**Scenario 1:**

CREATE OR REPLACE PROCEDURE SafeTransferFunds(

p\_from\_account IN NUMBER,

p\_to\_account IN NUMBER,

p\_amount IN NUMBER

) IS

BEGIN

BEGIN

-- Start a transaction

SAVEPOINT before\_transfer;

-- Transfer funds

UPDATE Accounts

SET Balance = Balance - p\_amount

WHERE AccountID = p\_from\_account;

IF SQL%ROWCOUNT = 0 THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Source account not found.');

END IF;

IF (SELECT Balance FROM Accounts WHERE AccountID = p\_from\_account) < p\_amount THEN

RAISE\_APPLICATION\_ERROR(-20002, 'Insufficient funds.');

END IF;

UPDATE Accounts

SET Balance = Balance + p\_amount

WHERE AccountID = p\_to\_account;

IF SQL%ROWCOUNT = 0 THEN

RAISE\_APPLICATION\_ERROR(-20003, 'Destination account not found.');

END IF;

-- Commit the transaction

COMMIT;

EXCEPTION

WHEN OTHERS THEN

-- Rollback the transaction

ROLLBACK TO before\_transfer;

-- Log error message

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

END;

END;

/

**Scenario 2:**

CREATE OR REPLACE PROCEDURE UpdateSalary(

p\_employee\_id IN NUMBER,

p\_percentage IN NUMBER

) IS

BEGIN

UPDATE Employees

SET Salary = Salary \* (1 + p\_percentage / 100)

WHERE EmployeeID = p\_employee\_id;

IF SQL%ROWCOUNT = 0 THEN

RAISE\_APPLICATION\_ERROR(-20004, 'Employee ID does not exist.');

END IF;

COMMIT;

EXCEPTION

WHEN OTHERS THEN

-- Log error message

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

ROLLBACK;

END;

/

**Scenario 3:**

CREATE OR REPLACE PROCEDURE AddNewCustomer(

p\_customer\_id IN NUMBER,

p\_name IN VARCHAR2,

p\_balance IN NUMBER

) IS

BEGIN

BEGIN

INSERT INTO Customers (CustomerID, Name, Balance)

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

COMMIT;

EXCEPTION

WHEN DUP\_VAL\_ON\_INDEX THEN

-- Log error message

DBMS\_OUTPUT.PUT\_LINE('Customer with ID ' || p\_customer\_id || ' already exists.');

ROLLBACK;

WHEN OTHERS THEN

-- Log error message

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

ROLLBACK;

END;

END;

/