Stored Procedure Lab Manual + Northwind



Session: 2022 – 2026

**Submitted by:**

Saad Ahmad Malik 2022-CS-1

**Supervised by:**

Mr. Nazeef ul haq

Department of Computer Science

**University of Engineering and Technology**

**Lahore Pakistan**

# SQL stored procedure

A SQL stored procedure (SP) is a collection of SQL statements and SQL command logic, which is compiled and stored on the database. Stored procedures in SQL allow us to create SQL queries to be stored and executed on the server. Stored procedures can also be cached and reused. The main purpose of stored procedures is to hide direct SQL queries from the code and improve performance of database operations such as select, update, and delete data.

# • Types of stored procedures

There are two types of stored procedures available in SQL Server:

1. User defined stored procedures.
2. System stored procedures.

## User defined stored procedures.

User defined stored procedures are created by database developers or database administrators. These SPs contains one more SQL statements to select, update, or delete records from database tables. User defined stored procedure can take input parameters and return output parameters. User defined stored procedure is mixture of DDL (Data Definition Language) and DML (Data Manipulation Language) commands. User defined SPs are further classified into two types:

**T-SQL stored procedures:** T-SQL (Transact SQL) SPs receive and return parameters. These SPs process the Insert, Update and Delete queries with or without parameters and return data of rows as output. This is one of the most common ways to write SPs in SQL Server.

**CLR stored procedures**: CLR (Common Language Runtime) SPs are written in a CLR based programming language such as C# or VB.NET and are executed by the .NET Framework.

System stored procedures.

System stored procedures are created and executed by SQL Server for the server administrative activities. Developers usually don't interfere with system SPs.

My database name is Northwind.

**How to create a SELECT stored procedure?**

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Saad Ahmad

-- Create date: 23 Mar 2024

-- Description: Return all members

-- =============================================

--Store procedure name is --> stpGetCategories

CREATE PROCEDURE stpGetCategories

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

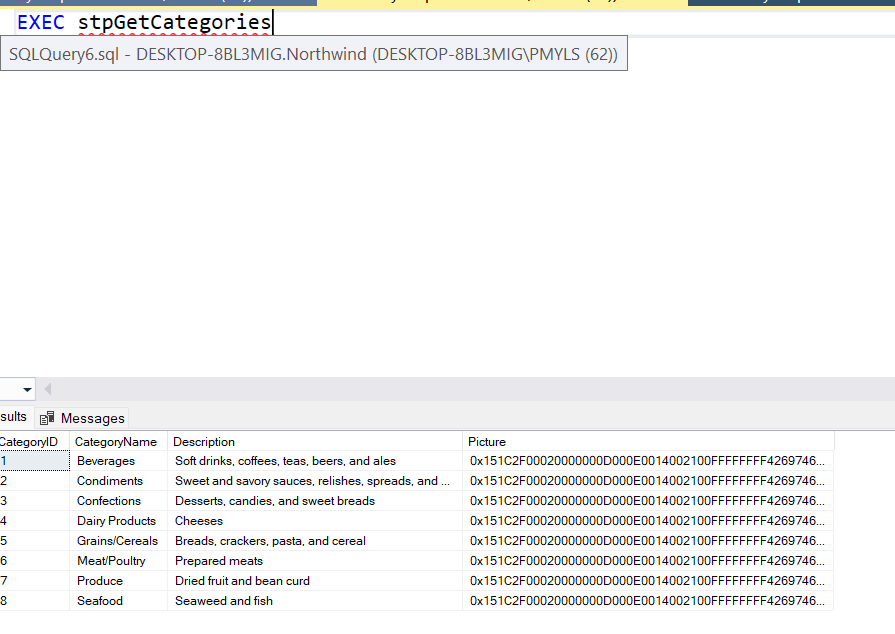
-- Select statements for procedure here

Select \* from Categories

END

GO

## Output:

****

**What are parameters in stored procedures?**

Parameters in SPs are used to pass input values and return output values. There are two types of parameters:

1. Input parameters - Pass values to a stored procedure.

2. Output parameters - Return values from a stored procedure.

Here is the updated SP with a parameter @CategoryName.

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Saad Ahmad

-- Create date: 25 March 2024

-- Description: Return specifc CategoryName records

-- =============================================

--Store procedure name is --> stpGetAllMembers

CREATE PROCEDURE stpGetCategoriesByName

-- Add the parameters for the stored procedure here

@CategoryName nvarchar(30)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

-- Select statements for procedure here

Select \* from Categories

WHERE CategoryName like '%' + @CategoryName + '%'

END

GO

## Output



The code to execute looks like the following:

USE [Northwind]

GO

DECLARE @return\_value int

EXEC @return\_value = [dbo].[stpGetCategoriesByName]

@CategoryName = N'Beverages'

SELECT 'Return Value' = @return\_value

GO

**How to create a INSERT query based stored procedure?**

We can use an INSERT INTO SQL query to insert data into a table. The following SQL statement creates an INSERT SP with three parameters.

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Saad Ahmad

-- Create date: 25 March 2024

-- Description: Insertiing Records

-- =============================================

--Store procedure name is --> stpGetAllMembers

CREATE PROCEDURE stpInsertMember

@CategoryId varchar(50),

@CategoryName varchar(25),

@Description varchar(50),

@Picture varchar(50)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

Insert into Categories(CategoryId,CategoryName,Description,Picture)

Values (@CategoryId,@CategoryName,@Description, @Picture)

END

GO

## Execution:

USE [Northwind]

GO

DECLARE @return\_value int

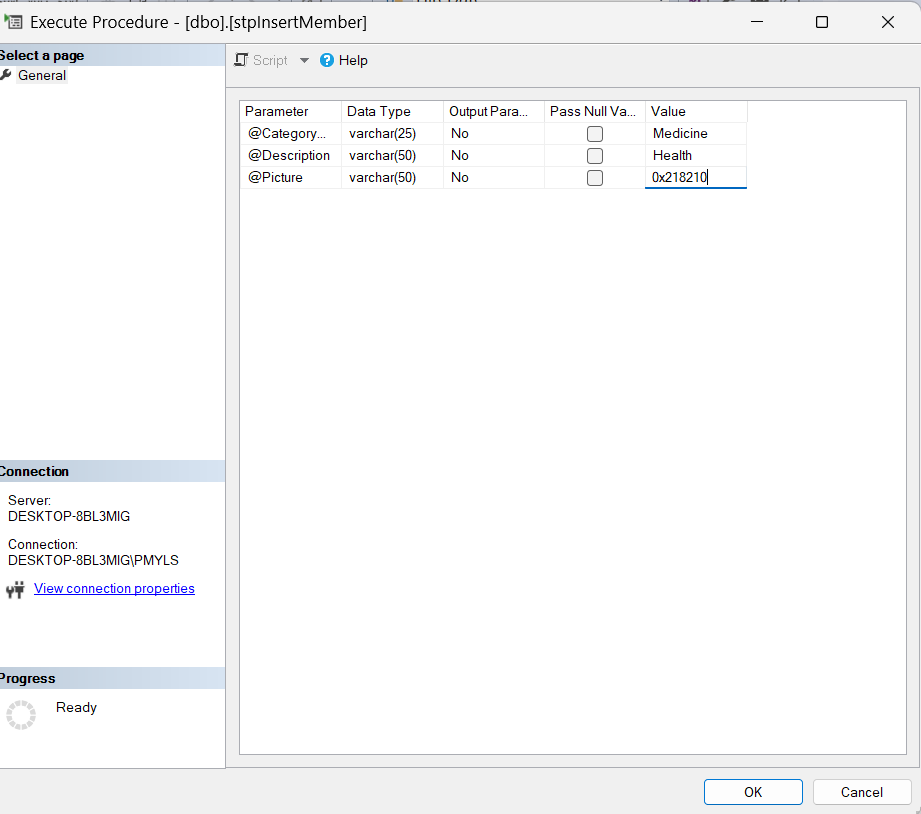
EXEC @return\_value = [dbo].[stpInsertMember]

@CategoryName = N'Medicine',

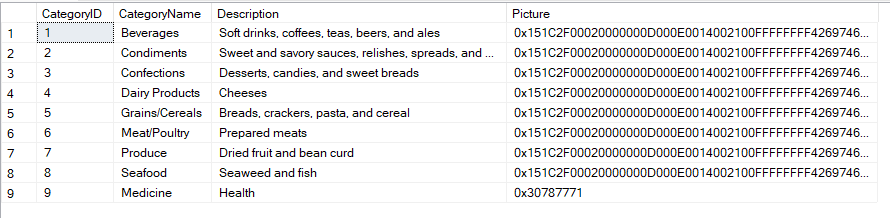
@Description = N'Health',

@Picture = N'0xwq'

SELECT 'Return Value' = @return\_value GO



Output



The 9th record is the newly added record.

**How to create an UPDATE quert based stored procedure?**

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Saad Ahmad

-- Create date: 25 March 2024

-- Description: Updating Records

-- =============================================

--Store procedure name is --> stpUpdateData

CREATE PROCEDURE stpUpdateData

@CategoryId varchar(50),

@CategoryName varchar(25),

@Description varchar(50),

@Picture varchar(50)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

Update Categories

SET CategoryName = @CategoryName,

Description = @Description,

Picture = @Picture

WHERE CategoryId = @CategoryId

END

GO

To Execute

USE [Northwind]

GO

DECLARE @return\_value int

EXEC @return\_value = [dbo].[stpUpdateData]

@CategoryId = N'9',

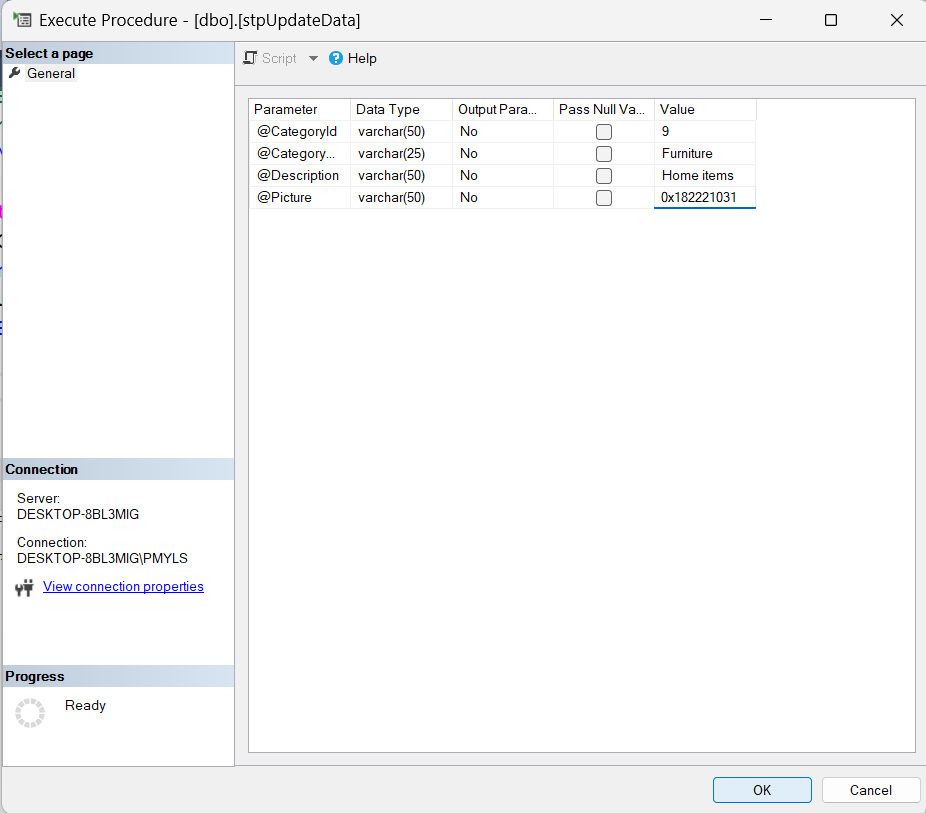
@CategoryName = N'Furniture',

@Description = N'Home items',

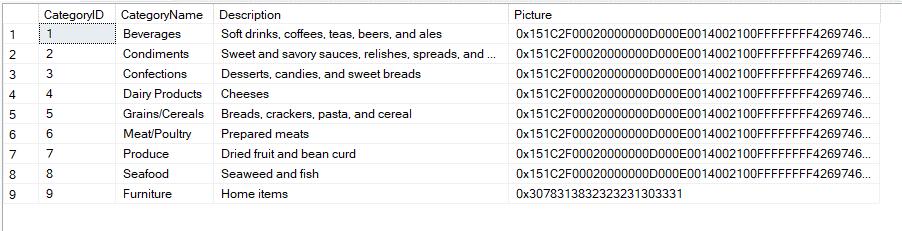
@Picture = N'0x182221031'

SELECT 'Return Value' = @return\_value

GO

Execution:  


## Output



Id 9 is updated

**How to create a DELETE query based stored procedure?**

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- Author: Saad Ahmad

-- Create date: 25 March 2024

-- Description: Deleting Records

-- =============================================

--Store procedure name is --> stpDeleteData

CREATE PROCEDURE stpDeleteData

@CategoryId varchar(50)

AS

BEGIN

-- SET NOCOUNT ON added to prevent extra result sets from

-- interfering with SELECT statements.

SET NOCOUNT ON;

DELETE FROM Categories

WHERE CategoryId = @CategoryId

END

GO

## To Execute

USE [Northwind]

GO

DECLARE @return\_value int

EXEC @return\_value = [dbo].[stpDeleteData]

@CategoryId = N'9'

SELECT 'Return Value' = @return\_value

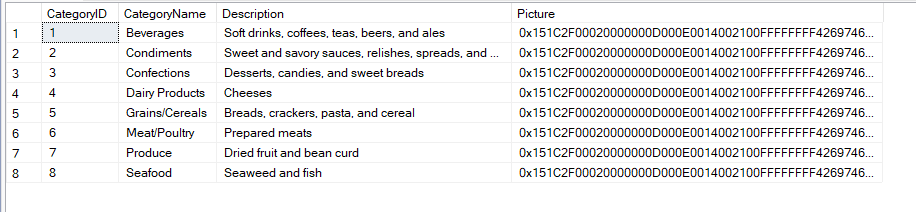
GO

Execution

A screenshot of a computer

Description automatically generated

## Output



Id 9 has been Deleted