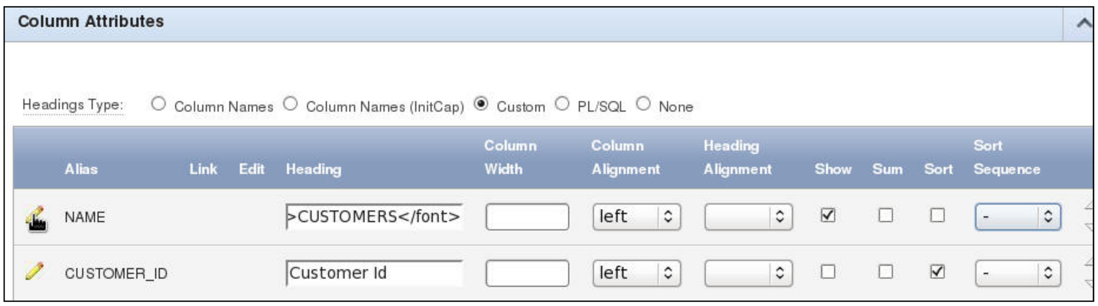


4) Enter `<font color="blue">CUSTOMERS</font>` for the **NAME** heading, deselect the **Sort** check box for **NAME**, and select "**-**" for Sort Sequence.

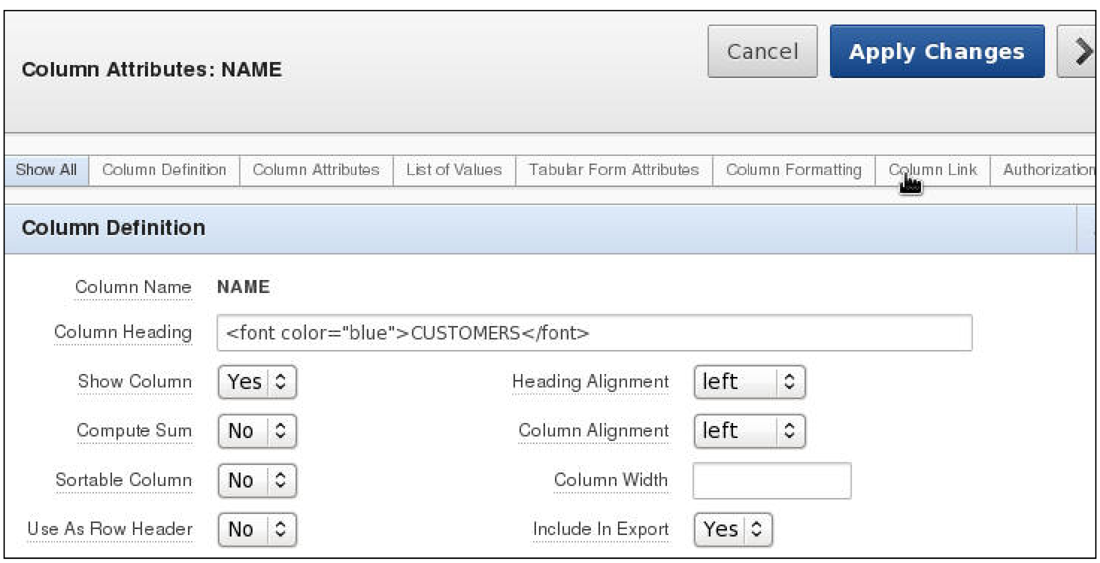


c. Create a link to the Customer Details page on the `NAME` column. Pass the **P<n>**`_CUSTOMER_ID` item from the Customer Details page with the value `#CUSTOMER_ID#` from this page.

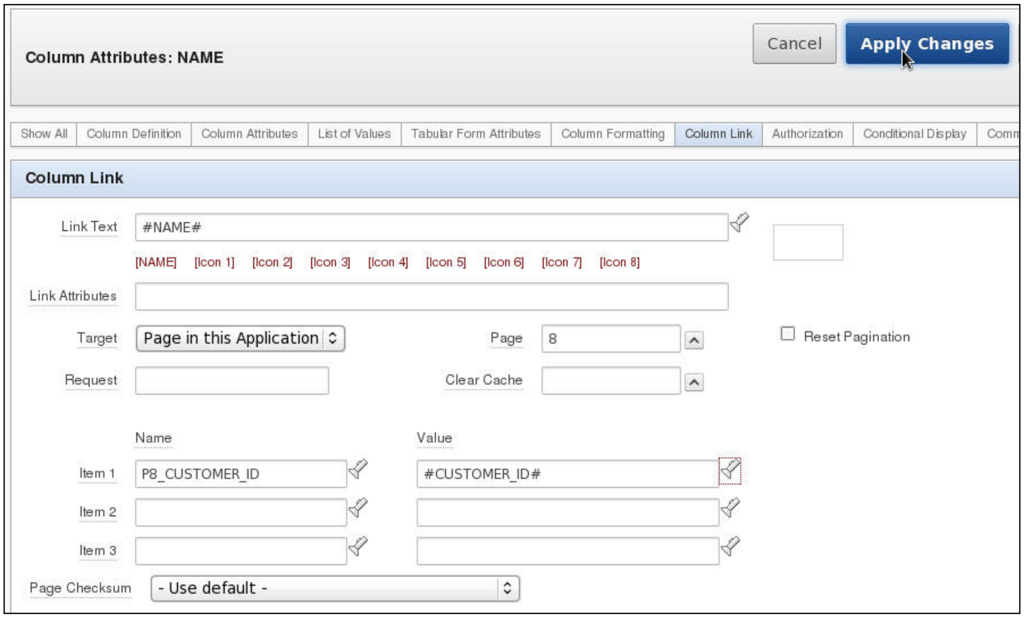
1) Click the edit icon in front of the `NAME` column.



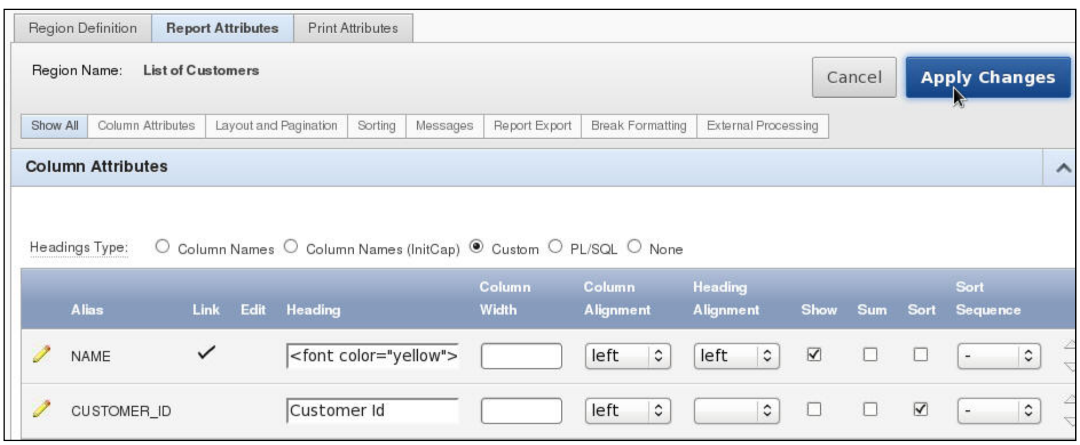
2) Click the **Column Link** tab.



3) Click the **[NAME]** quick pick for Link Text. Select the **Customer Details** page for **Page** (in this case, **8**) by clicking the arrow to the right of **Page.** Under Name, select **P8_CUSTOMER_ID** for Item 1 and under Value, select **#CUSTOMER_ID#** and click **Apply Changes.**

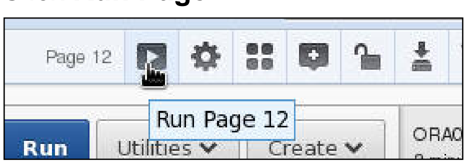


4) Click **Apply Changes**.

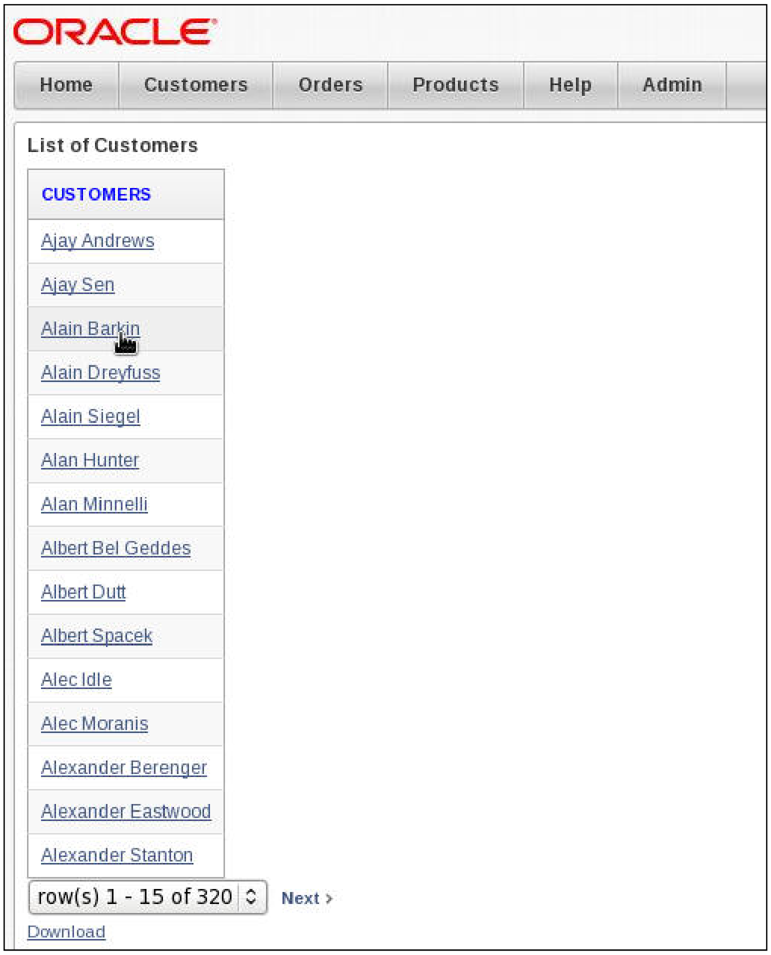


d. Run the page to see the database-driven report.

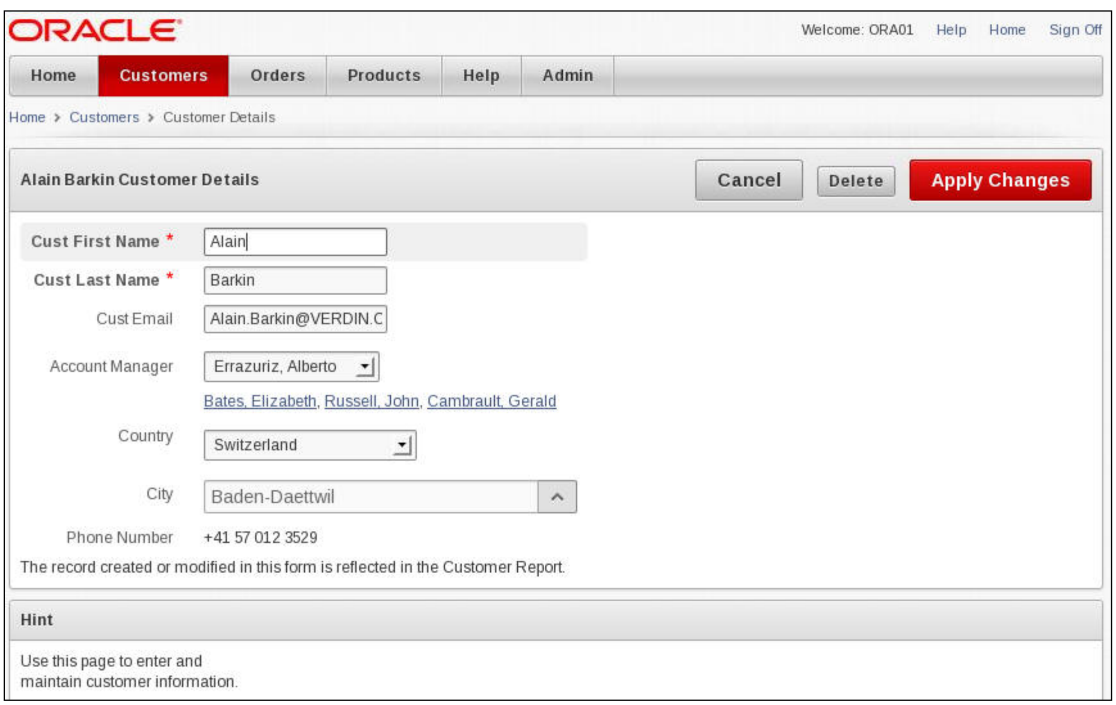
1) Click **Run Page**.



2) Click one of the Customers to view the Customer Details.



3) The Customer Details page is displayed for the customer.



Solution 15-3: Building a Site Map

Overview

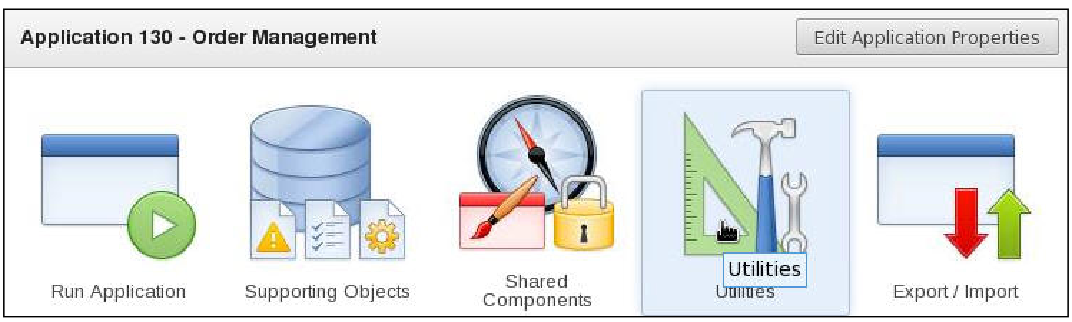
In this solution, you build a site map page that will be added as a navigation bar entry.

Steps

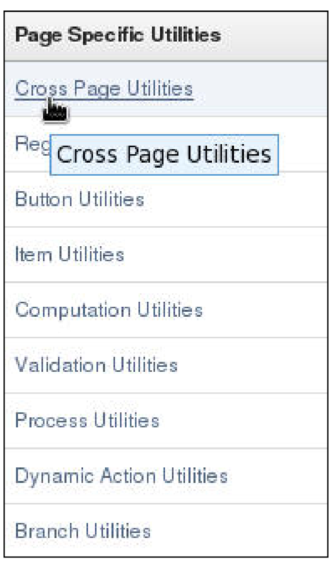
a. Assign each page that you want in the site map to the same page group. Create a Page Group called Site Map and assign the following pages to the Site Map page group.

Customers
Products
List of Orders
Customer Feedback
Access Control Administration Page

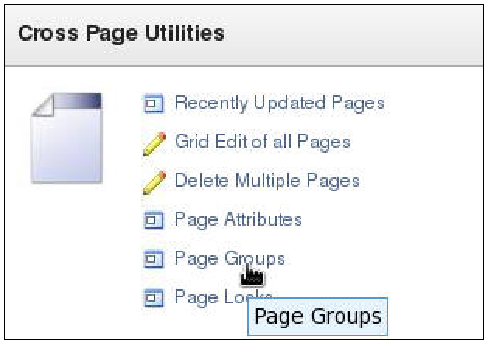
1) Navigate to **Application Builder** and click **Utilities**.



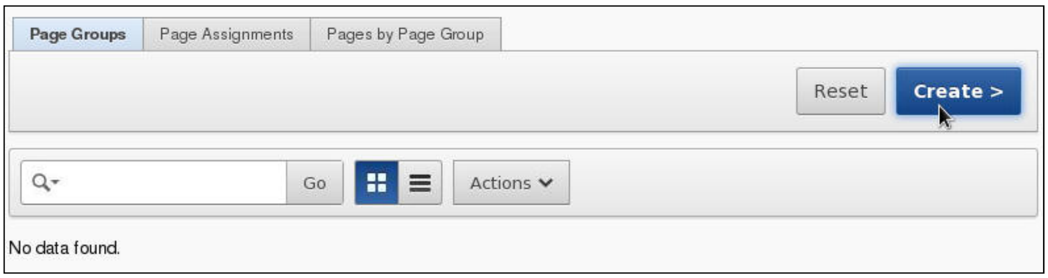
2) Select **Cross Page Utilities** under Page Specific Utilities.



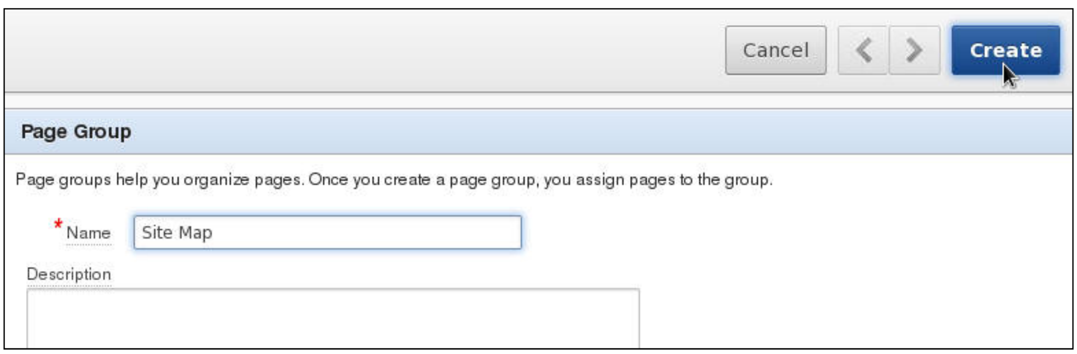
3) Select **Page Groups**.



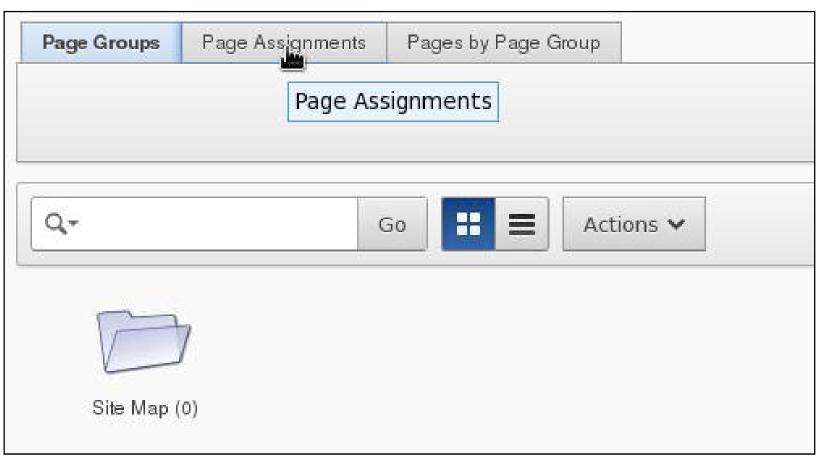
4) Click **Create >**.



5) Enter `Site Map` and click **Create**.

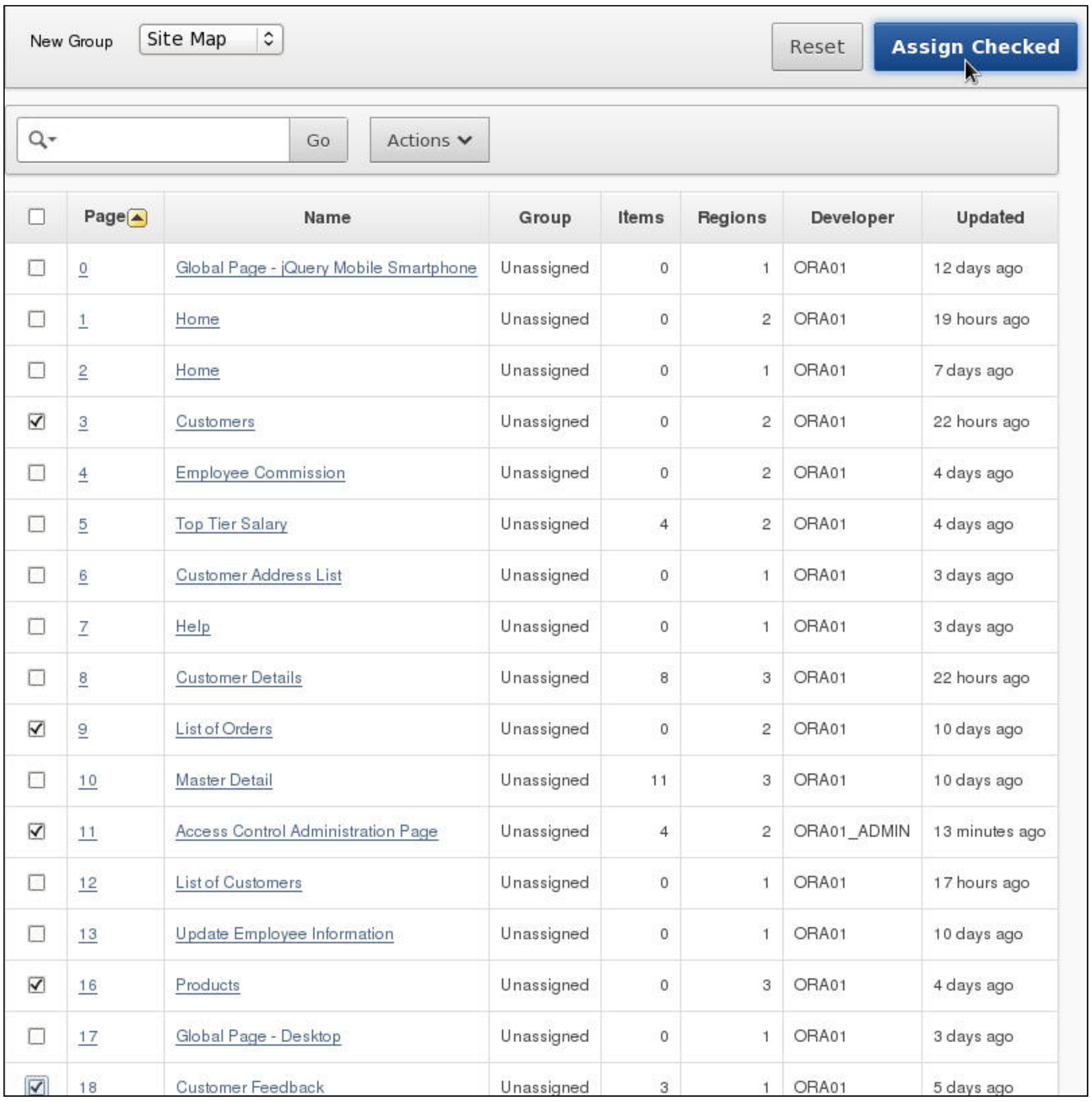


6) Click the **Page Assignments** tab.



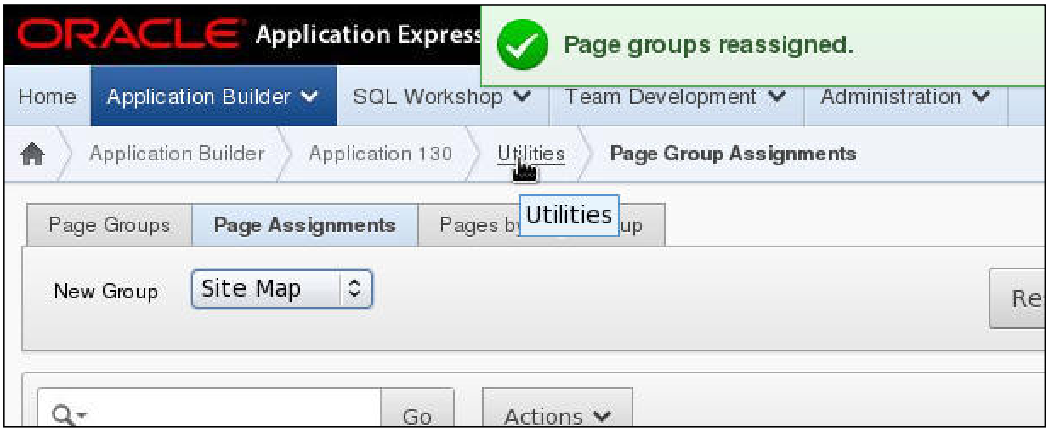
7) Select **Site Map** for New Group. Then select the check box next to each of the following pages and click **Assign Checked**.

Customers
Products
List of Orders
Customer Feedback
Access Control Administration Page

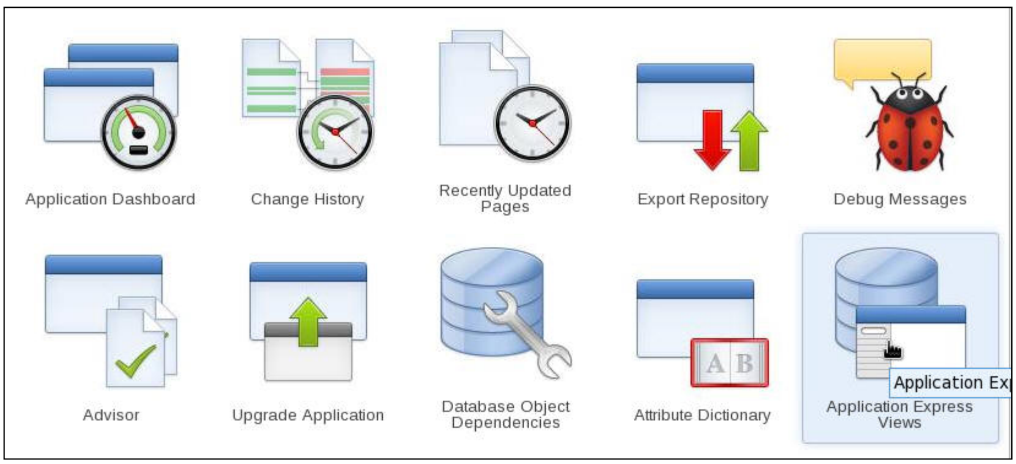


b. Create a new page called Site Map with a Classic Report that shows only the pages in the Site Map page group.

1) Click the **Utilities** breadcrumb.

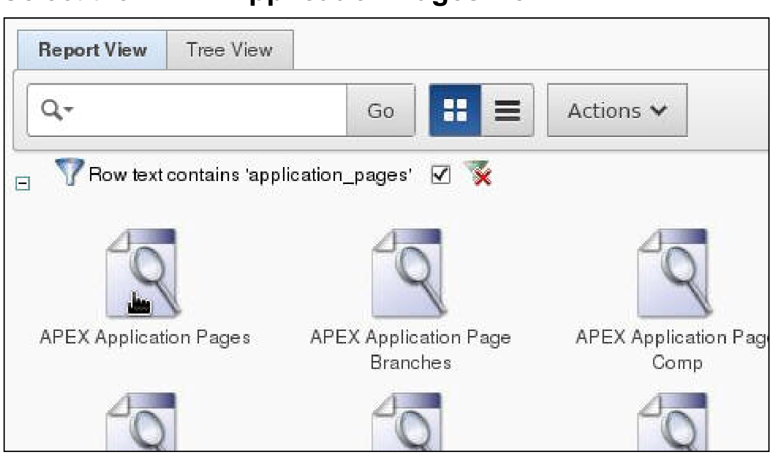


2) Select **Application Express Views**.

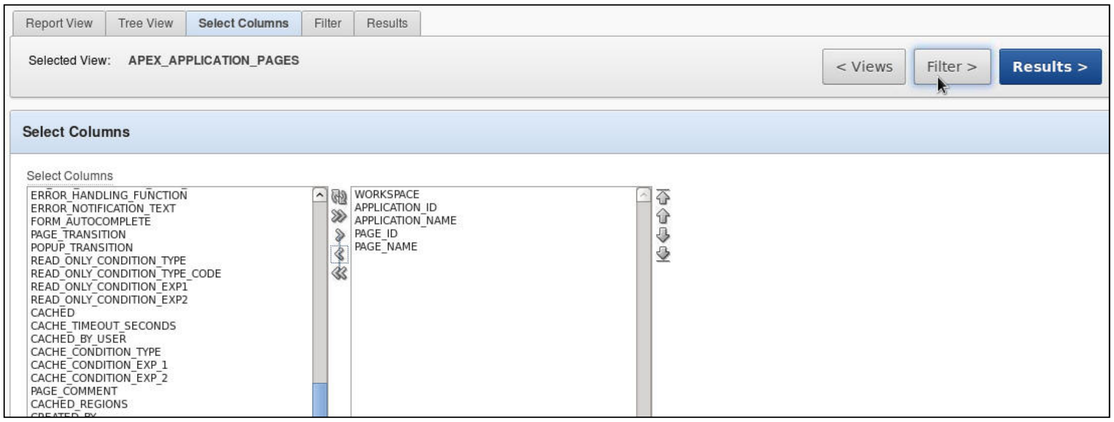


3) Enter **`Application_Pages`** in the search area and click **Go**.

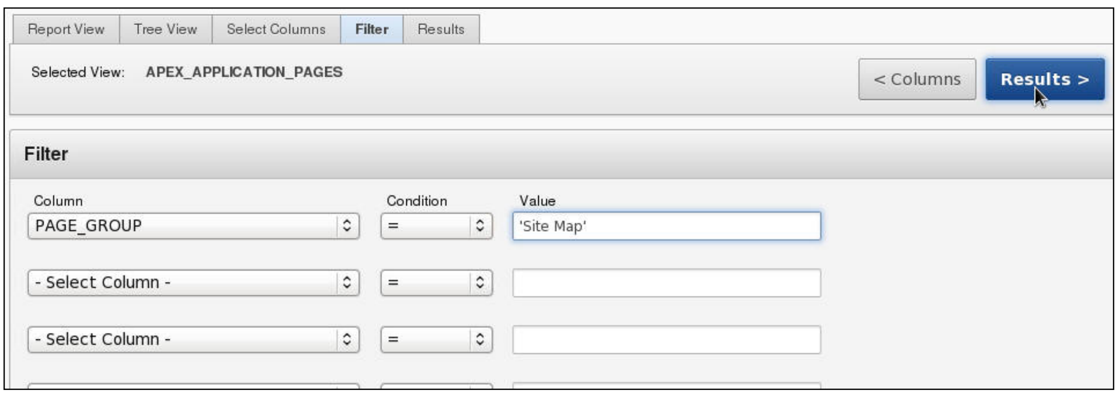
4) Select the **APEX Application Pages** view.



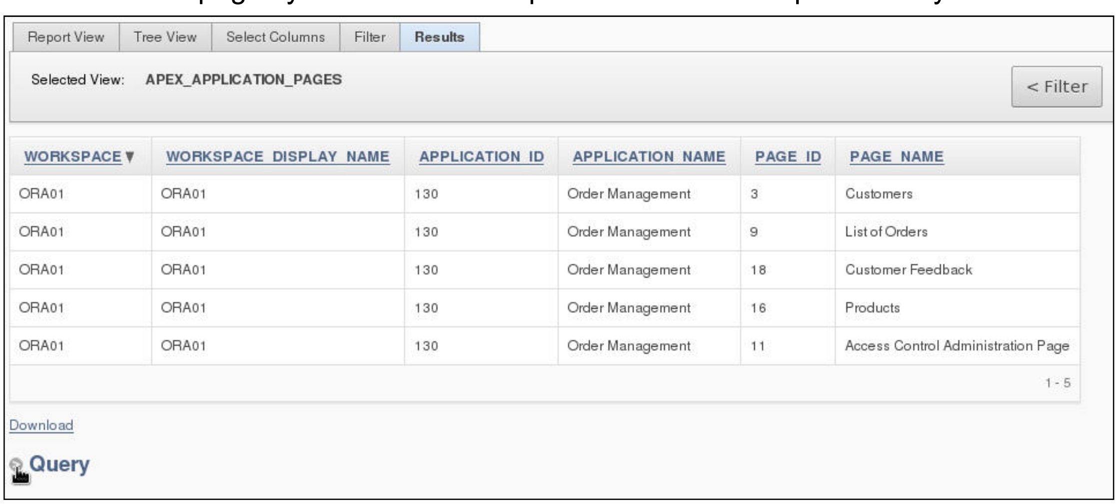
5) On the Select Columns tab, if **PAGE_NAME** is not included in the list on the right, then select **PAGE_NAME** column and click the right arrow (**>**) to move it to the columns selected list. Similarly, you do not want to display the **WORKSPACE_DISPLAY_NAME**. Select **WORKSPACE_DISPLAY_NAME** and click the left arrow (<) to move it to the column list on the left. Click **Filter**.



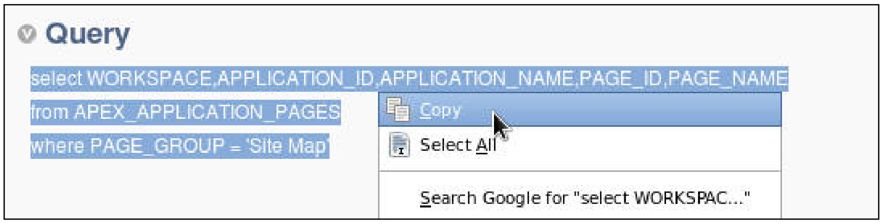
6) Select **PAGE_GROUP** for Column and enter **'Site Map'** for Value. Click **Results**.



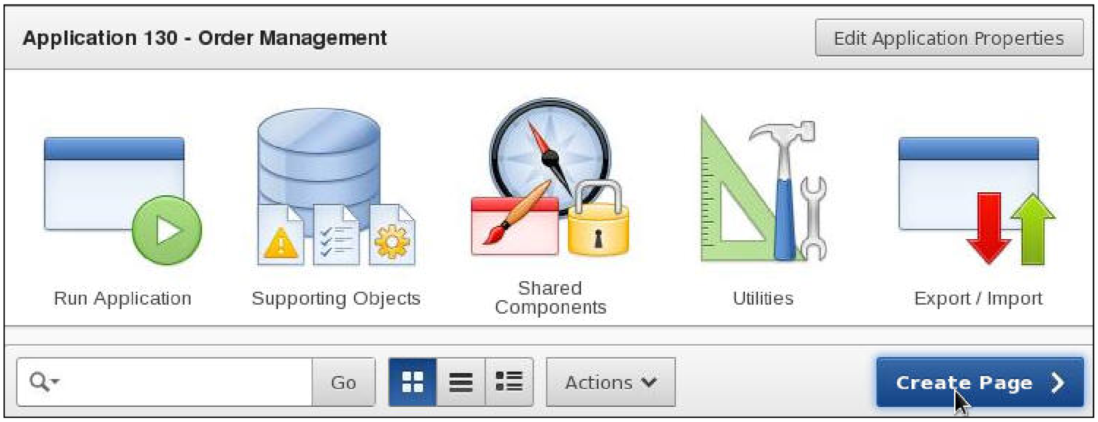
7) Notice that the pages you selected in step 6 are in the list. Expand Query.



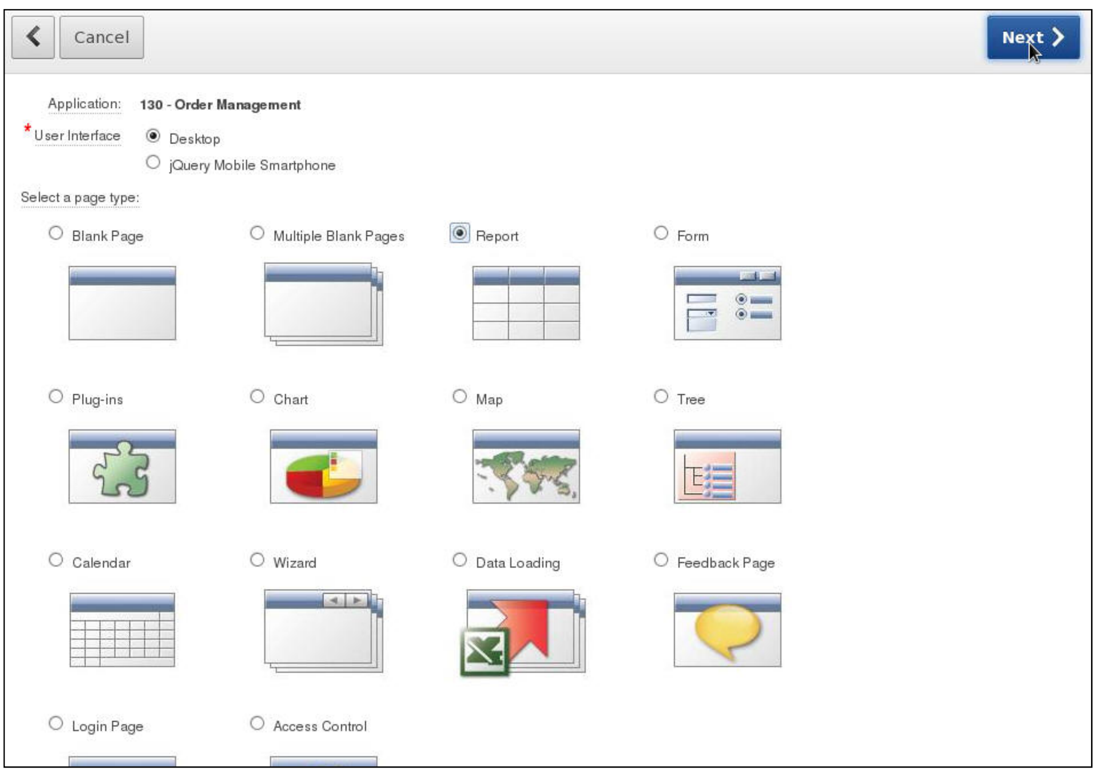
8) Select the query and copy it to your clipboard.



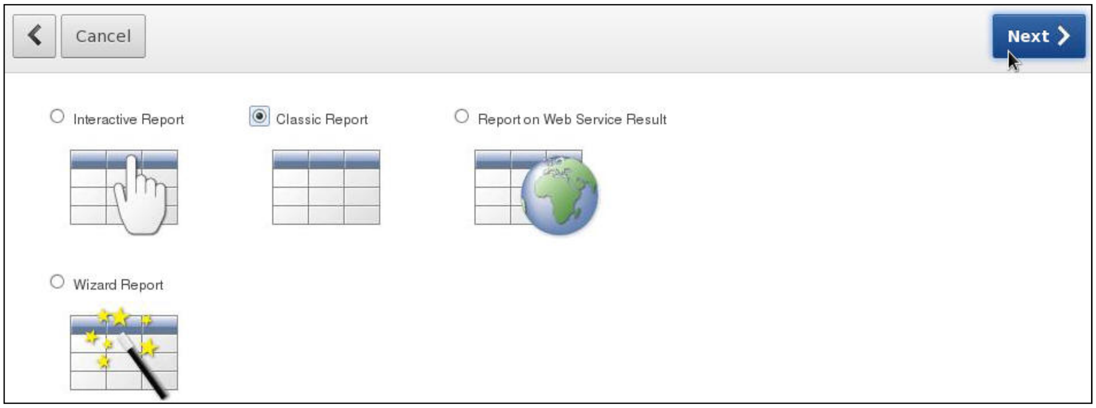
9) Navigate to the Order Management Home page and click **Create Page**.



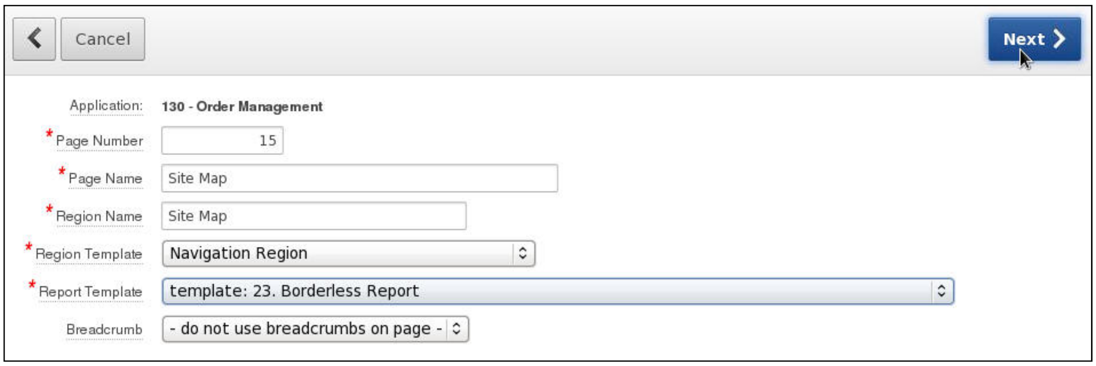
10) Select **Report** and click **Next >.**



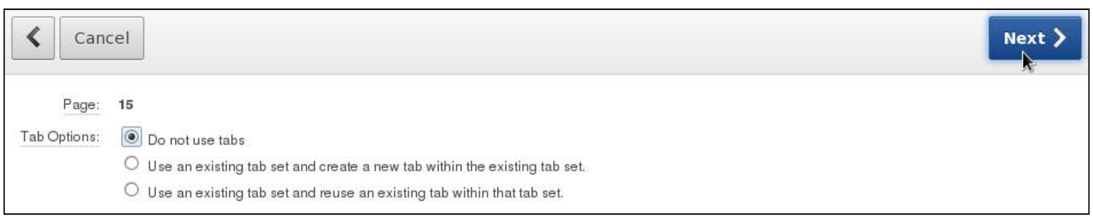
11) Select **Classic Report** and click **Next >**.



12) Enter **`Site Map`** for Page Name and Region Name. Select **Navigation Region** for Region Template and **template: 23. Borderless Report** for Report Template. Click **Next >**.

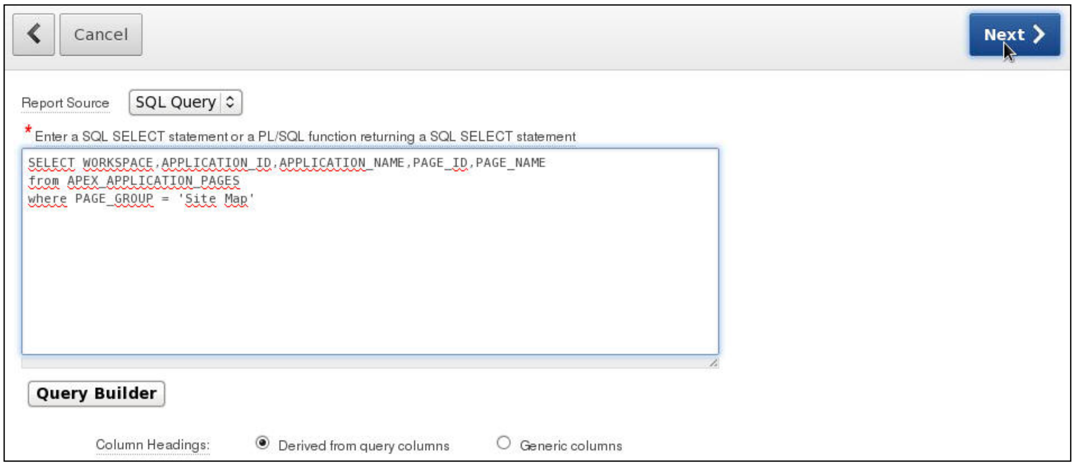


13) Click **Next >**.

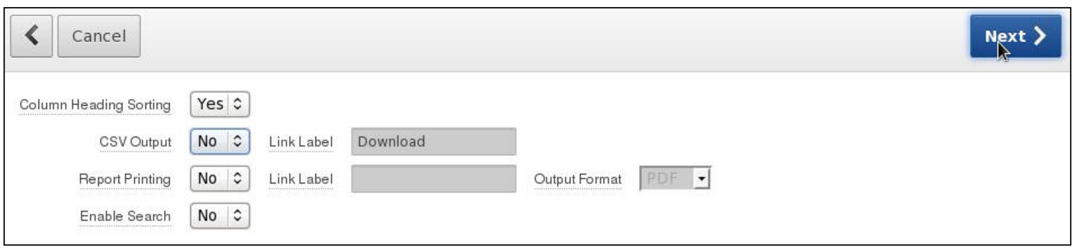


14) Select **SQL Query** for Report Source. Paste your query in the SQL area and click **Next >**. You can also copy and paste the query from the `home\oracle\labs\files\lab15_3_1.txt` file.

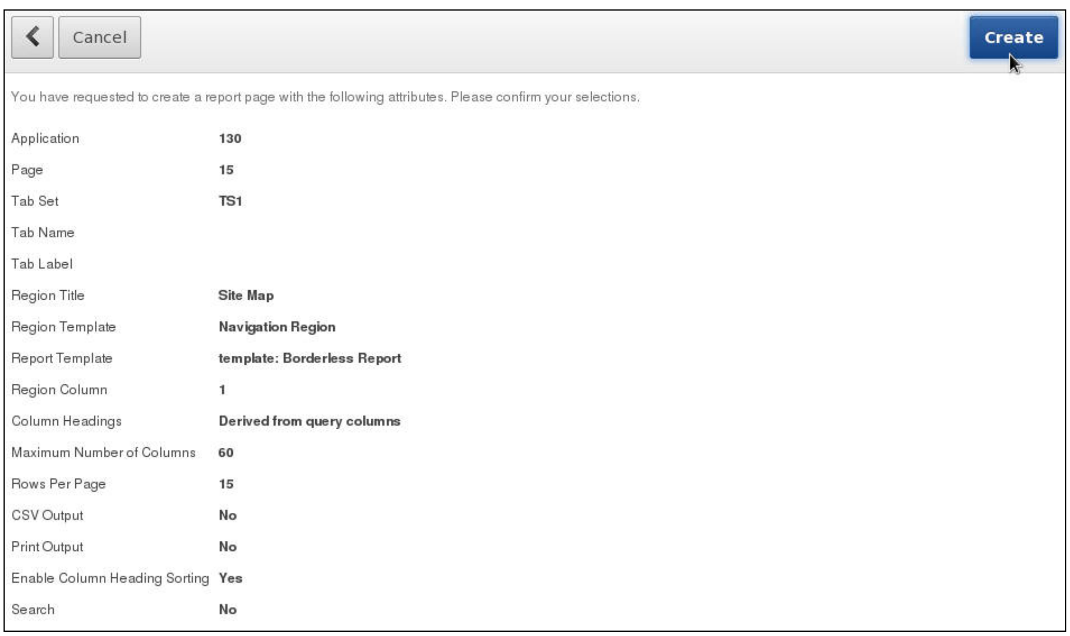
```
select WORKSPACE,APPLICATION_ID,APPLICATION_NAME,PAGE_ID,PAGE_NAME
from APEX_APPLICATION_PAGES
where PAGE_GROUP = 'Site Map'
```



15) Select **No** for **CSV Output** and click **Next >**.

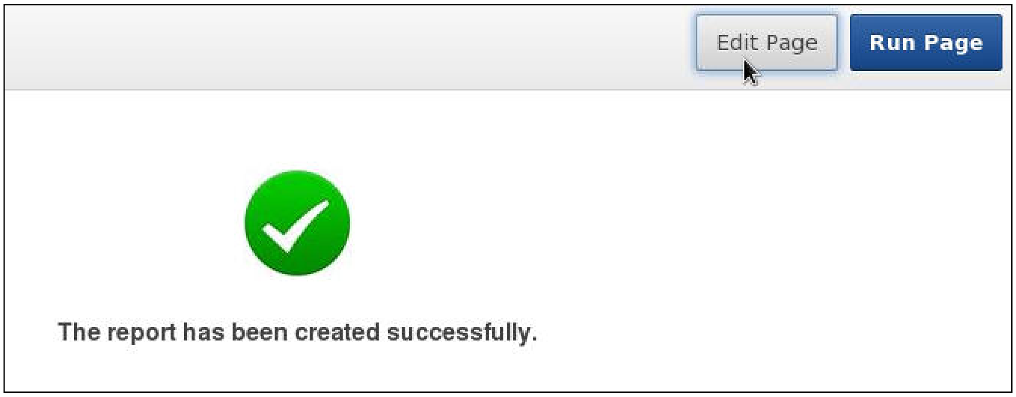


16) Click **Create**.

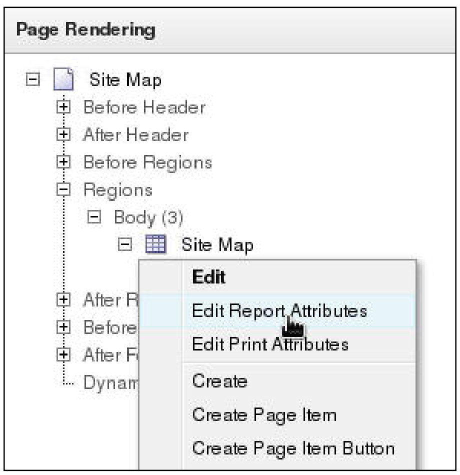


c. Change the Pagination Scheme to None and do not show the `PAGE_ID` column. Link the `PAGE_NAME` column to the `#PAGE_ID#` so that it opens the corresponding page when clicked.

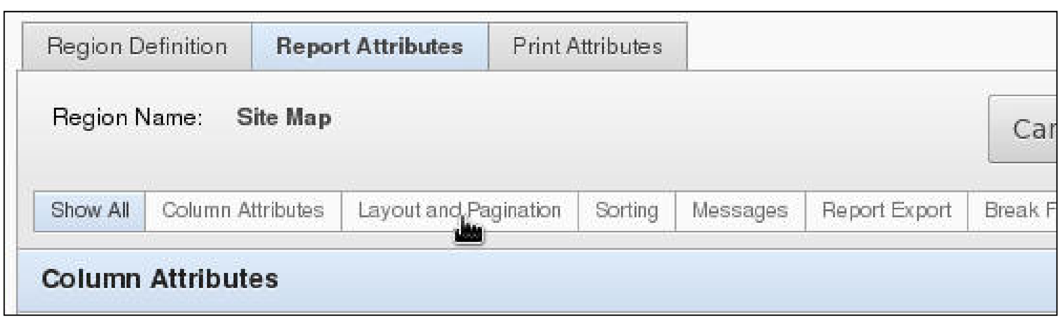
1) Click **Edit Page**.



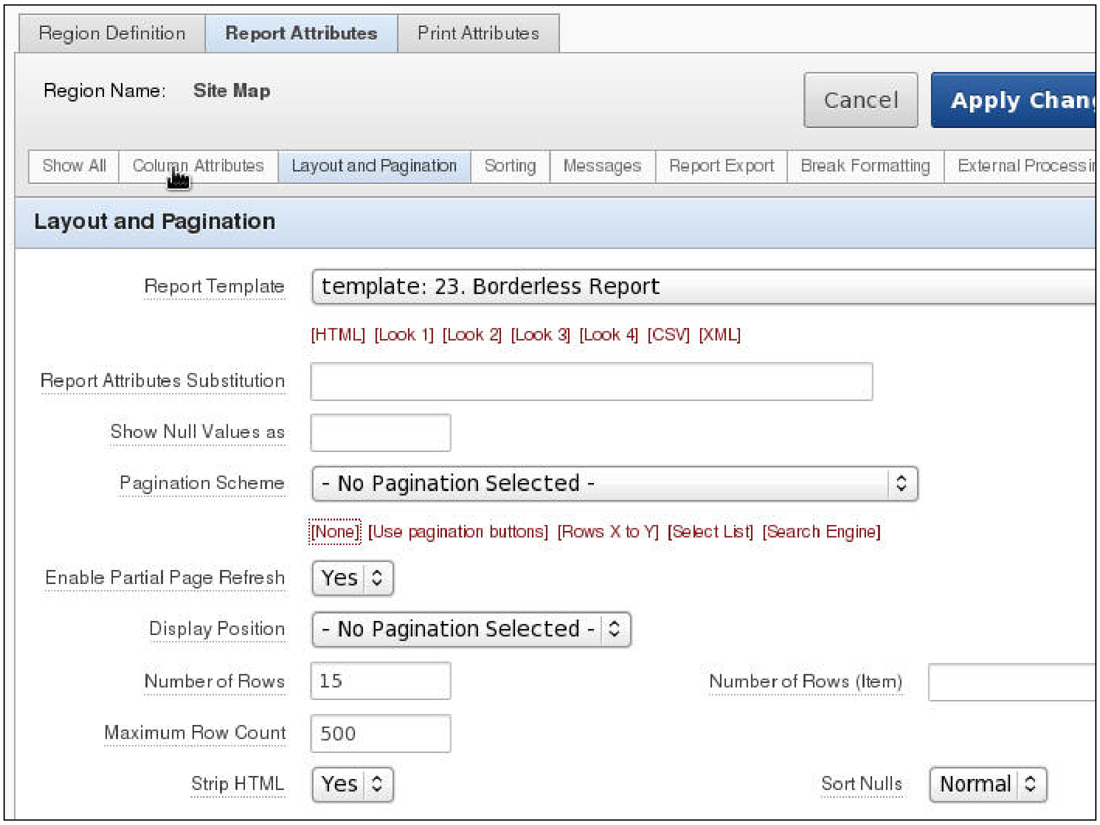
2) Under Regions, right-click the **Site Map** region and select **Edit Report Attributes**.



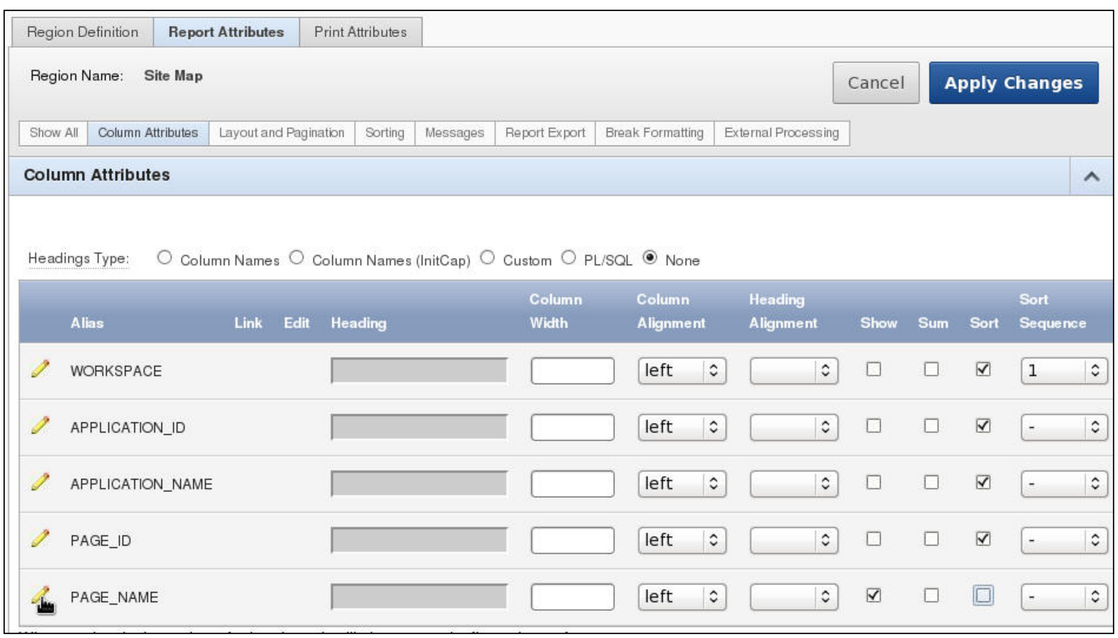
3) Click the **Layout and Pagination** tab.



4) Select **[None]** from the Pagination Scheme quick picks and click the **Column Attributes** tab.

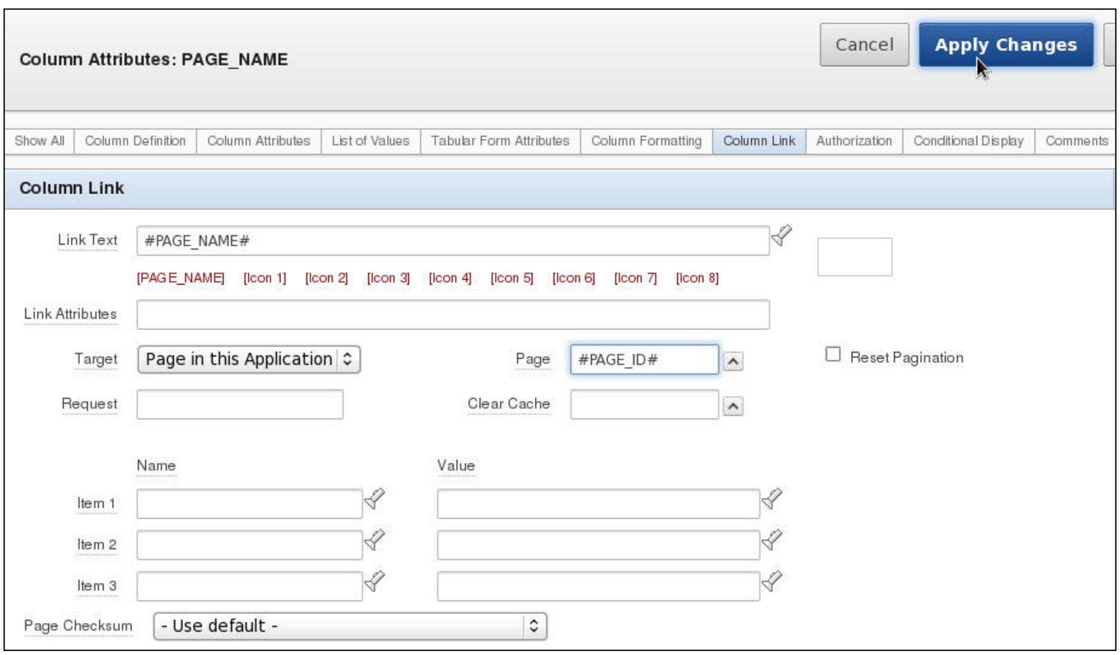


5) Deselect all the Show check boxes for all columns except `PAGE_NAME`. Select the **None** option for Headings Type. Deselect Sort for `PAGE_NAME` and click the edit icon next to the row.



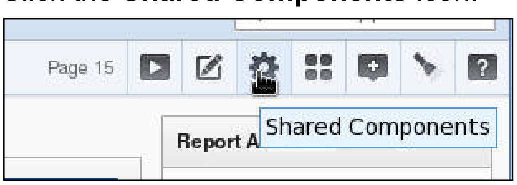
6) Click the **Column Link** tab.

Click the **[PAGE_NAME]** quick pick. Make sure that **Page in this Application** is selected for Target and enter **`#PAGE_ID#`** in the Page text box. Click **Apply Changes**.

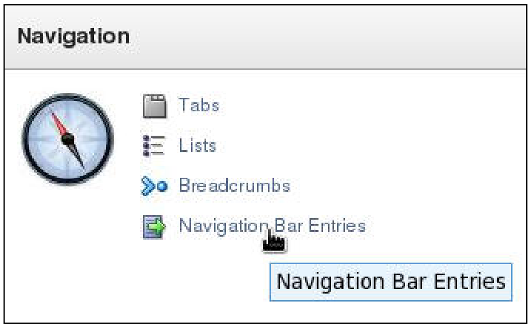


d. Add this page to your Navigation Bar and call it Site Map.

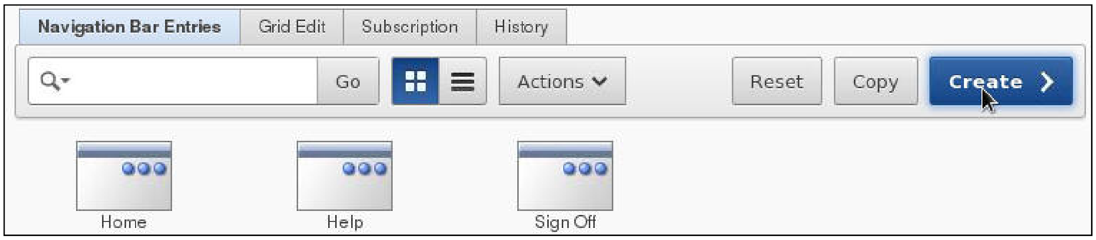
1) Click the **Shared Components** icon.



2) Select **Navigation > Navigation Bar Entries**.

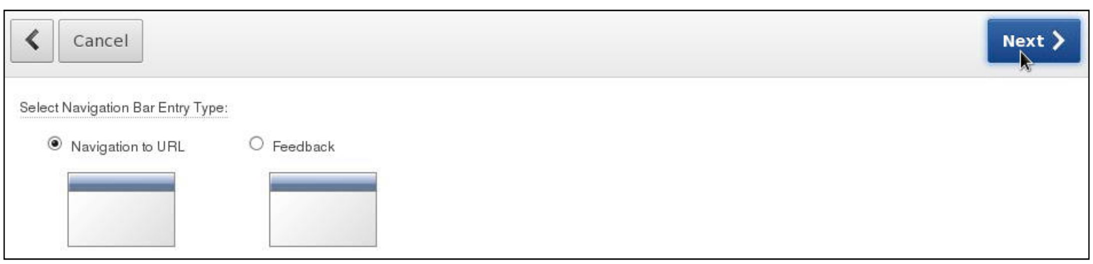


3) Click **Create >**.

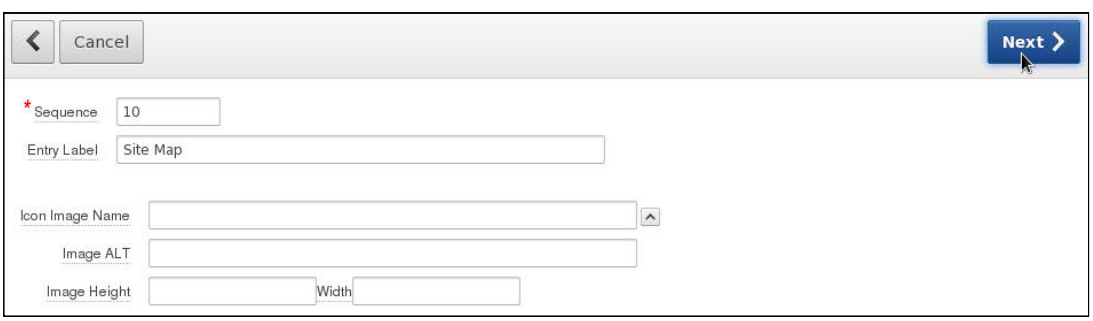


4) Click **Next >**.

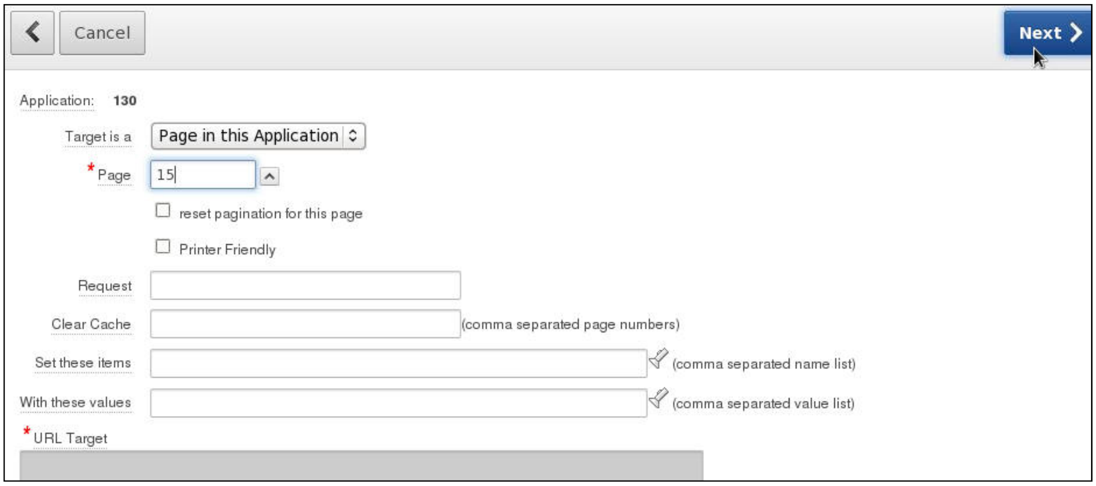
5) Accept default **Navigation to URL** and click **Next >**.



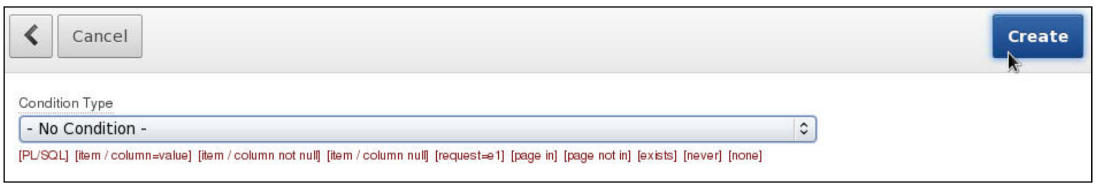
6) Enter **`Site Map`** for Entry Label and click **Next >**.



7) Select your Site Map page number for Page and click **Next >**.

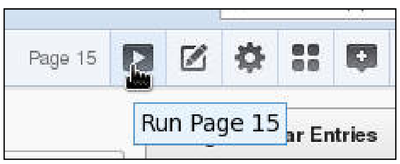


8) Click **Create**.



e. Run the page and click one of the page names to view the corresponding page.

1) Click **Run Page**.



2) Click one of the Page Names to view the corresponding page.

Solution 15-4: Enforcing Authorization in Site Map

Overview

In this practice, you add a function that determines whether you are authorized to see a particular page in your site map. You also change the Classic Report query for the Site Map to make sure that the page is selected only if you are authorized to see it.

Steps

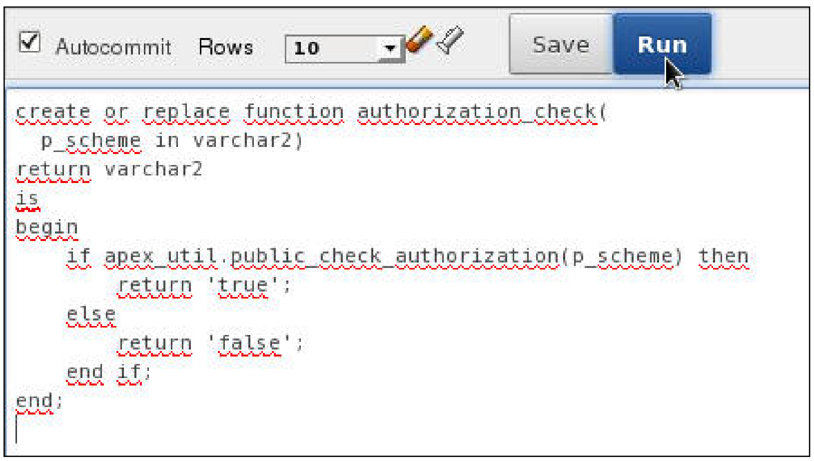
a. Navigate to **SQL Workshop > SQL Commands**. Copy and paste the following code (located in the `/home/oracle/labs/files/lab15_4_1.txt` file) to create the `authorization_check` function.

```
create or replace function authorization_check(
  p_scheme in varchar2)
return varchar2
is
begin
    if apex_util.public_check_authorization(p_scheme) then
        return 'true';
    else
        return 'false';
    end if;
end;
```

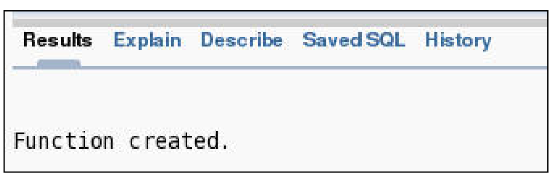
1) Click **SQL Workshop**. Select **SQL Commands**.

2) Paste the following code (located in the/home/oracle/labs/files/lab15_4_1.txt file) to create the **authorization_check** function. Click **Run**.

```
create or replace function authorization_check(
  p_scheme in varchar2)
return varchar2
is
begin
    if apex_util.public_check_authorization(p_scheme) then
        return 'true';
    else
        return 'false';
    end if;
end;
```

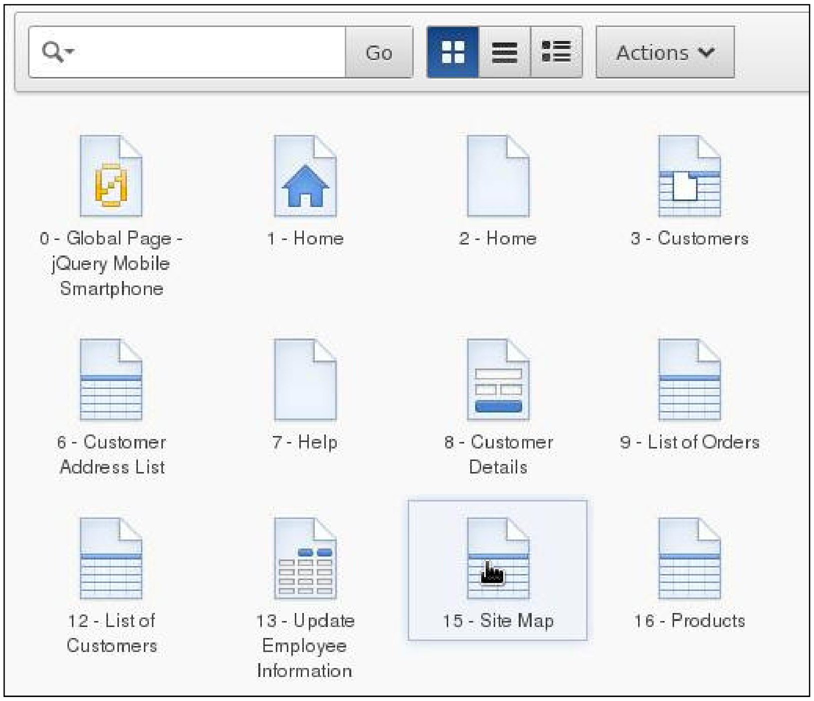


3) The function is created successfully.

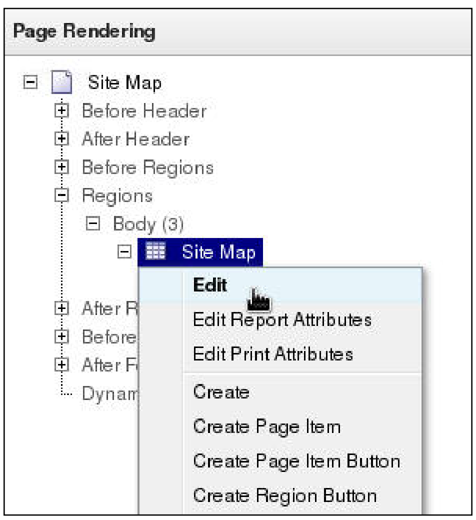


b. For Site Map, change the Source for your Classic Report to the code provided in step 3 below (also located in the home\oracle\labs\files\lab15_4_2.txt file), so that only the pages that are authorized will be shown based on the user logged in.

1) Navigate to your Order Management application and select the **Site Map** page.

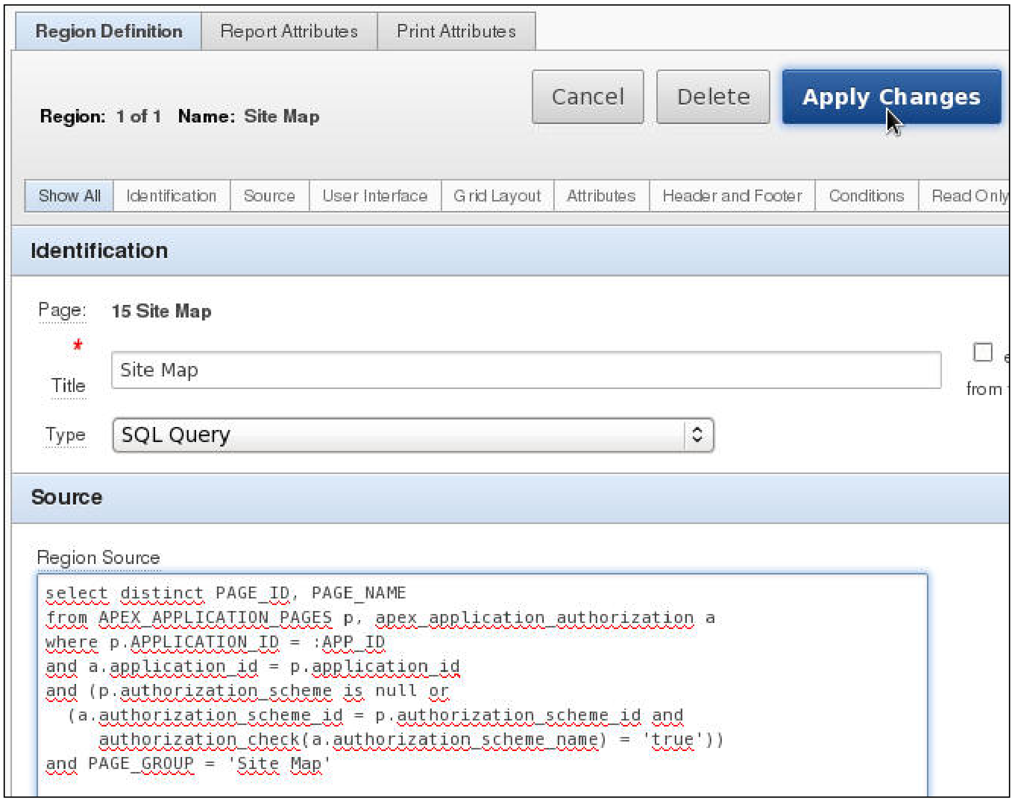


2) Under Regions, right-click **Site Map** and select **Edit**.



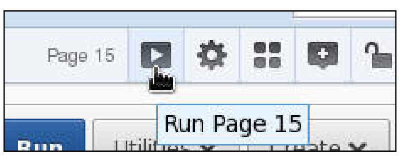
3) Paste the following code (located in the `home\oracle\labs\files\lab15_4_2.txt` file) in the **Source** area and click **Apply Changes**.

```
select distinct PAGE_ID, PAGE_NAME
from APEX_APPLICATION_PAGES p, apex_application_authorization a
where p.APPLICATION_ID = :APP_ID
and a.application_id = p.application_id
and (p.authorization_scheme is null or
  (a.authorization_scheme_id = p.authorization_scheme_id and
     authorization_check(a.authorization_scheme_name) = 'true'))
and PAGE_GROUP = 'Site Map'
```



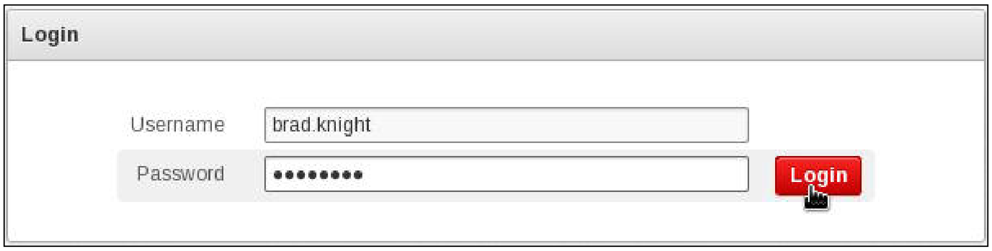
c. You do not want end users or developers to see the Access Control Administration page. Make sure that the authorization schemes for this page is set to **access control - Administrator**. Run the page to see if only an authorized user can view it.

1) Click **Run Page**.

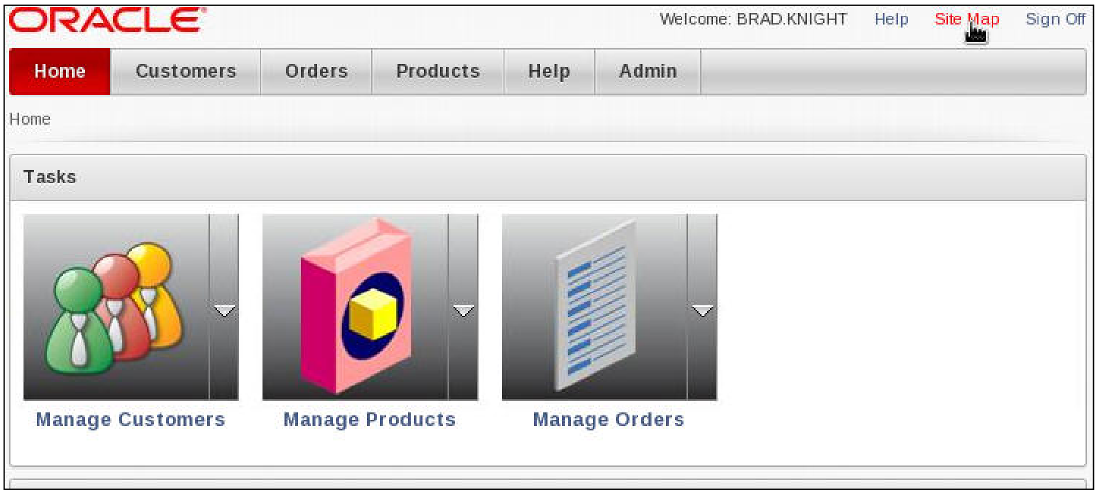


2) Notice that you are currently logged in as `ora<n>`. As a result, you currently see Access Control Administration Page in the site map list. Click **Sign Off**.

3) To check the Site Map as an administrator, log in as **brad.knight** with **welcome1** as the password. Click **Login**.



4) In the navigation bar, click **Site Map**.



5) Brad Knight does not have access to the Access Control Administration page, so you no longer see it in the Site Map list. Click **Sign Off**.

6) Click the **Application <*n*>** link on the Developer toolbar.

Practices for Lesson 16: Extending Your Application

Chapter 16

Practices for Lesson 16: Overview

Practices Overview

There are three practices for this lesson. In these practices, you add a Data Upload Wizard, create an upload and download page, and use BLOB data in a report and form.

Practice 16-1: Adding a Data Upload Wizard

Overview

In this practice, you create a series of wizard pages to upload data into the **`TASKS`** table.

Assumptions

You have performed the previous practices.

Tasks

a. You create a series of wizard pages to upload data into the **TASKS** table.

b. Test the upload by using the following data (located in the `/home/oracle/labs/files/lab16_1_1.txt` file). Use the Copy and Paste Delimited Data area.

```
TASK_ID,TASK_NAME,COST,BUDGET,PROJECT_ID,ASSIGNED_TO
1,"HR software needs upgrade",3000,3500,"Maintain Support Systems",Sue Mavris
101,"Plan rollout schedule",120,120,"Public Website",Valli Pataballa
102,"Apply Billing System updates",500,500,"Maintain Support Systems",Sue
Mavris
3,"Investigate new Virus Protection software",150,200,”Maintain Support
Systems",David Austin
```

Practice 16-2: Creating an Upload and Download Page

Overview

In this practice, you create a form in an HTML region with a file upload item and a button. The button will submit the page and return the user to the same page. Also, you create a report on the document table that has links to download documents. Provide links to download the documents in the report.

Assumptions

You have performed the previous practices.

Tasks

a. Create a blank page called Upload Download Files with an HTML region called Submit File that contains a File Browse item type on the source type **`WWV_FLOW_FILES`**. Create a Submit button and upload the `oralogo.gif` file from the `/home/oracle/labs/files` directory.

   Note that every time you upload a script in SQL Workshop or upload a file in Shared Components for an application, the reference to the file is placed in the **`WWV_FLOW_FILES`** table.

b. Create an Interactive Report subregion called Uploaded Files with the following SQL query (located in `/home/oracle/labs/files/lab16_2_1.txt`):

```
SELECT id, filename FROM APEX_APPLICATION_FILES order by created_on desc
```

c. Modify the report to create a link on the ID column to download the file.

Practice 16-3: Using BLOB Data in a Report and Form

Overview

In this practice, you add binary large object (BLOB) columns to your `OEHR_EMPLOYEES` table, create a form with a report, add a thumbnail image to your report, and add a delete item region to the form.

Assumptions

You have performed the previous practices.

Tasks

a. Upload and run the **`addblob.sql`** script contained in your **`/home/oracle/labs/files`** directory to alter the `OEHR_CUSTOMERS` table with the BLOB columns.
b. Add the Photo column to the Customer Details page.
c. Change the Customers Report to include the Photo column.
d. Add a Delete Image region by creating a new HTML region called Photo Image.
e. Create a Display Image item in the Photo Image region called **`P<n>`**`_IMAGE` that shows the image.
f. Create a delete button called DELETE_IMAGE. Create a PL/SQL process called "delete image" that is invoked to perform the delete. Run the page.

Solution 16-1: Adding a Data Upload Wizard

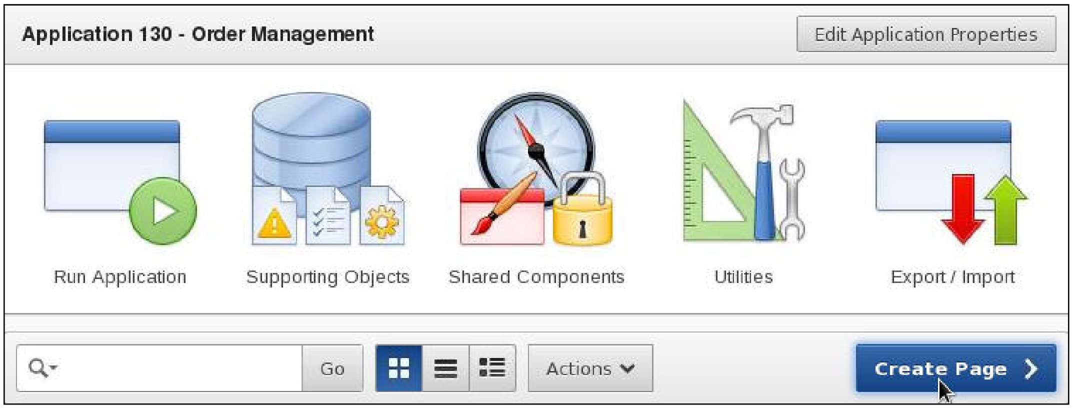
Overview

In this practice, you create a series of wizard pages to upload data into the **TASKS** table.

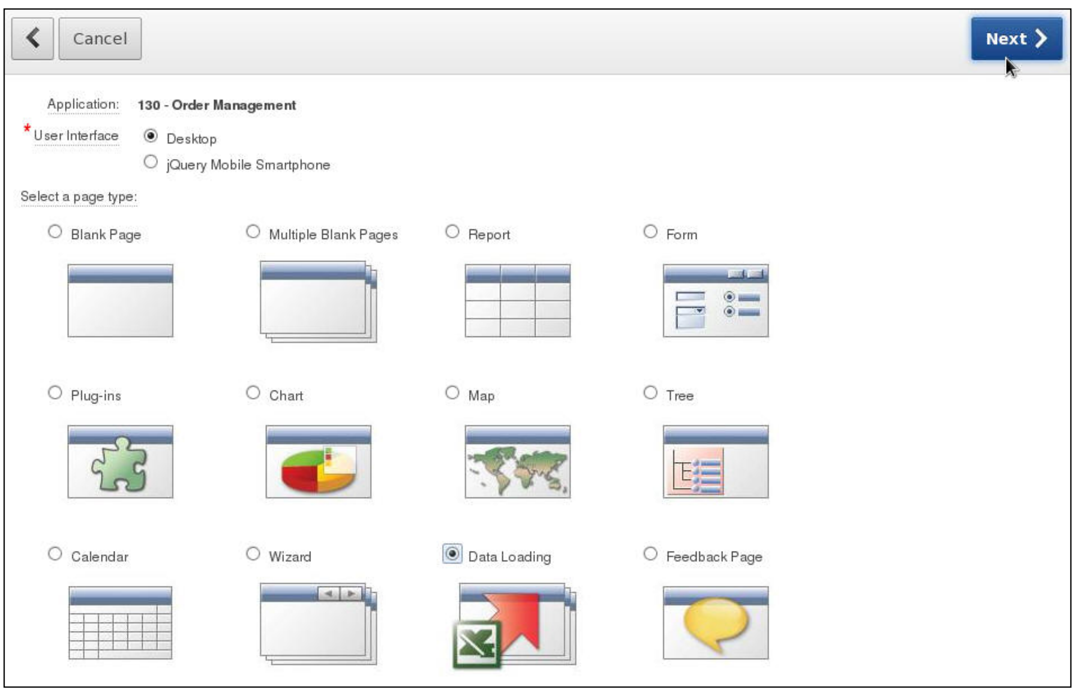
Tasks

a. You create a series of wizard pages to upload data into the **TASKS** table.

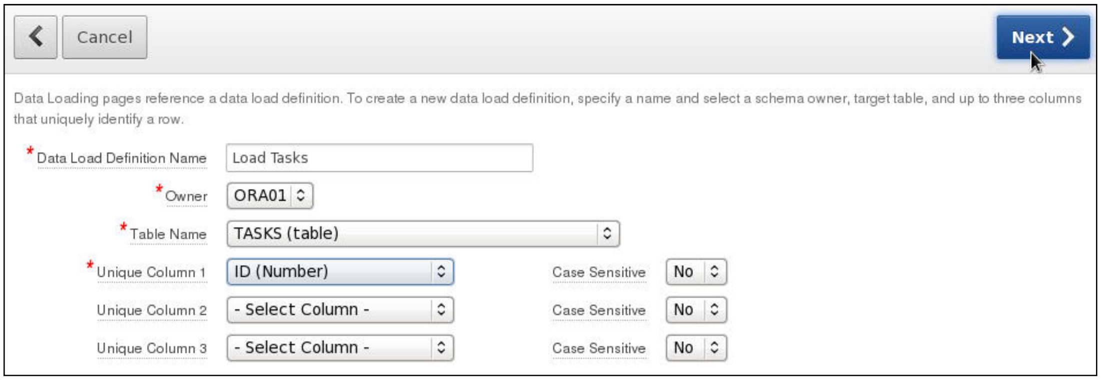
1) Navigate to the **Order Management** application home page and click **Create Page >**.



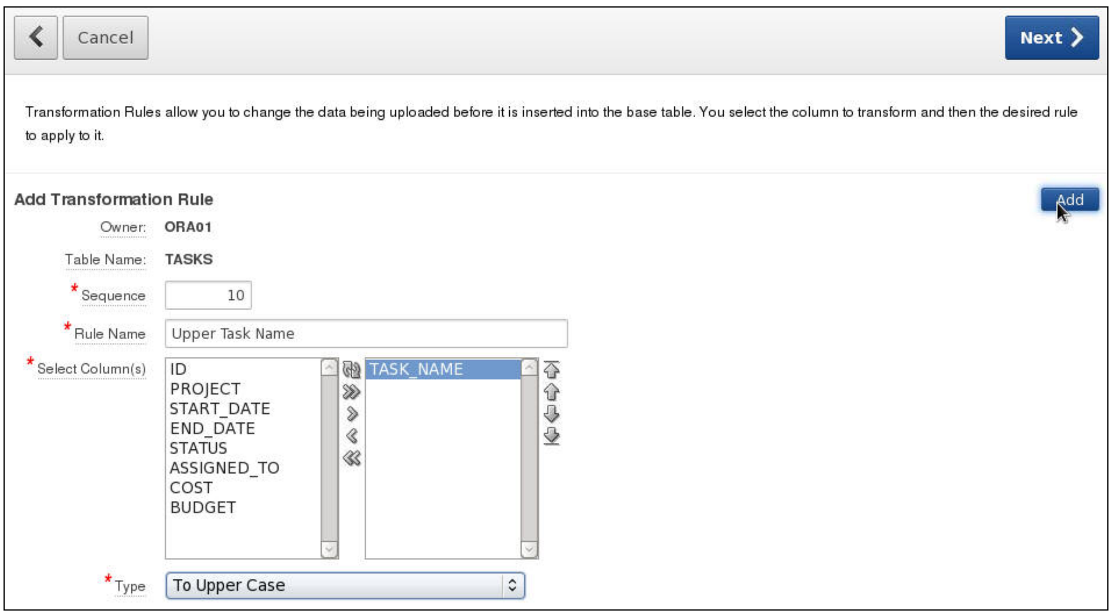
2) Select the **Data Loading** page type and click **Next >**.



3) Enter `Load Tasks` for Data Load Definition Name, select `TASKS` for Table Name, and select `ID` for Unique Column 1. Then click **Next >**.

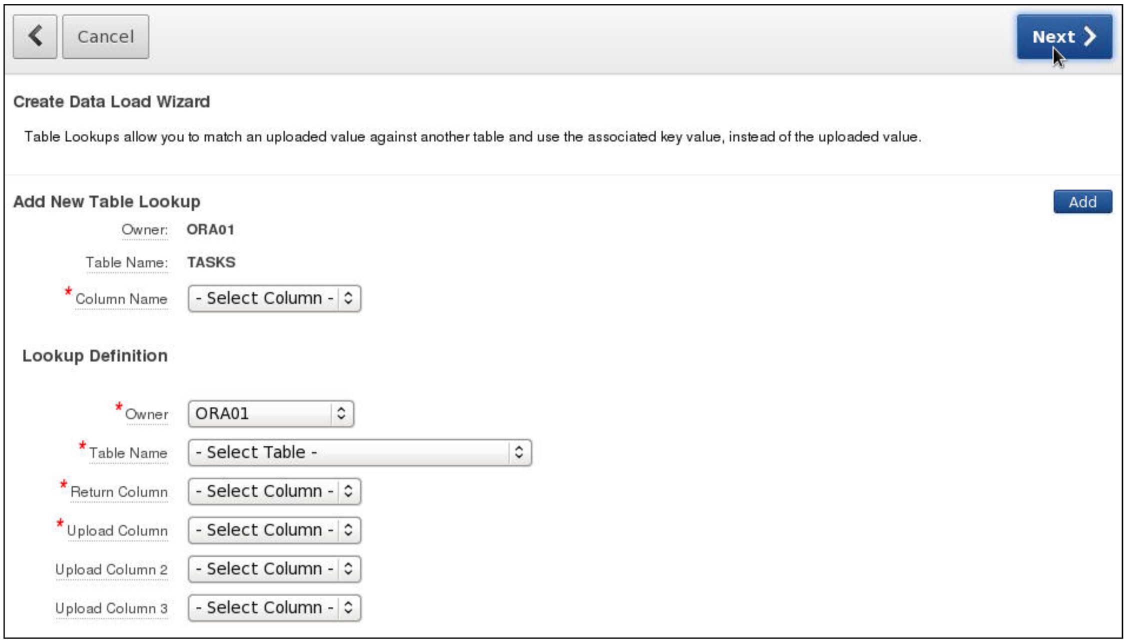


4) You want to create a rule that will change the Task Name to uppercase. Enter **Upper Task Name** for Rule Name, select `TASK_NAME` for Column Name, and select **To Upper Case** for Type. Then click **Add**.

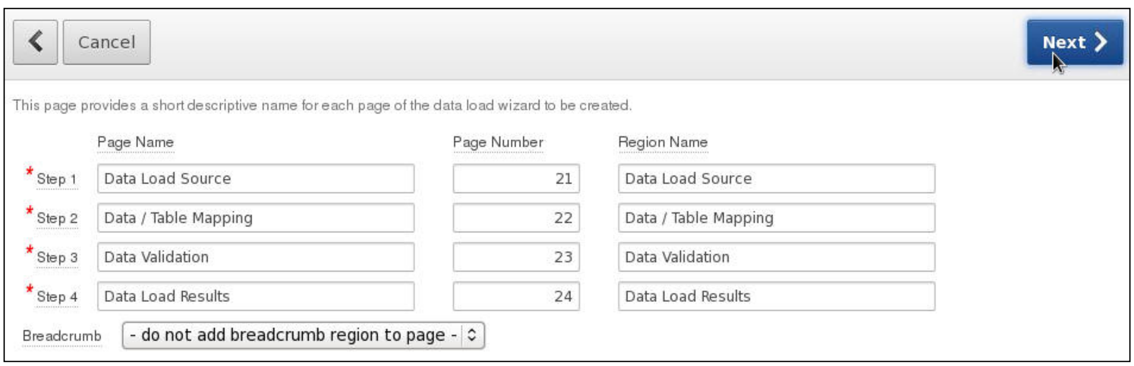


5) Click **Next >**.

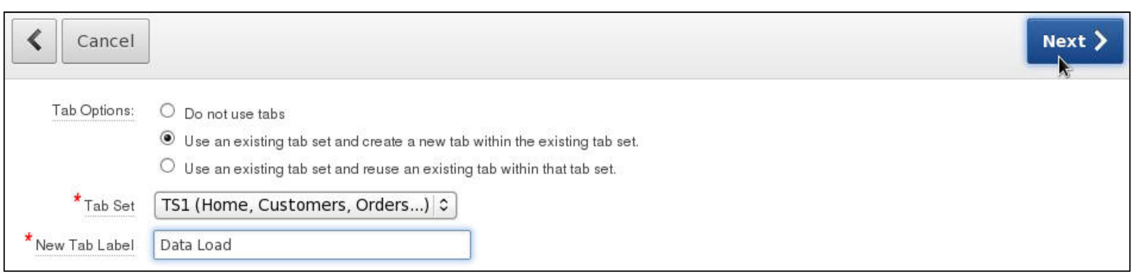
6) You do not want to create a table lookup. Click **Next >**.



7) Accept the default pages that will be created and click **Next >**.

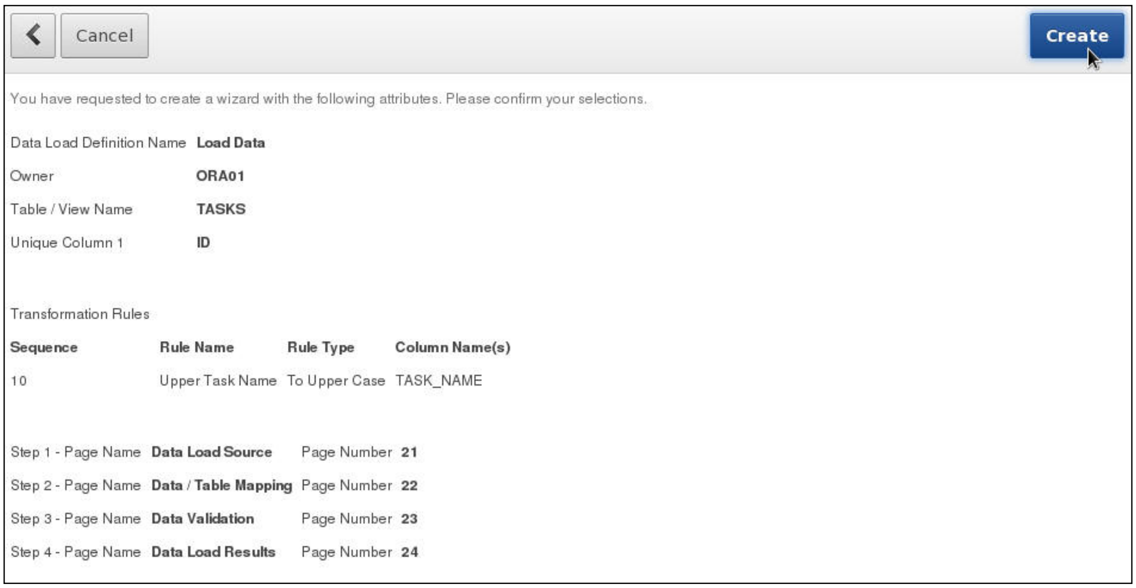


8) You want to create a new tab. Select **Use an existing tab set and create a new tab within the existing tab set** and enter **Data Load** for the New Tab Label and click **Next >**.

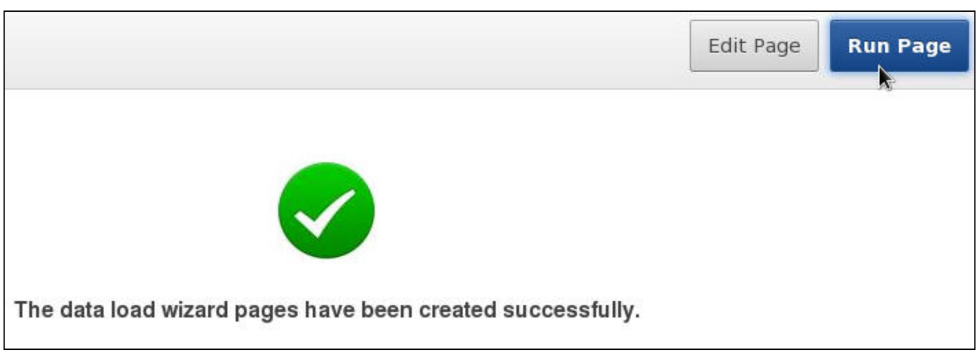


9) You need to specify which page to go to when you press Cancel. Enter `1` for the Home Page for "Branch to Page" for both the Cancel and Finish buttons. Then click **Next >**.

10) You are ready to create the Data Load Wizard pages. Click **Create**.



11) Click **Run Page**.

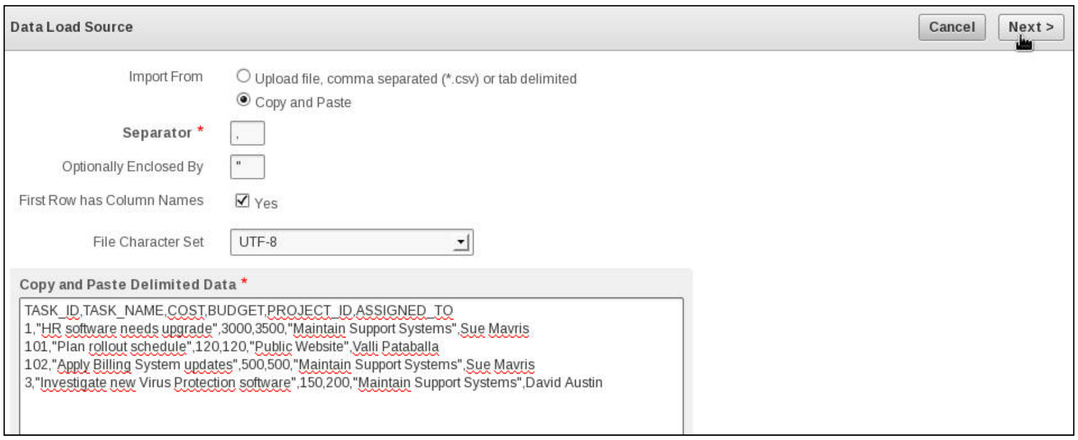


b. Test the upload by using the following data (located in the `/home/oracle/labs/files/lab16_1_1.txt` file). Use the Copy and Paste Delimited Data area.

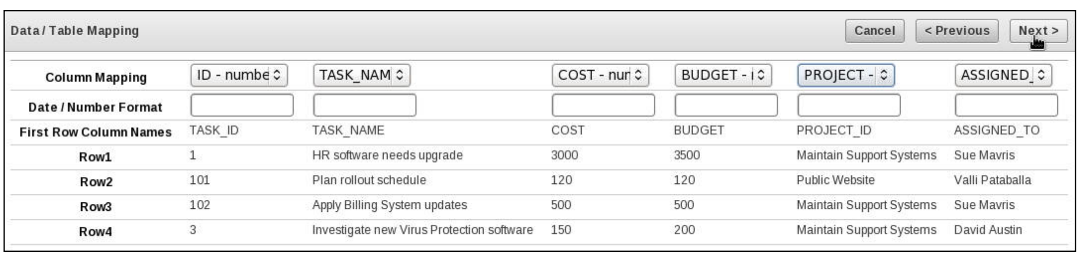
```
TASK_ID,TASK_NAME,COST,BUDGET,PROJECT_ID,ASSIGNED_TO
1,"HR software needs upgrade",3000,3500,"Maintain Support Systems",Sue Mavris
101,"Plan rollout schedule",120,120,"Public Website",Valli Pataballa
102,"Apply Billing System updates",500,500,"Maintain Support Systems",Sue
Mavris
3,"Investigate new Virus Protection software",150,200,"Maintain Support
Systems",David Austin
```

1) The first wizard page is where you specify the Data Load Source. You want to Copy and Paste the following text (located in the `/home/oracle/labs/files/lab16_1_1.txt` file) into the Copy and Paste Delimited Data area. Change the Separator to a **','** (comma) and select the **Yes** check box for **First Row has Column Names**. Then click **Next >**.

```
TASK_ID,TASK_NAME,COST,BUDGET,PROJECT_ID,ASSIGNED_TO
1,"HR software needs upgrade",3000,3500,"Maintain Support Systems",Sue Mavris
101,"Plan rollout schedule",120,120,"Public Website",Valli Pataballa
102,"Apply Billing System updates",500,500,"Maintain Support Systems",Sue Mavris
3,"Investigate new Virus Protection software",150,200,”Maintain Support Systems",David Austin
```

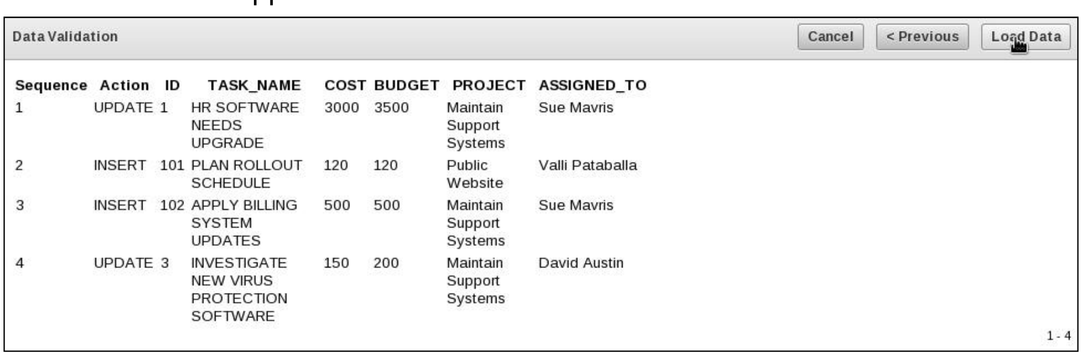


2) The Data / Table Mappings are displayed. Notice that the Column Name for some of the columns are not selected. This is because the names of the columns in the file do not match the names of the columns in the database. Select the **ID** and **PROJECT** columns from their respective pull downs and click **Next >**.



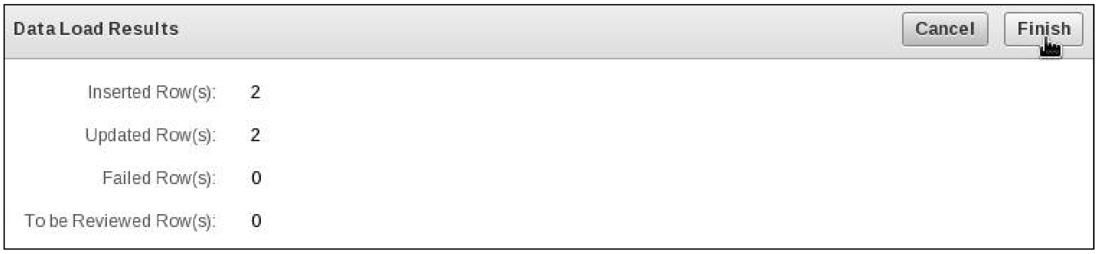
Column Mapping	ID - numbe	TASK_NAM	COST - nur	BUDGET - I	PROJECT -	ASSIGNED_
Date / Number Format						
First Row Column Names	TASK_ID	TASK_NAME	COST	BUDGET	PROJECT_ID	ASSIGNED_TO
Row1	1	HR software needs upgrade	3000	3500	Maintain Support Systems	Sue Mavris
Row2	101	Plan rollout schedule	120	120	Public Website	Valli Pataballa
Row3	102	Apply Billing System updates	500	500	Maintain Support Systems	Sue Mavris
Row4	3	Investigate new Virus Protection software	150	200	Maintain Support Systems	David Austin

3) The Data Validation page is displayed. This page displays the data that will be inserted and/or updated into the database. Notice that the **TASK_NAME** was transformed to all uppercase. Click **Load Data**.



Sequence	Action	ID	TASK_NAME	COST	BUDGET	PROJECT	ASSIGNED_TO
1	UPDATE	1	HR SOFTWARE NEEDS UPGRADE	3000	3500	Maintain Support Systems	Sue Mavris
2	INSERT	101	PLAN ROLLOUT SCHEDULE	120	120	Public Website	Valli Pataballa
3	INSERT	102	APPLY BILLING SYSTEM UPDATES	500	500	Maintain Support Systems	Sue Mavris
4	UPDATE	3	INVESTIGATE NEW VIRUS PROTECTION SOFTWARE	150	200	Maintain Support Systems	David Austin

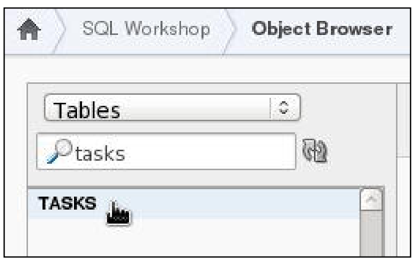
4) Notice that two rows were inserted and two rows were updated. Click **Finish**.



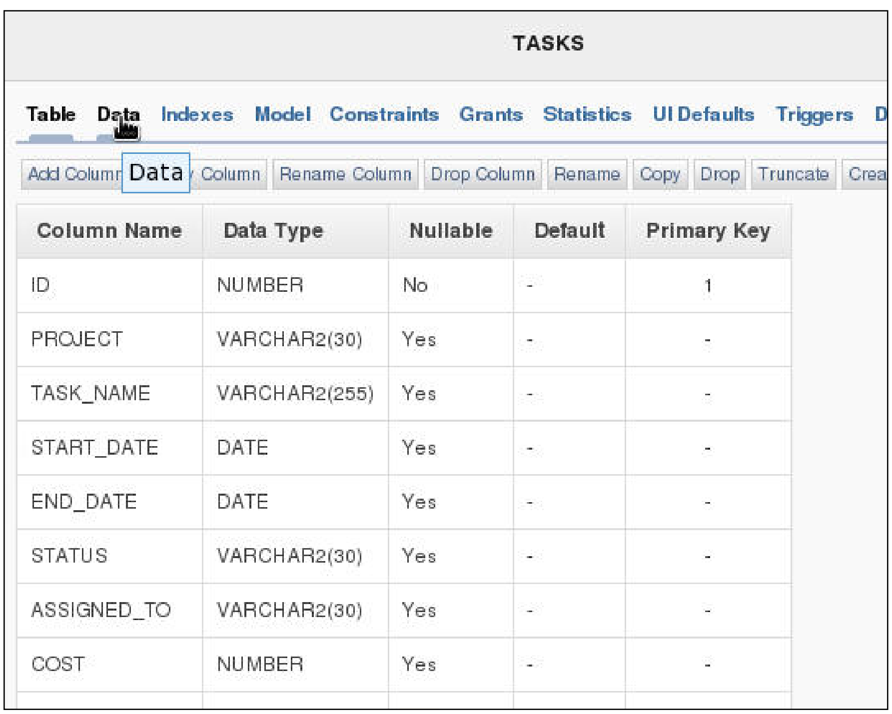
5) Click **Application<*n*>**.

6) Click **SQL Workshop** and select **Object Browser**.

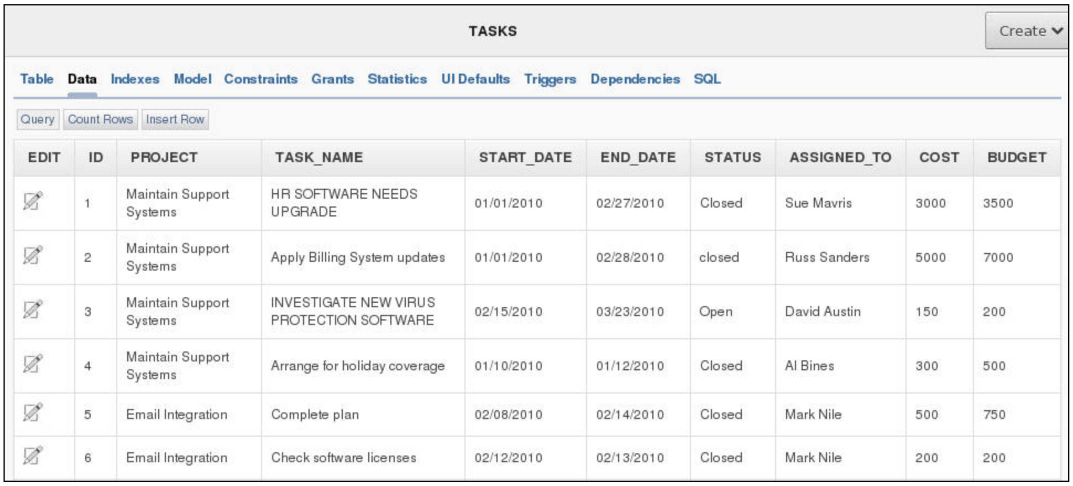
7) In the search field for tables, enter `tasks`. Select the **Tasks** table.



8) The tasks table definition is displayed. Click the **Data** tab.



9) The newly inserted and updated rows are displayed.



EDIT	ID	PROJECT	TASK_NAME	START_DATE	END_DATE	STATUS	ASSIGNED_TO	COST	BUDGET
	1	Maintain Support Systems	HR SOFTWARE NEEDS UPGRADE	01/01/2010	02/27/2010	Closed	Sue Mavris	3000	3500
	2	Maintain Support Systems	Apply Billing System updates	01/01/2010	02/28/2010	closed	Russ Sanders	5000	7000
	3	Maintain Support Systems	INVESTIGATE NEW VIRUS PROTECTION SOFTWARE	02/15/2010	03/23/2010	Open	David Austin	150	200
	4	Maintain Support Systems	Arrange for holiday coverage	01/10/2010	01/12/2010	Closed	Al Bines	300	500
	5	Email Integration	Complete plan	02/08/2010	02/14/2010	Closed	Mark Nile	500	750
	6	Email Integration	Check software licenses	02/12/2010	02/13/2010	Closed	Mark Nile	200	200

Solution 16-2: Creating an Upload and Download Page

Overview

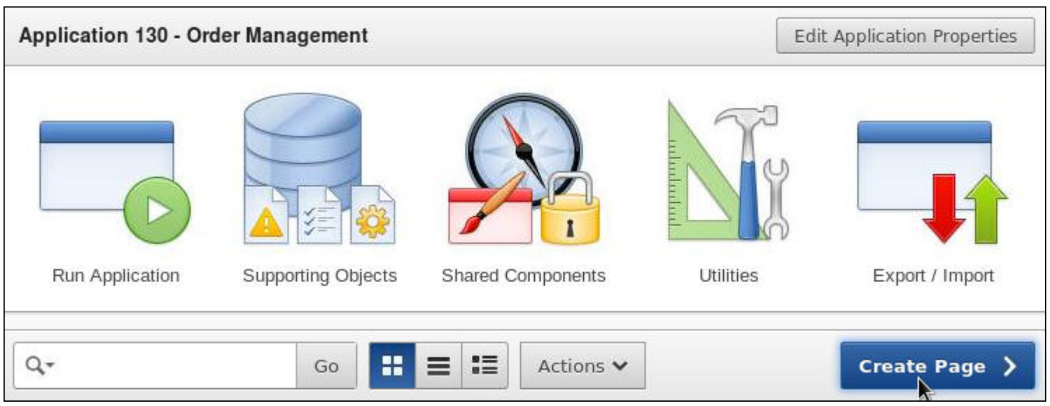
In this practice, you create a form in an HTML region with a file upload item and a button. The button will submit the page and return the user to the same page. Also, you create a report on the document table that has links to download documents. Provide links to download the documents in the report.

Tasks

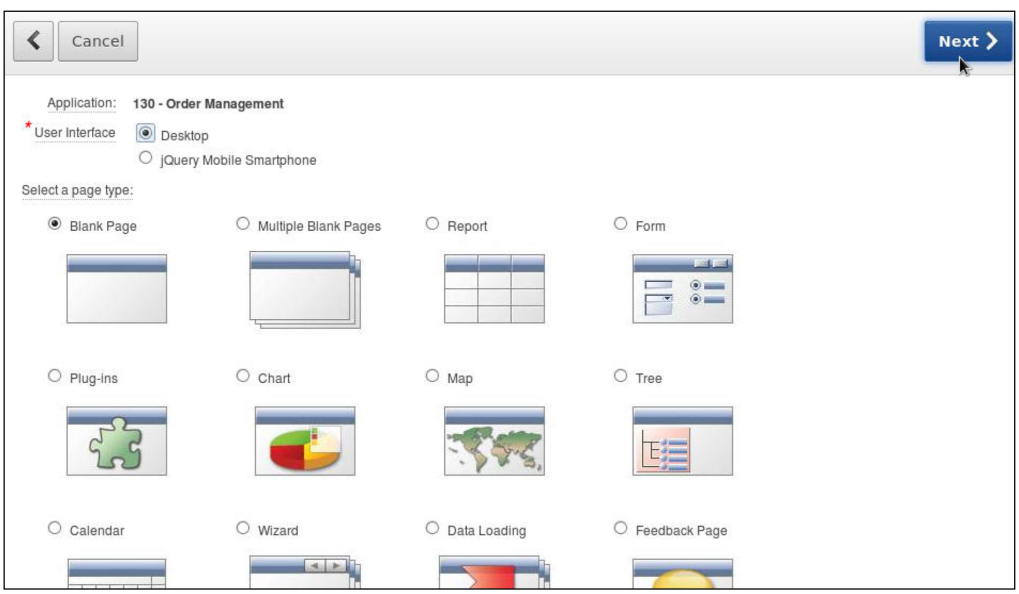
a. Create a blank page called **Upload Download Files** with an HTML region called Submit File that contains a File Browse item type on the source type `WWV_FLOW_FILES`. Create a Submit button and upload the `oralogo.gif` file from the `/home/oracle/labs/files` directory.

   Note that every time you upload a script in SQL Workshop or upload a file in Shared Components for an application, the reference to the file is placed in the `WWV_FLOW_FILES` table.

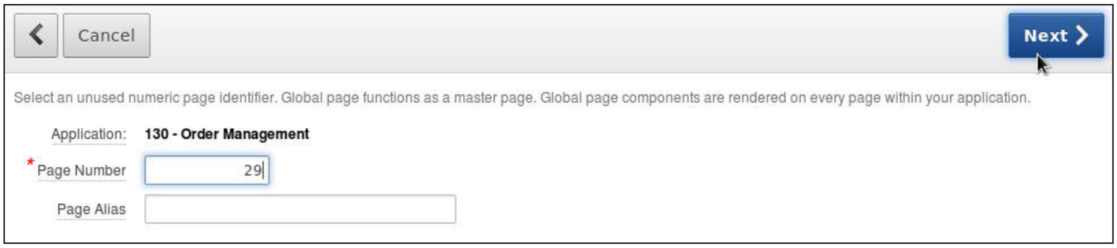
   1) Navigate to the **Order Management** application home page and click **Create Page >**.



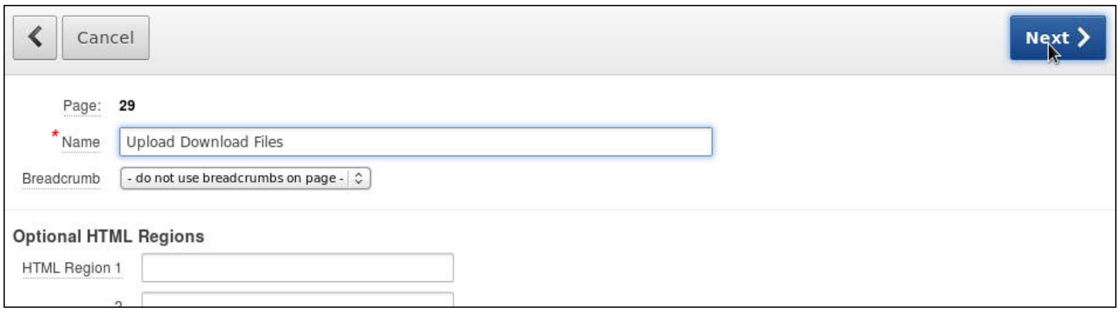
   2) For page type, select the **Blank Page** option and click **Next >**.



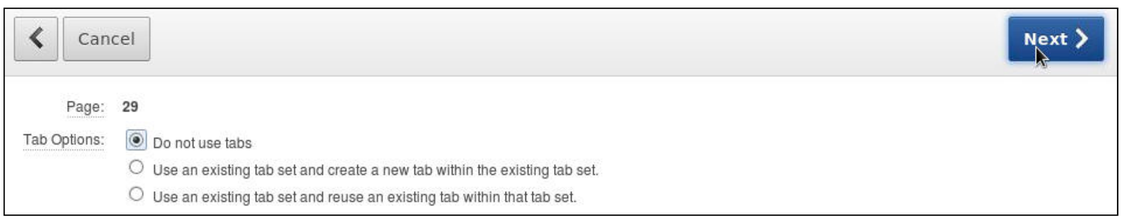
3) Click **Next >**.



4) For Page Name, enter `Upload Download Files` and click **Next >**.



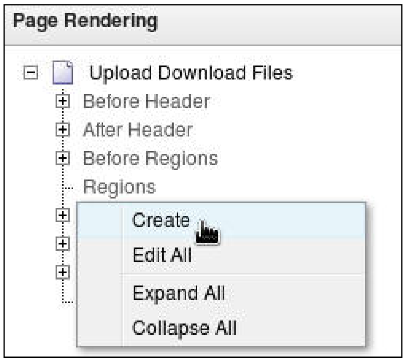
5) Leave the defaults for the Tab Options and click **Next >**.



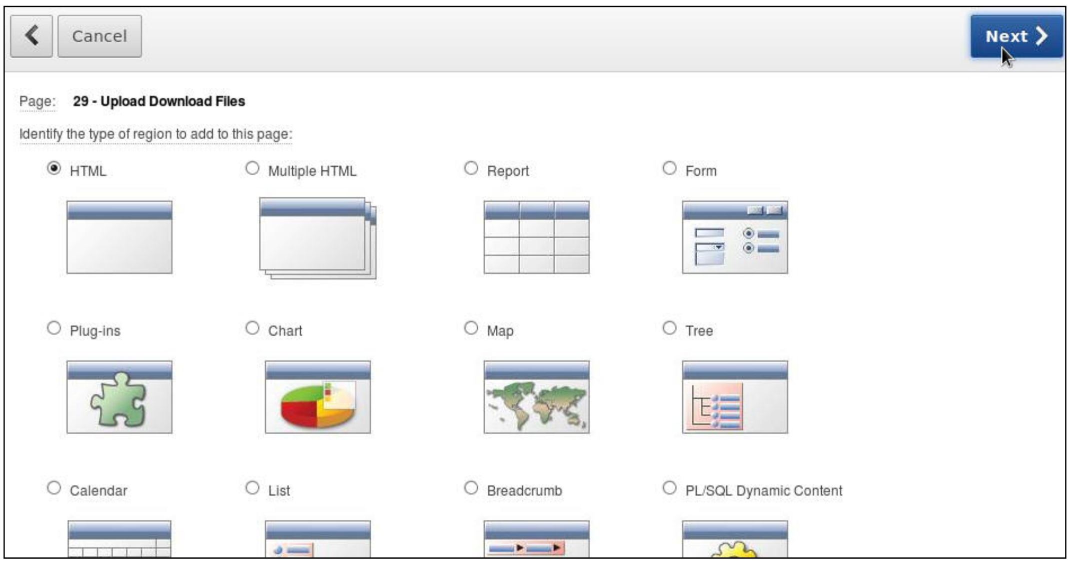
6) Click **Finish**.

7) Click the **Edit Page** icon.

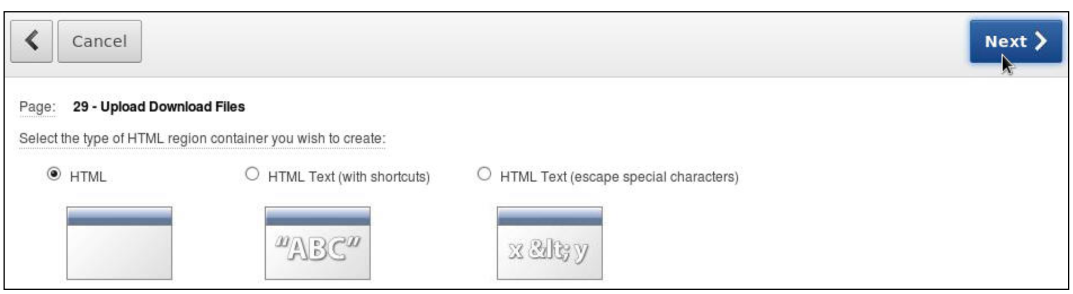
8) Right-click Regions and select **Create**.



9) For Region, select the **HTML** option and click **Next >**.

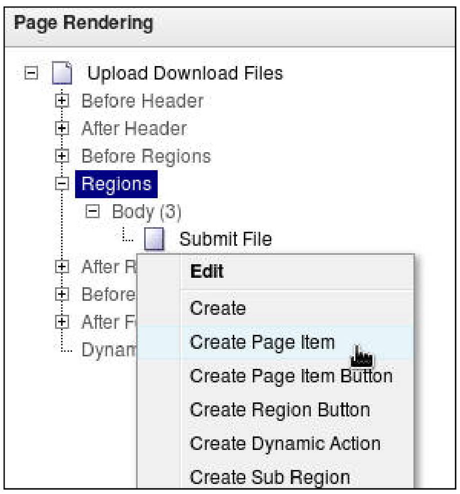


10) Select **HTML** again and click **Next >**.

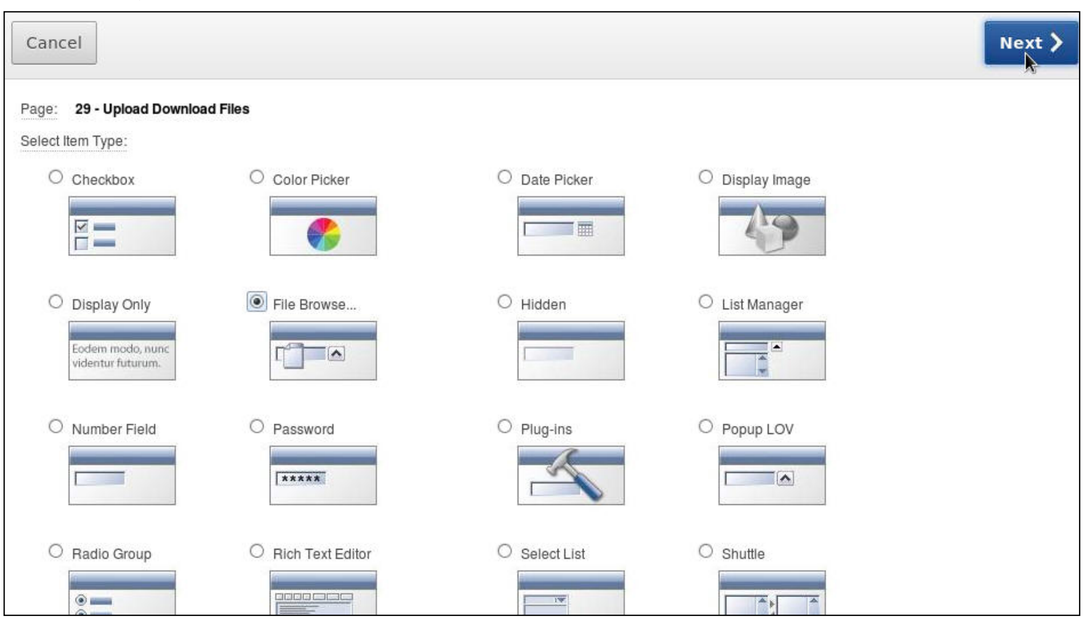


11) For Title, enter `Submit File` and click **Create Region**.

12) Right-click the **Submit File** region and select **Create Page Item**.

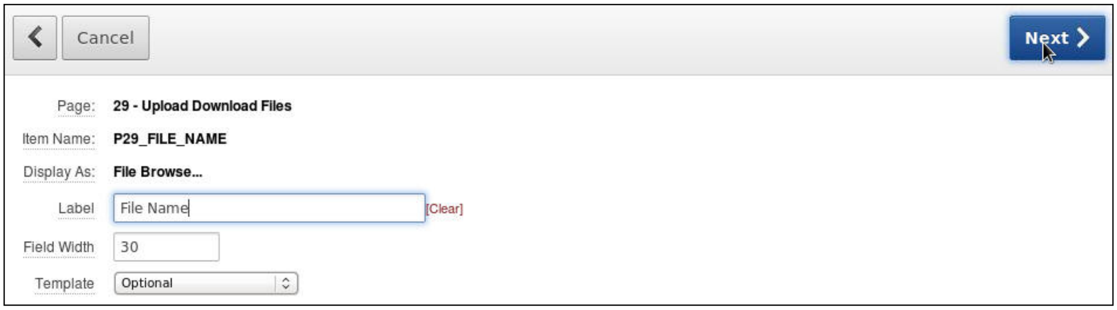


13) For Item Type, select **File Browse** and click **Next >**.

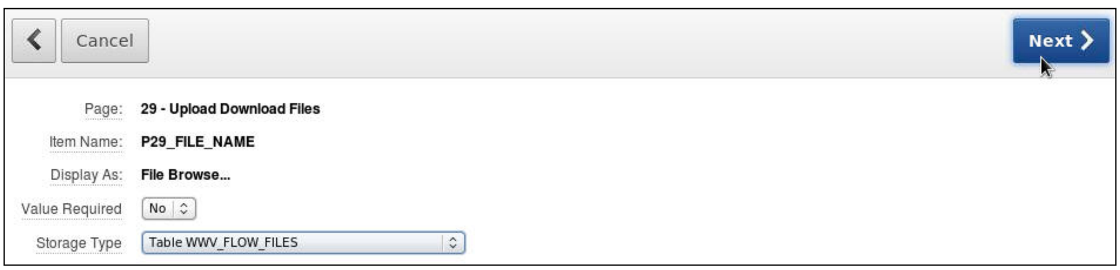


14) For Item Name, enter `P<n>_FILE_NAME` and click **Next >**.

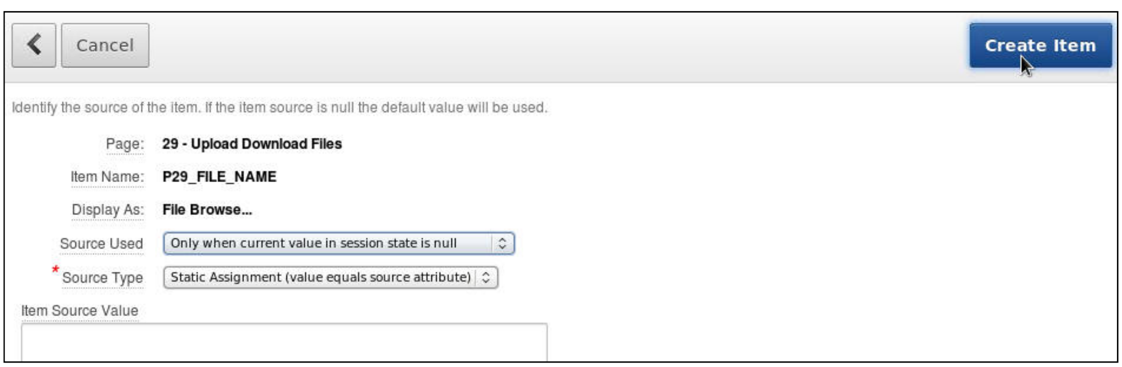
15) Accept the defaults and click **Next >**.



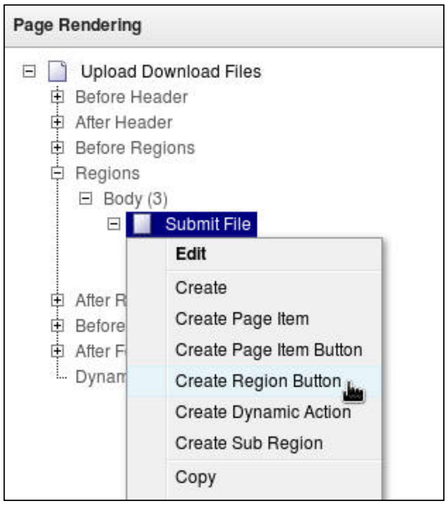
16) Select the **`Table WWV_FLOW_FILES`** Storage Type option. Note that every time you upload a script in SQL Workshop or upload a file in Shared Components for an application, the reference to the file is placed in the `WWV_FLOW_FILES` table. Click **Next >**.



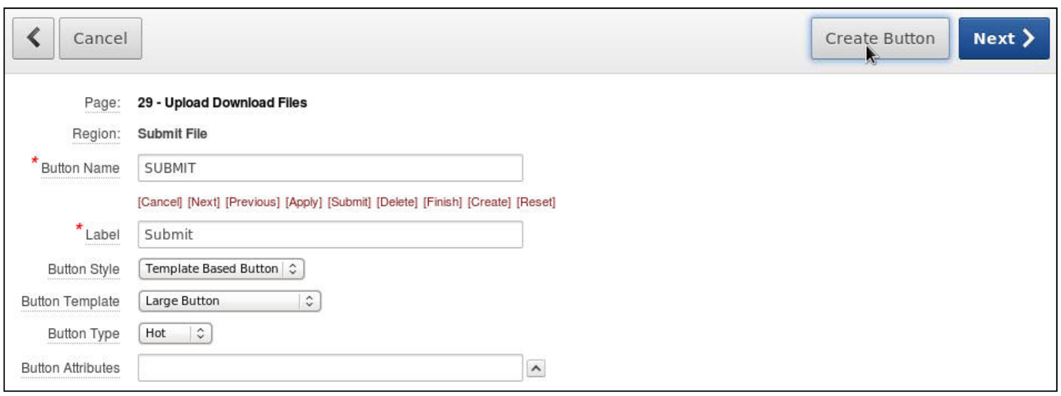
17) Click **Create Item**.



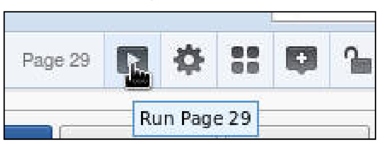
18) You want to create a Submit button that will perform the upload. Right-click the **Submit File** region and select **Create Region Button**.



19) Click the **[Submit]** quick pick, select **Template Based Button** for Button style, **Large Button** for Button Template and **Hot** for Button type. Then click **Create Button**.

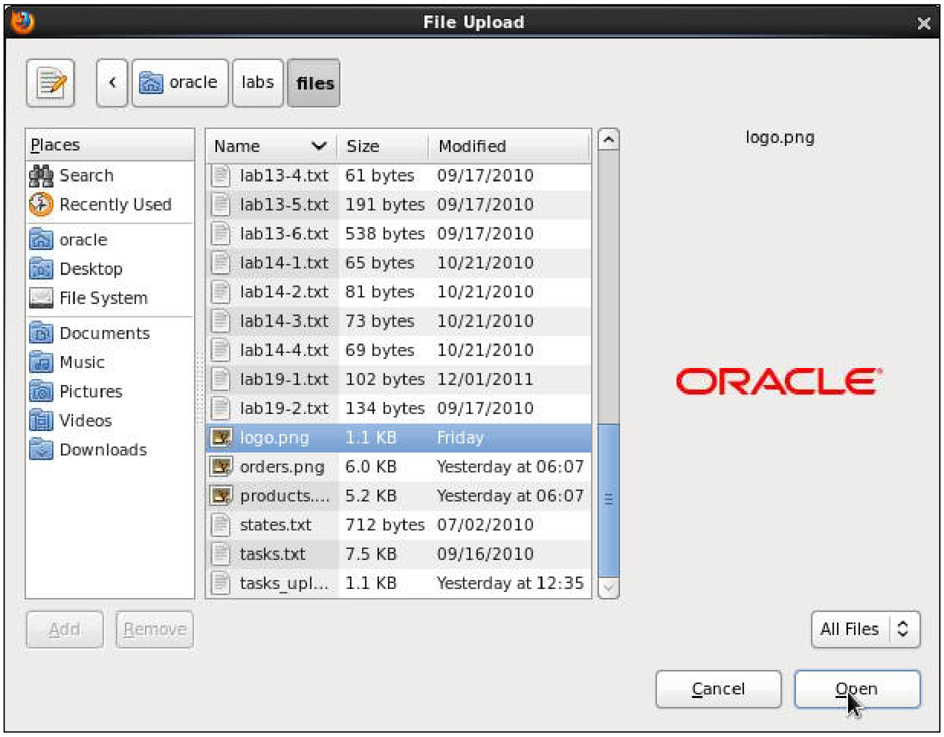


20) Run the page. Click **Run**.



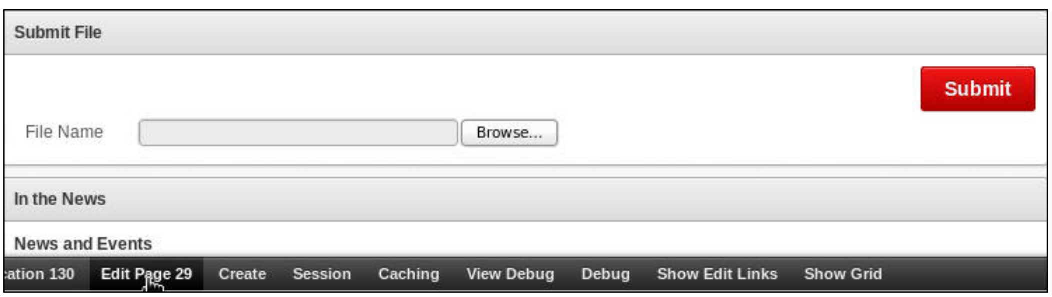
21) Click **Browse**.

22) Select the `/home/oracle/labs/files/logo.png` file and click **Open**.



23) Click **Submit** to upload files.

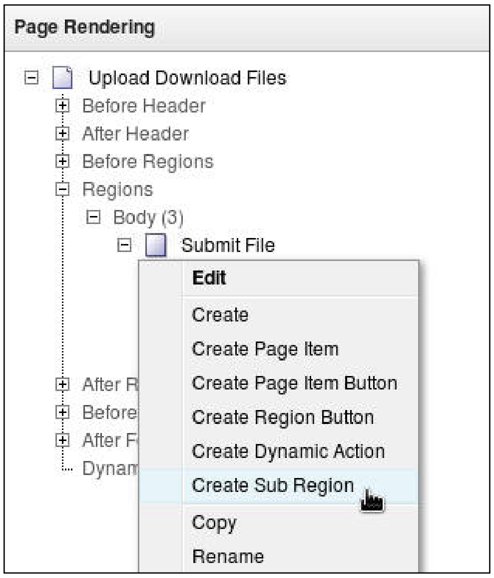
24) The file is submitted. At this point, you want to create a report to show the list of files that are uploaded already. Click the **Edit Page** link on the Developer toolbar.



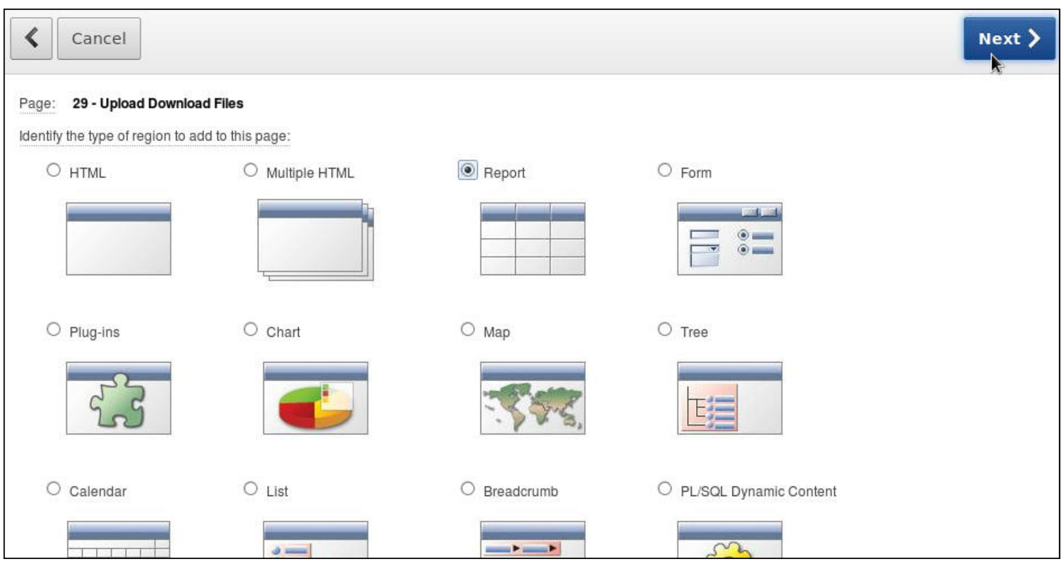
b. Create an Interactive Report subregion called Uploaded Files with the following SQL query (located in `/home/oracle/labs/files/lab16_2_1.txt`):

```
SELECT id, filename FROM APEX_APPLICATION_FILES order by created_on desc
```

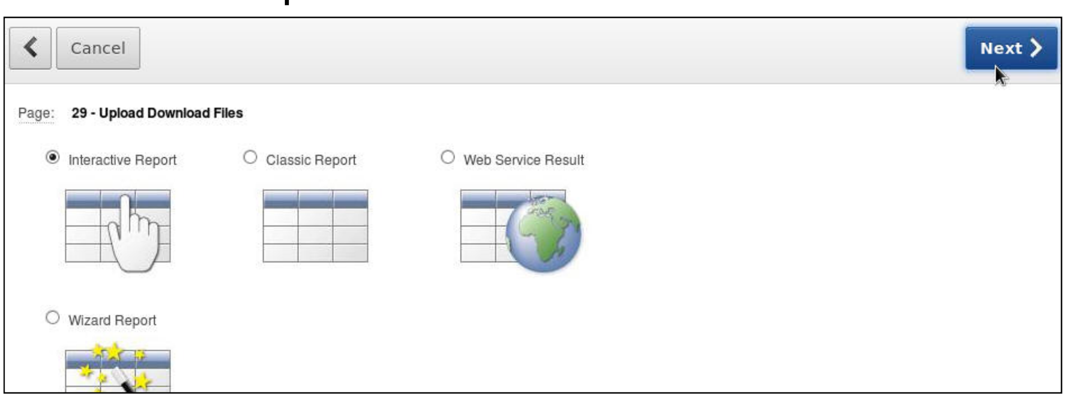
1) You will create the report as a subregion. Right-click the **Submit File** region and select **Create Sub Region**.



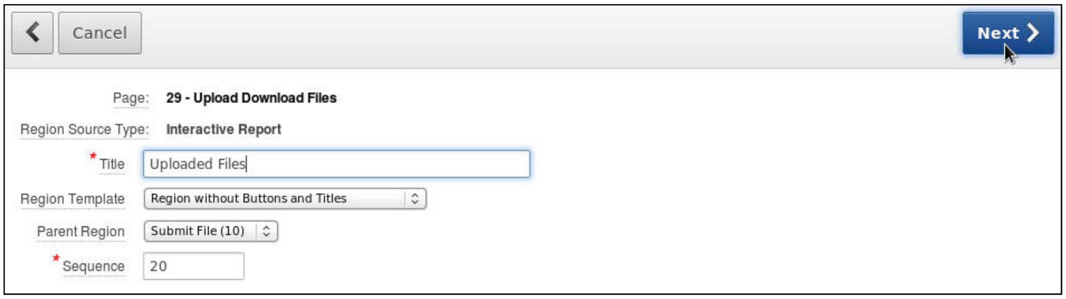
2) For the region type, select **Report** and click **Next >**.



3) Select **Interactive Report** and click **Next >**.

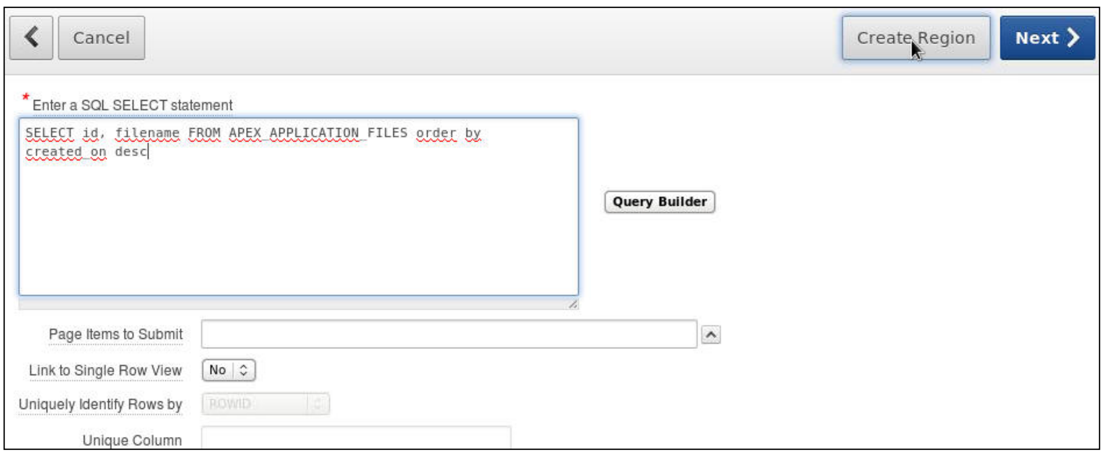


4) For Title, enter **`Uploaded Files`** and click **Next >**.

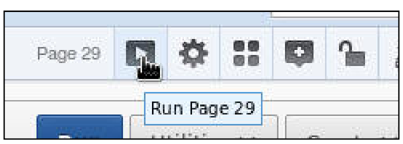


5) When you use the file browse item type, the files that you upload are stored in a table called `wwv_flow_file_objects$`. Every workspace has access to this table through a view called `APEX_APPLICATION_FILES`. Enter the following SQL query (located in `/home/oracle/labs/files/lab16_2_1.txt`) and click **Create Region**.

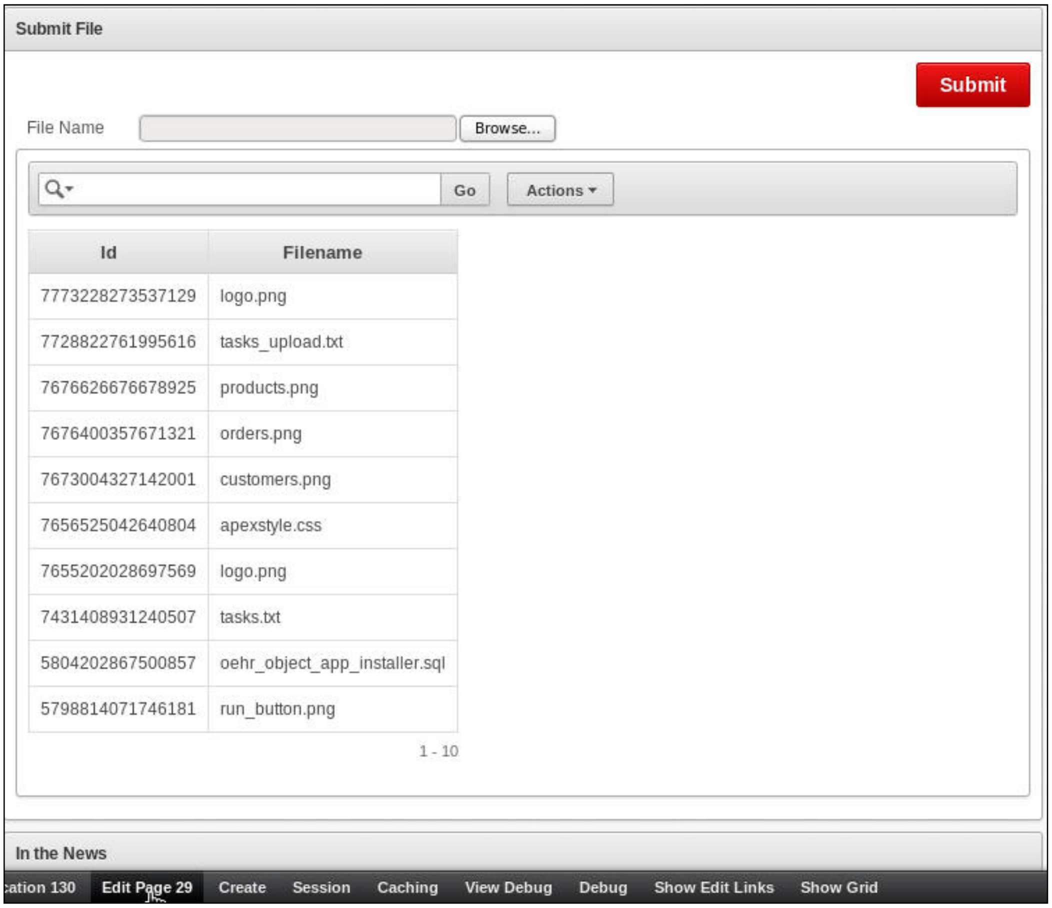
```
SELECT id, filename FROM APEX_APPLICATION_FILES order by created_on desc
```



6) Run the page to see the files that you uploaded. (**Note:** Your files may differ slightly.) Click **Run**.

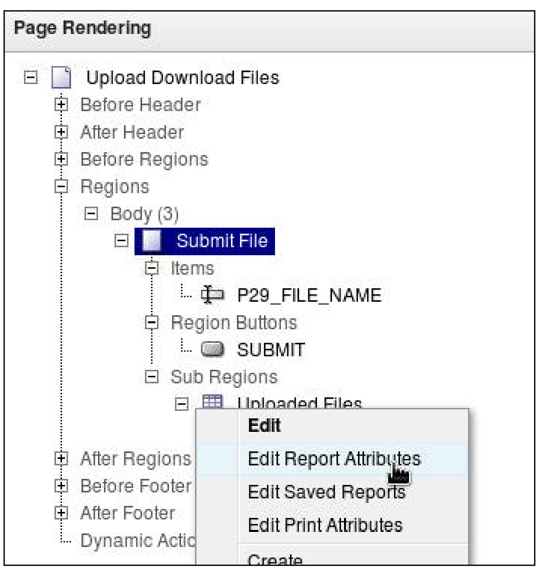


7) The report that you just created shows all the documents that have been uploaded. Note that your list may be different depending on what you uploaded. Next, you provide a link to download the document. Click the **Edit Page** link on the Developer toolbar.

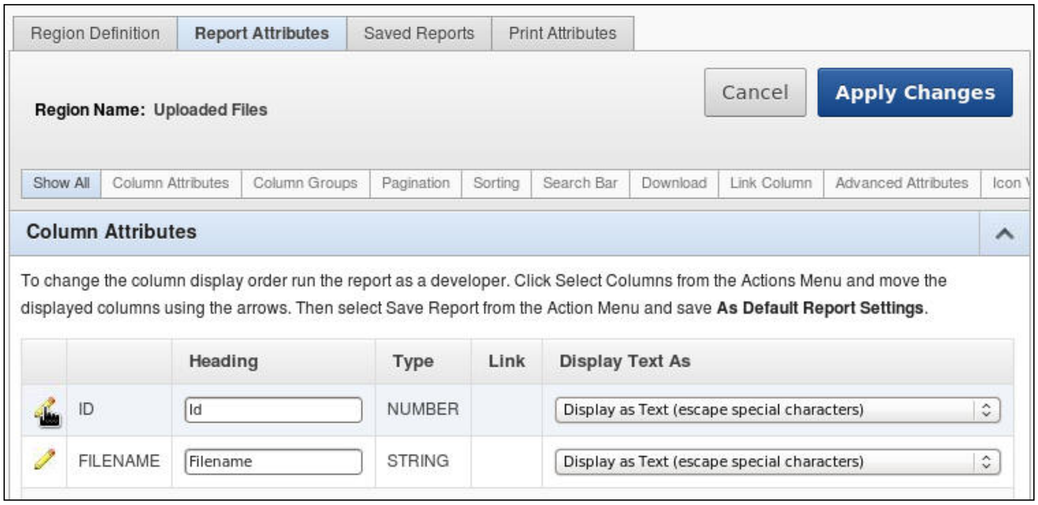


c. Modify the report to create a link on the ID column to download the file.

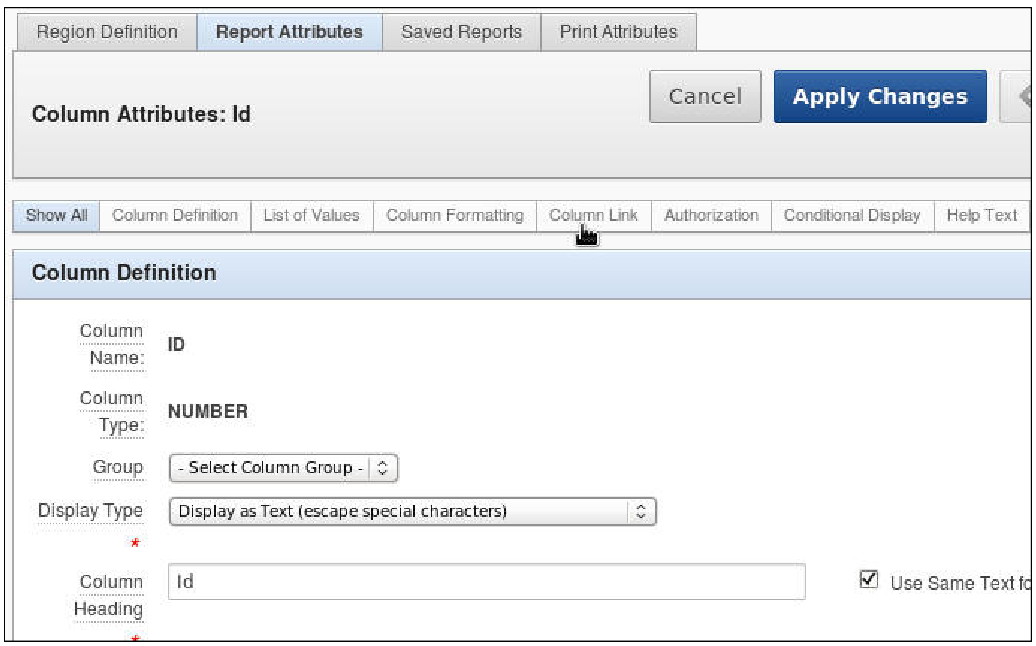
1) Right-click the **Uploaded Files** region and select **Edit Report Attributes**



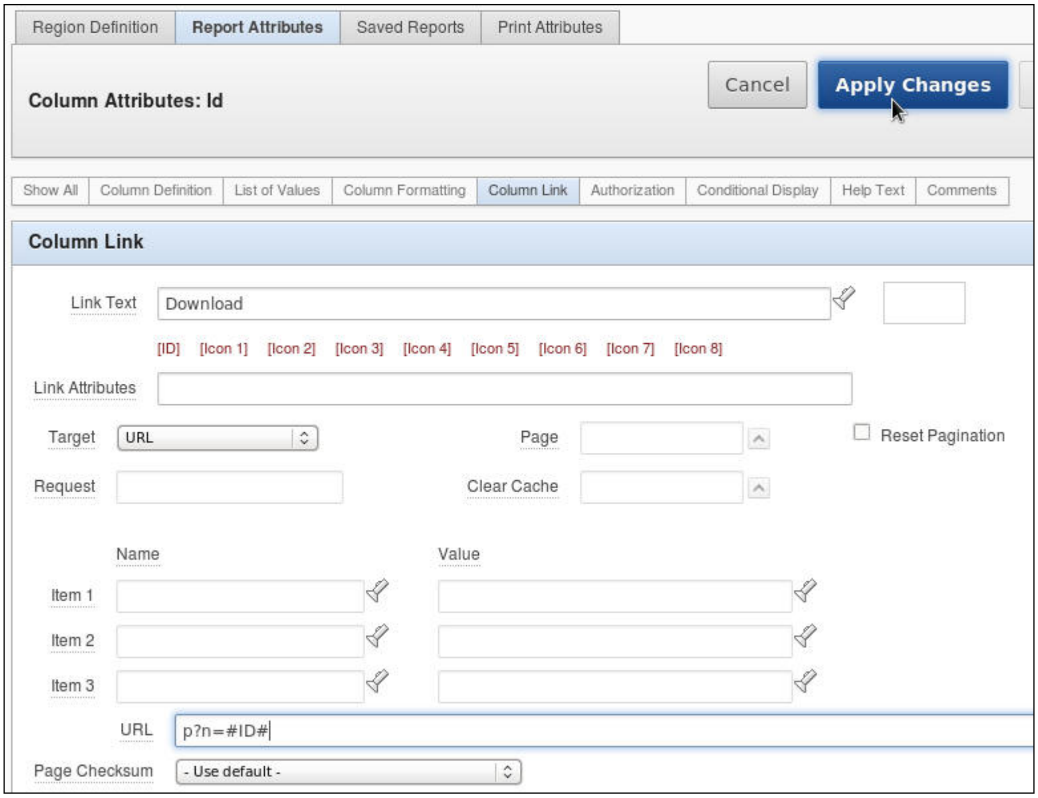
2) In **Column Attributes**, click the **Edit** icon next to the **ID** column.



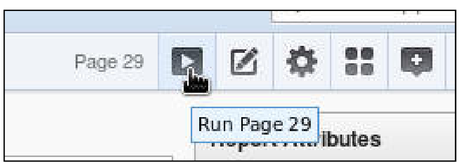
3) Click the **Column Link** tab.



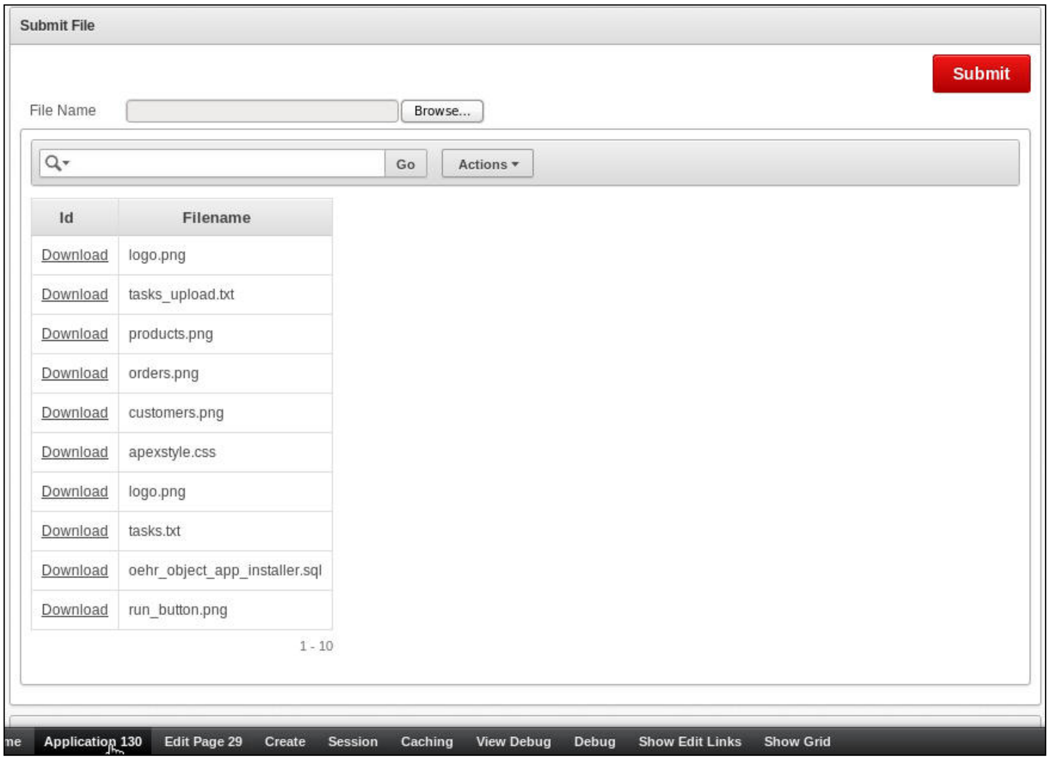
4) In the Link Text field, enter `Download`. From the Target drop-down list, select **URL**. In the URL field, enter the following: **`p?n=#ID#`** and click **Apply Changes**. Note that `#ID#` passes the value contained in the column, where ID is the column alias.



5) Click the **Run Page** icon.



6) The report is displayed with a link to download the file. Click the **Application <*n*>** link on the Developer toolbar.



Solution 16-3: Using BLOB Data in a Report and Form

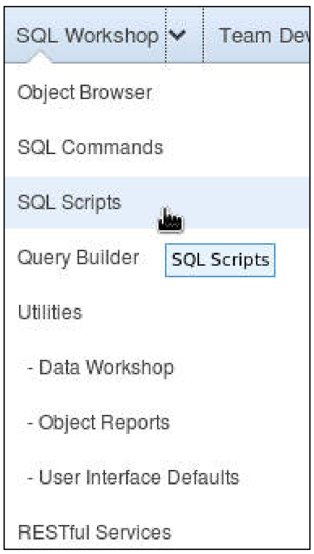
Overview

In this practice, you add binary large object (BLOB) columns to your `OEHR_CUSTOMERS` table, modify the Customer Details form region, add a thumbnail image to your Customer report, and add a delete item region to the Customer Details page.

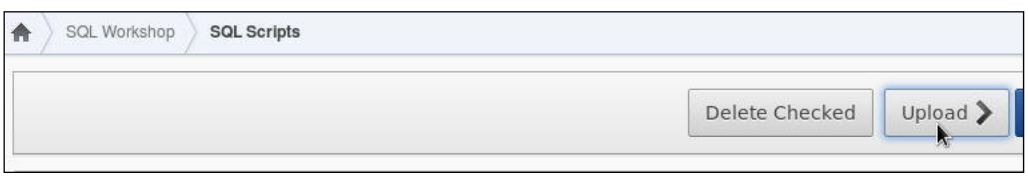
Tasks

a. Upload and run the **`addblob.sql`** script contained in your **`/home/oracle/labs/files`** directory to alter the `OEHR_CUSTOMERS` table with the BLOB columns.

1) Navigate to **SQL Workshop > SQL Scripts**.

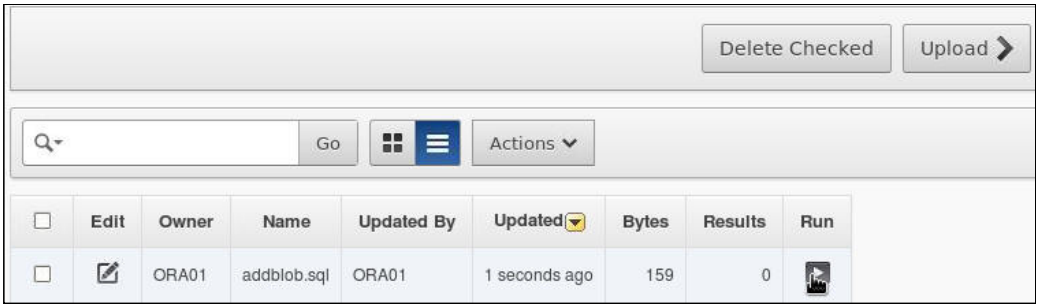


2) Click **Upload >**.

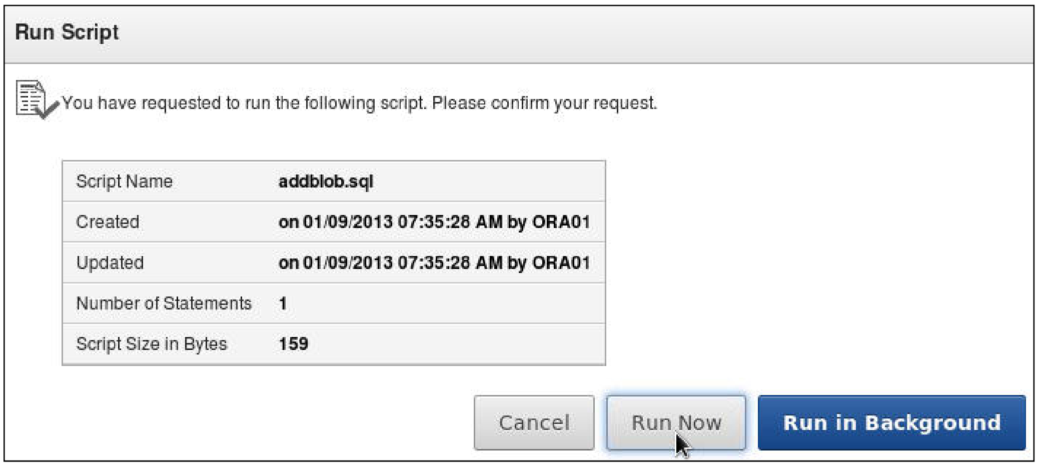


3) Click Browse and select the **`addblob.sql`** file (located in the `/home/oracle/labs/files` directory) and click **Open**. Then click **Upload**.

4) Your script was uploaded successfully. To run it, click the **Run** icon.

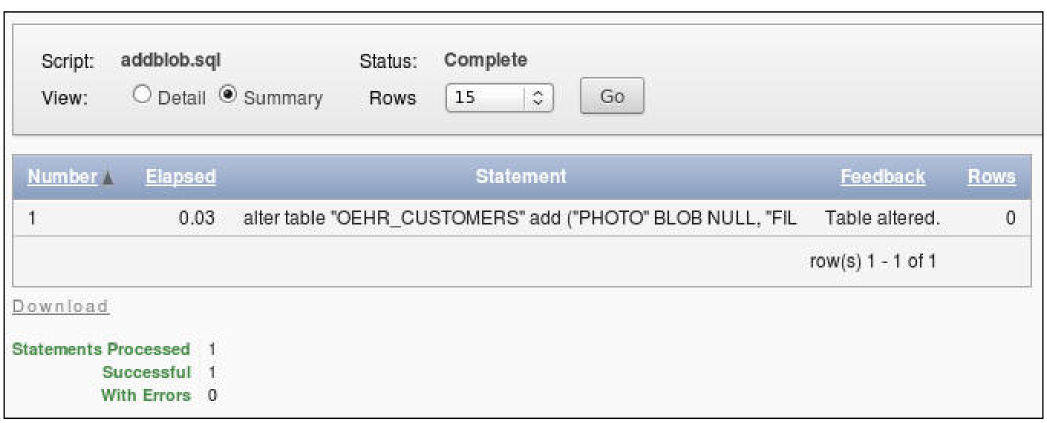


5) Click **Run Now**.



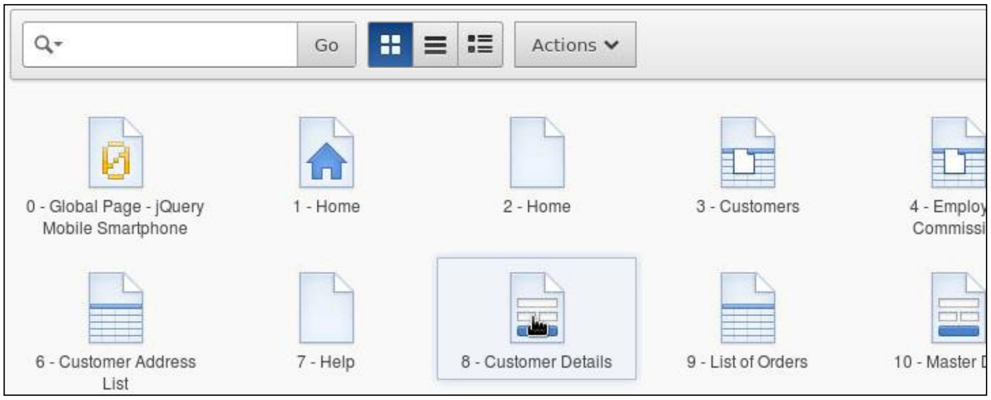
6) Click the **View Results** icon.

7) The OEHR_CUSTOMERS table was altered successfully.

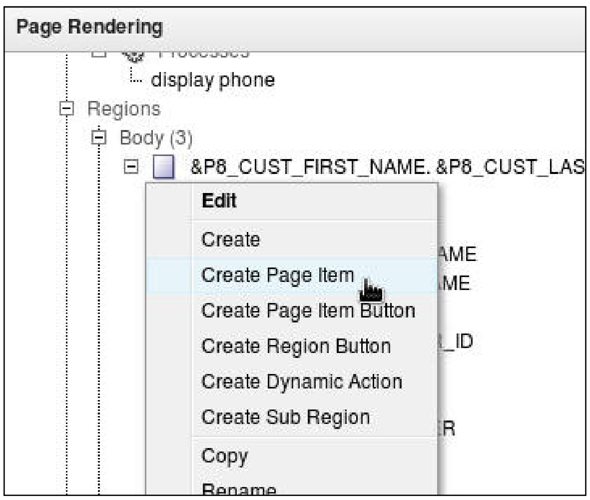


b. Add the Photo column to the Customer Details page.

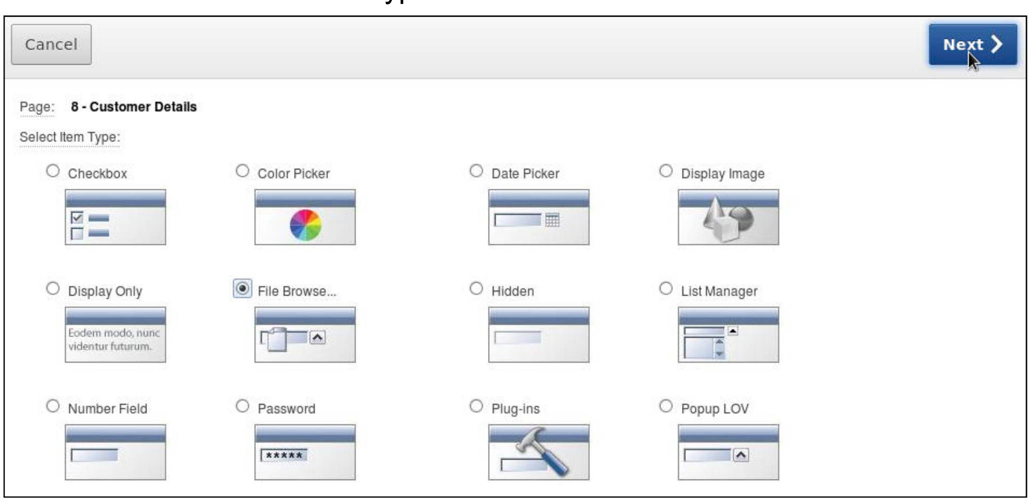
1) Navigate to the Order Management home page and click the **Customer Details** page.



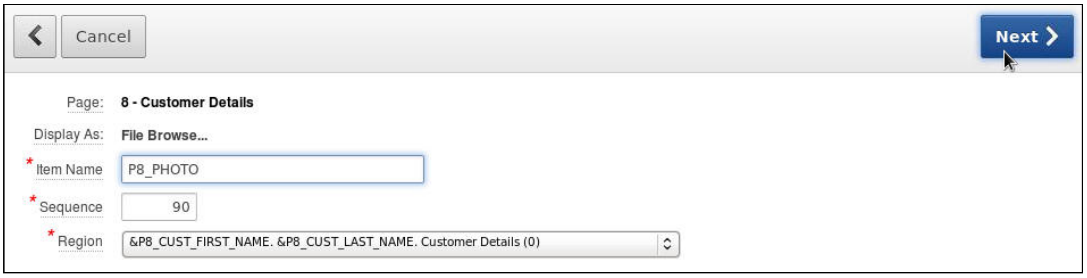
2) Right-click the first region in the list and select **Create Page Item**.



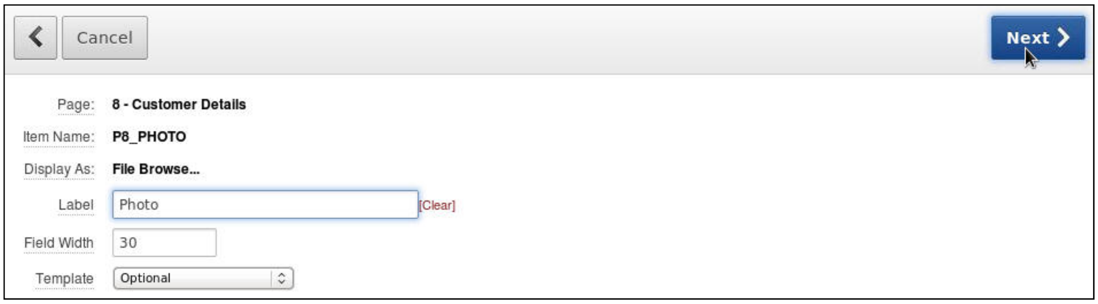
3) Select the **File Browse** item type and click **Next >**.



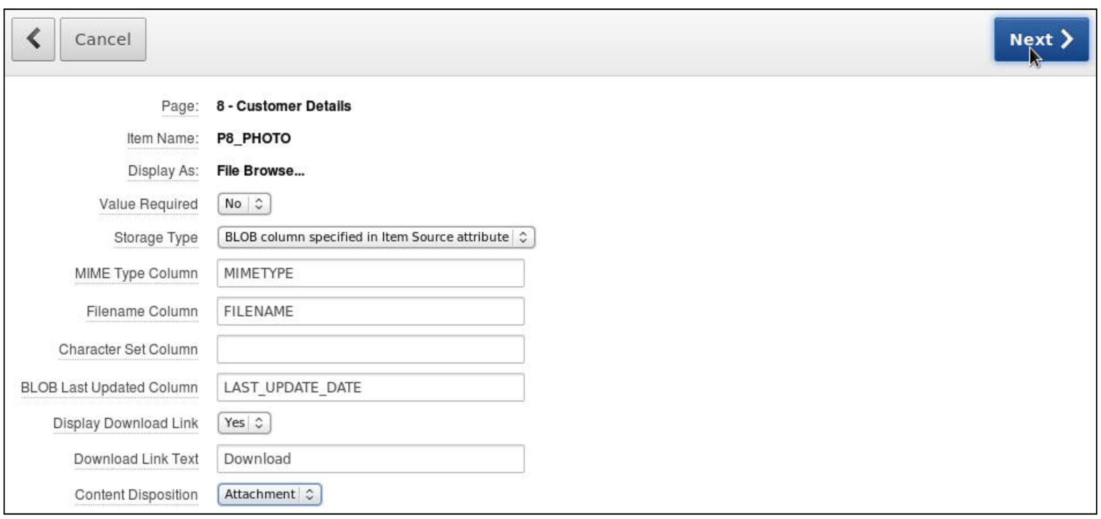
4) Change the Item Name to **P<*n*>_PHOTO** and click **Next >**.



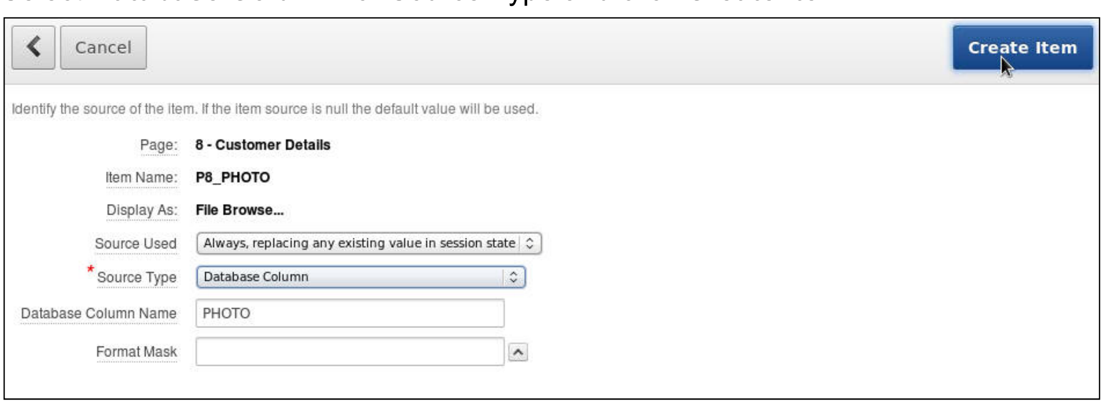
5) Accept the defaults and click **Next >**.



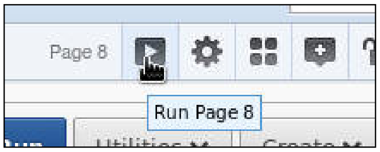
6) Enter **MIMETYPE** for MIME Type Column, **FILENAME** for Filename Column, and **LAST_UPDATE_DATE** for BLOB Last Updated Column. Select **Yes** for Display Download Link and **Attachment** for Content Disposition, and click **Next >**.



7) Select **Database Column** for Source Type and click **Create Item**.

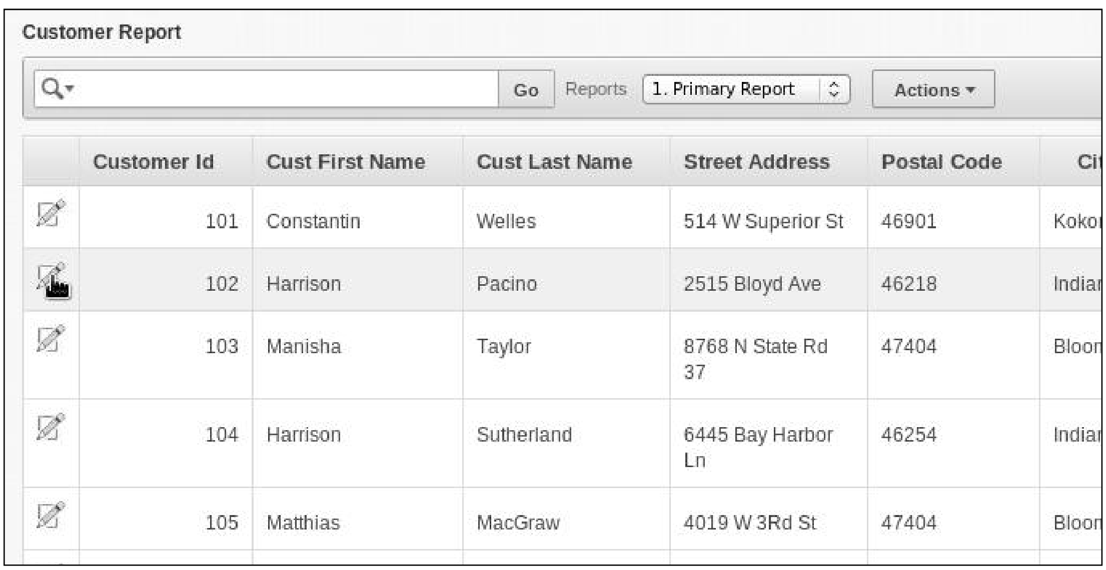


8) Click the **Run Page** icon.

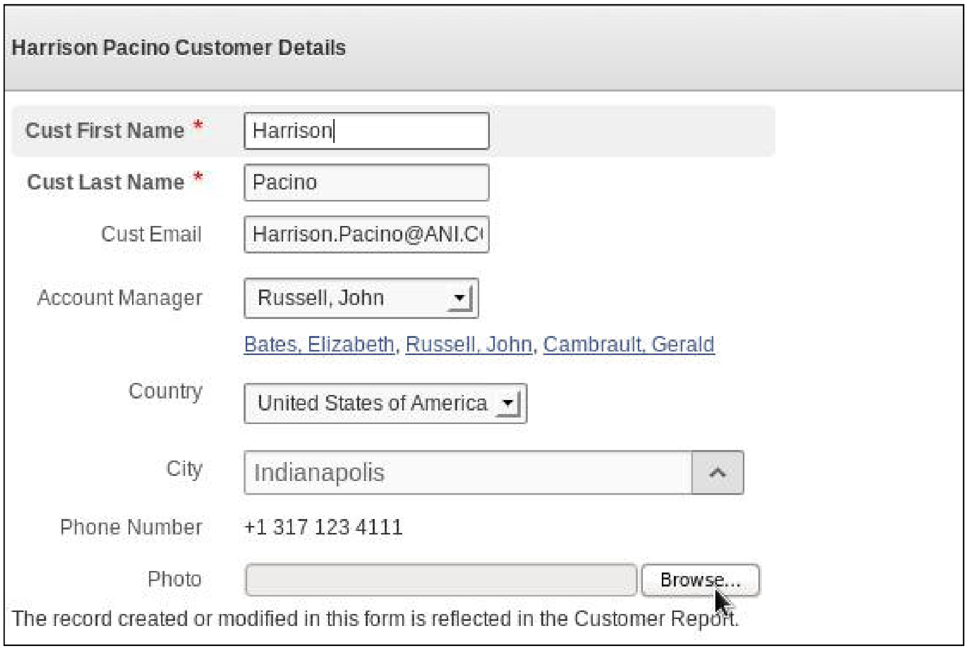


9) The page is displayed. Before you add a photo, you want to select a customer. Click **Cancel**.

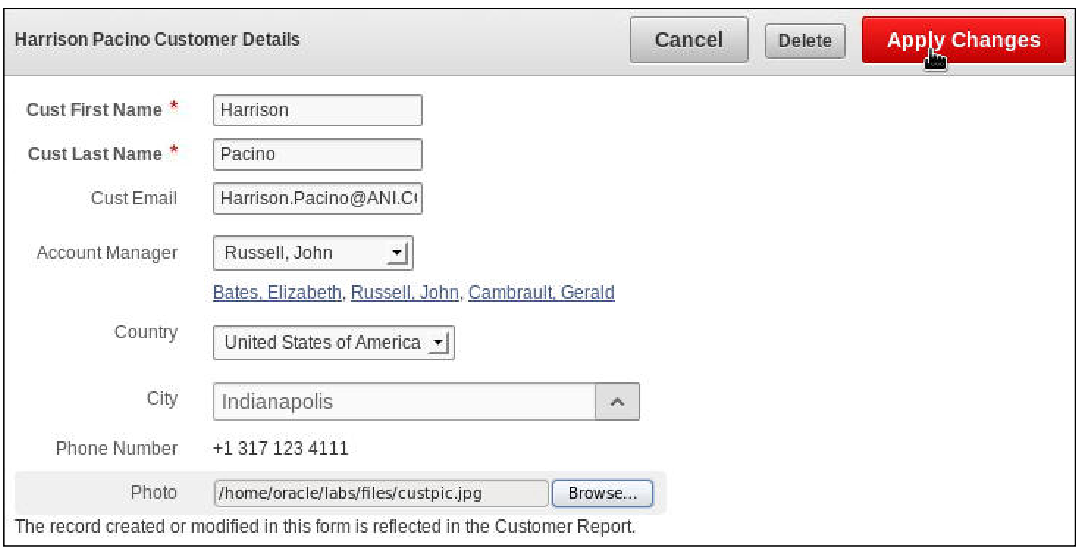
10) Click the **Edit** icon next to one of the rows.



11) Click the **Browse** button.

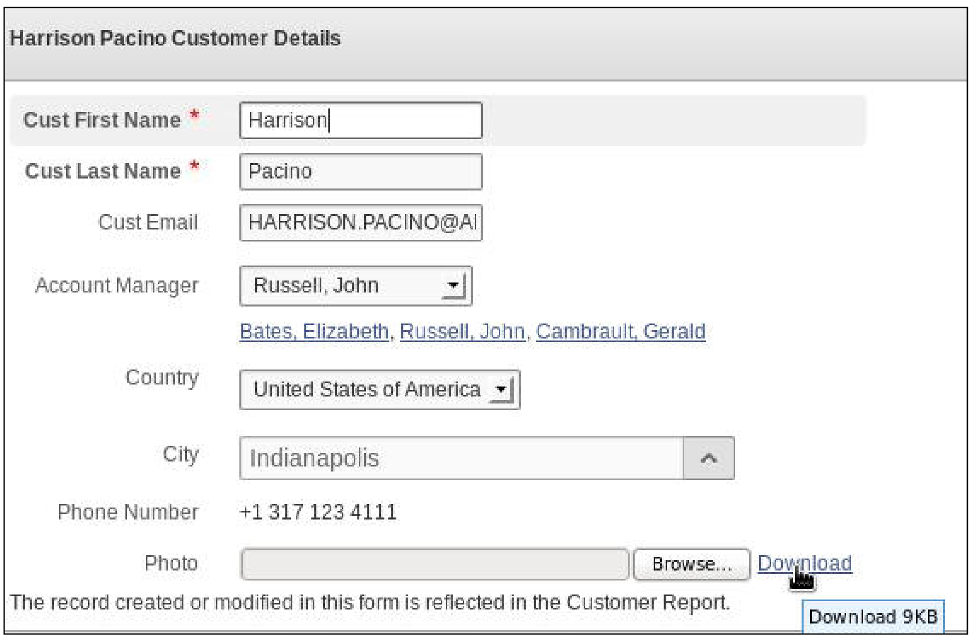


12) Select the **`custpic.jpg`** file (located in the `/home/oracle/labs/files` directory) and click **Open**. Then click **Apply Changes**.

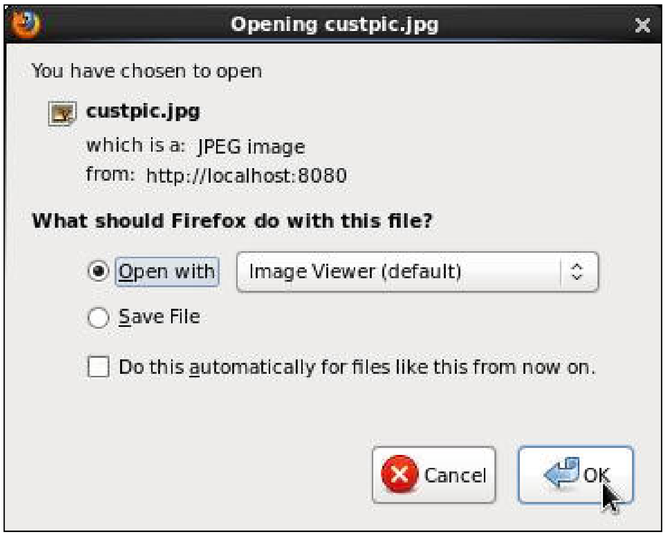


13) Click the same **Edit** icon that you just added the photo for.

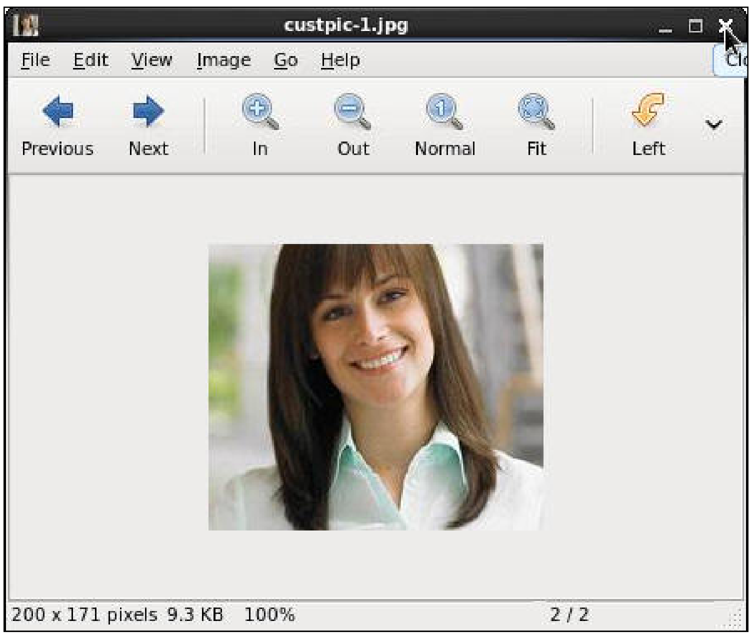
14) Note that there is a link to download the photo. Click the **Download** link.



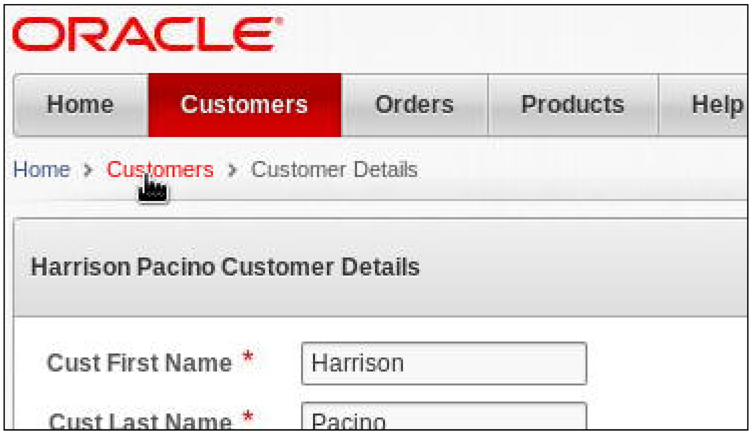
15) You are prompted to open the attachment or save it. Accept the default to open it in the Image Viewer. Click **OK**.



16) The image is displayed. Close the Image Viewer window.

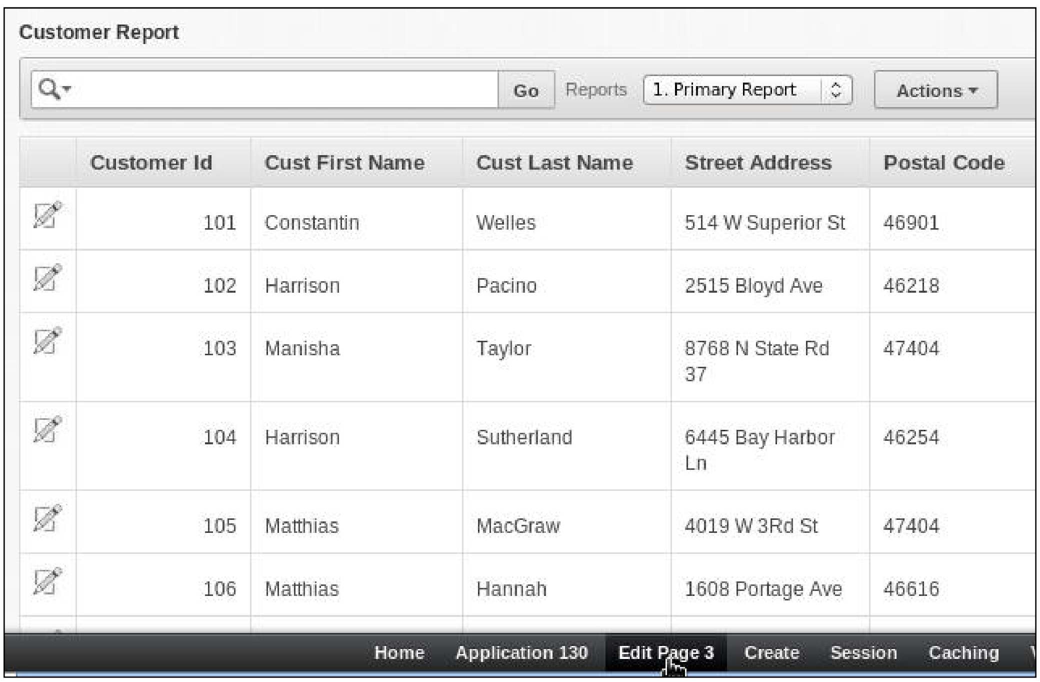


17) Click the **Customers** breadcrumb to return to the customers report.

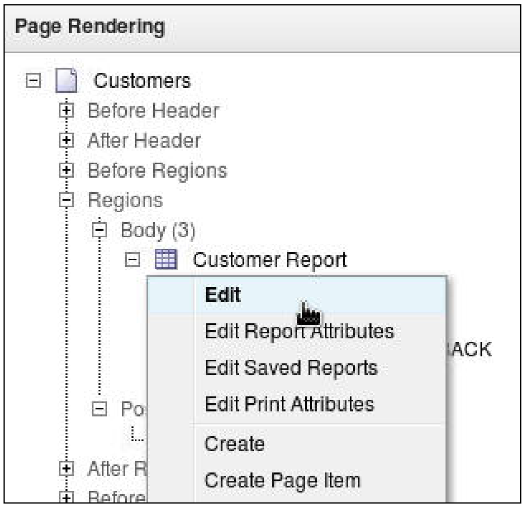


c. Change the Customers Report to include the Photo.

1) Click **Edit Page <*n*>** on the Developer toolbar.

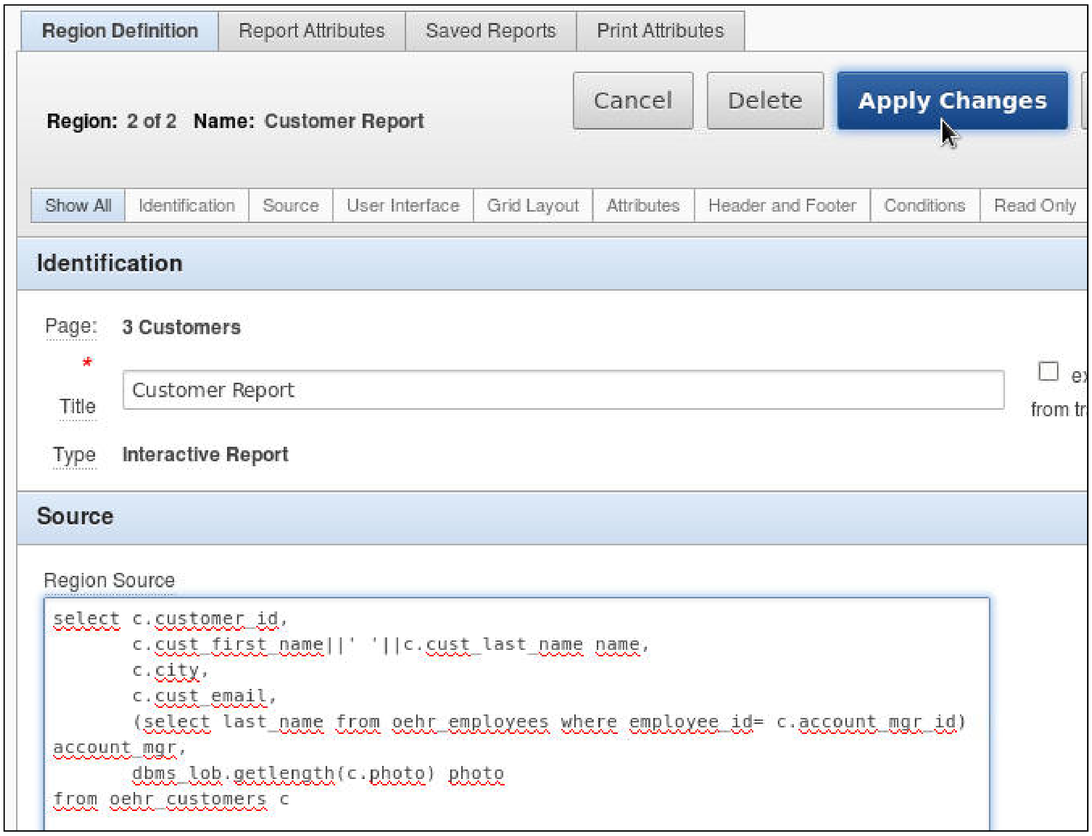


2) Right-click the **Customer Report** region and select **Edit**.

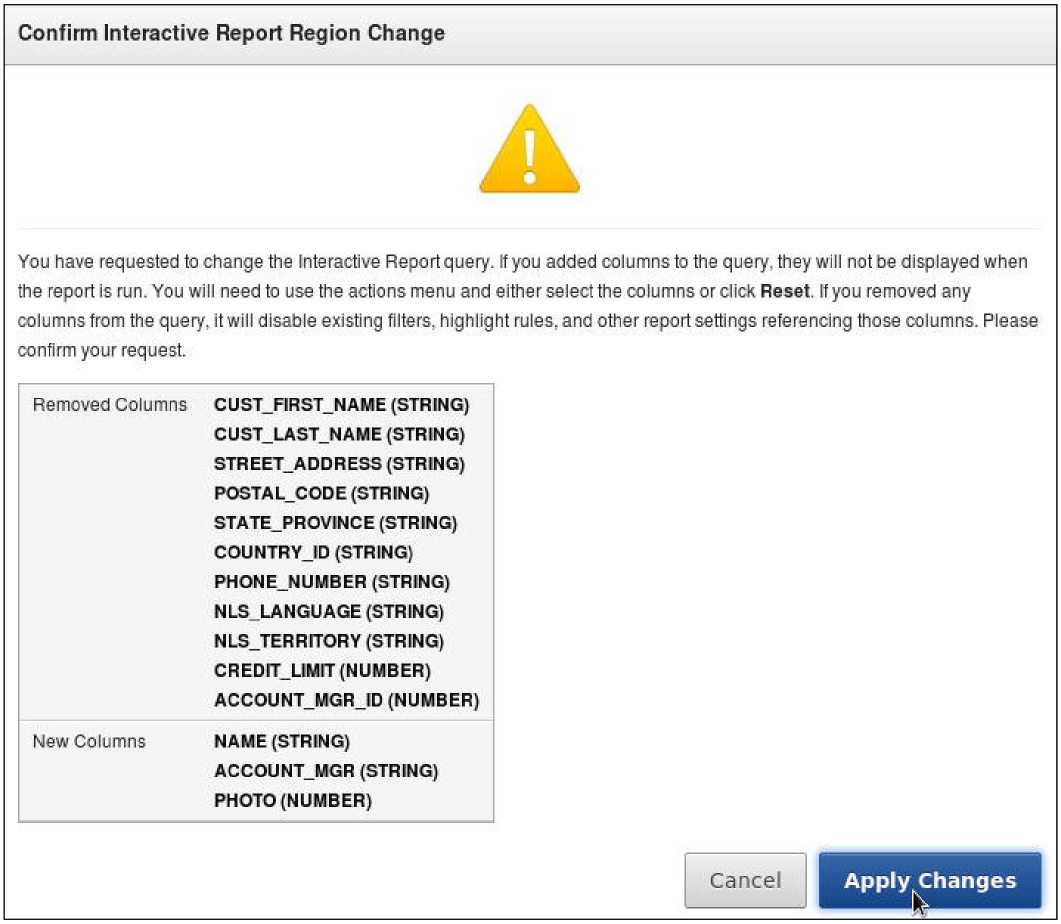


3) Replace the Region Source with the following code (located in the `/home/oracle/labs/files/lab_16_3_1.txt` file) and click **Apply Changes**.

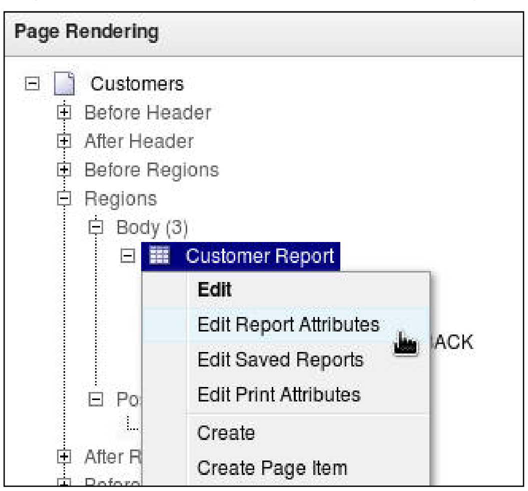
```
select c.customer_id,
       c.cust_first_name||' '||c.cust_last_name name,
       c.city,
       c.cust_email,
       (select last_name from oehr_employees where employee_id=
c.account_mgr_id) account_mgr,
       dbms_lob.getlength(c.photo) photo
from oehr_customers c
```


Region Definition
Report Attributes
Saved Reports
Print Attributes
Cancel
Delete
Apply Changes
Region: 2 of 2 Name: Customer Report
Show All
Identification
Source
User Interface
Grid Layout
Attributes
Header and Footer
Conditions
Read Only
Identification
Page: 3 Customers
Title
Customer Report
Type Interactive Report
Source
Region Source
select c.customer_id,
c.cust_first_name||' '||c.cust_last_name name,
c.city,
c.cust_email,
(select last_name from oehr_employees where employee_id= c.account_mgr_id)
account_mgr,
dbms_lob.getlength(c.photo) photo
from oehr_customers c

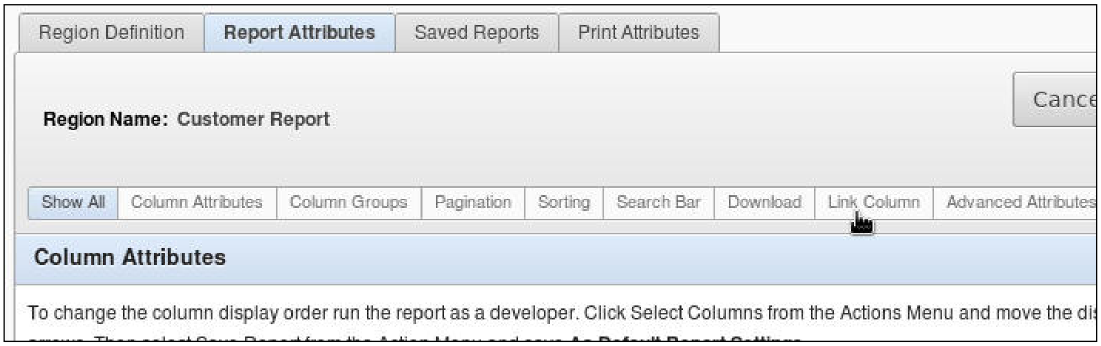
4) A confirmation window appears. Click **Apply Changes**.



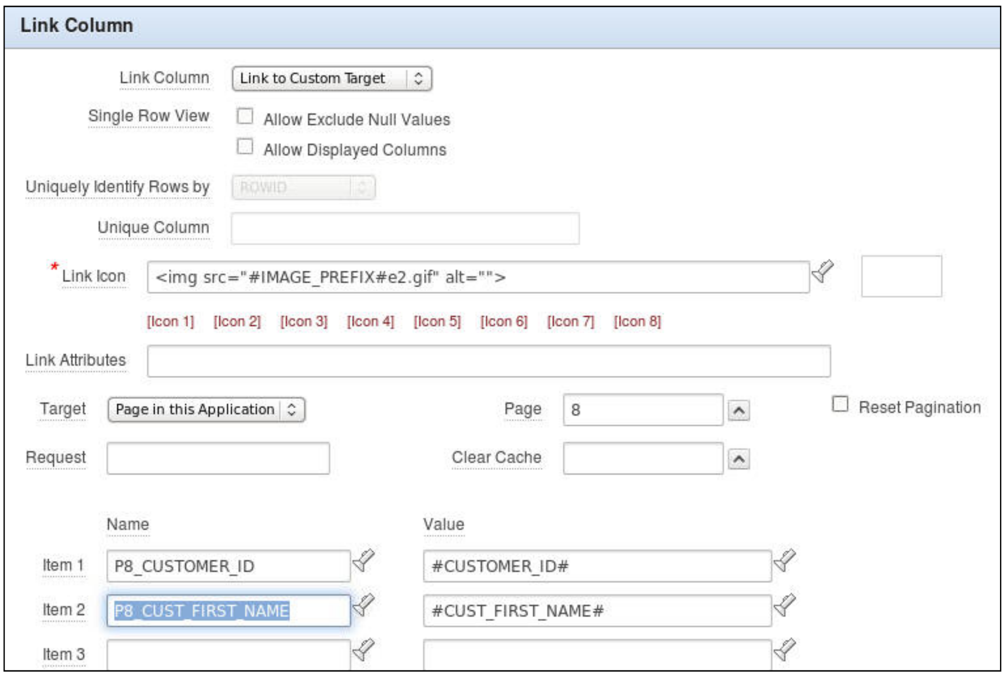
5) Because you no longer have the CUST_FIRST_NAME column in the query (it was changed to first and last name), you need to remove it from the link column area. Right-click the **Customer Report** region again and select **Edit Report Attributes**.



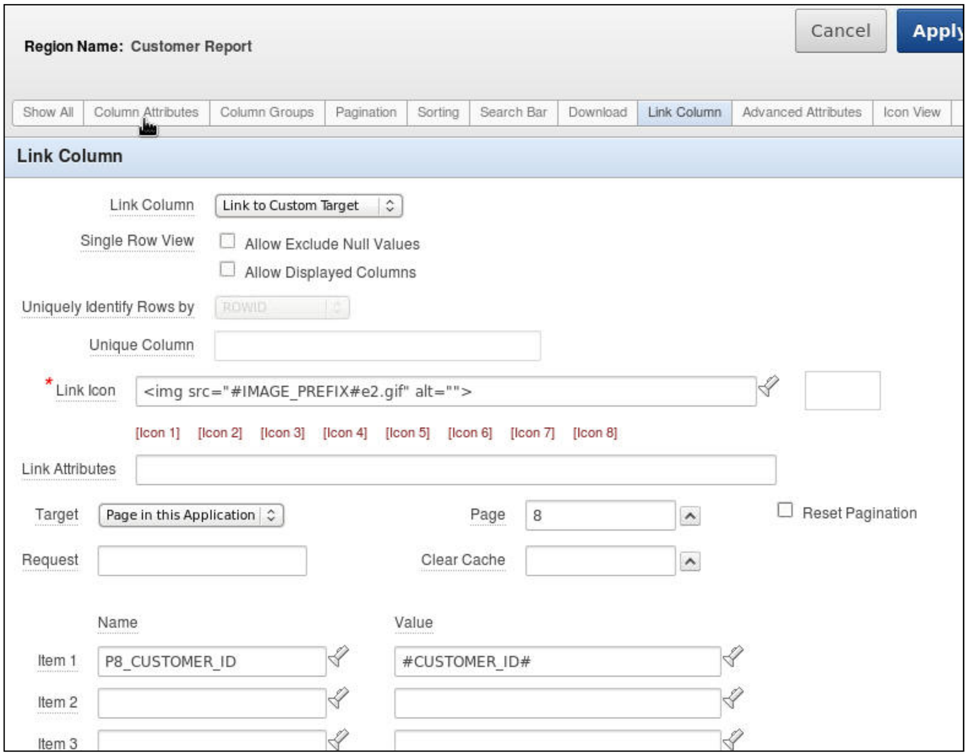
6) Click the **Link Column** subtab.



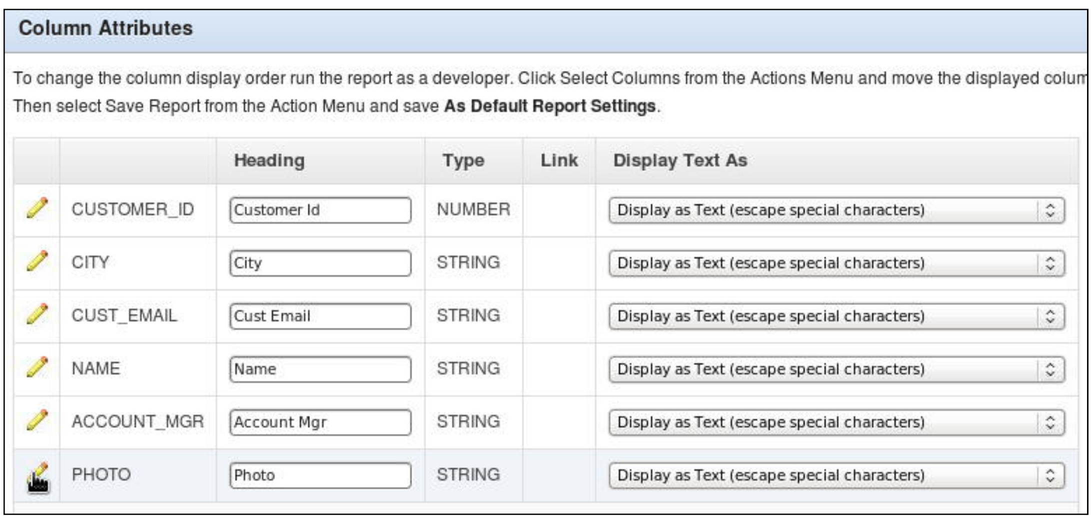
7) Notice the Item 2 items **P<n>_CUST_FIRST_NAME** and **#CUST_FIRST_NAME#**. These no longer appear in the SQL Query of your report, so you need to remove them from the link.



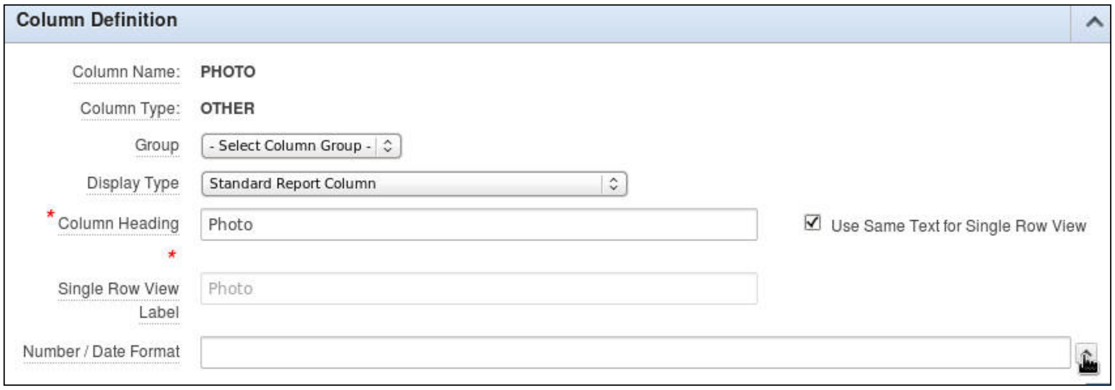
8) After the items are removed, click the **Column Attributes** subtab.



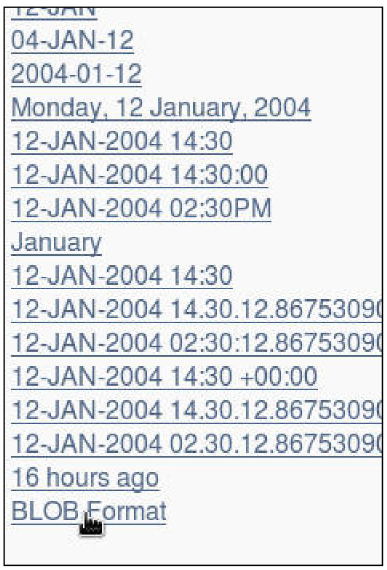
9) Click the **Edit** icon next to the **PHOTO** column.



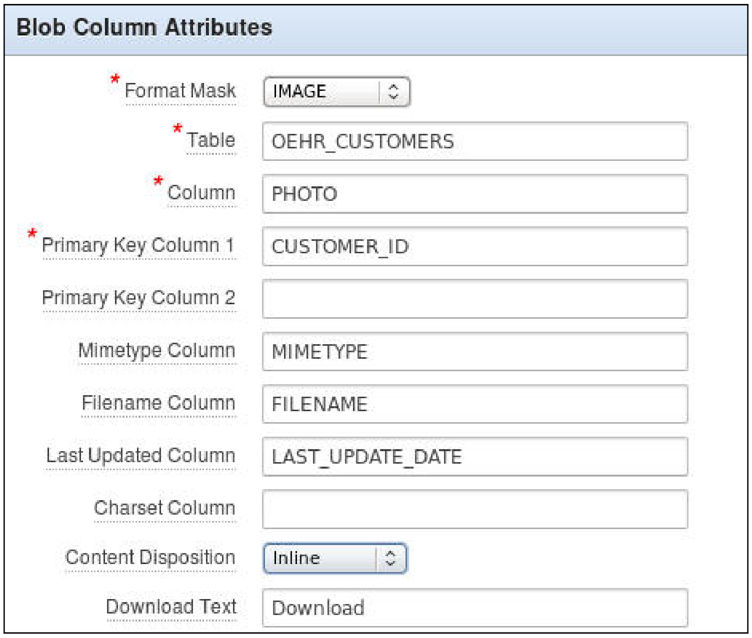
10) Select **Standard Report Column** for Display Type and click the up arrow for Number / Date Format.



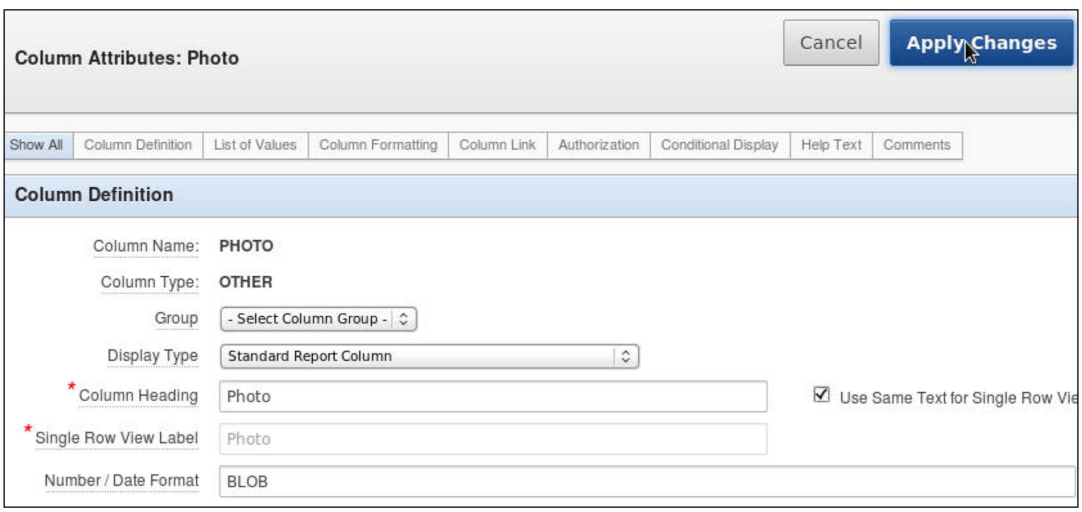
11) Scroll to the bottom of the list and select **BLOB Format**.



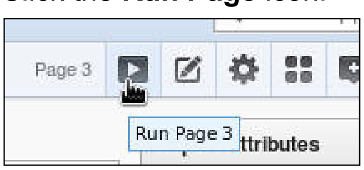
12) Scroll down to the Blob Column Attributes section. Select **IMAGE** for Format Mask, enter **OEHR_CUSTOMERS** for Table, **PHOTO** for Column Name, **CUSTOMER_ID** for Primary Key Column 1, **MIMETYPE** for Mimetype Column, **FILENAME** for Filename Column, **LAST_UPDATE_DATE** for Last Updated Column, select **Inline** for Content Disposition.



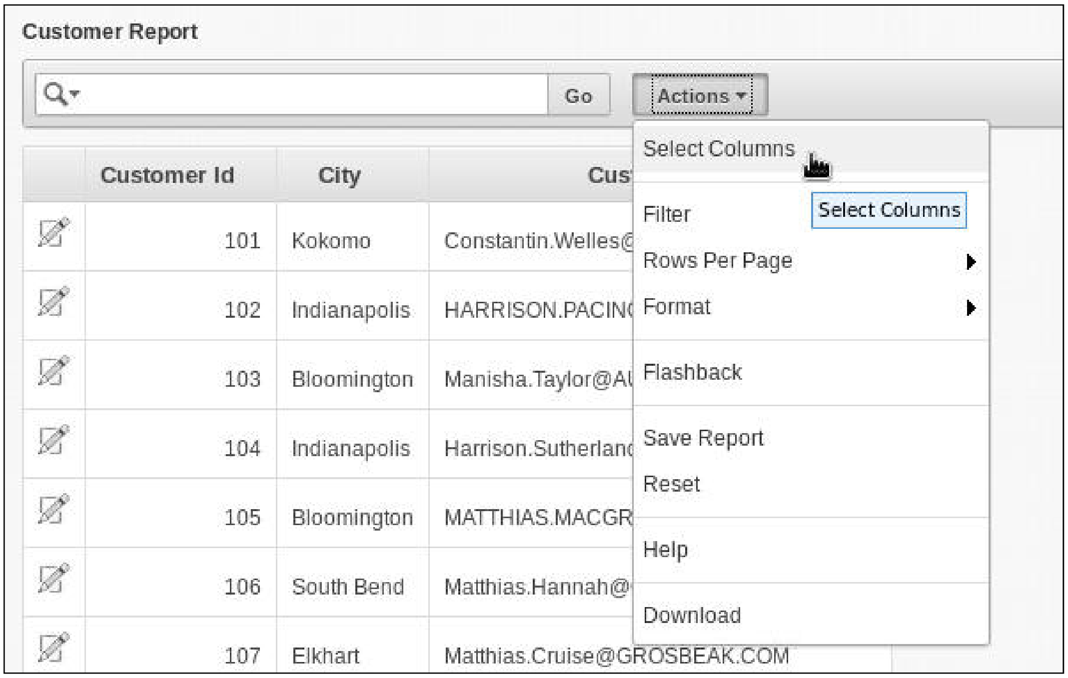
13) Scroll to the top of the page and click **Apply Changes**.



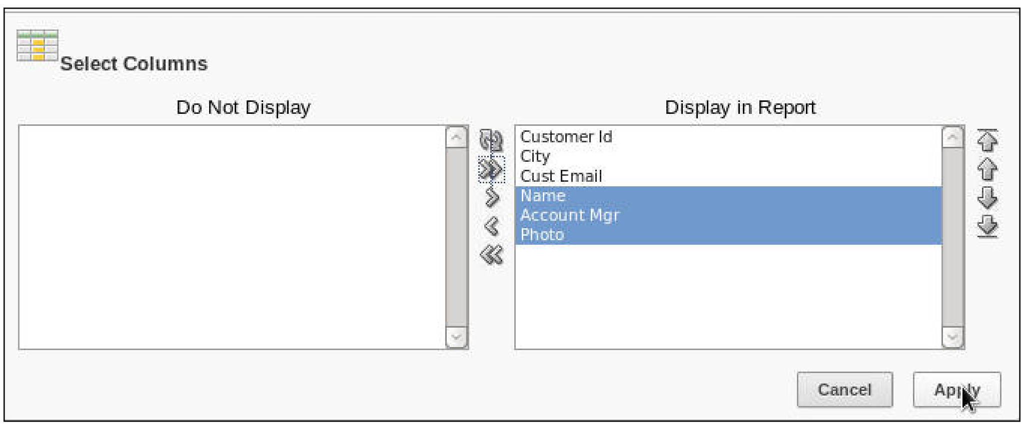
14) Click the **Run Page** icon.



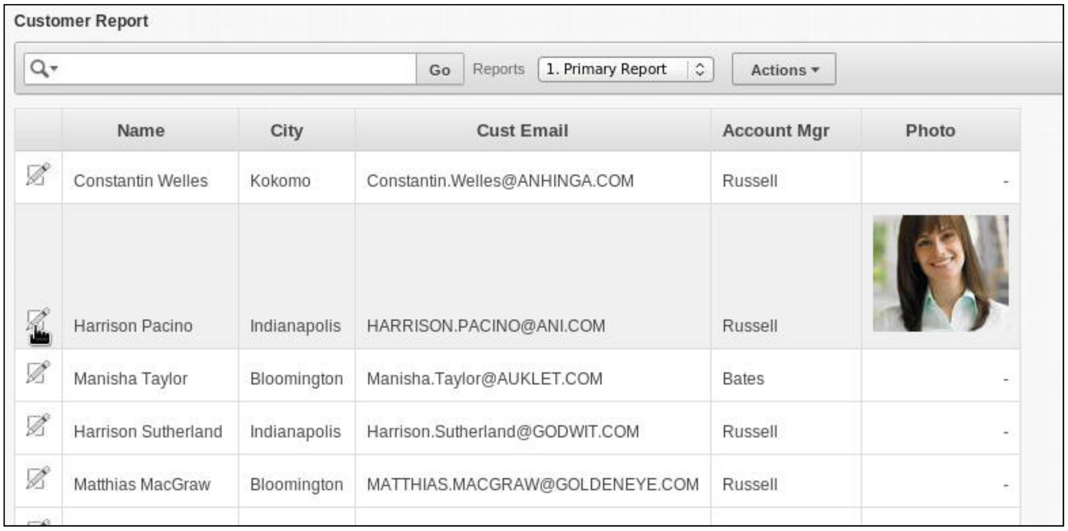
15) You need to add the columns you added to the SQL Query. Select **Actions > Select Columns**.



16) Select the **>>** to move all the columns to **Display in Report** and click **Apply**.

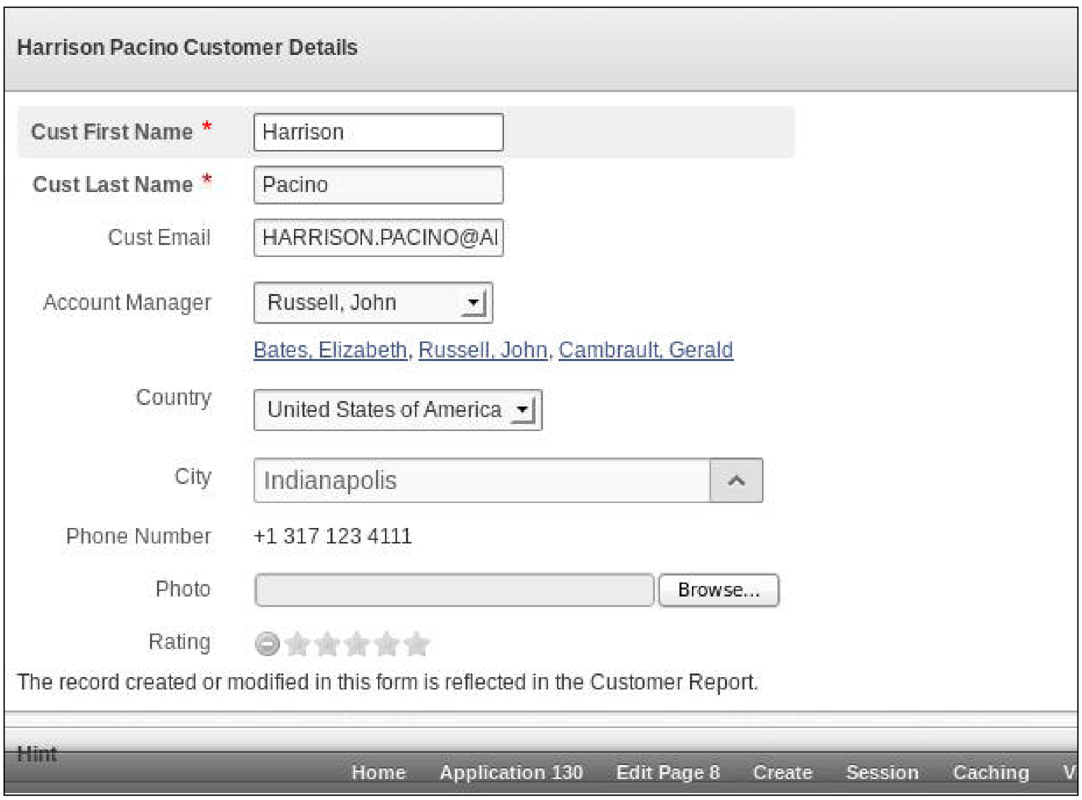


17) The photo is displayed in the report. Click the **Edit** icon next to the row.

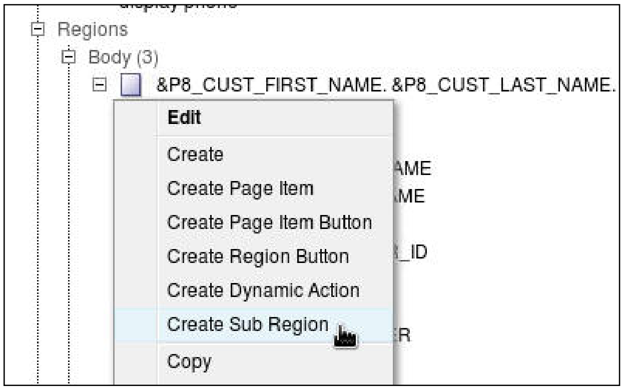


d. Add a Delete Image region by creating a new HTML region called Photo Image.

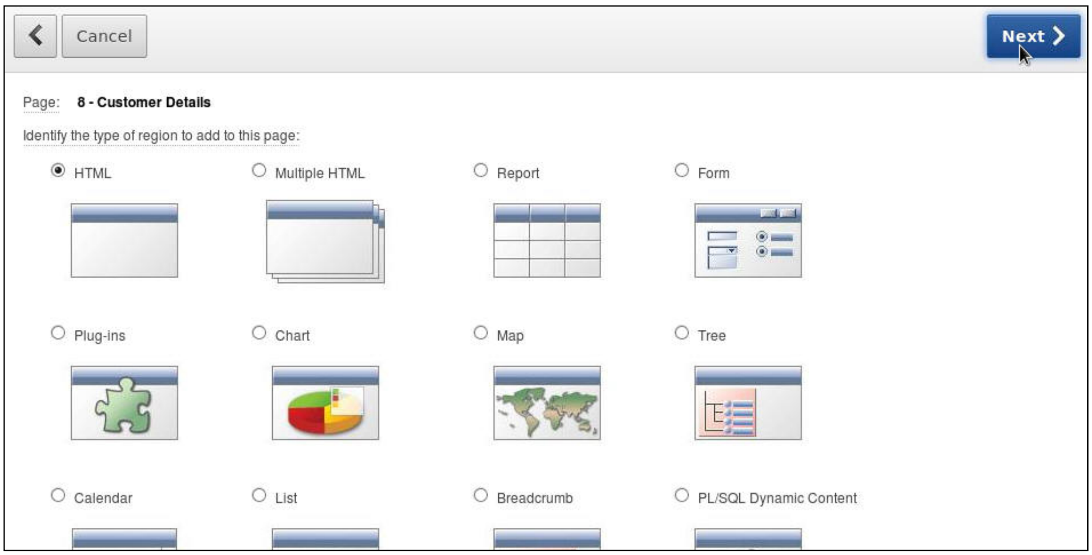
1) Click the **Edit Page** link on the Developer toolbar.



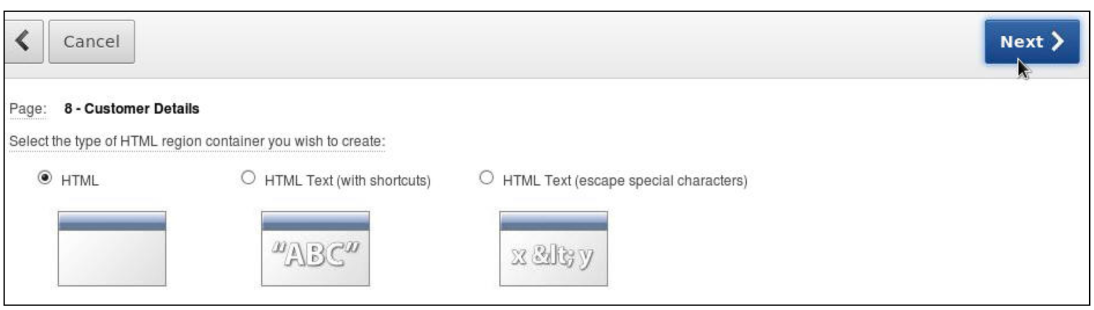
2) Right-click the **Customer Detail** region and select **Create Sub Region**.



3) Select **HTML** and click **Next >**.



4) Select **HTML** and click **Next >**.

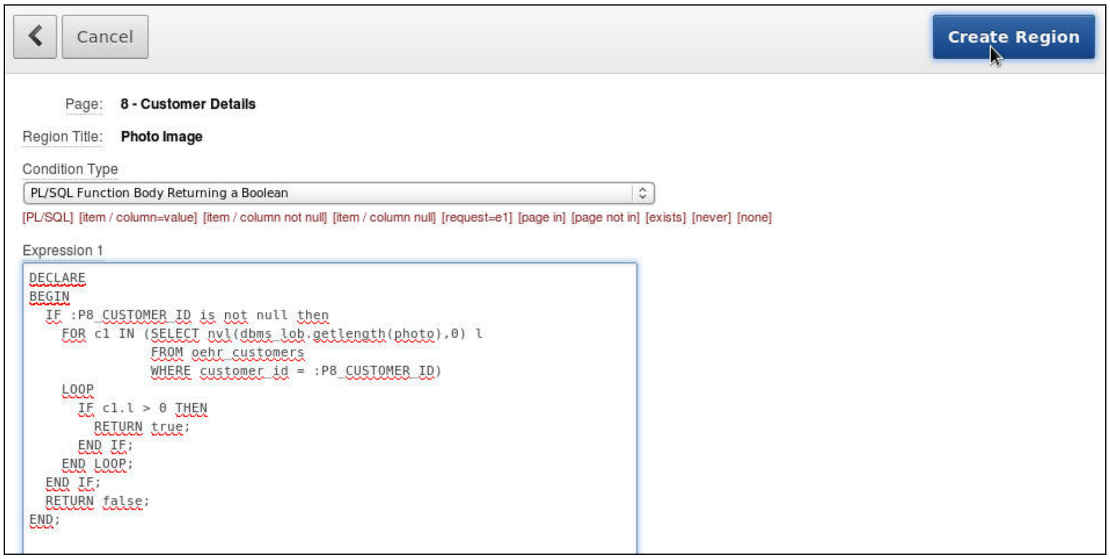


5) Enter **`Photo Image`** for **Title**. Select **Sidebar Region** for Region Template and click **Next >**.

6) Click **Next >**.

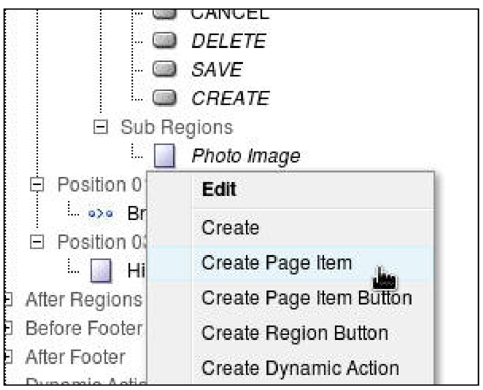
7) Select **PL/SQL Function Body Returning a Boolean** for Condition Type and enter the following code for Expression (located in the `/home/oracle/labs/files/lab16_3_2.txt` file). Replace **`P<n>`** with the page number. Click **Create Region**.

```
DECLARE
BEGIN
  IF :P<n>_CUSTOMER_ID is not null then
    FOR c1 IN (SELECT nvl(dbms_lob.getlength(photo),0) l
               FROM oehr_customers
               WHERE customer_id = :P<n>_CUSTOMER_ID)
    LOOP
      IF c1.l > 0 THEN
        RETURN true;
      END IF;
    END LOOP;
  END IF;
  RETURN false;
END;
```

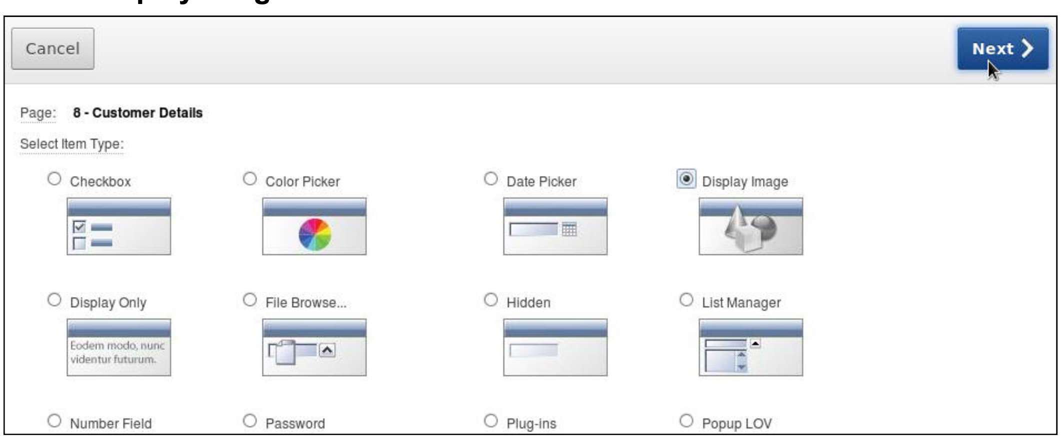


e. Create a Display Image item in the Photo Image region called `P<n>_IMAGE` that shows the image.

1) Under Regions, right-click the **Photo Image** region that you just created and select **Create Page Item**.



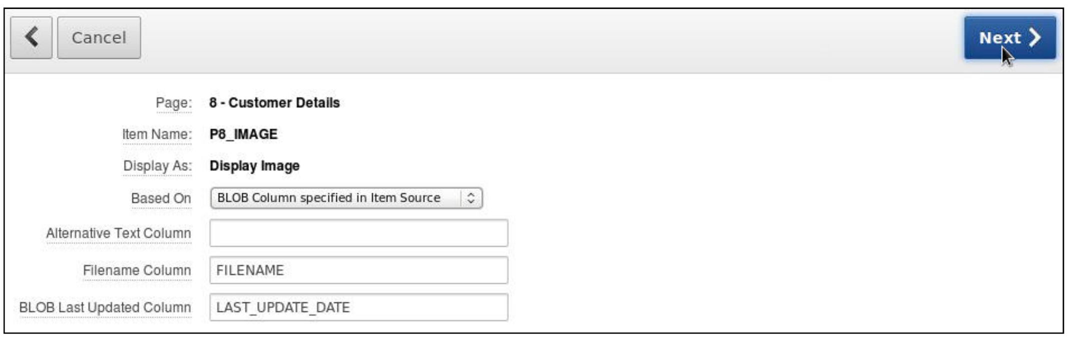
2) Select **Display Image** and click **Next >**.



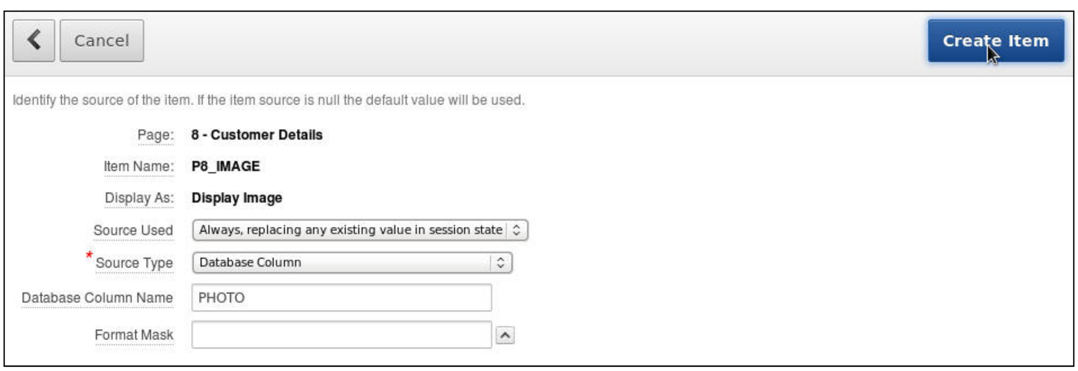
3) Enter `P<n>_IMAGE` for the item and click **Next >**.

4) Click **[Clear]** for the label and click **Next >**.

5) Make sure that **BLOB Column specified in Item Source** is selected for Based On, and enter **FILENAME** for Filename Column and **LAST_UPDATE_DATE** for BLOB Last Updated Column. Click **Next >**.

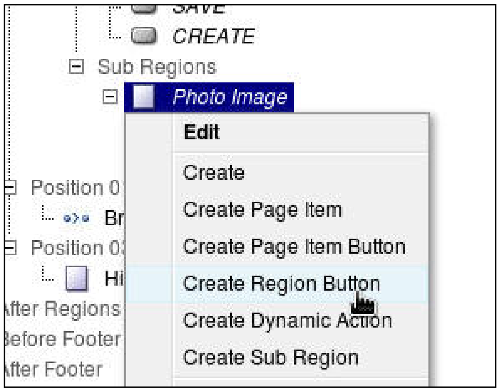


6) Select **Database Column** for Source Type and change the Database Column Name to **PHOTO**. Click **Create Item**.

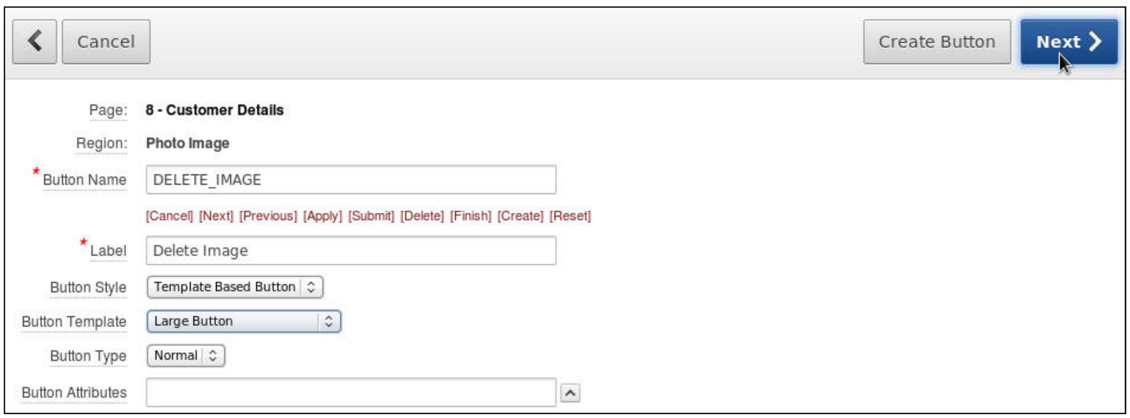


f. Create a delete button called DELETE_IMAGE. Create a PL/SQL process called “delete image” that is invoked to perform the delete. Run the page.

1) Right-click the **PHOTO IMAGE** region and select **Create Region Button**.

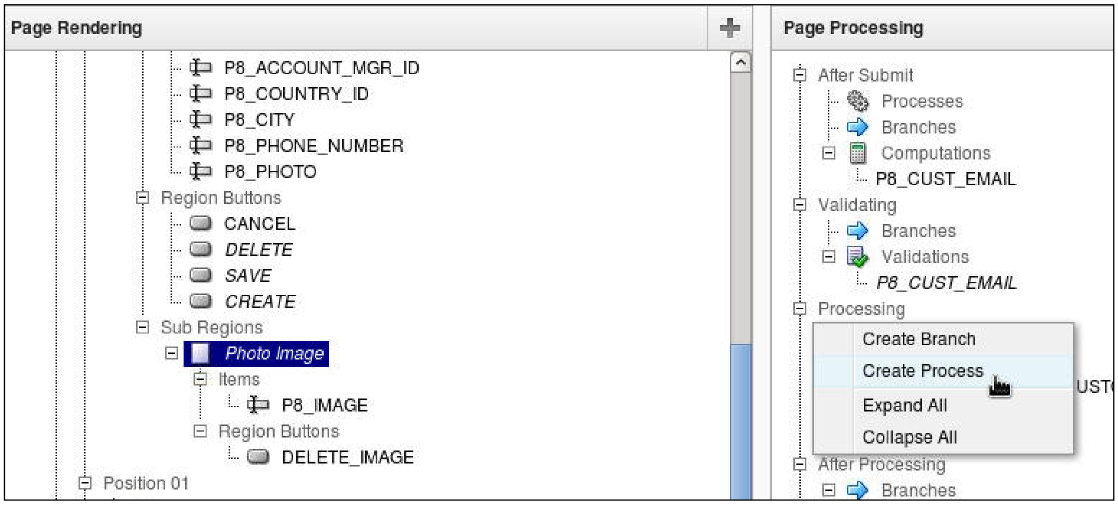


2) Enter `DELETE_IMAGE` for button name and `Delete Image` for Label. Select **Template Based Button** for Button Style and **Large Button** for Button Template, and click **Next >**.

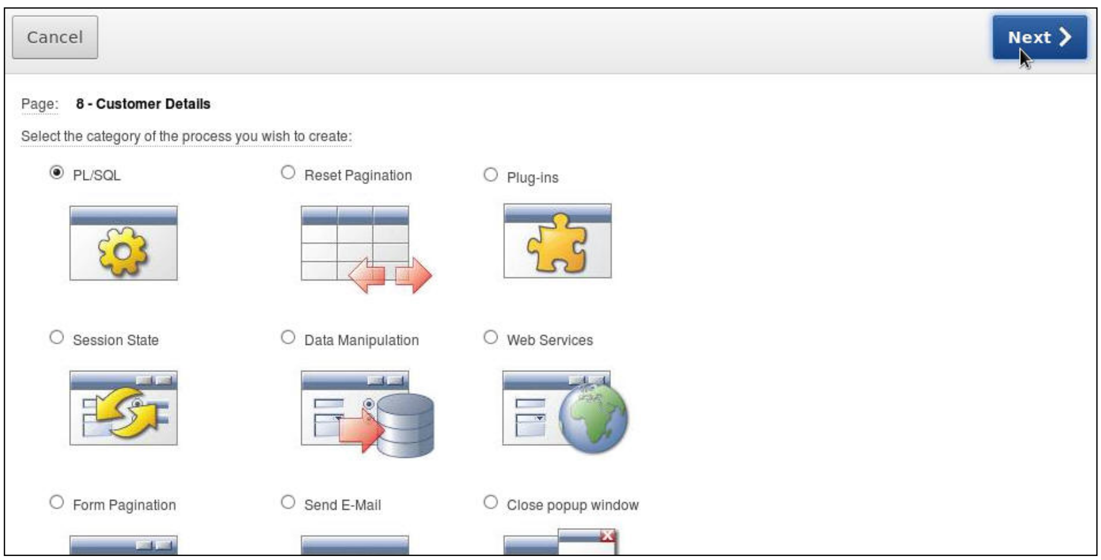


3) Select **Below Region** for Position and click **Create Button**.

4) Under Page Processing, right-click **Processing** and select **Create Process**.



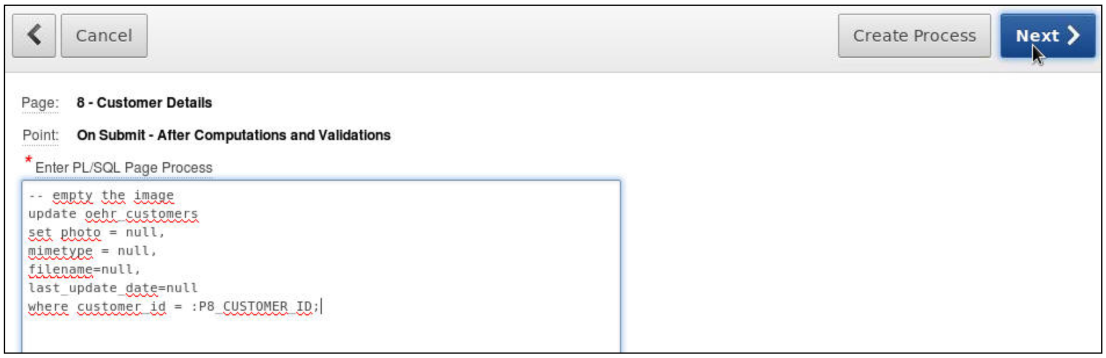
5) Select **PL/SQL** and click **Next >**.



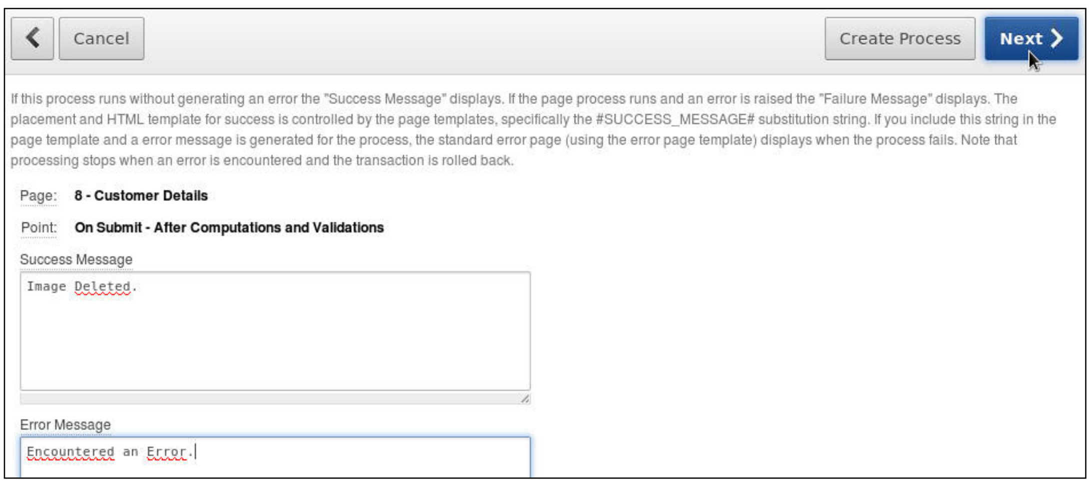
6) You want this process to be placed before the DML process. Enter `delete image` for Name and change the sequence number to `5` so that it will be executed first and click **Next >**.

7) Enter the following code in the PL/SQL Page Process area (located in the `home/oracle/labs/files/lab16_3_3.txt` file). Make sure to replace **`P<n>`** with the page number. Click **Next >**.

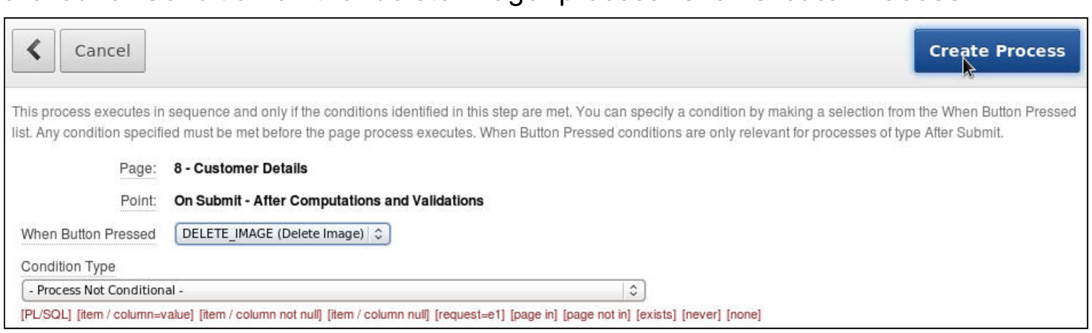
```
-- empty the image
update oehr_customers
set photo = null,
mimetype = null,
filename=null,
last_update_date=null
where customer_id = :P<n>_CUSTOMER_ID;
```



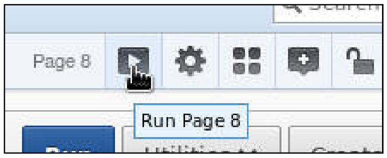
8) Enter success and failure messages. Click **Next >**.



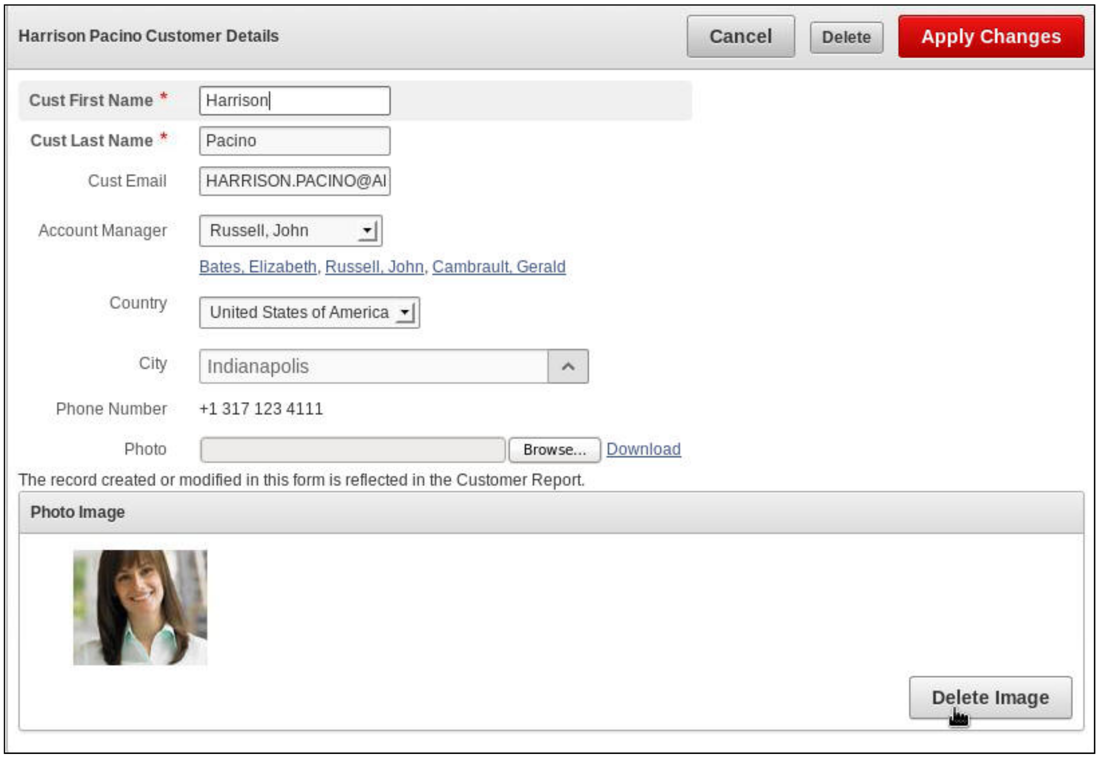
9) Select **DELETE_IMAGE (Delete Image)** for When Button Pressed. The button is clicked for Condition on the “delete image” process. Click **Create Process**.



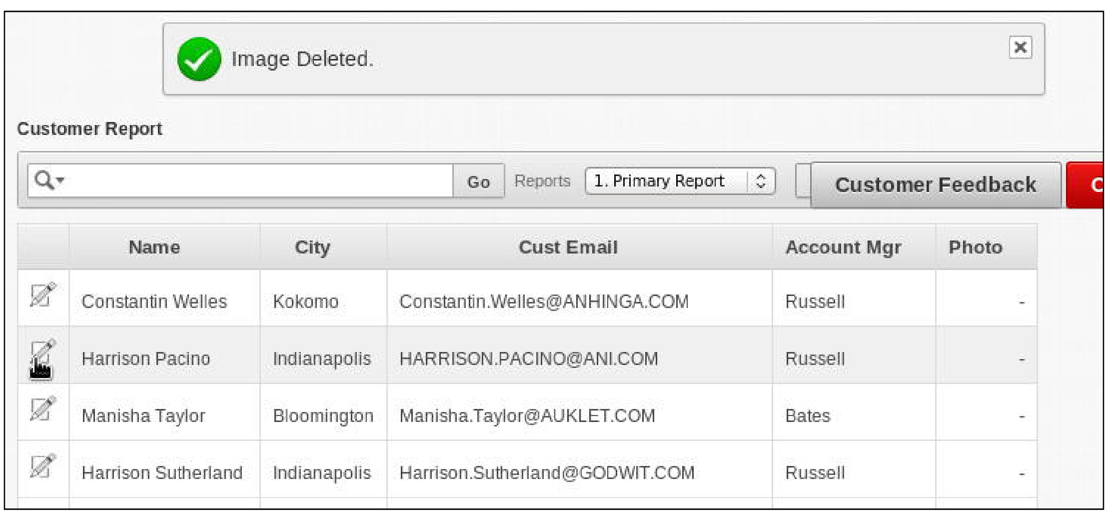
10) Click the **Run Page** icon.



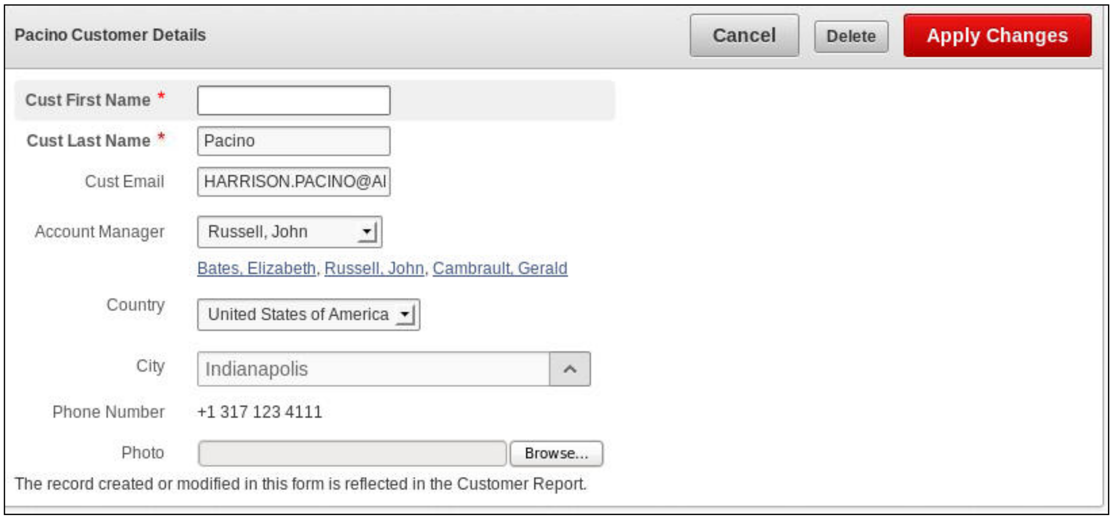
11) Click the **Delete Image** button.



12) The image was deleted. Notice that the image is no longer displayed in the report. Click the **Edit** icon for the same row.



13) The image is no longer available to download and the Delete Image region is not displayed.



Practices for Lesson 17: Creating and Editing Charts

Chapter 17

Practices for Lesson 17: Overview

Practices Overview

In these practices, you will learn how to create and modify charts as well as HTML 5 charts for mobile applications.

Practice 17-1: Creating and Editing Charts

Overview

In this practice you create a new chart with a Horizontal Bar – 3D Bar Chart. You then modify the chart and change it to a 3D Column Chart and change some of the settings to see how they work.

Assumptions

You have performed the previous practices.

Tasks

a. Create a new Flash chart page that includes a Horizontal Bar – 3D Bar Chart. Use "Employees by Department" for the Title, "Number of Employees" for the X-Axis, and "Departments" for the Y-Axis. Use the following query (located in `/home/oracle/labs/files/lab17_1_1.txt`). Test your page.

```
select null link,
department_name chart_label,
count(*) chart_value
from oehr_employees e,
oehr_departments d
where e.department_id = d.department_id
group by department_name
```

b. Modify your chart and change it to a 3D Column Chart with look 7. Add X-Axis scrollbars, enter `45` for the X-Axis label rotation, reverse the X and Y axes titles, and test your page.

Practice 17-2: Creating an HTML5 Chart for Mobile Applications

Overview

In this practice, you create an HTML5 chart for mobile application.

Assumptions

You have performed the previous practices.

Tasks

a. Create an HTML5 Pie & Doughnut chart. Use “Departments Chart” for the Title. Use the following query (located in `/home/oracle/labs/files/lab17_2_1.txt`). Test your mobile page.

```
select null link,
department_name chart_label,
count(*) chart_value
from oehr_employees e,
oehr_departments d
where e.department_id = d.department_id
group by department_name
```

Practice 17-3: Creating Additional Charts

Overview

In this practice, you create some additional charts, such as a combined chart, a project Gantt chart, and a Gauge chart.

Assumptions

You have performed the previous practices.

Tasks

a. You build a combined chart. First, you upload and run the **add_oehr_employees3.sql**. script to create a table called oehr_employees3.

b. Create a 3D Column Flash Chart named **Salary Commission Combined Chart**. Use the following SQL Query (located in the /home/oracle/labs/files/lab17_3_1.txt file). Specify **Employee** for X-Axis Title and **Salary** for Y-Axis title. Edit the chart by changing the Series attribute to Line.

```
SELECT NULL LINK,
       LAST_NAME LABEL,
       SALARY "Salary",
       COMMISSION_PCT "COMMISSION"
FROM   OEHR_EMPLOYEES3
WHERE MANAGER_ID = 100
Order by LAST_NAME
```

c. Upload and run the **tasks_ddl.sql**. script to create a table called tasks1.

d. Create a Project Gantt Chart that uses the following SQL Query (located in the /home/oracle/labs/files/lab17_3_2.txt file).

```
SELECT NULL LINK,
       tasks task_name, id, parent_id,
       START_DATE, END_DATE,
       STATUS,
       START_DATE-3, END_DATE+2
from TASKS1
```

e. Create a Salary Gauge chart for employees in department 80. Use the following SQL Query (located in the /home/oracle/labs/files/lab17_3_3.txt file).

```
select sum(case when salary <10000 then 0 else 1 end) value,
       count(*) max_value
from oehr_employees
where department_id=80
```

Solution 17-1: Creating and Editing Charts

Overview

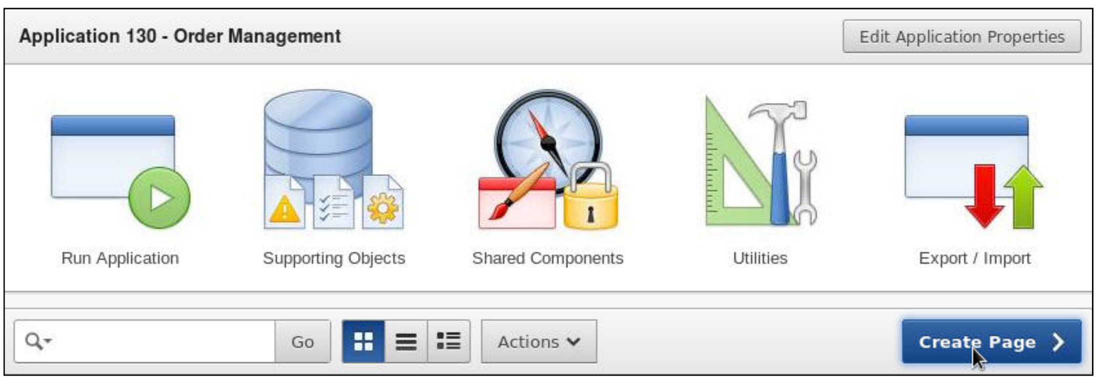
In this solution, you create a new chart with a Horizontal Bar – 3D Bar Chart. You then modify the chart and change it to a 3D Column Chart and change some of the settings to see how they work.

Tasks

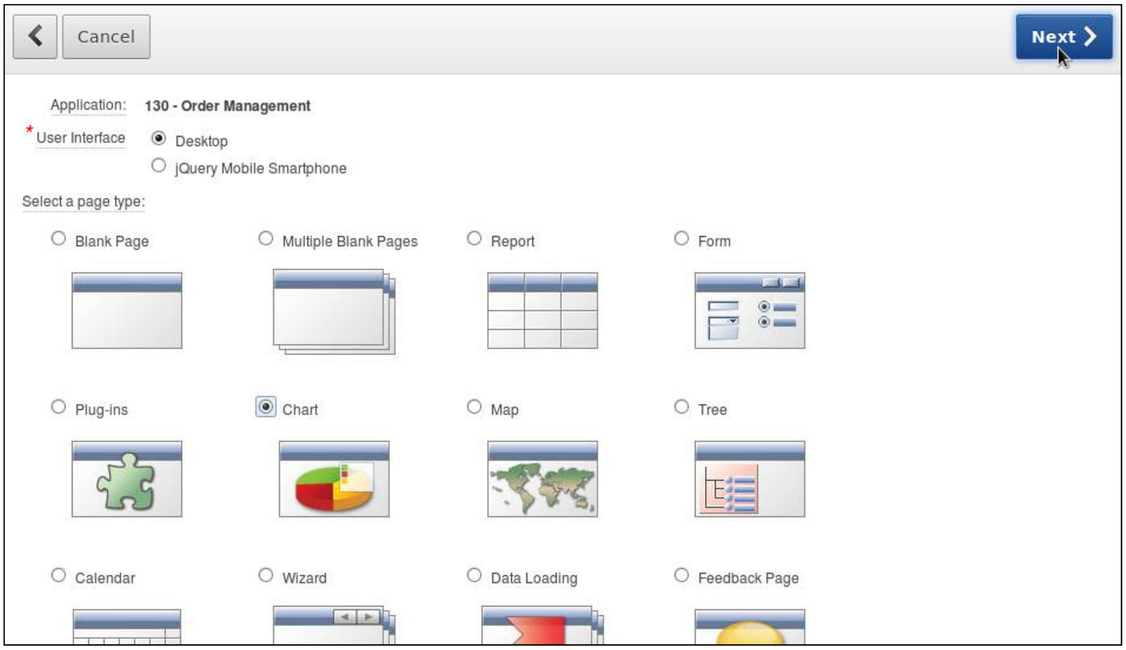
a. Create a new Flash chart page that includes a Horizontal Bar – 3D Bar Chart. Use "Employees by Department" for the Title, "Number of Employees" for the X-Axis, and "Departments" for the Y-Axis. Use the following query (located in `/home/oracle/labs/files/lab17-1.txt`). Test your page.

```
select null link,
department_name chart_label,
count(*) chart_value
from oehr_employees e,
oehr_departments d
where e.department_id = d.department_id
group by department_name
```

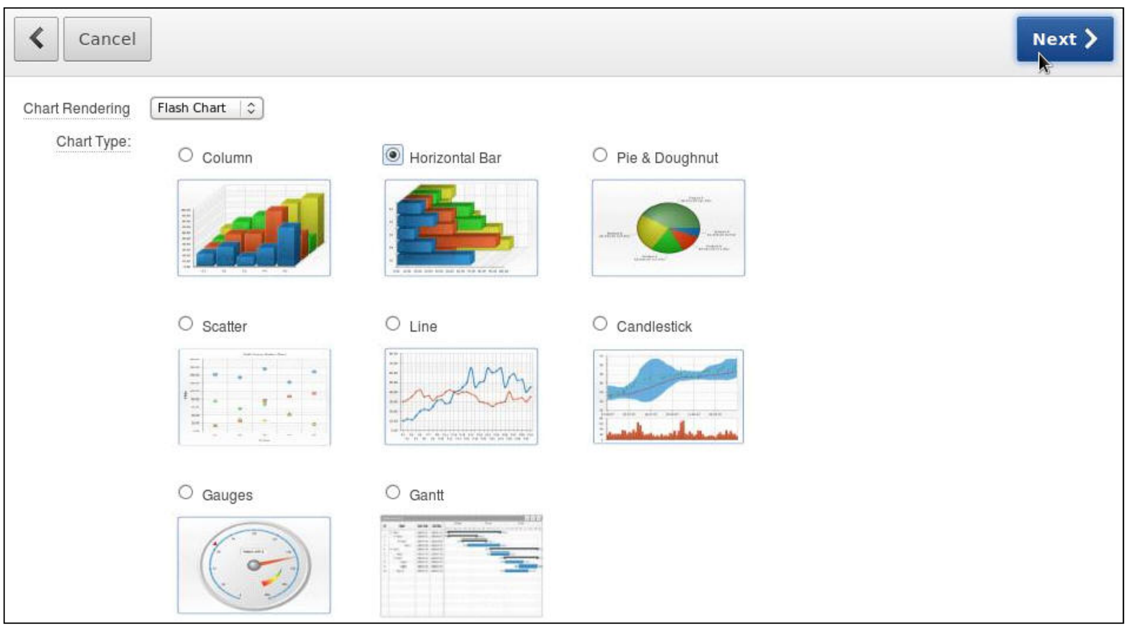
1) On the Order Management application home page, click **Create Page >**.



2) Select **Chart** and click **Next >**.



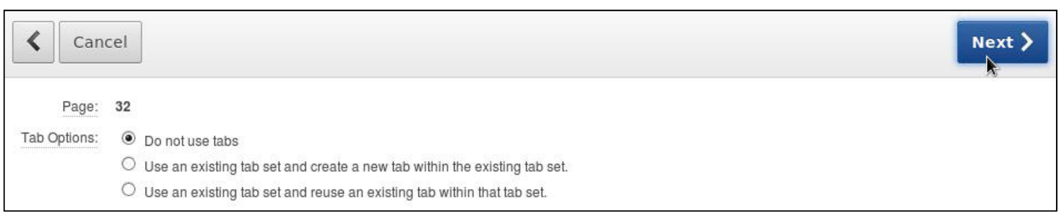
3) Select **Flash Chart**. Select **Horizontal Bar** and click **Next >**.



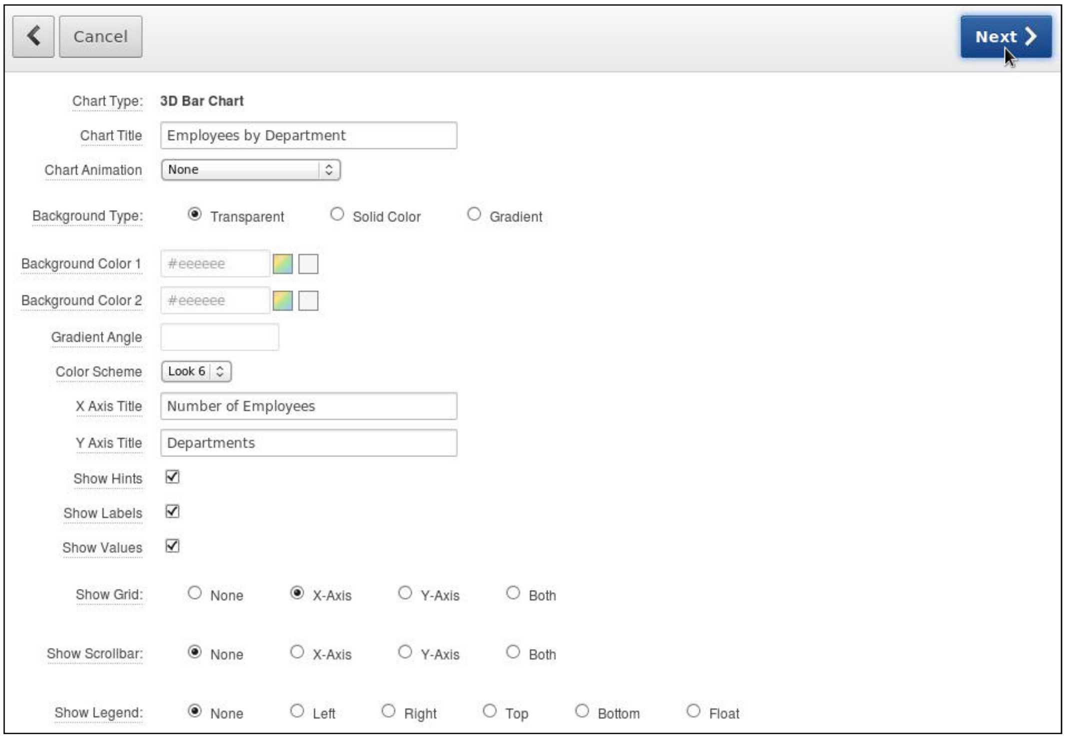
4) Select **3D Bar Chart** and click **Next >**.

5) Enter **Employees by Department** for Page Name and Region Name and click **Next >**.

6) Accept the defaults and click **Next >**.

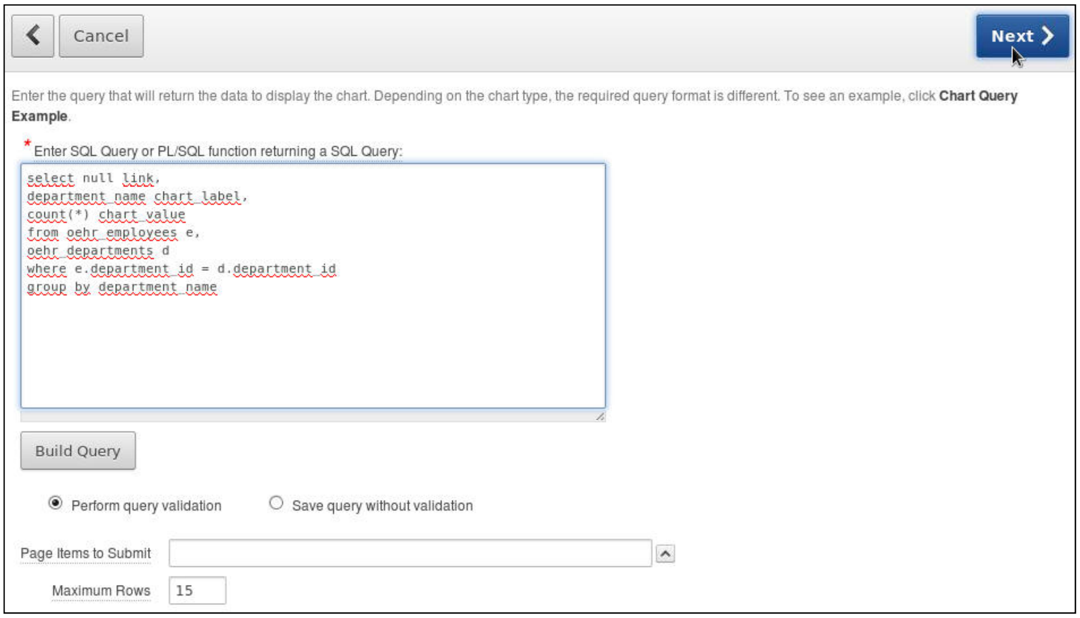


7) Enter **Employees by Department** for chart Title, **Number of Employees** for X Axis Title, and **Departments** for Y Axis Title. Click **Next >**.



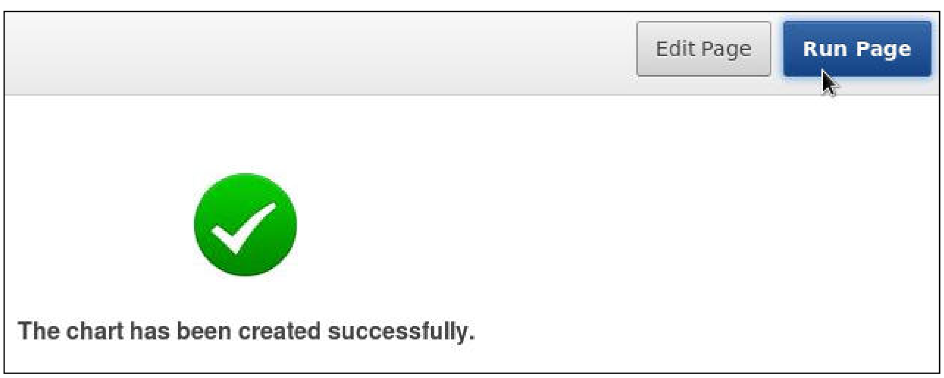
8) Enter the following SQL query (located in `/home/oracle/labs/files/lab17_1.txt)` and click **Next >**.

```
select null link,
department_name chart_label,
count(*) chart_value
from oehr_employees e,
oehr_departments d
where e.department_id = d.department_id
group by department_name
```

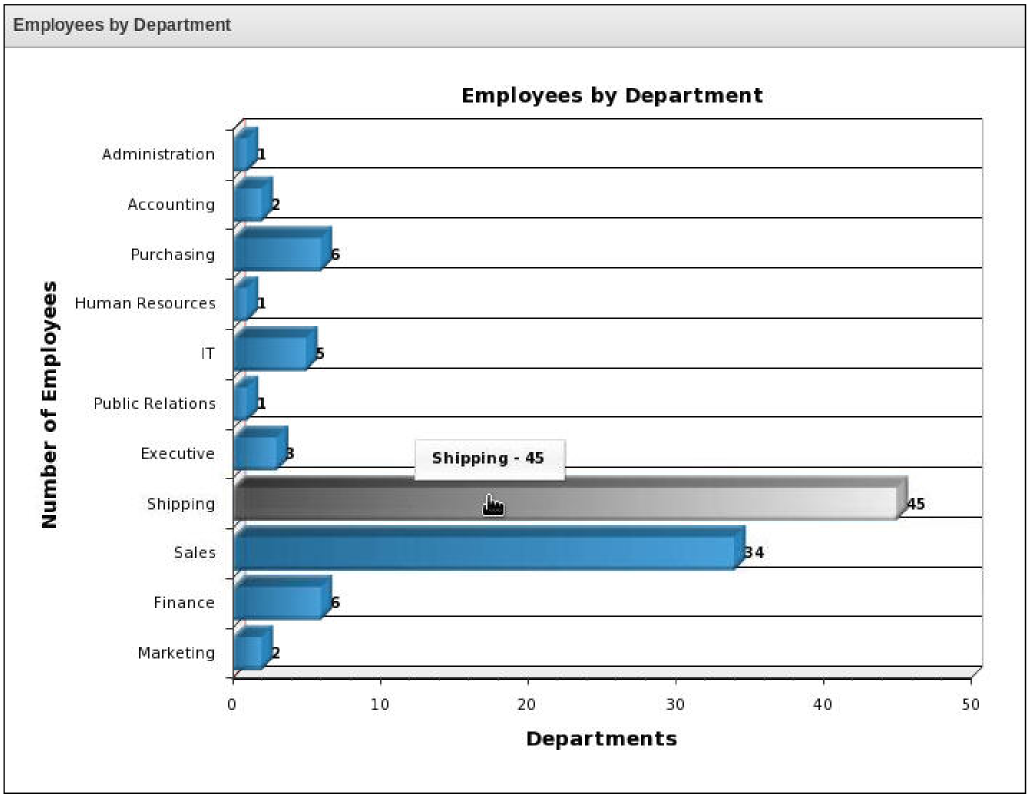


9) Click **Create**.

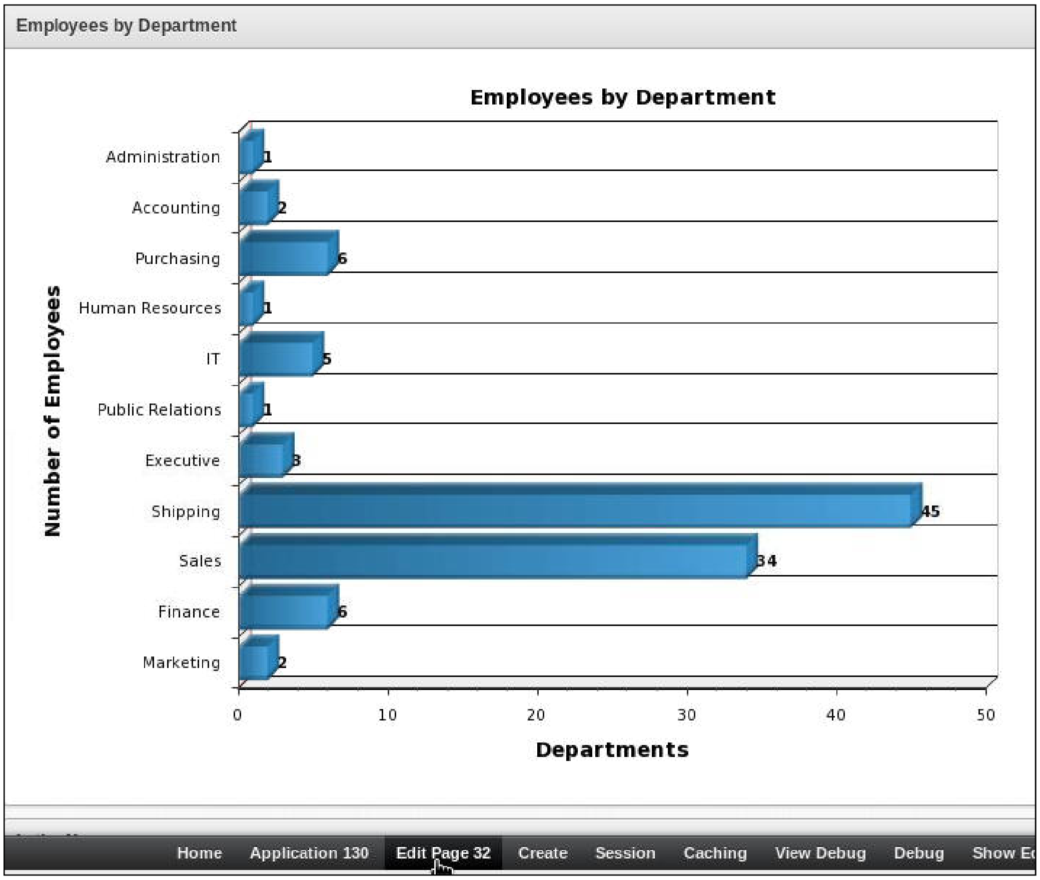
10) Run the page. Click **Run Page**.



11) Place your cursor over one of the bars to see the Department and its value.

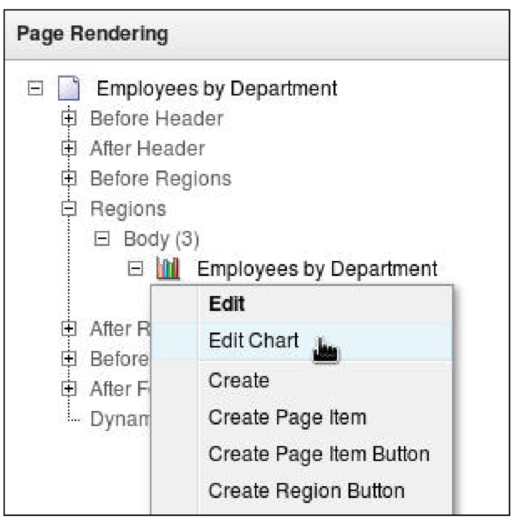


12) You want to change the 3D Bar Chart to a 3D Column Chart and alter the way it looks. Click the **Edit Page** link on the Developer toolbar.

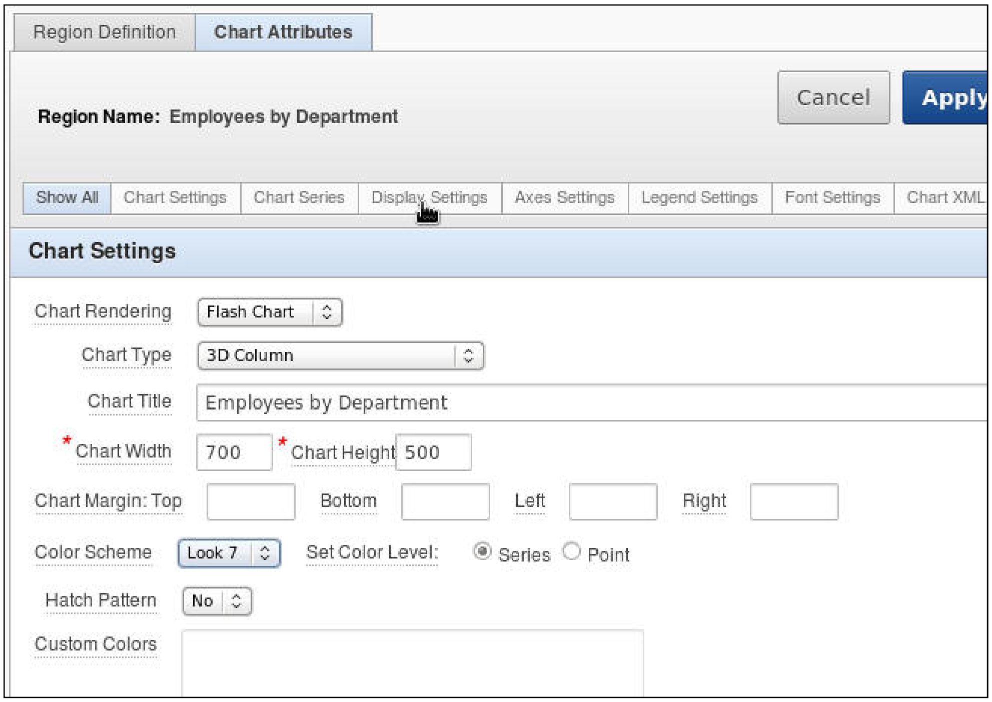


b. Modify your chart and change it to a 3D Column Chart with look 7. Add X-Axis scrollbars, enter `45` for X-Axis label rotation, reverse the X and Y axes titles, and test your page.

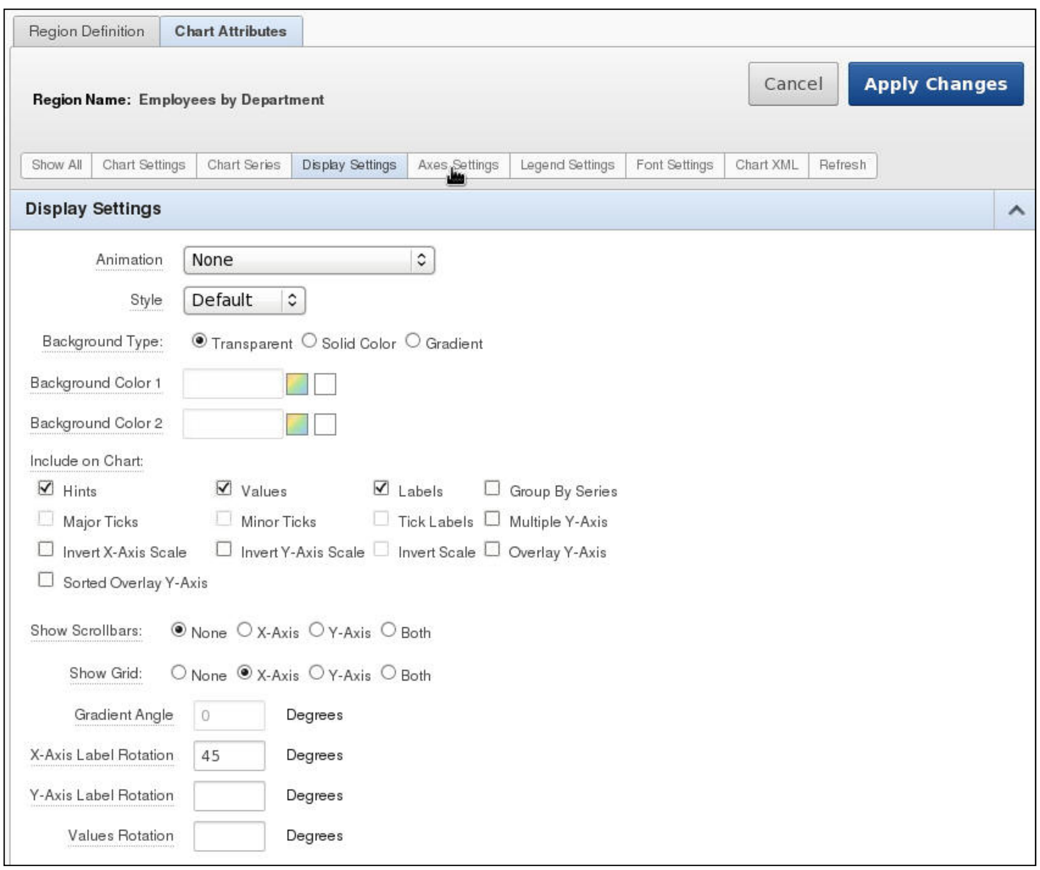
1) Right-click the **Employees by Department** region and select **Edit Chart**.



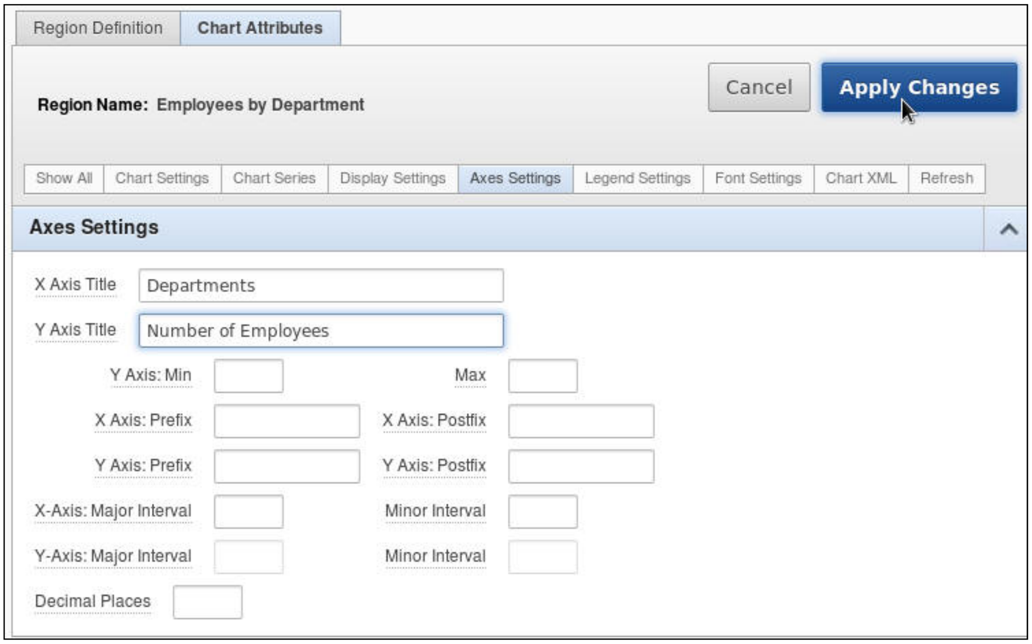
2) Select **3D Column** for Chart Type and select **Look 7** for Color Scheme, and then click the **Display Settings** tab.



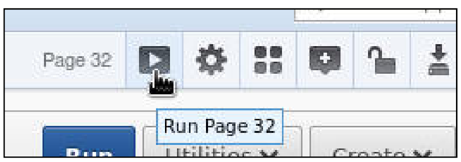
3) Enter **45** for X-Axis Label Rotation, and click the **Axes Settings** tab.



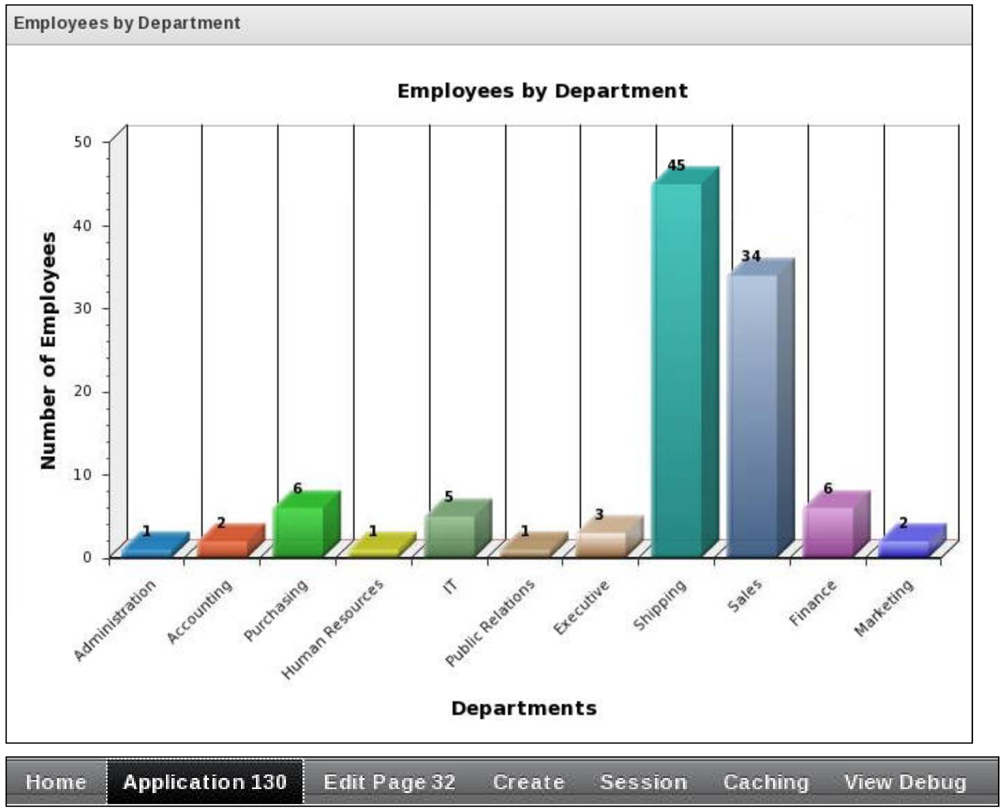
4) Because you changed the chart type, you must reverse the titles for the X and Y axes. Enter **Departments** for X Axis Title and **Number of Employees** for Y Axis Title. Click **Apply Changes**.



5) Run the page. Click the **Run Page** icon.



6) Notice that the chart is now a 3D Column chart, the look is different, and the X-Axes labels are rotated by 45 degrees.



Solution 17-2: Creating an HTML5 Chart for Mobile Applications

Overview

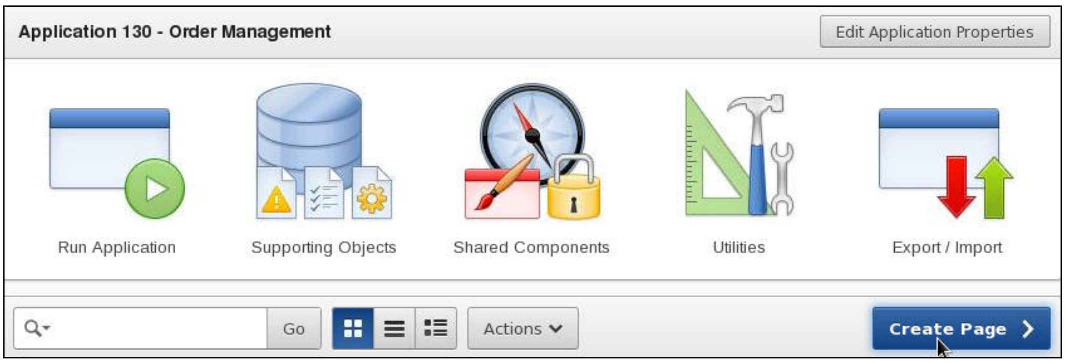
In this practice, you create an HTML5 chart for a mobile application.

Tasks

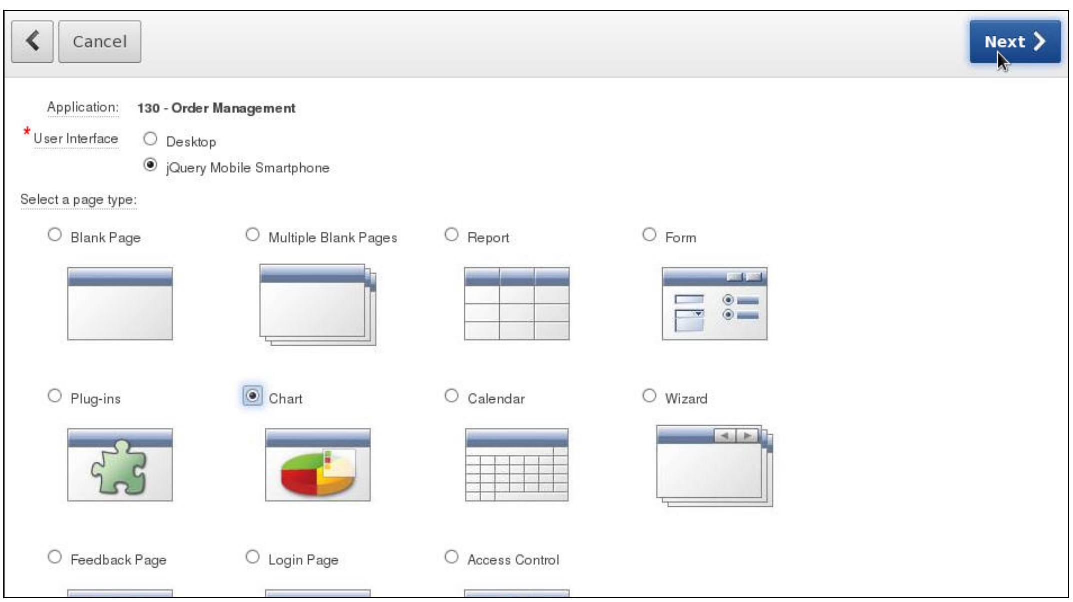
a. Create an HTML5 Pie and Doughnut chart. Use "Departments Chart" for the Title. Use the following query (located in `/home/oracle/labs/files/lab17_2_1.txt`). Test your mobile page.

```
select null link,
department_name chart_label,
count(*) chart_value
from oehr_employees e,
oehr_departments d
where e.department_id = d.department_id
group by department_name
```

1) Navigate to the Order Management application home page and click **Create Page**.

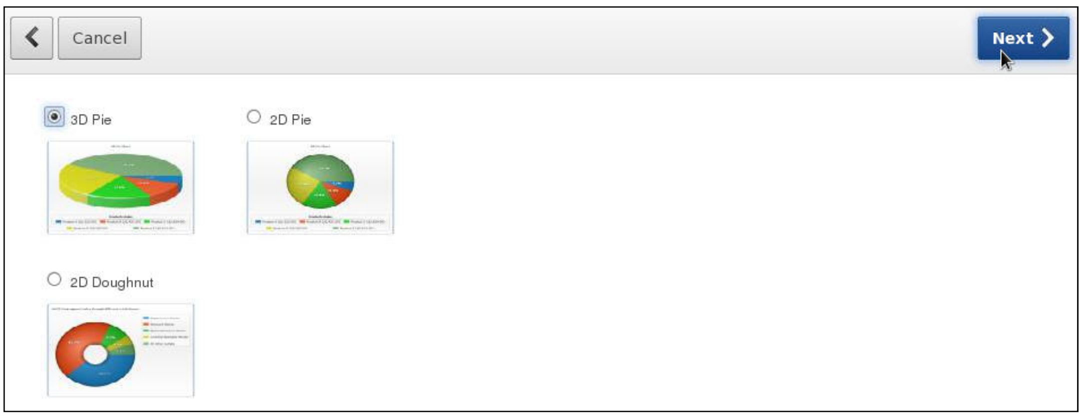


2) Select **jQuery Mobile Smartphone** for User Interface and select **Chart**. Click **Next >**.



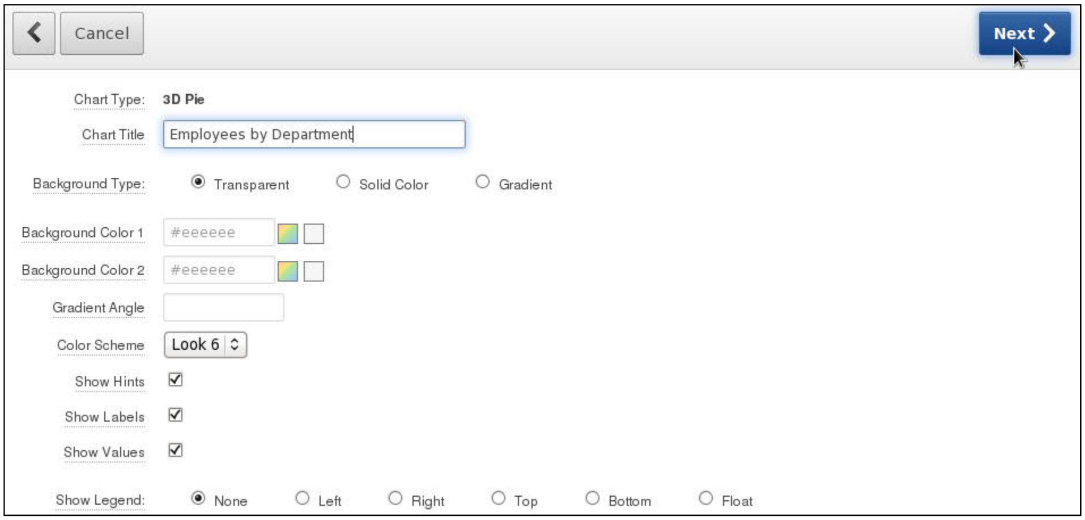
3) Select **Pie & Doughnut** and click **Next >**.

4) Select **3D Pie** and click **Next**.



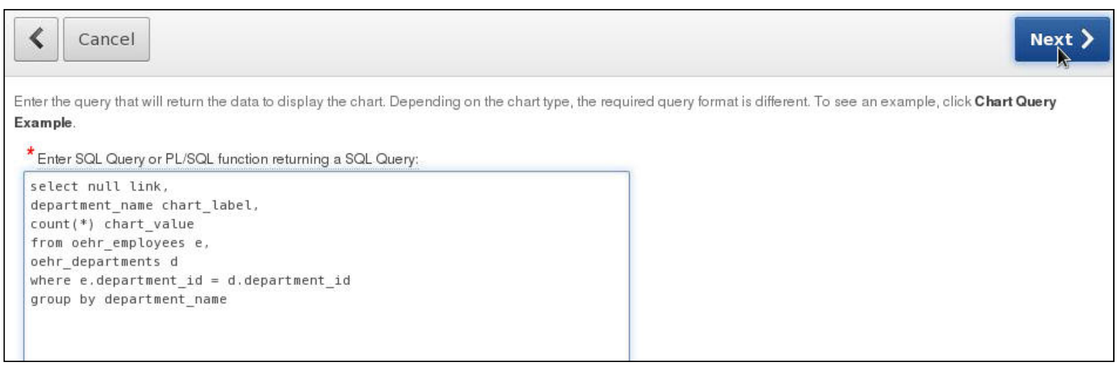
5) Select **Departments Chart** for both Page and Region names. Click **Next**.

6) Enter **Employees by Department** as the Chart title. Click **Next**.

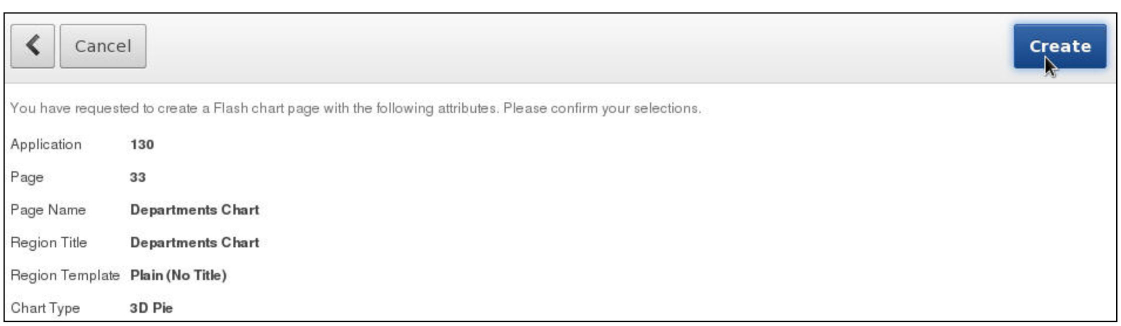


7) Enter the following SQL query (located in `/home/oracle/labs/files/lab17_2_1.txt)` and click **Next >**.

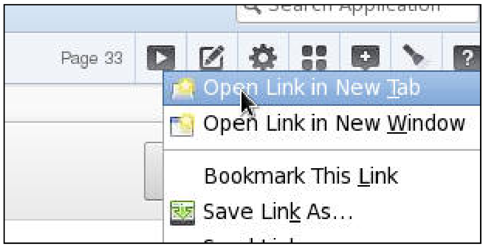
```
select null link,
department_name chart_label,
count(*) chart_value
from oehr_employees e,
oehr_departments d
where e.department_id = d.department_id
group by department_name
```



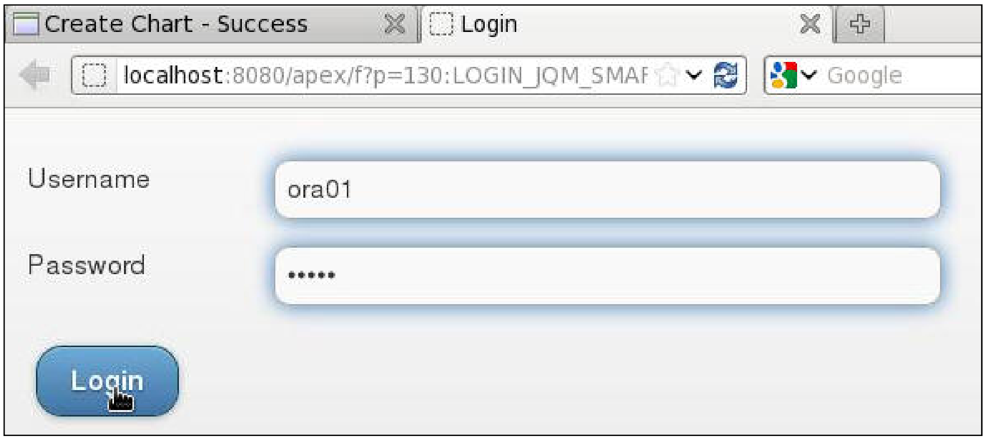
8) Click **Create**.



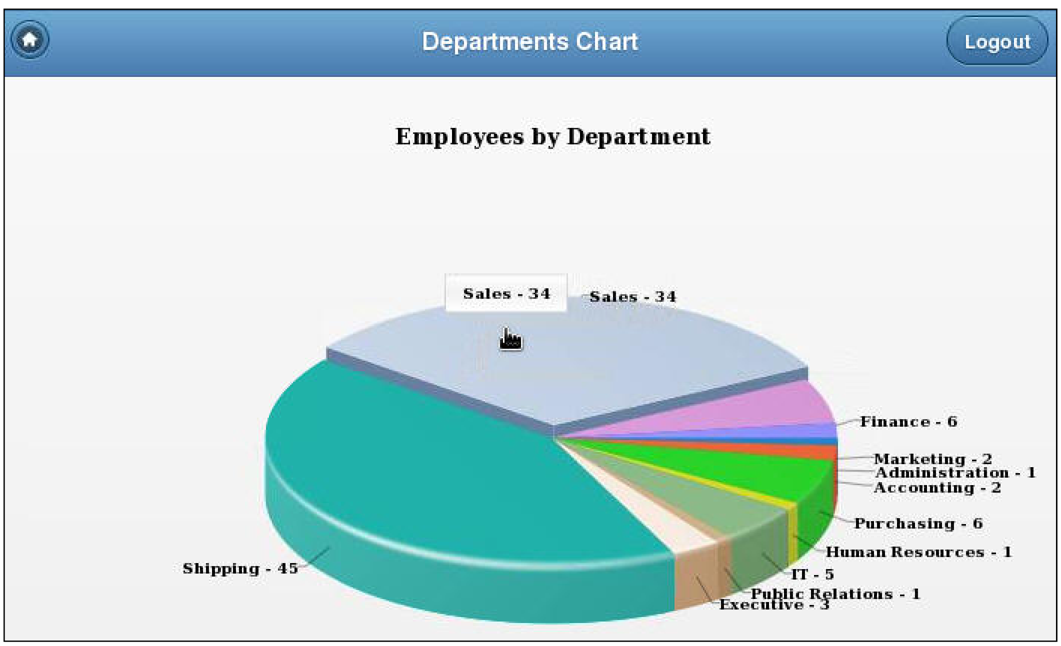
9) Run the page on a new tab.



10) You may be asked to log in. Enter your login credentials. Click **Login**.



11) The chart is displayed.



Solution 17-3: Creating Additional Charts

Overview

In this practice, you create some additional charts, such as a combined chart, a project Gantt chart, and a Gauge chart.

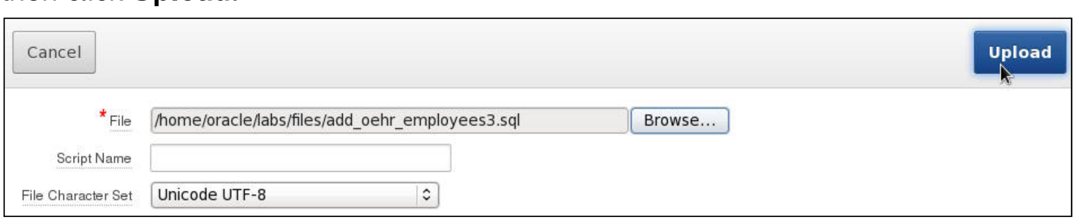
Tasks

a. You build a combined chart. First, you upload and run the **`add_oehr_employees3.sql`**. script to create a table called `oehr_employees3`.

1) Navigate to **SQL Workshop** > **SQL Scripts**.

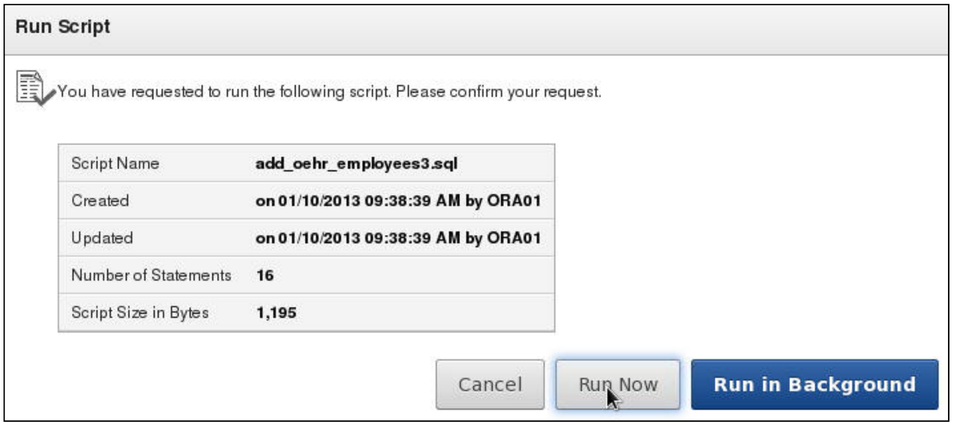
2) Click **Upload**.

3) Browse and select `home/oracle/labs/files/add_oehr_employees3.sql`, then click **Upload**.



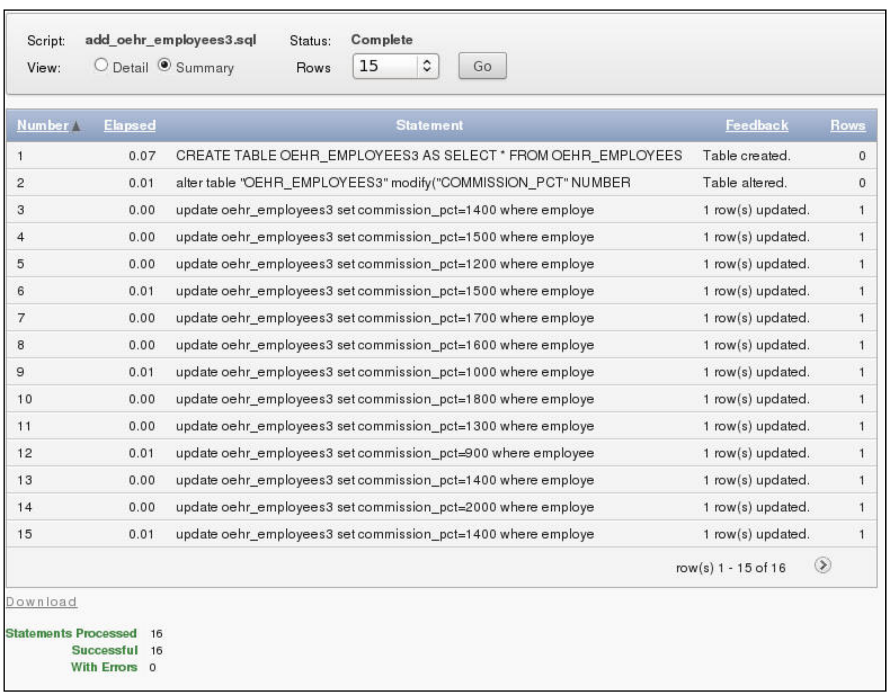
4) Click **Run** to create the `oehr_employees3` table.

5) Click **Run Now**.



6) Click **View Results**.

7) The results are displayed.



Script: add_oehr_employees3.sql Status: Complete

View: Detail Summary Rows 15 Go

Number	Elapsed	Statement	Feedback	Rows
1	0.07	CREATE TABLE OEHR_EMPLOYEES3 AS SELECT * FROM OEHR_EMPLOYEES	Table created.	0
2	0.01	alter table "OEHR_EMPLOYEES3" modify("COMMISSION_PCT" NUMBER	Table altered.	0
3	0.00	update oehr_employees3 set commission_pct=1400 where employ	1 row(s) updated.	1
4	0.00	update oehr_employees3 set commission_pct=1500 where employ	1 row(s) updated.	1
5	0.00	update oehr_employees3 set commission_pct=1200 where employ	1 row(s) updated.	1
6	0.01	update oehr_employees3 set commission_pct=1500 where employ	1 row(s) updated.	1
7	0.00	update oehr_employees3 set commission_pct=1700 where employ	1 row(s) updated.	1
8	0.00	update oehr_employees3 set commission_pct=1600 where employ	1 row(s) updated.	1
9	0.01	update oehr_employees3 set commission_pct=1000 where employ	1 row(s) updated.	1
10	0.00	update oehr_employees3 set commission_pct=1800 where employ	1 row(s) updated.	1
11	0.00	update oehr_employees3 set commission_pct=1300 where employ	1 row(s) updated.	1
12	0.01	update oehr_employees3 set commission_pct=900 where employee	1 row(s) updated.	1
13	0.00	update oehr_employees3 set commission_pct=1400 where employ	1 row(s) updated.	1
14	0.00	update oehr_employees3 set commission_pct=2000 where employ	1 row(s) updated.	1
15	0.01	update oehr_employees3 set commission_pct=1400 where employ	1 row(s) updated.	1

row(s) 1 - 15 of 16

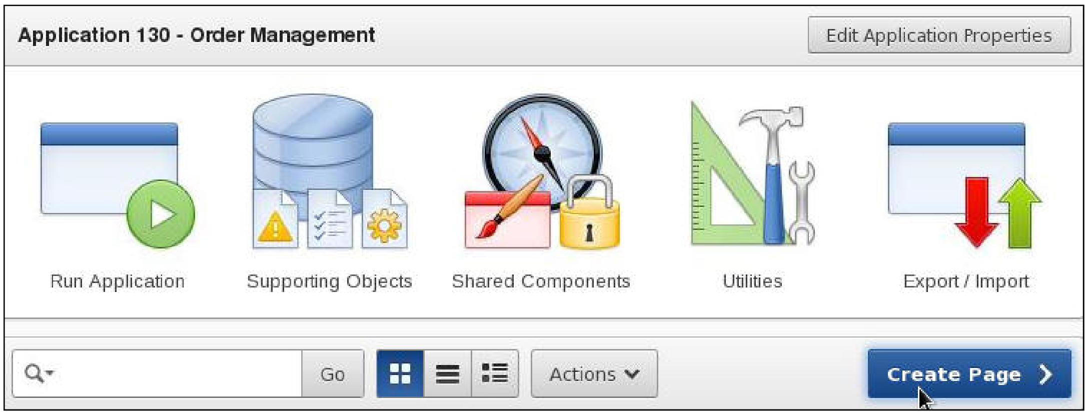
Download

Statements Processed 16
Successful 16
With Errors 0

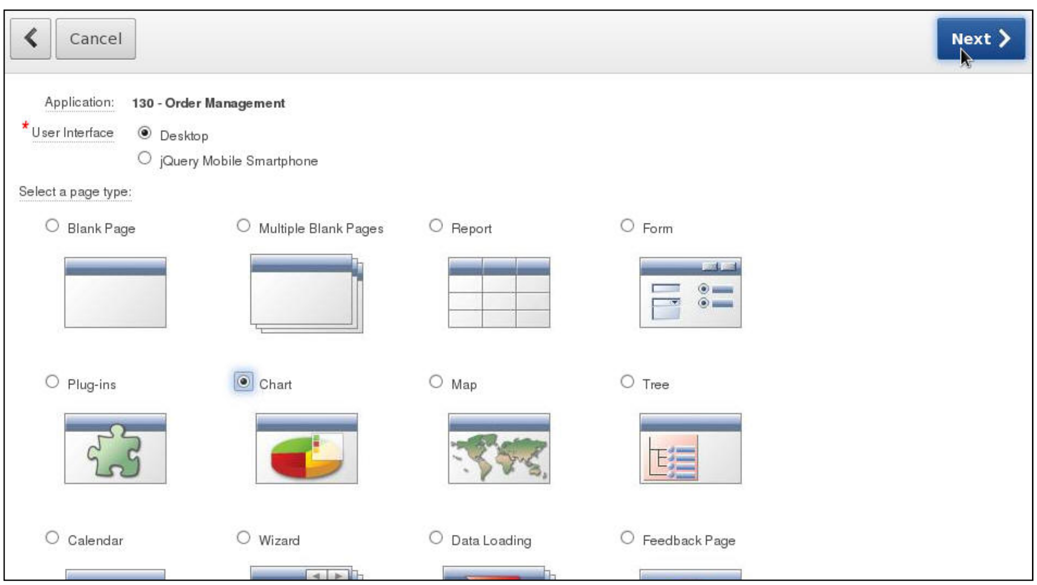
b. Create a 3D Column Flash Chart named **`Salary Commission Combined Chart.`** Use the following SQL Query (located in the `/home/oracle/labs/files/lab17_3_1.txt` file). Specify **`Employee`** for X-Axis Title and **`Salary`** for Y-Axis title. Edit the chart by changing the Series attribute to Line.

```
SELECT NULL LINK,
       LAST_NAME LABEL,
       SALARY "Salary",
       COMMISSION_PCT "COMMISSION"
FROM   OEHR_EMPLOYEES3
WHERE MANAGER_ID = 100
Order by LAST_NAME
```

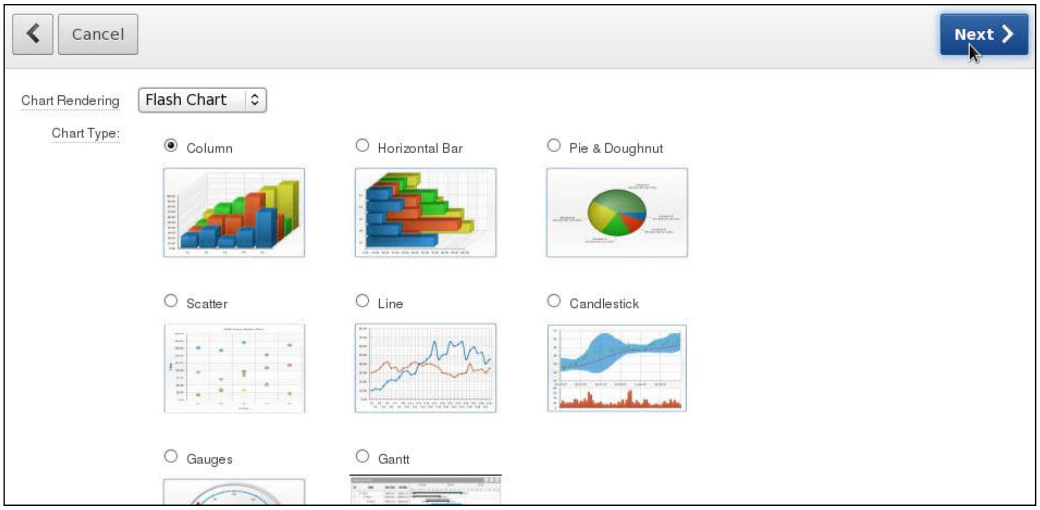
1) Navigate to your application. In your application, click **Create Page**.



2) Select **Chart** page type and click **Next >**.

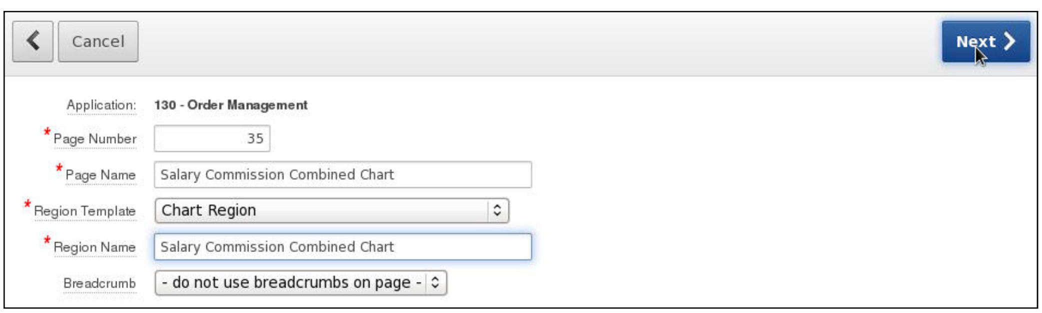


3) Select **Flash Chart** and select **Column**. Click **Next**.

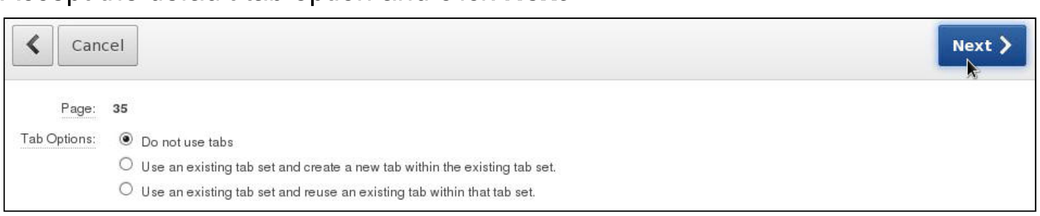


4) Select **3D Column** and click **Next**.

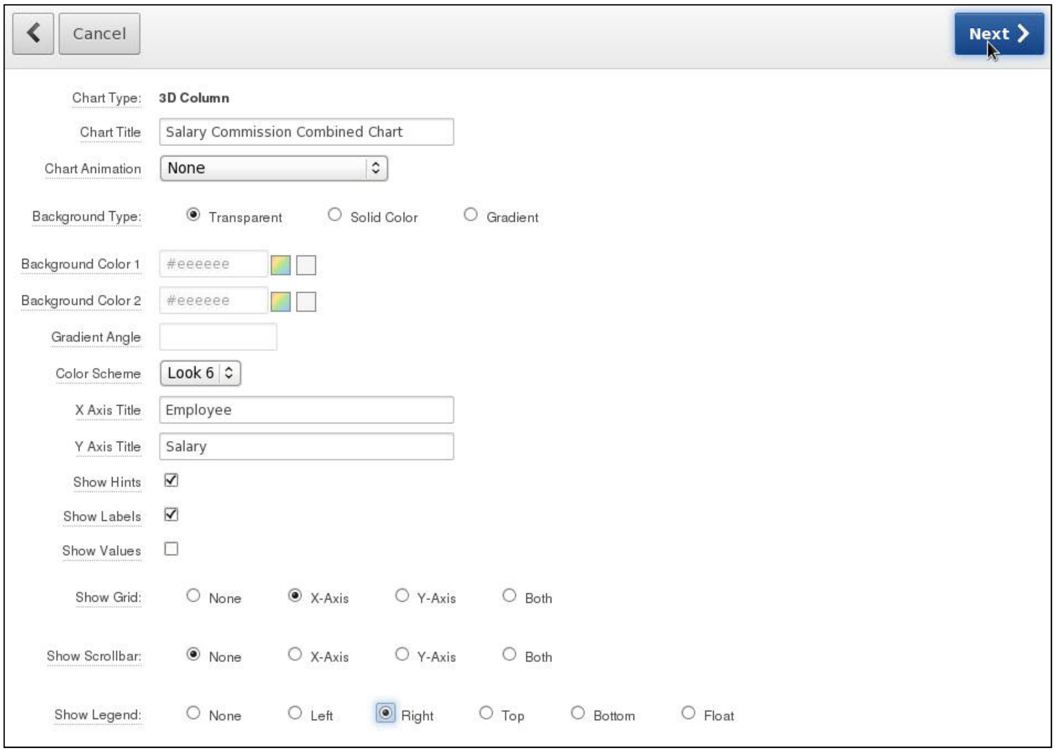
5) Enter `Salary Commission Combined Chart` for Page Name and Region Name. Click **Next >**.



6) Accept the default tab option and click **Next**.

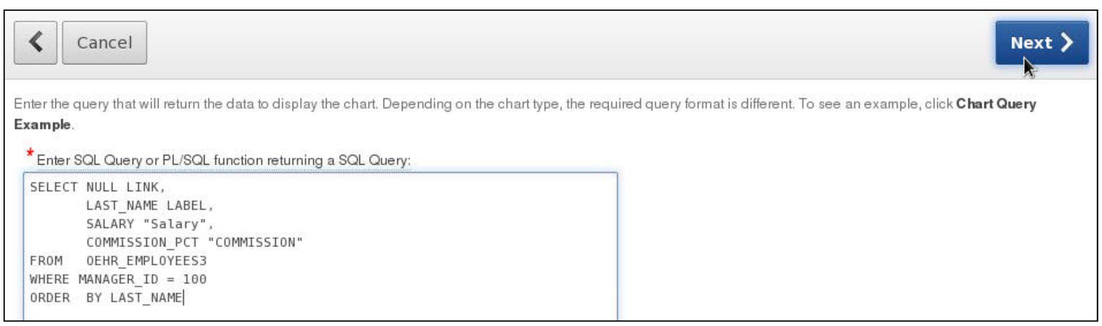


7) Enter **`Salary Commission Combined Chart`** for Chart Title. Deselect the **Show Values** check box. Specify **`Employee`** for X-Axis Title and **`Salary`** for Y-Axis title. For Show Legend, select the **Right** option. Click **Next**.



8) Use the following SQL query (located in **`home/oracle/labs/files/lab17_3_1.txt`**) to enter in the SQL Query text area. Click **Next**.

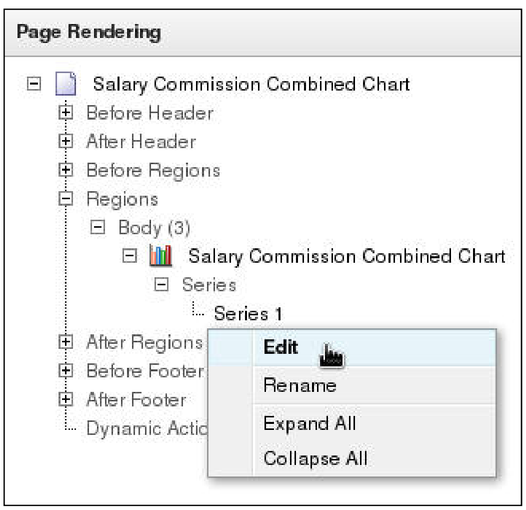
```
SELECT NULL LINK,
       LAST_NAME LABEL,
       SALARY "Salary",
       COMMISSION_PCT "COMMISSION"
FROM   OEHR_EMPLOYEES3
WHERE MANAGER_ID = 100
ORDER  BY LAST_NAME
```



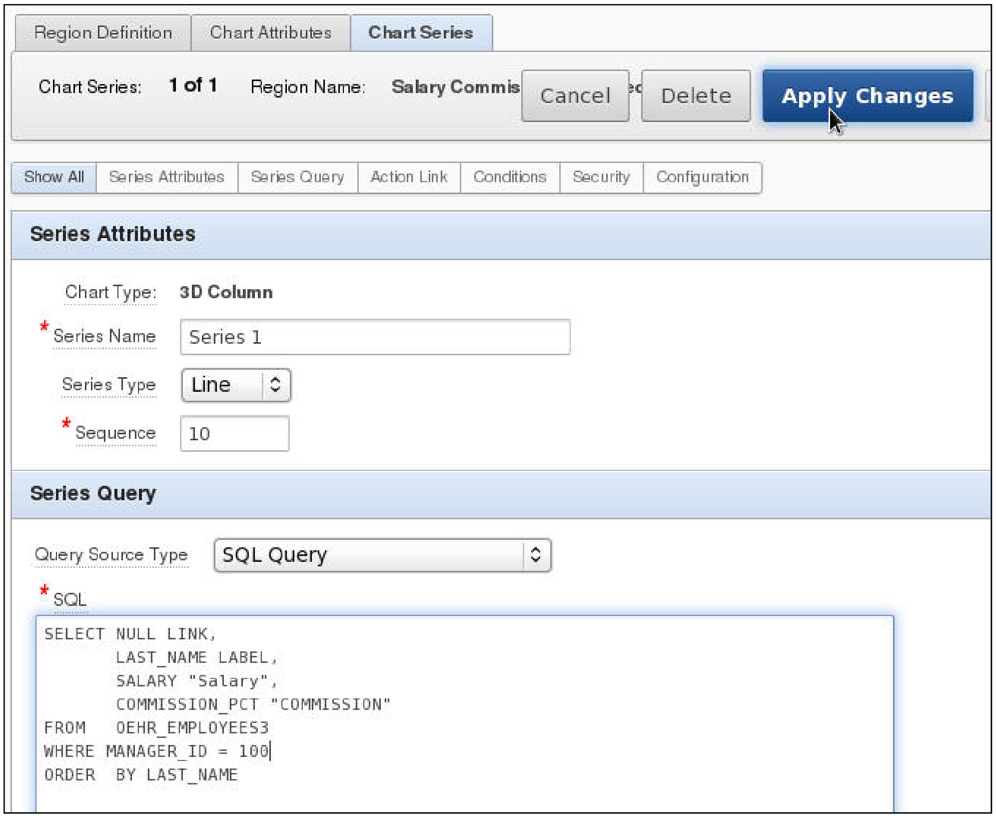
9) Click **Create**.

10) Click **Edit Page**.

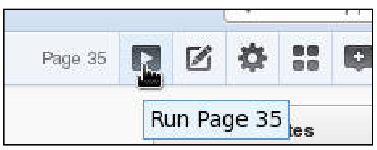
11) Under Regions node, right-click **Series > Series 1** and select **Edit**.



12) Under Series Attributes, select **Line** for Series Type. Click **Apply Changes**.



13) Run the page.

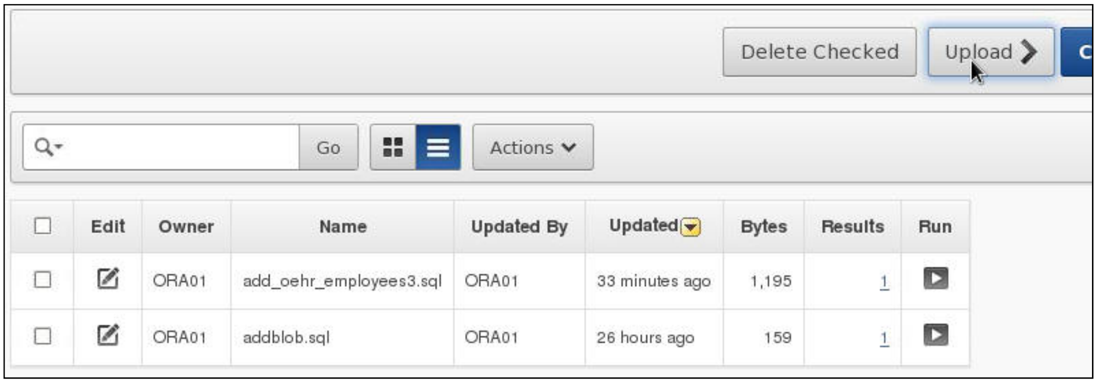


14) You can now see a combination of salary as Line and commission as Bar charts. Place your cursor over one of the points in the line chart. Click the **Application <*n*>** link on the Developer toolbar.

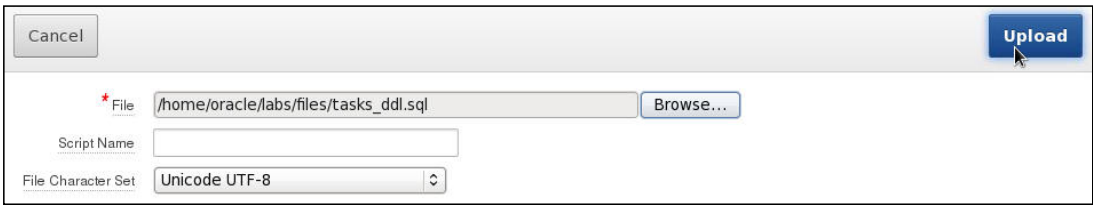
c. Upload and run the **`tasks_ddl.sql`** script to create a table called `tasks1`.

1) You first want to run a script to create the `Tasks1` table. Navigate to **SQL Workshop > SQL Scripts**.

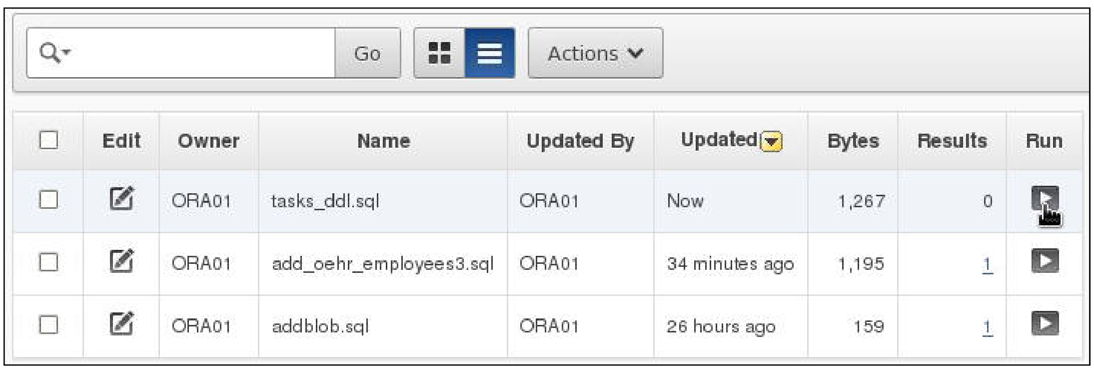
2) Click **Upload**.



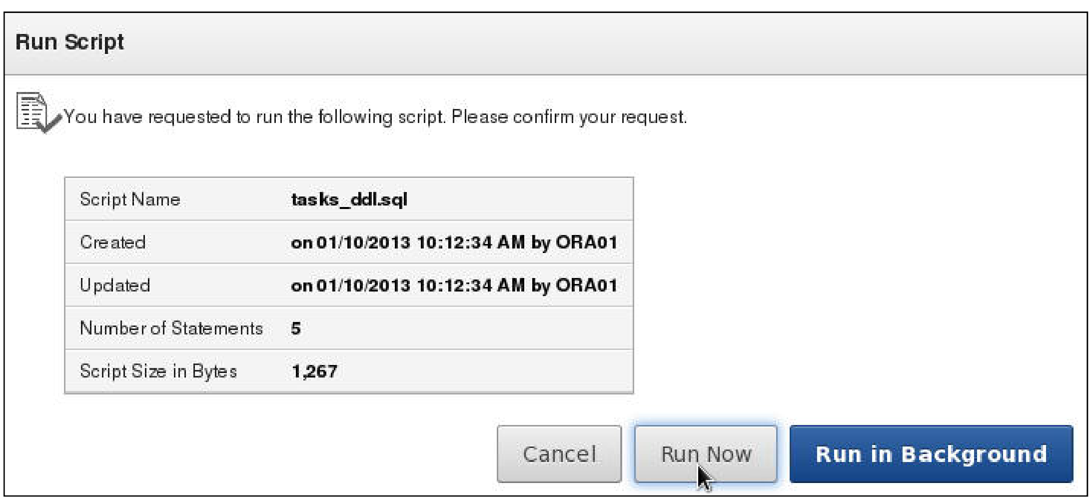
3) Click **Browse** and select **`home/oracle/labs/files/tasks_ddl.sql`**. Click **Upload**.



4) Run the script.

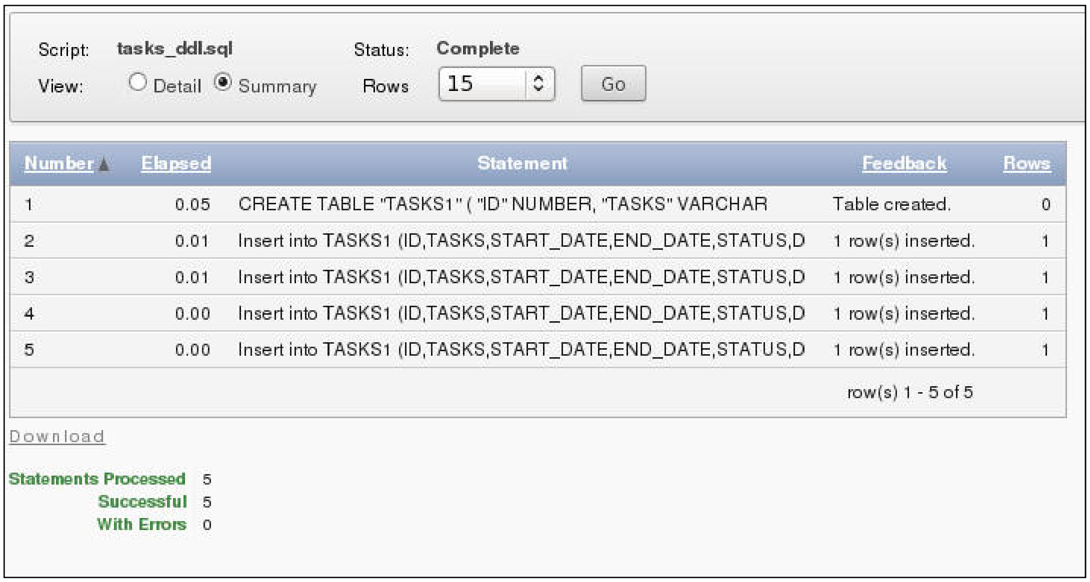


Click **Run Now**.



Click **View Results**.

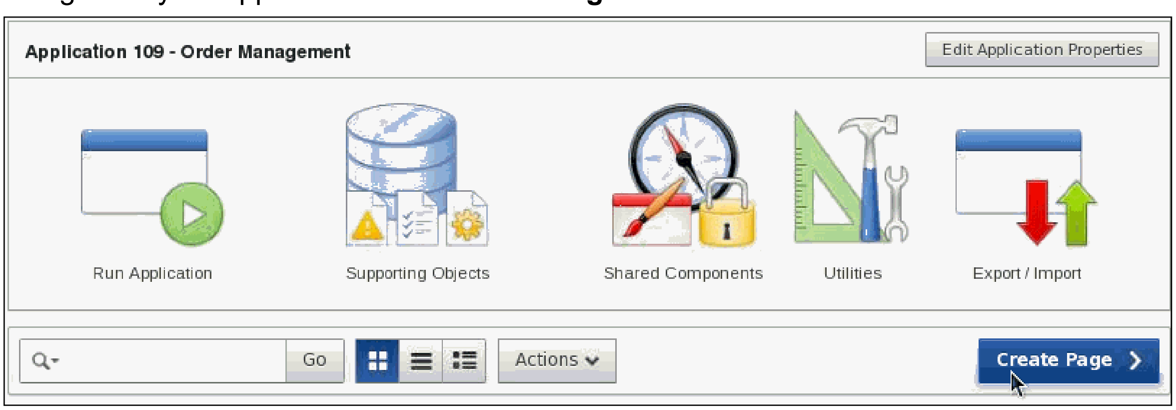
5) The results are displayed.



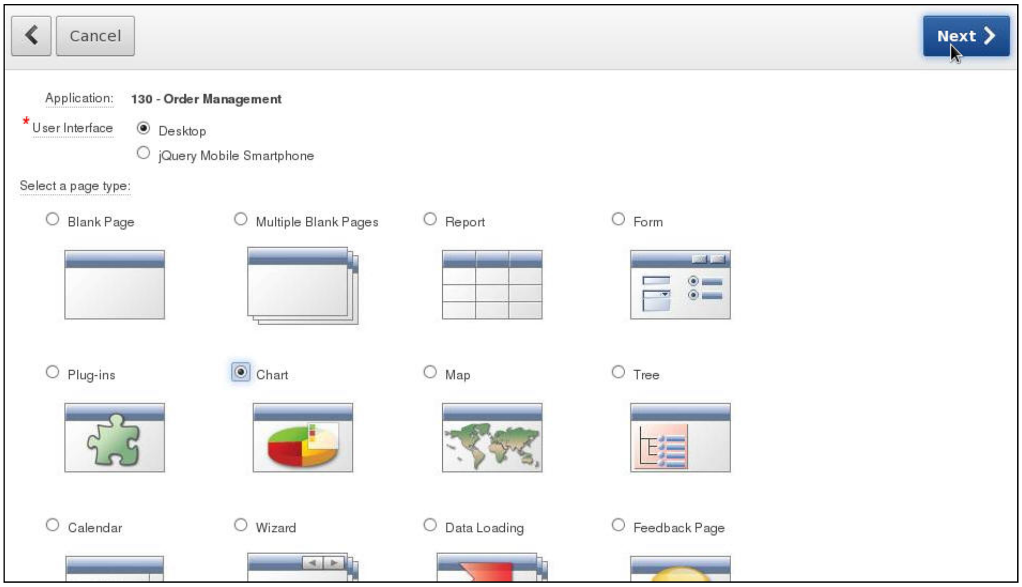
d. Create a Project Gantt Chart that uses the following SQL Query (located in the `/home/oracle/labs/files/lab17_3_2.txt` file).

```
SELECT NULL LINK,
       tasks task_name, id, parent_id,
       START_DATE, END_DATE,
       STATUS,
       START_DATE-3, END_DATE+2
from TASKS1
```

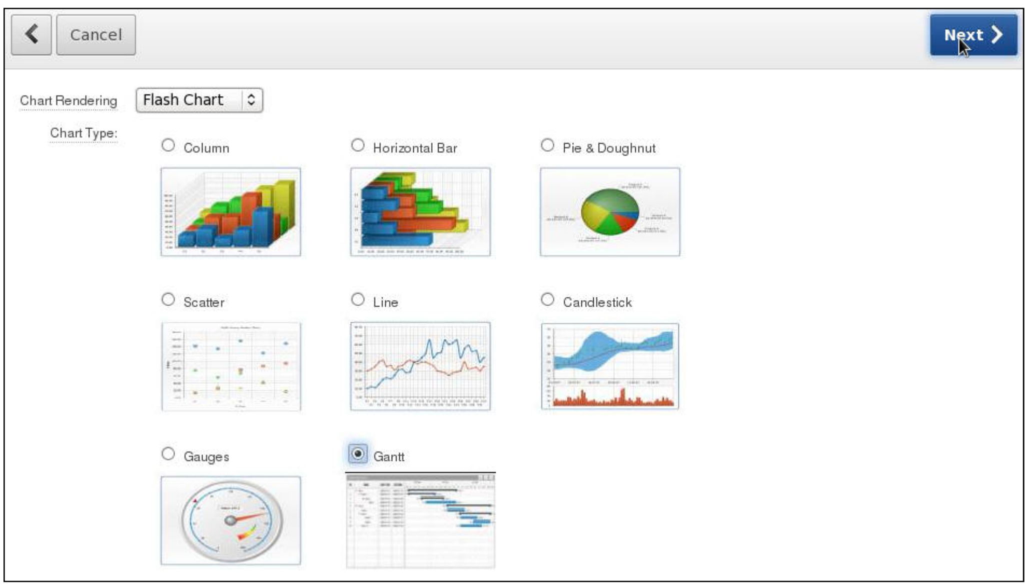
1) Navigate to your application. Click **Create Page**.



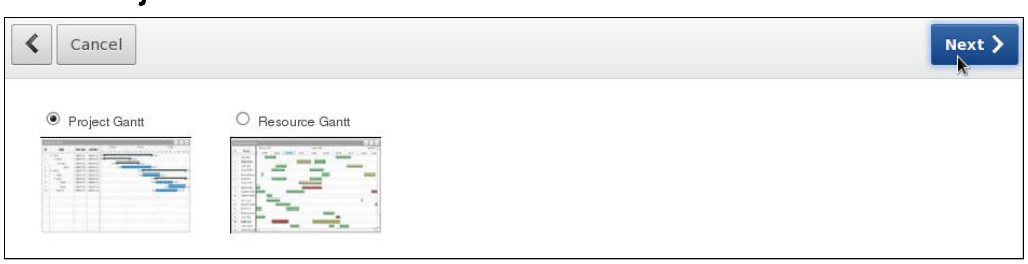
2) Select the **Chart** page type and click **Next >**.



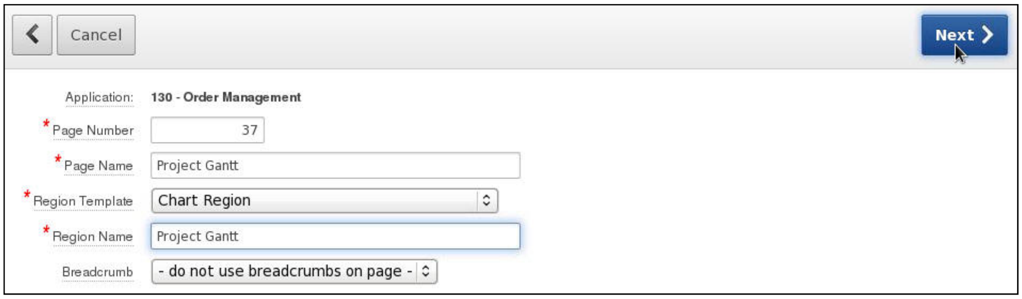
3) Select **Flash Chart** and select **Gantt** .Click **Next >**.



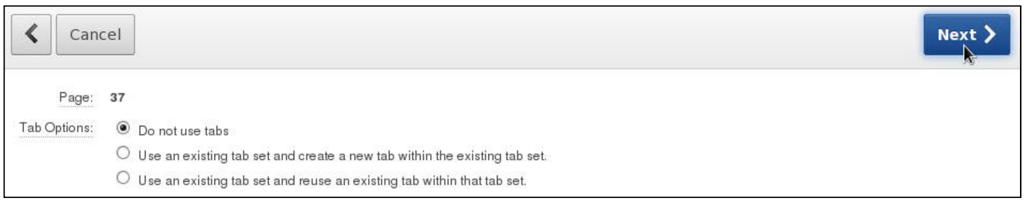
4) Select **Project Gantt** and click **Next**.



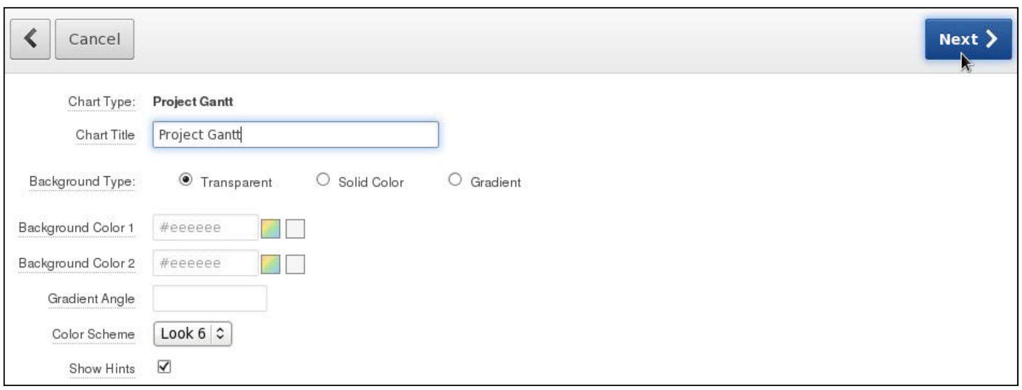
5) Enter **Project Gantt** for Page Name and Region name. Click **Next**.



6) Accept the default tab option and click **Next**.

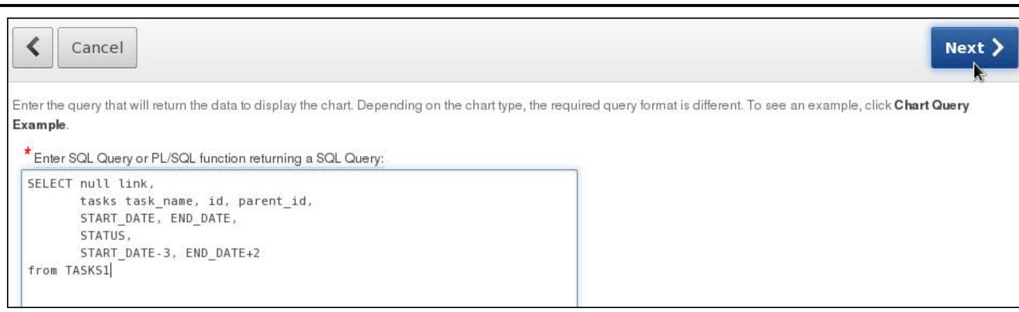


7) Enter `Project Gantt` for Chart Title. Click **Next**.



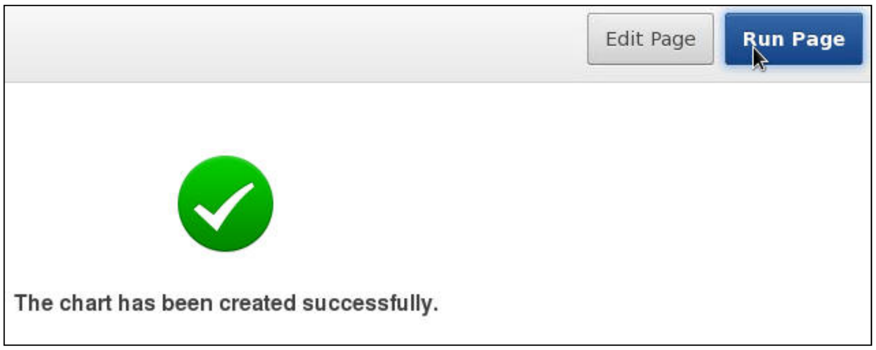
8) Enter the following SQL and click **Next**. Use **`home/oracle/labs/files/lab17_3_2.txt`**.

```
SELECT null link,
       tasks task_name, id, parent_id,
       START_DATE, END_DATE,
       STATUS,
       START_DATE-3, END_DATE+2
from TASKS1
```

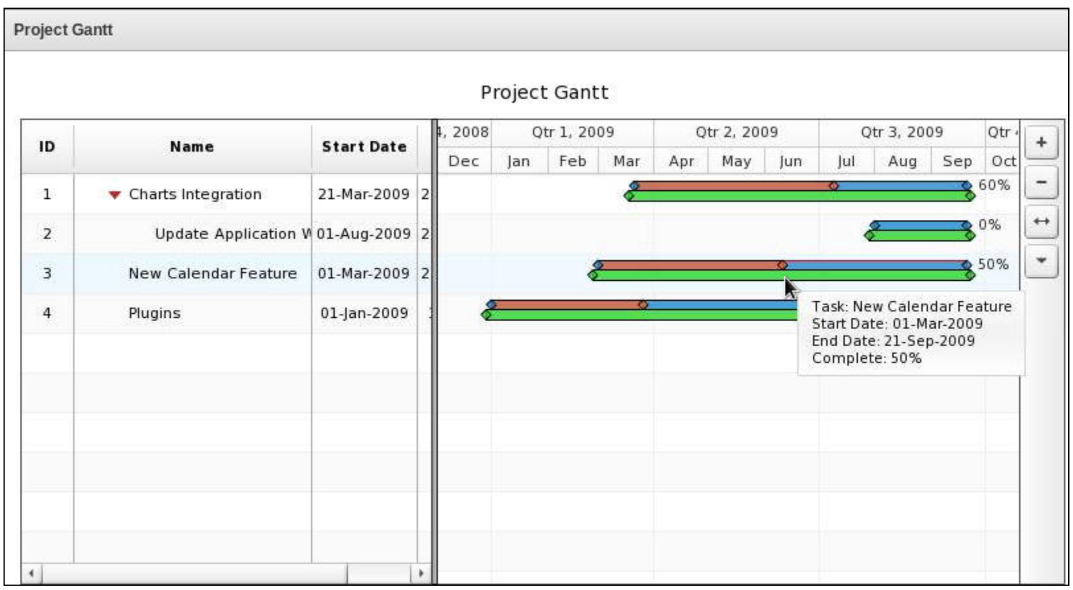


9) On the Confirmation page, click **Create**.

10) Run the page.



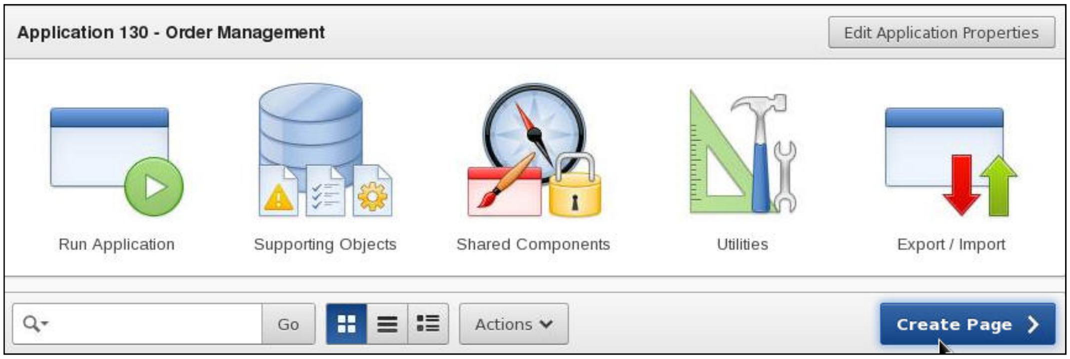
11) The Gantt chart is displayed. If you place your cursor over one of the tasks, you will see the details. Click the **Application <*n*>** link on the Developer toolbar.



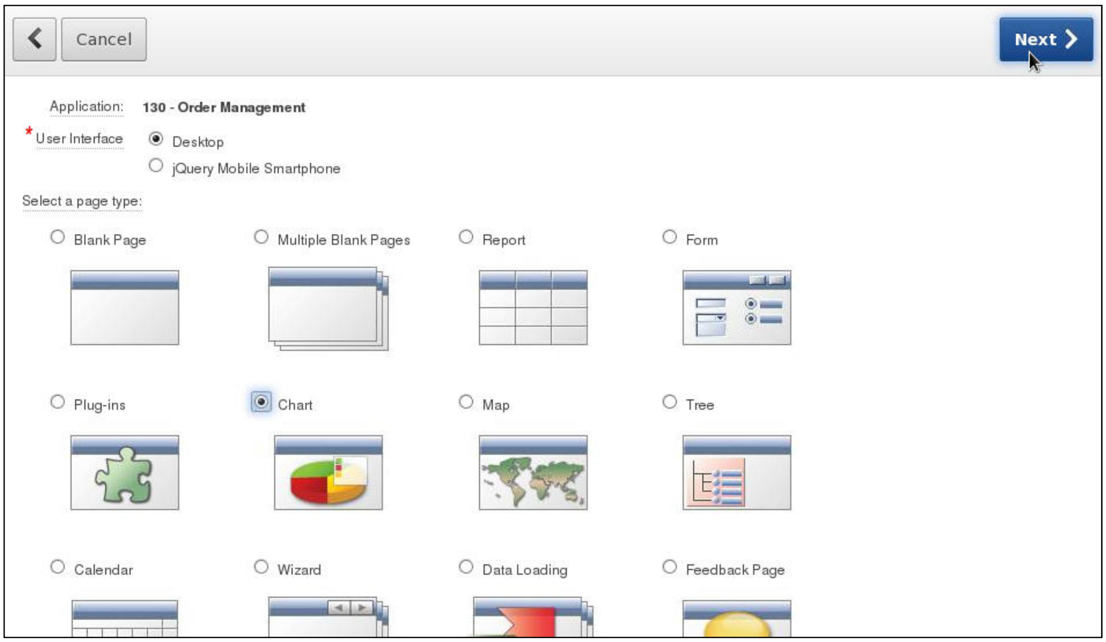
e. Create a Salary Gauge chart for employees in department 80. Use the following SQL Query (located in the `/home/oracle/labs/files/lab17_3_3.txt` file):

```
select sum(case when salary <10000 then 0 else 1 end) value,
       count(*) max_value
from oehr_employees
where department_id=80
```

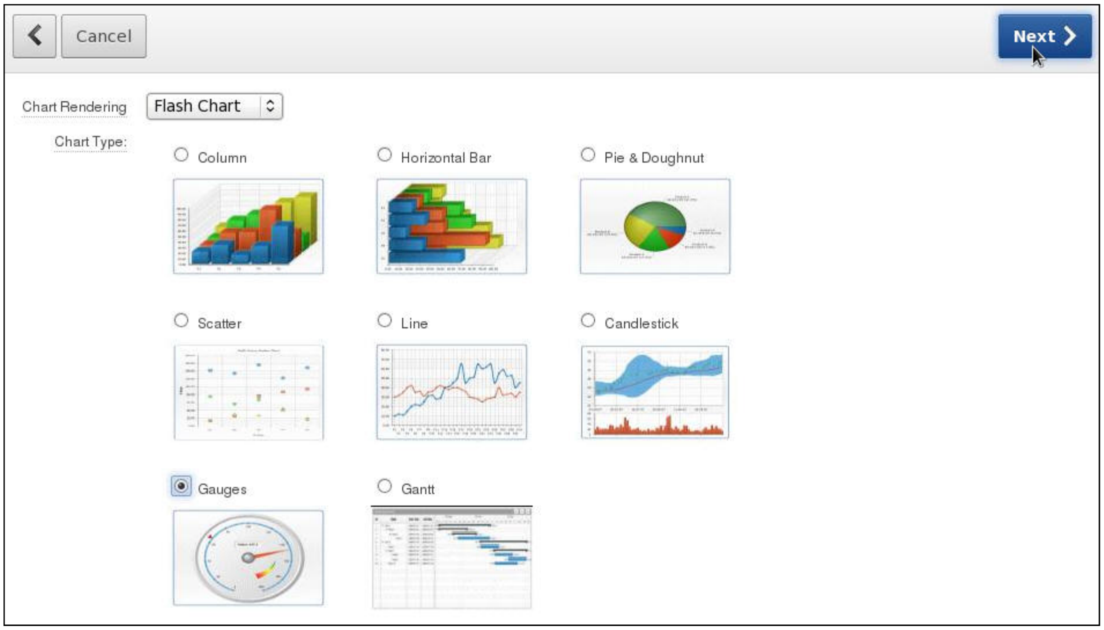
1) Navigate to your application. Click **Create Page >**.



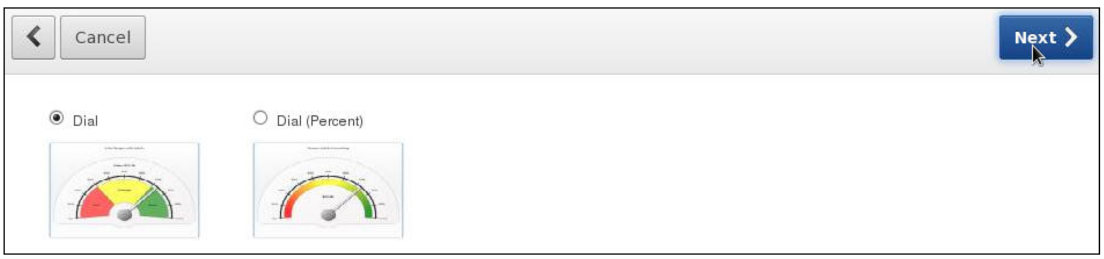
2) Select **Chart** page type and click **Next >**.



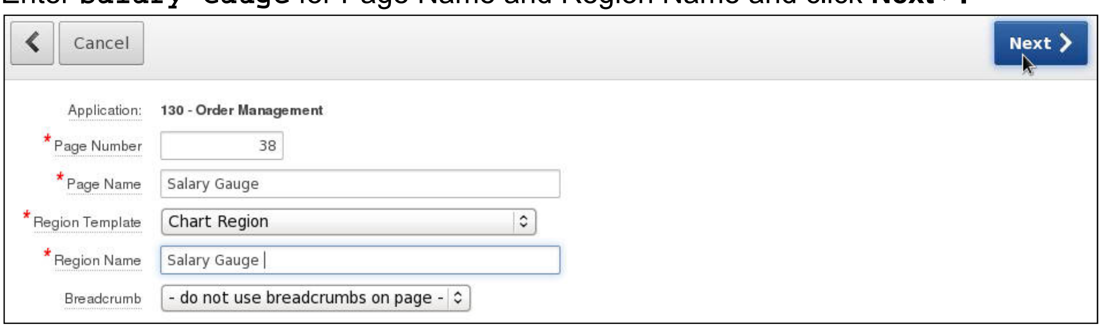
3) Select **Flash Chart** and select **Gauges**. Click **Next >**.



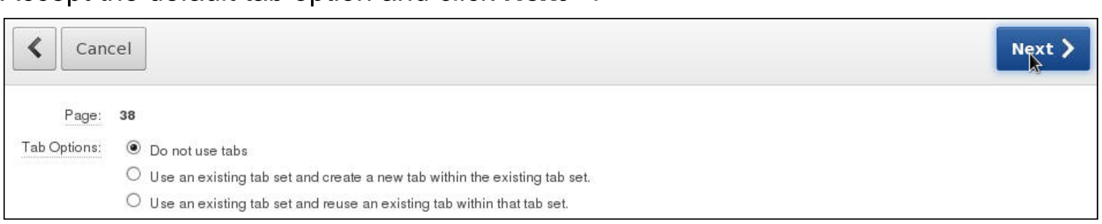
4) Select **Dial** and click **Next >**.



Enter `Salary Gauge` for Page Name and Region Name and click **Next >.**



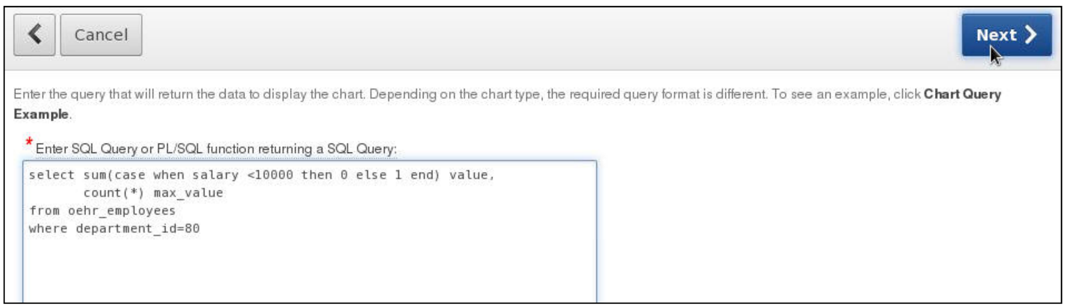
5) Accept the default tab option and click **Next >**.



6) Enter **`Salary Gauge`** for Chart Title, **`Solid Color`** for Background Type, and enter **`#FFFFF`** for Background Color 1. Click **Next**.

7) Enter the following SQL (located in `home/oracle/labs/files/lab17_3_3.txt`) and click **Next >**.

```
select sum(case when salary <10000 then 0 else 1 end) value,
       count(*) max_value
from oehr_employees
where department_id=80
```



8) On the confirmation page, click **Create**.

9) Run the page.

10) The dial is displayed. Click the **Edit Page** link on the Developer toolbar.

Practices for Lesson 18: Adding Calendars and Trees

Chapter 18

Practices for Lesson 18: Overview

Practices Overview

In these practices, you create a calendar page that displays the order total per day. You also create a link to a page so that you can edit the Order selected. To create Trees, you create a new page with a tree region and link to the Dynamic Employee Details page. Add a button on the Dynamic Employee Details page to link back to the tree page.

Practice 18-1: Creating a Calendar

Overview

In this practice, you create a calendar page that displays the order total per day. You also create a page to link to so you can change an order.

Assumptions

You have performed the previous practices.

Tasks

a. Execute the following command to update ORDER_DATE in the OEHR_ORDERS table to make it more current (located in the `/home/oracle/labs/files/lab18_1_1.txt` file).

```
UPDATE oehr_orders SET order_date = order_date + 4615
```

b. Create a new Easy Calendar calendar page called Order Calendar. Base the calendar on the `OEHR_ORDERS` table, and set the Date Column to `ORDER_DATE` and Display Column to `ORDER_TOTAL`. Allow for drag and drop order total to different dates. Create a new edit page so you can edit all the columns in the `OEHR_ORDERS` table. Test your pages.

c. Create a new page called Mobile Calendar for mobile application. Select jQuery Mobile Smartphone User Interface to create this page. Select SQL Calendar. Enter a `SELECT` SQL statement (located in the `/home/oracle/labs/files/lab18_1_2.txt` file) to display the name column and start date column from the **`TASKS`** table. Assign the Link Target to create a new edit page, which contains all the columns from the **`TASKS`** table. Test your page.

```
select id, project, task_name, start_date, cost from tasks
```

Practice 18-2: Creating a Tree Whose Nodes Link to a Different Page

Overview

In this practice, you create a new page with a tree region and link to the Dynamic Employee Details page. Add a button on the Dynamic Employee Details page to link back to the tree page.

Assumptions

You have performed the previous practices.

Tasks

a. Create a tree page called "Employees by Manager" that uses the default tree template. Select the appropriate options in the wizard so that the current query (located in `/home/oracle/labs/files/lab18_2_1.txt`) looks as follows: (<*n*> is the page number for the Dynamic Employee Details page.)

```
select case when connect_by_isleaf = 1 then 0
            when level = 1             then 1
            else                           -1
       end as status,
       level,
       "LAST_NAME" as title,
       null as icon,
       "EMPLOYEE_ID" as value,
       null as tooltip,
       null as link
from "ORA01"."OEHR_EMPLOYEES"
start with "MANAGER_ID" is null
connect by prior "EMPLOYEE_ID" = "MANAGER_ID"
order siblings by "LAST_NAME"
```

b. Add a button to the Employee Details page called Employee Hierarchy that navigates to the "Employees by Manager" tree page.

Solution 18-1: Creating a Calendar

Overview

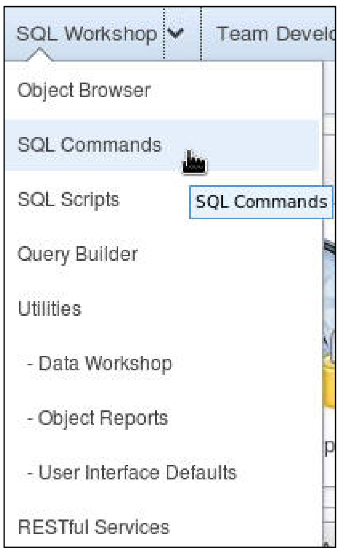
In this solution, you create a calendar on the List of Orders page that displays the order total per day. You also link to the Master Detail page for the Order selected.

Tasks

a. Execute the following command to update the ORDER_DATE in the OEHR_ORDERS table to make them more current (located in the `/home/oracle/labs/files/lab18_1_1.txt` file).

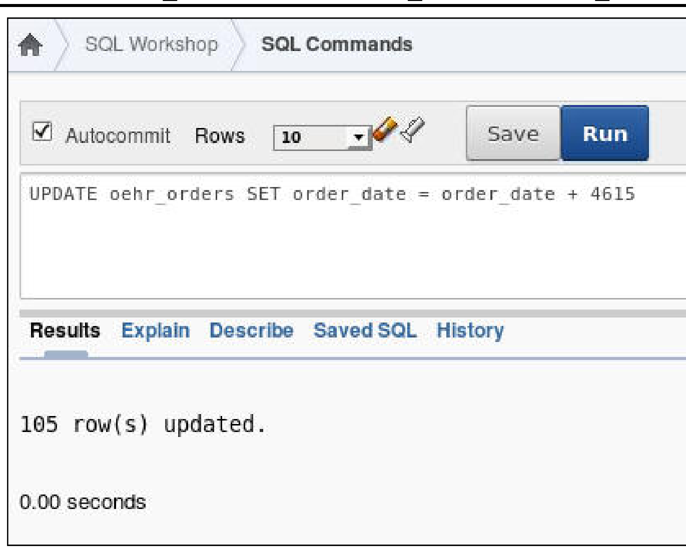
```
UPDATE oehr_orders SET order_date = order_date + 4615
```

1) In order to see data on your calendar, you may need to modify the dates. Select **SQL Workshop** > **SQL Commands**.



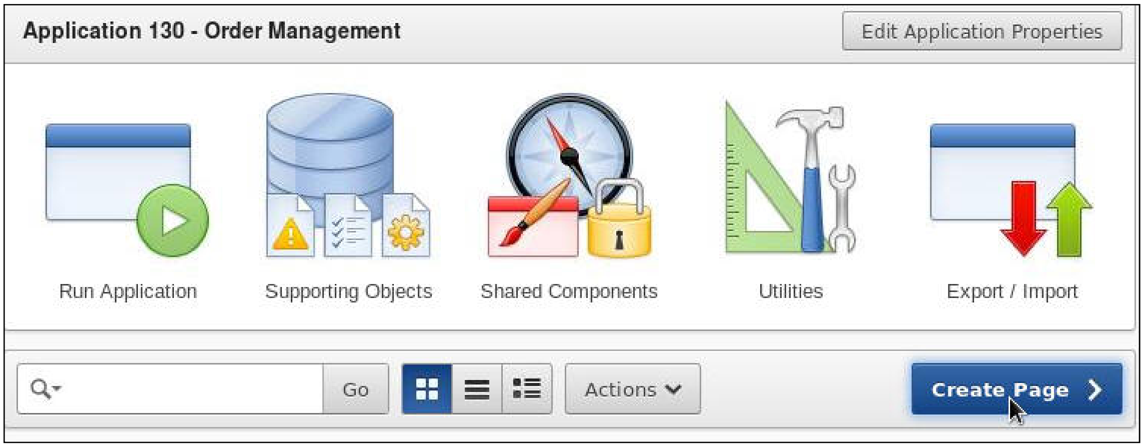
2) Enter the following SQL statement and click the **Run** button. This statement modifies the order dates in the `OEHR_ORDERS` table so they contain more recent dates that will appear in the calendar created in the next set of steps.

```
UPDATE oehr_orders SET order_date = order_date + 4615
```

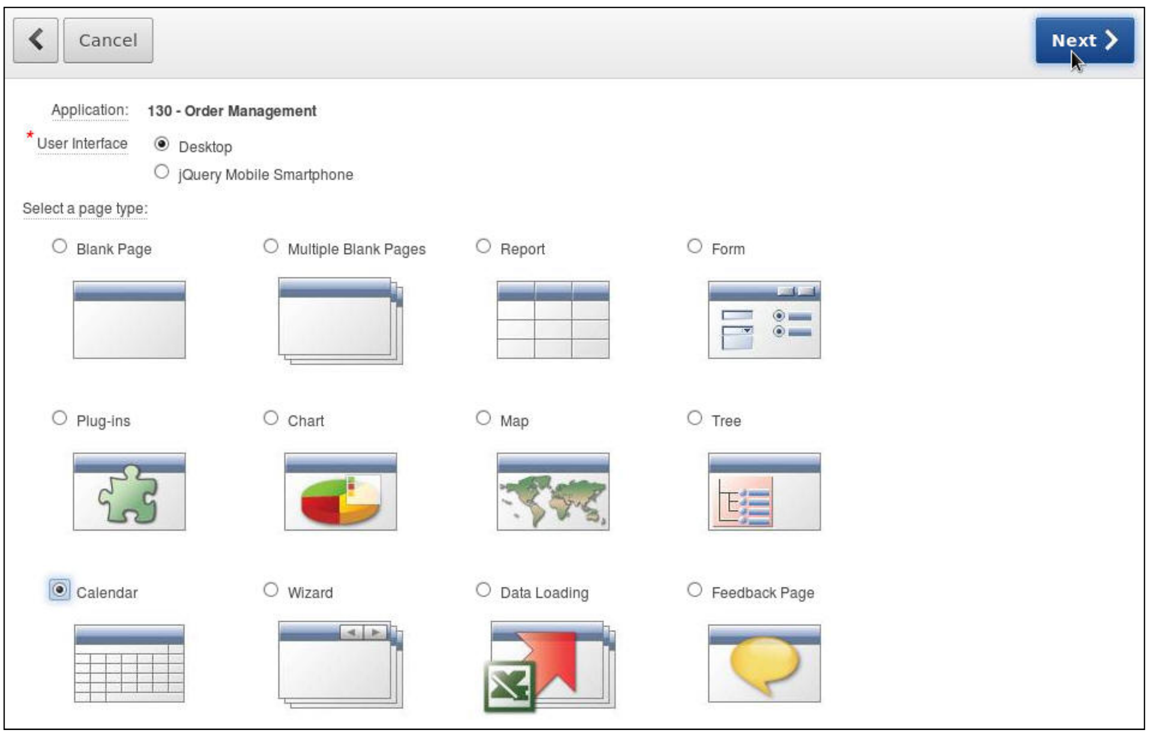


b. Create a new Easy Calendar calendar page called Order Calendar. Base the calendar on the `OEHR_ORDERS` table and set the Date Column to `ORDER_DATE` and Display Column to `ORDER_TOTAL`. Allow for drag and drop order total to different dates. Create a new edit page, so you can edit all the columns in the `OEHR_ORDERS` table. Test your pages.

1) Now you will create your Calendar page. Navigate to the Order Management application Home page and click **Create Page >**.

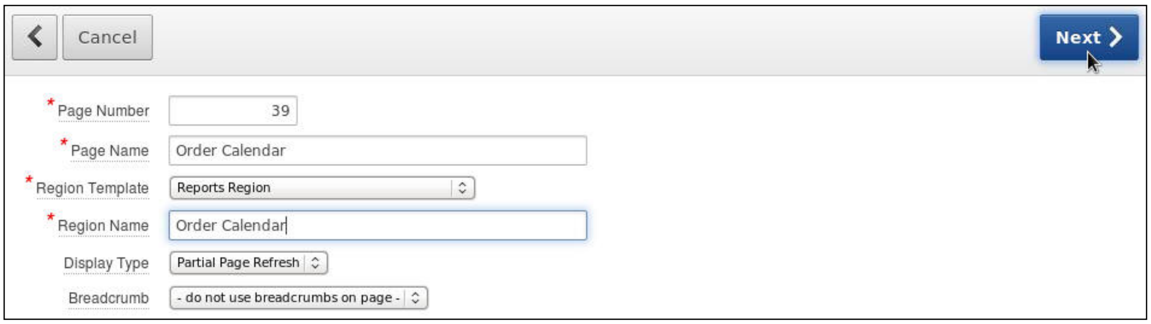


2) Select **Calendar** for page type and click **Next >**.

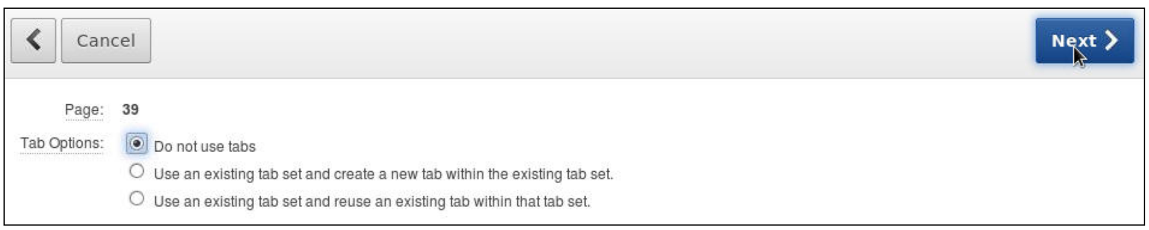


3) Select **Easy Calendar** and click **Next >**.

4) Enter **Order Calendar** for Page Name and Region Name and click **Next >**.

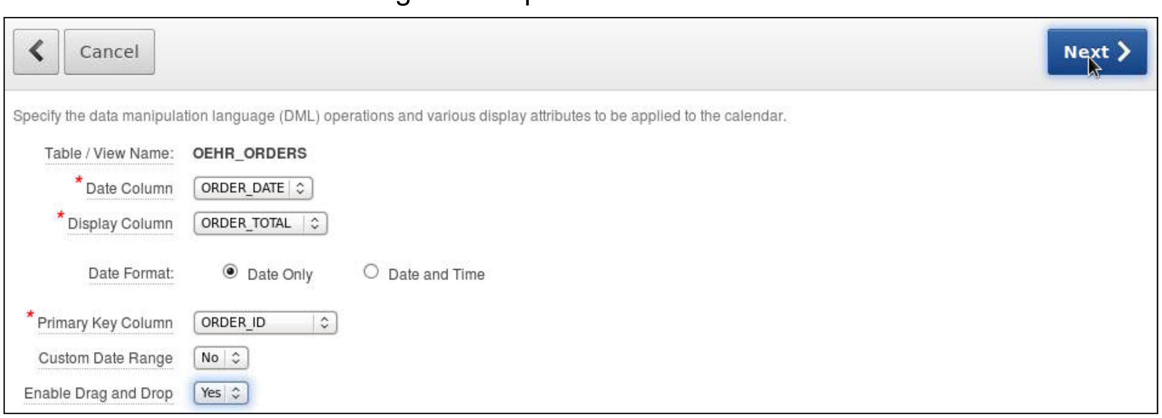


5) Accept the defaults and click **Next >**.

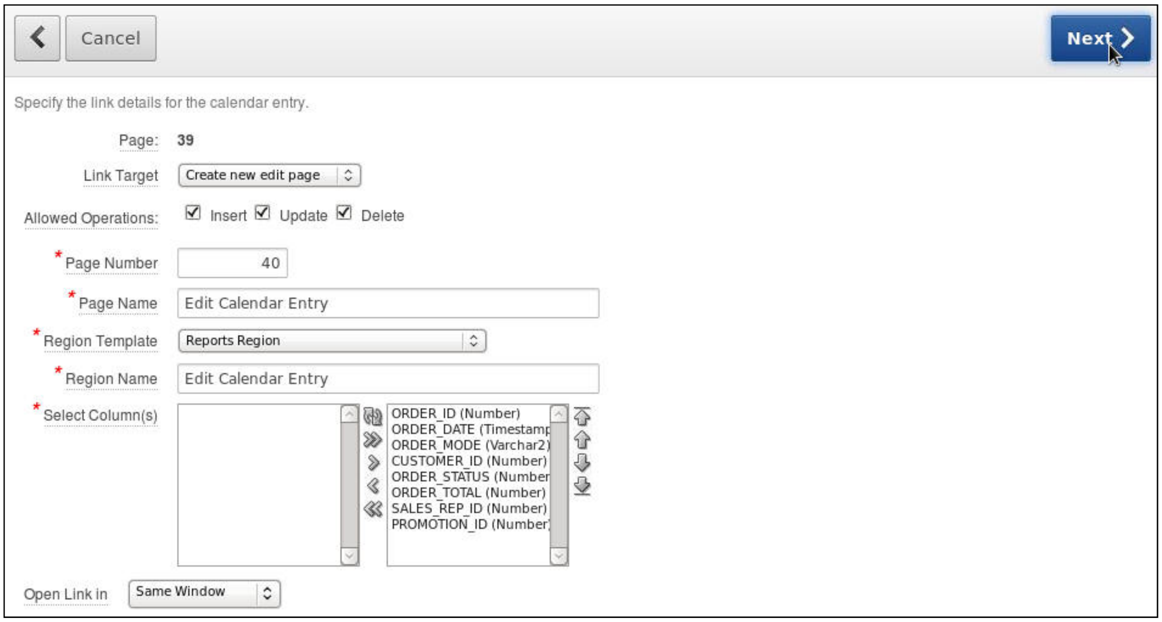


6) Select **OEHR_ORDERS (table)** for Table / View Name and click **Next >**.

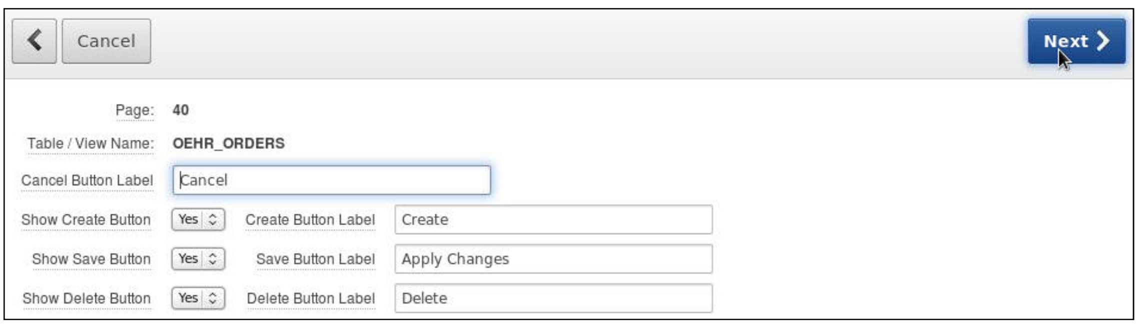
7) Make sure **ORDER_DATE** is selected for Date Column, **ORDER_TOTAL** is selected for Display Column, **ORDER_ID** is selected for Primary Key Column and Yes is selected for Enable Drag and Drop. Then click **Next >**.



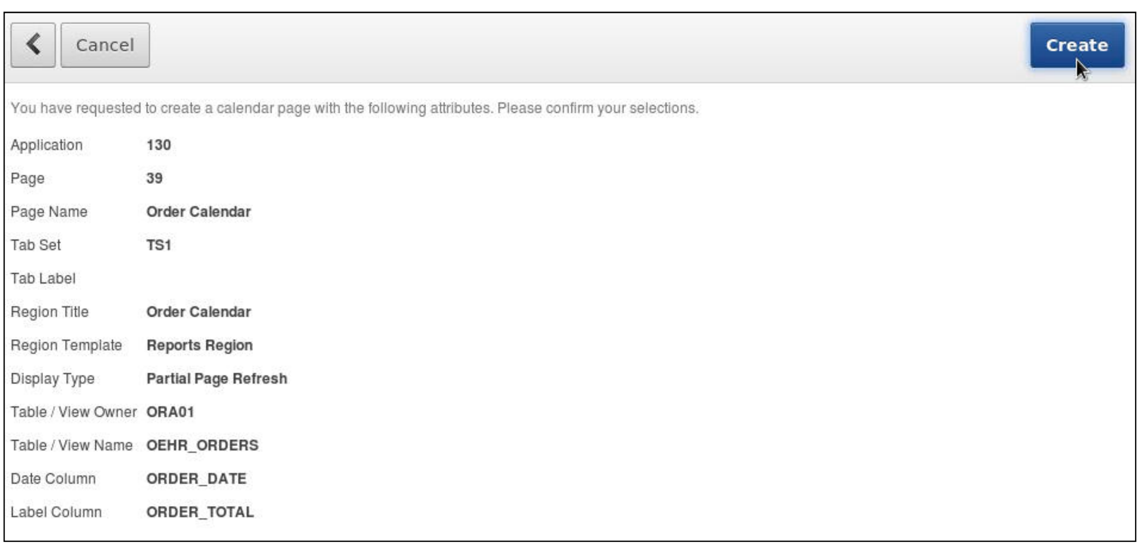
8) Select **Create new edit page** for Link Target and select all the columns for Select Column(s). Click **Next >**.



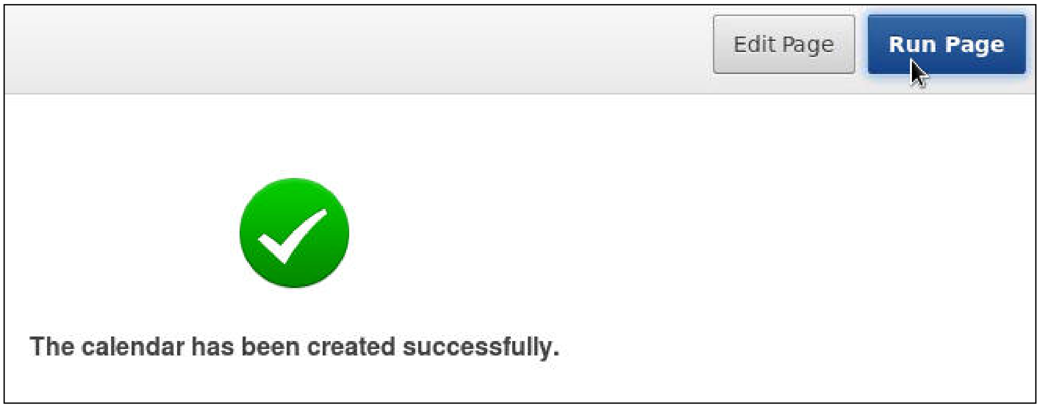
9) Accept the defaults and click **Next >**.



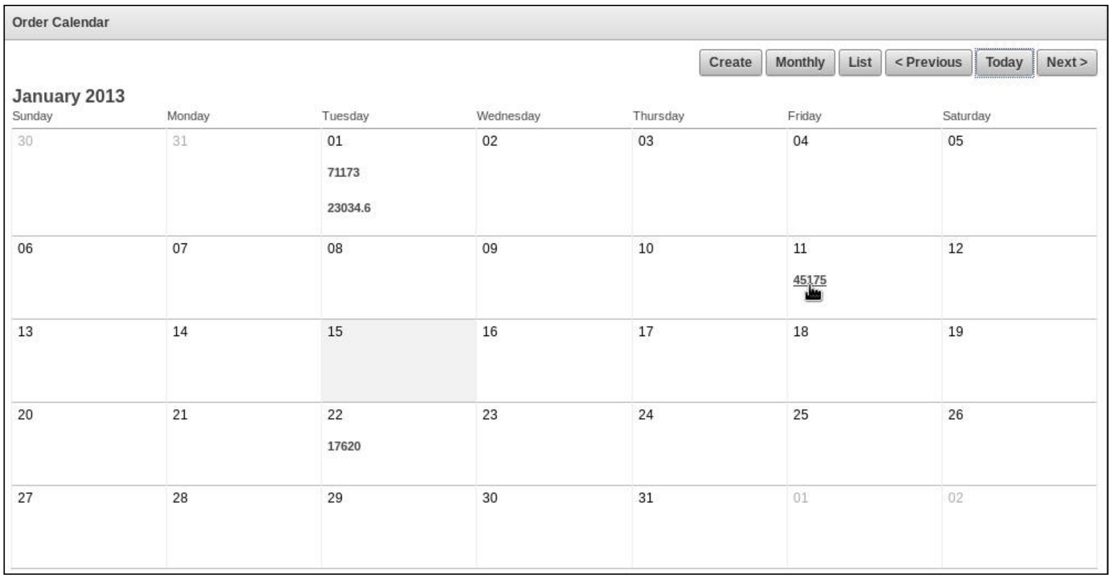
10) Click **Create**.



11) To view the calendar, click **Run Page**.

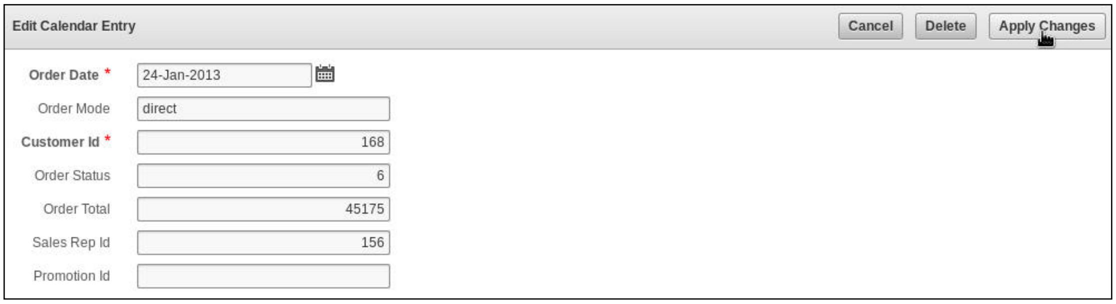


12) Find a month where there is an order total listed for a particular day. Click the link.

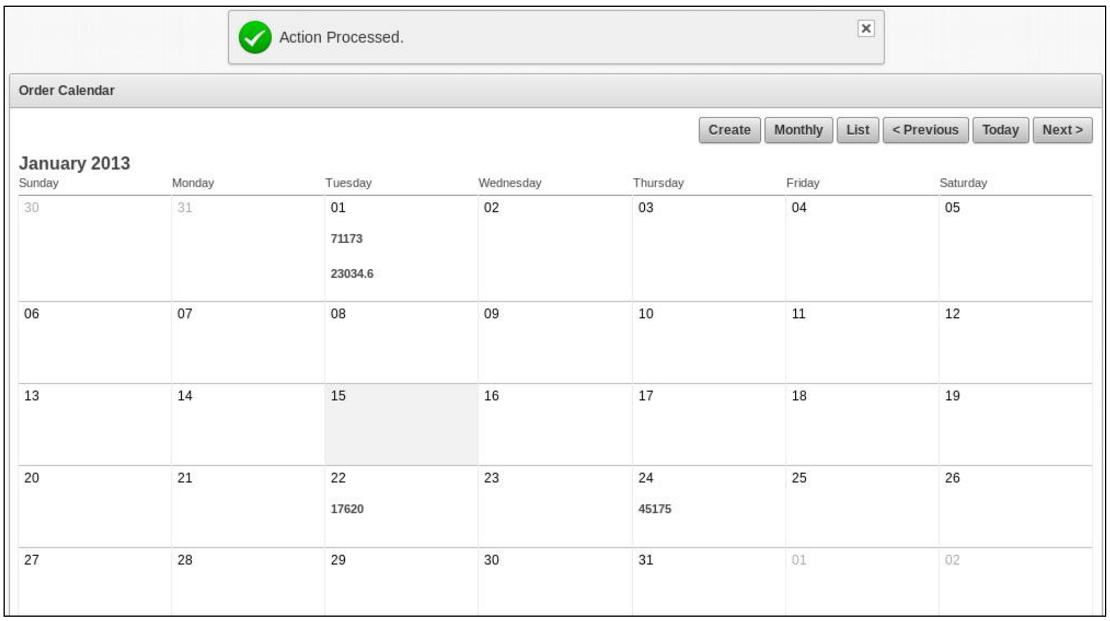


13) Change the Order Date to a different date in the same month. Click the Date Picker icon.

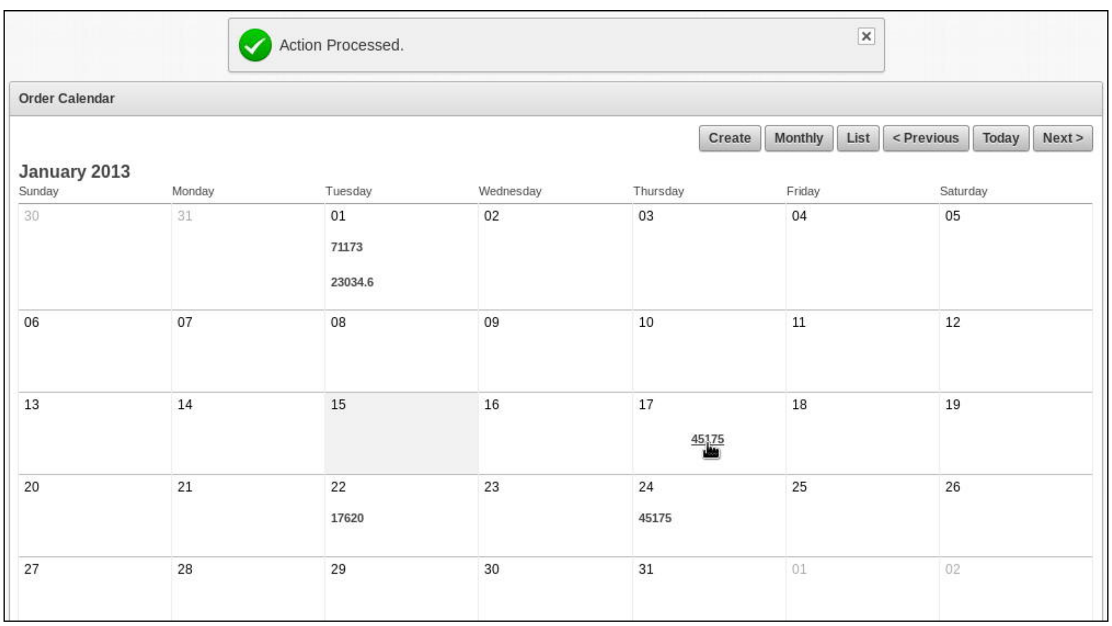
14) Click **Apply Changes**.



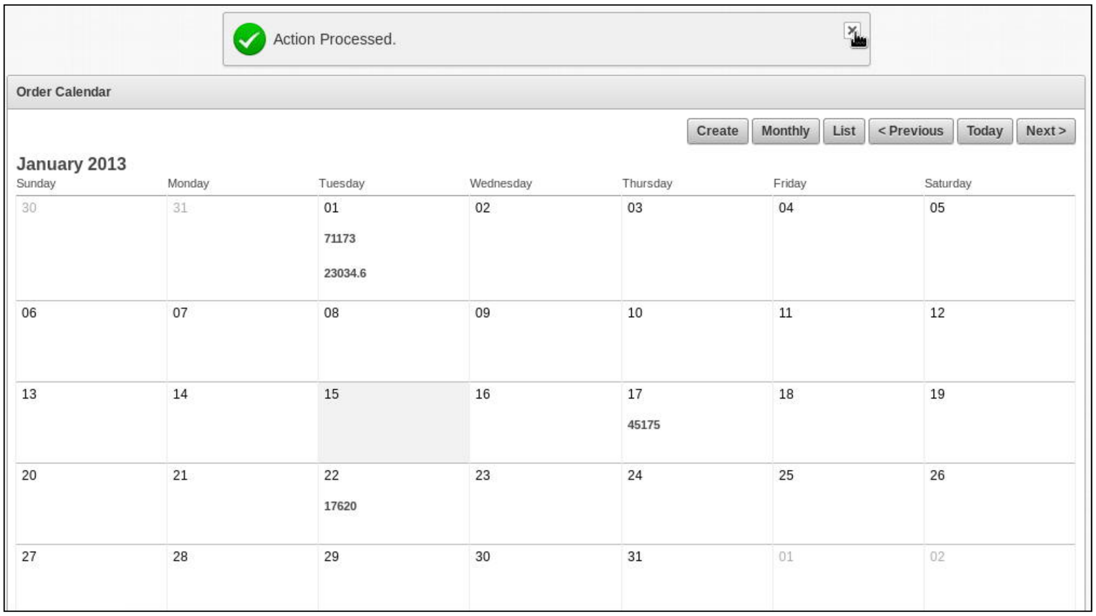
15) The change was applied. Note that the Order Total was moved to the date you changed it to.



16) You can also drag and drop an order total to a different day. Drag the same order total to a different day.



17) Note that after you release your mouse, the order total was moved to the new location. Click the **Application <*n*>** link on the Developer toolbar.

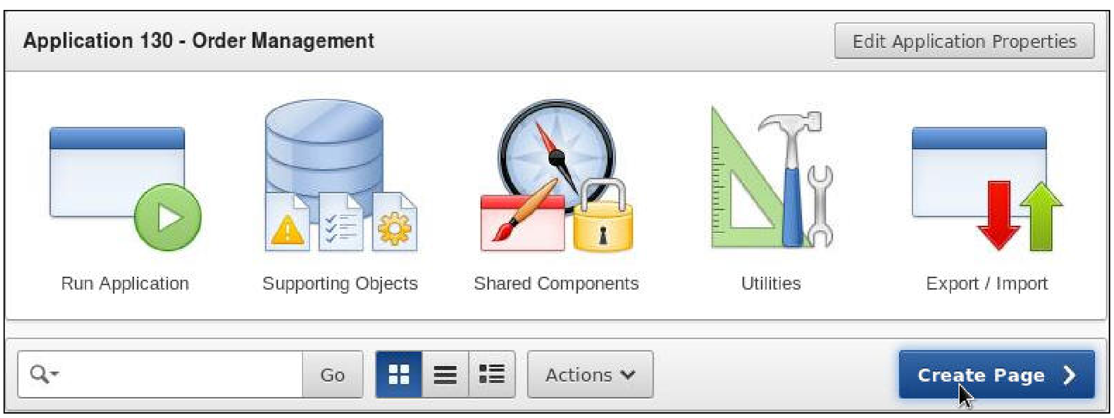


18) To learn more about what you can do with calendaring, you can install the Sample Calendars packaged application.

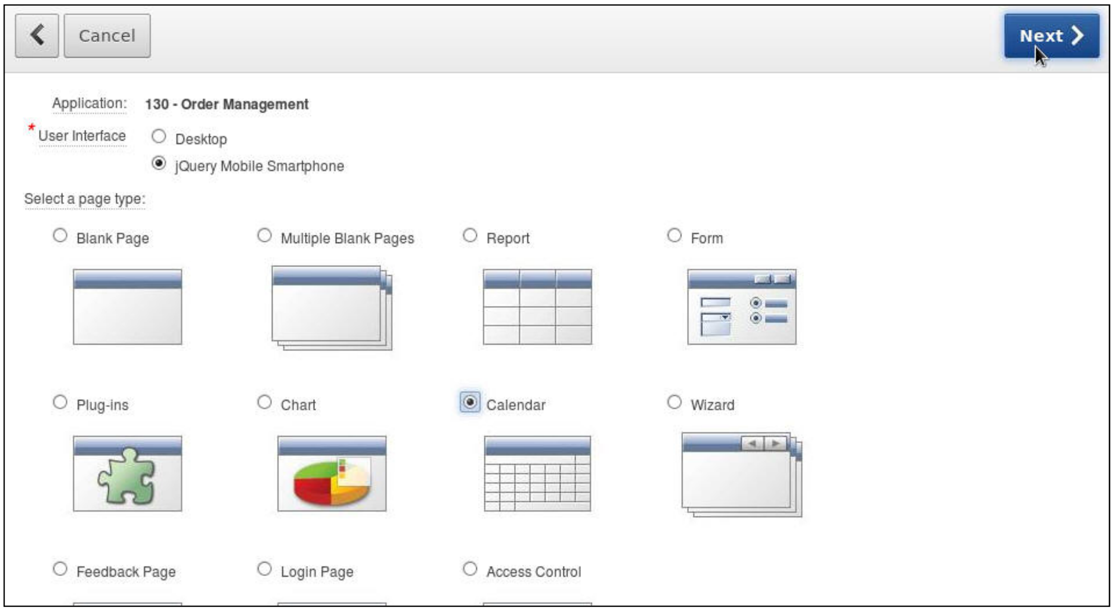
c. Create a new page called Mobile Calendar for mobile application. Select jQuery Mobile Smartphone User Interface to create this page. Select SQL Calendar. Enter a `SELECT` SQL statement (located in the `/home/oracle/labs/files/lab18_1_2.txt` file) to display the name column and start date column from the **`TASKS`** table. Assign the Link Target to create a new edit page, which contains all the columns from the **`TASKS`** table. Test your page.

```
select id, project, task_name, start_date, cost from tasks
```

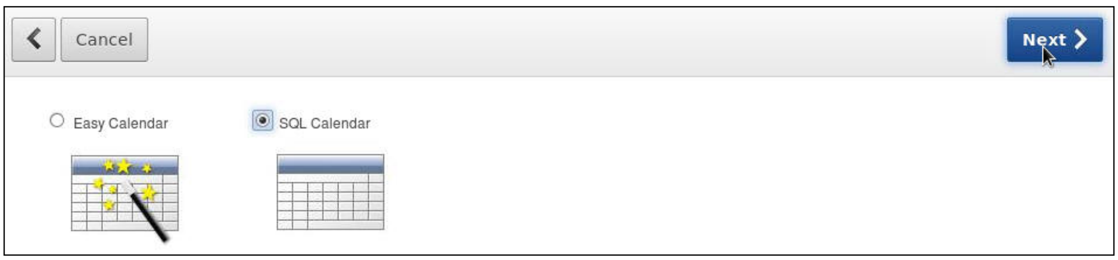
1) Navigate to the **Order Management** application home page, click **Create Page >**.



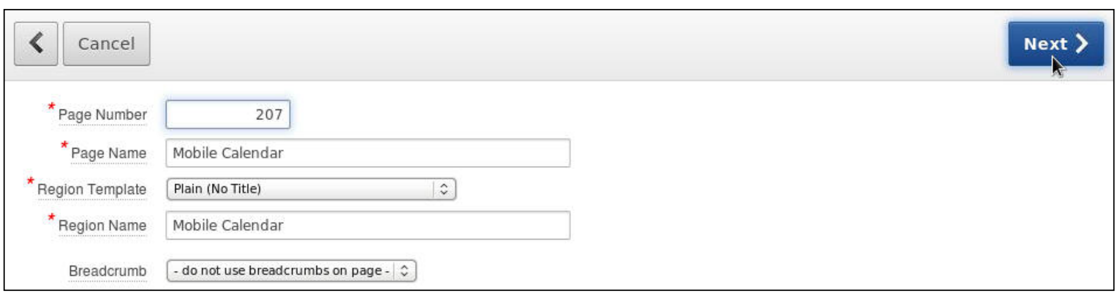
2) Select **jQuery Mobile Smartphone** for User Interface. Select **Calendar** for page type and click **Next >**.



3) Select **SQL Calendar** and click **Next >**.

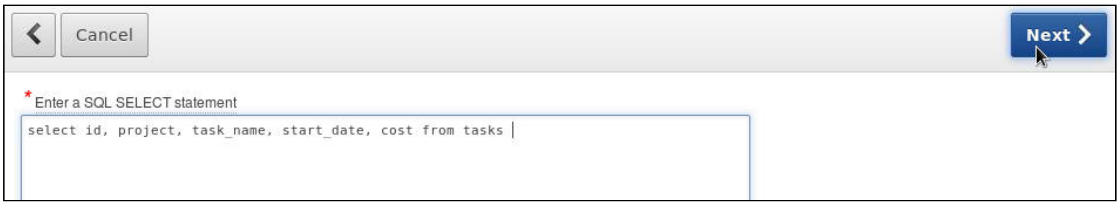


4) Enter a new page number. (Enter a number in the 200 series to distinguish it from desktop pages. Identify a page number that has not already been assigned to another page in the mobile application.) Enter **Mobile Calendar** for Page Name and Region Name. Click **Next >**.

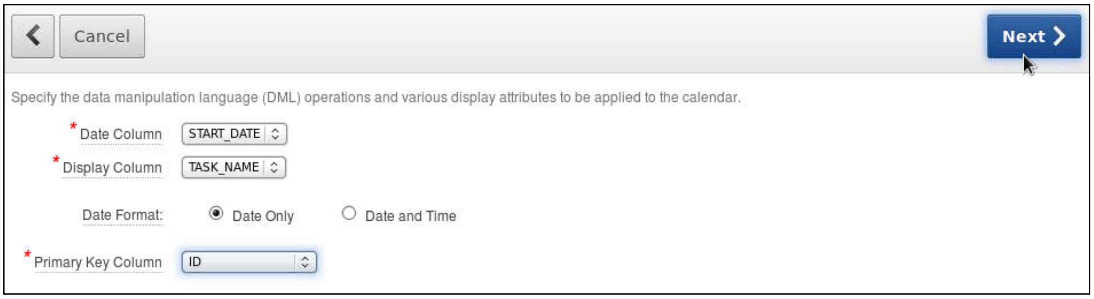


5) Enter the following `SELECT` SQL statement (located in `/home/oracle/labs/files/lab18_1_2.txt`) and click **Next >**.

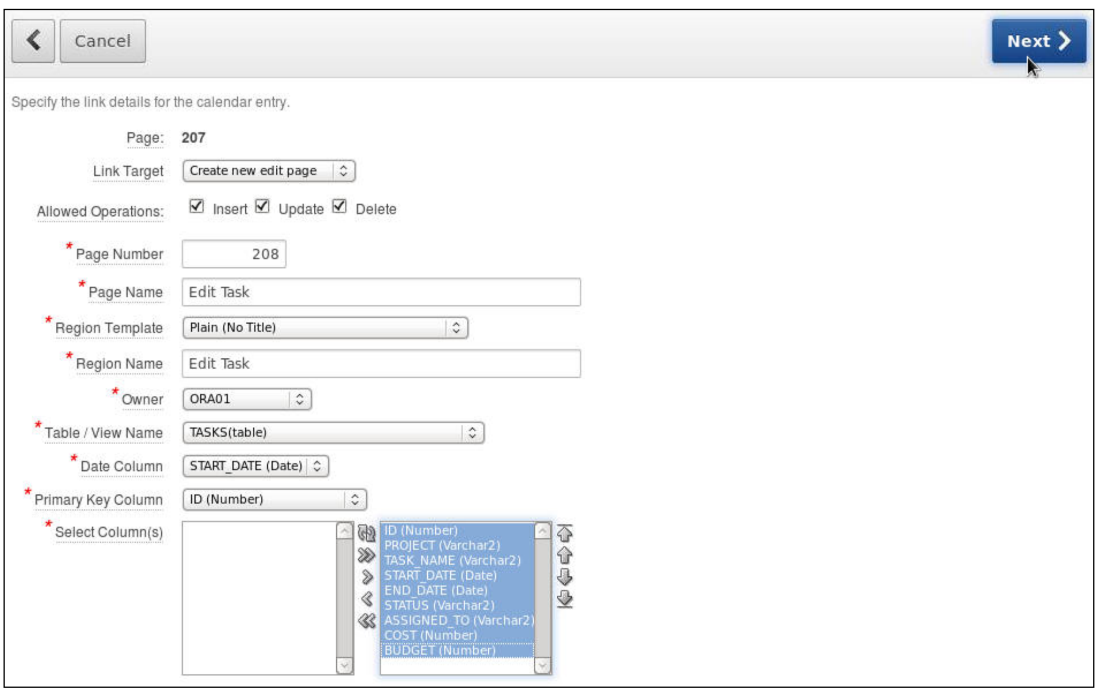
```
select id, project, task_name, start_date, cost from tasks
```



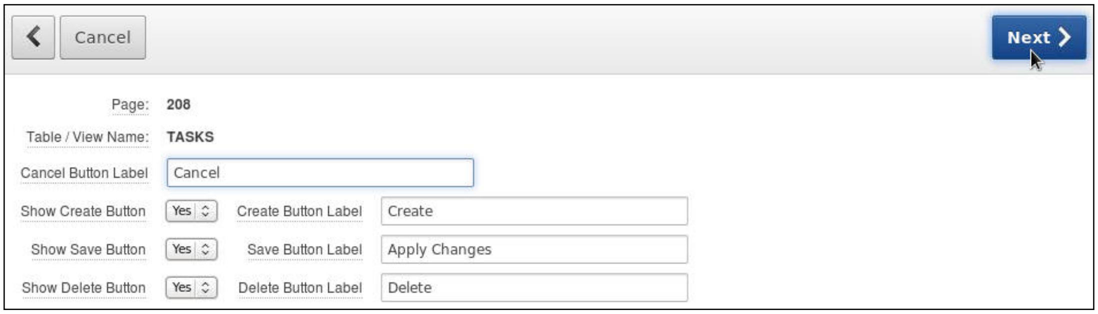
6) Select **START_DATE** for Date Column, **TASK_NAME** for Display Column, and **ID** for the Primary Key Column. Then click **Next >**.



7) Select **Create new edit page** for Link Target, enter **208** for Page Number, and select **Edit Task** for Page Name and Region Name. Select **`TASKS`** for Table/View Name and **`START_DATE (Date)`** for Date Column. Select **all** the columns in the Tasks table and click **Next >**.

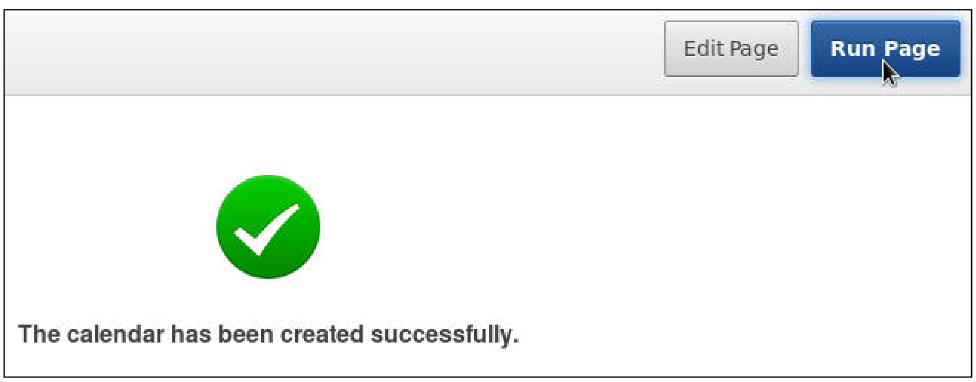


8) Accept the defaults and click **Next >**.

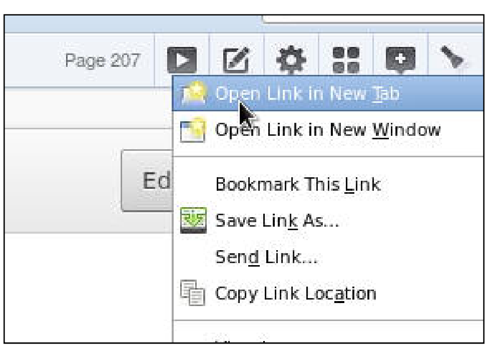


9) Click **Create**.

10) The calendar was created successfully.



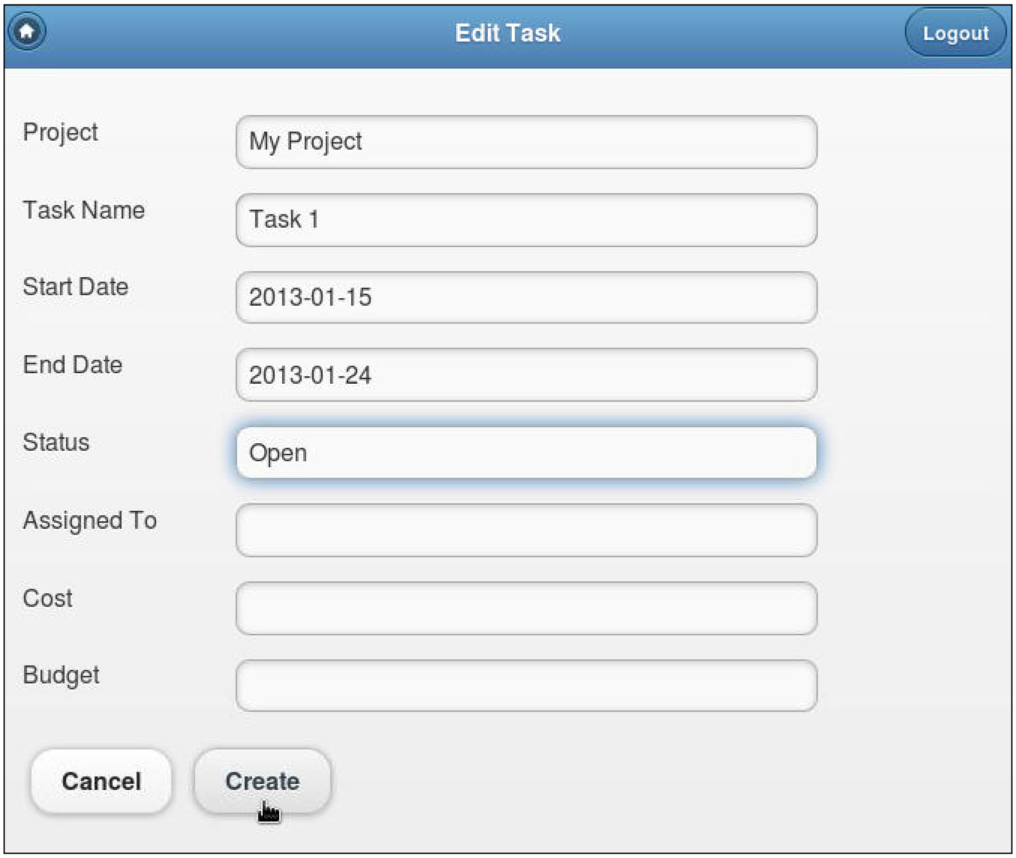
11) On mobile pages, the Developer toolbar is not displayed, so you want to run this page on a different tab. Right-click the **Run Page** icon and select **Open Link in New Tab**.



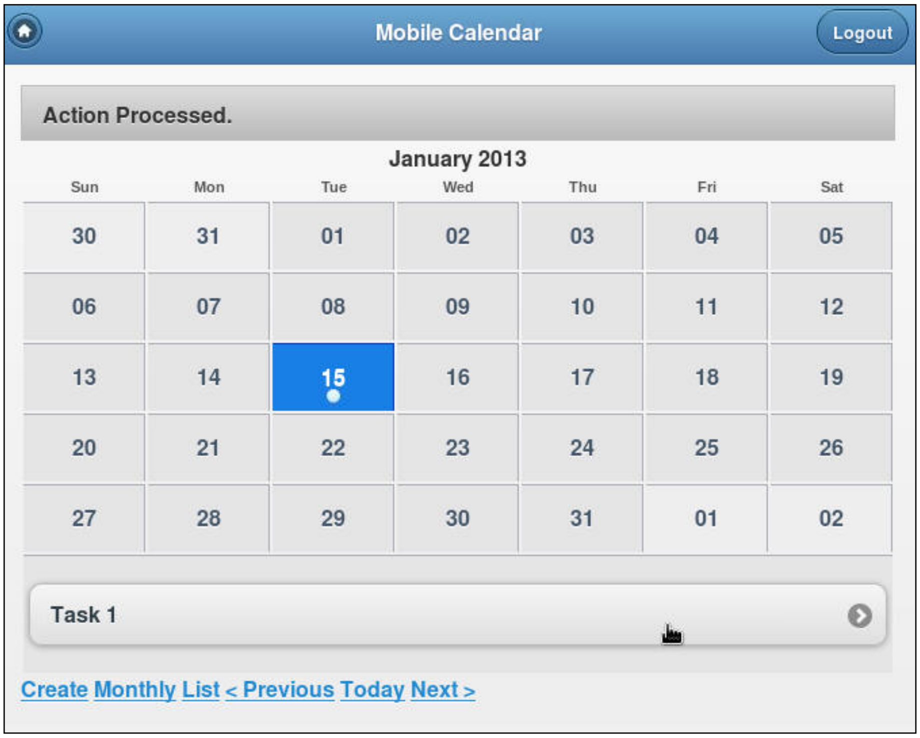
12) Enter your login credentials and click **Login**.

13) The calendar is displayed. You want to create a new task in the current month. Click the **Create** link.

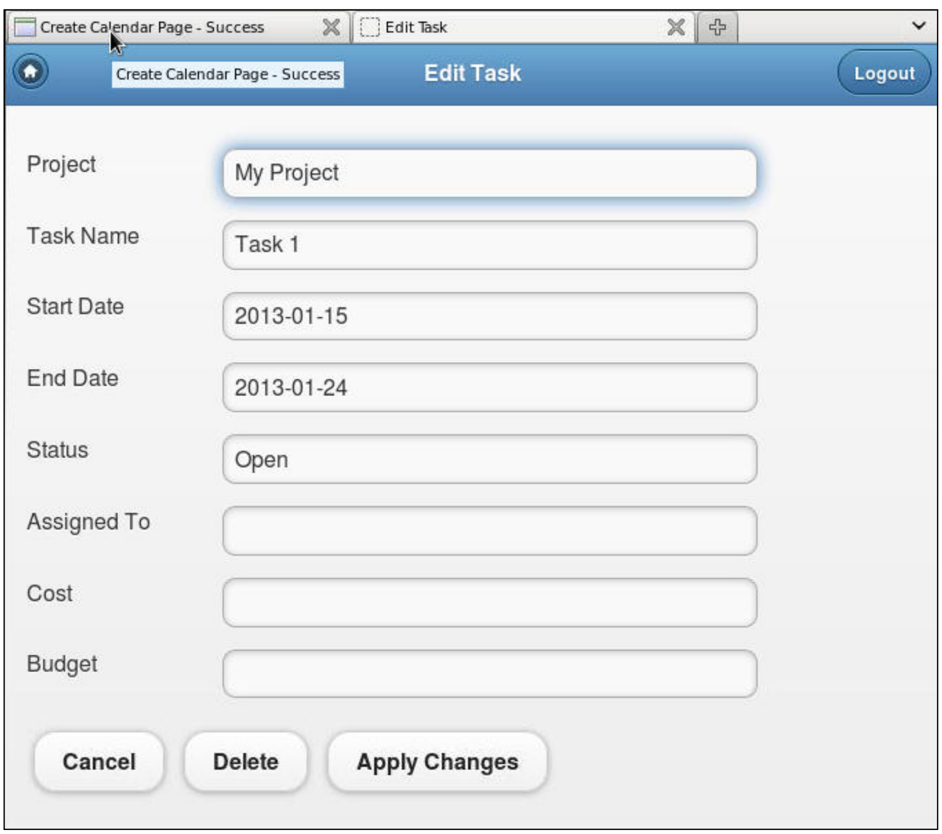
14) The Edit Task form page is displayed. Enter some information and click **Create**. Note that if you were running this on a mobile device you would receive a Date Picker when you clicked in the Start Date and End Date items. In this case, use the format YYYY-MM-DD for those values.



15) Notice that you see a blue dot on the date that you specified for the Start Date. Because that day is selected, you see a list view at the bottom of the report with the name of the Task. You can click that link to edit the task.



16) The form is displayed. To return to the Application Builder, click the tab.



Solution 18-2: Creating a Tree Whose Nodes Link to a Different Page

Overview

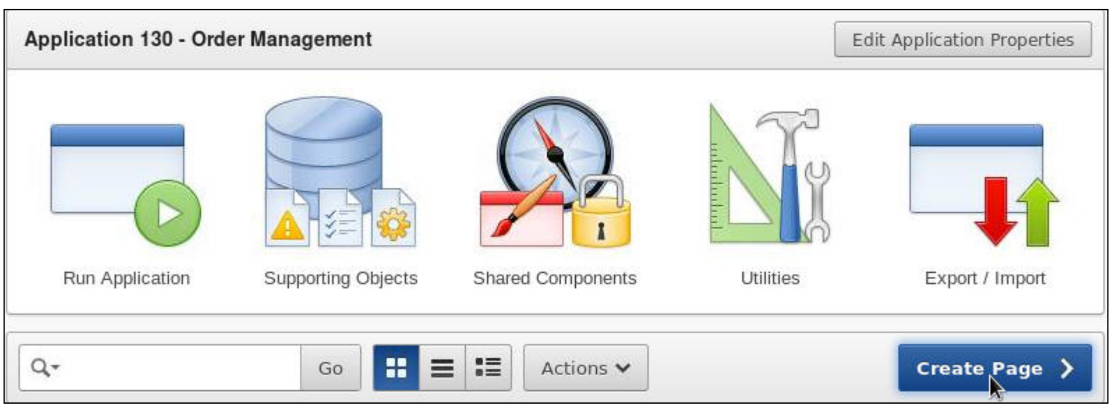
In this practice, you create a new page with a tree region and link to the Dynamic Employee Details page. Add a button on the Dynamic Employee Details back to the tree page.

Tasks

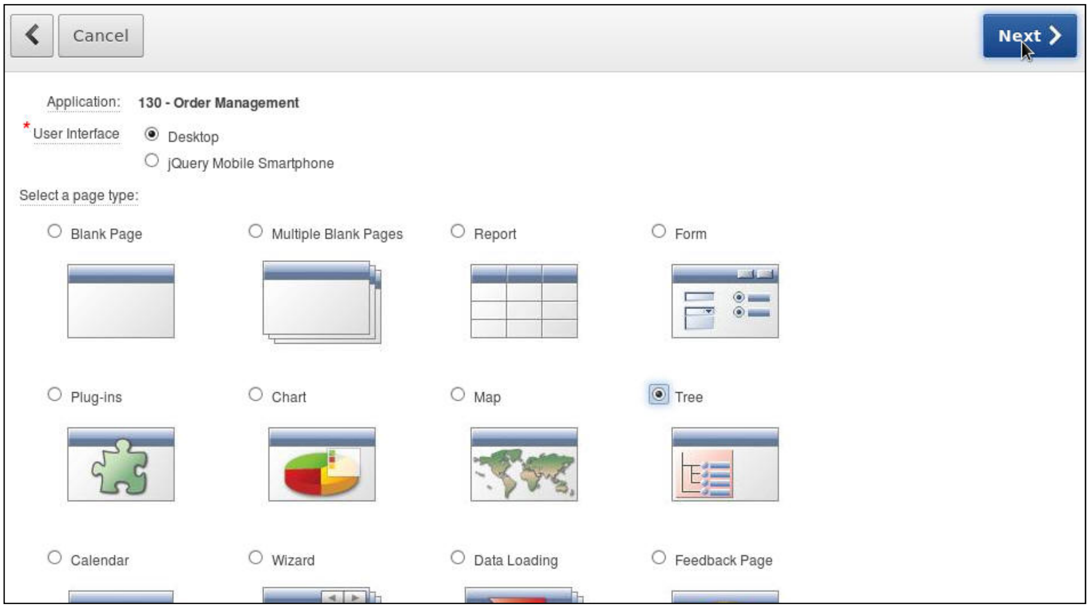
a. Create a tree page called "Employees by Manager" that uses the default tree template. Select the appropriate options in the wizard so that the current query (located in `/home/oracle/labs/files/lab18_2_1.txt`) looks as follows: (<*n*> is the page number for the Dynamic Employee Details page.)

```
select case when connect_by_isleaf = 1 then 0
            when level = 1             then 1
            else                           -1
       end as status,
       level,
       "LAST_NAME" as title,
       null as icon,
       "EMPLOYEE_ID" as value,
       null as tooltip,
       null as link
from "ORA01"."OEHR_EMPLOYEES"
start with "MANAGER_ID" is null
connect by prior "EMPLOYEE_ID" = "MANAGER_ID"
order siblings by "LAST_NAME"
```

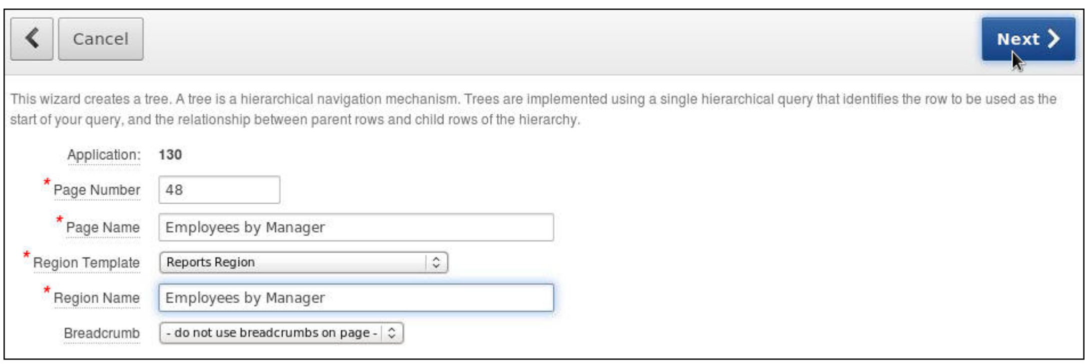
1) On the **Order Management** application home page, click **Create Page >**.



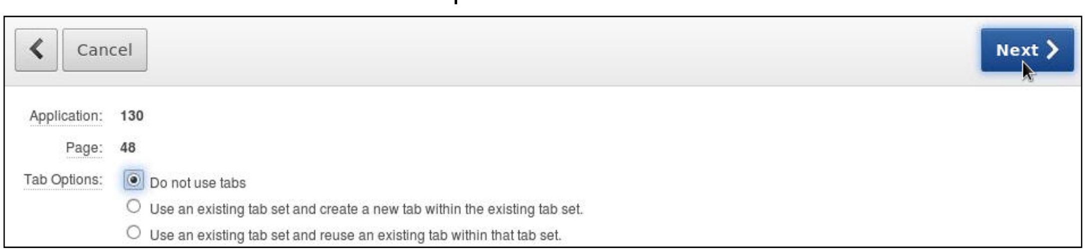
2) Select the **Tree** option, and click **Next >**.



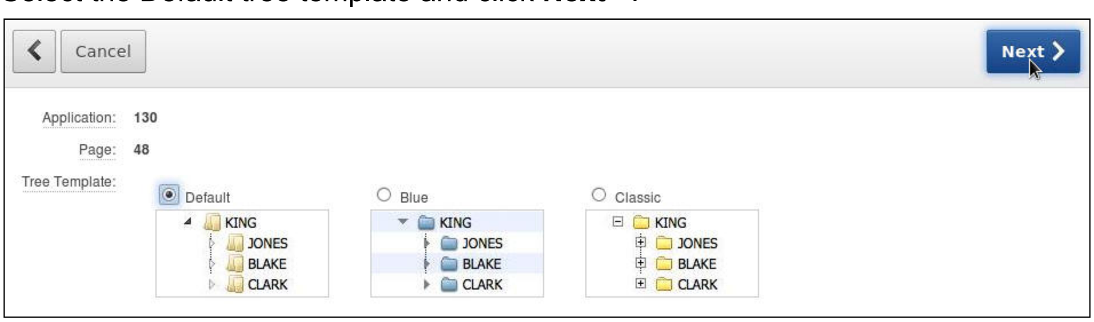
3) Enter **Employees by Manager** for Page Name and Region Name, and click **Next >**.



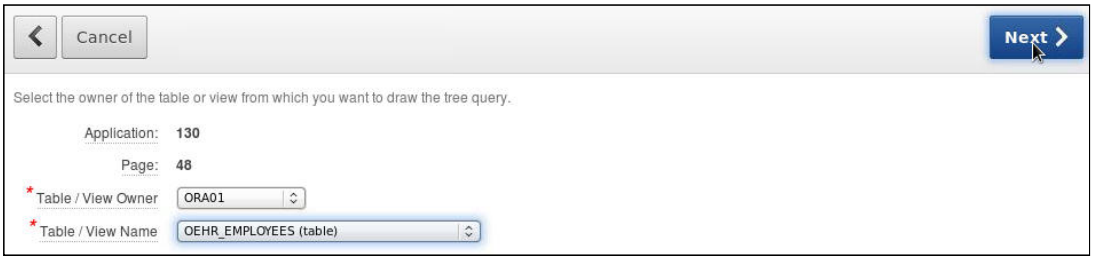
4) Ensure that the **Do not use tabs** option is selected. Click **Next >**.



5) Select the Default tree template and click **Next >**.

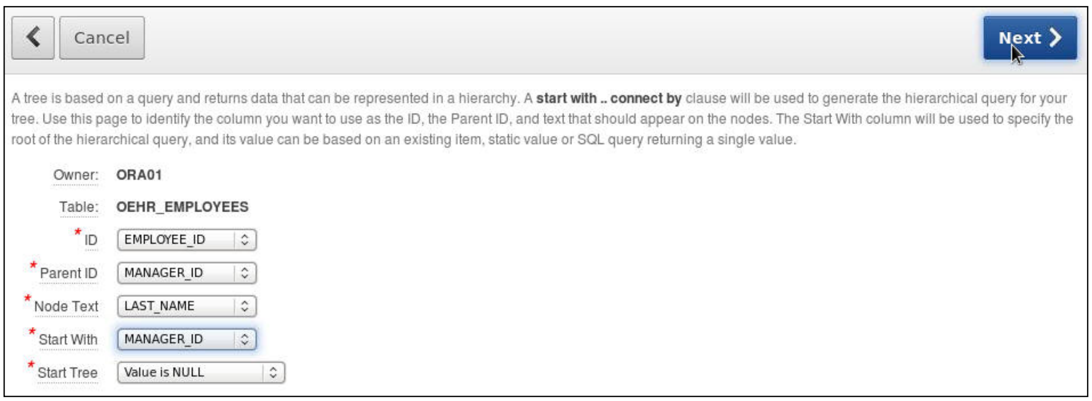


6) Select **`OEHR_EMPLOYEES`** for Table / View Name and click **Next >**.

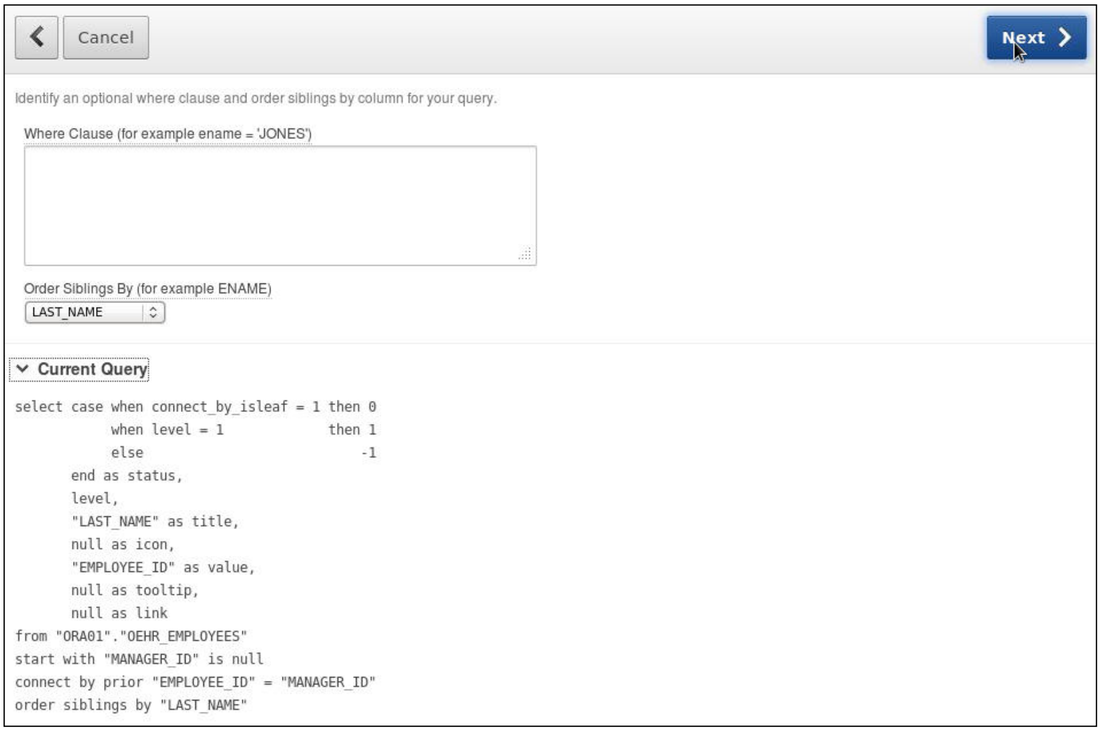


7) Change the following values and click **Next >**.

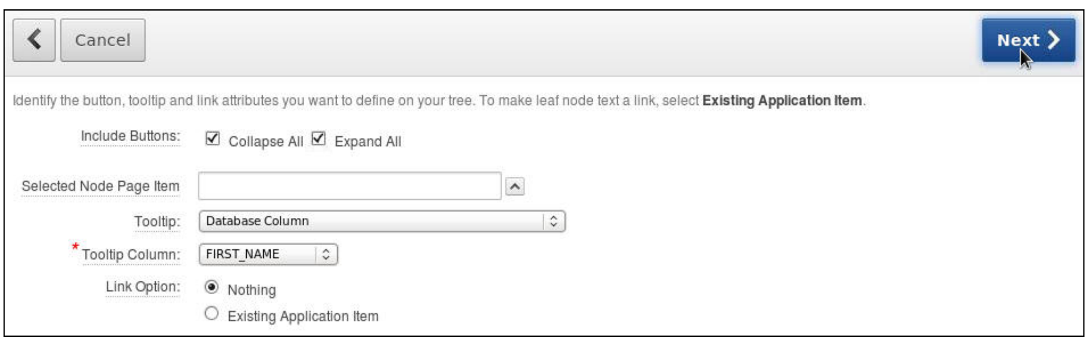
Item	Value
ID	EMPLOYEE_ID
PARENT_ID	MANAGER_ID
Node Text	LAST_NAME
Start With	MANAGER_ID



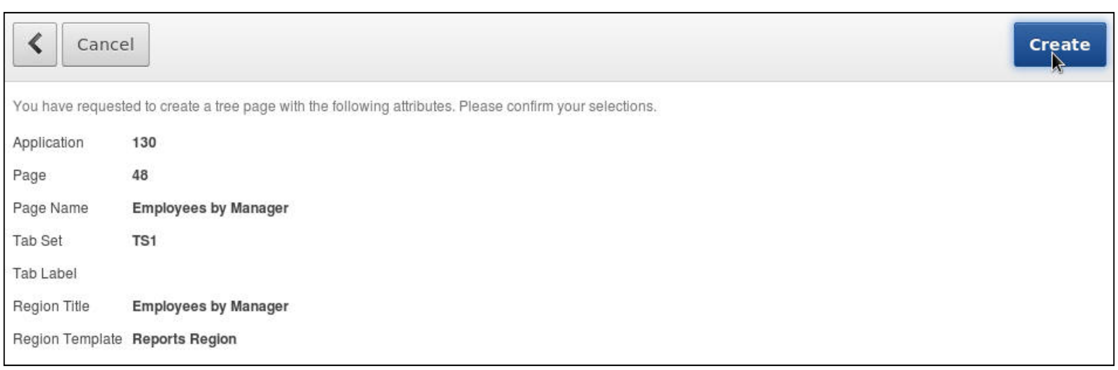
8) You can see the query that is generated by expanding Current Query. Click **Next >**.



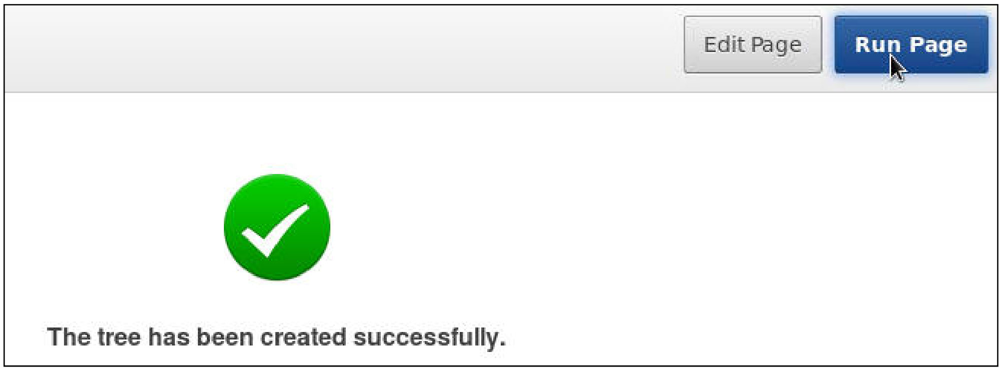
9) Select **Database Column** for Tooltip and `FIRST_NAME` for Tooltip Column. Click **Next >**.



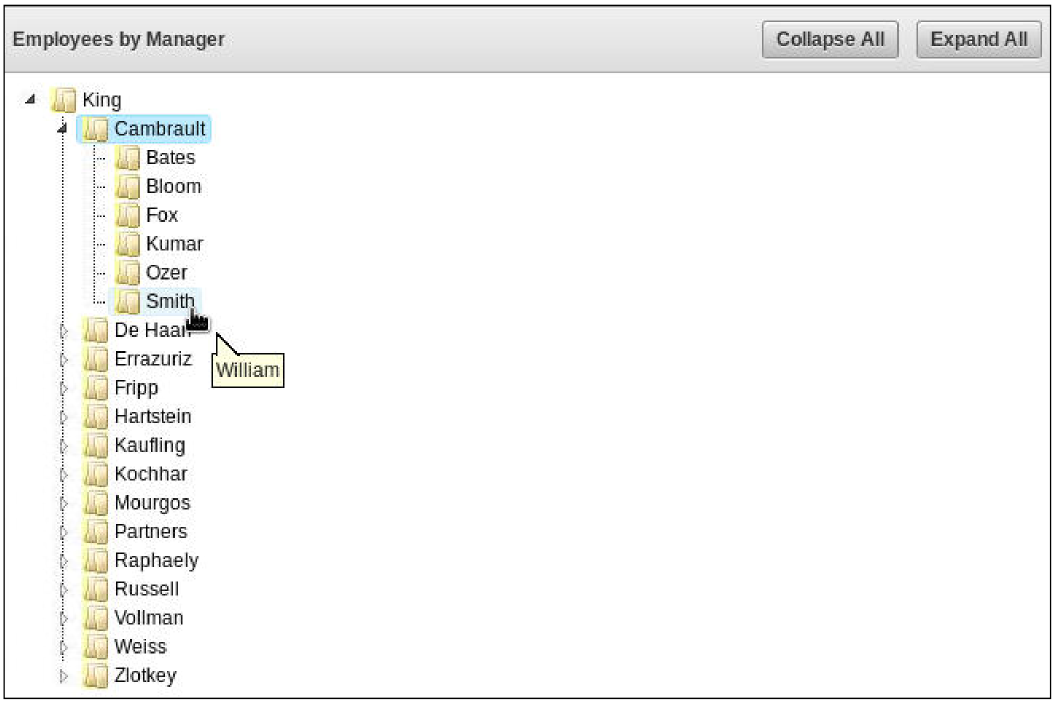
10) Click **Create**.



11) Click **Run Page**.

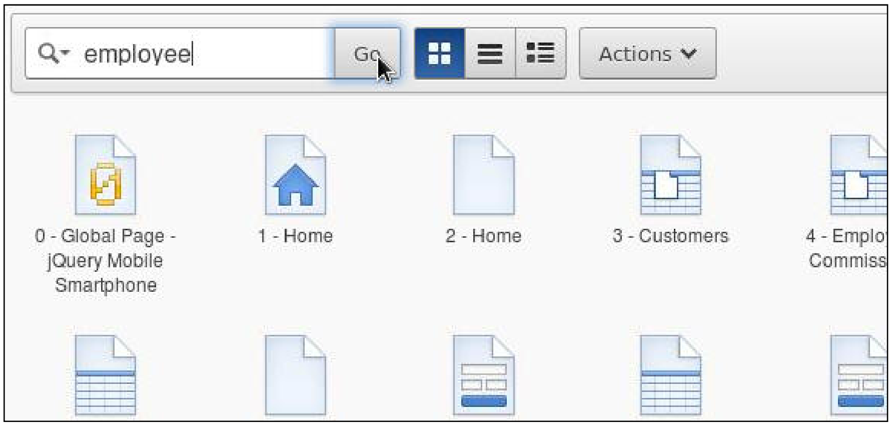


12) Expand one of the manager nodes. Notice that when you place your cursor over one of the names, you see the `FIRST_NAME` of the person as a tooltip. Click the **Application <*n*>** link on the Developer toolbar.

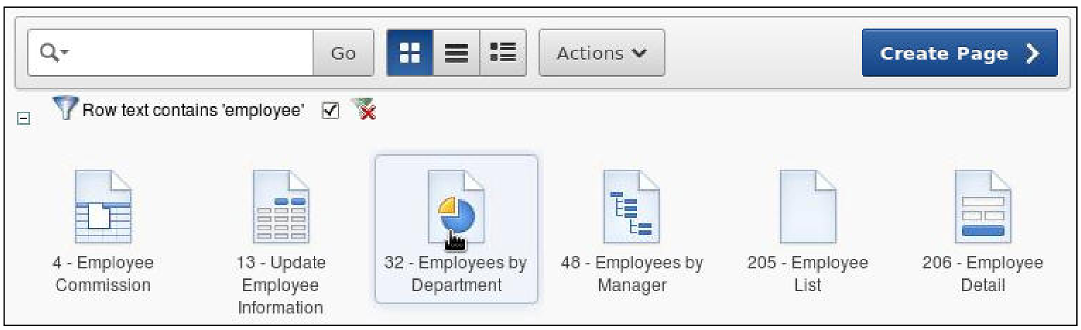


b. Add a button to the Dynamic Employee Details page called Employee Hierarchy that navigates to the “Employees by Manager” tree page.

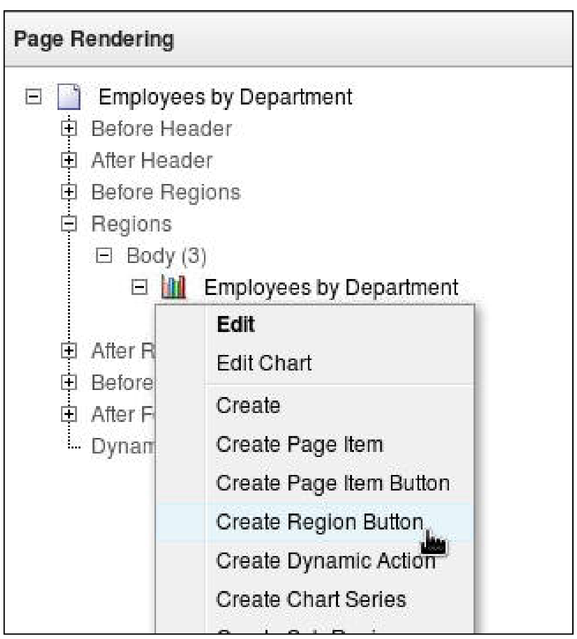
1) To find all the Employee related pages in your application, enter **employee** in the search area and click **Go**.



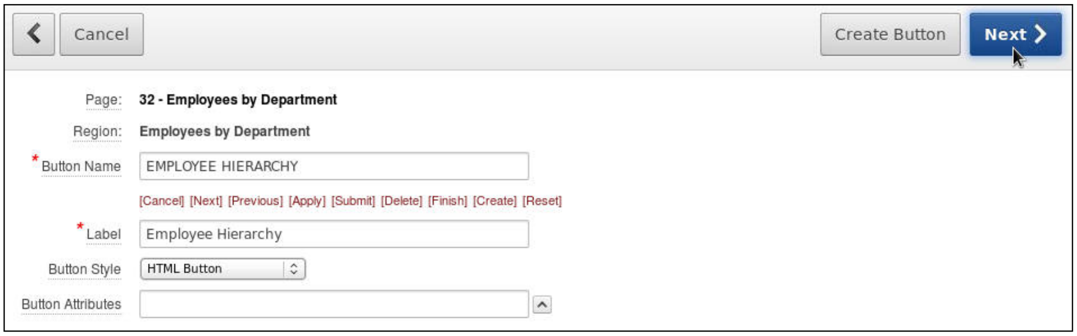
2) Select the Chart page you created earlier, **Employees by Department**.



3) Right-click **Employees by Department** Region and select **Create Region Button**.

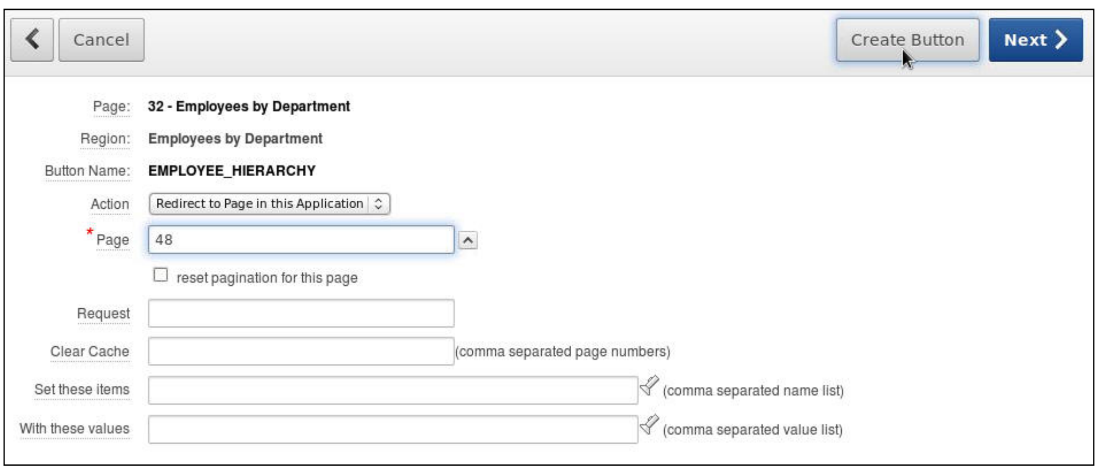


4) Enter **Employee Hierarchy** for Button Name and Label, and click **Next >**.

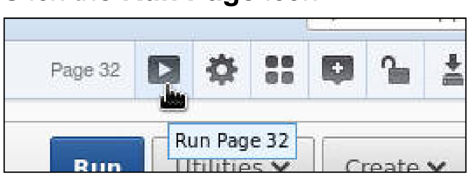


5) Accept the default and click **Next >**.

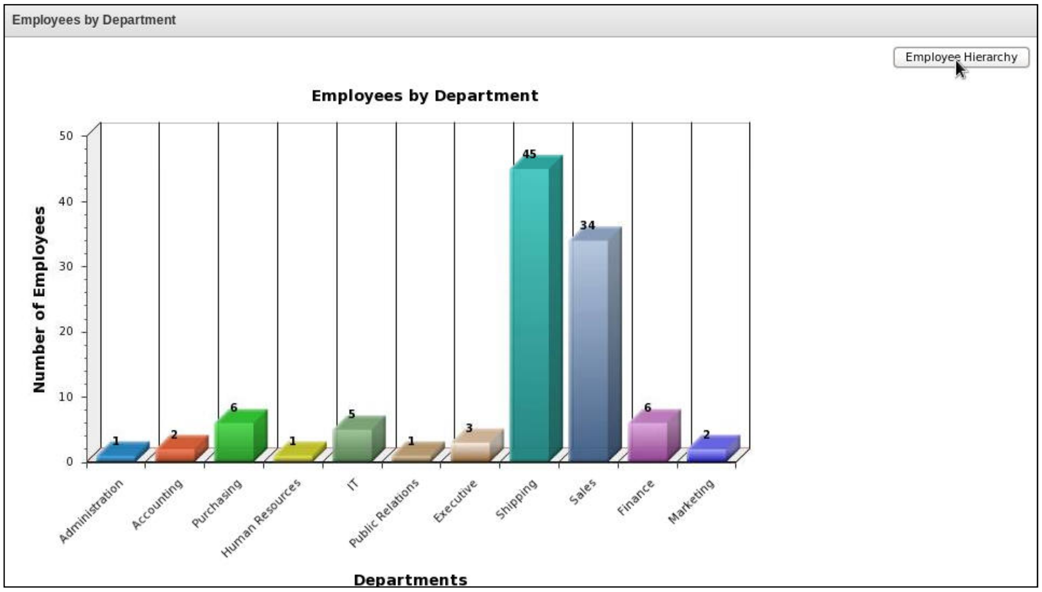
6) Select **Redirect to Page in this Application** for Action, select the **Employees by Manager** page (where your tree is located), and click **Create Button**.



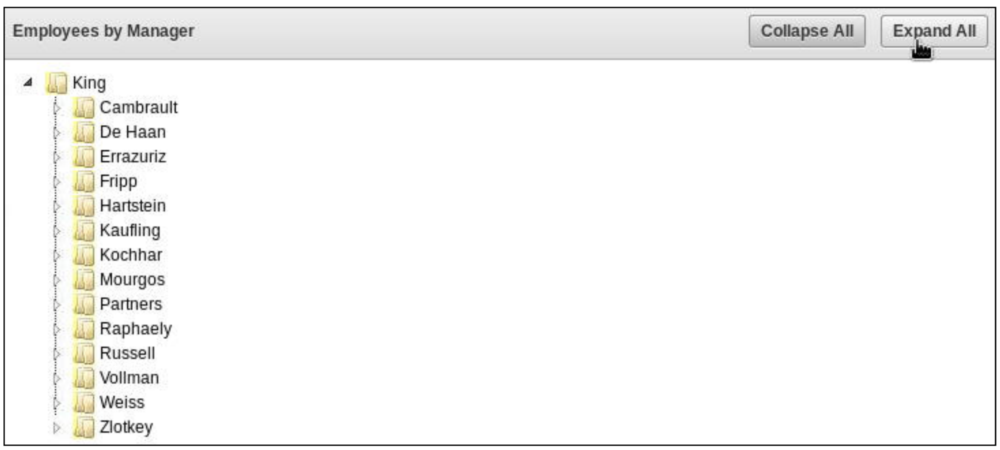
7) Click the **Run Page** icon.



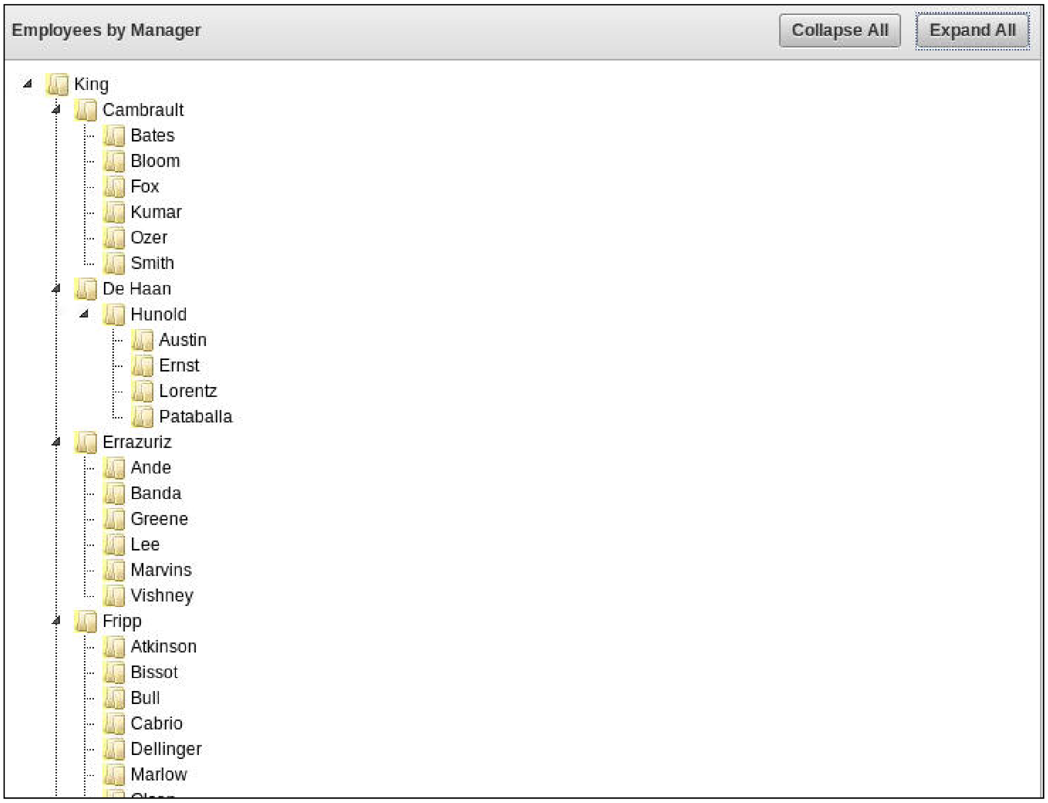
8) Click the **Employee Hierarchy** button.



9) The tree is displayed. Click **Expand All**.



10) The tree is expanded.



Practices for Lesson 19: Using Dynamic Actions and Plug-ins

Chapter 19

Practices for Lesson 19: Overview

Practices Overview

In these practices, you examine how dynamic actions and plug-ins work.

Practice 19-1: Using Dynamic Actions on Your Page

Overview

In this practice, you create the following dynamic actions.

Assumptions

You have performed the previous practices.

Tasks

a. **Show Item when Another item is Not Null:** On the Customer Details page, show P<n>_CUST_EMAIL only if the P<n>_CUST_FIRST_NAME is not null.

b. **Changing the Class When an Item Is Null:** On the Customer Details page, add a class to the P<n>_CUST_EMAIL if the item is null. Add an inline class to your page definition (located in the `/home/oracle/labs/files/lab19_1.txt` file). The dynamic action should fire when the item loses focus.

c. **Refreshing the data in a report using Custom filters:** On the Customers page, create a Quick Filter region with two select list items for P<n>_CUSTOMER_ID and P<n>_CITY_ID. Create a dynamic action on the select lists to refresh the report when the values of the select lists change. Make sure you add the two filter items to the Page Items to Submit.

   - **P<n>_CUSTOMER_ID SQL Query:** Located in the `/home/oracle/labs/files/lab19_2.txt`
   - **P<n>_CITY_ID SQL Query:** Located in the `/home/oracle/labs/files/lab19_3.txt`
   - **Customer Report where clause:** Located in the `/home/oracle/labs/files/lab19_4.txt`

Practice 19-2: Importing and Using Plug-ins

Overview

In this practice, you import and use an item type plug-in and a dynamic action plug-in.

Assumptions

You have performed the previous practices.

Tasks

a. **Importing and Using an Item Type Plug-in:** In this practice, you import the plug-ins, and use them in your application.
   1) Add a rating column to the OEHR_CUSTOMERS table.
   2) Import the Star Rating Item Plug-In and the Notification Dynamic action Plug-in files.
   3) Add the Star Rating Item Plug-in to your Customer Details page.
   4) Add the Notification Dynamic Action Plug-in to your Customer Details page.

Solution 19-1: Using Dynamic Actions on Your Page

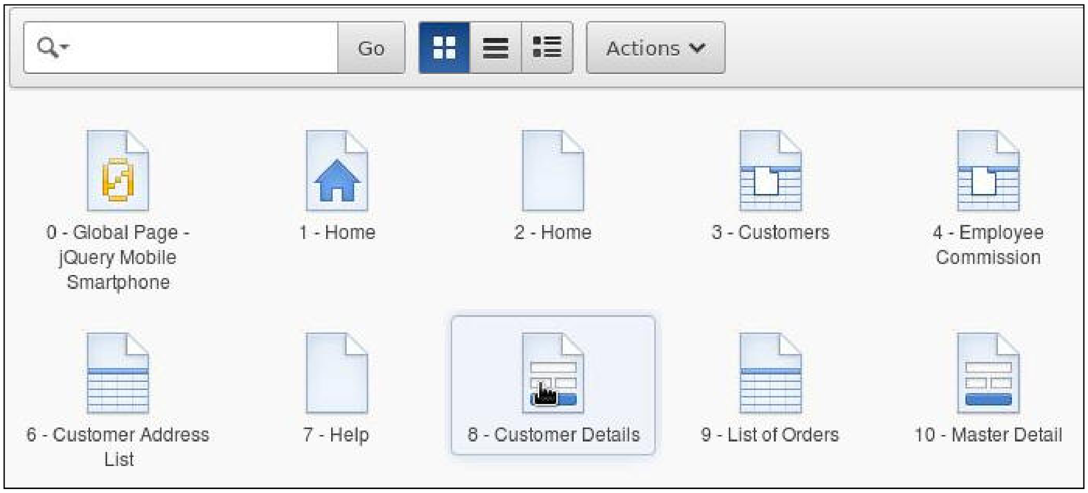
Overview

In this practice, you create various dynamic actions on your page as examples.

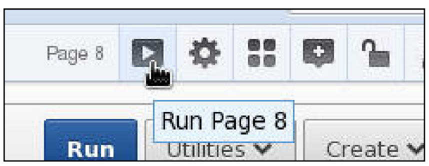
Steps

a. **Show Item when Another item is Not Null:** On the Customer Details page, show P<n>_CUST_EMAIL only if the P<n>_CUST_FIRST_NAME is not null.

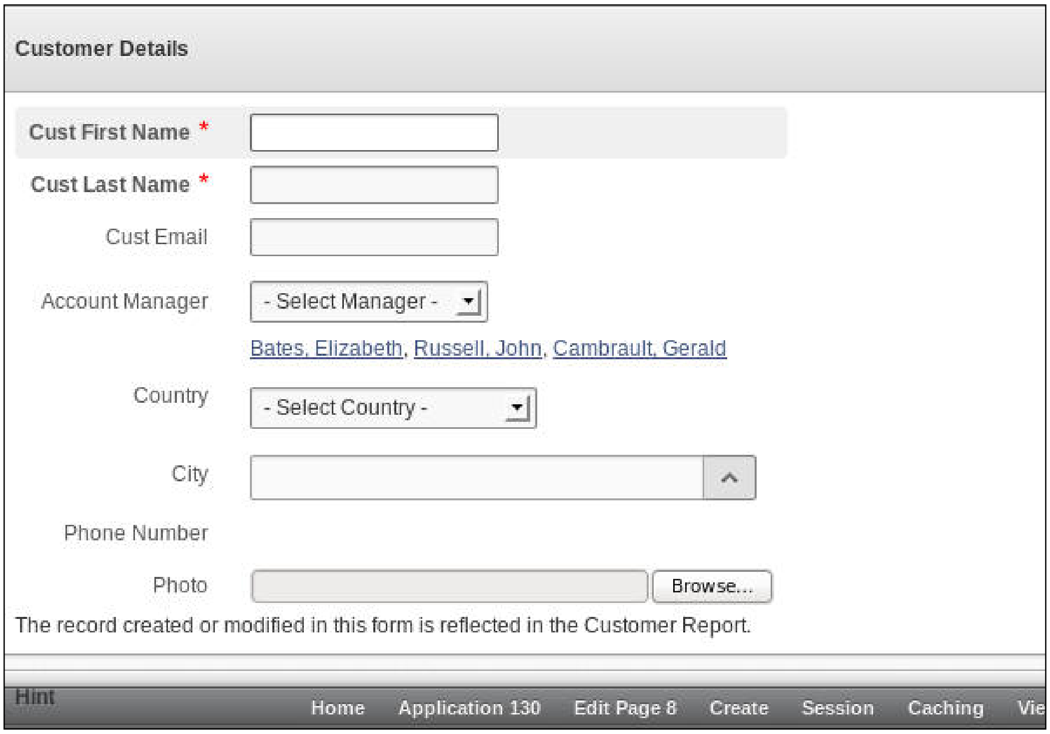
1) Navigate to the Order Management application and select the **Customer Details** page.



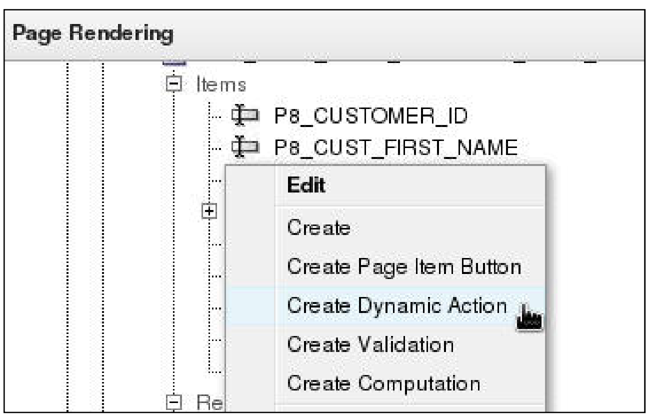
2) Click the **Run Page** icon.



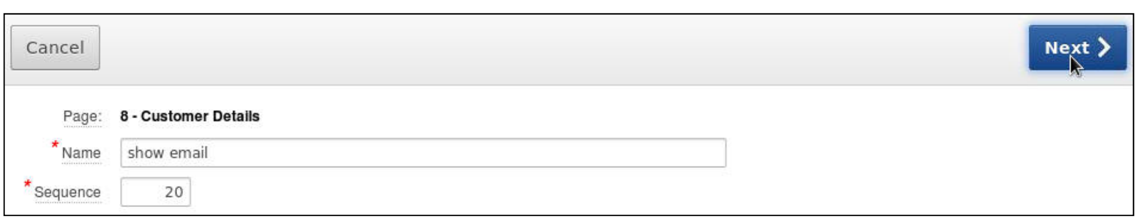
3) You want to add a dynamic action that will only show the Cust Email when a Cust First Name is entered. Click the **Edit Page** icon on the Developer toolbar.



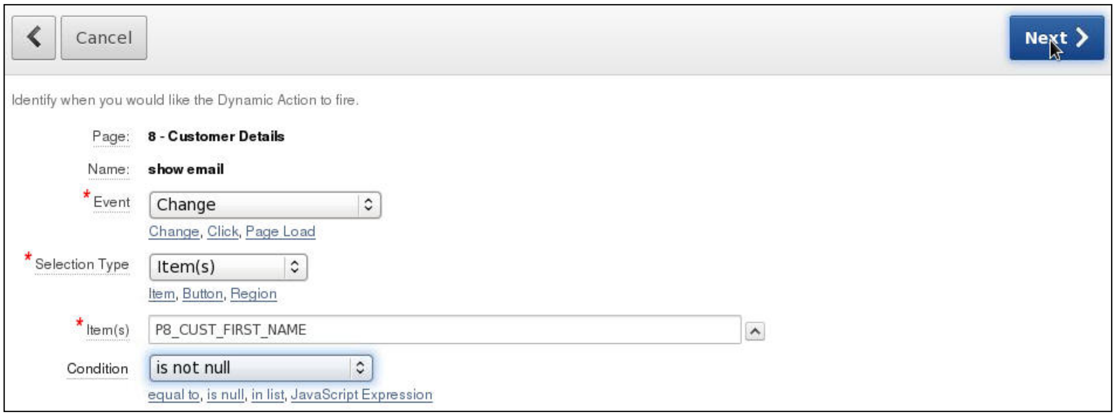
4) Right-click **P<n>_CUST_FIRST_NAME** and select **Create Dynamic Action**.



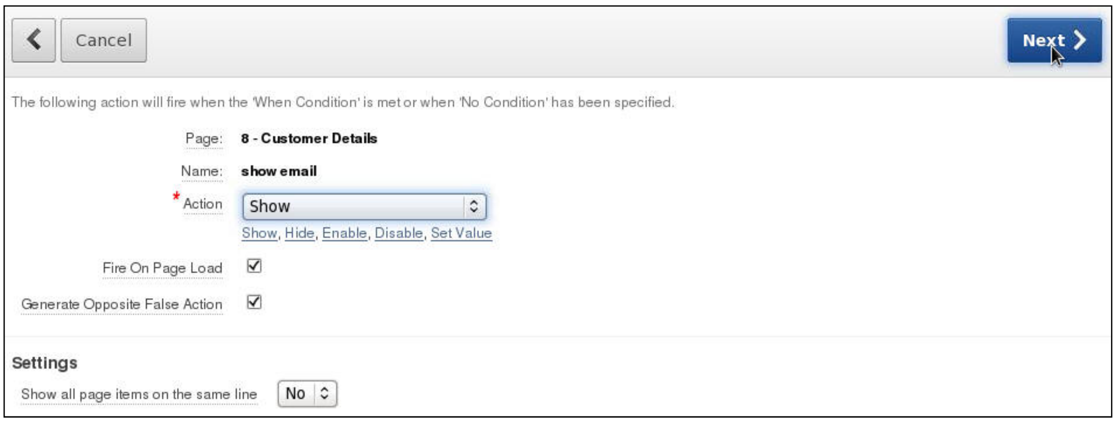
5) Enter **show email** for Name and click **Next >**.



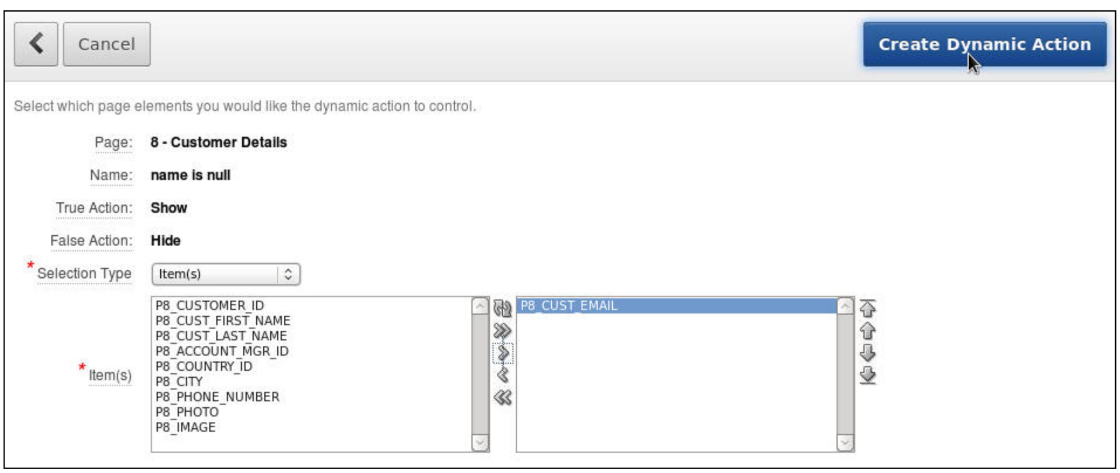
6) Accept the defaults, select **is not null** for Condition and click **Next >**.



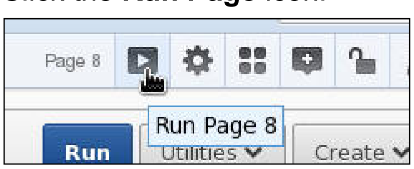
7) Select **Show** for Action and click **Next >**.



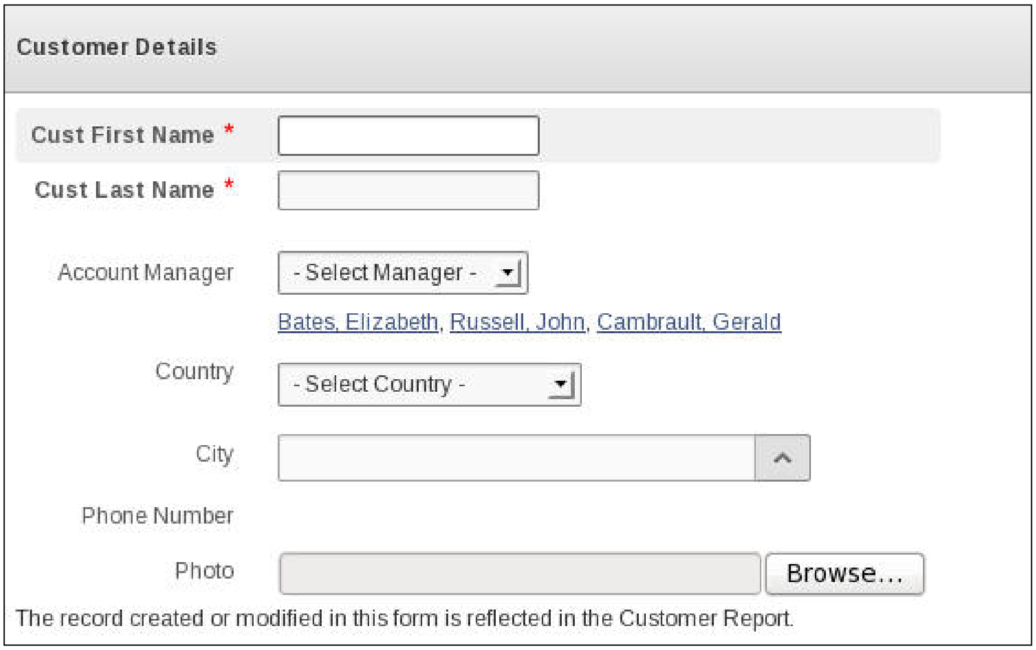
8) Select **Item(s)** for Selection Type, move the **P<n>_CUST_EMAIL** item to the right and click **Create Dynamic Action**.



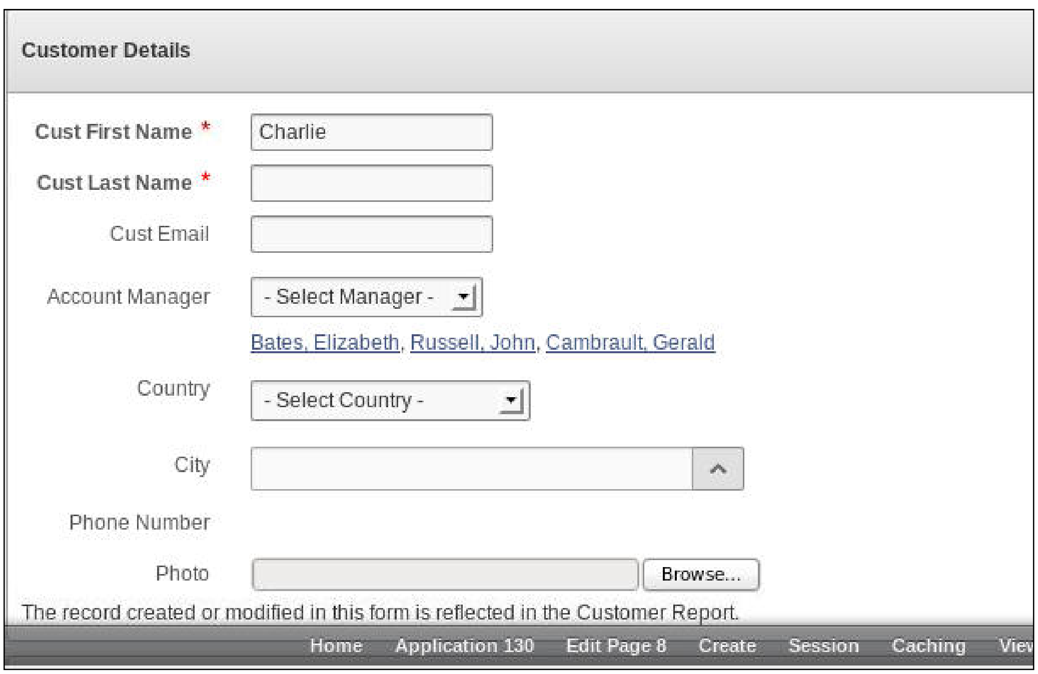
9) Click the **Run Page** icon.



10) Notice that when you first run the page, the Cust Email is hidden. This is due to the dynamic action you just created.

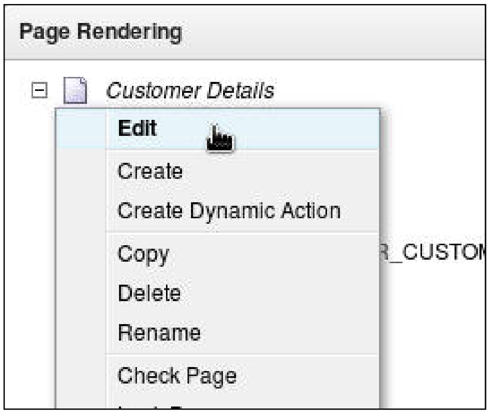


11) Enter something in the Cust First Name item and when you click away from the item, you notice that the Cust Email item is now shown. Click the **Edit Page** link on the Developer toolbar.

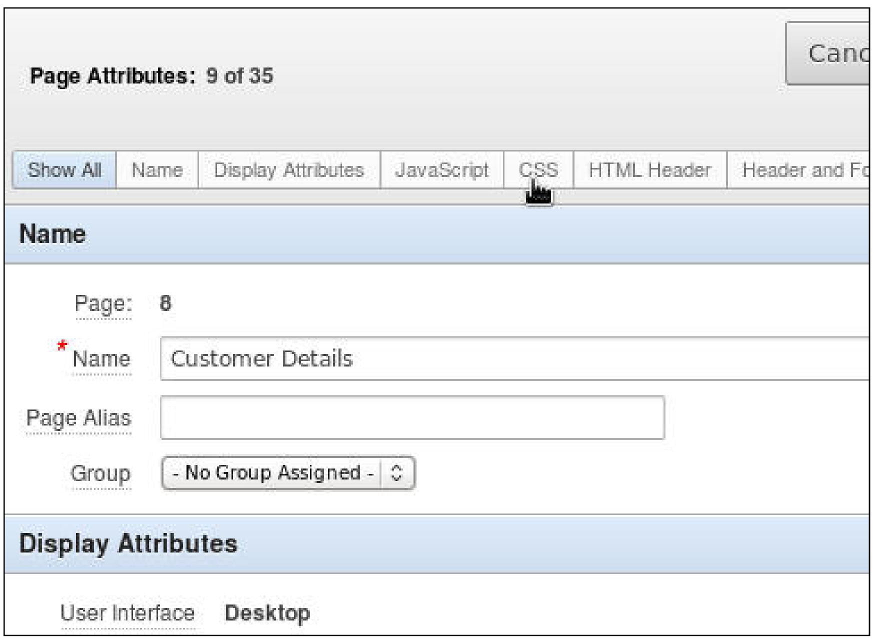


b. **Changing the Class When an Item Is Null:** On the Customer Details page, add a class to the P<n>_CUST_EMAIL if the item is null. Add an inline class to your page definition (located in the `/home/oracle/labs/files/lab19_1.txt` file). The dynamic action should fire when the item loses focus.

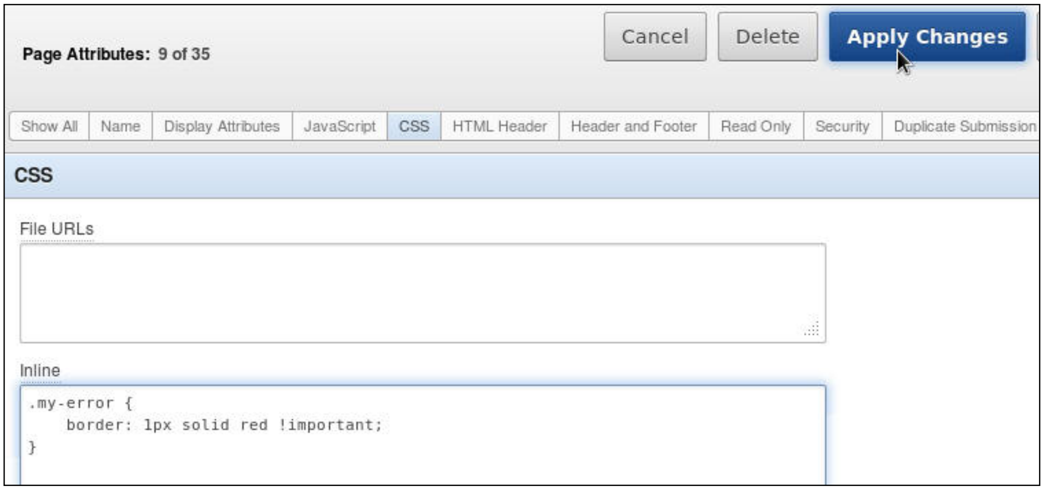
1) You first want to add the class that you want to use in the dynamic action you create. You will add it to the page definition. Right-click **Customer Details** and select **Edit**.



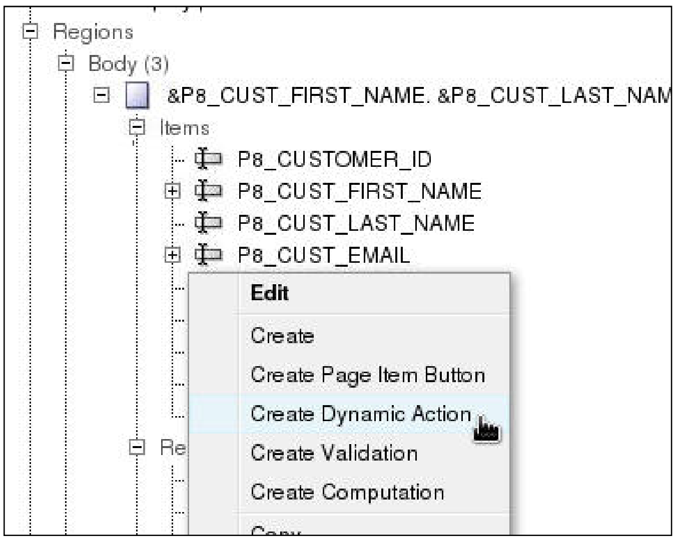
2) Click the **CSS** subtab.



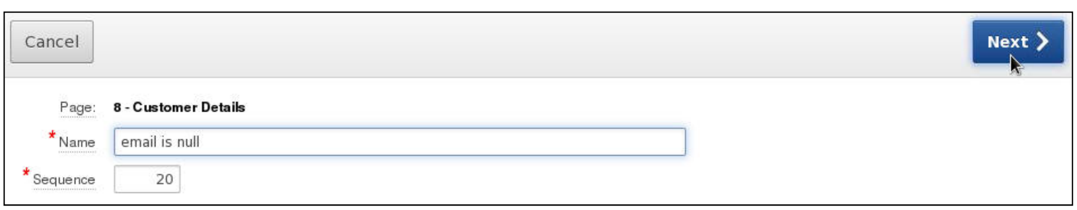
3) Enter the following CSS code (located in the `/home/oracle/labs/files/lab19_1.txt` file) in the Inline area and click **Apply Changes**. Note that you need to make sure you specify the important style so that it overrides the other styles used on the page.



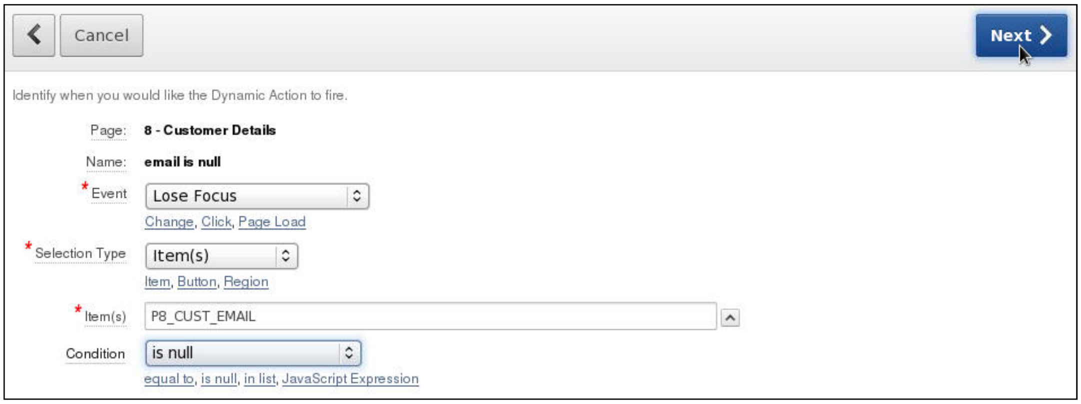
4) Now you can create the dynamic action. Right-click **P<n>_CUST_EMAIL** and select **Create Dynamic Action**.



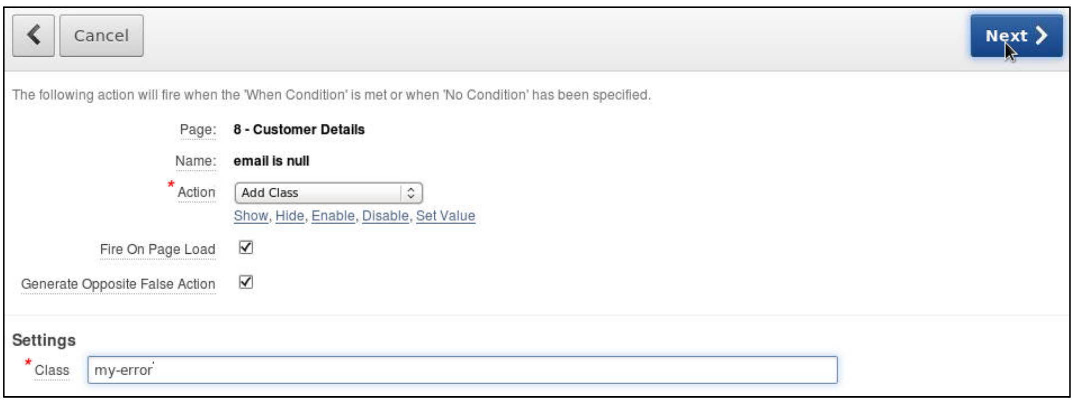
5) Enter **email is null** for Name and click **Next >**.



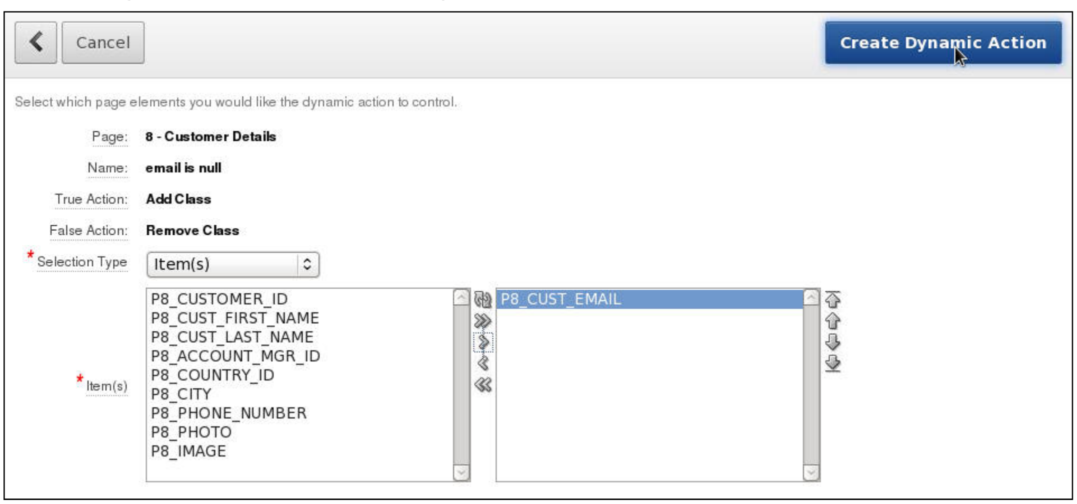
6) Select **Lose Focus** for Event, select **Item(s)** for Selection Type, make sure that **P<n>_CUST_EMAIL** is selected for Item(s), and select **is null** for Condition. Then click **Next >**.



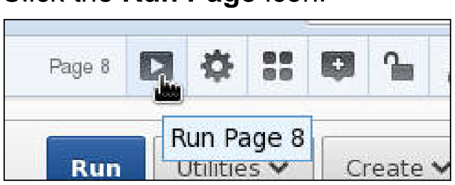
7) Select **Add Class** for Action and enter **my-error** for Class. Then click **Next >**.



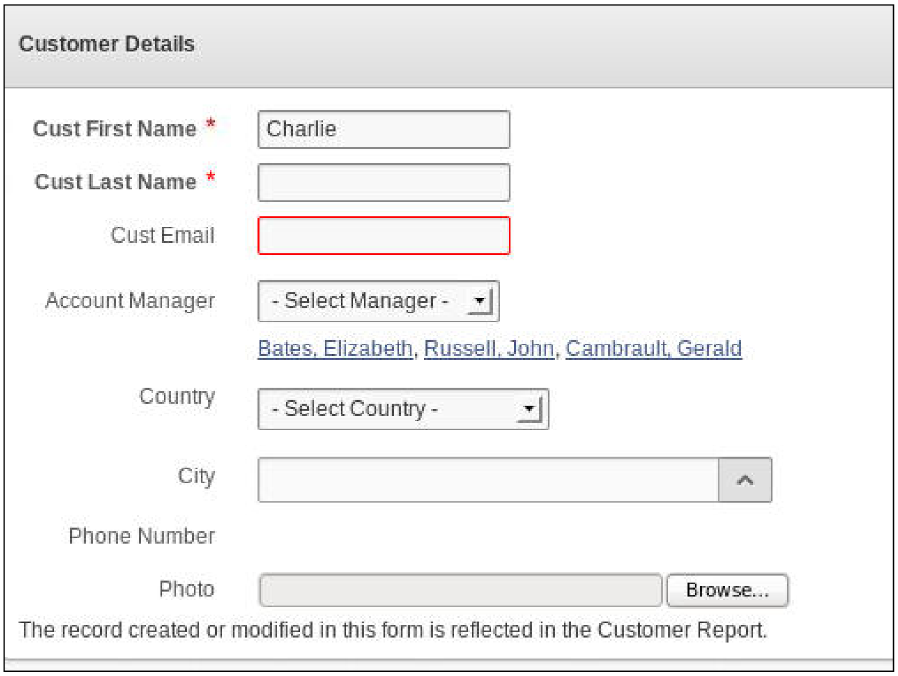
8) Select **Item(s)** for Selection Type. Select the **P<n>_CUST_EMAIL** item and move it to the right. Then click **Create Dynamic Action**.



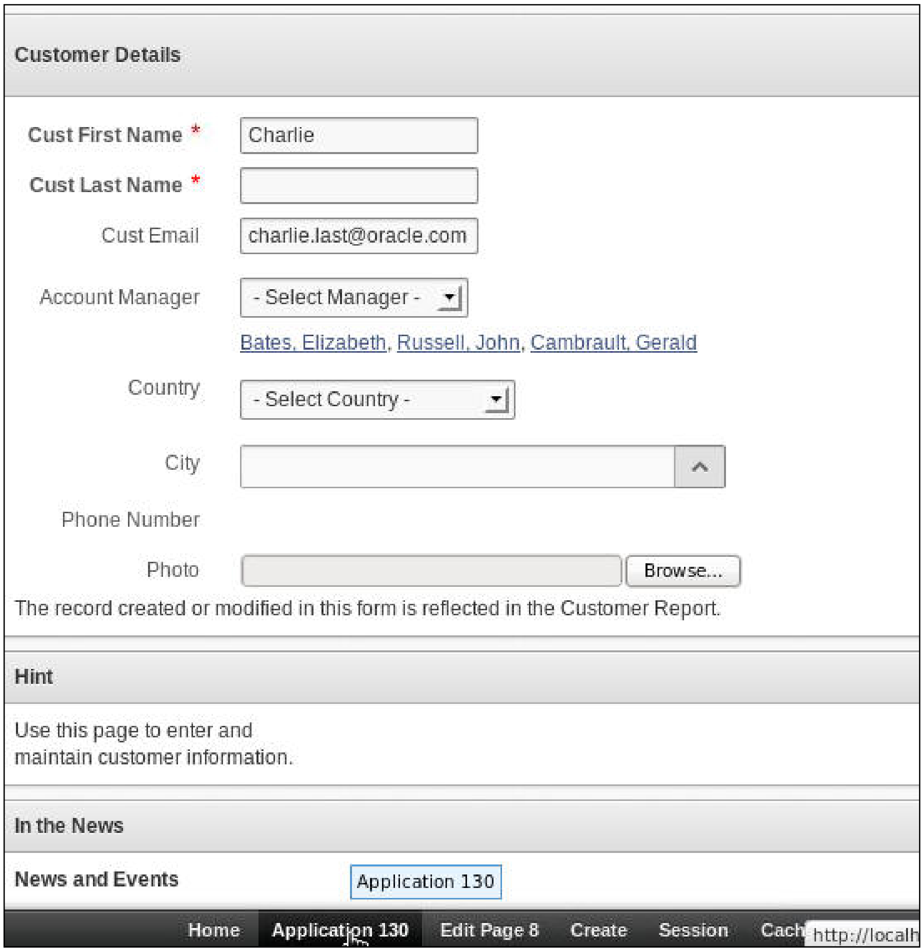
9) Click the **Run Page** icon.



10) Notice that now when you enter a Cust First Name, the Cust Email item is shown and it turns red.

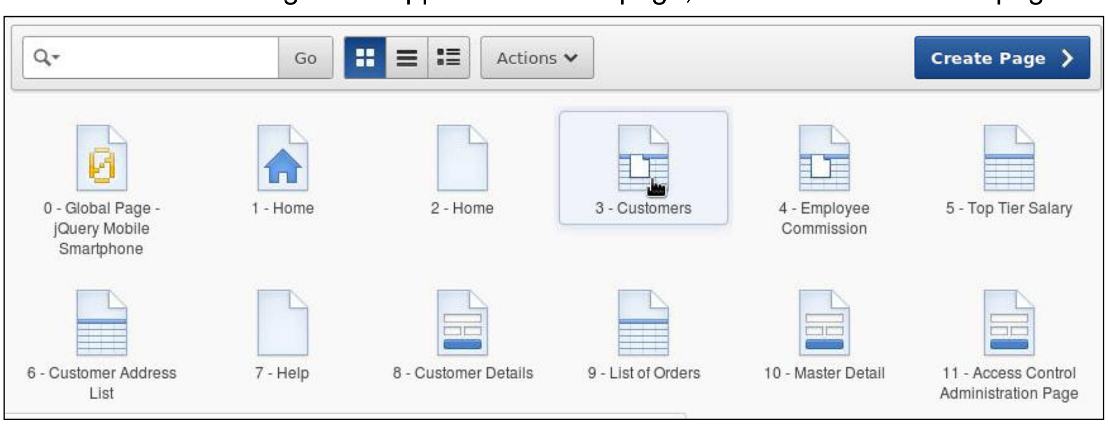


11) Enter an email address, and then click somewhere else on the page to lose focus. You notice that the red border disappears. Click the **Application <*n*>** link on the developer toolbar.

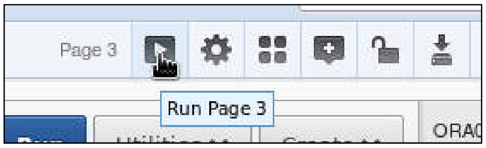


c. **Refreshing the data in a report using Custom filters:** On the Customers page, create a Quick Filter region with two select list items for P<n>_CUSTOMER_ID and P<n>_JOB_ID. Create a dynamic action on the select lists to refresh the report when the values of the select lists change. Make sure that you add the two filter items to the Page Items to Submit.

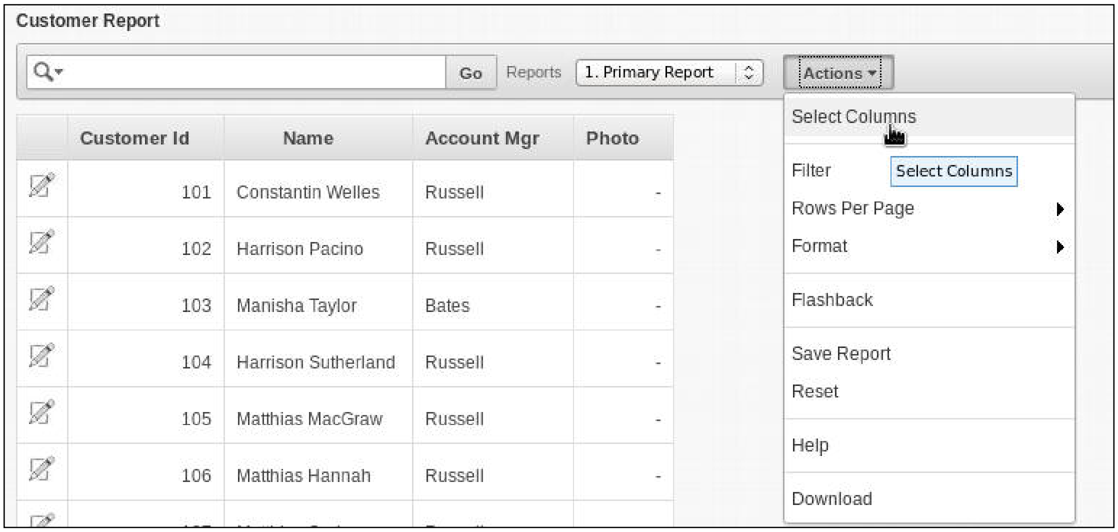
1) From the Order Management application home page, select the **Customers** page.



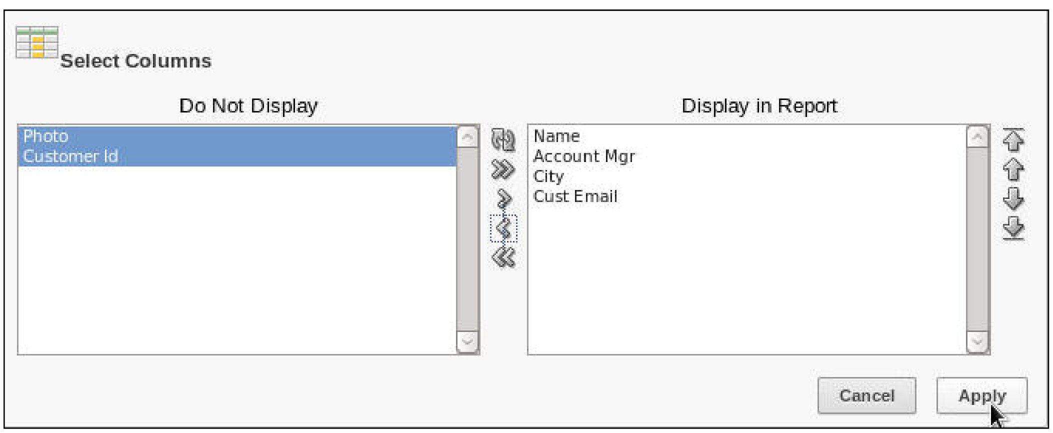
2) Click the **Run Page** icon.



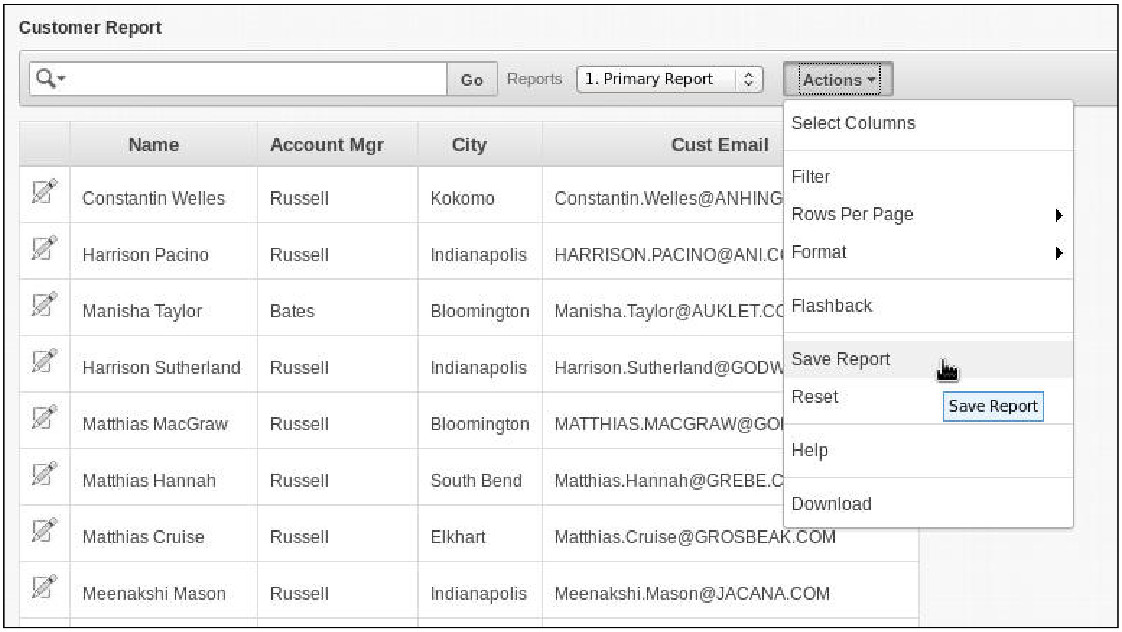
3) You want to change the columns that the report displays. Click **Actions** and select **Select Columns**.



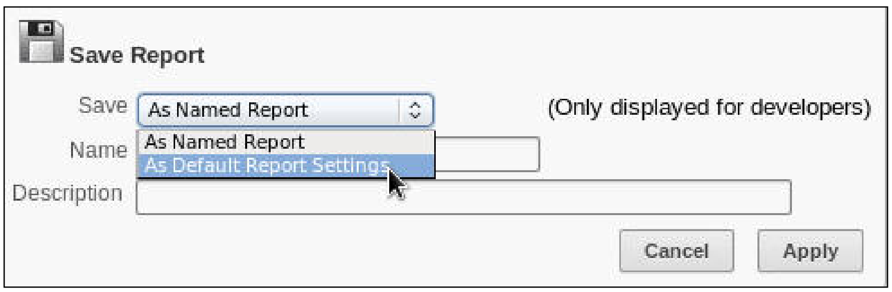
4) Remove **Photo** and **Customer Id** from the list of Displayed Columns and add **Cust Email** to the list of Displayed Columns. Then click **Apply**.



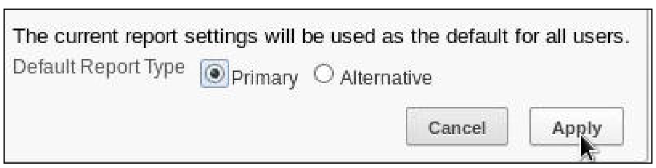
5) You want to save these settings. Click **Actions** and select **Save Report**.



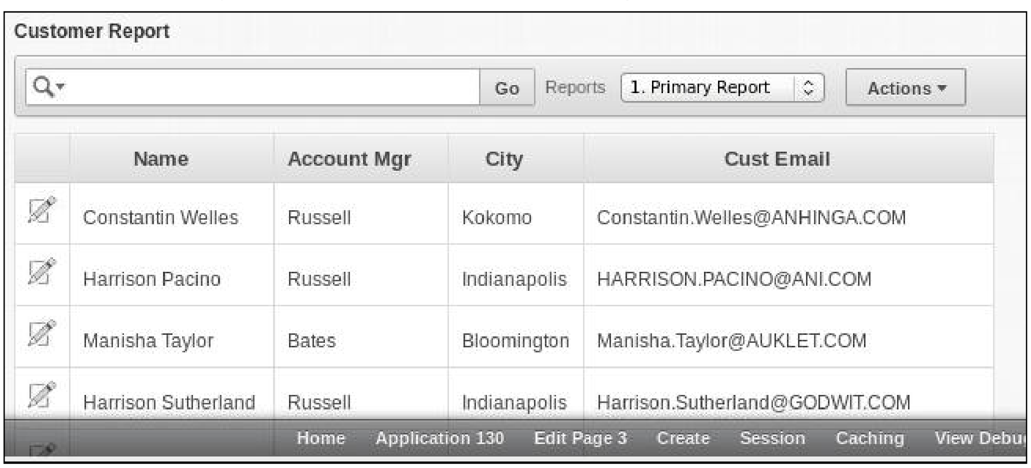
6) Select **As Default Report Settings** from the list of Save options.



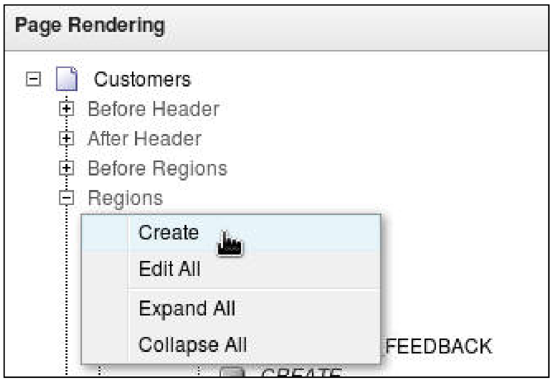
7) Make sure that **Primary** is selected and click **Apply**.



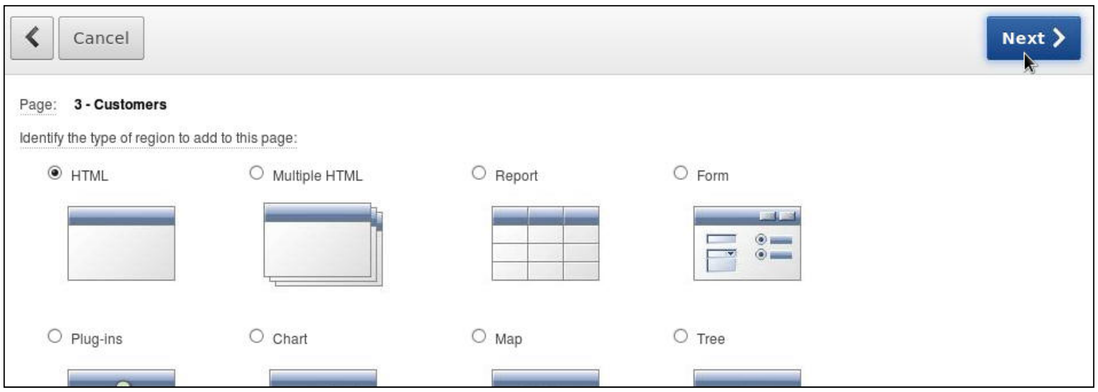
8) The report settings are saved. Click the **Edit Page** link on the Developer toolbar.



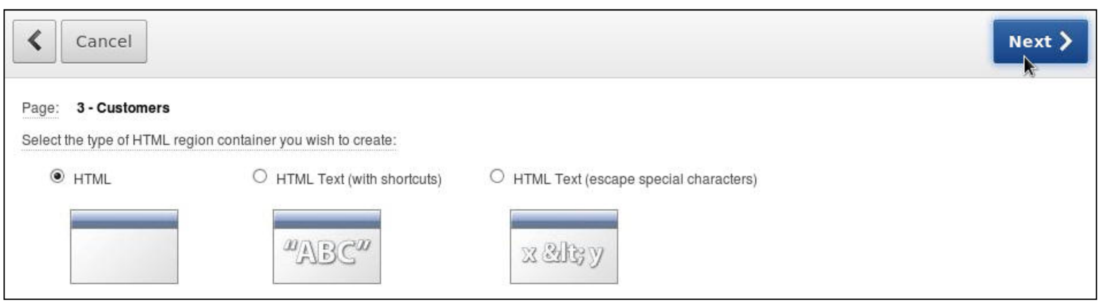
9) You want to create the region in which the filtered items will be created. Right-click **Region** and select **Create**.



10) Select **HTML** and click **Next**.



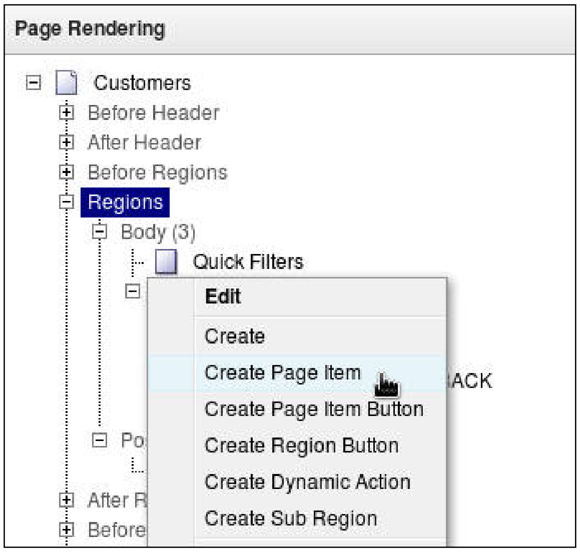
11) Select **HTML** and click **Next**.



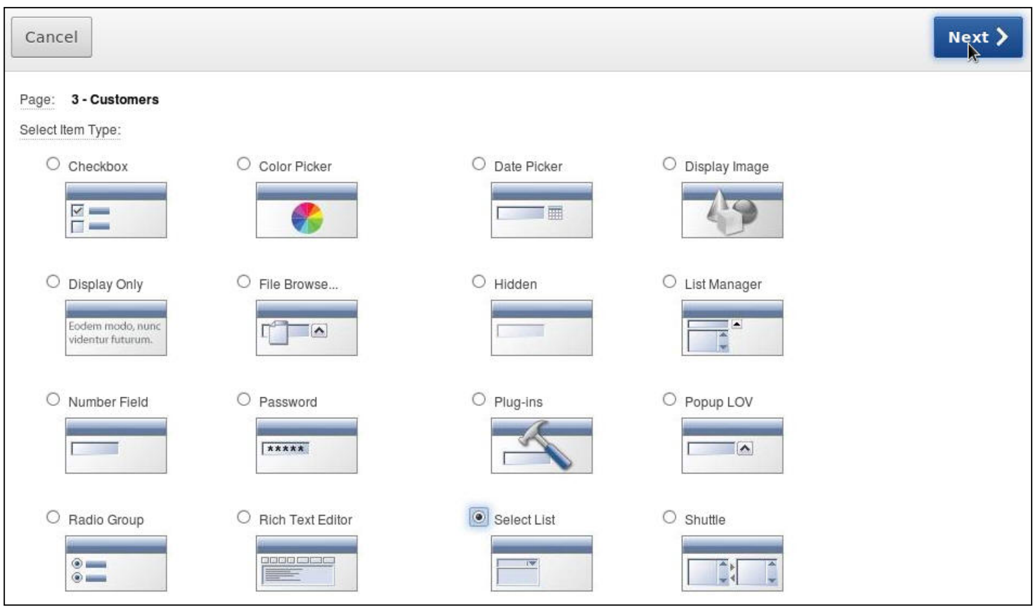
12) Enter `Quick Filters` for Title and select **Report Filter – Single Row** for Region Template. Replace the sequence with `5` so that this region appears before the interactive report region on the page. Click **Create Region**.

13) On the Quick Filters region, create two filter items: **`P<N>_CUSTOMER_ID`** and **`P<n>_CITY_ID`** to filter the employee report by department name and job ID.

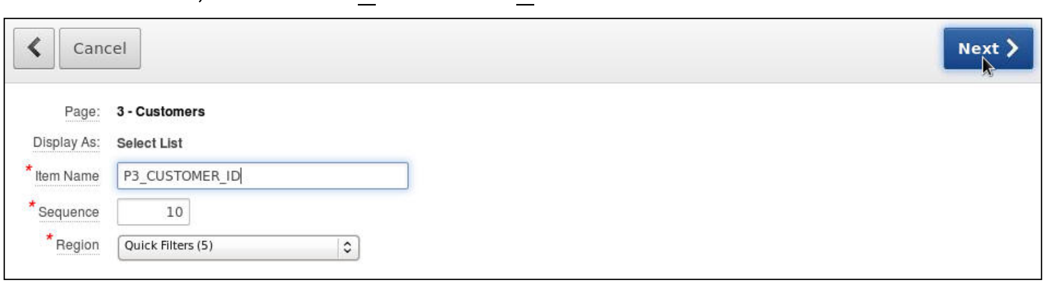
a) Right-click the **Quick Filters** region and select **Create Page Item**.



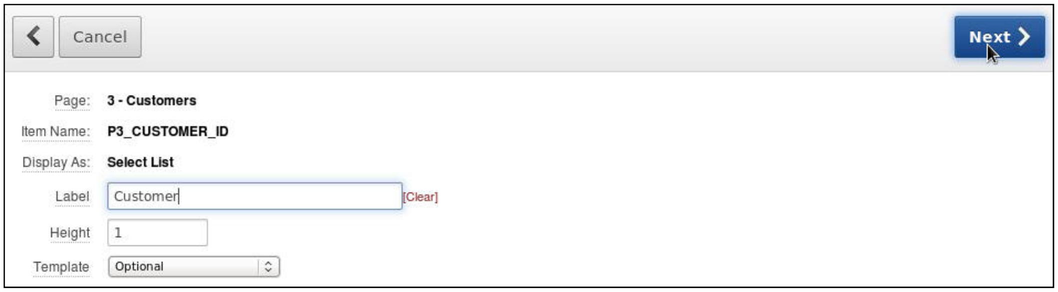
b) Choose **Select List** and click **Next**.



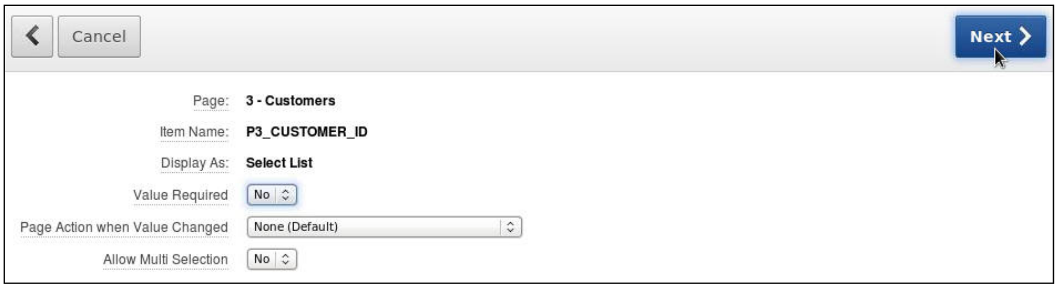
c) For Item Name, enter **`P<n>_CUSTOMER_ID`**. Click **Next**.



d) Change the Label to `Customer` and accept the remaining defaults for Item Attributes. Click **Next**.

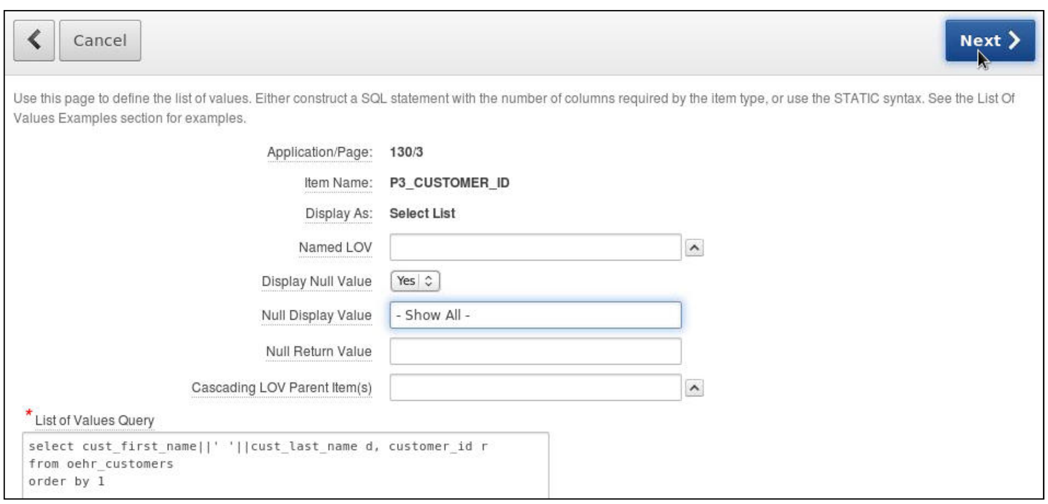


e) For Settings, accept defaults and click **Next**.

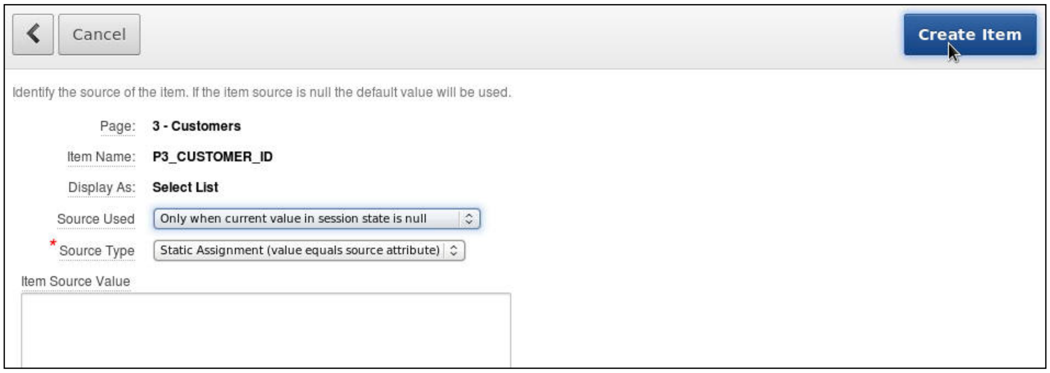


f) Ensure Display Null Value is **Yes**. For Null Display Value, enter **`- Show All -`**. For List of Values Query, enter the following query and click **Next**. Use `home/labs/files/lab19_2.txt`

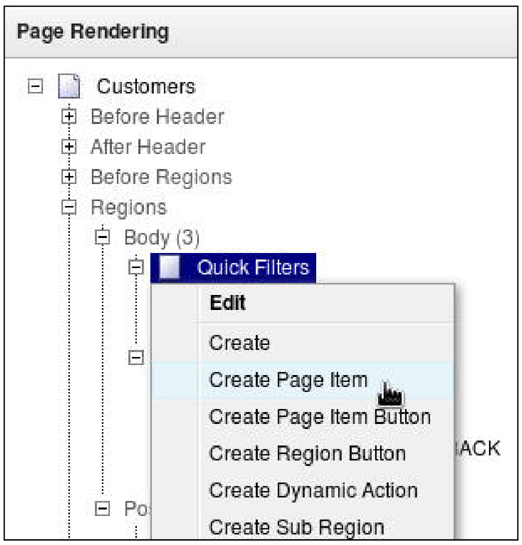
```
select cust_first_name||' '||cust_last_name d, customer_id r
from oehr_customers
order by 1
```



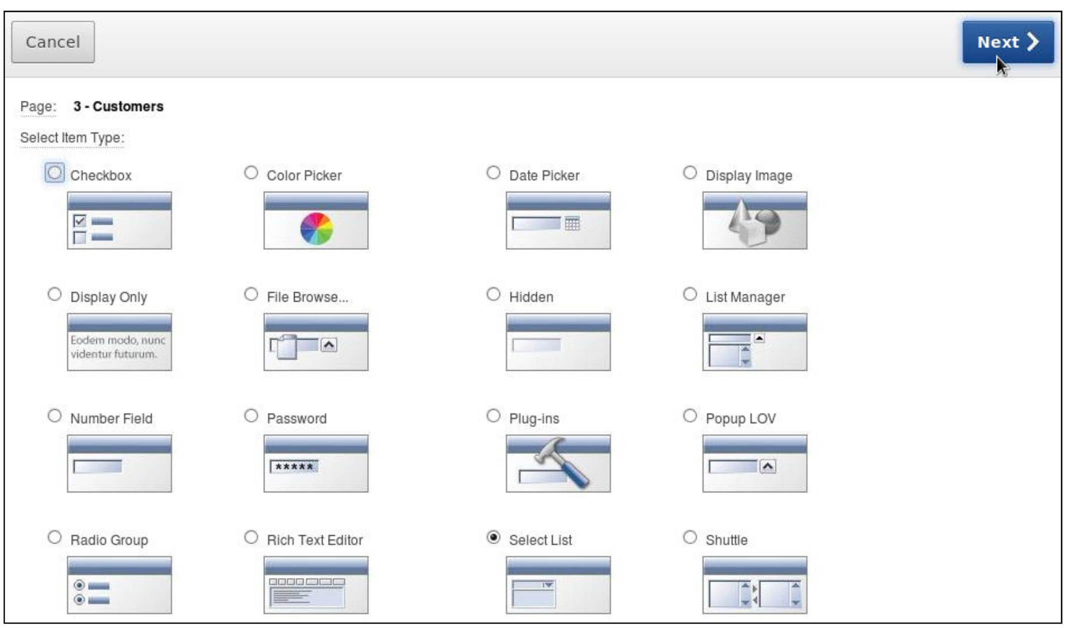
g) Click **Create Item**.



h) You want to create one more page item for City. Right-click the **Quick Filters** region and select **Create Page Item**.

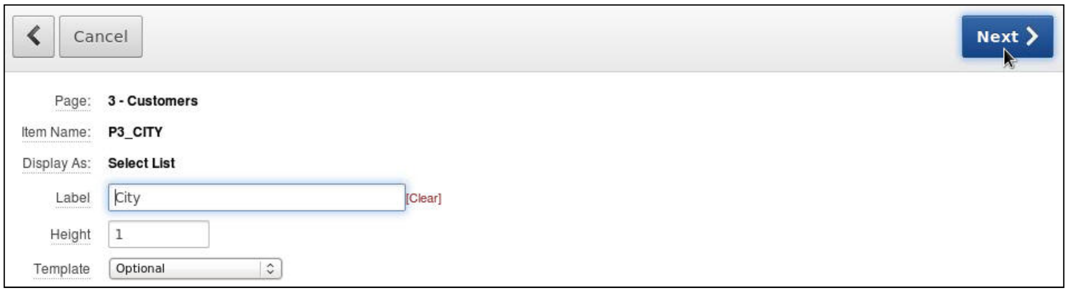


i) Choose **Select List** and click **Next**.

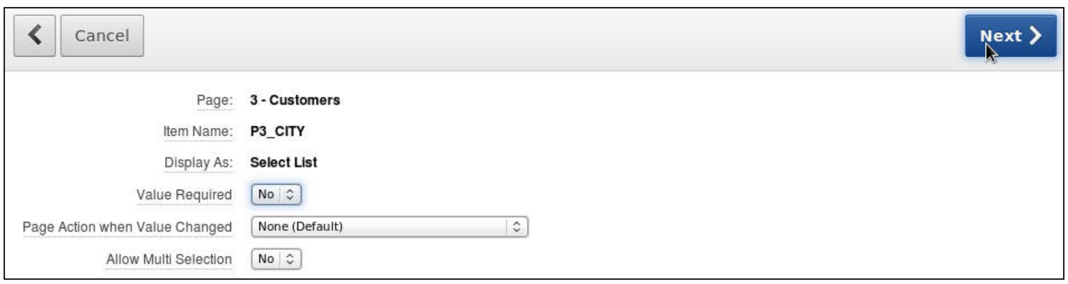


j) For Item Name, enter **P<*n*>_CITY**. Click **Next**.

k) Accept the defaults and click **Next**.

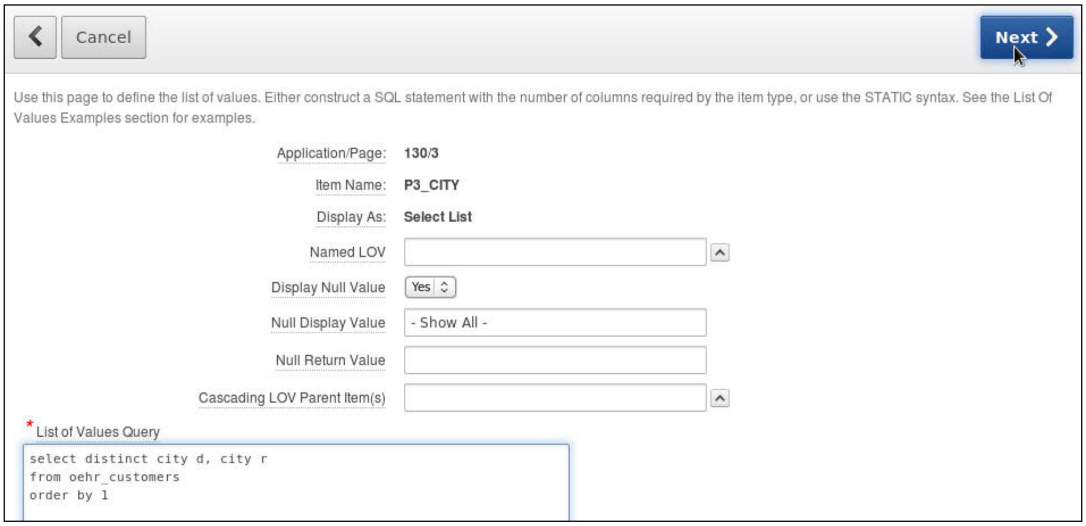


l) Accept the defaults and click **Next**.

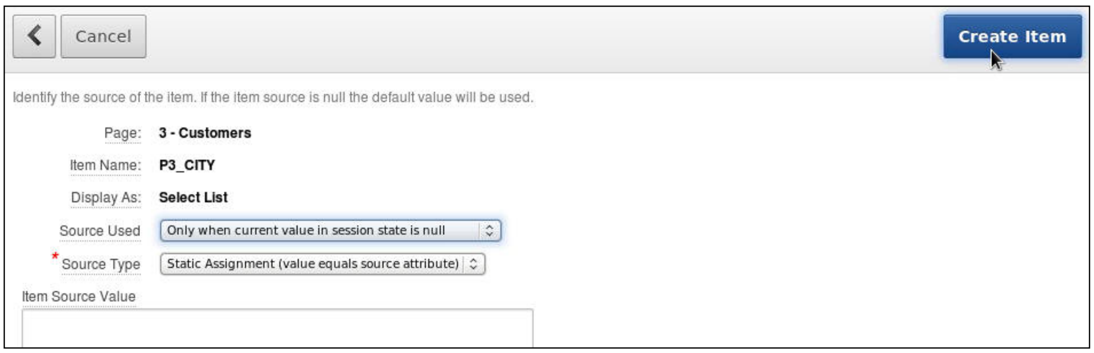


m) Ensure Display Null Value is set to **Yes**. For Null Display Value enter **`- Show All -`**. For List of Values Query, enter the following query and click **Next**. Use `home/labs/files/lab19_3.txt`.

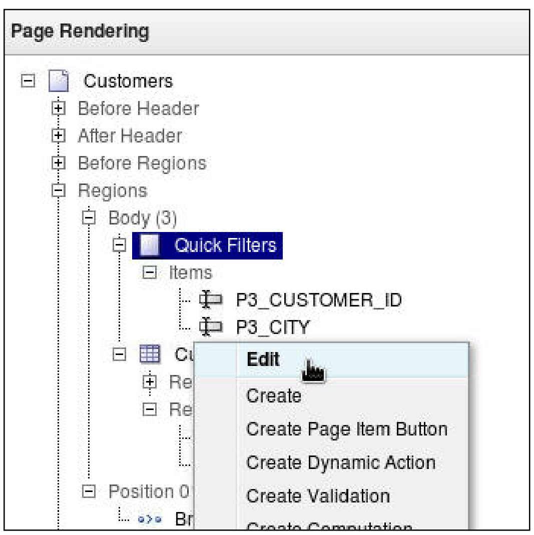
```
select distinct city d, city r
from oehr_customers
order by 1
```



n) Click **Create Item**.

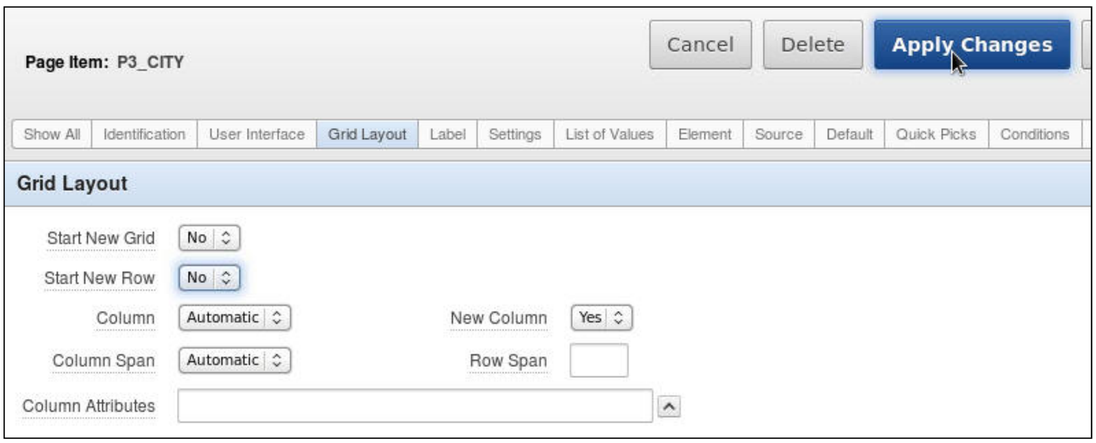


14) Right-click **`P<n>_CITY`** and select **Edit**.

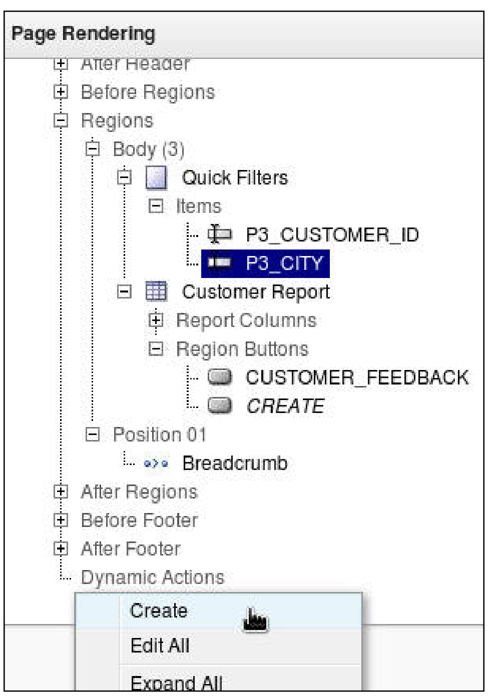


15) Click the **Grid Layout** subtab.

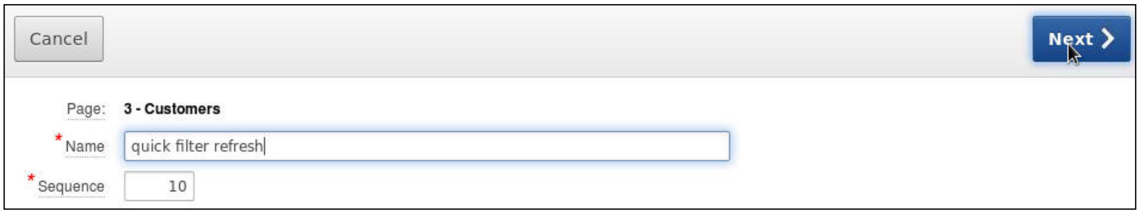
16) Select **No** for **Start New Row.** Click **Apply Changes.**



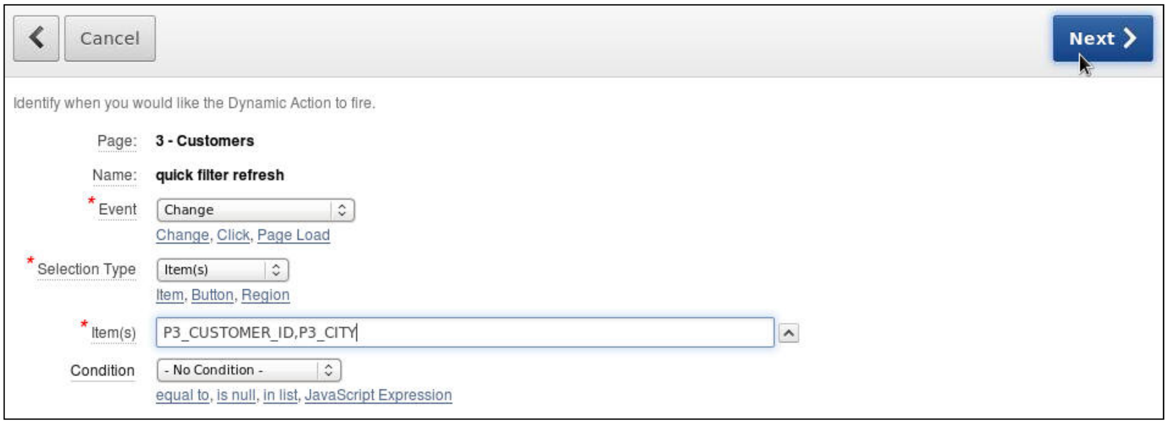
17) Right-click the **Dynamic Actions** node and select **Create**.



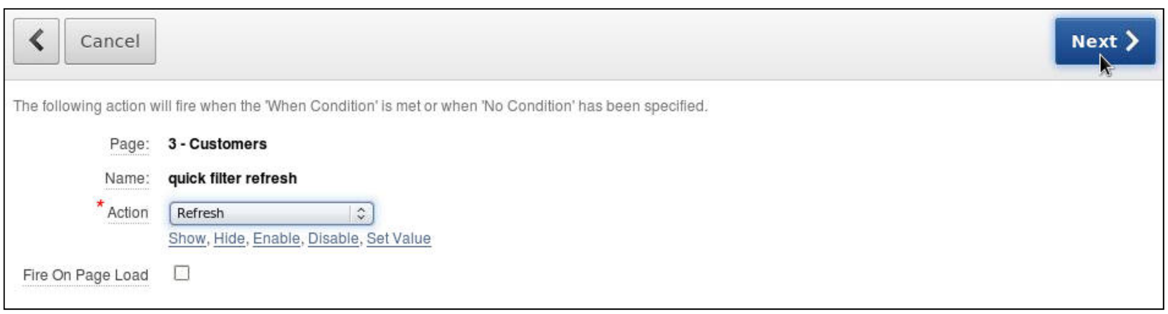
18) Enter **`QUICK FILTER REFRESH`** for Name, click **Next**.



19) Select **Change** for Event, **Item(s)** for Selection Type. For Item(s), enter **`P<n>_CUSTOMER_ID, P<n>_CITY`**. Accept the defaults and click **Next**.

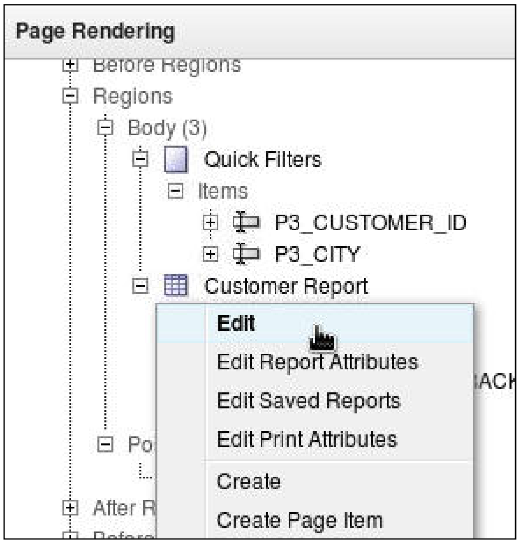


20) Select **Refresh** for Action. Ensure that the Fire On Page Load check box is deselected and click **Next**.

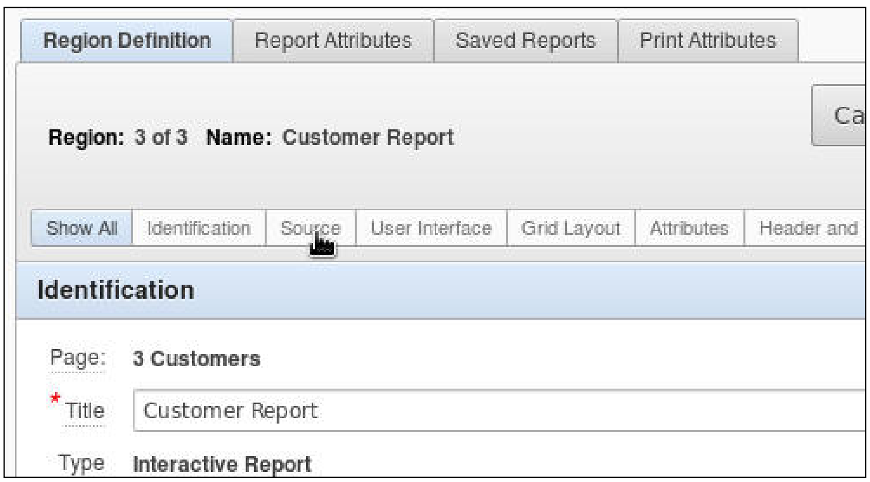


21) Select **Region** for Selection Type and select **Customer Report** for Region and click **Create Dynamic Action**.

22) Under Regions, right-click the **Customer Report** and select **Edit**.

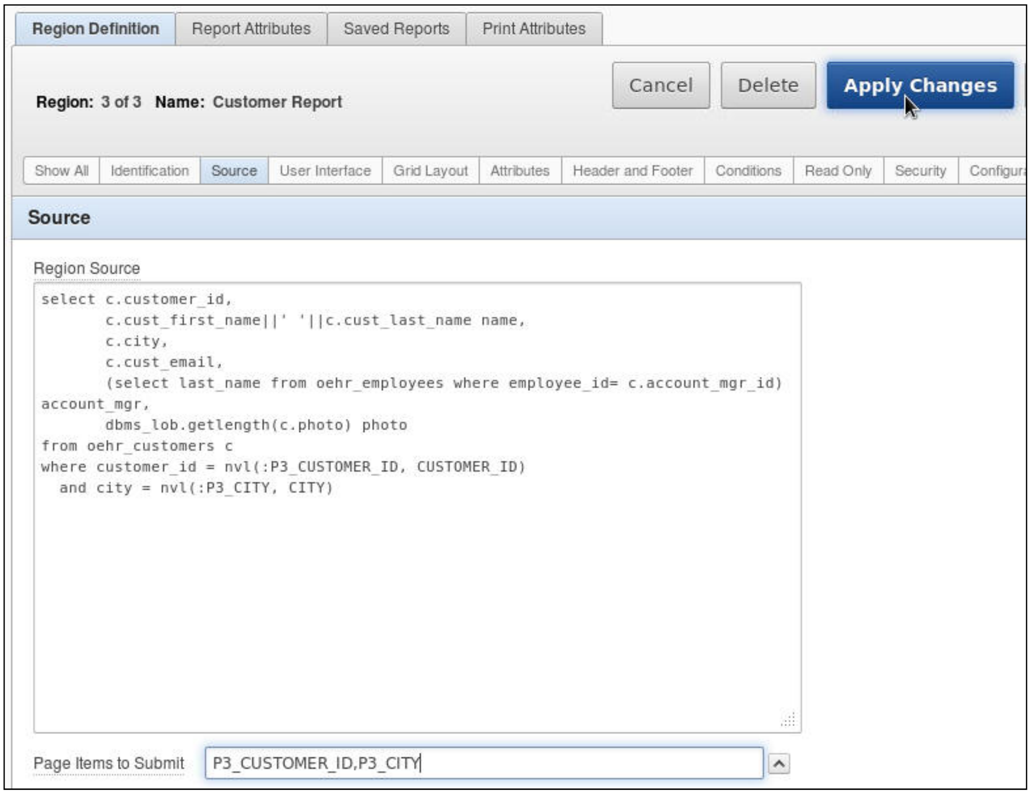


23) Click the **Source** tab.

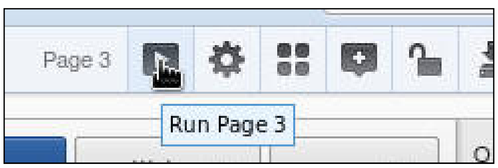


24) Add the following `where` clause (located in the `/home/oracle/labs/files/lab19_4.txt` file) to the `SELECT` statement. Then enter **`P<n>_CUSTOMER_ID,P<n>_CITY`** for Page Items to Submit. Click **Apply Changes**.

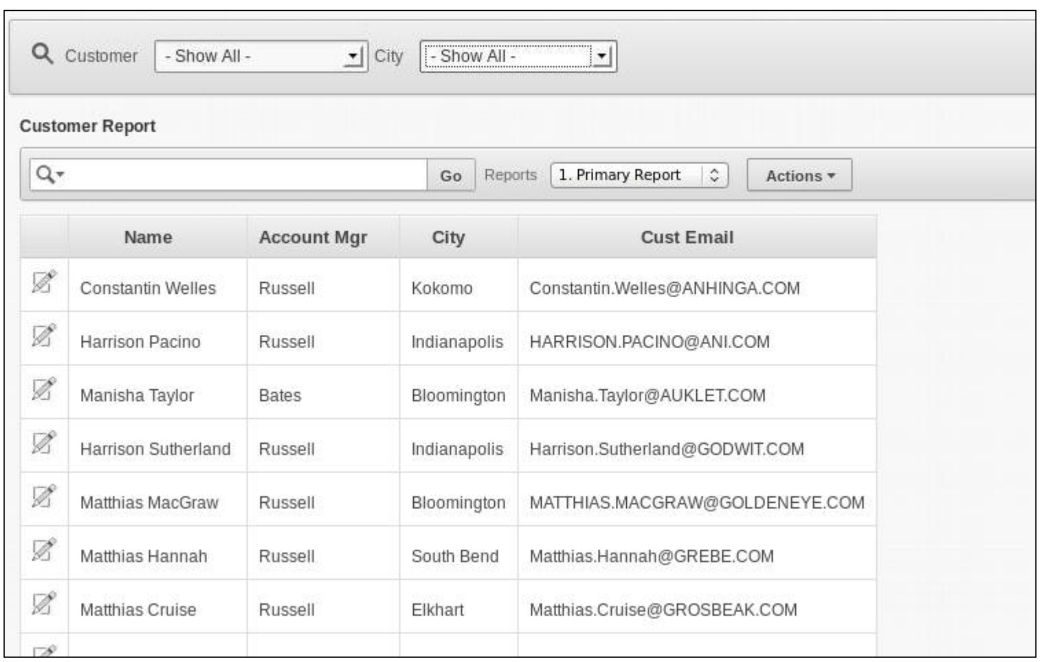
```
where customer_id = nvl(:P<n>_CUSTOMER_ID, CUSTOMER_ID)
    and city = nvl(:P<n>_CITY, CITY)
```



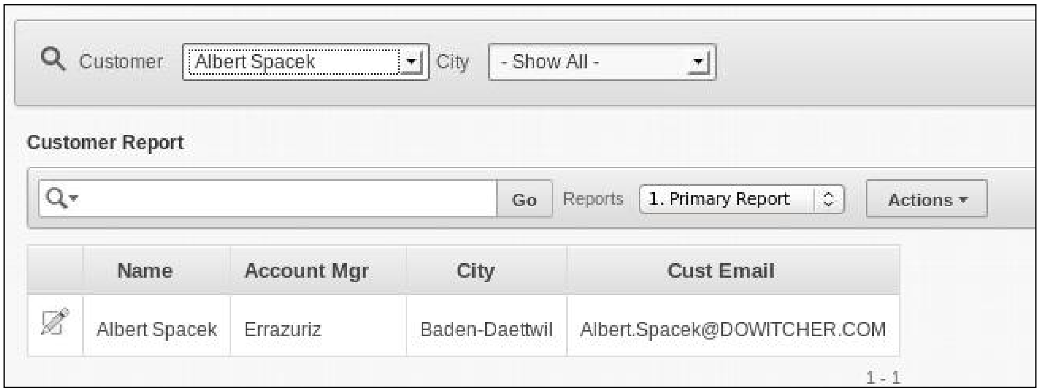
25) Click the **Run Page** icon.



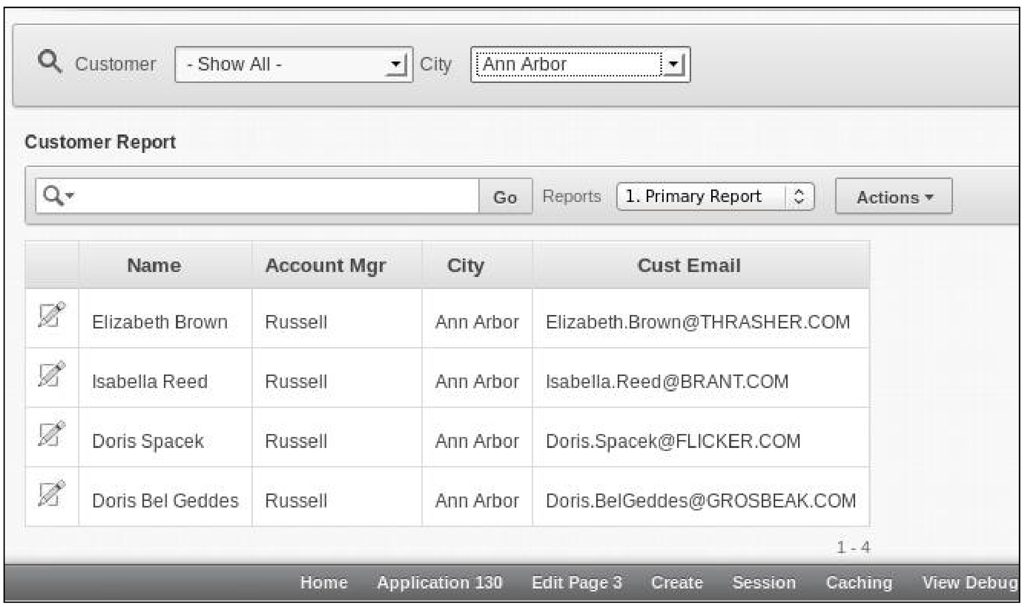
26) Note that when both filters are set to - Show All -, all the records are displayed.



27) Select a customer from the list. What do you observe?



28) Select **-Show All-** for Customer and select a city from the list. What do you observe now? When done, click the **Application <*n*>** link on the Developer toolbar.



Solution 19-2: Importing and Using Plug-Ins

Overview

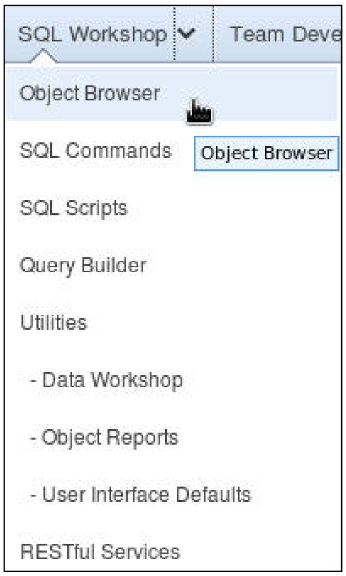
In this practice, you import and use an item type plug-in and a dynamic action plug-in.

Tasks

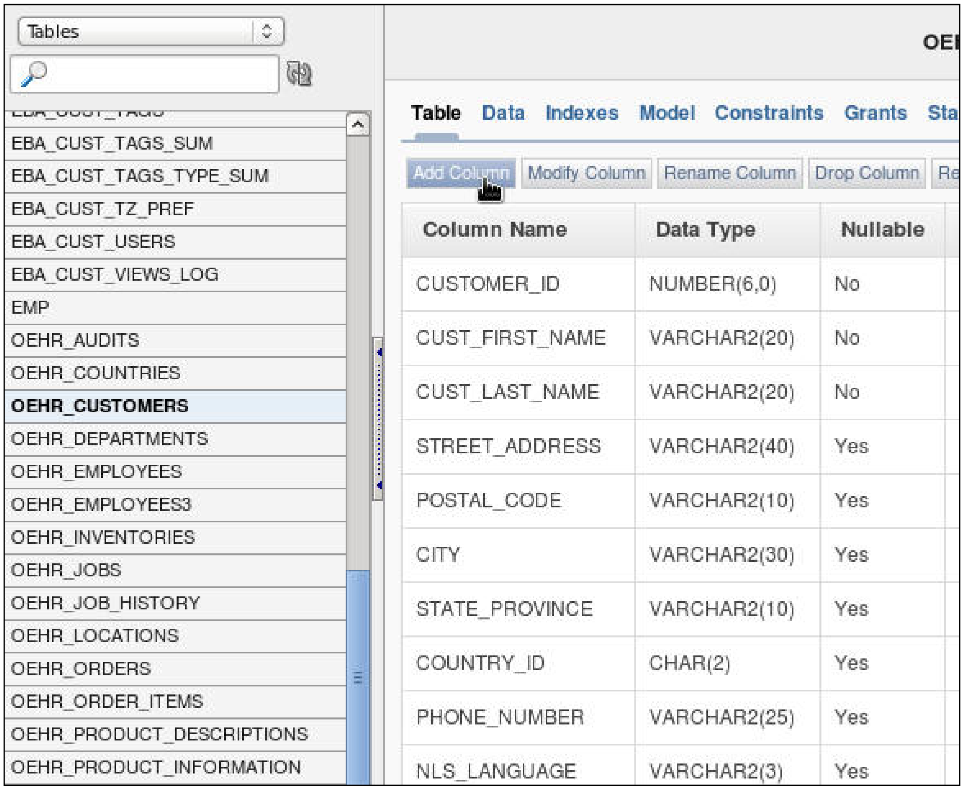
a. **Importing and Using an Item Type Plug-in**: In this practice, you import the plug-ins, and use them in your application.

1) Add a rating column to the `OEHR_CUSTOMERS` table.

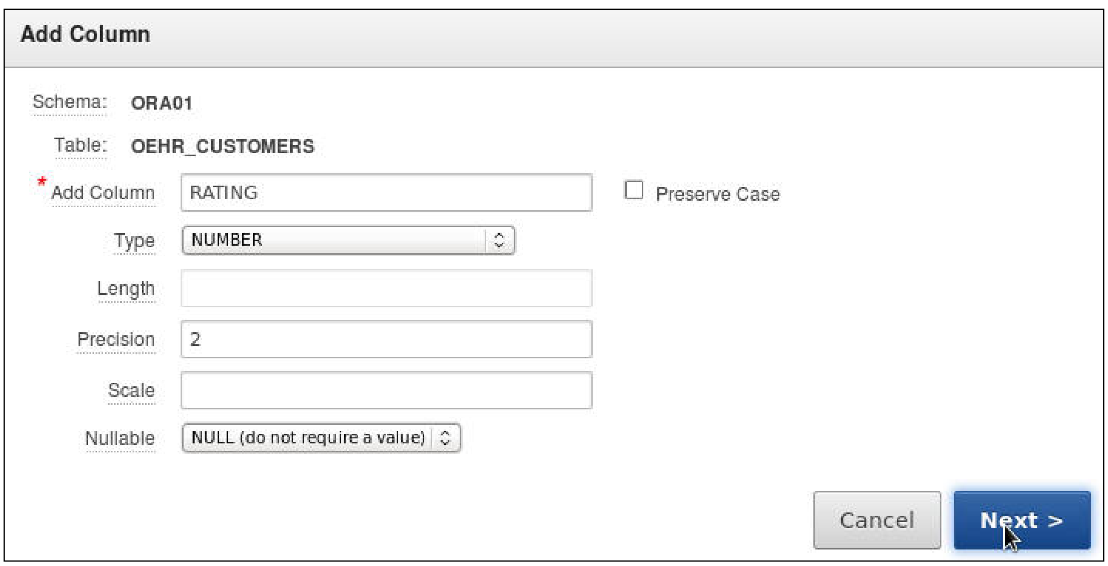
a) Select **SQL Workshop** > **Object Browser**.



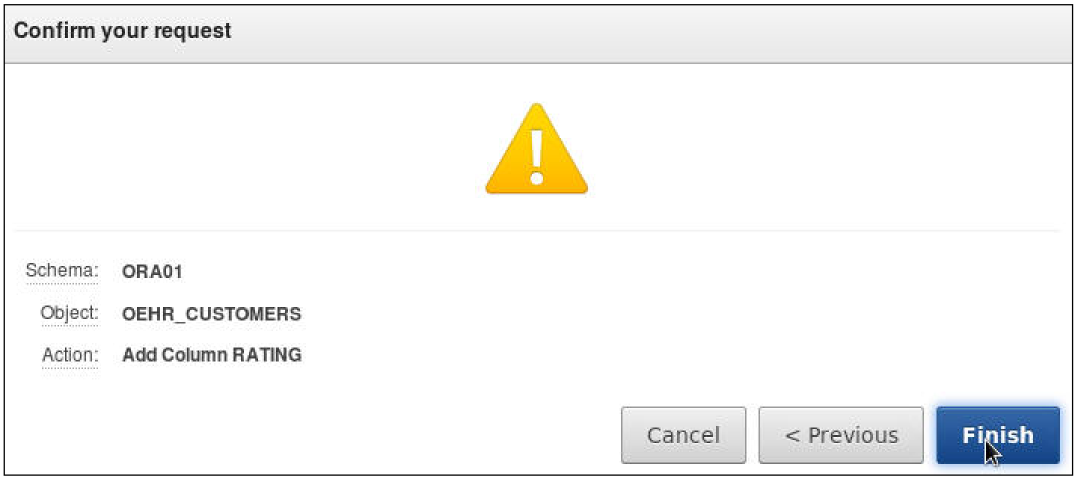
b) Select `OEHR_CUSTOMERS` from the list of tables, and then click the **Add Column** subtab.



c) Enter `RATING` for Add Column and `2` for Precision. Click **Next >**.

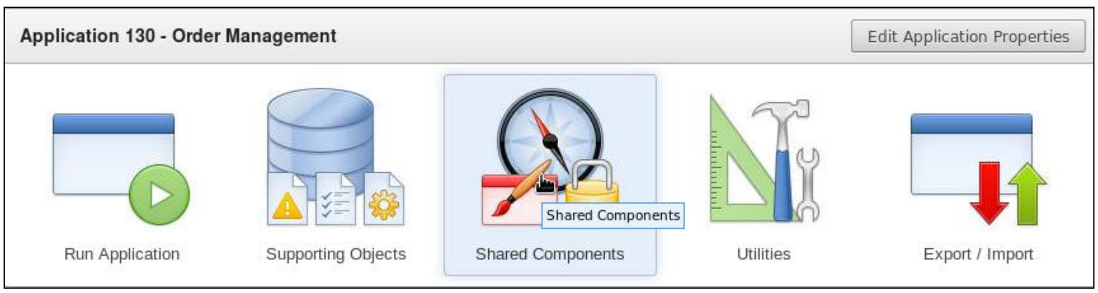


d) Click **Finish**.

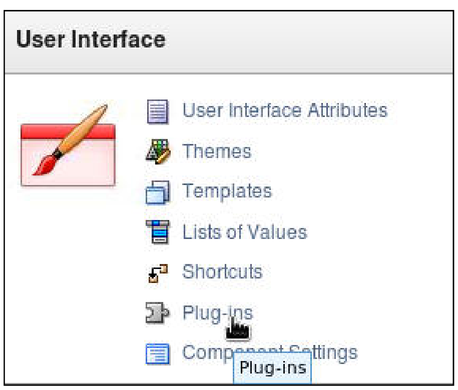


2) Import the Star Rating Item Plug-In and the Notification Dynamic action Plug-in files.

a) Navigate to your Order Management application. Click **Shared Components**.

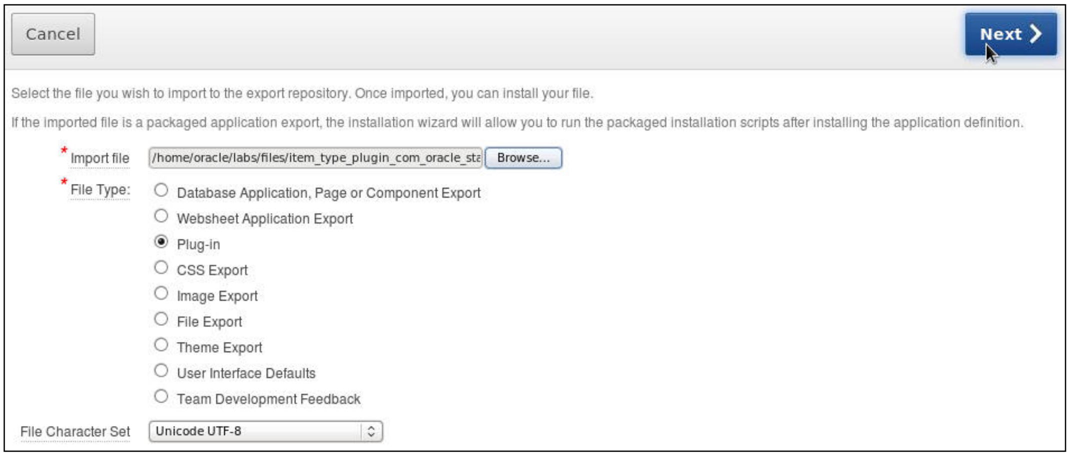


b) Under User Interfaces, select **Plug-Ins**.

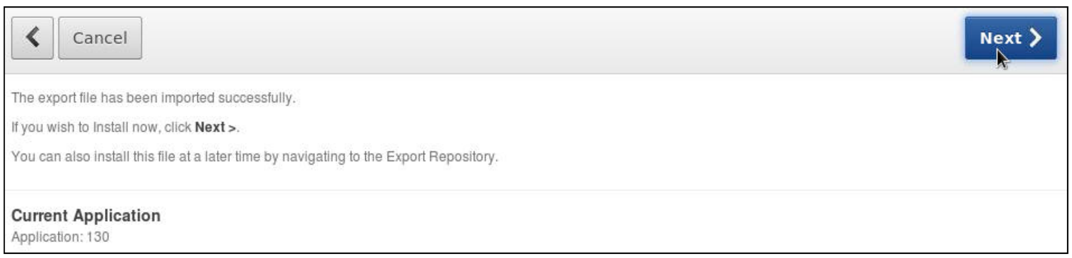


c) Click **Import**.

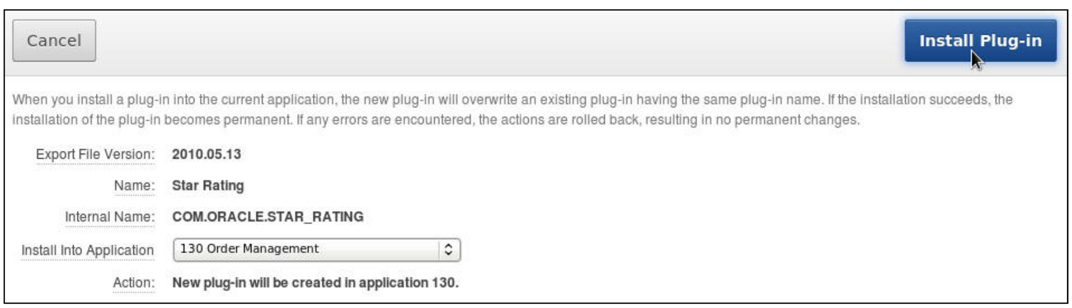
d) Click **Browse** and select `home/oracle/labs/files/item_type_plugin_com_oracle_apex_star_rating.sql`. Click **Open**. Then click **Next**.



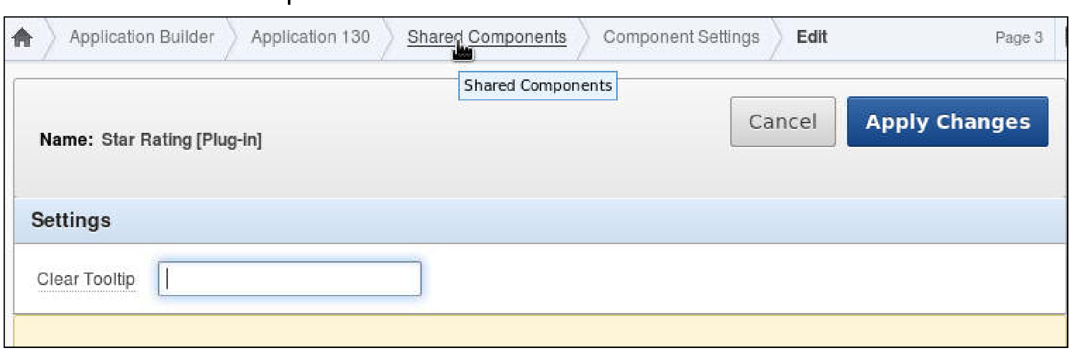
e) After the file is imported, click **Next** to install it.



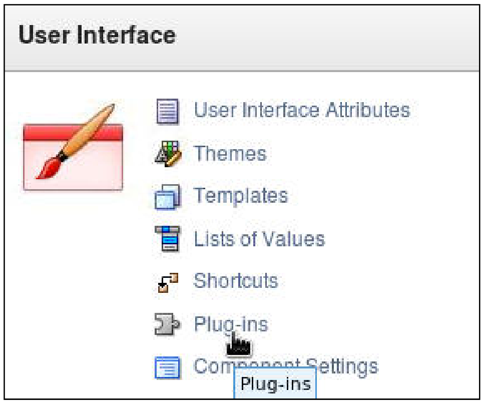
f) Make sure the **Order Management** application is selected and click **Install Plug-in**. Note that if the plug-in already exists, it will be replaced.



g) The plug-in was installed. You will be asked to enter a component setting. Click the Shared Components breadcrumb.

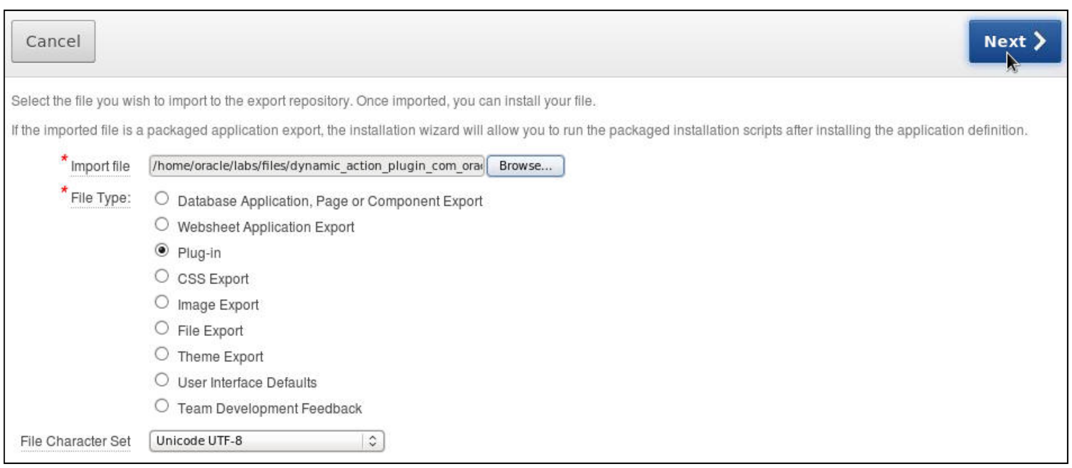


h) Under User Interface, select **Plug-Ins**.

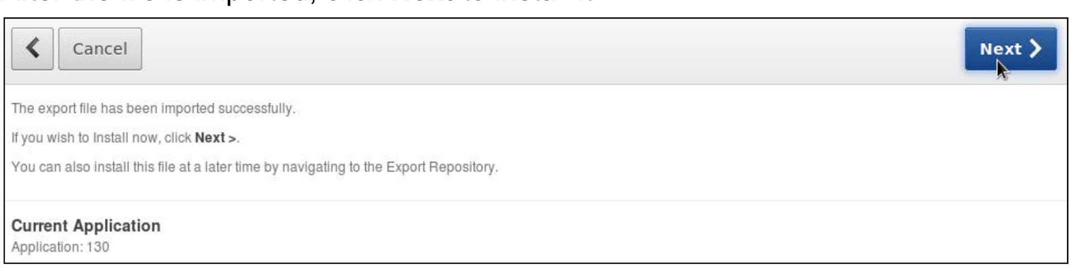


i) You want to install one more plug-in. Click **Import**.

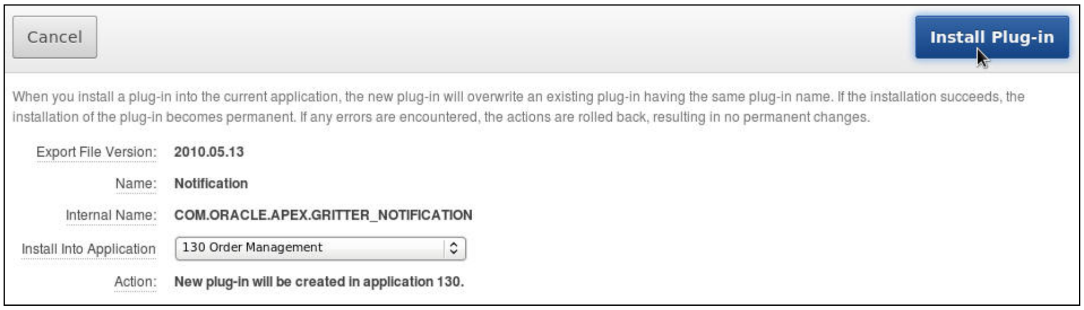
j) Click **Browse** and select `home/oracle/labs/files/dynamic_action_plugin_com_oracle_apex_gritter_notification.sql`. Click **Open**. Then click **Next**.



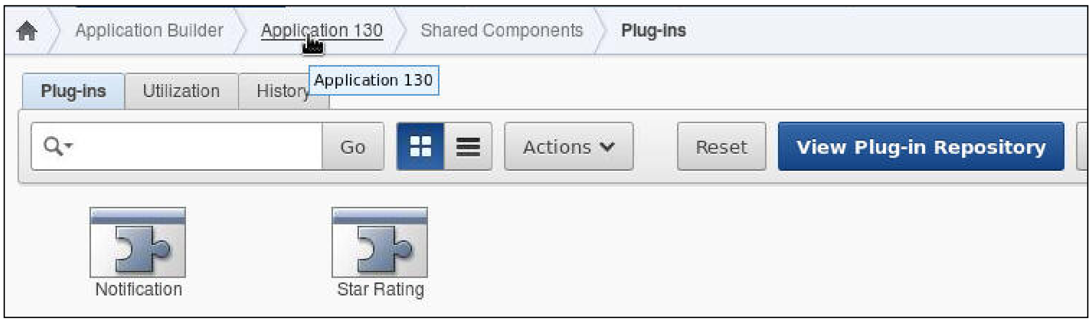
k) After the file is imported, click **Next** to install it.



l) Select the **Order Management** application and click **Install Plug-in**. Note that if the plug-in already exists, it will be replaced.

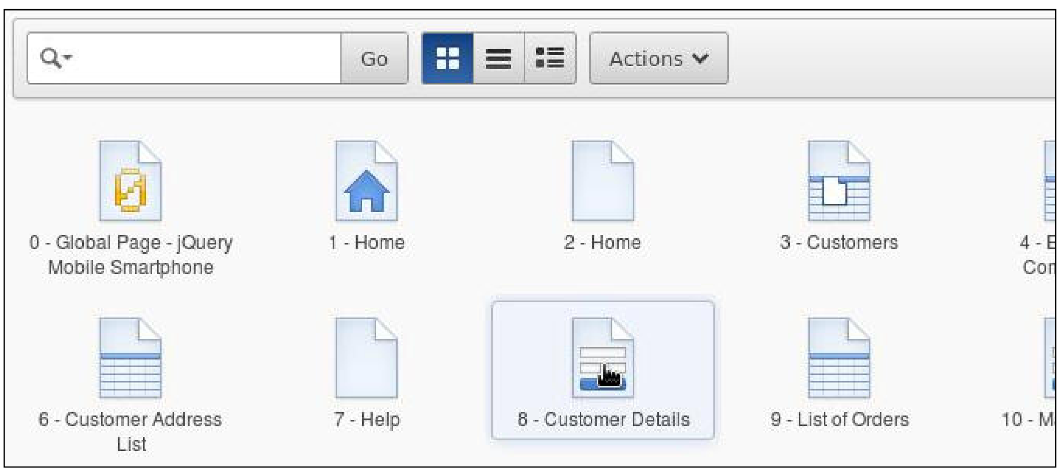


m) The plug-ins are installed successfully. Click the **Application** breadcrumb.

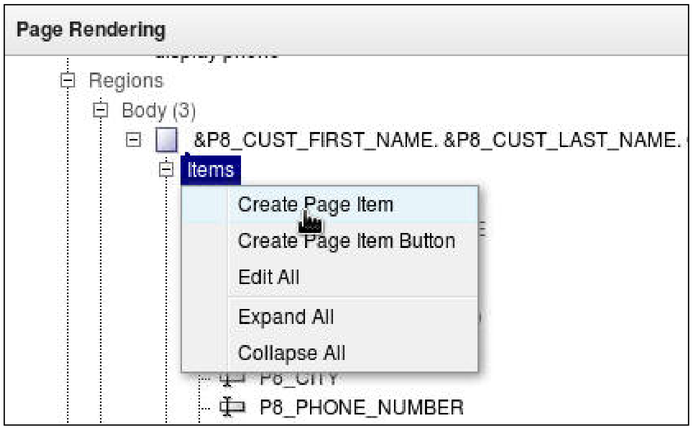


3) Add the Star Rating Item Plug-in to your Customer Details page.

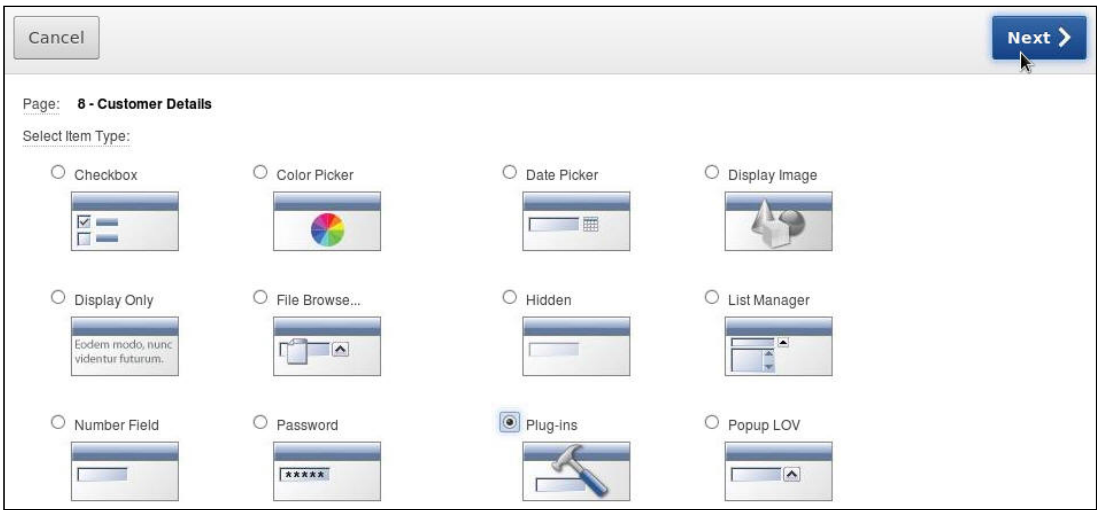
a) Navigate to the **Customer Details** page.



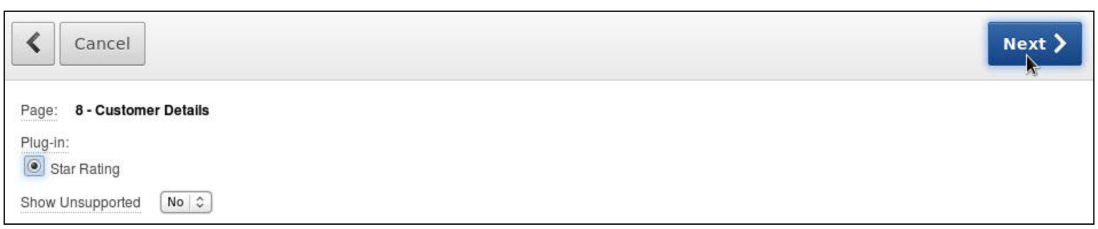
b) Right-click **Items** and select **Create Page Item.**



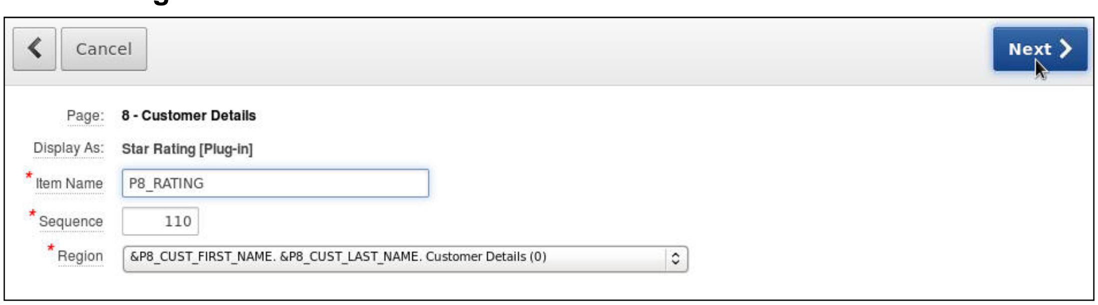
c) Select the **Plug-ins** item type and click **Next >**.



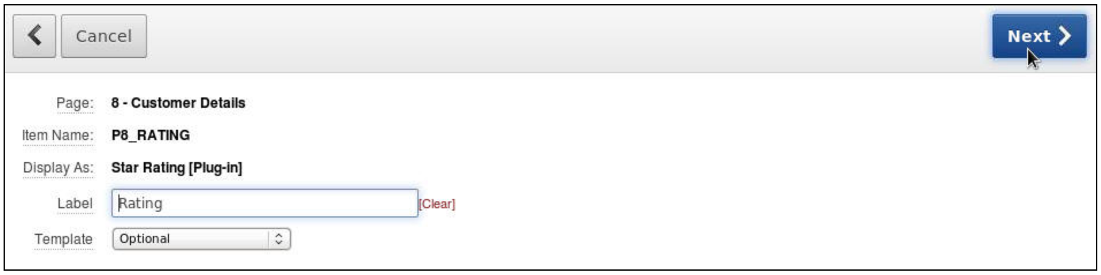
d) Select the **Star Rating** option and click **Next >**.



e) Enter **Rating** for Item Name and click **Next >**.

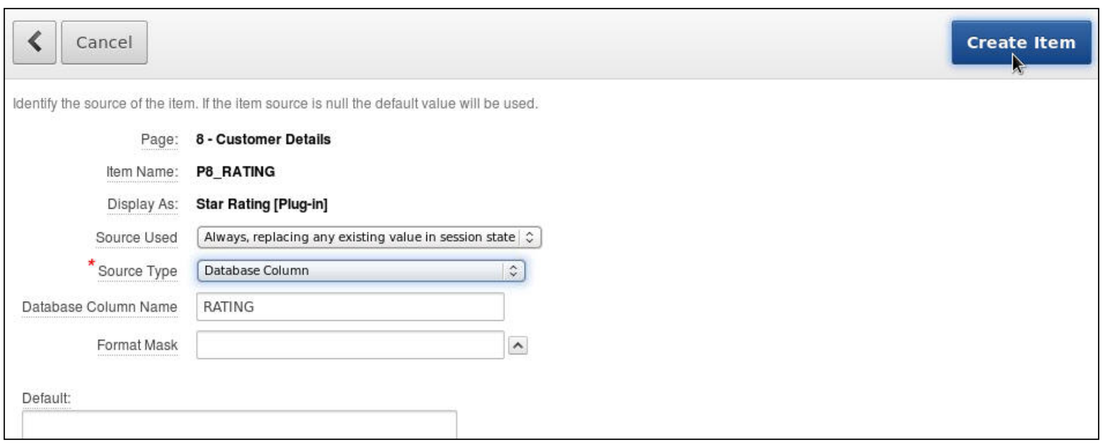


f) Accept the defaults and click **Next >**.

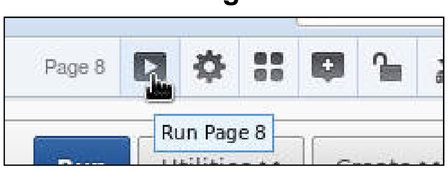


g) Make sure “Number of Stars” is set to **5** and click **Next >**.

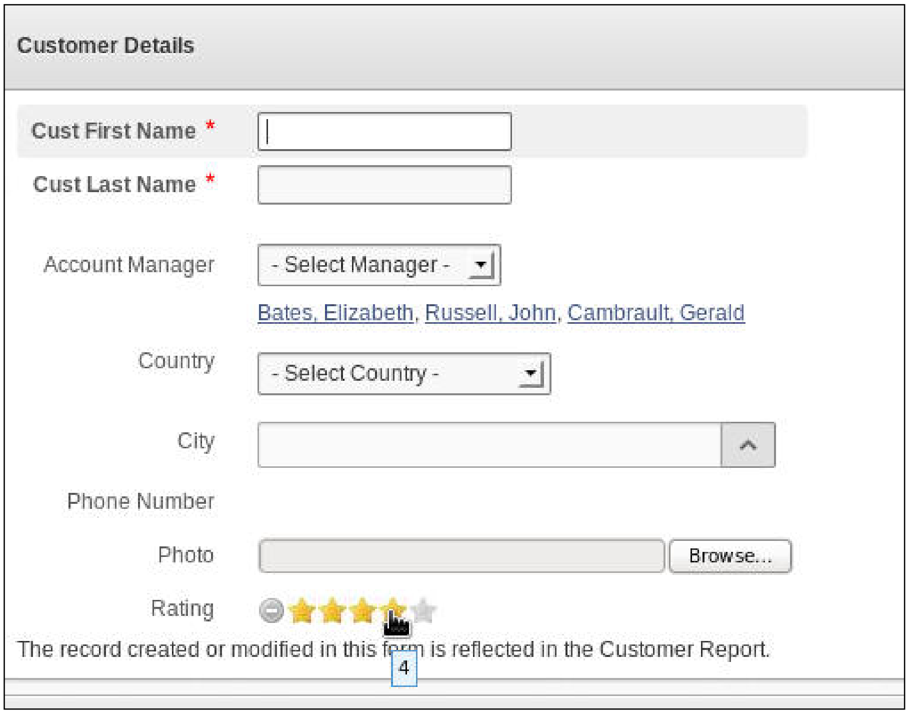
h) Select **Database Column** for Source type (automatically it should select the `RATING` column) and click **Create Item**.



i) Click the **Run Page** icon.

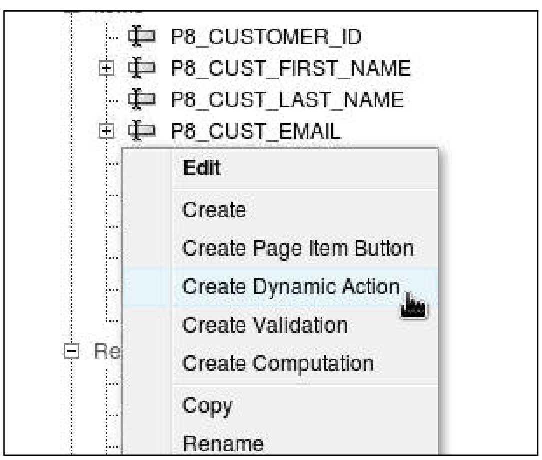


j) Note that the page now includes the Rating item. If you move your cursor over the stars, you can see the value. Click the **Edit Page** link on the Developer toolbar.

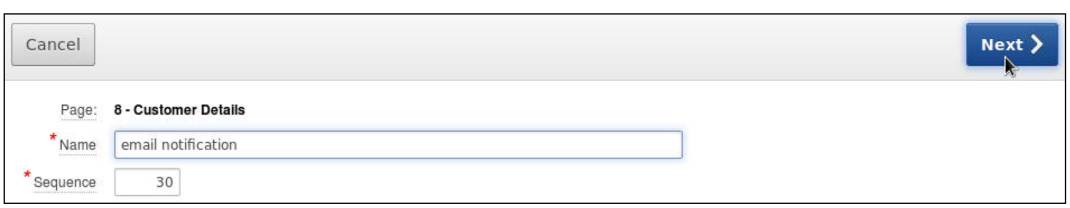


4) Add the Notification Dynamic Action Plug-in to your Customer Details page.

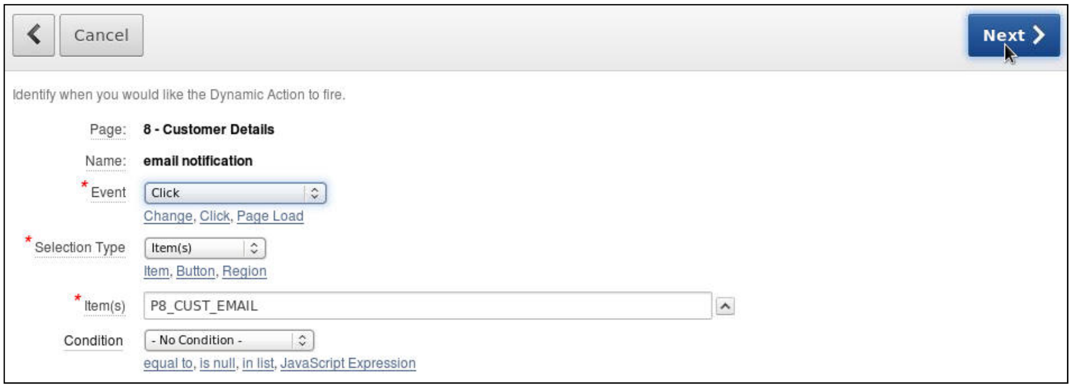
a) Right-click the **P<n>_EMAIL** page item and select **Create Dynamic Action**.



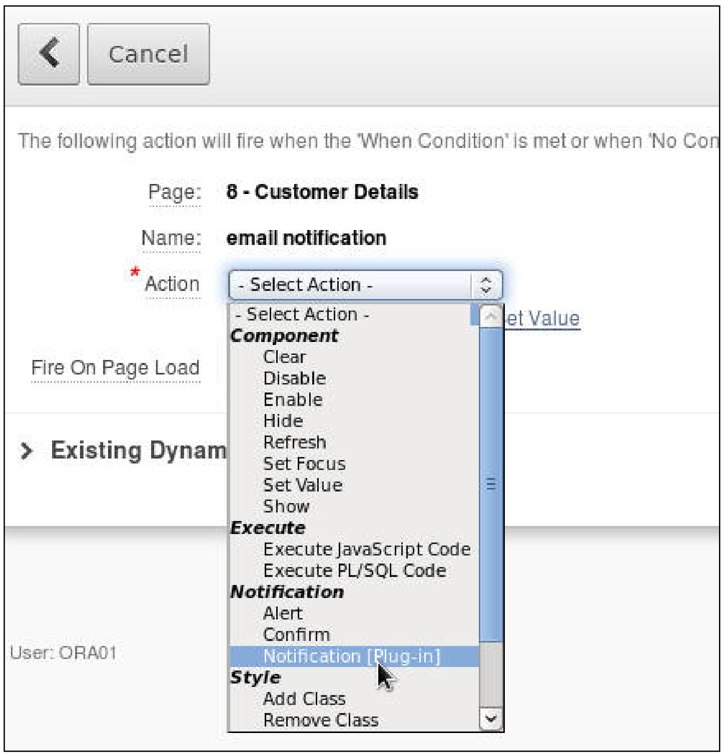
b) Enter **email notification** for Name and click **Next >**.



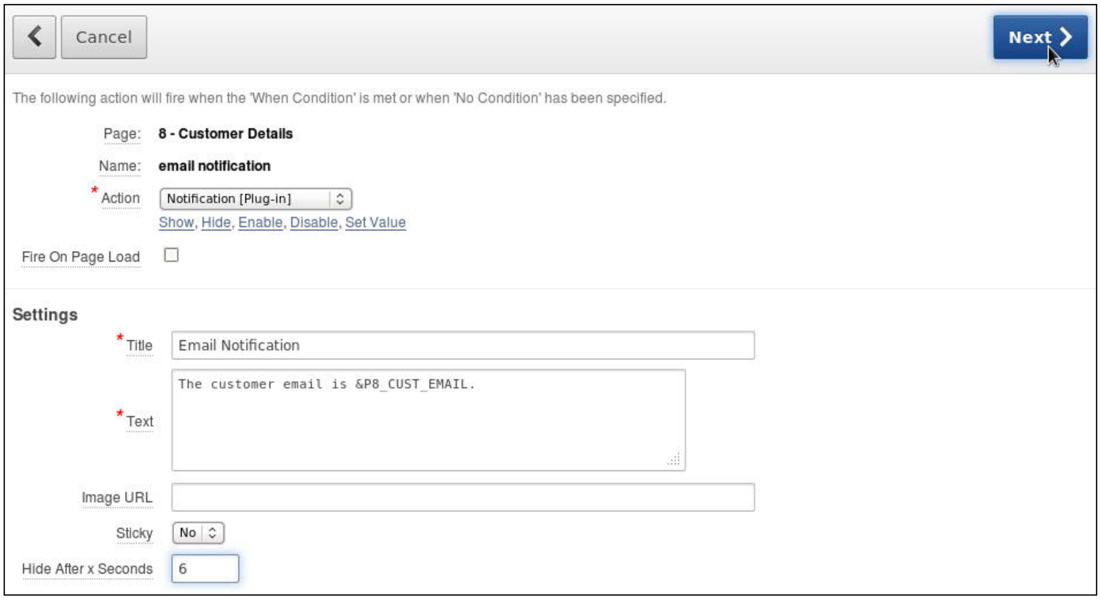
c) Select **Click** for Event, accept the rest of the defaults, and click **Next >**.



d) Select **Notification [Plug-In]** from the list of Actions.



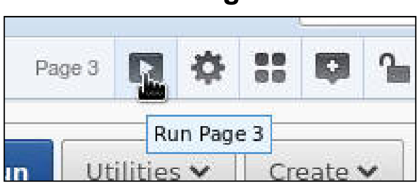
e) Enter **`Email Notification`** for Title and **`The customer email is &P<n>_CUST_EMAIL.`** for Text. You can control the fade in and fade out by specifying a number of seconds and whether to show a Sticky. Enter **6** for “Hide After x Seconds” and click **Next >**.



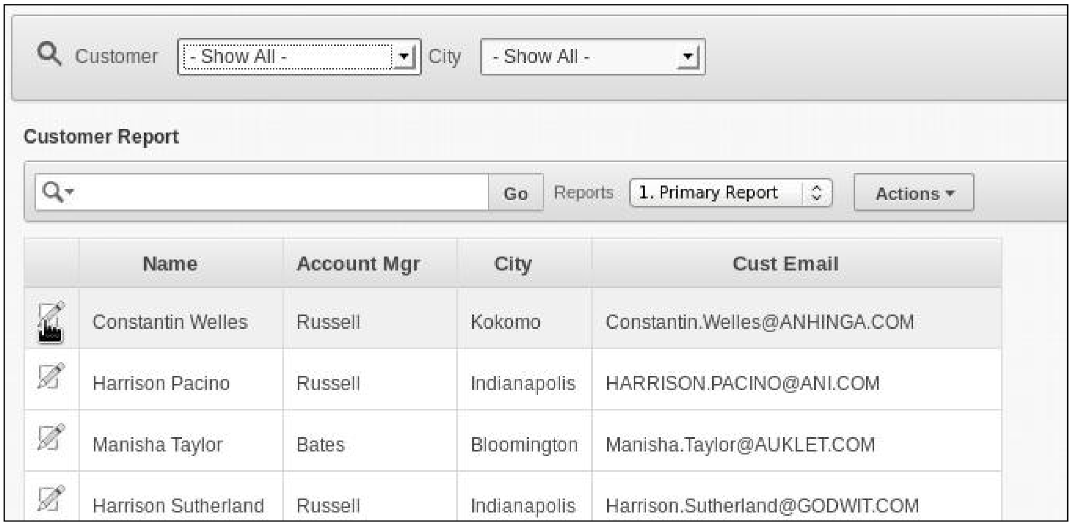
f) Click **Create Dynamic Action**.

g) You want to show the email for an existing customer, so you want to run the Customer report and edit a record from there. Enter **3** (or the Customers page) and click **Go**.

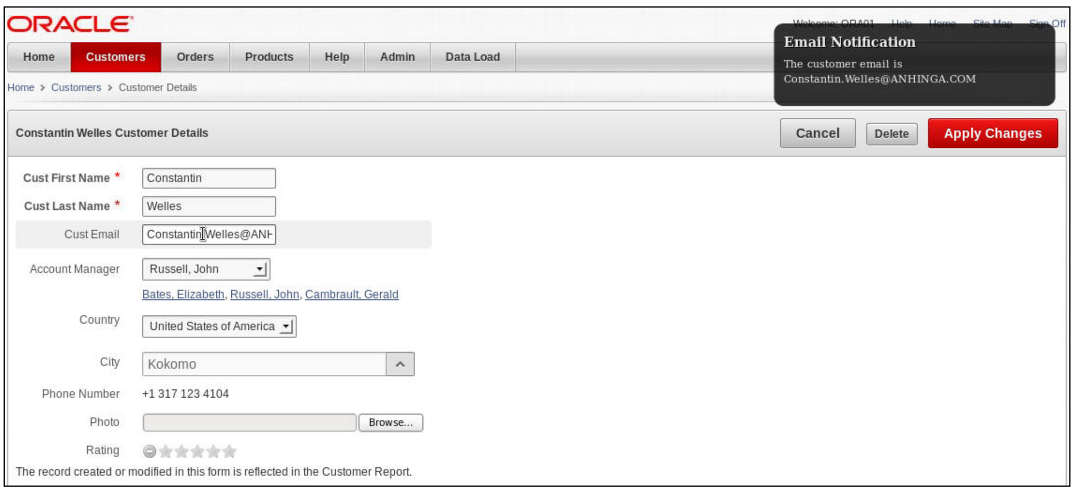
h) Click the **Run Page** icon.



i) Click the **Edit** icon next to one of the rows.



j) Click in the **Cust Email** item area. Notice the notification is displayed.



Practices for Lesson 20: Utilizing Application Express Printing

Chapter 20

Practices for Lesson 20: Overview

Practices Overview

In these practices, you print a standard report with derived output and create a PDF report with multiple queries.

Practice 20-1: Printing a Standard Report with Derived Output

Overview

In this practice, you enable printing of a custom report, creating an output format item, and then deriving the printing of the report based on the item using a button.

Assumptions

You have performed the previous practices.

Tasks

a. On the Top Tier Salary page, you edit Print Attributes for the Top Tier Salary report region and enable Report Printing. You run the page.
b. Change the report to be printed based on a derived format value. Create a select list item and specify a static list of values on the item. Then create a button called Print to run the report when an Output Format is selected.
c. Update the Print Attributes for the region to derive Output Format from the item. Create a branch and add a condition for the application to know what to do when the Print button is clicked.

Practice 20-2: Creating a PDF Report with Multiple Queries

Overview

In this practice, you create a report query that contains two queries that uses an existing report layout. You then create a report that uses the report query you created and invoke it from a button.

Assumptions

You have performed the previous practices.

Tasks

a. Create a report query called **multiquery** with the following SQL (located in the `/home/oracle/labs/files/lab20_1.txt` file):

```
select product_id, product_name, product_status, list_price from
oehr_product_information
```

b. Add another query to the report with the following SQL (located in the `/home/oracle/labs/files/lab20_2.txt` file):

```
select task_name, start_date, end_date, status from tasks
```

c. Use the RTF file called `multiquery.rtf` (located in **`/home/oracle/labs/files`**) for the report layout. Test the report.

d. Create a button called Print Report on the Home page and run the page.

Solution 20-1: Printing a Standard Report with Derived Output

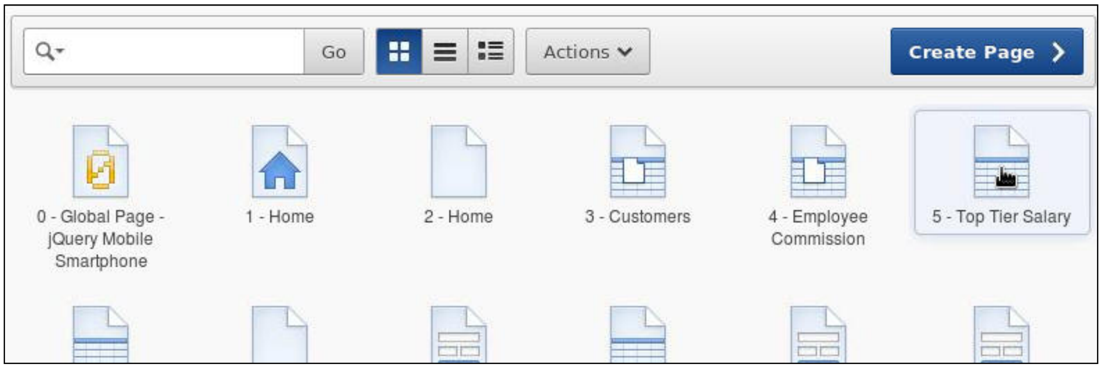
Overview

In this solution, you enable printing of a custom report, creating an output format item, and then deriving the printing of the report based on the item using a button.

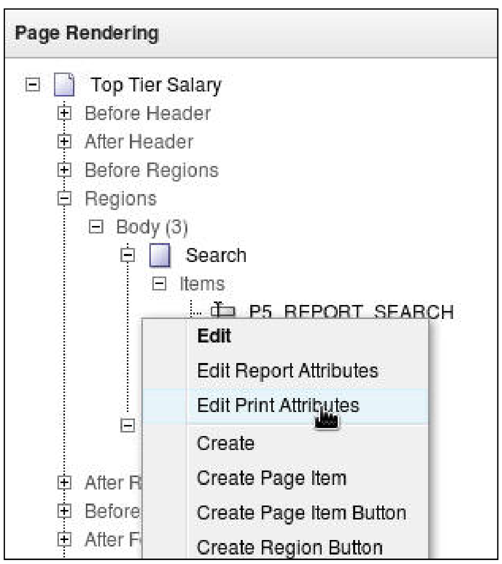
Tasks

a. On the Top Tier Salary page, you edit Print Attributes for the Top Tier Salary report region and enable Report Printing. You run the page.

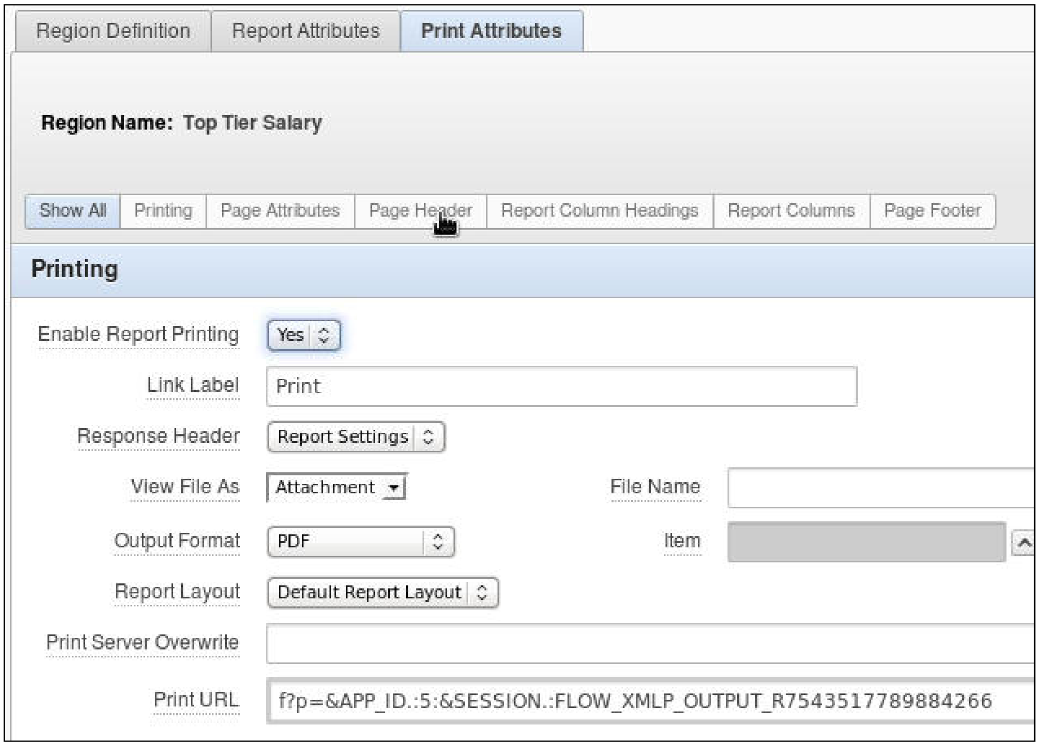
1) On the Order Management application home page, click the **Top Tier Salary** page.



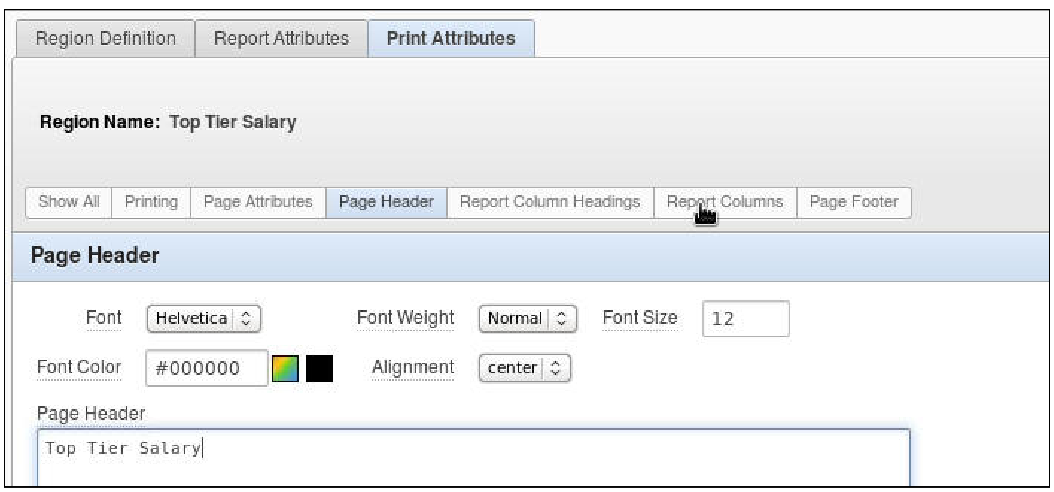
2) On the Top Tier Salary page, right-click **Top Tier Salary** report region and select **Edit Print Attributes**.



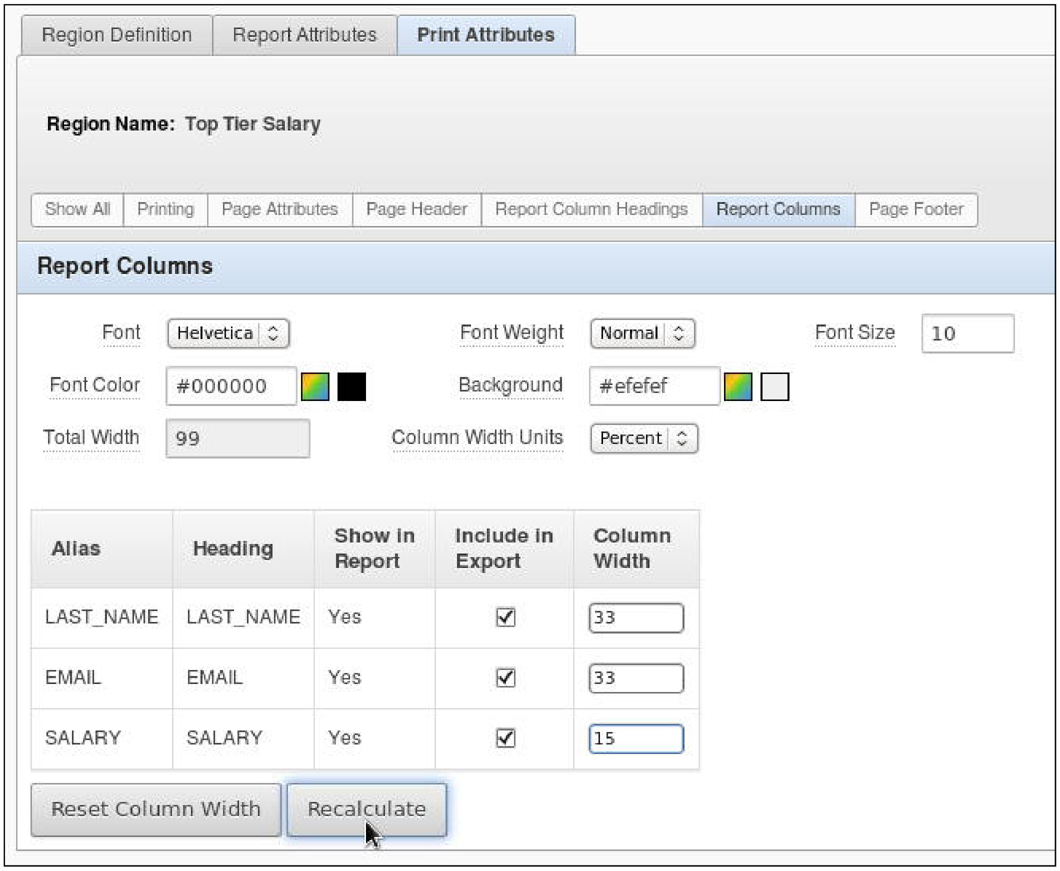
3) Select **Yes** for Enable Report Printing, and then click the **Page Header** subtab.



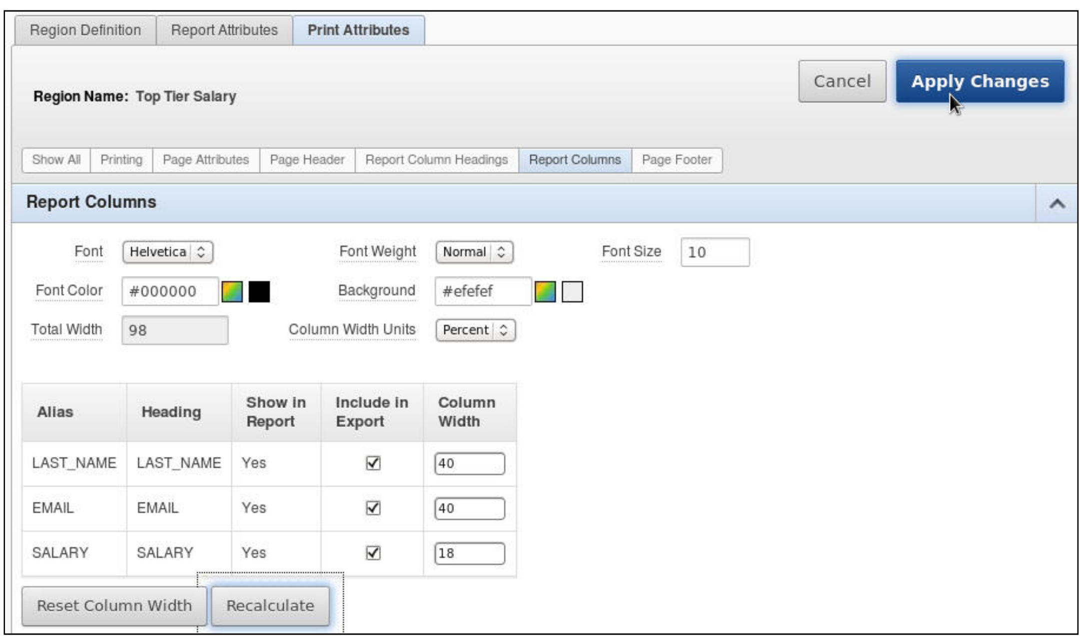
4) Enter `Top Tier Salary` for Page Header and click **Report Columns**.



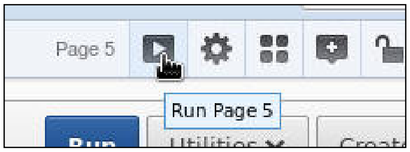
5) Change Column Width for Salary to **15** and click **Recalculate**.



6) Notice that because you had Column Width Units set to Percent, the calculation changed based on the percentage rather than pixels. This is why it now says 18. Click **Apply Changes**.



7) Run the page.



8) View the PDF report by using the **Print** Link at the bottom of the report. Notice that the header is shown and the column widths are proportioned to what you set them to in the definition.

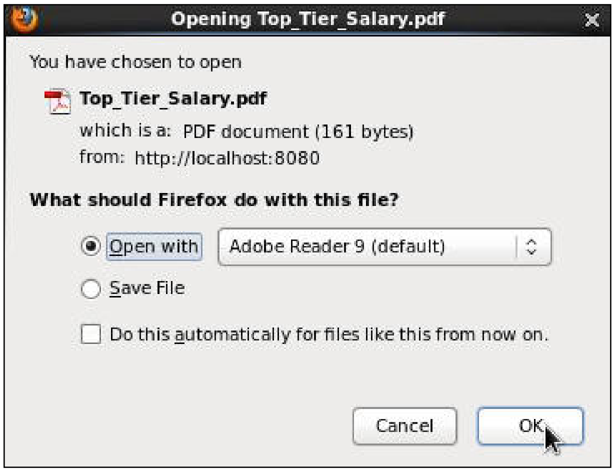

Top Tier Salary

LAST_NAME	EMAIL	SALARY
Hunold	AHUNOLD	9000
Ernst	BERNST	10000
Greenberg	NGREENBE	12000
Faviet	DFAVIET	9000
Chen	JCHEN	8200
Sciarra	ISCIARRA	7700
Urman	JMURMAN	7800
Popp	LPOPP	6900
Raphaely	DRAPHEAL	11000
Weiss	MWEISS	8000
Fripp	AFRIPP	8200
Kaufling	PKAUFLIN	7900
Vollman	SVOLLMAN	6500
Mourgos	KMOURGOS	5800
Errazuriz	AERRAZUR	12000

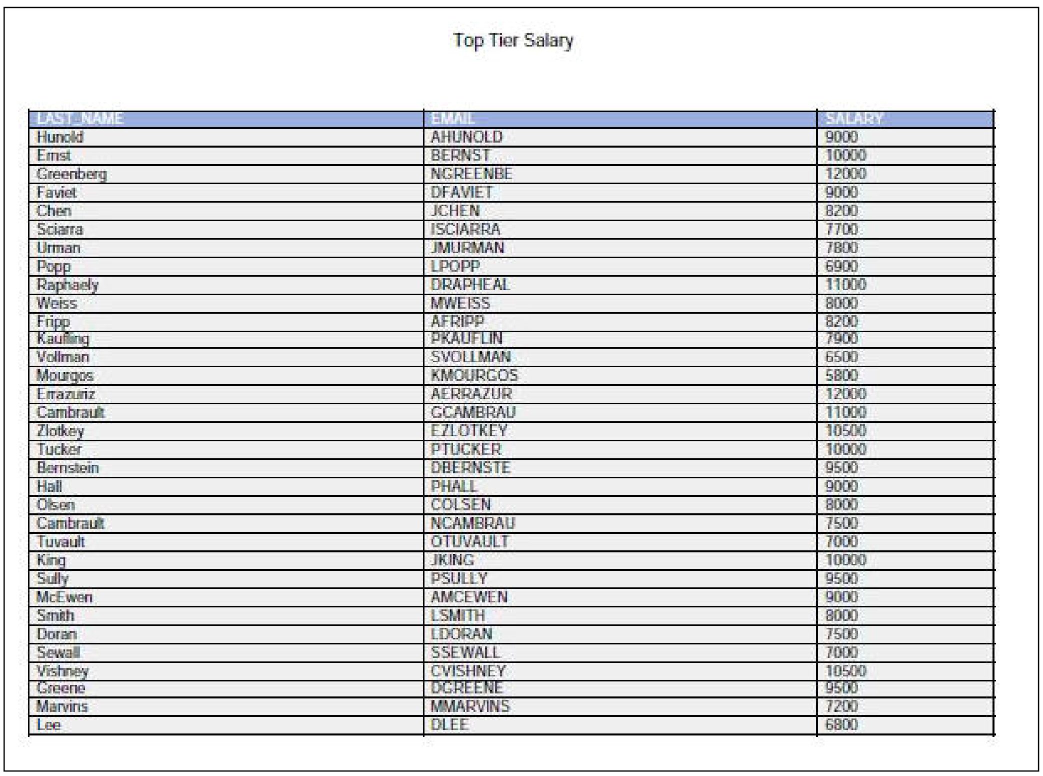
row(s) 1 - 15 of 52 Next >

Print

9) Click **OK**.



10) The PDF is displayed.



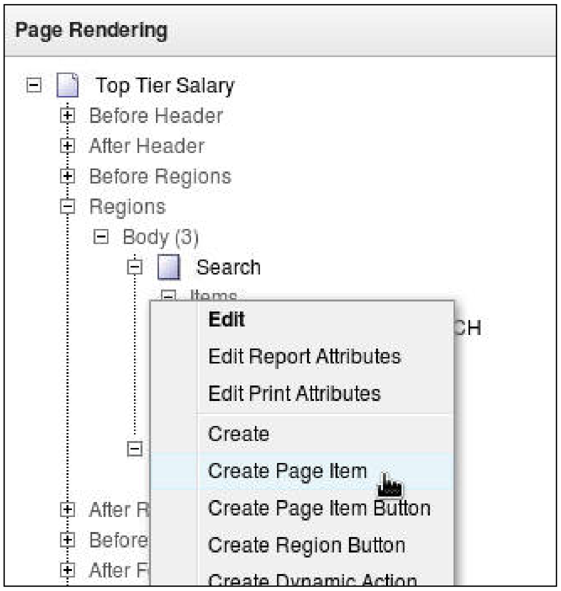
Top Tier Salary

LAST_NAME	EMAIL	SALARY
Hunold	AHUNOLD	9000
Ernst	BERNST	10000
Greenberg	NGREENBE	12000
Faviet	DFAVIET	9000
Chen	JCHEN	8200
Sciarra	ISCIARRA	7700
Urman	JMURMAN	7800
Popp	LPOPP	6900
Raphaely	DRAPHEAL	11000
Weiss	MWEISS	8000
Fripp	AFRIPP	8200
Kaufling	PKAUFLIN	7900
Vollman	SVOLLMAN	6500
Mourgos	KMOURGOS	5800
Errazuriz	AERRAZUR	12000
Cambrault	GCAMBRAU	11000
Zlotkey	EZLOTKEY	10500
Tucker	PTUCKER	10000
Bernstein	DBERNSTE	9500
Hall	PHALL	9000
Olsen	COLSEN	8000
Cambrault	NCAMBRAU	7500
Tuvault	OTUVAULT	7000
King	JKING	10000
Sully	PSULLY	9500
McEwen	AMCEWEN	9000
Smith	LSMITH	8000
Doran	LDORAN	7500
Sewall	SSEWALL	7000
Vishney	CVISHNEY	10500
Greene	DGREENE	9500
Marvins	MMARVINS	7200
Lee	DLEE	6800

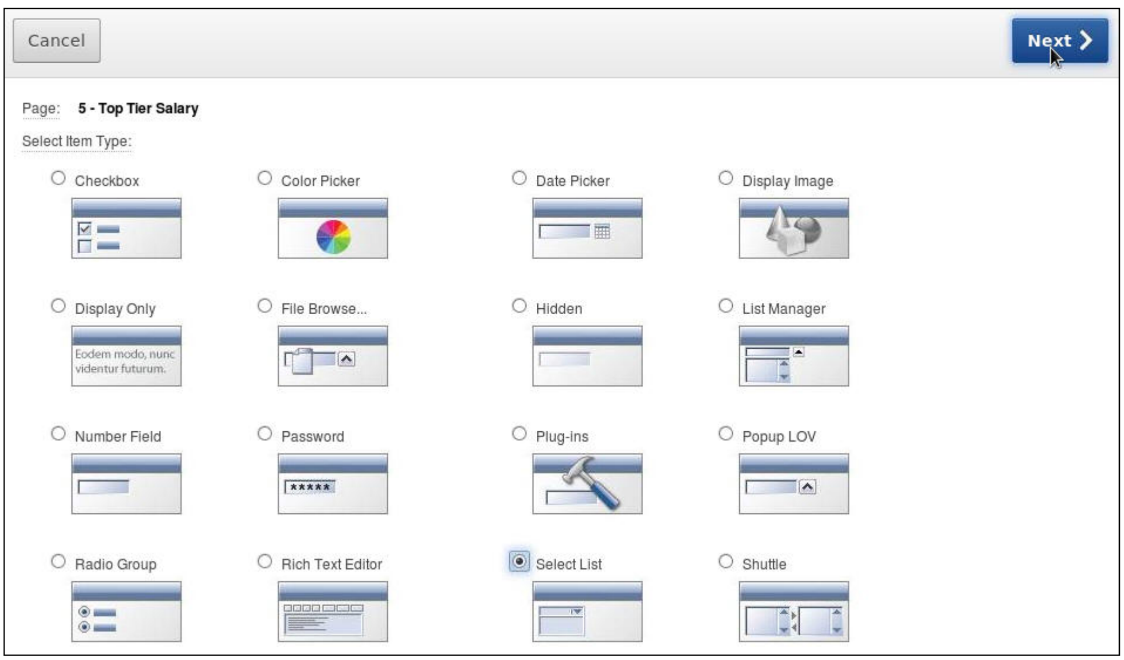
b. Change the report to be printed based on a derived format value. Create a select list item and specify a static list of values on the item. Then create a button called Print to run the report when an Output Format is selected.

1) Click **Edit Page <*n*>** on the Developer toolbar.

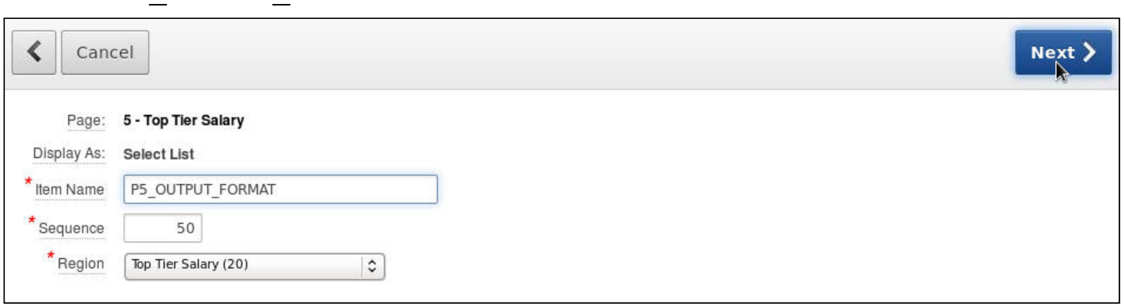
2) Under Regions, right-click **Top Tier Salary** and select **Create Page Item**.



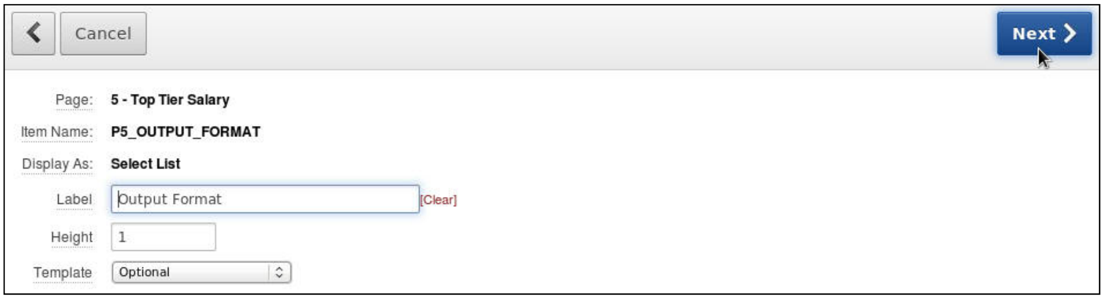
3) Click **Select List**.



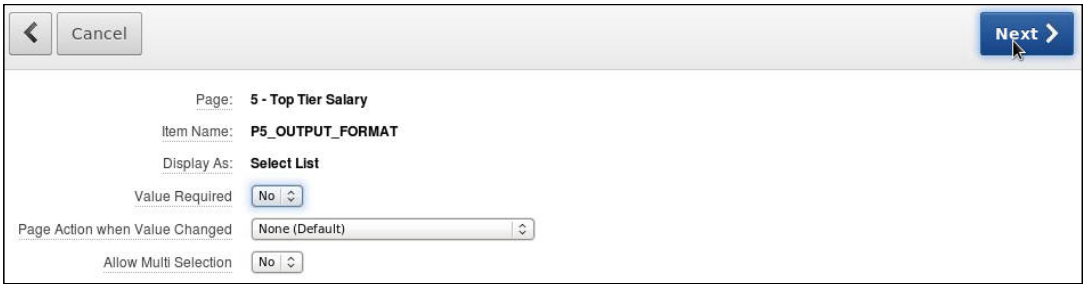
4) Enter `P<n>_OUTPUT_FORMAT` and click **Next >**.



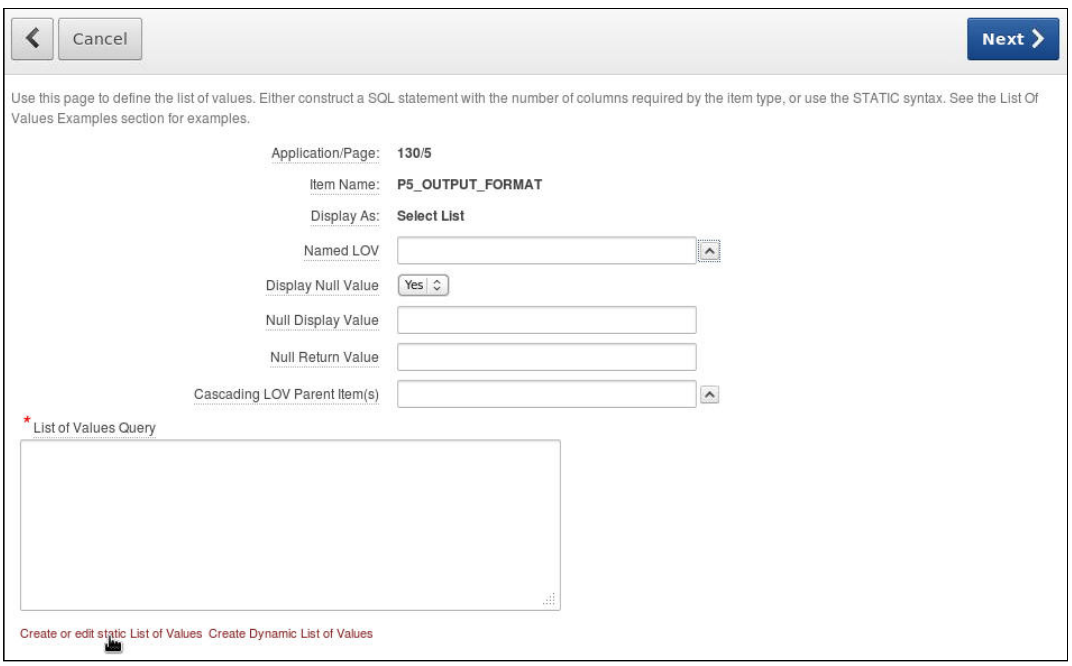
5) Click **Next >**.



6) Click **Next >**.



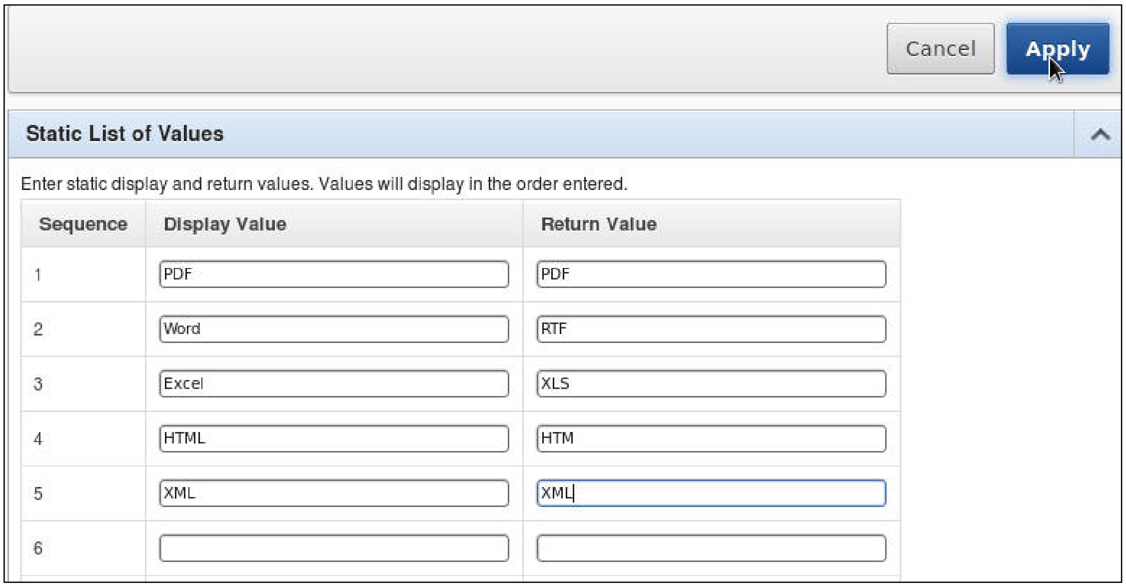
7) Click **Create or edit static List of Values**.



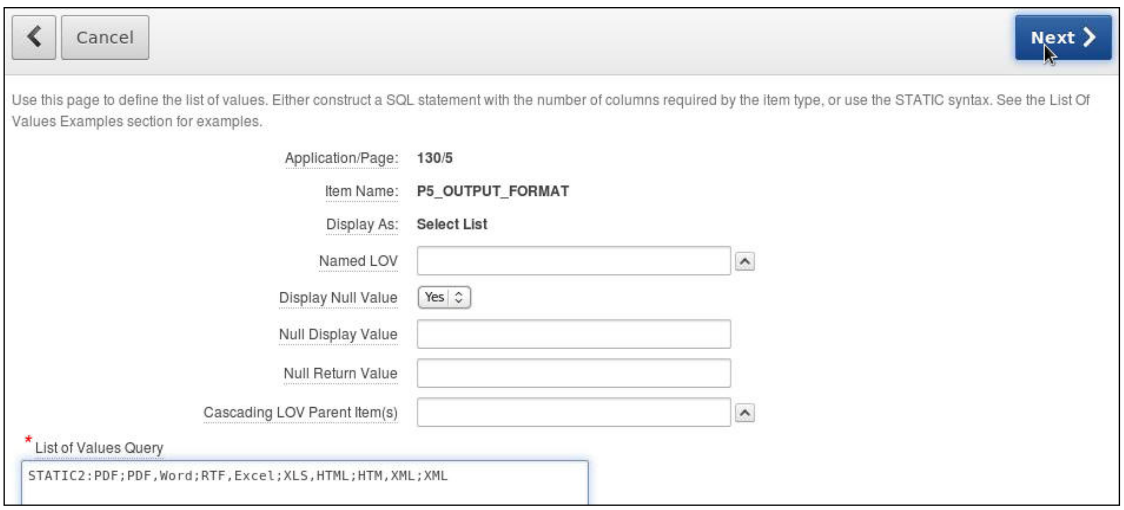
8) Specify a static list of values on the item with the following values (no null values):

Display Value	Return Value
PDF	PDF
Word	RTF
Excel	XLS
HTML	HTM
XML	XML

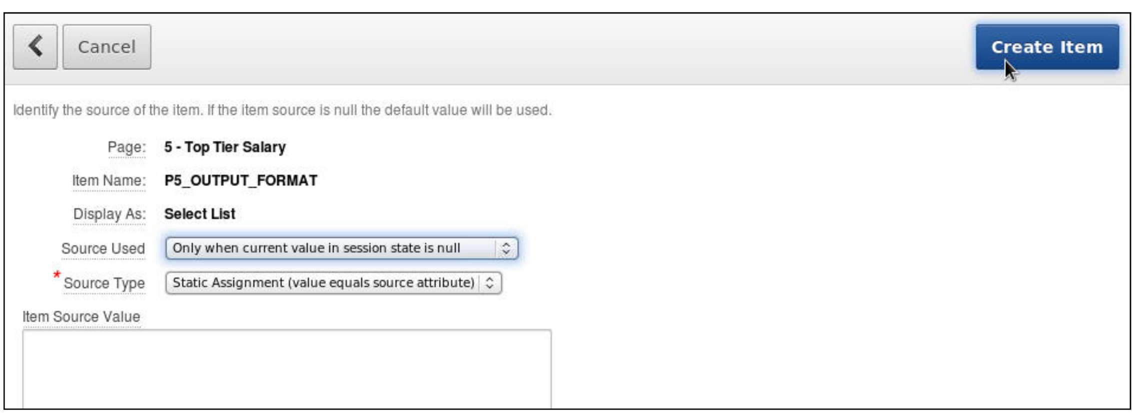
Use the table above to enter values for Display Value and Return Value. Click **Apply**.



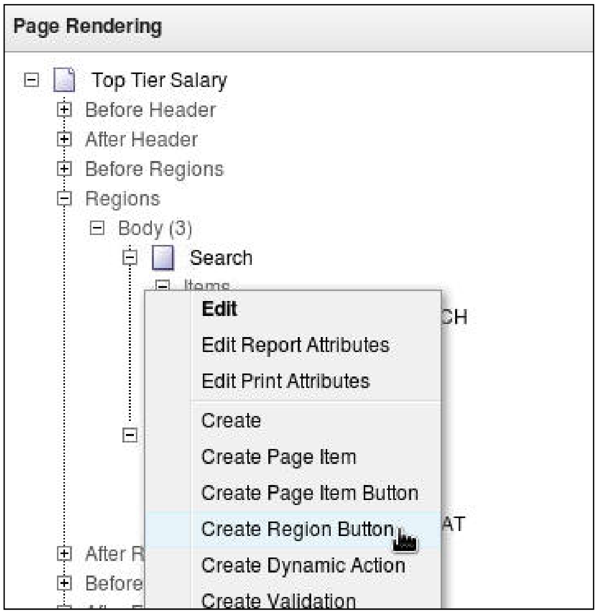
9) Click **Next >**.



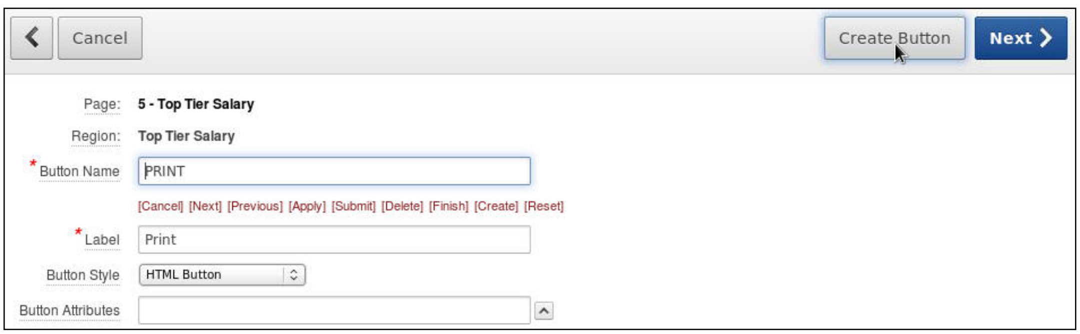
10) Click **Create Item**.



11) Under Regions, right-click **Top Tier Salary** and select **Create Region Button**.

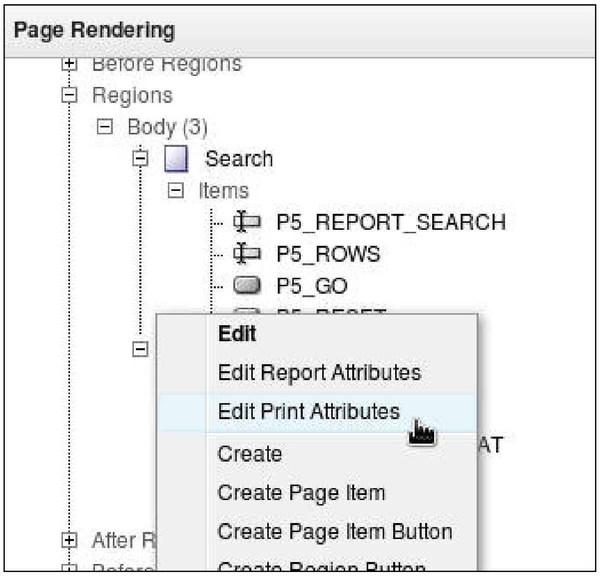


12) For Button Name, enter `PRINT`. Enter `Print` for Label. You want to accept all the other defaults and click **Create Button**.

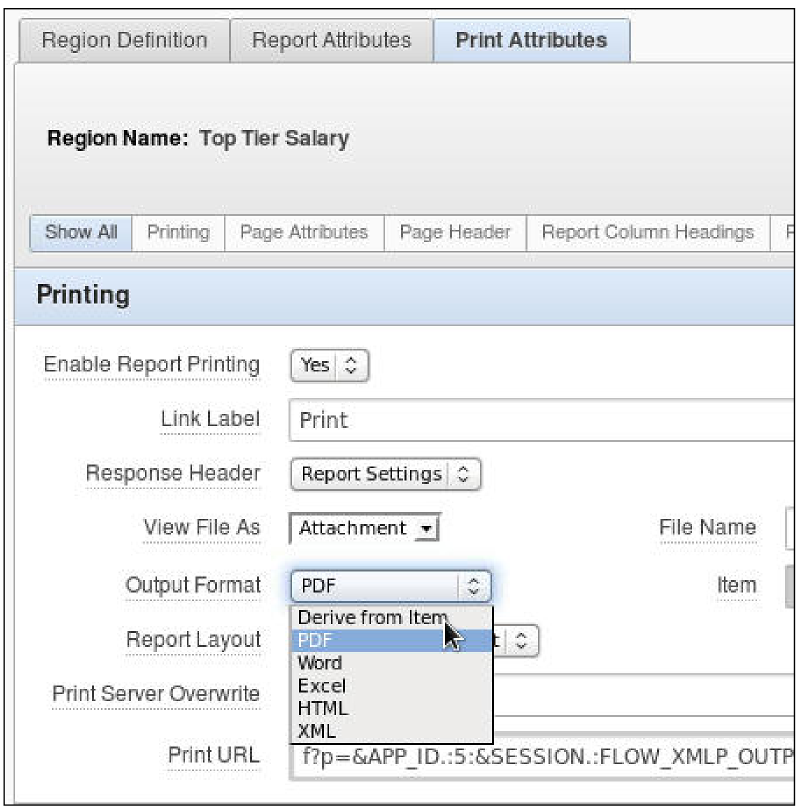


c. Update the Print Attributes for the region to derive Output Format from the item. Create a branch and add a condition for the application to know what to do when the Print button is clicked.

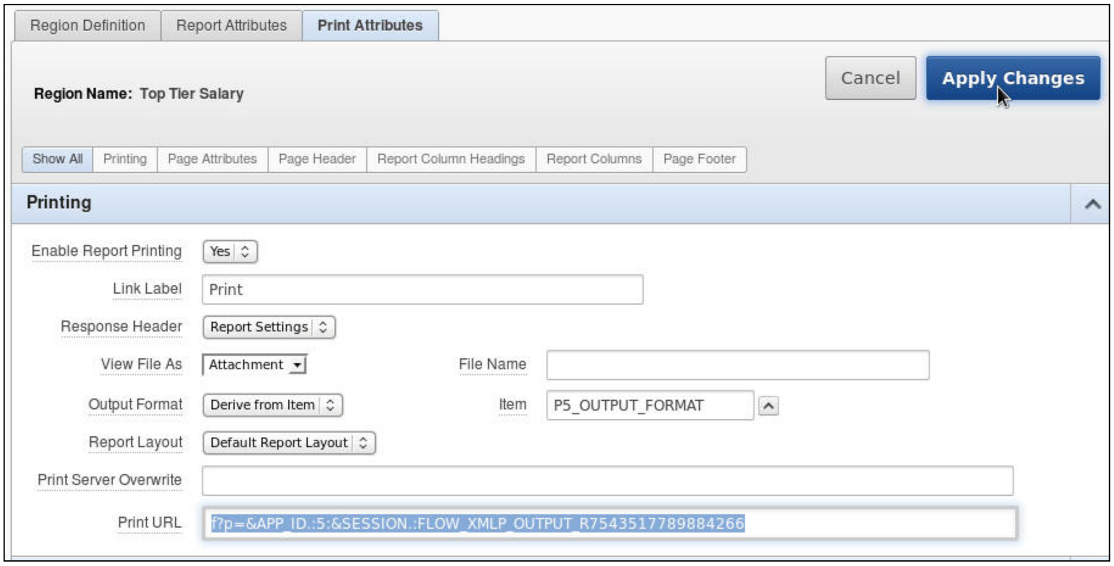
1) Right-click **Top Tier Salary** region and select **Edit Print Attributes**.



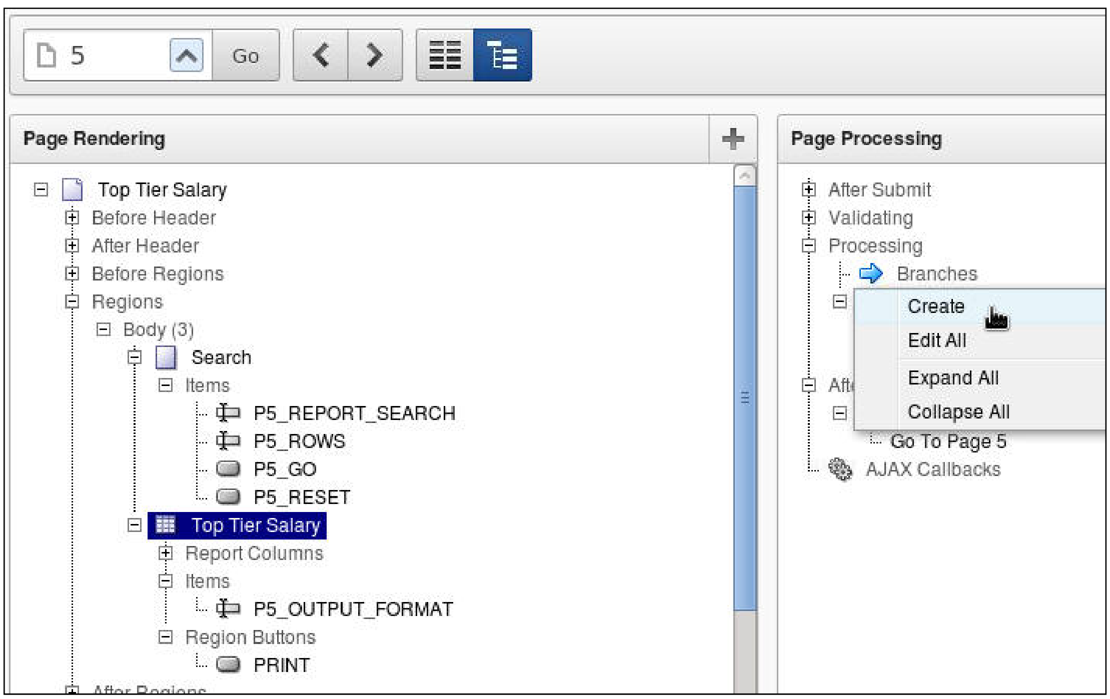
2) For Output Format, select **Derive from Item.**



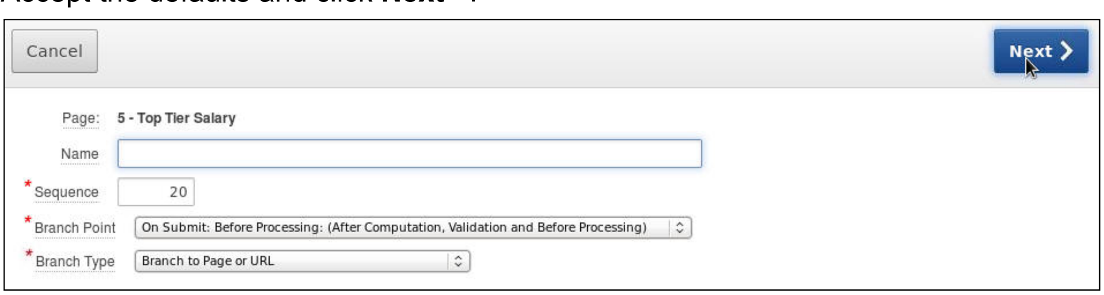
3) For Item, select `P<n>_OUTPUT_FORMAT`. Copy the **`FLOW_XMLP_OUTPUT_R<nnn>`** value to the Clipboard, and click **Apply Changes**.



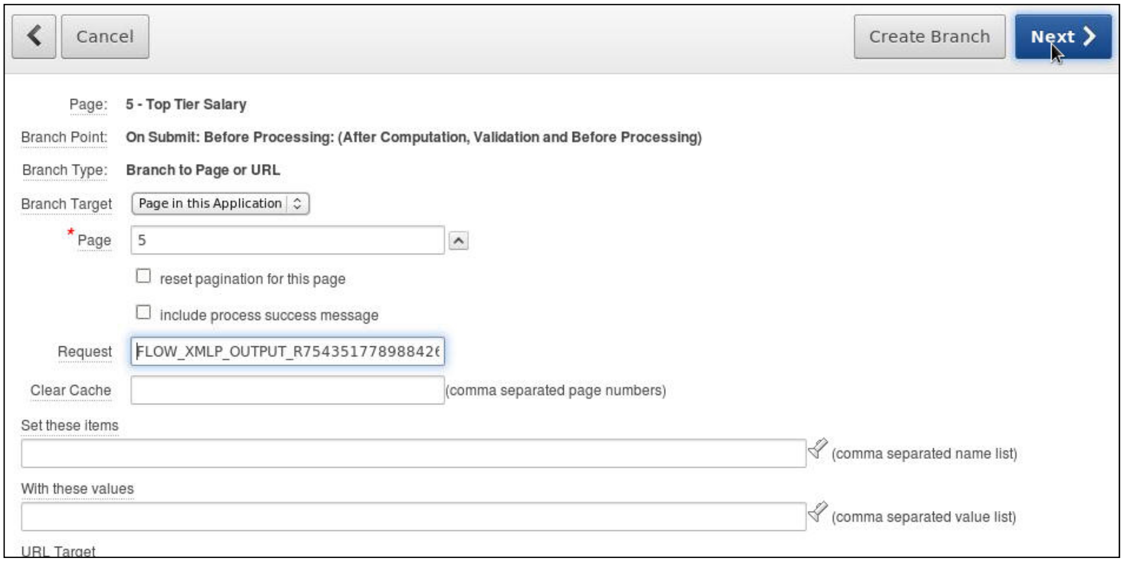
4) Under Page Processing, right-click **Branches** and select **Create**.



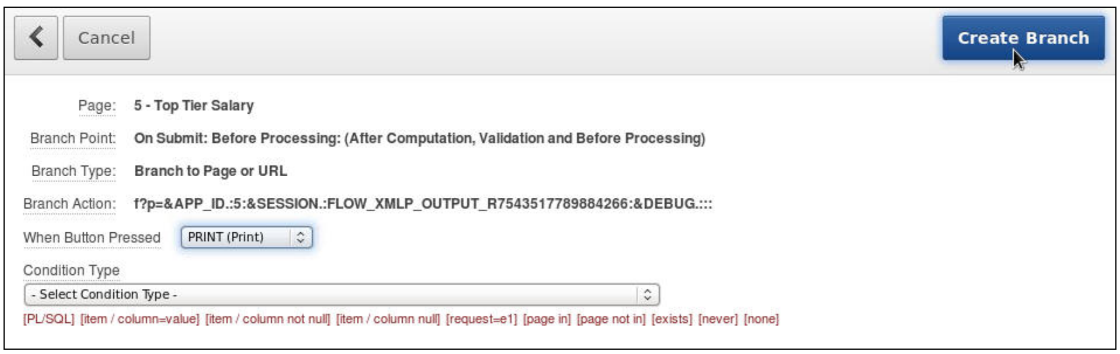
5) Accept the defaults and click **Next >**.



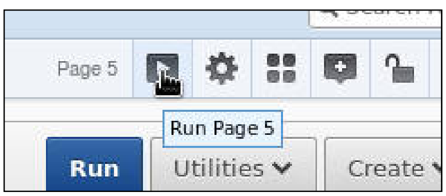
6) Select the **Top Tier Salary** page number for Page and enter the `FLOW_XMLP_OUTPUT_R<nnn>` value in the Request field. Click **Next >**.



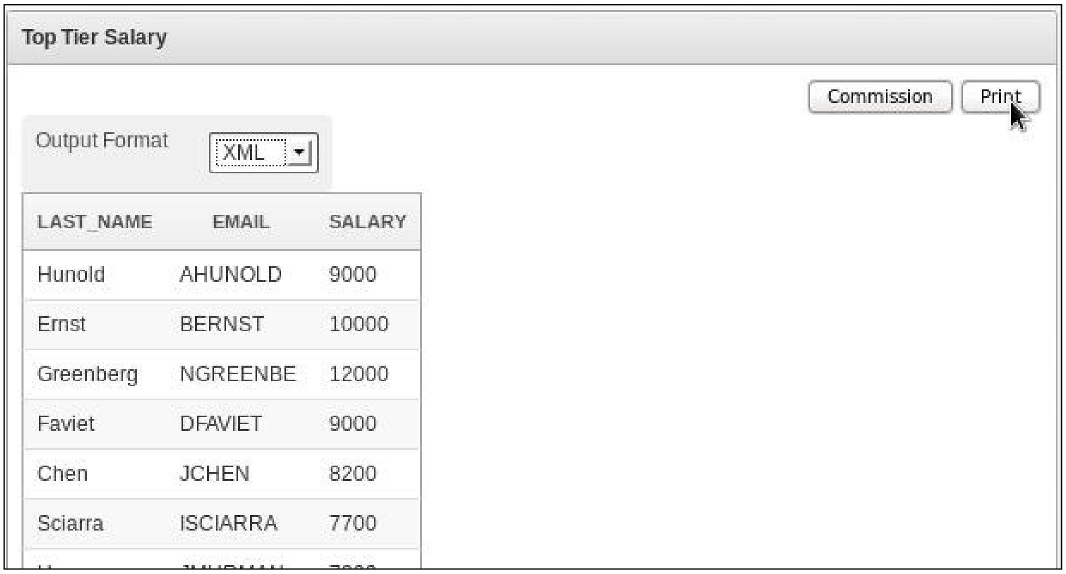
7) Select **PRINT (Print)** for When Button Pressed and click **Create Branch**.



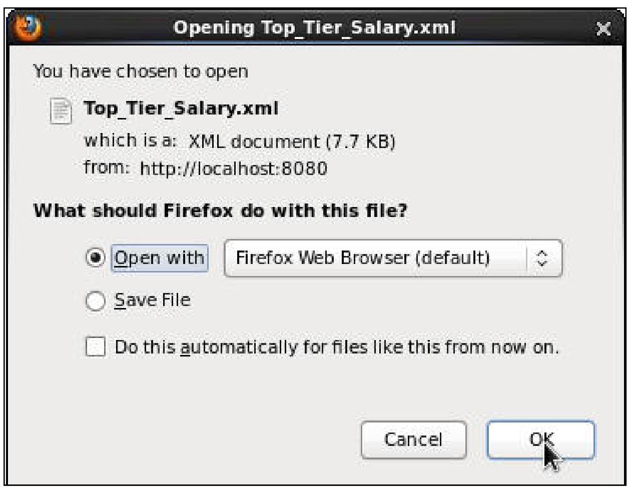
8) Click the **Run Page** icon.



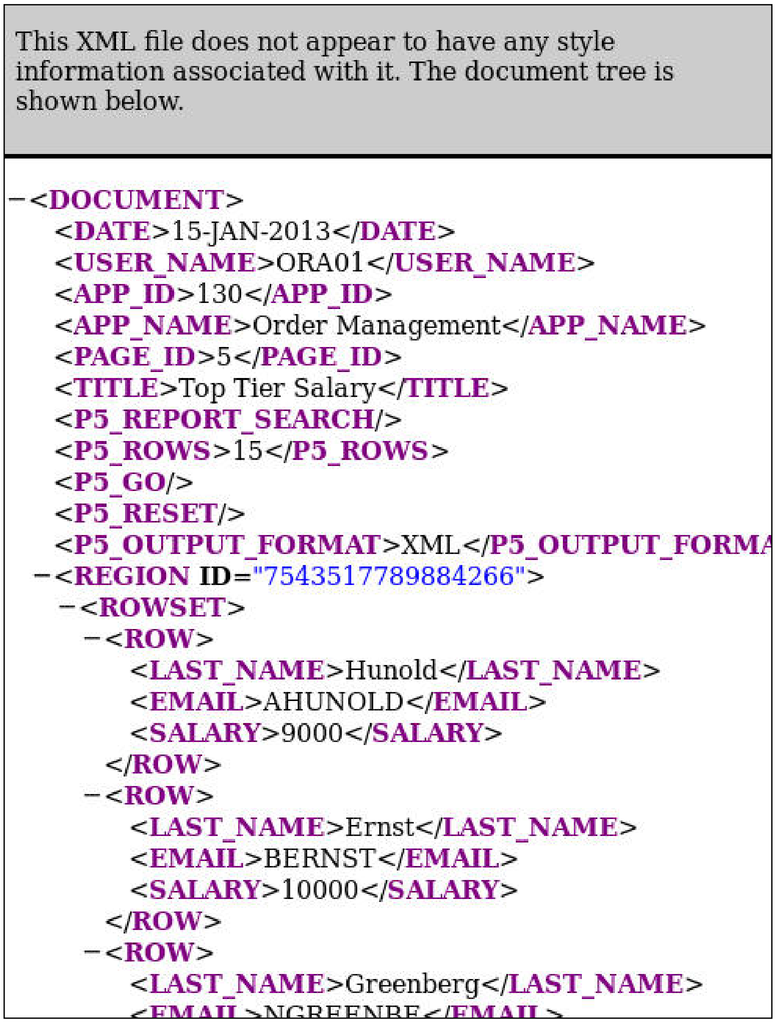
9) Select **XML** for Output Format and click **Print**.



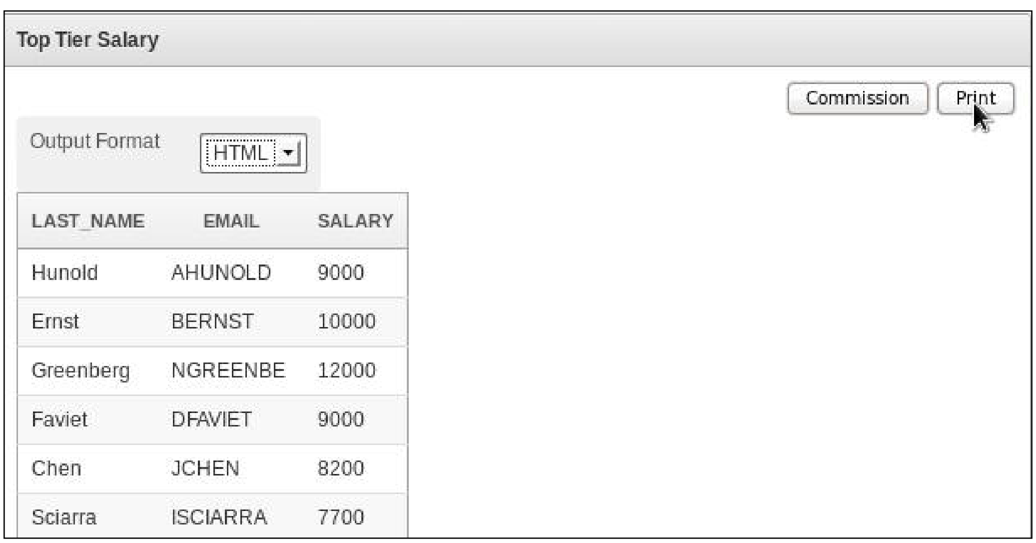
10) Click **OK**.



11) The XML is displayed.

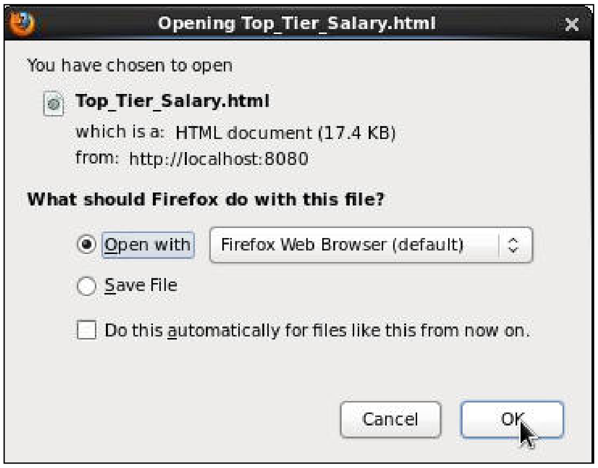


12) Select **HTML** for Output Format and click **Print**.



LAST_NAME	EMAIL	SALARY
Hunold	AHUNOLD	9000
Ernst	BERNST	10000
Greenberg	NGREENBE	12000
Faviet	DFAVIET	9000
Chen	JCHEN	8200
Sciarra	ISCIARRA	7700

13) Click **OK**.



14) The HTML report is displayed.

Top Tier Salary

LAST_NAME	EMAIL	SALARY
Hunold	AHUNOLD	9000
Ernst	BERNST	10000
Greenberg	NGREENBE	12000
Faviet	DFAVIET	9000
Chen	JCHEN	8200
Sciarra	ISCIARRA	7700
Urman	JMURMAN	7800
Popp	LPOPP	6900
Raphaely	DRAPHEAL	11000
Weiss	MWEISS	8000
Fripp	AFRIPP	8200
Kaufling	PKAUFLIN	7900
Vollman	SVOLLMAN	6500
Mourgos	KMOURGOS	5800
Errazuriz	AERRAZUR	12000
Cambrault	GCAMBRAU	11000
Zlotkey	EZLOTKEY	10500
Tucker	PTUCKER	10000
Bernstein	DBERNSTE	9500
Hall	PHALL	9000
Olsen	COLSEN	8000
Cambrault	NCAMBRAU	7500
Tuvault	OTUVAULT	7000
King	JKING	10000
Sully	PSULLY	9500
McEwen	AMCEWEN	9000
Smith	LSMITH	8000
Doran	LDORAN	7500
Sewall	SSEWALL	7000
Vishney	CVISHNEY	10500

Solution 20-2: Creating a PDF Report with Multiple Queries

Overview

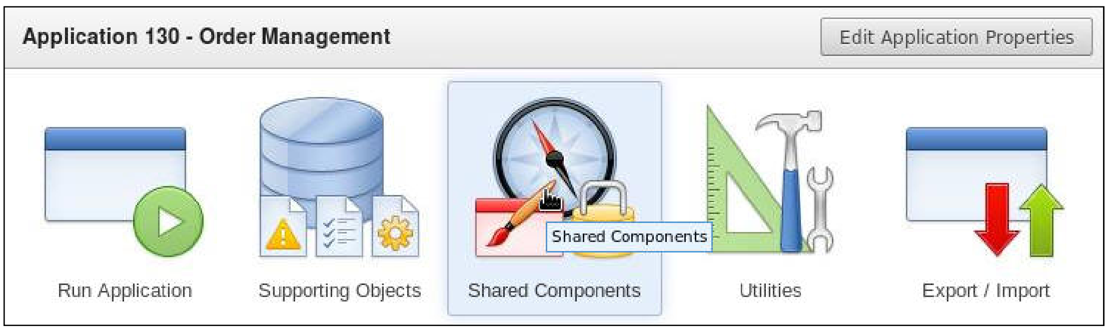
In this practice, you create a report query that contains two queries that uses an existing report layout. You then create a report that uses the report query you created and invoke it from a button.

Tasks

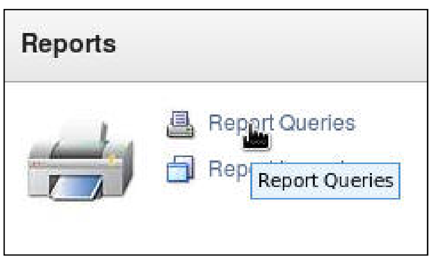
a. Create a report query called **multiquery** with the following SQL (located in `/home/oracle/labs/files/lab20_1.txt` file):

```
select product_id, product_name, product_status, list_price from
oehr_product_information
```

1) On the Order Management application home page, click **Shared Components**.

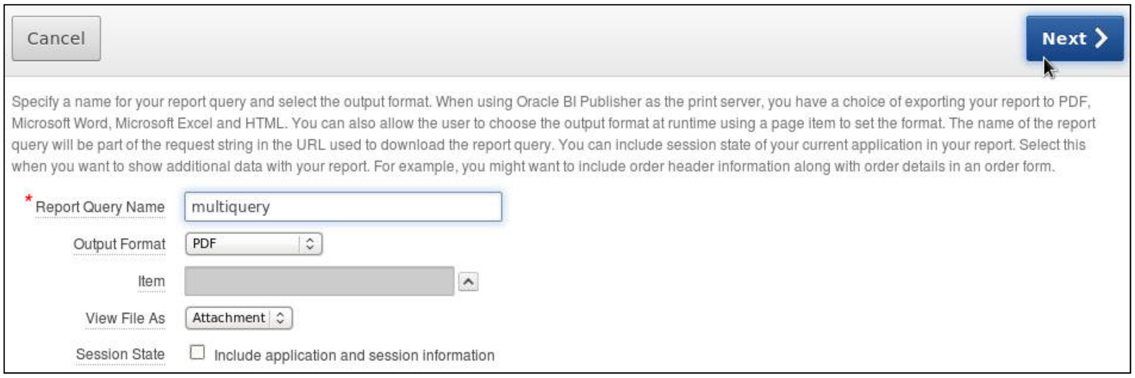


2) Under Reports, select **Report Queries**.



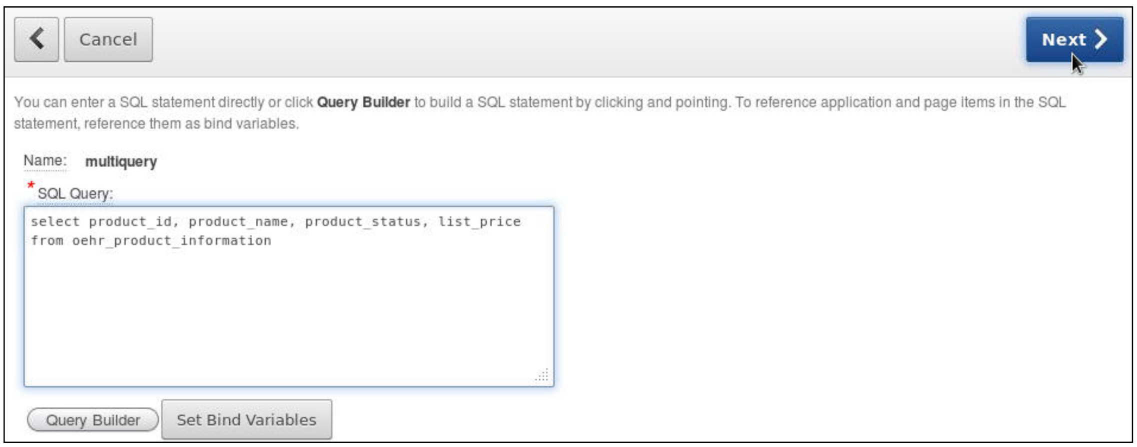
3) Click **Create**.

4) Enter **multiquery** for Report Query Name, and click **Next >**.



5) Enter the following SQL statement (located in the `/home/oracle/labs/files/lab20_1.txt` file), and click **Next >**:

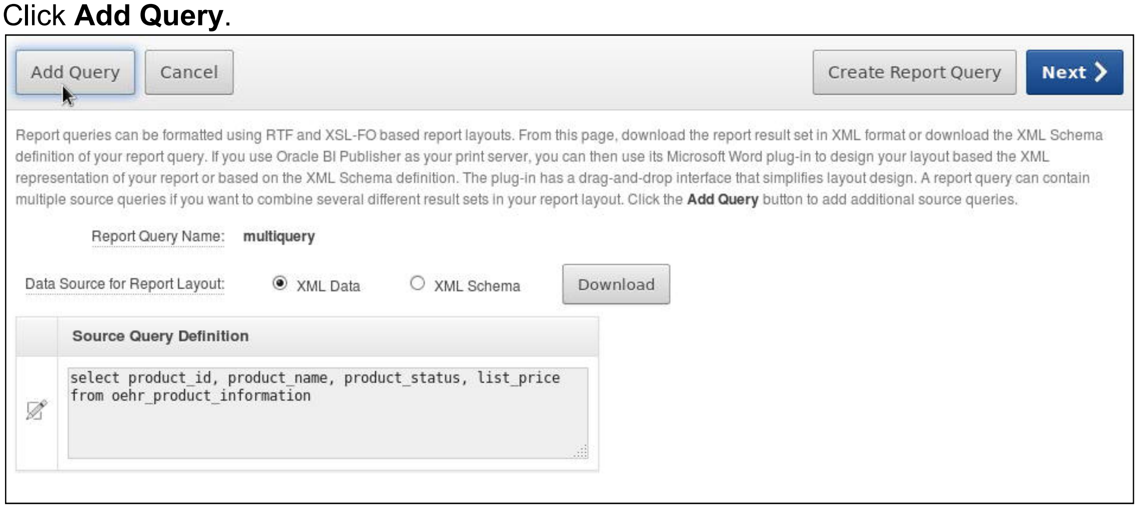
```
select product_id, product_name, product_status, list_price from
oehr_product_information
```



b. Add another query to the report with the following SQL (located in the `/home/oracle/labs/files/lab20_2.txt` file):

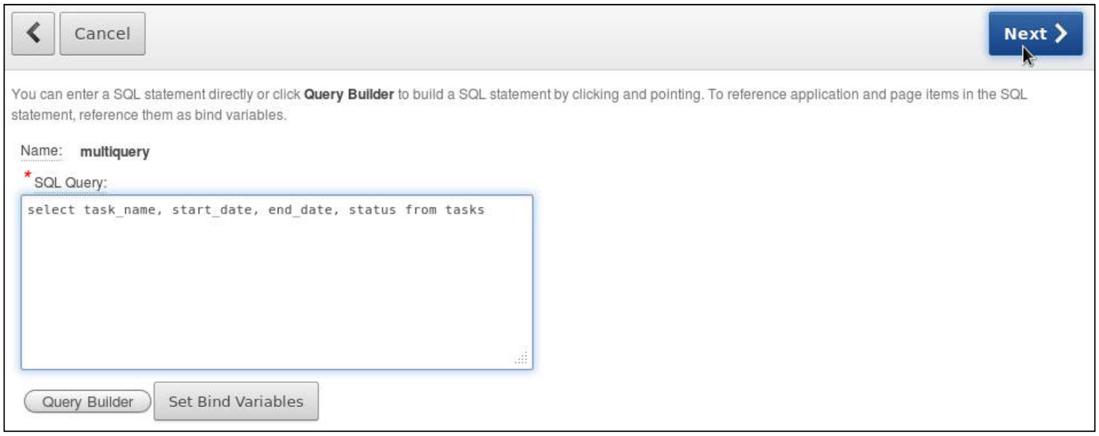
```
select task_name, start_date, end_date, status from tasks
```

1) Click **Add Query**.



2) Enter another query with the following SQL (located in the `/home/oracle/labs/files/lab20_2.txt` file). Click **Next >**.

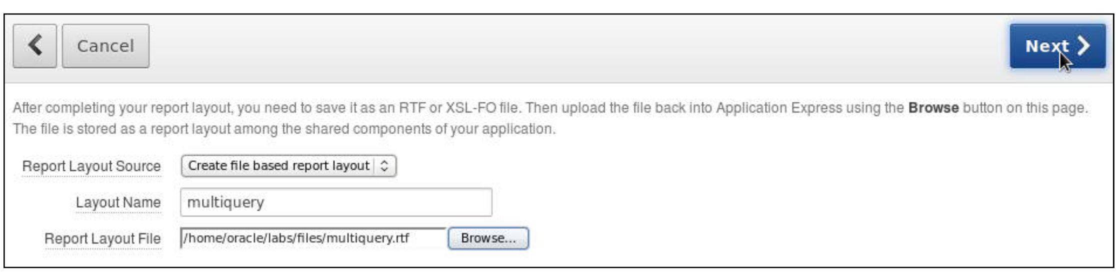
```
select task_name, start_date, end_date, status from tasks
```



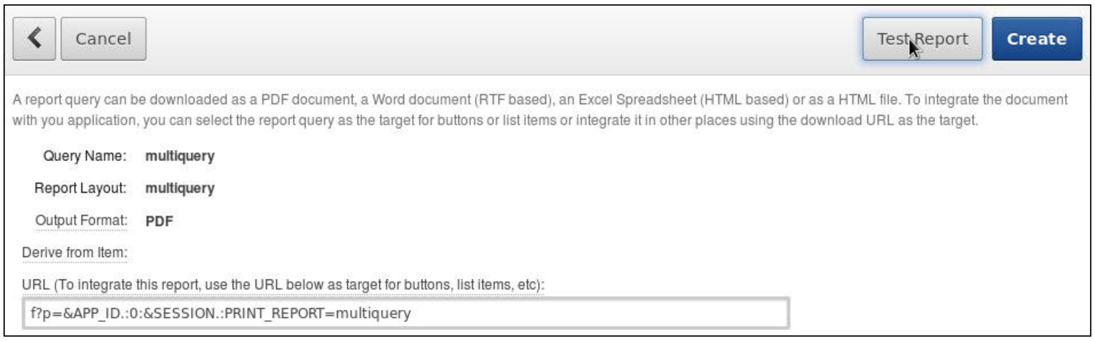
3) Click **Next >**.

c. Use the RTF file called `multiquery.rtf` (located in **`/home/oracle/labs/files`**) for the report layout. Test the report.

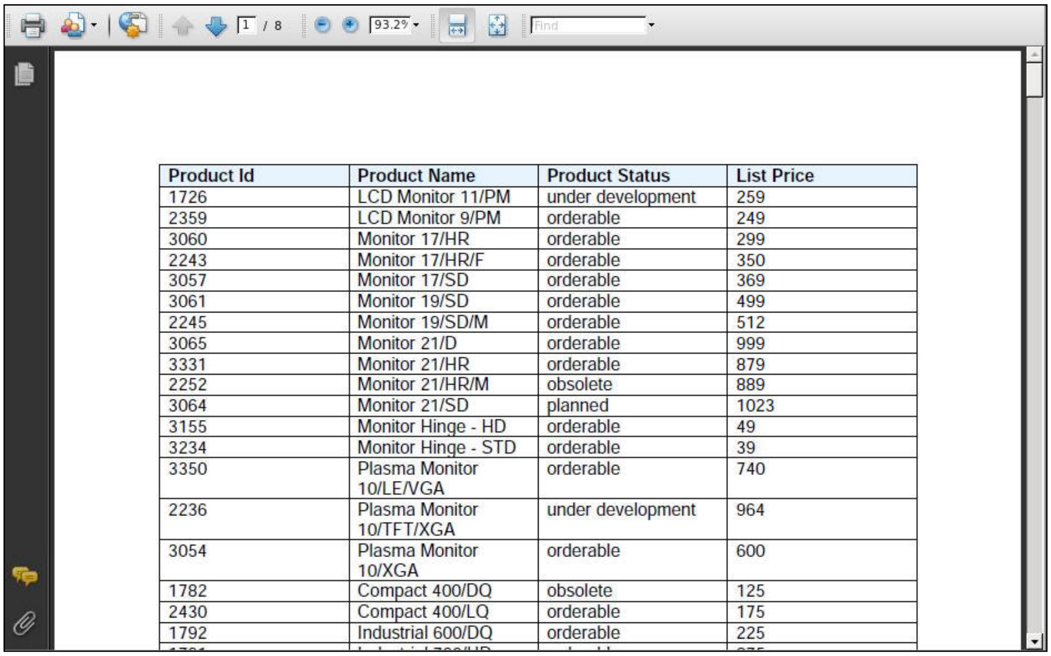
1) Click **Browse** and select the **`multiquery.rtf`** file (located in **`/home/oracle/labs/files`**) and click **Open**. Click **Next >**.



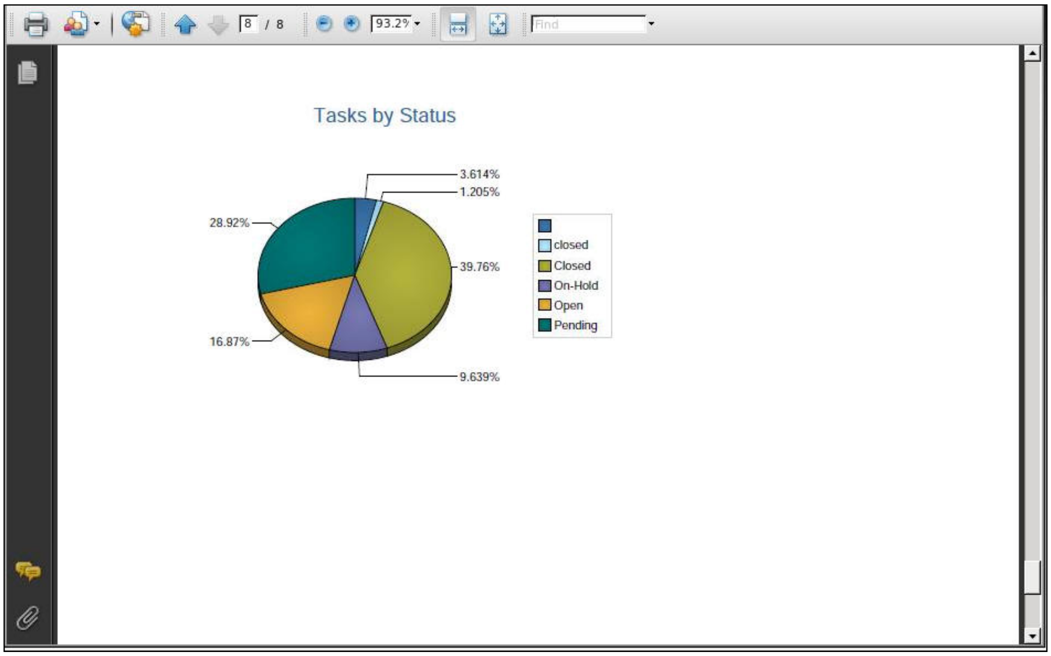
2) Click **Test Report** to make sure that the report runs successfully and that the data is shown. Close the window.



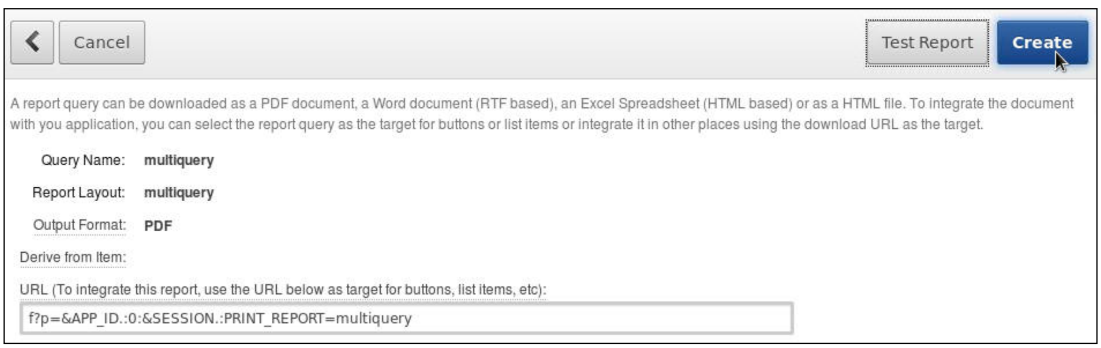
3) The report ran successfully and the data is shown. Scroll down to see the chart. When done, close the window.



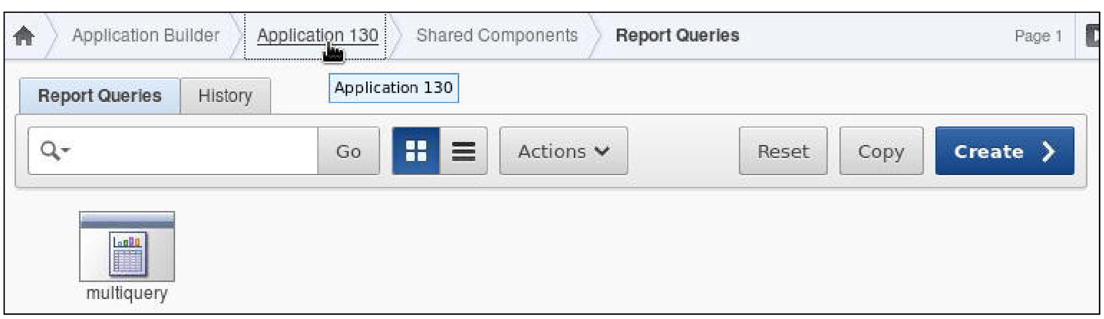
Product Id	Product Name	Product Status	List Price
1726	LCD Monitor 11/PM	under development	259
2359	LCD Monitor 9/PM	orderable	249
3060	Monitor 17/HR	orderable	299
2243	Monitor 17/HR/F	orderable	350
3057	Monitor 17/SD	orderable	369
3061	Monitor 19/SD	orderable	499
2245	Monitor 19/SD/M	orderable	512
3065	Monitor 21/D	orderable	999
3331	Monitor 21/HR	orderable	879
2252	Monitor 21/HR/M	obsolete	889
3064	Monitor 21/SD	planned	1023
3155	Monitor Hinge - HD	orderable	49
3234	Monitor Hinge - STD	orderable	39
3350	Plasma Monitor 10/LE/VGA	orderable	740
2236	Plasma Monitor 10/TFT/XGA	under development	964
3054	Plasma Monitor 10/XGA	orderable	600
1782	Compact 400/DQ	obsolete	125
2430	Compact 400/LQ	orderable	175
1792	Industrial 600/DQ	orderable	225



4) Click **Create**.

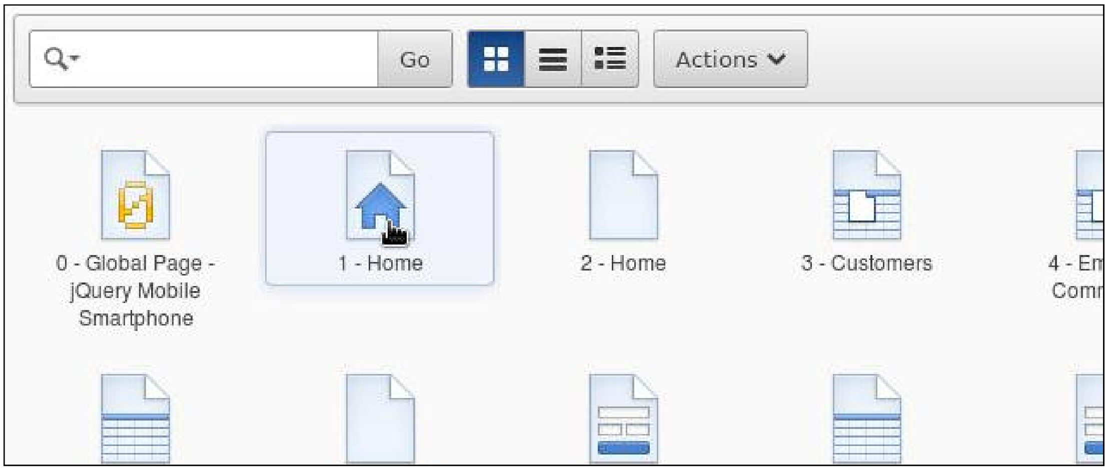


5) You have successfully created the report queries and layout. Click the Application breadcrumb.

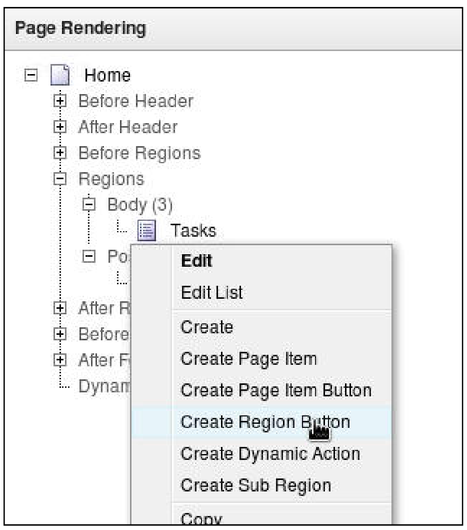


d. Create a button called Print Report on the Home page and run the page.

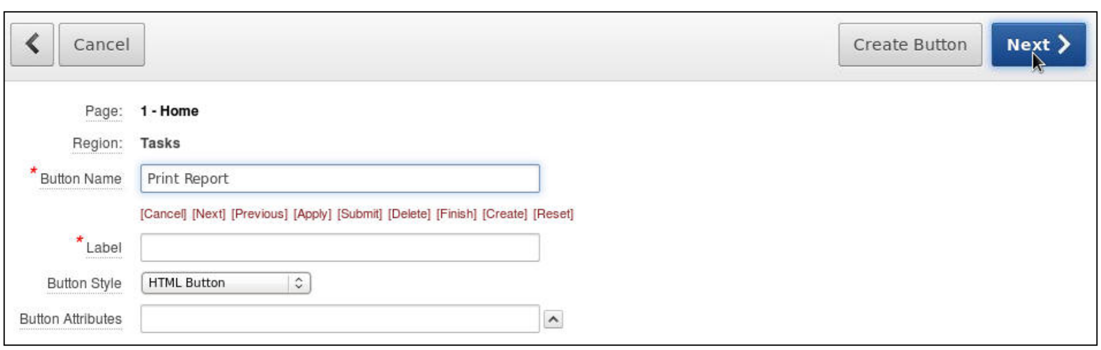
1) Create a button called Print Report on the Home page. Click the Home page.



2) Right-click the **Tasks** Region and select **Create Region Button**.



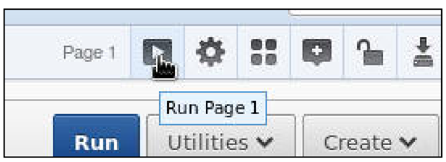
3) Enter **Print Report** for Button Name and Label. Click **Next**.



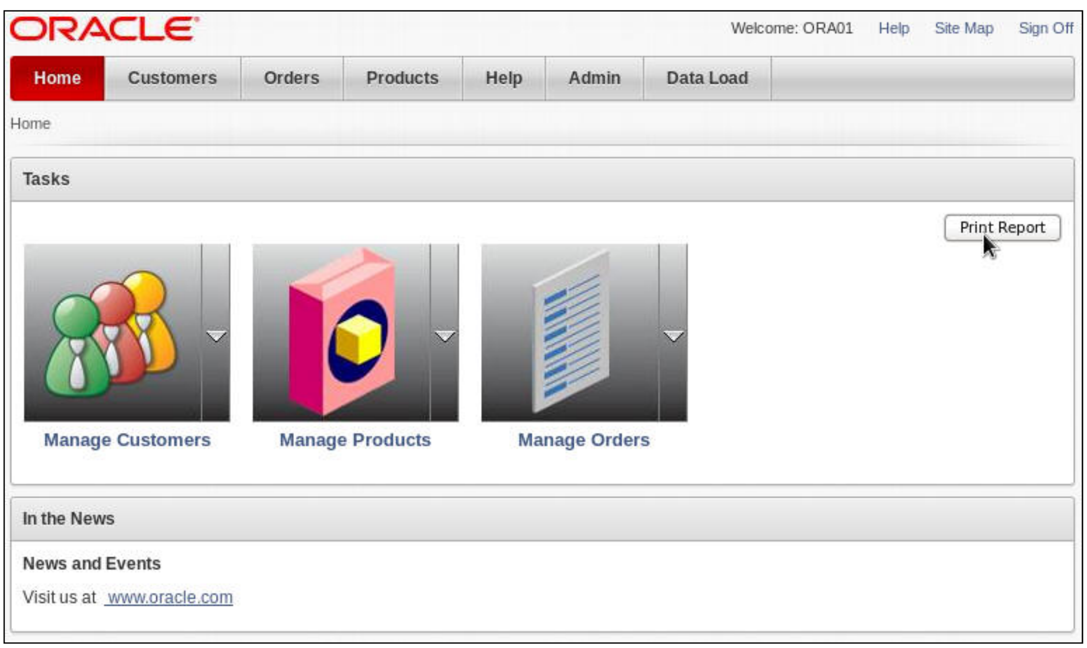
4) Select **Top of Region** for Position and click **Next**.

5) Select **Download Printable Report Query** for Action and **multiquery** for Report Query. Click **Create Button**.

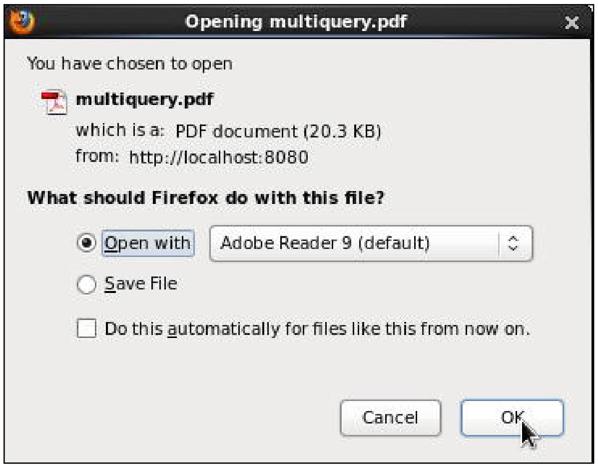
6) Run the page.



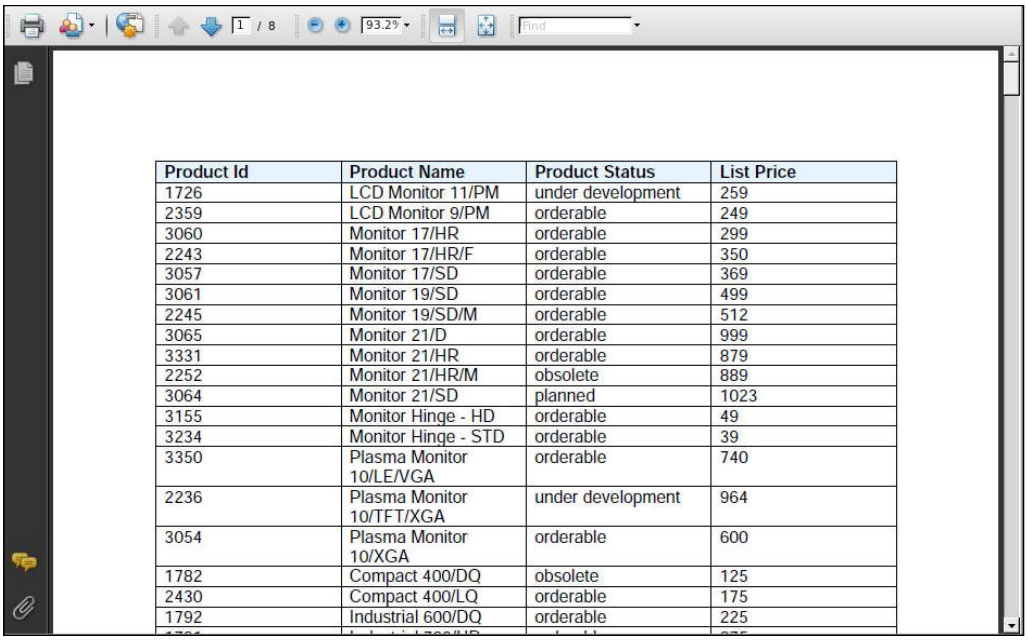
7) Click **Print Report**.



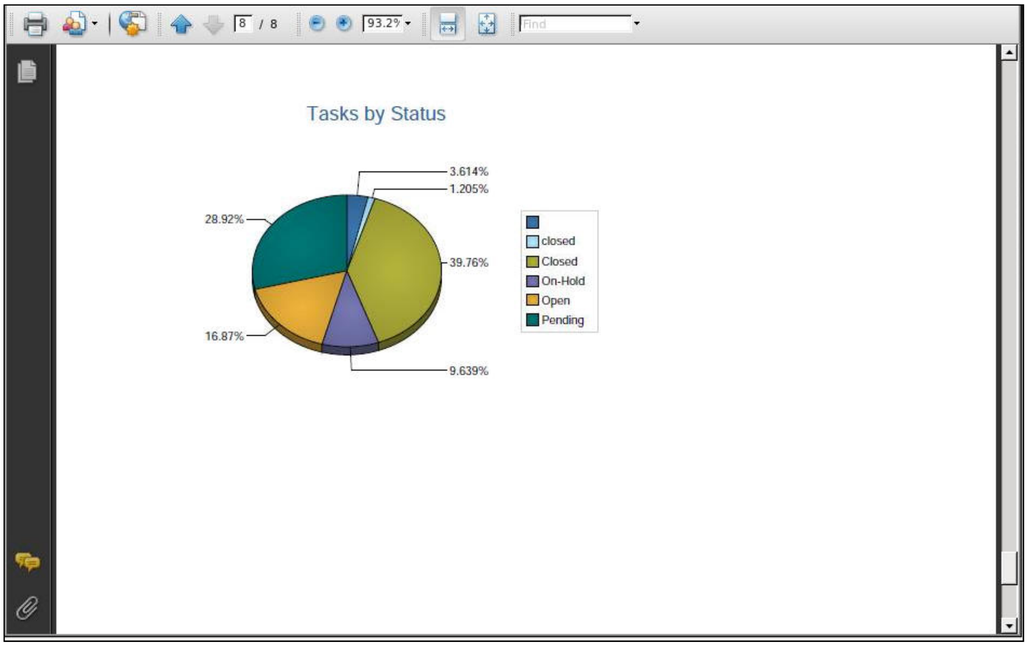
8) Click **OK**.



9) The Report is displayed.



Product Id	Product Name	Product Status	List Price
1726	LCD Monitor 11/PM	under development	259
2359	LCD Monitor 9/PM	orderable	249
3060	Monitor 17/HR	orderable	299
2243	Monitor 17/HR/F	orderable	350
3057	Monitor 17/SD	orderable	369
3061	Monitor 19/SD	orderable	499
2245	Monitor 19/SD/M	orderable	512
3065	Monitor 21/D	orderable	999
3331	Monitor 21/HR	orderable	879
2252	Monitor 21/HR/M	obsolete	889
3064	Monitor 21/SD	planned	1023
3155	Monitor Hinge - HD	orderable	49
3234	Monitor Hinge - STD	orderable	39
3350	Plasma Monitor 10/LE/VGA	orderable	740
2236	Plasma Monitor 10/TFT/XGA	under development	964
3054	Plasma Monitor 10/XGA	orderable	600
1782	Compact 400/DQ	obsolete	125
2430	Compact 400/LQ	orderable	175
1792	Industrial 600/DQ	orderable	225


Tasks by Status
3.614%
1.205%
28.92%
39.76%
16.87%
9.639%
closed
Closed
On-Hold
Open
Pending

Practices for Lesson 21: Managing Application Feedback

Chapter 21

Practices for Lesson 21: Overview

Practices Overview

In this practice, you create a feedback form in your application, submit feedback, and review and categorize the feedback.

Practice 21-1: Adding and Monitoring Feedback in Your Application

Overview

In this practice, you create a feedback form in your application, submit feedback, and review and categorize the feedback.

Assumptions

You have performed the previous practice.

Tasks

a. Create a feedback form for your Order Management application.

b. Create the following feedback from your application:

Feedback Number	Feedback Text	Feedback Type
1.	Date format needs to be changed on Master detail	General comment
2.	Error when deleting a line item on an order	Bug
3.	Add label help text on Orders page	General comment

c. Review your feedback in Team Development.

d. Log feedback 1 (“Date format needs to be changed on Master detail”) as a feature with a status of “Functionally complete - 80%.”

e. Log feedback 3 (Add label help text on Orders page) as a To Do, assign it to `Tom`, and change the status to “Assigned - 30%.” Delete the feedback entry after logging it as a To Do.

f. Edit the To Do, changing the status to “Work Progressing - 80%” and specifying Documentation for the category.

g. Log feedback 2 (“Error when deleting a line item on an order”) as a bug. Set the severity to “4. Moderate Impact” and assign it to `Tom`.

h. Edit the bug and change the status to “80. Fixed in development,” and set the priority to “1. As soon as possible.”

i. Review the Team Development dashboard.

Solution 21-1: Adding and Monitoring Feedback in Your Application

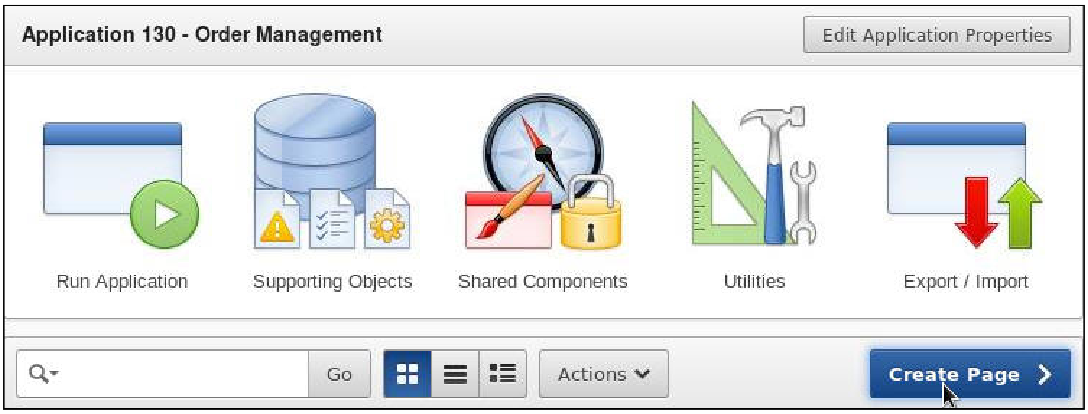
Overview

In this practice, you create a feedback form in your application, submit feedback, and review and categorize the feedback.

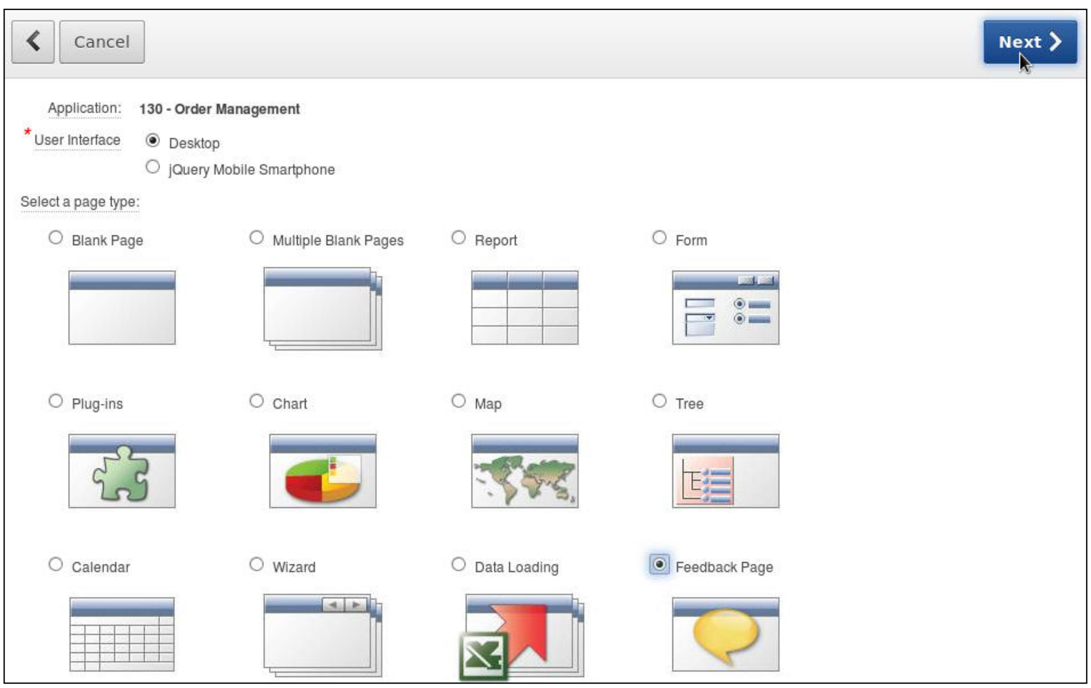
Steps

a. Create a feedback form for your Order Management application.

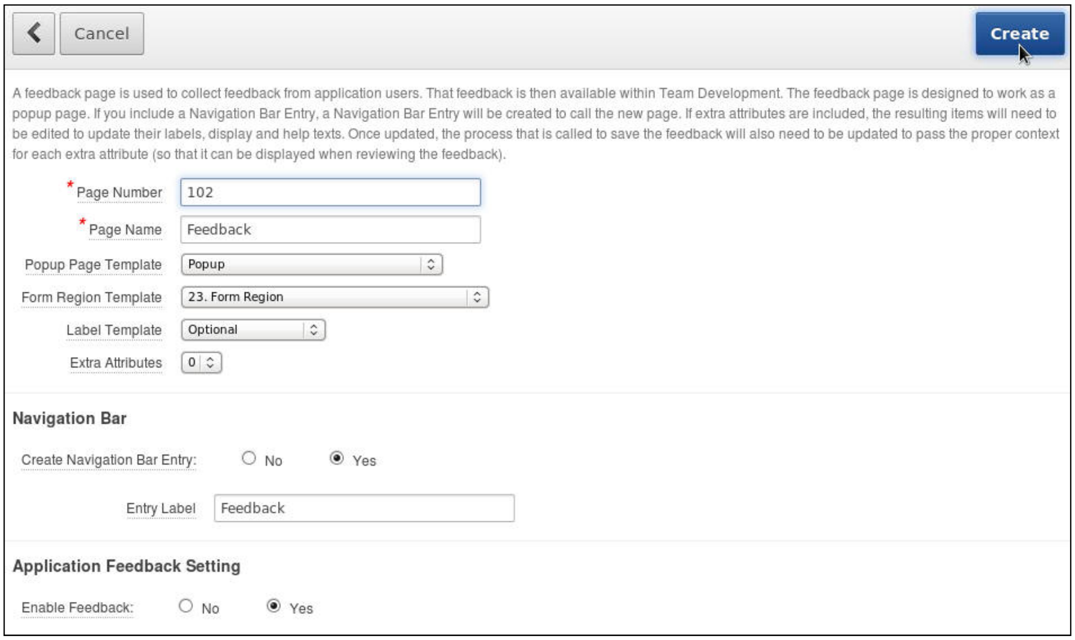
1) On the Order Management application home page, click **Create Page >**.



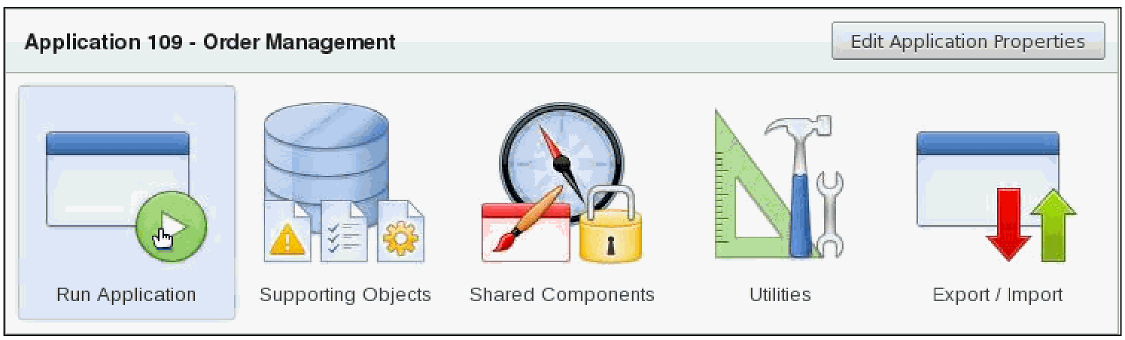
2) Select **Feedback Page** and click **Next >**.



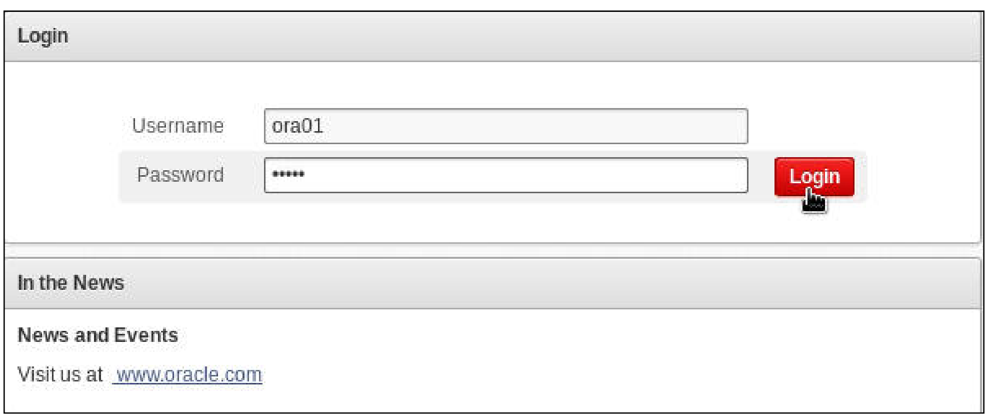
3) Accept all the defaults and click **Create**. Notice that a navigation bar entry will be created and feedback will be enabled.



4) Click **Run Application**.



5) If you receive the login window, enter your username and password and click **Login**. If you do not receive the login window, click Logout and log in again.

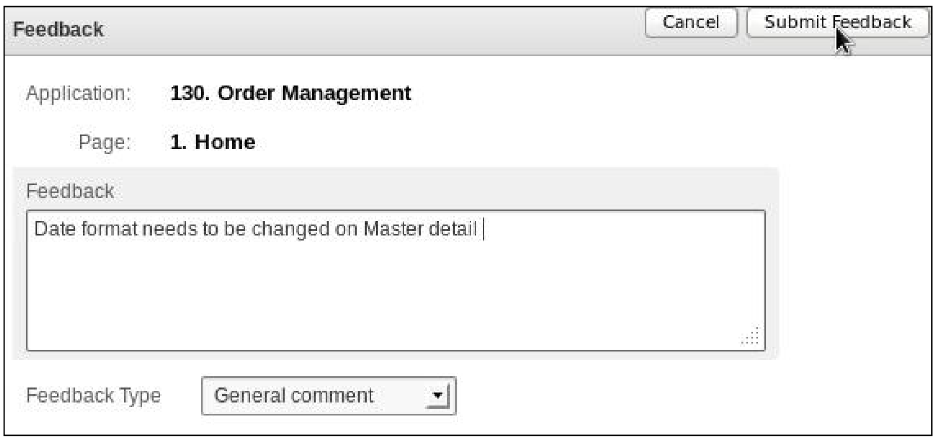


b. Create the following feedback from your application:

Feedback Number	Feedback Text	Feedback Type
1.	Date format needs to be changed on Master detail	General comment
2.	Error when deleting a line item on an order	Bug
3.	Add label help text on Orders page	General comment

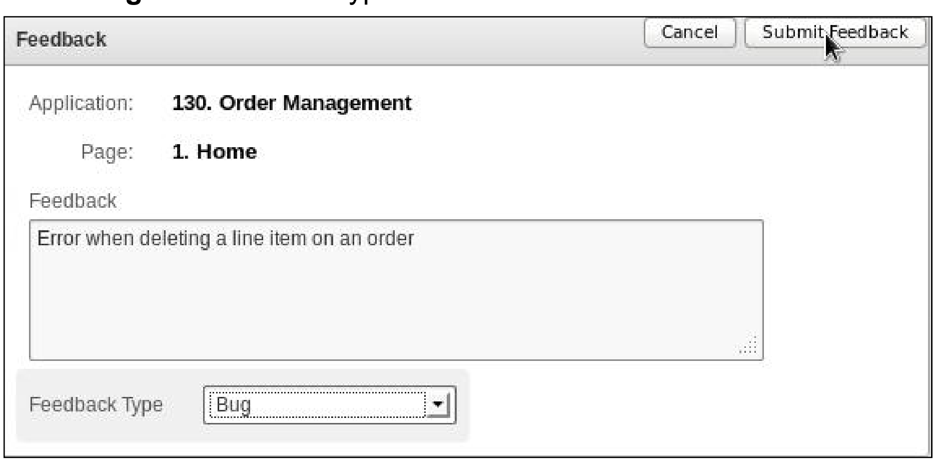
1) Click **Feedback** in the navigation bar.

2) Enter **Date format needs to be changed on Master detail** in the feedback text area and click **Submit Feedback**.



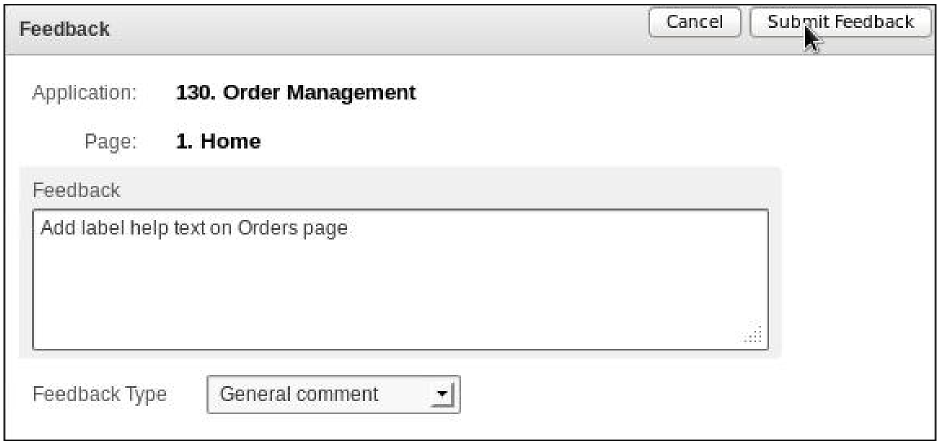
3) You want to create another feedback item. Click **Feedback** in the navigation bar.

4) Enter **Error when deleting a line item on an order** in the feedback text area and select **Bug** for Feedback Type. Click **Submit Feedback**.



5) You want to create one more feedback item. Click **Feedback** in the navigation bar.

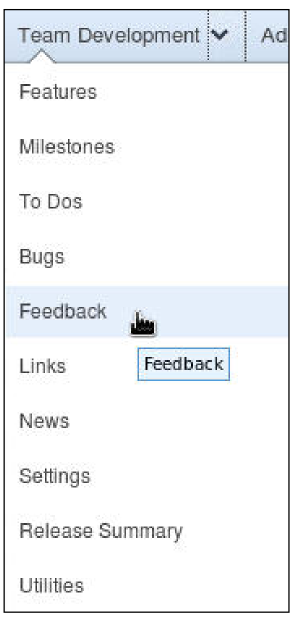
6) Enter **Add label help text on Orders page** in the feedback text area and click **Submit Feedback**.



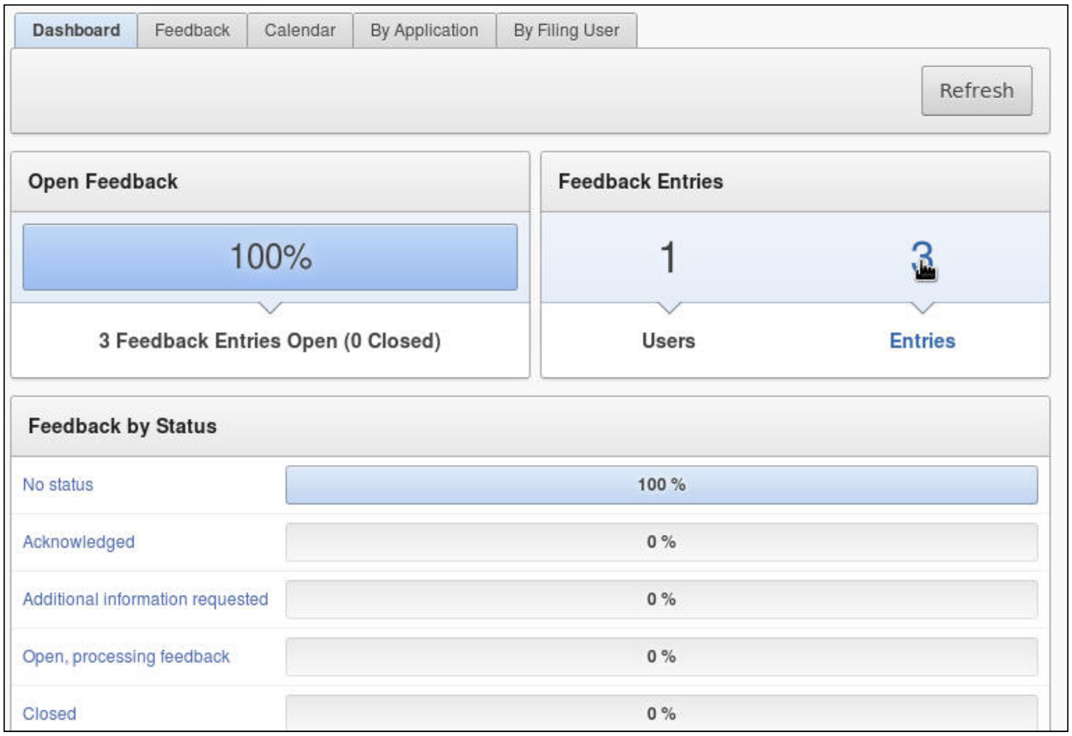
7) You have entered all your feedback. Now you want to view it in Team Development. Click **Home** on the Developer toolbar.

c. Review your feedback in Team Development.

1) Select **Feedback** from the **Team Development** drop-down list.

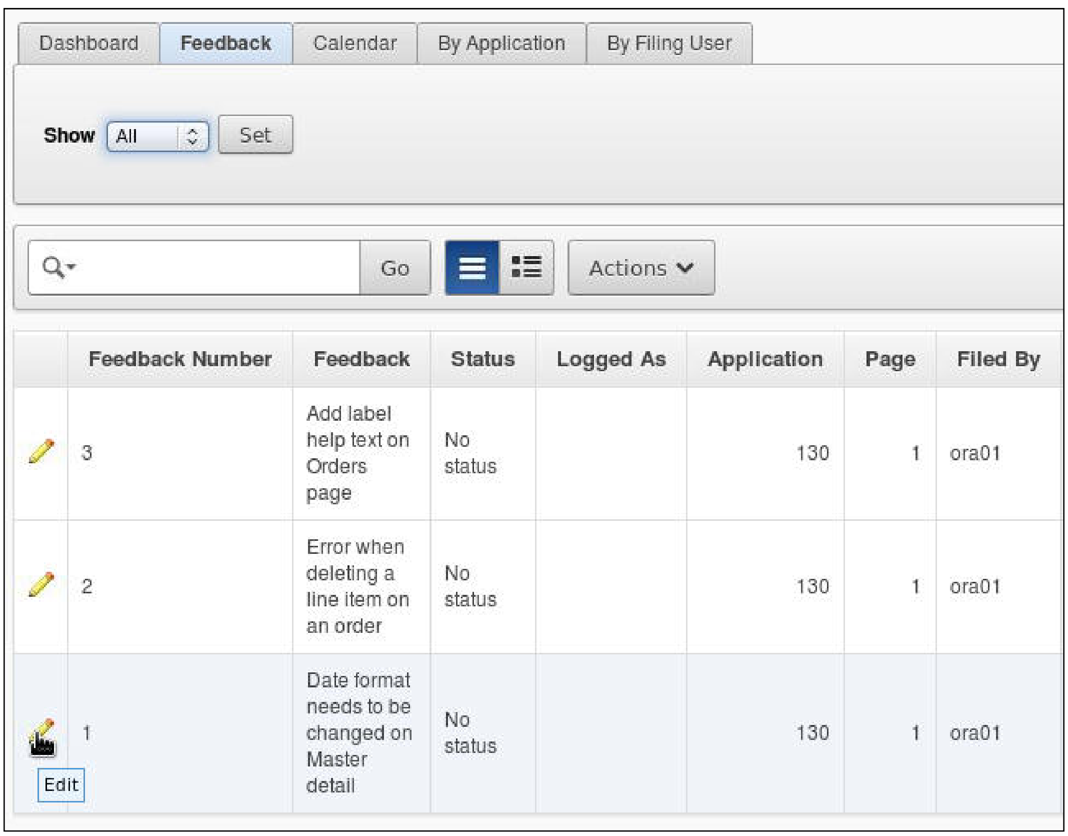


2) The Feedback dashboard is displayed. Notice that feedback items have been filed. Under **Feedback Entries**, click **Entries**.

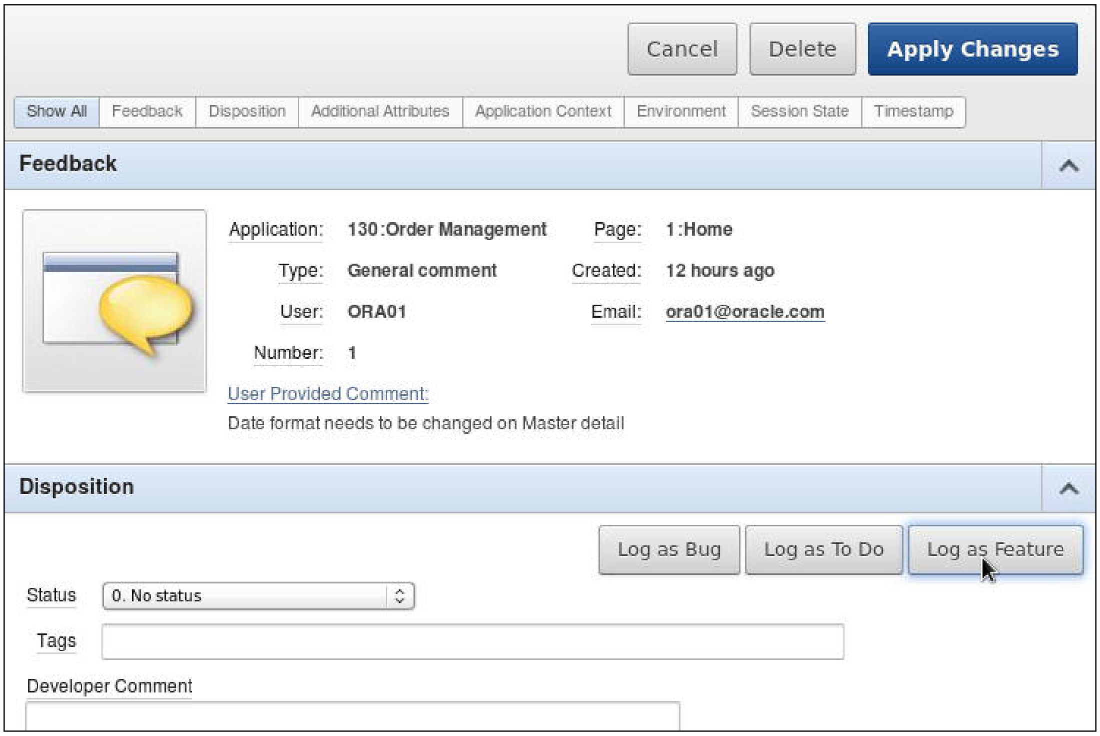


d. Log feedback 1 (“Date format needs to be changed on Master detail”) as a feature with a status of “Functionally complete - 80%.”

1) Click the **Edit** icon for Feedback 1.

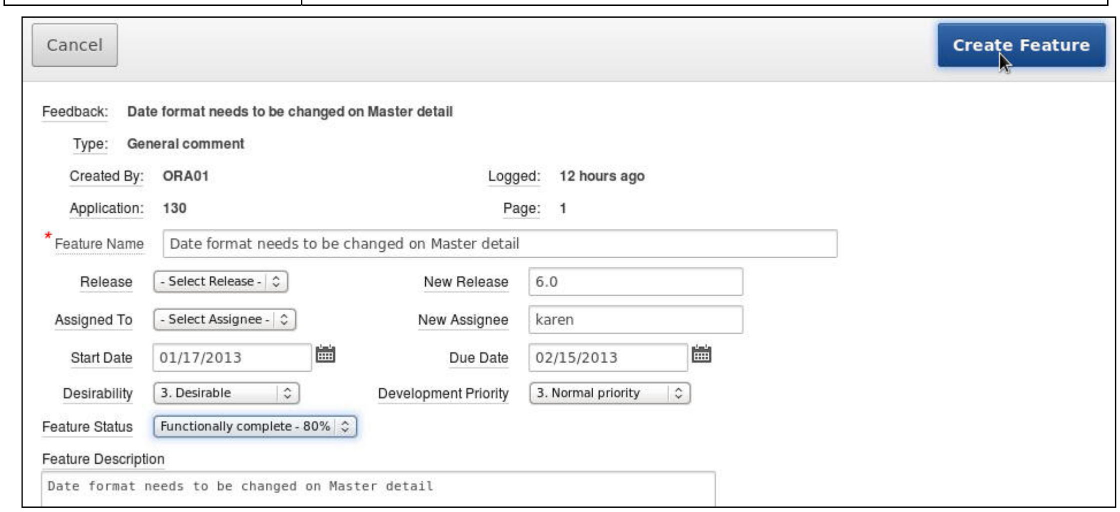


2) Your feedback is displayed. Click the **Log as Feature** button.



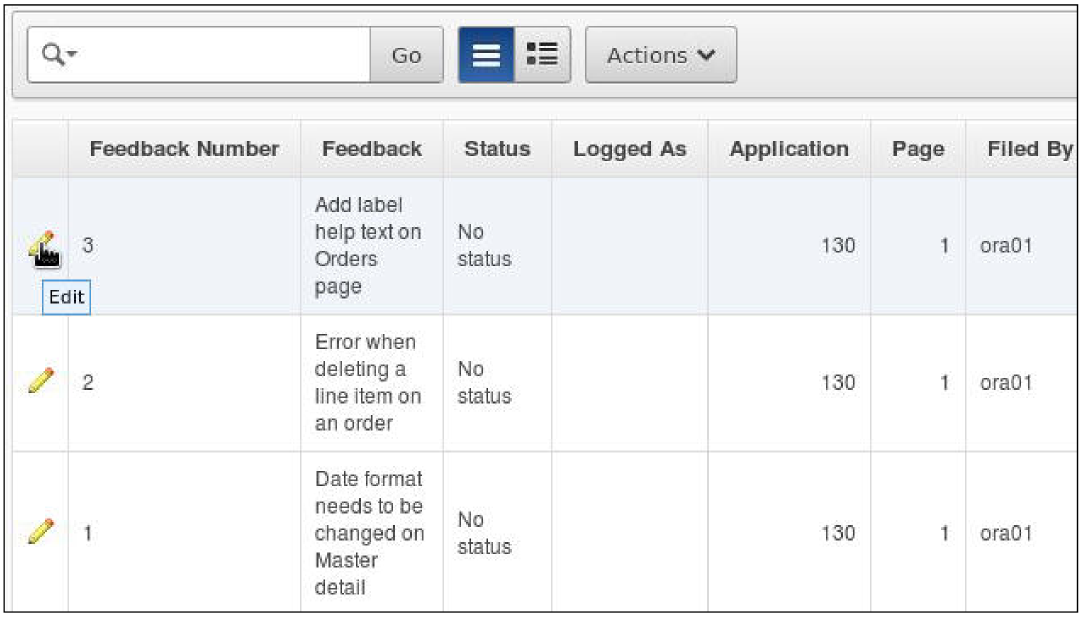
3) Specify the following and click **Create Feature**:

Name	Value
Feature Name	**Date format needs to be changed on Master detail**
New Release	**6.0**
New Assignee	**karen**
Start Date	**Select tomorrow's date**
Due Date	**Select date one month from now**
Desirability	**3. Desirable**
Priority	**3. Normal Priority**
Feature Status	**Functionally complete - 80%**

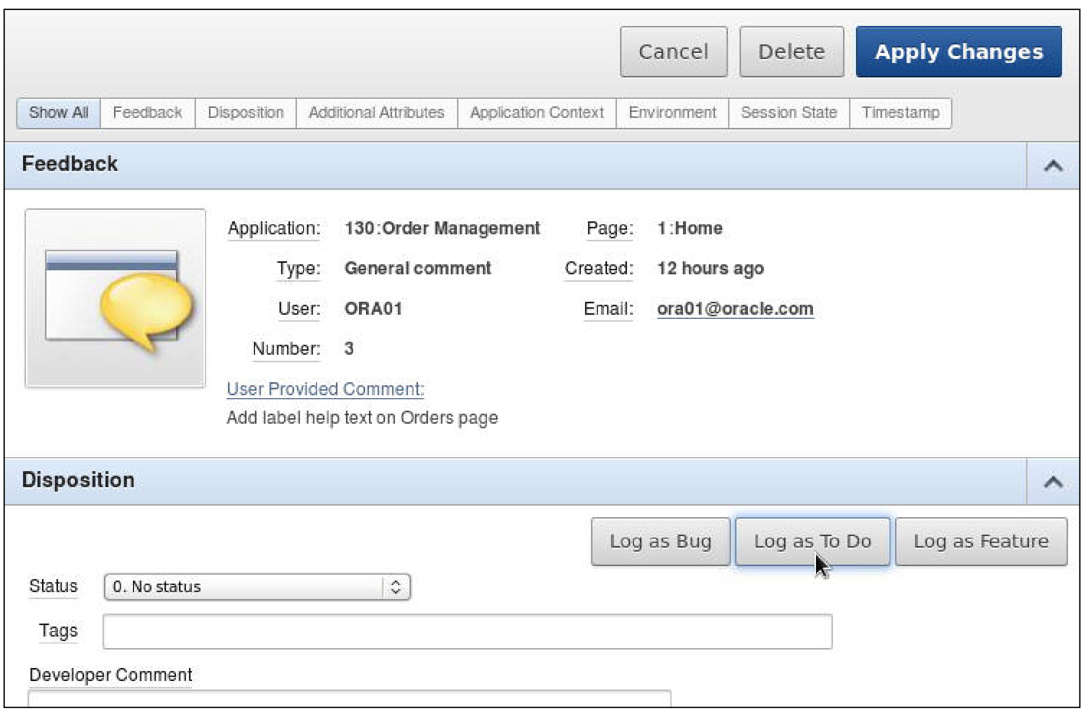


e. Log feedback 3 (“Add label help text on Orders page”) as a To Do, assign it to `tom`, and change the status to “Assigned - 30%.” Delete the feedback entry after logging as a To Do.

1) Click **Edit** icon for Feedback 3.

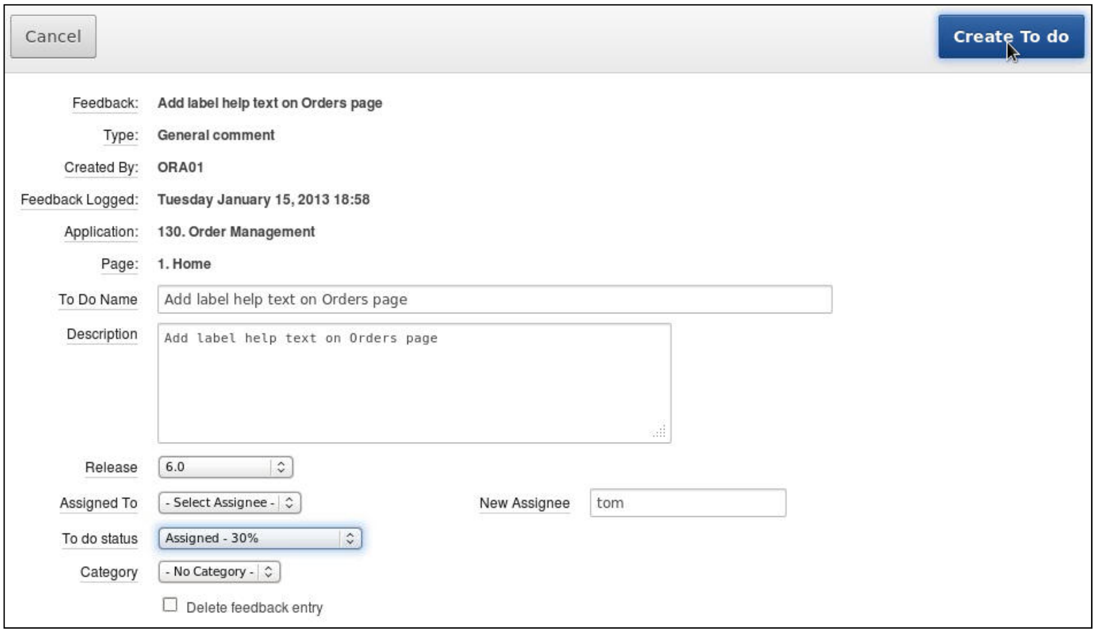


2) You want to make Feedback 3 a To Do. Click the **Log as To do** button for Feedback 3.



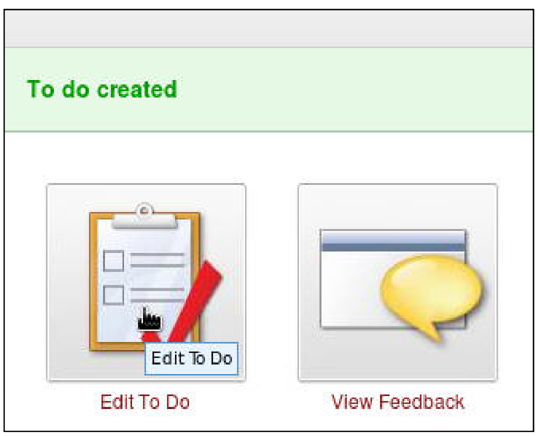
3) Specify the following and click **Create To do**.

Name	Value
Release	**6.0**
New Assignee	**tom**
To do status	**Assigned - 30%**

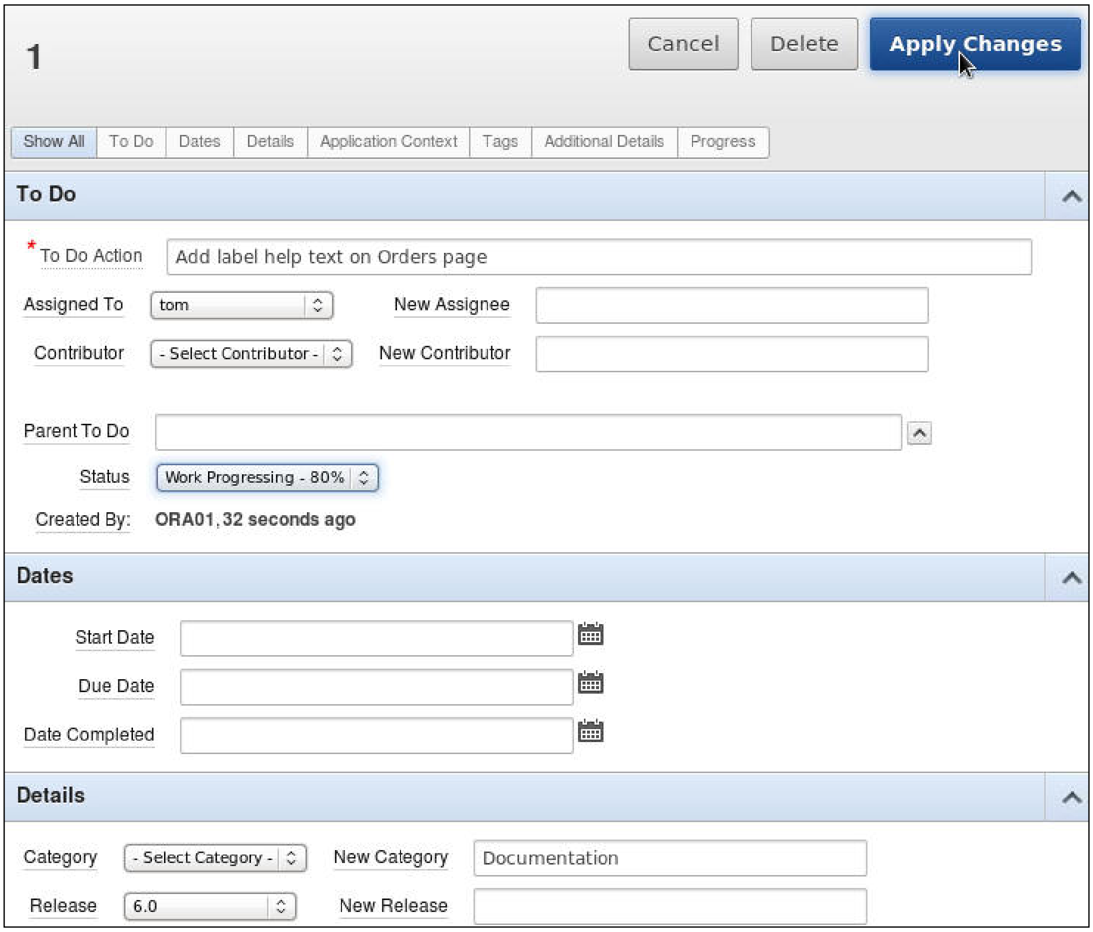


f. Edit the To Do, changing the status to “Work Progressing - 80%” and specifying Documentation for the category.

1) Click **Edit To Do**.

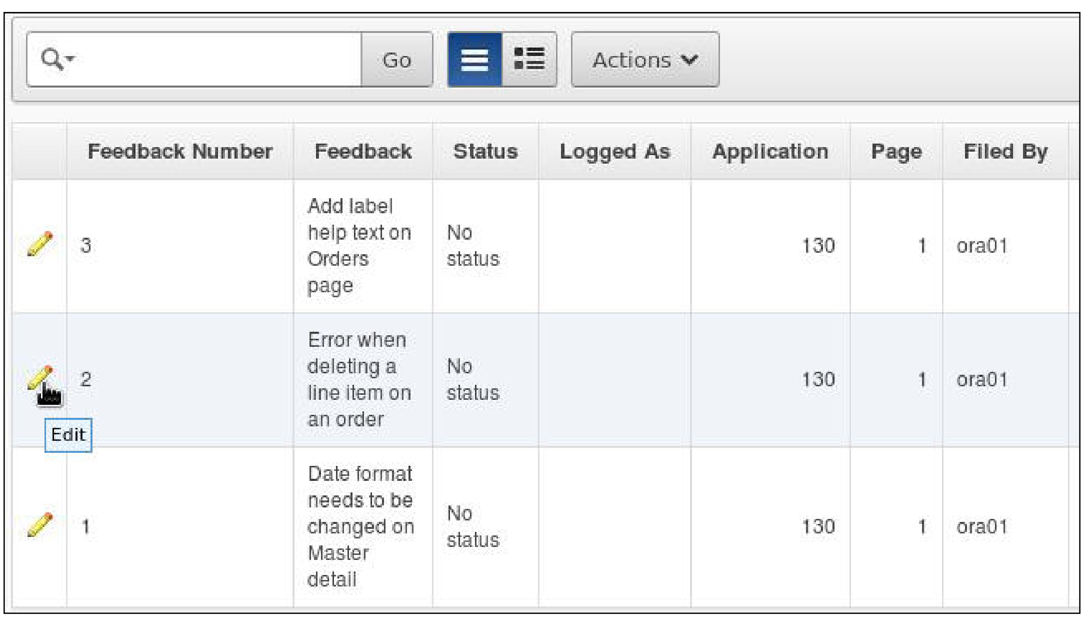


2) Change the Status to **Work Progressing – 80%**, enter **Documentation** for New Category, and click **Apply Changes**.

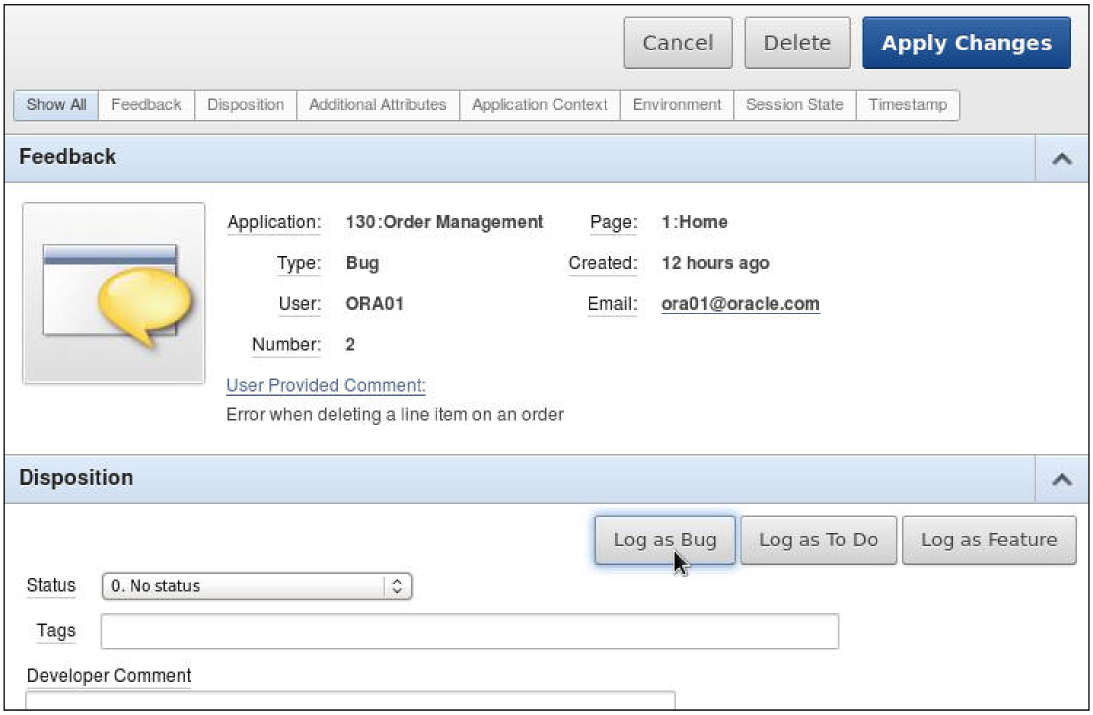


g. Log Feedback 2 ("Error when deleting a line item on an order") as a bug. Set the severity to "4. Moderate Impact," and assign it to `tom`.

1) Click **Edit** icon for Feedback 2.

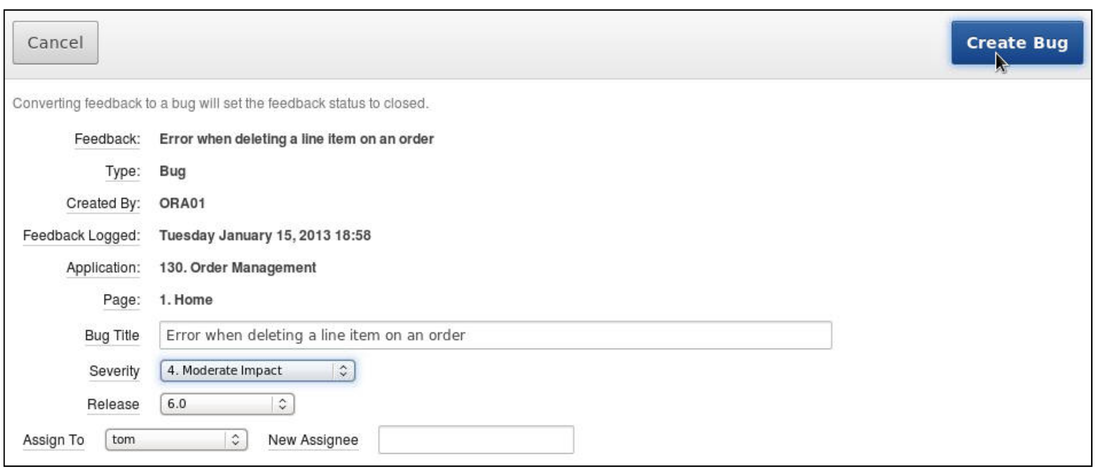


2) Feedback 2 is a bug. Click the **Log as Bug** button for Feedback 2.



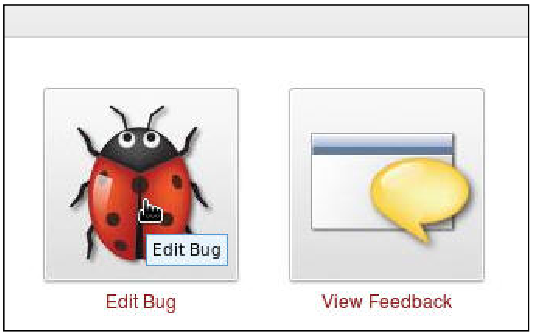
3) Specify the following and click **Create Bug**.

Name	Value
Severity	**4. Moderate Impact**
Release	**6.0**
Assign To	**tom**

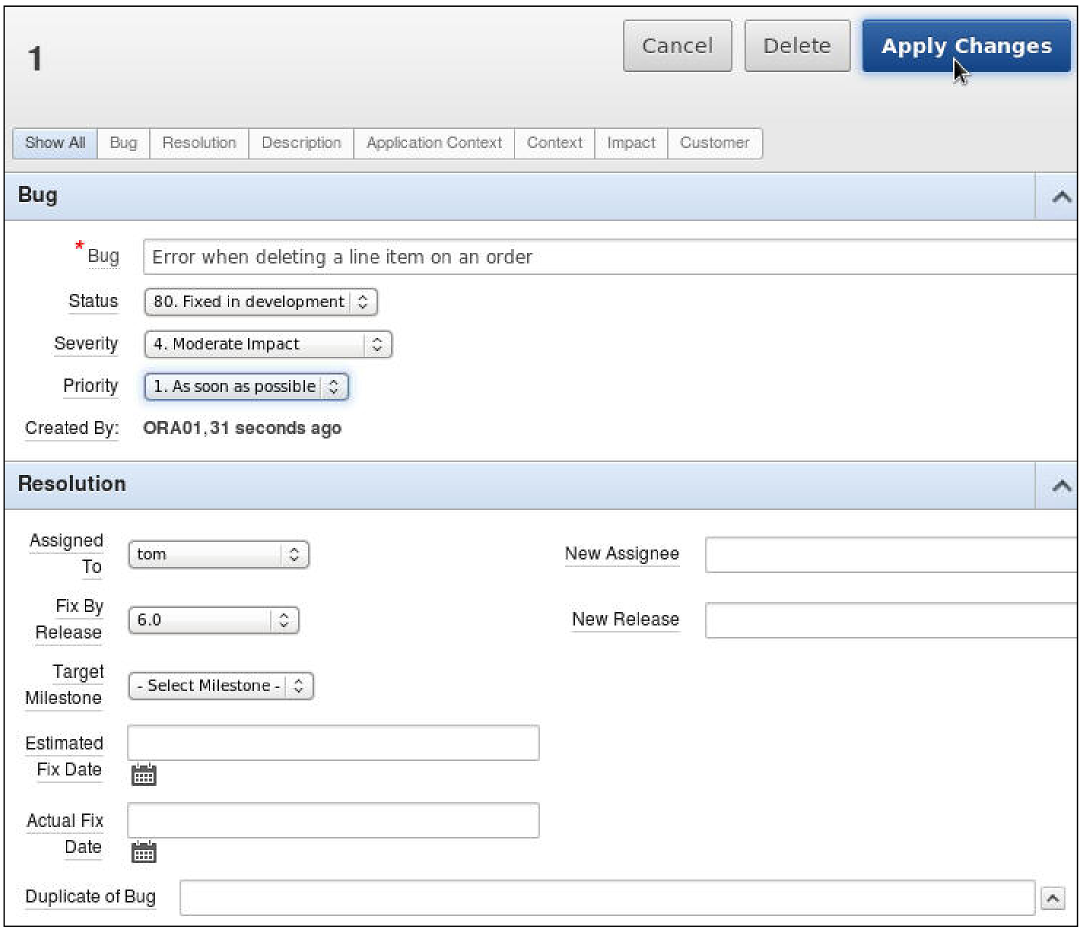


h. Edit the bug and change the status to “80. Fixed in development”, and set the priority to “1. As soon as possible.”

1) In the **Team Development** breadcrumb, click **Edit Bug**.

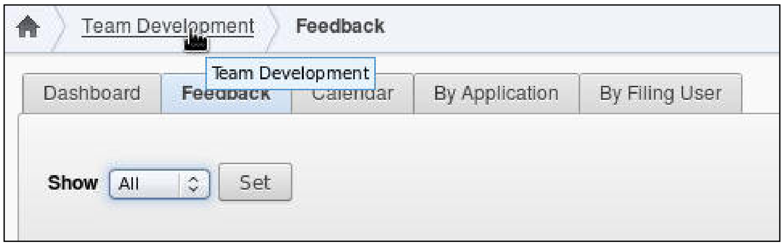


2) Select **80. Fixed in development** for Status, **As soon as possible** for Priority Click **Apply Changes.**



i. Review the Team Development dashboard.

1) Click the **Team Development** breadcrumb.



2) An overview of all your features, milestones, to dos, and bugs are displayed.