Practical Assignment - 3

Q-1. Create PL/SQL Blocks
Simple programming:

2. Write a block for declaring variable A with value 50 add 20 in A and display the message "The value of A = 70".

→
```
DECLARE
        a int := 50;
BEGIN
        a := a + 20;
        DBMS_OUTPUT.PUT_LINE('The ~~sum~~ value of A =' || a);
END;
/
```

4. Write a block to take input of two variables and ~~display the same string on screen.~~ average of two number

→
```
DECLARE
        a int;
        b int;
        avg int;
BEGIN
        a := &a;
        b := &b;
        avg := (a+b) / 2;
        DBMS_OUTPUT.PUT_LINE('The average of
                two number is ' || avg);
END;
/
```

vision

6. Write a block to take input of Rollno, Name, Pe
card birthdate. Display each in seprate line wi
proper heading.

→
```
DECLARE
        rno int;
        name varchar2(50);
        per float;
        dob date;
BEGIN
        rno := &rno;
        name := '&name';
        per := &per;
        dob := '&dob';
    DBMS_OUTPUT.PUT_LINE('Rno = ' || rno);
    DBMS_OUTPUT.PUT_LINE('Name = ' || name);
    DBMS_OUTPUT.PUT_LINE('Per = ' || per);
    DBMS_OUTPUT.PUT_LINE('DOB = ' || dob);
END;
/
```

8. Write a block to take input of radius from user and display the area of circle.

→
```
DECLARE
    r float;
    aoc float;
BEGIN
    r := &r;
    aoc := 3.14 * (r*r);
    DBMS_OUTPUT.PUT_LINE('Area of circle is ' || aoc);
END;
/
```

10. Write a block to take input of laptop price, quantity and discount. Display the total amount and payble amount (after deduction of discount).

→
```
DECLARE
        price int;
        qty  int;
        dis  int;
BEGIN
        price := &price;
        qty := &qty;
        dis := &dis;
        DBMS_OUTPUT.PUT_LINE ('The total amount is'
                || price * qty);
        DBMS_OUTPUT.PUT_LINE ('The total payble amount
        is' || (price * qty) - ((price * qty * dis) / 100));
END;
/
```

Q.2 Programming with database

* Create tblemp and tbldept

→ CREATE TABLE tbldept (
did VARCHAR2(10) PRIMARY KEY,
dname VARCHAR2(50)
);

→ CREATE TABLE tblemp (
empid INT PRIMARY KEY,
fname VARCHAR2(50) NOT NULL,
lname VARCHAR2(50),
jdate DATE,
salary INT
manager_id INT,
post VARCHAR2(50),
dept_id VARCHAR2(10),
FOREIGN KEY (dept_id) REFERENCES tbldept(did)
);

* Inserting records

→ INSERT INTO tbldept (did, dname) VALUES
('D1', 'Finance'),
('D2', 'Production'),
('D3', 'Marketing'),
('D4', 'IT');

→ INSERT INTO tblemp (empid, fname, lname, jdate, salary
manager_id, post, dept_id) VALUES
(101, 'Hemant', 'Sharma', '1995-01-20', 50000, 0, 'Manager', 'D1')

(102,'Ritu','Gandhi','20-12-1997',45000,0,'Manager','D2'),
(103,'Maya','Mistry','2002-11-12',30,000,101,'Programmer','D1'),
(104,'Riya','Patel','2003-09-15',28000,102,'Programmer','D2'),
(105,'Shreya','Patel','2010-10-17',5000,101,'Peon','D1'),
(106,'Karan','Patel','2015-08-18',5000,102,'Peon','D2'),
(107,'Reyanshi','Vyas','2011-06-05',15000,101,'Clerk','D1'),
(108,'Mehul','Mehta','2020-05-06',17000,102,'Clerk','D2'),
(109,'Kinjal','Patil','2015-04-08',10000,101,'Accountant','D1'),
(110,'Maitry','Vyas','2017-03-11',10000,102,'Accountant','D2'),
(111,'Mohan','Mehta','2020-02-27',15000,101,'Electrician','D1');

11 Display the fname of employee whose EID is 107.

→
```
DECLARE
    vfname tblemp.fname%TYPE;
BEGIN
    SELECT fname INTO vfname from tblemp
    WHERE empid = 107;
    DBMS_OUTPUT.PUT_LINE('Fname : ' || vfname);
END;
/
```

13. Display the joining date of employee whose surname is Patil.

→
```
DECLARE
    vjdate tblemp.jdate%TYPE;
BEGIN
    SELECT jdate INTO vjdate FROM tblemp
    WHERE lname = 'Patil';
    DBMS_OUTPUT.PUT_LINE('Joining date : ' || vjdate);
END;
/
```

15. Display the detail of employee whose joining date is 15th September 2003.

→
```
DECLARE
        emp tblemp%ROWTYPE;
BEGIN
        SELECT * INTO emp FROM tblemp
        WHERE jdate = TO_DATE('15-SEP-2003','DD-MON-YYYY');

    DBMS_OUTPUT.PUT_LINE('Emp ID :' || emp.empid);
    DBMS_OUTPUT.PUT_LINE('Fname :' || emp.fname);
    DBMS_OUTPUT.PUT_LINE('Lname :' || emp.lname);
    DBMS_OUTPUT.PUT_LINE('Jdate :' || emp.jdate);
    DBMS_OUTPUT.PUT_LINE('Salary :' || emp.salary);
    DBMS_OUTPUT.PUT_LINE('Manage ID :' || emp.manager_id);
    DBMS_OUTPUT.PUT_LINE('Post :' || emp.post);
    DBMS_OUTPUT.PUT_LINE('Dept. ID :' || emp.dept_id);
END;
/
```

17. Display the detail of employee whose EID is inputted by user.

→
```
DECLARE
    e tblemp%ROWTYPE;
    eid int;
BEGIN
    eid := &Eid;
    SELECT * INTO e FROM tblemp WHERE empid=eid;

    DBMS_OUTPUT.PUT_LINE('Empid :' || e.empid);
    DBMS_OUTPUT.PUT_LINE('Fname :' || e.fname);
    DBMS_OUTPUT.PUT_LINE('Lname :' || e.lname);
```

```
DBMS_OUTPUT.PUT_LINE('Jdate :' || e.jdate);
DBMS_OUTPUT.PUT_LINE('Salary :' || e.salary);
DBMS_OUTPUT.PUT_LINE('Manager id :' || e.manager_id);
DBMS_OUTPUT.PUT_LINE('Post :' || e.post);
DBMS_OUTPUT.PUT_LINE('Dept id :' || e.dept_id);
END;
/
```

19. Display the eid, fname, salary and dname of the employee whose eid is given by the user.

→
```
DECLARE
    veid    tblemp.empid%TYPE;
    vfname  tblemp.fname%TYPE;
    vsalary tblemp.salary%TYPE;
    vdname  tbldept.dname%TYPE;
BEGIN
    veid := &eid;
    SELECT e.fname, e.salary, d.dname
    INTO veid, vfname, vsalary, vdname
    FROM tblemp e
    JOIN tbldept d ON e.dept_id = d.did
    WHERE e.empid = veid;

    DBMS_OUTPUT.PUT_LINE('Emp id :' || veid);
    DBMS_OUTPUT.PUT_LINE('Fname :' || vfname);
    DBMS_OUTPUT.PUT_LINE('salary :' || vsalary);
    DBMS_OUTPUT.PUT_LINE('Dname :' || vdname);
END;
/
```

vision

Q.3 Programming with conditional statements:

23. Take input of single number and display whether number is even or odd.

→
```
DECLARE
    n int;
BEGIN
    n := &number;
    IF mod(n,2) = 0 THEN
        DBMS_OUTPUT.PUT_LINE('EVEN');
    ELSE
        DBMS_OUTPUT.PUT_LINE('odd');
    END IF;
END;
/
```

24. Take input of a year and check whether it is a leap year or not

→
```
DECLARE
    year NUMBER;
BEGIN
    year := &year;
    IF mod(year, 4) = 0 THEN
        DBMS_OUTPUT.PUT_LINE('Leap year');
    ELSE
        DBMS_OUTPUT.PUT_LINE(' is not Leap year');
    END IF;
END;
/
```

26. Take input of two number and perform the addition, subtraction, multiplication or division as per user choice.

→ ex 1 for addition
2 for subtraction
3 for Multiplication
4 for Division
otherwise Wrong Input

```
DECLARE
    n1 NUMBER;
    n2 NUMBER;
    c  NUMBER;
BEGIN
    n1 := &Enter_num_1;
    n2 := &Enter_num_2;
    c := &Enter_choice;

  CASE c
    WHEN 1 THEN
      DBMS_OUTPUT.PUT_LINE('A+B=' || n1+n2);
    WHEN 2 THEN
      DBMS_OUTPUT.PUT_LINE('A-B=' || n1-n2);
    WHEN 3 THEN
      DBMS_OUTPUT.PUT_LINE('A*B=' || n1*n2);
    WHEN 4 THEN
      DBMS_OUTPUT.PUT_LINE('A/B=' || A/B n1/n2);
    ELSE
      DBMS_OUTPUT.PUT_LINE('Wrong Input.');
    END CASE;
END;
/
```

Q_4 Programming with database

27. Display the employee detail of given EID if his salary is less than 20000

→
```
DECLARE
    ej tblemp1%.ROWTYPE;
    veid tblemp1.empid%.TYPE;
BEGIN
    veid:=&eid;
    SELECT * INTO ej FROM tblemp1
    WHERE empid = veid &AND salary < 20000;

    DBMS_OUTPUT.PUT_LINE('Eid :' || ej.empid);
    DBMS_OUTPUT.PUT_LINE('Fname :' || ej.fname);
    DBMS_OUTPUT.PUT_LINE('Lname :' || ej.lname);
    DBMS_OUTPUT.PUT_LINE('Salary :' || ej.salary);
    DBMS_OUTPUT.PUT_LINE('Post :' || ej.post);
    DBMS_OUTRT.PUT_LINE('Dept ID :' || ej.dept_id);
END;
/
```

29. Display the employee detail of Peon of D2 department if his salary is more than 8000

→
```
DECLARE
    CURSOR peon_cursor IS
    SELECT * FROM tblemp1
    WHERE post = 'Peon'
    AND dept_id = 'D2'
    AND salary > 8000;
    ej tblemp1%.ROWTYPE;
BEGIN
```

vision

```
OPEN peon_cursor;
LOOP
    FETCH peon_cursor INTO ej;
    EXIT WHEN peon_cursor%.NOTFOUND;

    DBMS_OUTPUT.PUT_LINE('Eid :' || ej.empid ||
                         ',Name:' || ej.fname ||
                         'Salary:' || ej.salary ||
                         'Post:' || ej.post ||
                         'Dept:' || ej.dept_id);
    END LOOP;
    CLOSE peon_cursor;
END;
/
```

31. Update the salary of given eid by adding 5000 if his joining date is in year 2015.

→
```
DECLARE
    e tblempl%.ROWTYPE;
    veid tblempl.empid%.TYPE;
BEGIN
    veid := &eid;
    SELECT * INTO e FROM tblempl
    WHERE empid = veid AND
    TO_CHAR(jdate,'YYYY') = '2015';

    UPDATE tblempl
    SET salary = salary + 5000
    WHERE empid = veid
END;
/
```

33. Display the employee details with department name of a given fname if they are an Accountant

→
```
DECLARE
    e tblempl%.ROWTYPE;
    vdept tbldept.dname%TYPE;
    vfname tblempl.fname%TYPE;
BEGIN
    vfname := '&fname';
    SELECT tblempl.*, tbldept.dname INTO e, vdept
    FROM tblempl
    JOIN tbldept ON tblempl.dept_id = tbldept.did
    WHERE tblempl.fname = vfname AND
     tblempl.post = 'Accountant';

    DBMS_OUTPUT.PUT_LINE('Eid:' || e.empid);
    DBMS_OUTPUT.PUT_LINE('Fname:' || e.vfname);
    DBMS_OUTPUT.PUT_LINE('Lname:' || e.lname);
    DBMS_OUTPUT.PUT_LINE('Jdate:' || e.jdate);
    DBMS_OUTPUT.PUT_LINE('Salary:' || e.salary);
    DBMS_OUTPUT.PUT_LINE('Post:' || e.post);
    DBMS_OUTPUT.PUT_LINE('Manager ID:' || e.manager_id);
    DBMS_OUTPUT.PUT_LINE('Dept:' || vdept);
END;
/
```

35. Delete the record of given eid if it belongs to the production

→
```
DECLARE
    vdept tbldept.dname%TYPE;
    veid tblempl.empid%TYPE;
BEGIN

    WHERE empid = veid;
```

vision

vision

```
veid := &eid;
SELECT tbldept.dname INTO vdept FROM tblempl
JOIN tbldept ON tblempl.dept_id = tbldept.did
WHERE tblempl.empid = veid;

IF vdept = 'Production' THEN
    DELETE FROM tblempl WHERE empid = veid;
END IF;
END;
/
```

36. Update the employee's salary of given EID as per following criteria:

salary < 10000	add 5000
< 30000	7000
< 40000	9000
otherwise	10,000

→
```
DECLARE
    vsalary tblempl.salary%TYPE;
    veid tblempl.empid%TYPE;
BEGIN
    veid := &veid;
    SELECT salary INTO vsalary FROM tblempl
    WHERE empid = veid;

    IF vsalary < 10000 THEN
        UPDATE tblempl SET salary = salary + 5000
        WHERE empid = veid;
    ELSEIF vsalary < 30000 THEN
        UPDATE tblempl SET salary = salary + 7000
        WHERE empid = veid;
```

```
    ELSE IF salary < 40000 THEN
        UPDATE tblempl SET salary = salary + 9000
        WHERE empid = veid;
    ELSE
        UPDATE tblempl SET salary = salary + 10000
        WHERE empid = veid;
    END IF;
END;
/
```

Q.5 Database Programming

46. Employees of D1 Department

```
DECLARE
    CURSOR emp IS
    SELECT * FROM tblemp WHERE dept_id = 'D1';
    e1 tblemp%ROWTYPE;
BEGIN
    OPEN emp;
    LOOP
        FETCH emp INTO e1;
        EXIT WHEN emp%NOTFOUND;
        DBMS_OUTPUT.PUT_LINE ('Eid :' || e1.empid ||
                              'Fname :' || e1.fname ||
                              'Lname :' || e1.lname ||
                              'Jdate :' || e1.jdate ||
                              'Salary :' || e1.salary ||
                              'Manager id :' || e1.Manager_id ||
                              'Post :' || e1.post ||
                              'Dept id :' || e1.dept_id );
    END LOOP;
    CLOSE emp;
END;
/
```

48. Display employee's detail of Production department

```
DECLARE
    CURSOR emp IS
    SELECT e.* FROM tblemp e
    JOIN tbldept d ON e.dept_id = d.did
    WHERE d.dname = 'Production';
```

```
    ej  tblemp%.ROWTYPE;
BEGIN
    OPEN emp;
    LOOP
        FETCH emp INTO ej;
        EXIT WHEN emp%.NOTFOUND;
        DBMS_OUTPUT.PUT_LINE('Eid :' || ej.empid ||
                'Fname:' || ej.fname ||
                'Lname:' || ej.lname ||
                'Jdate:' || ej.jdate ||
                'Salary:' || ej.salary ||
                'Post :' || ej.post ||
                'Dept: Production');
    END LOOP;
    CLOSE emp;
END;
/
```

5o. Display employees detail whose surname is Patel

→

```
DECLARE
    CURSOR emp IS
    SELECT * FROM tblemp WHERE lname='Patel';
    e tblemp%.ROWTYPE;
BEGIN
    OPEN emp;
    LOOP
        FETCH emp INTO e;
        EXIT WHEN emp%.NOTFOUND;
        DBMS_OUTPUT.PUT_LINE('Eid :' || e.empid || 'Fname
          :' || e.fname || 'Lname:' || e.lname || 'Jdate :' ||
          e.jdate || 'salary:' || e.salary || 'Post :' || e.post);
```

```
        END LOOP;
        CLOSE emp;
END;
/
```

Q.6 Perform following on tblempl table

52. Delete records of peon employees.

→
```
BEGIN
    DELETE FROM tblempl WHERE POST = 'Peon';
    COMMIT;
END;
/
```

53. Update salary of Accountant by adding ₹2000 if their jdate is before 1st Jan 2016

→
```
DECLARE BEGIN
    UPDATE tblempl
    SET salary = salary + 2000
    WHERE post = 'Accountant' AND
    jdate < TO_DATE('01-JAN-2016','DD-MON-YYYY');
    COMMIT;
END;
/
```

54. Delet Pate employees whose salary is less than ₹10,000

→
```
BEGIN
    DELETE FROM tblempl
    WHERE lname = 'Patel' AND salary < 10000;
    COMMIT;
END;
/
```

55. Update salary based on given conditions

→ BEGIN

```
    UPDATE tblemp1
    SET salary =
        CASE
            WHEN salary < 10000 THEN
              salary = salary + 5000
            WHEN salary < 30000 THEN
              salary + 7000
            WHEN salary < 40000 THEN
              salary + 9000
            ELSE salary + 10000
    END;
    COMMIT;
END;
/
```