**Aim**: SQL Functions and Procedures

**Objectives:**

The purpose of learning SQL Functions and Procedures is to understand how to simplify and organize database operations. Functions and procedures help developers write reusable code directly in the database, making tasks like calculations, data processing, or enforcing rules easier and faster. The goal is to learn how to use these tools to avoid repetition, improve efficiency, and ensure the database works smoothly.

**Tools Used:**

* MySQL Workbench

**Concept:**

SQL Functions and Procedures are features that allow you to group tasks into a single block of code.

* **Functions**: These are used to perform a specific task and return a single value, like calculating a total or formatting text.
* **Procedures**: These can perform multiple tasks, like inserting or updating records, and they don’t always return a value.

Using functions and procedures makes it easier to handle repetitive tasks, follow business rules, and keep the database organized and efficient.

**Example:**

Function  
CREATE DEFINER=`root`@`localhost` FUNCTION `cubeCalutaor`( num int) RETURNS int

DETERMINISTIC

BEGIN

declare cube\_of\_num int;

set cube\_of\_num = num \* num \* num;

RETURN cube\_of\_num;

END  
  
Procedure

CREATE DEFINER=`root`@`localhost` PROCEDURE `Get\_Orders\_Status`(IN input\_status varchar(20))

BEGIN

select

year(orderDate) as Year,

count(\*) as Total\_Orders

from

orders

where

status = input\_status

group by

year(orderDate);

END

**Problem Statement:**

Assignment on Function Question:

1) Create a function to find the cube of a number.

2) Use Classicmodels. Create a function which will return city of the given officeCode.

3) Use Classicmodels. Create a function to show the highest MSRP for each productline using window functions.

4) Use Classicmodels. Create a function to show the customername who has used thehighest CreditLimit.

Assignment on Procedure Question:

Use classicmodels. Create a procedure Get\_Orders\_Status which should accept the status value from user and show the number of orders for each year for that status.

Table- Orders The output should look as shown in below image.



**Solution:**

Function

1)

CREATE DEFINER=`root`@`localhost` FUNCTION `cubeCalutaor`( num int) RETURNS int

DETERMINISTIC

BEGIN

declare cube\_of\_num int;

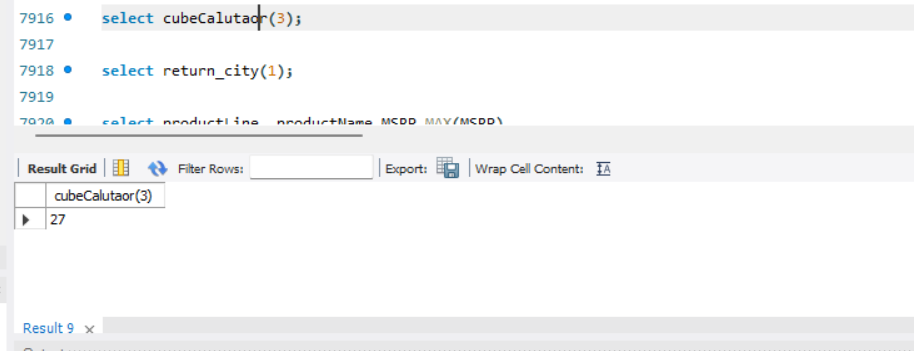
set cube\_of\_num = num \* num \* num;

RETURN cube\_of\_num;

END

--------

select cubeCalutaor(3);



2)

CREATE DEFINER=`root`@`localhost` FUNCTION `return\_city`(officeCode\_entered int) RETURNS varchar(50) CHARSET latin1

DETERMINISTIC

BEGIN

declare city\_op varchar(50);

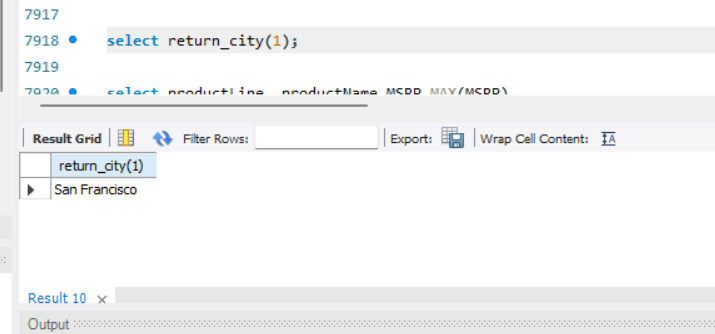
select city into city\_op from offices where officeCode = officeCode\_entered;

RETURN city\_op;

END

------------

select return\_city(1);



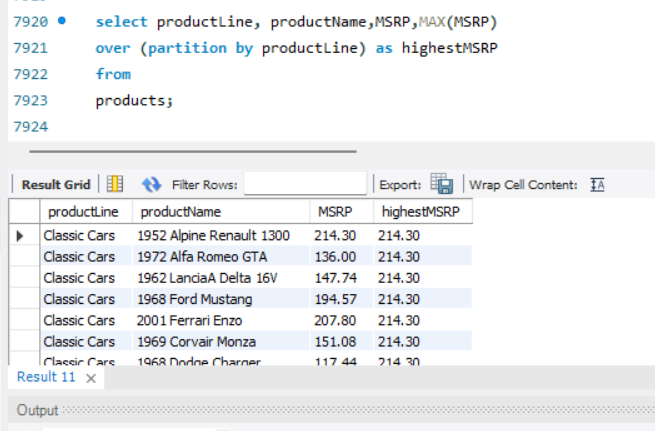
3)

select productLine, productName,MSRP,MAX(MSRP)

over (partition by productLine) as highestMSRP

from

products;



4)

CREATE DEFINER=`root`@`localhost` FUNCTION `highest\_credit\_customer`() RETURNS varchar(50) CHARSET latin1

DETERMINISTIC

BEGIN

DECLARE customerName\_op varchar(50);

SELECT customerName into customerName\_op

FROM customers

ORDER BY CreditLimit DESC

LIMIT 1;

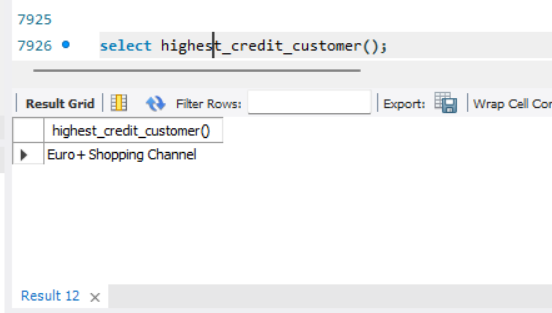
RETURN customerName\_op

;

END

--------------------

select highest\_credit\_customer();



Procedure

CREATE DEFINER=`root`@`localhost` PROCEDURE `Get\_Orders\_Status`(IN input\_status varchar(20))

BEGIN

select

year(orderDate) as Year,

count(\*) as Total\_Orders

from

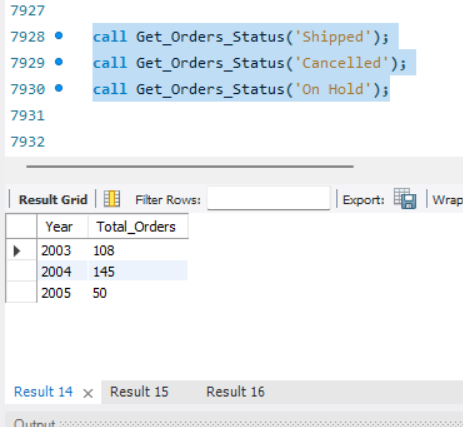
orders

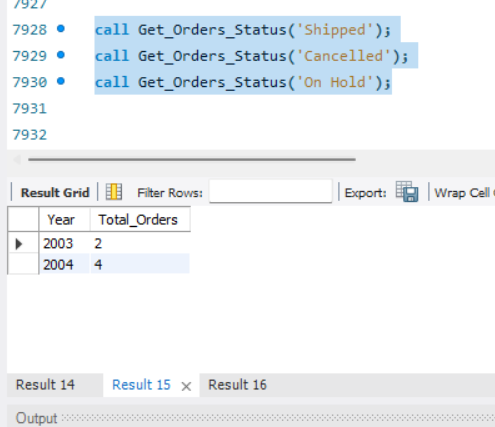
where

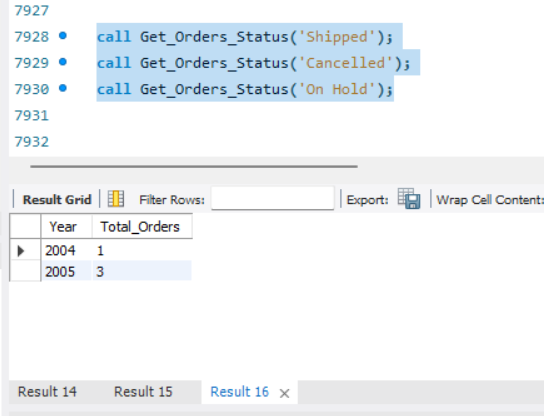
status = input\_status

group by

year(orderDate);

END





**Observation:**

Functions and procedures in SQL are important tools that help make database operations easier and more efficient. They allow you to group tasks into reusable blocks of code. Functions are great for doing calculations or getting a specific result, while procedures are used for performing multiple steps or tasks. By using these, you can save time, avoid repeating code, and make sure the database runs smoothly. Functions and procedures also help keep data accurate and organized, reducing errors and improving the system’s reliability.