**Basic SQL**

1. Find the ID, names of all the students from departments whose name

contain character '功'.

2. Find the ID, names and total credits of students in 邪门学院

department or in 兵器学院 department whose total credits are higher

than 50 credits

3. For the instructor 83821, show course\_id and title of all courses

taught by the instructor

select course.course\_id,title,credits

from course,teaches

where course.course\_id=teaches.course\_id and teaches.ID=83821

4. As above, but show the total number of credits for such courses

(taught by that instructor). You should use SQL aggregation on courses

taught by that instructor.

select sum(course.credits) as sumcredits

from course,teaches

where course.course\_id=teaches.course\_id and teaches.ID=83821

5. As above, but display the total credits for each of the instructors,

along with the ID of the instructor; don't bother about the name of the

instructors. (Don't bother about instructors who have not taught any

course, they can be omitted)

select instructor.ID,SUM(credits)as sum\_credits

from instructor,course,teaches

where instructor.ID=teaches.ID and teaches.course\_id=course.course\_id

group by instructor.ID

6. Find average instructors' salaries for each of courses, along with the

course\_id and title of the course, taught by instructors of 内功学院, the

result should be sorted from the lowest to the highest according to the

average salaries.

select SUM(salary)/COUNT(instructor.ID) as avgsal,course.course\_id,title

from instructor,teaches,course

where instructor.ID=teaches.ID and teaches.course\_id=course.course\_id and course.dept\_name='内功学院'

group by course.course\_id,course.title

order by avgsal asc

7. Find the names of all courses which have been taught in 南疆雨林

ever (there should be no duplicate names)

select distinct title

from course,section

where section.building='南疆雨林' and section.course\_id=course.course\_id

8. Display the IDs and names of all students who have never registered

for a course

select distinct student.ID,name

from student

except

(select distinct student.ID,name

from student,takes,course

where student.ID=takes.ID and takes.course\_id=course.course\_id)

9. Find the id and names of the courses which have been registered by

some students without evaluated grade.

select course.course\_id,title

from course,takes

where course.course\_id=takes.course\_id and takes.grade is null

10. Find the courses which are the Subsequence courses of other

courses. The result should involve the ids and titles of the Subsequence

courses and the ids and titles of its prerequisites (note: the names of

columns in result should show the roles of the courses clearly)

select course.course\_id,course.title,m.course\_id,m.title

from course ,prereq ,course as m

where course.course\_id=prereq.course\_id and prereq.prereq\_id=m.course\_id;