**MODERN DATABASE SYSTEM LAB 8**

**UNIVERSITY COURSE ENROLLMENT DATA ANAYTICS**

**NAME: PAVITHIRAN.V ROLL.NO: 235229122**

**Question 1**:

SELECT p.PNAME

FROM prof\_data p

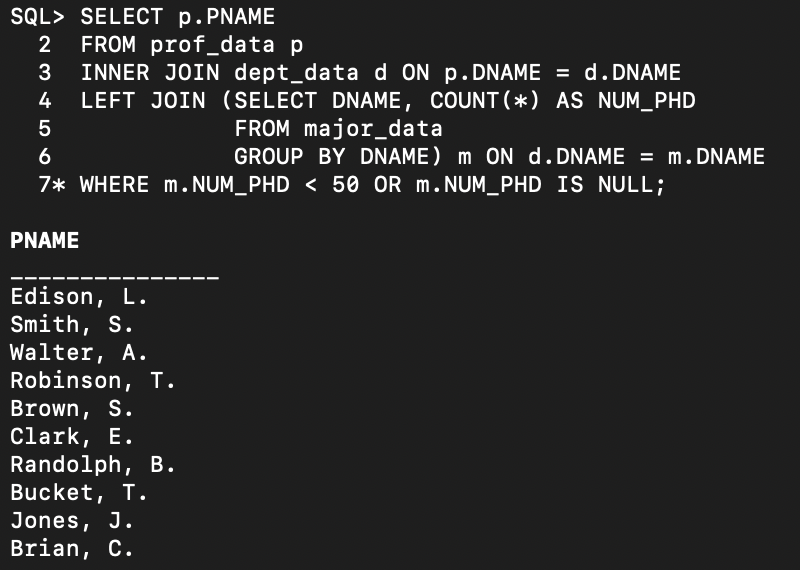
INNER JOIN dept\_data d ON p.DNAME = d.DNAME

LEFT JOIN (SELECT DNAME, COUNT(\*) AS NUM\_PHD

FROM major\_data

GROUP BY DNAME) m ON d.DNAME = m.DNAME

WHERE m.NUM\_PHD < 50 OR m.NUM\_PHD IS NULL;

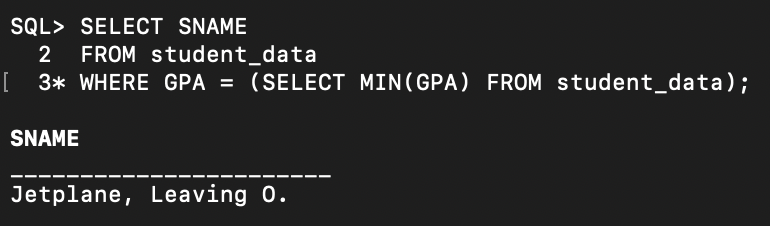


**Question 2**:

SELECT SNAME

FROM student\_data

WHERE GPA = (SELECT MIN(GPA) FROM student\_data);



**Question 3**:

SELECT

s.CNO AS ClassNumber,

s.SEC\_NO AS SectionNumber,

AVG(GPA) AS AverageGPA

FROM

section\_data s

INNER JOIN

course\_data c ON s.CNO = c.CNO

LEFT JOIN

enroll\_data e ON s.CNO = e.CNO AND s.SEC\_NO = e.SEC\_NO

LEFT JOIN

student\_data sd ON e.SID = sd.SID

WHERE

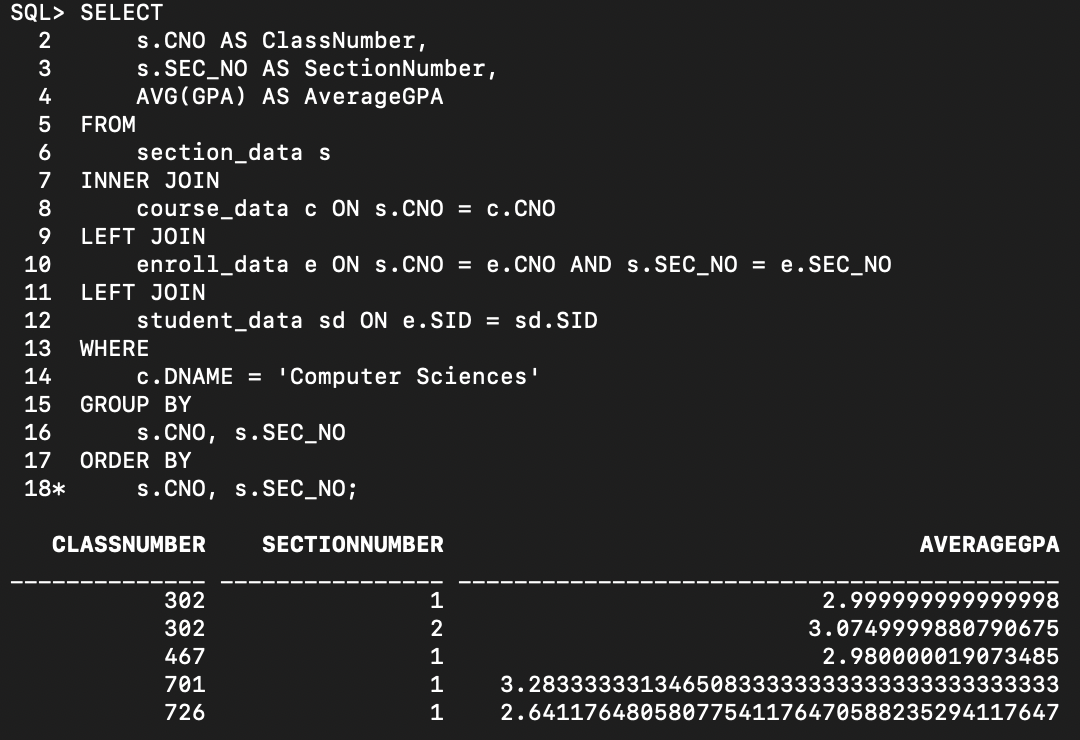
c.DNAME = 'Computer Sciences'

GROUP BY

s.CNO, s.SEC\_NO

ORDER BY

s.CNO, s.SEC\_NO;



**Question 4**:

SELECT

s.PNAME AS ProfessorName,

s.SEC\_NO AS SectionNumber

FROM

section\_data s

LEFT JOIN

enroll\_data e ON s.CNO = e.CNO AND s.SEC\_NO = e.SEC\_NO

GROUP BY

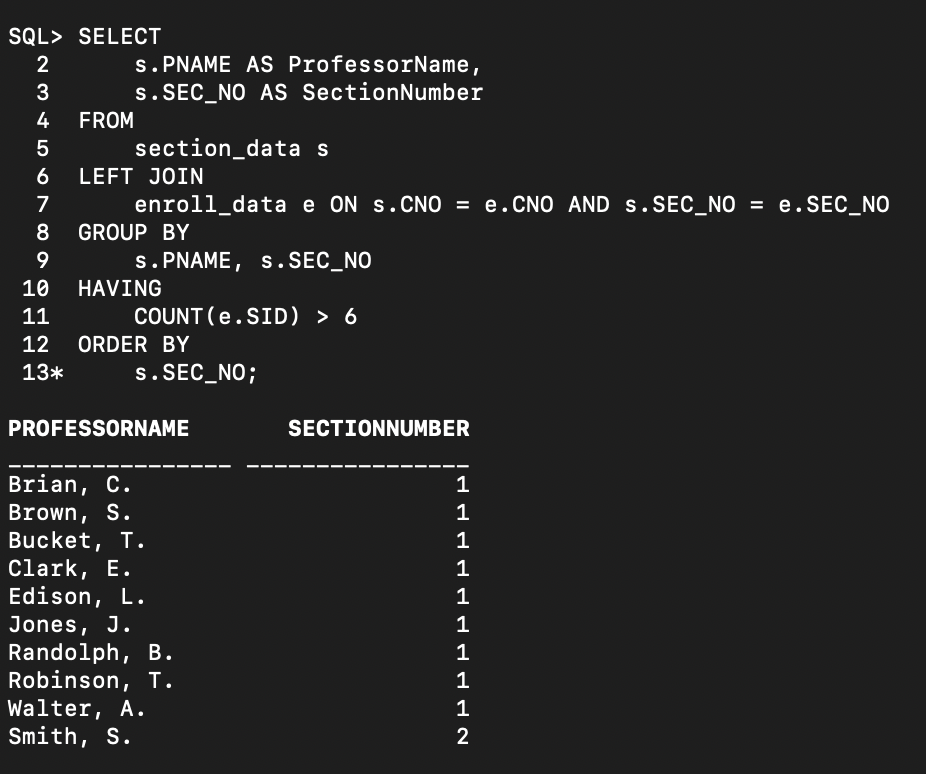
s.PNAME, s.SEC\_NO

HAVING

COUNT(e.SID) > 6

ORDER BY

s.SEC\_NO;



**Question 5**:

SELECT s.SID, st.SNAME

FROM (

SELECT SID, RANK() OVER (ORDER BY SectionCount DESC) AS rank

FROM (

SELECT e.SID, COUNT(\*) AS SectionCount

FROM enroll\_data e

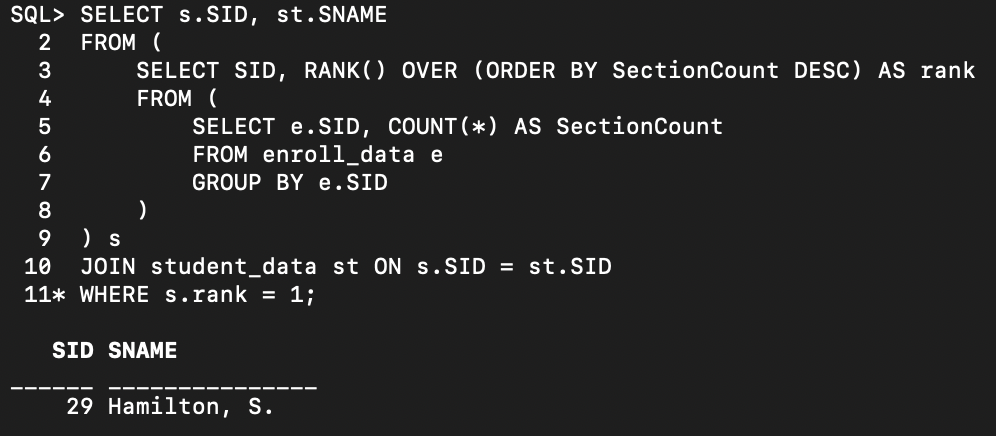
GROUP BY e.SID

)

) s

JOIN student\_data st ON s.SID = st.SID

WHERE s.rank = 1;



**Question 6**:

SELECT DISTINCT d.DNAME

FROM dept\_data d

WHERE EXISTS (

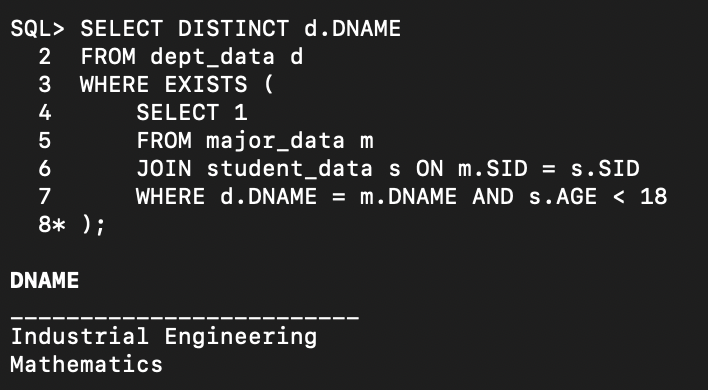
SELECT 1

FROM major\_data m

JOIN student\_data s ON m.SID = s.SID

WHERE d.DNAME = m.DNAME AND s.AGE < 18

);



**Question 7**:

SELECT DISTINCT s.SNAME, m.DNAME AS Major

FROM student\_data s

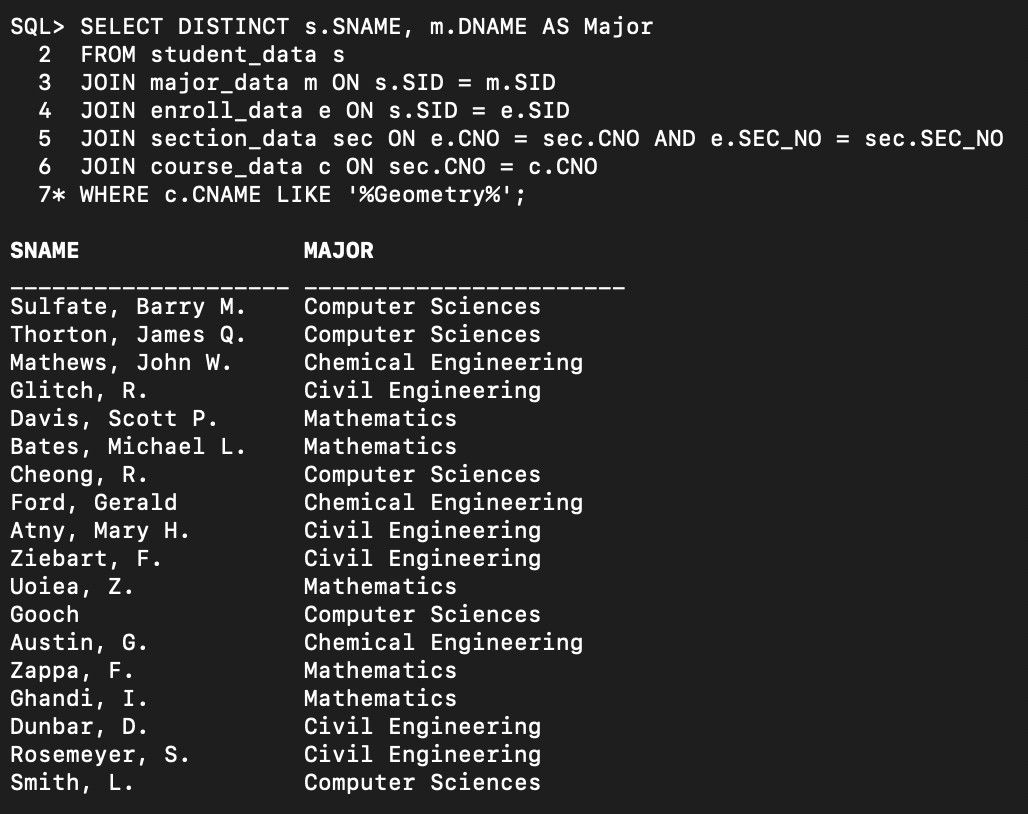
JOIN major\_data m ON s.SID = m.SID

JOIN enroll\_data e ON s.SID = e.SID

JOIN section\_data sec ON e.CNO = sec.CNO AND e.SEC\_NO = sec.SEC\_NO

JOIN course\_data c ON sec.CNO = c.CNO

WHERE c.CNAME LIKE '%Geometry%';



**Question 8**:

SELECT

d.DNAME AS DepartmentName,

NVL(d.NUM\_PHD, 0) AS NumPhDStudents

FROM

dept\_data d

LEFT JOIN

(

SELECT DISTINCT m.DNAME

FROM major\_data m

JOIN enroll\_data e ON m.SID = e.SID

JOIN section\_data sec ON e.CNO = sec.CNO AND e.SEC\_NO = sec.SEC\_NO

JOIN course\_data c ON sec.CNO = c.CNO

WHERE c.CNAME LIKE '%Geometry%'

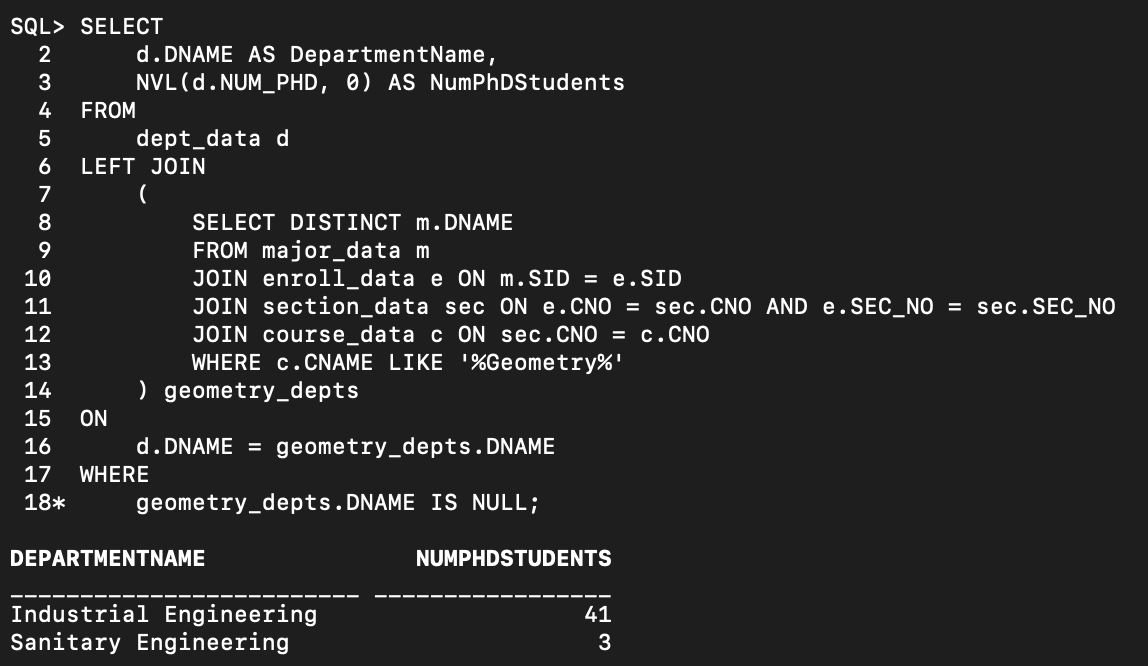
) geometry\_depts

ON

d.DNAME = geometry\_depts.DNAME

WHERE

geometry\_depts.DNAME IS NULL;



**Question 9**:

SELECT DISTINCT s.SNAME

FROM student\_data s

WHERE s.SID IN (

SELECT m.SID

FROM major\_data m

JOIN enroll\_data e ON m.SID = e.SID

JOIN section\_data sec ON e.CNO = sec.CNO AND e.SEC\_NO = sec.SEC\_NO

JOIN course\_data c ON sec.CNO = c.CNO

WHERE m.DNAME = 'Computer Sciences'

)

AND s.SID IN (

SELECT m.SID

FROM major\_data m

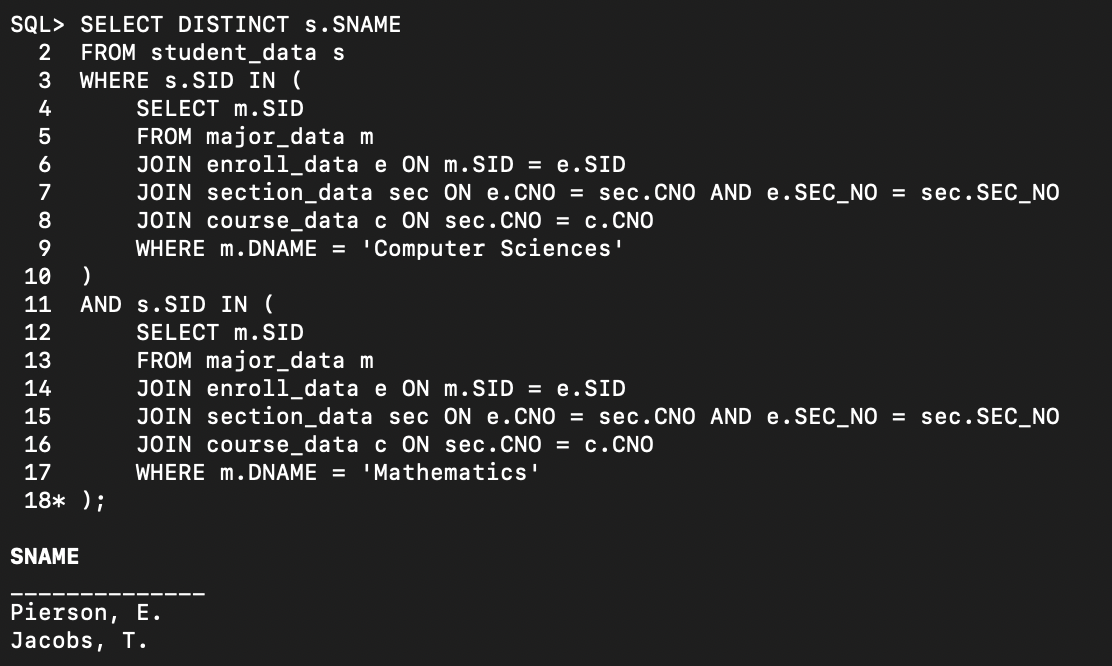
JOIN enroll\_data e ON m.SID = e.SID

JOIN section\_data sec ON e.CNO = sec.CNO AND e.SEC\_NO = sec.SEC\_NO

JOIN course\_data c ON sec.CNO = c.CNO

WHERE m.DNAME = 'Mathematics'

);



**Question 10**:

SELECT MAX(s.AGE) - MIN(s.AGE) AS AgeDifference

FROM student\_data s

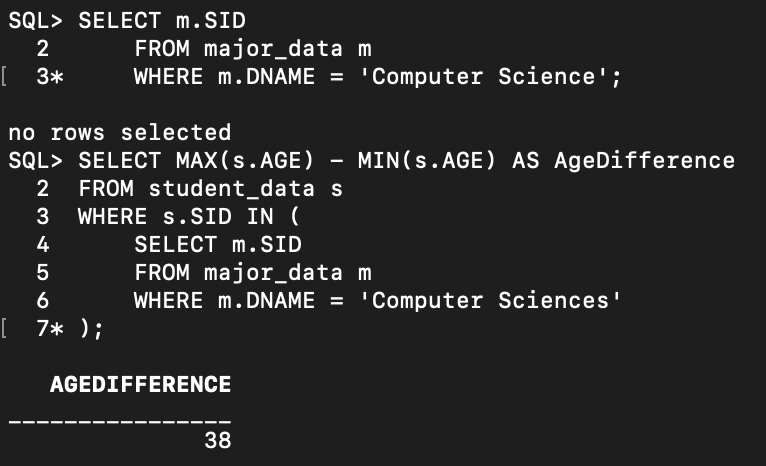
WHERE s.SID IN (

SELECT m.SID

FROM major\_data m

WHERE m.DNAME = 'Computer Sciences'

);



**Question 11**:

SELECT

d.DNAME AS DepartmentName,

AVG(s.GPA) AS AverageGPA

FROM

dept\_data d

JOIN

major\_data m ON d.DNAME = m.DNAME

JOIN

student\_data s ON m.SID = s.SID

WHERE

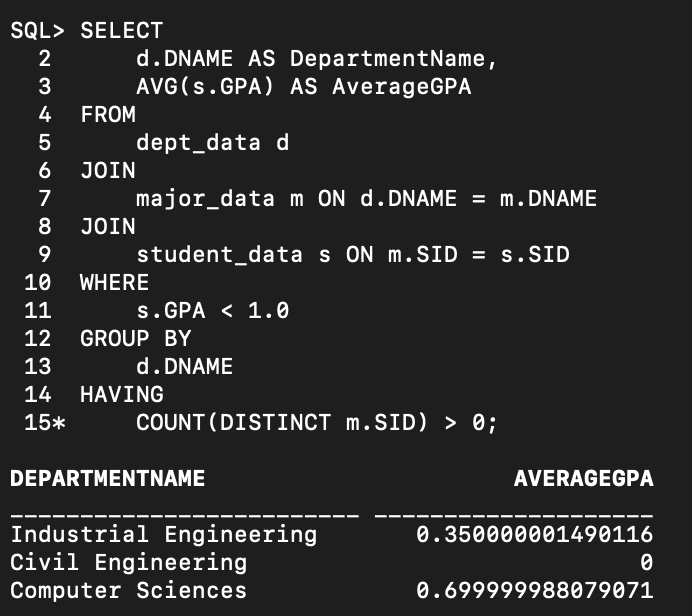
s.GPA < 1.0

GROUP BY

d.DNAME

HAVING

COUNT(DISTINCT m.SID) > 0;



**Question 12**:

SELECT s.SID, s.SNAME, s.GPA

FROM student\_data s

WHERE s.SID IN (

SELECT e.SID

FROM enroll\_data e

WHERE e.CNO IN (

SELECT CNO

FROM course\_data

WHERE DNAME = 'Civil Engineering'

)

GROUP BY e.SID

HAVING COUNT(DISTINCT e.CNO) = (

SELECT COUNT(DISTINCT CNO)

FROM course\_data

WHERE DNAME = 'Civil Engineering'

)

);

