

ASSIGNMENT 1

NAME: SHIVAM SINGH

DATABASE:TECHSHOP

TASK1:

Database Design:

1. Create the database named "TechShop"
2. Define the schema for the Customers, Products, Orders, OrderDetails and Inventory tables based on the provided schema.
3. Create an ERD (Entity Relationship Diagram) for the database.
4. Create appropriate Primary Key and Foreign Key constraints for referential integrity.
5. Insert at least 10 sample records into each of the following tables. a. Customers b. Products c. Orders d. OrderDetails e. Inventory

Q.1 CREATE DATABASE TECHSHOP.

```
mysql> CREATE DATABASE TECHSHOP;
Query OK, 1 row affected (0.01 sec)

mysql> SHOW DATABASES;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'DATABASE' at line 1
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| hexabatch2 |
| information_schema |
| mysql |
| performance_schema |
| sakila |
| sys |
| techshop |
| world |
+-----+
8 rows in set (0.00 sec)
```

Q.2 CUSTOMER TABLE:

```
MySQL 8.0 Command Line Client
| techshop          |
| world            |
+-----+
8 rows in set (0.00 sec)

mysql> USE TECHSHOP
Database changed
mysql> CREATE TABLE CUSTOMER(
    -> CUSTOMERID INT PRIMARY KEY NOT NULL,
    -> FIRSTNAME VARCHAR(50),
    -> LASTNAME VARCHAR(30),
    -> EMAIL TEXT,
    -> PHONE BIGINT,
    -> ADDRESS LONGTEXT);
Query OK, 0 rows affected (0.03 sec)

mysql> SELECT*CUSTOMER;
ERROR 1064 (42000): You have an error in your SQL syntax; check the m
mysql> SELECT* CUSTOMER;
ERROR 1064 (42000): You have an error in your SQL syntax; check the m
mysql> DESC CUSTOMER;
+-----+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CUSTOMERID | int        | NO   | PRI | NULL    |       |
| FIRSTNAME  | varchar(50) | YES  |     | NULL    |       |
| LASTNAME   | varchar(30) | YES  |     | NULL    |       |
| EMAIL      | text        | YES  |     | NULL    |       |
| PHONE      | bigint     | YES  |     | NULL    |       |
| ADDRESS    | longtext   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

PRODUCT TABLE:

```
MySQL 8.0 Command Line Client

mysql> CREATE TABLE PRODUCTS(
    -> PRODUCTID INT PRIMARY KEY NOT NULL,
    -> PRODUCTNAME VARCHAR(20),
    -> DESCRIPTION LONGTEXT,
    -> PRICE DOUBLE);
Query OK, 0 rows affected (0.02 sec)

mysql> DESC PRODUCTS;
+-----+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| PRODUCTID  | int        | NO   | PRI | NULL    |       |
| PRODUCTNAME| varchar(20)| YES  |     | NULL    |       |
| DESCRIPTION | longtext   | YES  |     | NULL    |       |
| PRICE      | double     | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

ORDERS TABLE:

```
mysql> CREATE TABLE ORDERS(
    -> ORDERID INT PRIMARY KEY NOT NULL,
    -> CUSTOMERID INT,
    -> FOREIGN KEY(CUSTOMERID) REFERENCES CUSTOMER(CUSTOMERID),
    -> ORDERDATE DATE,
    -> TOTALAMOUNT DOUBLE);
Query OK, 0 rows affected (0.07 sec)
```

```
mysql> DESC ORDERS;
```

Field	Type	Null	Key	Default	Extra
ORDERID	int	NO	PRI	NULL	
CUSTOMERID	int	YES	MUL	NULL	
ORDERDATE	date	YES		NULL	
TOTALAMOUNT	double	YES		NULL	

4 rows in set (0.00 sec)

ORDERDETAILS TABLE:

```
MySQL 8.0 Command Line Client
mysql> CREATE TABLE ORDERDETAILS(
    ->     ORDERDETAILID INT PRIMARY KEY NOT NULL,
    ->     ORDERID INT,
    ->     FOREIGN KEY(ORDERID) REFERENCES ORDERS(ORDERID),
    ->     PRODUCTID INT,
    ->     FOREIGN KEY(PRODUCTID) REFERENCES PRODUCTS(PRODUCTID),
    ->     QUANTITY INT);
Query OK, 0 rows affected (0.08 sec)
```

```
mysql> DESC ORDERDETAILS;
```

Field	Type	Null	Key	Default	Extra
ORDERDETAILID	int	NO	PRI	NULL	
ORDERID	int	YES	MUL	NULL	
PRODUCTID	int	YES	MUL	NULL	
QUANTITY	int	YES		NULL	

4 rows in set (0.00 sec)

```
mysql>
```

INVENTORY TABLE:

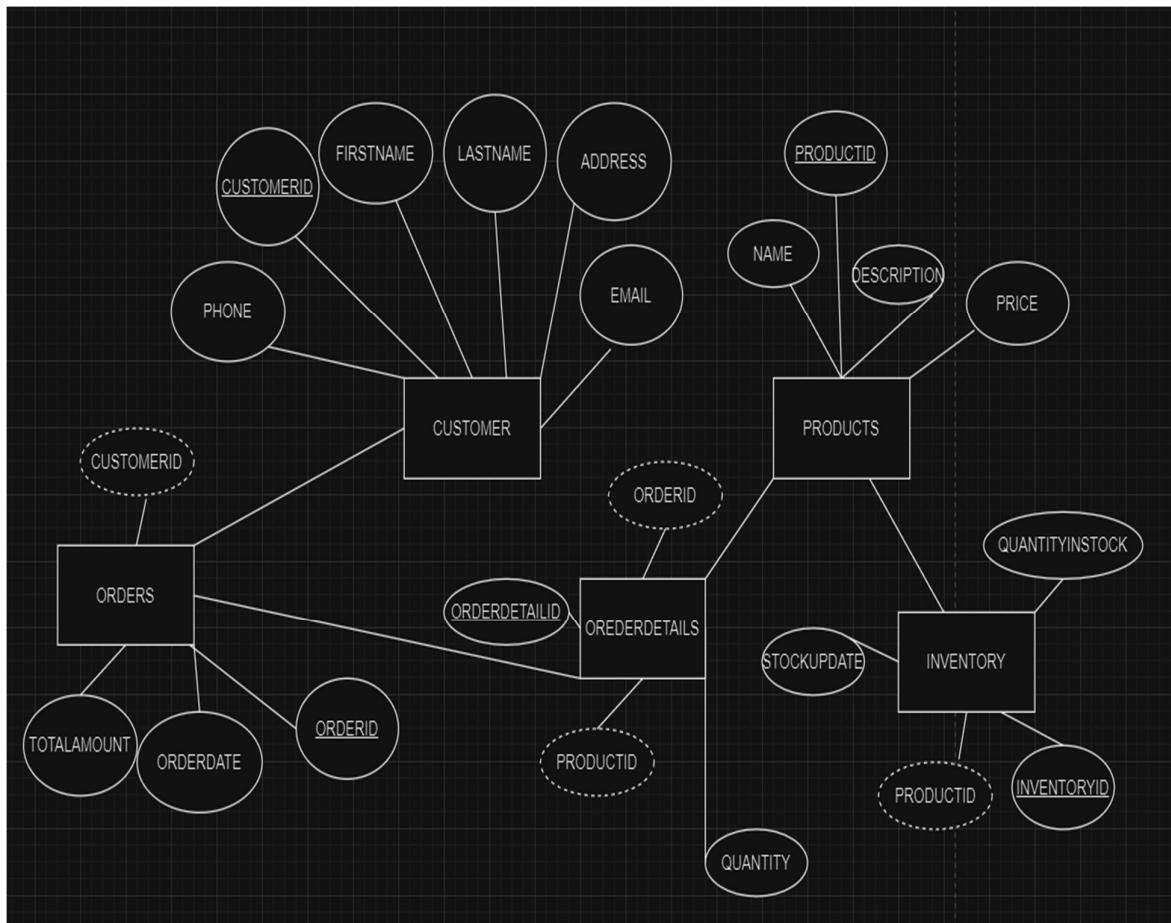
```
mysql> CREATE TABLE INVENTORY(
    -> INVENTORYID INT PRIMARY KEY NOT NULL,
    -> PRODUCTID INT,
    -> FOREIGN KEY(PRODUCTID) REFERENCES PRODUCTS(PRODUCTID),
    -> QUANTITYINSTOCK INT,
    -> LASTSTOCKUPDATE DATE);
Query OK, 0 rows affected (0.15 sec)
```

```
mysql> DESC INVENTORY;
```

Field	Type	Null	Key	Default	Extra
INVENTORYID	int	NO	PRI	NULL	
PRODUCTID	int	YES	MUL	NULL	
QUANTITYINSTOCK	int	YES		NULL	
LASTSTOCKUPDATE	date	YES		NULL	

4 rows in set (0.00 sec)

Q.3 ER DIAGRAM



Q.4

ORDERS TABLE

CREATE TABLE ORDERS(

```
-> ORDERID INT PRIMARY KEY NOT NULL,  
-> CUSTOMERID INT,  
-> FOREIGN KEY(CUSTOMERID) REFERENCES CUSTOMER(CUSTOMERID),  
-> ORDERDATE DATE,  
-> TOTALAMOUNT DOUBLE);
```

Q.5 INSERT 10 ENTRIES:

1.CUSTOMER:

The screenshot shows a terminal window for MySQL 8.0 Command Line Client. The user has created a table named 'CUSTOMER' with columns 'CUSTOMERID', 'FIRSTNAME', 'LASTNAME', 'EMAIL', 'PHONE', and 'ADDRESS'. After creating the table, the user inserts 10 rows of data. The data is as follows:

CUSTOMERID	FIRSTNAME	LASTNAME	EMAIL	PHONE	ADDRESS
1	Abhi	Kumar	abhi@gmail.com	100	ABC
2	Abhay	Singh	abhay@gmail.com	101	Bhopal
3	Abhishek	Sharma	abhishek@gmail.com	102	chennai
4	Bebo	Kumari	bebo@gmail.com	103	delhi
5	Chris	Gayle	chris@gmail.com	104	jamaica
6	David	jones	david@gmail.com	105	ahmedabad
7	Falguni	Kumari	falguni@gmail.com	106	surat
8	hari	om	hari@gmail.com	107	ranchi
9	shivam	singh	shivam@gmail.com	108	hyderabad
10	harsh	raj	harsh@gmail.com	109	mumbai

After inserting the data, the user runs a select query to verify the 10 rows were inserted successfully.

2.PRODUCT:

```
MySQL 8.0 Command Line Client

-> (4, 'Coffee Maker', 'Programmable coffee maker with timer', 59.95),
-> (5, 'Backpack', 'Durable backpack for laptops and tablets', 39.99),
-> (6, 'Smartwatch', 'Fitness tracker with heart rate monitor', 129.00),
-> (7, 'Bluetooth Speaker', 'Portable speaker with built-in microphone', 34.50),
-> (8, 'Digital Camera', '20MP digital camera with 4K video recording', 299.99),
-> (9, 'Wireless Mouse', 'Ergonomic wireless mouse with customizable buttons', 24.95),
->
-> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 11
mysql> INSERT INTO PRODUCTS VALUES((1, 'Laptop', 'High-performance laptop with SSD', 999.99),
-> (2, 'Smartphone', '6.5-inch display, 128GB storage', 499.50),
-> (3, 'Headphones', 'Over-ear headphones with noise cancellation', 89.99),
-> (4, 'Coffee Maker', 'Programmable coffee maker with timer', 59.95),
-> (5, 'Backpack', 'Durable backpack for laptops and tablets', 39.99),
-> (6, 'Smartwatch', 'Fitness tracker with heart rate monitor', 129.00),
-> (7, 'Bluetooth Speaker', 'Portable speaker with built-in microphone', 34.50),
-> (8, 'Digital Camera', '20MP digital camera with 4K video recording', 299.99),
-> (9, 'Wireless Mouse', 'Ergonomic wireless mouse with customizable buttons', 24.95),
-> (10, 'External Hard Drive', '2TB external hard drive for data storage', 79.99);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 10
mysql> INSERT INTO PRODUCTS VALUES((1, 'Laptop', 'High-performance laptop with SSD', 999.99);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1
mysql> INSERT INTO PRODUCTS VALUES(1, 'Laptop', 'High-performance laptop with SSD', 999.99),
-> (2, 'Smartphone', '6.5-inch display, 128GB storage', 499.50),
-> (3, 'Headphones', 'Over-ear headphones with noise cancellation', 89.99),
-> (4, 'Coffee Maker', 'Programmable coffee maker with timer', 59.95),
-> (5, 'Backpack', 'Durable backpack for laptops and tablets', 39.99),
-> (6, 'Smartwatch', 'Fitness tracker with heart rate monitor', 129.00),
-> (7, 'Bluetooth Speaker', 'Portable speaker with built-in microphone', 34.50),
-> (8, 'Digital Camera', '20MP digital camera with 4K video recording', 299.99),
-> (9, 'Wireless Mouse', 'Ergonomic wireless mouse with customizable buttons', 24.95),
-> (10, 'External Hard Drive', '2TB external hard drive for data storage', 79.99);
Query OK, 10 rows affected (0.01 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM PRODUCTS;
+-----+-----+-----+-----+
| PRODUCTID | PRODUCTNAME | DESCRIPTION | PRICE |
+-----+-----+-----+-----+
| 1 | Laptop | High-performance laptop with SSD | 999.99 |
| 2 | Smartphone | 6.5-inch display, 128GB storage | 499.5 |
| 3 | Headphones | Over-ear headphones with noise cancellation | 89.99 |
| 4 | Coffee Maker | Programmable coffee maker with timer | 59.95 |
| 5 | Backpack | Durable backpack for laptops and tablets | 39.99 |
| 6 | Smartwatch | Fitness tracker with heart rate monitor | 129 |
| 7 | Bluetooth Speaker | Portable speaker with built-in microphone | 34.5 |
| 8 | Digital Camera | 20MP digital camera with 4K video recording | 299.99 |
| 9 | Wireless Mouse | Ergonomic wireless mouse with customizable buttons | 24.95 |
| 10 | External Hard Drive | 2TB external hard drive for data storage | 79.99 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

3.ORDERS:

```
MySQL 8.0 Command Line Client

mysql>
mysql> SELECT*FROM ORDERS
-> ;
+-----+-----+-----+-----+
| ORDERID | CUSTOMERID | ORDERDATE | TOTALAMOUNT |
+-----+-----+-----+-----+
|      11 |          1 | 2024-01-11 |       99.5 |
|      12 |          2 | 2024-01-12 |     179.95 |
|      13 |          3 | 2024-01-13 |        45 |
|      14 |          4 | 2024-01-14 |    299.99 |
|      15 |          5 | 2024-01-15 |    129.75 |
|      16 |          6 | 2024-01-16 |     54.99 |
|      17 |          7 | 2024-01-17 |      89.5 |
|      18 |          8 | 2024-01-18 |     199 |
|      19 |          9 | 2024-01-19 |     74.95 |
|      20 |         10 | 2024-01-20 |   149.99 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql> ■
```

4.ORDERDETAILS:

```
mysql> INSERT INTO ORDERDETAILS (ORDERDETAILID, ORDERID, PRODUCTID, QUANTITY)
-> VALUES
->      (1, 11, 1, 2),
->      (2, 12, 2, 1),
->      (3, 13, 3, 4),
->      (4, 14, 4, 3),
->      (5, 15, 5, 5),
->      (6, 16, 6, 1),
->      (7, 17, 7, 3),
->      (8, 18, 8, 2),
->      (9, 19, 9, 4),
->      (10, 20, 10, 1);
Query OK, 10 rows affected (0.03 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM ORDERDETAILS;
+-----+-----+-----+-----+
| ORDERDETAILID | ORDERID | PRODUCTID | QUANTITY |
+-----+-----+-----+-----+
|          1 |     11 |          1 |        2 |
|          2 |     12 |          2 |        1 |
|          3 |     13 |          3 |        4 |
|          4 |     14 |          4 |        3 |
|          5 |     15 |          5 |        5 |
|          6 |     16 |          6 |        1 |
|          7 |     17 |          7 |        3 |
|          8 |     18 |          8 |        2 |
|          9 |     19 |          9 |        4 |
|         10 |     20 |         10 |        1 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

5.INVENTORY:

```
mysql> INSERT INTO INVENTORY (INVENTORYID, PRODUCTID, QUANTITYINSTOCK, LASTSTOCKUPDATE)
-> VALUES
->     (1, 1, 50, '2024-01-01'),
->     (2, 2, 100, '2024-01-02'),
->     (3, 3, 25, '2024-01-03'),
->     (4, 4, 75, '2024-01-04'),
->     (5, 5, 60, '2024-01-05'),
->     (6, 6, 30, '2024-01-06'),
->     (7, 7, 40, '2024-01-07'),
->     (8, 8, 90, '2024-01-08'),
->     (9, 9, 20, '2024-01-09'),
->     (10, 10, 70, '2024-01-10');
Query OK, 10 rows affected (0.01 sec)
Records: 10  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM INVENTORY;
+-----+-----+-----+-----+
| INVENTORYID | PRODUCTID | QUANTITYINSTOCK | LASTSTOCKUPDATE |
+-----+-----+-----+-----+
|      1 |       1 |        50 | 2024-01-01 |
|      2 |       2 |       100 | 2024-01-02 |
|      3 |       3 |        25 | 2024-01-03 |
|      4 |       4 |        75 | 2024-01-04 |
|      5 |       5 |        60 | 2024-01-05 |
|      6 |       6 |        30 | 2024-01-06 |
|      7 |       7 |        40 | 2024-01-07 |
|      8 |       8 |        90 | 2024-01-08 |
|      9 |       9 |        20 | 2024-01-09 |
|     10 |      10 |        70 | 2024-01-10 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

TASK2:

Select, Where, Between, AND, LIKE:

1. Write an SQL query to retrieve the names and emails of all customers.
2. Write an SQL query to list all orders with their order dates and corresponding customer names.
3. Write an SQL query to insert a new customer record into the "Customers" table. Include customer information such as name, email, and address.
4. Write an SQL query to update the prices of all electronic gadgets in the "Products" table by increasing them by 10%.
5. Write an SQL query to delete a specific order and its associated order details from the "Orders" and "OrderDetails" tables. Allow users to input the order ID as a parameter.
6. Write an SQL query to insert a new order into the "Orders" table. Include the customer ID, order date, and any other necessary information.
7. Write an SQL query to update the contact information (e.g., email and address) of a specific customer in the "Customers" table. Allow users to input the customer ID and new contact information.
8. Write an SQL query to recalculate and update the total cost of each order in the "Orders" table based on the prices and quantities in the "OrderDetails" table.
9. Write an SQL query to delete all orders and their associated order details for a specific customer from the "Orders" and "OrderDetails" tables. Allow users to input the customer ID
10. Write an SQL query to insert a new electronic gadget product into the "Products" table, including product name, category, price, and any other relevant details.
11. Write an SQL query to update the status of a specific order in the "Orders" table (e.g., from "Pending" to "Shipped"). Allow users to input the order ID and the new status.
12. Write an SQL query to calculate and update the number of orders placed by each customer in the "Customers" table based on the data in the "Orders" table.

1.

```
mysql> select firstname,lastname,email from customer;
+-----+-----+-----+
| firstname | lastname | email
+-----+-----+-----+
| Abhi     | Kumar   | abhi@gmail.com
| Abhay    | Singh   | abhay@gmail.com
| Abhishek | Sharma  | abhishek@gmail.com
| Bebo     | Kumari  | bebo@gmail.com
| Chris    | Gayle   | chris@gmail.com
| David    | jones   | david@gmail.com
| Falguni  | Kumari  | falguni@gmail.com
| hari     | om      | hari@gmail.com
| shivam   | singh   | shivam@gmail.com
| harsh    | raj     | harsh@gmail.com
+-----+-----+-----+
10 rows in set (0.00 sec)

mysql>
```

2.

```
5
mysql> select orderid,customerid,orderdate from orders;
+-----+-----+-----+
| orderid | customerid | orderdate |
+-----+-----+-----+
|       11 |          1 | 2024-01-11 |
|       12 |          2 | 2024-01-12 |
|       13 |          3 | 2024-01-13 |
|       14 |          4 | 2024-01-14 |
|       15 |          5 | 2024-01-15 |
|       16 |          6 | 2024-01-16 |
|       17 |          7 | 2024-01-17 |
|       18 |          8 | 2024-01-18 |
|       19 |          9 | 2024-01-19 |
|       20 |         10 | 2024-01-20 |
+-----+-----+-----+
10 rows in set (0.00 sec)

J
mysql>
```

3.

```
mysql> insert into customer values(11,'mani','shankar','mani@gmail.com',300,'bokaro');
Query OK, 1 row affected (0.00 sec)

mysql> select*from customer;
+-----+-----+-----+-----+-----+-----+
| CUSTOMERID | FIRSTNAME | LASTNAME | EMAIL | PHONE | ADDRESS |
+-----+-----+-----+-----+-----+-----+
| 1 | Abhi | Kumar | abhi@gmail.com | 100 | ABC |
| 2 | Abhay | Singh | abhay@gmail.com | 101 | Bhopal |
| 3 | Abhishek | Sharma | abhishek@gmail.com | 102 | chennai |
| 4 | Bebo | Kumari | bebo@gmail.com | 103 | delhi |
| 5 | Chris | Gayle | chris@gmail.com | 104 | jamaica |
| 6 | David | jones | david@gmail.com | 105 | ahmedabad |
| 7 | Falguni | Kumari | falguni@gmail.com | 106 | surat |
| 8 | hari | om | hari@gmail.com | 107 | ranchi |
| 9 | shivam | singh | shivam@gmail.com | 108 | hyderabad |
| 10 | harsh | raj | harsh@gmail.com | 109 | mumbai |
| 11 | mani | shankar | mani@gmail.com | 300 | bokaro |
+-----+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)

mysql>
```

4.

```
MySQL 8.0 Command Line Client

mysql> select*from product;
ERROR 1146 (42S02): Table 'techshop.product' doesn't exist
mysql> select*from products;
+-----+-----+-----+-----+
| PRODUCTID | PRODUCTNAME | DESCRIPTION | PRICE |
+-----+-----+-----+-----+
| 1 | Laptop | High-performance laptop with SSD | 999.99 |
| 2 | Smartphone | 6.5-inch display, 128GB storage | 499.5 |
| 3 | Headphones | Over-ear headphones with noise cancellation | 89.99 |
| 4 | Coffee Maker | Programmable coffee maker with timer | 59.95 |
| 5 | Backpack | Durable backpack for laptops and tablets | 39.99 |
| 6 | Smartwatch | Fitness tracker with heart rate monitor | 129 |
| 7 | Bluetooth Speaker | Portable speaker with built-in microphone | 34.5 |
| 8 | Digital Camera | 20MP digital camera with 4K video recording | 299.99 |
| 9 | Wireless Mouse | Ergonomic wireless mouse with customizable buttons | 24.95 |
| 10 | External Hard Drive | 2TB external hard drive for data storage | 79.99 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql> select UPDATE Products
      -> SET price = price * 1.10;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server
SET price = price * 1.10' at line 1
mysql> UPDATE Products
      -> SET price = price * 1.10;
Query OK, 10 rows affected (0.01 sec)
Rows matched: 10  Changed: 10  Warnings: 0

mysql> select*from products;
+-----+-----+-----+-----+
| PRODUCTID | PRODUCTNAME | DESCRIPTION | PRICE |
+-----+-----+-----+-----+
| 1 | Laptop | High-performance laptop with SSD | 1099.989 |
| 2 | Smartphone | 6.5-inch display, 128GB storage | 549.45 |
| 3 | Headphones | Over-ear headphones with noise cancellation | 98.989 |
| 4 | Coffee Maker | Programmable coffee maker with timer | 65.94500000000001 |
| 5 | Backpack | Durable backpack for laptops and tablets | 43.98900000000004 |
| 6 | Smartwatch | Fitness tracker with heart rate monitor | 141.9 |
| 7 | Bluetooth Speaker | Portable speaker with built-in microphone | 37.95 |
| 8 | Digital Camera | 20MP digital camera with 4K video recording | 329.98900000000003 |
| 9 | Wireless Mouse | Ergonomic wireless mouse with customizable buttons | 27.445 |
| 10 | External Hard Drive | 2TB external hard drive for data storage | 87.989 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql>
```

5.

```
mysql>
mysql> DECLARE @OrderID INT;
ERROR 1064 (42000): You have an error in your SQL syntax;
mysql>
mysql> -- Set the order ID parameter to the desired value
mysql> SET @OrderID = <your_order_id>;
ERROR 1064 (42000): You have an error in your SQL syntax;
mysql>
mysql> -- Delete from OrderDetails table
mysql> DELETE FROM OrderDetails
    -> WHERE OrderID = @OrderID;
Query OK, 0 rows affected (0.00 sec)

mysql>
mysql> -- Delete from Orders table
mysql> DELETE FROM Orders
    -> WHERE OrderID = @OrderID;
Query OK, 0 rows affected (0.00 sec)

mysql>
mysql> -
```

6.

```
mysql>
mysql> insert into orders values(21,12,2024-01-21,100);
ERROR 1292 (22007): Incorrect date value: '2002' for column 'ORDERDATE' at row 1
mysql> insert into orders values(21,12,'2024-01-21',100);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fa
mysql> insert into orders values(21,11,'2024-01-21',100.00);
Query OK, 1 row affected (0.00 sec)

mysql> select* from orders;
+-----+-----+-----+-----+
| ORDERID | CUSTOMERID | ORDERDATE | TOTALAMOUNT |
+-----+-----+-----+-----+
|      11 |          1 | 2024-01-11 |        99.5 |
|      12 |          2 | 2024-01-12 |      179.95 |
|      13 |          3 | 2024-01-13 |         45 |
|      14 |          4 | 2024-01-14 |      299.99 |
|      15 |          5 | 2024-01-15 |      129.75 |
|      16 |          6 | 2024-01-16 |       54.99 |
|      17 |          7 | 2024-01-17 |       89.5 |
|      18 |          8 | 2024-01-18 |       199 |
|      19 |          9 | 2024-01-19 |       74.95 |
|      20 |         10 | 2024-01-20 |      149.99 |
|      21 |         11 | 2024-01-21 |       100 |
+-----+-----+-----+-----+
11 rows in set (0.00 sec)
```

7.

```
mysql> Delimiter @@
mysql> CREATE PROCEDURE disp(IN cust_id int, ema text, addr text)
      -> BEGIN
      -> UPDATE Customer SET EMAIL=ema, ADDRESS=addr WHERE CUSTOMERID=cust_id;
      -> END @@
Query OK, 0 rows affected (0.05 sec)
```

```
mysql> DELIMITER ;
mysql> CALL DISP(11,'SHIVASINGH414@GMAIL.COM','BOKARO STEEL CITY');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> SELECT * FROM CUSTOMER;
```

CUSTOMERID	FIRSTNAME	LASTNAME	EMAIL	PHONE	ADDRESS	OrdersPlaced
1	Abhi	Kumar	abhi@gmail.com	100	ABC	1
2	Abhay	Singh	abhay@gmail.com	101	Bhopal	1
3	Abhishek	Sharma	abhishek@gmail.com	102	chennai	1
4	Bebo	Kumari	bebo@gmail.com	103	delhi	1
5	Chris	Gayle	chris@gmail.com	104	jamaica	1
6	David	jones	david@gmail.com	105	ahmedabad	1
7	Falguni	Kumari	falguni@gmail.com	106	surat	1
8	hari	om	hari@gmail.com	107	ranchi	1
9	shivam	singh	shivam@gmail.com	108	hyderabad	1
10	harsh	raj	harsh@gmail.com	109	mumbai	1
11	mani	shankar	SHIVASINGH414@GMAIL.COM	300	BOKARO STEEL CITY	1

11 rows in set (0.00 sec)

8.

```
mysql> UPDATE Orders
      -> SET TOTALAMOUNT = (
      ->     SELECT SUM(od.QUANTITY * p.PRICE)
      ->     FROM OrderDetails od
      ->     JOIN Products p ON od.PRODUCTID = p.PRODUCTID
      ->     WHERE od.ORDERID = Orders.ORDERID
      -> )
      -> WHERE ORDERID IN (SELECT ORDERID FROM OrderDetails);
Query OK, 10 rows affected (0.01 sec)
Rows matched: 10  Changed: 10  Warnings: 0
```

```
mysql> select*from orders;
```

ORDERID	CUSTOMERID	ORDERDATE	TOTALAMOUNT
11	1	2024-01-11	2199.978
12	2	2024-01-12	549.45
13	3	2024-01-13	395.956
14	4	2024-01-14	197.83500000000004
15	5	2024-01-15	219.94500000000002
16	6	2024-01-16	141.9
17	7	2024-01-17	113.85000000000001
18	8	2024-01-18	659.9780000000001
19	9	2024-01-19	109.78
20	10	2024-01-20	87.989
21	11	2024-01-21	100

11 rows in set (0.00 sec)

9.

```

mysql> DELIMITER @@
mysql> CREATE PROCEDURE del1(IN id int)
-> BEGIN
-> DELETE FROM OrderDetails WHERE OrderID IN (SELECT OrderID FROM Orders WHERE CustomerID=id);
-> DELETE FROM Orders WHERE CustomerID=id;
-> END @@
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql> CALL del1(3);
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM Orders;
+-----+-----+-----+-----+
| OrderID | CustomerID | OrderDate | TotalAmount |
+-----+-----+-----+-----+
| 1 | 1 | 2024-01-15 | 3080 |
| 4 | 5 | 2024-01-18 | 660 |
| 5 | 4 | 2024-01-19 | 880 |
| 6 | 1 | 2024-01-20 | 4950 |
| 7 | 6 | 2024-01-21 | 1320 |
| 8 | 7 | 2024-01-22 | 440 |
| 9 | 9 | 2024-01-23 | 700 |
| 10 | 8 | 2024-01-24 | 220 |
| 15 | 9 | 2023-05-25 | 440 |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> SELECT * FROM OrderDetails;
+-----+-----+-----+-----+
| OrderDetailID | OrderID | ProductID | Quantity |
+-----+-----+-----+-----+
| 1 | 1 | 2 | 3 |
| 2 | 1 | 5 | 2 |
| 5 | 4 | 4 | 2 |
| 6 | 5 | 2 | 1 |
| 7 | 6 | 6 | 3 |
| 8 | 7 | 8 | 2 |
| 9 | 8 | 10 | 1 |
| 10 | 10 | 7 | 4 |
| 11 | 15 | 5 | 2 |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)

```

10.

```

mysql> select*from products;
+-----+-----+-----+-----+
| PRODUCTID | PRODUCTNAME | DESCRIPTION | PRICE |
+-----+-----+-----+-----+
| 1 | Laptop | High-performance laptop with SSD | 1099.989 |
| 2 | Smartphone | 6.5-inch display, 128GB storage | 549.45 |
| 3 | Headphones | Over-ear headphones with noise cancellation | 98.989 |
| 4 | Coffee Maker | Programmable coffee maker with timer | 65.94500000000001 |
| 5 | Backpack | Durable backpack for laptops and tablets | 43.98900000000004 |
| 6 | Smartwatch | Fitness tracker with heart rate monitor | 141.9 |
| 7 | Bluetooth Speaker | Portable speaker with built-in microphone | 37.95 |
| 8 | Digital Camera | 20MP digital camera with 4K video recording | 329.98900000000003 |
| 9 | Wireless Mouse | Ergonomic wireless mouse with customizable buttons | 27.445 |
| 10 | External Hard Drive | 2TB external hard drive for data storage | 87.989 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql> INSERT INTO Products (PRODUCTID, PRODUCTNAME, DESCRIPTION, PRICE, CATEGORY)
-> VALUES (11, 'New Gadget', 'Description of the new gadget', 149.99, 'Electronics');
ERROR 1054 (42S22): Unknown column 'CATEGORY' in 'field list'
mysql> Insert into products values(11,'pen-drive','128GB',399);
Query OK, 1 row affected (0.00 sec)

mysql> select*from products;
+-----+-----+-----+-----+
| PRODUCTID | PRODUCTNAME | DESCRIPTION | PRICE |
+-----+-----+-----+-----+
| 1 | Laptop | High-performance laptop with SSD | 1099.989 |
| 2 | Smartphone | 6.5-inch display, 128GB storage | 549.45 |
| 3 | Headphones | Over-ear headphones with noise cancellation | 98.989 |
| 4 | Coffee Maker | Programmable coffee maker with timer | 65.94500000000001 |
| 5 | Backpack | Durable backpack for laptops and tablets | 43.98900000000004 |
| 6 | Smartwatch | Fitness tracker with heart rate monitor | 141.9 |
| 7 | Bluetooth Speaker | Portable speaker with built-in microphone | 37.95 |
| 8 | Digital Camera | 20MP digital camera with 4K video recording | 329.98900000000003 |
| 9 | Wireless Mouse | Ergonomic wireless mouse with customizable buttons | 27.445 |
| 10 | External Hard Drive | 2TB external hard drive for data storage | 87.989 |
| 11 | pen-drive | 128GB | 399 |
+-----+-----+-----+-----+
11 rows in set (0.00 sec)

```

11.

```
mysql> Delimiter @@
mysql> CREATE PROCEDURE C(IN O int, s text)
      -> BEGIN
      -> UPDATE ORDERS SET STATUS = s WHERE ORDERID=O;
      -> END @@
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql> CALL C(12,'SHIPPED');
Query OK, 1 row affected (0.00 sec)

mysql> SELECT * FROM ORDERS;
+-----+-----+-----+-----+-----+
| ORDERID | CUSTOMERID | ORDERDATE | TOTALAMOUNT | STATUS |
+-----+-----+-----+-----+-----+
| 11 | 1 | 2024-01-11 | 2199.978 | Shipped |
| 12 | 2 | 2024-01-12 | 549.45 | SHIPPED |
| 13 | 3 | 2024-01-13 | 395.956 | NULL |
| 14 | 4 | 2024-01-14 | 197.83500000000004 | NULL |
| 15 | 5 | 2024-01-15 | 219.94500000000002 | NULL |
| 16 | 6 | 2024-01-16 | 141.9 | NULL |
| 17 | 7 | 2024-01-17 | 113.85000000000001 | NULL |
| 18 | 8 | 2024-01-18 | 659.9780000000001 | NULL |
| 19 | 9 | 2024-01-19 | 109.78 | NULL |
| 20 | 10 | 2024-01-20 | 87.989 | NULL |
| 21 | 11 | 2024-01-21 | 100 | NULL |
+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)
```

12.

ALTER TABLE customer

-> ADD COLUMN OrdersPlaced INT;

```
mysql> UPDATE customer
      -> SET OrdersPlaced =
      ->     (SELECT COUNT(*)
      ->     FROM Orders
      ->     WHERE Orders.CustomerID = customer.CUSTOMERID
      -> );
Query OK, 11 rows affected (0.00 sec)
Rows matched: 11  Changed: 11  Warnings: 0

mysql> select*from customer;
+-----+-----+-----+-----+-----+-----+-----+-----+
| CUSTOMERID | FIRSTNAME | LASTNAME | EMAIL | PHONE | ADDRESS | OrdersPlaced |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Abhi | Kumar | abhi@gmail.com | 100 | ABC | 1 |
| 2 | Abhay | Singh | abhay@gmail.com | 101 | Bhopal | 1 |
| 3 | Abhishek | Sharma | abhishek@gmail.com | 102 | chennai | 1 |
| 4 | Bebo | Kumari | bebo@gmail.com | 103 | delhi | 1 |
| 5 | Chris | Gayle | chris@gmail.com | 104 | jamaica | 1 |
| 6 | David | jones | david@gmail.com | 105 | ahmedabad | 1 |
| 7 | Falguni | Kumari | falguni@gmail.com | 106 | surat | 1 |
| 8 | hari | om | hari@gmail.com | 107 | ranchi | 1 |
| 9 | shivam | singh | shivam@gmail.com | 108 | hyderabad | 1 |
| 10 | harsh | raj | harsh@gmail.com | 109 | mumbai | 1 |
| 11 | mani | shankar | mani@gmail.com | 300 | bokaro | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

TASK 3:

1. Write an SQL query to retrieve a list of all orders along with customer information (e.g., customer name) for each order.
2. Write an SQL query to find the total revenue generated by each electronic gadget product. Include the product name and the total revenue.
3. Write an SQL query to list all customers who have made at least one purchase. Include their names and contact information.
4. Write an SQL query to find the most popular electronic gadget, which is the one with the highest total quantity ordered. Include the product name and the total quantity ordered.
5. Write an SQL query to retrieve a list of electronic gadgets along with their corresponding categories.
6. Write an SQL query to calculate the average order value for each customer. Include the customer's name and their average order value.
7. Write an SQL query to find the order with the highest total revenue. Include the order ID, customer information, and the total revenue.
8. Write an SQL query to list electronic gadgets and the number of times each product has been ordered.
9. Write an SQL query to find customers who have purchased a specific electronic gadget product. Allow users to input the product name as a parameter.
10. Write an SQL query to calculate the total revenue generated by all orders placed within a specific time period. Allow users to input the start and end dates as parameters.

1.

```
mysql> SELECT
->     Orders.ORDERID,
->     Orders.CUSTOMERID,
->     customer.FIRSTNAME AS CustomerFirstName,
->     customer.LASTNAME AS CustomerLastName,
->     customer.EMAIL AS CustomerEmail,
->     customer.PHONE AS CustomerPhone,
->     customer.ADDRESS AS CustomerAddress,
->     Orders.ORDERDATE,
->     Orders.TOTALAMOUNT
-> FROM Orders
-> JOIN customer ON Orders.CUSTOMERID = customer.CUSTOMERID;
```

ORDERID	CUSTOMERID	CustomerFirstName	CustomerLastName	CustomerEmail	CustomerPhone	CustomerAddress	ORDERDATE	TOTALAMOUNT
11	1	Abhi	Kumar	abhi@gmail.com	100	ABC	2024-01-11	2199.978
12	2	Abhay	Singh	abhay@gmail.com	101	Bhopal	2024-01-12	549.45
13	3	Abhishek	Sharma	abhishek@gmail.com	102	chennai	2024-01-13	395.956
14	4	Bebo	Kumari	bebo@gmail.com	103	delhi	2024-01-14	197.83500000000004
15	5	Chris	Gayle	chris@gmail.com	104	jamaica	2024-01-15	219.94500000000002
16	6	David	jones	david@gmail.com	105	ahmedabad	2024-01-16	141.9
17	7	Falguni	Kumari	falguni@gmail.com	106	surat	2024-01-17	113.85000000000001
18	8	hari	om	hari@gmail.com	107	ranchi	2024-01-18	659.97800000000001
19	9	shivam	singh	shivam@gmail.com	108	hyderabad	2024-01-19	109.78
20	10	harsh	raj	harsh@gmail.com	109	mumbai	2024-01-20	87.989
21	11	mani	shankar	mani@gmail.com	300	bokaro	2024-01-21	100

2.

```
mysql> SELECT P.ProductName, SUM(P.Price*O.Quantity) FROM
-> Products P JOIN OrderDetails O
-> ON P.ProductId=O.ProductId
-> GROUP BY P.ProductID
-> ORDER BY P.ProductID;
```

ProductName	SUM(P.Price*O.Quantity)
Laptop	5280
Smartphone	3520
Headphones	165
Tablet	660
Smartwatch	880
Desktop PC	4950
Bluetooth Speaker	220
Camera	1320
Gaming Console	440

9 rows in set (0.00 sec)

3.

```
mysql> SELECT
    ->     c.CUSTOMERID,
    ->     c.FIRSTNAME,
    ->     c.LASTNAME,
    ->     c.EMAIL,
    ->     c.PHONE,
    ->     c.ADDRESS,
    ->     c.OrdersPlaced
    -> FROM customer c
    -> WHERE c.OrdersPlaced >= 1;
+-----+-----+-----+-----+-----+-----+-----+
| CUSTOMERID | FIRSTNAME | LASTNAME | EMAIL      | PHONE | ADDRESS | OrdersPlaced |
+-----+-----+-----+-----+-----+-----+-----+
|      1 | Abhi     | Kumar    | abhi@gmail.com | 100  | ABC     | 1          |
|      2 | Abhay    | Singh    | abhay@gmail.com | 101  | Bhopal  | 1          |
|      3 | Abhishek | Sharma   | abhishek@gmail.com | 102  | chennai | 1          |
|      4 | Bebo     | Kumari   | bebo@gmail.com  | 103  | delhi   | 1          |
|      5 | Chris    | Gayle    | chris@gmail.com | 104  | jamaica | 1          |
|      6 | David    | jones    | david@gmail.com | 105  | ahmedabad | 1          |
|      7 | Falguni  | Kumari   | falguni@gmail.com | 106  | surat   | 1          |
|      8 | hari     | om       | hari@gmail.com  | 107  | ranchi  | 1          |
|      9 | shivam   | singh    | shivam@gmail.com | 108  | hyderabad | 1          |
|     10 | harsh    | raj      | harsh@gmail.com | 109  | mumbai  | 1          |
|     11 | mani     | shankar  | mani@gmail.com  | 300  | bokaro  | 1          |
+-----+-----+-----+-----+-----+-----+-----+
```

4.

```
mysql> SELECT P.ProductName, SUM(O.Quantity) AS TotalQuantity
    -> FROM Products P
    -> JOIN OrderDetails O ON P.ProductID = O.ProductID
    -> GROUP BY P.ProductID
    -> HAVING SUM(O.Quantity) = (
    ->     SELECT MAX(TotalQuantity) FROM (
    ->         SELECT SUM(Quantity) AS TotalQuantity
    ->         FROM OrderDetails
    ->         GROUP BY ProductID
    ->     ) AS SubQuery
    -> );
+-----+-----+
| ProductName | TotalQuantity |
+-----+-----+
| Smartphone | 4           |
| Smartwatch | 4           |
| Laptop     | 4           |
| Bluetooth Speaker | 4           |
+-----+-----+
4 rows in set (0.00 sec)
```

5.

```
mysql> SELECT
    ->     PRODUCTNAME,
    ->     DESCRIPTION
    -> FROM
    ->     products;
+-----+-----+
| PRODUCTNAME | DESCRIPTION
+-----+-----+
| Laptop      | High-performance laptop with SSD
| Smartphone  | 6.5-inch display, 128GB storage
| Headphones   | Over-ear headphones with noise cancellation
| Coffee Maker| Programmable coffee maker with timer
| Backpack     | Durable backpack for laptops and tablets
| Smartwatch   | Fitness tracker with heart rate monitor
| Bluetooth Speaker | Portable speaker with built-in microphone
| Digital Camera | 20MP digital camera with 4K video recording
| Wireless Mouse | Ergonomic wireless mouse with customizable buttons
| External Hard Drive | 2TB external hard drive for data storage
| pen-drive    | 128GB
+-----+-----+
11 rows in set (0.00 sec)
```

6.

```
mysql> SELECT
    ->     c.CUSTOMERID,
    ->     c.FIRSTNAME,
    ->     c.LASTNAME,
    ->     AVG(o.TOTALAMOUNT) AS AverageOrderValue
    -> FROM
    ->     customer c
    -> JOIN
    ->     orders o ON c.CUSTOMERID = o.CUSTOMERID
    -> GROUP BY
    ->     c.CUSTOMERID, c.FIRSTNAME, c.LASTNAME;
+-----+-----+-----+-----+
| CUSTOMERID | FIRSTNAME | LASTNAME | AverageOrderValue |
+-----+-----+-----+-----+
| 1          | Abhi       | Kumar    | 2199.978           |
| 2          | Abhay      | Singh    | 549.45             |
| 3          | Abhishek   | Sharma   | 395.956            |
| 4          | Bebo       | Kumari   | 197.83500000000004 |
| 5          | Chris      | Gayle    | 219.94500000000002 |
| 6          | David      | jones    | 141.9              |
| 7          | Falguni    | Kumari   | 113.85000000000001 |
| 8          | hari       | om       | 659.97800000000001 |
| 9          | shivam     | singh    | 109.78              |
| 10         | harsh      | raj      | 87.989              |
| 11         | mani       | shankar  | 100                 |
+-----+-----+-----+-----+
```

7.

```
mysql> SELECT
->     o.ORDERID,
->     o.CUSTOMERID,
->     c.FIRSTNAME,
->     c.LASTNAME,
->     c.EMAIL,
->     c.PHONE,
->     c.ADDRESS,
->     o.TOTALAMOUNT AS TotalRevenue
-> FROM
->     orders o
-> JOIN
->     customer c ON o.CUSTOMERID = c.CUSTOMERID
-> ORDER BY
->     TotalRevenue DESC
-> LIMIT 1;
+-----+-----+-----+-----+-----+-----+-----+
| ORDERID | CUSTOMERID | FIRSTNAME | LASTNAME | EMAIL      | PHONE | ADDRESS | TotalRevenue |
+-----+-----+-----+-----+-----+-----+-----+
|      11 |          1 | Abhi     | Kumar    | abhi@gmail.com |   100 | ABC     |      2199.978 |
+-----+-----+-----+-----+-----+-----+-----+
```

8.

```
mysql> SELECT
->     p.PRODUCTID,
->     p.PRODUCTNAME,
->     COUNT(od.ORDERDETAILID) AS OrderCount
-> FROM
->     products p
-> LEFT JOIN
->     orderdetails od ON p.PRODUCTID = od.PRODUCTID
-> GROUP BY
->     p.PRODUCTID, p.PRODUCTNAME;
+-----+-----+-----+
| PRODUCTID | PRODUCTNAME | OrderCount |
+-----+-----+-----+
|      1 | Laptop       |          1 |
|      2 | Smartphone   |          1 |
|      3 | Headphones   |          1 |
|      4 | Coffee Maker |          1 |
|      5 | Backpack     |          1 |
|      6 | Smartwatch   |          1 |
|      7 | Bluetooth Speaker |          1 |
|      8 | Digital Camera |          1 |
|      9 | Wireless Mouse |          1 |
|     10 | External Hard Drive |          1 |
|     11 | pen-drive    |          0 |
+-----+-----+-----+
```

9.

```
mysql> CALL cust('Smartphone');
```

CustomerID	FirstName	LastName	email	Phone	Address
1	John	Doe	john.doe@example.com	1234567890	123 Main St
4	Bob	Williams	bob.williams@example.com	9998887777	101 Elm St

```
2 rows in set (0.00 sec)
```

10.

```
mysql> DELIMITER //
mysql>
mysql> CREATE PROCEDURE CalculateTotalRevenue(IN start_date DATE, IN end_date DATE)
      -> BEGIN
      ->     SELECT SUM(od.QUANTITY * p.PRICE) AS TotalRevenue
      ->     FROM orders o
      ->     JOIN orderdetails od ON o.ORDERID = od.ORDERID
      ->     JOIN products p ON od.PRODUCTID = p.PRODUCTID
      ->     WHERE o.ORDERDATE BETWEEN start_date AND end_date;
      -> END //
Query OK, 0 rows affected (0.03 sec)

mysql>
mysql> DELIMITER ;
mysql> CALL CalculateTotalRevenue('2024-01-11', '2024-01-21');
+-----+
| TotalRevenue          |
+-----+
| 4676.660999999999    |
+-----+
1 row in set (0.02 sec)

Query OK, 0 rows affected (0.04 sec)
```

TASK 4:

Subquery and its type:

1. Write an SQL query to find out which customers have not placed any orders.
2. Write an SQL query to find the total number of products available for sale.
3. Write an SQL query to calculate the total revenue generated by TechShop.
4. Write an SQL query to calculate the average quantity ordered for products in a specific category. Allow users to input the category name as a parameter.
5. Write an SQL query to calculate the total revenue generated by a specific customer. Allow users to input the customer ID as a parameter.
6. Write an SQL query to find the customers who have placed the most orders. List their names and the number of orders they've placed.
7. Write an SQL query to find the most popular product category, which is the one with the highest total quantity ordered across all orders.
8. Write an SQL query to find the customer who has spent the most money (highest total revenue) on electronic gadgets. List their name and total spending.
9. Write an SQL query to calculate the average order value (total revenue divided by the number of orders) for all customers.

1.

```
mysql> SELECT Customer.CUSTOMERID, Customer.FIRSTNAME, Customer.LASTNAME
   -> FROM Customer
   -> LEFT JOIN Orders ON Customer.CUSTOMERID = Orders.CUSTOMERID
   -> WHERE Orders.CUSTOMERID IS NULL;
Empty set (0.00 sec)
```

2.

```
mysql> SELECT COUNT(*) AS TotalProducts
   -> FROM Product;
ERROR 1146 (42S02): Table 'techshop.product' doesn't exist
mysql> ^C
mysql> SELECT COUNT(*) AS TotalProducts
   -> FROM products;
+-----+
| TotalProducts |
+-----+
|          11 |
+-----+
1 row in set (0.03 sec)
```

3.

```
mysql> SELECT SUM(Orders.TOTALAMOUNT) AS TotalRevenue
-> FROM Orders;
+-----+
| TotalRevenue |
+-----+
| 4776.660999999999 |
+-----+
1 row in set (0.00 sec)
```

4.

```
mysql> SELECT
->     AVG(OD.Quantity) AS AverageQuantityOrdered
->     FROM
->         OrderDetails AS OD
->     JOIN
->         Products AS P ON OD.ProductID = P.ProductID
->     WHERE
->         P.Category = 'Hearing Device';
+-----+
| AverageQuantityOrdered |
+-----+
| 2.6667 |
+-----+
1 row in set (0.00 sec)

mysql>
```

5.

```
mysql> DELIMITER //
mysql>
mysql> CREATE PROCEDURE CalculateCustomerRevenue(IN customer_id INT)
-> BEGIN
->     SELECT SUM(od.QUANTITY * p.PRICE) AS TotalCustomerRevenue
->     FROM orders o
->     JOIN orderdetails od ON o.ORDERID = od.ORDERID
->     JOIN products p ON od.PRODUCTID = p.PRODUCTID
->     WHERE o.CUSTOMERID = customer_id;
-> END //
Query OK, 0 rows affected (0.02 sec)

mysql>
mysql> DELIMITER ;
mysql> CALL CalculateCustomerRevenue(1);
+-----+
| TotalCustomerRevenue |
+-----+
| 2199.978 |
+-----+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.03 sec)
```

6.

```
mysql> SELECT
    ->      c.FIRSTNAME,
    ->      c.LASTNAME,
    ->      COUNT(o.ORDERID) AS OrdersPlaced
    -> FROM
    ->      customer c
    -> JOIN
    ->      orders o ON c.CUSTOMERID = o.CUSTOMERID
    -> GROUP BY
    ->      c.CUSTOMERID
    -> ORDER BY
    ->      OrdersPlaced DESC
    -> LIMIT 1;
+-----+-----+-----+
| FIRSTNAME | LASTNAME | OrdersPlaced |
+-----+-----+-----+
| Abhi     | Kumar    |          1 |
+-----+-----+-----+
1 row in set (0.01 sec)
```

7.

```
mysql> SELECT
    ->      P.Category,
    ->      SUM(OD.Quantity) AS TotalQuantityOrdered
    -> FROM
    ->      OrderDetails AS OD
    -> JOIN
    ->      Products AS P ON OD.ProductID = P.ProductID
    -> GROUP BY
    ->      P.Category
    -> ORDER BY
    ->      TotalQuantityOrdered DESC
    -> LIMIT 1;
+-----+-----+
| Category | TotalQuantityOrdered |
+-----+-----+
| Hearing Device |          8 |
+-----+-----+
1 row in set (0.01 sec)
```

8.

```
mysql> SELECT
    ->     C.CustomerID,
    ->     C.FirstName,
    ->     C.LastName,
    ->     SUM(OD.Quantity * P.Price) AS TotalSpending
    -> FROM
    ->     Customers AS C
    -> JOIN
    ->     Orders AS O ON C.CustomerID = O.CustomerID
    -> JOIN
    ->     OrderDetails AS OD ON O.OrderID = OD.OrderID
    -> JOIN
    ->     Products AS P ON OD.ProductID = P.ProductID
    -> WHERE
    ->     P.Category = 'Hearing Device'
    -> ORDER BY
    ->     TotalSpending DESC
    -> LIMIT 1;
+-----+-----+-----+-----+
| CustomerID | FirstName | LastName | TotalSpending |
+-----+-----+-----+-----+
|          2 | sumit     | thakur   |      32048.42 |
+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

9.

```
mysql> SELECT
    ->     C.CustomerID,
    ->     C.FirstName,
    ->     C.LastName,
    ->     COUNT(O.OrderID) AS NumberOfOrders,
    ->     SUM(O.TotalAmount) AS TotalRevenue,
    ->     AVG(O.TotalAmount) AS AverageOrderValue
    -> FROM
    ->     Customers AS C
    -> JOIN
    ->     Orders AS O ON C.CustomerID = O.CustomerID
    -> ORDER BY
    ->     AverageOrderValue DESC;
+-----+-----+-----+-----+-----+-----+
| CustomerID | FirstName | LastName | NumberOfOrders | TotalRevenue | AverageOrderValue |
+-----+-----+-----+-----+-----+-----+
|          2 | sumit     | thakur   |             9 |    223379.24 |      24819.915556 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)

mysql> _
```