**1)**

**Scenario 1**

DECLARE

    CURSOR cust\_cursor IS

        SELECT CustomerID, LoanInterestRate

        FROM Customers

        WHERE Age > 60;

BEGIN

    FOR cust\_rec IN cust\_cursor LOOP

        UPDATE Customers

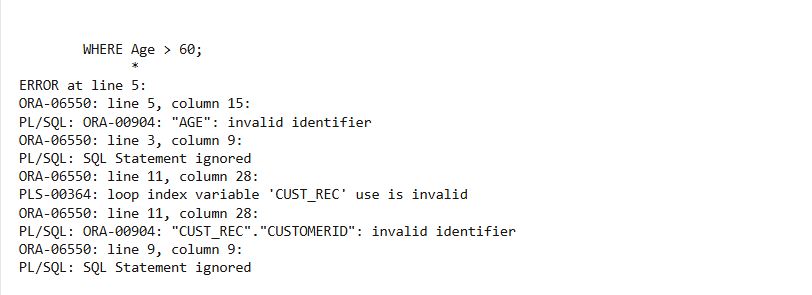
        SET LoanInterestRate = LoanInterestRate - 0.01

        WHERE CustomerID = cust\_rec.CustomerID;

    END LOOP;

    COMMIT;

END;



**Scenario 2**

ALTER TABLE Customers ADD (IsVIP VARCHAR2(5));

DECLARE

CURSOR vip\_candidates IS

SELECT CustomerID

FROM Customers

WHERE Balance > 10000;

BEGIN

FOR rec IN vip\_candidates LOOP

UPDATE Customers

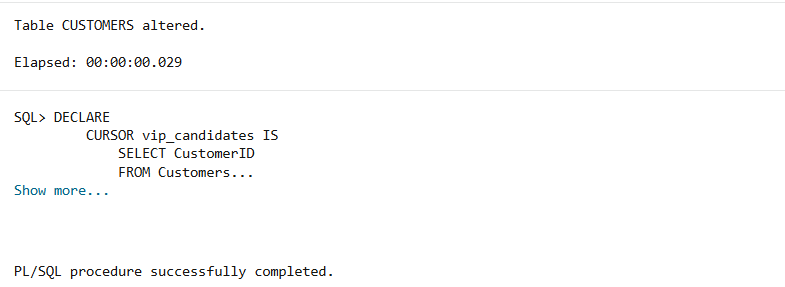
SET IsVIP = 'TRUE'

WHERE CustomerID = rec.CustomerID;

END LOOP;

COMMIT;

END;



**Scenario 3**

DECLARE

CURSOR due\_soon\_loans IS

SELECT l.LoanID, c.Name, l.EndDate

FROM Loans l

JOIN Customers c ON l.CustomerID = c.CustomerID

WHERE l.EndDate <= SYSDATE + 30;

BEGIN

FOR rec IN due\_soon\_loans LOOP

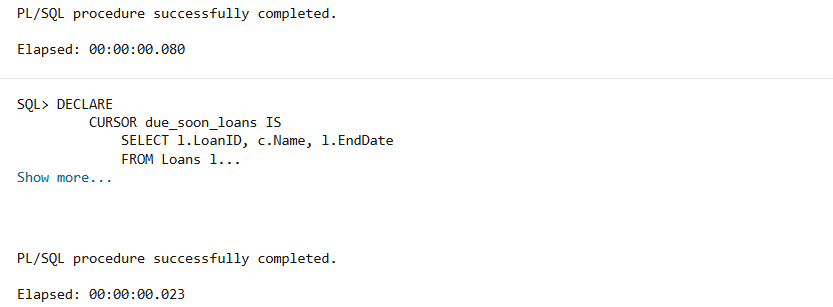
DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan ID ' || rec.LoanID ||

' for customer ' || rec.Name ||

' is due on ' || TO\_CHAR(rec.EndDate, 'DD-MON-YYYY'));

END LOOP;

END;



**3)**

**Scenario 1**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest AS

BEGIN

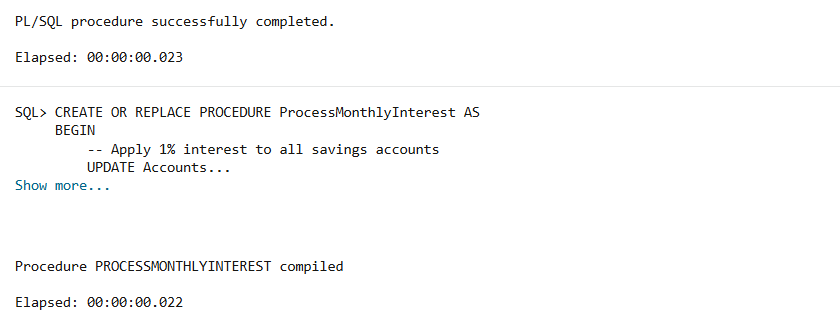
UPDATE Accounts

SET Balance = Balance + (Balance \* 0.01)

WHERE AccountType = 'Savings';

COMMIT;

END;



**Scenario 2**

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus(

p\_Department IN VARCHAR2,

p\_BonusPercent IN NUMBER

) AS

BEGIN

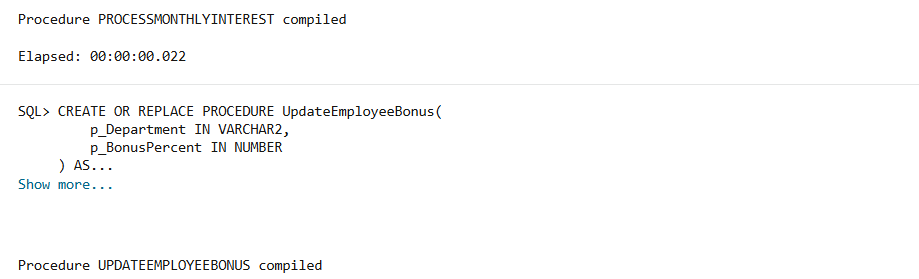
UPDATE Employees

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

WHERE Department = p\_Department;

COMMIT;

END;



**Scenario 3**

CREATE OR REPLACE PROCEDURE TransferFunds(

    p\_FromAccountID IN NUMBER,

    p\_ToAccountID IN NUMBER,

    p\_Amount IN NUMBER

) AS

    v\_FromBalance NUMBER;

BEGIN

    SELECT Balance INTO v\_FromBalance

    FROM Accounts

    WHERE AccountID = p\_FromAccountID;

    IF v\_FromBalance < p\_Amount THEN

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

    END IF;

    UPDATE Accounts

    SET Balance = Balance - p\_Amount

    WHERE AccountID = p\_FromAccountID;

    UPDATE Accounts

    SET Balance = Balance + p\_Amount

    WHERE AccountID = p\_ToAccountID;

    COMMIT;

END;

