Advanced SQL III

Demo - Stored Procedures

Stored Procedure

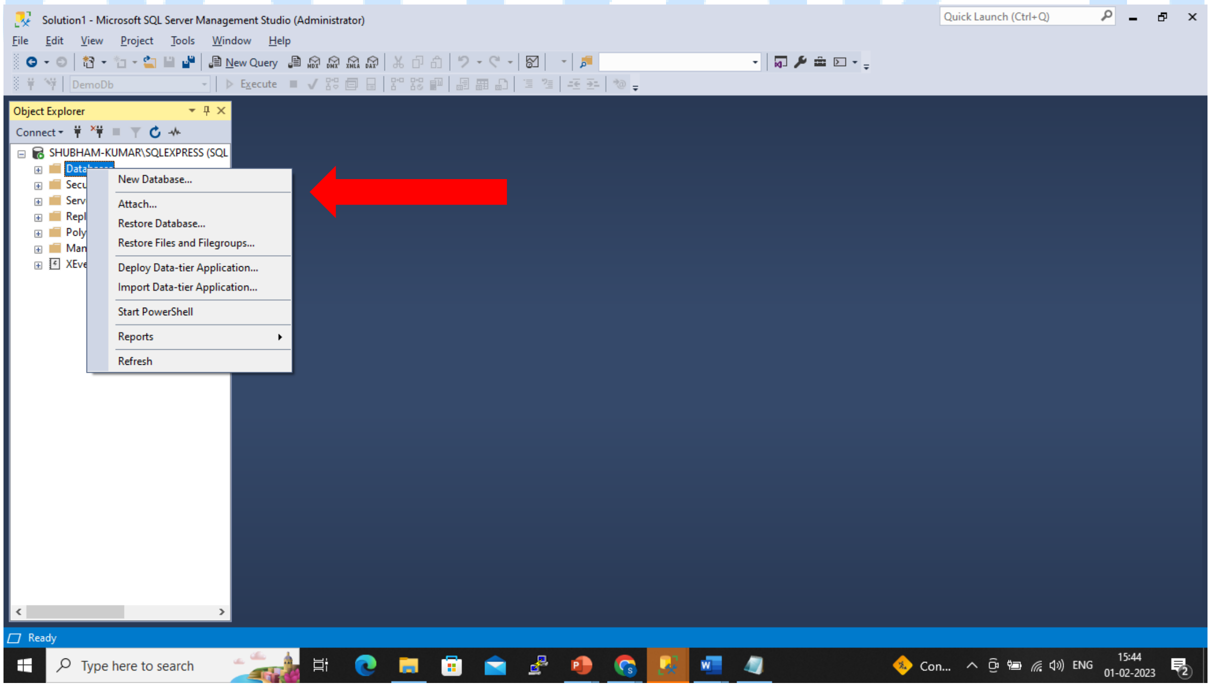
Problem Statement: Create a database named '**MyStoredDB**' depicting employee details in the ABC company then create a table namely employee. Perform several operations on the User Defined Stored Procedure such as creation, modify, parameters, and encrypting Stored Procedure.
Database description:

employee

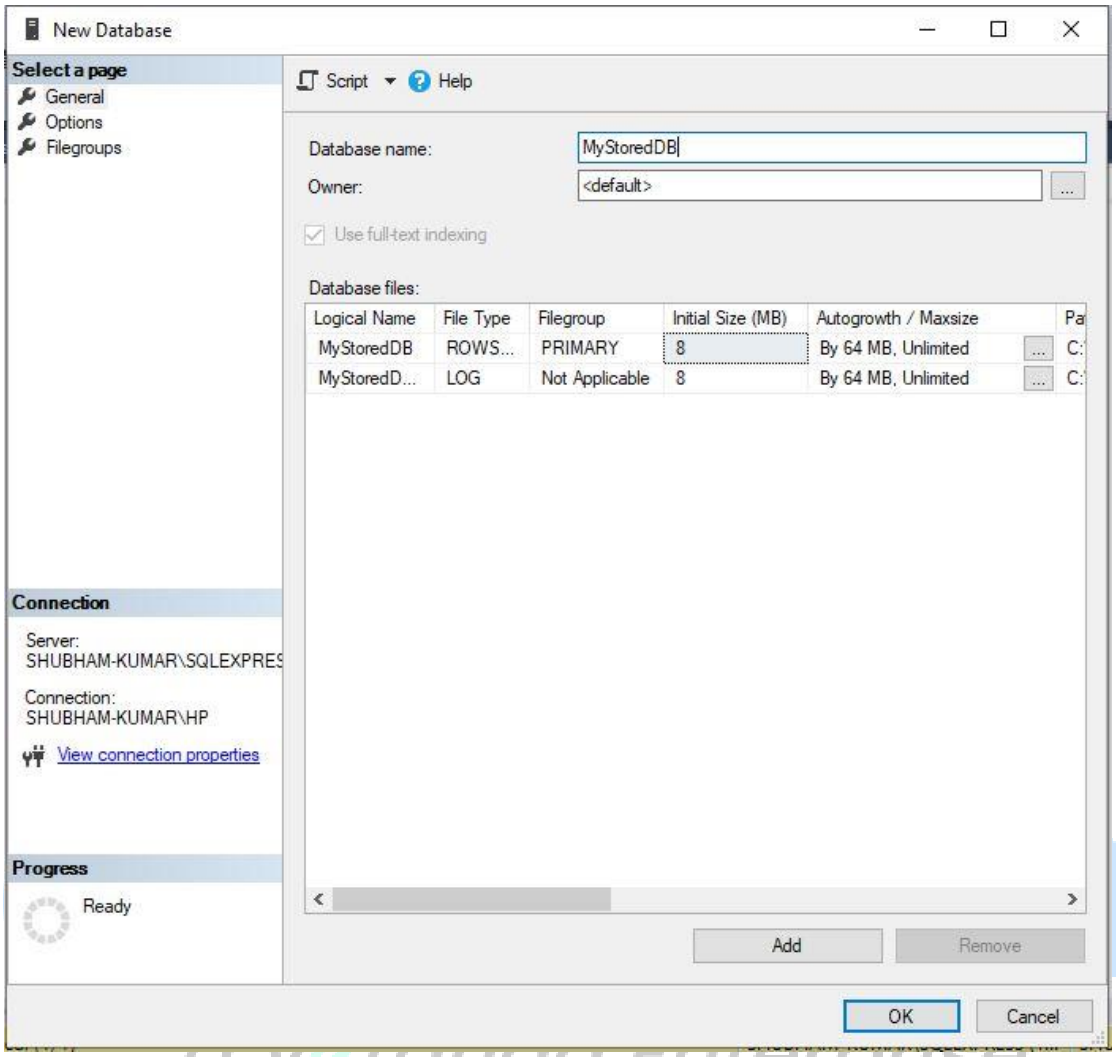
- **empID** – It represents the unique ID of employee.
- **empName** – Represents the employee's name.
- **deptID** – It represents the department unique ID.
- **salary –** Specifies the salary of the employee.
- **joinYear** – Represents the joining year of the employee.

.

Working on the Demo
Step 1: Create a database named **MyStoredDB** by right-clicking on databases.


Solution1 - Microsoft SQL Server Management Studio (Administrator)
Object Explorer
SHUBHAM-KUMAR\SQLEXPRESS (SQL
New Database...
Attach...
Restore Database...
Restore Files and Filegroups...
Deploy Data-tier Application...
Import Data-tier Application...
Start PowerShell
Reports
Refresh

Step 2: Put in the Database name **MyStoredDB**


New Database

Select a page: General, Options, Filegroups

Database name: MyStoredDB
Owner: <default>
Use full-text indexing

Database files:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize	Pa
MyStoredDB	ROWS...	PRIMARY	8	By 64 MB, Unlimited ...	C:
MyStoredD...	LOG	Not Applicable	8	By 64 MB, Unlimited ...	C:

Connection
Server: SHUBHAM-KUMAR\SQLEXPRES
Connection: SHUBHAM-KUMAR\HP
View connection properties

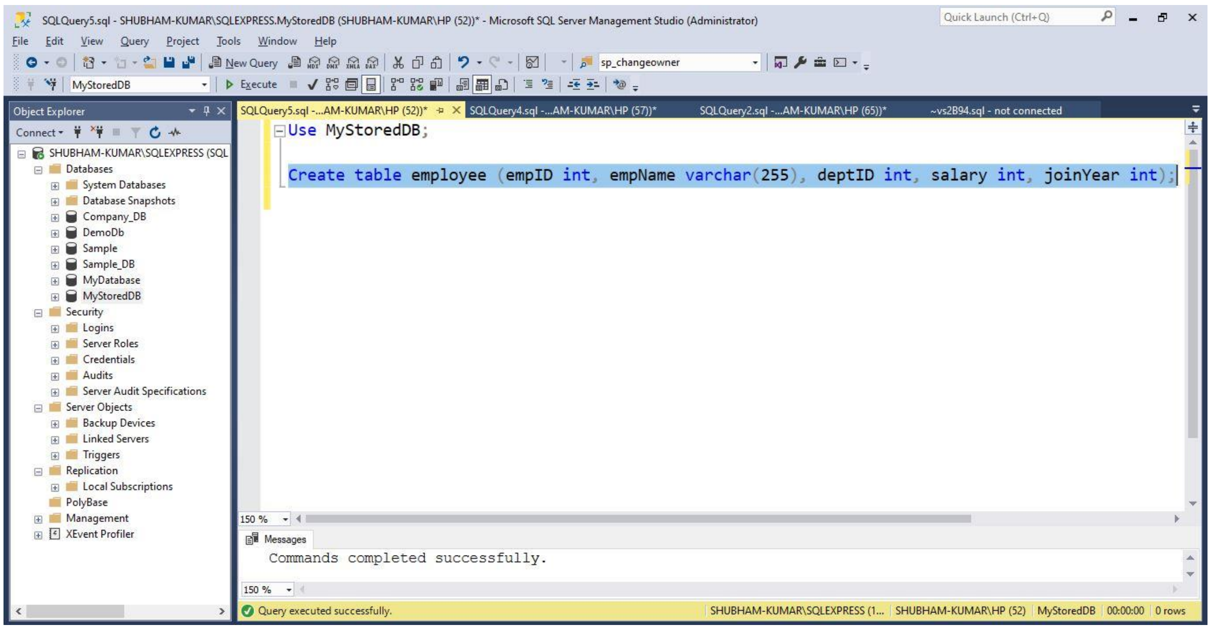
Progress: Ready

Note: Insert the values in all the respective tables.

For the reference, the database '**MyStoredDB**' file has been attached.

MyStoredDB.bak

Step 3: Create a table name employee with the parameters such as **empID empName**, **deptID**, **salary, and joinYear**.

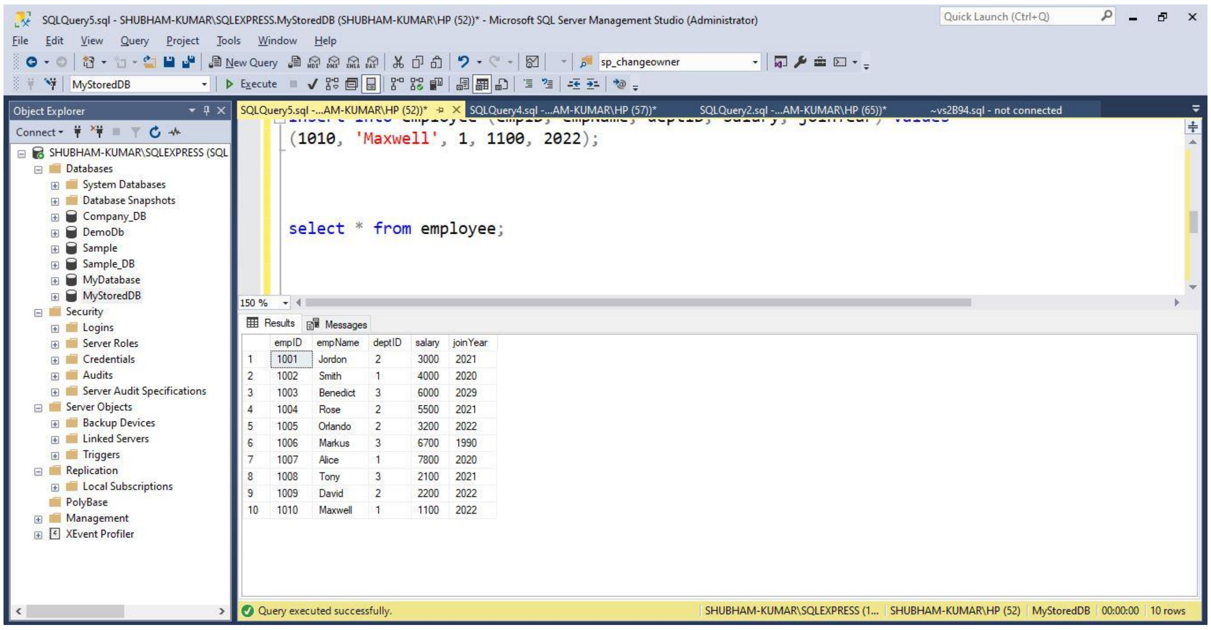


```
Use MyStoredDB;

Create table employee (empID int, empName varchar(255), deptID int, salary int, joinYear int);
```

```
Commands completed successfully.
```

Query executed successfully.

Step 4: Insert the data in the employee table.



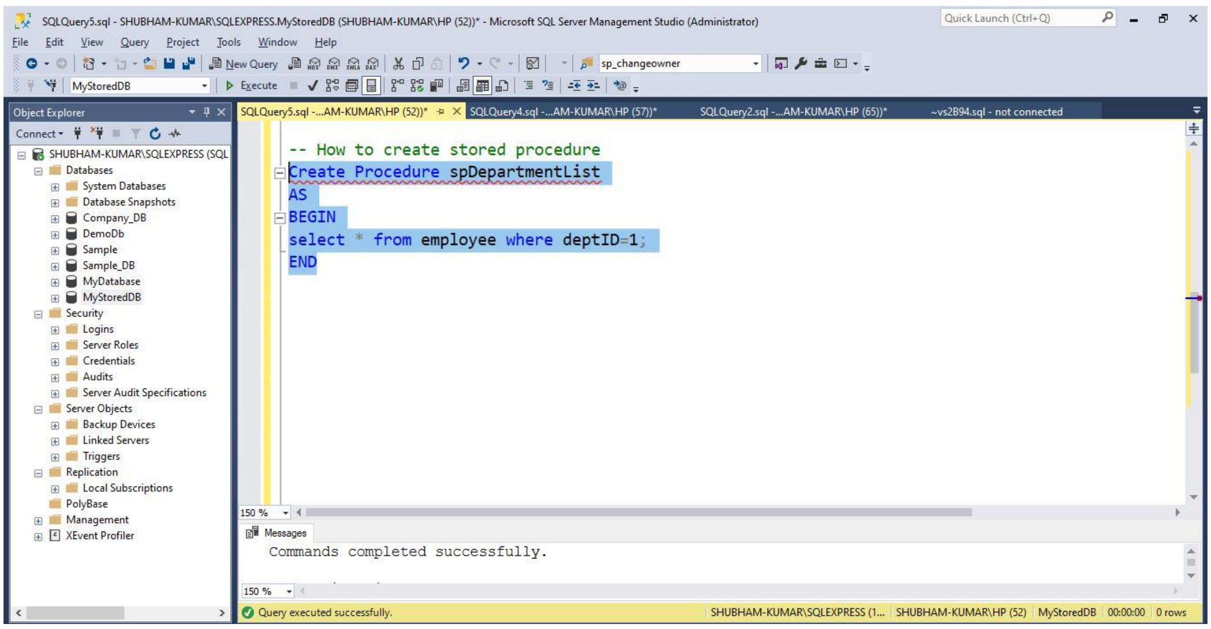
```
(1010, 'Maxwell', 1, 1100, 2022);


select * from employee;
```

	empID	empName	deptID	salary	joinYear
1	1001	Jordon	2	3000	2021
2	1002	Smith	1	4000	2020
3	1003	Benedict	3	6000	2029
4	1004	Rose	2	5500	2021
5	1005	Orlando	2	3200	2022
6	1006	Markus	3	6700	1990
7	1007	Alice	1	7800	2020
8	1008	Tony	3	2100	2021
9	1009	David	2	2200	2022
10	1010	Maxwell	1	1100	2022

Query executed successfully.

Step 5: Create a Stored Procedure **spDepartmentList** for selecting the employees where deptID=1

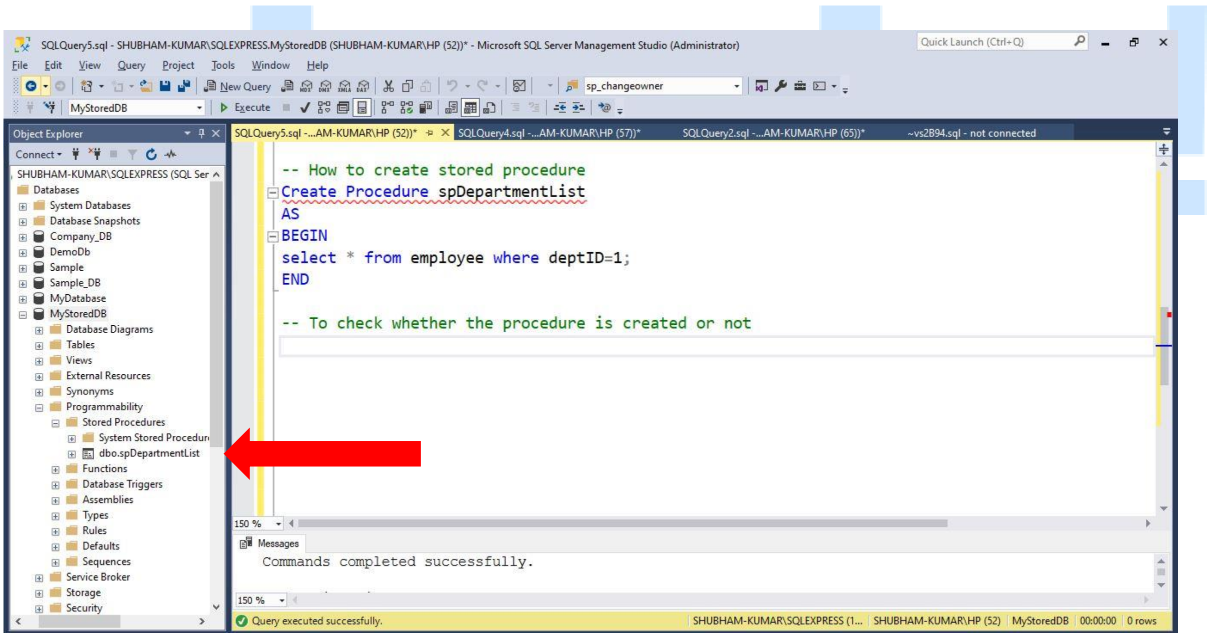
```
Create Procedure spDepartmentList
AS
BEGIN
select * from employee where deptID=1;
END
```



```
-- How to create stored procedure
Create Procedure spDepartmentList
AS
BEGIN
select * from employee where deptID=1;
END
```

```
Commands completed successfully.
```

Step 6: In the Object Explorer panel, the **spDepartmentList** is created under the Stored Procedure.

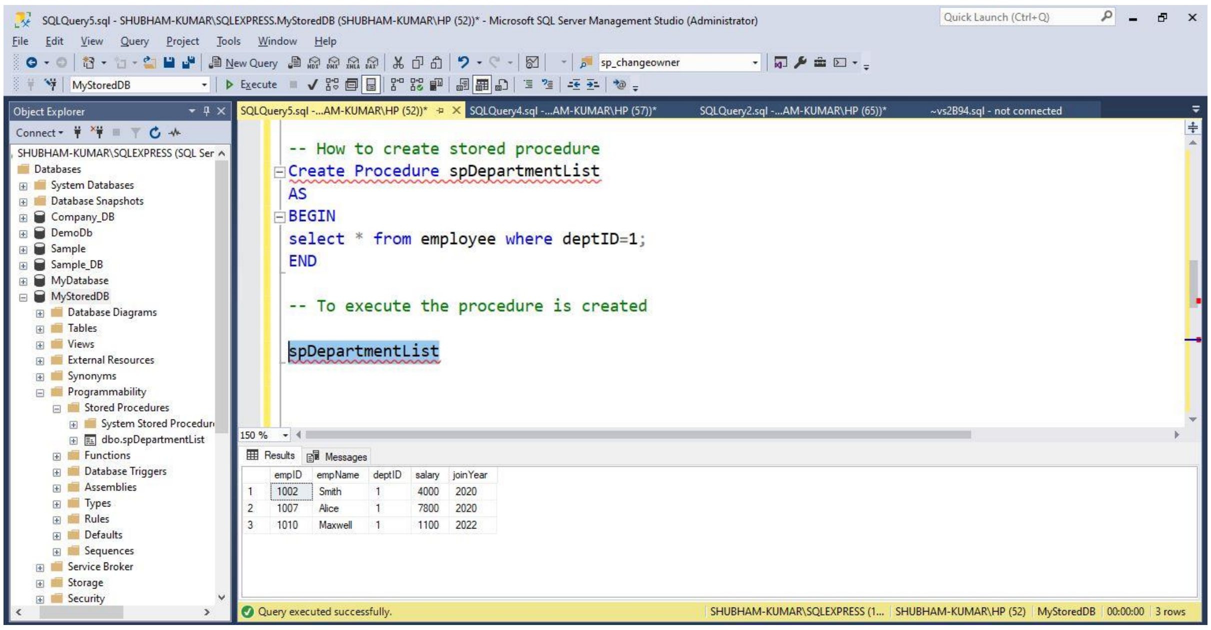


```
-- How to create stored procedure
Create Procedure spDepartmentList
AS
BEGIN
select * from employee where deptID=1;
END

-- To check whether the procedure is created or not
```

```
Commands completed successfully.
```

Step 7: Execute the Stored procedure **spDepartmentList** command.

```
spDepartmentList
```



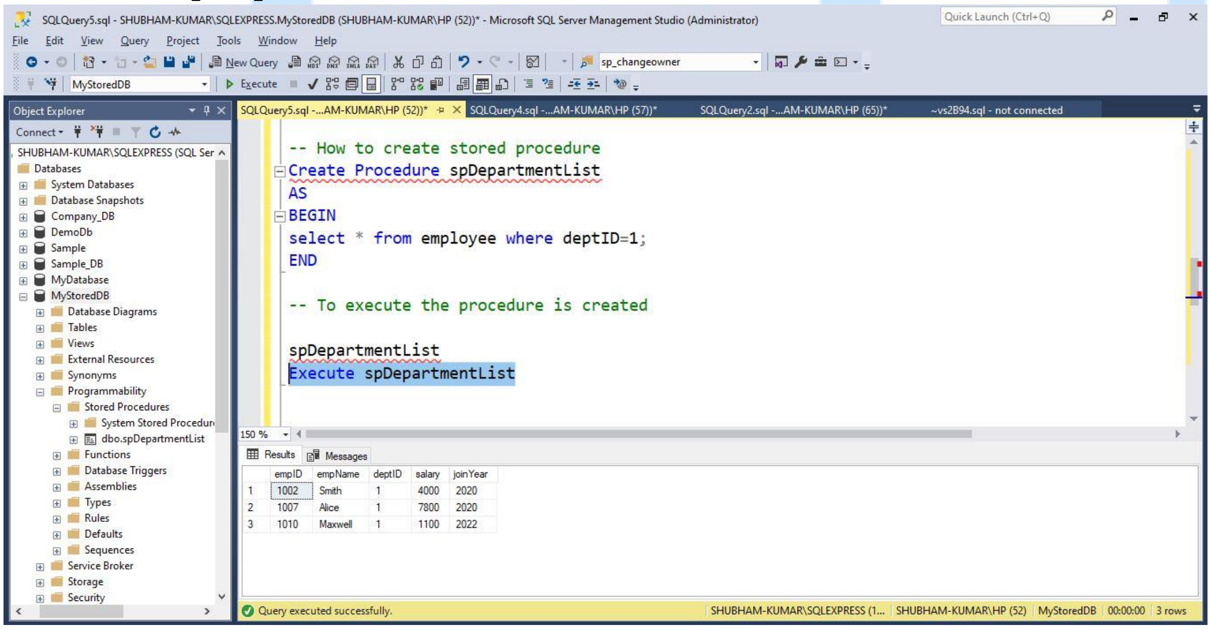
```sql
-- How to create stored procedure
Create Procedure spDepartmentList
AS
BEGIN
select * from employee where deptID=1;
END

-- To execute the procedure is created

spDepartmentList
```

	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020
2	1007	Alice	1	7800	2020
3	1010	Maxwell	1	1100	2022

Query executed successfully.

Step 8: Alternatively, execute the Stored procedure using **Execute spDepartmentList** command.

```
Execute spDepartmentList
```



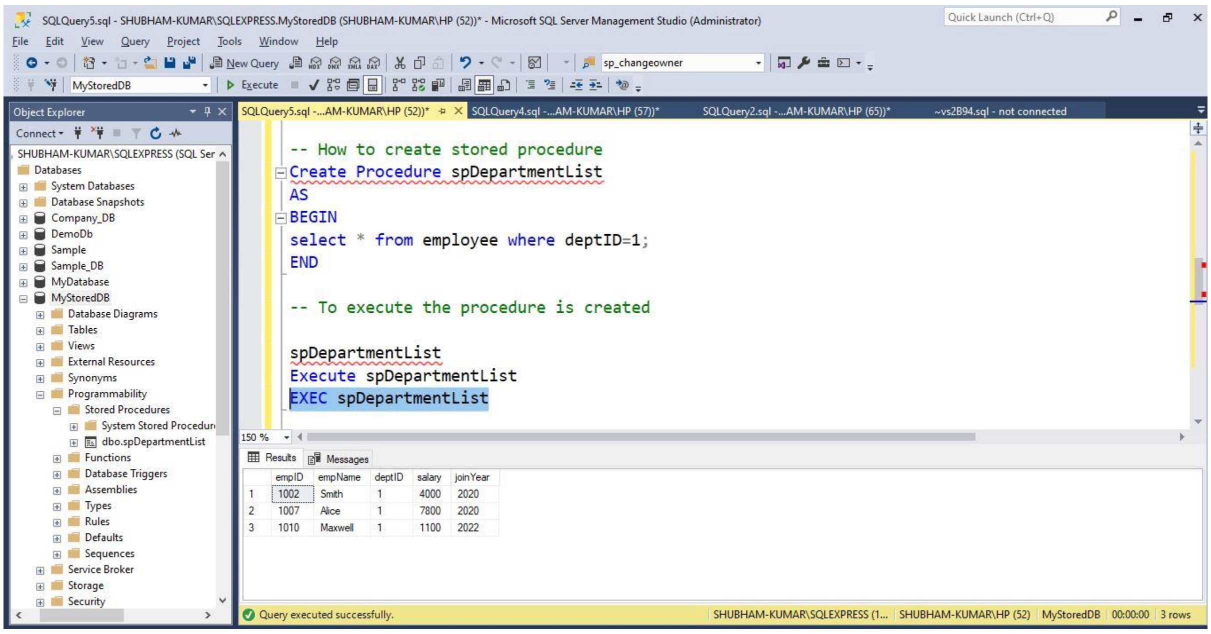
```sql
-- How to create stored procedure
Create Procedure spDepartmentList
AS
BEGIN
select * from employee where deptID=1;
END

-- To execute the procedure is created

spDepartmentList
Execute spDepartmentList
```

	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020
2	1007	Alice	1	7800	2020
3	1010	Maxwell	1	1100	2022

Query executed successfully.

Step 9: EXEC spDepartmentList can also be used to execute the Stored procedure spDepartmentList

```
EXEC spDepartmentList
```



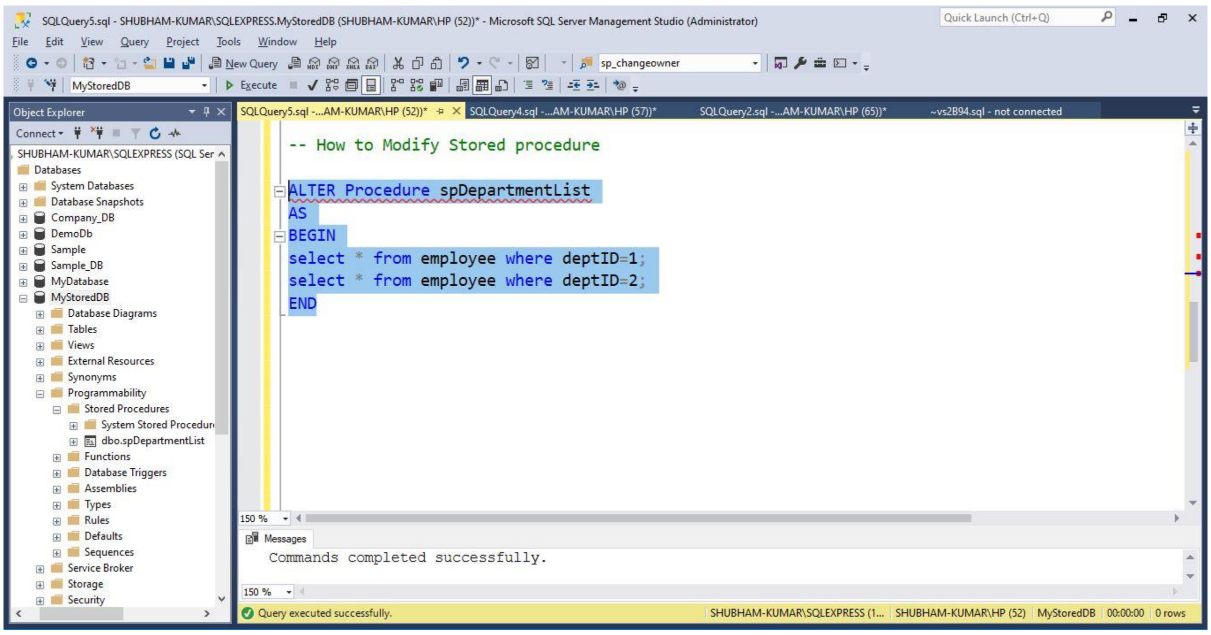
```
-- How to create stored procedure
Create Procedure spDepartmentList
AS
BEGIN
select * from employee where deptID=1;
END

-- To execute the procedure is created

spDepartmentList
Execute spDepartmentList
EXEC spDepartmentList
```

	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020
2	1007	Alice	1	7800	2020
3	1010	Maxwell	1	1100	2022

```
Query executed successfully.
```

Step 10: Modify the existing Stored Procedure using the ALTER command.

```
ALTER Procedure spDepartmentList
AS
BEGIN
select * from employee where deptID=1;
select * from employee where deptID=2;
END
```



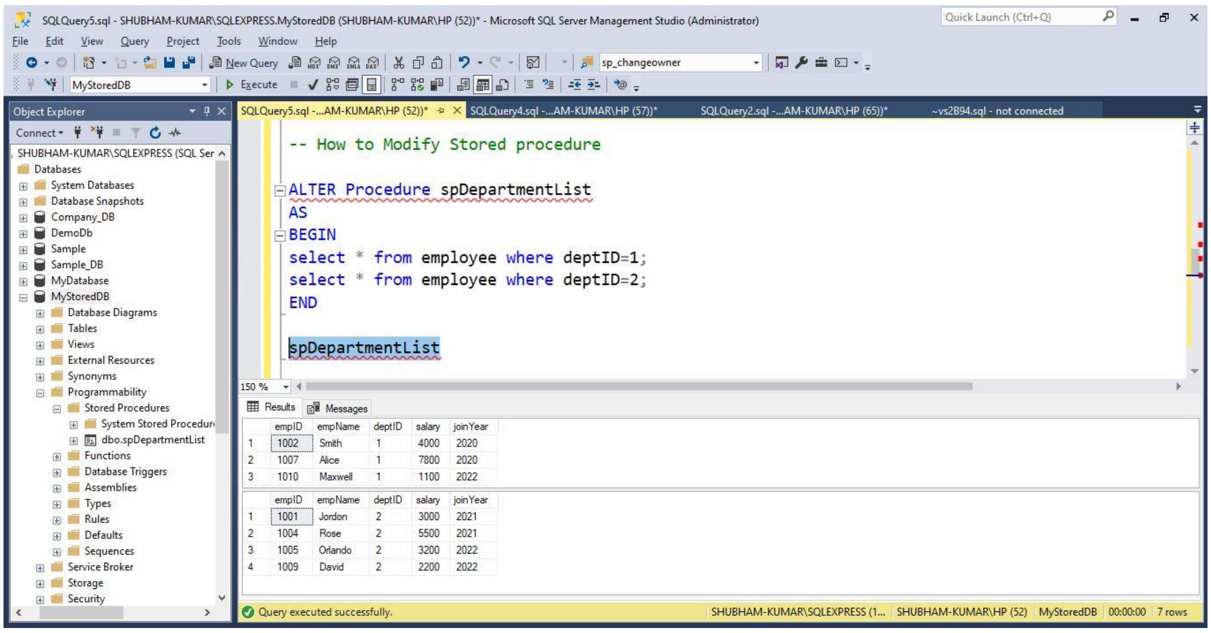
```
-- How to Modify Stored procedure

ALTER Procedure spDepartmentList
AS
BEGIN
select * from employee where deptID=1;
select * from employee where deptID=2;
END
```

```
Commands completed successfully.
```

```
Query executed successfully.
```

Step 11: Execute the Alter Stored procedure using **spDepartmentList** command.

```
spDepartmentList
```



```
-- How to Modify Stored procedure

ALTER Procedure spDepartmentList
AS
BEGIN
select * from employee where deptID=1;
select * from employee where deptID=2;
END

spDepartmentList
```

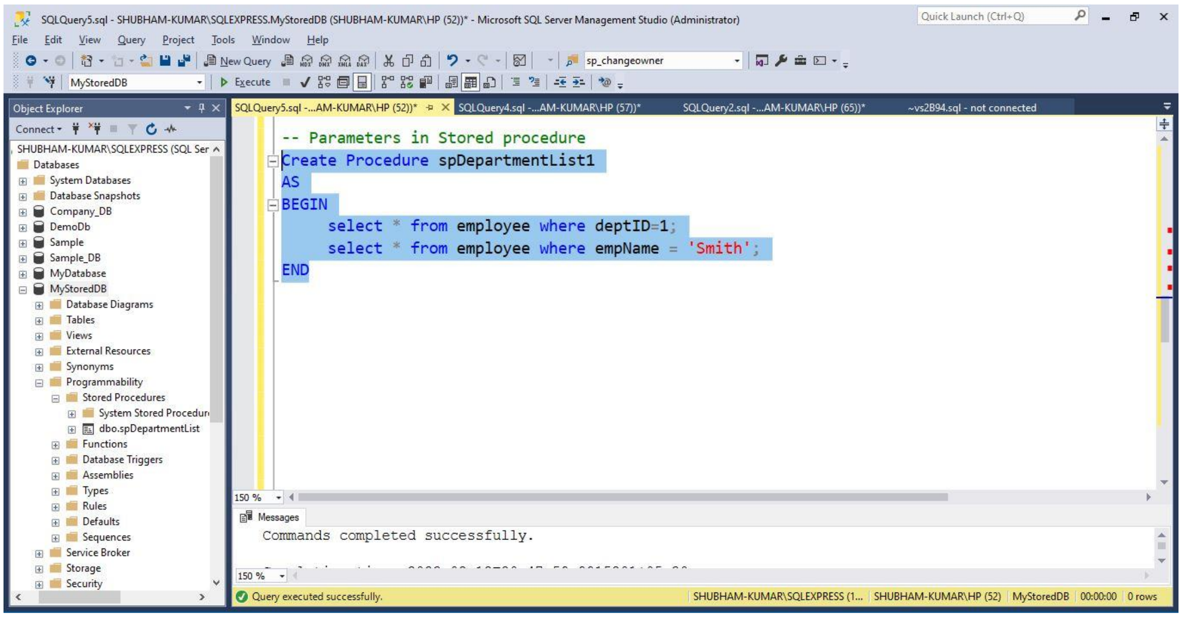
	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020
2	1007	Alice	1	7800	2020
3	1010	Maxwell	1	1100	2022

	empID	empName	deptID	salary	joinYear
1	1001	Jordon	2	3000	2021
2	1004	Rose	2	5500	2021
3	1005	Orlando	2	3200	2022
4	1009	David	2	2200	2022

```
Query executed successfully.    SHUBHAM-KUMAR\SQLEXPRESS (1...   SHUBHAM-KUMAR\HP (52)   MyStoredDB   00:00:00   7 rows
```

Step 12: Create another stored procedure **spDepartmentList1** for selecting the employees where deptID=1 and empName = 'Smith';

```
Create Procedure spDepartmentList1
AS
BEGIN
        select * from employee where deptID=1;
         select * from employee where empName = 'Smith';
END
```



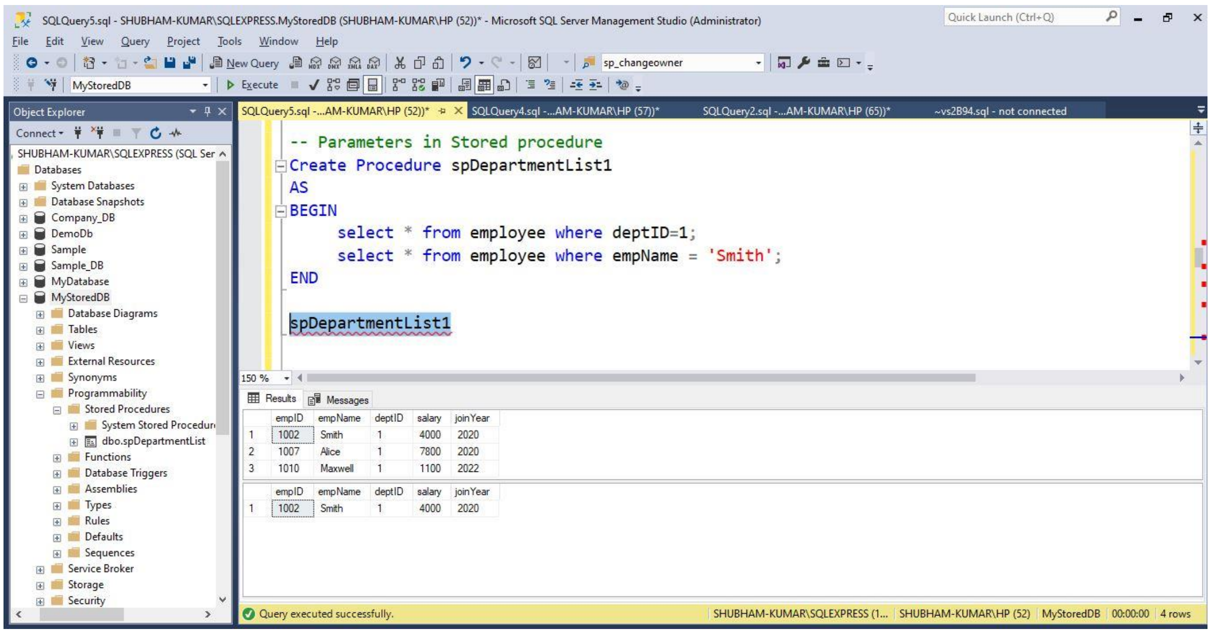
```
-- Parameters in Stored procedure
Create Procedure spDepartmentList1
AS
BEGIN
    select * from employee where deptID=1;
    select * from employee where empName = 'Smith';
END
```

```
Commands completed successfully.
```

```
Query executed successfully.    SHUBHAM-KUMAR\SQLEXPRESS (1...   SHUBHAM-KUMAR\HP (52)   MyStoredDB   00:00:00   0 rows
```

Step 13: Execute the **spDepartmentList1.**

```
spDepartmentList1
```



SQLQuery5.sql - SHUBHAM-KUMAR\SQLEXPRESS.MyStoredDB (SHUBHAM-KUMAR\HP (52))* - Microsoft SQL Server Management Studio (Administrator)

```
-- Parameters in Stored procedure
Create Procedure spDepartmentList1
AS
BEGIN
      select * from employee where deptID=1;
      select * from employee where empName = 'Smith';
END

spDepartmentList1
```

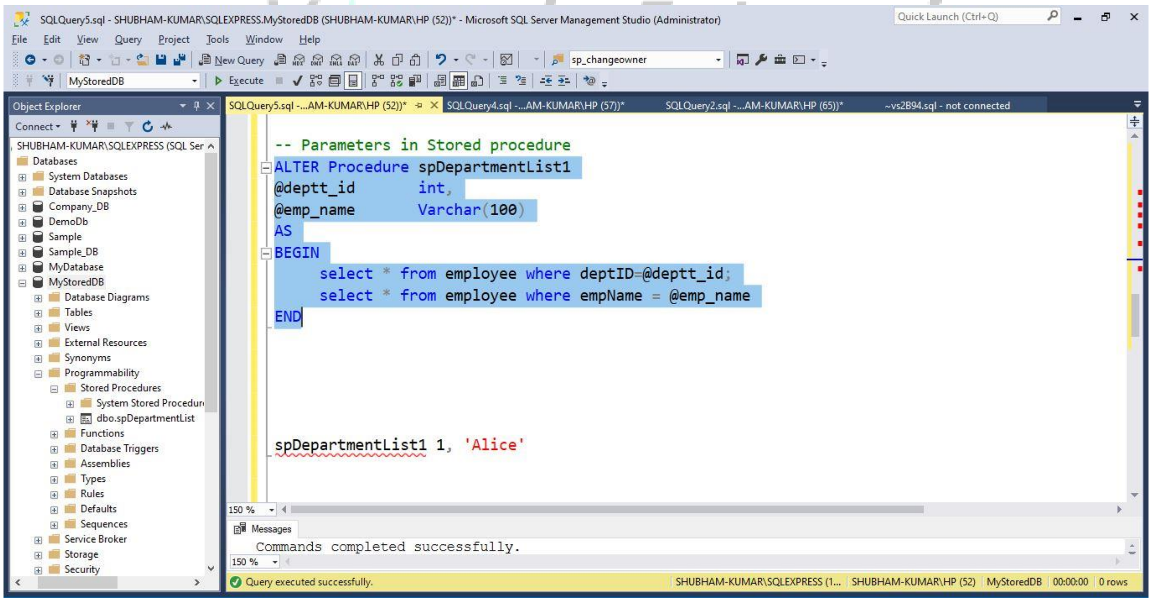
	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020
2	1007	Alice	1	7800	2020
3	1010	Maxwell	1	1100	2022

	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020

Query executed successfully. | SHUBHAM-KUMAR\SQLEXPRESS (1... | SHUBHAM-KUMAR\HP (52) | MyStoredDB | 00:00:00 | 4 rows

Step 14: Declare the parameter **deptt_id** and **emp_name** in the Stored Procedure. **deptt_id** is of integer type and **emp_name** is of character type.

```
ALTER Procedure spDepartmentList1
@deptt_id       int,
@emp_name       Varchar(100)
AS
BEGIN
     select * from employee where deptID=@deptt_id;
      select * from employee where empName = @emp_name
END
```



SQLQuery5.sql - SHUBHAM-KUMAR\SQLEXPRESS.MyStoredDB (SHUBHAM-KUMAR\HP (52))* - Microsoft SQL Server Management Studio (Administrator)

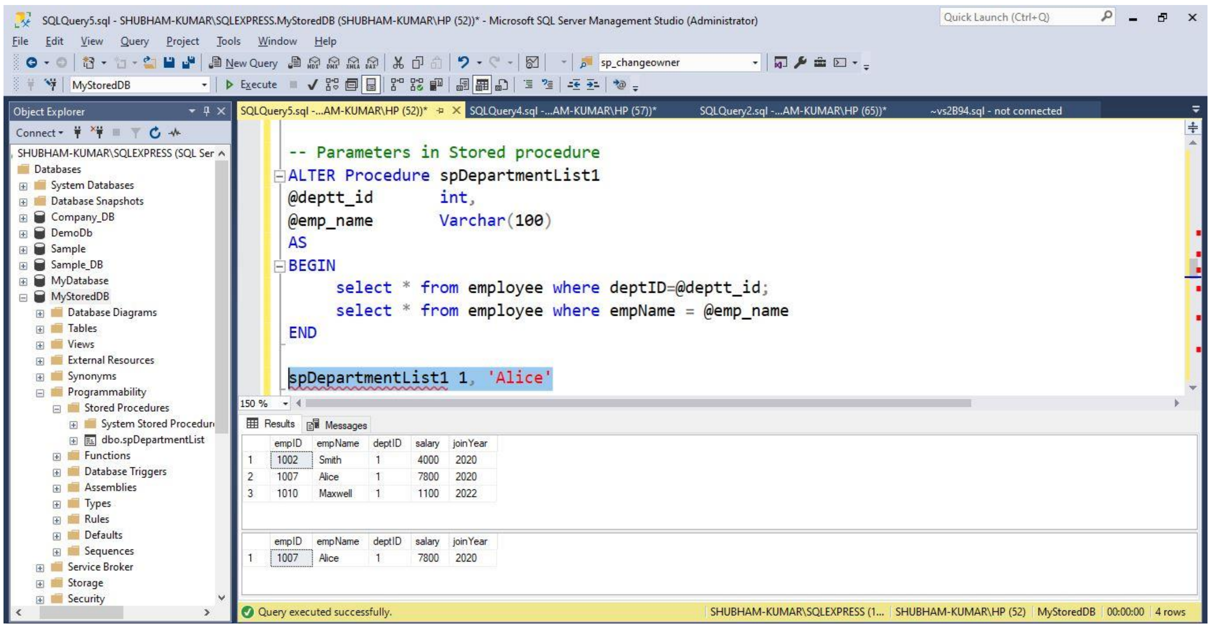
```
-- Parameters in Stored procedure
ALTER Procedure spDepartmentList1
@deptt_id       int,
@emp_name       Varchar(100)
AS
BEGIN
      select * from employee where deptID=@deptt_id;
      select * from employee where empName = @emp_name
END

spDepartmentList1 1, 'Alice'
```

```
Commands completed successfully.
```

Query executed successfully. | SHUBHAM-KUMAR\SQLEXPRESS (1... | SHUBHAM-KUMAR\HP (52) | MyStoredDB | 00:00:00 | 0 rows

Step 15: Execute the Stored Procedure **spDepartmentList1** with the parameter. Here the order of the parameter is important.

```
spDepartmentList1 1, 'Alice'
```



SQLQuery5.sql - SHUBHAM-KUMAR\SQLEXPRESS.MyStoredDB (SHUBHAM-KUMAR\HP (52))* - Microsoft SQL Server Management Studio (Administrator)

```sql
-- Parameters in Stored procedure
ALTER Procedure spDepartmentList1
@deptt_id       int,
@emp_name       Varchar(100)
AS
BEGIN
      select * from employee where deptID=@deptt_id;
      select * from employee where empName = @emp_name
END

spDepartmentList1 1, 'Alice'
```

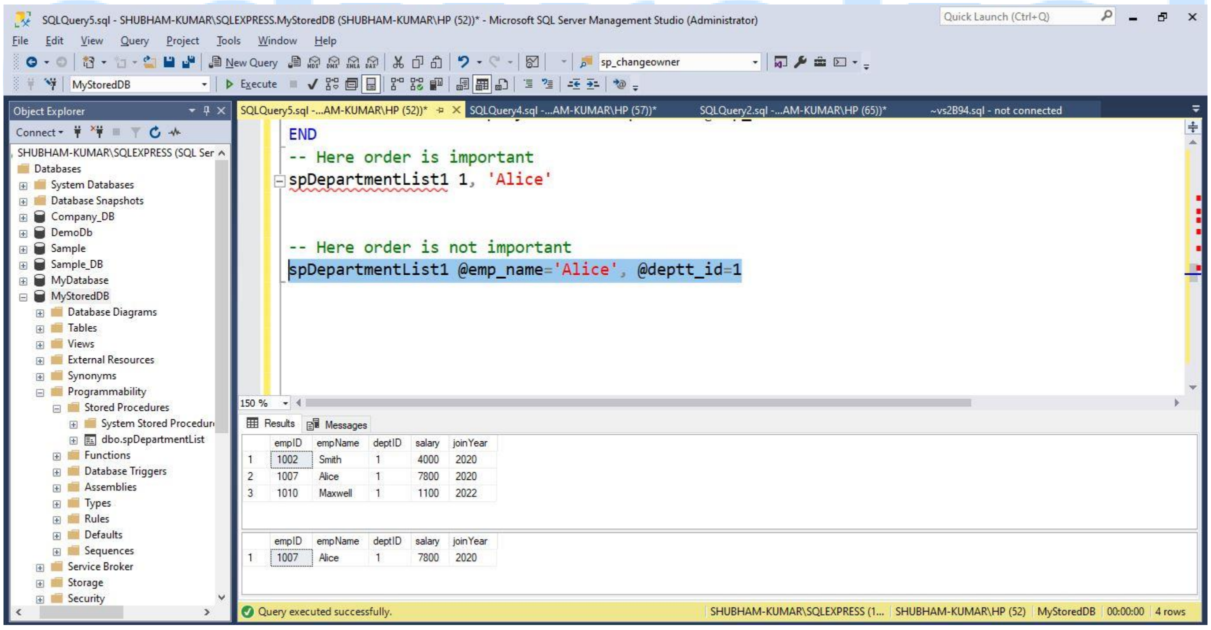
	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020
2	1007	Alice	1	7800	2020
3	1010	Maxwell	1	1100	2022

	empID	empName	deptID	salary	joinYear
1	1007	Alice	1	7800	2020

Query executed successfully. SHUBHAM-KUMAR\SQLEXPRESS (1... SHUBHAM-KUMAR\HP (52) MyStoredDB 00:00:00 4 rows

Step 16: Executing the Stored Procedure **spDepartmentList1** with the parameter, but here the order of the parameter is not important.

```
spDepartmentList1 @emp_name='Alice', @deptt_id=1
```



SQLQuery5.sql - SHUBHAM-KUMAR\SQLEXPRESS.MyStoredDB (SHUBHAM-KUMAR\HP (52))* - Microsoft SQL Server Management Studio (Administrator)

```sql
END
-- Here order is important
spDepartmentList1 1, 'Alice'


-- Here order is not important
spDepartmentList1 @emp_name='Alice', @deptt_id=1
```

	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020
2	1007	Alice	1	7800	2020
3	1010	Maxwell	1	1100	2022

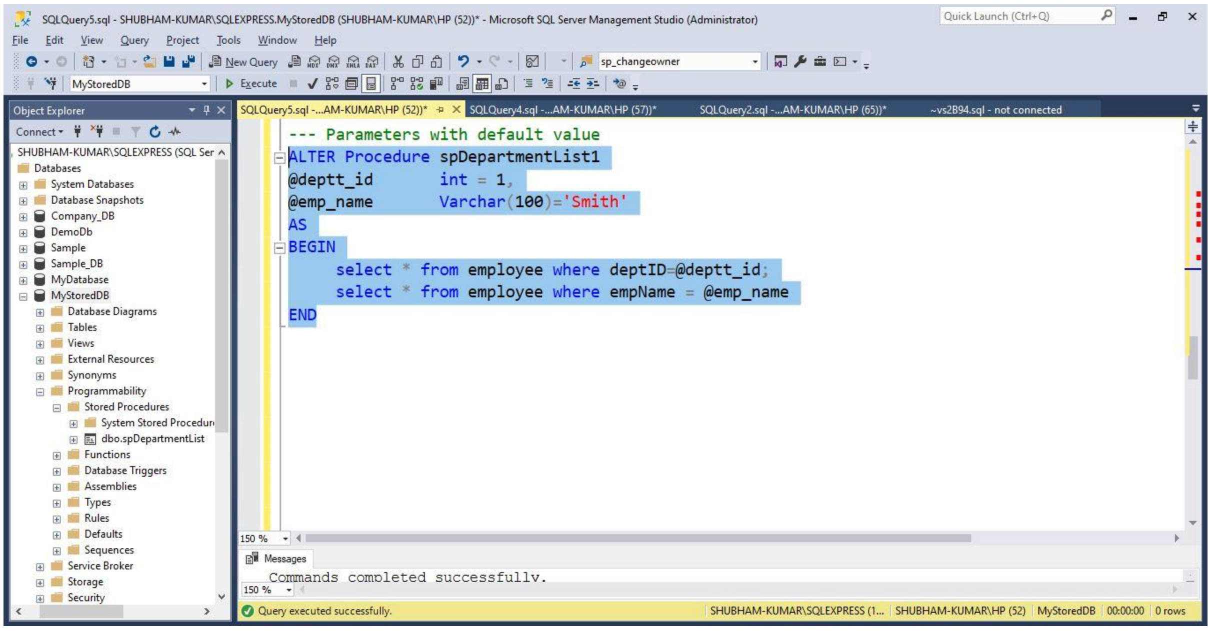
	empID	empName	deptID	salary	joinYear
1	1007	Alice	1	7800	2020

Query executed successfully. SHUBHAM-KUMAR\SQLEXPRESS (1... SHUBHAM-KUMAR\HP (52) MyStoredDB 00:00:00 4 rows

Step 17: Default (initial) value is assigned to the parameter.

```
ALTER Procedure spDepartmentList1
@deptt_id         int = 1,
@emp_name         Varchar(100)='Smith'
AS
BEGIN
     select * from employee where deptID=@deptt_id;
```

```
        select * from employee where empName = @emp_name
END
```



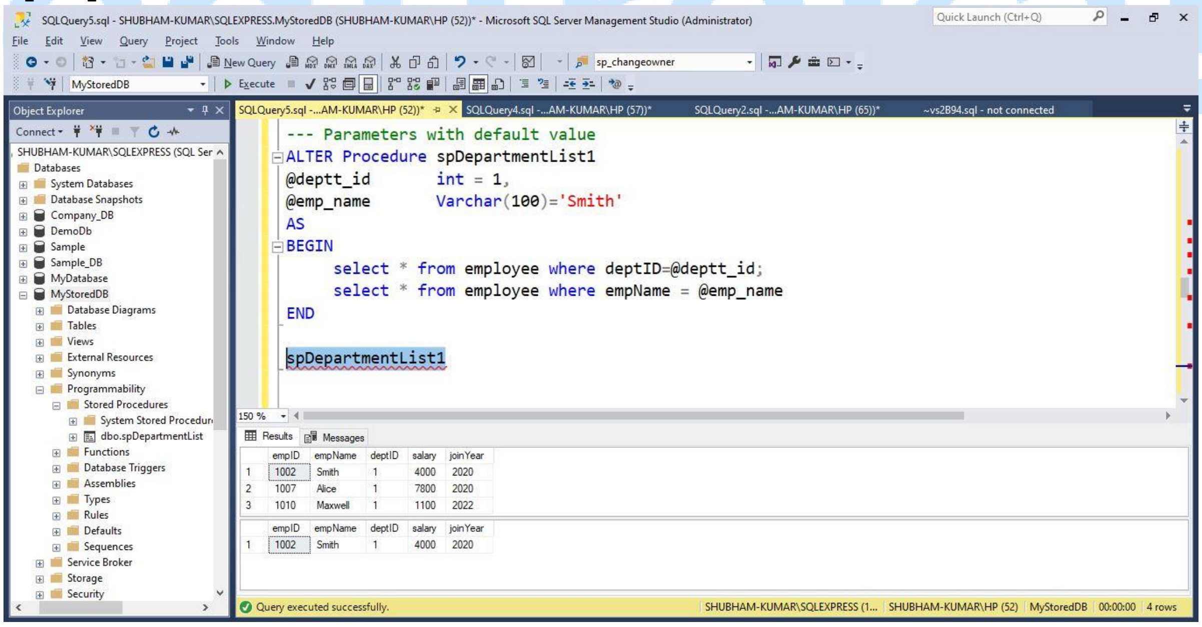
```
--- Parameters with default value
ALTER Procedure spDepartmentList1
@deptt_id       int = 1,
@emp_name       Varchar(100)='Smith'
AS
BEGIN
      select * from employee where deptID=@deptt_id;
      select * from employee where empName = @emp_name
END
```

```
Commands completed successfully.
```

Query executed successfully.

Step 18: Execute the spDepartmentList1.

```
spDepartmentList1
```



```
--- Parameters with default value
ALTER Procedure spDepartmentList1
@deptt_id       int = 1,
@emp_name       Varchar(100)='Smith'
AS
BEGIN
      select * from employee where deptID=@deptt_id;
      select * from employee where empName = @emp_name
END

spDepartmentList1
```

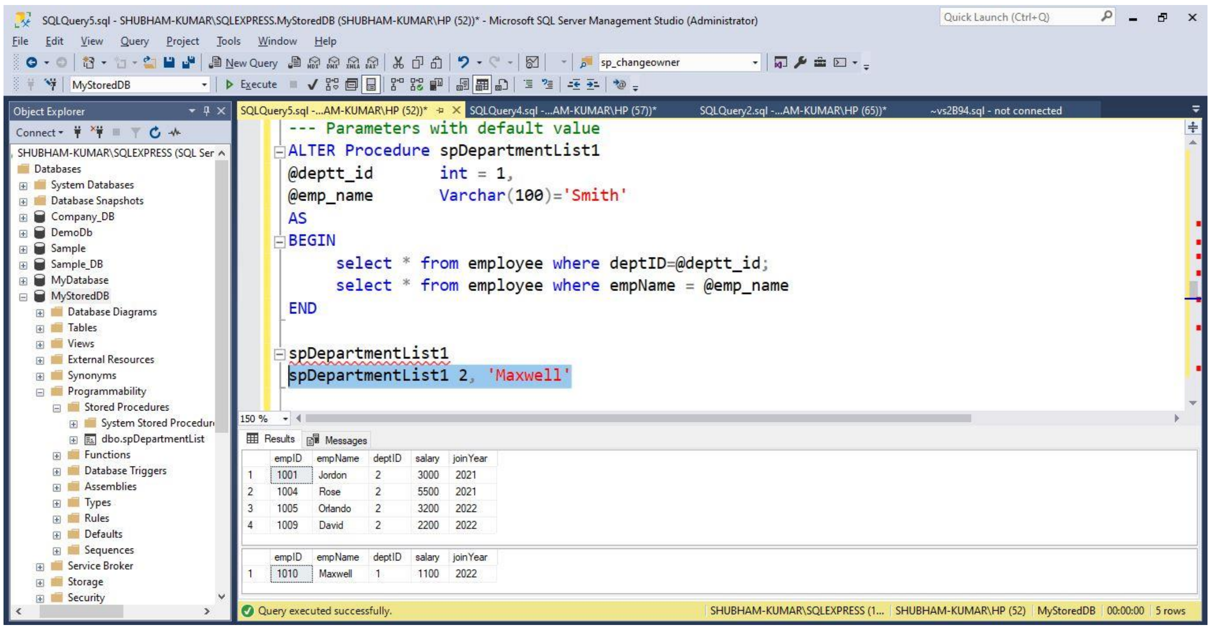
	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020
2	1007	Alice	1	7800	2020
3	1010	Maxwell	1	1100	2022

	empID	empName	deptID	salary	joinYear
1	1002	Smith	1	4000	2020

Query executed successfully.

Step 19: Here the actual value passed will be accepted by the parameter and default value will be discarded.

```
spDepartmentList1 2, 'Maxwell'
```



```
--- Parameters with default value
ALTER Procedure spDepartmentList1
@deptt_id       int = 1,
@emp_name       Varchar(100)='Smith'
AS
BEGIN
      select * from employee where deptID=@deptt_id;
      select * from employee where empName = @emp_name
END

spDepartmentList1
spDepartmentList1 2, 'Maxwell'
```

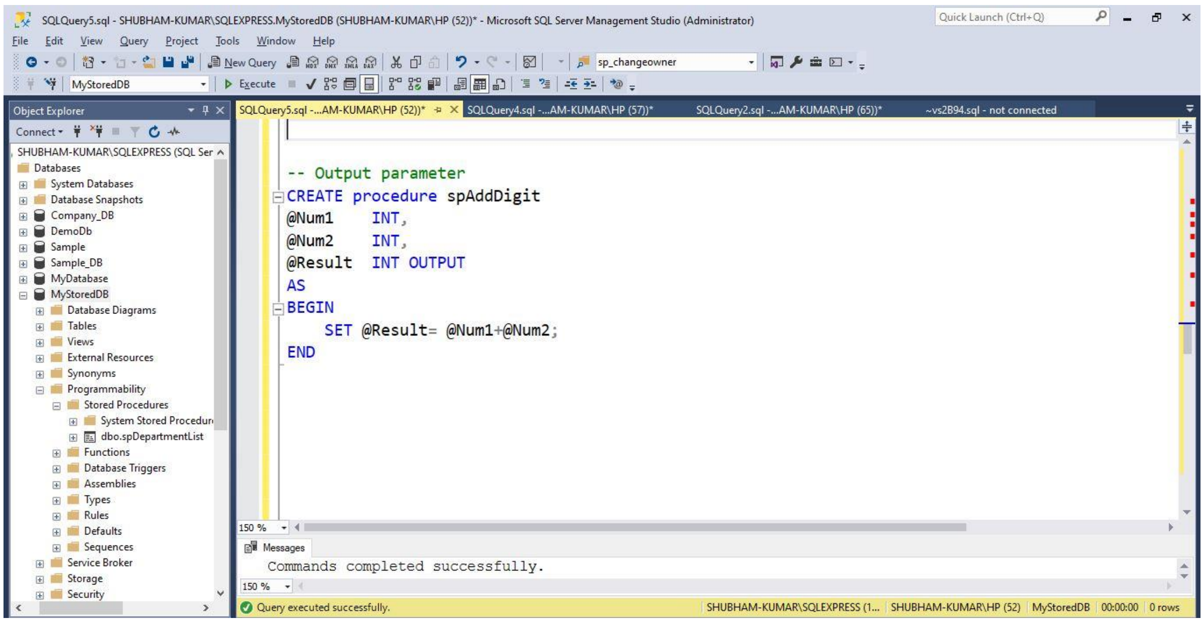
	empID	empName	deptID	salary	joinYear
1	1001	Jordon	2	3000	2021
2	1004	Rose	2	5500	2021
3	1005	Orlando	2	3200	2022
4	1009	David	2	2200	2022

	empID	empName	deptID	salary	joinYear
1	1010	Maxwell	1	1100	2022

```
Query executed successfully.    SHUBHAM-KUMAR\SQLEXPRESS (1... | SHUBHAM-KUMAR\HP (52) | MyStoredDB | 00:00:00 | 5 rows
```

Step 20: Create a procedure **spAddDigit** with parameter to add two digits.

```
CREATE procedure spAddDigit
@Num1     INT,
@Num2     INT,
@Result   INT OUTPUT
AS
BEGIN
    SET @Result= @Num1+@Num2;
END
```



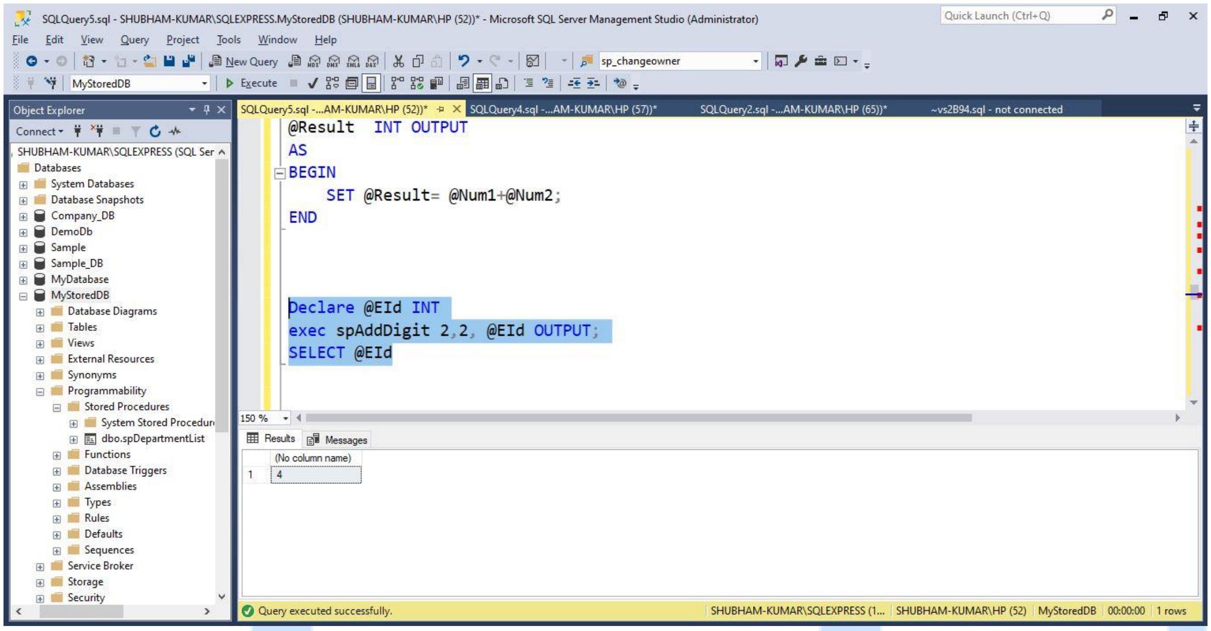
```
-- Output parameter
CREATE procedure spAddDigit
@Num1    INT,
@Num2    INT,
@Result  INT OUTPUT
AS
BEGIN
    SET @Result= @Num1+@Num2;
END
```

```
Commands completed successfully.
```

```
Query executed successfully.    SHUBHAM-KUMAR\SQLEXPRESS (1... | SHUBHAM-KUMAR\HP (52) | MyStoredDB | 00:00:00 | 0 rows
```

Step 21: The following command is used to declare and execute the **spAddDigit** Stored procedure and gives the result set as summation of two numbers entered by the user.

```
Declare @EId INT
exec spAddDigit 2,2, @EId OUTPUT;
SELECT @EId
```



SQLQuery5.sql - SHUBHAM-KUMAR\SQLEXPRESS.MyStoredDB (SHUBHAM-KUMAR\HP (52))* - Microsoft SQL Server Management Studio (Administrator)

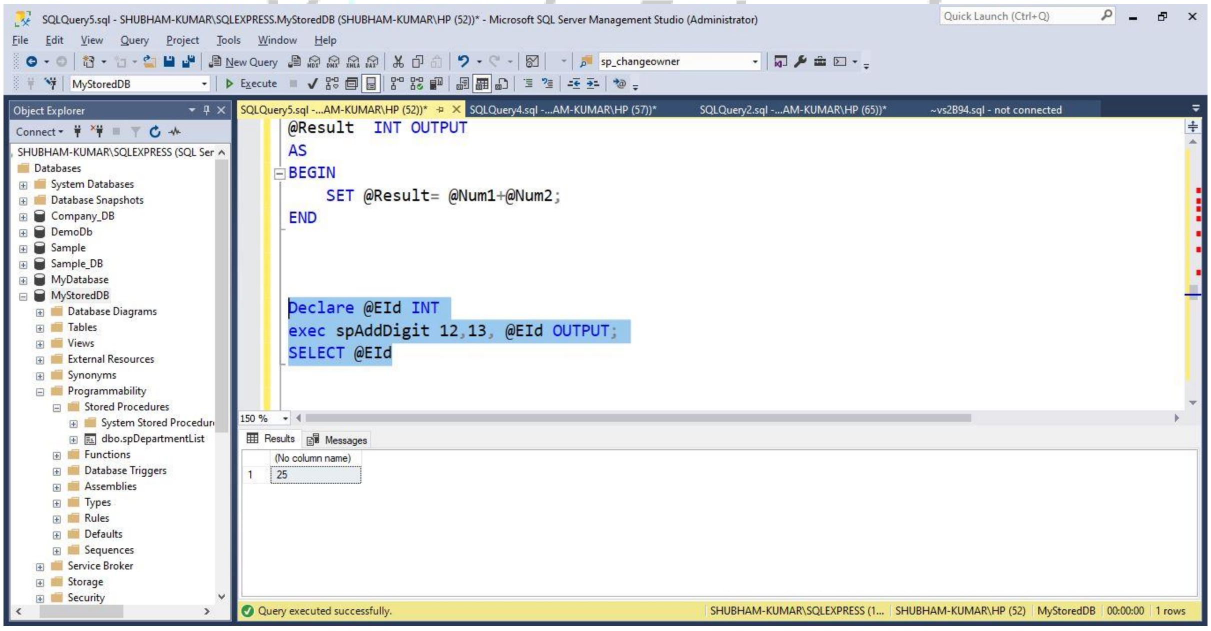
```
@Result  INT OUTPUT
AS
BEGIN
    SET @Result= @Num1+@Num2;
END


Declare @EId INT
exec spAddDigit 2,2, @EId OUTPUT;
SELECT @EId
```

	(No column name)
1	4

Query executed successfully. | SHUBHAM-KUMAR\SQLEXPRESS (1... | SHUBHAM-KUMAR\HP (52) | MyStoredDB | 00:00:00 | 1 rows

Step 22: Check it for another set of numbers.

```
Declare @EId INT
exec spAddDigit 12,13, @EId OUTPUT;
SELECT @Eid
```



SQLQuery5.sql - SHUBHAM-KUMAR\SQLEXPRESS.MyStoredDB (SHUBHAM-KUMAR\HP (52))* - Microsoft SQL Server Management Studio (Administrator)

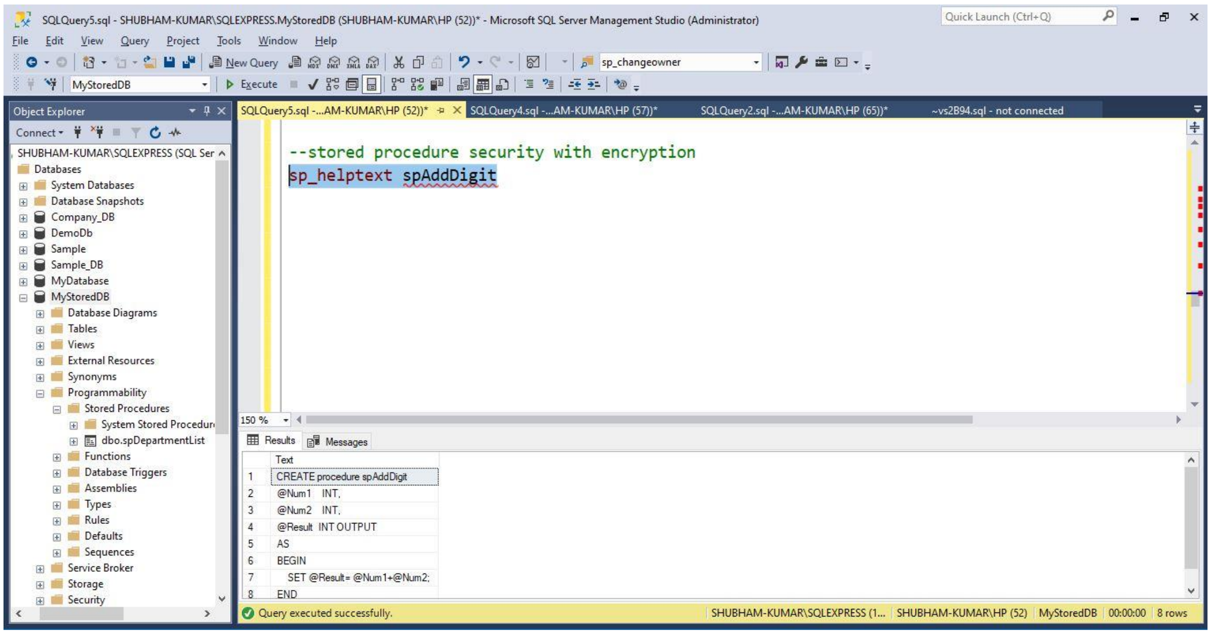
```
@Result  INT OUTPUT
AS
BEGIN
    SET @Result= @Num1+@Num2;
END


Declare @EId INT
exec spAddDigit 12,13, @EId OUTPUT;
SELECT @EId
```

	(No column name)
1	25

Query executed successfully. | SHUBHAM-KUMAR\SQLEXPRESS (1... | SHUBHAM-KUMAR\HP (52) | MyStoredDB | 00:00:00 | 1 rows

Step 23: sp_helptext displays the definition that is used to create an object in multiple rows for the Stored Procedure spAddDigit.

```
sp_helptext spAddDigit
```



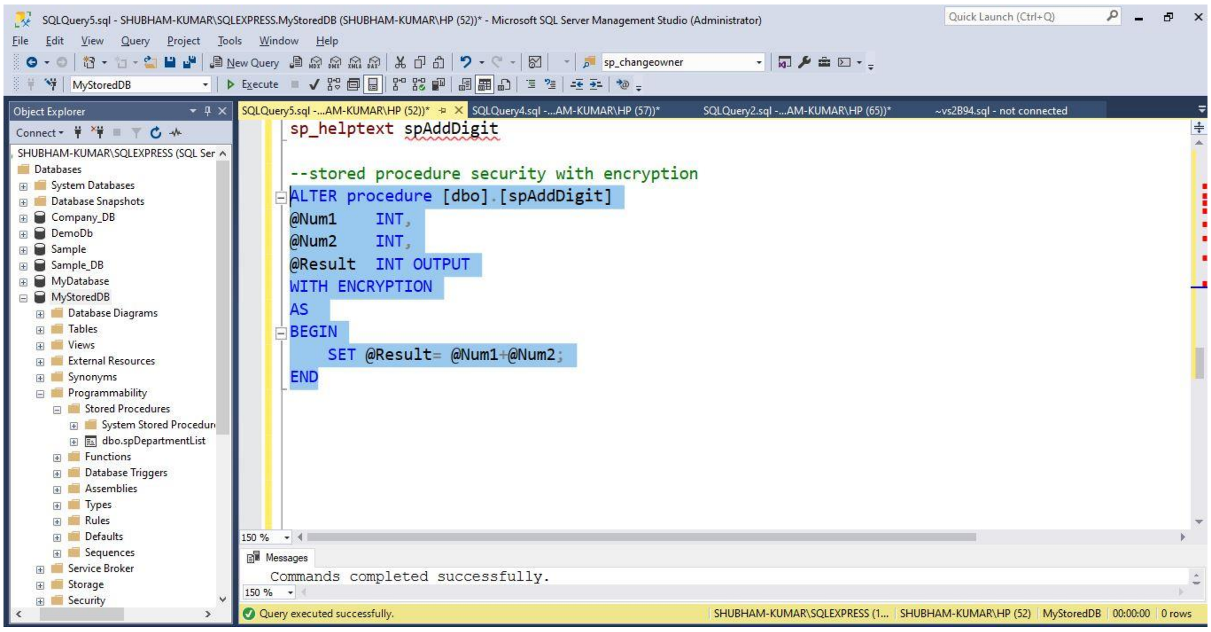
```
--stored procedure security with encryption
sp_helptext spAddDigit
```

	Text
1	CREATE procedure spAddDigit
2	@Num1 INT,
3	@Num2 INT,
4	@Result INT OUTPUT
5	AS
6	BEGIN
7	SET @Result= @Num1+@Num2;
8	END

```
Query executed successfully.   SHUBHAM-KUMAR\SQLEXPRESS (1...   SHUBHAM-KUMAR\HP (52)   MyStoredDB   00:00:00   8 rows
```

Step 24: Encrypting the Stored Procedure.

```
ALTER procedure [dbo].[spAddDigit]
@Num1     INT,
@Num2     INT,
@Result   INT OUTPUT
WITH ENCRYPTION
AS
BEGIN
    SET @Result= @Num1+@Num2;
END
```



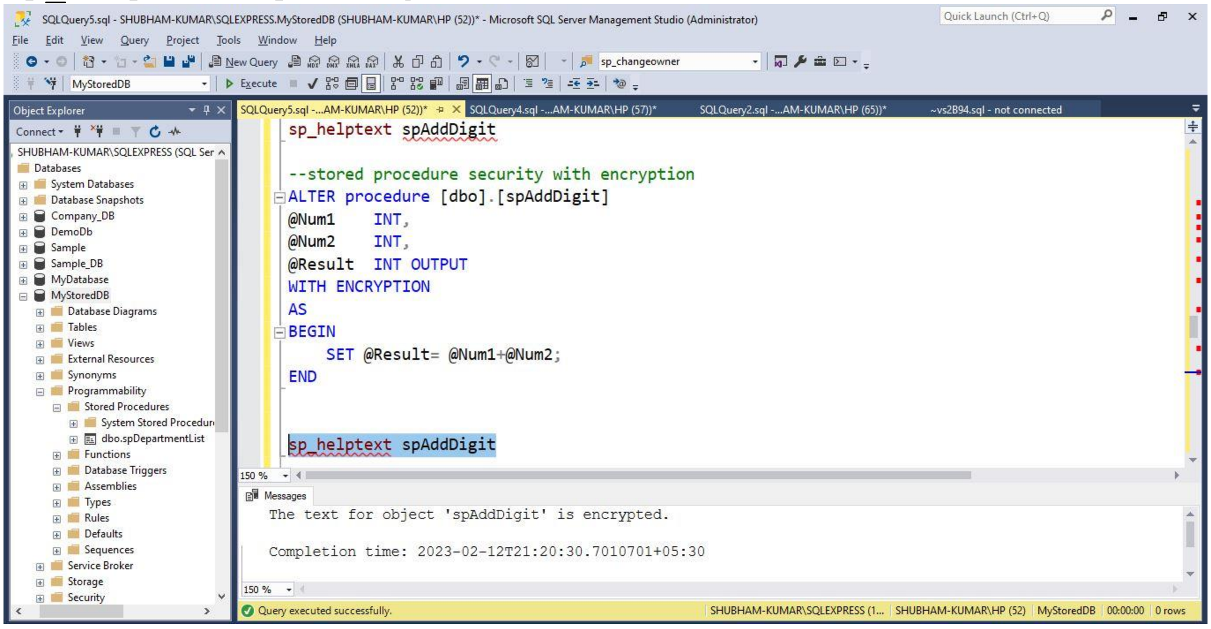
```
sp_helptext spAddDigit

--stored procedure security with encryption
ALTER procedure [dbo].[spAddDigit]
@Num1    INT,
@Num2    INT,
@Result  INT OUTPUT
WITH ENCRYPTION
AS
BEGIN
    SET @Result= @Num1+@Num2;
END
```

```
Commands completed successfully.
```

```
Query executed successfully.   SHUBHAM-KUMAR\SQLEXPRESS (1...   SHUBHAM-KUMAR\HP (52)   MyStoredDB   00:00:00   0 rows
```

Step 25: Executing the encrypted stored procedure.

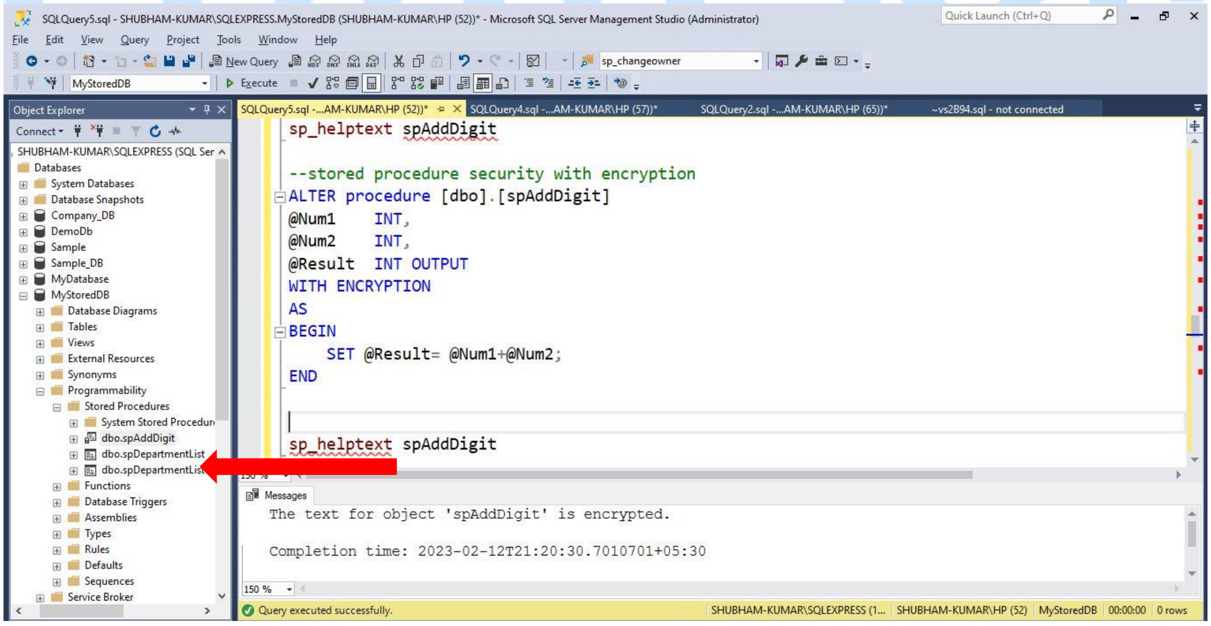
```
sp_helptext spAddDigit
```



```
sp_helptext spAddDigit

--stored procedure security with encryption
ALTER procedure [dbo].[spAddDigit]
@Num1    INT,
@Num2    INT,
@Result  INT OUTPUT
WITH ENCRYPTION
AS
BEGIN
    SET @Result= @Num1+@Num2;
END


sp_helptext spAddDigit
```

```
The text for object 'spAddDigit' is encrypted.

Completion time: 2023-02-12T21:20:30.7010701+05:30
```

Step 26: Hence the Stored procedure **spAddDigit** is encrypted.



```
sp_helptext spAddDigit

--stored procedure security with encryption
ALTER procedure [dbo].[spAddDigit]
@Num1    INT,
@Num2    INT,
@Result  INT OUTPUT
WITH ENCRYPTION
AS
BEGIN
    SET @Result= @Num1+@Num2;
END


sp_helptext spAddDigit
```

```
The text for object 'spAddDigit' is encrypted.

Completion time: 2023-02-12T21:20:30.7010701+05:30
```