**Practical No 21 and 22: Use conditional statement and loop structure in PL/SQL.**

1. **Practical Related Questions:**
2. Give the use of CASE statement in PL/SQL.

Like the IF statement, the CASE statement selects one sequence of statements to execute. However, to select the sequence, the CASE statement uses a selector rather than multiple Boolean expressions. A selector is an expression, the value of which is used to select one of several alternatives.

1. State use of EXIT WHEN statement in PL/SQL.

The EXIT-WHEN statement allows the condition in the WHEN clause to be evaluated. If the condition is true, the loop completes and control passes to the statement immediately after the END LOOP. A statement inside the loop must change the value of the condition.

1. List some special characteristics of PL/SQL for Loop.

* The initial step is executed first, and only once. This step allows you to declare and initialize any loop control variables.
* Next, the condition, i.e., initial\_value.
* After the body of the for loop executes, the value of the counter variable is increased or decreased.
* The condition is now evaluated again.

1. Give an example of nested loop.

Declare

i number(3);

j number(3);

Begin

i:=2;

Loop

j:=2;

Loop

exit When ((mod(i,j)=0)or(j=1));

j:=j+1;

End loop;

if(j=1) Then

dbms\_output.put\_line(i||'is prime');

End if;

i:=i+1;

exit When i=50;

End loop;

End;

1. **Exercise:**
2. Write a program to print odd numbers between 1 to 100 using while loop.

Declare

n number:=1;

Begin

While n<=100

Loop

dbms\_output.put\_line(n);

n:=n+2;

End Loop;

End;

1. Classify the entered alphabet is vowel or consonant using CASE statement.

Declare

ch varchar2(10):='i';

Begin

case ch

when 'a' then dbms\_output.put\_line('Vowel');

when 'e' then dbms\_output.put\_line('Vowel');

when 'i' then dbms\_output.put\_line('Vowel');

when 'o' then dbms\_output.put\_line('Vowel');

when 'u' then dbms\_output.put\_line('Vowel');

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

when 'E' then dbms\_output.put\_line('Vowel');

when 'I' then dbms\_output.put\_line('Vowel');

when 'O' then dbms\_output.put\_line('Vowel');

when 'U' then dbms\_output.put\_line('Vowel');

else dbms\_output.put\_line('Constant');

end case;

End;

1. Using nested for loop print table of 2, 3, 4 and 5.

Declare

n number;

i number;

Begin

n:=2;

for i in 1..10

loop

dbms\_output.put\_line(n||'x'||i||'='||n\*i);

end loop;

End;

Declare

n number;

i number;

Begin

n:=3;

for i in 1..10

loop

dbms\_output.put\_line(n||'x'||i||'='||n\*i);

end loop;

End;

Declare

n number;

i number;

Begin

n:=4;

for i in 1..10

loop

dbms\_output.put\_line(n||'x'||i||'='||n\*i);

end loop;

End;

Declare

n number;

i number;

Begin

n:=5;

for i in 1..10

loop

dbms\_output.put\_line(n||'x'||i||'='||n\*i);

end loop;

End;

1. Find some of first 10 natural numbers

Declare

s number;

i number;

Begin

s:=0;

for i in 1..10

loop

s:=s+i;

end loop;

dbms\_output.put\_line('Sum = '||s);

End;