Name: Debashmita Saha, Enrollment Number: EBEON0524136576, Batch: 2024-RDMS

PLSQLASSIGNMENT

```
-- Topic 1
DELIMITER //

CREATE PROCEDURE GetCustomerOrders
    @CustomerID INT
AS
BEGIN
    SELECT CustomerName, CustomerEmail, COUNT(OrderID) AS TotalOrders
    FROM Customers
    INNER JOIN Orders ON Customers.CustomerID = Orders.CustomerID
    WHERE Customers.CustomerID = @CustomerID
    GROUP BY CustomerName, CustomerEmail;
END;

DELIMITER;
-- Topic 2
DELIMETER //

CREATE TRIGGER UpdateLastModified
AFTER INSERT OR UPDATE ON ProductTable
FOR EACH ROW
BEGIN
    UPDATE ProductTable
    SET last_modified = CURRENT_TIMESTAMP
    WHERE ProductID = NEW.ProductID;
END;

DELIMITER;

-- topic 3
DELIMITER //

CREATE FUNCTION CalculateTotalRevenue（@ProductID INT）
RETURNS DECIMAL（10,2）
AS
BEGIN
    DECLARE @TotalRevenue DECIMAL（10,2）；

    SELECT @TotalRevenue = SUM（quantity_sold ＊ unit_price）
    FROM Sales
    WHERE product_id = @ProductID;

    RETURN @TotalRevenue;
END;

DELIMITER;

-- Topic 4
DELIMITER //

CREATE PROCEDURE TransferFunds
    @FromAccount INT,
    @ToAccount INT,
    @Amount DECIMAL（10，2）
AS
BEGIN
    DECLARE @FromBalance DECIMAL（10，2）；
    DECLARE @ToBalance DECIMAL（10，2）；

    -- Validate accounts
    SELECT @FromBalance = Balance FROM Accounts WHERE AccountID = @FromAccount;
    SELECT @ToBalance = Balance FROM Accounts WHERE AccountID = @ToAccount;

    IF @FromBalance >= @Amount
    BEGIN
        -- Update balances
        UPDATE Accounts SET Balance = @FromBalance - @Amount WHERE AccountID = @FromAccount;
        UPDATE Accounts SET Balance = @ToBalance + @Amount WHERE AccountID = @ToAccount;
    END;
END;
GO

CREATE FUNCTION GetAccountBalance
    （@AccountID INT）
RETURNS DECIMAL（10，2）
AS
BEGIN
    DECLARE @Balance DECIMAL（10，2）；
    SELECT @Balance = Balance FROM Accounts WHERE AccountID = @AccountID;
    RETURN @Balance;
END;
GO

DELIMITER;
```