**Packages**

**Scenario 1:**

CREATE OR REPLACE PACKAGE CustomerManagement IS

  PROCEDURE AddCustomer(cid NUMBER, nam VARCHAR2, dob DATE, bal NUMBER);

  PROCEDURE UpdateCustomer(cid NUMBER, nam VARCHAR2);

  FUNCTION GetBalance(cid NUMBER) RETURN NUMBER;

END;

/

CREATE OR REPLACE PACKAGE BODY CustomerManagement IS

  PROCEDURE AddCustomer(cid NUMBER, nam VARCHAR2, dob DATE, bal NUMBER) IS

  BEGIN

    INSERT INTO Customers (CustomerID, Name, DOB, Balance, LastModified)

    VALUES (cid, nam, dob, bal, SYSDATE);

  END;

  PROCEDURE UpdateCustomer(cid NUMBER, nam VARCHAR2) IS

  BEGIN

    UPDATE Customers

    SET Name = nam

    WHERE CustomerID = cid;

  END;

  FUNCTION GetBalance(cid NUMBER) RETURN NUMBER IS

    bal NUMBER;

  BEGIN

    SELECT Balance INTO bal FROM Customers WHERE CustomerID = cid;

    RETURN bal;

  END;

END;

/

SELECT CustomerManagement.GetBalance(1) AS Balance FROM dual;

**Output:**

**Scenario 2:**

CREATE OR REPLACE PACKAGE EmployeeManagement IS

  PROCEDURE HireEmployee(eid NUMBER, ename VARCHAR2, pos VARCHAR2, sal NUMBER, dept VARCHAR2);

  PROCEDURE UpdateEmployee(eid NUMBER, new\_sal NUMBER);

  FUNCTION AnnualSalary(eid NUMBER) RETURN NUMBER;

END;

/

CREATE OR REPLACE PACKAGE BODY EmployeeManagement IS

  PROCEDURE HireEmployee(eid NUMBER, ename VARCHAR2, pos VARCHAR2, sal NUMBER, dept VARCHAR2) IS

  BEGIN

    INSERT INTO Employees (EmployeeID, Name, Position, Salary, Department, HireDate)

    VALUES (eid, ename, pos, sal, dept, SYSDATE);

  END;

  PROCEDURE UpdateEmployee(eid NUMBER, new\_sal NUMBER) IS

  BEGIN

    UPDATE Employees

    SET Salary = new\_sal

    WHERE EmployeeID = eid;

  END;

  FUNCTION AnnualSalary(eid NUMBER) RETURN NUMBER IS

    sal NUMBER;

  BEGIN

    SELECT Salary INTO sal FROM Employees WHERE EmployeeID = eid;

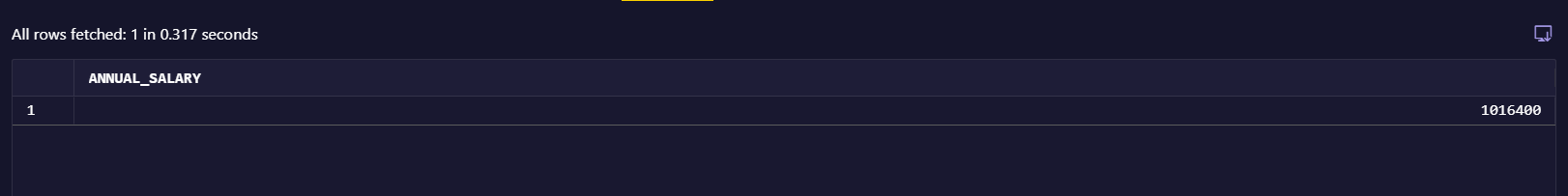
    RETURN sal \* 12;

  END;

END;

/

SELECT EmployeeManagement.AnnualSalary(1) AS Annual\_Salary FROM dual;

**Output:**

**Scenario 3:**

CREATE OR REPLACE PACKAGE AccountOperations IS

  PROCEDURE OpenAccount(acc\_id NUMBER, cust\_id NUMBER, acc\_type VARCHAR2, bal NUMBER);

  PROCEDURE CloseAccount(acc\_id NUMBER);

  FUNCTION TotalBalance(cust\_id NUMBER) RETURN NUMBER;

END;

/

CREATE OR REPLACE PACKAGE BODY AccountOperations IS

  PROCEDURE OpenAccount(acc\_id NUMBER, cust\_id NUMBER, acc\_type VARCHAR2, bal NUMBER) IS

  BEGIN

    INSERT INTO Accounts (AccountID, CustomerID, AccountType, Balance, LastModified)

    VALUES (acc\_id, cust\_id, acc\_type, bal, SYSDATE);

  END;

  PROCEDURE CloseAccount(acc\_id NUMBER) IS

  BEGIN

    DELETE FROM Accounts WHERE AccountID = acc\_id;

  END;

  FUNCTION TotalBalance(cust\_id NUMBER) RETURN NUMBER IS

    total\_bal NUMBER;

  BEGIN

    SELECT SUM(Balance) INTO total\_bal

    FROM Accounts

    WHERE CustomerID = cust\_id;

    RETURN NVL(total\_bal, 0);

  END;

END;

/

BEGIN

  CustomerManagement.AddCustomer(4, 'Elsa', TO\_DATE('2000-02-02', 'YYYY-MM-DD'), 6000);

END;

SELECT \* FROM Customers WHERE CustomerID = 4;

**Output:**

