



**Java考核**

课程名称 Java程序设计

题目名称 光盘借阅管理系统

学生学院 计算机

专业班级 计算机17(5)班

学 号 3117004622

学生姓名 刘友滨

指导教师 赵锐

2018 年 12 月 4 日

|  |  |
| --- | --- |
| 难度系数 |  |
| 独立完成工作量占总工作量的比例 |  |
| 程序功能完成情况 |  |
| 报告内容与格式 |  |
| 总评成绩 |  |

光盘借阅管理系统

**光盘借阅管理**

设计以下几个类：**光盘类：用户类：借阅记录类**，包括下列模块：

1) 新建、修改、删除光盘信息；

2) 新建、修改、删除用户信息；

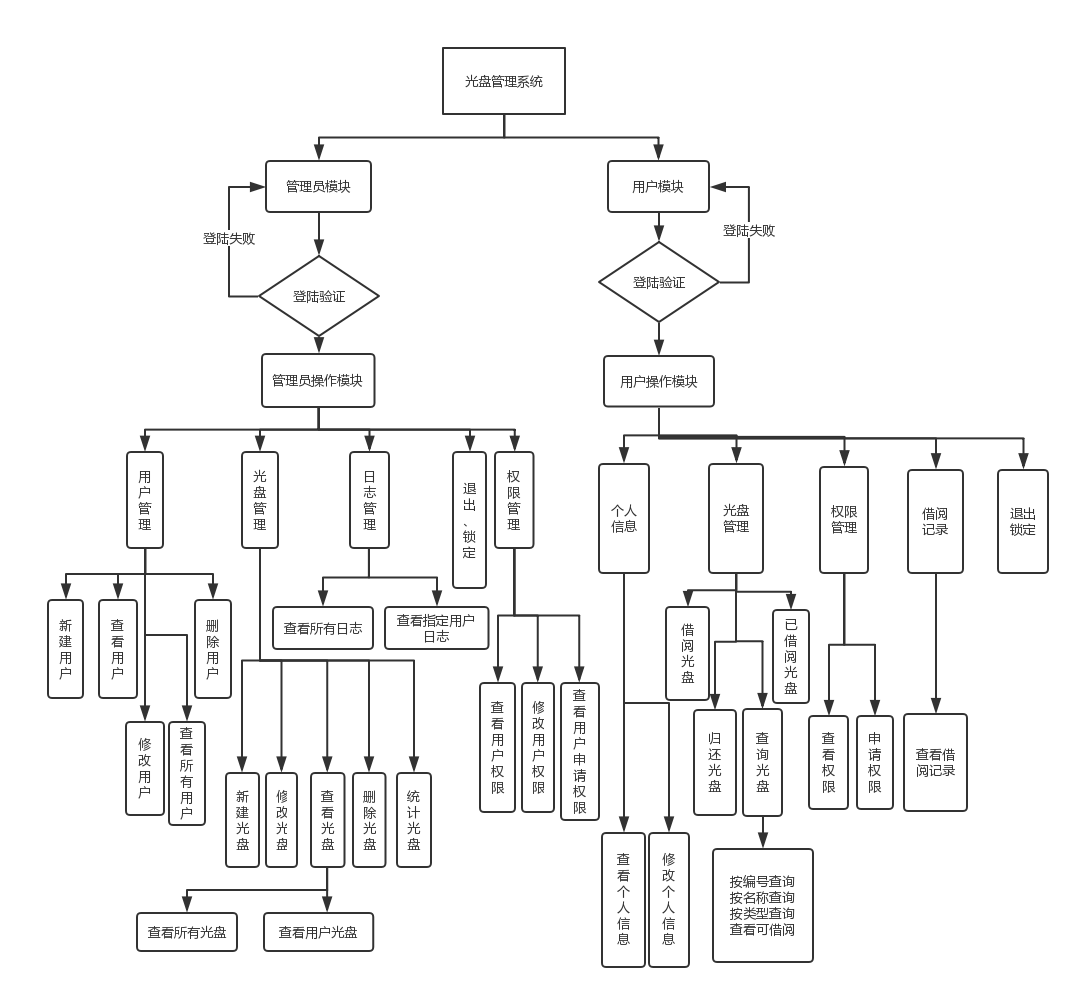
3) 光盘借阅/归还信息记录；

4) 查询光盘名称，用户的借阅历史；

要求：基于JFrame窗体设计、使用菜单和数据库。

开发环境：Eclipse JDK1.8 MySQL8.0

题目需求：开发一个光盘借阅管理系统。主要用于满足用户的光盘借阅归还需求，管理员的管理用户/光盘的需求。要求实现新建、修改、删除用户信息，新建、修改、删除光盘信息，要求可以记录、查询光盘的借阅记录，可以查询用户的借阅历史，要求可以查询光盘的信息。



界面设计过程

本次课设的界面采用了Window Builder插件进行辅助设计。根据对软件功能的设计，本次界面共设计了五个窗口。分别是“总登陆窗口（LoginWindow）”、“管理员登陆窗口(ManagerLoginWiindow)”、“用户登陆窗口(UserLoginWindow)”、“管理员操作窗口(ManagerOperateWindow)”、“用户操作窗口(UserOperateWindow)”。

窗口界面的设计主要采用菜单条进行功能分类管理。每一个主要的功能采用面板添加组件的方式进行交互实现。例如:新建、删除、修改用户/光盘等功能采用单独的面板+组件进行交互，当用户触发事件时，调用相应面板进行交互。同时对于查询功能进行了一些布局的修改，比如查询光盘、借阅信息等功能则采用表（JTable）组件进行信息的展示，增强用户体验；同时对于查询功能进行了尽可能地细分比如：查询光盘功能。为了方便用户查询，我将查询光盘细分为按光盘编号、按光盘名称、按光盘类型、查询可借阅光盘、查询已借阅光盘进行细分。同时在界面设计还注重交互体验，例如改变了默认的窗口风格、添加适和的背景图片、增加按钮动态交互。同时考虑到用户的信息安全，在原先的功能基础上添加了锁定功能。在界面上独立设置一个面板用于锁定，提高安全性能。

数据库设计

本次课设的数据库采用MySQL 8.0。

针对所需的软件功能，本次设计了5张表：光盘表（cd）、用户表（user）、管理员表（manager）、借阅记录表（record）、权限表（root）。

【光盘表（cd）】用于记录光盘的信息包括：光盘编号（cid）、光盘名称（cname）、

光盘类型（ctype）、借阅人（borrowed）、借阅时间（btime）。

cid为主键，borrowed为外键关联用户表的（uacoount）字段。同时设置成置空删除。

【用户表（user）】用于记录用户的基本信息：账户（uacoount）、密码（upwd）

、用户姓名(uname)、用户年龄（age）、用户性别（sex）、用户出生日期（birthday）。uaccount为主键。

【管理员表（manager）】用于记录管理员的信息：编号（mid）、账号（maccount）、密码（mpwd）。mid为主键。

【借阅记录表（record）】用于记录用户的借阅\归还信息：记录编号（id）、用户账户（useraccount）、用户操作行为（action）、用户操作时间（btime）、光盘编号（cid）。

Id为主键，useraccount为外键关联用户表的uaccount，同时设置为级联删除。

【权限表（root）】用于记录用户的权限信息：编号（id）、用户账户（useraccount）、借阅权限（borrowroot）、修改权限（modifyroot）、申请借阅权限（applyborrowroot）、申请修改权限（applymodifyroot）。id为主键，useraccount为外键关联用户表的uaccount，同时设置为级联删除。

数据库作为管理系统重要的一部分，针对系统的功能进行了充分的设计。其中已用户表（user）为主表，借阅记录表（record）、光盘表（cd）、权限表（root）为从表。在一定程度上方便了开发，规避了一定的数据出错风险。同时数据库的设计还根据了实际情况进行了约束的设置。

程序实现的论述

经过对光盘管理系统的功能分析，采用基于面向对象思想的设计方法，设计了各种类包括主要的逻辑代码类，窗体类，工具类，数据库映射类，数据库管理类。为了代码便于阅读以及维护，对代码进行的分包，分包情况如下：

bean包： 存放配置文件的封装类

core包： 存放数据库的管理类

image包： 存放软件所需的图片

managerDao包：存放管理员登陆操作主要的逻辑代码

po包： 存放数据库表的映射类

userDao包： 存放用户登陆操作主要的逻辑代码

util包： 存放各种工具类

window包： 存放窗体类

po包里存放了数据库表的映射类：CD、Manager、User、Record、Root，每一个类对应了数据库的一张表。这样设计的好处是便于将数据封装成对应的对象实现往数据里插入、删除、更新数据。便于从数据库里查询数据封装成对应的对象实现数据的获取。

bean包存放的Configration类主要是封装配置文件的配置信息，便于配置信息的修改与加载。

core包存放了DBManager类，当此类加载的时候加载配置文件，管理数据库的连接与关闭。

window包存放了LoginWindow、ManagerLoginWindow、UserLoginWindow、ManagerOperateWindow、UserOperateWindow五个窗体类。用于功能的交互实现。

util包存放各种常用的工具类。DBUtil类封装了数据库的更新操作、StringUtil类封装了字符串的常用操作、JDBCUtil类封装了常用的jdbc操作、ConversionUtil封装了数据转换的操作、QueryUtil封装了数据库查询的操作。

ManagerDao包存放了管理员的操作类和接口。checkAccountAndPwd接口规范了账户验证操作、ManagerQuery类存放了管理员的账号密码验证操作、ManagerOperateInt接口规范了管理员的操作方法、MabagerOperation类存放了管理员的操作方法。

UserDao包存放了用户的操作类和接口。checkAccountAndPwd接口规范了账户验证操作、UserQuery类存放了用户的账号密码验证操作、UserOperateInt接口规范用户的操作方法、UserOperation类存放了用户的操作方法。

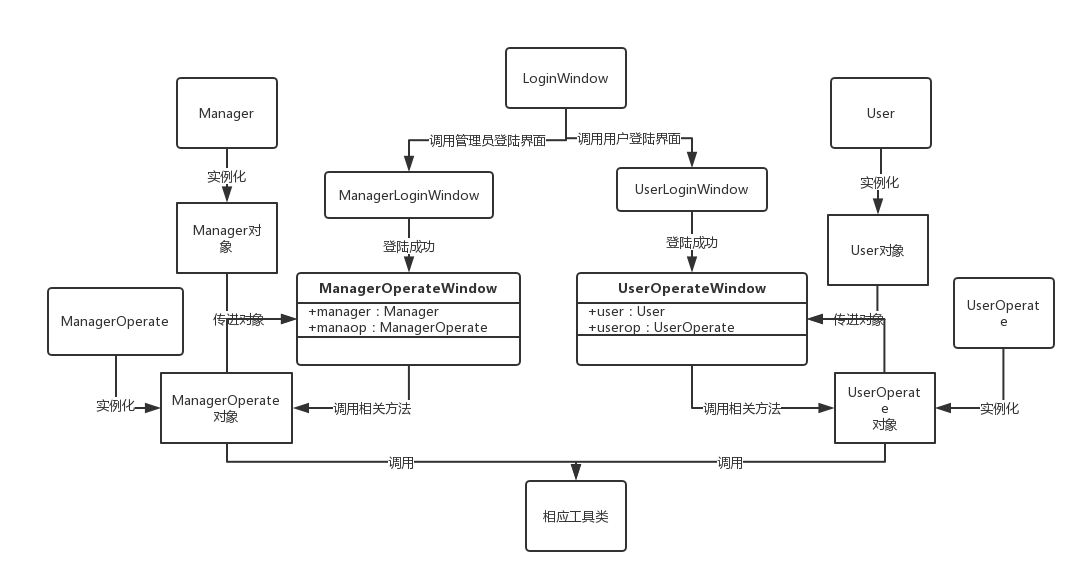
程序的实现思路：

设计两个模块：管理员登陆操作模块、用户登陆操作模块。管理员/用户登陆时首先进行账户密码验证，当账户密码与数据库数据符合时登陆成功，进入相应的操作窗体，同时在验证账户成功时，将账户信息封装成相应的对象传到操作界面，这样保证一个界面对应唯一的用户，便于数据的获取与更新。

操作窗体采用菜单条进行功能分类管理。每个功能采用一个独立的面板进行交互实现，采用方法进行封装。每一个操作窗体包含一个唯一的使用者对象（登陆成功时使用者对象引用传到界面层），以及一个操作模块对象（便于功能的实现）。

当用户进行菜单交互时，触发事件监听，调用相应的方法进行面板的调用。当功能进行时，界面层就调用相应的Dao层进行逻辑处理，Dao层接受数据调用相应的工具方法实现数据库的更新。例如用户要进行个人信息的修改，当用户点击确认修改时，程序首先对数据进行验证，当数据通过验证时，调用UserOperate对象（Dao层）的相应方法进行处理，Dao层方法又继续调工具方法进行数据库操作。这样着，将用户界面代码、逻辑代码、数据库操作代码分开，有利于解耦程序的维护。

主要类以及其关系



测试分析

1 本次课设的java扩展包有：

mysql-connection-java5.1.44-bin.jar

gnujaxp.jar

jcommon-1.0.23.jar

jreechart-1.0.19.jar

部署过程：打开eclipse-选中项目-点击右键选择Bulide path-在Libraries-选择Add External JARs-添加所需的jar包。

数据库配置信息：

{

driver=com.mysql.jdbc.Driver

url=jdbc:mysql://localhost:3306/cdleding?useUnicode=true&characterEncoding=utf-8&useSSL=false

ures=root

pwd=lyb980328

}

由于本次数据库的加载配置信息写在一个porperties文件中，以上是配置文件中的内容

在程序加载时将配置信息封装到一个Configration对象中，以便数据库连接时使用

2．测试数据的构建以及预计结果

针对软件功能，下面给出测试数据以及预计结果

管理员模块测试数据

|  |  |  |
| --- | --- | --- |
| 测试功能 | 测试数据 | 预计结果 |
| 登陆验证 | 账号：65432 密码：111 | 账号错误，登陆不成功 |
| 账号：123456 密码：222 | 密码错误，登陆不成 |
| 账号：123456 密码：123 | 登陆成功 |
| 新建用户 | 进入新建用户页面  输入符合规范的数据 | 新建用户成功 |
| 修改用户信息 | 输入账号：734635746 | 进入指定账户修改页面 |
| 查看用户信息 | 输入账号：734635746 | 查看指定账号的信息 |
| 查看所有用户信息 | 无 | 展示所有用户信息 |
| 删除用户 | 输入存在的账户 | 进入删除用户的页面 |
| 新建光盘 | 进入新建光盘页面  输入光盘数据 | 新建光盘成功 |
| 修改光盘 | 输入光盘编号：015  进入015光盘修改页面 | 修改光盘成功 |
| 删除光盘 | 输入存在的光盘编号，选择删除 | 删除光盘成功 |
| 查看所有光盘 | 无 | 展示所有光盘的信息 |
| 查看指定用户借阅光盘 | 输入账号：734635746 | 展示光盘信息 |
| 光盘借阅统计 | 选择统计图 | 展示统计信息 |
| 查看所有日志 | 无 | 展示借阅日志 |
| 查看指定用户日志 | 输入账号：734635746 | 展示指定用户的借阅日志 |
| 用户权限申请 | 无 | 展示用户的权限申请 |
| 修改用户权限 | 输入账号：734635746  进入修改页面进行修改 | 修改成功 |
| 查看用户权限 | 输入账号：734635746 | 展示指定用户的权限信息 |
| 锁定 | 无 | 锁定窗口 |

用户模块测试数据

|  |  |  |
| --- | --- | --- |
| 测试功能 | 测试数据 | 预计结果 |
| 登陆测试 | 输入账号：73463574  输入密码：111 | 账号错误，登陆失败 |
| 输入账号：734635746  输入密码：111 | 密码错误，登陆失败 |
| 输入账号：734635746  输入密码：980328 | 登陆成功 |
| 查看个人信息 | 无 | 展示个人信息 |
| 修改个人信息 | 进入修改页面，修改 | 修改成功 |
| 借阅光盘 | 输入空闲光盘编号 | 借阅成功 |
| 归还光盘 | 输入已借阅光盘 | 归还成功 |
| 查看已借阅光盘 | 无 | 展示已借阅光盘信息 |
| 按光盘编号查询 | 输入光盘编号015 | 展示指定光盘的信息 |
| 按光盘名称查询 | 输入光盘名称：动物世界 | 查看指定光盘信息 |
| 按光盘类型查询 | 选择光盘类型 | 查看指定光盘信息 |
| 查看已借阅光盘 | 无 | 展示已借阅光盘信息 |
| 查看权限信息 | 无 | 展示权限信息 |
| 申请权限 | 进入申请界面 | 申请成功 |
| 查看借阅日志 | 无 | 展示借阅日志 |
| 锁定 | 无 | 显示锁定窗口 |

3 测试结果

总登陆界面 管理员登陆验证



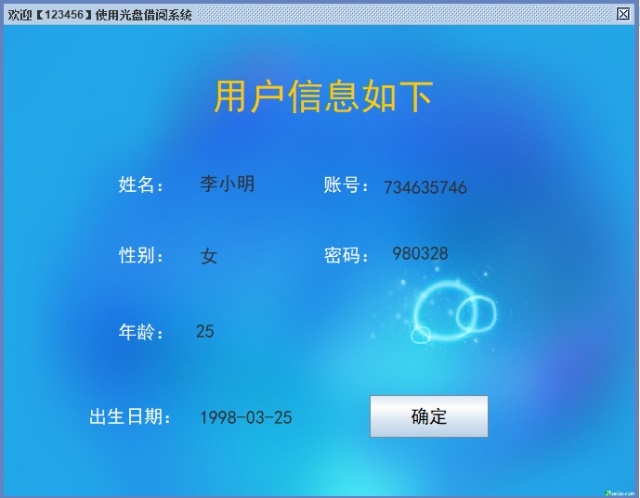
账号错误 密码错误



登陆成功 新建用户



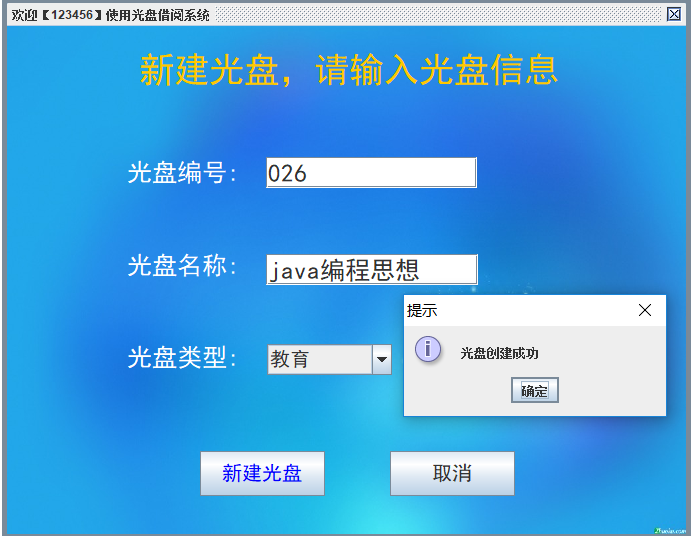
查看用户信息 修改用户信息



查看所有用户 删除用户



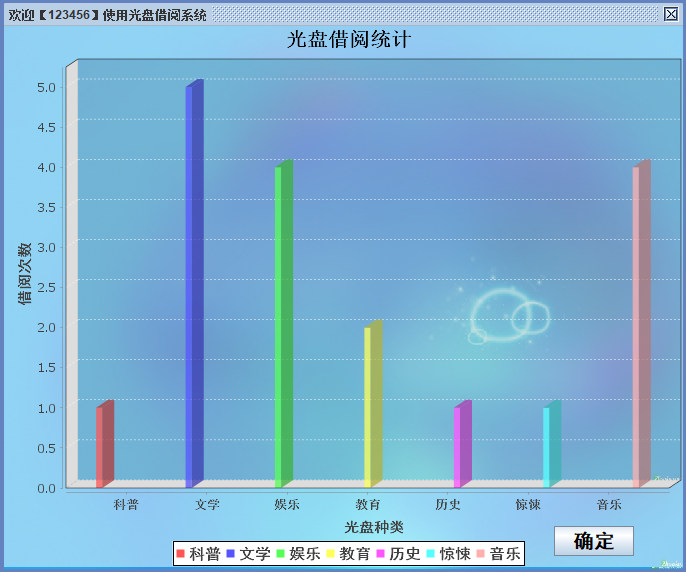
新建光盘 修改光盘



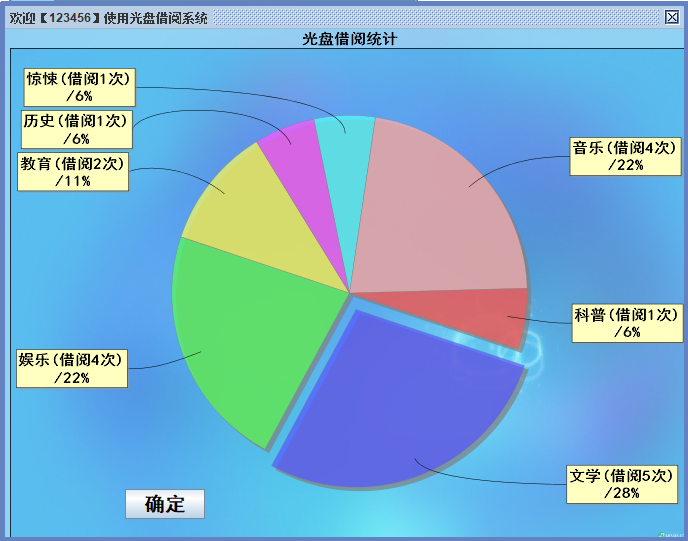
查看所有光盘 查看指定用户光盘



删除光盘 柱状图统计光盘借阅情况



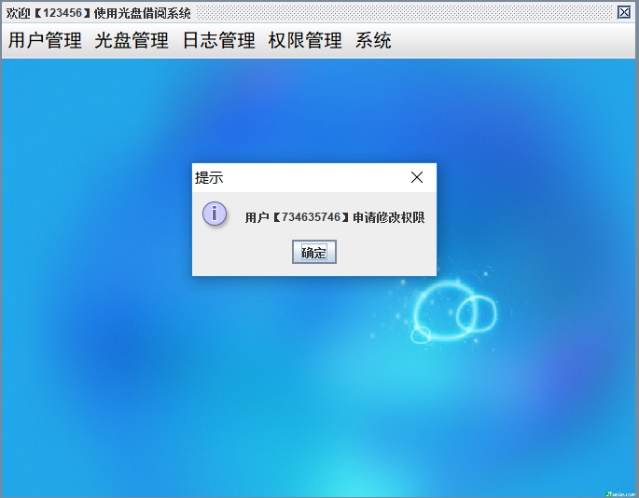
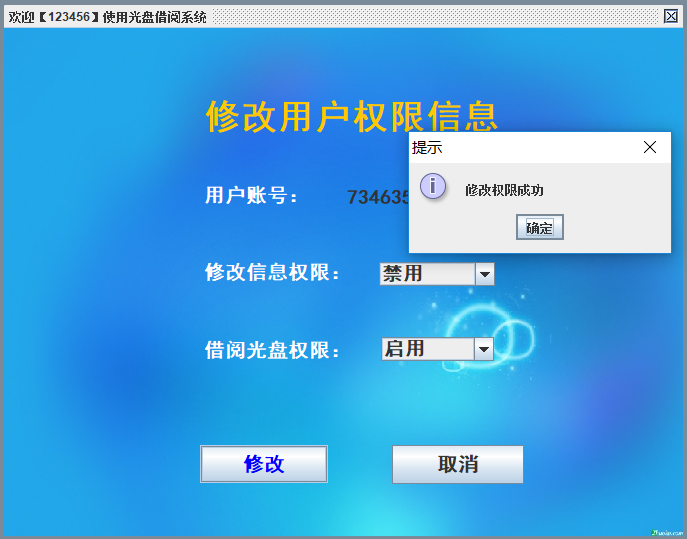
圆饼图统计光盘借阅情况 查询所有用户的借阅日志

查询指定的用户日志 查看用户权限



修改用户权限 用户权限申请



锁定

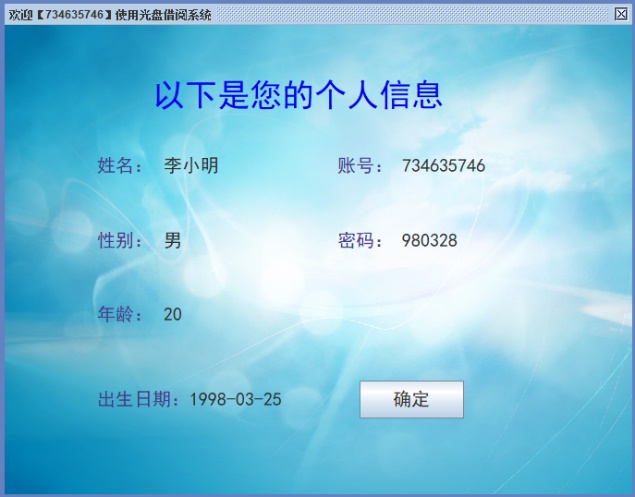


用户模块测试结果

账户错误 密码错误



登陆成功 查看个人信息



修改个人信息 借阅光盘



归还光盘 按编号查询光盘



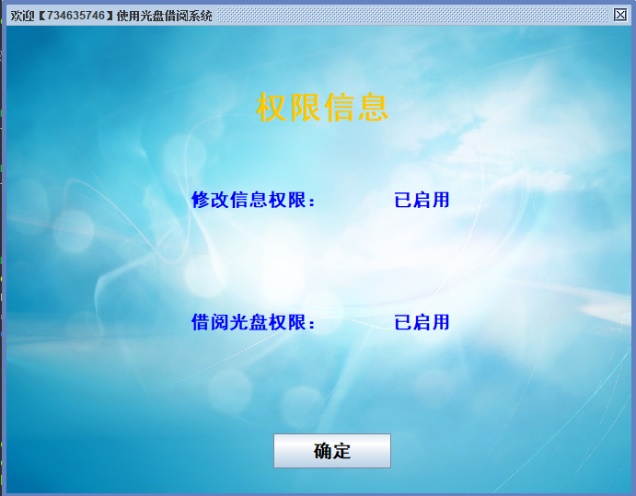
按名称查询光盘 按类型查询光盘



查看可借阅光盘 查看已借阅光盘



查看个人权限 申请权限



查询借阅日志 锁定



以上为程序实际运行结果截图，与预计结果基本一致。

设计体会

通过本此课设，我进一步掌握了java的基本语法与高级特性；对面向对象的思想有了更进一步的了解。同时更加熟练地使用数据库进行开发，锻炼了发现问题、解决问题的能力。

本次设计中，我首先对题目要求做了长时间的分析与架构规划，将程序分成两大模块进行开发。针对架构图进行分包与类的设计，在类的设计上运用封装多态的思想进行设计，注重界面层与逻辑层的解耦。同时根据功能需求进行数据库表的设计，在设计过程中注重开发效率与用户体验，进行合适的表设计，再用关联表进行设计提高开发效率。在开发中，注重面向对象思想，利用封装、多态的思想进行开发，提升代码的复用性。在界面设计中注重用户体验，例如增加了按钮动态交互，增加背景图片，改变窗口风格。

在本次设计中还存在着一些不足。例如有的代码没有很好地做到解耦与复用；在界面设计上组件过于单一；在维护数据库连接时没有采用连接池进行管理。

总体来说，这次设计是一次非常好的锻炼机会，在这次开发中收益良多。

## 附录：源代码

bean

Configuration.java

core

DBManager.java

po

CD.java

User.java

Manager.java

Personnel.java

Record.java

Root.java

window

LoginWindow.java

ManagerLoginWindow.java

ManagerOperateWindow.java

UserLoginWindow.java

UserOperateWindow.java

util

ConversionUtil.java

DBUtil.java

JDBCutil.java

QueryUtil.java

StringUtil.java

ManagerDao

checkAccountAndPwd.java

managerOperateInt.java

ManagerOperation.java

ManagerQuery.java

UserDao

checkAccountAndPwd.java

UserOperate.java

userOperateInt.java

UserQuery.java

Image

po.porperties