class Node:  
 def \_\_init\_\_(self, student\_id, name,sex,birthday,Chinese\_score,math\_score,English\_score):  
 self.student\_id = student\_id  
 self.name = name  
 self.sex = sex  
 self.birthday = birthday  
 self.Chinese\_score = Chinese\_score  
 self.Math\_score = math\_score  
 self.English\_score = English\_score  
 self.score = int(math\_score)+int(Chinese\_score)+int(English\_score)  
 self.next = None  
  
class StudentList:  
 def \_\_init\_\_(self):  
 self.head = None  
  
 def login(self, user\_id, password):  
 if user\_id == "teacher" and password == "123456":  
 print("登录成功！")  
 return True  
 else:  
 print("用户名或密码错误！")  
 return False  
 def save(self):  
 p = self.head  
 with open('student.txt','w',encoding='utf-8') as fins:  
 while p is not None:  
 fins.write(str(p.student\_id)+','+p.name+','+p.sex+','+p.birthday+','+str(p.Chinese\_score)+','+str(p.English\_score)+','+str(p.Math\_score)+'\n')  
 p = p.next  
 def download(self):  
 with open('student.txt','r',encoding='utf-8') as fins:  
 for line in fins:  
 student\_id,name,sex,birthday,Chinese\_score,English\_score,math\_score = line.strip().split(',')  
 self.add\_student(student\_id,name,sex,birthday,Chinese\_score,English\_score,math\_score)  
 def add\_student(self, student\_id, name,sex,birthday,Chinese\_score,math\_score,English\_score):  
 node = Node(student\_id, name,sex,birthday,Chinese\_score,math\_score,English\_score)  
 if self.head is None:  
 self.head = node  
 else:  
 p = self.head  
 while p.next is not None:  
 p = p.next  
 p.next = node  
  
 def delete\_student(self, student\_id):  
 if self.head is None:  
 print("学生信息为空")  
 return  
 if self.head.student\_id == student\_id:  
 self.head = self.head.next  
 print(f"已删除学号为 {student\_id} 的学生信息")  
 return  
 p = self.head  
 while p.next is not None:  
 if p.next.student\_id == student\_id:  
 p.next = p.next.next  
 print(f"已删除学号为 {student\_id} 的学生信息")  
 return  
 p = p.next  
 print(f"未找到学号为 {student\_id} 的学生信息")  
  
 def update\_student(self, student\_id, name,sex,birthday,Chinese\_score,math\_score,English\_score):  
 if self.head is None:  
 print("学生信息为空")  
 return  
 p = self.head  
 while p is not None:  
 if p.student\_id == student\_id:  
 p.name = name  
 p.sex = sex  
 p.birthday = birthday  
 p.Chinese\_score = Chinese\_score  
 p.Math\_score = math\_score  
 p.English\_score = English\_score  
 print(f"已更新学号为 {student\_id} 的学生信息")  
 return  
 p = p.next  
 print(f"未找到学号为 {student\_id} 的学生信息")  
  
 def search\_student(self, student\_id):  
 if self.head is None:  
 print("学生信息为空")  
 return  
 p = self.head  
 while p is not None:  
 if p.student\_id == student\_id:  
 print(f"学号：{p.student\_id}，姓名：{p.name}，性别：{p.sex},生日：{p.birthday},语文成绩：{p.Chinese\_score},数学成绩：{p.Math\_score},英语成绩：{p.English\_score},总分:{p.score}")  
 return  
 p = p.next  
 print(f"未找到学号为 {student\_id} 的学生信息")  
  
 def sort\_score(self):  
 *# if self.head is None:  
 # print("学生信息为空！")  
 # return  
 # Student = []  
 # p = self.head  
 # while p is not None:  
 # Student.append(p)  
 # p = p.next  
 # Student.sort(key=lambda x: x.score, reverse=True)  
 # print("学号\t姓名\t性别\t生日\t语文成绩\t数学成绩\t英语成绩\t总分")  
 # for stu in Student:  
 # print(f"{stu.student\_id}\t{stu.name}\t{stu.sex}\t{stu.birthday}\t{stu.Chinese\_score}\t{stu.Math\_score}\t{stu.English\_score}\t{stu.score}")* if not self.head or not self.head.next:  
 return  
 p = self.head  
 cnt = 0  
 while p is not None:  
 cnt+=1  
 p = p.next  
 self.display\_all\_students()  
 for i in range(cnt-1):  
 p = self.head  
 q = self.head.next  
 while q != None:  
 if p.score > q.score:  
 p.student\_id,q.student\_id = q.student\_id,p.student\_id  
 p.name,q.name = q.name,p.name  
 p.sex,q.sex = q.sex,p.sex  
 p.birthday,q.birthday = q.birthday,p.birthday  
 p.English\_score,q.English\_score = q.English\_score,p.English\_score  
 p.Chinese\_score,q.Chinese\_score = q.Chinese\_score,p.Chinese\_score  
 p.Math\_score,q.Math\_score = q.Math\_score,p.Math\_score  
 p.score,q.score = q.score,p.score  
 p = p.next  
 q = q.next  
 self.display\_all\_students()  
 def fuzzy\_search\_student(self, keyword):  
 if self.head is None:  
 print("学生信息为空！")  
 return  
 print("学号\t姓名\t性别\t生日\t语文成绩\t数学成绩\t英语成绩\t总分")  
 p = self.head  
 while p is not None:  
 if keyword in p.name:  
 print(f"{p.student\_id}\t{p.name}\t{p.sex}\t{p.birthday}\t{p.Chinese\_score}\t{p.Math\_score}\t{p.English\_score}\t{p.score}")  
 return  
 p = p.next  
 else:  
 print("未找到该学生")  
  
 def display\_all\_students(self):  
 if self.head is None:  
 print("学生信息为空！")  
 return  
 print("学号\t姓名\t性别\t生日\t语文成绩\t数学成绩\t英语成绩\t总分")  
 p = self.head  
 while p is not None:  
 print(f"{p.student\_id}\t{p.name}\t{p.sex}\t{p.birthday}\t{p.Chinese\_score}\t{p.Math\_score}\t{p.English\_score}\t{p.score}")  
 p = p.next  
if \_\_name\_\_ == "\_\_main\_\_":  
 Stu = StudentList()  
 while True:  
 print("欢迎使用学生信息管理系统！")  
 user\_id = input("请输入用户名：")  
 password = input("请输入密码：")  
 if Stu.login(user\_id, password):  
 Stu.download()  
 break  
 while True:  
 print('--------------------------------欢迎来到学生信息管理系统----------------------------------------')  
 print("请选择操作：")  
 print("1. 添加学生信息")  
 print("2. 删除学生信息")  
 print("3. 修改学生信息")  
 print("4. 查询学生信息")  
 print("5. 显示所有学生信息")  
 print("6. 模糊查找")  
 print("7. 将学生信息进行排序")  
 print("0. 退出系统")  
 choice = input('请输入您的选项')  
 if choice == "1":  
 student\_id = input("请输入学号：")  
 name = input("请输入姓名：")  
 sex = input('请输入学生性别')  
 birthday = input('请输入学生生日')  
 Chinese\_score = input('请输入学生语文成绩')  
 math\_score = input('请输入学生数学成绩')  
 English\_score = input('请输入学生英语成绩')  
 Stu.add\_student(student\_id, name,sex,birthday,Chinese\_score,math\_score,English\_score)  
 elif choice == "2":  
 student\_id = input("请输入要删除的学生的学号：")  
 Stu.delete\_student(student\_id)  
 elif choice == "3":  
 student\_id = input("请输入要修改的学生的学号：")  
 name = input("请输入新的姓名：")  
 sex = input('请输入新的学生性别')  
 birthday = input('请输入新的学生生日')  
 Chinese\_score = input('请输入新的学生语文成绩')  
 math\_score = input('请输入新的学生数学成绩')  
 English\_score = input('请输入新的学生英语成绩')  
 Stu.update\_student(student\_id, name,sex,birthday,Chinese\_score,math\_score,English\_score)  
 elif choice == "4":  
 student\_id = input("请输入要查询的学生的学号：")  
 Stu.search\_student(student\_id)  
 elif choice == "5":  
 Stu.display\_all\_students()  
 elif choice == '6':  
 keyword = input('请输入关键字:')  
 Stu.fuzzy\_search\_student(keyword)  
 elif choice == '7':  
 Stu.sort\_score()  
 elif choice == "0":  
 print("谢谢使用")  
 Stu.save()  
 break  
 else:  
 print("没有这个选项，请重新输入")