LABORATORY REPORT

LABORATORY REPORT

Common Data	
Student name	Das Anjan Kumar
Neptun code	D42DQA
Department	Dept. of Automation and Applied Informatics
Instructor name	AL MAGSOOSI HUSAM
Laboratory place	IL206
Laboratory time	23rd March,2022, 10.15-11.45
Title or sequence number	3

Exercises	
Task 1	□
Task 2	□
Task 3	□
Task 4	□
Task 5	□
Task 6	□
Task 7	□
Task 8	□
Task 9	□

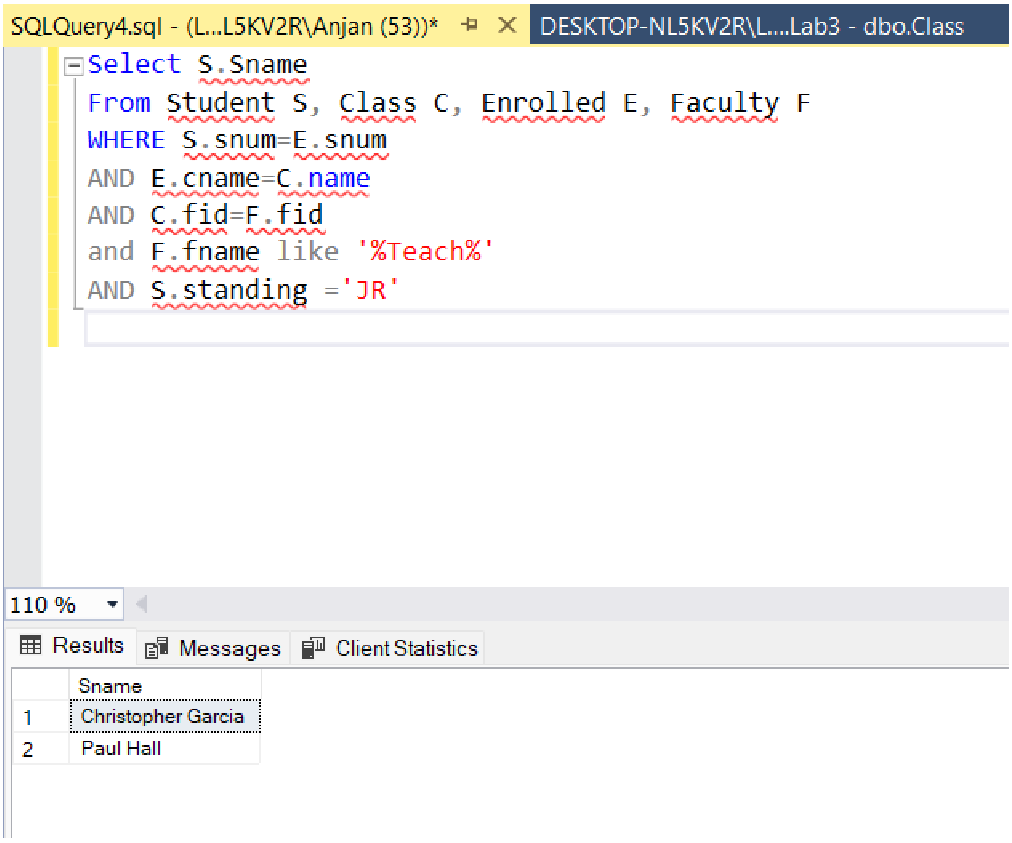
EXERCISES

TASK #1

Problem statement: Find the names of all Juniors (level = JR) who are enrolled in a class taught by I. Teach

Solution:

```
Select S.Sname /*as we only need the students name*/
From Student S, Class C, Enrolled E, Faculty F /*as we will connect the foreign keys
so we are aliasing the table names*/
WHERE S.snum=E.snum /*now we should match the foreign keys of each table*/
AND E.cname=C.name
AND C.fid=F.fid
and F.fname like '%Teach%' /*as we dont know the full name of the teacher*/
AND S.standing ='JR'/*finding students with that level*/
```



SQLQuery4.sql - (L...L5KV2R\Anjan (53))*
DESKTOP-NL5KV2R\L....Lab3 - dbo.Class
110 %
Results
Messages
Client Statistics

```
Select S.Sname
From Student S, Class C, Enrolled E, Faculty F
WHERE S.snum=E.snum
AND E.cname=C.name
AND C.fid=F.fid
and F.fname like '%Teach%'
AND S.standing ='JR'
```

	Sname
1	Christopher Garcia
2	Paul Hall

Reasoning:

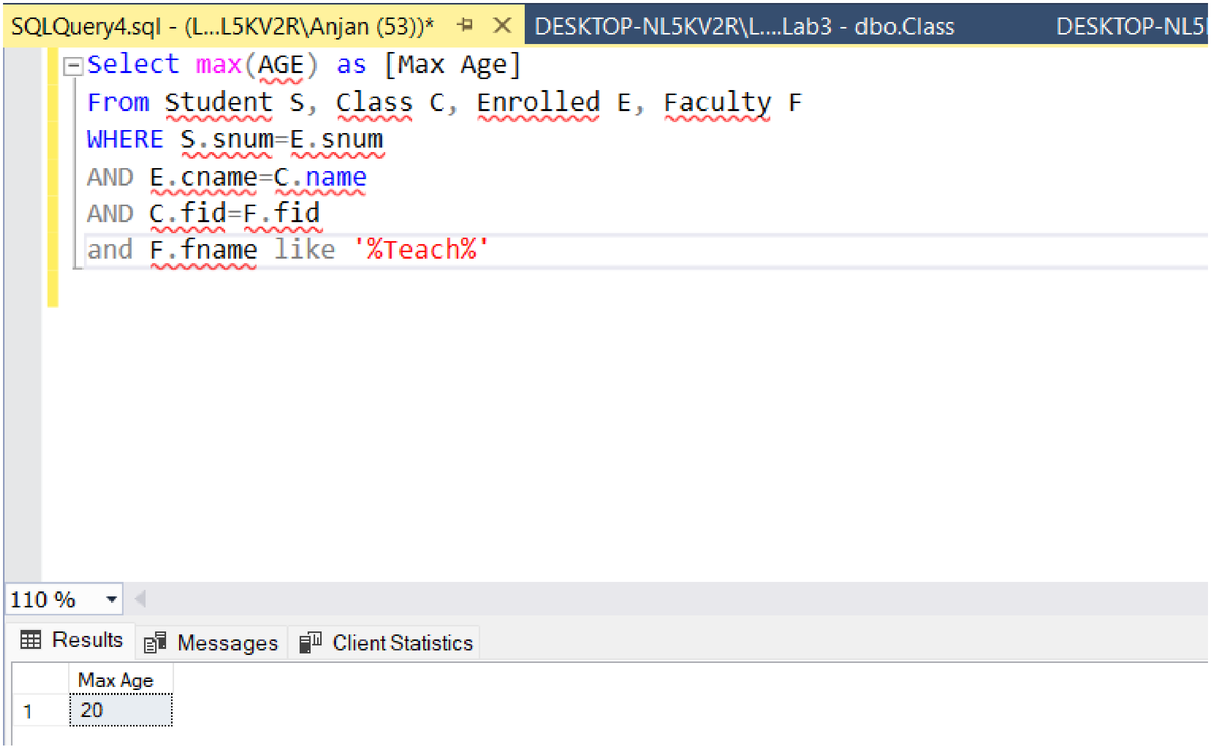
As we need to match data from all the tables and connect them with their foreign keys of each connected table, that's why we mentioned all the tables and aliased them. Then we used the LIKE keyword as we don't know the full name of the teacher, and lastly use proper syntax for finding JR level students. And finally, we got the expected result.

TASK #2

Problem statement: Find the age of the oldest student who is either a History major or enrolled in a course taught by I. Teach. Rename the resulting column to “Max Age”.

Solution:

```
 Select max(AGE) as [Max Age] /*using the max function to find the maximum age*/
From Student S, Class C, Enrolled E, Faculty F /*mentioning the tables we need to
connect using foreign keys*/
WHERE S.snum=E.snum /*now we need to find the matching foreign keys*/
AND E.cname=C.name
AND C.fid=F.fid
and F.fname like '%Teach%' /*as we don't know the full name of the teacher*/
```


SQLQuery4.sql - (L...L5KV2R\Anjan (53))*
DESKTOP-NL5KV2R\L....Lab3 - dbo.Class
DESKTOP-NL5
110 %
Results
Messages
Client Statistics

```
Select max(AGE) as [Max Age]
From Student S, Class C, Enrolled E, Faculty F
WHERE S.snum=E.snum
AND E.cname=C.name
AND C.fid=F.fid
and F.fname like '%Teach%'
```

	Max Age
1	20

Reasoning:

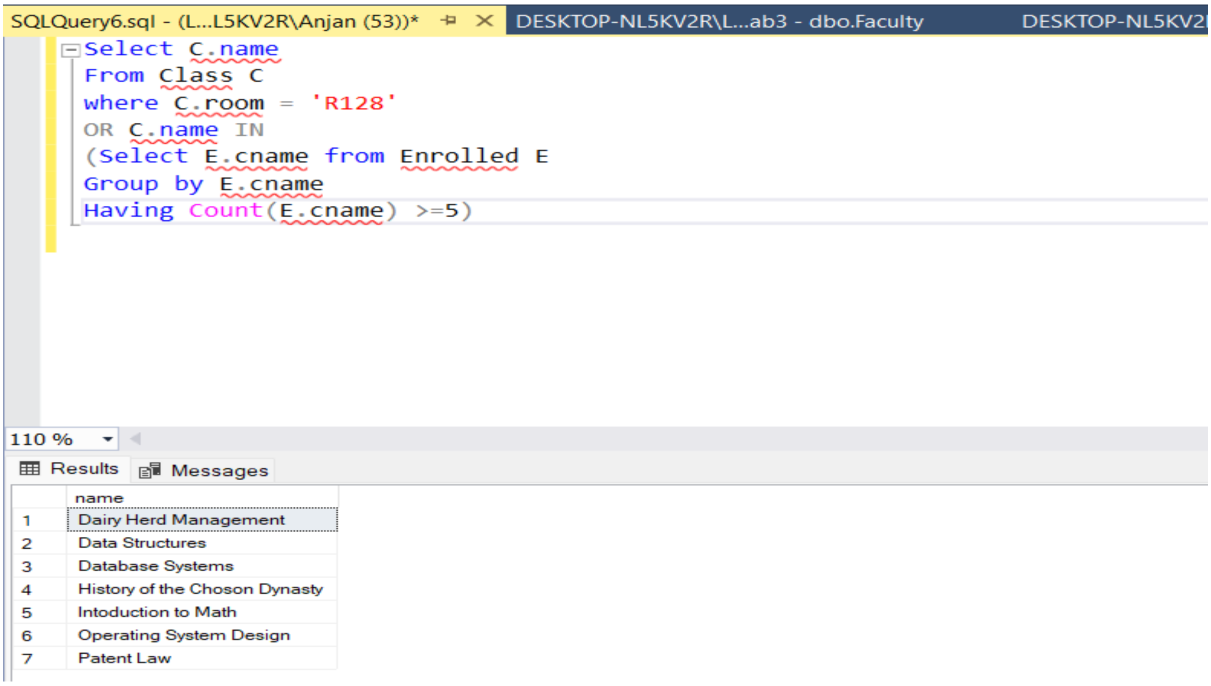
In order to find the students with maximum age we used the proper aggregate function and then connected the tables with proper foreign keys to extract the proper information. And also, as the previous task we don’t know the full teacher name that’s why we used the like functionality. And finally got the proper result.

TASK #3

Problem statement: Find the names of all classes that either meet in room R128 or have five or more students enrolled

Solution:

```
Select C.name
From Class C  /*we need to find the name of courses from class table*/
where C.room = 'R128' /*the courses that takes place in that room*/
OR C.name IN  /*also we have to meet another condition for that we will use a nested 
loop*/
(Select E.cname from Enrolled E
Group by E.cname
Having Count(E.cname) >=5) /*in the course where the number of enrolled students five 
or more*/
```



SQLQuery6.sql - (L...L5KV2R\Anjan (53))*
DESKTOP-NL5KV2R\L...ab3 - dbo.Faculty
DESKTOP-NL5KV2
110 %
Results
Messages

```
Select C.name
From Class C
where C.room = 'R128'
OR C.name IN
(Select E.cname from Enrolled E
Group by E.cname
Having Count(E.cname) >=5)
```

	name
1	Dairy Herd Management
2	Data Structures
3	Database Systems
4	History of the Choson Dynasty
5	Intoduction to Math
6	Operating System Design
7	Patent Law

Reasoning:

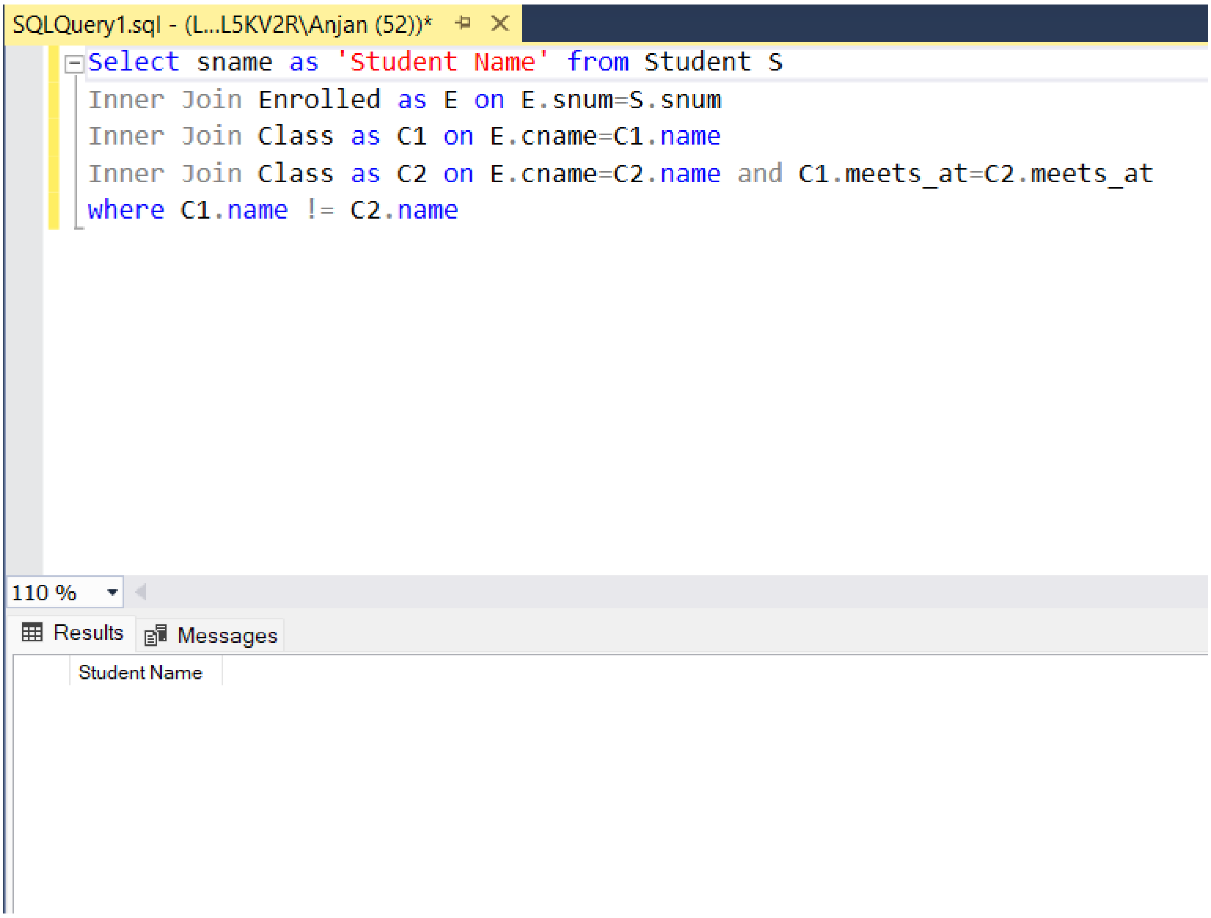
In this task we need to check two conditions, if any one of them is true then our expected result should what we want. To find the proper result we use nested loop for judging both of the conditions, in the first condition we checked the classroom and in the second condition we checked if that course has more than five students or not. Finally, we got the expected result.

TASK #4

Problem statement: Find the names of all students who are enrolled in two different classes that meet at the same time

Solution:

```
Select sname as 'Student Name' from Student S /*we will select students*/
Inner Join Enrolled as E on E.snum=S.snum/*we will  inner join the tables with foreign
keys*/
Inner Join Class as C1 on E.cname=C1.name /*as there have to be two classes that are
attended by the same student*/
Inner Join Class as C2 on E.cname=C2.name and C1.meets_at=C2.meets_at /*checking if
they meet at the same time*/
where C1.name != C2.name
```



```
Select sname as 'Student Name' from Student S
Inner Join Enrolled as E on E.snum=S.snum
Inner Join Class as C1 on E.cname=C1.name
Inner Join Class as C2 on E.cname=C2.name and C1.meets_at=C2.meets_at
where C1.name != C2.name
```

```
Student Name
```

Reasoning:

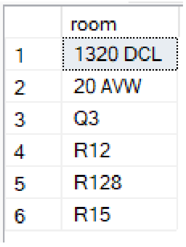
First we join all the necessary tables with proper foreign key connection and check if the name of the two students in two classes are same. But as we know it is not possible to attend two classes at the same time, so we get empty result.

TASK #5

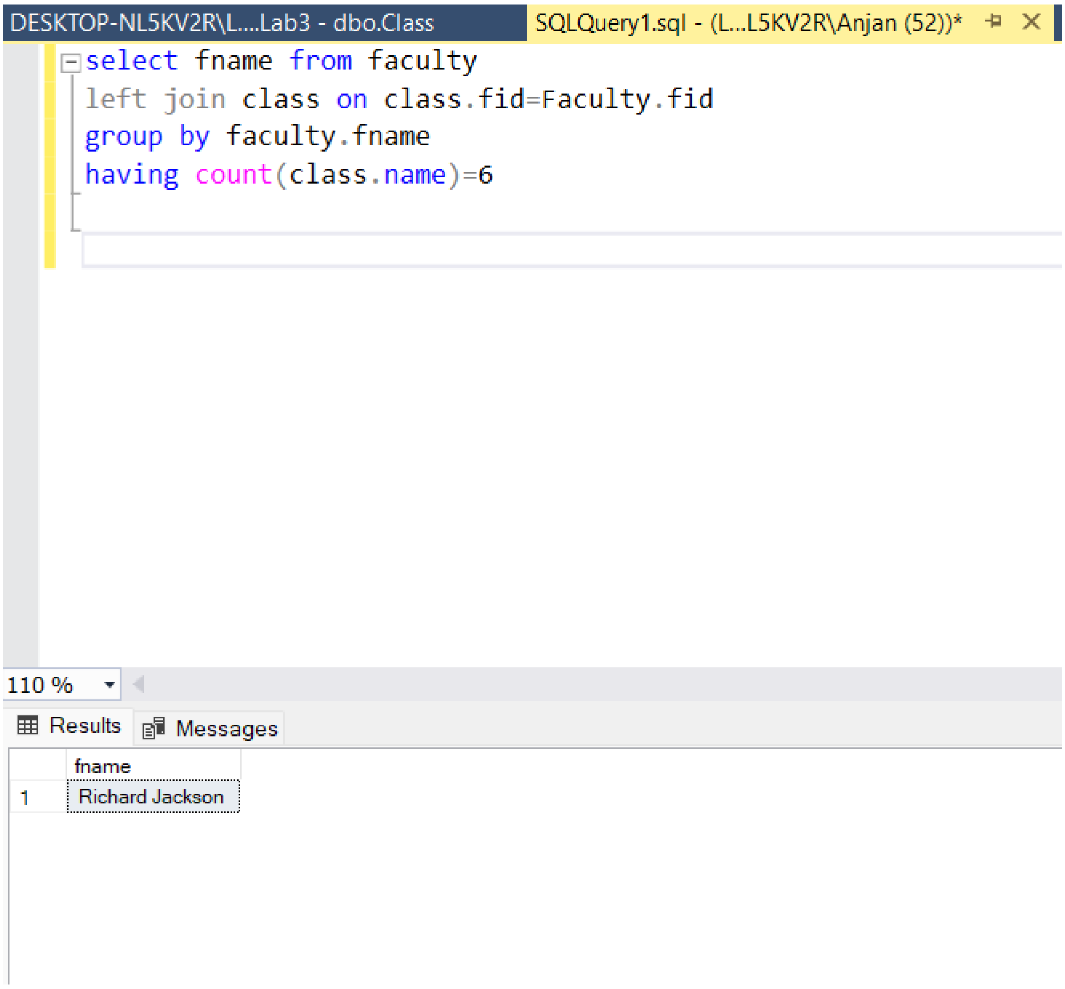
Problem statement: Find the names of faculty members who teach in every room some class is taught

Solution:

```
select distinct room from class
```



	room
1	1320 DCL
2	20 AVW
3	Q3
4	R12
5	R128
6	R15



DESKTOP-NL5KV2R\L....Lab3 - dbo.Class
SQLQuery1.sql - (L...L5KV2R\Anjan (52))*
110 %
Results
Messages

```
select fname from faculty
left join class on class.fid=Faculty.fid
group by faculty.fname
having count(class.name)=6
```

	fname
1	Richard Jackson

Reasoning:

First we check the number of rooms where at least one class is taken. Then we find the faculty who takes classes that number of times in those rooms. And we get the expected result.

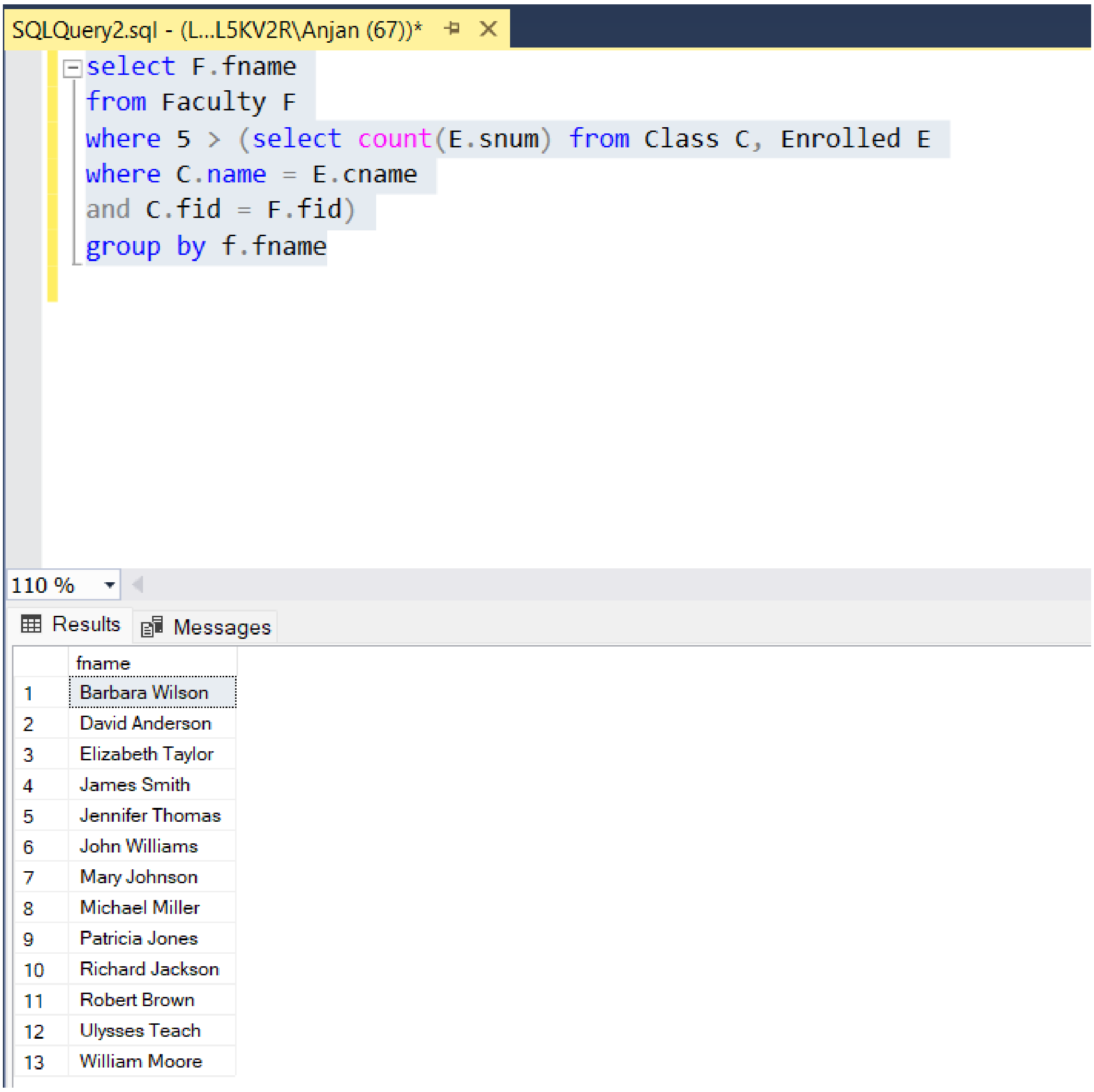
TASK #6

Problem statement: Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five

Solution:

```
select F.fname from Faculty as  F /*as we need to find faculty names*/
where 5 > (select count(E.snum) from Class C, Enrolled E /*and the condition is their
combined
enrollment of the courses that they teach is less than five*/
where C.name = E.cname /*matching with the proper foreign keys*/
and C.fid = F.fid)

group by f.fname/*using this to sort according their names*/
```



```
select F.fname
from Faculty F
where 5 > (select count(E.snum) from Class C, Enrolled E
where C.name = E.cname
and C.fid = F.fid)
group by f.fname
```

	fname
1	Barbara Wilson
2	David Anderson
3	Elizabeth Taylor
4	James Smith
5	Jennifer Thomas
6	John Williams
7	Mary Johnson
8	Michael Miller
9	Patricia Jones
10	Richard Jackson
11	Robert Brown
12	Ulysses Teach
13	William Moore

Reasoning:

Here we needed to find the teachers whose combined enrollment of the courses that they teach is less than five. For that we needed to use a nested loop. Because we need to find the enrolled students in class and enrolled table and finally attaching it with the data of faculty. Lastly we use group by to sort. Finally we get the expected result.

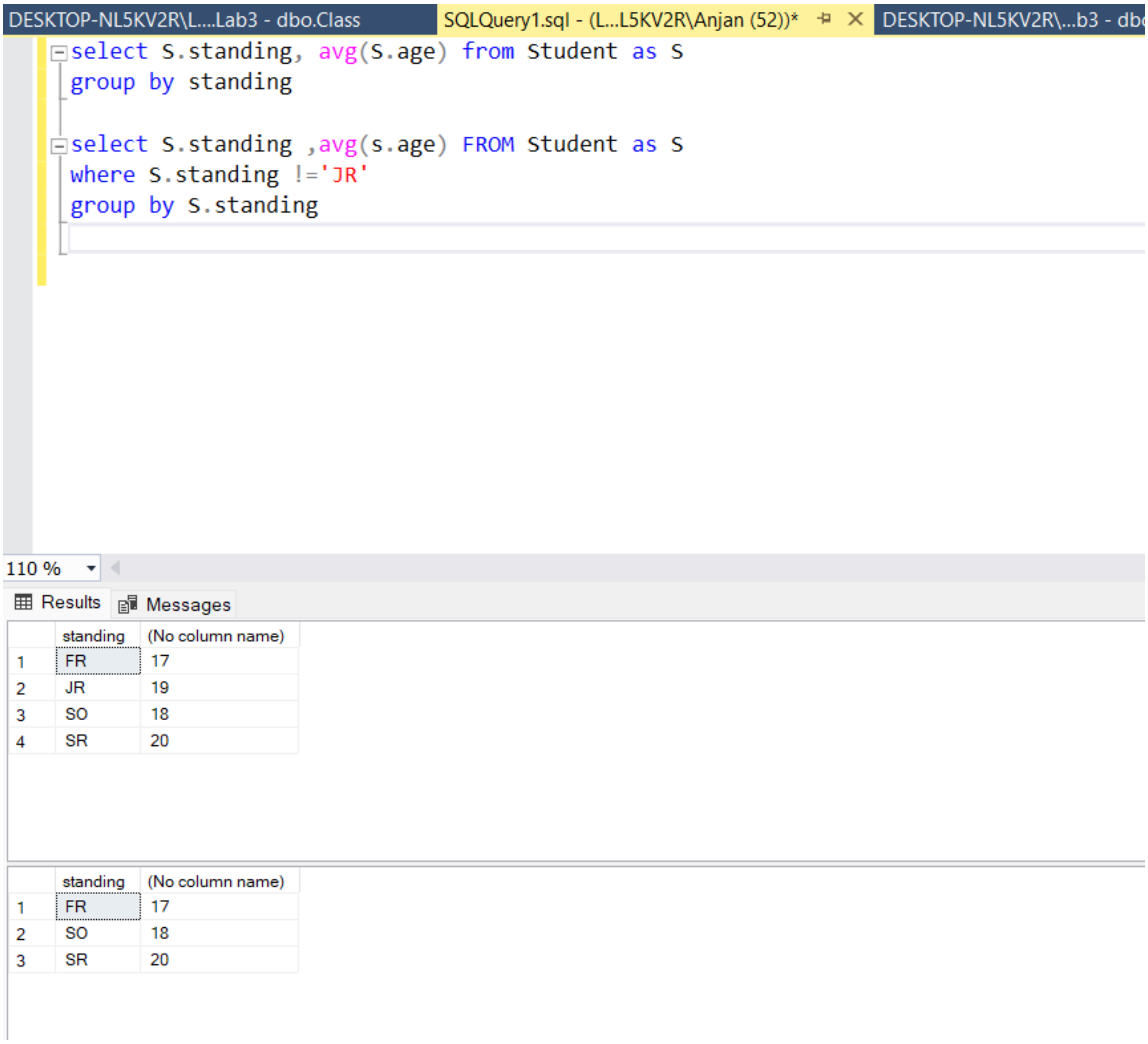
TASK #7

Problem statement: For each level, print the level and the average age of students for that level .For all levels except JR print the level and the average age of students for that level

Solution:

```
select S.standing, avg(S.age) from Student as S /*selecting standing and average age
from the student table using the avg function*/
group by standing

select S.standing ,avg(s.age) FROM Student as S
where S.standing !='JR'/*except JR level*/
group by S.standing
```



DESKTOP-NL5KV2R\L....Lab3 - dbo.Class
SQLQuery1.sql - (L...L5KV2R\Anjan (52))*
DESKTOP-NL5KV2R\...b3 - dbo
110 %
Results
Messages

```
select S.standing, avg(S.age) from Student as S
group by standing

select S.standing ,avg(s.age) FROM Student as S
where S.standing !='JR'
group by S.standing
```

	standing	(No column name)
1	FR	17
2	JR	19
3	SO	18
4	SR	20

	standing	(No column name)
1	FR	17
2	SO	18
3	SR	20

Reasoning:

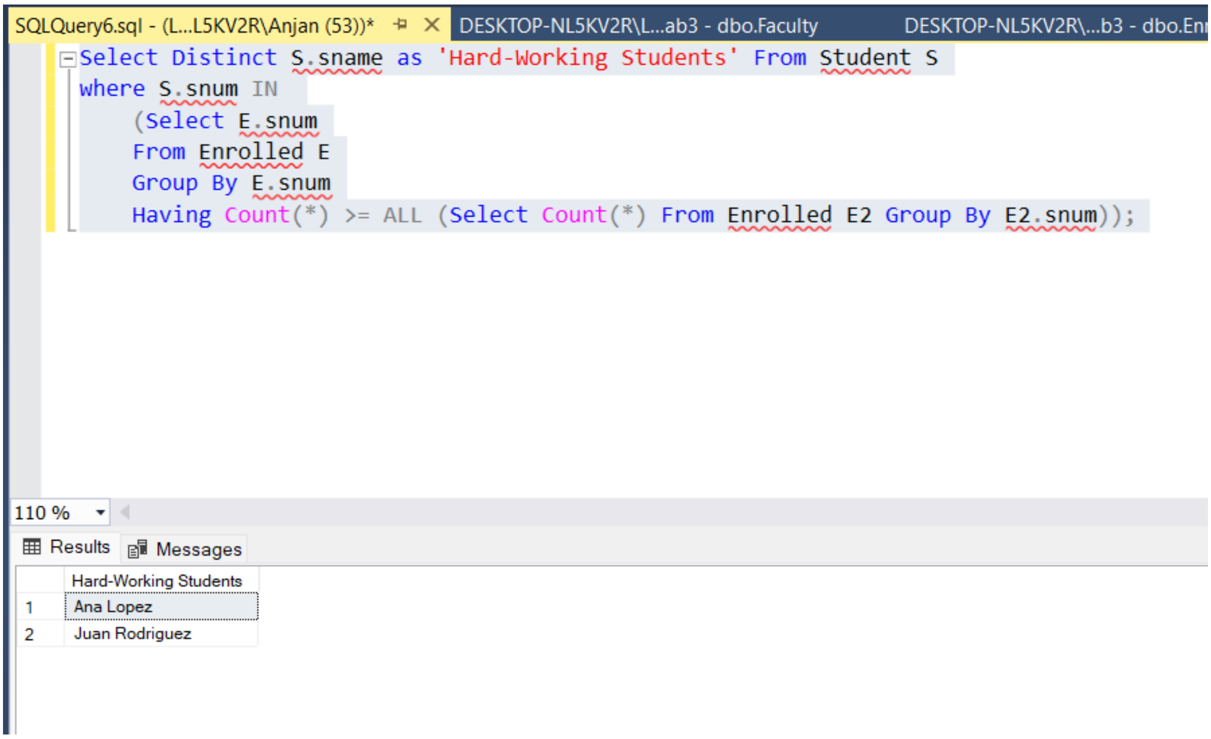
This is a comparatively simple task where we just have to select the students from the student table according to their standing and average age. And we get the expected result.

TASK #8

Problem statement: Find the names of students enrolled in the maximum number of classes. Rename the resulting column to “Hard-working students”

Solution:

```
Select Distinct S.sname as 'Hard-Working Students' From Student S /*renaming the
stundent column as hard working student*/
where S.snum IN  /* using a nested loop*/
    (Select E.snum /*here we will create connection with the enrolled table*/
    From Enrolled E
       Group By E.snum
       Having Count(*) >= ALL (Select Count(*) From Enrolled E2 Group By E2.snum))
/*here we count among all the studnets who have most number of enrollment in classes*/
```


SQLQuery6.sql - (L...L5KV2R\Anjan (53))*
DESKTOP-NL5KV2R\L...ab3 - dbo.Faculty
DESKTOP-NL5KV2R\...b3 - dbo.Enr
110 %
Results
Messages

```
Select Distinct S.sname as 'Hard-Working Students' From Student S
where S.snum IN
    (Select E.snum
    From Enrolled E
    Group By E.snum
    Having Count(*) >= ALL (Select Count(*) From Enrolled E2 Group By E2.snum));
```

	Hard-Working Students
1	Ana Lopez
2	Juan Rodriguez

Reasoning:

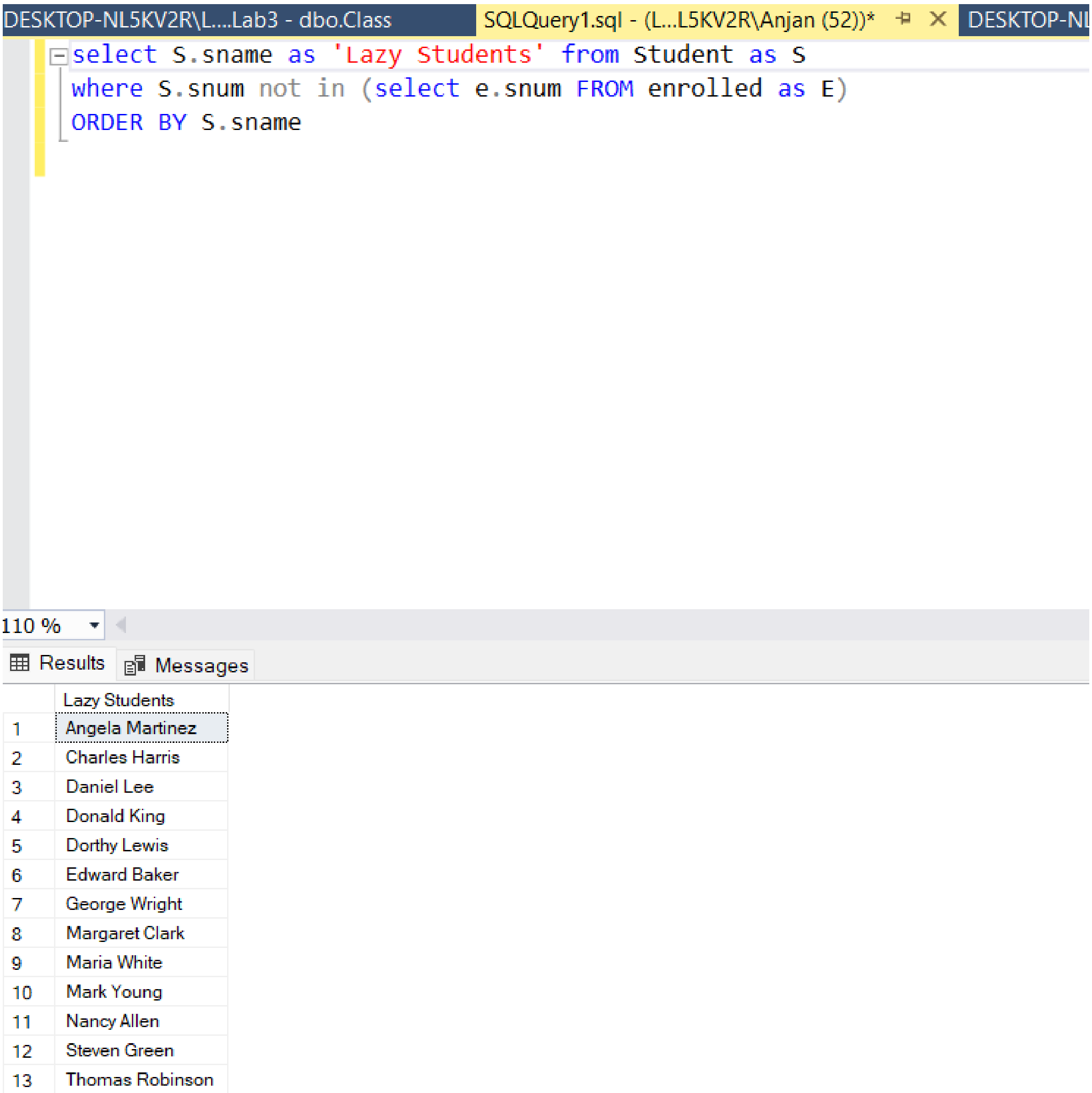
Here we need to find the students who are enrolled in most number of courses and they will be considered as the hard working student. To find this we need to again use nested loop, where we count all the students enrolled number of courses and compare with each students and find the students who have most number of classes. And we got the expected result.

TASK #9

Problem statement: Find the names of students not enrolled in any class. Rename the resulting column to “Lazy students”

Solution:

```
select S.sname as 'Lazy Students' from Student as S /*selecting students and renaming
as lazy students*/
where S.snum not in (select e.snum FROM enrolled as E) /*just checking which studnets
are not enrolled in any class*/
order by S.sname
```


DESKTOP-NL5KV2R\L....Lab3 - dbo.Class
SQLQuery1.sql - (L...L5KV2R\Anjan (52))*
DESKTOP-NL
110 %
Results
Messages

```
select S.sname as 'Lazy Students' from Student as S
where S.snum not in (select e.snum FROM enrolled as E)
ORDER BY S.sname
```

	Lazy Students
1	Angela Martinez
2	Charles Harris
3	Daniel Lee
4	Donald King
5	Dorthy Lewis
6	Edward Baker
7	George Wright
8	Margaret Clark
9	Maria White
10	Mark Young
11	Nancy Allen
12	Steven Green
13	Thomas Robinson

Reasoning:

This is also a simple task where we use have to use two tables, and find the students who are in the student table but not in the enrolled table. And we got the expected result.

INSTRUCTIONS

1. **Problem statement is mandatory.**
2. **A solution without explanation is NOT accepted.**
3. **If you need to copy the source code, you can do it with copy/paste commands. Please do not use screenshots for code listings.**
4. **Other screenshots (figures, graphs, etc.) should be scaled appropriately. Please cut off unnecessary elements on the images.**