**Exercise 7: Packages**

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

CREATE PACKAGE CustomerManagement AS

PROCEDURE AddNewCustomer(

p\_customer\_id NUMBER,

p\_name VARCHAR2,

p\_dob DATE,

p\_balance NUMBER);

PROCEDURE UpdateCustomerDetails(

p\_customer\_id NUMBER,

p\_name VARCHAR2,

p\_dob DATE);

FUNCTION GetCustomerBalance(

p\_customer\_id NUMBER)

RETURN NUMBER;

END CustomerManagement;

CREATE PACKAGE BODY CustomerManagement AS

PROCEDURE AddNewCustomer(

p\_customer\_id NUMBER,

p\_name VARCHAR2,

p\_dob DATE,

p\_balance NUMBER)

AS

BEGIN

INSERT INTO customers (customer\_id, name, dob, balance, last\_modified)

VALUES (p\_customer\_id, p\_name, p\_dob, p\_balance, SYSDATE);

END AddNewCustomer;

PROCEDURE UpdateCustomerDetails(

p\_customer\_id NUMBER,

p\_name VARCHAR2,

p\_dob DATE)

AS

BEGIN

UPDATE customers

SET name = p\_name, dob = p\_dob

WHERE customer\_id = p\_customer\_id;

END UpdateCustomerDetails;

FUNCTION GetCustomerBalance(

p\_customer\_id NUMBER)

RETURN NUMBER

AS

v\_balance NUMBER;

BEGIN

SELECT balance INTO v\_balance

FROM customers

WHERE customer\_id = p\_customer\_id;

RETURN v\_balance;

END GetCustomerBalance;

END CustomerManagement;

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

CREATE PACKAGE EmployeeManagement AS

PROCEDURE HireNewEmployee(

p\_employee\_id NUMBER,

p\_name VARCHAR2,

p\_position VARCHAR2,

p\_salary NUMBER,

p\_department VARCHAR2,

p\_hire\_date DATE);

PROCEDURE UpdateEmployeeDetails(

p\_employee\_id NUMBER,

p\_name VARCHAR2,

p\_position VARCHAR2,

p\_salary NUMBER,

p\_department VARCHAR2);

FUNCTION CalculateAnnualSalary(

p\_employee\_id NUMBER)

RETURN NUMBER;

END EmployeeManagement;

CREATE PACKAGE BODY EmployeeManagement AS

PROCEDURE HireNewEmployee(

p\_employee\_id NUMBER,

p\_name VARCHAR2,

p\_position VARCHAR2,

p\_salary NUMBER,

p\_department VARCHAR2,

p\_hire\_date DATE)

AS

BEGIN

INSERT INTO employees (employee\_id, name, position, salary, department, hire\_date)

VALUES (p\_employee\_id, p\_name, p\_position, p\_salary, p\_department, p\_hire\_date);

END HireNewEmployee;

PROCEDURE UpdateEmployeeDetails(

p\_employee\_id NUMBER,

p\_name VARCHAR2,

p\_position VARCHAR2,

p\_salary NUMBER,

p\_department VARCHAR2)

AS

BEGIN

UPDATE employees

SET name = p\_name, position = p\_position, salary = p\_salary, department = p\_department

WHERE employee\_id = p\_employee\_id;

END UpdateEmployeeDetails;

FUNCTION CalculateAnnualSalary(

p\_employee\_id NUMBER)

RETURN NUMBER

AS

v\_salary NUMBER;

BEGIN

SELECT salary INTO v\_salary

FROM employees

WHERE employee\_id = p\_employee\_id;

RETURN v\_salary \* 12;

END CalculateAnnualSalary;

END EmployeeManagement;

**Scenario 3: Group all account-related operations into a package**CREATE PACKAGE AccountOperations AS

PROCEDURE OpenNewAccount(

p\_account\_id NUMBER,

p\_customer\_id NUMBER,

p\_account\_type VARCHAR2,

p\_balance NUMBER);

PROCEDURE CloseAccount(

p\_account\_id NUMBER);

FUNCTION GetTotalBalance(

p\_customer\_id NUMBER)

RETURN NUMBER;

END AccountOperations;

CREATE PACKAGE BODY AccountOperations AS

PROCEDURE OpenNewAccount(

p\_account\_id NUMBER,

p\_customer\_id NUMBER,

p\_account\_type VARCHAR2,

p\_balance NUMBER)

AS

BEGIN

INSERT INTO accounts (account\_id, customer\_id, account\_type, balance, last\_modified)

VALUES (p\_account\_id, p\_customer\_id, p\_account\_type, p\_balance, SYSDATE);

END OpenNewAccount;

PROCEDURE CloseAccount(

p\_account\_id NUMBER)

AS

BEGIN

UPDATE accounts

SET status = 'Closed'

WHERE account\_id = p\_account\_id;

END CloseAccount;

FUNCTION GetTotalBalance(

p\_customer\_id NUMBER)

RETURN NUMBER

AS

v\_total\_balance NUMBER;

BEGIN

SELECT SUM(balance) INTO v\_total\_balance

FROM accounts

WHERE customer\_id = p\_customer\_id;

RETURN v\_total\_balance;

END GetTotalBalance;

END AccountOperations;\_id NUMBER)