

Query 1)

Get the course name and courseId for all the courses which have been allotted more than 3 credits, offered by department of Finance and is or has been taught by a male professor.

```
-- Query 1
select c.cname as name,c.courseId as id
from course as c, teaching as t, professor as p, department as d
where c.credits > 3 and
      t.courseId = c.courseId and
      p.sex = "male" and
      t.empId = p.empId and
      c.deptNo = d.deptId and
      d.name = "Finance";
-- done
```

course_name	course_id
World History	319
Embedded Systems	362

Query 2)

Get the name, empId, deptNo of the professors who have taught in the classroom R1 or R2 in some time of their tenure.

```
-- Query 2
⊖ (select p.name, p.empId, p.deptNo
   from professor as p, teaching as t
  where p.empId = t.empId and t.classRoom = "R1")
union
⊖ (select p.name, p.empId, p.deptNo
   from professor as p, teaching as t
  where p.empId = t.empId and t.classRoom = "R2")
-- done
```

name	empld	deptNo
Kean	28097	7
Bondi	34175	9
Mingoz	6569	11
Wieland	19368	6
DAgostino	22591	17
Queiroz	80759	2

Query 3)

Get the name of professors, their department name and department number who have failed students. Here failed means U grade was given.

```
-- Query 3
SET sql_mode=(SELECT REPLACE(@@sql_mode, 'ONLY_FULL_GROUP_BY', ''));
SET SQL_SAFE_UPDATES = 0;
drop table if exists coursesInWhichStudentsFailed;
create table coursesInWhichStudentsFailed as(
select e.courseId, e.sem, e.year
from enrollment as e
where e.grade = "U"
order by e.courseId
);
SET sql_mode=(SELECT REPLACE(@@sql_mode, 'ONLY_FULL_GROUP_BY', ''));
drop table if exists table1;
create table table1 as(
SELECT ROW_NUMBER() OVER() AS num_row,
courseId,
sem,
year
FROM coursesInWhichStudentsFailed);
DELETE t1 FROM table1 t1
INNER JOIN table1 t2
WHERE
t1.num_row < t2.num_row AND
t1.courseId = t2.courseId AND
t1.sem = t2.sem AND
t1.year=t2.year;
```

```

-- table1 contains the courseIds
SET sql_mode=(SELECT REPLACE(@@sql_mode, 'ONLY_FULL_GROUP_BY', ''));
drop table if exists table2;
create table table2 as(
select t.empId
from teaching as t, table1 as t1
where t.courseId = t1.courseId and t.sem = t1.sem and t.year = t1.year
order by t.empId
);

SET sql_mode=(SELECT REPLACE(@@sql_mode, 'ONLY_FULL_GROUP_BY', ''));
drop table if exists table2WithNums;
create table table2WithNums as(
SELECT ROW_NUMBER() OVER() AS num_row,
      empId
FROM table2);

DELETE t1 FROM table2WithNums t1
INNER JOIN table2WithNums t2
WHERE
      t1.num_row < t2.num_row AND
      t1.empId = t2.empId;
-- select * from table2WithNums;
select p.name as pName, p.deptNo, d.name as depName
from professor as p, department as d, table2WithNums as t20
where p.empId = t20.empId and d.deptId = p.deptNo
order by depName;
-- -- done

```

pName	deptNo	depName
Ullman	8	Accounting
Romero	14	Astronomy
Bawa	15	Athletics
Tung	15	Athletics
Queiroz	2	Biology
Valtchev	2	Biology
Bondi	9	Comp. Sci.
Pimenta	13	Cybernetics
Dale	13	Cybernetics
Gustafsson	19	Elec. Eng.
Vicentino	19	Elec. Eng.
Sullivan	19	Elec. Eng.
Kean	7	English
Sakurai	7	English
Mingoz	11	Finance
Mahmoud	12	Geology
Liley	10	Languages
Morris	5	Marketing
Voronina	4	Physics
Wieland	6	Pol. Sci.
D'Agostino	17	Psychology
Atanassov	16	Statistics
Choll	16	Statistics

Query 4)

Get the cname of all courses which have been taught by a female professor and has more than 1 prerequisites and is taught in an odd semester.

```
-- Query 4
SET sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROUP_BY',''));
drop table if exists coursesWithPrereqsAndCount;
create table coursesWithPrereqsAndCount as(
select p.courseId, count(p.preReqCourse) as numPrereqs
from prerequisite as p
group by p.courseId
having count(p.preReqCourse) > 1
);
-- select * from coursesWithPrereqsAndCount;
SET sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROUP_BY',''));
drop table if exists coursesWithEmpIdOddSem;
create table coursesWithEmpIdOddSem as(
select t.courseId, t.empId
from teaching as t, coursesWithPrereqsAndCount as t1
where t.courseId = t1.courseId and t.sem = "Odd"
group by t.courseId
);
-- select * from coursesWithEmpIdOddSem;
select p.name as pName, p.sex, c.cname as cName
from professor as p, course as c, coursesWithEmpIdOddSem as c1
where c.courseId = c1.courseId and p.empId = c1.empId and p.sex = "female";
-- -- done
```

pName	sex	cName
Dale	female	Aquatic Chemistry
Wieland	female	International Practicum

Query 5)

Get the grade distribution of the course “Plastics” (even sem and year 2004) and order the grades according to number of students that obtained that grade in descending order.

```
-- Query 5
select grade, count(rollNo) as numStudents
from course as c, enrollment as e
where e.year = 2004 and e.sem = "even" and c.cname = "Plastics" and e.courseId = c.courseId
group by grade
order by numStudents desc;
-- done
```

grade	numStudents
B	78
A	65
C	39
D	30
S	30
E	28
U	20

Query 6)

Get the number of professors working in each department and order the departments in decreasing number of profs.

```
-- Query 6
select d.name, count(empId) as numProfs
from department as d, professor as p
where d.deptId = p.deptNo
group by d.name
order by numProfs desc
-- done
```

name	numProfs
Statistics	6
Athletics	5
Cybernetics	4
Elec. Eng.	4
Marketing	4
English	4
Accounting	4
Comp. Sci.	4
Languages	3
Pol. Sci.	3
Psychology	2
Biology	2
Mech. Eng.	2
Physics	2
Finance	1
Geology	1
Astronomy	1

Query 7)

Get the average number of courses taken by any student, group by department in the even sem, year 2003.

```

-- Query 7
SET sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROUP_BY',''));
drop table if exists numCoursesTakenByEachStudent;
create table numCoursesTakenByEachStudent as (
select s.rollNo, d.deptId, d.name, count(courseId) as numCourses
from student as s,department as d,enrollment as e
where e.year = 2003 and e.sem = "even" and d.deptId = s.deptNo and e.rollNo = s.rollNo
group by s.rollNo
order by d.deptId );

select t.deptId, t.name, avg(t.numCourses) as avgCourses
from numCoursesTakenByEachStudent as t
group by t.deptId
order by t.deptId;
-- done

```

deptId	name	avgCourses
1	Civil Eng.	1.5190
10	Languages	1.5570
11	Finance	1.5094
12	Geology	1.4074
13	Cybernetics	1.4902
14	Astronomy	1.4507
15	Athletics	1.4000
16	Statistics	1.3953
17	Psychology	1.4923
18	Math	1.3043
19	Elec. Eng.	1.5672
2	Biology	1.5231
20	Mech. Eng.	1.5205
3	History	1.4337
4	Physics	1.4138
5	Marketing	1.5625
6	Pol. Sci.	1.3559
7	English	1.5424
8	Accounting	1.4923
9	Comp. Sci.	1.5303

Query 8)

Get the number of students from each department that have shown interest in any other department.

Here interest in some department means that the student has taken course/courses in the that department.

```
-- Query 8
SET sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROUP_BY',''));
drop table if exists courseWithDepWithRollNo;
create table courseWithDepWithRollNo as (
select e.rollNo, e.courseId, c.deptNo from course as c, enrollment as e
where e.courseId = c.courseId
order by e.rollNo);

SET sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROUP_BY',''));
drop table if exists AddStudentDeptNocourseWithDepWithRollNo;
create table AddStudentDeptNocourseWithDepWithRollNo as (
select s.deptNo as sdepId, t.rollNo, t.courseId, t.deptNo as courseDepNo from student as s, courseWithDepWithRollNo as t
where s.rollNo = t.rollNo
order by s.rollNo);

SET sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROUP_BY',''));
drop table if exists table1;
create table table1 as (
select t.sdepId, t.rollNo
from AddStudentDeptNocourseWithDepWithRollNo as t
where t.sdepId != t.courseDepNo
order by t.rollNo);

SET sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROUP_BY',''));
drop table if exists table2;
create table table2 as(
select distinct sdepId, rollNo from table1
order by sdepId);
select sdepId as depId, count(rollNo) as studentCount from table2
group by depId;
-- done
```

depld	studentCount
1	120
10	118
11	97
12	91
13	86
14	106
15	92
16	85
17	100
18	90
19	98
2	100
20	105
3	117
4	96
5	85
6	109
7	95
8	99
9	108