**Module 4 – Introduction to DBMS (Assesment)**

* **Write SQL query to solve the problem given below.**

-- **Create the city table**

CREATE TABLE city (

id INT PRIMARY KEY,

city\_name VARCHAR(100),

lat DECIMAL(9,6),

long DECIMAL(9,6),

country\_id INT

);

-- **Insert data into the city table**

INSERT INTO city (id, city\_name, lat, long, country\_id) VALUES

(1, 'Berlin', 52.520008, 13.404954, 1),

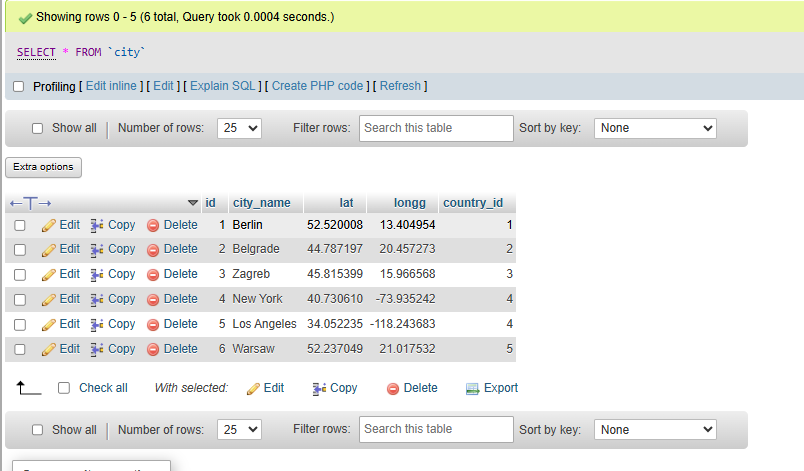
(2, 'Belgrade', 44.787197, 20.457273, 2),

(3, 'Zagreb', 45.815399, 15.966568, 3),

(4, 'New York', 40.730610, -73.935242, 4),

(5, 'Los Angeles', 34.052235, -118.243683, 4),

(6, 'Warsaw', 52.237049, 21.017532, 5);



-- **Create the customer table**

CREATE TABLE customer (

id INT PRIMARY KEY,

customer\_name VARCHAR(100),

city\_id INT,

customer\_address VARCHAR(255),

next\_call\_date DATE,

ts\_inserted DATETIME

);

-- **Insert data into the customer table**

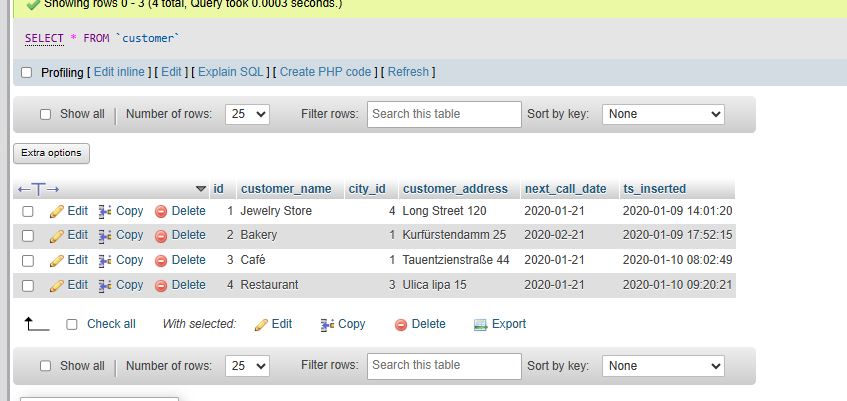
INSERT INTO customer (id, customer\_name, city\_id, customer\_address, next\_call\_date, ts\_inserted) VALUES

(1, 'Jewelry Store', 4, 'Long Street 120', '2020-01-21', '2020-01-09 14:01:20.000'),

(2, 'Bakery', 1, 'Kurfürstendamm 25', '2020-02-21', '2020-01-09 17:52:15.000'),

(3, 'Café', 1, 'Tauentzienstraße 44', '2020-01-21', '2020-01-10 08:02:49.000'),

(4, 'Restaurant', 3, 'Ulica lipa 15', '2020-01-21', '2020-01-10 09:20:21.000');



-- **Create the country table**

CREATE TABLE country (

id INT PRIMARY KEY,

country\_name VARCHAR(100),

country\_name\_eng VARCHAR(100),

country\_code VARCHAR(10)

);

-- **Insert data into the country table**

INSERT INTO country (id, country\_name, country\_name\_eng, country\_code) VALUES

(1, 'Deutschland', 'Germany', 'DEU'),

(2, 'Srbija', 'Serbia', 'SRB'),

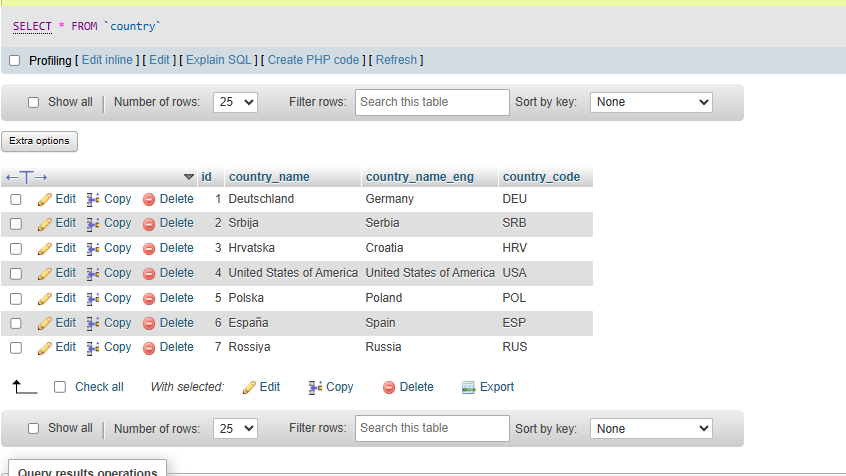
(3, 'Hrvatska', 'Croatia', 'HRV'),

(4, 'United States of America', 'United States of America', 'USA'),

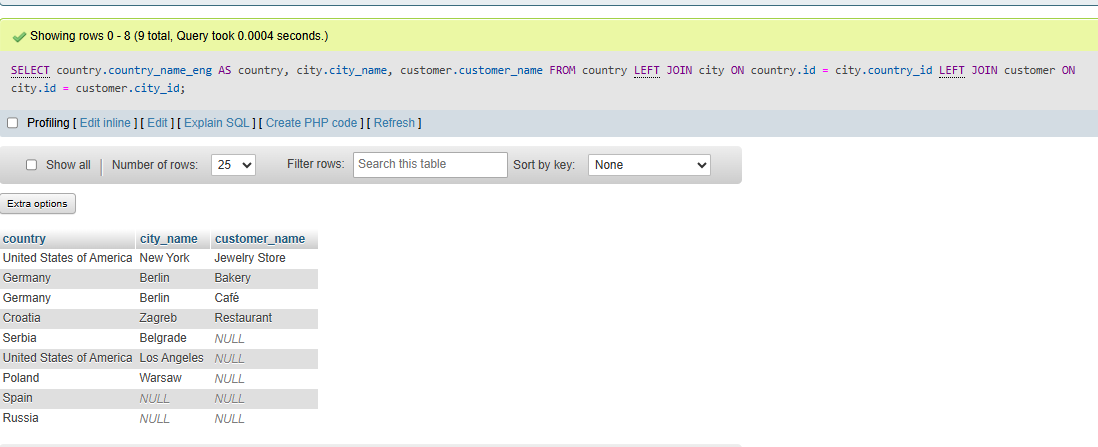
(5, 'Polska', 'Poland', 'POL'),

(6, 'España', 'Spain', 'ESP'),

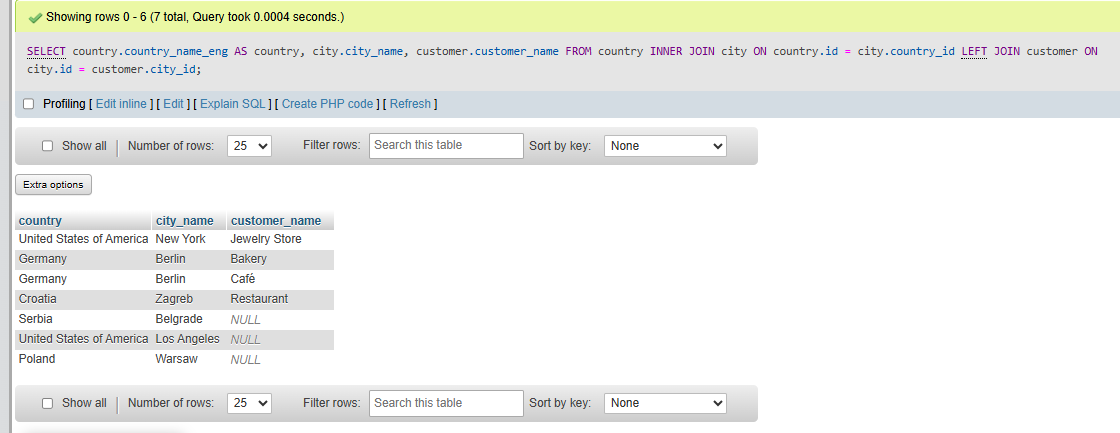
(7, 'Rossiya', 'Russia', 'RUS');



**Task : 1 (join multiple tables using left join) List all Countries and customers related to these countries. For each country displaying its name in English, the name of the city customer is located in as well as the name of the customer.**



**Task : 2 (join multiple tables using both left and inner join) Return the list of all countries that have pairs(exclude countries which are not referenced by any city). For such pairs return all customers.**

****