

# **ADBMS Lab**

**Submitted by: Submitted to:**

**NAME : Rahul Gusain Dr.Ankit Khare**

**SAP: 500084143**

**ROLL NO.: R214220900**

**BATCH: 1 DevOps**

**LAB 10**

**EXPERIMENT-10**

**Title: 10. To understand the concepts of function and procedure in PL/SQL.**

Objective: Students will be able to implement the Pl/SQL programs using function and procedure.

**1) Write a PL/SQL code to accept the value of A, B & C display which is greater**.

DECLARE

x number: = 12;

y number: = 18;

z number: = 24;

maximum number;

FUNCTION find\_max (x IN number, y IN number, z IN number)

RETURN number

IS

largest number;

BEGIN

if x>y and x>z then

largest: = x;

else

if y>x and y>z then

largest: = y;

else

largest: = z;

end if;

end if;

RETURN largest;

END;

begin

dbms\_output.put\_line ('X = '||x||', Y = '||y||', Z = '||z);

maximum: = find\_max (12, 18, 24);

IF maximum = x then

dbms\_output.put\_line ('X is the greatest');

end if;

if maximum = y then

dbms\_output.put\_line ('Y is the greatest');

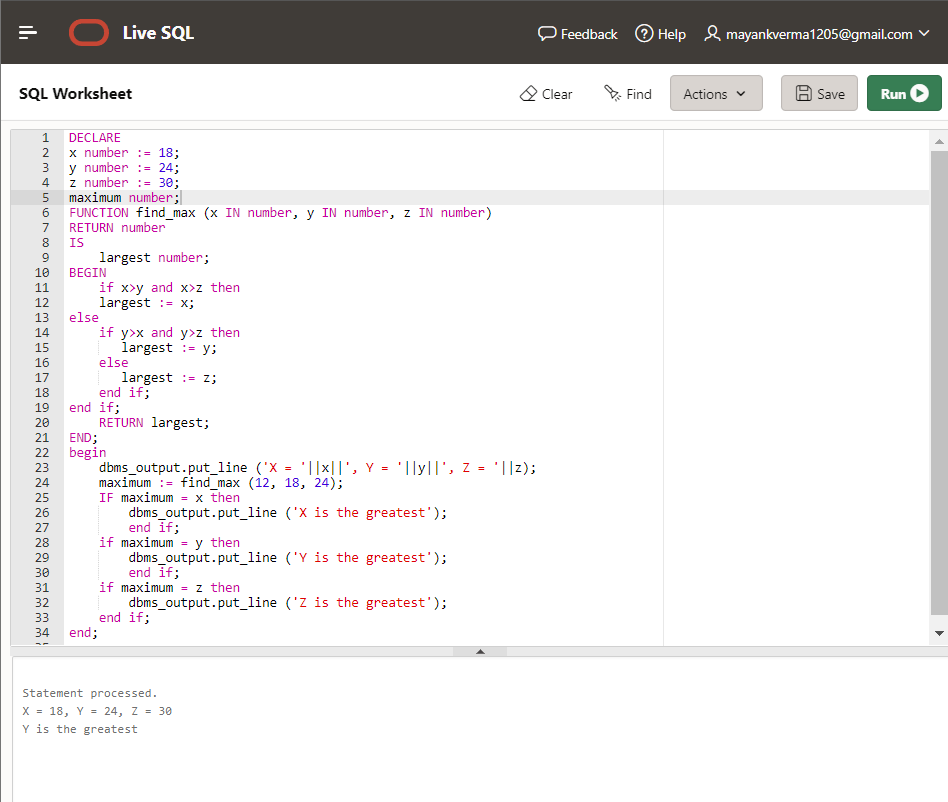
end if;

if maximum = z then

dbms\_output.put\_line ('Z is the greatest');

end if;

end;



**2) Using PL/SQL Statements create a simple loop that display message “Welcome to PL/SQL Programming” 20 times.**

DECLARE

counter number: =1;

FUNCTION Print (count IN number)

RETURN number

IS

c number;

BEGIN

c: = 1;

WHILE c <= 20

LOOP

dbms\_output.put\_line (c || ' Welcome to PL/SQL Programming');

c: = c+1;

END LOOP;

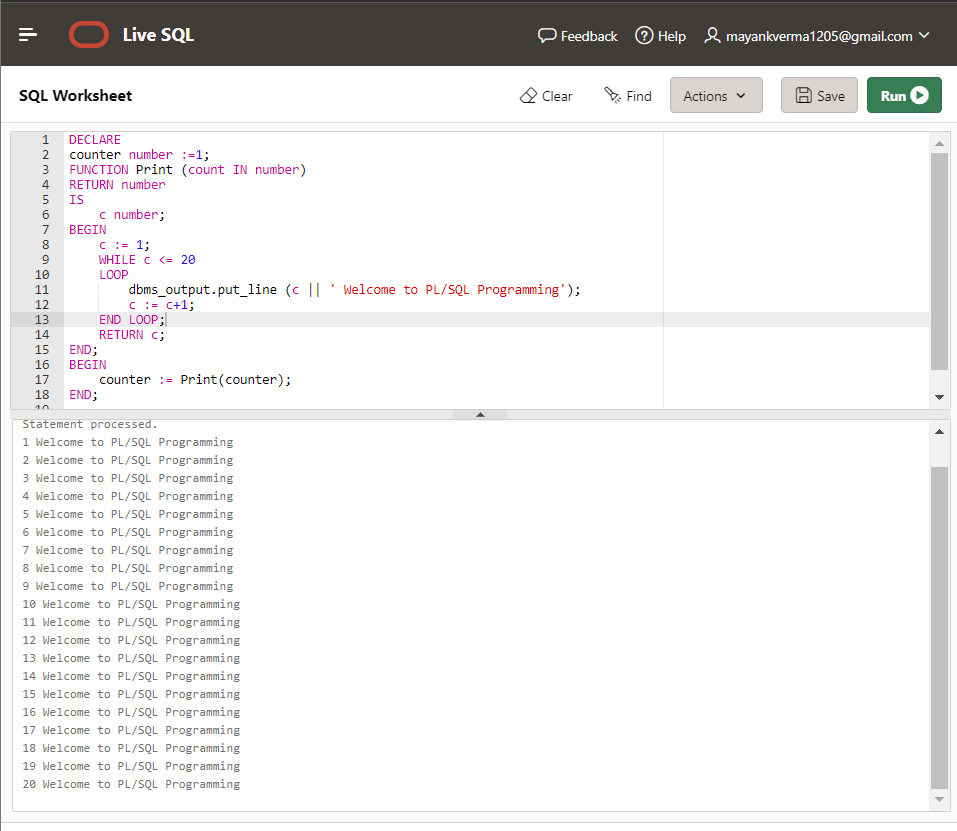
RETURN c;

END;

BEGIN

counter: = Print(counter);

END;



**3) Write a PL/SQL code block to find the factorial of a number.**

DECLARE

x number;

factorial\_store number;

FUNCTION find\_factorial (x number)

RETURN number

IS

fact number;

BEGIN

IF x = 0 THEN

fact: = 1;

ELSE

fact: = x \* find\_factorial(x-1);

END IF;

RETURN fact;

END;

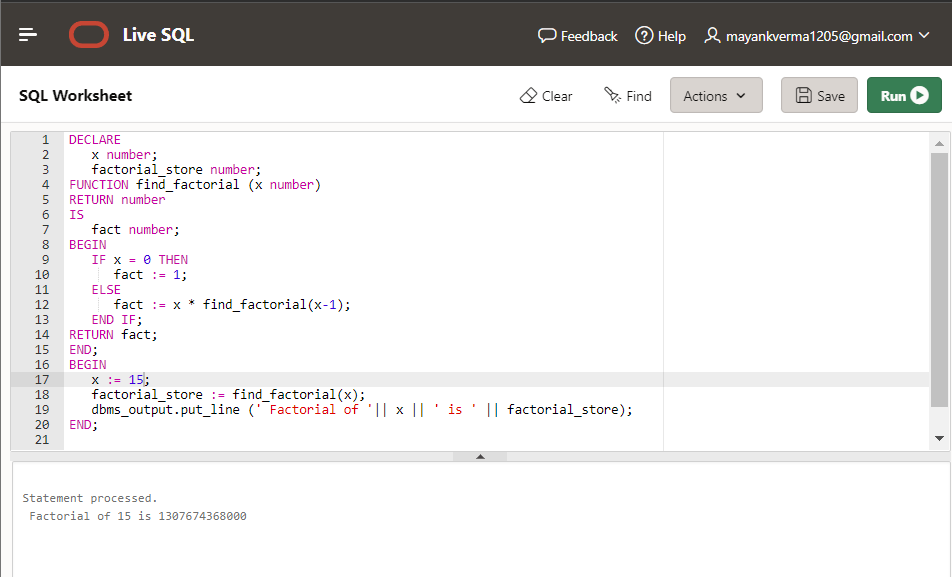
BEGIN

x: = 10;

factorial\_store: = find\_factorial(x);

dbms\_output.put\_line (' Factorial of '|| x || ' is ' || factorial\_store);

END;



**4) Write a PL/SQL program to generate Fibonacci series**

DECLARE

x number: = 0;

y number: = 1;

z number: = 0;

initialize number: = 1;

n number: = 10;

temp\_store number;

temp\_return number: =0;

FUNCTION Fibonacci\_Generator (first\_term IN OUT number, second\_term IN OUT number, add\_counter IN OUT number, count\_var IN OUT number, limithead IN OUT number)

RETURN number

IS

temp\_return number;

BEGIN

while count\_var <= limithead

loop

dbms\_output.put\_line (first\_term || ' ');

add\_counter: = second\_term;

second\_term: = first\_term + second\_term;

first\_term: = add\_counter;

count\_var: = count\_var+1;

END LOOP;

RETURN temp\_return;

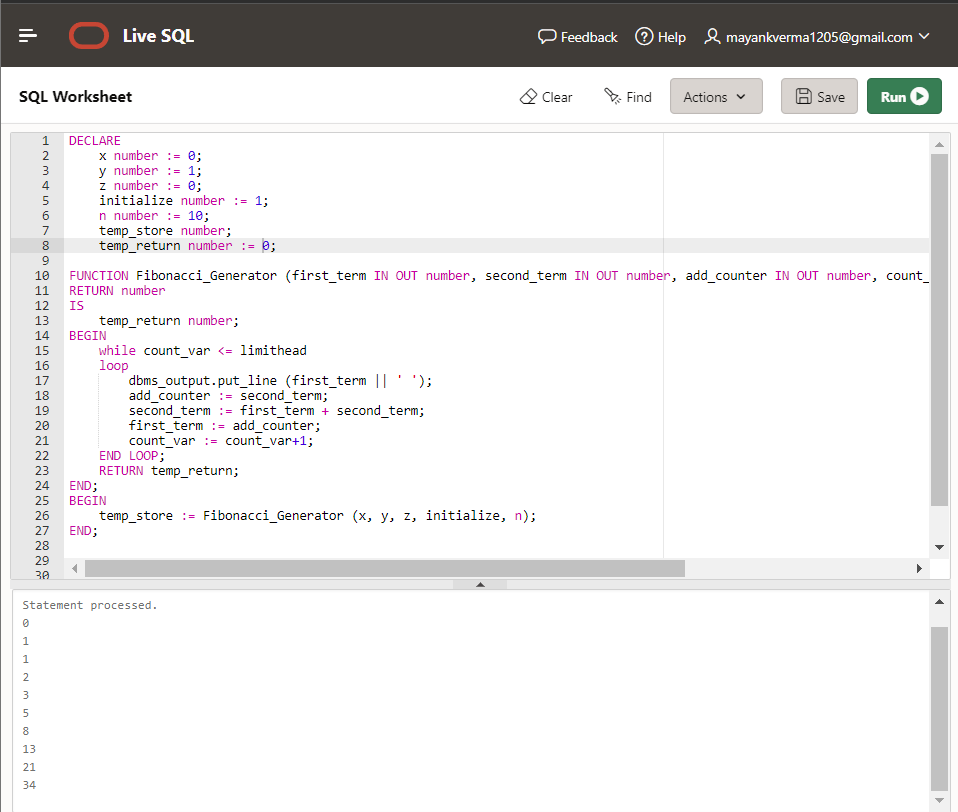
END;

BEGIN

temp\_store: = Fibonacci\_Generator (x, y, z, initialize, n);

END;

Output



**5) Write a PL/SQL code to fund the sum of first N numbers**

DECLARE

x NUMBER;

n NUMBER;

i NUMBER;

FUNCTION find\_sum (n IN NUMBER)

RETURN NUMBER

IS

summation NUMBER: = 0;

BEGIN

FOR i IN 1...n

LOOP

summation: = i\*(i+1)/2;

END LOOP;

RETURN summation;

END;

BEGIN

n: = 10;

x: = find\_sum(n);

dbms\_output.put\_line('Sum of first 10 numbers is: ' || x);

END;

