

**江 苏 科 技 大 学**

**图 书 管 理 系 统**

**（安装部署手册）**

学 院： 计算机学院

专 业： 计算机科学与技术

小组成员： 周 瑞 192210701143

朱庆壁 192210701144

指导教师： 张明

**2023年2月**

目录

[1 创建数据表 3](#_Toc126427330)

[1.1 书籍信息表 3](#_Toc126427331)

[1.2 交易信息表 5](#_Toc126427332)

[1.3 用户信息表 5](#_Toc126427333)

[2 安装Maven 6](#_Toc126427334)

[2.1 从官网下载 6](#_Toc126427335)

[2.2 一些建议修改的maven配置 6](#_Toc126427336)

[2.3 需要的依赖 7](#_Toc126427337)

[3 安装tomcat 8](#_Toc126427338)

[3.1 从官网下载tomcat 8](#_Toc126427339)

[3.2 在intellij idea中配置tomcat 8](#_Toc126427340)

[4 其他 10](#_Toc126427341)

[4.1 jdk版本 10](#_Toc126427342)

[4.2 前端服务器 10](#_Toc126427343)

# 创建数据表

该项目使用的是MySQL数据库，利用mybatis框架实现Java和MySQL的交互。因此为了使项目能够在本地正常运行，第一步便是创建项目需要的数据表。

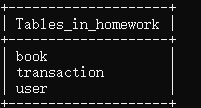
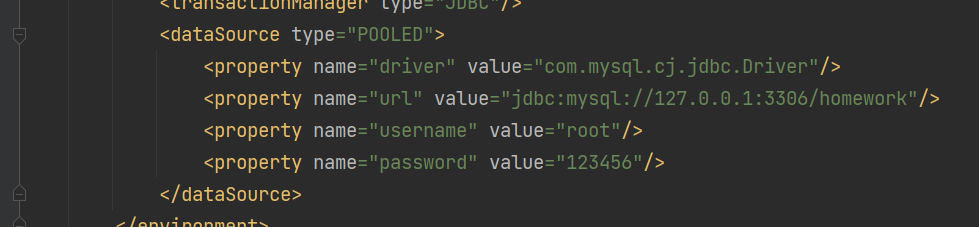
本项目共使用了三个数据表，如图1.1所示。创建表时请确保表名和图片中所示的表名保持一致，否则需要对项目中代码进行大范围修改。另外容纳以上三个表的数据库名应为“homework”（不含引号），数据库的跟用户名为“root”（不含引号），登录密码为“123456”。如图1.2所示，改图展示的是此项目中mybatis的关于数据库的部分配置文件。

图1.2

图1.1

## 书籍信息表

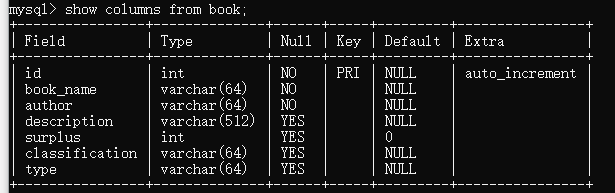
接下来分别给出三个表的属性，首先是书籍信息表，如图1.3所示。该表共有7个属性，其中有三个属性值得注意，surplus描述的是目前图书馆中还剩余的该书籍的数量。读者的借书、还书操作以及管理员的添加数据、修改数据操作都会影响到该属性的值。classification表示的是该书籍属于哪些分类。该项目中一共有5种书籍分类选项，每种书籍可同时拥有多种分类属性，分别是“哲学”、“科学”、“政治”、“军事”以及“经济“。从图中可以看到该属性的类型是长度上限为64的字符串类型。存储方式采用字符串拼接的方式，若一本书同时属于”哲学“、”政治“和”军事“三个分类，则其classification属性对应的值为”哲学&政治&军事&“（如图1.4所示）。随着后期对书籍该属性的维护发现这种字符串拼接的数据存储方式实际上是非常繁琐的。type属性存储的是该书籍的类型，该系统中共有两种书籍类型，分别是“图书”和“期刊”，每本书籍有且只有其中一种类型。

图1.3

图1.4

相关的属性还可以从图1.5的添加书籍界面看出，分类对应的是复选框，而类型对应的是单选框（不得不说这种命名区分方式实在是太容易让用户混淆了）。

图1.5

## 交易信息表

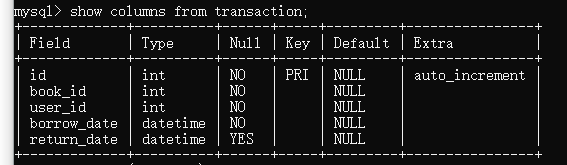
该表的属性如图1.6所示。可以看到该表有5个属性，只记录了书本id，用户id和借还时间，其中属性还书时间的类型是可以为空的，用户刚借书时对应记录的还书时间即为空，但借书时间是不为空的。当用户归还对应书籍时，该系统会找到该用户借书时对应的记录，并修改该记录的还书时间，修改成功后即表示该用户还书成功。

图1.6

另外由于该表中并没有存储书籍名称和用户名称等信息，所有在实际操作时，该表往往要和书籍信息表和用户信息表联合使用。这应该是数据库设计的不合理的地方之一。

## 用户信息表

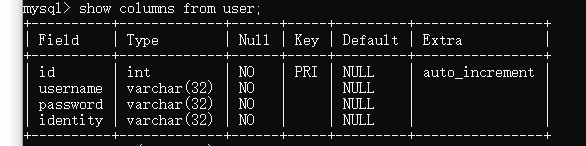
该表的属性如图1.7所示。其中应注意一下identity属性，本系统为用户设计了三种属性，分别时“teacher”，“student”和“administrator”。在用户实际使用时，学生身份和教师身份并无区别，这两种身份无法对书籍进行添加和修改操作，此类操作只有管理员身份用户能进行，不过本系统并没有从前端页面上提供管理员身份的注册功能，因此若想拥有管理员权限只能直接修改数据表内对应用户的身份信息。

图1.7

# 安装Maven

## 从官网下载

我们可以从maven官网下载该软件的二进制文件，这也是本项目下载maven的方式。下载链接是：<https://maven.apache.org/download.cgi>。详细的下载及安装过程此处不再赘述。本项目使用的maven版本为apache-maven-3.8.7。

## 一些建议修改的maven配置

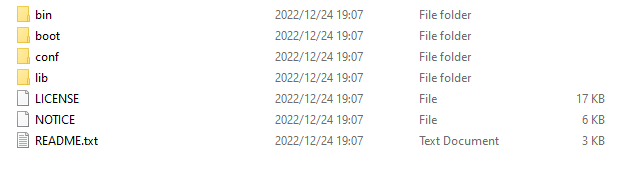
下载并解压后可以得图2.1所示或与之类似的文件结构。其中conf目录下有个我们应该注意的maven配置文件settings.xml。我们使用maven的目的是用其方便地添加我们在项目开发中所需要的依赖，这些依赖需要从maven中央仓库下载，但我们国内用户在访问maven中央仓库时，文件的下载速度总是不尽人意，主要原因是maven中央仓库部署在国外，因此我们为了提升访问速度，可以利用阿里云提供的镜像仓库，只需要在配置文件内添加对应设置，即可将maven默认的仓库地址修改至阿里云的镜像仓库，具体配置代码如图2.2所示。

图2.1

## 需要的依赖

图2.2

需要的依赖有mybatis、junit、javax.servlet-api、mysql-connector-java、以及由阿里云提供的fastjson。

# 安装tomcat

## 从官网下载tomcat

下载链接为<https://tomcat.apache.org/>。本系统使用的tomcat版本为apache-tomcat-9.0.70，有一点请务必注意，就是本系统与tomcat10或以上版本并不兼容，原因可见官网描述，原文内容为：Tomcat 10以后版本的用户应该知道，作为Java EE向Eclipse Foundation迁移的一部分，从Java EE迁移到Jakarta EE，所有已实现API的主包已从javax.\*更改为Jakarta.\*。这几乎肯定需要更改代码才能使应用程序从Tomcat 9及更早版本迁移到Tomcat 10及更高版本。已经开发了一种迁移工具来帮助这一进程。建议使用tomcat9。

## 在intellij idea中配置tomcat

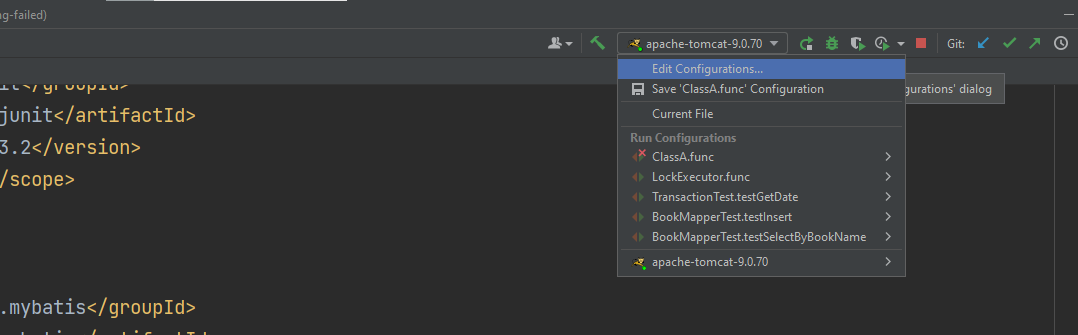
如图3.1所示在IDE右上角可以看到配置选项，点击后弹出面板，选择面板左上角的“+”号，如图3.2所示。选择本地tomcat服务器。在右边面板进行额外对应设置后点击应用和完成即可，详细过程此处不再赘述。

图3.1

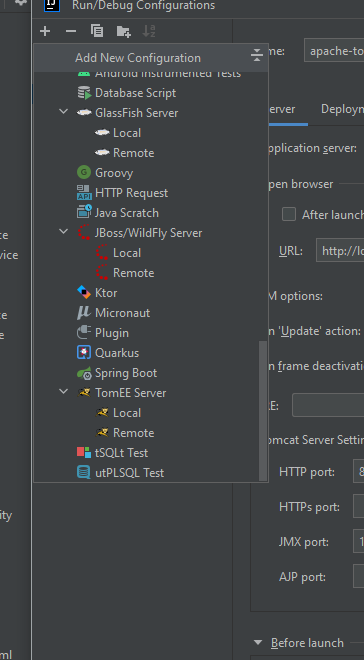


图3.2

# 其他

## jdk版本

如图4.1所示，该项目中使用的是Java 17。下载链接为<https://www.oracle.com/java/technologies/downloads/#java17>。注意可能需要提前将JAVA\_HOME配置到环境变量中。

## 前端服务器

该项目中前端中用到了Live Server本地服务器。可以通过VScode下载。

## 项目代码

项目代码已上传至GitHub，链接为：<https://github.com/maryon23Nov2022/Homework_10_Jan>。