CYCLE 2 – PL/SQL

Exp No:12
PL/SQL Introduction.

AIM: To implement various various control structures like IF-THEN,IF-THEN-ELSE,IF-THEN ELSIF,CASE ,WHILE USING PL/SQL

Procedural Language/Structured Query Language (PL/SQL) is an extension of SQL

Basic Syntax of PL/SQL

```
DECLARE
/* Variables can be declared here */
BEGIN
/* Executable statements can be written here */
EXCEPTION
/* Error handlers can be written here. */
END;
```

As we want output of PL/SQL Program on screen, before Starting writing anything type (Only Once per session)

```
SET SERVEROUTPUT ON
```

Sample 1: Hello World Program

```
DECLARE
   age integer;
   name VARCHAR(20);
BEGIN
   dbms_output.put_line('Hello world');
END;
/
```

Sampe 2: **Find the largest of two integers.**

(Use of “If else” in PL/SQL)

```
DECLARE
   a integer := &a;
   b integer := &b;
BEGIN
  if (a > b) then
      dbms_output.put_line(a || ' is the largest number');
  else
      dbms_output.put_line(b || ' is the largest number');
  end if;
END;
```

Sample 3: **Print the range of two integers.**

(Use of if elsif ladder)

```
DECLARE
   c integer := &c;
BEGIN
   if (c >= 0 and c < 10) then
      dbms_output.put_line(' is less than 10');
   elsif (c >= 10 and c < 20) then
      dbms_output.put_line(' is less than 20');
   elsif (c >= 20 and c < 30) then
      dbms_output.put_line(' is less than 30');
   else
      dbms_output.put_line(' is grater than or equal 30');
   end if;
END;
```

***Sample 4*: Print the performance rating.**

(Use of case statement)

```
DECLARE
   c char(1) := '&c';
BEGIN
   case c
      when 'A' then dbms_output.put_line('Excellent');
      when 'B' then dbms_output.put_line('Very good');
      when 'C' then dbms_output.put_line('Well done');
      when 'D' then dbms_output.put_line('You passed');
      when 'F' then dbms_output.put_line('Better try again');
      else dbms_output.put_line('No such grade');
   end case;
END;
```

Sample 5: Use of Array and Loops in PL/SQL

Please note:

- ***Default index starts from 1***
- **Declared using the TYPE keyword**

```
DECLARE
   type intArray IS VARRAY(10) OF INTEGER;
   type namesArray IS VARRAY(5) OF VARCHAR2(5);

   arr intArray;
   names namesArray;
   i integer;

BEGIN
   arr :=  intArray(1,5,2,3,6,7,4,8,9,10);
   names := namesArray('Alice', 'Bob', 'Cindy', 'Sam', 'Eric');

   i := 1;
```

```
   /*While loop...! */
   while( i <= 10) loop
      dbms_output.put_line('arr[' || i ||'] =' ||arr(i));
      i := i+1;
   end loop;

   /*For loop...! */
   for i in 1 .. 10 loop
      dbms_output.put_line('arr[' || i ||'] =' ||arr(i));
   end loop;
  /* while loop */
   i := 1;
   while( i <= 5) loop
      dbms_output.put_line('names[' || i ||'] =' ||names(i));
      i := i+1;
   end loop;

   /*For loop...! */
   for i in 1 .. 5 loop
      dbms_output.put_line('names[' || i ||'] =' ||names(i));
   end loop;
END;
```

Question 1: Write a plsql program to check whether a given number is ODD or EVEN

```
DECLARE
   number integer;
BEGIN
   -- get role no from user
   number := &number;

   --TODO calculate & print result

END;
/
```

Question 2: Write a PL/SQL block to find the maximum number from given three numbers.

```
DECLARE
   number1 integer := &number1;
   number2 integer := &number2;
   number3 integer := &number3;

BEGIN
   --TODO calculate & print result

END;
/
```

Question 3: Write a program to accept a number and find the sum of the digits

```
DECLARE
   num integer :=&num;
   total integer:=0;
   digit integer:=0;
BEGIN
   --TODO calculate & print result
END;
/
```

Question 4: Write a program to accept a number and find the sum of the digits

```
DECLARE
   num integer :=&num;
   total integer:=0;
   digit integer:=0;
BEGIN
   while (num !=0) loop
      digit := mod(num,10);
      total := total + digit;
      num := trunc(num/10);
   end loop;
   dbms_output.put_line('sum of digits of given number is '||total);
END;
/
```

Question 5: Program to print the days names in the week.

```
DECLARE
d number:=&num1;
BEGIN
case d
  when 1 then
    dbms_output.put_line('sunday');
  when 2 then
    dbms_output.put_line('monday');
  when 3 then
    dbms_output.put_line('tuesday');
  when 4 then
    dbms_output.put_line('wednesday');
  when 5 then
    dbms_output.put_line('thursday');
  when 6 then
    dbms_output.put_line('friday');
  when 7 then
    dbms_output.put_line('saturday');
  else
    dbms_output.put_line('invalid day');
end case;
END;
/
```