DBMS LAB REPORT

Lab Number : 08
Date : 21 April, 2022
Name of the student : Ashish Kumar Mishra
Roll Number : 20051685
Branch : Computer Science and Engineering
Section : CSE -10

Q1. Write a PLSQL program to display Largest among 3 numbers.

Source Code :

```
Declare
    a number;
    b number;
    c number;
Begin
    dbms_output.put_line('Enter a:');
        a:=&a;
    dbms_output.put_line('Enter b:');
        b:=&b;
    dbms_output.put_line('Enter c:');
        c:=&C;
if (a>b) and (a>c)
    then
    dbms_output.put_line('A is GREATEST'||A);
elsif (b>a) and (b>c)
    then
    dbms_output.put_line('B is GREATEST'||B);
else
    dbms_output.put_line('C is GREATEST'||C);
end if;
End;
/
```

Output :

```
SQL> connect C_20051685;
Enter password:
Connected.
SQL> set serveroutput on;
SQL> @"K:\I Folder\SQL\LAB8\Q1.sql";
Enter value for a: 5
old   7:          a:=&a;
new   7:          a:=5;
Enter value for b: 8
old   9:          b:=&b;
new   9:          b:=8;
Enter value for c: 2
old  11:          c:=&C;
new  11:          c:=2;
Enter a:
Enter b:
Enter c:
B is GREATEST8

PL/SQL procedure successfully completed.
```

Q2. Write a PLSQL program to display sum and average of 5 numbers.

Source Code :

```
DECLARE
    a     NUMBER := 12;
    b     NUMBER := 14;
    c     NUMBER := 20;
    d     NUMBER := 22;
    e     NUMBER := 25;
    sumOf5 NUMBER;
    avgOf5 NUMBER;
BEGIN
    sumOf5 := a + b + c + d + e;
    avgOf5 := sumOf5 / 5;
    dbms_output.Put_line('Sum = '
                         ||sumOf5);
    dbms_output.Put_line('Average = '
                         ||avgOf5);
END;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q2.sql";
Sum = 93
Average = 18.6

PL/SQL procedure successfully completed.
```

Q3. Write a PLSQL program to display square and cube of a number.

Source Code :

```
declare
a int:=8;
cube int;
square int;
begin
square:=a*a;
cube:= a*a*a;
```

```
dbms_output.put_line('The Cube of the number is'||cube);
dbms_output.put_line('The Square of number is' || square);
end;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q3.sql";
The Cube of the number is512
The Square of number is64

PL/SQL procedure successfully completed.
```

Q4. Write a PLSQL program to check the age of a person is eligible to vote or not.

Source Code :

```
Declare
Age number;
Begin
--Accept Age number prompt 'Enter your age : ';
dbms_output.put_line('Enter Age:');
Age := &Age;
IF Age >= 18 THEN
dbms_output.Put_line( 'The user is eligible to cast vote');
ELSE
dbms_output.Put_line( 'The user is not eligible to cast vote');
END IF;
end;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q4.sql";
Enter value for age: 22
old   6: Age := &Age;
new   6: Age := 22;
Enter Age:
The user is eligible to cast vote

PL/SQL procedure successfully completed.
```

Q5. Write a PLSQL program to check the amount entered in rupees is multiples of hundred or not.

Source Code :

```
Declare
a int;
begin
a:=&a;
if (MOD (a,100)=0) then
dbms_output.put_line('Money is multiple of 100');
else
dbms_output.put_line('Money is not a  multiple of 100');
end if;
end;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q5.sql";
Enter value for a: 8
old   4: a:=&a;
new   4: a:=8;
Money is not a  multiple of 100

PL/SQL procedure successfully completed.
```

Q6. Write a PLSQL program to display smallest among 4 numbers.

Source Code :

```
Declare
    a number;
    b number;
    c number;
    d number;
Begin
```

```
    dbms_output.put_line('Enter a:');
        a:=&a;
    dbms_output.put_line('Enter b:');
        b:=&b;
    dbms_output.put_line('Enter c:');
        c:=&C;
        dbms_output.put_line('Enter d:');
        d:=&d;
if (a<b) and (a<c) and (a<d)
    then
    dbms_output.put_line('A is Smallest'||A);
elsif (b<a) and (b<c) and (b<d)
    then
    dbms_output.put_line('B is smallest'||B);
elsif (c<a) and (c<b) and (c<d)
    then
    dbms_output.put_line('C is smallest'||C);
 else
    dbms_output.put_line('d is smallest'||d);
    End if;
End;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q6.sql";
Enter value for a: 5
old    8:           a:=&a;
new    8:           a:=5;
Enter value for b: 6
old   10:           b:=&b;
new   10:           b:=6;
Enter value for c: 4
old   12:           c:=&C;
new   12:           c:=4;
Enter value for d: 9
old   14:           d:=&d;
new   14:           d:=9;
Enter a:
Enter b:
Enter c:
Enter d:
C is smallest4

PL/SQL procedure successfully completed.
```

Q7. Write a PLSQL program to check the entered number is Prime number or not.

Source Code :

```
declare
n number;
i number;
flag number;
begin
i:=2;
flag:=1;
n:=&n;
for i in 2..n/2
loop
if mod(n,i)=0
then
flag:=0;
exit;
end if;
end loop;
if flag=1
then
dbms_output.put_line('prime');
else
dbms_output.put_line('not prime');
end if;
end;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q7.sql";
Enter value for n: 5
old   8: n:=&n;
new   8: n:=5;
prime

PL/SQL procedure successfully completed.
```

Q8. Write a PLSQL program to display the average number of records of any database table.

Source Code :

```
CREATE TABLE LAB8Q8_table
(id NUMBER(4),
salary NUMBER(10));
BEGIN
FOR idx IN 1..10 LOOP
INSERT INTO LAB8Q8_table(ID,salary)
VALUES(idx, --ID value
DBMS_RANDOM.VALUE(7000,8000));
END LOOP
COMMIT;
END;
/
SELECT id,salary FROM LAB8Q8_table;
DECLARE
CURSOR my_cursor IS
SELECT AVG(salary) AS avg_salary
FROM LAB8Q8_table;
c_my_cursor_rec my_cursor%ROWTYPE;
BEGIN
OPEN my_cursor;
LOOP
FETCH my_cursor INTO c_my_cursor_rec;
EXIT WHEN my_cursor%NOTFOUND;
DBMS_OUTPUT.PUT_LINE('Average(salary) = '||c_my_cursor_rec.avg_salary);
END LOOP;
CLOSE my_cursor;
END;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q8.sql";

Table created.


PL/SQL procedure successfully completed.


        ID     SALARY
---------- ----------
         1       7557
         2       7187
         3       7131
         4       7533
         5       7223
         6       7156
         7       7779
         8       7566
         9       7405
        10       7430

10 rows selected.

Average(salary) = 7396.7

PL/SQL procedure successfully completed.
```

Q9. Write a PLSQL program to display the product of 2 numbers without multiplication.

Source Code :

```
declare
a number;
b number;
i number:=1;
s number:=0;
begin
dbms_output.put_line('Enter a:');
        a:=&a;
dbms_output.put_line('Enter b:');
        b:=&b;
while i<=b
loop
s:=s+a;
i:=i+1;
end loop;
dbms_output.put_line('product='||s);
end;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q9.sql";
Enter value for a: 5
old   8:          a:=&a;
new   8:          a:=5;
Enter value for b: 4
old  10:          b:=&b;
new  10:          b:=4;
Enter a:
Enter b:
product=20

PL/SQL procedure successfully completed.
```

Q10. Write a PLSQL program to sum of all the numbers from 1 to n.

Source Code :

```
DECLARE
sumVal NUMBER;
n NUMBER;
i NUMBER;
```

```
FUNCTION Findmax(n IN NUMBER)
   RETURN NUMBER
IS
   sums NUMBER := 0;
BEGIN
   FOR i IN 1..n
   LOOP
   sums := (i*(i+1))/2;
   END LOOP;
   RETURN sums;
END;
BEGIN
   n := &n;
   sumVal := findmax(n);
   dbms_output.Put_line('Sum of natural numbers is ' || sumVal);
END;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q10.sql";
Enter value for n: 4
old  17:    n := &n;
new  17:    n := 4;
Sum of natural numbers is 10

PL/SQL procedure successfully completed.
```

Q11. Write a PLSQL program to accept the name of a student and display it in Upper case and Lowercase.

Source Code :

```
DECLARE
   Test_String string(10) := 'Ashish';
BEGIN
   dbms_output.put_line('STUDENT NAME IN UPPERCASE IS :' || UPPER(Test_String));
   dbms_output.put_line('STUDENT NAME IN LOWERCASE IS:' || LOWER(Test_String));
END;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q11.sql";
STUDENT NAME IN UPPERCASE IS :ASHISH
STUDENT NAME IN LOWERCASE IS:ashish

PL/SQL procedure successfully completed.
```

Q12. Write a PLSQL program to display all the odd numbers within a given range.

Source Code :

```
declare
a number;
b number;
c number;
d number;
begin
a:=&a;
b:=&b;
dbms_output.put_line('the entered range is : ' || a || ' to ' || b);
if (mod(a,2)=0)
 then
 c:=a+1;
 while c<=b
 loop
 dbms_output.put_line(c);
 c:=c+2;
 end loop;
else
 d:=a;
 while d<=b
 loop
 dbms_output.put_line(d);
 d:=d+2;
 end loop;
end if ;
end;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q12.sql";
Enter value for a: 1
old   7: a:=&a;
new   7: a:=1;
Enter value for b: 10
old   8: b:=&b;
new   8: b:=10;
the entered range is : 1 to 10
1
3
5
7
9

PL/SQL procedure successfully completed.
```

Q13. Write a PLSQL program to display multiplication table up to 10 for a given number.

Source Code :

```
declare
    n number;
    i number;
    begin
    n:=&n;
    for i in 1..10
    loop
    dbms_output.put_line(n||' x '||i||' = '||n*i);
   end loop;
   end;
   /
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q13.sql";
Enter value for n: 4
old   5:      n:=&n;
new   5:      n:=4;
4 x 1 = 4
4 x 2 = 8
4 x 3 = 12
4 x 4 = 16
4 x 5 = 20
4 x 6 = 24
4 x 7 = 28
4 x 8 = 32
4 x 9 = 36
4 x 10 = 40

PL/SQL procedure successfully completed.
```

Q14. Write a PLSQL program to display the sum of every digit of a given number.

Source Code :

```
DECLARE
      n        INTEGER;
      temp_sum INTEGER;
      r        INTEGER;
  BEGIN
      n := &n;
      temp_sum := 0;
      WHILE n <> 0 LOOP
```

```
        r := MOD(n, 10);
        temp_sum := temp_sum + r;
        n := Trunc(n / 10);
    END LOOP;
    dbms_output.Put_line('sum of every digit  = '|| temp_sum);
END;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q14.sql";
Enter value for n: 121
old   6:          n := &n;
new   6:          n := 121;
sum of every digit  = 4

PL/SQL procedure successfully completed.
```

Q15. Write a PLSQL program to check the given number is Palindrome or not.

Source Code :

```
declare
      n number;
      m number;
      temp number:=0;
      rem number;
   begin
      n:=&n;
      m:=n;
     while n>0
     loop
         rem:=mod(n,10);
         temp:=(temp*10)+rem;
         n:=trunc(n/10);
     end loop;
     if m = temp
     then
         dbms_output.put_line('Yes , Number is Palindrome');
     else
         dbms_output.put_line('No , Number is not Palindrome');
    end if;
```

```
end;
/
```

Output :

```
SQL> @"K:\I Folder\SQL\LAB8\Q15.sql";
Enter value for n: 121
old   7:          n:=&n;
new   7:          n:=121;
Yes , Number is Palindrome

PL/SQL procedure successfully completed.
```