**Exercise 7: Packages**

**Scenario 1:** Group all customer-related procedures and functions into a package.

**Question:** Create a package **CustomerManagement** with procedures for adding a new customer, updating customer details, and a function to get customer balance.

**PL/SQL Procedure:**

DELIMITER //

CREATE FUNCTION GetCustomerBalance (

p\_customerID INT

)

RETURNS DECIMAL(10,2)

DETERMINISTIC

READS SQL DATA

BEGIN

DECLARE balance DECIMAL(10,2);

SELECT Balance INTO balance FROM Customers WHERE CustomerID = p\_customerID;

RETURN balance;

END;

//

DELIMITER ;

**Scenario 2:** Create a package to manage employee data.

**Question:** Write a package **EmployeeManagement** with procedures to hire new employees, update employee details, and a function to calculate annual salary.

**PL/SQL Procedure:**

DELIMITER //

CREATE FUNCTION CalculateAnnualSalary (

p\_employeeID INT

)

RETURNS DECIMAL(10,2)

DETERMINISTIC

READS SQL DATA

BEGIN

DECLARE salary DECIMAL(10,2);

SELECT Salary INTO salary FROM Employees WHERE EmployeeID = p\_employeeID;

RETURN salary \* 12;

END;

//

DELIMITER ;

**Scenario 3:** Group all account-related operations into a package.

**Question:** Create a package **AccountOperations** with procedures for opening a new account, closing an account, and a function to get the total balance of a customer across all accounts

**PL/SQL Procedure:**

DELIMITER //

CREATE FUNCTION GetTotalCustomerBalance (

p\_customerID INT

)

RETURNS DECIMAL(10,2)

DETERMINISTIC

READS SQL DATA

BEGIN

DECLARE total\_balance DECIMAL(10,2);

SELECT SUM(Balance) INTO total\_balance FROM Accounts WHERE CustomerID = p\_customerID;

RETURN total\_balance;

END;

//

DELIMITER ;

**Testing the Functions and Procedures:**

USE Bank;

CALL AddNewCustomer(3, 'Alice Wonderland', '1982-12-30', 2000);

SELECT GetCustomerBalance(3) AS CustomerBalance;