Instructor:

Nazeef- ul Haq
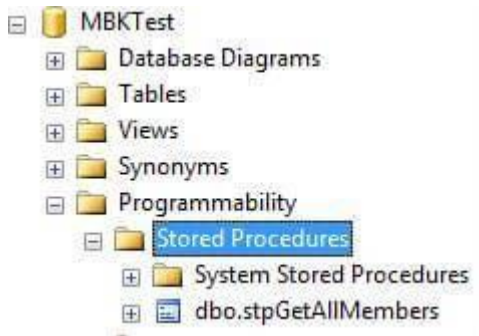# Database System

Session: 2022 – 2025

Name:	Muhammad Ahmad
Registration Number:	2022-CS-144
Section:	C

Department of Computer Science

University of Engineering and Technology

Lahore Pakistan

Execute stored procedures in SQL Server

In below UI, right click on the SP name and select Execute Stored Procedure... to execute a SP. From here, you can also modify an existing SP.
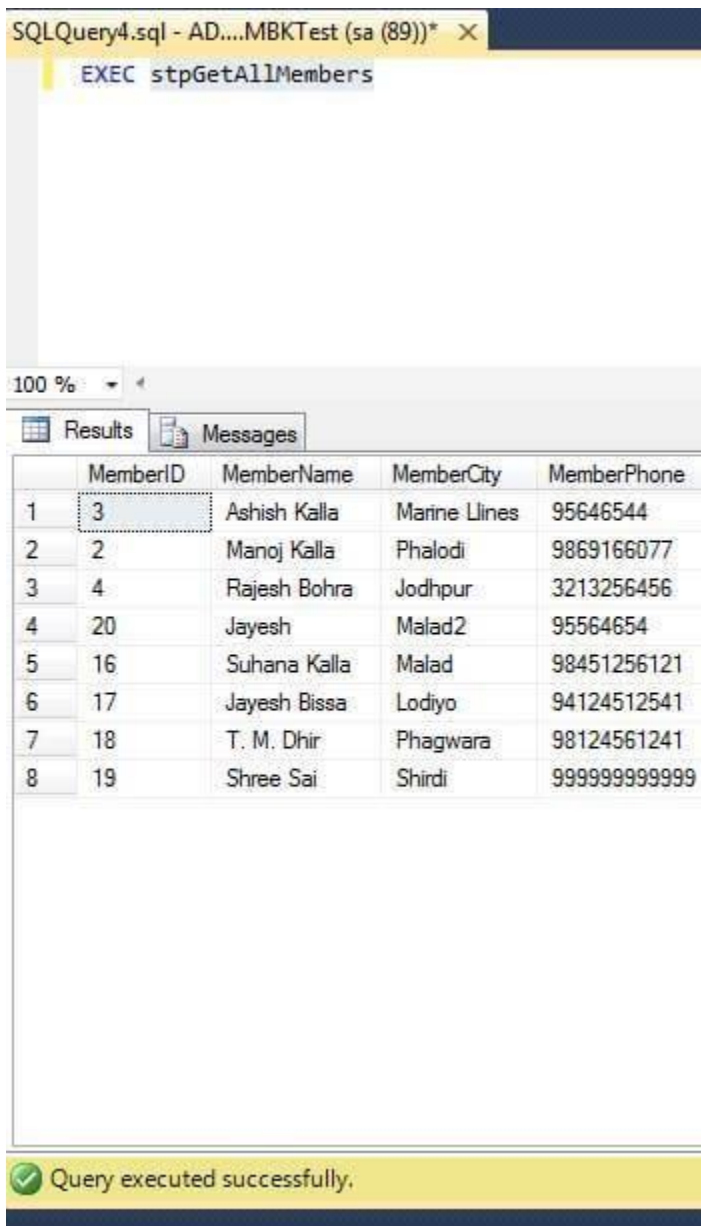
Alternatively, you can also execute a SP from the Query window.

To run stored procedure in SQL Server Management Studio, switch to Query window or CTRL +N to open an new query window and type the following command.

- Syntax - EXEC <stored procedure name>
- Example - EXEC stpGetAllMembers

Now, we run our stored procedure called stpGetAllMembers. The output looks like the following:

OUTPUT


```
SQLQuery4.sql - AD....MBKTest (sa (89))*  ×
    EXEC stpGetAllMembers
```

100 %   ▾ ◂

	MemberID	MemberName	MemberCity	MemberPhone
1	3	Ashish Kalla	Marine Llines	95646544
2	2	Manoj Kalla	Phalodi	9869166077
3	4	Rajesh Bohra	Jodhpur	3213256456
4	20	Jayesh	Malad2	95564654
5	16	Suhana Kalla	Malad	98451256121
6	17	Jayesh Bissa	Lodiyo	94124512541
7	18	T. M. Dhir	Phagwara	98124561241
8	19	Shree Sai	Shirdi	999999999999

Query executed successfully.

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.

How to create a SELECT query SP with parameters?

In the previous steps, we created a simple SP that returned all rows from a table. Now, let's create a new SP that will take a city name as an inpurt parameter and will return all rows where city name matches the input parameter value.

Here is the updated SP with a parameter @CityName.

```
1.  SET ANSI_NULLS ON
2.  GO
3.  SET QUOTED_IDENTIFIER ON
4.  GO
5.  -- =============================================
6.  -- Author:     Muhammad Ahmad
7.  -- Create date: 21-Nov-2024
8.  -- Description: Delete a Member by Member ID
9.  -- =============================================
10. CREATE PROCEDURE stpGetMembersByCityName
11.     -- Add the parameters for the stored procedure here
12.     @CityName nvarchar(30)
13.
14. AS
15. BEGIN
16.     -- SET NOCOUNT ON added to prevent extra result sets from
17.     -- interfering with SELECT statements.
18.     SET NOCOUNT ON;
19.
20.     Select * From tblMembers
21.     where MemberCity like '%'+@CityName+'%'
22.
23. END
24. GO
```

Execute it.

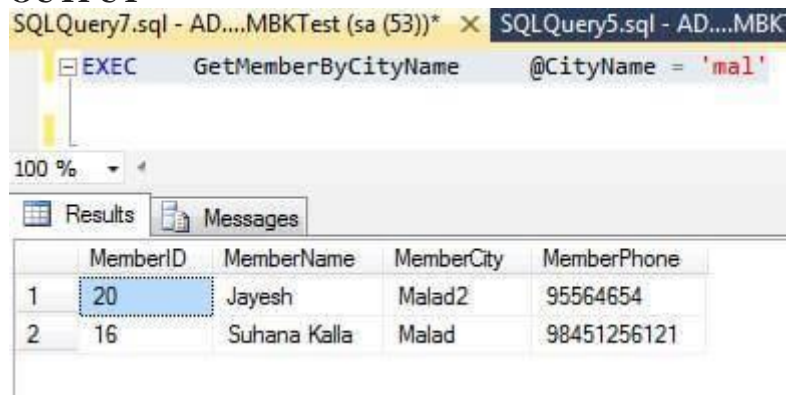To run this SP, type the following command in SQL query tool:

EXEC GetMemberByCityName @CityName = 'mal'

OR from the UI, run the SP and provide the following input.

The code to execute looks like the following:

```
1.  USE [MBKTest]
2.  GO
3.
4.  DECLARE @return_value int
5.
6.  EXEC    @return_value = [dbo].[GetMemberByCityName]
7.       @CityName = N'mal'
8.
9.  SELECT 'Return Value' = @return_value
10.
11. GO
```
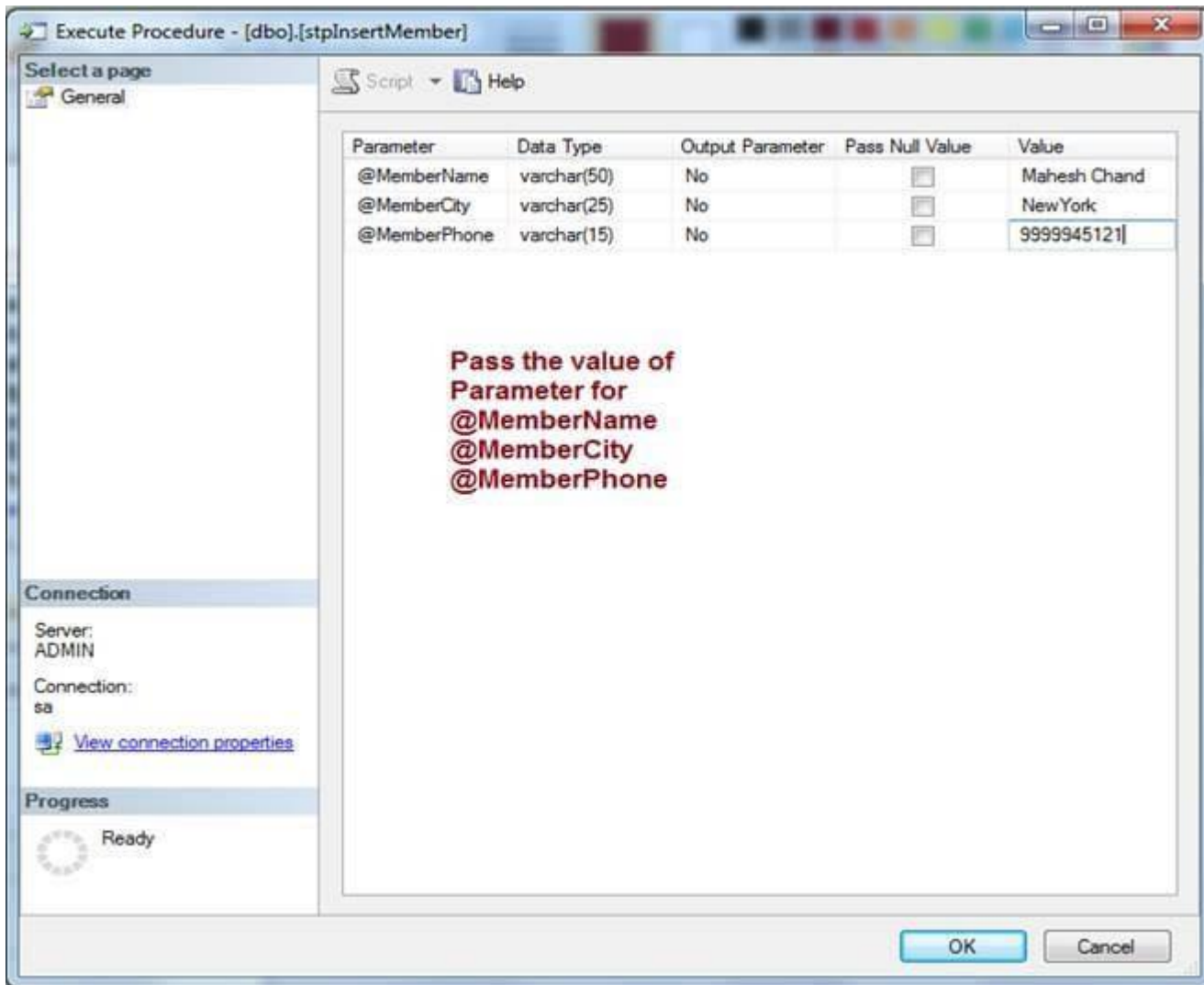
OUTPUT

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.

```
1.  SET ANSI_NULLS ON
2.  GO
3.  SET QUOTED_IDENTIFIER ON
4.  GO
5.  -- =============================================
6.  -- Author:      Muhammad Ahmad
7.  -- Create date: 21-Nov-2024
8.  -- Description: Delete a Member by Member ID

10. CREATE PROCEDURE stpInsertMember
```

```
11. @MemberName varchar(50),
12. @MemberCity varchar(25),
13. @MemberPhone varchar(15)
14.
15. AS
16. BEGIN
17.     -- SET NOCOUNT ON added to prevent extra result sets from
18.     -- interfering with SELECT statements.
19.     SET NOCOUNT ON;
20.
21.     Insert into tblMembers (MemberName,MemberCity,MemberPhone)
22.         Values (@MemberName,@MemberCity, @MemberPhone)
23.
24. END
25. GO
```

Right click on stored procedure in Object Explorer and select Refresh.
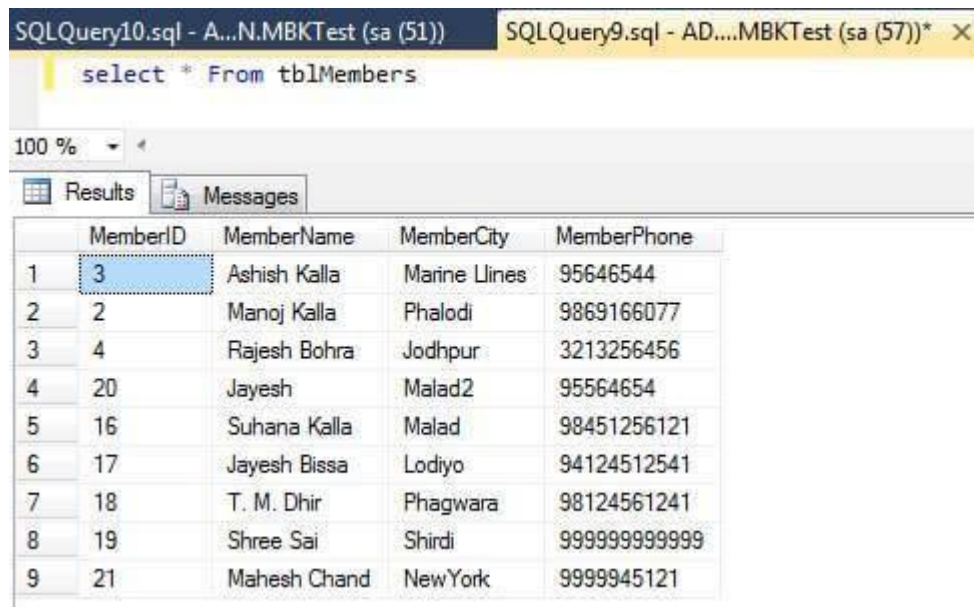
The following code can be used to execute this SP in SSMS.

```
1.  USE [MBKTest]
2.  GO
3.
4.  DECLARE @return_value int
5.
6.   EXEC    @return_value = [dbo].[stpInsertMember]
7.       @MemberName = N'Mahesh Chand',
8.       @MemberCity = N'NewYork',
9.       @MemberPhone = N'9999945121'
10. SELECT  'Return Value' = @return_value
11. GO
```

OUTPUT

In the query window, you can check if a new record for Member Name 'Mahesh Chand' is added to the table.
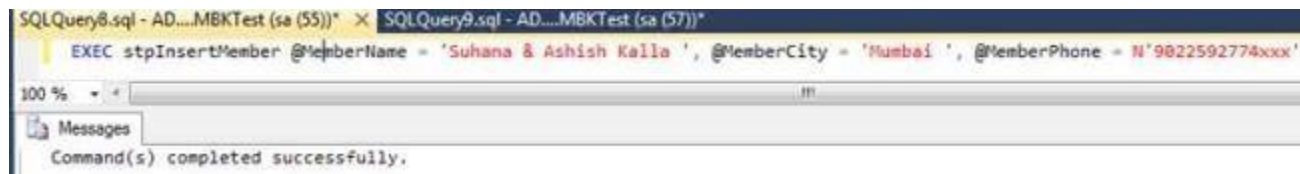

OUTPUT

You can check "Suhana & Ashish Kalla" record is added successfully.

How to create an UPDATE quert based stored procedure?

Let's create a new SP that will update a table records based on the Member ID column. The ID is passed as an input parameter. Here is the new SP that uses an UPDATE..SET..WHERE command.

```
1.  SET ANSI_NULLS ON
2.  GO
3.  SET QUOTED_IDENTIFIER ON
4.  GO
5.  -- =============================================
6.  -- Author:      Manoj Kalla
7.  -- Create date: 20-Nov-2017
8.  -- Description: Update a member detail by ID
9.  -- =============================================
10. CREATE PROCEDURE stpUpdateMemberByID
11. @MemberID int,
12. @MemberName varchar(50),
13. @MemberCity varchar(25),
14. @MemberPhone varchar(15)
15.
16. AS
17. BEGIN
18.     -- SET NOCOUNT ON added to prevent extra result sets from
19.     -- interfering with SELECT statements.
20.     SET NOCOUNT ON;
21.
22.     UPDATE tblMembers
23.     Set MemberName = @MemberName,
```
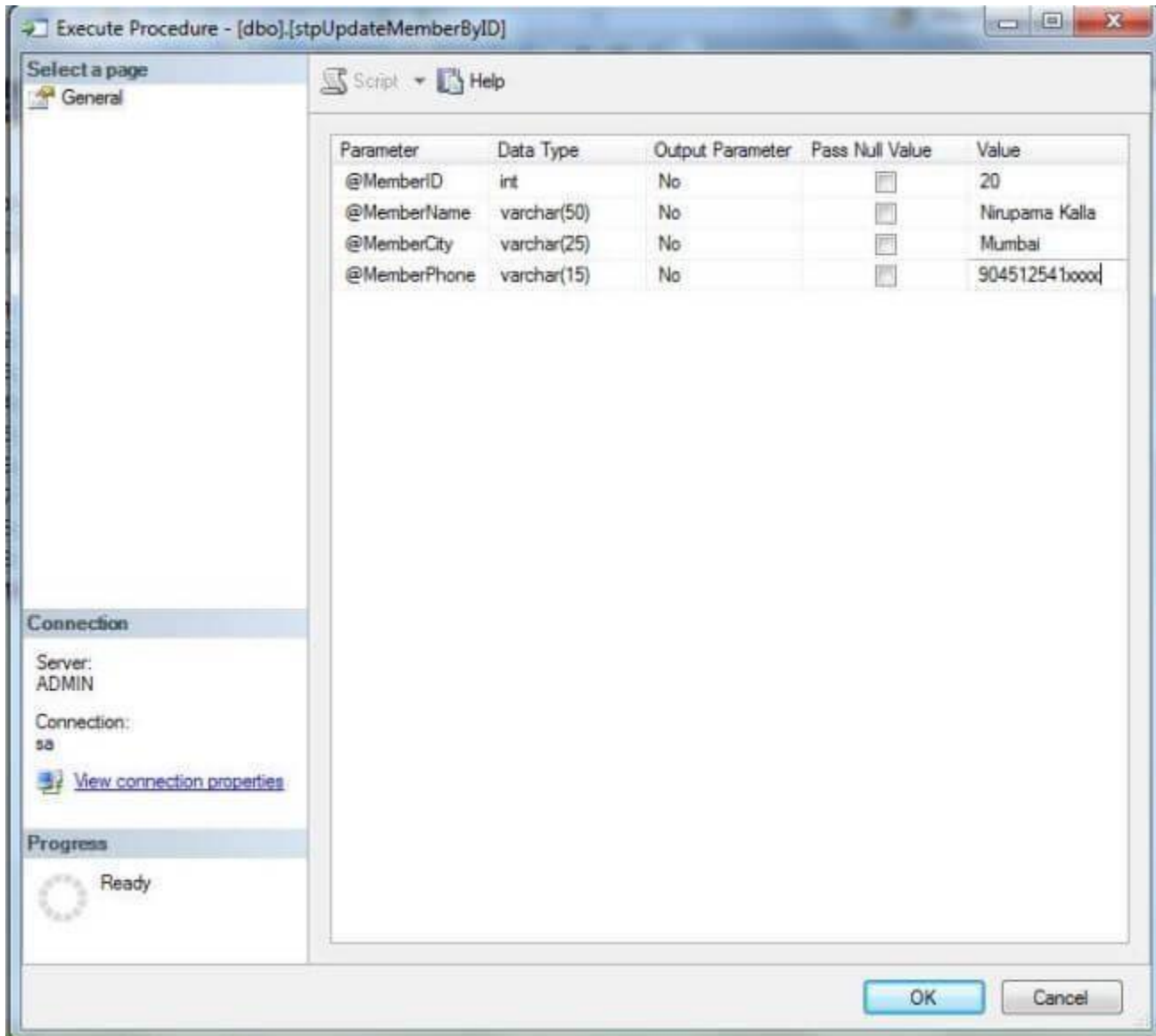
```
24.        MemberCity = @MemberCity,
25.        MemberPhone = @MemberPhone
26.    Where MemberID = @MemberID
27. END
28. GO
```

Right click on stored procedure in the Object Explorer and select Refresh. You will see the SP is created.

Now, Right click on SP name and select Execute stored procedure…. Provide the input values and execute.

```
5.  -- =============================================
6.  -- Author:      Muhammad Ahmad
7.  -- Create date: 21-Nov-2024
8.  -- Description: Delete a Member by Member ID
9.  -- =============================================
10. CREATE PROCEDURE stpDeleteMemberByMemberID
11.     @MemberID int
12. AS
13. BEGIN
14.     -- SET NOCOUNT ON added to prevent extra result sets from
15.     -- interfering with SELECT statements.
16.     SET NOCOUNT ON;
17.
18.     Delete from tblMembers
19.     where MemberId = @MemberID
20.
21. END
22. GO
```
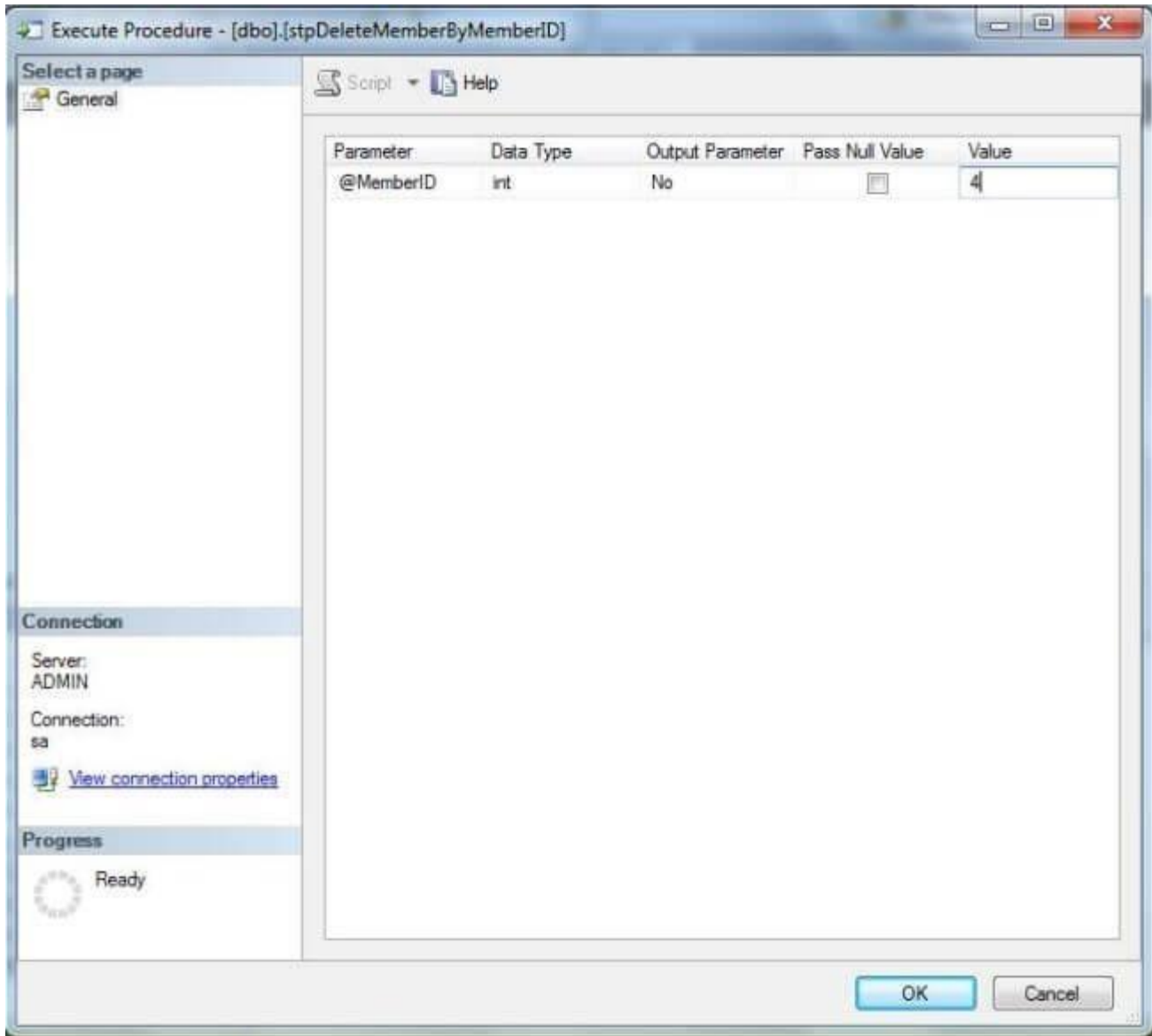
Execute it.

Right click on Stored Procedures in the Object Explorer and select Refresh.

RUN stored procedure BY UI

Now again right click on stored procedure and select Execute stored procedure…

As you can see in the image, I passed @MemberID parameter value = 4.

RUN DELETE stored procedure BY MANUALLY (CODING)

EXEC stpDeleteMemberByMemberID 2

OUTPUT

You can see in image MemberID = 4 record has been deleted successfully.

	MemberID	MemberName	MemberCity	MemberPhone
1	3	Ashish Kalla	Marine Llines	95646544
2	2	Manoj Kalla	Phalodi	9869166077
3	20	Nirupama Kalla	Mumbai	904512541xxxx
4	16	Suhana Kalla	Malad	98451256121
5	17	Gopal Madhavrai	Bikaner	90454564xxx
6	18	T. M. Dhir	Phagwara	98124561241
7	19	Shree Sai	Shirdi	999999999999
8	21	Mahesh Chand	NewYork	9999945121
9	22	Suhana & Ashish Kalla	Mumbai	9022592774xxx