

兰州城市学院

# 电子与信息工程学院

## 课程设计报告

(2020-2021 学年第二学期)

课 程： 大数据应用语言

题 目： 学生信息管理系统

专业班级： 大数据 191 班

学 号： 20190804050145

姓 名： 周文华

指导教师： 张明

完成周数： 第 16 周—第 19 周

2021 年 07 月 11 日

# 目录

一. 知识点回顾 .....	4
1. python 基础 .....	4
2. tkinter 库 .....	4
2. MySQL 数据库 .....	4
一. 前期准备 .....	5
1. 工具使用 .....	5
2. 数据库创建 (students 库) .....	5
① 账户信息表 (verify) .....	5
② 学生信息表 (student) .....	5
③ 学生成绩表 (score) .....	5
二. 设计基本思路 .....	6
1. 登录 .....	6
2. 注册 .....	6
3. 学生端 .....	6
4. 教师端 .....	6
三. 设计流程 .....	7
1. 登录和注册页面 .....	8
2. 学生端页面 .....	8
3. 教师端页面 .....	8
4. 学生基本信息修改页面 .....	9
5. 学生成绩录入界面 .....	9

6. 学生信息表及学生成绩表.....	10
7. 密码修改页面.....	11
六. 总结.....	11
1. 心得体会 .....	11
2. Python 对大数据专业的优势 .....	11
七. 核心代码.....	12
1. 学生信息修改.....	12
2. 学生成绩录入.....	14
3. 学生信息删除.....	17
4. 注册 .....	17

# 一. 知识点回顾

## 1. python 基础

## 2. tkinter 库

Python 自带了 tkinter 模块，实质上是一种流行的面向对象的 GUI 工具包 TK 的 Python 编程接口，提供了快速便利地创建 GUI 应用程序的方法。其图像化编程的基本步骤通常包括：导入 tkinter 模块，创建 GUI 根窗体，添加人机交互控件并编写相应的函数。在主事件循环中等待用户触发事件响应。

Tkinter 的优势：相比 Python 的其他 GUI 库更加简单，跨平台，Python 的标准库，不需要安装。

## 2. MySQL 数据库

对数据统一管理，方便数据的管理与程序的调用，MySQL 数据库作为一个开源的数据库，具有软件体积小，安装使用简单，并且易于维护，安装及维护成本低的特点。

### 数据库的优点：

1. 持久化数据到本地
2. 可以实现结构化查询，方便

### 数据库相关概念：

- 1、DB：数据库，保存一组有组织的数据的容器
- 2、DBMS：数据库管理系统，又称为数据库软件（产品），用于管理 DB 中的数据
- 3、SQL：结构化查询语言，用于和 DBMS 通信的

### 数据库存储数据的特点：

- 1、将数据放到表中，表再放到库中
- 2、一个数据库中可以有多个表，每个表都有一个的名字，用来标识自己。

表名具有唯一性。

3、表具有一些特性，这些特性定义了数据在表中如何存储，类似 java 中“类”的设计。

4、表由列组成，我们也称为字段。所有表都是由一个或多个列组成的，每一列类似 java 中的“属性”

5、表中的数据是按行存储的，每一行类似于 java 中的“对象”

## 一. 前期准备

### 1. 工具使用

Pycharm, vscode

数据库: MySQL, SQLyog

### 2. 数据库创建 (students 库)

#### ① 账户信息表 (verify)

包含账号，密码，类型三个列，账号为主键。

列名	数据类型	长度	默认	主键?	非空?
name	char	10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
password	char	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>
mold	char	8		<input type="checkbox"/>	<input checked="" type="checkbox"/>

#### ② 学生信息表 (student)

包含学号，姓名，性别，年龄，班级，电话，学院七个列，学号为主键。

列名	数据类型	长度	默认	主键?	非空?
id	char	10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
name	char	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>
sex	char	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>
age	char	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>
grade	char	20		<input type="checkbox"/>	<input checked="" type="checkbox"/>
phone	char	20		<input type="checkbox"/>	<input checked="" type="checkbox"/>
college	char	30		<input type="checkbox"/>	<input checked="" type="checkbox"/>

#### ③ 学生成绩表 (score)

包含学号，姓名，班级，高数，物理，体育，思修七个列，学号为主键。

列名	数据类型	长度	默认	主键?	非空?
id	char	5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
name	char	5		<input type="checkbox"/>	<input checked="" type="checkbox"/>
grade	char	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>
gaoshu	int	3		<input type="checkbox"/>	<input checked="" type="checkbox"/>
wuli	int	3		<input type="checkbox"/>	<input checked="" type="checkbox"/>
tiyu	int	3		<input type="checkbox"/>	<input checked="" type="checkbox"/>
sixiu	int	3		<input type="checkbox"/>	<input checked="" type="checkbox"/>

## 二. 设计基本思路

### 1. 登录

登录功能分学生登录和教师登录，用户可通过登录界面的文本选择框选择，选择完成并输入账号和密码后，点击登录按钮，程序会从数据库中的账户信息表（verify）进行用户名查找并进行密码和身份信息比对，信息正确后会跳转到账户身份所对应的页面（教师端和学生端），密码错误或查找不到账号则会弹出提示账号或密码错误。

### 2. 注册

注册页面主要功能就是账号的注册，通过登录页面的注册按钮进入后，有身份选择，账号输入，密码输入和确认密码。输入并验证完成后将输入的账号，密码及对应身份信息添加到账户信息表（verify）中。学生注册时会将学生信息表（student）和学生成绩表（score）同步添加。没有数据的用“0”补齐。

### 3. 学生端

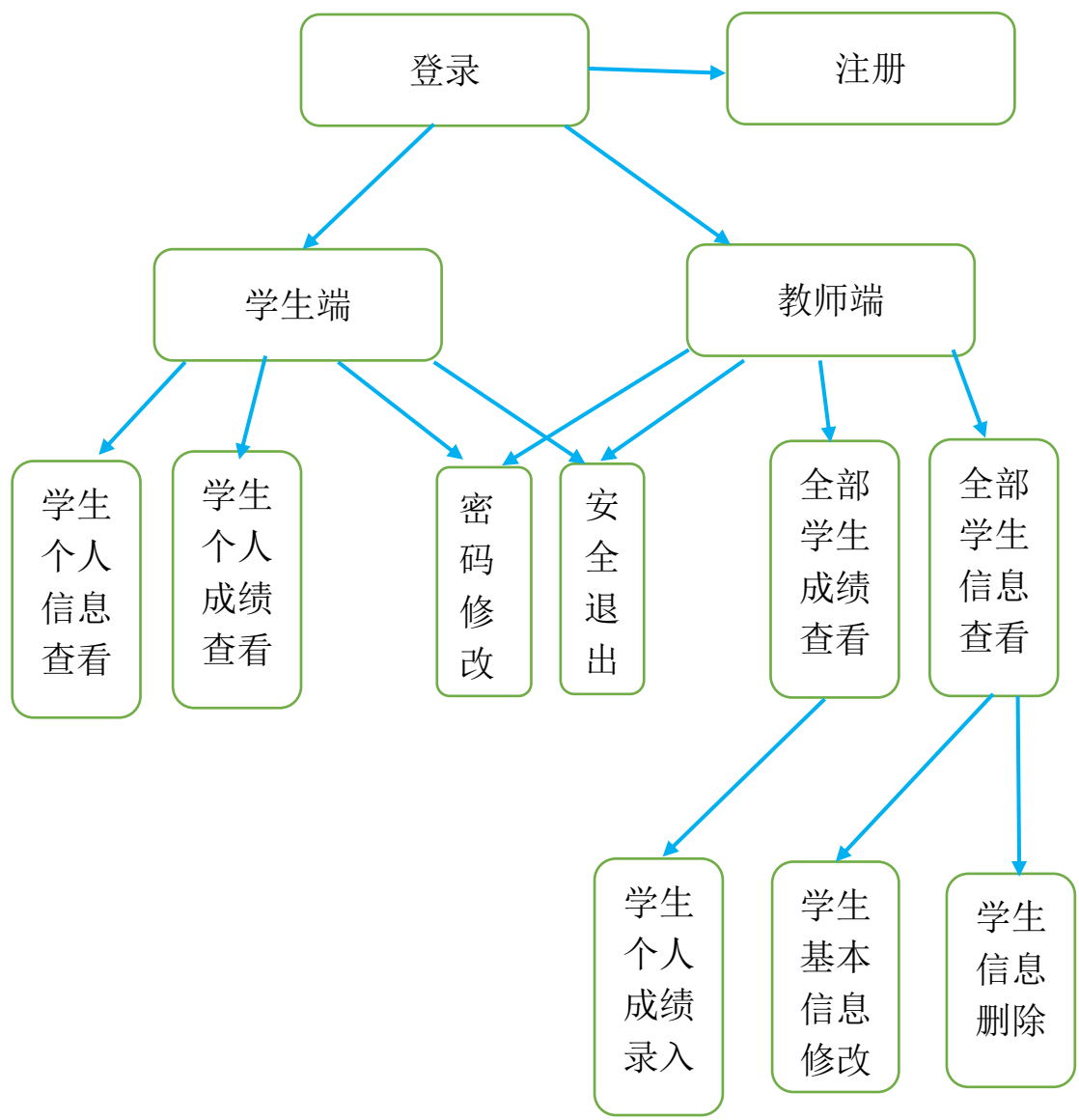
学生端包含的功能有学生基本信息修改，密码修改和安全退出。学生端页面上显示的内容有学生个人的基本信息和学生个人成绩。学生端的权限仅限于修改和查看登录学生自己的个人信息以及查看个人的成绩。

### 4. 教师端

教师端实现了对全部学生的基本信息和成绩的查看，以及通过班级、学院、

姓名、学号查询学生进本信息和成绩并对查询到的信息以表格的形式显示出来。通过学号对学生个人信息修改和删除，对学生成绩录入。在删除学生信息的同时，该生的账号、成绩也将一并删除。

### 三. 设计流程



### 五. 界面展示

## 1. 登录和注册页面

学生管理系编登录

兰州城市学院学生管理信息系统



学生

学生

教师

账号

密码

登录

注册

注意：账号为学号或教师编号，由1~5位阿拉伯数字组成

账号注册

教师

账号

密码

确认密码

注册

返回

注意：账号为学号或教师编号，由1~5位阿拉伯数字组成

## 2. 学生端页面

学生信息表

学生信息端

学生信息

学号	姓名	性别	年龄	班级	电话	所在学院
1111	小王	男	22	大数据191	18364646464	电子与信息工程学院

公共课成绩信息

高数	大学物理	大学体育	思修
10	100	59	33

学生个人信息修改

密码修改

安全退出

## 3. 教师端页面



教师管理端

全部学生信息

全部学生成绩

班级学生信息查询

班级学生成绩查询

电子与信息学院学生信息查询

学生个人成绩查询

学生个人信息查询

学生个人成绩录入

学生个人信息修改

学生信息删除

修改密码

安全退出

#### 4. 学生基本信息修改页面

学生个人信息修改

姓名: 小王

学号: 1111

性别: 男

提交信息

年龄: 22

退出

班级: 大数据191

电话: 1836464646

学院: 电子与信息

#### 5. 学生成绩录入界面

学号: 1111

姓名: 小王

班级: 大数据191

高数: 10

物理: 100

体育: 59

思修: 33

提交信息

退出

## 6. 学生信息表及学生成绩表

学号	姓名	性别	年龄	班级	电话	所在学院
0002	小明	男	20	大数据191	1833838438	电子与信息工程学
0005	张三	男	22	通信181	14556569696	电子与信息工程学
0006	小花	女	18	数学191	12345678996	数学学院
0022	李华	男	21	数学201	2147483647	数学学院
1111	小王	男	22	大数据191	18364646464	电子与信息工程学
44274	小娟	女	23	通信181	15213907969	电子与信息工程学
45454	小花	女	18	大数据201	13245669878	电子与信息工程学
4554	王麻子	男	23	通信201	18338384538	电子与信息工程学

学号	姓名	班级	高数	物理	体育	思修
0002	小明	大数据191	89	95	100	99
0005	张三	通信181	100	89	99	98
0006	小花	数学191	56	89	54	93
0022	李华	数学201	90	91	92	90
1111	小王	大数据191	10	100	59	33
44274	小娟	通信181	100	95	95	98
45454	小花	大数据201	58	69	56	84
4554	王麻子	通信201	56	89	78	56

## 7. 密码修改页面



密码修改

旧密码:

新密码:

确认密码:

确定 取消

## 六. 总结

### 1. 心得体会

在课程设计的制作中，我认识了自己很多不足，以我们的经验，当前的软件的功能日趋复杂，不学到一定的深度和广度是难以在实际工作中应付自如的。因此反映出学习的还不够，缺点疏漏。虚在加以刻苦钻研及学习，不断开拓视野，增强自己的实践操作技能，为以后能的更出色而努力。

在我们这次实践中，我收获了很多，从中体会到了自己再平常学习中的不足，为今后的学习找到了一个更好的方向，对我们今后的学习有很大的帮助。

### 2. Python 对大数据专业的优势

Python 是一种解释语言，因此程序不需要编译解释器来解析程序代码并生成输出。

Python 是动态类型的，因此变量类型是自动定义的。

Python 是强类型的，因此开发人员需要手动转换类型。更少的代码和更多

的使用使它更容易接受。

Python 是可移植的、可扩展的和可伸缩的。

Python 由于其简单的用法和广泛的数据处理库集，是大数据处理的一个非常好的选择。对于可伸缩的应用程序，它也是首选的。python 的另一个重要方面是它能够轻松地与 web 应用程序集成，所有这些特性都为大数据处理和快速洞察提供了支持。这种快速和动态的洞察力(经常变化)对企业很有价值。因此，他们需要一些强大的语言/平台/工具来立即获得这种价值，并在市场上保持竞争力。Python 在这里发挥着重要作用，并支持业务需求。

## 七. 核心代码

### 1. 学生信息修改

```
def xinxixiugei(s):
    def close():
        rt6.destroy()
    def tijiao():
        s1 = e1.get()
        s2 = c2.get()
        s3 = c3.get()
        s4 = e4.get()
        s5 = e5.get()
        s6 = c6.get()
        if s1 == " or s2 == " or s3 == " or s4 == " or s5 == " or s6 == ":
            y2 = mb.showinfo('信息修改', '信息修改成功')
            print(y2)
        else:
            if s5.isdigit():
                if 9999999999 < int(s5) < 10000000000000:
                    cur.execute('update students set name=%s, sex=%s, age=%s, grade=%s,
phone=%s, college=%s where id=%s', (s1,s2,s3,s4,s5,s6,s))
                    cur.execute('update score set name=%s, grade=%s where id=%s',(s1, s4,
s))
                    cn.commit()
                    y1 = mb.showinfo('信息修改', '信息修改成功')
                    print(y1)
                    rt6.destroy()
            else:
```

```

        y3 = mb.showinfo('信息修改', '电话号码位数错误')
        print(y3)
    else:
        y3 = mb.showinfo('信息修改', '电话号码格式错误')
        print(y3)
rt6 = tk.Tk()
rt6.title("学生个人信息修改")
rt6.geometry('500x400')
rt6.update()
curx = rt6.winfo_width()
cury = rt6.winfo_height()
scnx = rt6.winfo_screenwidth()
scny = rt6.winfo_screenheight()
tm6 = '%dx%d+%d+%d' % (curx, cury, (scnx - curx) / 2, (scny - cury) / 2)
rt6.geometry(tm6)
rt6.resizable(False, False)
cur.execute('select * from students where id=%s', s)
tm1 = cur.fetchone()

b1 = tk.Label(rt6, text="姓名: ", font=("宋体", 20))
b1.place(x=40, y=40)
var1 = tk.StringVar()
e1 = tk.Entry(rt6, textvariable=var1, width=10, font=("宋体", 20))
e1.place(x=120, y=40)
e1.insert(0, tm1[1])

b2 = tk.Label(rt6, text="性别: ", font=("宋体", 20))
b2.place(x=40, y=100)
var2 = tk.StringVar()
c2 = ttk.Combobox(rt6, textvariable=var2, width=9, font=("宋体", 20))
c2['values'] = ("男", "女")
c2["state"] = "readonly"
c2.current(0)
c2.set(tm1[2])
c2.place(x=120, y=100)

b3 = tk.Label(rt6, text="年龄: ", font=("宋体", 20))
b3.place(x=40, y=160)
var3 = tk.StringVar()
c3 = ttk.Combobox(rt6, textvariable=var3, width=9, font=("宋体", 20))
c3['values'] = ("17", "18", "19", "20", '21', '22', '23', '24', '25')
c3["state"] = "readonly"
c3.current(2)
c3.set(tm1[3])

```

```

c3.place(x=120, y=160)

b4 = tk.Label(rt6, text="班级: ", font=("宋体", 20))
b4.place(x=40, y=220)
var4 = tk.StringVar()
e4 = tk.Entry(rt6, textvariable=var4, width=10, font=("宋体", 20))
e4.insert(0, tm1[4])
e4.place(x=120, y=220)

b5 = tk.Label(rt6, text="电话: ", font=("宋体", 20))
b5.place(x=40, y=280)
var5 = tk.StringVar()
e5 = tk.Entry(rt6, textvariable=var5, width=10, font=("宋体", 20))
e5.insert(0, tm1[5])
e5.place(x=120, y=280)

b6 = tk.Label(rt6, text="学院: ", font=("宋体", 20))
b6.place(x=40, y=340)
var6 = tk.StringVar()
c6 = ttk.Combobox(rt6, textvariable=var6, width=9, font=("宋体", 20))
c6['values'] = ("电子与信息工程学院", "机械工程学院", '数学学院')
c6["state"] = "readonly"
c6.current(0)
c6.set(tm1[6])
c6.place(x=120, y=340)

b7 = tk.Label(rt6, text='学号: '+tm1[0], font=('华文行楷', 20))
b7.place(x=300, y=40)
bu7 = tk.Button(rt6, text="提交信息", width=16, height=1, font=("宋体", 15),
command=lambda: tijiao())
bu7.place(x=300, y=100)
bu8 = tk.Button(rt6, text="退出", width=10, height=1, font=("宋体", 15),
command=lambda: close())
bu8.place(x=300, y=200)
rt6.mainloop()

```

## 2. 学生成绩录入

```

def chenji():
    s = e35.get()

    def chenji2():
        def close():
            rt6.destroy()

```

```

def tijiao():
    s1 = e1.get()
    s2 = e2.get()
    s3 = e3.get()
    s4 = e4.get()
    if s1 == " or s2 == " or s3 == " or s4 == ":
        y2 = mb.showinfo('学生成绩录入', '请输入成绩信息')
        print(y2)
    else:
        if s1.isdigit() and s2.isdigit() and s3.isdigit() and s4.isdigit():
            if 0 <= int(s1) <= 100 and 0 <= int(s2) <= 100 and 0 <= int(s3) <= 100
and 0 <= int(s4) <= 100:
                cur.execute('update                                score                                set
gaoshu=%s,wuli=%s,tiyu=%s,sixiu=%s where id=%s',(s1, s2, s3, s4, s))
                cn.commit()
                y1 = mb.showinfo('学生成绩录入', '成绩录入成功')
                print(y1)
                rt6.destroy()
            else:
                y3 = mb.showinfo('学生成绩录入', '成绩不在正确区间')
                print(y3)

        else:
            y3 = mb.showinfo('学生成绩录入', '成绩格式错误')
            print(y3)

rt6 = tk.Tk()
rt6.title("学生成绩录入")
rt6.geometry('500x400')
rt6.update()
curx = rt6.winfo_width()
cury = rt6.winfo_height()
scnx = rt6.winfo_screenwidth()
scny = rt6.winfo_screenheight()
tm6 = '%dx%d+%d+%d' % (curx, cury, (scnx - curx) / 2+500, (scny - cury) / 2)
rt6.geometry(tm6)
rt6.resizable(False, False)
cur.execute('select * from score where id=%s', s)
tm1 = cur.fetchone()

b1 = tk.Label(rt6, text="高数: ", font=("宋体", 20))
b1.place(x=40, y=40)
var1 = tk.StringVar()
e1 = tk.Entry(rt6, textvariable=var1, width=10, font=("宋体", 20))

```

```

e1.place(x=120, y=40)
e1.insert(0, tm1[3])

b2 = tk.Label(rt6, text="物理: ", font=("宋体", 20))
b2.place(x=40, y=100)
var2 = tk.StringVar()
e2 = tk.Entry(rt6, textvariable=var2, width=10, font=("宋体", 20))
e2.insert(0, tm1[4])
e2.place(x=120, y=100)

b3 = tk.Label(rt6, text="体育: ", font=("宋体", 20))
b3.place(x=40, y=160)
var3 = tk.StringVar()
e3 = tk.Entry(rt6, textvariable=var3, width=10, font=("宋体", 20))
e3.insert(0, tm1[5])
e3.place(x=120, y=160)

b4 = tk.Label(rt6, text="思修: ", font=("宋体", 20))
b4.place(x=40, y=220)
var4 = tk.StringVar()
e4 = tk.Entry(rt6, textvariable=var4, width=10, font=("宋体", 20))
e4.insert(0, tm1[6])
e4.place(x=120, y=220)

b7 = tk.Label(rt6, text='学号: ' + tm1[0], font=('华文行楷', 20))
b7.place(x=300, y=40)
b8 = tk.Label(rt6, text='姓名: ' + tm1[1], font=('华文行楷', 20))
b8.place(x=300, y=100)
b9 = tk.Label(rt6, text='班级: ' + tm1[2], font=('华文行楷', 20))
b9.place(x=300, y=160)
bu7 = tk.Button(rt6, text="提交信息", width=16, height=1, font=("宋体", 15),
command=lambda: tijiao())
bu7.place(x=300, y=220)
bu8 = tk.Button(rt6, text="退出", width=16, height=1, font=("宋体", 15),
command=lambda: close())
bu8.place(x=300, y=260)
rt6.mainloop()
if s.isdigit():
    if 0 <= int(s) < 1000000:
        cur.execute('select * from score where id=%s', s)
        lg = cur.fetchone()
        if lg:
            chenji2()
        else:

```



```

        y6 = mb.showinfo('提示', '该学生不存在')
        print(y6)
    else:
        y4 = mb.showinfo('提示', '学号位数错误')
        print(y4)
    else:
        y5 = mb.showinfo('提示', '学号格式错误')
        print(y5)

```

### 3. 学生信息删除

```

def delete():
    s = e36.get()
    if s == "":
        y7 = mb.showinfo('提示', '请输入学号')
        print(y7)
    else:
        if s.isdigit():
            if 0 < int(s) < 100000:
                cur.execute('select * from score where id=%s', s)
                lg = cur.fetchone()
                if lg:
                    cur.execute('delete from students where id=%s', s)
                    cur.execute('delete from verify where name=%s', s)
                    cur.execute('delete from score where id=%s', s)
                    cn.commit()
                    y9 = mb.showinfo('提示', '学生信息删除成功')
                    print(y9)
                else:
                    y6 = mb.showinfo('提示', '该学生不存在')
                    print(y6)
            else:
                y4 = mb.showinfo('提示', '学号位数错误')
                print(y4)
        else:
            y5 = mb.showinfo('提示', '学号格式错误')
            print(y5)

```

### 4. 注册

```

def login():
    def close():
        rt1.destroy()

```

```

def into():
    s7 = var7.get()
    s6 = var6.get()
    s4 = c2.get()
    s5 = var5.get()
    if s4 == " or s5 == " or s6 == " or s7 == ":
        y4 = mb.showinfo('注册', '请输入账户与密码')
        print(y4)
    else:
        if s6 == s7:
            if s5.isdigit():
                if 0 < int(s5) < 100000:
                    cur.execute('select * from verify where name=%s', s5)
                    lg = cur.fetchone()
                    if lg:
                        y6 = mb.showinfo('注册', '用户名已存在')
                        print(y6)
                    else:
                        if s4 == '教师':
                            cur.execute('insert into verify values(%s,%s,%s)', (s5, s6, s4))
                            cn.commit()
                            y8 = mb.showinfo('注册', '注册成功')
                            print(y8)
                            rt1.destroy()
                        elif s4 == '学生':
                            cur.execute('insert into verify values(%s,%s,%s)', (s5, s6, s4))
                            cur.execute('insert into students values(%s,0,0,0,0,0,0)', s5)
                            cur.execute('insert into score values(%s,0,0,0,0,0,0)', s5)
                            cn.commit()
                            print(s4 + s5 + s6)
                            y8 = mb.showinfo('注册', '注册成功')
                            print(y8)
                            rt1.destroy()
                        else:
                            y7 = mb.showinfo('注册', '账号格式错误')
                            print(y7)
            else:
                y8 = mb.showinfo('注册', '账号格式错误')
                print(y8)
        else:
            y5 = mb.showinfo('注册', '两次输入密码不同')
            print(y5)

```

```

rt1 = tk.Toplevel()
rt1.title("账号注册")
rt1.geometry('400x400')
rt1.update()
curx1 = rt1.winfo_width()
cury1 = rt1.winfo_height()
scnx1 = rt1.winfo_screenwidth()
scny1 = rt1.winfo_screenheight()
tm1 = '%dx%d+%d+%d' % (curx1, cury1, (scnx1 - curx1) / 2 + 400, (scny1 - cury1)
/ 2)
rt1.geometry(tm1)
rt1.resizable(False, False)
var4 = tk.StringVar()
c2 = ttk.Combobox(rt1, textvariable=var4, font=("宋体", 20), state='readonly')
c2['values'] = ("学生", "教师")
c2["state"] = "readonly"
c2.set("学生")
c2.current(0)
c2.place(x=40, y=40)

b14 = tk.Label(rt1, text="账号", font=("宋体", 20))
b14.place(x=40, y=100)
var5 = tk.StringVar()
e13 = tk.Entry(rt1, textvariable=var5, width=15, font=("宋体", 20))
e13.place(x=120, y=100)

b15 = tk.Label(rt1, text="密码", font=("宋体", 20))
b15.place(x=40, y=160)
var6 = tk.StringVar()
e14 = tk.Entry(rt1, textvariable=var6, width=15, font=("宋体", 20), show='*')
e14.place(x=120, y=160)
e14.focus_set()

b6 = tk.Label(rt1, text="确认密码", font=("宋体", 20))
b6.place(x=20, y=220)
var7 = tk.StringVar()
e6 = tk.Entry(rt1, textvariable=var7, width=15, font=("宋体", 20), show='*')
e6.place(x=130, y=220)
e6.focus_set()

bu3 = tk.Button(rt1, text="注册", width=7, height=1, font=("华文行楷", 20),
command=lambda: into())
bu3.place(x=60, y=300)
bu4 = tk.Button(rt1, text="返回", width=7, height=1, font=("华文行楷", 20),

```

```
command=lambda: close())
    bu4.place(x=200, y=300)
    b5 = tk.Label(rt1, text="注意：账号为学号或教师编号，由 1~5 位阿拉伯数字
组成", font=("宋体", 10), width=45, height=2)
    b5.place(x=40, y=360)
    rt1.mainloop()
```