-- Create Database

CREATE DATABASE EcommerceDB;

GO

USE EcommerceDB;

GO

-- USERS

CREATE TABLE Users (

UserID INT IDENTITY(1,1) PRIMARY KEY,

FirstName NVARCHAR(50) NOT NULL,

LastName NVARCHAR(50),

Email NVARCHAR(100) UNIQUE NOT NULL,

PasswordHash NVARCHAR(255) NOT NULL,

Phone NVARCHAR(20),

CreatedAt DATETIME DEFAULT GETDATE(),

IsActive BIT DEFAULT 1

);

-- ADDRESSES

CREATE TABLE Addresses (

AddressID INT IDENTITY(1,1) PRIMARY KEY,

UserID INT FOREIGN KEY REFERENCES Users(UserID) ON DELETE CASCADE,

Line1 NVARCHAR(255) NOT NULL,

Line2 NVARCHAR(255),

City NVARCHAR(100) NOT NULL,

State NVARCHAR(100),

PostalCode NVARCHAR(20),

Country NVARCHAR(50) DEFAULT 'India',

IsBilling BIT DEFAULT 0,

IsShipping BIT DEFAULT 1

);

-- CATEGORIES

CREATE TABLE Categories (

CategoryID INT IDENTITY(1,1) PRIMARY KEY,

Name NVARCHAR(100) UNIQUE NOT NULL,

ParentID INT FOREIGN KEY REFERENCES Categories(CategoryID)

);

-- PRODUCTS

CREATE TABLE Products (

ProductID INT IDENTITY(1,1) PRIMARY KEY,

SKU NVARCHAR(50) UNIQUE NOT NULL,

Name NVARCHAR(100) NOT NULL,

Description NVARCHAR(MAX),

Price DECIMAL(10,2) NOT NULL,

CategoryID INT FOREIGN KEY REFERENCES Categories(CategoryID),

CreatedAt DATETIME DEFAULT GETDATE(),

IsActive BIT DEFAULT 1

);

-- PRODUCT IMAGES

CREATE TABLE ProductImages (

ImageID INT IDENTITY(1,1) PRIMARY KEY,

ProductID INT FOREIGN KEY REFERENCES Products(ProductID) ON DELETE CASCADE,

Url NVARCHAR(255) NOT NULL,

AltText NVARCHAR(255),

Ordering INT DEFAULT 0

);

-- INVENTORY

CREATE TABLE Inventory (

ProductID INT PRIMARY KEY FOREIGN KEY REFERENCES Products(ProductID) ON DELETE CASCADE,

QtyAvailable INT NOT NULL DEFAULT 0,

SafetyStock INT NOT NULL DEFAULT 5,

LastRestock DATETIME

);

-- SUPPLIERS

CREATE TABLE Suppliers (

SupplierID INT IDENTITY(1,1) PRIMARY KEY,

Name NVARCHAR(100) NOT NULL,

ContactEmail NVARCHAR(100),

Phone NVARCHAR(20)

);

CREATE TABLE ProductSupplier (

ProductID INT FOREIGN KEY REFERENCES Products(ProductID) ON DELETE CASCADE,

SupplierID INT FOREIGN KEY REFERENCES Suppliers(SupplierID),

SupplierSKU NVARCHAR(50),

PRIMARY KEY (ProductID, SupplierID)

);

-- ORDERS

CREATE TABLE Orders (

OrderID INT IDENTITY(1,1) PRIMARY KEY,

UserID INT FOREIGN KEY REFERENCES Users(UserID),

OrderDate DATETIME DEFAULT GETDATE(),

Status NVARCHAR(20) DEFAULT 'Pending', -- could use CHECK constraint

ShippingAddressID INT FOREIGN KEY REFERENCES Addresses(AddressID),

BillingAddressID INT FOREIGN KEY REFERENCES Addresses(AddressID),

Subtotal DECIMAL(12,2) NOT NULL,

ShippingFee DECIMAL(8,2) DEFAULT 0,

DiscountAmt DECIMAL(10,2) DEFAULT 0,

TaxAmt DECIMAL(10,2) DEFAULT 0,

TotalAmt DECIMAL(12,2) NOT NULL

);

-- ORDER ITEMS

CREATE TABLE OrderItems (

OrderItemID INT IDENTITY(1,1) PRIMARY KEY,

OrderID INT FOREIGN KEY REFERENCES Orders(OrderID) ON DELETE CASCADE,

ProductID INT FOREIGN KEY REFERENCES Products(ProductID),

UnitPrice DECIMAL(10,2) NOT NULL,

Quantity INT NOT NULL,

LineTotal DECIMAL(12,2) NOT NULL

);

-- PAYMENTS

CREATE TABLE Payments (

PaymentID INT IDENTITY(1,1) PRIMARY KEY,

UserID INT FOREIGN KEY REFERENCES Users(UserID),

PaidAt DATETIME DEFAULT GETDATE(),

Amount DECIMAL(12,2) NOT NULL,

Method NVARCHAR(50) NOT NULL,

ExternalRef NVARCHAR(100),

Status NVARCHAR(50) DEFAULT 'Completed'

);

-- COUPONS

CREATE TABLE Coupons (

CouponCode NVARCHAR(50) PRIMARY KEY,

Description NVARCHAR(255),

DiscountPct DECIMAL(5,2),

DiscountAmt DECIMAL(10,2),

MinOrderAmt DECIMAL(12,2),

StartsAt DATETIME,

EndsAt DATETIME,

Active BIT DEFAULT 1

);

-- REVIEWS

CREATE TABLE Reviews (

ReviewID INT IDENTITY(1,1) PRIMARY KEY,

ProductID INT FOREIGN KEY REFERENCES Products(ProductID) ON DELETE CASCADE,

UserID INT FOREIGN KEY REFERENCES Users(UserID),

Rating INT CHECK (Rating >= 1 AND Rating <= 5),

Title NVARCHAR(255),

Body NVARCHAR(MAX),

CreatedAt DATETIME DEFAULT GETDATE()

);

-- CART

CREATE TABLE Carts (

CartID INT IDENTITY(1,1) PRIMARY KEY,

UserID INT UNIQUE FOREIGN KEY REFERENCES Users(UserID),

UpdatedAt DATETIME DEFAULT GETDATE()

);

CREATE TABLE CartItems (

CartItemID INT IDENTITY(1,1) PRIMARY KEY,

CartID INT FOREIGN KEY REFERENCES Carts(CartID) ON DELETE CASCADE,

ProductID INT FOREIGN KEY REFERENCES Products(ProductID),

Qty INT NOT NULL DEFAULT 1,

AddedAt DATETIME DEFAULT GETDATE(),

CONSTRAINT UQ\_Cart\_Product UNIQUE (CartID, ProductID)

);

-- WISHLIST

CREATE TABLE Wishlists (

WishlistID INT IDENTITY(1,1) PRIMARY KEY,

UserID INT FOREIGN KEY REFERENCES Users(UserID),

Name NVARCHAR(100) DEFAULT 'My Wishlist',

CreatedAt DATETIME DEFAULT GETDATE()

);

CREATE TABLE WishlistItems (

WishlistItemID INT IDENTITY(1,1) PRIMARY KEY,

WishlistID INT FOREIGN KEY REFERENCES Wishlists(WishlistID) ON DELETE CASCADE,

ProductID INT FOREIGN KEY REFERENCES Products(ProductID),

AddedAt DATETIME DEFAULT GETDATE(),

CONSTRAINT UQ\_Wishlist\_Product UNIQUE (WishlistID, ProductID)

);

-- SHIPMENTS

CREATE TABLE Shipments (

ShipmentID INT IDENTITY(1,1) PRIMARY KEY,

OrderID INT FOREIGN KEY REFERENCES Orders(OrderID),

ShippedAt DATETIME,

DeliveredAt DATETIME,

Carrier NVARCHAR(100),

TrackingNo NVARCHAR(100),

Status NVARCHAR(50)

);

-- ORDER STATUS HISTORY

CREATE TABLE OrderStatusHistory (

HistoryID INT IDENTITY(1,1) PRIMARY KEY,

OrderID INT FOREIGN KEY REFERENCES Orders(OrderID) ON DELETE CASCADE,

Status NVARCHAR(20),

ChangedAt DATETIME DEFAULT GETDATE(),

ChangedBy NVARCHAR(100)

);

-- USERS

INSERT INTO Users (FirstName, LastName, Email, PasswordHash, Phone) VALUES

('Aman', 'Kumar', 'aman@example.com', 'hash1', '+91-9000000001'),

('Priya', 'Sharma', 'priya@example.com', 'hash2', '+91-9000000002'),

('Rohit', 'Verma', 'rohit@example.com', 'hash3', '+91-9000000003');

-- ADDRESSES

INSERT INTO Addresses (UserID, Line1, City, State, PostalCode, Country, IsBilling, IsShipping) VALUES

(1, '23 MG Road', 'New Delhi', 'Delhi', '110001', 'India', 1, 1),

(2, '12 Park Street', 'Kolkata', 'West Bengal', '700016', 'India', 1, 1),

(3, '45 Linking Road', 'Mumbai', 'Maharashtra', '400050', 'India', 1, 1);

-- CATEGORIES

INSERT INTO Categories (Name, ParentID) VALUES

('Electronics', NULL),

('Mobiles', 1),

('Accessories', 1),

('Home & Kitchen', NULL);

-- PRODUCTS

INSERT INTO Products (SKU, Name, Description, Price, CategoryID) VALUES

('SKU-001', 'Smartphone A', 'Midrange smartphone, 6GB RAM', 15999.00, 2),

('SKU-002', 'Wireless Earbuds', 'Bluetooth earbuds with charging case', 2999.00, 3),

('SKU-003', 'Stainless Steel Pan', 'Non-stick fry pan 24cm', 1299.00, 4);

-- INVENTORY

INSERT INTO Inventory (ProductID, QtyAvailable, SafetyStock, LastRestock) VALUES

(1, 120, 10, GETDATE()-10),

(2, 45, 5, GETDATE()-3),

(3, 10, 2, GETDATE()-20);

-- SUPPLIERS

INSERT INTO Suppliers (Name, ContactEmail, Phone) VALUES

('ABC Electronics', 'sales@abcelect.com', '+91-11-2345000'),

('HomeGoods Pvt Ltd', 'contact@homegoods.com', '+91-22-7889000');

INSERT INTO ProductSupplier (ProductID, SupplierID, SupplierSKU) VALUES

(1, 1, 'ABC-SM-A-01'),

(2, 1, 'ABC-EB-02'),

(3, 2, 'HG-PAN-24');

-- COUPONS

INSERT INTO Coupons (CouponCode, Description, DiscountPct, DiscountAmt, MinOrderAmt, StartsAt, EndsAt, Active) VALUES

('WELCOME10', '10% off for new users', 10.00, NULL, NULL, GETDATE()-30, GETDATE()+30, 1),

('FLAT500', 'Flat ?500 off on orders over ?5000', NULL, 500.00, 5000.00, GETDATE()-1, GETDATE()+60, 1);

-- PAYMENTS

INSERT INTO Payments (UserID, PaidAt, Amount, Method, ExternalRef) VALUES

(1, GETDATE()-2, 15998.00, 'Card', 'TXN12345'),

(2, GETDATE()-1, 2999.00, 'UPI', 'TXN12346');

-- ORDERS

INSERT INTO Orders (UserID, OrderDate, Status, ShippingAddressID, BillingAddressID, Subtotal, ShippingFee, DiscountAmt, TaxAmt, TotalAmt)

VALUES

(1, GETDATE()-2, 'Delivered', 1, 1, 14999.00, 99.00, 100.00, 900.00, 15998.00),

(2, GETDATE()-1, 'Shipped', 2, 2, 2999.00, 0.00, 0.00, 0.00, 2999.00);

-- ORDER ITEMS

INSERT INTO OrderItems (OrderID, ProductID, UnitPrice, Quantity, LineTotal) VALUES

(1, 1, 14999.00, 1, 14999.00),

(2, 2, 2999.00, 1, 2999.00);

-- REVIEWS

INSERT INTO Reviews (ProductID, UserID, Rating, Title, Body) VALUES

(1, 1, 5, 'Excellent phone', 'Fast performance and great battery life'),

(2, 2, 4, 'Good earbuds', 'Sound quality is nice, but case is a bit large'),

(3, 3, 3, 'Decent pan', 'Works fine but handle gets hot');

-- CARTS

INSERT INTO Carts (UserID) VALUES (1), (2), (3);

-- CART ITEMS

INSERT INTO CartItems (CartID, ProductID, Qty) VALUES

(1, 3, 1),

(2, 1, 1),

(3, 2, 2);

-- WISHLISTS

INSERT INTO Wishlists (UserID, Name) VALUES

(1, 'Aman''s Wishlist'),

(2, 'Priya''s Wishlist'),

(3, 'Rohit''s Wishlist');

-- WISHLIST ITEMS

INSERT INTO WishlistItems (WishlistID, ProductID) VALUES

(1, 2),

(1, 3),

(2, 1),

(3, 3);

-- SHIPMENTS

INSERT INTO Shipments (OrderID, ShippedAt, DeliveredAt, Carrier, TrackingNo, Status) VALUES

(1, GETDATE()-2, GETDATE()-1, 'BlueDart', 'BD123456', 'Delivered'),

(2, GETDATE()-1, NULL, 'Delhivery', 'DL987654', 'Shipped');

-- ORDER STATUS HISTORY

INSERT INTO OrderStatusHistory (OrderID, Status, ChangedAt, ChangedBy) VALUES

(1, 'Pending', GETDATE()-3, 'System'),

(1, 'Delivered', GETDATE()-1, 'System'),

(2, 'Pending', GETDATE()-2, 'System'),

(2, 'Shipped', GETDATE()-1, 'System');

*-- USERS*

*SELECT \* FROM Users;*

*-- ADDRESSES*

*SELECT \* FROM Addresses;*

*-- CATEGORIES*

*SELECT \* FROM Categories;*

*-- PRODUCTS*

*SELECT \* FROM Products;*

*-- INVENTORY*

*SELECT \* FROM Inventory;*

*-- SUPPLIERS*

*SELECT \* FROM Suppliers;*

*-- PRODUCT-SUPPLIER RELATION*

*SELECT \* FROM ProductSupplier;*

*-- COUPONS*

*SELECT \* FROM Coupons;*

*-- PAYMENTS*

*SELECT \* FROM Payments;*

*-- ORDERS*

*SELECT \* FROM Orders;*

*-- ORDER ITEMS*

*SELECT \* FROM OrderItems;*

*-- REVIEWS*

*SELECT \* FROM Reviews;*

*-- CARTS*

*SELECT \* FROM Carts;*

*-- CART ITEMS*

*SELECT \* FROM CartItems;*

*-- WISHLISTS*

*SELECT \* FROM Wishlists;*

*-- WISHLIST ITEMS*

*SELECT \* FROM WishlistItems;*

*-- SHIPMENTS*

*SELECT \* FROM Shipments;*

*-- ORDER STATUS HISTORY*

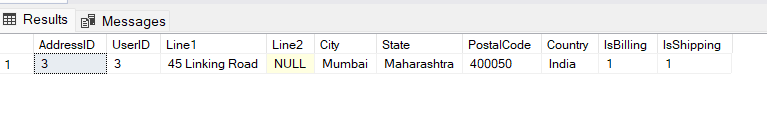
*SELECT \* FROM OrderStatusHistory;*

--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*STUDY\*\*\*\*\*\*\*\*\*\*\*

**--List all customers who live in "Mumbai".**

*select \* from ADDRESSES*

*where City='Mumbai'*

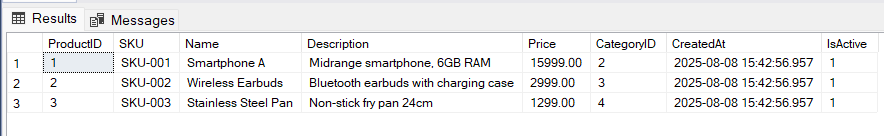


**--Find all products with a price greater than 1000, sorted from most expensive to least.**

*SELECT \* FROM PRODUCTS*

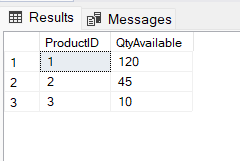
*WHERE Price>1000*

*order by Price DESC*



**--Display the current stock of each product from the Inventory table.**

*SELECT ProductID , QtyAvailable FROM INVENTORY*

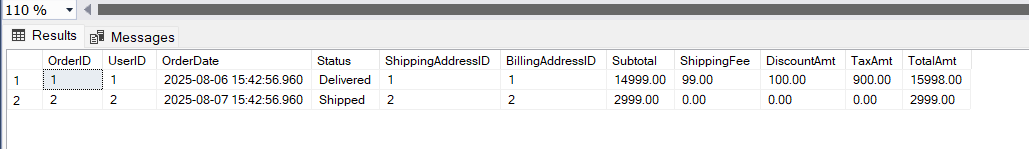


**--List all orders placed in the last 30 days.**

*SELECT \**

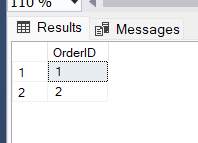
*FROM Orders*

*WHERE OrderDate >= DATEADD(DAY, -30, GETDATE());*



**--Find all orders that have not yet been shipped.**

*SELECT OrderID From OrderStatusHistory where Status='Pending'*

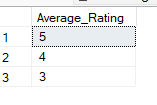


**--Show the average rating for each product from the Reviews table.**

*select AVG(Rating) AS Average\_Rating*

*from Reviews*

*Group by ProductID*



**--Display the most recent status change for each order from OrderStatusHistory.**

*select OrderID , Status*

*from OrderStatusHistory*

