



Oracle Application Express: Developing Database Web Applications

Hands-On-Labs Guide

Unit 2: Using SQL Workshop

This exercise includes two hands-on-labs.

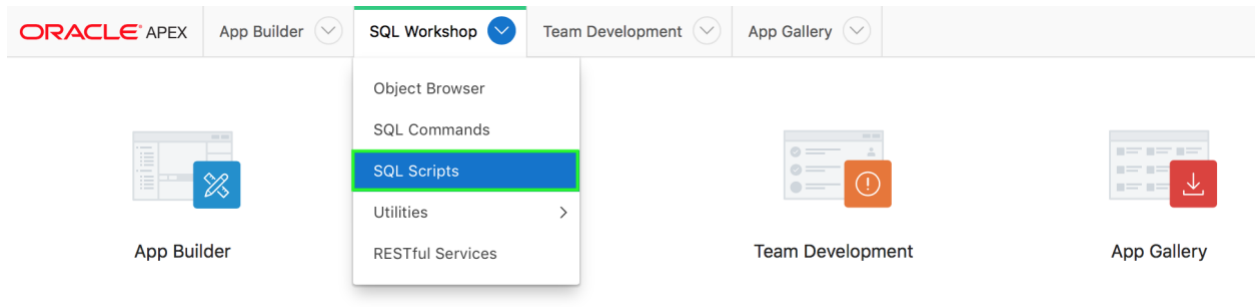
HOL 2-1: Loading the Tables and Data: In this lab, you will use SQL Workshop to create the underlying database objects and data required for you to build the Demo Projects application.

HOL 2-2: Creating a Lookup Table: In this lab, you will create a table named `HARDWARE` and load data into the table. Then, you create a lookup table.

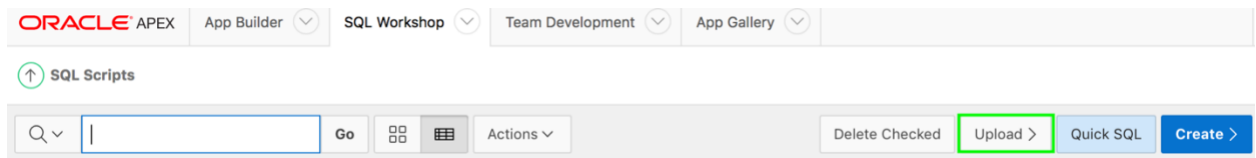
HOL 2-1: Loading the Tables and Data

It is essential to have at least the tables defined in order for the Create Application wizard to generate pages in your application. In this hands-on-lab, you create the required database objects, and populate the tables with sample data.

1. Use SQL Workshop to upload a script that creates the tables for the Demo Projects application. Perform the following steps:
 - a) Navigate to **SQL Workshop** and select **SQL Scripts**.



- b) Click **Upload**.



- c) Click **Choose File**.

Upload Script

* File **Choose File** no file selected ?

Script Name ?

File Character Set ?

- d) Navigate to the working directory where you extracted **apex-course-labs.zip**. Locate the **Project_Tables.sql** file, and double-click the file or click the file and then click

Open.
Click **Upload**.

- e) Review the uploaded script to see what tables will be created.
In the SQL Scripts list, click the Edit icon (pencil), to the left of the script you just uploaded.

↑

SQL Scripts

Q

▼

Go

Actions ▼

Delete Checked

Upload >

Quick SQL

Create >

<input type="checkbox"/>	Edit	Owner	Name	Created	Updated By	Updated ⌵	Bytes	Results	Run
<input type="checkbox"/>	<div><div></div></div>	LOWCODE	Project_Tables.sql	1 seconds ago	LOWCODE	1 seconds ago	15,367	0	<div><div></div></div>

- f) Click the **Run** icon to the right of the script you uploaded.

SQL Scripts \ Script Editor

Script Name: ?


Cancel Download Delete Save Create App **Run**

1 Rem Copyright © 2020 Oracle and/or its affiliates. All rights reserved.
 2 Rem Licensed under the Universal Permissive License v 1.0 as shown at http://oss.oracle.com/licenses/upl.
 3
 4
 5 --<< Drop existing DB Objects >>--
 6 -- Uncomment the following lines if rereunning the script
 7 -- drop table demo_proj_constraints cascade constraints;
 8 -- drop table demo_proj_status cascade constraints;
 9 -- drop table demo_proj_team_members cascade constraints;
 10 -- drop table demo_projects cascade constraints;
 11 -- drop table demo_proj_milestones cascade constraints;
 12 -- drop table demo_proj_tasks cascade constraints;
 13 -- drop table demo_proj_task_todos cascade constraints;
 14 -- drop table demo_proj_task_links cascade constraints;
 15 -- drop table demo_proj_comments cascade constraints;
 16 -- drop package demo_projects_data_pkg;
 17
 18 --<< Create all of the necessary tables for Demo Projects >>--
 19
 20
 21
 22 -- Demo Project Constraint Lookup table
 23
 24 create table demo_proj_constraints(
 25 constraint_name varchar2(30) not null
 26 constraint demo_proj_const_lookup_pk
 27 primary key,
 28 message varchar2(4000) not null)
 29 /

g) Then, click **Run Now**.

SQL Scripts \ Run Script

Schema: ?



Run Script

You have requested to run the following script. Please confirm your request.

Script Name	Project_Tables.sql
Created	on 12/11/2019 05:16:12 AM by LOWCODE
Updated	on 12/11/2019 05:19:04 AM by LOWCODE
Number of Statements	45
Script Size in Bytes	15,369

Cancel **Run Now**

- h) Scroll down to view the results of the script that you just ran.

45	45	0
Statements Processed	Successful	With Errors

2. Currently the tables you created do not have any data. A script has been provided that creates an Oracle database package which can be run at any time to insert or reset the data in the tables. Use SQL Workshop to upload a script that you can use to populate table data. Perform the following steps:

- a) Click **SQL Scripts**. Click **Upload**.

The screenshot shows the Oracle APEX SQL Workshop interface. The top navigation bar includes 'ORACLE APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'App Gallery'. Below the navigation bar, there are tabs for 'SQL Scripts' (highlighted with a green box) and 'Results'. The main area displays the execution details for a script named 'Project_Tables.sql' with a status of 'Complete'. The script is viewed in 'Summary' mode, showing 15 rows. Below this, a table lists the execution results for each statement.

Number	Elapsed	Statement	Feedback	Rows
1	0.07	create table demo_proj_constraints(constraint_name varc	Table created.	0
2	0.07	create table demo_proj_status (cd varc	Table created.	0
3	0.09	create or replace trigger biu_demo_proj_status before in	Trigger created.	0
4	0.08	create table demo_proj_team_members (id	Table created.	0

- b) Click Choose File, where you extracted **apex-course-labs.zip**.
- c) Locate the **Project_Data.sql** file, and double-click the file or click the file and then click **Open**.
Click **Upload**.
- d) Click the **Run** icon to the right of the script you uploaded (top row).

SQL Scripts

<input type="checkbox"/>	Edit	Owner	Name	Created	Updated By	Updated ↓	Bytes	Results	Run
<input type="checkbox"/>		LOWCODE	Project_Data.sql	1 seconds ago	LOWCODE	1 seconds ago	134,129	0	
<input type="checkbox"/>		LOWCODE	Project_Tables.sql	11 minutes ago	LOWCODE	8 minutes ago	15,369	1	

e) Then, click **Run Now**.

f) Scroll down, to view the results.

SQL Scripts \ Results

Script: **Project_Data.sql** Status: **Complete**

View: ☐ Detail ☒ Summary Rows: 15

Number ↑	Elapsed	Statement	Feedback	Rows
1	0.06	create or replace package demo_projects_data_pkg as functi	Package created.	0
2	0.54	create or replace package body demo_projects_data_pkg as	Package Body created.	0
3	0.81	begin demo_projects_data_pkg.load_sample_data; end;	Statement processed.	1

[Download](#)

row(s) 1 - 3 of 3

3	3	0
Statements Processed	Successful	With Errors

3. In step 2, you uploaded a package called DEMO_PROJECTS_DATA_PKG. However, this package has not yet been run so the tables you created still do not have any data. The SQL Commands facility, within SQL Workshop, allows a developer to run any valid SQL commands. You will run a SQL command to execute the data package and populate the tables. Use SQL Commands to execute Oracle Database package. Perform the following steps:

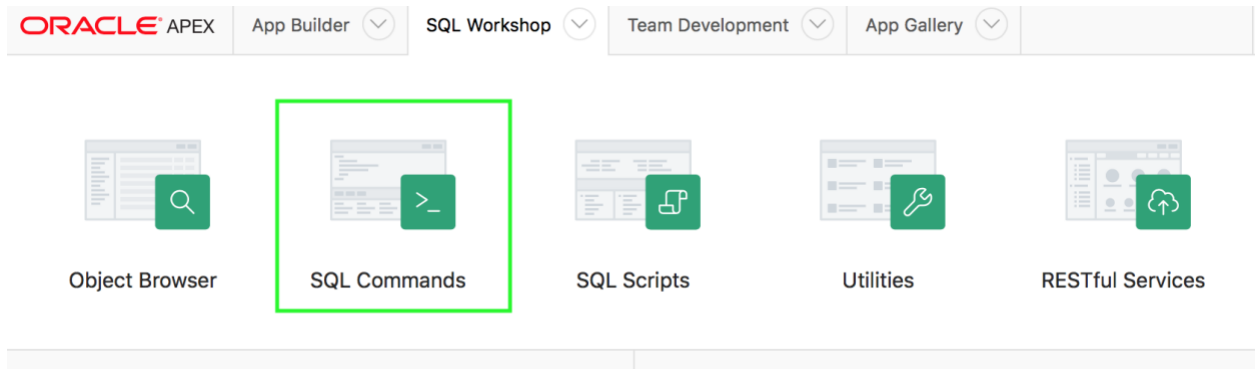
a) Click the Up arrow, before SQL Scripts.

SQL Scripts \ Results

Script: **Project_Data.sql** Status: **Complete**

View: ☐ Detail ☒ Summary Rows: 15

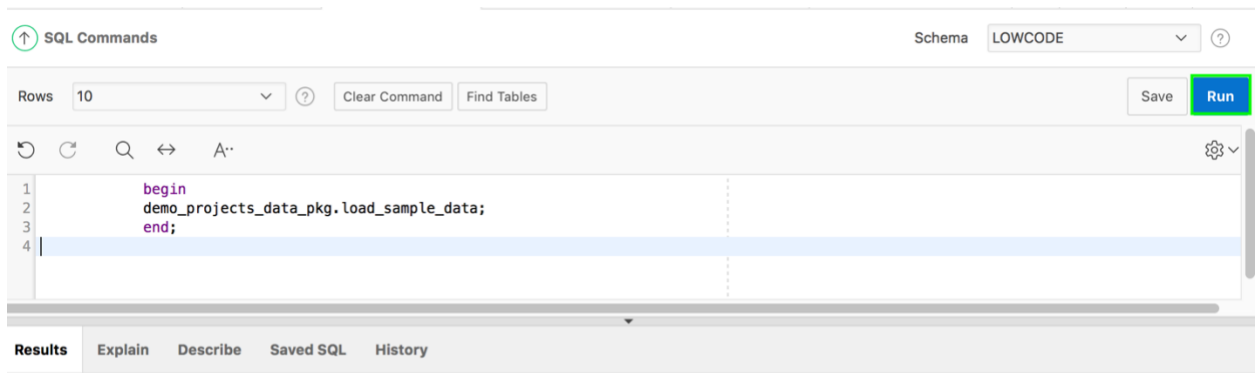
b) Click **SQL Commands**.



c) Enter the following code:

```
begin
demo_projects_data_pkg.load_sample_data;
end;
```

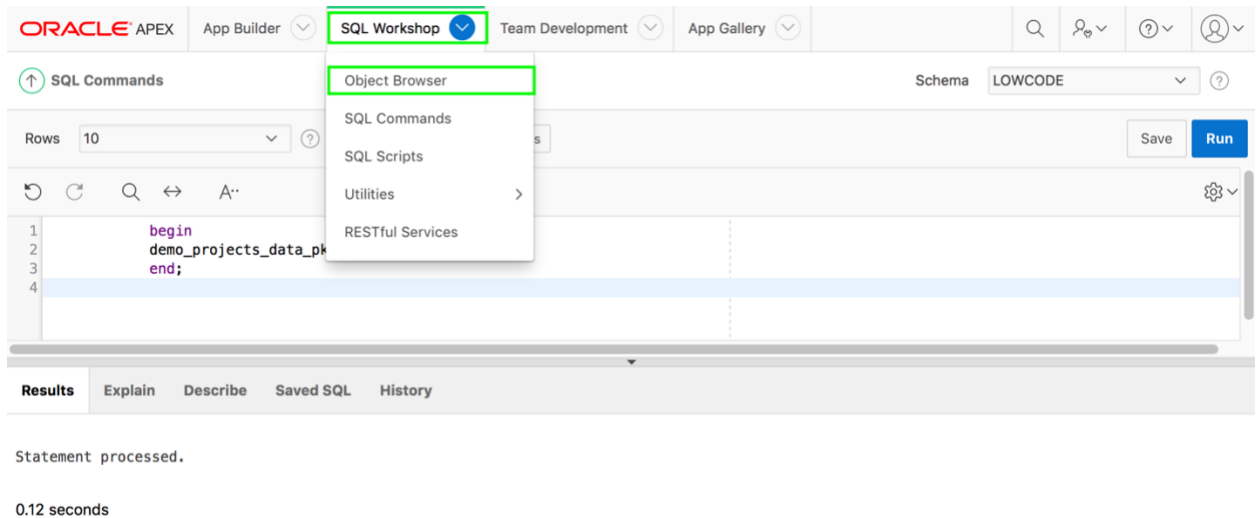
d) Click **Run**.



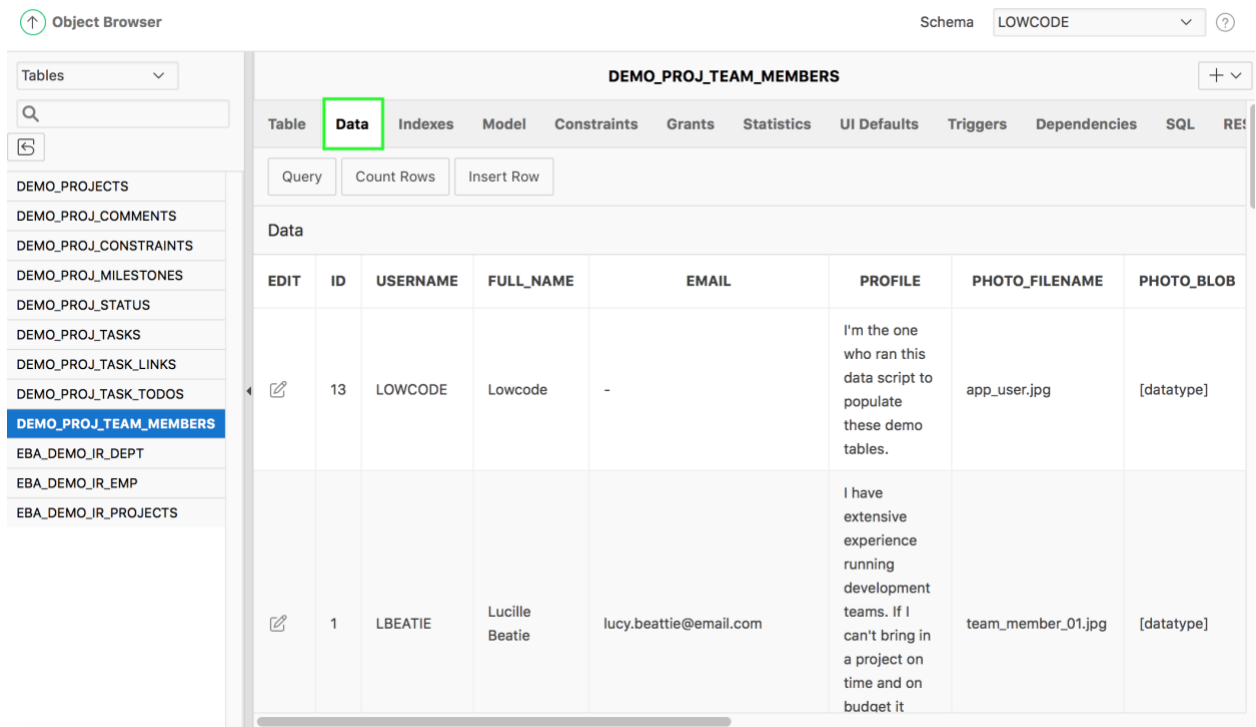
Enter SQL statement or PL/SQL command and click Run to see the results.

4. Use the Object Browser within SQL Workshop to review all of the database objects, such as the tables and packages you created, available in the underlying Oracle database schema which is associated with the Application Express workspace you logged into. Perform the following steps:

a) At the top of the page, select **SQL Workshop** and then select **Object Browser**.

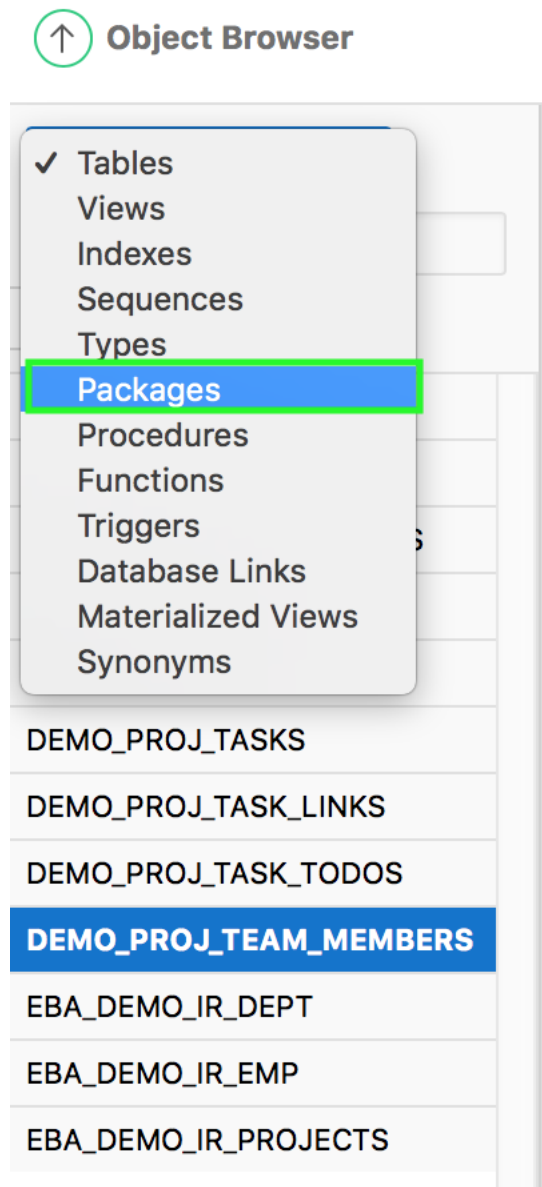


- b) In Object Browser, select the **DEMO_PROJ_TEAM_MEMBERS** table, and then click on the **Data** tab.

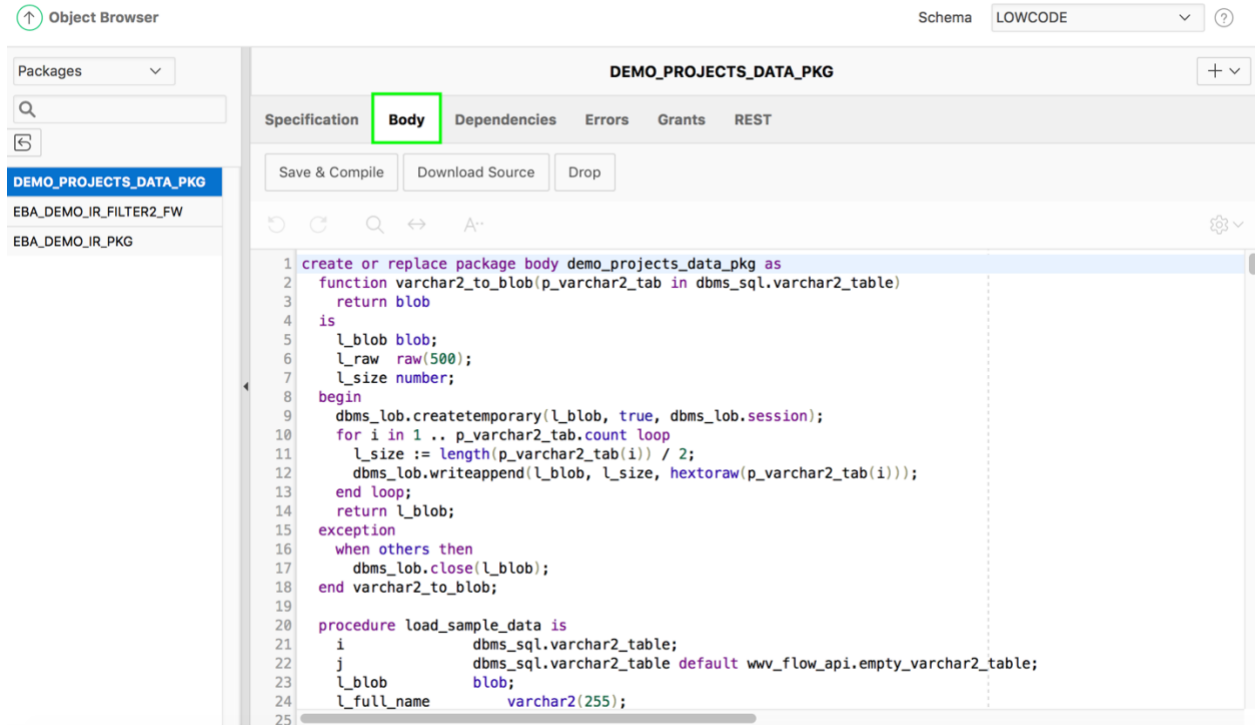


Note: There are a number of other tables listed, outside of those you created using the script file above.

- c) To review the package you created, select **Packages**.



- d) Then, select **DEMO_PROJECTS_DATA_PKG**.
- e) Click **Body** to review the primary PL/SQL rather than the specification.

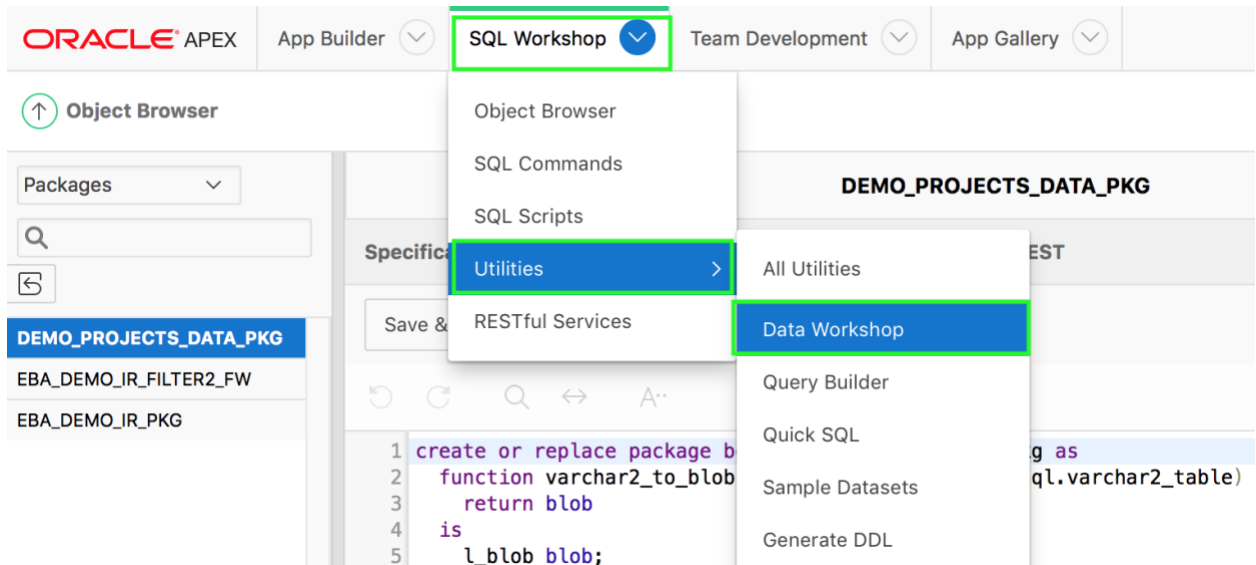


Note: This package includes complex PL/SQL code to insert images and replicate users entering in records. It is not important that you understand the PL/SQL code in this package, as you will not normally have to populate data in this matter. Generally, you would create the tables with no data and then use the application you build to insert the records.

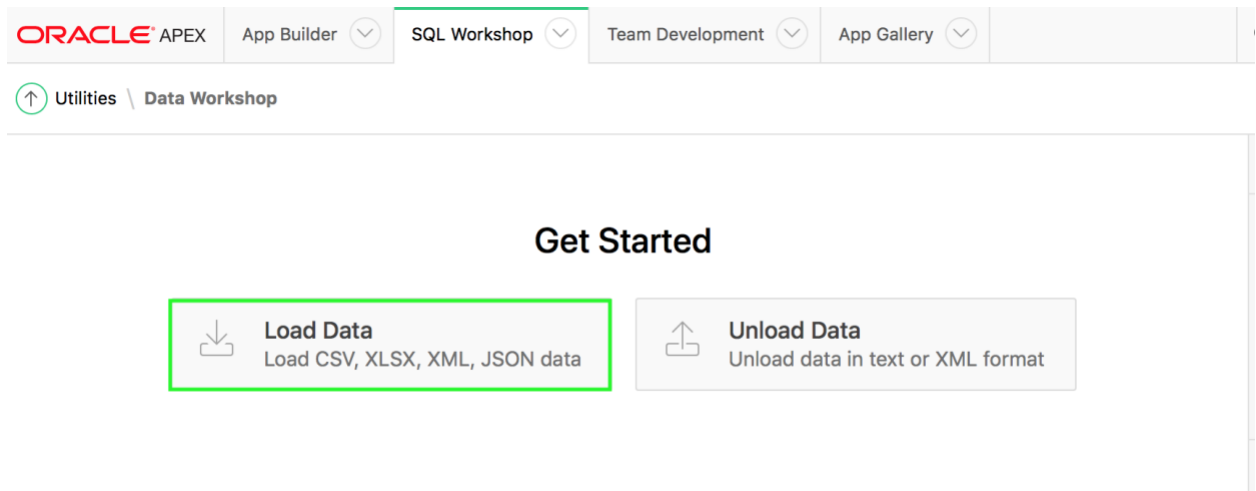
HOL 2-2: Creating a Lookup Table

In this hands-on-lab, you use the Data Workshop utility to create a table and populate the table with data. Once this table is created, you also create a lookup table.

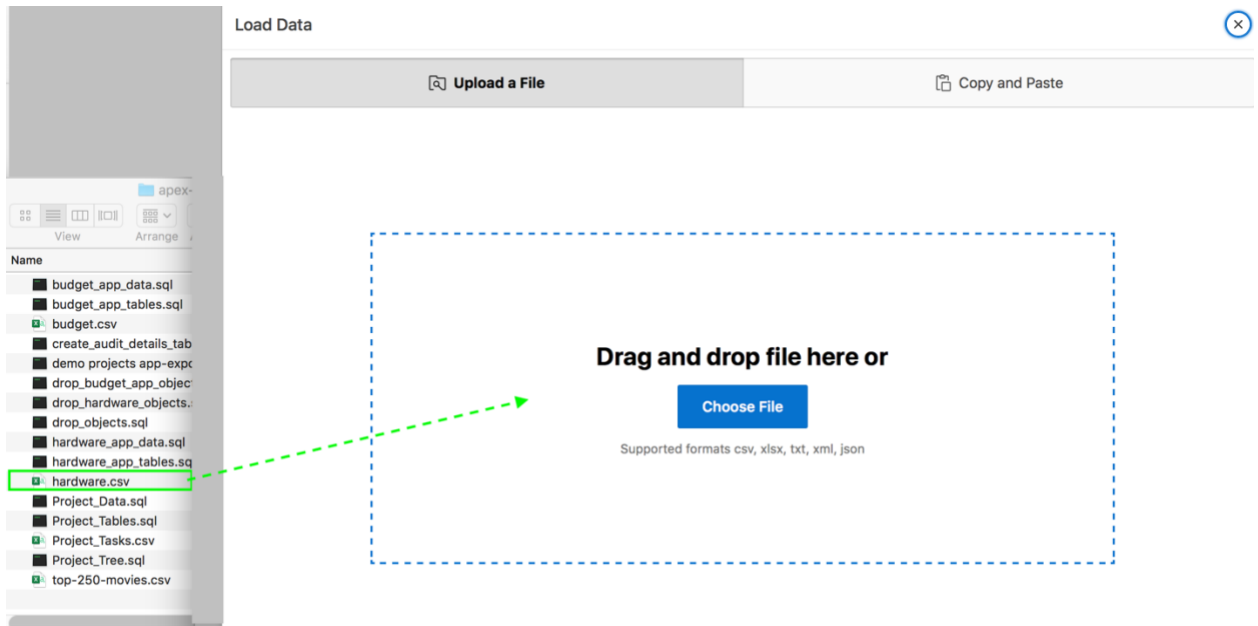
1. Click **SQL Workshop > Utilities > Data Workshop**.



2. Click **Load Data**.



3. The Load Data Wizard appears.
Navigate to the working directory where you have **apex-course-labs.zip**.
4. Locate the **hardware.csv** file, and then drag and drop the file to the **Load Data** dialog.
Alternatively, in the Load Data dialog, click **Choose File**, and then double-click the **hardware.csv** file.



5. Enter **Hardware** for Table Name, accept the remaining defaults and click **Load Data**.

Load Data

hardware.csv

Where do you want to load this data?

Load To

New Table

Existing Table

* Table Owner

LOWCODE

?

* Table Name

HARDWARE

?

Please select the columns to load.

Configure

* Error Table Name

HARDWARE_ERR\$

?

Primary Keys

SYS_GUID

Identity Column

?

☒ Use Column Data Types

?

Settings

Column Headers

☒ First line contains headers

Column Delimiter

,

;

|

#

tab

Enclosed By

None

"

'

File Encoding

Unicode UTF-8

Preview

<

Cancel

Load Data

Notice that the Error Table Name is populated.

6. The new table is now created and is populated with the data. Click **View Table**.

Load Data

Table **HARDWARE** created with 37 rows!

View Table

Create Application >

- In the Object Selection pane, click **Hardware**.
The Detail pane now shows details about Hardware table. For the Hardware table, review the column names and data types.
Click **Data**.

ORACLE APEX

App Builder

SQL Workshop

Team Development

App Gallery

Object Browser

Schema LOWCODE

Tables

DEMO_PROJECTS

DEMO_PROJ_COMMENTS

DEMO_PROJ_CONSTRAINTS

DEMO_PROJ_MILESTONES

DEMO_PROJ_STATUS

DEMO_PROJ_TASKS

DEMO_PROJ_TASK_LINKS

DEMO_PROJ_TASK_TODOS

DEMO_PROJ_TEAM_MEMBERS

EBA_DEMO_IR_DEPT

EBA_DEMO_IR_EMP

EBA_DEMO_IR_PROJECTS

HARDWARE

HARDWARE

Data

Indexes

Model

Constraints

Grants

Statistics

UI Defaults

Triggers

Dependencies

SQL

REST

Sample Queries

Query

Count Rows

Insert Row

Data

EDIT	ID	SERIAL	CPU_TYPE	CPU_SPEED	PURCHASE_DATE	BRAND	MODEL	FORM_FACTOR	PURCHASE_PRICE	DEPARTMENT
	1	OC8765T	Pentium III	200	10/09/2004	Dell	Optiplex	D	2343	Training
	2	1U09CM2423	Pentium III	667	06/01/2005	Dell	Optiplex	D	3256	Training
	3	OKBU4	Pentium IV	1000	05/12/2001	Dell	Optiplex	L	4532	Marketing
	4	51214246AB	Pentium IV	450	06/14/2001	Dell	Optiplex	L	5401	Consulting
	5	HQMHC12	Pentium IV	1700	06/15/2002	Dell	Optiplex	D	6588	Public Relations
	6	W112FGHZ456	Pentium IV	800	01/12/2001	Dell	Optiplex	D	5330	Development
	7	2DLP1	Celeron	366	04/01/2005	Dell	Optiplex	D	1399	Consulting
	8	6ZQ11	Pentium III	200	10/24/2004	Dell	Optiplex	D	2378	Training
	9	59785246GH	Pentium IV	650	03/20/2000	Dell	Optiplex	L	5122	Development

Now you see a report of the data contained in the Hardware table.

8. You want to create a lookup table now. Perform the following steps:

a) Click **Table**. Click **Create Lookup Table**.

HARDWARE					+ v	
Table	Data	Indexes	Model	Constraints	Grants	Statistics
					UI Defaults	Triggers
					Dependencies	SQL
					REST	Sample Queries
Add Column Modify Column Rename Column Drop Column Rename Copy Drop Truncate Create Lookup Table Create App						
Column Name	Data Type	Nullable	Default		Primary Key	
ID	NUMBER	No	"LOWCODE"."ISEQ\$\$_92310465".nextval		1	
SERIAL	VARCHAR2(50)	Yes	-		-	
CPU_TYPE	VARCHAR2(50)	Yes	-		-	
CPU_SPEED	NUMBER	Yes	-		-	
PURCHASE_DATE	DATE	Yes	-		-	
BRAND	VARCHAR2(50)	Yes	-		-	
MODEL	VARCHAR2(50)	Yes	-		-	
FORM_FACTOR	VARCHAR2(1)	Yes	-		-	
PURCHASE_PRICE	NUMBER	Yes	-		-	
DEPARTMENT	VARCHAR2(50)	Yes	-		-	

b) Select **DEPARTMENT** for Column. Click **Next**.

HARDWARE					+ v			
Create Lookup Table								
Select the column you would like to create a lookup table for. The selected column will become a foreign key to the lookup table.								
Schema:	LOWCODE ?							
Table Name:	HARDWARE ?							
Show:	<input type="radio"/> All Column Types ? <input checked="" type="radio"/> VARCHAR Column Types							
* Column:	<div> <input type="radio"/> SERIAL - varchar2 ? <input type="radio"/> CPU_TYPE - varchar2 <input type="radio"/> BRAND - varchar2 <input type="radio"/> MODEL - varchar2 <input type="radio"/> FORM_FACTOR - varchar2 <input checked="" type="radio"/> DEPARTMENT - varchar2 </div>							
<div> <div>Cancel</div> <div>Next ></div> </div>								

c) Accept the defaults on this page and click **Next**.

+ ▾

HARDWARE

Create Lookup Table

Specify the new table you wish to create. This new table will store a normalized list of values (that is, a unique list of values in the selected column).

Schema: **LOWCODE** ?

Table Name: **HARDWARE** ?

Column to Normalize: **DEPARTMENT** ?

New Table Name: ?

New Sequence: ?

☐ Preserve Case

Next >

d) Click Create **Lookup Table**.

+ ▾

HARDWARE

Create Lookup Table

Confirming this request, creates a new table and adjusts your current table structure.

Schema: **LOWCODE** ?

Lookup table: **DEPARTMENT_LOOKUP** ?

Table: **HARDWARE** ?

Lookup table primary key: **DEPARTMENT_ID** ?

Lookup based on Column: **DEPARTMENT** ?

Lookup table sequence: **DEPARTMENT_LOOKUP_SEQ** ?

Create Lookup Table

SQL

e) Review the table definition of **DEPARTMENT_LOOKUP** table.

+ ▾

DEPARTMENT_LOOKUP

Table	Data	Indexes	Model	Constraints	Grants	Statistics	UI Defaults	Triggers	Dependencies	SQL	REST	Sample Queries
<div style="display: flex; justify-content: space-between; padding: 5px;"> Add Column Modify Column Rename Column Drop Column Rename Copy Drop Truncate Create Lookup Table Create App </div>												
Column Name	Data Type	Nullable	Default	Primary Key								
DEPARTMENT_ID	NUMBER	No	-	1								
DEPARTMENT	VARCHAR2(4000)	No	-	-								

[Download](#) | [Print](#)

- f) In the Object Selection pane, click **HARDWARE**.
Review the table details. Notice that the DEPARTMENT column has been extracted from the HARDWARE table and is now available in the DEPARTMENT_LOOKUP table.

Object Browser

Schema: LOWCODE

Tables

DEMO_PROJECTS
DEMO_PROJ_COMMENTS
DEMO_PROJ_CONSTRAINTS
DEMO_PROJ_MILESTONES
DEMO_PROJ_STATUS
DEMO_PROJ_TASKS
DEMO_PROJ_TASK_LINKS
DEMO_PROJ_TASK_TODOS
DEMO_PROJ_TEAM_MEMBERS
DEPARTMENT_LOOKUP
EBA_DEMO_IR_DEPT
EBA_DEMO_IR_EMP
EBA_DEMO_IR_PROJECTS
HARDWARE

HARDWARE

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

Add Column Modify Column Rename Column Drop Column Rename Copy Drop Truncate Create Lookup Table Create App

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER	No	"LOWCODE"."ISEQ\$\$_92310465".nextval	1
SERIAL	VARCHAR2(50)	Yes	-	-
CPU_TYPE	VARCHAR2(50)	Yes	-	-
CPU_SPEED	NUMBER	Yes	-	-
PURCHASE_DATE	DATE	Yes	-	-
BRAND	VARCHAR2(50)	Yes	-	-
MODEL	VARCHAR2(50)	Yes	-	-
FORM_FACTOR	VARCHAR2(1)	Yes	-	-
PURCHASE_PRICE	NUMBER	Yes	-	-
DEPARTMENT_ID	NUMBER	Yes	-	-

Download | Print



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

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

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

Copyright © 2020, Oracle and/or its affiliates. All rights reserved. Licensed under the Creative Commons Attribution 4.0 International License as shown at <https://creativecommons.org/licenses/by/4.0/legalcode>.

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

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

