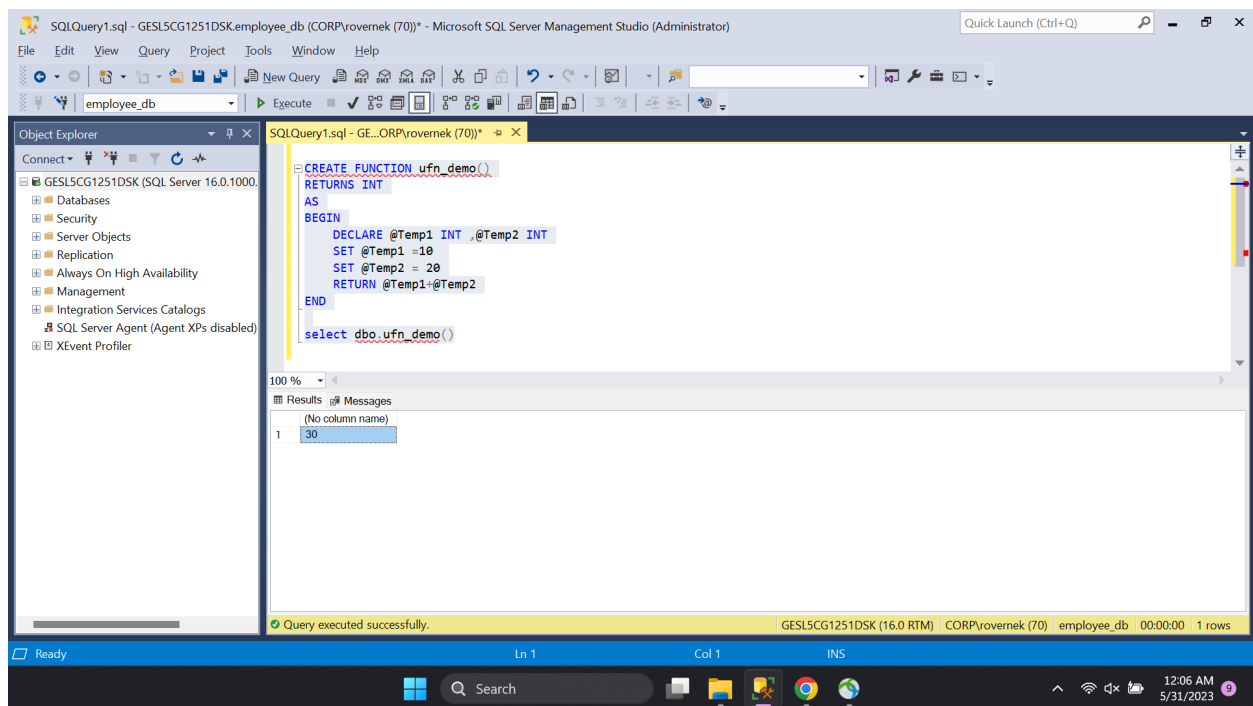


Demonstrate the following:

1) A Table Variable can be used inside a Function.

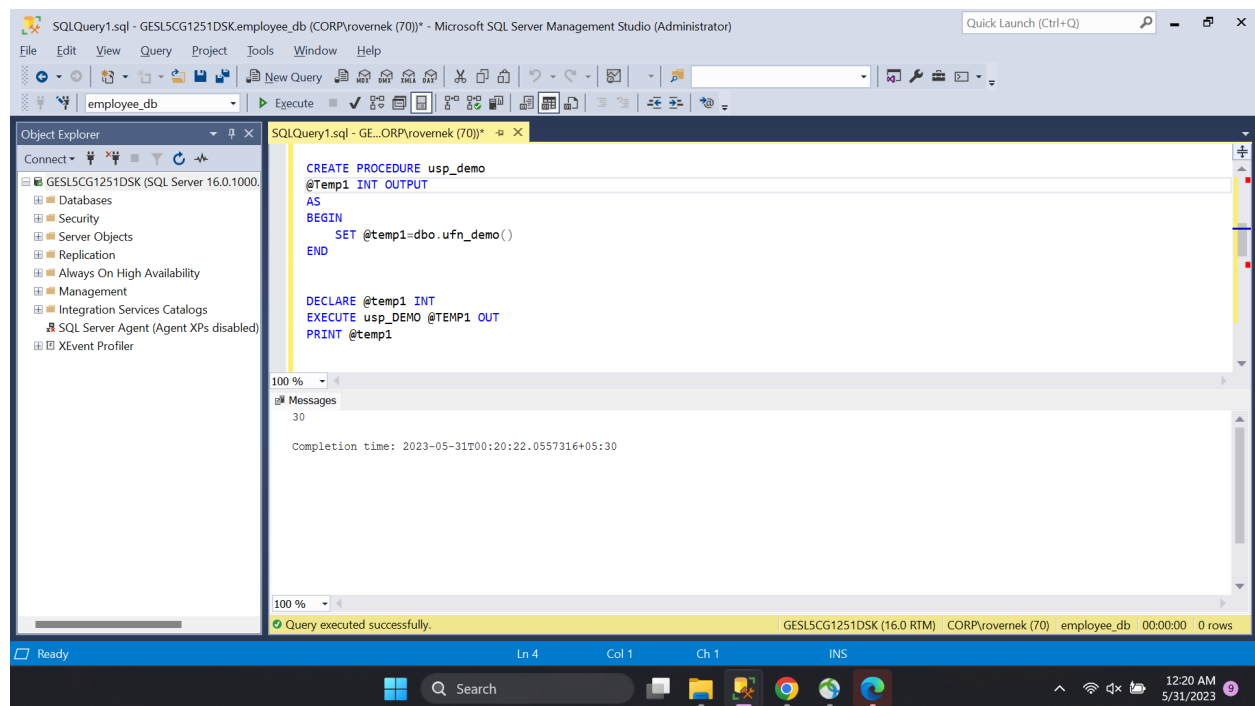
```
CREATE FUNCTION ufn_demo()  
RETURNS INT  
AS  
BEGIN  
    DECLARE @Temp1 INT ,@Temp2 INT  
    SET @Temp1 =10  
    SET @Temp2 = 20  
    RETURN @Temp1+@Temp2  
END  
  
select dbo.ufn_demo()
```



2)Can we call a function inside a stored procedure

```
CREATE PROCEDURE usp_demo
@Temp1 INT OUTPUT
AS
BEGIN
    SET @temp1=dbo.ufn_demo()
END
```

```
DECLARE @temp1 INT
EXECUTE usp_DEMO @TEMP1 OUT
PRINT @temp1
```

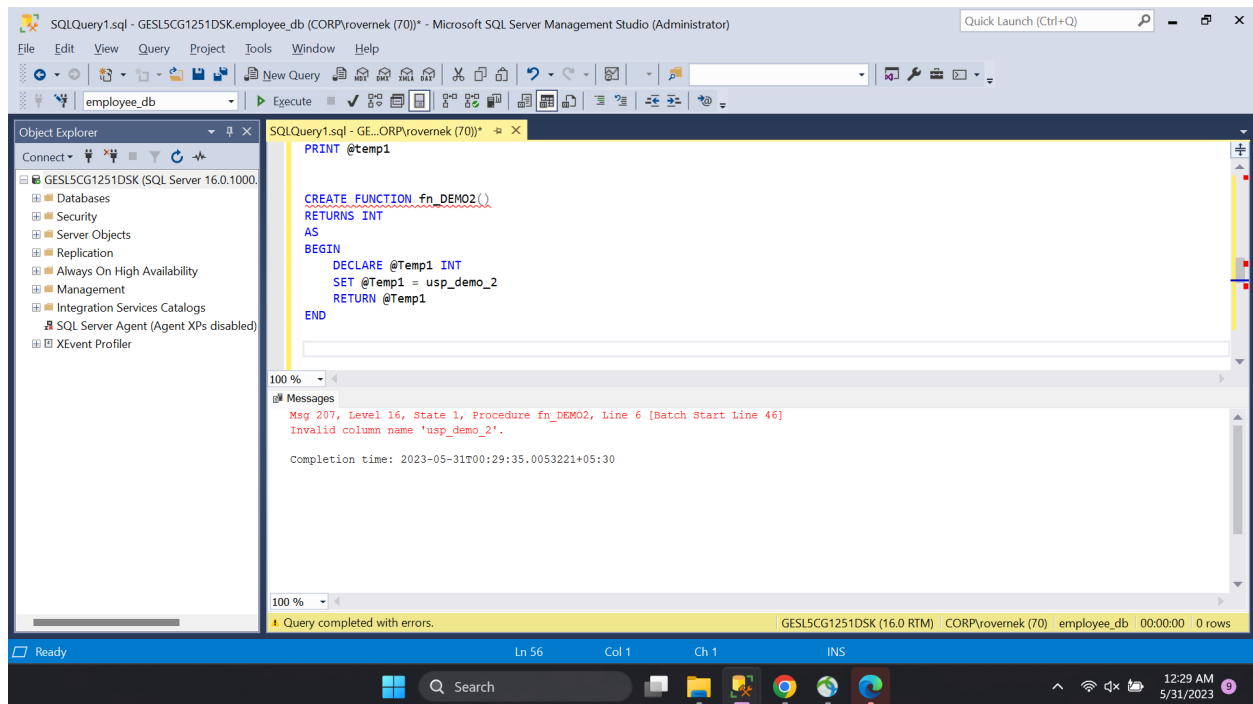


3) Can we call a stored procedure inside a function?

```
CREATE PROCEDURE usp_demo_2
@TEMP1 int OUTPUT
as
begin
    set @TEMP1=50
end
```

```
DECLARE @temp1 INT
EXECUTE usp_demo_2 @temp1 OUT
PRINT @temp1
```

```
CREATE FUNCTION fn_DEMO2()
RETURNS INT
AS
BEGIN
    DECLARE @Temp1 INT
    SET @Temp1 = usp_demo_2
    RETURN @Temp1
END
```



4)Can we call trigger inside a Stored Procedure

```
CREATE TABLE demo (  
    a INT PRIMARY KEY,    b int,    c int  
);
```

```
CREATE TRIGGER UpdateTrigger  
ON demo  
AFTER INSERT  
AS  
BEGIN  
    UPDATE demo  
    SET a = b + c  
END;
```

```
CREATE PROCEDURE InsertWithTrigger  
    @a INT,  
    @b int,  
    @c int  
AS  
BEGIN  
    INSERT INTO demo VALUES (@a, @b, @c);  
    EXEC UpdateTrigger;  
END;
```

```
EXEC InsertWithTrigger @a = 34, @b = 4, @c = 30;
```

```
select * from demo
```

SQLQuery1.sql - GESL5CG1251DSK.employee_db (CORP\rovernek (70)) - Microsoft SQL Server Management Studio (Administrator)

File Edit View Query Project Tools Window Help

Connect | employee_db | Execute | SQL Server Enterprise Edition | SQL Server 16.0.1000.1000 | SQL Server Agent (Agent XPs disabled) | XEvent Profiler

SQLQuery1.sql - GE...ORP\rovernek (70)*

```
INSERT INTO demo VALUES (@a, @b, @c);  
  
EXEC UpdateTrigger;  
  
END;  
  
EXEC InsertWithTrigger @a = 34, @b = 4, @c = 30;  
  
select * from demo
```

110 %

Results Messages

	a	b	c
1	34	4	30

Query executed successfully. GESL5CG1251DSK (16.0 RTM) CORP\rovernek (70) employee_db 00:00:00 1 rows

Ready Ln 100 Col 1 Ch 1 INS

Search

1:51 AM 5/31/2023

5)Join table valued function with a table

```
SELECT * FROM tbl_employee
```

```
CREATE FUNCTION ufn_empInfo()
```

```
RETURNS TABLE
```

```
AS
```

```
RETURN(SELECT * FROM tbl_EMPLOYEE)
```

```
select * from tbl_Department
```

```
select empno,ename,job,d.deptno,dname from ufn_empInfo() e join tbl_department d on  
e.deptno=d.deptno
```

The screenshot displays the Microsoft SQL Server Management Studio (SSMS) interface. The title bar indicates the connection to 'GESL5CG1251DSK\employee_db (CORP\rovernek (70))'. The left-hand 'Object Explorer' pane shows the database structure, including 'Databases', 'Security', 'Server Objects', 'Replication', 'Always On High Availability', 'Management', 'Integration Services Catalogs', 'SQL Server Agent (Agent XPs disabled)', and 'XEvent Profiler'. The central 'Query Editor' window contains the following SQL script:

```
create FUNCTION ufn_empInfo()  
RETURNS TABLE  
AS  
RETURN(SELECT * FROM tbl_EMPLOYEE)  
  
select * from tbl_Department  
  
select empno,ename,job,d.deptno,dname from ufn_empInfo() e join tbl_department d on e.deptno=d.deptno
```

The bottom pane shows the 'Results' tab with 14 rows of data. The status bar at the bottom indicates 'Query executed successfully.' and 'GESL5CG1251DSK (16.0 RTM) | CORP\rovernek (70) | employee_db | 00:00:00 | 14 rows'.

empno	ename	job	deptno	dname	
2	8499	Anya	SALESMAN	30	Sales
3	8521	Seth	SALESMAN	30	Sales
4	8566	Mahadevan	MANAGER	20	Research
5	8654	Momin	SALESMAN	30	Sales
6	8698	Bina	MANAGER	30	Sales
7	8882	Shiavsh	MANAGER	10	Accounting
8	8888	Scott	ANALYST	20	Research
9	8839	Amir	PRESIDENT	10	Accounting
10	8844	Kuldeep	SALESMAN	30	Sales
11	8886	Anoop	CLERK	20	Research
12	8100	Yash	ANALYST	20	Research

6) Join table valued function(with input parameters) with a table

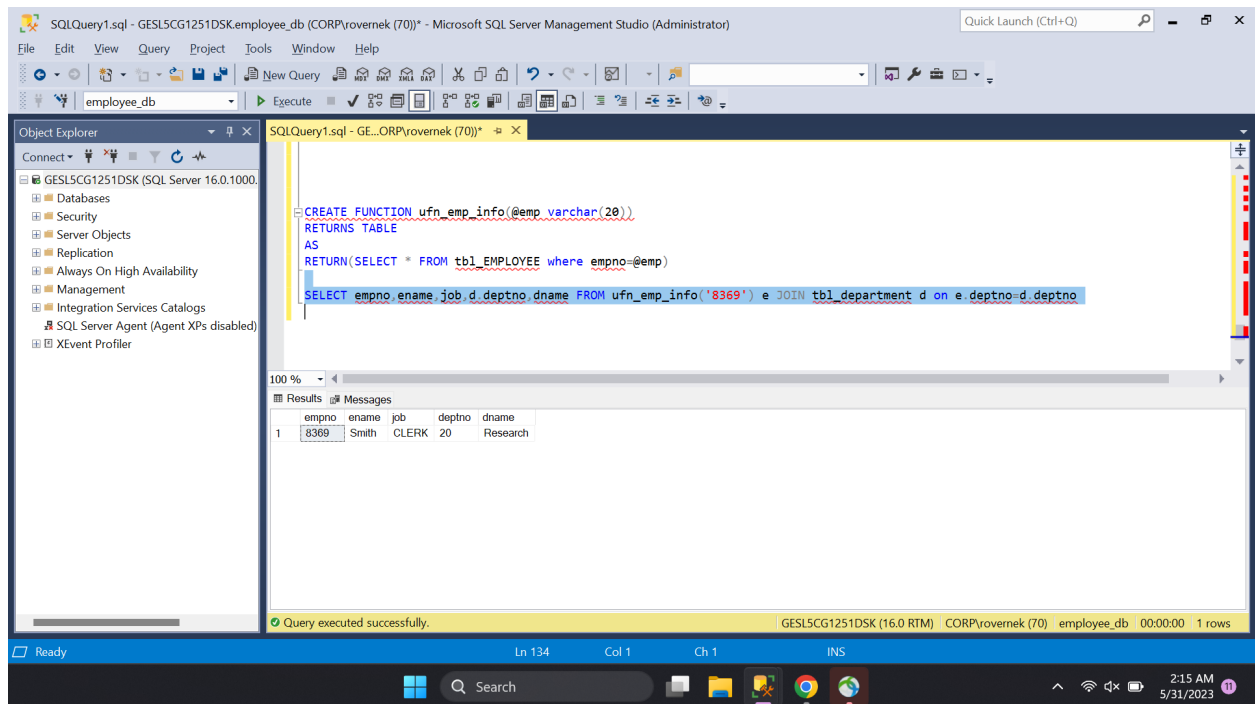
```
CREATE FUNCTION ufn_emp_info(@emp varchar(20))
```

```
RETURNS TABLE
```

```
AS
```

```
RETURN(SELECT * FROM EMPLOYEE where empno=@emp)
```

```
SELECT empno,ename,job,d.deptno,dname FROM ufn_emp_info('8369') e JOIN department d  
on e.deptno=d.deptno
```



7) Instead of triggers for tables

```
CREATE TABLE demo2 (  
    a INT PRIMARY KEY,  
    b INT,  
    c DATE  
);
```

```
CREATE TRIGGER InsteadOfInsertTrigger  
ON demo2  
INSTEAD OF INSERT  
AS  
BEGIN  
    INSERT INTO demo2 (a, b, c)  
    SELECT a, b, GETDATE()  
    FROM inserted;  
END;
```

```
select * from demo2
```

```
INSERT INTO demo2(a, b) VALUES (2, 200);
```

