**《数据库课程实践报告》**

题 目： 图书管理系统

学生姓名：

学 号：

专业班级：

指导教师：

时 间： 2022.9.5——2022.9.9

分 数：

**二0二二 年 九月**

1. 实践的要求

为某个部门或单位开发一个数据库应用系统，具体内容包括：对某个部门或单位业务和数据进行调查，系统分析，系统设计，数据库设计，数据库创建和数据加载，数据库应用软件开发，系统测试，系统分析设计和开发文档撰写，软件、文档和数据库提交，数据库应用系统运行演示和大作业汇报。

能够针对某个部门或单位的应用需求，通过系统分析，从数据库数据和应用系统功能两方面进行综合设计，实现一个完整的数据库应用系统。同时培养团队合作精神。

二、数据库设计过程

2.1 概要设计

本次实践选择的项目为开发一个面向图书馆的图书管理系统，其功能应该满足一般用户的借阅和归还书籍需求，同时为管理员提供方便管理用户和图书的功能。

2.2 详细设计

2.2.1 概念结构设计

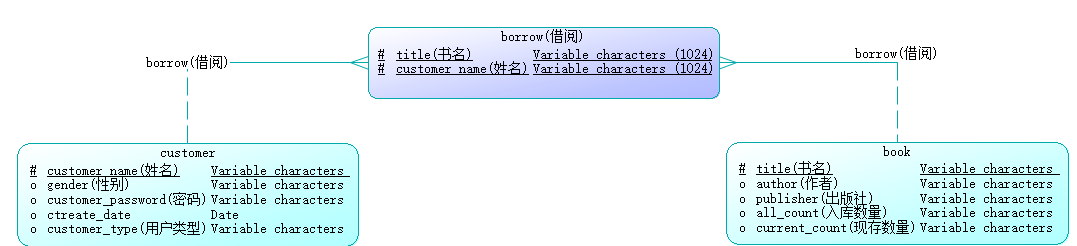
根据实际分析，本次数据库模型需要用到两个实体----书籍(book)实体和用户实体(customer)。书籍实体包含的属性有书名、作者、出版商、入库数量、现有数量。用户实体包含的属性有姓名、性别、密码、建户日期和用户类型。同时，两者实体之间存在者多对多的borrow(借阅)联系。该模型对于的E-R如下：



根据上述分析，利用PowerDesigner设计的概念模型如下：

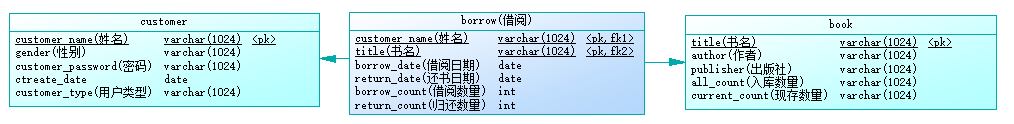


2.2.2 逻辑结构设计

按照数据库设计原理中概念结构转化成逻辑结构的规则，每个实体转换成一个关系，多对多的联系也转换成一个关系。因此，根据上述的E-R图设计数据库逻辑结构。此处使用PowerDesigner将2.2.1中设计的概念模型转换为逻辑模型： 

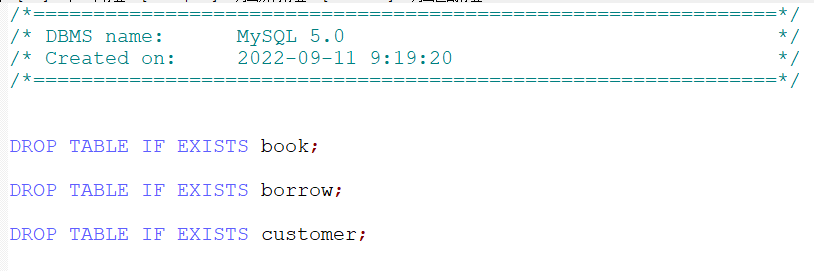
2.2.3 物理结构设计

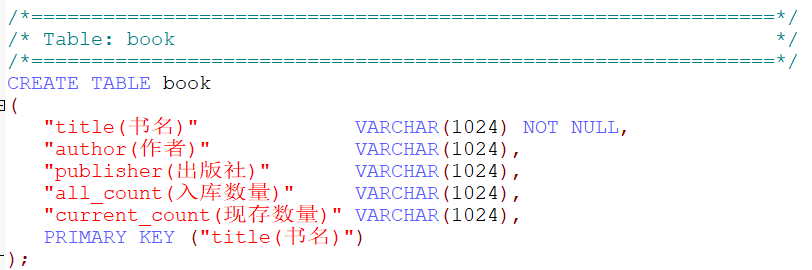
利用PowerDesigner,选择数据库引擎MySQL 5.0,其余设置默认，将2.2.2中的逻辑模型转换成物理模型，结果如下：

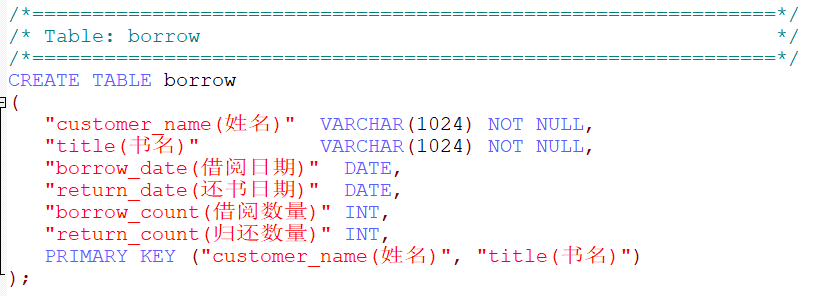


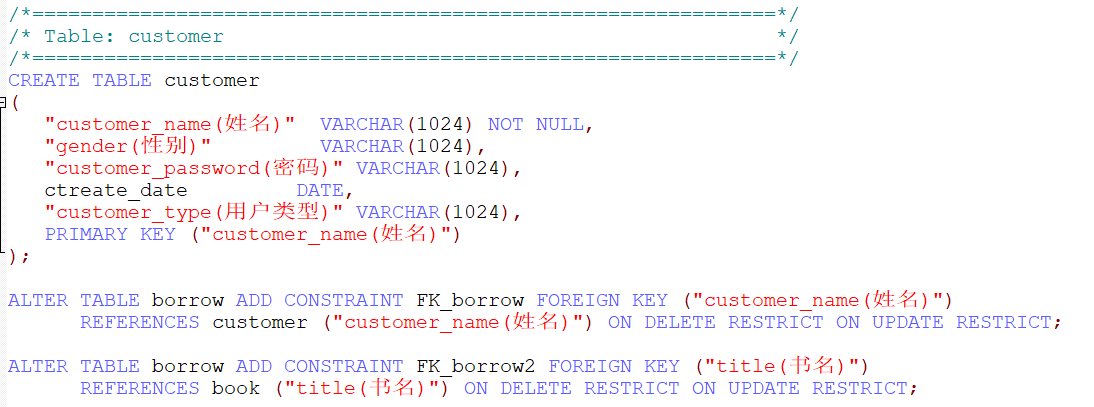
2.2.4 数据库模式SQL语句的生成

利用PowerDesigner自动生成MySql 5.0数据库管理系统SQL语句，部分结果如下:



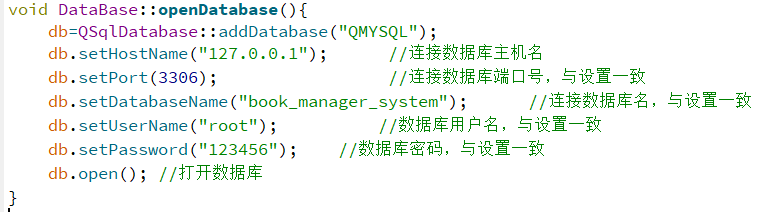






三、编程实现

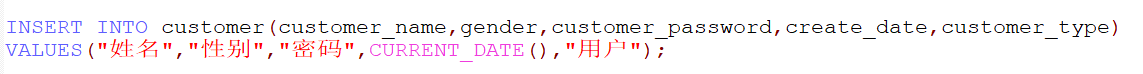
3.1 QT 连接并打开数据库



3.2 用户功能：用户注册



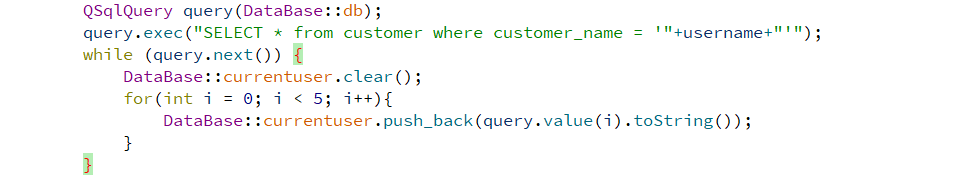
该函数为检查用户输入的姓名以及密码是否有效，如果无效则弹出消息框提示用户错误信息。如果通过了检查，则执行sql语句，在数据库中的customer表插入一条数据，然后提醒用户注册成功。执行的sql语句如下：

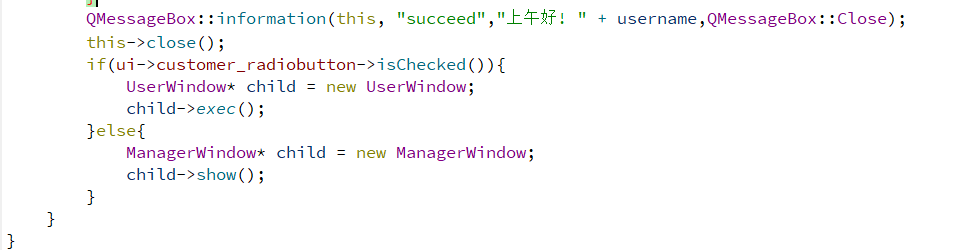


执行过程中，动态将姓名、性别、密码替换为用户的有效输入。

3.3 用户/管理员登录







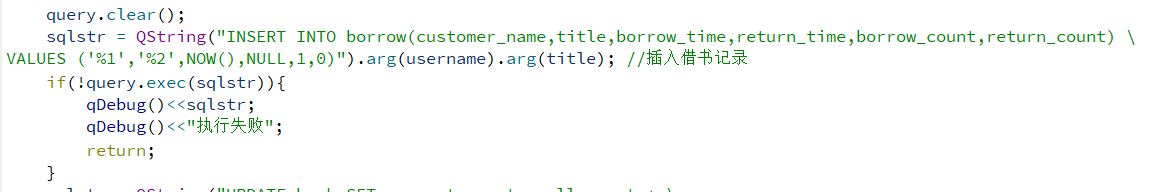
该函数检查用户输入的姓名、密码是否为空，如果不空，检查用户是否选择组别及其是否正确。如果上述均正确，则执行sql语句，将登录的用户信息保存到本地。执行的sql语句如下：

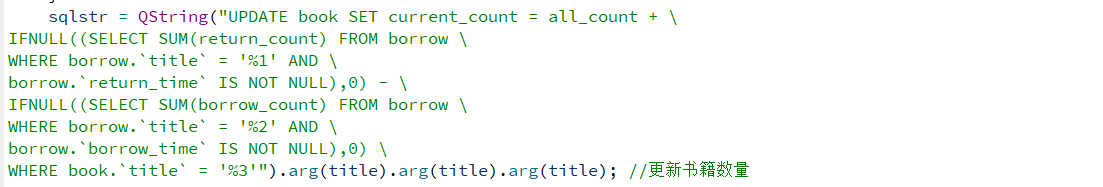


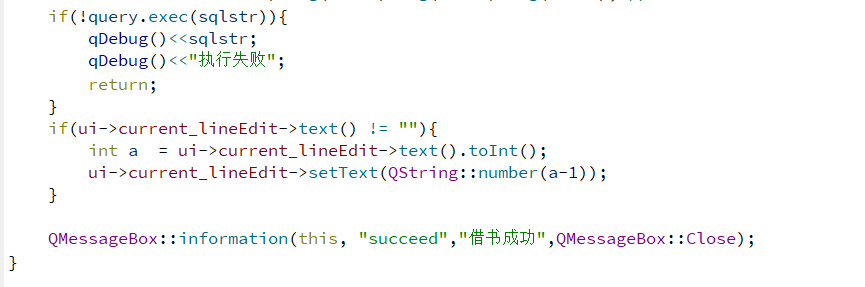
执行过程中，动态将姓名替换为用户的有效输入。

3.4 用户功能：图书借阅





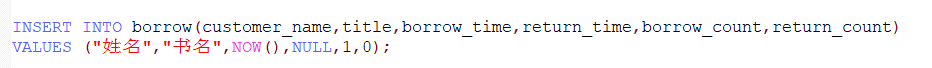


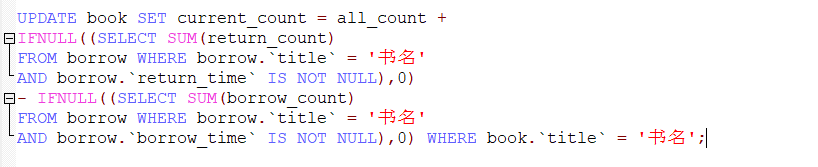


借书总共分为三步----第一步，检查。从数据库中查询用户选择的书籍的剩余量，如果当前书籍用户选择的书籍剩余量不足(即为0)，则提示用户借阅失败。第二步，插入。如果书籍能够借阅，则向数据库插入一条借阅记录。第三步，更新。用户完成借阅后，更新数据库中对应书籍的剩余量。

上述三步涉及的sql语句分别为：

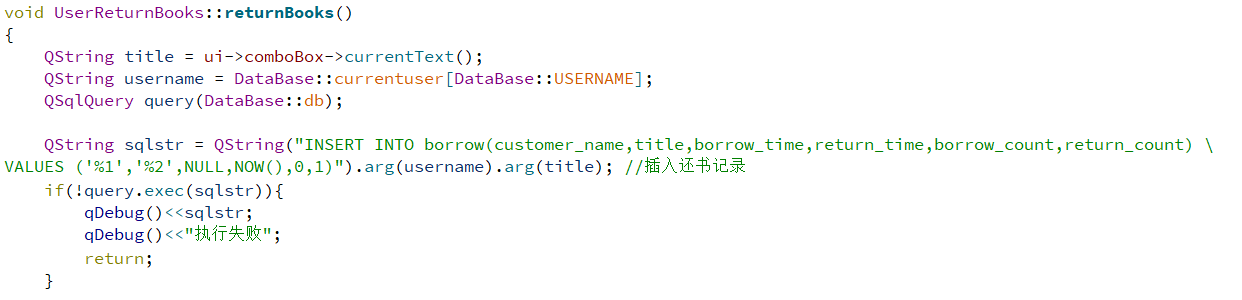


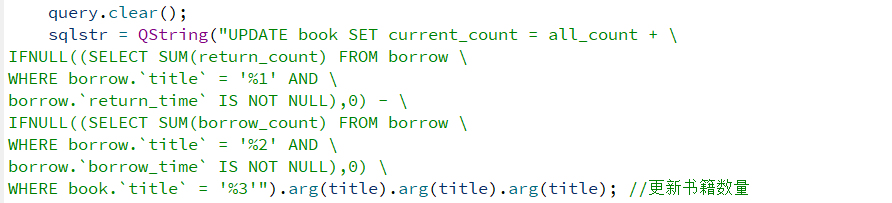


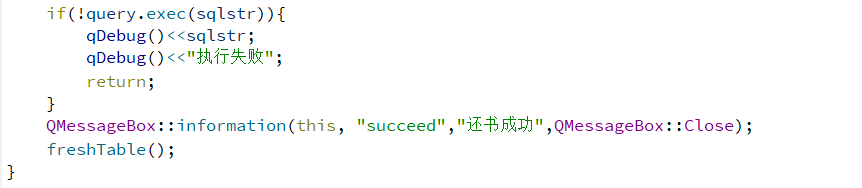


执行过程中，动态将姓名替换为当前用户姓名，书名动态替换为用户选择的书名。

3.4 用户功能：图书归还

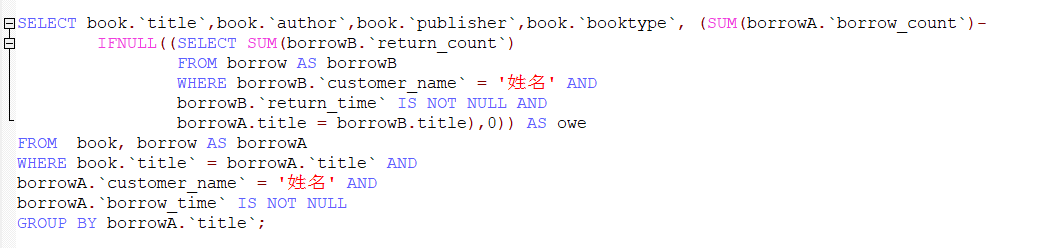


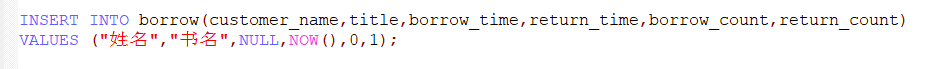


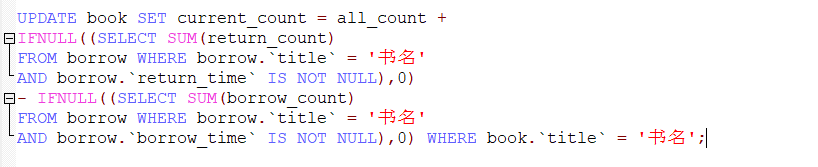


还书流程与借书大致相同----第一步，查询。从数据库中查询当前用户的借阅记录以及计算需要归还的数量。第二步，插入。如果用户选择了归还当前书籍，则向数据库插入一条还书记录。第三步，更新。用户完成图书归还后，更新数据库中对应书籍的剩余量。

上述三步涉及的sql语句分别为：



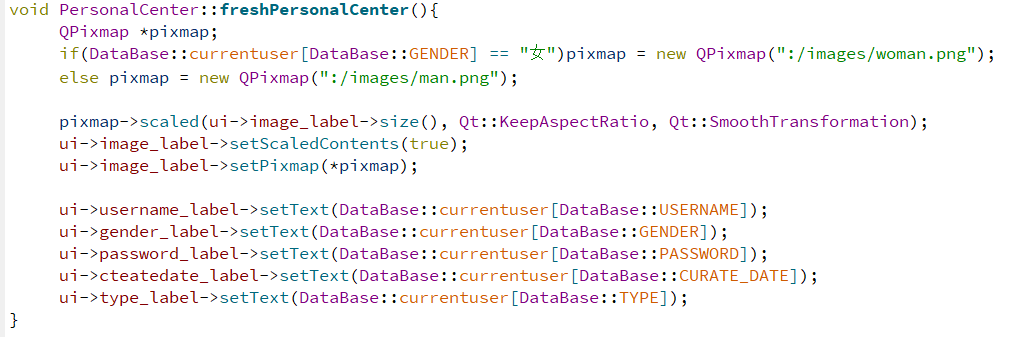




执行过程中，动态将姓名替换为当前用户姓名，书名动态替换为用户选择的书名。

3.5 用户功能：个人信息查询及修改

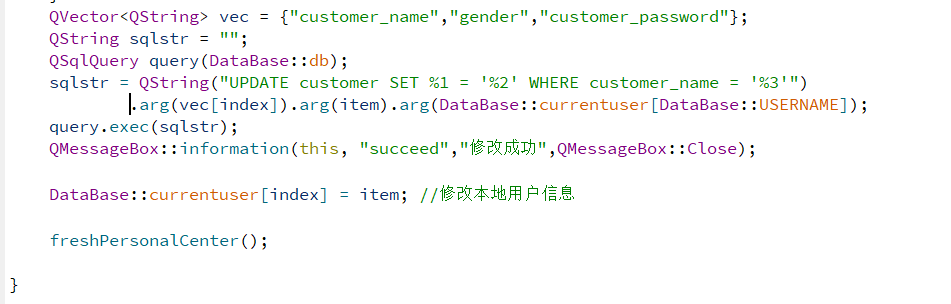
个人信息查询：



由于在用户登录的时候，个人信息已经被加载到了本地，于是此处便不再需通过数据库获取用户信息，直接读取本地信息即可。如DataBase::USERNAME即保存的用户姓名，DataBase::GENDER保存的用户性别，其他信息均类似。

个人信息修改：





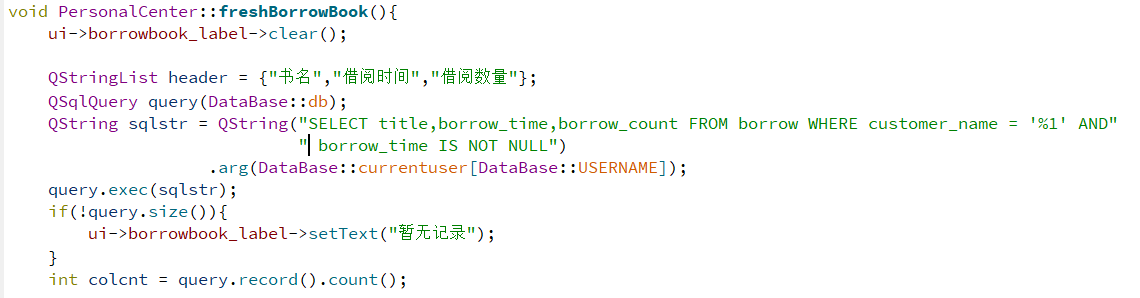
该函数先检查用户输入的合法性，比如输入为空，输入内容过长(长度大于12)或者修改性别时，输入内容不是“男”或“女”。检查完用户输入的合法性后，执行更新语句，更新数据库的个人信息。同时修改本地用户的个人信息并提醒用户修改成功。

更新用户个人信息的sql语句如下：



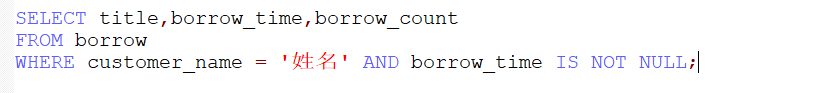
执行过程中，将“属性”替换为用户选择的需要修改的属性，比如姓名、性别等；将“新属性”替换为用户输入想要修改的内容，如将姓名从“张三”修改为“李四”；将“用户名”替换为当前用户的姓名。

3.6 用户功能：借阅记录查询





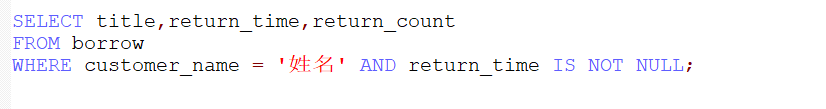
该函数根据根据当前用户的姓名，从数据库中查找该姓名的借阅记录，并将其显示在特定区域。查询过程执行的sql语句如下：



执行过程中，动态将姓名替换为当前用户姓名。

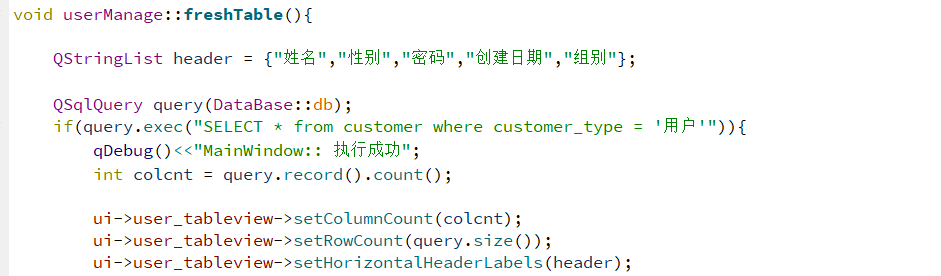
3.7 用户功能：还书记录查询

用户还书记录查询与用户借书记录查询类似，唯一不同的是执行的sql语句不同，此处便不再展示程序代码。查询过程执行的sql语句如下：



3.8 管理员功能：用户管理

查询所有用户信息：





该函数主要用于实现查询当前所有普通用户的所有个人信息，并将其显示在特定位置。用于的查询sql语句为：



执行过程中，动态将姓名替换为管理员选择的姓名。

删除用户：



删除用户首先先获取管理员输入的待删除的用户的姓名，然后执行查询，从数据库中检索该用户，如果未找到，则提醒管理员用户不存在。否则，在管理员输入的验证码有效的前提下，执行删除语句，删除该用户。该用户的所有信息，如借阅记录，还书记录都将被删除。

查询特定用户的sql语句如下：



删除特定用户的sql语句如下：



执行过程中，动态将姓名替换为管理员选择的用户名。

3.9 管理员功能：图书入库





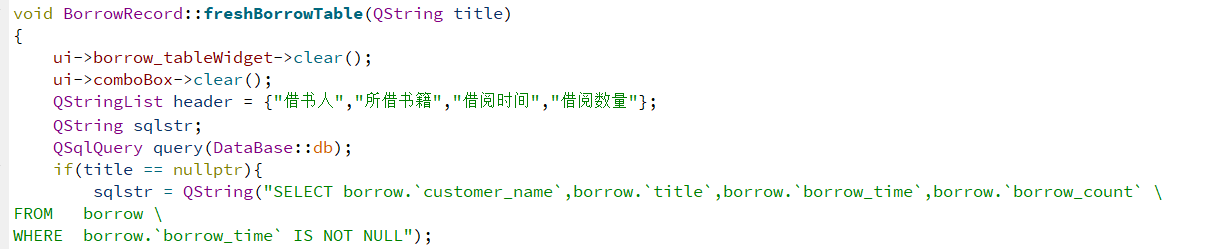
该函数主要用于检查管理员输入的图书信息是否有效----图书名、作者、出版社、入库数量、图书类型均不能为空。特别的，图书名不能为书库已有的书籍(想要增加已有图书数量，可以选择修改图书信息，不过该功能怎么未提供)，入库数量也必须为大于0的正整数。如果一切输入均为合法，则执行插入操作，向数据库中的book表中插入一条图书信息。

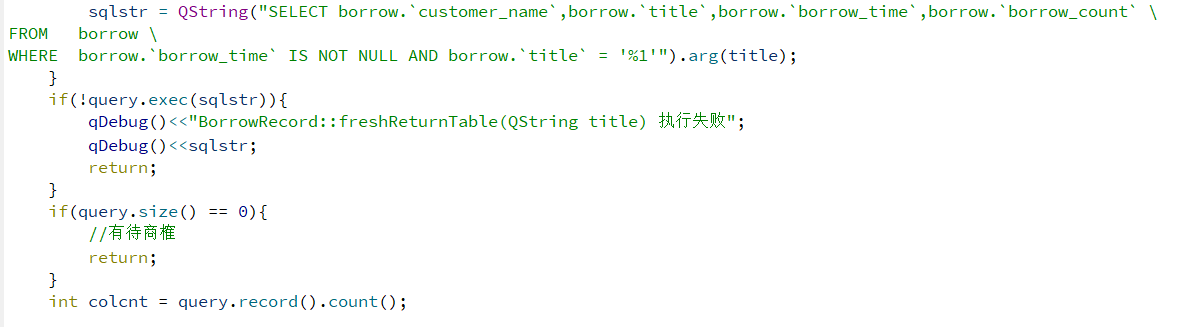
执行插入的sql语句如下：



执行过程中，动态地将各图书信息替换为管理员输入。

3.10 管理员功能：借阅记录查询



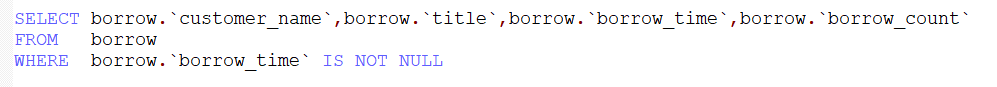




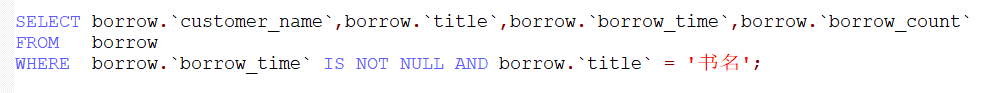
该函数提供两种查询模式----查询所有书籍的借阅情况以及查询特定书籍的借阅情况。

初始进入该界面时，管理员看到的是所有书籍的借阅情况。同时，管理员也可以选择自己想看的特定书籍的借阅情况。

查询所有书籍借阅情况的sql语句如下：



查询特定书籍借阅情况的sql语句如下：

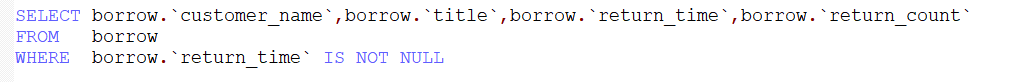


执行过程中，动态地将书名替换为管理员所选择的书籍名。

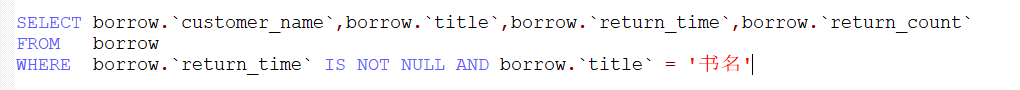
3.11 管理员功能：归还记录查询

归还记录查询与借阅记录查询流程相似，同样提供两种查询模式----查询所有书籍的借阅情况以及查询特定借阅的借阅情况。两者代码也相似，唯一不同的是执行的sql语句不同，此次便不再重复展示程序执行代码。

查询所有书籍还书情况的sql语句如下：



查询特定书籍借阅情况的sql语句如下：



执行过程中，动态地将书名替换为管理员所选择的书籍名。

1. 测试结果

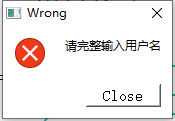
4.1 用户功能：用户登录(主界面)

界面样式

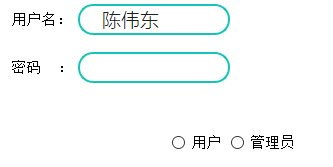
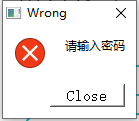


用户登录测试

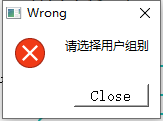
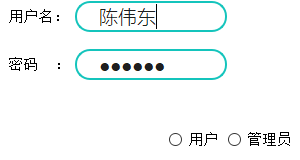
用户名输入为空时，提示：



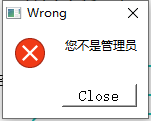
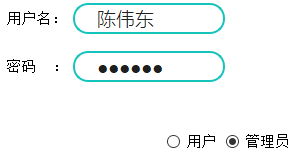
密码输入为空时，提示：

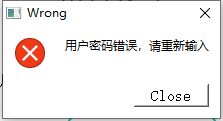
未选择组别时，提示：

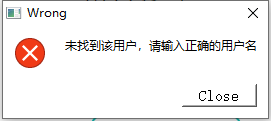
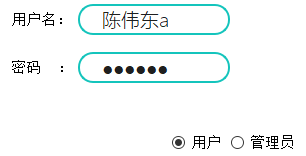


组别选择错误，提示：

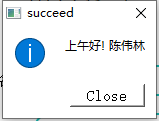
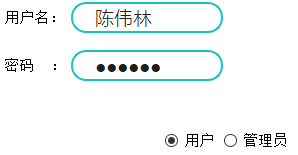


组别选择正确，但用户名或者密码错误提示：





登录成功，提示登录成功：

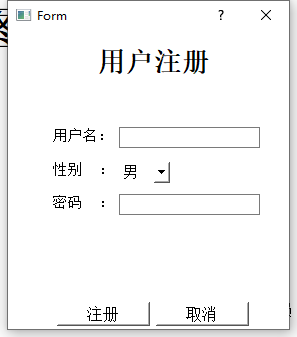


同时进入用户界面：



4.2 用户功能：用户注册

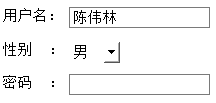
如果用户并未用户账户，则可以点击主界面右下注册按钮，弹出注册界面。界面样式：



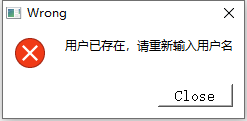
用户注册测试

信息不完整，提示：

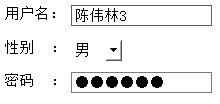
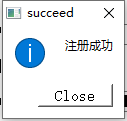


重复注册，提示



用户注册成功，提示：

同时关门注册窗口，返回主界面

4.3 用户功能：图书借阅

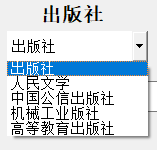
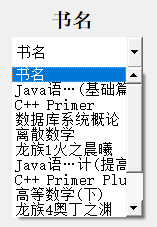
点击用户界面的“图书借阅”按钮，将进入图书借阅界面。



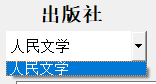
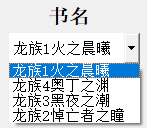
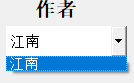
图书借阅界面样式



该界面初始时加载所有图书信息，下拉”类别”，“书名”，“作者”，“出版社”五个选项框，可以看到图书的各项信息。如：

当用户选择某个属性，点击“筛选”按钮，可以筛选满足特定属性的图书。如，下拉“类别”选项框，并选择“小说”类别，然后点击“筛选”按钮，得到：

结果显示，当前书库中类别为“小说”的图书只有4本，作者均为“江南”，出版社也只有“人民文学”出版社。其他属性筛选同理，此处便不错展示。点击“重置”按钮，则回到最初始状态，加载所有图书信息。

该借阅界面同时提供搜索功能，匹配用户的输入。其搜索逻辑为，先进行完全匹配，如果搜索不到则进行以用户输入为开头的不完全匹配，如果搜索不到，则匹配用户输入的任意一个字符。如果仍然搜索不到，则提示没有搜索结果。

如，输入“离散数学”，点击“搜索”按钮，结果如下：



当搜索结果只有一条时，将会加载该图书的入库数量(总量)以及剩余量。

如，输入”离散”，点击“搜索”按钮，结果如下：



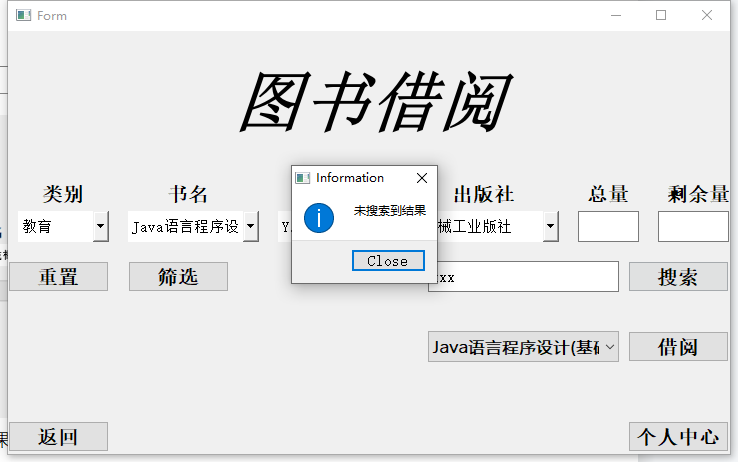
可以看到搜索结果仍为离散数学。

如，输入“数学”，点击“搜索”按钮，结果如下：



可以看到搜索结果有四本不同类型的书籍。

如，用户输入“xxx”，搜索结果如下：



用户进行筛选或者搜索功能，获得到结果时，“借阅”按钮旁边的结果框会同时出现搜索结果。选择想要的书籍，点击“借阅”，便能进行借阅。

如，在搜索框中输入“离散数学”，点击“搜索”按钮，然后看到“借阅”按钮旁边的结果框出现“离散数学”，点击“借阅”按钮，便借阅结果：





可以看到“剩余量”由50变为49，说明借书成功。

4.4 用户功能：图书归还

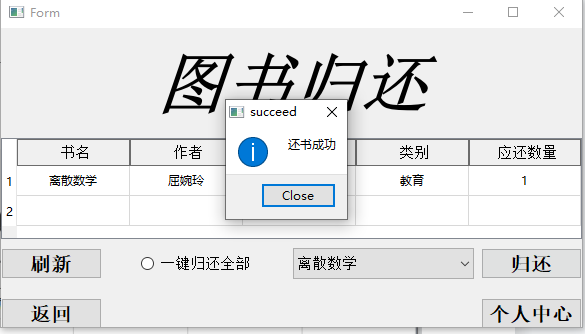
点击用户界面的“图书归还”按钮，将进入图书归还界面。

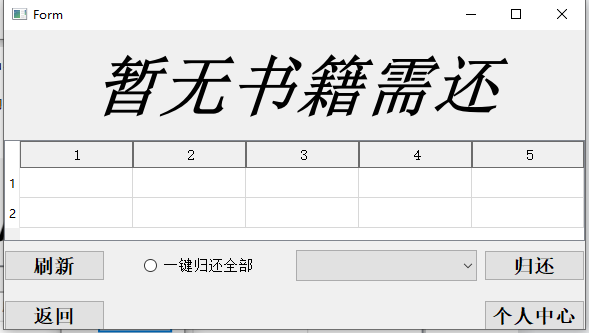


图书归还界面样式



用户进入后将会出现用户借阅的书籍情况及应当归还的数量。“归还”按钮旁边的结果下拉框显示所有需要归还的图书书名。选中书名，点击“归还”按钮即可归还。

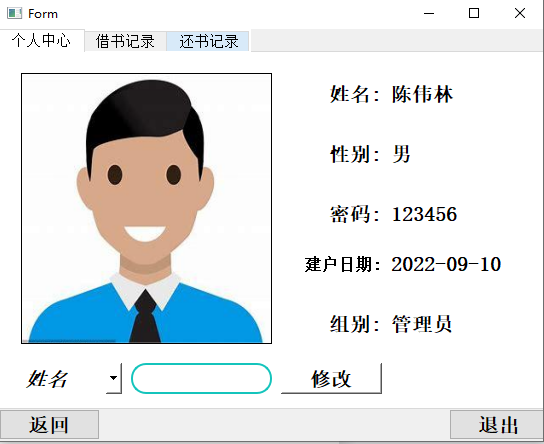




可以看到，当用户归还了所借的唯一一本“离散数学”后，归还界面刷新，提示用户暂无书籍需还。

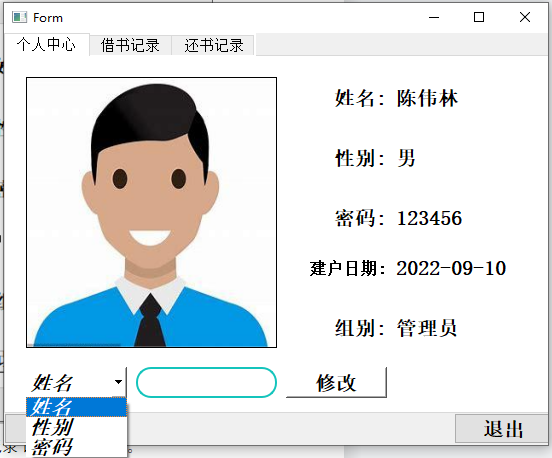
4.5 用户功能：个人信息修改、查询及借阅情况

程序各个界面右下均匀“个人中心”按钮，点击该按钮，进入个人中心，可以查看或者修改个人信息。界面样式：

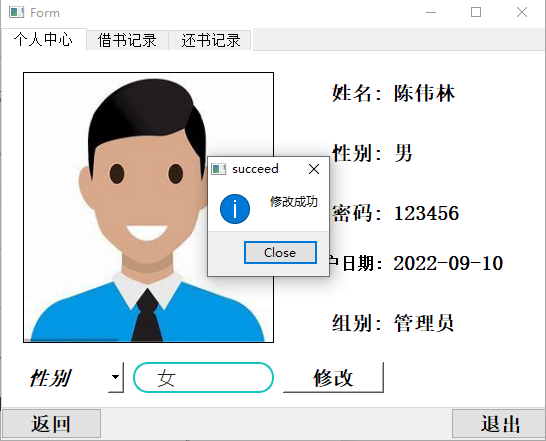


个人中心有三个页面，分别为“个人中心”、“借书记录”以及“还书记录”。

个人中心可以查看个人信息，以及修改个人信息。左下侧有用户修改属性选择框，下拉选择框，可以看到改变的属性。



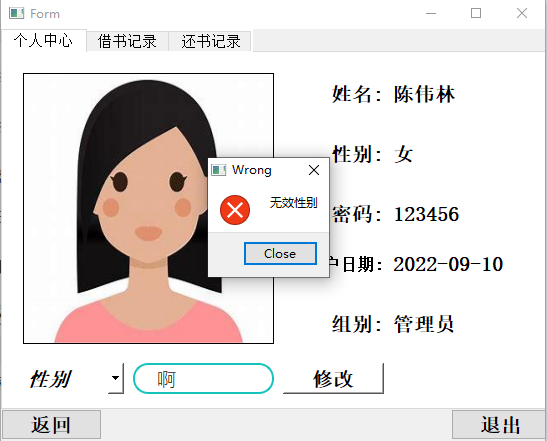
如，选择“性别”属性，输入“女”，点击“修改”按钮，执行修改：



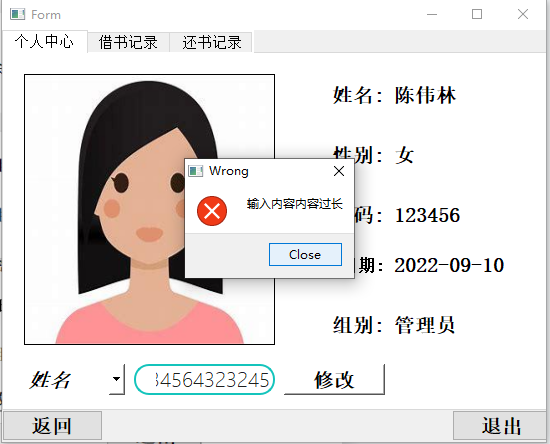


当然，程序将会对输入进行判断。

如用户修改性别时，输入的不为“男”或者“女”，则会出现提醒拒绝此处修改。



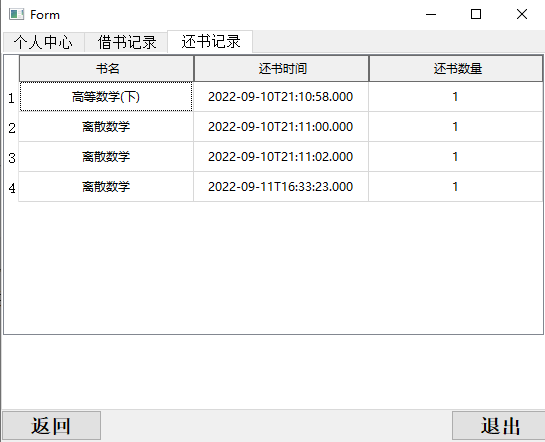
如用户修改姓名或者密码时输入字串长度大于12，则出现提醒拒绝此处修改。



用户点击界面顶部的“借书记录”按钮，便能查看借书记录：



用户点击界面顶部的“还书记录”按钮，便能查看还书记录：



可以通过此表查看上述演示时归还情况，在2022-09-11时进行16：33分时归还。

4.6 管理员功能：管理员界面

当管理员在主界面登陆时，选择“管理员”，则进入管理员界面，界面样式：



4.7 管理员功能：用户管理

管理员在管理员界面，点击“用户管理”，进入“用户管理”界面。界面样式：



在次可以看到用户组别为“用户”的所有个人情况。管理员可以选择删除用户，用户的所有信息均会被删除。当然，在删除之前，程序会对管理的输入进入判断，如果数据库中没有待删除的用户或者验证码错误，则程序会拒绝删除：





只有管理员正确输入时，删除才能执行：





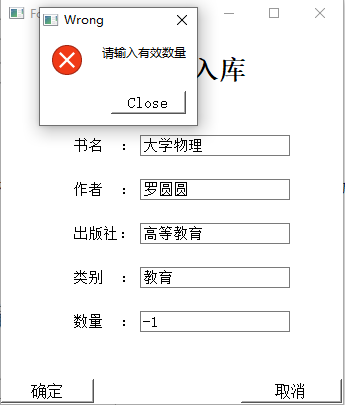
可以看到，用户“肖晓红”已被删除。

4.8 管理员功能：图书入库

管理员在管理员界面，点击“图书入库”，弹出“图书入库”界面。界面样式：



程序将会对管理员输入信息进行严格检查，如图书某项信息不完善或者入库数量为非有效字符：

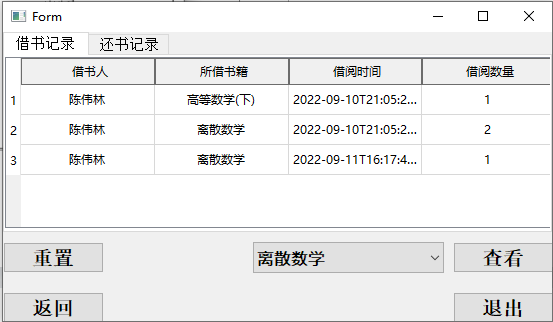
   

如果各项检查均通过，则提示管理员入库成功：



4.9 管理员功能：借阅/归还记录查询

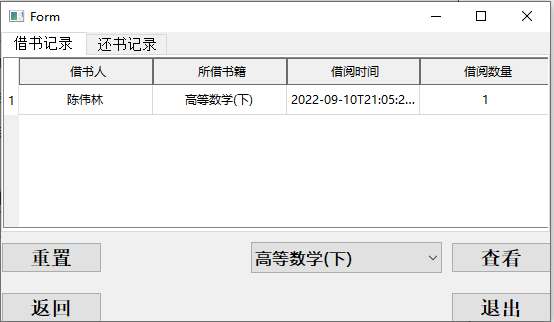
管理员在管理员界面，点击“借阅记录”，弹出“借书记录”界面。界面样式：



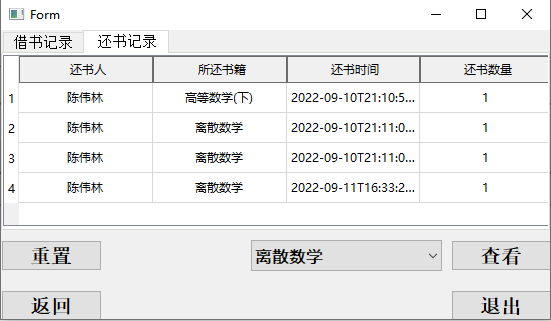
初始进入时，程序加载所以书籍的借阅记录。只要有书籍被借阅，便会显示在表格中。“查看”按钮旁边的结果框显示被借阅的书籍，下拉可以选择特定数据的借阅情况。比如选择“离散数学”，点击“查看”按钮：



点击“重置”按钮，重新加载所有书籍情况。选择“高等数学(下)”，点击“查看”按钮：



管理员点击顶部的“还书记录”按钮，便可以查询书籍的还书情况。



“还书记录”页面的操作方法及功能同“借书记录”页面。此处便不再赘述。

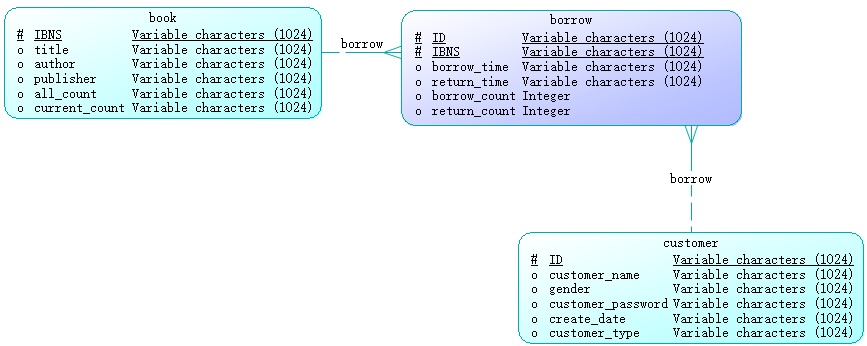
1. 问题和讨论

问题一：系统设计不够完善。

书籍表(book)以书名做为主键，那么如果存在两种同名的书，它们作者或者出版社等其他属性不同，管理员执行图书入库操作时，系统将会拒绝该操作，因为主键冲突，违反了数据库的实体完整性。但在现实中，这是合理的，确实存在大量同名书籍。

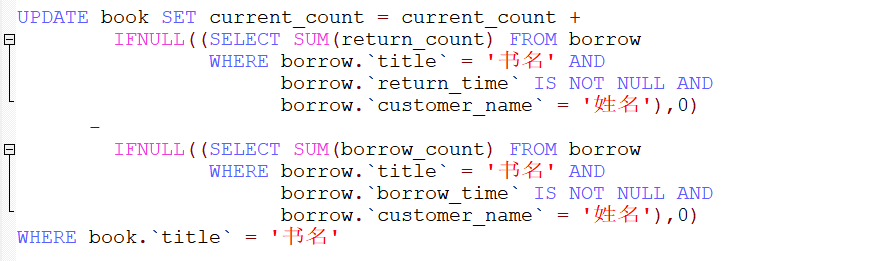
同样的，用户表(customer)以姓名作为主键，那么当一个与系统用户同名的人注册账户时，系统仍然会拒绝此次操作。上述两个问题是本人在设计数据库系统时未考虑到的，也是此次课程作业的缺陷。改进方法如下：

书籍表(book)以ISBN号作为主键，用户表(customer)以借书证号作为主键。其逻辑模型如下：



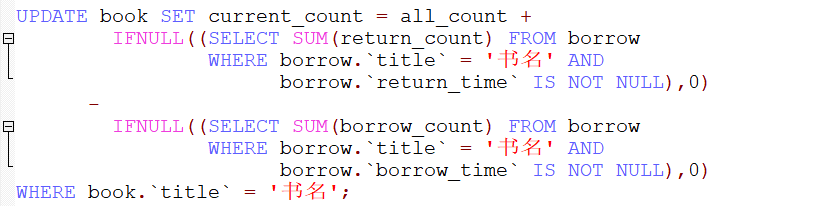
问题二：还书时数据更新的sql语句逻辑出错

在用户完成还书后，需要执行sql语句，更新书籍的数量。最初时，书籍时更新sql语句如下：



其逻辑为，当前书籍数量 = 当前书籍数量 + 当前用户归还该书的数量 - 当前用户借阅该书的数量。ifnull(exper1,exper2)判断exper1是否为空，是则用exper2代替。使用该函数主要是当用户没有归还图书记录时，将其查询结果置为0。

当程序内部套用该sql语句时，就会出现一个“有趣”的bug----用户归还图书时，图书数量越还越少。因为(当前用户归还该书的数量 - 当前用户借阅该书的数量)的执行结果始终不会大于0。正确语句如下：



其逻辑为，当前书籍数量 = 当前书籍的入库数量 + 所有用户归还该书的数量 - 所有用户借阅该书的数量。

六、总结体会

通过这次实验，我对于mysql的嵌入式开发与odbc语言的运用有了更深的认识，也渐渐认识到开发一款可视化数据库应用软件的重要性与难度，为日后的软件开发奠定了基础。此外，我对于QT图形库的运用有了更加熟练的操作。