**齐鲁工业大学课程设计专用纸** 成绩

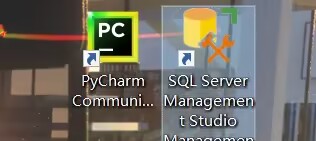
**课程名称 数据库原理课程设计 指导教师 姜合**

**院 （系） 计算机科学与技术学院 专业班级 计科21-2**

**学生姓名 安玉坤 学号 202103010056 设计日期 2022.6.15**

**课程设计题目 图书借阅管理系统**

1. **软件环境及开发工具**



 1. PyCharm Community Edition 2022.2.32. SQL Server Management Studio Management Studio 19

**二、设计要求**

**1、 目的和要求**

实现现实中工作人员与用户对于图书借阅的相关操作

**2、主要内容**

实现管理员账号密码登录注册，书籍的借阅归还，超时罚款缴费，查询书籍未及时归还书籍人员等信息。

**三、需求分析**

用户管理：系统需要支持注册、登录功能。，管理员可以管理图书、借阅记录等信息。

图书管理：系统需要支持图书的添加、删除、修改、查询等功能。每本图书需要包含图书编号、书名、作者、出版社、出版时间、分类等基本信息。管理员可以对图书信息进行编辑和维护，并且可以根据需要将图书进行分类。

借阅管理：系统需要支持用户对图书的借阅、归还操作。借阅时需要记录借阅者、借阅时间、归还时间等信息。

界面友好：系统需要具有良好的用户界面，操作简单易懂，方便用户进行操作。

扩展性强：系统需要支持扩展功能，例如支持多种语言、支持不同的数据库等。

**四、总体功能设计**

Return\_book用户输入书籍编号、借书者编号和还书时间后，系统会检查该书是否被该读者借阅，并且尚未归还。如果借阅记录存在，则更新该记录的归还日期，并且计算罚款金额（如果有逾期）。同时，系统会生成一条罚款记录，并将罚款状态设置为“未缴纳”。如果没有借阅记录，则提示用户该书未被借出。最后，系统会关闭还书界面。

Borrow\_book用户输入书籍编号、借阅者编号和所借本数后，系统会查询该书的库存数量和该读者当前已经借阅的书籍数量。如果借阅数量超过了该读者的借阅限制或者库存数量不足，则系统会分别提示用户。如果借阅数量不超过限制并且库存数量充足，则系统会更新该书的库存数量，并且插入一条借阅记录到 BorrowRecord 表中。最后，系统会弹出一个提示框，告知用户借阅成功，并关闭借书界面。

search\_bookinfo1() 函数定义了一个嵌套函数 search\_book\_()，用于查询符合条件的书籍信息。在 search\_book\_() 中，我们首先获取用户输入的出版社名称，然后使用 SQL 查询语句从 Book 表中查询符合条件的书籍信息，并将查询结果打印出来。在主函数中，我们首先创建一个 Tkinter 窗口，并设置窗口的标题和大小。然后，我们添加一个输入框和一个按钮，用于接收用户输入和触发查询操作。在按钮被点击时，会调用 search\_book\_() 函数来进行查询操作。最后，我们添加一个显示区域，用于显示查询结果。最后，进入主循环，等待用户的操作

search\_readerinfo1(),search\_readerinfo2(),search\_bookinfo2()和search\_bookinfo1()相似

add\_reader()增加读者信息

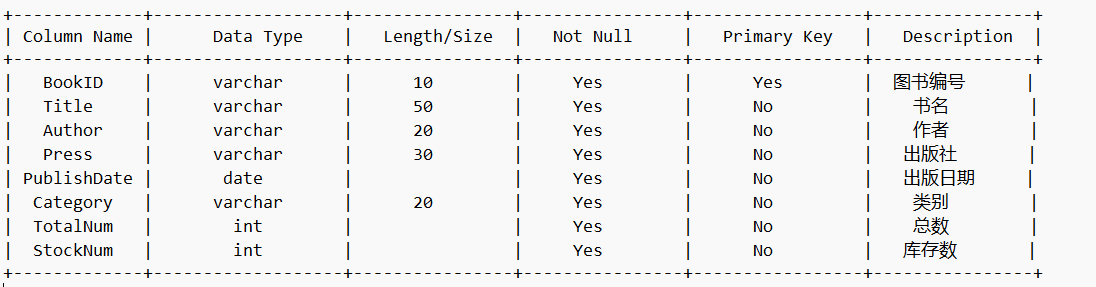
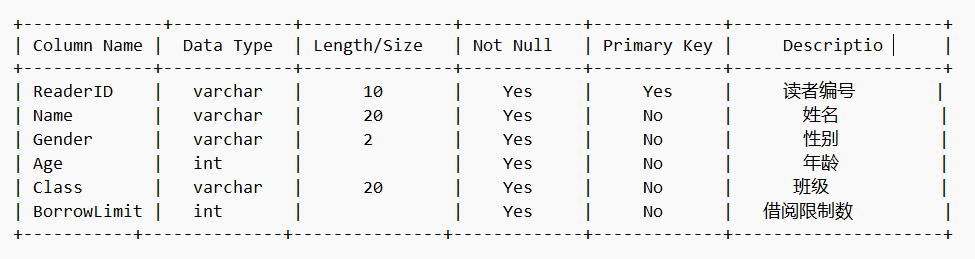
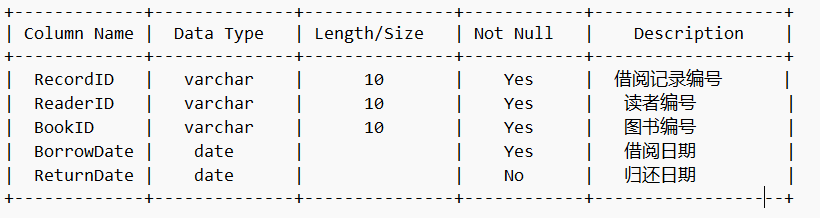
1. **数据库设计**

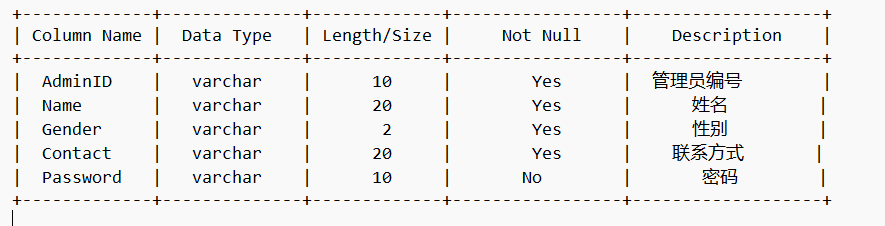
1. 概念结构设计

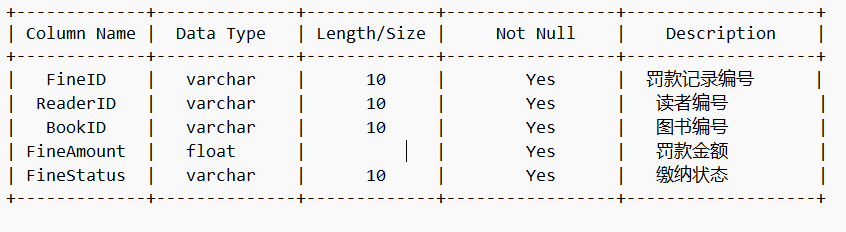
* 关系：
  + 图书（Book）和借阅记录（BorrowRecord）之间是一对多的关系，一个图书可以被多个读者借阅，一个借阅记录对应一本图书。
  + 图书（Book）和管理员（Admin）之间是多对多的关系，一个图书可以被多个读者借阅，一个管理员管理多本图书。
  + 读者（Reader）和借阅记录（BorrowRecord）之间也是一对多的关系，一个读者可以借阅多本图书，一本图书可以被多个读者借阅。
  + 读者（Reader）和罚款记录（FineRecord）之间也是一对多的关系，一个读者可以有多条罚款记录，一条罚款记录对应一个读者。

2. 逻辑结构设计

* 数据库表设计：

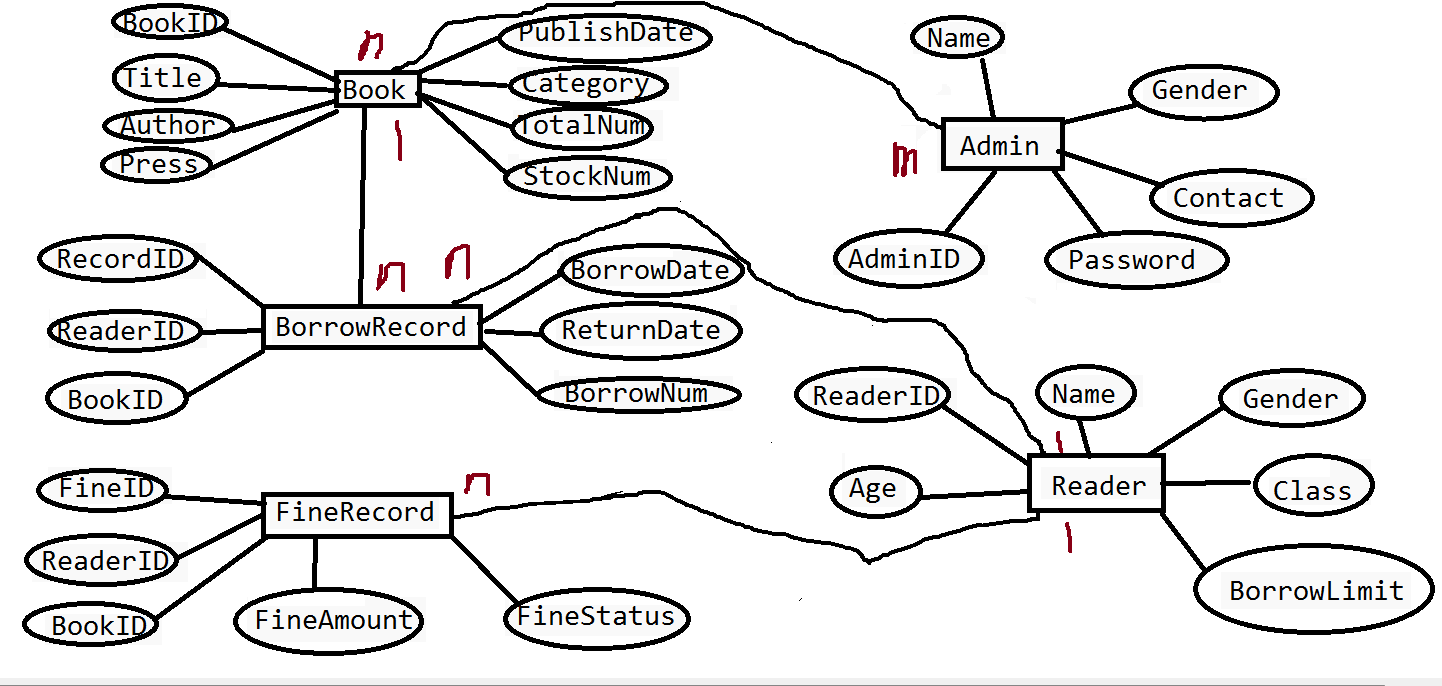
1. Book表：2.Reader表：3.BorrowRecord表：

4.Admin表：

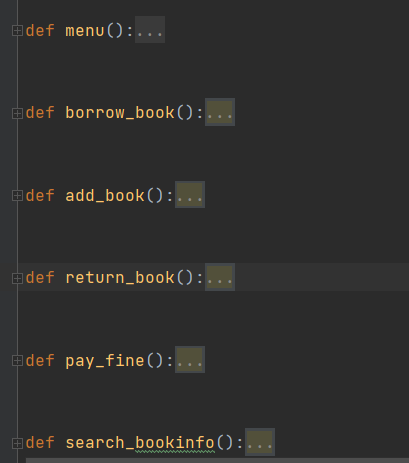
5.FineRecord表：

* 关系设计：Book表和BorrowRecord表之间是一对多的关系，一个图书可以被多个读者借阅，一个借阅记录对应一本图书。在BorrowRecord表中加入BookID作为外键，建立关系。
  + Reader表和BorrowRecord表之间也是一对多的关系，一个读者可以借阅多本图书，一本图书可以被多个读者借阅。在BorrowRecord表中加入ReaderID作为外键，建立关系。
  + Reader表和FineRecord表之间也是一对多的关系，一个读者可以有多条罚款记录，一条罚款记录对应一个读者。在FineRecord表中加入ReaderID作为外键，建立关系。

一次只能借一种书，还的时候一起还



**六、应用程序设计**



借书需要先根据BookID查找库存，库存大于所借本数才可以

还需要满足已经借到的本数加上这次借的数量大于能借的最大值

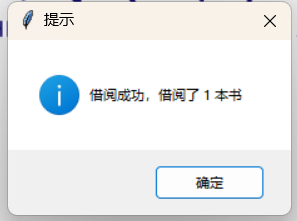
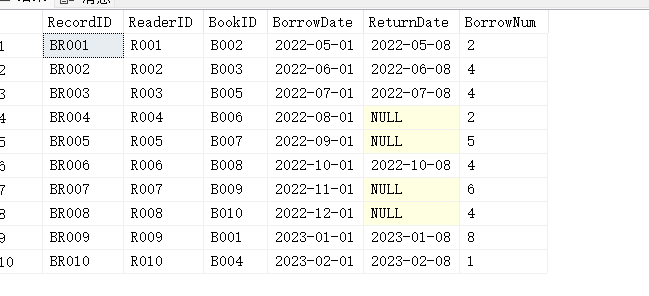




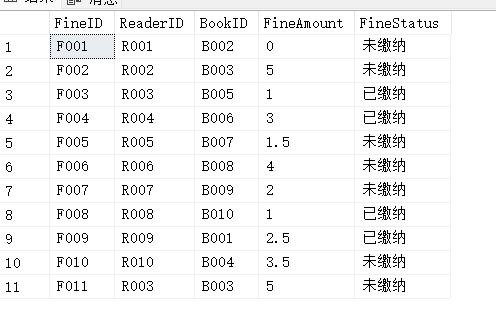
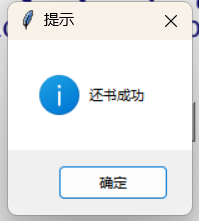
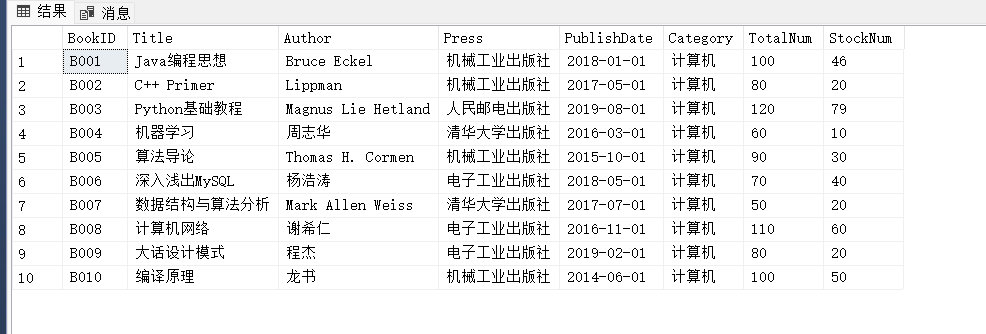
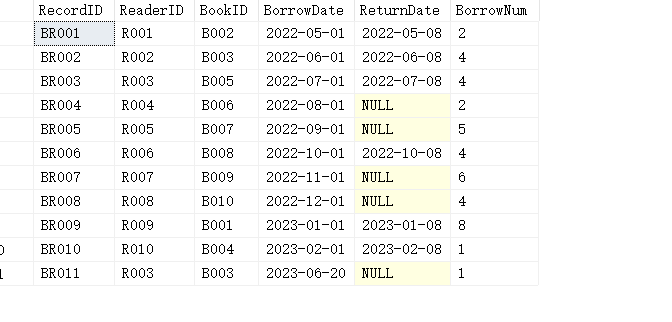


还书需要查找未归还的所有记录，需要判断借阅时间是否大于30天，大于则每本每超一天0.5元，插入FineRecord 

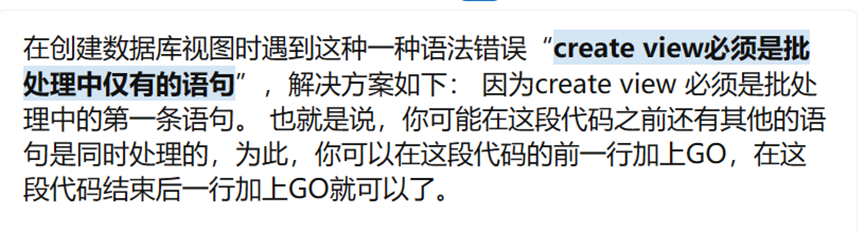
**七、调试运行结果**

以借书为例

接完之后增加相应记录



1. **心得体会**

认为简单的东西执行起来总会有各种奇奇怪怪的bug,纸上得来终觉浅，学完之后一定要动手实践一下，在实践的过程中也会学到和重新想起很多忘掉的知识点

不允许保存更改。您所做的更改要求删除并重新创建以下表。您对无法重新创建的表进行了更改或者启用了“阻止保存要求重新创建表的更改"选项。

