**Software Engineering Assessment**

**MODULE: 5 (Database)**

**SQL Queries**

1. Create table name: Country



* **Query to create table Country and enter data into it:**

CREATE TABLE Country

(

id int AUTO\_INCREMENT PRIMARY KEY,

country\_name char(50) ,

country\_name\_eng char(50),

country\_code varchar(30)

);

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, 'Espana', 'Spain','ESP'),

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

1. Create table name: City

* **Query to create table City and enter data into it:**

CREATE TABLE City

(

id int AUTO\_INCREMENT PRIMARY KEY,

city\_name varchar(50) ,

lat decimal(10,6),

long decimal(10,6),

country\_id int,

FOREIGN KEY (country\_id) REFERENCES Country (id)

);

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');

1. Create table name: Customer



* **Query to create table Customer and enter data into it:**

CREATE TABLE Customers

(

id int AUTO\_INCREMENT PRIMARY KEY,

customer\_name varchar(50) ,

city\_id int,

FOREIGN KEY (city\_id) REFERENCES City(id),

customer\_address varchar(50),

next\_call\_date date,

ts\_inserted datetime(3)

);

INSERT INTO Customers (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','Kurfurstendamm 25', '2020-02-21', '2020-01-09 17:52:15.000'),

(3, 'Cafe', '1','TauentzienstraBe 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');

* **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.

Return even countries without related cities and customers.

SELECT

Country.id AS country\_id,

Country.country\_name\_eng AS country\_name,

City.city\_name,

Customers.customer\_name

FROM Country

LEFT JOIN City ON Country.id = City.country\_id

LEFT JOIN Customers ON City.id = Customers.city\_id

ORDER BY Country.id, City.id, Customers.id;

* **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.

Return even pairs of not having a single customer

SELECT

Country.country\_name,

City.city\_name,

Customers.customer\_name,

Customers.customer\_address,

Customers.next\_call\_date,

Customers.ts\_inserted

FROM

Country

LEFT JOIN City ON Country.id = City.country\_id

LEFT JOIN Customers ON City.id = Customers.city\_id;