**Exercise 2: Error Handling**

CREATE OR REPLACE PROCEDURE SafeTransferFunds (

    p\_from\_account\_id IN NUMBER,

    p\_to\_account\_id IN NUMBER,

    p\_amount IN NUMBER

)

IS

    v\_balance NUMBER;

BEGIN

    SELECT Balance INTO v\_balance

    FROM Accounts

    WHERE AccountID = p\_from\_account\_id;

    IF v\_balance < p\_amount THEN

        RAISE\_APPLICATION\_ERROR(-20001, 'Insufficient funds.');

    END IF;

    UPDATE Accounts

    SET Balance = Balance - p\_amount

    WHERE AccountID = p\_from\_account\_id;

    UPDATE Accounts

    SET Balance = Balance + p\_amount

    WHERE AccountID = p\_to\_account\_id;

    COMMIT;

    DBMS\_OUTPUT.PUT\_LINE('Funds transferred successfully.');

EXCEPTION

    WHEN OTHERS THEN

        ROLLBACK;

        DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);

END;

/

BEGIN

  SafeTransferFunds(1, 2, 200);

END;

/

CREATE OR REPLACE PROCEDURE UpdateSalary (

    p\_employee\_id IN NUMBER,

    p\_percentage IN NUMBER

)

IS

BEGIN

    UPDATE Employees

    SET Salary = Salary + (Salary \* p\_percentage / 100)

    WHERE EmployeeID = p\_employee\_id;

    IF SQL%ROWCOUNT = 0 THEN

        RAISE\_APPLICATION\_ERROR(-20002, 'Employee not found.');

    END IF;

    COMMIT;

    DBMS\_OUTPUT.PUT\_LINE('Salary updated successfully.');

EXCEPTION

    WHEN OTHERS THEN

        ROLLBACK;

        DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);

END;

/

BEGIN

  UpdateSalary(1, 10);

  UpdateSalary(999, 10);

END;

/

CREATE OR REPLACE PROCEDURE AddNewCustomer (

    p\_customer\_id IN NUMBER,

    p\_name IN VARCHAR2,

    p\_dob IN DATE,

    p\_balance IN NUMBER

)

IS

BEGIN

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

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

    COMMIT;

    DBMS\_OUTPUT.PUT\_LINE('New customer added successfully.');

EXCEPTION

    WHEN DUP\_VAL\_ON\_INDEX THEN

        DBMS\_OUTPUT.PUT\_LINE('Error: Customer with this ID already exists.');

    WHEN OTHERS THEN

        DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);

END;

/

BEGIN

  AddNewCustomer(3, 'New Customer', TO\_DATE('1995-10-10', 'YYYY-MM-DD'), 2000);

  AddNewCustomer(1, 'Duplicate ID', TO\_DATE('1980-01-01', 'YYYY-MM-DD'), 3000);

END;

/

SELECT \* FROM Accounts;

SELECT \* FROM Employees;

SELECT \* FROM Customers;















