Assignment-8

Thota GuruTheja reddy

19BCD7034

1) Write PL/SQL code to find Largest of three numbers.

```
Declare
Var1 integer;
Var2 integer;
Var3 integer;
Begin
Var1:=&var1;
Var2:=&var2;
Var3:=&var3;
if(Var1>Var2 and Var1>Var3)
then
dbms_output.put_line('Largest number is ' || Var1);
elsif(Var2>Var1 and Var2>Var3)
then
dbms_output.put_line('Largest number is ' || Var2);
else
dbms_output.put_line('Largest number is ' || Var3);
end if;
end;
/
```

```
SQL> Declare
  2  Var1 integer;
  3  Var2 integer;
  4  Var3 integer;
  5  Begin
  6  Var1:=&var1;
  7  Var2:=&var2;
  8  Var3:=&var3;
  9  if(Var1>Var2 and Var1>Var3)
 10  then
 11  dbms_output.put_line('Largest number is ' || Var1);
 12  elsif(Var2>Var1 and Var2>Var3)
 13  then
 14  dbms_output.put_line('Largest number is ' || Var2);
 15  else
 16  dbms_output.put_line('Largest number is ' || Var3);
 17  end if;
 18  end;
 19  /
Enter value for var1: 89
old   6: Var1:=&var1;
new   6: Var1:=89;
Enter value for var2: 69
old   7: Var2:=&var2;
new   7: Var2:=69;
Enter value for var3: 79
old   8: Var3:=&var3;
new   8: Var3:=79;
Largest number is 89

PL/SQL procedure successfully completed.
```

2) Write PL/SQL code to find Factorial of a given number

```
declare
factorial number :=1;
num number:= &num;
begin
while num > 0
loop
```

```
factorial:=num*factorial;
num:=num-1;
end loop;
dbms_output.put_line('Factorial of given number is ' || factorial);
end;
/
```

```
SQL> declare
  2  factorial number :=1;
  3  num number:= &num;
  4  begin
  5  while num > 0
  6  loop
  7  factorial:=num*factorial;
  8  num:=num-1;
  9  end loop;
 10  dbms_output.put_line('Factorial of given number is ' || factorial);
 11  end;
 12  /
Enter value for num: 4
old   3: num number:= &num;
new   3: num number:= 4;
Factorial of given number is 24

PL/SQL procedure successfully completed.
```

3) Write PL/SQL code to Read number and prints its Multiplication Table

```
declare
num number;
temp number:=1;
result number;
begin
num :=&num;
while temp <= 10
loop
result := num*temp;
dbms_output.put_line(num || ' x ' || temp || ' = ' || result);
```

temp:=temp+1;

end loop;

end;

/

```
PL/SQL procedure successfully completed.

SQL> declare
  2  num number;
  3  temp number:=1;
  4  result number;
  5  begin
  6  num :=&num;
  7  while temp <= 10
  8  loop
  9  result := num*temp;
 10  dbms_output.put_line(num || ' x ' || temp || ' = ' || result);
 11  temp:=temp+1;
 12  end loop;
 13  end;
 14  /
Enter value for num: 7
old   6: num :=&num;
new   6: num :=7;
7 x 1 = 7
7 x 2 = 14
7 x 3 = 21
7 x 4 = 28
7 x 5 = 35
7 x 6 = 42
7 x 7 = 49
7 x 8 = 56
7 x 9 = 63
7 x 10 = 70

PL/SQL procedure successfully completed.
```

4) Write PL/SQL code to find given number is Prime or not

```
declare
    num number:=&num;
    i number:=1;
    c number:=0;
```

```
 begin
     for i in 1..num
     loop
       if((mod(num,i))=0)
        then
           c:=c+1;
       end if;
    end loop;
   if(c>2)
   then
      dbms_output.put_line(num||' not a prime');
   else
     dbms_output.put_line(num||' is prime');
   end if;
 end;
  /
```

```
SQL>  declare
  2         num number:=&num;
  3         i number:=1;
  4         c number:=0;
  5    begin
  6         for i in 1..num
  7         loop
  8            if((mod(num,i))=0)
  9             then
 10                c:=c+1;
 11           end if;
 12        end loop;
 13       if(c>2)
 14       then
 15           dbms_output.put_line(num||' not a prime');
 16       else
 17          dbms_output.put_line(num||' is prime');
 18       end if;
 19    end;
 20     /
Enter value for num: 19
old   2:         num number:=&num;
new   2:         num number:=19;
19 is prime

PL/SQL procedure successfully completed.
```

5) Write PL/SQL code to accept the text and reverse the text and test whether the given character is Palandrome or not.

```
declare
text1 varchar(30);
text2 varchar(30);
len number;
i number;
begin
text1:= &text1;
len:=length(text1);
for i in reverse 1..len
loop
text2:=text2 || substr(text1,i,1);
end loop;
if text1=text2
then
dbms_output.put_line(text1||' is a palindrome');
else
dbms_output.put_line(text1||' is not a palindrome');
end if;
end;
/
```

```
SQL> declare
  2  text1 varchar(30);
  3  text2 varchar(30);
  4  len number;
  5  i number;
  6  begin
  7  text1:= &text1;
  8  len:=length(text1);
  9  for i in reverse 1..len
 10  loop
 11  text2:=text2 || substr(text1,i,1);
 12  end loop;
 13  if text1=text2
 14  then
 15  dbms_output.put_line(text1||' is a palindrome');
 16  else
 17  dbms_output.put_line(text1||' is not a palindrome');
 18  end if;
 19  end;
 20  /
Enter value for text1: 'gtr'
old   7: text1:= &text1;
new   7: text1:= 'gtr';
gtr is not a palindrome

PL/SQL procedure successfully completed.
```

6) Write PL/SQL code to find Reverse of a given number.

```
declare
num NUMBER;
rev NUMBER;
begin
num:=&num;
rev:=0;
while num>0
loop
rev:=(rev*10) + mod(num,10);
num:=floor(num/10);
end loop;
```

dbms_output.put_line('Reverse of the number is: ' || rev);

END;

/

```
SQL> declare
  2  num NUMBER;
  3  rev NUMBER;
  4  begin
  5  num:=&num;
  6  rev:=0;
  7  while num>0
  8  loop
  9  rev:=(rev*10) + mod(num,10);
 10  num:=floor(num/10);
 11  end loop;
 12  dbms_output.put_line('Reverse of the number is: ' || rev);
 13  END;
 14  /
Enter value for num: 12345
old   5: num:=&num;
new   5: num:=12345;
Reverse of the number is: 54321

PL/SQL procedure successfully completed.
```

7) Write PL/SQL code to generate Fibonacci series for given number.

declare

first number := 0;

second number := 1;

temp number;

n number :=&n;

i number;

begin

dbms_output.put_line('Series:');

dbms_output.put_line(first);

```
dbms_output.put_line(second);
for i in 2..n
loop
temp:=first+second;
first := second;
second := temp;
dbms_output.put_line(temp);
end loop;
end;
/
```

```
SQL> declare
  2  first number := 0;
  3  second number := 1;
  4  temp number;
  5  n number :=&n;
  6  i number;
  7  begin
  8  dbms_output.put_line('Series:');
  9  dbms_output.put_line(first);
 10  dbms_output.put_line(second);
 11  for i in 2..n
 12  loop
 13  temp:=first+second;
 14  first := second;
 15  second := temp;
 16  dbms_output.put_line(temp);
 17  end loop;
 18  end;
 19  /
Enter value for n: 8
old   5: n number :=&n;
new   5: n number :=8;
Series:
0
1
1
2
3
5
8
13
21

PL/SQL procedure successfully completed.
```

8) Write PL/SQL code to print the numbers in this form

1
1 2
1 2 3

```
declare
i number;
j number;
begin
for i in 1..3
loop
for j in 1..i
loop
dbms_output.put(to_char(j) ||' ');
end loop;
dbms_output.new_line;
end loop;
end;
/
```

```
PL/SQL procedure successfully completed.

SQL> declare
  2  i number;
  3  j number;
  4  begin
  5  for i in 1..3
  6  loop
  7  for j in 1..i
  8  loop
  9  dbms_output.put(to_char(j) ||' ');
 10  end loop;
 11  dbms_output.new_line;
 12  end loop;
 13  end;
 14  /
1
1 2
1 2 3

PL/SQL procedure successfully completed.
```