TASK-3

SAMPLE ECOMMERCE DATA

-- Create Customers table

CREATE TABLE Customers (

CustomerID INTEGER PRIMARY KEY,

CustomerName TEXT,

Region TEXT

);

-- Create Orders table

CREATE TABLE Orders (

OrderID INTEGER PRIMARY KEY,

CustomerID INTEGER,

OrderDate TEXT,

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)

);

-- Create OrderDetails table

CREATE TABLE OrderDetails (

OrderDetailID INTEGER PRIMARY KEY,

OrderID INTEGER,

Product TEXT,

Category TEXT,

Quantity INTEGER,

Price REAL,

FOREIGN KEY (OrderID) REFERENCES Orders(OrderID)

);

-- Insert sample Customers

INSERT INTO Customers VALUES

(1, 'John Doe', 'North'),

(2, 'Jane Smith', 'West'),

(3, 'Emily Chen', 'East'),

(4, 'Michael Brown', 'South');

-- Insert sample Orders

INSERT INTO Orders VALUES

(101, 1, '2025-01-15'),

(102, 2, '2025-01-18'),

(103, 3, '2025-01-20'),

(104, 1, '2025-02-01'),

(105, 4, '2025-02-10');

-- Insert sample OrderDetails

INSERT INTO OrderDetails VALUES

(1, 101, 'Wireless Mouse', 'Electronics', 2, 25.50),

(2, 102, 'Yoga Mat', 'Fitness', 1, 20.00),

(3, 103, 'LED Monitor', 'Electronics', 1, 150.00),

(4, 104, 'Dumbbells', 'Fitness', 3, 35.00),

(5, 105, 'Blender', 'Home Appliances', 1, 45.00);

-- Sample SELECT

SELECT \* FROM Customers;

-- JOIN: Orders with Customer names

SELECT o.OrderID, c.CustomerName, o.OrderDate

FROM Orders o

JOIN Customers c ON o.CustomerID = c.CustomerID;

-- JOIN: Orders with product details

SELECT o.OrderID, c.CustomerName, od.Product, od.Quantity, od.Price

FROM Orders o

JOIN Customers c ON o.CustomerID = c.CustomerID

JOIN OrderDetails od ON o.OrderID = od.OrderID;

-- Total revenue by Category

SELECT Category, SUM(Quantity \* Price) AS TotalRevenue

FROM OrderDetails

GROUP BY Category;

-- View for order summary

CREATE VIEW OrderSummary AS

SELECT o.OrderID, c.CustomerName, SUM(od.Quantity \* od.Price) AS TotalAmount

FROM Orders o

JOIN Customers c ON o.CustomerID = c.CustomerID

JOIN OrderDetails od ON o.OrderID = od.OrderID

GROUP BY o.OrderID;

-- Query the view

SELECT \* FROM OrderSummary WHERE TotalAmount > 50;

-- Index creation (SQLite ignores some index types, but this works)

CREATE INDEX idx\_orders\_customerid ON Orders(CustomerID);