CS3700 Introduction to Database Systems
Assignment 2: SQL on Academic DB
CS18B050 Aniswar Srivatsa Krishnan

1)
Description:
Obtain the course names and the department names of courses and their prerequisites such that the department offering the prerequisite course is different from the department offering the required course.

Query:
```
select d1.name as prereq_dept, c1.cname as prereq_course, d2.name as dept, c2.cname as course
        from prerequisite p, course c1, course c2, department d1, department d2
        where p.preReqCourse = c1.courseId and p.courseId = c2.courseId and c1.deptNo <> c2.deptNo and c1.deptNo = d1.deptId and c2.deptNo = d2.deptId;
```

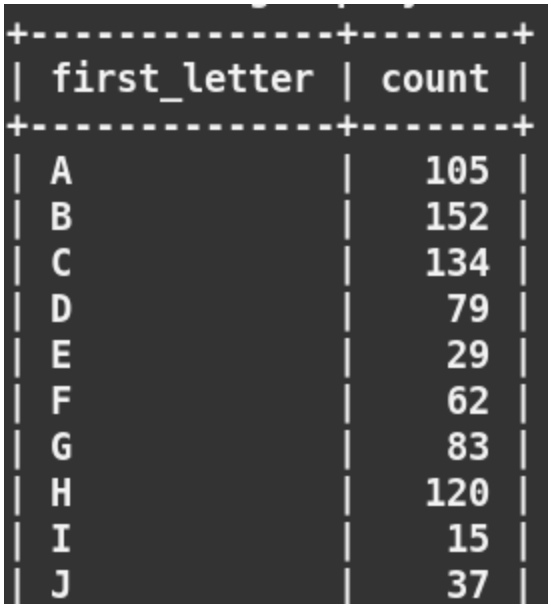
Result:

prereq_dept	prereq_course	dept	course
Athletics	World History	Biology	Antidisestablishmentarianism in Modern America
Accounting	Bankruptcy	Cybernetics	Elastic Structures
Statistics	Care and Feeding of Cats	Elec. Eng.	Marine Mammals
Psychology	Geology	Cybernetics	International Trade
Languages	Elastic Structures	Athletics	International Finance
History	Music of the 50s	Mech. Eng.	Design and Analysis of Algorithms
Accounting	Existentialism	Physics	The Music of the Ramones
Finance	Music 2 New for your Instructor	Marketing	Rock and Roll
Finance	Cognitive Psychology	Marketing	Rock and Roll
Biology	Composition and Literature	Cybernetics	Security

2)
Description:
Obtain the number of students whose names start with a particular letter along with the corresponding letter, for each letter such that there exists atleast one student whose name starts with that particular letter. Sort your result in the increasing order of first letters.

Query:
```
select substr(name, 1, 1) as first_letter, count(rollNo) as count
        from student
        group by substr(name, 1, 1) order by substr(name, 1, 1);
```

Result:



first_letter	count
A	105
B	152
C	134
D	79
E	29
F	62
G	83
H	120
I	15
J	37

3)
Description:
Obtain the department name for the department with maximum number of students enrolled in each offering (a course offered for a particular year is referred to as an offering) of the course 'Image Processing' along with the corresponding number of students enrolled and year

Query:

```
select d1 as department_name, y2 as year, number_of_students
        from(
        select e1.courseId as co1,  e1.year as y2, dep.name as d1, count(e1.rollNo) as
number_of_students
        from enrollment e1, course c1, student s1, department dep
        where c1.courseId = e1.courseId and s1.rollNo = e1.rollNo and c1.cname = 'Image
Processing' and s1.deptNo = dep.deptId group by co1, y2, d1) t2
        join
        (select courseId, year, max(number_of_students) as ns
                from (
                select e.courseId, e.year, s.deptNo, count(e.rollNo) number_of_students
                from enrollment e, course c, student s
                where c.courseId = e.courseId and s.rollNo = e.rollNo and c.cname = 'Image
Processing'
                group by courseId, year, s.deptNo) as T
                group by courseId, year) t1
        on co1 = t1.courseId and y2 = t1.year and number_of_students = t1.ns;
```

Result:

department_name	year	number_of_students
History	2002	26

4)
Description:
Obtain the names and roll numbers of the students from the CSE 2002 batch who have scored the first, second and third highest number of S grades, along with the number of S grades they have scored.

Query:

```
select s.rollNo, s.name, number_of_s
        from (
        select rollNo as r1, count(*) as number_of_s
        from enrollment
        where grade = 'S'
        group by rollNo) as T, student s, department d
        where T.r1 = s.rollNo and s.deptNo = d.deptId and d.name = 'Comp. Sci.' and s.year =
'2002' order by number_of_s desc limit 3;
```

Result:

rollNo	name	number_of_s
97590	Rammer	2
75082	Havill	2
65703	Goldman	2

5)
Description:
Obtain the number of A grades given by professor mingoz in each course he has taught (include only the courses with atleast 60 A grades), along with the course name, sem and year.

Query:

```
select c.cname, T.sem, T.year, T.number_of_a
        from (
        select t.empId, e.courseId, e.sem, e.year, count(*) as number_of_a
        from enrollment e, teaching t, professor p
```

where grade = 'A' and t.empId = p.empId and p.name='Mingoz' and t.courseId = e.courseId and t.sem = e.sem and t.year=e.year
group by t.empId, e.courseId, e.sem, e.year having number_of_a > 59) as T, course c
where c.courseId = T.courseId;

Result:

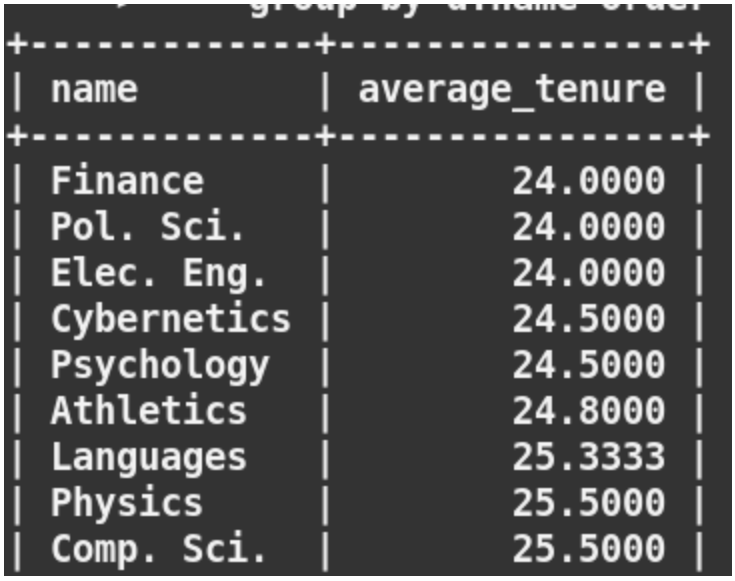
cname	sem	year	number_of_a
Manufacturing	even	2002	67
World History	even	2003	67
Embedded Systems	odd	2005	82
Biostatistics	even	2001	72

6)
Description:
Obtain the average tenure of professors for each department, along with the department name sorted in the increasing order of average tenure. Consider tenure as the number of years since the professor joined.

Query:

```
select d.name, avg(2021-p.startyear) as average_tenure
    from professor p, department d
    where p.deptNo = d.deptId
    group by d.name order by average_tenure asc;
```

Result:



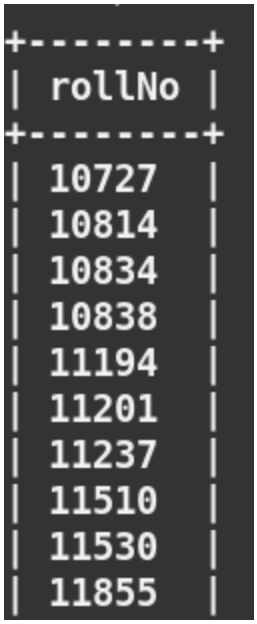
name	average_tenure
Finance	24.0000
Pol. Sci.	24.0000
Elec. Eng.	24.0000
Cybernetics	24.5000
Psychology	24.5000
Athletics	24.8000
Languages	25.3333
Physics	25.5000
Comp. Sci.	25.5000

7)
Description:
Obtain the roll numbers of students who have got a B grade in a 4 credit course and a C grade in a 3 credit course

Query:

```
select s.rollNo
        from student s
        where exists(
        select *
        from enrollment e, course c
        where e.rollNo = s.rollNo and c.courseId = e.courseId and c.credits = 4 and e.grade='B')
and exists (
                select *
                from enrollment e, course c
                where e.rollNo = s.rollNo and c.courseId = e.courseId and c.credits = 3 and
e.grade='C');
```

Result:



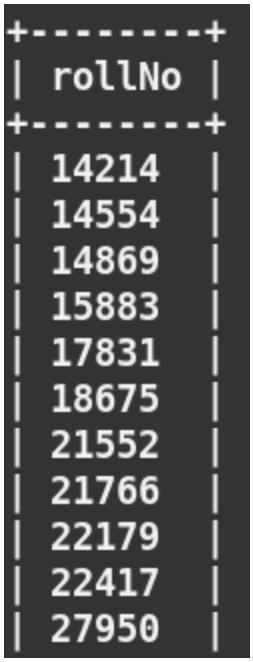
rollNo
10727
10814
10834
10838
11194
11201
11237
11510
11530
11855

8)
Description:
Obtain the roll numbers of students whose advisors are female and have taught more than 4 courses

Query:

```
select s.rollNo
        from student s, professor p
        where 4 < ALL (
        select count(distinct t.courseId)
        from teaching t
        where t.empId = s.advisor) and p.empId = s.advisor and p.sex='female';
```

Result:



rollNo
14214
14554
14869
15883
17831
18675
21552
21766
22179
22417
27950

Note: There are outputs that contain more than 100 rows. Only the top 10 rows are shown here for the sake of brevity.