Left and Right Join in SQL

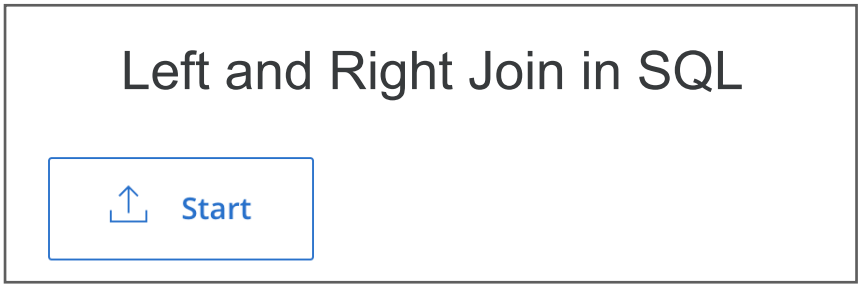
Welcome to this Lab activity

In this lab activity you will be exploring how to use the right and left join functions. For the purpose of this lab you will be working with the Terminal panel inside Visual Studio Code.

Start the Lab environment application

It is simple to launch a lab exercise. You only need to click on the button “Start” below the activity title to enter a lab environment.

Let’s explore this lab activity. Go ahead and click on the “Start” button!



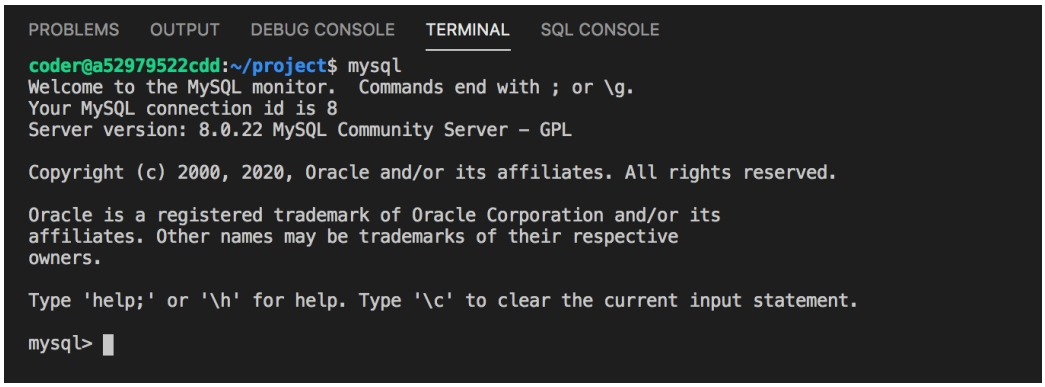
Task 1: Accessing the MySQL interactive shell

The folder structure has already been partially constructed for you and organised into different topics. For the purpose of this lab, you are not required to make any changes to the folder structure. You can see a folder called “topic9” inside this lab environment; it is only there as a reference for you and you are not required to add any content to it. Let’s get started!

In order to access your mysql interactive shell use the Visual Studio Code Terminal and run the following command:

- **mysql**: type this command and press *Enter*. This command will log you into mysql shell as the root(default) user.

If you have successfully followed all the above steps you should now be logged in inside mysql and see the following result on the Terminal:



Task 2: Create a new database

- First see what databases you already have linked to your virtual server:

SHOW DATABASES;

- You will already have a couple, including the “myBookshop” database you created previously.
- To create a new database, use:

CREATE DATABASE myfirstArt;

In the above, “myfirstArt" is the name of the new database. You can call your database what you like, it will contain tables about artworks and museums.

- To check if your new database was successfully created, use:

SHOW DATABASES;

Task 3: Create tables

- Switch to the new database you created:

USE <DATABASE_NAME>;

- Replace <DATABASE_NAME> with the name of your database (“myfirstArt”)
- Create a table named “artwork” based on the database design you see below, what kind of field types are the best to be used for this table?

I'll give you the answer for the first table:

CREATE TABLE Museum (id INT AUTO_INCREMENT, name VARCHAR(50), address VARCHAR(100) , website VARCHAR(100), PRIMARY KEY(id));

- Can you create another table named “Artwork” based on the database design you can see above?
- What sort of association would you use to connect museum and artwork tables?
- Please note that the “artwork” table includes a foreign key to the museum table. How can you create a table including foreign keys?

CREATE TABLE Artwork (id INT AUTO_INCREMENT, museum_id INT, name VARCHAR(50), artist VARCHAR(50), PRIMARY KEY (id), FOREIGN KEY(museum_id) REFERENCES Museum(id));

- See what tables were created:

SHOW TABLES;

- See what fields have been defined within a given table:

DESCRIBE TableName;

Replace TableName with the name of your tables.

Task 4: Input the dummy data

- To insert some dummy data into the database, you need to use INSERT INTO statement, If you are not sure how to do this refer to the “Insert dummy data in the database” lab instructions.
- Insert dummy data into the museum table, including the rows that you can see in the database design above.

INSERT INTO Museum(name, address, website) values ('National museum','Somewhere1','Some www1'),('Art museum','Somewhere2','Some www2'),('Modern museum','Somewhere3','Some www3');

- Insert dummy data into Artwork. I'll give you an example of how you can insert data in a table (Artwork) with a foreign key:

INSERT INTO Artwork (museum_id, name, artist) VALUES (NULL, 'My smile','JJ Arty');

INSERT INTO Artwork (museum_id, name, artist) VALUES ((SELECT id FROM Museum WHERE Museum.name='Art museum'), 'Colour madness','Junior Picasso');

INSERT INTO Artwork (museum_id, name, artist) VALUES ((SELECT id FROM Museum WHERE Museum.name='National museum'), 'Beautiful lake','Luke Real');

Task 5: Query the data in mySQL shell

Now that you have inserted the data, you are able to perform SQL queries on it.
You may use the wildcard (*) to return all the fields in a table:

SELECT * FROM TableName;

- See what data is in the “Artwork” and “Museum” tables.
- Performing a JOIN operation:

SELECT Artwork.name, Artwork.artist, Museum.name FROM Artwork JOIN Museum ON Artwork.museum_id = Museum.id;

- Performing a LEFT JOIN operation:

SELECT Artwork.name, Artwork.artist, Museum.name FROM Artwork LEFT JOIN Museum ON Artwork.museum_id = Museum.id;

- Performing a RIGHT JOIN operation:

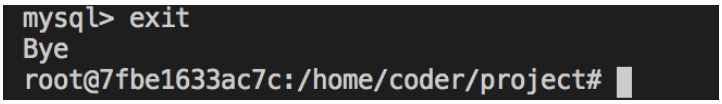
SELECT Artwork.name, Artwork.artist, Museum.name FROM Artwork RIGHT JOIN Museum ON Artwork.museum_id = Museum.id;

Task 6: Exit mysql shell

Exiting the mysql shell is very straight forward. In your Terminal panel type the following command:

- **exit**: type this command and press *Enter*. This command will log you out from your mysql virtual server.

If you have successfully exited the database you will get the following confirmation:



End of Section

Congratulations for completing this section. In the next lab activity you will be practicing more with database operations and explore how nested select can be used to query your databases.