DAY 2 ASSIGNMENT

- HARSHINI V

-- Create a database "ecom"

create database ecom;
use ecom;

0 %

Messages
Commands completed successfully.

Completion time: 2024-09-19T16:19:56.6958945+05:30

-- Create table customer

CREATE TABLE customer (
customerID INT PRIMARY KEY,
firstName VARCHAR(50),
lastName VARCHAR(50),
Email VARCHAR(100),
address VARCHAR(255)
);

90 %

Media Messages
Commands completed successfully.
Completion time: 2024-09-20T16:44:01.3666243+05:30

-- Create table product

CREATE TABLE product (
productID INT PRIMARY KEY,
name VARCHAR(100),
Description VARCHAR(255),
price DECIMAL(10, 2),
stockQuantity INT
);

90 %

Messages
Commands completed successfully.
Completion time: 2024-09-20T16:48:20.1787531+05:30

-- Create table cart

-- Create table orders

CREATE TABLE orders(
orderID INT PRIMARY KEY,
customerID INT,
orderDate DATE,
totalAmount DECIMAL(10, 2),
FOREIGN KEY (customerID) REFERENCES customer(customerID)
);

Messages
Commands completed successfully.
Completion time: 2024-09-20T16:49:52.1171114+05:30

-- Create table orderitems

CREATE TABLE orderitems (
 orderItemID INT PRIMARY KEY,
 orderID INT,
 productID INT,
 quantity INT,
 itemAmount DECIMAL(10, 2),
 FOREIGN KEY (orderID) REFERENCES orders(orderID),
 FOREIGN KEY (productID) REFERENCES product(productID)
);

90 %

Messages
 Commands completed successfully.

Completion time: 2024-09-20T16:50:59.6483488+05:30

-- insert values in table customer

insert into customer (customerid, firstName, lastName, Email, address) values

(1, 'John', 'Doe', 'johndoe@example.com', '123 Main St, City'),

(2, 'Jane', 'Smith', 'janesmith@example.com', '456 Elm St, Town'),

(3, 'Robert', 'Johnson', 'robert@example.com', '789 Oak St, Village'),

(4, 'Sarah', 'Brown', 'sarah@example.com', '101 Pine St, Suburb'),

(5, 'David', 'Lee', 'david@example.com', '234 Cedar St, District'),

(6, 'Laura', 'Hall', 'laura@example.com', '567 Birch St, County'),

(7, 'Michael', 'Davis', 'michael@example.com', '890 Maple St, State'),

(8, 'Emma', 'Wilson', 'emma@example.com', '321 Redwood St, Country'),

(9, 'William', 'Taylor', 'william@example.com', '432 Spruce St, Province'),

(10, 'Olivia', 'Adams', 'olivia@example.com', '765 Fir St, Territory');

Messages

(10 rows affected)

Completion time: 2024-09-20T16:53:29.4467679+05:30

-- insert values in table product

INSERT INTO product (productID, name, Description, price, stockQuantity) VALUES

(1, 'Laptop', 'High-performance laptop', 800.00, 10),
(2, 'Smartphone', 'Latest smartphone', 600.00, 15),
(3, 'Tablet', 'Portable tablet', 300.00, 20),
(4, 'Headphones', 'Noise-canceling', 150.00, 30),
(5, 'TV', '4K Smart TV', 900.00, 5),
(6, 'Coffee Maker', 'Automatic coffee maker', 50.00, 257),
(7, 'Refrigerator', 'Energy-efficient', 700.00, 10),
(8, 'Microwave Oven', 'Countertop microwave', 80.00, 15),
(9, 'Blender', 'High-speed blender', 70.00, 20),
(10, 'Vacuum Cleaner', 'Bagless vacuum cleaner', 120.00, 10);

Messages

(10 rows affected)

Completion time: 2024-09-20T16:54:51.2553758+05:30

-- insert values in table cart

```
INSERT INTO cart (cartID, customerID, productID, quantity) VALUES

(1, 1, 1, 2),
(2, 1, 3, 1),
(3, 2, 2, 3),
(4, 3, 4, 4),
(5, 3, 5, 2),
(6, 4, 6, 1),
(7, 5, 1, 1),
(8, 6, 10, 2),
(9, 6, 9, 3),
(10, 7, 2, 2);

Messages

(10 rows affected)

Completion time: 2024-09-20T16:56:05.8750179+05:30
```

-- insert values in table orders

```
INSERT INTO orders (orderID, customerID, orderDate, totalAmount) VALUES

(1, 1, '2023-01-05', 1200.00),
(2, 2, '2023-02-10', 900.00),
(3, 3, '2023-03-15', 300.00),
(4, 4, '2023-04-20', 150.00),
(5, 5, '2023-05-25', 1800.00),
(6, 6, '2023-06-30', 400.00),
(7, 7, '2023-07-05', 700.00),
(8, 8, '2023-08-10', 160.00),
(9, 9, '2023-09-15', 140.00),
(10, 10, '2023-10-20', 1400.00);

D %

Messages

(10 rows affected)

Completion time: 2024-09-20T16:56:56.5866061+05:30
```

-- insert values in table orderitems

-- 1) Update refrigerator product price to 800.

Completion time: 2024-09-20T16:57:42.5274320+05:30

	productID	name	Description	price	stockQuantity
1	1	Laptop	High-performance laptop	800.00	10
2	2	Smartphone	Latest smartphone	600.00	15
3	3	Tablet	Portable tablet	300.00	20
4	4	Headphones	Noise-canceling	150.00	30
5	5	TV	4K Smart TV	900.00	5
6	6	Coffee Maker	Automatic coffee maker	50.00	257
7	7	Refrigerator	Energy-efficient	800.00	10
8	8	Microwave Oven	Countertop microwave	80.00	15
9	9	Blender	High-speed blender	70.00	20
10	10	Vacuum Clean	Bagless vacuum cleaner	120.00	10

-- 2) Remove all cart items for a specific customer.

```
DELETE FROM cart
WHERE customerID = 3;

90 %

Messages

(2 rows affected)

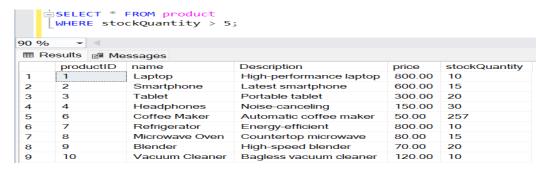
Completion time: 2024-09-20T17:01:25.6400251+05:30
```

	cartID	customerID	productID	quantity				
1	1	1	1	2				
2	2	1	3	1				
3	3	2	2	3				
4	6	4	6	1				
5	7	5	1	1				
6	8	6	10	2				
7	9	6	9	3				
8	10	7	2	2				

-- 3) Retrieve Products Priced Below \$100.



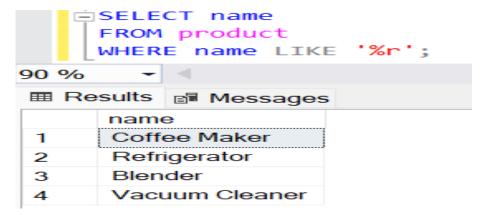
-- 4) Find Products with Stock Quantity Greater Than 5.



-- 5) Retrieve Orders with Total Amount Between \$500 and \$1000.



--6) Find Products which name end with letter 'r'.



--7) Retrieve Cart Items for Customer 5.



-- 8) Subquery to Find Products Not in the Cart.

```
SELECT * FROM product
    WHERE productID NOT IN (
    SELECT DISTINCT productID
    FROM cart );
productID name
4 Headphones
                              Description
                                                   price
                                                          stockQuantity
                              Noise-canceling
                                                   150.00
                                                          30
2
     5
               TV
                              4K Smart TV
                                                   900.00
                                                          5
                                                   800.00
     7
                              Energy-efficient
               Refrigerator
                                                          10
3
4
     8
               Microwave Oven | Countertop microwave | 80.00
                                                          15
```

-- 9) Subquery to Find Customers Who Haven't Placed Orders.

