Hands-on Lab


Oracle APEX

Unit 10: Adding Computations, Processes, and Validations

Lesson Objectives

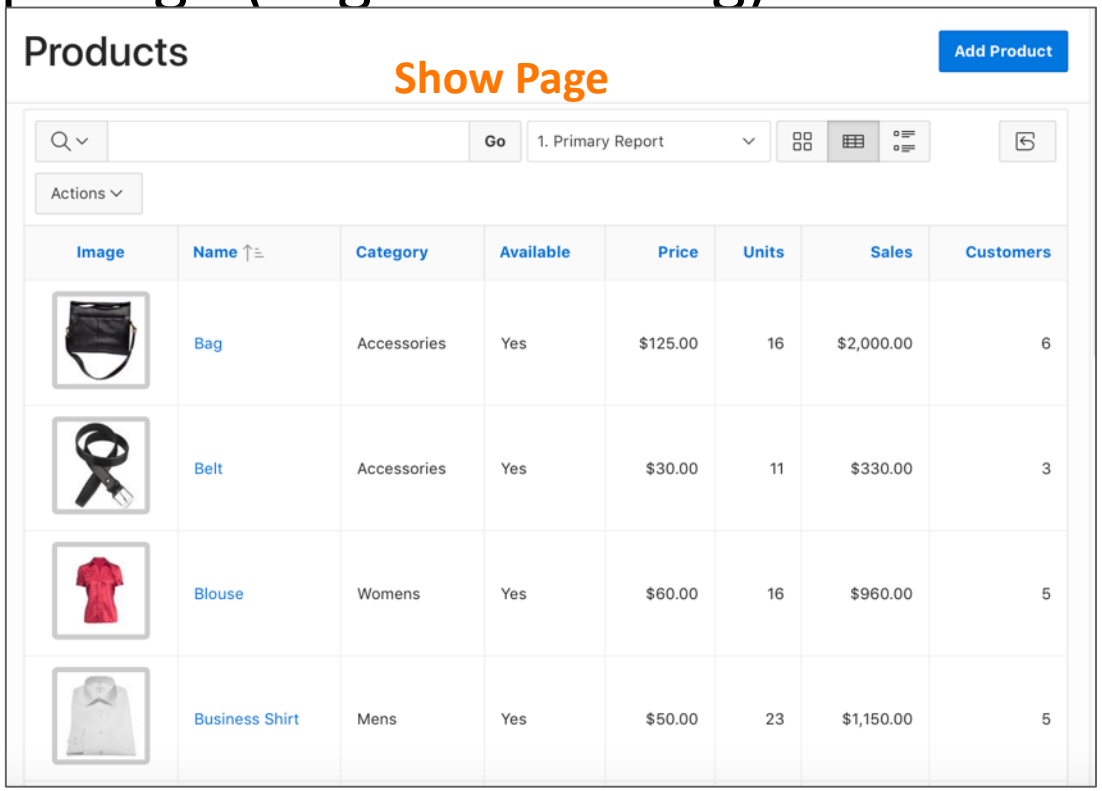
After completing this lesson, you should be able to:

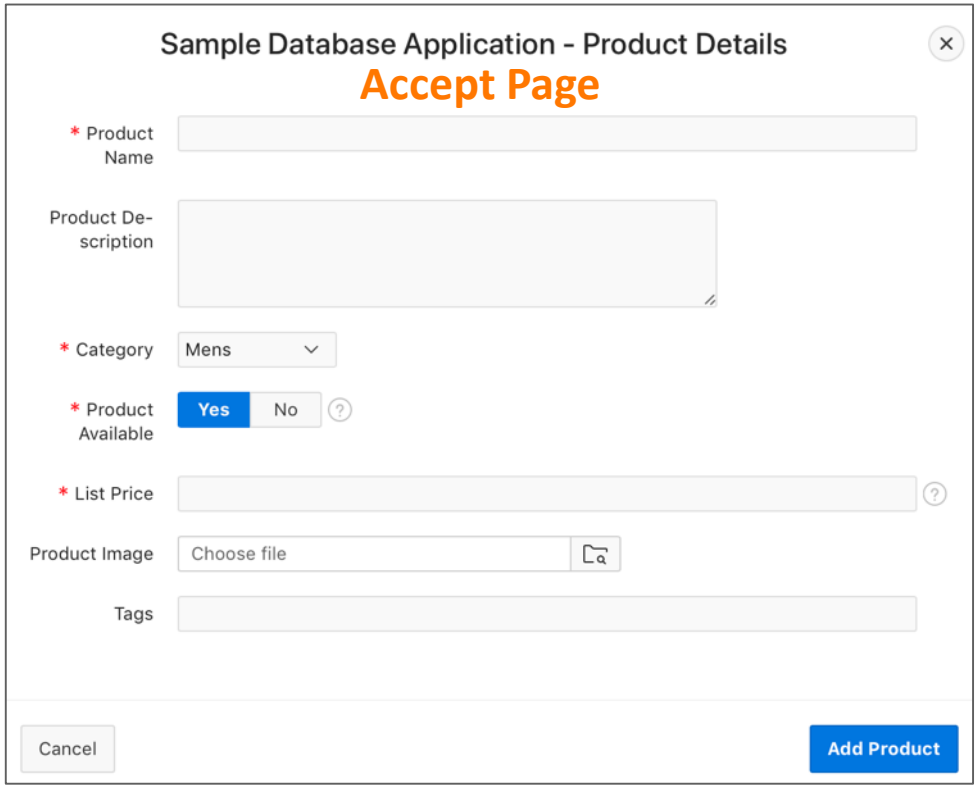
- Explain page rendering and page processing
- Create computations
- Create page processes
- Create validations
- Create branches

Understanding Page Rendering and Page Processing

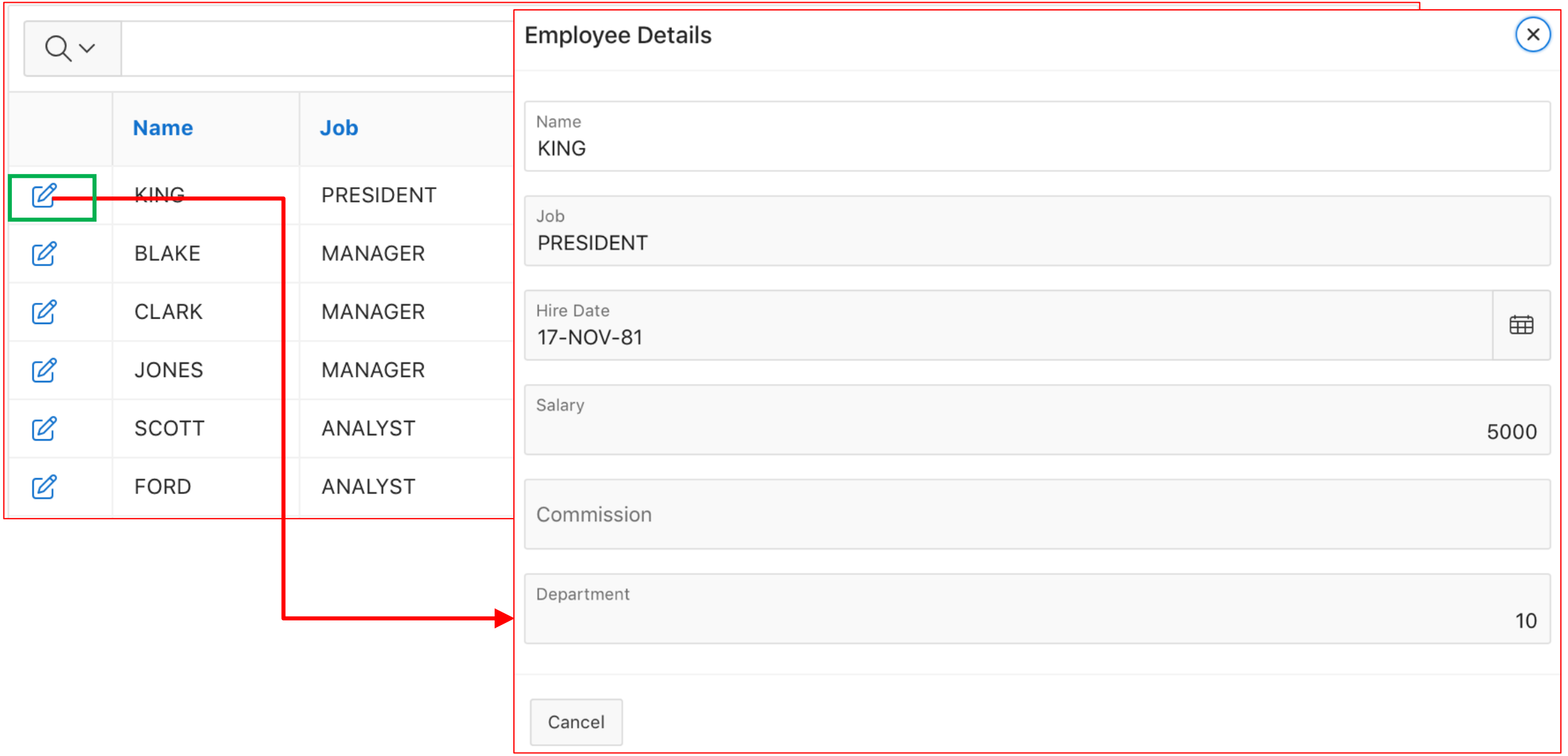
When you run an application, Application Express engine relies on two processes:

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

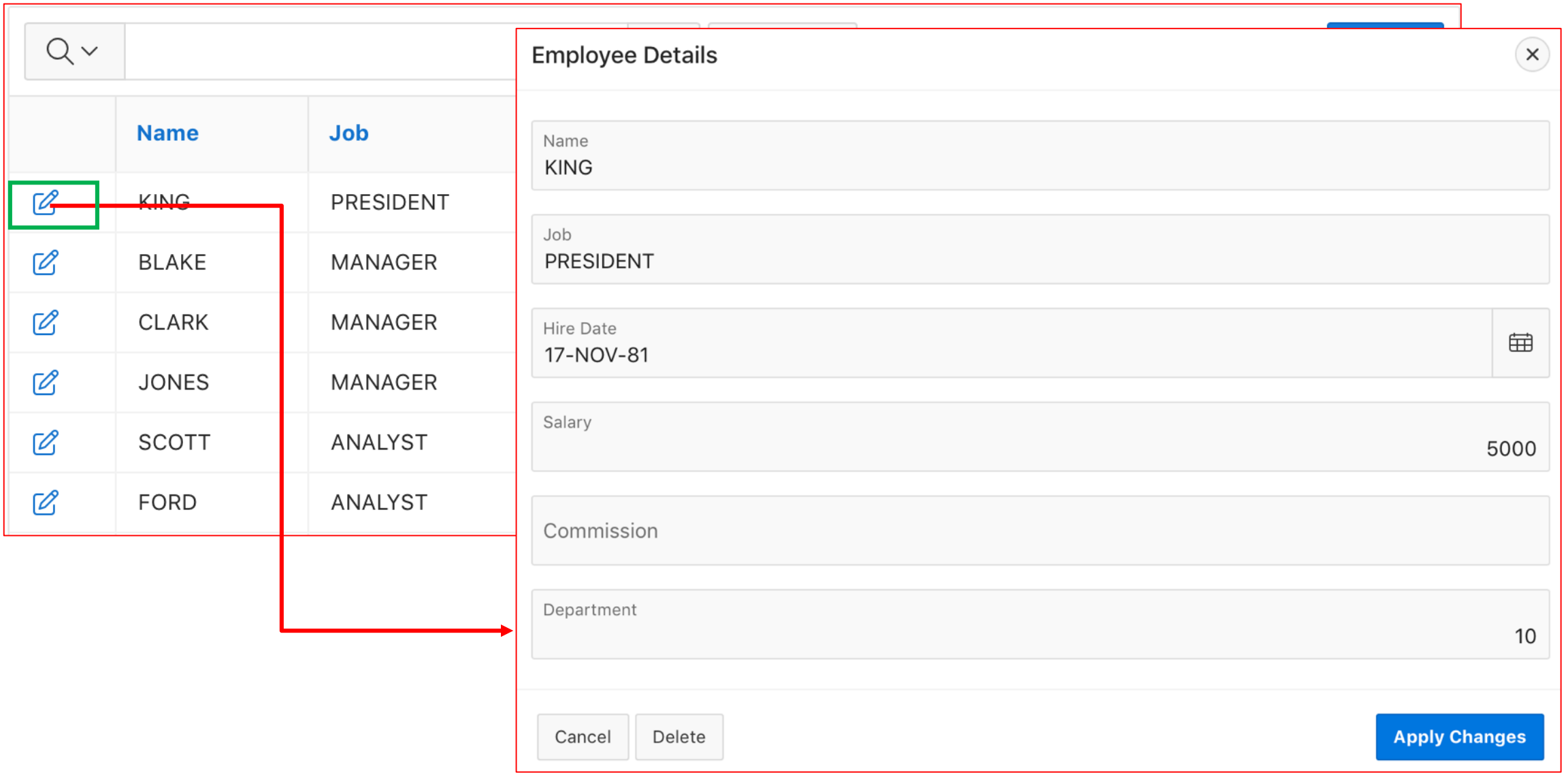




Page Rendering: Example



Page Processing: Example



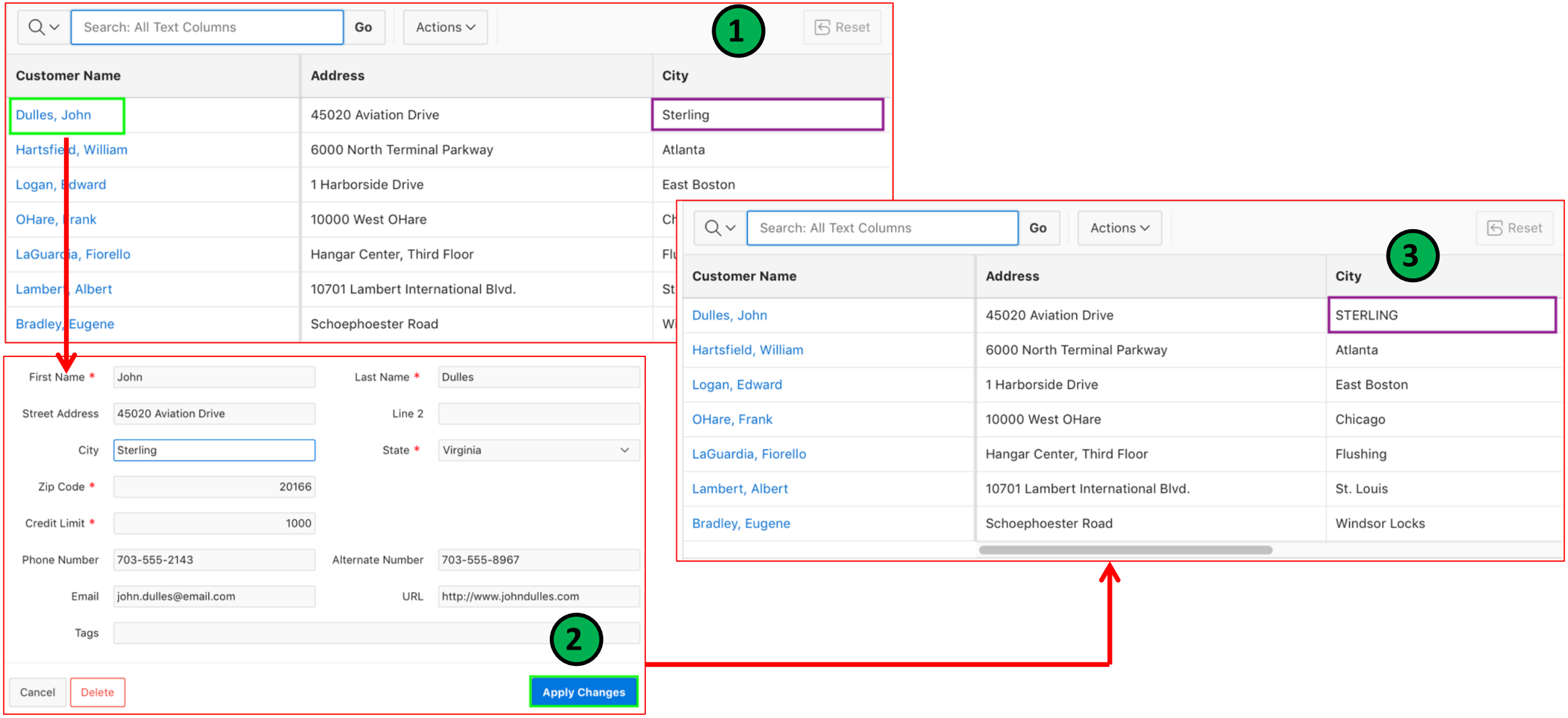
What is a Computation?

- 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).

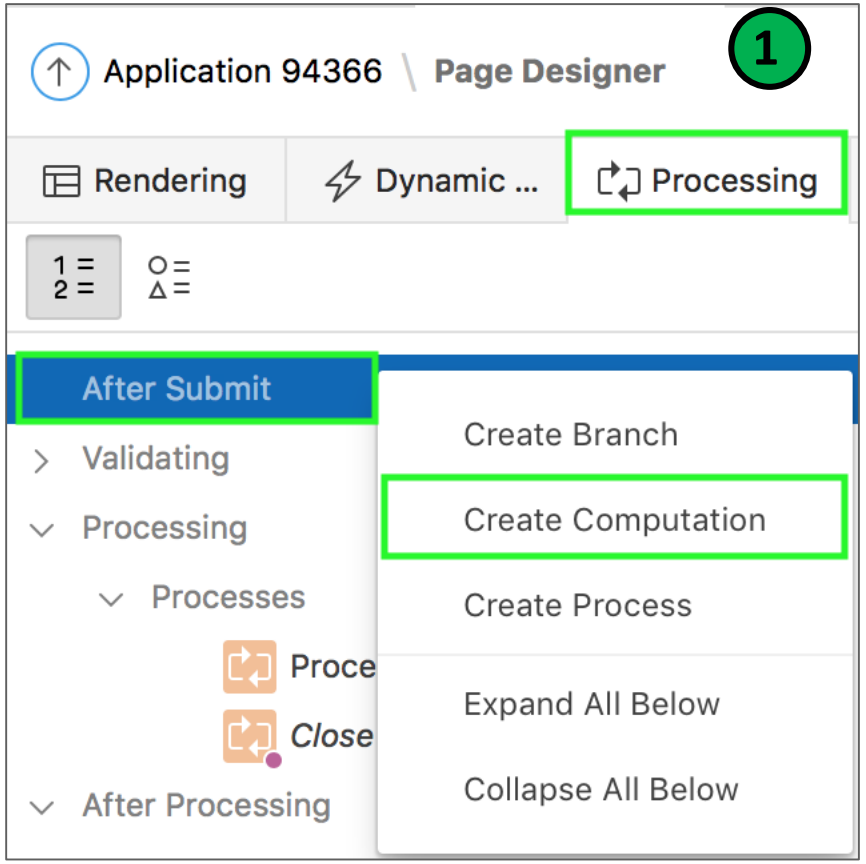
Creating a Page Computation: Steps

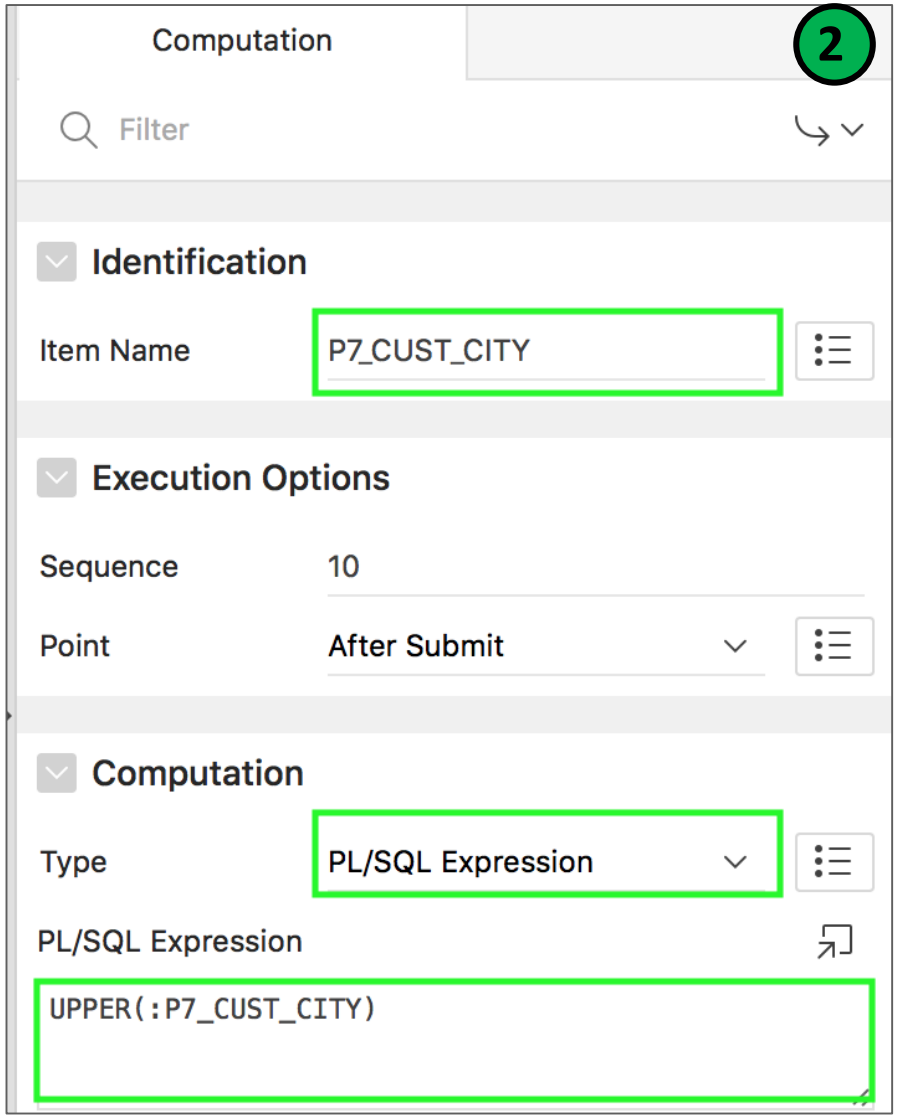
1. Navigate to the appropriate page in Page Designer.
2. In the Rendering tab, expand the Pre-Rendering or Post-Rendering node and locate where the computation should be.
3. Right-click the location and select Create Computation..
4. Edit the attributes in the property editor
5. Click Save

Creating a Page Computation: Example



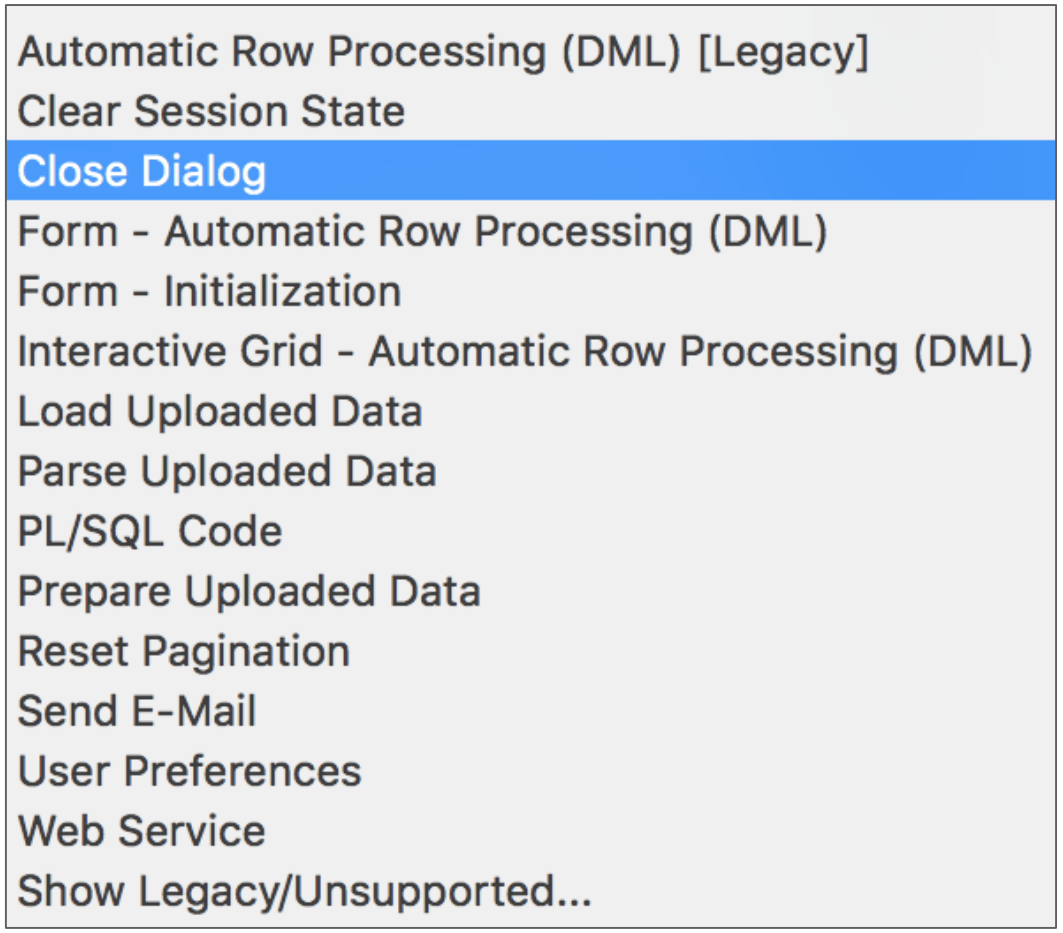
Creating a Page Computation: Example





What is a Page Process?

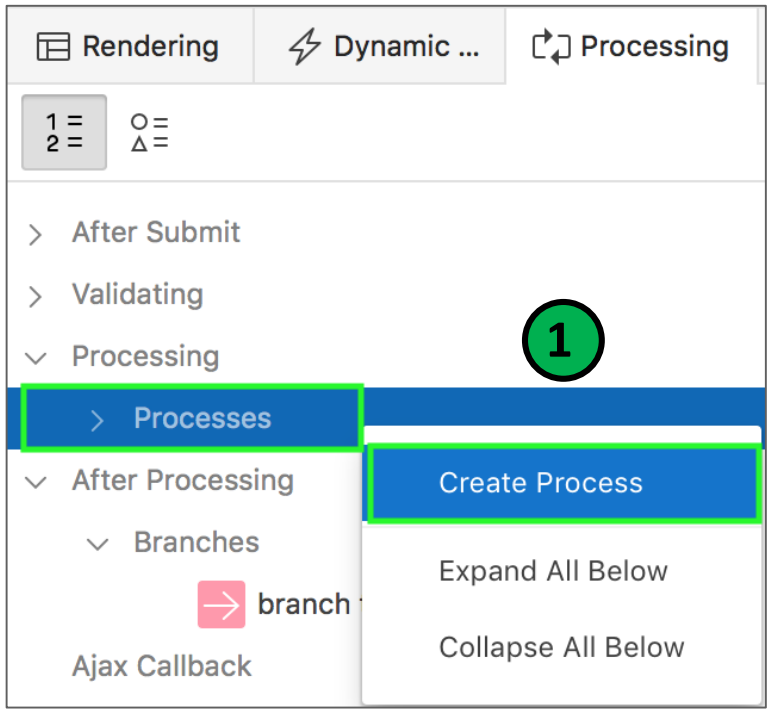
A page process is a specific event that runs when a page is loaded or submitted

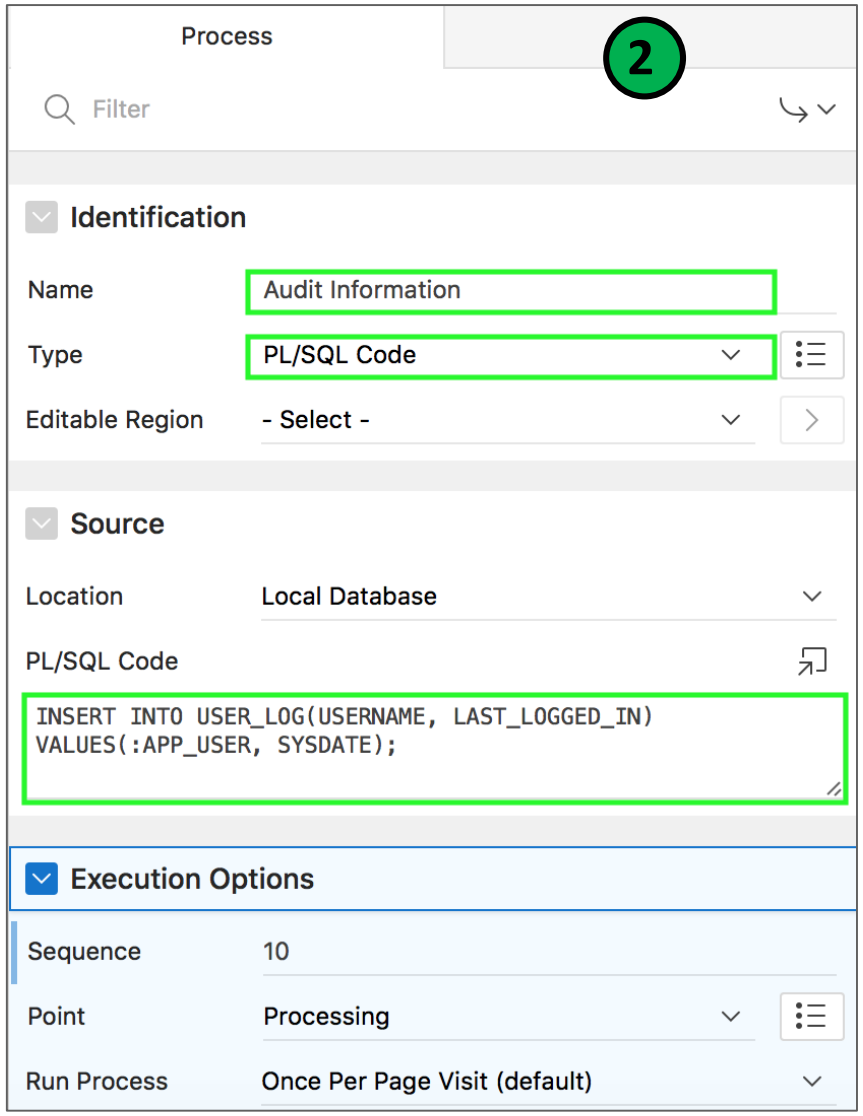


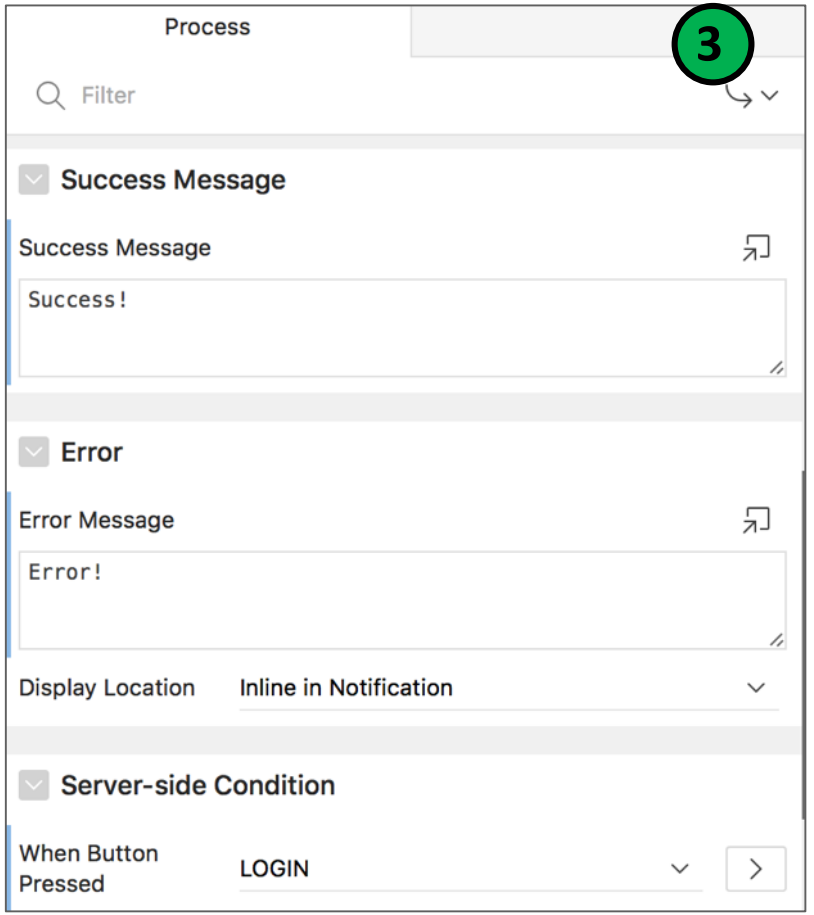
Creating a Page Process: Steps

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

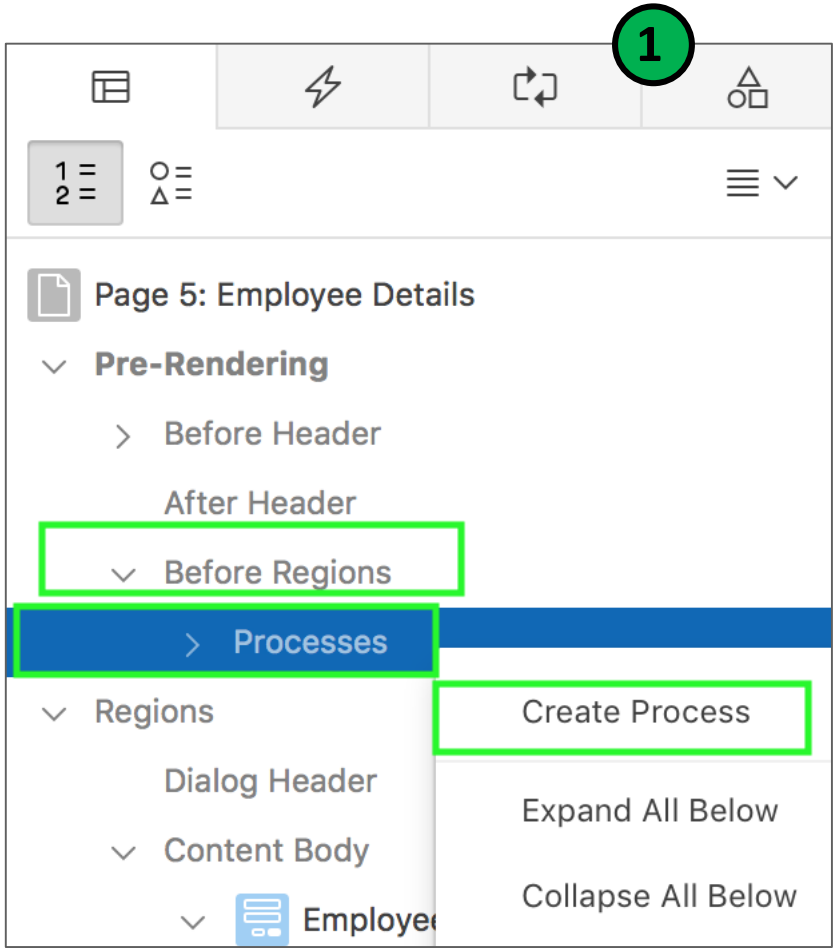
Creating an On Submit Process

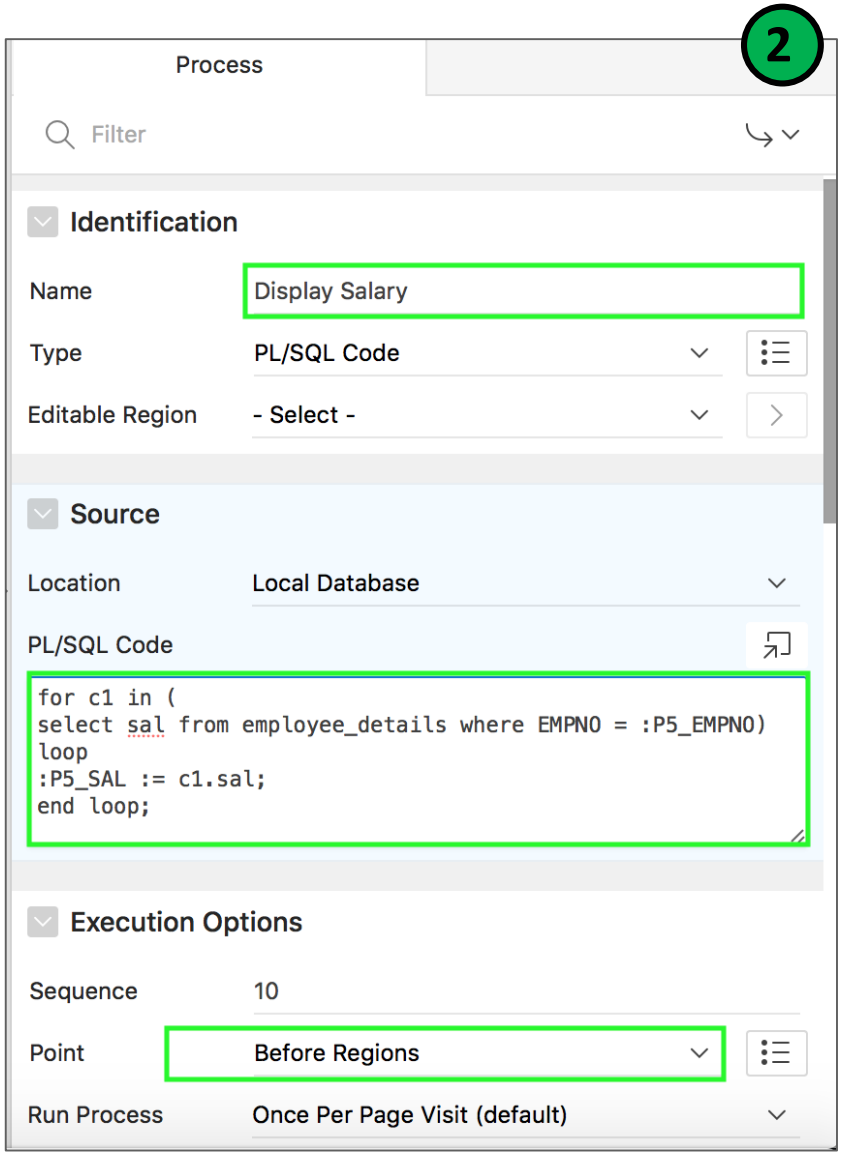




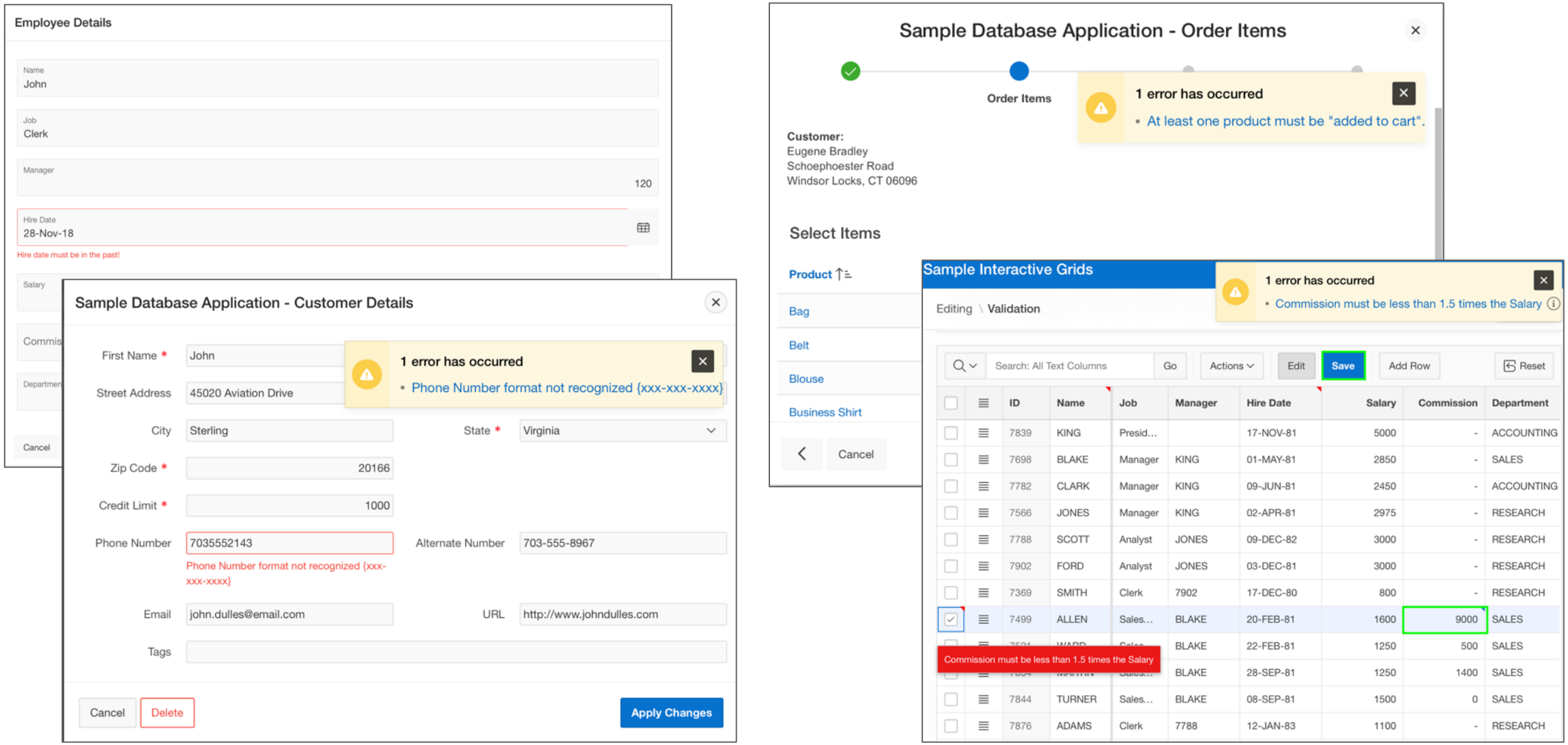


Creating an On Load Process





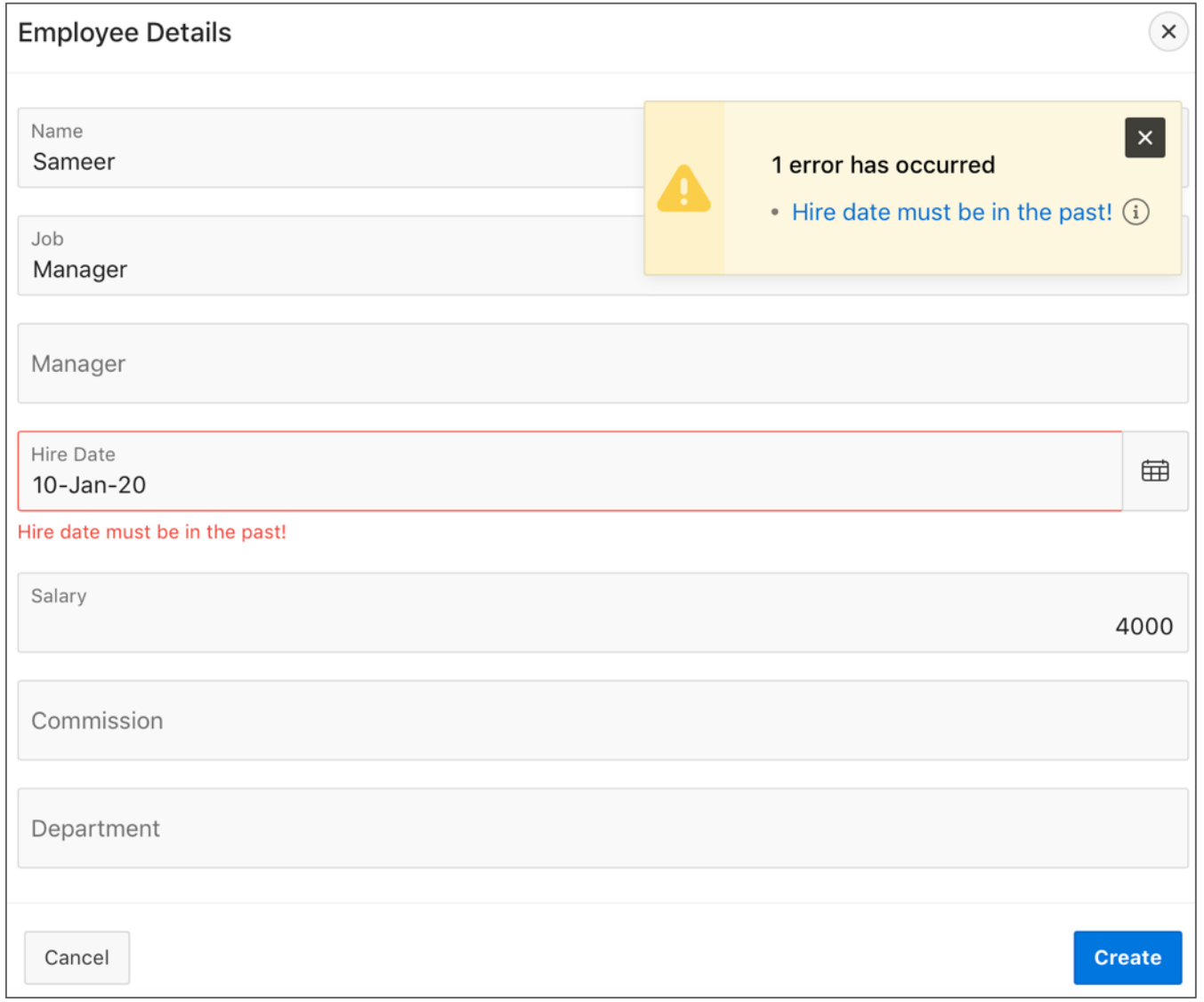
Understanding Validations



Creating a Validation for a Page Item

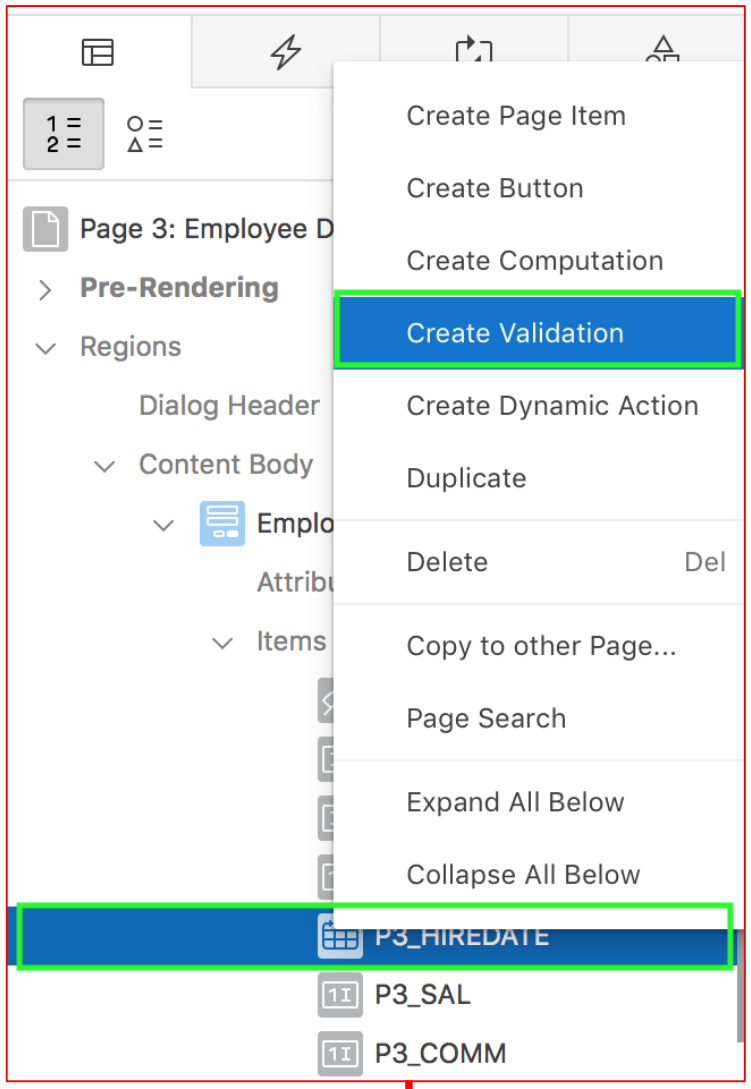
1. View the page in page designer. Click the **Processing** tab in the left pane.
2. Right-click **Validating** and select **Create Validation**.
3. Under Validation, for Type, select the type of quality to be tested for this validation. Depending on your selection, one or more additional attributes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

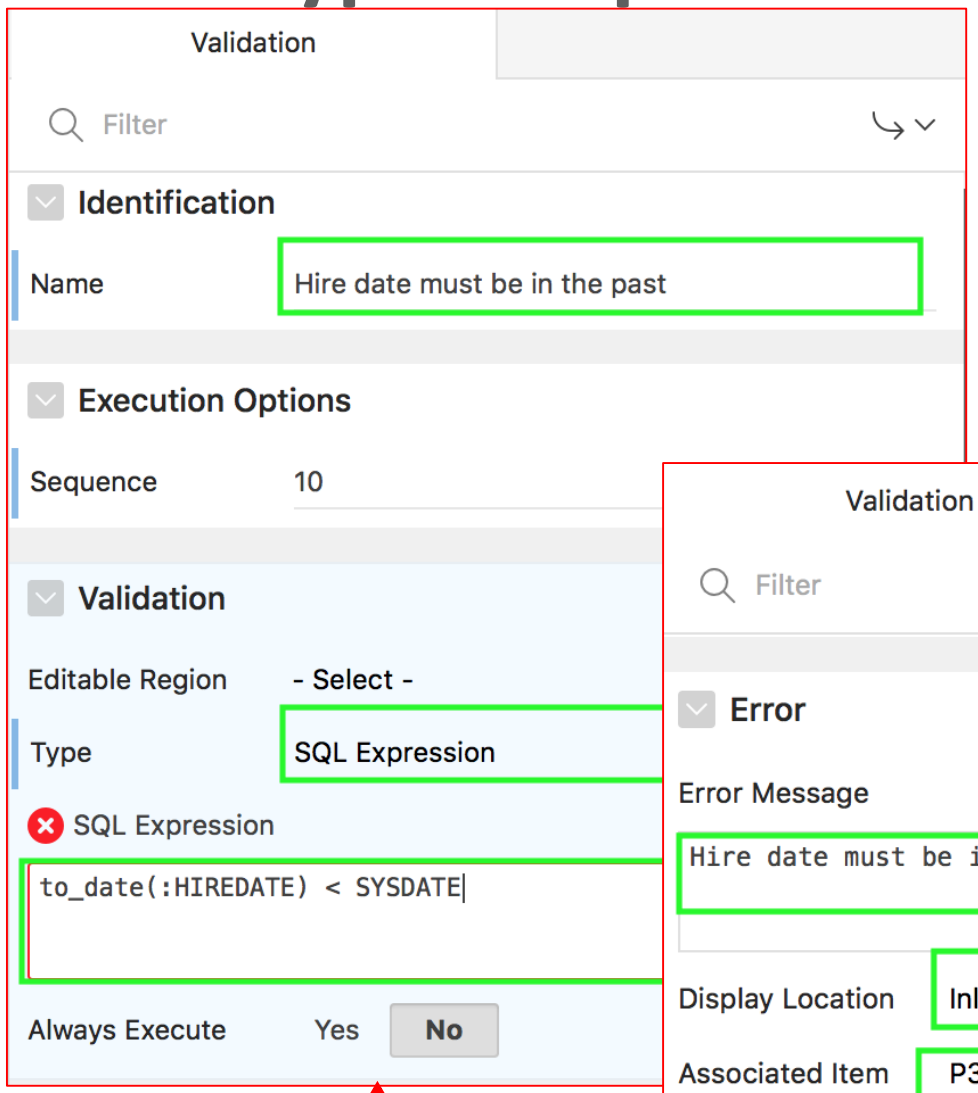
Creating a SQL Validation Type - Example

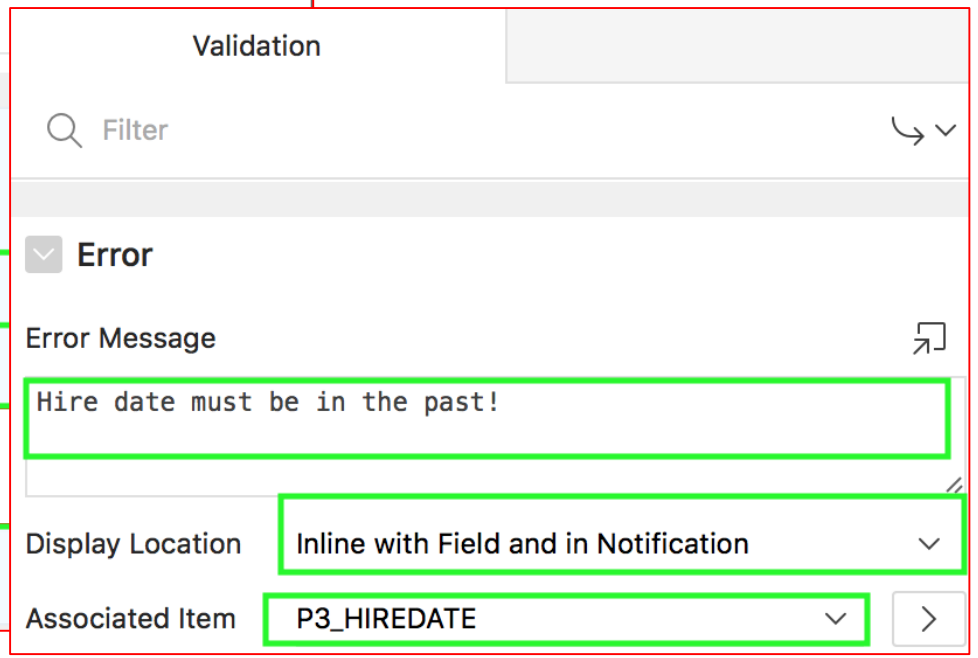


Create a Validation to ensure Hire Date is less than the current date.

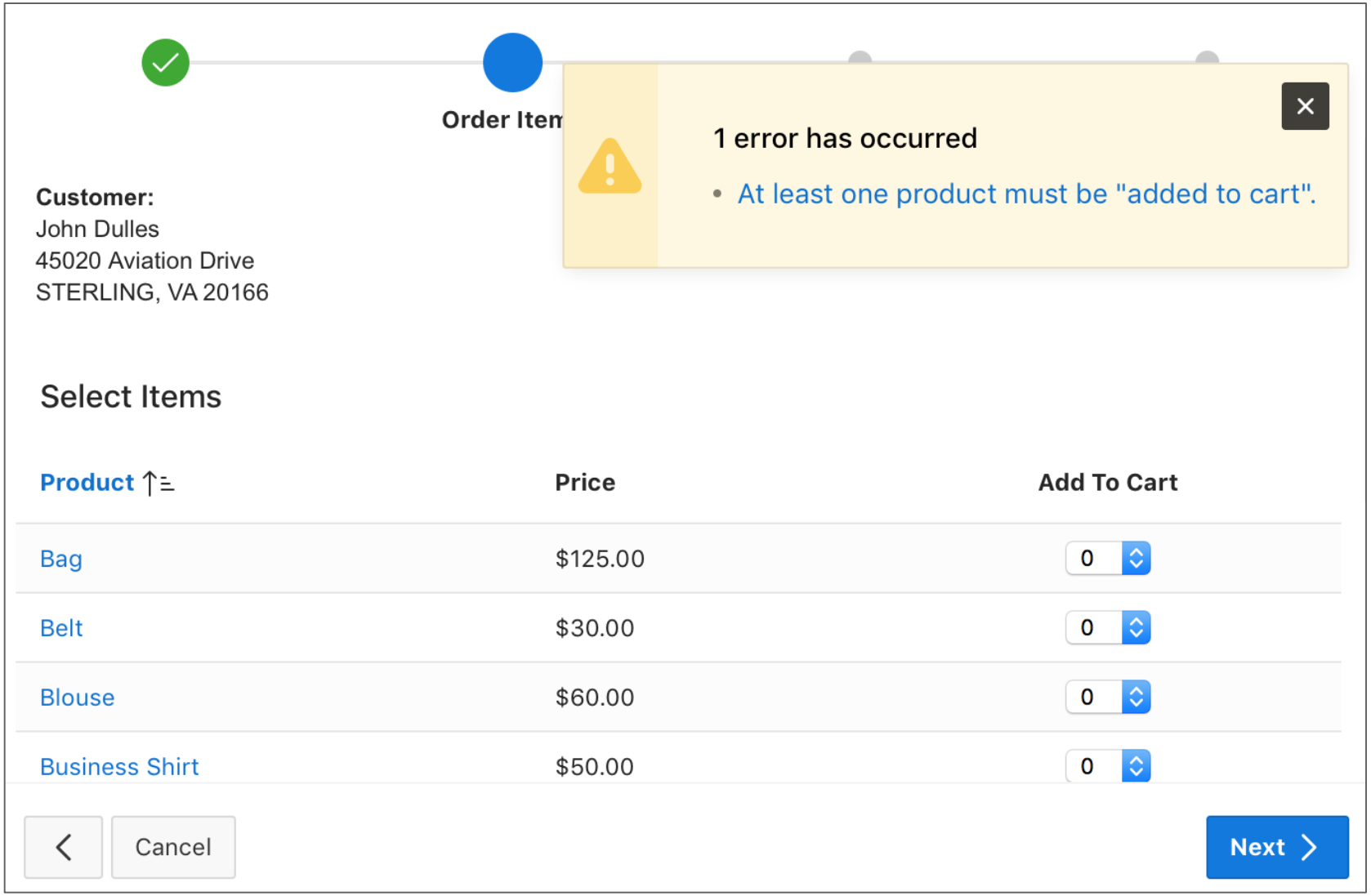
Creating a SQL Validation Type: Steps





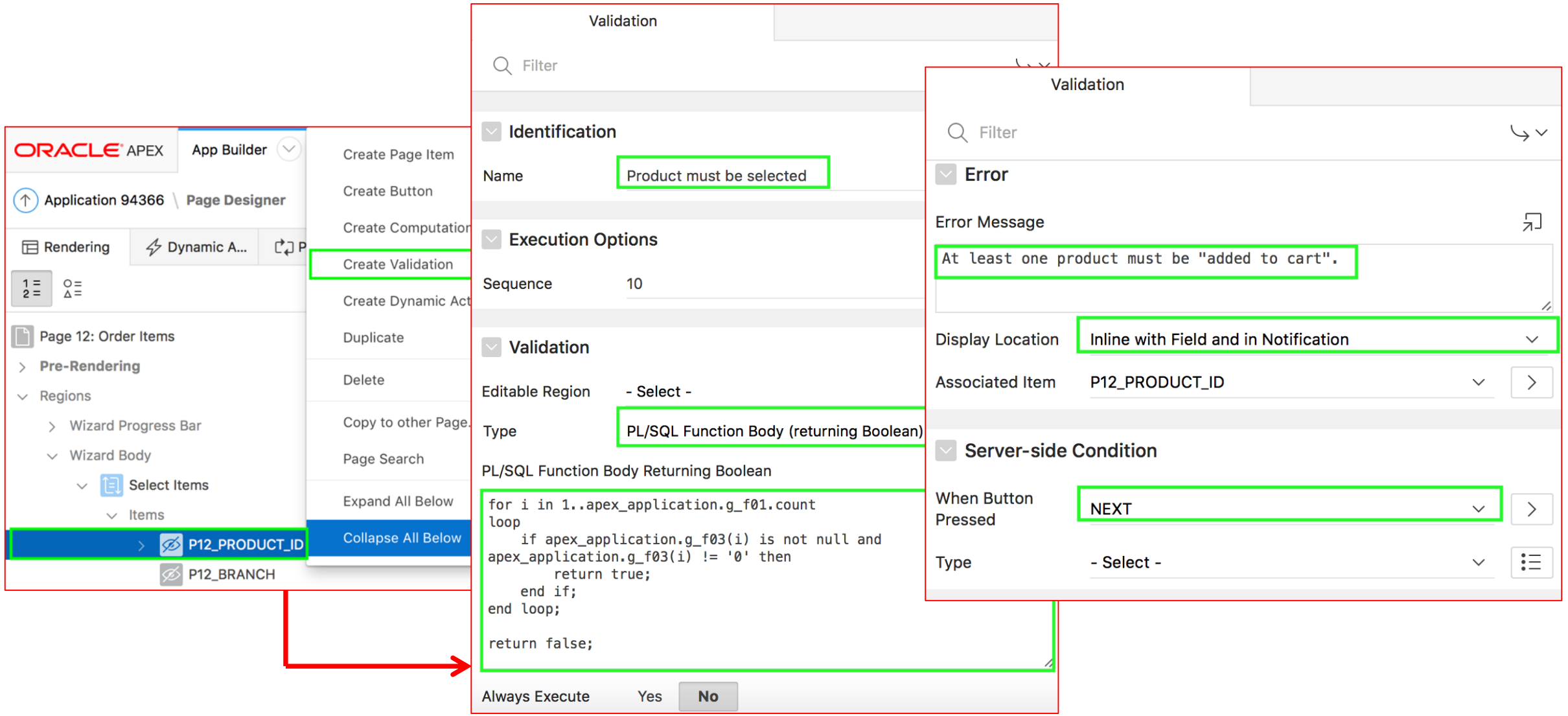


Creating a PL/SQL Validation Type - Example

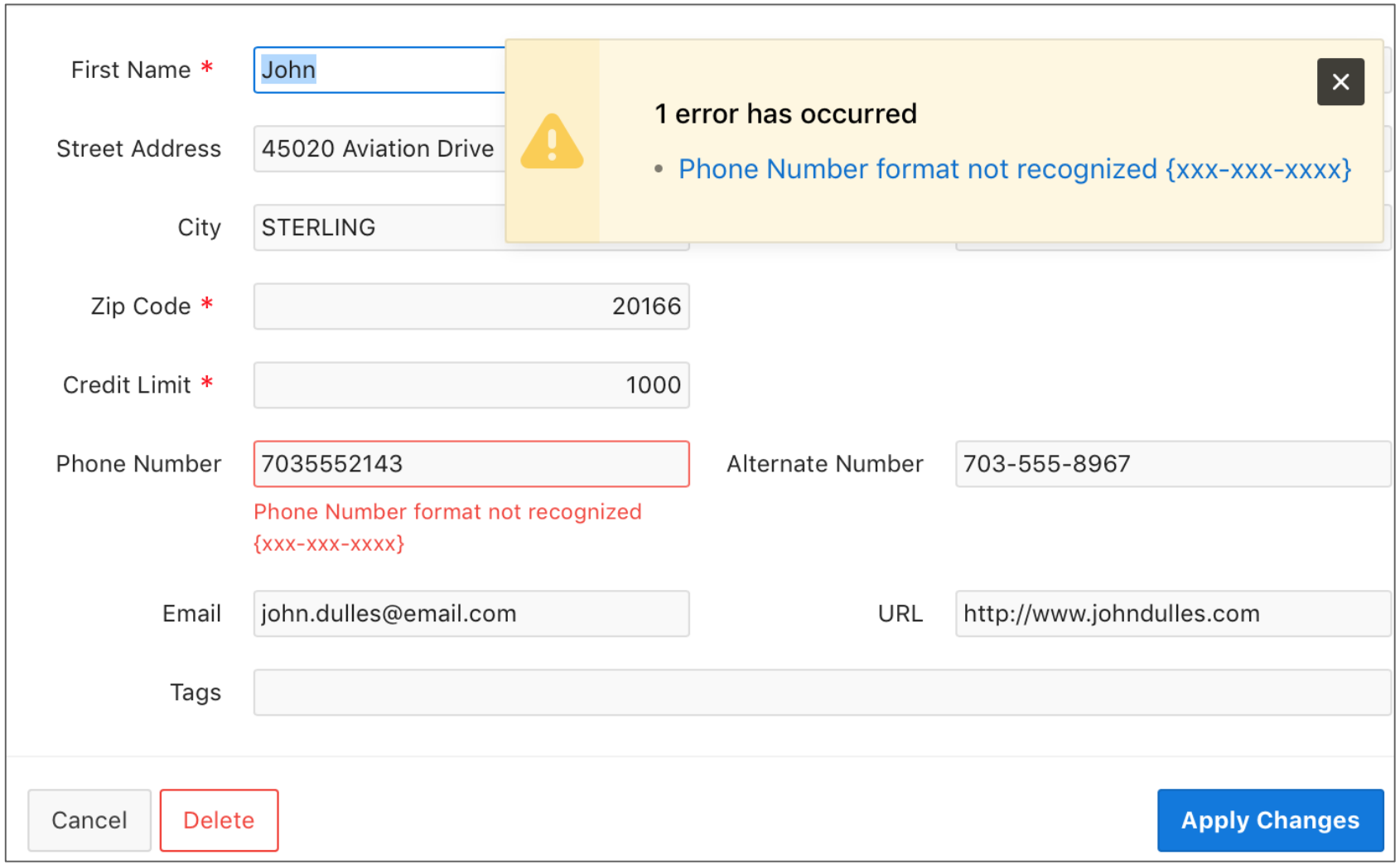


Create a validation to ensure at least one product must be added to the cart to complete the order in the Place Order wizard.

Creating a PL/SQL Validation Type: Steps

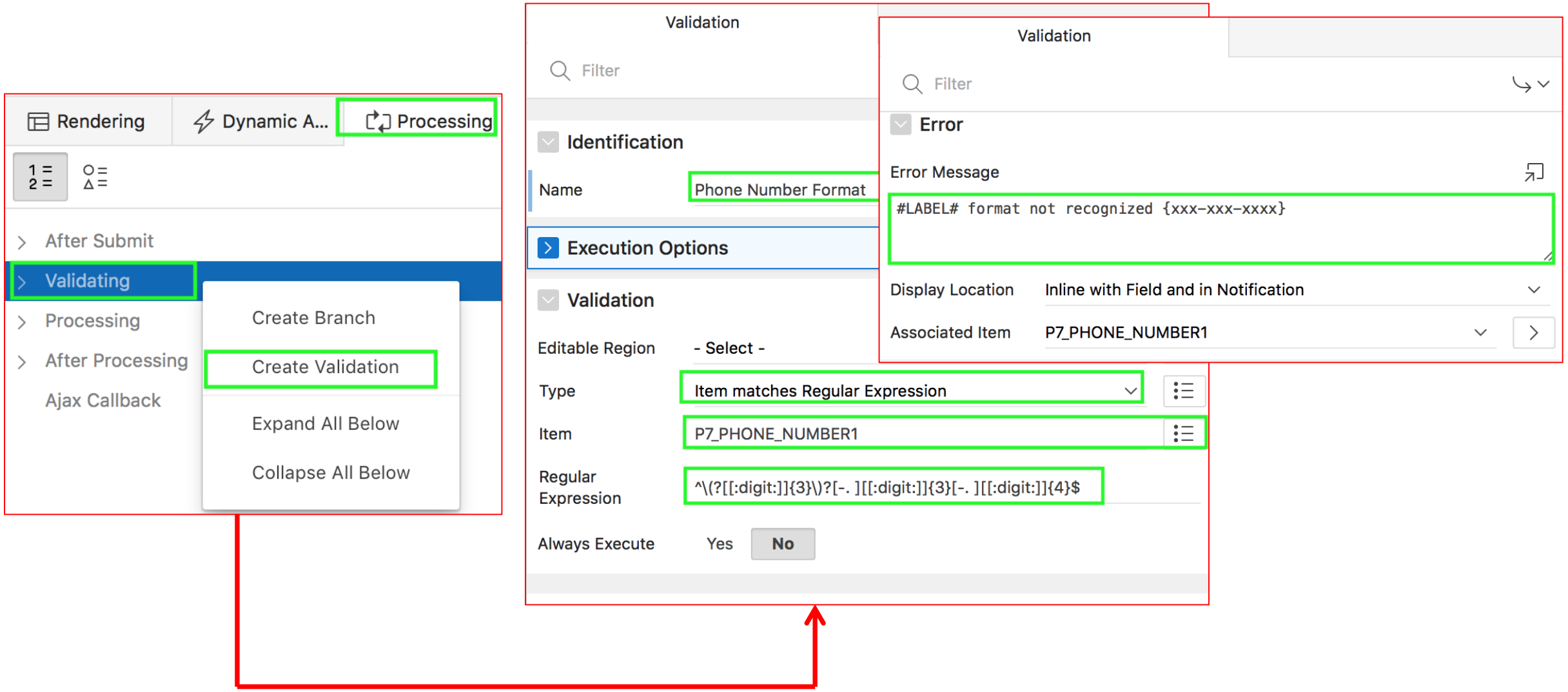


Creating a Regular Expression Validation Type - Example

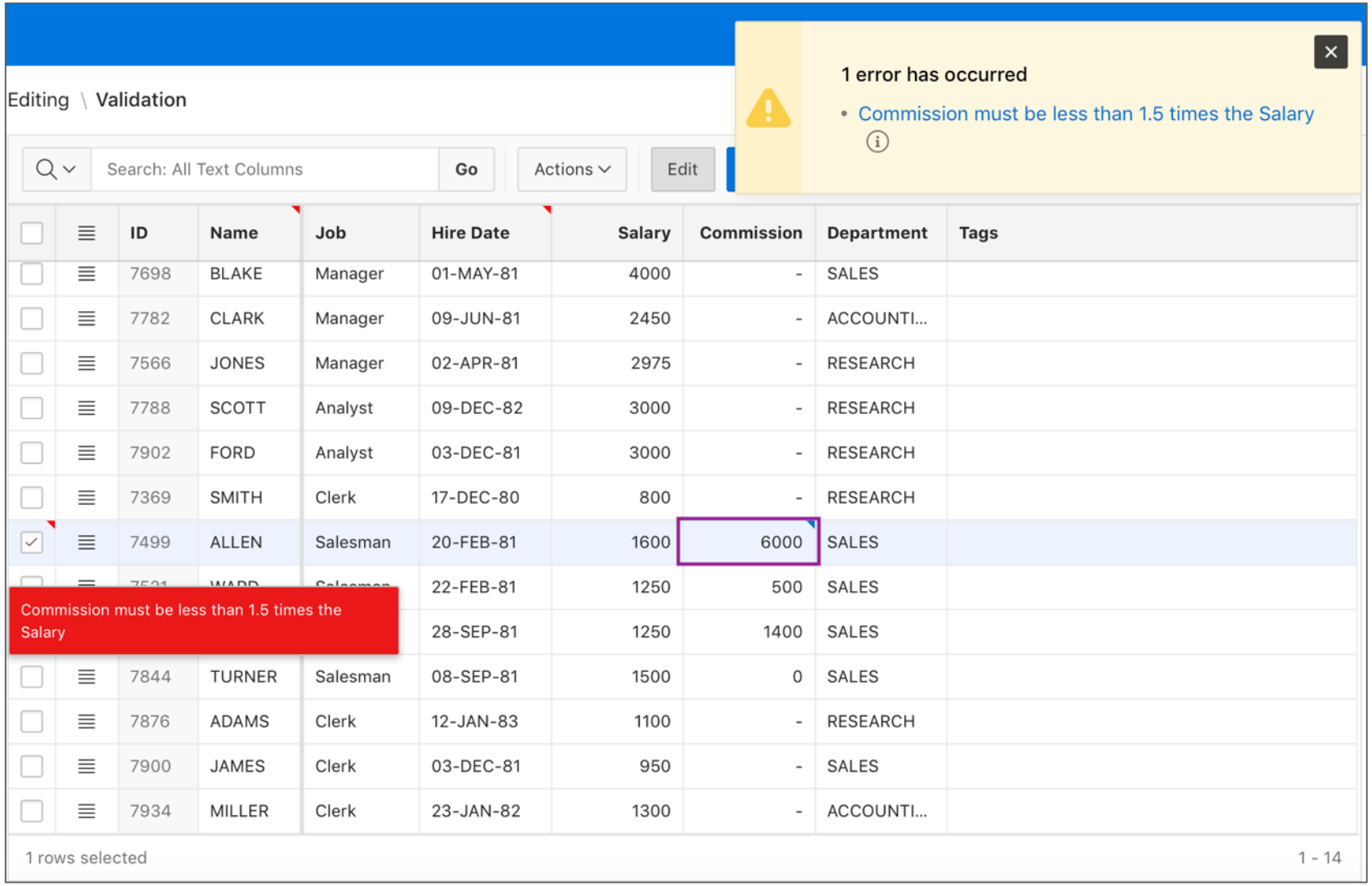


Create a validation to ensure that the customer's phone number entered is in the xxx-xxx-xxxx format.

Creating a Regular Expression Validation Type: Steps

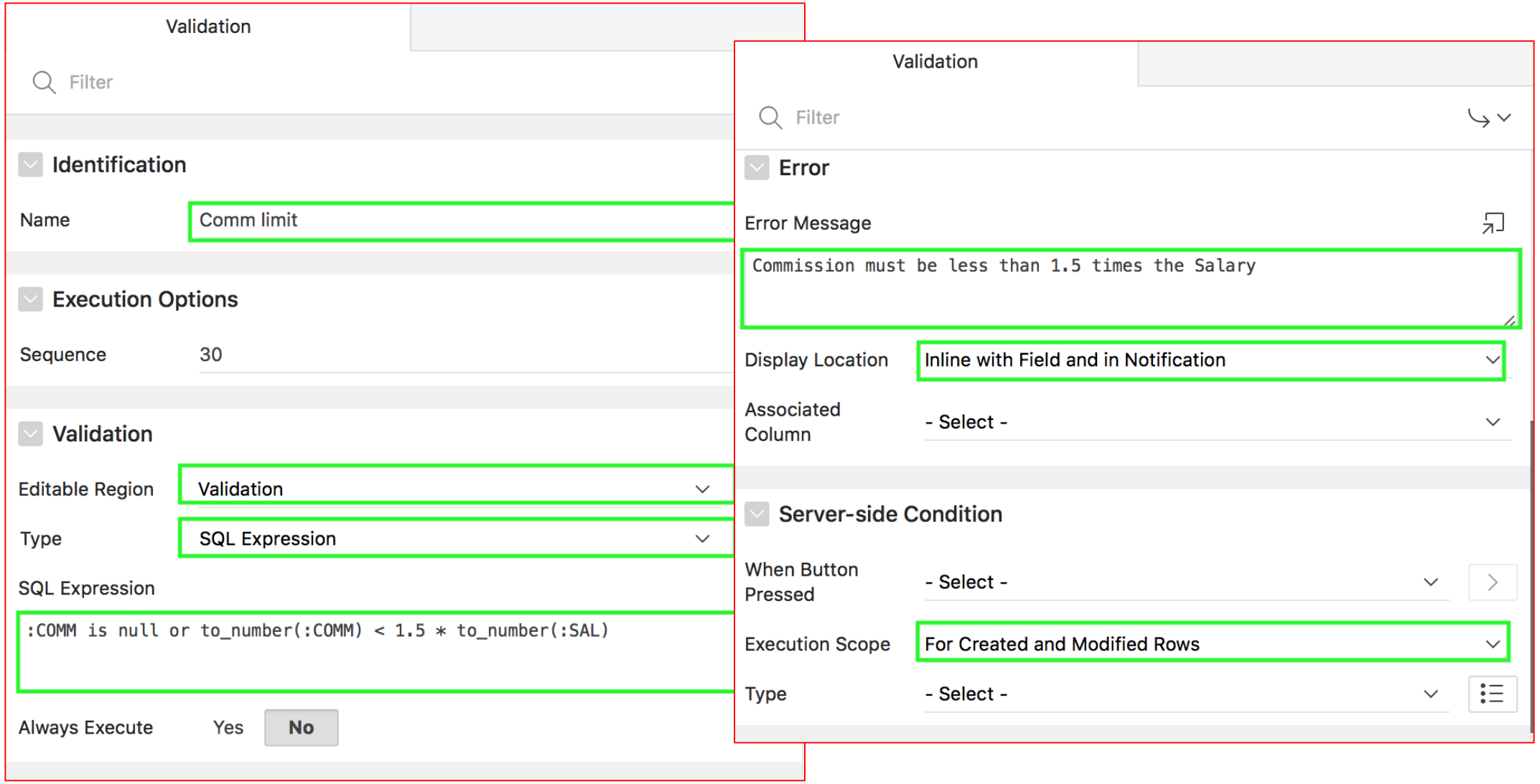


Creating a Row Validation in an Interactive Grid - Example

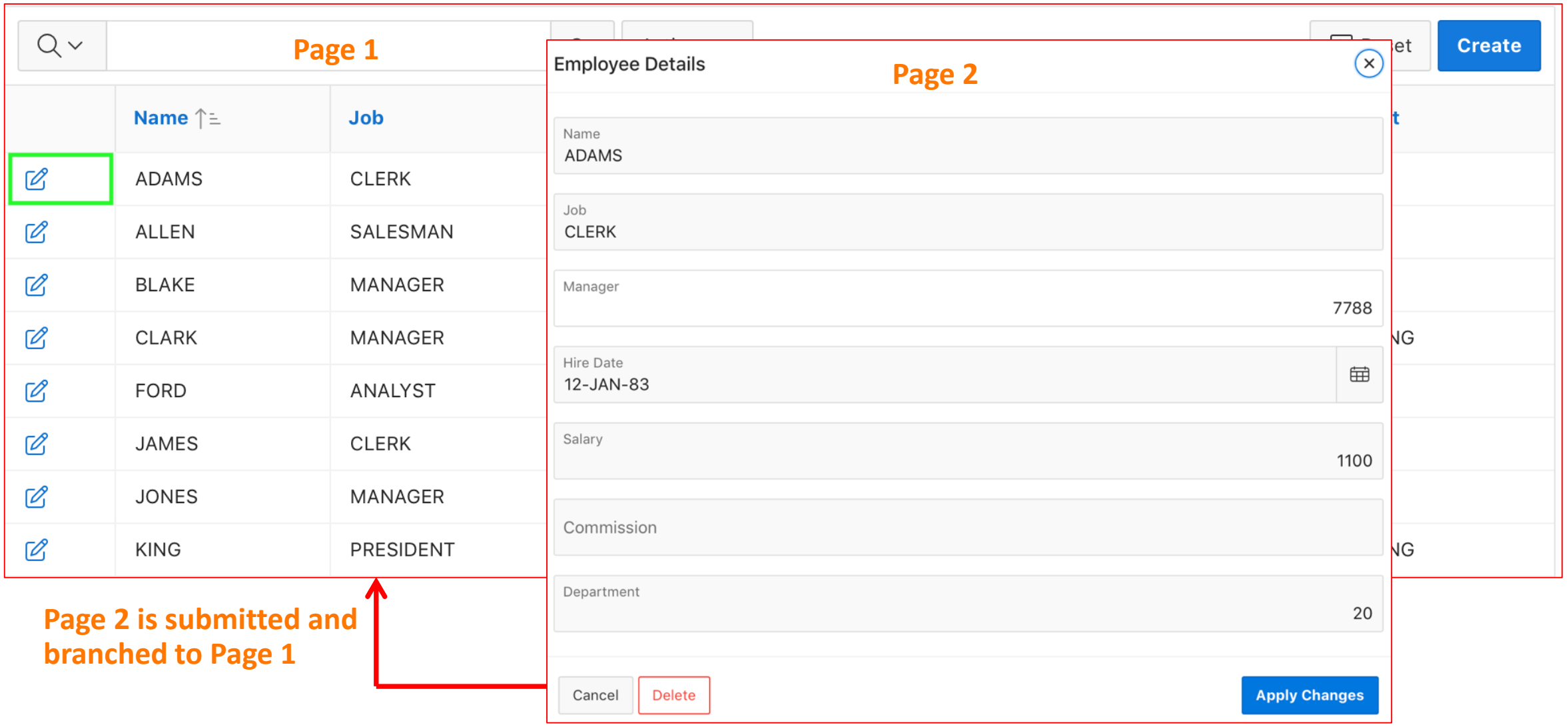


Create a row validation in an editable interactive grid region. This validation ensures commission is less than 1.5 times the salary.

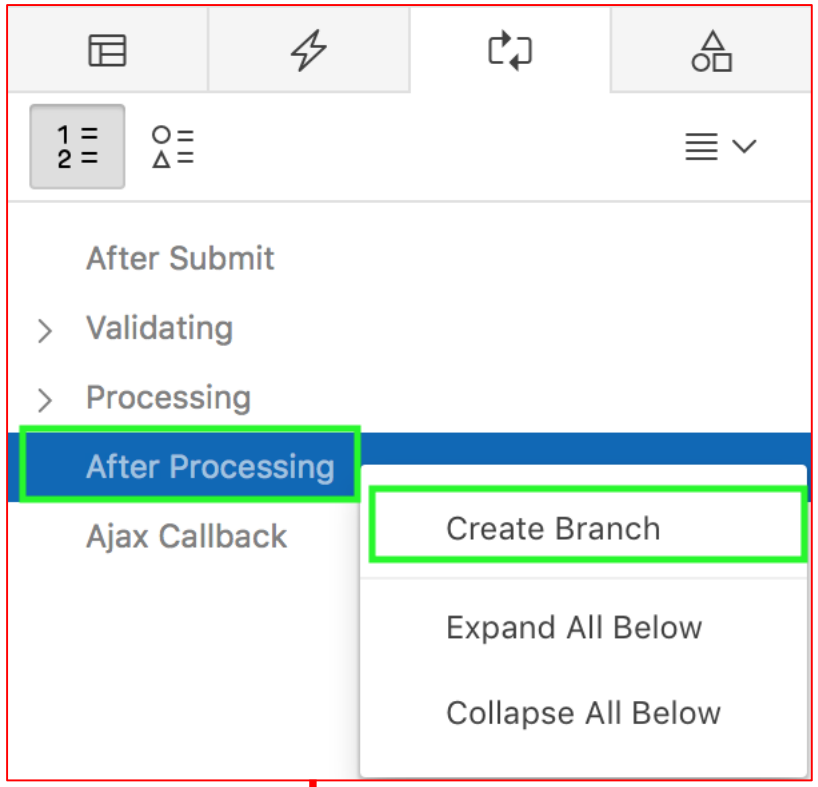
Creating a Row Validation in an Interactive Grid: Steps

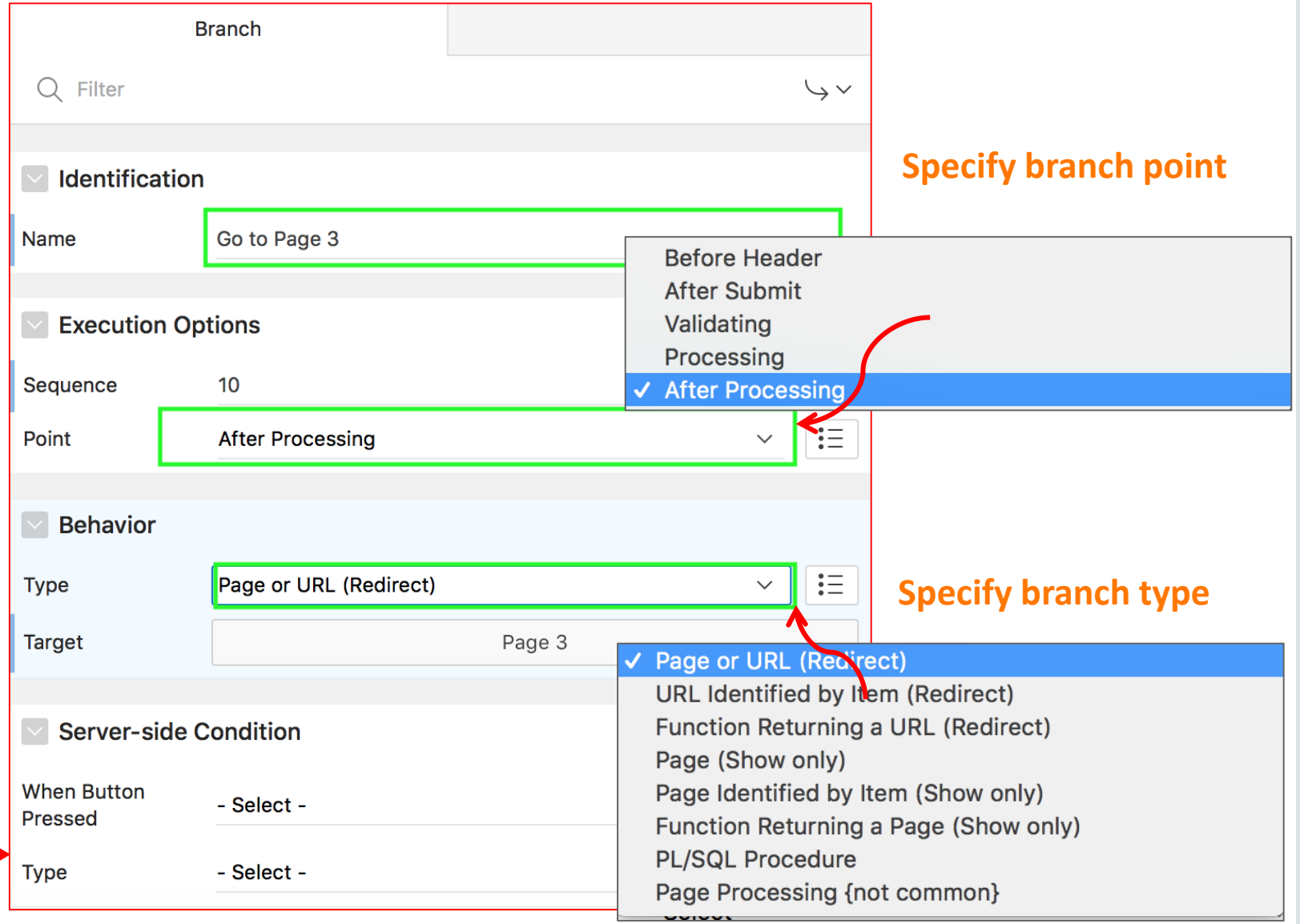


Understanding Branching



Creating a Branch





Specify branch point

Specify branch type

Summary

In this lesson, you learned how to create :

- Computations
- Processes
- Validations
- Branches

Hands-on Lab


Oracle APEX

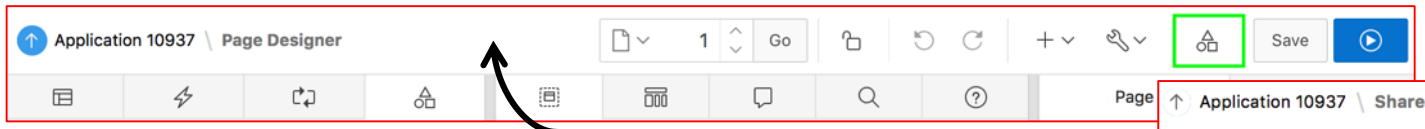
Unit 11: Implementing Navigation in your Application

Lesson Objectives

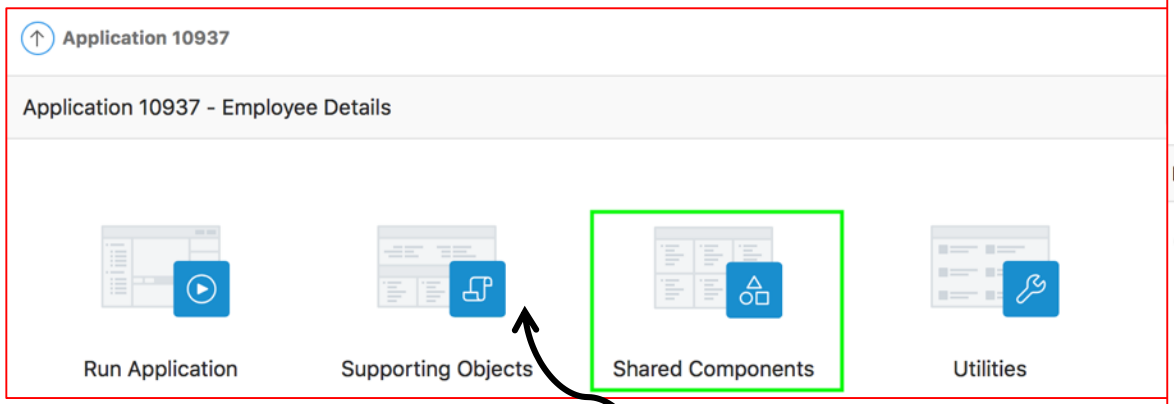
After completing this lesson, you should be able to:

- 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

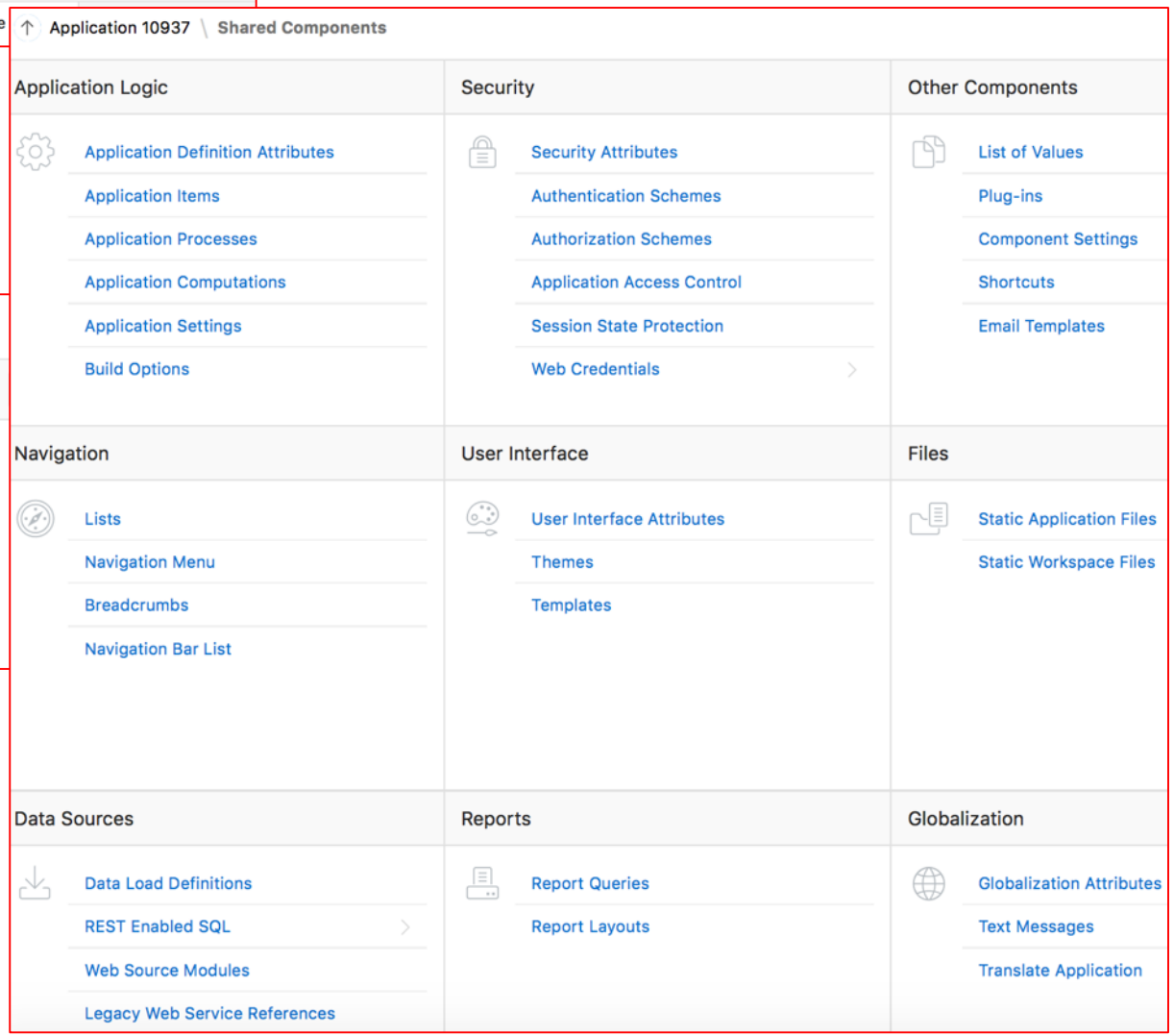
Understanding Shared Components



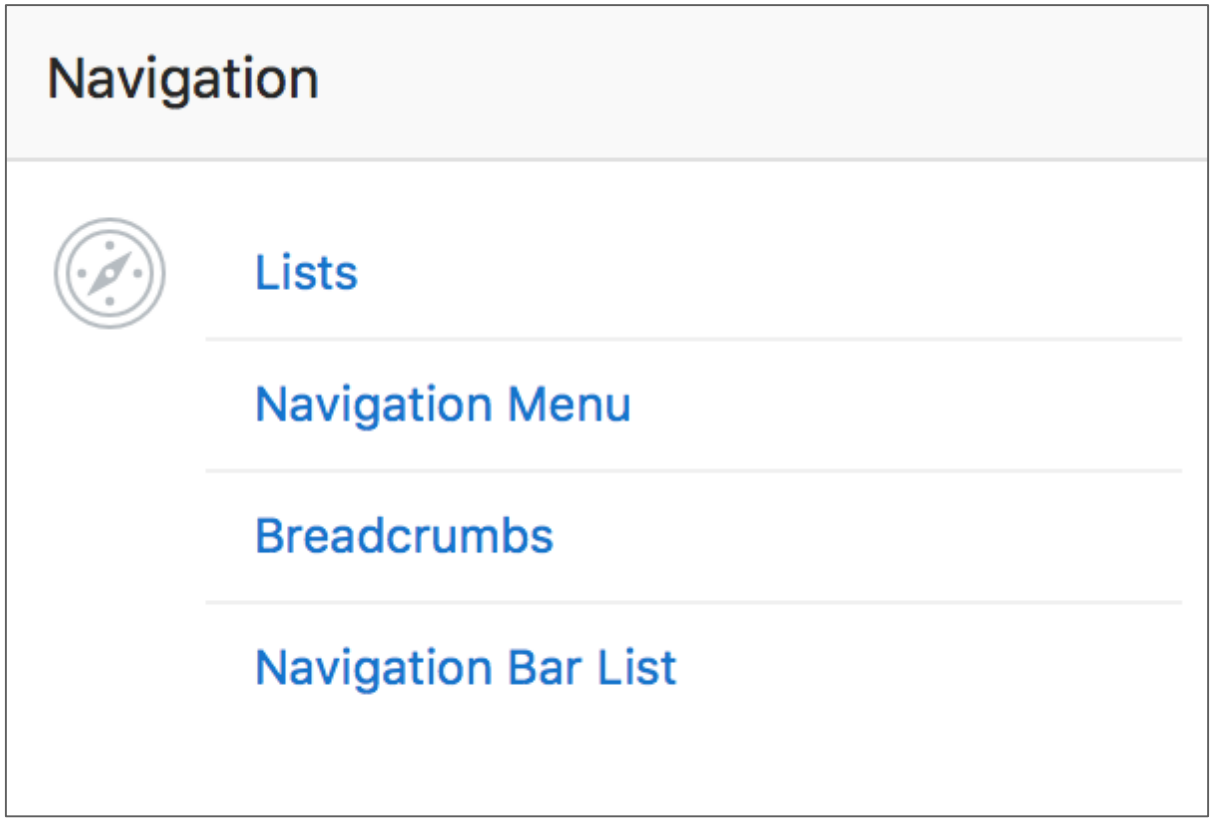
Page Designer



Application Home Page

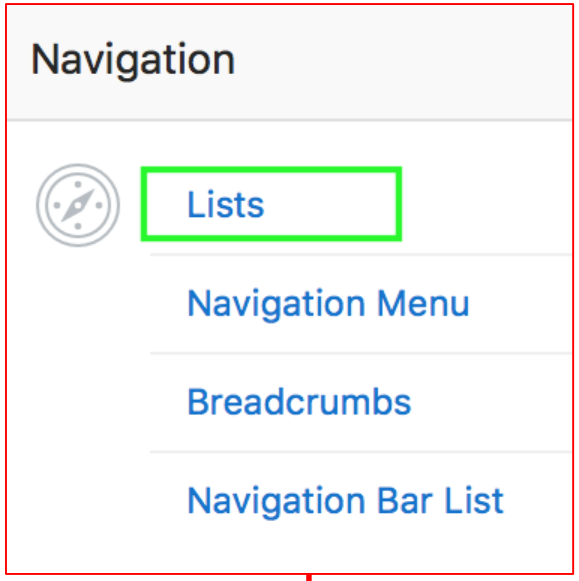


Understanding Navigational Shared Components



What are Lists?

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



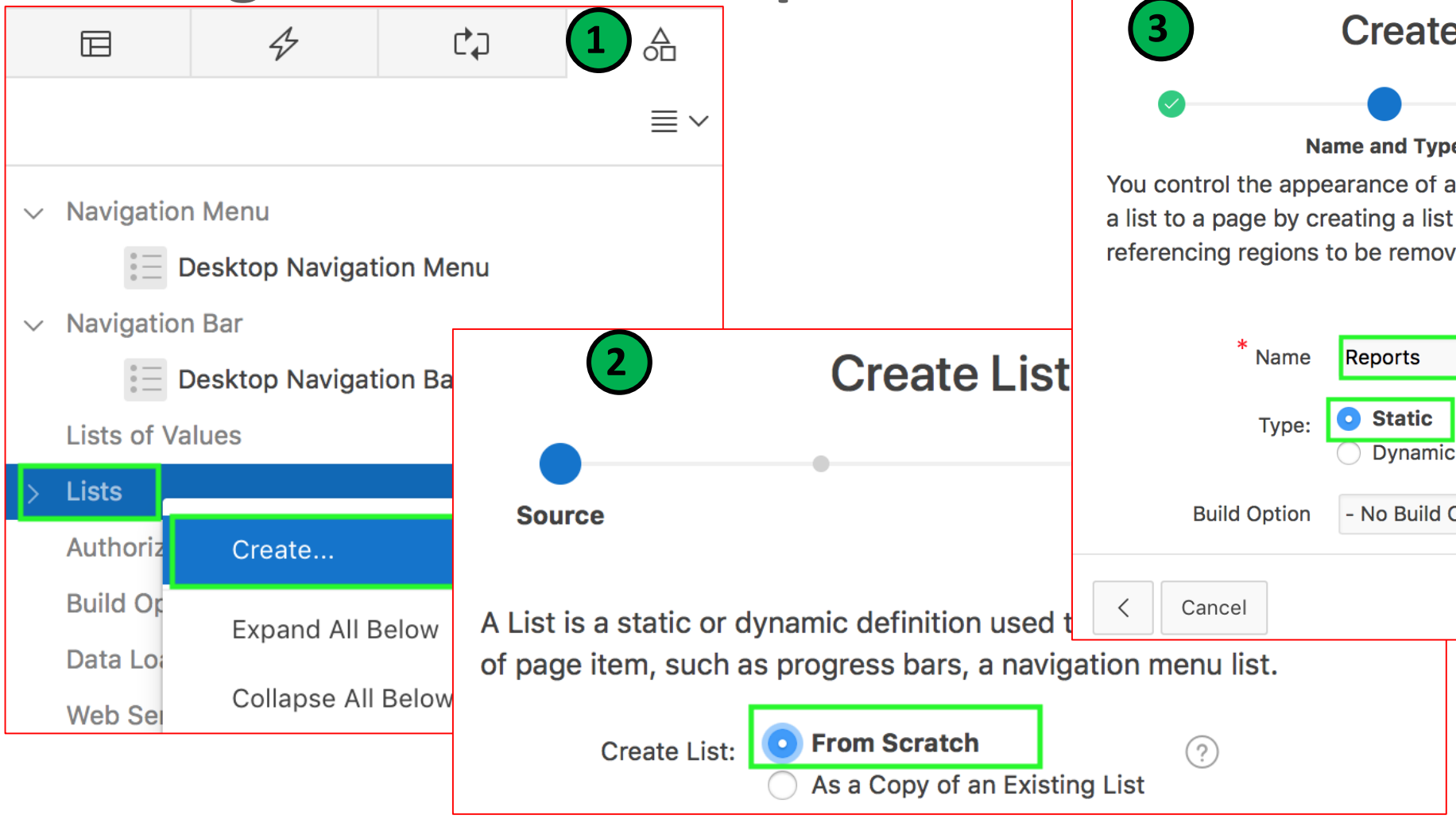
Application 94366 \ Shared Components \ Lists

Lists | List Details | Unused | Conditional Entries | Utilization | History

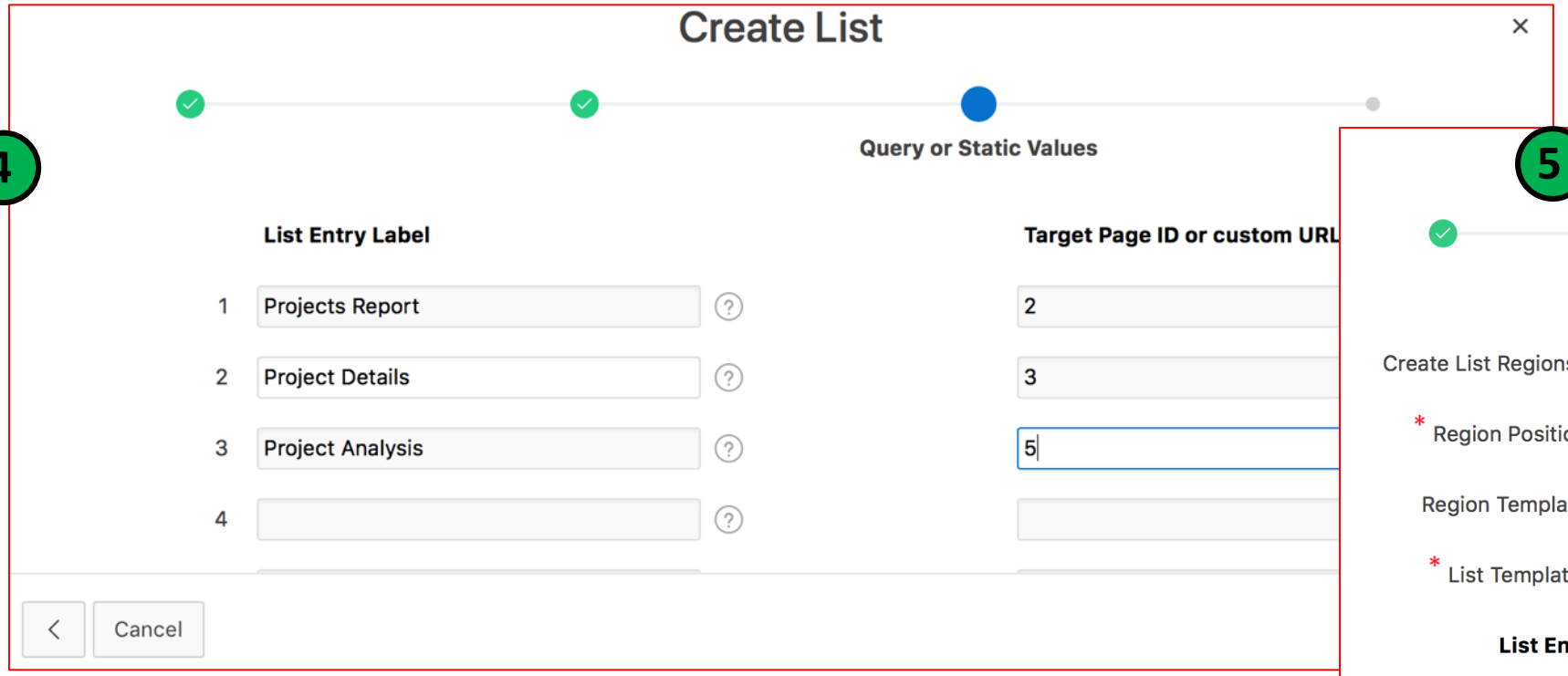
Go | Actions | Reset | Copy | Create

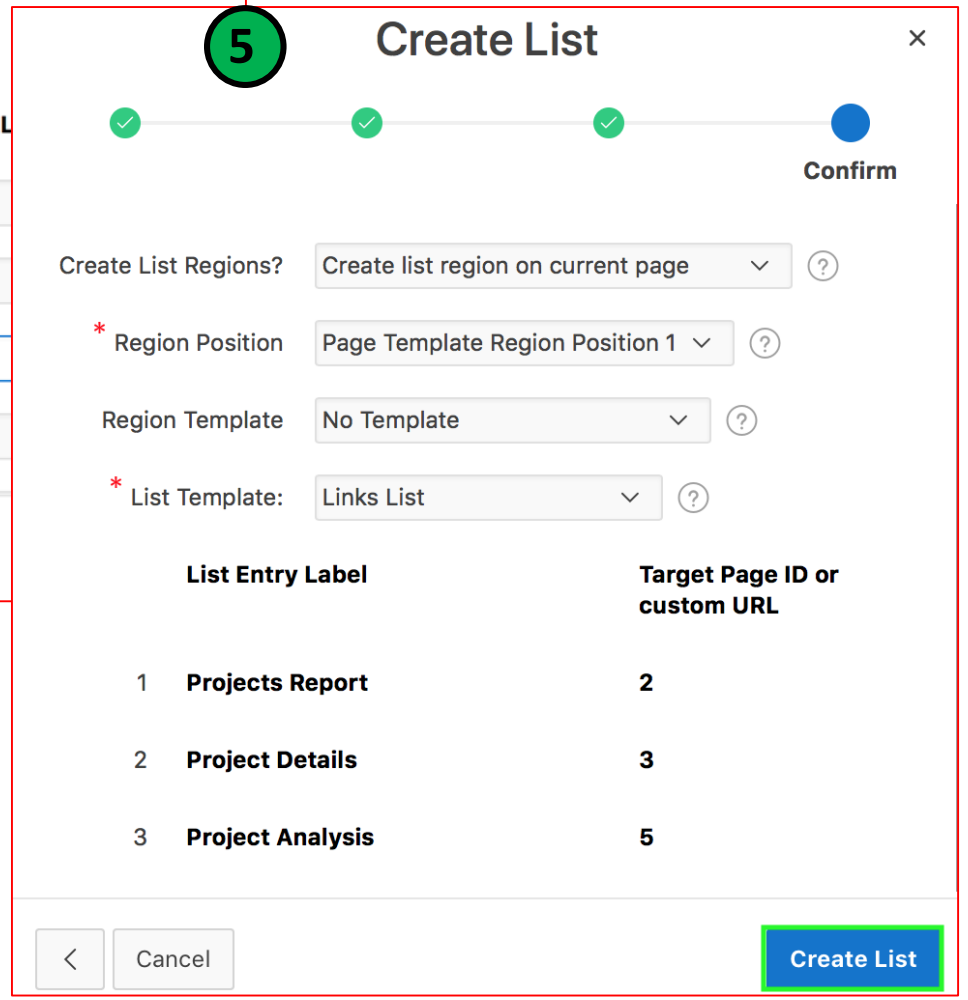
Name	Type	Entries	References	Entries Updated	List Updated	Navigation Bar	Navigation Menu
Activity Reports	Static	5	1	-	-	No	No
Administration	Static	3	1	-	-	No	No
Application Navigation	Static	15	0	-	-	No	Yes
Data Load Wizard Progress - Load Customers	Static	4	4	-	-	No	No
Feedback	Static	2	1	-	-	No	No
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

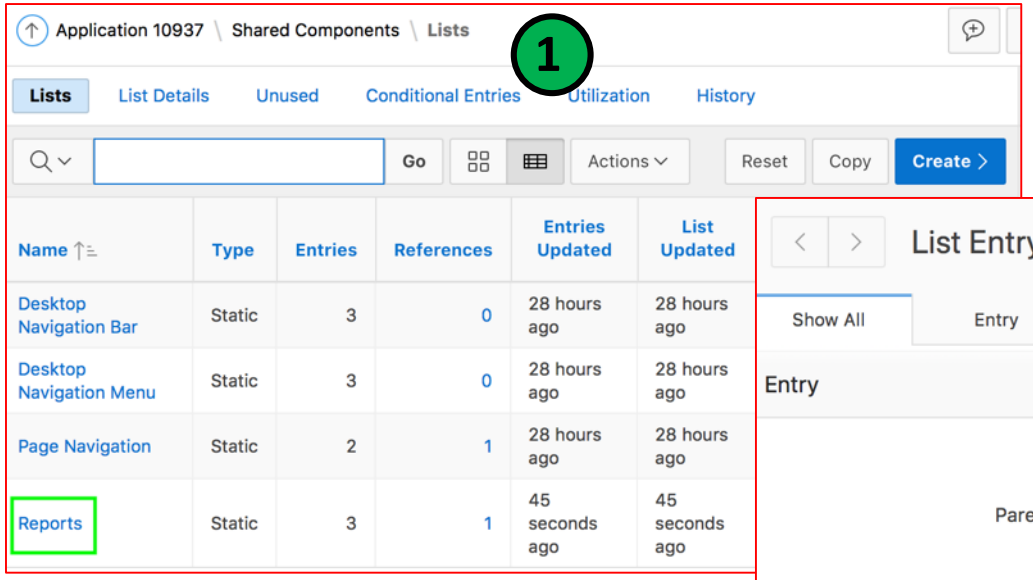


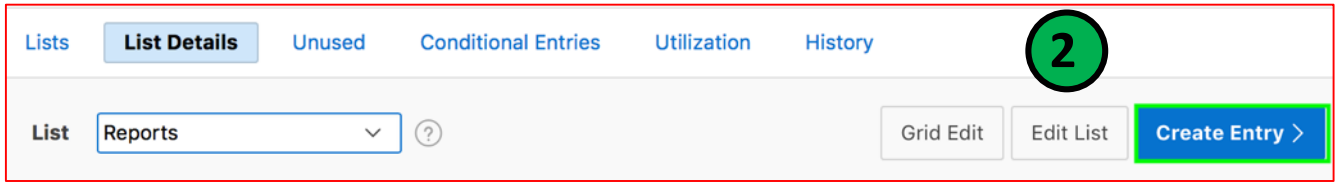
Creating a Static List: Steps

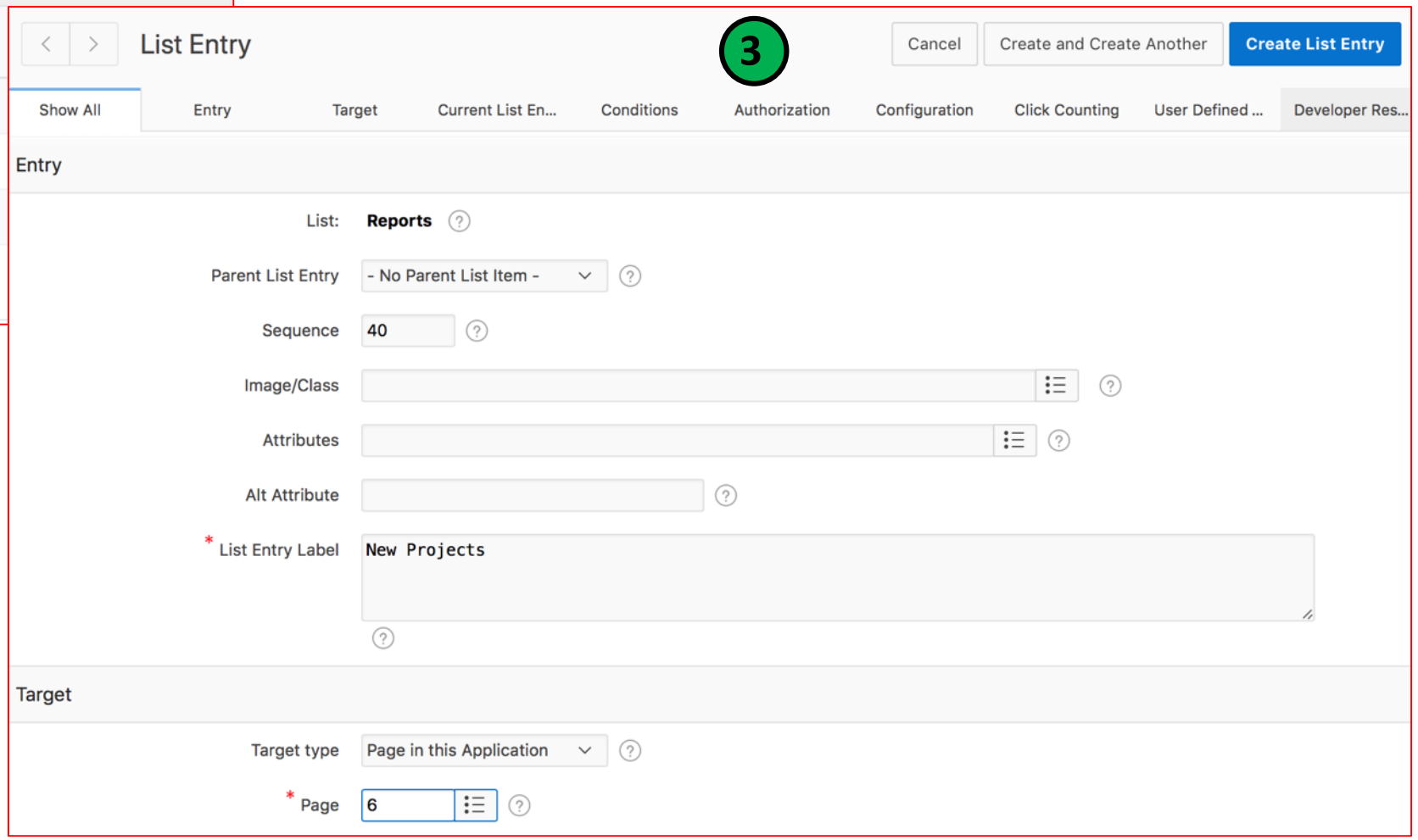




Adding Entries to a List



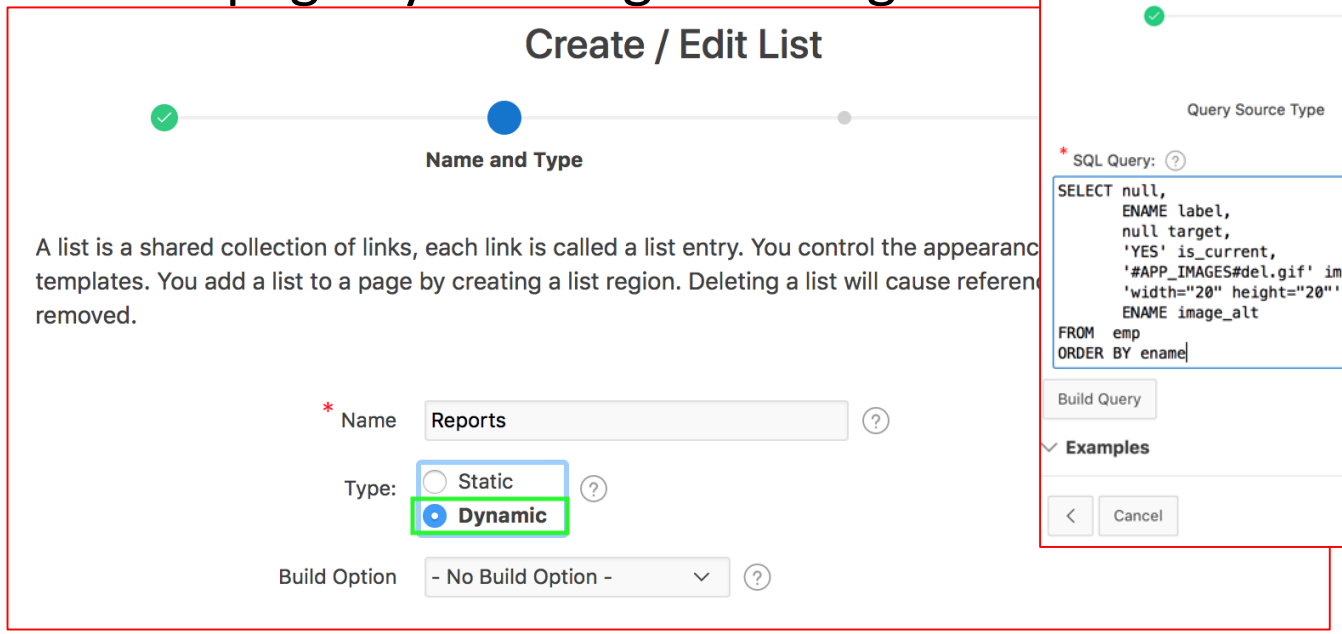


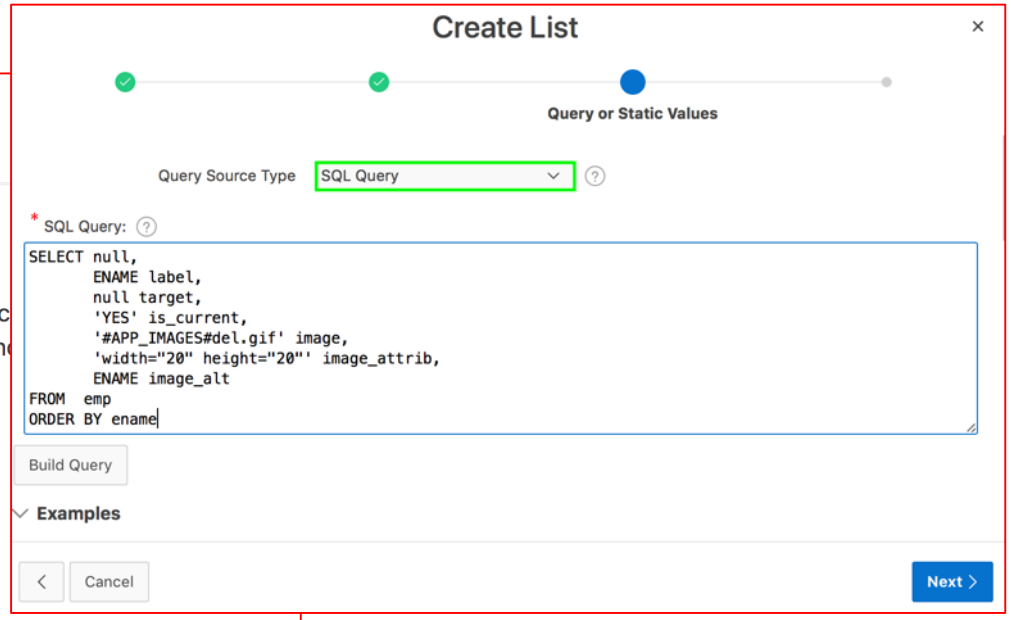


Creating a Dynamic List

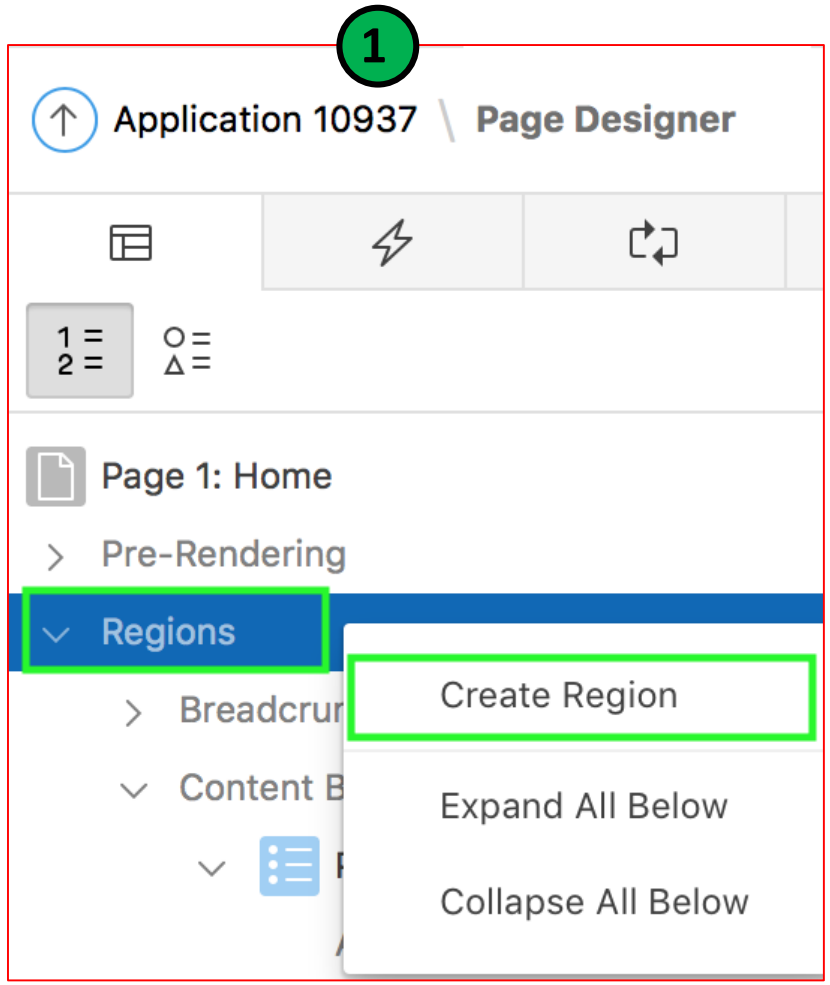
Steps:

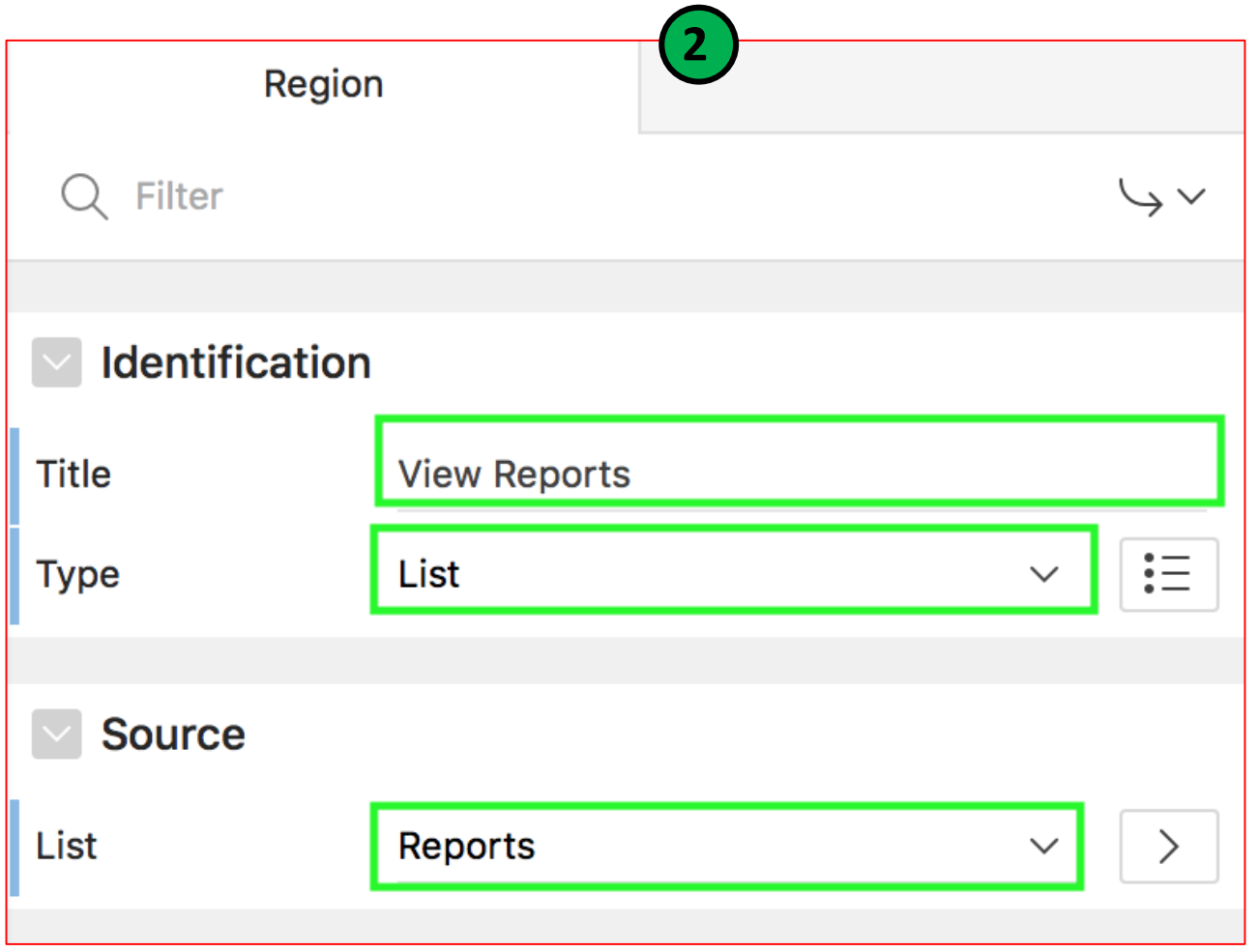
1. Create the list by running the Create List Wizard.
2. Specify whether to create the list from scratch or by copying an existing list.
3. If creating a list from scratch, you are prompted to select a list type. Select Dynamic.
4. Enter a SQL query or a PL/SQL function returning a SQL query.
5. Add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



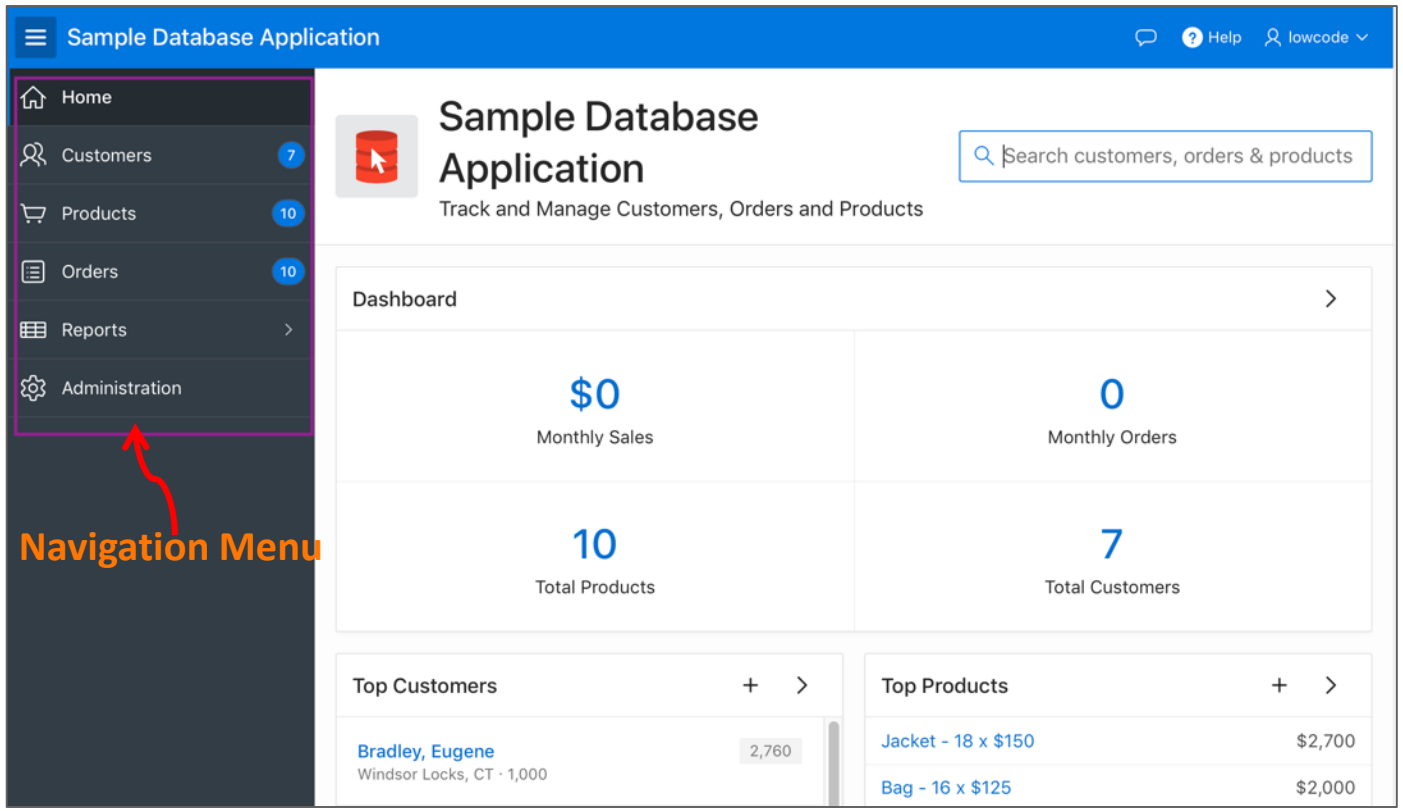


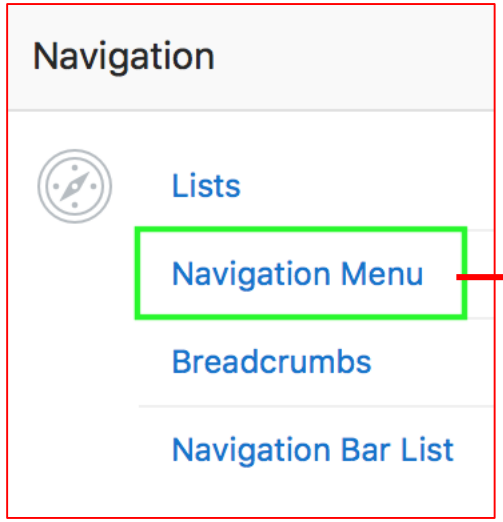
Adding a List to a Page

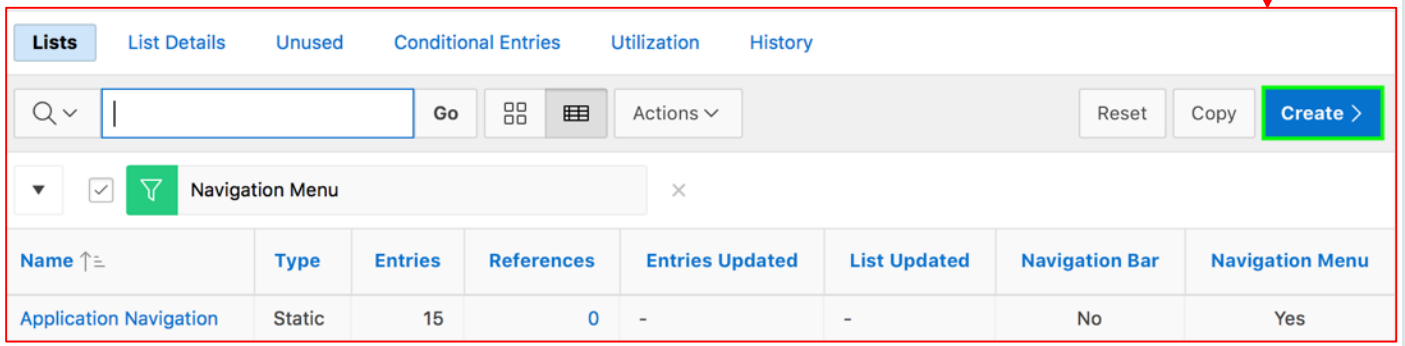




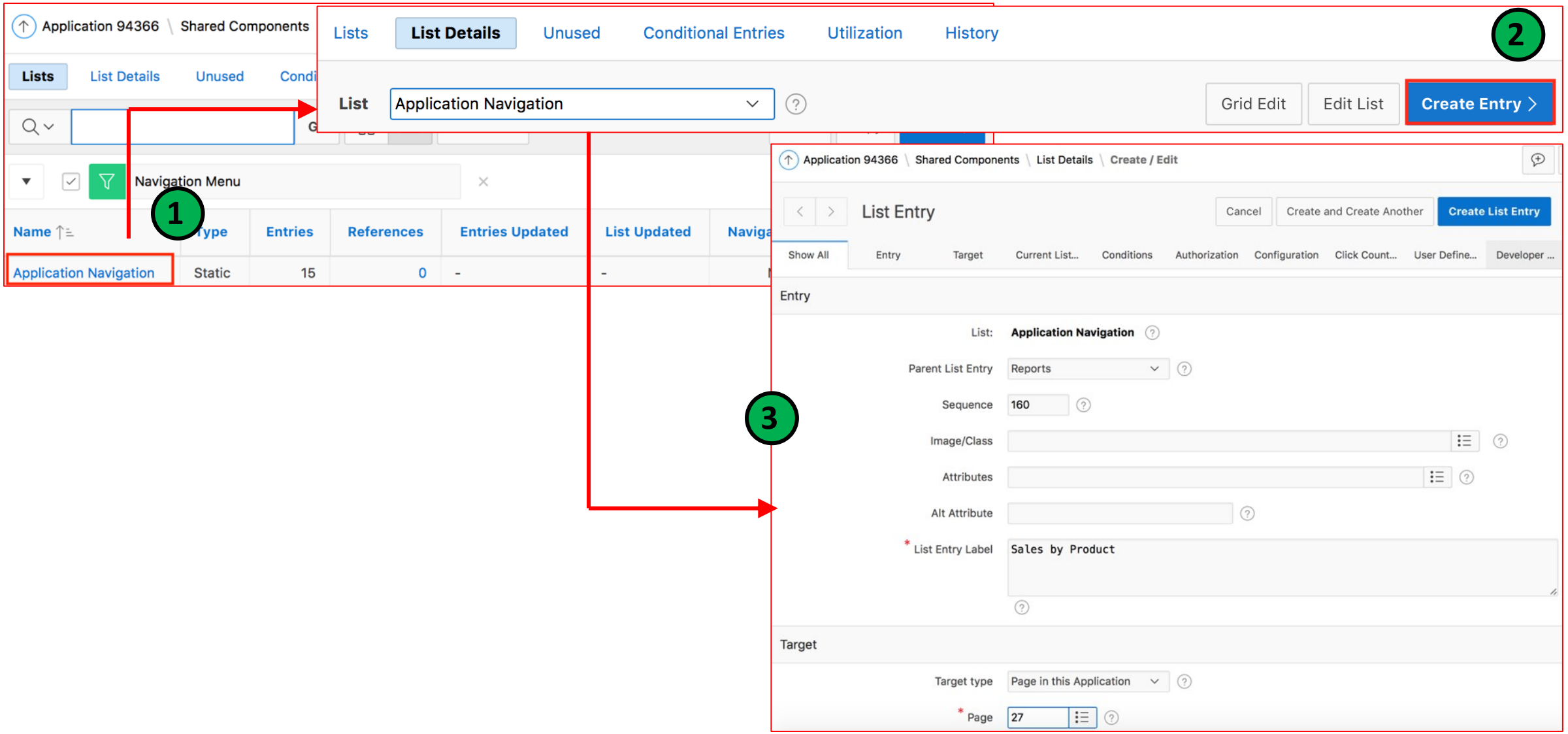
What is a Navigation Menu?





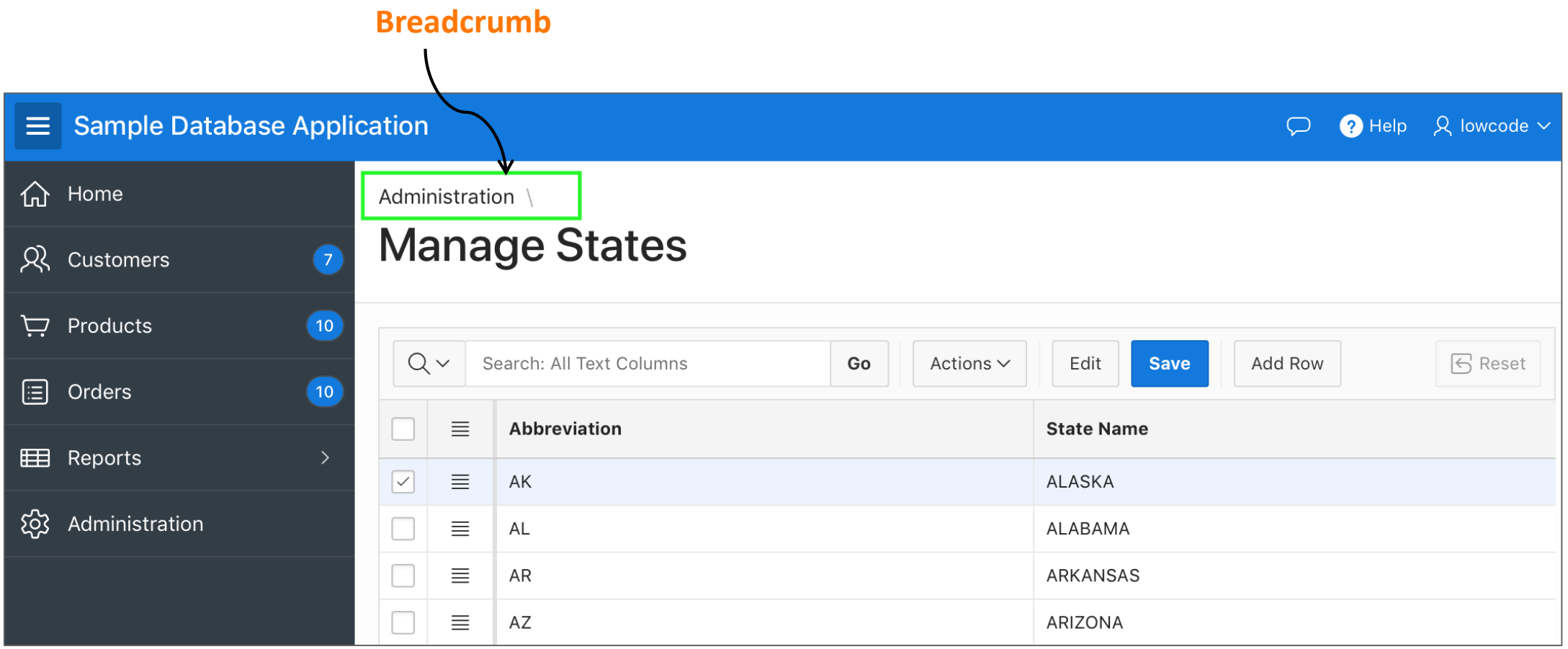


Creating a Navigation Menu Entry



What is a Breadcrumb?

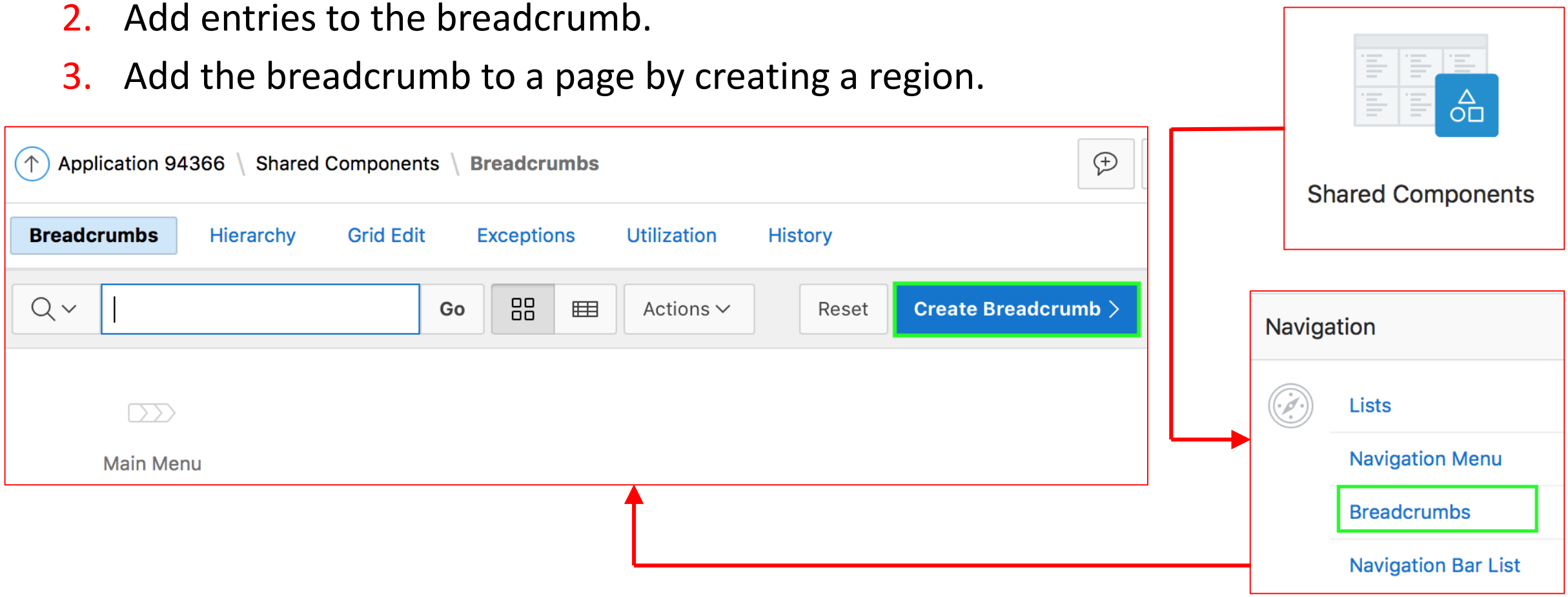
A breadcrumb is a hierarchical list of links that display using templates



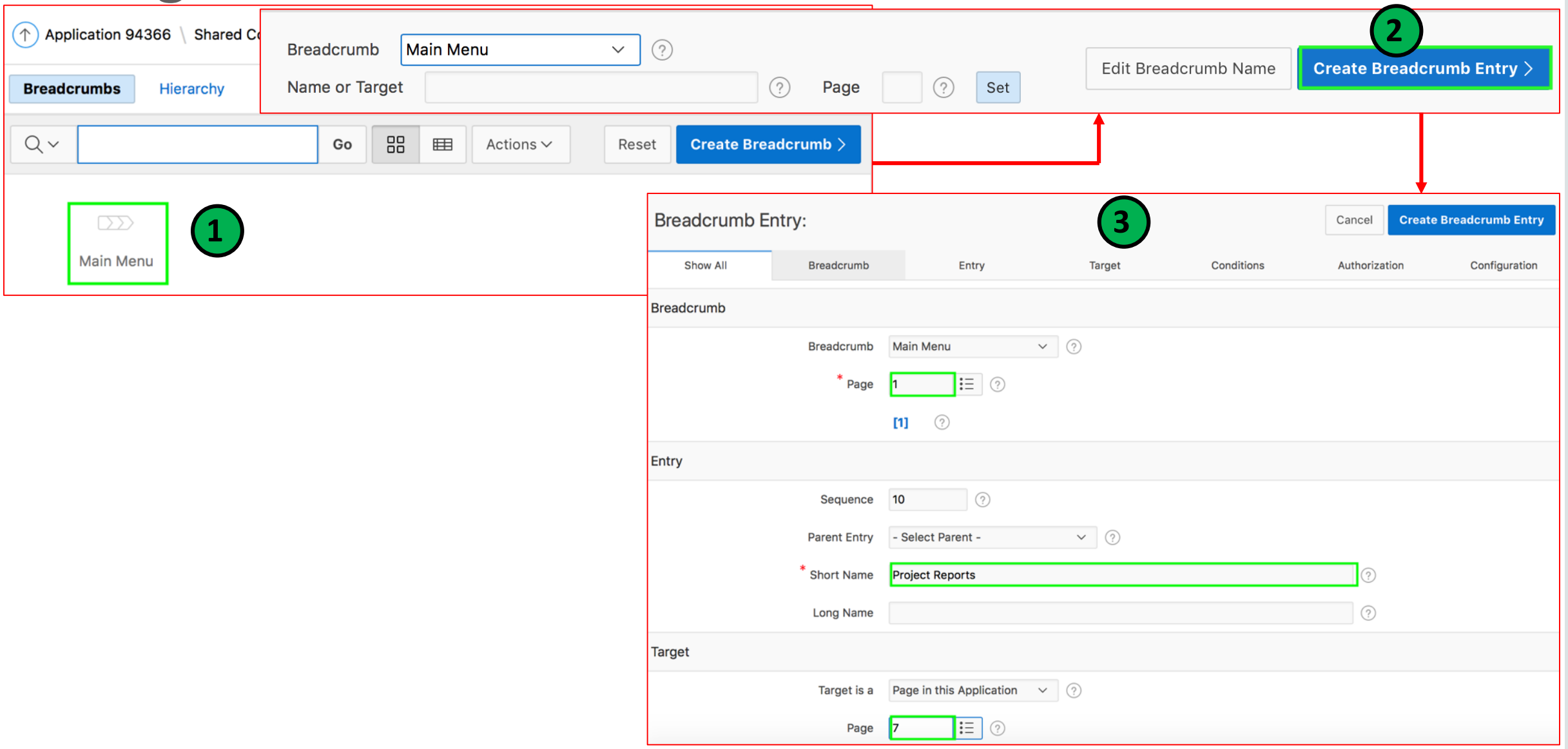
Creating a Breadcrumb

Steps:

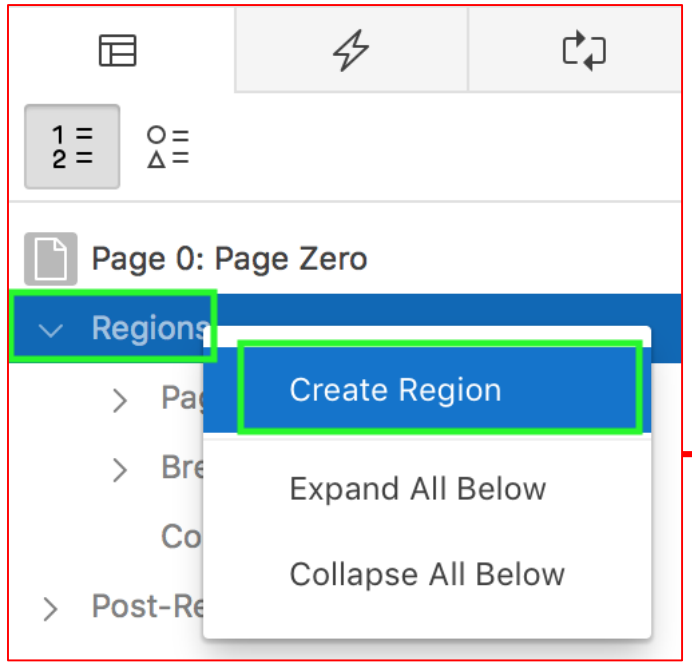
1. Create the breadcrumb by running the Create/Edit Breadcrumb Wizard.
2. Add entries to the breadcrumb.
3. Add the breadcrumb to a page by creating a region.

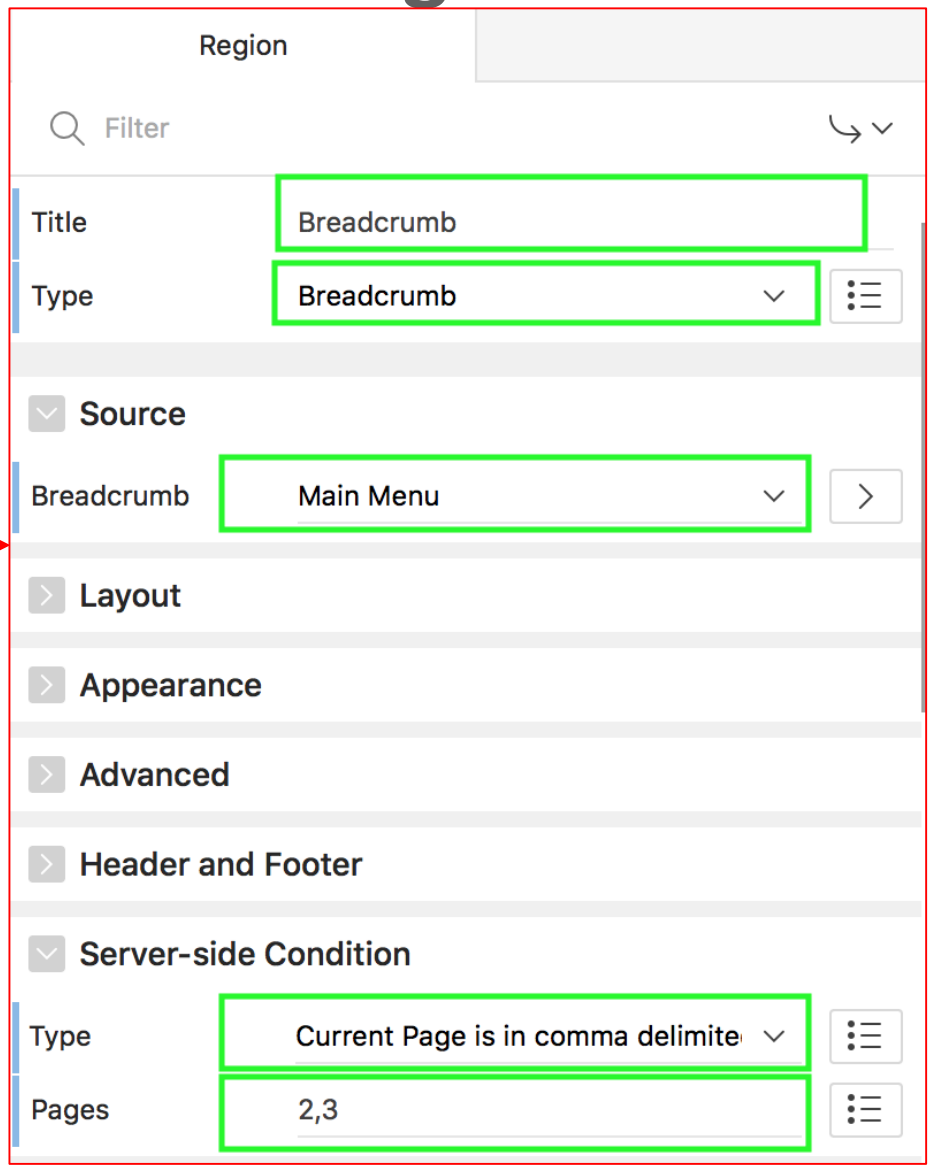


Adding Breadcrumb Entries



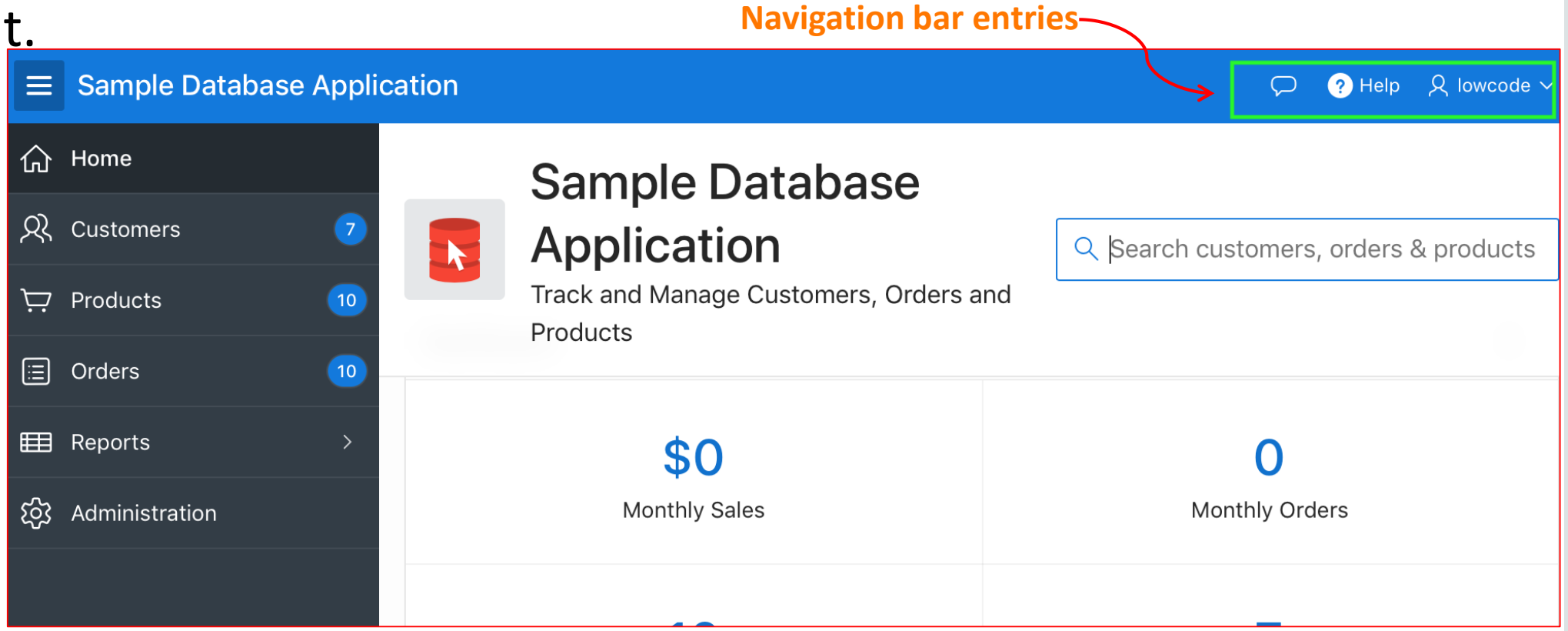
Adding a Breadcrumb Region in a Page



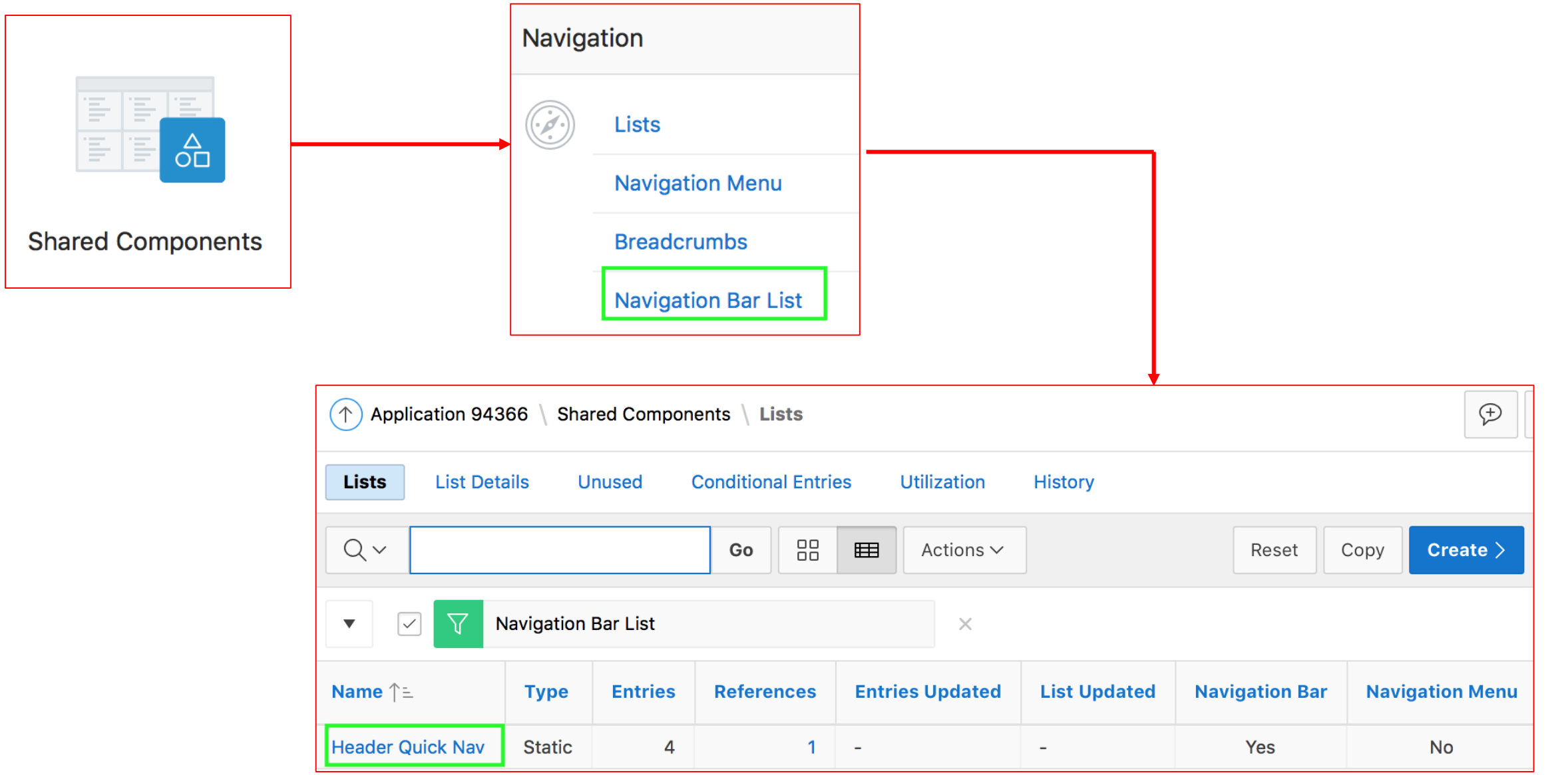


What is a Navigation Bar Entry?

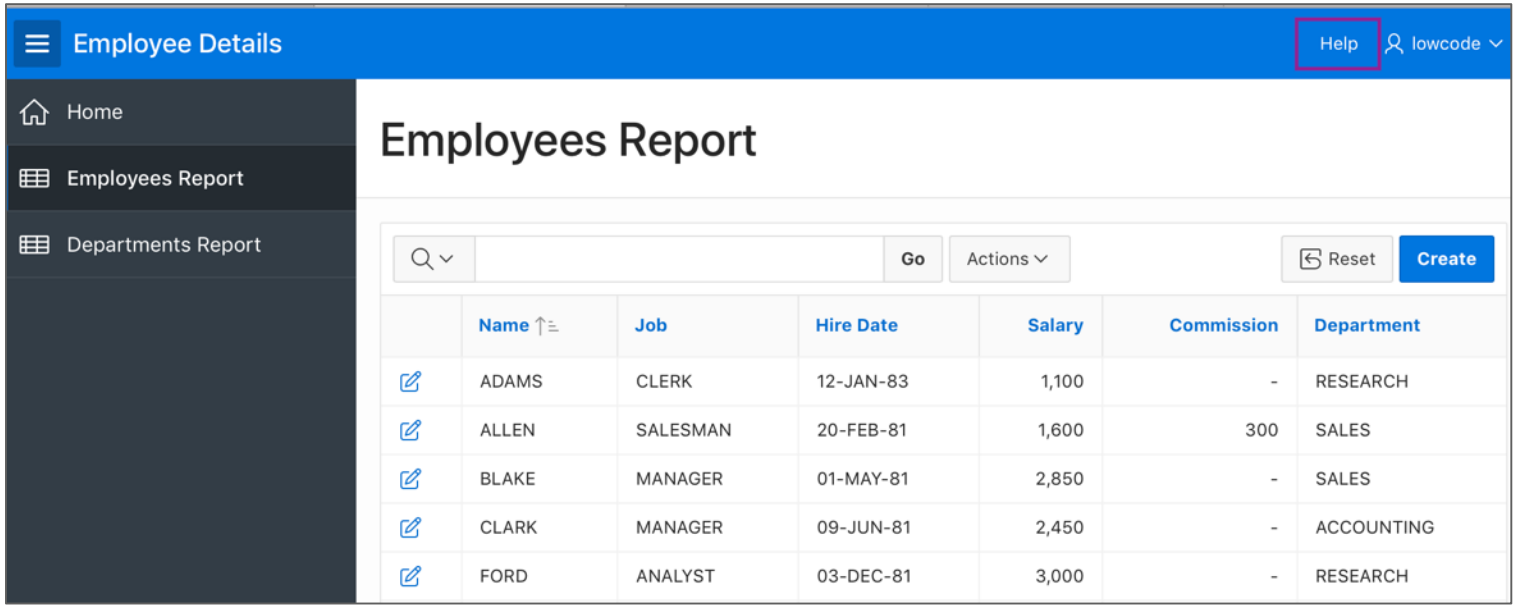
- Navigation bar entries offer an easy way to move users between pages in an application.
- A navigation bar entry can be an image, text, or an image with text beneath it.



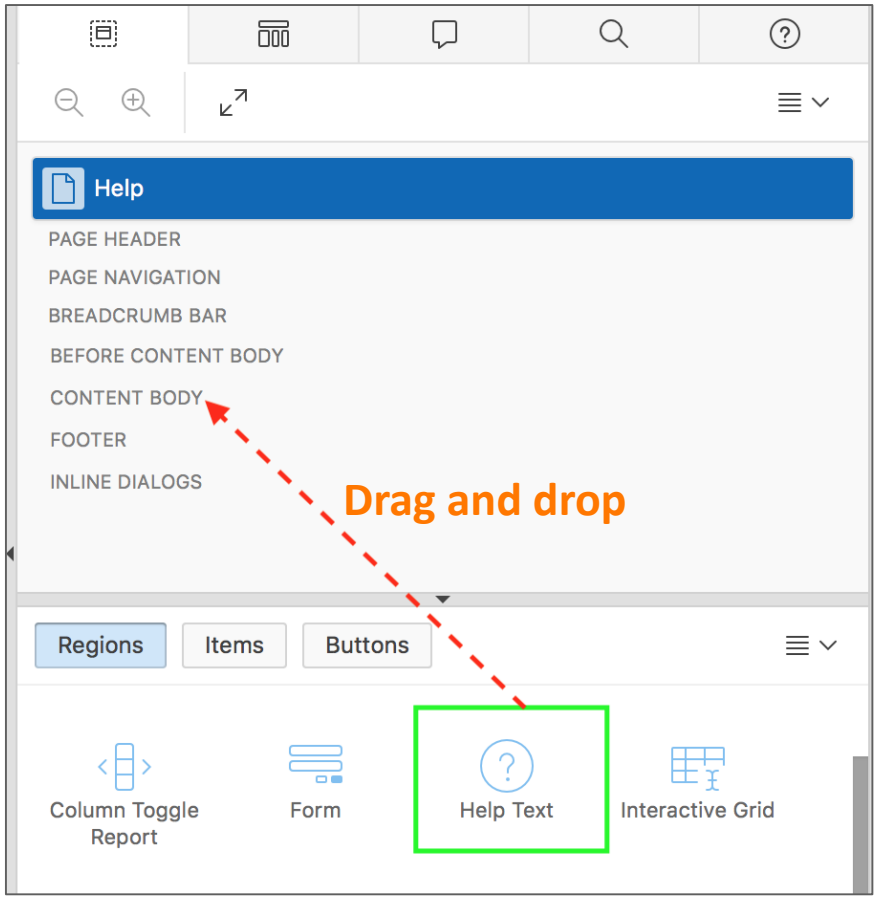
Accessing the Navigation Bar List



Creating a Navigation Bar Entry: Example

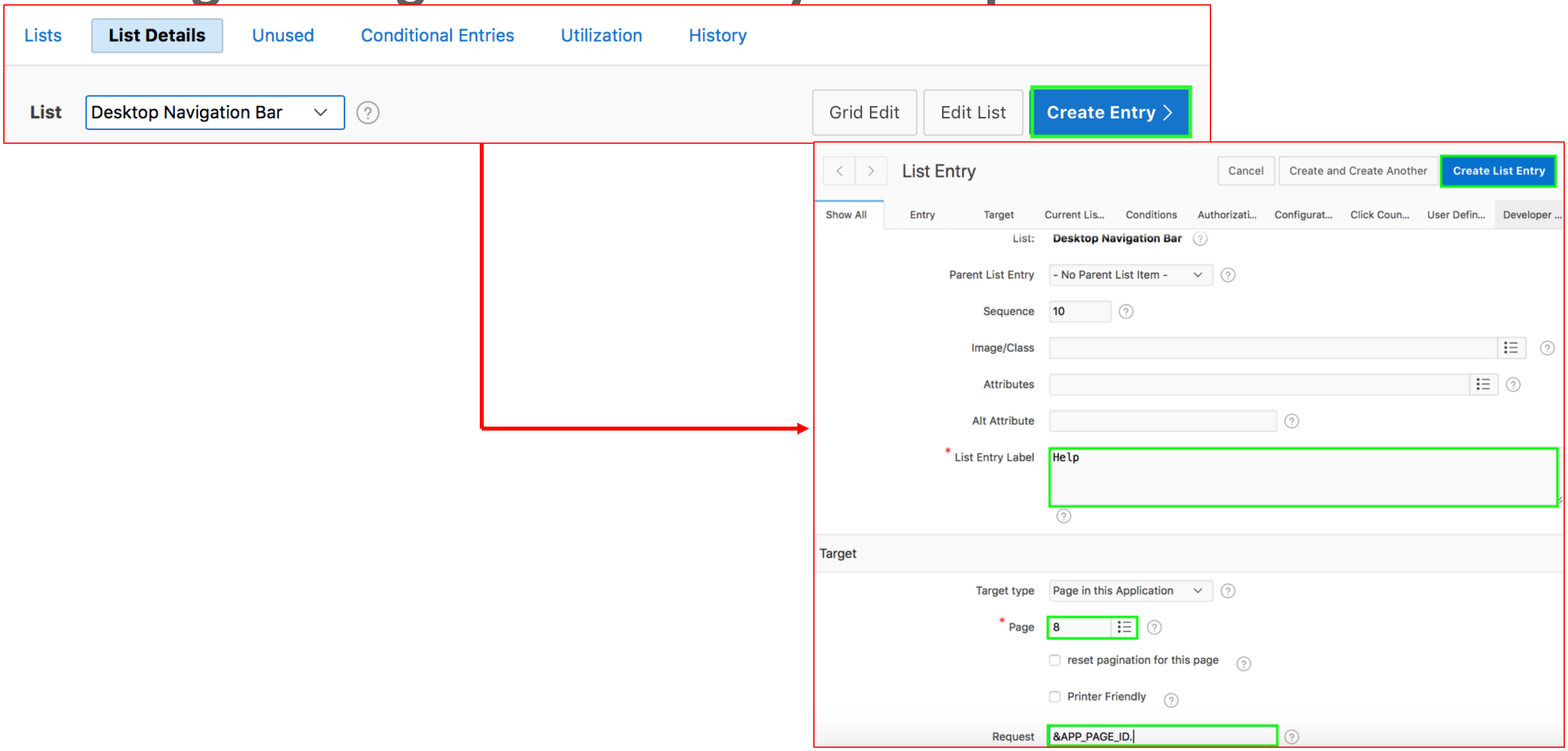


1. Create a blank page and add a Help Text region type
2. Create a navigation bar entry for Help



Adding a Help Text region type

Creating a Navigation Bar Entry: Example



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

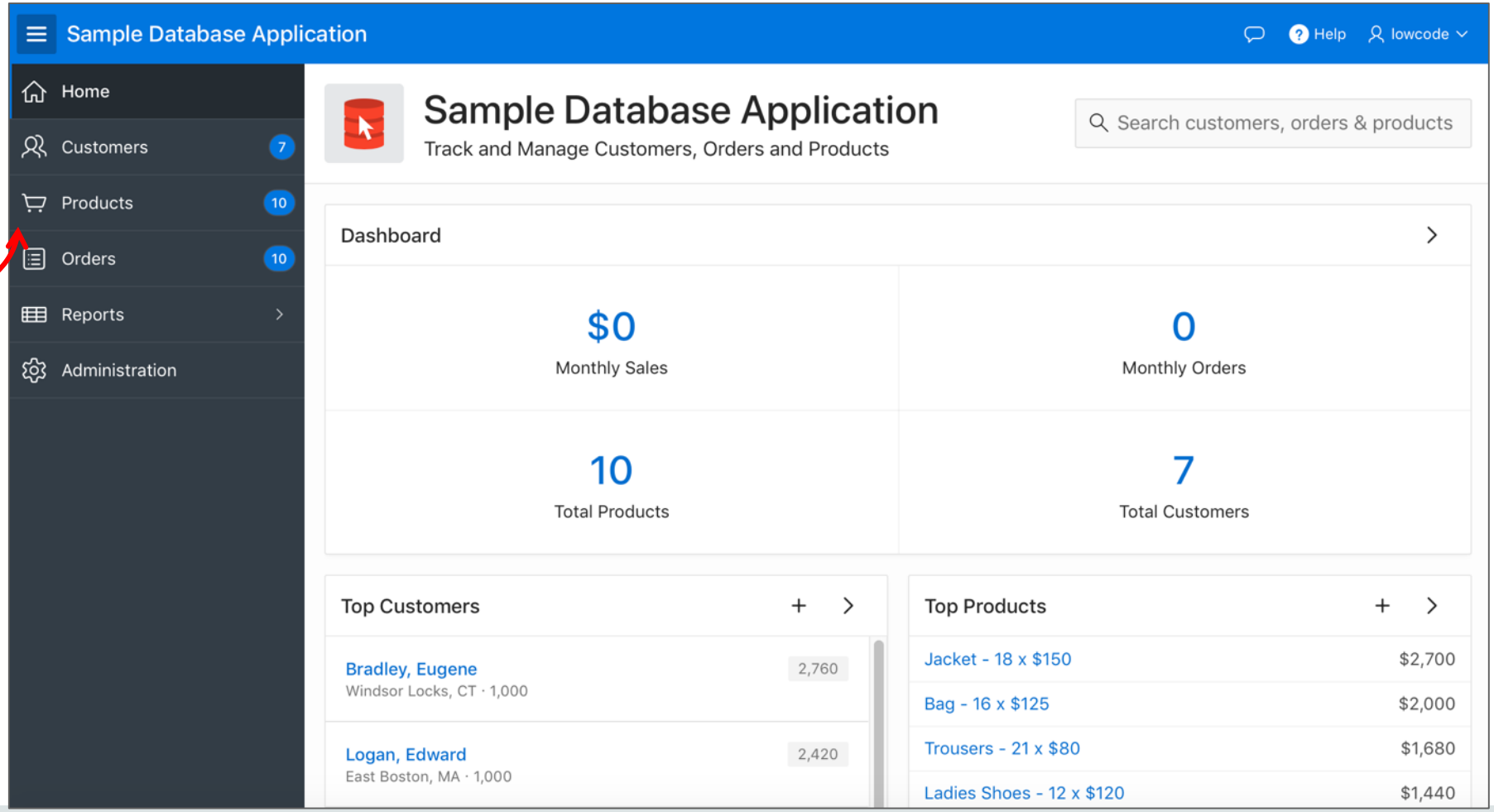
After completing this lesson, you should be able to:

- 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.

Universal Theme



What is Universal Theme?

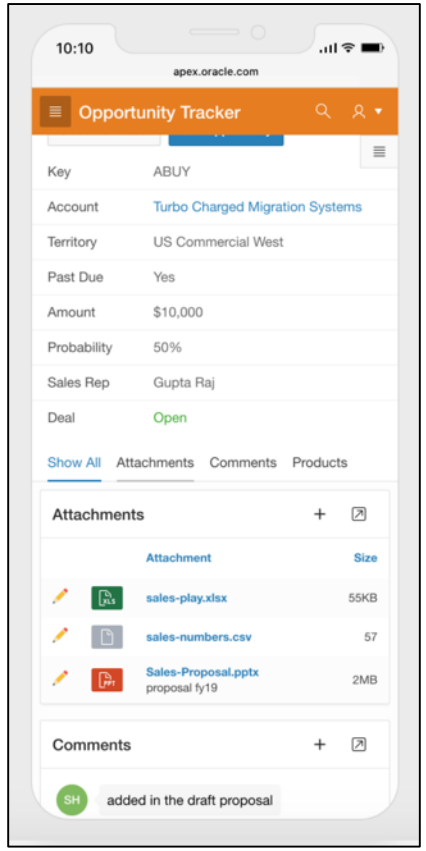
- Default theme for Oracle APEX applications
- Uses responsive HTML5 templates
- Grid-layout for HTML forms
- Modern flat-look
- Ships with a variety of pre-built theme styles

https://apex.oracle.com/ut

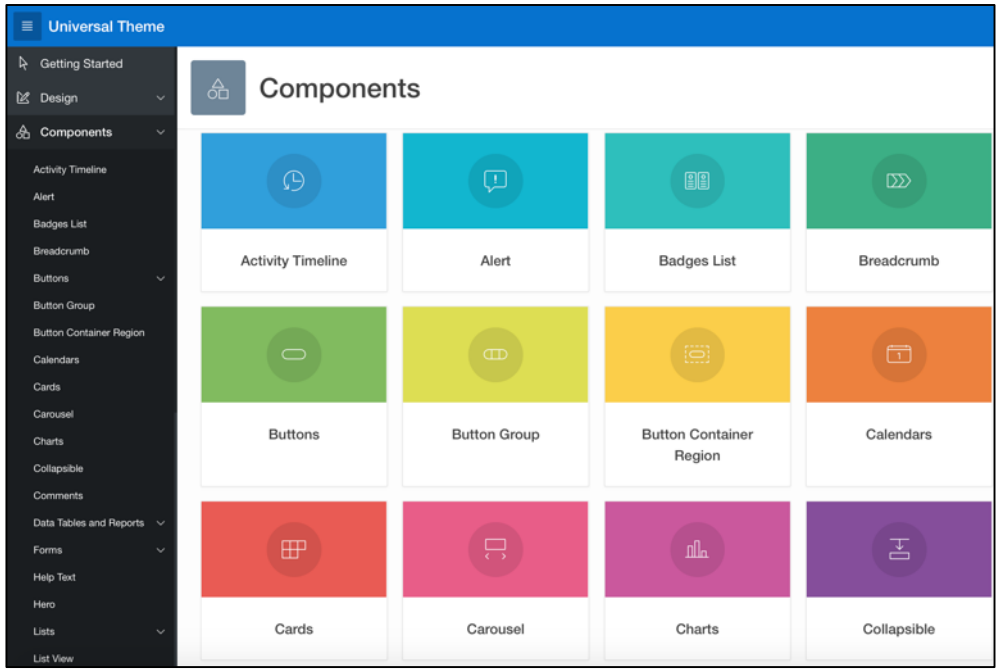
Understanding Universal Theme: Key Features

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

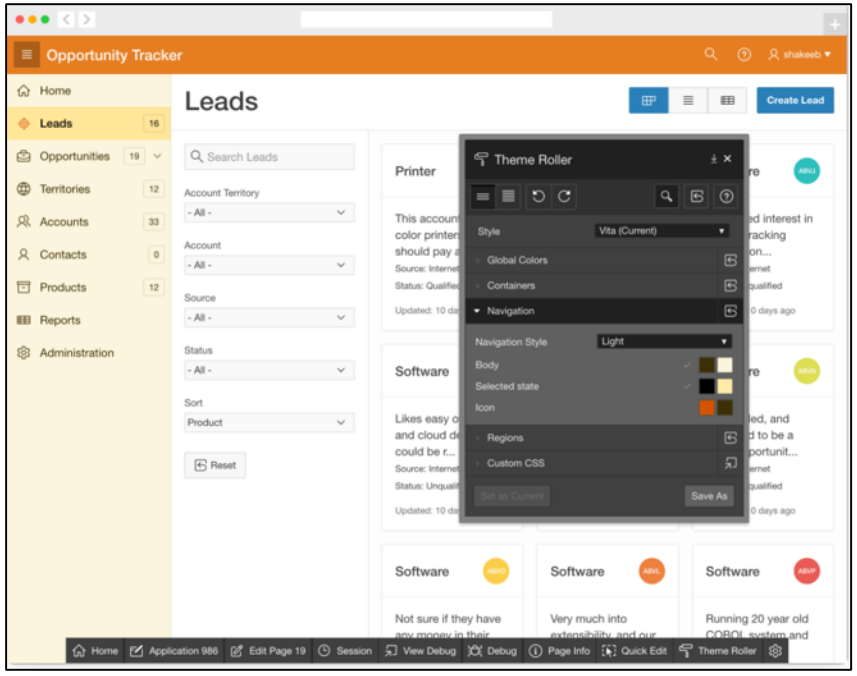
Responsive Design



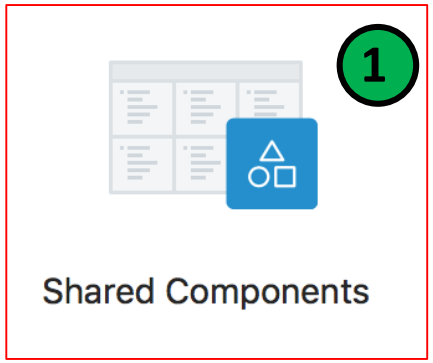
Versatile UI Components

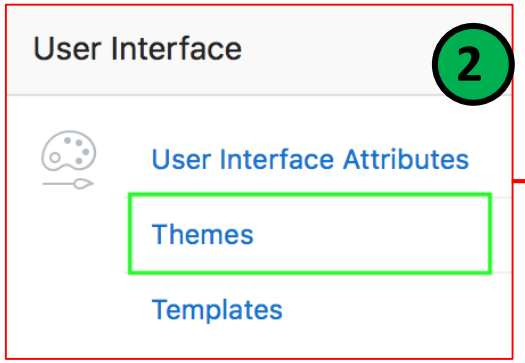


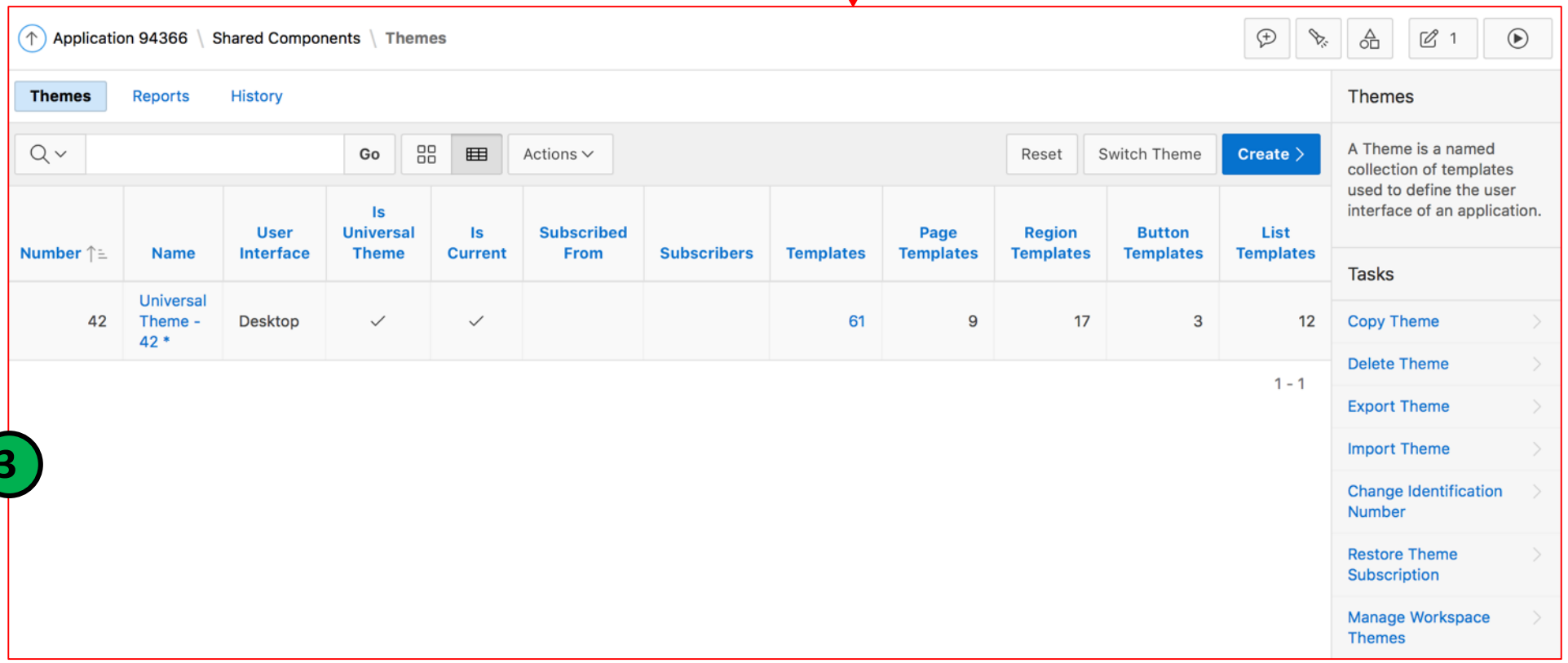
Easy Customization



Accessing the Themes Page



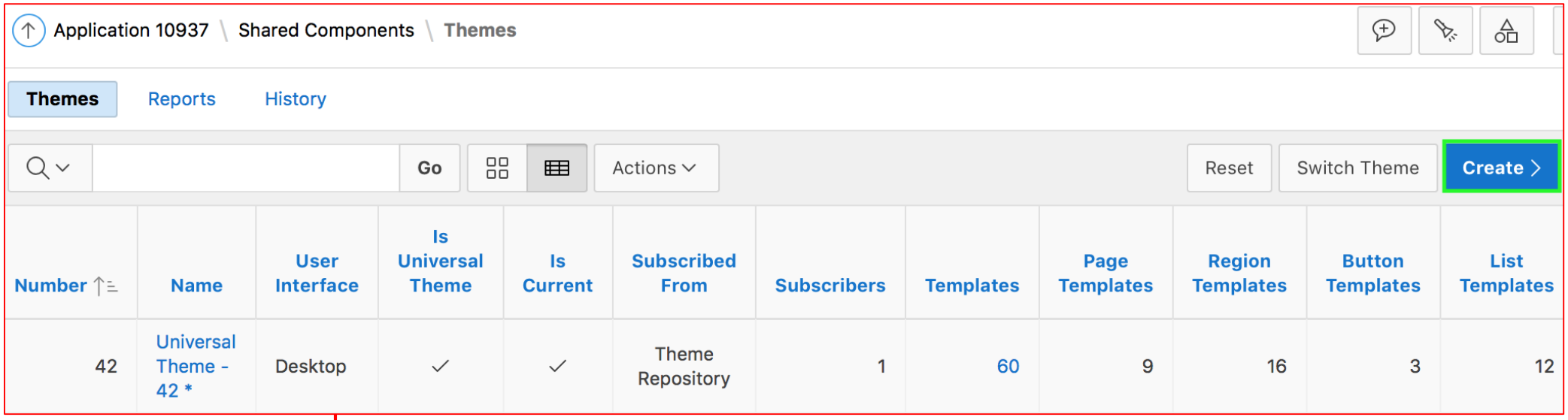


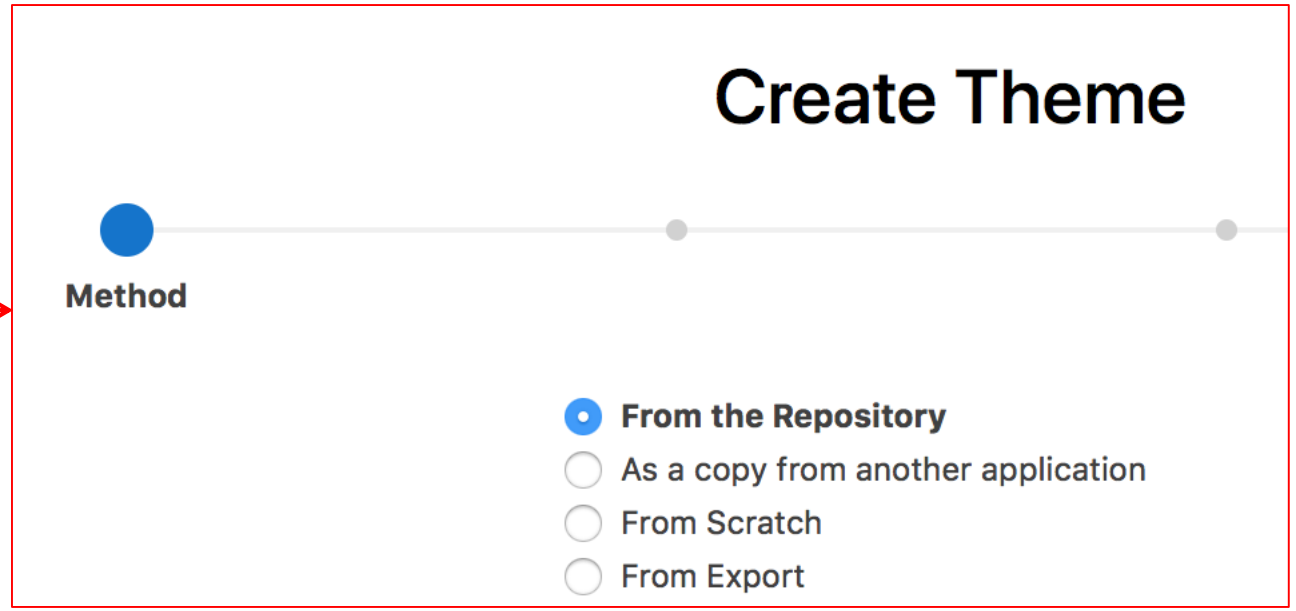


Managing Themes Using Theme Subscriptions

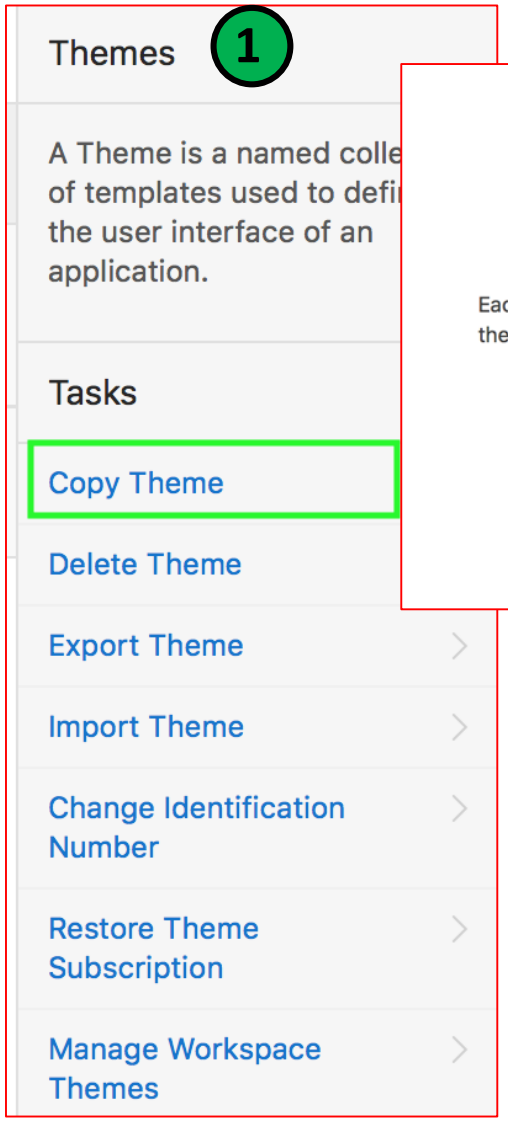
- You can manage themes directly or through theme subscriptions
- Subscribing to a master theme enables the theme to be upgraded during future Oracle APEX releases
- You subscribe to a theme when:
  - ✓Running the Create Application Wizard
  - ✓Creating a new theme from the theme repository
- Once an 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

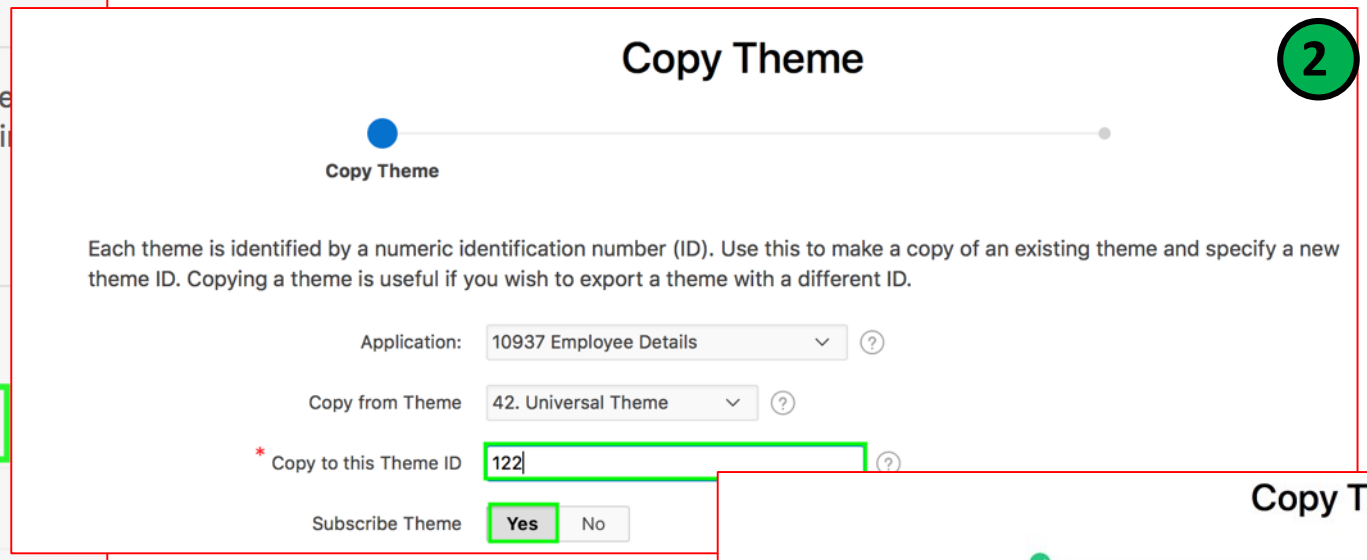
Creating a Theme

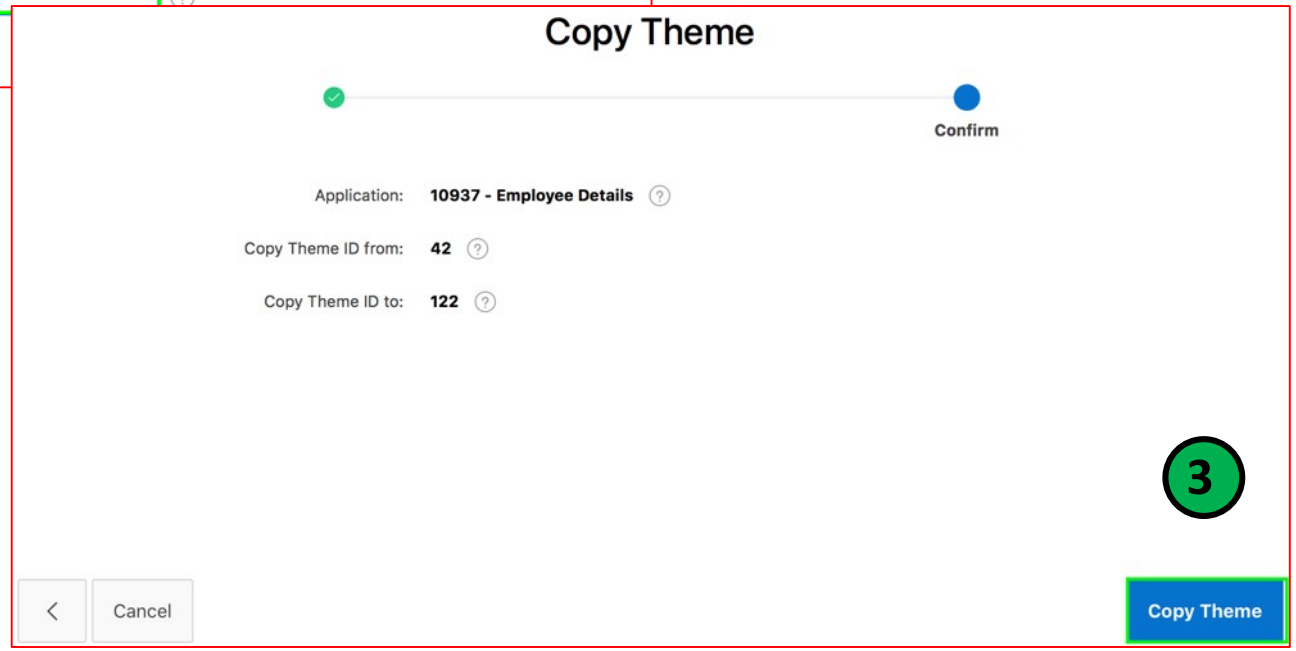




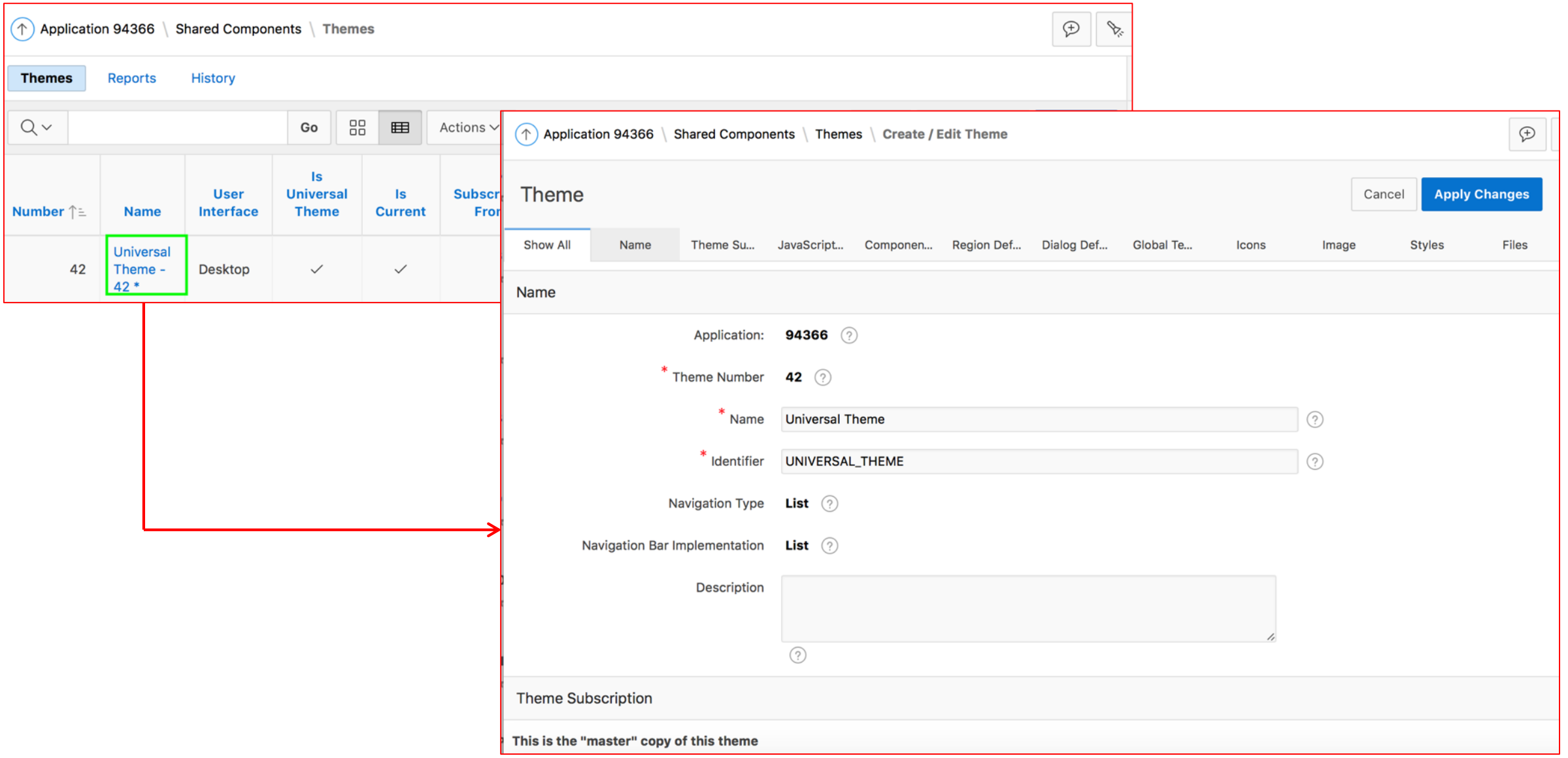
Creating a Copy of an Existing Theme



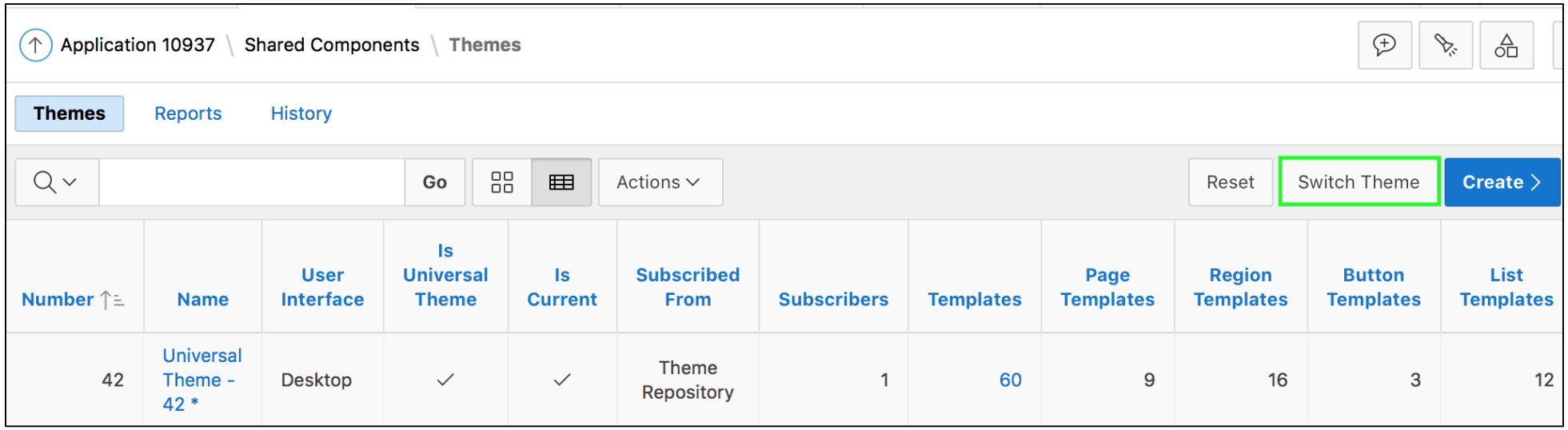




Editing a Theme

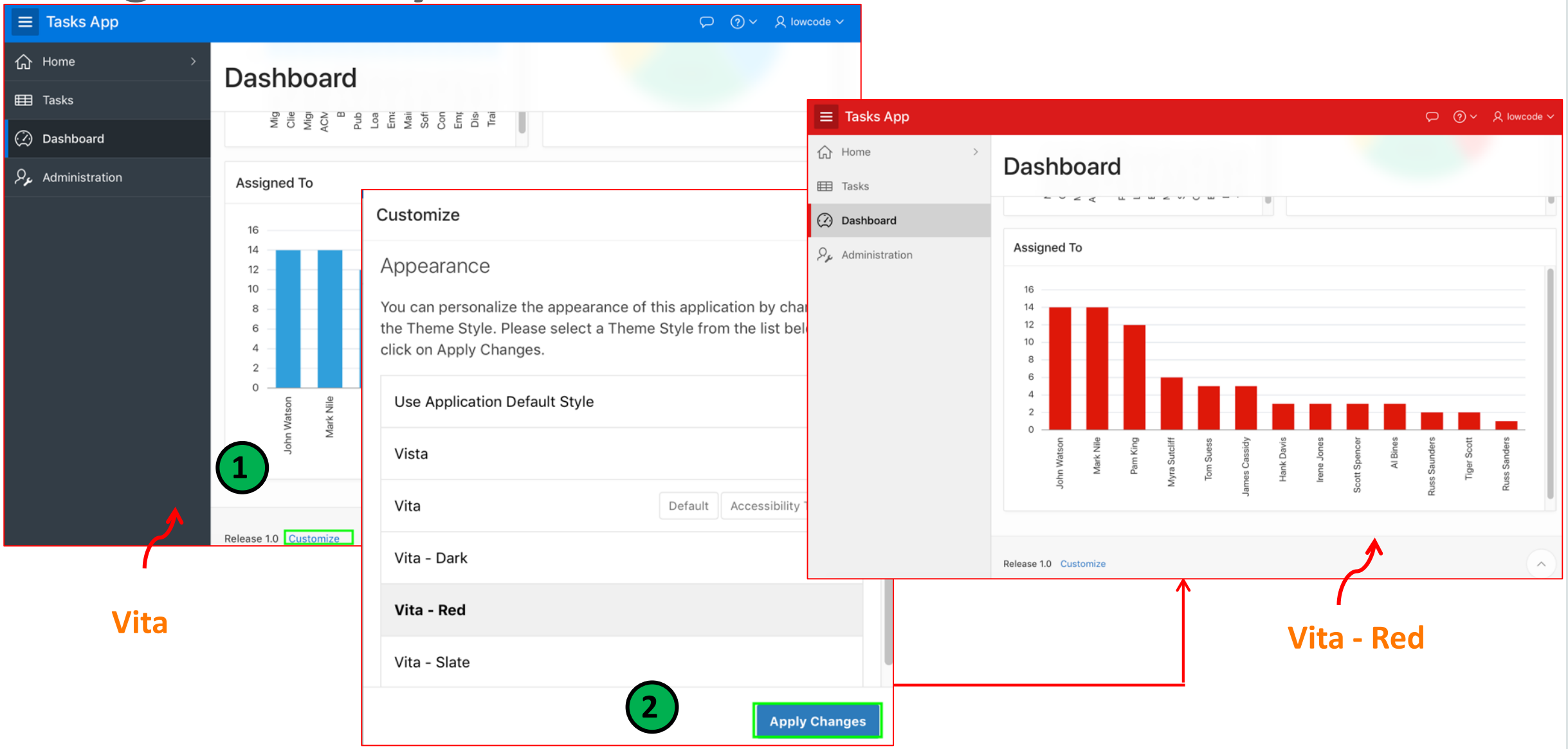


Switching an Active Theme



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

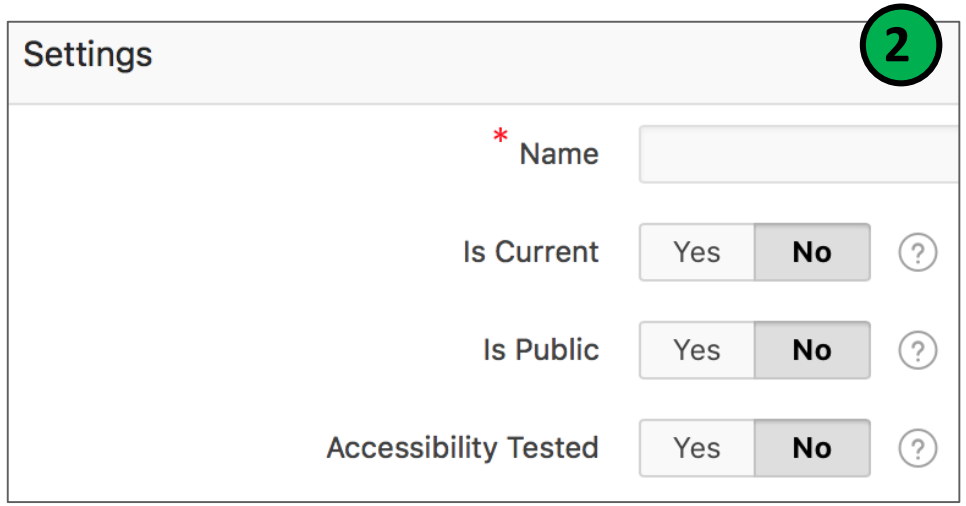
Using Theme Styles

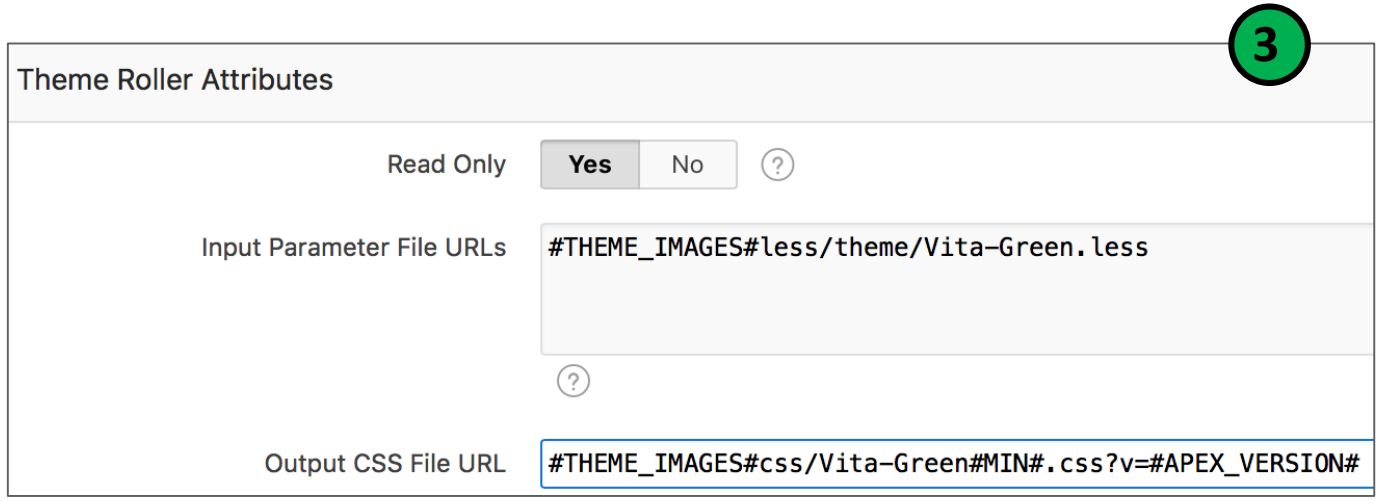


Understanding Theme Styles

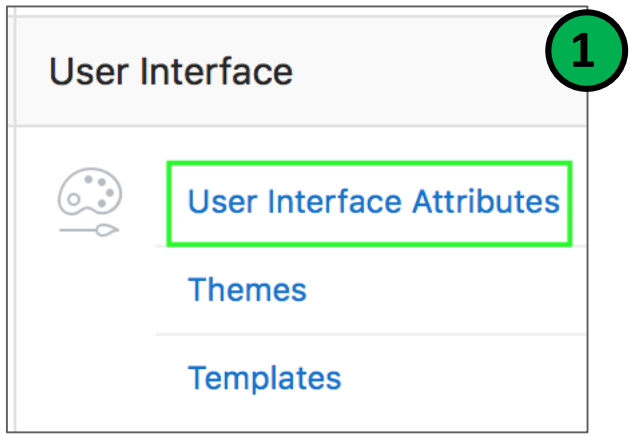
- You use theme styles to customize themes, switch to a different color scheme, apply a flat look, or make a theme responsive
- A theme style defines a CSS style sheet that is added to the base CSS
- Universal Theme includes several pre-built styles
- You can easily generate additional styles by using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

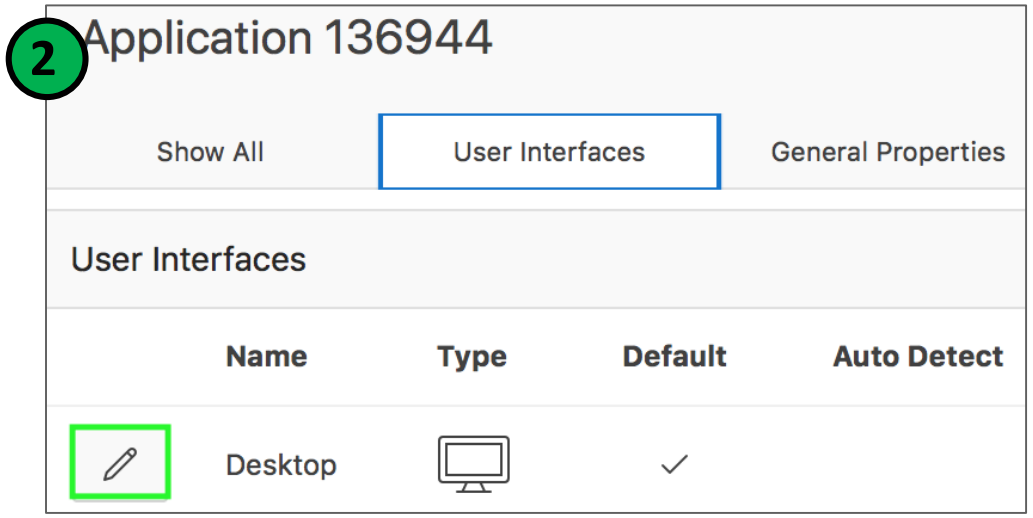
Creating a Theme Style

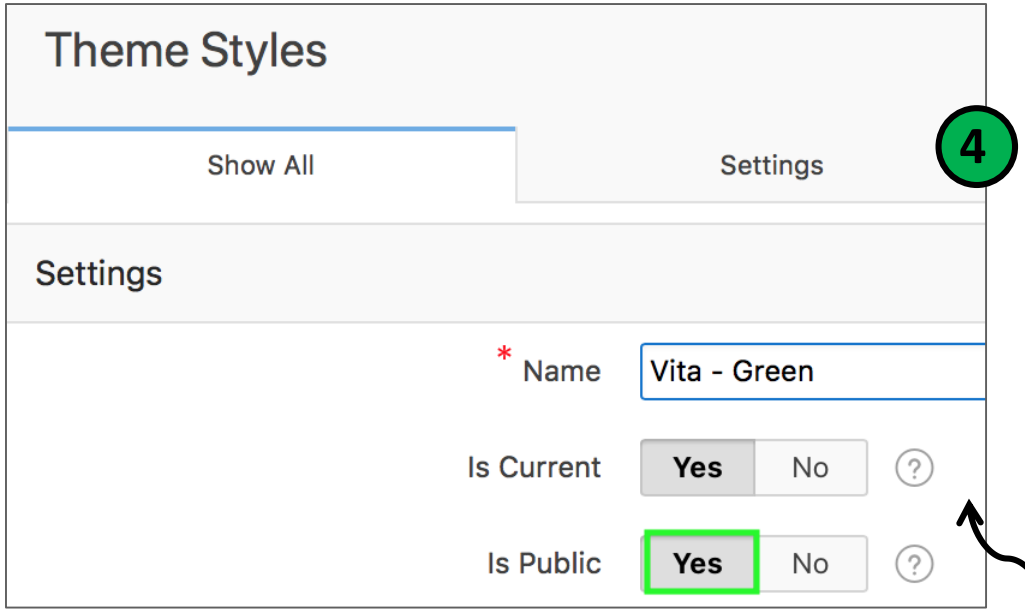


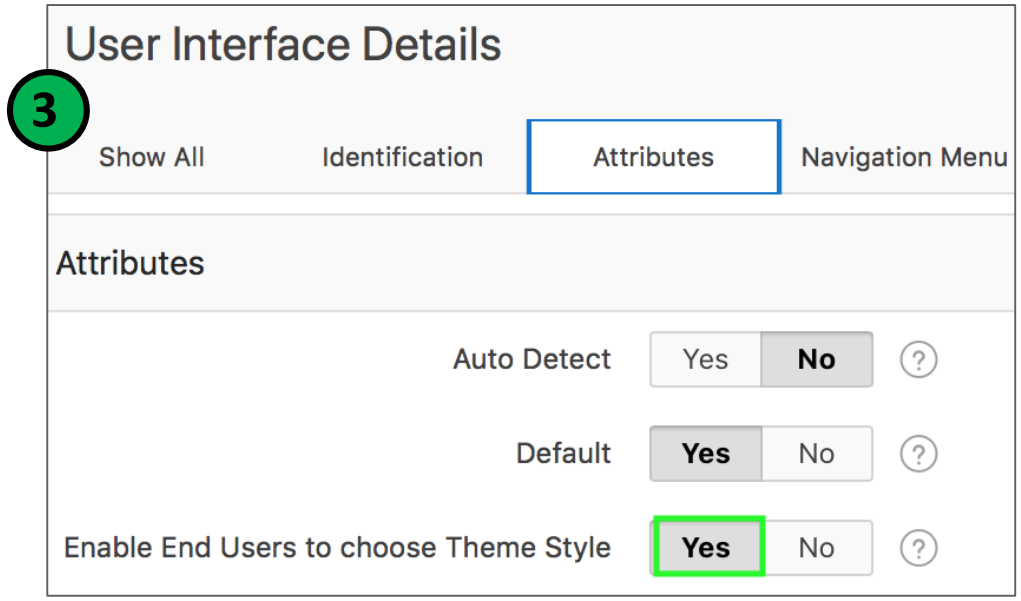


Enabling Users to Select a Theme Style







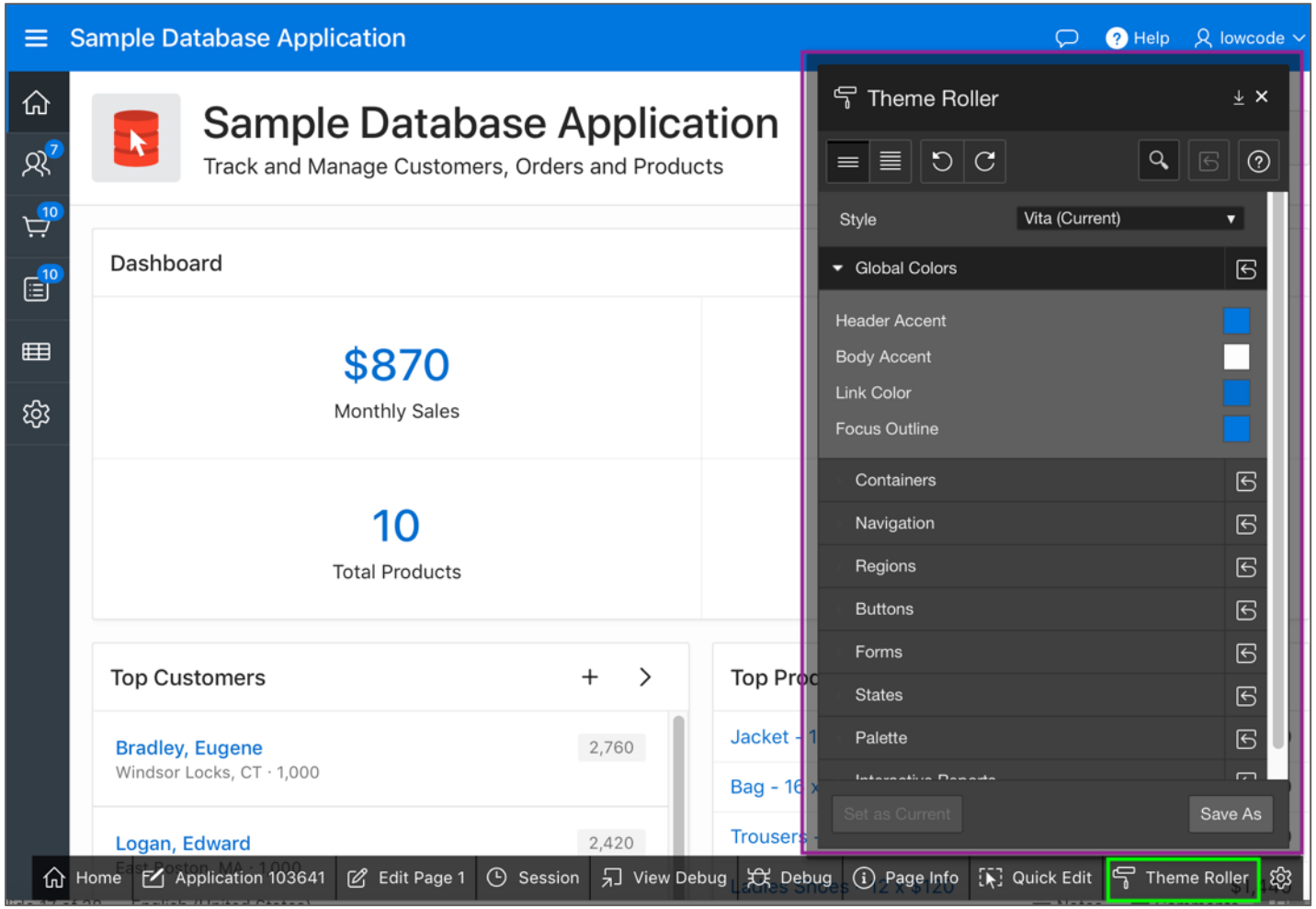


For each theme style, set "Is Public" to "Yes"

Using Theme Roller

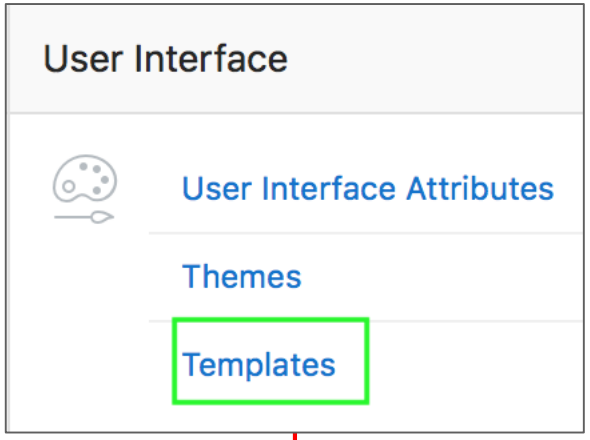
Theme Roller:

- 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



Application 103641 \ Shared Components \ Templates

Templates | Subscription | Publish | Utilization | History

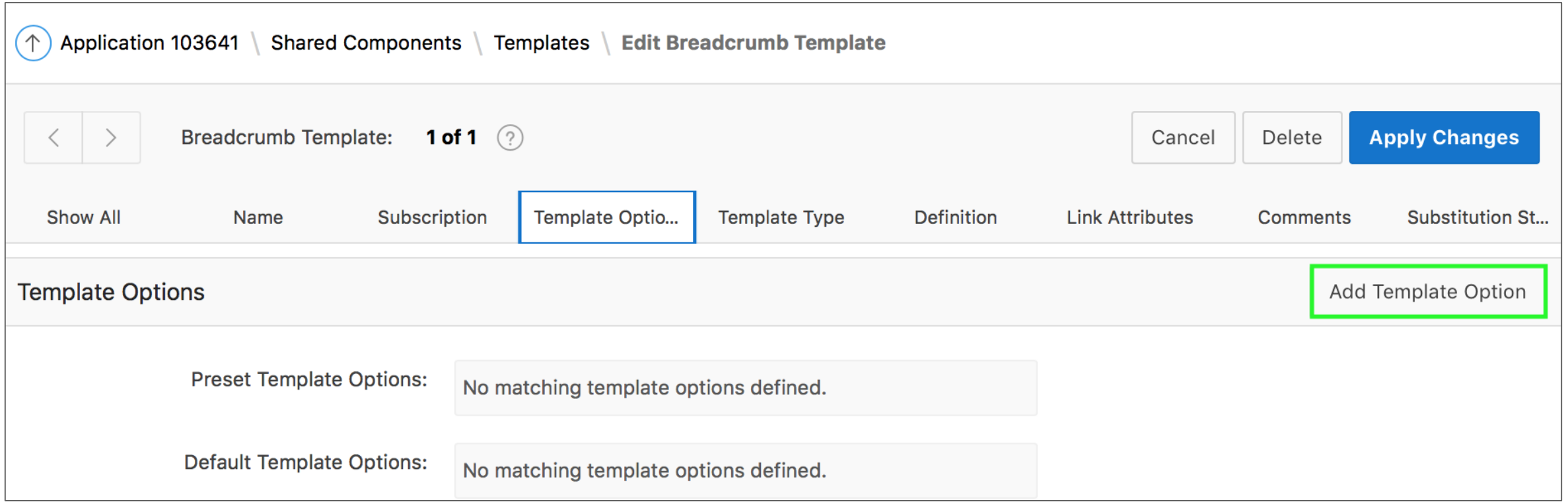
Go | Actions | Reset | Create

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	

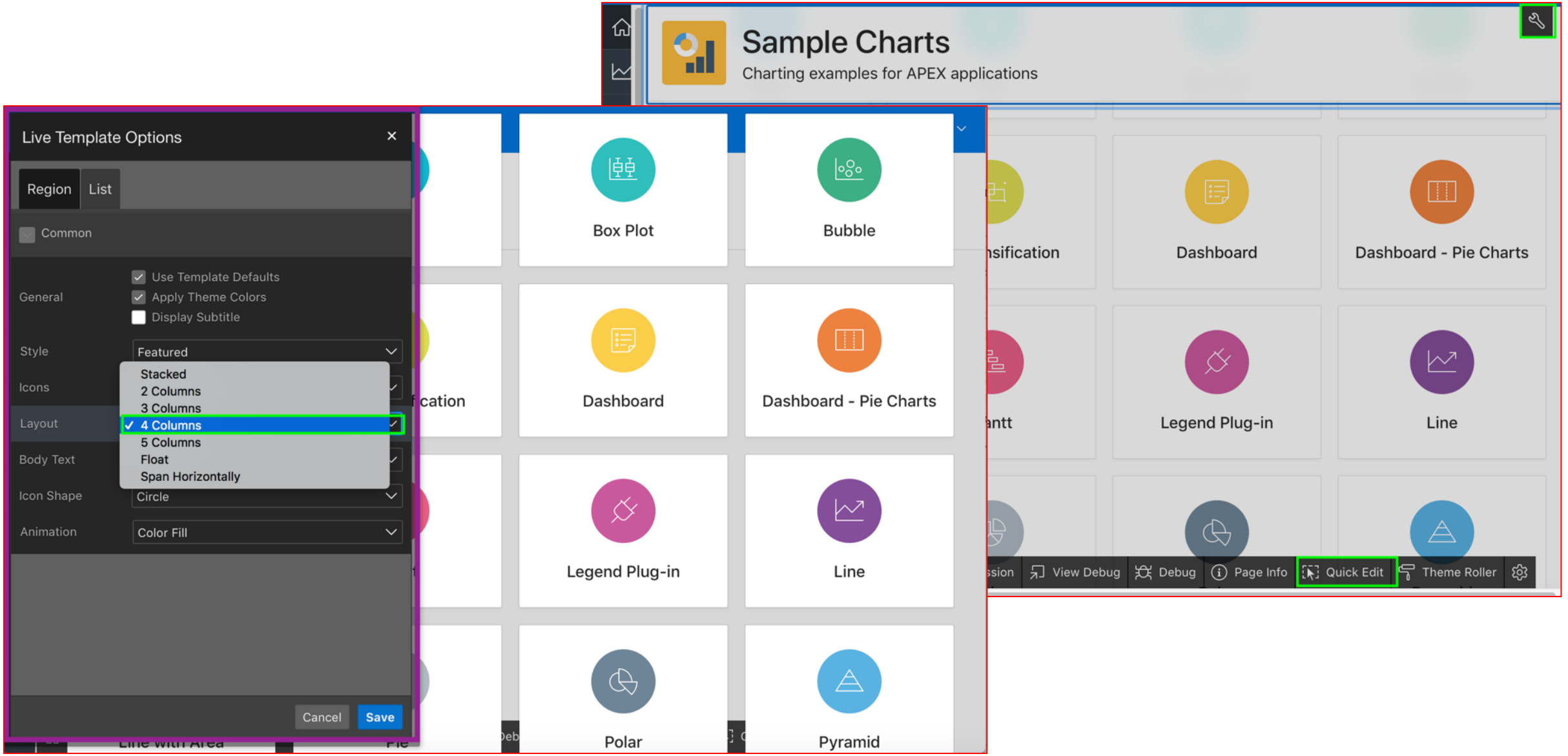
Understanding Template Options

- 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

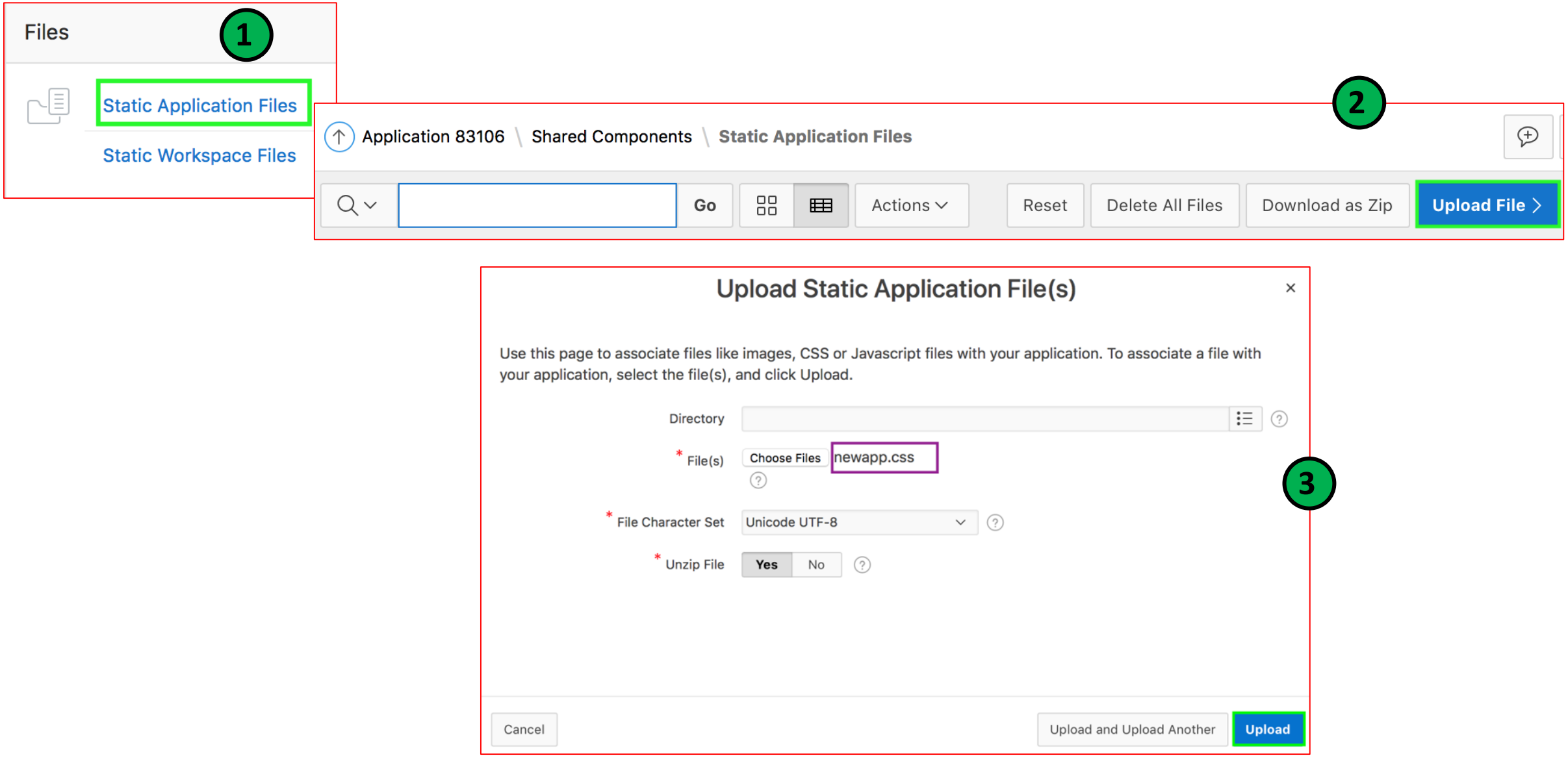
Creating Template Options



Using Live Template Options



Using Custom Cascading Style Sheets



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

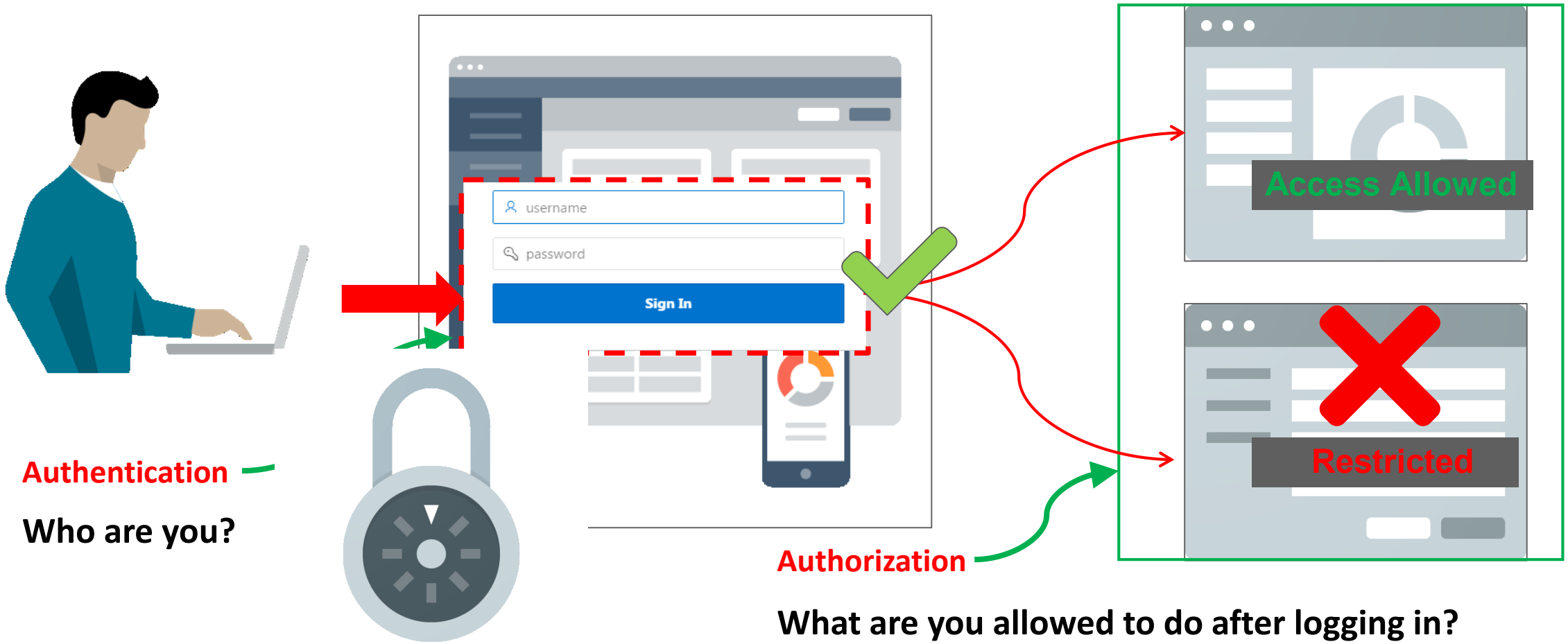
Unit 13: Implementing Security in your Application

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

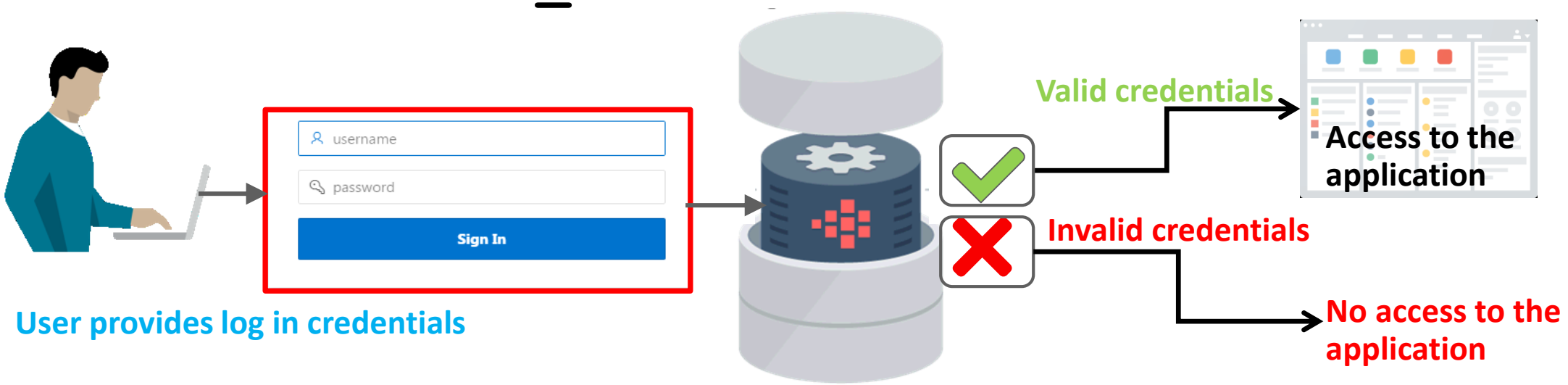
Who are you?

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 requires 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:

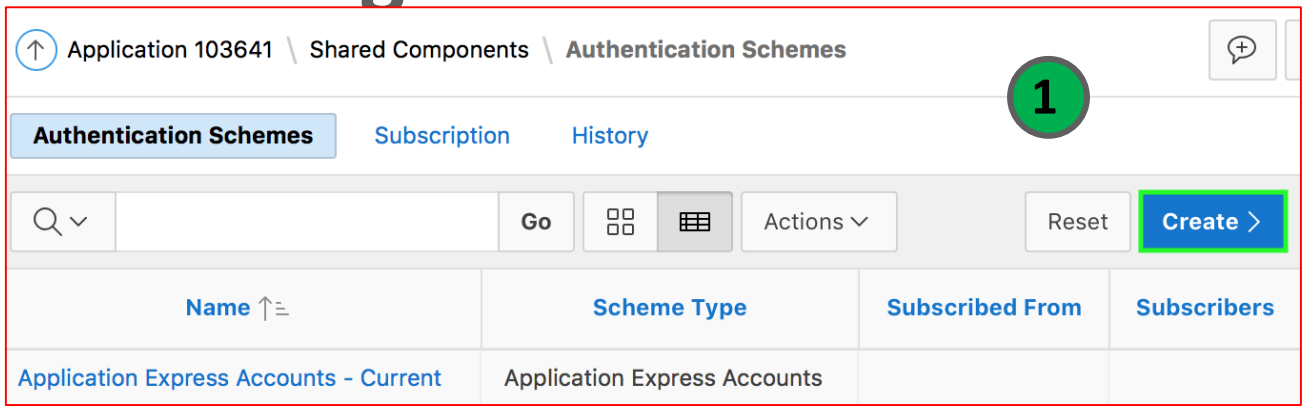
- Selecting a built-in authentication scheme
- Creating a custom authentication scheme

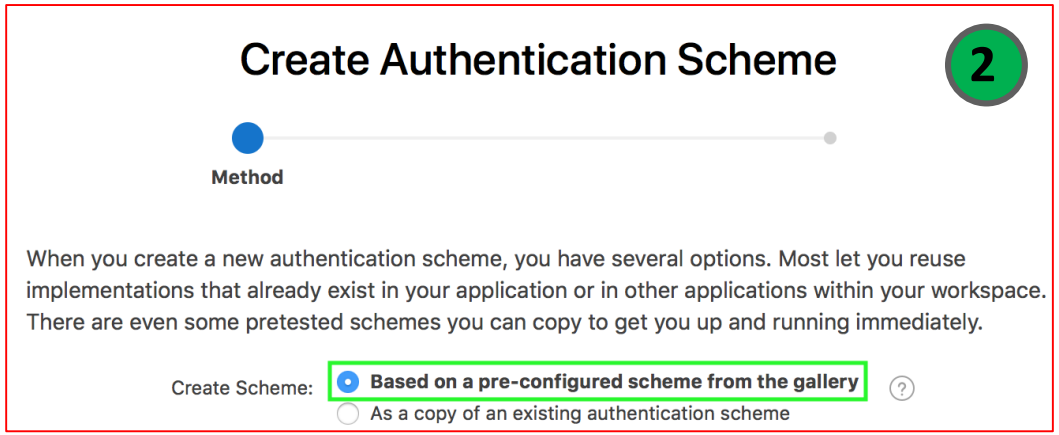
Understanding Preconfigured Authentication Schemes

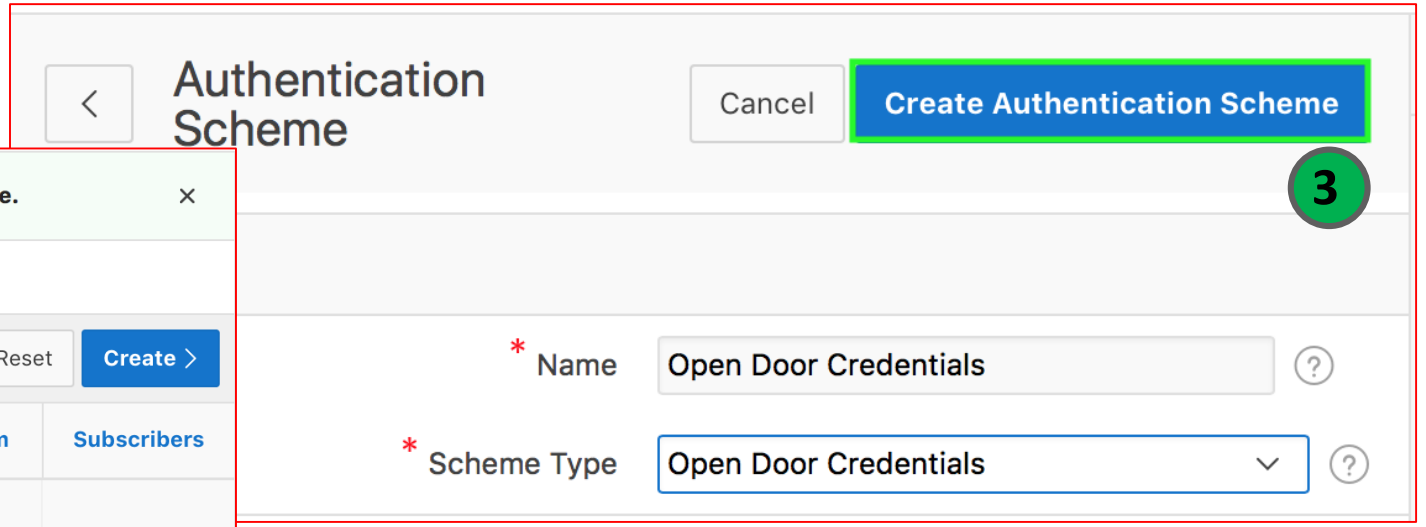
Oracle Application Express supports the following preconfigured authentication schemes:

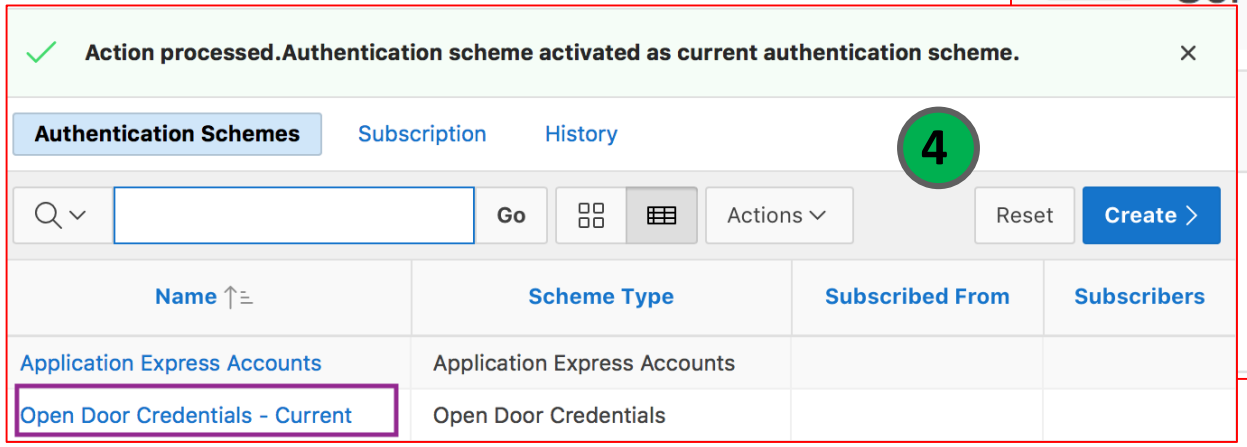
- Open Door Credentials
- Application Express Accounts
- Database Accounts
- LDAP Directory
- No Authentication (Using DAD)
- Oracle Application Server Single Sign-On
- Custom
- HTTP Header Variable
- Social Sign-In

Creating an Authentication Scheme Based on a Pre-configured Scheme

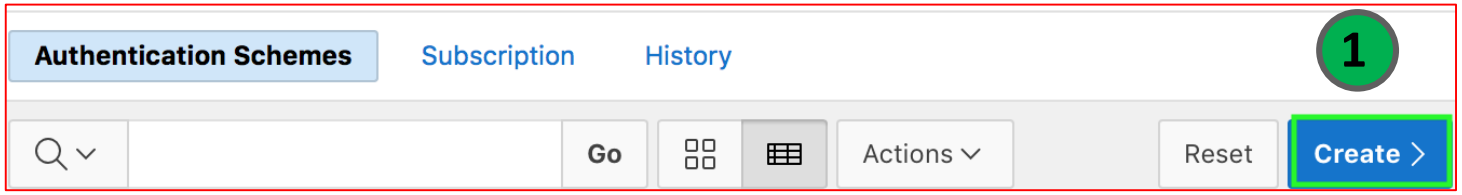


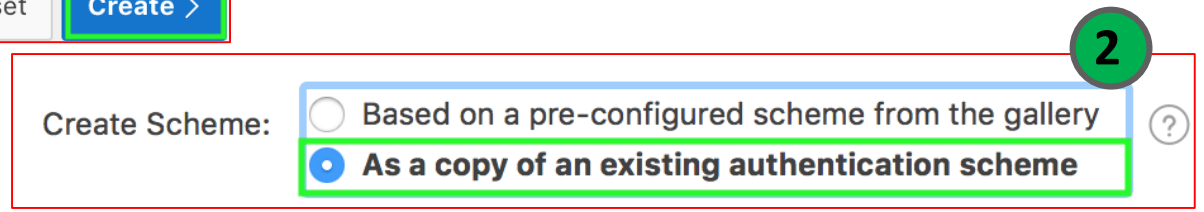


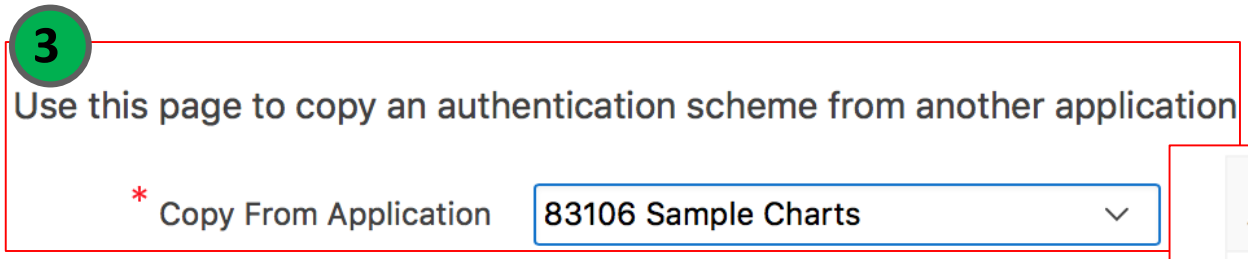


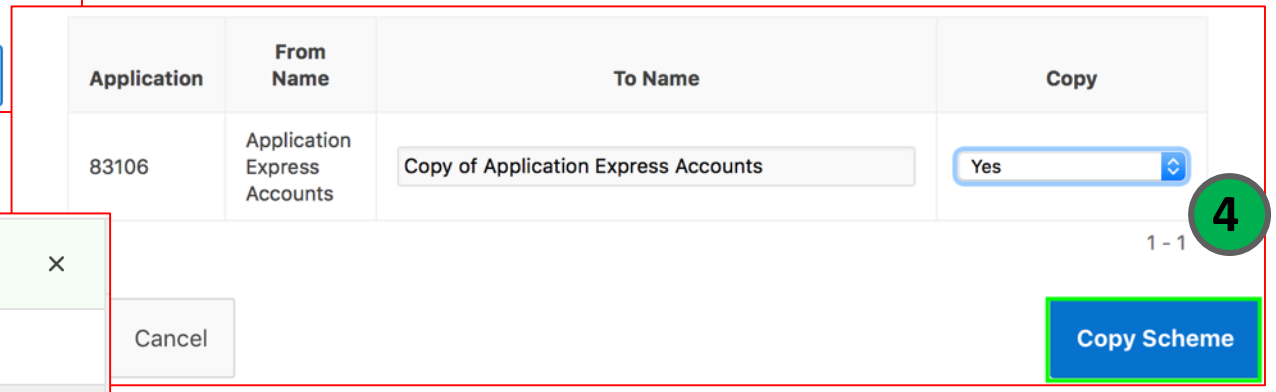


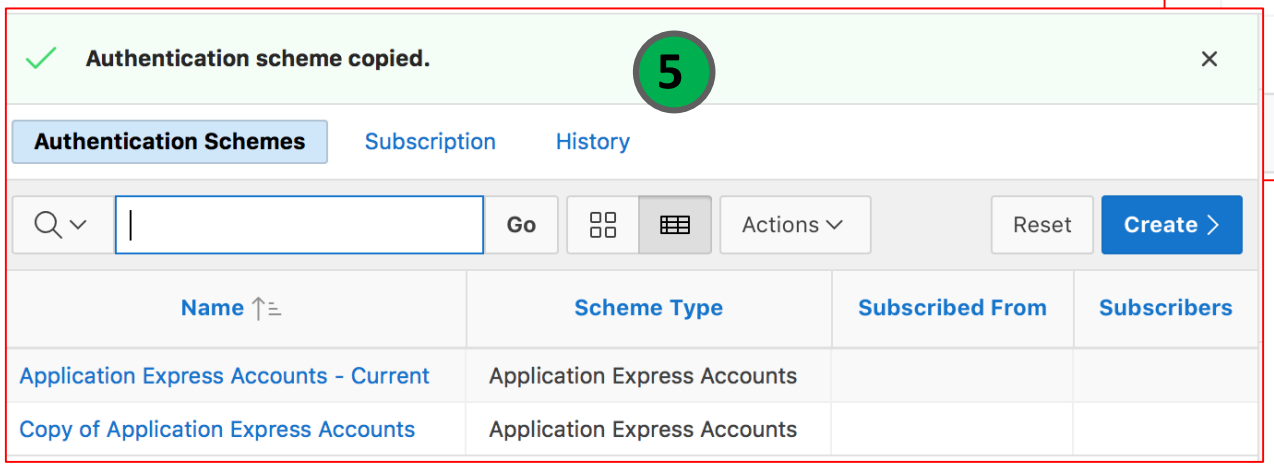
Creating a Copy of an Existing Authentication Scheme











Providing Security Through Authorization

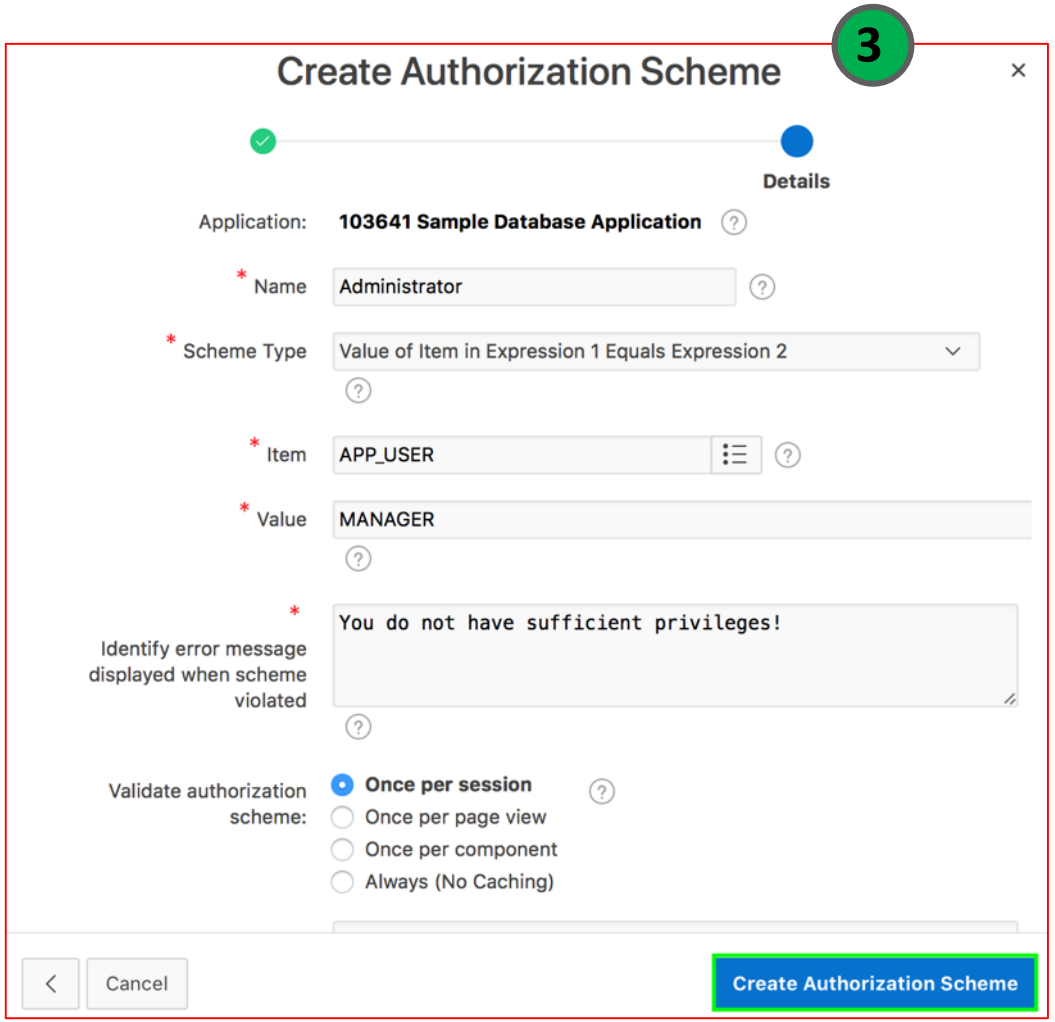
- Authorization involves controlling access to resources based on user privileges
- Use authorization schemes to implement authorization in an application
- You can apply an authorization scheme for:
  - An entire application
  - A page
  - Specific control such as a region, item, or button

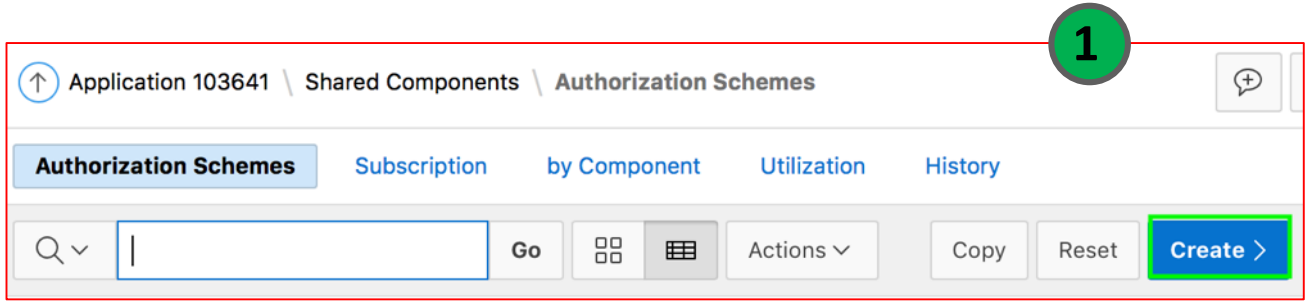
How to Implement Authorization?

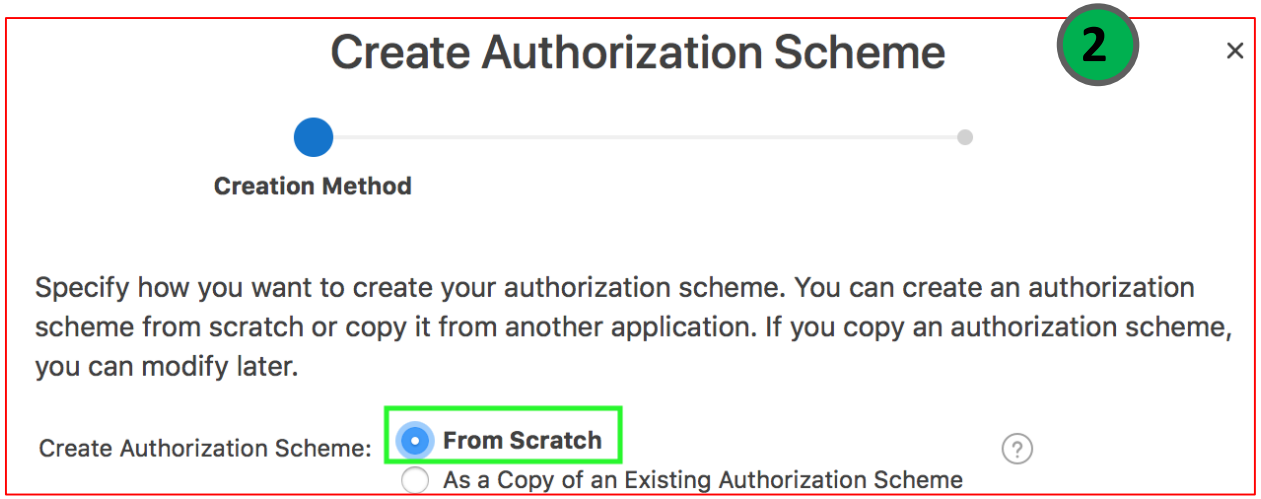
Oracle Application Express allows you to implement authorization to an application or its components in two different ways:

- 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



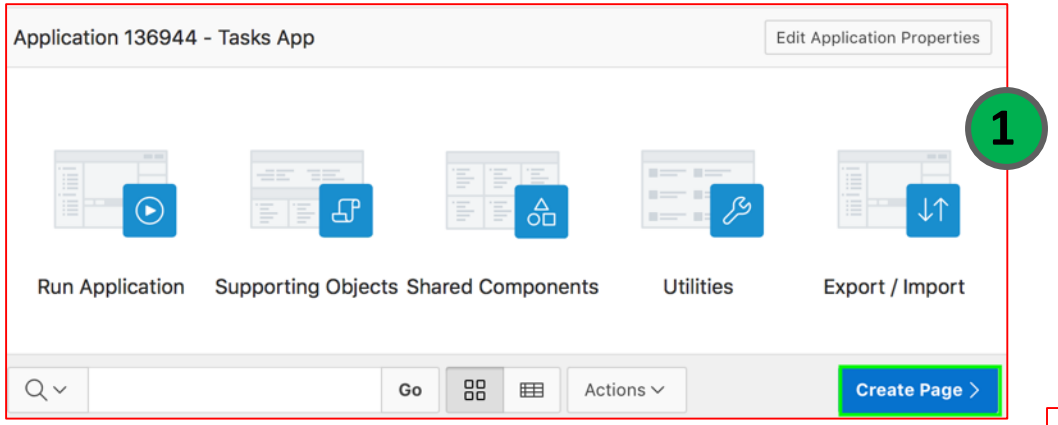


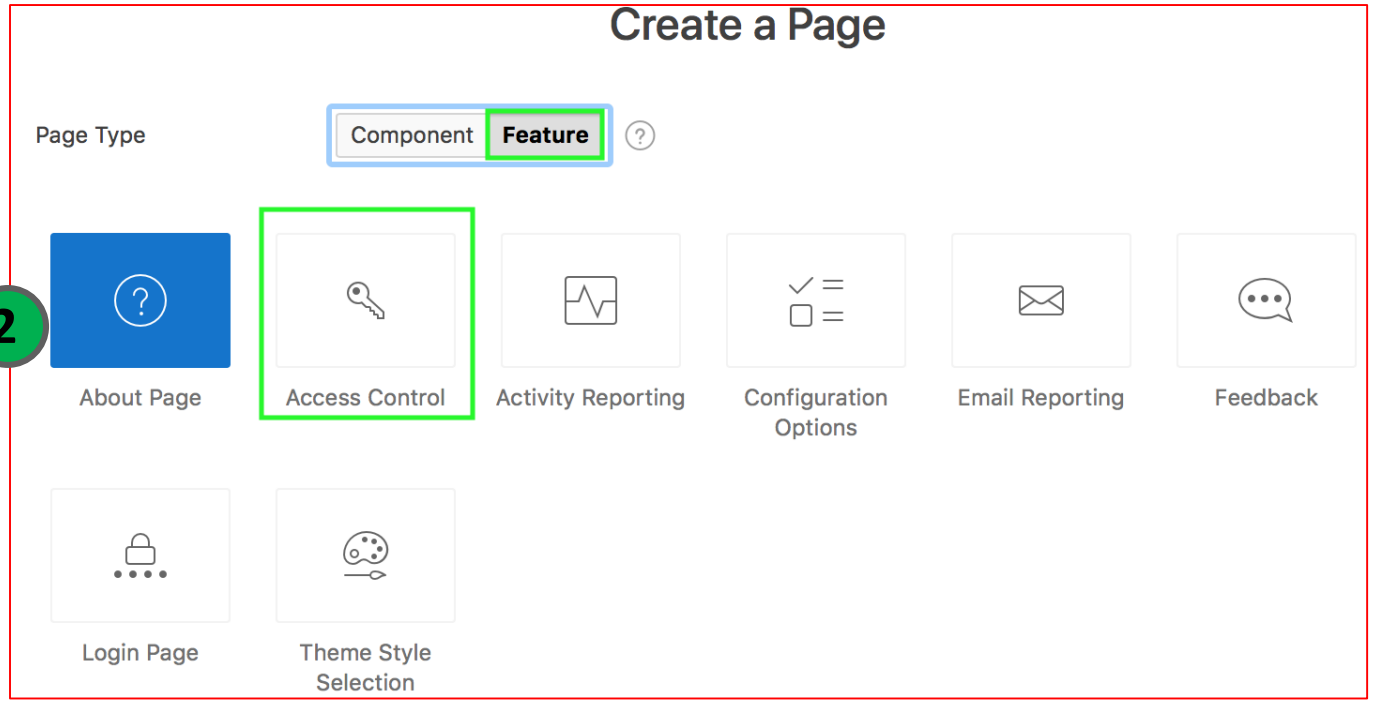


Understanding Access Control

- 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





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: 1001
Page Group: - Select Page Group -
Build Option: **Feature: Access Control**
* Administration Page Preference: (•) **Create a new page** ( ) Identify an existing page
* Administration Page Number: 1000
* Administration Page Name: Administration
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: Administration
Parent Navigation Menu Entry: - No parent selected - / Home

3

< Cancel

Create Access Control Pages - Confirmation

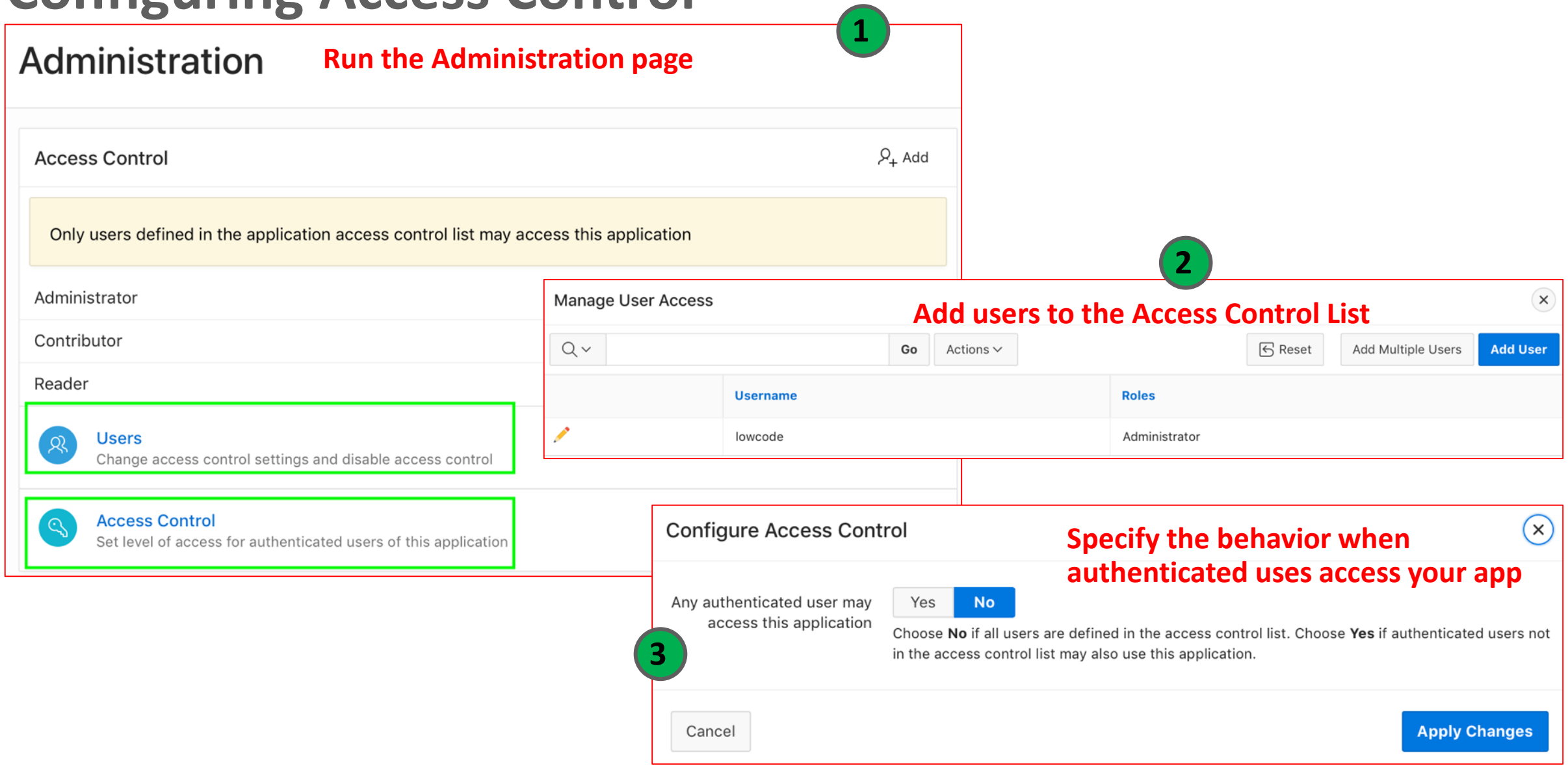
4

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

< Cancel **Create**

Configuring Access Control



Attaching an Authorization Scheme to an Application

1

Shared Components

Security

2

Security Attributes

Authentication Schemes

Authorization Schemes

Application Access Control

Session State Protection

Web Credentials

3

Definition | **Security** | Globalization | User Interface

Application 53503

Cancel | **Apply Changes**

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

Authorization — Define Authorization Schemes >

Application authorization schemes control access to all pages within an application. Unauthorized access to the application, regardless of which page is requested, will cause an error page to be displayed.

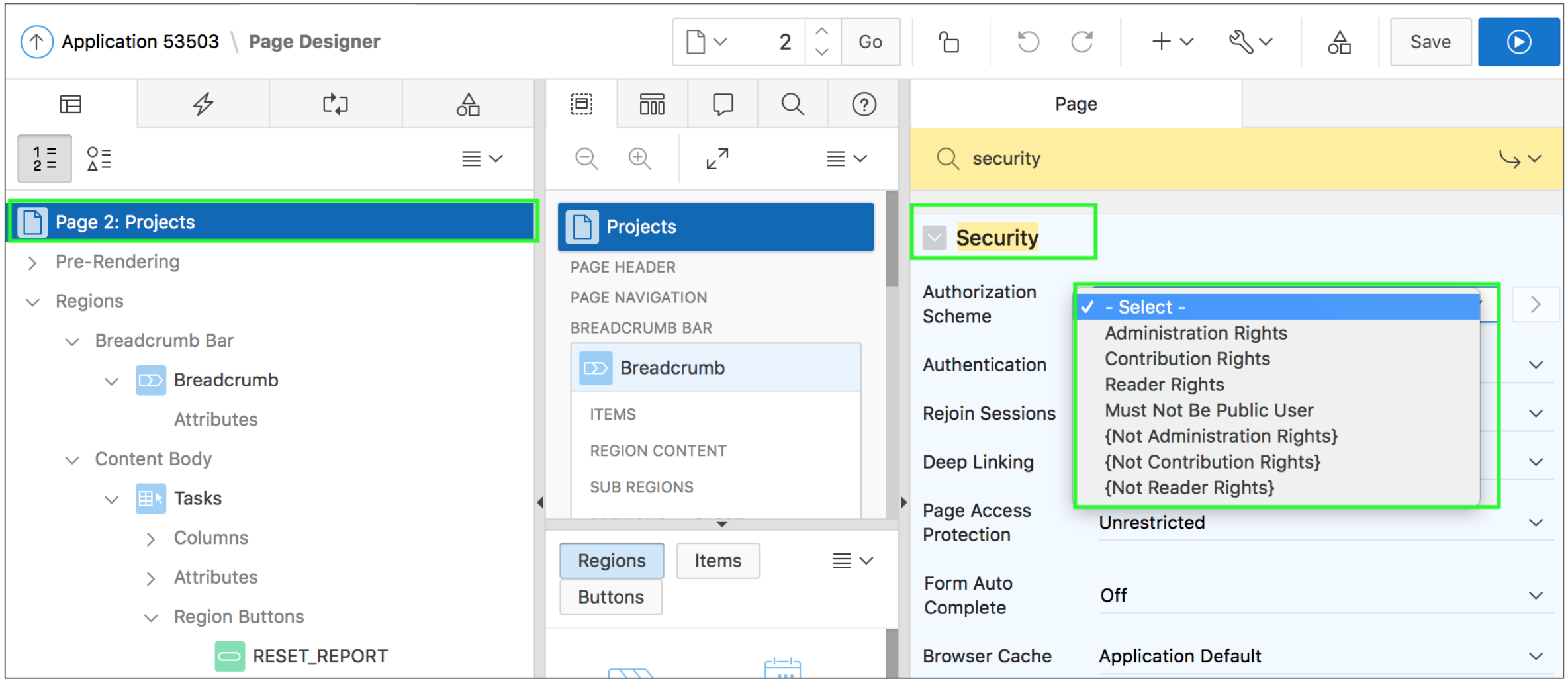
Authorization Scheme: ✓ - No application authorization required -
Administration Rights
Contribution Rights
Must Not Be Public User
Reader Rights
{Not Administration Rights}
{Not Contribution Rights}
{Not Reader Rights}

Run on Public Pages

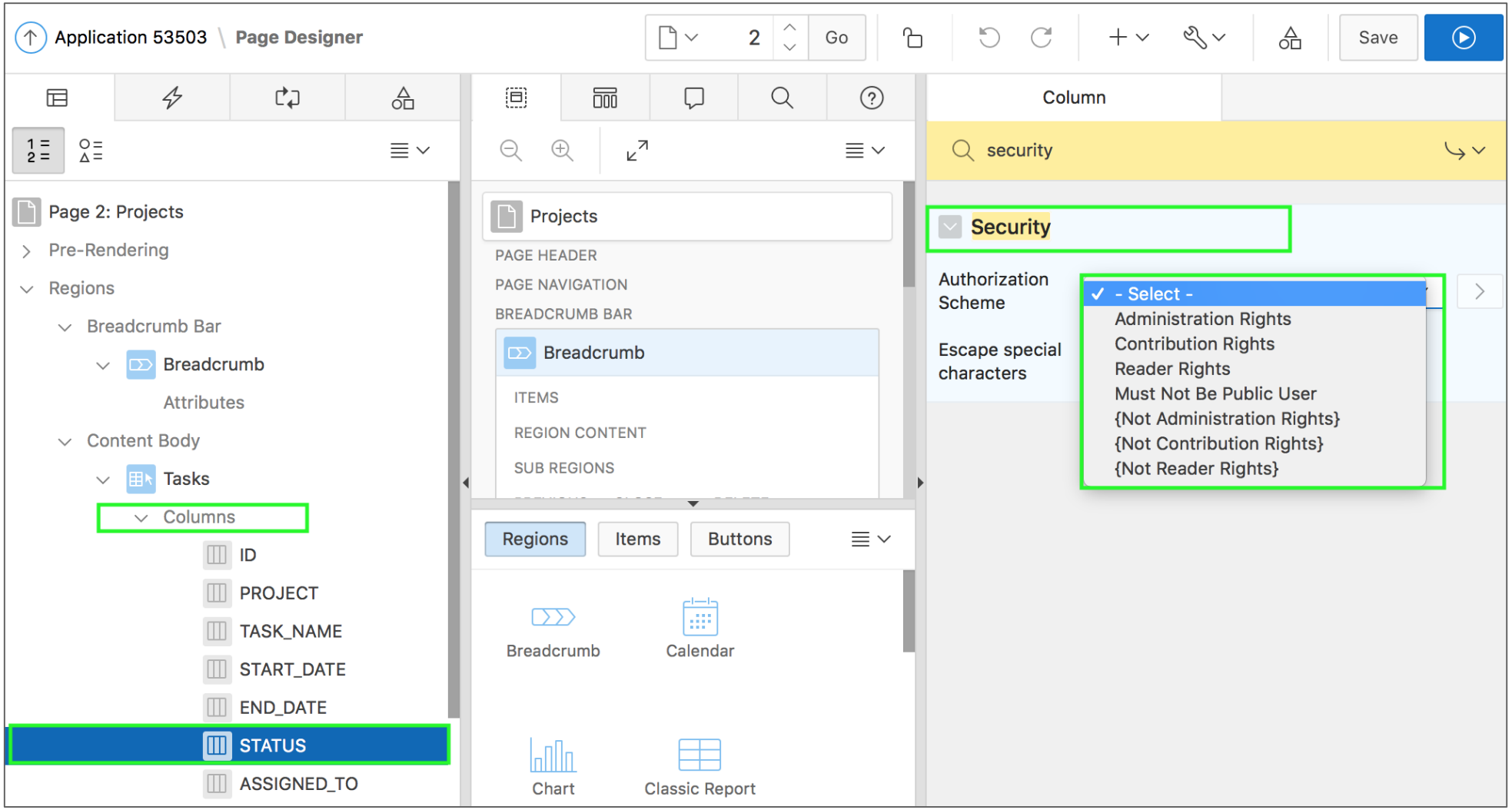
Source for Role or Group Schemes

History

Attaching an Authorization Scheme to a Page



Attaching an Authorization Scheme to a Control or Component



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

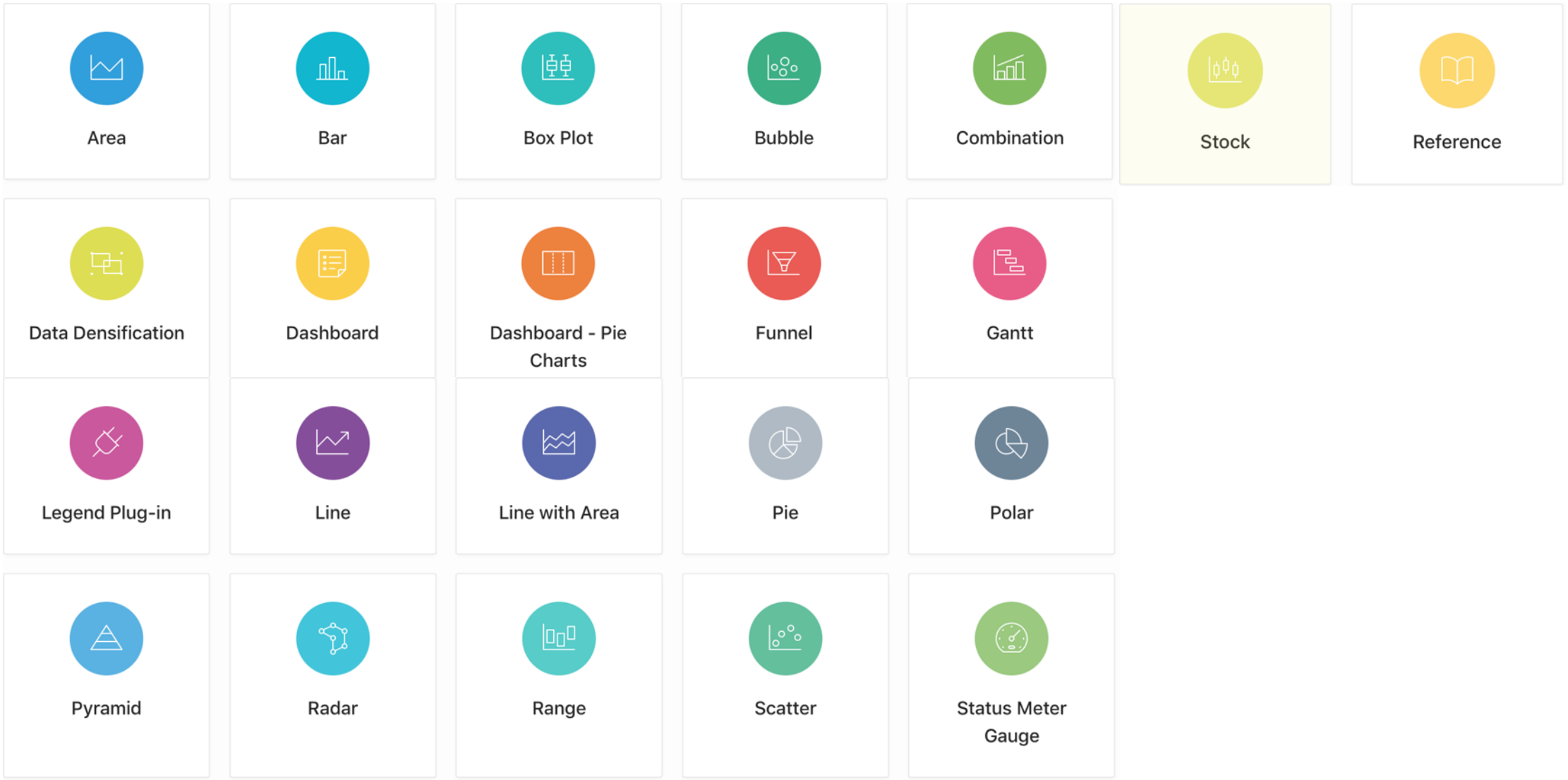
Unit 14: Adding Additional Pages to your Application

Lesson Objectives

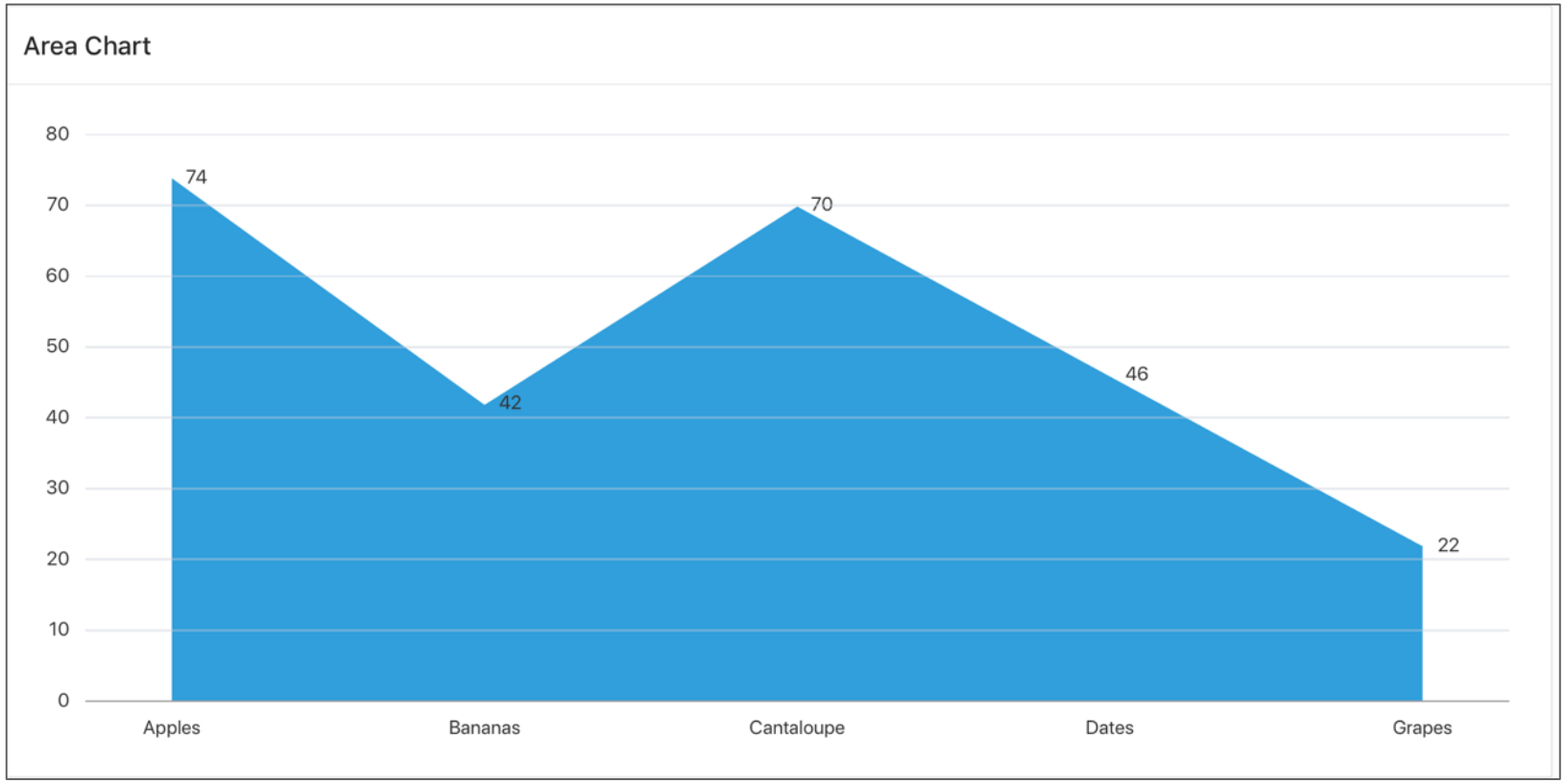
After completing this lesson, you should be able to create, customize, and use the following in your application:

- Oracle JET Charts
- Calendars
- Trees

Creating Charts



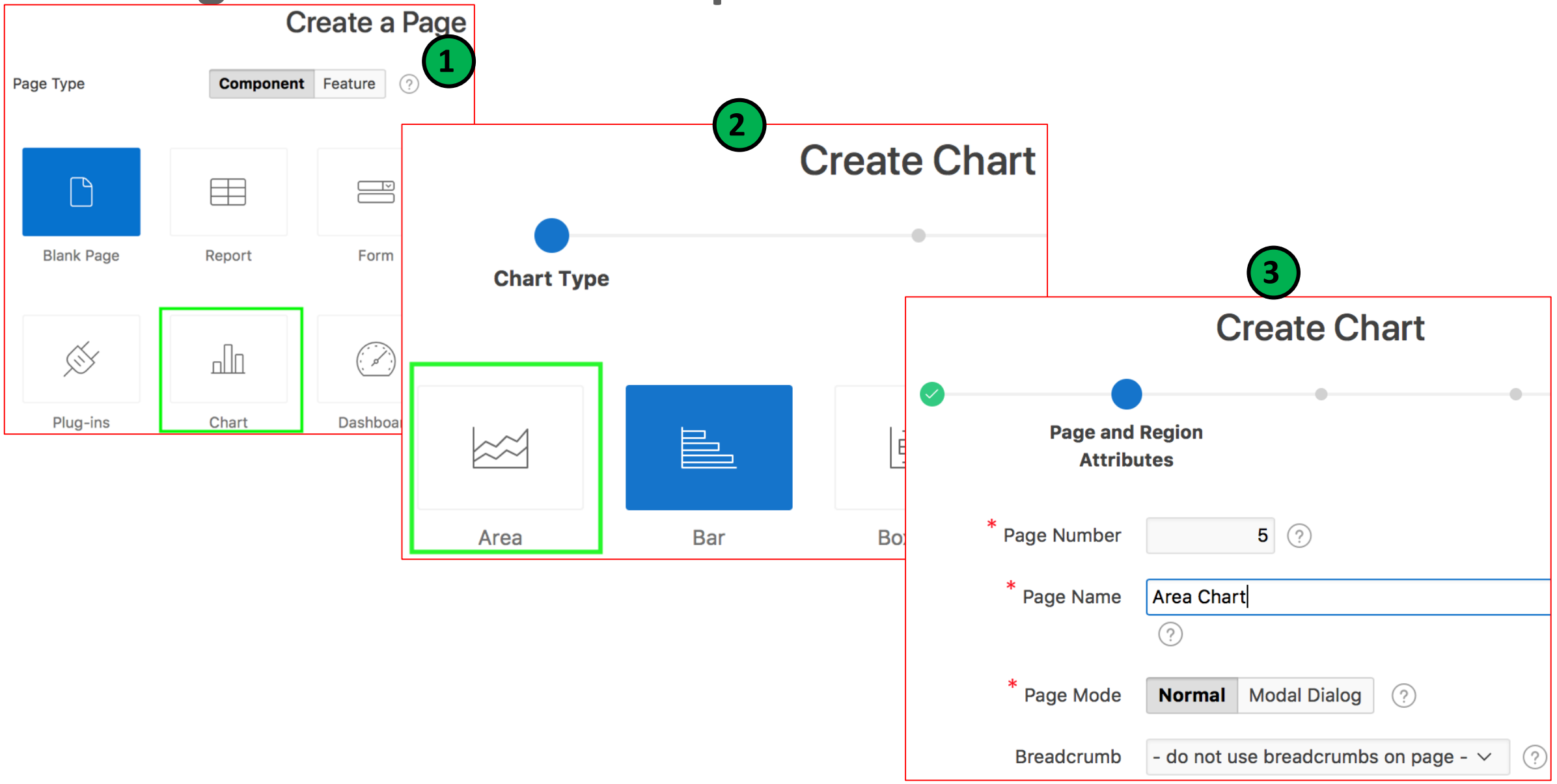
Creating Charts: Area Chart



SQL query used for the chart series

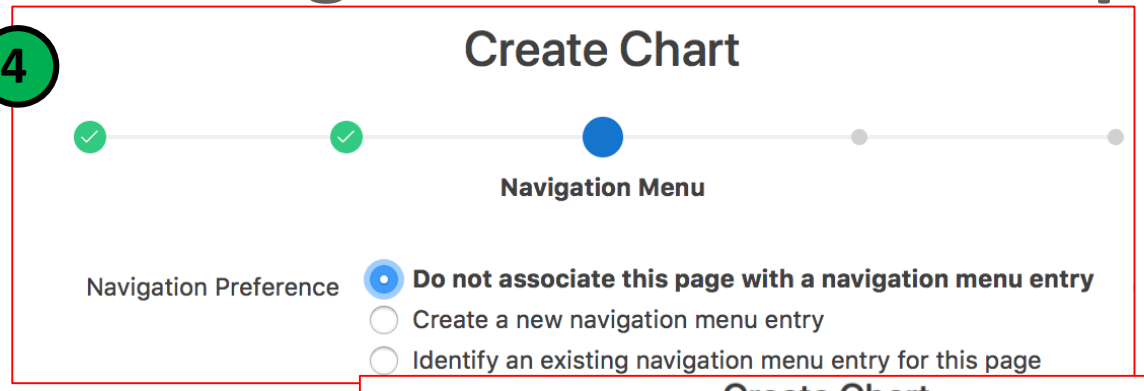
```
select a.product_name, b.quantity, b.customer, a.product_description
from eba_demo_chart_products a, eba_demo_chart_orders b
where a.product_id = b.product_id
and b.customer = 'Deli'
```

Creating an Area Chart: Steps

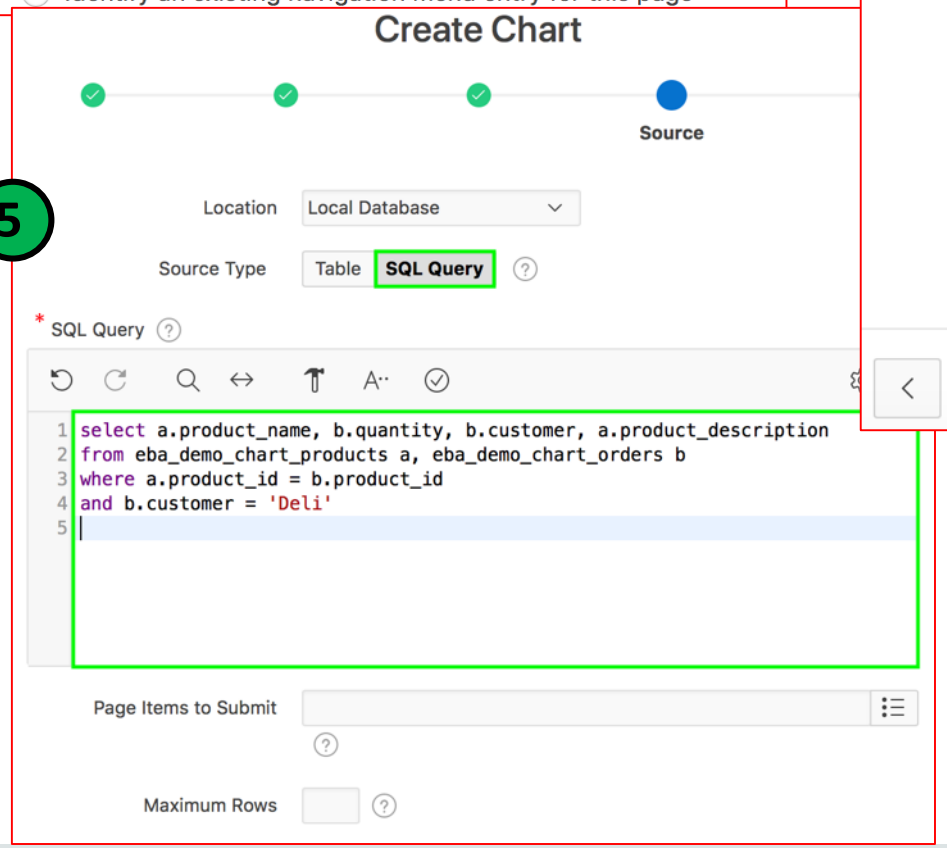


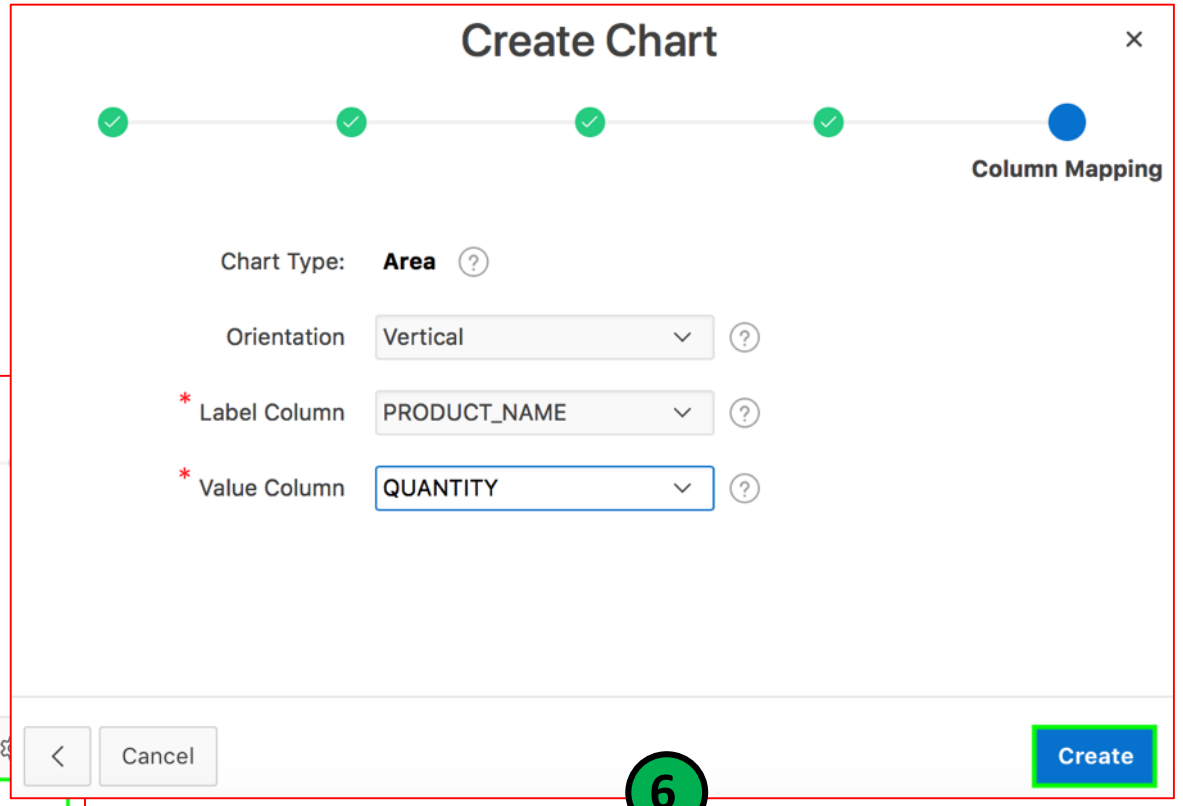
Creating an Area Chart: Steps

4



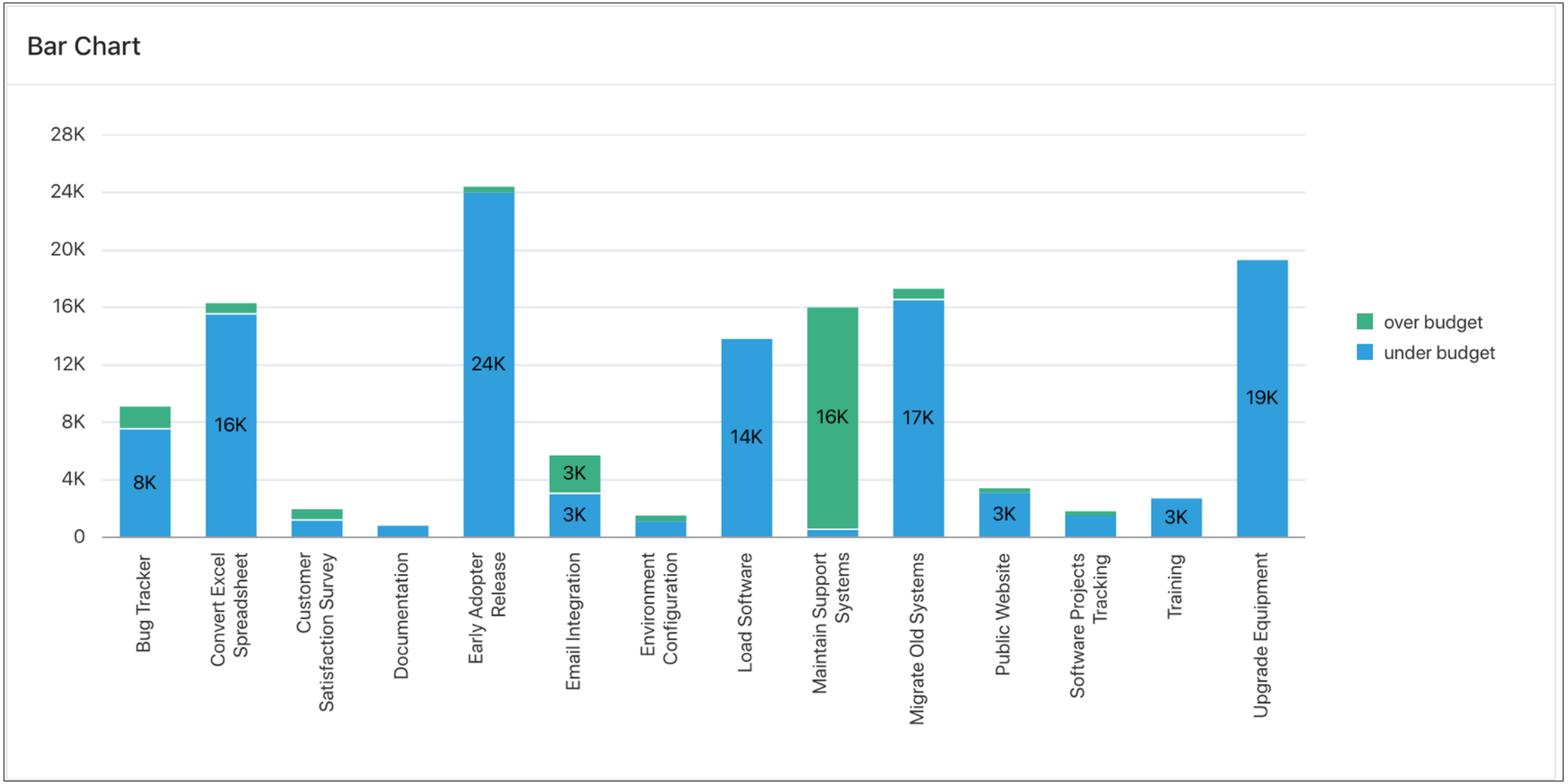
5



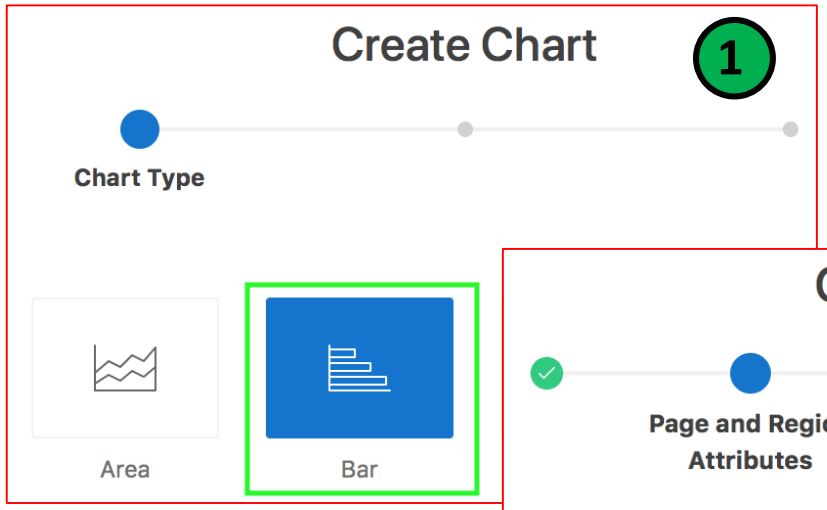


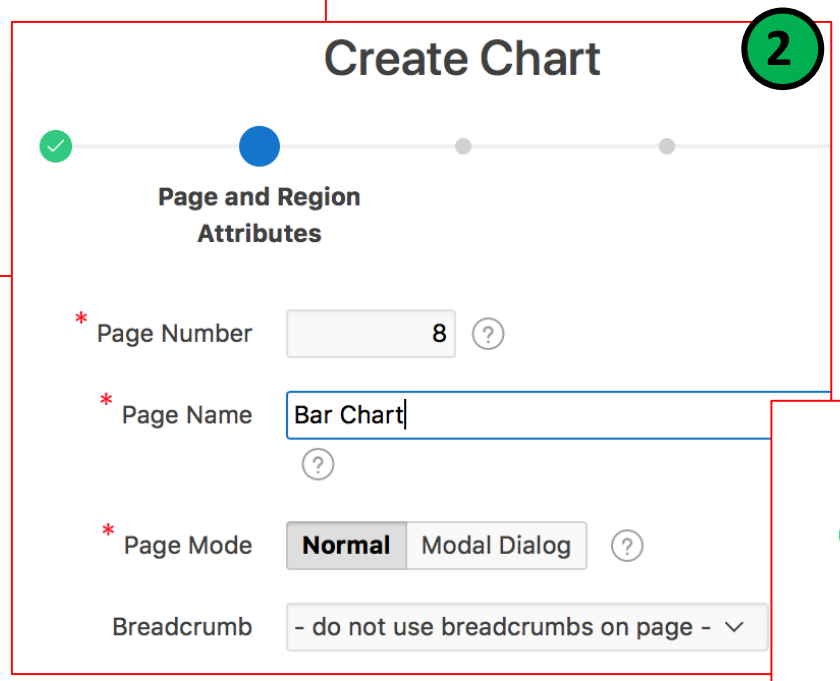
6

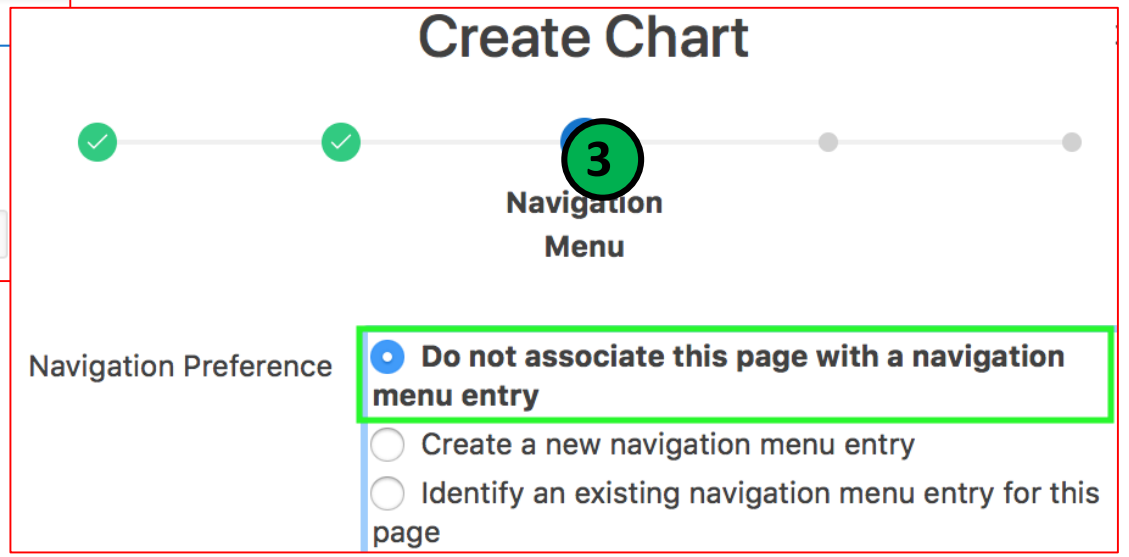
Creating Charts: Bar Chart



Creating a Bar Chart: Steps

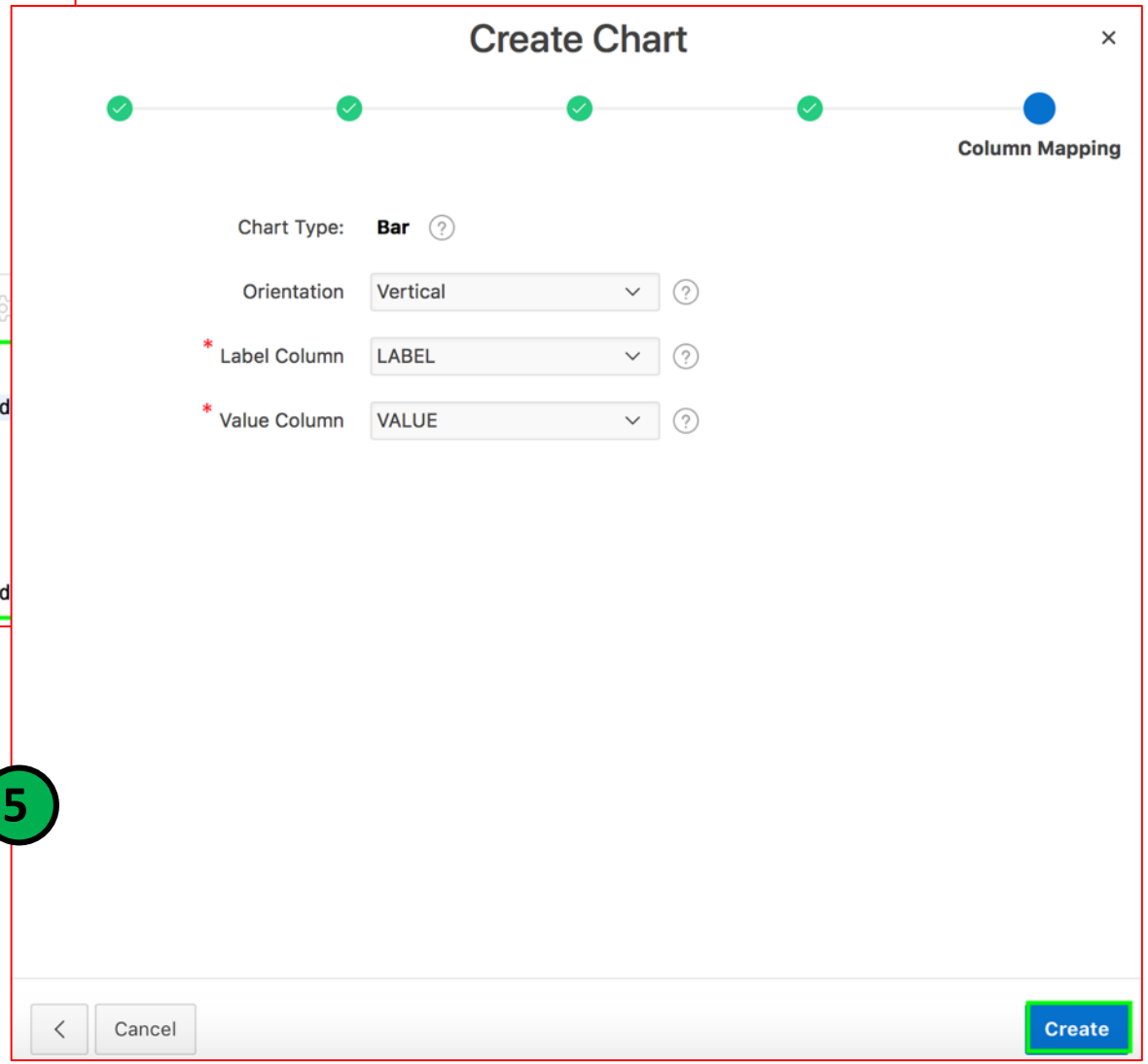






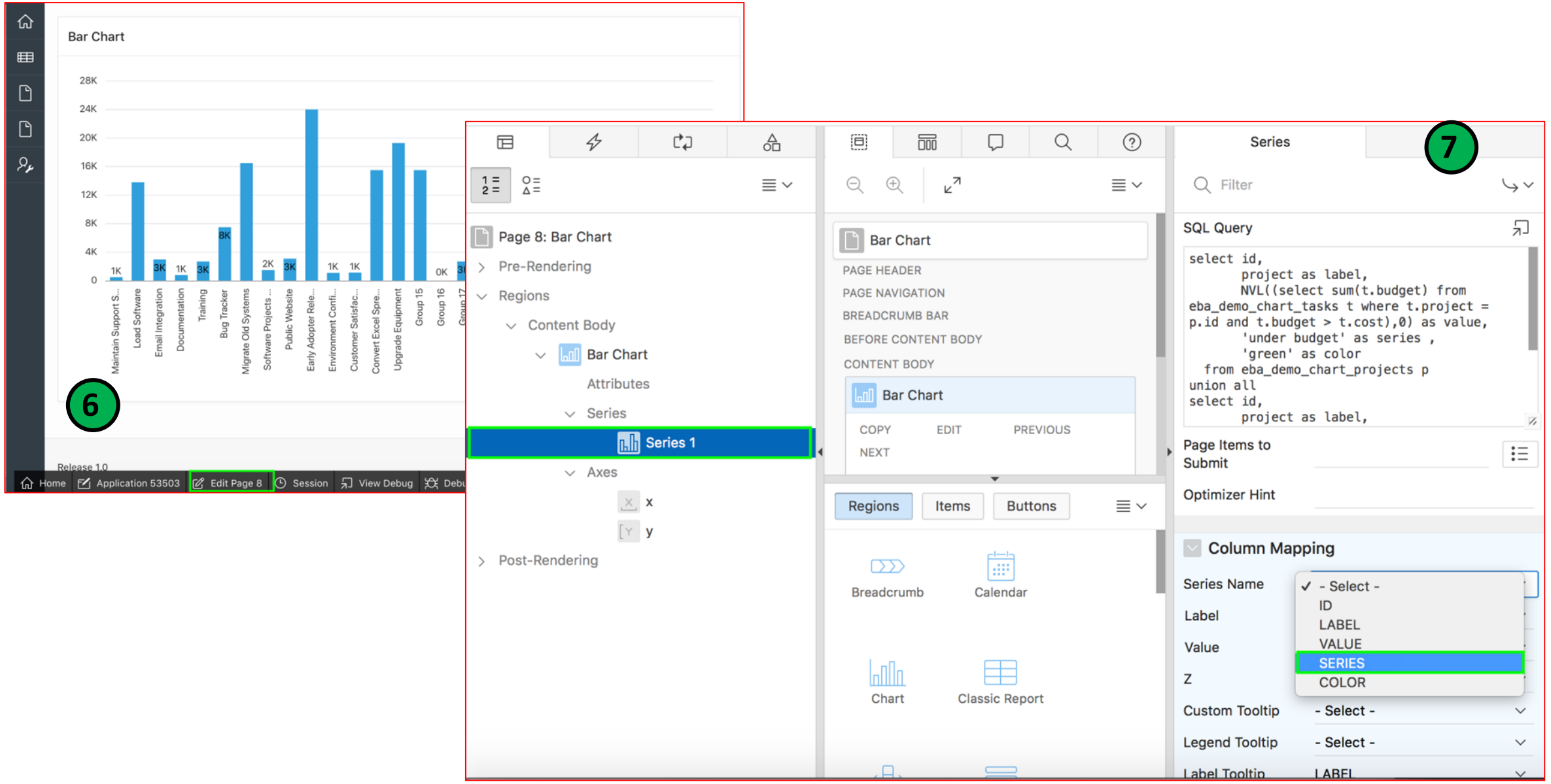
Creating a Bar Chart: Steps

4

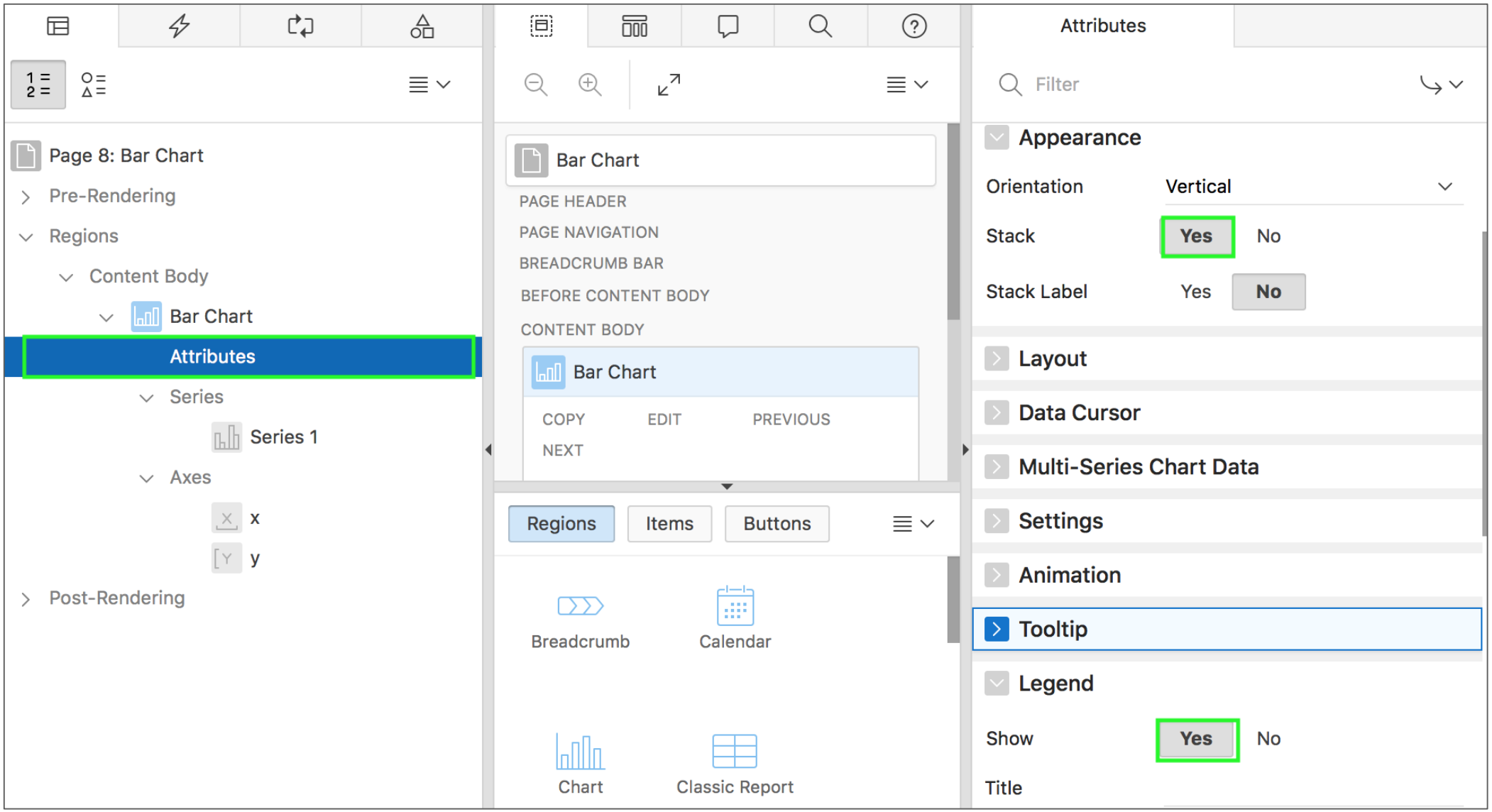


5

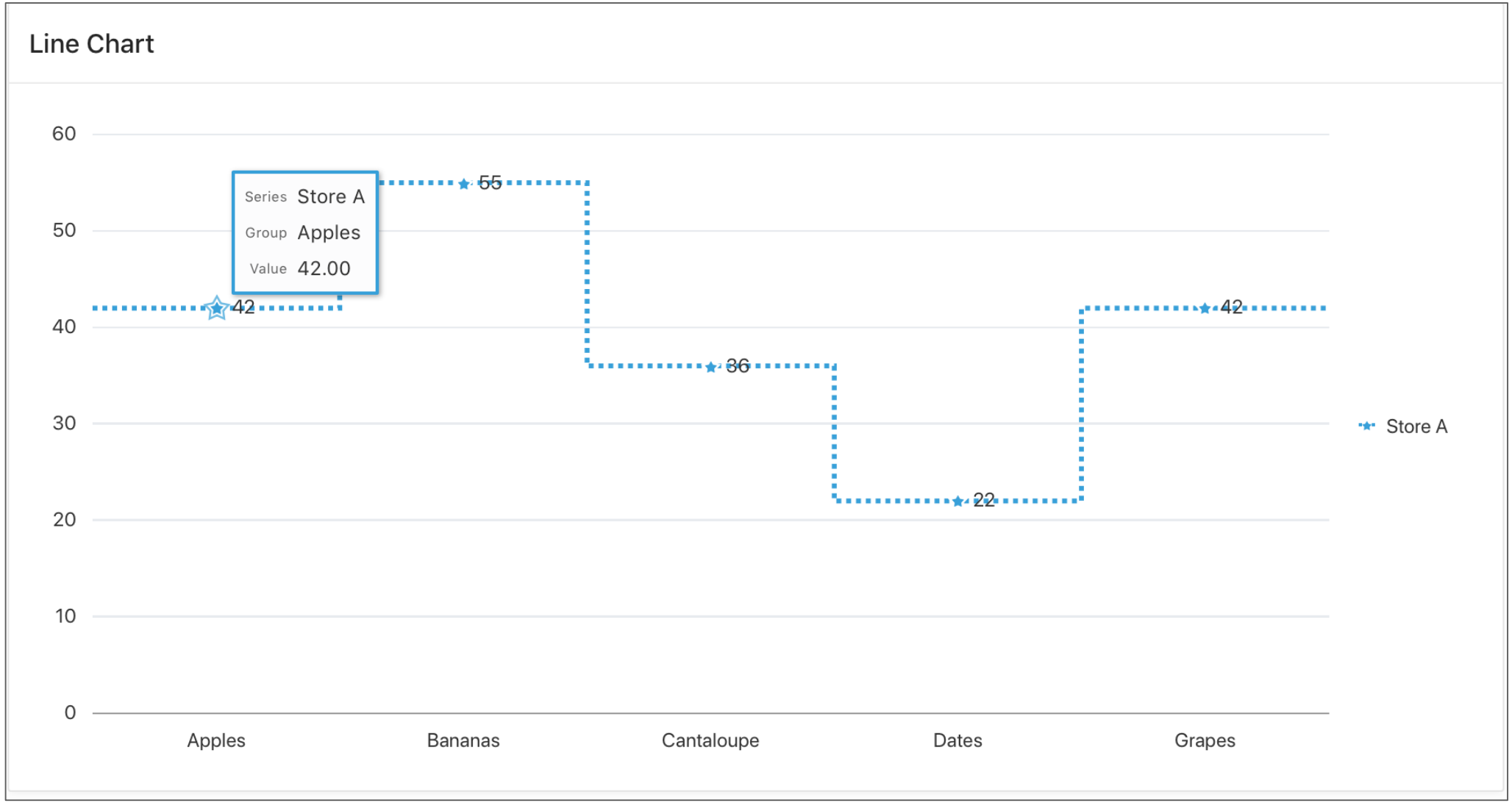
Creating a Bar Chart: Steps



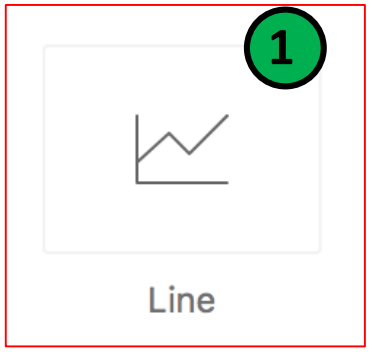
Editing Chart Attributes

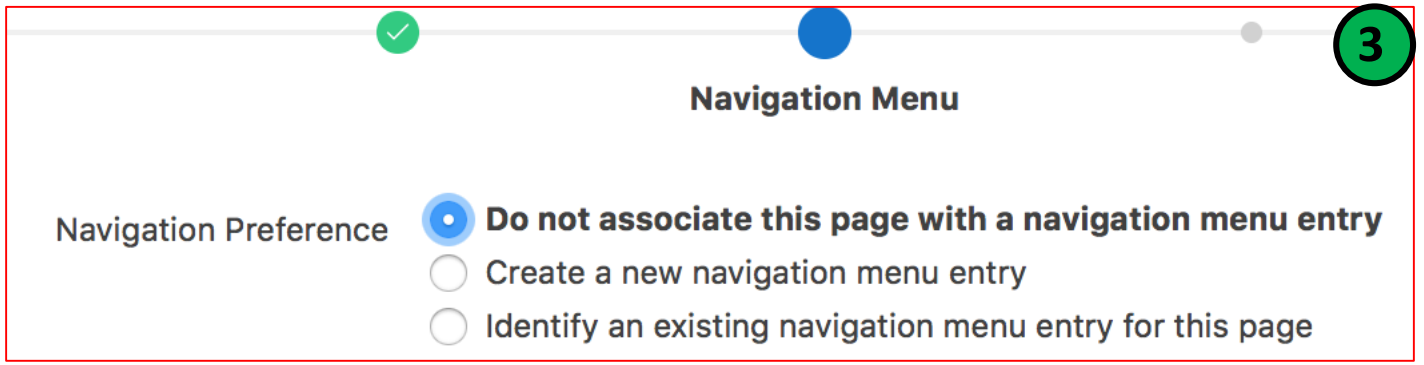


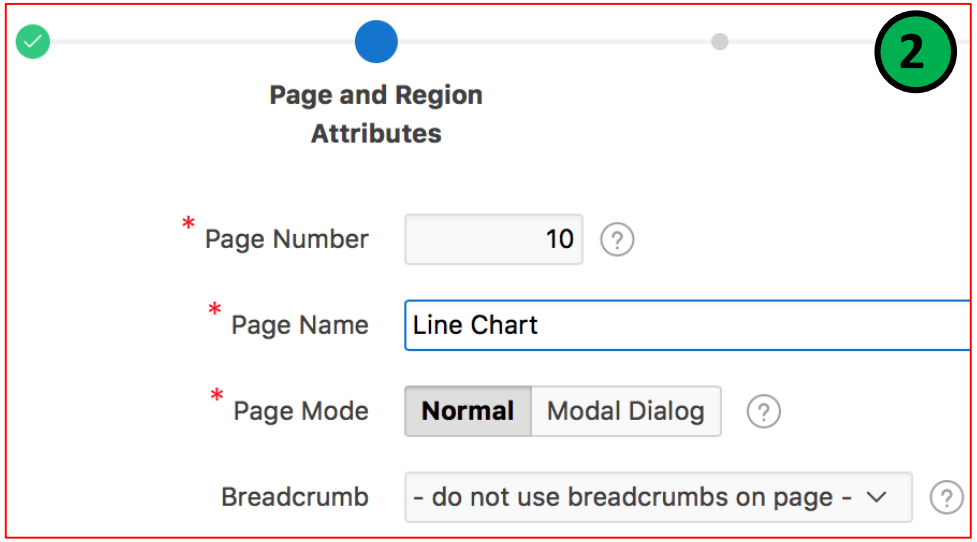
Creating Charts: Line Chart

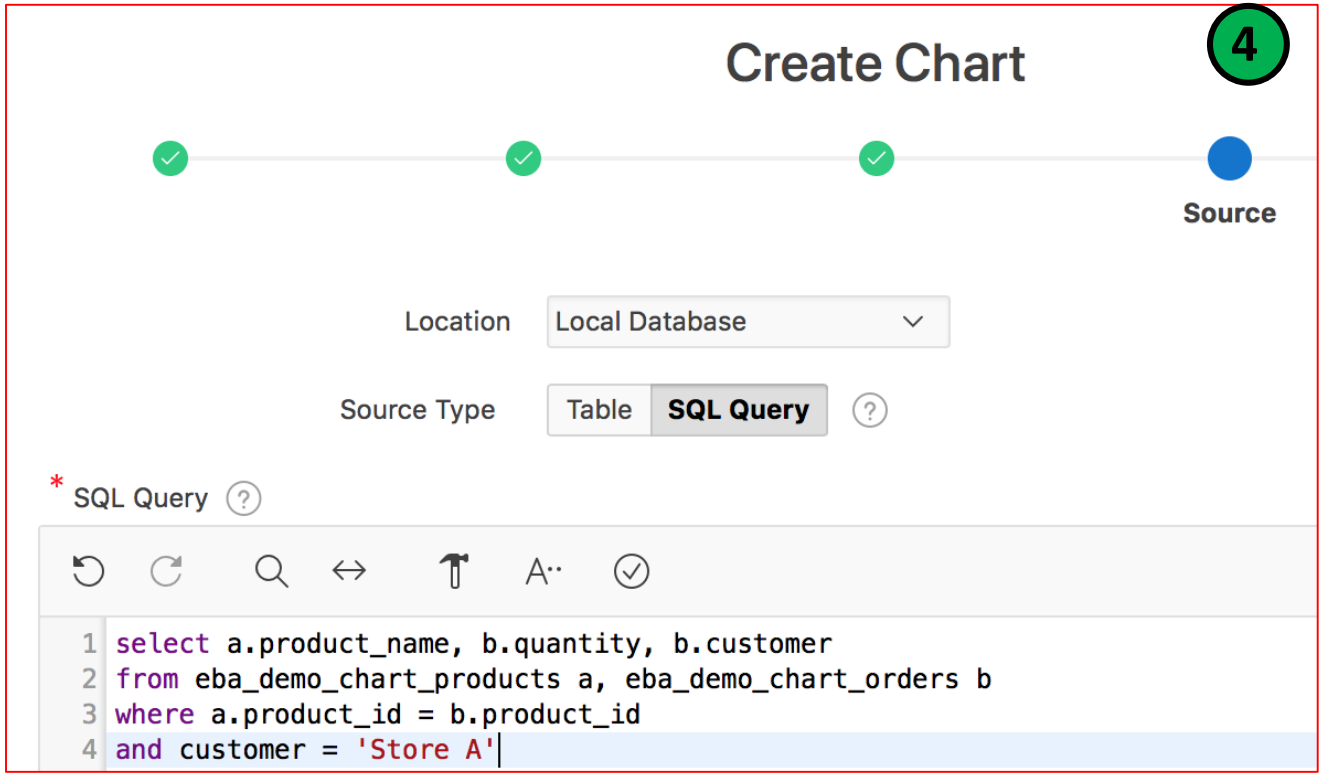


Creating a Line Chart: Steps









Creating a Line Chart: Steps

5

Create Chart

Column Mapping

Chart Type: **Line**

Orientation: Vertical

* Label Column: PRODUCT_NAME

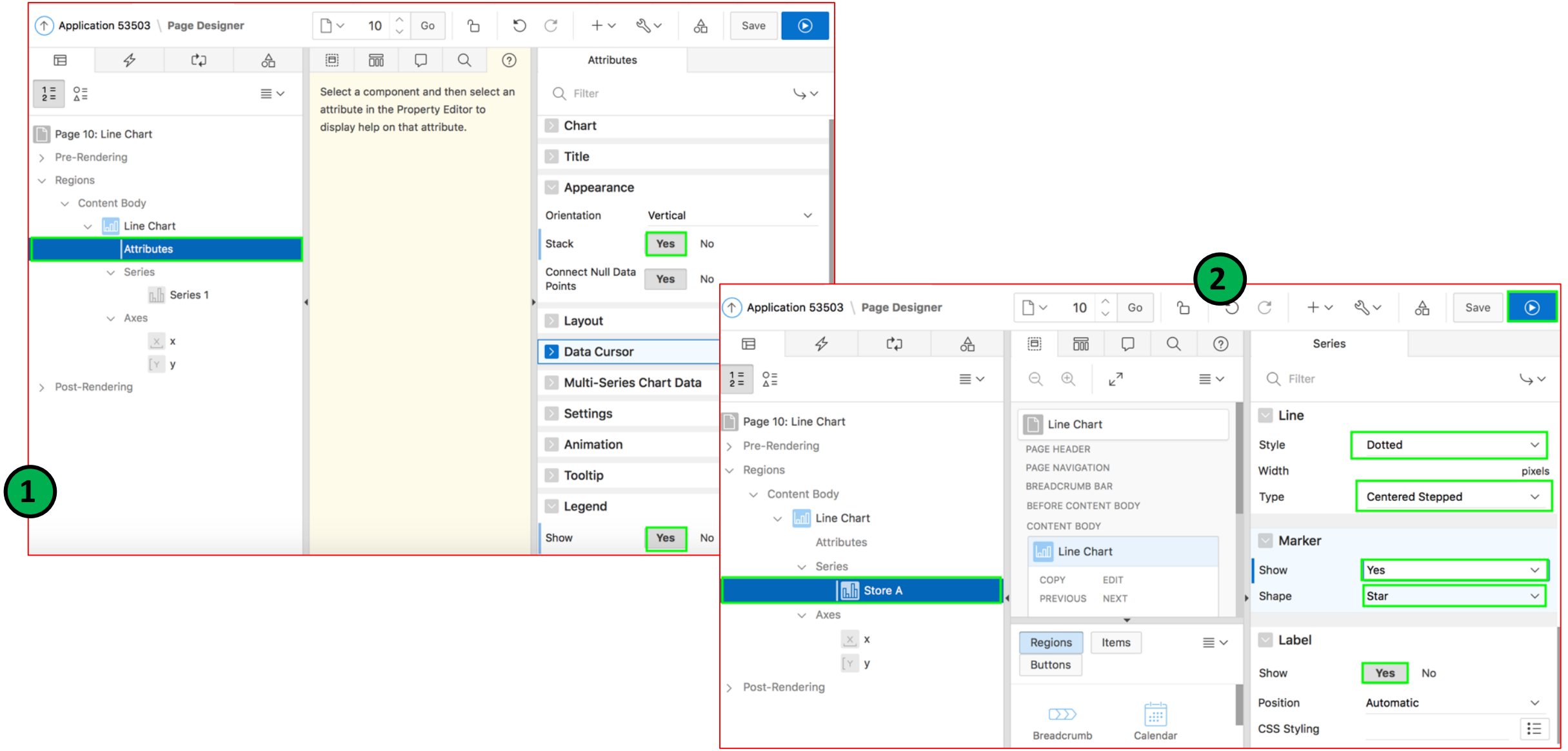
* Value Column: QUANTITY

Cancel

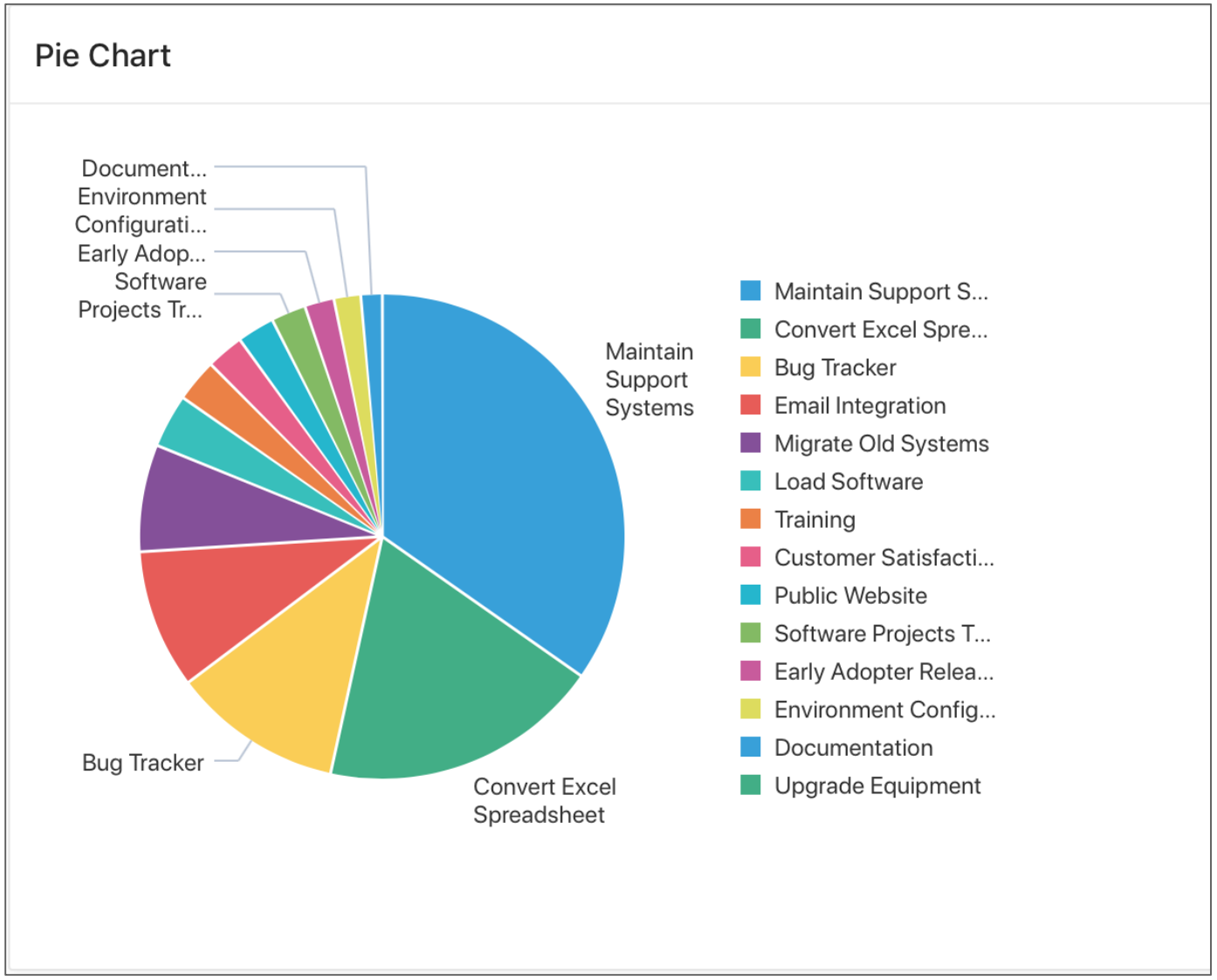
Create

6

Altering the Display of a Line Chart



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

1

Pie

2

Page and Region Attributes

* Page Number 11

* Page Name Pie Chart

* Page Mode **Normal** Modal Dialog

Breadcrumb - do not use breadcrumbs on page -

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

4

Source

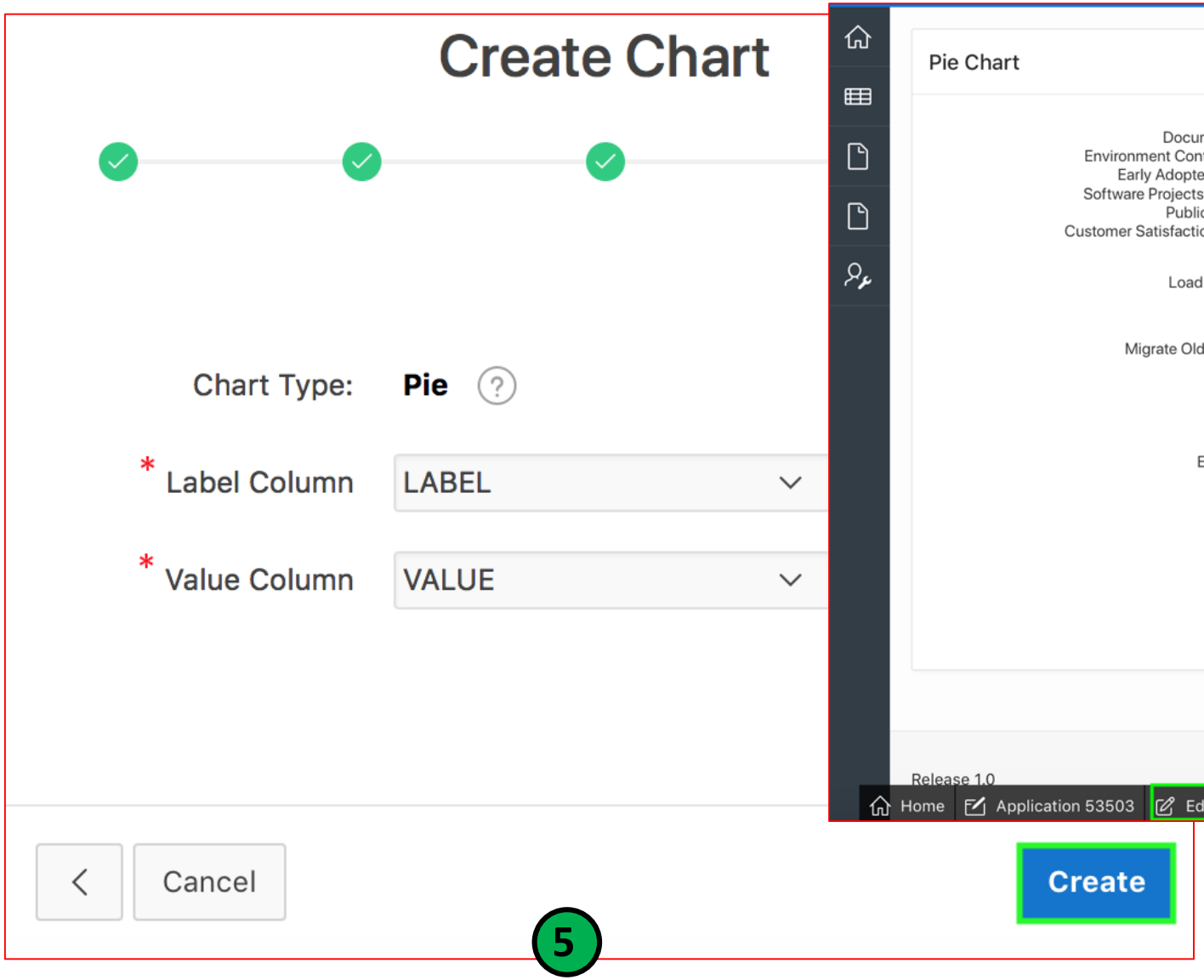
Location Local Database

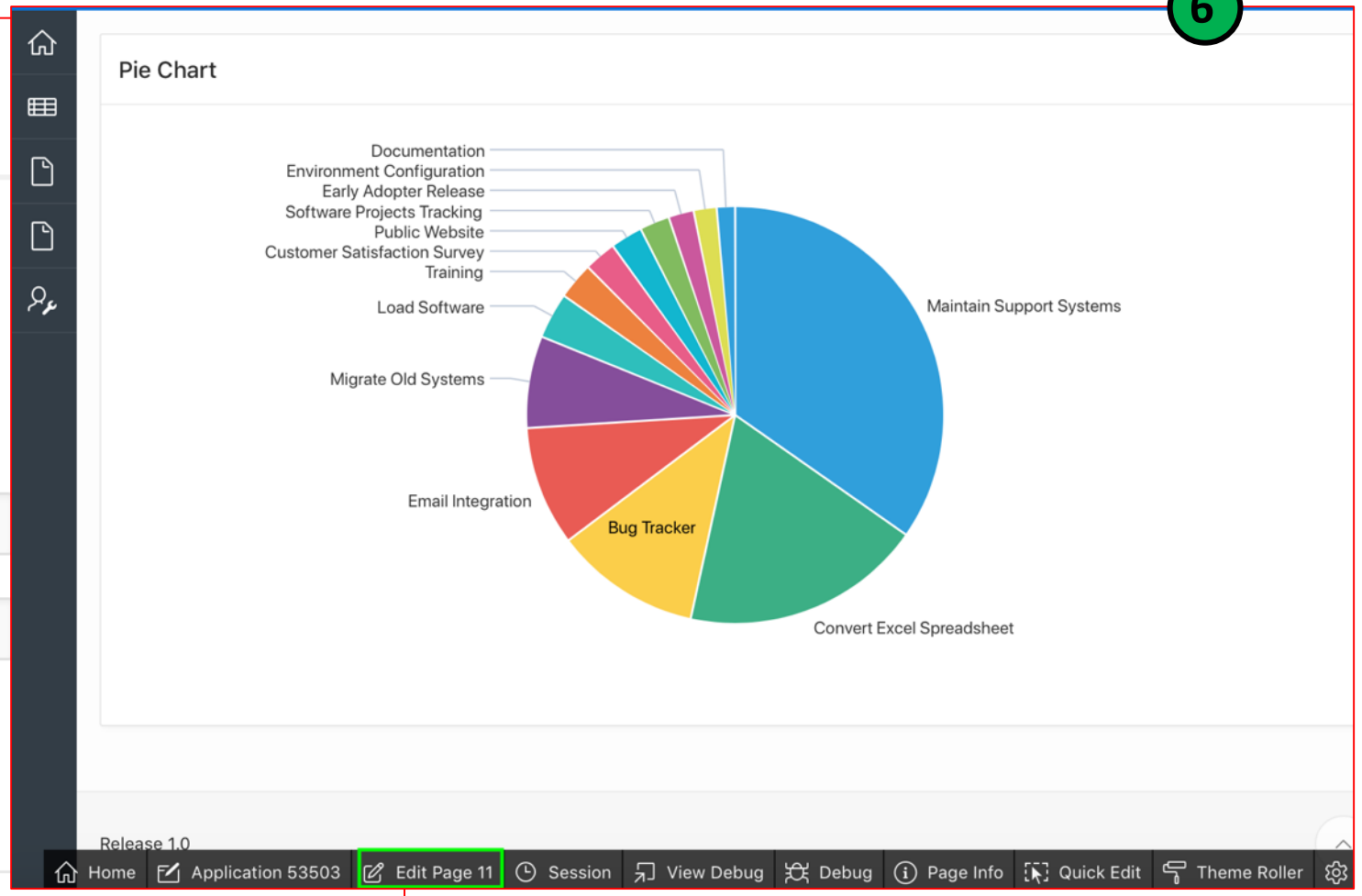
Source Type Table **SQL Query**

* SQL Query

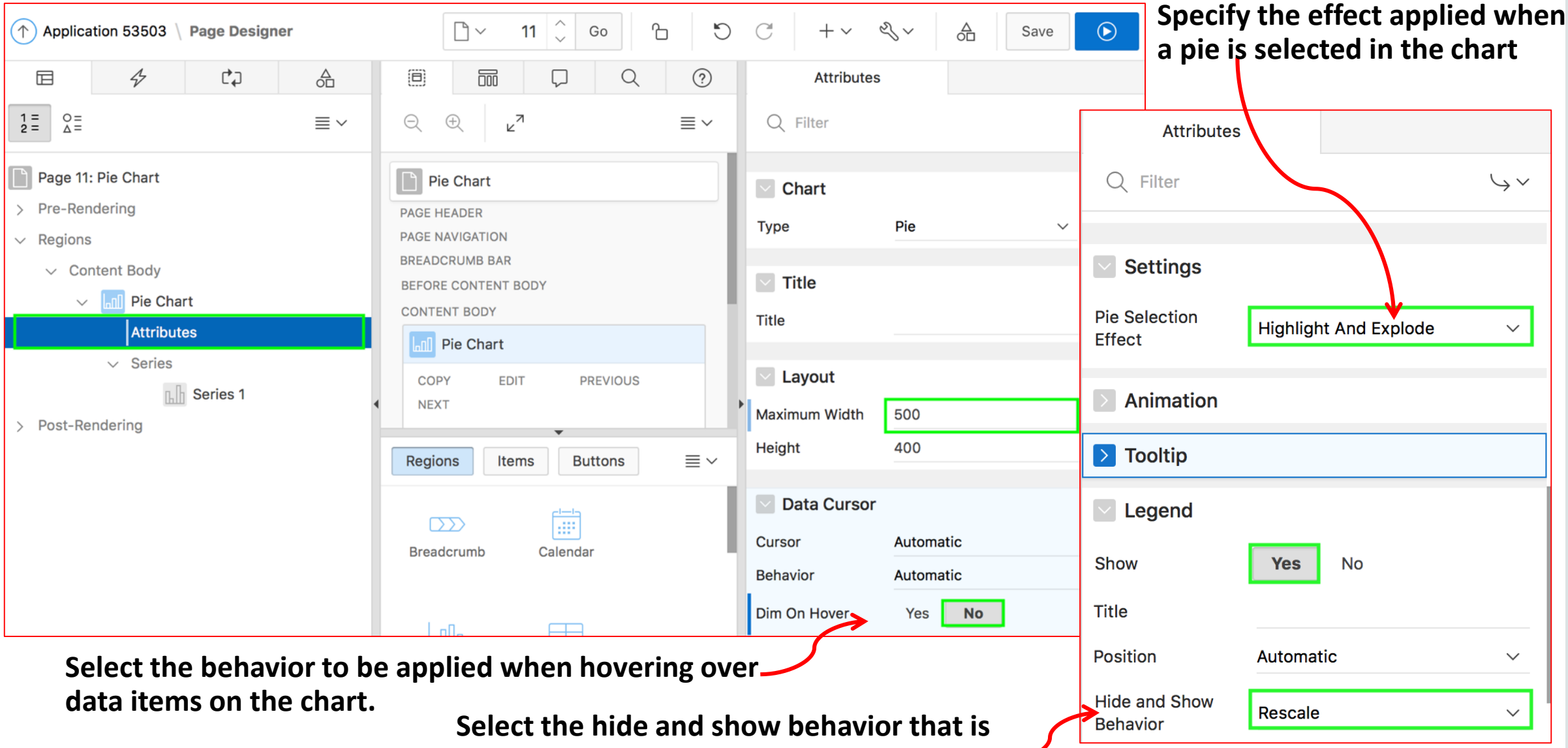
```
1 select b.id,
2        b.project label,
3        sum(a.cost) value
4   from eba_demo_chart_tasks a,
5        eba_demo_chart_projects b
6  where a.project = b.id
7  group by b.id, b.project
8 order by 3 desc
```

Creating a Pie Chart: Steps

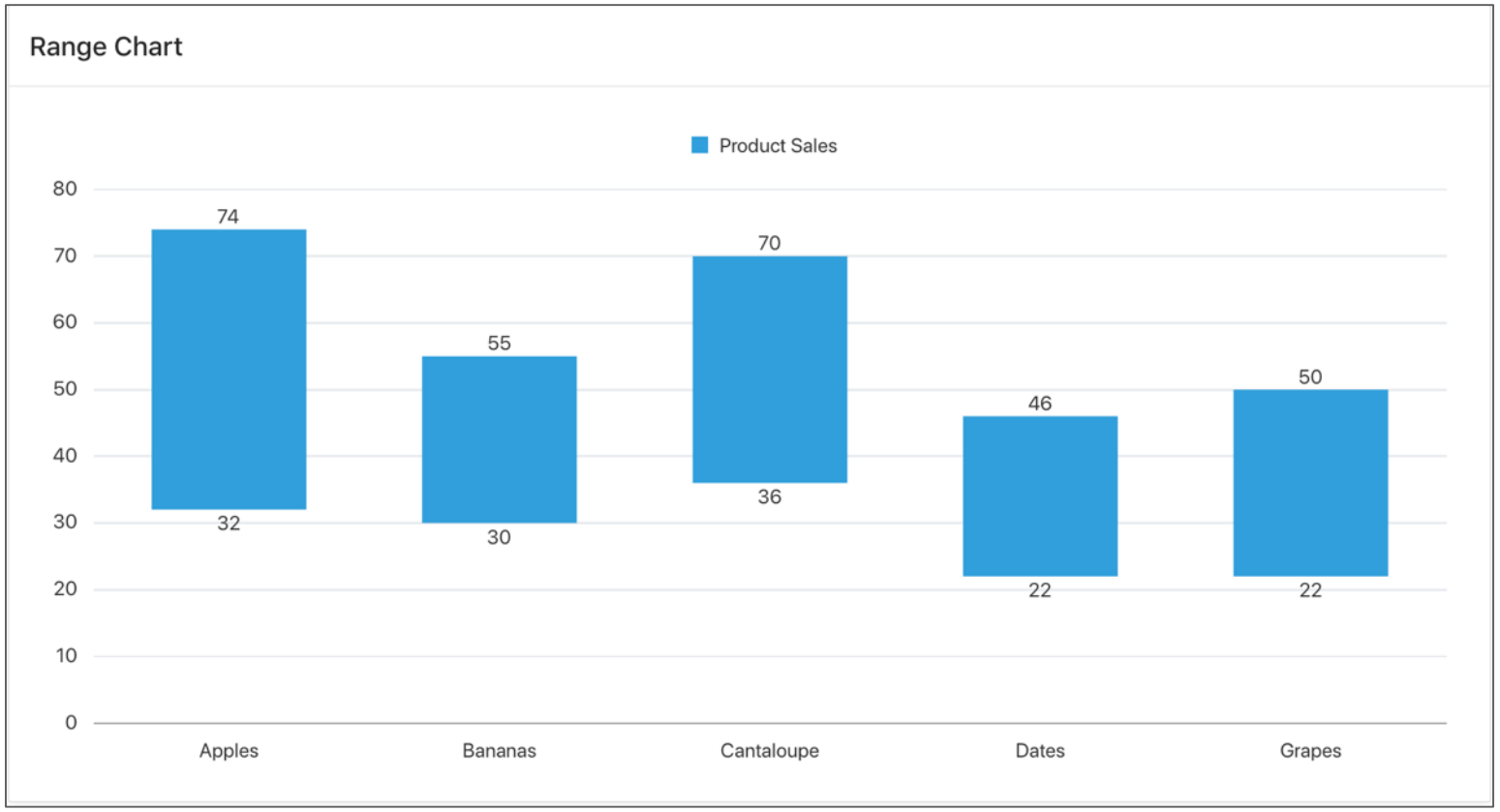




Altering the Display of a Pie Chart



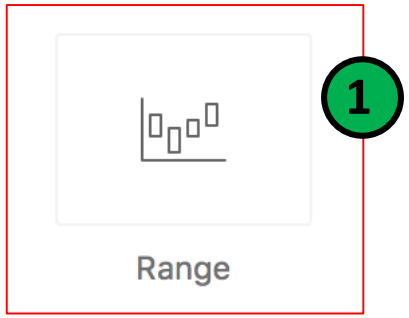
Creating a Range Chart: Example

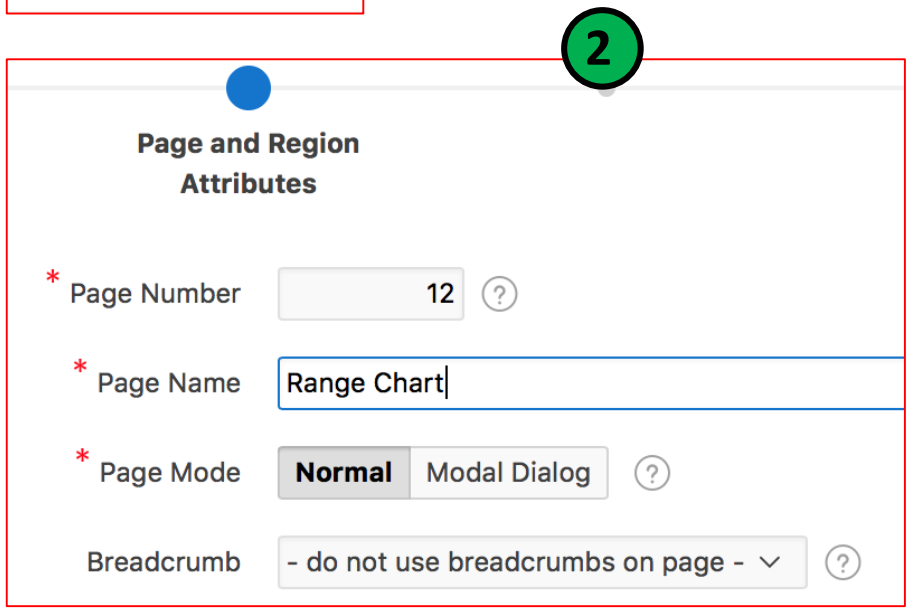


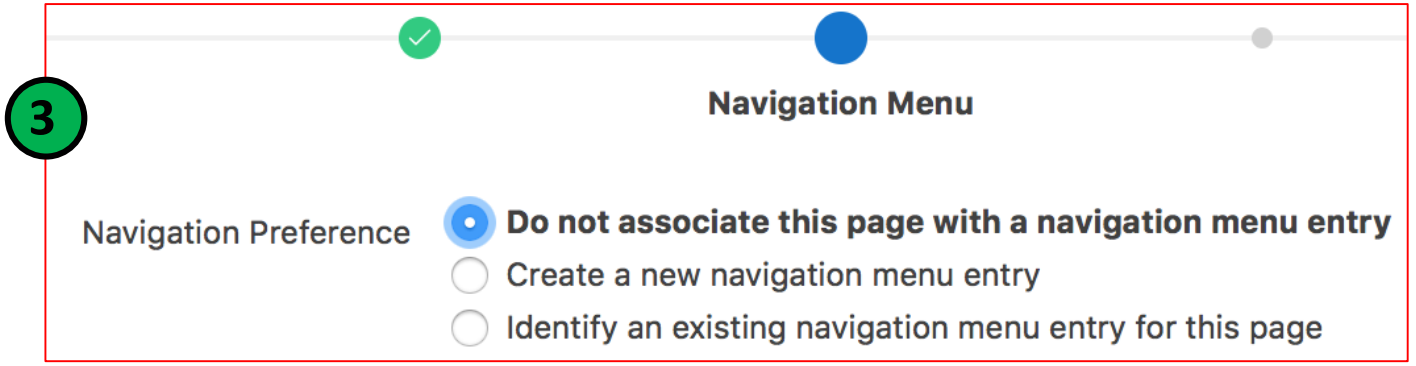

```
select b.product_name, min(a.quantity),
max(a.quantity) from eba_demo_chart_orders a,
eba_demo_chart_products b
where a.product_id = b.product_id
group by a.product_id, b.product_name
order by b.product_name asc
```

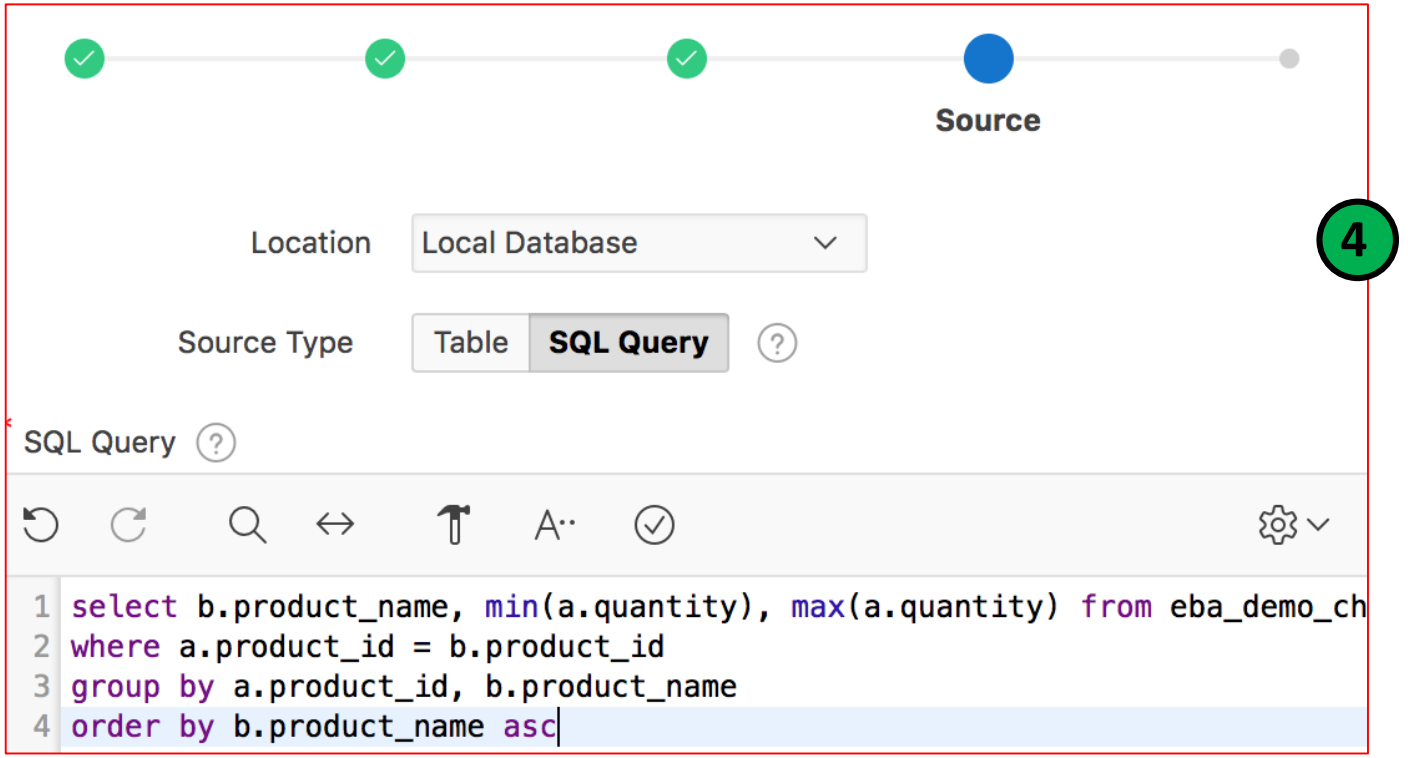
SQL query used for the chart

Creating a Range Chart: Steps









Creating a Range Chart: Steps

5

Create Chart

Chart Type: **Range**

Orientation: Vertical

* Label Column: PRODUCT_NAME

* Low Value Column: MIN(A.QUANTITY)

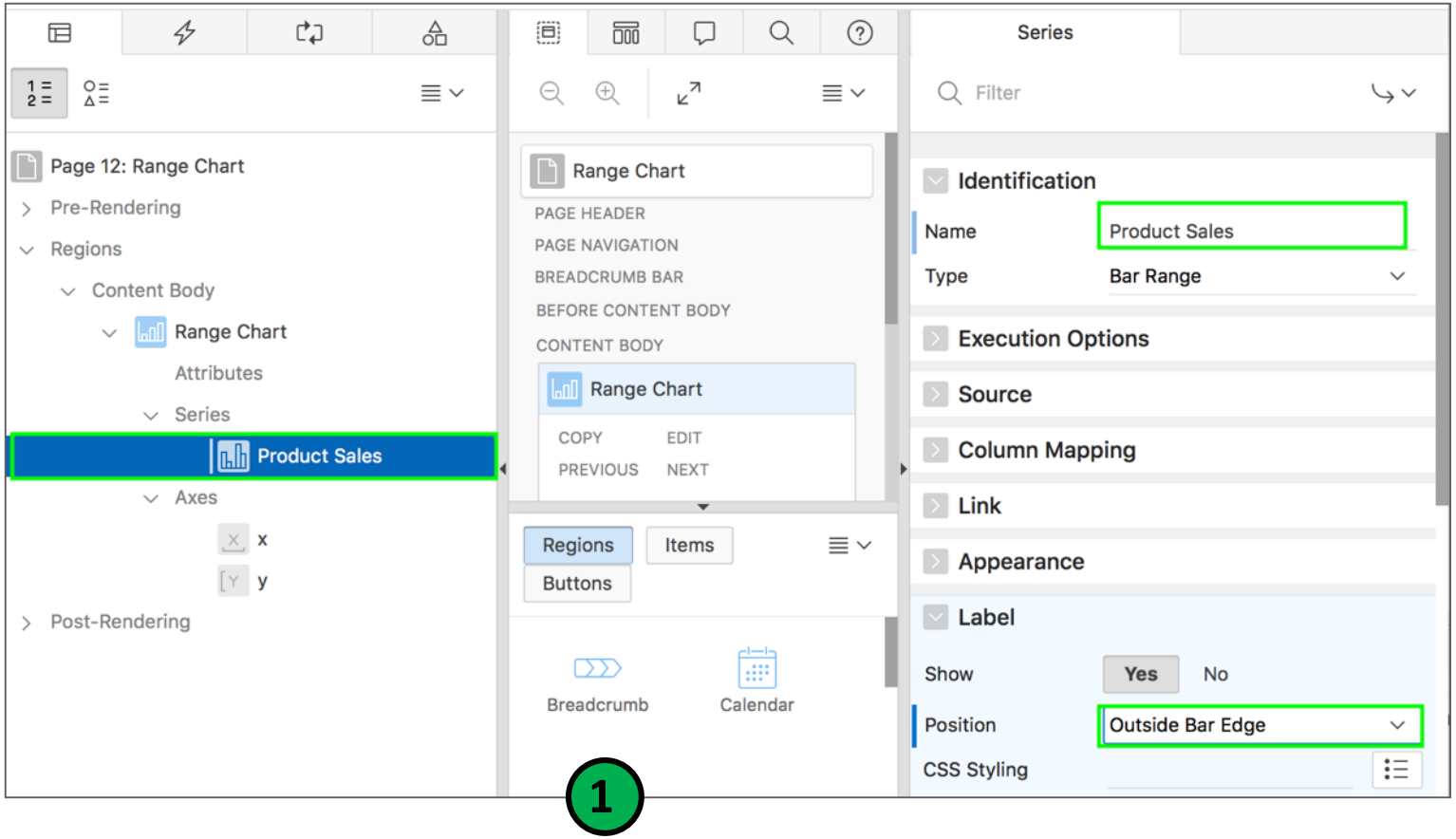
* High Value Column: MAX(A.QUANTITY)

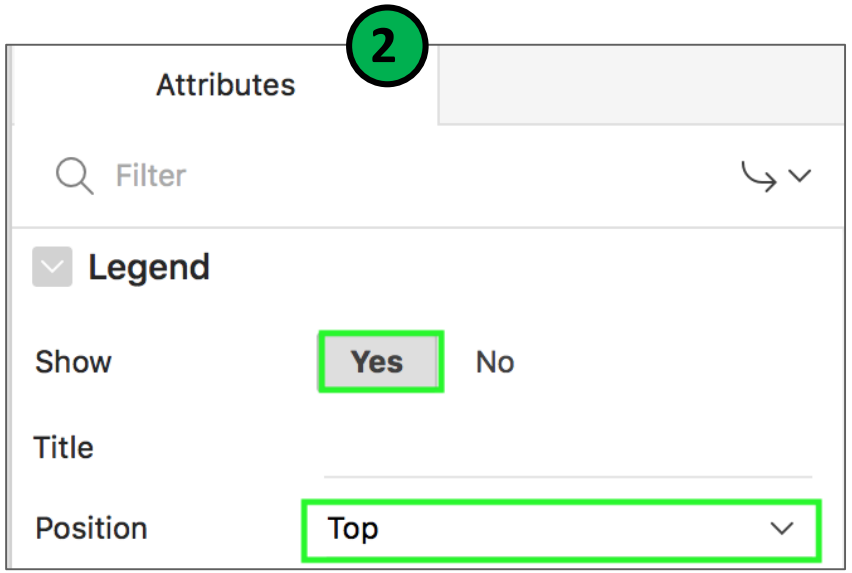
Cancel

Create

6

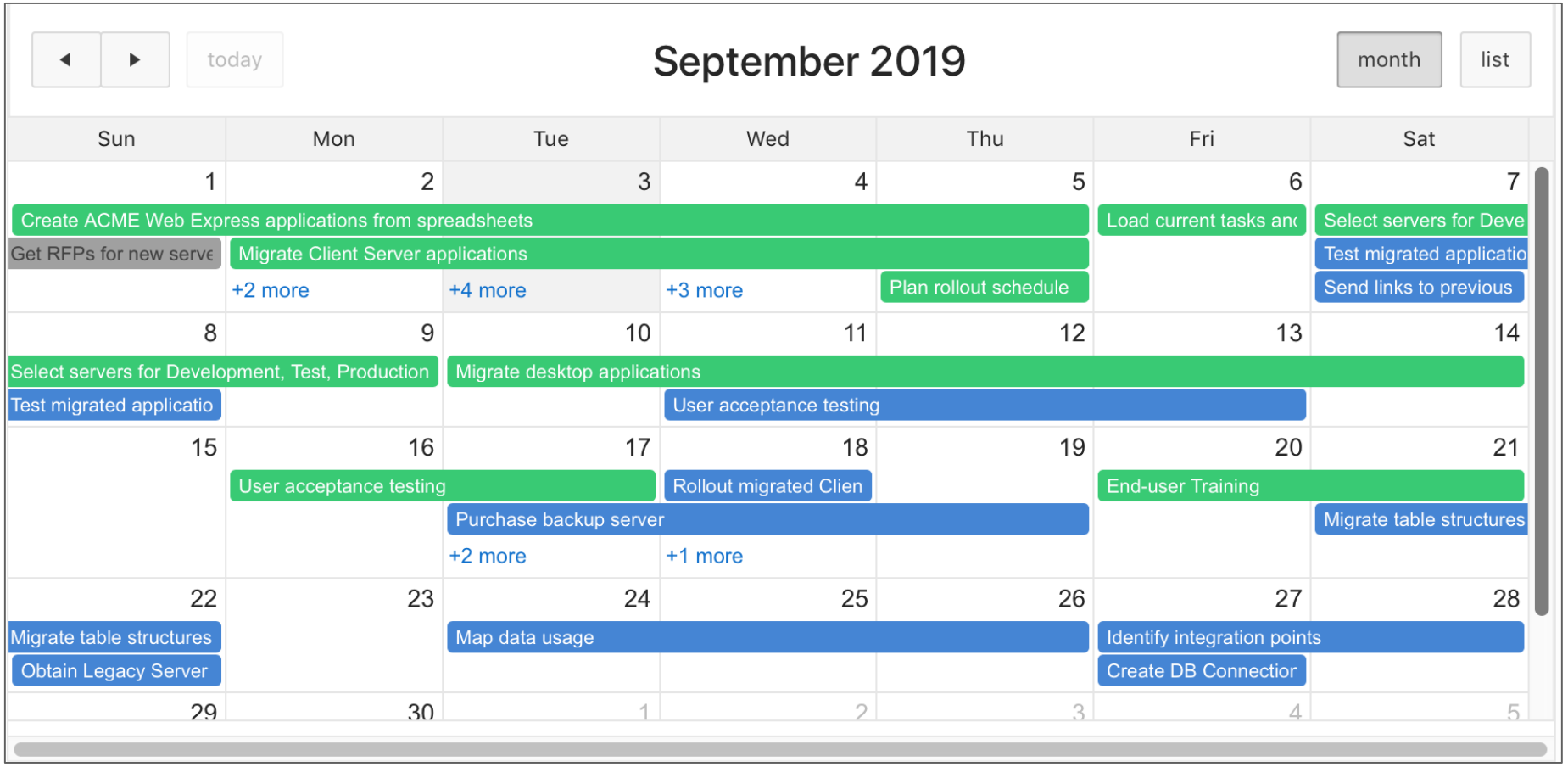
Customizing the Display of a Range Chart



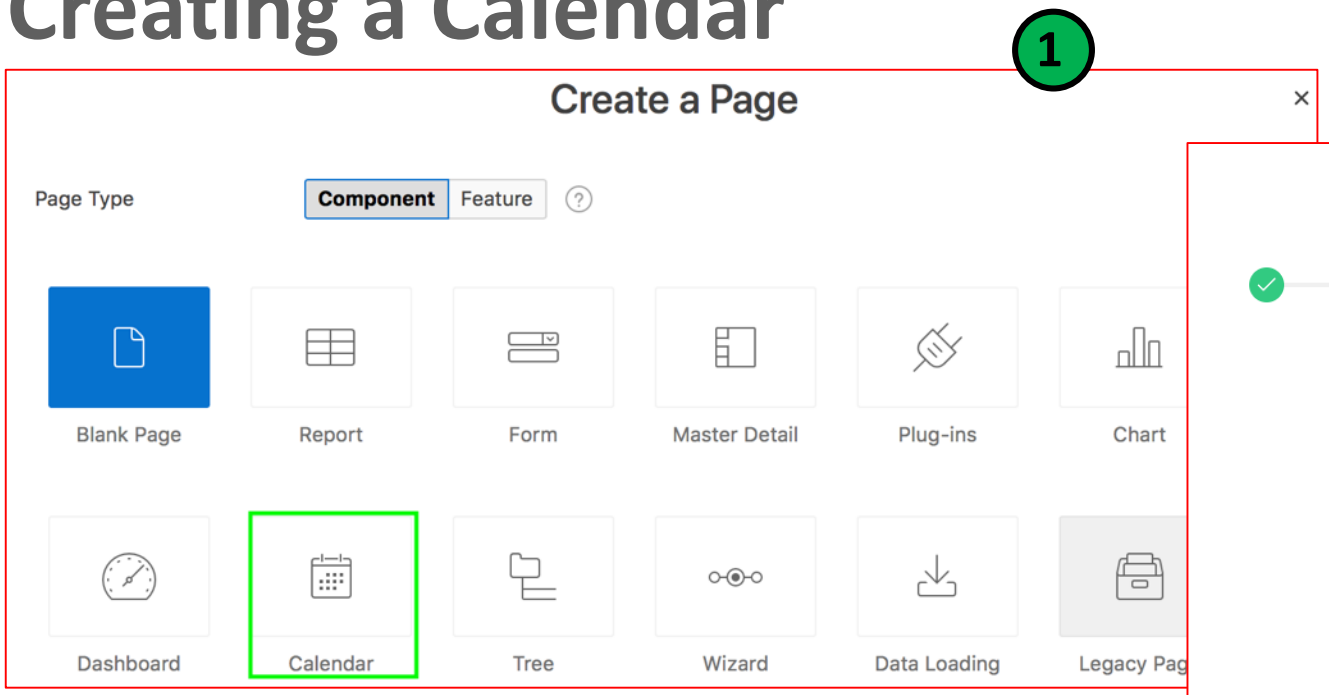


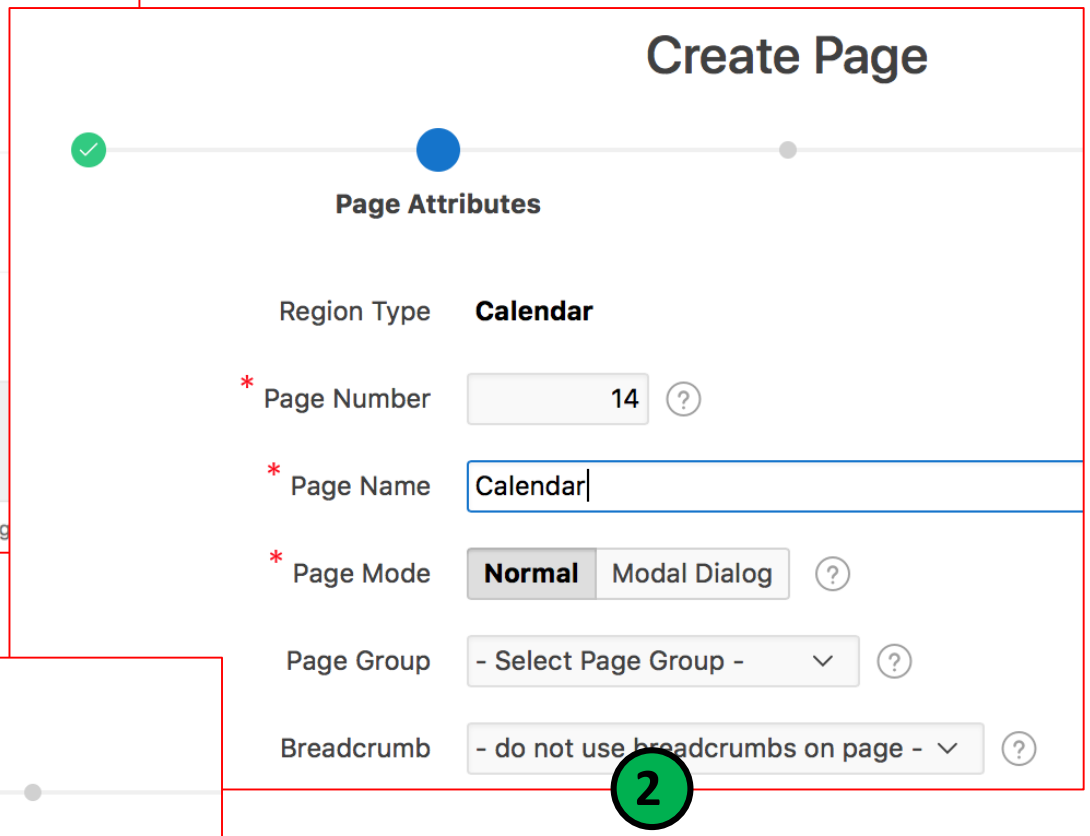
Creating Calendars

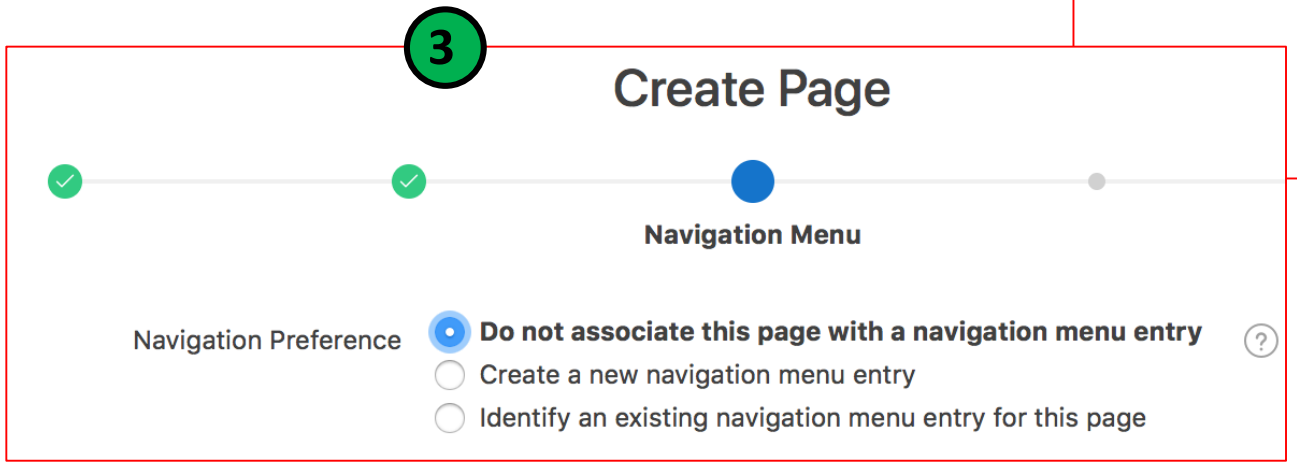
- Oracle Application Express has an integrated calendaring component
- Calendars in APEX are based on a table or SQL query you provide



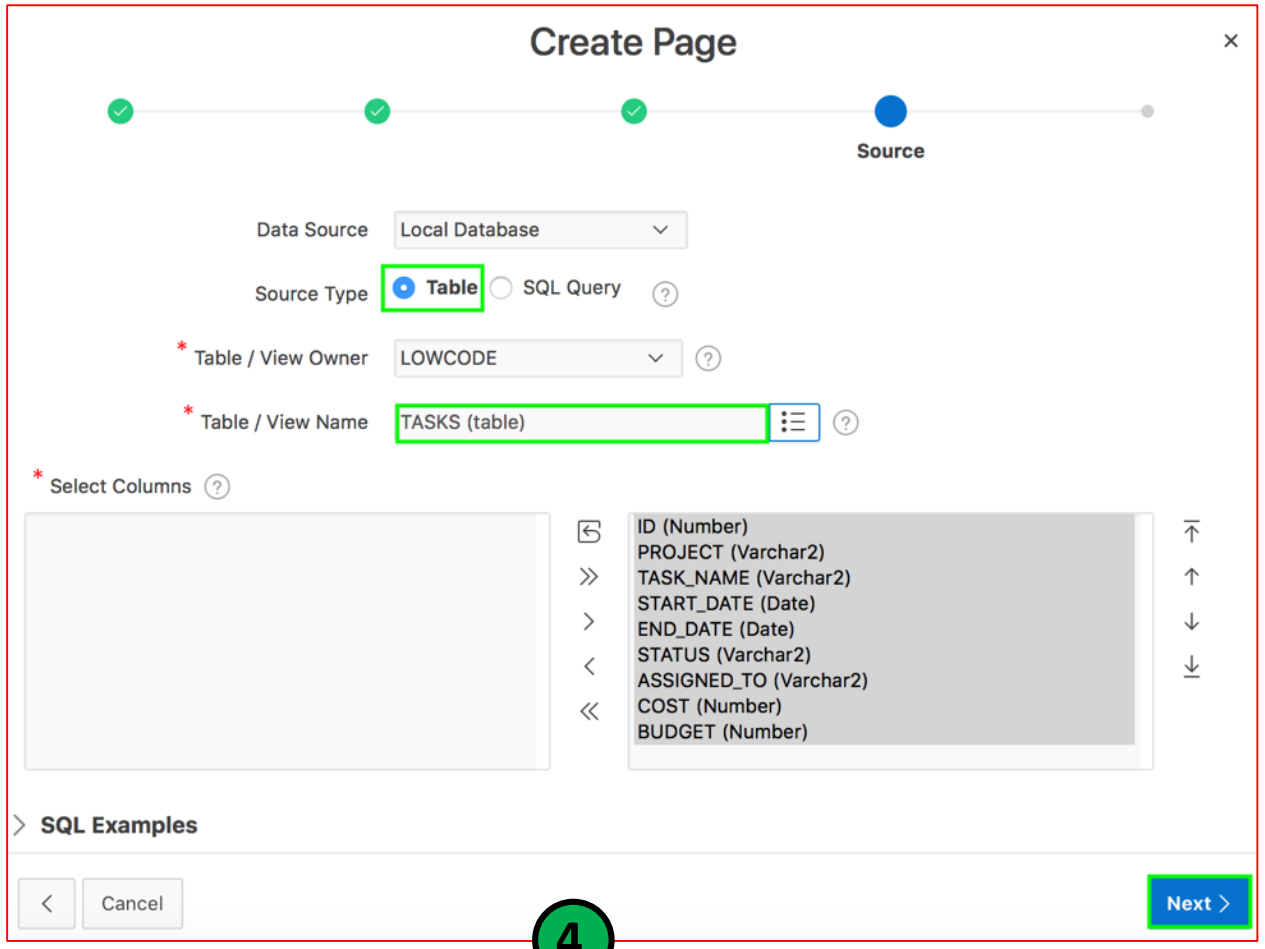
Creating a Calendar

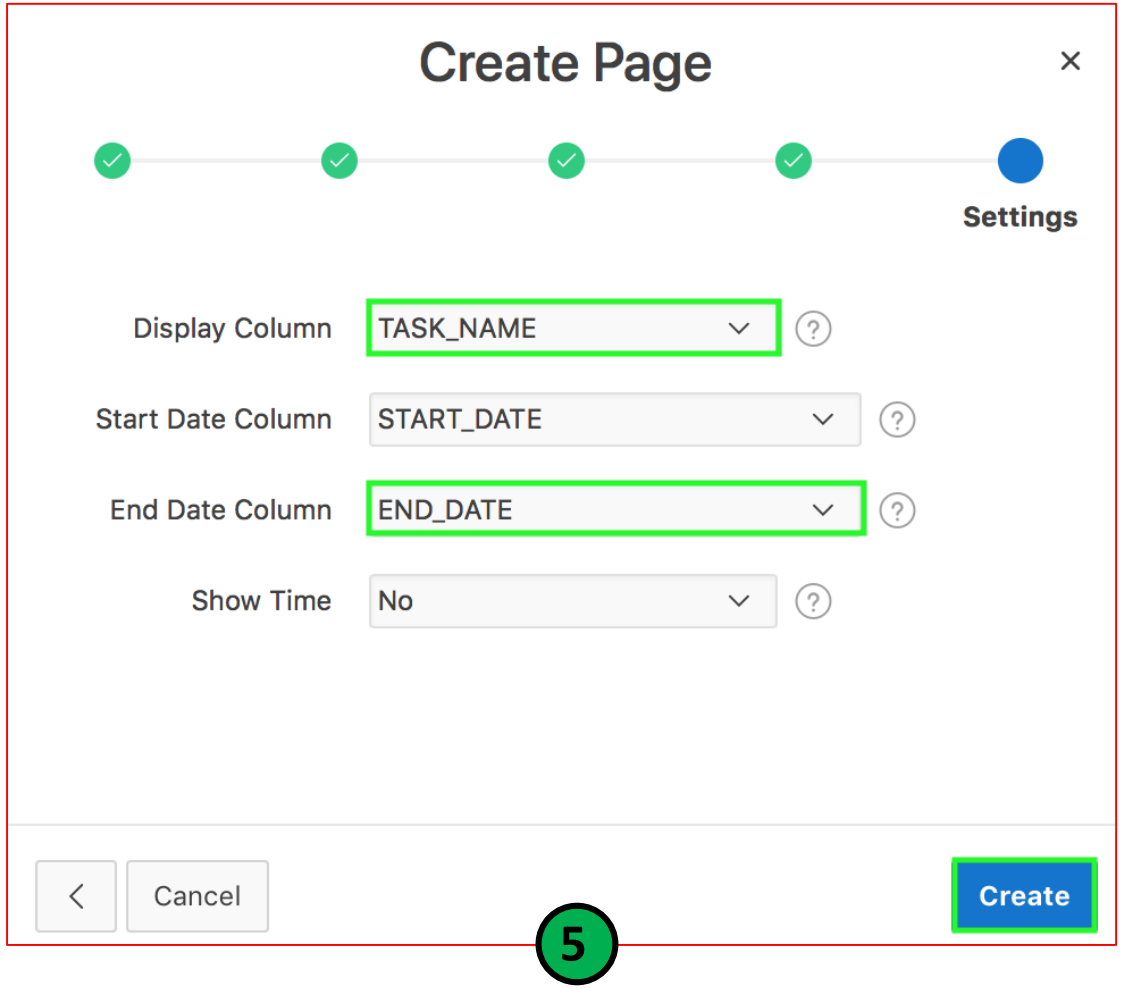




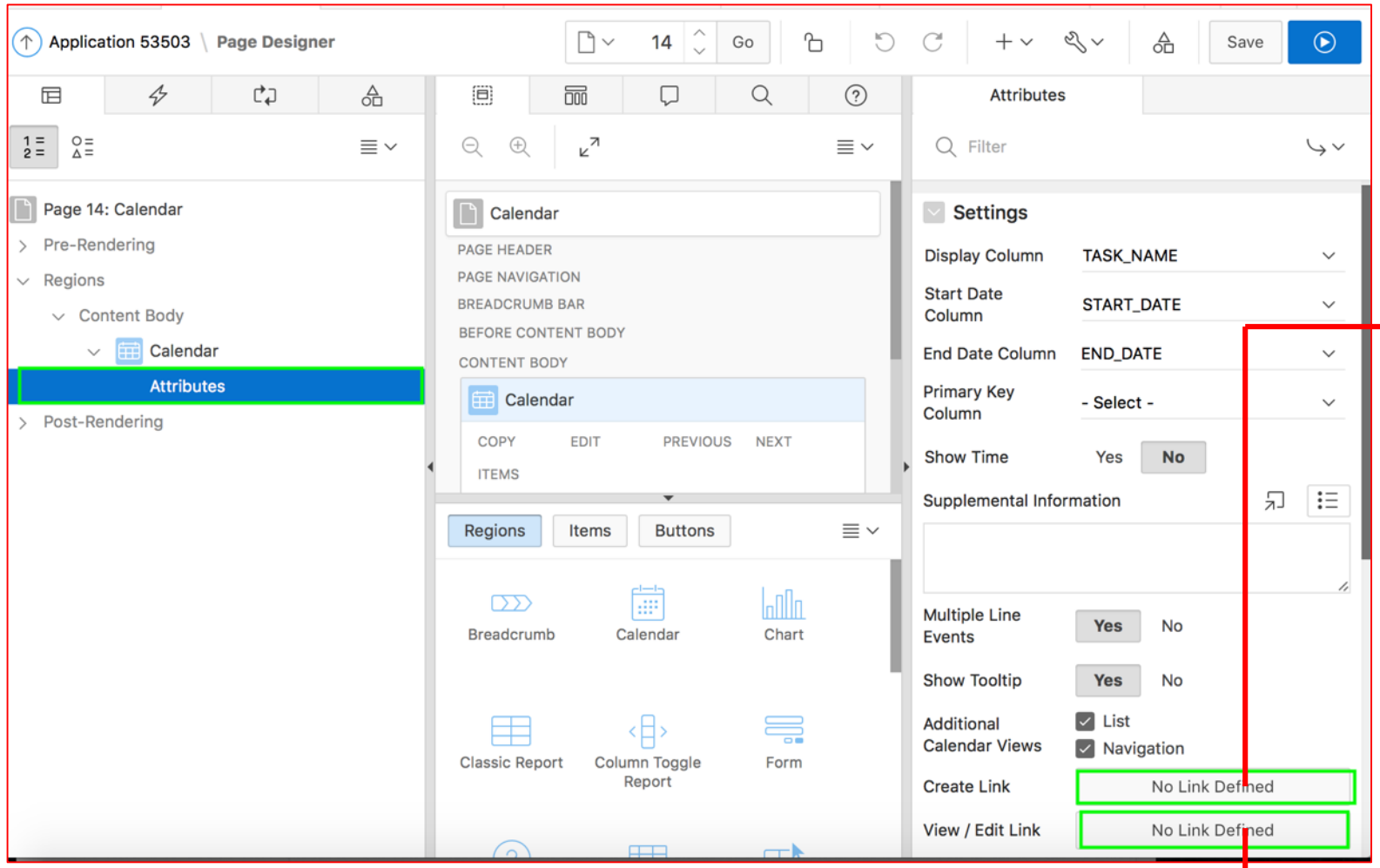


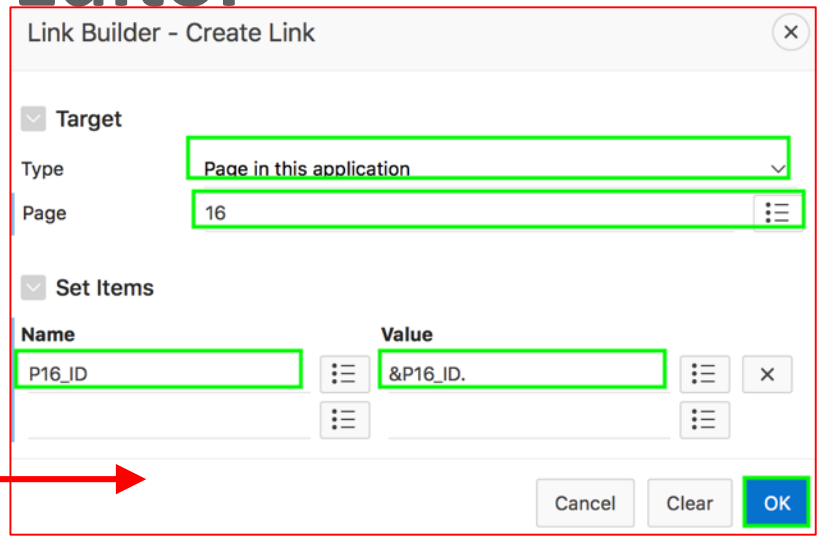
Creating a Calendar

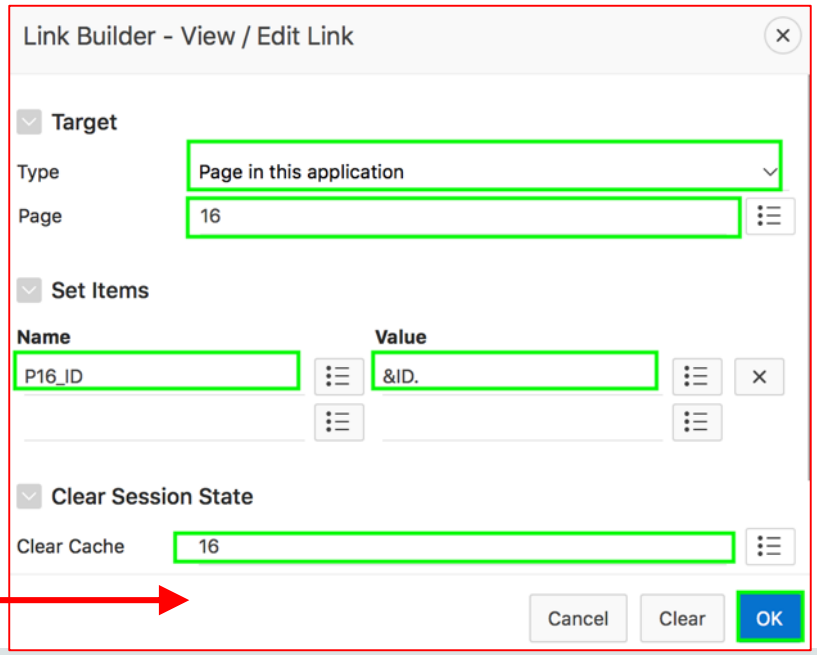




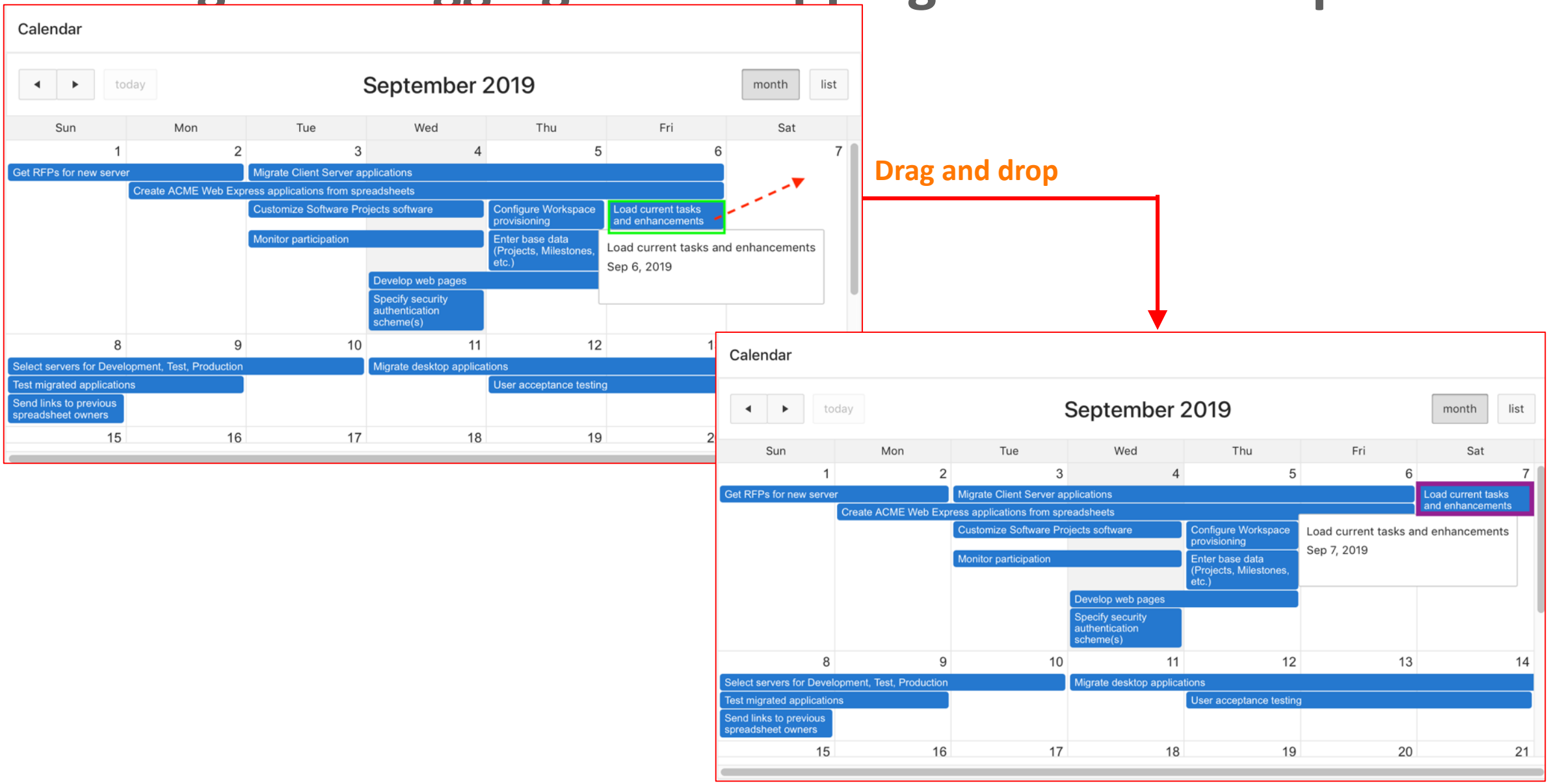
Editing Calendar Attributes in Property Editor



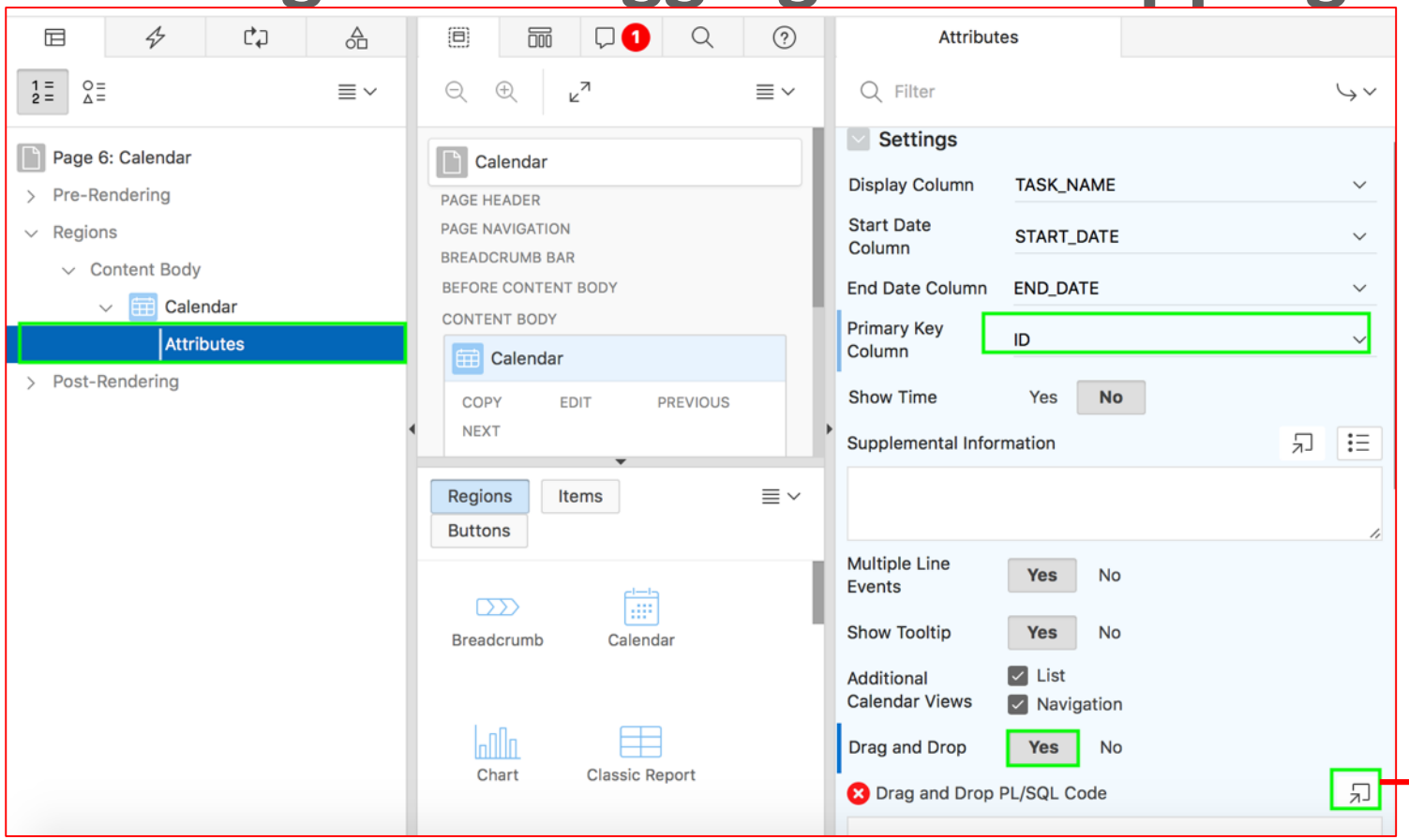




Enabling the Dragging and Dropping of Data: Example



Enabling the Dragging and Dropping of Data: Steps




```
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

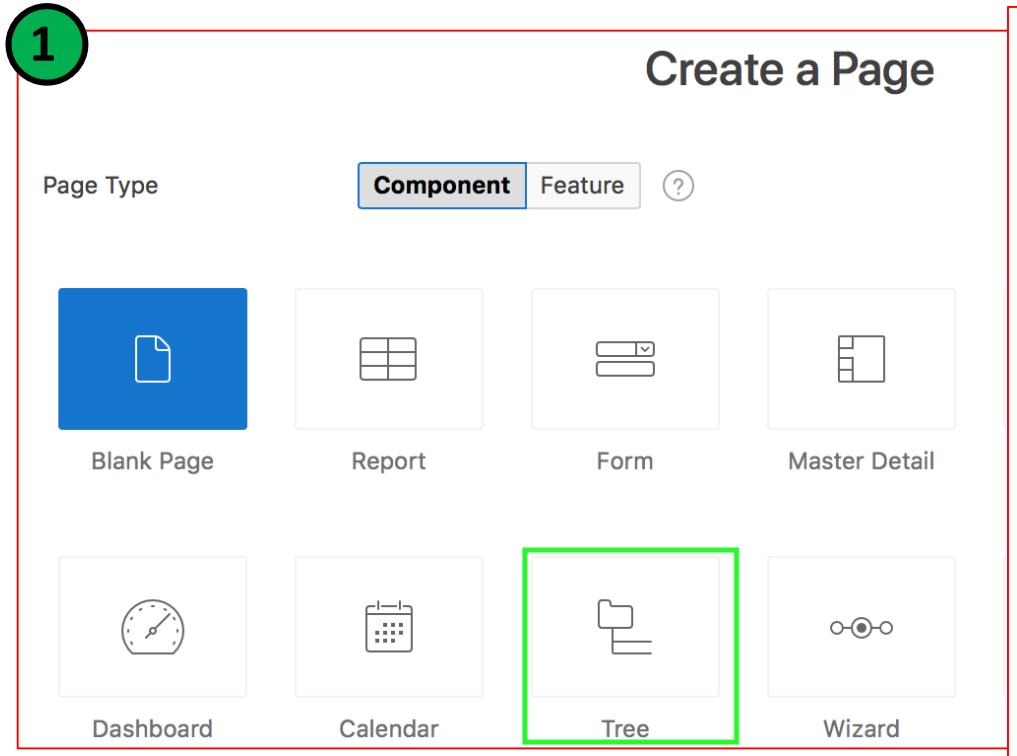
- A tree is a hierarchical navigation mechanism
- Trees are implemented using a single hierarchical query that identifies the row to be used as the start of your query, and the relationship between parent rows and child rows of the hierarchy


```
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"
```

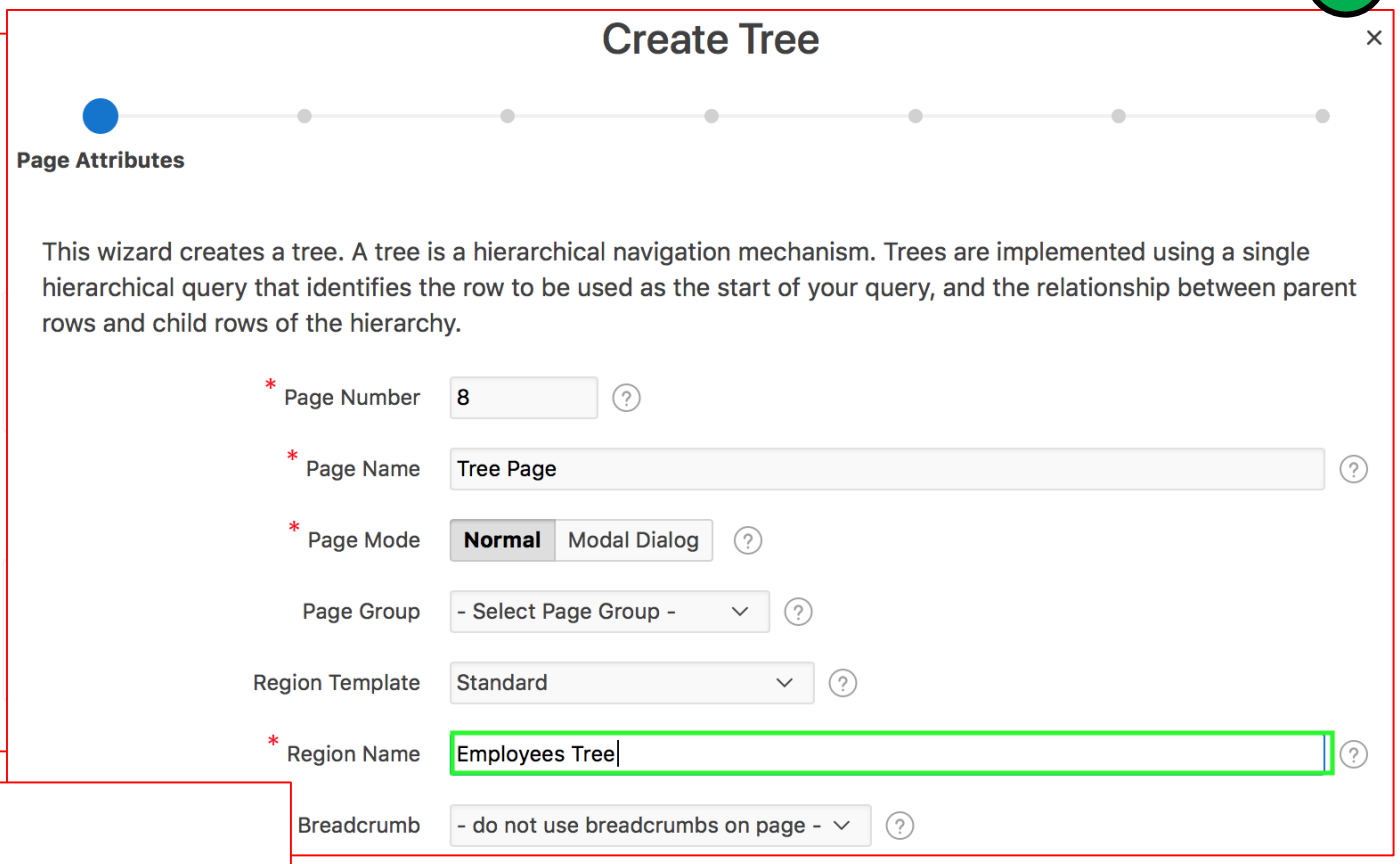
SQL used for the Employees Tree

Creating a Tree: Example

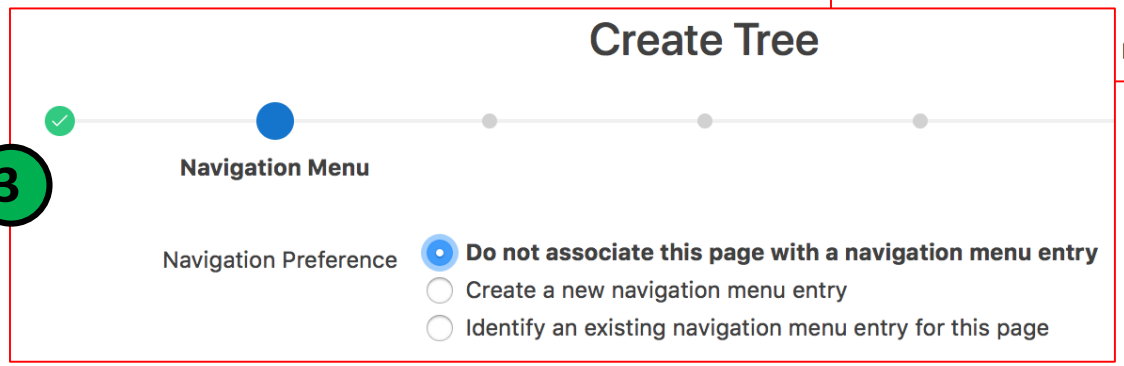
1



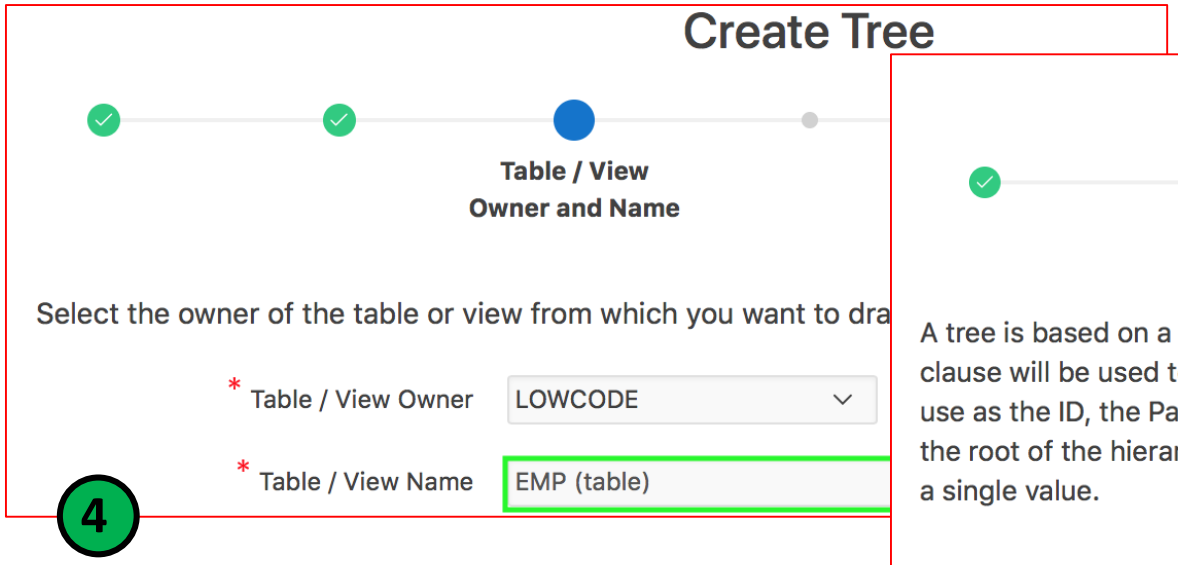
2

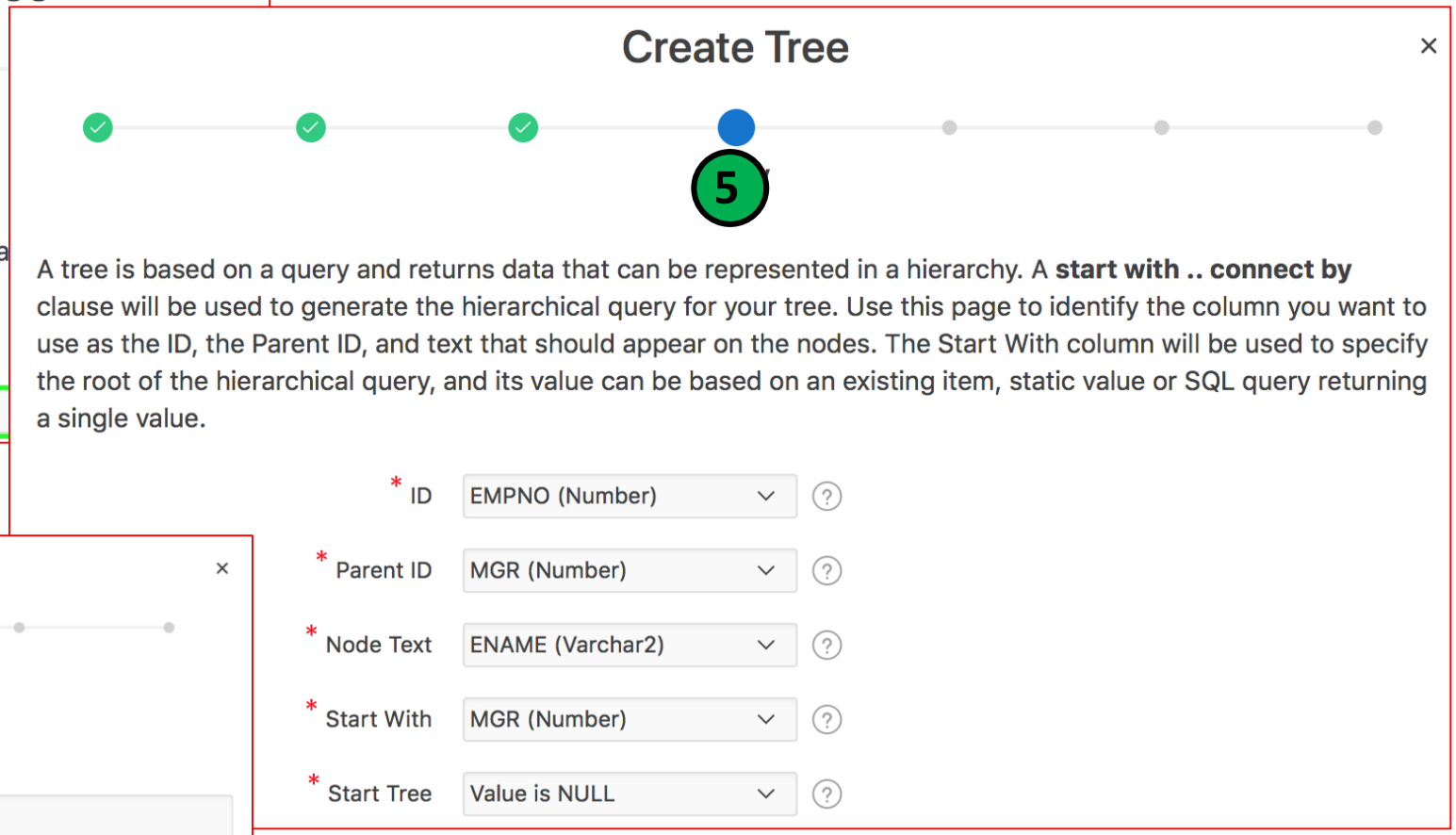


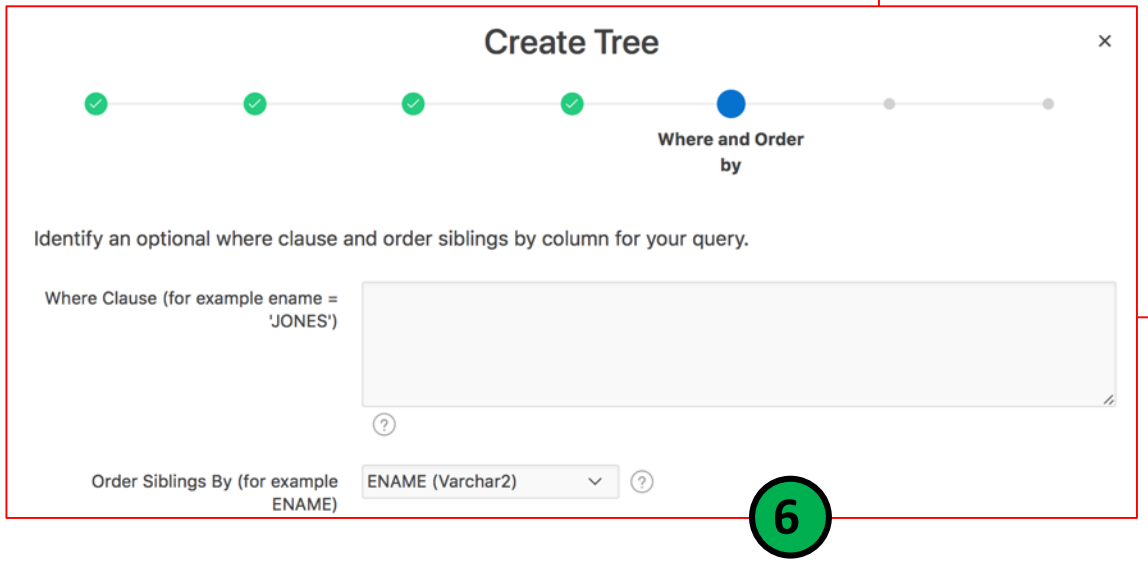
3



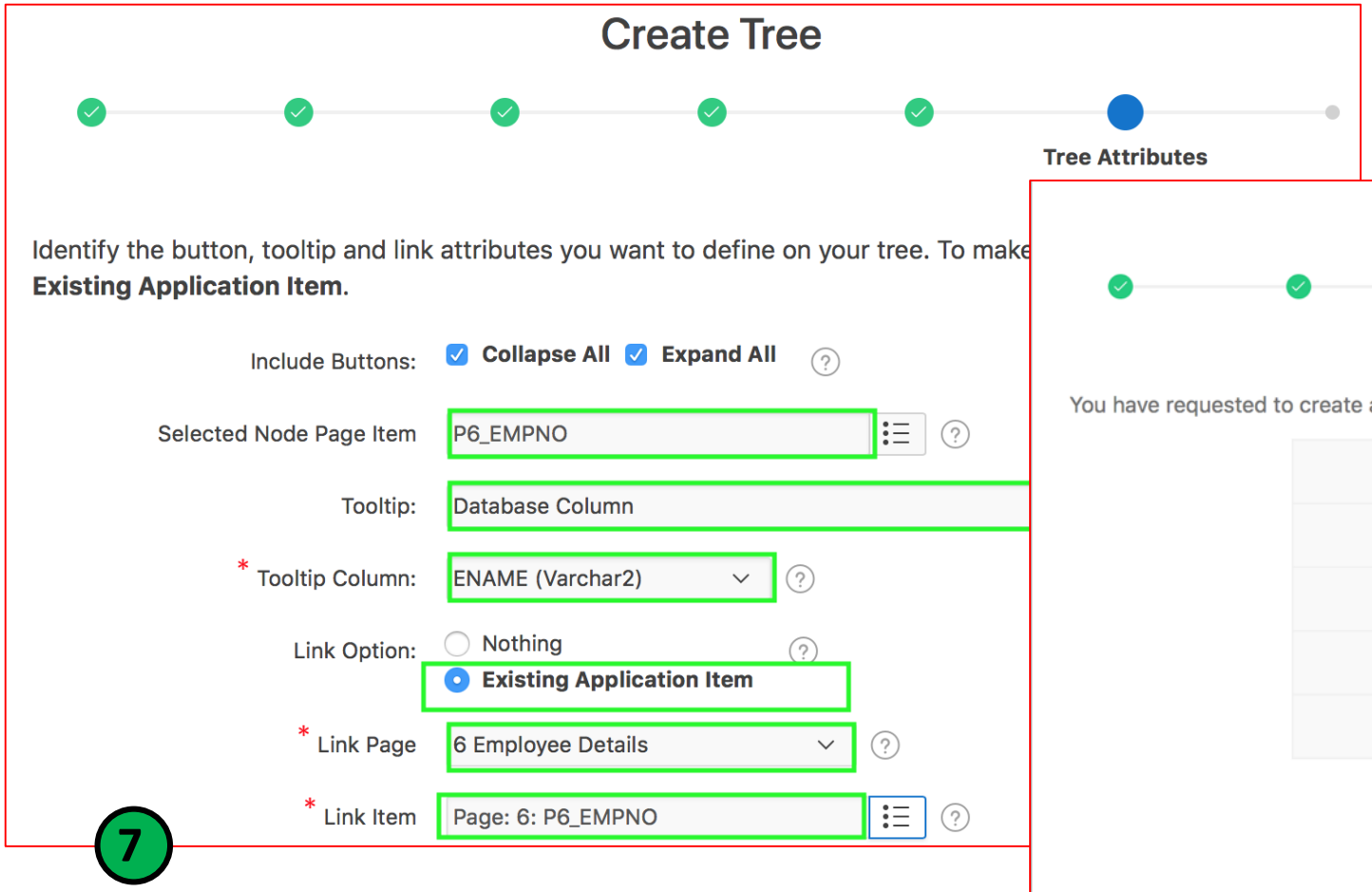
Creating a Tree: Example

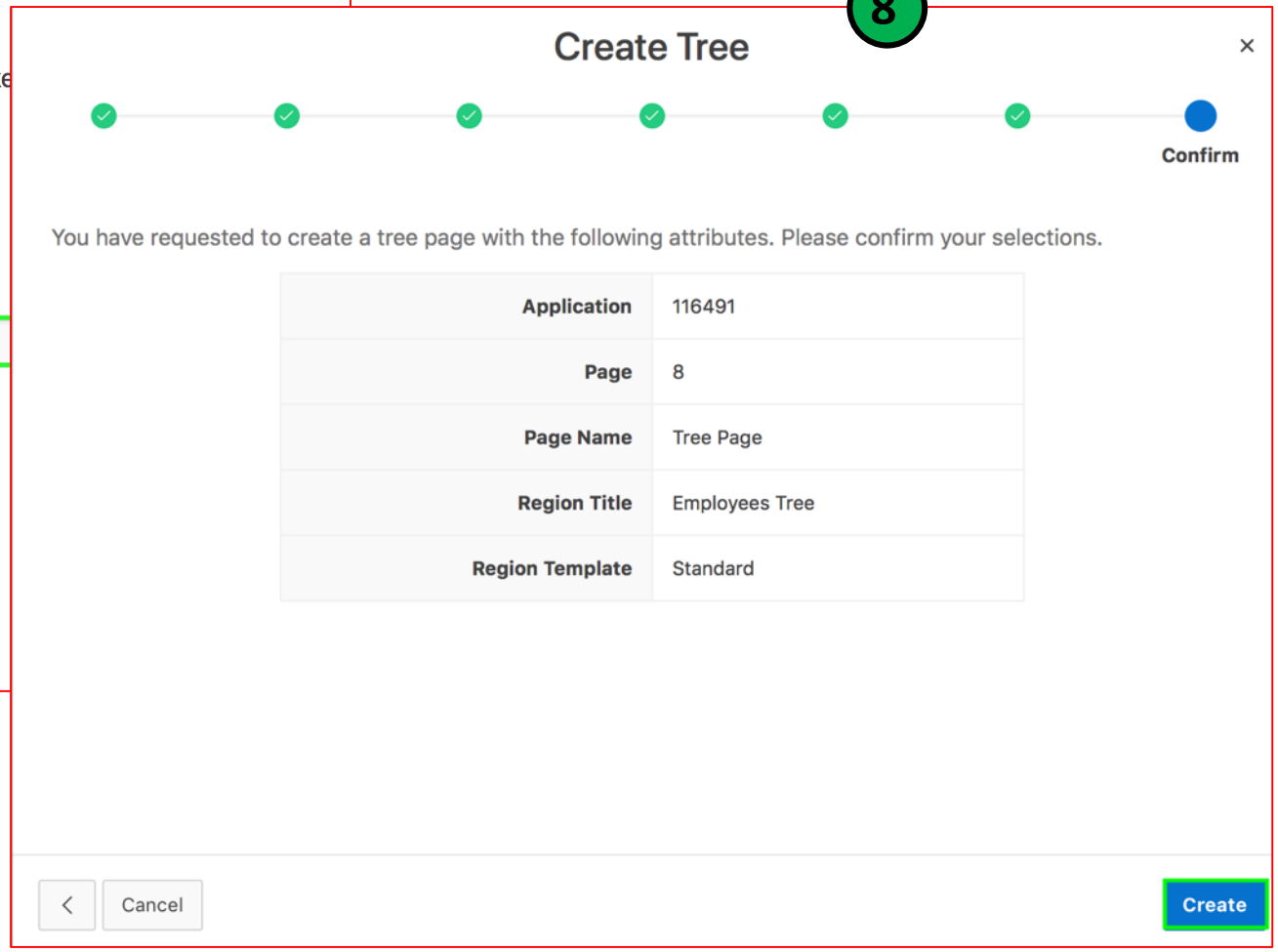




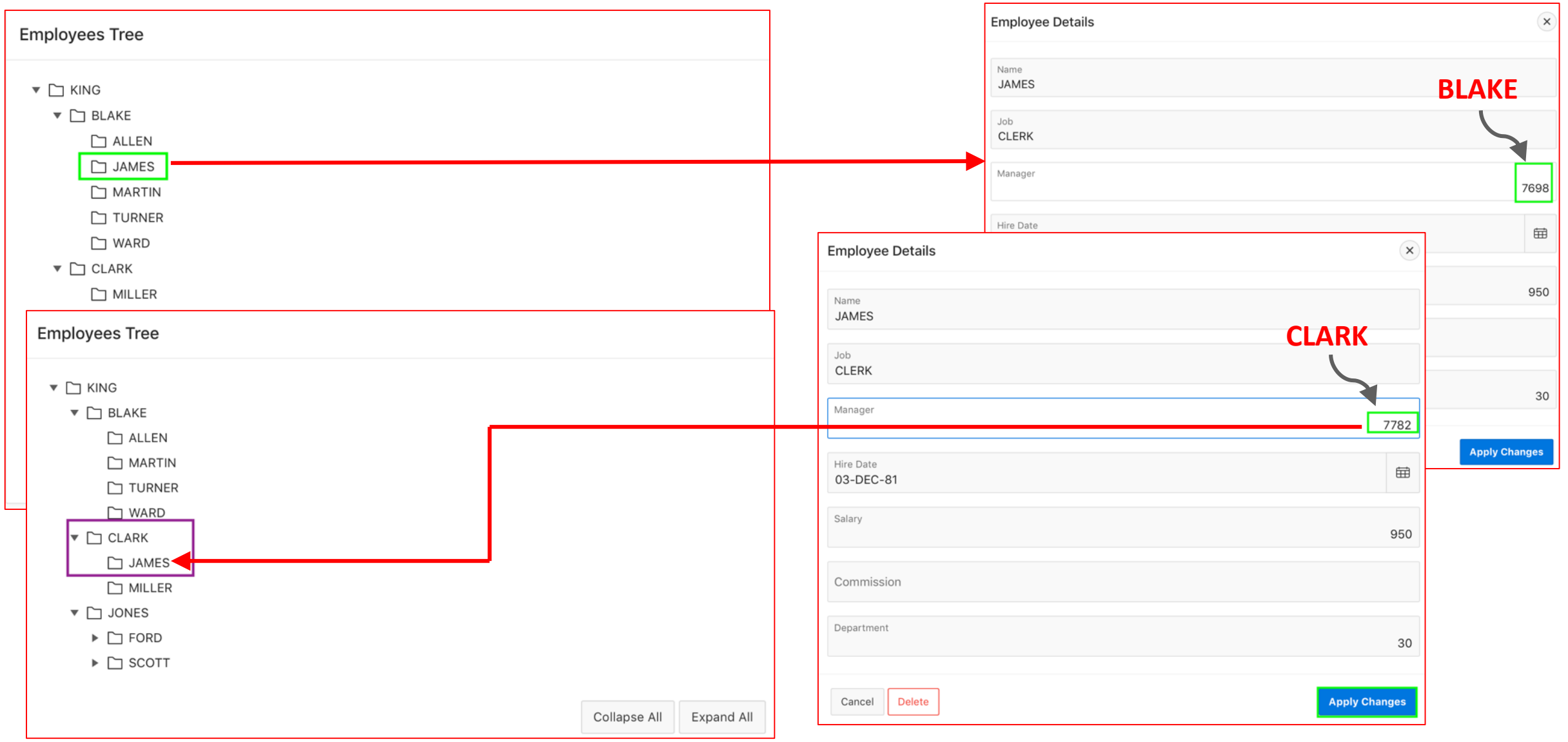


Creating a Tree: Example





Modifying a Tree



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

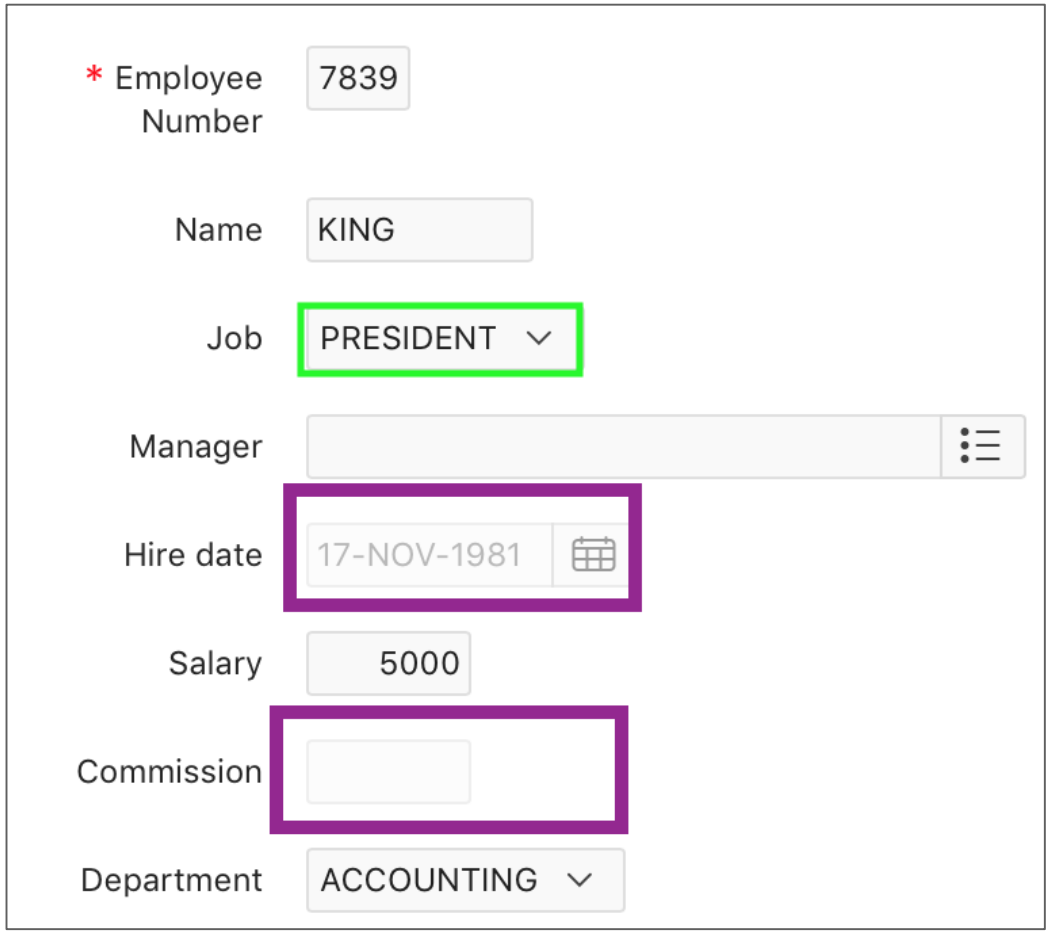
Lesson Objectives

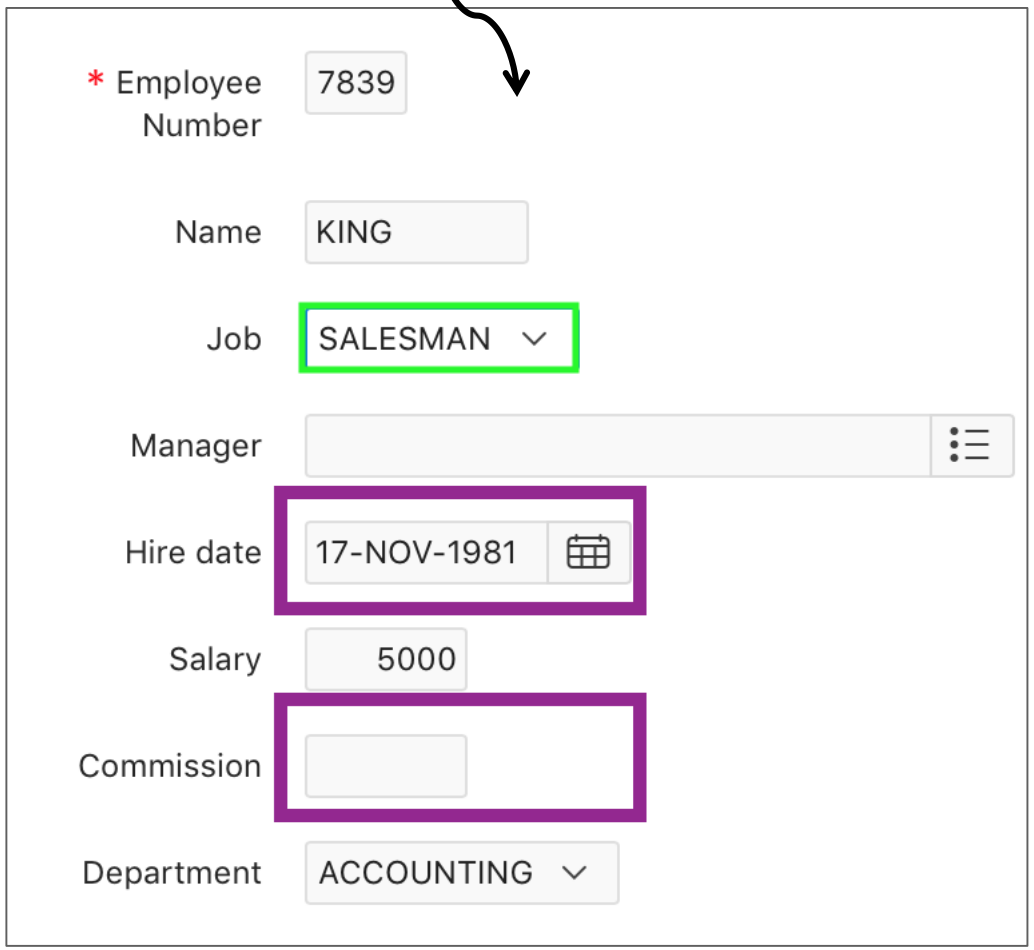
After completing this lesson, you should be able to:

- 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





Implementing a Dynamic Action

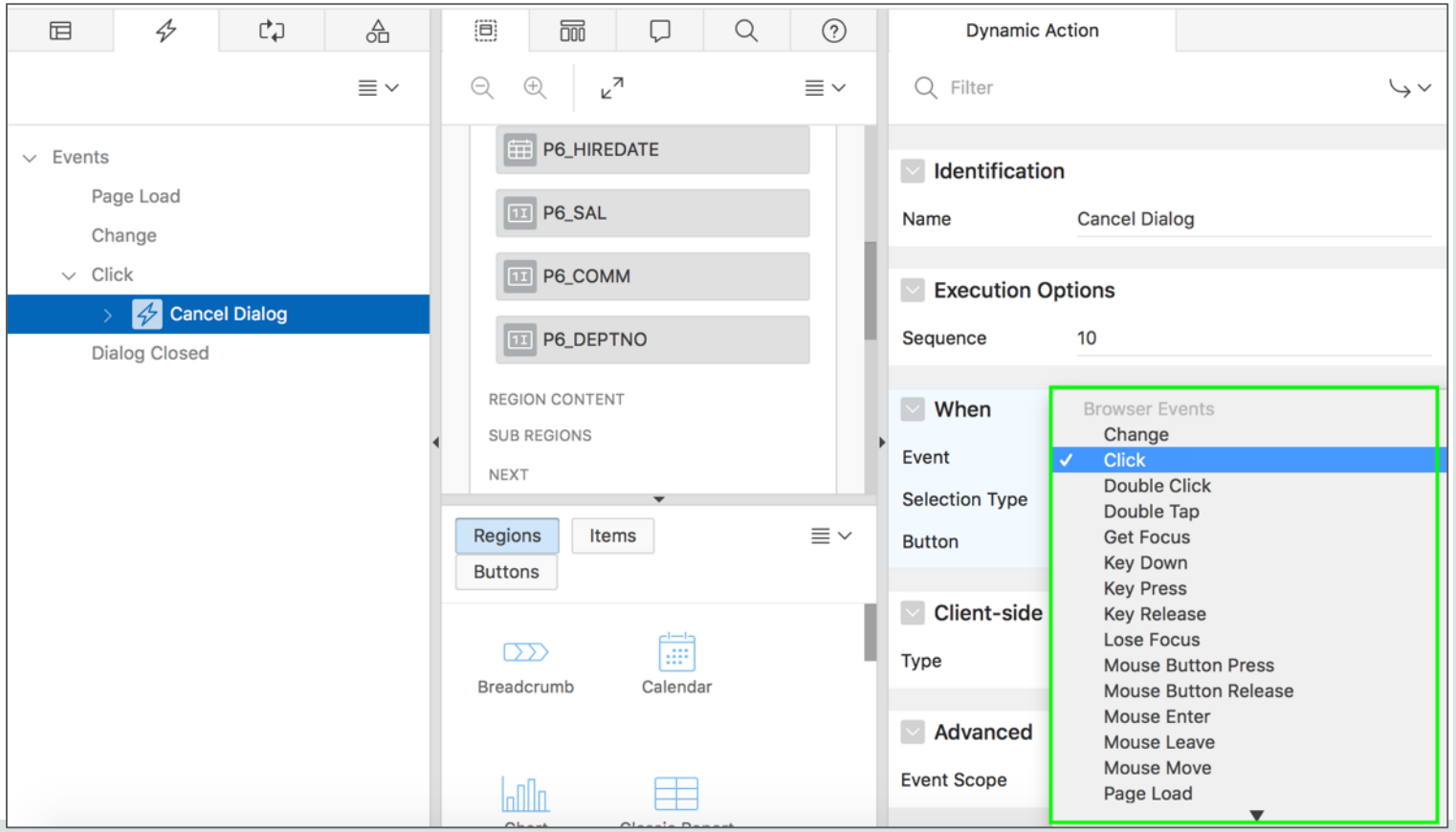
1. Create or edit an item, interactive grid column, button, region, JavaScript Expression, or jQuery selector on a page
2. Create a dynamic action from the application page that invokes the action
3. Run your application to test the dynamic action.

Creating a Dynamic Action: Steps

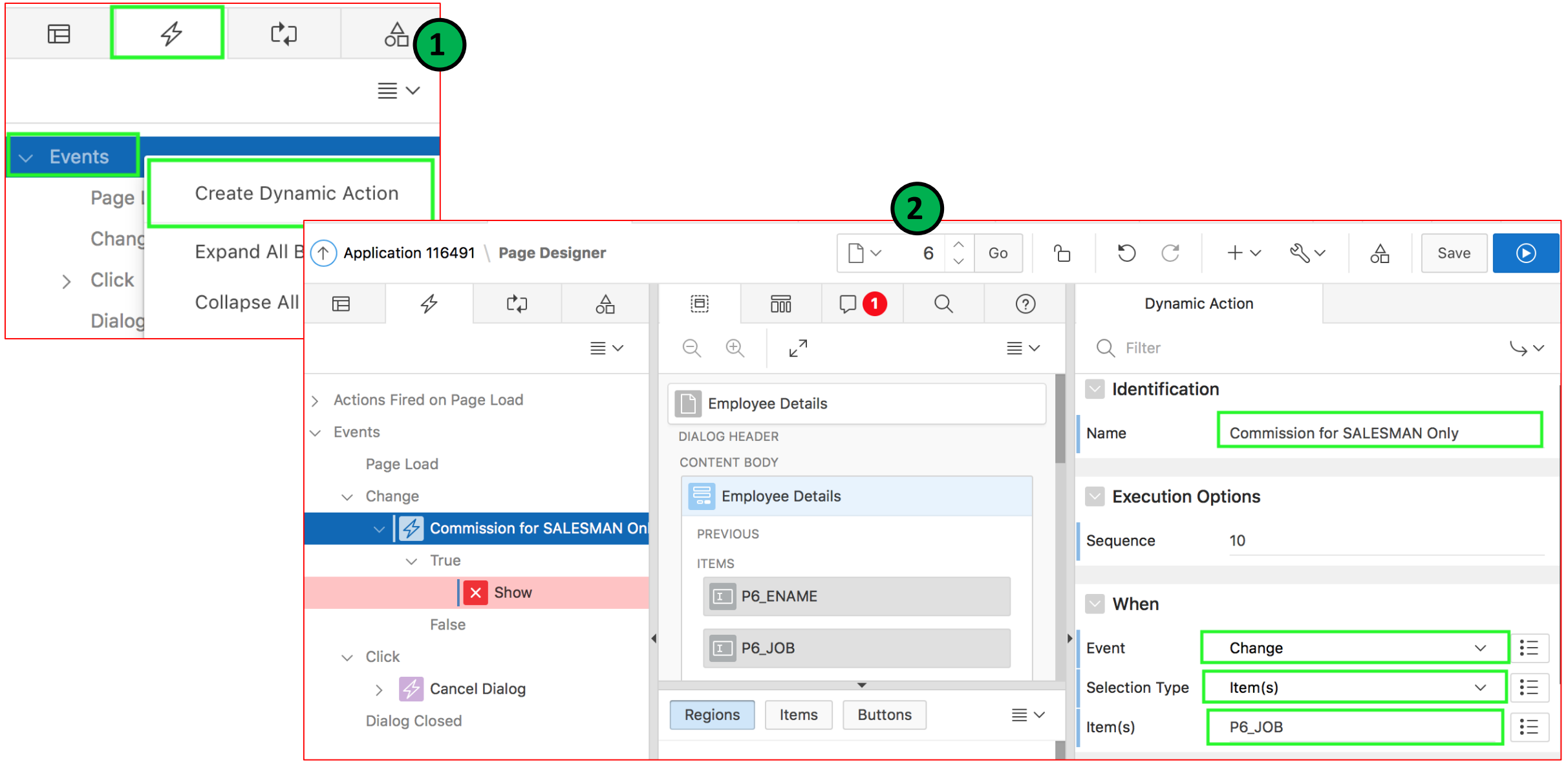
1. View the page to contain the item in Page Designer. Click the Dynamic Actions tab in the left pane.
2. Under Dynamic Actions, right-click Events and select Create Dynamic Action.
3. In the Property Editor, edit the following Dynamic Action attributes:
   - Identification > Name, Execution Options> 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

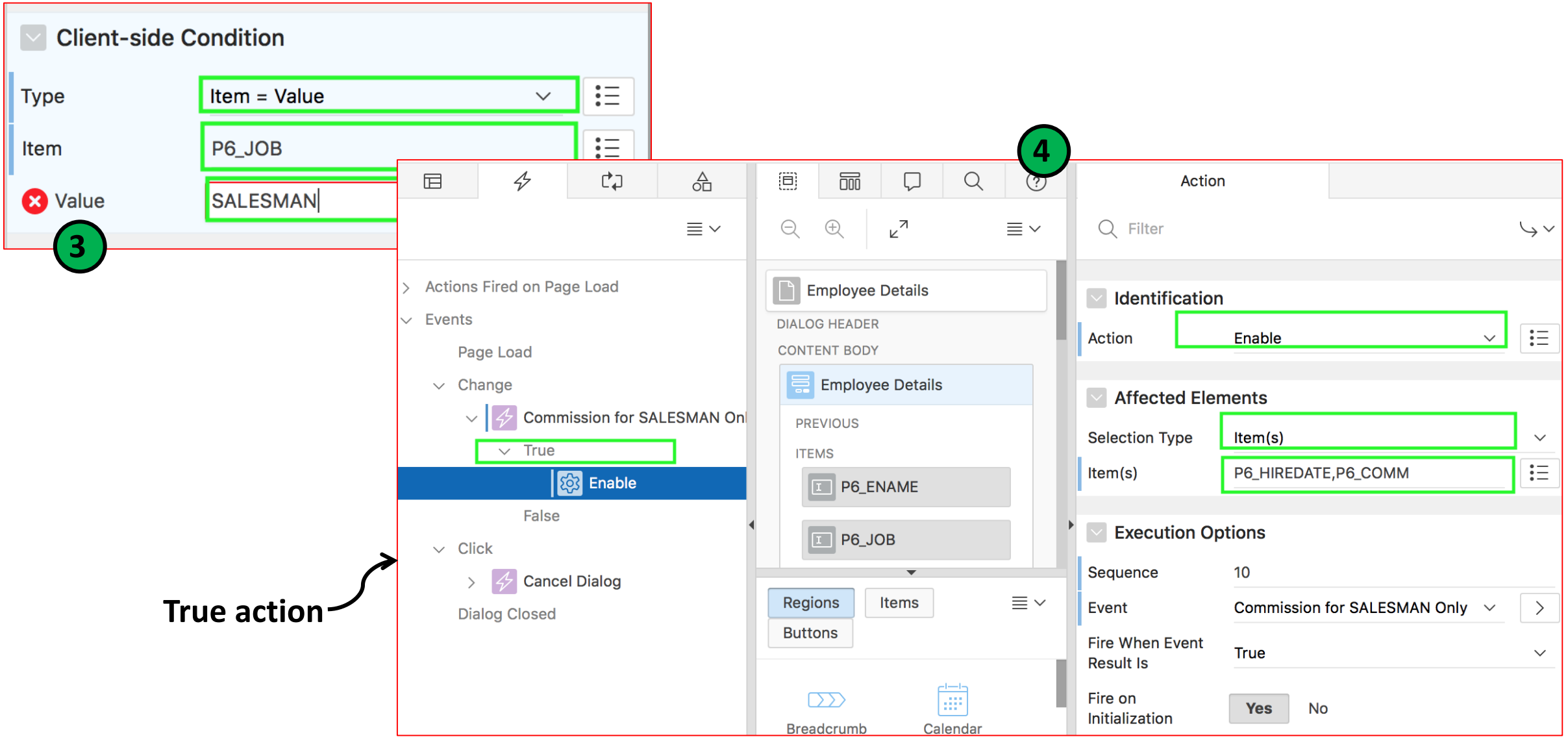
- You can define dynamic actions to fire based on events that happen on the page.
- Oracle Application Express includes four different categories of events:
  - Browser Events
  - Framework Events
  - Component Events
  - Custom Events



Enabling and Disabling Items: Steps

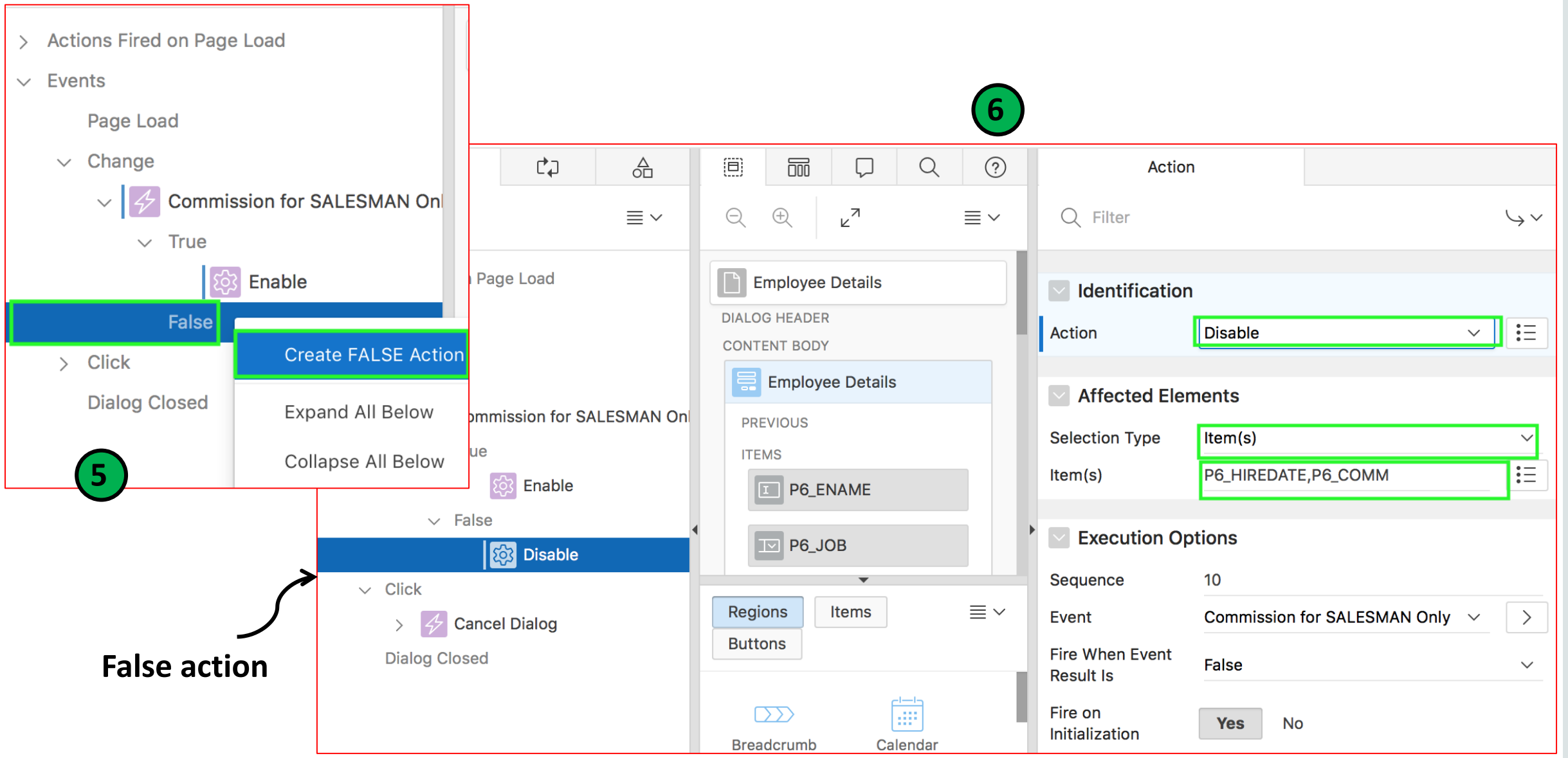


Enabling and Disabling Items: Steps

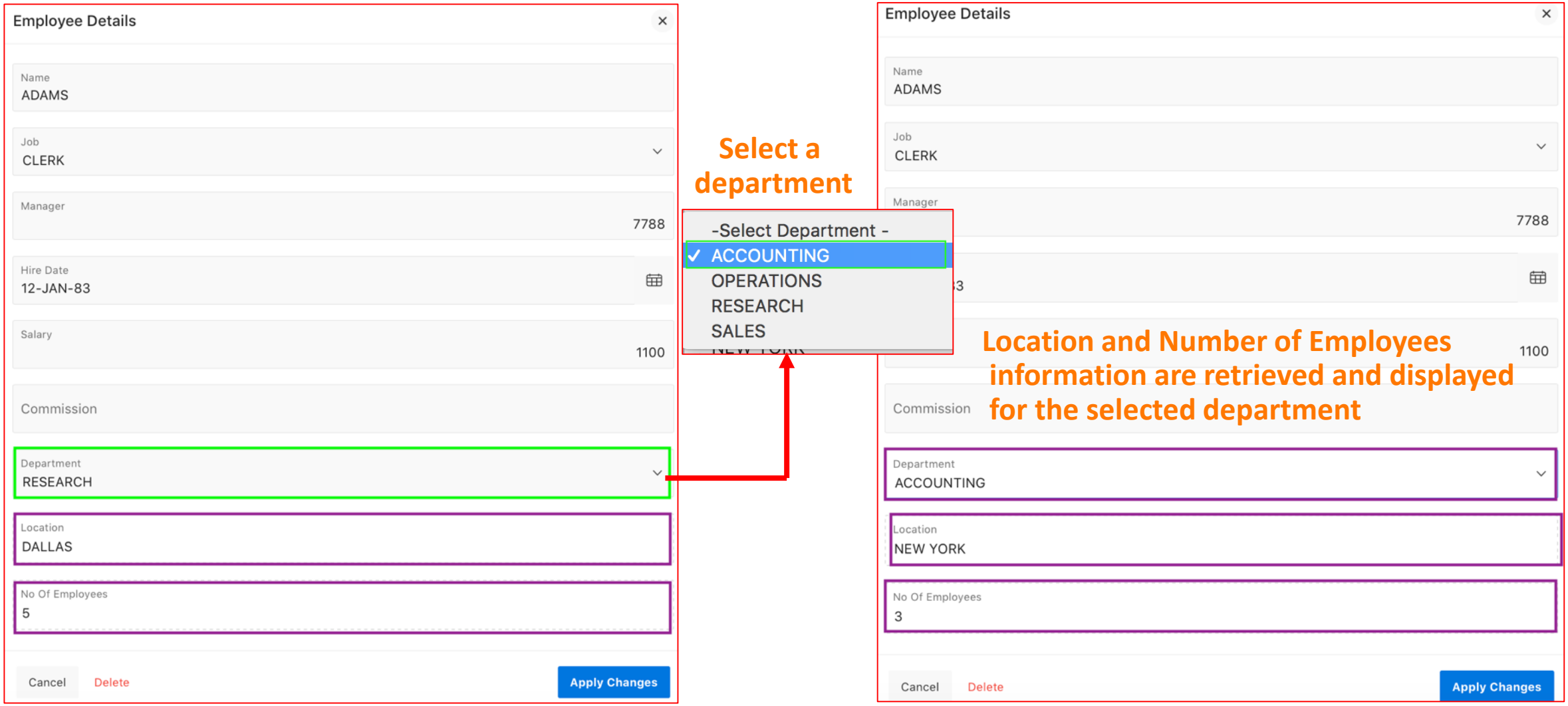


True action

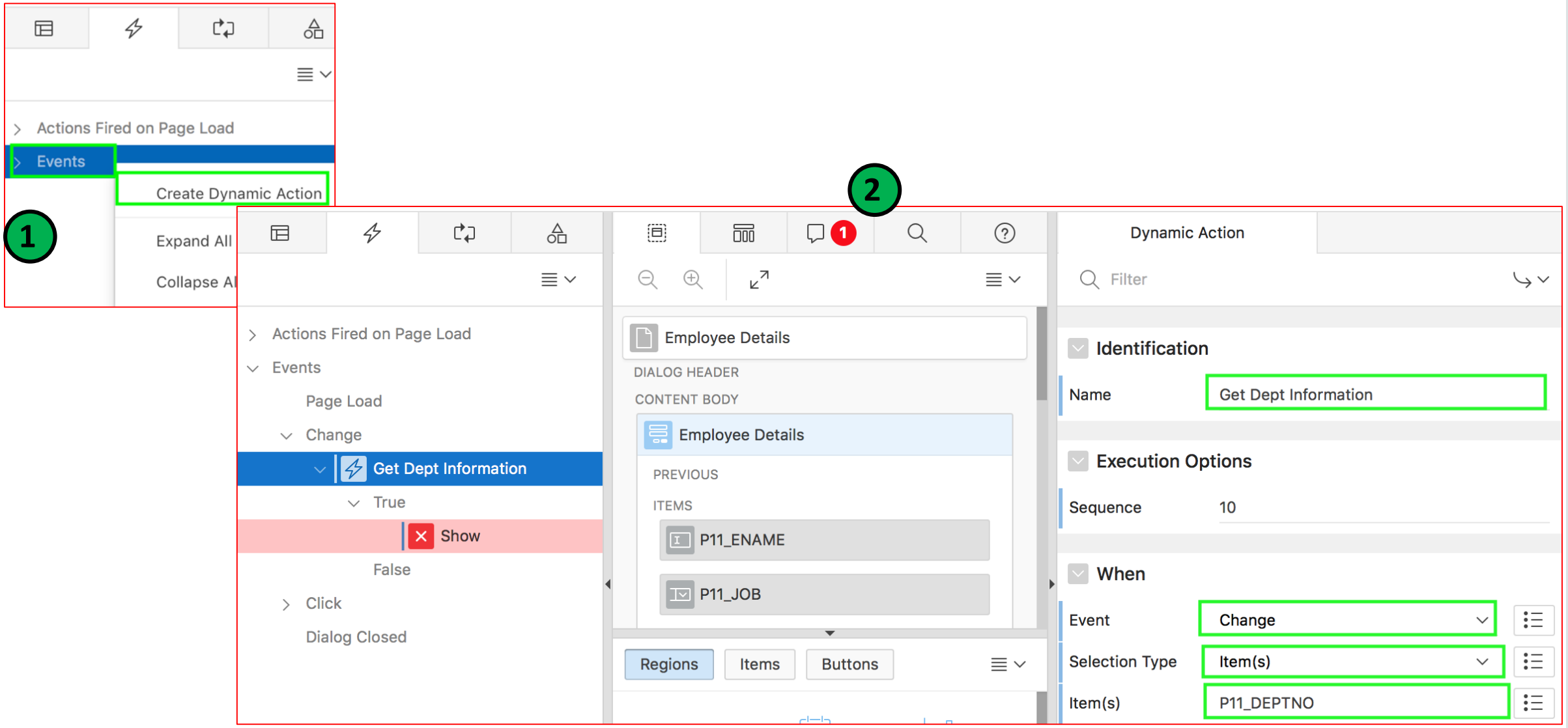
Enabling and Disabling Items: Steps



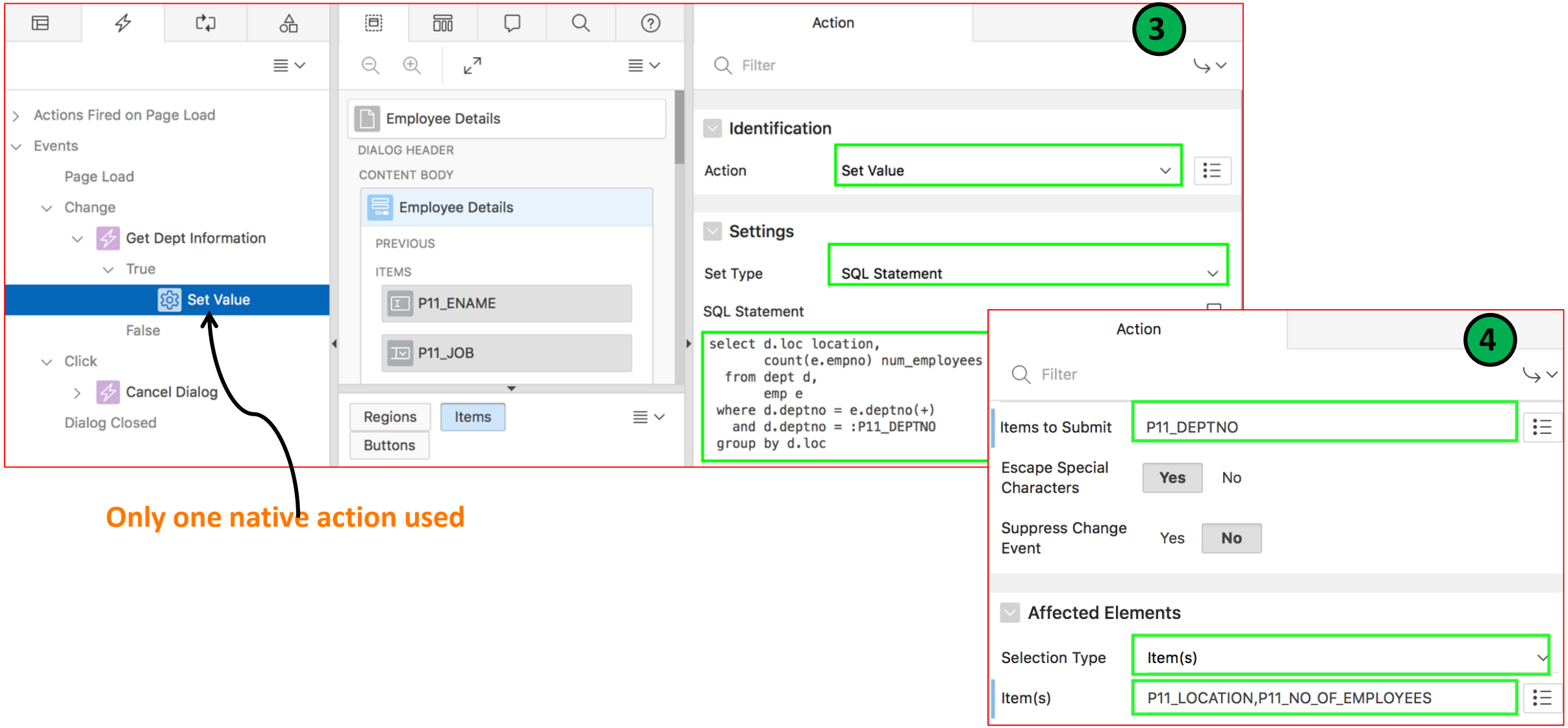
Setting the Value of Page Items: Set Value Example



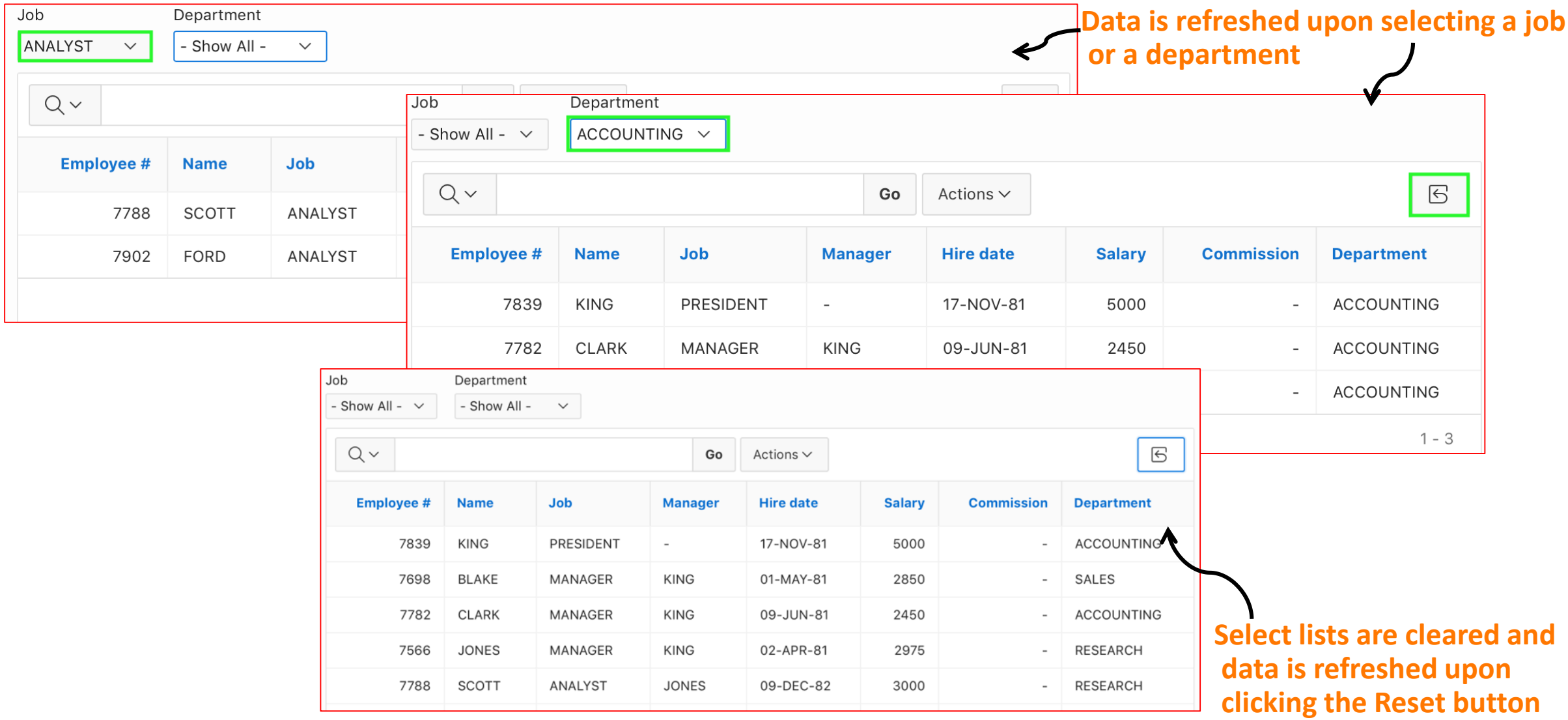
Setting the Value of Page Items: Steps



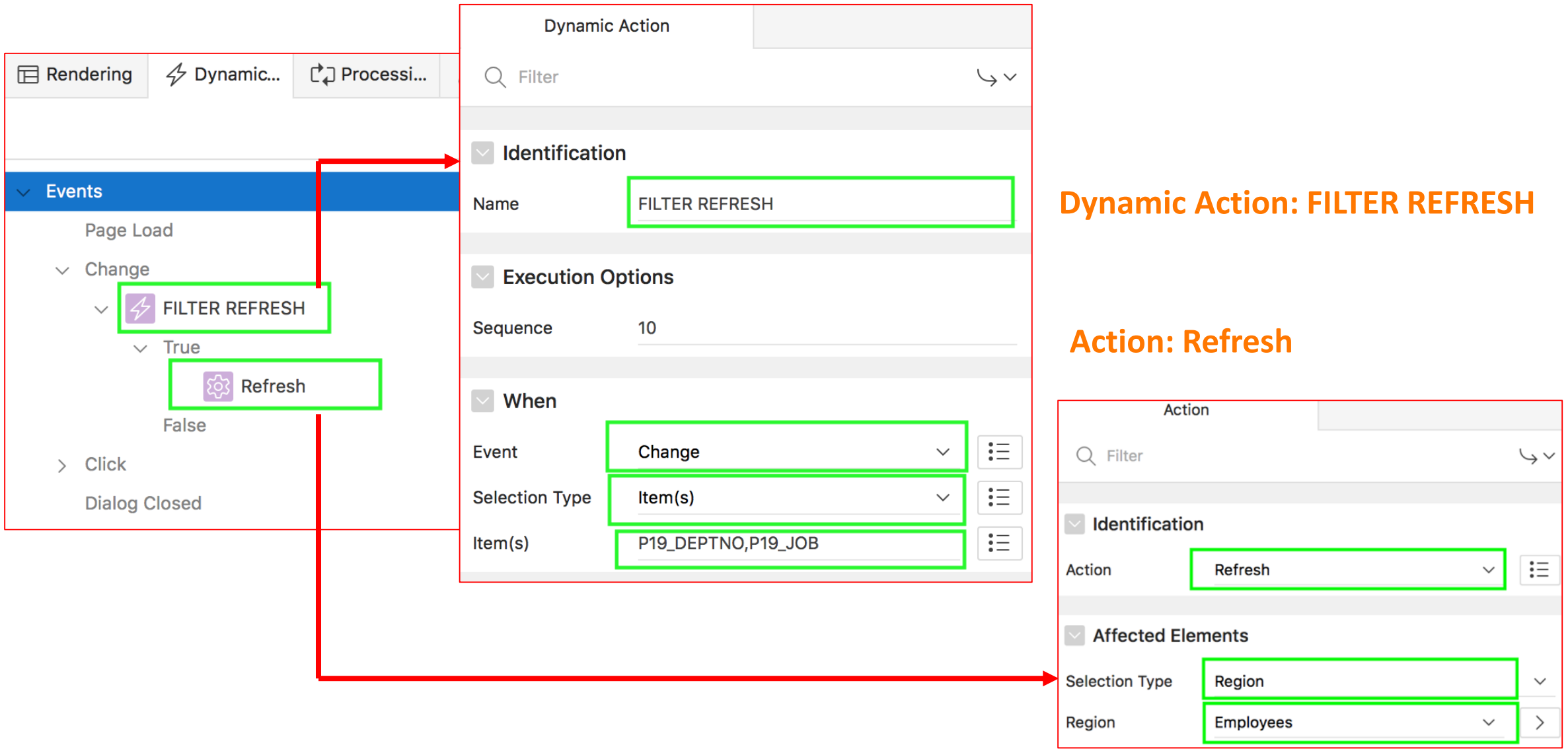
Setting the Value of Page Items: Steps



Filtering and Refreshing a Report: Example

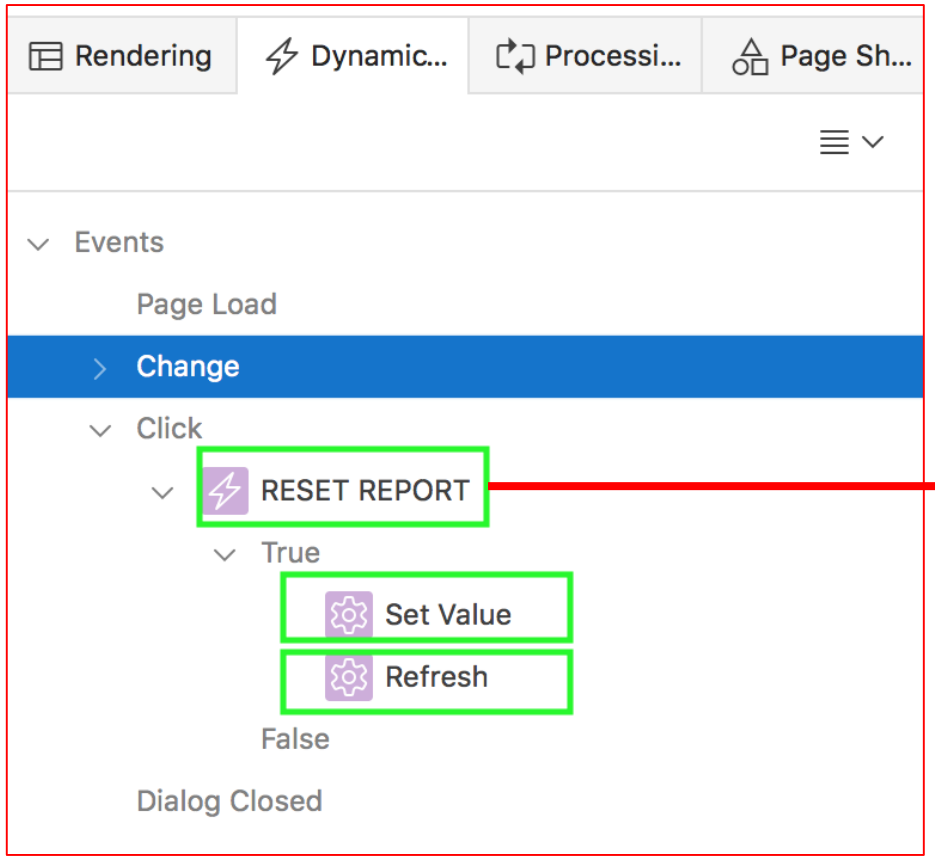


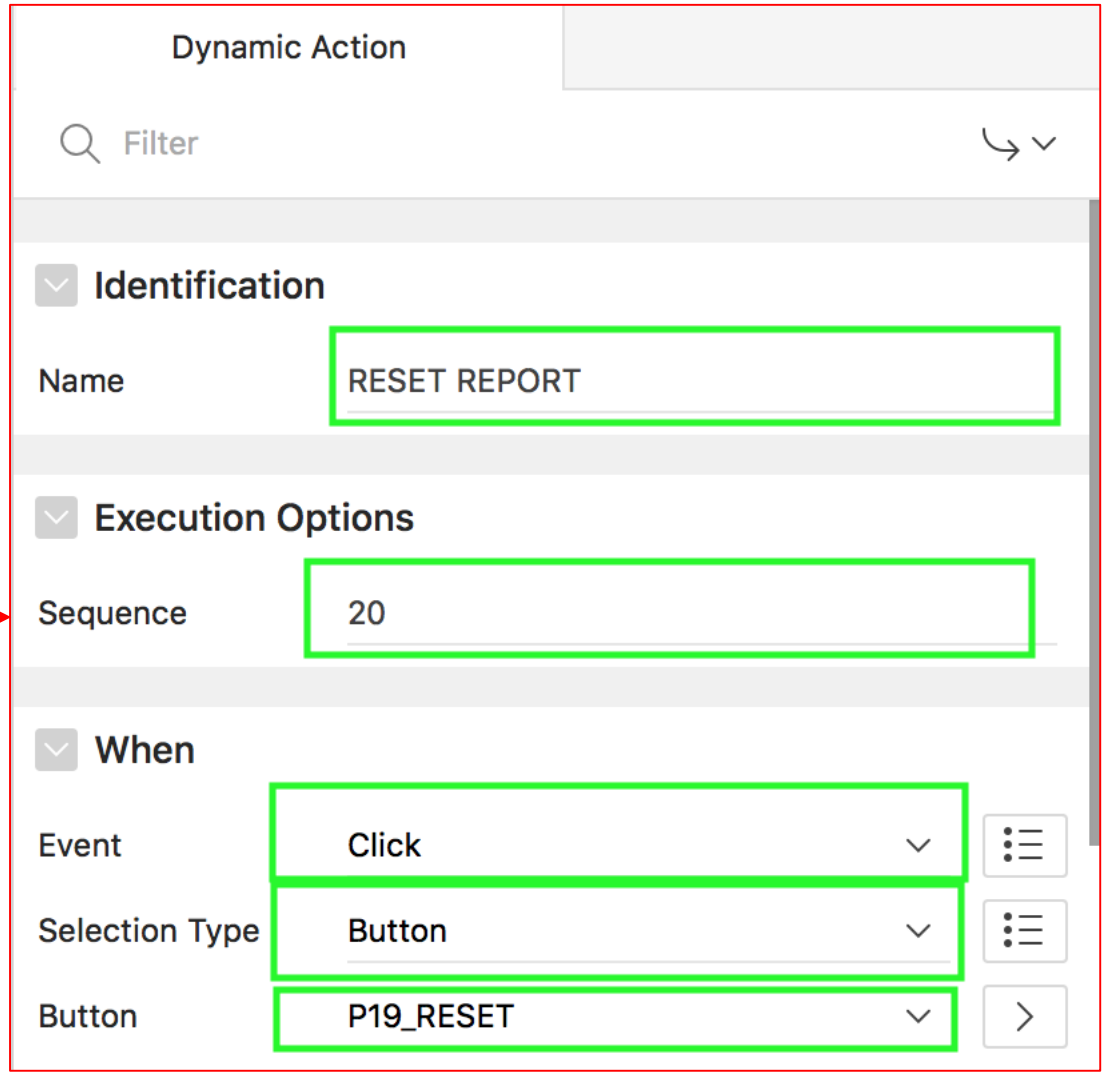
Filtering and Refreshing a Report: Steps



Filtering and Refreshing a Report: Steps

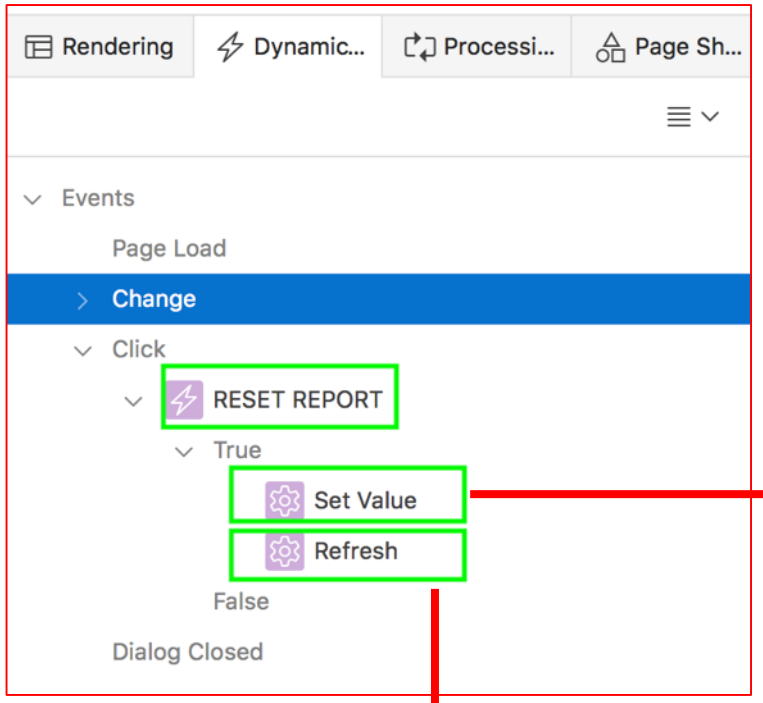
Dynamic Action: RESET REPORT

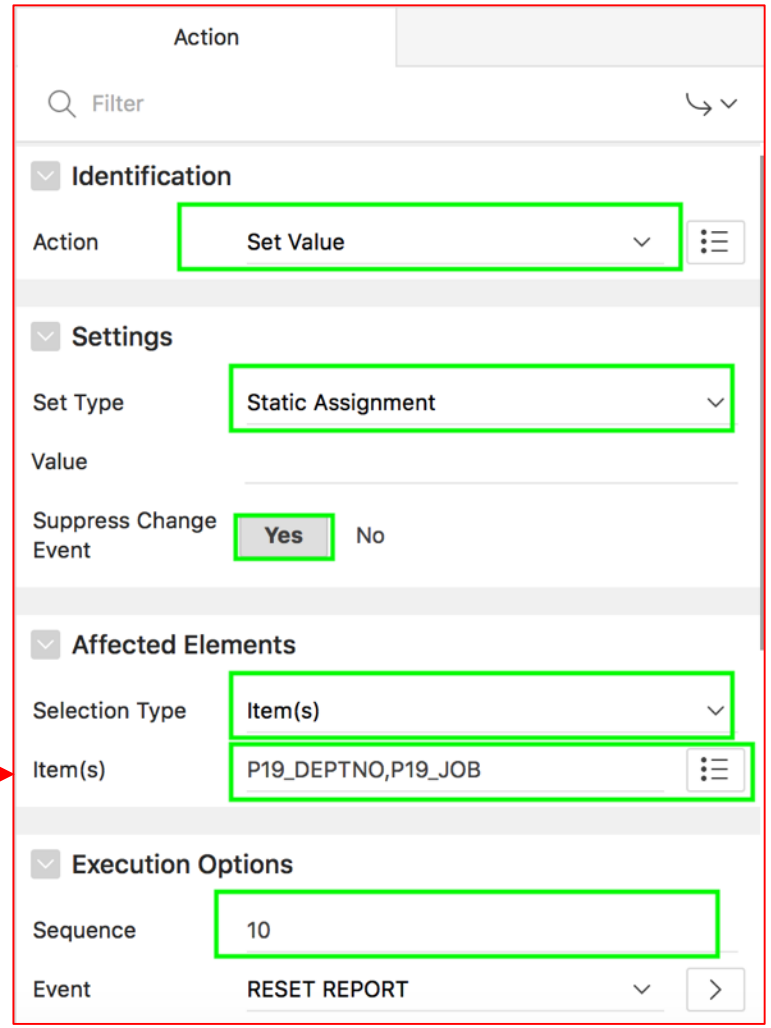


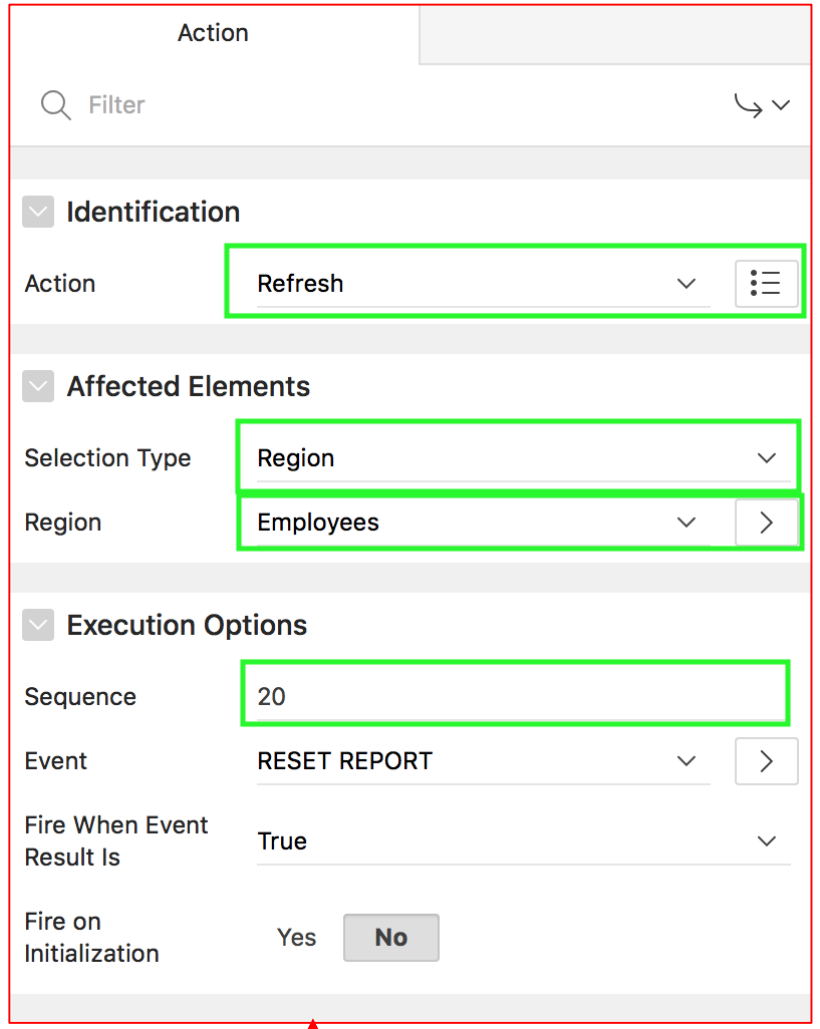


Filtering and Refreshing a Report: Steps

Actions: Set Value, and Refresh

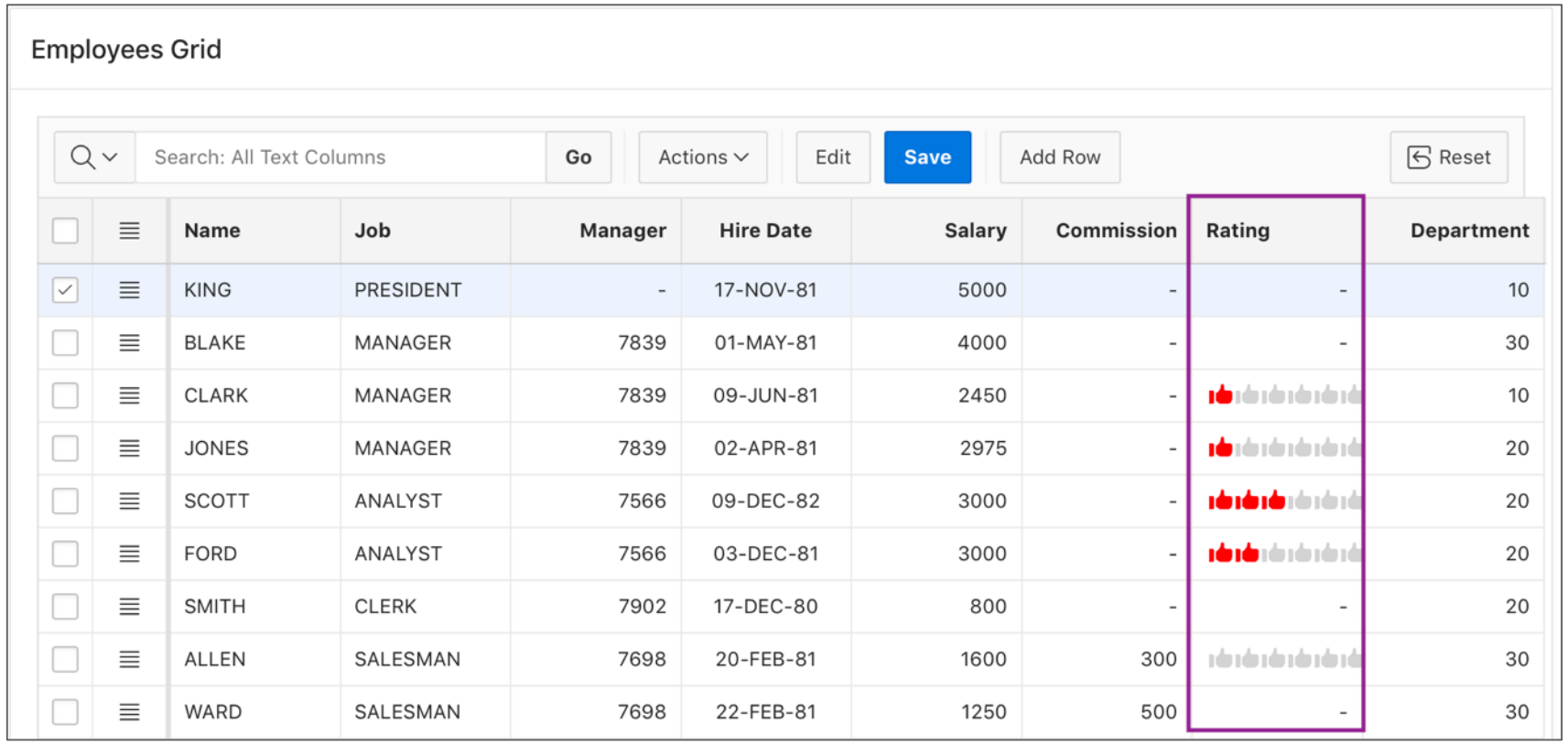






What is a Plug-in?

- 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



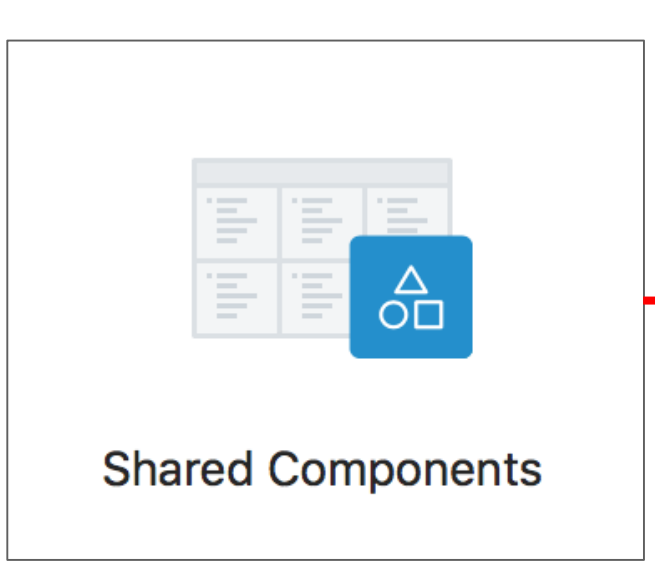
Example of implementing the Star Rating Plug-In

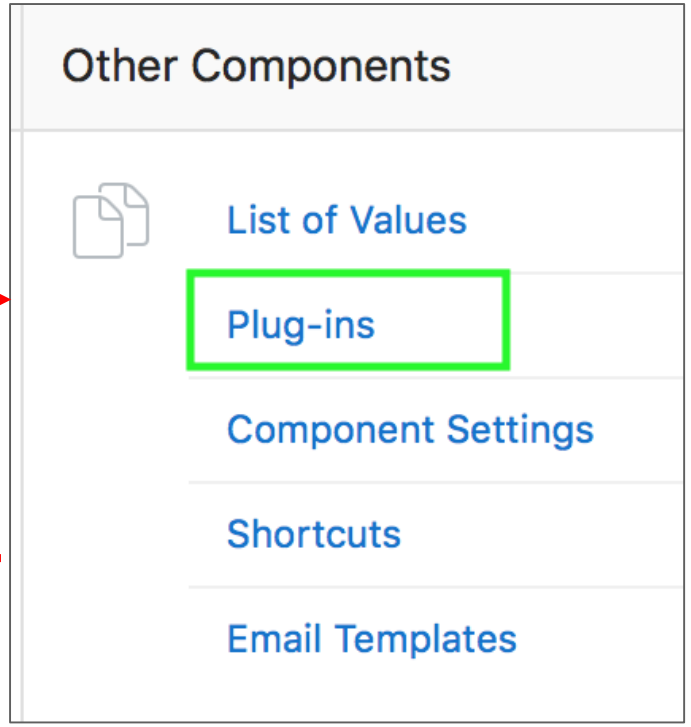
Implementing a Plug-in in your Application

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

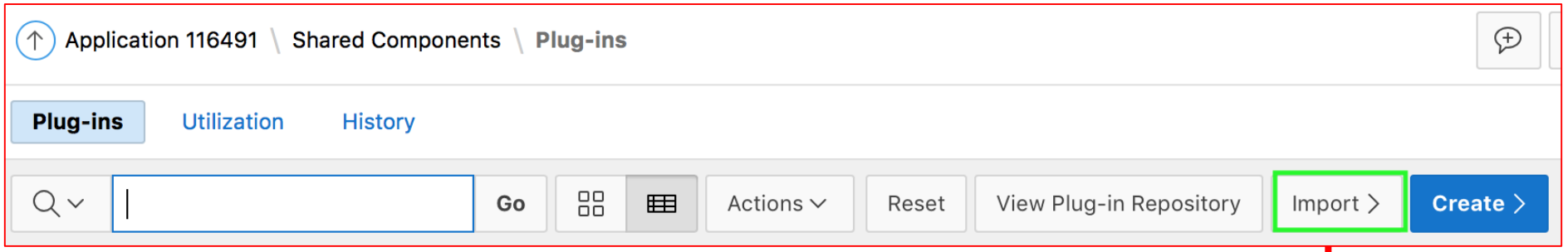
Viewing the Plug-in Repository

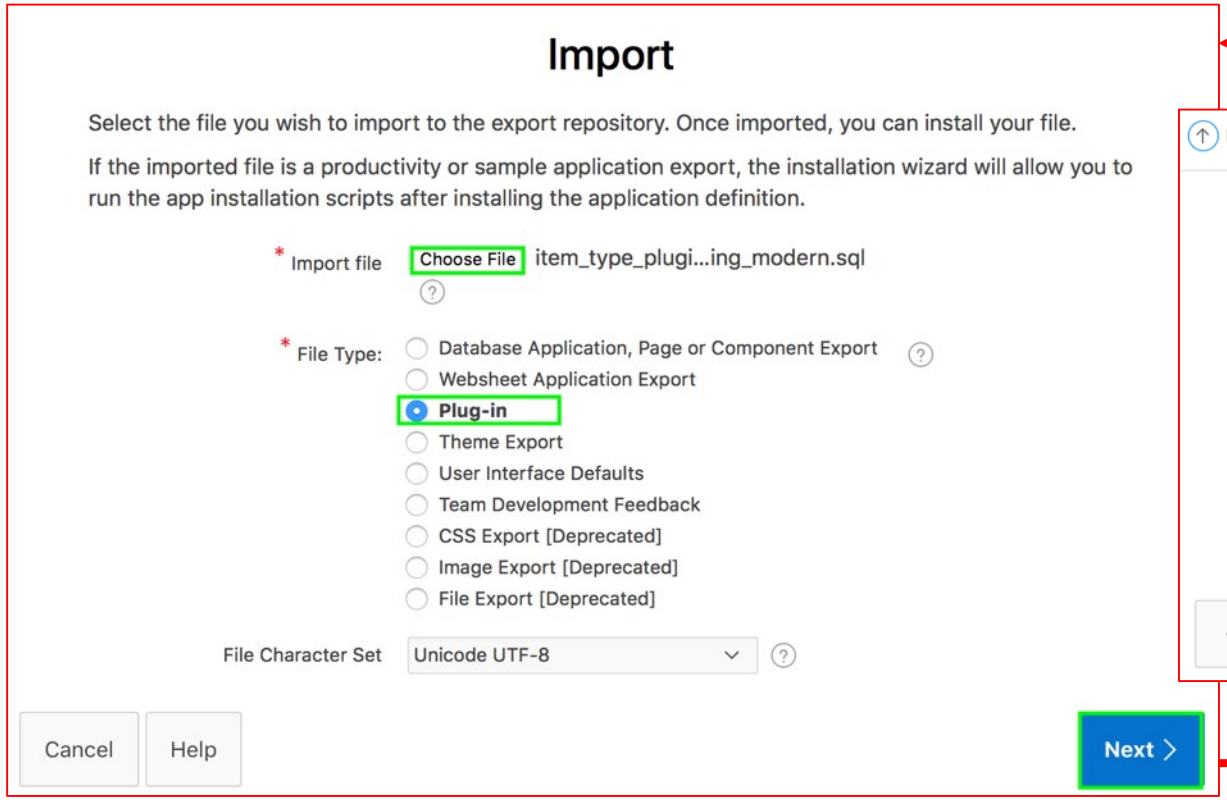


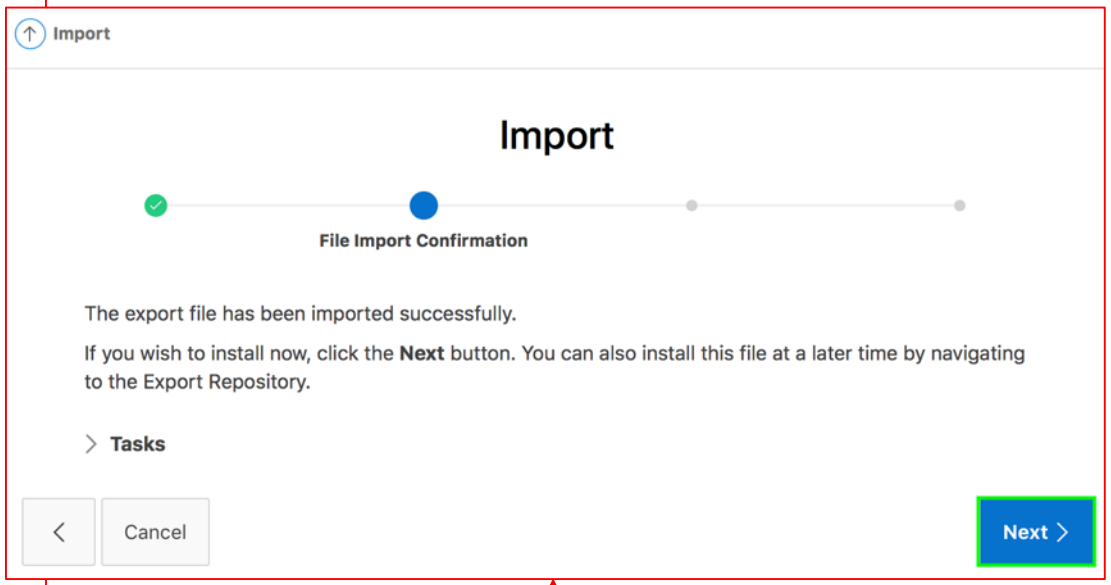


https://apex.oracle.com/plugins

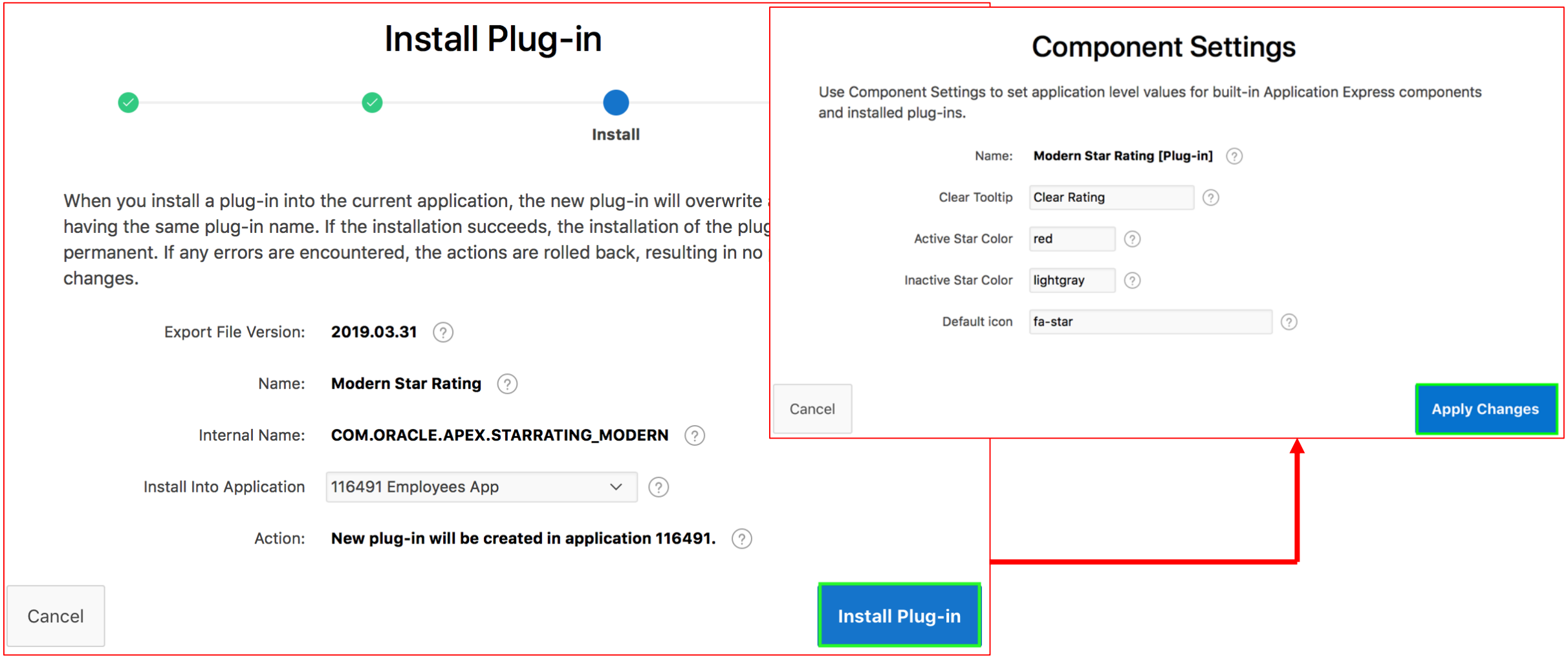
Importing and Installing a Plug-in



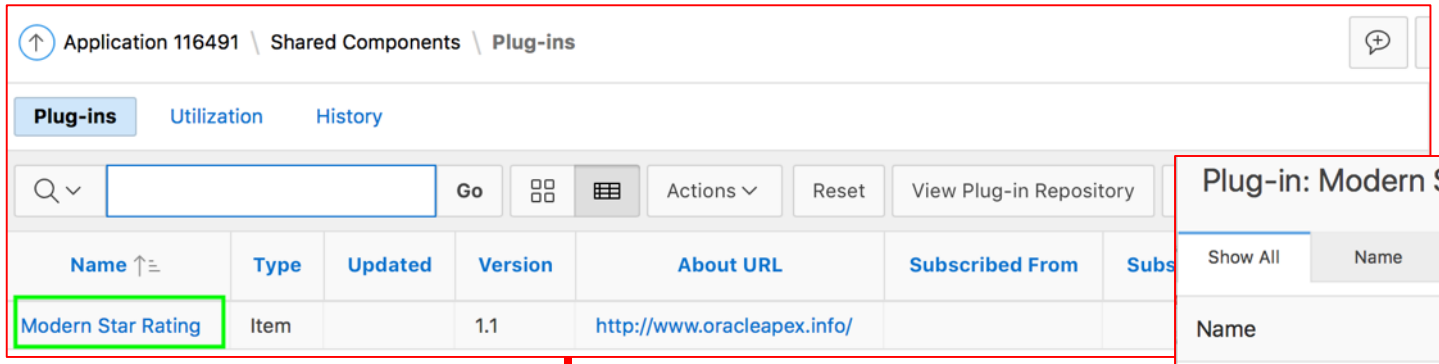


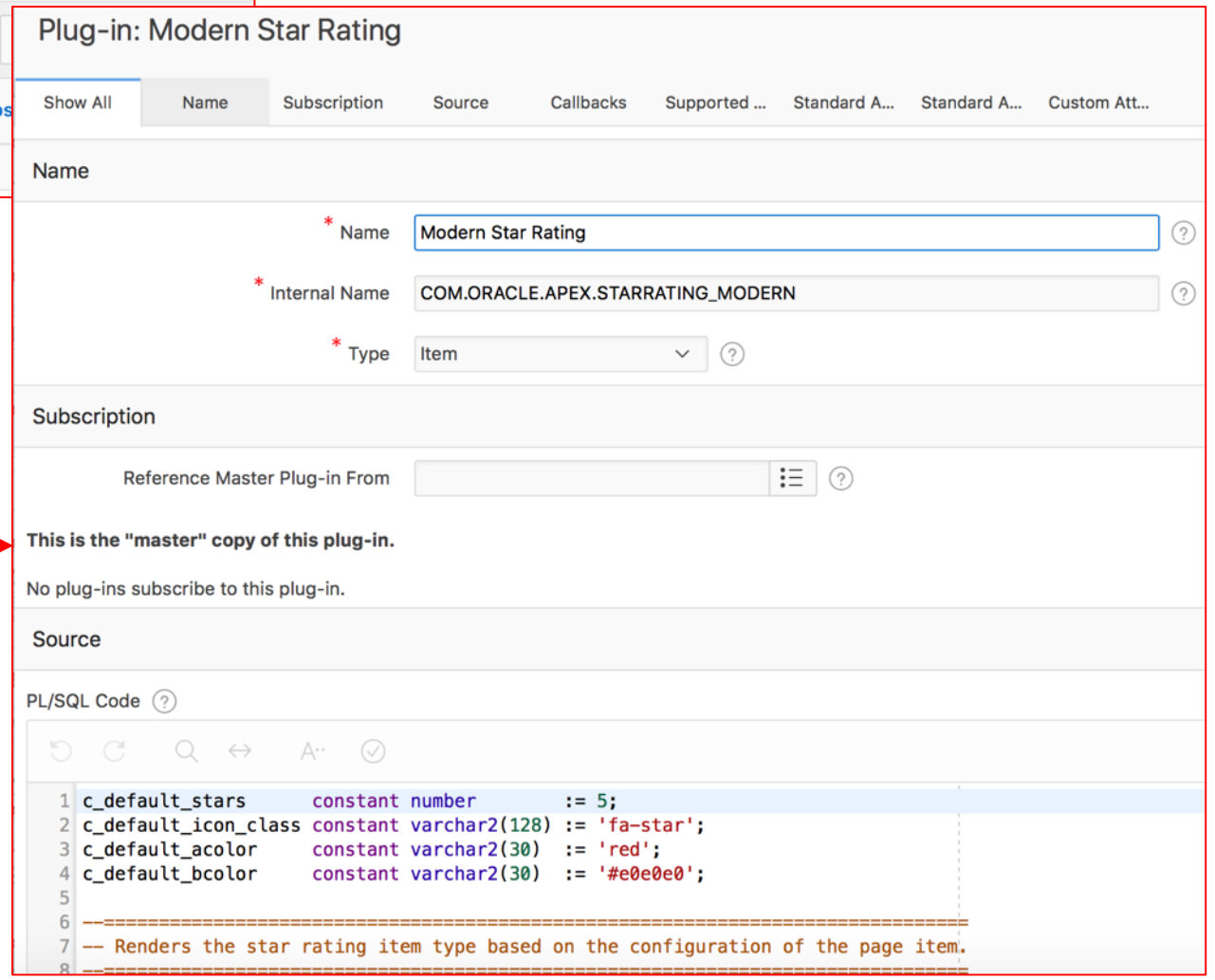


Importing and Installing a Plug-in



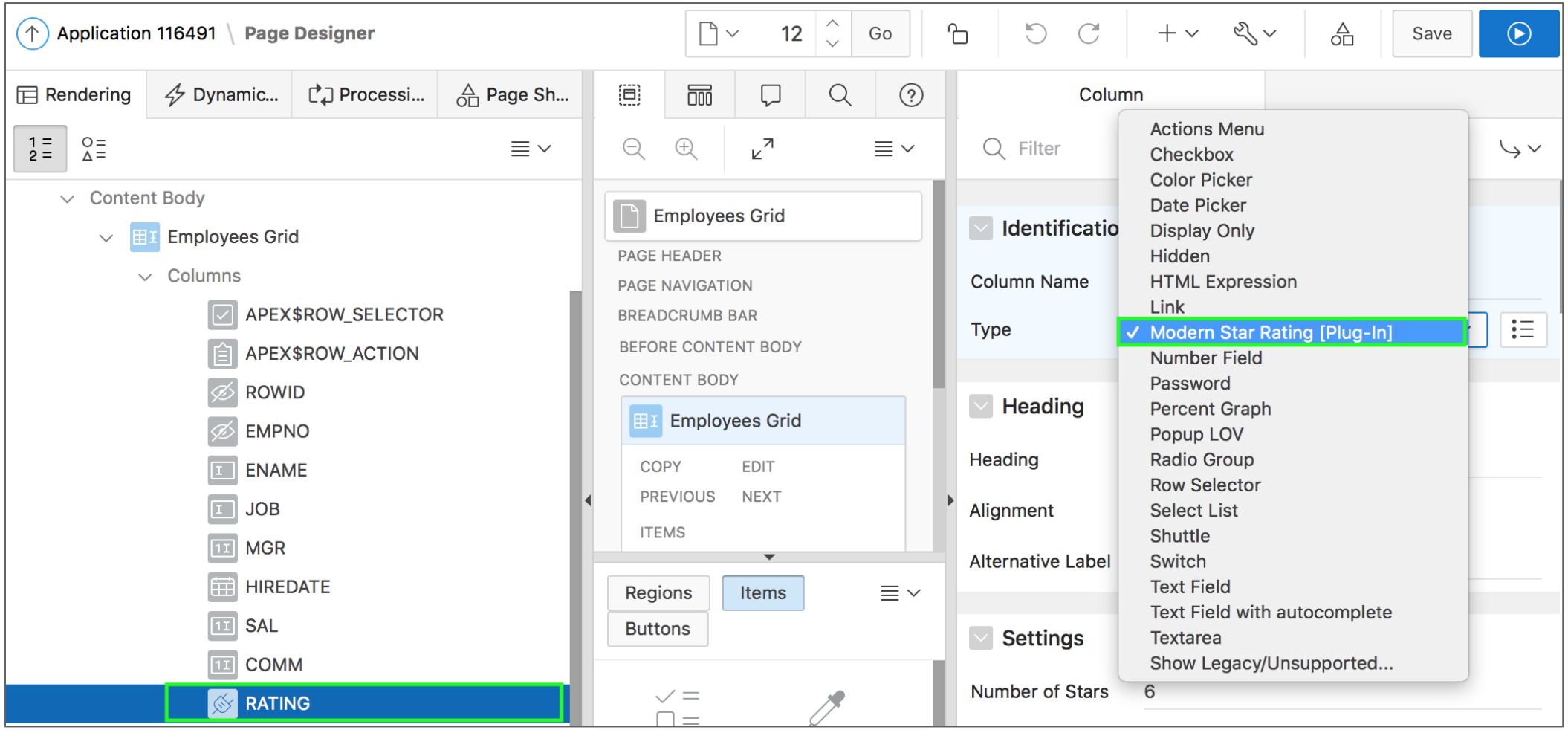
Viewing a Plug-in Definition



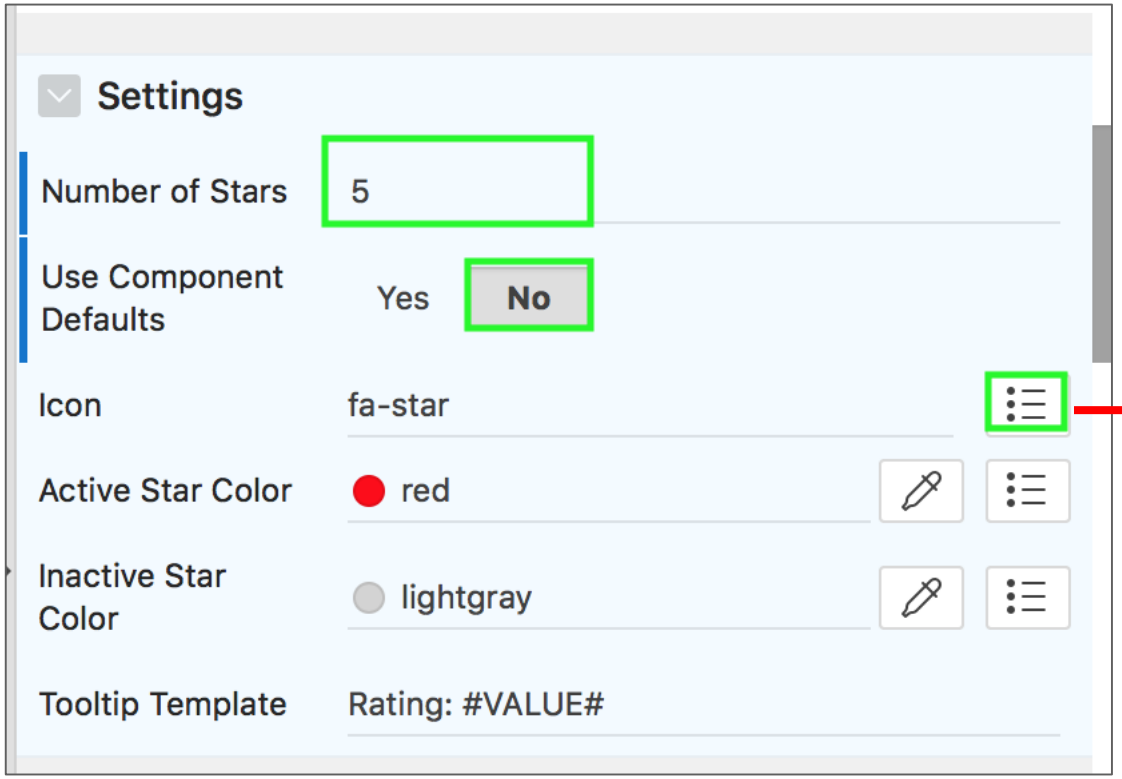


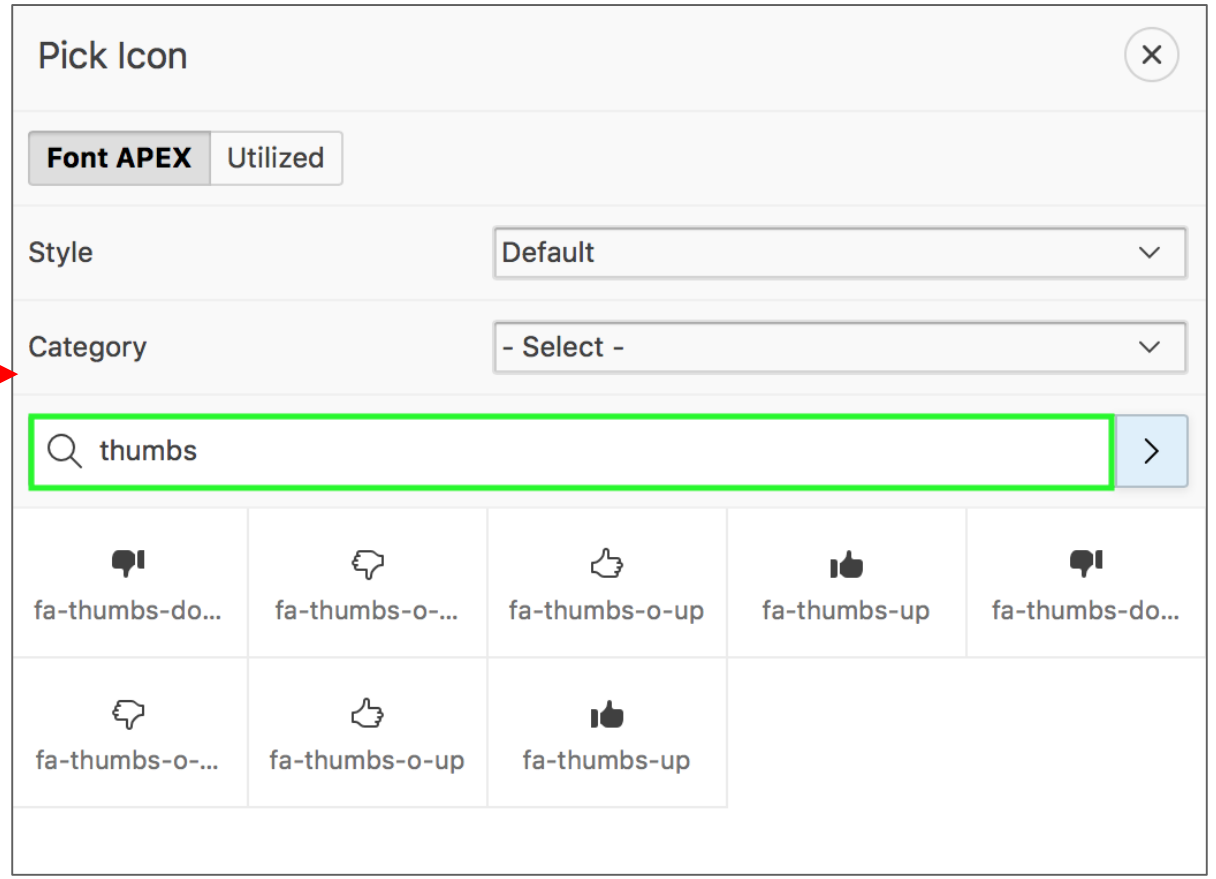
Using the Plug-in in your Application

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



Using the Plug-in in your Application





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:

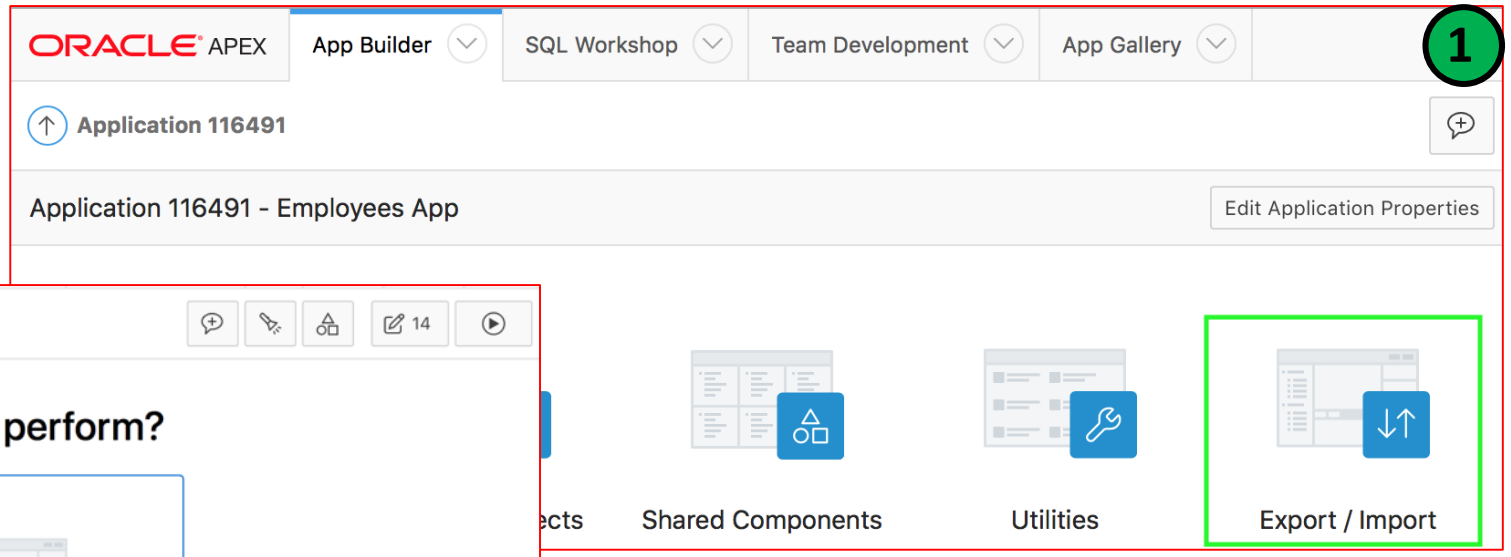
- Describe the steps involved in migrating an application between environments
- Export the application definition, underlying database objects, and data from your development environment
- Import the application definition, underlying database objects, and data in the target development environment

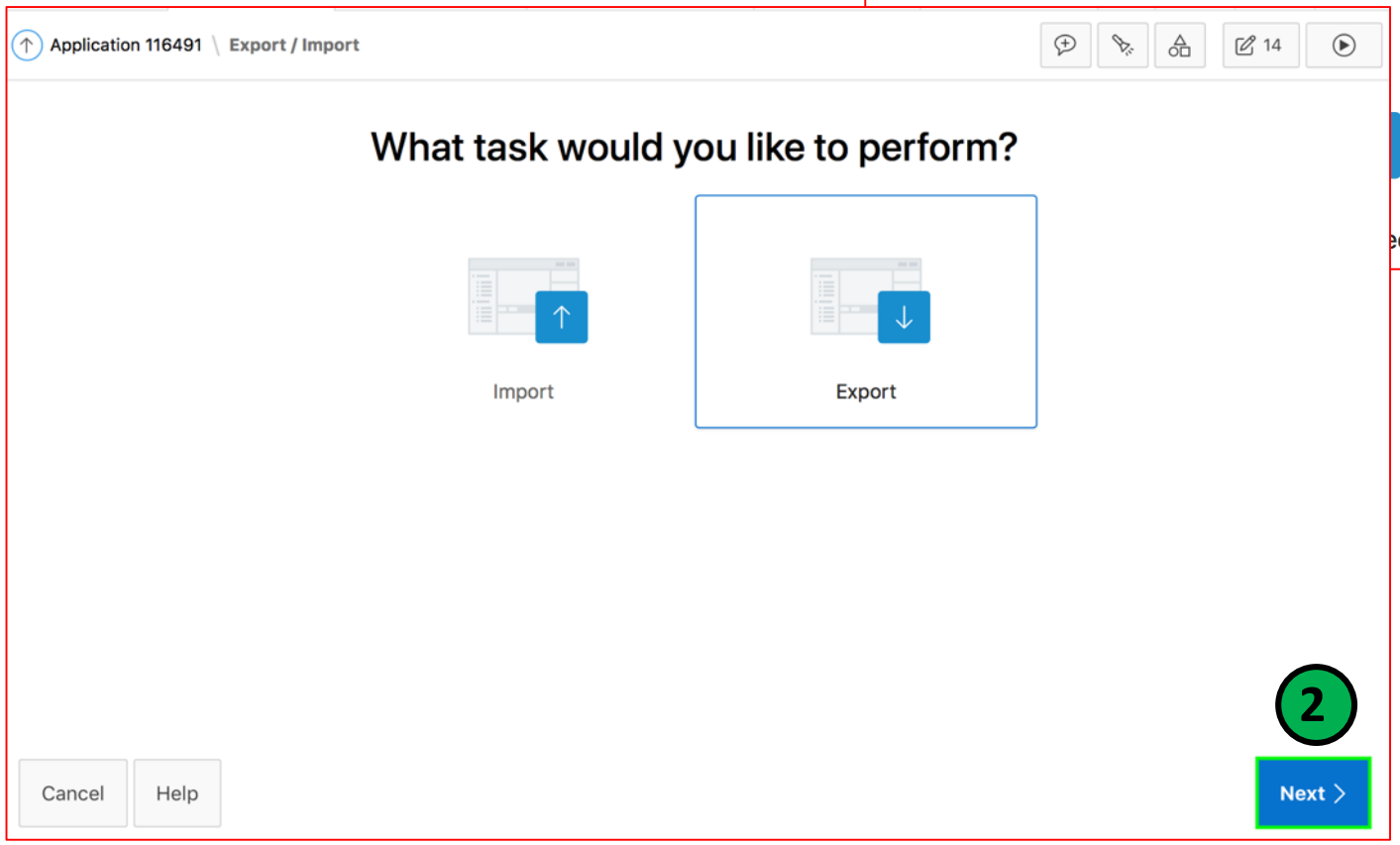
Migrating your Application Development Between Environments: Overview

Migrating your Application Development Between Environments: Steps

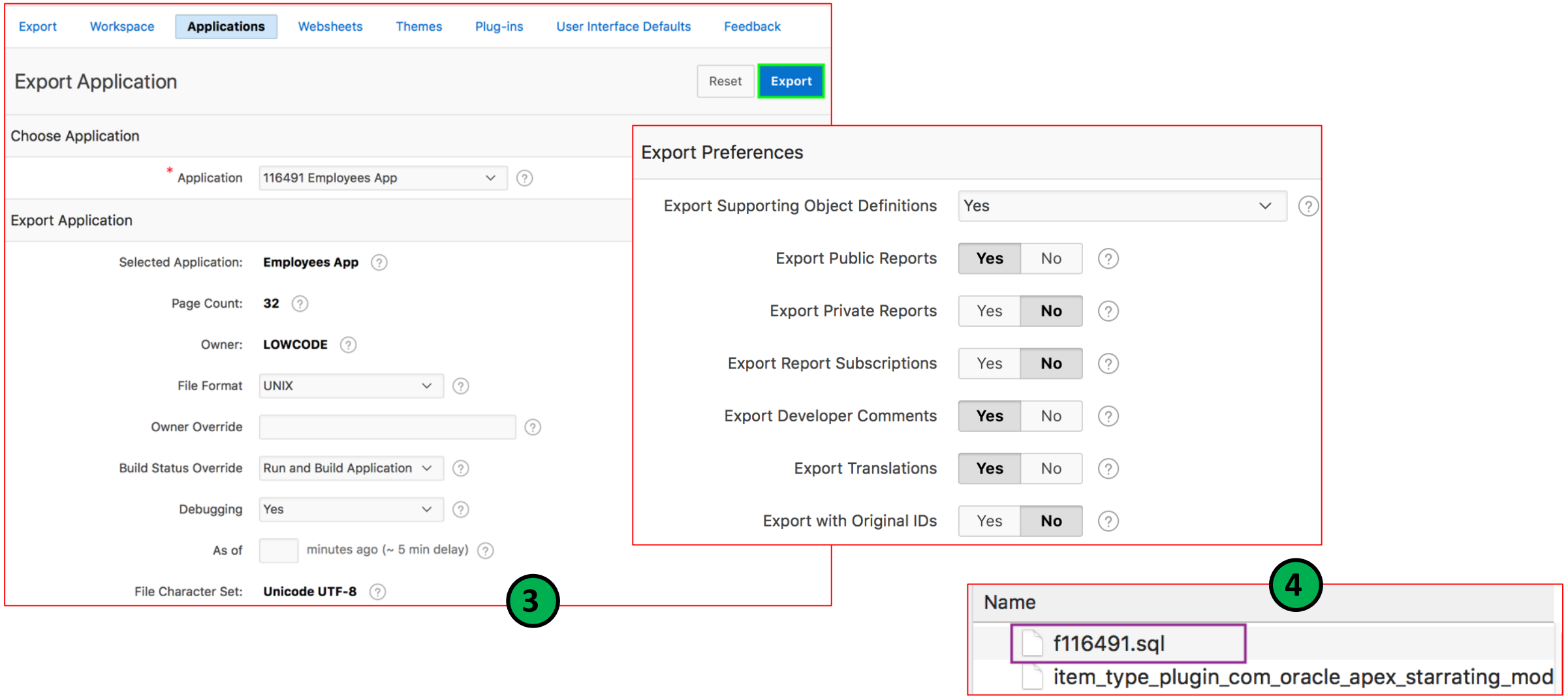
1. Exporting the application from the current environment, and importing it into the new environment
2. Recreating all of the database objects, and then populating the tables with the data from the current environment
3. Testing the application in the new environment

Exporting the Application

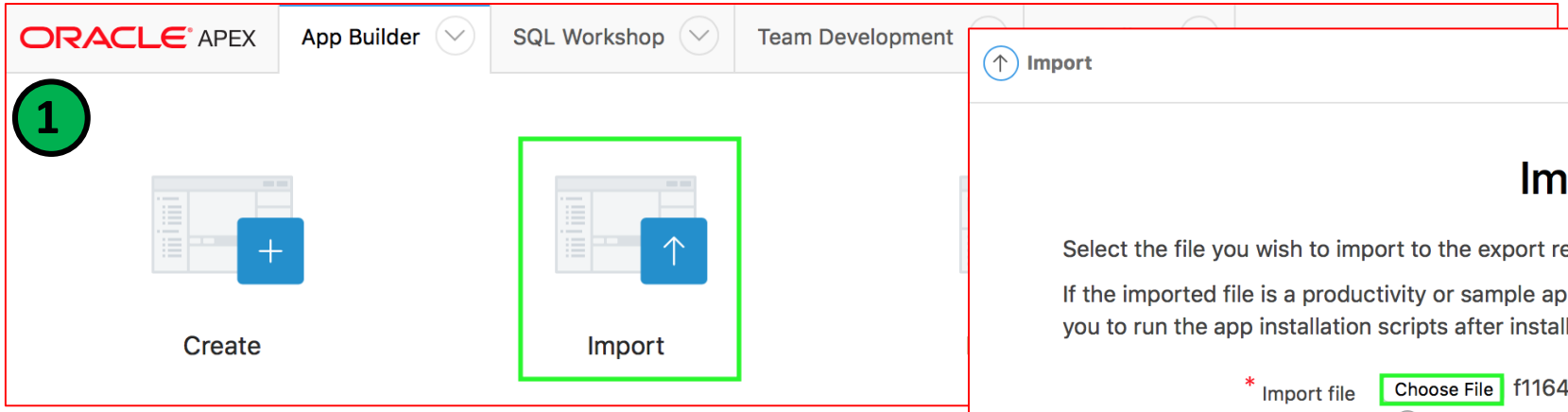


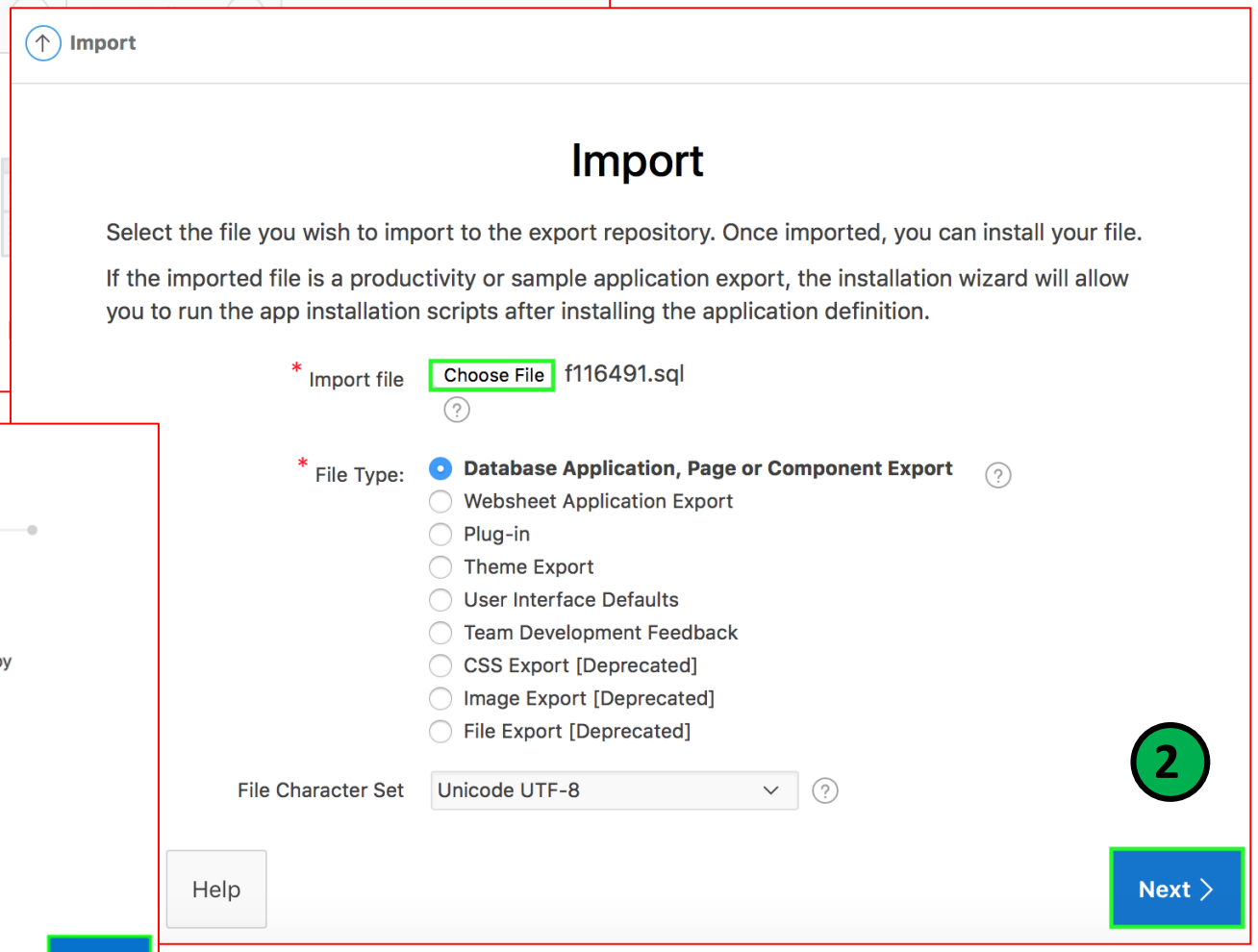


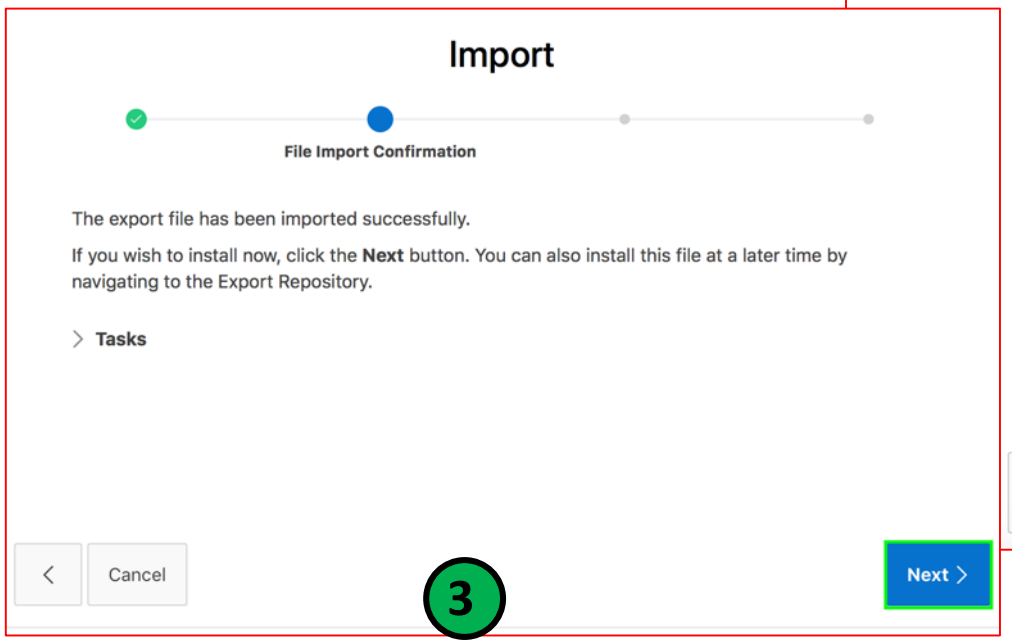
Exporting the Application



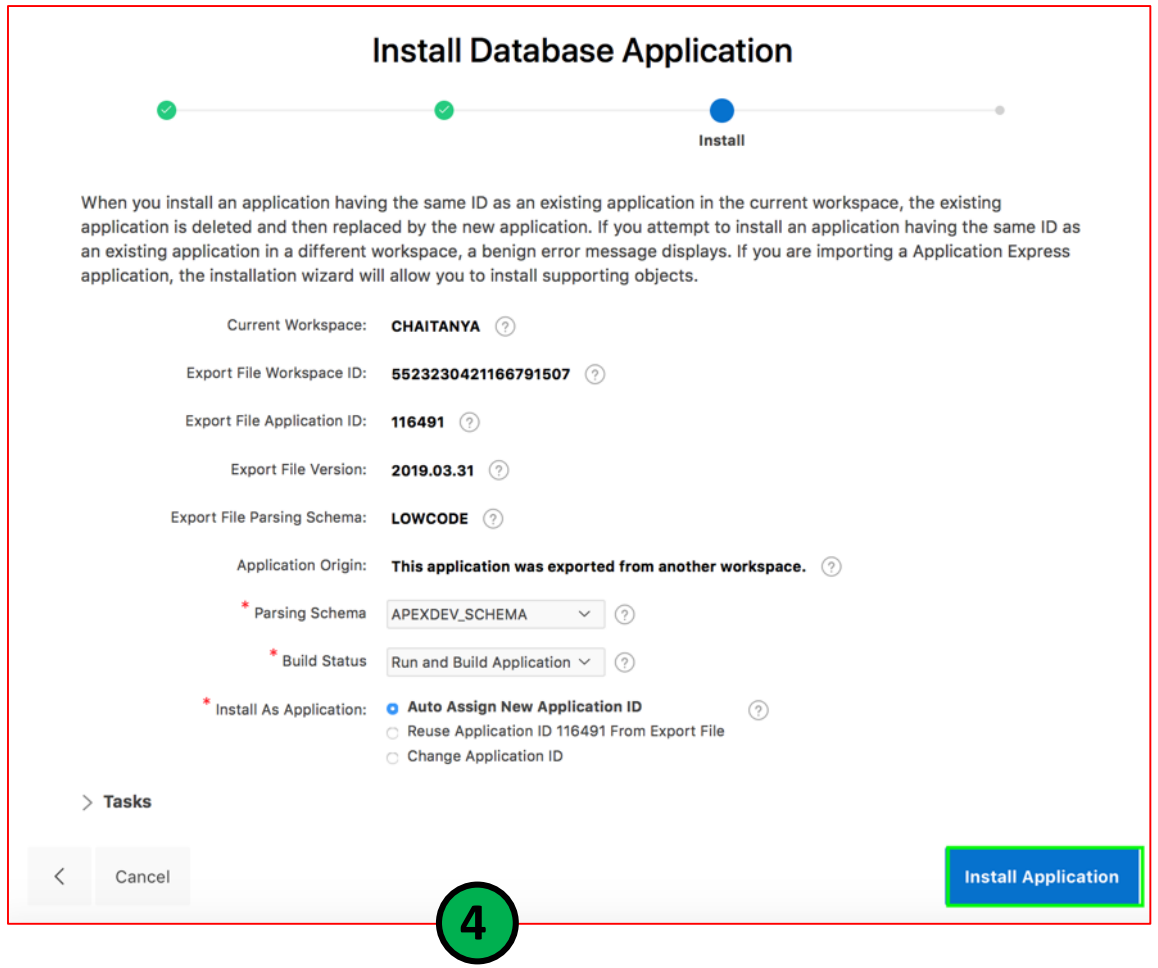
Importing the Application into the Target Environment

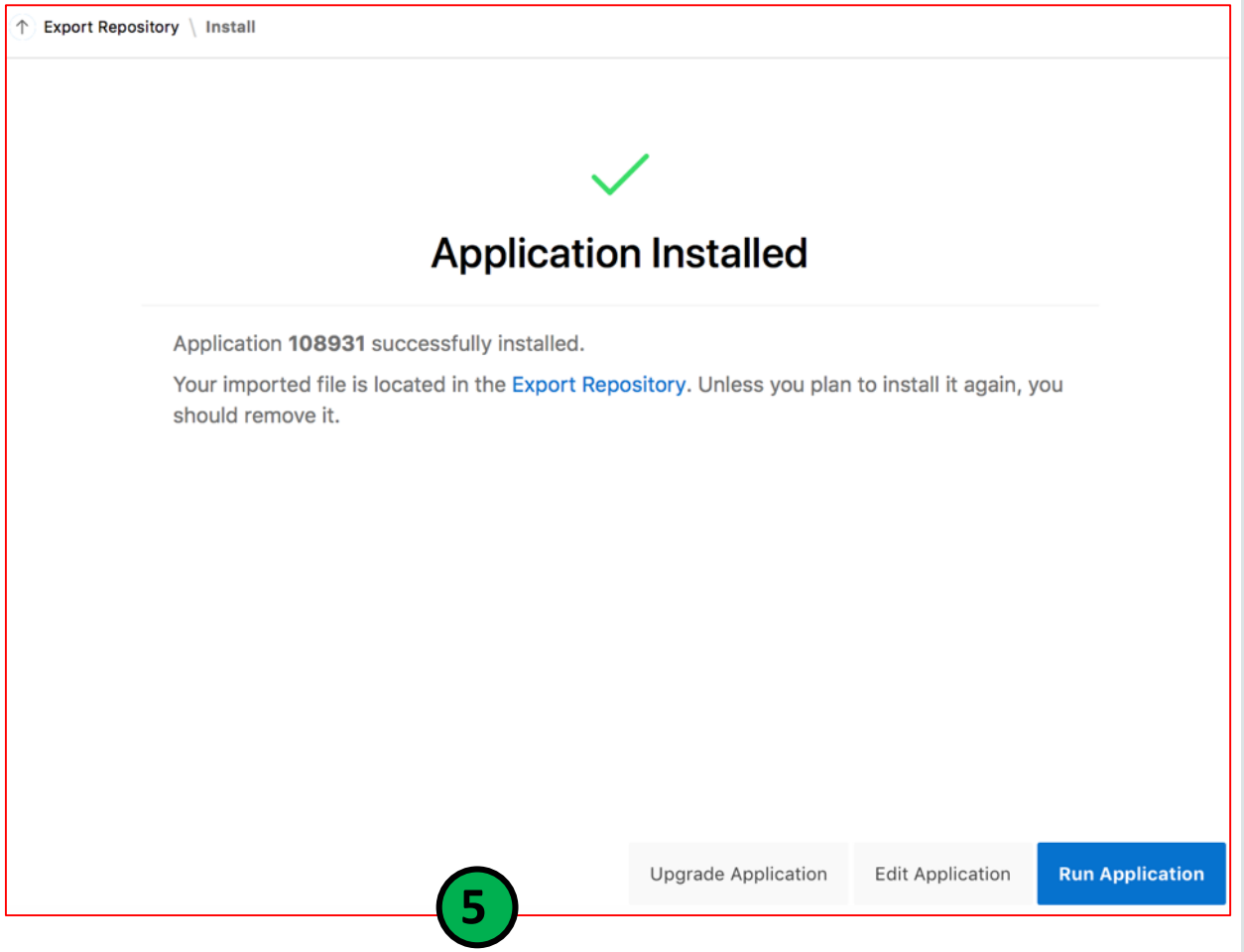




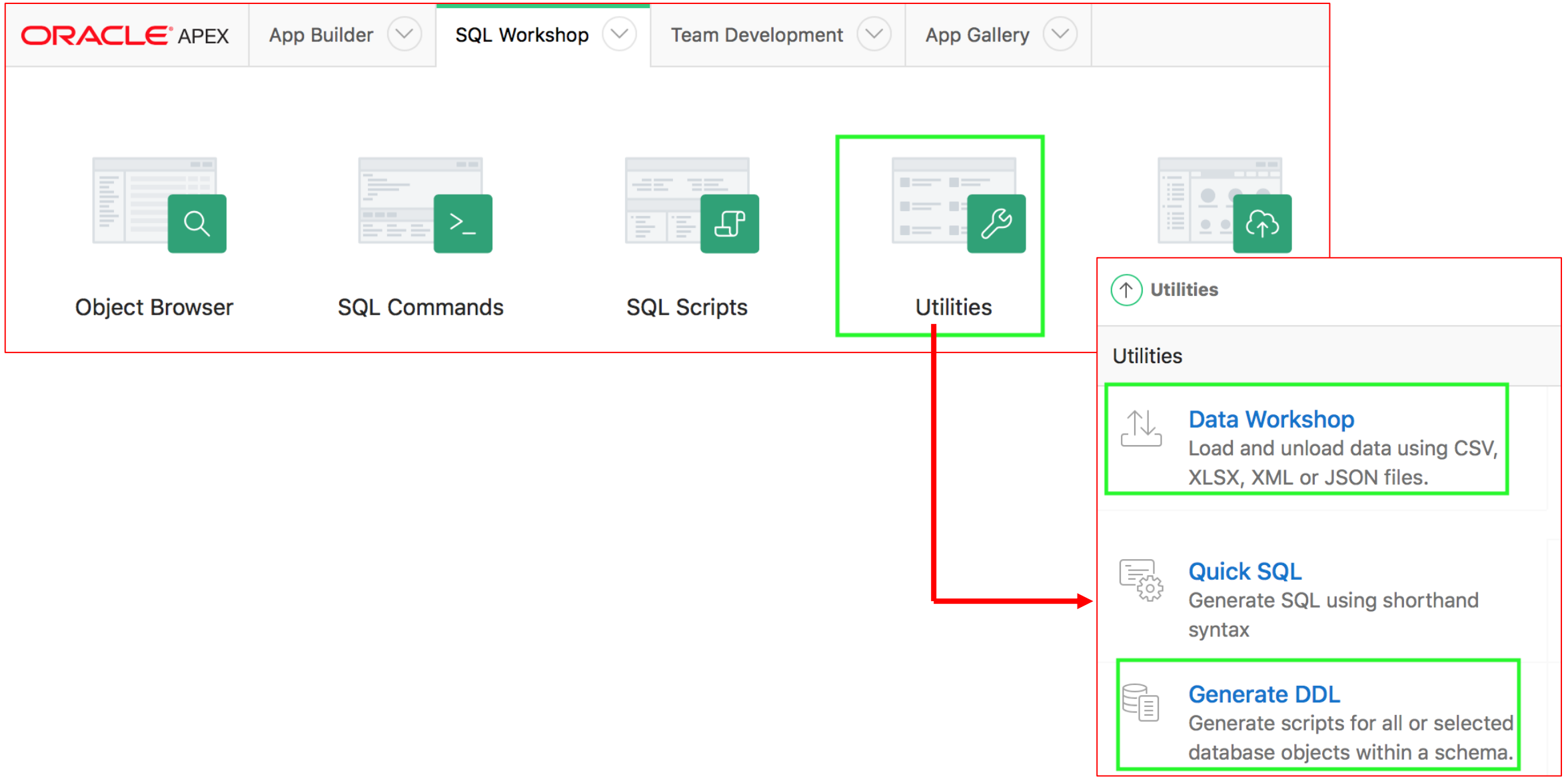


Importing the Application into the Target Environment

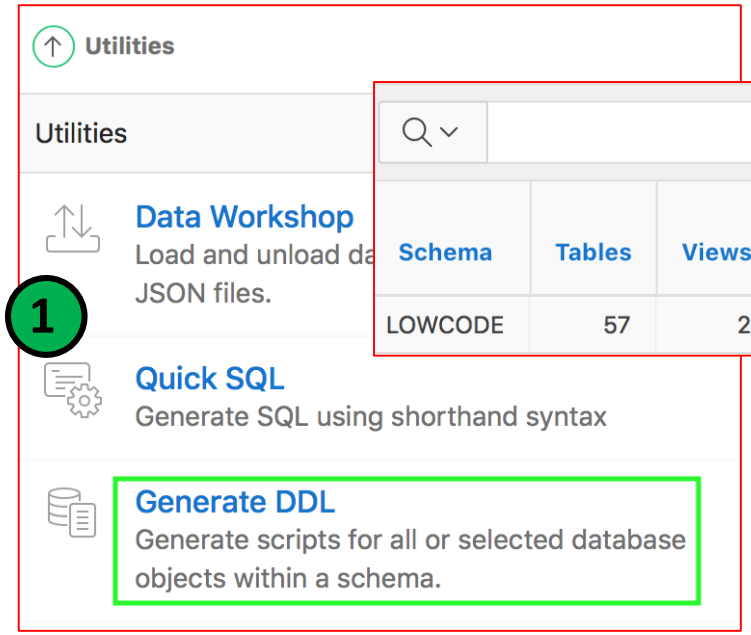


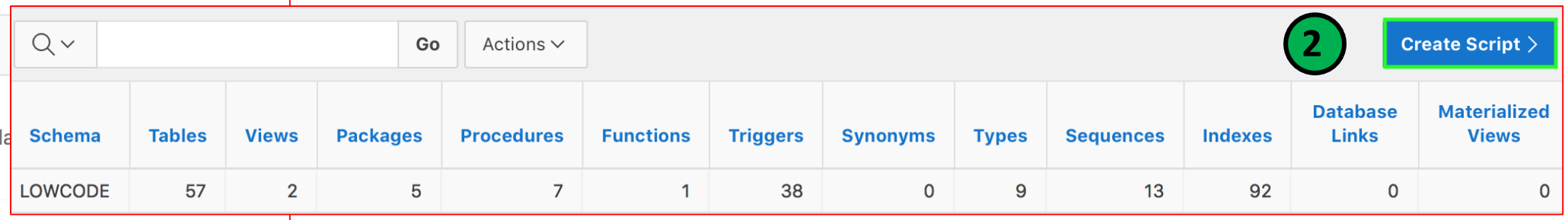


Migrating the Database Objects and Data

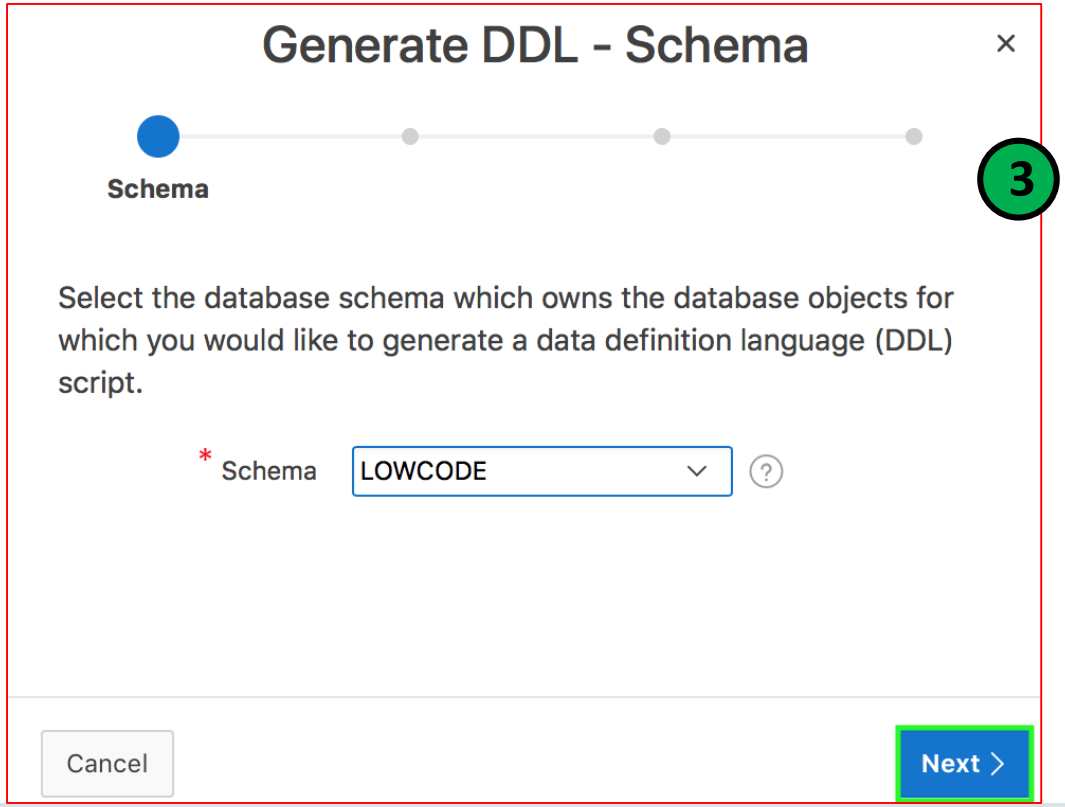


Migrating the Database Objects: Tables

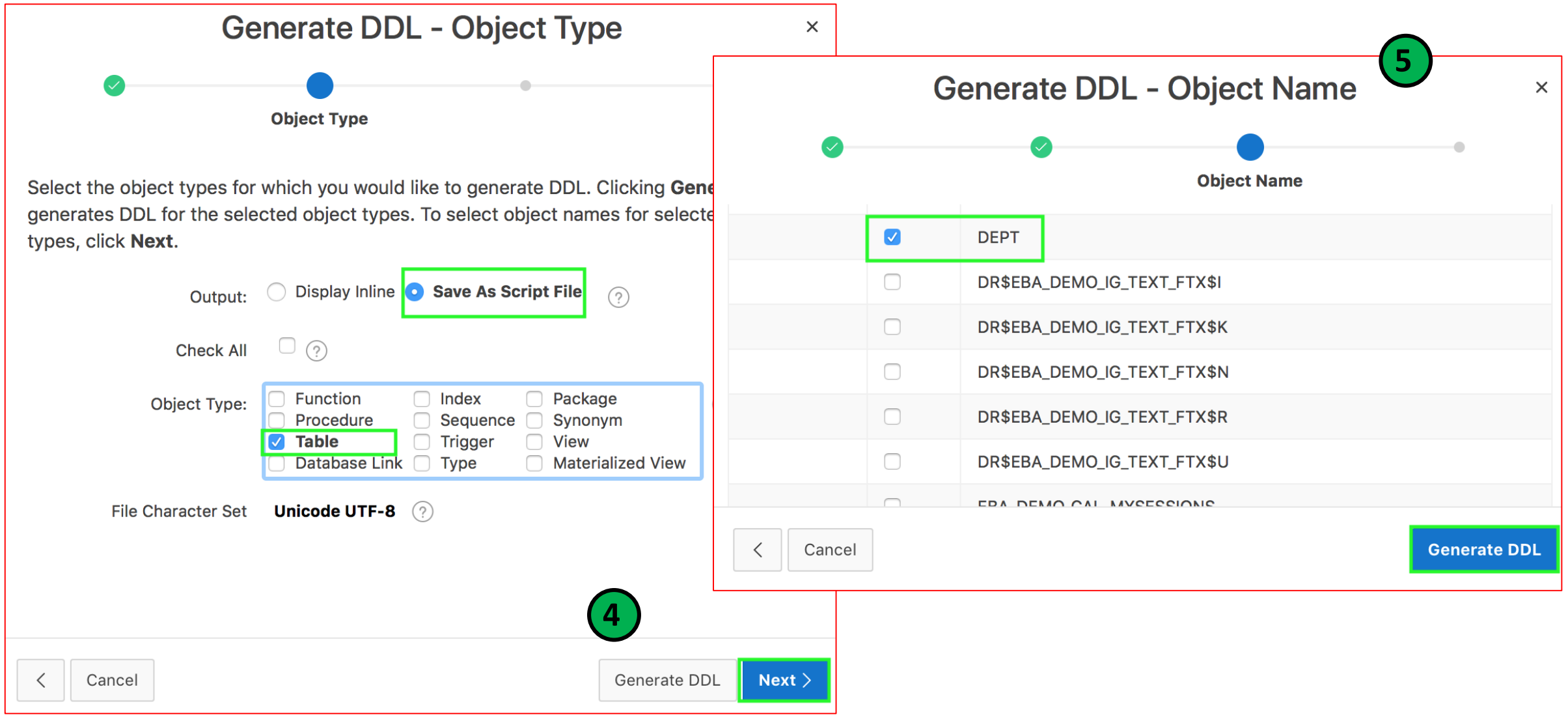




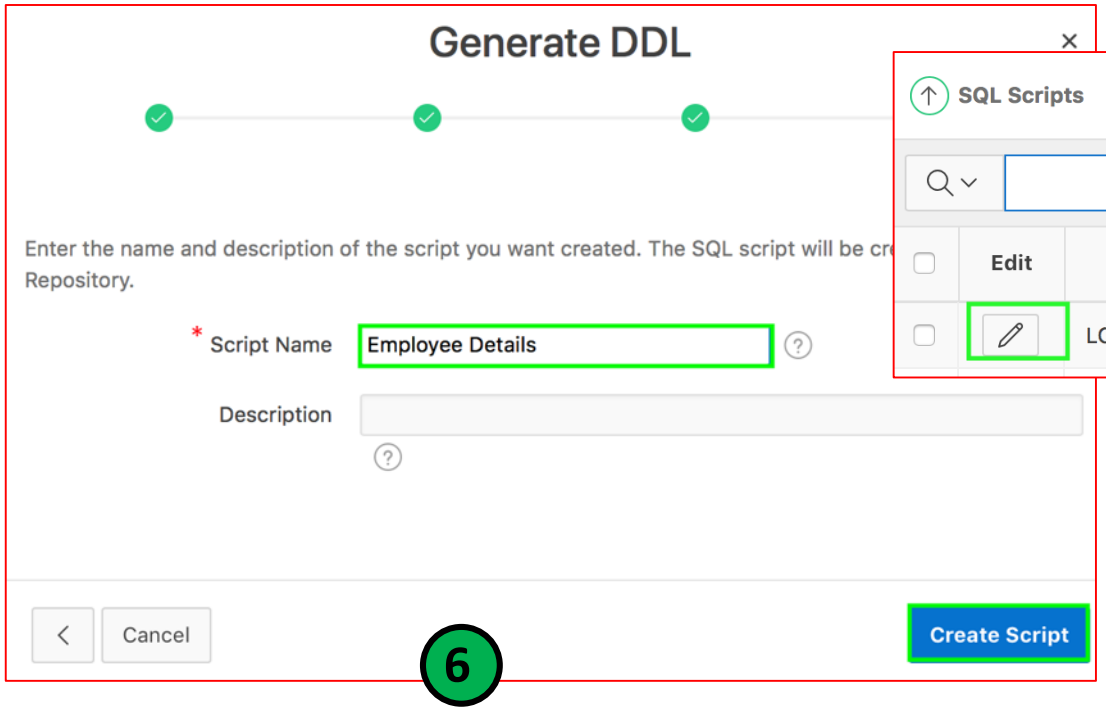
Schema	Tables	Views	Packages	Procedures	Functions	Triggers	Synonyms	Types	Sequences	Indexes	Database Links	Materialized Views
LOWCODE	57	2	5	7	1	38	0	9	13	92	0	0

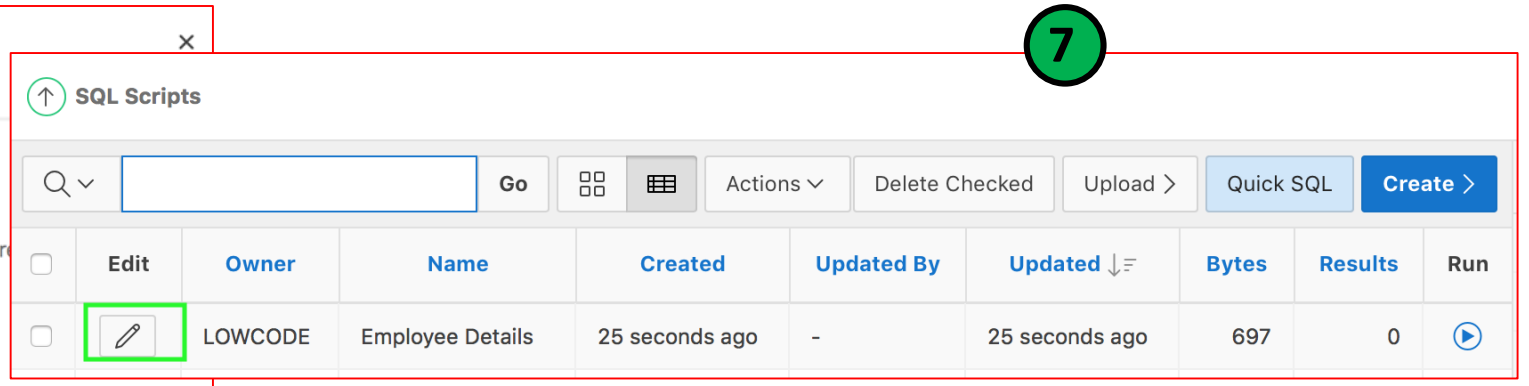


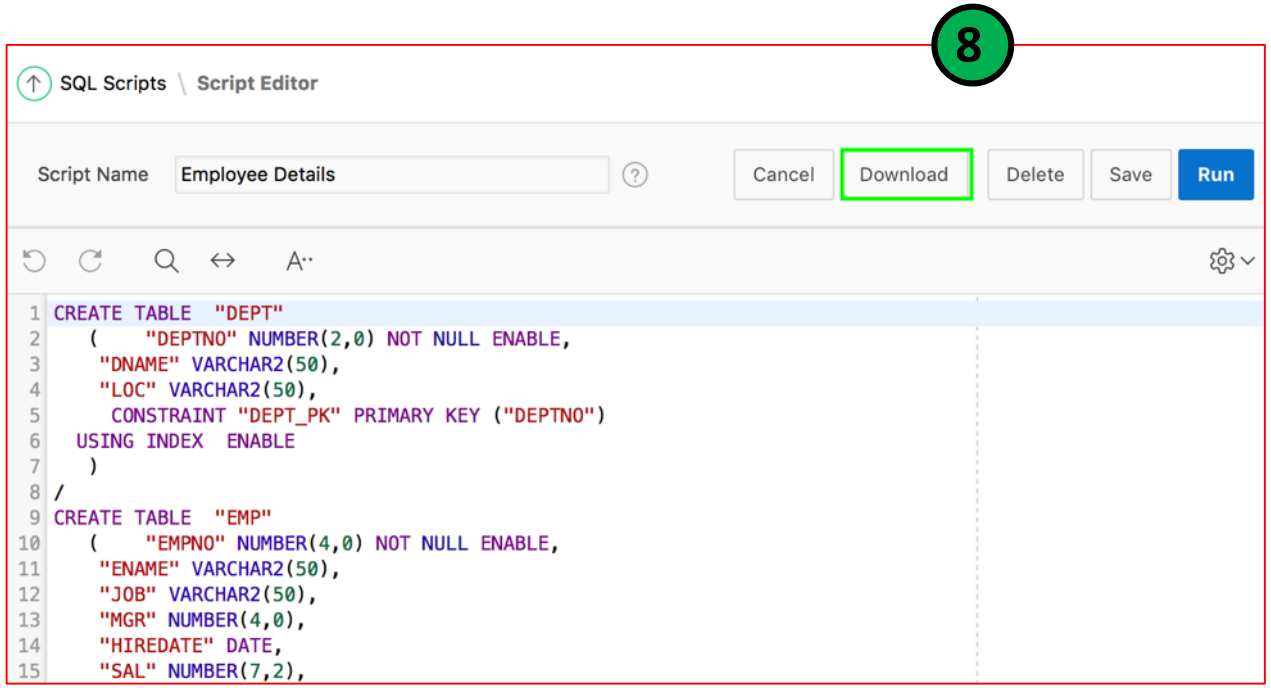
Migrating the Database Objects: Tables



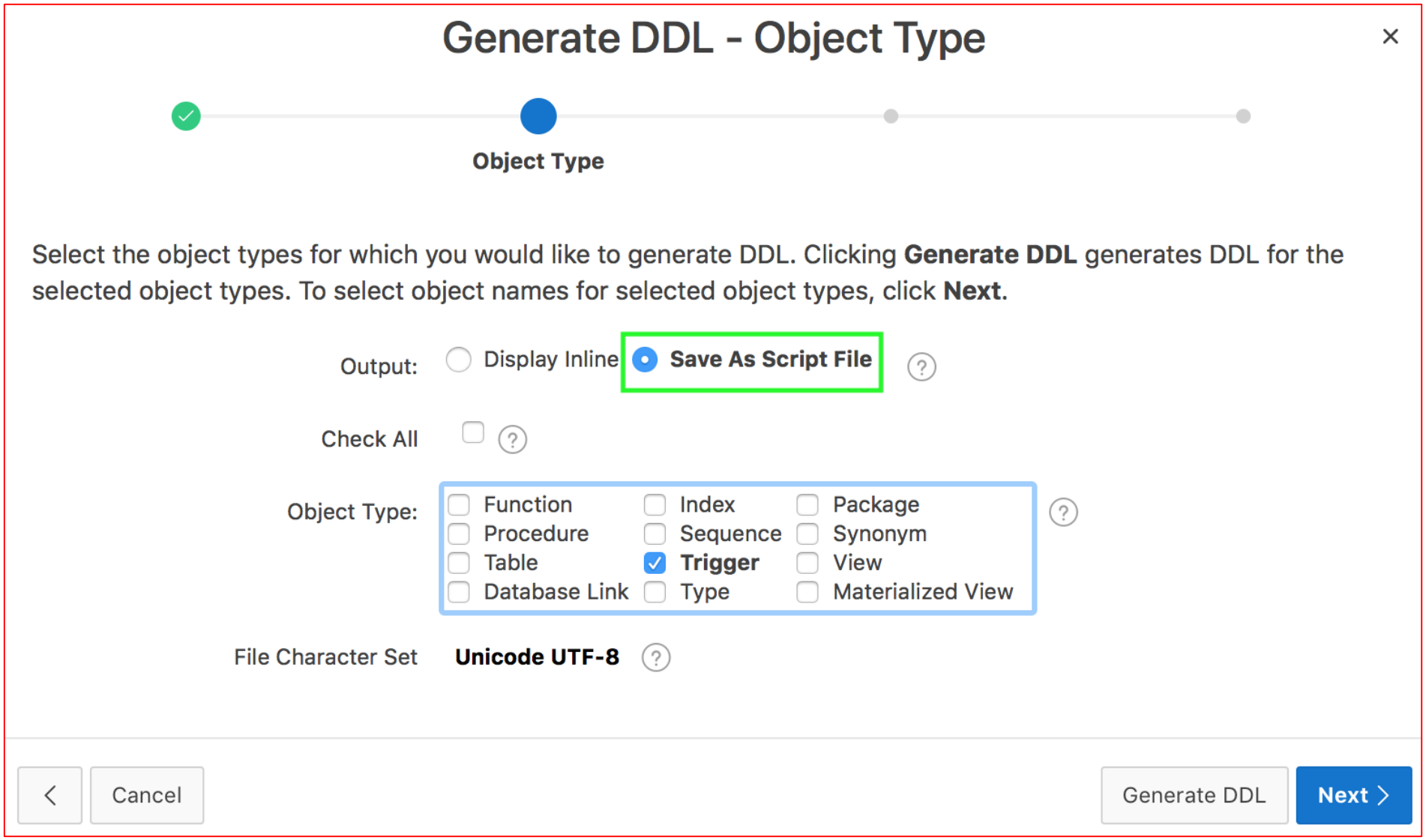
Migrating the Database Objects: Tables



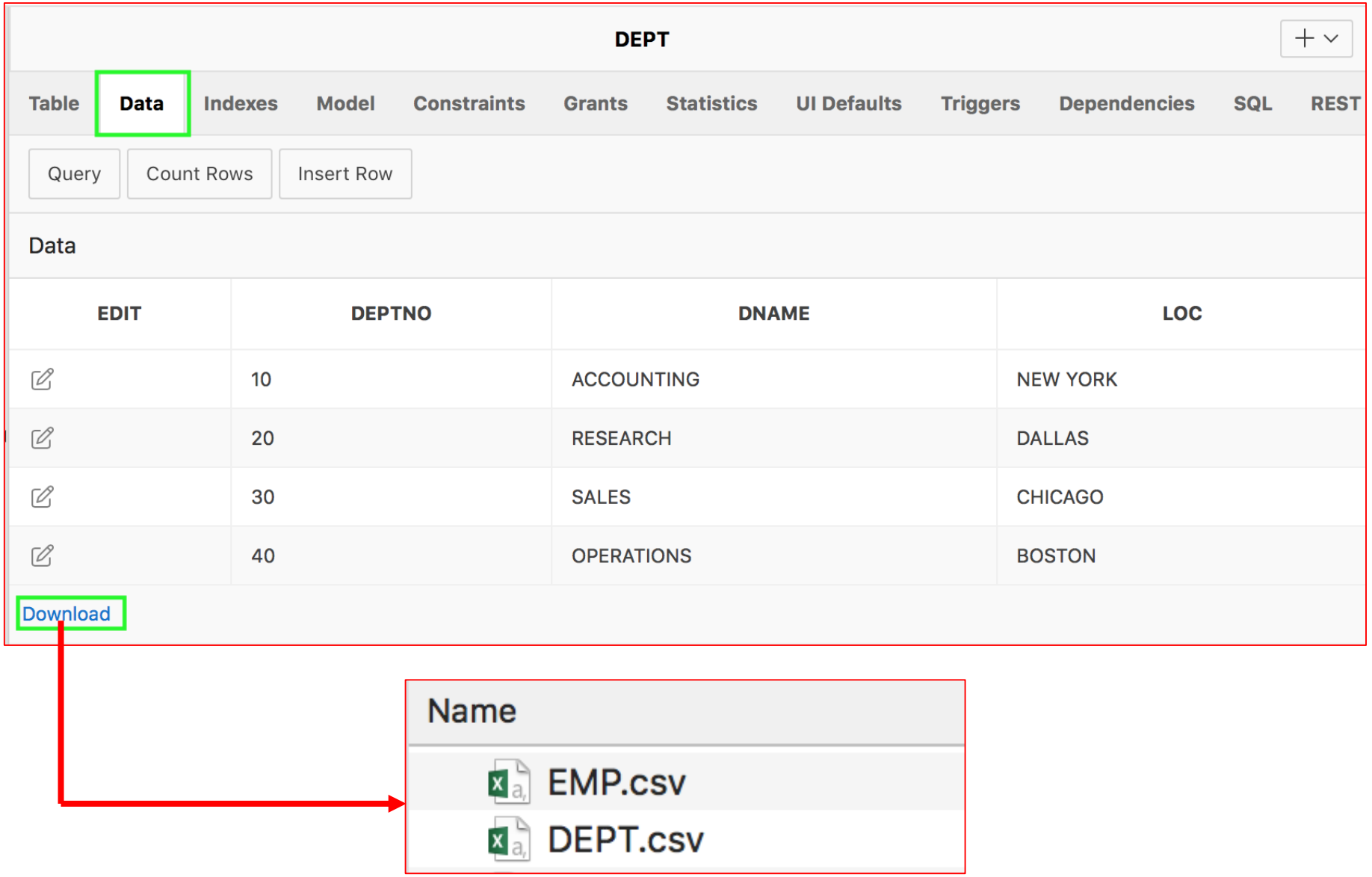




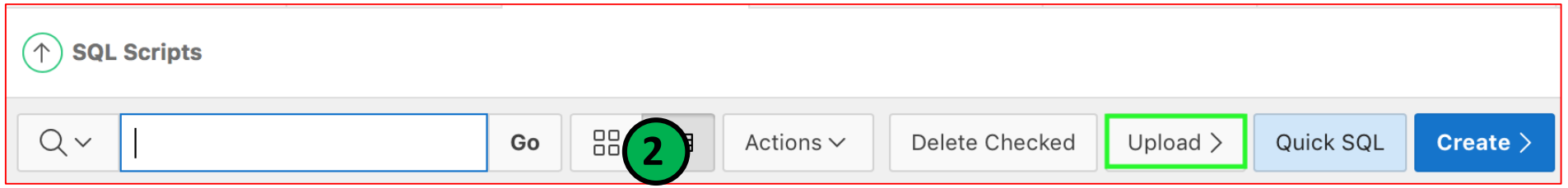
Migrating the Database Objects: Triggers

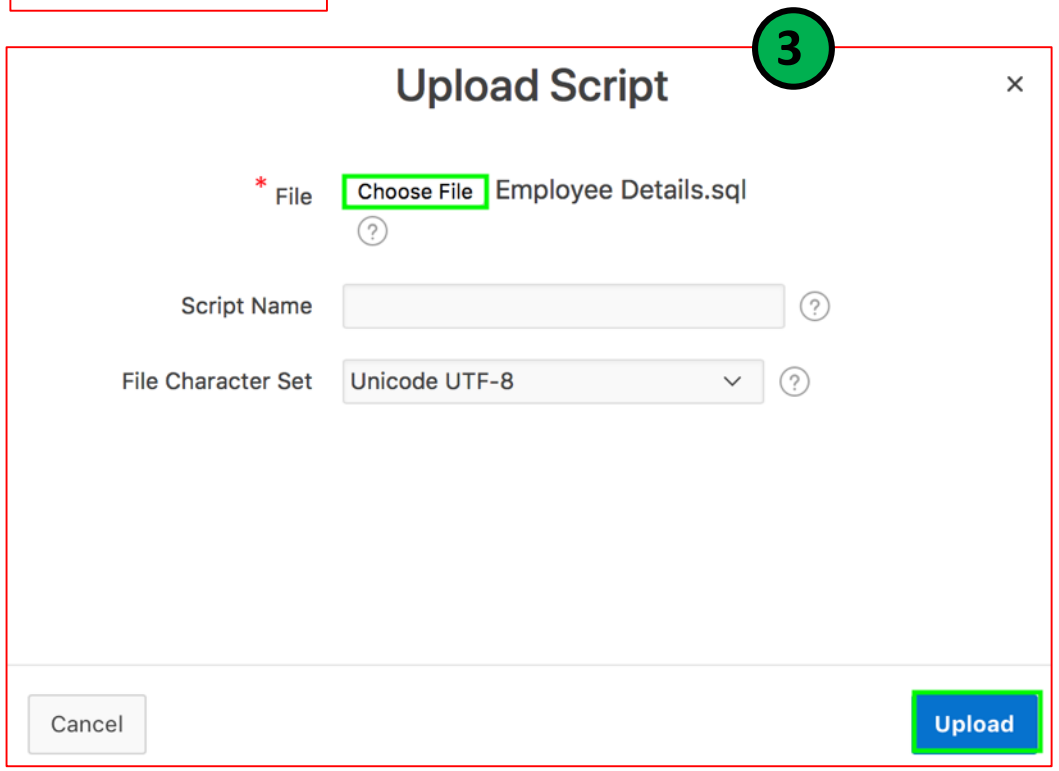


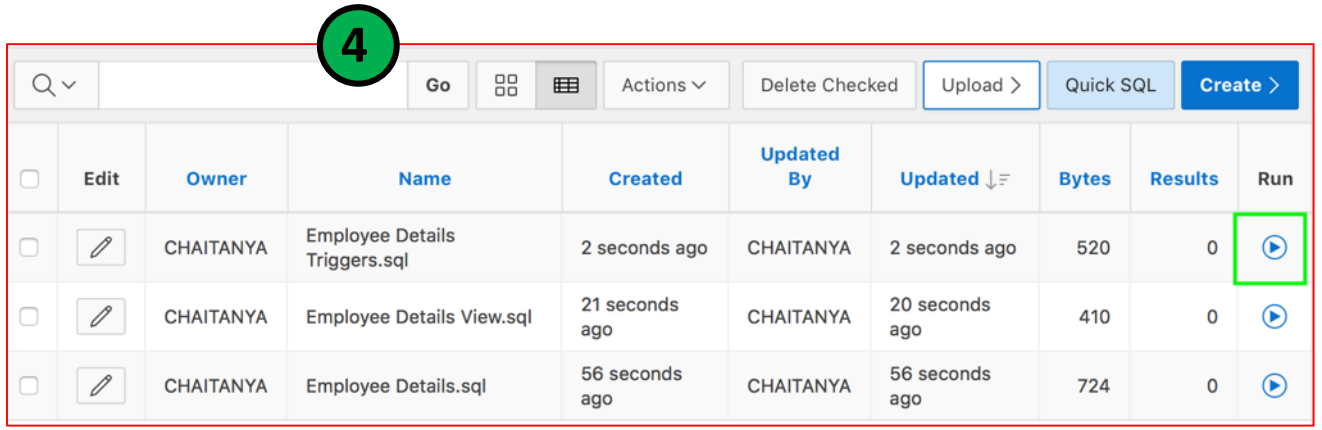
Migrating the Data



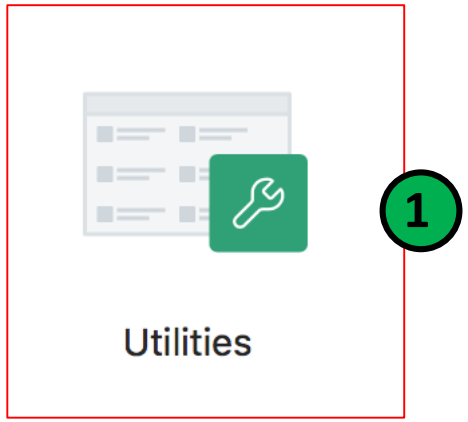
Recreating the Database Objects in the Target Environment

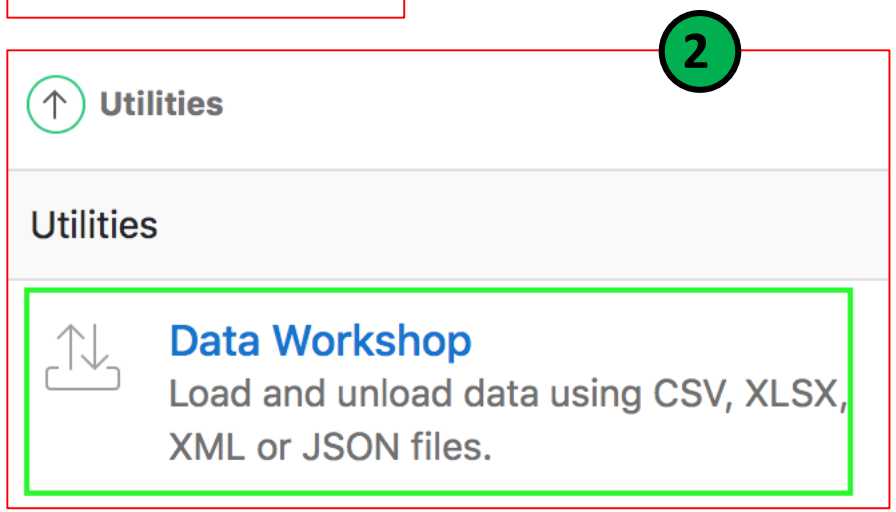


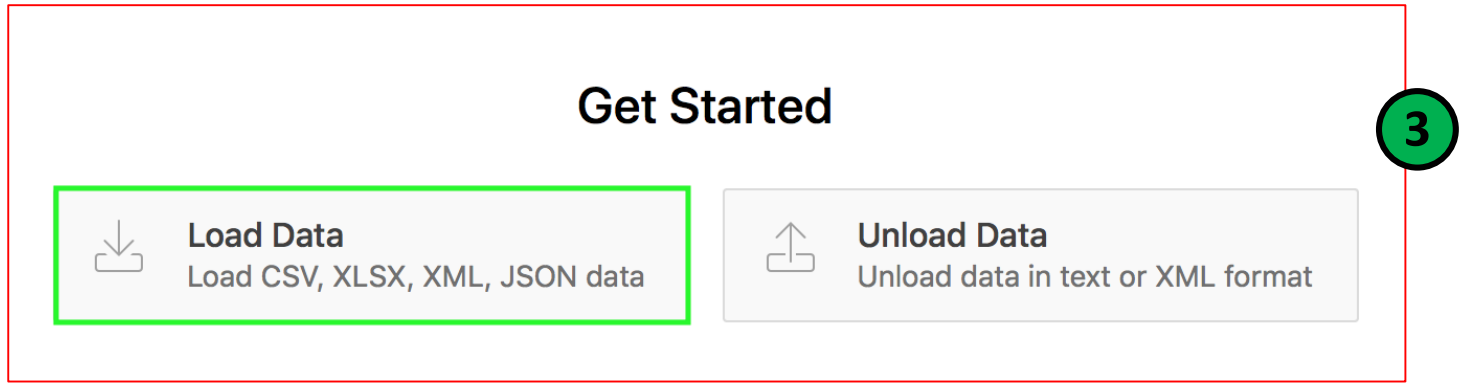


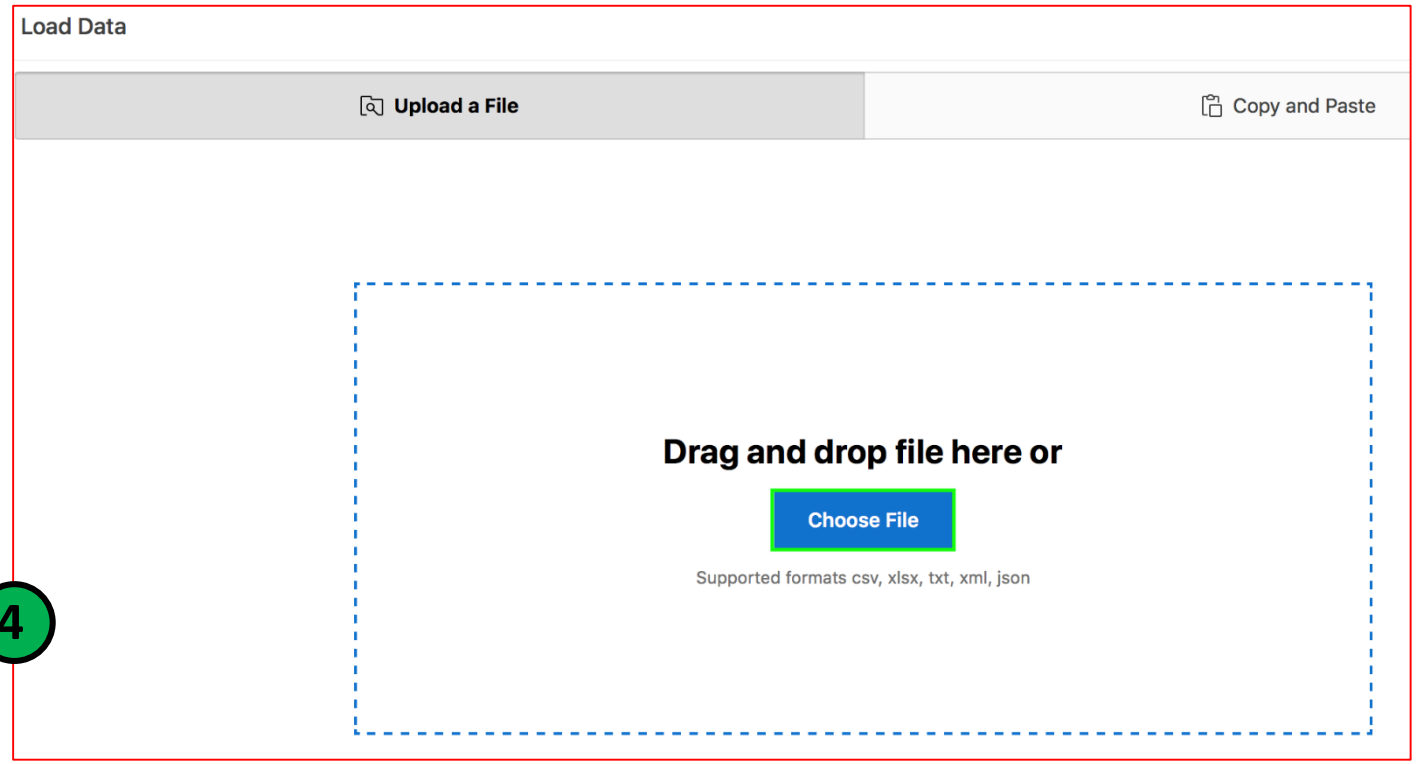


Migrating Data to the Target Environment1

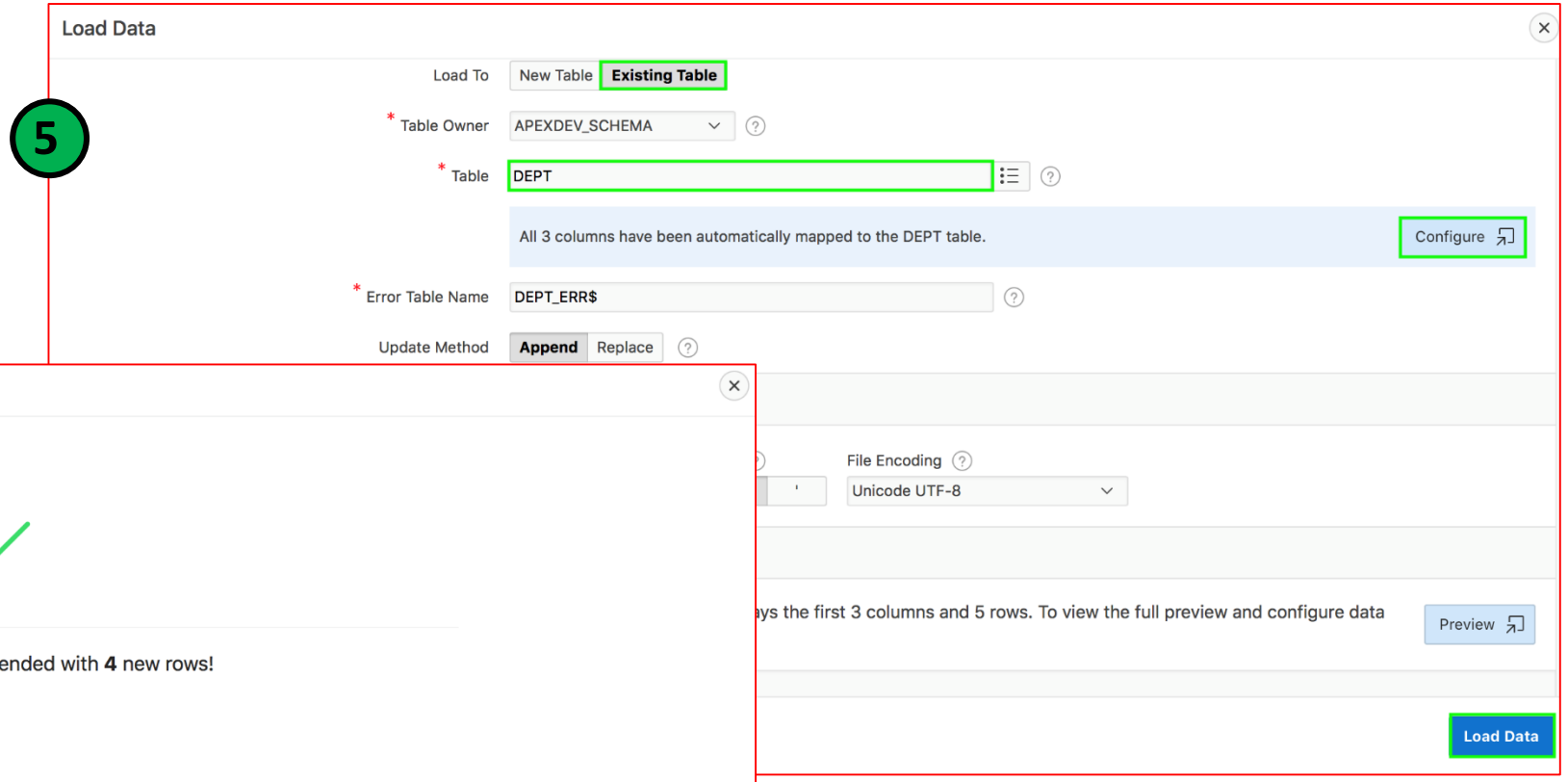




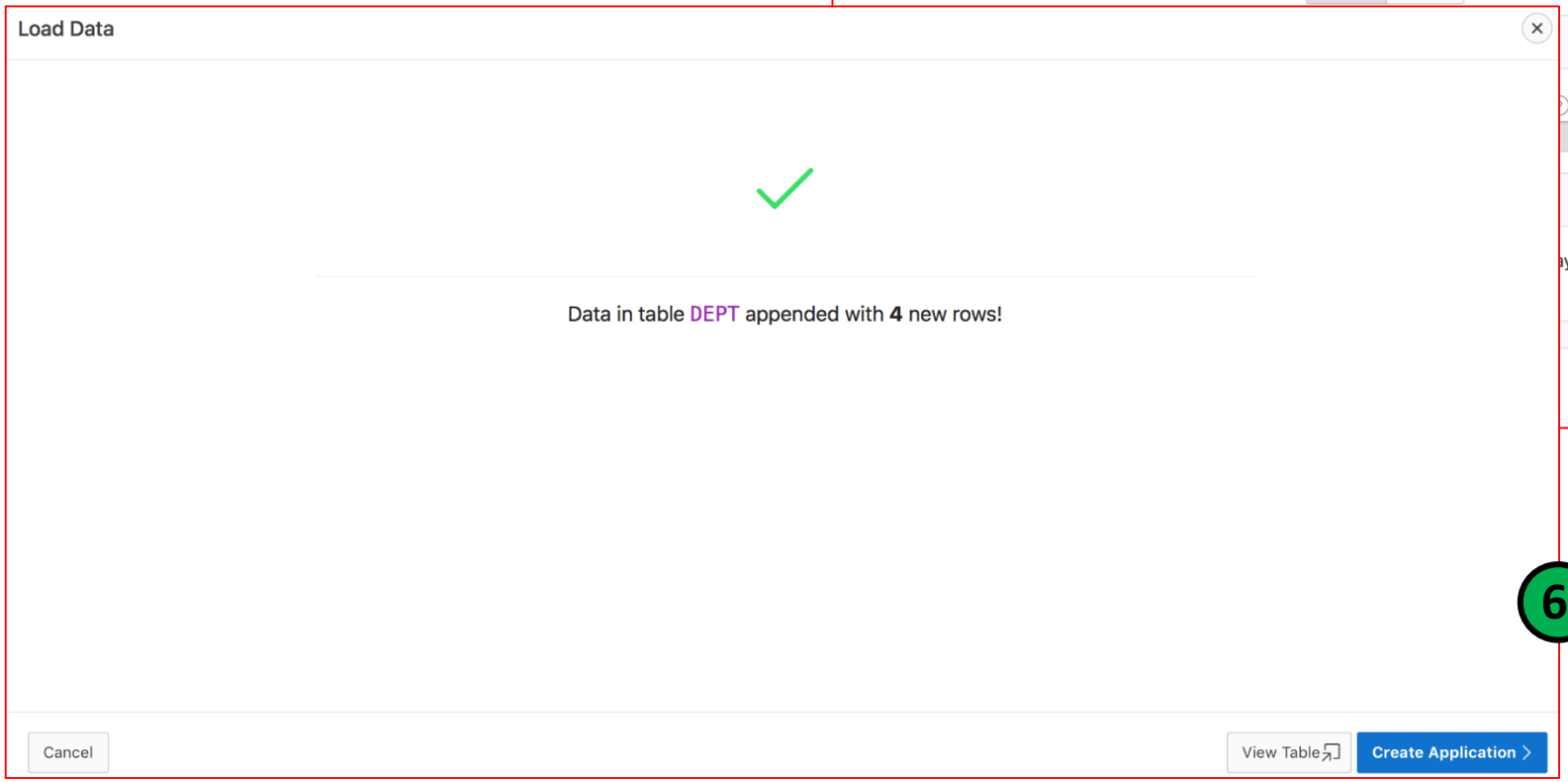




Migrating Data to the Target Environment

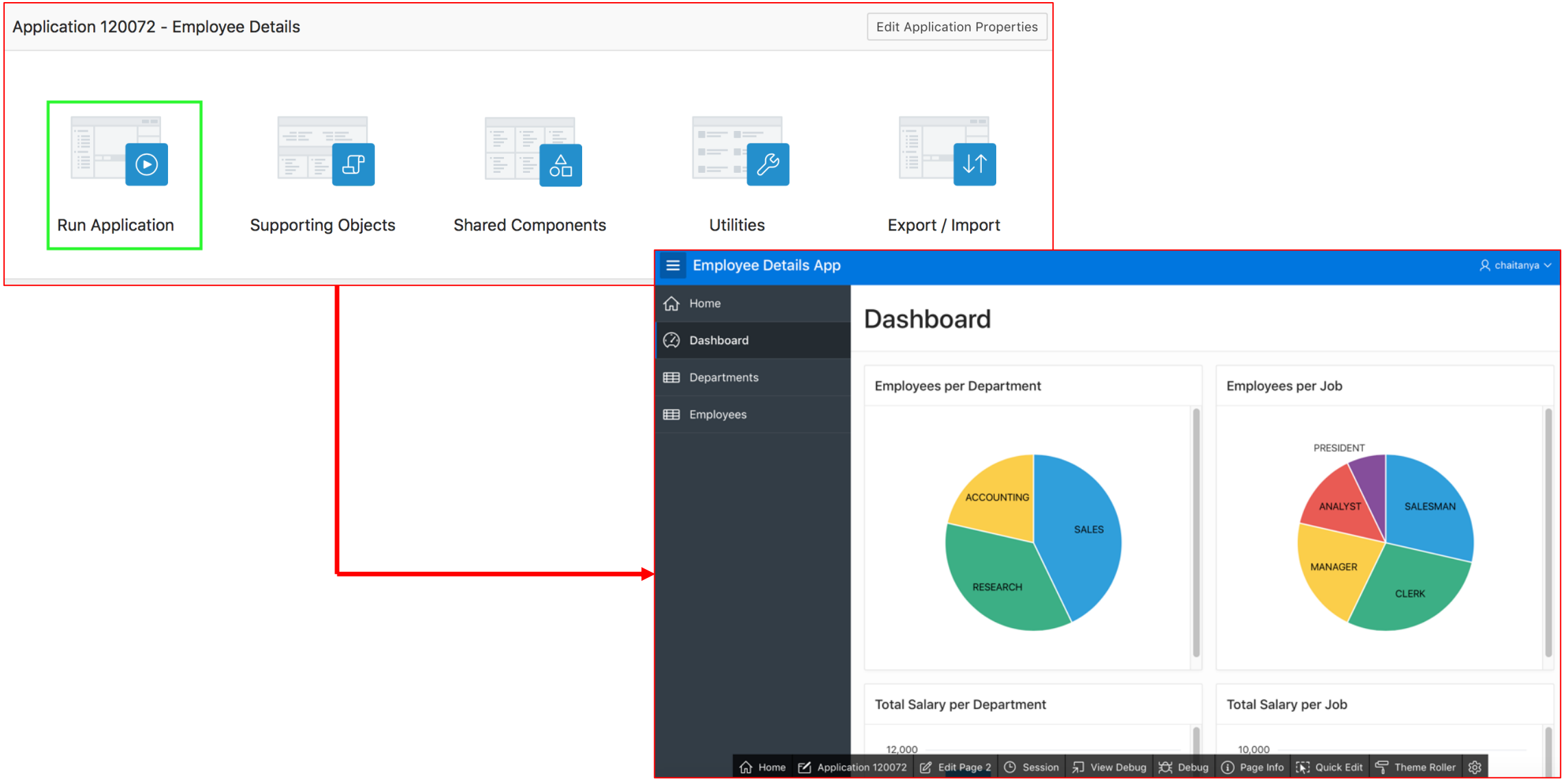


5



6

Testing the Migrated Application



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