



Hands-on Lab



Oracle **APEX**



Unit 10: Adding Computations, Processes, and Validations

Lesson Objectives

After completing this lesson, you should be able to:

- Explain page rendering and page processing
- Create computations
- Create page processes
- Create validations
- Create branches

Understanding Page Rendering and Page Processing

When you run an application, Application Express engine relies on two processes:

- Show Page (Page Rendering)
- Accept Page (Page Processing)

Products

Show Page

The screenshot shows a table with columns: Image, Name, Category, Available, Price, Units, Sales, and Customers. The data includes:

Image	Name	Category	Available	Price	Units	Sales	Customers
	Bag	Accessories	Yes	\$125.00	16	\$2,000.00	6
	Belt	Accessories	Yes	\$30.00	11	\$330.00	3
	Blouse	Womens	Yes	\$60.00	16	\$960.00	5
	Business Shirt	Mens	Yes	\$50.00	23	\$1,150.00	5

Add Product

Sample Database Application - Product Details

Accept Page

The screenshot shows a form with fields:

- * Product Name: Text input field.
- Product Description: Text area.
- * Category: Select dropdown with options Mens and Womens.
- * Product Available: Radio button group with Yes selected and No as an option.
- * List Price: Text input field.
- Product Image: File input field labeled "Choose file".
- Tags: Text input field.

Cancel Add Product

Page Rendering: Example

Employee Details

Name: KING

Job: PRESIDENT

Hire Date: 17-NOV-81

Salary: 5000

Commission:

Department: 10

	Name	Job
	KING	PRESIDENT
	BLAKE	MANAGER
	CLARK	MANAGER
	JONES	MANAGER
	SCOTT	ANALYST
	FORD	ANALYST

Page Processing: Example

	Name	Job
	KING	PRESIDENT
	BLAKE	MANAGER
	CLARK	MANAGER
	JONES	MANAGER
	SCOTT	ANALYST
	FORD	ANALYST

Employee Details

Name
KING

Job
PRESIDENT

Hire Date
17-NOV-81

Salary
5000

Commission

Department
10

What is a Computation?

- A computation is a logic that assigns values to a single item
- Two types of computations:
 - ✓ Application Computation
 - ✓ Page Computation
- A page computation assigns a value to an identified item when a page is displayed or submitted (rendered and processed).

Creating a Page Computation: Steps

1. Navigate to the appropriate page in Page Designer.
2. In the Rendering tab, expand the Pre-Rendering or Post-Rendering node and locate where the computation should be.
3. Right-click the location and select Create Computation..
4. Edit the attributes in the property editor
5. Click Save

Creating a Page Computation: Example

1

Customer Name	Address	City
Dulles, John	45020 Aviation Drive	Sterling
Hartsfield, William	6000 North Terminal Parkway	Atlanta
Logan, Edward	1 Harborside Drive	East Boston
OHare, Frank	10000 West OHare	Chicago
LaGuardia, Fiorello	Hangar Center, Third Floor	Flushing
Lambert, Albert	10701 Lambert International Blvd.	St. Louis
Bradley, Eugene	Schoephoester Road	Windsor Locks

2

First Name * John Last Name * Dulles

Street Address 45020 Aviation Drive Line 2

City Sterling State * Virginia

Zip Code * 20166

Credit Limit * 1000

Phone Number 703-555-2143 Alternate Number 703-555-8967

Email john.dulles@email.com URL http://www.johndulles.com

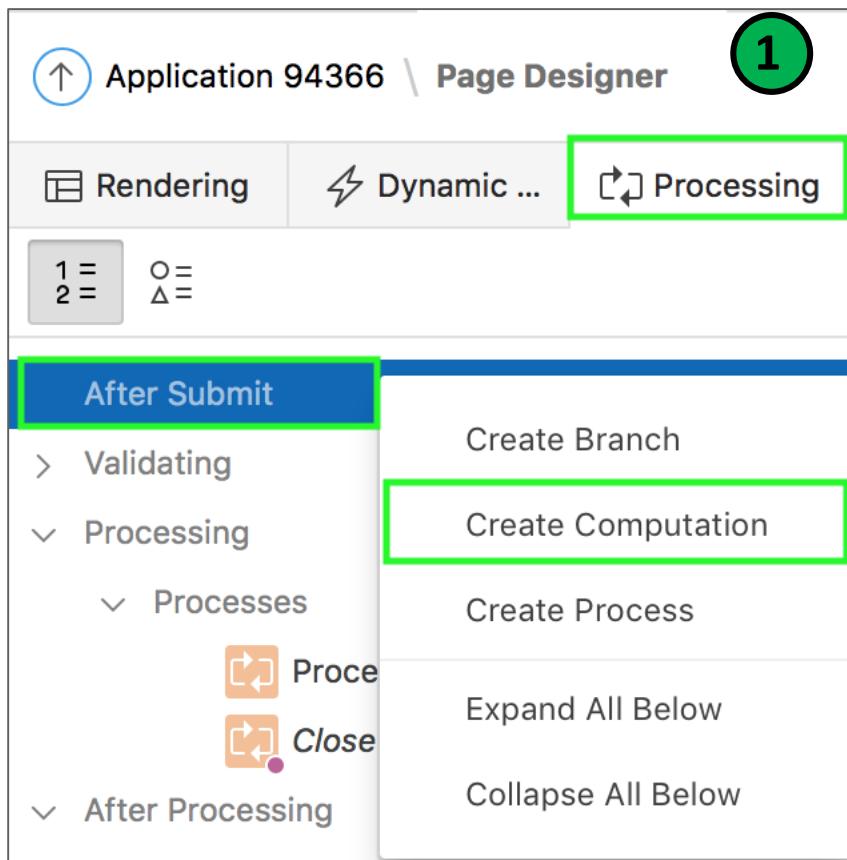
Tags

Cancel Delete Apply Changes

3

Customer Name	Address	City
Dulles, John	45020 Aviation Drive	STERLING
Hartsfield, William	6000 North Terminal Parkway	Atlanta
Logan, Edward	1 Harborside Drive	East Boston
OHare, Frank	10000 West OHare	Chicago
LaGuardia, Fiorello	Hangar Center, Third Floor	Flushing
Lambert, Albert	10701 Lambert International Blvd.	St. Louis
Bradley, Eugene	Schoephoester Road	Windsor Locks

Creating a Page Computation: Example



This screenshot shows the Computation configuration screen. At the top, there's a title bar with 'Computation' and a search bar labeled 'Filter'. A large green circle with the number 2 is at the top right of the title bar. Below the title bar, there are sections for Identification, Execution Options, and Computation. The Identification section contains 'Item Name' set to 'P7_CUST_CITY'. The Execution Options section contains 'Sequence' set to '10' and 'Point' set to 'After Submit'. The Computation section contains 'Type' set to 'PL/SQL Expression' with the value 'UPPER(:P7_CUST_CITY)' entered below it. Several fields have small edit icons (three dots) to their right.

What is a Page Process?

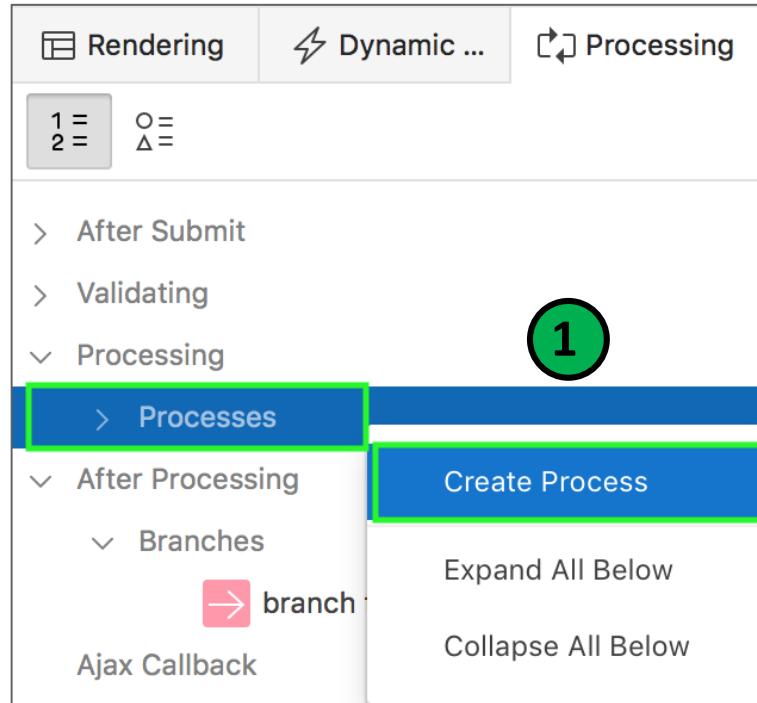
A page process is a specific event that runs when a page is loaded or submitted

- Automatic Row Processing (DML) [Legacy]
- Clear Session State
- Close Dialog**
- Form - Automatic Row Processing (DML)
- Form - Initialization
- Interactive Grid - Automatic Row Processing (DML)
- Load Uploaded Data
- Parse Uploaded Data
- PL/SQL Code
- Prepare Uploaded Data
- Reset Pagination
- Send E-Mail
- User Preferences
- Web Service
- Show Legacy/Unsupported...

Creating a Page Process: Steps

1. In either the Rendering or Processing tab, locate the node where the process will occur. Right-click and select Create Process.
2. Specify the name of the process and process type to execute
3. Under Settings, edit the appropriate attributes
4. Under Execution options, specify the sequence, and point of execution of the process
5. Specify the success and error messages
6. Click Save

Creating an On Submit Process



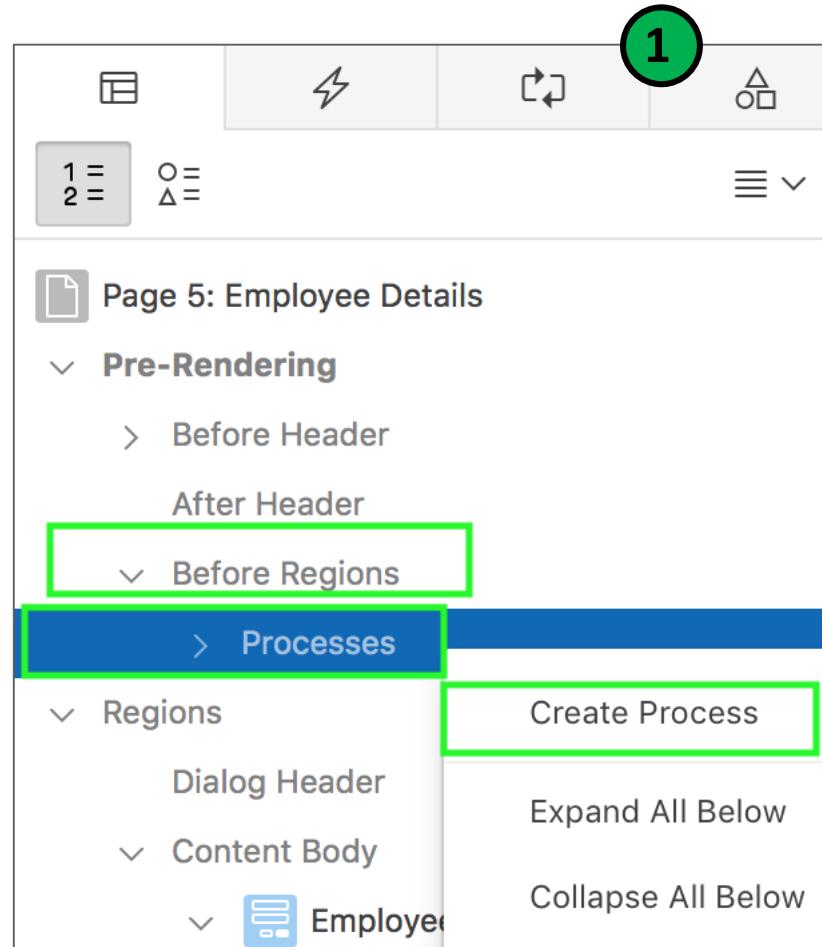
The screenshot shows the 'Create Process' configuration screen. The process name is 'Audit Information' and the type is 'PL/SQL Code'. The PL/SQL code is:

```
INSERT INTO USER_LOG(USERNAME, LAST_LOGGED_IN)  
VALUES(:APP_USER, SYSDATE);
```

The execution options are set to Sequence 10, Point Processing, and Run Process Once Per Page Visit (default).

The screenshot shows the 'Success Message' configuration screen. The success message is 'Success!'. The error message is 'Error!'. The display location is set to 'Inline in Notification'. There is also a 'Server-side Condition' section with the condition 'When Button Pressed LOGIN'.

Creating an On Load Process



The screenshot shows the 'Process' creation dialog. A green circle labeled '2' highlights the 'Display Salary' name in the 'Identification' section. The 'Type' is set to 'PL/SQL Code'. The 'Source' section shows the location as 'Local Database' and the PL/SQL code as:

```
for c1 in (
select sal from employee_details where EMPNO = :P5_EMPNO)
loop
:P5_SAL := c1.sal;
end loop;
```

The 'Execution Options' section shows a sequence of 10 and a point set to 'Before Regions'. The 'Run Process' option is set to 'Once Per Page Visit (default)'.

Understanding Validations

Employee Details

Name	John
Job	Clerk
Manager	120
Hire Date	28-Nov-18
Hire date must be in the past!	
Salary	
Commission	
Department	
Cancel	

Sample Database Application - Customer Details

First Name *	John
Street Address	45020 Aviation Drive
City	Sterling
State *	Virginia
Zip Code *	20166
Credit Limit *	1000
Phone Number	7035552143
Phone Number format not recognized {xxx-xxx-xxxx}	
Email	john.dulles@email.com
URL	http://www.johndulles.com
Tags	
Cancel	
Delete	
Apply Changes	

Sample Database Application - Order Items

Customer: Eugene Bradley Schoephoester Road Windsor Locks, CT 06096

Select Items

Product ↑=

Bag
Belt
Blouse
Business Shirt

Order Items

1 error has occurred

- At least one product must be "added to cart".

Sample Interactive Grids

Editing \ Validation

	ID	Name	Job	Manager	Hire Date	Salary	Commission	Department
<input type="checkbox"/>	7839	KING	Presid...		17-NOV-81	5000	-	ACCOUNTING
<input type="checkbox"/>	7698	BLAKE	Manager	KING	01-MAY-81	2850	-	SALES
<input type="checkbox"/>	7782	CLARK	Manager	KING	09-JUN-81	2450	-	ACCOUNTING
<input type="checkbox"/>	7566	JONES	Manager	KING	02-APR-81	2975	-	RESEARCH
<input type="checkbox"/>	7788	SCOTT	Analyst	JONES	09-DEC-82	3000	-	RESEARCH
<input type="checkbox"/>	7902	FORD	Analyst	JONES	03-DEC-81	3000	-	RESEARCH
<input type="checkbox"/>	7369	SMITH	Clerk	7902	17-DEC-80	800	-	RESEARCH
<input checked="" type="checkbox"/>	7499	ALLEN	Sales...	BLAKE	20-FEB-81	1600	9000	SALES
	7601	WARD	Sales...	BLAKE	22-FEB-81	1250	500	SALES
	7684	MARTIN	Sales...	BLAKE	28-SEP-81	1250	1400	SALES
	7844	TURNER	Sales...	BLAKE	08-SEP-81	1500	0	SALES
	7876	ADAMS	Clerk	7788	12-JAN-83	1100	-	RESEARCH

1 error has occurred

- Commission must be less than 1.5 times the Salary

Creating a Validation for a Page Item

1. View the page in page designer. Click the **Processing** tab in the left pane.
2. Right-click **Validating** and select **Create Validation**.
3. Under Validation, for Type, select the type of quality to be tested for this validation. Depending on your selection, one or more additional attributes are required to fully define this validation.
4. Specify the text for error message and select the display location.
5. Click **Save**.

Creating a SQL Validation Type - Example

Employee Details

Name
Sameer

Job
Manager

Manager

Hire Date
10-Jan-20 [Calendar icon]

Hire date must be in the past!

Salary
4000

Commission

Department

Cancel Create

! **1 error has occurred**

- Hire date must be in the past! (i)

Create a Validation to ensure Hire Date is less than the current date.

Creating a SQL Validation Type: Steps

The image shows the Oracle APEX interface for creating a validation rule. It consists of two main panels: 'Identification' and 'Execution Options'.

Identification Panel:

- Name: Hire date must be in the past

Execution Options Panel:

- Sequence: 10
- Validation Type: SQL Expression
- SQL Expression: `to_date(:HIREDATE) < SYSDATE`
- Always Execute: Yes

Associated Item: P3_HIREDATE

Left Sidebar:

- Page 3: Employee D
- Pre-Rendering
- Regions
- Content Body
- Items
 - P3_HIREDATE
 - P3_SAL
 - P3_COMM

Creating a PL/SQL Validation Type - Example

The screenshot shows a step in a 'Place Order' wizard. A green checkmark icon is at the top left, and a blue circle labeled 'Order Item' is at the top center. To the right, a yellow warning box displays an exclamation mark icon and the message: '1 error has occurred' followed by a bulleted list: 'At least one product must be "added to cart".' Below the error message is a close button ('X'). On the left, under 'Customer:', the details are listed: John Dulles, 45020 Aviation Drive, STERLING, VA 20166. The main area is titled 'Select Items' and contains a table:

Product ↑≡	Price	Add To Cart
Bag	\$125.00	<input type="text" value="0"/> <input type="button" value="▼"/>
Belt	\$30.00	<input type="text" value="0"/> <input type="button" value="▼"/>
Blouse	\$60.00	<input type="text" value="0"/> <input type="button" value="▼"/>
Business Shirt	\$50.00	<input type="text" value="0"/> <input type="button" value="▼"/>

At the bottom left are 'Back' and 'Cancel' buttons, and at the bottom right is a 'Next >' button.

Create a validation to ensure at least one product must be added to the cart to complete the order in the Place Order wizard.

Creating a PL/SQL Validation Type: Steps

The screenshot illustrates the process of creating a PL/SQL validation type in Oracle APEX. It shows two main panels: the left panel is the 'App Builder' interface for 'Page Designer', and the right panel is the 'Validation' configuration screen.

Left Panel (App Builder):

- The application ID is 94366.
- The page number is 12.
- The item 'P12_PRODUCT_ID' is selected.
- A context menu is open over the item, with the 'Create Validation' option highlighted.

Right Panel (Validation Configuration):

- Identification:** Name is set to "Product must be selected".
- Execution Options:** Sequence is set to 10.
- Validation:** Editable Region is set to "- Select -". Type is set to "PL/SQL Function Body (returning Boolean)". The function body is defined as follows:

```
for i in 1..apex_application.g_f01.count
loop
    if apex_application.g_f03(i) is not null and
apex_application.g_f03(i) != '0' then
        return true;
    end if;
end loop;

return false;
```
- Error:** Error Message is set to "At least one product must be \"added to cart\"". Display Location is set to "Inline with Field and in Notification". Associated Item is set to P12_PRODUCT_ID.
- Server-side Condition:** When Button Pressed is set to "NEXT". Type is set to "- Select -".
- Always Execute:** Set to "No".

Creating a Regular Expression Validation Type - Example

First Name *

Street Address

City

Zip Code *

Credit Limit *

Phone Number Phone Number format not recognized {xxx-xxx-xxxx}

Alternate Number

Email

URL

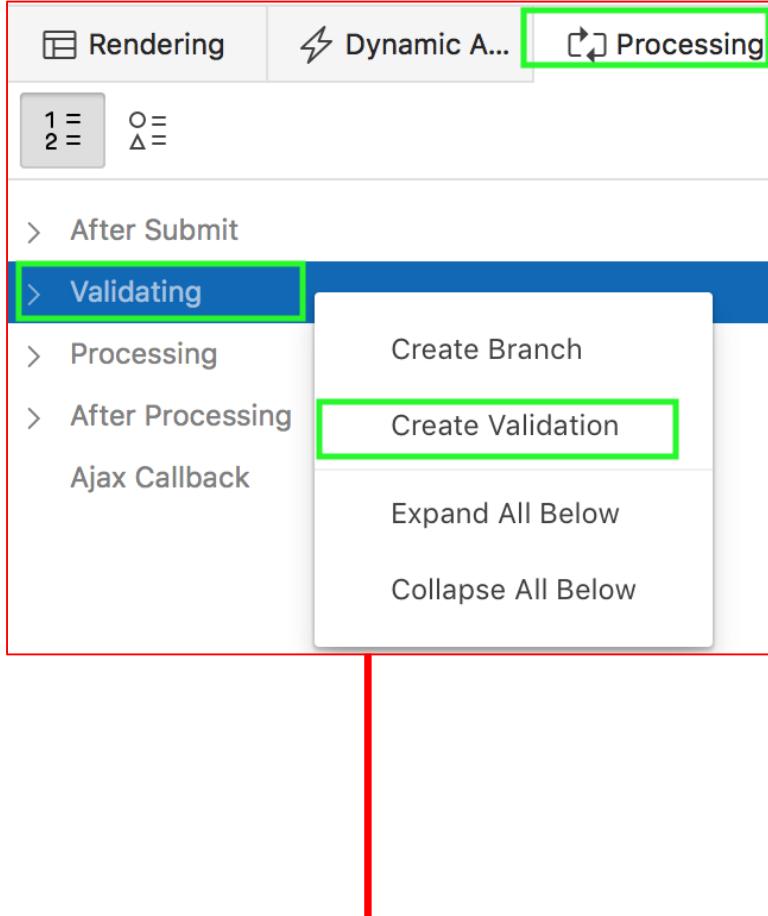
Tags

1 error has occurred

- Phone Number format not recognized {xxx-xxx-xxxx}

Create a validation to ensure that the customer's phone number entered is in the xxx-xxx-xxxx format.

Creating a Regular Expression Validation Type: Steps



The screenshot shows the 'Validation' configuration dialog. It has two tabs: 'Identification' and 'Execution Options'. In the 'Identification' tab, the 'Name' field is set to 'Phone Number Format'. In the 'Execution Options' tab, the 'Type' is set to 'Item matches Regular Expression', the 'Item' is 'P7_PHONE_NUMBER1', and the 'Regular Expression' is '^(\d{3})\d{2}\d{4}\$'. The 'Associated Item' is 'P7_PHONE_NUMBER1'. The 'Error' tab is also visible, showing the error message '#LABEL# format not recognized {xxx-xxx-xxxx}'. A red box highlights the 'Create Validation' option in the context menu and the 'Regular Expression' field in the dialog.

Creating a Row Validation in an Interactive Grid - Example

Editing \ Validation

Search: All Text Columns Go Actions ▾ Edit

	ID	Name	Job	Hire Date	Salary	Commission	Department	Tags
<input type="checkbox"/>	7698	BLAKE	Manager	01-MAY-81	4000	-	SALES	
<input type="checkbox"/>	7782	CLARK	Manager	09-JUN-81	2450	-	ACCOUNTI...	
<input type="checkbox"/>	7566	JONES	Manager	02-APR-81	2975	-	RESEARCH	
<input type="checkbox"/>	7788	SCOTT	Analyst	09-DEC-82	3000	-	RESEARCH	
<input type="checkbox"/>	7902	FORD	Analyst	03-DEC-81	3000	-	RESEARCH	
<input type="checkbox"/>	7369	SMITH	Clerk	17-DEC-80	800	-	RESEARCH	
<input checked="" type="checkbox"/>	7499	ALLEN	Salesman	20-FEB-81	1600	6000	SALES	
<input type="checkbox"/>	7521	WARD	Salesman	22-FEB-81	1250	500	SALES	
				28-SEP-81	1250	1400	SALES	
<input type="checkbox"/>	7844	TURNER	Salesman	08-SEP-81	1500	0	SALES	
<input type="checkbox"/>	7876	ADAMS	Clerk	12-JAN-83	1100	-	RESEARCH	
<input type="checkbox"/>	7900	JAMES	Clerk	03-DEC-81	950	-	SALES	
<input type="checkbox"/>	7934	MILLER	Clerk	23-JAN-82	1300	-	ACCOUNTI...	

Commission must be less than 1.5 times the Salary

1 rows selected 1 - 14

1 error has occurred

- Commission must be less than 1.5 times the Salary

Create a row validation in an editable interactive grid region. This validation ensures commission is less than 1.5 times the salary.

Creating a Row Validation in an Interactive Grid: Steps

The image displays two side-by-side configuration panels for creating a row validation rule in an Oracle APEX application.

Left Panel (Validation Rule Configuration):

- Identification:** Name: Comm limit
- Execution Options:** Sequence: 30
- Validation:**
 - Editable Region: Validation
 - Type: SQL Expression
 - SQL Expression:
:COMM is null or to_number(:COMM) < 1.5 * to_number(:SAL)
- Always Execute:** Yes

Right Panel (Validation Rule Configuration):

- Error:** Error Message: Commission must be less than 1.5 times the Salary
- Display Location:** Inline with Field and in Notification
- Associated Column:** - Select -
- Server-side Condition:**
 - When Button Pressed: - Select -
 - Execution Scope: For Created and Modified Rows
 - Type: - Select -

Understanding Branching

Page 1

	Name ↑=	Job
	ADAMS	CLERK
	ALLEN	SALESMAN
	BLAKE	MANAGER
	CLARK	MANAGER
	FORD	ANALYST
	JAMES	CLERK
	JONES	MANAGER
	KING	PRESIDENT

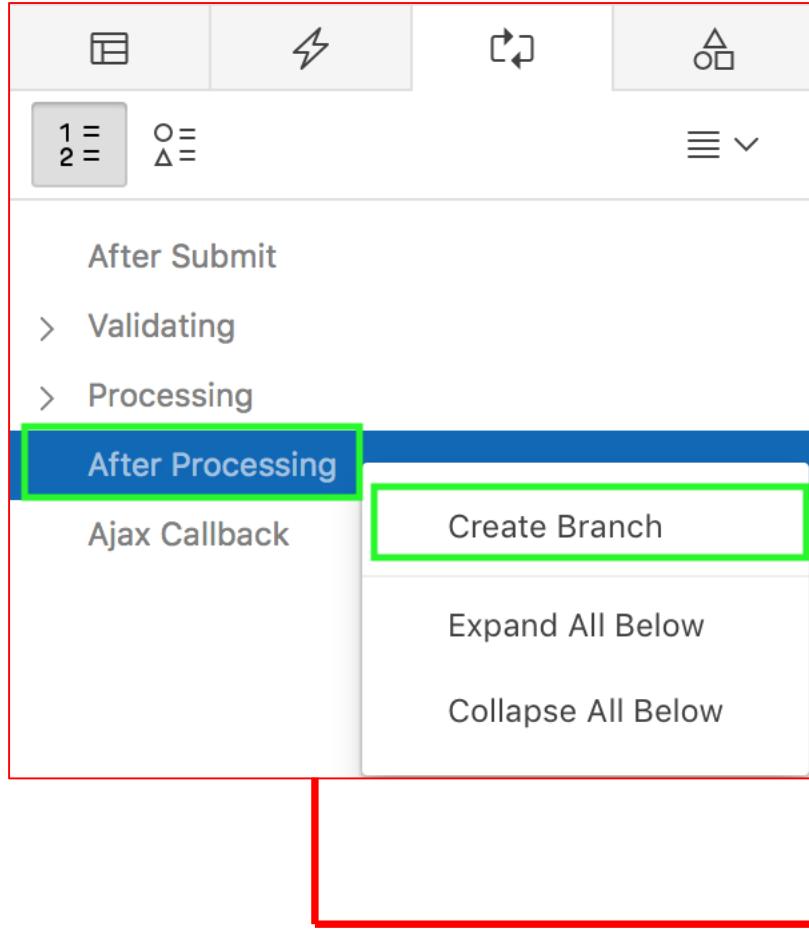
Page 2

Employee Details

Name ADAMS	Manager 7788
Job CLERK	Hire Date 12-JAN-83
Salary 1100	Commission
Department 20	Apply Changes

Page 2 is submitted and branched to Page 1

Creating a Branch



Specify branch point

The screenshot shows the 'Branch' configuration page. The 'Point' dropdown is set to 'After Processing'. A red arrow points from the 'Create Branch' menu item in the left screenshot to this dropdown. A red box highlights the 'After Processing' option in the dropdown menu.

Specify branch type

The screenshot shows the 'Branch' configuration page. The 'Type' dropdown is set to 'Page or URL (Redirect)'. A red arrow points from the 'Create Branch' menu item in the left screenshot to this dropdown. A red box highlights the 'Page or URL (Redirect)' option in the dropdown menu.

Branch configuration details:

- Name: Go to Page 3
- Sequence: 10
- Point: After Processing
- Type: Page or URL (Redirect)
- Target: Page 3

Server-side Condition:

- When Button Pressed: - Select -
- Type: - Select -

Summary

In this lesson, you learned how to create :

- Computations
- Processes
- Validations
- Branches





Hands-on Lab



Oracle **APEX**



Unit 11: Implementing Navigation in your Application

Lesson Objectives

After completing this lesson, you should be able to:

- Describe the use of shared components in an application
- Create, edit, and use the following shared components that aid navigation in an application:
 - ✓ Lists
 - ✓ Navigation Menus
 - ✓ Breadcrumbs
 - ✓ Navigation bar entries

Understanding Shared Components

The screenshot illustrates the Oracle Application Designer interface, specifically focusing on the Page Designer and Shared Components.

Page Designer: The top section shows the Page Designer toolbar with various icons for navigation, search, and save operations. A red box highlights the "Save" button, which is also highlighted with a green border. Below the toolbar is a header bar with application details and a "Page" tab. The main workspace is labeled "Application 10937 - Employee Details". It contains four main components: "Run Application", "Supporting Objects", "Shared Components" (which is highlighted with a green box), and "Utilities".

Shared Components: The bottom section displays the "Shared Components" page. This page is organized into several categories:

- Application Logic:** Application Definition Attributes, Application Items, Application Processes, Application Computations, Application Settings, Build Options.
- Security:** Security Attributes, Authentication Schemes, Authorization Schemes, Application Access Control, Session State Protection, Web Credentials.
- Other Components:** List of Values, Plug-ins, Component Settings, Shortcuts, Email Templates.
- Navigation:** Lists, Navigation Menu, Breadcrumbs, Navigation Bar List.
- User Interface:** User Interface Attributes, Themes, Templates.
- Files:** Static Application Files, Static Workspace Files.
- Data Sources:** Data Load Definitions, REST Enabled SQL, Web Source Modules, Legacy Web Service References.
- Reports:** Report Queries, Report Layouts.
- Globalization:** Globalization Attributes, Text Messages, Translate Application.

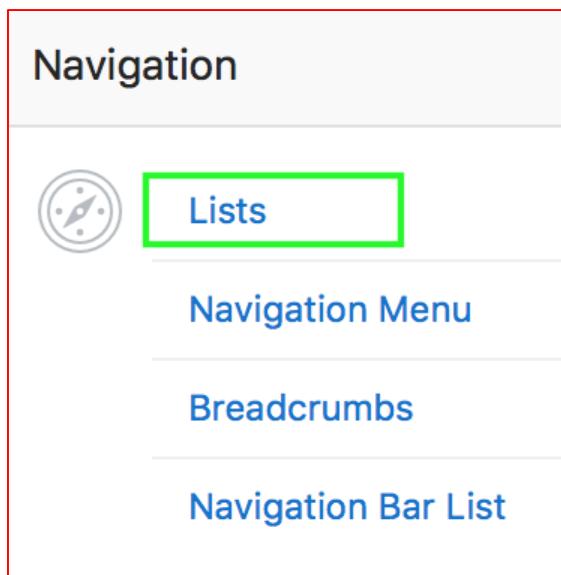
Understanding Navigational Shared Components

Navigation

-  [Lists](#)
- [Navigation Menu](#)
- [Breadcrumbs](#)
- [Navigation Bar List](#)

What are Lists?

- A list is a template driven, shared collection of links
- Oracle APEX supports two types of lists: Static, and Dynamic



The screenshot shows the Oracle APEX "Lists" page for Application 94366. The page has a header with tabs: Lists, List Details, Unused, Conditional Entries, Utilization, and History. Below the header is a search bar and a toolbar with "Reset", "Copy", and "Create >". The main area is a table listing various navigation items:

Name	Type	Entries	References	Entries Updated	List Updated	Navigation Bar	Navigation Menu
Activity Reports	Static	5	1	-	-	No	No
Administration	Static	3	1	-	-	No	No
Application Navigation	Static	15	0	-	-	No	Yes
Data Load Wizard Progress - Load Customers	Static	4	4	-	-	No	No
Feedback	Static	2	1	-	-	No	No
Footer Controls	Static	2	0	-	-	No	No
Header Quick Nav	Static	4	1	-	-	Yes	No
Mobile Navigation Menu	Static	0	0	-	-	No	No
Navigation Menu	Static	0	0	-	-	No	No
Order Wizard	Static	4	4	-	-	No	No
Report Tabs	Static	8	1	-	-	No	No
Reports	Static	3	0	-	-	No	No

Creating a Static List: Steps

1

Navigation Menu

- Desktop Navigation Menu

Navigation Bar

- Desktop Navigation Bar

Lists of Values

> **Lists**

Create...

Authorize

Build Opt

Expand All Below

Data Loc

Collapse All Below

Web Ser

2

Create List

Source

A List is a static or dynamic definition used to define a page item, such as progress bars, a navigation menu list.

Create List: **From Scratch**

As a Copy of an Existing List

3

Create / Edit List

Name and Type

You control the appearance of a list through list templates. You add a list to a page by creating a list region. Deleting a list will cause referencing regions to be removed.

* Name: **Reports**

Type: **Static**

Dynamic

Build Option: - No Build Option -

Cancel

Next >

Creating a Static List: Steps

4

Create List

Query or Static Values

List Entry Label	Target Page ID or custom URL
1 Projects Report	2
2 Project Details	3
3 Project Analysis	5
4	

< Cancel

5

Create List

Confirm

Create List Regions? Create list region on current page

* Region Position Page Template Region Position 1

Region Template No Template

* List Template: Links List

List Entry Label	Target Page ID or custom URL
1 Projects Report	2
2 Project Details	3
3 Project Analysis	5

< Cancel **Create List**

Adding Entries to a List

The image consists of three screenshots illustrating the process of adding entries to a list in Oracle APEX:

- Screenshot 1:** Shows the "Lists" page. A green circle labeled "1" highlights the "Reports" row in the grid, which is selected.
- Screenshot 2:** Shows the "List Details" tab for the "Reports" list. A green circle labeled "2" highlights the "Create Entry >" button.
- Screenshot 3:** Shows the "List Entry" creation dialog. A green circle labeled "3" highlights the "List" dropdown set to "Reports". Other fields include "Parent List Entry" (set to "- No Parent List Item -"), "Sequence" (set to 40), and "List Entry Label" (set to "New Projects").

Screenshot 1: Lists Page

The "Lists" page displays a grid of lists. The "Reports" row is highlighted with a green border. The columns are: Name, Type, Entries, References, Entries Updated, and List Updated. The "Reports" row has the following values: Desktop Navigation Bar, Static, 3, 0, 28 hours ago, 28 hours ago; Desktop Navigation Menu, Static, 3, 0, 28 hours ago, 28 hours ago; Page Navigation, Static, 2, 1, 28 hours ago, 28 hours ago; Reports, Static, 3, 1, 45 seconds ago, 45 seconds ago.

Screenshot 2: List Details Tab

The "List Details" tab is active for the "Reports" list. The "List" dropdown is set to "Reports". Other tabs include Lists, Unused, Conditional Entries, Utilization, and History. Buttons at the bottom include Grid Edit, Edit List, and Create Entry >.

Screenshot 3: List Entry Dialog

The "List Entry" dialog is open. The "List" dropdown is set to "Reports". Other fields include:

- Parent List Entry: - No Parent List Item -
- Sequence: 40
- Image/Class: (empty)
- Attributes: (empty)
- Alt Attribute: (empty)
- List Entry Label: * New Projects

The "Target" section shows:

- Target type: Page in this Application
- * Page: 6

Creating a Dynamic List

Steps:

1. Create the list by running the Create List Wizard.
2. Specify whether to create the list from scratch or by copying an existing list.
3. If creating a list from scratch, you are prompted to select a list type. Select Dynamic.
4. Enter a SQL query or a PL/SQL function returning a SQL query.
5. Add the list to a page by creating a List region.

The image shows two overlapping windows from the Oracle Database Create / Edit List wizard.

Create / Edit List (Left Window):

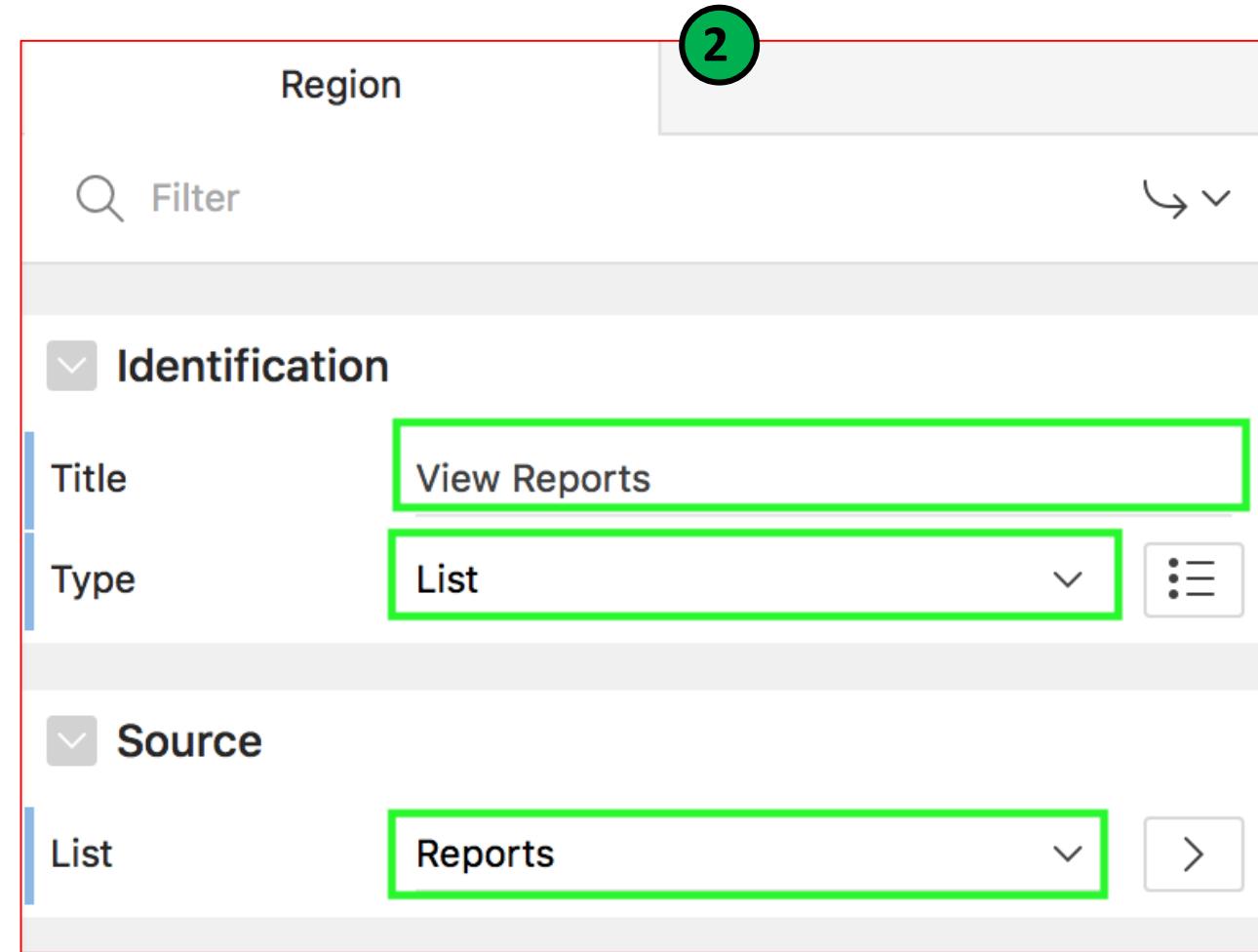
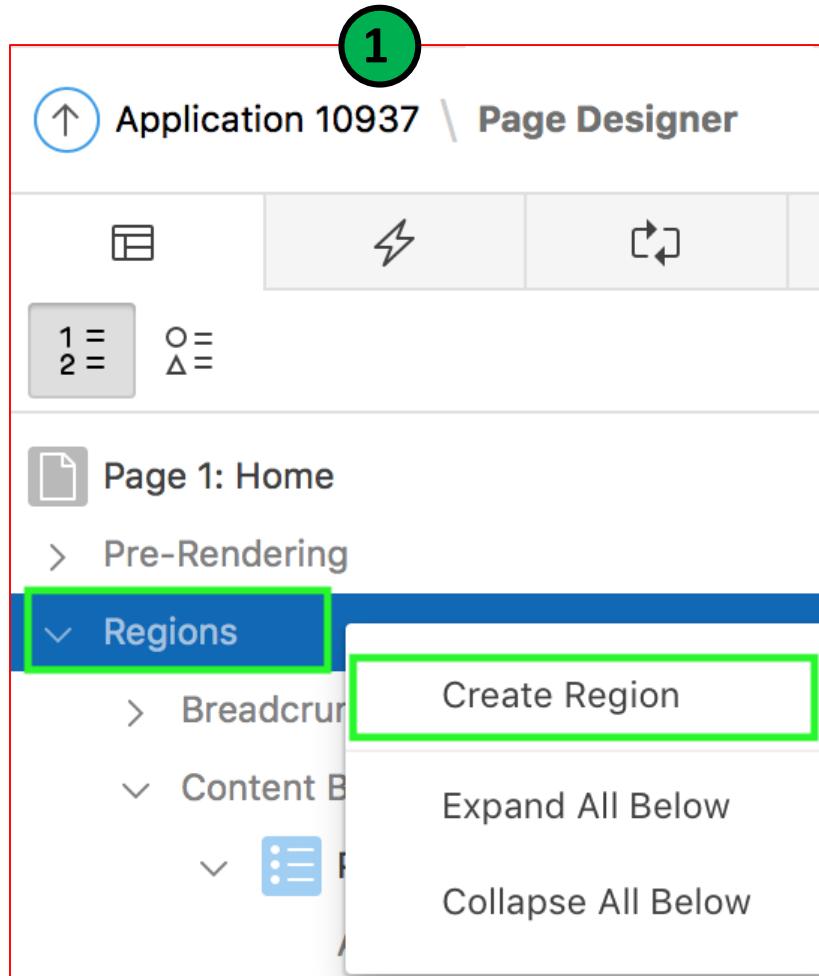
- Name and Type:** Step 1 of 5.
- A descriptive text area: "A list is a shared collection of links, each link is called a list entry. You control the appearance of the links using templates. You add a list to a page by creating a list region. Deleting a list will cause references to it to be removed."
- Name:** Reports
- Type:** Dynamic (highlighted with a green border)
- Build Option:** - No Build Option -

Create List (Right Window):

- Query Source Type:** SQL Query (highlighted with a green border)
- SQL Query:** (Text area containing the following SQL code)

```
SELECT null,
       ENAME label,
       null target,
       'YES' is_current,
       '#APP_IMAGES#del.gif' image,
       'width="20" height="20"' image_attrib,
       ENAME image_alt
  FROM emp
 ORDER BY ename|
```
- Buttons:** Build Query, Examples, < Cancel, Next >

Adding a List to a Page



What is a Navigation Menu?

The screenshot shows the 'Sample Database Application' dashboard. On the left, there is a vertical navigation menu with the following items:

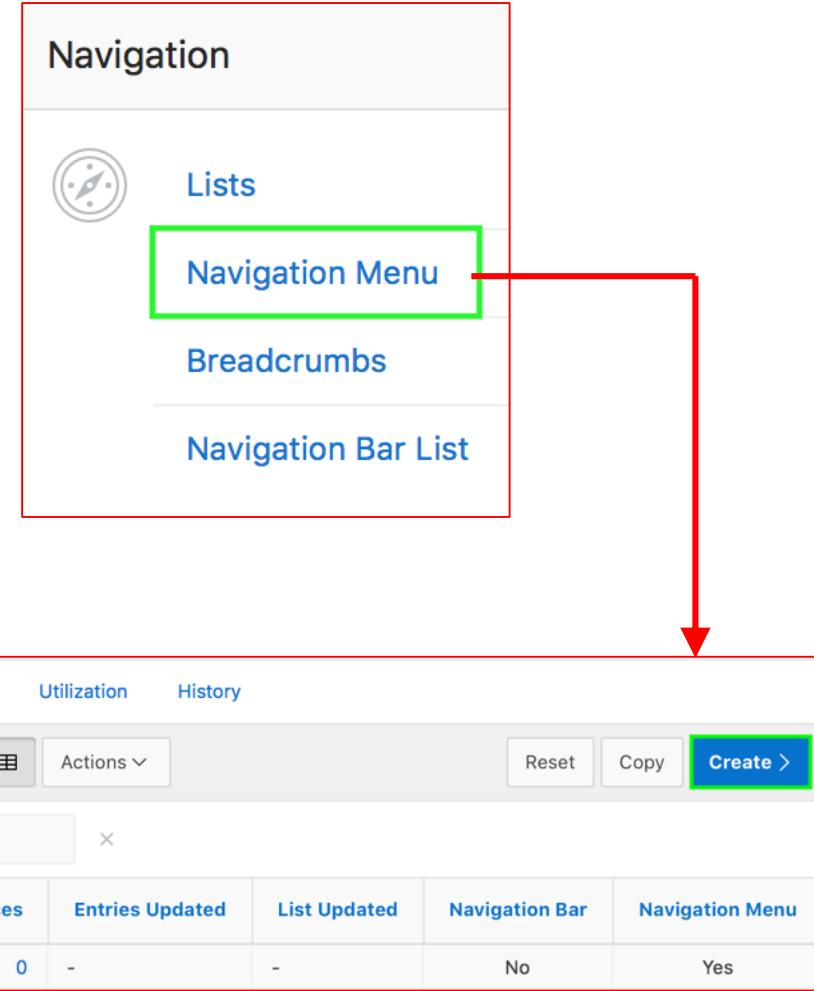
- Home
- Customers (7)
- Products (10)
- Orders (10)
- Reports >
- Administration

A red arrow points from the text 'Navigation Menu' to the 'Administration' item in the sidebar.

The main dashboard area displays the following statistics:

- Monthly Sales: \$0
- Monthly Orders: 0
- Total Products: 10
- Total Customers: 7

Below these are sections for 'Top Customers' and 'Top Products'.



Creating a Navigation Menu Entry

The screenshot shows the Oracle APEX 'List Details' page for the 'Application Navigation' list. The 'List Details' tab is selected. A red box highlights the 'List' dropdown and the 'Create Entry >' button. A green circle labeled '1' highlights the 'Application Navigation' row in the list table. A red arrow points from the 'Create Entry >' button to the 'Create / Edit' dialog window on the right. A green circle labeled '2' is on the 'Create / Edit' dialog window. A red box highlights the 'List' field in the 'Create / Edit' dialog. A green circle labeled '3' is on the 'List' field in the 'Create / Edit' dialog.

Application 94366 \ Shared Components

Lists List Details Unused Conditional Entries Utilization History

Lists List Details Unused Cond List Application Navigation Grid Edit Edit List Create Entry >

Navigation Menu

Name	Type	Entries	References	Entries Updated	List Updated	Navigation
Application Navigation	Static	15	0	-	-	-

Application 94366 \ Shared Components \ List Details \ Create / Edit

List Entry

Entry Target Current List... Conditions Authorization Configuration Click Count... User Define... Developer ...

Entry

List: Application Navigation

Parent List Entry: Reports

Sequence: 160

Image/Class:

Attributes:

Alt Attribute:

* List Entry Label: Sales by Product

Target

Target type: Page in this Application

* Page: 27

What is a Breadcrumb?

A breadcrumb is a hierarchical list of links that display using templates

Breadcrumb

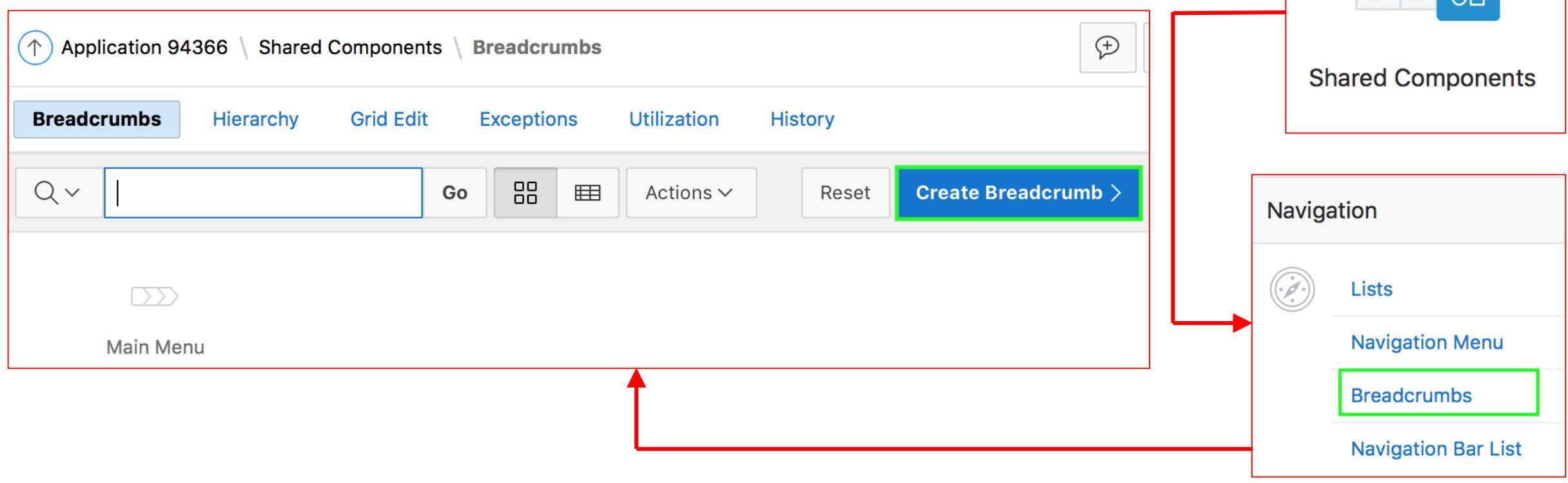
The screenshot shows a web application interface titled "Sample Database Application". On the left is a dark sidebar with navigation links: Home, Customers (7), Products (10), Orders (10), Reports >, and Administration. The "Administration" link is highlighted with a green box and has a black arrow pointing from the word "Breadcrumb" above it. The main content area has a blue header "Manage States". Below the header is a search bar with "Search: All Text Columns" and a "Go" button. To the right are buttons for "Actions", "Edit", "Save" (which is blue), "Add Row", and "Reset". A table below lists state abbreviations and names:

	Abbreviation	State Name
<input checked="" type="checkbox"/>	AK	ALASKA
<input type="checkbox"/>	AL	ALABAMA
<input type="checkbox"/>	AR	ARKANSAS
<input type="checkbox"/>	AZ	ARIZONA

Creating a Breadcrumb

Steps:

1. Create the breadcrumb by running the Create/Edit Breadcrumb Wizard.
2. Add entries to the breadcrumb.
3. Add the breadcrumb to a page by creating a region.



Adding Breadcrumb Entries

The screenshot illustrates the process of adding breadcrumb entries in an Oracle ADF application. The interface is divided into two main sections: the top navigation bar and the central configuration area.

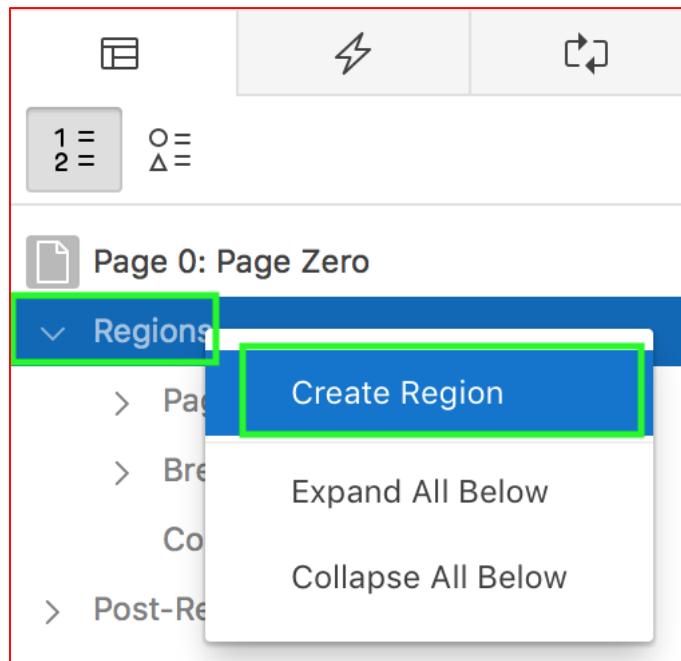
Top Navigation Bar:

- Application 94366 \ Shared Content
- Breadcrumb dropdown set to Main Menu.
- Name or Target input field.
- Page selection dropdown.
- Edit Breadcrumb Name button.
- Create Breadcrumb Entry > button (highlighted with a green box and circled with a green number 2).

Central Configuration Area:

- Main Menu:** A button labeled "Main Menu" is highlighted with a green box and circled with a green number 1.
- Create Breadcrumb >:** A button labeled "Create Breadcrumb >" is located in the top right of the main toolbar.
- Breadcrumb Entry Dialog:** An open dialog titled "Breadcrumb Entry:" (circled with a green number 3).
 - Breadcrumb:** Set to Main Menu.
 - Page:** Value 1 is highlighted with a green box.
 - Entry:** Sequence is 10, Parent Entry is "- Select Parent -". Short Name is "Project Reports" (highlighted with a green box).
 - Target:** Target is a "Page in this Application". Page value is 7.

Adding a Breadcrumb Region in a Page

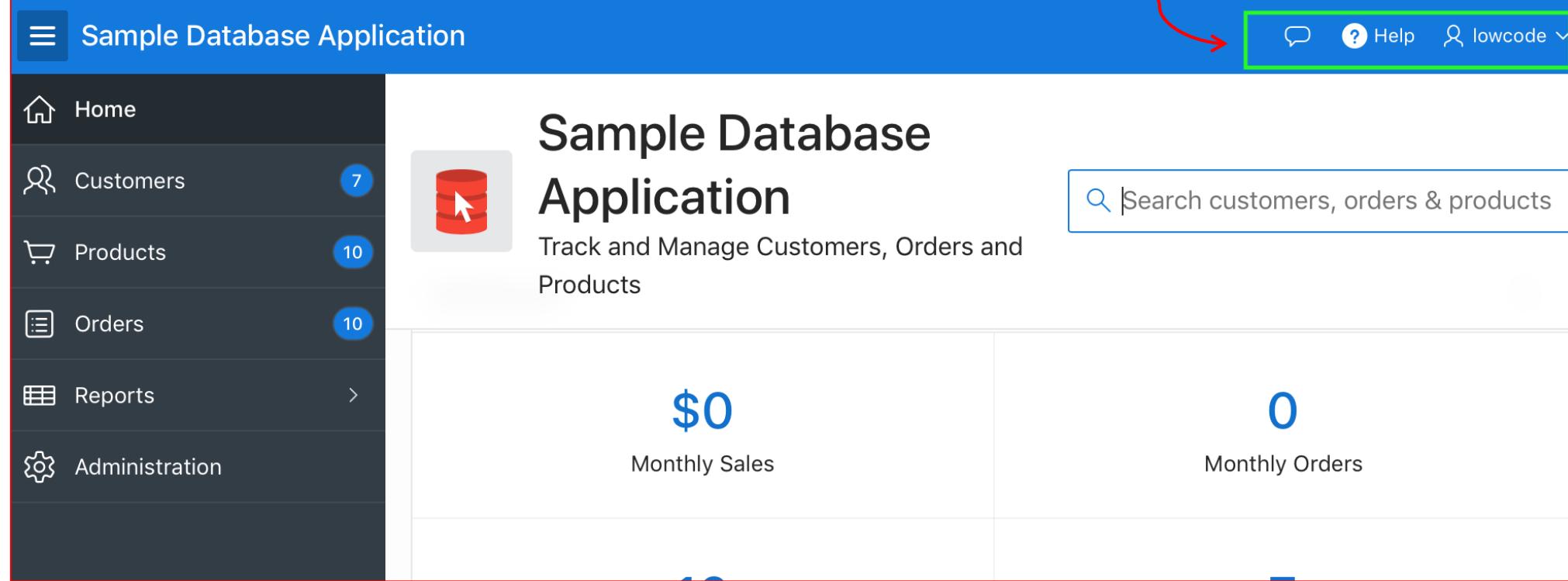


该部分展示了“Region”配置对话框。对话框中包含以下设置：

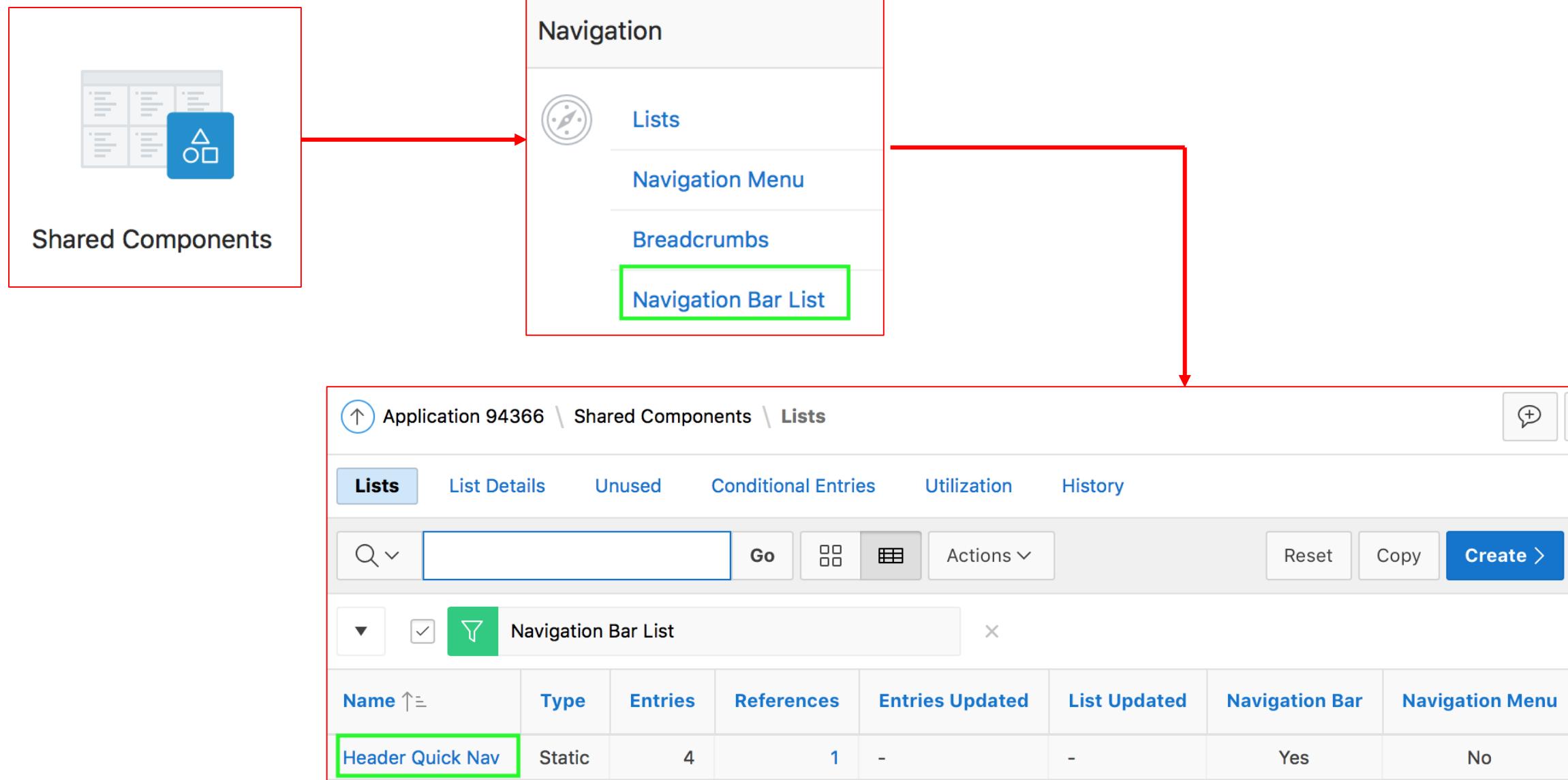
- Title:** Breadcrumb
- Type:** Breadcrumb
- Source:** Breadcrumb (下拉菜单显示为 Main Menu)
- Server-side Condition:**
 - Type:** Current Page is in comma delimited list
 - Pages:** 2,3

What is a Navigation Bar Entry?

- Navigation bar entries offer an easy way to move users between pages in an application.
- A navigation bar entry can be an image, text, or an image with text beneath it.



Accessing the Navigation Bar List



Creating a Navigation Bar Entry: Example

The screenshot shows the 'Employee Details' application. The navigation bar includes a 'Help' button highlighted with a red box. The main content area is titled 'Employees Report' and contains a table with the following data:

	Name ↑↓	Job	Hire Date	Salary	Commission	Department
Edit	ADAMS	CLERK	12-JAN-83	1,100	-	RESEARCH
Edit	ALLEN	SALESMAN	20-FEB-81	1,600	300	SALES
Edit	BLAKE	MANAGER	01-MAY-81	2,850	-	SALES
Edit	CLARK	MANAGER	09-JUN-81	2,450	-	ACCOUNTING
Edit	FORD	ANALYST	03-DEC-81	3,000	-	RESEARCH

1. Create a blank page and add a Help Text region type
2. Create a navigation bar entry for Help

The screenshot shows the Oracle APEX builder's regions palette. A red dashed arrow points from the 'CONTENT BODY' label to the 'Help Text' icon, which is highlighted with a green box. The text 'Drag and drop' is overlaid on the arrow. The palette lists various regions: PAGE HEADER, PAGE NAVIGATION, BREADCRUMB BAR, BEFORE CONTENT BODY, CONTENT BODY, FOOTER, and INLINE DIALOGS.

Regions Items Buttons

Column Toggle Report Form Help Text Interactive Grid

Adding a Help Text region type

Creating a Navigation Bar Entry: Example

The screenshot shows the Oracle ADF Faces application interface for managing navigation bar entries. The top navigation bar includes tabs for Lists, List Details (selected), Unused, Conditional Entries, Utilization, and History. Below the tabs, a list of navigation bars is displayed, with "Desktop Navigation Bar" selected. On the right, there are buttons for Grid Edit, Edit List, and Create Entry > (highlighted with a green box). A red arrow points from the "List Details" tab area to the "Create Entry >" button.

List Entry

Entry Target Current Lis... Conditions Authorizati... Configurat... Click Coun... User Defin... Developer ...

List: Desktop Navigation Bar

Parent List Entry: - No Parent List Item -

Sequence: 10

Image/Class:

Attributes:

Alt Attribute:

* List Entry Label: Help

Target

Target type: Page in this Application

* Page: 8

reset pagination for this page

Printer Friendly

Request: &APP_PAGE_ID|

A red box highlights the "Create List Entry" button in the top right corner of the modal window.

Summary

In this lesson, you learned how to create, edit and use the following shared components that aid navigation in an application:

- Lists
- Navigation Menus
- Breadcrumbs
- Navigation bar entries





Hands-on Lab



Oracle **APEX**



Unit 12: Using Themes and Theme Styles

Lesson Objectives

After completing this lesson, you should be able to:

- Describe themes and templates
- Explain the key features of Universal Theme
- Create and use a new theme
- Describe Universal Theme, Theme Styles, Theme Roller, and Template Options
- Use Theme Styles
- Use Theme Roller, and Live Template Options
- Upload and use a CSS

Understanding Themes

A theme identifies a collection of templates which define the look and feel of the application.

The screenshot shows a web application interface titled "Sample Database Application". The top navigation bar includes a "Help" link and a "lowcode" dropdown. The main header features a database icon and the text "Sample Database Application" followed by "Track and Manage Customers, Orders and Products". A search bar is located in the top right. The left sidebar, labeled "Universal Theme" with a red arrow pointing to it, contains links for "Home", "Customers" (with a badge of 7), "Products" (with a badge of 10), "Orders" (with a badge of 10), "Reports", and "Administration". The main content area displays a dashboard with four summary cards: "Monthly Sales" (\$0), "Monthly Orders" (0), "Total Products" (10), and "Total Customers" (7). Below the dashboard are two tables: "Top Customers" and "Top Products".

Top Customers		Top Products	
Bradley, Eugene	Windsor Locks, CT · 1,000	Jacket - 18 x \$150	\$2,700
Logan, Edward	East Boston, MA · 1,000	Bag - 16 x \$125	\$2,000
		Trousers - 21 x \$80	\$1,680
		Ladies Shoes - 12 x \$120	\$1,440

What is Universal Theme?

- Default theme for Oracle APEX applications
- Uses responsive HTML5 templates
- Grid-layout for HTML forms
- Modern flat-look
- Ships with a variety of pre-built theme styles

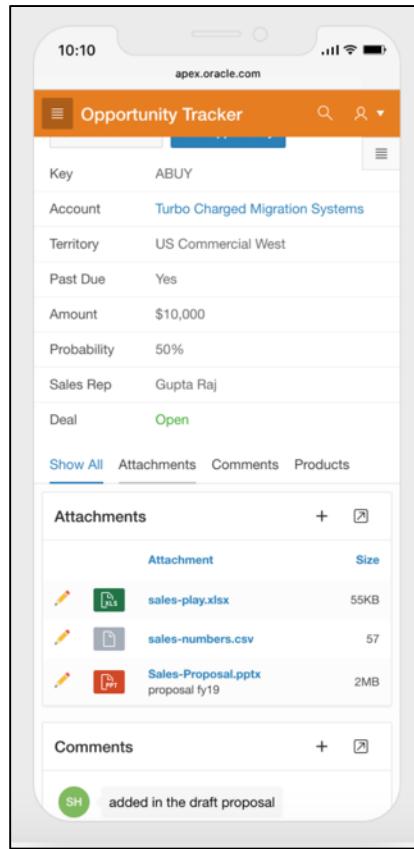


<https://apex.oracle.com/ut>

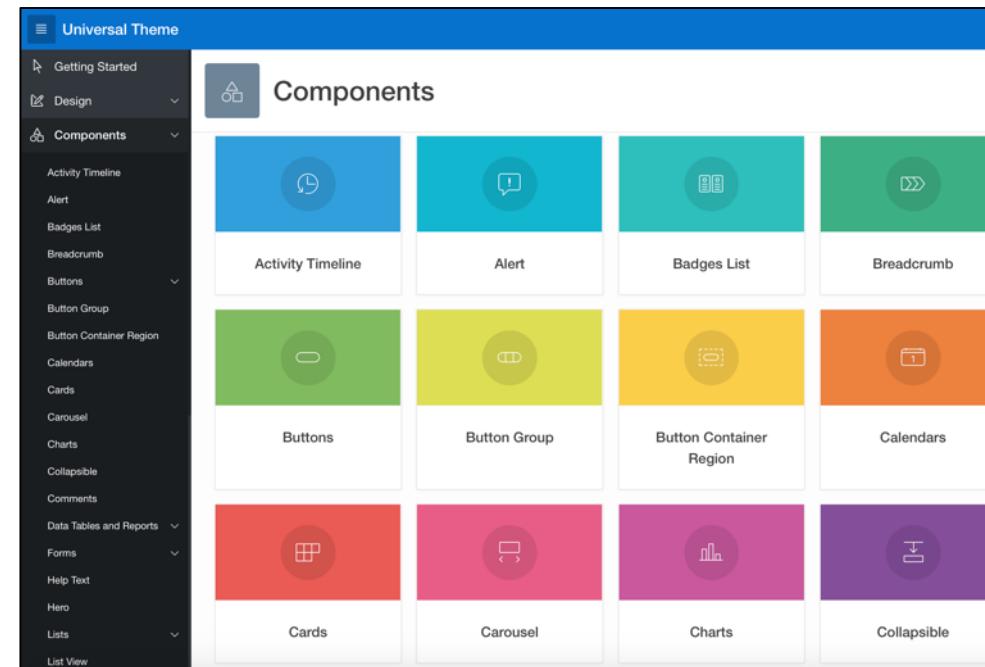
Understanding Universal Theme: Key Features

Three key aspects of Universal Theme that make it ideally suited for Oracle APEX development are:

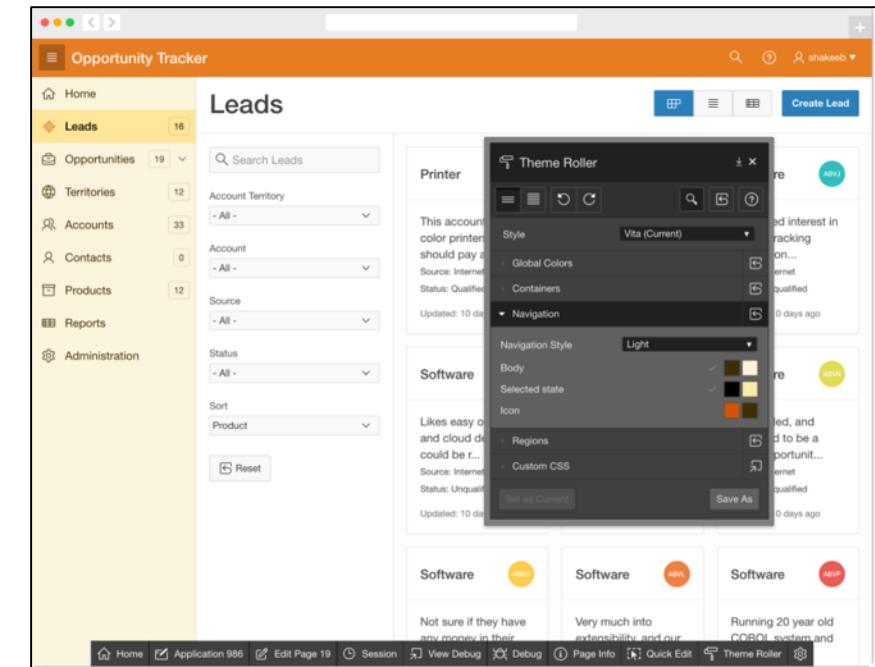
Responsive Design



Versatile UI Components



Easy Customization



Accessing the Themes Page

The diagram illustrates the navigation path to the Themes page:

- 1**: Shared Components icon (grid with triangle and square) on the Shared Components page.
- 2**: User Interface Attributes section on the User Interface page, with the **Themes** item highlighted.
- 3**: Themes page for Application 94366, showing a table of themes and a sidebar with various tasks.

Shared Components

User Interface

Themes

Themes Page (Application 94366)

Number	Name	User Interface	Is Universal Theme	Is Current	Subscribed From	Subscribers	Templates	Page Templates	Region Templates	Button Templates	List Templates
42	Universal Theme - 42 *	Desktop	✓	✓			61	9	17	3	12

Tasks

- Copy Theme
- Delete Theme
- Export Theme
- Import Theme
- Change Identification Number
- Restore Theme Subscription
- Manage Workspace Themes

Managing Themes Using Theme Subscriptions

- You can manage themes directly or through theme subscriptions
- Subscribing to a master theme enables the theme to be upgraded during future Oracle APEX releases
- You subscribe to a theme when:
 - ✓ Running the Create Application Wizard
 - ✓ Creating a new theme from the theme repository
- Once an application subscribes to a master theme:
 - ✓ You can change only the default templates
 - ✓ All theme attributes, subscribed template options, and subscribed templates are set to read-only

Creating a Theme

Application 10937 \ Shared Components \ Themes

Themes Reports History

Actions: Go, Actions, Reset, Switch Theme, Create >

Number	Name	User Interface	Is Universal Theme	Is Current	Subscribed From	Subscribers	Templates	Page Templates	Region Templates	Button Templates	List Templates
42	Universal Theme - 42 *	Desktop	✓	✓	Theme Repository	1	60	9	16	3	12

Create Theme

Method

From the Repository

- As a copy from another application
- From Scratch
- From Export

Creating a Copy of an Existing Theme

Themes **1**

A Theme is a named collection of templates used to define the user interface of an application.

Tasks

Copy Theme

Delete Theme

Export Theme >

Import Theme >

Change Identification Number >

Restore Theme Subscription >

Manage Workspace Themes >

Copy Theme

Application: 10937 Employee Details [?](#)

Copy from Theme: 42. Universal Theme [?](#)

* Copy to this Theme ID: **122** [?](#)

Subscribe Theme: **Yes** [No](#)

Copy Theme

Application: 10937 - Employee Details [?](#)

Copy Theme ID from: **42** [?](#)

Copy Theme ID to: **122** [?](#)

< Cancel **Copy Theme** **Confirm**

This diagram illustrates the three-step process of creating a copy of an existing theme. Step 1 shows the 'Copy Theme' button being selected in the sidebar. Step 2 shows the 'Copy Theme' dialog where the target theme ID '122' is specified. Step 3 shows the confirmation dialog with the copied theme ID '122'.

Editing a Theme

The screenshot illustrates the process of editing a theme in an Oracle ADF application. It consists of two main panels:

- Left Panel (List View):** Shows a table of themes. A specific row is selected, highlighted with a green border. The "Name" column for this row contains the value "Universal Theme - 42 *".
- Right Panel (Edit View):** A detailed form for editing the selected theme. The "Name" field is populated with "Universal Theme". Other fields include "Identifier" (UNIVERSAL_THEME), "Navigation Type" (List), and "Navigation Bar Implementation" (List). A note at the bottom states: "This is the 'master' copy of this theme".

A red box highlights the "Universal Theme - 42 *" entry in the list view. A red arrow points from this entry to the corresponding "Name" field in the edit view.

Left Panel (List View) Headers:

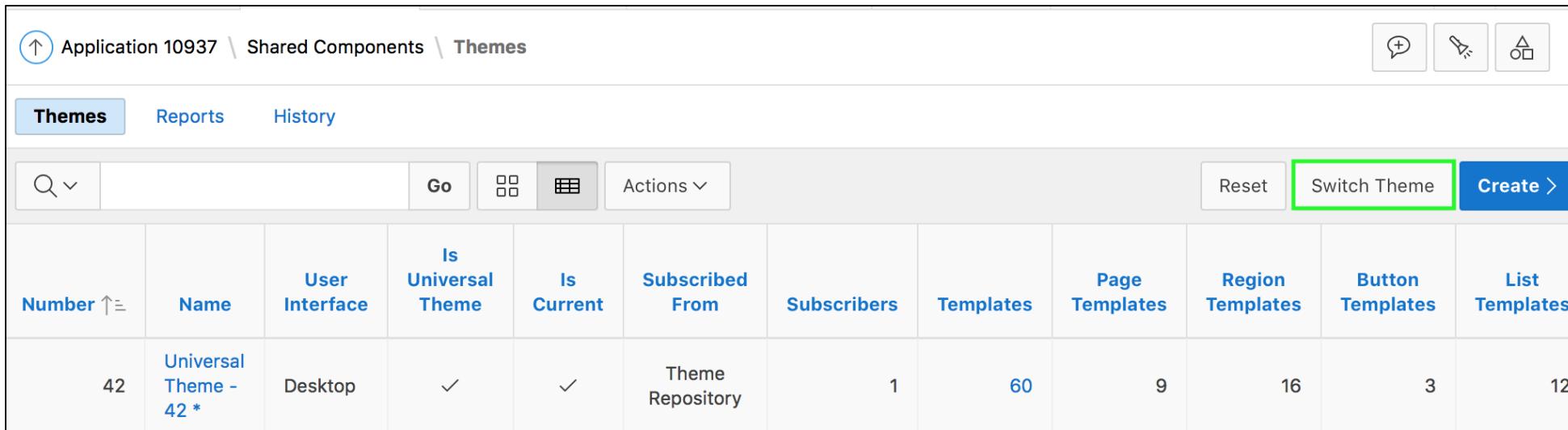
- Number
- Name
- User Interface
- Is Universal Theme
- Is Current
- Subscription Frontend

Right Panel (Edit View) Fields:

- Application: 94366
- * Theme Number: 42
- * Name: Universal Theme
- * Identifier: UNIVERSAL_THEME
- Navigation Type: List
- Navigation Bar Implementation: List
- Description:

Note: This is the "master" copy of this theme

Switching an Active Theme



The screenshot shows the 'Themes' page in Oracle APEX. The top navigation bar includes 'Application 10937', 'Shared Components', and 'Themes'. Below the navigation is a toolbar with icons for search, refresh, and other actions. The main content area has tabs for 'Themes' (selected), 'Reports', and 'History'. A search bar and a 'Go' button are followed by a grid icon and a 'Actions' dropdown. On the right, there are 'Reset', 'Switch Theme' (which is highlighted with a green border), and 'Create >' buttons. The table below lists themes, with columns for Number, Name, User Interface, Is Universal Theme, Is Current, Subscribed From, Subscribers, Templates, Page Templates, Region Templates, Button Templates, and List Templates. One row is selected, showing 'Universal Theme - 42 *' as the current theme.

Number	Name	User Interface	Is Universal Theme	Is Current	Subscribed From	Subscribers	Templates	Page Templates	Region Templates	Button Templates	List Templates
42	Universal Theme - 42 *	Desktop	✓	✓	Theme Repository	1	60	9	16	3	12

1. On the Themes page, click Switch Theme
2. Specify the current theme for the app and select the new theme to switch to
3. Review the details in the Verify Compatibility page and resolve the warnings if any
4. Click Switch to Theme

Using Theme Styles

Tasks App

Dashboard

Assigned To

Customize

Appearance

You can personalize the appearance of this application by changing the Theme Style. Please select a Theme Style from the list below and click on Apply Changes.

Use Application Default Style

Vista

Vita

Vita - Dark

Vita - Red

Vita - Slate

Release 1.0 [Customize](#)

1

2

Apply Changes

Tasks App

Dashboard

Assigned To

Release 1.0 [Customize](#)

1

2

Vita

Vita - Red

Understanding Theme Styles

- You use theme styles to customize themes, switch to a different color scheme, apply a flat look, or make a theme responsive
- A theme style defines a CSS style sheet that is added to the base CSS
- Universal Theme includes several pre-built styles
- You can easily generate additional styles by using the built-in Theme Roller utility

Creating a Theme Style

Theme

Show All Name Theme S... JavaScript... Compon... Region D... Dialog D... Global T... Icons Image Styles **Styles** Files

Styles

Add Style >

1

Settings

* Name

Is Current Yes **No** [?](#)

Is Public Yes **No** [?](#)

Accessibility Tested Yes **No** [?](#)

2

Theme Roller Attributes

Read Only Yes **No** [?](#)

Input Parameter File URLs `#THEME_IMAGES#less/theme/Vita-Green.less`

Output CSS File URL `#THEME_IMAGES#css/Vita-Green#MIN#.css?v=#APEX_VERSION#`

3

Enabling Users to Select a Theme Style

1

User Interface

User Interface Attributes

Themes

Templates

2

Application 136944

Show All User Interfaces General Properties

User Interfaces

Name	Type	Default	Auto Detect
Desktop	monitor	✓	

3

User Interface Details

Show All Identification Attributes Navigation Menu

Attributes

Auto Detect	Yes	No
Default	Yes	No

Enable End Users to choose Theme Style Yes No

4

Theme Styles

Show All Settings

Settings

* Name Vita - Green

Is Current Yes No

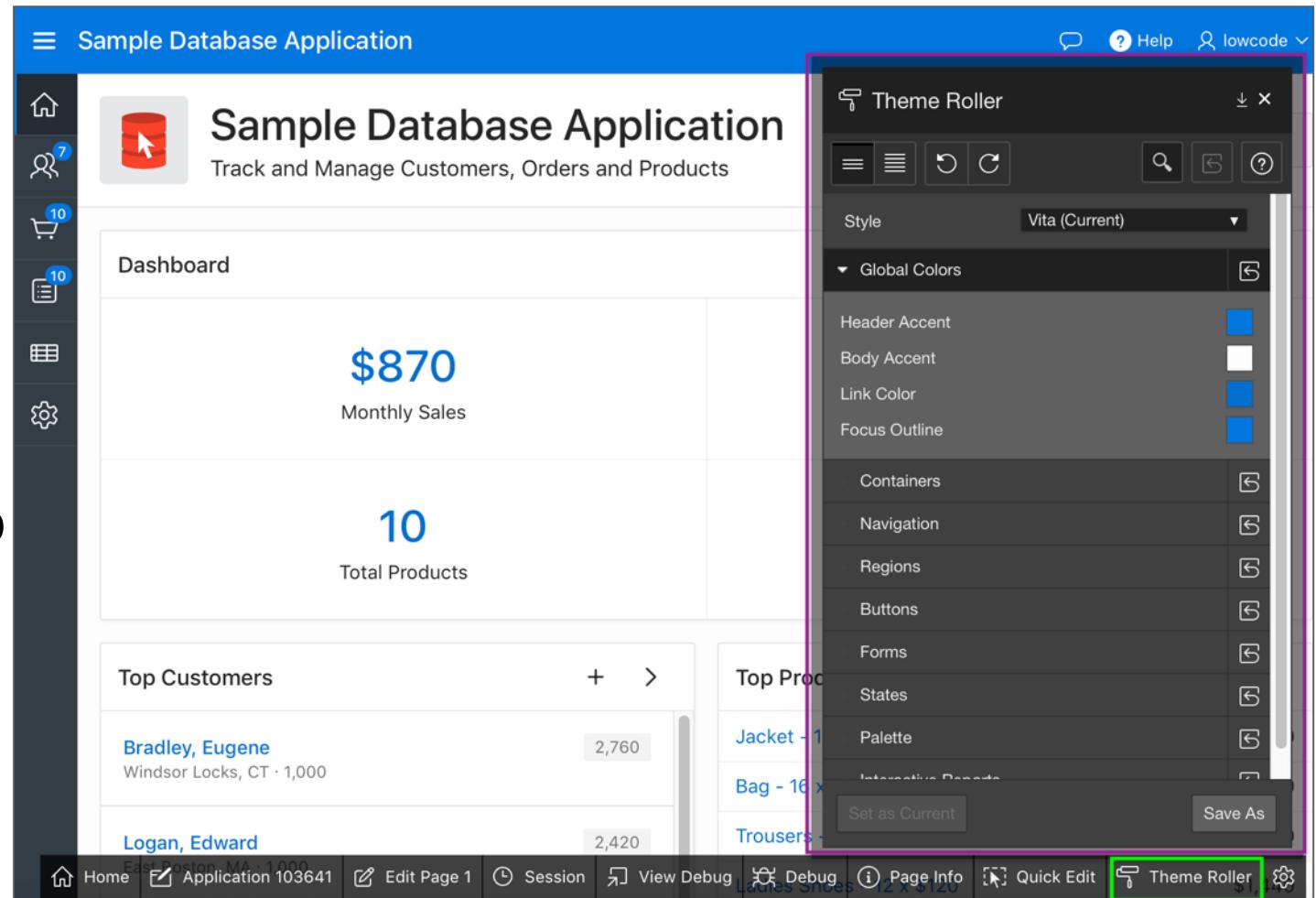
Is Public Yes No

For each theme style, set "Is Public" to "Yes"

Using Theme Roller

Theme Roller:

- Enables easy customization of the look and feel of your application
- Allows you to modify a number of style attributes and see changes applied to your application in real time
- Allows you to save your changes as a Theme Style



Understanding Templates

- The Application Express engine constructs the appearance of each page in an application using Templates
- Templates define how pages, page controls, and page components display

User Interface

User Interface Attributes

Themes

Templates



Application 103641 \ Shared Components \ Templates

Templates Subscription Publish Utilization History

Reset Create >

Type	Name	References	Updated	Updated By	Default	Theme	Copy
Breadcrumb	Breadcrumb	2			✓	42	
Button	Icon	18				42	
Button	Text	46			✓	42	
Button	Text with Icon	13				42	
Label	Hidden	6				42	
Label	Optional	45			✓	42	
Label	Optional - Above	0				42	
Label	Optional - Floating	13				42	
Label	Required	21				42	

Understanding Template Options

- Template options provide a declarative way for developers to apply different styles to components on an application page
- Examples of how developers can use template options include:
 - ✓ Applying different colors or accents
 - ✓ Applying different spacing and padding
 - ✓ Rendering buttons in different styles, with and without icons
 - ✓ Displaying form fields with different alignments
- Application components that support template options:
 - ✓ Pages ,regions, classic reports
 - ✓ Breadcrumbs, lists, items and labels, and buttons

Creating Template Options

Application 103641 \ Shared Components \ Templates \ Edit Breadcrumb Template

Breadcrumb Template: 1 of 1 [?](#)

Show All Name Subscription [Template Option...](#) Template Type Definition Link Attributes Comments Substitution St...

[Add Template Option](#)

Preset Template Options: No matching template options defined.

Default Template Options: No matching template options defined.

Using Live Template Options

The screenshot shows the Oracle APEX 'Sample Charts' application interface. A red box highlights the 'Live Template Options' dialog on the left side of the screen. This dialog has tabs for 'Region' and 'List'. Under 'Region', there is a 'Common' section with checkboxes for 'Use Template Defaults' (checked), 'Apply Theme Colors' (checked), and 'Display Subtitle' (unchecked). The 'Layout' section is expanded, showing options for 'Featured' (Stacked, 2 Columns, 3 Columns, 4 Columns, 5 Columns, Float, Span Horizontally), 'Body Text' (Color Fill), 'Icon Shape' (Circle), and 'Animation' (Color Fill). The '4 Columns' option is selected and highlighted with a green border. At the bottom of the dialog are 'Cancel' and 'Save' buttons. The main content area displays various chart types as cards, such as Box Plot, Bubble, Classification, Dashboard, Dashboard - Pie Charts, Legend Plug-in, Line, Polar, and Pyramid. A red box also highlights the 'Quick Edit' button in the bottom right corner of the application toolbar.

Sample Charts
Charting examples for APEX applications

Live Template Options

Region List

Common

General

- Use Template Defaults
- Apply Theme Colors
- Display Subtitle

Style

Icons

Layout

- Featured
 - Stacked
 - 2 Columns
 - 3 Columns
 - 4 Columns**
 - 5 Columns
 - Float
 - Span Horizontally

Body Text

Icon Shape

Animation

Cancel Save

Box Plot

Bubble

Classification

Dashboard

Dashboard - Pie Charts

Legend Plug-in

Line

Polar

Pyramid

View Debug

Debug

Page Info

Quick Edit

Theme Roller

LINE WITH AREA

Using Custom Cascading Style Sheets

The screenshot shows the Oracle Application Development Framework interface for managing static application files.

- Step 1:** The left sidebar shows navigation categories: **Static Application Files** (highlighted with a green box and circled with a red line) and **Static Workspace Files**.
- Step 2:** The main content area displays the path **Application 83106 \ Shared Components \ Static Application Files**. It includes a search bar, a toolbar with **Go**, **Actions**, **Reset**, **Delete All Files**, **Download as Zip**, and a prominent **Upload File >** button (highlighted with a green box and circled with a red line).
- Step 3:** A modal dialog titled **Upload Static Application File(s)** is open. It contains instructions: "Use this page to associate files like images, CSS or Javascript files with your application. To associate a file with your application, select the file(s), and click Upload." The dialog has fields for **Directory** (disabled), *** File(s)** (set to **Choose Files newapp.css**), *** File Character Set** (**Unicode UTF-8**), *** Unzip File** (**Yes** selected), and buttons for **Cancel**, **Upload and Upload Another**, and a highlighted **Upload** button (circled with a red line).

Referencing an Uploaded CSS in the Page Template

1. Navigate to your application home page and click Shared Components.
2. Under User Interface, select Themes. The Themes page appears.
3. Select your theme and then, on the Tasks list, click View Templates.
4. Select the name of the page template you want to edit.
5. Use the `<link>` tag within the Header section to reference the appropriate style sheet.
 - Use the substitution string `#APP_IMAGES#` to reference an uploaded file that is associated with a specific application
 - Use the substitution string `#WORKSPACE_IMAGES#` to reference an uploaded file that is associated with a specific Workspace

Summary

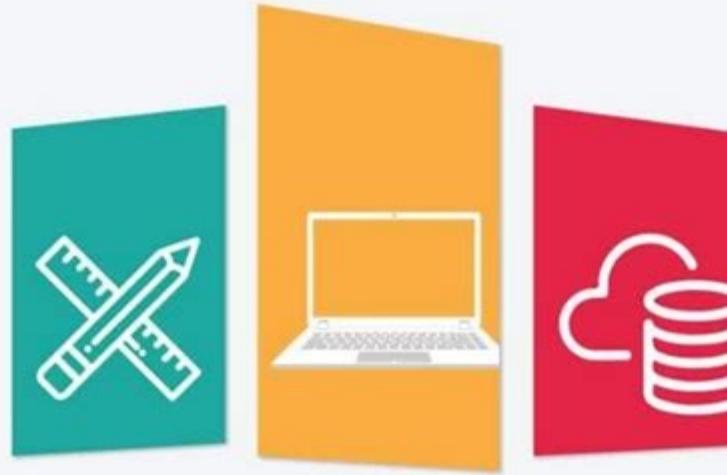
In this lesson, you learned how to:

- Create and use a new theme
- Switch a Theme
- Use Theme Styles
- Use Theme Roller, and Live Template Options
- Upload and use a CSS

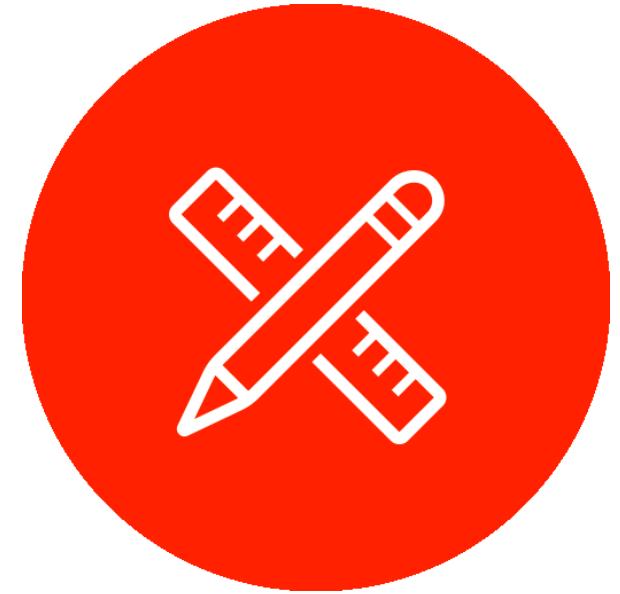




Hands-on Lab



Oracle **APEX**



Unit 13: Implementing Security in your Application

Lesson Objectives

After completing this lesson, you should be able to:

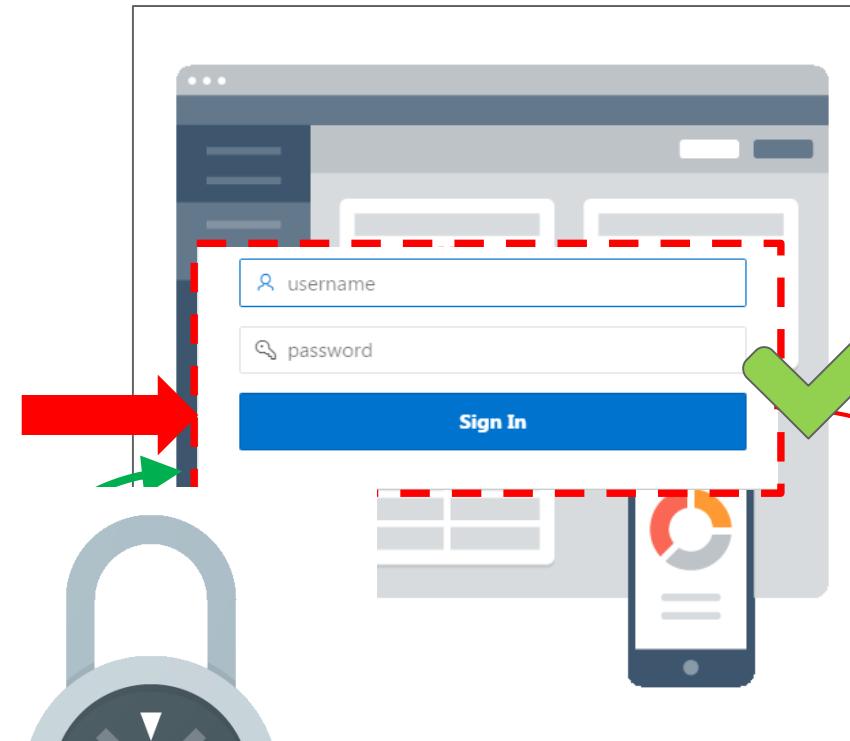
- Explain how to secure your application
- Describe authentication and authorization
- Create and use an authentication scheme for your application
- Create and attach an authorization scheme to your application, page, or components

How to Secure an APEX Application?



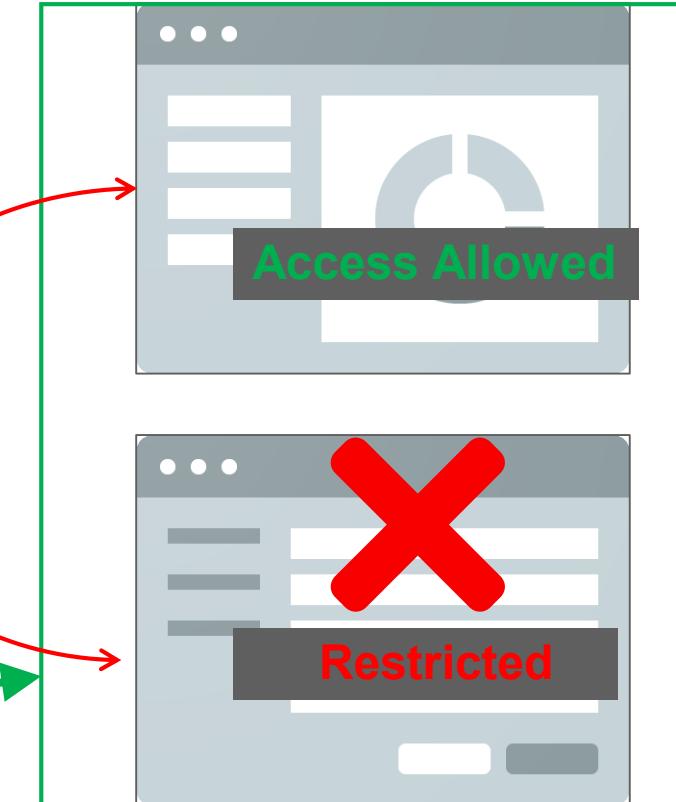
Authentication —

Who are you?



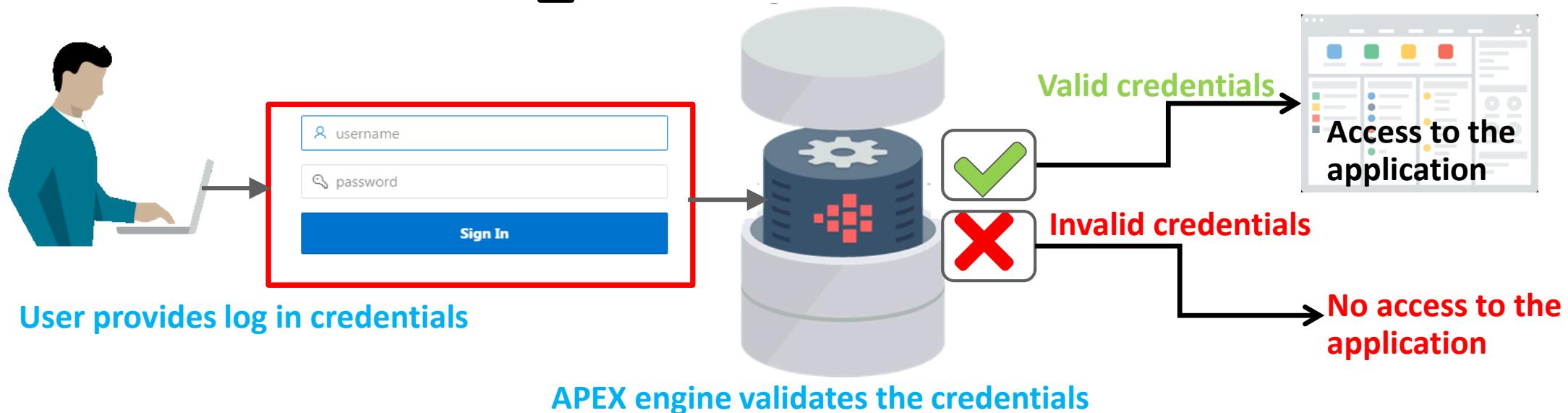
Authorization

What are you allowed to do after logging in?



Establishing User Identity through Authentication

- Authentication is the process of establishing each user's identity before they can access your application
- Many authentication processes require users to provide credentials such as username and password
- APEX engine uses **APP_USER** to track each user's session state



How to Implement Authentication?

In Application Express, you can create authentication by:

- Selecting a built-in authentication scheme
- Creating a custom authentication scheme



Understanding Preconfigured Authentication Schemes

Oracle Application Express supports the following preconfigured authentication schemes:

- Open Door Credentials
- Application Express Accounts
- Database Accounts
- LDAP Directory
- No Authentication (Using DAD)
- Oracle Application Server Single Sign-On
- Custom
- HTTP Header Variable
- Social Sign-In

Creating an Authentication Scheme Based on a Pre-configured Scheme

The screenshot shows the Oracle Database Control interface with the path: Application 103641 \ Shared Components \ Authentication Schemes. The 'Authentication Schemes' tab is selected. A red box highlights the table below. The table has columns: Name, Scheme Type, Subscribed From, and Subscribers. One row is selected, showing 'Application Express Accounts - Current' in the Name column and 'Application Express Accounts' in the Scheme Type column.

The screenshot shows the 'Create Authentication Scheme' dialog box. A green circle labeled '2' is on the top right. The title is 'Create Authentication Scheme'. Below it is a section titled 'Method'. A text block explains that when creating a new authentication scheme, several options are available to reuse existing implementations or copy pretested schemes. Under 'Create Scheme:', there are two radio buttons: one selected, 'Based on a pre-configured scheme from the gallery', and another, 'As a copy of an existing authentication scheme'.

The screenshot shows the Oracle Database Control interface with the path: Application 103641 \ Shared Components \ Authentication Schemes. A green circle labeled '3' is on the top right. A message box at the top left says: '✓ Action processed. Authentication scheme activated as current authentication scheme.' The 'Authentication Schemes' tab is selected. A red box highlights the table below. The table has columns: Name, Scheme Type, Subscribed From, and Subscribers. Two rows are visible: 'Application Express Accounts' and 'Open Door Credentials - Current'. A purple box surrounds the 'Open Door Credentials - Current' row. A green circle labeled '4' is on the top right of the table area. To the right, a detailed view of the 'Open Door Credentials' row is shown in a modal window. It contains fields: 'Name' (value: 'Open Door Credentials') and 'Scheme Type' (value: 'Open Door Credentials').

Creating a Copy of an Existing Authentication Scheme

The screenshot illustrates the process of creating a copy of an existing authentication scheme in Oracle Database Control.

- Step 1:** Click the **Create >** button in the top right corner of the **Authentication Schemes** page.
- Step 2:** In the **Create Scheme** dialog, select the radio button for **As a copy of an existing authentication scheme**.
- Step 3:** Use the **Copy From Application** dropdown to select the application from which to copy the scheme. In this example, **83106 Sample Charts** is selected.
- Step 4:** In the **Copy** section of the dialog, set the **Yes** option for the **Copy** button.
- Step 5:** After clicking **Copy Scheme**, a confirmation message **Authentication scheme copied.** is displayed.

Table Data (From Step 3):

Application	From Name	To Name	Copy
83106	Application Express Accounts	Copy of Application Express Accounts	Yes

Table Data (From Step 5):

Name ↑=	Scheme Type	Subscribed From	Subscribers
Application Express Accounts - Current	Application Express Accounts		
Copy of Application Express Accounts	Application Express Accounts		

Providing Security Through Authorization

- Authorization involves controlling access to resources based on user privileges
- Use authorization schemes to implement authorization in an application
- You can apply an authorization scheme for:
 - An entire application
 - A page
 - Specific control such as a region, item, or button



How to Implement Authorization?

Oracle Application Express allows you to implement authorization to an application or its components in two different ways:

- By navigating to Shared Components and then either creating an authorization scheme from scratch or copying an existing authorization scheme
- By creating an Access Control Page and then performing the following steps:
 1. Configuring the Access Control Page:
 - a) Setting the application mode
 - b) Adding users to the Access Control List
 2. Applying the authorization scheme to application components

Creating an Authorization Scheme from Scratch

3

Create Authorization Scheme

Application: 103641 Sample Database Application

* Name: Administrator

* Scheme Type: Value of Item in Expression 1 Equals Expression 2

* Item: APP_USER

* Value: MANAGER

* Identify error message displayed when scheme violated: You do not have sufficient privileges!

Validate authorization scheme: Once per session

Once per session
 Once per page view
 Once per component
 Always (No Caching)

[Create Authorization Scheme](#)

1

Application 103641 \ Shared Components \ Authorization Schemes

Authorization Schemes Subscription by Component Utilization History

Actions: [Create >](#)

2

Create Authorization Scheme

Creation Method

Specify how you want to create your authorization scheme. You can create an authorization scheme from scratch or copy it from another application. If you copy an authorization scheme, you can modify later.

Create Authorization Scheme:

From Scratch
 As a Copy of an Existing Authorization Scheme

Understanding Access Control

- You control access to an app, individual page, or page components by creating an access control list
- When you run the Access Control Wizard, it creates multiple pages and the following components:
 - ✓ Adds an Access Control region to the Administration Page you specify
 - ✓ Creates the access roles: Administrator, Contributor, and Reader
 - ✓ Creates the authorization schemes: Administration Rights, Contribution Rights, and Reader Rights
 - ✓ Creates the build option, Feature: Access Control
 - ✓ Creates the Application Setting, ACCESS_CONTROL_SCOPE
- You use the access control list to associate the access roles with application users

Creating the Access Control Administration Page

The screenshot shows the Oracle Application Builder interface for an application named "Tasks App". A red box highlights the top navigation bar. A green circle labeled "1" points to the "Create Page >" button in the toolbar. Another green circle labeled "2" points to the "Access Control" option under the "Page Type" section of the "Create a Page" dialog.

Create a Page

Page Type

Component Feature ⓘ

About Page Access Control Activity Reporting Configuration Options Email Reporting Feedback

Login Page Theme Style Selection

Creating the Access Control Administration Page

Create Access Control Pages

administrator role and corresponding authorization scheme to your application. Apply these authorization schemes to pages and page components to manage access by user and role.

* Starting Page Number (?)

Page Group (?)

Build Option **Feature: Access Control** (?)

* Administration Page Preference **Create a new page** (?)
 Identify an existing page

* Administration Page Number (?)

* Administration Page Name (?)

Administration Page Navigation Preference Do not associate this page with a navigation menu entry (?)
 Create a new navigation menu entry (?)
 Identify an existing navigation menu entry for this page

* New Navigation Menu Entry (?)

Parent Navigation Menu Entry (?)
Home

Create Access Control Pages - Confirmation

4

You have requested to create a page with the following attributes. Please confirm your selections.

- Multiple pages will be created, starting with Page: 1001
- A new Administration page will be added, Page: 1000 - Administration
- The following Application Access Role(s) will be created:
 - Administrator
 - Contributor
 - Reader
- The following Authorization Scheme(s) will be created:
 - Administration Rights
 - Contribution Rights
 - Reader Rights
- The following Build Option(s) will be created:
 - Feature: Access Control
- The following Application Setting(s) will be created:
 - ACCESS_CONTROL_SCOPE

Configuring Access Control

Administration

Run the Administration page

1

The screenshot shows the 'Administration' page with the 'Access Control' section highlighted. A callout bubble labeled '1' points to the 'Add' button in the top right corner of the access control table. The table lists three roles: Administrator, Contributor, and Reader. The 'Reader' row contains a 'Users' link and an 'Access Control' link, both of which are highlighted with green boxes.

	Username	Roles
Administrator		
Contributor		
Reader	lowcode	Administrator

2

A modal dialog titled 'Manage User Access' is open, showing a table with one user entry: 'lowcode' with the role 'Administrator'. The 'Add User' button is highlighted with a red box. The title bar of the dialog has a red box around it, and the text 'Add users to the Access Control List' is displayed in red.

3

A modal dialog titled 'Configure Access Control' is open. It contains a question: 'Any authenticated user may access this application' with 'Yes' and 'No' buttons. A note below says: 'Choose No if all users are defined in the access control list. Choose Yes if authenticated users not in the access control list may also use this application.' The 'No' button is highlighted with a red box. The title bar of the dialog has a red box around it, and the text 'Specify the behavior when authenticated users access your app' is displayed in red.

Attaching an Authorization Scheme to an Application

1



Shared Components

2

Security

- Security Attributes
- Authentication Schemes
- Authorization Schemes
- Application Access Control
- Session State Protection
- Web Credentials

3

Definition Security Globalization User Interface

Application 53503

Show All Authentication Authorization Session Management Session State Protec... Browser Security Database Session

Define Authorization Schemes >

Authorization Scheme ✓ - No application authorization required -
Administration Rights
Contribution Rights
Must Not Be Public User
Reader Rights
{Not Administration Rights}
{Not Contribution Rights}
{Not Reader Rights}

Run on Public Pages

Source for Role or Group Schemes

History

Cancel Apply Changes

Attaching an Authorization Scheme to a Page

The screenshot shows the Oracle ADF Page Designer interface. The top navigation bar indicates the application is "Application 53503" and the current view is "Page Designer". The toolbar includes standard file operations like Save and Run.

The left sidebar displays the page structure:

- Page 2: Projects (highlighted with a green box)
- Pre-Rendering
- Regions
 - Breadcrumb Bar
 - Breadcrumb (highlighted with a green box)
 - Attributes
 - Content Body
 - Tasks
 - Columns
 - Attributes
 - Region Buttons
- RESET_REPORT

The center pane shows the page components:

- Projects
- PAGE HEADER
- PAGE NAVIGATION
- BREADCRUMB BAR
 - Breadcrumb (highlighted with a green box)
 - ITEMS
 - REGION CONTENT
 - SUB REGIONS
- Regions (highlighted with a green box)
- Items
- Buttons

The right pane is the "Page" configuration panel:

- Search bar: security
- Authorization Scheme (highlighted with a green box):
 - Select - (highlighted with a green box)
 - Administration Rights
 - Contribution Rights
 - Reader Rights
 - Must Not Be Public User
 - {Not Administration Rights}
 - {Not Contribution Rights}
 - {Not Reader Rights}
 - Unrestricted
- Authentication
- Rejoin Sessions
- Deep Linking
- Page Access Protection
- Form Auto Complete
- Browser Cache: Application Default
- Off

Attaching an Authorization Scheme to a Control or Component

The screenshot shows the Oracle APEX Page Designer interface. On the left, the page structure is defined with regions like 'Breadcrumb Bar' and 'Content Body'. In the 'Content Body' region, a 'Tasks' section contains a 'Columns' section. A specific column, 'STATUS', is highlighted with a green border. On the right, the 'Authorization Scheme' section is expanded, showing a search bar with 'security' and a list of rights. The 'Select' dropdown is open, displaying options such as 'Administration Rights', 'Contribution Rights', 'Reader Rights', and several conditions involving these rights.

Application 53503 \ Page Designer

Column

security

Page 2: Projects

Pre-Rendering

Regions

- Breadcrumb Bar
- Breadcrumb

Attributes

Content Body

- Tasks
- Columns

ID

PROJECT

TASK_NAME

START_DATE

END_DATE

STATUS

ASSIGNED_TO

Regions

Items

Buttons

Breadcrumb

Calendar

Chart

Classic Report

Authorization Scheme

Escape special characters

- Select -

- Administration Rights
- Contribution Rights
- Reader Rights
- Must Not Be Public User
- {Not Administration Rights}
- {Not Contribution Rights}
- {Not Reader Rights}

Summary

In this lesson, you learned how to:

- Create and use an authentication scheme for your application
- Create and apply an authorization scheme to your application, page, or components





Hands-on Lab



Oracle **APEX**



Unit 14: Adding Additional Pages to your Application

Lesson Objectives

After completing this lesson, you should be able to create, customize, and use the following in your application:

- Oracle JET Charts
- Calendars
- Trees

Creating Charts



Area



Bar



Box Plot



Bubble



Combination



Stock



Reference



Data Densification



Dashboard



Dashboard - Pie Charts



Funnel



Gantt



Legend Plug-in



Line



Line with Area



Pie



Polar



Pyramid



Radar



Range

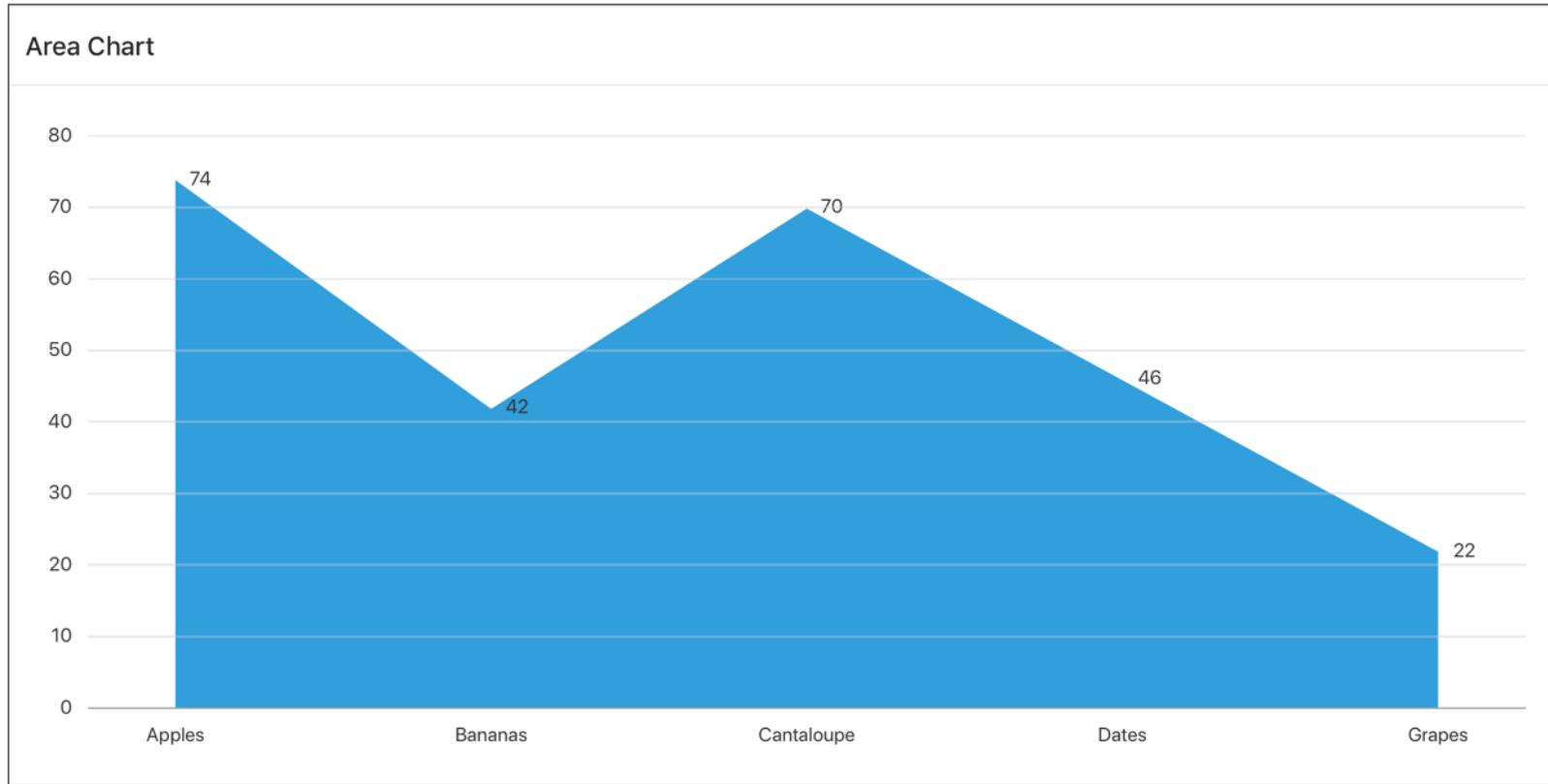


Scatter



Status Meter
Gauge

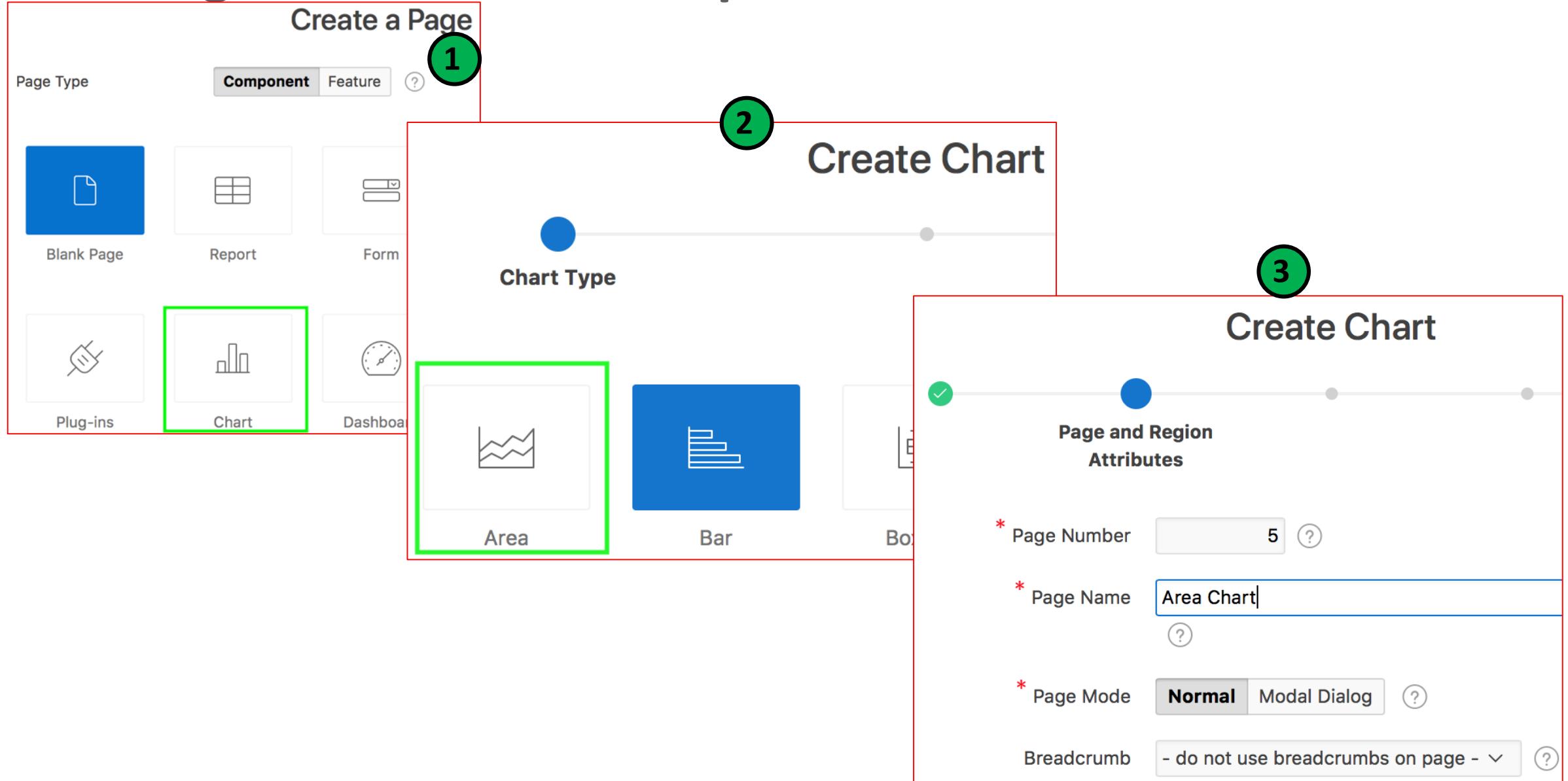
Creating Charts: Area Chart



SQL query used for the chart series

```
select a.product_name, b.quantity, b.customer, a.product_description  
from eba_demo_chart_products a, eba_demo_chart_orders b  
where a.product_id = b.product_id  
and b.customer = 'Deli'
```

Creating an Area Chart: Steps



Creating an Area Chart: Steps

4

Create Chart

Navigation Menu

Navigation Preference Do not associate this page with a navigation menu entry Create a new navigation menu entry Identify an existing navigation menu entry

5

Create Chart

Source

Location Local Database

Source Type Table **SQL Query**

* SQL Query

```
1 select a.product_name, b.quantity, b.customer, a.product_description
2 from eba_demo_chart_products a, eba_demo_chart_orders b
3 where a.product_id = b.product_id
4 and b.customer = 'Deli'
```

Page Items to Submit

Maximum Rows

6

Create Chart

Column Mapping

Chart Type: **Area**

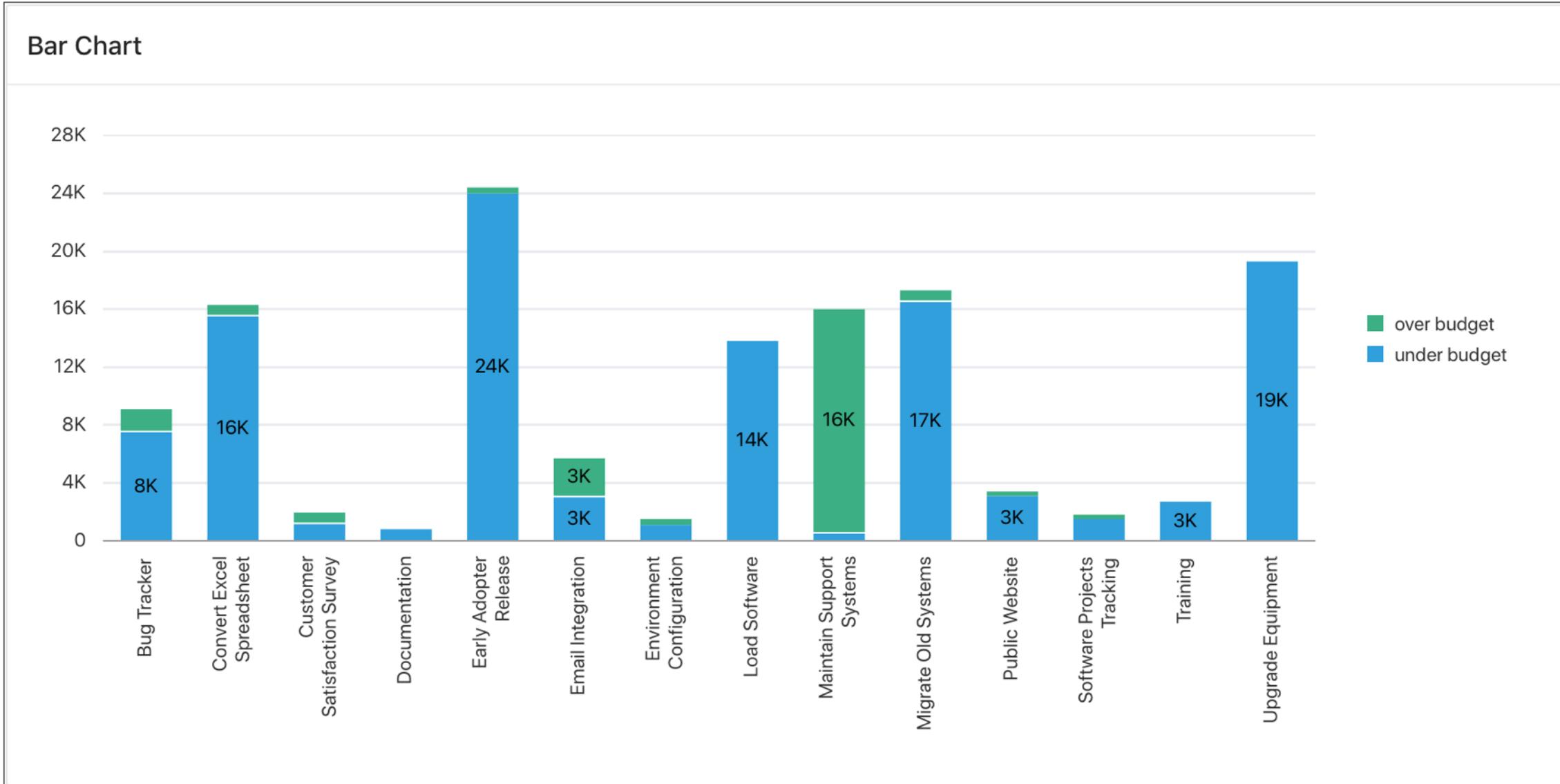
Orientation Vertical

* Label Column PRODUCT_NAME

* Value Column QUANTITY

Create

Creating Charts: Bar Chart



Creating a Bar Chart: Steps

Create Chart

1

Chart Type

Area

Bar

Create Chart

2

Page and Region Attributes

* Page Number: 8

* Page Name: Bar Chart

* Page Mode: Normal

Breadcrumb: - do not use breadcrumbs on page -

Create Chart

3

Navigation Menu

Navigation Preference

Do not associate this page with a navigation menu entry

Create a new navigation menu entry

Identify an existing navigation menu entry for this page

This diagram illustrates the three steps involved in creating a bar chart. It shows a sequence of three panels, each with a red border and a green circular step indicator (1, 2, or 3) at the top right. Step 1, 'Chart Type', shows two options: 'Area' and 'Bar', with 'Bar' selected and highlighted by a green border. Step 2, 'Page and Region Attributes', includes fields for 'Page Number' (set to 8), 'Page Name' (set to 'Bar Chart'), and 'Page Mode' (set to 'Normal'). Step 3, 'Navigation Preference', contains a single option: 'Do not associate this page with a navigation menu entry', which is selected and highlighted by a green border. The panels are connected by a horizontal dotted line with circular markers.

Creating a Bar Chart: Steps

4

Create Chart

Source

Location: Local Database

Source Type: Table **SQL Query**

* SQL Query

```
1 select id,
2       project as label,
3       NVL((select sum(t.budget) from eba_demo_chart_tasks t where t.project = p.id
4             'under budget' as series ,
5             'green' as color
6       from eba_demo_chart_projects p
7     union all
8   select id,
9         project as label,
10        NVL((select sum(t.budget) from eba_demo_chart_tasks t where t.project = p.id
```

5

Create Chart

Column Mapping

Chart Type: **Bar**

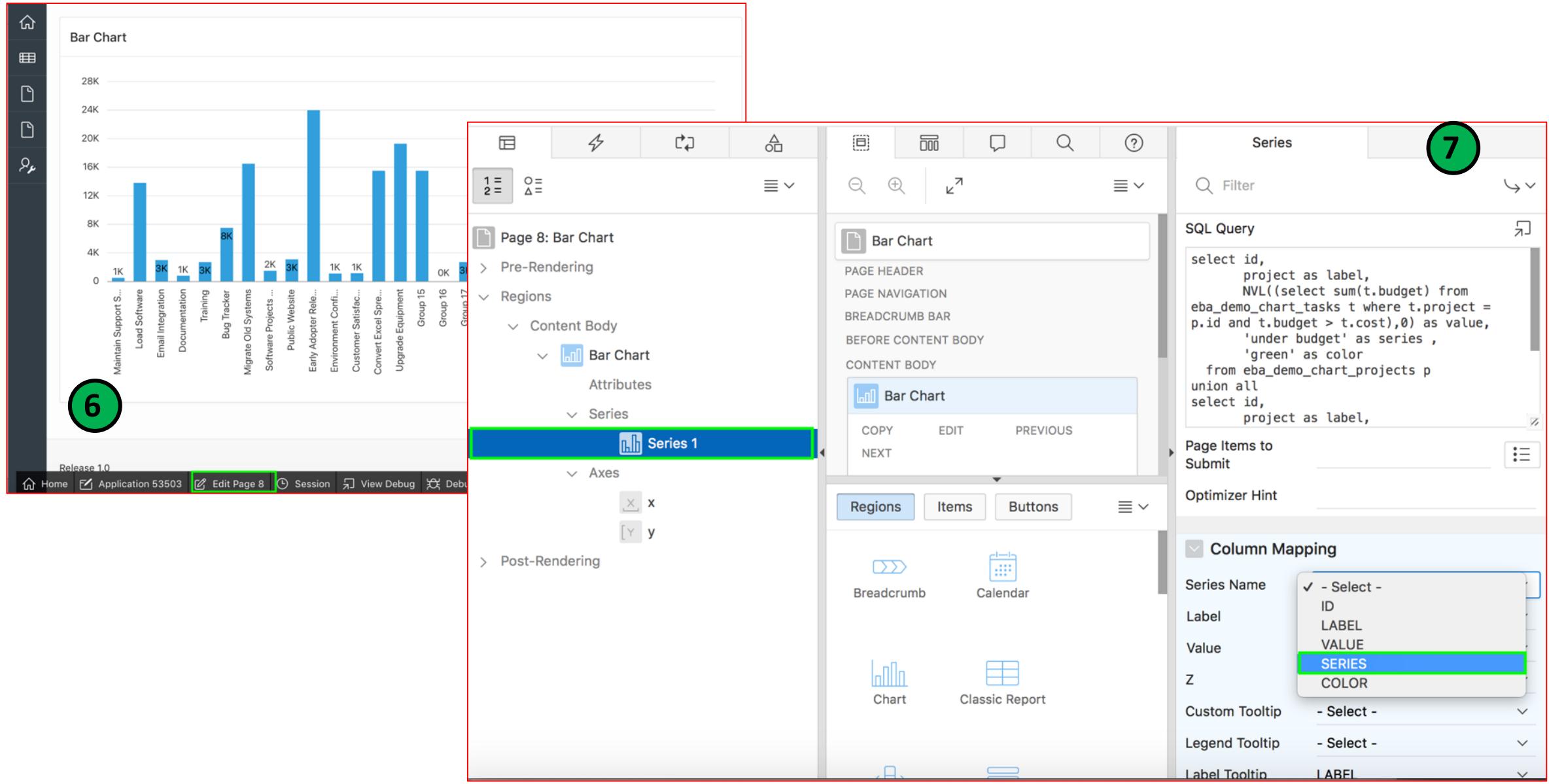
Orientation: Vertical

* Label Column: LABEL

* Value Column: VALUE

< Cancel Create

Creating a Bar Chart: Steps



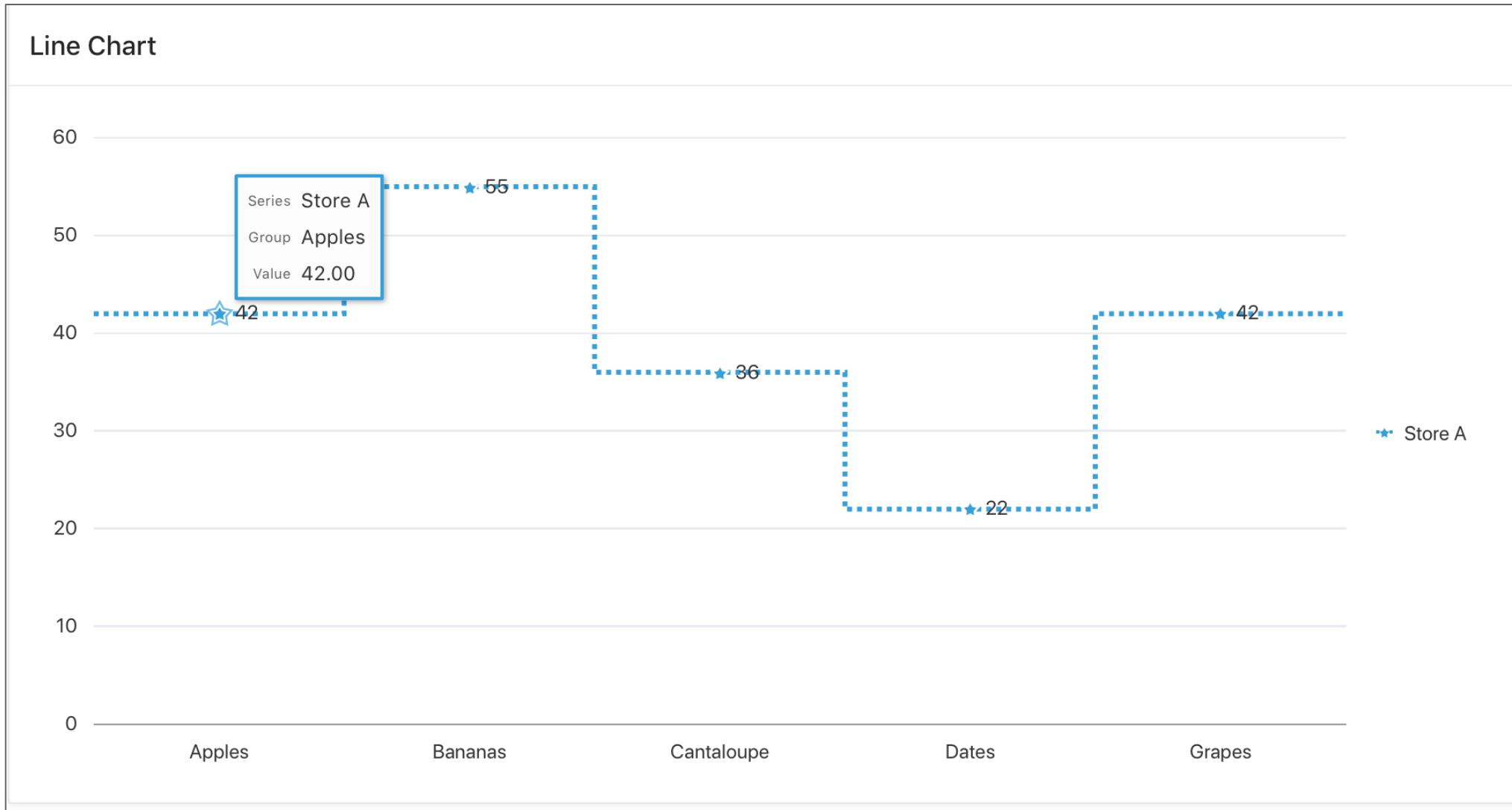
Editing Chart Attributes

The screenshot shows the Oracle APEX interface for editing chart attributes. On the left, the page navigation bar includes icons for refresh, search, and help. Below it, the breadcrumb trail shows "Page 8: Bar Chart". The main content area displays the "Bar Chart" component's attributes. The "Attributes" section on the right is expanded, showing various settings:

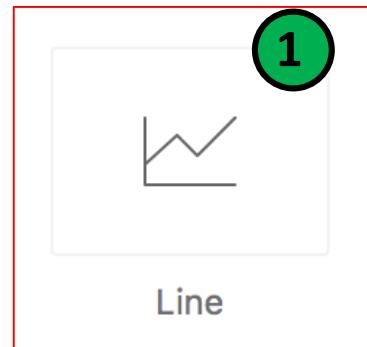
- Appearance**: Orientation is set to "Vertical". Stack is set to "Yes" (highlighted with a green box). Stack Label is set to "No".
- Layout**: Not currently expanded.
- Data Cursor**: Not currently expanded.
- Multi-Series Chart Data**: Not currently expanded.
- Settings**: Not currently expanded.
- Animation**: Not currently expanded.
- Tooltip**: This section is expanded, showing "Show" set to "Yes" (highlighted with a green box) and "Title" set to an empty field.
- Legend**: Not currently expanded.

The central panel shows the "Bar Chart" component's regions: PAGE HEADER, PAGE NAVIGATION, BREADCRUMB BAR, BEFORE CONTENT BODY, and CONTENT BODY. The CONTENT BODY region contains a "Bar Chart" item with options to COPY, EDIT, and navigate to PREVIOUS and NEXT items. Below this are buttons for Regions, Items, and Buttons, followed by Breadcrumb and Calendar items. At the bottom are Chart and Classic Report items.

Creating Charts: Line Chart



Creating a Line Chart: Steps



✓

Navigation Preference

Do not associate this page with a navigation menu entry
 Create a new navigation menu entry
 Identify an existing navigation menu entry for this page

3

Navigation Menu

✓

2

Page and Region Attributes

* Page Number

* Page Name

* Page Mode

Breadcrumb

4

Create Chart

✓ ✓ ✓

Source

Location

Source Type

* SQL Query

SQL Query Editor:

```
1 select a.product_name, b.quantity, b.customer
2 from eba_demo_chart_products a, eba_demo_chart_orders b
3 where a.product_id = b.product_id
4 and customer = 'Store A'|
```

Creating a Line Chart: Steps

5

Create Chart

Column Mapping

Chart Type: **Line** [?](#)

Orientation: **Vertical** [?](#)

* Label Column: **PRODUCT_NAME** [?](#)

* Value Column: **QUANTITY** [?](#)

< Cancel **Create**

6

Line Chart

A line chart titled "Line Chart" showing the quantity of different fruits. The x-axis lists products: Apples, Bananas, Cantaloupe, Dates, and Grapes. The y-axis ranges from 0 to 60. A blue line connects data points with values: Apples (42), Bananas (55), Cantaloupe (36), Dates (22), and Grapes (42). The chart has a light gray background with horizontal grid lines at intervals of 10.

Product	Quantity
Apples	42
Bananas	55
Cantaloupe	36
Dates	22
Grapes	42

Release 1.0

Home Application 53503 Edit Page 10 Session View Debug Debug Page Info Quick Edit Theme Roller

Altering the Display of a Line Chart

The screenshot illustrates the process of altering the display of a line chart in Oracle Page Designer. It consists of two main panels:

Left Panel (Component Tree and Properties):

- Component Tree:** Shows the page structure with "Page 10: Line Chart" selected.
- Properties Panel:** Displays the "Attributes" tab for the selected "Line Chart" component. Key settings include:
 - Orientation:** Vertical
 - Stack:** Yes (highlighted with a green box)
 - Connect Null Data Points:** Yes
- Legend:** Show is set to Yes.

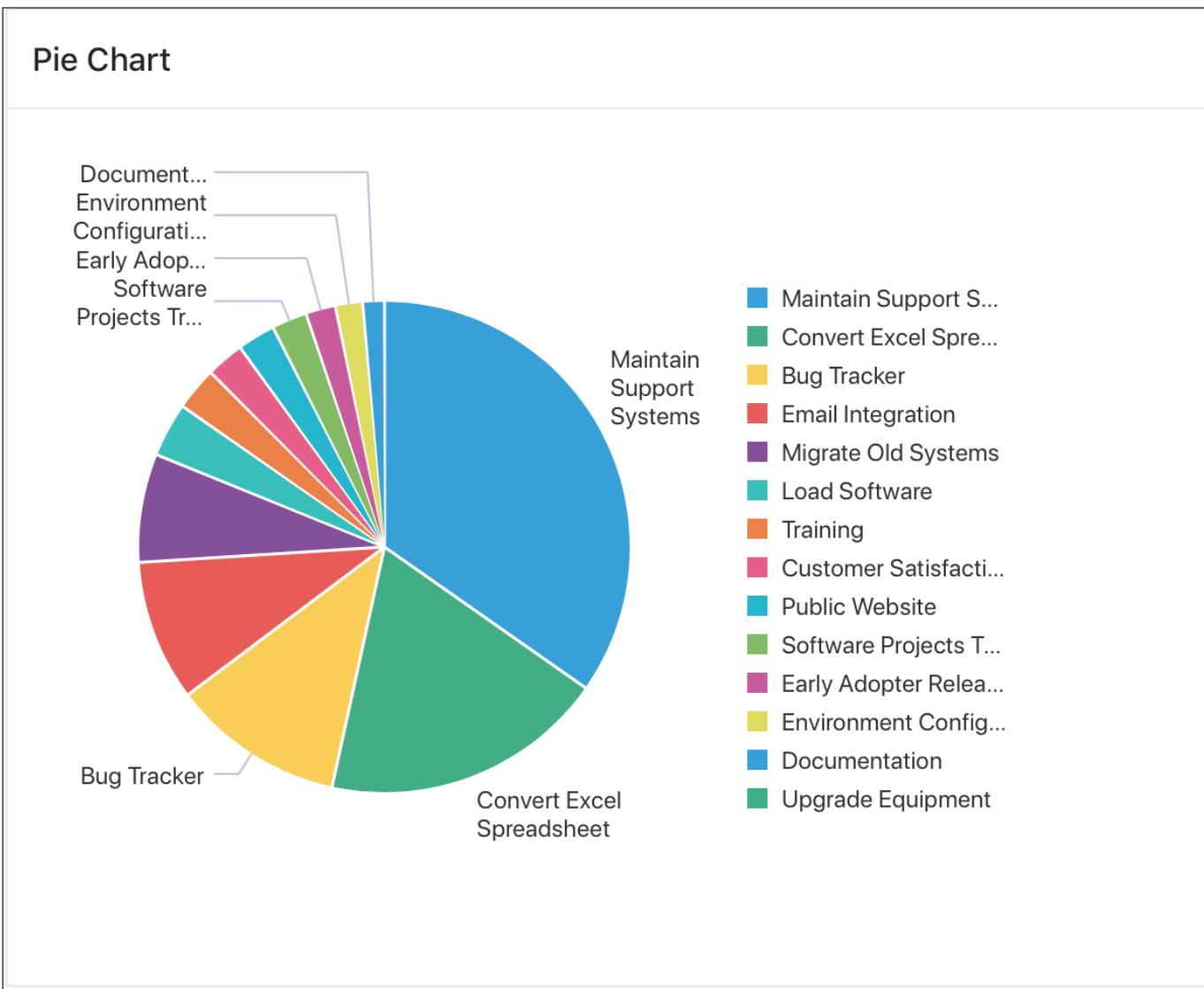
Right Panel (Chart Properties):

- Properties Panel:** Displays the "Series" tab for the "Line" component. Key settings include:
 - Style:** Dotted
 - Type:** Centered Stepped (highlighted with a green box)
 - Marker:** Show is Yes, Shape is Star (highlighted with a green box)
 - Label:** Show is Yes (highlighted with a green box)

Annotations:

- Circle 1:** A green circle with the number 1 is located on the left side of the left panel's red border.
- Circle 2:** A green circle with the number 2 is located on the right side of the right panel's red border.

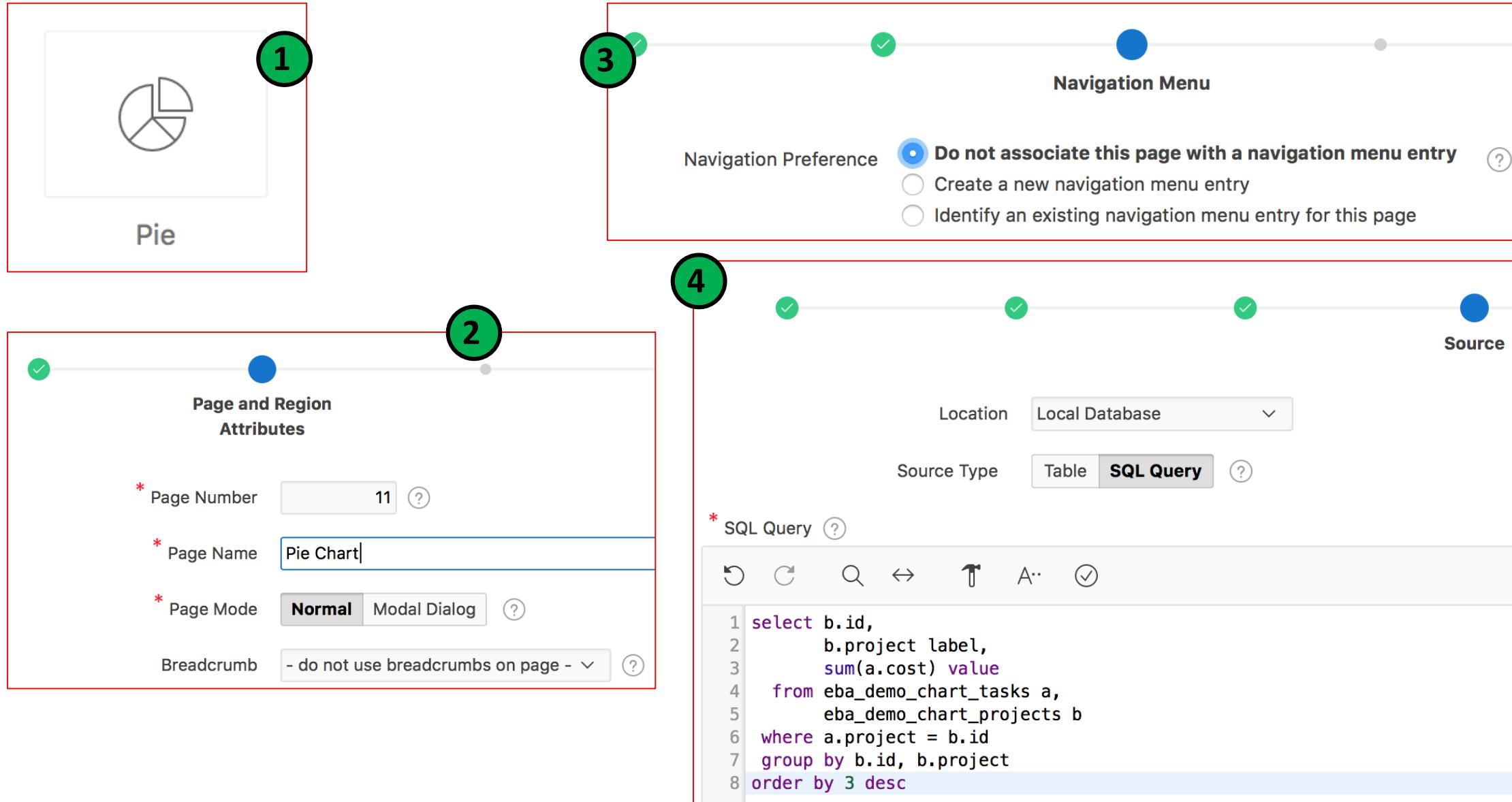
Creating Charts: Pie Chart



```
select b.id,  
       b.project_label,  
       sum(a.cost) value  
  from eba_demo_chart_tasks a,  
       eba_demo_chart_projects b  
 where a.project = b.id  
 group by b.id, b.project  
 order by 3 desc
```

SQL query used for the chart series

Creating a Pie Chart: Steps



Creating a Pie Chart: Steps

Create Chart

Three green checkmarks indicate completed steps.

Chart Type: **Pie** (?)

* Label Column: **LABEL**

* Value Column: **VALUE**

Buttons: < Cancel Create

Step 5 is highlighted with a red border and a green circle containing the number 5.

Pie Chart

Task	Value (approx.)
Maintain Support Systems	35%
Convert Excel Spreadsheet	15%
Bug Tracker	10%
Email Integration	8%
Migrate Old Systems	7%
Load Software	5%
Training	4%
Customer Satisfaction Survey	3%
Public Website	2%
Software Projects Tracking	2%
Early Adopter Release	1%
Environment Configuration	1%
Documentation	1%

Release 1.0

Home Application 53503 Edit Page 11 Session View Debug Debug Page Info Quick Edit Theme Roller

Step 6 is highlighted with a red border and a green circle containing the number 6.

Altering the Display of a Pie Chart

The screenshot shows the Oracle Page Designer interface with a pie chart selected in the regions tree. The right panel displays the chart's attributes and settings.

Specifying the effect applied when a pie is selected: The "Pie Selection Effect" dropdown is set to "Highlight And Explode".

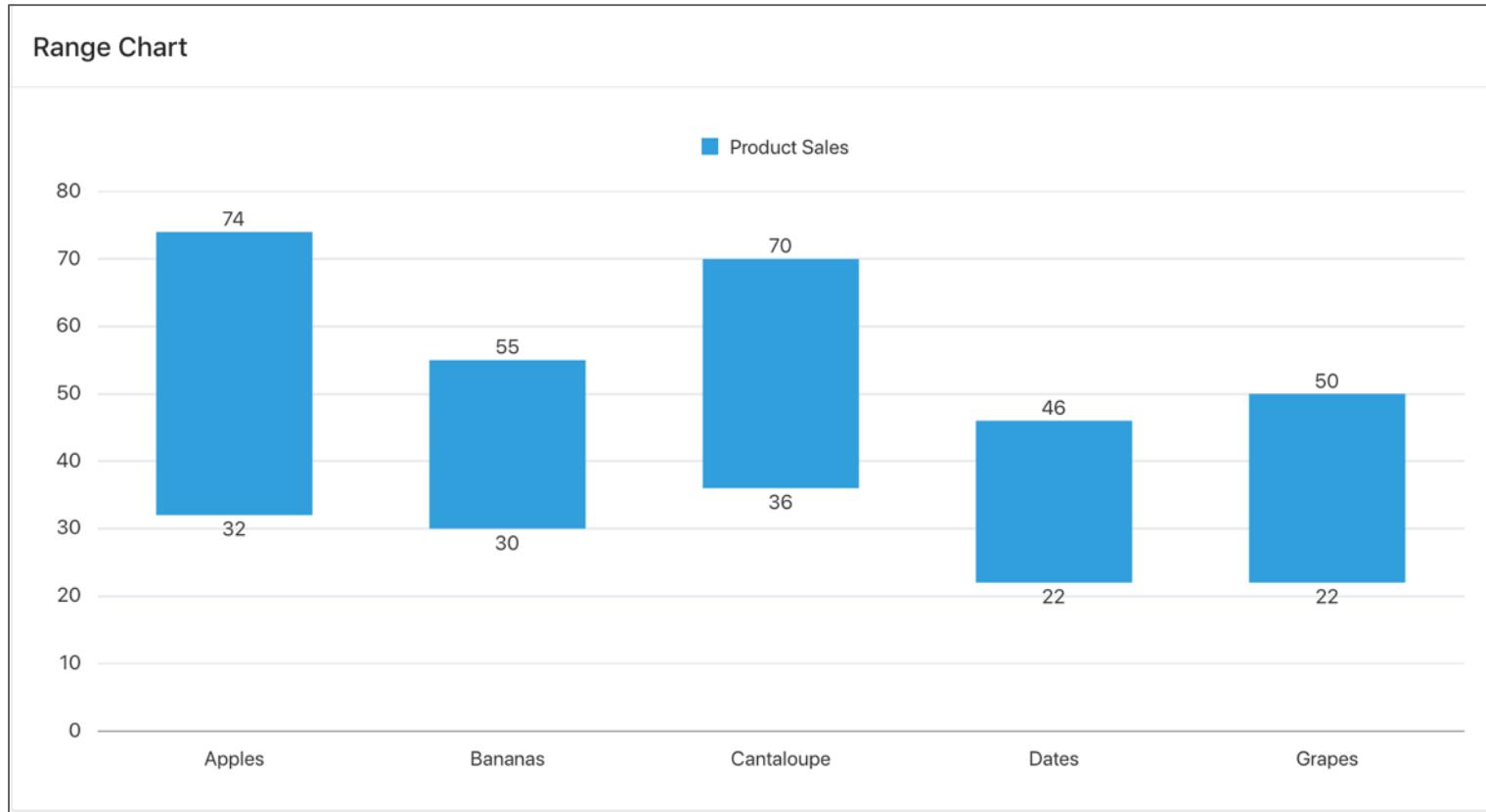
Selecting behavior for hovering over data items: The "Dim On Hover" checkbox is unchecked ("No").

Selecting hide and show behavior for legend items: The "Hide and Show Behavior" dropdown is set to "Rescale".

Text annotations:

- Select the behavior to be applied when hovering over data items on the chart.
- Select the hide and show behavior that is performed when clicking on a legend item
- Specify the effect applied when a pie is selected in the chart

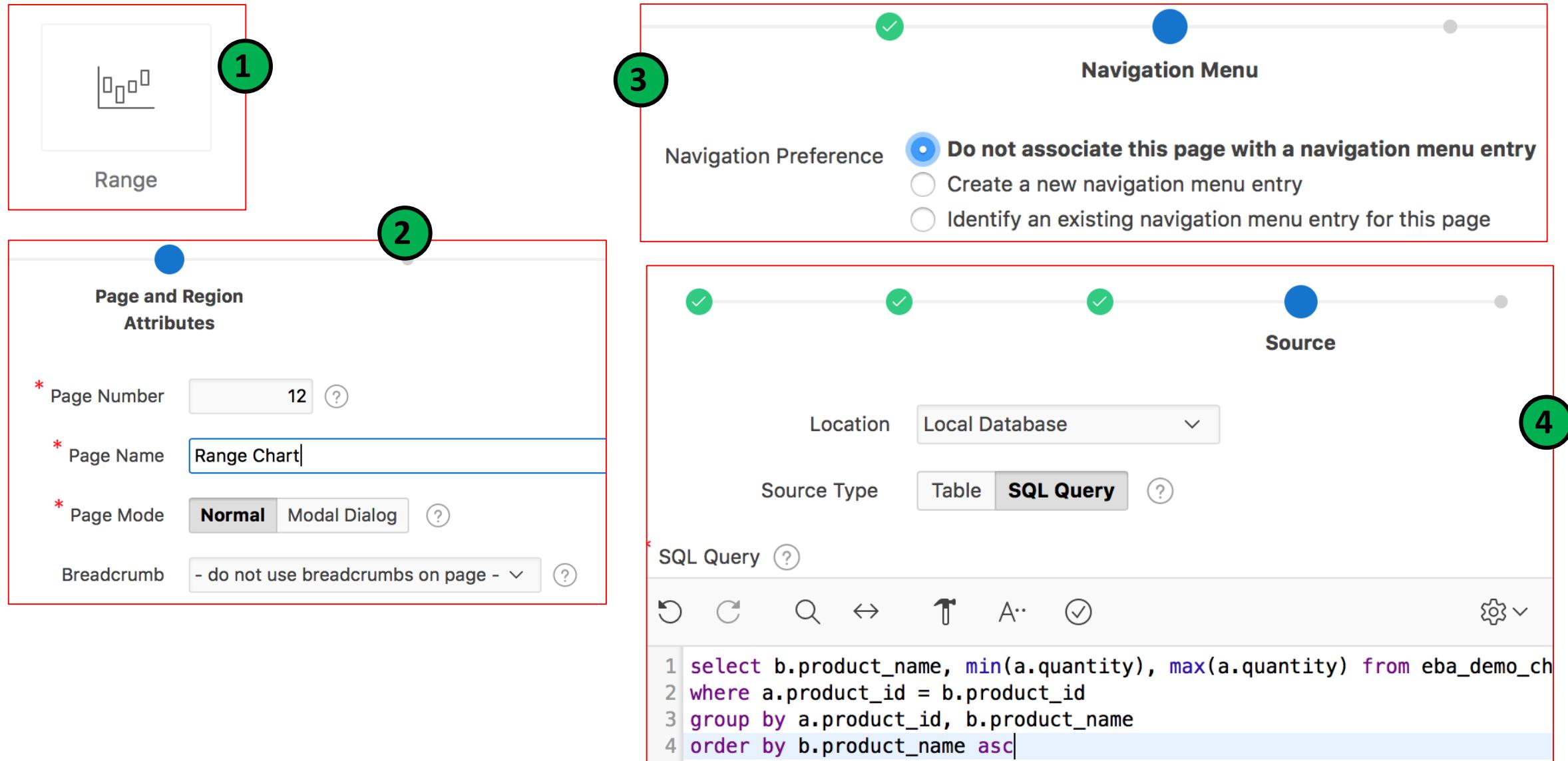
Creating a Range Chart: Example

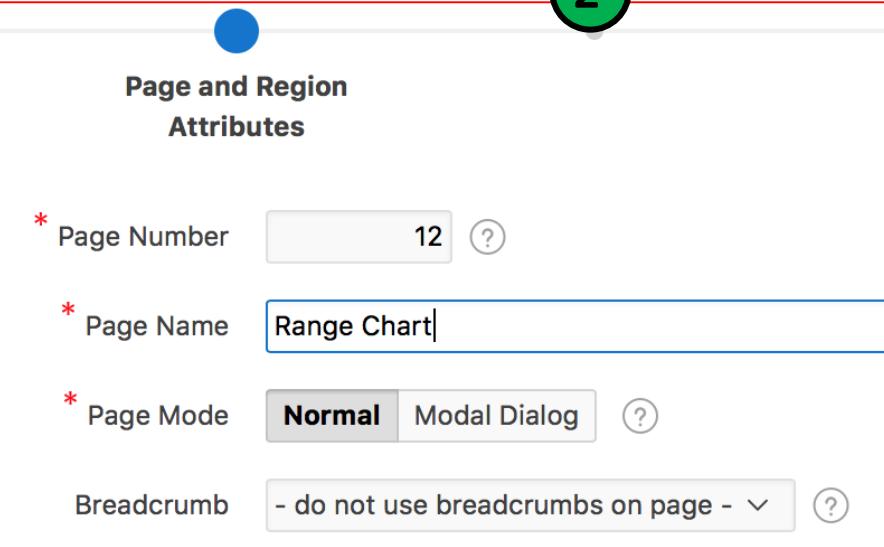


```
select b.product_name, min(a.quantity),  
max(a.quantity) from eba_demo_chart_orders a,  
eba_demo_chart_products b  
where a.product_id = b.product_id  
group by a.product_id, b.product_name  
order by b.product_name asc
```

SQL query used for the chart

Creating a Range Chart: Steps



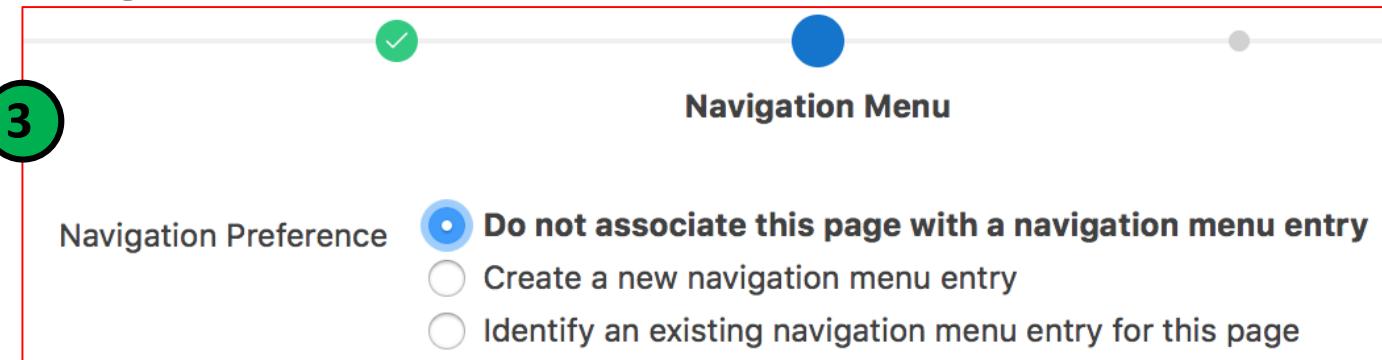
- 1  Range
- 2 

Page and Region Attributes

* Page Number: 12 

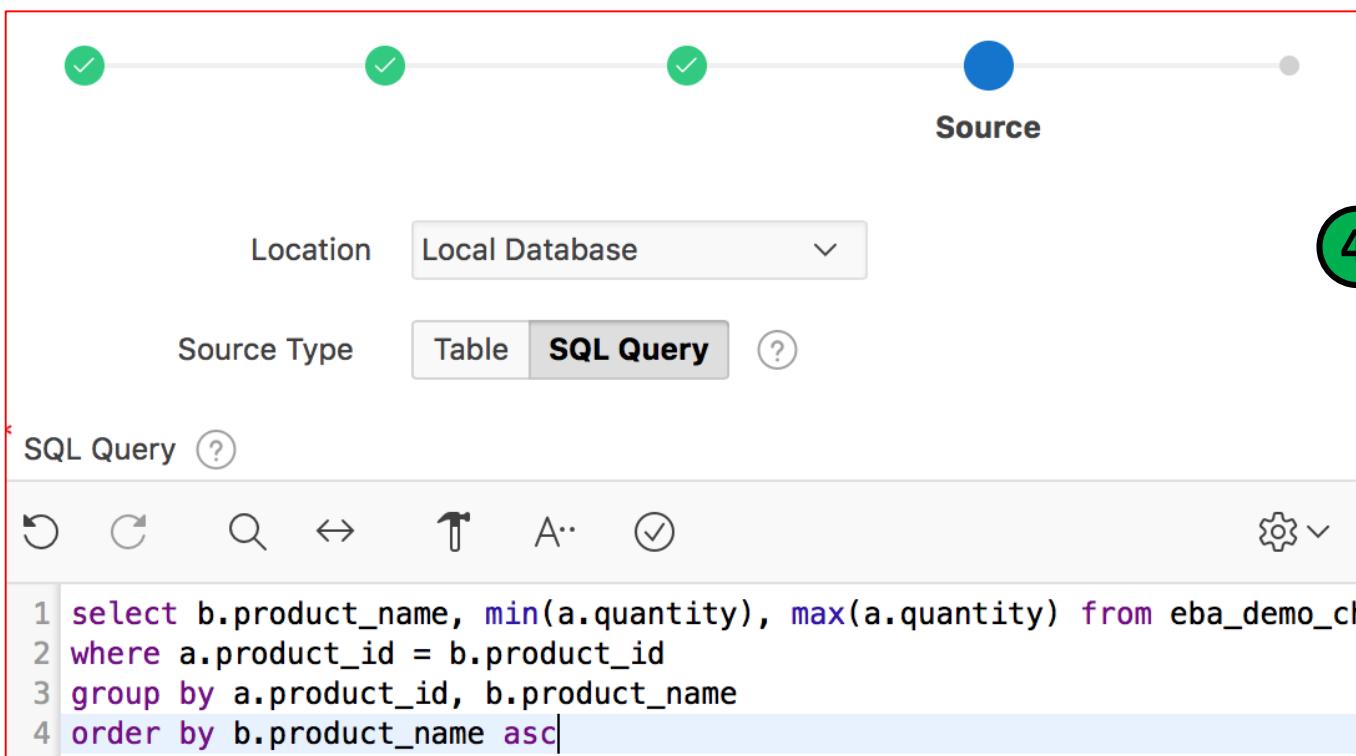
* Page Name: Range Chart 

* Page Mode: **Normal**  

Breadcrumb: - do not use breadcrumbs on page - 
- 3 

Navigation Preference

 - Do not associate this page with a navigation menu entry
 - Create a new navigation menu entry
 - Identify an existing navigation menu entry for this page

Navigation Menu  
- 4 

Source

Location: Local Database 

Source Type: **SQL Query**  

SQL Query 

     A..   

```
1 select b.product_name, min(a.quantity), max(a.quantity) from eba_demo_ch
2 where a.product_id = b.product_id
3 group by a.product_id, b.product_name
4 order by b.product_name asc|
```

Creating a Range Chart: Steps

The screenshot shows a user interface for creating a chart. A red box highlights the configuration area, and a red line connects two green circles labeled 5 and 6.

5 Create Chart

Chart Type: **Range** (?)

Orientation: **Vertical** (?)

* Label Column: **PRODUCT_NAME** (?)

* Low Value Column: **MIN(A.QUANTITY)** (?)

* High Value Column: **MAX(A.QUANTITY)** (?)

Buttons: < Cancel Create

6

Range Chart

Category	Low Value	High Value
Apples	32	74
Bananas	30	55
Cantaloupe	36	70
Dates	22	46
Grapes	22	50

Release 1.0

Home Application 53503 Edit Page 12 Session View Debug Debug Page Info Quick Edit Theme Roller

Customizing the Display of a Range Chart

The image shows a user interface for customizing a range chart. On the left, a sidebar navigation menu is open, showing the current page is "Page 12: Range Chart". Under "Regions", the "Content Body" section is expanded, showing a "Range Chart" item which is selected and highlighted with a blue background. This item has a sub-menu with options: COPY, EDIT, PREVIOUS, and NEXT. Below this are "Attributes" and "Series". The "Series" section contains a single item named "Product Sales", which is also highlighted with a blue background. A green circle with the number "1" is positioned over the "Product Sales" item in the series list. To the right of the sidebar, there is a main content area with a toolbar at the top containing various icons. Below the toolbar, there is a search bar and a "Series" section. The "Series" section includes a "Filter" input field and a dropdown menu. A green circle with the number "2" is positioned over the "Top" option in the "Position" dropdown menu.

Page 12: Range Chart

Pre-Rendering

Regions

- Content Body
 - Range Chart
 - Attributes
 - Series
 - Product Sales

Post-Rendering

Series

Identification

- Name: Product Sales
- Type: Bar Range

Execution Options

Source

Column Mapping

Link

Appearance

Label

- Show: Yes
- Position: Outside Bar Edge

Attributes

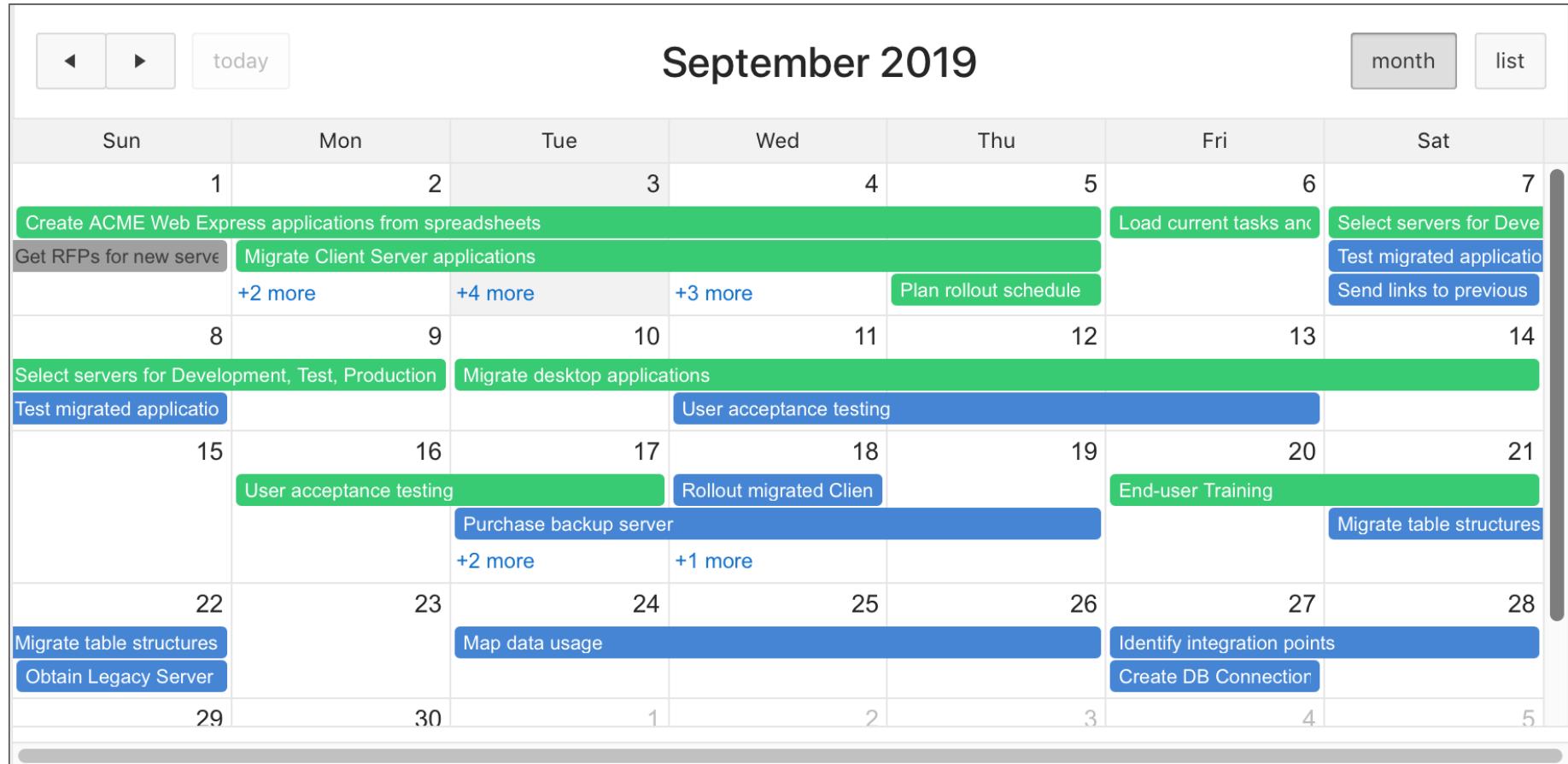
Filter

Legend

- Show: Yes
- Title:
- Position: Top

Creating Calendars

- Oracle Application Express has an integrated calendaring component
- Calendars in APEX are based on a table or SQL query you provide



Creating a Calendar

1

Create a Page

Page Type

Component Feature ?

Blank Page Report Form Master Detail Plug-ins Chart

Dashboard **Calendar** Tree Wizard Data Loading Legacy Page

2

Create Page

Page Attributes

Region Type **Calendar**

* Page Number 14 ?

* Page Name **Calendar**

* Page Mode **Normal** Modal Dialog ?

Page Group - Select Page Group - ?

Breadcrumb - do not use breadcrumbs on page - ?

3

Create Page

Navigation Menu

Navigation Preference **Do not associate this page with a navigation menu entry** Create a new navigation menu entry Identify an existing navigation menu entry for this page

Creating a Calendar

Create Page

Source

Data Source Local Database

Source Type Table SQL Query

* Table / View Owner LOWCODE

* Table / View Name **TASKS (table)**

* Select Columns

ID (Number)
PROJECT (Varchar2)
TASK_NAME (Varchar2)
START_DATE (Date)
END_DATE (Date)
STATUS (Varchar2)
ASSIGNED_TO (Varchar2)
COST (Number)
BUDGET (Number)

SQL Examples

Next >

4

Create Page

Settings

Display Column **TASK_NAME**

Start Date Column START_DATE

End Date Column **END_DATE**

Show Time No

< Cancel Create

5

Editing Calendar Attributes in Property Editor

The screenshot illustrates the process of creating and editing links for a calendar item in the Oracle APEX Page Designer.

Page Designer View:

- Left Sidebar:** Shows the application structure with "Page 14: Calendar" selected.
- Content Area:** Displays the "Calendar" item under "Regions".
- Toolbar:** Includes standard page navigation and save buttons.
- Attributes Panel:** The "Attributes" tab is selected, showing the following settings:
 - Display Column:** TASK_NAME
 - Start Date Column:** START_DATE
 - End Date Column:** END_DATE
 - Primary Key Column:** - Select -
 - Show Time:** Yes (radio button selected)
 - Supplemental Information:** Includes options for Multiple Line Events (Yes), Show Tooltip (Yes), Additional Calendar Views (List, Navigation checked), Create Link (No Link Defined), and View / Edit Link (No Link Defined).

Link Builder - Create Link (Top Right):

- Target:** Page in this application, Page 16.
- Set Items:** P16_ID =&P16_ID.
- Buttons:** Cancel, Clear, OK (highlighted).

Link Builder - View / Edit Link (Bottom Right):

- Target:** Page in this application, Page 16.
- Set Items:** P16_ID =&ID.
- Clear Session State:** Clear Cache, 16.
- Buttons:** Cancel, Clear, OK (highlighted).

Enabling the Dragging and Dropping of Data: Example

Calendar

September 2019

Sun Mon Tue Wed Thu Fri Sat

1	2	3	4	5	6	7
Get RFPs for new server		Migrate Client Server applications				
Create ACME Web Express applications from spreadsheets						
Customize Software Projects software			Configure Workspace provisioning			Load current tasks and enhancements
Monitor participation			Enter base data (Projects, Milestones, etc.)			Load current tasks and enhancements Sep 6, 2019
Develop web pages			Specify security authentication scheme(s)			
8	9	10	11	12	13	14
Select servers for Development, Test, Production		Migrate desktop applications				
User acceptance testing						
15	16	17	18	19	20	21

Drag and drop

Calendar

September 2019

Sun Mon Tue Wed Thu Fri Sat

1	2	3	4	5	6	7
Get RFPs for new server		Migrate Client Server applications				Load current tasks and enhancements
Create ACME Web Express applications from spreadsheets						
Customize Software Projects software			Configure Workspace provisioning			Load current tasks and enhancements Sep 7, 2019
Monitor participation			Enter base data (Projects, Milestones, etc.)			
Develop web pages			Specify security authentication scheme(s)			
8	9	10	11	12	13	14
Select servers for Development, Test, Production		Migrate desktop applications				User acceptance testing
Test migrated applications						
Send links to previous spreadsheet owners						
15	16	17	18	19	20	21

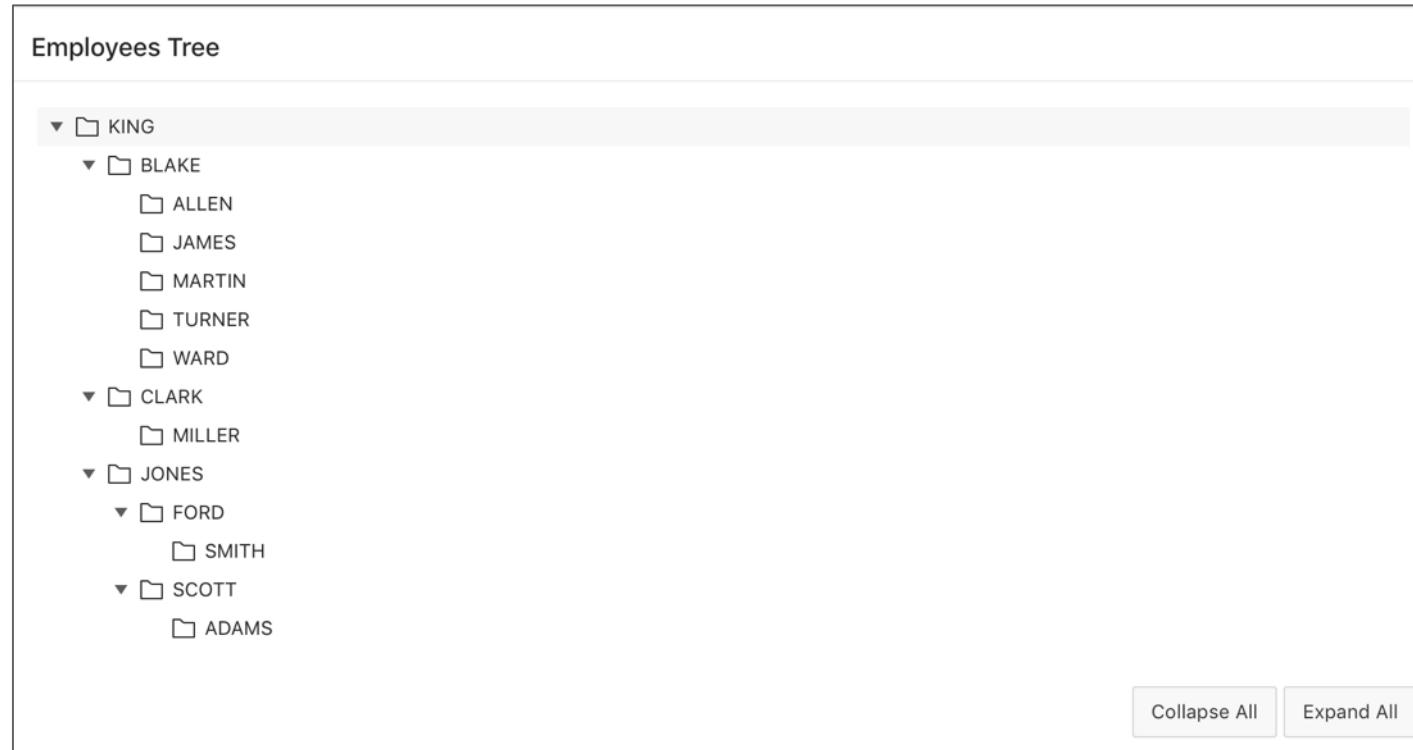
Enabling the Dragging and Dropping of Data: Steps

The screenshot shows the Oracle APEX page builder interface. A red box highlights the 'Attributes' tab for a 'Calendar' item in the 'Content Body' region. Inside the 'Attributes' tab, the 'Drag and Drop' setting is configured to 'Yes'. A green box highlights the 'ID' field under 'Primary Key Column'. A red arrow points from the 'Drag and Drop' setting to a PL/SQL code block at the bottom of the screen.

```
begin
    update sample$tasks_ss
        set start_date = to_date(:APEX$NEW_START_DATE, 'YYYYMMDDHH24MISS'),
            end_date = to_date(:APEX$NEW_END_DATE, 'YYYYMMDDHH24MISS')
        where id = :APEX$PK_VALUE;
end;
```

Creating Trees

- A tree is a hierarchical navigation mechanism
- Trees are implemented using a single hierarchical query that identifies the row to be used as the start of your query, and the relationship between parent rows and child rows of the hierarchy



```
select case when connect_by_isleaf = 1 then 0
           when level = 1 then 1
           else -1
      end as status,
      level,
      "ENAME" as title,
      null as icon,
      "EMPNO" as value,
      "ENAME" as tooltip,
      'f?p=&APP_ID.:6'||:APP_SESSION||'::::P6_EMPNO:'||"EMPNO" as link
from "#OWNER#"."EMP"
start with "MGR" is null
connect by prior "EMPNO" = "MGR"
order siblings by "ENAME"
```

Creating a Tree: Example

1

Create a Page

Page Type **Component** Feature ?

Blank Page Report Form Master Detail

Dashboard Calendar **Tree** Wizard

2

Create Tree

Page Attributes

This wizard creates a tree. A tree is a hierarchical navigation mechanism. Trees are implemented using a single hierarchical query that identifies the row to be used as the start of your query, and the relationship between parent rows and child rows of the hierarchy.

* Page Number 8 ?

* Page Name Tree Page ?

* Page Mode **Normal** Modal Dialog ?

Page Group - Select Page Group - ?

Region Template Standard ?

* Region Name Employees Tree ?

Breadcrumb - do not use breadcrumbs on page - ?

3

Create Tree

Navigation Menu

Navigation Preference **Do not associate this page with a navigation menu entry**
 Create a new navigation menu entry
 Identify an existing navigation menu entry for this page

Creating a Tree: Example

Create Tree

Table / View Owner and Name

Select the owner of the table or view from which you want to draw the tree.

* Table / View Owner: LOWCODE

* Table / View Name: EMP (table)

Create Tree

A tree is based on a query and returns data that can be represented in a hierarchy. A **start with .. connect by** clause will be used to generate the hierarchical query for your tree. Use this page to identify the column you want to use as the ID, the Parent ID, and text that should appear on the nodes. The Start With column will be used to specify the root of the hierarchical query, and its value can be based on an existing item, static value or SQL query returning a single value.

Create Tree

Where and Order by

Identify an optional where clause and order siblings by column for your query.

Where Clause (for example ename = 'JONES'):

Order Siblings By (for example ENAME): ENAME (Varchar2)

Create Tree

ID: EMPNO (Number)

Parent ID: MGR (Number)

Node Text: ENAME (Varchar2)

Start With: MGR (Number)

Start Tree: Value is NULL

4

5

6

Creating a Tree: Example

Create Tree

Identify the button, tooltip and link attributes you want to define on your tree. To make this easier, we've pre-selected some common values.

Existing Application Item.

Include Buttons: Collapse All Expand All [?](#)

Selected Node Page Item: [:](#) [?](#)

Tooltip: Database Column

* Tooltip Column: ENAME (Varchar2) [?](#)

Link Option: Nothing [?](#) Existing Application Item [?](#)

* Link Page: 6 Employee Details [?](#)

* Link Item: Page: 6: P6_EMPNO [:](#) [?](#)

Tree Attributes

8

You have requested to create a tree page with the following attributes. Please confirm your selections.

Application	116491
Page	8
Page Name	Tree Page
Region Title	Employees Tree
Region Template	Standard

< Cancel Create

Modifying a Tree

Employees Tree

- ▼ KING
 - ▼ BLAKE
 - ALLEN
 - JAMES
 - MARTIN
 - TURNER
 - WARD
 - ▼ CLARK
 - MILLER

Employees Tree

- ▼ KING
 - ▼ BLAKE
 - ALLEN
 - MARTIN
 - TURNER
 - WARD
 - ▼ CLARK
 - JAMES
 - MILLER
- ▼ JONES
 - FORD
 - SCOTT

Collapse All

Expand All

Employee Details

Name
JAMES

Job
CLERK

Manager

Hire Date

BLAKE

7698

Employee Details

Name
JAMES

Job
CLERK

Manager

Hire Date
03-DEC-81

Salary

950

Commission

30

Department

Apply Changes

Cancel

Delete

Summary

In this lesson, you learned how to create and customize:

- Oracle JET Charts
- Calendars
- Trees





Hands-on Lab



Oracle **APEX**



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

Lesson Objectives

After completing this lesson, you should be able to:

- Describe Dynamic Actions
- Create and use Dynamic Actions
- Describe Plug-ins
- Implement Plug-ins in your application

Understanding Dynamic Actions

Dynamic Action ensures that the Commission and Hire date fields are enabled only if the Job is SALESMAN

* Employee Number	7839
Name	KING
Job	PRESIDENT ▾
Manager	
Hire date	17-NOV-1981 
Salary	5000
Commission	
Department	ACCOUNTING ▾



* Employee Number	7839
Name	KING
Job	SALESMAN ▾
Manager	
Hire date	17-NOV-1981 
Salary	5000
Commission	
Department	ACCOUNTING ▾

Implementing a Dynamic Action

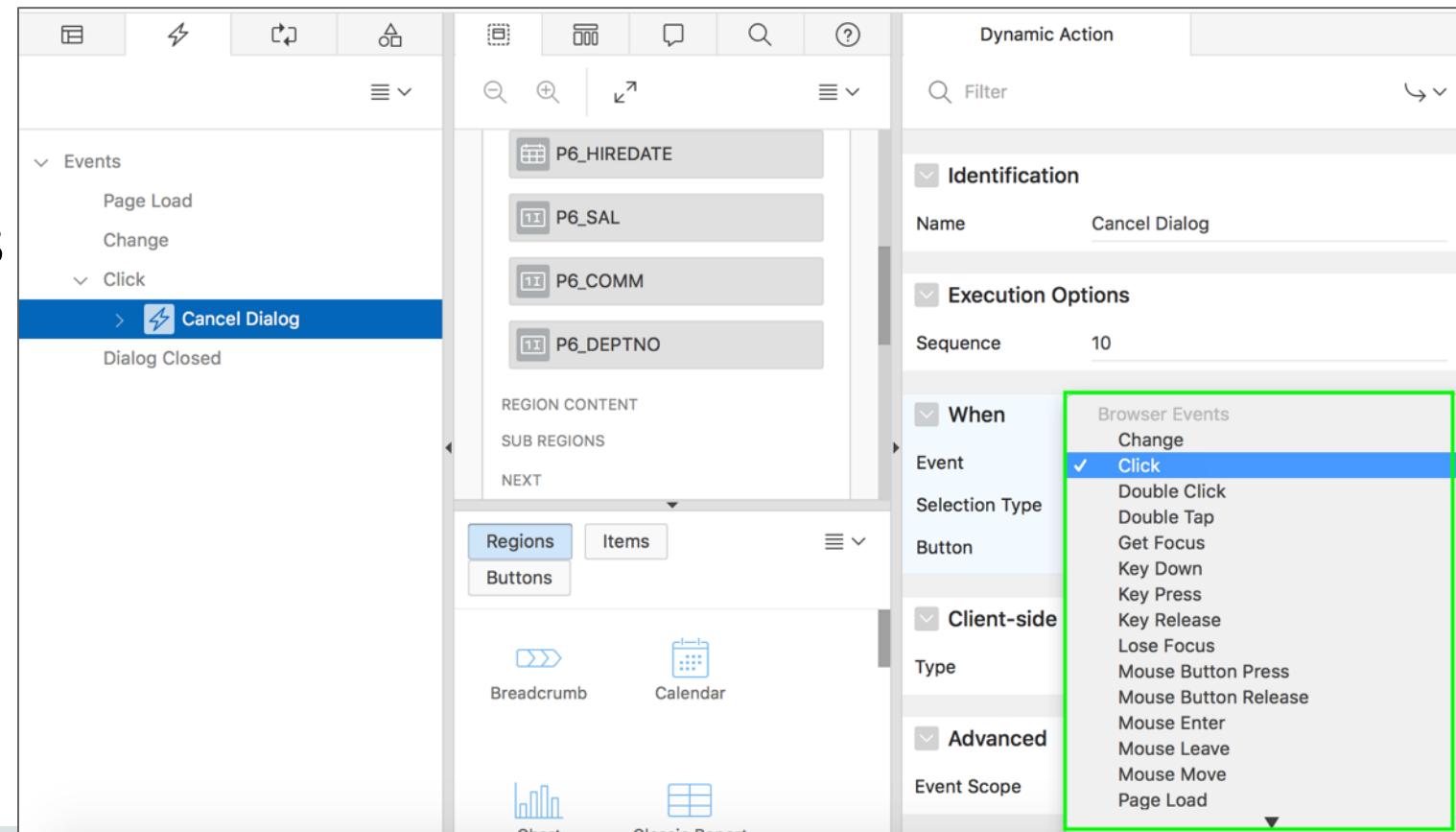
1. Create or edit an item, interactive grid column, button, region, JavaScript Expression, or jQuery selector on a page
2. Create a dynamic action from the application page that invokes the action
3. Run your application to test the dynamic action.

Creating a Dynamic Action: Steps

1. View the page to contain the item in Page Designer. Click the Dynamic Actions tab in the left pane.
2. Under Dynamic Actions, right-click Events and select Create Dynamic Action.
3. In the Property Editor, edit the following Dynamic Action attributes:
 - Identification > Name, Execution Options> Sequence
 - When > Event , When > Selection Type
 - Client-side Condition > Type, Optionally select the condition you want to control the true and false action processing of the dynamic action
4. Define the action that to be performed if the event evaluates to True or False
5. Click Save.

Understanding Dynamic Action Events

- You can define dynamic actions to fire based on events that happen on the page.
- Oracle Application Express includes four different categories of events:
 - Browser Events
 - Framework Events
 - Component Events
 - Custom Events



Enabling and Disabling Items: Steps

The screenshot illustrates the Oracle APEX Page Designer interface, specifically focusing on the creation and configuration of a dynamic action.

Step 1: The top navigation bar features several icons: a grid icon, a lightning bolt icon (highlighted with a green box), a refresh icon, and a dropdown arrow icon. The lightning bolt icon is circled in green with the number '1'.

Step 2: The main workspace shows the 'Events' section of the page designer. A 'Create Dynamic Action' button is highlighted with a green box and circled with the number '2'. The 'Events' section also includes options like 'Page Load', 'Change', 'Click', and 'Dialog'.

Dynamic Action Configuration:

- Identification:** Name: Commission for SALESMAN Only
- Execution Options:** Sequence: 10
- When:** Event: Change, Selection Type: Item(s), Item(s): P6_JOB

Page Structure:

- DIALOG HEADER:** Employee Details
- CONTENT BODY:** Employee Details
- ITEMS:** P6_ENAME, P6_JOB

Regions, Items, Buttons: These tabs are visible at the bottom of the page structure area.

Enabling and Disabling Items: Steps

The screenshot shows the Oracle APEX application builder interface with several panels and annotations:

- Client-side Condition Panel (Left):** Contains a condition setup:
 - Type: Item = Value
 - Item: P6_JOB
 - Value: SALESMANA green box highlights the "Value" field, and a red circle labeled **3** is positioned to its left.
- Action Panel (Right):** Shows an action configuration:
 - Action: Enable
 - Affected Elements: Selection Type: Item(s); Item(s): P6_HIREDATE, P6_COMM
 - Execution Options: Sequence: 10; Event: Commission for SALESMAN Only; Fire When Event Result Is: True; Fire on Initialization: YesA green box highlights the "Action" field, and a red circle labeled **4** is positioned above the "Affected Elements" section.
- Annotations:**
 - A black arrow points from the text "True action" to the "True" option under the "Commission for SALESMAN Only" event in the Action panel.

Enabling and Disabling Items: Steps

The screenshot shows the Oracle APEX application builder interface. On the left, a tree view displays various application components and their configurations. On the right, a detailed configuration panel is shown for a specific item.

Left Panel (Tree View):

- > Actions Fired on Page Load
- Events
 - Page Load
 - Change
 - Commission for SALESMAN Only
 - True
 - Enable
 - False (highlighted with a green box and circled with a green circle labeled 5)
 - Create FALSE Action
 - Expand All Below
 - Collapse All Below
 - Disable
 - Click
 - Cancel Dialog
 - Dialog Closed
 - Page Load
 - Dialog Closed

Right Panel (Configuration):

Action: Disable (highlighted with a green box and circled with a green circle labeled 6)

Affected Elements: Item(s) P6_HIREDATE,P6_COMM (highlighted with a green box)

Execution Options:

 - Sequence: 10
 - Event: Commission for SALESMAN Only
 - Fire When Event Result Is: False
 - Fire on Initialization: Yes

Annotations:

 - A green box highlights the "Create FALSE Action" step under the "False" condition.
 - A green box highlights the "Disable" action in the execution options panel.
 - A green box highlights the "Item(s)" field in the "Affected Elements" section.
 - A green box highlights the "P6_HIREDATE,P6_COMM" value in the "Item(s)" field.
 - A green box highlights the "Disable" button in the tree view under the "False" condition.
 - A green box highlights the "False" condition in the tree view under the "Commission for SALEMAN Only" node.
 - A green box highlights the "Commission for SALEMAN Only" node in the tree view.
 - A green box highlights the "Page Load" node in the tree view.
 - A green box highlights the "Actions Fired on Page Load" node in the tree view.
 - A green box highlights the "Events" node in the tree view.
 - A green box highlights the "Page Load" node in the tree view.
 - A green box highlights the "Change" node in the tree view.
 - A green box highlights the "Commission for SALEMAN Only" node in the tree view.
 - A green box highlights the "True" node in the tree view.
 - A green box highlights the "Enable" button in the tree view under the "True" condition.
 - A green box highlights the "False" condition in the tree view under the "True" condition.
 - A green box highlights the "Create FALSE Action" step in the tree view under the "False" condition.
 - A green box highlights the "Expand All Below" link in the tree view.
 - A green box highlights the "Collapse All Below" link in the tree view.
 - A green box highlights the "Disable" button in the tree view under the "False" condition.
 - A green box highlights the "Click" node in the tree view.
 - A green box highlights the "Cancel Dialog" button in the tree view under the "Click" node.
 - A green box highlights the "Dialog Closed" node in the tree view under the "Click" node.
 - A green box highlights the "Regions" tab in the bottom navigation bar.
 - A green box highlights the "Items" tab in the bottom navigation bar.
 - A green box highlights the "Buttons" tab in the bottom navigation bar.
 - A green box highlights the "Breadcrumb" icon in the bottom navigation bar.
 - A green box highlights the "Calendar" icon in the bottom navigation bar.

Setting the Value of Page Items: Set Value Example

Employee Details

Name	ADAMS
Job	CLERK
Manager	7788
Hire Date	12-JAN-83
Salary	1100
Commission	
Department	RESEARCH
Location	DALLAS
No Of Employees	5

Select a department

- Select Department -
- ACCOUNTING
- OPERATIONS
- RESEARCH
- SALES

Employee Details

Name	ADAMS
Job	CLERK
Manager	7788
Hire Date	12-JAN-83
Salary	1100
Commission	
Department	ACCOUNTING
Location	NEW YORK
No Of Employees	3

Location and Number of Employees information are retrieved and displayed for the selected department

Setting the Value of Page Items: Steps

The screenshot shows the Oracle APEX Dynamic Action configuration interface. A red box highlights the 'Events' section of the left sidebar, and a green circle labeled '1' is on the 'Events' button. Another green circle labeled '2' is on the 'Create Dynamic Action' button.

Dynamic Action Identification:

- Name:** Get Dept Information

Execution Options:

- Sequence:** 10

When:

- Event:** Change
- Selection Type:** Item(s)
- Item(s):** P11_DEPTNO

Content Body:

- Regions:** Employee Details
- Items:** P11_ENAME, P11_JOB

True Step:

- Action:** Show
- Item(s):** P11_ENAME, P11_JOB

False Step:

- Action:** None

Regions: Employee Details

Items: P11_ENAME, P11_JOB

Setting the Value of Page Items: Steps

The screenshot shows the Oracle APEX page builder interface with a red border around the main content area.

Left Panel (Actions Fired on Page Load):

- Events > Page Load:
 - Change > Get Dept Information (True):
 - Action: Set Value (highlighted with a blue background)
 - Condition: False
 - Click > Cancel Dialog
 - Dialog Closed

Middle Panel (Employee Details Dialog):

- DIALOG HEADER: Employee Details
- CONTENT BODY:
 - Employee Details
 - PREVIOUS
 - ITEMS:
 - P11_ENAME
 - P11_JOB

Right Panel (Action Definition):

- Step 3:** Action: Set Value (highlighted with a green box). Set Type: SQL Statement.

```
select d.loc location,
       count(e.empno) num_employees
  from dept d,
       emp e
 where d.deptno = e.deptno(+)
   and d.deptno = :P11_DEPTNO
  group by d.loc
```
- Step 4:** Action: Items to Submit: P11_DEPTNO. Escape Special Characters: Yes. Suppress Change Event: No.

Bottom Text: Only one native action used

Filtering and Refreshing a Report: Example

Job Department
ANALYST - Show All -

Employee # Name Job
7788 SCOTT ANALYST
7902 FORD ANALYST

Job Department
- Show All - ACCOUNTING

Employee # Name Job Manager Hire date Salary Commission Department
7839 KING PRESIDENT - 17-NOV-81 5000 - ACCOUNTING
7782 CLARK MANAGER KING 09-JUN-81 2450 - ACCOUNTING
- ACCOUNTING

1 - 3

Job Department
- Show All - - Show All -

Employee # Name Job Manager Hire date Salary Commission Department
7839 KING PRESIDENT - 17-NOV-81 5000 - ACCOUNTING
7698 BLAKE MANAGER KING 01-MAY-81 2850 - SALES
7782 CLARK MANAGER KING 09-JUN-81 2450 - ACCOUNTING
7566 JONES MANAGER KING 02-APR-81 2975 - RESEARCH
7788 SCOTT ANALYST JONES 09-DEC-82 3000 - RESEARCH

Data is refreshed upon selecting a job or a department

Select lists are cleared and data is refreshed upon clicking the Reset button

Filtering and Refreshing a Report: Steps

The diagram illustrates the configuration of a Dynamic Action for filtering and refreshing a report, divided into three main sections: Events, Dynamic Action, and Action.

Events: The Events section shows a tree structure of events. A red box highlights the "FILTER REFRESH" event under the "Change" category. A red arrow points from this event to the "Dynamic Action" configuration screen.

Dynamic Action: The Dynamic Action configuration screen shows the following details:

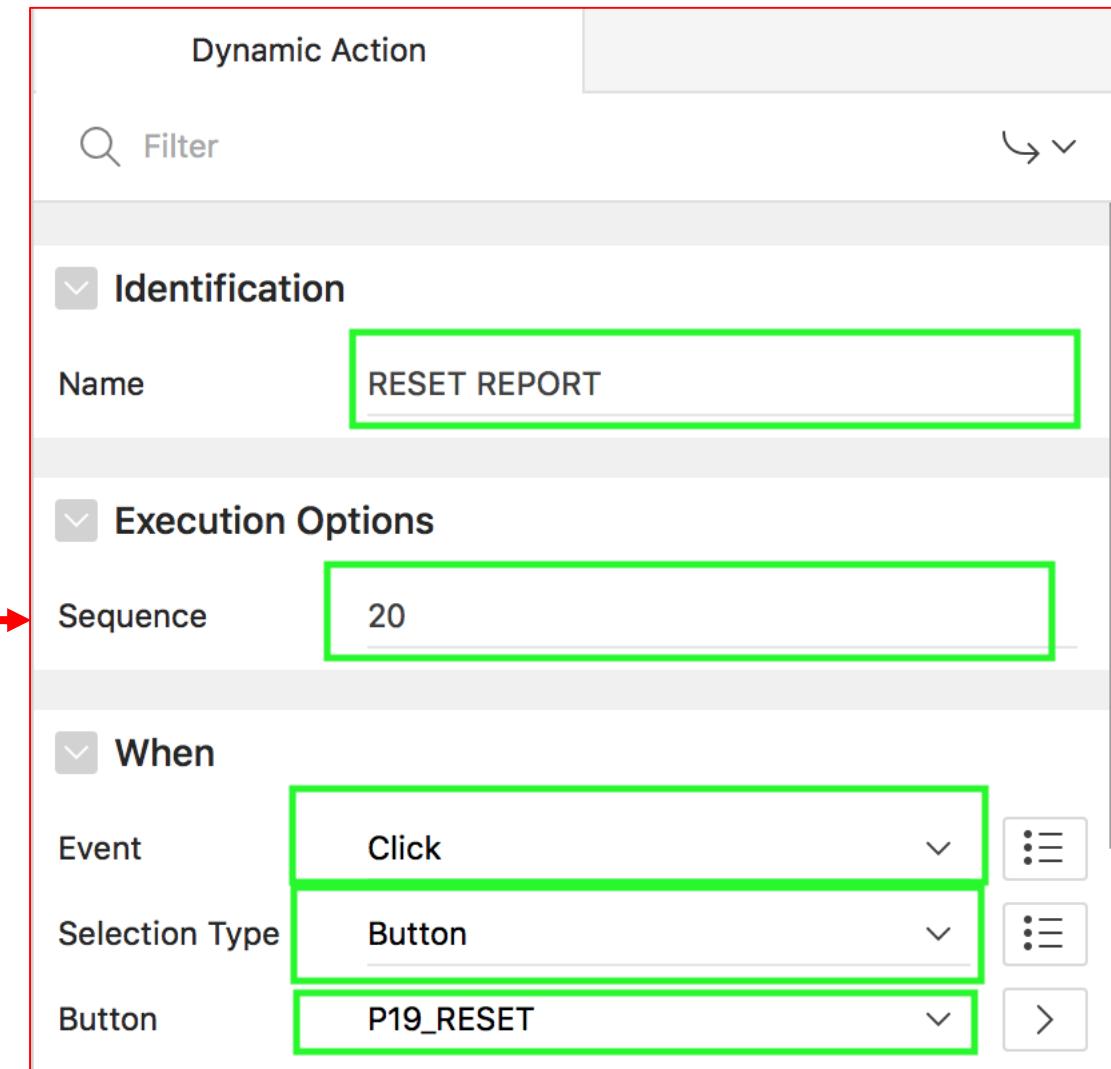
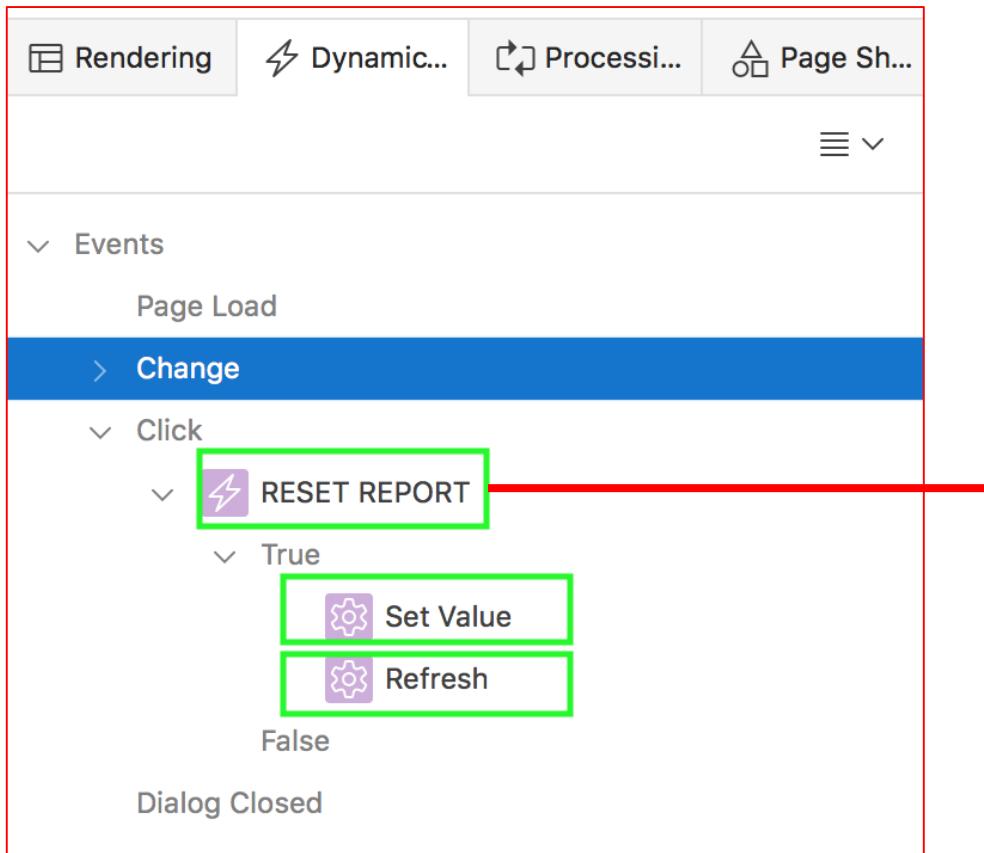
- Identification:** Name: FILTER REFRESH
- Execution Options:** Sequence: 10
- When:** Event: Change, Selection Type: Item(s), Item(s): P19_DEPTNO,P19_JOB

Action: The Action configuration screen shows the following details:

- Identification:** Action: Refresh
- Affected Elements:** Selection Type: Region, Region: Employees

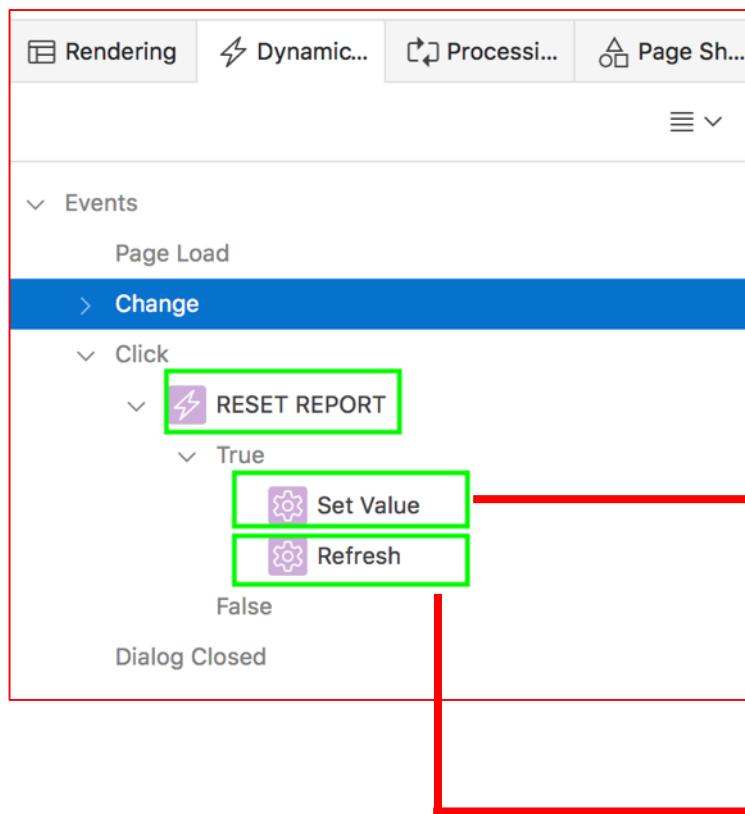
Filtering and Refreshing a Report: Steps

Dynamic Action: RESET REPORT



Filtering and Refreshing a Report: Steps

Actions: Set Value, and Refresh



This is a configuration dialog for a "Set Value" action. It includes sections for Identification (Action set to "Set Value"), Settings (Set Type set to "Static Assignment"), and Affected Elements (Selection Type set to "Item(s)" with values "P19_DEPTNO,P19_JOB"). There are also Execution Options (Sequence set to 10, Event set to "RESET REPORT").

This is a configuration dialog for a "Refresh" action. It includes sections for Identification (Action set to "Refresh"), Affected Elements (Selection Type set to "Region" with Region set to "Employees"), and Execution Options (Sequence set to 20, Event set to "RESET REPORT", Fire When Event Result Is set to "True", and Fire on Initialization set to "No").

What is a Plug-in?

- A plug-in is an extension to the built-in types of Application Express
- Use plug-ins for declarative use of new item, region, process and dynamic action types in your application

Employees Grid

		Search: All Text Columns		Go	Actions	Edit	Save	Add Row	Reset
		Name	Job	Manager	Hire Date	Salary	Commission	Rating	Department
<input checked="" type="checkbox"/>	<input type="checkbox"/>	KING	PRESIDENT	-	17-NOV-81	5000	-	-	10
<input type="checkbox"/>	<input type="checkbox"/>	BLAKE	MANAGER	7839	01-MAY-81	4000	-	-	30
<input type="checkbox"/>	<input type="checkbox"/>	CLARK	MANAGER	7839	09-JUN-81	2450	-		10
<input type="checkbox"/>	<input type="checkbox"/>	JONES	MANAGER	7839	02-APR-81	2975	-		20
<input type="checkbox"/>	<input type="checkbox"/>	SCOTT	ANALYST	7566	09-DEC-82	3000	-		20
<input type="checkbox"/>	<input type="checkbox"/>	FORD	ANALYST	7566	03-DEC-81	3000	-		20
<input type="checkbox"/>	<input type="checkbox"/>	SMITH	CLERK	7902	17-DEC-80	800	-	-	20
<input type="checkbox"/>	<input type="checkbox"/>	ALLEN	SALESMAN	7698	20-FEB-81	1600	300		30
<input type="checkbox"/>	<input type="checkbox"/>	WARD	SALESMAN	7698	22-FEB-81	1250	500	-	30

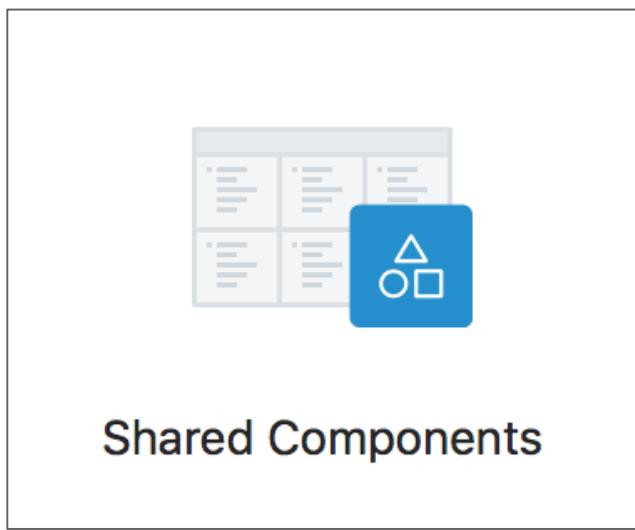
Example of implementing
the Star Rating Plug-In

Implementing a Plug-in in your Application

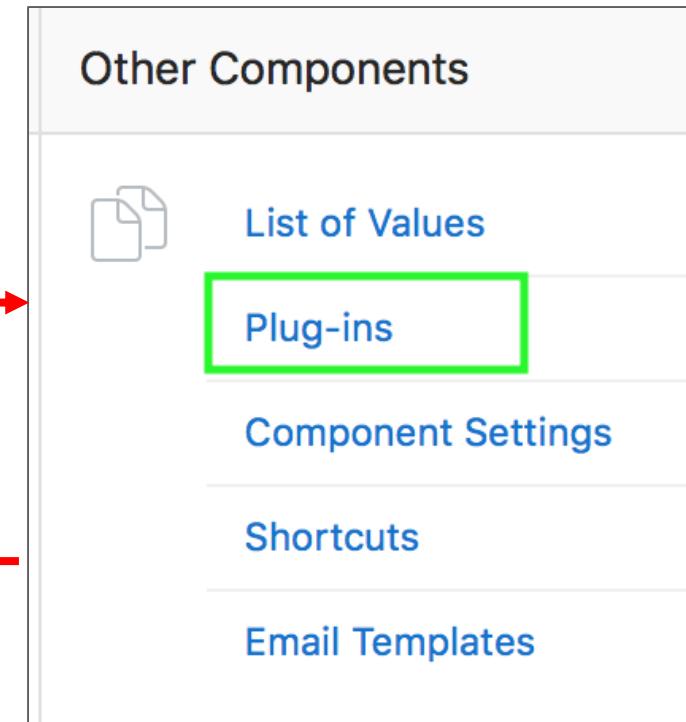
The process of implementing a plug-in involves:

1. Create a plug-in or import a plug-in into your application
2. Review, edit, or create an authorization scheme, item, region, process, or dynamic action type to use the plug-in
3. Run your application to test the plug-in

Viewing the Plug-in Repository



<https://apex.oracle.com/plugins>



Application 116491 \ Shared Components \ Plug-ins

Plug-ins Utilization History

Q | Go Actions ▾ Reset View Plug-in Repository Import > Create >

Importing and Installing a Plug-in

Application 116491 \ Shared Components \ Plug-ins

Plug-ins Utilization History

Search bar (highlighted by a red box)

Import > button (highlighted by a green box)

Create > button

Import

Select the file you wish to import to the export repository. Once imported, you can install your file.

If the imported file is a productivity or sample application export, the installation wizard will allow you to run the app installation scripts after installing the application definition.

* Import file: Choose File item_type_plug...ing_modern.sql

* File Type: Plug-in Database Application, Page or Component Export Websheet Application Export

File Character Set: Unicode UTF-8

Cancel Help

Import

File Import Confirmation

The export file has been imported successfully.

If you wish to install now, click the Next button. You can also install this file at a later time by navigating to the Export Repository.

> Tasks

< Cancel Next >

Next > (highlighted by a green box)

Importing and Installing a Plug-in

Install Plug-in

When you install a plug-in into the current application, the new plug-in will overwrite any existing plug-in having the same plug-in name. If the installation succeeds, the installation of the plug-in becomes permanent. If any errors are encountered, the actions are rolled back, resulting in no changes.

Export File Version: **2019.03.31**

Name: **Modern Star Rating**

Internal Name: **COM.ORACLE.APEX.STARRATING_MODERN**

Install Into Application: **116491 Employees App**

Action: **New plug-in will be created in application 116491.**

Cancel

Install Plug-in

Component Settings

Use Component Settings to set application level values for built-in Application Express components and installed plug-ins.

Name: **Modern Star Rating [Plug-in]**

Clear Tooltip: **Clear Rating**

Active Star Color: **red**

Inactive Star Color: **lightgray**

Default icon: **fa-star**

Cancel

Apply Changes

Viewing a Plug-in Definition

The screenshot shows the Oracle APEX interface for viewing plug-in definitions. On the left, a list of plug-ins is displayed, with one item selected: "Modern Star Rating". This item is highlighted with a green border and has a red arrow pointing from the list to its detailed configuration page on the right.

Plug-in: Modern Star Rating

Name

- * Name: Modern Star Rating
- * Internal Name: COM.ORACLE.APEX.STARRATING_MODERN
- * Type: Item

Subscription

Reference Master Plug-in From: [empty]

This is the "master" copy of this plug-in.

No plug-ins subscribe to this plug-in.

Source

PL/SQL Code

```
1 c_default_stars constant number      := 5;
2 c_default_icon_class constant varchar2(128) := 'fa-star';
3 c_default_acolor  constant varchar2(30)   := 'red';
4 c_default_bcolor  constant varchar2(30)   := '#e0e0e0';
5
6 -----
7 -- Renders the star rating item type based on the configuration of the page item.
8
```

Using the Plug-in in your Application

Select Modern Star Rating [Plug-in] for Identification > Type

The screenshot shows the Oracle APEX Page Designer interface. In the bottom left corner, there is a blue bar with a white icon and the text "RATING". To the right of this bar, a context menu is open, listing various item types under the "Type" section. The "Modern Star Rating [Plug-in]" option is highlighted with a green border and a checkmark.

Application 116491 \ Page Designer

Rendering Dynamic... Processing... Page Sh...

Content Body

Employees Grid

Columns

- APEX\$ROW_SELECTOR
- APEX\$ROW_ACTION
- ROWID
- EMPNO
- ENAME
- JOB
- MGR
- HIREDATE
- SAL
- COMM

RATING

Employees Grid

PAGE HEADER

PAGE NAVIGATION

BREADCRUMB BAR

BEFORE CONTENT BODY

CONTENT BODY

Employees Grid

COPY EDIT

PREVIOUS NEXT

ITEMS

Regions Items Buttons

Actions Menu

Checkbox

Color Picker

Date Picker

Display Only

Hidden

HTML Expression

Link

✓ Modern Star Rating [Plug-in]

Number Field

Password

Percent Graph

Popup LOV

Radio Group

Row Selector

Select List

Shuttle

Switch

Text Field

Text Field with autocomplete

Textarea

Show Legacy/Unsupported...

Column

Filter

Identification

Column Name

Type

Heading

Alignment

Alternative Label

Settings

Number of Stars

6

Using the Plug-in in your Application

Settings

Number of Stars

Use Component Defaults Yes

Icon fa-star 

Active Star Color  red  

Inactive Star Color  lightgray  

Tooltip Template Rating: #VALUE#

Pick Icon

Font APEX Utilized

Style Default

Category - Select -

Search: thumbs  

 fa-thumbs-do...	 fa-thumbs-o...	 fa-thumbs-o-up	 fa-thumbs-up	 fa-thumbs-do...
 fa-thumbs-o...	 fa-thumbs-o-up	 fa-thumbs-up		

Summary

In this lesson, you learned how to :

- Create and use Dynamic Actions in your application
- Import and install Plug-ins in your application
- Implement Plug-ins in your application





Hands-on Lab



Oracle **APEX**



Unit 15: Migrating Application Development Between Environments

Lesson Objectives

After completing this lesson, you should be able to:

- Describe the steps involved in migrating an application between environments
- Export the application definition, underlying database objects, and data from your development environment
- Import the application definition, underlying database objects, and data in the target development environment

Migrating your Application Development Between Environments: Overview



Migrating your Application Development Between Environments: Steps

1. Exporting the application from the current environment, and importing it into the new environment
2. Recreating all of the database objects, and then populating the tables with the data from the current environment
3. Testing the application in the new environment

Exporting the Application

The screenshot shows the Oracle APEX App Builder interface. At the top, there is a navigation bar with tabs: ORACLE APEX, App Builder (selected), SQL Workshop, Team Development, and App Gallery. A green circle with the number 1 is positioned at the top right of the navigation bar. Below the navigation bar, the application title is "Application 116491 - Employees App". On the right side of the title, there is a "Edit Application Properties" button. In the center, a red box highlights the "Export / Import" tab under the "Application 116491 \ Export / Import" heading. Inside this box, the question "What task would you like to perform?" is displayed above two options: "Import" and "Export". The "Export" option is highlighted with a blue border. To the right of this central box, there are three other tabs: "Objects", "Shared Components", and "Utilities". A green box highlights the "Export / Import" tab. At the bottom left, there are "Cancel" and "Help" buttons. At the bottom right, there is a large blue "Next >" button with a green circle containing the number 2 above it.

Exporting the Application

The screenshot illustrates the process of exporting an application in Oracle APEX. It shows the main export interface and a detailed view of the export preferences.

Main Export Interface:

- Header:** Export, Workspace, Applications (selected), Websheets, Themes, Plug-ins, User Interface Defaults, Feedback.
- Section:** Export Application.
- Buttons:** Reset, Export (highlighted with a green box).

Export Application Details:

- Application:** 116491 Employees App.
- Selected Application:** Employees App.
- Page Count:** 32.
- Owner:** LOWCODE.
- File Format:** UNIX.
- Owner Override:** (empty field).
- Build Status Override:** Run and Build Application.
- Debugging:** Yes.
- As of:** (empty field) minutes ago (~ 5 min delay).
- File Character Set:** Unicode UTF-8.

Export Preferences (Detailed View):

- Export Supporting Object Definitions:** Yes.
- Export Public Reports:** Yes.
- Export Private Reports:** No.
- Export Report Subscriptions:** Yes.
- Export Developer Comments:** Yes.
- Export Translations:** Yes.
- Export with Original IDs:** No.

File Name Input:

- Name:** f116491.sql (highlighted with a purple box).
- Item Type:** item_type_plugin_com_oracle_apex_starrating_mod.

Annotations:

- A red box surrounds the "Export" button and the "Export Preferences" section.
- A green circle with the number 3 points to the "File Character Set" field.
- A green circle with the number 4 points to the "Name" input field.

Importing the Application into the Target Environment

The screenshot illustrates the Oracle APEX Import process across three main steps:

- Create:** Shows the navigation bar with ORACLE APEX, App Builder, SQL Workshop, and Team Development. A green circle labeled "1" is positioned above the "Create" button, which has a plus sign icon.
- Import:** Shows the "Import" step. A green box highlights the "Import" button, which has an upward arrow icon. A red box surrounds the entire import interface. A green circle labeled "2" is positioned to the right of the "Choose File" input field, which contains the file name "f116491.sql".

Import

Select the file you wish to import to the export repository. Once imported, you can install your file. If the imported file is a productivity or sample application export, the installation wizard will allow you to run the app installation scripts after installing the application definition.

* Import file f116491.sql

* File Type: Database Application, Page or Component Export
 Websheet Application Export
 Plug-in
 Theme Export
 User Interface Defaults
 Team Development Feedback
 CSS Export [Deprecated]
 Image Export [Deprecated]
 File Export [Deprecated]

File Character Set

Next >
- File Import Confirmation:** Shows the confirmation step. A green circle labeled "3" is positioned to the left of the "Next >" button. This step indicates that the export file has been imported successfully and provides options to install now or later.

Importing the Application into the Target Environment

Install Database Application



When you install an application having the same ID as an existing application in the current workspace, the existing application is deleted and then replaced by the new application. If you attempt to install an application having the same ID as an existing application in a different workspace, a benign error message displays. If you are importing a Application Express application, the installation wizard will allow you to install supporting objects.

Current Workspace: CHAITANYA [?](#)

Export File Workspace ID: 5523230421166791507 [?](#)

Export File Application ID: 116491 [?](#)

Export File Version: 2019.03.31 [?](#)

Export File Parsing Schema: LOWCODE [?](#)

Application Origin: This application was exported from another workspace. [?](#)

* Parsing Schema: APEXDEV_SCHEMA [?](#)

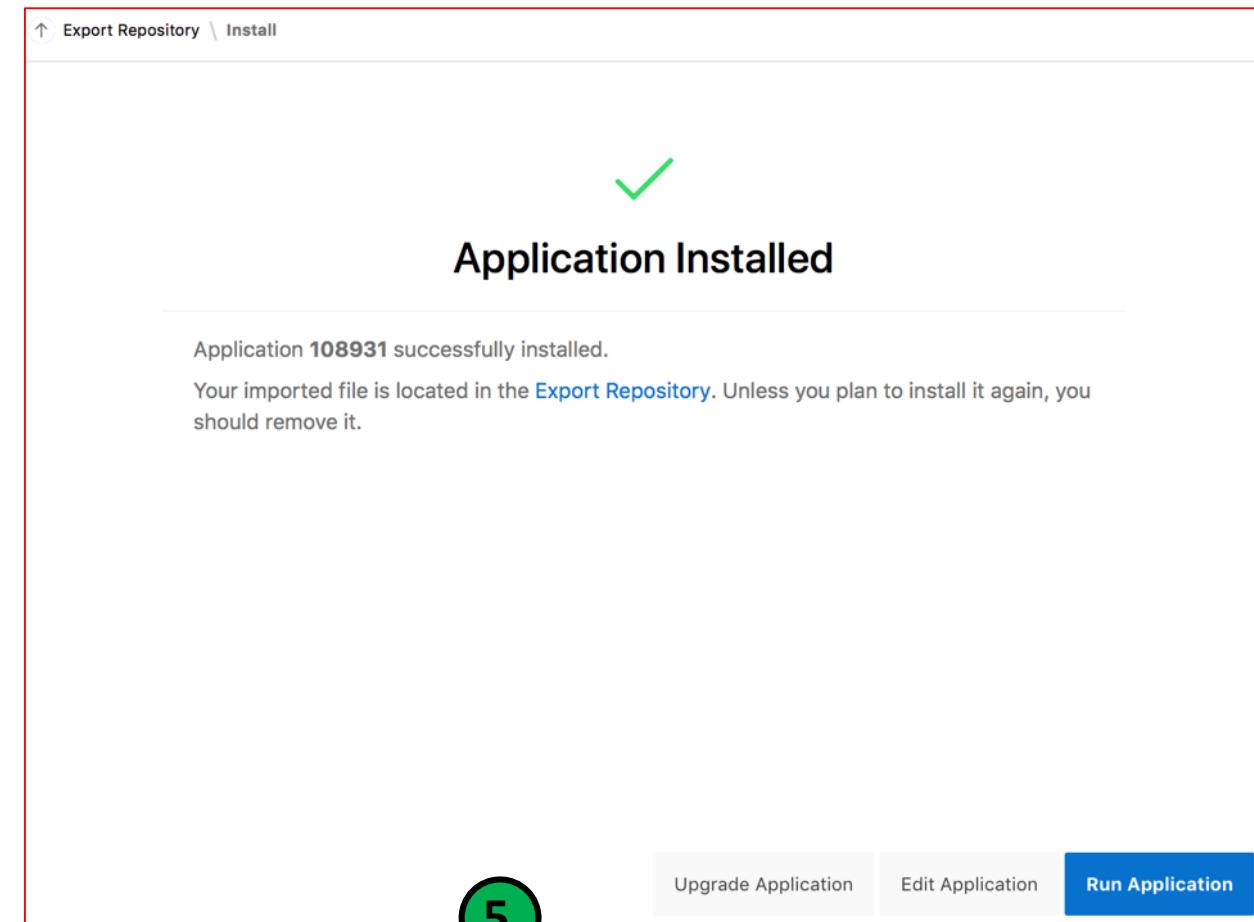
* Build Status: Run and Build Application [?](#)

* Install As Application: Auto Assign New Application ID
 Reuse Application ID 116491 From Export File
 Change Application ID

> Tasks

Cancel Install Application

4



Migrating the Database Objects and Data

The screenshot shows the Oracle APEX interface with the following navigation bar:

- ORACLE® APEX
- App Builder
- SQL Workshop
- Team Development
- App Gallery

The SQL Workshop tab is currently selected, indicated by a green border around its icon and the word "Workshop". Below the navigation bar, there are four main icons:

- Object Browser (grid icon)
- SQL Commands (script icon)
- SQL Scripts (script icon)
- Utilities (gear icon)

The Utilities icon is highlighted with a green box and has a red arrow pointing from it to a detailed view on the right side of the screen.

Utilities

- Data Workshop**
Load and unload data using CSV, XLSX, XML or JSON files.
- Quick SQL**
Generate SQL using shorthand syntax
- Generate DDL**
Generate scripts for all or selected database objects within a schema.

Migrating the Database Objects: Tables

Utilities

Utilities Q ▼ Go Actions ▼

Data Workshop Load and unload data from JSON files.

Schema Tables Views Packages Procedures Functions Triggers Synonyms Types Sequences Indexes Database Links Materialized Views

LOWCODE 57 2 5 7 1 38 0 9 13 92 0 0

1 2 Create Script >

Quick SQL Generate SQL using shorthand syntax

Generate DDL Generate scripts for all or selected database objects within a schema.

Generate DDL - Schema

Schema

Select the database schema which owns the database objects for which you would like to generate a data definition language (DDL) script.

* Schema LOWCODE ?

Cancel Next >

The screenshot shows the Oracle Database Utilities interface. The main area displays statistics for the 'LOWCODE' schema, including 57 tables, 2 views, 5 packages, 7 procedures, 1 function, 38 triggers, 0 synonyms, 9 types, 13 sequences, 92 indexes, 0 database links, and 0 materialized views. A 'Data Workshop' section allows loading and unloading data from JSON files. Below it is a 'Quick SQL' option for generating shorthand SQL. On the left, a 'Generate DDL' button is highlighted with a green box and a circled '1'. In the center, a modal dialog titled 'Generate DDL - Schema' is open, asking to select a schema for generating DDL. The 'LOWCODE' schema is selected in a dropdown menu. The dialog has a 'Next >' button at the bottom right. A circled '2' is over the 'Create Script >' button in the top right of the main interface, and a circled '3' is over the 'Next >' button in the dialog.

Migrating the Database Objects: Tables

Generate DDL - Object Type

Object Type

Select the object types for which you would like to generate DDL. Clicking **Generate DDL** generates DDL for the selected object types. To select object names for selected types, click **Next**.

Output: Display Inline Save As Script File [?](#)

Check All [?](#)

Object Type:

<input type="checkbox"/> Function	<input type="checkbox"/> Index	<input type="checkbox"/> Package
<input type="checkbox"/> Procedure	<input type="checkbox"/> Sequence	<input type="checkbox"/> Synonym
<input checked="" type="checkbox"/> Table	<input type="checkbox"/> Trigger	<input type="checkbox"/> View
<input type="checkbox"/> Database Link	<input type="checkbox"/> Type	<input type="checkbox"/> Materialized View

File Character Set **Unicode UTF-8** [?](#)

4

Generate DDL [Next >](#)

Generate DDL - Object Name

Object Name

<input checked="" type="checkbox"/> DEPT	DR\$EBA_DEMO_IG_TEXT_FTX\$I
<input type="checkbox"/>	DR\$EBA_DEMO_IG_TEXT_FTX\$K
<input type="checkbox"/>	DR\$EBA_DEMO_IG_TEXT_FTX\$N
<input type="checkbox"/>	DR\$EBA_DEMO_IG_TEXT_FTX\$R
<input type="checkbox"/>	DR\$EBA_DEMO_IG_TEXT_FTX\$U

EBA DEMO CAL MSESSIONS

5

Generate DDL

Migrating the Database Objects: Tables

The screenshot illustrates the process of generating and downloading DDL scripts for database tables.

Generate DDL (Step 6): A dialog box titled "Generate DDL" shows three green checkmarks indicating successful validation. It prompts the user to enter a script name and description. The "Script Name" field is set to "Employee Details". The "Create Script" button is highlighted with a green circle and labeled "6".

SQL Scripts (Step 7): A list of existing scripts is shown. The "Employee Details" script is listed with details: Owner: LOWCODE, Name: Employee Details, Created: 25 seconds ago, Updated By: -, Updated: 25 seconds ago, Bytes: 697, Results: 0. The "Edit" icon for this script is highlighted with a green circle and labeled "7".

Script Editor (Step 8): The generated DDL script is displayed in the "Script Editor". The "Download" button is highlighted with a green circle and labeled "8". The script content is as follows:

```
1 CREATE TABLE "DEPT"
2   (
3     "DEPTNO" NUMBER(2,0) NOT NULL ENABLE,
4     "DNAME" VARCHAR2(50),
5     "LOC" VARCHAR2(50),
6     CONSTRAINT "DEPT_PK" PRIMARY KEY ("DEPTNO")
7   )
8 /
9 CREATE TABLE "EMP"
10  (
11    "EMPNO" NUMBER(4,0) NOT NULL ENABLE,
12    "ENAME" VARCHAR2(50),
13    "JOB" VARCHAR2(50),
14    "MGR" NUMBER(4,0),
15    "HIREDATE" DATE,
16    "SAL" NUMBER(7,2),
```

Migrating the Database Objects: Triggers

Generate DDL - Object Type

Object Type

Select the object types for which you would like to generate DDL. Clicking **Generate DDL** generates DDL for the selected object types. To select object names for selected object types, click **Next**.

Output: Display Inline Save As Script File [?](#)

Check All [?](#)

Object Type:

<input type="checkbox"/> Function	<input type="checkbox"/> Index	<input type="checkbox"/> Package
<input type="checkbox"/> Procedure	<input type="checkbox"/> Sequence	<input type="checkbox"/> Synonym
<input type="checkbox"/> Table	<input checked="" type="checkbox"/> Trigger	<input type="checkbox"/> View
<input type="checkbox"/> Database Link	<input type="checkbox"/> Type	<input type="checkbox"/> Materialized View

File Character Set **Unicode UTF-8** [?](#)

[<>](#) [Cancel](#) [Generate DDL](#) **Next >**

Migrating the Data

DEPT

+ ▾

Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL REST

Query Count Rows Insert Row

Data

EDIT	DEPTNO	DNAME	LOC
	10	ACCOUNTING	NEW YORK
	20	RESEARCH	DALLAS
	30	SALES	CHICAGO
	40	OPERATIONS	BOSTON

[Download](#)



Recreating the Database Objects in the Target Environment

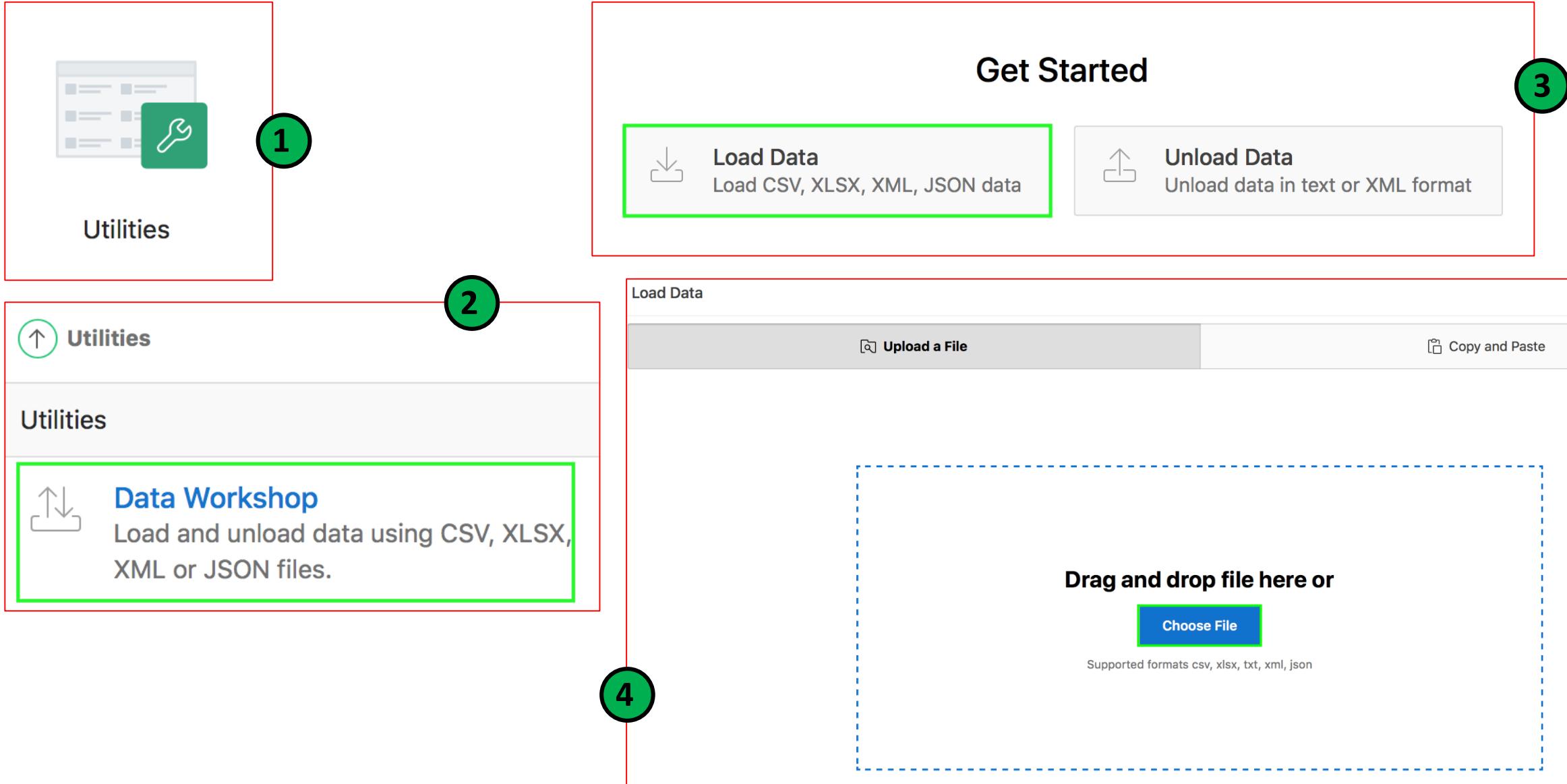
The screenshot illustrates the steps to upload SQL scripts for recreating database objects in a target environment. The interface consists of three main panels:

- SQL Scripts Panel (Top Left):** Shows a list of scripts. A green circle labeled **1** highlights the green 'Upload' icon next to the 'SQL Scripts' label.
- SQL Scripts List View (Top Right):** Shows a grid of uploaded scripts. A green circle labeled **2** highlights the 'Upload >' button in the top toolbar. Other buttons visible include 'Go', 'Actions', 'Delete Checked', 'Quick SQL', and 'Create >'. The 'Upload >' button is highlighted with a green border.
- Upload Script Dialog (Bottom Left):** A modal dialog titled 'Upload Script'. It contains fields for 'File' (with 'Employee Details.sql' selected), 'Script Name' (empty), and 'File Character Set' (set to 'Unicode UTF-8'). A green circle labeled **3** highlights the 'Choose File' button. The 'Choose File' button and the file name 'Employee Details.sql' are highlighted with a green border.
- Uploaded Scripts Grid (Bottom Right):** Shows a list of scripts uploaded earlier. A green circle labeled **4** highlights the 'Run' column. The first script in the list is highlighted with a green border.

SQL Scripts List View Data:

	Edit	Owner	Name	Created	Updated By	Updated	Bytes	Results	Run
<input type="checkbox"/>		CHAITANYA	Employee Details Triggers.sql	2 seconds ago	CHAITANYA	2 seconds ago	520	0	
<input type="checkbox"/>		CHAITANYA	Employee Details View.sql	21 seconds ago	CHAITANYA	20 seconds ago	410	0	
<input type="checkbox"/>		CHAITANYA	Employee Details.sql	56 seconds ago	CHAITANYA	56 seconds ago	724	0	

Migrating Data to the Target Environment1



Migrating Data to the Target Environment

Load Data

Load To: New Table **Existing Table**

* Table Owner: APEXDEV_SCHEMA

* Table: DEPT

All 3 columns have been automatically mapped to the DEPT table.

* Error Table Name: DEPT_ERR\$

Update Method: Append Replace

Configure 

Load Data

✓

Data in table DEPT appended with 4 new rows!

File Encoding: Unicode UTF-8

Shows the first 3 columns and 5 rows. To view the full preview and configure data

Preview 

Load Data 

Cancel 

View Table  Create Application > 

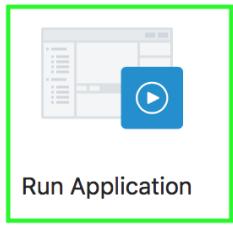
5

6

Testing the Migrated Application

Application 120072 - Employee Details

Edit Application Properties



Run Application



Supporting Objects



Shared Components



Utilities



Export / Import

Employee Details App

chaitanya ▾

Home

Dashboard

Departments

Employees

Dashboard

Employees per Department



Employees per Job



Total Salary per Department

12,000

Total Salary per Job

10,000

Home Application 120072 Edit Page 2 Session View Debug Debug Page Info Quick Edit Theme Roller

Summary

In this lesson, you learned how to:

- Export an application from your development environment and import it into the target APEX environment
- Migrate database objects and data between environments





Hands-on Lab



Oracle **APEX**