**Exercise 2 – Error Handling**

**Scenario 1: Handle Exceptions During Fund Transfers:**

CREATE OR REPLACE PROCEDURE SafeTransferFunds (

p\_FromAccountID IN NUMBER,

p\_ToAccountID IN NUMBER,

p\_Amount IN NUMBER

)

IS

e\_InsufficientFunds EXCEPTION;

v\_FromBalance NUMBER;

BEGIN

-- Check the balance of the from account

SELECT Balance INTO v\_FromBalance FROM Accounts WHERE AccountID = p\_FromAccountID FOR UPDATE;

-- Check if there are sufficient funds

IF v\_FromBalance < p\_Amount THEN

RAISE e\_InsufficientFunds;

END IF;

-- Perform the transfer

UPDATE Accounts SET Balance = Balance - p\_Amount WHERE AccountID = p\_FromAccountID;

UPDATE Accounts SET Balance = Balance + p\_Amount WHERE AccountID = p\_ToAccountID;

COMMIT;

EXCEPTION

WHEN e\_InsufficientFunds THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: Insufficient funds for transfer.');

WHEN OTHERS THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: An unexpected error occurred during the transfer.');

END SafeTransferFunds;

/

**Scenario 2: Manage Errors When Updating Employee Salaries:**

CREATE OR REPLACE PROCEDURE UpdateSalary (

p\_EmployeeID IN NUMBER,

p\_Percentage IN NUMBER

)

IS

e\_EmployeeNotFound EXCEPTION;

v\_CurrentSalary NUMBER;

BEGIN

-- Attempt to select the current salary

SELECT Salary INTO v\_CurrentSalary FROM Employees WHERE EmployeeID = p\_EmployeeID FOR UPDATE;

-- Update the salary

UPDATE Employees

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

WHERE EmployeeID = p\_EmployeeID;

COMMIT;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: Employee with ID ' || p\_EmployeeID || ' does not exist.');

WHEN OTHERS THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: An unexpected error occurred during the salary update.');

END UpdateSalary;

/

**Scenario 3: Ensure Data Integrity When Adding a New Customer:**

CREATE OR REPLACE PROCEDURE AddNewCustomer (

p\_CustomerID IN NUMBER,

p\_Name IN VARCHAR2,

p\_Age IN NUMBER,

p\_Balance IN NUMBER

)

IS

e\_CustomerAlreadyExists EXCEPTION;

BEGIN

-- Attempt to insert a new customer

INSERT INTO Customers (CustomerID, Name, Age, Balance)

VALUES (p\_CustomerID, p\_Name, p\_Age, p\_Balance);

COMMIT;

EXCEPTION

WHEN DUP\_VAL\_ON\_INDEX THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: A customer with ID ' || p\_CustomerID || ' already exists.');

WHEN OTHERS THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Error: An unexpected error occurred while adding the customer.');

END AddNewCustomer;

/