**PRACTICAL 4**

**Aim: Writing PL/SQL Blocks with basic programming constructs by including following.**

**A.If…then….ELSE, If…ELSIF…ELSE…END IF**

**4.1 Write a PL/SQL program to check whether number is less than 50**

**Program:**

SQL> declare

2 n number:=10;

3 begin

4 if(n<50)then

5 dbms\_output.put\_line('Number is less than 50');

6 end if;

7 dbms\_output.put\_line('The value of n is' ||n);

8 end;

9 /

**Output:**

Number is less than 50

The value of n is10

PL/SQL procedure successfully completed.

**4.2 Write a PL/SQL program to check number entered by user is less than 50.**

**Program:**

SQL> declare

2 n number;

3 begin

4 n:=&n;

5 if(n<50)then

6 dbms\_output.put\_line('Number is less than 50');

7 end if;

8 dbms\_output.put\_line('The value of n is' ||n);

9 end;

10 /

**Output:**

Enter value for n: 25

old 4: n:=&n;

new 4: n:=25;

Number is less than 50

The value of n is25

PL/SQL procedure successfully completed.

**4.3 Write a PL/SQL program to update salary of employee by 2000 if salary is less than or equal to20000**

**Program:**

SQL> create table emp\_anu(eid number(5),ename varchar2(30),salary number(5));

Table created.

SQL> insert into emp\_anu values(&eid, '&ename', &salary);

Enter value for eid: 1

Enter value for ename: kukki

Enter value for salary: 20000

old 1: insert into emp\_anu values(&eid, '&ename', &salary)

new 1: insert into emp\_anu values(1, 'kukki', 20000)

1 row created.

SQL> /

Enter value for eid: 2

Enter value for ename: monika

Enter value for salary: 15000

old 1: insert into emp\_anu values(&eid, '&ename', &salary)

new 1: insert into emp\_anu values(2, 'monika', 15000)

1 row created.

SQL> /

Enter value for eid: 3

Enter value for ename: krutika

Enter value for salary: 50000

old 1: insert into emp\_anu values(&eid, '&ename', &salary)

new 1: insert into emp\_anu values(3, 'krutika', 50000)

1 row created.

SQL> /

Enter value for eid: 4

Enter value for ename: rutu

Enter value for salary: 12000

old 1: insert into emp\_anu values(&eid, '&ename', &salary)

new 1: insert into emp\_anu values(4, 'rutu', 12000)

1 row created.

SQL> /

Enter value for eid: 5

Enter value for ename: pookie

Enter value for salary: 10000

old 1: insert into emp\_anu values(&eid, '&ename', &salary)

new 1: insert into emp\_anu values(5, 'pookie', 10000)

1 row created.

SQL> select\* from emp\_anu

2 /

EID ENAME SALARY

---------- ------------------------------ ----------

1 kukki 20000

2 monika 15000

3 krutika 50000

4 rutu 12000

5 pookie 10000

SQL> declare

2 id emp\_anu.eid%type:=1;

3 sal emp\_anu.salary%type;

4 begin

5 select salary into sal from emp\_anu where eid=id;

6 if(sal<=20000) then

7 update emp\_anu set salary=salary+2000 where eid=id;

8 dbms\_output.put\_line('salary updated');

9 end if;

10 end;

11 /

**Output:**

salary updated

PL/SQL procedure successfully completed.

SQL> select\* from emp\_anu

2 /

EID ENAME SALARY

---------- ------------------------------ ----------

1 kukki 22000

2 monika 15000

3 krutika 50000

4 rutu 12000

5 pookie 10000

**4.4 Write a PL/SQL program to update salary of employee by 2000 by user if salary is less than or equal to20000**

**Program:**

SQL> declare

2 id emp\_anu.eid%type;

3 sal emp\_anu.salary%type;

4 begin

5 id:=&id;

6 select salary into sal from emp\_anu where eid=id;

7 if(sal<=20000) then

8 update emp\_anu set salary=salary+2000 where eid=id;

9 dbms\_output.put\_line('salary updated');

10 end if;

11 end;

12 /

**Output:**

Enter value for id: 2

old 5: id:=&id;

new 5: id:=2;

salary updated

PL/SQL procedure successfully completed.

SQL> select\* from emp\_anu

2 /

EID ENAME SALARY

---------- ------------------------------ ----------

1 kukki 22000

2 monika 17000

3 krutika 50000

4 rutu 12000

5 pookie 10000

**B.Case statement.**

**4.5 write a pl/sql program to display which remark got.**

**Program:**

SQL> declare

2 grade char;

3 begin

4 grade:=&grade;

5 case grade

6 when 'A' then dbms\_output.put\_line('Exellent');

7 when 'B' then dbms\_output.put\_line('Very Good');

8 when 'C' then dbms\_output.put\_line('Well Done');

9 when 'D' then dbms\_output.put\_line('You Passed');

10 when 'F' then dbms\_output.put\_line('better try again');

11 else dbms\_output.put\_line('No Such Grade');

12 end case;

13 end;

14 /

**O/P:**

Enter value for grade: 'A'

old 4: grade:=&grade;

new 4: grade:='A';

Exellent

PL/SQL procedure successfully completed.

**4.6 IF..ELSIF..ELSE..END IF.**

**PROGRAM:**

SQL> DECLARE

2 a number(3) := 100;

3 BEGIN

4 IF ( a = 10 ) THEN

5 dbms\_output.put\_line('Value of a is 10' );

6 ELSIF ( a = 20 ) THEN

7 dbms\_output.put\_line('Value of a is 20' );

8 ELSIF ( a = 30 ) THEN

9 dbms\_output.put\_line('Value of a is 30' );

10 ELSE

11 dbms\_output.put\_line('None of the values is matching');

12 END IF;

13 dbms\_output.put\_line('Exact value of a is: '|| a );

14 END;

15 /

**O/P:**

None of the values is matching

Exact value of a is: 100

PL/SQL procedure successfully completed.