|  |  |
| --- | --- |
|  |  |
| **学 号：** | 0121708910310 |

****

|  |  |
| --- | --- |
| **题 目** | 教学管理信息系统--学生选课及课程安排数据库综合实验 |
| **学 院** | 计算机科学与技术学院 |
| **专 业** | 计算机科学与技术专业 |
| **班 级** | 计算机1701班 |
| **姓 名** | 张超杰 |
| **指导教师** | 施继珉 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 2019 | 年 | 12 | 月 | 30 | 日 |

# 原创声明

本人郑重声明：

本人所呈交的有关项目成果，是本人在老师的指导下独立进行开发所取得的成果。除实验报告书中已经注明的内容外，本实验项目不包含任何其他个人或集体已经提交、发表的成果。对本次实验项目做出重要贡献的个人和集体，均已在实验报告书中以明确方式标明。本人完全意识到本声明的法律责任由本人承担。

本人签名：

# 1教学管理信息系统之需求分析

## 1.1需求分析的任务

需求分析是软件定义时期的最后一个阶段，它的基本任务是回答“系统必须做什么”这个问题。即根据用户的需要提取有效信息，进行调查分析，并且选择合理的方法进行建模分析，并出具相应的软件需求规格说明书。

## 1.2任务描述

本次软件需求分析的主要任务是对教学管理信息系统的学生选课模块进行需求分析，明确教务管理信息系统选课模块的功能要求以及性能要求，明确教学管理信息系统选课模块的开发进度要求，并且出具教学管理信息系统的软件需求规格说明书。为后续开发人员提供开发依据及要求，起着重要的指导作用，最大程度的提高开发效率。

## 1.3系统项目概述

### 1.3.1问题定义

项目名称：教学管理信息系统-选课系统

用户：学生，老师，教务管理人员等

开发人员：武汉理工大学计算机1701班张超杰

### 1.3.2系统必要性和意义

必要性：教学信息是一个很庞大的数据，仅靠人工管理，不仅会耗费大量的人力物力，也会使操作复杂、繁琐，并且安全性也不是很好。而使用数据库系统进行管理，则会极大的减少人力物力资源消耗，操作简单，安全性高，良好的用户交互界面，有着人工管理不可比拟的优势。所以，教务管理信息系统应运而生。

### 1.3.3系统功能以及性能要求简述

功能要求：登录功能、添加课程、查询信息（学生选课信息、选课学生名单、开课信息等）

性能要求：系统易操作性、系统可维护性、系统高效性、系统开放性

## 1.4系统分析

### 1.4.1数据流图

1.学生功能数据流图



图1-1

2.教师功能数据流图



图1-2

3.管理员功能数据流图



图1-3

### 1.5.2数据字典

1.数据项的词条描述

数据项是数据流图中数据块的数据结构中的数据项说明，数据项不可再分。数据项一般包括数据项、含义说明、类型、长度、取值范围、取值含义、与其他数据项的关系

表1-1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 数据项 | 含义说明 | 类型 | 长度 | 取值范围 | 取值含义 | 与其他数据项的关系 |
| 学院编号 | 标识每个学院 | 字符型 | 4 |  | 后两位取0，用于标识专业 | 决定与学院有关的数据项 |
| 学院名称 | 学院姓名 | 字符型 | 12 |  | 学院标志，不可重复 | 与学院编号一一对应 |
| 专业编号 | 标识每个专业 | 字符型 | 4 |  | 前两位为学院编号前两位 | 决定与专业有关的数据项 |
| 专业名称 | 专业名称 | 字符型 | 12 |  | 专业标志，不可重复 | 与专业编号一一对应 |
| 班级编号 | 标识每个班级 | 字符型 | 8 |  | 前四位为专业编号 | 决定与班级有关的数据项 |
| 班级名称 | 班级名称 | 字符型 | 12 |  | 班级标志，不可重复 | 与班级编号一一对应 |
| 学号 | 标识每个学生，也是账号 | 字符型 | 13 |  | 包含专业、班级等信息 | 决定于学生有关的数据项 |
| 姓名 | 学生和教师姓名 | 字符型 | 12 |  | 存在少数民族同学 |  |
| 性别 | 标识性别 | 字符型 | 2 |  | 只有男女性别 |  |
| 民族 | 标识民族 | 字符型 | 8 |  | 存在少数民族同学 |  |
| 籍贯 | 学生和老师的籍贯 | 字符型 | 8 |  | 省市级别 |  |
| 入学年份 | 学生入学年份 | Year类型 | 4 | 1000~9999 | Year类型表示 |  |
| 职工号 | 标识每个职工，也是账号 | 字符型 | 11 |  | 职工唯一标识 | 决定与职工有关的数据项 |
| 职称 | 职工的职称 | 字符型 | 5 |  | 职工职称，可重复 |  |
| 入职年份 | 职工入职年份 | Year类型 | 4 | 1000~9999 | Year类型表示 |  |
| 管理员编号 | 标识每个管理员，也是登录的账号 | 字符型 | 6 |  | 不同学院不同管理员编号 | 决定与管理员有关的数据项 |
| 密码 | 学生、教师和管理员登录的密码 | 字符型 | 8~16 |  | 密码最多16位，最低8位，安全性高 |  |
| 课程编号 | 标识每个课程 | 字符型 | 8 |  | 前四位为专业编号 | 决定与课程的数据项 |
| 课程安排编号 | 标识每个课程安排 | 字符型 | 9 |  | 前八位为课程编号 | 决定与课程安排有关的数据项 |
| 课程名称 | 课程的名称 | 字符型 | 8 |  | 定义课程的名称，不可重复 |  |
| 课程性质 | 标识课程的性质 | 字符型 | 4 |  | 标识课程选修、必修等性质 |  |
| 课程状态 | 标识课程的状态 | 字符型 | 3 |  | 标识课程待审核、已通过、已开课、待开课等状态 |  |
| 学分 | 课程学分的个数 | 实点数类型 | 3 | 0~6 | 每门课的课程学分数 |  |
| 课时数 | 课程的课时数 | 整型 | 2 | 2~100 | 定义每门课的课时数 |  |
| 课程周数 | 课程的周数时长 | 整型 | 2 | 1~20 | 定义每门课的上课周数 |  |
| 课程容量 | 每门课可容纳学生数量 | 整型 | 3 | 10~150 | 定义课程容纳的学生数量 |  |
| 开课学期 | 上课的学期 | 字符型 | 11 |  | 课程在不同的学期开课 |  |
| 已选人数 | 课程已选人数 | 整型 | 3 | 0~150 | 实际选课学生数目 |  |
| 成绩分数 | 学生选修课程的分数 | 整型 | 3 | 0~100 | 定义学生获得成绩 |  |
| 成绩时间 | 获得成绩的时间 | 字符型 | 11 |  | 定义获得时间的学期 |  |
| 上课周次 | 每次课程安排的周次 | 字符型 | 10 |  | 定义课程安排上课周次 |  |
| 上课天次 | 一周中的周几上课 | 字符型 | 3 |  | 定义一周中的某一天 |  |
| 上课课次 | 一天中的第几节上课 | 字符型 | 2 |  | 定义一天中的某一节课 |  |
| 上课地点 | 上课的具体地点 | 字符型 | 6 |  | 定义上课的具体地点 |  |

2.数据结构词条描述

数据结构反映了数据之间的组合关系，一个数据结构有若干个数据项组成，通常包括以下数据结构、含义说明、组成等

表1-2

|  |  |  |
| --- | --- | --- |
| 数据结构 | 含义说明 | 组成 |
| 学生 | 定义了一个学生的相关信息 | 学号、姓名、性别、民族、籍贯、入学年份、班级编号、密码 |
| 教师 | 定义了一个教师的相关信息 | 职工号、姓名、性别、民族、籍贯、入职年份、密码、职称，学院编号 |
| 管理员 | 定义了一个管理员的相关信息 | 管理员编号、密码 |
| 课程 | 老师申请开课后的结果以及原有课程的基本相关信息 | 课程编号、课程名称、课程性质、课程状态、学分、课时数、开课学期、职工号、学院编号 |
| 课程安排 | 管理员安排课程的结果 | 课程安排编号、课程容量、已选人数、课程周数、上课周次、上课天次、上课课次、上课地点、课程编号 |
| 选课信息 | 学生选课后提交的选课结果 | 学号、课程安排编号、课程名称、课程性质、成绩分数、成绩时间 |
| 课程安排列表 | 学生可选课程的列表 | 教师姓名、课程安排、课程名称、学分、课程性质 |
| 已选课程列表 | 学生已选上的课程的列表 | 教师姓名、课程安排、课程名称、学分、课程性质 |
| 学生上课课表 | 学生上课的课表 | 课程安排、课程名称 |
| 选课学生名单 | 教师查看选课学生名单 | 学号、学生姓名、课程安排编号、课程名称、成绩 |
| 教师教学课表 | 教师进行教学的课程列表 | 课程安排、课程名称 |
| 开课信息 | 教师申请开课的课程 | 课程 |
| 添加课程安排 | 管理员进行课程安排的结果 | 课程安排 |
| 验证信息 | 学生、教师、管理员登录系统的验证信息 | 账号、密码 |

3.数据流词条描述

数据流是数据结构在系统内传输的路径。对数据流的描述通常包括以下内容:数据流名、说明、数据流来源、数据流去向、组成等

表1-3

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 数据流编号 | 数据流名称 | 数据流描述 | 数据流来源 | 数据流去向 | 数据流组成 | 数据流量 | 高峰流量 |
| F1 | 选课信息 | 学生选课后提交的选课结果 | 学生 | 选课信息表 | 课程安排编号、学号、课程性质、成绩分数、成绩时间 | 50/天 | 5000/天 |
| F1’ | 选课信息 | 管理员查询的选课信息 | 选课信息表 | 管理员 | 课程安排编号、学号、课程性质、成绩分数、成绩时间 | 50/天 | 5000/天 |
| F2 | 课程安排列表 | 学生可选课程的列表 | 课程安排信息表 | 学生 | 教师姓名、课程安排、课程名称、学分、课程性质 | 50/天 | 5000/天 |
| F3 | 已选课程列表 | 学生已选上的课程的列表 | 选课信息表 | 学生 | 教师姓名、课程安排、课程名称、学分、课程性质 | 50/天 | 5000/天 |
| F4 | 学生上课课表 | 学生上课的课表 | 课程安排信息表 | 学生 | 课程安排、课程名称 | 50/天 | 50/天 |
| F5 | 选课学生名单 | 教师查看选课学生名单 | 选课信息表 | 教师 | 学号、学生姓名、课程安排编号、课程名称、成绩 | 10/天 | 10/天 |
| F6 | 教师教学课表 | 教师进行教学的课程列表 | 课程安排信息表 | 教师 | 课程安排、课程名称 | 10/天 | 10/天 |
| F7 | 开课信息 | 教师申请开课的课程 | 管理员 | 课程安排信息表 | 课程 | 10/天 | 10/天 |
| F8 | 添加课程安排 | 管理员进行课程安排的结果 | 管理员 | 课程安排信息表 | 课程安排 | 10/天 | 10/天 |
| F9 | 验证信息 | 用户进行登录验证 | 用户信息表 | 登录系统 | 账号、密码 | 100/天 | 100/天 |

4.处理过程

处理过程是数据流图中功能块的说明，数据字典中只需要描述处理过程的说明性信息，通常包括处理过程名，含义说明，输入，输出，处理等

表1-4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 处理过程名 | 含义说明 | 输入 | 输出 | 处理 |
| 登录 | 用于验证用户身份 | 账号、密码 |  | 验证输入的信息与原有信息是否一致 |
| 查询已选课程 | 查看学生已选课程 |  | 已选课程列表 | 返回给学生选课列表 |
| 申请选课 | 学生提交申请开课 | 课程安排列表 | 选课信息F1 | 选取课程安排中的课程 |
| 查询上课课表 | 查询课表安排 |  | 学生上课列表 | 从表中返回课表安排 |
| 查询选课学生名单 | 查询学生名单 |  | 选课学生名单 | 从选课信息表中返回学生名单 |
| 申请开课 | 教师提交申请 | 课程信息 | 课程信息 | 申请提交给管理员 |
| 查询教学课表 | 教师教学课表 |  | 教学课程列表 | 从表中返回课表安排 |
| 选课设置 | 设置选课时间段 | 选课时间段 | 开/关选课 | 根据输入的时间段决定学生是否可以选课 |
| 审核申请开课 | 审核教师提交的申请 | 课程 | 开课信息 | 将结果写入到课程表中 |
| 课程安排 | 具体地安排课程 | 课程 | 课程安排表 | 讲课程 安排结果写到表中 |
| 查询选课信息表 | 查看所有选课信息 |  | 选课信息表 | 学生选课结果 |

5.数据存储

数据存储是数据结构保留或存放的地方，也是数据流的来源和去向之一，它的内容一般包括数据存储名，说明，编号，输入，输出，组成，数据量，存取频度，存取方式等

表1-5

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据存储名 | 说明 | 编号 | 输入 | 输出 | 组成 | 数据量 | 存取频度 | 存取方式 |
| 选课信息表 | 存放学生选课信息 | D1 | 选课信息 | 选课信息 | 选课信息 | 大 | 高 | 随机存取 |
| 课程安排信息表 | 存放课程安排信息 | D2 | 添加课程安排 | 学生上课课表，教学课程列表 | 课程安排列表 | 大 | 高 | 随机存取 |
| 课程基本信息表 | 存放课程的基本信息 | D3 | 开课信息 | 课程基本信息 | 课程 | 大 | 中 | 随机存取 |
| 学生信息 | 存放学生信息 | D4 | 学生信息 | 学生信息 | 学生 | 大 | 高 | 随机存取 |
| 教师信息 | 存放教师信息 | D5 | 教师信息 | 教师信息 | 教师 | 大 | 中 | 随机存取 |
| 管理员信息 | 存放管理员信息 | D6 | 管理员信息 | 管理员信息 | 管理员 | 小 | 低 | 随机存取 |

## 1.6性能需求

1.系统易操作性

所开发的系统应做到操作简单，尽量使系统操作不受用户对电脑知识水平的限制，简单易用是首要目的。

2.系统可维护性

由于系统涉及的信息比较广，数据库中的数据需定期修改，系统可利用的空间及性能也随之下降，为了使系统更好地运转，学院可以对系统数据及一些简单的功能进行独立的维护及调整。

3.系统高效性

该系统面向老师、学生使用，会存在大量学生同时登录系统的情况，要充分考虑系统的并发性和高效性，要有足够快的响应速度，保证用户良好的使用体验。

4.系统具有开放性

该系统能够在开放的硬件体系结构中运行，并且能与其他系统顺利连接，不会因外部系统的不同面要做内在量的修改工作。

## 1.7开发工具及环境

开发工具：JDK-1.8、IDEA集成开发环境、MySQL等主要开发工具

运行环境：Windows、macOS等平台

# 2教学管理信息系统之数据库设计

## 2.1概念结构设计

### 2.1.1概念结构设计任务

将需求分析得到的用户需求抽象为信息结构即概念模型的过程就是概念结构设计。本阶段的任务就是根据教务管理信息系统-选课系统的需求分析文档，进行数据库的概念结构设计，抽象得到实体型以及实体之间的联系。

### 2.1.2实体型以及实体集之间的联系

1.抽象得到的实体型（主码用下划线标出）

学院（学院编号，学院名称）

专业（专业编号，专业名称）

班级（班级编号，班级名称）

学生（学号，姓名，性别，民族，籍贯，入学年份，班级，密码）

教师（职工号，姓名，性别，民族，籍贯，入职年份，职称，密码）

课程（课程编号，课程名称，课程性质，课程状态，学分，课时数，开课学期）

课程安排（课程安排编号，课程容量，已选人数，课程周数，上课周次，上课天次，上课课次，上课地点）

管理员（管理员编号，密码），管理员属于整个系统的管理人员，单独存在一个实体型

2.实体集之间的联系描述

学院同时开设多门课程；每门课程有一个主讲教师，一个老师可以主讲多门课程；每门课可以有多个课程安排，有多名学生选修；一个学生可选修多门课程并获得相应的学分和成绩；上课的基本单位是“次”（一次2学时），每一次课都规定了上课时间和教室

### 2.1.3教学管理信息系统E-R图设计

教学管理信息系统-选课系统E-R图如图2-1所示

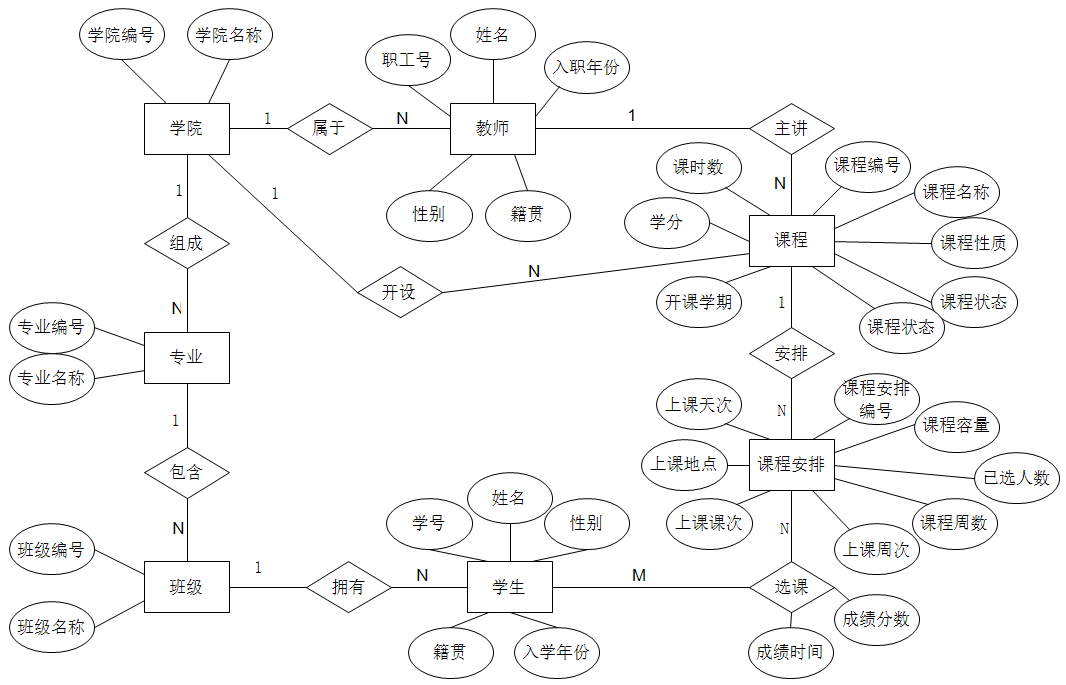


图2-1

## 2.2逻辑结构设计

### 2.2.1逻辑结构设计任务

逻辑设计的任务就是把概念设计阶段设计的E-R图转换为与选用DBMS产品所支持的数据模型相符合的逻辑结构。本阶段具体的工作就是将选课系统的E-R图转化成关系模式。

### 2.2.2教务管理信息系统关系模式

1.实体转换的关系模式有学院、专业、班级、学生、教师、课程、课程安排、管理员。联系转换的关系模式有选课。其中，“组成”联系与专业关系模式合并，“开设”与课程关系模式合并，“包含”联系与班级关系模式合并，“拥有”与学生关系模式合并，“属于”联系与教师关系模式合并，“安排”与课程安排关系模式合并，“主讲”联系与课程关系模式合并。

2.教务管理信息系统-选课系统关系模式（主码使用下划线标出）如下：

学院（学院编号，学院名称）

专业（专业编号，专业名称，学院编号），外码为学院编号

班级（班级编号，班级名称，专业编号），外码为专业编号

学生（学号，姓名，性别，民族，籍贯，入学年份，班级编号，密码），外码为班级编号

教师（职工号，姓名，性别，民族，籍贯，入职年份，职称，学院编号，密码），外码为学院编号

课程（课程编号，课程名称，课程性质，课程状态，学分，课时数，开课学期，职工号，学院编号），外码为职工号、学院编号

课程安排（课程安排编号，课程容量，已选人数，课程周数，上课周次，上课天次，上课课次，上课地点，课程编号），外码为课程编号

选课（课程安排编号，学号，成绩分数，成绩时间），外码为课程安排编号、学号

管理员（管理员编号，密码）

3.定义用户子模式如下：

课程安排列表（课程安排编号，课程名称，教师姓名，课程周数，上课周次，上课天次，上课课次，上课地点，课程容量，已选人数）

选课信息（课程安排编号，课程名称，职工号，教师姓名，学号，学生姓名）

上课课表（课程安排编号，课程名称，职工号，教师姓名，学号，学生姓名，课程周数，上课周次，上课天次，上课课次，上课地点）

选课学生名单（课程安排编号，学号，课程名称，学生姓名，成绩时间，成绩分数）

## 2.3物理结构设计

### 2.4.1物理结构设计任务

为一个给定的逻辑数据模型选取一个最适合应用环境的物理结构的过程就是数据库的物理设计。数据库在物理设备上的存储结构与存取方法称为物理结构。本阶段的任务就是为选课系统的逻辑数据模型选取最适合应用环境的物理结构。

### 2.4.2教学管理信息系统物理结构具体设计

1.有关数据文件的存储：教务管理信息系统数据库的数据文件、日志文件存放到指定的硬盘上，该硬盘最好不安装操作系统、DBMS等软件，数据库备份文件存放到移动硬盘。

2.根据实际应用需要建立索引，如表2.1所示。下表中由于学院中教师人数较多，且课程数量也相对较多，所以在职工号和课程编号上另外建立UNIQUE类型索引。下表中的其他关系模式则建立下表中的CLUSTER类型索引。

表2-1 索引列表

|  |  |  |
| --- | --- | --- |
| 关系模式 | 索引属性列 | 索引类型 |
| 专业 | 学院编号 | CLUSTER |
| 班级 | 专业编号 | CLUSTER |
| 学生 | 班级编号 | CLUSTER |
| 教师 | 学院编号 | CLUSTER |
| 教师 | 职工号 | UNIQUE |
| 课程 | 学院编号 | CLUSTER |
| 课程 | 课程编号 | UNIQUE |
| 课程安排 | 课程编号 | CLUSTER |

# 3教学管理信息系统之数据库实施

## 3.1数据库实施阶段的任务

数据库实施阶段的任务就是运用DBMS提供的数据语言、工具及宿主语言，根据逻辑设计和物理设计的结果 建立数据库，编制与调试应用程序，组织数据入库，并进行试运行。本阶段的任务就是将教学管理信息系统的概念结构设计和物理结构设计用MySQL严格描述出来，并且进行试运行。

## 3.2MySQL 5.6.24 Community Server概述

### 3.2.1安装

1.MySql安装向导提供了一个功能树以用来安装所有MySql组件：

①MySql Servers

②Applications

③MySQL Connectors

④Documentation

⑤Samples and Examples

2.MySql所需的网络协议：

①TCP/IP协议

②Shared memory协议

③Named Pipes协议

④Unix Socket协议

⑤TLS/SSL协议

3.MySql所需的软件：

①Navicat

4.MySql 5.6.24 Community Server所需的软硬件（最低配置）：

①32位Windows操作系统，如9x,Me,NT,2000,XP或Windows Server 2003

②Pentium Ⅲ兼容处理器 2.0GHz/内存 4G/硬盘 80G/VGA 1024x768

### 3.2.2配置

1.配置环境变量，在系统变量path后面追加MySql安装路径。

2.MySql配置向导，从MySql安装帮助启动 MySQL Configuration Wizard(配置向导)，或进入MySQL安装bin目录直接启动MySQLInstanceConfig.exe文件。通过配置向导，设置my.ini文件

3.导入和导出数据

Navicat提供了一套移动、复制及转换数据的图形化工具。

4.DatabaseUsage（数据库使用）对话框，通过该对话框可指出创建MySql表时使用的表处理器，可选择的有InnoDB存储引擎。MultiFunctional Database（多功能数据库）：选择该选项，则同时使用InnoDB和MyISAM存储引擎，并在两个引擎之间平分资源。

5.联网选项对话框

在Networking Options（网络选项）对话框中可以启用或禁用TCP/IP网络，并配置用来连接MySQL服务器的端口号。

6.字符集对话框

MySql服务器支持多种字符集，可以设置使用于所有表、列和数据表的默认服务器字符集。使用Character Set（字符集对话框）来更改MySql服务器的默认字符集

7.Navicat可视化工具

Navicat用于编辑和执行查询，创建索引等数据库操作的可视化。

### 3.2.3管理

1.服务器管理

MySQL Configuration Wizard（配置向导）将MySql服务器安装位服务，服务名为MySQL.

2.管理数据库引擎服务

可以使用Navicat管理数据库引擎，或修改配置文件my.ini管理。

3.备份和还原

使用Navicat创建和还原数据库、差异数据库、事务日志以及文件备份和文件组备份。

4.自动化管理

在管理员的工作中，有许多管理职责每天都是固定不变的，非常单调乏味。通过自动执行重复发生的管理任务和响应服务器事件，可以留出时间来执行其他需要创造性且缺乏可预测性或可编程性的响应的任务。

要使管理自动化，需要完成下列工作：

①建立定期发生的并且可以按编程方式进行管理的管理职责或服务器事件。

②定义一组批作业和警报，设置计划任务。

## 3.3数据库创建

### 3.3.1创建数据库

定义eduadmin\_db数据库：

CREATE DATABASE eduadmin\_db;

### 3.3.2创建基本表

在eduadmin\_db数据库上，根据关系模式定义基本表，结构如下：

1. 学院表：存储学院编号和名称信息

表3-1 college（学院表）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性列名 | 属性说明 | 数据类型 | 主码 | 外码 | 备注 |
| collNo | 学院编号 | CHAR(4) | 主码 |  | NOT NULL |
| collName | 学院名称 | VARCAHR(12) |  |  |  |

2.专业表：存储专业编号和名称信息

表3-2 major（专业表）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性列名 | 属性说明 | 数据类型 | 主码 | 外码 | 备注 |
| majorNo | 专业编号 | CHAR(4) | 主码 |  | NOT NULL |
| majorName | 专业名称 | VARCHAR(12) |  |  |  |
| collNo | 学院编号 | CHAR(4) |  | 外码 | NOT NULL |

3.班级表：存储班级编号和名称信息

表3-3 class（班级表）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性列名 | 属性说明 | 数据类型 | 主码 | 外码 | 备注 |
| classNo | 班级编号 | CHAR(8) | 主码 |  | NOT NULL |
| classNmae | 班级名称 | VARCAHR(12) |  |  |  |
| majorNo | 专业编号 | CHAR(4) |  | 外码 | NOT NULL |

4.学生表：存储学生基本信息

表3-4 student（学生表）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性列名 | 属性说明 | 数据类型 | 主码 | 外码 | 备注 |
| Sno | 学号 | CHAR(13) | 主码 |  | NOT NULL |
| Sname | 学生姓名 | VARCHAR(12) |  |  |  |
| Ssex | 性别 | CHAR(2) |  |  |  |
| Sethnic | 民族 | VARCHAR(8) |  |  |  |
| Shome | 籍贯 | ARCHAR(8) |  |  |  |
| Syear | 入学年份 | YEAR(4) |  |  |  |
| classNo | 班级编号 | CHAR(8) |  | 外码 | NOT NULL |
| Spassword | 密码 | VARCHAR(16) |  |  |  |

5.教师表：存储老师基本信息

表3-5 teacher（教师表）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性列名 | 属性说明 | 数据类型 | 主码 | 外码 | 备注 |
| teaNo | 教师编号 | CHAR(11) | 主码 |  | NOT NULL |
| teaName | 教师姓名 | VARCHAR(12) |  |  |  |
| teaSex | 性别 | CHAR(2) |  |  |  |
| teaEthnic | 民族 | VARCHAR(8) |  |  |  |
| teaHome | 籍贯 | VARCHAR(8) |  |  |  |
| teaYear | 入职年份 | YEAR(4) |  |  |  |
| teaTitle | 职称 | VARCHAR(5) |  |  |  |
| collNo | 学院编号 | CHAR(8) |  | 外码 | NOT NULL |
| Spassword | 密码 | VARCHAR(16) |  |  |  |

6.课程表：存储课程学分数、课时数、主讲老师等基本信息

表3-6 course（课程表）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性列名 | 属性说明 | 数据类型 | 主码 | 外码 | 备注 |
| cNo | 课程编号 | CHAR(8) | 主码 |  | NOT NULL |
| cName | 课程名称 | VARCHAR(8) |  |  |  |
| cNature | 课程性质 | VARCHAR(4) |  |  |  |
| cStatus | 课程状态 | CHAR(3) |  |  |  |
| cCredit | 学分 | NUMERIC(3,2) |  |  |  |
| cHour | 课时数 | INT(2) |  |  |  |
| cTerm | 开课学期 | CHAR(11) |  |  |  |
| teaNo | 教师编号 | CHAR(13) |  | 外码 | NOT NULL |
| collNo | 学院编号 | CHAR(4) |  | 外码 |

7.课程安排表：存储课程安排时间、地点等基本信息

表3-7 courseschedule（课程安排表）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性列名 | 属性说明 | 数据类型 | 主码 | 外码 | 备注 |
| csNo | 课程安排编号 | CHAR(9) | 主码 |  | NOT NULL |
| csCapacity | 课程容量 | INT(3) |  |  |  |
| csSelNum | 已选人数 | INT(3) |  |  |  |
| csWeeks | 上课周数 | INT(2) |  |  |  |
| csWeek | 上课周次 | VARCHAR(10) |  |  |  |
| csDay | 上课天次 | CHAR(3) |  |  |  |
| csClass | 上课课次 | CHAR(2) |  |  |  |
| csAdress | 上课地点 | CHAR(6) |  |  |  |
| cNo | 课程编号 | CHAR(8) |  | 外码 | NOT NULL |

8.选课表：存储课程选课学生学号、成绩等基本信息

表3-8 sc（选课表）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性列名 | 属性说明 | 数据类型 | 主码 | 外码 | 备注 |
| csNo | 课程安排编号 | CHAR(9) | 主码 | 外码 | NOT NULL |
| Sno | 学号 | CHAR(13) |
| SCgrade | 成绩分数 | INT(3) |  |  |  |
| SCgradeTime | 成绩时间 | CHAR(11) |  |  |  |

9.管理员表：存储管理员账号和密码

表3-9 adminUser（管理员表）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性列名 | 属性说明 | 数据类型 | 主码 | 外码 | 备注 |
| adminNo | 管理员编号 | CHAR(6) | 主码 |  | NOT NULL |
| adminPass | 密码 | VARCHAR(16) |  |  |  |

### 3.3.3创建视图

根据概念结构设计部分的结果进行视图的具体实现，语句如下：

1.课程安排列表视图（csList\_view）

CREATE VIEW csList\_view AS SELECT csNo,cName,teaName,csWeeks,csWeek,csDay,csClass,csAdress,csCapacity,csSelNum

FROM courseschedule,teacher,course

WHERE courseschedule.cNo=course.cNo

AND course.teaNo=teacher.teaNo;

2.上课课表安排视图（goclassList\_view）

CREATE VIEW goclassList\_view AS SELECT courseschedule.csNo,cName,course.teaNo,teaName,sc.Sno,Sname,csWeeks,csWeek,csDay,csClass,csAdress

FROM courseschedule,teacher,student,course,sc

WHERE courseschedule.cNo=course.cNo

AND course.teaNo=teacher.teaNo

AND sc.csNo=courseschedule.csNo

AND sc.Sno=student.Sno;

3.选课信息视图（sc\_view）

CREATE VIEW sc\_view AS SELECT courseschedule.csNo,cName,teacher.teaNo,teaName,student.Sno,Sname

FROM courseschedule,teacher,student,course,sc

WHERE courseschedule.cNo=course.cNo

AND course.teaNo=teacher.teaNo

AND courseschedule.csNo=sc.csNo

AND sc.Sno=student.Sno;

4.选课学生列表视图（stuList\_view）

CREATE VIEW stuList\_view AS SELECT courseschedule.csNo,student.Sno,cName,Sname,SCgrade,SCgradeTime

FROM courseschedule,student,course,sc

WHERE courseschedule.cNo=course.cNo

AND courseschedule.csNo=sc.csNo

AND sc.Sno=student.Sno;

### 3.3.4创建索引

根据物理设计阶段结果进行索引的具体实现，创建语句如下：

1.CREATE INDEX collNo\_index ON major(collNo);

2.CREATE INDEX majorNo\_index ON class(majorNo);

3.CREATE INDEX classNo\_index ON student(classNo);

4.CREATE INDEX collNo\_index ON teacher(collNo);

CREATE UNIQUE INDEX teaNo\_index ON teacher(teaNo);

5.CREATE INDEX collNo\_index ON course(collNo);

CREATE UNIQUE INDEX cNo\_index ON course(cNo);

6.CREATE INDEX cNo\_index ON courseschedule(cNo);

## 3.4数据加载

1.组织数据

按照设计的数据库结构，使用Excel组织数据库需要的学院、专业、班级、学生、教师、课程等相关数据，将表格中的数据进行标准化处理，并对应的数据列的名称插入进表格上方第一行。

2. 导入数据

①在Navicat可视化操作的数据库表中找到“导入”，并在弹窗中找到对应的execl表格文件类型，点击下一步。

②选择需要导入的文件路径，并选择文件中的工作表，点击下一步。

③确定默认设置，确认目标表和源表，源栏位和目标栏位的对应关系，最后确定导入模式，若添加数据，默认选项即可实现需求，参数设置完成后，点击开始，等待文件添加完成，当出现Finished-Successfully，即导入成功。

# 4.教学管理信息系统之数据访问

## 4.1 JDBC应用程序接口

### 4.1.2 JDBC简介

JDBC（Java DataBase Connectivity, Java数据库连接）是一种用于执行SQL语句的Java API，可以为多种关系数据库提供统一访问，它由一组用Java语言编写的类和接口组成。JDBC提供了一种基准，据此可以构建更高级的工具和接口，使数据库开发人员能够编写数据库应用程序。

JDBC API 既支持数据库访问的两层模型（C/S），同时也支持三层模型（B/S）。在两层模型中，Java applet或应用程序将直接与数据库进行对话。这将需要一个JDBC驱动程序来与所访问的特定数据库管理系统进行 通讯。用户的SQL语句被送往数据库中，而其结果将被送回给用户。数据库可以位于另一台计算机上，用户通过网络连接到上面。这就叫做客户机/服务器配置，其中用户的计算机为客户机，提供数据库的计算机为服务器。网络可以是 Intranet（它可将公司职员连接起来），也可以是 Internet。

在三层模型中，命令先是被发送到服务的"中间层"，然后由它将SQL 语句发送给数据库。数据库对 SQL 语句进行处理并将结果送回到中间层，中间层再将结果送回给用户。MIS 主管们都发现三层模型很吸引人，因为可用中间层来控制对公司数据的访问和可作的的更新的种类。中间层的另一个好处是，用户可以利用易于使用的高级API，而中间层将把它转换为相应的低级调用。最后，许多情况下三层结构可提供一些性能上的好处。

### 4.1.3 Java程序连接数据库的方式：OCI方式

这种实现方法是直接使用数据库厂商提供的用专用的网络协议创建的驱动程序，通过它可以直接将JDBC API调用转换为直接网络调用。这种调用方式一般性能比较好，而且也是实用中最简单的方法。因为它不需要安装其他的库或中间件。几乎所有的数据库厂商都为他们的数据库提供了这种数据库提供了这种JDBC驱动程序，也可以从第三方厂商获得这些驱动程序。

基本流程：应用程序---JDBC API---驱动程序---数据源

## 4.2数据库连接池

### 4.2.1数据库连接池定义

数据库连接是一种关键的有限的昂贵的资源，这一点在多用户的网页应用程序中体现得尤为突出。对数据库连接的管理能显著影响到整个应用程序的伸缩性和健壮性，影响到程序的性能指标。数据库连接池正是针对这个问题提出来的。

数据库连接池负责分配、管理和释放数据库连接，它允许应用程序重复使用一个现有的数据库连接，而不是再重新建立一个；释放空闲时间超过最大空闲时间的数据库连接来避免因为没有释放数据库连接而引起的数据库连接遗漏。这项技术能明显提高对数据库操作的性能。

### 4.2.2数据库连接池实现

一般来说，Java应用程序访问数据库的过程是：

①装载数据库驱动程序；

②通过jdbc建立数据库连接；

③访问数据库，执行sql语句；

④断开数据库连接

### 4.2.3使用开源的数据库连接池DBCP

1.导入相关jar包

① Commons-dbcp.jar：连接池的实现

② Commons-pool.jar：连接池实现的依赖库

2、在类目录下加入dbcp的配置文件：dbcpconfig.properties

# 5.教学管理信息系统之应用系统设计

## 5.1系统设计

### 5.1.2开发环境

本系统采用MVC三层结构模式，在Windows平台上开发。软硬件的配置如下：

①Intel（R）Core（TM）i5-8250U 8G/内存 256G/固态硬盘

②Windows 10简体中文专业版

③MySQL 5.6.24

④IntelliJ IDEA 2019 旗舰版

⑤JDK1.8

5.1.2系统总体设计

选课子系统功能结构图如下所示：



### 5.1.3功能说明

1.登录功能：用户共有三种角色，分别是学生、教师和管理员。其中学生账号、初始密码均为学号，教师的账号、初始密码均为职工号，管理员账号、初始密码均为管理员编号。不同角色登录到不同界面。

2.学生选课：学生可以根据教务处给出的课程安排进行添加、删除选课

3.学生查询已选课程：学生可以查看自己已选课程的基本信息，其中包括成绩、学分、课程性质等

4.教师查询课程安排：教师可以查看自己所授课程的安排

5.教师申请开课：教师可以选择申请开设课程，等待管理员审核

6.管理员同意申请开课：管理员审核教师申请开设的课程，并给出结果，此界面会显示所有课程信息

7.管理员添加课程安排：管理员可以根据已开设的课程进行排课，供学生选课

8.管理员设置选课设置：管理员设开启/关闭选课

9.管理员查询选课信息：管理员可以查看有关选课信息

### 5.1.3系统详细设计

**1、主要程序流程图：**

学生功能流程图如图5-1所示：



图5-1

教师功能流程图如图5-2所示：



图5-2

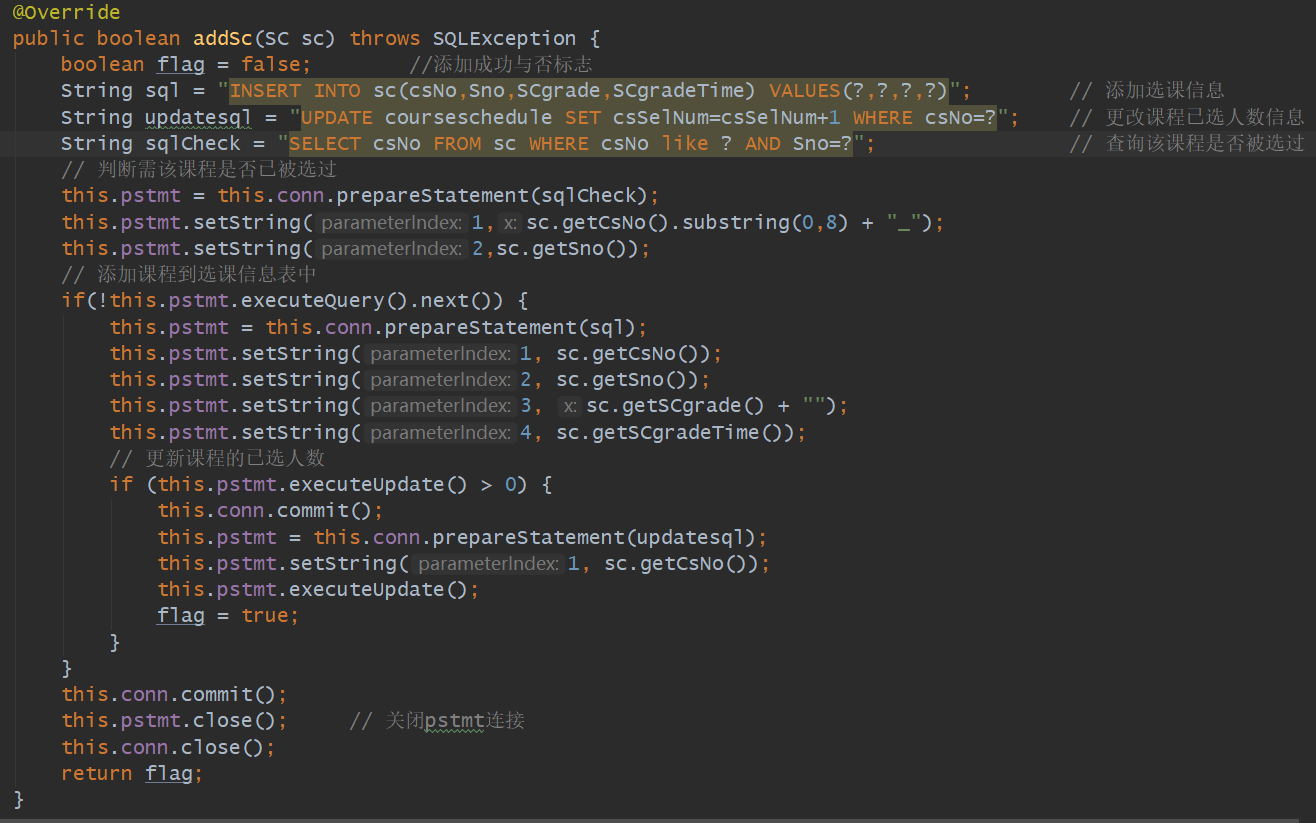
管理员功能流程图如图5-3所示：



图5-3

**2、主要代码展示：**

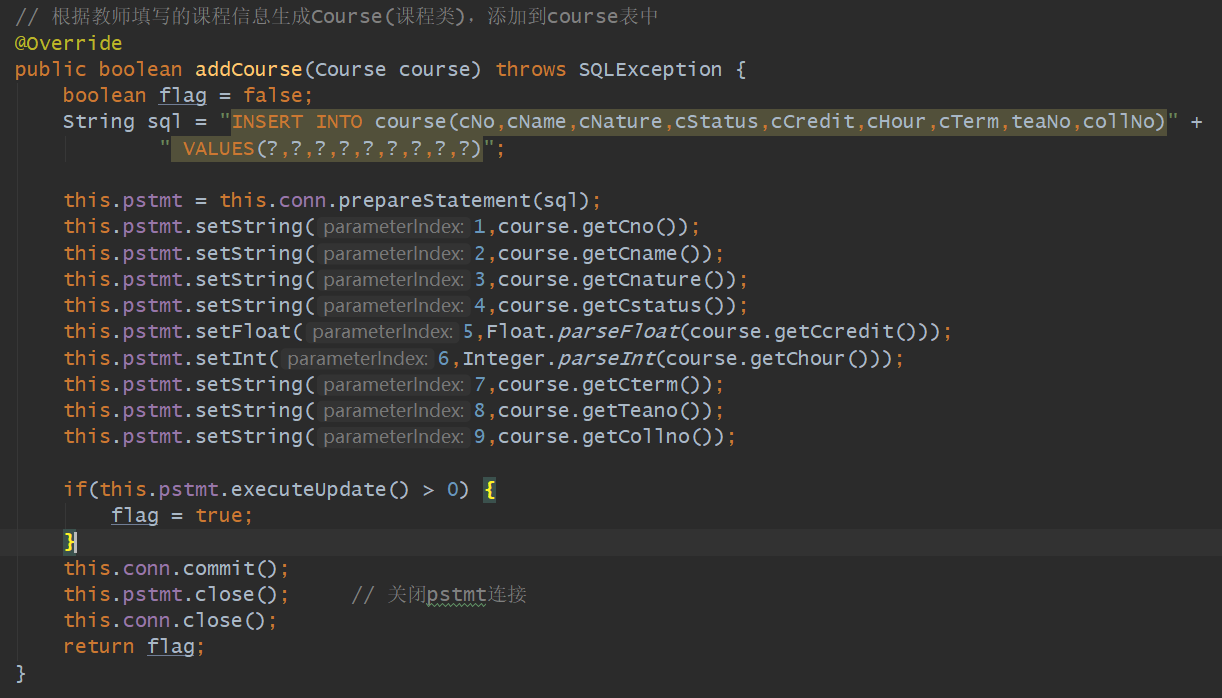
选课代码：



申请开课代码：



添加课程安排代码：



## 5.2系统实现及结果测试

登录模块：

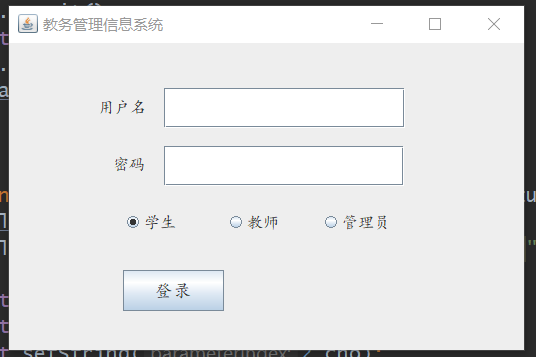


图5-4

学生界面：



图5-5

选课界面：学生可以在选课时间段内根据需要进行选课

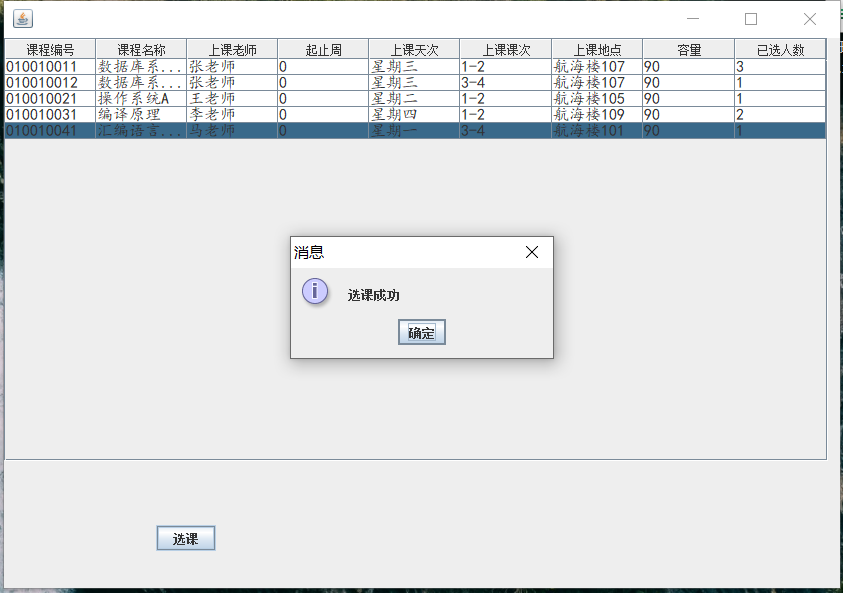


图5-6

查看已选课程：学生可以查看已选择之后的课程，可以根据需要自行退选

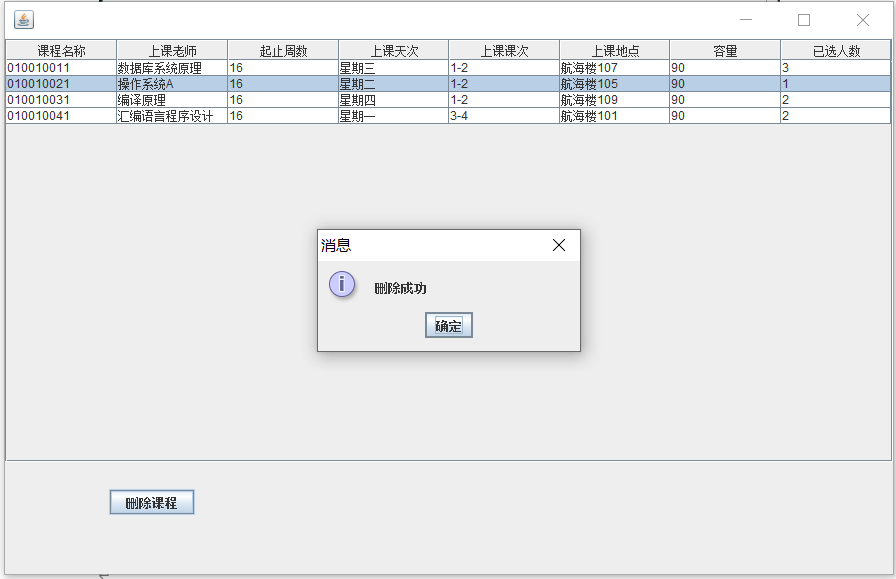


图5-7

申请开课模块：教师根据需要申请开课，管理员审核通过后即可进行排课

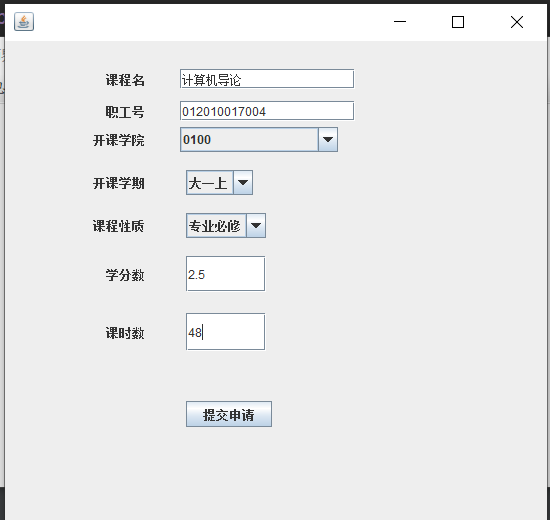


图5-8

添加课程安排模块：管理员根据已有课程进行排课，供学生选课

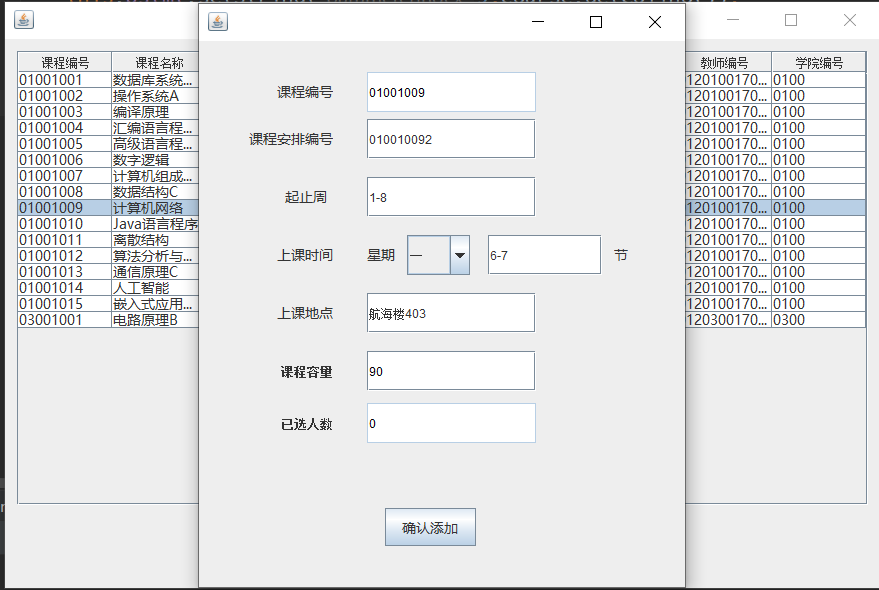


图5-9

# 6.总结体会

通过本次的实验，我对数据库系统原理这门课进一步的认识和理解。并且我亲自动手实现了一个简单的信息管理系统，将理论知识应用于实践，不仅要对数据库的原理牢牢掌握，更要学会实际使用。纸上空谈始终是达不到好的效果的，如果想要真正的理解这门课，就必须进行大量的实践。

在本次的实验中我也遇到了一些问题，例如sql语句书写错误导致无法得到正确结果、数据库事务提交等问题。对于这些问题，常常需要学会使用工具进行调试，找到问题并解决。更加有效的办法就是多写代码，通过大量的代码积累经验，遇到的问题或许不会减少，但能够快速找到问题所在并解决。

实践出真知，我们需要大量的实践去检验自己的学习成果，不断地踩坑，积累经验，从而提升自我。在这次实验中，我也是受益匪浅，收获颇多。

# 7.参考文献

[1].数据库系统概论（第5版），王珊 萨师煊，高等教育出版社，2017.3

[2].软件工程导论（第6版），张海藩 牟永敏，清华大学出版社，2013.10

[3].Java核心技术卷Ⅰ（第10版），Cay S. Horstmann，机械工业出版社，2016.9

**成绩评定表**

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 评分项目 | 满分 | 实得分 |
| 1 | 学习态度认真、出勤情况 | 10 |  |
| 2 | 设计分析合理性 | 10 |  |
| 3 | 设计方案正确性、可行性、创造性、设计结果正确性 | 30 |  |
| 4 | 系统测试、运行情况 | 40 |  |
| 5 | 设计报告的规范性 | 10 |  |
|  |  | 总得分 |  |