

All Online Learning

www.allonlinelearning.com

PL/SQL Function

- The PL/SQL Function is very similar to PL/SQL Procedure.
- The main difference between procedure and a function is, a function must always return a value, and on the other hand a procedure may or may not return a value.
- Except this, all the other things of PL/SQL procedure are true for PL/SQL function too.

Syntax to create a function:

```
CREATE [OR REPLACE] FUNCTION function_name [parameters]
[(parameter_name [IN | OUT | IN OUT] type [, ...])]
RETURN return_datatype
{IS | AS}
BEGIN
    < function_body >
END [function_name];
```

Here:

- **Function_name:** specifies the name of the function.
- **[OR REPLACE]** option allows modifying an existing function.
- The **optional parameter list** contains name, mode and types of the parameters.
- **IN** represents that value will be passed from outside and **OUT** represents that this parameter will be used to return a value outside of the procedure.

The function must contain a return statement.

- **RETURN** clause specifies that data type you are going to return from the function.
- Function body contains the executable part.
- The **AS** keyword is used instead of the **IS** keyword for creating a standalone function.

PL/SQL Function Example

```
create or replace function adder(n1 in number, n2 in number)
return number
is
n3 number(8);
begin
n3 :=n1+n2;
return n3;
end;
/
```

All Online Learning

www.allonlinelearning.com

Now write another program to **call the function**.

DECLARE

n3 number(2);

BEGIN

n3 := adder(11,22);

dbms_output.put_line('Addition is: ' || n3);

END;

/

Output:

Addition is: 33

Statement processed.

0.05 seconds

Another PL/SQL Function Example

DECLARE

a number;

b number;

c number;

FUNCTION findMax(x IN number, y IN number)

RETURN number

IS

z number;

BEGIN

IF x > y **THEN**

z:= x;

ELSE

Z:= y;

END IF;

RETURN z;

END;

BEGIN

a:= 23;

b:= 45;

c := findMax(a, b);

dbms_output.put_line(' Maximum of (23,45): ' || c);

END;

/

All Online Learning

www.allonlinelearning.com

Output:

Maximum of (23,45): 45
Statement processed.
0.02 seconds

PL/SQL function example using table

Customers			
Id	Name	Department	Salary
1	alex	web developer	35000
2	ricky	program developer	45000
3	mohan	web designer	35000
4	dilshad	database manager	44000

```
CREATE OR REPLACE FUNCTION totalCustomers
RETURN number IS
    total number(2) := 0;
BEGIN
    SELECT count(*) INTO total
    FROM customers;
    RETURN total;
END;
/
```

Calling PL/SQL Function:

```
DECLARE
    c number(2);
BEGIN
    c := totalCustomers();
    dbms_output.put_line('Total no. of Customers: ' || c);
END;
/
```

Output:

Total no. of Customers: 4
PL/SQL procedure successfully completed.

```
DROP FUNCTION function_name;
```