E:\shopping.sql

```sql
 1    CREATE DATABASE onlineshopping;
 2   Use onlineshopping;
 3
 4  CREATE TABLE onlineshoppingsite
 5  (
 6    Regno INTEGER ,
 7    Name TEXT ,
 8    web_address TEXT
 9  );
10
11  -- insert
12  INSERT INTO onlineshoppingsite VALUES (0001, 'Shoppingzone', '197.510');
13  INSERT INTO onlineshoppingsite VALUES (0002, 'Shoppingzone', '193.023');
14
15
16  -- fetch
17  SELECT * FROM onlineshoppingsite;
18
19  create table Customer(c_id INTEGER, Name TEXT, phoneno INTEGER, address TEXT);
20  INSERT INTO Customer VALUES (091, 'john', 9995278999, 'prestige building  bangalore');
21  INSERT INTO Customer VALUES (092, 'honay', 9290003328, 'galleria apartments bangalore');
22  INSERT INTO Customer VALUES (093, 'usman', 9924560987, 'orion building bangalore');
23
24  SELECT * FROM Customer;
25
26
27
28  CREATE TABLE Products (p_id INTEGER, Name TEXT, price INTEGER, Model TEXT);
29  INSERT INTO Products VALUES (001, 'xiaomi mobile', 33000, 'note 9 pro' );
30  INSERT INTO Products VALUES (002, 'whirlpool refrigirator',75000, 'doubledoor');
31  INSERT INTO Products VALUES (003, 'samsung TV', 65000, 'uhd');
32  INSERT INTO Products VALUES (004, 'prestige cooker', 990, 'pg990');
33  INSERT INTO Products VALUES (005, 'TP link', 2500, 'tp89');
34
35  SELECT * FROM Products;
36
37
38  CREATE TABLE cartitem (item_id INTEGER, quantity INTEGER, Totalcost INTEGER);
39  INSERT INTO cartitem VALUES (001, 1, 33000);
40  INSERT INTO cartitem VALUES (002, 1, 75000);
41  INSERT INTO cartitem VALUES (003, 1, 65000);
42  INSERT INTO cartitem VALUES (004, 2, 65000);
43  INSERT INTO cartitem VALUES (005, 2, 2500);
44
45  SELECT * FROM cartitem;
46
47  CREATE TABLE cart (cart_id INTEGER, total_cost INTEGER);
48  INSERT INTO cart VALUES (098, 240500);
```

```sql
49
50   SELECT * FROM cart;
51
52
53   CREATE TABLE payment (pay_id INTEGER, p_type TEXT, Amount INTEGER);
54   INSERT INTO payment VALUES (2440, 'card', 240500);
55
56   SELECT * FROM payment;
57   SELECT *
58   FROM Products
59   WHERE product_id = 1; -- Replace 1 with the desired product_id
60   SELECT p.product_id, p.product_name, p.price, pr.discount_percentage
61   FROM Products p
62   LEFT JOIN Promotions pr ON p.product_id = pr.product_id
63   WHERE pr.start_date <= CURRENT_DATE AND pr.end_date >= CURRENT_DATE;
64   SELECT *
65   FROM Customers
66   WHERE email = 'john.doe@example.com'; -- Replace with the desired email address
67   SELECT o.order_id, o.order_date, o.total_amount
68   FROM Orders o
69   WHERE o.customer_id = 1; -- Replace 1 with the desired customer_id
70   UPDATE Products p
71   JOIN Order_Items oi ON p.product_id = oi.product_id
72   SET p.stock_quantity = p.stock_quantity - oi.quantity
73   WHERE oi.order_id = 1; -- Replace 1 with the order_id of the completed order
74   SELECT oi.product_id, p.product_name, SUM(oi.quantity) AS total_sold
75   FROM Order_Items oi
76   JOIN Products p ON oi.product_id = p.product_id
77   GROUP BY oi.product_id, p.product_name
78   ORDER BY total_sold DESC
79   LIMIT 5; -- Adjust limit as needed
80
```