



SQL



ASSESSMENT 2

Lisbeth Herrera

SQL- STRUCTURE QUERY LANGUAGE WORKSHOP

11/22/22

Table of Contents

SQL- STRUCTURE QUERY LANGUAGE	2
Install SQL MyWorkbench.....	5
Task 1	6
Task 2.....	11
Task 3.....	13
CREATING EER DIAGRAM.....	16

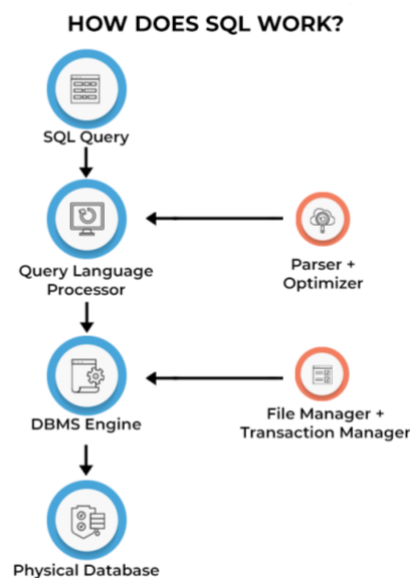
SQL- STRUCTURE QUERY LANGUAGE

SQL is a standard programming language; this programme is used to manage relational database and perform various operations on the data.

This programme is normally use for:

- Extract
- Organize
- Manage and manipulate data stored in relational databases.

SQL is referred to as a database language that can execute activities on databases that consist of tables made up of rows and columns.



Performing basic database operations

Data Definition Language (DDL): SQL uses commands to create and maintain databases such as CREATE, DROP, ALTER, TRUNCATE, and COMMENT.

Data Query Language (DQL): This refers to SQL commands used to retrieve data from databases. The commonly used SQL command here is SELECT.

Data Manipulation Language (DML): These refer to SQL commands used to manipulate data and perform critical operations on databases, such as INSERT, UPDATE, and DELETE.

Data Control Language (DCL): These refer to SQL control commands that grant users permission or access to perform certain operations. For example, REVOKE is the control command that revokes the access permission given to users.

SQL statements:

When you use SQL statements you should start with a SQL command and end with a semicolon (;), for example:

```
SELECT * customers ;
```

The meaning of this is that SELECT statement extracts all of the contents of a table called customers.

SQL statements are case-insensitive, meaning that they can be written using lowercase or uppercase. However, it is professional to write out SQL keywords commands in **CAPS** and table/column names in lowercase. Words in the statement can be treated as case-sensitive using quotes, so the following two statements produce identical results.

- SELECT * FROM customers;
- select * from CUSTOMERS;

These two statements are different:

- SELECT * FROM customers;
- SELECT * FROM "Customers";

SQL statements are terminated only by the semicolon, meaning that more complex statements can be rendered across multiple lines, like this one:

```
SELECT name, address, city  
FROM customers;
```

Shows that this will select the contents of the columns name, address and city in the table customers.

SQL statements can incorporate program flow controls, meaning that a statement can incorporate table and row selection and then operate on the data contained in those columns.

Like the following statement:

```
SELECT name, address, city  
FROM customers  
WHERE country=England;
```

Commonly used SQL commands with examples:

Apart from the statements used in our practice and in the assignment, I have looked from some similar queries, some commonly used SQL commands where I can show some examples of SQL statements using those commands, follow.

SQL SELECT example:

```
SELECT title,author, pub_date  
FROM catalog  
WHERE pub_date =2021;
```

The example above, the table name as Catalog will be selected and from there the information of the title, author and publication date columns will be extracted.

SQL CREATE, used to create a new SQL database in which tables and other database objects are stored as files.

```
CREATE DATABASE 'Human_Resources';
```

In the example above creates a new SQL database named Human_Resources

SQL DELETE. The DELETE command removes rows from a named table. This example shows how to delete employees from with a last name Smith

```
DELETE FROM Employees WHERE  
last_name='Smith';
```

SQL UPDATE. The UPDATE command, this command is used to make changes to rows or records in a specific table.

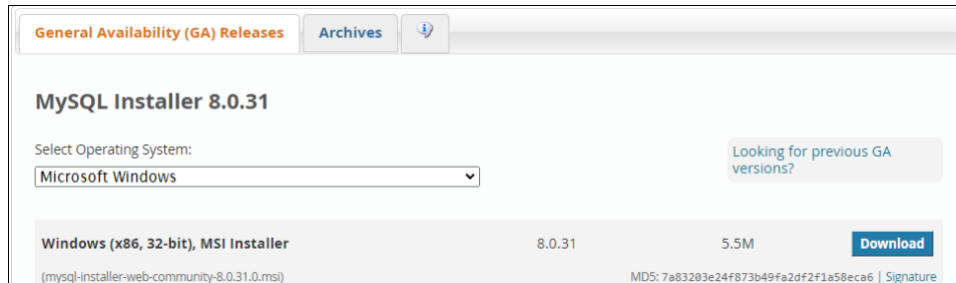
```
UPDATE Employees  
SET last_name = 'Smith',  
WHERE last_name = 'Smithee';
```

Install SQL MyWorkbench

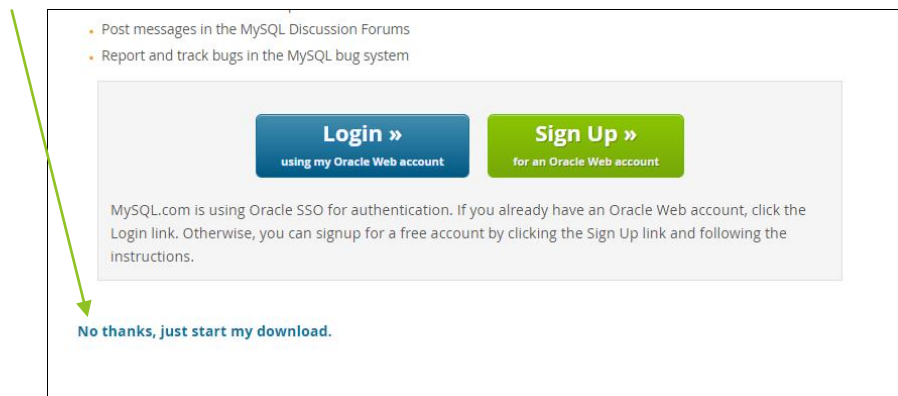
In this task we have downloaded the SQL workbench accordingly of the systems, there is one path for Windows systems and another for Macs.

My operating system is Windows system so I used the link below to start with the task:

- <https://dev.mysql.com/downloads/installer/>
- I have selected the 5.5M as this version use the least space



- Next step, I have selected

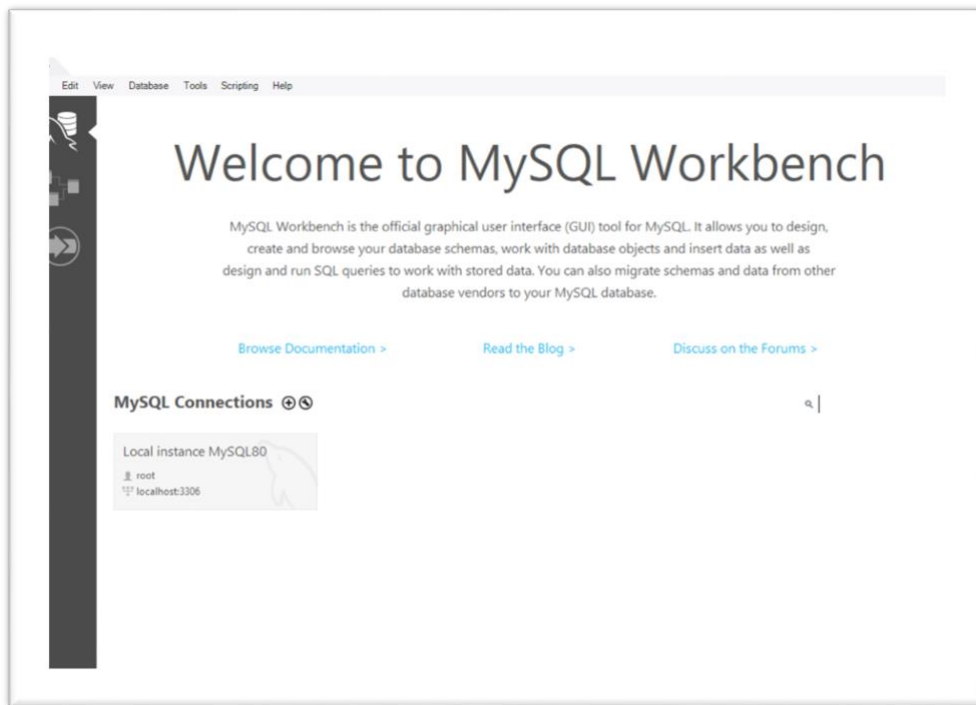


After this I followed the steps provided in a word document, start with choose a set up type: **Custom**

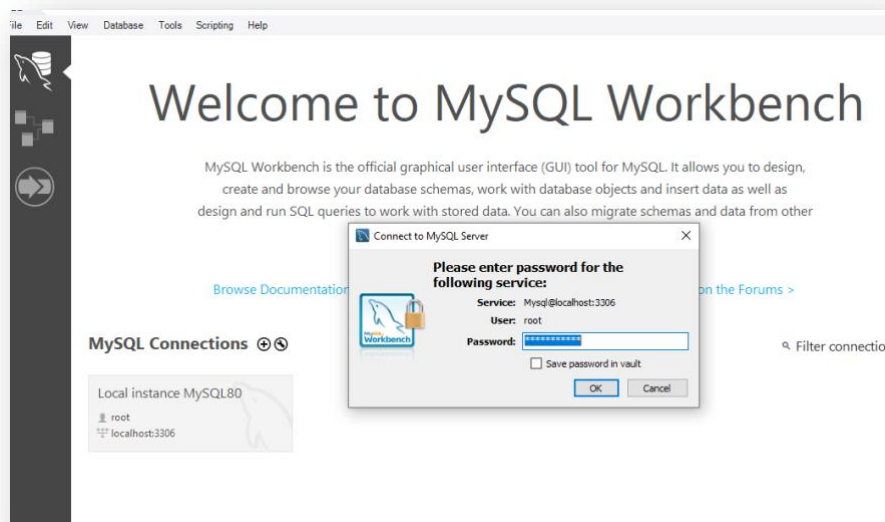
Following by selecting the My SQL SERVER 5.7.20 x64 option

Task 1

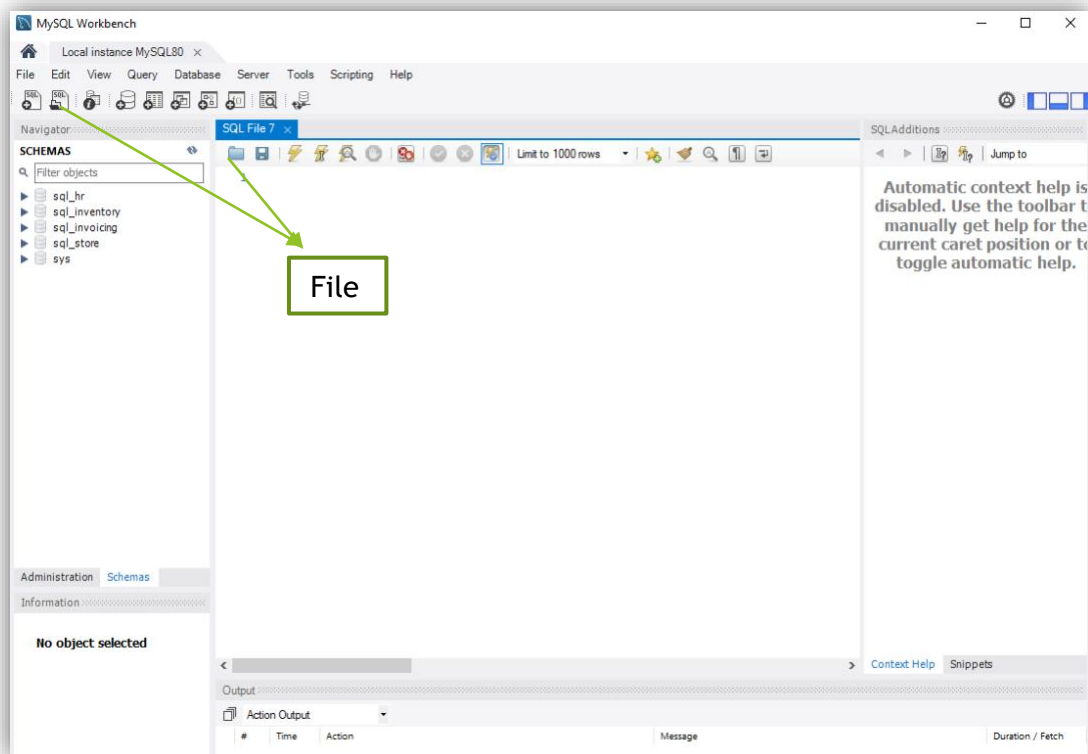
To start with the task 1, I have selected MySQL Workbench as the below picture.



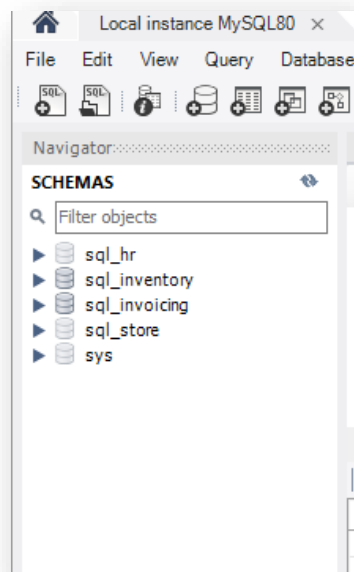
- Click in Local instance MySQL and added my password, as follows,

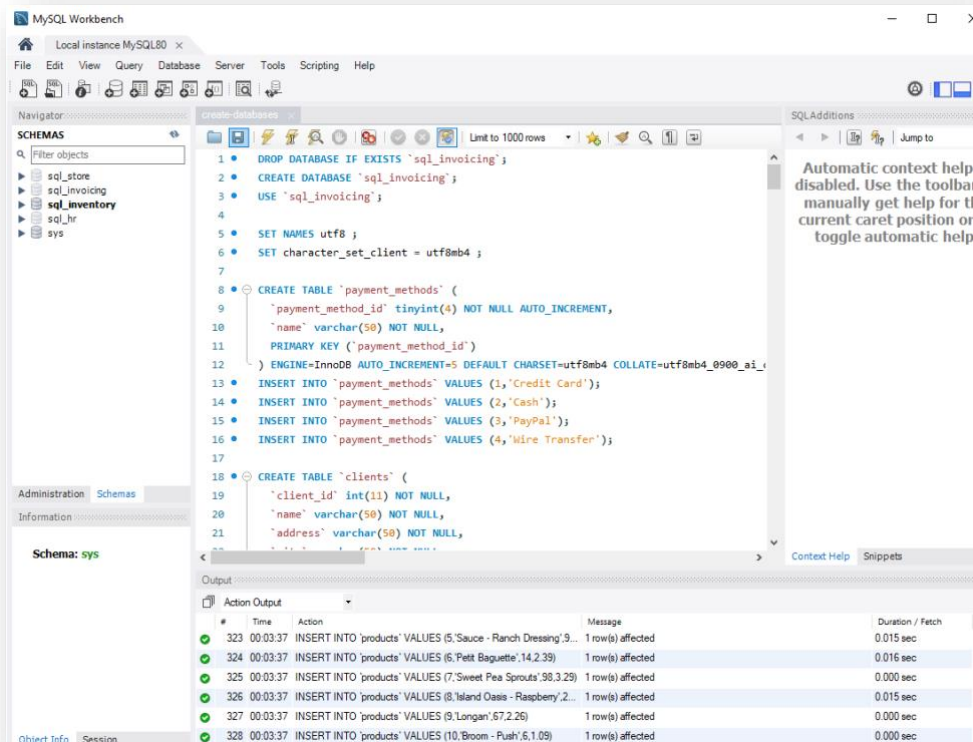


- We were asked to import data from our store, I selected the file that was given as a Class material and saved in my PC. Following I have uploaded the file from the icon file, this is also its possible to add from file.



- After I have executed the file, it showed different files like



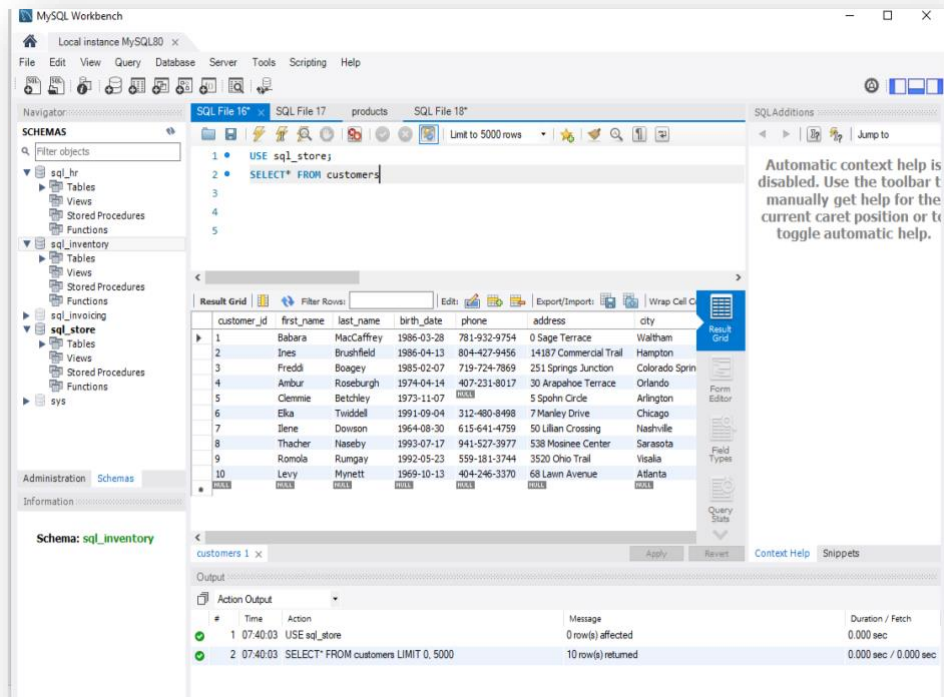


- In the picture above you can see the different commands already executed, at the beginning you have the DROP command which means if you have another table with the name SQL_invoicing, delete it.
- CREATE DATABASE: it says to CREATE a table with the data :sql_invoicing, when we used single quotations means SQL identified the start of the data and when does finish.
- To start the assignment, I have added a new query as shows below. Selects the only the customers from the sql_store database:

```

USE sql_store;
SELECT*
FROM costumers

```

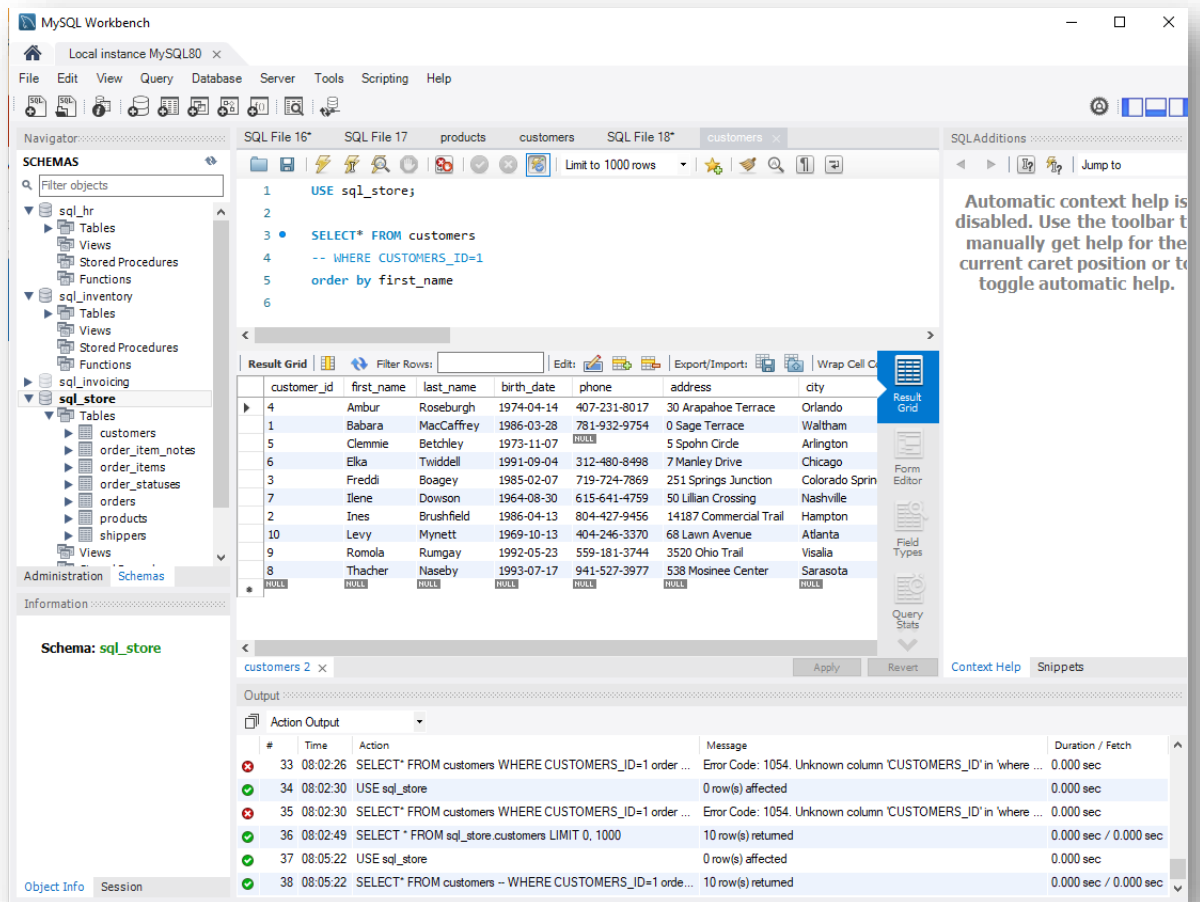


- After this I have added the following query where it did select only the customers from the sql_store database and order by first name in alphabetic order (I added a space between - and WHERE).
-

```

USE sql_store;
SELECT*
FROM costumers
--WHERE CUSTOMERS_ID=1
Order by first_name

```

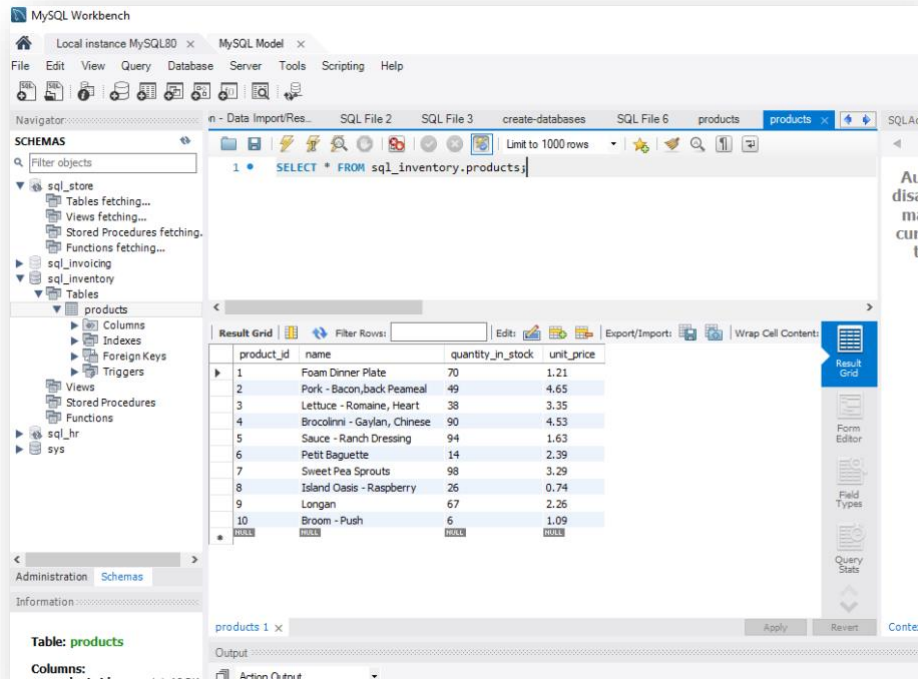


- After execute and created the new database and I have added the following query:

SELECT*FROM sql_inventory.products;

And press Execute

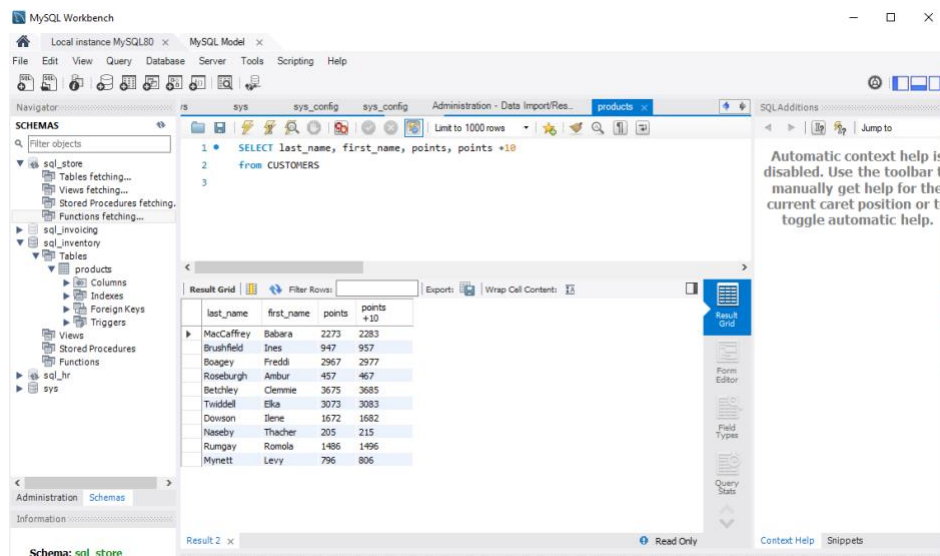
- Inserting this query, I could select products for the Table inventory.



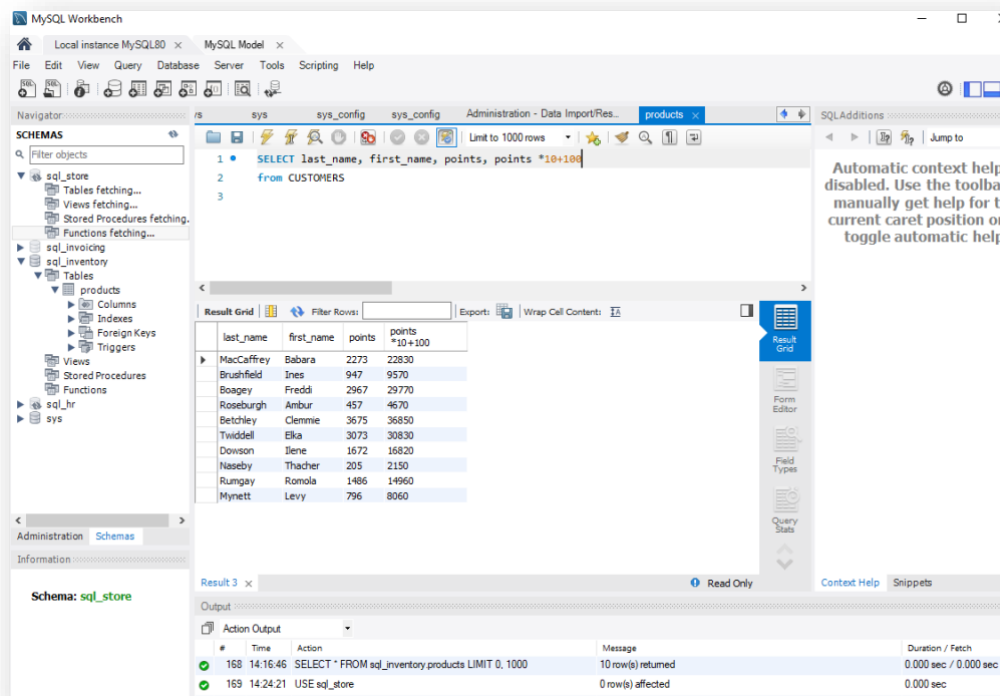
Task 2

- In this task I have created a new query and added the following statement, shows as result a column with new 10 points added.

SELECT last_name, first_name, points, points +10
FROM customers

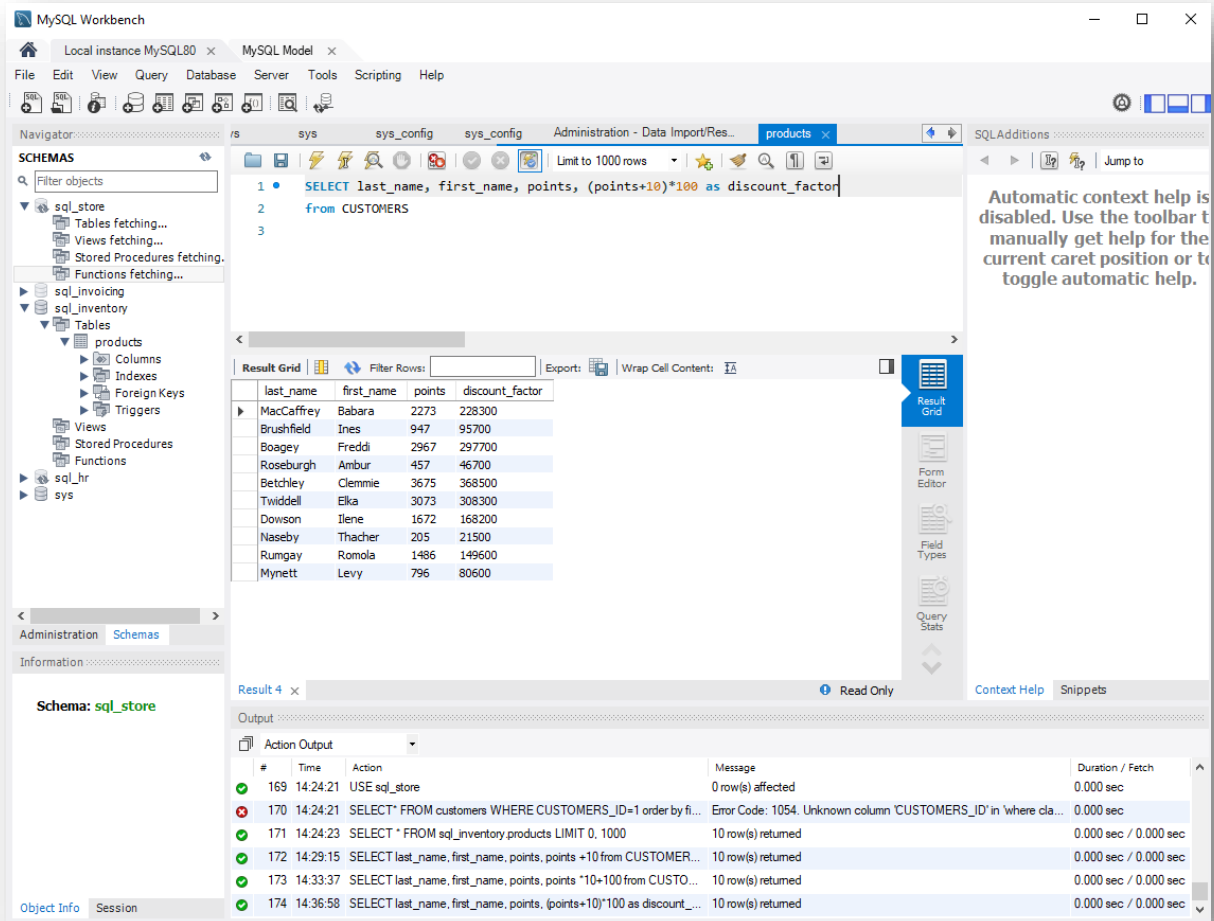


- I have added to the previous statement: ***10+100**, result is a new column with the addition of points results.



- Then for the next statement I have created a discount factor, showing header with discount factor:

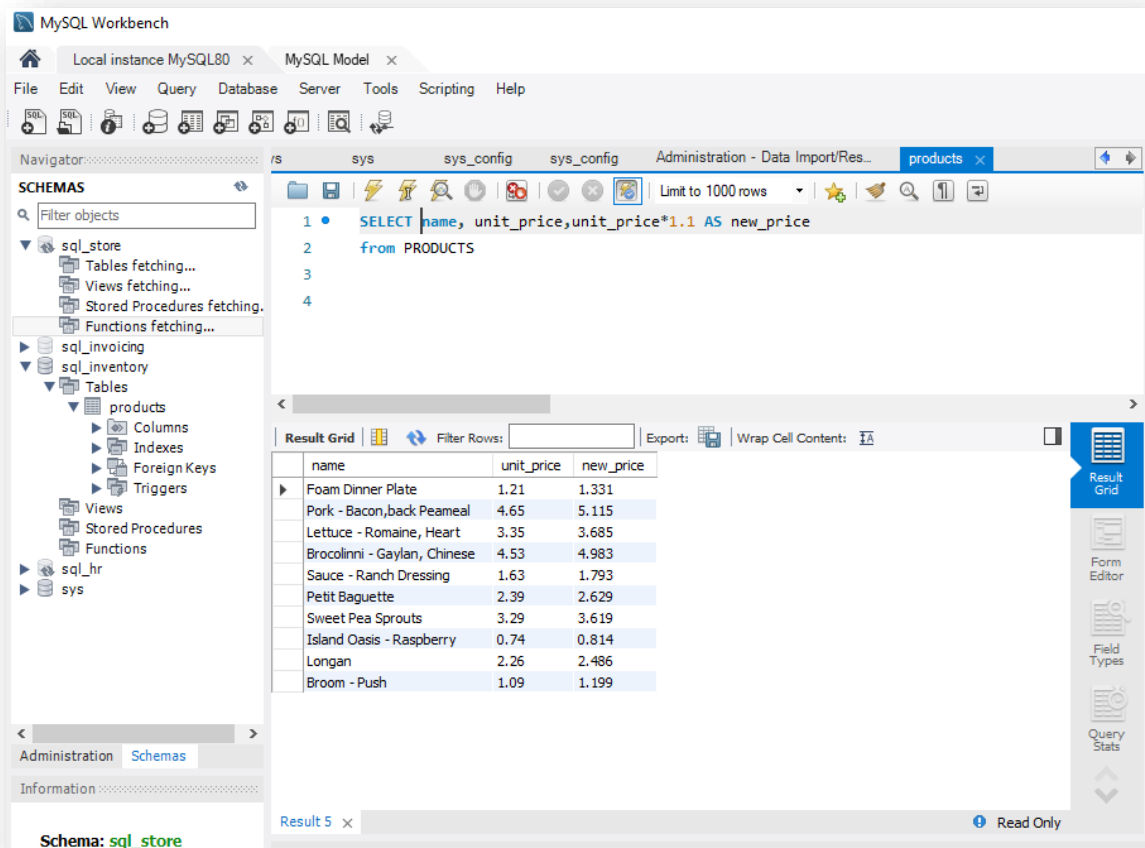
SELECT last_name, first_name, points, (points +10)*100 as discount factor
FROM customers



Task 3

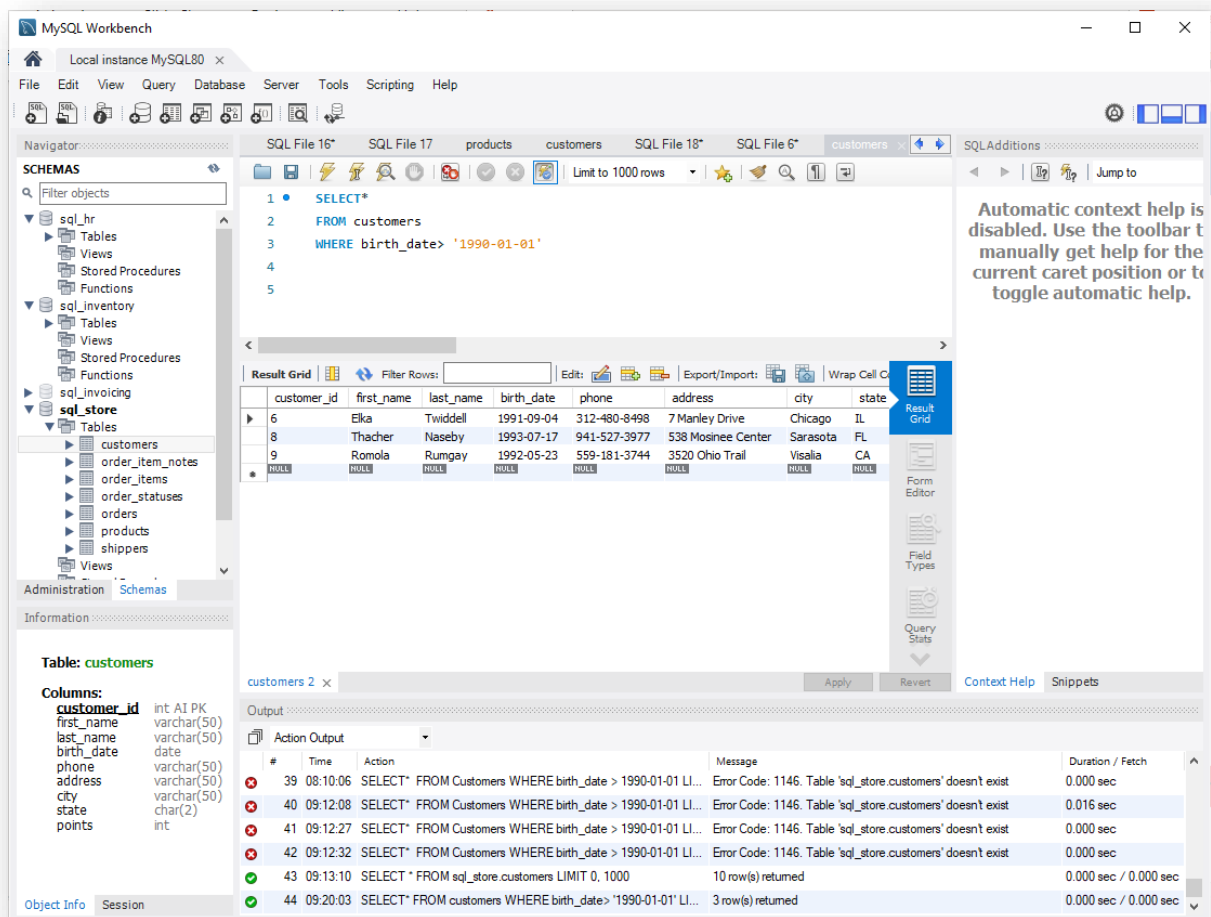
- For this Task we were ask to enter I wrote a SQL query to return all the products with the result set and make 3 columns, name, unit price, and new column base on unit price*1.1 Increasing the price for 10%, so I showed the original price and the new price

```
SELECT name, unit_price, unit_price*1.1 AS new_price
FROM products
```



- In this task I have created a new query to find all the customers with a birth date of >'1990-01-01', which is the follow, as a result shows only the birthdate of customers who are born on or after 1990-01-01:

```
SELECT *
FROM Customers
WHERE birth_date > '1990-01-01'
```



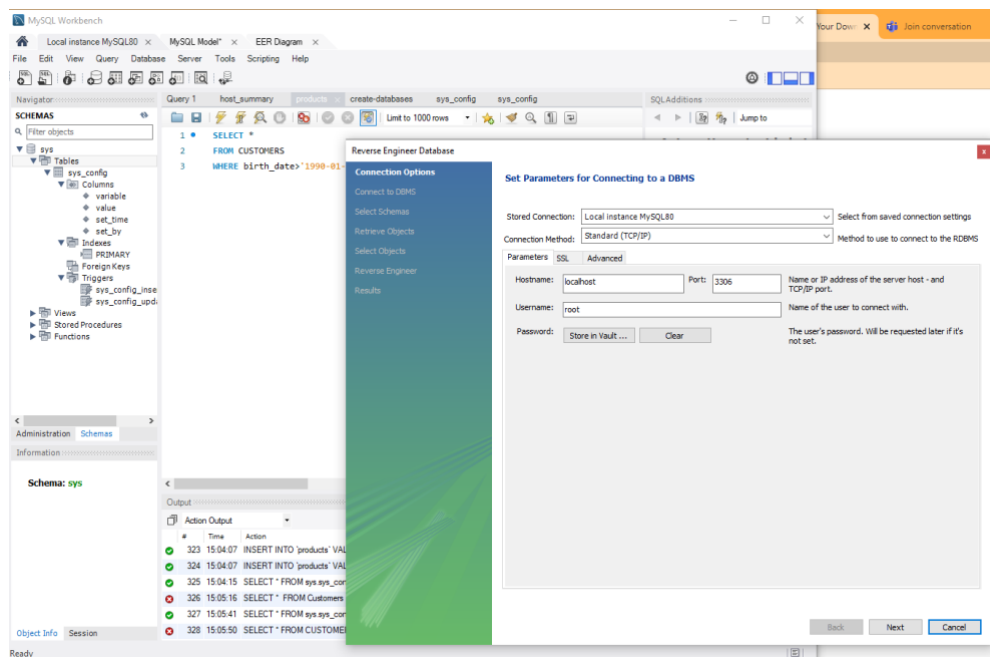
SUMMARIZE CODE USED IN ALL THESE TASKS:

In our assignment we have perform different keywords in order to get various operations on the databases. The following shows some of the examples of such keywords with their functional roles:

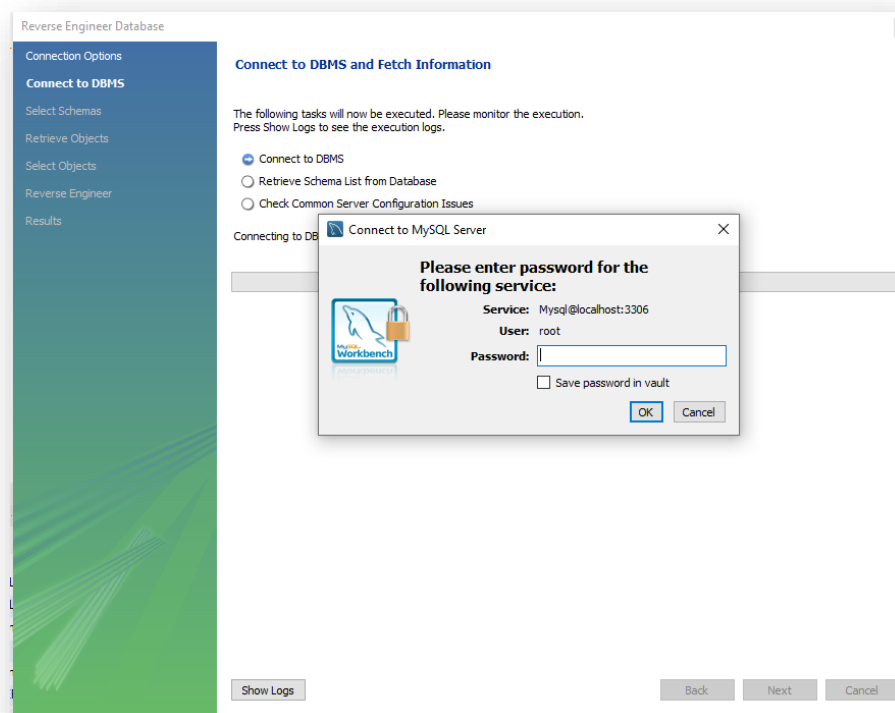
- **SELECT:** Selects data from database or table
- **FROM:** Indicates the table from which data needs to be fetched
- **SEMICOLON (;):** such important keyword as it helps to end a query
- **CREATE:** This keyword helps in creating a database structure or simply tables, views, and an index
- **INSERT:** It adds data to the rows of a table
- **WHERE:** It filters the data so that only relevant data matching certain conditions is fetched
- **UPDATE:** Updates existing rows in a table
- **DELETE:** It deletes the existing rows in a table

CREATING EER DIAGRAM

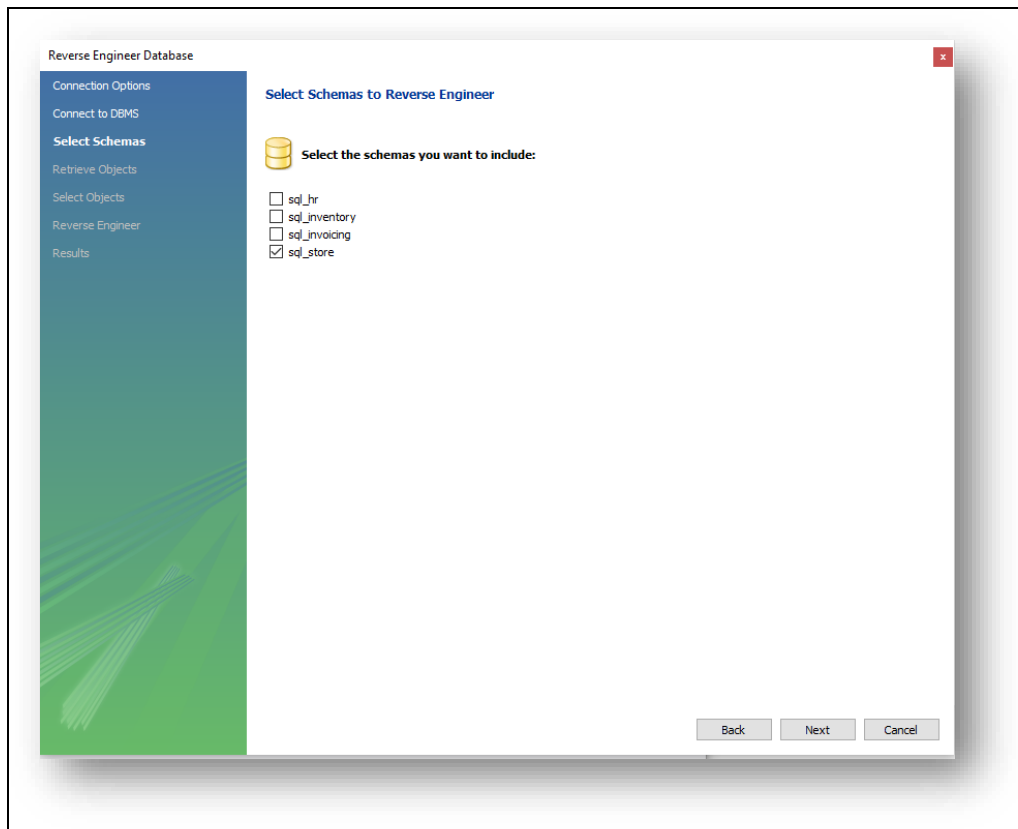
In order to create an EER DIAGRAM , I went to DATABASE and select Reverse Engineer Database,



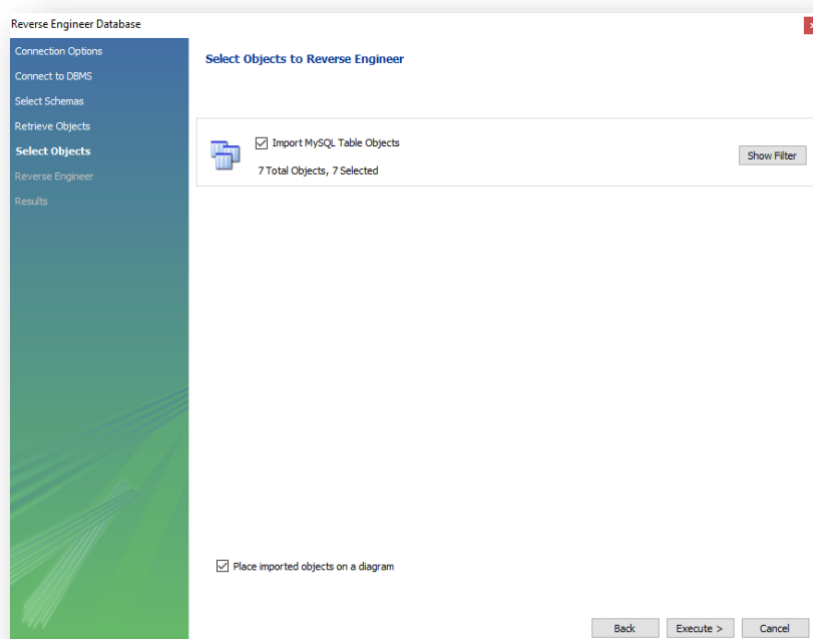
- Next step is enter my password in order to connect and show as below



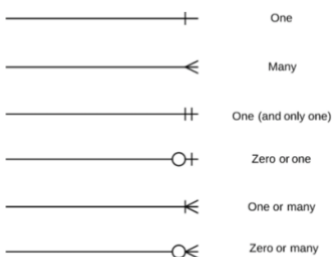
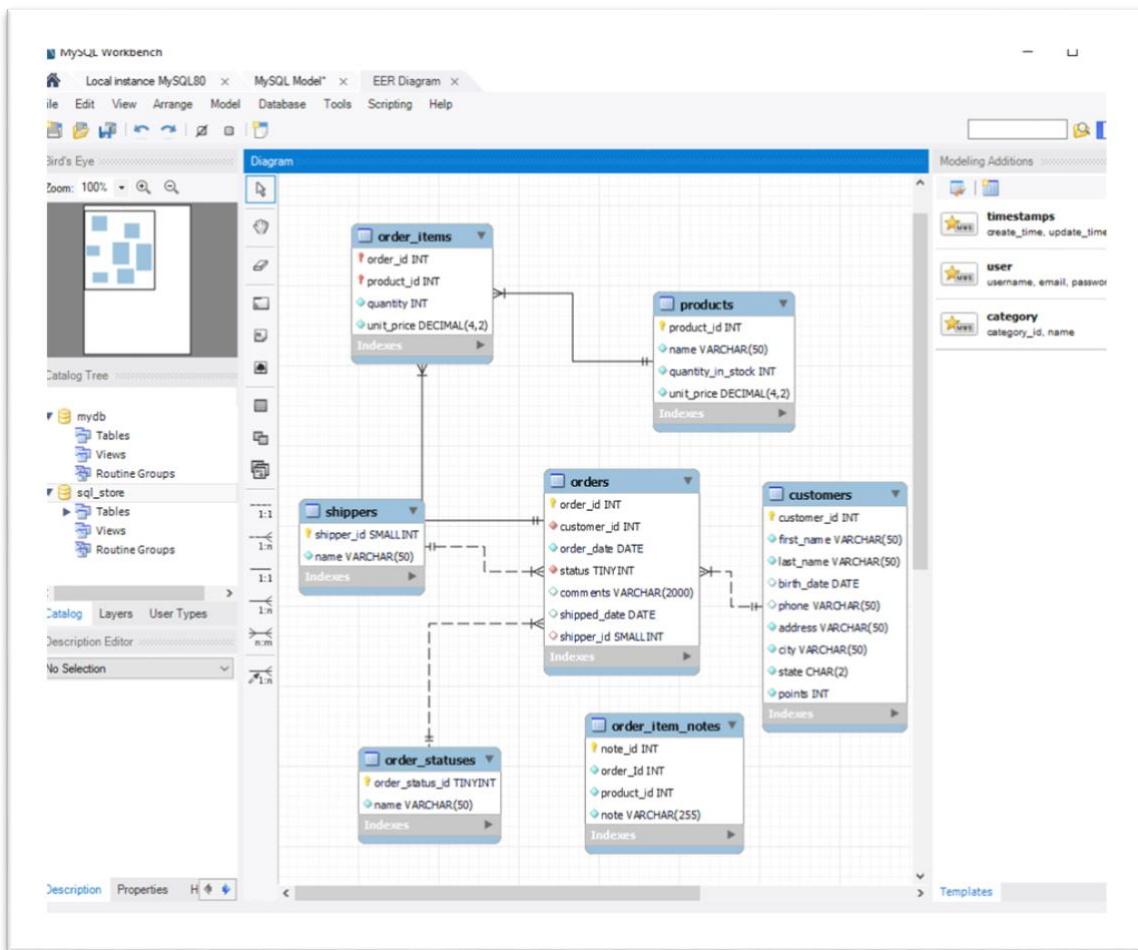
- Following I have selected **sqlstore**:



- After this press EXECUTE



- Lastly, I have learnt how to create a **EER Diagram**, this shows the relationship between tables, when you organise to show your data you can move around however it's important to know that this is an snowflake scheme.



Conclusion:

MySQL is an advanced feature, powerful and versatile relational database management system, in this assessment I had the opportunity to work with basics queries, managing the data store in a reliable and efficient way, I have also learnt about the difficulties when writing down queries and how careful you need to state your queries.

As a result I have created a EER diagram which shows an snowflake scheme, in this scheme shows different entities that represents a table and their different Cardinal relationships, we have worked with the data of sql_store which shows the relationships between Customers, orders_ID, products, shippers and orders status.