

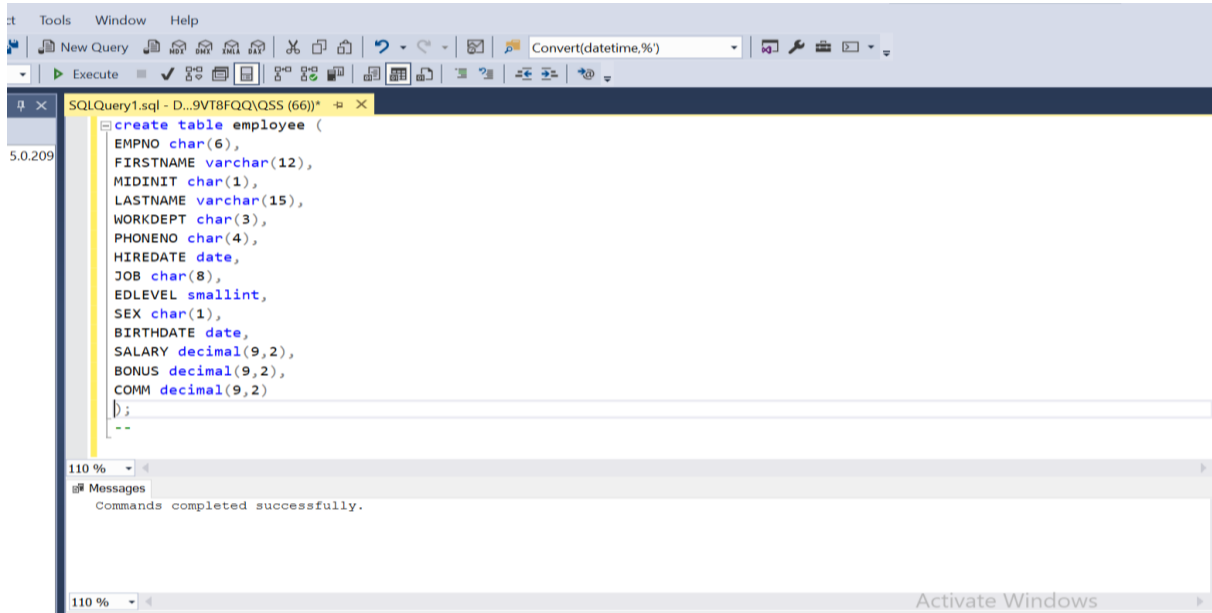
## **ASSIGNMENT 2 – USER DEFINED FUNCTIONS**

**Name- Amrit Pal Singh**

**Student ID- 110355211**

# 1. Preparation

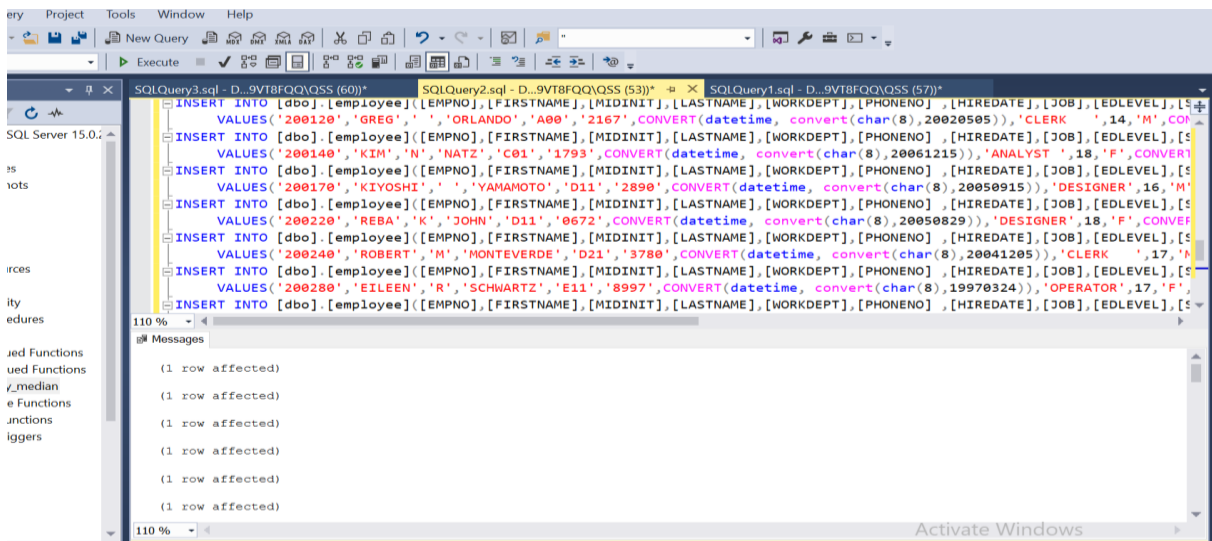
## Create Employee Table



The screenshot shows the SQL Server Enterprise Manager interface. The 'Messages' pane at the bottom indicates that the commands were completed successfully. The SQL query executed is as follows:

```
create table employee (  
    EMPNO char(6),  
    FIRSTNAME varchar(12),  
    MIDINIT char(1),  
    LASTNAME varchar(15),  
    WORKDEPT char(3),  
    PHONENO char(4),  
    HIREDATE date,  
    JOB char(8),  
    EDLEVEL smallint,  
    SEX char(1),  
    BIRTHDATE date,  
    SALARY decimal(9,2),  
    BONUS decimal(9,2),  
    COMM decimal(9,2)  
);
```

## Insert data into Employee table



The screenshot shows the SQL Server Enterprise Manager interface with three SQL queries executed. The 'Messages' pane at the bottom shows that each of the six INSERT statements affected one row. The SQL queries are as follows:

```
INSERT INTO [dbo].[employee] ([EMPNO],[FIRSTNAME],[MIDINIT],[LASTNAME],[WORKDEPT],[PHONENO],[HIREDATE],[JOB],[EDLEVEL],[SEX])  
VALUES ('200120','GREG',' ','ORLANDO','A00','2167',CONVERT(datetime, convert(char(8),20020505)), 'CLERK', '14','M', 'C')  
INSERT INTO [dbo].[employee] ([EMPNO],[FIRSTNAME],[MIDINIT],[LASTNAME],[WORKDEPT],[PHONENO],[HIREDATE],[JOB],[EDLEVEL],[SEX])  
VALUES ('200140','KIM','N','NATZ','C01','1793',CONVERT(datetime, convert(char(8),20061215)), 'ANALYST', '18','F', 'C')  
INSERT INTO [dbo].[employee] ([EMPNO],[FIRSTNAME],[MIDINIT],[LASTNAME],[WORKDEPT],[PHONENO],[HIREDATE],[JOB],[EDLEVEL],[SEX])  
VALUES ('200170','KIYOSHI',' ','YAMAMOTO','D11','2890',CONVERT(datetime, convert(char(8),20050915)), 'DESIGNER', '16','M', 'C')  
INSERT INTO [dbo].[employee] ([EMPNO],[FIRSTNAME],[MIDINIT],[LASTNAME],[WORKDEPT],[PHONENO],[HIREDATE],[JOB],[EDLEVEL],[SEX])  
VALUES ('200220','REBA','K','JOHN','D11','0672',CONVERT(datetime, convert(char(8),20050829)), 'DESIGNER', '18','F', 'C')  
INSERT INTO [dbo].[employee] ([EMPNO],[FIRSTNAME],[MIDINIT],[LASTNAME],[WORKDEPT],[PHONENO],[HIREDATE],[JOB],[EDLEVEL],[SEX])  
VALUES ('200240','ROBERT','M','MONTEVERDE','D21','3780',CONVERT(datetime, convert(char(8),20041205)), 'CLERK', '17','M', 'C')  
INSERT INTO [dbo].[employee] ([EMPNO],[FIRSTNAME],[MIDINIT],[LASTNAME],[WORKDEPT],[PHONENO],[HIREDATE],[JOB],[EDLEVEL],[SEX])  
VALUES ('200280','EILEEN','R','SCHWARTZ','E11','8997',CONVERT(datetime, convert(char(8),19970324)), 'OPERATOR', '17','F', 'C')
```

SQLQuery3.sql - D...9VT8FQQ\QSS (60))\* X SQLQuery2.sql - D...9VT8FQQ\QSS (53))\* SQLQuery1.sql - D...9VT8FQQ\QSS (57))\*

```
Select count(*) from employee
Select * from employee
```

110 %

Results Messages

(No column name)

1 42

	EMPNO	FIRSTNAME	MIDINIT	LASTNAME	WORKDEPT	PHONENO	HIREDATE	JOB	EDLEVEL	SEX	BIRTHDATE	SALARY	BONUS	COMM
1	000010	CHRISTINE	I	HAAS	A00	3978	1995-01-01	PRES	18	F	1963-08-24	152750.00	1000.00	4220.00
2	000020	MICHAEL	L	THOMPSON	B01	3476	2003-10-10	MANAGER	18	M	1978-02-02	94250.00	800.00	3300.00
3	000030	SALLY	A	KWAN	C01	4738	2005-04-05	MANAGER	20	F	1971-05-11	98250.00	800.00	3060.00
4	000050	JOHN	B	GEYER	E01	6789	1979-08-17	MANAGER	16	M	1955-09-15	80175.00	800.00	3214.00
5	000060	IRVING	F	STERN	D11	6423	2003-09-14	MANAGER	16	M	1975-07-07	72250.00	500.00	2580.00
6	000070	EVA	D	PULASKI	D21	7831	2005-09-30	MANAGER	16	F	2003-05-26	96170.00	700.00	2893.00
7	000090	EILEEN	W	HENDER...	E11	5498	2000-08-15	MANAGER	16	F	1971-05-15	89750.00	600.00	2380.00
8	000100	THEODORE	Q	SPENSER	E21	0972	2000-06-19	MANAGER	14	M	1980-12-18	86150.00	500.00	2092.00
9	000110	VINCENZO	G	IUCCHES	A00	3490	1988-05-16	SAI FSR	19	M	1959-11-05	66500.00	800.00	3720.00

Query executed successfully. DESKTOP-9VT8FQQ (15.0 RTM) DESKTOP-9VT8FQQ\QSS (60) Test: 00:00:00 43 rows

## Overview of the employee table

## Create Staff Table

SQLQuery1.sql - D...9VT8FQQ\QSS (66))\* X

```
-- Create the STAFF table
--
create table staff (
  ID smallint,
  NAME varchar(9),
  DEPT smallint,
  JOB char(5),
  YEARS smallint,
  SALARY decimal(7,2),
  COMM decimal(7,2)
);
```

110 %

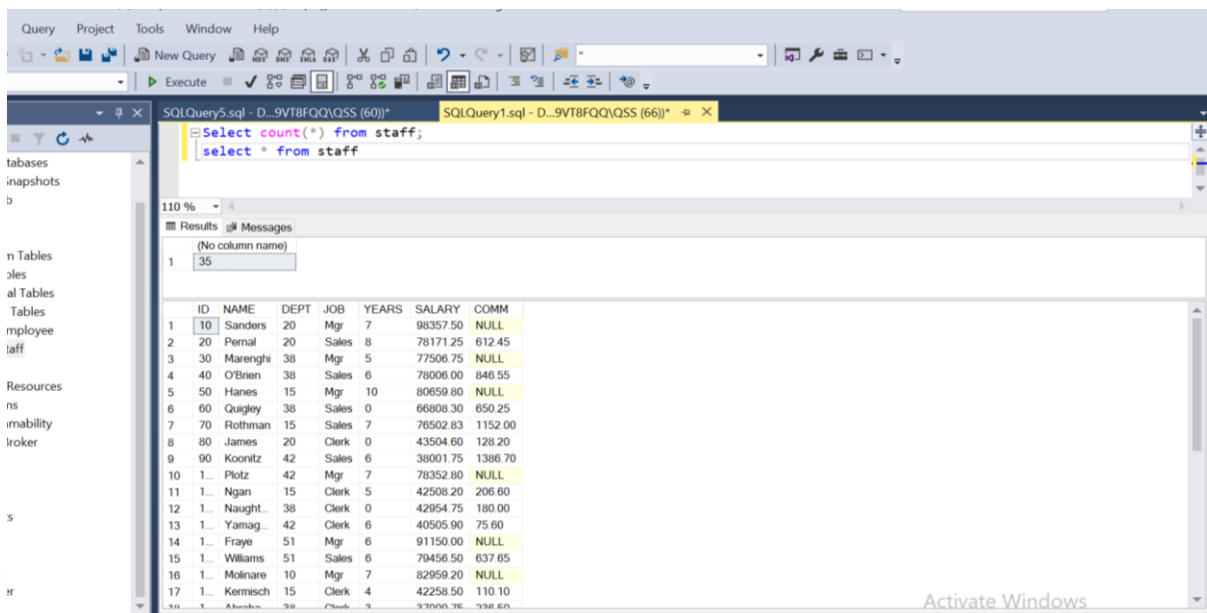
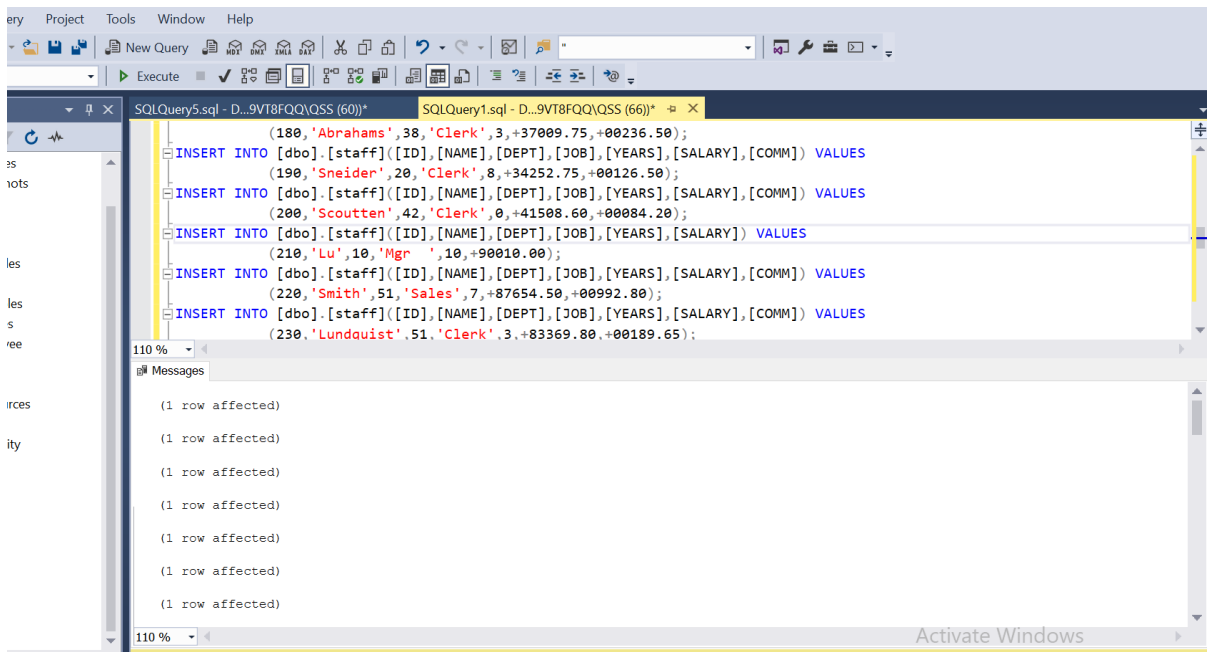
Messages

Commands completed successfully.

110 %

Activate Windows

## Insert data into Staff table



## Overview of the staff table

## 2. my\_median

Write a function called *my\_median* which takes the values in a column (like salary) as input and will output the mathematical median of the values in that column.

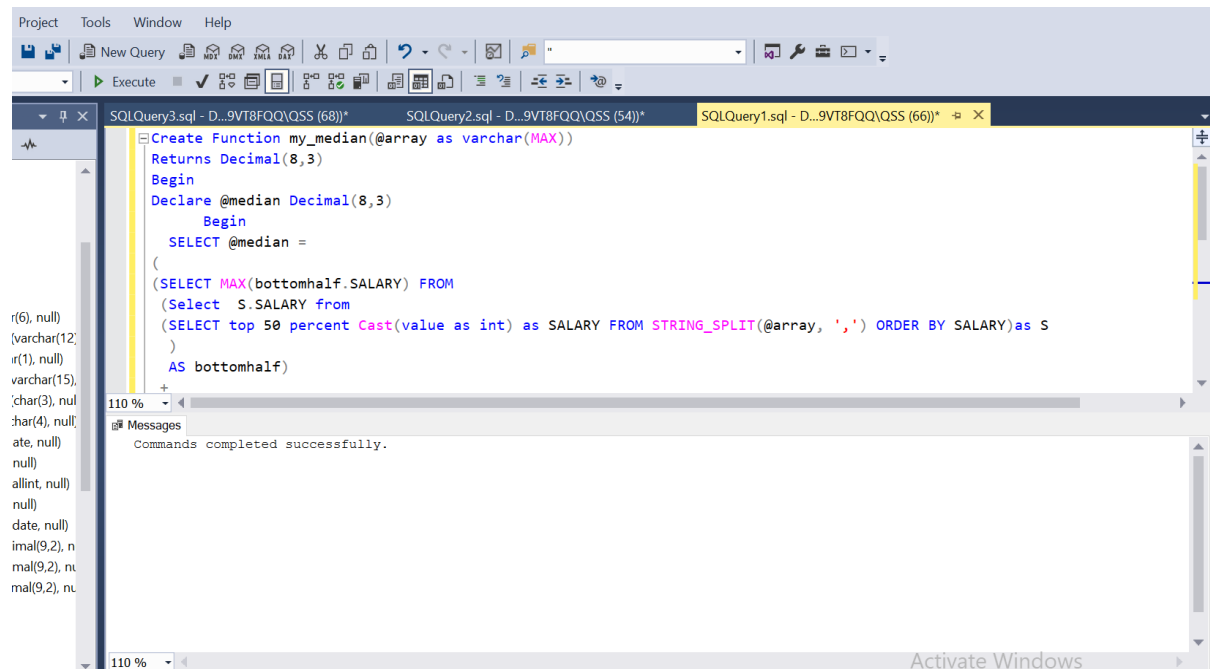
The definition of median is:

**X** is the **ordered** list of elements

**n** is the number of elements in the list

$\text{Median}(X) = X[(n+1)/2]$ , if **n** is odd

$\text{Median}(X) = (X[n/2] + X[(n/2) + 1])/2$ , if **n** is even



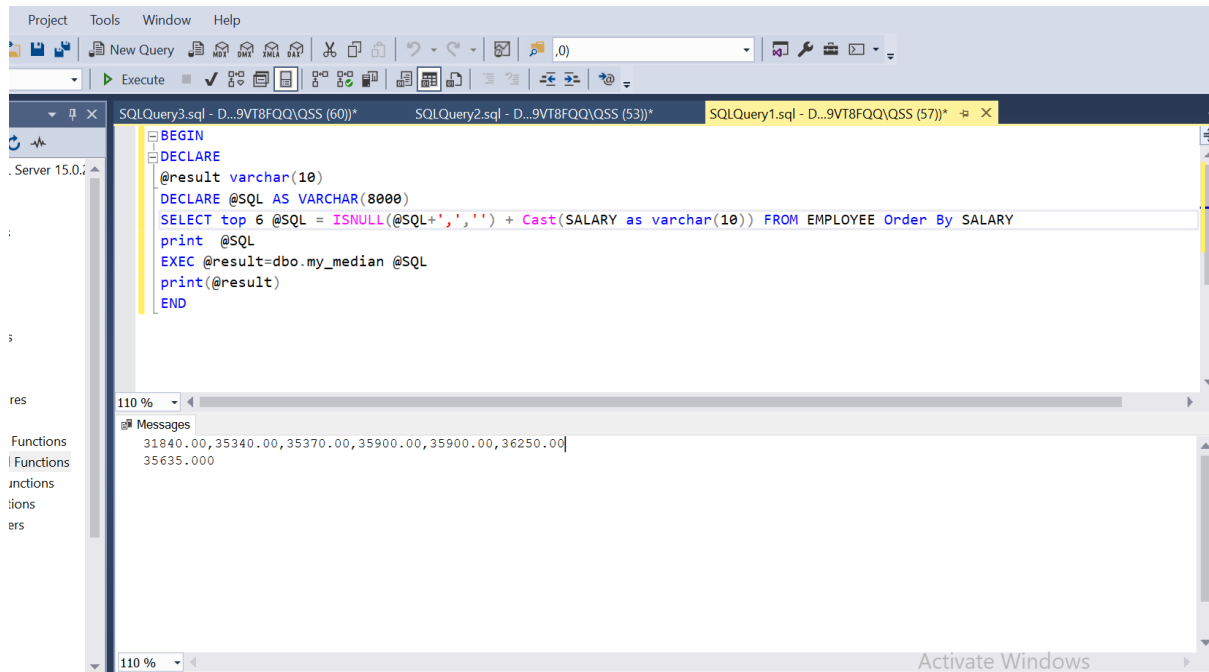
The screenshot shows the SQL Server Enterprise Manager interface. The main window displays the following T-SQL code for creating a function named `my_median`:

```
Create Function my_median(@array as varchar(MAX))
Returns Decimal(8,3)
Begin
Declare @median Decimal(8,3)
Begin
SELECT @median =
(
SELECT MAX(bottomhalf.SALARY) FROM
(Select S.SALARY from
(SELECT top 50 percent Cast(value as int) as SALARY FROM STRING_SPLIT(@array, ',')) AS bottomhalf)
)
)

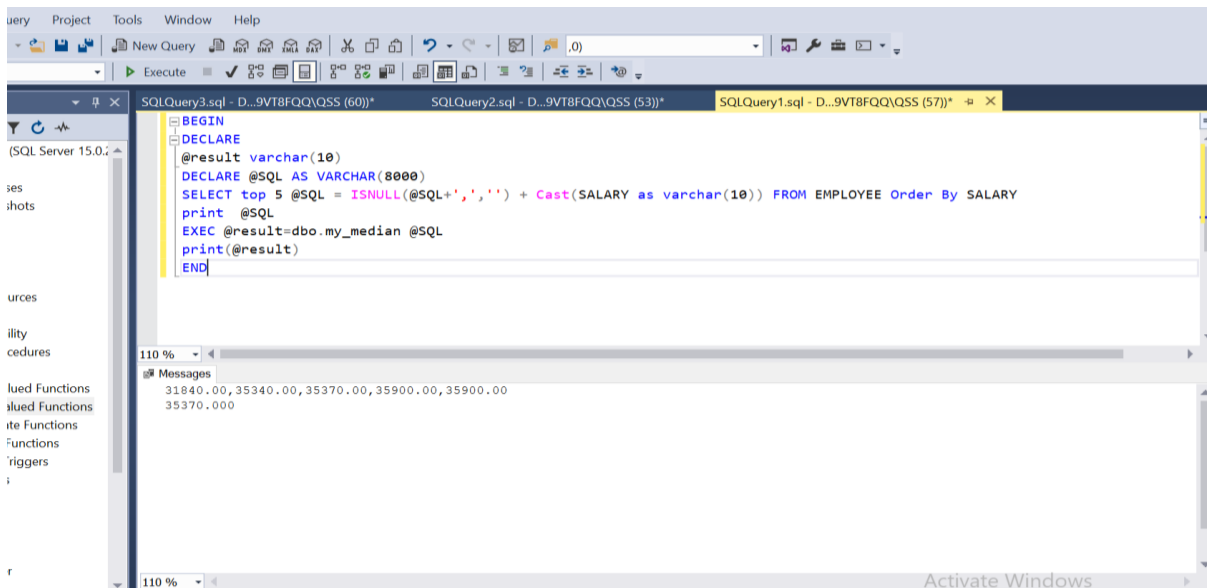
```

The Messages pane at the bottom indicates that the commands were completed successfully.

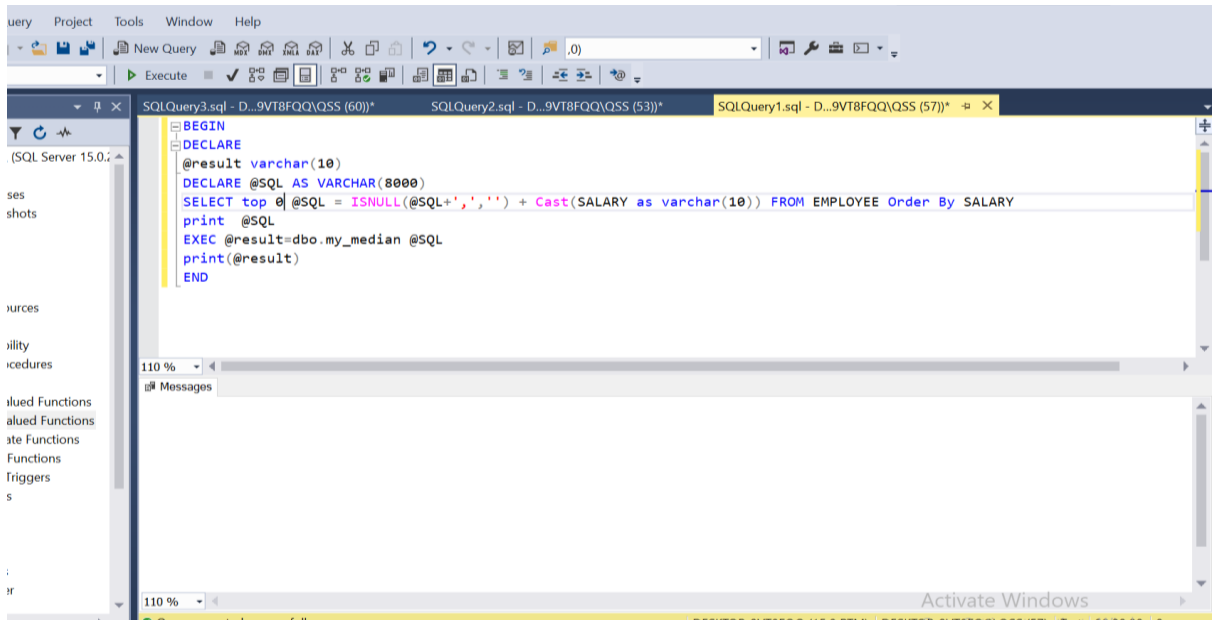
**my\_median**



**A list with an even amount of elements**



**A list with an odd amount of elements**



## An Empty List

### 3. my\_mode

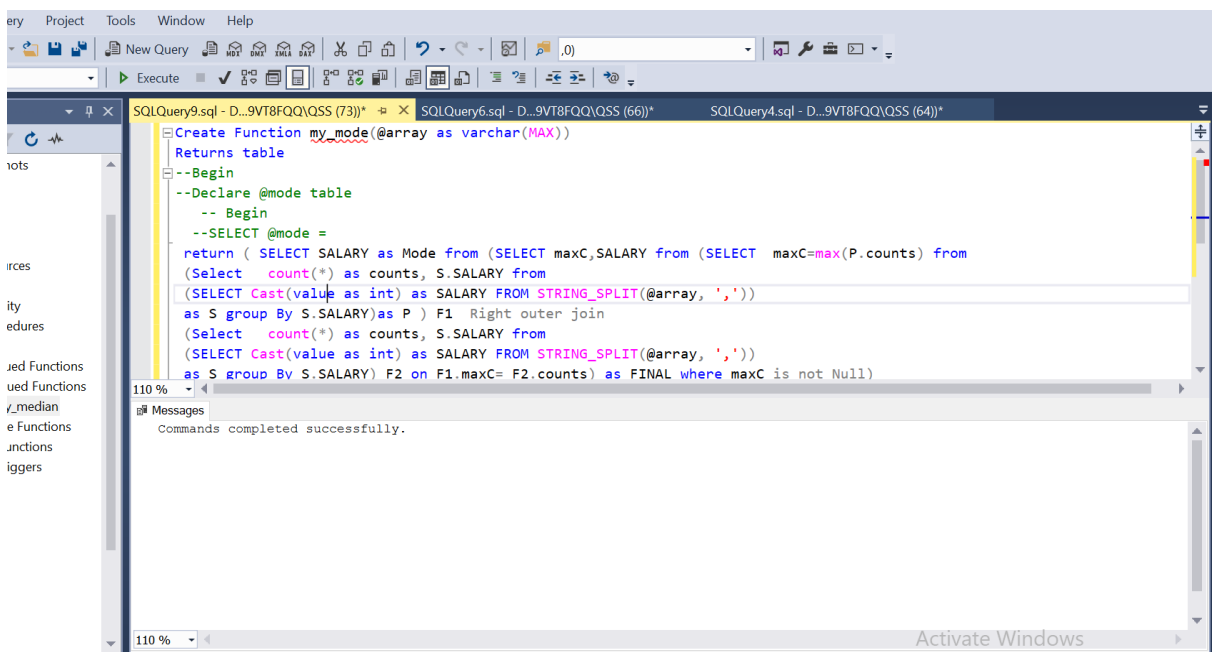
Write a function called *my\_mode* which takes the values in a column (like department) as input and will output the mathematical mode of the values in that column.

The definition of mode is the value which occurs most frequently in the list. There are three cases:

A – There is one mode in the list

B – There are multiple modes in the list

C – There are no modes in the list



```
SQLQuery9.sql - D:\9VT8FQQ\QSS (73)*
SQLQuery6.sql - D:\9VT8FQQ\QSS (66)*
SQLQuery4.sql - D:\9VT8FQQ\QSS (64)*

Create Function my_mode(@array as varchar(MAX))
Returns table
--Begin
--Declare @mode table
-- Begin
--SELECT @mode =
return ( SELECT SALARY as Mode from (SELECT maxC,SALARY from (SELECT maxC=max(P.counts) from
(Select count(*) as counts, S.SALARY from
(SELECT Cast(value as int) as SALARY FROM STRING_SPLIT(@array, ','))
as S group By S.SALARY)as P ) F1 Right outer join
(Select count(*) as counts, S.SALARY from
(SELECT Cast(value as int) as SALARY FROM STRING_SPLIT(@array, ','))
as S group By S.SALARY) F2 on F1.maxC= F2.counts) as FINAL where maxC is not Null)
)

Messages
Commands completed successfully.
```



## my\_mode

The screenshot shows the SQL Server Enterprise Manager interface. The query editor contains the following T-SQL code:

```
Begin
--Declare @result table
Declare @SQL as varchar(8000)
Select top 3 @SQL= ISNULL(@SQL+',') +CAST(DEPT As VARCHAR(10)) From STAFF
Select @SQL as [DEPT]
Select * from dbo.my_mode(@SQL)
--print(@result)

END
```

The Results pane shows the output of the query:

DEPT
1 20,20,38

Mode
1 20

The interface includes a menu bar (Query, Project, Tools, Window, Help), a toolbar, and a left-hand pane with a tree view showing the database structure.

## A list with one mode

The screenshot shows the SQL Server Enterprise Manager interface. The query editor contains the following T-SQL code:

```
Begin
--Declare @result table
Declare @SQL as varchar(8000)
Select top 5 @SQL= ISNULL(@SQL+',') +CAST(DEPT As VARCHAR(10)) From STAFF
Select @SQL as [DEPT]
Select * from dbo.my_mode(@SQL)
--print(@result)

END
```

The Results pane shows the output of the query:

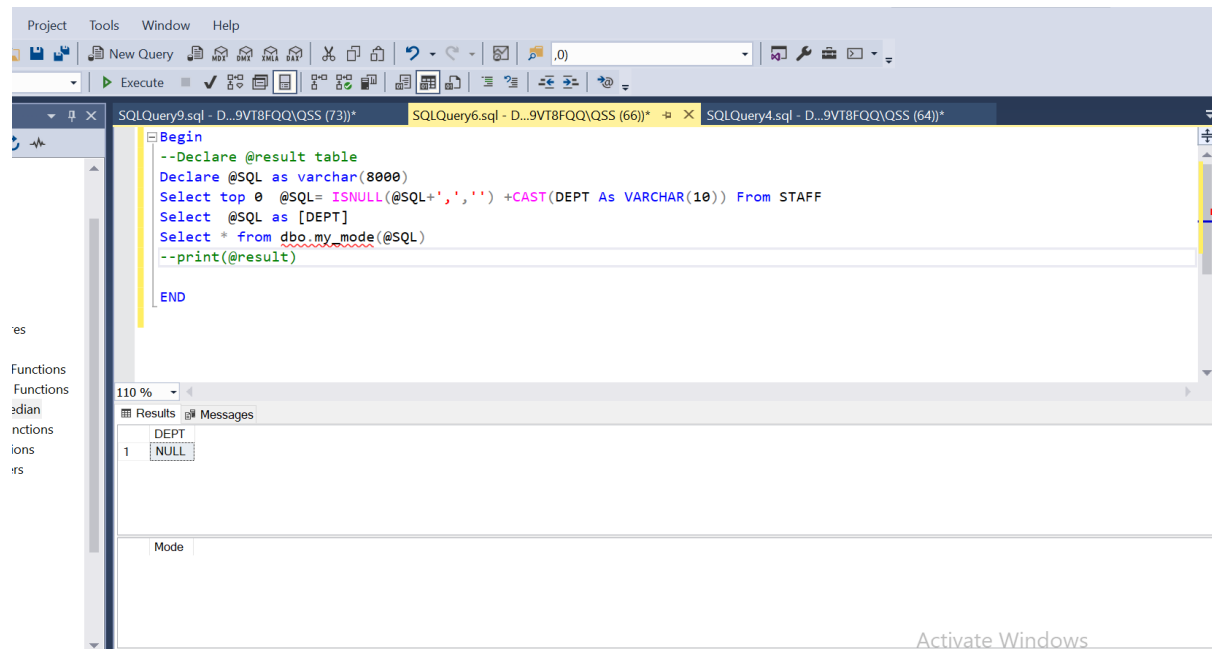
DEPT
1 20,20,38,38,15

Mode
1 20
2 38

The interface includes a menu bar (Query, Project, Tools, Window, Help), a toolbar, and a left-hand pane with a tree view showing the database structure.

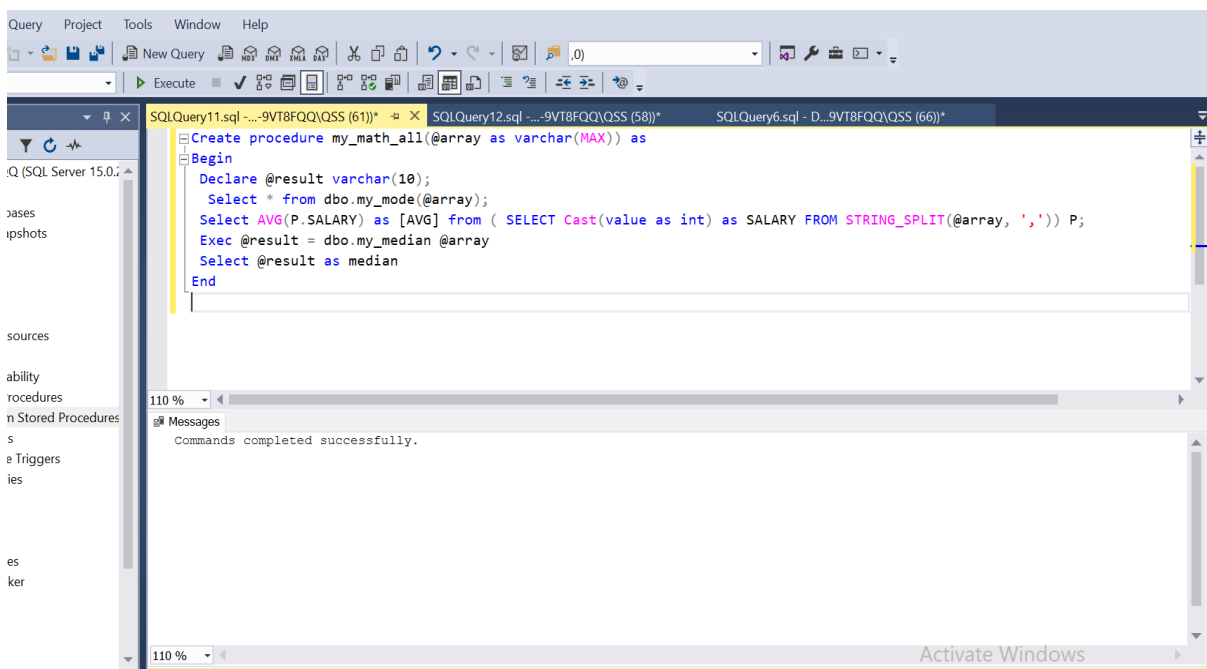
## A list with two modes



## An Empty list

## 4. my\_math\_all

Write a procedure called *my\_math\_all* which takes the values in a column (like department or salary) as input and will output the mathematical median, mode and mean of the values in that column.



The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'Stored Procedures' folder under the 'sources' tree. The right pane shows the SQL query editor with the following code:

```
CREATE procedure my_math_all(@array as varchar(MAX)) as
Begin
    Declare @result varchar(10);
    Select * from dbo.my_mode(@array);
    Select AVG(P.SALARY) as [AVG] from ( SELECT Cast(value as int) as SALARY FROM STRING_SPLIT(@array, ',')) P;
    Exec @result = dbo.my_median @array
    Select @result as median
End
```

Below the query editor, the 'Messages' pane shows the message: 'Commands completed successfully.'

**my\_math\_all**

SQL Server 15.0.2

```

Declare @SQL as varchar(8000)
Select top 5 @SQL= ISNULL(@SQL+',','') +CAST(DEPT As VARCHAR(10)) From STAFF
Select @SQL as [DEPT]
EXEC [dbo].[my_math_all] @SQL

```

Results

DEPT
20,20,38,38,15

Mode

Mode
20
38

AVG

AVG
26

median

median
20.000

Activate Windows

**A list which shows a median, mode and mean (average)**

SQL Server 15.0.2

```

Declare @SQL as varchar(8000)
Select top 0 @SQL= ISNULL(@SQL+',','') +CAST(DEPT As VARCHAR(10)) From STAFF
Select @SQL as [DEPT]
EXEC [dbo].[my_math_all] @SQL

```

Results

DEPT
NULL

Mode

Mode
NULL

AVG

AVG
NULL

median

median
NULL

Activate Windows

**An Empty List**