Using Control Structures in PL/SQL

Lab session 12

Part 01

Conditional Constructs with IF statements

Conditional Constructs with IF statements

Part o1

There are three forms of IF statements

- IF-THEN-END IF
- IF-THEN-ELSE-END IF
- IF-THEN-ELSIF-END IF

Students you are quite intelligent to understand these three types by following examples:

- ***IF-THEN-END IF*** Let say we have 2 variables ***V-ename*** hold ename and ***v_mgr*** holds mgr value of emp table

```
IF v_ename = 'RAFAY' THEN
    v_mgr := 303;
END IF;
```

Conditional Constructs with IF statements

Part o1

- ***IF-THEN-ELSE-END IF***

```
IF v_ename = ‘ALI’ THEN
     v_mgr := 92;
ELSE
     v_mgr := 02;
END IF;
```

- ***IF-THEN-ELSIF-END IF***

```
IF v_ename = ‘MAHAM’ THEN
     v_mgr := 03;
ELSIF v_ename = ‘SANIA’
     v_mgr := 08 ;
ELSE
     v_mgr := 83 ;
END IF;
```

Part 02

Iterative Control

Iterative Control

Part o2

There are three Types of Loops in PL/SQL

- Basic Loop
- For Loop
- While Loop

- ***Basic Loop :***

```
DECLARE
    v_counter NUMBER(2) := 1;

BEGIN
    LOOP
        dbms_output.put_line(v_counter);
        V_counter := v_counter + 1;
        EXIT WHEN v_counter > 10;
    END LOOP;
END;
/
```

This is terminating condition with "exit" keyword

OutPut :

```
1
2
3
4
5
6
7
8
9
10
```

PL/SQL procedure successfully completed.

Iterative Control

Part o2

- ***For Loop:***

```
DECLARE
    v_counter NUMBER(2) := 1;
BEGIN
    FOR I IN 1..10 LOOP
        dbms_output.put_line(v_counter);
    END LOOP;
END;
/
```

OutPut:

```
1
1
1
1
1
1
1
1
1
1

PL/SQL procedure successfully completed.
```

Iterative Control

Part o2

- ***While Loop:***

```
ACCEPT C_Count PROMPT ”Enter count”
DECLARE
    v_counter NUMBER(2) := 1;
BEGIN
    WHILE v_counter <= &C_Count LOOP
        dbms_output.put_line(v_counter);
        v_counter := v_counter + 1;
    END LOOP;
END;
/
```

OutPut:

1
2
3
4
5
6
7
8
9
10

PL/SQL procedure successfully completed.

That’s all!

Finished! ☺