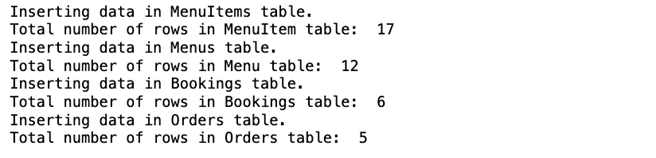
## **Task 1**

Insert data in all four tables in the little\_lemon database using Python. Use the following INSERT queries to populate the tables with relevant data. Use the SHOW TABLE command to check the output and ensure that each query has been executed successfully.



Use the SELECT \* FROM Table\_Name command to check the output and ensure that each query has been executed successfully and the data is inserted.

## **Task 2**

In the first task, you created tables to hold the restaurant’s data. Now Little Lemon requires you to retrieve the following data for each guest:

* Guest first and last names.
* The table number assigned to each guest.

You can help Little Lemon to read this data from the Bookings table using Python.

The SELECT query you need to complete this task is as follows:

all\_bookings = """SELECT GuestFirstName, GuestLastName, TableNo FROM Bookings"""

## **Task 3**

Little Lemon’s queries are returning large volumes of data. They need to find a way to return the data in smaller, more manageable chunks. Let’s see if you can help them to return just the first three items from the menu. In other words, return the first three items from the cursor.

Once the .execute() method is called on the select statement, you will have access to the rows effected. All of this data can returned at once using the  .fetchall() method or returned in smaller chunks using some of cursor’s other methods.  Write the code that instantiates a cursor and executes the select\_stmt but only returns 3 rows.

**Solution for all**

import mysql.connector as connector

# Establish connection between Python and MySQL database via connector API

connection=connector.connect(

user="root", # use your own

password="Natimysql1//", # use your own

)

# Create cursor object to communicate with entire MySQL database

cursor = connection.cursor()

# Setting little\_lemon for use

cursor.execute("use testDb")

# Confirming

print("Database is use is:", connection.database)

print()

print("The existing tables in the testDb database are:")

cursor.execute("SHOW TABLES")

# Print table names

for table in cursor:

print(table)

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#

# Insert query to populate "MenuItems" table is:

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#

insert\_menuitmes="""

INSERT INTO MenuItems (ItemID, Name, Type, Price)

VALUES

(1,'Olives','Starters',5),

(2,'Flatbread','Starters', 5),

(3, 'Minestrone', 'Starters', 8),

(4, 'Tomato bread','Starters', 8),

(5, 'Falafel', 'Starters', 7),

(6, 'Hummus', 'Starters', 5),

(7, 'Greek salad', 'Main Courses', 15),

(8, 'Bean soup', 'Main Courses', 12),

(9, 'Pizza', 'Main Courses', 15),

(10,'Greek yoghurt','Desserts', 7),

(11, 'Ice cream', 'Desserts', 6),

(12, 'Cheesecake', 'Desserts', 4),

(13, 'Athens White wine', 'Drinks', 25),

(14, 'Corfu Red Wine', 'Drinks', 30),

(15, 'Turkish Coffee', 'Drinks', 10),

(16, 'Turkish Coffee', 'Drinks', 10),

(17, 'Kabasa', 'Main Courses', 17);"""

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#

# Insert query to populate "Menu" table is:

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#

insert\_menu="""

INSERT INTO Menus (MenuID,ItemID,Cuisine)

VALUES

(1, 1, 'Greek'),

(1, 7, 'Greek'),

(1, 10, 'Greek'),

(1, 13, 'Greek'),

(2, 3, 'Italian'),

(2, 9, 'Italian'),

(2, 12, 'Italian'),

(2, 15, 'Italian'),

(3, 5, 'Turkish'),

(3, 17, 'Turkish'),

(3, 11, 'Turkish'),

(3, 16, 'Turkish');"""

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#

# Insert query to populate "Bookings" table is:

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#

insert\_bookings="""

INSERT INTO Bookings (BookingID, TableNo, GuestFirstName,

GuestLastName, BookingSlot, EmployeeID)

VALUES

(1,12,'Anna','Iversen','19:00:00',1),

(2, 12, 'Joakim', 'Iversen', '19:00:00', 1),

(3, 19, 'Vanessa', 'McCarthy', '15:00:00', 3),

(4, 15, 'Marcos', 'Romero', '17:30:00', 4),

(5, 5, 'Hiroki', 'Yamane', '18:30:00', 2),

(6, 8, 'Diana', 'Pinto', '20:00:00', 5);"""

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#

# Insert query to populate "Orders" table is:

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#

insert\_orders="""

INSERT INTO Orders (OrderID, TableNo, MenuID, BookingID, Quantity, BillAmount)

VALUES

(1, 12, 1, 1, 2, 86),

(2, 19, 2, 2, 1, 37),

(3, 15, 2, 3, 1, 37),

(4, 5, 3, 4, 1, 40),

(5, 8, 1, 5, 1, 43);"""

print("Inserting data in MenuItems table.")

# Populate MenuItems table

cursor.execute(insert\_menuitmes)

print("Total number of rows in MenuItem table: ", cursor.rowcount)

# Once the query is executed, you commit the change into the database

connection.commit()

print("Inserting data in Menus table.")

# Populate MenuItems table

cursor.execute(insert\_menu)

print("Total number of rows in Menu table: ", cursor.rowcount)

connection.commit()

print("Inserting data in Bookings table.")

# Populate Bookings table

cursor.execute(insert\_bookings)

print("Total number of rows in Bookings table: ", cursor.rowcount)

connection.commit()

print("Inserting data in Orders table.")

# Populate Orders table

cursor.execute(insert\_orders)

print("Total number of rows in Orders table: ", cursor.rowcount)

connection.commit()

# Query to retrieve only first three records from the bookings table is:

all\_Orders = """SELECT \* FROM Orders LIMIT 5;"""

# Execute query

cursor.execute(all\_Orders)

# Fetch fist 3 records in results

results = cursor.fetchall()

# Retrieve column names

cols = cursor.column\_names

# Print column names and records from results using for loop

print("""Data in the "Orders" table:""")

print(cols)

for result in results:

print(result)

all\_bookings = """SELECT GuestFirstName, GuestLastName,

TableNo FROM Bookings;"""

# all\_bookings = """SELECT \* FROM Bookings;"""

# Eexecute query

cursor.execute(all\_bookings)

# Fetch all results that satisfy the query

results = cursor.fetchall()

# Retrieve column names

cols = cursor.column\_names

# Print column names and records from results using for loop

print("""Data in the "Bookings" table:""")

print(cols)

for result in results:

print(result)

# Query to retrieve all menus is:

all\_menus = """SELECT \* FROM Menus;"""

# Execute query

cursor.execute(all\_menus)

# Fetch fist 3 records in results

results = cursor.fetchmany(size=3)

# Retrieve column names

cols = cursor.column\_names

# Print column names and records from results using for loop

print("""Data in the "Menus" table:""")

print(cols)

for result in results:

print(result)

# Remaining records after fetching the first three

results= cursor.fetchall()

for result in results:

print(result)

# Query to retrieve only first three records from the bookings table is:

all\_menus = """SELECT \* FROM Menus LIMIT 3;"""

# Execute query

cursor.execute(all\_menus)

# Fetch fist 3 records in results

results = cursor.fetchall()

# Retrieve column names

cols = cursor.column\_names

# Print column names and records from results using for loop

print("""Data in the "Menus" table:""")

print(cols)

for result in results:

print(result)

# Let's close the cursor and the connection

if connection.is\_connected():

cursor.close()

print("The cursor is closed.")

connection.close()

print("MySQL connection is closed.")

else:

print("Connection is already closed")