Introduction to Stored Procedures in SQL Server

tutorialgateway.org/stored-procedures-in-sql

The Stored Procedures in SQL Server are saved collections of one, or group of SQL statements stored in the SQL Server. In this article we will explain you, How to create, rename, Modify, and delete Stored Procedures in SQL Server with examples.

We are going to use the below shown data for this demonstration

```
USE [SQL Tutorial]
GO

SELECT [FirstName]
      ,[LastName]
      ,[Education]
      ,[Occupation]
      ,[YearlyIncome]
      ,[Sales]
      ,[HireDate]
FROM [Employee]
```

100 %

Results | Messages ©tutorialgateway.org

	FirstName	LastName	Education	Occupation	YearlyIncome	Sales	HireDate
1	John	Yang	Bachelors	Professional	90000	3578.27	2006-01-28
2	Rob	Johnson	Bachelors	Management	80000	3399.99	2010-12-29
3	Ruben	Torres	Partial College	Skilled Manual	50000	699.0982	2011-12-29
4	Christy	Zhu	Bachelors	Professional	80000	3078.27	2012-12-28
5	Rob	Huang	High School	Skilled Manual	60000	2319.99	2008-09-22
6	John	Ruiz	Bachelors	Professional	70000	539.99	2009-07-06
7	Tutorial	Gateway	Masters Degree	Management	80000	2320.49	2009-08-12
8	Christy	Mehta	Partial High School	Clerical	50000	24.99	2007-07-05
9	Rob	Verhoff	Partial High School	Clerical	45000	24.99	2013-09-15
10	Christy	Carlson	Graduate Degree	Management	70000	2234.99	2014-01-25
11	Gail	Erickson	Education	Professional	90000	4319.99	2006-10-02
12	Barry	Johnson	Education	Management	80000	4968.59	2014-05-15
13	Peter	Krebs	Graduate Degree	Clerical	50000	59.53	2013-01-14
14	Greg	Alderson	Partial High School	Clerical	45000	23.5	2013-07-05

I suggest you to refer following examples to understand the Stored procedures in detail:

- SELECT Stored Procedure in SQL Server to understand, How to write the SELECT Statements inside the Stored Procedure
- INSERT Stored Procedure in SQL Server to understand the steps involved in writing the INSERT Statements inside the Stored Procedure
- UPDATE Stored Procedure in SQL Server to write the UPDATE Statements inside the Stored Procedure
- Input Parameters in Stored Procedures to declare and use the Input Parameters and

- Output Parameters in Stored procedures to output the values
- Return Values in Stored Procedure

Uses of Stored Procedures in SQL Server

Following are benefits of using the SQL Server stored Procedures

- All the Stored Procedures are pre-compiled, and their execution plan is cached. So, when you execute the same Stored Procedures again, then it will used the cache.
- It will help you to encapsulate the business rules and policies. For example, database admin will create a procedure, and multiple users will access the procedure from JAVA, C#, C++, Python, R etc.
- Instead of sending hundreds of lines of SQL code, it is better to use stored procedure. So that we can call the single statement (stored procedure name), rather than writing the complex statement or sending it over the network.
- This helps you to access the database objects in more secured and uniform way
- Network bandwidth conservation (protect from harm or delay)

Before we get into the practical example, let us see the syntax of a stored procedure.

Syntax of the Stored Procedures in SQL Server

The basic syntax behind the SQL Server stored procedure is as shown below:

```
1   -- Create Stored procedures in SQL Server
2   CREATE[ORALTER]PROCEDURE[Schema_Name].Procedure_Name
3   @Parameter_NameData_type,
4   ….
5   @Parameter_NameData_type
6   AS
7   BEGIN
8   -- Procedure or SQL Query
9   -- SELECT, Insert, Update, Or Delete Statements
10  -- You can Use CTE also
11  END
```

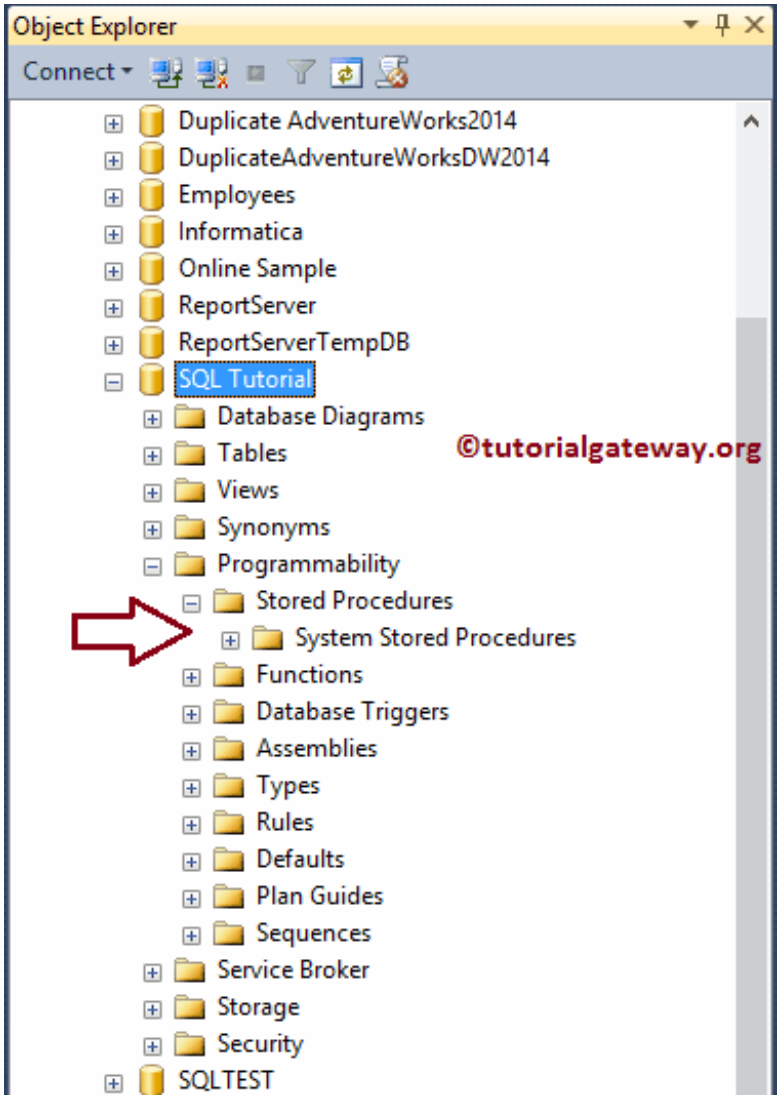
- **Schema_name:** Please specify the schema name. For example, dbo, or Humanresource etc.
- **Procedure_Name:** You can specify any name you wish to give other than the system reserved keywords. Please try to use meaningful names so that you can identify them easily.
- **@Parameter_Name:** Every stored procedure accepts zero or more parameters, it's completely depends upon the user requirements. While declaring the parameters don't forget the appropriate data type. For example (@FullName VARCHAR(50), @Age INT)

Create Stored Procedures in SQL Server

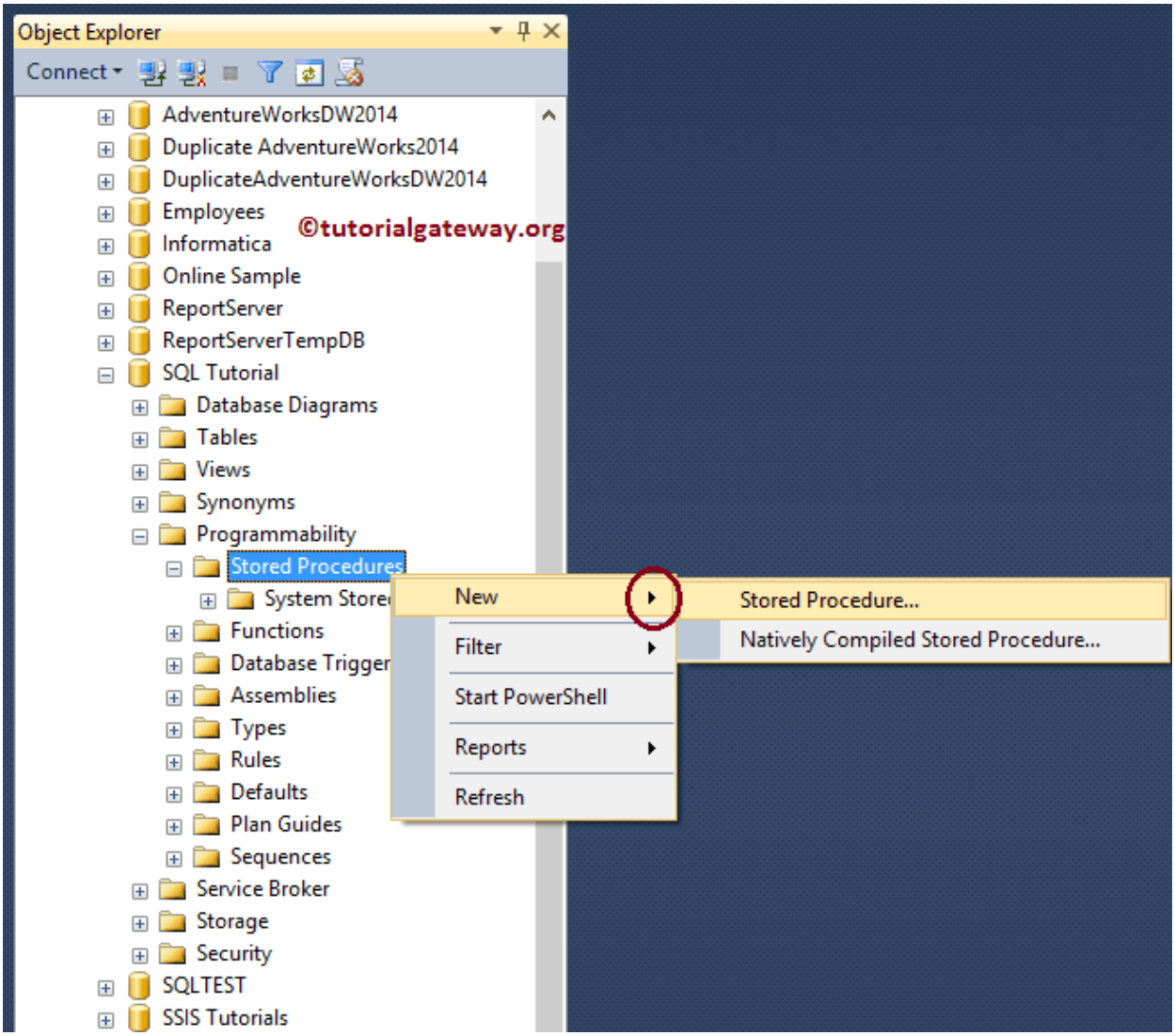
We can create the SQL stored procedure in two ways:

Create SQL Stored Procedures using SQL Server Management Studio

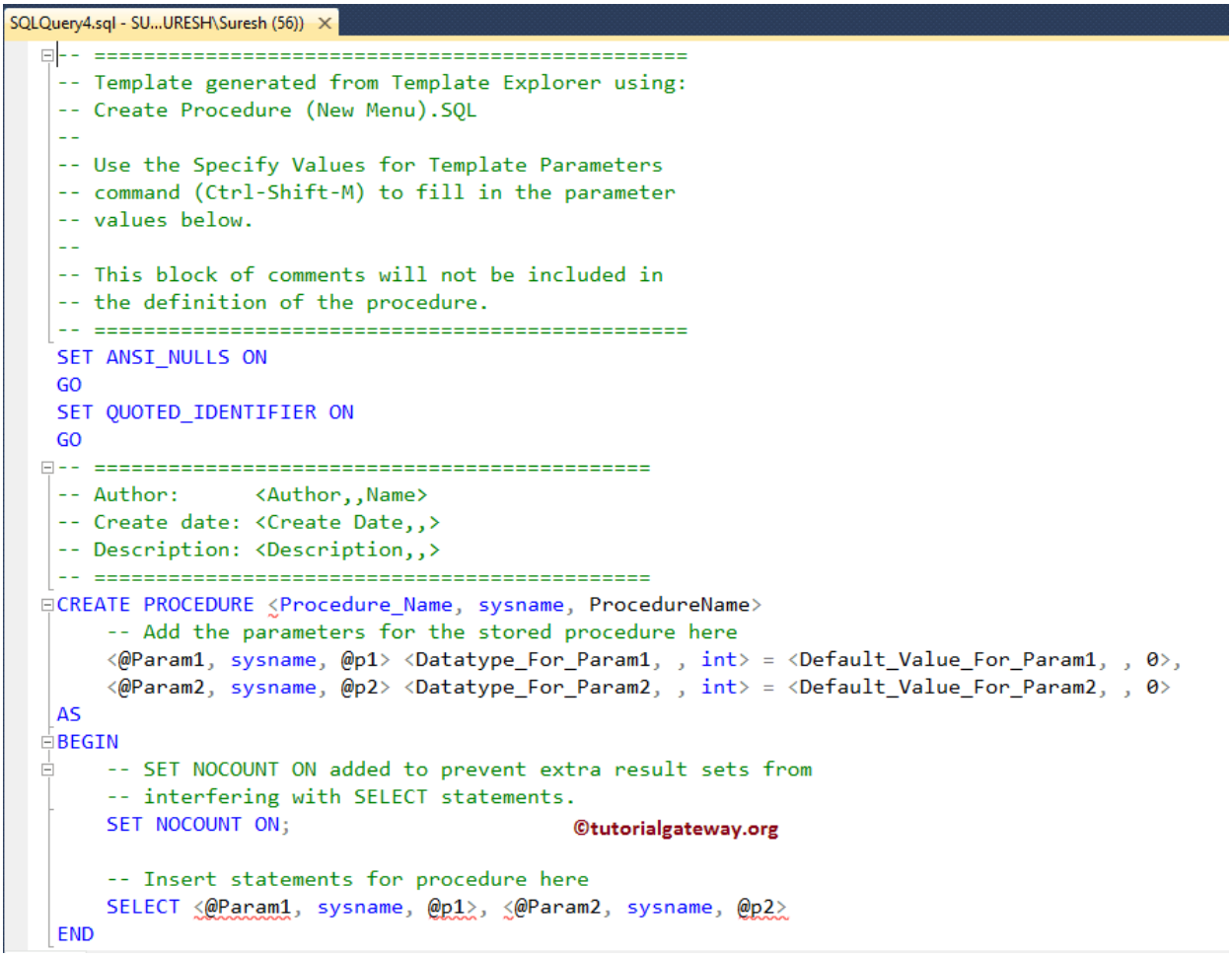
In this example we will show you the steps involved in create a SQL stored procedure using the SQL Server Management Studio. Before we start creating any procedure, let us see whether our database had any stored procedures. In order to view the existing Stored Procedures, Please select the Database -> Programmability -> Stored procedures. From the below screenshot you can observe that, our [SQL Tutorial] database has no procedures.



Right click on the Stored Procedures folder will open the context menu. Please select the **New -> Stored Procedure..** option from the context menu as shown below.



Once you click on the **New Stored Procedure…** option it will open the new Query window with default Stored Procedure Template as we shown below



Here, you can add the Procedure name, Parameters (if required), and the Procedure (or SQL Query) you want to use.

Create SQL Stored Procedures using Transact-SQL Query

In this example we will show you, How to create Stored Procedures in SQL Server using the Create Procedure Statement

SQL CODE

```
1   --Create SQL Stored Procedures
2   USE[SQLTutorial]
3   GO
4   CREATEPROCEDURESelectingAllRecordsInEmployee
5   AS
6   BEGIN
7   SETNOCOUNTON;
8   SELECT[FirstName]
9   ,[LastName]
10  ,[Education]
11  ,[Occupation]
12  ,[YearlyIncome]
13  ,[Sales]
14  ,[HireDate]
15  FROM[Employee]
16  END
17  GO
18
19
```

OUTPUT

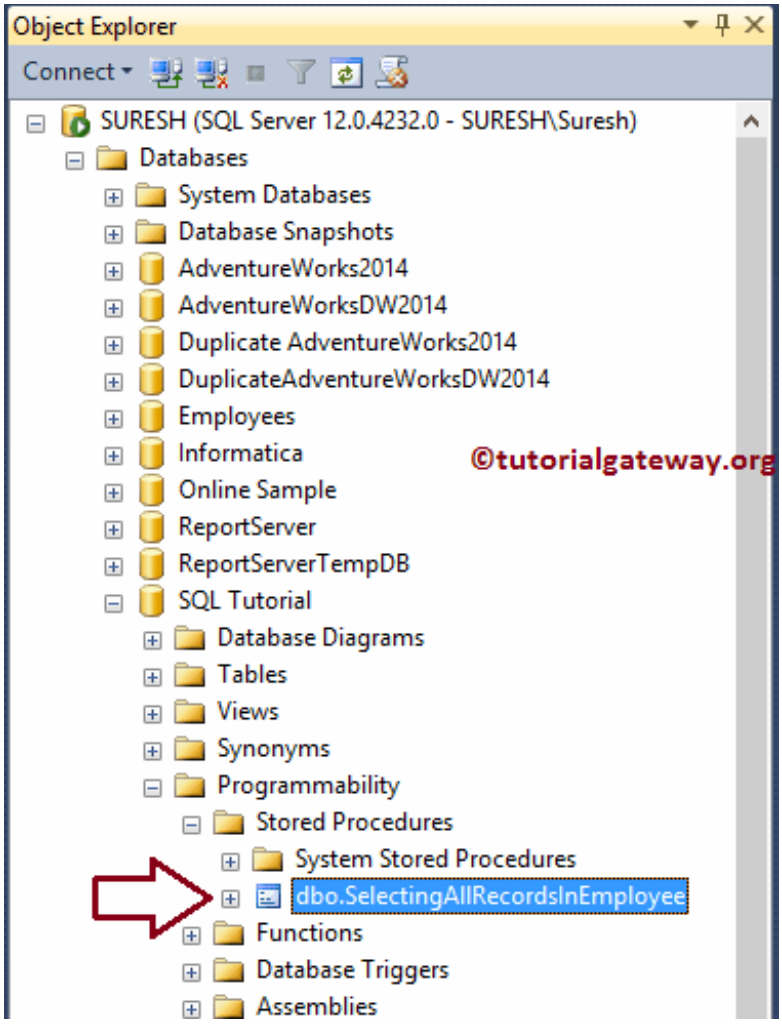
```
USE [SQL Tutorial]
GO

CREATE PROCEDURE SelectingAllRecordsInEmployee
AS
BEGIN

    SET NOCOUNT ON;
    SELECT [FirstName]
          ,[LastName]
          ,[Education]
          ,[Occupation]
          ,[YearlyIncome]
          ,[Sales]
          ,[HireDate]
    FROM [Employee]
END
GO
```

100 % ©tutorialgateway.org

Messages

```
Command(s) completed successfully.
```

As you can see that, the command has completed successfully. The following screenshot will show you the Stored Procedure that we created now.

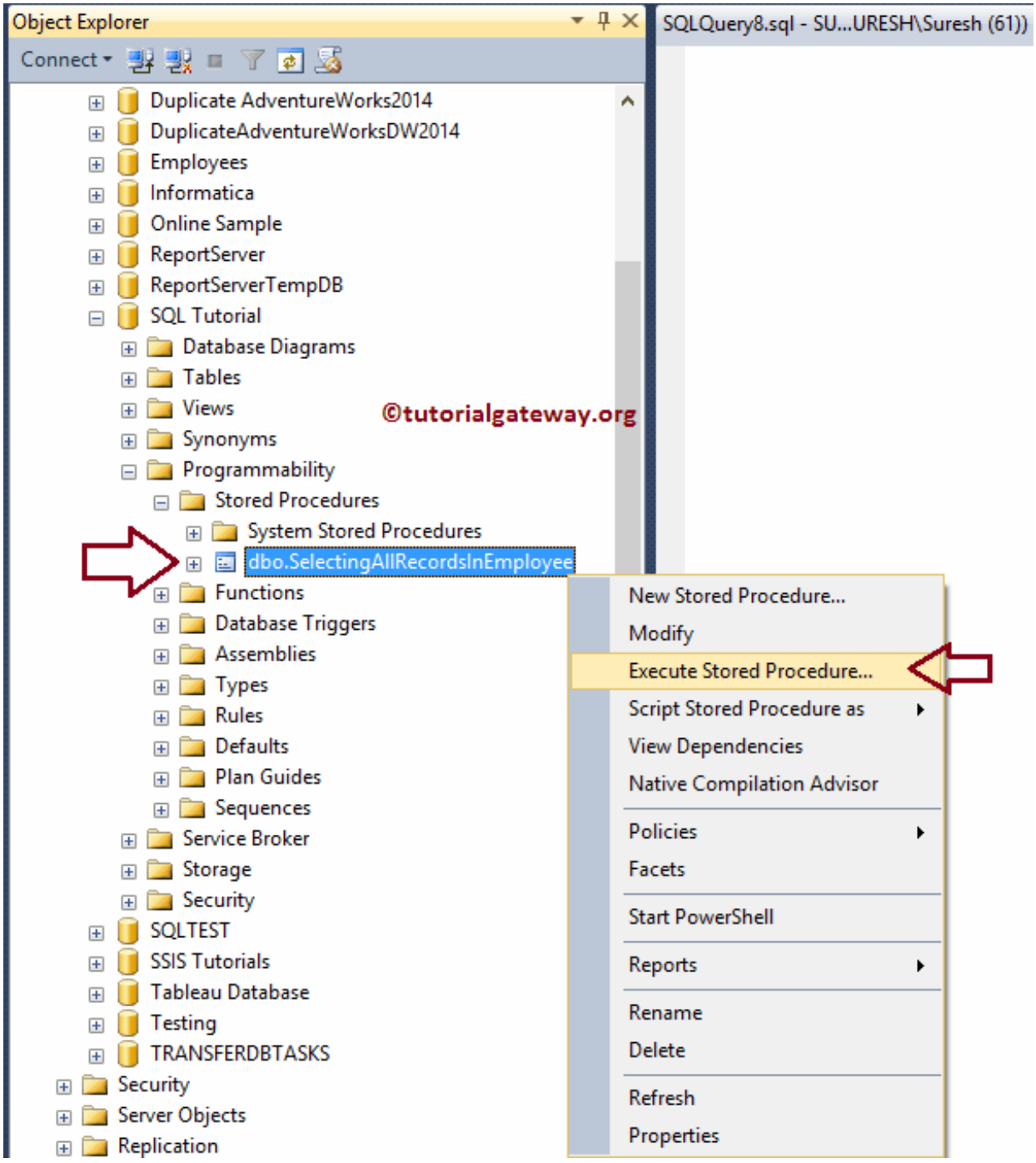


Execute Stored Procedures in SQL Server

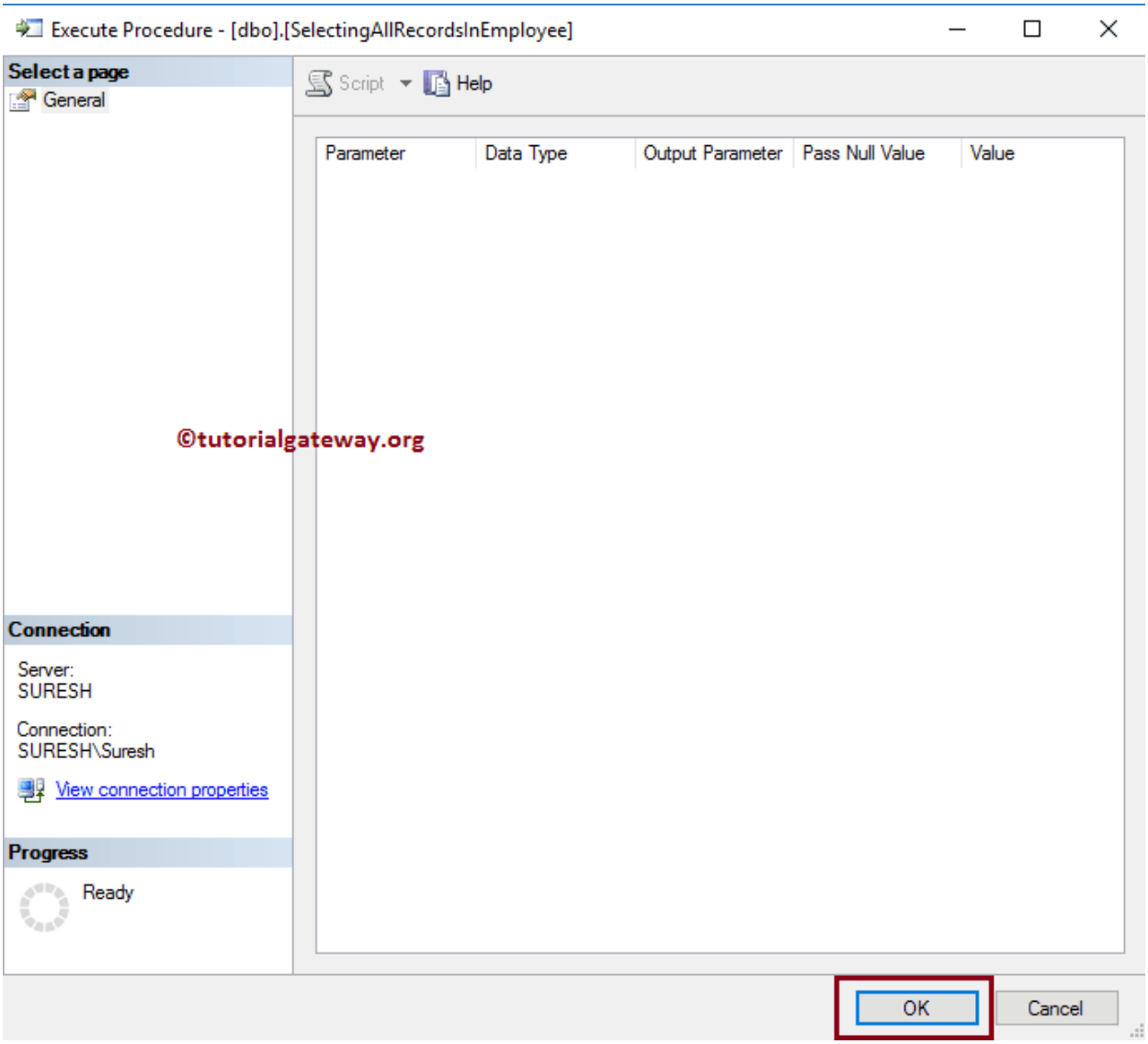
We can execute the SQL stored procedure in two ways:

Execute SQL Stored Procedures using SSMS

In this example we will show you, How to execute a Stored Procedure in SQL Server using the SQL Server Management Studio (SSMS). In order to excute, Please navigate to the Programmability -> Stored Procedures, and then select the Stored Procedure that you want to execute (SelectingAllRecordsInEmployee) and right-click on it will open the context menu. Here, select the **Execute Stored Procedure..** option as shown below



Once you select the **Execute Stored Procedure..** option, a Execute Procedure window will be opened as shown below. If the procedure has any parameters then we have to assign / pass those values, and then click OK to execute. Here, our stored procedure does not have any parameters so, click OK to execute the procedure



Once you click on the OK button, a new query window will be opened with the following auto generated code

SQL CODE

```
USE[SQLTutorial]
GO
DECLARE@return_valueint
EXEC@return_value=[dbo].[SelectingAllRecordsInEmployee]
SELECT'Return Value'=@return_value
GO
```

OUTPUT

```
USE [SQL Tutorial]
GO

DECLARE @return_value int

EXEC    @return_value = [dbo].[SelectingAllRecordsInEmployee]

SELECT  'Return Value' = @return_value

GO
```

100 % ©tutorialgateway.org

Results Messages

	FirstName	LastName	Education	Occupation	YearlyIncome	Sales	HireDate
1	John	Yang	Bachelors	Professional	90000	3578.27	2006-01-28
2	Rob	Johnson	Bachelors	Management	80000	3399.99	2010-12-29
3	Ruben	Torres	Partial College	Skilled Manual	50000	699.0982	2011-12-29
4	Christy	Zhu	Bachelors	Professional	80000	3078.27	2012-12-28
5	Rob	Huang	High School	Skilled Manual	60000	2319.99	2008-09-22
6	John	Ruiz	Bachelors	Professional	70000	539.99	2009-07-06
7	Tutorial	Gateway	Masters Degree	Management	80000	2320.49	2009-08-12
8	Christy	Mehta	Partial High School	Clerical	50000	24.99	2007-07-05
9	Rob	Verhoff	Partial High School	Clerical	45000	24.99	2013-09-15
10	Christy	Carlson	Graduate Degree	Management	70000	2234.99	2014-01-25
11	Gail	Erickson	Education	Professional	90000	4319.99	2006-10-02
12	Barry	Johnson	Education	Management	80000	4968.59	2014-05-15
13	Peter	Krebs	Graduate Degree	Clerical	50000	59.53	2013-01-14
14	Greg	Alderson	Partial High School	Clerical	45000	23.5	2013-07-05

	Return Value
1	0

Execute SQL Stored Procedures using T-SQL

This example will show you, How to execute a Stored Procedure in SQL Server using the EXEC Command (Execute Command)

SQL CODE

```
1  USE[SQLTutorial]
2  GO
3  EXEC[dbo].[SelectingAllRecordsInEmployee]
4
```

OUTPUT

```
USE [SQL Tutorial]
GO

EXEC  [dbo].[SelectingAllRecordsInEmployee]
```

100 % ©tutorialgateway.org

Results Messages

	FirstName	LastName	Education	Occupation	YearlyIncome	Sales	HireDate
1	John	Yang	Bachelors	Professional	90000	3578.27	2006-01-28
2	Rob	Johnson	Bachelors	Management	80000	3399.99	2010-12-29
3	Ruben	Torres	Partial College	Skilled Manual	50000	699.0982	2011-12-29
4	Christy	Zhu	Bachelors	Professional	80000	3078.27	2012-12-28
5	Rob	Huang	High School	Skilled Manual	60000	2319.99	2008-09-22
6	John	Ruiz	Bachelors	Professional	70000	539.99	2009-07-06
7	Tutorial	Gateway	Masters Degree	Management	80000	2320.49	2009-08-12
8	Christy	Mehta	Partial High School	Clerical	50000	24.99	2007-07-05
9	Rob	Verhoff	Partial High School	Clerical	45000	24.99	2013-09-15
10	Christy	Carlson	Graduate Degree	Management	70000	2234.99	2014-01-25
11	Gail	Erickson	Education	Professional	90000	4319.99	2006-10-02
12	Barry	Johnson	Education	Management	80000	4968.59	2014-05-15
13	Peter	Krebs	Graduate Degree	Clerical	50000	59.53	2013-01-14
14	Greg	Alderson	Partial High School	Clerical	45000	23.5	2013-07-05

Rename Stored Procedure in SQL Server

There is no such thing called renaming the Stored procedure. This is because, renaming a stored procedure will not change its corresponding object name within the **sys.sql_modules.** So, if you want to rename any existing stored procedure, then DROP and recreate it with new name.

SQL CODE

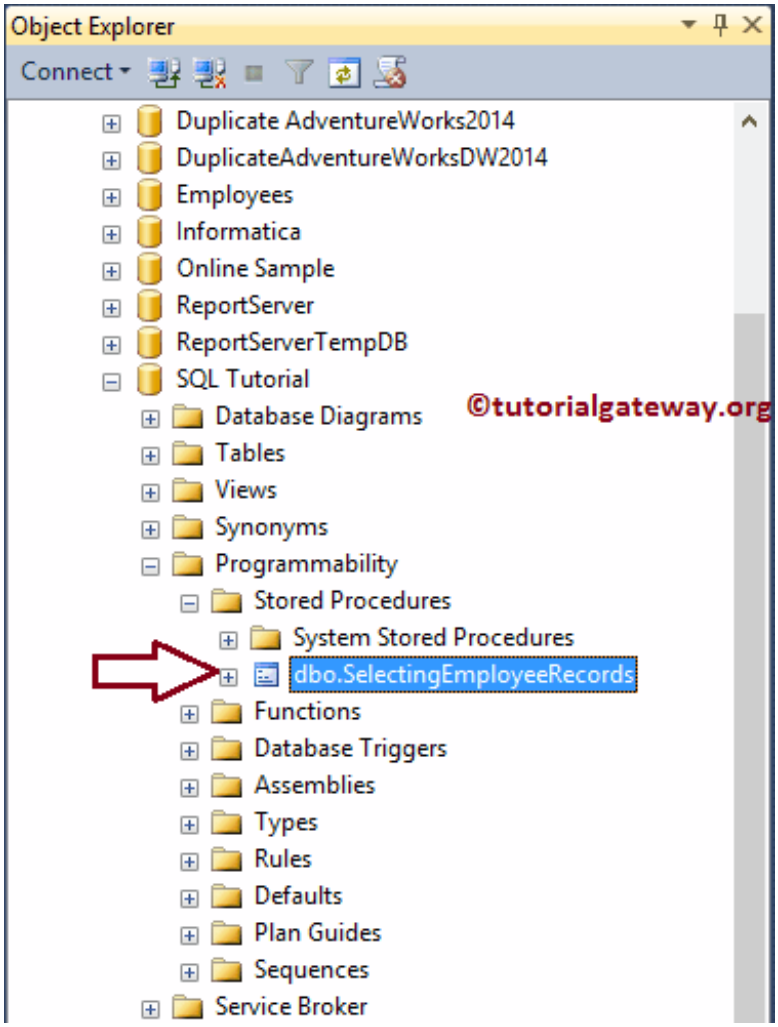
```
1   -- Renaming SQL Stored Procedures
2   USE[SQLTutorial]
3   GO
4   -- Check whether there is a Stored Procedure with the Following Name
5   IFOBJECT_ID ('SelectingAllRecordsInEmployee','P')ISNOTNULL
6   -- If so, Drop that Procedure
7   DROPPROCEDURESelectingAllRecordsInEmployee;
8   GO
9   -- Creating  Stored Procedure with New name
10  CREATEPROCEDURESelectingEmployeeRecords
11  AS
12  BEGIN
13  SETNOCOUNTON;
14  SELECT[FirstName]
15  ,[LastName]
16  ,[Education]
17  ,[Occupation]
18  ,[YearlyIncome]
19  ,[Sales]
20  ,[HireDate]
21  FROM[Employee]
22  END
23  GO
24
25
26
```

OUTPUT

```
USE [SQL Tutorial]
GO

IF OBJECT_ID ( 'SelectingAllRecordsInEmployee', 'P' ) IS NOT NULL
    DROP PROCEDURE SelectingAllRecordsInEmployee;
GO

CREATE PROCEDURE SelectingEmployeeRecords
AS
BEGIN

    SET NOCOUNT ON;
    SELECT [FirstName]
          ,[LastName]
          ,[Education]
          ,[Occupation]
          ,[YearlyIncome]
          ,[Sales]
          ,[HireDate]
    FROM [Employee]
END
GO
```

```
100 %
©tutorialgateway.org
Messages
Command(s) completed successfully.
```

From the below screenshot you can see that, we successfully deleted the old stored procedure, and created the new one.

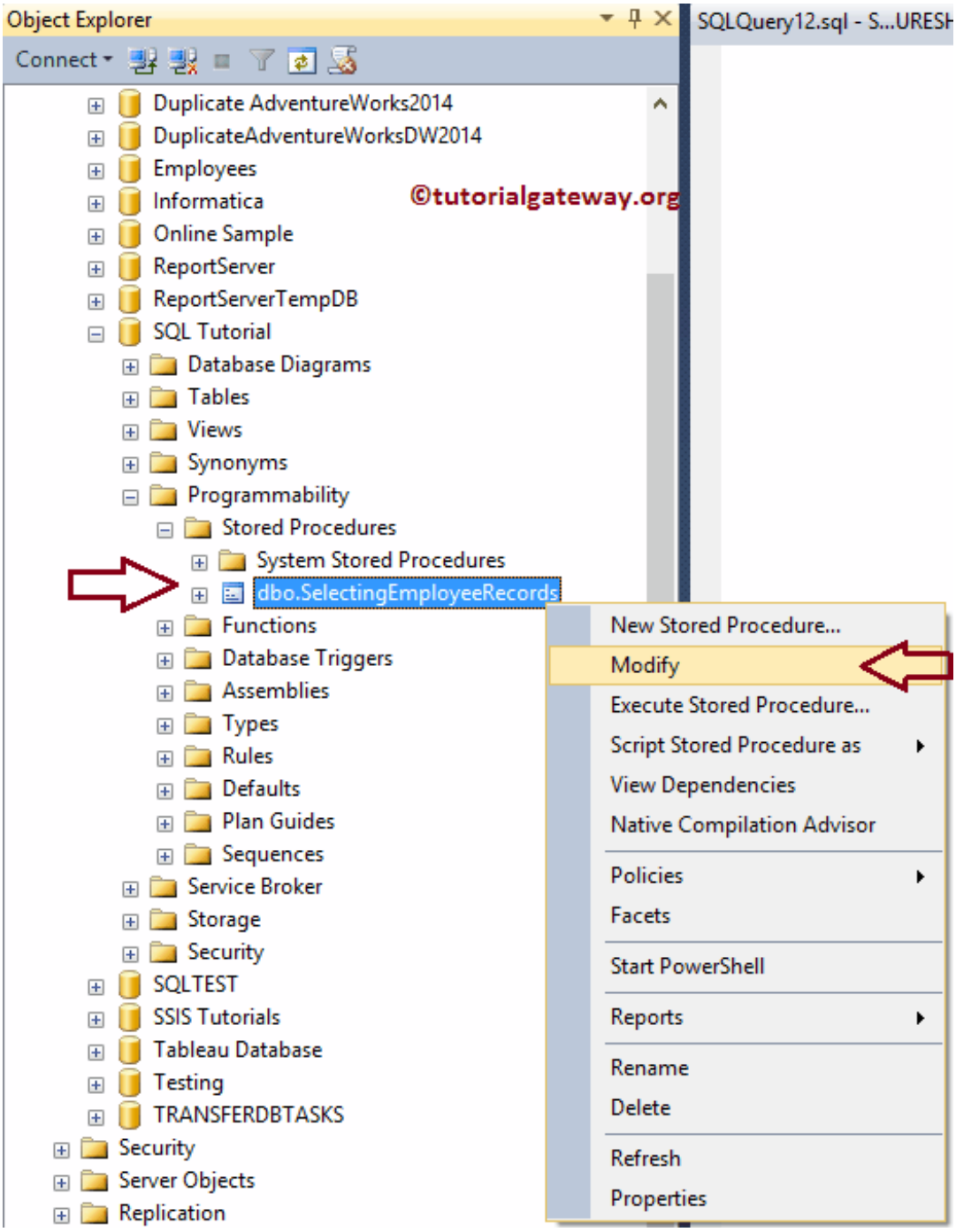


Modify Stored Procedures in SQL Server

The following examples will help you understand the steps involved in modifying the SQL Stored Procedures using the Microsoft SQL Server Management Studio (SSMS), and Transact-SQL Query.

Modify Stored procedures in SQL Server using SSMS

In order to modify the SQL Stored procedures using the SQL Server Management Studio, Please navigate to the stored procedure that you want to modify (SelectingEmployeeRecords), and right-click on it will open the context menu. Here, select the Modify option as shown below



Once you select the Modify option, a new query window will be opened with auto generated ALTER PROCEDURE code. You can modify as per your requirement.

```
USE [SQL Tutorial]
GO
/****** Object:  StoredProcedure [dbo].[SelectingEmployeeRecords] ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

ALTER PROCEDURE [dbo].[SelectingEmployeeRecords]
AS
BEGIN

    SET NOCOUNT ON;
    SELECT [FirstName]
          ,[LastName]
          ,[Education]
          ,[Occupation]
          ,[YearlyIncome]
          ,[Sales]
          ,[HireDate]
    FROM [Employee]
END
```

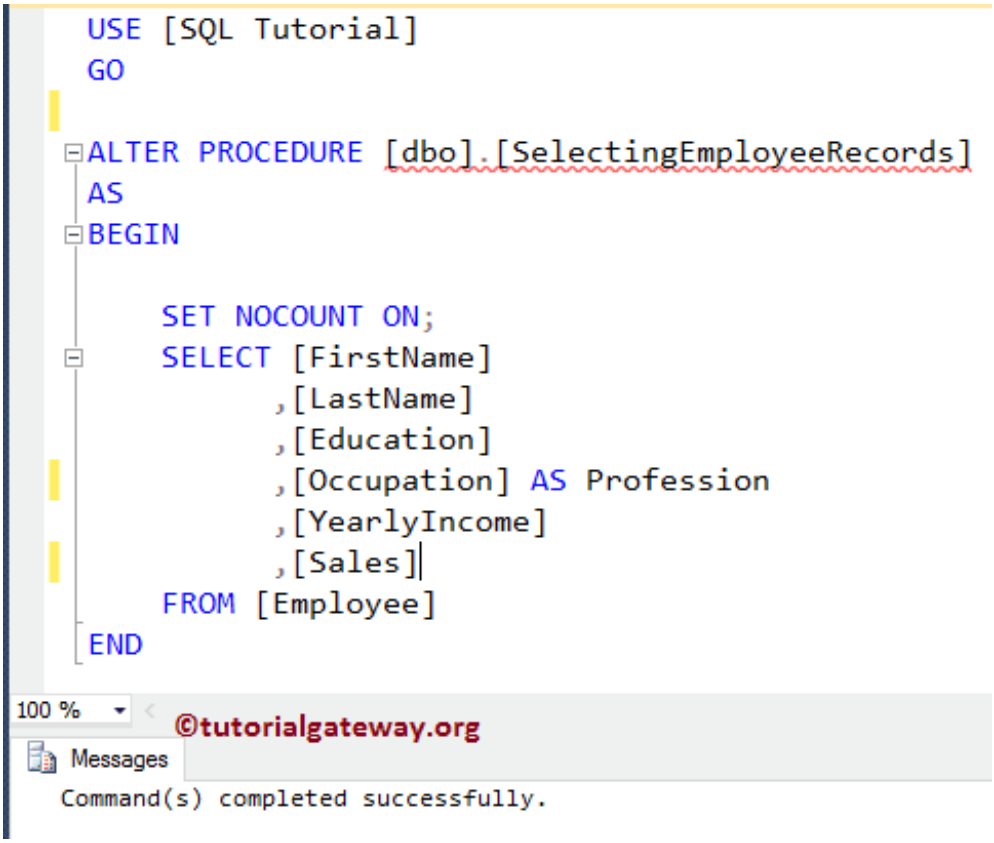

Use T-SQL Query to Modify Stored Procedures in SQL Server

This example will show you, the steps involved in modifying the existing stored procedure using ALTER PROCEDURE statement. To do so, Click New query and return the following query.

SQL CODE

```
1   USE[SQLTutorial]
2   GO
3   ALTERPROCEDURE[dbo].[SelectingEmployeeRecords]
4   AS
5   BEGIN
6   SETNOCOUNTON;
7   SELECT[FirstName]
8   ,[LastName]
9   ,[Education]
10  ,[Occupation]ASProfession
11  ,[YearlyIncome]
12  ,[Sales]
13  FROM[Employee]
14  END
15
```

OUTPUT



Let us execute the procedure to check whether we successfully modifies the procedure or not.

SQL CODE

```
1  USE[SQLTutorial]
2  GO
3  EXEC[dbo].[SelectingEmployeeRecords]
4  GO
5
```

OUTPUT

```
USE [SQL Tutorial]
GO

EXEC [dbo].[SelectingEmployeeRecords]
GO
```

100 %

Results | Messages

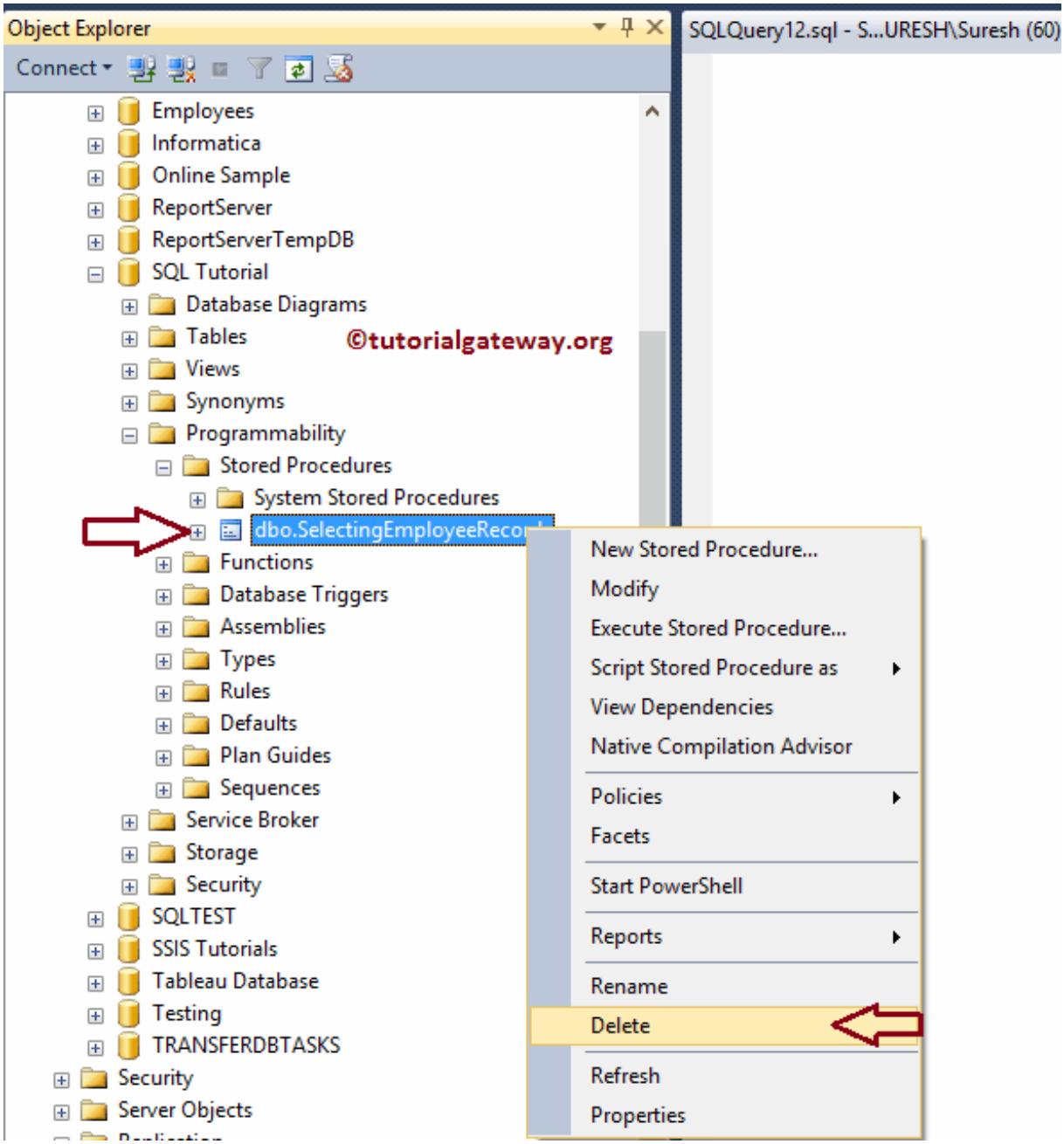
	FirstName	LastName	Education	Profession	YearlyIncome	Sales
1	John	Yang	Bachelors	Professional	90000	3578.27
2	Rob	Johnson	Bachelors	Management	80000	3399.99
3	Ruben	Torres	Partial College	Skilled Manual	50000	699.0982
4	Christy	Zhu	Bachelors	Professional	80000	3078.27
5	Rob	Huang	High School	Skilled Manual	60000	2319.99
6	John	Ruiz	Bachelors	Professional	70000	539.99
7	Tutorial	Gateway	Masters Degree	Management	80000	2320.49
8	Christy	Mehta	Partial High School	Clerical	50000	24.99
9	Rob	Verhoff	Partial High School	Clerical	45000	24.99
10	Christy	Carlson	Graduate Degree	Management	70000	2234.99
11	Gail	Erickson	Education	Professional	90000	4319.99
12	Barry	Johnson	Education	Management	80000	4968.59
13	Peter	Krebs	Graduate Degree	Clerical	50000	59.53
14	Greg	Alderson	Partial High School	Clerical	45000	23.5

Delete Stored Procedures in SQL Server

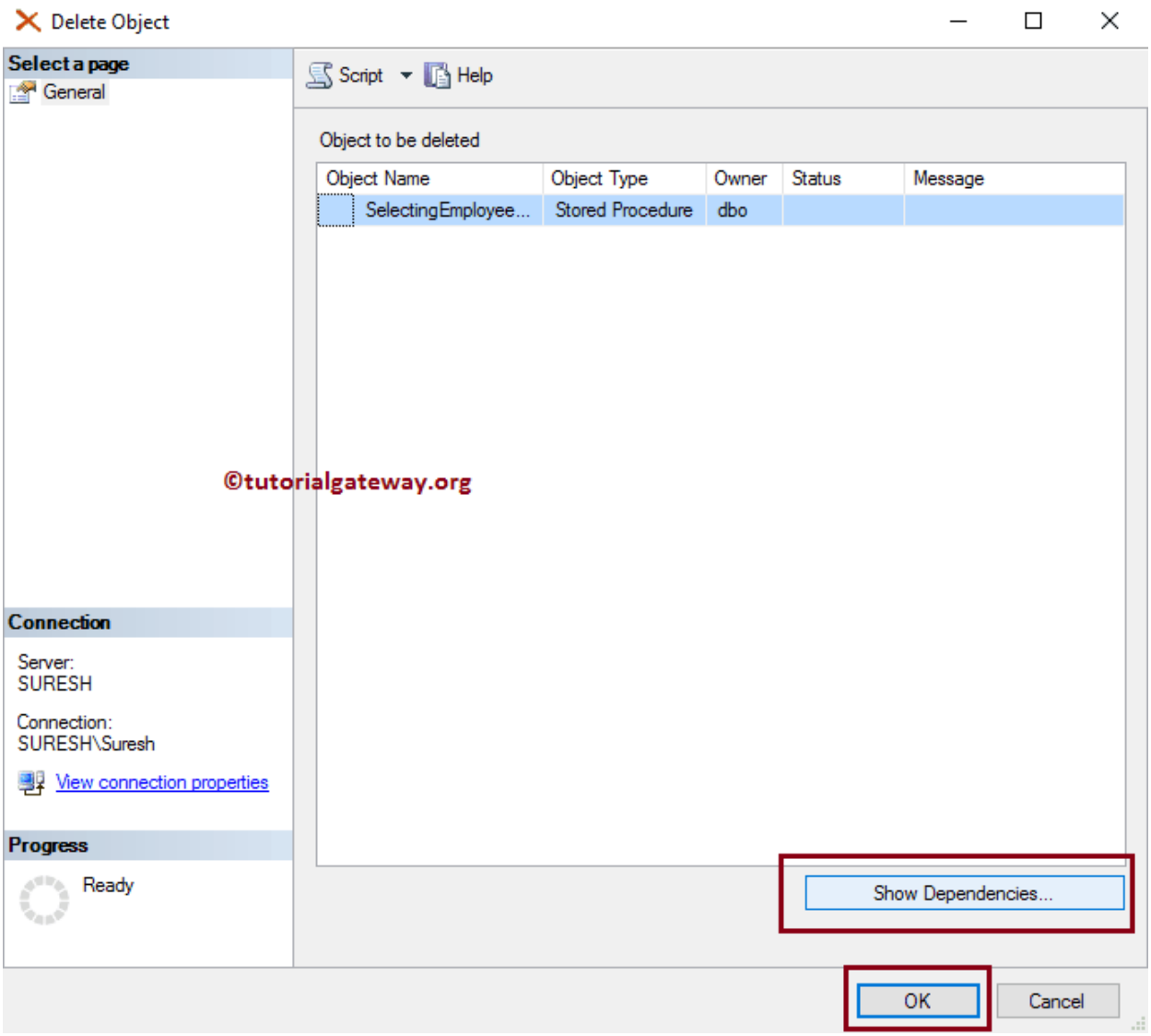
The following examples will show you, How to delete or drop the SQL Stored Procedures using the Transact-SQL Query, and Microsoft SQL Server Management Studio (SSMS).

Delete SQL Stored Procedures using SQL Server Management Studio

In this example we will show you, How to delete a SQL Stored Procedures using the SSMS (SQL Server Management Studio). In order to do so, Please navigate to the Stored Procedure that you want to delete, and right click on the procedure name to open the context menu. You can click on the Delete option as shown below



For this demonstration we are going to delete SelectingEmployeerecords. Once you select the delete option, a Delete Object window will be opened as shown below. Click on the ***show Dependencies*** button to check the dependencies, and then click OK to delete the stored procedure.



Delete SQL Stored procedure using SQL Query

In this example we will show you, How to delete the stored procedures in SQL server using the T-SQL query.

SQL CODE

```
1  USE[SQLTutorial]
2  GO
3  IFOBJECT_ID ('SelectingEmployeeRecords','P')ISNOTNULL
4  DROPPROCEDURESelectingEmployeeRecords;
5  GO
6
```

TIP: It is good practice to check, whether the stored procedure exists in database or not using IF OBJECT_ID (N'Stored_Procedure_Name', 'P') IS NOT NULL

Stored Procedures Best Practices

Following are the list of suggestions that might help you to improve the stored procedures performance.

- Try to use the Schema Names while you creating, or referencing any database object. It will decrease the database engine processing time.

- Always specify the required column names within the SELECT Statement. And avoid the SELECT * Statement
- While Creating, or altering a table using (CREATE TABLE or ALTER TABLE), use the DEFAULT Keyword to assign the default values to the Columns. This will prevent the NULL values, and assign those default values to the column data.
- When you are creating Temporary Tables inside the stored procedure, You have to explicitly specify whether the Column accepts NULLS, or NOT. It can done by using NULL, or NOT NULL
- Rather than Extracting, or Inserting large amount of data, try to work with less, and essential data. This will reduce the query processing load, and increase the query performance.
- Use the SET NOCOUNT ON statement within the stored procedure. This will turn off the messages that are sent by the SQL Server to the Client. It includes number of rows updated, deleted etc.
- Try to replace the UNION Operator, or OR Operator with the UNION ALL Operator, unless you are looking for distinct values.
- If possible, avoid using the SCALAR Functions in the SELECT Statement that returns large amount of data. This is because, scalar function is applied on each row (row basic), so it will the query performance.
- To handle the errors, SQL allows us to use the TRY CATCH feature inside the Stored Procedures. So, try to use the TRY CATCH feature
- Always use the BEGIN..COMMIT TRANSACTION within the Stored Procedure. Remember that, the transaction should be as short as possible, otherwise there is a danger in either deadlock or longer locking.

Best Approach to create Stored Procedures in SQL Server

This will be ideal way to create a stored procedure in sql server

SQL CODE

```
1  USE[SQLTutorial]
2  GO
3  IFOBJECT_ID ('SelectingEmployeeRecords','P')ISNOTNULL
4  DROPPROCEDURESelectingEmployeeRecords;
5  GO
6  CREATEPROCEDURESelectingEmployeeRecords
7  AS
8  BEGIN
9  SETNOCOUNTON;
10 SELECT[FirstName]
11 ,[LastName]
12 ,[Education]
13 ,[Occupation]
14 ,[YearlyIncome]
15 ,[Sales]
16 ,[HireDate]
17 FROM[Employee]
18 END
19 GO
20
21
```