

+ Introduction

+ Getting Started

- Lab 1: Defining New Data Structures

- Introduction
- Task 1: Open Quick SQL
- Task 2: Enter Shorthand for Tables
- Task 3: Improve the Shorthand
- Task 4: Enter a Child Table
- Task 5: Update the Settings
- Task 6: Copy in the Complete Shorthand
- Task 7: Implement the Script
- Summary
- Acknowledgments

+ Lab 2: Creating the Application

+ Lab 3: Regenerating the Application

+ Lab 4: Improving Tasks

+ Lab 5: Improving To Dos

+ Lab 6: Improving Links

+ Need Help?

Define New Data Structures Using Quick SQL

Introduction

Memorizing the SQL to create and maintain database objects can be difficult. However, there are tools that can generate the code for you based on a shorthand syntax or a graphical user interface. In this lab, you will learn how to create tables and views by defining data structures in Quick SQL.

Estimated Time: 15 minutes

Problem Definition

Within most organizations, the use of spreadsheets is prolific. Spreadsheets are excellent for doing simple data review and tracking personal data, however, they are very ill suited to gathering data amongst team members for example.

A very common use case is that a manager needs to track the different projects his team is working on. The manager has built a spreadsheet with the critical data elements. However, now that the team is growing, trying to manage the spreadsheet updates is becoming unmanageable, and it is near impossible to get a single source of truth.

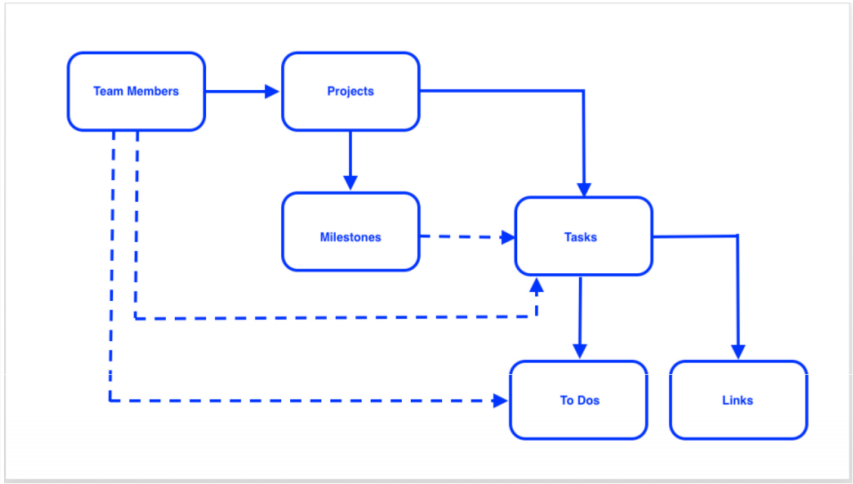
Below is an excerpt from the spreadsheet the manager has been sending out to the team:

	A	B	C	D	E	F	G	H
1	PROJECT	TASK_NAME	START_DATE	END_DATE	STATUS	ASSIGNED_TO	COST	BUDGET
2	ACME Web Configuration	Identify server requirements	5/17/19	5/18/19	Closed	John Watson	100	200
3	Maintain Support Systems	HR software upgrades	5/17/19	7/14/19	On-Hold	Pam King	8000	7000
4	Maintain Support Systems	Apply Billing System updates	5/17/19	7/18/19	On-Hold	Russ Sanders	9500	7000
5	ACME Web Configuration	Determine Web listener configuration(s)	5/18/19	5/18/19	Closed	James Cassidy	100	100
6	ACME Web Configuration	Specify security authentication scheme(s)	5/19/19	5/21/19	Closed	John Watson	200	300
7	ACME Web Configuration	Select servers for Development, Test, Production	5/19/19	5/24/19	Closed	James Cassidy	200	600
8	Email Integration	Complete plan	5/24/19	7/1/19	Closed	Mark Nile	3000	1500
9	ACME Web Configuration	Configure Workspace provisioning	5/26/19	5/26/19	Closed	John Watson	200	100
10	ACME Web Configuration	Create pilot workspace	5/26/19	5/26/19	Closed	John Watson	100	100
11	ACME Web Configuration	Run installation	5/27/19	5/27/19	Closed	James Cassidy	100	100
12	Bug Tracker	Implement bug tracking software	5/31/19	5/31/19	Closed	Myra Sutcliff	100	100
13	Bug Tracker	Review automated testing tools	6/1/19	7/2/19	On-Hold	Myra Sutcliff	2750	1500
14	Train Developers	Publish development standards	6/13/19	8/21/19	On-Hold	John Watson	1000	2000
15	Train Developers	Publish links to self-study courses	6/16/19	6/16/19	Closed	John Watson	100	100
16	Train Developers	Create training workspace	6/16/19	6/23/19	Closed	James Cassidy	500	700
17	Load Packaged Applications	Identify point solutions required	6/18/19	6/20/19	Closed	John Watson	200	300
18	Software Projects Tracking	Conduct project kickoff meeting	6/20/19	6/20/19	Closed	Pam King	100	100
19	Client Server Conversion	Identify pilot client server applications	6/20/19	6/21/19	Closed	Scott Spencer	200	200
20	Public Website	Determine host server	6/21/19	6/22/19	Closed	Tiger Scott	200	200
21	Load Packaged Applications	Install in development	6/22/19	6/22/19	Closed	John Watson	100	100
22	Public Website	Check software licenses	6/22/19	6/22/19	Closed	Tom Suess	100	100
23	Client Server Conversion	Migrate pilot client server to web	6/22/19	7/7/19	Closed	Scott Spencer	400	500
24	Load Packaged Applications	Customize solutions	6/23/19	9/18/19	Open	John Watson	1500	4000
25	Software Projects Tracking	Enter base data (Projects, Milestones, etc.)	6/25/19	6/26/19	Closed	Tom Suess	200	200
26	Migrate Desktop Application	Identify pilot applications	6/25/19	7/2/19	Closed	Mark Nile	300	500
27	Public Website	Purchase additional software licenses, if needed	6/25/19	7/20/19	On-Hold	Al Bines	300	1000
28	Email Integration	Check software licenses	6/27/19	6/30/19	Closed	Mark Nile	200	200
29	Software Projects Tracking	Load current tasks and enhancements	6/27/19	7/3/19	Closed	Tom Suess	400	500

Napkin Design - Improved data model for Projects

Rather than just creating a single Project Tasks table based on the spreadsheet, you can define a collection of data structures to better model the relationships and collect additional information that would be valuable.

Below is a napkin design for collecting project information:



You should notice that the new model allows the assignment of Team Members to Projects, as well as optionally to Tasks, and To Dos. The concept of a Project having Milestones has been introduced, however, it is optional that Tasks are associated with a Milestone. Further, To Dos and Links have been added to Tasks.

Objectives

- Add tables and insert data into your Oracle APEX Workspace

What Do You Need?

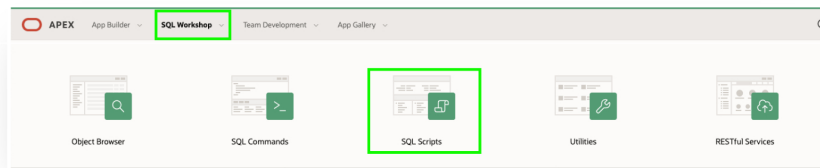
- An Oracle Cloud paid account, LiveLabs account or free trial.
- An APEX Workspace

Expand All Tasks

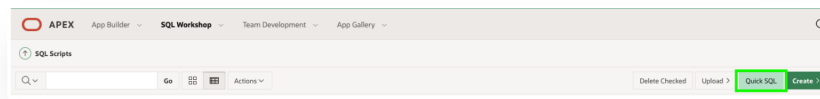
Task 1: Open Quick SQL

- Log into your workspace
- Click **SQL Workshop**

3. Click **SQL Scripts**



4. Click **Quick SQL**



+ Task 2: Enter Shorthand for Tables

+ Task 3: Improve the Shorthand

+ Task 4: Enter a Child Table

— Task 5: Update the Settings

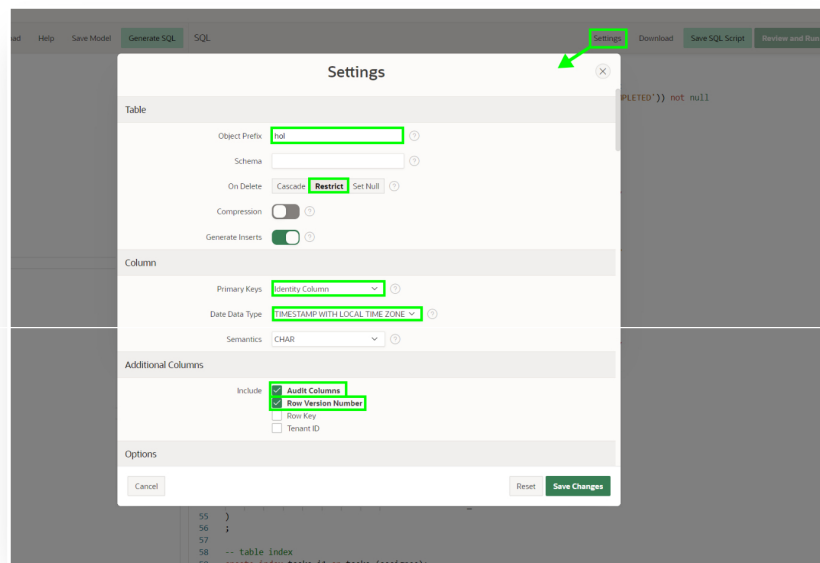
You can define numerous settings to further improve the generated SQL. You can click **Settings** within the toolbar under SQL (right pane), if you only plan to run the script once. As outlined in the **Help**, you can also define settings in Quick SQL (left pane). Settings defined in Quick SQL do not need to be reentered each time you return to Quick SQL to rerun a previously run script.

1. In the SQL (right pane) toolbar, click **Settings**

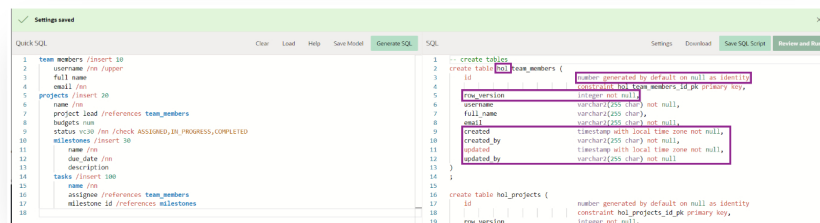
In the Settings dialog, enter the following:

- Object Prefix - enter **hol**
- On Delete - select **Restrict**
- Primary Keys - select **Identity Column**
- Date Data Type - select **TIMESTAMP WITH LOCAL TIME ZONE**
- Audit Columns - **Check**
- Row Version Number - **Check**

Click **Save Changes**



The generated SQL has changed markedly. The table names are updated, there are additional columns, and updated triggers.



+ Task 6: Copy in the Complete Shorthand

+ Task 7: Implement the Script

+ **Summary**

