**PL/SQL PROGRAMMING:**

**Exercise 1: Control Structures**

**Scenario 1: Apply 1% Discount to Interest Rate if Age > 60**

BEGIN

FOR cust\_rec IN (SELECT customer\_id FROM CUSTOMERS WHERE age > 60) LOOP

UPDATE LOANS

SET interest\_rate = interest\_rate - 1

WHERE customer\_id = cust\_rec.customer\_id;

END LOOP;

COMMIT;

END;

/

**Scenario 2: Promote to VIP if Balance > $10,000**

BEGIN

FOR cust\_rec IN (SELECT customer\_id FROM CUSTOMERS WHERE balance > 10000) LOOP

UPDATE CUSTOMERS

SET IsVIP = 'TRUE'

WHERE customer\_id = cust\_rec.customer\_id;

END LOOP;

COMMIT;

END;

/

**Scenario 3: Send Reminders for Loans Due in Next 30 Days**

DECLARE

v\_due\_date DATE;

v\_customer\_id NUMBER;

BEGIN

FOR loan\_rec IN (

SELECT customer\_id, due\_date

FROM LOANS

WHERE due\_date BETWEEN SYSDATE AND SYSDATE + 30

) LOOP

DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan for Customer ID ' || loan\_rec.customer\_id ||

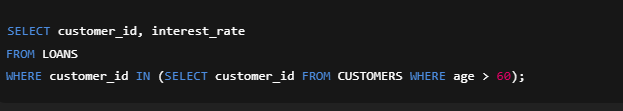
' is due on ' || TO\_CHAR(loan\_rec.due\_date, 'DD-MON-YYYY'));

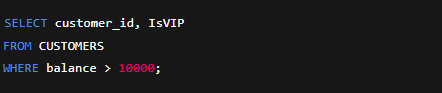
END LOOP;

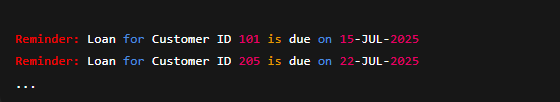
END;

/

**OUTPUT:**

****

****

****

**Exercise 3: Stored Procedures**

**Scenario 1: Apply 1% Interest to Savings Accounts**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

UPDATE Accounts

SET balance = balance + (balance \* 0.01)

WHERE account\_type = 'SAVINGS';

COMMIT;

END;

/

Scenario 2: UpdateEmployeeBonus – Apply Bonus to a Department

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus(

p\_department\_id IN NUMBER,

p\_bonus\_percent IN NUMBER

) IS

BEGIN

UPDATE Employees

SET salary = salary + (salary \* (p\_bonus\_percent / 100))

WHERE department\_id = p\_department\_id;

COMMIT;

END;

/

**Scenario 3: Transfer Money Between Accounts**

CREATE OR REPLACE PROCEDURE TransferFunds(

p\_from\_account IN NUMBER,

p\_to\_account IN NUMBER,

p\_amount IN NUMBER

) IS

v\_balance NUMBER;

BEGIN

SELECT balance INTO v\_balance

FROM Accounts

WHERE account\_id = p\_from\_account

FOR UPDATE;

IF v\_balance < p\_amount THEN

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

END IF;

UPDATE Accounts

SET balance = balance - p\_amount

WHERE account\_id = p\_from\_account;

UPDATE Accounts

SET balance = balance + p\_amount

WHERE account\_id = p\_to\_account;

COMMIT;

END;

/

**OUTPUT:**

****

****

****