《数据库原理与应用实验》

大作业设计文档

———学生成绩管理系统

**一、系统概述**

**1. 系统背景与目标**

随着教育信息化的发展，学生成绩管理已成为学校教务管理工作的重要组成部分。本系统旨在设计一个基于C#窗体应用程序的学生成绩管理系统，以提高学校对学生成绩管理的效率和质量。系统将分为学生管理模块和成绩管理模块两大功能模块，满足学生、教师和管理员的不同需求。

**2. 系统功能模块**

**学生管理模块**

（1）学生用户创建与基本信息修改；

（2）学生选课功能，包括预选、正选、退选等；

**成绩管理模块**

（1）教师对学生成绩进行录入的功能，包括平时成绩的录入、末考成绩的录入、综合成绩的自动计算（平时成绩占30%，末考成绩占70%）；

（2）管理员对成绩进行变更的功能，需要记录更改成绩的学生学号、课程号、变动原因、变动时间、学院教学管理办公室审批意见、学院院长审批意见、学校教务处审批意见。在界面上显示学生历史成绩变化趋势和排名。同时提供权限控制，防止用户越权修改成绩信息。

**3. 系统优点**

(1)界面友好：使用C#窗体应用程序开发，界面简洁明了，易于使用；

(2)数据安全：采用加密存储和传输技术，确保数据安全；

(3)性能优化：通过优化数据库操作和代码逻辑，提高系统的运行效率。

**4. 系统实现技术**

本系统采用C#作为开发语言，使用WinForms窗体应用程序框架进行开发。同时，采用数据库技术SQL Server实现数据的存储和交互。系统支持批量操作和数据导入导出，以提高管理效率。

**5. 系统应用场景**

本系统适用于各类学校，特别是需要进行学生成绩管理的学校。系统可以帮助学校更好地管理学生选课、成绩录入和变更等事务，提高教务管理效率和质量。同时，该系统也可用于其他需要对学生成绩进行管理的场合。

**二、需求分析**

**1.用户功能资料收集**

学生成绩管理系统由4个部分组成:学生、教师、管理员用户的创建及信息修改;学生选课功能;教师成绩的录入;管理员对成绩的修改。

本系统主要流程为:学生、教师、管理员首先注册对应账号输入相应信息;管理员输入本部门课程信息,学生根据本部门的课程进行选课,管理员根据学生的选课信息和教师信息排课形成授课表,教师根据授课表进行授课并给出课程成绩;管理员后期可进行成绩修改操作。

**2.需求分析**

**系统输入与输出信息**

(1)系统的输入信息

学生基本信息：学号、姓名、性别、出生日期、专业代号。

教师基本信息：教师号、姓名、性别、出生日期、专业代号。

成绩信息：学生选修课程的成绩(包括平时成绩和末考成绩)、更改前后成绩及原因和学院意见等。

专业信息：专业代号和专业名称。

课程信息：课程号、课程名称、课程性质。

(2)系统的输出信息

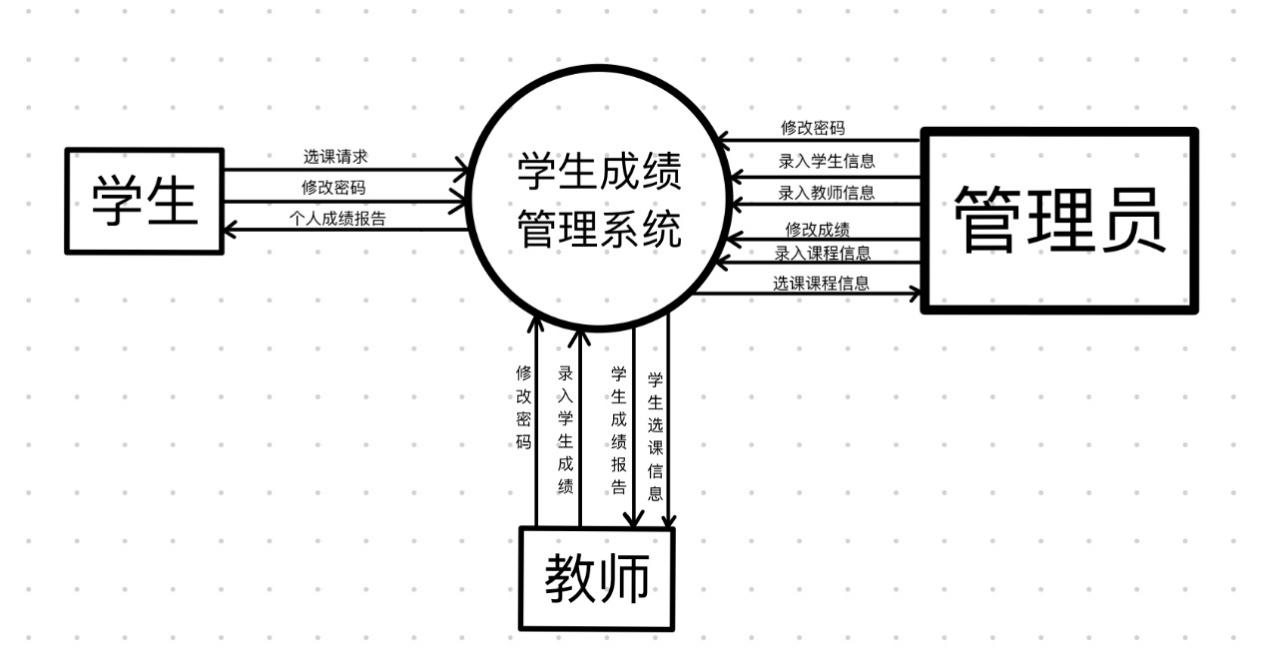
学生基本信息：学号、姓名、性别、出生日期、专业代号。

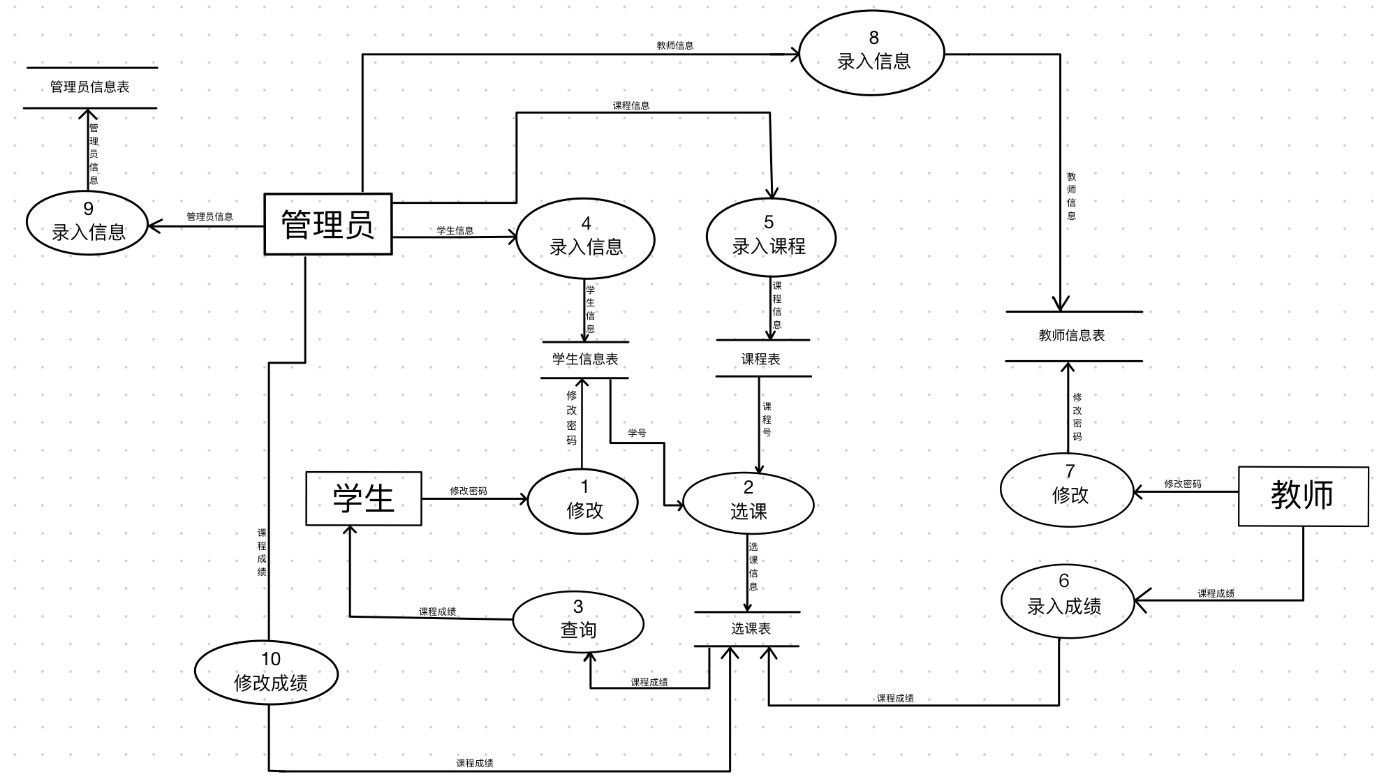
教师基本信息：教师号、姓名、性别、出生日期、专业代号。

成绩信息：学号、课程号、成绩、修改原因、修改时间、学院意见。

授课信息：课程号、教师号

**数据流图与数据字典**

数据流图:



数据字典:

**a.学生信息表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 默认值 | 是否允许为空 | 描述 |
| 学号 | char | 10 | - | 否 | 主键 |
| 学生姓名 | nvarchar | 50 | - | 否 |  |
| 密码 | nvarchar | 50 | - | 否 |  |
| 性别 | bit | - | 1 | 否 | 1为男 |
| 出生日期 | date | - | - | 否 |  |
| 专业代号 | char | 3 | - | 否 |  |

**b.教师信息表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 默认值 | 是否允许为空 | 描述 |
| 教师号 | char | 10 | - | 否 | 主键 |
| 教师姓名 | nvarchar | 50 | - | 否 |  |
| 密码 | nvarchar | 50 | - | 否 |  |
| 性别 | bit | - | 1 | 否 | 1为男 |
| 出生日期 | date | - | - | 否 |  |
| 专业代号 | char | 3 | - | 否 |  |

**c.管理员信息表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 默认值 | 是否允许为空 | 描述 |
| 管理员号 | char | 10 | - | 否 | 主键 |
| 管理员  姓名 | nvarchar | 50 | - | 否 |  |
| 密码 | nvarchar | 50 | - | 否 |  |
| 性别 | bit | - | 1 | 否 | 1为男 |
| 出生日期 | date | - | - | 否 |  |

**d.课程表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 默认值 | 是否允许为空 | 描述 |
| 课程号 | char | 10 | - | 否 | 主键 |
| 课程名称 | nvarchar | 100 | - | 否 |  |
| 教师姓名 | nvarchar | 100 | - | 否 |  |

**e.选课表**

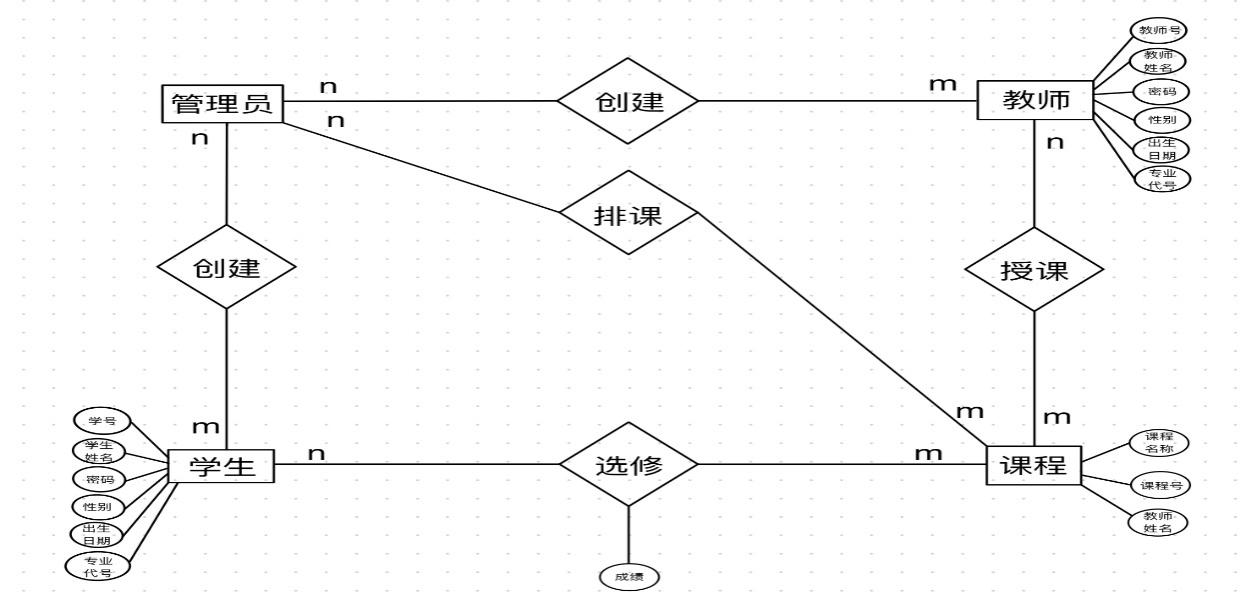
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 默认值 | 是否允许为空 | 描述 |
| 学号 | char | 10 | - | 否 | 主键 |
| 学生姓名 | nvarchar | 100 | - | 否 |  |
| 课程号 | char | 10 | - | 否 | 主键 |
| 课程名称 | nvarchar | 100 | - | 否 |  |
| 教师姓名 | nvarchar | 100 | - | 否 |  |
| 平时成绩 | Int | - | - | 是 |  |
| 末考成绩 | Int | - | - | 是 |  |
| 综合成绩 | Int | - | - | 是 |  |
| 是否修改成绩 | nchar | 10 | - | 是 | 有修改填是 |
| 意见 | nchar | 10 | - | 是 |  |
| 原平时成绩 | Int | - | - | 是 |  |
| 原末考成绩 | Int | - | - | 是 |  |
| 修改时间 | date | - | - | 是 |  |

**f.录入成绩**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 默认值 | 是否允许为空 | 描述 |
| 录入成绩开关 | Int | - | - | 否 | 1为教师可以录入成绩,反之为不可录入 |

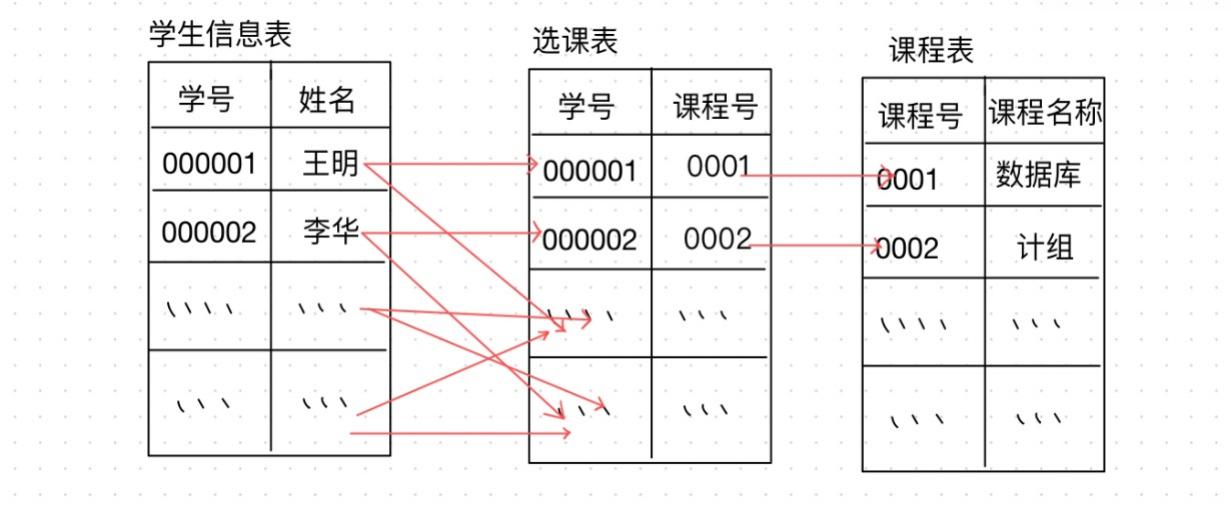
**E-R图与关系表**

选择这种实体与实体间的关系设计方案构建ER图，是因为它清晰直观地揭示了各个表之间的内在联系，便于理解和维护数据之间的逻辑关系，从而更有效地支持数据管理和应用需求。

E-R图

关系表

此图为学生信息表、选课表与课程表三表之间的关系表,相同地,教师信息表和管理员信息表也可与之产生联系.



**关系模式和BCNF**

关系模式

学生(学号,学生姓名,密码,性别,出生日期,专业代号)

教师(教师号,教师姓名,密码,性别,出生日期,专业代号)

管理员(管理员号,管理员姓名,密码,性别,出生日期)

选课(学号,学生姓名,课程号,课程名称,教师姓名,平时成绩,末考成绩,综合成绩)

**学生表**

学号是主键，所有非主属性都完全依赖于学号，且学号是唯一的决定因素。

满足BCNF。

**教师表**

教师号是主键，所有非主属性都完全依赖于教师号，且教师号是唯一的决定因素。

满足BCNF。

**管理员表**

管理员号是主键，所有非主属性都完全依赖于管理员号，且管理员号是唯一的决定因素。

满足BCNF。

**选课表**

(学号, 课程号)是主键，所有非主属性都完全依赖于(学号, 课程号)。

但是，存在函数依赖：学号 → 学生姓名，课程号 → 课程名称，教师姓名 → 教师号。

这些函数依赖的决定因素不是候选键，因此选课表不满足BCNF。

为了满足BCNF，选课表需要分解为以下几个表：

**学生选课表**（学号, 课程号, 平时成绩, 末考成绩, 综合成绩）

**课程表**（课程号, 课程名称）

**教师表**（教师号, 教师姓名）

这样，每个表的主键都是唯一的决定因素，满足BCNF。

**三、功能实现**

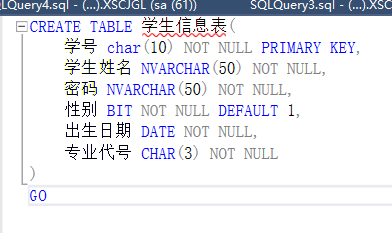
**1.建立数据库**

建立了名为**XSCJGL**的数据库。

**2.建立表格**

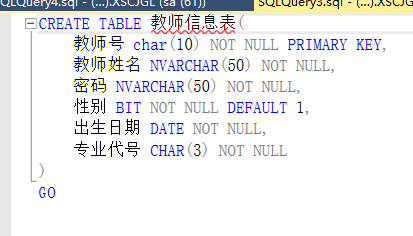
**a.学生信息表**

存储学生信息。



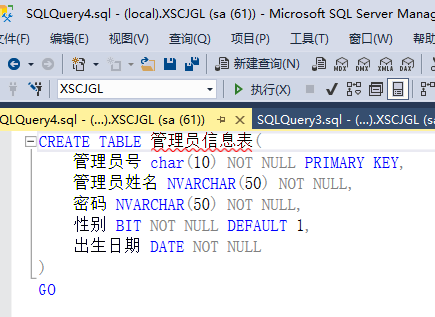
**b.教师信息表**

存储教师信息。

****

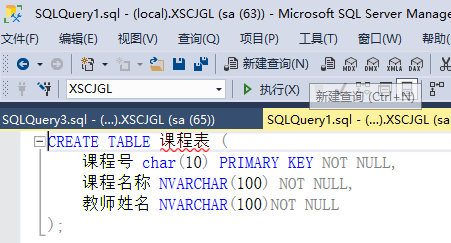
**c.管理员信息表**

存储管理员信息。

****

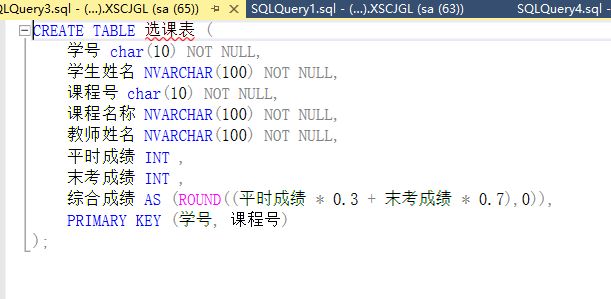
**d.课程表**

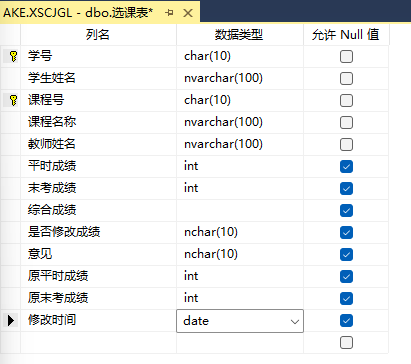
用于管理员向其中录入课程信息。

****

**e.选课表**

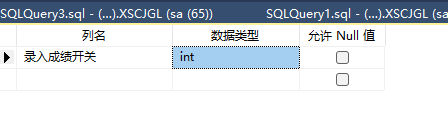
用于记录学生选课信息及老师录入成绩。(之后补了修改成绩的一些必要的列)

****

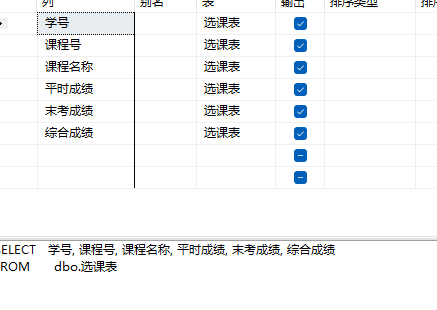
****

**f.录入成绩**

用于管理员管理教师是否能录入成绩到选课表中。

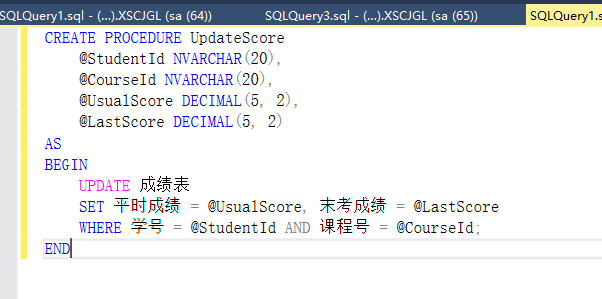
****

**3.建立视图**

使用学生信息表和选课表****,建立成绩表视图，包含以下信息，用于学生查看成绩及教师录入成绩。

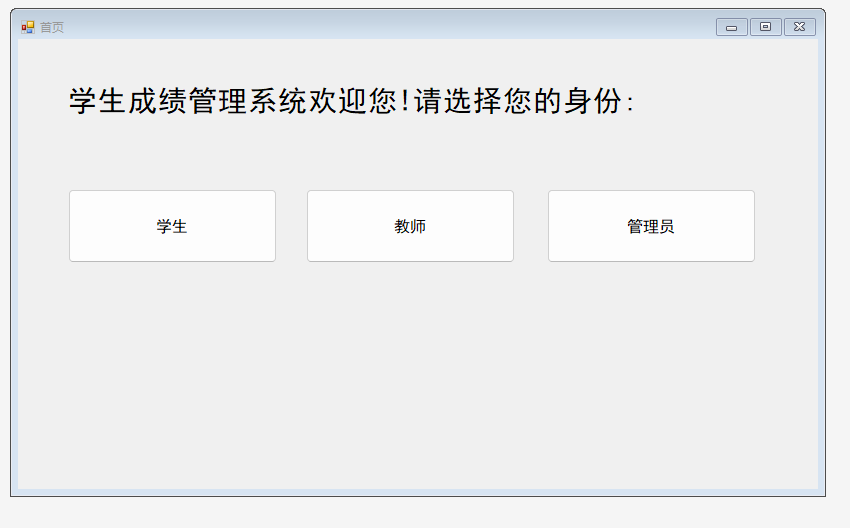
**4.建立存储过程**

建立存储过程,教师可调用此存储过程向成绩表录入学生成绩。

****

**5. C#与SQL server结合建立此学生管理系统**

**主页:**

****点击不同按钮进入不同登录页面,与相应信息表产生联系,检索账号与密码是否匹配,是否登录成功。

**登录页面:**

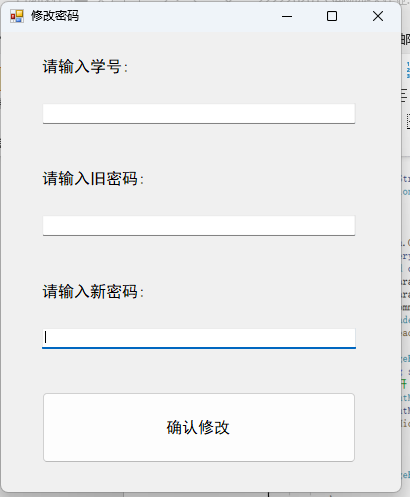
以学生登录页面为示例,其余与其基本相同。使用select语句查询学生姓名传值到下一页面显示登录学生姓名。

****

****

**修改密码页面:**

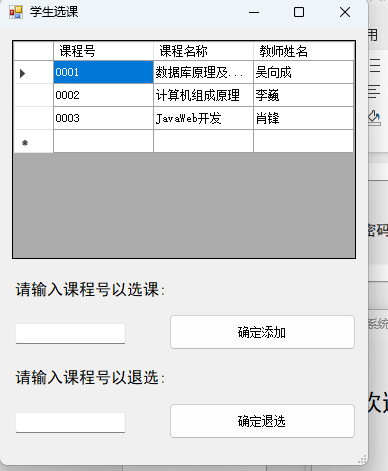
以学生修改页面为示例。输入学号，在数据库中使用select语句查询其密码是否为旧密码，若相同则使用update语句更改其密码

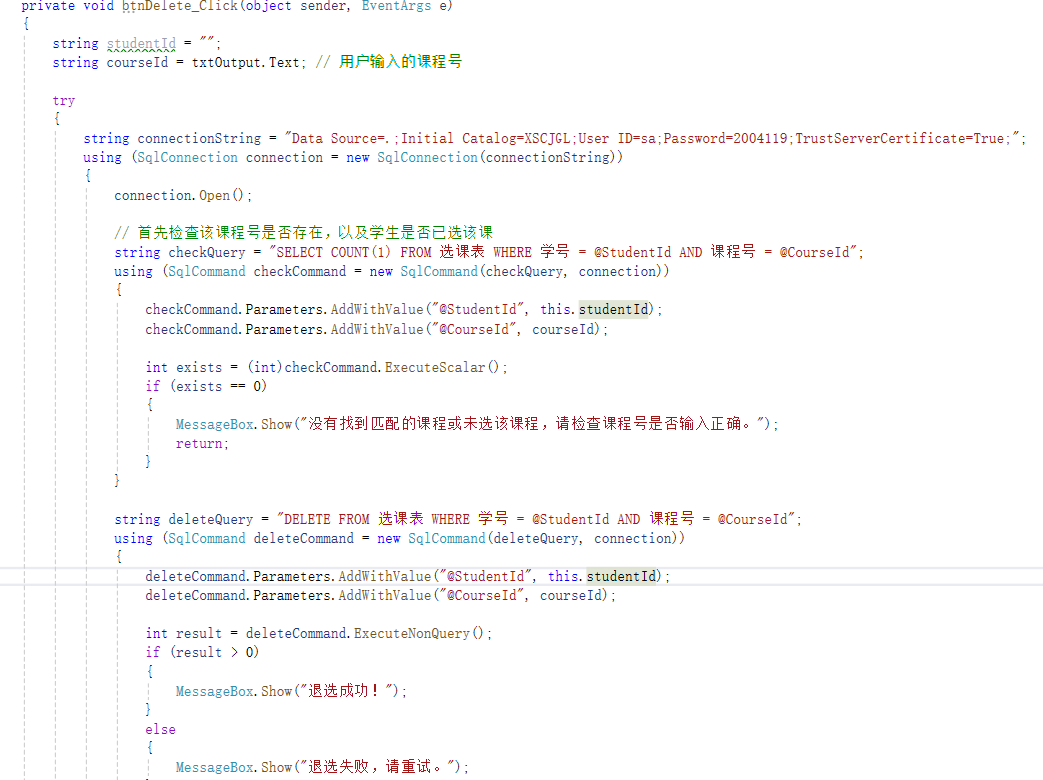




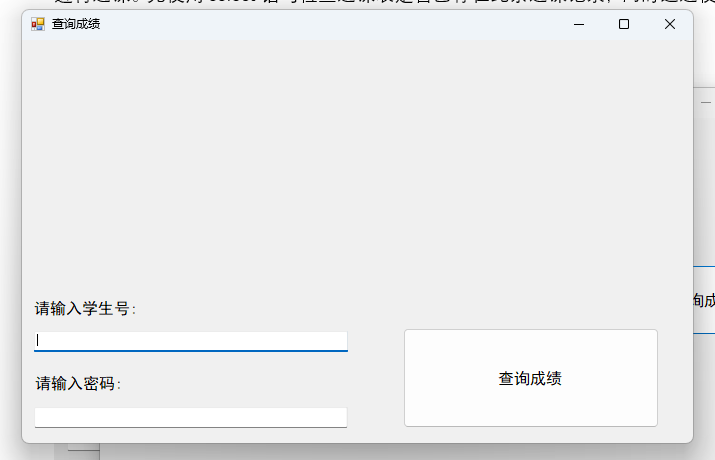
**选课页面：**

学生进行选课，利用C#中datagridview控件显示课程表，使学生能根据课程表进行选课。先使用select语句检查选课表是否已存在此条选课记录，同时退选使用delete语句对记录进行删除。

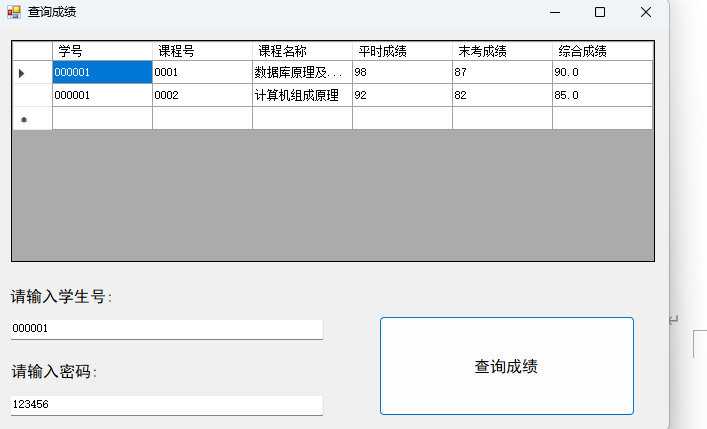




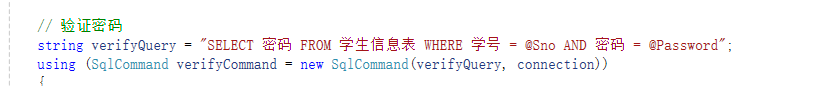
**查询成绩页面：**

****

验证是否登陆成功，登陆成功则展示成绩（防止查看到其他同学成绩）。

****

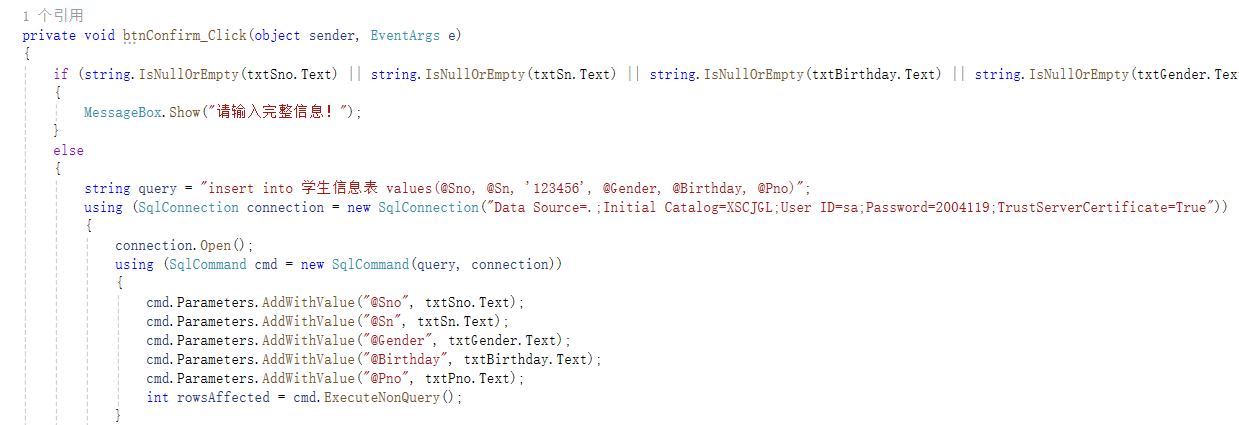
使用select语句进行验证。

****

**管理员创建学生信息页面：**

使用insert语句向学生信息表录入数据(密码默认为123456)，同理，向教师信息表，课程表录入数据，即创建用户与开设课程步骤;包含使用update语句重置密码。







**录入成绩与修改成绩页面：**

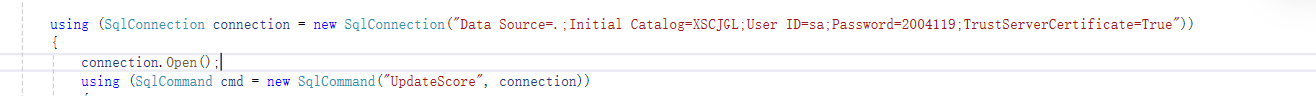
同理，使用相关控件显示视图成绩表。输入学号和课程号查询对应老师和学生的课程，并调用存储过程UpdateScore更新平时成绩和末考成绩。

管理员修改成绩需要向数据库输入更多信息，比如是否修改，学院意见，旧成绩，修改时间等。









**限制教师是否可以录入成绩功能：**

通过使用update语句更改表录入成绩中录入成绩开关是0（关闭录入）是1（开启录入）来对教师进行限制。

****

****

**四、总结**

经过本次大作业报告的撰写以及功能的实现，更加了解了SQL server的使用。在此过程中加强了自己撰写代码的能力及遇到困难找资料解决的能力。解决了绝大部分问题，却依然存在一定问题例如限制教师不可更改成绩的功能，本应在数据库中使用create login语句创建用户，通过grant语句赋予权限，使教师不具备update功能向成绩表中录入数据，但由于C#学得比较浅，不知道怎么使用该语句，但使用其他方法：建立了一个录入成绩开关的表，表中数据为1则可录入成绩反之则不可录入，管理员可通过修改这个开关的值进行限制。相信在更深的研究之后能找到解决方法。