Lab01.ipynb

```
#%% md
### Lab 01 - Assignment - By - Vinay S
#%% md
### Q1
#%% md
### Load Required Libraries
#%%
import mysql.connector
import pandas as pd
from streamlit import columns

#%% md
### Connect to DB using Mysql-connector-python package

#%%
# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    #database='e_commerce' # e.g., 'test_db'
)

#%%
db_name = 'e_commerce'

#%%
#############
# Drop the DB

myquery = f"""DROP DATABASE IF EXISTS {db_name};"""
#%%
cursor = connection.cursor()

#%%
cursor.execute(myquery)
#%%
connection.commit()
cursor.close()
#%%
connection.close()
#%%
# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    #database='e_commerce' # e.g., 'test_db'
)

#%%
db_name = 'e_commerce'

#%%
myquery = f"""CREATE DATABASE IF NOT EXISTS {db_name}"""
#%%
cursor = connection.cursor()
cursor.execute(myquery)

#%%
connection.commit()
cursor.close()
#%%
connection.close()

#%%
db_name = 'e_commerce'

# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    database=db_name # e.g., 'test_db'
)
#%%
myquery = """
    CREATE TABLE supplier (
        SUPP_ID INT PRIMARY KEY,
        SUPP_NAME VARCHAR(50),
        SUPP_CITY VARCHAR(50),
        SUPP_PHONE VARCHAR(10)
    );

    CREATE TABLE customer (
        CUS_ID INT NOT NULL,
        CUS_NAME VARCHAR(20) NULL DEFAULT NULL,
        CUS_PHONE VARCHAR(10),
        CUS_CITY VARCHAR(30),
        CUS_GENDER CHAR,
        PRIMARY KEY (CUS_ID)
    );

    CREATE TABLE category (
```

```
        CAT_ID INT NOT NULL,
        CAT_NAME VARCHAR(20) NULL DEFAULT NULL,
        PRIMARY KEY (CAT_ID)
    );

    CREATE TABLE product (
        PRO_ID INT NOT NULL,
        PRO_NAME VARCHAR(20) NULL DEFAULT NULL,
        PRO_DESC VARCHAR(60) NULL DEFAULT NULL,
        CAT_ID INT NOT NULL,
        PRIMARY KEY (PRO_ID),
        FOREIGN KEY (CAT_ID) REFERENCES category (CAT_ID)
    );

    CREATE TABLE product_details (
        PROD_ID INT NOT NULL,
        PRO_ID INT NOT NULL,
        SUPP_ID INT NOT NULL,
        PROD_PRICE INT NOT NULL,
        PRIMARY KEY (PROD_ID),
        FOREIGN KEY (PRO_ID) REFERENCES product (PRO_ID),
        FOREIGN KEY (SUPP_ID) REFERENCES supplier (SUPP_ID)
    );

    CREATE TABLE orders (
        ORD_ID INT NOT NULL,
        ORD_AMOUNT INT NOT NULL,
        ORD_DATE DATE,
        CUS_ID INT NOT NULL,
        PROD_ID INT NOT NULL,
        PRIMARY KEY (ORD_ID),
        FOREIGN KEY (CUS_ID) REFERENCES customer (CUS_ID),
        FOREIGN KEY (PROD_ID) REFERENCES product_details (PROD_ID)
    );

    CREATE TABLE rating (
        RAT_ID INT NOT NULL,
        CUS_ID INT NOT NULL,
        SUPP_ID INT NOT NULL,
        RAT_RATSTARS INT NOT NULL,
        PRIMARY KEY (RAT_ID),
        FOREIGN KEY (SUPP_ID) REFERENCES supplier (SUPP_ID),
        FOREIGN KEY (CUS_ID) REFERENCES customer (CUS_ID)
    );
"""

#%%
cursor = connection.cursor()

#%%
cursor.execute(myquery)

#%%
#connection.commit()
cursor.close()
connection.close()

#%%

#%% md
### Q2
#%%
### Inserting Values into Supplier Table
#%%
db_name = 'e_commerce'

# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    database=db_name # e.g., 'test_db'
)

#%%
myquery = """
INSERT INTO supplier (SUPP_ID, SUPP_NAME, SUPP_CITY, SUPP_PHONE) VALUES
(1, 'Rajesh Retails', 'Delhi', '1234567890'),
(2, 'Appario Ltd.', 'Mumbai', '258963147032'),
(3, 'Knome products', 'Bangalore', '9785462315'),
(4, 'Bansal Retails', 'Kochi', '8975463285'),
(5, 'Mittal Ltd.', 'Lucknow', '7898456532');
"""

#%%
cursor = connection.cursor()

#%%
cursor.execute(myquery)
connection.commit()

#%%
cursor.close()
connection.close()

#%%
### Inserting Values into Customer Table
```

```
#%%
db_name = 'e_commerce'

# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    database=db_name # e.g., 'test_db'
)

#%%
myquery = """
INSERT INTO customer (CUS_ID, CUS_NAME, CUS_PHONE, CUS_CITY, CUS_GENDER) VALUES
(1, 'AAKASH', 9999999999, 'DELHI', 'M'),
(2, 'AMAN', 9785463215, 'NOIDA', 'M'),
(3, 'NEHA', 9999999998, 'MUMBAI', 'F'),
(4, 'MEGHA', 9994562399, 'KOLKATA', 'F'),
(5, 'PULKIT', 7895999999, 'LUCKNOW', 'M');
"""
#%%
cursor = connection.cursor()

#%%
cursor.execute(myquery)
connection.commit()

#%%
cursor.close()
connection.close()

#%%
### Inserting Values into Category Table
#%%
db_name = 'e_commerce'

# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    database=db_name # e.g., 'test_db'
)

#%%
myquery = """
INSERT INTO category (CAT_ID, CAT_NAME) VALUES
(1, 'BOOKS'),
(2, 'GAMES'),
(3, 'GROCERIES'),
(4, 'ELECTRONICS'),
(5, 'CLOTHES');
"""

#%%
cursor = connection.cursor()

#%%
cursor.execute(myquery)
connection.commit()

#%%
cursor.close()
connection.close()

#%%
### Inserting Values into Product Table
#%%
db_name = 'e_commerce'

# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    database=db_name # e.g., 'test_db'
)

#%%
myquery = """
INSERT INTO product (PRO_ID, PRO_NAME, PRO_DESC, CAT_ID) VALUES
(1, 'GTA V', 'DFJDJFDJFDJFDJFJF', 2),
(2, 'TSHIRT', 'DFDFJDFJDKFD', 5),
(3, 'ROG LAPTOP', 'DFNTTNTNTERND', 4),
(4, 'OATS', 'REURENTBTOTH', 3),
(5, 'HARRY POTTER', 'NBEMCTHTJTH', 1);
"""

#%%
cursor = connection.cursor()

#%%
cursor.execute(myquery)
connection.commit()

#%%
cursor.close()
```

```
connection.close()

#%%
### Inserting Values into Product_Details Table
#%%
db_name = 'e_commerce'

# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    database=db_name # e.g., 'test_db'
)

#%%
myquery="""
INSERT INTO product_details (PROD_ID, PRO_ID, SUPP_ID, PROD_PRICE) VALUES
(1, 1, 2, 1500),
(2, 3, 5, 30000),
(3, 5, 1, 3000),
(4, 2, 3, 2500),
(5, 4, 1, 1000);
"""

#%%
cursor = connection.cursor()

#%%
cursor.execute(myquery)
connection.commit()

#%%
cursor.close()
connection.close()

#%%
### Inserting Values into Orders Table
#%%
db_name = 'e_commerce'

# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    database=db_name # e.g., 'test_db'
)

#%%
myquery = """
INSERT INTO orders (ORD_ID, ORD_AMOUNT, ORD_DATE, CUS_ID, PROD_ID) VALUES
(20, 1500, '2021-10-12', 3, 5),
(25, 30500, '2021-09-16', 5, 2),
(26, 2000, '2021-10-05', 1, 1),
(30, 3500, '2021-08-16', 4, 3),
(50, 2000, '2021-10-06', 2, 1);
"""

#%%
cursor = connection.cursor()

#%%
cursor.execute(myquery)
connection.commit()

#%%
cursor.close()
connection.close()

#%%
### Inserting Values into Rating Table
#%%
db_name = 'e_commerce'

# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    database=db_name # e.g., 'test_db'
)

#%%
myquery = """
INSERT INTO rating (RAT_ID, CUS_ID, SUPP_ID, RAT_RATSTARS) VALUES
(1, 2, 2, 4),
(2, 3, 4, 3),
(3, 5, 1, 5),
(4, 1, 3, 2),
(5, 4, 5, 4);
"""

#%%
cursor = connection.cursor()

#%%
```

```
cursor.execute(myquery)
connection.commit()

#%%
cursor.close()
connection.close()

#%%

#%% md
### Q3
#%%
### Q3) Display the number of the customer group by their genders who have placed any order of amount greater than or equal to Rs.3000
#%%
db_name = 'e_commerce'

# Establish a connection to the database
connection = mysql.connector.connect(
    host='localhost',        # e.g., 'localhost'
    user='root',    # e.g., 'root'
    password='',# e.g., 'password'
    database=db_name # e.g., 'test_db'
)

#%%
myquery = """
SELECT CUS_GENDER, COUNT(*) AS num_customers
FROM customer
JOIN orders ON customer.CUS_ID = orders.CUS_ID
WHERE ORD_AMOUNT >= 3000
GROUP BY CUS_GENDER;
"""

#%%
cursor = connection.cursor()

#%%
cursor.execute(myquery)

#%%
output = cursor.fetchall()

#%%
df = pd.DataFrame(output, columns=['CUS_GENDER', 'num_customers'])

#%%
df.head()

#%%

#%% md
### Q4
#%%
### Q4) Display all the order along with product name ordered by a customer having Customer_Id=2;

#%%
myquery = """
SELECT orders.ORD_ID, orders.ORD_AMOUNT, orders.ORD_DATE, product.PRO_NAME
FROM orders
JOIN product_details ON orders.PROD_ID = product_details.PROD_ID
JOIN product ON product_details.PRO_ID = product.PRO_ID
WHERE orders.CUS_ID = 2;
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)

#%%
output = cursor.fetchall()
print(output)

#%%
df = pd.DataFrame(output, columns=['orders.ORD_ID', 'orders.ORD_AMOUNT', 'orders.ORD_DATE', 'product.PRO_NAME'])

df.head()

#%%
### Alternative Query as provided by Faculty
myquery = """
select `orders`.*, product.pro_name
from `orders` ,product_details,product
where `orders`.cus_id=2 and `orders`.prod_id=product_details.prod_id and product_details.pro_id=product.pro_id;
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['orders.ORD_ID', 'orders.ORD_AMOUNT', 'orders.ORD_DATE', "product.cust_id", 'product.prod_id', 'product.pro_name'])
df.head()

```

```
#%%

#%% md
### Q5

#%%
### Q5) Display the Supplier details who can supply more than one product.

#%%
myquery = """
SELECT SUPP_ID, SUPP_NAME
FROM supplier
WHERE SUPP_ID IN (
    SELECT SUPP_ID
    FROM product_details
    GROUP BY SUPP_ID
    HAVING COUNT(*) > 1
);
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['SUPP_ID', 'SUPP_NAME'])
df.head()

#%%
# Alternative Query provided by the Faculty
myquery = """
select supplier.*
from supplier,product_details
where supplier.supp_id in
(select product_details.supp_id from product_details
group by product_details.supp_id having count(product_details.supp_id)>1)
group by supplier.supp_id;
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['SUPP_ID','SUPP_NAME','SUPP_CITY','SUPP_PHONE'])
df

#%%

#%% md
### Q6

#%%
### Q6) Find the category of the product whose order amount is minimum.

#%%
myquery = """
SELECT category.CAT_NAME
FROM category
JOIN product ON category.CAT_ID = product.CAT_ID
JOIN product_details ON product.PRO_ID = product_details.PRO_ID
JOIN orders ON product_details.PROD_ID = orders.PROD_ID
WHERE orders.ORD_AMOUNT = (
    SELECT MIN(ORD_AMOUNT)
    FROM orders
);
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['category.CAT_NAME'])
df.head()

#%%
# Alternative Query Provided by the Faculty
myquery = """
SELECT category.cat_id, category.cat_name, MIN(orders.ord_amount) AS min_order_amount
FROM orders
INNER JOIN product_details ON orders.prod_id = product_details.prod_id
INNER JOIN product ON product.pro_id = product_details.pro_id
INNER JOIN category ON category.cat_id = product.cat_id
GROUP BY category.cat_id, category.cat_name;
```

```
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
# Another Query from Faculty
myquery = """
select category.*
from `orders` inner join product_details on `orders`.prod_id=product_details.prod_id
inner join product on product.pro_id=product_details.pro_id
inner join category on category.cat_id=product.cat_id having min(`orders`.ord_amount);
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['category.cat_id','category.cat_name'])

#%%
df.head()

#%%

#%% md
### Q7

#%%
### Q7) Display the Id and Name of the Product ordered after “2021-10-05”.

#%%
myquery = """
SELECT product.PRO_ID, product.PRO_NAME
FROM product
JOIN product_details ON product.PRO_ID = product_details.PRO_ID
JOIN orders ON product_details.PROD_ID = orders.PROD_ID
WHERE orders.ORD_DATE > '2021-10-05';
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['product.PRO_ID','product.PRO_NAME'])
df.head()

#%%

#%% md
### Q8
#%%
### Q8) Print the top 3 supplier name and id and rating on the basis of their rating along with the customer name who has given the rating.

#%%
myquery = """
SELECT supplier.SUPP_ID, supplier.SUPP_NAME, AVG(rating.RAT_RATSTARS) AS avg_rating, customer.CUS_NAME
FROM supplier
JOIN rating ON supplier.SUPP_ID = rating.SUPP_ID
JOIN customer ON rating.CUS_ID = customer.CUS_ID
GROUP BY supplier.SUPP_ID, supplier.SUPP_NAME
ORDER BY avg_rating DESC
LIMIT 3;
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['SUPPLIER_ID', 'SUPPLIER_NAME', 'AVG_Rating', 'CUSTOMER_NAME'])
df

#%%
# Alternative Query as provided by Faculty
myquery = """
select supplier.supp_id,supplier.supp_name,customer.cus_name,rating.rat_ratstars
from rating inner join supplier on rating.supp_id=supplier.supp_id
```

```
inner join customer on rating.cus_id=customer.cus_id order by rating.rat_ratstars desc limit 3;
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['Supplier_ID', 'Supplier_Name', 'CUS_NAME', 'Rating'])
df.head()

#%%

#%% md
### Q9

#%%
myquery = """
SELECT CUS_NAME, CUS_GENDER
FROM customer
WHERE CUS_NAME LIKE 'A%' OR CUS_NAME LIKE '%A';
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['CUS_NAME', 'CUS_GENDER'])
df.head()

#%%

#%% md
### Q10

#%%
### Q10) Display the total order amount of the male customers

#%%
myquery = """
SELECT SUM(orders.ORD_AMOUNT) AS total_order_amount
FROM orders
JOIN customer ON orders.CUS_ID = customer.CUS_ID
WHERE customer.CUS_GENDER = 'M';
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['Amount'])
df.head()

#%%

#%% md
### Q11

#%%
### Q11) Display all the Customers left outer join with  the orders

#%%
myquery = """
SELECT
customer.CUS_ID, customer.CUS_NAME, customer.CUS_PHONE, customer.CUS_CITY, customer.CUS_GENDER, orders.ORD_ID, orders.ORD_AMOUNT, orders.ORD_DATE
FROM customer
LEFT JOIN orders ON customer.CUS_ID = orders.CUS_ID;
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['CUST_ID', 'CUST_NAME', 'PHONE', 'CITY', 'GENDER', 'ORD_ID', 'AMOUNT', 'ORD_DATE'])
df

#%%
# Alternative Query as provided by Faculty
```

```

#%%
myquery = """
select *  from customer left outer join `orders` on customer.cus_id=`orders`.cus_id;
"""

#%%
cursor = connection.cursor()
cursor.execute(myquery)
output = cursor.fetchall()

#%%
print(output)

#%%
df = pd.DataFrame(output, columns=['CUS_ID','CUS_NAME','CUS_PHONE','CUS_CITY','CUS_GENDER','ORD_ID','ORD_AMOUNT','ORD_DATE','CUS_ID','PROD_ID'])
df

#%%
cursor.close()
connection.close()

#%%

#%%

#%% md
### Lab-01 - Assignment - Concluded
```