WEEK – 2

**Exercise 1: Control Structures**

**Scenario 1:**

DECLARE

CURSOR c\_customers IS

SELECT id, interestRate FROM customers WHERE age>60;

BEGIN

OPEN c\_customers;

FOR c\_customer IN c\_customers LOOP

UPDATE customers

SET interestRate = c\_customer.interestRate - (c\_customer.interestRate/100)

WHERE id = c\_customer.id;

DBMS\_OUTPUT.PUT\_LINE('1% discuount applied to all customers who are of age above 60');

END LOOP;

COMMIT;

END;

/

**Scenario 2:**

DECLARE

CURSOR c\_customers IS

SELECT id, IsVIP FROM customers where balance >= 10000;

BEGIN

OPEN c\_customers;

FOR c\_customer IN c\_customers LOOP

UPDATE customers

SET IsVIP = 'Y'

WHERE id = c\_customer.id;

DBMS\_OUTPUT.PUT\_LINE('VIP Status Upgraded');

END LOOP;

COMMIT;

END;

/

**Scenario 3:**

DECLARE

CURSOR c\_customers IS

SELECT id, name, loanId, dueDate FROM customers where dueDate BETWEEN SYSDATE AND SYSDATE+30;

BEGIN

OPEN c\_customers;

FOR c\_customer IN c\_customers LOOP

DBMS\_OUTPUT.PUT\_LINE('This is a remainder message for '|| c\_customer.name ||

' - ' || c\_customer.id || ' regarding that your Loan ' || c\_customer.loanId ||

'is due on ' || c\_customer.dueDate);

END LOOP;

END;

/

**Exercise 3: Stored Procedures**

**Scenario 1:**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

UPDATE customers SET balance = balance + (balance/100) WHERE accountType = 'SAVINGS';

DBMS\_OUTPUT.PUT\_LINE('Savings Account Balance Updated successfully with 1% interest.');

COMMIT;

END ProcessMonthlyInterest;

/

**Scenario 2:**

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus(bonusPercent IN NUMBER,

departmentId IN NUMBER) IS

BEGIN

UPDATE employees SET salary = salary + (salary \* (bonusPercent/100))

WHERE deptID = departmentId;

DBMS\_OUTPUT.PUT\_LINE('Salary of employees in department ' || departmentId ||

' has been updated with ' || bonusPercent || '% of bonus to their current salary.');

COMMIT;

END UpdateEmployeeBonus;

/

**Scenario 3:**

**CREATE OR REPLACE PROCEDURE TransferFunds(srcAccount IN NUMBER,**

**destAccount IN NUMBER, amount IN NUMBER) IS**

**srcBalance NUMBER;**

**BEGIN**

**SELECT balance INTO srcBalance FROM accounts WHERE id = srcAccount FOR UPDATE;**

**IF srcBalance >= amount THEN**

**UPDATE accounts SET balance = balance - amount WHERE id = srcAccount;**

**UPDATE accounts SET balance = balance + amount WHERE id = destAccount;**

**DBMS\_OUTPUT.PUT\_LINE('Amount Transferred Successfully from ' || srcAccount ||**

**' to ' || destAccount);**

**COMMIT;**

**ELSE**

**DBMS\_OUTPUT.PUT\_LINE('Insufficient funds on ' || srcAccount);**

**END IF;**

**END TransferFunds;**

**/**