NAME

MOHAMMAD SHAAD

REG. NO.

21BCE1542

SUBJECT

DATABASE MANAGEMENT SYSTEM

EXERCISE

SIX

```
CREATE TABLE EMP21BCE1542 (EMPNO NUMBER(4), ENAME
VARCHAR(30), JOB VARCHAR2(15), SAL NUMBER(8), DEPTNO
NUMBER(2), COMMISSION NUMBER(7) );


INSERT INTO EMP21BCE1542 VALUES (1005,'SHAAD','FULL
STACK DEV',60000,10,526);
```

I. (a)

```
DECLARE
  CURSOR emp_cur IS
    SELECT Empno, Ename, Job
    FROM EMP21BCE1542
    WHERE DeptNo = 10;
BEGIN
  FOR emp_rec IN emp_cur LOOP
    DBMS_OUTPUT.PUT_LINE(emp_rec.Empno || ' ' ||
emp_rec.Ename || ' ' || emp_rec.Job);
  END LOOP;
END;
/
```

(b)

```
DECLARE
  CURSOR sal_cur IS
    SELECT Empno, Sal, DeptNo
    FROM EMP21BCE1542;
  v_empno emp21bce1542.empno%TYPE;
  v_old_sal emp21bce1542.sal%TYPE;
  v_new_sal emp21bce1542.sal%TYPE;
BEGIN
  FOR sal_rec IN sal_cur LOOP
    v_empno := sal_rec.Empno;
    v_old_sal := sal_rec.Sal;
    IF sal_rec.DeptNo = 10 THEN
      v_new_sal := v_old_sal + 1000;
    ELSIF sal_rec.DeptNo = 20 THEN
      v_new_sal := v_old_sal + 500;
    ELSIF sal_rec.DeptNo = 30 THEN
      v_new_sal := v_old_sal + 800;
    END IF;
    UPDATE EMP21BCE1542
    SET Sal = v_new_sal
    WHERE Empno = v_empno;
    INSERT INTO TEMP (Empid, Old, New)
    VALUES (v_empno, v_old_sal, v_new_sal);
  END LOOP;
END;
/
```

(c)

```
DECLARE
  v_avg_sal emp21bce1542.sal%TYPE;
CURSOR emp_cur IS
  SELECT Ename, Sal
  FROM EMP21BCE1542
  WHERE Sal < v_avg_sal;
BEGIN
  SELECT AVG(Sal) INTO v_avg_sal
  FROM EMP21BCE1542;
  FOR emp_rec IN emp_cur LOOP
    DBMS_OUTPUT.PUT_LINE(emp_rec.Ename || ' ' ||
emp_rec.Sal);
  END LOOP;
END;
/
```

II. (a)

```
CREATE OR REPLACE TRIGGER sal_update_trigger
BEFORE UPDATE ON EMP21BCE1542
FOR EACH ROW
BEGIN
  IF :new.Sal <= :old.Sal THEN
    RAISE_APPLICATION_ERROR(-20001, 'Salary cannot be
decreased');
  END IF;
END;
/
```

(b)

```
CREATE OR REPLACE TRIGGER dept_emp_limit_trigger
BEFORE INSERT ON EMP21BCE1542
FOR EACH ROW
DECLARE
  v_emp_count NUMBER;
BEGIN
  SELECT COUNT(*) INTO v_emp_count
  FROM EMP21BCE1542
  WHERE DeptNo = 2;
  IF v_emp_count >= 5 THEN
    RAISE_APPLICATION_ERROR(-20002, 'Cannot add
employee, department employee limit exceeded');
  END IF;
END;
/
```

(c)

```
CREATE OR REPLACE TRIGGER negative_balance_trigger
AFTER UPDATE ON Account
FOR EACH ROW
WHEN (NEW.balance < 0)
DECLARE
    loan_no Account.acctno%TYPE := :NEW.acctno;
    br_name Account.br_name%TYPE := :NEW.br_name;
    loan_amount Loan.amount%TYPE := -1 *
:NEW.balance;
BEGIN
    -- Insert new tuple into Loan table
    INSERT INTO Loan (loan_no, br_name, amount)
VALUES (loan_no, br_name, loan_amount);

    -- Insert new tuple into Borrower table
    INSERT INTO Borrower (custname, loan_no) VALUES
('Jones', loan_no);

    -- Update the balance in Account table
    UPDATE Account SET balance = 0 WHERE acctno =
loan_no;

    DBMS_OUTPUT.PUT_LINE('Trigger executed
successfully!');
END;
/
```