实验4 SQL数据统计处理

* **实验环境：**

操作系统：Windows 10 专业版

数据库管理系统：8.0.18 MySQL Community Server - GPL

客户端：MySQL Workbench 8.0 Community+VScode 1.71.2

数据库模式：Test2

学生表：Student(Sno,Sname,Ssex,Sage,Sdept)

课程表：Course(Cno,Cname,Cpno,Ccredit)

选课表：SC(Sno,Cno,Grade)

数据库的值：如下图所示



1、在学生选课表中，统计学生平均分大于70的成绩记录。包括学生学号、总分、平均分、课程门数、课程最高分、最低分字段。

实验代码：

-- 1.在学生选课表中，统计学生平均分大于70的成绩记录。

-- 包括学生学号、总分、平均分、课程门数、课程最高分、最低分字段。

select Sno as 学号,

       sum(Grade) as 总分,

       floor(avg(Grade)) as 平均分,

       count(Grade) as 课程门数,

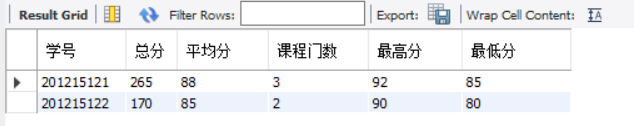
       max(Grade) as 最高分,

       min(Grade) as 最低分

from Sc

group by Sno

运行结果：





返回两条查询结果。

2、在学生选课表和学生表中，统计学生平均分在60~70和90~100的学生成绩记录，包括学生学号、学生姓名、总分、平均分、最高分、最低分、所修课程门数字段。

实验代码：

-- 2.在学生选课表和学生表中，统计学生平均分在60~70和90~100的学生成绩记录，

-- 包括学生学号、学生姓名、总分、平均分、最高分、最低分、所修课程门数

select Sc.Sno as 学号,

       Sname as 姓名,

       sum(Grade) as 总分,

       avg(Grade) as 平均分,

       max(Grade) as 最高分,

       min(Grade) as 最低分,

       count(Grade) as 所修课程门数

from Student,Sc

where Student.Sno=Sc.Sno

group by Sc.Sno

having avg(Grade) between 60 and 70 AND avg(Grade) between 90 and 100

运行结果：



3、在学生选课表和学生表中，查询性别为女并且平均分大于80的学生基本信息。

实验代码：

-- 3.在学生选课表和学生表中，查询性别为女并且平均分大于80的学生基本信息。

select Student.\*

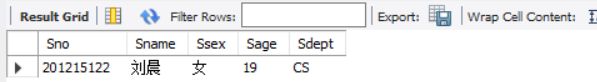
from Student,Sc

where Student.Sno=Sc.Sno

group by Sc.Sno

having Ssex='女' and avg(Grade)>80

运行结果：





4、统计平均分各分数段人数。

实验代码：

-- 4.统计平均分各分数段人数

-- 先创建一个新表，名为ScoreInfo

create table ScoreInfo

(

 分数段 char(30),

 人数 smallint

);

-- 向表中插入查询结果

insert into scoreinfo(分数段,人数)

select '60分以下',count(case  when grade<60 then 1 end)

from (select sno,avg(grade) from sc group by sno)as ScoreInfo(sno,grade);

insert into scoreinfo(分数段,人数)

select '60-80',count(case when grade >=60 and grade<80 then 1 end)

from (select sno,avg(grade) from sc group by sno)as ScoreInfo(sno,grade);

insert into scoreinfo(分数段,人数)

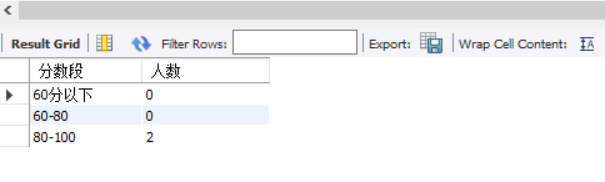
select '80-100', count(case when grade>= 80 and grade<=100 then 1 end)

from (select sno,avg(grade) from sc group by sno)as ScoreInfo(sno,grade);

-- 查询新表内容

select \* from scoreinfo

运行结果：



5.在学生选课表、学生表中，查询学生成绩重修（成绩<60）门数大于10门的学生基本信息（查询结果包括学号、姓名、性别字段）。

实验代码：

-- 5.在学生选课表、学生表中，查询学生成绩重修（成绩<60）门数大于10门的学生基本信息

-- （查询结果包括学号、姓名、性别字段）。

select Sno,Sname,Ssex

from Student

where Sno in

(   -- 子查询结果为重修门数大于10的学号

    select Sno from Sc

    where Grade<60

    group by Sno

    having count(\*)>10 -- 小于60分的门数

)

-- 外嵌套的查询为该学生学号是否在子查询的集合中

运行结果：





6. 统计每门课程成绩均在80分以上的各学生平均分。

实验代码：

-- 6.统计每门课程成绩均在80分以上的各学生平均分。

select Sno as sno,

       floor(avg(Grade)) as 平均分

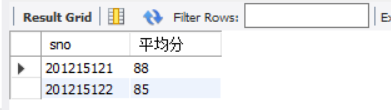
from Sc

group by Sno

having min(Grade)>=80

-- 最低的成绩如果也在80分以上，那所有成绩一定都在80分以上

运行结果：



* 实验总结：

1.使用聚集函数可以将一些查询操作简单化，例如求和、计数、平均值以及最小最大值都有相关的函数操作。统计某一列或者元组的一些信息。

2.注意where中不能有聚集函数的存在，若使用会导致报错。正确方式是使用having代替相关条件查询。

3.当一层查询无法满足要求时，可以考虑嵌套查询，即先进行子查询找到一个所有满足条件的集合，然后外部查询将在该集合中的元素筛选出来，这是一个好的思路。

4.“每门课程成绩均在80分以上”等同于“最小成绩在80分以上”

5.看清楚参考实验结果里输出的内容都有哪些，以及属性名有无重命名，明察秋毫。