Day 2
Name : Siddarth
Date : 25/7/2024

Question:
Create a DBMS using OLTP for ECOMMERCE.
1. Fetch all customers and display if there are any order names.
2. Fetch all the order names placed from a particular location.
3. What is the max price of products from a particular category?
4. Display any product with the product name like mobile, speaker.
5. Create a function that calculates 10% GST from the original price.
6. Create a stored procedure that increases all the product prices by 100. Ensure Atomicity.

Solution

Creation of tables:
NOTE: SS is the name of the ecommerce
TABLE 1 : ss_customers

```sql
CREATE TABLE ss_customers (
    customer_id INT PRIMARY KEY,
    customer_name VARCHAR(100),
    email VARCHAR(100),
    phone VARCHAR(15),
    location VARCHAR(100)
);
```

TABLE 2 : ss_orders

```sql
CREATE TABLE ss_orders (
    order_id INT PRIMARY KEY,
    customer_id INT,
    order_name VARCHAR(100),
    order_date DATE,
    status VARCHAR(50),
    FOREIGN KEY (customer_id) REFERENCES ss_customers(customer_id)
);
```

TABLE 3 : ss_products

```sql
CREATE TABLE ss_products (
    product_id INT PRIMARY KEY,
    product_name VARCHAR(100),
    category VARCHAR(100),
    price DECIMAL(10, 2),
    stock_quantity INT
);
```

TABLE 4 : ss_order_items

```sql
CREATE TABLE ss_order_items (
    order_item_id INT PRIMARY KEY,
    order_id INT,
    product_id INT,
    quantity INT,
    price DECIMAL(10, 2),
    FOREIGN KEY (order_id) REFERENCES ss_orders(order_id),
    FOREIGN KEY (product_id) REFERENCES ss_products(product_id)
);
```

Inserting values into the created table

```sql
INSERT INTO ss_customers (customer_id, customer_name, email, phone, location) VALUES
(1, 'Arun Kumar', 'arun@example.com', '9123456780', 'Chennai'),
(2, 'Meena Ramesh', 'meena@example.com', '9234567891', 'Coimbatore'),
(3, 'Rajesh Kumar', 'rajesh@example.com', '9345678902', 'Madurai'),
(4, 'Priya Dinesh', 'priya@example.com', '9456789013', 'Trichy'),
(5, 'Karthik Subramanian', 'karthik@example.com', '9567890124', 'Salem'),
(6, 'Anitha Ravi', 'anitha@example.com', '9678901235', 'Tirunelveli');

INSERT INTO ss_orders (order_id, customer_id, order_name, order_date, status) VALUES
(1, 1, 'Order1', '2023-07-01', 'Shipped'),
(2, 2, 'Order2', '2023-07-02', 'Processing'),
(3, 3, 'Order3', '2023-07-03', 'Delivered'),
(4, 4, 'Order4', '2023-07-04', 'Shipped'),
(5, 5, 'Order5', '2023-07-05', 'Processing'),
(6, 6, 'Order6', '2023-07-06', 'Delivered');

INSERT INTO ss_products (product_id, product_name, category, price, stock_quantity) VALUES
(1, 'Laptop', 'Electronics', 1000.00, 50),
(2, 'Mobile Phone', 'Electronics', 500.00, 100),
(3, 'Speaker', 'Electronics', 150.00, 200),
(4, 'Tablet', 'Electronics', 300.00, 150),
(5, 'Headphones', 'Electronics', 75.00, 300),
(6, 'Washing Machine', 'Home Appliances', 700.00, 30);

INSERT INTO ss_order_items (order_item_id, order_id, product_id, quantity, price) VALUES
(1, 1, 1, 1, 1000.00),
(2, 1, 3, 2, 300.00),
(3, 2, 2, 1, 500.00),
(4, 3, 4, 3, 900.00),
(5, 4, 5, 4, 300.00),
(6, 5, 6, 1, 700.00);
```

ANS 1: Fetch all customers and display if there are any order names

```sql
SELECT c.customer_id, c.customer_name, o.order_name
FROM ss_customers c
LEFT JOIN ss_orders o ON c.customer_id = o.customer_id;
```

% ▾ ◄

Results ▦ Messages ▦ Client Statistics

customer_id	customer_name	order_name
1	Arun Kumar	Order1
2	Meena Ramesh	Order2
3	Rajesh Kumar	Order3
4	Priya Dinesh	Order4
5	Karthik Subramanian	Order5
6	Anitha Ravi	Order6

ANS 2: Fetch all the order names placed from a particular location

```sql
SELECT c.customer_name, o.order_name
FROM ss_orders o
JOIN ss_customers c ON o.customer_id = c.customer_id
WHERE c.location = 'coimbatore';
```

% ▾ ◄

Results ▦ Messages ▦ Client Statistics

customer_name	order_name
Meena Ramesh	Order2

ANS 3: Find the max price of products from a particular category

```sql
DECLARE @Category VARCHAR(50) = 'Home Appliances'
SELECT MAX(price) AS max_price
FROM ss_products
WHERE category = @Category;
```

00 % ▾ ◄

▦ Results ▦ Messages ▦ Client Statistics

max_price
700.00

ANS 4: Display any product with the product name as like mobile, speaker

```sql
SELECT *
FROM ss_products
WHERE product_name LIKE '%mobile%' OR product_name LIKE '%speaker%';
GO
```

100 % ▾ ◄

▦ Results ▦ Messages ▦ Client Statistics

	product_id	product_name	category	price	stock_quantity
1	2	Mobile Phone	Electronics	500.00	100
2	3	Speaker	Electronics	150.00	200

ANS 5: Create a function that calculates 10% GST from the original price

```
CREATE FUNCTION calculate_gst (@price DECIMAL(10, 2))
RETURNS DECIMAL(10, 2)
AS
BEGIN
    RETURN @price * 0.10;
END;
GO
select dbo.calculate_gst(1000);
```

100 %

⊞ Results 📧 Messages 📠 Client Statistics

	(No column name)
1	100.00

ANS 6: Create a stored procedure that increases all the product prices by 100

```
CREATE or alter PROCEDURE increase_product_prices
AS
BEGIN
    BEGIN TRY
    BEGIN TRANSACTION;
    UPDATE ss_products
    SET price = price + 100;
    COMMIT;
    end TRY
    BEGIN CATCH
        DECLARE @ErrorMessage NVARCHAR (400);
        DECLARE @ErrorSeverity INT;
        DECLARE @ErrorState INT;
        SELECT @ErrorMessage= ERROR_MESSAGE(),
                @ErrorSeverity= ERROR_SEVERITY(),
                @ErrorState= ERROR_STATE()
                RAISERROR (@ERRORMessage,@ErrorSeverity,@ErrorState)
        END CATCH
END;
GO
```

When the Sp is executed the price of the products is increased by 100

```
EXEC increase_product_prices;
Select * from ss_products
```

100 %

⊞ Results 📧 Messages 📠 Client Statistics

	product_id	product_name	category	price	stock_quantity
1	1	Laptop	Electronics	1100.00	50
2	2	Mobile Phone	Electronics	600.00	100
3	3	Speaker	Electronics	250.00	200
4	4	Tablet	Electronics	400.00	150
5	5	Headphones	Electronics	175.00	300
6	6	Washing Machine	Home Ap...	800.00	30