SQL Sub Query-Topics

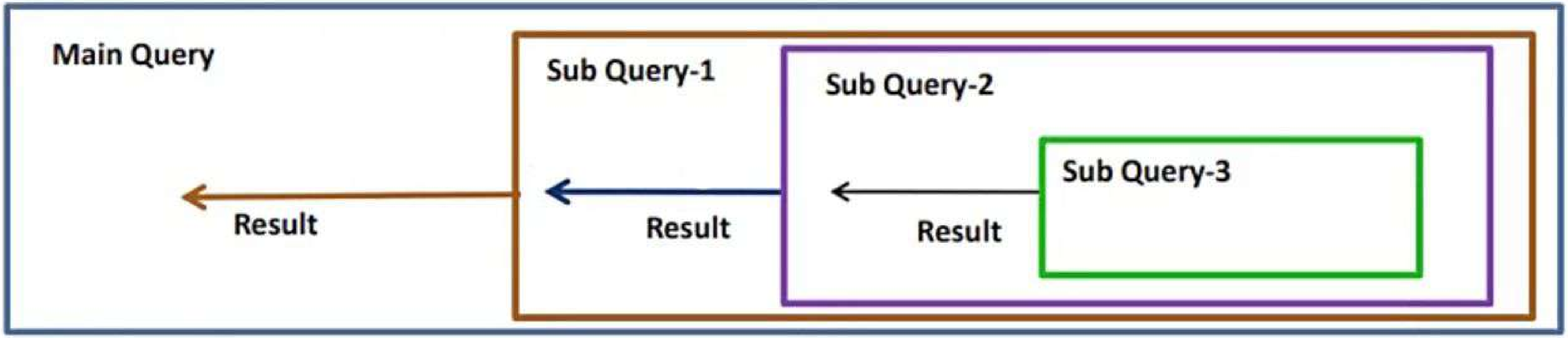
- What is sub query in SQL server?
- Types of sub queries.
    - Single row sub query
    - Multiple row sub query
    - Correlated sub query
- Examples on sql server
- Explanation with diagram

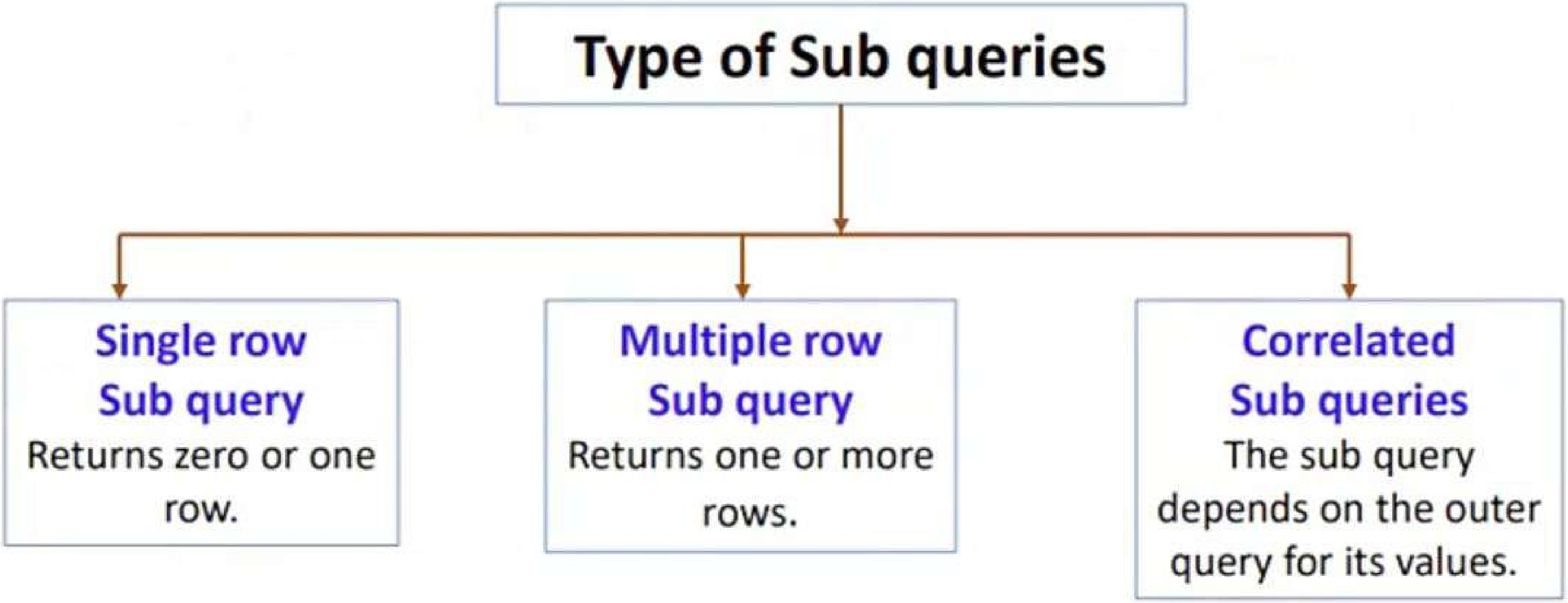
What is Sub Query ?

- A subquery is also called an inner query or inner select, while the statement containing a subquery is also called an outer query or outer select.
- **Rules**:
  - Sub queries must be enclosed within parentheses.
  - You can use the comparison operators, such as **>, <, or =, IN, ANY, or ALL.**
  - An ORDER BY command cannot be used in a sub query.

SQL Sub Query

NOTE: The inner query executes first before its parent query so that the results of an inner query can be passed to the outer query.



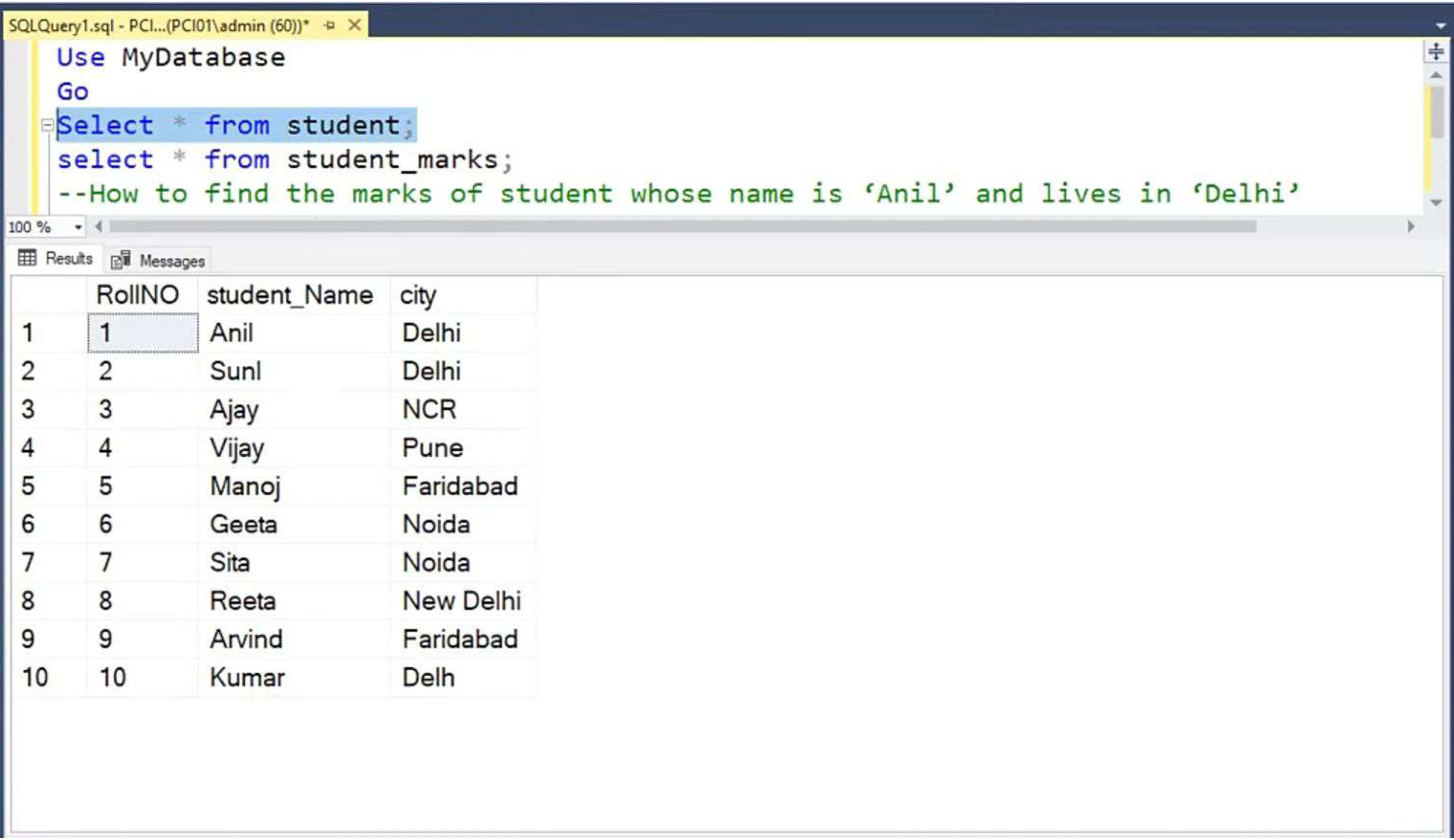


Type of Sub queries

Single row Sub query
Returns zero or one row.

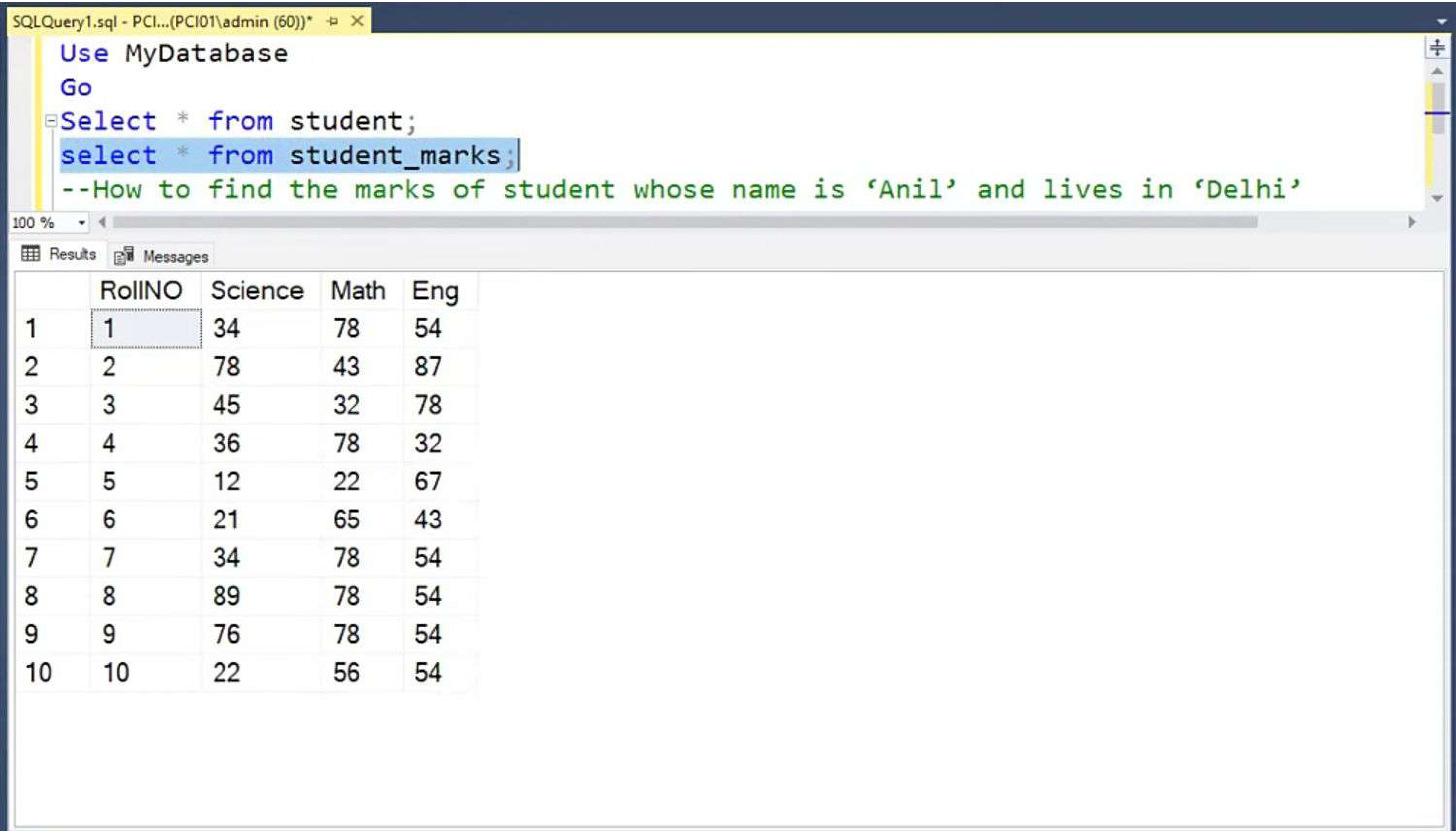
Multiple row Sub query
Returns one or more rows.

Correlated Sub queries
The sub query depends on the outer query for its values.



```
Use MyDatabase
Go
Select * from student;
select * from student_marks;
--How to find the marks of student whose name is 'Anil' and lives in 'Delhi'
```

	RollNO	student_Name	city
1	1	Anil	Delhi
2	2	Sunl	Delhi
3	3	Ajay	NCR
4	4	Vijay	Pune
5	5	Manoj	Faridabad
6	6	Geeta	Noida
7	7	Sita	Noida
8	8	Reeta	New Delhi
9	9	Arvind	Faridabad
10	10	Kumar	Delh



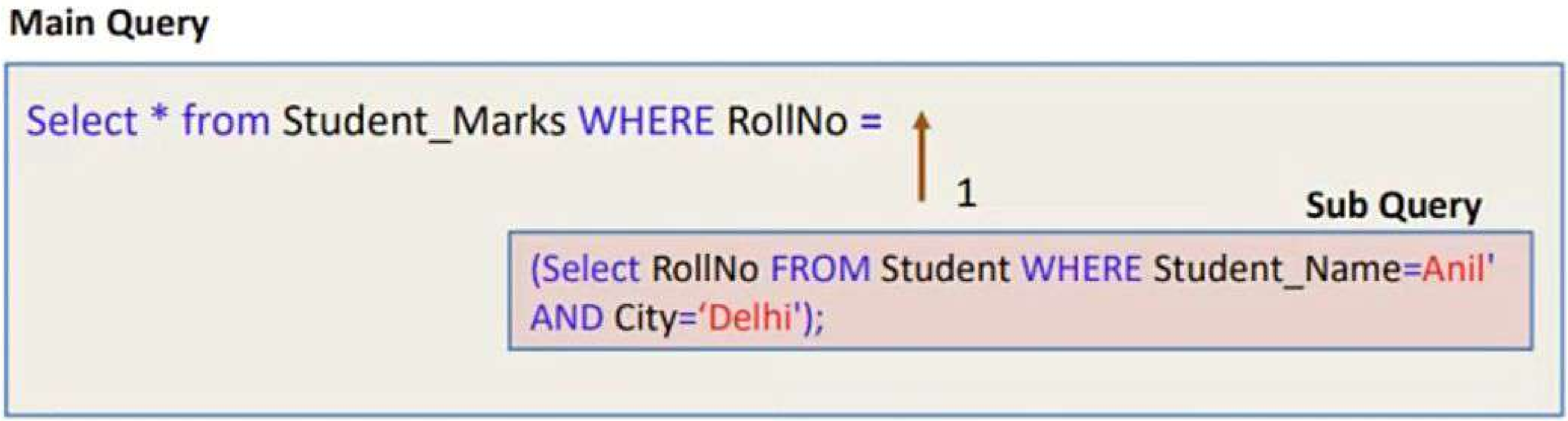
```sql
Use MyDatabase
Go
Select * from student;
select * from student_marks;
--How to find the marks of student whose name is ‘Anil’ and lives in ‘Delhi’
```

	RollNO	Science	Math	Eng
1	1	34	78	54
2	2	78	43	87
3	3	45	32	78
4	4	36	78	32
5	5	12	22	67
6	6	21	65	43
7	7	34	78	54
8	8	89	78	54
9	9	76	78	54
10	10	22	56	54

Sub Query- Single Row Return example

- Sub query must return single row value. A equal (=) operator is used for single row return.

Question: How to find the marks of student whose name is **'Anil'** and lives in **'Delhi'**



Main Query

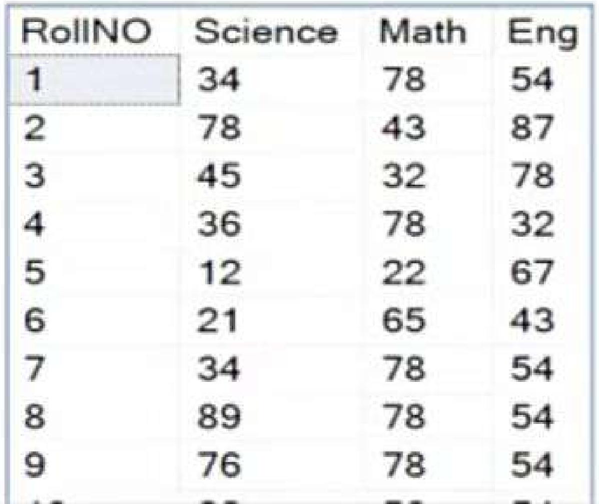
```
Select * from Student_Marks WHERE RollNo =
```

1

Sub Query

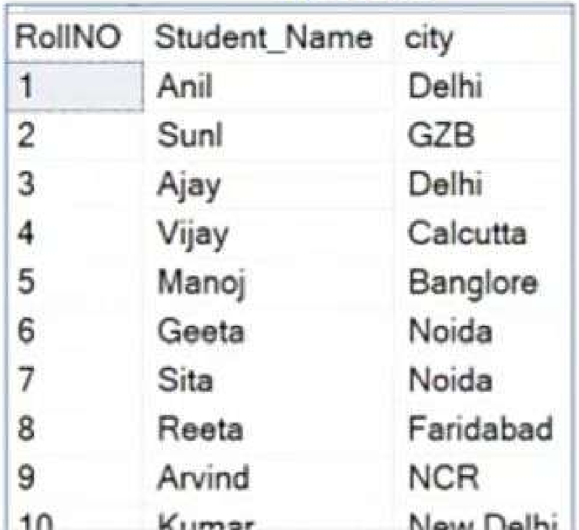
```
(Select RollNo FROM Student WHERE Student_Name=Anil'
AND City='Delhi');
```

Student_Marks

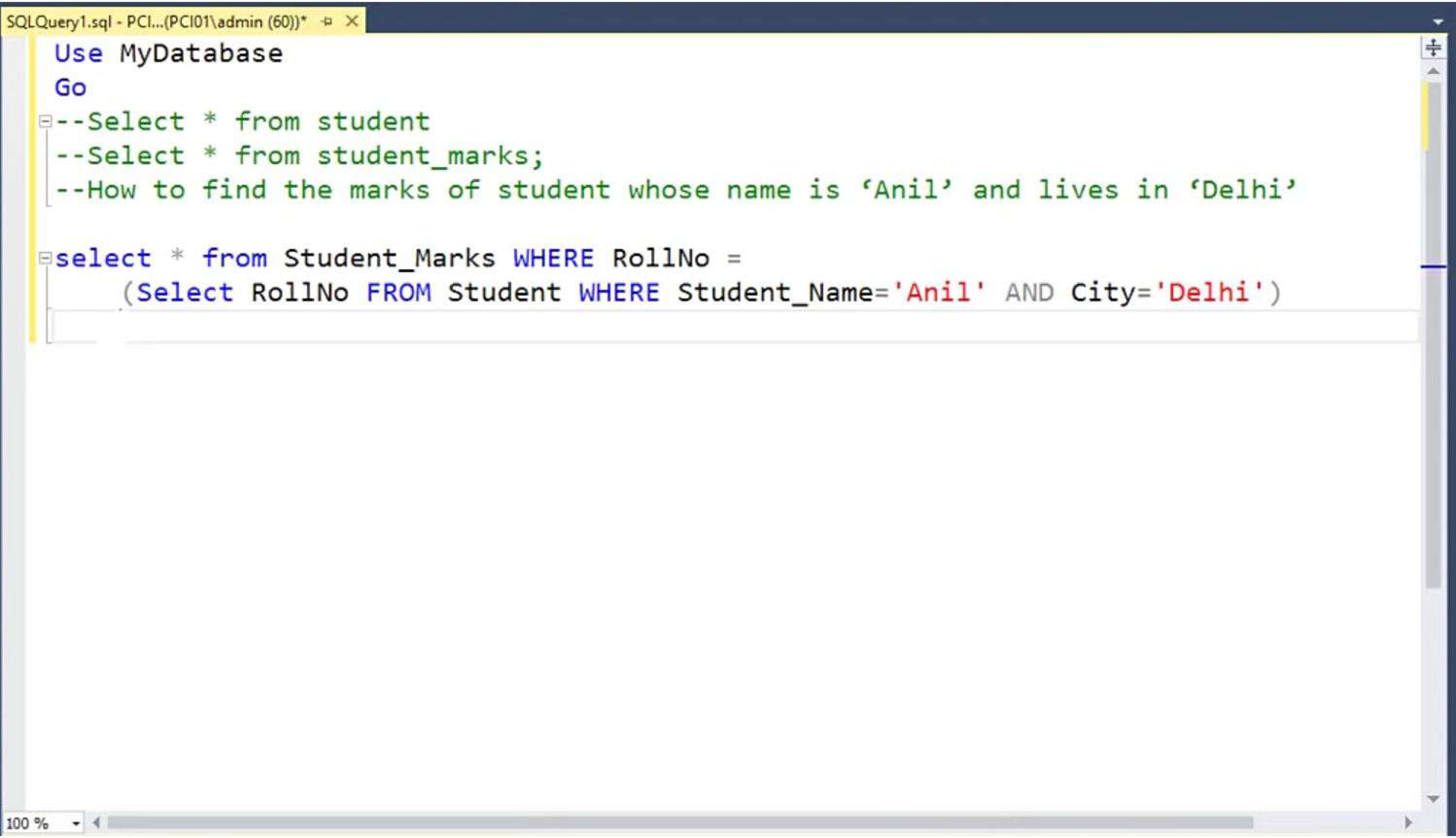


RollNO	Science	Math	Eng
1	34	78	54
2	78	43	87
3	45	32	78
4	36	78	32
5	12	22	67
6	21	65	43
7	34	78	54
8	89	78	54
9	76	78	54
10	[illegible]	[illegible]	[illegible]

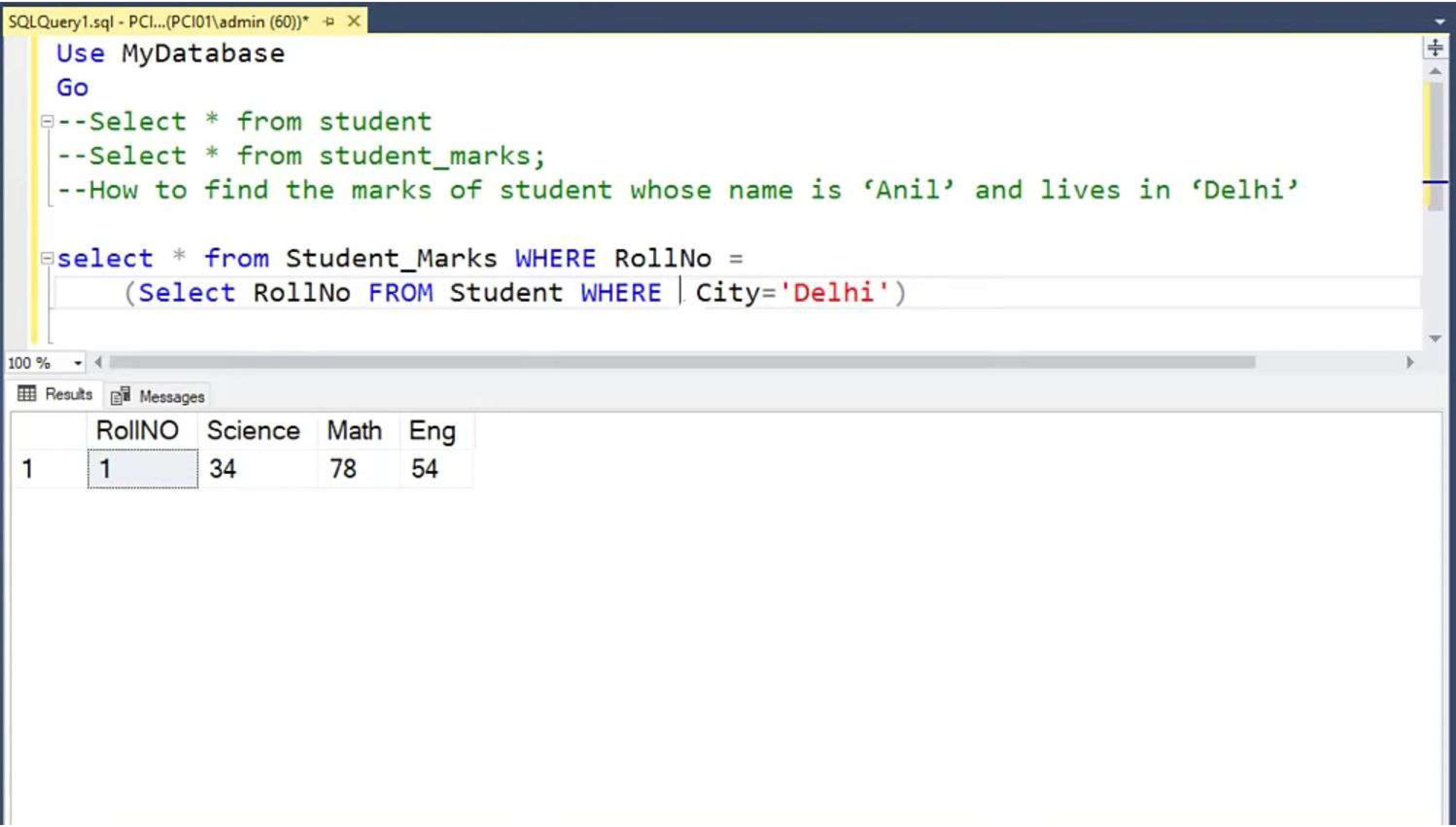
Student



RollNO	Student_Name	city
1	Anil	Delhi
2	Sunl	GZB
3	Ajay	Delhi
4	Vijay	Calcutta
5	Manoj	Banglore
6	Geeta	Noida
7	Sita	Noida
8	Reeta	Faridabad
9	Arvind	NCR
10	Kumar	New Delhi



```
Use MyDatabase
Go
--Select * from student
--Select * from student_marks;
--How to find the marks of student whose name is ‘Anil’ and lives in ‘Delhi’

select * from Student_Marks WHERE RollNo =
    (Select RollNo FROM Student WHERE Student_Name='Anil' AND City='Delhi')
```



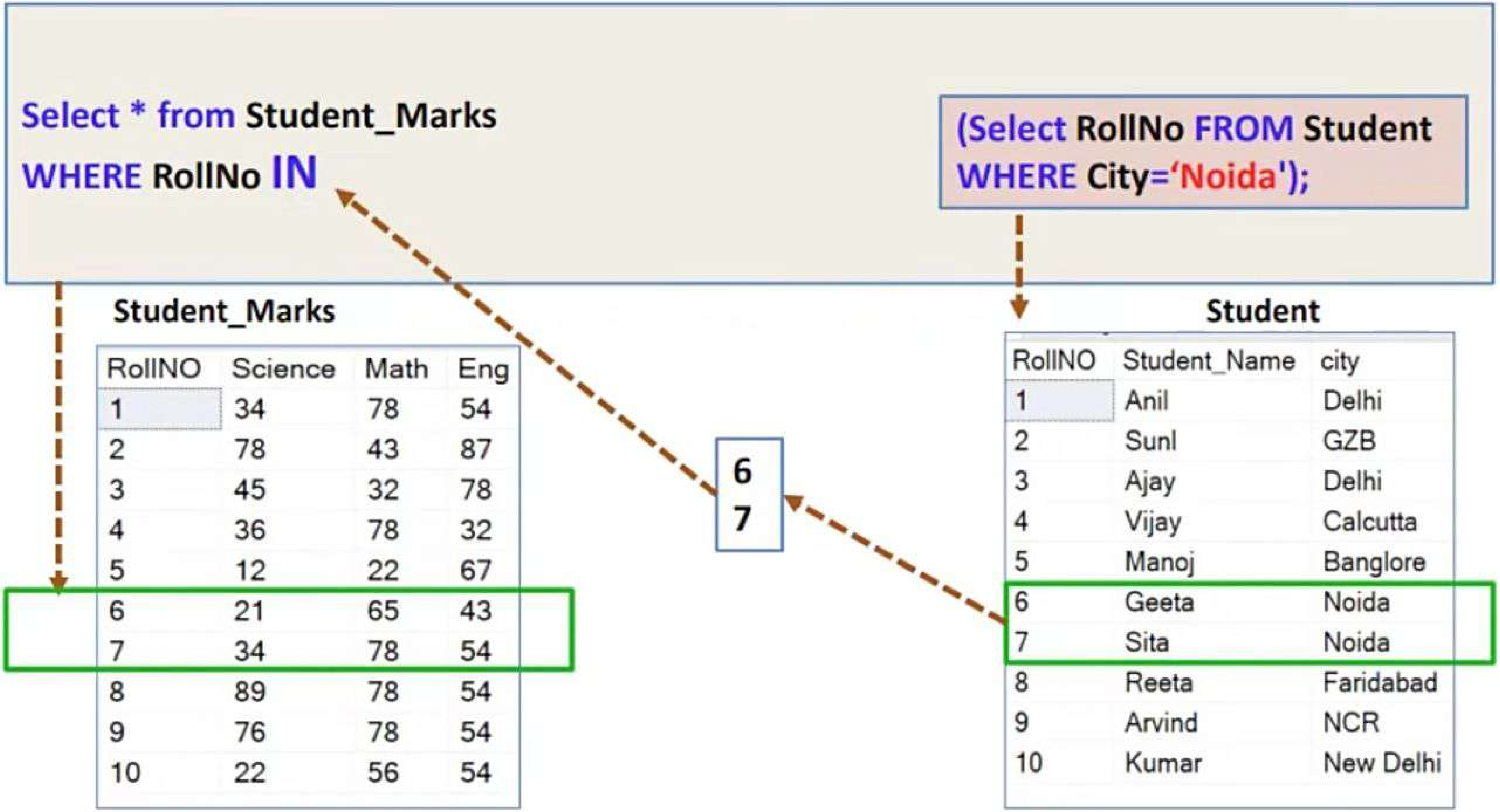
```
Use MyDatabase
Go
--Select * from student
--Select * from student_marks;
--How to find the marks of student whose name is ‘Anil’ and lives in ‘Delhi’

select * from Student_Marks WHERE RollNo =
    (Select RollNo FROM Student WHERE  City='Delhi')
```

	RollNO	Science	Math	Eng
1	1	34	78	54

Sub Query- Multiple Row Return example

- A **IN** operator is used for multiple row return.

Question: How to find the marks of student who lives in '**Noida**'



```
Select * from Student_Marks
WHERE RollNo IN
```

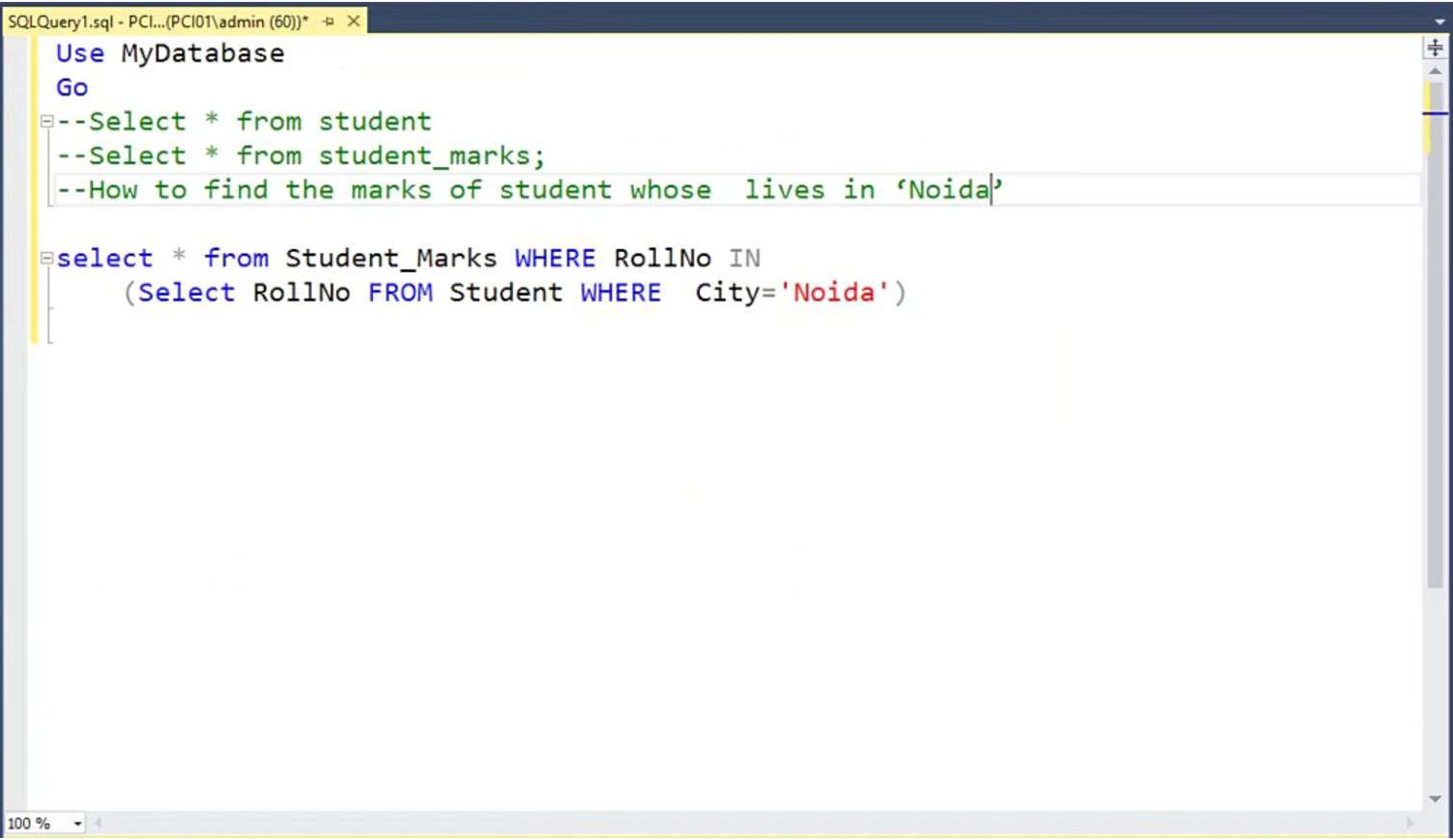
```
(Select RollNo FROM Student
WHERE City='Noida');
```

Student_Marks

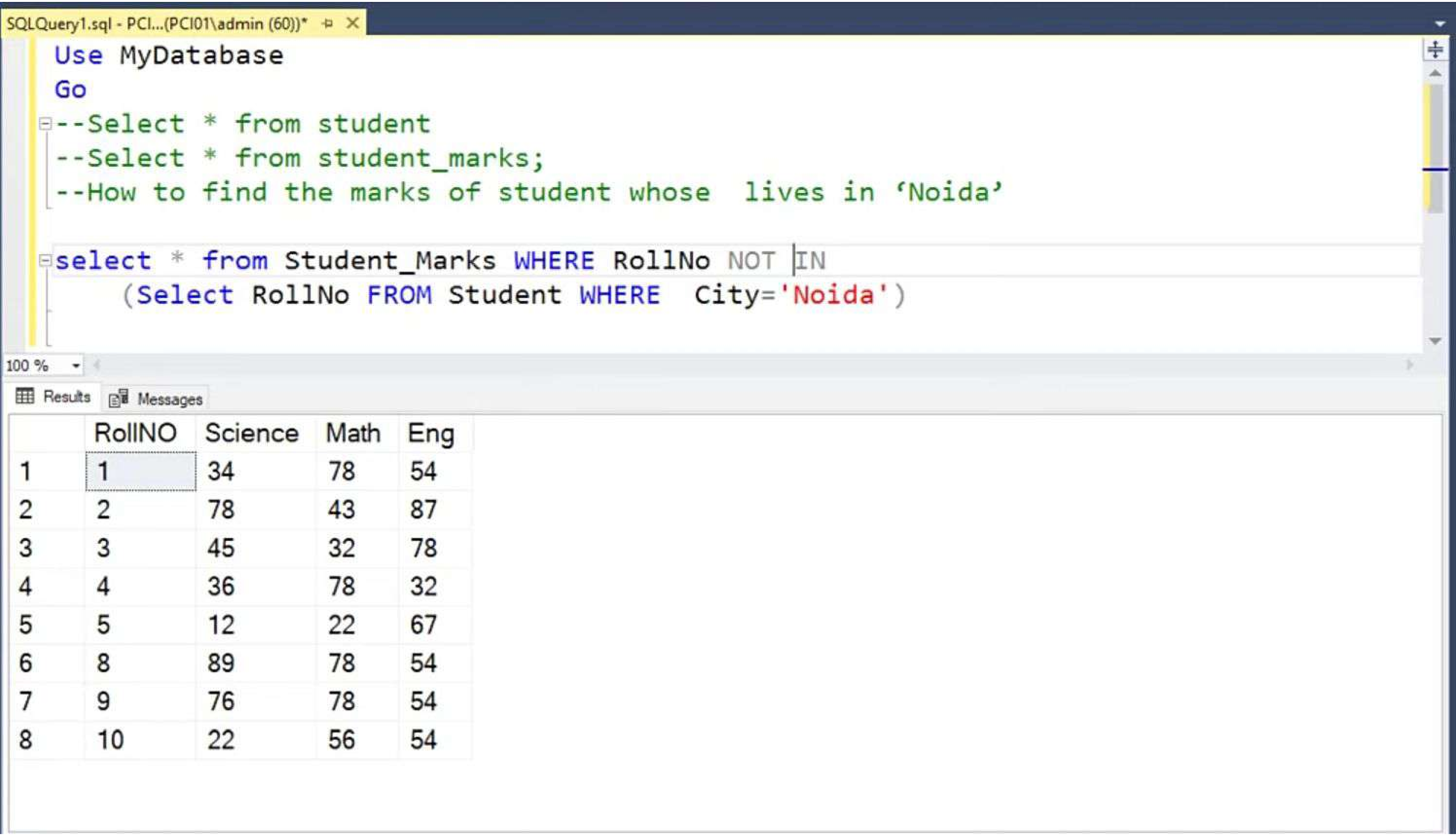
RollNO	Science	Math	Eng
1	34	78	54
2	78	43	87
3	45	32	78
4	36	78	32
5	12	22	67
6	21	65	43
7	34	78	54
8	89	78	54
9	76	78	54
10	22	56	54

Student

RollNO	Student_Name	city
1	Anil	Delhi
2	Sunl	GZB
3	Ajay	Delhi
4	Vijay	Calcutta
5	Manoj	Banglore
6	Geeta	Noida
7	Sita	Noida
8	Reeta	Faridabad
9	Arvind	NCR
10	Kumar	New Delhi



```
Use MyDatabase
Go
--Select * from student
--Select * from student_marks;
--How to find the marks of student whose  lives in ‘Noida’

select * from Student_Marks WHERE RollNo IN
    (Select RollNo FROM Student WHERE  City='Noida')
```

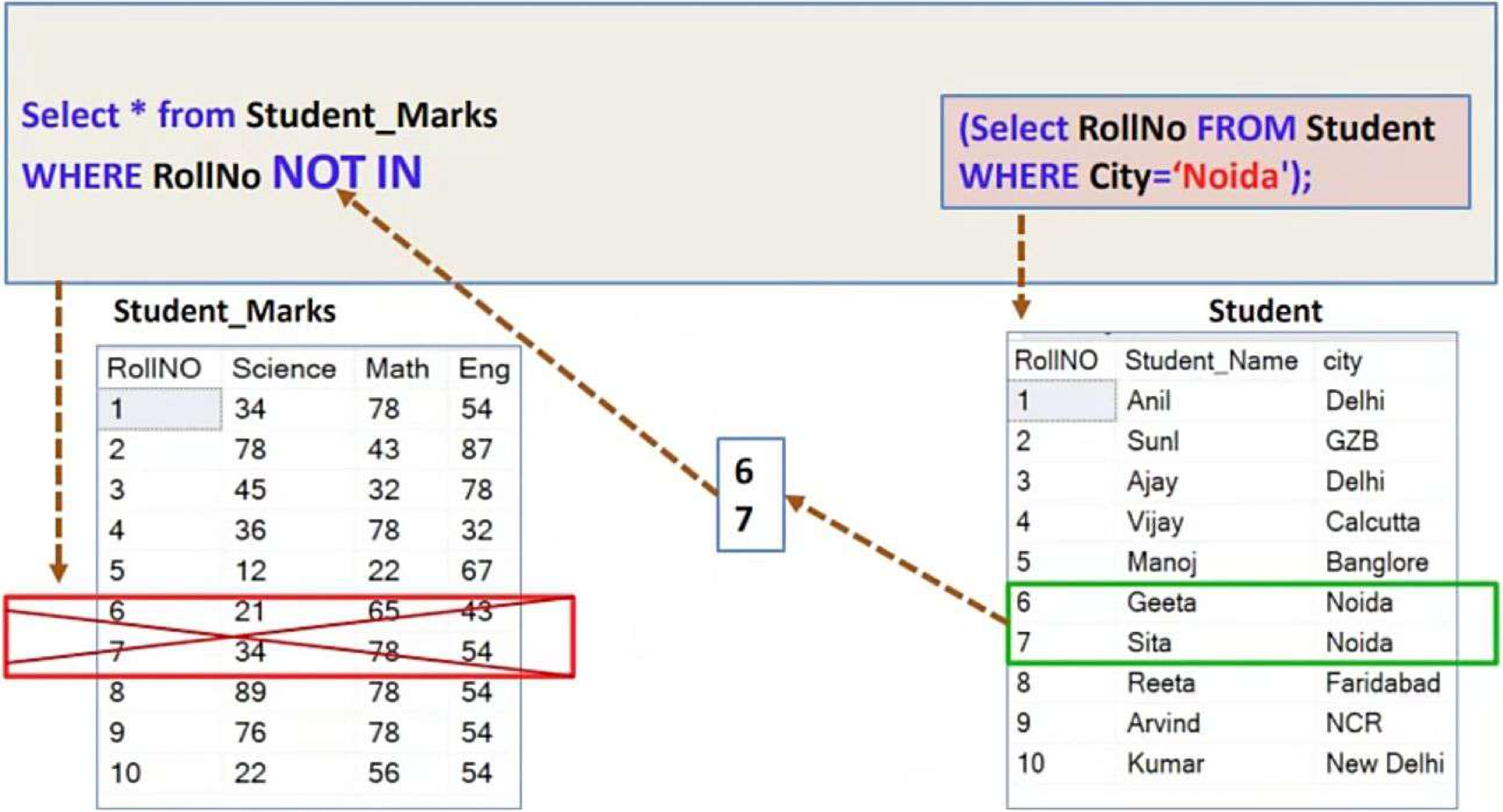


```
Use MyDatabase
Go
--Select * from student
--Select * from student_marks;
--How to find the marks of student whose  lives in ‘Noida’

select * from Student_Marks WHERE RollNo NOT IN
    (Select RollNo FROM Student WHERE  City='Noida')
```

	RollNO	Science	Math	Eng
1	1	34	78	54
2	2	78	43	87
3	3	45	32	78
4	4	36	78	32
5	5	12	22	67
6	8	89	78	54
7	9	76	78	54
8	10	22	56	54

Sub Query- Multiple Row Return example

- Sub query may return one or more rows value.
- A **NOT IN** operator is used for multiple row return.



```
Select * from Student_Marks
WHERE RollNo NOT IN
```

```
(Select RollNo FROM Student
WHERE City='Noida');
```

Student_Marks

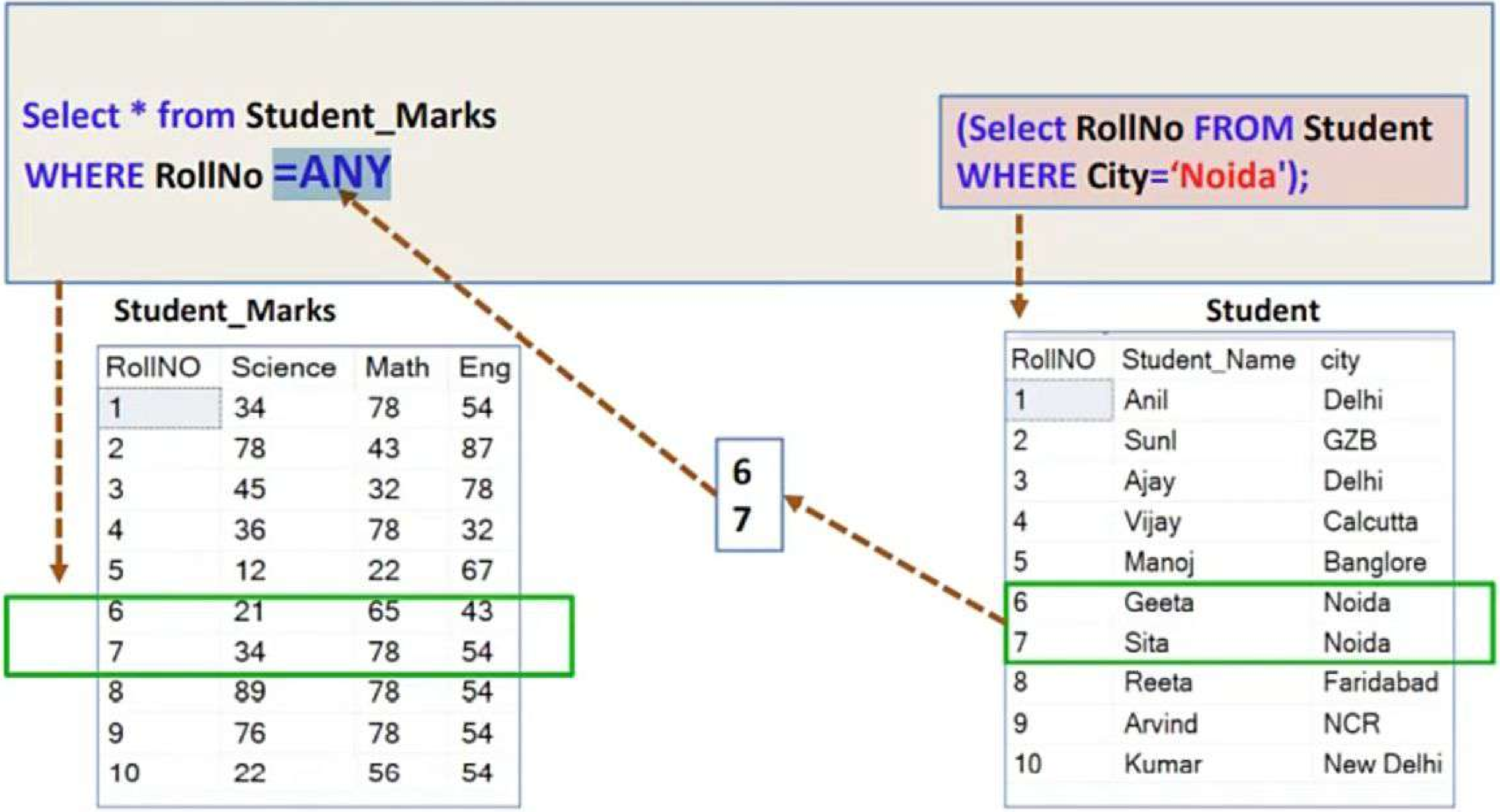
RollNO	Science	Math	Eng
1	34	78	54
2	78	43	87
3	45	32	78
4	36	78	32
5	12	22	67
~~6~~	~~21~~	~~65~~	~~43~~
~~7~~	~~34~~	~~78~~	~~54~~
8	89	78	54
9	76	78	54
10	22	56	54

Student

RollNO	Student_Name	city
1	Anil	Delhi
2	Sunl	GZB
3	Ajay	Delhi
4	Vijay	Calcutta
5	Manoj	Banglore
6	Geeta	Noida
7	Sita	Noida
8	Reeta	Faridabad
9	Arvind	NCR
10	Kumar	New Delhi

Sub Query- Multiple Row Return example

- ➢ The ANY operator returns a Boolean value as a result, must be used with comparison operator.
- ➢ It returns TRUE if ANY of the sub query values meet the condition
- ➢ It is similar to IN operator



```
Select * from Student_Marks
WHERE RollNo =ANY
```

```
(Select RollNo FROM Student
WHERE City='Noida');
```

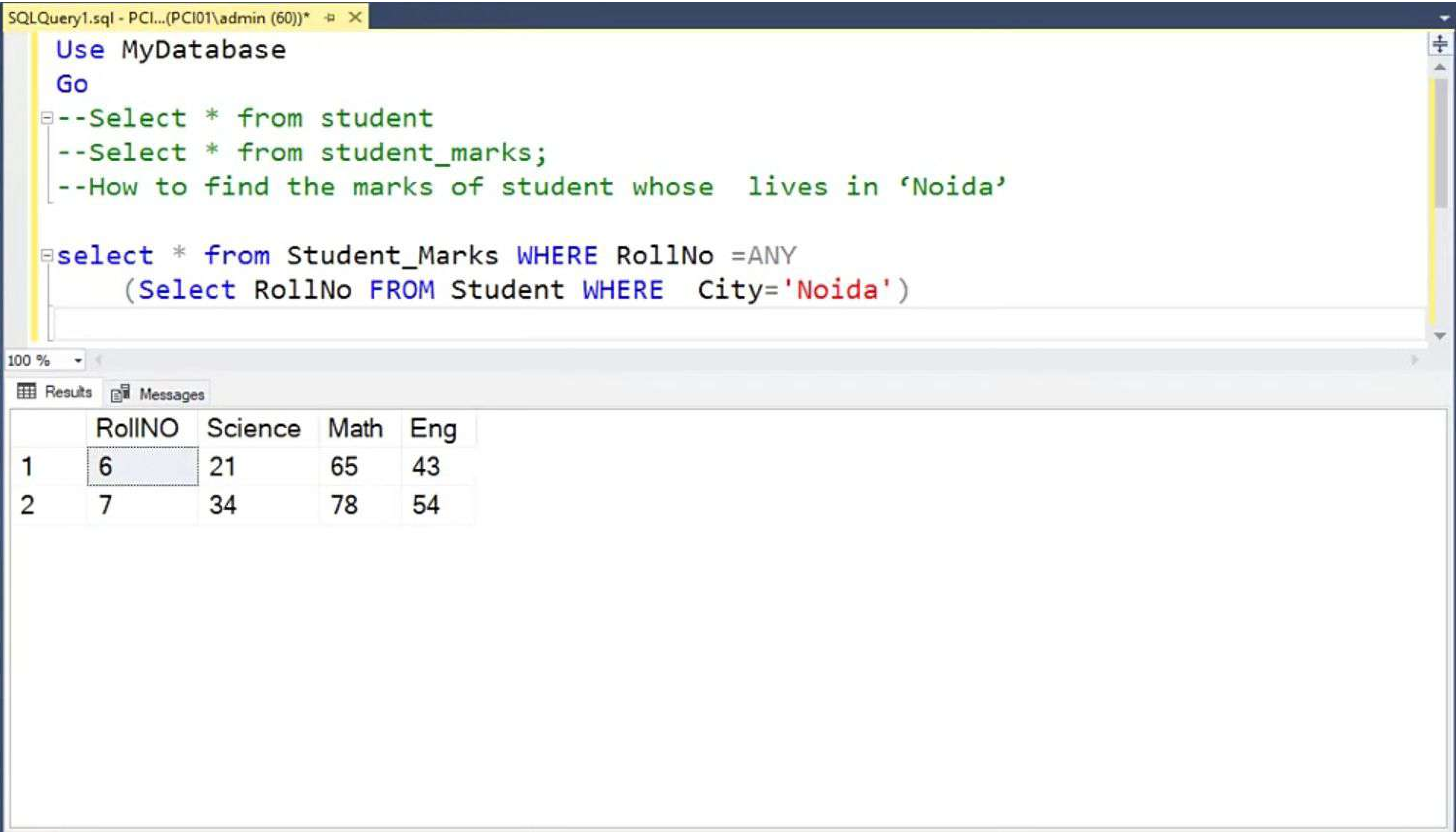
6
7

Student_Marks

RollNO	Science	Math	Eng
1	34	78	54
2	78	43	87
3	45	32	78
4	36	78	32
5	12	22	67
6	21	65	43
7	34	78	54
8	89	78	54
9	76	78	54
10	22	56	54

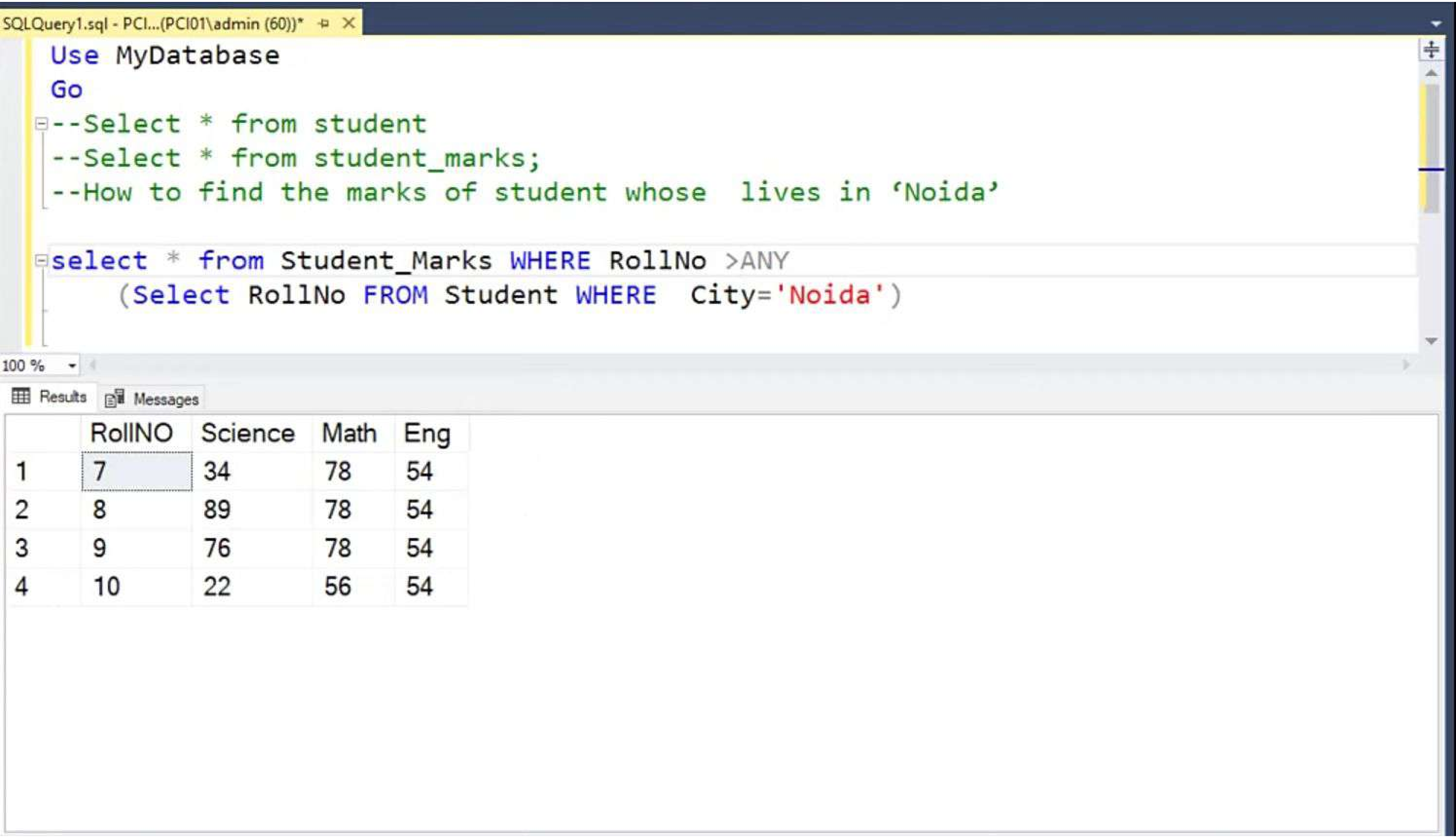
Student

RollNO	Student_Name	city
1	Anil	Delhi
2	Sunl	GZB
3	Ajay	Delhi
4	Vijay	Calcutta
5	Manoj	Banglore
6	Geeta	Noida
7	Sita	Noida
8	Reeta	Faridabad
9	Arvind	NCR
10	Kumar	New Delhi



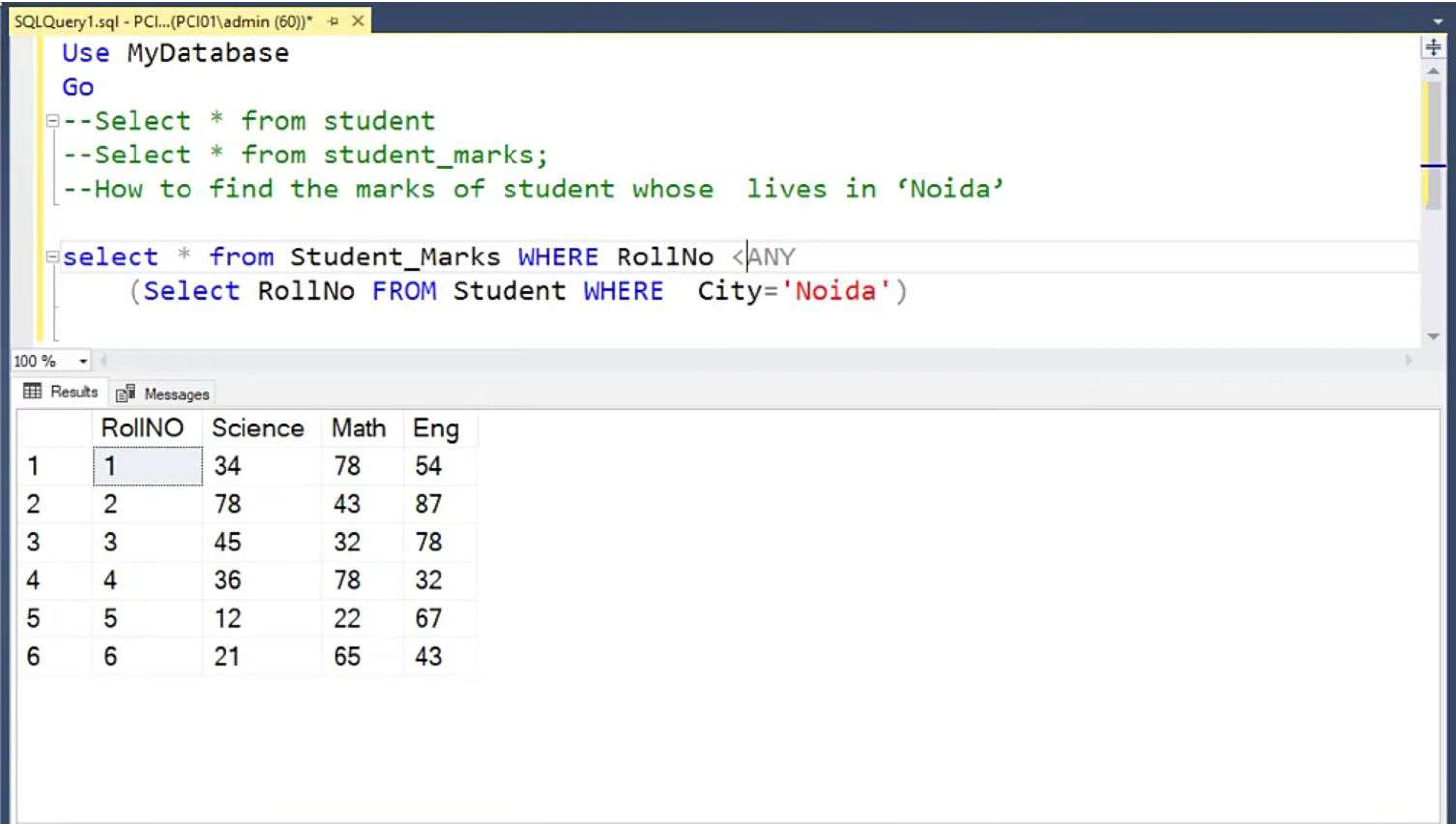
```
Use MyDatabase
Go
--Select * from student
--Select * from student_marks;
--How to find the marks of student whose  lives in ‘Noida’

select * from Student_Marks WHERE RollNo =ANY
    (Select RollNo FROM Student WHERE  City='Noida')
```

	RollNO	Science	Math	Eng
1	6	21	65	43
2	7	34	78	54



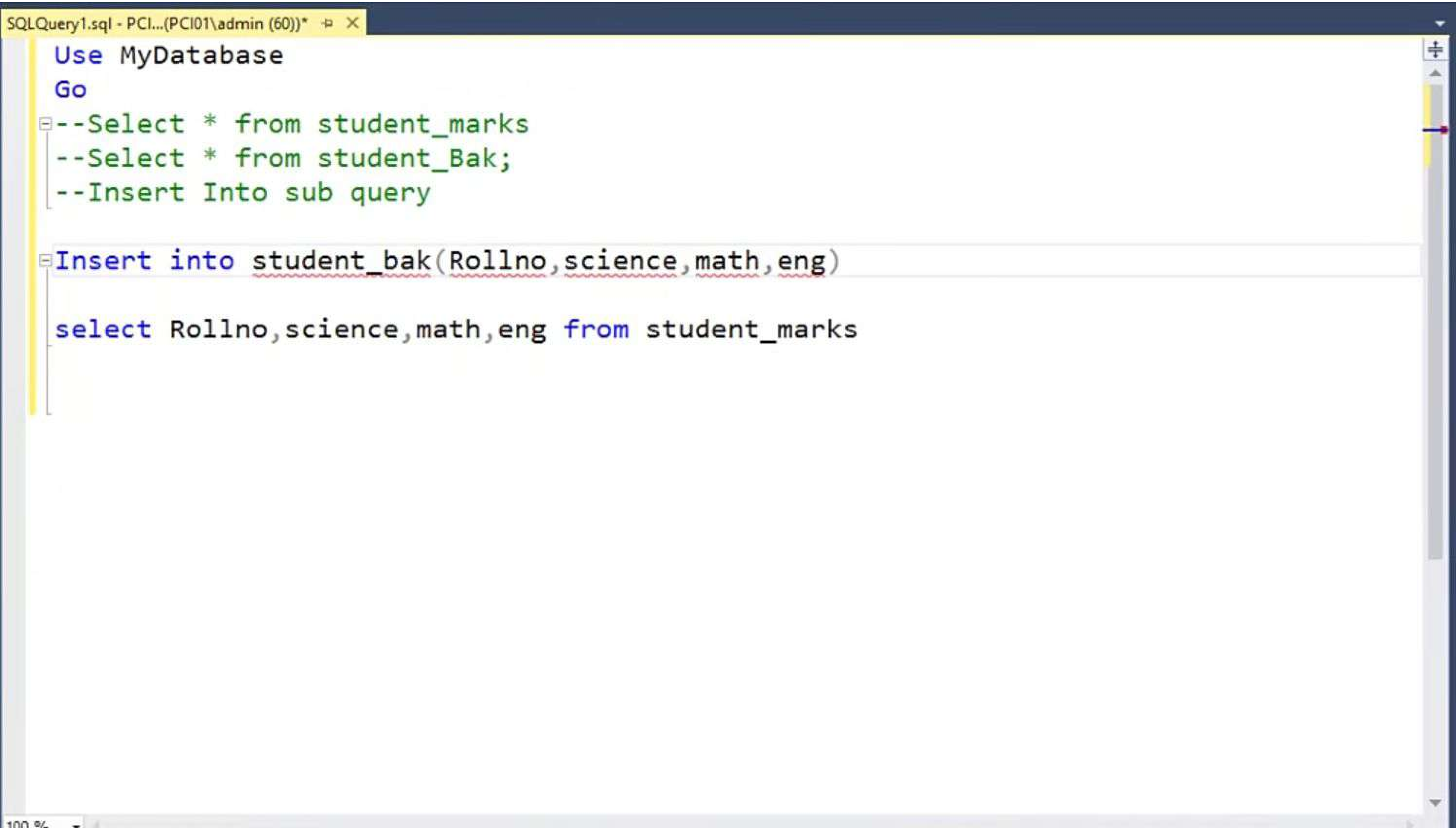
```
Use MyDatabase
Go
--Select * from student
--Select * from student_marks;
--How to find the marks of student whose  lives in ‘Noida’

select * from Student_Marks WHERE RollNo >ANY
    (Select RollNo FROM Student WHERE  City='Noida')
```

	RollNO	Science	Math	Eng
1	7	34	78	54
2	8	89	78	54
3	9	76	78	54
4	10	22	56	54

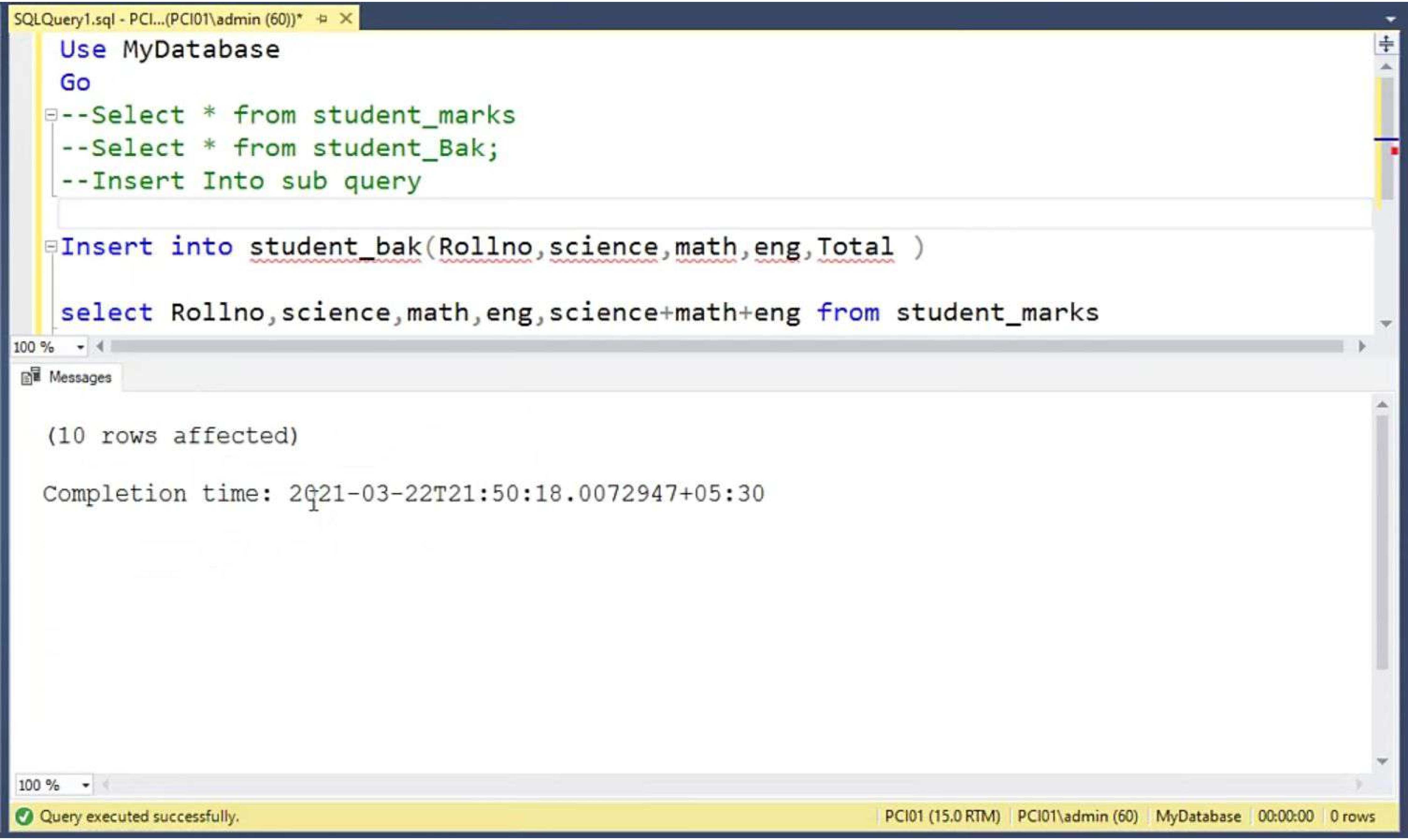


```
Use MyDatabase
Go
--Select * from student
--Select * from student_marks;
--How to find the marks of student whose  lives in ‘Noida’

select * from Student_Marks WHERE RollNo <ANY
    (Select RollNo FROM Student WHERE  City='Noida')
```

	RollNO	Science	Math	Eng
1	1	34	78	54
2	2	78	43	87
3	3	45	32	78
4	4	36	78	32
5	5	12	22	67
6	6	21	65	43

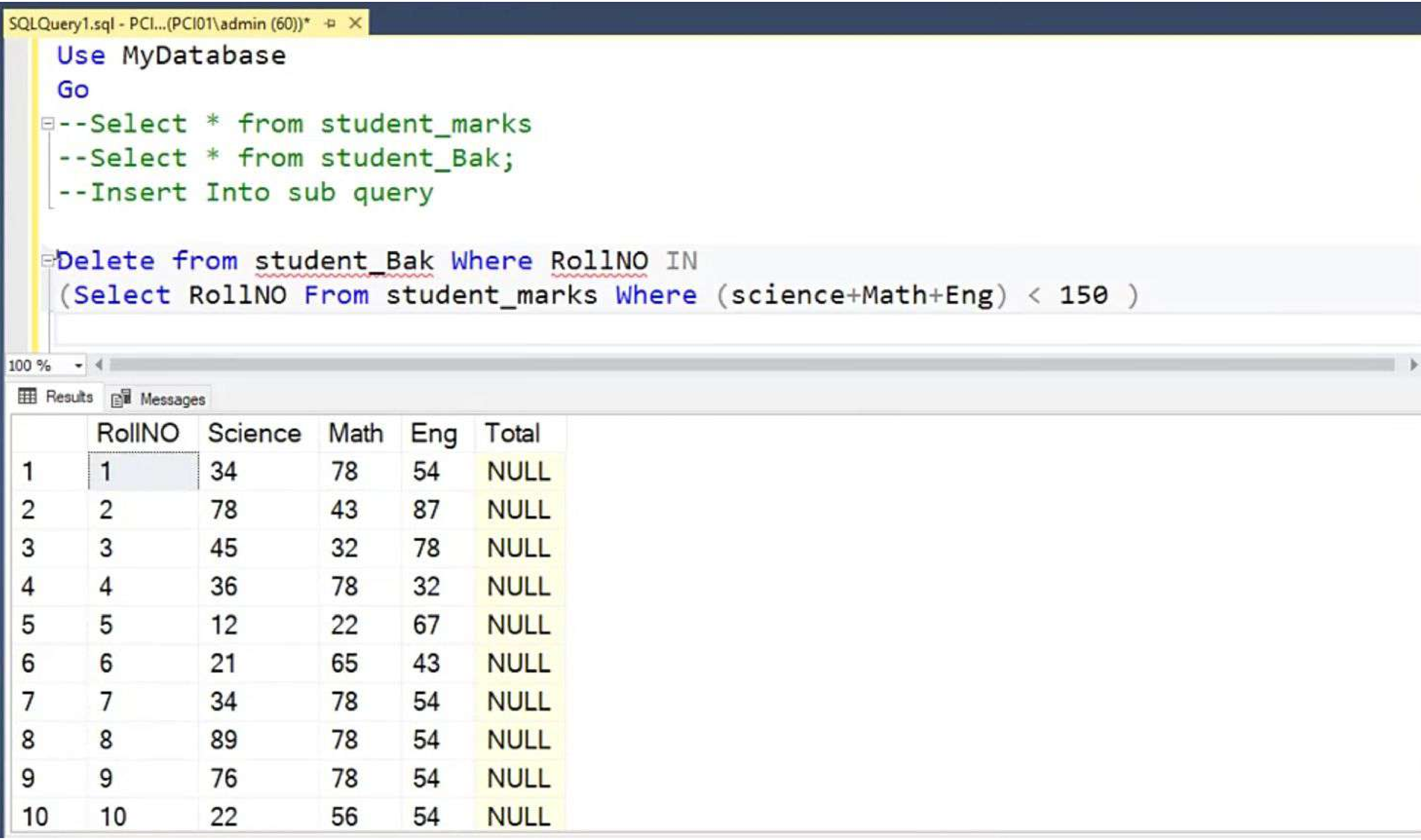


```
Use MyDatabase
Go
--Select * from student_marks
--Select * from student_Bak;
--Insert Into sub query

Insert into student_bak(Rollno,science,math,eng)

select Rollno,science,math,eng from student_marks
```



```sql
Use MyDatabase
Go
--Select * from student_marks
--Select * from student_Bak;
--Insert Into sub query

Insert into student_bak(Rollno,science,math,eng,Total )

select Rollno,science,math,eng,science+math+eng from student_marks
```

```
(10 rows affected)

Completion time: 2021-03-22T21:50:18.0072947+05:30
```

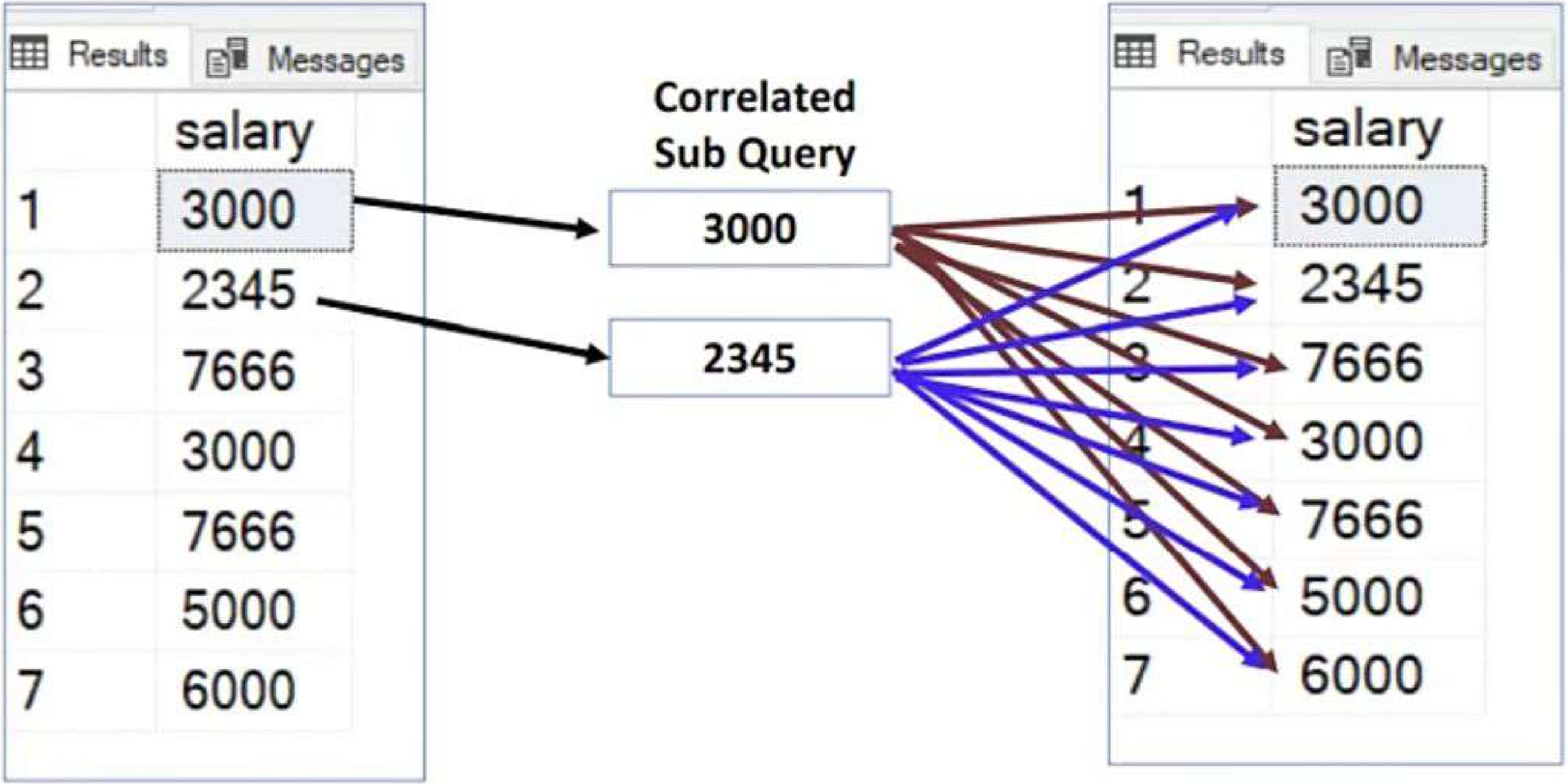


```
Use MyDatabase
Go
--Select * from student_marks
--Select * from student_Bak;
--Insert Into sub query

Delete from student_Bak Where RollNO IN
(Select RollNO From student_marks Where (science+Math+Eng) < 150 )
```

	RollNO	Science	Math	Eng	Total
1	1	34	78	54	NULL
2	2	78	43	87	NULL
3	3	45	32	78	NULL
4	4	36	78	32	NULL
5	5	12	22	67	NULL
6	6	21	65	43	NULL
7	7	34	78	54	NULL
8	8	89	78	54	NULL
9	9	76	78	54	NULL
10	10	22	56	54	NULL

Using Correlated sub query

Correlated sub queries are used for row-by-row processing. Each sub query is executed once for every row of the outer query.



	salary
1	3000
2	2345
3	7666
4	3000
5	7666
6	5000
7	6000

Correlated Sub Query

3000

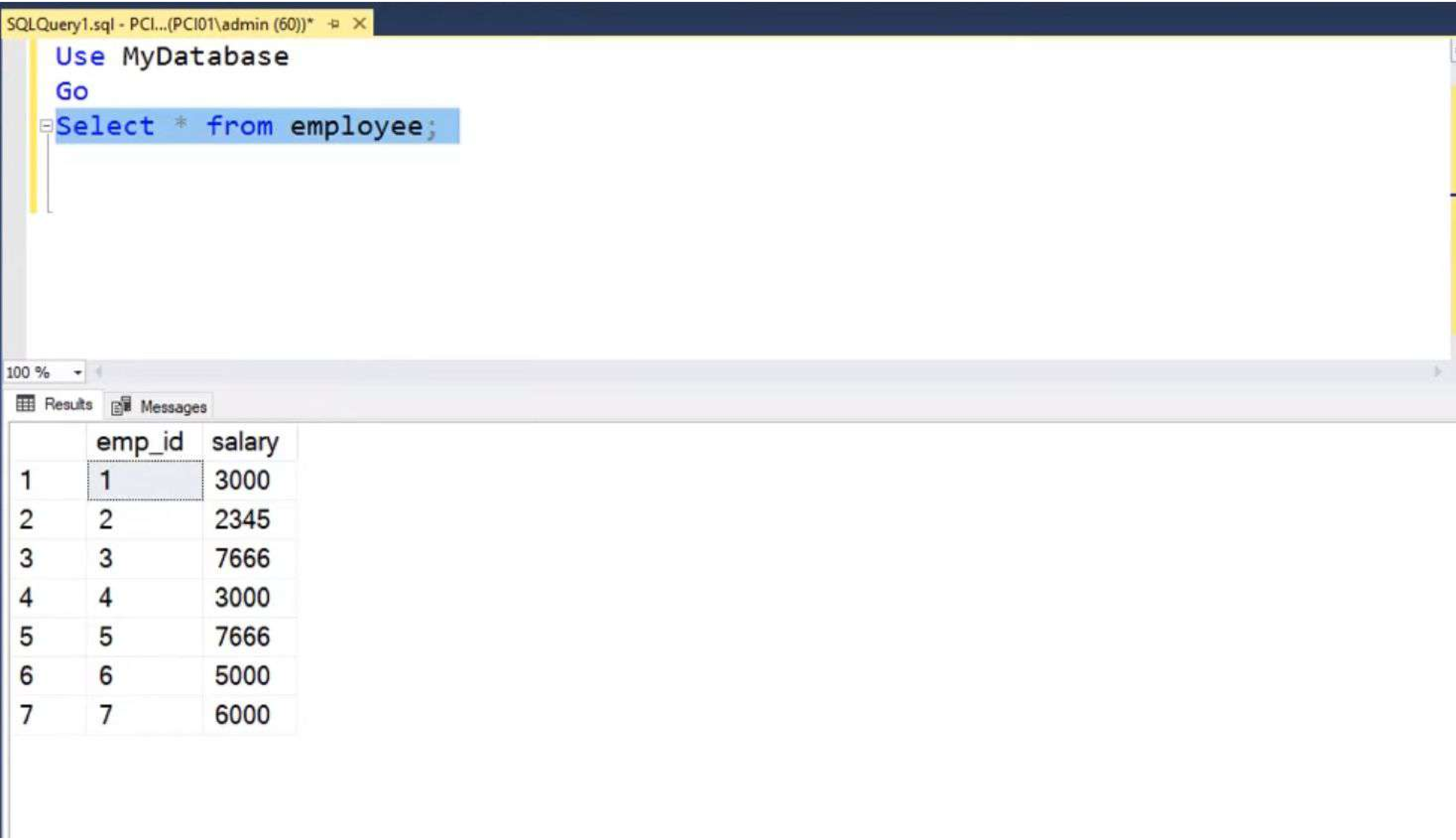
2345

	salary
1	3000
2	2345
3	7666
4	3000
5	7666
6	5000
7	6000

How Correlated sub query work ?

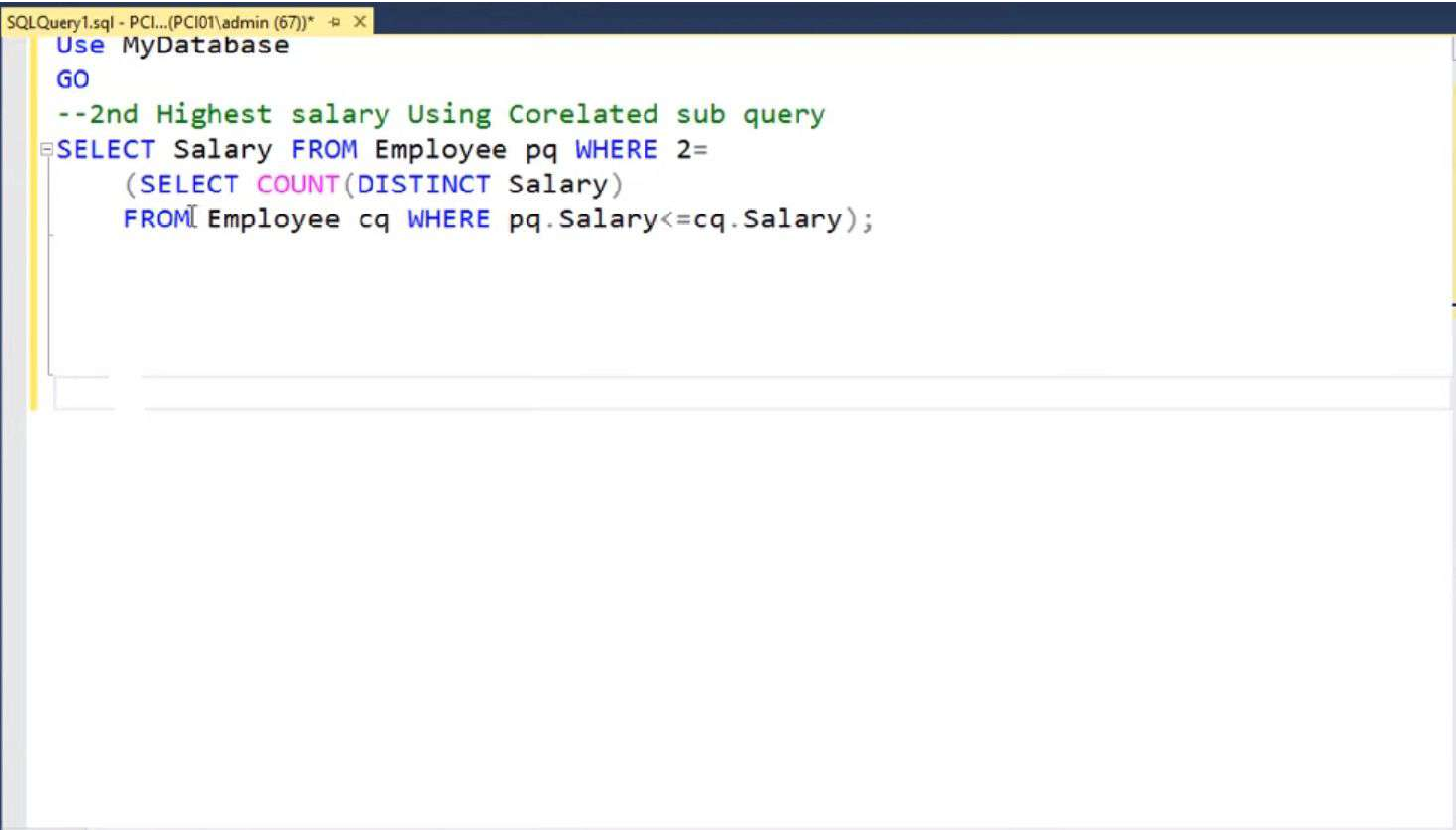
Parent Row Value	Operator	Numbers less than Parent row value	Distinct Count
10	<=	10 10	1
8	<=	10 10 8	2
7	<=	10 10 8 7	3
6	<=	10 10 8 7 6	4

Emp_id
1
2
3
4
5
6
7
8
10
10



```sql
Use MyDatabase
Go
Select * from employee;
```

	emp_id	salary
1	1	3000
2	2	2345
3	3	7666
4	4	3000
5	5	7666
6	6	5000
7	7	6000



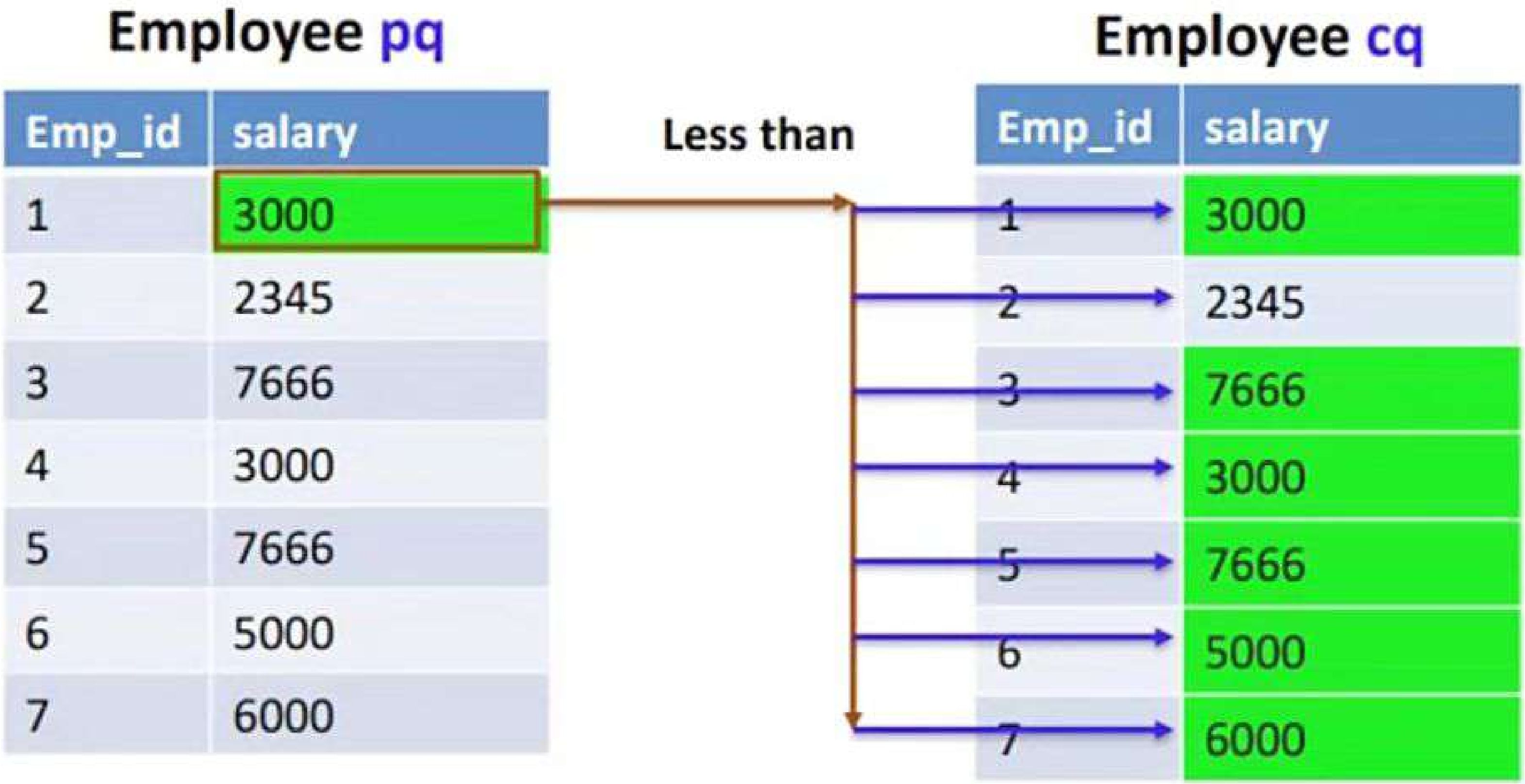
```
Use MyDatabase
GO
--2nd Highest salary Using Corelated sub query
SELECT Salary FROM Employee pq WHERE 2=
    (SELECT COUNT(DISTINCT Salary)
    FROM Employee cq WHERE pq.Salary<=cq.Salary);
```

Using Correlated sub query

- **2nd Highest Salary**

```
SELECT  Salary FROM Employee pq WHERE 2=(SELECT COUNT(DISTINCT Salary)
FROM Employee cq WHERE pq.Salary<=cq.Salary);
```

Step-1



Employee pq

Emp_id	salary
1	3000
2	2345
3	7666
4	3000
5	7666
6	5000
7	6000

Less than

Employee cq

Emp_id	salary
1	3000
2	2345
3	7666
4	3000
5	7666
6	5000
7	6000

4th Highest Salary

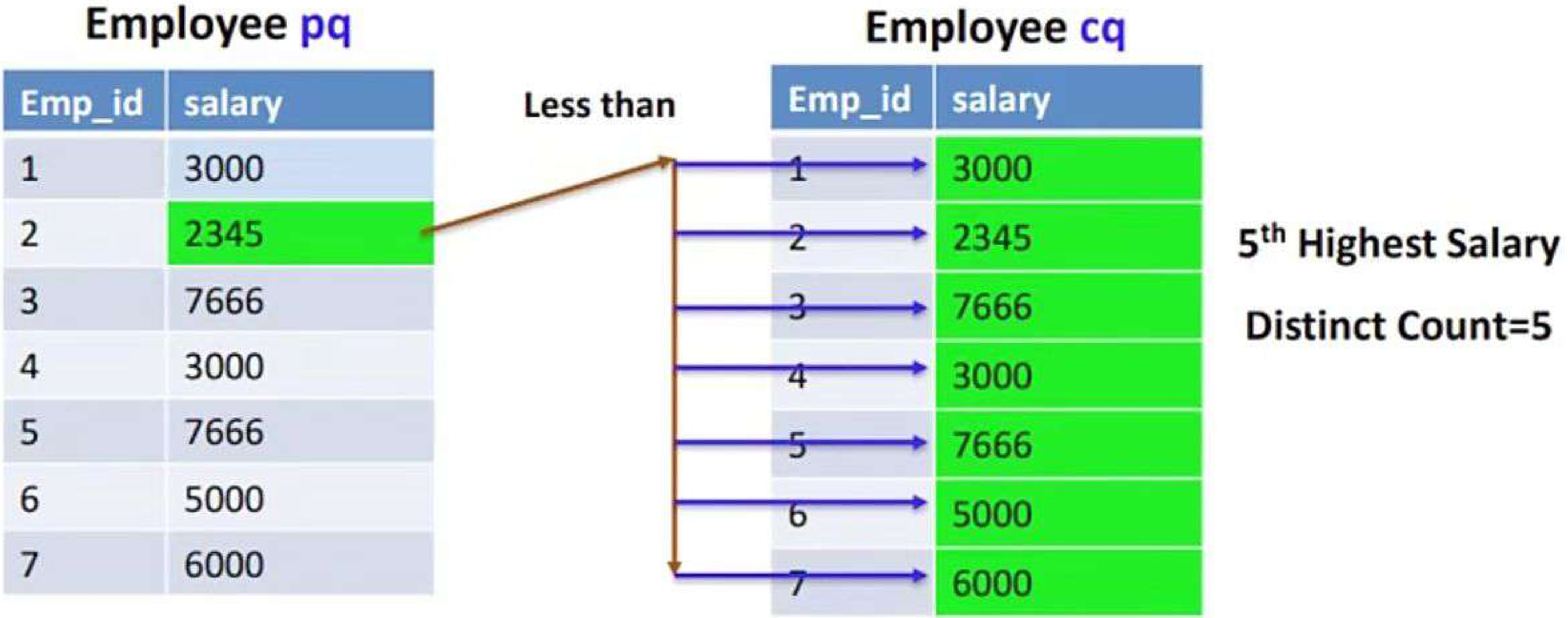
Distinct Count=4

Using Correlated sub query

➢ 2nd Highest Salary

```
SELECT  Salary FROM Employee pq WHERE 2=(SELECT COUNT(DISTINCT Salary)

FROM Employee cq WHERE pq.Salary<=cq.Salary);
```

Step-2



Employee pq

Emp_id	salary
1	3000
2	2345
3	7666
4	3000
5	7666
6	5000
7	6000

Less than

Employee cq

Emp_id	salary
1	3000
2	2345
3	7666
4	3000
5	7666
6	5000
7	6000

5th Highest Salary

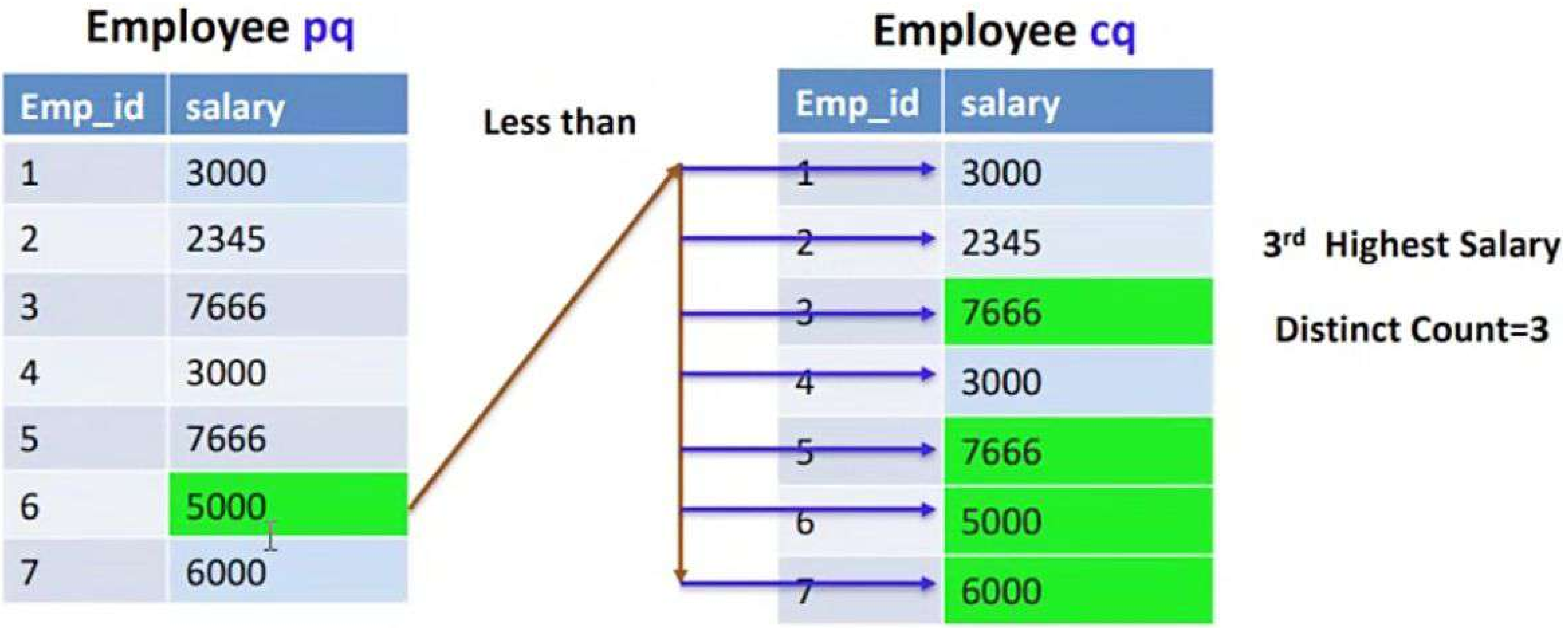
Distinct Count=5

Using Correlated sub query

- 2nd Highest Salary

```
SELECT  Salary FROM Employee pq WHERE 2=(SELECT COUNT(DISTINCT Salary)
FROM Employee cq WHERE pq.Salary<=cq.Salary);
```

Step-6



Employee pq

Emp_id	salary
1	3000
2	2345
3	7666
4	3000
5	7666
6	5000
7	6000

Less than

Employee cq

Emp_id	salary
1	3000
2	2345
3	7666
4	3000
5	7666
6	5000
7	6000

3rd Highest Salary

Distinct Count=3