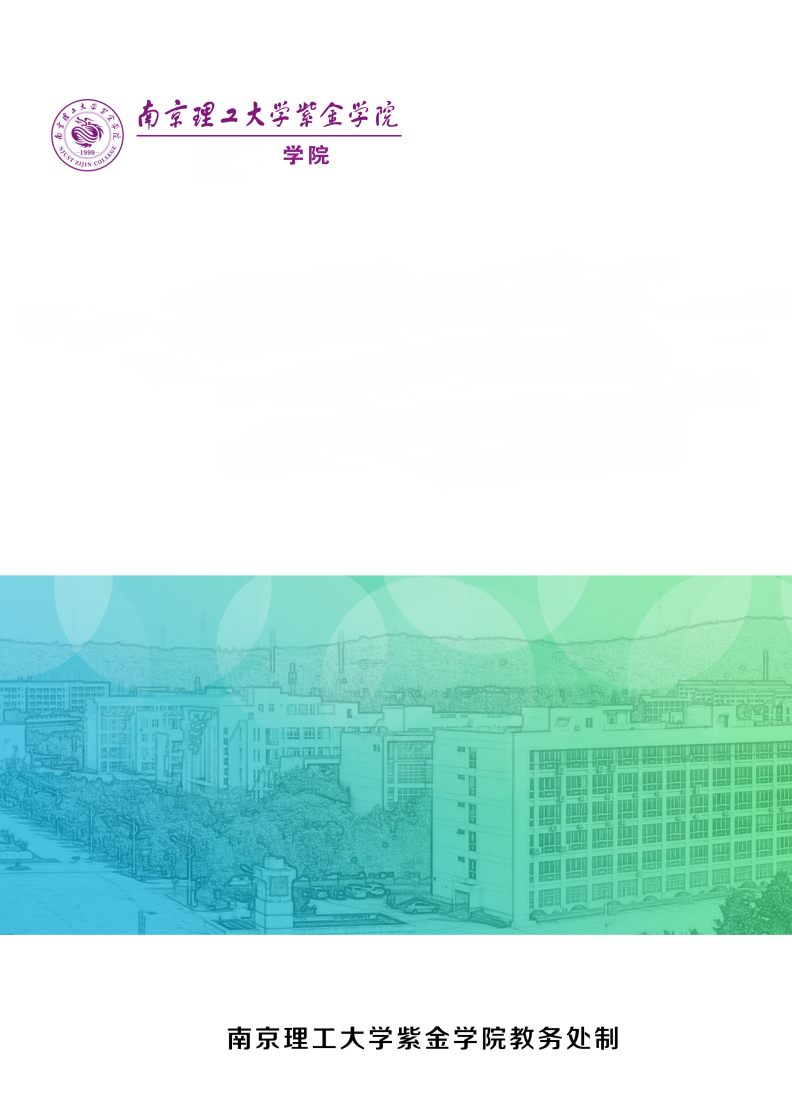
****

**主编：王颖 朱俊**

**《Java课程设计》实验指导书**

**计算机**机

**内部教材仅供紫金学院使用**

**前 言**

Java课程设计是计算机类专业的一门综合实践课程，主要目的是加深学生对面向对象编程的了解，培养学生利用计算机处理问题的思维方式和程序设计的基本方法，启发学生主动将计算机程序设计思想引入到其它基础课和专业课，提高同学们的动手实践能力。

南京理工大学紫金学院

计算机学院实验指导书



**课程编号： 03021011**

**课程名称： Java课程设计**

**课程学分： 1学分**

**实验学时： 16学时**

**适用专业： 软工、计科 、网工、人工智能**

**学生实验守则**

第一条 学生进入实验室从事实验、科研活动，必须严格遵守实验室的各项规章制度。

第二条 实验前必须做好预习，明确实验的目的、内容、方法和步骤，未经预习或无故迟到15分钟以上者，指导人员有权取消其实验资格。

第三条 保持实验室的严肃、安静，不得在实验室内大声喧哗、嬉闹，不准在实验室内进食、吸烟和乱吐乱丢杂物。

第四条 学生必须在实验指导老师规定实验开始时，方可动手实验，严禁未经许可就开始实验。学生必须以实事求是的科学态度进行实验，严格遵守操作规程，服从实验教师或实验技术人员的指导(对有特殊要求的实验，必须按要求穿戴安全防护用具后方可进行实验),如违反操作规程或不听从指导而造成仪器设备损坏等事故者，按学校有关规定进行处理。

第五条 学生应备有专用实验记录本，实验记录是原始性记录，是撰写实验报告的主要依据，内容要求真实、客观地反映实际情况，实验结果须经实验教师或实验技术人员认可。

第六条 严防事故，确保实验室的安全。发现异常情况，及时报告实验教师或实验技术人员，并采取相应的措施，减少事故造成的损失。

第七条 实验完成后，应将仪器、工具及实验场地等进行清理、归还，经实验教师或实验技术人员同意后，方可离开实验室。

第八条 独立完成实验报告，按时交给实验教师，不得抄袭或臆造。

实验报告是实验完成后的全面总结，它主要包括：实验名称、实验目的、实验原理、实验仪器设备、实验条件、实验数据、结果分析和问题讨论。实验报告一律用钢笔或圆珠笔书写，统一采用国家标准所规定的单位与符号，要求文字书写工整，不得潦草；作图规范，不得随手勾画。

**目 录**

实验一 学生信息管理系统开发与实现………………………………………………1

## 实验一 学生信息管理系统开发与实现

**一、实验目的**

综合利用Java课程所学知识，开发设计学生信息管理系统：

1. 了解如何运用Java语言开发一个小型信息管理系统。
2. 重点掌握面向对象的基础知识，面向对象编程思想。
3. 掌握类，数组，GUI编程，文件流等的使用。

**二、实验原理及说明**

本课程设计主要是巩固和检验《Java程序设计》课程所学知识点，实验原理参照每章的综合案例部分。根据教材第六章内容利用自定义绘图、常用组件、容器和布局管理器完成UI界面的设计，同时利用理论课堂所学：类，内部类，数组，流等概念实现系统数据的读写和数据文件的传输，在GUI方式下利用GUI事件处理机制，完成菜单、按钮、标签等组件的鼠标键盘事件响应。

要求掌握按钮、输入框、选择框等常用组件的应用，及其鼠标，键盘的处理。使用GUI界面实现用户登录和学生信息维护、图片的上传下载、数据库连接及数据的增、删、改、查等功能。

参照教材第六章和第十一章熟练掌握以下知识点：

1、组件、容器和布局管理器；

2、GUI编程步骤；

3、GUI事件处理机制及三要素；

4、JDBC体系结构和JDBC编程。

**本次课程设计内容主要参考教材每章最后一节的综合案例的实现！**

**三、实验内容**

本次实验要求实现一个功能简单的学生信息管理系统，该系统具有按照账户名密码登陆功能，登陆后，可以添加，删除，修改、查询（显示学生相片）学生信息，添加学生信息时，要求能添加学生的相片信息（实现相片文件的上传和下载功能）。

**四、实验安全事项**

1. 注意用电安全。
2. 爱护实验室的设备。
3. 实验室保持安静、不得追逐打闹。
4. 发现异常情况，及时报告实验教师，并采取相应的措施，减少事故造成的损失。

**五、实验提交方式**

√ 实验报告 □ 现场打分 √线上平台提交 √ 其它（程序源代码）

南京理工大学紫金学院

# Java课程设计报告



|  |  |
| --- | --- |
| **课程编号：** | 03021011 |
| **课程名称：** | Java课程设计 |
| **学 院：** | 计算机学院 |
| **专 业：** | 计算机技术与科学 |
| **学 号：** | 2015302440 |
| **姓 名：** | 李兴虎 |
| **起止时间：** | 2023.4.15 |
| **指导教师：** | 陈小云 |

2023 年 4 月

## 实验一 学生信息管理系统开发与实现

1 **问题描述**

描述要求编程解决的问题。

学生成绩信息管理系统是一种用于管理学生的学术成绩和相关信息的软件系统。它旨在提供一个高效、方便和可靠的解决方案，用于记录、存储和管理学生的学术成绩，包括课程成绩、考试成绩、作业成绩等。该系统将学生的成绩信息与学生个人信息关联起来，包括学生的姓名、学号、班级、联系方式等。学生成绩信息管理系统的主要功能包括学生成绩录入、查询、统计和分析。教师可以通过系统录入学生的成绩信息，并可以根据学号、课程、班级等条件查询学生成绩。学生和家长可以通过系统查询学生的成绩信息，了解学生的学术表现。系统还可以进行成绩的统计和分析，生成成绩报表和图表，帮助教师和学生更好地了解学生的学术进展和评估学生的学习情况。学生成绩信息管理系统的开发和实现需要包括用户管理、权限管理、成绩信息录入、查询和统计分析等功能模块。系统需要具备良好的用户界面和用户体验，方便用户进行操作和管理学生的成绩信息。系统还需要保证数据的安全性和可靠性，包括对学生成绩信息的备份、恢复和权限控制，防止未经授权的访问和篡改。通过学生成绩信息管理系统的开发和实现，可以提高学校管理学生成绩信息的效率和准确性，方便教师和学生查询和分析成绩信息，帮助学生提高学术成绩，促进学校的教学管理和学生成绩监控。

**2 基本要求**

给出程序要达到的具体的要求。

①学生成绩数据管理：系统应能够有效地管理学生的成绩信息，包括学生的基本信息（如学号、姓名、班级等）、学科成绩（如语文、数学、英语等）、考试类型（如期末考、期中考等）、考试时间、成绩录入和修改等功能。系统应支持对学生成绩信息的增加、查询、修改、删除等操作，并保证数据的安全性和完整性。

②成绩分析和统计功能：系统应具备成绩分析和统计功能，可以对学生的成绩数据进行多维度的分析和统计，如学科平均分、班级平均分、学生个人成绩分析和决策。

③成绩查询与报告打印功能：系统应提供学生和家长查询学生成绩的功能，包括通过学号、姓名、班级等方式进行成绩查询，并可以打印成绩报告单。系统应支持成绩单的自定义格式，方便学校根据需要进行个性化的报告单设计。

④用户友好的界面和操作：系统应具备简洁、直观、易于使用的界面和操作，方便用户进行成绩信息的输入、查询和管理，减少用户的学习成本和使用难度。

⑤系统稳定性和安全性：系统应保证在长时间的运行过程中稳定性和安全性，能够处理大量的成绩信息和用户请求，防止数据丢失、损坏和非法访问，确保系统的可靠性和安全性。

⑥错误处理和异常处理：系统应具备良好的错误处理和异常处理机制，能够及时捕获和处理系统错误和异常情况，避免系统崩溃和数据丢失，保证系统的稳定性和可靠性。

⑦安全性保障：系统应采取相应的安全措施，包括用户身份认证、数据加密传输、防止SQL注入、XSS攻击等安全策略，以保障学生成绩信息的安全性和防止系统遭受恶意攻击。

综上所述，学生成绩信息管理系统应具备学生成绩数据管理、成绩分析和统计、成绩查询、用户友好的界面和操作、系统稳定性和安全性、数据备份和恢复功能、系统性能优化、可扩展性和兼容性、错误处理和异常处理、安全性保障、日志记录和审计功能、用户管理、界面定制和多语言支持、良好的系统文档和技术支持等基本要求，以实现一个功能完善、稳定可靠、安全高效的学生成绩信息管理系统。

**3 需求分析**

　　以无歧义的陈述说明程序设计的任务，强调的是程序要做什么？并明确规定：

　　输入的形式和输入值的范围；

　　输出的形式；

　　程序所能达到的功能；

①输入的形式和输入值的范围：

·输入形式：系统应接受用户输入的学生信息和成绩信息，可以通过文本框、表单或文件上传等方式实现。

·输入值的范围：学生信息包括学生的姓名、学号、班级等，输入值的范围应符合实际情况，例如学号应为合法的学号格式；成绩信息包括科目和成绩，输入值的范围应符合科目和成绩的规定，例如科目应为合法的科目名称，成绩应在合理的范围内，例如0到100之间。

②输出的形式：

·输出形式：系统应以易于阅读和理解的方式输出学生成绩信息和统计结果，可以通过列表、表格、图表等形式展示。

·输出内容：输出内容应包括学生的基本信息、科目和成绩信息，以及根据输入的成绩信息生成的统计和分析结果，例如平均成绩、最高成绩、最低成绩等。

③程序所能达到的功能：

学生信息的录入和管理：包括学生的基本信息录入、信息的修改、删除和查询等。

·成绩信息的录入和管理：包括学生的科目和成绩录入、成绩的修改、删除和查询等。

成绩统计和分析：包括对学生成绩的统计和分析，例如计算平均成绩、最高成绩、最低成绩等。

·成绩报表生成：根据输入的成绩信息，生成成绩报表，以便用户查看和导出。

·数据的安全和权限管理：保障学生信息和成绩信息的安全，限制不同用户的权限和操作范围，确保系统的安全性和可靠性。

·测试数据：包括正确的输入及其输出结果和含有错误的输入及其输出结果。

总之，该学生成绩信息管理系统的任务是通过清晰明了的输入和输出形式，实现学生信息和成绩信息的录入、管理、统计、分析和报表生成等功能，并确保数据的安全。

**4 概要设计**

　　说明本程序中主程序的流程以及各程序模块之间的层次(调用)关系。

学生成绩信息管理系统的主程序流程通常包含以下几个步骤：

①用户登录：用户通过登录界面输入用户名和密码进行身份认证，系统进行验证并授权登录。

②成绩数据管理：用户可以通过系统提供的功能对学生成绩数据进行管理，包括录入学生成绩、修改学生成绩、删除学生成绩等操作。

③成绩查询：用户可以通过系统提供的查询功能查询学生成绩，包括按学生、按课程、按班级等方式进行查询

④成绩分析和统计：系统可以根据学生成绩数据进行成绩分析和统计，包括生成成绩排名、计算平均分、计算总分等操作，以便用户对学生的成绩情况进行综合分析。

⑤用户管理：系统应该有用户管理功能，包括用户的注册、登录、密码管理、权限设置、用户信息维护等功能，方便管理员进行用户账号管理和权限控制。

在学生成绩信息管理系统中，通常包含以下几个程序模块，并且之间存在一定的层次关系（调用关系）：

①用户管理模块：负责处理用户注册、登录、密码管理等功能，并与主程序进行交互，提供用户认证。

②成绩数据管理模块：负责处理学生成绩数据的录入、修改、删除等操作，并与主程序进行交互，提供成绩数据的管理功能。

③成绩查询：负责处理学生成绩的查询，并与主程序进行交互。

④成绩分析和统计模块：负责处理成绩的分析和统计功能，包括成绩排名、平均分计算、总分计算等操作，并与主程序进行交互，提供分析和统计的接口。

**5 详细设计**

　　实现概要设计中定义的所有数据类型，给出关键部分源程序的清单，要求程序有充分的注释语句，至少要注释每个函数参数的含义和函数返回值的含义。

1. create database StudentSystemDao;//创数据库StudentSystemDao
2. use StudentSystemDao;
3. create table tb\_Department( *#创建院系表*
4. Department\_ID varchar(30) primary key, *#院系编号*
5. Department\_Name nvarchar(50) unique not null  *#院系名*
6. );
7. use StudentSystemDao;
8. create table tb\_Major(  *#创建专业表*
9. Major\_ID varchar(30) primary key, *#专业编号*
10. Major\_Name nvarchar(50) unique not null, *#专业名称*
11. Department\_ID varchar(30) not null, *#所属院系ID*
12. foreign key(Department\_ID) references tb\_Department(Department\_ID) *#所属院系设置为院系表的外键*
13. );
14. use StudentSystemDao;
15. create table tb\_Classe(  *#创建班级表*
16. Classe int not null, *#班级*
17. Grade int not null, *#年级*
18. Major\_ID varchar(30) not null, *#专业编号*
19. primary key(Classe,Grade,Major\_ID) *#班级，年级和专业编号共为主键*
20. );
21. use StudentSystemDao;
22. create table tb\_Student(  *#创建学生表*
23. Student\_Id varchar(30) primary key, *#学生学号*
24. Student\_Name nvarchar(50) not null,  *#学生姓名*
25. Student\_Sex nchar(1) default '男' not null check (StudentSex in('男','女')), *#学生性别*
26. Grade int not null, *#年级*
27. Classe int not null, *#班级*
28. Major\_ID varchar(30) not null, *#专业编号*
29. Major\_Name nvarchar(50)  not null, *#专业名称*
30. Department\_ID varchar(30) not null, *#所属院系ID*
31. Department\_Name nvarchar(50)  not null, *#院系名称*
32. foreign key(Department\_ID) references tb\_Department(Department\_ID), *#所属院系编号设置为外键*
33. foreign key(Major\_ID) references tb\_Major(Major\_ID), *#专业编号设置为外键*
34. foreign key(Department\_Name) references tb\_Department(Department\_Name), *#所属院系名称设置为外键*
35. foreign key(Major\_Name) references tb\_Major(Major\_Name) *#专业名称设置为外键*
36. );
37. use StudentSystemDao;
38. create table tb\_User(     *#创建用户表*
39. User\_name nvarchar(30) primary key,  *#用户名*
40. Password\_ varchar(30) not null,   *#密码*
41. IsLogin bit not null DEFAULT 0  *#是否重复登陆*
42. );
43. use StudentSystemDao;
44. create table tb\_Course(        *#创建课程表*
45. Course\_Name nvarchar(50) not null,   *#课程名称*
46. Major\_ID varchar(30) not null,      *#所属专业编号*
47. foreign key(Major\_ID) references tb\_Major(Major\_ID),*#专业编号设置为专业表的外键*
48. Grade int not null, *#年级*
49. primary key(Course\_Name,Major\_ID,Grade) *#课程名称，专业，年级共为主键*
50. );
51. use StudentSystemDao;
52. create table tb\_Score(            *#创建成绩表*
53. Student\_Id varchar(30) not null,        *#学生ID*
54. Student\_Name nvarchar(50) not null,  *#学生姓名*
55. Course\_Name varchar(30) not null,       *#课程名称*
56. primary key(Course\_Name,Student\_Id),    *#学生ID和课程名称共同组成主键*
57. foreign key(Student\_Id) references tb\_Student(Student\_Id), *#学生ID设置为学生表的外键*
58. foreign key(Course\_Name) references tb\_Course(Course\_Name),
59. Score dec(4,1) default null  check(Score between 0.0 and 100.0) *#成绩在0到100之间*
60. );
61. USE StudentSystemDao;
62. DELIMITER // #编写触发器
63. CREATE TRIGGER trg\_DeleteStudent
64. BEFORE DELETE ON tb\_Student FOR EACH ROW
65. BEGIN
66. *-- 删除学生在成绩表中的数据*
67. DELETE FROM tb\_Score WHERE Student\_Id = OLD.Student\_Id;*#在删除一个学生前将他在成绩表中的数据全部删除*
68. END;
69. //
70. DELIMITER ;

**6 调试分析**

　　内容包括：调试过程中遇到的问题是如何解决的以及对设计与实现的回顾讨论和分析；

①数据库连接出现java.lang.ClassNotFoundException: com.cj.mysql.jdbc.Driver报错。

解决方法：

在源程序目录下建立lib文件夹,去mysql官网下载mysql对应的版本的jdbc驱动包。

②删除学生表数据报错: Cannot delete or update a parent row: a foreign key constraint fails.出现[外键约束](https://so.csdn.net/so/search?q=%E5%A4%96%E9%94%AE%E7%BA%A6%E6%9D%9F&spm=1001.2101.3001.7020)失败。学生表中的student\_id是成绩表中student\_id的外键，两者相互绑定，在删除学生表数据之前应该保证成绩表中没有该学生的成绩信息。

解决方法：

在数据库中编写一个触发器，实现在删除学生表的数据之前将该学生在成绩表中的成绩数据删除。

1. CREATE TRIGGER trg\_DeleteStudent
2. BEFORE DELETE ON tb\_Student FOR EACH ROW
3. BEGIN
4. *-- 删除学生在成绩表中的数据*
5. DELETE FROM tb\_Score WHERE Student\_Id = OLD.Student\_Id;*#写一个触发器在删除一个学生前将他在成绩表中的数据全部删除*
6. END;

③成绩录入与实际情况不相符合。（例如原系统中成绩数据可以大于100或者小于0，与实际情况不符合）

解决办法：

输入成绩之后，判断输入的成绩是否与实际相符合（大于等于0且小于等于100），若输入的成绩与实际不符，弹出窗口

）

实现的源码：

1. public void actionPerformed(ActionEvent e) {
2. for(int i=0;i<courses.size();i++){
3. JTextField field = jtextFieldHashMap.get(courses.get(i));
4. String score = field.getText().trim();
5. if(score.equals("")){
6. JOptionPane.showMessageDialog(jd, "成绩不能为空！","",JOptionPane.WARNING\_MESSAGE);
7. return ;
8. }
9. if(Double.parseDouble(score)<0 || Double.parseDouble(score)>100){
10. JOptionPane.showMessageDialog(jd, "请输入0~100之间的成绩", "", JOptionPane.WARNING\_MESSAGE);
11. return ;
12. }
13. }
14. int i;
15. for(i=0;i<courses.size();i++){
16. JTextField field = jtextFieldHashMap.get(courses.get(i));
17. String score = field.getText().trim();
18. boolean b = helper.updateStudentScore(sm.getValueAt(rowNum, 0).toString(),courses.get(i),score);
19. if(!b){
20. break;
21. }
22. }
23. if(i<courses.size()){
24. JOptionPane.showMessageDialog(jd, "修改失败！","",JOptionPane.WARNING\_MESSAGE);
25. return ;
26. }else{
27. JOptionPane.showMessageDialog(jd, "修改成功！","",JOptionPane.WARNING\_MESSAGE);
28. jd.dispose();
29. return ;
30. }
32. }

**7 用户使用说明**

　　说明如何使用你编写的程序，详细列出每一步的操作步骤。

学生成绩信息管理系统是一个用于管理学生成绩信息的程序，以下是用户使用该程序的详细操作步骤：

①步骤1：登录系统

·打开学生成绩信息管理系统程序。

·输入用户名和密码进行登录。如果是首次使用系统，可以使用默认的管理员账号进行登录，或者根据系统要求创建一个新的账号。

·确认登录信息无误后，点击登录按钮。

②步骤2：管理学生信息

·登录成功后，系统会显示学生信息管理的主界面。

·在主界面上，可以选择添加新学生信息、查看学生信息、编辑学生信息或删除学生信息等操作。

·若要添加新学生信息，点击添加学生按钮，输入学生的基本信息，如学号、姓名、性别、年龄等，并点击确认按钮保存信息。

·若要查看学生信息，点击查看学生按钮，系统会显示已有学生的信息列表，可以根据需要选择查看特定学生的详细信息。

·若要编辑学生信息，点击编辑学生按钮，选择要编辑的学生，修改学生信息后点击确认按钮保存修改。

·若要删除学生信息，点击删除学生按钮，选择要删除的学生，系统会提示确认删除，点击确认按钮删除学生信息。

③步骤3：管理学生成绩

·在主界面上，可以选择录入学生成绩、查看学生成绩、编辑学生成绩或删除学生成绩等操作。

·若要录入学生成绩，点击录入成绩按钮，选择要录入成绩的学生，输入对应的科目和成绩，并点击确认按钮保存成绩信息。

·若要查看学生成绩，点击查看成绩按钮，系统会显示已有学生成绩的列表，可以根据需要选择查看特定学生的成绩信息。

·若要编辑学生成绩，点击编辑成绩按钮，选择要编辑成绩的学生，修改学生成绩后点击确认按钮保存修改。

·若要删除学生成绩，点击删除成绩按钮，选择要删除成绩的学生，系统会提示确认删除，点击确认按钮删除学生成绩。

④步骤4：查询学生信息和成绩

·在主界面上，可以选择按学号查询学生信息、按姓名查询学生信息、按科·目查询学生成绩或按学号查询学生成绩等操作。

·若要按学号查询学生信息，点击按学号查询按钮，输入学号并点击确认按钮，系统会显示匹配的学生信息。

·若要按姓名查询学生信息，点击按姓名查询按钮，输入姓名并点击确认按钮，系统会显示匹配的学生信息。

·若要按科目查询学生成绩，点击按科目查询按钮，输入科目名称并点击确认按钮，系统会显示匹配的学生成绩信息。

·若要按学号查询学生成绩，点击按学号查询按钮，输入学号并点击确认按钮，系统会显示匹配

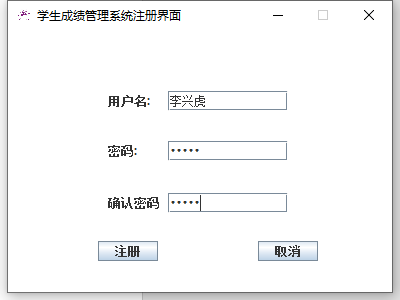
**8 测试结果**

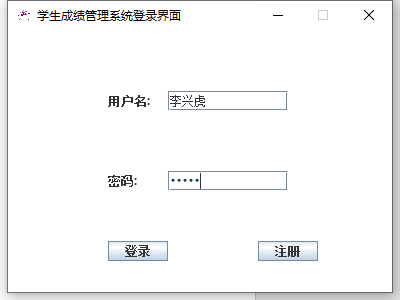
设计测试数据，或具体给出测试数据。要求测试数据完整和严格，能全面地测试所设计程序的功能。

①登录系统

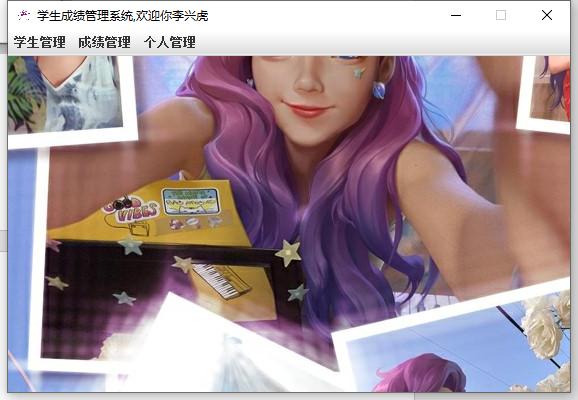
·打开学生成绩信息管理系统程序。

·输入用户名和密码进行登录。如果是首次使用系统根据系统要求创建一个新的账号。





登录成功：



②添加学生成绩：

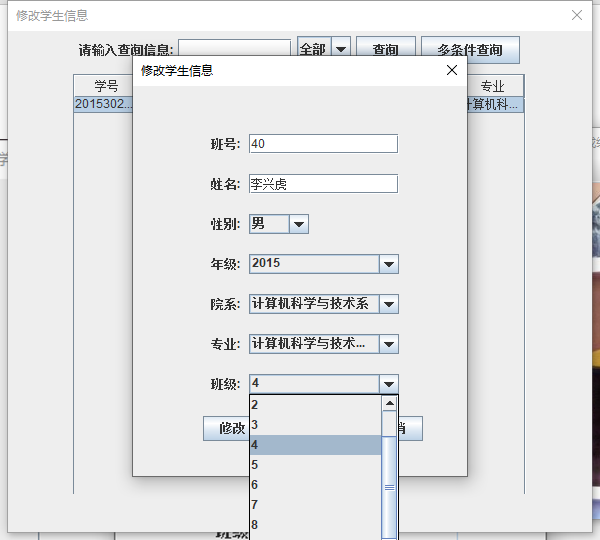


③查询已经录入的学生：

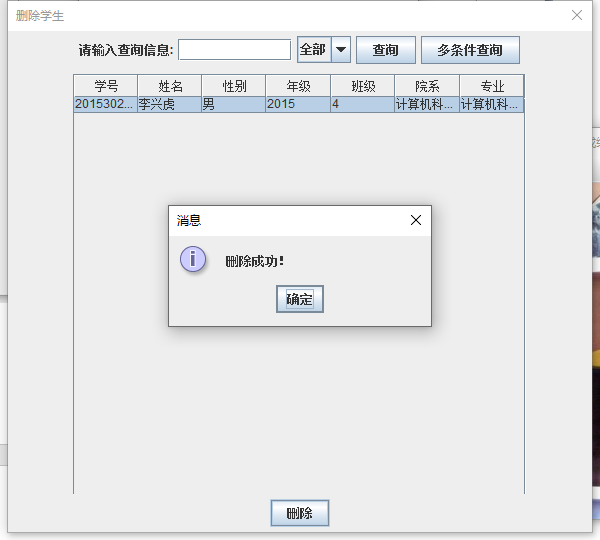
·可以根据学号、姓名、性别、班级、年级、专业、院系来查询详细的学生信息



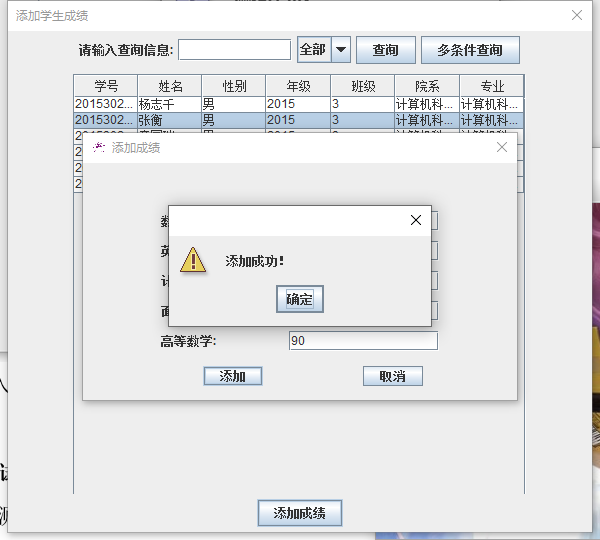
④修改学生信息



⑤删除学生信息



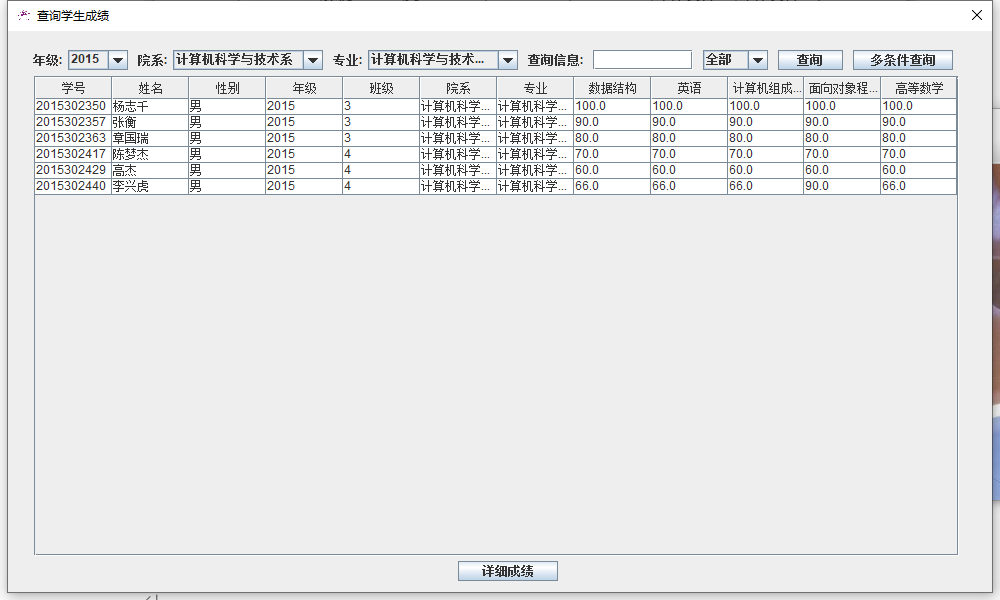
⑥录入成绩：

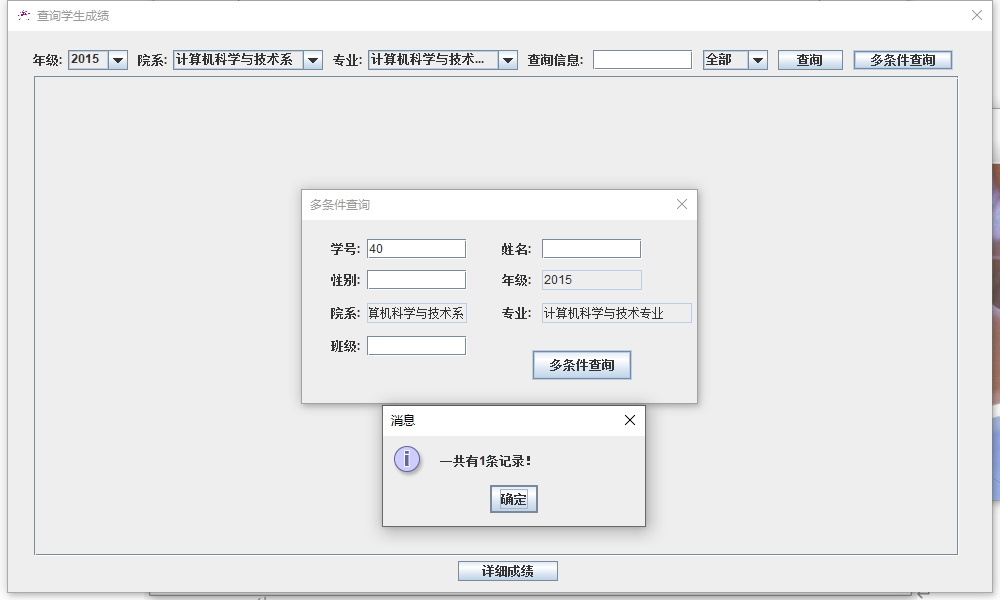


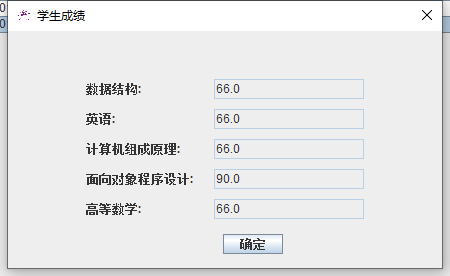
⑦修改成绩



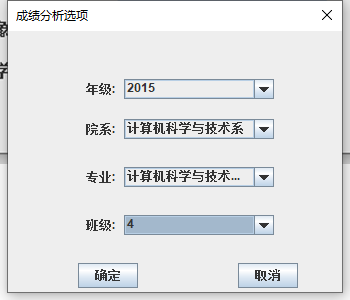
⑧查询成绩



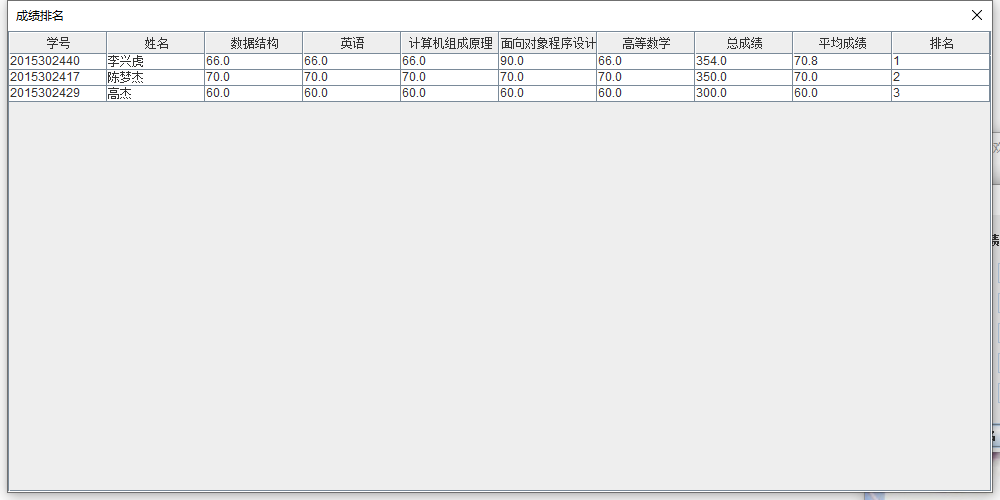




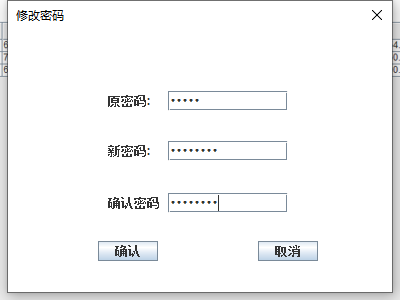
⑨成绩统计：







⑩修改用户密码和登出：



**9 课程设计总结**

本次课程设计主要实现了一个学生成绩信息管理系统，通过编写软件程序，实现了学生信息和成绩的添加、查看、编辑和删除等功能。在使用该系统时，用户需要登录系统，输入用户名和密码，然后可以进行学生信息和成绩的管理操作。通过本次课程设计，我深刻理解了软件开发的基本流程，包括需求分析、系统设计、编码实现和测试调试等环节。在需求分析阶段，我明确了系统的功能需求，包括学生信息和成绩的管理需求；在系统设计阶段，我设计了用户界面和数据库的结构；在编码实现阶段，我运用编程语言和数据库技术，完成了系统的开发；在测试调试阶段，我进行了系统的功能测试和错误修复，确保系统的正常运行。此外，我还学习了如何进行用户需求的收集和分析，如何进行数据库设计和编码实现，以及如何进行系统的测试和调试。通过课程设计的实践，我加深了对软件开发流程和技术的理解，掌握了一定的软件开发技能。在课程设计中，我遇到了一些困难和挑战，例如在数据库设计和编码实现过程中遇到了一些复杂的查询和操作逻辑，以及在系统测试中发现了一些错误和漏洞。但通过自学和查找资料，我克服了这些问题，并最终完成了一个能够满足需求的学生成绩信息管理系统。总的来说，本次课程设计使我掌握了软件开发的基本流程和技能，提高了我在需求分析、系统设计、编码实现和测试调试等方面的能力。通过实际操作，我深入理解了软件开发的实际应用，为我今后从事软件开发和项目管理等工作奠定了坚实的基础。

**10 参考文献**

列出参考的相关资料和书籍。

·《Java编程思想》- Bruce Eckel：这是一本经典的Java编程入门书籍，介绍了Java语言的基本概念、面向对象编程和Java编程的常用技术，包括数据库访问和GUI编程等内容。

·《Java数据库编程实战》- 王顶：详细介绍了Java程序如何通过JDBC技术与MySQL数据库进行交互，包括连接数据库、执行SQL语句、事务处理和连接池等方面的内容。

·《MySQL必知必会》- 珍妮弗·尼贝尔、比尔·卡尔：适合初学者的MySQL入门书籍，介绍了MySQL数据库的基本概念、SQL语句的使用、数据库设计和管理等内容，对学习MySQL数据库有很好的指导作用。

·《Java图形界面编程》- 曹春晖：详细介绍了Java的GUI编程技术，包括Swing和JavaFX库的使用、事件处理、布局管理和图形绘制等方面的内容，对学习Java图形界面编程有较全面的讲解。

·《Java核心技术 卷1 基础知识》- 凯 S.霍斯特曼：这是一本全面介绍Java语言的经典书籍，包括Java基础知识、面向对象编程和Java的高级特性等内容，对Java编程有深入的讲解。

·《深入理解Java虚拟机：JVM高级特性与最佳实践》- 周志明：详细介绍了Java虚拟机的原理、内存管理、垃圾回收等高级特性，对Java程序的性能优化和调优有很好的指导作用。

·《Java数据库编程技术详解》- 朱大刚、许华锋：介绍了Java程序如何通过JDBC技术与关系型数据库进行交互，包括连接数据库、执行SQL语句、事务处理和连接池等方面的内容，对Java数据库编程有很好的指导作用。

·《Java Swing 界面设计实战》- 程杰：详细介绍了Java Swing库的使用和界面设计的实践，包括Swing组件、布局管理、事件处理和界面设计的一些技巧和实践经验等内容。

·《JavaFX实战》- 张春：介绍了JavaFX库的使用和开发实践，包括界面设计、场景图、事件处理和动画等