

# Using Stored Procedures in SQL Server

The following document contains some test results on the execution of Stored Procedures in SQL Server. Below are listed the specs on the computer used for testing:

- FactoryStudio version *fs-9.1.9*.
- Microsoft SQL Server Management Studio 18 (SSMS).
- Stored Procedure described in the next sections.

The next sections will describe how to create a Stored Procedure that reproduced the error in question and the results of the same procedure with the new version of the product.

## Reported Issues

The reported error was that a Stored Procedure worked fine when executed in SQL Server Management Studio 18 but failed in FactoryStudio.

The error message returned from the SP call is detailed below:

```
DataSet Error: Column "Spec_id" is constrained to be unique. Value "34" is already present. [Code: 4]
```

## Test Procedure

### Creating a Stored Procedure

The Stored Procedures used in this test can be created by following the description below:

Creating DB.

```
USE master;  
CREATE DATABASE test; GO
```

Creating Tables with Primary Keys (*Spec id* and *Char id*).

```
USE test GO  
CREATE TABLE dbo.Characteristic (Char_Id int PRIMARY KEY NOT NULL,  
Char_Desc varchar(25) NOT NULL, OptionsChar varchar(25))  
GO  
INSERT dbo.Characteristic (Char_Id, Char_Desc, OptionsChar) VALUES(10,'C010','Char1'),(15,'C015','Char2'),(19,'  
C019','Char3'),(14,'C014','Char4'),(18,'C018','Char5'),(17,'C017','Char6'),(16,'C016','Char7')GO  
CREATE TABLE dbo.Specification (Spec_ID int PRIMARY KEY NOT NULL,  
Spec_Desc varchar(25) NOT NULL, OptionsSpec varchar(25))  
GO  
INSERT dbo.Specification (Spec_ID, Spec_Desc, OptionsSpec) VALUES  
(30, 'Spec 30', 'Spec1'),  
(32, 'Spec 32', 'Spec2'),  
(33, 'Spec 33', 'Spec3'),  
(37, 'Spec 37', 'Spec4'),  
(34, 'Spec 34', 'Spec5'),  
(39, 'Spec 39', 'Spec6'),  
(35, 'Spec 35', 'Spec7') GO
```

Creating a third table that contains the PrimaryKeys from the other two.

```

CREATE TABLE dbo.Active_Spec (AS_ID int PRIMARY KEY NOT NULL,
Char_ID int NOT NULL, Spec_ID int NOT NULL, OptionsAS varchar(25))
GO
INSERT dbo.Active_Spec(AS_ID, Spec_ID, Char_ID, OptionsAS) VALUES(1,30,10,'AS1'),(2,37,19,'AS2'),(3,34,19,'
AS3'),(7,35,16,'AS7'),(4,34,19,'AS4'),(6,39,18,'AS6'),(5,32,19,'AS5')GO
USE [Test] GO
SET ANSI_NULLS ON GO
SET QUOTED_IDENTIFIER ON GO
CREATE PROCEDURE [dbo].[TestSP]
@Char_Desc varchar(50)
AS BEGIN
SET NOCOUNT ON;

Declare @Char_Id int, @Spec_Id int

Select @Char_Id = Char_Id from Characteristic where Char_Desc = @Char_Desc

Select c.char_Desc, s.Spec_Id, s.Spec_Desc, c.OptionsChar, s.OptionsSpec, asp.OptionsAS
From Specification s
Join Active_Spec asp on asp.Spec_Id = s.Spec_Id Join Characteristic c on c.Char_Id = asp.Char_Id
Where c.Char_Id = @Char_Id GROUP BY c.Char_Desc,
s.Spec_Id, s.Spec_Desc, c.OptionsChar, s.OptionsSpec, asp.OptionsAS
END

```

Executing Procedure in SSMS.

```
EXEC TestSP 'C019'
```

EXEC TestSP 'C019'

100 %

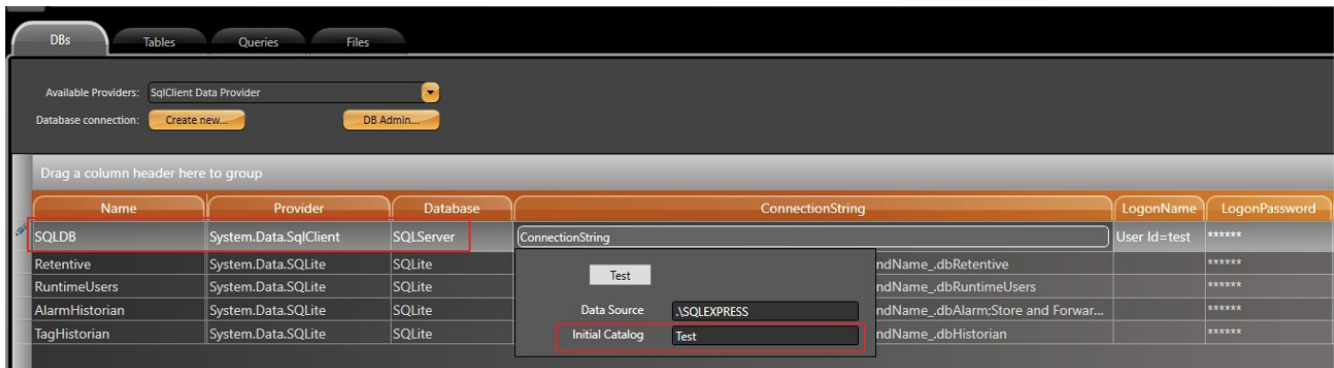
Results Messages

	char_Desc	Spec_Id	Spec_Desc	OptionsChar	OptionsSpec	OptionsAS
1	C019	32	Spec 32	Char3	Spec2	AS5
2	C019	34	Spec 34	Char3	Spec5	AS3
3	C019	34	Spec 34	Char3	Spec5	AS4
4	C019	37	Spec 37	Char3	Spec4	AS2

## Project Configuration

In your Engineering Environment, go to the **Datasets > DBs** page, and create a new DB for the *SQL Server* provider. The user **must** have administrator privileges in order to input the logon name and password.

In this test, the Stored Procedure was created in the Test catalog.



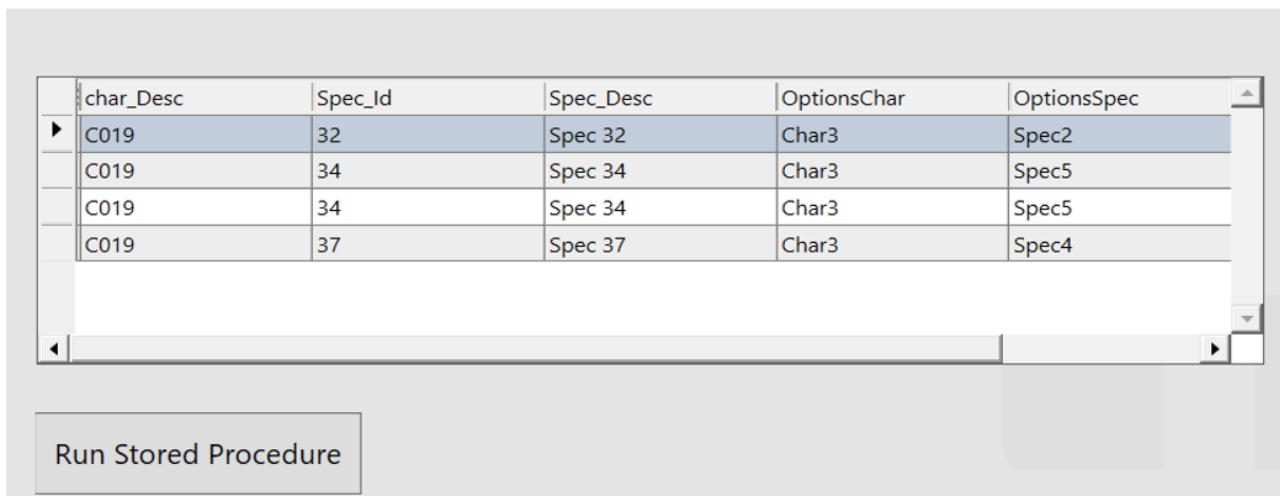
In **Datasets > Queries**, create a new query assigned for this DB connection.

In the Draw Environment, a DataGrid was added and its DataSource was set for the Query created above. Then, a button is used to run the script below:

```
public void RunSP_Btn(object sender, System.Windows.Input.InputEventArgs e)
{
    string sqlStatement = string.Format("exec TestSP '{0}'", "C019"); @Dataset.Query.Query_SqlServer.SqlStatement = sqlStatement;

    string sts = ""; @Dataset.Query.Query_SqlServer.ExecuteCommandWithStatus(out sts);
}
```

After executing the code above, the result should be the same as in SSMS.



## Test Results

Test Description	Results
Creating Procedure in SQL Server.	OK.
Running Procedure in SSMS.	OK.
Running Procedure in <i>fs-9.1.9</i> .	OK.