

SQL COMMANDS:

– Create a new database ecommerce

```
CREATE DATABASE ecommerce
USE ecommerce
```

– Create three tables: customers, orders, and products.

```
CREATE TABLE Customers(
  id INTEGER PRIMARY KEY AUTOINCREMENT,
  name varchar(30),
  email varchar(50),
  address varchar(255)
);
CREATE TABLE Products(
  id INTEGER PRIMARY KEY AUTOINCREMENT,
  name varchar(30),
  price INTEGER,
  description TEXT
);
CREATE TABLE Orders(
  id INTEGER PRIMARY KEY AUTOINCREMENT,
  customer_id INTEGER,
  order_date DATE,
  total_amount INTEGER,
  FOREIGN KEY(customer_id) REFERENCES Customers(id)
);
```

– Inserting sample data into the tables.

```
INSERT INTO Customers(name,email,address) VALUES
("Vikram","vikram1996@gmail.com","18,gandhi road,
cbe-636554"),("Sabarish","Sab1952ri@yahoo.com","57/85 anna nagar, chennai -
636854"),("Madhumitha","Madhumids@gmail.com","plot no.8, Dharmapuri -
636984"),("Hemanth","Hemanth695785@gmail.com","25-96, saradha college road, Salem
-636003"),("Badri","Badrivisalakshi41@gmail.com","25-65/A, Omalur main road, Salem
-636004"),("Anu","Anumeenu@gmail.com","Meenambakkam, Chennai
-636574"),("Alice","Alice2001@gmail.com","Flat no. 248, Saradha apartments, Chennai
-695874"),("Bhavani","Bhava@gmail.com","789 ayz street, Kanyakumari -535584");
```

```

INSERT INTO Products(name,price,description) VALUES
("MBJ Women's Solid Short Sleeve Shirt","499","95% RAYON 5% SPANDEX, Made in USA
or Imported, Do Not Bleach, Lightweight fabric with great stretch for comfort"),("Opna
Women's Short Sleeve Pink t-shirt","699","100% Polyester, Machine wash, 100% cationic
polyester interlock, Machine Wash & Pre Shrunk for a Great Fit, Lightweight, roomy and highly
breathable with moisture wicking fabric "),("DANVOY Womens T Shirt Casual
Cotton","599","95%Cotton,5%Spandex, Features: Casual, Short Sleeve, Letter
Print,V-Neck,Fashion Tees, The fabric is soft and has some stretch"),("Pearl necklace with
stud","2999","Pearl necklaces are a timeless piece of pearl jewellery that never goes out of style.
Buy this stunning white pearl necklace with a beautiful pair of earrings"),("Mens Casual
Premium Slim Fit T-Shirts","399","Slim-fbreathable and comfortable wearing. And Solid
stitched shirts with round neck made for durability and a great fit for casual wear"),("Mens
Cotton Jacket","1999","Great outerwear jackets for Spring/Autumn/Winter, suitable for many
occasions, such as working, hiking, camping, mountain/rock climbing, cycling, traveling or other
outdoors."),("Mens Casual Slim Fit","699","The color could be slightly different between on the
screen and in practice."),("Lock and Love Men's Moto Biker Jacket","3999","Faux leather
material for style and comfort - 2 pockets of front, 2-For-One Hooded denim style faux leather
jacket, Button detail on waist - Detail stitching at sides, HAND WASH ONLY - DO NOT
BLEACH -LINE DRY - DO NOT IRON");

```

```

INSERT INTO Orders(customer_id,order_date,total_amount) VALUES
("1",DATE("now","-5 days"),"1198"),("4",DATE("now","-15
days"),"699"),("5",DATE("now","-30 days"),"3999"),("2",DATE("now","-30
days"),"4698"),("8",DATE("now","-45 days"),"1999"),("3",DATE("now","-5
days"),"599"),("6",DATE("now","-60 days"),"1098"),("7",DATE("now","-45 days"),"499");

```

Queries:

– Customers who have placed an order in the last 30 days

```

Select c.*
FROM Customers c
JOIN Orders o ON c.id=o.customer_id
WHERE o.order_date>=date('now','-30 days');

```

– Total amount of all orders placed by each customer.

```

SELECT c.id, c.name, SUM(p.price) AS total_amount
FROM customers c
JOIN orders o ON c.id = o.customer_id
JOIN Order_items oi ON o.id=oi.order_id

```

```
JOIN Products p ON oi.product_id=p.id
GROUP BY c.id, c.name;
```

--UPDATE the price of Product DANVOY Womens T Shirt Casual Cotton to 45.00

```
UPDATE Products
SET price=45
WHERE name= "DANVOY Womens T Shirt Casual Cotton";
```

--Add a new column discount to the products table.

```
ALTER TABLE Products
ADD COLUMN Discount INTEGER DEFAULT 0;
```

--Retrieve the top 3 products with the highest price.

```
SELECT * FROM Products
Order by price DESC
Limit 3;
```

— Normalizing the database by creating a order_items table to reference the order items

```
CREATE TABLE order_items (
  id INTEGER PRIMARY KEY AUTOINCREMENT,
  order_id INTEGER,
  product_id INTEGER,
  quantity INTEGER,
  FOREIGN KEY (order_id) REFERENCES orders(id),
  FOREIGN KEY (product_id) REFERENCES products(id)
);
INSERT INTO order_items (order_id, product_id, quantity) VALUES
(1,7,1),(2,1,1),(3,3,1),(4,6,1),(5,6,1),(6,3,1),(7,5,1),(8,1,1),(9,3,1),(10,2,1),(11,6,1);
```

– To get the names of customers who have ordered Product “Mens Cotton Jacket”

```
SELECT c.name
FROM Customers c
JOIN Orders o ON c.id=o.customer_id
JOIN order_items oi ON o.id=oi.order_id
JOIN products p ON oi.product_id=p.id
WHERE p.name="Mens Cotton Jacket";
```

--Join the orders and customers tables to retrieve the customer's name and order date for each order.

```
Select c.name,o.order_date  
FROM Customers c  
JOIN Orders o ON c.id=o.customer_id
```

--Retrieve the orders with a total amount greater than 150.00.

```
SELECT c.name,o.*  
FROM Customers c  
JOIN Orders o ON c.id=o.customer_id  
WHERE o.total_amount>150
```

--Retrieve the average total of all orders.

```
CREATE VIEW order_total AS  
SELECT  
o.id AS orderID,o.customer_id AS CustomerID,SUM(p.price*oi.quantity) AS total_amount  
FROM Orders o  
JOIN order_items oi ON o.id=oi.order_id  
JOIN products p ON oi.product_id=p.id  
GROUP BY o.id  
  
SELECT AVG(total_amount) FROM order_total;
```