**第四次上机练习**

#### 在完成本实验之后，用户将完成以下任务：

#### 掌握复杂数据查询操作；

#### 掌握子查询操作。

#### 掌握数据操作的高级应用

#### 掌握视图、存储过程的创建

**练习一、自己创建数据库SampleDB，该数据库包括以下5个表格（创建脚本在后面）**

**G\_stuinfo Exam\_Score**

|  |  |  |
| --- | --- | --- |
| 主键 | 字段名 | 描述 |
| ● | TermID | 学期代码 |
| ● | StuID | 学生学号 |
| ● | CourseID | 课程代码 |
|  | Score | 成绩 |

|  |  |  |
| --- | --- | --- |
| 主键 | 字段名 | 描述 |
| ● | StuID | 学生学号 |
|  | StuName | 学生姓名 |
|  | SexCode | 性别代码 |
|  | ClassID | 专业班级 |
|  | SpecialityID | 专业代码 |

|  |  |  |
| --- | --- | --- |
| 主键 | 字段名 | 描述 |
| ● | CourseID | 课程代码 |
|  | Course | 课程名称 |
|  | CreditHour | 学分 |

G\_SpecialityInfo

|  |  |  |
| --- | --- | --- |
| 主键 | 字段名 | 描述 |
| ● | SpecialityID | 专业代码 |
|  | SpecialityName | 专业名称 |
|  | StudyYear | 学年 |
|  | Memo | 备注 |

**G\_CourseInfo**

**G\_SexInfo**

**G\_SexInfo**

|  |  |  |
| --- | --- | --- |
| 主键 | 字段名 | 描述 |
| ● | SexCode | 性别代码 |
|  | SexName | 性别名称 |

**参考以下表和数据，检查自己的查询语句是否正确**

Create database SampleDB

COLLATE Chinese\_PRC\_CI\_AS

go

Use SampleDB

go

create table G\_SexInfo(

sexcode char(1) not null primary key,

sexname char(2)

);

create table G\_SpecialityInfo(

Specialityid char(10) not null primary key,

Specialityname varchar(20) not null,

studyyear tinyint,

memo varchar(200)

);

create table G\_stuinfo(

stuid char(10) not null primary key,

stuname varchar(10) not null,

sexcode char(1) default '1' references g\_SexInfo(sexcode ) ,

classid char(10),

specialityid char(10) references g\_SpecialityInfo(specialityid )

);

create table G\_CourseInfo(

Courseid char(10) not null primary key,

course varchar(20) not null,

credithour tinyint

);

create table Exam\_Score(

termid char(3) not null,

stuid char(10) not null references G\_stuinfo(stuid),

courseid char(10) not null references g\_CourseInfo(Courseid),

score tinyint,

constraint pk\_score primary key(termid,stuid,courseid)

);

insert into G\_CourseInfo values('10000005','计算机数学','4');

insert into G\_CourseInfo values('10000006','程序设计基础','4');

insert into G\_CourseInfo values('10000007','数学','4');

insert into G\_CourseInfo values('10000008','大学英语','4');

insert into G\_SexInfo values('1','男');

insert into G\_SexInfo values('0','女');

insert into G\_SpecialityInfo values('001','软件','3',null);

insert into G\_SpecialityInfo values('002','网络','3',null);

insert into G\_SpecialityInfo values('003','电子商务','3',null);

insert into G\_Stuinfo values('95001','张三','1','111','001');

insert into G\_Stuinfo values('95002','王强','1','111','001');

insert into G\_Stuinfo values('95003','王力','1','111','001');

insert into G\_Stuinfo values('95004','周刚','1','111','001');

insert into G\_Stuinfo values('95005','周小岚','0','111','001');

insert into G\_Stuinfo values('95006','吴波','1','112','002');

insert into G\_Stuinfo values('95007','郑棋','1','112','002');

insert into G\_Stuinfo values('95008','郑红','0','112','002');

insert into G\_Stuinfo values('95009','郑红','0','112','002');

insert into G\_Stuinfo values('95010','吴波','0','112','002');

insert into G\_Stuinfo values('95011','李芳','0','113','003');

insert into G\_Stuinfo values('95012','陈婷','0','113','003');

insert into G\_Stuinfo values('95013','吴平','1','113','003');

insert into G\_Stuinfo values('95034','吴平','1','113','003');

insert into G\_Stuinfo values('95015','郑红','0','113','003');

insert into G\_Stuinfo values('95016','蔡小明','0','113','003');

insert into Exam\_Score values('071','95001','10000005','65');

insert into Exam\_Score values('071','95002','10000005','88');

insert into Exam\_Score values('071','95003','10000005','97');

insert into Exam\_Score values('071','95004','10000005','76');

insert into Exam\_Score values('071','95005','10000005','40');

insert into Exam\_Score values('071','95006','10000006','63');

insert into Exam\_Score values('071','95007','10000006','85');

insert into Exam\_Score values('071','95008','10000006','93');

insert into Exam\_Score values('071','95009','10000006','100');

insert into Exam\_Score values('071','95010','10000006','74');

insert into Exam\_Score values('071','95011','10000007',null);

insert into Exam\_Score values('071','95012','10000007','99');

insert into Exam\_Score values('071','95013','10000007','87');

insert into Exam\_Score values('071','95034','10000008','70');

insert into Exam\_Score values('071','95015','10000008','55');

insert into Exam\_Score values('071','95016','10000008',null);

insert into Exam\_Score values('071','95006','10000005','60');

insert into Exam\_Score values('071','95007','10000005','80');

insert into Exam\_Score values('071','95008','10000005','93');

insert into Exam\_Score values('071','95009','10000005','80');

insert into Exam\_Score values('071','95010','10000005','74');

insert into Exam\_Score values('071','95001','10000006','30');

insert into Exam\_Score values('071','95002','10000006','99');

insert into Exam\_Score values('071','95003','10000006','46');

insert into Exam\_Score values('071','95004','10000006','69');

insert into Exam\_Score values('071','95005','10000006','80');

Part 1：

**（1）统计每个专业班级的课程代码为10000005的考试平均分，并按从高到低排序**



**（2）统计每个专业的’程序设计基础’考试平均分，并按从高到低排序**



**（3）查询学生“张三”所在专业的总人数**



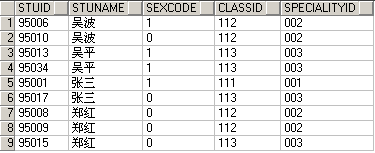
**考虑：可能有多个叫张三的学生，分别在不同的专业，你处理这种情况了吗？**

**（执行：insert into G\_Stuinfo values('95017','张三','0','113','003');）**



**（4）查询重名的学生信息**

**注意：处理多个学生的重名情况**



**（5）查询数学考试成绩超过75分的男学生的总人数(要求关联g\_Sexinfo)**



**（6）查询”王”姓学生的’计算机数学’最高分**



**（7）查询姓是”周”,”吴”,”郑”,”王”的男学生数量**



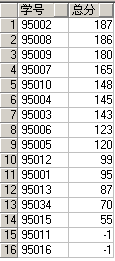
**（8）查询缺考学生的姓名**



**（9）查询学号倒数第二位为3，专业为电子商务的男学生**



**（10）查询所有学生的总分，并按从大到小进行排序。**



**Part 2：**

**（1）在选修了“计算机数学”这门课的学生中，将考试成绩高于该课程平均分的学生的本课程的成绩上调2分。**

**（2）张三同学转专业到“电子商务”专业。**

**（3）将选课人数低于15人的课程的学分下调1分。**

**Part 3：**

1. **创建视图vw\_CourceInfo，查看所有课程的选课人数、平均分，若无人选课则显示0。**
2. **创建视图vw\_Top1perSpecial，查看每个专业总分排名第一的同学学号、姓名。**
3. **创建存储过程up\_GetSCinfoBySno，根据学生的学号，查看该学号同学的选课情况，显示选课名、成绩。**
4. **创建存储过程up\_GetScoreByName，根据学生姓名、课程名，查看该学生该课程的成绩。**
5. **创建存储过程up\_GetInfoBySpecialID，根据专业代码，查看该专业的男生人数、女生人数。**
6. **创建存储过程up\_AddStudent，根据提供的学生学号、姓名、性别、班级号、专业名，将该学生信息插入到数据表G\_stuinfo中。**

**该存储过程的执行脚本为：**

**exec up\_AddStudent '96001','王倩','女','211','网络'**

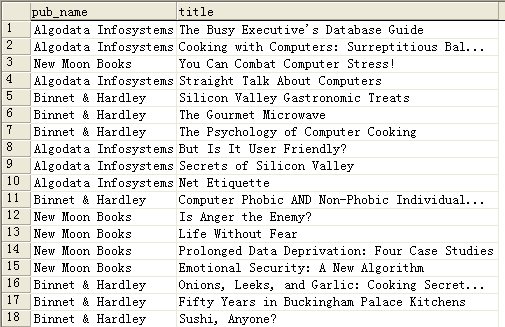
1. **创建存储过程up\_UpdateScoreByName，根据学生学号、课程名，修改该学生该课程的成绩为新的成绩（新成绩通过参数传递）。**

**该存储过程的执行脚本为：**

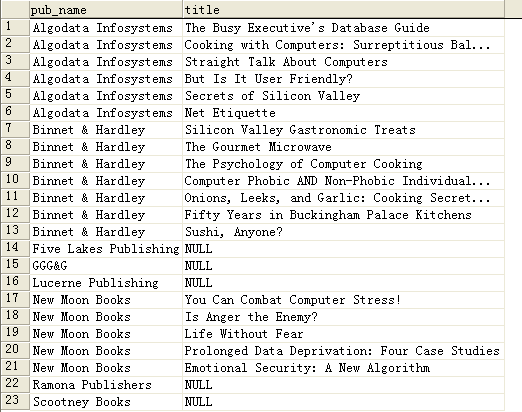
**exec up\_UpdateScoreByName ‘95001’,’计算机数学’,85**

**练习二、使用pubs数据库**

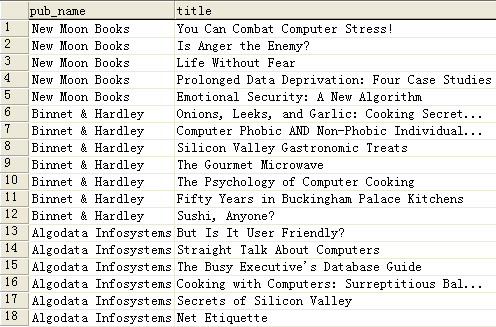
1. 查找出版过书籍的出版商名及其出版的书籍名称



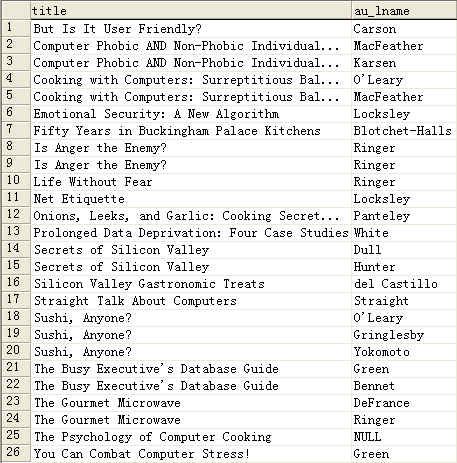
1. 查找所有出版商及其出版的书籍名，按出版商姓名排序



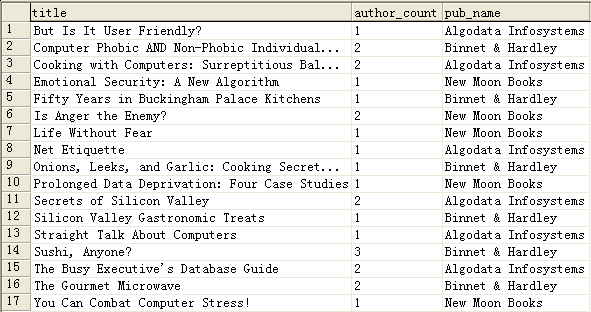
3、查找所有书籍的名称及其出版商名



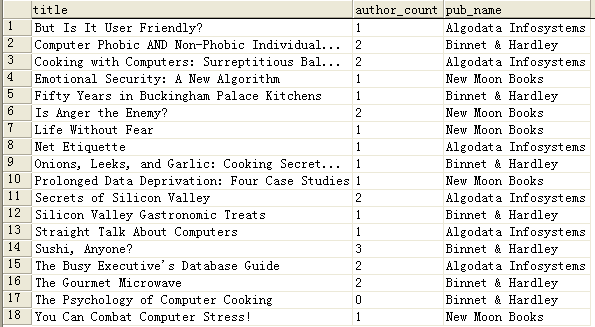
4、查询所有书籍的作者（如果一本书有多个作者，请都列出来）



5、查询具有作者和出版商的书籍的名称，及其该书的作者人数和该书的出版商。



6、查询所有书籍的作者人数、出版商姓名及书籍名称。

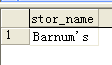


**补充提高题（可选做）**

\*1.查询出版书最多的出版商名



\*2.查询1994年9月份销售量最大的书店



\*\*3.查询1993、1994年度销售量最大的书店（提示：用year（）函数提取年份）

