南京航空航天大学

实验报告

题 目 学生选课管理系统设计

学生姓名 李应飞

学 号 161610338

学 院 计算机科学与技术学院

专 业 计算机科学与技术

班 级 1616103

指导教师 郑吉平

二〇一九年六月

学生选课管理系统的设计

# 摘 要

为了提升对数据库的理解能力和动手能力，以个人为单位，进一步在实验一到五的基础上，进一步完成一个学生选课管理系统的设计 。

研究的结果与主要结论：完成学生选课管理系统的初步开发，集成数据的增加、删除、修改和查询等基本功能，实现良好的用户界面。提供高级查询，能够对比数据库查询语句的时间，从而实现查询优化。

关键词：数据库，处，用户界面，高级查询，优化

The Design of a Management System for Student Course Selection

Abstract

In order to improve the understanding ability and practical ability of the database, the design of a student course selection management system is further completed on the basis of Experiment 1 to 5.

The results and main conclusions of the study are as follows: the preliminary development of the student course selection management system is completed, and the basic functions of data addition, deletion, modification and query are integrated to achieve a good user interface. Provide advanced query, which can compare the time of database query statements, so as to achieve query optimization.

Key Words：Database, Basic Functions, User Interface, Advanced Query, Optimization

**目录**

[摘 要 2](#_Toc12226940)

[一、 实验任务及目的 5](#_Toc12226941)

[1.1开发运行环境 5](#_Toc12226942)

[1.2. 实验目的 5](#_Toc12226943)

[1.3. 实验要求 5](#_Toc12226944)

[二、 数据库的详细设计 5](#_Toc12226945)

[2.1数据库设计 5](#_Toc12226946)

[2.2数据库表的详细设计 6](#_Toc12226947)

[2.2.1账号和密码表设计 6](#_Toc12226948)

[2.2.2学生信息表设计 6](#_Toc12226949)

[2.2.3选课成绩表设计 6](#_Toc12226950)

[2.2.4课程表设计 7](#_Toc12226951)

[2.2.5可选课程表设计 7](#_Toc12226952)

[三、 模块、函数结构设计 8](#_Toc12226953)

[3.1模块之间的关系 8](#_Toc12226954)

[3.2函数结构设计举例 9](#_Toc12226955)

[3.2.1登录（login.py） 9](#_Toc12226956)

[3.2.2教师管理（index.py） 10](#_Toc12226957)

[3.2.3学生管理（index\_stu.py 10](#_Toc12226958)

[四、 主要功能分析及界面设计 11](#_Toc12226959)

[4.1开始界面 11](#_Toc12226960)

[4.1.1登录界面 11](#_Toc12226961)

[4.1.2注册界面 12](#_Toc12226962)

[4.1.3修改密码界面 13](#_Toc12226963)

[4.2教师界面 14](#_Toc12226964)

[4.2.1学生信息管理 15](#_Toc12226965)

[4.2.2课程信息管理 16](#_Toc12226966)

[4.2.3学生选课管理 18](#_Toc12226967)

[4.2.4可选课程 19](#_Toc12226968)

[4.3学生界面 19](#_Toc12226969)

[4.3.1修改学生信息 20](#_Toc12226970)

[4.3.2查询课程成绩 21](#_Toc12226971)

[4.3.3选课 21](#_Toc12226972)

[4.3.4查询已选课程 22](#_Toc12226973)

[五、 扩展功能（查询优化） 22](#_Toc12226974)

[5.1数据生成 22](#_Toc12226975)

[5.2数据量 23](#_Toc12226976)

[5.3对IN语句的优化 23](#_Toc12226977)

[5.3.1高级查询 23](#_Toc12226978)

[5.3.2三条语句比较 24](#_Toc12226979)

[5.3.3结论 26](#_Toc12226980)

[5.4生成海量数据代码及时间的计算代码 26](#_Toc12226981)

[5.4.1 生成S表数据：（random\_date\_S.py） 26](#_Toc12226982)

[5.4.2 生成SC表数据：（random\_date\_SC.py） 27](#_Toc12226983)

[5.4.3时间的计算：（height\_search.py） 29](#_Toc12226984)

[六、 心得体会 30](#_Toc12226985)

[参 考 文 献 31](#_Toc12226986)

# 实验任务及目的

## 1.1开发运行环境

Pycharm+Mysql+ tkinter（图形界面）

## 1.2. 实验目的

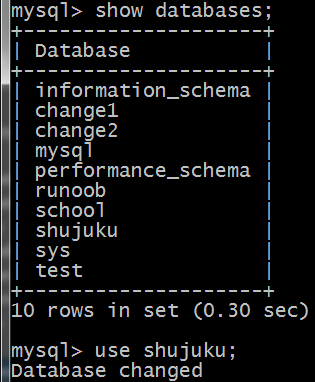
设计和实现学生通讯录或学生选课或类似的一个小型管理信息系统

## 1.3. 实验要求

要求具有数 据的增加、删除、修改和查询的基本功能，并尽可能提供较多的查询功能，集成 一半以上实验一～实验五的功能，用户界面要友好。可选内容：数据库中存放 100 万条记录，测试访问时间；如效率较低，提供优化方案

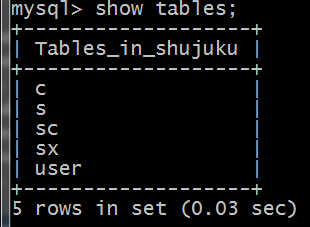
# 数据库的详细设计

## 2.1数据库设计



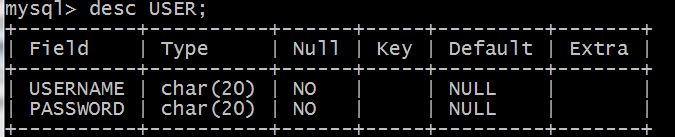
## 2.2数据库表的详细设计

字段，类型以及主键外键在截图中已显示



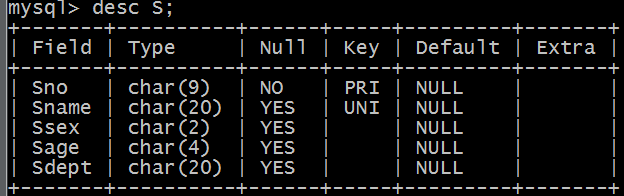
### 2.2.1账号和密码表设计

USER（USERNAME，PASSWORD）用户（用户名，密码）



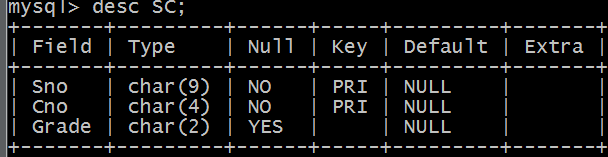
### 2.2.2学生信息表设计

S(Sno,Sname,Ssex,Sage,Sdept) 学生（学号，姓名，性别，年龄，系）



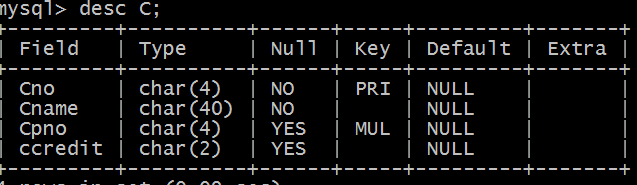
### 2.2.3选课成绩表设计

SC(Sno,Cno,Grade) 选课（学号，课程号，成绩）



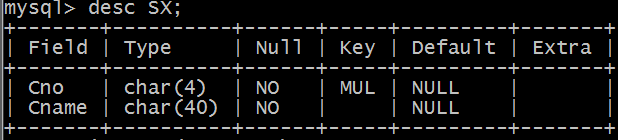
### 2.2.4课程表设计

C(Cno,Cname,Cpno,Ccredit) 课程（课程号，课程名，先行课，学分）



### 2.2.5可选课程表设计

SX(Cno,Cname) 可选课程（课程号，课程名）



# 模块、函数结构设计

## 3.1模块之间的关系

## 3.2函数结构设计举例

### 3.2.1登录（login.py）

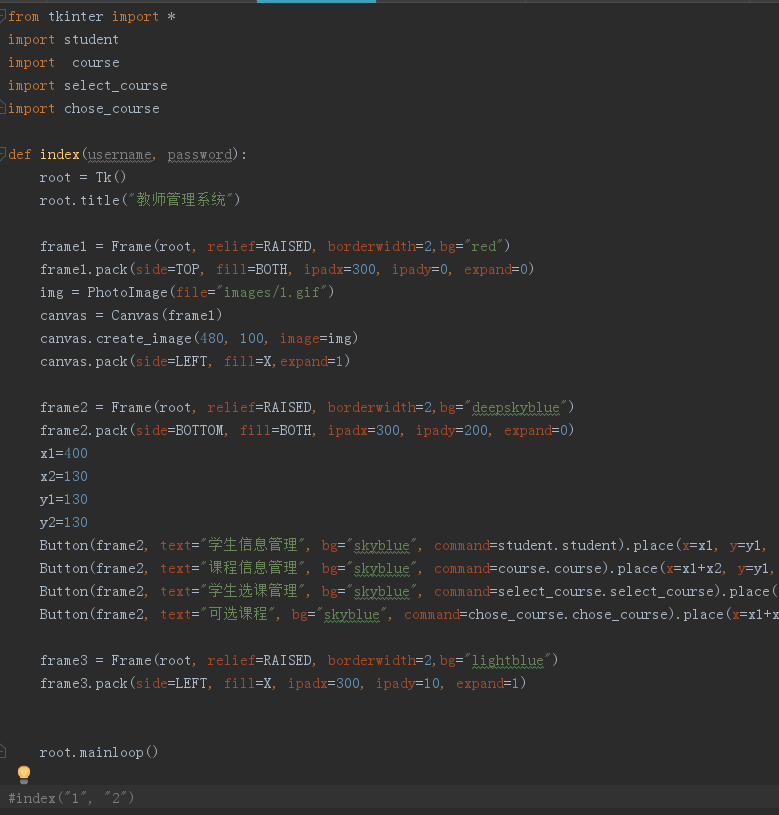


账号密码输入框

注册、教师登录、学生登录、修改密码四个按钮

登录函数，验证账号、密码的正确性

### 3.2.2教师管理（index.py）



点击按钮，利用command转到处理的函数去

窗体三，空白

窗体二，放置四个功能模块

窗体1，放置图片

3.2.3学生管理（index\_stu.py）



点击按钮，利用command转到处理的函数去

窗体三，空白

窗体二，放置四个功能模块

窗体1，放置图片

# 主要功能分析及界面设计

## 4.1开始界面

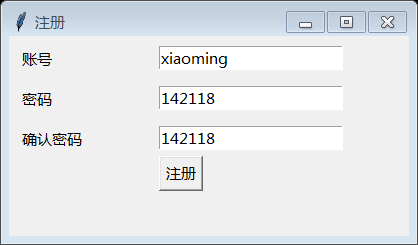
实现用户（教师，学生）的登录、注册及密码修改

### 4.1.1登录界面

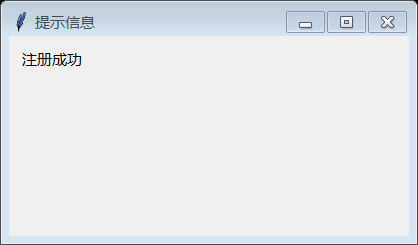
当输入账号和密码后，可以以教师和学生（账号一般为学号）两种身份登录。



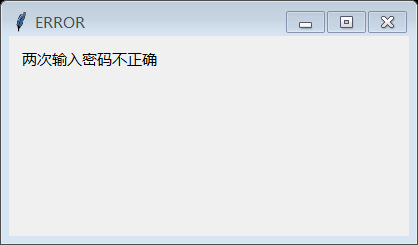
### 4.1.2注册界面



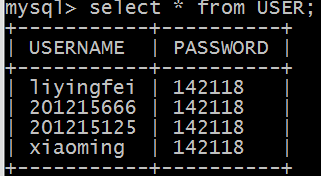
提示信息：

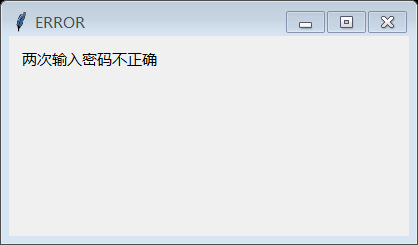


如果出错：

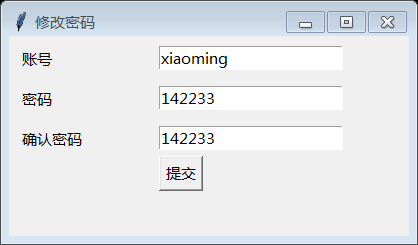


查表：

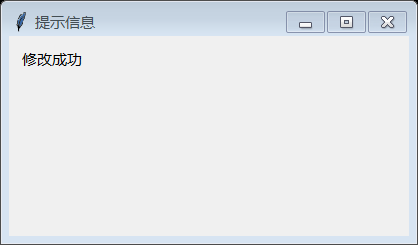




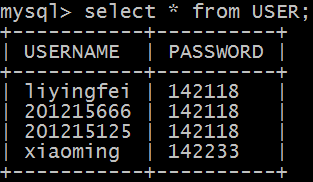
### 4.1.3修改密码界面



提示：



查看是否修改成功：



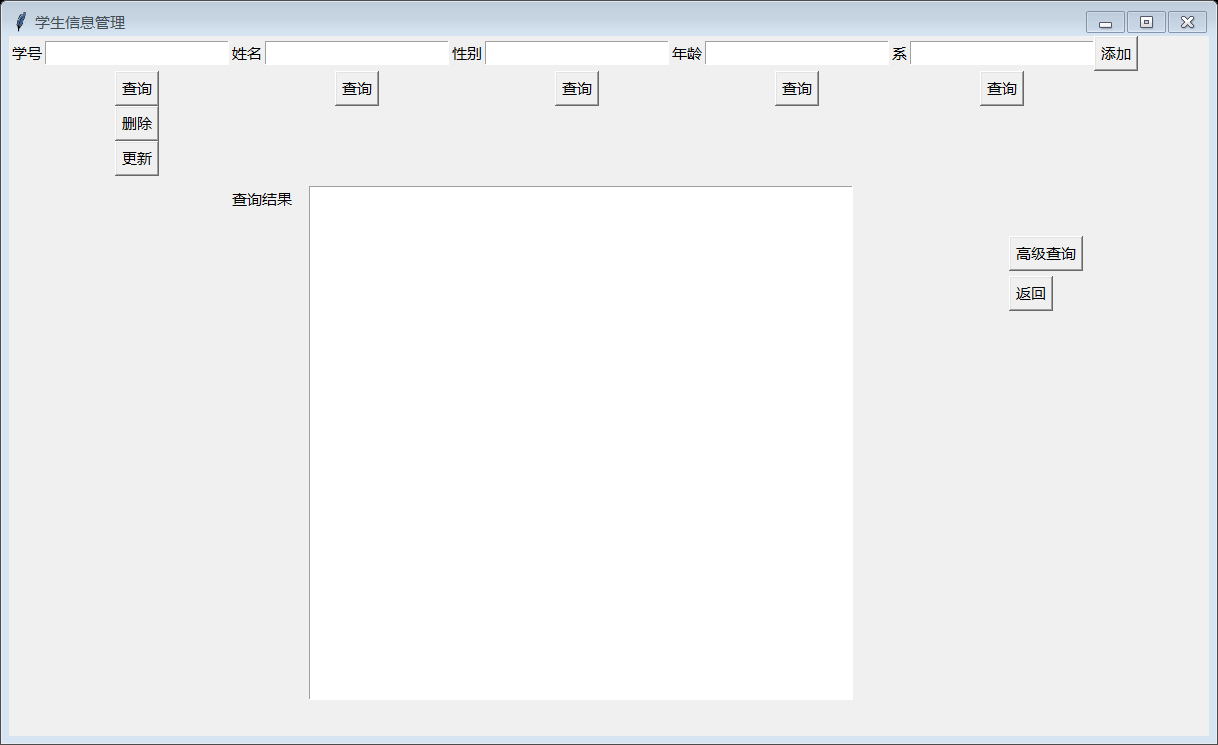
## 4.2教师界面

教师登录之后，可以对学生信息、课程信息、学生选课及可选课程的增删改查

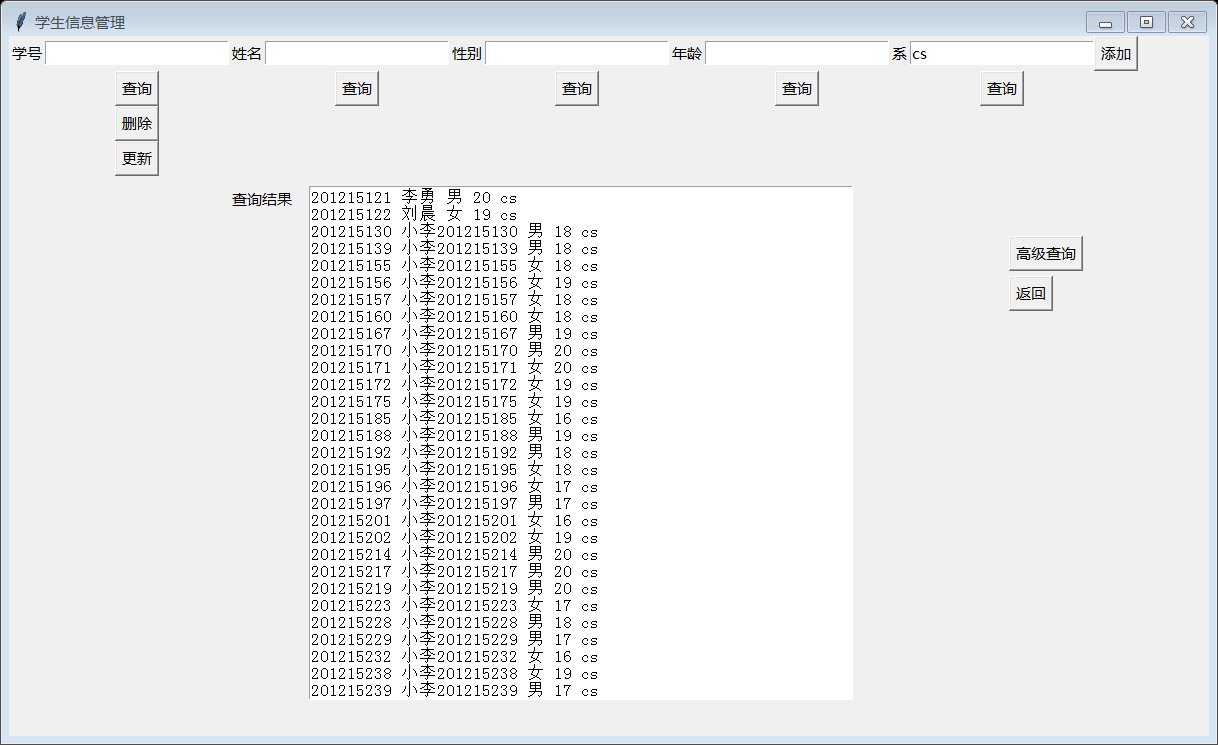


### 4.2.1学生信息管理

对每一个属性都有相应的一个查询按钮。删除与更新则以学号为依据。



查询演示（每个查询的依据是对应的上面输入框）：

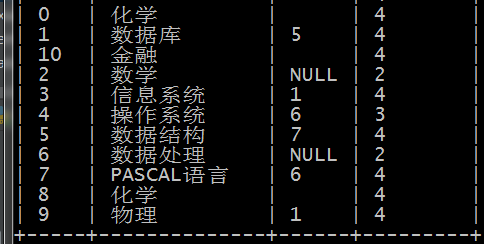


### 4.2.2课程信息管理



添加演示：



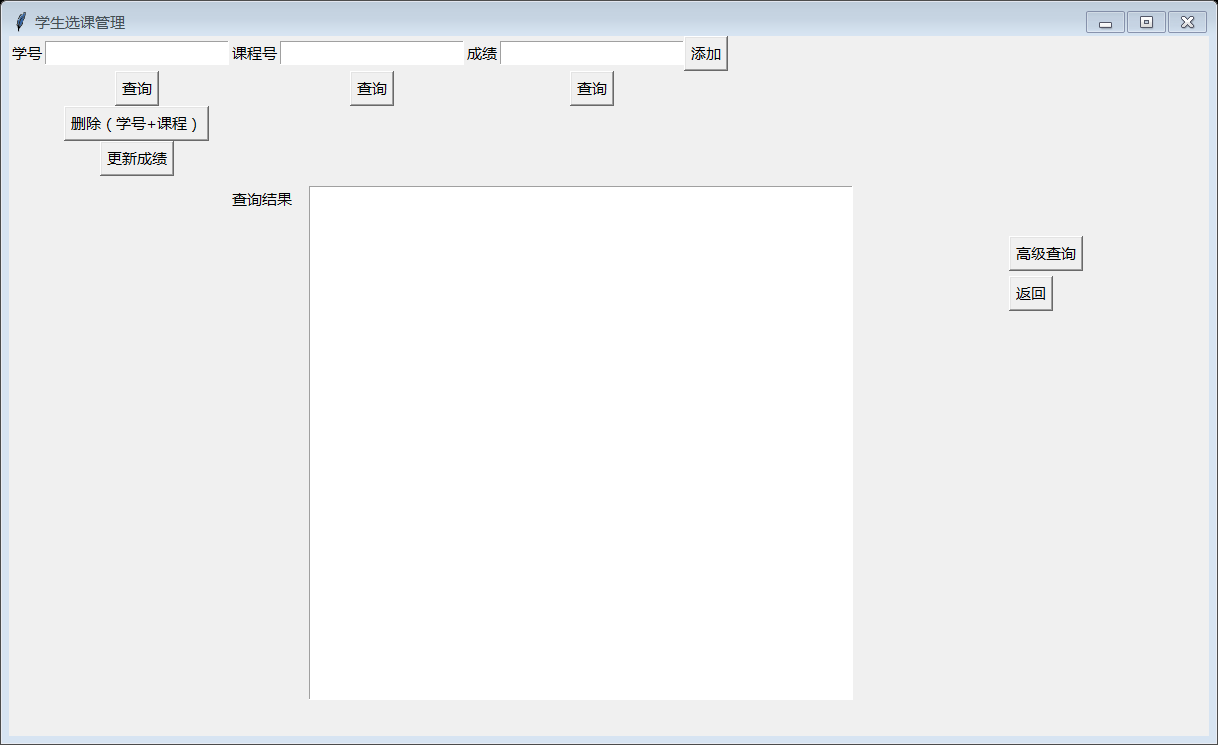


查询演示：

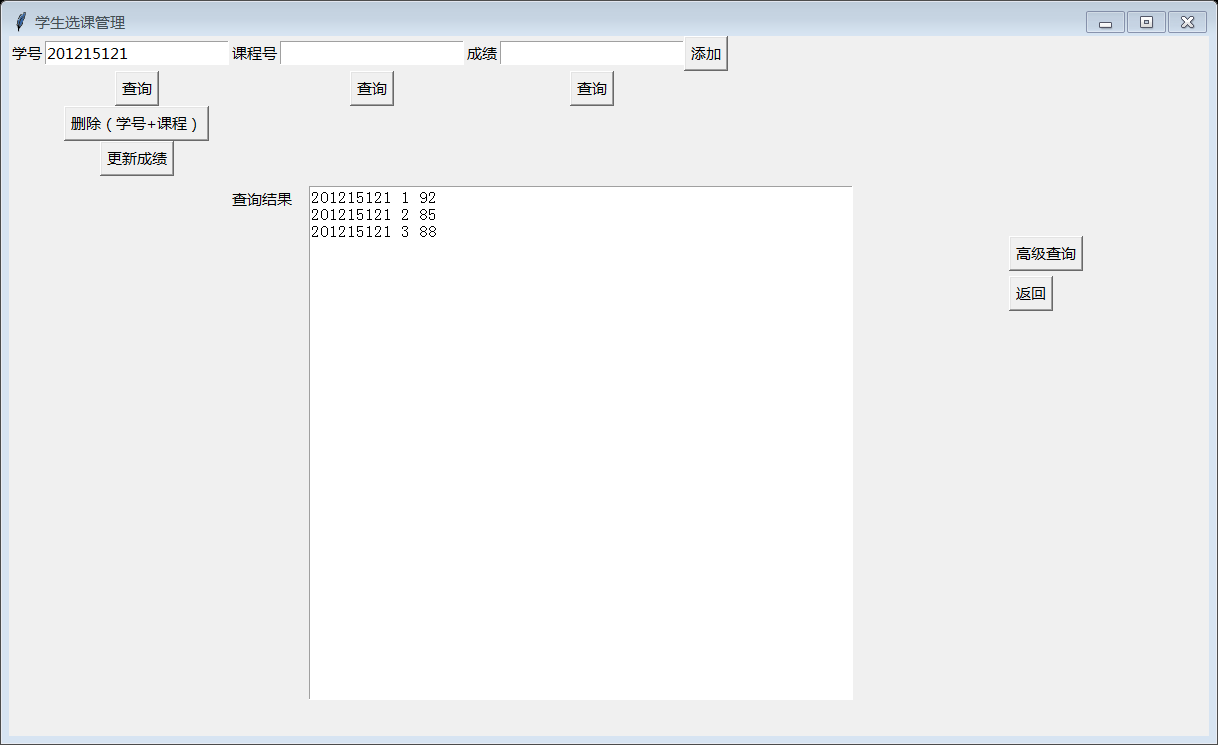




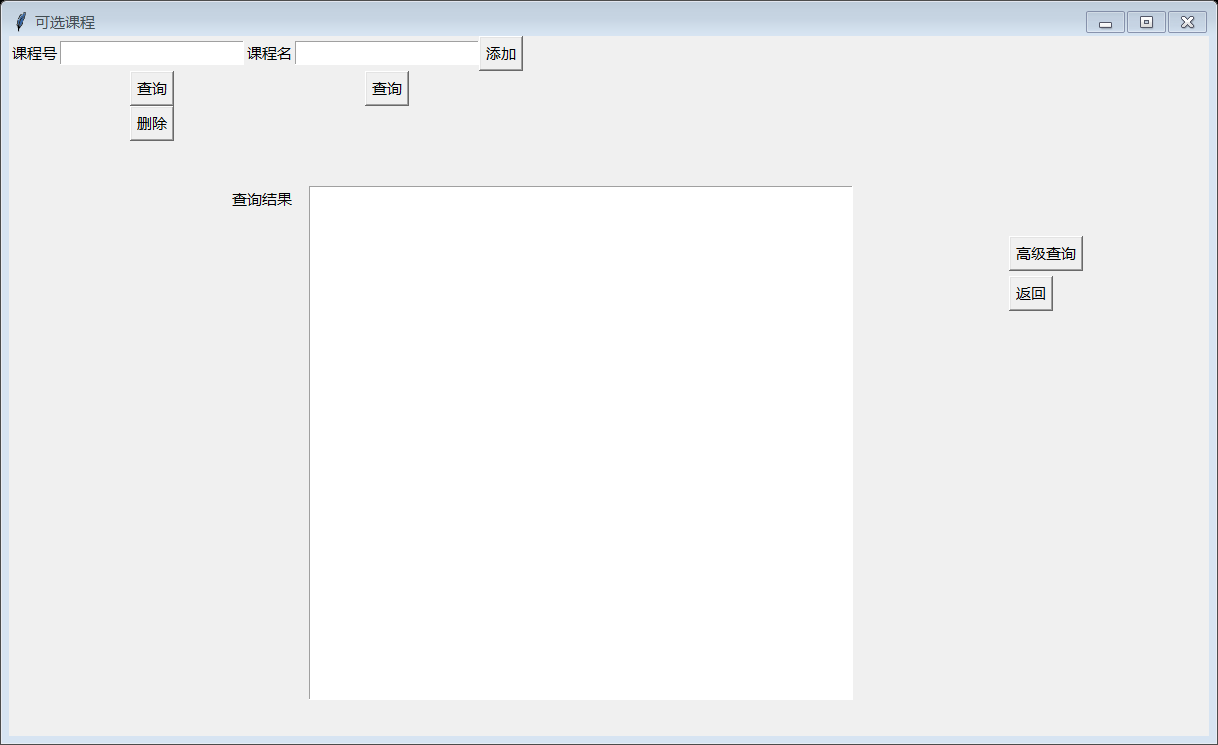
### 4.2.3学生选课管理



查询演示：



### 4.2.4可选课程



## 4.3学生界面

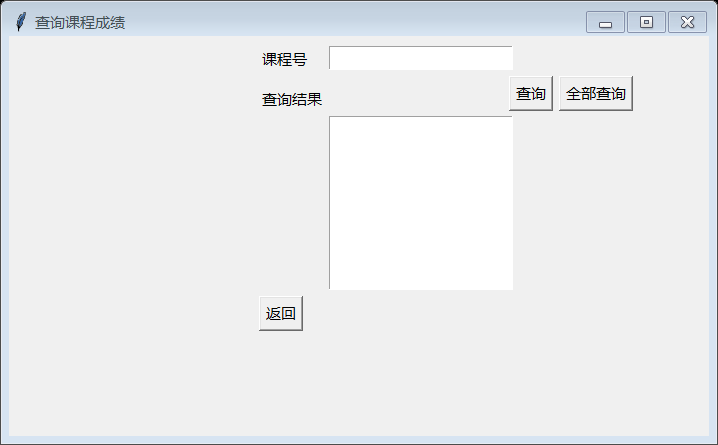
学生登录后，可以对自己的信息、课程成绩、选课及已选课程的增删改查等操作



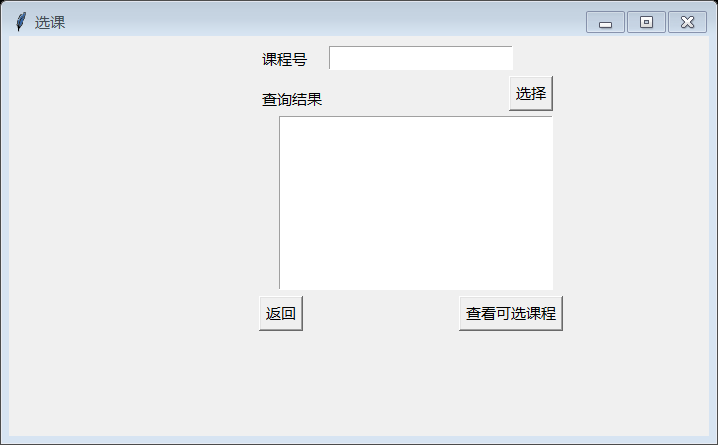
### 4.3.1修改学生信息



### 4.3.2查询课程成绩



### 4.3.3选课



### 4.3.4查询已选课程

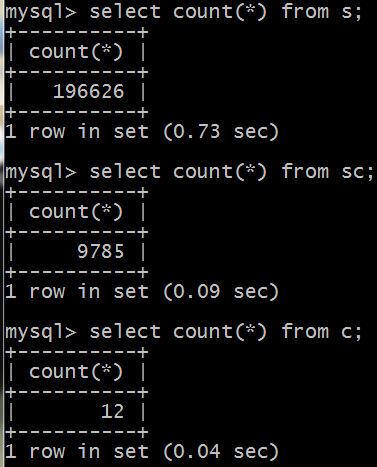


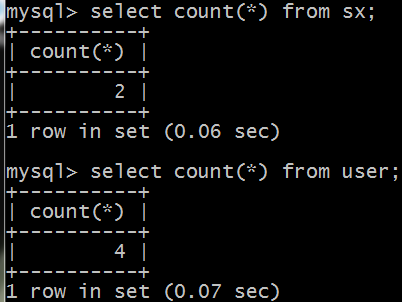
# 扩展功能（查询优化）

## 5.1数据生成

对S表和SC两个表使用随机函数生成记录然后一条一条的插。可以批量插入。但我试了，没有成功。

## 5.2数据量



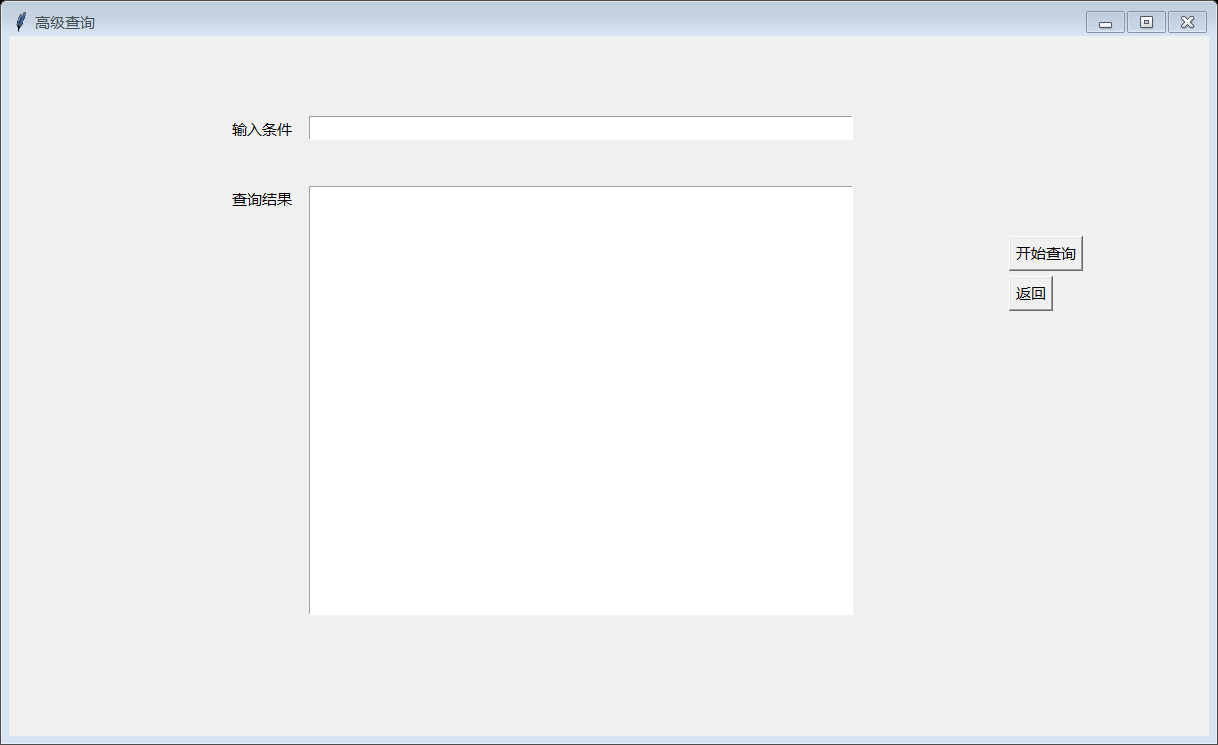


## 5.3对IN语句的优化

### 5.3.1高级查询

每个教师管理的模块有一个高级查询，能够根据输入的sql语句，查询我们想要的结果。就如同excel里能够用公式来筛选数据

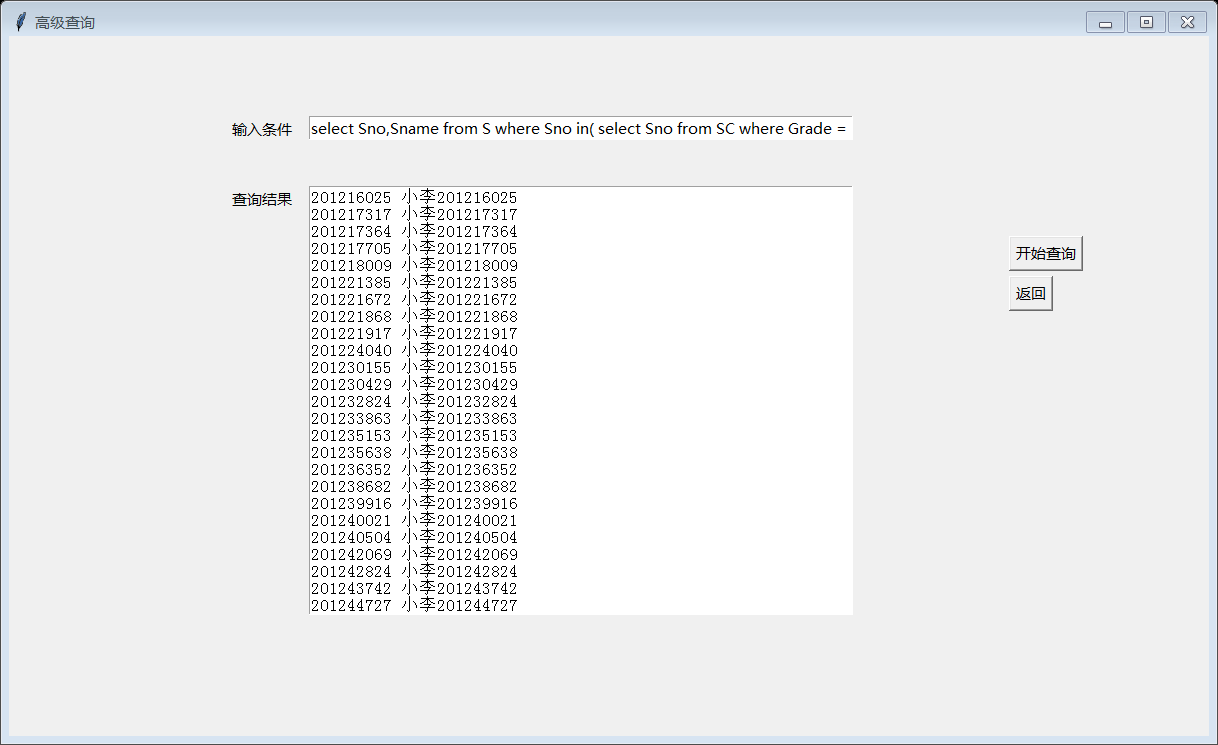
界面如下：

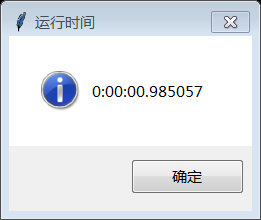


### 5.3.2三条语句比较

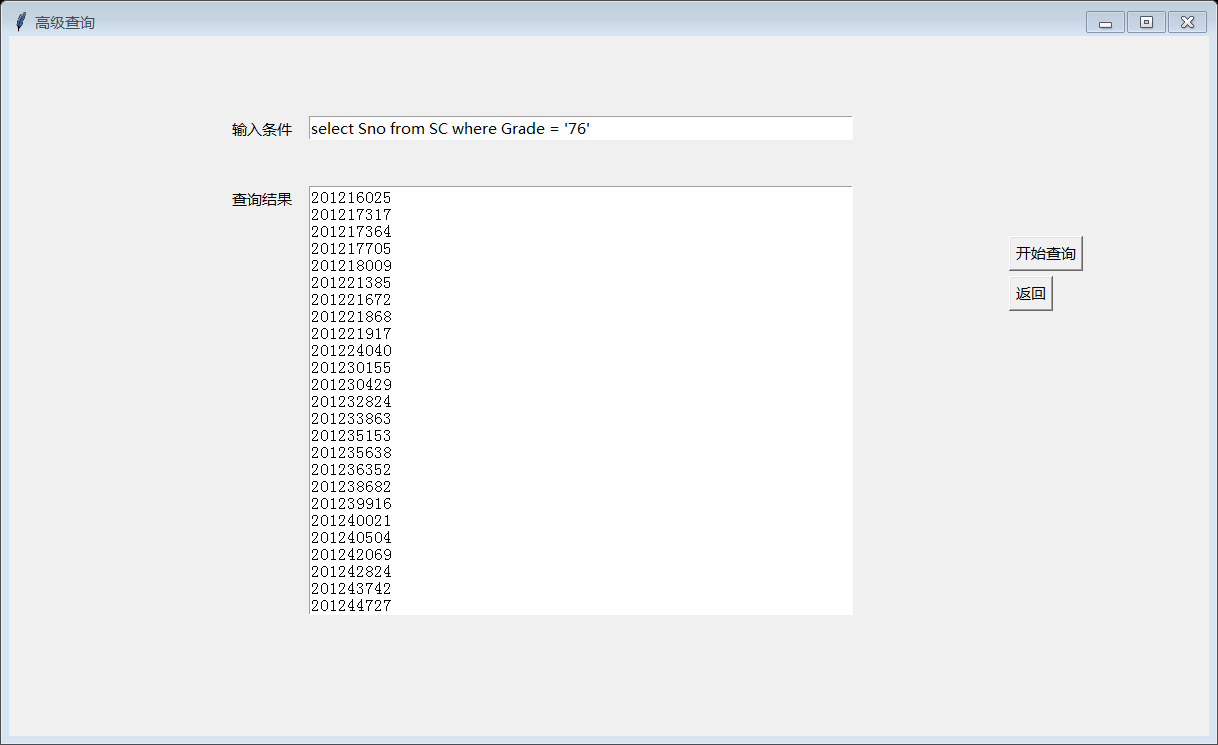
（1）与（3）等价，（2）是（1）的子查询

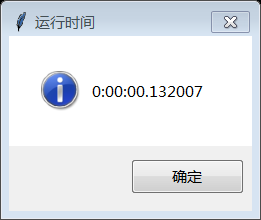
(1) select Sno,Sname from S where Sno in( select Sno from SC where Grade = '76' )



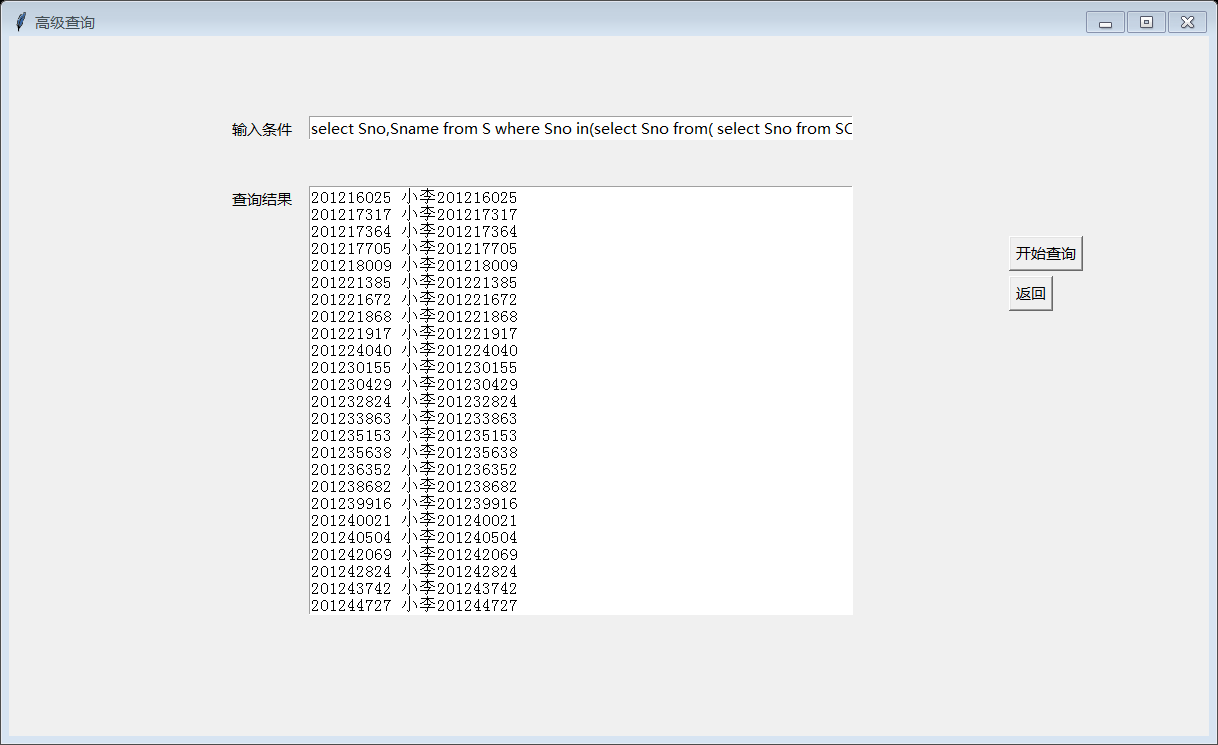


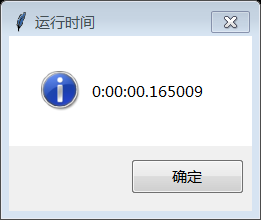
(2) select Sno from SC where Grade = '76'





(3) select Sno,Sname from S where Sno in(select Sno from( select Sno from SC where Grade = '76' ) as tb)





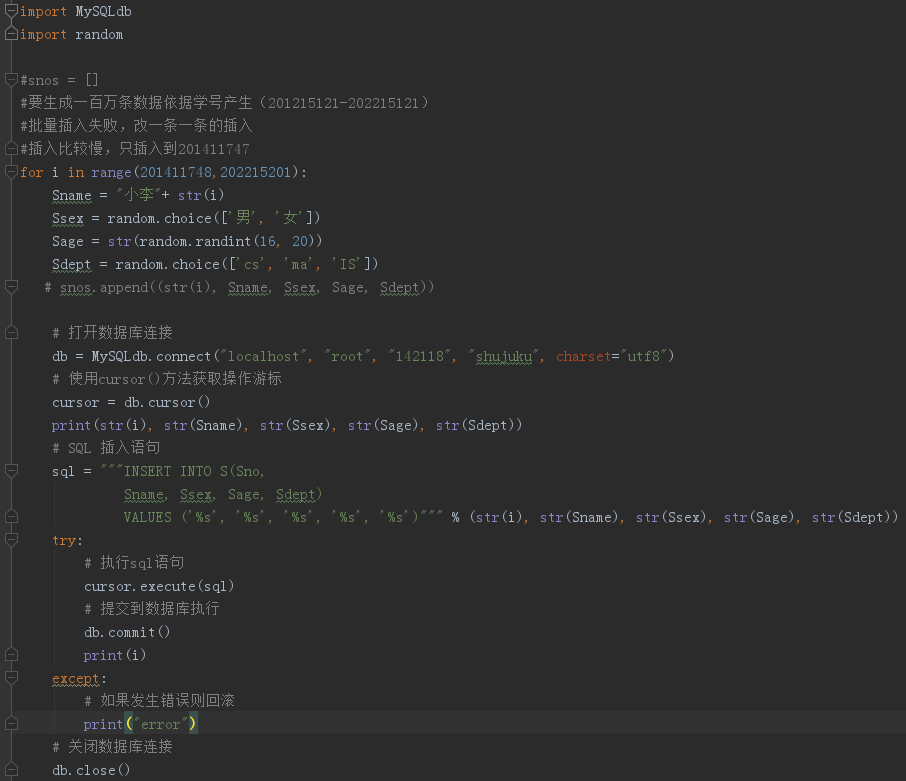
### 5.3.3结论

通过上述的比较，我们发现子查询（2）的速度非常快，而（1）的查询非常慢。因此我们通过代数优化的方法改变（1）的结构形成(3)。从而（3）比之(1)的速度近乎提高了8倍。可以看出代数优化的重要性。

## 5.4生成海量数据代码及时间的计算代码

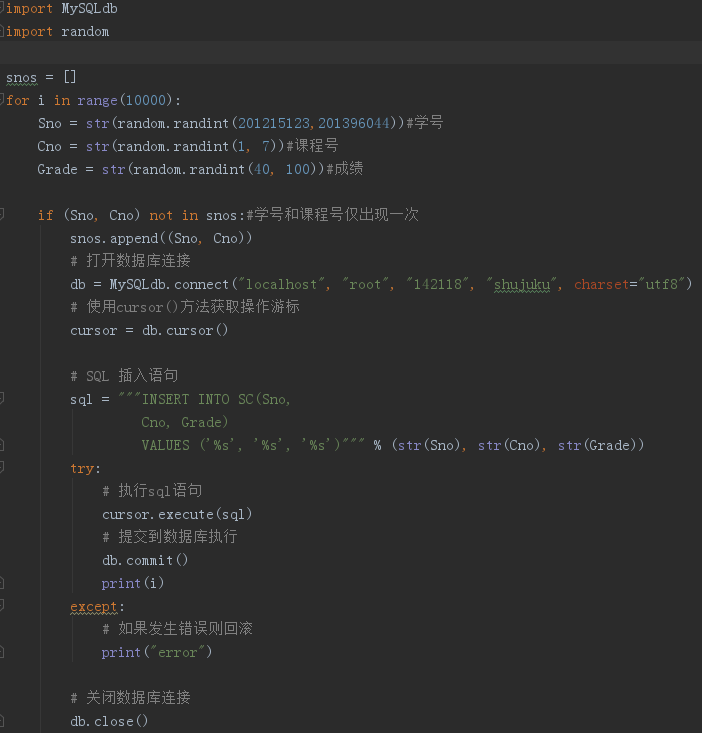
### 5.4.1 生成S表数据：（random\_date\_S.py）

部分print（）输出是在控制台显示插入数据的过程

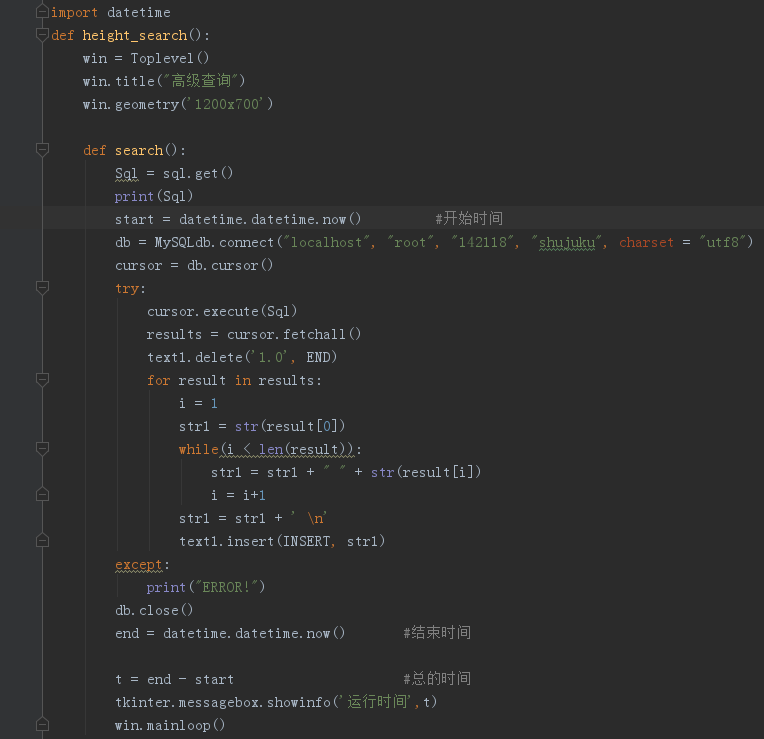


### 5.4.2 生成SC表数据：（random\_date\_SC.py）

部分print（）输出是在控制台显示插入数据的过程



### 5.4.3时间的计算：（height\_search.py）



# 心得体会

通过这次的实验使自己收获了很多，不仅让自己对python使用更上一层楼，而且对数据库的使用更加的了解了。让自己对数据库的认知并不是停留在课本上，而是在实践中。对课本中还在模糊的增删改查语句模糊的地方也清晰了很多。也明白了查询优化对数据库操作的重要性。除此之外，因为计算机网络课的实验是用python和tkinter写的，也是关于图形界面的。所以我选择了python来完成此次的实验。当然，自己对python和tkinter的使用，其实也是这个学期才接触的。算不上很熟悉，我喜欢他的原因是布局比较清晰，函数容易重构。虽然不像其他的图形界面可以直接拖。这个则需要自己布局，一行一行代码的去写。我的大部分时间就是花在这个方面。除此之外，还有涉及中文也花费了自己很多的时间去验证。最后，自己成功做出了学生选课管理系统。

当然，自己做的这个系统还有很多需要改进的地方，当时验收的时候，学长学姐也提出来了。这主要是自己的原因，考虑不够清晰。比如自己设计的高级查询，缺点就是并非所有的人都知道sql语句怎么写。我自己的想法是高级查询就像excel里的通过函数来筛选数据。自己做这个也是为了对查询in优化的一种准备，也是为能够为用户筛选出更多的数据。其他界面做的就比较简陋，就是简单的增删改查。自己对tkinter也是不怎么熟悉，导致很多有想法的地方没法实现，这是一个很遗憾的事。

# 参 考 文 献

1. 王珊，萨师煊等. 数据库系统概论. 第 5版，北京：高等教育出版社，2014.9
2. 刘春茂，裴雨龙等.Python程序设计案例教程课堂，北京：清华大学出版社，2017.