

PROCEDURE

- First create procedure than run the procedure by name.

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left lists various system stored procedures under the master database. The central Query Editor window contains the following T-SQL code:

```
[Contact_no.]  
,[Specialist_in]  
,[Salary]  
,[Password]  
FROM [Test DB0].[dbo].[Add_Doctor]  
  
SELECT * FROM Add_doctor;  
  
/*CREATE PROCEDURE spGetDoctor  
AS  
BEGIN  
SELECT [Doctor_name], [Specialist_in] FROM Add_doctor  
END*/  
  
spGetDoctor
```

The Results pane displays the output of the stored procedure, listing nine rows of data:

Doctor_name	Specialist_in
Waniya	child specialist
ALI	child specialist
qrat	child specialist
gigi	child specialist
lady gaga	child specialist
blacke michael	child specialist
Mahnroor	Colon and Rectal Surgeons
Sarah	Critical Care Medicine Specialists
Haniya	Allergists/Immunologists

The Properties pane on the right shows connection details for the current session.

- Create procedure with certain parameters and execute by giving values

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left lists various system stored procedures under the master database. The central Query Editor window contains the following T-SQL code:

```
SELECT [Doctor_name], [Specialist_in] FROM Add_doctor  
END*/  
  
EXEC spGetDoctor  
  
CREATE PROCEDURE spGetDoctorBySpecialistAndSalary  
@specialist_in nvarchar(100),  
@Salary int  
AS  
BEGIN  
SELECT Doctor_name, Specialist_in, Salary FROM Add_doctor  
WHERE Specialist_in = @specialist_in AND Salary = @Salary  
END  
  
EXEC spGetDoctorBySpecialistAndSalary 'child specialist', 40000
```

The Results pane displays the output of the stored procedure, listing seven rows of data:

Doctor_name	Specialist_in	Salary
Waniya	child specialist	40000
ALI	child specialist	40000
qrat	child specialist	40000
gigi	child specialist	40000
lady gaga	child specialist	40000
blacke michael	child specialist	40000
Waniya	child specialist	40000

The Properties pane on the right shows connection details for the current session.

- Using built-in procedure to get stored procedure

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the master database is selected. In the center pane, a query window titled 'SQLQuery1.sql - DE...ousuf TRaders (52)*' contains the following T-SQL code:

```

END*/
Exec spGetDoctor
CREATE PROCEDURE spGetDoctorBySpecialistAndSalary
@specialist_in nvarchar(100),
@Salary int
AS
BEGIN
    SELECT Doctor_name, Specialist_in, Salary FROM Add_doctor
    WHERE Specialist_in = @specialist_in AND Salary = @Salary
END
EXEC spGetDoctorBySpecialistAndSalary 'child specialist', 40000
sp_helptext spGetDoctor

```

The 'Properties' window on the right displays connection details for the current session. The 'Results' pane shows the execution message: 'Query executed successfully.' and the number of rows returned: 6.

- Altering Procedure table, for using ORDER BY statement

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, the master database is selected. In the center pane, a query window titled 'SQLQuery1.sql - DE...ousuf TRaders (52)*' contains the following T-SQL code:

```

SELECT Doctor_name, Specialist_in, Salary FROM Add_doctor
WHERE Specialist_in = @specialist_in AND Salary = @Salary
EXEC spGetDoctorBySpecialistAndSalary 'child specialist', 40000
sp_helptext spGetDoctor
ALTER PROCEDURE spGetDoctor
AS
BEGIN
    SELECT [Doctor_name], [Specialist_in] FROM Add_doctor ORDER BY Doctor_name
END
spGetDoctor

```

The 'Properties' window on the right displays connection details for the current session. The 'Results' pane shows the execution message: 'Query executed successfully.' and the number of rows returned: 11. The results table lists 11 rows of data:

Doctor_name	Specialist_in
ALI	child specialist
blake michael	child specialist
gigi	child specialist
Haniya	Allergists/Immunologists
lady gaga	child specialist
Mahnood	Colon and Rectal Surgeons
qirat	child specialist
Sarah	Critical Care Medicine Specialists
Wanya	child specialist
Wanya	child specialist
Zamin	Family Physicians

- Drop the following Procedure

The screenshot shows the Microsoft SQL Server Management Studio interface. The query window displays the following T-SQL code:

```

-- EXEC spGetDoctorBySpecialistAndSalary 'child specialist', 40000
sp_helptext spGetDoctor
ALTER PROCEDURE spGetDoctor
AS
BEGIN
    SELECT [Doctor_name], [Specialist_in] FROM Add_doctor ORDER BY Doctor_name
END

DROP PROCEDURE spGetDoctor

```

The 'Messages' pane shows the output: "Command(s) completed successfully." The 'Properties' pane on the right provides connection details.

- Encrypted the other Procedure (means text not visible)

The screenshot shows the Microsoft SQL Server Management Studio interface. The query window displays the following T-SQL code, which is partially obscured by redaction:

```

SELECT * FROM Add_doctor;
--CREATE PROCEDURE spGetDoctor
AS
BEGIN
    SELECT [Doctor_name], [Specialist_in] FROM Add_doctor
END;
EXEC spGetDoctor;
--ALTER PROCEDURE spGetDoctorBySpecialistAndSalary
--@specialist_in nvarchar(100),
--@Salary int
--WITH ENCRYPTION
AS
BEGIN
    SELECT Doctor_name, Specialist_in, Salary FROM Add_doctor
    WHERE Specialist_in = @specialist_in AND Salary = @Salary
END

```

The 'Messages' pane shows the output: "Command(s) completed successfully." The 'Properties' pane on the right provides connection details.

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The main window displays a T-SQL script for creating a stored procedure named spGetDoctorBySpecialistAndSalary. The script includes a comment block, an exec statement, and an alter statement. The alter statement creates a new version of the procedure with an encryption clause. A message in the messages pane states that the text for the object is encrypted. The properties pane on the right shows connection parameters like connection name, state, and SPID. The status bar at the bottom right shows the date and time as 15-Jul-20 3:39:06 PM.

- Create procedure and using OUTPUT Parameter

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The main window displays a T-SQL script for a stored procedure named spGetDoctorCount. The script uses dynamic SQL to count doctors by specialist. The 'Messages' pane shows the output of the execution, indicating that the variable @TotalCount is not null. The 'Properties' pane on the right shows connection parameters like Name, Start time, and State. The 'Output' pane at the bottom shows the execution status as successful.

```
spGetDoctorCount
DROP PROCEDURE spGetDoctorCount
CREATE PROCEDURE spGetDoctorCountBySpecialist
    @Specialist nvarchar(100),
    @DoctorCount int OUTPUT
AS
BEGIN
    SELECT @DoctorCount = COUNT(Doctor_ID) FROM Add_doctor WHERE Specialist_in = @Specialist
END

DECLARE @TotalCount int
EXEC spGetDoctorCountBySpecialist 'child specialist', @TotalCount out
If(@TotalCount IS NULL)
Print '#TotalCount is null'
else
Print '#TotalCount is not null'
print @TotalCount
```

Messages
@TotalCount is not null

Properties

Current connection parameters	
Name	DESKTOP-EN53T3S\SQLEXPRESS
Aggregate Status	
Connection failures	0
Elapsed time	00:00:00.022
Finish time	15-Jul-20 4:12:08 PM
Name	DESKTOP-EN53T3S\SQLEXPRESS
Rows returned	0
Start time	15-Jul-20 4:12:08 PM
State	Open
Connection	
Connection name	DESKTOP-EN53T3S\SQLEXPRESS
Connection Details	
Connection elapsed t	00:00:00.022
Connection finish tim	15-Jul-20 4:12:08 PM
Connection rows retu	0
Connection start time	15-Jul-20 4:12:08 PM
Connection state	Open
Display name	DESKTOP-EN53T3S\SQLEXPRESS
Login name	DESKTOP-EN53T3S\Yousuf
Server name	DESKTOP-EN53T3S\SQLEXPRESS
Server version	12.0.2269
Session Tracing ID	
SPID	52

Utility Explorer Object Explorer

Query executed successfully.

DESKTOP-EN53T3S\SQLEXPRESS... DESKTOP-EN53T3S\Yousuf... master | 00:00:00 | 0 rows

Output

Show output from:

Task List Output Call Hierarchy

Matches: BEGIN

Ln 64 Col 20 Ch 20

4:12 PM 15-Jul-20

- using [sp_depends]

SQLQuery1.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (52)) - Microsoft SQL Server Management Studio

```

USE [master]
EXEC spGetDoctorCountBySpecialist 'child specialist', @TotalCount out
If (@TotalCount IS NULL)
    Print '@TotalCount is null'
else
    Print '@TotalCount is not null'
print @TotalCount
sp_help spGetDoctorCountBySpecialist
sp_depends spGetDoctorCountBySpecialist

```

Results

name	type	updated	selected	column
dbo.Add_doctor	user table	no	yes	Doctor_ID
dbo.Add_doctor	user table	no	yes	Specialist_in

Properties

Current connection parameters

Aggregate Status

Connection failures

Elapsed time 00:00:00.532

Finish time 15-Jul-20 4:15:20 PM

Name DESKTOP-EN53T3S\SQLEX

Rows returned 2

Start time 15-Jul-20 4:15:19 PM

State Open

Connection

Connection name DESKTOP-EN53T3S\SQLEX

Connection Details

Connection elapsed 0:00:00.532

Connection finish tim 15-Jul-20 4:15:20 PM

Connection rows retu 2

Connection start time 15-Jul-20 4:15:19 PM

Connection state Open

Display name DESKTOP-EN53T3S\SQLEX

Login name DESKTOP-EN53T3S\Yousuf

Server name DESKTOP-EN53T3S\SQLEX

Server version 12.0.2269

Session Tracing ID

SPID 52

Activate Windows

Name The name of the connection: Windows

- create procedure to count certain parameter using OUTPUT

SQLQuery1.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (52)) - Microsoft SQL Server Management Studio

```

CREATE PROCEDURE spGetDoctorByCounting
@TotalCount int output
AS
BEGIN
    SELECT @TotalCounting = COUNT(Doctor_ID) FROM Add_doctor
END

DECLARE @TotalDoctors int
EXEC spGetDoctorByCounting @TotalDoctors output
SELECT @TotalDoctors

```

Results

(No column name)
11

Properties

Current connection parameters

Aggregate Status

Connection failures

Elapsed time 00:00:00.031

Finish time 15-Jul-20 5:15:03 PM

Name DESKTOP-EN53T3S\SQLEX

Rows returned 1

Start time 15-Jul-20 5:15:03 PM

State Open

Connection

Connection name DESKTOP-EN53T3S\SQLEX

Connection Details

Connection elapsed 0:00:00.031

Connection finish tim 15-Jul-20 5:15:03 PM

Connection rows retu 1

Connection start time 15-Jul-20 5:15:03 PM

Connection state Open

Display name DESKTOP-EN53T3S\SQLEX

Login name DESKTOP-EN53T3S\Yousuf

Server name DESKTOP-EN53T3S\SQLEX

Server version 12.0.2269

Session Tracing ID

SPID 52

Activate Windows

Name The name of the connection: Windows

- Using RETURN value not using OUTPUT

```

SQLQuery1.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (52)) - Microsoft SQL Server Management Studio
File Edit View Query Project Debug Tools Window Help
master Execute Debug
Object Explorer
SQLQuery1.sql - DESKTOP-EN53T3S\SQLEXPRESS (SQL Server)
Databases
System Databases
AttendanceDB
educators
Hospital_mang
Test DB0
Database Diagrams
Tables
Views
Synonyms
Programmability
Service Broker
Storage
Security
Server Objects
Replication
Management
Results Messages
(No column name)
1 11
Properties
Current connection parameters
Aggregate Status
Connection failures
Elapsed time 00:00:00.100
Finish time 15-Jul-20 5:20:04 PM
Name DESKTOP-EN53T3S\SQLEXP
Rows returned 1
Start time 15-Jul-20 5:20:04 PM
State Open
Connection
Connection name DESKTOP-EN53T3S\SQLEXP
Connection Details
Connection elapsed t 0:00:00.100
Connection finish time 15-Jul-20 5:20:04 PM
Connection rows retu 1
Connection start time 15-Jul-20 5:20:04 PM
Connection state Open
Display name DESKTOP-EN53T3S\SQLEXP
Login name DESKTOP-EN53T3S\Yousuf
Server name DESKTOP-EN53T3S\SQLEXP
Server version 12.0.2269
Session Tracing ID
SPID 52
Output
Show output from:
Task List Output Call Hierarchy
Matches: BEGIN
Type here to search

```

- Create procedure to call by ID

```

SQLQuery1.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (52)) - Microsoft SQL Server Management Studio
File Edit View Query Project Debug Tools Window Help
master Execute Debug
Object Explorer
SQLQuery1.sql - DESKTOP-EN53T3S\SQLEXPRESS (SQL Server)
Databases
System Databases
AttendanceDB
educators
Hospital_mang
Test DB0
Database Diagrams
Tables
Views
Synonyms
Programmability
Service Broker
Storage
Security
Server Objects
Replication
Management
Results Messages
Doctor_name
1 Waniya
Properties
Current connection parameters
Aggregate Status
Connection failures
Elapsed time 00:00:00.037
Finish time 15-Jul-20 5:27:32 PM
Name DESKTOP-EN53T3S\SQLEXP
Rows returned 1
Start time 15-Jul-20 5:27:32 PM
State Open
Connection
Connection name DESKTOP-EN53T3S\SQLEXP
Connection Details
Connection elapsed t 0:00:00:037
Connection finish time 15-Jul-20 5:27:32 PM
Connection rows retu 1
Connection start time 15-Jul-20 5:27:32 PM
Connection state Open
Display name DESKTOP-EN53T3S\SQLEXP
Login name DESKTOP-EN53T3S\Yousuf
Server name DESKTOP-EN53T3S\SQLEXP
Server version 12.0.2269
Session Tracing ID
SPID 52
Output
Show output from:
Task List Output Call Hierarchy
Matches: BEGIN
Type here to search

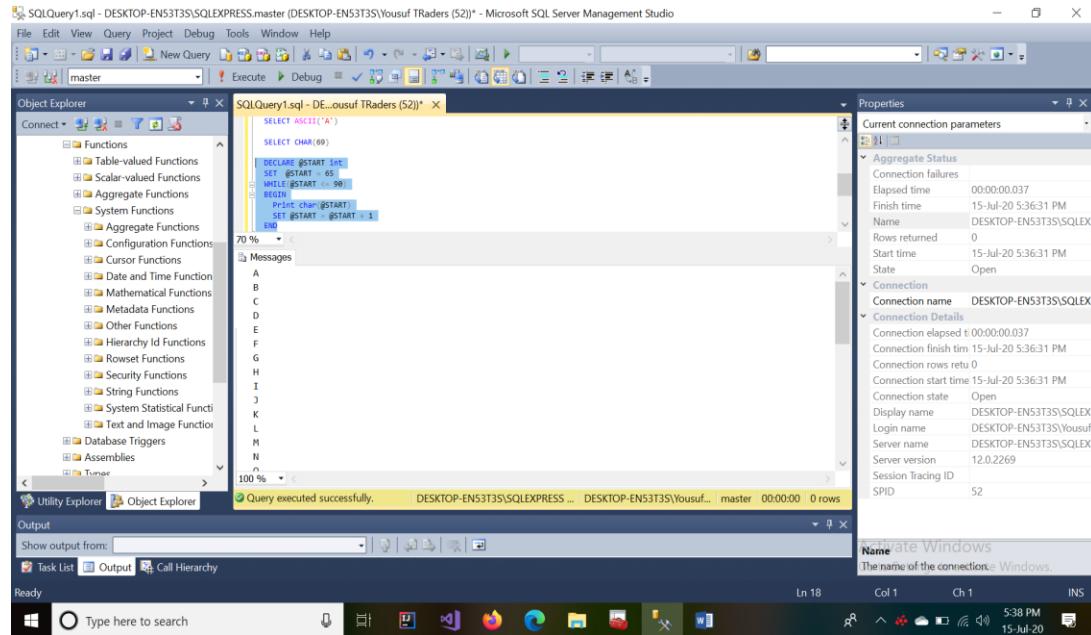
```

ADVANTAGES OF PROCEDURE

- Execution reusability
 - Reduce network traffic
 - Code reusability and better maintenance
 - Better security
 - Avoid SQL injection attack
-
-

FUNCTIONS

- Using built in Function



The screenshot shows the Microsoft SQL Server Management Studio interface. In the center, there is a query window titled "SQLQuery1.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf Traders (52))". The query window contains the following T-SQL code:

```
SELECT ASCII('A')  
      SELECT CHAR(99)  
      DECLARE @START INT  
      SET @START = 65  
      WHILE (@START <= 90)  
      BEGIN  
          PRINT CHAR(@START)  
          SET @START = @START + 1  
      END
```

The "Messages" pane below the code shows the output of the query: "A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N", "O". The status bar at the bottom indicates "Query executed successfully." To the right of the query window is a "Properties" pane displaying connection parameters and details about the current session.

- Create functions with condition

The screenshot shows the Microsoft SQL Server Management Studio interface. In the center, there is a query window titled "SQLQuery2.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (53))". The code in the window is:

```

DECLARE @DOB DATE
DECLARE @Age INT
SET @DOB = '2000-2-2'

SET @Age = DATEDIFF (YEAR, @DOB, GETDATE())
CASE
    WHEN (MONTH (@DOB) > MONTH (GETDATE())) OR
        (MONTH (@DOB) = MONTH (GETDATE())) AND DAY (@DOB) > DAY (GETDATE())
    THEN 1
    ELSE 0
END

SELECT @Age

```

The "Messages" pane shows the result: "1". The status bar at the bottom indicates "Query executed successfully." and "1 rows". To the right of the query window is the "Properties" pane, which displays connection details for the current session.

- Scalar Function

The screenshot shows the Microsoft SQL Server Management Studio interface. In the center, there is a query window titled "SQLQuery2.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (53))". The code in the window is:

```

CREATE FUNCTION CalculateAge(@DOB DATE)
RETURNS INT
AS
BEGIN
    DECLARE @Age INT

    SET @Age = DATEDIFF (YEAR, @DOB, GETDATE())
    CASE
        WHEN (MONTH (@DOB) > MONTH (GETDATE())) OR
            (MONTH (@DOB) = MONTH (GETDATE())) AND DAY (@DOB) > DAY (GETDATE())
        THEN 1
        ELSE 0
    END

    RETURN @Age
END

```

The "Messages" pane shows the result: "Command(s) completed successfully.". The status bar at the bottom indicates "Query executed successfully." and "0 rows". To the right of the query window is the "Properties" pane, which displays connection details for the current session.

- Execution of other table

SQLQuery2.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (53)) - Microsoft SQL Server Management Studio

File Edit View Query Project Debug Tools Window Help

master Execute Debug

Object Explorer

Databases System Databases AttendanceDB educators Hospital_mang Test DBO Database Diagrams Tables Views Synonyms Programmability Stored Procedures Functions Table-valued Functions Scalar-valued Functions Aggregate Functions System Functions Database Triggers Assemblies Types Roles

SQLQuery2.sql - DESKTOP-EN53T3S\Yousuf TRaders (53)* DESKTOP-EN53T3S\dbo.Doctor_Entry

```
***** Script for SelectTopNRows command from SSMS *****
SELECT TOP 1000 [ID]
,[Dr. Name]
,[Joining Date]
FROM [Test DBO].[dbo].[Doctor_Entry]

SELECT *FROM [Test DBO].[dbo].[Doctor_Entry]

SELECT dbo.CalculateAge('2000-2-2')
```

Results Messages

(No column name)
1 20

Properties

Current connection parameters

Aggregate Status

Connection failures

Elapsed time 00:00:03.677

Finish time 15-Jul-20 6:32:40 PM

Name DESKTOP-EN53T3S\SQLEXPRESS

Rows returned 1

Start time 15-Jul-20 6:32:40 PM

State Open

Connection

Connection name DESKTOP-EN53T3S\SQLEXPRESS

Connection Details

Connection elapsed t:00:00:03.677

Connection finish tim 15-Jul-20 6:32:40 PM

Connection rows retu 1

Connection start time 15-Jul-20 6:32:40 PM

Connection state Open

Display name DESKTOP-EN53T3S\SQLEXPRESS

Login name DESKTOP-EN53T3S\Yousuf

Server name DESKTOP-EN53T3S\SQLEXPRESS

Server version 12.0.2269

Session Tracing ID 53

SPID 53

Output

Show output from:

Task List Output Call Hierarchy

Ready

Type here to search

Ln 14 Col 7 Ch 7 INS

6:33 PM 15-Jul-20

Name Windows

The name of the connection is Windows.

- Using Function with other parameter with certain condition

SQLQuery2.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (53)) - Microsoft SQL Server Management Studio

File Edit View Query Project Debug Tools Window Help

master Execute Debug

Object Explorer

Databases System Databases AttendanceDB educators Hospital_mang Test DBO Database Diagrams Tables Views Synonyms Programmability Stored Procedures Functions Table-valued Functions Scalar-valued Functions Aggregate Functions System Functions Database Triggers Assemblies Types Roles

SQLQuery2.sql - DESKTOP-EN53T3S\Yousuf TRaders (53)* DESKTOP-EN53T3S\dbo.Doctor_Entry

```
***** Script for SelectTopNRows command from SSMS *****
SELECT TOP 1000 [ID]
,[Dr. Name]
,[Joining Date]
FROM [Test DBO].[dbo].[Doctor_Entry]

SELECT *FROM [Test DBO].[dbo].[Doctor_Entry]

SELECT dbo.CalculateAge('2000-2-2') AS AGE
```

```
= SELECT ID ,[Dr. Name] ,dbo.CalculateAge([Joining Date]) AS DURATION FROM [Test DBO].[dbo].[Doctor_Entry]
WHERE dbo.CalculateAge([Joining Date]) < 30
```

Results Messages

ID	Dr. Name	DURATION
1	Zavyar	5
2	Neha	13
3	Rabya	11
4	Sarah	19
5	Shahid	17
6	Afridi	17
7	Wajha	12
8	Hans	0
9	Yumna	1

Properties

Current connection parameters

Aggregate Status

Connection failures

Elapsed time 00:00:11.5

Finish time 15-Jul-20 6:40:12 PM

Name DESKTOP-EN53T3S\SQLEXPRESS

Rows returned 9

Start time 15-Jul-20 6:40:12 PM

State Open

Connection

Connection name DESKTOP-EN53T3S\SQLEXPRESS

Connection Details

Connection elapsed t:00:00:11.5

Connection finish tim 15-Jul-20 6:40:12 PM

Connection rows retu 9

Connection start time 15-Jul-20 6:40:12 PM

Connection state Open

Display name DESKTOP-EN53T3S\SQLEXPRESS

Login name DESKTOP-EN53T3S\Yousuf

Server name DESKTOP-EN53T3S\SQLEXPRESS

Server version 12.0.2269

Session Tracing ID 53

SPID 53

Output

Show output from:

Task List Output Call Hierarchy

Ready

Type here to search

Ln 11 Col 1 Ch 1 INS

6:40 PM 15-Jul-20

Name Windows

The name of the connection is Windows.

- Table Function

The screenshot shows the Microsoft SQL Server Management Studio interface. A query window titled "SQLQuery3.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (55))" displays the following T-SQL code:

```

CREATE FUNCTION fn_DoctorByGender(@sex nvarchar(10))
RETURNS TABLE
AS
RETURN (select ID, [Dr. Name], [Joining Date] AS Duration, Gender
       FROM [Test DB0].[dbo].[Doctor_Entry]
      where Gender = @sex)

SELECT *FROM fn_DoctorByGender('female')
    
```

The results pane shows a table with columns ID, Dr. Name, Duration, and Gender. The data is as follows:

ID	Dr. Name	Duration	Gender
1	Neha	2007-02-17	female
2	Rabya	2009-02-28	female
3	Sarah	2009-10-10	female
4	Wajha	2008-05-06	female
5	Yumna	2019-06-18	female

The status bar at the bottom right indicates "Query executed successfully." and "15-Jul-20 6:55:31 PM".

- Multiline function

The screenshot shows the Microsoft SQL Server Management Studio interface. A query window titled "SQLQuery3.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (55))" displays the following T-SQL code:

```

SELECT *FROM [Test DB0].[dbo].[Doctor_Entry]

CREATE FUNCTION fn_DoctorByGender(@sex nvarchar(10))
RETURNS TABLE
AS
RETURN (select ID, [Dr. Name], [Joining Date] AS Duration, Gender
       FROM [Test DB0].[dbo].[Doctor_Entry]
      where Gender = @sex)

SELECT *FROM fn_DoctorByGender('female')

select [Dr.Name], Gender, [Specialist in]
  from fn_DoctorByGender('female') E
  join Add_doctor D ON D.Doctor_ID = E.ID
    
```

The status bar at the bottom right indicates "Query completed with errors." and "15-Jul-20 7:04:22 PM".

TRIGGER

- Insert

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, a database named 'master' is selected. In the center pane, a query window displays the creation of a trigger:

```

CREATE TRIGGER tr_AddDocInsert
on Add_doctor
for insert
as
begin
    --select * from inserted
    declare @id int
    select @id = Doctor_ID from inserted

    insert into [test DBO].[dbo].[Doctor_info]
    values ('New Doctor with ID = '+cast(@id as varchar(5)) +' is added at '+cast(GETDATE() as varchar(20)))
end
sp_helptext tr_AddDocInsert

```

The message pane shows: "Query executed successfully." The Properties pane on the right provides connection details.

The screenshot shows the Microsoft SQL Server Management Studio interface. In the Object Explorer, a database named 'DESKTOP-EN53T3S\SQLEXPRESS (SQL)' is selected. In the center pane, a query window displays the creation of a trigger, but it ends with an error:

```

CREATE TRIGGER tr_AddDocInsert
on Add_doctor
for insert
as
begin
    --select * from inserted
    declare @id int
    select @id = Doctor_ID from inserted

    insert into [test DBO].[dbo].[Doctor_info]
    values ('New Doctor with ID = '+cast(@id as varchar(5)) +' is added at '+cast(GETDATE() as varchar(20)))
end
insert into Add_doctor values(13, 'NAIRA', 'Nagan chow.', 'n1999@yahoo.com', '789802099-89', 'Gynaecologist', 2000000, 'qerty')

```

The message pane shows: "Query completed with errors." The Properties pane on the right provides connection details.

- Update

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including tables like 'dbo.Add_Doctor' and 'dbo.Add_Patient'. The central query editor displays the following T-SQL code:

```

select @Id = Doctor_ID from deleted
insert into [test DBO].[dbo].[Doctor_Info](ID, [Doctor_Data])
values (1, 'New Doctor with ID = ' + cast(@Id as mvarchar(5)) + ' is added at ' + cast(GETDATE() as mvarchar(20)))
end

CREATE TRIGGER tr_AddDocUpdate
on Add_doctor
for UPDATE
as
begin
    select * from deleted
    select * from inserted
end

```

The status bar at the bottom indicates "Query executed successfully." and "0 rows". The Properties pane on the right shows connection details for the session.

- Delete

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including tables like 'dbo.Add_Doctor' and 'dbo.Add_Patient'. The central query editor displays the following T-SQL code:

```

declare @Id int
select @Id = Doctor_ID from inserted
insert into [test DBO].[dbo].[Doctor_Info](ID, [Doctor_Data])
values (1, 'New Doctor with ID = ' + cast(@Id as mvarchar(5)) + ' is added at ' + cast(GETDATE() as mvarchar(20)))
end

sp_helptext tr_AddDocInsert

insert into Add_doctor values(13, 'NAIRA', 'Nagan chow.', 'n1999@yahoo.com', '789002099-89', 'Gasenacologist', 2000000, 'query')

CREATE TRIGGER tr_AddDocDelete
on Add_doctor
for DELETE
as
begin
    select * from deleted
    declare @Id int
    select @Id = Doctor_ID from deleted
    insert into [test DBO].[dbo].[Doctor_Info](ID, [Doctor_Data])
    values (1, 'New Doctor with ID = ' + cast(@Id as mvarchar(5)) + ' is added at ' + cast(GETDATE() as mvarchar(20)))
end

```

The status bar at the bottom indicates "Query executed successfully." and "0 rows". The Properties pane on the right shows connection details for the session.

SQLQuery5.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (53)) - Microsoft SQL Server Management Studio

File Edit View Query Project Debug Tools Window Help

New Query Execute Debug

master Properties

Object Explorer

SQLQuery7.sql - DE...ousuf TRaders (54) SQLQuery1.sql - not connected SQLQuery5.sql - DE...ousuf TRaders (53)*

```

CREATE TRIGGER tr_AddDocDelete
on Add_doctor
for DELETE
as
begin
--select *from inserted
declare @Id int
select @Id = Doctor_ID from deleted

insert into [test DB0].[dbo].[Doctor_Info](ID, [Doctor_Data])
values (1,'New Doctor with ID = '+cast(@Id as varchar(5)) +' is added at '+cast(GETDATE() as varchar(20)))
end

```

Results Messages

ID Doctor Data

Query executed successfully.

DESKTOP-EN53T3S\SQLEXPRESS ... DESKTOP-EN53T3S\Yousuf... master 00:00:00 0 rows

Properties

Current connection parameters

Aggregate Status

Connection failures

Elapsed time 00:00:00.031

Finish time 15-Jul-20 8:53:46 PM

Name DESKTOP-EN53T3S\SQLEX

Rows returned 0

Start time 15-Jul-20 8:53:46 PM

State Open

Connection

Connection name DESKTOP-EN53T3S\SQLEX

Connection Details

Connection elapsed t 00:00:00.031

Connection finish tim 15-Jul-20 8:53:46 PM

Connection rows retu 0

Connection start time 15-Jul-20 8:53:46 PM

Connection state Open

Display name DESKTOP-EN53T3S\SQLEX

Login name DESKTOP-EN53T3S\Yousuf

Server name DESKTOP-EN53T3S\SQLEX

Server version 12.0.2269

Session Tracing ID 53

SPID 53

Name Activate Windows

The name of the connection. Windows.

Output

Show output from:

Task List Output Call Hierarchy

Ready

Type here to search

Ln 12 Col 1 Ch 1 8:53 PM 15-Jul-20

- Instead of

SQLQuery8.sql - DESKTOP-EN53T3S\SQLEXPRESS.master (DESKTOP-EN53T3S\Yousuf TRaders (55)) - Microsoft SQL Server Management Studio

File Edit View Query Project Debug Tools Window Help

New Query Execute Debug

master Properties

Object Explorer

SQLQuery5.sql - DE...ousuf TRaders (53)* SQLQuery8.sql - DE...ousuf TRaders (55)*

```

CREATE TRIGGER tr_AddEntryInsert
on [test DB0].[dbo].[Doctor_Entry]
Instead of insert
as
begin
select *from inserted
select *from deleted
end

```

Messages

Msg 2108, Level 15, State 1, Procedure tr_AddEntryInsert, Line 10

Query completed with errors.

DESKTOP-EN53T3S\SQLEXPRESS ... DESKTOP-EN53T3S\Yousuf... master 00:00:00 0 rows

Properties

Current connection parameters

Aggregate Status

Connection failures

Elapsed time 00:00:00

Finish time 15-Jul-20 9:22:44 PM

Name DESKTOP-EN53T3S\SQLEX

Rows returned 0

Start time 15-Jul-20 9:22:44 PM

State Open

Connection

Connection name DESKTOP-EN53T3S\SQLEX

Connection Details

Connection elapsed t 00:00:00

Connection finish tim 15-Jul-20 9:22:44 PM

Connection rows retu 0

Connection start time 15-Jul-20 9:22:44 PM

Connection state Open

Display name DESKTOP-EN53T3S\SQLEX

Login name DESKTOP-EN53T3S\Yousuf

Server name DESKTOP-EN53T3S\SQLEX

Server version 12.0.2269

Session Tracing ID 55

Name Activate Windows

The name of the connection. Windows.

Output

Show output from:

Task List Output Call Hierarchy

Ready

Type here to search

Ln 16 Col 25 Ch 22 9:24 PM 15-Jul-20