

Connected to:  
Oracle Database 10g Enterprise Edition Release 10.2.0.1.0 - Production  
With the Partitioning, OLAP and Data Mining options

```
SQL> SET SERVEROUTPUT ON
SQL> SET LINESIZE 150;
SQL> SET PAGESIZE 20;
SQL> DECLARE
2     a real;
3     b real;
4     c real;
5     maximum real;
6 BEGIN
7     a := &a;
8     dbms_output.put_line(a);
9     b := &b;
10    dbms_output.put_line(b);
11    c := &c;
12    dbms_output.put_line(c);
13    IF a > b THEN
14        IF a > c THEN
15            maximum := a;
16        ELSE
17            maximum := c;
18        END IF;
19    ELSE
20        IF b > c THEN
21            maximum := b;
22        ELSE
23            maximum := c;
24        END IF;
25    END IF;
26    dbms_output.put_line('Maximum Value is : ' || maximum);
27 END;
28 /
```

Enter value for a: 10

old 7: a := &a;

new 7: a := 10;

Enter value for b: 20

old 9: b := &b;

new 9: b := 20;

Enter value for c: 30

old 11: c := &c;

new 11: c := 30;

10

20

30

Maximum Value is : 30

PL/SQL procedure successfully completed.

```
old 9:      b := &b;
new 9:      b := 20;
Enter value for c: 30
old 11:     c := &c;
new 11:     c := 30;
10
20
30
Maximum Value is : 30
```

PL/SQL procedure successfully completed.

```
SQL> DECLARE
2     num integer;
3     remainder integer;
4 BEGIN
5     dbms_output.put_line('Enter a number :- ');
6     num := &num;
7     dbms_output.put_line(num);
8     remainder := num MOD 55;
9     IF remainder = 0 THEN
10        dbms_output.put_line('Number is Divisible by 5 and 11');
11    ELSE
12        dbms_output.put_line('Not Divisible by 5 and 11');
13    END IF;
14 END;
15 /
```

```
Enter value for num: 29
old 6:      num := &num;
new 6:      num := 29;
Enter a number :-
29
Not Divisible by 5 and 11
```

PL/SQL procedure successfully completed.

```
SQL> DECLARE
2     length integer;
3     breadth integer;
4     side integer;
5     base integer;
6     height integer;
7     area real;
8 BEGIN
9     dbms_output.put_line('Enter the length and breadth of the rectangle :- ');
10    length := &length;
11    dbms_output.put_line(length);
12    breadth := &breadth;
13    dbms_output.put_line(breadth);
14    area := length * breadth;
```



Enter a number :-

29

Not Divisible by 5 and 11

PL/SQL procedure successfully completed.

SQL> DECLARE

```

1  length integer;
2  breadth integer;
3  side integer;
4  base integer;
5  height integer;
6  area real;
7
8  BEGIN
9      dbms_output.put_line('Enter the length and breadth of the rectangle :- ');
10     length := &length;
11     dbms_output.put_line(length);
12     breadth := &breadth;
13     dbms_output.put_line(breadth);
14     area := length * breadth;
15     dbms_output.put_line('Area of the rectangle is : ' || area);
16     dbms_output.put_line('Enter the side of the square :- ');
17     side := &side;
18     dbms_output.put_line(side);
19     area := side * side;
20     dbms_output.put_line('Area of the square is : ' || area);
21     dbms_output.put_line('Enter the Base of the Triangle :- ');
22     base := &base;
23     dbms_output.put_line(base);
24     dbms_output.put_line('Enter the Height of the Triangle :- ');
25     height := &height;
26     dbms_output.put_line(height);
27     area := 0.5 * base * height;
28     dbms_output.put_line('Area of the Triangle is : ' || area);
29 END;
30 /

```

Enter value for length: 10

old 10: length := &length;

new 10: length := 10;

Enter value for breadth: 2

old 12: breadth := &breadth;

new 12: breadth := 2;

Enter value for side: 5

old 17: side := &side;

new 17: side := 5;

Enter value for base: 10

old 22: base := &base;

new 22: base := 10;

Enter value for height: 5

old 25: height := &height;

```

10 length := &length;
11 dbms_output.put_line(length);
12 breadth := &breadth;
13 dbms_output.put_line(breadth);
14 area := length * breadth;
15 dbms_output.put_line('Area of the rectangle is : ' || area);
16 dbms_output.put_line('Enter the side of the square :- ');
17 side := &side;
18 dbms_output.put_line(side);
19 area := side * side;
20 dbms_output.put_line('Area of the square is : ' || area);
21 dbms_output.put_line('Enter the Base of the Triangle :- ');
22 base := &base;
23 dbms_output.put_line(base);
24 dbms_output.put_line('Enter the Height of the Triangle :- ');
25 height := &height;
26 dbms_output.put_line(height);
27 area := 0.5 * base * height;
28 dbms_output.put_line('Area of the Triangle is : ' || area);
29 END;
30 /

```

Enter value for length: 10

old 10: length := &length;

new 10: length := 10;

Enter value for breadth: 2

old 12: breadth := &breadth;

new 12: breadth := 2;

Enter value for side: 5

old 17: side := &side;

new 17: side := 5;

Enter value for base: 10

old 22: base := &base;

new 22: base := 10;

Enter value for height: 5

old 25: height := &height;

new 25: height := 5;

Enter the length and breadth of the rectangle :-

10

2

Area of the rectangle is : 20

Enter the side of the square :-

5

Area of the square is : 25

Enter the Base of the Triangle :-

10

Enter the Height of the Triangle :-

5

Area of the Triangle is : 25

PL/SQL procedure successfully completed.



Enter the Height of the Triangle :-

5

Area of the Triangle is : 25

PL/SQL procedure successfully completed.

```
SQL> DECLARE
2     phy integer;
3     chem integer;
4     bio integer;
5     math integer;
6     computer integer;
7     total integer;
8     percentage real;
9 BEGIN
10    dbms_output.put_line('Enter the marks of Physics, Chemistry, Biology, Mathematics and Computer :- ');
11    phy := &phy;
12    dbms_output.put_line(phy);
13    chem := &chem;
14    dbms_output.put_line(chem);
15    bio := &bio;
16    dbms_output.put_line(bio);
17    math := &math;
18    dbms_output.put_line(math);
19    computer := &computer;
20    dbms_output.put_line(computer);
21    total := phy + chem + bio + math + computer;
22    percentage := total / 5;
23    dbms_output.put_line('Student Percentage : ' || percentage);
24    IF percentage >= 90 THEN
25        dbms_output.put_line('Grade A');
26    ELSIF percentage >= 80 THEN
27        dbms_output.put_line('Grade B');
28    ELSIF percentage >= 70 THEN
29        dbms_output.put_line('Grade C');
30    ELSIF percentage >= 60 THEN
31        dbms_output.put_line('Grade D');
32    ELSIF percentage >= 40 THEN
33        dbms_output.put_line('Grade E');
34    ELSE
35        dbms_output.put_line('Grade F');
36    END IF;
37 END;
38 /
```

Enter value for phy: 95

old 11: phy := &phy;

new 11: phy := 95;

Enter value for chem: 93

old 13: chem := &chem;

new 13: chem := 93;

```

16  dbms_output.put_line(bio);
17  math := &math;
18  dbms_output.put_line(math);
19  computer := &computer;
20  dbms_output.put_line(computer);
21  total := phy + chem + bio + math + computer;
22  percentage := total / 5;
23  dbms_output.put_line('Student Percentage : ' || percentage);
24  IF percentage >= 90 THEN
25      dbms_output.put_line('Grade A');
26  ELSIF percentage >= 80 THEN
27      dbms_output.put_line('Grade B');
28  ELSIF percentage >= 70 THEN
29      dbms_output.put_line('Grade C');
30  ELSIF percentage >= 60 THEN
31      dbms_output.put_line('Grade D');
32  ELSIF percentage >= 40 THEN
33      dbms_output.put_line('Grade E');
34  ELSE
35      dbms_output.put_line('Grade F');
36  END IF;
37  END;
38  /
Enter value for phy: 95
old 11:  phy := &phy;
new 11:  phy := 95;
Enter value for chem: 93
old 13:  chem := &chem;
new 13:  chem := 93;
Enter value for bio: 92
old 15:  bio := &bio;
new 15:  bio := 92;
Enter value for math: 90
old 17:  math := &math;
new 17:  math := 90;
Enter value for computer: 99
old 19:  computer := &computer;
new 19:  computer := 99;
Enter the marks of Physics, Chemistry, Biology, Mathematics and Computer :-
95
93
92
90
99
Student Percentage : 93.8
Grade A

PL/SQL procedure successfully completed.

SQL> DECLARE

```



```
old 13:      chem := &chem;
new 13:      chem := 93;
Enter value for bio: 92
old 15:      bio := &bio;
new 15:      bio := 92;
Enter value for math: 90
old 17:      math := &math;
new 17:      math := 90;
Enter value for computer: 99
old 19:      computer := &computer;
new 19:      computer := 99;
Enter the marks of Physics, Chemistry, Biology, Mathematics and Computer :-
95
93
92
90
99
Student Percentage : 93.8
Grade A
```

PL/SQL procedure successfully completed.

```
SQL> DECLARE
2      s integer := 0;
3      i integer := 1;
4  BEGIN
5      WHILE i <= 100 LOOP
6          s := s + i;
7          i := i + 1;
8      END LOOP;
9      dbms_output.put_line('Sum of First 100 natural numbers is : ' || s);
10 END;
11 /
Sum of First 100 natural numbers is : 5050
```

PL/SQL procedure successfully completed.

```
SQL> CREATE TABLE Empinfo(id number(5), name varchar2(20), age number(3), address varchar2(20), salary number(10));

Table created.
```

```
SQL> INSERT ALL
2      INTO Empinfo(id, name, age, address, salary) VALUES (1, 'Ramesh', 32, 'Ahmedabad', 2000)
3      INTO Empinfo(id, name, age, address, salary) VALUES (2, 'Khilan', 25, 'Delhi', 1500)
4      INTO Empinfo(id, name, age, address, salary) VALUES (3, 'Kaushik', 23, 'Kota', 2000)
5      INTO Empinfo(id, name, age, address, salary) VALUES (4, 'Chaital', 25, 'Mumbai', 6500)
6      INTO Empinfo(id, name, age, address, salary) VALUES (5, 'Hardik', 27, 'Bhopal', 8500)
7      INTO Empinfo(id, name, age, address, salary) VALUES (6, 'Komal', 22, 'MP', 4500)
8  SELECT * FROM dual;
```

```
9      dbms_output.put_line('Sum of First 100 natural numbers is : ' || s);
10 END;
11 /
```

Sum of First 100 natural numbers is : 5050

PL/SQL procedure successfully completed.

```
SQL> CREATE TABLE Empinfo(id number(5), name varchar2(20), age number(3), address varchar2(20), salary number(10));
```

Table created.

```
SQL> INSERT ALL
```

```
2      INTO Empinfo(id, name, age, address, salary) VALUES (1, 'Ramesh', 32, 'Ahmedabad', 2000)
3      INTO Empinfo(id, name, age, address, salary) VALUES (2, 'Khilan', 25, 'Delhi', 1500)
4      INTO Empinfo(id, name, age, address, salary) VALUES (3, 'Kaushik', 23, 'Kota', 2000)
5      INTO Empinfo(id, name, age, address, salary) VALUES (4, 'Chaital', 25, 'Mumbai', 6500)
6      INTO Empinfo(id, name, age, address, salary) VALUES (5, 'Hardik', 27, 'Bhopal', 8500)
7      INTO Empinfo(id, name, age, address, salary) VALUES (6, 'Komal', 22, 'MP', 4500)
8 SELECT * FROM dual;
```

6 rows created.

```
SQL> DECLARE
```

```
2      c_id Empinfo.id%TYPE;
3      c_name Empinfo.name%TYPE;
4      c_age Empinfo.age%TYPE;
5      c_salary Empinfo.salary%TYPE;
6 BEGIN
7      SELECT id, name INTO c_id, c_name
8      FROM Empinfo WHERE id=1;
9      dbms_output.put_line('The name of person having id=1 is ' || c_name);
10     SELECT name, age, salary INTO c_name, c_age, c_salary
11     FROM Empinfo WHERE address='Kota';
12     dbms_output.put_line('The name,age, and salary lives in Kota is ' || c_name || ', ' || c_age || ',and ' || c_salary || '.');
13 END;
14 /
```

The name of person having id=1 is Ramesh

The name,age, and salary lives in Kota is Kaushik, 23,and 2000.

PL/SQL procedure successfully completed.

```
SQL> DROP TABLE Empinfo;
```

Table dropped.

```
SQL>
```