**北京理工大学**

**毕业设计（论文）**

**论文题目：学生管理信息系统**

**专 业：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**班 级：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**姓 名：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**学 号：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**任务书**

**题目内容**

**题目：学生管理信息系统**

要求学生通过该系统可以对学生个人信息、班级信息、教师信息、课程信息进行检索，课程选择与成绩打印；教师通过该系统可以对教师个人信息、班级信息、学生信息、课程信息进行检索及课程成绩的录入；学生管理信息系统管理员通过该系统可以实现对以上信息进行检索与维护。

**一、任务要求**

1．系统开发环境：PowerBuilder + SQL2005

2．系统开发语言：PowerBuilder

3．系统功能：

(1) 实现学生、教师以及统管理员的登录。

(2) 实现学生对个人信息的维护。

(3) 实现教师对个人信息的维护。

(4) 实现学生对课程的选择、成绩的查看与打印。

(5) 实现教师对课程的维护与成绩的录入。

(6) 实现学生对个人信息、班级信息、教师信息、课程信息进行检索。

(7) 实现系统管理员对学生个人信息、班级信息、教师信息、课程信息的检索与维护。

**二、要求按期完成毕业设计（论文）各阶段任务**

要求应用软件工程的方法与数据库设计技术完成系统需求分析；系统概要设计及详细设计；应用PowerBuilder开发技术完成学生管理信息系统的实现和测试；根据毕业设计论文的格式要求完成毕业设计论文的撰写。

**三、进度安排**

1．第一阶段：要求完成系统需求分析；

2．第二阶段：要求完成系统设计；

3．第三阶段：要求完成系统功能实现与测试；

4．第四阶段：要求完成论文撰写。

指导教师签字： 年 月 日

# 毕业设计（论文）评语表（一）

|  |  |  |  |
| --- | --- | --- | --- |
| 指导教师对毕业设计（论文）的评语：  指导教师（签字）：  年 月 日 | | | |
| 评阅人对毕业设计（论文）的评语：  评阅人（签字）：  年 月 日 | | | |
| 答辩委员会（小组）成员 | | | |
| 姓名 | 职务 | 主要分工 | 备注 |
| 宋晔 | 副教授 | 组长 |  |
| 刘琼昕 | 副教授 | 组员 |  |
| 钟光太 | 副教授 | 组员 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| 答辩中提出的主要问题及回答的简要情况：  答辩委员会代表（签字）：  年 月 日 | | | |

# 毕业设计（论文）评语表（二）

|  |
| --- |
| 答辩委员会的评语：  答辩委员会代表（签字）：  年 月 日 |
| 答辩委员会给定的成绩：  答辩委员会代表（签字）：  年 月 日 |
| 毕业设计（论文）起止日期： 　年 月 日 至 年 月 日  毕业设计（论文）答辩日期: 年 月 日 |

# 摘要

本论文的研究课题是基于PowerBuilder程序设计技术的学生信息管理系统的设计与实现，要求学生通过该系统可以对学生个人信息、班级信息、教师信息、课程信息进行检索，课程选择与成绩打印；教师通过该系统可以对教师个人信息、班级信息、学生信息、课程信息进行检索及课程成绩的录入；学生管理信息系统管理员通过该系统可以实现对以上信息进行检索与维护。

本文前言叙述了课题背景和意义；第一章概述了系统搭建的框架，相关的软件选择和技术介绍；第二章叙述了系统需求分析与概要设计；第三章叙述了系统模块的相关流程及详细的设计与实现过程；第四章叙述了系统测试的目标与方法。

本系统重点阐述了学生、教师、班级、课程信息检索，学生选课及成绩录入与查询等问题的设计方案，详细设计了系统中的各管理模块，实现了学生信息管理的基本功能，达到了系统设计的预期目标，提高了教务管理工作的效率。

[关键词]： 信息管理; 学生信息管理;成绩管理;

# Abstract

This thesis research topic is based on PowerBuilder Program Designing Technology Student Information Management System Design and Implementation, Require students through the system to achieve the following functions: student personal information、class information、teacher inform -ation、course information for retrieval、Course Selection and transcripts print; Teachers through the system to achieve the following functions: teacher personal inform -ation、class information、student information、course information retrieval and academic record input; System administrator through the system can retrieve the above information and maintenance.  
 This introduction describes the background and signi -ficance of issues; Chapter One outlines the framework of the system structure, relevant software selection and technical presentations; The second chapter describes the system requirements analysis and summary of design; third chapter describes the relevant processes and system modules in detail design and implementation; fourth chapter- describes the objectives and methods of system testing.  
 The system focuses on the students, teachers, classes, course information retrieval, student course selection and transcripts input and query design, detailed design of the system in the management module, the students realize the basic functions of information management, reaching system design targets, improve the efficiency of educational administration.

[Keywords]: Information management; student information management; performance management;**目 录**

[前言 1](#_Toc259980449)

[1 系统开发相关技术概述 2](#_Toc259980450)

[1.1软件工程方法概述 2](#_Toc259980451)

[1.2 SQL2005数据库概述 2](#_Toc259980452)

[1.3 PowerBuilder技术概述 3](#_Toc259980453)

[2 系统需求分析与概要设计 5](#_Toc259980454)

[2.1系统功能预览 6](#_Toc259980455)

[2.2系统的业务流程图 6](#_Toc259980456)

[2.3系统功能结构分析与设计 7](#_Toc259980457)

[2.4数据库概念模型设计 9](#_Toc259980458)

[2.5数据流程设计 15](#_Toc259980459)

[2.5.1数据流程设计概述 15](#_Toc259980460)

[2.5.2学生管理信息系统的数据流设计流程 17](#_Toc259980461)

[2.6数据字典 18](#_Toc259980462)

[2.6.1数据流设计 18](#_Toc259980463)

[2.6.2数据处理设计 21](#_Toc259980464)

[2.6.3数据存储 26](#_Toc259980465)

[3 系统设计与实现 31](#_Toc259980466)

[3.1软件总体设计 31](#_Toc259980467)

[3.2数据库详细设计 32](#_Toc259980468)

[3.3系统功能模块设计 35](#_Toc259980469)

[3.3.1系统登录模块流程设计 35](#_Toc259980470)

[3.3.2登录模块界面设计 36](#_Toc259980471)

[3.3.3密码变更模块流程设计 39](#_Toc259980472)

[3.3.4密码变更模块界面设计 40](#_Toc259980473)

[3.3.5基础维护模块流程设计 42](#_Toc259980474)

[3.3.6基础信息维护模块界面设计 43](#_Toc259980475)

[3.3.7学生选课模块流程设计 47](#_Toc259980476)

[3.3.8 学生选课模块界面设计 48](#_Toc259980477)

[3.3.9学生成绩查询（打印）模块流程设计 50](#_Toc259980478)

[3.3.10成绩单打印模块界面设计 51](#_Toc259980479)

[3.3.11系统主窗体的设计 52](#_Toc259980480)

[3.3.12信息查询模块的设计 55](#_Toc259980481)

[3.3.13课表查询模块 58](#_Toc259980482)

[3.3.14成绩输入模块 60](#_Toc259980483)

[4 系统测试 62](#_Toc259980484)

[4.1软件测试概述 62](#_Toc259980485)

[4.2软件测试的目标 62](#_Toc259980486)

[4.3软件测试的方法 62](#_Toc259980487)

[4.3.1 静态测试 63](#_Toc259980488)

[4.3.2 动态测试 63](#_Toc259980489)

[4.3.3 黑盒测试与白盒测试 63](#_Toc259980490)

[4.4主要功能模块测试 64](#_Toc259980491)

[4.4.1登陆管理模块的测试 64](#_Toc259980492)

[4.4.2密码变更模块的测试 65](#_Toc259980493)

[4.4.3信息查询模块的测试 67](#_Toc259980494)

[4.4.4信息管理模块的测试 70](#_Toc259980495)

[4.4.5个人信息修改模块的测试 74](#_Toc259980496)

[4.4.6基础维护模块的测试 76](#_Toc259980497)

[结论 79](#_Toc259980498)

[致 谢 80](#_Toc259980499)

[参考文献 81](#_Toc259980500)

[附录 82](#_Toc259980501)

[使用说明书 82](#_Toc259980502)

[主要代码 83](#_Toc259980503)

# 前言

随着计算机及其应用技术的日益发展,特别是信息网络技术的飞速发展，已经大大地改变我们的生活方式，人类进入了信息时代。通过计算机网络，我们可以很方便地存储、交换以及搜索信息，给人的工作、生活以及娱乐带来了极大的方便,不仅提高了工作效率，而且大大的人与人之间沟通与交流的成本。

这股强大的信息化潮流早已拥入到大学校园的各个角落，无论是学校的教学管理还是后勤工作都与计算机网络有着千丝万缕的联系，当然，学生信息的管理工作也是各个高等院校教学管理工作中的一个重要环节，是整个院校管理工作的重点和基础。随着现代教育制度的改革，现代化的教学方法与教学思相的引进，越来越多的莘莘学子可以步入自己梦寐以求的高等院校，因些面对大量、种类繁多的数据和报表，手工处理方式已经很难跟上现代化管理的步伐，同时随着计算机及通信技术的飞速发展，高等教育对学生信息管理工作也提出了更高的要求。尽快改变传统的管理模式，运用现代化手段进行科学管理，已经成为整个教育系统亟待解决的课题之一。

因此，有必要开发一个学生信息管理系统，方便教务工作的开展与管理，为教师和学生提供一个良好的信息化互动与沟通的平台，使学生信息管理工作规范化，系统化，程序化，提高学生选课的速度和成绩查阅的准确性，能够及时、准确、有效的把学生信息、课程信息、成绩信息在学生与教师之间传递。

学生信息管理系统就是要运用先进的管理手段，组织、协调、指挥与指导学生与老师参于到学校教务工作活动中来，以便高效率、高质量地配合整个教务管理系统来完成教学任务，完成国家所制订的教育目标。

# 第一章 系统开发相关技术概述

## 1.1软件工程方法概述

软件工程是一门研究用工程化方法构建和维护有效的、实用的和高质量的软件的学科。它涉及到程序设计语言、数据库、软件开发工具、系统平台、标准、设计模式等方面。

在现代社会中，软件应用于多个方面。典型的软件比如有电子邮件、嵌入式系统、人机界面、办公包、操作系统、编译器、数据库、游戏等。同时，各个行业几乎都有计算机软件的应用，比如工业、农业、银行、航空、政府部门等。这些应用促进了经济和社会的发展，提高人们的工作效率，同时提升了生活质量。

软件工程师是对应用软件创造软件的人们的统称，软件工程师按照所处的领域不同可以分为系统分析员、软件设计师、系统架构师、程序员、测试员等等。人们也常常用程序员来泛指各种软件工程师。

软件生命周期: 同任何事物一样,一个软件产品或软件系统也要经历孕育、诞生、成长、成熟、衰亡等阶段，一般称为软件生存周期（软件生命周期）。把整个软件生存周期划分为若干阶段，使得每个阶段有明确的任务，使规模大，结构复杂和管理复杂的软件开发变的容易控制和管理。通常，软件生存周期包括可行性分析与开发项计划、需求分析、设计（概要设计和详细设计）、编码、测试、维护等活动，可以将这些活动以适当的方式分配到不同的阶段去完成。

软件生命周期(SDLC，软件生存周期)是软件的产生直到报废的生命周期，周期内有问题定义、可行性分析、总体描述、系统设计、编码、调试和测试、验收与运行、维护升级到废弃等阶段，这种按时间分程的思想方法是软件工程中的一种思想原则，即按部就班、逐步推进，每个阶段都要有定义、工作、审查、形成文档以供交流或备查，简化各阶段之间的联系，以提高软件的质量。

## 1.2 SQL2005数据库概述

数据库设计是数据库应用系统开发过程中首要的和基本的内容。数据库是信息系统的核心和基础，它把信息系统中的大量数据按照一定的模型组织起来，提供存储、维护、检索数据的功能，是信息系统可以方便、及时、准确从数据库中获取所需的信息。一个信息系统的各个部分能否紧密的结合在一起以及如何结合，关键在数据库,因此必须对数据库进行合理设计。

这一设计阶段是在需求分析的基础上，设计出能够满足用户需求的各种实体，以及他们之间的关系，为后面的逻辑结构设计打下基础。这个阶段不用考虑所采取的数据库管理系统、操作系统类型和机器类型等问题。这阶段可用的工具很多。

本系统采用SQL2005数据库，SQL Server 2005其强大的数据存储、分析、备份、还原、监控及安全功能， 使部署、管理和优化企业数据以及分析应用程序变得更简单、更容易。作为一个企业数据管理平台，它提供单一管理控制台，使数据管理员能够在任何地方监视、管理和调谐企业中所有的数据库和相关的服务。它还提供了一个可以使用 SQL 管理对象轻松编程的可扩展的管理基础结构，使得用户可以定制和扩展他们的管理环境，同时使独立软件供应商 (ISV) 也能够创建附加的工具和功能来更好地扩展打开即得的能力。

SQL Server 2005 为IT专家和信息工作者带来了强大的、熟悉的工具，同时减少了在从移动设备到企业数据系统的多平台上创建、部署、管理及使用企业数据和分析应用程序的复杂度。 通过全面的功能集、和现有系统的集成性、以及对日常任务的自动化管理能力，SQL Server 2005 为不同规模的企业提供了一个完整的数据解决方案, 为企业数据和分析应用程序提供更高的安全性、延展性和可用性，同时使它们更容易建置、部署和管理。帮助任何规模的企业和组织建置、部署和管理更安全、可延展又可靠的企业应用程序；简化开发及支持数据库应用程序的工作，以达到最高的 IT 生产力；实现跨多个平台、应用程序和装置共享数据，以便连接内部和外部系统；控制成本、降低风险、增强安全。

## 1.3 PowerBuilder技术概述

Powerbuilder是Powersoft（美国著名的数据库应用开发工具厂商，现已被Sybase收购）推出的可视化数据库集成开发工具，它是一个专业的客户机／服务器C/S架构开发模式，是第一个基于商业开发人员的面向对象编程（OOP）的应用程序，是一种“快速构建商业应用程序”的开发工具。PowerBuilder不仅有自己内置的数据库管理系统SYBASE SQL ANYWHERE，而且提供了对目前流行的大多数关系数据库管理系统的支持，其中最重要的一点就是提供了对MICROSOFT ODBC（数据库标准连接接口）的支持。由于在PowerBuilder的应用程序中对数据库访问的部分一般采用国际化标准数据库查询语言SQL，使得用PowerBuilder开发的应用程序可以不做修改或者只做少量的修改就可以在不同的后台数据库管理系统上使用。

PowerBuilder 提供了强大的开发功能，它集设计、建模、开发、部署、管理等各项功能为一体，进一步整合了新的增强特性、Web和N层应用开发的功能，包括了对.Net、PBDOM、XML DataWindow、RAD JavaServer Pages与PBNI等的支持，同时还提供了与Sybase应用服务器（Enterprise Application Server，EAServer）更为紧密的集成，支持跨平台的同一协作，其丰富的数据显示方式，可以满足各种不同的开发环境及功能上的需求，帮助开发者轻松、高效地完成开发工作。

本系统采用PowerBuilder11.0开发工具, 与SQL 2005 Enterprise Edition 相结合，基于在Windows server 2003 操作系统中完成对系统的开发。

# 第二章 系统需求分析与概要设计

软件需求分析工作是软件生存期中重要的一步，也是关键的一步。因此，必须要有对原业务的一个深入了解、提取、抽象、升华的过程。在需求分析阶段，要对经过可行性分析所确定的系统目标和功能做进一步的详细论述，以确定系统是“做什么”的。

需求阶段我们可以细分为需求调研和需求分析两个小阶段，需求调研需要充分细致的了解客户目标，客户业务内容、流程等，这是一个对需求的采集过程，是进行需求分析的基础准备。需求分析是对客户业务需求的一个升华，是一个把用户业务管理流程优化，转化为软件产品，从而提升管理而实现的质的飞跃，这一步是否成功，直接关系到开发出来的软件产品能否得到用户认可，顺利交付给客户，客户能否真正运用我们的产品帮助他解决业务或管理问题。

需求分析的主要任务是：定义软件的范围及必须满足的约束；确定软件的功能和性能及与其他系统成分的接口，建立数据模型、功能模型和行为模型；最终提供需求规格说明，并用做评估软件质量的依据。

需求分析的结果是系统开发的基础,关系到工程的成败和软件产品的量。因此,必须用行之有效的方法对软件进行严格的审查验证。

本系统所要实现的功能主要分为：

学生信息管理系统设计的主要目的是要将学生的个人信息、学生选课和学生成绩通过网络进行管理。为学生、教师和管理员提供便利。系统的用户有3种类型，分别为系统管理员、学生、教师，系统对于一个用户只允许以一种身份登录。系统管理员登录系统后可对系统进行管理，基主要操作是维护学生、教师、课程和班级等基本信息；学生登录后主要操作是选课和成绩查询和人个信息维护。教师登录后的主要操作是选择学生并为学生登录成绩。

* 将系统需求加以总结，得出系统需求如下。
* 系统可以运行在Windows操作系统平台上，并具有友好的用户界面。
* 系统用户类型为：学生、教师、管理员
* 系统对于一个用户只允许以一种身份登录
* 管理员可以维护学生、教师、课程和班级的基本信息
* 学生可以选课、成绩查询、教师信息查询和人个信息维护
* 教师查询课程、登录成绩、学生信息查询、教师信息查询、教师个人信息维护

## 2.1系统功能预览

通过前面的分析已经明确了系统用户共有如下3类。

* 管理员：管理学生、教师、课程和班级信息。
* 学生：选课、查看成绩、教师信息查询、修改个人信息
* 教师：询课程、登录成绩、学生信息查询、教师信息查询、教师个人信息维护

## 2.2系统的业务流程图

系统的功能模块图如图2-1所示:



图2-1业务流程图

## 2.3系统功能结构分析与设计

需求确定之后需要对系统进行整体分析和设计。这包括系统功能的描述、对功能模块的划分和系统流程的分析。下面首先对系统功能进行描述。

模块分析是描述系统需求的一个过程，需要将需求分析中的感性描述进行抽象，提到出要实现的功能，这是整个系统开发的一个关键过程。

需求分析的第一步，是描述学生信息管理系统的功能，以此确定系统的功能需求，学生信息管理系统的角色是管理员、教师和学生，管理员对学生、教师、课程和班级信息进行维护；学生选择想要上的课程，查看所选的课程的成绩、修改个人信息及教师信息的查询；教师可以对学生信息、课程信息、教师个人信息及学生成绩的录入。

根据以上用户操作需求，将系统划分为如下3大功能，并对其模块的划分和功能进行描述。

* 管理员功能
* 登录：登录
* 学生信息维护：查询、添加、修改、删除。
* 教师信息维护：查询、添加、修改、删除。
* 课程信息维护：查询、添加、修改、删除。
* 班级信息维护：查询、添加、修改、删除。
* 学生功能
* 登录：登录
* 学生个人信息：修改。
* 教师信息：查询。
* 选课：查询课程、选课。
* 成绩：查询、打印。
* 教师功能
* 登录：登录
* 学生信息：查询。
* 课程信息：查询。
* 成绩录入：查询课程、选择学生、成绩录入。

学生管理信息系统的功能模块图：如图2-2所示:



图2-2 系统功能模块图

## 2.4数据库概念模型设计

E-R图是描述数据实体及其关系的一种直观的描述工具。这种图中包括：

1、实体。用方框表示，方框内为实体的名称。

2、实体的属性。用椭圆表示，椭圆内为属性名称。使用线段将其和相应的实体联系起来。

3、、实体之间的关系。用菱形表示，菱形内为联系的名称。

E－R模型比较接近于人的思维，浅显易懂，容易让人接受，所以使用E－R模型作为用户与系统分析之间的工具。

本论文主要研究的实体属性图参见图2-3至图2-16，实体之间的关系E-R图参见图2-17.

4、本系统主要实体E-R图如下所示：

学生信息实体属性图2-3：

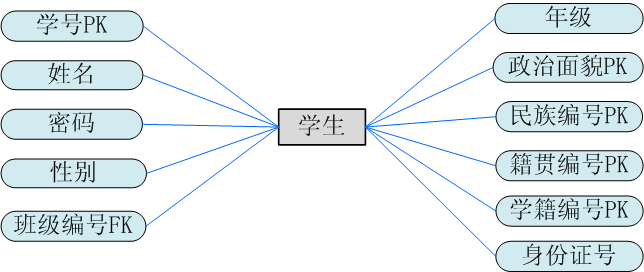


图2-3 学生信息实体的E-R图：

教师信息实体属性图2-4：



图2-4 教师信息实体的E-R图

班级信息实体属性图2-5：



图2-5 班级信息实体的E-R图

课程信息实体属性图2-6：



图2-6 课程信息实体的E-R图

课程表信息实体属性图2-7：



图2-7课表信息实体的E-R图

成绩信息实体属性图2-8：



图2-8 成绩信息实体的E-R图

系信息实体属性图2-9：

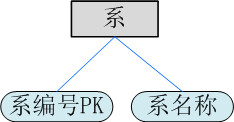


图2-9 系信息实体的E-R图

选课信息实体属性图2-10：



图2-10 选课实体的E-R图

政治面貌信息实体属性图2-11：



图2-11 政治面貌信息实体的E-R图

民族代码实体属性图2-12：



图2-12 民族代码实体的E-R图

职称代码实体属性图2-13：



图2-13 职称代码实体的E-R图

学籍实体属性图2-14：



图2-14 学籍实体的E-R图

籍贯实体属性图2-15：



图2-15 籍贯实体的E-R图

管理员实体属性图2-16：



图2-16 管理员实体的E-R图

学生管理信息系统总体E—R图2-17：



图2-17学生管理信息系统总E-R图

## 2.5数据流程设计

数据流程图（DFD—Data Flow Diagram)描绘系统的逻辑模型，图中没有任何具体的物理元素，只是描绘信息在系统中流动和处理的情况。因为数据流程图是逻辑系统的图形表示，即使不是专业的计算机技术人员也容易理解，所以是极好的通信工具。此外，设计数据流图只需考虑系统必须完成的基本逻辑功能，完全不需要考虑如何具体地实现这些功能，所以它也是软件设计的很好的出发点。

### 2.5.1数据流程设计概述

2.5.1.1 数据流程图  
 图是结构化系统分析的工具。它既可以表达数据在系统内部的逻辑流向及存储，又可以表达系统的逻辑功能和数据的逻辑变换。数据流程图既能表达现行人工系统的数据流程和逻辑处理功能，也能表达自动化系统的数据流程和逻辑处理功能。数据流程有四种基本符号：外部项、数据流、处理逻辑（加工）、数据元素和数据存储。

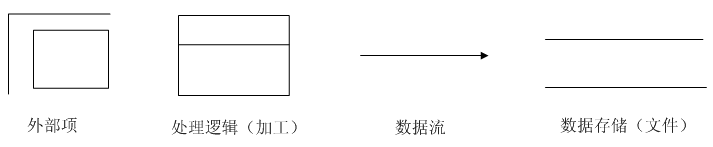


图2-18 数据流程基本符号

1. 外部项  
    外部项又称外部实体，是指不受系统控制的，在系统之外的事物或人。它表达了该系统的数据的外部来源或去处。它也可以是另外一个数据处理系统，它向该系统提供数据或接收来自该系统向它发出的数据。

（2） 数据流  
 数据流用箭头表示数据流动的方向，并给予命名。一般采用单箭头，偶尔使用双箭头。数据流可以由某一个外部项产生，也可以由某一个处理逻辑产生，还可以来自某一个数据存储。一般来说，对每一个数据流可以在数据流箭头的上方加以简单的描述；对一些含义比较明显的数据流，就不一定作描述。也可以在数据流上写记号，然后另外描述记号的意义。

（3） 处理逻辑（加工）  
 处理逻辑对数据的变换方式有两种：

A、变换数据的结构

B、在原有数据内容基础上产生新的数据内容

（4） 数据元素  
 数据元素是数据的最小组成单位，也就是不可分的数据单位。数据元素是数据流或数据存储中的基本成分。

（5）数据存储（文件）  
 数据存储用长方条表记，在长方条内部写上该数据存储的名称。用作标识的编号一般用英文字母D和数字组成。同外部项一样，允许在一张数据流程式图上重复出现相同的数据存储，以避免数据流线的交叉，这时应在重复的数据存储符号的左侧再加一条竖线。

一个处理逻辑可能要从数据存储中读出某些数据，或者可能把一些数据存入到某个数据存储中，甚至修改数据存储中的某些数据，那么就得用数据流将处理逻辑和数据存储联结起来。

2.5.1.2 数据流程图的分解  
 编制复杂的数据流程图，最好的办法是采用自顶向下扩展逐层分解。首先是系统关联图，给出外部实体与即将开发的计算机管理信息系统之间的数据流。哪些数据流从外部实体进入系统，又有哪些数据流从系统输出给外部实体。关联图回答系统从外部世界得到什么，系统将给外部世界又是什么。从关联图分解得到顶层图，又从顶层图分解得到一层数据流程图，再分解出二层数据流程图。在分解过程中，随着更具体和更详细，新的数据流和数据存储被引入，但外部世界输入到系统，系统输出到外部世界，在关联图中提及的那些数据流是不能再增加，也不允许被减少的。

在上述分解过程中，上层的一个处理逻辑可能被分解成多个更具体的处理逻辑，新的数据存储和数据流被被引入。如此逐一分解扩展，直至不需要再分解为止。

### 2.5.2****学生管理信息系统的数据流设计流程****



图2-19 数据流程图

## 2.6数据字典

系统分析中所使用的数据字典，主要用来描述数据流程图中的数据流、数据存储、处理过程和外部实体。数据字典把数据的最小组成单位看成是数据元素(基本数据项)，若干个数据元素可以组成一个数据结构(组合数据项)。数据结构是一个递归概念，即数据结构的成分也可以是数据结构。数据字典通过数据元素和数据结构来描写数据流、数据存储的属性，数据元素组成数据结构，数据结构组成数据流和数据存储。

### 2.6.1数据流设计

表2.6.1.1系统数据流设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **编号** | **名称** | **组成** | **来源** | **去向** |
| D1.1 | 班级信息录入 | 班级编号 + 年级 + 班级名称 + 班级简称 + 人数 + 班主任+系编号 | 管理员 | 班级信息录入处理 |
| D1.2 | 班级信息 | 班级编号 + 年级 + 班级名称 + 班级简称 + 人数 + 班主任 | 管理员 | F1.1班级信息表 |
| D1.3 | 班级信息查询 | 班级编号 / 年级 / 班级名称 | F1.1班级信息表 | 班级信息查询处理 |
| D1.4 | 班级信息  维护 | 班级编号 + 年级 + 班级名称 + 班级简称 + 人数 + 班主任+系编号 | F1.1班级信息表 | F1.2班级信息表 |
| D2.1 | 学生信息录入 | 学号 + 姓名 + 班级编号 + 性别 + 年级 + 政治面貌编号 + 民族编号 + 籍贯编号 + 身份证号 + 学籍编号 | 管理员 | 学生信息录入处理 |
| D2.2 | 学生信息 | 学号 + 姓名 + 班级编号 + 性别 + 年级 + 政治面貌编号 + 民族编号 + 籍贯编号 + 身份证号 + 学籍编号 | 管理员 | F2.1学生信息表 |
| D2.3 | 学生信息查询 | 姓名 / 姓名 / 年级 | F2.1学生信息表 | 学生信息查询处理 |
| D2.4 | 学生信息维护 | 学号 + 姓名 + 班级编号 + 性别 + 年级 + 政治面貌编号 + 民族编号 + 籍贯编号 + 身份证号 + 学籍编号 | F2.1学生信息表 | F2.2学生信息表 |
| D3.1 | 课程信息录入 | 课程编号 + 课程名称 + 课程简称 + 拼音码 + 本学期课程 + 教师 + 开课系别 + 学分 | 管理员 | 课程信息录入处理 |
| D3.2 | 课程信息 | 课程编号 + 课程名称 + 课程简称 + 拼音码 + 本学期课程 + 教师 + 开课系别 + 学分 | 管理员 | F3.1课程信息表 |
| D3.3 | 课程信息查询 | 课程编号 / 课程名称 / 拼音码 | F3.1课程信息表 | 课程信息查询处理 |
| D3.4 | 课程信息维护 | 课程编号 + 课程名称 + 课程简称 + 拼音码 + 本学期课程 + 教师 + 开课系别 + 学分 | F3.1课程信息表 | F3.2课程信息表 |
| D4.1 | 教师信息录入 | 教师编号 + 姓名 + 密码 + 性别 + 政治面貌 + 民族编号 + 籍贯编号 + 职称编号 + 家庭地址 + 联系电话 + 系编号 | 管理员 | 教师信息录入处理 |
| D4.2 | 教师信息 | 教师编号 + 姓名 + 密码 + 性别 + 政治面貌 + 民族编号 + 籍贯编号 + 职称编号 + 家庭地址 + 联系电话 + 系编号 | 管理员 | F4.1教师信息表 |
| D4.3 | 教师信息查询 | 教师编号 / 姓名 | F4.1教师信息表 | 教师信息查询处理 |
| D4.4 | 教师信息维护 | 教师编号 + 姓名 + 密码 + 性别 + 政治面貌 + 民族编号 + 籍贯编号 + 职称编号 + 家庭地址 + 联系电话 + 系编号 | F4.1教师信息表 | F4.2教师信息表 |
| D5.1 | 教师登录 | 姓名 + 密码(教师) | F4.1教师信息表 | 教师个人信息验证 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| D6.1 | 课程信息查询 | 课程编号 / 课程名称 / 拼音码 | F3.1课程信息表 | 课程信息查询处理 |
| D6.2 | 课程信息表 | 课程编号 + 课程名称 + 课程简称 + 拼音码 + 本学期课程 + 教师 + 开课系别 + 学分 | F3.1课程信息表 | 查询结果 |
| D7.1 | 学生信息查询 | 姓名 / 姓名 / 年级 | F2.1学生信息表 | 学生信息查询处理 |
| D7.2 | 学生信息表 | 学号 + 姓名 + 班级编号 + 性别 + 年级 + 政治面貌编号 + 民族编号 + 籍贯编号 + 身份证号 + 学籍编号 | F2.1学生信息表 | 查询结果 |
| D8.1 | 汇总信息 | 入库信息+出库信息+维修信息  +报废信息 | 汇总信息 | F5机房设备汇总信息表 |
| D9.1 | 课程信息查询 | 课程编号 / 课程名称 / 拼音码 | F3.1课程信息表 | 课程信息查询处理 |
| D9.2 | 课程信息表 | 课程编号 + 课程名称 + 课程简称 + 拼音码 + 本学期课程 + 教师 + 开课系别 + 学分 | F3.1课程信息表 | 查询结果 |
| D10.1 | 教师信息查询 | 教师编号 / 姓名 | F4.1教师信息表 | 教师信息查询处理 |
| D10.2 | 教师信息表 | 教师编号 + 姓名 + 密码 + 性别 + 政治面貌 + 民族编号 + 籍贯编号 + 职称编号 + 家庭地址 + 联系电话 + 系编号 | F4.1教师信息表 | 查询结果 |
| D11.1 | 学生选课 | 课程编号 / 课程名称 / 课程拼音 | 课程信息表+课程表 | 学生选课处理 |
| D12.1 | 教师查询课程 | 课程编号 / 课程名称 / 课程拼音 | F5.2课程表 | 课程表查询处理 |
| D12.2 | 成绩录入 | 课程编号 + 课程名称 + 学号 + 姓名 + 成绩 | 教师 | 课程成绩录入处理 |
| D13.1 | 学生成绩查询 | 学号 | F6.2课程成绩表 | 成绩查询处理 |
| D14.1 | 打印成绩单 | 课程名称 + 开课系别 + 教师 + 成绩 | F7.2学生各科成绩单 | 成绩打印处理 |

### 2.6.2数据处理设计

|  |
| --- |
| 编号：P1.1  输入的数据流：D1.1  文件信息：F1.1  输出的数据流：D1.2  功能说明：班级信息录入，管理员录入班级信息。 |

|  |
| --- |
| 编号：P1.2  输入的数据流：D1.3  输出的数据流：D1.4  功能说明：班级信息查询处理，管理员查询班级信息，查看班级是否在表。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P1.3  输入的数据流：D1.3  文件信息：F1.2  输出的数据流：D1.4  功能说明：班级信息维护处理，管理员对班级信息进行新增、修改、删除、保存操作。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P2.1  输入的数据流：D2.1  文件信息：F2.1  输出的数据流：D2.2  功能说明：学生信息录入，管理员录入学生信息。 |

|  |
| --- |
| 编号：P2.2  输入的数据流：D2.3  输出的数据流：D2.4  功能说明：学生信息查询处理，管理员查询学生信息，查看学生是否在表。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P2.3  输入的数据流：D2.3  文件信息：F2.2  输出的数据流：D2.4  功能说明：学生信息维护处理，管理员对学生信息进行新增、修改、删除、保存操作。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P3.1  输入的数据流：D2.1  文件信息：F2.1  输出的数据流：D2.2  功能说明：课程信息录入，管理员录入课程信息。 |

|  |
| --- |
| 编号：P3.2  输入的数据流：D2.3  输出的数据流：D2.4  功能说明：课程信息查询处理，管理员查询课程信息，查看课程是否在表。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P3.3  输入的数据流：D2.3  文件信息：F2.2  输出的数据流：D2.4  功能说明：课程信息维护处理，管理员对课程信息进行新增、修改、删除、保存操作。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P4.1  输入的数据流：D4.1  文件信息：F4.1  输出的数据流：D4.2  功能说明：教师信息录入，管理员录入教师信息。 |

|  |
| --- |
| 编号：P4.2  输入的数据流：D4.3  输出的数据流：D4.4  功能说明：教师信息查询处理，管理员查询教师信息，查看教师是否在表。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P4.3  输入的数据流：D4.3  文件信息：F4.2  输出的数据流：D4.4  功能说明：教师信息维护处理，管理员对教师信息进行新增、修改、删除、保存操作。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P5.1  输入的数据流：D5.1  文件信息：F4.2  输出的数据流：D5.2  功能说明：教师个人信息验证处理，进行系统登录。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P6.1  输入的数据流：D6.1  输出的数据流：D6.2  功能说明：课程信息查询处理，教师查看相关课程信息。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P7.1  输入的数据流：D7.1  输出的数据流：D7.2  功能说明：学生信息查询处理，教师查看学生相关信息。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P8.1  输入的数据流：D8.1  输出的数据流：D8.2  功能说明：学生个人信息验证处理，进行系统登录。  出错信息：出错后提示管理8 |

|  |
| --- |
| 编号：P9.1  输入的数据流：D6.1  输出的数据流：D6.2  功能说明：课程信息查询处理，学生查看相关课程信息。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P10.1  输入的数据流：D10.1  输出的数据流：D10.2  功能说明：教师信息查询处理，学生查询查看相关教师信息。  出错信息：出错后提示管理员 |

|  |
| --- |
| 编号：P11.1  输入的数据流：D11.1  文件信息：F5.1  输出的数据流：D11.2  功能说明：学生选课处理，构成课程信息表。 |

|  |
| --- |
| 编号：P12.1  输入的数据流：D12.1  输出的数据流：D12.2  功能说明：课程信息查询处理，教师查询查看相关课程信息。  出错信息：出错后提示管教师 |

|  |
| --- |
| 编号：P12.2  输入的数据流：D12.2  文件信息：F6.1  输出的数据流：D13.2  功能说明：课程成绩录入处理，形成课程成绩表。 |

|  |
| --- |
| 编号：P13.1  输入的数据流：D13.1  文件信息：F7.1  输出的数据流：D14.1  功能说明：成绩查询处理，输出相应的成绩信息。 |

### 2.6.3数据存储

表2.6.3**-**1 学生信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 学号 | Varchar | 14 | PK | NO |
| 姓名 | Varchar | 8 |  | NO |
| 密码 | Varchar | 8 |  | NO |
| 班级编号 | Varchar | 14 |  | YES |
| 性别 | Varchar | 2 |  | YES |
| 年级 | Varchar | 8 |  | YES |
| 政治面貌编号 | Varchar | 2 | FK | NO |
| 民族编号 | Varchar | 2 | FK | NO |
| 籍贯编号 | Varchar | 2 | FK | NO |
| 身份证号 | Varchar | 18 |  | NO |
| 学籍编号 | Varchar | 2 | FK | YES |

表2.6.3-2 教师信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 教师编号 | varchar | 14 | PK | NO |
| 姓名 | varchar | 8 |  | NO |
| 密码 | varchar | 8 |  | NO |
| 班级编号 | varchar | 14 | FK | NO |
| 性别 | varchar | 2 |  | NO |
| 政治面貌编号 | varchar | 2 | FK | NO |
| 民族编号 | varchar | 2 | FK | NO |
| 籍贯编号 | varchar | 2 | FK | NO |
| 职称证号 | varchar | 2 | FK | NO |
| 家庭地址 | varchar | 30 |  | YES |
| 联系电话 | varchar | 20 |  | YES |
| 系编号 | varchar | 3 | FK | NO |

表2.6.3**-**3 班级信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 班级编号 | varchar | 14 | PK | NO |
| 系编号 | varchar | 2 | FK | NO |
| 年级 | varchar | 4 |  | YES |
| 班级名称 | varchar | 30 |  | YES |
| 班级简称 | varchar | 16 |  | YES |
| 人数 | Int | 3 |  | YES |
| 班主任 | varchar | 8 |  | YES |

表2.6.3**-**4 课程信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 课程编号 | int | 4 | PK | NO |
| 课程名称 | char | 40 |  | YES |
| 课程简称 | char | 40 |  | YES |
| 课程简码 | char | 6 |  | YES |
| 本学期课程 | char | 2 |  | YES |
| 教师编号 | varchar | 14 | FK | NO |
| 系编号 | varchar | 2 | FK | NO |
| 学分 | int | 2 |  | YES |

表2.6.3**-**5 课程表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 编号 | int | 2 | PK | NO |
| 课序号 | varchar | 14 | FK | NO |
| 课程编号 | int | 2 | FK | NO |
| 上课时间天 | int | 2 |  | YES |
| 上课时间节 | int | 2 |  | YES |
| 上课地点 | varchar | 20 |  | YES |

表2.6.3**-**6 成绩表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 编号 | int | 2 | PK | NO |
| 学号 | varchar | 14 | FK | NO |
| 课程编号 | int | 2 | FK | NO |
| 成绩 | int | 2 |  | YES |
| 考试次数 | int | 1 |  | YES |
| 是否补修 | varchar | 2 |  | YES |
| 是否重考 | varchar | 2 |  | YES |
| 是否已确定成绩 | varchar | 2 |  | YES |

表2.6.3**-**7选课表表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 编号 | int | 2 | PK | NO |
| 学号 | varchar | 14 | FK | NO |
| 课序号 | varchar | 14 | FK | NO |

表2.6.3-8 系代码表表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 系编号 | varchar | 2 | PK | NO |
| 系名称 | varchar | 8 |  | NO |

表2.6.3**-**9政治面貌代码表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 政治面貌编号 | varchar | 2 | PK | NO |
| 政治面貌 | varchar | 10 |  | NO |

表2.6.3**-**10 民族代码表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 民族编号 | varchar | 2 | PK | NO |
| 民族 | varchar | 14 |  | NO |

表2.6.3**-**11 籍贯代码表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 籍贯编号 | varchar | 2 | PK | NO |
| 籍贯 | varchar | 14 |  | NO |

表2.6.3**-**12 职称代码表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 职称编号 | varchar | 2 | PK | NO |
| 职称 | varchar | 14 |  | NO |

表2.6.3**-**13 学籍代码表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 学籍编号 | int | 2 | PK | NO |
| 学籍名称 | varchar | 8 |  | NO |

表2.6.3**-**14 系统用户信息表（SYSTEMUSER\_INFO)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **数据类型** | **字段大小** | **索引** | **是否可为空** |
| 管理员编号 | varchar | 14 | PK | NO |
| 姓名 | varchar | 8 |  | NO |
| 密码 | varchar | 8 |  | NO |

# 第三章 系统设计与实现

## 3.1软件总体设计

系统总体设计是在根据任务书、数据流程图、业务流程图、数据字典、功能模块、ER-图等所有相关因素的基础上，结合实体生产环境中的应用条例，完成系统需求分析中规定的所有功能。确定系统的实施方案，即系统的物理模型。

系统总流程图如图3-1所示：



图3-1 系统总体设计

## 3.2数据库详细设计

数据库详细设计即为数据库逻辑设计，根据数据库的概念模型来设计数据模型，再根据数据模型在开发环节中来设计数据表关系。

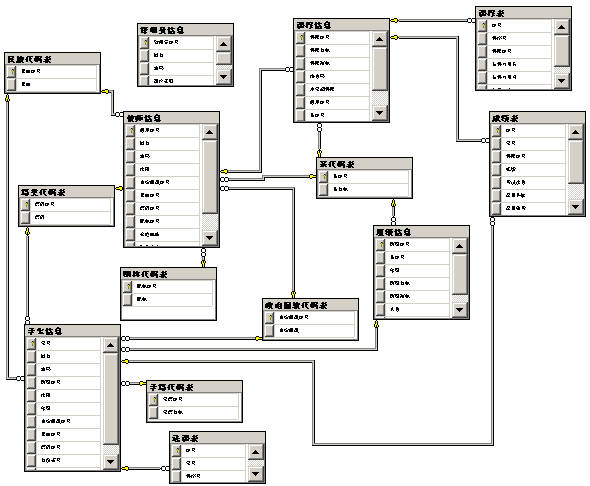


图3-2 库表关系图（SQL2005）

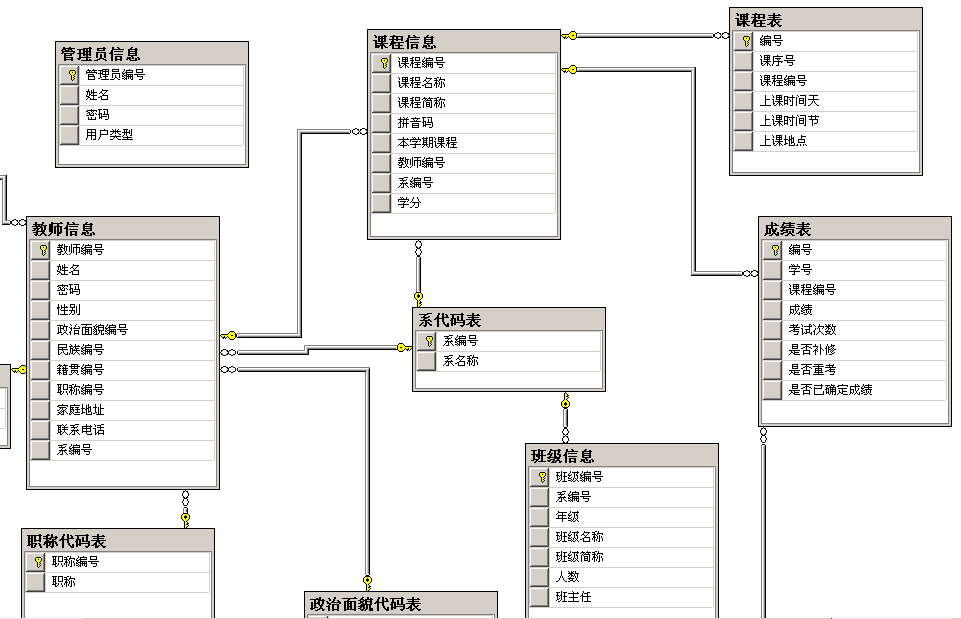


图3-2 库表关系图（SQL2005）局部放大图(1)

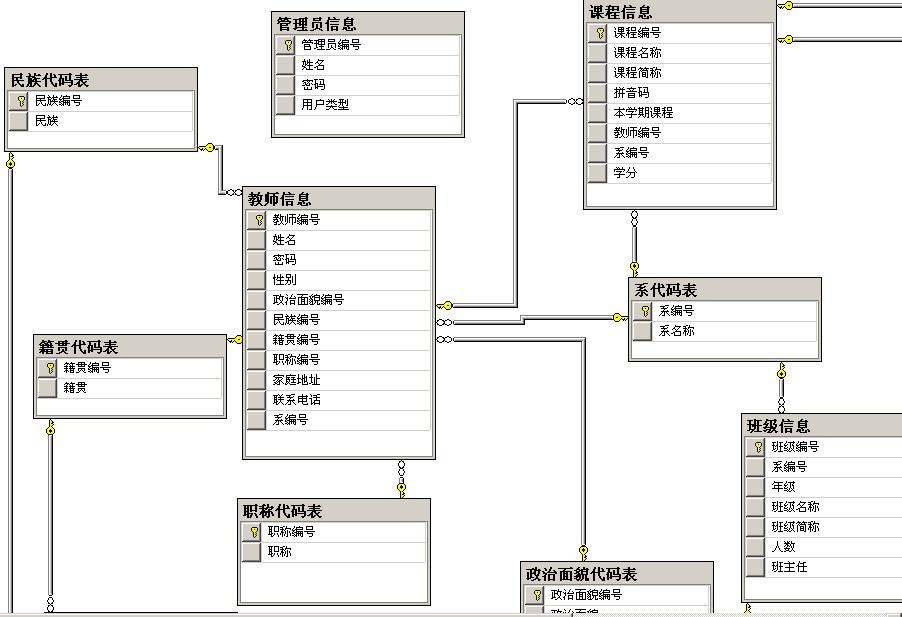


图3-2 库表关系图（SQL2005）局部放大图(2)

图3-3 库表关系图（PB11.0）

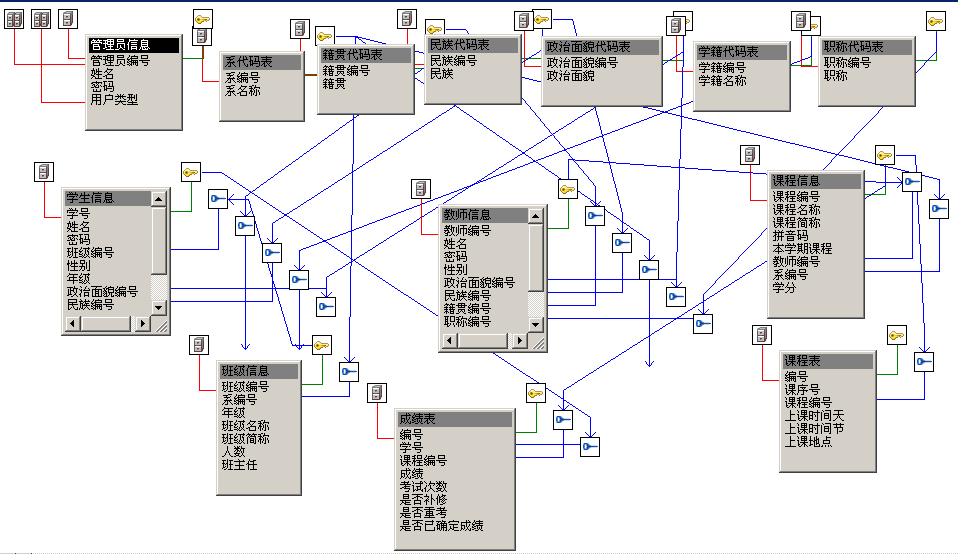


图3-3 库表关系图（PB11.0）

## 3.3系统功能模块设计

### 3.3.1系统登录模块流程设计

功能：本系统共有三个角色，学生、教师、管理员，不同的角色登录后有不同的权限和相对应的菜单才登录界面。

系统登录流程图3-4所示：

图3-4 系统登录流程图

学生

管理员

教师

### 3.3.2登录模块界面设计

* 1. 功能概述：用户类别设定为3个等级：管理员、教师、学生，用户必须输入正确的用户名和密码才能进入，系统根据用户输入的用户名及密码判定用户的权限，进入相应的界面。
  2. 登录界面
  + 学生登录界面 (如图3-5所示)



图3-5 学生登录界面

* + 教师登录界面 (如图3-6所示)



图3-6 教师登录界面

* + 管理员登录界面 (如图3-7所示)



图3-7 管理员登录界面

* 1. 重要代码
* “确定”按钮clicked()

string password,account,password2

int reinput

password=trim(sle\_2.text) //trim去除空格

if password='' then

messagebox('密码错误！','管理员的密码为空值，请重新输入!')

sle\_2.setfocus()

else

account=trim (sle\_1.text)

if lb\_type.text = '管理员' then

select 密码

into:password2

from 管理员信息

where 姓名=:account;

elseif lb\_type.text = '学生' then

select 密码

into:password2

from 学生信息

where 姓名=:account;

elseif lb\_type.text = '教师' then

select 密码

into:password2

from 教师信息

where 姓名=:account;

end if // if lb\_type.text = '管理员' then

if password2='' then

messagebox('账号错误','此帐号不存在或密码错误，请重新输入')

sle\_1.setfocus()

sle\_1.text=''

sle\_2.text=''

elseif password=password2 then

g\_account = account

g\_accountType = lb\_type.text

if lb\_type.text='管理员' then

open(w\_main)

close(w\_login)

elseif lb\_type.text='教师' then

open(w\_main)

### 3.3.3密码变更模块流程设计

功能：本系统共有三个角色，学生、教师、管理员，不同的角色成功登录后可以对自己的密码进行变更。

 密码变更流程图3-8所示：



学生

管理员

教师

图3-8密码变更模块流程图

### 3.3.4密码变更模块界面设计

* 1. 功能概述：用户类别设定为3个等级：管理员、教师、学生，用户成功登录后，可以对自己的密码进行修改，以确定登录信息的安全性。
  2. 界面图设计(如图3-9所示)



图3-9 密码修改界面

* 1. 重要代码
* 确定“确定”按钮clicked()

string oldpwd, newpwd, newpwd2

string pwdverify

oldpwd=trim(sle\_2.text) //trim去除空格

newpwd=trim(sle\_3.text)

newpwd2=trim(sle\_4.text

if oldpwd=''then

messagebox('密码错误！','用户密码为空值，请重新输入!')

sle\_2.setfocus()

else

if g\_accountType = '管理员' then

select 密码

into:pwdverify

from 管理员信息

where 姓名=:g\_account;

elseif g\_accountType = '学生' then

select 密码

into:pwdverify

from 学生信息

where 姓名=:g\_account;

elseif g\_accountType = '教师' then

select 密码

into:pwdverify

from 教师信息

where 姓名=:g\_account;

end if // if g\_accountType = '管理员' then

if oldpwd=pwdverify then

if newpwd='' or newpwd2='' then

messagebox('错误提示','新密码为空')

sle\_3.text=''

sle\_4.text=''

sle\_3.setfocus( )

elseif newpwd<>newpwd2 then

messagebox('错误提示','新密码不一致')

sle\_3.text=''

sle\_4.text=''

sle\_3.setfocus( )

else

if g\_accountType = '管理员' then

update 管理员信息

### 3.3.5基础维护模块流程设计

功能：管事员成功登录后，可以对学生信息、班级信息、教师信息、课程信息等进行维护（添加／修改／删除）。

**** 基础维护模块流程图3-10所示：



图3-10基础维护模块流程

### 3.3.6基础信息维护模块界面设计

* 基础维护模块

1. 功能概述：管理员登录后，可以对学生信息、班级信息、课程信息、教师信息进行维护（添加／修改／删除）。
2. 界面图设计(如图3-11、12、13、14所示)

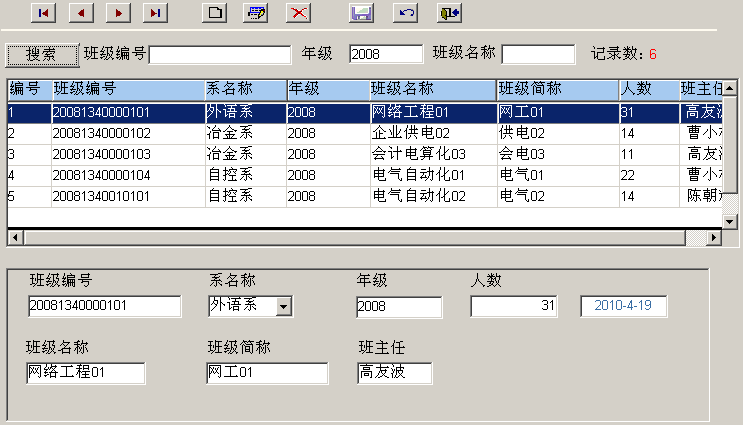


图3-11班级信息维护界面



图3-12学生信息维护界面



图3-13教师信息维护界面



图3-14课程信息维护界面

1. 主要代码

string ls\_select,ls\_bjbh,ls\_nj,ls\_njmc

ls\_bjbh = trim(sle\_bjbh.text) // 拼音码

ls\_nj = trim(sle\_nj.text) // 编号

ls\_njmc = trim(sle\_njmc.text) // 名称

ls\_select = ""

// 班级编号

if not isnull(ls\_bjbh) and ls\_bjbh <> "" then

ls\_bjbh = "%" + ls\_bjbh + "%"

ls\_select = ls\_select + " 班级编号 LIKE '" + ls\_bjbh + "'"

end if

// 年级

if not isnull(ls\_nj) and ls\_nj <> "" then

ls\_nj = "%" + ls\_nj + "%"

if ls\_select = "" then

ls\_select = " 年级 LIKE '" + ls\_nj + "'"

else

ls\_select = ls\_select + " AND 年级 LIKE '" + ls\_nj + "'"

end if

end if

if not isnull(ls\_njmc) and ls\_njmc <> "" then

ls\_njmc = "%" + ls\_njmc + "%"

if ls\_select = "" then

ls\_select = " 班级名称 LIKE '" + ls\_njmc + "'"

else

ls\_select = ls\_select + " AND 班级名称 LIKE '" + ls\_njmc + "'

end if

end if

pb\_add:

//设置数据窗状态

dw\_1.enabled = false

dw\_2.enabled = true

### 3.3.7学生选课模块流程设计

功能：学生成功登录后，如果是在籍的学生（未毕业），可以对课程信息进行检索，查询相关的课程，进行选课。

**** 学生选课模块流程图3-15所示：



图3-15学生选课模块流程

### 3.3.8 学生选课模块界面设计

1. 功能概述：学生可对课程信息查询，并进行相关课程的选择。
2. 界面图设计(如图3-16所示)



图3-16 学生选课界面

1. 重要代码

代码：



integer li\_row,li\_insertrow,li\_count

string ls\_kxh,ls\_kcmc,ls\_jsbh,ls\_xbh,ls\_dd,ls\_xh,ls\_oldkxh

integer li\_sjt,li\_sjj,li\_kcbh,i

datastore lds\_kxh

li\_row = dw\_2.getselectedrow(0)

if li\_row < 1 then return

ls\_xh = trim(sle\_xh.text)

if isnull(ls\_xh) or ls\_xh = "" then

messagebox("提示","请先输入学号！")

return

end if

li\_kcbh = dw\_2.getitemnumber(li\_row,"课程编号")

ls\_kxh = dw\_2.getitemstring(li\_row,"课序号")

//判断是否已经选择该课程

SELECT count(\*)

INTO :li\_count

FROM 选课表,

课程信息,

课程表

WHERE ( 课程表.课程编号 = 课程信息.课程编号 ) and

( 选课表.课序号 = 课程表.课序号 ) and

( 选课表.学号 = :ls\_xh ) and

( 课程表.课程编号 = :li\_kcbh ) ;

if li\_count > 0 then

messagebox("提示","已经选择该课程！")

return

end if

//判断选择课程时间跟已经选择时间是否重复

//创建数据存储

lds\_kxh = create datastore

lds\_kxh.dataobject = 'd\_kxh\_store'

//连接到数据库

lds\_kxh.settransobject(sqlca)

//检索数据

lds\_kxh.retrieve(ls\_kxh)

//循环，判断时间是否重复

### 3.3.9学生成绩查询（打印）模块流程设计

功能：学生成功登录后，如果是在籍的学生（未毕业），可以对成绩信息进行检索，查询相关的成绩，并可以自己进行打印。

**** 学生成绩查询（打印）模块程图3-17所示：



图3-17 学生成绩查询（打印）模块流程图

### 3.3.10成绩单打印模块界面设计

1. 功能概述：学生和教师可以对学生的成绩进行查询（学生只能打印本人的成绩），并可以进行打印可对课程表信息查询，了解上课的时间及地点。
2. 界面图设计(如图3-18所示)



图3-18成绩打印界面

1. 主要代码

string ls\_xh,ls\_xm,ls\_xjbh

string ls\_kcmc,ls\_skdd,ls\_str

integer li\_nj,li\_tian,li\_Jie

ls\_xh = trim(sle\_xh.text)

//得到学生信息

SELECT 学生信息.姓名,

学生信息.年级,

学生信息.学籍编号

INTO :ls\_xm,

:li\_nj,

:ls\_xjbh

FROM 学生信息

WHERE 学号 = :ls\_xh;

//显示姓名，年级

sle\_xm.text = ls\_xm

sle\_nj.text = string(li\_nj)

//检索成绩

dw\_1.retrieve(ls\_xh)

### 3.3.11系统主窗体的设计

* + 本系统采菜单和界面的双重显示方式，方便给不同操作习惯的用户可以有不同的点击方式，同时以体现窗口操作的规范性和统一性。
  + 各管理模块相互独立，不同的用户登录后有不同的权限和相对应的操作界面，符合系统安全性设计的要求与考量。
  + 主要采用在窗口中调用数据窗口，以及在窗口中添加各类控件的方法，来实现各功能模块功能的实现与操作。

1. 系统主窗体界面功能概述：本应用程序的主要功能都是在主窗口w\_main及在其下打开的众多窗口中完成的。不同用户登录后，由于级别不同，主界面也会有些差异，不能操作的菜单项为灰色。
2. 界面图设计
3. 学生登录后的界面：

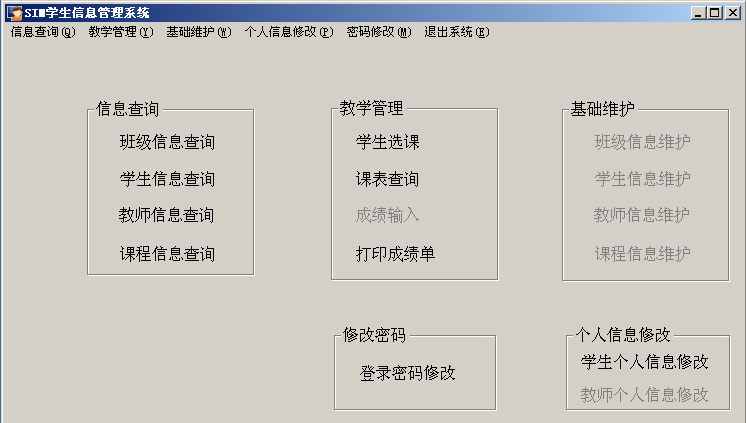


图3-19 学生登录界面

1. 教师登录后的界面

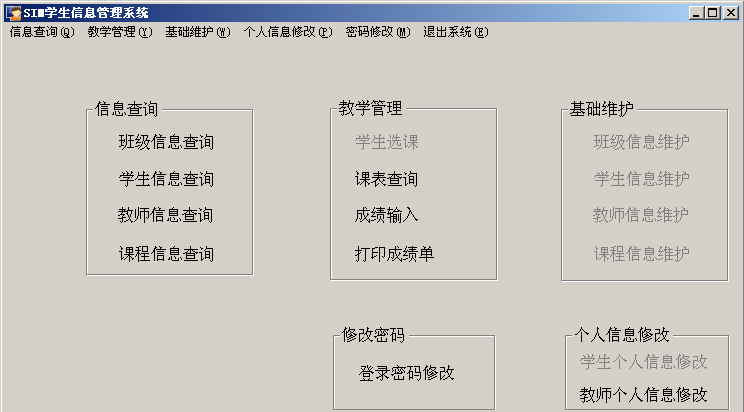


图3-20 教师登录界面

1. 管理员登录后的界面

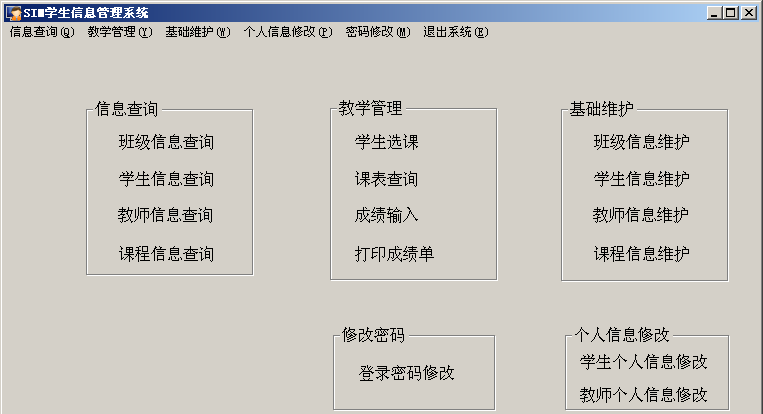


图3-21 管理员登录界面

1. 重要代码

信息查询模块

* + - 班级信息查询：

Clicked事件程序代码opensheet(w\_bjxx\_cx,w\_main,1, Original!)

* + - 学生信息查询：

Clicked事件程序代码opensheet(w\_xsxx\_cx,w\_main,1, Original!)

* + - 教师信息查询：

Clicked事件程序代码opensheet(w\_jsxx\_cx,w\_main,1, Original!)

* + - 课程信息查询：

Clicked事件程序代码opensheet(w\_kcxx\_cx,w\_main,1, Original!)

信息查询模块

* + - 学生选课：

Clicked事件程序代码opensheet(w\_xsxk,w\_main,1, Original!)

* + - 课表查询：

Clicked事件程序代码opensheet(w\_kbcx,w\_main,1, Original!)

* + - 成绩输入：

Clicked事件程序代码opensheet(w\_cjsr,w\_main,1, Original!)

* + - 打印成绩单：

Clicked事件程序代码opensheet(w\_dycjd,w\_main,1, Original!)

基础维护模块

* + - 班级信息维护：

Clicked事件程序代码opensheet(w\_bjxx,w\_main,1, Original!)

* + - 学生信息维护：

Clicked事件程序代码opensheet(w\_xsxx,w\_main,1, Original!)

* + - 教师信息维护：

Clicked事件程序代码opensheet(w\_jsxx,w\_main,1, Original!)

* + - 课程信息维护：

Clicked事件程序代码opensheet(w\_kcxx,w\_main,1, Original!)

### 3.3.12信息查询模块的设计

1. 功能概述：学生、教师或管事登录后，可以对学生、班级、课程、老师信息进行检索和查询操作。

界面图设计(如图3-22、23、24、25所示)



图3-22班级信息查询界面

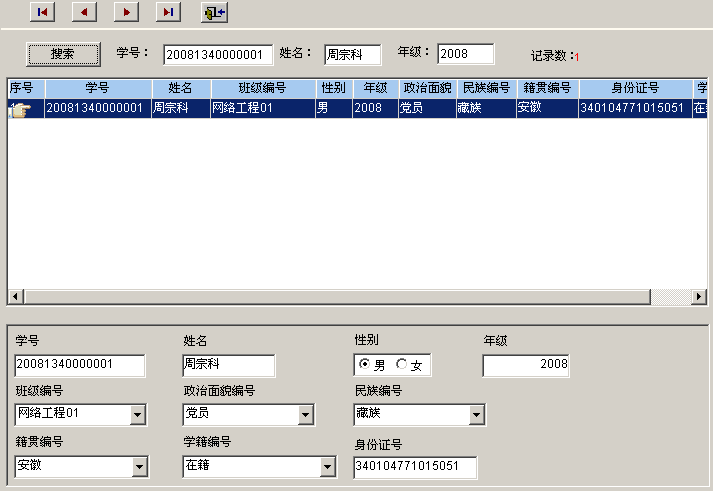


图3-23学生信息查询界面



图3-24教师信息查询界面

图3-25课程信息查询界面



3、重要代码

* 代码：

//没有记录则返回

if il\_rowcount < 1 then return

il\_row = 1 //当前行为1

//取消选择

dw\_1.selectrow(0,false)

//选择当前行

dw\_1.selectrow(il\_row,true)

//移到当前行

dw\_1.scrolltorow(il\_row)

//设置按钮状态

pb\_first.enabled = false

pb\_prior.enabled = false

pb\_next.enabled = true

pb\_buttom.enabled = true

//调用自定义事件

parent.event ue\_search()

* 代码：

parent.event ue\_search()//没有数据则返回

if il\_rowcount < 1 then return

//当前记录为第一行则返回

if il\_row <= 1 then return

il\_row -= 1 //记录减一

//取消选择

dw\_1.selectrow(0,false)

//选择当前行

dw\_1.selectrow(il\_row,true)

//移动到第一行

dw\_1.scrolltorow(il\_row)

//控制按钮状态

if il\_row = 1 then

pb\_first.enabled = false

pb\_prior.enabled = false

end if

pb\_next.enabled = true

pb\_buttom.enabled = true

//调用窗口自定义事件

parent.event ue\_search()

### 3.3.13课表查询模块设计

1. 功能概述：学生可对课程表信息查询，了解上课的时间及地点。
2. 界面图设计(如图3-26所示)



图3-26 学生选课界面

1. 主要代码

integer li\_row,li\_bh

string ls\_xh,ls\_xm,ls\_xjbh

string ls\_kcmc,ls\_skdd,ls\_str

integer li\_nj,li\_tian,li\_Jie,i

ls\_xh = trim(sle\_xh.text)

dw\_1.reset()

//插入课时节

for i = 1 to 10

dw\_1.insertrow(i)

dw\_1.setitem(i,"jieshu",i)

next

//得到 姓名，年级,学籍编号

SELECT 学生信息.姓名,

学生信息.年级,

学生信息.学籍编号

INTO :ls\_xm,

:li\_nj,

:ls\_xjbh

FROM 学生信息

WHERE 学号 = :ls\_xh;

//提示不是在籍学生

if ls\_xjbh <> '1' then

messagebox("提示","不是在籍的学生！")

sle\_xh.text = ""

return

end if

//显示姓名，年级

sle\_xm.text = ls\_xm

sle\_nj.text = string(li\_nj)

declare lc\_kc Cursor For

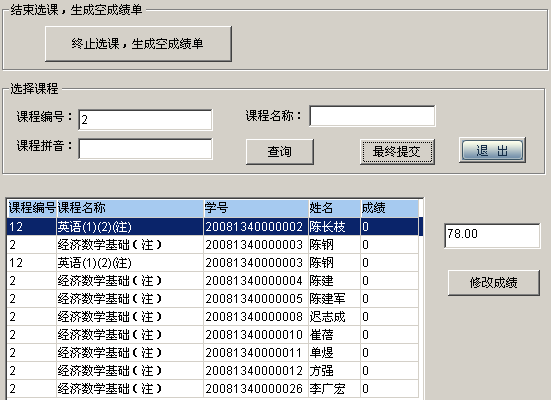
SELECT 课程信息.课程名称,

课程表.上课地点,

### 3.3.14成绩输入模块设计

1. 功能概述：教师登录后，可以对他所任的课程进行查询，查看就有那些学生选学，在考试结束后，可以终止学生选课，进行课和成绩的输入。
2. 界面图设计(如图3-27所示)

图3-27成绩输入界面



1. 主要代码

integer li\_row,li\_bh

decimal ld\_cj

li\_row = dw\_1.getselectedrow(0)

if li\_row < 1 then return

//得到编号

li\_bh = dw\_1.getitemnumber(li\_row,"编号")

ld\_cj = dec(em\_1.text)

//更新成绩

update 成绩表

set 成绩 = :ld\_cj

where 编号 = :li\_bh ;

//提交

if sqlca.sqlcode = 0 then

commit ;

parent.triggerevent('ue\_search')

else

//回滚

rollback;

messagebox("提示","修改成绩失败！")

integer li\_bh,li\_rtn,li\_row

integer li\_kcbh

li\_row = dw\_1.getselectedrow(0)

if li\_row < 1 then

messagebox("提示","请先选择需要提交成绩的记录！")

return

end if

li\_kcbh = dw\_1.getitemnumber(li\_row,"课程编号")

li\_rtn = messagebox("提示","成绩提交后不能修改，确实需要提交吗？",question!,okcancel!,2)

if li\_rtn = 2 then return

//定义存储过程

DECLARE sp\_cjtj PROCEDURE FOR sf\_成绩提交

(:li\_kcbh);

EXECUTE sp\_cjtj; //执行数据存储

parent.triggerevent('ue\_search')

messagebox("提示","提交成绩成功！")

# 第四章 系统测试

## 4.1软件测试概述

软件测试就是利用测试工具按照测试方案和流程对产品进行功能和性能测试，甚至根据需要编写不同的测试工具，设计和维护测试系统，对测试方案可能出现的问题进行分析和评估。执行测试用例后，需要跟踪故障，以确保开发的产品适合需求。

## 4.2软件测试的目标

软件测试是指使用人工或者自动的手段来运行或测定某个软件产品系统的过程，其目的是在于检验是否满足规定的需求或者弄清预期的结果与实际结果的区别。本文主要描述软件测试的类型。

* 数据和数据库完整性测试
* 数据与数据库完整测试是指测试关系型数据库完整性原则以及数据合理性测试。
* 数据库完整性原即：
  + 主码完整性：主码不能为空；
  + 外码完整性：外码必须等于对应的主码或者为空。
  + 数据合理性指数据在数据库中的类型，长度，索引等是否建的比较合理。

## 4.3软件测试的方法

软件测试方法一般分为动态测试方法与静态测试方法。动态测试方法中又根据测试用例的设计方法不同，分为黑盒测试与白盒测试两类。

### 4.3.1 静态测试

静态测试是采用人工检测盒计算机辅助静态分析的手段对程序进行检测，方法如下：

1. 人工检测：是指不依靠计算机运行程序，而靠人工审查盒评审软件。
2. 计算机辅助静态分析：指利用静态分析软件工具对被测试程序进行特性分析，从程序中提取一些信息，来检查可能出错的变量、参数与代码等。

### 4.3.2 动态测试

动态测试与静态测试相反，只要是设计一组输入数据，然后通过运行程序来发现错误。一般有两种方法：黒盒测试和白盒测试。

### 4.3.3 黑盒测试与白盒测试

1、黑盒测试

黑盒测试也称功能测试或数据驱动测试，它是在已知产品所应具有的功能，通过测试来检测每个功能是否都能正常使用。在测试时，把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，测试者在程序接口进行测试，只检查程序功能是否按照需求规格说明书的规定正常使用，程序是否能适当接收输入数据而产生正确的输出信息，并且保持外部信息的完整性。

2、白盒测试

白盒测试是基于代码的测试，测试人员通过阅读程序代码或者通过使用开发工具中的单步调试来判断软件的质量，一般黑盒测试由项目经理在程序员开发中来实现。白盒测试分为动态白盒测试和静态白盒测试

* 2.1 静态白盒测试

利用眼睛，浏览代码，凭借经验，找出代码中的错误或者代码中不符合书写规范的地方。比如，代码规范中规定，函数必须为动宾结构。

* 2.2 动态白盒测试

利用开发工具中的调式工具进行测试。比如一段代码有4个分支，输入4组不同的测试数据使4组分支都可以走通而且结果必须正确。

## 4.4主要功能模块测试

该测试计划主要包括对软件各个主要功能模块测试。主要有6大模块测试（登录模块+主界面），如下图4-1所示

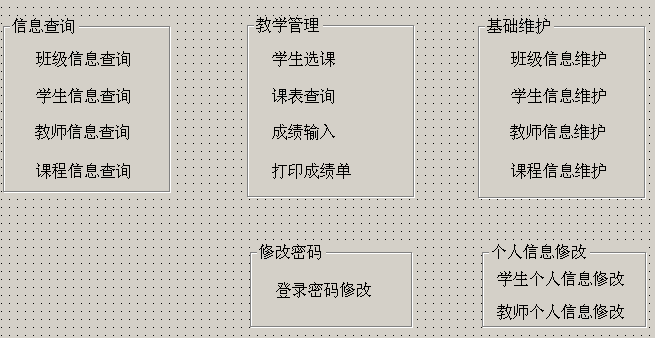


图4-1 主要功能模块

### 4.4.1登陆管理模块的测试

该测试的目的是保证用户的登录类型与用户姓名及密码三者完全匹配，才可以验证通过，系统默认是以学生的角色来登录的，但用户也可以选择其它用户类型来登录。

当用户只选择用户类型，不输入帐户和密码时能给出正确的错误提示。

当输入信息错误时，则显示如图4-2所示：



图4-2 输入错误信息

当输入正确信息时，则显示如图4-4所示：

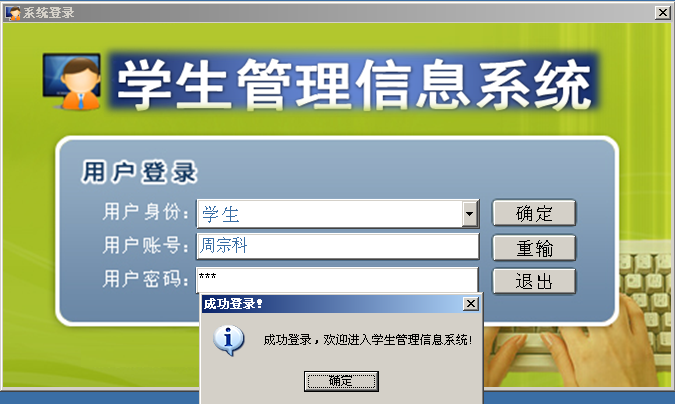


图4-3 输入信息有效

### 4.4.2密码变更模块的测试

当用户成功登录时，点击“登录密码修改”则进入如下界面，如图4-4所示：



图4-4 登录密码修改界面

当输入帐户点击提交搜索时，则显示界面如图4-5所示：

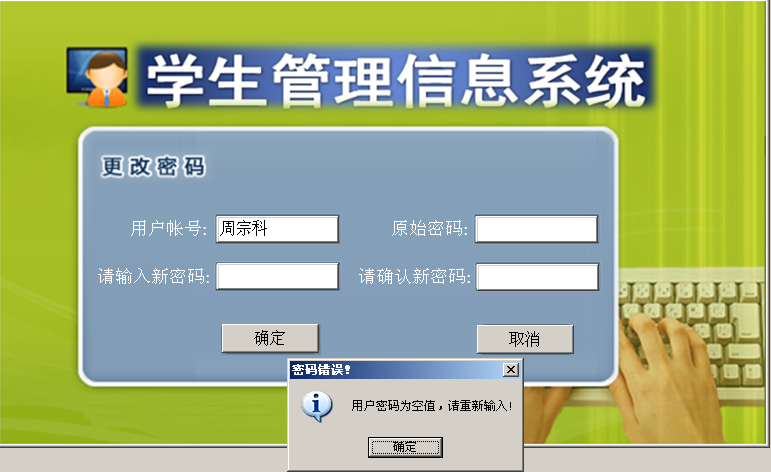


图4-5 更改密码错误提示界面

当输入正确的帐户、原始密码和新密码点击提交搜索时，则显示界面则提示“密码修改成功”如图4-6所示：

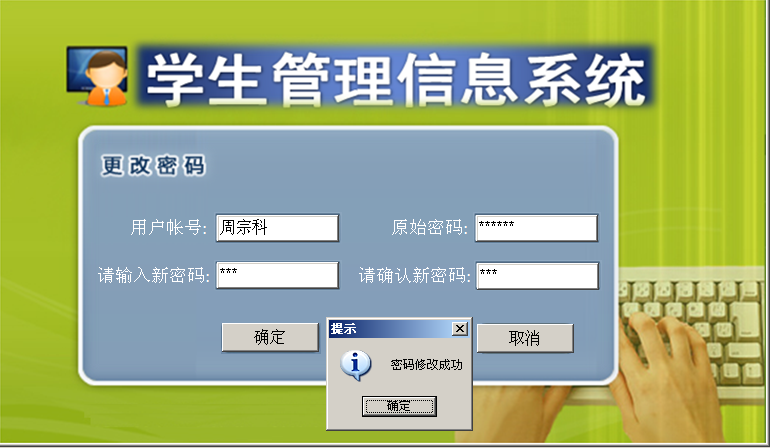


图4-6 密码修改成功提示界面

### 4.4.3信息查询模块的测试

班级信息的查询:如果不输入任何过滤条件，则默认搜索出所有的相关记录。

如图4-7所示

图4-7班级信息的查询结果



当输入相关的滤条件时，则会只显示与之匹配的记录如图4-8所示：

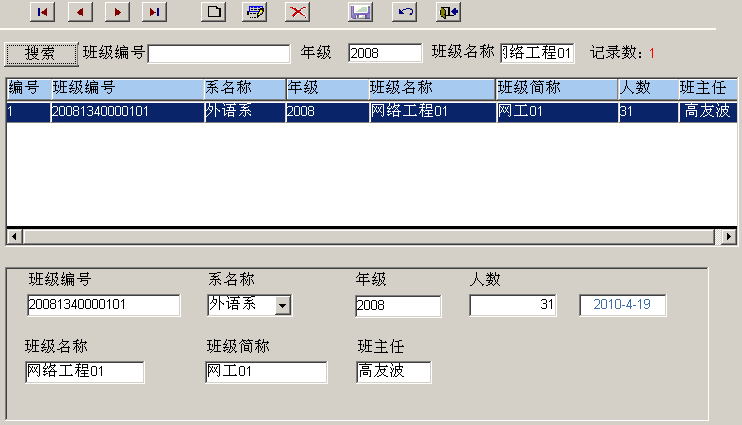


图4-8 班级信息(过滤条件时)搜索结果

学生信息的查询:如果不输入任何过滤条件，则默认搜索出所有的相关记录。

如图4-9所示

图4-9学生信息的查询结果



当输入相关的滤条件时，则会只显示与之匹配的记录如图4-10所示：

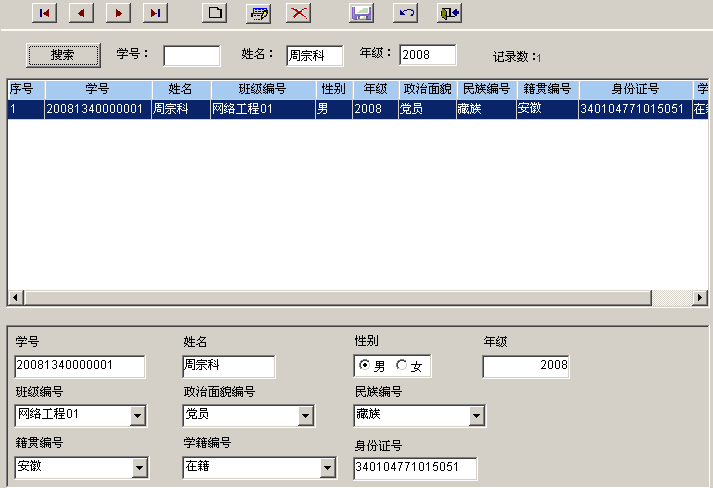


图4-10 学生信息(过滤条件时)搜索结果

教师信息的查询:如果不输入任何过滤条件，则默认搜索出所有的相关记录。

如图4-11所示

图4-11教师信息的查询结果



当输入相关的滤条件，例如查看“治金系”有那些教师，则会只显示与之匹配的记录如图4-12所示：

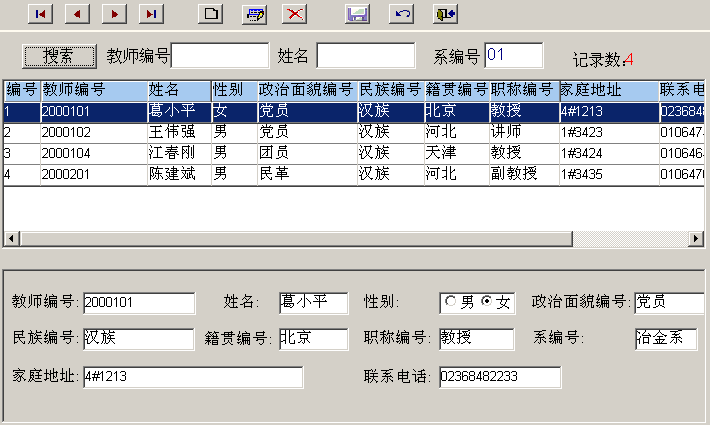


图4-12 教师信息(过滤条件时)搜索结果

### 4.4.4信息管理模块的测试

学生选课:先输入学号，然后查询相关的课程并进行选课，如果已经选或上课时间冲突，则给出信息提示。如图4-13所示

图4-13选课冲突界面



先输入学号，然后查询相关的课程并进行选课，如果已经选或上课时间不冲突，则给出信息提示：保存成功（选课成功）。

如图4-14所示

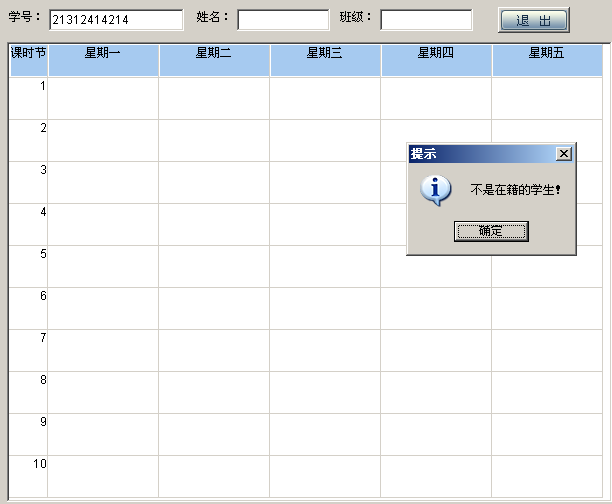


图4-14选课成功界面

课表查询:如果输入的是错误的学号，则提示相应的搜索结果提示信息。

如图4-15所示

图4-15课表查询结果提示信息



如果输入的是正确的学号，则提自动显示课表信息。

如图4-16所示



图4-16 学生信息(过滤条件时)搜索结果

教师-成绩输入查询:教师登录后，可以对他所任的课程进行查询，查看就有那些学生选学，在考试结束后，可以终止学生选课，进行课和成绩的输入。

如图4-17所示

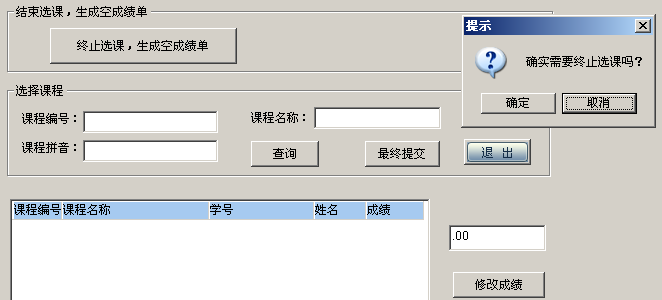


图4-17终止选课

进行课程信息查询，然后输入成绩。如图4-18~19所示：



图4-18 成绩输入

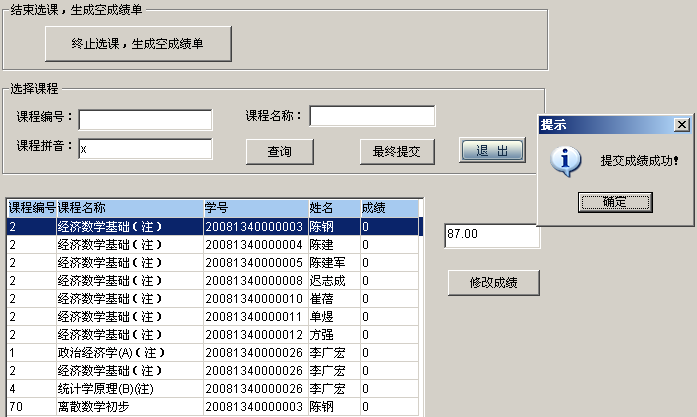


图4-19 成绩提交

成绩查询:输入正确的学号，进行成绩单的查询。

如图4-20所示

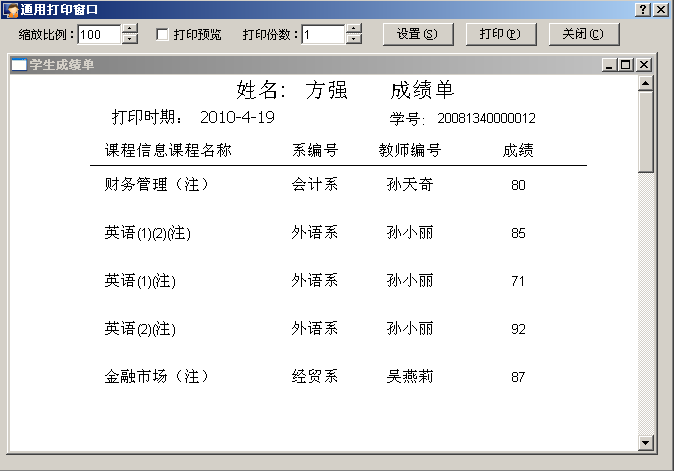
图4-15课表查询结果提示信息



成绩查询:输入正确的学号，进行成绩单的查询，查询到结果后选择“打印”，可以选择要打印的份数，放大的比例，还里以更改打印的相关设置及打印预览。下图所示为为“成绩单的打印”窗口。

如图4-21所示

图4-21成绩单打印



### 4.4.5个人信息修改模块的测试

学生个人信息修改：学生登录后，可以对其他学生信息进行查询和检索，但不能修改其他学生的信息，只能对登录本人“学生个人信息修改”。如图4-22所示：



图4-22学生个人信息修改界面

此时，对学生个人的信息进行修改，例如：修改周宗科本人的“籍贯编号”由“安徽”改为“北京”；“民族编号”由“藏族”改为“汉族”，并提交。则显示如图4-23所示



图4-23学生个人的信息进行修改

教师的个人信息修改状态操作与此类似，再次不在做累赘阐述。

### 4.4.6基础维护模块的测试

管理员登录后，可以对学生信息、班级信息、课程信息、教师信息进行维护（添加／修改／删除）。

此处测试以“学生信息维护”为例，界面图设计(如图4-24、25、26所示)



图4-24学生信息添加



图4-25学生信息保存



图4-26学生信息删除提示界面



图4-27学生信息删除

其他维护模块：学生信息、班级信息、课程信息维护（添加／修改／删除）。

测试与此类似，在此不作赘述。

至此，对所有学生信息管理系统的主要功能模块进行了测试，并输入相关的数据和相应的功能测试操作，基本功能都已实现，系统达到预期目标！！

# 结论

本《学生管理信息系统》运用软件工程的概念进行系统分析和设计，应用数据库原理的基本知识进行概念结构设计，以关系数据库SQL2005 为后台数据库，采用目前版本较新的PB11.0数据库开发软件，完成了系统的设计和可执行程序的生成与应用程序的分发。在测试阶段时采用黑盒测试的方法，对各个功能实现进行了测试。

整个研究过程经过需求分析、系统设计、系统开发、系统运行和测试五个阶段的工作, 开发出来的软件具有良好的用户界面，主要实现了以下几个功能模块：

* + 1. 实现学生、教师以及统管理员的登录。
    2. 实现学生对个人信息的维护。
    3. 实现学生对课程的选择与成绩的查看。
    4. 实现教师对课程的维护与成绩的录入。
    5. 实现学生对个人信息、班级信息、教师信息、课程信息进行检索。
    6. 实现系统管理员对学生个人信息、班级信息、教师信息、课程信息的检索与维护。

本系统解决了学生信息管理过程中学生信息管理、成绩管理、课程管理等时效性相对低问题，提高了教务工作的效率，达到了系统设计的预期目标。

本系统是基于C/S 结构，为更好的适应发展趋势，今后将继续完善各方面的功能，把握用户的更详尽的需求，不断满足学生管理信息系统的新需求。

# 致 谢

在论文的设计和开发过程中，钟光太导师给予了我细心的指导与帮助。在开始阶段，钟老师给我指出了正确的程序设计方法与步聚，为我毕业设计的顺利进行指明方向，使我少走了不少弯路。在后期阶段，钟老师又不断地提出很好的改进意见，使得我的学生管理信息系统日趋完善。从钟老师身上，我体会到了对那种对科学与实事一丝不苟严谨教学精神与理念，以及对待开发项目的高标准要求，为我今后参加实际的项目开发打下了坚实的基础。同时，也感谢宋晔给我们上的PB课，使我在这次的设计当中能够很快的上手，总之，钟光太老师和宋晔老师在这次的程序设计过程中给予的指导与启迪，使我受益匪浅。

感谢所有教育过和帮助过我的老师们，你们的教导是我一生中最宝贵的财富。感谢我所有的朋友和同学，一起学习、一起奋斗的美好时光里，你们给予我的真诚的鼓励和无私的帮助令我终身难忘。

在此再一次衷心地谢谢钟光太老师对我的辛勤培养和指导！！！

# 参考文献

[1] 杨斌 著 信息管理系统中数据分布和传输机制研究与实现[D].北京：北京交通大学.硕士论文,2008.南区学位论文阅览室( 1/ 0)

[2]乔好勤．信息管理学研究方法导论［M］．北京：北京图书馆出版社 2007，2007

[3]程学先，宋克振．管理信息系统及其开发［M］．北京：清华大学出版社，2008

[4]张瑞军, 徐丽娜等编著．信息系统开发工具［M］．北京：清华大学出版社，2007

[5] 壮志剑. 数据库应用实训教程[M]. 西安:交通大学出版社, 2006年6月

[6] 汪斌 钱梅. PowerBuilder7与SQL Server [M]. 北京：电子工业出版社, 2002

[7] 张英杰.PowerBuilde应用工程软件开发指南[M].西安:交通大学出版社,2005年

[8] 郑阿奇. PowerBuilder 实用教程（第3版）[M]. 北京：电子工业出版社，2009

[9] 陈永强. PowerBuilder编程技术全接触[M]. 北京：清华大学出版社，2007年

[10] Paul Nielsen．SQL Server 2005 Bible [M].Indianapolis，IN：Sams Pub，2006．

# 附录

## 使用说明书

本系统适合学院的教务管理与学生信息的管理工作。通过该系统可以对学生个人信息、班级信息、教师信息、课程信息进行检索，课程选择与成绩打印；教师通过该系统可以对教师个人信息、班级信息、学生信息、课程信息进行检索及课程成绩的录入；学生管理信息系统管理员通过该系统可以实现对以上信息进行检索与维护。

**学生与教师登录使用**

1.1信息查询

此界面显示班级信息、学生信息、教师信息及课程信息的查询。

1.2教学管理

使用方法：学生登录后可以输入学号进行选课、课表查询和成绩的打印。教师登录后可以对课程进行查询，并对选学这门的学生输入成绩。

1.3个人信息修改

使用方法：学生个人信息修改：学生登录后，可以对其他学生信息进行查询和检索，但不能修改其他学生的信息，只能对登录本人“学生个人信息修改”。

教师个人信息修改：教师登录后，可以对其他教师信息进行查询和检索，但不能修改其他教师的信息，只能对登录本人“教师个人信息修改”。

1.4密码更改

使用方法：学生、教师、管理员成功登录后，可以对自己的密码进行更改。

**管理员登录使用**

1.1系统界面：管理员登录

可以实现以上所有功能模块的操作，并可以对班级、学生、教师、课程信息进行添加、修改、删除等维护工作。

## 主要代码

* 登录窗口重要代码
  + “确定”按钮clicked()

string password,account,password2

int reinput

password=trim(sle\_2.text) //trim去除空格

if password='' then

messagebox('密码错误！','管理员的密码为空值，请重新输入!')

sle\_2.setfocus()

else

account=trim (sle\_1.text)

if lb\_type.text = '管理员' then

select 密码

into:password2

from 管理员信息

where 姓名=:account;

elseif lb\_type.text = '学生' then

select 密码

into:password2

from 学生信息

where 姓名=:account;

elseif lb\_type.text = '教师' then

select 密码

into:password2

from 教师信息

where 姓名=:account;

end if // if lb\_type.text = '管理员' then

if password2='' then

messagebox('账号错误','此帐号不存在或密码错误，请重新输入')

sle\_1.setfocus()

sle\_1.text=''

sle\_2.text=''

elseif password=password2 then

g\_account = account

g\_accountType = lb\_type.text

if lb\_type.text='管理员' then

open(w\_main)

close(w\_login)

elseif lb\_type.text='教师' then

open(w\_main)

m\_main.m\_教学管理.m\_学生选课.enabled = false

m\_main.m\_基础维护.m\_班级信息维护.enabled = false

m\_main.m\_基础维护.m\_教师信息维护.enabled = false

m\_main.m\_基础维护.m\_学生信息维护.enabled = false

m\_main.m\_基础维护.m\_课程信息维护.enabled = false

m\_main.m\_个人信息修改.m\_学生个人信息修改.enabled = false

w\_main.st\_xsxk.enabled=false

w\_main.st\_xsxk.TextColor= rgb(128, 128, 128)

w\_main.st\_xsxx\_pr.enabled=false

w\_main.st\_xsxx\_pr.TextColor= rgb(128, 128, 128)

w\_main.st\_bjxx.enabled=false

w\_main.st\_bjxx.TextColor= rgb(128, 128, 128)

w\_main.st\_xsxx.enabled=false

w\_main.st\_xsxx.TextColor= rgb(128, 128, 128)

w\_main.st\_jsxx.enabled=false

w\_main.st\_jsxx.TextColor= rgb(128, 128, 128)

w\_main.st\_kcxx.enabled=false

w\_main.st\_kcxx.TextColor= rgb(128, 128, 128)

close(w\_login)

elseif lb\_type.text='学生' then

open(w\_main)

m\_main.m\_教学管理.m\_成绩输入.enabled = false

m\_main.m\_基础维护.m\_班级信息维护.enabled = false

m\_main.m\_基础维护.m\_教师信息维护.enabled = false

m\_main.m\_基础维护.m\_学生信息维护.enabled = false

m\_main.m\_基础维护.m\_课程信息维护.enabled = false

m\_main.m\_个人信息修改.m\_教师个人信息修改.enabled = false

w\_main. st\_cjsr.enabled=false

w\_main. st\_cjsr.TextColor= rgb(128, 128, 128)

w\_main.st\_jsxx\_pr.enabled=false

w\_main.st\_jsxx\_pr.TextColor= rgb(128, 128, 128)

w\_main.st\_bjxx.enabled=false

w\_main.st\_bjxx.TextColor= rgb(128, 128, 128)

w\_main.st\_xsxx.enabled=false

w\_main.st\_xsxx.TextColor= rgb(128, 128, 128)

w\_main.st\_jsxx.enabled=false

w\_main.st\_jsxx.TextColor= rgb(128, 128, 128)

w\_main.st\_kcxx.enabled=false

w\_main.st\_kcxx.TextColor= rgb(128, 128, 128)

close(w\_login)

end if // if lb\_type.text='管理员' then

else

messagebox('账号错误','此帐号不存在或密码错误，请重新输入')

sle\_1.text=''

sle\_2.text = ''

sle\_1.setfocus( )

end if // if password=password2 then

end if // if password=''then

* 信息查询模块主要代码
* 代码：

//没有记录则返回

if il\_rowcount < 1 then return

il\_row = 1 //当前行为1

//取消选择

dw\_1.selectrow(0,false)

//选择当前行

dw\_1.selectrow(il\_row,true)

//移到当前行

dw\_1.scrolltorow(il\_row)

//设置按钮状态

pb\_first.enabled = false

pb\_prior.enabled = false

pb\_next.enabled = true

pb\_buttom.enabled = true

//调用自定义事件

parent.event ue\_search()

* 代码：

parent.event ue\_search()//没有数据则返回

if il\_rowcount < 1 then return

//当前记录为第一行则返回

if il\_row <= 1 then return

il\_row -= 1 //记录减一

//取消选择

dw\_1.selectrow(0,false)

//选择当前行

dw\_1.selectrow(il\_row,true)

//移动到第一行

dw\_1.scrolltorow(il\_row)

//控制按钮状态

if il\_row = 1 then

pb\_first.enabled = false

pb\_prior.enabled = false

end if

pb\_next.enabled = true

pb\_buttom.enabled = true

//调用窗口自定义事件

parent.event ue\_search()

代码：



//记录为最后行则返回

if il\_rowcount = il\_row then return

il\_row += 1 //记录加一

//取消选择

dw\_1.selectrow(0,false)

//选择当前行

dw\_1.selectrow(il\_row,true)

//移动到当前行

dw\_1.scrolltorow(il\_row)

//控制按钮状态

if il\_row = il\_rowcount then

pb\_next.enabled = false

pb\_buttom.enabled = false

end if

if il\_rowcount > 1 then

pb\_first.enabled = true

pb\_prior.enabled = true

end if

//调用窗口自定义事件

parent.event ue\_search()

代码：



//记录为最后行则返回

if il\_rowcount = il\_row then return

il\_row = il\_rowcount //当前记录为最后行

//取消选择

dw\_1.selectrow(0,false)

//选择当前行

dw\_1.selectrow(il\_row,true)

//移动到最后行

dw\_1.scrolltorow(il\_row)

//控制按钮状态

pb\_next.enabled = false

pb\_buttom.enabled = false

if il\_rowcount > 1 then

pb\_first.enabled = true

pb\_prior.enabled = true

end if

//调用窗口自定义事件

parent.event ue\_search()

* 代码：close(parent)



string ls\_select,ls\_bjbh,ls\_nj,ls\_njmc

ls\_bjbh = trim(sle\_bjbh.text) // 拼音码

ls\_nj = trim(sle\_nj.text) // 编号

ls\_njmc = trim(sle\_njmc.text) // 名称

ls\_select = ""

// 班级编号

if not isnull(ls\_bjbh) and ls\_bjbh <> "" then

ls\_bjbh = "%" + ls\_bjbh + "%"

ls\_select = ls\_select + " 班级编号 LIKE '" + ls\_bjbh + "'"

end if

// 年级

if not isnull(ls\_nj) and ls\_nj <> "" then

ls\_nj = "%" + ls\_nj + "%"

if ls\_select = "" then

ls\_select = " 年级 LIKE '" + ls\_nj + "'"

else

ls\_select = ls\_select + " AND 年级 LIKE '" + ls\_nj + "'"

end if

end if

// 班级名称

if not isnull(ls\_njmc) and ls\_njmc <> "" then

ls\_njmc = "%" + ls\_njmc + "%"

if ls\_select = "" then

ls\_select = " 班级名称 LIKE '" + ls\_njmc + "'"

else

ls\_select = ls\_select + " AND 班级名称 LIKE '" + ls\_njmc + "'"

end if

end if

//获得SQL语法

if ls\_select <> "" then

ls\_select = is\_sql + " where " + ls\_select

else

ls\_select = is\_sql

end if

//重新连接数据库

dw\_1.SetSQLSelect(ls\_select)

il\_rowcount = dw\_1.retrieve() //重新检索数据

sle\_record.text = string(il\_rowcount) //显示记录数

* 学生选课模块主要代码

代码：



integer li\_row,li\_insertrow,li\_count

string ls\_kxh,ls\_kcmc,ls\_jsbh,ls\_xbh,ls\_dd,ls\_xh,ls\_oldkxh

integer li\_sjt,li\_sjj,li\_kcbh,i

datastore lds\_kxh

li\_row = dw\_2.getselectedrow(0)

if li\_row < 1 then return

ls\_xh = trim(sle\_xh.text)

if isnull(ls\_xh) or ls\_xh = "" then

messagebox("提示","请先输入学号！")

return

end if

li\_kcbh = dw\_2.getitemnumber(li\_row,"课程编号")

ls\_kxh = dw\_2.getitemstring(li\_row,"课序号")

//判断是否已经选择该课程

SELECT count(\*)

INTO :li\_count

FROM 选课表,

课程信息,

课程表

WHERE ( 课程表.课程编号 = 课程信息.课程编号 ) and

( 选课表.课序号 = 课程表.课序号 ) and

( 选课表.学号 = :ls\_xh ) and

( 课程表.课程编号 = :li\_kcbh ) ;

if li\_count > 0 then

messagebox("提示","已经选择该课程！")

return

end if

//判断选择课程时间跟已经选择时间是否重复

//创建数据存储

lds\_kxh = create datastore

lds\_kxh.dataobject = 'd\_kxh\_store'

//连接到数据库

lds\_kxh.settransobject(sqlca)

//检索数据

lds\_kxh.retrieve(ls\_kxh)

//循环，判断时间是否重复

for i = 1 to lds\_kxh.rowcount()

li\_sjt = lds\_kxh.getitemnumber(i,"上课时间天")

li\_sjj = lds\_kxh.getitemnumber(i,"上课时间节")

select count(\*)

into :li\_count

from 课程表,选课表

where 课程表.课序号 = 选课表.课序号

and 课程表.上课时间天 = :li\_sjt

and 课程表.上课时间节 = :li\_sjj

and 选课表.学号 = :ls\_xh;

if li\_count > 0 then

messagebox("提示","选择的时间跟已选时间重叠！")

destroy lds\_kxh ;

return

end if

next

//破坏数据存储

destroy lds\_kxh ;

//得到数据窗中选择的行，并得到需要的值

ls\_kxh = dw\_2.getitemstring(li\_row,"课序号")

ls\_kcmc = dw\_2.getitemstring(li\_row,"课程名称")

ls\_jsbh = dw\_2.getitemstring(li\_row,"教师编号")

li\_kcbh = dw\_2.getitemnumber(li\_row,"课程编号")

ls\_xbh = dw\_2.getitemstring(li\_row,"系编号")

//定义游标，取得课程表信息

DECLARE ld\_ke CURSOR FOR

SELECT 上课时间天 ,

上课时间节 ,

上课地点

FROM 课程表

WHERE 课序号 = :ls\_kxh;

//打开游标

open ld\_ke ;

//取得课程的上课时间等信息

do while sqlca.sqlcode = 0

fetch ld\_ke into :li\_sjt,:li\_sjj,: ls\_dd ;

if sqlca.sqlcode = 0 then

//赋值数据窗dw\_1

li\_insertrow = dw\_1.insertrow(0)

//dw\_1.setitem(li\_insertrow,"课程编号",li\_kcbh)

dw\_1.setitem(li\_insertrow,"学号",ls\_xh)

dw\_1.setitem(li\_insertrow,"课序号",ls\_kxh)

dw\_1.setitem(li\_insertrow,"课程名称",ls\_kcmc)

dw\_1.setitem(li\_insertrow,"教师编号",ls\_jsbh)

dw\_1.setitem(li\_insertrow,"系编号",ls\_xbh)

dw\_1.setitem(li\_insertrow,"上课地点",ls\_dd)

dw\_1.setitem(li\_insertrow,"上课时间天",li\_sjt)

dw\_1.setitem(li\_insertrow,"上课时间节",li\_sjj)

end if

loop

close ld\_ke ;

commit ;

//插入选修课程

INSERT INTO 选课表

( 学号,

课序号 )

VALUES ( :ls\_xh,

:ls\_kxh )

;

//插入失败则回滚

if sqlca.sqlcode <> 0 then

rollback ;

messagebox("提示","选课失败！")

return

else

commit ;

messagebox("提示","保存成功！")

end if