



Oracle Application Express: Developing Database Web Applications

Hands-On-Labs Guide

Unit 15: Creating and Using Dynamic Actions and Plug-ins

This exercise includes five hands-on-labs and uses the Demo Projects application.

HOL 15-1 Creating and Using a Dynamic Action on the Maintain Project Page: In this lab, you add a dynamic action on the Maintain Project page to show or hide the Completed Date based on the value of Status.

HOL 15-2 Creating and Using a Dynamic Action on the Maintain Task Page: In this lab, you add a dynamic action on the Maintain Task page to populate the Due Date whenever the Milestone is changed.

HOL 15-3 Creating and Using Dynamic Actions on the Project Tree Page: This lab covers creating two buttons that you use to expand and contract the project tree nodes. You create a dynamic action on each of these buttons. Then, for each of the button, you set the action as defined by the dynamic action.

HOL 15-4 Creating and Using a Plug-in: In this hands-on lab you create a copy of the Badge List Plug-in from the Sample Charts packaged app. Then, on the application home page, you create a badge list region. On this region, you display three different badges: Open Projects, Upcoming Milestones, and Open Tasks.

HOL 15-5 Updating the Home Page: In this hands-on lab you create a dashboard by adding new components to the Home page of the application. You replace the Project Tasks chart with the Recent Projects classic report region. Then, you adjust the Recent Projects and My Outstanding Tasks regions so that they are placed horizontally next to each other. Finally, you modify the Summary region (Badge) attributes to enable the colorful display of the badges.

HOL 15-1: Creating and Using a Dynamic Action on the Maintain Project Page

On the Maintain Project page, the Completed Date field is displayed irrespective of the value of Status. In this lab, you add a dynamic action on the page to show or hide the Completed Date based on the value of Status.

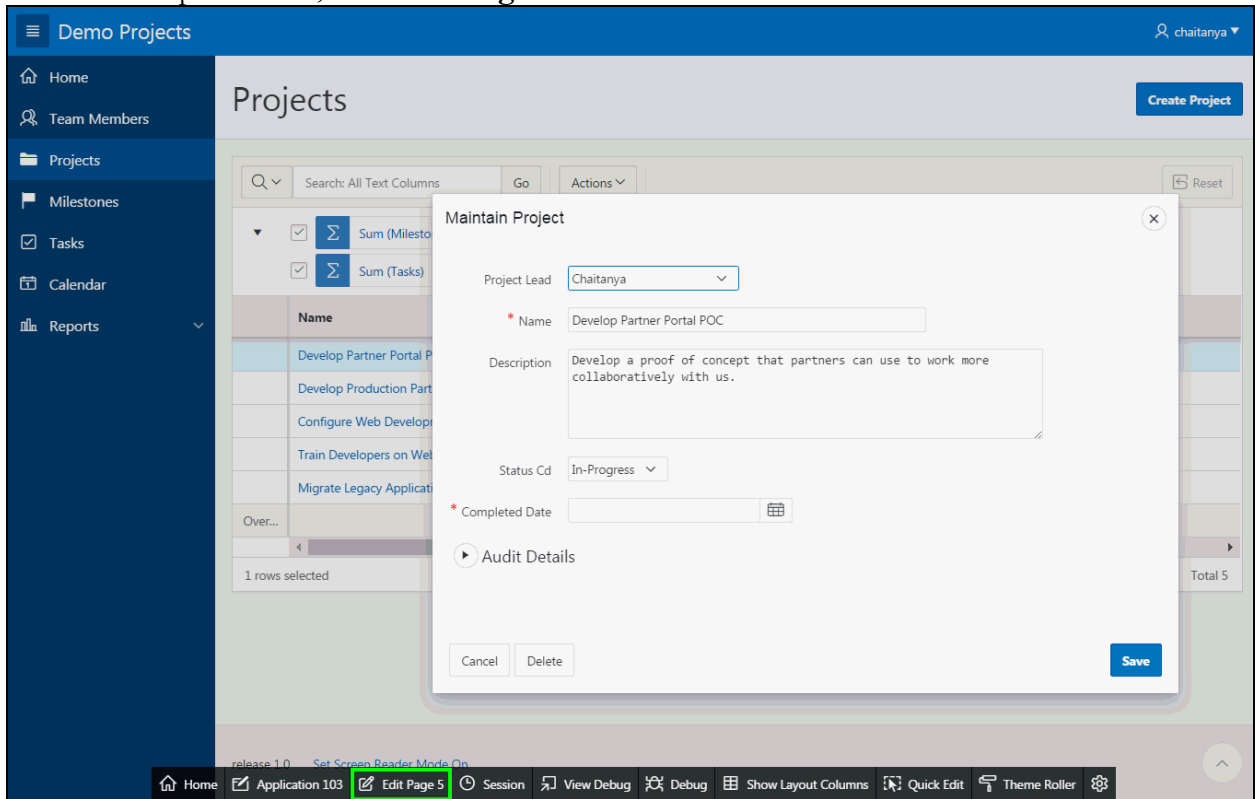
1. In the Demo Projects application home page, click **Shared Components**.
2. You want to create a dynamic LOV for Status. Under Other Component, click **List of Values**.
3. Click **Create**.
Accept the default for Source and click **Next**.
4. For Name, enter **STATUS**, select **Dynamic** for Type and click **Next**.
5. For Query, copy and paste the following SQL:

```
select description d, cd r
from demo_proj_status
order by display_order
```
6. Click **Create List of Values**.
7. In the Application Express toolbar, click **Run Page 1**.
In the Demo Projects application runtime environment, click **Projects** in the navigation menu. Then, in the interactive grid, select a project name with status as IN-PROGRESS or ASSIGNED.

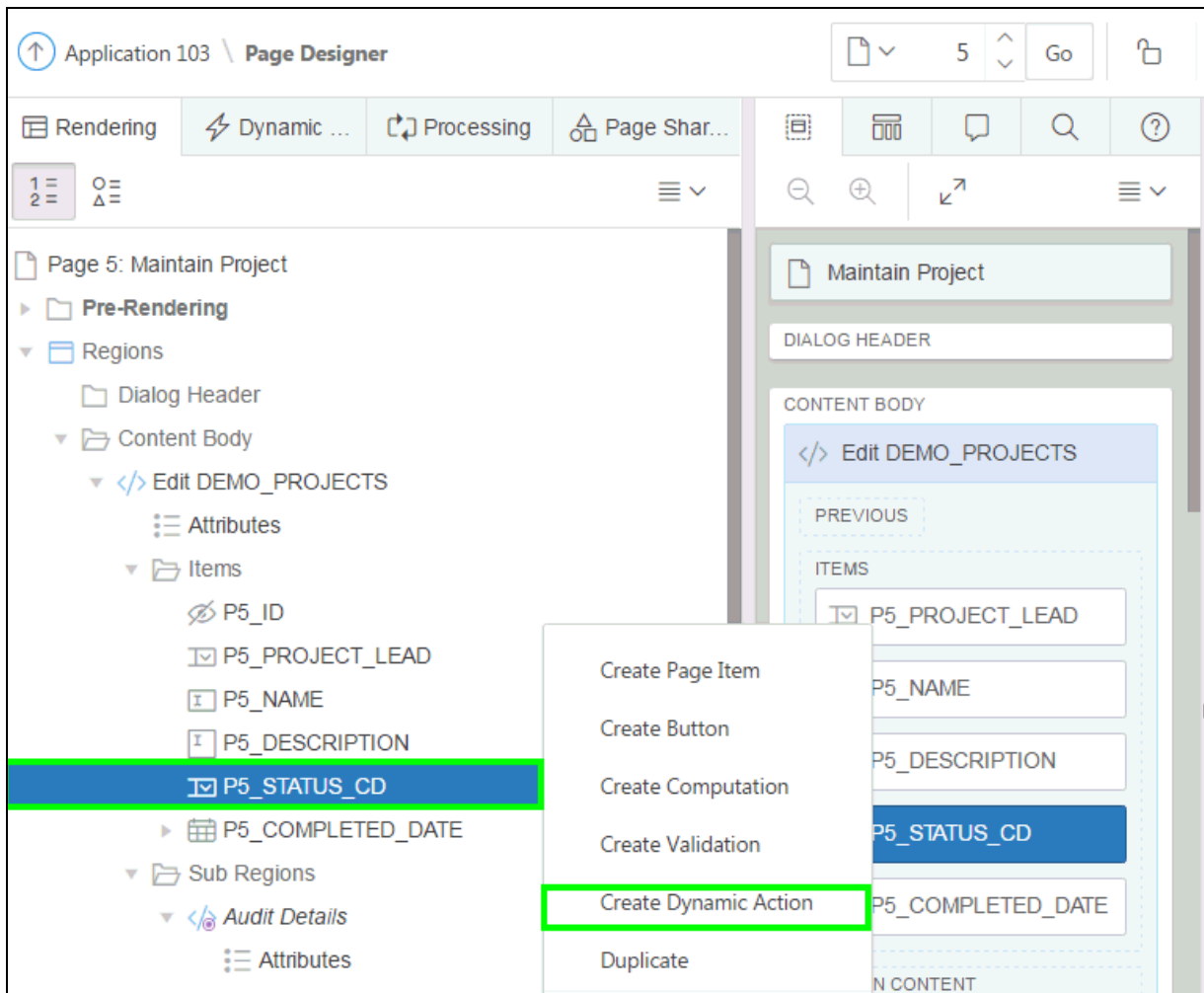
In the dialog page, for Status Cd, select each of the values Assigned, Completed, and In-Progress. Notice that the Completed Date field is displayed irrespective of the status.

You want to create a dynamic action that will allow the display of the Completed Date field only if the status is *Completed*.

In the Developer toolbar, click **Edit Page 5**.



- In the page designer, expand Regions > Edit Demo_Projects > Items. Right-click the **P5_STATUS_CD** item and select **Create Dynamic Action**.



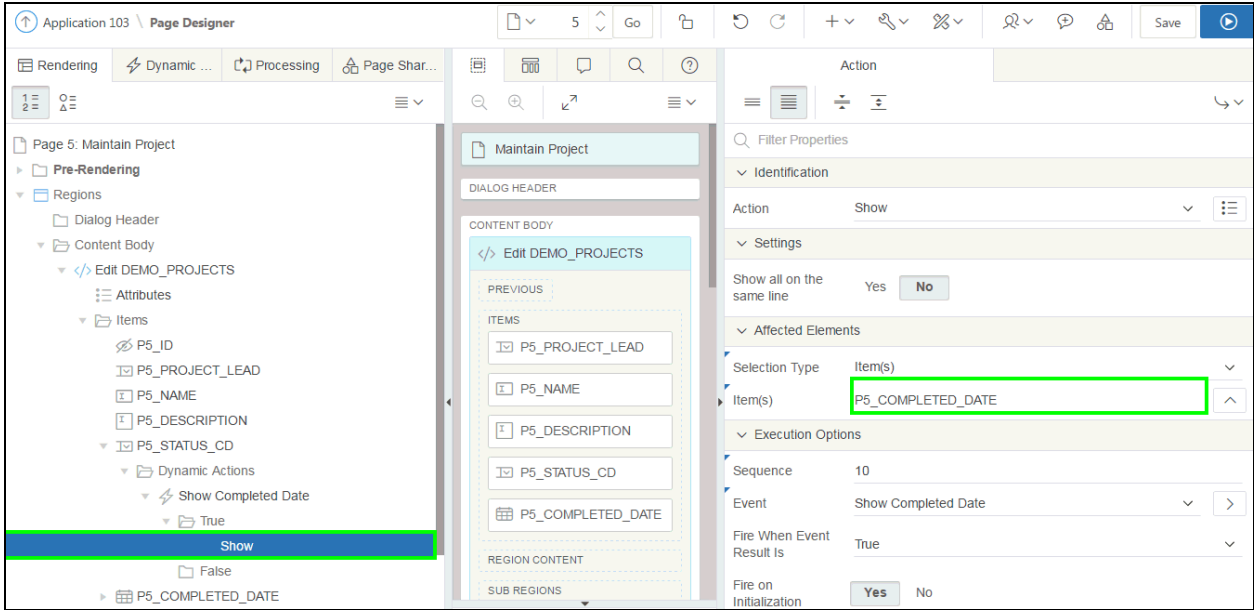
9. In the property editor:

- Identification > Name: Enter **Show Completed Date**
- Client-side Condition > Type: Select **Item = Value**
- Client-side Condition > Item: Select **P5_STATUS_CD**
- Client-side Condition > Value: Enter **COMPLETED**

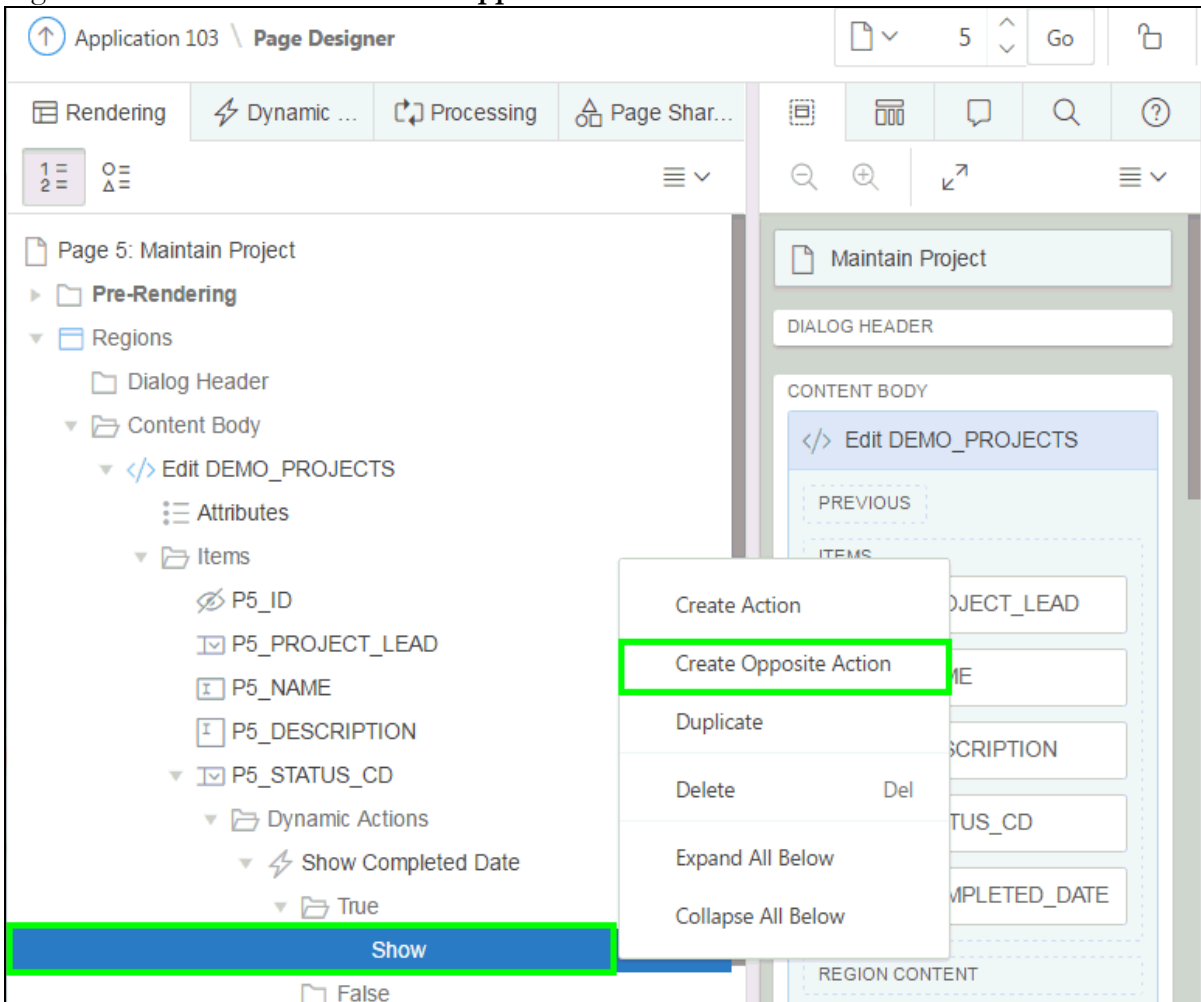
Note: The capitalization and spelling of the value must match the data entry value exactly in order for the dynamic action to fire.

Dynamic Action	
<div style="display: flex; justify-content: space-between; align-items: center;"> ☰ ☰ ↕ ↕ ↶ </div>	
<input type="text" value="Filter Properties"/>	
<div style="background-color: #f2f2f2; padding: 2px;"> ▼ Identification </div>	
Name	<input style="border: 2px solid green;" type="text" value="Show Completed Date"/>
<div style="background-color: #f2f2f2; padding: 2px;"> ▼ Execution Options </div>	
Sequence	20
<div style="background-color: #f2f2f2; padding: 2px;"> ▼ When </div>	
Event	Change ▼ ☰
Selection Type	Item(s) ▼ ☰
Item(s)	P5_STATUS_CD ▲
<div style="background-color: #f2f2f2; padding: 2px;"> ▼ Client-side Condition </div>	
Type	<input style="border: 2px solid green;" type="text" value="Item = Value"/> ▼ ☰
Item	P5_STATUS_CD ▲
Value	<input style="border: 2px solid green;" type="text" value="COMPLETED"/>
<div style="background-color: #f2f2f2; padding: 2px;"> ▼ Advanced </div>	

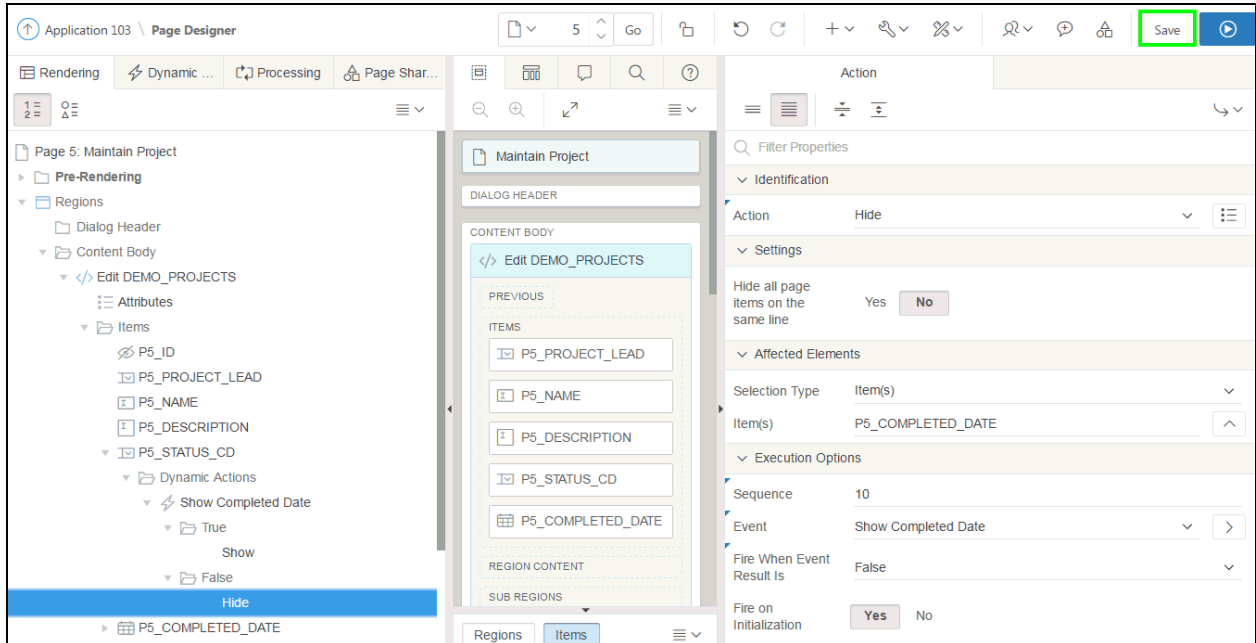
- In the Rendering tree, under the Show Completed Date dynamic action, expand the **True** node and select **Show**.
In the Property Editor, for Item(s) select **P5_COMPLETED_DATE**.



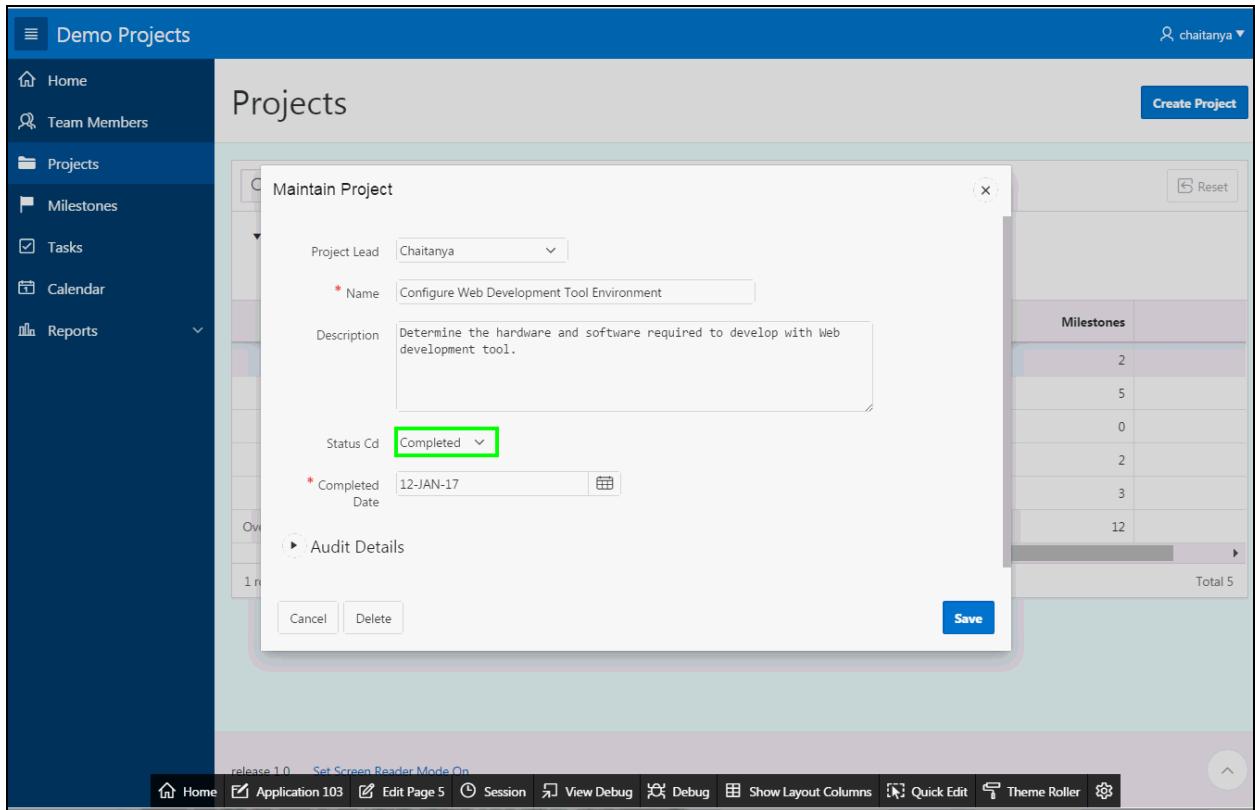
11. Right-click **Show** and select **Create Opposite Action**.



12. Notice that the opposite action is created now. The Completed Date field will be hidden if the Status includes a value other than COMPLETED.
Click **Save**.



13. Now, you can verify to see if the dynamic action works as expected.
Navigate to the Demo Projects application runtime environment. If the Maintain Project dialog is open, click **Cancel**.
In the navigation menu, click **Projects** and then in the report, click any project name with the Status as COMPLETED.
Notice that the Completed Date field is displayed.



- In the Maintain Project dialog, for Status Cd, select either Assigned or In-Progress. Notice that this time, the Completed Date field is hidden. Click the **Cancel** button.

The screenshot shows the Oracle APEX 'Maintain Project' dialog box. The dialog contains the following fields and controls:

- Project Lead:** Chaitanya (dropdown menu)
- Name:** Configure Web Development Tool Environment (text input)
- Description:** Determine the hardware and software required to develop with Web development tool. (text area)
- Status Cd:** In-Progress (dropdown menu)
- Audit Details:** (expandable section)
- Buttons:** Cancel, Delete, and Save.

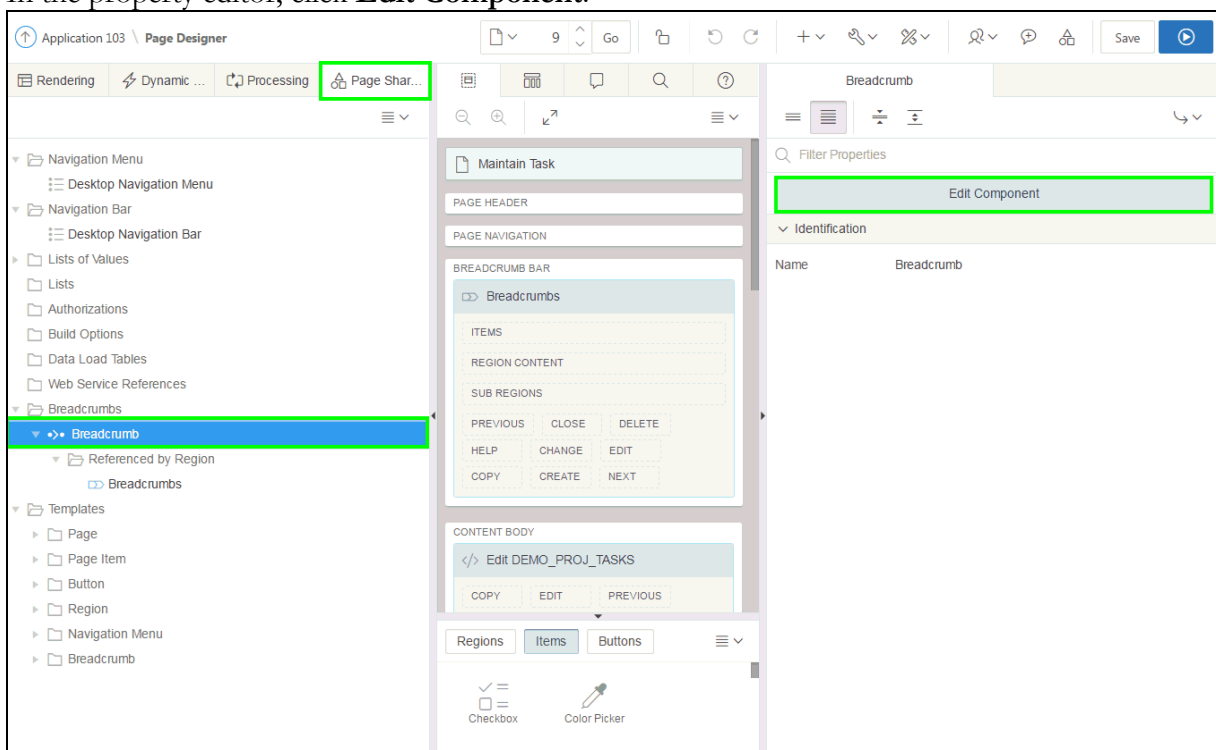
The background shows the 'Projects' page with a 'Create Project' button and a 'Milestones' table. The 'Milestones' table has the following data:

Milestones	
2	
5	
0	
2	
3	
12	
Total 5	

HOL 15-2: Creating and Using a Dynamic Action on the Maintain Task Page

In this hands-on lab, first, you rename the breadcrumb entry for Page 9 as Maintain Task. Then, you add a dynamic action on this page to populate the Due Date whenever the Milestone is changed.

1. In the Demo Projects application runtime environment, click **Tasks** in the navigation menu.
In the interactive grid, click the **Edit** icon for a task name.
Notice that the breadcrumb entry still shows **Demo Proj Tasks**.
In the Developer Toolbar, click **Edit Page 9**.
2. In the page designer, click **Page Shared Components**.
Expand **Breadcrumbs** and select **Breadcrumb**.
In the property editor, click **Edit Component**.



3. Click **Demo Proj Tasks**.

The screenshot shows the Oracle APEX interface for managing breadcrumbs. The breadcrumb is set to 'Breadcrumb'. The page number is 9. The breadcrumb entry 'Demo Proj Tasks' is highlighted in a green box. The table below shows the details of the breadcrumb entry.

Name	Sequence	Page	Parent
Demo Proj Tasks	60	9	8. Tasks

4. Under Entry, for Short Name, enter **Maintain Task**.
Click **Apply Changes**.

The screenshot shows the 'Breadcrumb Entry: Demo Proj Tasks' configuration page. The 'Apply Changes' button is highlighted in a green box. The 'Entry' section is expanded, and the 'Short Name' field is set to 'Maintain Task' and highlighted in a green box.

Breadcrumb Entry: Demo Proj Tasks [Cancel] [Delete] **Apply Changes**

Show All | Breadcrumb | Entry | Target | Conditions | Authorization | Configuration

Breadcrumb

Breadcrumb: Breadcrumb

* Page: 9

Entry

Sequence: 60

Parent Entry: Tasks (Page 8)

* Short Name: **Maintain Task**

Long Name:

5. In the Application Express toolbar, click **Edit Page 9**.

On this page, you want to populate the Due Date whenever the Milestone is changed. In order to achieve this you will use a Dynamic Action with an action of Set Value, which can execute an AJAX call to retrieve data from the database.

6. Under Rendering > Regions > Edit Demo_Proj_Tasks, expand items.
Right-click **P9_MILESTONE_ID** and select **Create Dynamic Action**.

Application 103 \ Page Designer

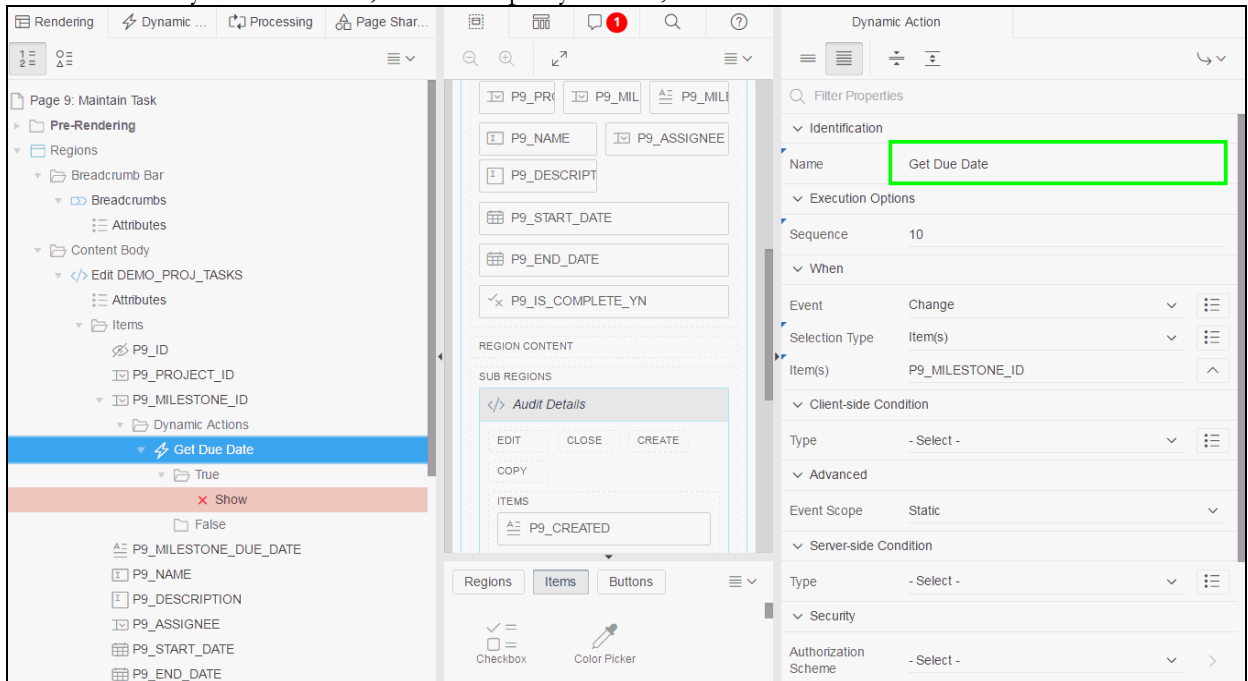
Rendering Dynamic ... Processing Page Shar...

1 2 O Δ

Page 9: Maintain Task

- Pre-Rendering
- Regions
 - Breadcrumb Bar
 - Breadcrumbs
 - Attributes
 - Content Body
 - Edit DEMO_PROJ_TASKS
 - Attributes
 - Items
 - P9_ID
 - P9_PROJECT_ID
 - P9_MILESTONE_ID**
 - P9_MILESTONE_DUE
 - P9_NAME
 - P9_DESCRIPTION
 - P9_ASSIGNEE
 - P9_START_DATE
 - P9_END_DATE
 - P9_IS_COMPLETE_YN

- Create Page Item
- Create Button
- Create Computation
- Create Validation
- Create Dynamic Action**
- Duplicate
- Delete Del
- Copy to other Page...

7. For the new Dynamic Action, in the Property Editor, for Name enter **Get Due Date**.8. In the Rendering tree, under the P9_MILESTONE_ID item, select the **Action** under the True node (currently labeled X Show).

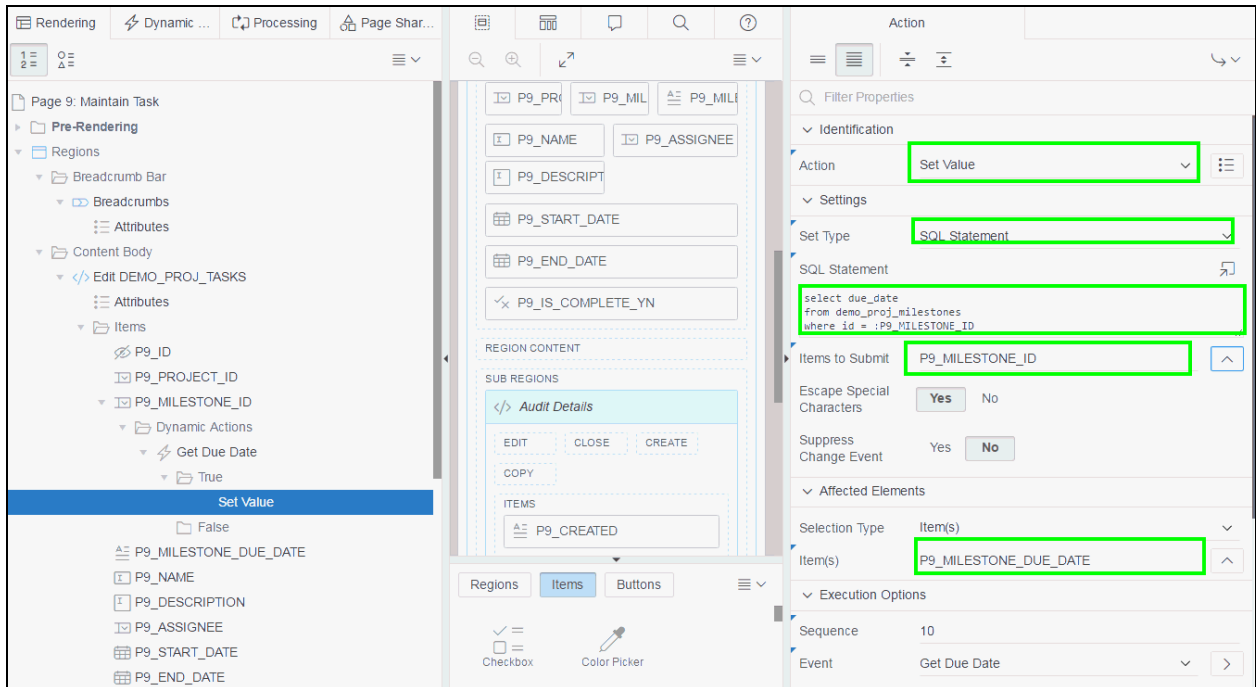
In the Property Editor:

- Identification: Action - select **Set Value**
- Settings: Set Type - select **SQL Statement**
- Settings: SQL Statement - copy and paste the following code:

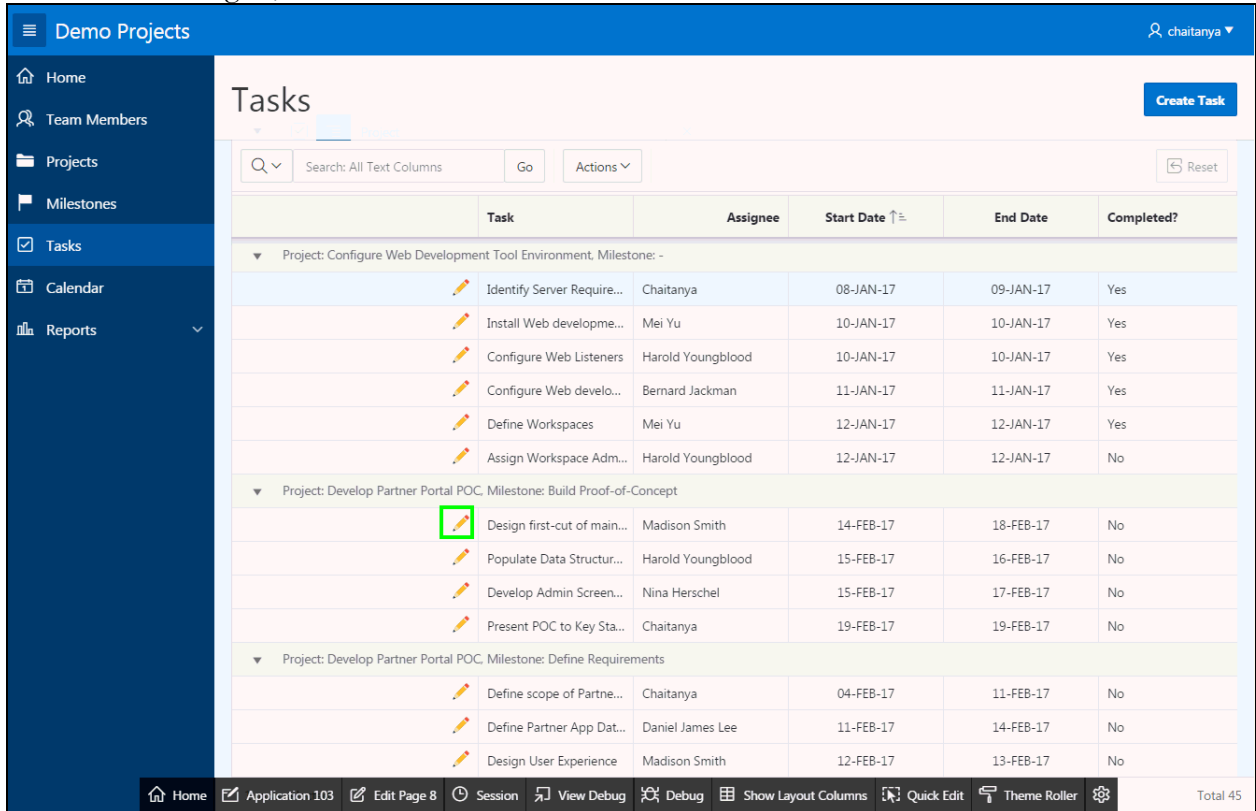
```
select due_date
from demo_milestones
where id = :P9_MILESTONE_ID
```

- Settings: Page Items to Submit - select **P9_MILESTONE_ID**
- Affected Elements: Item(s) - select **P9_MILESTONE_DUE_DATE**

Click **Save**.



- Navigate to Demo Projects application runtime environment. In the navigation menu, click Tasks. In the interactive grid, click the Edit icon for a task with milestones.



10. Try selecting different values for **Milestone** to see how the Due Date is updated based on the selection.

Tasks \

Maintain Task

* Project Milestone Due Date **20-FEB-17**

* Task

Description

Start Date

End Date

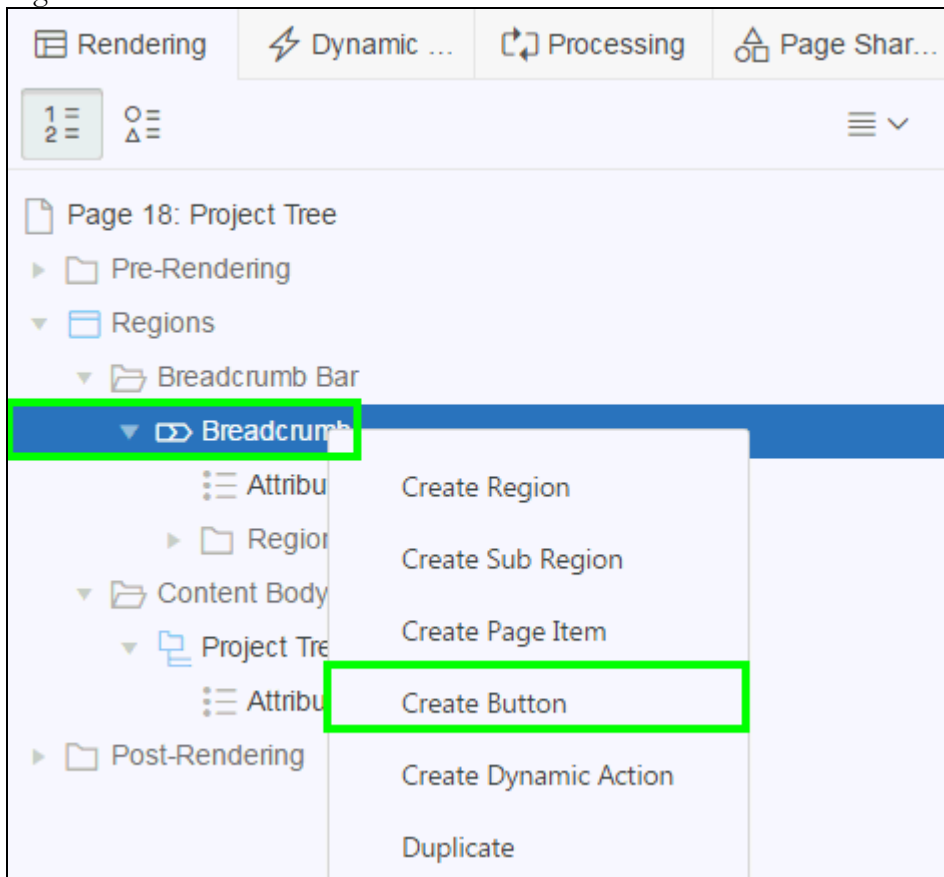
Completed?

▶ Audit Details

HOL 15-3: Creating and Using Dynamic Actions on the Project Tree Page

In this lab, you update the Project Tree page that you created in HOL 14-5. You add two buttons: Collapse All and Expand All. You use them to expand and contract the project tree nodes. You create a dynamic action on each of these buttons. Then, for each of the button, you set the action as defined by the dynamic action.

1. Navigate to Demo Projects application runtime environment. In the navigation menu, under Reports, click **Project Tree**.
In the Developer Toolbar, click **Edit Page 18**.
2. Now, to expand and collapse the project tree nodes, you want to create Expand All and Collapse All buttons on the Project Tree button. These buttons' actions are defined by dynamic actions. Under Rendering, expand Regions > Breadcrumb. Right-click **Breadcrumb** and select **Create Button**.



3. In the Property Editor:
 - Identification > Button Name: Enter **CONTRACT_ALL**
 - Identification > Label: Enter **Collapse All**
 - Layout > Button Position: Select **Edit**
 - Behavior > Action: Select **Defined by Dynamic Action**

Under Appearance, click **Template Options** button. Deselect the **Use Template Defaults** check box and click **OK**.

Button

Filter Properties

Identification

Button Name: CONTRACT_ALL

Label: Collapse All

Layout

Sequence: 10

Region: Breadcrumb

Button Position: Edit

Appearance

Button Template: Text

Hot: Yes No

Template Options: None selected

CSS Classes

Icon CSS Classes

Behavior

Action: Defined by Dynamic Action

Execute Validations: Yes No

4. Now, right-click the **CONTRACT_ALL** button and select **Duplicate**.
5. In the Property Editor:
 - Identification > Button Name: Enter **EXPAND_ALL**
 - Identification > Label: Enter **Expand All**
 - Layout > Button Position: Select **Edit**
 - Behavior > Action: Select **Defined by Dynamic Action**Under Appearance, click **Template Options** button. Deselect the **Use Template Defaults** check box and click **OK**.

Button

Filter Properties

Identification

Button Name: EXPAND_ALL

Label: Expand All

Layout

Sequence: 20

Region: Breadcrumb

Button Position: Edit

Appearance

Button Template: Text

Hot: Yes No

Template Options: None selected

CSS Classes

Icon CSS Classes

Behavior

Action: Defined by Dynamic Action

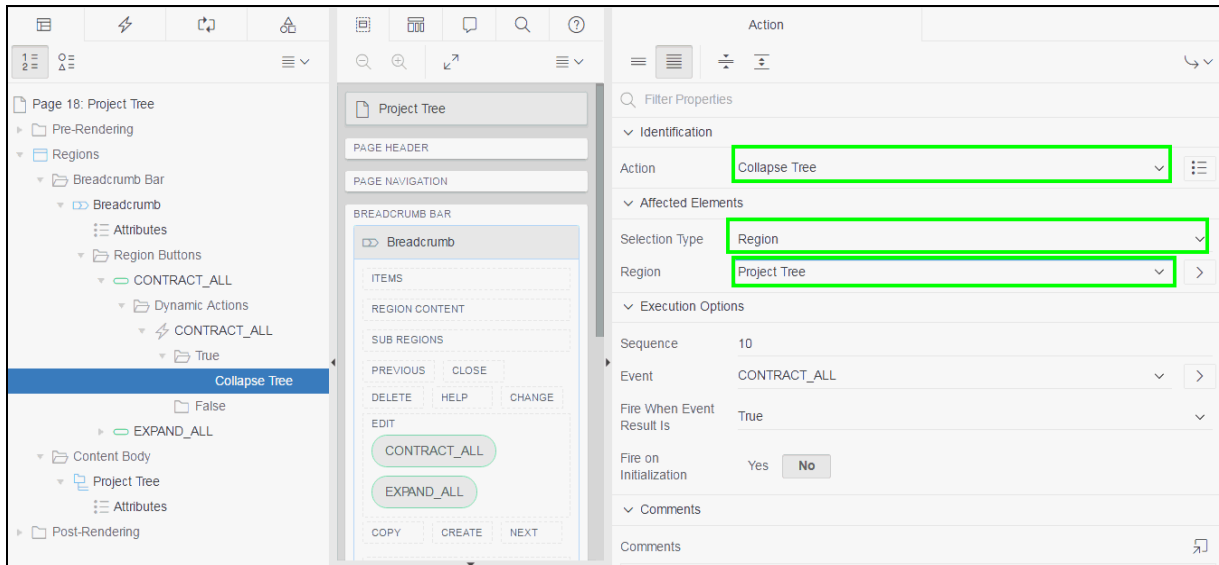
- Now you want to create a dynamic action for each of these two buttons. Under Rendering, expand Regions > Breadcrumb > Region Buttons. Right-click the **CONTRACT_ALL** button and select **Create Dynamic Action**.

7. In the property editor, for Identification > Name, enter **CONTRACT_ALL**.

The screenshot shows the 'Dynamic Action' property editor. The 'Identification' section is expanded, and the 'Name' field is highlighted with a green border, containing the text 'CONTRACT_ALL'. Below this, the 'Execution Options' section is expanded, showing 'Sequence' set to '10'. The 'When' section is also expanded, showing 'Event' set to 'Click', 'Selection Type' set to 'Button', and 'Button' set to 'CONTRACT_ALL'.

Dynamic Action	
Filter Properties	
Identification	
Name	CONTRACT_ALL
Execution Options	
Sequence	10
When	
Event	Click
Selection Type	Button
Button	CONTRACT_ALL

8. Under Rendering, navigate to the CONTRACT_ALL region button and expand Dynamic Actions. Select **Show**.
 In the property editor, for Identification > Action, select **Collapse Tree**.
 For Affected Elements > Selection Type, select **Region**.
 For Affected Elements > Region, select **Project Tree**.



9. Under Rendering, expand Regions > Breadcrumb > Region Buttons. Right-click the **EXPAND_ALL** button and select **Create Dynamic Action**.

10. In the property editor, for Identification > Name, enter **EXPAND_ALL**.



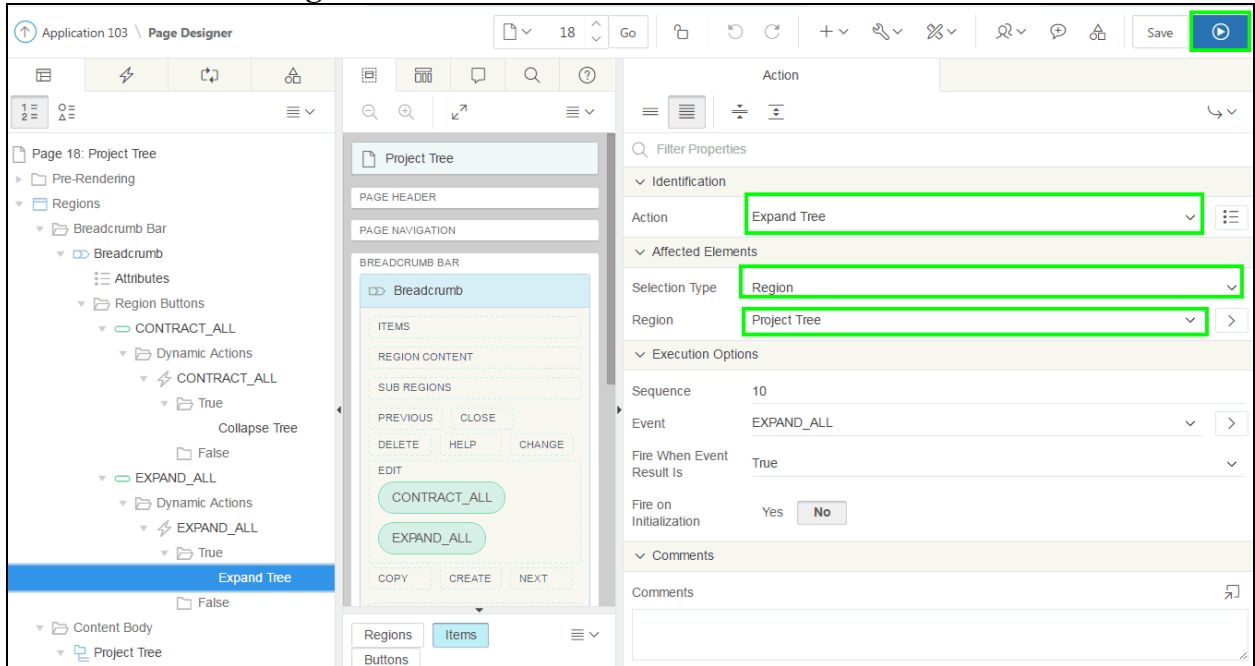
11. Under Rendering, navigate to the EXPAND_ALL region button and expand Dynamic Actions. Select **Show**.

In the property editor, for Identification > Action, select **Expand Tree**.

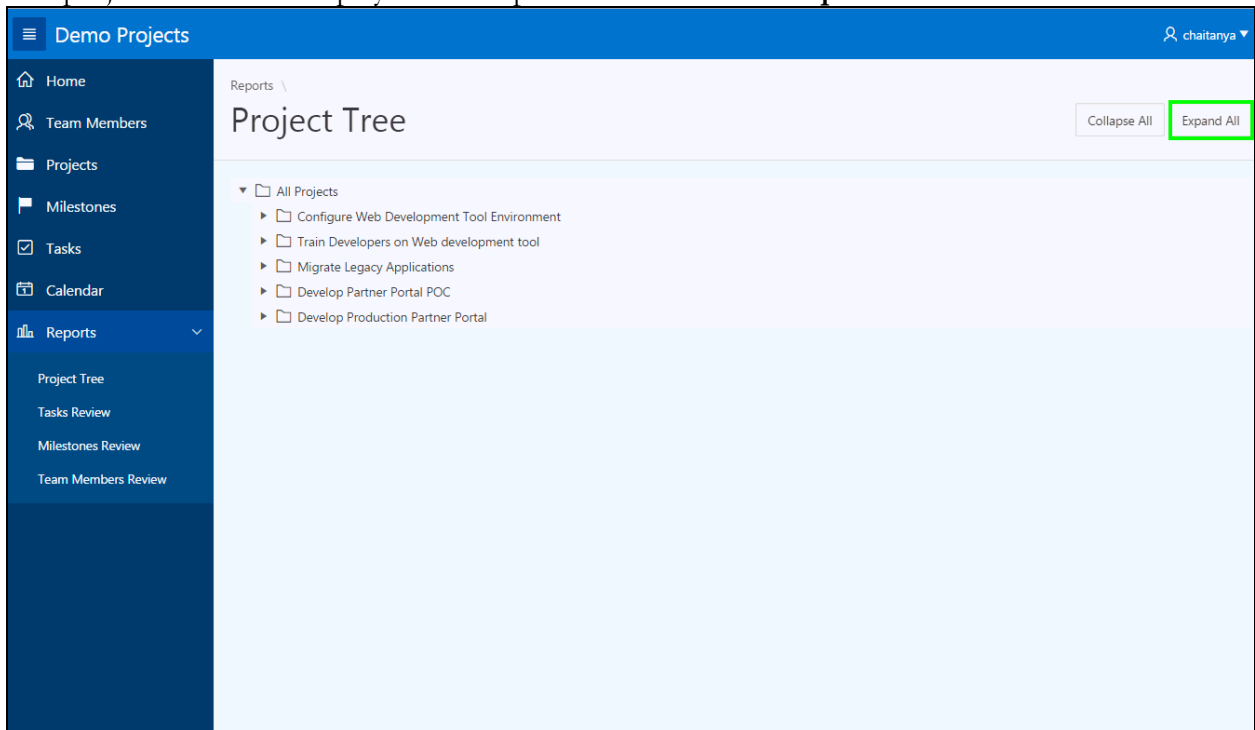
For Affected Elements > Selection Type, select **Region**.

For Affected Elements > Region, select **Project Tree**.

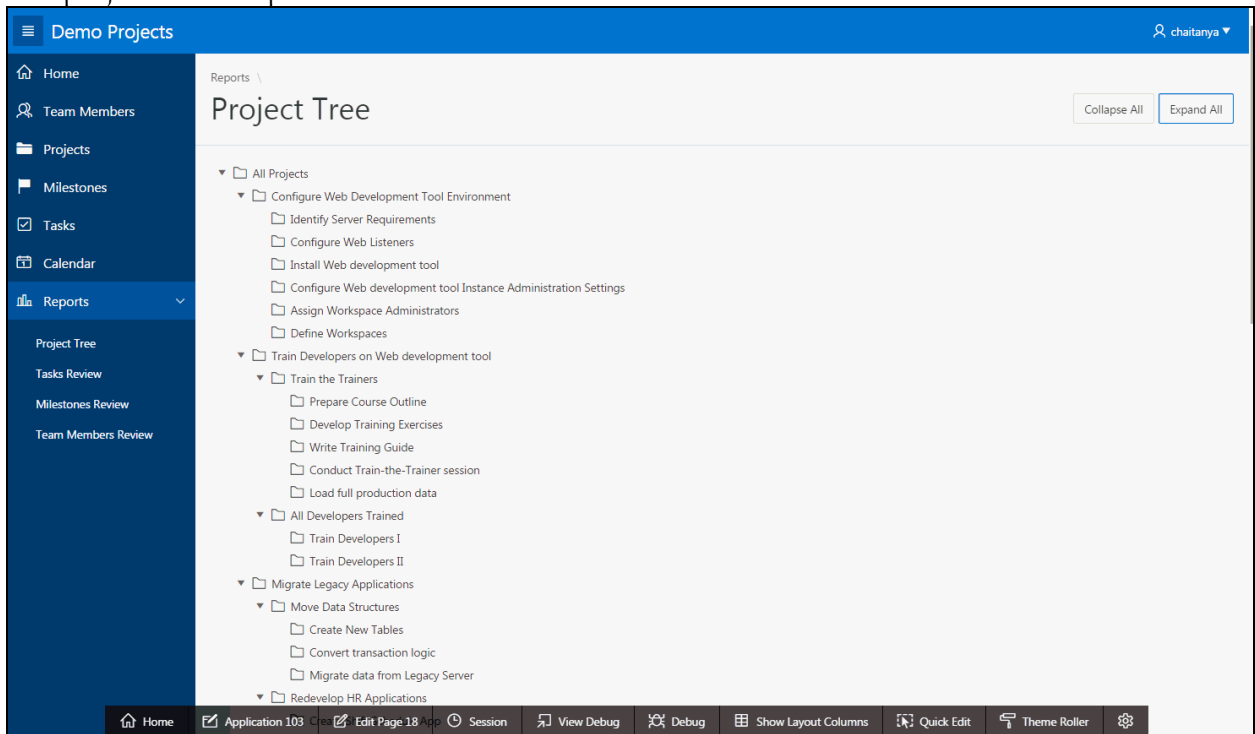
Click **Save and Run Page**.



12. The project tree is now displayed in collapsed mode. Click the **Expand All** button.



13. The project tree is expanded now.

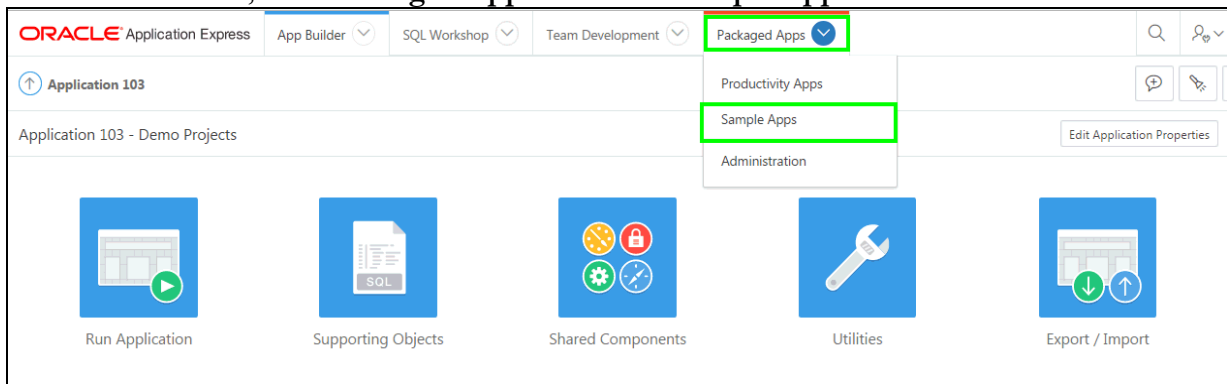


14. In the Developer Toolbar, click **Application<n>**.

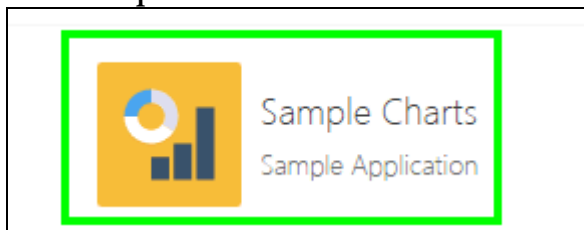
HOL 15-4: Creating and Using a Plug-in

In this hands-on lab, you create the Badge List Plug-in as a copy of the plug-in from the Sample Charts application. On the application home page, you create a badge list region. On this region, you display three different badges: Open Projects, Upcoming Milestones, and Open Tasks.

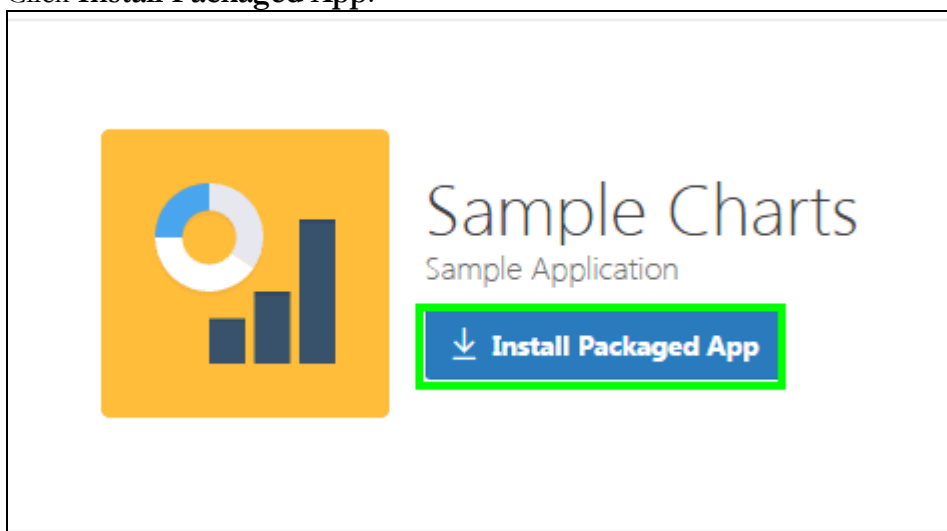
1. Install the Sample Charts application so that you can subscribe to the badge list plug-in. In the main toolbar, click **Packaged Apps** and select **Sample Apps**.



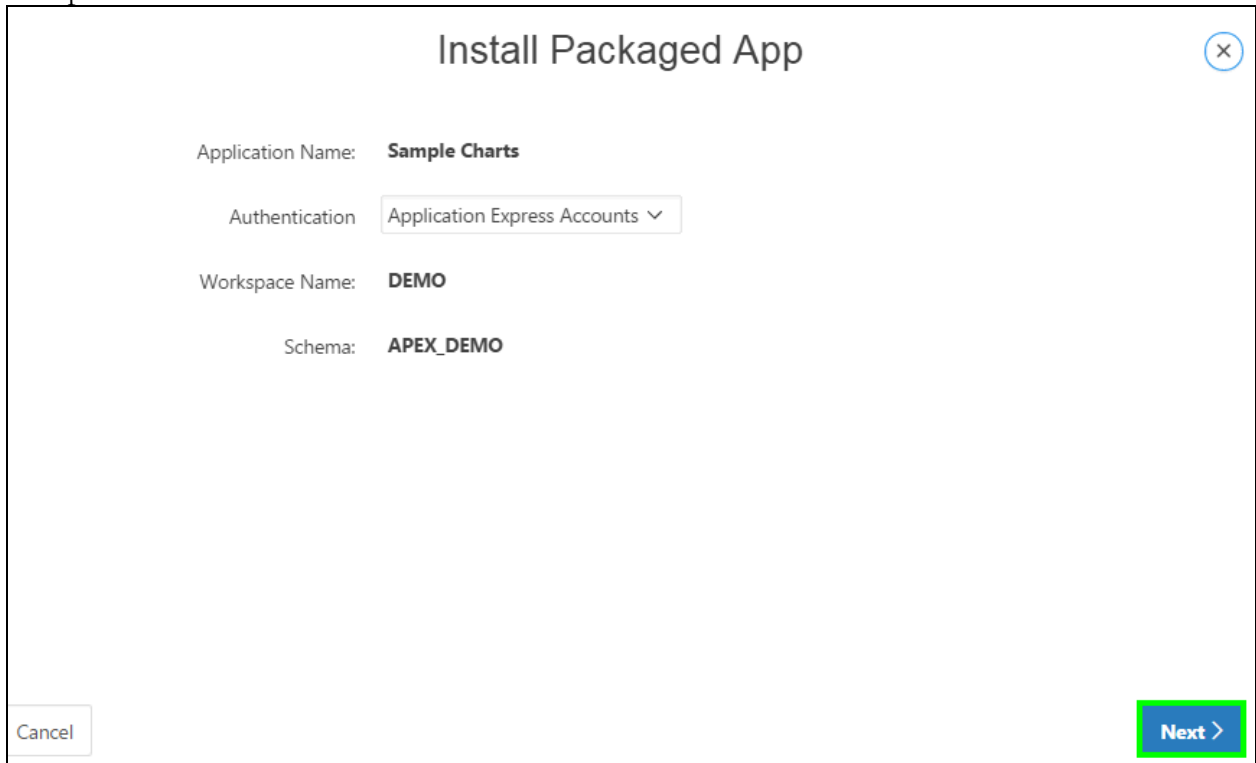
2. Click **Sample Charts**.



3. Click **Install Packaged App**.



4. Accept the default authentication and click **Next**.



Install Packaged App

Application Name: **Sample Charts**

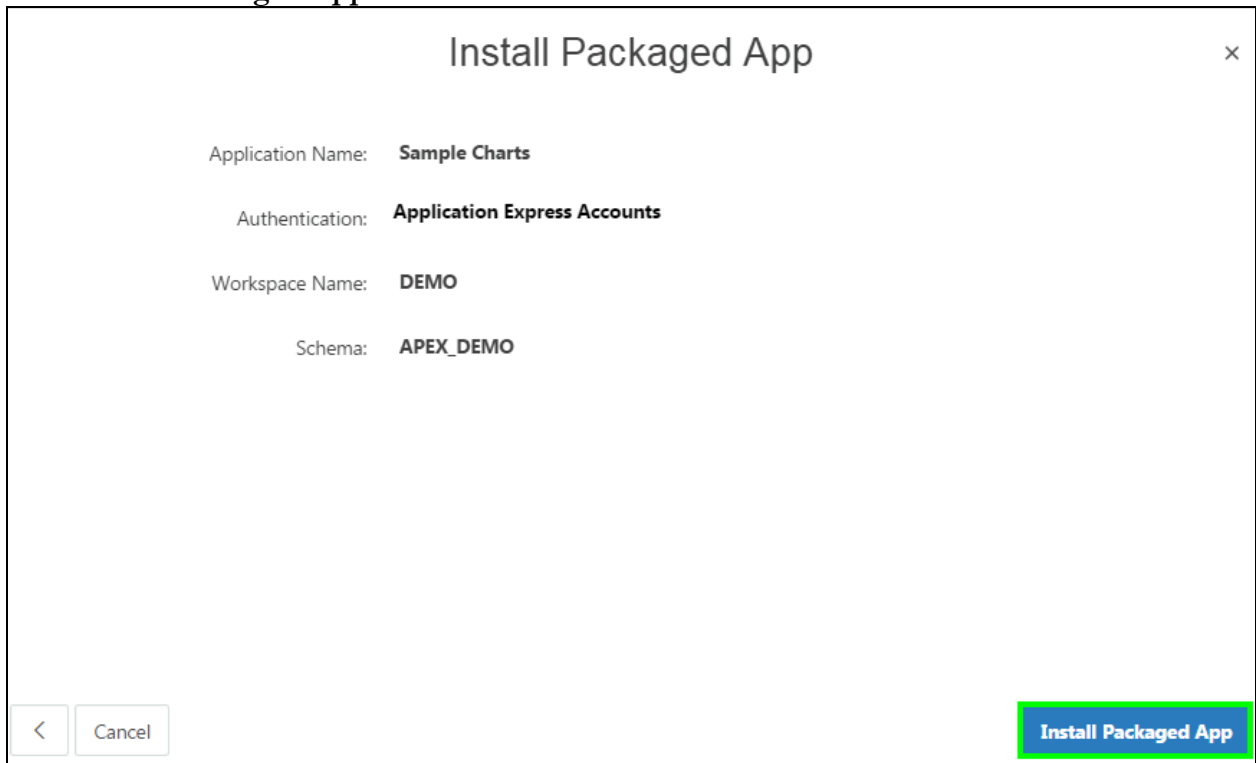
Authentication: Application Express Accounts ▾

Workspace Name: **DEMO**

Schema: **APEX_DEMO**

Cancel **Next >**

5. Click **Install Packaged App**.



Install Packaged App

Application Name: **Sample Charts**

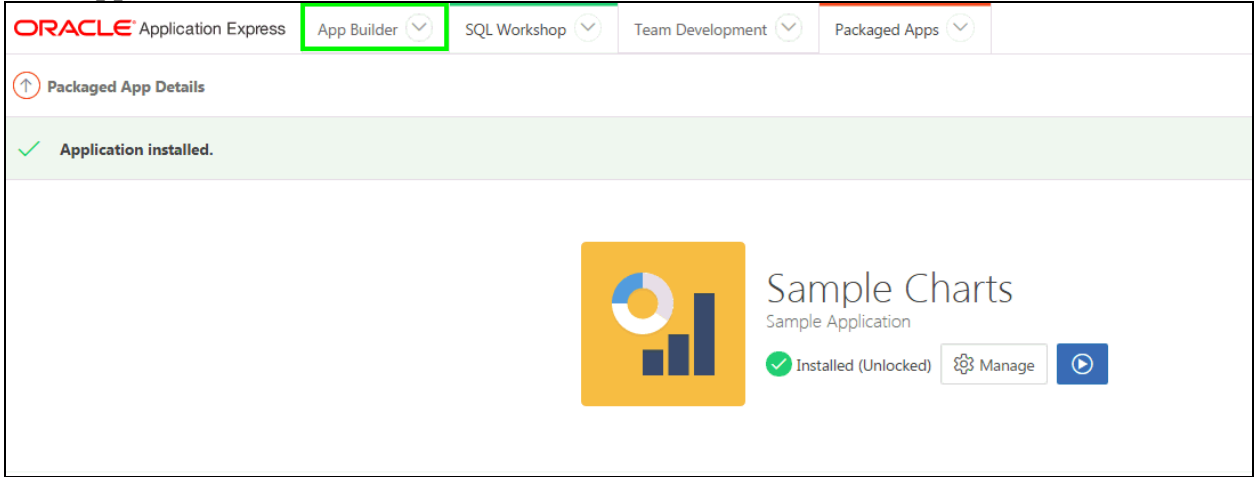
Authentication: **Application Express Accounts**

Workspace Name: **DEMO**

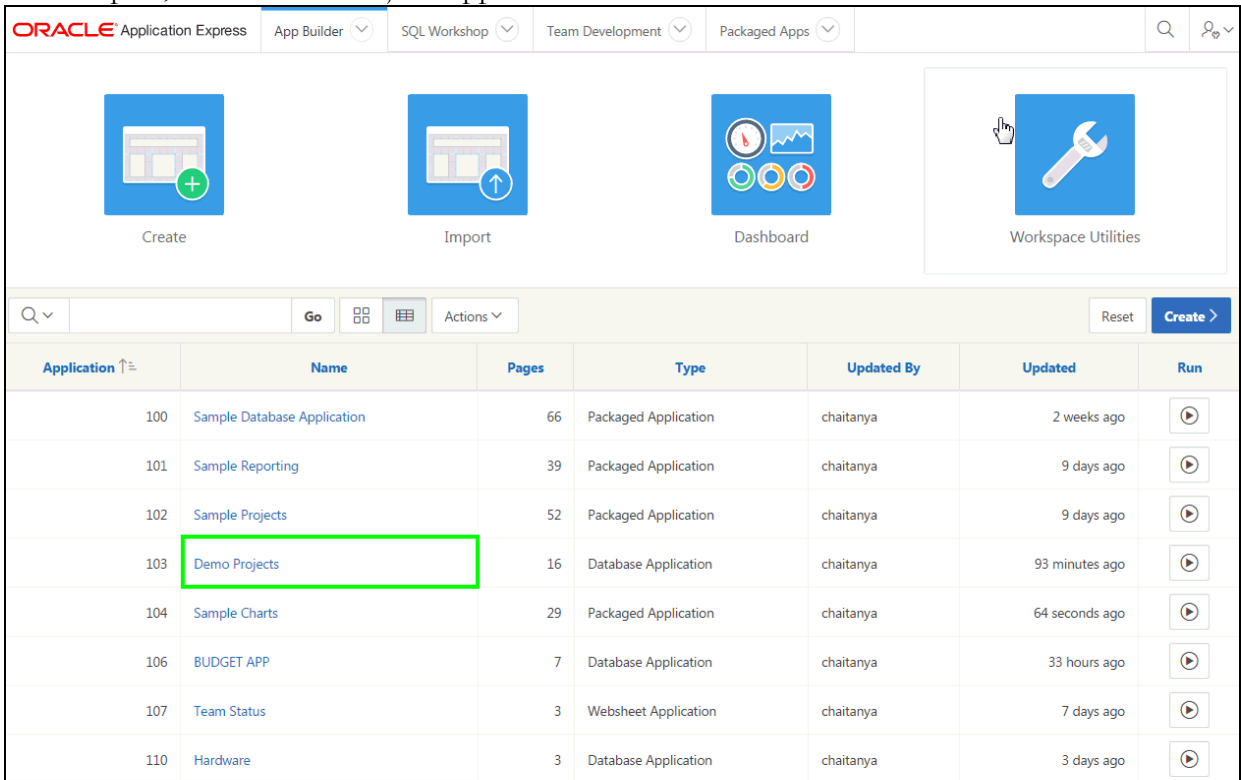
Schema: **APEX_DEMO**

< Cancel **Install Packaged App**

- The Sample Charts application is application.
Click **App Builder**.

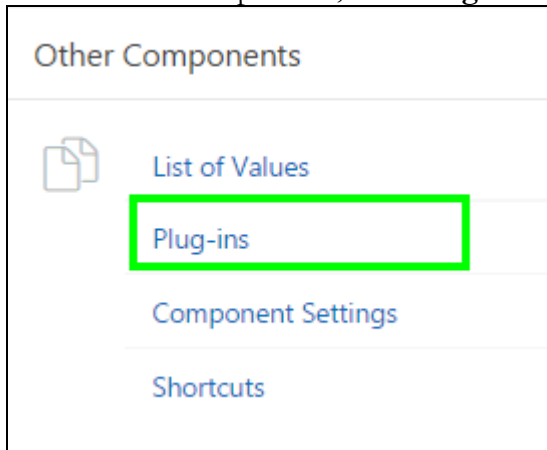


- In the report, select Demo Projects application.

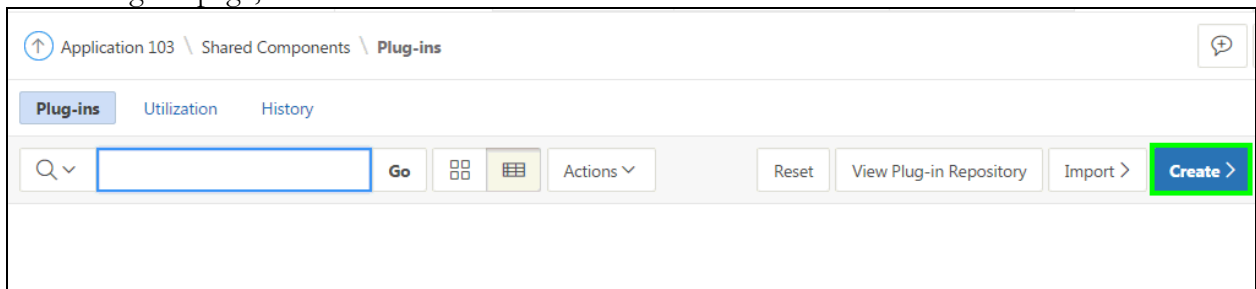


- In the application home page, click **Shared Components**.

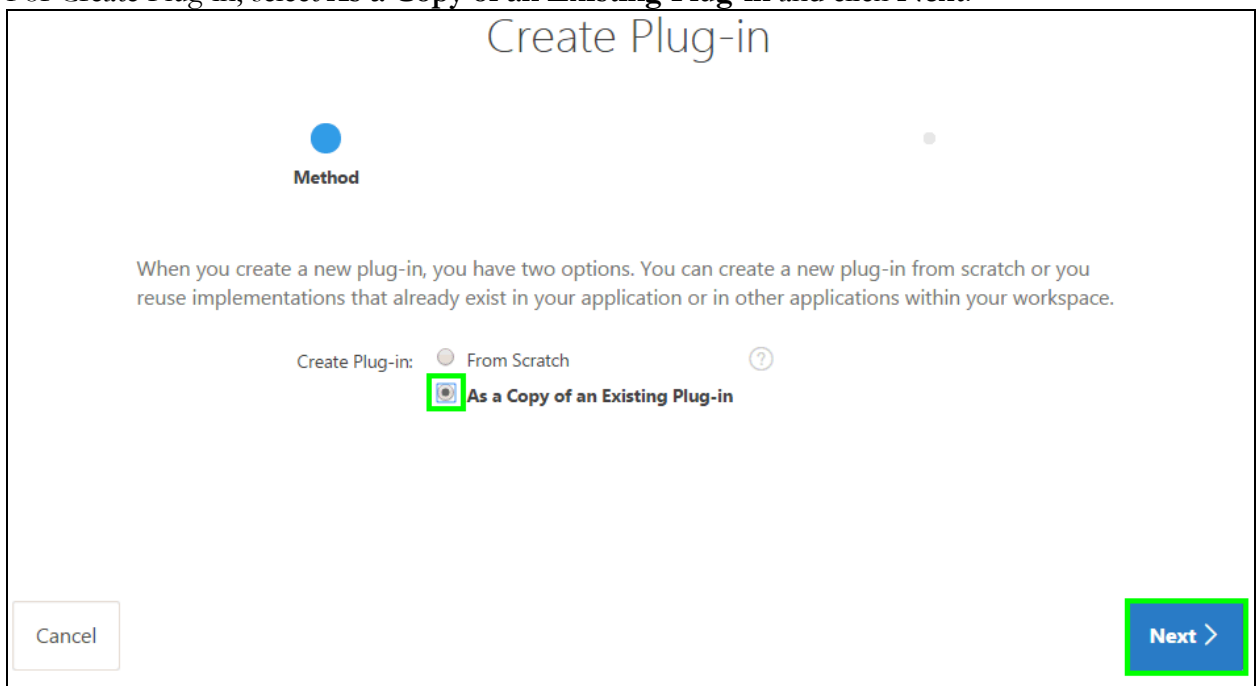
9. Under Other Components, click **Plug-ins**.



10. In the Plug-ins page, click **Create**.



11. For Create Plug-in, select **As a Copy of an Existing Plug-in** and click **Next**.



12. For Copy From Application, select <xxx>**Sample Charts** and click **Next**.

The screenshot shows the 'Create Plug-in' wizard in Oracle APEX. The title 'Create Plug-in' is centered at the top. Below the title are three progress indicators: a green checkmark, a blue circle, and a grey dot. The blue circle is labeled 'Copy From'. Below the progress indicators, the text reads 'Use this page to copy a plug-in from another application.' Underneath, there is a label '* Copy From Application' followed by a dropdown menu. The dropdown menu is highlighted with a red border and contains the text '104 Sample Charts' with a downward arrow. To the right of the dropdown is a help icon (a question mark in a circle). At the bottom left, there are two buttons: a back arrow and a 'Cancel' button. At the bottom right, there is a blue 'Next >' button, which is highlighted with a red border.

13. At the bottom of the page, click the **Deselect All** button.

Identify the Plug-in to be copied and select the copy option. Copy options include:

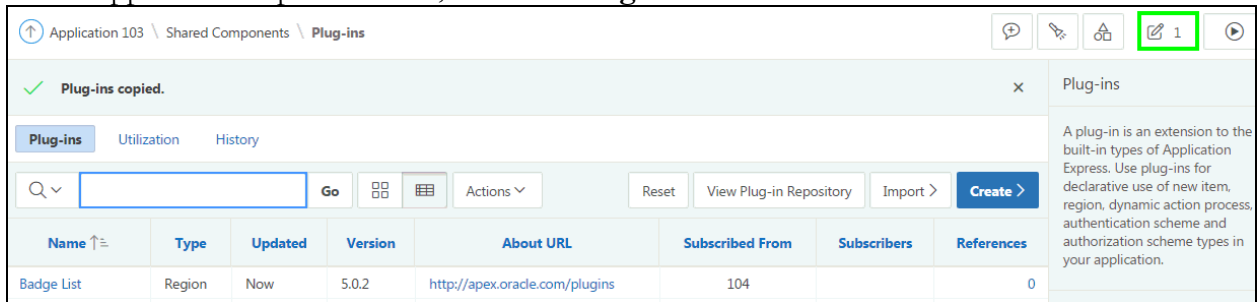
- **Yes.** Copy the Plug-in.
- **No.** Do not copy this Plug-in.
- **Copy and Subscribe.** Subscribing to a Plug-in puts a copy of it in your application and sets the plugin master reference.

Type	Name	Subscribed	To Name	Copy?
Region	Badge List		<input type="text" value="Badge List"/>	Copy and Subscribe ▼
Region	D3 Bubble Chart		<input type="text" value="D3 Bubble Chart"/>	Copy and Subscribe ▼
Region	D3 Collapsible Tree Chart		<input type="text" value="D3 Collapsible Tree Chart"/>	Copy and Subscribe ▼
Region	D3 Pie Chart		<input type="text" value="D3 Pie Chart"/>	Copy and Subscribe ▼
Region	D3 Sunburst Chart		<input type="text" value="D3 Sunburst Chart"/>	Copy and Subscribe ▼
Region	D3 Treemap Chart		<input type="text" value="D3 Treemap Chart"/>	Copy and Subscribe ▼
Region	Flot Pie Chart		<input type="text" value="Flot Pie Chart"/>	Copy and Subscribe ▼
Region	Gantt Chart		<input type="text" value="Gantt Chart"/>	Copy and Subscribe ▼
Region	HTML 5 Bar Chart		<input type="text" value="HTML 5 Bar Chart"/>	Copy and Subscribe ▼
Region	JustGage Gauge		<input type="text" value="JustGage Gauge"/>	Copy and Subscribe ▼
Region	Sample Apps Footer		<input type="text" value="Sample Apps Footer"/>	Copy and Subscribe ▼
Region	Slide Tooltip		<input type="text" value="Slide Tooltip"/>	Copy and Subscribe ▼
Region	Source Display		<input type="text" value="Source Display"/>	Copy and Subscribe ▼

14. Now, for Badge List, select **Copy and Subscribe**.
Then, click **Copy Plug-ins**.

Type	Name	Subscribed	To Name	Copy?
Region	Badge List		<input type="text" value="Badge List"/>	Copy and Subscribe ▼

15. In the Application Express toolbar, click **Edit Page 1**.

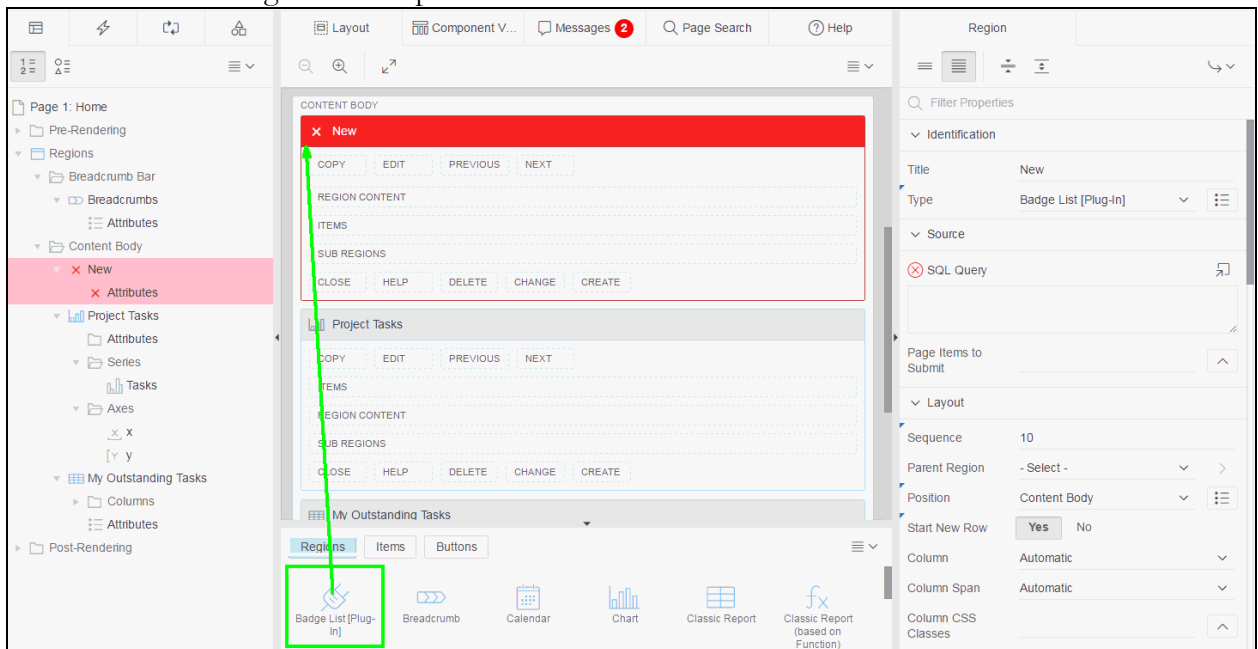


16. Use Page Designer's drag-and-drop functionality to quickly add a Badge List from the Gallery to the Layout.

In Page Designer, within the Gallery (directly below the Layout), click **Regions**, and locate **Badge List**.

Click and hold **Badge List [Plug-In]** and drag it into **Content Body**, to a position directly above the Project Tasks region. It should appear as a darkened tile before you drop it into place.

Note: When you drag the region up, and hover over the small yellow section, below Content Body, the yellow section will expand. A darker yellow section, with a black box around it, will indicate where the region will be placed.

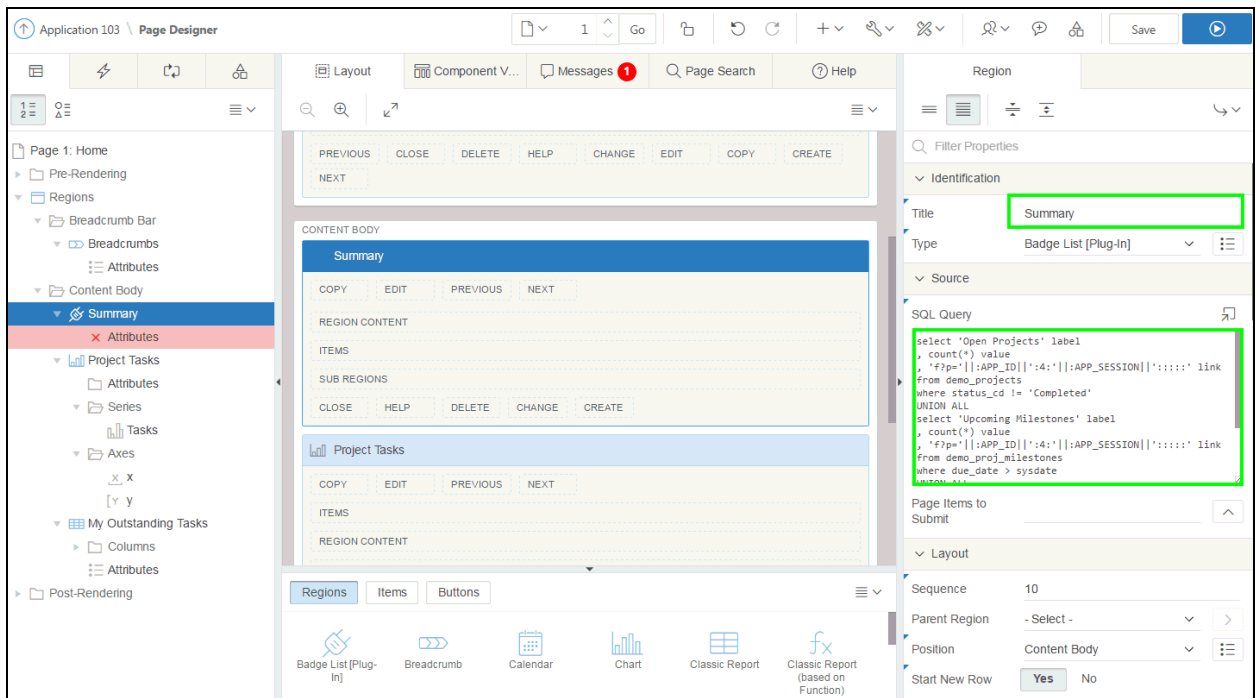


17. In the Property Editor:

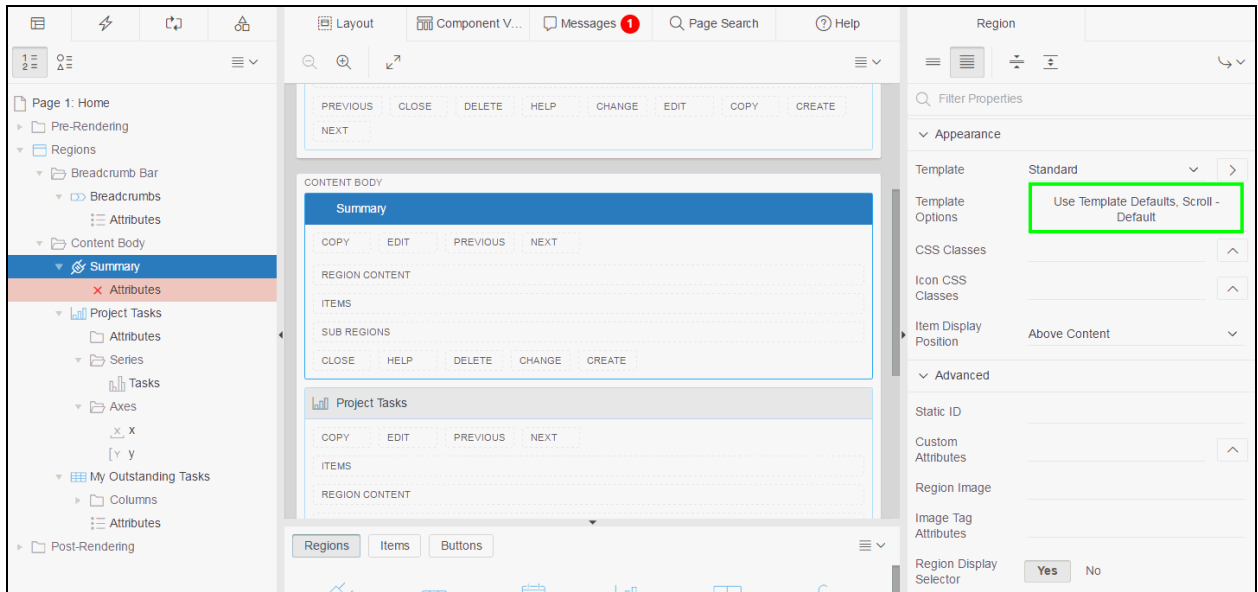
- Identification: Title - enter **Summary**
- Source: SQL Query - copy and paste the following:

```
select 'Open Projects' label
, count(*) value
, 'f?p=||:APP_ID||:4:':||:APP_SESSION||':::::' link
from demo_projects
where status_cd != 'Completed'
UNION ALL
select 'Upcoming Milestones' label
, count(*) value
, 'f?p=||:APP_ID||:4:':||:APP_SESSION||':::::' link
from demo_proj_milestones
where due_date > sysdate
UNION ALL
select 'Open Tasks' label
, count(*) value
, 'f?p=||:APP_ID||:4:':||:APP_SESSION||':::::' link
from demo_proj_tasks
where is_complete yn = 'N'
```

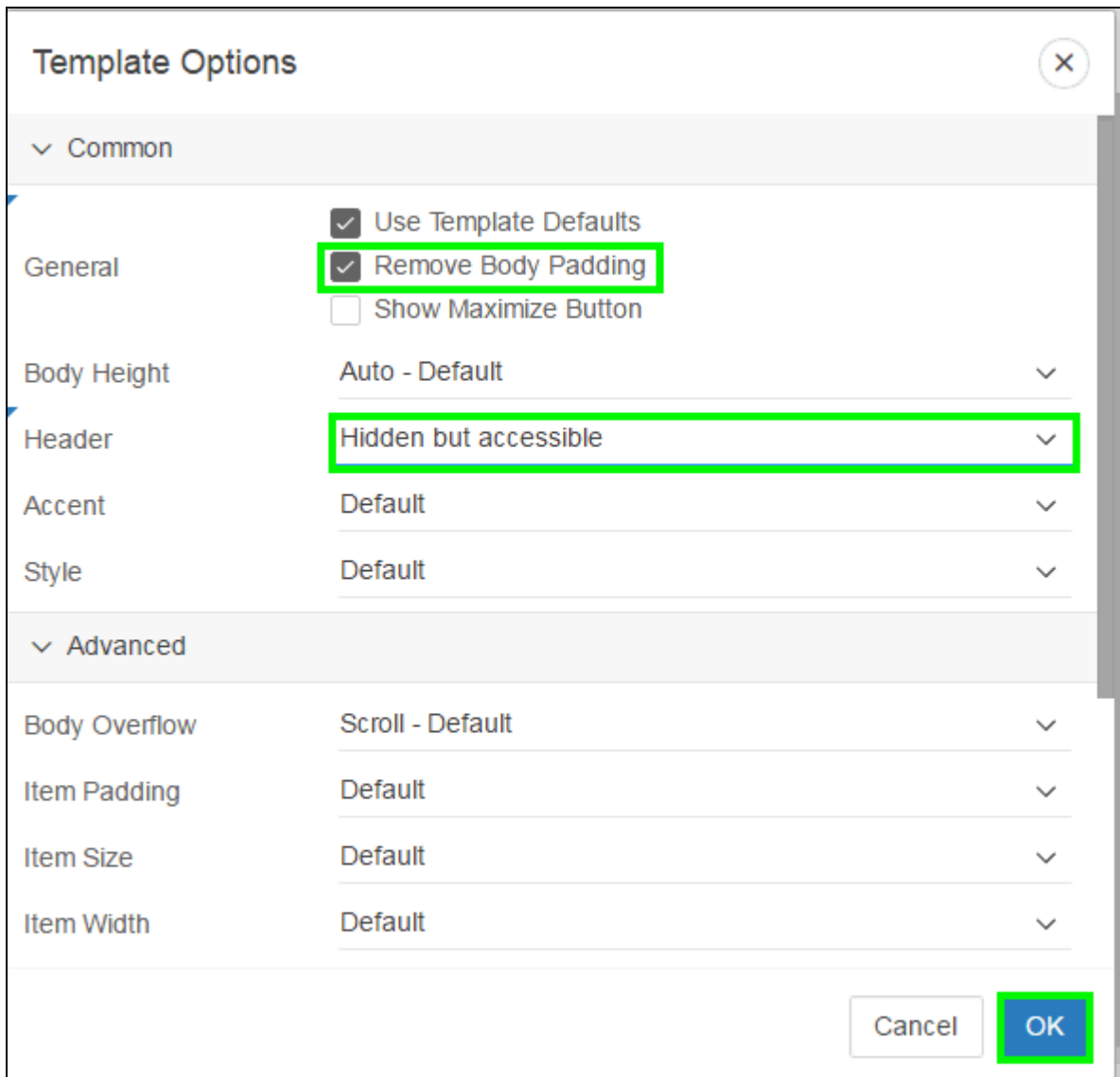
Note: This query has three sub-queries, separated by UNION ALL that will be displayed as three separate badges within the region.



18. In the property editor, navigate to Appearance and click **Template Options**.



19. In the Template Options dialog, select the **Remove Body Padding** check box.
For Header, select **Hidden but accessible**.
Click **OK**.



Template Options

Common

General

- Use Template Defaults
- Remove Body Padding
- Show Maximize Button

Body Height: Auto - Default

Header: Hidden but accessible

Accent: Default

Style: Default

Advanced

Body Overflow: Scroll - Default

Item Padding: Default

Item Size: Default

Item Width: Default

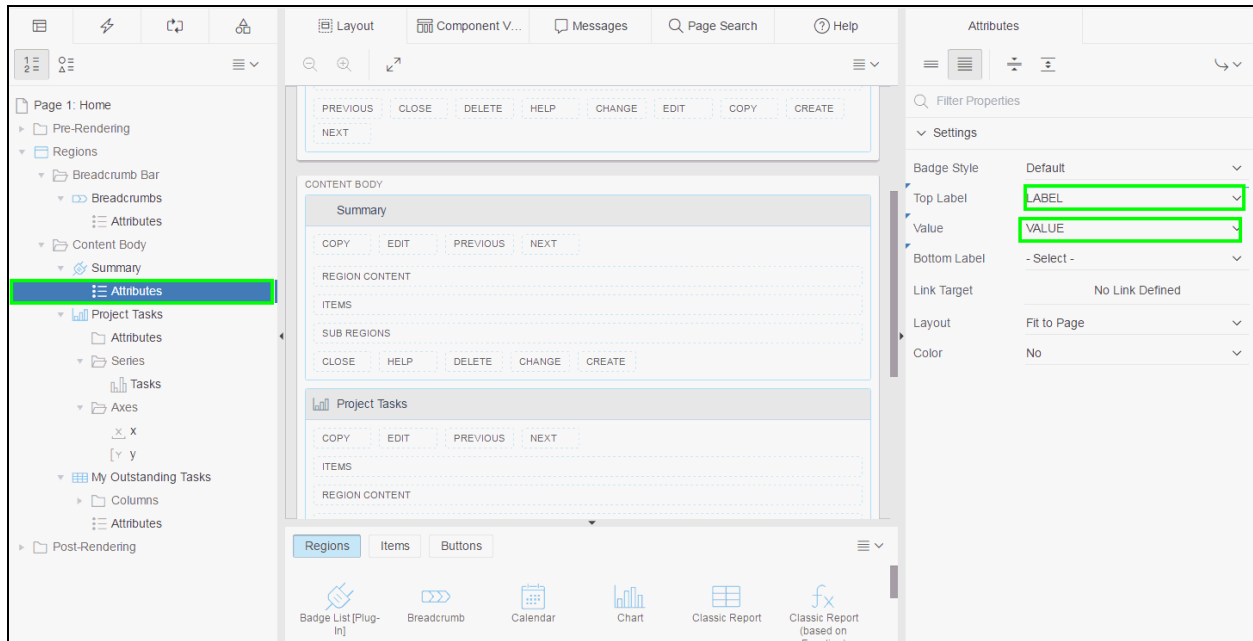
Cancel OK

20. In the Rendering tree, under the Summary region, click Attributes.

In the Property Editor:

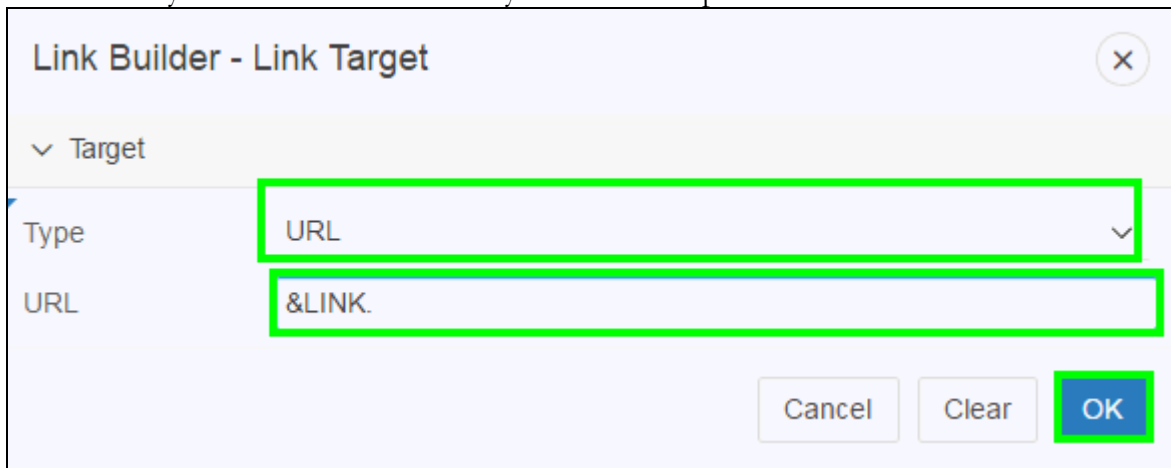
- Settings: Top Label - select **LABEL**
- Settings: Value - select **VALUE**

For Link Target, click **No Link Defined**.



21. In the Link Builder – Link Target dialog, select URL for Type. For URL, enter **&LINK.**. Click **OK**.

Note: When you enter **&LINK.** ensure you include the period.



22. Click **Save and Run Page**.

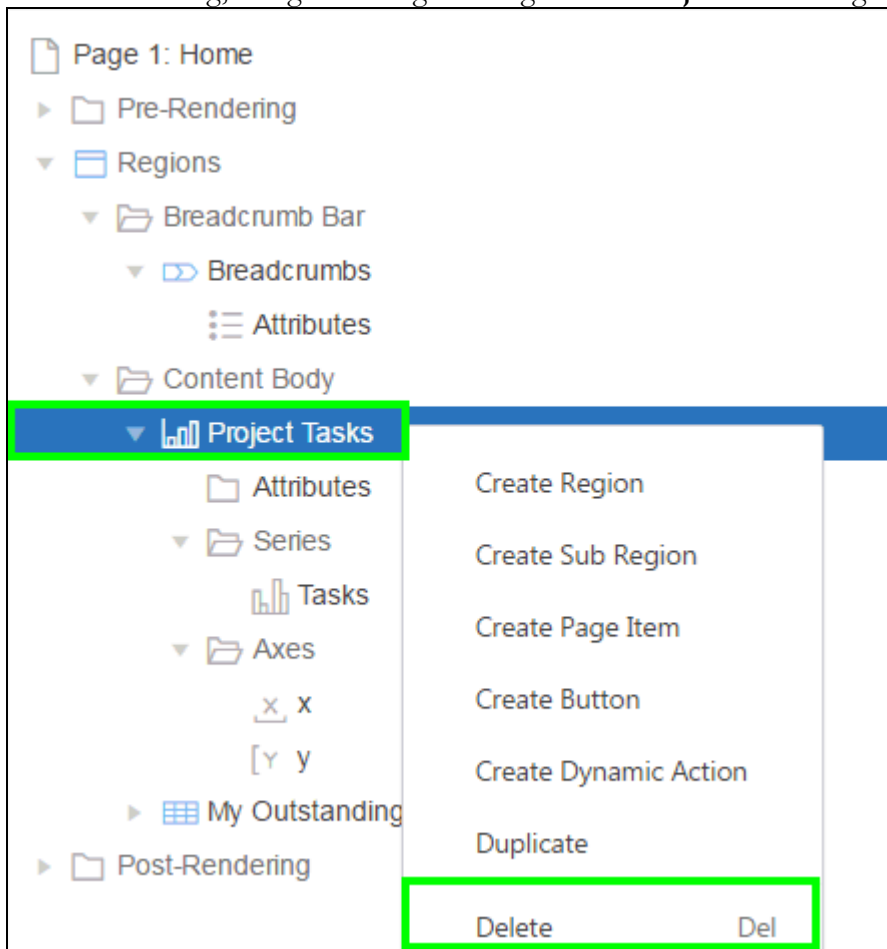
HOL 15-5: Updating the Home Page

In this lab, you create a dashboard by adding new components to the Home page of the Demo Projects application. You already created a badge list region in HOL 15-4. This lab covers adding the Recent Projects classic report region and modifying the Summary region attributes.

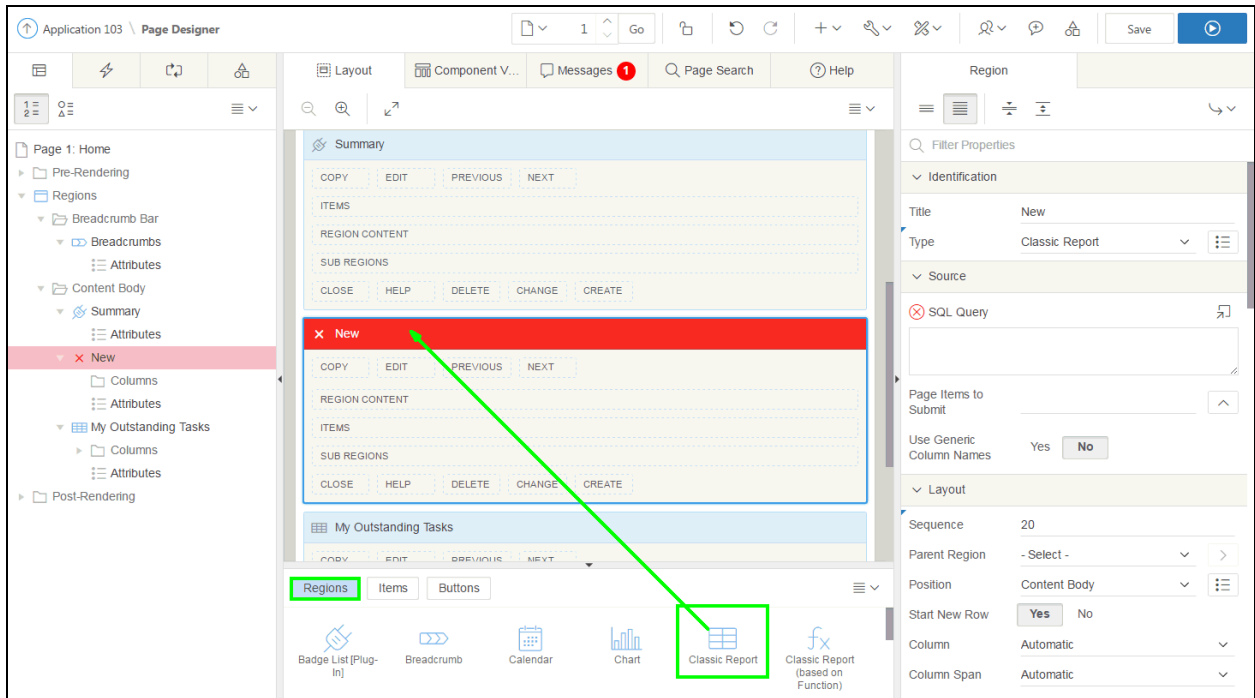
1. You already have the Project Tasks chart added as a page under Reports. Therefore, you do not need the same chart on the Home page anymore. Instead, you add a new classic report region here.

View Page 1 in page designer.

Under Rendering, navigate to Regions. Right-click **Project Tasks** region and select **Delete**.



2. Now, in the Layout, navigate to Gallery and click **Regions**. Click and hold **Classic Report** and drag it into **Content Body**, to a position directly between the Summary and the My Outstanding Tasks regions.



3. In the property editor:

- Identification > Title: Enter **Recent Projects**
- Identification > Source > SQL Query: Copy and paste the following code:

```

select p.id project_id
, t.id
, null event_modifiers
, null event_attributes
, null user_color
, dbms_lob.getlength('PHOTO_BLOB') user_avatar
, t.full_name user_name
, nvl(p.completed_date, p.updated) event_date
, (case p.status_cd
when 'ASSIGNED' then 'is-removed'
when 'IN-PROGRESS' then 'is-updated'
when 'COMPLETED' then 'is-new'
end) event_status
, (select s.description
from demo_proj_status s
where s.cd = p.status_cd
) event_type
, (case p.status_cd
when 'ASSIGNED' then 'fa fa-clock-o'
when 'IN-PROGRESS' then 'fa fa-refresh'
when 'COMPLETED' then 'fa fa-check'
end) event_icon
, p.name event_title
, p.description event_desc
from demo_projects p
, demo_proj_team_members t
where p.project_lead = t.id
order by p.updated desc

```

The screenshot displays the Oracle APEX Page Designer interface for 'Application 103 \ Page Designer'. The 'Recent Projects' region is selected in the left-hand navigation pane. The central preview area shows a summary of the page layout, including the 'Recent Projects' region. The right-hand configuration pane shows the following settings:

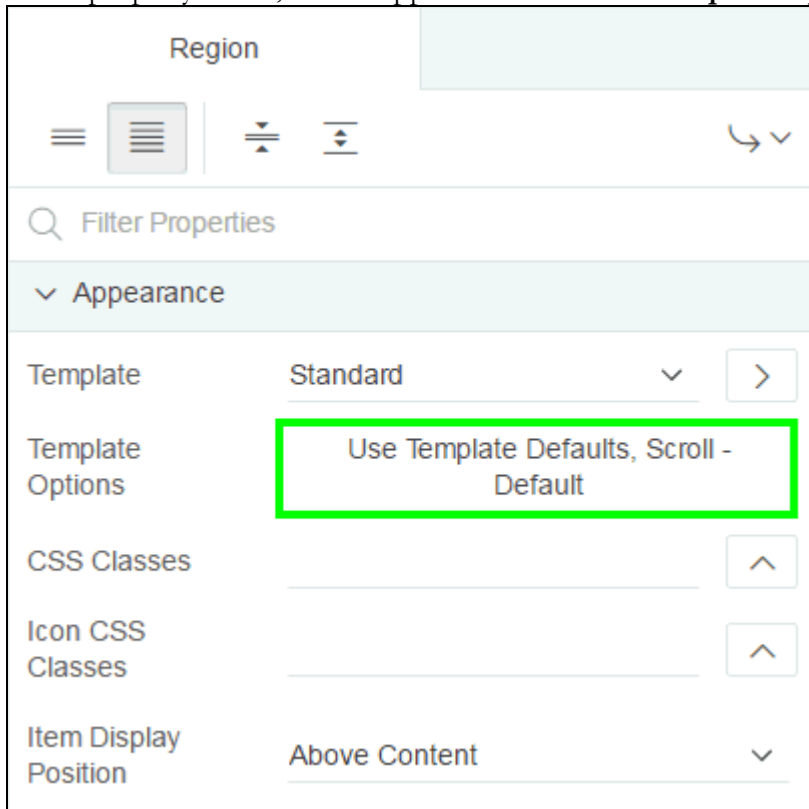
- Region:** Recent Projects
- Identification:** Title: Recent Projects, Type: Classic Report
- Source:** SQL Query:


```

select p.id project_id
, t.id
, null event_modifiers
, null event_attributes
, null user_color
, dbms_lob.getlength('PHOTO_BLOB') user_avatar
, t.full_name user_name
, nvl(p.completed_date, p.updated) event_date
, (case p.status_cd
when 'ASSIGNED' then 'is-removed'
when 'IN-PROGRESS' then 'is-updated'
when 'COMPLETED' then 'is-new'
end) event_status

```
- Page Items to Submit:** Submit
- Use Generic Column Names:** No
- Layout:** Sequence: 20, Parent Region: - Select -, Position: Content Body

4. In the property editor, locate Appearance and click **Template Options**.



5. In the Template Options dialog:
 - For General, select the Remove Body Padding check box.
 - For Body Height, select **480px**.

Click **OK**.

The image shows a 'Template Options' dialog box with the following settings:

Section	Option	Value
Common	General	<input checked="" type="checkbox"/> Use Template Defaults
		<input checked="" type="checkbox"/> Remove Body Padding
		<input type="checkbox"/> Show Maximize Button
	Body Height	480px
Advanced	Header	Visible - Default
	Accent	Default
	Style	Default
	Body Overflow	Scroll - Default
	Item Padding	Default
	Item Size	Default
	Item Width	Default

Buttons: Cancel, OK

6. In the property editor, under Advanced, select **No** for Region Display Selector.

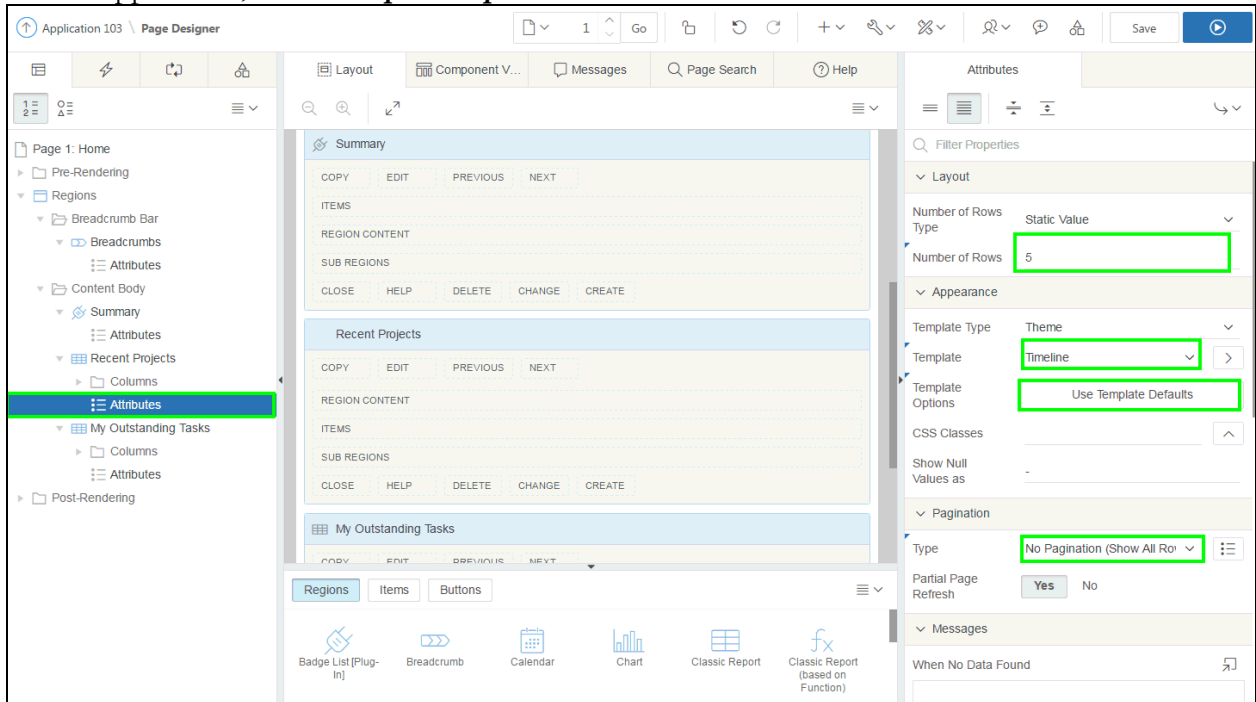
The screenshot shows the Oracle APEX Region property editor. The 'Advanced' section is expanded, and the 'Region Display Selector' property is set to 'No', which is highlighted with a green box. Other properties visible include Template (Standard), Template Options (Use Template Defaults, Remove Body Padding, 480px, Scroll - Default), CSS Classes, Icon CSS Classes, Item Display Position (Above Content), Static ID, Custom Attributes, Region Image, and Image Tag Attributes.

Region	
Filter Properties	
Appearance	
Template	Standard
Template Options	Use Template Defaults, Remove Body Padding, 480px, Scroll - Default
CSS Classes	
Icon CSS Classes	
Item Display Position	Above Content
Advanced	
Static ID	
Custom Attributes	
Region Image	
Image Tag Attributes	
Region Display Selector	Yes No
Exclude Title from Translation	Yes No

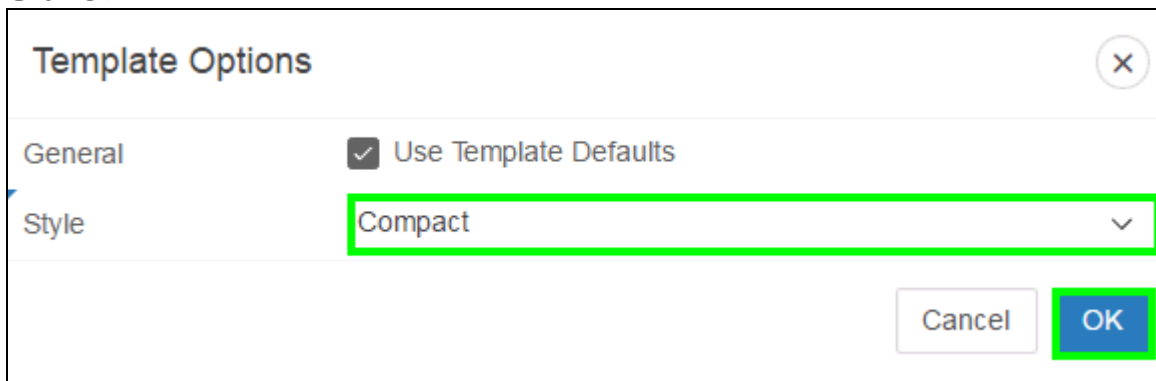
7. Under Rendering, expand Regions > Recent Projects and select **Attributes**.
In the property editor:
- Layout > Number of Rows: Enter **5**

- Appearance > Template: Select **Timeline**
- Pagination > Type: Select **No Pagination (Show All Rows)**

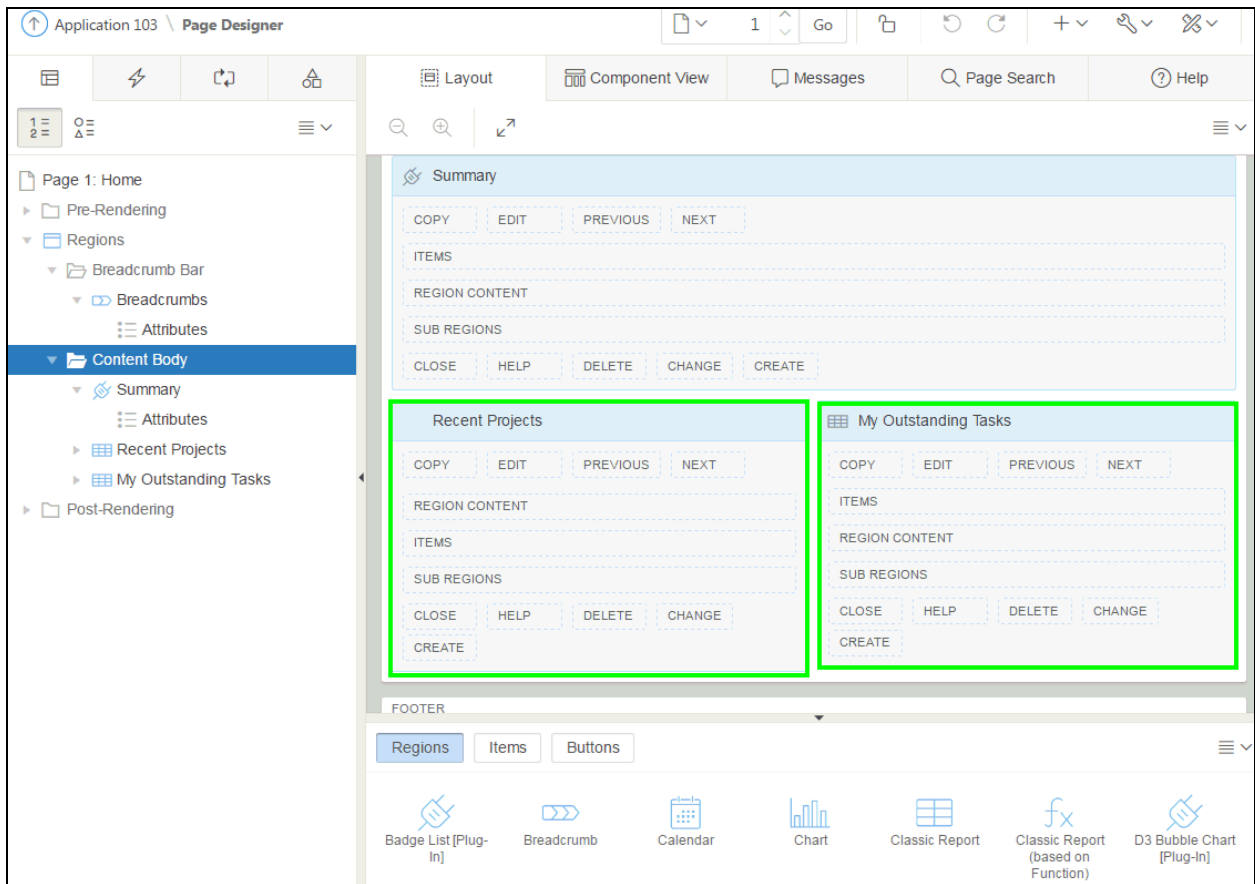
Under Appearance, click **Template Options**.



8. In the Template Options dialog, select **Compact** for Style. Click **OK**.



9. In the Layout, click and hold the **My Outstanding Tasks** region and drag it into the position next to the Recent Projects region. Both the Recent Projects and My Outstanding Tasks regions should now appear on the same line horizontally.



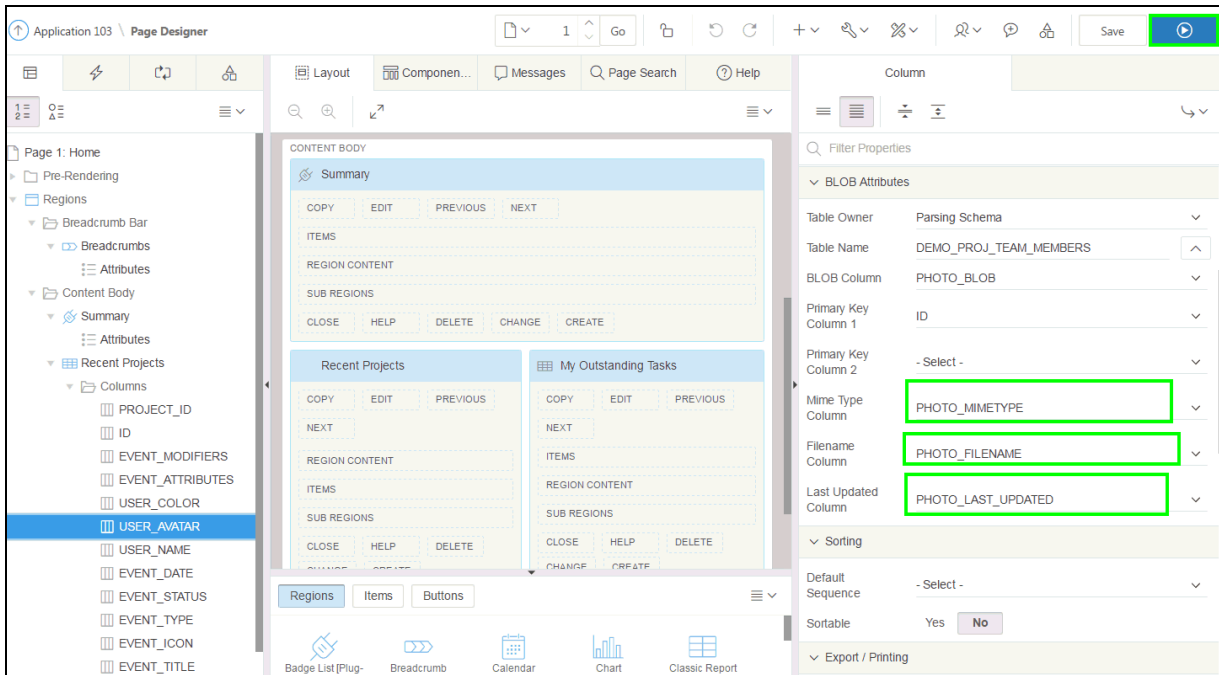
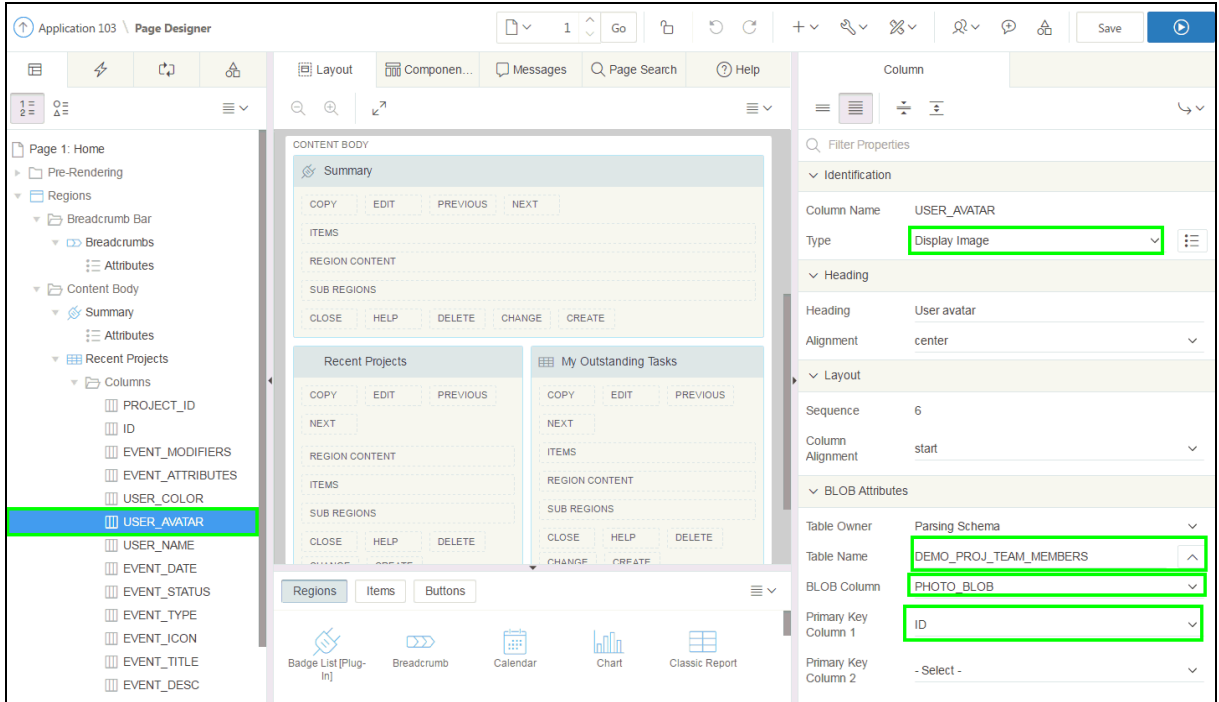
10. Under Rendering, expand **Regions > Recent Projects > Columns**.

Click **USER_AVATAR**.

In the property editor:

- Identification > Type: Select **Display Image**
- BLOB Attributes > Table Name: Select **DEMO_PROJ_TEAM_MEMBERS**
- BLOB Attributes > BLOB Column: Select **PHOTO_BLOB**
- Primary Key Column 1: Select **ID**
- BLOB Attributes > Mime Type Column: Select **PHOTO_MIMETYPE**
- BLOB Attributes > Filename Column: Select **PHOTO_FILENAME**
- BLOB Attributes > LAST_UPDATED Column: Select **PHOTO_LAST_UPDATED**

Click **Save and Run Page**.



The Home page is displayed now.

Project	Task	End date
Develop Partner Portal POC	Define scope of Partner Portal App.	11-FEB-17
Develop Partner Portal POC	Present POC to Key Stakeholders	19-FEB-17
Develop Production Partner Portal	Define production scope of Partner Portal App.	21-FEB-17
Develop Production Partner Portal	Review Beta Feedback	08-MAR-17
Develop Production Partner Portal	Provide user credentials for partners	07-APR-17

11. You can set color for the badges.

Navigate to page designer and under Rendering, expand Regions > Summary. Then, select **Attributes**.

In the property editor, under Settings, for Color, select **Yes**.

Click **Save and Run Page**.

12. The Home page now is displayed with the colored badges.

The screenshot shows the Oracle APEX Home page for a user named Chaitanya. The page features a navigation sidebar on the left with options like Home, Team Members, Projects, Milestones, Tasks, Calendar, and Reports. The main content area is titled 'Home' and includes three summary cards: 'Open Projects' (5), 'Upcoming Milestones' (6), and 'Open Tasks' (28). Below these are two main sections: 'Recent Projects' and 'My Outstanding Tasks'.

Recent Projects

- Lucille Beatie** (08-FEB-17) - **Assigned**: Develop Production Partner Portal. Description: Develop the production app that partners can use to work more collaboratively with us.
- Chaitanya** (08-FEB-17) - **In-Progress**: Develop Partner Portal POC. Description: Develop a proof of concept that partners can use to work more collaboratively with us.
- Miyazaki Yokohama** (08-FEB-17) - **In-Progress**: Migrate Legacy Applications. Description: Move the data and redevelop the applications currently running on top of legacy servers.
- Lucille Beatie** (27-JAN-17) - **Completed**: Train Developers on Web development tool. Description: Ensure all developers who will be developing with the new tool get the appropriate training.
- Chaitanya** (12-JAN-17) - **Completed**: Configure Web Development Tool Environment. Description: Determine the hardware and software required to develop with Web development tool.

My Outstanding Tasks

Project	Task	End date
Develop Partner Portal POC	Define scope of Partner Portal App.	11-FEB-17
Develop Partner Portal POC	Present POC to Key Stakeholders	19-FEB-17
Develop Production Partner Portal	Define production scope of Partner Portal App.	21-FEB-17
Develop Production Partner Portal	Review Beta Feedback	08-MAR-17
Develop Production Partner Portal	Provide user credentials for partners	07-APR-17

1 - 5

The bottom of the page shows a navigation bar with icons for Home, Application 103, Edit Page 1, Session, View Debug, Debug, Show Layout Columns, Quick Edit, and Theme Roller.



Oracle is committed to developing practices and products that help protect the environment

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Copyright © 2017, Oracle and/or its affiliates. All rights reserved. Licensed under the Creative Commons Attribution 4.0 International License as shown at <https://creativecommons.org/licenses/by/4.0/legalcode>.

This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Hardware and Software, Engineered to Work Together