# **Sql Server - How To Write a Stored Procedure in SQL Server**



[**Vivek Johari**](https://www.codeproject.com/script/Membership/View.aspx?mid=3667781), 12 Nov 2010

|  |  |
| --- | --- |
|  | 4.80 (113 votes) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | |  |  |  | | --- | --- | --- | | Rate this: | [vote 1vote 2vote 3vote 4vote 5](https://www.codeproject.com/articles/126898/sql-server-how-to-write-a-stored-procedure-in-sql#) |  | |

Stored Procedure in SQL Server can be defined as the set of logically group of SQL statement which are grouped to perform a specific task. This article will describe the Stored Procedure in SQL Server and their benefits.

**Stored Procedure**: Stored Procedure in SQL Server can be defined as the set of logical group of SQL statements which are grouped to perform a specific task. There are many benefits of using a stored procedure. The main benefit of using a stored procedure is that it increases the performance of the database.The other benefits of using the Stored Procedure are given below.

## **Benefits of Using the Stored Procedure**

1. One of the main benefits of using the Stored procedure is that it reduces the amount of information sent to the database server. It can become a more important benefit when the bandwidth of the network is less. Since if we send the SQL query (statement) which is executing in a loop to the server through network and the network gets disconnected, then the execution of the SQL statement doesn't return the expected results, if the SQL query is not used between Transaction statement and rollback statement is not used.
2. Compilation step is required only once when the stored procedure is created. Then after it does not require recompilation before executing unless it is modified and reutilizes the same execution plan whereas the SQL statements need to be compiled every time whenever it is sent for execution even if we send the same SQL statement every time.
3. It helps in re usability of the SQL code because it can be used by multiple users and by multiple clients since we need to just call the stored procedure instead of writing the same SQL statement every time. It helps in reducing the development time.
4. Stored procedure is helpful in enhancing the security since we can grant permission to the user for executing the Stored procedure instead of giving permission on the tables used in the Stored procedure.
5. Sometimes, it is useful to use the database for storing the business logic in the form of stored procedure since it makes it secure and if any change is needed in the business logic, then we may only need to make changes in the stored procedure and not in the files contained on the web server.

## **How to Write a Stored Procedure in SQL Server**

Suppose there is a table called tbl\_Students whose structure is given below:

Hide Copy Code

CREATE TABLE tbl\_Students  
  
(  
 [Studentid] [int] IDENTITY(1,1) NOT NULL,  
 [Firstname] [nvarchar](200) NOT NULL,  
 [Lastname] [nvarchar](200) NULL,  
 [Email] [nvarchar](100) NULL  
)

Support we insert the following data into the above table:

Hide Copy Code

Insert into tbl\_Students (Firstname, lastname, Email)  
 Values('Vivek', 'Johari', 'vivek@abc.com')  
  
Insert into tbl\_Students (Firstname, lastname, Email)  
 Values('Pankaj', 'Kumar', 'pankaj@abc.com')  
  
Insert into tbl\_Students (Firstname, lastname, Email)  
 Values('Amit', 'Singh', 'amit@abc.com')  
  
Insert into tbl\_Students (Firstname, lastname, Email)  
 Values('Manish', 'Kumar', 'manish@abc.comm')  
  
Insert into tbl\_Students (Firstname, lastname, Email)  
 Values('Abhishek', 'Singh', 'abhishek@abc.com')

Now, while writing a Stored Procedure, the first step will be to write the Create Procedure statement as the first statement:

Hide Copy Code

Create Procedure Procedure-name   
(  
Input parameters ,  
Output Parameters (If required)  
)  
As  
Begin  
 Sql statement used in the stored procedure  
End

Now, suppose we need to create a Stored Procedure which will return a student name whose studentid is given as the input parameter to the stored procedure. Then, the Stored Procedure will be:

Hide Copy Code

*/\* Getstudentname is the name of the stored procedure\*/*  
  
Create PROCEDURE Getstudentname(  
  
@studentid INT *--Input parameter , Studentid of the student*   
  
)  
AS  
BEGIN  
SELECT Firstname+' '+Lastname FROM tbl\_Students WHERE studentid=@studentid   
END

We can also collect the student name in the output parameter of the Stored Procedure. For example:

Hide Copy Code

*/\*   
GetstudentnameInOutputVariable is the name of the stored procedure which  
uses output variable @Studentname to collect the student name returns by the  
stored procedure  
\*/*  
  
Create PROCEDURE GetstudentnameInOutputVariable  
(  
  
@studentid INT, *--Input parameter , Studentid of the student*  
@studentname VARCHAR(200) OUT *-- Out parameter declared with the help of OUT keyword*  
)  
AS  
BEGIN  
SELECT @studentname= Firstname+' '+Lastname FROM tbl\_Students WHERE studentid=@studentid  
END

**Note:-/\* \*/** is used to write comments in one or multiple lines

**--** is used to write a comment in a single line

## **How to Alter a Stored Procedure in a SQL Server**

In SQL Server, a stored procedure can be modified with the help of the Alter **keyword**. Now if we want to get student email address through the same procedure GetstudentnameInOutputVariable. So we need to modify it by adding one more output parameter " @StudentEmail " which is shown below:

Hide Copy Code

*/\*   
Stored Procedure GetstudentnameInOutputVariable is modified to collect the  
email address of the student with the help of the Alert Keyword  
\*/*   
  
Alter PROCEDURE GetstudentnameInOutputVariable  
(  
  
@studentid INT, *--Input parameter , Studentid of the student*  
@studentname VARCHAR (200) OUT, *-- Output parameter to collect the student name*  
@StudentEmail VARCHAR (200)OUT *-- Output Parameter to collect the student email*  
)  
AS  
BEGIN  
SELECT @studentname= Firstname+' '+Lastname,   
 @StudentEmail=email FROM tbl\_Students WHERE studentid=@studentid  
END

**Note**: It is not necessary that a stored procedure will have to return. There can be a case when a stored procedure doesn't returns anything. For example, a stored procedure can be used to Insert, delete or update a SQL statement. For example, the below stored procedure is used to insert value into the table tbl\_students.

Hide Copy Code

*/\*  
This Stored procedure is used to Insert value into the table tbl\_students.   
\*/*  
  
Create Procedure InsertStudentrecord  
(  
 @StudentFirstName Varchar(200),  
 @StudentLastName Varchar(200),  
 @StudentEmail Varchar(50)  
)   
As  
 Begin  
 Insert into tbl\_Students (Firstname, lastname, Email)  
 Values(@StudentFirstName, @StudentLastName,@StudentEmail)  
 End

## **Execution of the Stored Procedure in SQL Server**

### Execution of the Stored Procedure which doesn't have an Output Parameter

A stored procedure is used in the SQL Server with the help of the "Execute" or "Exec" Keyword. For example, if we want to execute the stored procedure "Getstudentname", then we will use the following statement.

Hide Copy Code

Execute Getstudentname 1  
Exec Getstudentname 1

### Execution of the Stored Procedure using the Output Parameter

If we want to execute the Stored procedure "GetstudentnameInOutputVariable" , then we first need to declare the variable to collect the output values. For example:

Hide Copy Code

Declare @Studentname as nvarchar(200) *-- Declaring the variable to collect the Studentname*  
Declare @Studentemail as nvarchar(50) *-- Declaring the variable to collect the Studentemail*  
Execute GetstudentnameInOutputVariable 1 , @Studentname output, @Studentemail output  
select @Studentname,@Studentemail *-- "Select" Statement is used to show the output from Procedure*

## **Summary**

In the end, we can say that a Stored procedure not only enhances the possibility of reusing the code and execution plan, but it also increases the performance of the database by reducing the traffic of the network by reducing the amount of information sent over the network.