STORED PROCEDURES

* **Create stored procedure**

A stored procedure is a prepared SQL code that you can save, so the code can be reused over and over again .

So if you have an SQL query that you write over and over again, save it as a stored procedure, and then just call it to execute it.

**Syntax :**

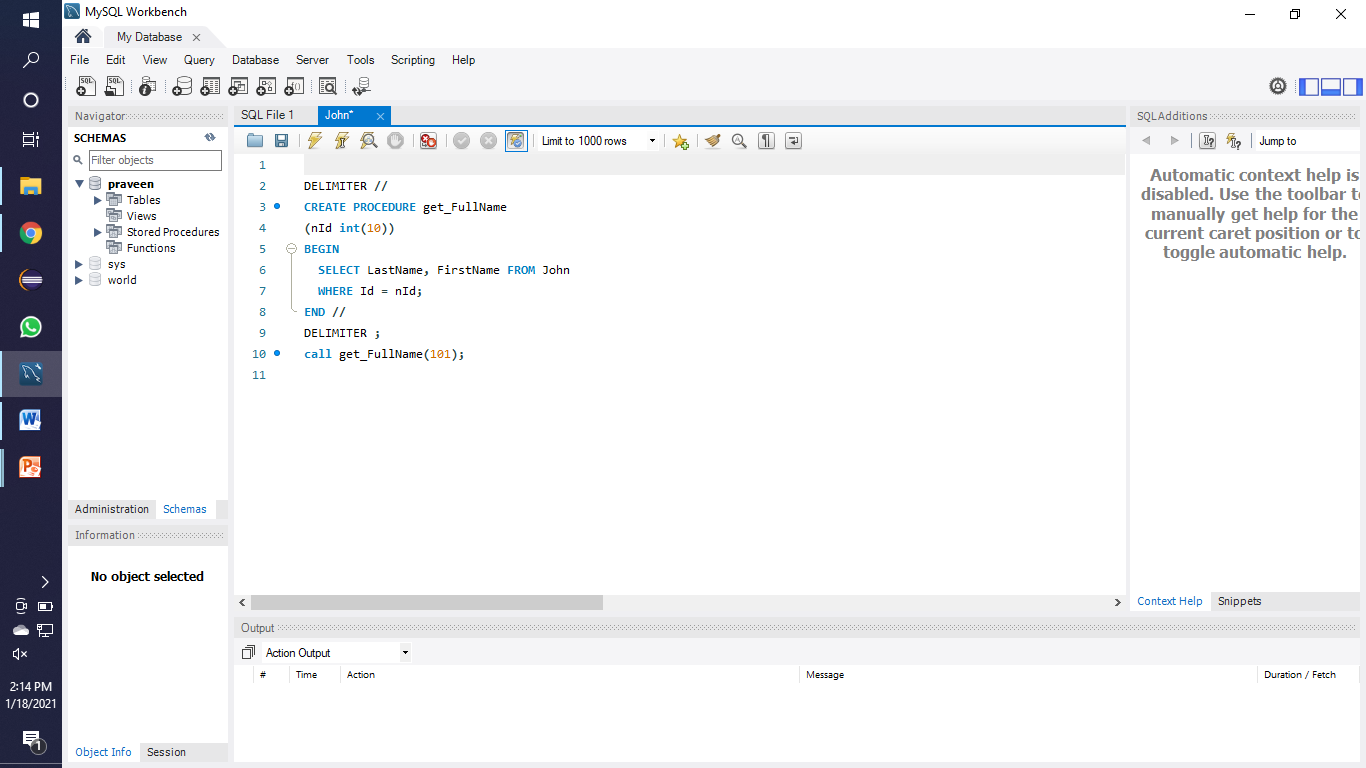
Stored procedures are declared using the following syntax:

Create Procedure <proc-name> ()

begin \

-- execution code

end;



* **Set-up Queries in stored Procedure**
* **With parameters:**

CALL `employee`.`getDetails`(104, @ename);

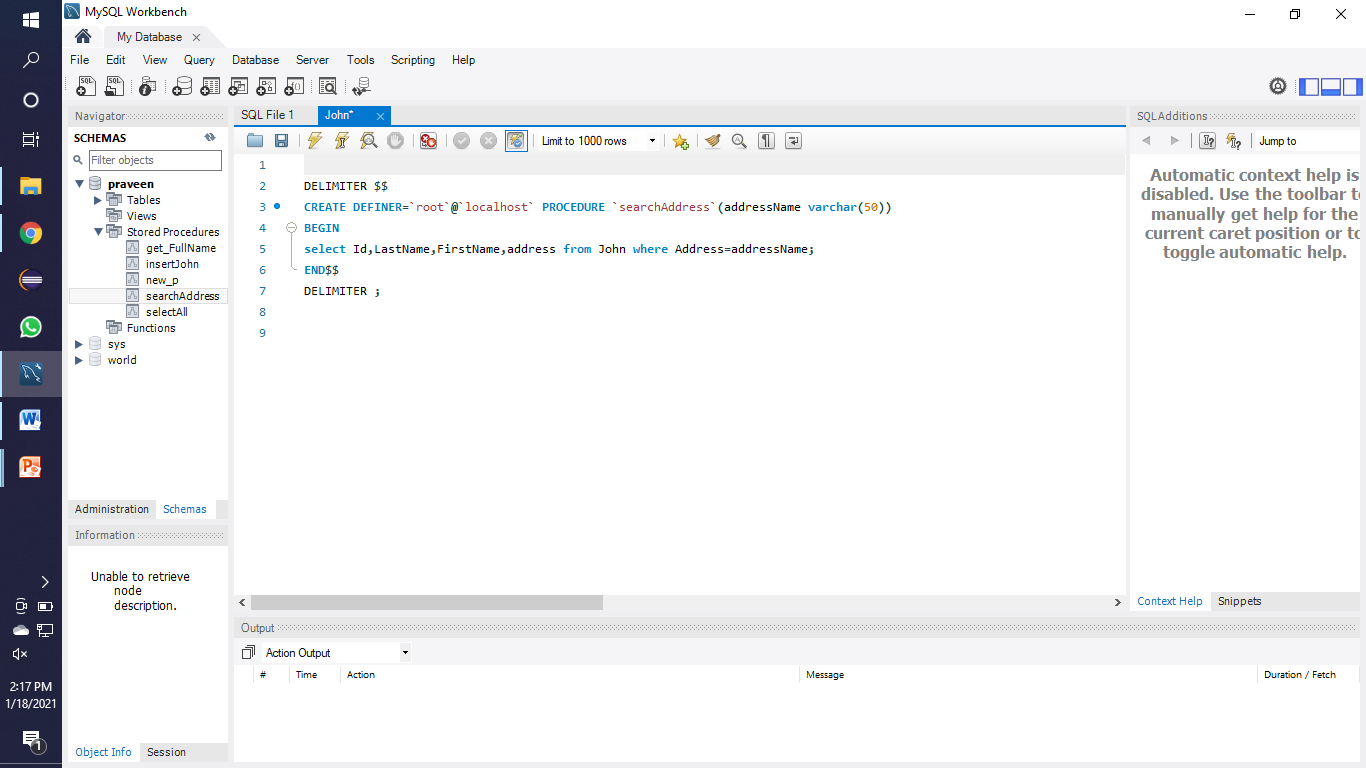
select @ename as employeeName;

CREATE PROCEDURE `getDetails`(IN emp\_id int, OUT ename varchar(225))

BEGIN

select concat(FirstName," ",ltName) into ename from employeedetail where employee\_id=emp\_id;

END



With Out Parameters :

Syntax :

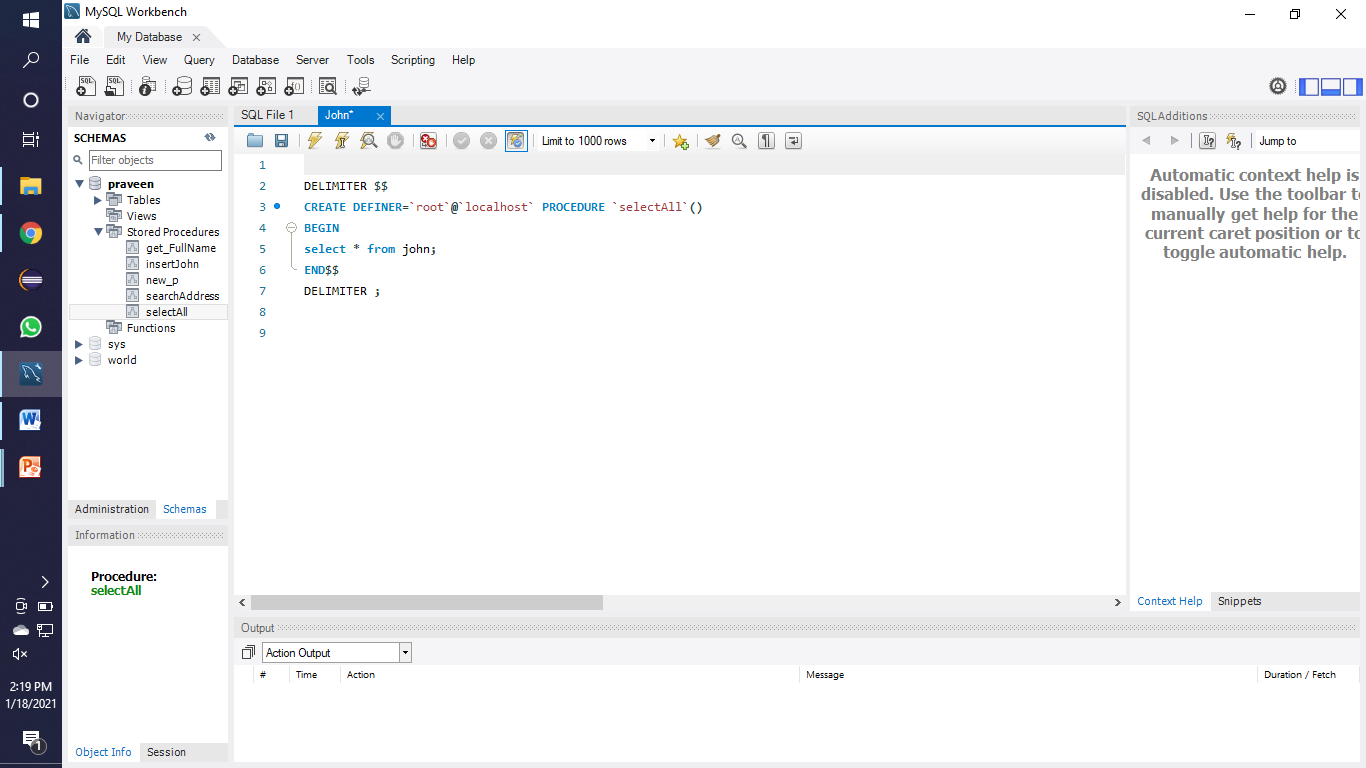
DELIMITER $$(without parameters) CREATE PROCEDURE `test`()

BEGIN

select \*from department;

END$$

DELIMITER ;



* **Optimize stored procedure Performance**
* Use stored procedures instead of heavy-duty queries.
* Call stored procedures using their fully qualified name.
* Include the SET NOCOUNT ON statement in your stored procedures to stop the message indicating the number of rows affected by a Transact-SQL statement.
* Use the sp\_executesql stored procedure instead of the EXECUTE statement.
* Try to avoid using temporary tables inside your stored procedures.
* Try to avoid using DDL (Data Definition Language) statements inside your stored procedure

**Advantages of stored procdures**

* **Reusability**
* **Transparent**
* **Traffic reducing**
* **Secure**
* **Portable**
* **Availability**

Stored procedures are reusable and transparent to any applications.

* Stored procedures expose the database interface to all applications so that developers don’t have to develop functions that are already supported in stored procedures
* Stored procedures helps reduce the traffic between application and database server because instead of sending multiple lengthy SQL statements, the application has to send only name and parameters of the stored procedure.
* Stored procedures are secure. Database administrator can grant appropriate permissions to applications that access stored procedures in the database without giving any permission on the underlying database tables.
* Stored procedures are portable. When you write your stored procedure in SQL, it will run on every platform that MySQL runs on. That's the advantage of writing in SQL rather than in an external language like Java or C or PHP.
* Stored procedures are always available as 'source code' in the database itself. And it makes sense to link the data with the processes that operate on the data.

**Operations On Stored Procedures :**

1. To list stored procedures of the databases that we have the privilage to access>

syntax: SHOW PROCEDURE STATUS

2. To show stored procedures in a particular database.

syntax: SHOW PROCEDURE STATUS WHERE db='database';

Example: SHOW PROCEDURE STATUS WHERE db='employee';

3.To show stored procedures that have a particular pattern.

syntax: SHOW PROCEDURE STATUS WHERE name LIKE '%pattern%';

example:SHOW PROCEDURE STATUS WHERE name LIKE '%All%';

4. Display source code of stored procedure.

syntax: SHOW CREATE PROCEDURE stored\_procedure\_name;

example:SHOW CREATE PROCEDURE getDetails;

5. Drop stored procedure.

syntax: DROP PROCEDURE IF EXISTS procedure\_name;

example: DROP PROCEDURE IF EXISTS getDetails;

6. Calling stored procedure.

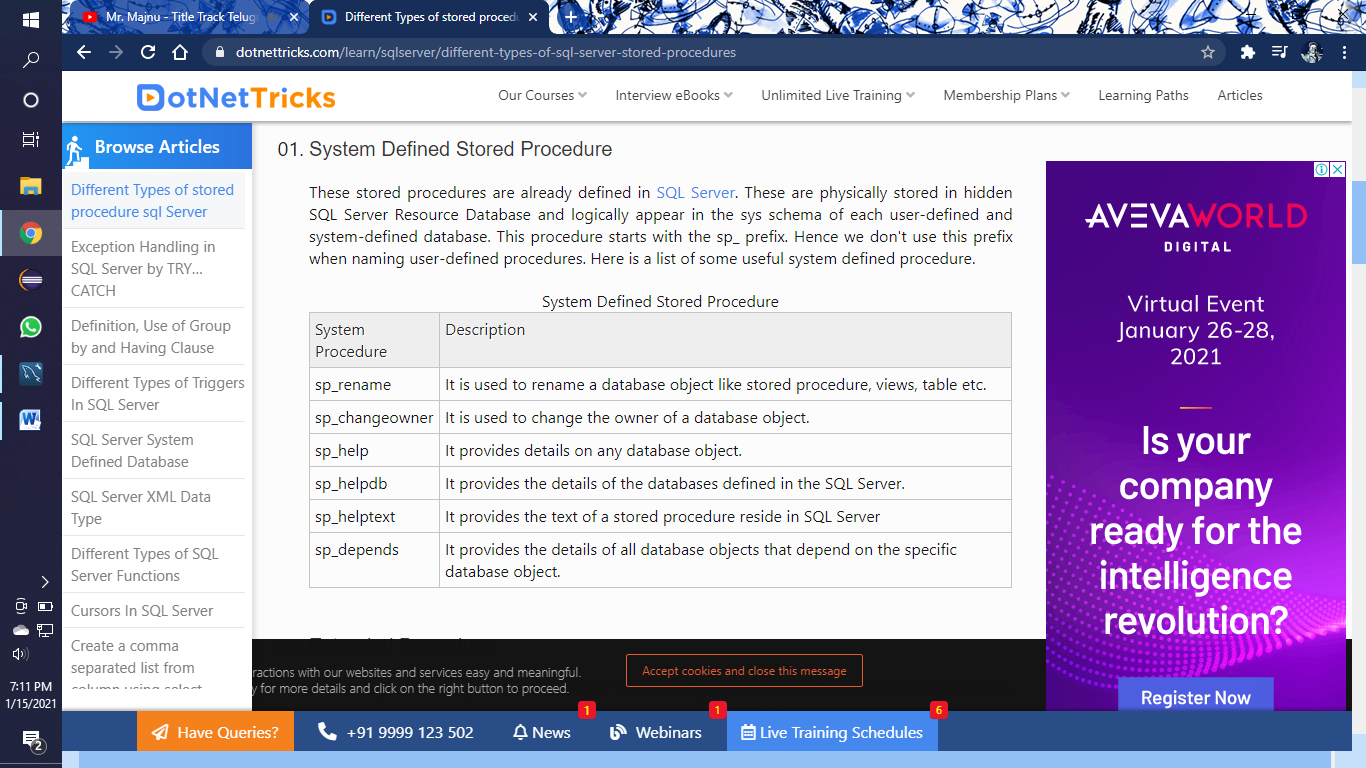
syntax: CALL stored\_procedure\_name;

EXAMPLE: CALL getAllDetails();

TYPES OF STORED PROCEDURES

1. System Defined Stored Procedure

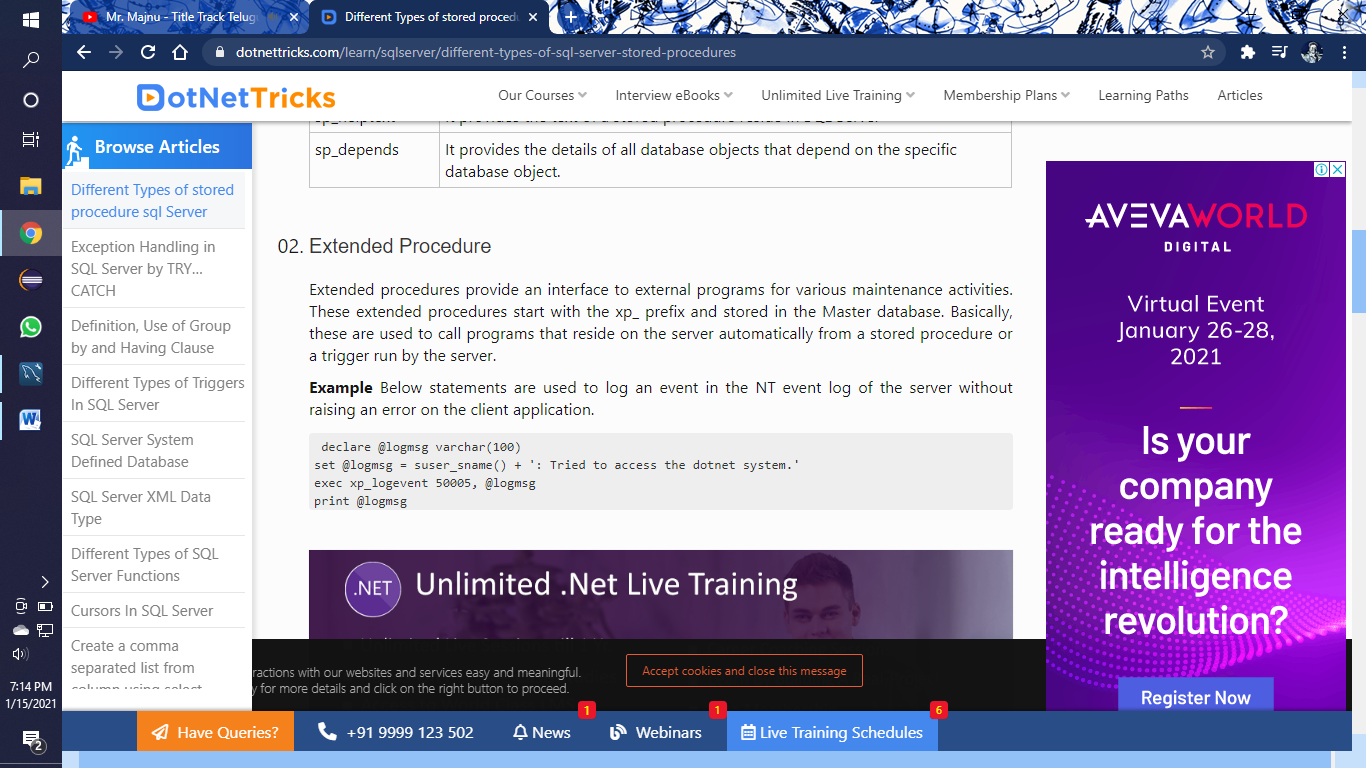
These stored procedures are already defined in[SQL Server](https://www.dotnettricks.com/learn/sqlserver). These are physically stored in hidden SQL Server Resource Database and logically appear in the sys schema of each user-defined and system-defined database. This procedure starts with the sp\_ prefix. Hence we don't use this prefix when naming user-defined procedures. Here is a list of some useful system defined procedure.



## Extended Procedure

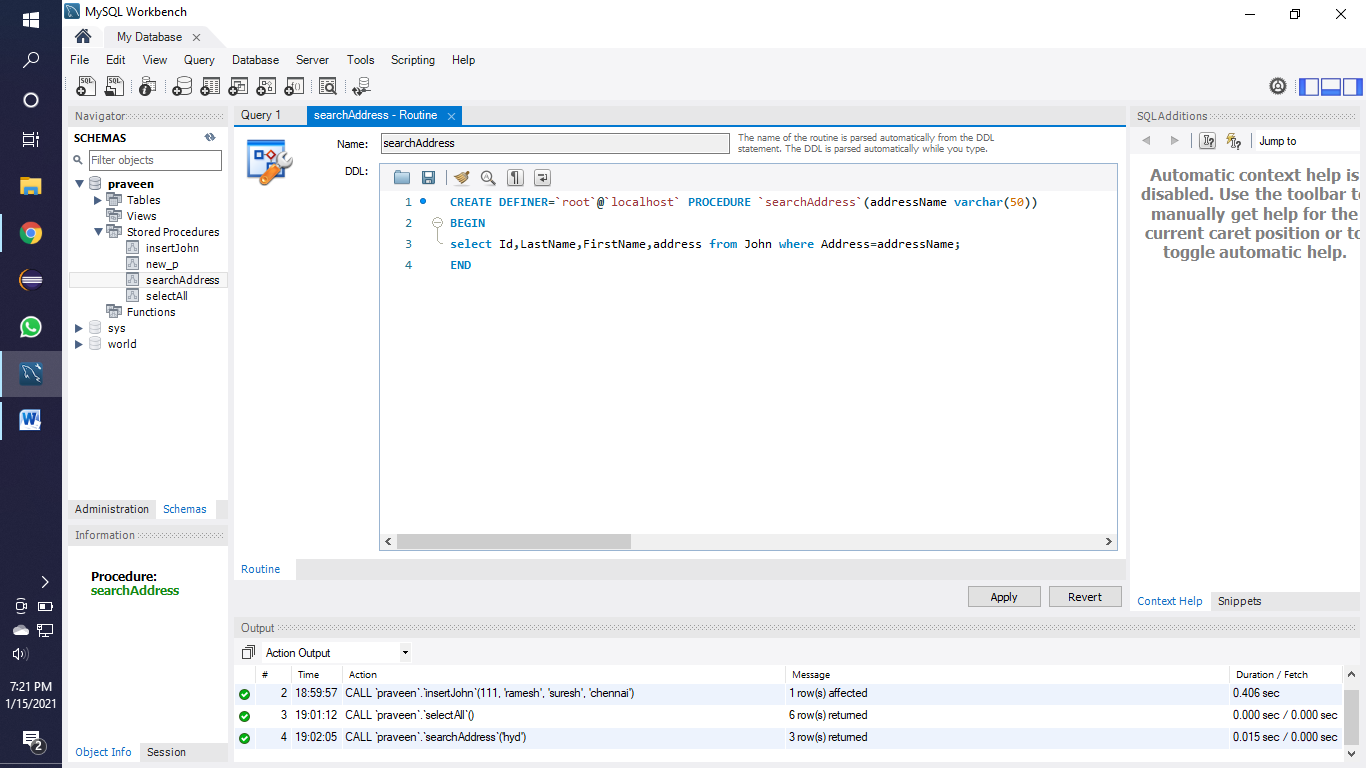
Extended procedures provide an interface to external programs for various maintenance activities. These extended procedures start with the xp\_ prefix and stored in the Master database. Basically, these are used to call programs that reside on the server automatically from a stored procedure or a trigger run by the server.

**Example**Below statements are used to log an event in the NT event log of the server without raising an error on the client application



## User-Defined Stored Procedure

These procedures are created by the user for own actions. These can be created in all system databases except the Resource database or in a user-defined database.



## CLR Stored Procedure

## CLR stored procedure is a special type of procedure that is based on the CLR (Common Language Runtime) in [.net framework](https://www.dotnettricks.com/learn/netframework). CLR integration of procedure was introduced with SQL Server 2008 and allow for the procedure to be coded in one of .NET languages like [C#](https://www.dotnettricks.com/learn/csharp), Visual Basic and F#