

****

**J I A N G S U U N I V E R S I T Y**

**“ 数据库系统原理 ”课程报告**

学生成绩管理系统

|  |  |
| --- | --- |
| 学院名称： | 计算机科学与通信工程 |
| 专业班级： | 信息安全2102 |
| 姓 名： | 唐瑞阳  范路雨  马天赐 |
| 学 号 | 3210207034  3212502013 |

3210613070

**2023年6月**

# 目录

[目录 0](#_Toc138923694)

[1．背景调查&需求分析 1](#_Toc138923695)

[1.1 项目开发背景 1](#_Toc138923696)

[1.1.1 需求分析 1](#_Toc138923697)

[1.1.2可行性分析 1](#_Toc138923698)

[1.1.3 系统的基础数据流图 2](#_Toc138923699)

[1.1.4 数据字典 2](#_Toc138923700)

[2．数据库系统设计—概念设计 3](#_Toc138923701)

[2.1概念结构设计 3](#_Toc138923702)

[2.1.1实体—联系方法(Entity-Relationship) 3](#_Toc138923703)

[3.数据库系统设计--逻辑结构设计 4](#_Toc138923704)

[3.1关系数据库的基本概念 4](#_Toc138923705)

[3.2 E-R图向关系模型转换的原则 4](#_Toc138923706)

[3.3 关系模式 5](#_Toc138923707)

[数据库关系图 5](#_Toc138923708)

[4.Javaweb技术 6](#_Toc138923709)

[5.JAVA-JDBC相关技术 7](#_Toc138923710)

[5.1Dao层 7](#_Toc138923711)

[6.具体实现 10](#_Toc138923712)

[6.1登录界面 10](#_Toc138923713)

[6.2注册界面 11](#_Toc138923714)

[6.3查询操作界面 11](#_Toc138923715)

[7.收获和体会 12](#_Toc138923716)

# 1．背景调查&需求分析

## 1.1 项目开发背景

21世纪是信息现代化的时代，管理信息系统在各行各业都越来越重要，特别是教育事业，每个学校都需要在期末进行考试成绩的统计分析工作，而这些工作都必须在考试结束后一个星期的时间内完成。大量的成绩数据的统计工作如果只靠人工完成，费时费力，还容易出错。随机计算机技术的快速发展，计算机在日常管理应用中迅速普及，利用计算机进行学生管理势在必行。因此需要开发出一个满足老师进行学生信息的录入、查询、修改等需求的功能完善，安全可靠，迅速便捷的管理系统。以学校管理信息系统建设中的教学管理子系统的开发为背景来开发学生管理系统以减轻学院教学人员的工作量，缩小开支，提高工作效率与准确率。

### 1.1.1 需求分析

该系统有以下需求：

1. 输入学生信息并保存：用户可以输入学生的基本信息，包括学号、姓名、性别、年龄、地址、QQ号和邮箱，并将信息保存到数据库中。

(2) 显示所有学生信息：系统可以将数据库中存储的所有学生信息展示给用户，以便查看和管理。

(3) 查询学生信息：用户可以根据学号或姓名等条件查询学生的详细信息。

(4) 修改学生信息并保存：用户可以修改学生的各项信息，并将修改后的信息保存到数据库中。

(5) 删除学生信息：用户可以根据学号删除指定学生的信息。

### 1.1.2可行性分析

1. 技术可行性

MySQL是一个开源的关系型数据库管理系统，因其性能高效、可靠和易于使用等优势，广泛应用于各种应用系统中，包括学生成绩管理系统。作为技术基础，我们已经具备使用MySQL进行数据库设计和构建的技能，同时有丰富的在线参考资源可供查阅和学习，因此项目在技术层面上是完全可行的。

1. 经济可行性

在经济层面，MySQL和IDEA作为开源产品，无需付出额外的费用即可使用，这大大降低了项目的成本；另外，学生成绩管理系统可以提高办公效率，减少人工输入错误，节约资源，所以从长远来看，这个系统是有经济效益的。

1. 操作可行性

学生成绩管理系统可以大幅度地提高教师录入、查询成绩的效率，同时也方便学生查看与分析自己的考试成绩，这种便利性使得系统具有高度的操作可行性。

### 1.1.3 系统的基础数据流图

通过对系统用户需求的分析，可以得出系统涉及到大量的基础数据。主要包括以下数据实体及数据项：数据流图如下图1-1：

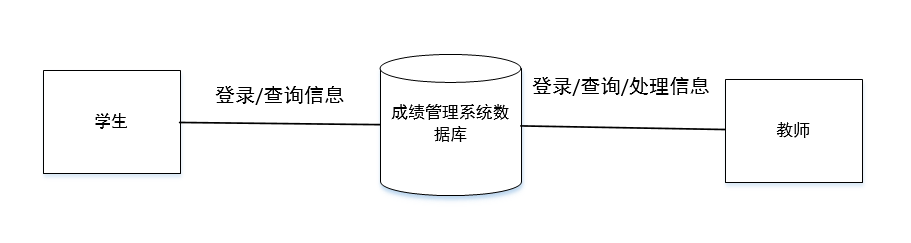


图1-1 学生成绩管理系统数据流图

### 1.1.4 数据字典

①学生student（编号ID、姓名NAME、性别GENDER、年龄AGE、地址ADRESS、QQ、邮箱EMAIL、）

属性名 字段类型 长度 主键或外键 说明

ID INT 11 主键 编号

NAME VARCHAR 20 姓名

GENDER VARCHAR 5 性别

AGE INT 11 年龄

ADRESS VARCHAR 32 地址

QQ VARCHAR 32 QQ

EMAIL VARCHAR 50 邮箱

②用户usr（编号ID、姓名USERNAME、密码PASSWORD）

属性名 字段类型 长度 主键或外键 说明

ID INT 11 主键 编号

USERNAME VARCHAR 50 姓名

PASSWORD VARCHAR 50 密码

# 2．数据库系统设计—概念设计

## 2.1概念结构设计

### 2.1.1实体—联系方法(Entity-Relationship)

确定实体之间的联系

根据实际需求得出，各实体之间的联系如下：

一个用户管理多个学生,一个学生被多个用户管理（学生和用户之间是一对多联系）

**2.1.2 E-R图设计**

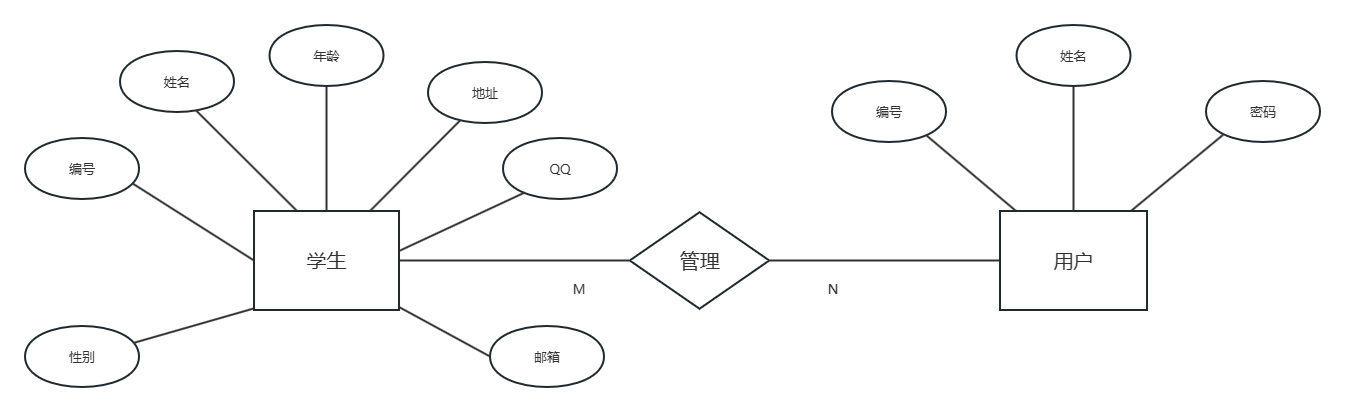


图2-1 数据库E-R图

# 3.数据库系统设计--逻辑结构设计

## 3.1关系数据库的基本概念

由实体间的联系导出关系模型，关系模型在关系数据库中即为一个二维表。

（1）记录：二维表中的一行，表示一个实体。

（2）字段：二维表的列，表示实体的属性、特征。

（3）主键：列或列的组合，能唯一地标识一行。表中只有一个主键，主键不允许NULL或重复。

（4）外键：表中某一字段与另一表中的主键相对应，则该字段称为表的外键。外键表示了两个表之间的联系。主键所在的表称为主表，外键所在的表为从表。

（5）数据完整性：指数据的正确性和可靠性。

（6）实体完整性：保证表中所有行具有唯一性。

（7）域完整性：保证给定字段的数据有效性。

（8）参照完整性：也称引用完整性，确保相关联的表之间数据的一致性。

（9）用户自定义完整性：用户根据实际系统要求，定义的约束条件，涉及到的数据必须满足语义要求。

## 3.2 E-R图向关系模型转换的原则

（1）实体类型的转换

将每个实体类型转换成一个关系模式，实体的属性即为关系的属性，实体标识符即为关系的键。

（2）联系类型的转换

如实体间的联系是1:1，则可以在两个实体类型转换成两个关系模式中的任意一个关系模式的属性中加入另一个关系模式的键和联系类型的属性。

如实体间的联系是1:N，则在N端实体类型转换成的关系模式中加入1端实体类型转换成的关系模式的键和联系类型的属性。

如实体间的联系是M:N，则将联系类型也转换成关系模式，其属性为两端实体类型的键加上联系类型的属性，而键为两端实体键的组合。

## 3.3 关系模式

### 数据库关系图

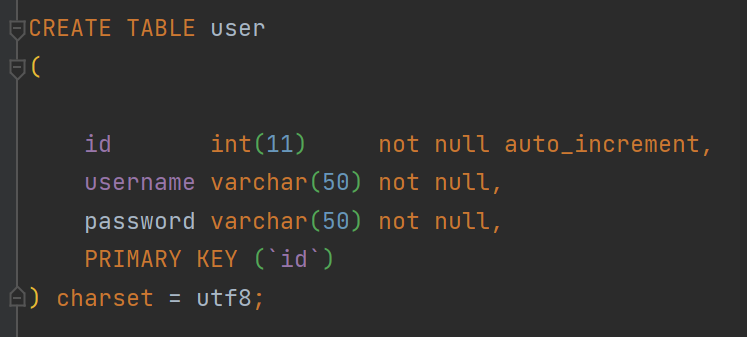
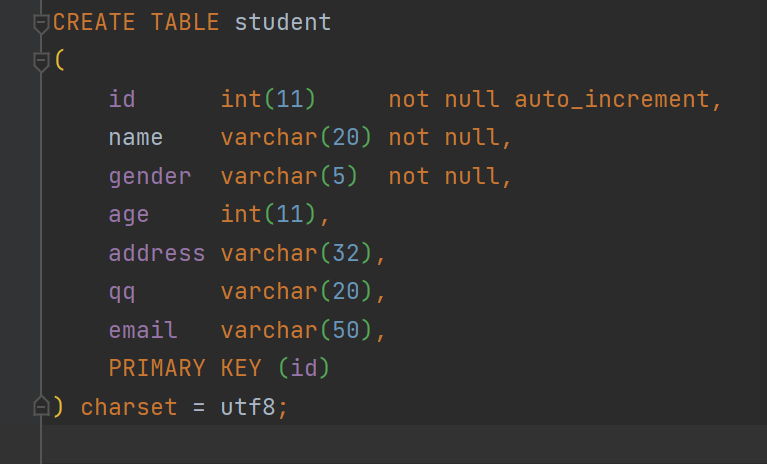


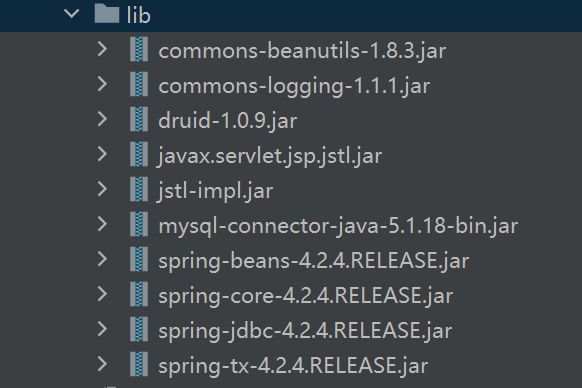
图3-1 数据库关系图

# 4.Javaweb技术

**4.1Maven**

Maven是一个优秀的项目构建工具，用于管理项目的依赖关系、构建过程和项目生命周期。通过简单的配置文件，可以自动下载所需的库文件，并进行项目构建、测试和部署等操作。它提供了一种规范化的项目结构和管理方式，简化了项目的开发和维护过程。

Maven配置文件如下：主要使用的依赖如下所示：



**4.2 JSP（JavaServer Pages）**

JSP是一种动态网页技术，可以将Java代码嵌入到HTML页面中，实现动态内容的生成和展示。它允许开发者在网页中嵌入Java代码，使用JSP标签和脚本语言，与客户端进行交互并生成动态内容。

**4.3 Servlet**

Servlet是用于接收和处理客户端请求的Java类。在使用Javaweb开发中，Servlet充当了Web服务器和Web应用程序之间的桥梁。当用户发起请求时，Servlet会接收请求并进行处理，生成响应结果返回给客户端。它可用于处理各种HTTP请求，并与前端页面进行交互。

**4.4 jQuery**

jQuery是一个优秀的JavaScript库，简化了JavaScript编程，并提供了一套易于使用的API，用于处理HTML文档的遍历、事件操作、动画效果等。它大大简化了前端开发的工作量，使开发者能够更快速地进行页面交互和动态效果的实现。

**4.5 Spring**

Spring是一个轻量级的Java开发框架，用于简化企业级应用程序的开发。它提供了诸如依赖注入、面向切面编程、事务管理等功能，使开发者能够专注于业务逻辑的实现，而无需过多考虑底层技术的细节。Spring框架还提供了Spring MVC用于Web开发和Spring JDBC用于数据库操作。

**4.6 Bean**

Bean指的是Java中的一个普通对象，用于封装数据和业务逻辑。在学生信息管理系统中，以学生信息表和用户表为基础，可以定义对应的Java Bean类，封装相应的属性和方法。

**4.7 Tomcat服务器**

Tomcat是一个流行的开源Java Web服务器，用于运行Java Web应用程序。它能够接收和处理来自客户端的HTTP请求，并将相应结果返回给客户端。在学生信息管理系统中，Tomcat用于部署和运行Javaweb项目。

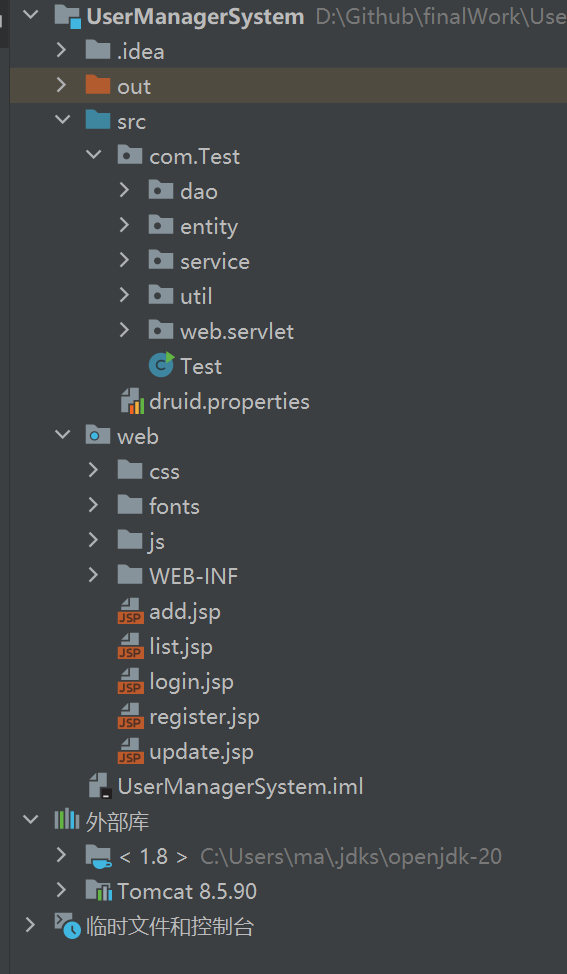
* 1. **MySQL数据库**

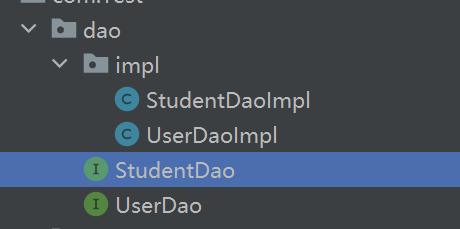
MySQL是一种常用的关系型数据库管理系统，用于存储和管理数据。在学生信息管理系统中，MySQL用于持久化数据，存储学生信息和用户信息。通过使用JDBC技术，与MySQL数据库建立连接并进行查询、插入、更新和删除等数据库操作。

# 5.JAVA-JDBC相关技术

# 5.1Dao层

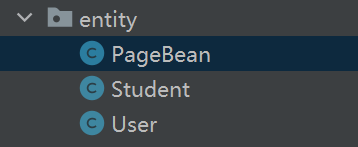
DAO（Data Access Object） 模型就是写一个类，把访问数据库的代码封装起来，DAO在数据库与业务逻辑（Service）之间。

Dao是数据访问层，Dao的作用是封装对数据库的访问：增删改查，不涉及业务逻辑，只是达到按某个条件获得指定数据的要求。项目整体架构。



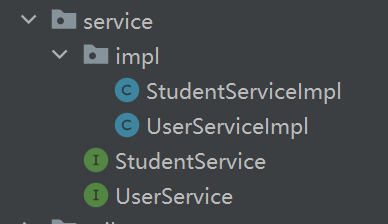
**5.2 Entity层**

Entity层是实体层，用于封装与数据库表对应的实体对象。在学生信息管理系统中，可以使用Entity类来表示学生信息和用户信息。Entity类包含了与表中字段对应的属性以及对应的getter和setter方法。通过Entity类，可以方便地进行数据的封装和操作，提高代码的可读性和可维护性。



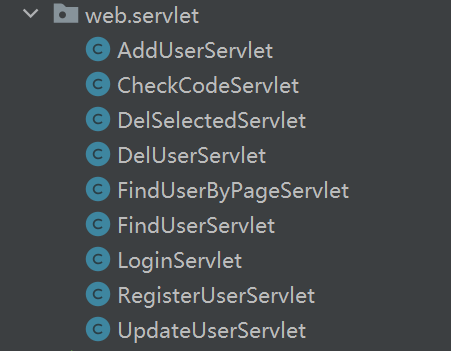
**5.3 Service层**

Service层是业务逻辑层，负责处理来自Servlet层的请求，并协调不同的Dao层方法来完成具体的业务操作。在学生信息管理系统中，Service层可以包含各种操作学生信息和用户信息的方法，如添加学生信息、查询学生信息、修改学生信息和删除学生信息等。Service层还可以处理业务逻辑的验证和转换，确保数据的完整性和一致性。



**5.4 Servlet层**

Servlet层是用来接收和处理客户端请求的Java类。在学生信息管理系统中，Servlet层负责接收来自前端页面的请求，并将请求参数传递给Service层进行处理。Servlet层还负责将Service层返回的结果进行处理，并生成响应结果返回给前端页面。Servlet层与前端页面进行交互，处理各种请求，包括展示学生信息、添加学生信息、修改学生信息和删除学生信息等。



**5.5 Utils层**

Utils层是工具类层，用于封装一些通用的功能和方法，提供给其他层使用。在学生信息管理系统中，Utils层可以包含一些与数据库连接、日期处理、密码加密等相关的功能。例如，可以创建一个数据库连接的工具类，封装了获取数据库连接、关闭连接等操作；还可以创建一个日期处理的工具类，封装了日期格式化和解析的方法。Utils层的目的是提供一些通用的功能，提高代码的重用性和可维护性。 通过以上的代码实现分层，可以使学生信息管理系统的代码结构清晰、功能模块化，并提高代码的可读性、可维护性和可扩展性。每个层都有特定的职责和功能，并通过定义好的接口和关联关系，实现了各个层之间的解耦和协同工作。这样的架构设计有助于团队合作、代码重用和系统的可持续发展。



这其中为了防止出现异常，在创建过程中使用了数据库的事务控制技术，实现了事物的begin（），commit（），rollback（）等操作。确保事务在运行过程中的稳定性

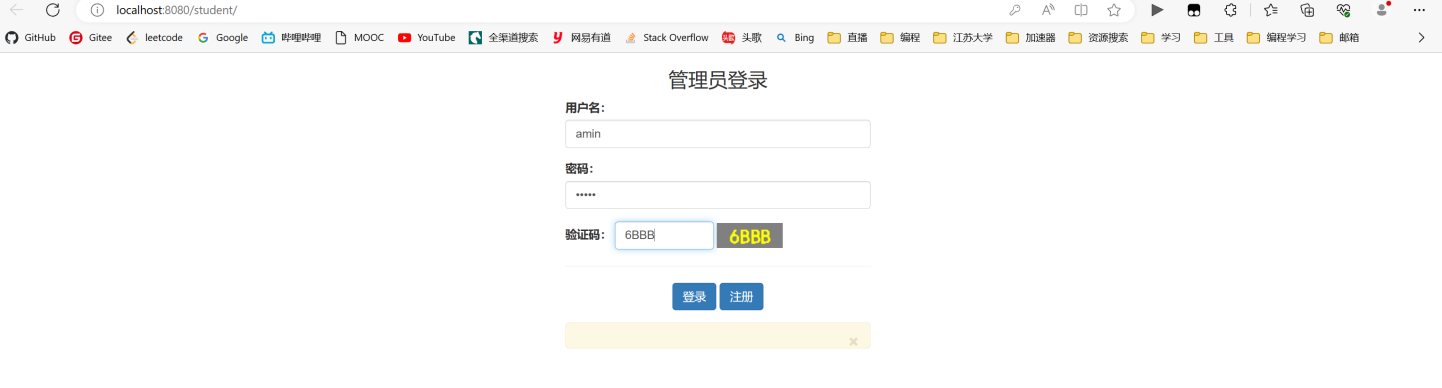
# 6.具体实现

## 6.1登录界面

登录界面如图所示

在管理员登陆处输入用户名和密码

在这下面有个验证码接收，输入验证码后在验证过后才可以登录



## 6.2注册界面



在这个界面中输入用户名和密码提交后即可完成注册

## 6.3查询操作界面



在登录成功后会出现如图所示列表可以通过条件来进行查询

如图便是对于籍贯来自北京的学生进行查询的结果



除此右上方还有添加学生界面



在这之中输入个人信息即可完成学生的添加。

# 7.收获和体会

通过学生管理系统项目设计，我们三人加深了对数据库设计的理解，对于设计一个数据库项目的、需求、设计流程有了一定的了解。纸上得来终觉浅，这次的课程设计我们三人把书上和试卷上的知识点在此次大作业中应用。

在本次课程设计中，我们获得了很多宝贵的经验和体会： 1. 数据库设计能力：通过设计学生信息管理系统的数据库，我们学会了如何根据需求分析和实体关系设计关系模式，并将其转化为具体的数据库表结构。我们了解了数据库设计的规范化过程和原则，以及如何优化表结构，提高数据库的性能和可维护性。 2. 技术综合应用能力：在实现学生信息管理系统的过程中，我们综合运用了多种技术和工具，如Javaweb、JSP、Servlet、Maven、MySQL等。通过实践，我们深入理解了各种技术的作用和优势，学会了如何根据实际业务需求选择合适的技术并进行整合。 3. 分层架构设计能力：为了使系统具有良好的结构和可维护性，我们将代码进行了分层设计，包括Dao层、Entity层、Service层、Servlet层和Utils层。通过这种分层架构，我们更好地组织和管理了代码，实现了代码的模块化和解耦，提高了代码的可读性和可维护性。 4. 前后端协作能力：本次课程设计要求实现一个具有前后端交互功能的学生信息管理系统。在这个过程中，我们与前端开发人员密切合作，进行接口的定义和数据交互的实现。这锻炼了我们的沟通和协作能力，提高了团队合作的效率和效果。 5. 问题解决能力：在开发过程中，我们遇到了一些技术难题和bug，但通过查阅文档、网络搜索和与同学讨论，我们成功地解决了这些问题。这培养了我们独立解决问题和持续学习的能力，提高了我们在开发过程中快速定位和解决问题的能力。

总的来说，本次课程设计让我们从理论到实践地学习了数据库设计和开发技术，并获得了宝贵的经验和体会。通过这个项目，我们不仅提高了自己的技术能力，还锻炼了团队合作、问题解决和持续学习的能力，为将来的工作和学习打下了良好的基础。项目最终开发出来的程序实现的功能比较简单，但也基本达到了项目设计的要求，可以看出我们也是是花了不少心思的。最重要的是，这次数据库项目的设计，为我们以后程序开发的道路打下了扎实的基础。