



南京邮电大学通达学院

Tongda College Of Nanjing University Of Posts & Telecommunications

# 数据库应用 大作业

(2021/2022 学年 第 二 学期)

题 目：高中学生信息管理系统(管理员)  
的设计与实现

专 业 数字媒体技术

学 生 姓 名 \_\_\_\_\_

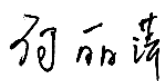
班 级 学 号 \_\_\_\_\_

指 导 教 师 何丽萍

指 导 单 位 教育科学与技术学院

日 期 2022 年 5 月 31 日-6 月 15 日

## 《数据库应用大作业》评分表

<b>题目</b>	高中学生信息管理系统（管理员）					
<b>学生姓名</b>		<b>班级学号</b>		<b>专业</b>	数字媒体技术	
<b>任课教师</b>	何丽萍		<b>指导单位</b>	教育科学与技术学院		
<b>评分内容</b>	具 体 要 求				<b>总分</b>	<b>评分</b>
需求分析	能针对某个数据库应用系统，给出系统数据与处理的用户需求。				10	
概念结构设计质量	数据库概念结构设计合理，应用系统的概念模式（E_R图）完整。				20	
逻辑结构设计质量	数据库逻辑结构设计合理，能将E_R图转换为有效的关系模式。				20	
软件编码	程序运行结果合理，能针对不同角色用户，执行对应的操作。				40	
设计报告	报告书撰写规范。				10	
<b>总评分</b>					100	
评  语	<p>该同学的数据库应用大作业，能针对某个数据库应用系统，给出系统数据与处理的用户需求。</p> <p>数据库概念结构设计合理，应用系统的概念模式（E_R图）完整。</p> <p>数据库逻辑结构设计合理，能将E_R图转换为有效的关系模式。</p> <p>程序运行结果合理，能针对不同角色用户，执行对应的操作。</p> <p>报告书撰写比较规范。</p> <p style="text-align: right;">2022 年 6 月 22 日</p>					
评分教师	姓名	职称	工作单位		电子签名	
	何丽萍	讲师	教育科学与技术学院			

## 目录

### 1.需求分析

- (1) 登陆时，需认证密码
- (2) 管理员模块可以查看学生、老师的信息。
- (3) 管理员对学生、老师的信息进行增删改。
- (4) 管理员模块可以查看学生的成绩。
- (5) 管理员对学生的成绩进行增删改。

### 2.概念结构设计（E-R 图）

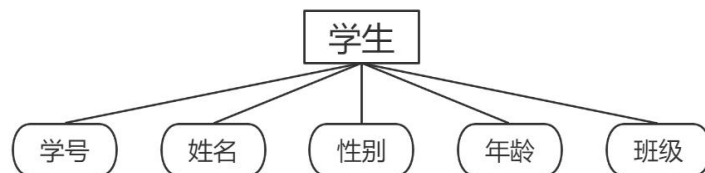


图 1 学生实体属性图



图 2 管理员实体属性图

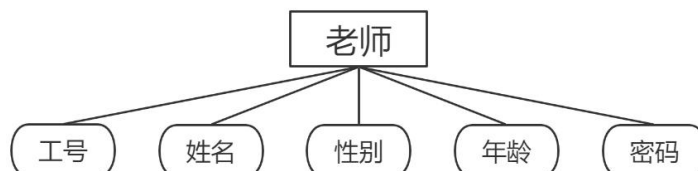


图 3 老师实体属性图

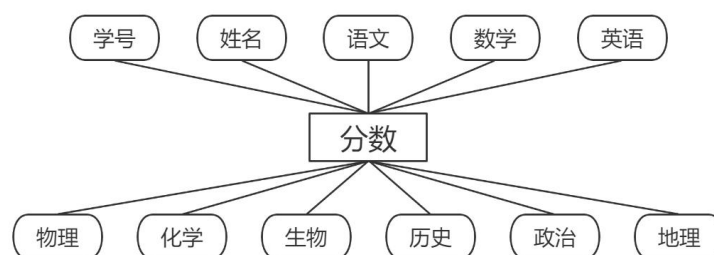


图 4 分数实体属性图

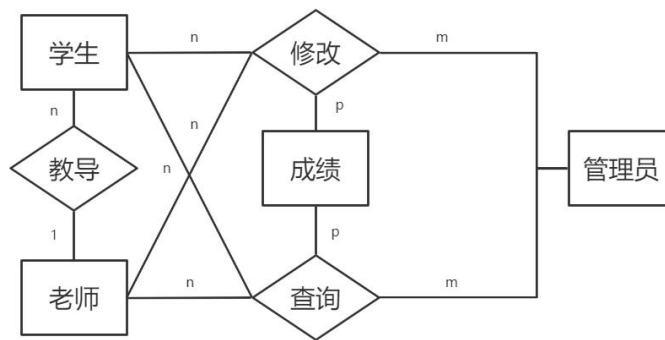


图 5 实体联系图

### 3.逻辑结构设计（数据库表结构）

#### 1) Administrator 表结构如下：

表 1 Administrator 表（管理员信息表）

字段名称	类 型	宽 度	允许空值	主 键	说 明
ID	varchar	10	NULL	是	管理员工号
Name	varchar	10	NULL		管理员姓名
Sex	varchar	2	NULL		管理员性别
Age	varchar	5	NULL		管理员年龄
Password	varchar	20	NULL		管理员密码

#### 2) Student 表结构如下：

表 2 Student 表（学生信息表）

字段名称	类 型	宽 度	允许空值	主 键	说 明
ID	varchar	50	NULL	是	学生学号
Name	varchar	50	NULL		学生姓名
Sex	varchar	50	NULL		学生性别
Age	varchar	50	NULL		学生年龄
Class	varchar	50	NULL		学生所在班级
Password	varchar	50	NULL		学生密码

#### 3) Teacher 表结构如下：

表 3 Teacher 表（老师信息表）

字段名称	类 型	宽 度	允许空值	主 键	说 明
ID	varchar	50	NULL	是	老师工号
Name	varchar	50	NULL		老师姓名
Sex	varchar	50	NULL		老师性别
Age	varchar	50	NULL		老师年龄

Password	varchar	50	NULL		老师密码
----------	---------	----	------	--	------

4) Score 表结构如下:

表 4 Score 表 (分数信息表)

字段名称	类 型	宽 度	允许空值	主 键	说 明
ID	varchar	50	NULL	是	学生学号
Name	varchar	50	NULL		学生姓名
Chinese	varchar	50	NULL		语文成绩
Maths	varchar	50	NULL		数学成绩
English	varchar	50	NULL		英语成绩
Physics	varchar	50	NULL		物理成绩
Chemistry	varchar	50	NULL		化学成绩
Biology	varchar	50	NULL		生物成绩
History	varchar	50	NULL		历史成绩
Politics	varchar	50	NULL		政治成绩
Geography	varchar	50	NULL		地理成绩

#### 4.应用系统的实现 (代码对应的流程图和运行效果)

##### (1)密码验证

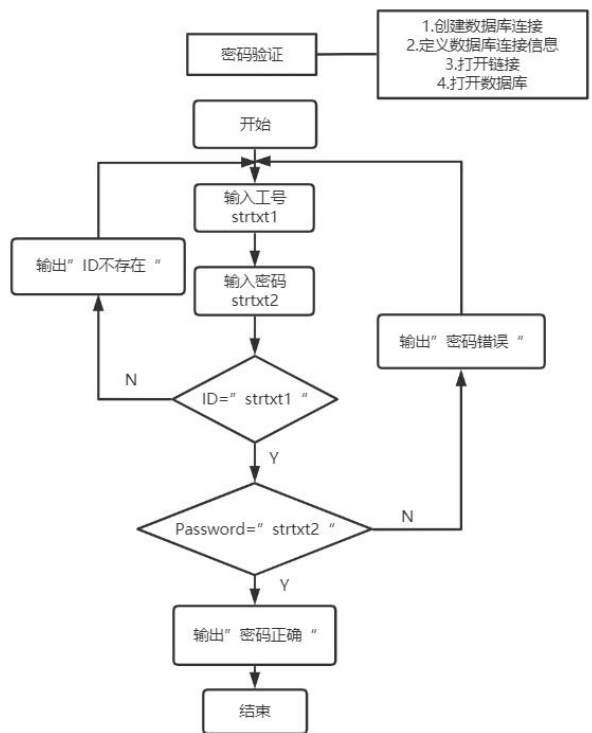
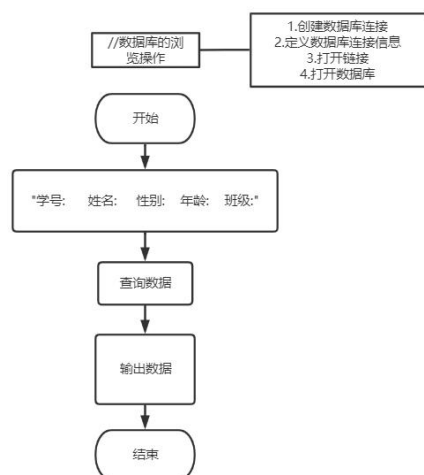


图 6.1 密码验证流程图

## (2) 数据库的浏览



```

1
已连接到学生成绩查询管理系统数据库
学号:      姓名:      性别:      年龄:      班级:
0101      章  阡  谭      男      19      一班
0205      吴  越  仁      男  女      20      二班
0544      秦  宇  莉      女      21      五班
1518      黄  糖  玉      女      19      十五班
0503      王  雪  染      女      21      五班
0506      王  雪  染      女      19      五班
    
```

图 7.2 运行效果

图 7.1 数据库浏览流程图

## (3) 增加学生信息

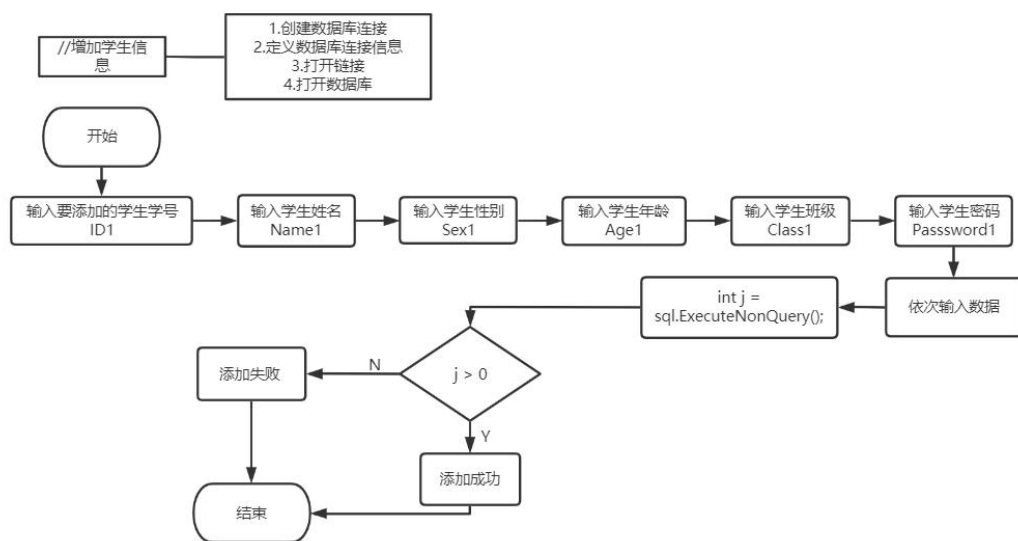


图 8.1 学生信息增加流程图

```

*****
*****      请输入要添加的学生学号      *****
0541
*****      请输入学生姓名      *****
徐珂清
*****      请输入学生性别      *****
男
*****      请输入学生年龄      *****
20
*****      请输入学生班级      *****
五班
*****      请输入学生密码      *****
111111
*****      添加成功      *****
    
```

图 8.2 运行效果

#### (4) 修改学生信息

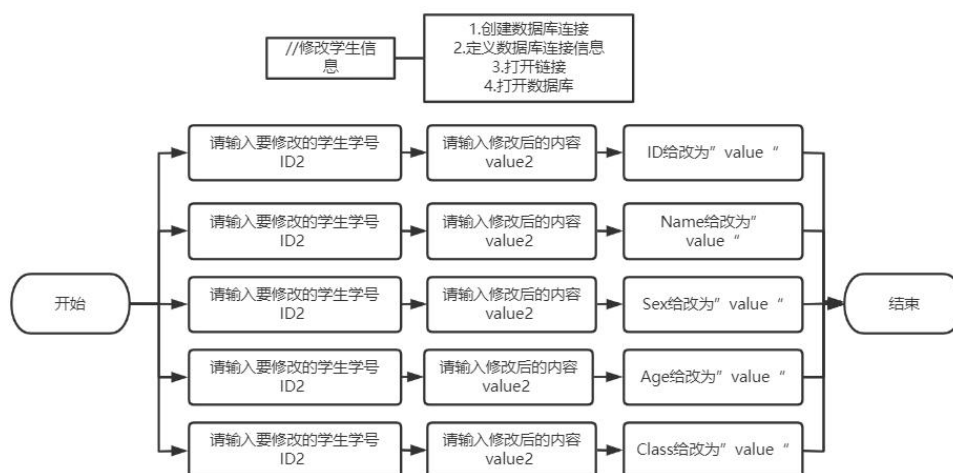


图 9.1 学生信息修改流程图

```

3
***** 1--修改学生学号 *****
***** 2--修改学生姓名 *****
***** 3--修改学生性别 *****
***** 4--修改学生年龄 *****
***** 5--修改学生班级 *****
***** 0--退出系统 *****
*****
2
***** 请输入要修改的学生姓名 *****
徐珂洁
***** 请输入修改后的内容的 *****
徐珂
修改生
  
```

图 9.2 运行效果

#### (5) 删除学生信息

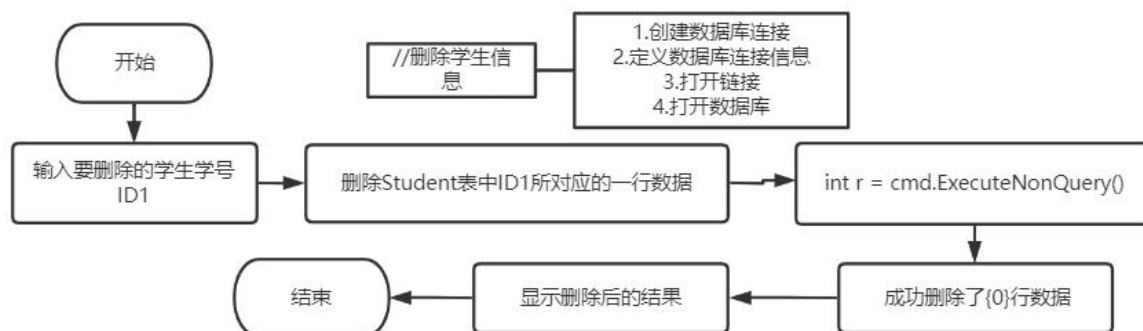


图 10.1 学生信息删除流程图

```

1
***** 请输入要删除的学生学号 *****
0541
***** 成功删除了1行数据 *****
***** 已连接到课程管理数据库 *****
***** 删除后的学生新数据为 *****
学号: 0101
学号: 0205
学号: 0544
学号: 1518
学号: 0503
学号: 0506
  
```

图 10.2 运行效果

## 5. 软件使用说明书

- (01) 打开软件，选择身份（以管理员为例）
- (02) 输入工号，密码。
- (03) 选择所要进行的项目
- (04) 项目 1：查看所有学生的基本信息。
- (05) 项目 2：依次输入学号，姓名，性别，年龄，班级，增加学生信息。
- (06) 项目 3：输入学号，输入修改列，输入修改值，修改学生信息。
- (07) 项目 4：输入学号，删除 Student 表中该学生的数据。
- (08) 项目 5：查看所有学生的成绩。
- (09) 项目 7：依次输入学号，姓名，语文成绩，数学成绩，英语成绩，物理成绩，化学成绩，生物成绩，历史成绩，政治成绩，地理成绩，增加学生成绩信息。
- (10) 项目 8：输入学号，删除 Score 表中该学生的数据。

## 6. 总结

- (1) 了解并基本掌握了数据库连接和使用的方法
- (2) 学习并运用了 SQL 语句。
- (3) 部分内容没有掌握。C#语言基础差，许多东西无法解决。
- (4) 由于版本过低，数据库链接不上。在寻求解决方案中，逐一去尝试。最后发现正剩下版本这一问题后，重下载 2019 版的。这件事，提高了我面对困难时的处理能力



## 附录一源代码

```
//密码验证
//管理员
case 1:
    {
        // authentication();扩展以后,可以对此函数对每个角色进行权限检查
        Administrator(); //创建数据库连接
        String connn = "Data Source=(LocalDB)\\MSS QLLocalDB;AttachDbFilena me=D:\\xscjgl\\
            \\wcxsgl\\wcxsgl.mdf;Integrated Security=True;Connect Timeout=30"; //定义数据库连接信息
        SqlConnection conl4 = new SqlConnection(connn);
        conl4.Open();
        Console.WriteLine("*****");
        Console.WriteLine("*****      请输入工号      *****");
        string strt1 = Console.ReadLine();
        Console.WriteLine("*****      请输入密码      *****");
        string strt2 = Console.ReadLine();
        Console.WriteLine("*****");
        string command5 = "select count(ID) from [Administrator] where ID=" + strt1;
        SqlCommand mySqlCommand = new SqlCommand(command5, conl4);
        int m = Convert.ToInt32(mySqlCommand.ExecuteScalar());
        if (m == 0)
            Console.WriteLine("用户不存在");
        else
        {
            command5 = "select count(ID) from [Administrator] where ID=" + strt1 + "and Password="
+ strt2;

            SqlCommand sql = new SqlCommand(command5, conl4);
            int f = Convert.ToInt32(sql.ExecuteScalar());
            if (f != 0)
            {
                Console.WriteLine("成功");
            }
            else
            {
                Console.WriteLine("密码错误");
            }
        }
        break;
    }

//数据库的浏览操作
case 1:
    1. 创建数据库连接 2. /定义数据库连接信息 3. 打开链接 4. 打开数据库
    string sqlStr = "select * from dbo.Student";
    SqlCommand comm = new SqlCommand(sqlStr, conl);
    SqlDataReader sdr = null;
```

```

try
{
    Console.WriteLine("学号:      姓名:      性别:      年龄:      班级:");
    sdr = comm.ExecuteReader();
    while (sdr.Read())
    {
        Console.WriteLine(sdr["ID"].ToString() + "      " + sdr["Name"] + "      " + sdr["Sex"] + "      "
+ sdr["Age"].ToString() + "      " + sdr["Class"].ToString());
    }
}
//增加学生信息
case 2:
    1. 创建数据库连接 2. /定义数据库连接信息 3. 打开链接 4. 打开数据库
Console.WriteLine("*****");
Console.WriteLine("*****      请输入要添加的学生学号      *****");
string ID1 = Console.ReadLine();
Console.WriteLine("*****      请输入学生姓名      *****");
string Name1 = Console.ReadLine();
Console.WriteLine("*****      请输入学生性别      *****");
string Sex1 = Console.ReadLine();
Console.WriteLine("*****      请输入学生年龄      *****");
string Age1 = Console.ReadLine();
Console.WriteLine("*****      请输入学生班级      *****");
string Class1 = Console.ReadLine();
Console.WriteLine("*****      请输入学生密码      *****");
Console.WriteLine("*****");
string Passsword1 = Console.ReadLine();
string MyInsert = "select count(ID) from Student where ID=" + ID1;//自定义 sql 查询语言
SqlCommand Mycommand = new SqlCommand(MyInsert, MyConnection);
MyInsert = string.Format("Insert into Student values(' {0}', ' {1}', ' {2}', ' {3}', ' {4}', ' {5} )", ID1,
Name1, Sex1, Age1, Class1, Passsword1);
SqlCommand sql = new SqlCommand(MyInsert, MyConnection);//sqlcommand 执行 sql 命令, 注意函参
int j = sql.ExecuteNonQuery();
    if (j > 0)
        Console.WriteLine("*****      添加成功      *****");
    else
        Console.WriteLine("*****      添加失败      *****");
//显示添加后的结果
1. 创建数据库连接 2. 定义数据库连接信息 3. 打开链接 4. 打开数据库
string sqlStr = "select ID from dbo.Student";
SqlCommand comm = new SqlCommand(sqlStr, con12);
SqlDataReader sdr = null;
try
{
    sdr = comm.ExecuteReader();

```

```

while (sdr.Read())
{
    Console.WriteLine("学号: " + sdr["ID"].ToString());
}
}

//修改学生信息
//学号 case 1:
1. 连接字符串 2. 新建通道
Console.WriteLine("*****      请输入要修改的学生学号      *****");
string ID2 = Console.ReadLine();
Console.WriteLine("*****      请输入修改后的内容      *****");
string value2 = Console.ReadLine();
string strcommand = "update Student set ID = 'value2' where ID = 'ID2' ";
3. 打开通道
int intres = -1;
intres = cmd.ExecuteNonQuery();
conn.Close();
//姓名 case 2:
//性别 case 3:
//年龄 case 4:
//班级 case 5:

// 删除学生信息
1. 创建连接字符串 2. 连接对象
eLine("*****      请输入要删除的学生学号      *****");
string ID = Console.ReadLine();
string sql = "delete from dbo.Student where ID='" + ID + "'";
4. 创建 sqlcommand 对象
5. 打开数据库连接
int r = cmd.ExecuteNonQuery();
Console.WriteLine("*****      成功删除了 {0} 行数据      *****", r);
Console.ReadKey();
//显示删除后的结果
(ps: 其余模块一与之前源代码相似)

```