Oracle® Application Express

Tutorial: Building an Application

Release 19.1
F13756-02
July 2019
Contents

Preface

Audience v
Documentation Accessibility v
Related Documents v
Conventions vi

Changes in This Release

Changes in Oracle Application Express Release 19.1 vii

1 About Oracle Application Express

2 Before You Begin

3 Populating Your Workspace

3.1 Installing the Sample Datasets 3-1
3.2 Reviewing the Created Tables with Object Browser 3-4

4 Creating the Initial Application

4.1 Running the Create Application Wizard 4-1
4.2 Adding a Cards Page 4-2
4.3 Adding a Dashboard Page 4-3
4.4 Moving Pages in the Create Application Wizard 4-5
4.5 Configuring Features and Advanced Settings 4-6
4.6 Completing the Create Application Wizard 4-7
4.7 Running the Application for the First Time 4-9
Developing the Projects Page

5.1 Navigating to Page Designer from the Application Runtime 5-1
5.2 Creating a Form Page Using the Create Page Wizard 5-2
5.3 Linking Project Cards to Maintain Project Form 5-3
5.4 Updating the Maintain Project Form in Page Designer 5-6
5.5 Creating a Dynamic Action 5-14
5.6 Adding Validations to Completed Date 5-16

Developing the Milestones Page

6.1 Creating an Interactive Grid Page 6-1
6.2 Updating the Navigation Menu for Milestones 6-2
6.3 Hiding Columns in the Application Runtime 6-4
6.4 Adding Project Names to the Milestones Interactive Grid 6-6
6.5 Changing the Appearance of an Interactive Grid in Page Designer 6-8
6.6 Finalizing the Milestones Interactive Grid in the Runtime 6-9

Developing the Tasks Page

7.1 Creating an Interactive Report Page 7-1
7.2 Updating the Tasks Page in the Runtime 7-2
7.3 Applying Lookups to the Tasks Interactive Report 7-4
7.4 Updating the Remaining Columns in the Tasks Report 7-5
7.5 Updating the Project and Milestone Fields in the Update Task Form 7-7
7.6 Updating the Remaining Fields in the Update Task Form 7-10

Developing the Dashboard Page

8.1 Deleting Regions in Page Designer 8-1
8.2 Updating the Series in the Project Costs Chart 8-3
8.3 Updating the Appearance of the Dashboard Chart 8-5
8.4 Updating the Live Template Options for the Dashboard Chart 8-6
8.5 Adding a Classic Report Region from Gallery 8-7

Importing an Application into a Workspace

Index
Preface

Oracle Application Express Tutorial: Building an Application demonstrates how to build a working application for managing projects and tasks using some of the latest features of Oracle Application Express.

Audience

Oracle Application Express Tutorial: Building an Application is intended for workspace administrators who want to set up an Oracle Application Express development environment and application developers who want to learn how to build database-centric web applications using Oracle Application Express.

To use this guide, you must have a general understanding of relational database concepts and the operating system environment under which Oracle Application Express is running.

Documentation Accessibility

For information about Oracle’s commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

For more information, see these Oracle resources:

• Oracle Application Express Release Notes
• Oracle Application Express Installation Guide
• Oracle Application Express End User’s Guide
• Oracle Application Express App Builder User’s Guide
• Oracle Application Express Administration Guide
• Oracle Application Express SQL Workshop Guide
• Oracle Application Express API Reference
• Oracle Application Express Application Migration Guide
Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
Changes in This Release

Changes in Oracle Application Express Release 19.1

The following are changes in Oracle Application Express Tutorial: Building an Application for Oracle Application Express release 19.1.

New Features

Oracle Application Express Tutorial: Building an Application contains expanded content for Oracle Application Express release 19.1:

• Creating a new Interactive Grid page
• Creating a new Interactive Report with Form page(s)
• Configuring charts and creating a new report region on the Dashboard page
• Appendix for importing application export scripts enabling the user to skip around the tutorial

Other Changes

The following are additional changes in the release:

• All content has been updated to reflect new functionality.
• Screen captures and graphics have been added and updated to reflect the current Oracle Application Express user interface.
About Oracle Application Express

Oracle Application Express is an environment designed for rapid application development and deployment.

You can develop professional applications that are both fast and secure using only a web browser and limited programming experience. Oracle Application Express accelerates the application development process with features such as user interface themes, navigational controls, form handlers, and flexible reports. The deployed applications require only a browser and access to an Oracle database running Application Express.

This tutorial is intended to guide you through almost every step of creating an app with a focused amount of context. For more in-depth information about Oracle Application Express, a good place to start is the "Quick Start" section of the Oracle Application Express App Builder User's Guide.

About This Tutorial

In Oracle Application Express Tutorial: Building an Application, you create a simple project management application for the fictitious AnyCo Corp. The application manages IT project information stored in database tables. This tutorial shows you how to build, modify, and run the application. Each chapter is a further lesson that advances the work of previous one so that the concepts build on one another.

Oracle Application Express Tutorial: Building an Application is intended to be followed in the order it is written. To skip to the beginning of any lesson, you can import one of the exported application files stored in the .zip file accompanying this tutorial. This feature is useful for skipping ahead in the tutorial to your skill level or checking your work.

See the appendix "Importing an Application into a Workspace."
Before You Begin

For this tutorial, you must have access to an Oracle Application Express environment and an account with development privileges.

Accessing an Oracle Application Express Environment

You must have access to an Oracle Application Express release 19.1 or later. You can access Oracle Application Express in one of the following ways:

- Request an account on apex.oracle.com.
- Subscribe to an account on the Oracle Database Cloud Service which includes Oracle Application Express.
- Install a local installation of Oracle Database which includes Oracle Application Express release 19.1 or later.

Oracle Application Express Development User Account

If you are accessing Oracle Application Express in a hosted environment, requesting an account creates a workspace and an account. If you are managing Oracle Application Express locally, you will create a workspace and account manually. In either scenario, your Oracle Application Express account must have developer privileges.
Populating Your Workspace

To build an Oracle Application Express app, you need a complete set of tables and data.

About This Lesson

In this lesson, you install **Sample Datasets** provided by Oracle.

Once installed, you also review the output of the scripts in Object Browser to confirm the presence of the tables and data.

The sample tables and data that you load in these steps resemble the kind of information that real Application Express project management apps use every day. They include information typical of data sets for employees, projects, and calendars.

3.1 Installing the Sample Datasets

The Application Express SQL Workshop comes with several sample datasets intended to enhance app experimentation.

To install the Project Data sample dataset in SQL Workshop:

1. Sign in to your Oracle Application Express workspace.
2. Click **SQL Workshop**, click **Utilities**, and click **Sample Datasets**.
Tip:
You can also click the down arrow icons to open the drop-down menus on the tabs at the top of the screen to shortcut your way around your workspace.

3. Find Project Data in the list and click the adjacent Install button.

The Manage Sample Dataset dialog displays.

4. Click Next.

5. Click Install Dataset.
6. In Results dialog, review the new objects.

The script created the following objects:

- SAMPLE$PROJECT_STATUS (Table)
- SAMPLE$PROJECTS (Table)
- SAMPLE$PROJECT_MILESTONES (Table)
- SAMPLE$PROJECT_TASKS (Table)
- SAMPLE$PROJECT_TASK_TODOS (Table)
- SAMPLE$PROJECT_TASK_LINKS (Table)
- SAMPLE$PROJECT_COMMENTS (Table)
- SAMPLE$PROJECTS_V (View)
- SAMPLE$PROJECTS_COMPLETED_V (View)

7. Click Exit.

Note:
The Create Application button is a shortcut to the Create Application Wizard loaded with the sample data.

You have populated the sample tables and data in your workspace.
3.2 Reviewing the Created Tables with Object Browser

Object Browser displays all the raw data present in the Oracle database schema associated with your workspace.

To navigate to Object Browser:

1. Click SQL Workshop to return to the SQL Workshop home page.
2. Click Object Browser.

**Tip:**
For quicker navigation, click the arrow next to SQL Workshop to display a drop-down menu.

3. In the list of tables, select SAMPLE$PROJECTS.

4. Click the Data tab. The Data tab displays the raw information of the SAMPLE$PROJECTS table.
5. When finished, click the **Oracle** logo in the top-left of the page to return to the Workspace home page.

You have fully populated your workspace and are ready to create a new application. Proceed to the next lesson.

---

**Note:**

You can edit the data directly by clicking the Edit icon on the Data report. However, leave the data unchanged for now.
Creating the Initial Application

About This Lesson

In this lesson, you use the Create Application Wizard to build the foundational pages of your application based on the sample dataset.

After you build the foundation of your app, use Page Designer's Create Page Wizard to add a new page to the existing app and run the app for the first time.

⚠️ WARNING:

It is important to complete this chapter in one sitting. You cannot save your work until the end of the chapter. If you stop before completion and your session times out, all your progress will be lost, and you will have to start the chapter again.

4.1 Running the Create Application Wizard

With the tables populated in your workspace, you can get started quickly with the Create Application Wizard.

The Create Application Wizard enables you to create many different types of pages, including reports, forms, and a master detail.

To create the initial application:

1. Click App Builder and select Create.
   
   The Create Application Wizard appears.

   ⚠️ Tip:
   
   You can also click the down arrow icon on the tabs at the top of the screen to view a list of shortcuts to the wizard and other pages.

2. Click New Application.

   To customize your app name and appearance:

3. In Name, enter Demo Projects

   ⚠️ Note:
   
   The Create Application Wizard creates a Home page automatically.

4. Click the Set Appearance icon.
5. Click **Vita - Red**.
6. For Application Icon, click **Choose New Icon**.
   The Choose Application Icon dialog displays.
7. Click the **red circle** icon.
8. Choose a new application icon, such as the trio of people.
9. Click **Set Application Icon** and click **Save Changes**.

### 4.2 Adding a Cards Page

Often at the beginning of development, you only know some of the pages that your application requires. You can create as few or as many pages as you want in the Create Application Wizard. Save some time by generating a small batch of pages now. You can always add more later with a different wizard (the Create Page Wizard).

Start with a cards page that enables employees to quickly view and maintain their contact information.

**Note:**

A cards page consists of individual boxes, which resemble index cards, laid out on a page. Each card holds up to three pieces of information.
To add a cards page based on the SAMPLESPROJECTS table:

1. In the Pages section, click Add Page.
2. Click Cards.
3. Enter the following:
   a. Page Name - enter Projects
   b. Table or View - click the LOV icon (bullet points and lines) and select SAMPLESPROJECTS
   c. Card Title - select NAME
   d. Description Column - select DESCRIPTION
   e. Additional Text Column - select BUDGET
   f. Click Add Page.

4.3 Adding a Dashboard Page

Next, add a dashboard page. A dashboard page consists of up to four charts. As a developer you can specify the specifics of each chart or leave the default sample charts, then update the page later.

To add a dashboard page:

1. Click Add Page.
2. Click Dashboard.

Dashboard pages come populated with useful metrics. Configure two of the charts for tracking project budget and project status, and leave the other two unconfigured. You can evaluate whether to keep them later.

To configure the charts:

3. Enter the following for Chart 1:
   a. Chart Type - click Bar
   b. Chart Name - enter Project Costs
   c. Table or View - select SAMPLESPROJECT_V
d. Label Column - select NAME

e. Aggregate Type - click Column Value

f. Value Column - select COST

4. Click Chart 2.

5. Enter the following:

   a. Chart Type - click Pie
   b. Chart Name - enter Project Status
   c. Table or View - select SAMPLESPROJECTS
   d. Label Column - select STATUS_ID
   e. Aggregate Type - click Count
6. Click Add Page.

4.4 Moving Pages in the Create Application Wizard

Once you create a page, you can click the Edit icon to edit the page name, change the icon that displays in the navigation menu, edit Advanced options, or delete the page. In addition to the Edit button, you can change the order of pages by dragging and dropping.

To move a page in the Create Application Wizard:

1. For the Dashboard page, adjacent to the Edit button, click and hold the hamburger icon (horizontal lines).

2. Drag the Dashboard page above the Projects page and release the mouse.

The Dashboard page snaps into place.
3. Verify that your list contains three pages in the following order:
   a. Home, Blank
   b. Dashboard, Dashboard
   c. Projects, Cards (sample$projects)

4.5 Configuring Features and Advanced Settings

Features provide application-level functionality and are typically added once per application. Features broaden the application’s capabilities, making it more suitable for wide distribution, rather than just being suitable for a proof-of-concept.

**Tip:**
While sometimes more complicated, you can also add Features to an existing app later with the Create Page Wizard.

To add Features to the application:
1. Next to the Features heading, click **Check All**.

Settings are application-wide and define attributes used to generate the application.
To edit advanced settings:
2. In the Settings section, in the Advanced Settings field, click the **Define Advanced Settings** icon (arrow in box).

3. Under the Description section, enter the following:
   a. **Short Description** - enter **Maintain project details**
   b. **Description** - copy and paste the following text:
      
      This application is used to maintain project details for the team. Projects include milestones, where you can define due dates. Tasks can be defined against a milestone or directly against the project. As well as tracking dates, this application also tracks budget, for the project, and costs against each task.

   ![Advanced Settings](image)

   c. Click **Save Changes**.

4.6 Completing the Create Application Wizard

To complete the Create Application Wizard:
1. Compare your app with the following screenshot:
2. Click **Create Application**.

A progress bar displays. When complete, Application Express navigates to the Application home page in App Builder.
4.7 Running the Application for the First Time

Your app is functionally ready to run.

When you "run" an app, you are telling Application Express to open a new browser window and render the app. The rendered app is called the “application runtime” (or just "runtime"). As you create new pages, you can run them individually, or run an entire application.

Run the app for the first time from the Application home page and review the application runtime.

To run the application from the Application home page:

1. Click Run Application.

A Sign In page appears.

2. Enter your Application Express credentials and click Sign In.

Your application opens to the Home page.
3. Explore the application by clicking on menu items and navigating to different pages in the navigation pane.
   The sample tables and data display on various pages in a raw format. The rest of this tutorial guides you through the development process to improve both the presentation and functionality of this data for your end users.

   Proceed to the next lesson.
Developing the Projects Page

Use the Create Page Wizard and Page Designer to add a sophisticated form page for the Projects report.

About This Lesson

In this lesson, you are introduced to the two primary means of development: Page Designer and the Application Runtime ("runtime" for short).

In Page Designer, use the Rendering tab and Layout tab to add new regions to the Projects page. Then, use the Property Editor to edit a wide array of attributes for region and page components, including SQL statements and appearance options.

In the runtime, you use the Runtime Developer toolbar to make changes on the page immediately and to quickly shift back to Page Designer.

Starting from here:

5 - Projects.sql
If you have not completed the previous lessons, you can use the appendix "Importing an Application into a Workspace" to import the above SQL script into your workspace and continue working from this lesson forward.

5.1 Navigating to Page Designer from the Application Runtime

At the end of the previous lesson, you ran the application from the Application home page in App Builder and opened the Home page of the application.

Running an application from App Builder also displays the Runtime Developer toolbar at the bottom of the window.

You can use the Runtime Developer toolbar to quickly edit the page (and some aspects of the entire application). This toolbar only displays for users with developer or administrator privileges.

To view the Home page in Page Designer using the Runtime Developer toolbar:

1. In the top-left of the application runtime, do one of the following:
   - Click the name of the app Demo Projects.
• Click the **Home** link in the sidebar. (Do not click Home on the Developer Toolbar.)

The runtime Home page displays.

2. On the Runtime Developer toolbar, click **Edit Page 1**.

---

### Note:

The number in the **Edit Page #** button corresponds to its assigned page number in Page Designer.

---

Page Designer displays page 1.

---

### Tip:

Page Designer is the primary asset for Application Express developers. It’s a powerful IDE (integrated development environment) with a robust array of tools for building, maintaining, and enhancing applications. Up to this point, you have mainly experienced the Create Application Wizard and SQL Workshop.

---

You rely primarily on Page Designer for the rest of this tutorial.

You can always learn more about Page Designer with the official *Oracle Application Express App Builder User’s Guide*.

---

## 5.2 Creating a Form Page Using the Create Page Wizard

Use the Create Page Wizard to add new pages to an app. It has a more robust interface for adding new pages than the Create Application Wizard.
Add a form to the Projects report to give your users a way to edit the information themselves.

To open the Create Page Wizard in Page Designer:

1. Click the **Create** icon (top-right) and select **Page**.

   ![Create Page Wizard](image)

   The Create Page Wizard appears.

2. Select **Form**.

3. Select **Form** (again).

4. Enter the following:
   a. **Page Number** - enter 4
   b. **Page Name** - enter **Maintain Project**
   c. **Page Mode** - select **Modal Dialog**
   d. Click **Next**.

5. For Navigation Preference, enter the following:
   a. Select **Do not associate this page with a navigation menu entry**.
   b. Click **Next**.

6. For Data Source, enter the following:
   a. **Table / View name** - click the **List of Values** icon and select **SAMPLE$PROJECTS (table)**
   b. Click **Next**.

7. For Create Form - Columns and Primary Key, enter the following:
   a. **Primary Key Type** - select **Managed by Database (ROWID)**
   b. Click **Create**.

Page Designer reloads displaying Page 4: Maintain Project.

**5.3 Linking Project Cards to Maintain Project Form**

You can now update the SQL queries on the Projects report to call the Maintain Project form.

Reconfigure the cards on the Projects page to display Maintain Project form when clicked.
To navigate to the Projects page (page 3) in Page Designer:

1. On the toolbar, locate the Page Finder area.

   ![Page Finder icon](image)

2. Do one of the following:
   - Click the **Page Finder** icon and select 3 from the list.
   - Change the page number to 3 and press Enter.
   - Click the **Navigate to Previous Page** icon (down arrow).


To update the SQL query for the Projects region:

3. Do one of the following to select the Projects region:
   - In the Rendering tab (left pane), under Regions - Content Body, select **Projects**.
   - In the Layout tab (center pane), scroll down and click the **Projects** region.

4. In Property Editor (right pane), in the Source group, locate the SQL Query attribute.
5. Click the **Code Editor: SQL Query** icon to read the code more easily. Reviewing the SQL Query, you can see that the SQL uses alias names, such as `CARD_MODIFIERS` and `CARD_LINK`. These specific aliases are required by the Cards region to produce the correct output. The alias `CARD_LINK` can be updated to link to the new page you just created.

6. In the SQL Query field, replace the contents with the following code:

```sql
select ' ' card_modifiers,
   apex_util.prepare_url( 'f?p=&APP_ID.:4:&SESSION.:::4:P4_ROWID:' || rowid ) card_link,
   ' ' card_color,
   ' ' card_icon,
   apex_string.get_initials(name) card_initials,
   name card_title,
   description card_text,
   'Budget ' || to_char(budget,'L99G999') card_subtext
from sample$projects
```
7. Click the **Validate** icon to verify. The message **Validation successful** displays at the top of the window.

8. Click **OK**.

9. Click **Save and Run Page**.

a. If you receive a warning dialog, click **Try Again**.

**Tip:**

You can also click the "Save" button and then "Save and Run Page" to avoid this warning.

The Projects page loads in the runtime.

10. Click a project to view the Maintain Project form for that project. You should hide several fields to restrict access. For example, if one had full access to project IDs or timestamps, one could easily compromise your database—by accident or otherwise.

To quickly return to Maintain Project form in Page Designer:


### 5.4 Updating the Maintain Project Form in Page Designer

Use Page Designer to make the following updates to the Maintain Project form (page 4) to improve usability and security:

- Hide a column by changing the type to hidden
- Move an item within an existing region
- Change an item to a select list
- Change multiple items to text fields
- Create a sub region to organize items into a read-only, expandable group

To rename a region:

1. In the Rendering or Layout tab, click **Create Form**.
2. In Property Editor:
End users do not need to see the ID column in the form. Change P4_ID to a hidden column to keep it from rendering in the app at all.

To hide a column in Page Designer:

3. In the Layout tab, click the P4_ID item.

4. In Property Editor:
   a. Identification: Type - select Hidden.

To move an item in a region:

5. In the Layout tab:
   a. Right-click the P4_STATUS_ID item.
   b. Select Move To, Maintain Project, Items, P4_BUDGET, and After.

Tip:

You can also use the mouse to drag and drop elements in the Rendering tab and Layout tab.

6. Click Save.

To convert an item to a select list:

7. Ensure P4_STATUS_ID is still selected.

8. In the Property Editor:
a. Identification: Type - select **Select List**.

b. Label: Label - enter **Status**

c. List of Values: Type - select **SQL Query**.

d. List of Values: SQL Query - enter the following:

```sql
select description d, id r
from SAMPLE$PROJECT_STATUS
order by display_order
```

e. List of Values: Display Extra Values - select **No**.

f. List of Values: Null Display Value - enter the following:

```sql
- Select Status -
```
Note:

The above SQL uses the base table `SAMPLE$PROJECT_STATUS` which includes a column called `display_order` to ensure the records display in the correct order.
9. Click **Save**.

To change multiple items simultaneously:

10. Press and hold the **Ctrl** key and click the following items to select both of them:
   a. **P4_NAME**
   b. **P4_PROJECT_LEAD**

11. In the Property Editor:
   a. Identification: Type - select **Text Field**.

   ![Image of Property Editor with P4_NAME and P4_PROJECT_LEAD selected]

   **Note:**
   When selecting multiple elements, the Property Editor only displays the attributes that all the selected elements have in common.

To create a collapsible sub region:

12. In the Rendering tab, right-click the **Maintain Project** region and select **Create Sub Region**.

   The New sub region appears within the Sub Regions folder.

   ![Image of Rendering tab with Create Sub Region selected]

   ![Image of Sub Regions folder with New sub region added]
13. In the Property Editor:
   a. Identification: Title - enter Audit Details
   b. Appearance: Template - select Collapsible.
   c. Server-side Condition: Type - select Item is NOT NULL.
   d. Server-side Condition: Item - enter P4_ROWID (you can also select the page from a pop-up list).

   **Note:**
   Setting the Server-side Condition checks that the Rowid is not null and only displays the region when a user updates the record.

14. Locate and click the Appearance: Template Options button (Use Template Defaults, Expanded, Scroll - Default).

15. In Template Options:
   a. General: Remember Collapsible State - select the check box
   b. Default State - select Collapsed
   c. Style - select Remove Borders
d. Click OK.
The Template Options button reflects the current configuration: "Use Template Defaults, Remember Collapsible State, Collapsed, Remove Borders, Scroll - Default".

16. Click Save.
Rather than using drag and drop, you can use Property Editor to move multiple items into the Audit Details sub region, convert them to Display Only, and switch their appearance template all at once.

To move items into a sub region with the Property Editor:

17. Hold the Ctrl key and select the following items:
   - P4_CREATED
   - P4_CREATED_BY
   - P4_UPDATED
   - P4_UPDATED_BY

18. In the Property Editor:
   a. Identification: Type - select Display Only
   b. Layout: Region - selectAudit Details
   c. Appearance: Template - select Optional - Floating
The double periods are a shortening device for the names of sub regions in the interface. Think of it as actually reading “the sub region Audit Details.”

The items move into the sub region.

19. Click **Save**.

The Maintain Project form is a modal page. You cannot run a modal page directly from Page Designer the way you can other pages. You must navigate to the associated non-modal page in the runtime (in this case, page 3 - Projects), then open the modal page from there.

To run the modal Maintain Project page:

20. In the runtime, reload the Projects page. (Click your browser's refresh or reload button.)

21. Click a project card.

Note the changed Status field and expandable Audit Details region. Trying clicking each and observe how the page responds.
5.5 Creating a Dynamic Action

Add a Dynamic Action to the P4_STATUS_ID item select lists so that when the user changes the project Status to “Completed,” the completion date displays.

Dynamic Actions within Application Express are used to declaratively define client-side behaviors without writing JavaScript or AJAX. Instead, the Application Express engine implements the necessary code based on your declaration.

To create a Dynamic Action:

1. In the Rendering tab, right-click the P4_STATUS_ID item and select Create Dynamic Action.

   A Dynamic Actions element populates as a child to P4_STATUS.

2. In the Property Editor:
   a. Identification: Name - enter Show Completed Date
   b. Client-side Condition: Type - select Item = Value
   c. Client-side Condition: Value - enter 3

   **Note:**

   Entering 3 in the Value field (status_ID=3) targets the third item in
   the select list, which in this case is the “Completed” status.

3. In the Rendering tab, under the Show Completed Date dynamic action, select the Show element.
4. In the Property Editor:
   a. Affected Elements: Items(s) - click the LOV icon and select P4_COMPLETED_DATE (you can also enter P4_COMPLETED_DATE).

   For some types of Dynamic Actions (such as Show, Enable, and Expand Tree), you must also create the corresponding opposite action. With an opposite action, if the True When Condition affects elements one way, then the False When Condition affects elements the opposite way.

   To add an opposite action:
   5. In the Rendering tab, under Show Completed Date, right-click Show.

   The Hide dynamic action populates (Hide is the opposite of Show).

   Here, the P4_COMPLETED_DATE item displays when P4_STATUS_ID equals Completed. When any other status is selected, P4_COMPLETED_DATE is hidden.
7. Click **Save**.

### 5.6 Adding Validations to Completed Date

Page validations ensure the data entered by end users is correct. If a validation fails, then an error message is raised, and the values are not saved to the database.

Add two validations to the Completed Date item so that:

- When a user changes the status to Completed, they must also add the date.
- A user cannot enter a date in the future.

To create a new validation:

1. On the Maintain Project page in Page Designer, click the **Processing tab** (rotating arrows) at the top of the Rendering tab.

2. Right-click the **Validations** element and select **Create Validation**. New validation appears.
3. In the Property Editor:
   a. Identification: Name - enter Completed Date is Not Null
   b. Validation: Type - select Item is NOT NULL
   c. Validation: Item - enter P4_COMPLETED_DATE
   d. Error: Error Message - enter #LABEL# must have some value.
   e. Error: Associated Item - select P4_COMPLETED_DATE
   f. Server-side Condition: Type - select Item = Value
   g. Server-side Condition: Item - enter P4_STATUS_ID
   h. Server-side Condition: Value - enter 3
      This condition ensures that the validation only fires when the Status item is Completed.

**Note:**

The label from the associated item replaces the #LABEL# text within the Error Message. If you update the item label, the error message also updates automatically. This improves consistency and ensures that messages have current label references.

**Tip:**

When you click within an attribute in Property Editor, you can view examples of content (such as error messages) for that attribute by clicking the Help tab at the top of the Layout tab.

i. Click Save.

To copy a validation:

4. In the Processing tab, right-click Completed Date is NOT NULL and select Duplicate.
The item “Completed Date is NOT NULL_1” appears, selected.

5. In the Property Editor:
   a. Identification: Name - enter Completed Date is not Forward Dated
   b. Validation: Type - select PL/SQL Expression
   c. Validation: PL/SQL Expression - enter :P4_COMPLETED_DATE <= SYSDATE

   **Tip:**

   Ensure you include the colon ( : ). To reference a page item in PL/SQL, you use the :ITEM_NAME syntax. SYSDATE returns the current date from the Oracle Database.

   d. Error: Error Message - enter #LABEL# cannot be a future date
   e. Server-side Condition: Type - select Item is NOT NULL
   f. Server-side Condition: Item - enter P4_COMPLETED_DATE

6. Click Save.
7. In the runtime, reload the Projects page (reload the browser tab) and click a project card to display Maintain Project.

The form is complete.
Try the following tasks to see how the application behaves:

- Change the Status and see how the Completed Date displays or vanishes.
- Find an In-Progress project, change the status to Completed but leave the Completion Date empty, and click **Apply Changes**. (An error message should display.)
- Enter a Completion Date in the future and click **Apply Changes**. (An error message should display.)

When finished, click **Application ####** on the Developer Toolbar to return to the Application home page.
Use the Create Page Wizard and Page Designer to add an interactive grid page to the application so that users can manage project Milestones.

In this lesson, you use the Create Page Wizard to add a new interactive grid page. Similar to the older interactive report, interactive grids do not require a paired form page to change the content displayed on the page.

In addition to the `DEMO_PROJECTS` table, the app also contains the `DEMO_MILESTONES` and `DEMO_TASKS` tables. These two tables contain data for incremental milestones and tasks that help to track the progress of an entire project more precisely.

Currently, your app only supports changes to the status of entire projects. Develop the Milestones interactive grid so that your users can view and change the status of these tasks and milestones.

Add and configure a Milestones entry in the Desktop Navigation Menu so it displays across the entire app.

Use Dynamic LOVs (or lookups) to intelligently substitute information from another table by checking against the ID of the content in the original table.

Repeat some familiar tasks to clean up both the report and form in Page Designer and in the runtime.

Starting from here:

**6 - Milestones.sql**

If you have not completed the previous lessons, you can use the appendix "Importing an Application into a Workspace" to import the above SQL script into your workspace and continue working from this lesson forward. You can find this script in the `/files` subdirectory of where you unzipped the .zip file accompanying this tutorial.

### 6.1 Creating an Interactive Grid Page

Start by using the Create Page Wizard to add a new page.

To add an interactive grid page to the application:

1. On the Application home page, click **Create Page**.
   - The Create Page Wizard displays.
2. Click **Report**.
3. Click **Interactive Grid**.
4. Enter the following:
a. Page Number - enter 5  

b. Page Name - enter Milestones  

c. Click Next.

5. For Navigation Menu, click **Create a new navigation menu entry.**

6. Use the default settings and click **Next.**

7. For Report Source, enter the following:
   a. Editing Enabled - select **Yes**
   b. Table / View Name - click the LOV icon and select **SAMPLE$PROJECT_MILESTONES (table)**

   ![Create Interactive Grid](image)

   c. Click **Create**.

   The new Milestones page loads in Page Designer.

8. Click **Save and Run Page.**

The Milestones report displays in the runtime.

### 6.2 Updating the Navigation Menu for Milestones

On the Milestones page in the runtime, examine the Milestones entry in the navigation sidebar. Note that the icon is blank and that it is listed below Administration.

Elements that appear on all pages (such as the Navigation Menu) are controlled in Shared Components.

To view the Navigation Menu in Shared Components:

1. Click **Edit Page 5** on the Runtime Developer toolbar to return to Page Designer.

2. Do one of the following:
   - Above the Rendering tab, click the **Shared Components** tab. Click **Desktop Navigation Menu** and in Property Editor click **Edit Component.**
Near the Save button, click the **Shared Components** icon ( ). Under Navigation, click **Navigation Menu**, and click **Desktop Navigation Menu**.

Since the Milestones page was created after the application, it is automatically added to the bottom of the list. As a best practice, you should separate developer-only pages from the other pages rather than mix them together. Update the sequence for both Administration and Milestones for them to appear in the right order.

To update the sequence for the Administration list entry:

3. Click **Administration**.
4. Enter the following:
   a. **Sequence** - enter **999**
      Setting the sequence to such a high number helps ensure Administration always appears at the bottom of the list when adding pages in the future.
   b. Click **Apply Changes**.

Next, update the sequence and the icon from the blank default for Milestones.

To update the Milestones list entry:

5. Click **Milestones**.
6. Enter the following:
   a. Sequence - enter 50
      Use 50 to stay in sync with the page number (5).
   b. Image/Class - click the LOV icon and select fa-table-pointer
      
      **Tip:**
      Use the search function in your browser to find the icon easily (Ctrl + F or Cmd + F).
   c. Click Apply Changes.

7. Click the Run icon in the top-right corner to run the app. Note the changes to the navigation sidebar.

### 6.3 Hiding Columns in the Application Runtime

Develop the Milestones interactive grid on page 5 in the application runtime by hiding several columns from view and moving the Due Date column.

**Tip:**
Oracle recommends hiding columns rather than deleting them in case you need them in the future.

In the runtime, you can hide columns using the Column Heading menu or the Actions menu.
To quickly hide a column in an interactive grid:
1. Click the **Id** column heading and select **Hide**.

To hide multiple columns in an interactive grid:

2. Click the **Actions** menu and select **Columns**.

3. In the list of columns on the left, click the check boxes adjacent to the following columns to deselect them:
   - Created
   - Created By
   - Updated
   - Updated By
4. Click Save.

5. To move a column, do one of the following:
   - Drag and Drop Column Heading
     a. Hover the mouse over the Due Date column heading. Note the thin dotted columns that display at the start of the heading. These are the grip handle.
     b. Click and hold the grip handle and drag and drop the Due Date column between Project Id and Name.
   - Columns Dialog in Actions Menu
     a. Click the Actions menu and select Columns.
     b. Click the Due Date column to select it. (Do not click the check box.)
     c. Use the Move Up and Move Down buttons below the list to move Due Date below Project Id.
     d. Click Save.

To save your changes to the report:

6. Click Actions, select Report, and click Save.

### 6.4 Adding Project Names to the Milestones Interactive Grid

While the Milestones interactive grid is based on SAMPLE$PROJECT_MILESTONES, note that none of the columns provide alphabetical names for the projects. Project names are stored in another table: the NAME column in SAMPLE$PROJECTS. Report regions can query other tables with a Dynamic LOV. This is also called a lookup.

First, create the Dynamic LOV in Shared Components and configure it to query SAMPLE$PROJECTS.
Then use Page Designer to convert the PROJECT_ID column to a Select List, and choose the Dynamic LOV / lookup to display the alphanumerical name of the referenced project instead of the numerical ID.

To return to Page Designer:
1. Click **Edit Page 5** on the Runtime Developer Toolbar.

To create a new Dynamic LOV lookup:

2. Click the **Shared Components** icon (Shared Components Icon).
3. Under Other Components, click **List of Values**.
4. Click **Create**.
5. Select **From Scratch** and click **Next**.
6. For Name and Type:
   a. **Name** - enter **Project Name**
   b. **Type** - select **Dynamic**
   c. **Click Next**.
7. Delete the Query contents then copy and paste the following code:

   ```sql
   select NAME d, ID r
   from SAMPLE$PROJECTS
   order by NAME
   ```

8. Click **Create List of Values**.
   The Lists of Values screen redisplay with the **Project Name** item.
9. Click the **Edit Page 5** icon in the top-right to return to the Milestones page in Page Designer.

   ![Edit Page 5 Icon](images/edit-page-5-icon.png)

To convert a column to a Select List:
10. In the Rendering tab, under Milestones, expand **Columns** and select **PROJECT_ID**.

11. In Property Editor:
   a. **Identification** - Type - select **Select List**
   b. **Heading** - Heading - enter **Project**
   c. **List of Values** - Type - select **Shared Component**
   d. **List of Values** - List of Values - select **Project Name**
   e. **List of Values** - Display Extra Values - check **No**
   f. **List of Values** - Null Display Value - enter the following:

         - Select Project -

12. Click **Save and Run Page**.
13. Review your changes to the Project column in the Runtime. When finished, click Edit Page 5 to return to Page Designer.

6.5 Changing the Appearance of an Interactive Grid in Page Designer

Perform some routine clean-up tasks to make the interactive grid more user-friendly:

- Change the heading of the Name column to instead refer specifically to Milestones.
- Convert the audit columns to Display Only to prevent end users from changing the contents.
- Change the Appearance Template of the interactive grid to the template for interactive reports. This removes the header and some UI decorations to simplify the look.

To change a column heading:
1. In the Rendering tab, under Milestones, expand Columns.
2. Select NAME.
3. In Property Editor:
   a. Identification: Type - select Text Field
   b. Heading: Heading - enter Milestone

To change multiple columns to Display Only:
4. Hold Ctrl and select the following items:
   a. CREATED
   b. CREATED_BY
c. UPDATED

d. UPDATED_BY

5. In Property Editor:
   a. Identification: Type - select Display Only

To change the Appearance Template for a region:

6. In the Rendering tab, under Content Body, select the Milestones region.

7. In Property Editor:
   a. Appearance: Template - select Interactive Report

8. Click Save and Run Page to enter the runtime.

---

6.6 Finalizing the Milestones Interactive Grid in the Runtime

Perform some routine clean-up tasks to complete the default Milestones report:

- Sort the report so that it displays projects alphabetically.
- Save your changes to the report layout as the new default.

To set multiple columns to sort ascending:

1. Hover over the Project column heading and click the Sort Ascending icon (up arrow).

2. Hold the Shift key, hover over the Due Date column heading, and click the Sort Ascending icon (up arrow).

   Icons display next to the Project and Due Date column names indicating their sort order.
To save the report layout as the new default:

3. Click the **Actions** menu, select **Report**, and select **Save**. You cannot use the Save button at the top of the report for this step because it does not overwrite the report default.

4. Click **Application ######** on the Runtime Developer toolbar to return to the Application home page.

The Milestones page is complete. Proceed to the next lesson.
Developing the Tasks Page

Use the Create Page Wizard to create a new Report with Form for end users to manage their project tasks.

About This Lesson

In this lesson, you add a new pair of pages to your app: Report with Form.

"Report with Form" (or "Report and Form") is the most common page construct within web applications. End users view the report and submit changes with a form page.

In your app, the Tasks report displays details for work to be done toward the associated project (based on the SAMPLE$PROJECT_TASKS table). The Update Task form enables users to update the task, such as changing the status or moving the due date.

Use the Create Page Wizard to generate the pair of pages. Begin with the Report page. Repeat some familiar tasks to develop the report (update Navigation Menu in Shared Components, hide extraneous columns, apply lookups to the report).

Apply a format mask for columns with set date and numbering formats.

Use the decode SQL statement to convert all instances of "Y" or "N" in a Yes/No column to display instead as Yes or No.

For the modal Form page, use Page Designer to refine the fields to mirror their counterparts in the Report. Many of the procedures for the Form are very similar to those for the Report.

Starting from here:

7 - Tasks.sql

If you have not completed the previous lessons, you can use the appendix "Importing an Application into a Workspace" to import the above SQL script into your workspace and continue working from this lesson forward. You can find this script in the /files subdirectory of where you unzipped the .zip file accompanying this tutorial.

7.1 Creating an Interactive Report Page

Use the Create Page Wizard to add a new interactive report with form page to your app. Base it on SAMPLE$PROJECTS_TASKS.

To add an interactive report with form to the application:

1. In the Application home page, click Create Page.
2. Click Report.
3. Click Report with Form.
4. For Page Attributes:
   a. Report Page Number - enter 6
   b. Report Page Name - enter Tasks
   c. Form Page Number - enter 7
   d. Form Page Name - enter Update Task
   e. Form Page Mode - select Modal Dialog
   f. Click Next.
5. For Navigation Menu:
   a. Navigation Preference - select Create a new navigation menu entry
   b. Click Next.
6. For Data Source:
   a. Table / View Name - click the LOV icon.
   b. Select SAMPLE$PROJECTS_TASKS
   c. In the list of Columns, double-click ID (Number) to move it to the left column (this removes it from display in the report).
      The ID column contains the numeric identifier for each row in the table. Users do not need to view or access this information at any time. Since you would hide the columns in both pages later anyway, this step saves you time.
   d. Click Next.
7. For Form Page:
   a. Primary Key Column - select ID (Number)
      Note that the wizard removes ID to match the columns in the report in the previous screen.
   b. Click Create.

The wizard creates the pages. Page Designer loads the interactive report page (page 6, Tasks).

The page contains the interactive report region Report 1.

To rename a report region:
8. Click Report 1 in Layout tab or Rendering tab.

9. In Property Editor:
   a. Identification: Title - enter Tasks

10. Click Save and Run Page to view the interactive report in the runtime.

In the runtime, note that the Tasks item in the Navigation Menu (left side) displays last and has a blank icon. Note also that as you scroll right, the report displays numerical identifiers in several columns as well as audit detail columns.

Perform the same tasks as you did for Milestones to update the Tasks page.

7.2 Updating the Tasks Page in the Runtime

Complete some routine tasks in the runtime to perform an initial update to the Tasks page (similar to those that you performed for the Milestones page, except Tasks uses an interactive report instead of an interactive grid):
• Update the Navigation Menu in Shared Components
• Hide the audit columns from the default report in the runtime

To view the Navigation Menu in Shared Components:
1. Click Edit Page 6 on the Runtime Developer toolbar to return to Page Designer.
2. Near the Save button, click the Shared Components icon.

To update the Tasks list entry:
5. Click Tasks.
6. For List Entry:
   a. Sequence - enter 60
   b. Image/Class - click the LOV icon and select fa-table
   c. Click Apply Changes.
7. Click the Run icon (play button) in the top-right corner to run the app. Note the changes to the navigation sidebar.

Hide the audit columns from displaying in the default report in the runtime.

To hide multiple columns in an interactive report:
8. Click the Actions menu and select Columns.
9. In the list of columns under Display in Report, double-click the following items to move them to Do Not Display:
   • Created
   • Created By
   • Updated
   • Updated By

![Select Columns](image)
10. Click **Apply**.

To save the report layout as the new default:

11. Click the **Actions** menu, select **Report**, and click **Save Report**.

12. In **Save Report**:
   a. Save - select **As Default Report Settings**
   b. Use the default setting (Primary) and click **Apply**.

### 7.3 Applying Lookups to the Tasks Interactive Report

The dynamic LOV you created earlier is also called a **lookup** (because the query "looks up" a value in another table).

First, copy the **PROJECT NAME** Dynamic LOV in Shared Components and configure it to query `SAMPLE$PROJECT_MILESTONES`.

Then apply the two lookups to the **PROJECT_ID** and **MILESTONE_ID** columns.

To copy an LOV:

1. Click **Edit Page 6** on the Runtime Developer toolbar to return to Page Designer.
2. Click the **Shared Components** icon.
3. Under Other Components, click **List of Values**.
4. Click **Copy**.
5. For Copy List of Values:
   a. Copy List of Values - select **PROJECT NAME - Dynamic**
   b. New List of Values Name - enter **Milestone Name**
   c. Click **Copy**.

The Lists of Values screen redisplays with the **MILESTONE NAME** item.

6. Click **MILESTONE NAME**.

7. Under Query, delete the contents then copy and paste the following code:

   ```sql
   select NAME d, ID r
   from SAMPLE$PROJECT_MILESTONES
   order by NAME
   ```

8. Click the **Validate** icon (checkmark) to confirm the query works.

9. Click **Apply Changes**.

10. Click the **Edit Page 6** icon in the top-right to return to the Tasks page in Page Designer.

Apply the Project Name dynamic LOV to the **PROJECT_ID** column (similar to the process for the Milestones page). Then apply the Milestone Name dynamic LOV to **MILESTONE_ID**.

To apply the **PROJECT NAME** LOV to an interactive report:

11. In the Rendering tab, under Tasks, expand **Columns** and select **PROJECT_ID**.

12. In Property Editor:
a. Identification: Type - select **Plain Text (based on List of Values)**
b. List of Values: List of Values - select **PROJECT NAME**
c. Heading: Heading - enter **Project**
d. Click the **Save** button.

To apply the MILESTONE NAME LOV to an interactive report:

13. In the Rendering tab, under Columns, select **MILESTONE_ID**.

14. In Property Editor:
   a. Identification: Type - select **Plain Text (based on List of Values)**
   b. List of Values: List of Values - select **MILESTONE NAME**
   c. Heading: Heading - enter **Milestone**

15. Click **Save and Run Page**.

16. Scroll down the report to review the Project and Milestone columns in the Runtime. When finished, click **Edit Page 6** to return to Page Designer.

### 7.4 Updating the Remaining Columns in the Tasks Report

Update the other remaining columns in the Tasks interactive report from their raw state generated by the Create Page Wizard:

- **NAME**
- **START_DATE**
- **END_DATE**
- **COST**
- **IS_COMPLETE_YN**

To update a column heading in an interactive report:

1. In the Rendering tab, under Tasks, expand **Columns** and select **NAME**.

2. In Property Editor:
   a. Heading: Heading - enter **Task**

To apply a date format mask to multiple columns:

3. In the Rendering tab, hold Ctrl and select the two following columns:
   a. **START_DATE**
   b. **END_DATE**

4. In Property Editor:
   a. Appearance: Format Mask - click the **LOV** icon
   b. Select **12-JAN-2004**.

**Tip:**

You can also enter **DD-MON-YYYY** in the field without opening the list.
To apply a dollar amount format mask to a column:

5. In the Rendering tab, select **COST**.

6. In Property Editor:
   a. Appearance: Format Mask - click the **LOV** icon
   b. Select **$5,234.10**.
   c. Click **Save**.

To configure an interactive report with a _YN column to display “Yes” and “No” (instead of Y or N):

7. In the Rendering tab, select **IS_COMPLETE_YN**.

8. In Property Editor:
   a. Heading: Heading - enter **Complete**
   b. Click **Save**.

9. Click the **Tasks** report region (under Content Body in the Rendering tab).

10. In Property Editor:
    a. Source: Type - select **SQL Query**.
    b. Source: SQL Query - delete the contents then copy and past the following code:

```sql
select ID,
    PROJECT_ID,
    MILESTONE_ID,
    NAME,
    DESCRIPTION,
    ASSIGNEE,
    START_DATE,
    END_DATE,
    COST,
    decode(is_complete_yn, 'Y', 'Yes', 'No') IS_COMPLETE_YN,
    CREATED,
    CREATED_BY,
    UPDATED,
    UPDATED_BY
from SAMPLE$PROJECT_TASKS
```

The **decode** statement is used to compare the column value against the first value ("Y") and replace it with the next value ("Yes"), and otherwise replace it with the final value ("No").

11. Click **Save and Run Page**.
12. Review your changes to the report in the runtime. When finished, proceed to the next section.

7.5 Updating the Project and Milestone Fields in the Update Task Form

When you created the Tasks Report with Form in the Create Page Wizard, you chose to make the form modal. (If you had left the check box unselected, it would be a separate page in the runtime instead of a pop-up overlay.) To view a modal page in the runtime, you must open it from its associated report.

Open the Update Task form by clicking one of the Edit icons in the Tasks interactive report.

To view a modal form for an interactive report in the runtime:

1. Click the Edit icon (pencil) for a row.
The Update Task form displays.

Note also that the page number on the Runtime Developer toolbar changed from the report (page 6) to the form (page 7).

The form is still in a raw state, but because it is based on SAMPLE$PROJECT_TASKS, the development process is very similar to what you did for the report:

- The Milestone and Project name fields use numeric identifiers instead of names.
- The names of some fields (such as "Project Id") should be user-friendly.
- Some fields are editable text fields instead of restricted select lists or display only.
- The audit columns do not need to display here (they are present in the report).

Start by applying the PROJECT NAME and MILESTONE NAME lookups to the appropriate fields.
To return to Page Designer using Quick Edit:

2. On the Runtime Developer toolbar, click **Quick Edit**.

   You enter Quick Edit mode. The screen darkens and when you mouse over different elements a wrench icon displays.

3. In Quick Edit mode, click the **Project Id** field (do not click the wrench icon).

Page Designer loads with P7_PROJECT_ID selected.

To apply lookups to the Project and Milestone fields:

4. In Property Editor:
   a. Identification: Type - select **Select List**.
   b. Label: Label - enter **Project**
   c. List of Values: Type - select **Shared Component**
   d. List of Values: List of Values - select **PROJECT NAME**

5. Click **P7_MILESTONE_ID**.

6. In Property Editor:
   a. Identification: Type - select **Select List**.
   b. Label: Label - enter **Milestone**
   c. List of Values: Type - select **Shared Component**
   d. List of Values: List of Values - select **MILESTONE NAME**

7. Click **Save**.
   Remember that you cannot run a modal page straight from Page Designer.

8. Return to the runtime and refresh the page, then click an **Edit** icon (pencil) to redisplay the form.
   (Remember that you cannot run a modal page directly from Page Designer. You must run the non-modal page associated with it, then reopen it.)

9. Note the changed Project and Milestone fields. Click each to see how they behave.

10. When finished, click **Edit Page 7** on the Runtime Developer toolbar.
7.6 Updating the Remaining Fields in the Update Task Form

Finally, perform a series of small tasks to update the Update Task form's presentation:

- Name - change the Name field to a text field and change the label to Task.
- Assignee - change to a select list based on SQL query.
- Start Date and End Date - change the format mask to DD-MM-YYYY.
- Cost - change the alignment and format mask ($5,234.10).
- Is Complete Yn - change the label and type to a switch.
- Hide the audit columns (Created, Created By, Updated, Updated By).

To update the P7_NAME field:
1. Click **P7_NAME**.
2. In Property Editor:
   a. Identification: Type - select **Text Field**.
   b. Label: Label - enter Task

To update the P7_ASSIGNEE field:
3. Click **P7_ASSIGNEE**.
4. In Property Editor:
   a. Identification: Type - select **Select List**.
   b. List of Values: Type - select **SQL Query**.
   c. List of Values: SQL Query - copy and paste the following code:

   ```sql
   select distinct ASSIGNEE as d,
               ASSIGNEE as r
   from SAMPLE$PROJECT_TASKS
   order by 1
   ```
   d. Click **Save**.

To update the P7_START_DATE and P7_END_DATE fields:
5. Hold Shift or Ctrl and click **P7_START_DATE** and **P7_END_DATE** to select them both.
6. In Property Editor:
   a. Appearance: Format Mask - click the **LOV** icon.
   b. Select **12-JAN-2004**.
   c. Settings: Show - select **Both**.
7. Click **P7_END_DATE** to only select it.
8. In Property Editor:
   a. Layout: Start New Row - select **No**.
   b. Click **Save**.

To update the P7_COST field:
9. Click **P7_COST**.

10. In Property Editor:
    a. Settings: Number Alignment - select **Left**.
    b. Appearance: Format Mask - click the **LOV** icon.
    c. Select **$5,234.10**.
    d. Click **Save**.

To update the **P7_IS_COMPLETE_YN** field:

11. Click **P7_IS_COMPLETE_YN**.

12. In Property Editor:
    a. Identification: Type - select **Switch**.
    b. Label: Label - enter **Complete**

To hide the audit column fields:

13. Hold Ctrl or Shift and select the four following items:
    - **P7_CREATED**
    - **P7_CREATED_BY**
    - **P7_UPDATED**
    - **P7_UPDATED_BY**

14. In Property Editor:
    a. Identification: Type - select **Hidden**.
    b. Click **Save**.

15. Return to the runtime and refresh the page.

16. Click an **Edit** icon (pencil) to redisplay the form.
17. Review your changes. Try clicking around the form to explore the new interface. When finished, click Application ###### on the Runtime Developer toolbar to return to the Application home page.

The Tasks report with form is complete. Proceed to the next lesson.
Developing the Dashboard Page

Update and add new regions to the Dashboard page with Page Designer.

About This Lesson

In this lesson, you modify the regions of an existing page without the aid of a wizard.

You develop the Dashboard page generated by the Create App Wizard by deleting all but one of the existing chart regions. You update the series, attributes, and Template Options of the remaining chart region to improve usability and appearance.

You add a new report region from scratch using the Gallery instead of a wizard. Reconfigure the report's SQL query and Template Options.

Starting from here:

8 - Dashboard
If you have not completed the previous lessons, you can use the appendix "Importing an Application into a Workspace" to import the above SQL script into your workspace and continue working from this lesson forward. You can find this script in the /files subdirectory of where you unzipped the .zip file accompanying this tutorial.

8.1 Deleting Regions in Page Designer

You created multiple charts for the Dashboard in the Create App Wizard. Review the output in the runtime.

To view the runtime:

1. On the Application home page, click 2 - Dashboard.
2. Click Save and Run Page.

Note the four charts on the page. You decide that the bar chart is the best way to present Costs info rather than a line or pie chart (Chart 3 and Chart 4). You also decide to replace one chart (Project Status) with a report later.
Start by deleting the charts you no longer want, then update the remaining chart with another series to display more budget information.

To delete multiple regions in Page Designer:

3. Click **Edit Page 2** on the Runtime Developer toolbar.
4. In Page Designer, hold Ctrl and click the three following items to select them:
   - **Project Status**
   - **Chart 3**
   - **Chart 4**
5. With the three chart regions selected, do one of the following:
   • Press **Delete**.
   • Right-click one of the selected chart regions and select **Delete**.
   Only Project Costs remains in Content Body.

6. Click **Save and Run Page**.

### 8.2 Updating the Series in the Project Costs Chart

The Project Costs chart only displays costs. This chart would be more meaningful if it also displayed project budgets for comparison.

First, update the SQL for the existing series in the chart to change the display order of the columns (among other updates), then create a new series in the chart to display project budgets.

To update the first series:

1. Click **Edit Page 2** on the Runtime Developer toolbar.
2. Click **Project Costs**.
3. In Property Editor:
   a. Identification: Title - enter **Project Budgets and Costs**
4. In the Rendering tab, under Series, click **Series 1**.
5. In Property Editor:
   a. Identification: Name - enter **Costs**
   b. Source: SQL Query - click the **Code Editor** icon ( ).
c. Copy and paste the following code:

```sql
select p.name, sum(cost) value 
from sample$project_tasks t 
, sample$projects p 
where p.id = t.project_id 
group by p.name 
order by 1
```

d. Click OK.

An error message displays under Column Mapping: Value.

Note:

The above SQL is a significant update to the old code. This revised SQL joins the `SAMPLE$PROJECT_TASKS` and `SAMPLE$PROJECTS` tables so the chart displays the project name instead of the project ID. This requires including the `where` condition to link the primary key column from `Projects` (p.id) with the foreign key column from `Tasks` (t.project_id). Finally, the `group by` clause was updated because the specified column (PROJECT_ID) no longer displays (changed to `p.name`).

e. Column Mapping: Label - select VALUE

To add a series to a chart:

6. In the Rendering tab, right-click Series and select Create Series.

7. In Property Editor:
   a. Identification: Name - enter Budgets
   b. Source: Table Name - click the LOV icon.
   c. Select `SAMPLE$PROJECTS`.
   d. Column Mapping: Label - select NAME
   e. Column Mapping: Value - select BUDGET

8. Click Save and Run Page. Review the changes to the labels along the X-axis and the new columns for each project. You can also mouse over each column for more details.
8.3 Updating the Appearance of the Dashboard Chart

While the chart is more usable now, it still needs some aesthetic improvements:

- The labels are too constrained by the space.
- The chart lacks a legend to indicate which column represents which value (cost or budget).
- The chart does not fit vertically in the region (note the scroll bar).
- The cost and budget values lack a dollar sign.

These changes are in the region attributes and axis properties for the chart.

To change the attributes of a region:

1. Click **Edit Page 2** on the Runtime Developer toolbar.
2. In the Rendering tab, under Project Budgets and Costs, click **Attributes**.
3. In Property Editor:
   a. Layout: Height - enter 600
   b. Legend: Show - click **Yes**
   c. Legend: Position - select **Top**
   d. Legend: Hide and Show Behavior - select **No Rescaling**

To change the properties of an axis:

4. In the Rendering tab, under Axes, click **y**.
5. In Property Editor:
   a. Value: Format - select **Currency**
   b. Value: Currency - enter $  
6. Click **Save and Run Page**.
8.4 Updating the Live Template Options for the Dashboard Chart

If you scroll down the chart region in the runtime, you can see your changes. (You may need to maximize your window to view all the labels properly.) The individual elements look good, but obviously you must alter the way the region displays on the page.

The final changes you need to make are in the Live Template Options, which you access using Quick Edit on the Runtime Developer toolbar.

To open Live Template Options:
1. Click Quick Edit on the Runtime Developer toolbar.
   You enter Quick Edit mode (screen darkens, mouse cursor becomes a crosshair).
2. Mouse over the Project Budgets and Costs chart region so that a wrench icon displays (top-right).

3. Click the wrench icon.
   The Live Template Options dialog displays.
4. In Live Template Options:
   a. Body Height - select **Auto - Default**
   b. Click **Save**.

The dashboard chart is complete.
Next, add a report region to replace the Project Status chart you deleted.

### 8.5 Adding a Classic Report Region from Gallery

Add a report to the page to display incomplete tasks.
Located beneath the central Layout tab, the **Gallery** is a quick way to peruse your options for a variety of elements.
To add a classic report region from the Gallery:

1. Click **Edit Page 2** on the Runtime Developer toolbar to return to Page Designer.

2. In the Gallery, right-click **Classic Report**, select **Add To**, select **Content Body**, select **Project Budgets and Costs**, and select **After**.

   **Tip:**
   
   You can also add elements from the Gallery with the mouse using drag and drop.

3. In Property Editor:
   
   a. Identification: Title - enter **Incomplete Tasks**
   
   b. Source: Type - select **SQL Query**
   
   c. Source: SQL Query - copy and paste the following code:

   ```sql
   select p.name project,
          m.name milestone,
          m.due_date,
          t.name task,
          t.start_date,
          t.end_date
   ```
from sample$projects p
, sample$project_milestones m
, sample$project_tasks t
where p.id = m.project_id
and p.id = t.project_id
and t.milestone_id = m.id (+)
and nvl(t.is_complete_yn, 'N') = 'N'

4. Click **Save and Run Page**. Scroll down to view the report in the runtime.

The report displays an the appropriate amount of information and columns, but it has some residual layout problems. You can change this value in Live Template Options.

To open Live Template Options in the runtime:

5. On the Runtime Developer toolbar, click **Quick Edit**.

6. Mouse over the Incomplete Tasks region and click the **wrench icon** (top-right).

7. In Live Template Options:
   a. General - check **Remove Body Padding**
   b. Click the **Attributes** tab (top).
   c. General - check **Stretch Report**
   d. Click **Save**.

The report is finished. Your application is complete. Congratulations!
Importing an Application into a Workspace

Upload a SQL script to your Oracle Application Express workspace to start at the beginning of any chapter in this tutorial.

You can use the Import functionality of App Builder to upload a script to your workspace to start at the beginning of a chapter. Each script creates an application to match the start of a chapter. Use these scripts to skip around the tutorial or to check your work prior to that chapter.

The files you upload are contained in a downloadable .zip file located here: http://www.oracle.com/technetwork/developer-tools/apex/application-express/apex19-demo-basic-5586158.zip

Each file incorporates all the changes you are instructed to make in the chapters preceding it and creates a new application named "Demo Projects" with a new application ID.

---

**Tip:**

If you already have a Demo Projects app in your workspace, you may want to rename your existing Demo Projects app by editing the Name attribute on the Edit Application Definition page. See "Editing Application Attributes" in the Oracle Application Express App Builder User’s Guide.

---

**Prerequisites**

To import an application, you must have the following:

- The .zip file containing the application export scripts: http://www.oracle.com/technetwork/developer-tools/apex/application-express/apex19-demo-basic-5586158.zip
- The sample dataset installed in "Populating Your Workspace"
- Sufficient space allocated to your workspace
  - to request more space, see "Requesting More Storage" in Oracle Application Express Administration Guide

**Procedure**

To import an application export file (*.sql):

1. On the Workspace home page, click **App Builder** to navigate to the App Builder Home Page.
2. Click **Import**.
   - The Import dialog displays.
3. For Import:
   - Import file - click the appropriate button to browse your system for a file (for example, in Chrome the **Choose File** button).
b. Locate the extracted .zip archive.

c. Select the desired script and click **Open**.

Tip:
To determine the appropriate script file name, see *Starting from here* at the beginning of a chapter.

d. For File Type, select **Database Application, Page or Component Export**.

e. Click **Next**.

4. On the File Import Confirmation page, click **Next**.

5. On the Install Database Application page, click **Install Application**.

The new application builds.

6. Click **Edit Application** to navigate to the imported application’s overview in Application Builder.

You have imported the new application into your workspace.
Index

A

alias (SQL), 5-3
App Gallery, 8-7
Appearance Template, 6-8
application runtime, 4-7, 4-9

C

Cards page, 4-2
Classic Report, 8-7
Create Application wizard
  Advanced Settings, 4-6
  Features, 4-6
  Settings, 4-6
Create Application Wizard, 4-1
  Add Page function, 4-1

D

Dashboard page, 4-3
Developer Toolbar, 5-1
Display Only, 6-8
dynamic action, 5-14
Dynamic LOV, 6-6, 7-4

F

form page, 5-2
Format Mask, 7-5

H

hidden column, 5-6

I

import an application, A-1
interactive grid, 6-1
interactive report
column heading, 7-5

L

link (SQL), 5-3
list of values (LOV), 5-6
Live Template Options, 8-6
lookup, 6-6, 7-4

N

Navigation Menu, 6-2, 7-2

O

opposite action, 5-14

P

Processing tab, 5-16
projects page, 5-2

Q

Quick Edit
  Live Template Options, 8-6

R

region, 5-6
runtime, 4-7, 4-9, 5-1

S

Save and Run Page, 4-7, 4-9
Select List, 6-6
Shared Components
  Navigation Menu, 6-2, 7-2
sub region, 5-6

T

Template
  Appearance, 6-8
  Template Options, 5-6
V

validation, 5-16