# Difference between Functions and Procedures

| **Key** | **Function** | **Procedure** |
| --- | --- | --- |
| Definition | A function is used to calculate result using given inputs. | A procedure is used to perform certain task in order. |
| Call | A function can be called by a procedure. | A procedure cannot be called by a function. |
| DML | DML statements cannot be executed within a function. | DML statements can be executed within a procedure. |
| SQL, Query | A function can be called within a query. | A procedure cannot be called within a query. |
| SQL, Call | Whenever a function is called, it is first compiled before being called. | A procedure is compiled once and can be called multiple times without being compiled. |
| SQL, Return | A function returns a value and control to calling function or code. | A procedure returns the control but not any value to calling function or code. |
| try-catch | A function has no support for try-catch | A procedure has support for try-catch blocks. |
| SELECT | A select statement can have a function call. | A select statement can't have a procedure call. |
| Explicit Transaction Handling | A function can does not have explicit transaction handling. | A procedure can use explicit transaction handling. |

# Functions

use analyticsdb;

show tables;

select \* from sales;

## -- Create function to get a new status flag to determine the type the sales done

Delimiter //

create function add\_sales\_count2(sales\_count DECIMAL(38,0))

returns VARCHAR(15)

deterministic

begin

declare sales\_status varchar(15);

if sales\_count > 0 and sales\_count< 100 then

set sales\_status = "Low Sales";

elseif sales\_count > 100 and sales\_count < 300 then

set sales\_status = "Moderate Sales";

elseif sales\_count > 400 and sales\_count < 600 then

set sales\_status = "Good Sales";

else

set sales\_status = "Excellent Sales";

End if;

return sales\_status;

end//

-- This will display a new column sales\_status which will hold values returned from function add\_sales\_count which accepts sales parameters

select order\_id,sales, add\_sales\_count(sales) as sales\_status from sales;

## -- Create new function to calculate Net\_profit

desc sales;

Delimiter //

create function net\_profit\_1( sales\_val decimal(38,0), discount\_val decimal(38,3),shipping\_cost\_val decimal(38,2), profit\_val decimal(38,8))

returns decimal(38,8)

begin

declare total\_profit\_val decimal(38,8);

set total\_profit\_val = 0;

set total\_profit\_val = profit\_val - shipping\_cost\_Val - (discount\_val \* sales\_val);

return total\_profit\_val;

end//

select order\_id, sales, discount,shipping\_cost, profit , net\_profit\_1(sales, discount,shipping\_cost, profit) as net\_profit from sales;

# #Stored Procedures

DELIMITER //

create procedure get\_order\_details2(amt Int)

Begin

select orderid, customerid, amount from orders where amount > amt;

End //

## #syntax to call Stored Procedures

call get\_order\_details2(10);

## #Stored Procedures with input and output variable

DELIMITER //

create procedure get\_order\_details5(IN amt Int, OUT orderid\_o varchar(3))

Begin

select orderid from orders where amount= amt into orderid\_o;

End //

call get\_order\_details5(50,@order\_id\_o);

select @order\_id\_o as order\_id\_output;

## #DROP Procedures

DROP PROCEDURE IF EXISTS get\_order\_details1;

## -- Procedure to create table and get values by calling function

Delimiter //

create procedure add\_net\_profit1()

begin

create table net\_profit\_1 (order\_id varchar(15), net\_profit\_1f decimal(38,8));

insert into net\_profit\_1

select order\_id, net\_profit\_1(sales, discount,shipping\_cost, profit) from sales;

end//

call add\_net\_profit1();

select \* from net\_profit\_1;

select \* from sales;

desc sales;

-

## --Procedure to add values into a database

Delimiter //

create procedure add\_net\_profit2()

begin

-- alter table sales add column (net\_profit decimal(38,8));

insert into sales(net\_profit) values(select b.net\_profit\_1f from net\_profit\_1 b , sales a where a.order\_id= b.order\_id);

end//

call add\_net\_profit2();