Name: \_\_\_\_\_\_\_\_JACKSON LIN\_\_\_\_\_\_\_\_\_\_\_\_\_

Lesson 9: Insert and Update

1. **Create** a table with the following parameters:
   * CustomerID
   * CustomerName
   * Address
   * City
   * PostalCode
   * Country
   * Email

CREATE TABLE Customers (

CustomerID serial NOT NULL,

CustomerName varchar(255) NOT NULL,

Address varchar(255),

City varchar(255),

PostalCode int,

Country varchar(255),

Email varchar(255) UNIQUE,

PRIMARY KEY (CustomerID)

);

1. Insert 3 rows of data into these columns using **INSERT**. The data you insert should make sense for the column.

INSERT INTO Customers (CustomerID, CustomerName, Address, City, PostalCode, Country, Email)

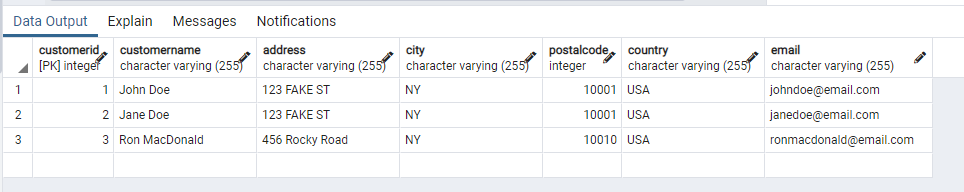
VALUES (1, ‘John Doe’, ‘123 FAKE ST’, ‘NY’, 10001, ‘USA’, ‘johndoe@email.com’);

INSERT INTO Customers (CustomerID, CustomerName, Address, City, PostalCode, Country, Email)

VALUES (2, ‘Jane Doe’, ‘123 FAKE ST’, ‘NY’, 10001, ‘USA’, ‘janedoe@email.com’);

INSERT INTO Customers (CustomerID, CustomerName, Address, City, PostalCode, Country, Email)

VALUES (3, ‘Ron MacDonald’, ‘456 Rocky Road’, ‘NY’, 10010, ‘USA’, ‘ronmacdonald@email.com’);

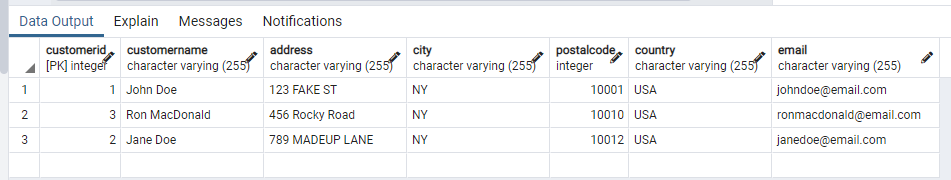


1. Use an **UPDATE** to modify any portion of the data

UPDATE Customers

SET Address = ‘789 MADEUP LANE’, PostalCode = 10012

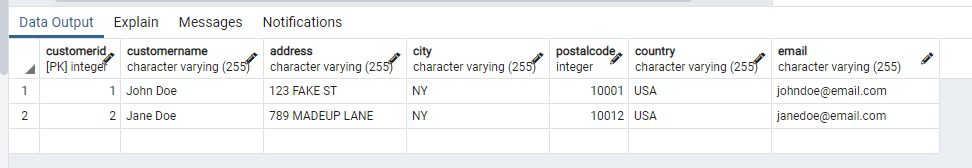
WHERE CustomerID = 2;



1. Finally, write a statement to **delete** one row of data.

DELETE FROM Customers

WHERE CustomerID = 3;



1. Using the following Link <https://github.com/niteen11/cuny_lagcc_micro_credential_data_analytics/tree/main/Track%20A/Unit%205%20-%20SQL_%20Relational%20Databases/guided%20exercise>

First you have to create a table than upload the data ,safe the table in to your Laptop and change the path accordingly.usr the following link for creating table,

<https://github.com/niteen11/cuny_lagcc_micro_credential_data_analytics/blob/main/Track%20A/Unit%205%20-%20SQL_%20Relational%20Databases/guided%20exercise/student.sql>

And attached data set (Student\_data and Student\_marks ) answer the following questions :

|  |
| --- |
| -- students with the highest marks in Unit 4 |
|  | SELECT student\_marks.unit4, student\_marks.student\_reg\_id, student.first\_name, student.last\_name  FROM student INNER JOIN student\_marks  ON student.id = student\_marks.student\_reg\_id  ORDER BY student\_marks.unit4 DESC; |
|  | -- Find students scored between 89 and 100 unit4  SELECT student\_marks.unit4, student\_marks.student\_reg\_id, student.first\_name, student.last\_name  FROM student INNER JOIN student\_marks  ON student.id = student\_marks.student\_reg\_id  WHERE student\_marks.unit4 BETWEEN 89 AND 100; |
|  |  |
|  | Open ended questions: |
|  | -- Take a closer look at the tables that you created and come up with 10 different scenarios/ questions and form SQL |
|  | -- Ask your colleagues |

What is the average of unit 1?

What is the average of unit 2?

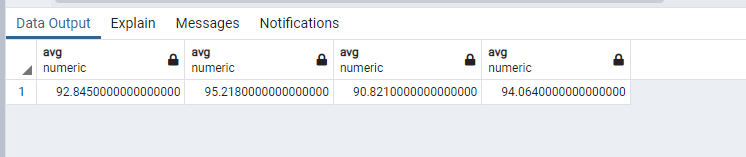
What is the average of unit 3?

What is the average of unit 4?

What is the average of unit 5?

SELECT AVG(unit2), AVG(unit3), AVG(unit4), AVG(unit5)

FROM student\_marks



What is the range of unit 1?

What is the range of unit 2?

What is the range of unit 3?

What is the range of unit 4?

What is the range of unit 5?

SELECT MAX(unit2)-MIN(unit2) AS Range\_Unit2, MAX(unit3)-MIN(unit3) AS RAnge\_Unit3, MAX(unit4)-MIN(unit4) AS Range\_Unit4, MAX(unit5)-MIN(unit5) AS Range\_Unit5

FROM student\_marks;

