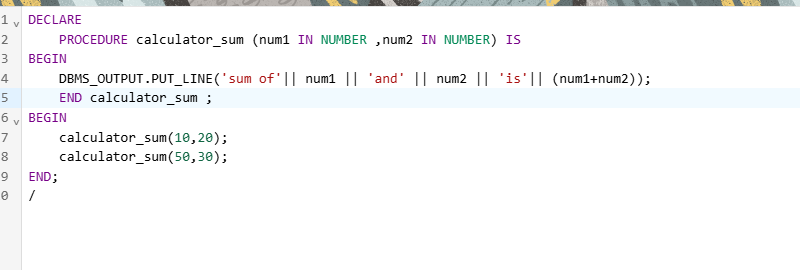
**QUESTION:1**

we define a local procedure within an anonymous plsql block to calculate and display the sum of two number  
  


DECLARE

PROCEDURE calculator\_sum (num1 IN NUMBER ,num2 IN NUMBER) IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('sum of'|| num1 || 'and' || num2 || 'is'|| (num1+num2));

END calculator\_sum ;

BEGIN

calculator\_sum(10,20);

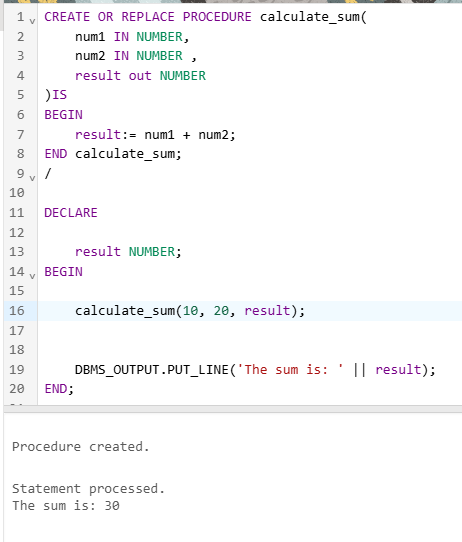
calculator\_sum(50,30);

END;

/

this example demostrate howto create a stored procedure that accept to number

calculate their sum and return the result



CREATE OR REPLACE PROCEDURE calculate\_sum(

num1 IN NUMBER,

num2 IN NUMBER ,

result out NUMBER

)IS

BEGIN

result:= num1 + num2;

END calculate\_sum;

/

DECLARE

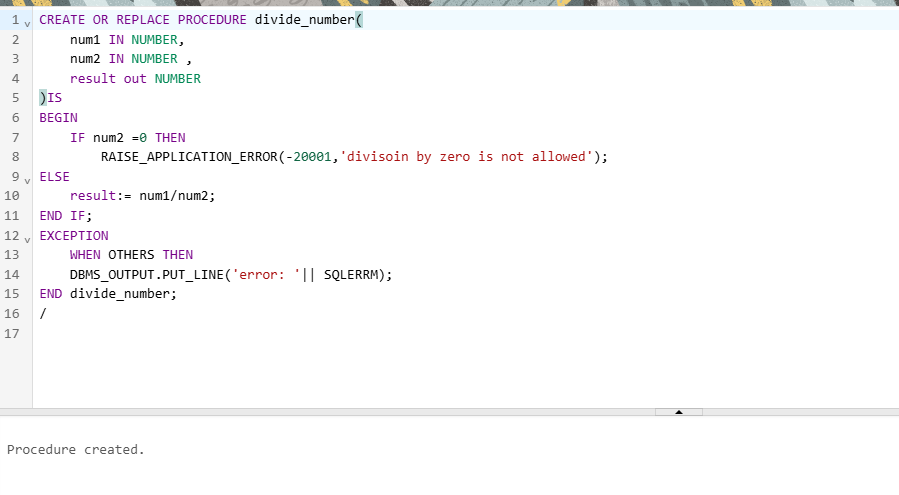
result NUMBER;

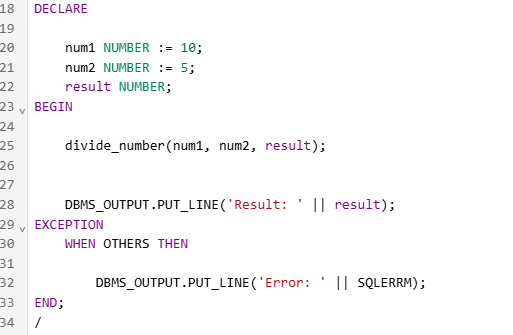
BEGIN

calculate\_sum(10, 20, result);

DBMS\_OUTPUT.PUT\_LINE('The sum is: ' || result);

END;





CREATE OR REPLACE PROCEDURE divide\_number(

num1 IN NUMBER,

num2 IN NUMBER ,

result out NUMBER

)IS

BEGIN

IF num2 =0 THEN

RAISE\_APPLICATION\_ERROR(-20001,'divisoin by zero is not allowed');

ELSE

result:= num1/num2;

END IF;

EXCEPTION

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('error: '|| SQLERRM);

END divide\_number;

/

DECLARE

num1 NUMBER := 10;

num2 NUMBER := 5;

result NUMBER;

BEGIN

divide\_number(num1, num2, result);

DBMS\_OUTPUT.PUT\_LINE('Result: ' || result);

EXCEPTION

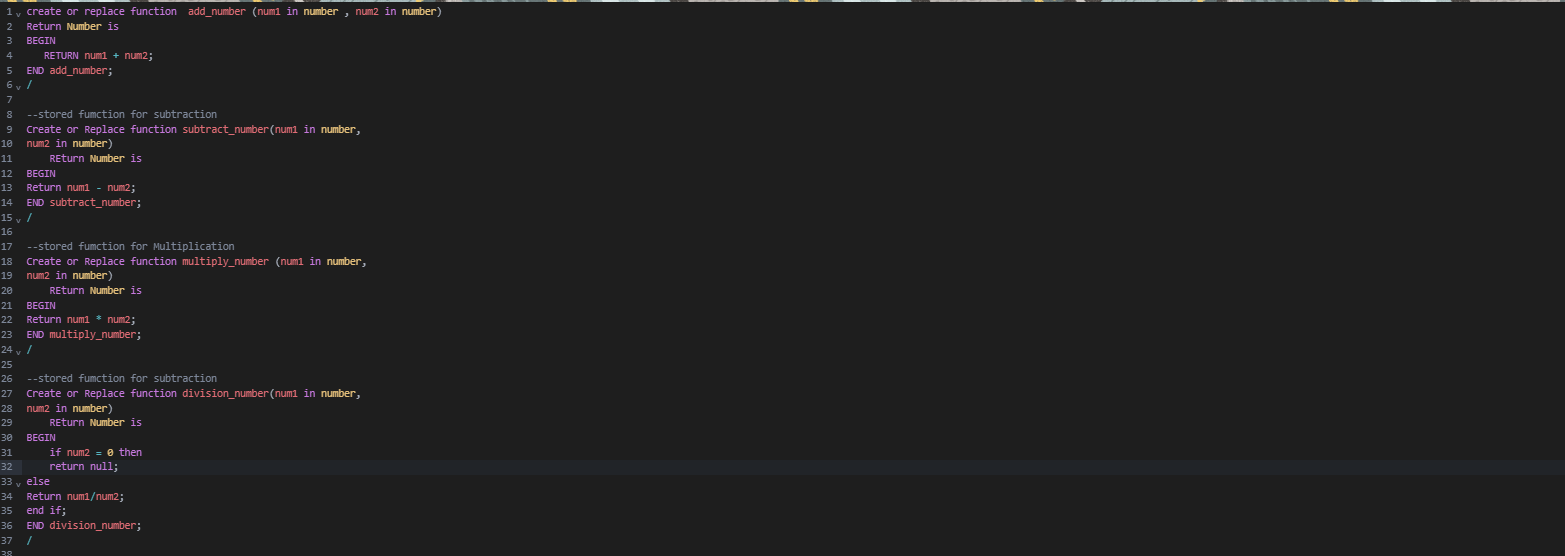
WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);

END;

/

**Question:3**



**create or replace function add\_number (num1 in number , num2 in number)**

**Return Number is**

**BEGIN**

**RETURN num1 + num2;**

**END add\_number;**

**/**

**--stored fumction for subtraction**

**Create or Replace function subtract\_number(num1 in number,**

**num2 in number)**

**REturn Number is**

**BEGIN**

**Return num1 - num2;**

**END subtract\_number;**

**/**

**--stored fumction for Multiplication**

**Create or Replace function multiply\_number (num1 in number,**

**num2 in number)**

**REturn Number is**

**BEGIN**

**Return num1 \* num2;**

**END multiply\_number;**

**/**

**--stored fumction for subtraction**

**Create or Replace function division\_number(num1 in number,**

**num2 in number)**

**REturn Number is**

**BEGIN**

**if num2 = 0 then**

**return null;**

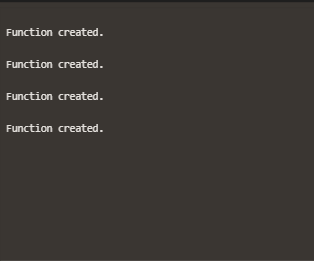
**else**

**Return num1/num2;**

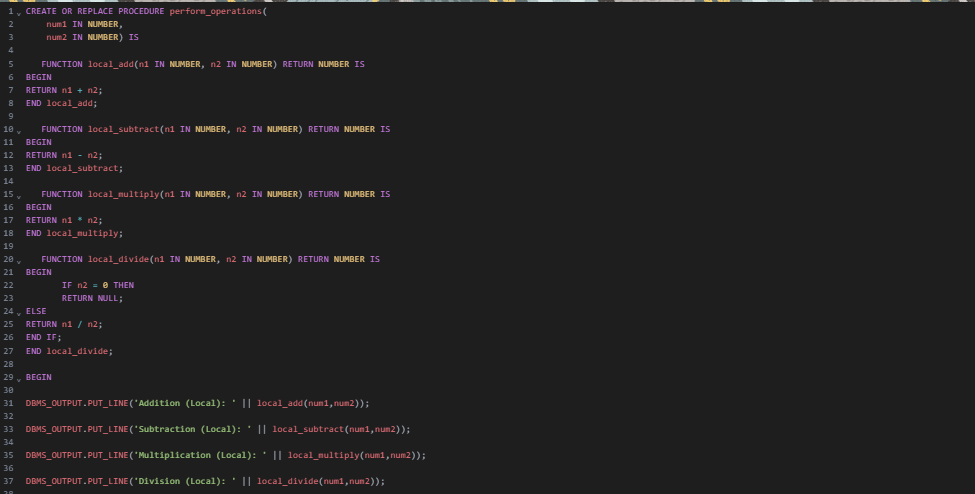
**end if;**

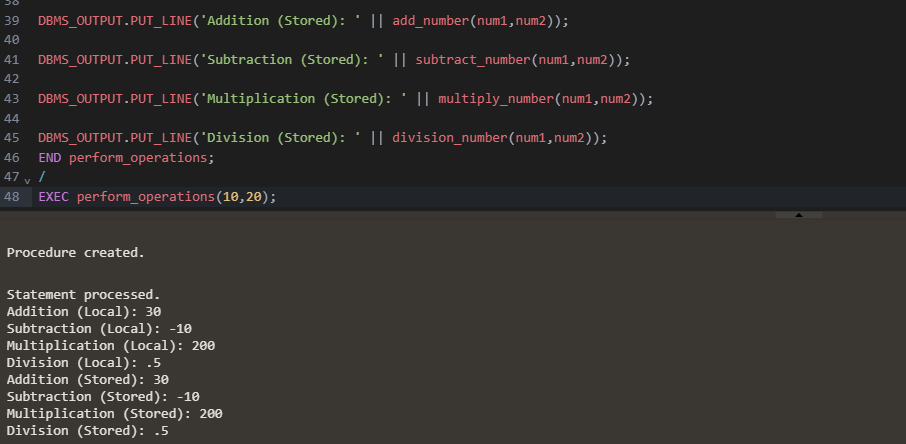
**END division\_number;**

**/**

****

**Question:4**

****

****

**CREATE OR REPLACE PROCEDURE perform\_operations(**

**num1 IN NUMBER,**

**num2 IN NUMBER) IS**

**FUNCTION local\_add(n1 IN NUMBER, n2 IN NUMBER) RETURN NUMBER IS**

**BEGIN**

**RETURN n1 + n2;**

**END local\_add;**

**FUNCTION local\_subtract(n1 IN NUMBER, n2 IN NUMBER) RETURN NUMBER IS**

**BEGIN**

**RETURN n1 - n2;**

**END local\_subtract;**

**FUNCTION local\_multiply(n1 IN NUMBER, n2 IN NUMBER) RETURN NUMBER IS**

**BEGIN**

**RETURN n1 \* n2;**

**END local\_multiply;**

**FUNCTION local\_divide(n1 IN NUMBER, n2 IN NUMBER) RETURN NUMBER IS**

**BEGIN**

**IF n2 = 0 THEN**

**RETURN NULL;**

**ELSE**

**RETURN n1 / n2;**

**END IF;**

**END local\_divide;**

**BEGIN**

**DBMS\_OUTPUT.PUT\_LINE('Addition (Local): ' || local\_add(num1,num2));**

**DBMS\_OUTPUT.PUT\_LINE('Subtraction (Local): ' || local\_subtract(num1,num2));**

**DBMS\_OUTPUT.PUT\_LINE('Multiplication (Local): ' || local\_multiply(num1,num2));**

**DBMS\_OUTPUT.PUT\_LINE('Division (Local): ' || local\_divide(num1,num2));**

**DBMS\_OUTPUT.PUT\_LINE('Addition (Stored): ' || add\_number(num1,num2));**

**DBMS\_OUTPUT.PUT\_LINE('Subtraction (Stored): ' || subtract\_number(num1,num2));**

**DBMS\_OUTPUT.PUT\_LINE('Multiplication (Stored): ' || multiply\_number(num1,num2));**

**DBMS\_OUTPUT.PUT\_LINE('Division (Stored): ' || division\_number(num1,num2));**

**END perform\_operations;**

**/**

**EXEC perform\_operations(10,20);**