SQL Assignment

Contents

[Part 1 3](#_Toc164984790)

[Importing Data 3](#_Toc164984791)

[Query 1 4](#_Toc164984792)

[Query 2 5](#_Toc164984793)

[Task 1 6](#_Toc164984794)

[Task 2 7](#_Toc164984795)

[Task 3 8](#_Toc164984796)

[Task 4 8](#_Toc164984797)

[Task 5 9](#_Toc164984798)

[Task 6 9](#_Toc164984799)

[EER Diagram 10](#_Toc164984800)

[Part 2 12](#_Toc164984801)

[Importing 12](#_Toc164984802)

[Task 1 13](#_Toc164984803)

[Task 2 14](#_Toc164984804)

[Task 3 14](#_Toc164984805)

[Task 4 14](#_Toc164984806)

[Task 5 15](#_Toc164984807)

[Task 6 15](#_Toc164984808)

[Task 7 15](#_Toc164984809)

[Task 8 16](#_Toc164984810)

[Task 9 17](#_Toc164984811)

[EER Diagram 17](#_Toc164984812)

[Bonus 19](#_Toc164984813)

[Modifying Data 21](#_Toc164984814)

[Adding Record 21](#_Toc164984815)

[Modifying Record 21](#_Toc164984816)

[Delete Record 22](#_Toc164984817)

[Interview Questions 22](#_Toc164984818)

[Query 22](#_Toc164984819)

[SELECT Statement 22](#_Toc164984820)

[WHERE Clause 22](#_Toc164984821)

[Primary Key 23](#_Toc164984822)

[Database 23](#_Toc164984823)

[Relationships 23](#_Toc164984824)

[Normalization 23](#_Toc164984825)

[Modify Query 1 23](#_Toc164984826)

[Modify Query 2 24](#_Toc164984827)

[Modify Query 3 24](#_Toc164984828)

[Modify Query 4 24](#_Toc164984829)

[Modify Query 5 24](#_Toc164984830)

## Part 1

### Importing Data

I went to the server tab and select data import.

A screenshot of a computer

Description automatically generated

I selected import from self contained file and inserted the file location of the database. This inserts my schema into the MySQL server.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

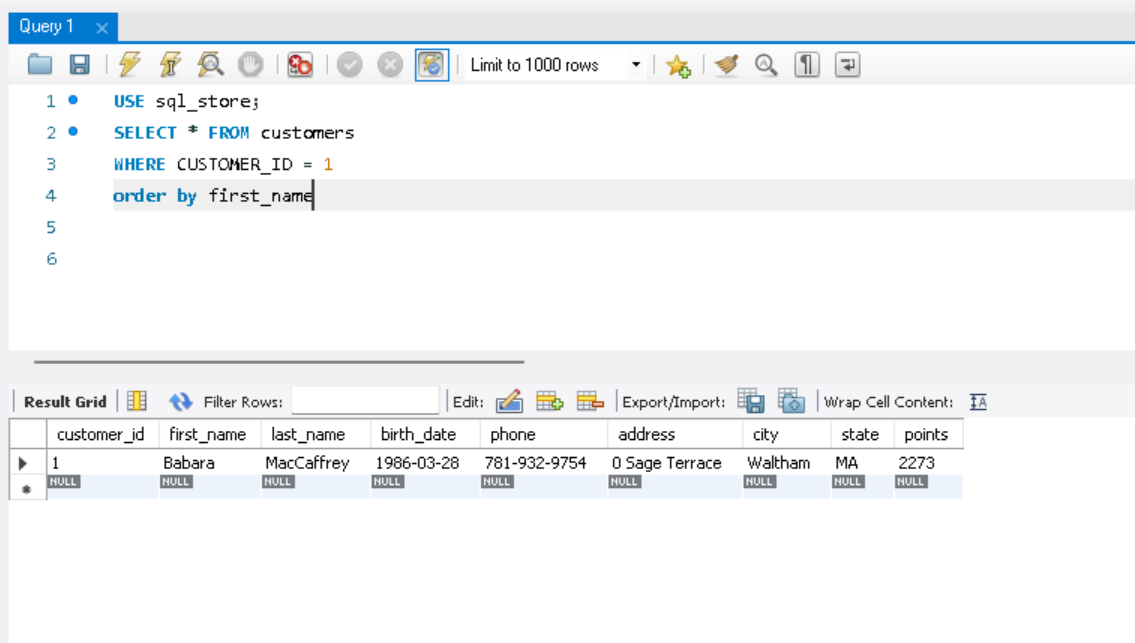
### Query 1

The USE statement specifies that I wanted to use the sql\_store database. The SELECT statement means what columns I wanted displayed and the asterisk means to select all of them. The FROM statement tells it which table in the database this information is in. In summary it chooses the sql\_store database and selects all information from the customers table.

A screenshot of a computer

Description automatically generated

This next query is similar, but it also adds WHERE which is a conditional on what will be displayed. In this case it shows only the record with the customer ID of 1. The order by statement will decide the order of the displayed records. In the case of the first name it will sort in alphabetical order.



### Query 2

The first selects the first name, last name and points columns from the customers table,

The second also created another column which is the points column but all data has + 10 added to it.

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

### Task 1

This query adds a points column that is multiplied by 10 and added to by 100

A screenshot of a computer

Description automatically generated

This query uses the AS statement to create a name for the new column called discount\_factor

A screenshot of a computer

Description automatically generated

### Task 2

This query shows the name of the products, the original unit price and the new price which is the original price increased by 10% and named new\_price.

A screenshot of a computer

Description automatically generated

### Task 3

This only selects the records from customers where the birth date is greater than 1990-01-01.

A screenshot of a computer

Description automatically generated

### Task 4

This shows the product with the highest quantity in stock. The LIMIT statement only shows the one record at the very top which will be the highest quantity product when sorted by qualitiy\_in\_stock.

A screenshot of a computer

Description automatically generated

### Task 5

This is similar to the previous query however it shows the highest priced product.

A screenshot of a computer

Description automatically generated

### Task 6

This takes the first and last name, address and birth date of the oldest customer in the sql\_store database.

A screenshot of a computer

Description automatically generated

### EER Diagram

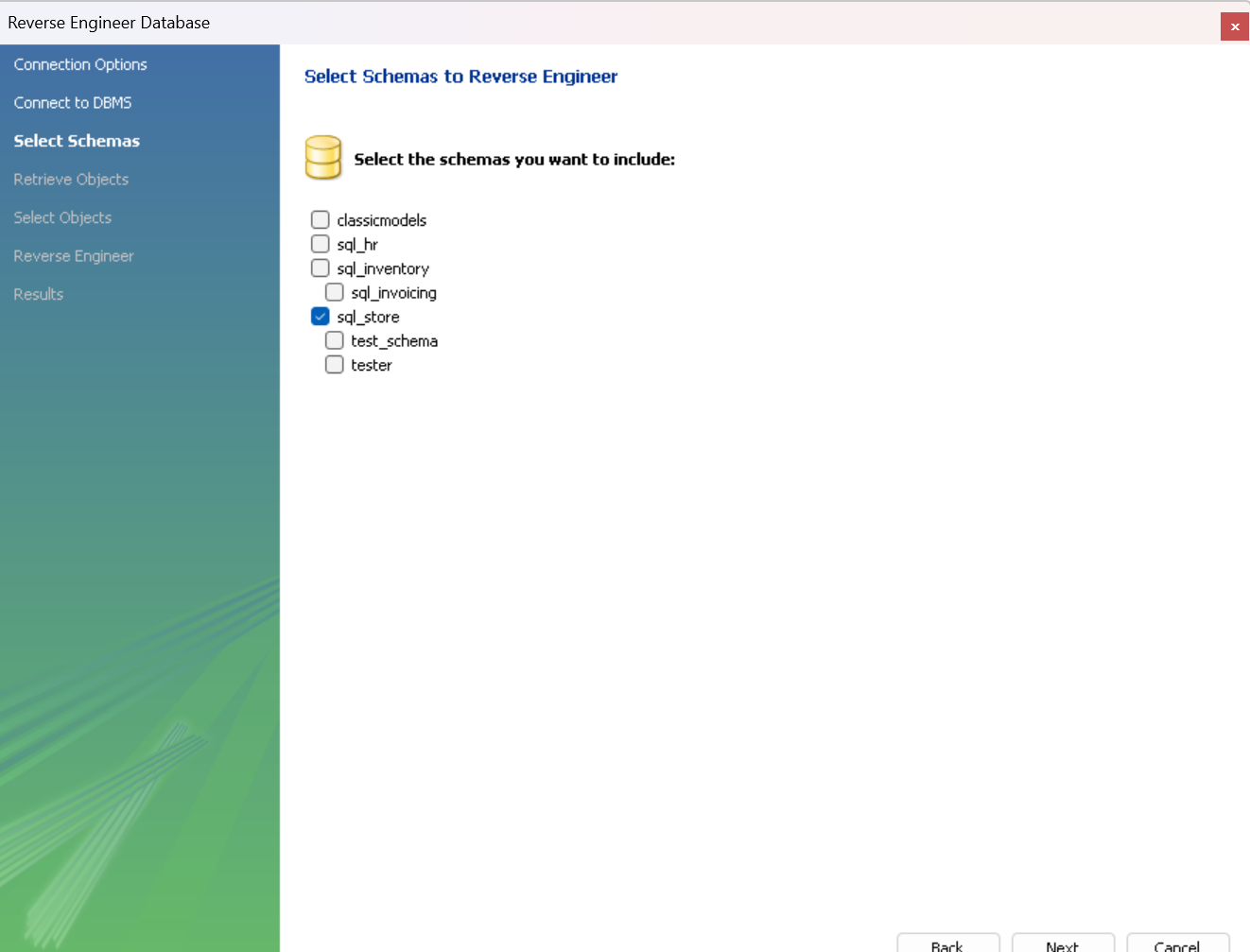
To create an enhanced entity-relationship diagram I went to the database tab and selected reverse engineer. Selected the sql\_store schema to reverse engineer and finished. A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

 A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The customer\_id column in the customers table is a primary key that has a one to many link to the customer\_id in the orders table. There are similar one to many relationships between the product\_id in the products and product \_id int the order\_items table, the order\_status\_id in the order\_statuses table and the status in the orders table as well as shippers\_id in the shippers table and shippers\_id in the orders table.

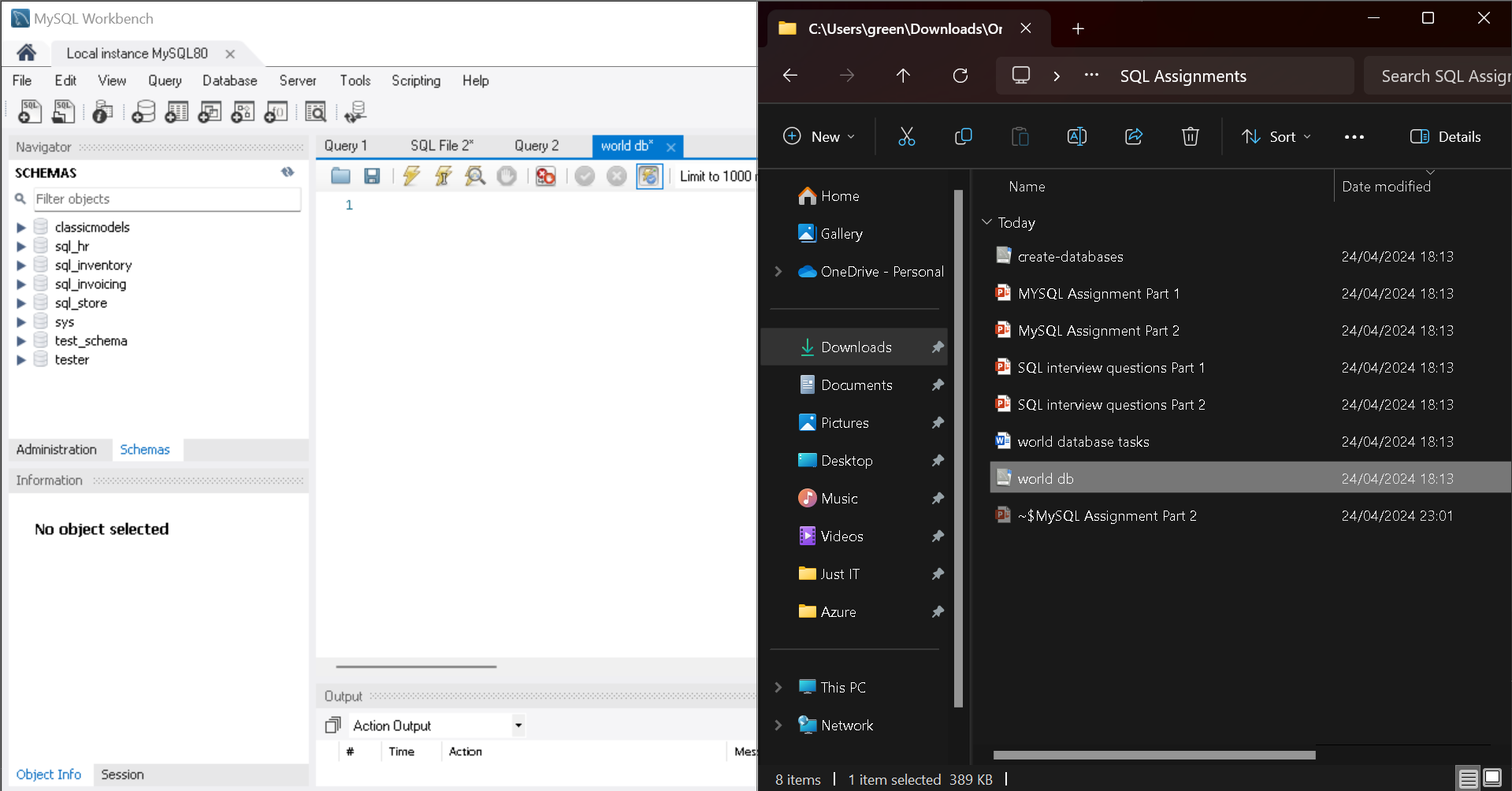
A screenshot of a computer

Description automatically generated

## Part 2

### Importing

To import, drag SQL text file into MySQL Workbench.



Run the query.

A screenshot of a computer

Description automatically generated

Refresh schema to see new database.

A screenshot of a computer

Description automatically generated

### Task 1

This shows the amount of cities in the database.

A screenshot of a computer

Description automatically generated

### Task 2

This shows the Life expectancy and population of Argentina.

A screenshot of a computer

Description automatically generated

### Task 3

This shows the country with the highest life expectancy.

A screenshot of a computer

Description automatically generated

### Task 4

This is 25 countries that begin with ‘F’.

A screenshot of a computer

Description automatically generated

### Task 5

This is the ID, name and population of the city limited to the first 10 results,

A screenshot of a computer

Description automatically generated

### Task 6

This is the name and population of cities where the population is over 2,000,000.

A screenshot of a computer

Description automatically generated

### Task 7

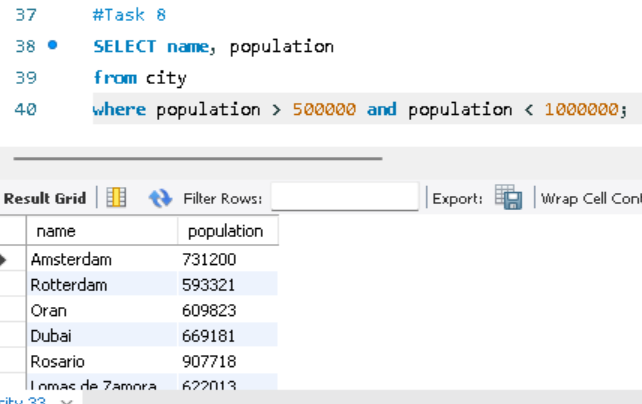
This is the name of cities that begin with ‘Be’.

A screenshot of a computer

Description automatically generated

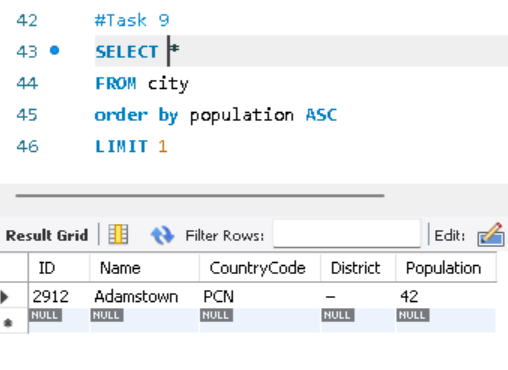
### Task 8

This is the name and population of cities between 500,000 and 1,000,000.

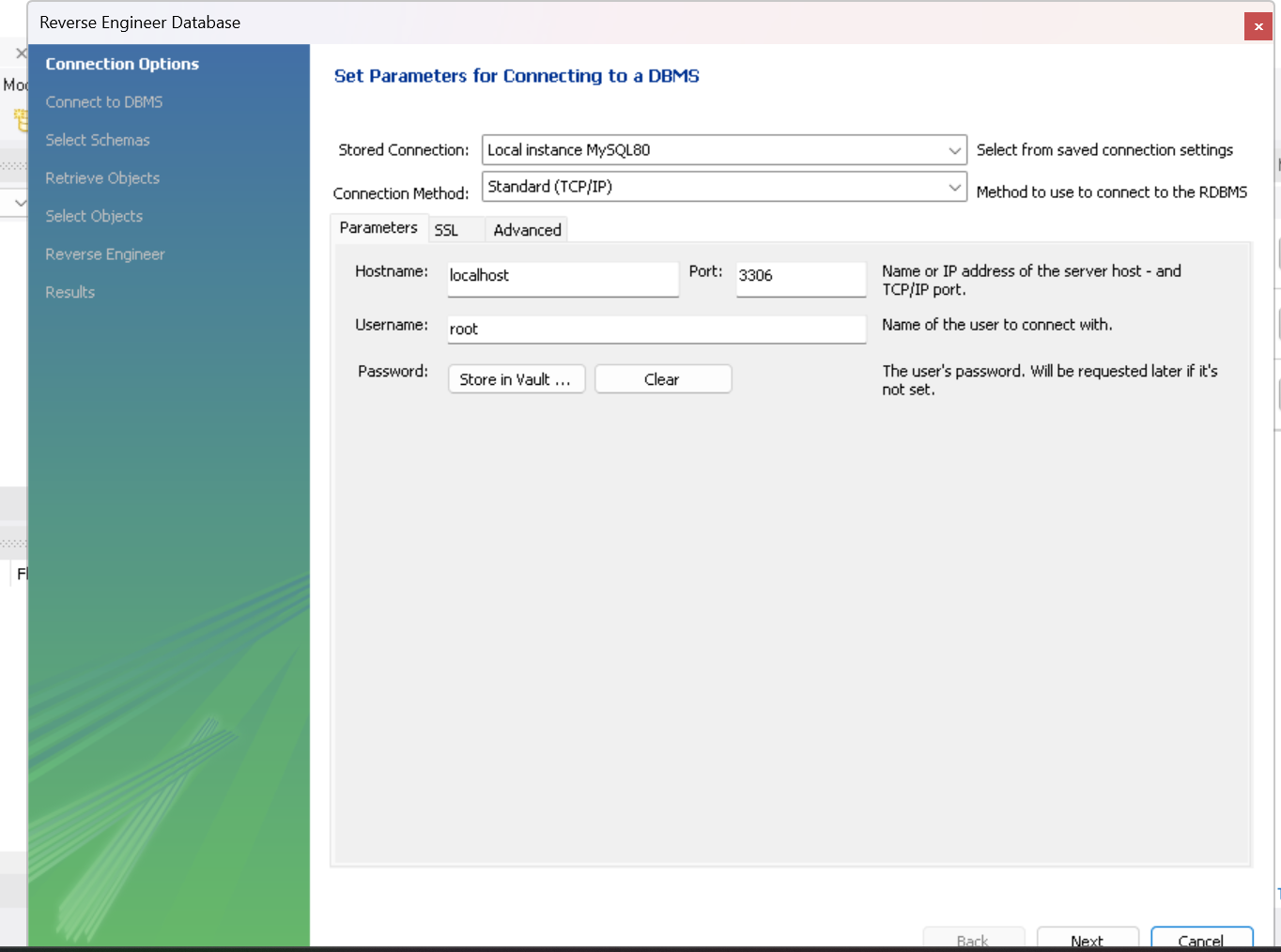


### Task 9

This shows the lowest population country



### EER Diagram

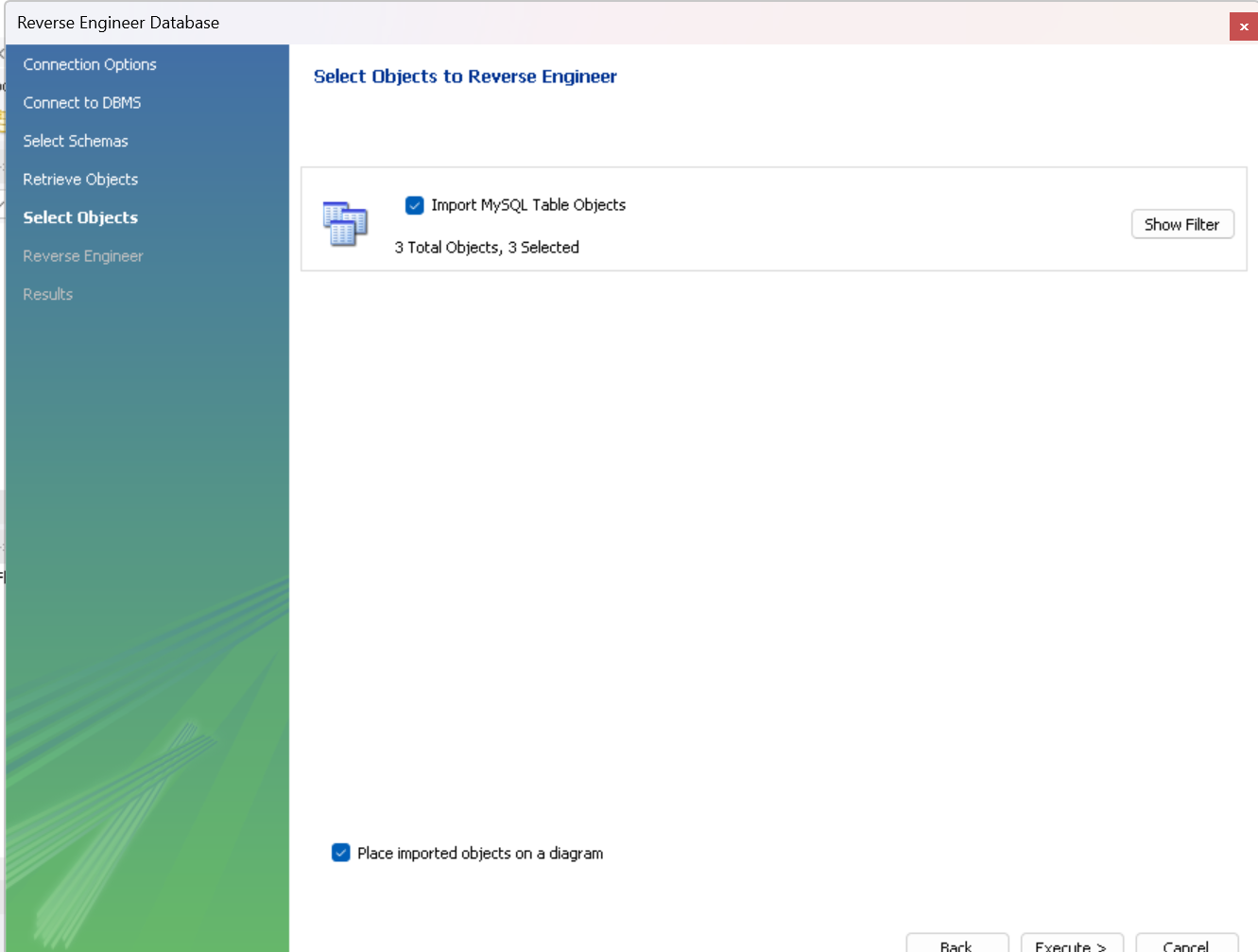
 A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated



Primary Key in County Table - Code

Primary Key in City Table - ID

Primary Key in countrylanguage Table - Language

Foreign Key in City Table - CountryCode

Foreign Key in countrylanguage Table - CountryCode

A screenshot of a computer

Description automatically generated

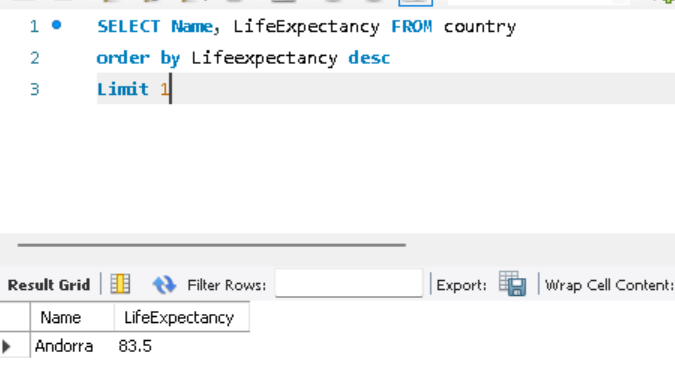
## Bonus

Create a SQL statement to find the capital of Spain (ESP).

A screenshot of a computer

Description automatically generated

Create a SQL statement to find the country with the highest life expectancy.



Create a SQL statement to find all cities from the Europe continent.

A screenshot of a computer

Description automatically generated

Create a SQL statement to find the most populated city in the city table.

A screenshot of a computer

Description automatically generated

## Modifying Data

### Adding Record

To add a new record, using insert into will add a new record to the declared table. You list each column you want to add data to before using the values statement to specify that the following data will be inserted into the table.A close-up of a number

Description automatically generated

A screenshot of a computer

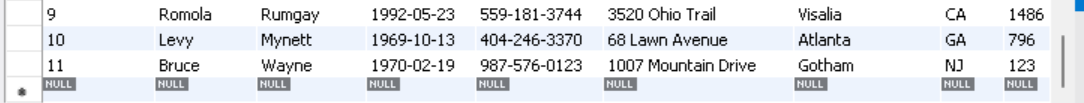
Description automatically generated

### Modifying Record

To modify the data in an existing record, the update statement is used. Then set is used to declare the data you want to add and in which column it will overwrite. The where statement is used as the conditional to target what records will be changed. I chose the customer\_id as it is a unique identifier of the record so it would only affect the one record.

A screenshot of a phone number

Description automatically generated



### Delete Record

The delete statement will delete the record from the customers table where the customer\_id is 11.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

## Interview Questions

### Query

A query is a request to retrieve or affect data in a database. It can be used to retrieve data, calculate data, combine data, add, modify, or delete data.

### SELECT Statement

The select statement retrieves data from columns from one or multiple tables.

### WHERE Clause

The where clause can filter the results of a select statement based on the conditions specified. For example, ‘WHERE first\_name = Charlie’ would only show records where the first name is Charlie.

### Primary Key

It is a unique identifier for every record in a table. For example an employee \_id would be used as a unique number for every individual employee in a company.

### Database

A database is a collection of stored and organized data. A database is used for the retrieval, manipulation, and management of data. A database can contain one or more table of records with columns which define each set of data. Each table can have relationships to each other so a change in one can effect the other. The structure of the database, the tables, columns, and relationships between tables, is called schema.

### Relationships

One-to-One – This means that each record in one table is related to exactly one record in another table, and vice versa. For example, a business may have a table for customers containing fields like name and birth date, and another table for additional customer details such as shipping and billing address. Each customer in the first table would have one corresponding record in the second table, and each record in the second table would relate to only one customer in the first table. This ensures that each customer's address information is uniquely associated with them.

One-to-many – This is when one record in a table can be linked to multiple records in another table, however each of the records in the second table are only related to one record in the first table. For example, a bakery a customer can order multiple items. This means they can have an individual customer in the ‘customers table’ connected to multiple orders in the ‘orders table’ but all those orders are only connected to the one customer.

Many to many - – This is when many records in a table can be linked to multiple records in another table. For example, an author can write many books and a book can also have many authors.

### Normalization

Normalization is a way of structuring the data in a database to reduce data redundancy and increase data integrity. This can involve sorting data into smaller tables and using relationships to connect them. Normalization can come in the forms of 1NF, 2NF, 3NF and BCNF which are each a level of normalization.

### Modify Query 1

**SELECT population FROM world**

**WHERE name = ‘Germany’;**

### Modify Query 2

**SELECT** name **FROM** world **WHERE** name **LIKE** 'U%'

### Modify Query 3

1. 'name' should be name

### Modify Query 4

**SELECT** name **FROM** world **WHERE** name **LIKE** '%a' **OR** name **LIKE** '%l'

### Modify Query 5

**SELECT** name, population **FROM** world **WHERE** population **BETWEEN** 1000000 **AND** 1250000