特别提醒：这就是实验报告，请下载后直接在本报告中进行编辑，只可以要求的添加实验截图或数据，不能删除原来的内容！！！完成后需要将本实验报告提交回课程平台！！！

《数据库原理与应用A》实验3

专业班级：\_计算机(职)201803 姓名：\_\_宋长河\_ 学号:\_201803607\_

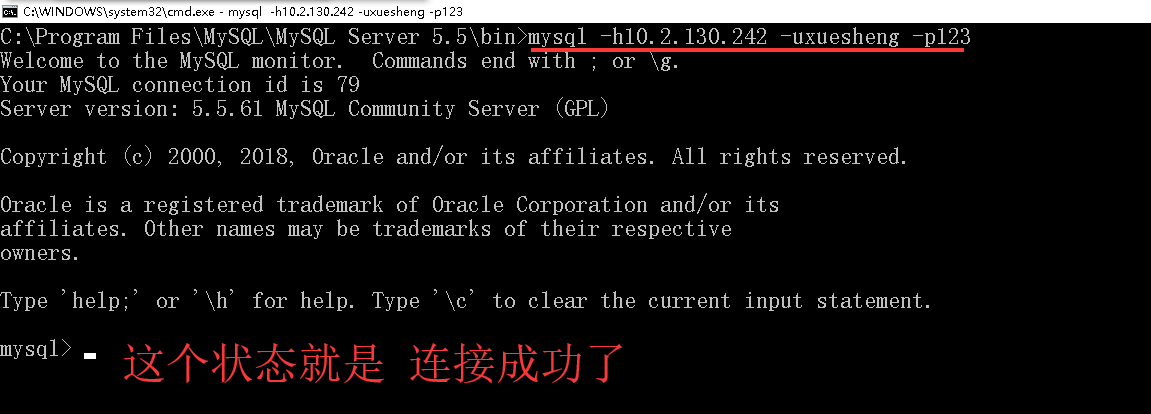
1. 【实验目的】
2. 掌握基本的SQL数据查询语句的用法
3. 掌握一定的SQL表达式的用法。
4. 掌握条件查询的用法。
5. 掌握DISTINCT等短语的用法。
6. 掌握LIMIT等短语的用法。
7. 掌握查询排序用法
8. 掌握使用SQL实现统计查询的方法
9. 掌握SQL分组统计查询的用法。
10. 【实验内容】
11. **实验准备**

查询练习使用潘sir在实验室的远程数据库。

首先进入windows命令控制台，输入以下命令进入MySQL的安装目标文件夹的bin文件夹下（默认是C:\Program Files\MySQL\MySQL Server 5.5\bin，如果是自己安装的请自行查找安装的位置）



然后在命令行中输入如下命令回车，成功连接到潘sir的服务器了:



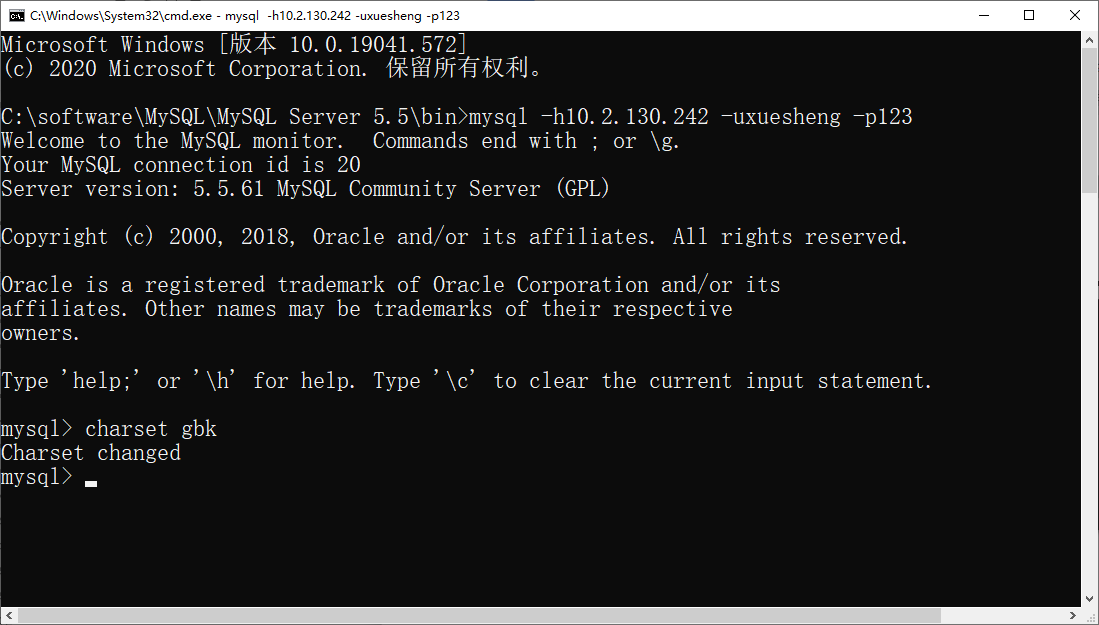
潘sir的服务器上同样用了中文，请先设置本次连接要用中文字符集。（命令前面做过哦！ 免费赠送： charset …）。

接下来就可以使用命令打开实验用的xjgl数据库 ：



准备完毕。

xjgl数据库中的teacher、student、course和score四张表，分别存储着老师、学生、课程和成绩的信息。



1. **查询命令练习**

练习以下SQL查询命令，巩固课堂教学内容。将你的命令及运行结果截图粘贴在相应的小题后面。

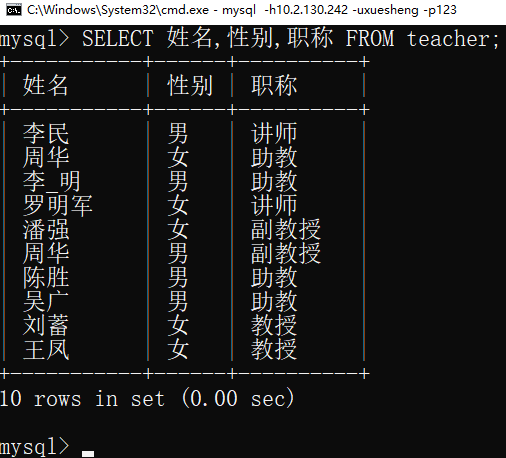
1. **SELECT子句练习**

1）-- 查询教师的姓名，性别和职称

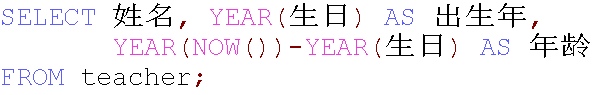


**将你命令及其执行结果截图放在下面：**

SELECT 姓名,性别,职称 FROM teacher;

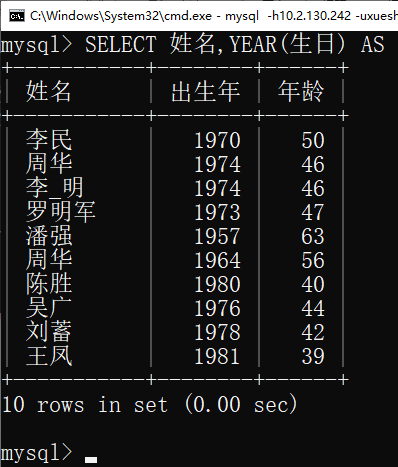


2）-- 利用表达式计算来查询教师姓名，生日，出生年和年龄



**将你命令及其执行结果截图放在下面：**

SELECT 姓名,YEAR(生日) AS 出生年,YEAR(NOW())-YEAR(生日) AS 年龄 FROM teacher;



**请回答：YEAR（）的作用是什么，NOW（）的作用是什么？**

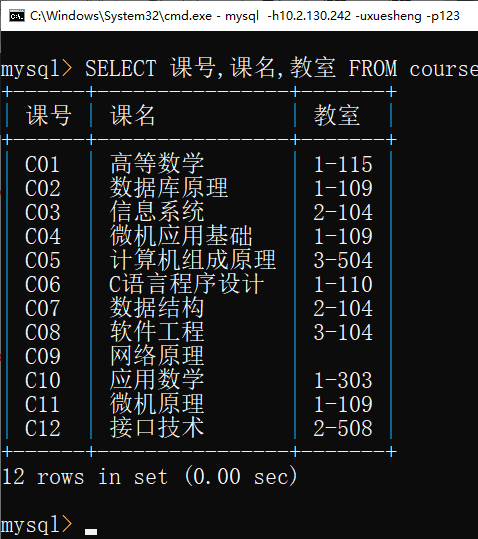
Year()函数为取出当前日期的年份；

Now()函数为系统当前日期和时间。

3）-- 查询课程的课号、课名和教室

**将你命令及其执行结果截图放在下面：**

SELECT 课号,课名,教室 FROM course;



4）-- 查询学生的学号、姓名、性别、年龄和出生年

（提示：想清楚年龄、现在时间和出生年之间的关系！）

**将你命令及其执行结果截图放在下面：**

SELECT 学号,姓名,性别,年龄,YEAR(NOW())-年龄 AS 出生年 FROM student;



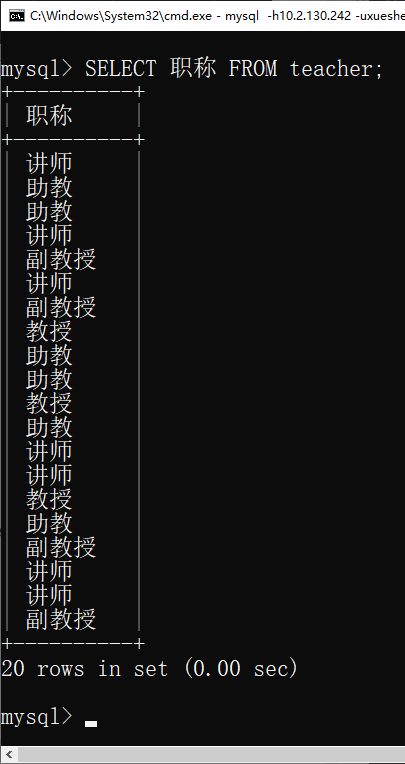
1. **DISTINCT短语练习**

1）--首先查询教师的职称



**将你命令及其执行结果截图放在下面：**

SELECT 职称 FROM teacher;



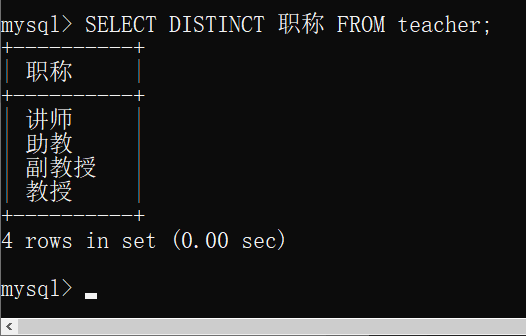
2）--然后加上DISTINCT消除结果中的重复行

--从而可以查询出只有哪几种职称



**将你命令及其执行结果截图放在下面：**

SELECT DISTINCT 职称 FROM teacher;

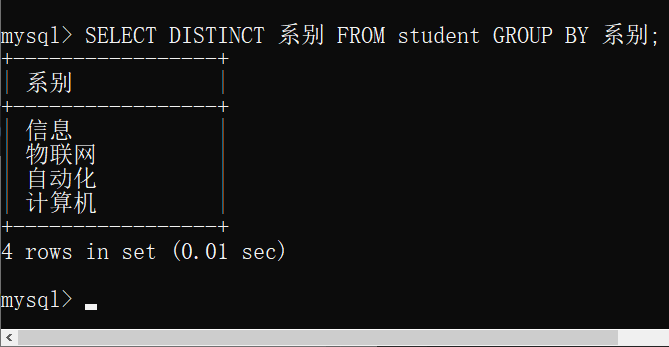


3）-- 查询Student表中有哪几个系

提示：有4个系

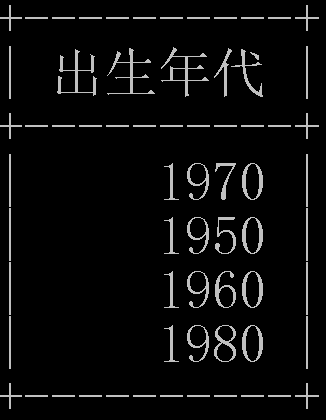
**将你命令及其执行结果截图放在下面：**

SELECT DISTINCT 系别 FROM student GROUP BY 系别;



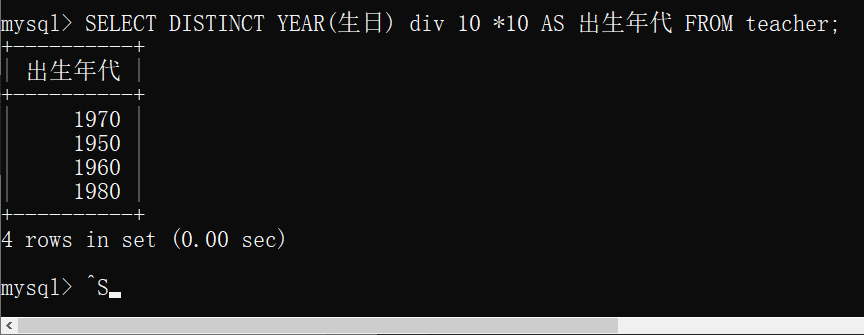
4）-- 如果以10年为一个年代，查询Teacher表中的老师出生于哪些年代。

提示：有4个年代。Mysql整除运算是div，如x div y表示x整除y的结果。

参考：

**将你命令及其执行结果截图放在下面：**

SELECT DISTINCT YEAR(生日) div 10 \*10 AS 出生年代 FROM teacher;



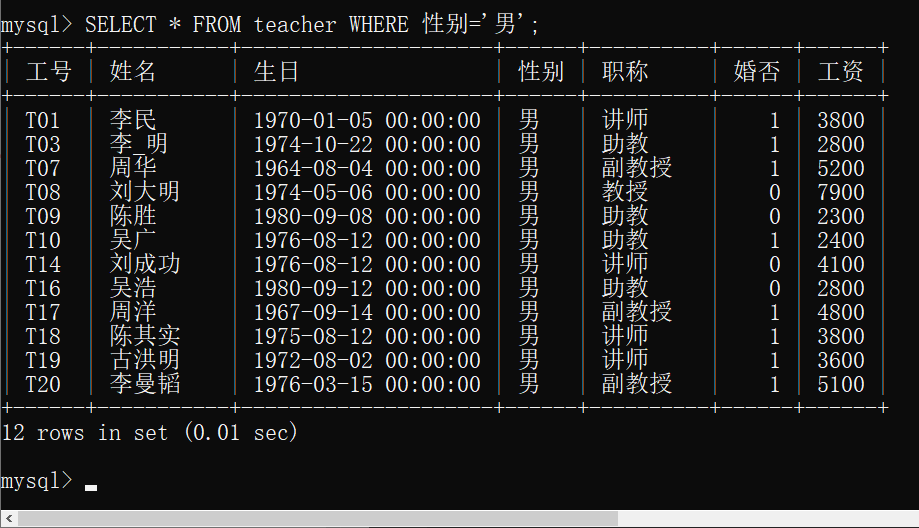
1. **条件查询（WHERE子句）练习**

1）-- 查询所有男教师



**将你命令及其执行结果截图放在下面：**

SELECT \* FROM teacher WHERE 性别='男';

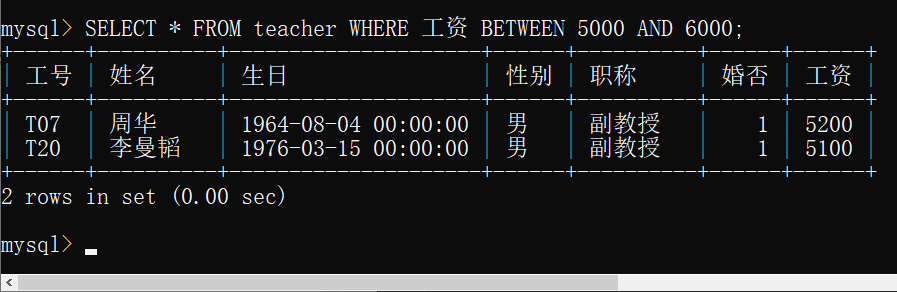


2）-- 查询工资在5000到6000之间的教师



**将你命令及其执行结果截图放在下面：**

SELECT \* FROM teacher WHERE 工资 BETWEEN 5000 AND 6000;

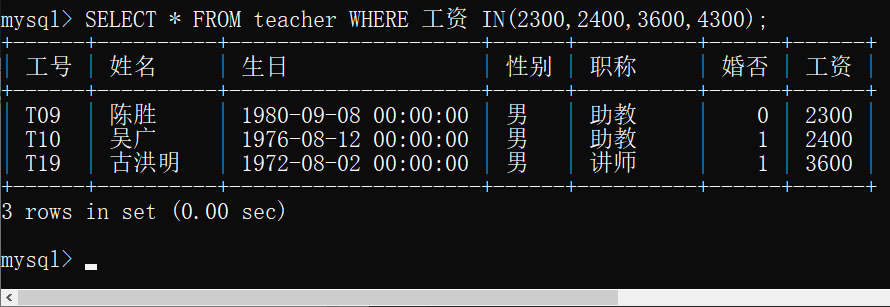


3）-- 查询工资为2300，2400,3600或4300之一的教师



**将你命令及其执行结果截图放在下面：**

SELECT \* FROM teacher WHERE 工资 IN(2300,2400,3600,4300);

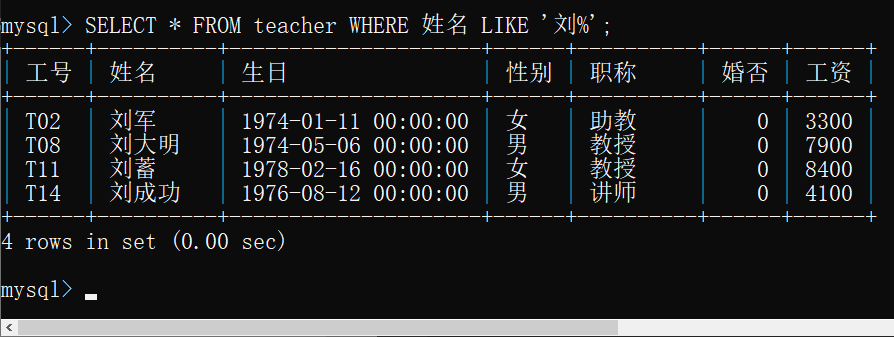


4）-- 查询姓刘的老师



**将你命令及其执行结果截图放在下面：**

SELECT \* FROM teacher WHERE 姓名 LIKE '刘%';

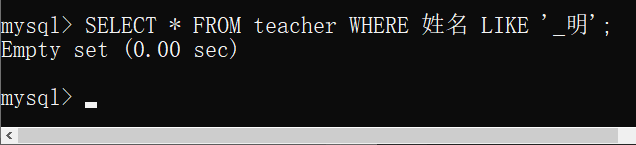


5）-- 查询名字只有2个字且最后一个字是“明”的教师

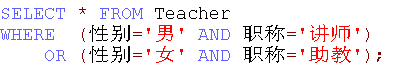


**将你命令及其执行结果截图放在下面：**

SELECT \* FROM teacher WHERE 姓名 LIKE '\_明';

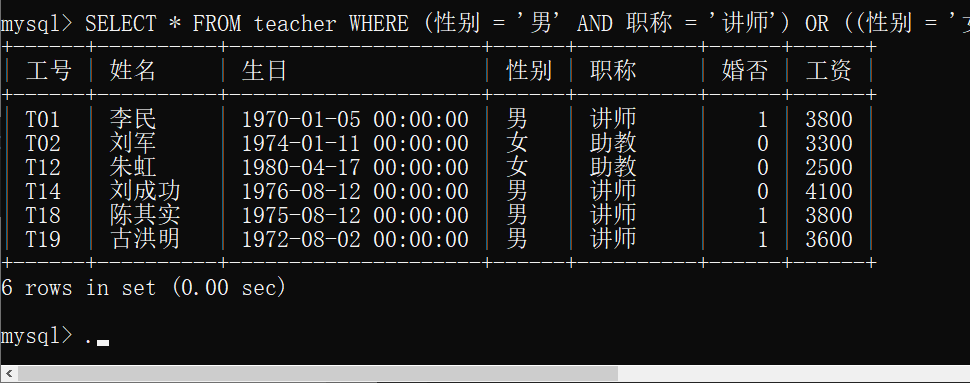


6）-- 查询所有的男讲师或者女助教



**将你命令及其执行结果截图放在下面：**

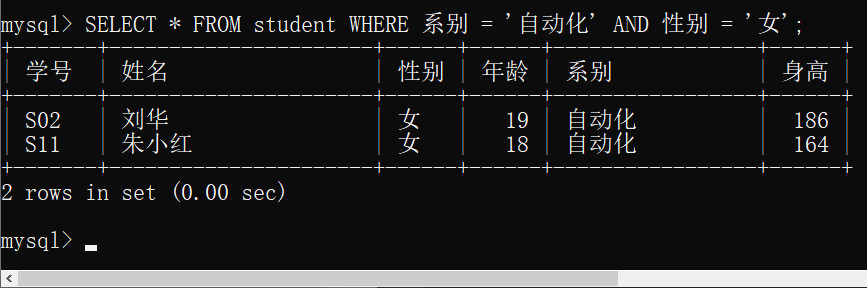
SELECT \* FROM teacher WHERE (性别 = '男' AND 职称 = '讲师') OR ((性别 = '女' AND 职称 = '助教'));



7）-- 查询自动化系的女同学

**将你命令及其执行结果截图放在下面：**

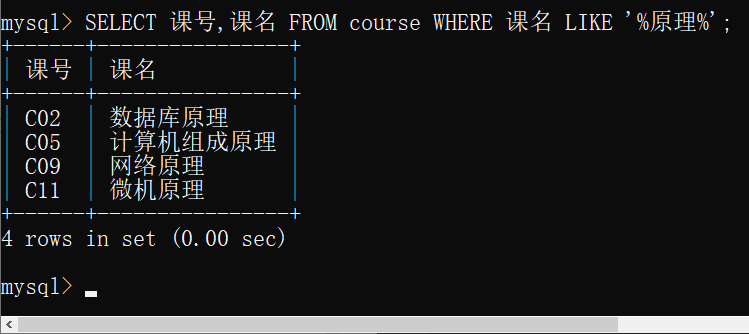
SELECT \* FROM student WHERE 系别 = '自动化' AND 性别 = '女';



8）-- 查询所有含“原理”两字的课程的课号和课名

**将你命令及其执行结果截图放在下面：**

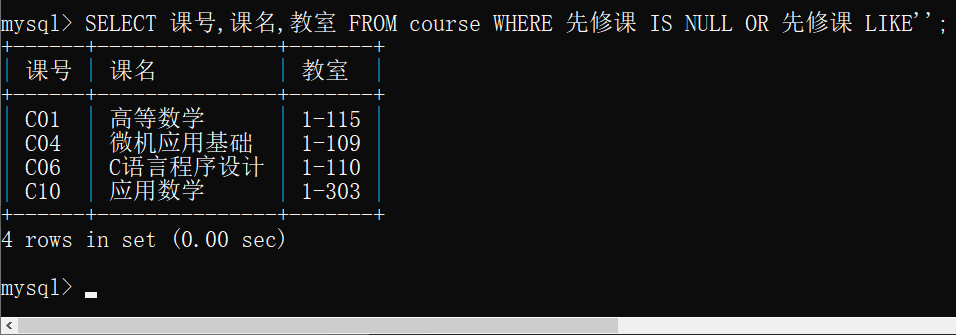
SELECT 课号,课名 FROM course WHERE 课名 LIKE '%原理%';



9）-- 查询没有先修课的课程的课号、课名和教室

**将你命令及其执行结果截图放在下面：**

SELECT 课号,课名,教室 FROM course WHERE 先修课 IS NULL OR 先修课 LIKE'';

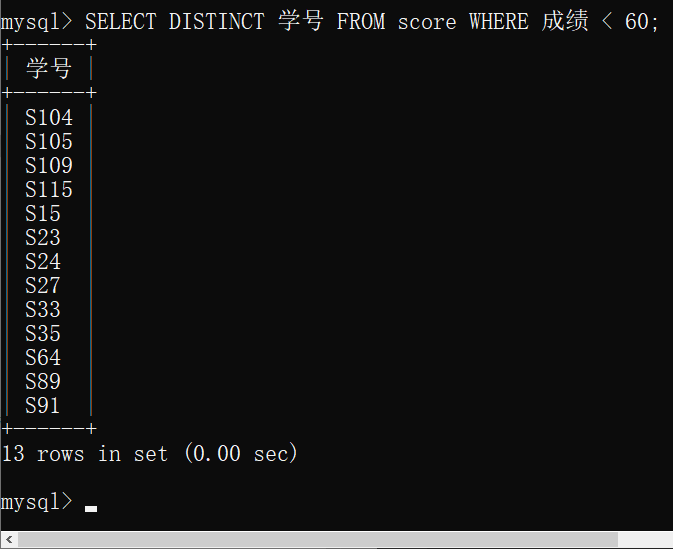


10）-- 查询有成绩不合格的学生的学号

提示：有13人

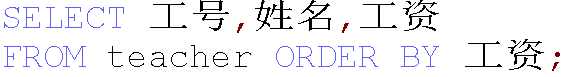
**将你命令及其执行结果截图放在下面：**

SELECT DISTINCT 学号 FROM score WHERE 成绩 < 60;



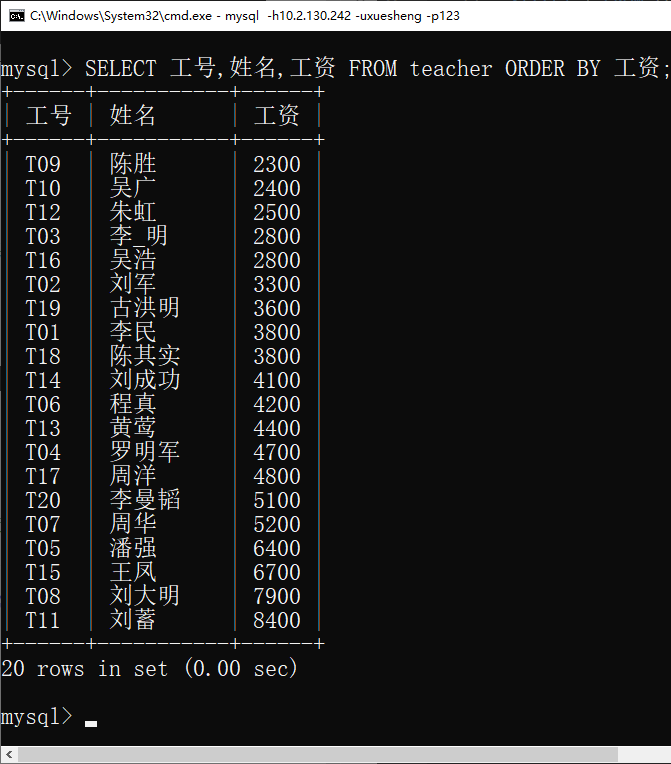
1. **ORDER BY子句练习**

1）-- 按工资的升序显示所有教师的工号、姓名和工资



**将你命令及其执行结果截图放在下面：**

SELECT 工号,姓名,工资 FROM teacher ORDER BY 工资;



2）-- 现在改为工资的降序显示所有教师的工号、姓名和工资



**将你命令及其执行结果截图放在下面：**

SELECT 工号,姓名,工资 FROM teacher ORDER BY 工资 DESC;

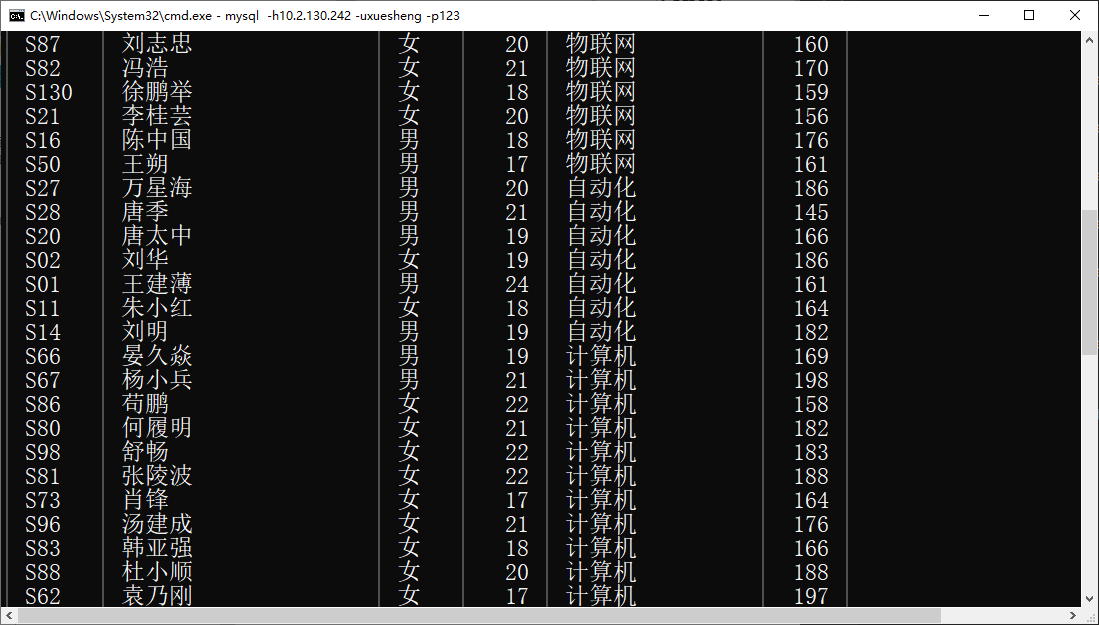


3）-- 查询学生，且要求系相同的学生显示在一起

提示：在一起，也是排序的体现哦

**将你命令及其执行结果截图放在下面：**

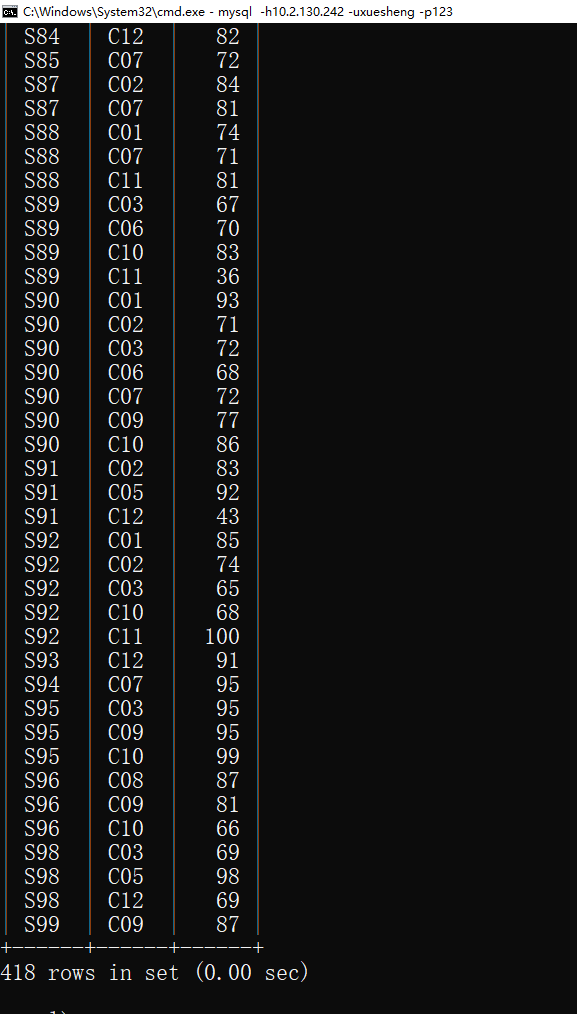
SELECT \* FROM student ORDER BY 系别;



4）-- 查询score表，要求按学号的从小大到排序，同一学生按其成绩的从大到小排序

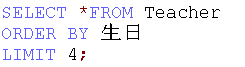
**将你命令及其执行结果截图放在下面：**

SELECT \* FROM score ORDER BY 学号;



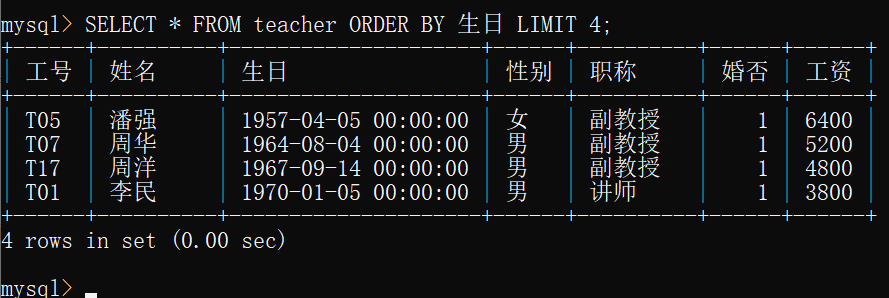
1. **Limit子句练习**

1）-- 查询年龄最大的4位教师



**将你命令及其执行结果截图放在下面：**

SELECT \* FROM teacher ORDER BY 生日 LIMIT 4;

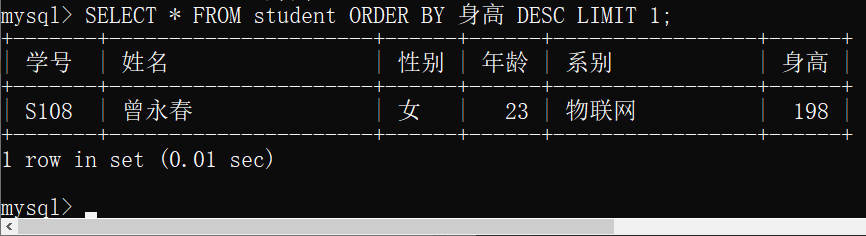


2）-- 查询长得最高的学生

（提示：排序+头条）

**将你命令及其执行结果截图放在下面：**

SELECT \* FROM student ORDER BY 身高 DESC LIMIT 1;

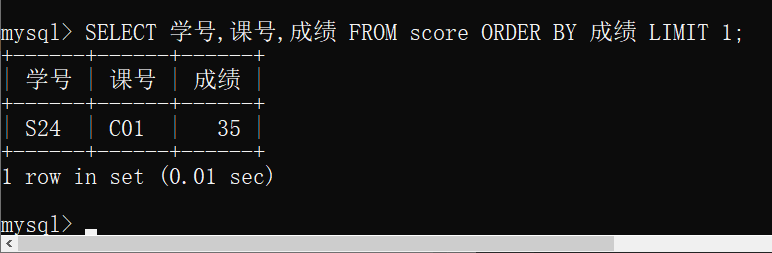


3）-- 查询成绩最差的学生的学号，课号和成绩

（提示：排序+头条）

**将你命令及其执行结果截图放在下面：**

SELECT 学号,课号,成绩 FROM score ORDER BY 成绩 LIMIT 1;



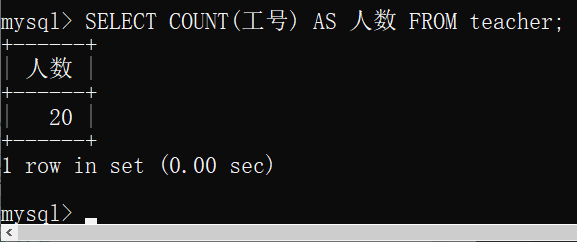
1. **统计查询练习**

1）-- 查询教师人数

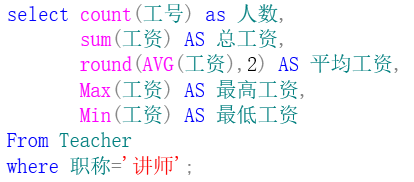


**将你命令及其执行结果截图放在下面：**

SELECT COUNT(工号) AS 人数 FROM teacher;



2）-- 查询老师的人数、总工资、平均工资、最高工资和最低工资



**将你命令及其执行结果截图放在下面：**

SELECT COUNT(工号) AS 人数,

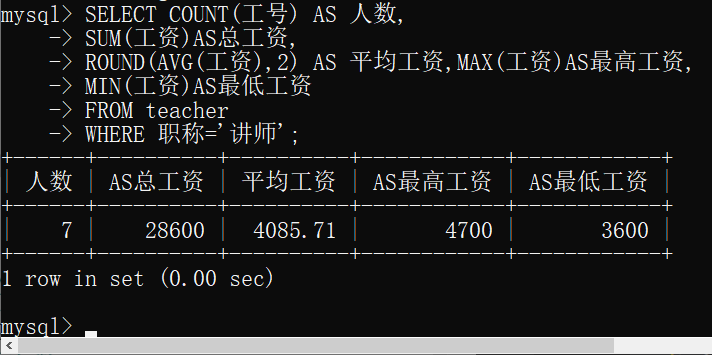
SUM(工资)AS总工资,

ROUND(AVG(工资),2) AS 平均工资,MAX(工资)AS最高工资,

MIN(工资)AS最低工资

FROM teacher

WHERE 职称='讲师';

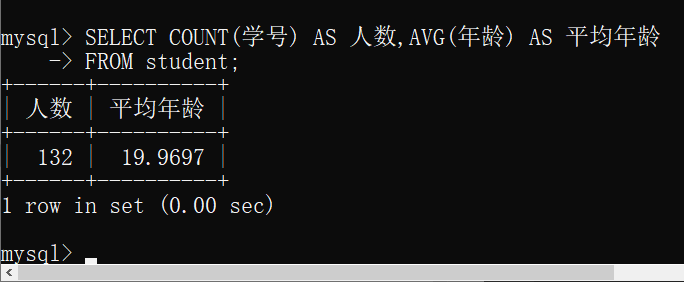


3）-- 查询计算机系学生的人数和平均年龄

**将你命令及其执行结果截图放在下面：**

SELECT COUNT(学号) AS 人数,AVG(年龄) AS 平均年龄

FROM student;



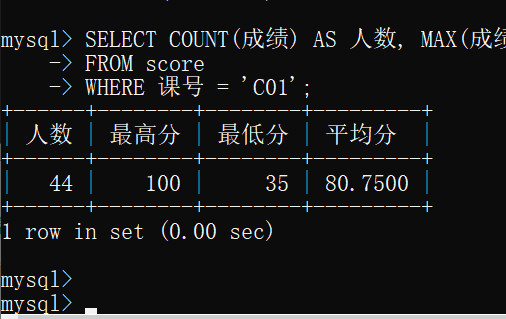
4）-- 查询学“C01”课程的学生的人数、最高分、最低分和平均分

**将你命令及其执行结果截图放在下面：**

SELECT COUNT(成绩) AS 人数, MAX(成绩) AS 最高分, MIN(成绩) AS 最低分, AVG(成绩) AS 平均分

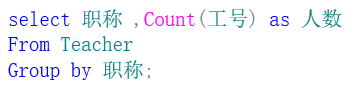
FROM score

WHERE 课号 = 'C01';



1. **分组统计查询练习**

1）-- 查询教师表中不同职称的人数

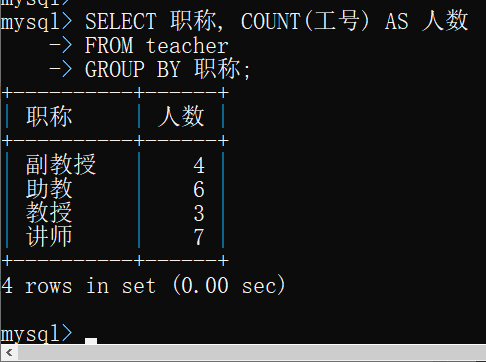


**将你命令及其执行结果截图放在下面：**

SELECT 职称, COUNT(工号) AS 人数

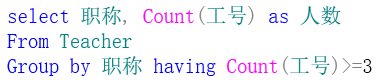
FROM teacher

GROUP BY 职称;



2）-- 查询教师人数在3人以上的职称及其人数

-- Having子句的作用

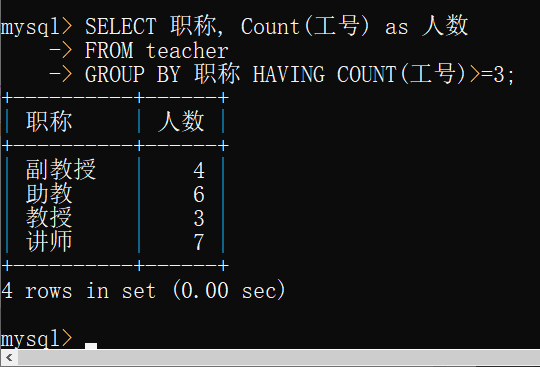


**将你命令及其执行结果截图放在下面：**

SELECT 职称, Count(工号) as 人数

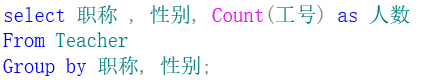
FROM teacher

GROUP BY 职称 HAVING COUNT(工号)>=3



3）-- 查询各职称的不同性别的教师人数

-- 多级分组的用法

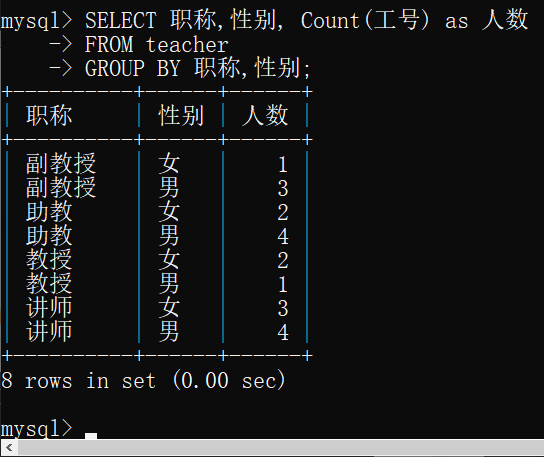


**将你命令及其执行结果截图放在下面：**

SELECT 职称,性别, Count(工号) as 人数

FROM teacher

GROUP BY 职称,性别;



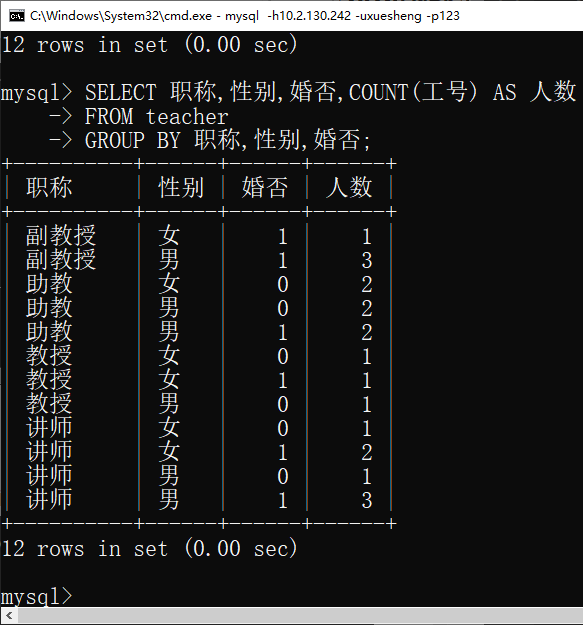
4）-- 查询各职称的不同性别的不同婚姻状况的人数。

**将你命令及其执行结果截图放在下面：**

SELECT 职称,性别,婚否,COUNT(工号) AS 人数

FROM teacher

GROUP BY 职称,性别,婚否;



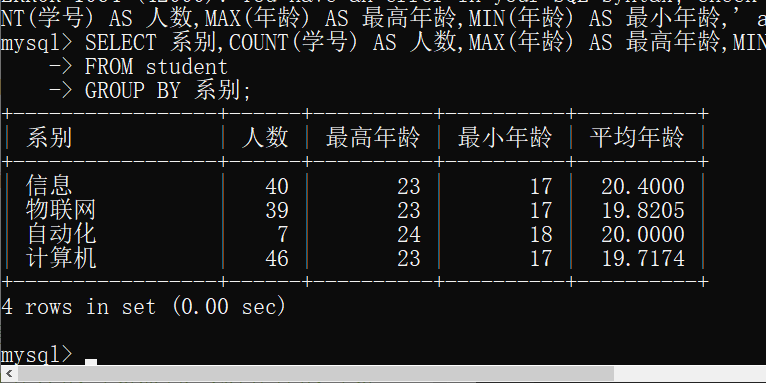
5）-- 查询各个系的男女学生的人数、最高年龄、最小年龄和平均年龄。

**将你命令及其执行结果截图放在下面：**

SELECT 系别,COUNT(学号) AS 人数,MAX(年龄) AS 最高年龄,MIN(年龄) AS 最小年龄,AVG(年龄) AS 平均年龄

FROM student

GROUP BY 系别;



6）-- 查询score表中每个课号有多少学生在学，平均分是多少。

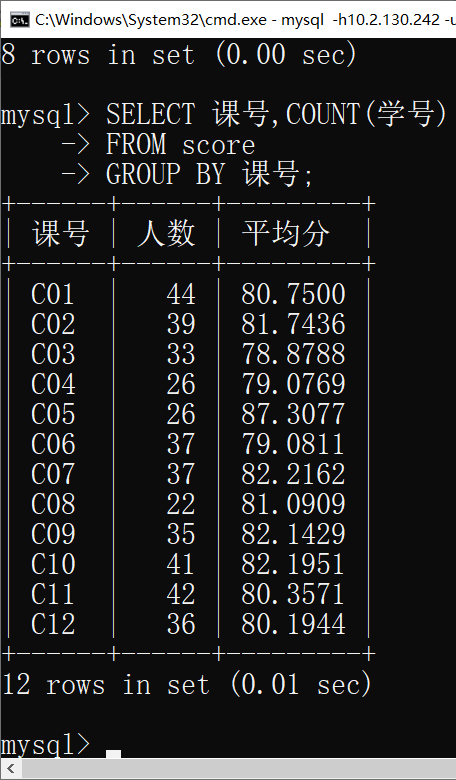
参考答案：结果有12行，其中C01有44人，平均80.75； C03有33人，平均78.8788……

**将你命令及其执行结果截图放在下面：**

SELECT 课号,COUNT(学号) AS 人数,AVG(成绩) AS 平均分

FROM score

GROUP BY 课号;



7）-- 查询score表中每个课程号不及格的人数是多少

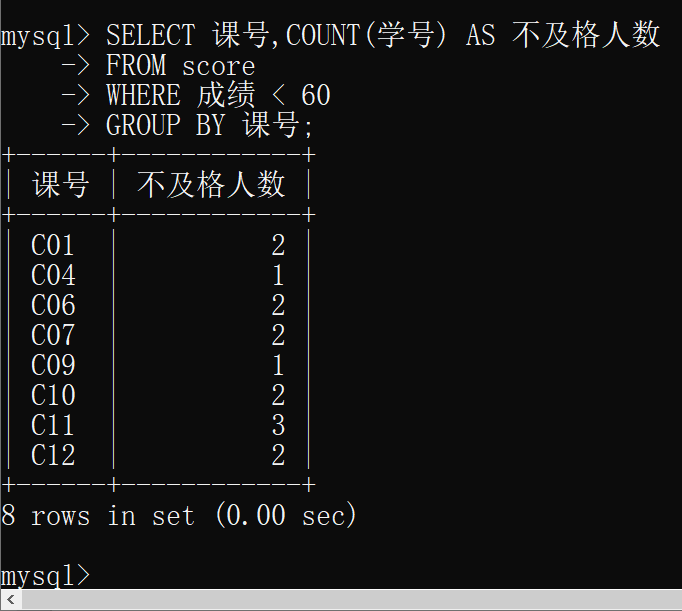
**将你命令及其执行结果截图放在下面：**

SELECT 课号,COUNT(学号) AS 不及格人数

FROM score

WHERE 成绩 < 60

GROUP BY 课号;



8）-- 查询score表中有哪些学号不及格的课程数超过了1门，他们不及格的课程数是多少？

参考答案：结果有1人，是S24，他有3门不及格

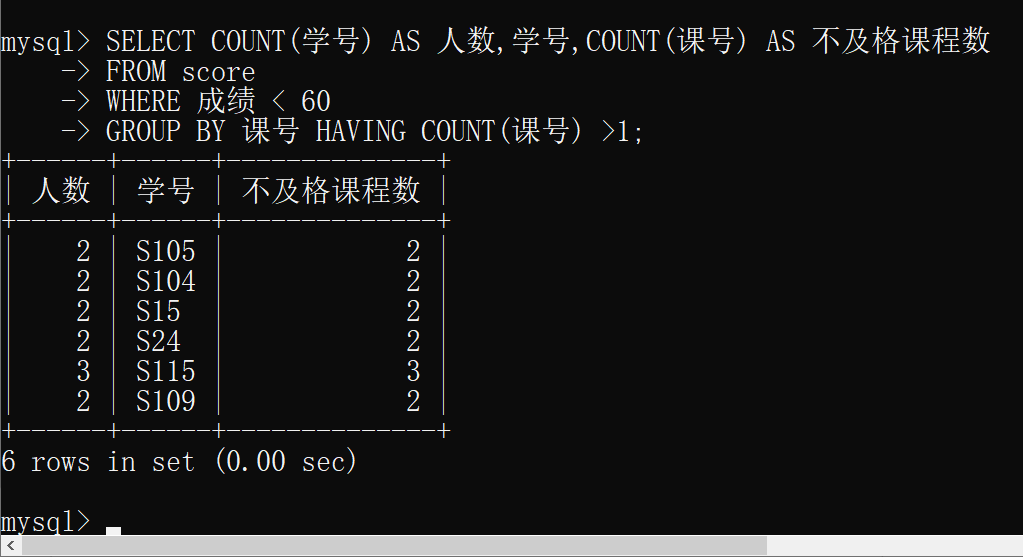
**将你命令及其执行结果截图放在下面：**

SELECT COUNT(学号) AS 人数,学号,COUNT(课号) AS 不及格课程数

FROM score

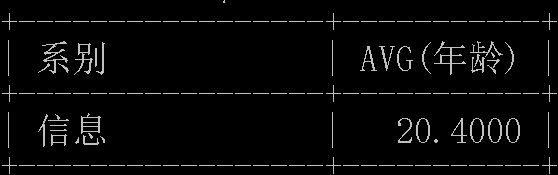
WHERE 成绩 < 60

GROUP BY 课号 HAVING COUNT(课号) >1;



9）-- 查询平均年龄最大的系的名字及其平均年龄值

提示：统计、排序、limit

结果：

**将你命令及其执行结果截图放在下面：**

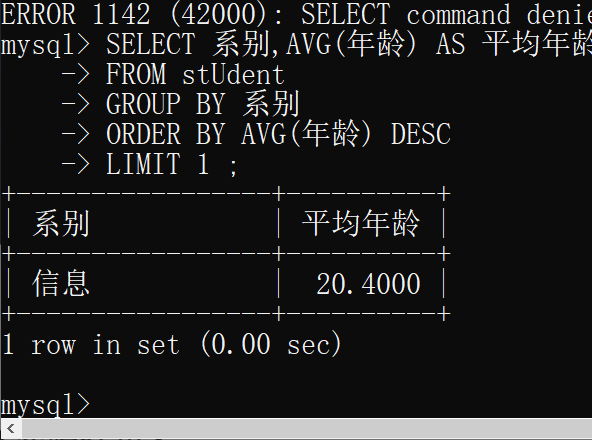
SELECT 系别,AVG(年龄) AS 平均年龄

FROM stUdent

GROUP BY 系别

ORDER BY AVG(年龄) DESC

LIMIT 1 ;



【实验结果】

**请将完成后的本实验报告提交到课程平台。**