

# ASSIGNMENT 1

NAME : - SUSHANT KUMAR SINGH

EMAIL :- [ssushant886@gmail.com](mailto:ssushant886@gmail.com)

## Task:1. Database Design:

1. Create the database named "TechShop"

```
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 31
Server version: 8.0.30 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

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

mysql> USE TECHSHOP;
```

2. Define the schema for the Customers, Products, Orders, OrderDetails and Inventory tables based on the provided schema.

a) Customers

```
mysql> USE TECHSHOP;
Database changed
mysql> CREATE TABLE Customers (
  ->   CustomerID INT PRIMARY KEY,
  ->   FirstName VARCHAR(25),
  ->   LastName VARCHAR(25),
  ->   Email VARCHAR(25),
  ->   Phone VARCHAR(20),
  ->   Address VARCHAR(250)
  -> );
Query OK, 0 rows affected (0.04 sec)

mysql> DESC CUSTOMERS;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CustomerID | int | NO | PRI | NULL | |
| FirstName | varchar(25) | YES | | NULL | |
| LastName | varchar(25) | YES | | NULL | |
| Email | varchar(25) | YES | | NULL | |
| Phone | varchar(20) | YES | | NULL | |
| Address | varchar(250) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

## b) PRODUCTS

```
mysql> CREATE TABLE Products (  
->     ProductID INT PRIMARY KEY,  
->     ProductName VARCHAR(25),  
->     Description TEXT,  
->     Price DECIMAL(10, 2)  
-> );
```

Query OK, 0 rows affected (0.04 sec)

```
mysql> DESC PRODUCTS;
```

Field	Type	Null	Key	Default	Extra
ProductID	int	NO	PRI	NULL	
ProductName	varchar(25)	YES		NULL	
Description	text	YES		NULL	
Price	decimal(10,2)	YES		NULL	

4 rows in set (0.00 sec)

## c) ORDERS

```
mysql> CREATE TABLE Orders (  
->     OrderID INT PRIMARY KEY,  
->     CustomerID INT,  
->     OrderDate DATE,  
->     TotalAmount DECIMAL(10, 2),  
->     FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)  
-> );
```

Query OK, 0 rows affected (0.04 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	decimal(10,2)	YES		NULL	

4 rows in set (0.00 sec)

## d) ORDERDETAILS

```
mysql> CREATE TABLE OrderDetails (  
->     OrderDetailID INT PRIMARY KEY,  
->     OrderID INT,  
->     ProductID INT,  
->     Quantity INT,  
->     FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),  
->     FOREIGN KEY (ProductID) REFERENCES Products(ProductID)  
-> );
```

Query OK, 0 rows affected (0.04 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	

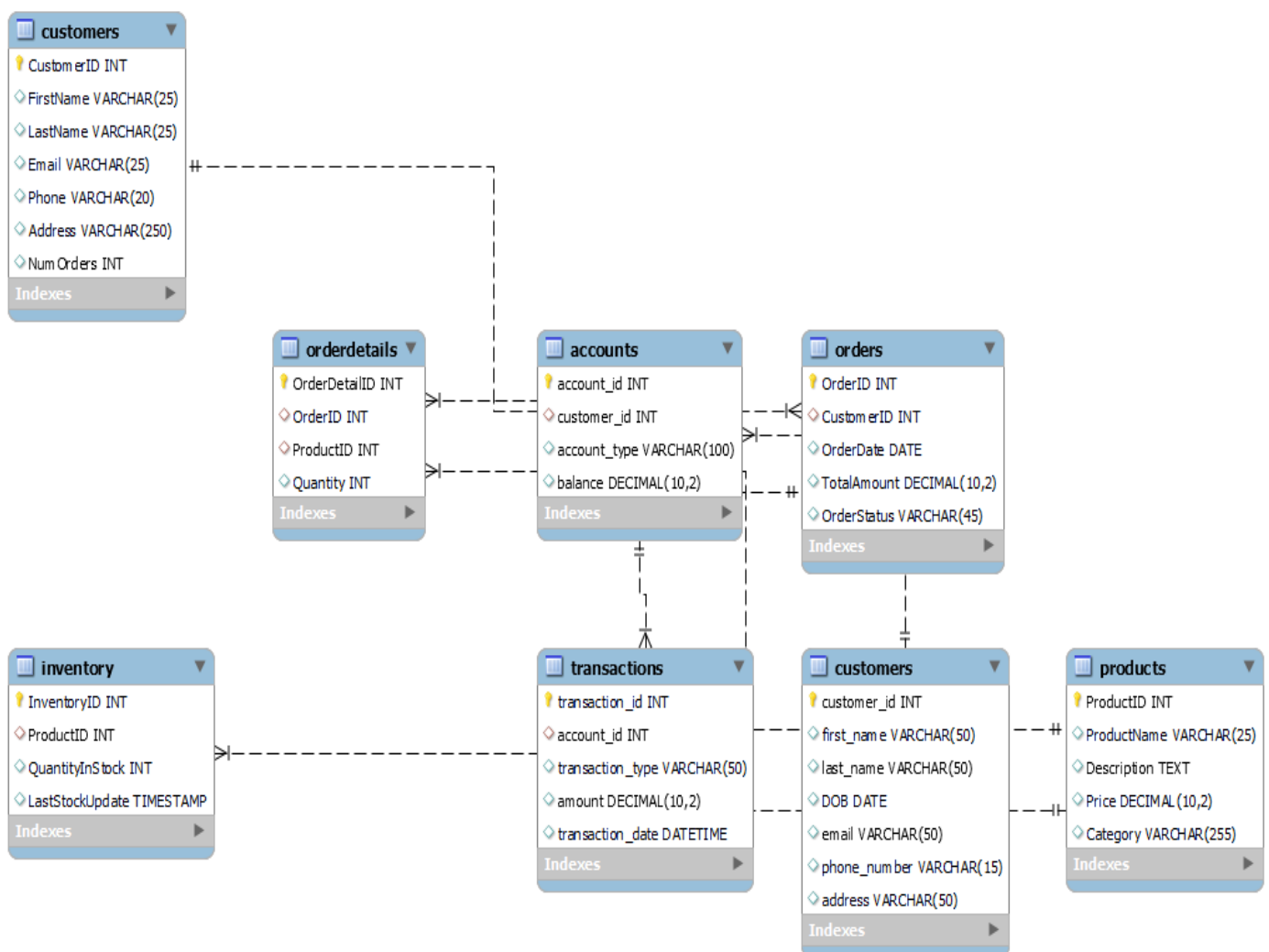
## e) INVENTORY

```
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version f
FOREIGN KEY (ProductID) REFERENCES Products(ProductID)
)' at line 5
mysql> CREATE TABLE Inventory (
->   InventoryID INT PRIMARY KEY,
->   ProductID INT,
->   QuantityInStock INT,
->   LastStockUpdate TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
->   FOREIGN KEY (ProductID) REFERENCES Products(ProductID)
-> );
Query OK, 0 rows affected (0.04 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 | timestamp | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql>
```

## 3. Create an ERD (Entity Relationship Diagram) for the database.



#### 4. Create appropriate Primary Key and Foreign Key constraints for referential integrity.

```
CREATE TABLE Orders (  
    OrderID INT PRIMARY KEY,  
    CustomerID INT,  
    OrderDate DATE,  
    TotalAmount DECIMAL(10, 2),  
    FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)  
);
```

#### 5. Insert at least 10 sample records into each of the following tables. a. Customers b. Products c. Orders d. OrderDetails

##### a) Customers

```
mysql> INSERT INTO Customers (CustomerID, FirstName, LastName, Email, Phone, Address)  
-> VALUES  
-> (1, 'SINGH', 'SUSHANT', 'ssushant@YAHOO.com', '123-456-7890', 'salt Main St'),  
-> (2, 'sumit', 'thakur', 'smith@gamil.com', '987-654-3210', '456 north St'),  
-> (3, 'Aman', 'Javed', 'aj@gamil.com', '555-123-4567', '789 south St'),  
-> (4, 'raju', 'singh', 'w@hotmail.com', '333-999-8888', '567 office road gve'),  
-> (5, 'alisha', 'ron', 'a.b@bing.com', '111-222-3333', '890 northoffice Blvd'),  
-> (6, 'Mohan', 'ram', 'miohan@gmail.com', '777-888-9999', '345 docter south Dr'),  
-> (7, 'Soni', 'aditi', 'sa@gmail.com', '444-666-7777', '678 mapplplaza '),  
-> (8, 'Darick', 'junaid', 'dj@gmail.com', '222-444-5555', '901 lasvagas Ln'),  
-> (9, 'sushant', 'White', 's.w@gmail.com', '888-111-2222', '234 chennai'),  
-> (10, 'bharat', 'kumar', 'bk@gmail.com', '666-333-4444', '789 Pine St');
```

Query OK, 10 rows affected (0.01 sec)

Records: 10 Duplicates: 0 Warnings: 0

```
mysql> select * from customers;
```

CustomerID	FirstName	LastName	Email	Phone	Address
1	SINGH	SUSHANT	ssushant@YAHOO.com	123-456-7890	salt Main St
2	sumit	thakur	smith@gamil.com	987-654-3210	456 north St
3	Aman	Javed	aj@gamil.com	555-123-4567	789 south St
4	raju	singh	w@hotmail.com	333-999-8888	567 office road gve
5	alisha	ron	a.b@bing.com	111-222-3333	890 northoffice Blvd
6	Mohan	ram	miohan@gmail.com	777-888-9999	345 docter south Dr
7	Soni	aditi	sa@gmail.com	444-666-7777	678 mapplplaza
8	Darick	junaid	dj@gmail.com	222-444-5555	901 lasvagas Ln
9	sushant	White	s.w@gmail.com	888-111-2222	234 chennai
10	bharat	kumar	bk@gmail.com	666-333-4444	789 Pine St

10 rows in set (0.00 sec)

## b. Products

```
mysql> INSERT INTO Products (ProductID, ProductName, Description, Price)
-> VALUES
-> (1, 'HeadPhone 921', 'Black 60Hrs ', 519.99),
-> (2, 'Trimer', 'Red 30Hrs titanium blade', 5529.99),
-> (3, 'AirDrops', 'white 90Hrs Fast', 5514.99),
-> (4, 'TV', '30 inch full HD 4K', 6539.99),
-> (5, 'LAPTOP', ' i9 8GB RAM 500GB SSD TI106i', 559.99),
-> (6, 'Charger', 'multipins 50hrz', 1149.99),
-> (7, 'ProG watch', 'Glod 5.6 inch fast 80Hrs', 5524.99),
-> (8, 'iPad', 'black 129GB 5000mAH', 5534.99),
-> (9, 'Screen touch laptop', ' i5 brown 90GB fast', 5559.99),
-> (10, 'Speaker', '4X sound', 6654.99);
```

Query OK, 10 rows affected (0.01 sec)

Records: 10 Duplicates: 0 Warnings: 0

```
mysql> select * from products;
```

ProductID	ProductName	Description	Price
1	HeadPhone 921	Black 60Hrs	519.99
2	Trimer	Red 30Hrs titanium blade	5529.99
3	AirDrops	white 90Hrs Fast	5514.99
4	TV	30 inch full HD 4K	6539.99
5	LAPTOP	i9 8GB RAM 500GB SSD TI106i	559.99
6	Charger	multipins 50hrz	1149.99
7	ProG watch	Glod 5.6 inch fast 80Hrs	5524.99
8	iPad	black 129GB 5000mAH	5534.99
9	Screen touch laptop	i5 brown 90GB fast	5559.99
10	Speaker	4X sound	6654.99

10 rows in set (0.00 sec)

## c) Orders

```
mysql> INSERT INTO Orders (OrderID, CustomerID, OrderDate, TotalAmount)
-> VALUES
-> (1, 1, '2024-01-12', 2249.98),
-> (2, 2, '2024-01-13', 5559.99),
-> (3, 3, '2024-01-14', 6529.97),
-> (4, 4, '2024-01-15', 66579.98),
-> (5, 5, '2024-01-16', 25819.99),
-> (6, 6, '2024-01-17', 45899.95),
-> (7, 7, '2024-01-18', 65844.97),
-> (8, 8, '2024-01-19', 25434.99),
-> (9, 9, '2024-01-20', 1239.98),
-> (10, 10, '2024-01-21', 35689.99);
```

Query OK, 10 rows affected (0.01 sec)

Records: 10 Duplicates: 0 Warnings: 0

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

OrderID	CustomerID	OrderDate	TotalAmount
1	1	2024-01-12	2249.98
2	2	2024-01-13	5559.99
3	3	2024-01-14	6529.97
4	4	2024-01-15	66579.98
5	5	2024-01-16	25819.99
6	6	2024-01-17	45899.95
7	7	2024-01-18	65844.97
8	8	2024-01-19	25434.99
9	9	2024-01-20	1239.98
10	10	2024-01-21	35689.99

10 rows in set (0.00 sec)

#### d. OrderDetails

```
mysql> INSERT INTO OrderDetails (OrderDetailID, OrderID, ProductID, Quantity)
-> VALUES
-> (1, 1, 1, 2),
-> (2, 1, 2, 1),
-> (3, 2, 1, 3),
-> (4, 2, 3, 1),
-> (5, 3, 2, 2),
-> (6, 3, 4, 1),
-> (7, 4, 3, 4),
-> (8, 4, 5, 2),
-> (9, 5, 4, 1),
-> (10, 5, 6, 3);
```

Query OK, 10 rows affected (0.01 sec)  
Records: 10 Duplicates: 0 Warnings: 0

```
mysql> select * from orderdetails;
```

OrderDetailID	OrderID	ProductID	Quantity
1	1	1	2
2	1	2	1
3	2	1	3
4	2	3	1
5	3	2	2
6	3	4	1
7	4	3	4
8	4	5	2
9	5	4	1
10	5	6	3

10 rows in set (0.00 sec)

#### e) INVENTORY

```
mysql> INSERT INTO Inventory (InventoryID, ProductID, QuantityInStock)
-> VALUES
-> (1, 1, 505),
-> (2, 2, 302),
-> (3, 3, 201),
-> (4, 4, 402),
-> (5, 5, 602),
-> (6, 6, 252),
-> (7, 7, 13),
-> (8, 8, 35),
-> (9, 9, 10),
-> (10, 10, 4533);
```

Query OK, 10 rows affected (0.01 sec)  
Records: 10 Duplicates: 0 Warnings: 0

```
mysql> select * from inventory;
```

InventoryID	ProductID	QuantityInStock	LastStockUpdate
1	1	505	2024-01-12 17:30:49
2	2	302	2024-01-12 17:30:49
3	3	201	2024-01-12 17:30:49
4	4	402	2024-01-12 17:30:49
5	5	602	2024-01-12 17:30:49
6	6	252	2024-01-12 17:30:49
7	7	13	2024-01-12 17:30:49
8	8	35	2024-01-12 17:30:49
9	9	10	2024-01-12 17:30:49
10	10	4533	2024-01-12 17:30:49

10 rows in set (0.00 sec)

```
mysql>
```

## Tasks 2: Select, Where, Between, AND, LIKE:

1. Write an SQL query to retrieve the names and emails of all customers.

```
mysql> SELECT FirstName, LastName, Email
-> FROM Customers;
+-----+-----+-----+
| FirstName | LastName | Email |
+-----+-----+-----+
| SINGH     | SUSHANT  | ssushant@YAHOO.com |
| sumit     | thakur   | smith@gamil.com    |
| Aman      | Javed    | aj@gamil.com        |
| raju      | singh    | w@hotmail.com       |
| alisha    | ron      | a.b@bing.com        |
| Mohan     | ram      | miohan@gmail.com    |
| Soni      | aditi    | sa@gmail.com        |
| Darick    | junaaid  | dj@gmail.com        |
| sushant   | White    | s.w@gmail.com       |
| bharat    | kumar    | bk@gmail.com        |
+-----+-----+-----+
10 rows in set (0.00 sec)

mysql> _
```

2) Write an SQL query to list all orders with their order dates and corresponding customer names

```
mysql> SELECT Orders.OrderID, Orders.OrderDate, Customers.FirstName, Customers.LastName
-> FROM Orders
-> JOIN Customers ON Orders.CustomerID = Customers.CustomerID;
+-----+-----+-----+-----+
| OrderID | OrderDate | FirstName | LastName |
+-----+-----+-----+-----+
| 1       | 2024-01-12 | SINGH     | SUSHANT  |
| 2       | 2024-01-13 | sumit     | thakur   |
| 3       | 2024-01-14 | Aman      | Javed    |
| 4       | 2024-01-15 | raju      | singh    |
| 5       | 2024-01-16 | alisha    | ron      |
| 6       | 2024-01-17 | Mohan     | ram      |
| 7       | 2024-01-18 | Soni      | aditi    |
| 8       | 2024-01-19 | Darick    | junaaid  |
| 9       | 2024-01-20 | sushant   | White    |
| 10      | 2024-01-21 | bharat    | kumar    |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)

mysql>
```



3. Write an SQL query to insert a new customer record into the "Customers" table. Include customer information such as name, email, and address.

```
mysql> INSERT INTO Customers (CustomerID, FirstName, LastName, Email, Phone, Address)
-> VALUES (12, 'SINGH', 'SUSHANTS', 'ssushant@gamil.com', '0061089873', '789 INDIA');
Query OK, 1 row affected (0.00 sec)
```

```
mysql> select * from customer;
ERROR 1146 (42S02): Table 'techshop.customer' doesn't exist
mysql> select * from customers;
```

CustomerID	FirstName	LastName	Email	Phone	Address
1	SINGH	SUSHANT	ssushant@YAHOO.com	123-456-7890	salt Main St
2	sumit	thakur	smith@gamil.com	987-654-3210	456 north St
3	Aman	Javed	aj@gamil.com	555-123-4567	789 south St
4	raju	singh	w@hotmail.com	333-999-8888	567 office road gve
5	alisha	ron	a.b@bing.com	111-222-3333	890 northoffice Blvd
6	Mohan	ram	miohan@gmail.com	777-888-9999	345 docter south Dr
7	Soni	aditi	sa@gmail.com	444-666-7777	678 mapplplaza
8	Darick	junaaid	dj@gmail.com	222-444-5555	901 lasvagas Ln
9	sushant	White	s.w@gmail.com	888-111-2222	234 chennai
10	bharat	kumar	bk@gmail.com	666-333-4444	789 Pine St
12	SINGH	SUSHANTS	ssushant@gamil.com	0061089873	789 INDIA

11 rows in set (0.00 sec)

```
mysql>
```

4. Write an SQL query to update the prices of all electronic gadgets in the "Products" table by increasing them by 10%.

```
mysql> UPDATE Products
-> SET Price = Price * 1.1;
Query OK, 10 rows affected, 10 warnings (0.01 sec)
Rows matched: 10 Changed: 10 Warnings: 10
```

```
mysql> select * from products;
```

ProductID	ProductName	Description	Price
1	HeadPhone 921	Black 60Hrs	571.99
2	Trimer	Red 30Hrs titanium blade	6082.99
3	AirDrops	white 90Hrs Fast	6066.49
4	TV	30 inch full HD 4K	7193.99
5	LAPTOP	i9 8GB RAM 500GB SSD TI106i	615.99
6	Charger	multipins 50hrz	1264.99
7	ProG watch	Glod 5.6 inch fast 80Hrs	6077.49
8	iPad	black 129GB 5000mAH	6088.49
9	Screen touch laptop	i5 brown 90GB fast	6115.99
10	Speaker	4X sound	7320.49

10 rows in set (0.00 sec)

```
mysql>
```



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.

```
mysql> DELETE FROM OrderDetails
-> WHERE OrderID = 1;
Query OK, 2 rows affected (0.01 sec)
```

```
mysql> select * from orderdetails;
+-----+-----+-----+-----+
| OrderDetailID | OrderID | ProductID | Quantity |
+-----+-----+-----+-----+
| 3 | 2 | 1 | 3 |
| 4 | 2 | 3 | 1 |
| 5 | 3 | 2 | 2 |
| 6 | 3 | 4 | 1 |
| 7 | 4 | 3 | 4 |
| 8 | 4 | 5 | 2 |
| 9 | 5 | 4 | 1 |
| 10 | 5 | 6 | 3 |
+-----+-----+-----+-----+
8 rows in set (0.00 sec)
```

```
mysql> _
```

```
select * from orders' at line 1
mysql> DELETE FROM Orders
-> WHERE OrderID = 1;
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> select * from orders;
+-----+-----+-----+-----+
| OrderID | CustomerID | OrderDate | TotalAmount |
+-----+-----+-----+-----+
| 2 | 2 | 2024-01-13 | 7782.46 |
| 3 | 3 | 2024-01-14 | 19359.97 |
| 4 | 4 | 2024-01-15 | 25497.94 |
| 5 | 5 | 2024-01-16 | 10988.96 |
| 6 | 6 | 2024-01-17 | 45899.95 |
| 7 | 7 | 2024-01-18 | 65844.97 |
| 8 | 8 | 2024-01-19 | 25434.99 |
| 9 | 9 | 2024-01-20 | 1239.98 |
| 10 | 10 | 2024-01-21 | 35689.99 |
| 11 | 1 | 2024-01-23 | 5000.00 |
+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

```
mysql>
```

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.

```
-> VALUES (11,1,'2024-01-23',5000);
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that cor
mysql> INSERT INTO Orders (OrderID, CustomerID, OrderDate, TotalAmount)
-> VALUES (11, 1, '2024-01-23', 5000);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from orders;
+-----+-----+-----+-----+
| OrderID | CustomerID | OrderDate | TotalAmount |
+-----+-----+-----+-----+
| 1 | 1 | 2024-01-12 | 2249.98 |
| 2 | 2 | 2024-01-13 | 5559.99 |
| 3 | 3 | 2024-01-14 | 6529.97 |
| 4 | 4 | 2024-01-15 | 66579.98 |
| 5 | 5 | 2024-01-16 | 25819.99 |
| 6 | 6 | 2024-01-17 | 45899.95 |
| 7 | 7 | 2024-01-18 | 65844.97 |
| 8 | 8 | 2024-01-19 | 25434.99 |
| 9 | 9 | 2024-01-20 | 1239.98 |
| 10 | 10 | 2024-01-21 | 35689.99 |
| 11 | 1 | 2024-01-23 | 5000.00 |
+-----+-----+-----+-----+
11 rows in set (0.00 sec)
```

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.

```
mysql> call userr(3);
ERROR 1318 (42000): Incorrect number of arguments for PROCEDURE techshop.userr; expected 3, got 1
mysql> delimiter ##
mysql> create procedure display(IN var1 int)
-> begin
-> select * from customers where CustomerID = var1;
-> end ##
Query OK, 0 rows affected (0.01 sec)

mysql> delimiter ;
mysql> call display(1);
+-----+-----+-----+-----+-----+-----+-----+
| CustomerID | FirstName | LastName | Email | Phone | Address | NumOrders |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | SINGH | SUSHANT | ssushant@YAHOO.com | 123-456-7890 | Salt Main St | 1 |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)

Query OK, 0 rows affected (0.01 sec)

mysql> _
```

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

```
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 EXISTS (
-> SELECT 1
-> FROM OrderDetails
-> WHERE OrderID = Orders.OrderID
-> );
Query OK, 5 rows affected (0.02 sec)
Rows matched: 5 Changed: 5 Warnings: 0

mysql> select * from orders;
+-----+-----+-----+-----+
| OrderID | CustomerID | OrderDate | TotalAmount |
+-----+-----+-----+-----+
| 1 | 1 | 2024-01-12 | 7226.97 |
| 2 | 2 | 2024-01-13 | 7782.46 |
| 3 | 3 | 2024-01-14 | 19359.97 |
| 4 | 4 | 2024-01-15 | 25497.94 |
| 5 | 5 | 2024-01-16 | 10988.96 |
| 6 | 6 | 2024-01-17 | 45899.95 |
| 7 | 7 | 2024-01-18 | 65844.97 |
| 8 | 8 | 2024-01-19 | 25434.99 |
| 9 | 9 | 2024-01-20 | 1239.98 |
| 10 | 10 | 2024-01-21 | 35689.99 |
| 11 | 1 | 2024-01-23 | 5000.00 |
+-----+-----+-----+-----+
11 rows in set (0.00 sec)

mysql> _
```

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 as a parameter.

```
mysql> DELIMITER $$
mysql> CREATE PROCEDURE delete_customer(IN cust_id INT)
-> BEGIN
->     DELETE FROM Customers WHERE CustomerID = cust_id;
-> END $$
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> DELIMITER ;
mysql> CALL delete_customer(3);
Query OK, 1 row affected (0.00 sec)
```

```
mysql> select * from customers;
```

CustomerID	FirstName	LastName	Email	Phone	Address	NumOrders
1	SINGH	SUSHANT	ssushant@YAHOO.com	123-456-7890	Salt Main St	1
2	sumit	thakur	smith@gamil.com	987-654-3210	456 north St	20
4	raju	singh	w@hotmail.com	333-999-8888	567 office road gve	1
5	alisha	ron	a.b@bing.com	111-222-3333	890 northoffice Blvd	1
6	Mohan	ram	miohan@gmail.com	777-888-9999	345 docter south Dr	1
7	Soni	aditi	sa@gmail.com	444-666-7777	678 mapplplaza	1
8	Darick	junaid	dj@gmail.com	222-444-5555	901 lasvagas Ln	1
9	sushant	White	rama@gmail.com	888-111-2222	parkstreet	1
10	bharat	kumar	bk@gmail.com	666-333-4444	789 Pine St	1
12	SINGH	SUSHANTS	ssushant@gamil.com	0061089873	789 INDIA	0

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

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.

```
ERROR 1062 (23000): Duplicate entry '2' for key 'products.PRIMARY'
mysql> INSERT INTO Products (ProductID, ProductName, Description, Price)
-> VALUES (11, 'Mouse', 'Wireless, fast connect', 999.09);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from products;
```

ProductID	ProductName	Description	Price
1	HeadPhone 921	Black 60Hrs	571.99
2	Trimer	Red 30Hrs titanium blade	6082.99
3	AirDrops	white 90Hrs Fast	6066.49
4	TV	30 inch full HD 4K	7193.99
5	LAPTOP	i9 8GB RAM 500GB SSD TI106i	615.99
6	Charger	multipins 50hrz	1264.99
7	ProG watch	Glod 5.6 inch fast 80Hrs	6077.49
8	iPad	black 129GB 5000MAH	6088.49
9	Screen touch laptop	i5 brown 90GB fast	6115.99
10	Speaker	4X sound	7320.49
11	Mouse	Wireless, fast connect	999.09

```
11 rows in set (0.01 sec)
```

```
mysql>
```

```
mysql> DELIMITER ##
mysql> CREATE PROCEDURE orstatus(IN order_id INT, IN new_status VARCHAR(50))
-> BEGIN
->     UPDATE Orders SET OrderStatus = new_status WHERE OrderID = order_id;
->     SELECT * FROM Orders;
-> END ##
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql> CALL orstatus(8, 'Shipped');

+-----+-----+-----+-----+-----+
| OrderID | CustomerID | OrderDate | TotalAmount | OrderStatus |
+-----+-----+-----+-----+-----+
| 2 | 2 | 2024-01-13 | 7782.46 | Pending |
| 4 | 4 | 2024-01-15 | 25497.94 | Pending |
| 5 | 5 | 2024-01-16 | 10988.96 | Pending |
| 6 | 6 | 2024-01-17 | 45899.95 | Pending |
| 7 | 7 | 2024-01-18 | 65844.97 | Pending |
| 8 | 8 | 2024-01-19 | 25434.99 | Shipped |
| 9 | 9 | 2024-01-20 | 1239.98 | Pending |
| 10 | 10 | 2024-01-21 | 35689.99 | Shipped |
| 11 | 1 | 2024-01-23 | 5000.00 | Shipped |
+-----+-----+-----+-----+-----+
9 rows in set (0.01 sec)

Query OK, 0 rows affected (0.06 sec)

mysql> _
```

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

mysql> select * from customers;
```

CustomerID	FirstName	LastName	Email	Phone	Address	NumOrders
1	SINGH	SUSHANT	ssushant@YAHOO.com	123-456-7890	salt Main St	1
2	sumit	thakur	smith@gamil.com	987-654-3210	456 north St	1
3	Aman	Javed	aj@gamil.com	555-123-4567	789 south St	0
4	raju	singh	w@hotmail.com	333-999-8888	567 office road gve	1
5	alisha	ron	a.b@bing.com	111-222-3333	890 northoffice Blvd	1
6	Mohan	ram	miohan@gmail.com	777-888-9999	345 docter south Dr	1
7	Soni	aditi	sa@gmail.com	444-666-7777	678 mapplplaza	1
8	Darick	junaaid	dj@gmail.com	222-444-5555	901 lasvagas Ln	1
9	sushant	White	rama@gmail.com	888-111-2222	parkstreet	1
10	bharat	kumar	bk@gmail.com	666-333-4444	789 Pine St	1
12	SINGH	SUSHANTS	ssushant@gamil.com	0061089873	789 INDIA	0

```
11 rows in set (0.00 sec)

mysql>
```

### Task 3. Aggregate functions, Having, Order By, GroupBy and Joins:

1. Write an SQL query to retrieve a list of all orders along with customer information (e.g., customer name) for each order.

```
ERROR 1054 (42S22): Unknown column 'NumberOfOrders' in 'field list'
mysql> SELECT
->   Orders.OrderID,
->   Orders.OrderDate,
->   Orders.TotalAmount,
->   Customers.FirstName AS CustomerFirstName,
->   Customers.LastName AS CustomerLastName,
->   Customers.Email AS CustomerEmail
-> FROM
->   Orders
-> JOIN
->   Customers ON Orders.CustomerID = Customers.CustomerID;
```

OrderID	OrderDate	TotalAmount	CustomerFirstName	CustomerLastName	CustomerEmail
2	2024-01-13	7782.46	sumit	thakur	smith@gamil.com
4	2024-01-15	25497.94	raju	singh	w@hotmail.com
5	2024-01-16	10988.96	alisha	ron	a.b@bing.com
6	2024-01-17	45899.95	Mohan	ram	miohan@gmail.com
7	2024-01-18	65844.97	Soni	aditi	sa@gmail.com
8	2024-01-19	25434.99	Darick	junaaid	dj@gmail.com
9	2024-01-20	1239.98	sushant	White	rama@gmail.com
10	2024-01-21	35689.99	bharat	kumar	bk@gmail.com
11	2024-01-23	5000.00	SINGH	SUSHANT	ssushant@YAHOO.com

```
9 rows in set (0.00 sec)

mysql>
```

2. Write an SQL query to find the total revenue generated by each electronic gadget product. Include the product name and the total revenue.

```
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)
HeadPhone 921	1715.97
AirDrops	30332.45
TV	7193.99
LAPTOP	1231.98
Charger	3794.97

```
5 rows in set (0.01 sec)
```

3) Write an SQL query to list all customers who have made at least one purchase. Include their names and contact information.

```
mysql> SELECT DISTINCT
-> C.CustomerID,
-> C.FirstName,
-> C.LastName,
-> C.Email,
-> C.Phone,
-> C.Address
-> FROM
-> Customers AS C
-> JOIN
-> Orders AS O ON C.CustomerID = O.CustomerID;
```

CustomerID	FirstName	LastName	Email	Phone	Address
1	SINGH	SUSHANT	ssushant@YAHOO.com	123-456-7890	salt Main St
2	sumit	thakur	smith@gamil.com	987-654-3210	456 north St
4	raju	singh	w@hotmail.com	333-999-8888	567 office road gve
5	alisha	ron	a.b@bing.com	111-222-3333	890 northoffice Blvd
6	Mohan	ram	miohan@gmail.com	777-888-9999	345 docter south Dr
7	Soni	aditi	sa@gmail.com	444-666-7777	678 mapplplaza
8	Darick	junaaid	dj@gmail.com	222-444-5555	901 lasvagas Ln
9	sushant	White	rama@gmail.com	888-111-2222	parkstreet
10	bharat	kumar	bk@gmail.com	666-333-4444	789 Pine St

```
9 rows in set (0.01 sec)

mysql> SELECT DISTINCT
```

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.

```
mysql> SELECT
-> CustomerID,
-> FirstName,
-> LastName,
-> NumOrders
-> FROM
-> Customers
-> WHERE
-> NumOrders = (SELECT MAX(NumOrders) FROM Customers);
```

CustomerID	FirstName	LastName	NumOrders
2	sumit	thakur	20

```
1 row in set (0.01 sec)
```

5) Write an SQL query to retrieve a list of electronic gadgets along with their corresponding categories.

```
mysql> SELECT
->     ProductName,
->     Category
-> FROM
->     Products;
+-----+-----+
| ProductName | Category |
+-----+-----+
| HeadPhone 921 | Hearing Device |
| Trimer | Selfcare Device |
| AirPods | Hearing Device |
| TV | Home Appliance |
| LAPTOP | Learning Device |
| Charger | Charging Device |
| ProG watch | Selfcare device |
| iPad | Learning Device |
| Screen touch laptop | learning Device |
| Speaker | Hearing Device |
| Mouse | Computer Accessories |
+-----+-----+
11 rows in set (0.01 sec)
```

6. Write an SQL query to calculate the average order value for each customer. Include the customer's name and their average order value.

```
mysql> SELECT
->     C.CustomerID,
->     C.FirstName,
->     C.LastName,
->     AVG(O.TotalAmount) AS AverageOrderValue
-> FROM
->     Customers AS C
-> JOIN
->     Orders AS O ON C.CustomerID = O.CustomerID
-> GROUP BY
->     C.CustomerID, C.FirstName, C.LastName;
+-----+-----+-----+-----+
| CustomerID | FirstName | LastName | AverageOrderValue |
+-----+-----+-----+-----+
| 2 | sumit | thakur | 7782.460000 |
| 4 | raju | singh | 25497.940000 |
| 5 | alisha | ron | 10988.960000 |
| 6 | Mohan | ram | 45899.950000 |
| 7 | Soni | aditi | 65844.970000 |
| 8 | Darick | junaaid | 25434.990000 |
| 9 | sushant | White | 1239.980000 |
| 10 | bharat | kumar | 35689.990000 |
| 1 | SINGH | SUSHANT | 5000.000000 |
+-----+-----+-----+-----+
9 rows in set (0.06 sec)
```



7) Write an SQL query to find the order with the highest total revenue. Include the order ID, customer information, and the total revenue.

```
mysql> SELECT
->     O.OrderID,
->     C.CustomerID,
->     C.FirstName,
->     C.LastName,
->     C.Email,
->     C.Phone,
->     C.Address,
->     SUM(OD.Quantity * P.Price) AS TotalRevenue
-> FROM
->     Orders AS O
-> JOIN
->     Customers AS C ON O.CustomerID = C.CustomerID
-> JOIN
->     OrderDetails AS OD ON O.OrderID = OD.OrderID
-> JOIN
->     Products AS P ON OD.ProductID = P.ProductID
-> GROUP BY
->     O.OrderID, C.CustomerID, C.FirstName, C.LastName, C.Email, C.Phone, C.Address
-> ORDER BY
->     TotalRevenue DESC
-> LIMIT 1;
```

OrderID	CustomerID	FirstName	LastName	Email	Phone	Address	TotalRevenue
4	4	raju	singh	w@hotmail.com	333-999-8888	567 office road gve	25497.94

1 row in set (0.01 sec)

```
mysql>
```

8. Write an SQL query to list electronic gadgets and the number of times each product has been ordered.

```
mysql> SELECT
->     P.ProductID,
->     P.ProductName,
->     P.Category,
->     COUNT(OD.ProductID) AS NumOrders
-> FROM
->     Products AS P
-> LEFT JOIN
->     OrderDetails AS OD ON P.ProductID = OD.ProductID
-> WHERE
->     P.Category = 'Selfcare Device'
-> GROUP BY
->     P.ProductID, P.ProductName, P.Category
-> ORDER BY
->     NumOrders DESC;
```

ProductID	ProductName	Category	NumOrders
2	Trimer	Selfcare Device	0
7	ProG watch	Selfcare device	0

2 rows in set (0.00 sec)

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

```
mysql> SELECT
->     C.CustomerID,
->     C.FirstName,
->     C.LastName,
->     C.Email,
->     C.Phone,
->     C.Address
-> 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.ProductName = 'Laptop';
```

CustomerID	FirstName	LastName	Email	Phone	Address
4	raju	singh	w@hotmail.com	333-999-8888	567 office road gve

1 row in set (0.00 sec)

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.

```
mysql> SELECT
->     SUM(O.TotalAmount) AS TotalRevenue
-> FROM
->     Orders AS O
-> WHERE
->     O.OrderDate BETWEEN '2024-01-19' AND '2024-01-23';
```

TotalRevenue
67364.96

1 row in set (0.00 sec)

## Task 4. Subquery and its type:

- 1) Write an SQL query to find out which customers have not placed any orders.

```
mysql> SELECT
->     C.CustomerID,
->     C.FirstName,
->     C.LastName,
->     C.Email,
->     C.Phone,
->     C.Address
-> FROM
->     Customers AS C
-> LEFT JOIN
->     Orders AS O ON C.CustomerID = O.CustomerID
-> WHERE
->     O.OrderID IS NULL;
+-----+-----+-----+-----+-----+-----+
| CustomerID | FirstName | LastName | Email | Phone | Address |
+-----+-----+-----+-----+-----+-----+
|          3 | Aman     | Javed   | aj@gamil.com | 555-123-4567 | 789 south St |
|         12 | SINGH    | SUSHANTS | ssushant@gamil.com | 0061089873 | 789 INDIA |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

2. Write an SQL query to find the total number of products available for sale.

```
mysql> SELECT COUNT(*) AS TotalProductsForSale
-> FROM Products;
+-----+
| TotalProductsForSale |
+-----+
|          11 |
+-----+
1 row in set (0.00 sec)

mysql>
```

3. Write an SQL query to calculate the total revenue generated by TechShop

```
mysql>
mysql> SELECT
  ->     IFNULL(SUM(O.TotalAmount), 0) AS TotalRevenue
  -> FROM
  ->     Orders AS O
  -> JOIN
  ->     Customers AS C ON O.CustomerID = C.CustomerID;
+-----+
| TotalRevenue |
+-----+
|    223379.24 |
+-----+
1 row in set (0.00 sec)

mysql> _
```

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.

```
mysql> DELIMITER ##
mysql> CREATE PROCEDURE CalculateAverageQuantity(IN category_name VARCHAR(255))
  -> BEGIN
  ->     SELECT
  ->         AVG(OD.Quantity) AS AverageQuantity
  ->     FROM
  ->         Products P
  ->     JOIN
  ->         OrderDetails OD ON P.ProductID = OD.ProductID
  ->     JOIN
  ->         Orders O ON OD.OrderID = O.OrderID
  ->     WHERE
  ->         P.Category = category_name;
  -> END ##
Query OK, 0 rows affected (0.00 sec)

mysql> DELIMITER ;
mysql> CALL CalculateAverageQuantity('Hearing Device');
+-----+
| AverageQuantity |
+-----+
|          2.6667 |
+-----+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)

mysql> _
```

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.

```
mysql> DELIMITER ##
mysql> CREATE PROCEDURE CalculateTotalRevenue(IN customer_id INT)
-> BEGIN
->     SELECT
->         COALESCE(SUM(O.TotalAmount), 0) AS TotalRevenue
->     FROM
->         Orders O
->     WHERE
->         O.CustomerID = customer_id;
-> END ##
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql> CALL CalculateTotalRevenue(1);
+-----+
| TotalRevenue |
+-----+
|      5000.00 |
+-----+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.03 sec)

mysql>
```

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.

```
mysql> SELECT
->     C.CustomerID,
->     C.FirstName,
->     C.LastName,
->     COUNT(O.OrderID) AS NumOrders
-> FROM
->     Customers AS C
-> JOIN
->     Orders AS O ON C.CustomerID = O.CustomerID
-> ORDER BY
->     NumOrders DESC
-> LIMIT 10;
+-----+-----+-----+-----+
| CustomerID | FirstName | LastName | NumOrders |
+-----+-----+-----+-----+
|          1 | SINGH     | SUSHANT  |          9 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

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.

```
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. 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.

```
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. Write an SQL query to calculate the average order value (total revenue divided by the number of orders) for all customers.

```
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> _
```

10. Write an SQL query to find the total number of orders placed by each customer and list their names along with the order count.

```
mysql> SELECT
->   C.CustomerID,
->   C.FirstName,
->   C.LastName,
->   COUNT(O.OrderID) AS OrderCount
-> FROM
->   Customers AS C
-> LEFT JOIN
->   Orders AS O ON C.CustomerID = O.CustomerID
-> GROUP BY
->   C.CustomerID, C.FirstName, C.LastName
-> ORDER BY
->   OrderCount DESC;
+-----+-----+-----+-----+
| CustomerID | FirstName | LastName | OrderCount |
+-----+-----+-----+-----+
|          1 | SINGH    | SUSHANT |          1 |
|          2 | sumit    | thakur  |          1 |
|          4 | raju     | singh   |          1 |
|          5 | alisha   | ron     |          1 |
|          6 | Mohan    | ram     |          1 |
|          7 | Soni     | aditi   |          1 |
|          8 | Darick   | junaaid |          1 |
|          9 | sushant  | white   |          1 |
|         10 | bharat   | kumar   |          1 |
|          3 | Aman     | Javed   |          0 |
|         12 | SINGH    | SUSHANTS |          0 |
+-----+-----+-----+-----+
11 rows in set (0.01 sec)
```