Name : Prathik Balaji N
Date : 25-07-24

1.Create a DBMS using OLTP for ECOMMERCE.

Res 1:

```
CREATE TABLE Customers (
    CustomerID INT PRIMARY KEY IDENTITY(1,1),
    CustomerName NVARCHAR(100),
    [Location] NVARCHAR(100)
);

CREATE TABLE Products (
    ProductID INT PRIMARY KEY IDENTITY(1,1),
    ProductName NVARCHAR(100),
    Category NVARCHAR(100),
    Price DECIMAL(10, 2)
);

CREATE TABLE Orders (
    OrderID INT PRIMARY KEY IDENTITY(1,1),
    CustomerID INT,
    OrderDate DATE,
    FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
);

CREATE TABLE OrderDetails (
    OrderDetailsID INT PRIMARY KEY IDENTITY(1,1),
    OrderID INT,
    ProductID INT,
    Quantity INT,
    FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),
    FOREIGN KEY (ProductID) REFERENCES Products(ProductID)
);
```

Res 2:

```
INSERT INTO Customers (CustomerName, [Location]) VALUES
('Prathik', 'Chennai'),
('Siddarth', 'Coimbatore'),
('Dhanapal', 'Coimbatore'),
('Tapan','Coimbatore'),
('Sanjai','Bangalore'),
('Harsha','Chennai');

INSERT INTO Products (ProductName, Category, Price) VALUES
('iPhone', 'Mobile', 50000.00),
('Samsung Galaxy', 'Mobile', 40000.00),
('JBL Speaker', 'Speaker', 5000.00),
('Sony Speaker', 'Speaker', 7000.00),
('Vivo','Mobile',15000.00),
('Oppo','Mobile',18000.00);


INSERT INTO Orders (CustomerID, OrderDate) VALUES
(1, getdate()),
(2, getdate()),
(6, getdate()),
(5, getdate());

INSERT INTO OrderDetails (OrderID, ProductID, Quantity) VALUES
(1, 1, 1),
(1, 3, 2),
(2, 2, 1),
(2, 4, 1),
(3, 1, 1),
(3, 5, 2),
(3, 6, 1),
(4, 5, 1);
```

2.Fetch all customers and display if there are any order names

Res 1:

```
SELECT c.CustomerName,
       STRING_AGG(p.ProductName, ',') AS OrderNames
FROM Customers c
LEFT JOIN Orders o ON c.CustomerID = o.CustomerID
LEFT JOIN OrderDetails od ON o.OrderID = od.OrderID
LEFT JOIN Products p ON od.ProductID = p.ProductID
GROUP BY c.CustomerID, c.CustomerName;
```

Res 2:

CustomerName	OrderNames
Prathik	iPhone,JBL Speaker
Siddarth	Samsung Galaxy,Sony Speaker
Dhanapal	NULL
Tapan	NULL
Sanjai	Vivo
Harsha	iPhone,Vivo,Oppo

3.Fetch all the order names placed from particular location

Res 1:

```
SELECT p.ProductName,c.Location
FROM Orders o
INNER JOIN Customers c ON o.CustomerID = c.CustomerID
INNER JOIN OrderDetails oD ON o.OrderID = od.OrderID
INNER JOIN Products p ON od.ProductID = p.ProductID
WHERE c.Location = 'Coimbatore';
```

Res 2:

ProductName	Location
Samsung Galaxy	Coimbatore
Sony Speaker	Coimbatore

4.what is the max price of products from particular category.

Res 1 :

```
SELECT MAX(Price) AS MaxPrice
FROM Products
WHERE Category = 'Mobile';
```

Res 2:

Results Messages

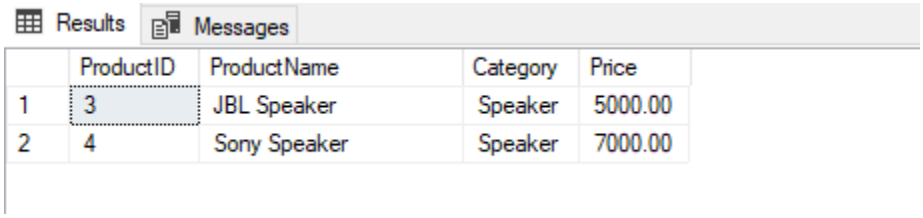
	MaxPrice
1	50000.00

5.Display any product with the productname as like mobile, speaker

Res 1:

```
SELECT *
FROM Products
WHERE ProductName LIKE '%speaker%';
```

Res 2:



Results Messages

	ProductID	ProductName	Category	Price
1	3	JBL Speaker	Speaker	5000.00
2	4	Sony Speaker	Speaker	7000.00

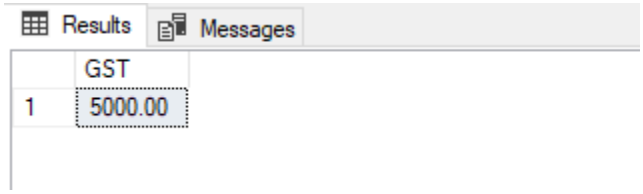
6.Create a function that calculate 10%gst from original price

Res 1:

```
CREATE OR ALTER FUNCTION CalculateGST(@Price DECIMAL(10, 2))
RETURNS DECIMAL(10, 2)
AS
BEGIN
    RETURN @Price * 0.10;
END;

Select dbo.CalculateGST(50000.00) as GST;
```

Res 2:



Results Messages

	GST
1	5000.00

7. Create stored procedure that increases all the product prices by 100. Ensure Atomicity.

Res 1:

```
CREATE OR ALTER PROCEDURE IncreasePrices
AS
BEGIN
    BEGIN TRY
        BEGIN TRANSACTION;
        UPDATE Products SET Price = Price + 100;
        COMMIT TRANSACTION;
    END TRY
    BEGIN CATCH
        IF @@TRANCOUNT > 0
        BEGIN
            ROLLBACK TRANSACTION;
        END

        SELECT ERROR_MESSAGE() , ERROR_STATE();
    END CATCH
END;
```

Res 2 :

Results | Messages

	ProductID	ProductName	Category	Price
1	1	iPhone	Mobile	50100.00
2	2	Samsung Galaxy	Mobile	40100.00
3	3	JBL Speaker	Speaker	5100.00
4	4	Sony Speaker	Speaker	7100.00
5	5	Vivo	Mobile	15100.00
6	6	Oppo	Mobile	18100.00