

浙江工业大学

数据库系统课程设计报告

2020 短学期



小组 第 16 小组

题目 高校成绩管理系统设计与实现

班级 计科 1802

学号 2018060062525

姓名 项潇

指导教师 陆忆红

所在学院 计算机学院

提交日期 2020. 7. 16

目录

高校成绩管理系统设计与实现	1
1、需求分析	1
1.1 数据需求描述	1
1.2 系统功能需求	3
1.3 其他性能需求	3
2、概念结构设计	3
2.1 局部 E-R 图	3
2.2 全局 E-R 图	5
2.3 优化 E-R 图	5
3、逻辑结构设计	6
3.1 关系模式设计	6
3.2 数据类型定义	6
3.3 关系模式的优化	6
4、物理结构设计	7
4.1 聚簇设计	7
4.2 索引设计	7
4.3 分区设计	7
5、数据库实施	7
5.1 基本表建立	8
5.2 视图的建立	12
5.3 索引的建立	14
5.4 触发器建立	14
5.5 建存储过程	16
6、应用系统开发与试运行	16
6.1 开发平台和开发环境介绍。	16
6.2 前台界面与后台数据库连接说明，代码实现。	17
6.3 系统各功能设计和运行界面截图。	17
7、实验总结	25
7.1 遇到的问题和解决的办法	25
7.2 系统设计的不足	25
7.3 进一步改进思路和体会	25

高校成绩管理系统设计与实现

小组组长： 章远 小组成员：项潇 方灿 张浩

小组详细分工情况：各自独立开发

1、需求分析

本系统用于可用于有效便利地管理学生教师课程相关信息,有效提高信息查询速度,操作简便。本系统基于 Microsoft SQL serve 的数据库系统,主要应用了 c#语言进行前台开发。

1.1 数据需求描述

本系统需要的数据:

专业: 专业号、专业名

班级: 班级号、专业号、班级人数

学生: 学号、班级号、姓名、性别、年龄、生源地、已修学分数

课程: 课程号、课程名、开设学期、学时、考核方式、学分

教师: 教师编号、姓名、性别、年龄、职称、联系电话

成绩: 学号、课程号、教师号、成绩

专业表: xiangx_major16

专业号: xx_mno16,char(10),primary key

专业名: xx_mname16, char(20),

班级表: xiangx_class16

班级号: xx_cno16, char(10),primary key

专业号: xx_mno16, char(10),foreign key

班级人数: xx_classnum16,int

学生表: xiangx_student16

学号: xx_sno16, char(10),primary key

班级号: xx_cno16, char(10),foreign key

姓名: xx_sname16, char(20),not null

性别: xx_ssex16 ,char(4),

年龄: xx_sage16 ,int

生源地: xx_sbp16, char(20)

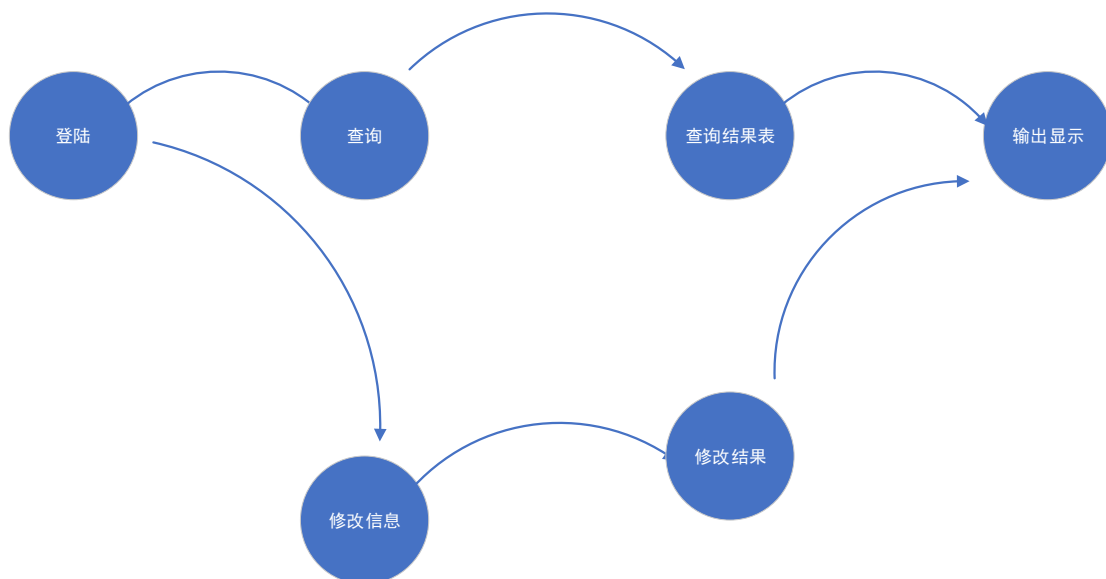
已经修学分数: xx_scredit16,int

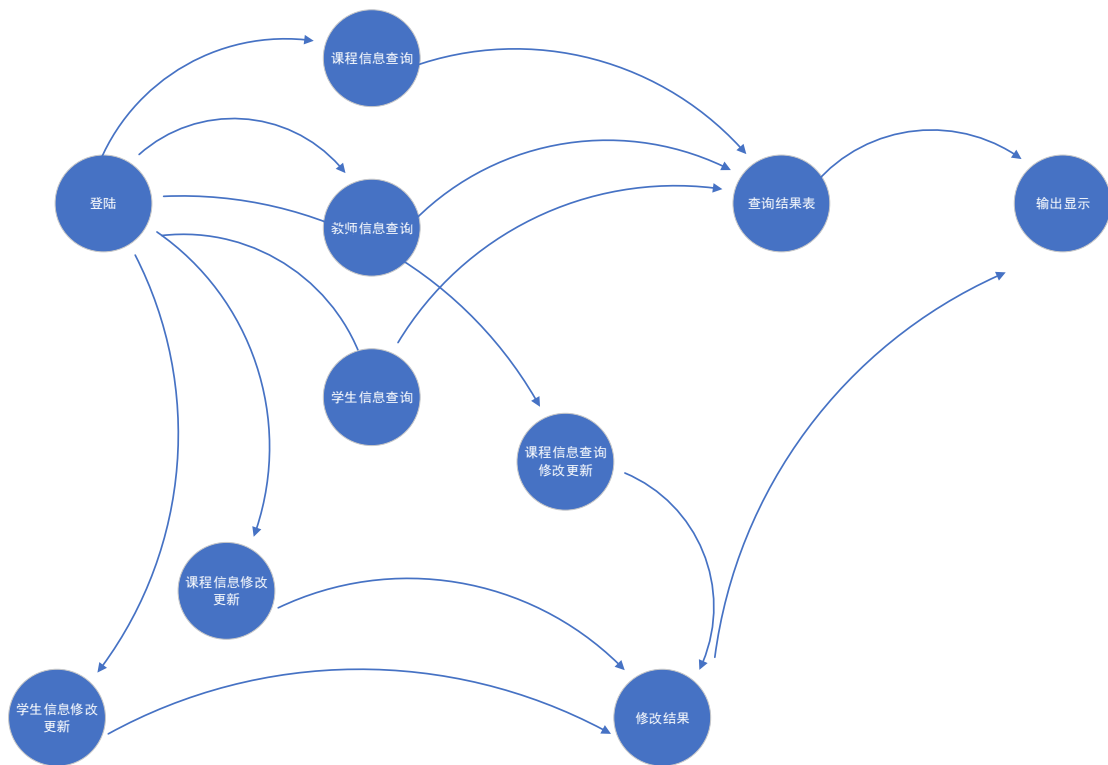
课程表: xiangx_course16
课程号: xx_ccno16, char(10),primary key
课程名: xx_ccname16 char(20)
开设学期: xx_ccopen16,char(10),
学时: xx_cchour16,int
考核方式: xx_ccway16,char(20),
学分: xx_cccredit16,int

教师表: xiangx_teacher16
教师编号: xx_tno16, char(10),primary key
姓名: xx_tname16, char(20),
性别: xx_tsex16, char(4)
年龄: xx_tage16, int
职称: xx_tposition16, char(20),
联系电话: xx_tcontact16, char(20),

成绩表: xiangx_record16
学号: xx_sno, char(10),foreign key
课程号: xx_ccno16, char(10),foreign key
教师号: xx_tno16, char(10),foreign key
成绩: xx_rresult16,int

数据流图:





1.2 系统功能需求

系统功能框架介绍，处理模块描述。

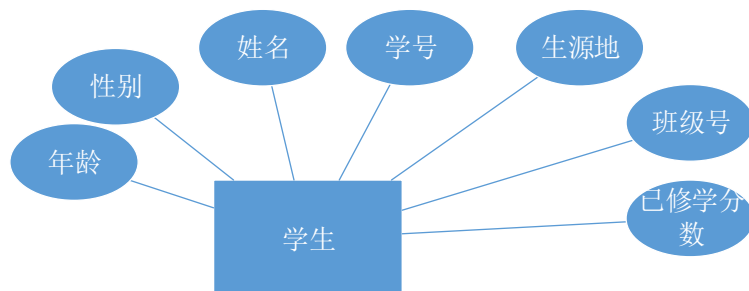
1.3 其他性能需求

其他性能分析，如并发用户数、响应时间和存储需求描述等。

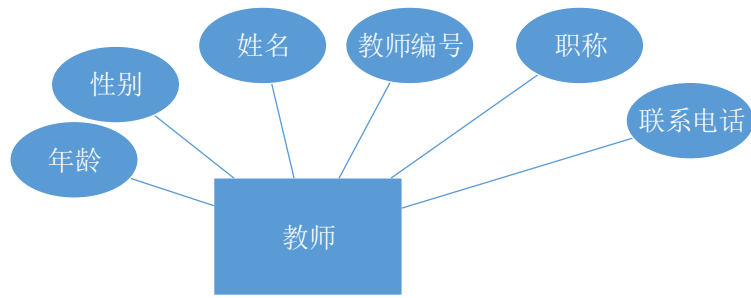
2、概念结构设计

2.1 局部 E-R 图

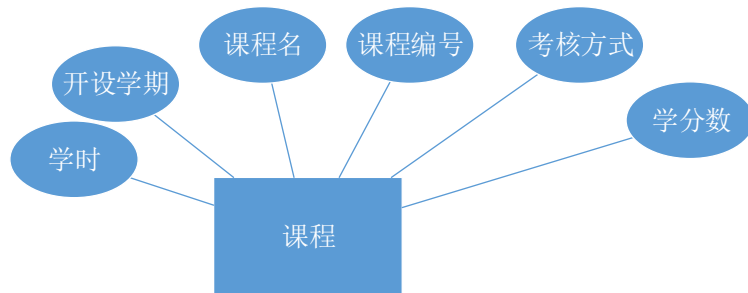
学生：



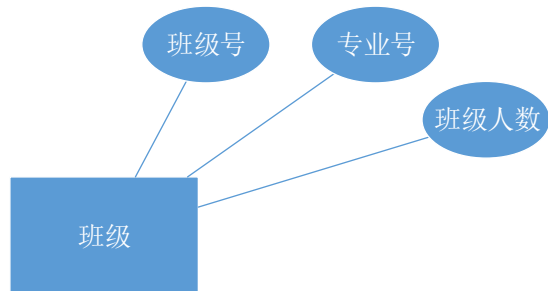
教师：



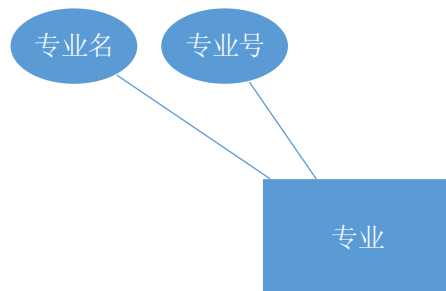
课程:



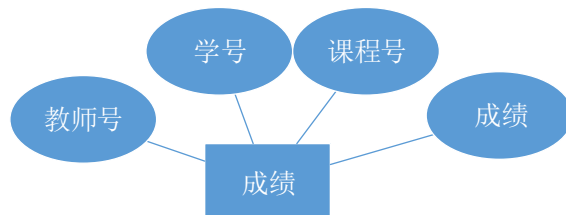
班级:



专业;

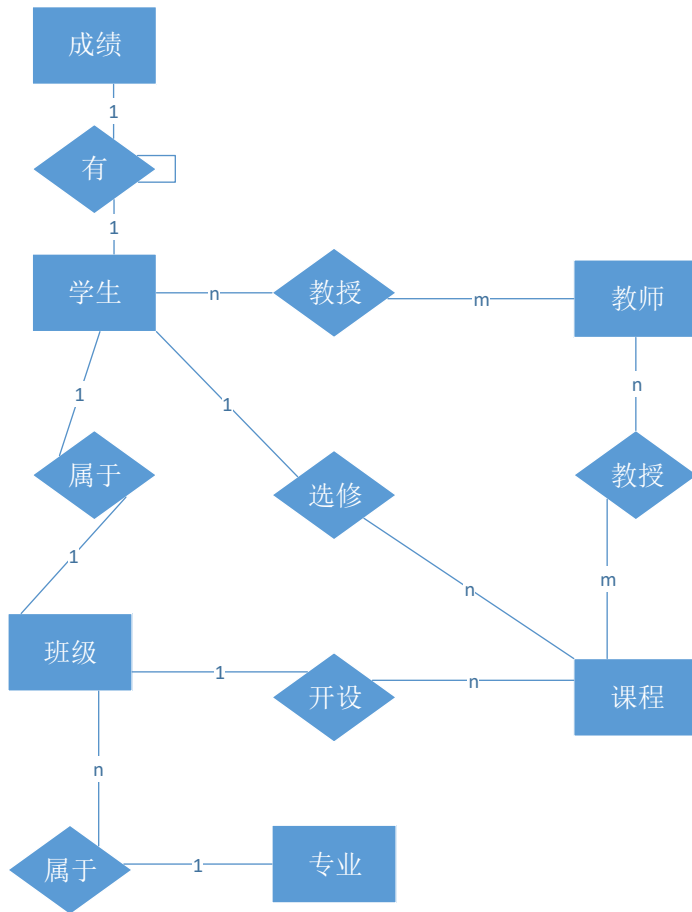


成绩:

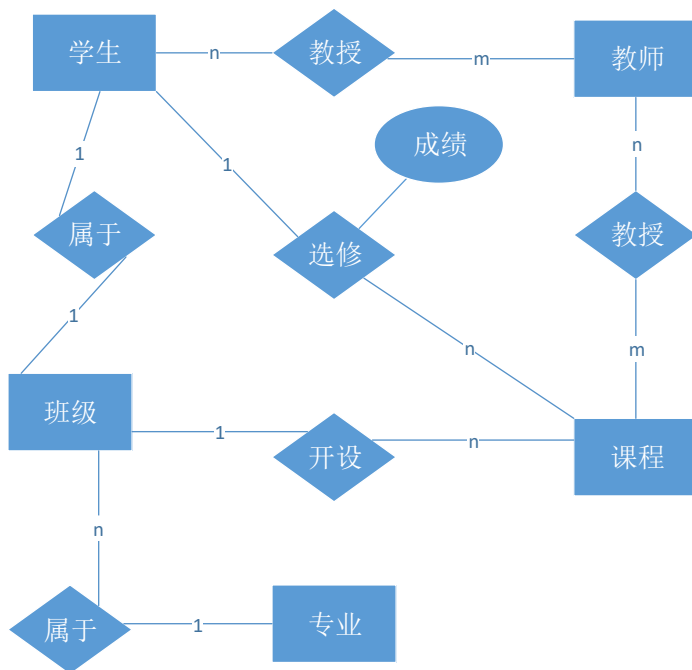


2.2 全局 E-R 图

这里关系较为复杂，此处不再添加属性



2.3 优化 E-R 图



3、逻辑结构设计

3.1 关系模式设计

专业（专业号，专业名）

班级（班级号，专业号，班级人数）

学生（学号，班级号，姓名，性别，年龄，生源地，已修学分）

成绩（课程号，教师号，学号，成绩）

课程（课程号，课程名，开放学期，学时，考核方式，学分）

教师（教师编号，姓名，性别，年龄，职称，联系电话）

3.2 数据类型定义

专业表：xiangx_major16

专业号：xx_mno16,char(10),primary key

专业名：xx_mname16, char(20),

班级表：xiangx_class16

班级号：xx_cno16, char(10),primary key

专业号：xx_mno16, char(10),foreign key

班级人数：xx_classnum16,int

学生表：xiangx_student16

学号：xx_sno16, char(10),primary key

班级号：xx_cno16, char(10),foreign key

姓名：xx_sname16, char(20),not null

性别：xx_ssex16 ,char(4),

年龄：xx_sage16 ,int

生源地：xx_sbp16, char(20)

已经修学分数：xx_scredit16,int

课程表：xiangx_course16

课程号：xx_ccno16, char(10),primary key

课程名：xx_ccname16 char(20)

开设学期：xx_ccopen16,char(10),

学时：xx_cchour16,int

考核方式：xx_ccway16,char(20),

学分：xx_cccredit16,int

教师表：xiangx_teacher16

教师编号：xx_tno16, char(10),primary key

姓名：xx_tname16, char(20),

性别：xx_tsex16, char(4)

年龄：xx_tage16, int

职称: xx_tposition16, char(20),
联系电话: xx_tcontact16, char(20),

成绩表: xiangx_record16
学号: xx_sno, char(10), foreign key
课程号: xx_ccno16, char(10), foreign key
教师号: xx_tno16, char(10), foreign key
成绩: xx_rresult16, int

3.3 关系模式的优化

专业 (专业号, 专业名)
班级 (班级号, 专业号, 班级人数)
学生 (学号, 班级号, 姓名, 性别, 年龄, 生源地, 已修学分)
成绩 (课程号, 教师号, 学号, 成绩)
课程 (课程号, 课程名, 开放学期, 学时, 考核方式, 学分)
教师 (教师编号, 姓名, 性别, 年龄, 职称, 联系电话)

4、物理结构设计

4.1 聚簇设计

聚簇索引: xiangx_Students16(xx_Sno16), xiangx_Class16(xx_Cno16),
xiangx_Major16(xx_Mno16), xiangx_Teacher16(xx_Tno16),
xiangx_Course16(xx_Ccno16)。这几列最经常查询, 基本不更新。。

4.2 索引设计

无

4.3 分区设计

无

5、数据库实施

(全部操作都要在 SQL Server 的查询分析器或 SQL Server 的 SSMS 环境中用命令实现, 并要求截图) 比如如图 5-1 所示。

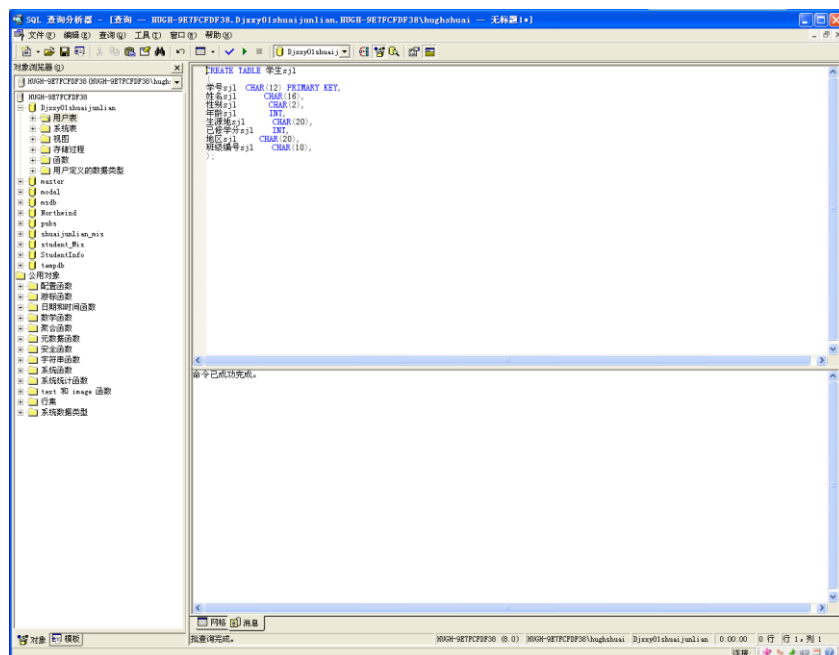
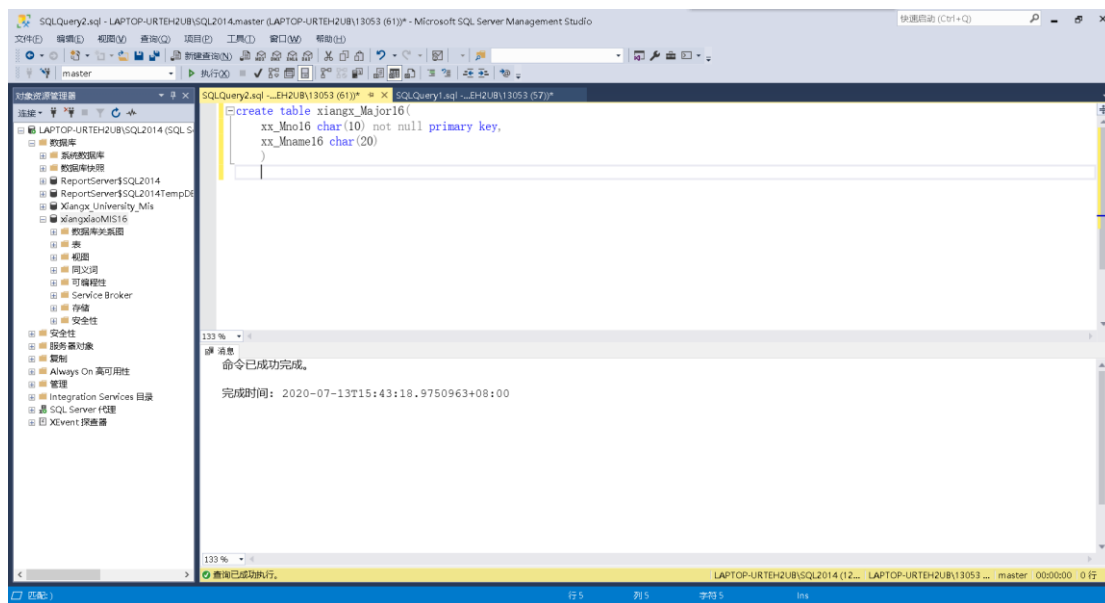


图 5-1 创建基本表”学生 sjl”

5.1 基本表建立

//建立专业表

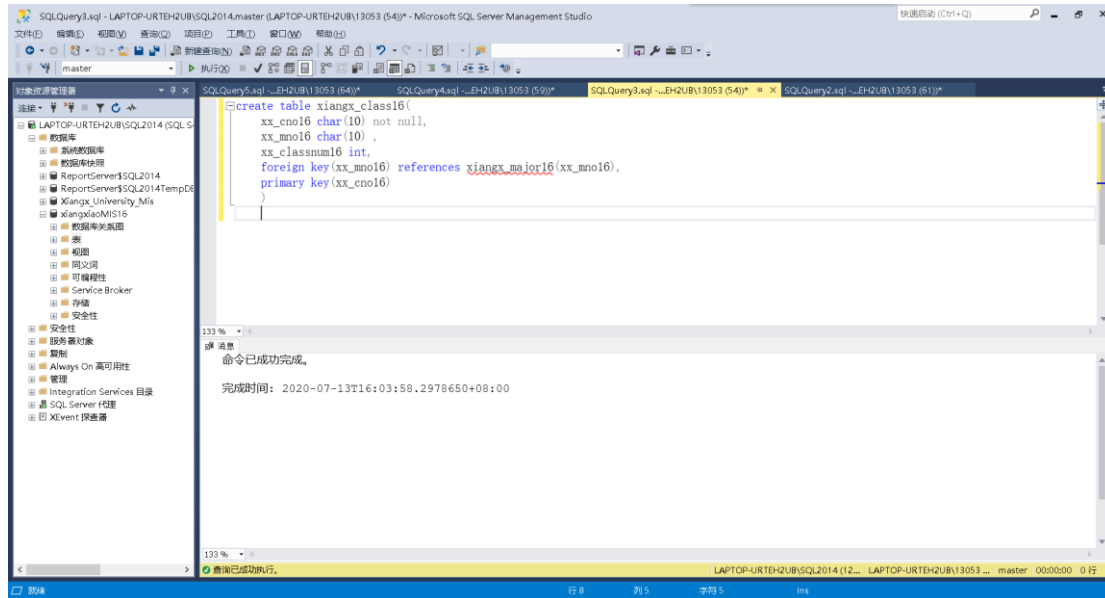
```
create table xiangx_Major16(  
    xx_Mno16 char(10) not null primary key,  
    xx_Mname16 char(20)  
)
```



//建立班级表

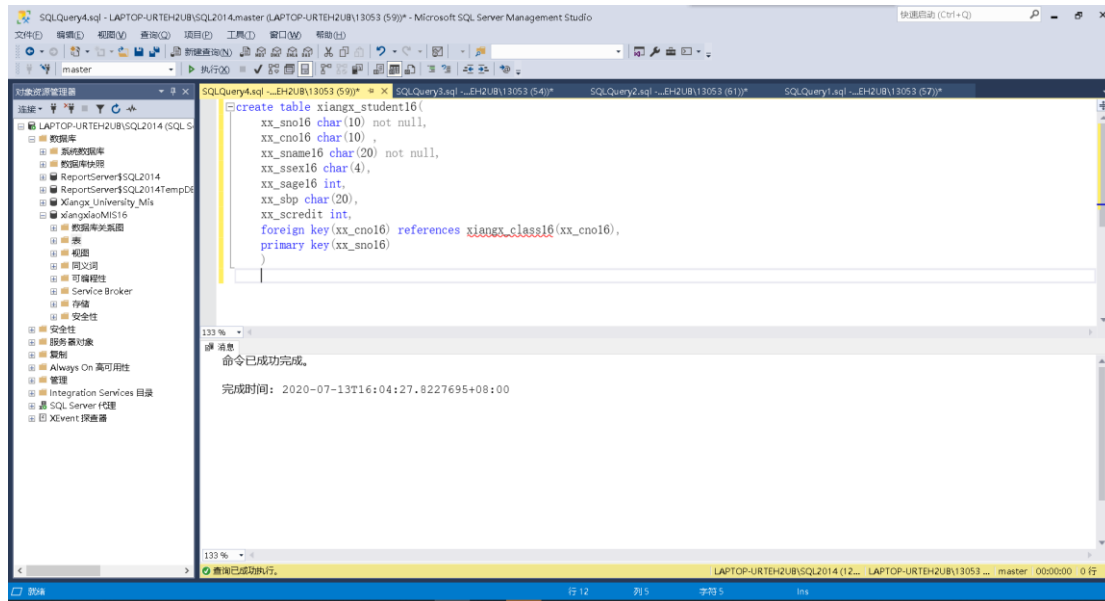
```
create table xiangx_class16(  
    xx_cno16 char(10) not null,
```

```
xx_mno16 char(10) ,  
xx_classnum16 int,  
foreign key(xx_mno16) references xiangx_major16(xx_mno16),  
primary key(xx_cno16)  
)
```



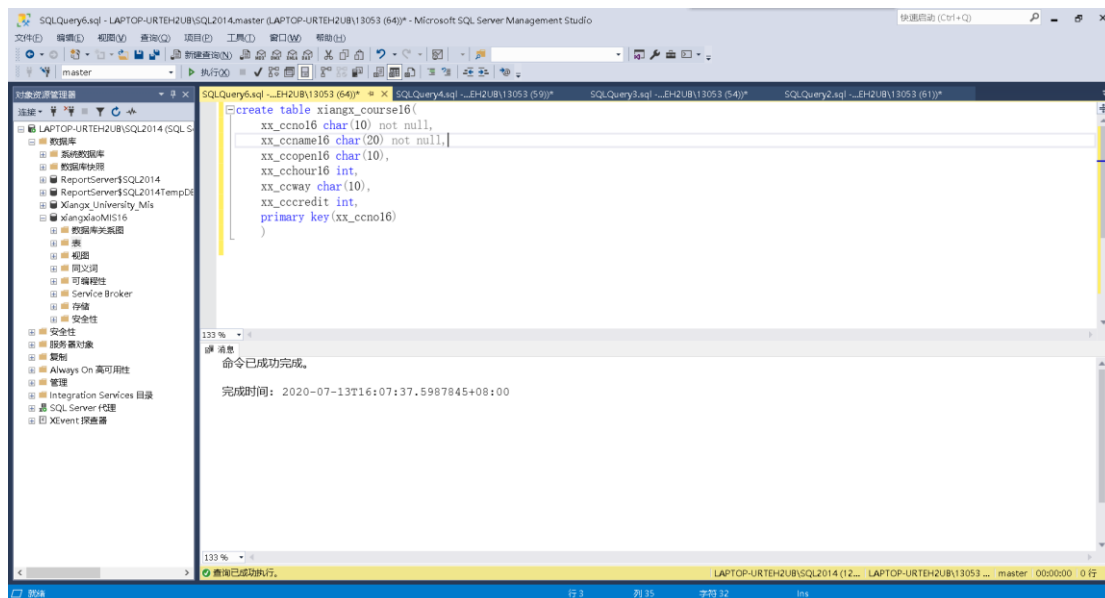
//建立学生表

```
create table xiangx_student16(  
    xx_sno16 char(10) not null,  
    xx_cno16 char(10) ,  
    xx_sname16 char(20) not null,  
    xx_ssex16 char(4),  
    xx_sage16 int,  
    xx_sbp char(20),  
    xx_scredit int,  
    foreign key(xx_cno16) references xiangx_class16(xx_cno16),  
    primary key(xx_sno16)  
)
```



//建立课程表

```
create table xiangx_course16(  
    xx_ccno16 char(10) not null,  
    xx_ccname16 char(20) not null,  
    xx_ccopen16 char(10),  
    xx_cchour16 int,  
    xx_ccway char(10),  
    xx_cccredit int,  
    primary key (xx_ccno16)  
)
```



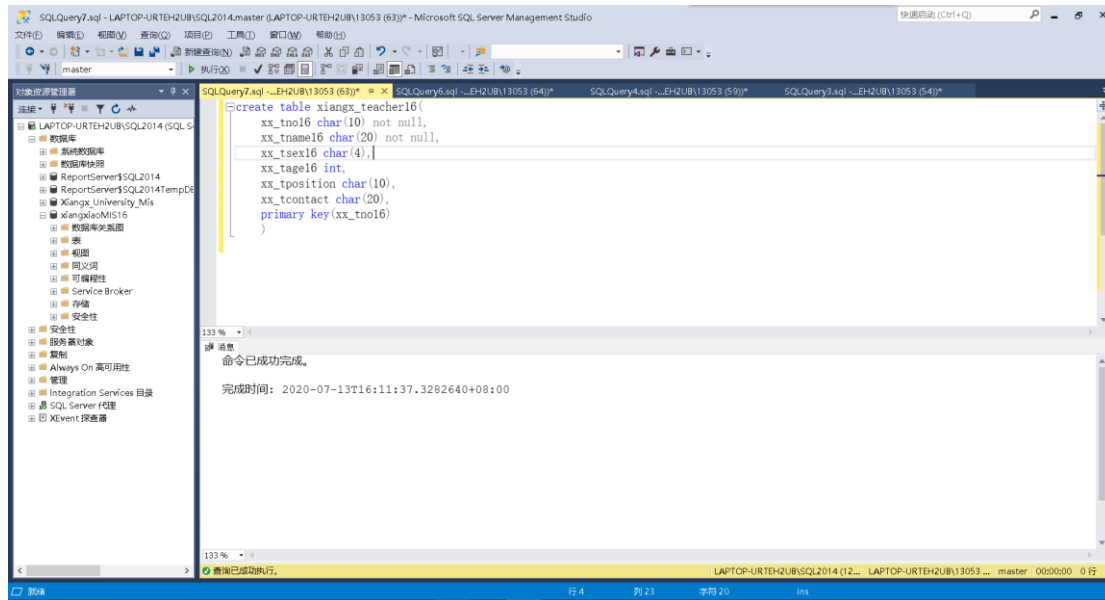
//建立教师表

```
create table xiangx_teacher16(  
    xx_tno16 char(10) not null,  
    xx_tname16 char(20) not null,
```

```

xx_tsex16 char(4),
xx_tage16 int,
xx_tposition char(10),
xx_tcontact char(20),
primary key(xx_tno16)
)

```

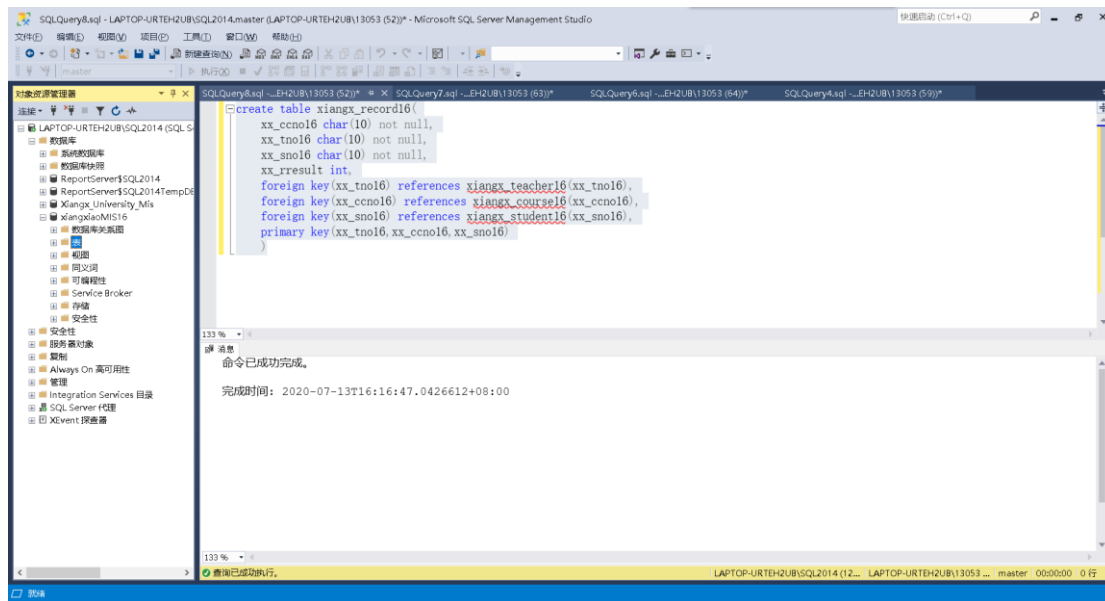


//建立成绩表

```

create table xiangx_record16(
    xx_ccno16 char(10) not null,
    xx_tno16 char(10) not null,
    xx_sno16 char(10) not null,
    xx_rresult int,
    foreign key(xx_tno16) references xiangx_teacher16(xx_tno16),
    foreign key(xx_ccno16) references xiangx_course16(xx_ccno16),
    foreign key(xx_sno16) references xiangx_student16(xx_sno16),
    primary key(xx_tno16,xx_ccno16,xx_sno16)
)

```



5.2 视图的建立

//视图 baogao

create view xiangx_baogao16

as

select

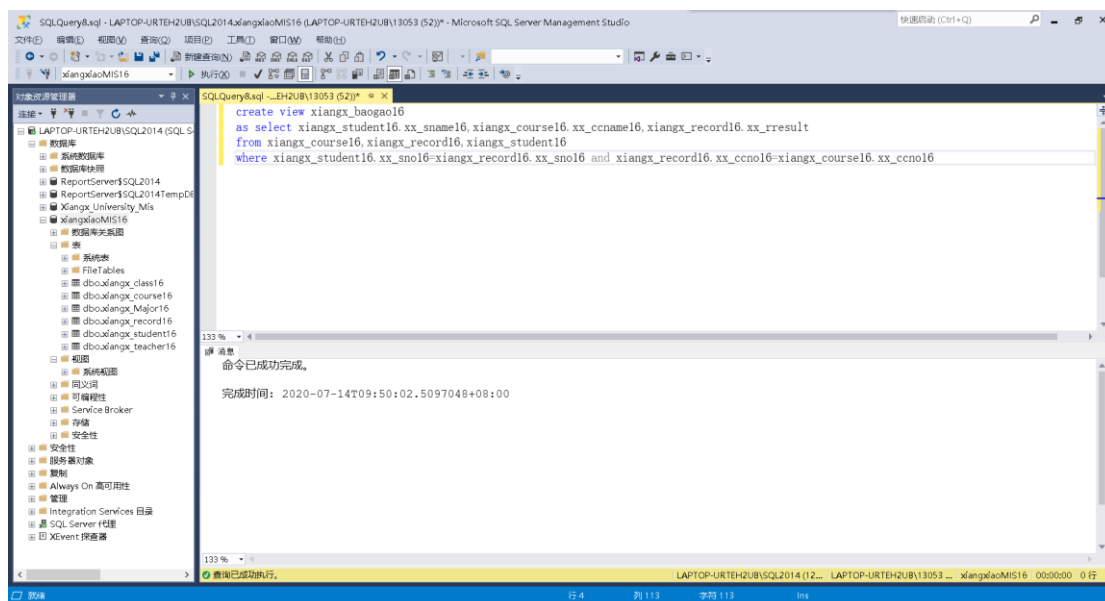
xiangx_student16.xx_sname16,xiangx_course16.xx_ccname16,xiangx_record16.xx_r
result

from xiangx_course16,xiangx_record16,xiangx_student16

where

xiangx_student16.xx_sno16=xiangx_record16.xx_sno16

and xiangx_record16.xx_ccno16=xiangx_course16.xx_ccno16



//rank 视图

create view xiangx_rankbycourse16

as

select

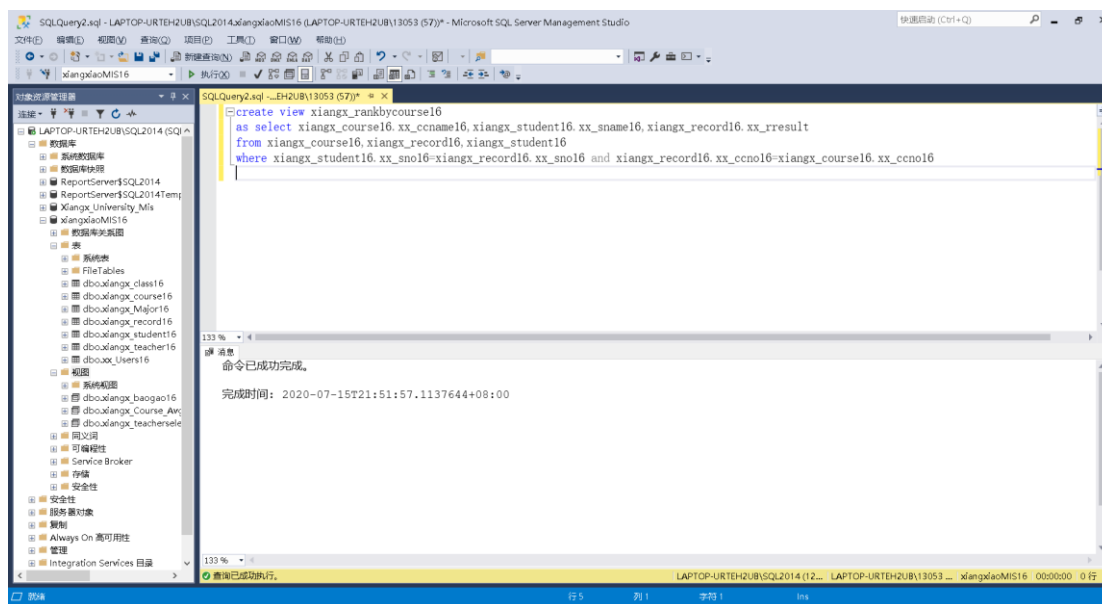
xiangx_course16.xx_ccname16,xiangx_student16.xx_sname16,xiangx_record16.xx_r
result

from xiangx_course16,xiangx_record16,xiangx_student16

where

xiangx_student16.xx_sno16=xiangx_record16.xx_sno16

and xiangx_record16.xx_ccno16=xiangx_course16.xx_ccno16



//视图 平均成绩 课程的

Create

view

xiangx_Course_AvgGrade16(xx_ccno16,xx_ccname16,xx_tname16,xx_AvgGrade16)

as

select xiangx_course16.xx_ccno16,xx_ccname16,xx_Tname16,AVG(xx_rresult)

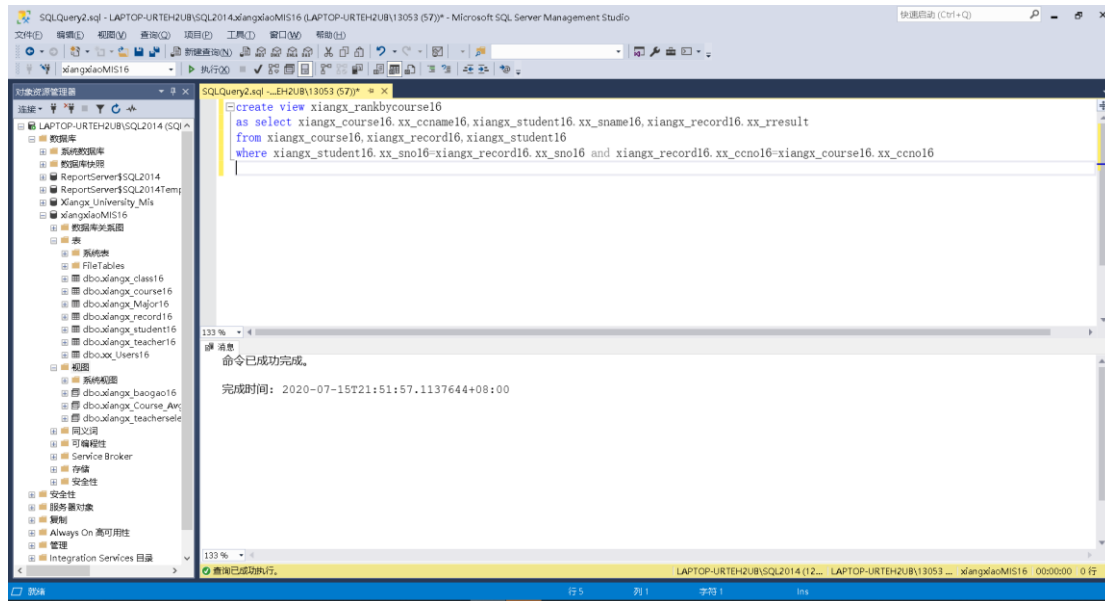
from xiangx_course16,xiangx_teacher16,xiangx_record16

where xiangx_record16.xx_ccno16=xiangx_course16.xx_ccno16 and

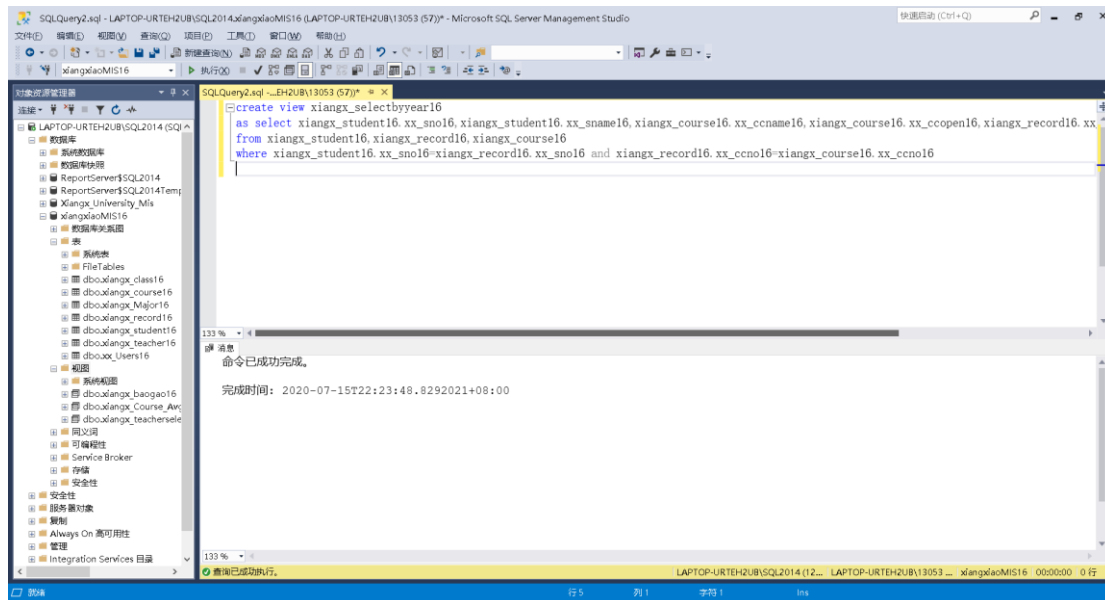
xiangx_course16.xx_ccno16=xiangx_record16.xx_ccno16 and

xiangx_record16.xx_tno16=xiangx_teacher16.xx_tno16

group by xiangx_course16.xx_ccno16,xx_ccname16,xx_Tname16



//建立按学年查询成绩信息视图



5.3 索引的建立

无

5.4 触发器建立

//触发器 更新学分

create trigger xiangx_update_Scredit16 on xiangx_record16

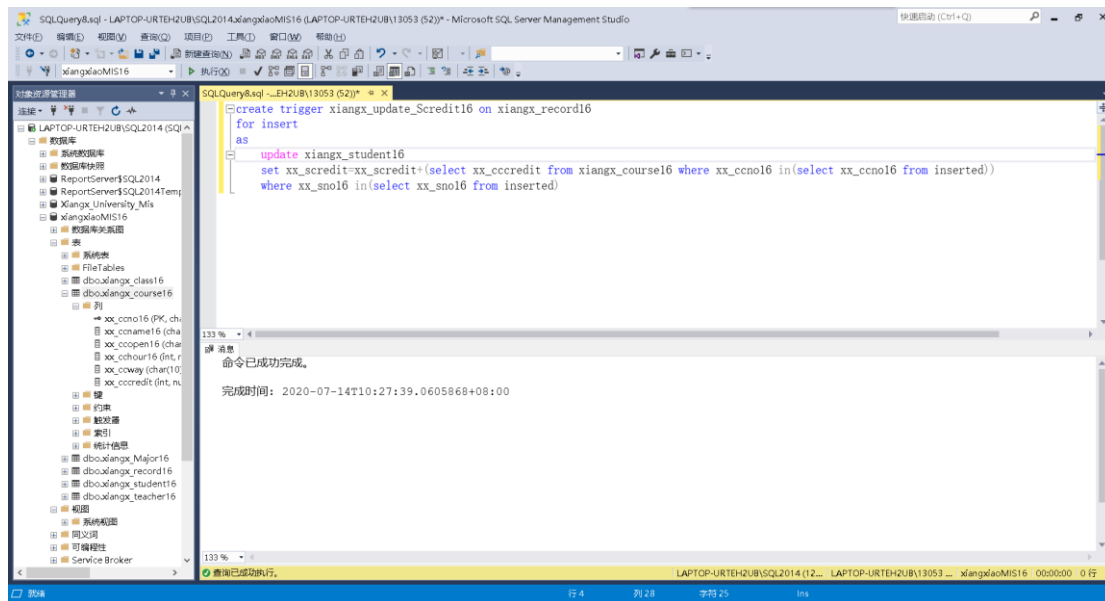
for insert

as

update xiangx_student16

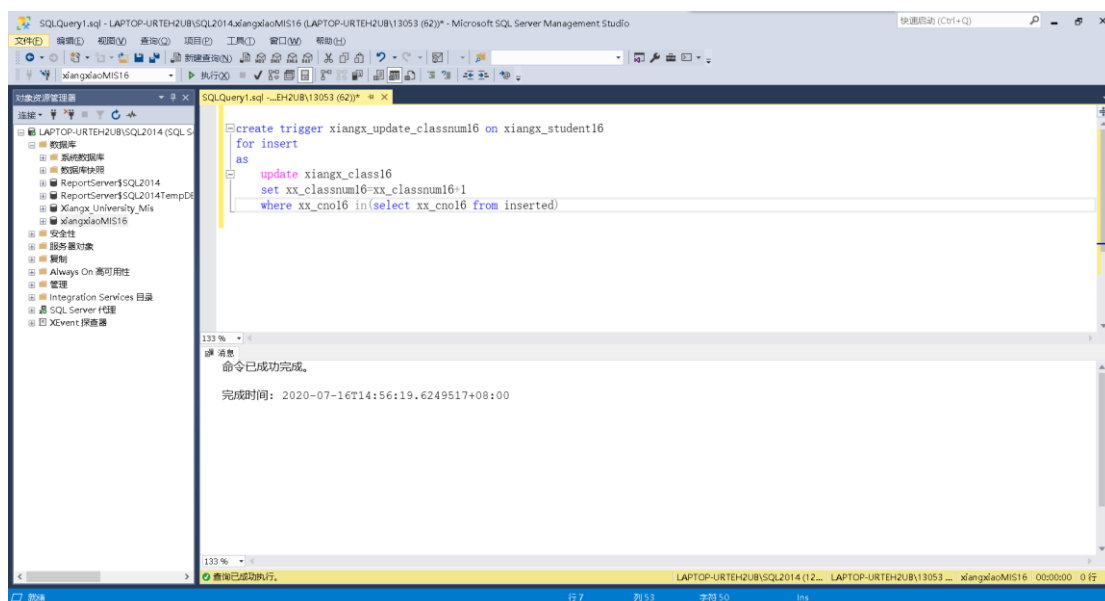
set xx_scredit=xx_scredit+(select xx_cccredit from xiangx_course16 where


```
xx_ccno16 in(select xx_ccno16 from inserted))
where      xx_sno16      in(select      xx_sno16      from
inserted)
```



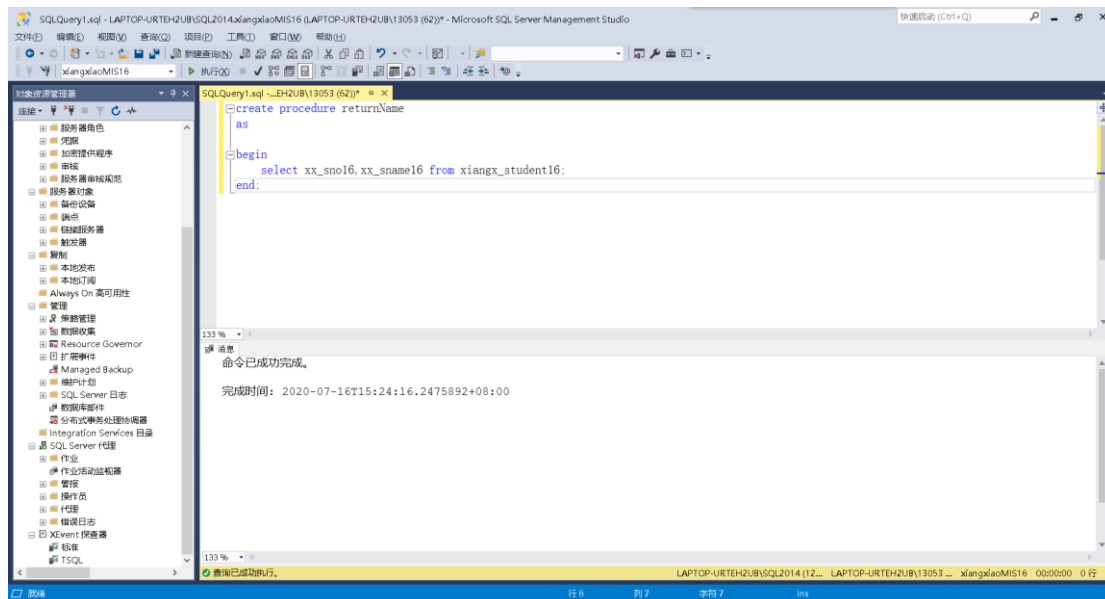
//班级人数自加触发器

```
create trigger xiangx_update_classnum16 on xiangx_student16
for insert
as
update xiangx_class16
set xx_classnum16=xx_classnum16+1
where xx_cno16 in(select xx_cno16 from inserted)
```

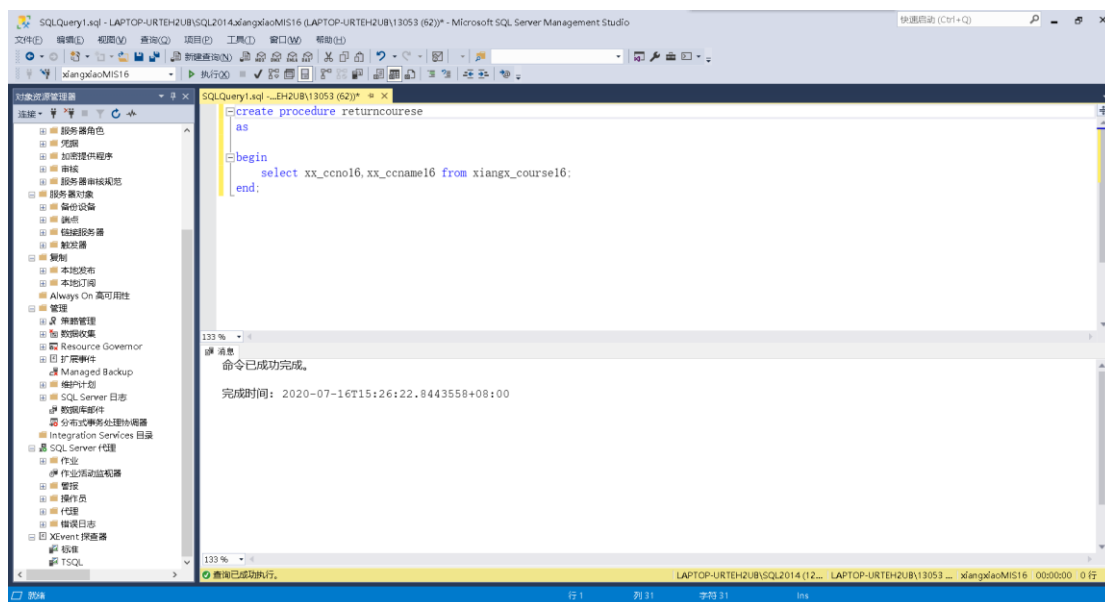


5.5 建存储过程

//名单



//课程表



6、应用系统开发与试运行

6.1 开发平台和开发环境介绍。

Microsoft Visual Studio 是 VS 的全称。VS 是美国微软公司的开发工具包系列产品。VS 是一个基本完整的开发工具集，它包括了整个软件生命周期所需要的大部分工具，如 UML 工具、代码管控工具、集成开发环境(IDE)等等。所写的目标代码适用于微软支持的所有平台

处理器: Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz 2.21 GHz
已安装的内存(RAM): 8.00 GB (7.88 GB 可用)
系统类型: 64 位操作系统, 基于 x64 的处理器

6.2 前台界面与后台数据库连接说明，代码实现。

```
public sqlConnect() {  
    if (conn == null) {  
        conn = new SqlConnection("Data Source=LAPTOP-URTEH2UB\\SQL2014;Initial  
Catalog=xiangxiaoMIS16;Integrated Security=True");  
        if (conn.State == ConnectionState.Closed) conn.Open();  
    }  
}
```

通过连接字符串连接数据库，连接字符串可从数据库属性信息里面获得。此处连接字符串为 Data Source=LAPTOP-URTEH2UB\\SQL2014;Initial Catalog=xiangxiaoMIS16;Integrated Security=True。

6.3 系统各功能设计和运行界面截图。

登陆界面结果如图所示

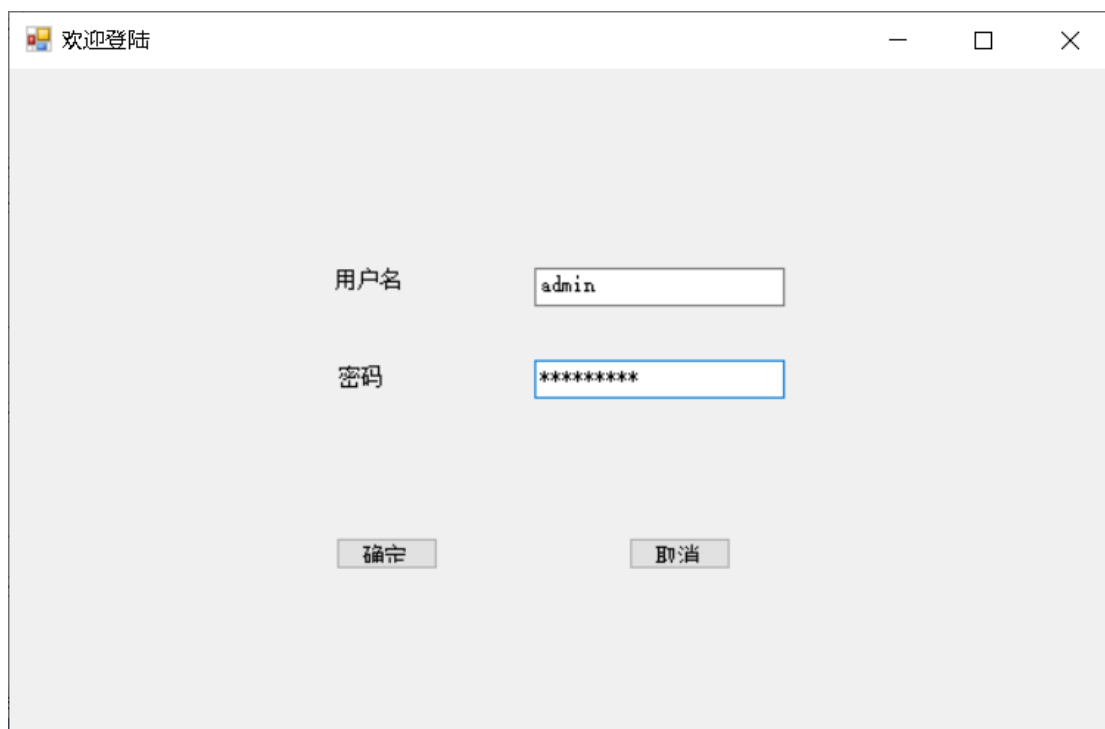


图 6-1 登陆界面展示图



图 6-2 主界面展示图



图 6-3 选课及学分统计展示图

需要进行统计的学年

groupBox1

xx_sno16	xx_sname16	...	xx_c
s05	沈六	...	体育
s01	张三	...	体育
s03	李四	...	计算
s04	王五	...	体育

图 6-4 学年统计成绩界面展示图

课程号 教师号

学号 成绩

groupBox1

xx_ccno16	xx_tno16	xx_sno16	xx_rresu
cc01	t01	s01	88
cc01	t01	s02	80
cc01	t01	s03	90
cc01	t01	s04	84

图 6-5 学生成绩查询更新界面展示图




学生信息管理窗口

学号: 班级号: 姓名: 性别:

年龄: 出生地: 已修学分:

	学号	班级号	学生姓名	性别	年龄
▶	s01	c01	张三 ...	男	20
	s02	c02	张四 ...	女	20
	s03	c02	李四 ...	男	20

图 6-6 学生信息界面展示图



地区人数统计

需要进行统计的地区人数

groupBox1

	地区	地区人数
▶	浙江杭州 ...	1
	浙江嘉兴 ...	1
	浙江台州 ...	1
	浙江温州 ...	1
	浙江诸暨 ...	1

图 6-7 地区人数统计图

课程管理查询窗口

课程号 课程名 开设学期

学时 考核方式 学分数

课程号	课程名	开课学期	课时
cc01	数据库 ...	第二学期	48
cc02	算法分析与设...	第二学期	48
cc03	计算机组成原...	第一学期	48
cc04	体育	第一学期	48

图 6-8 课程信息界面展示图

课程均分查询

需要进行查询的课程

xx_ccno16	xx_ccname16	xx_tname
cc01	数据库 ...	萧重儿
cc02	算法分析与设...	萧炎
cc03	计算机组成原理	
cc04	体育 ...	云山

图 6-9 课程均分界面展示图



图 6-10 课程排名界面展示图

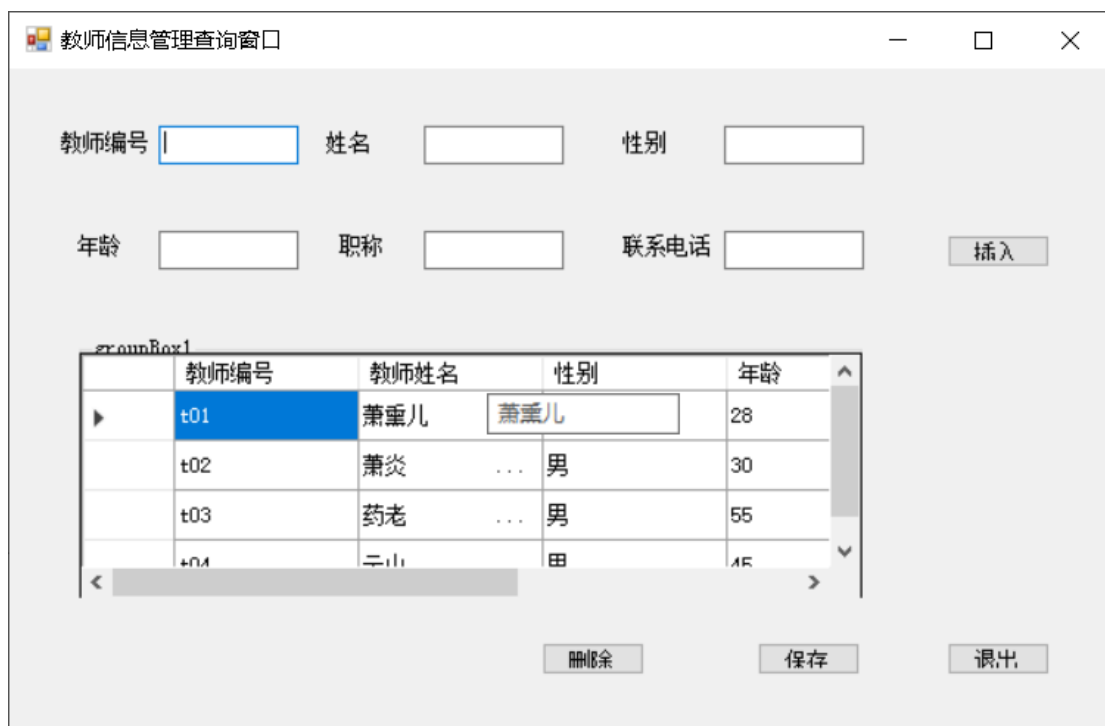


图 6-11 教师信息界面展示图



图 6-12 教师任课查询界面展示图

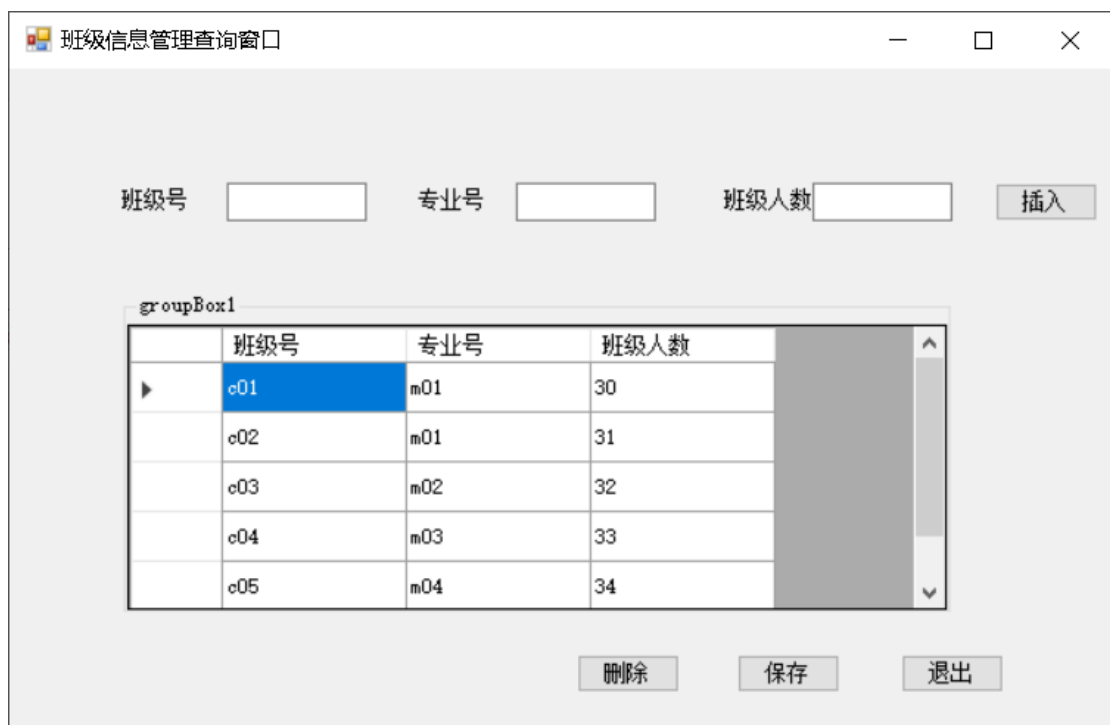


图 6-13 班级信息插入查询界面展示图



图 6-14 班级课程查询界面展示图

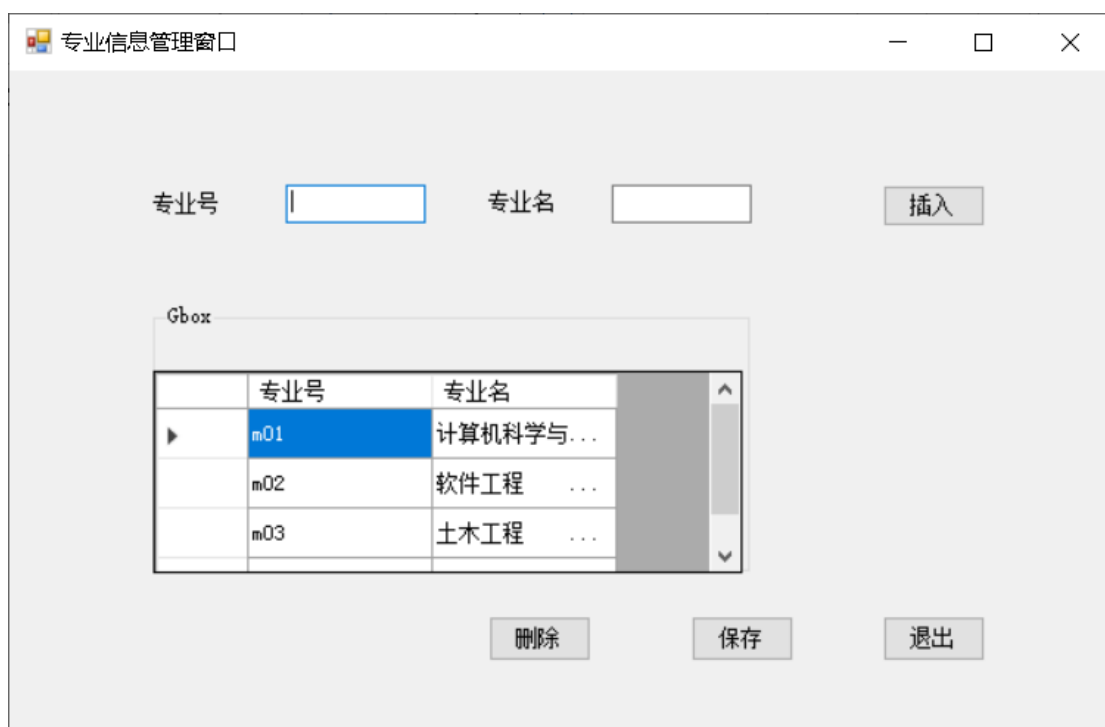


图 6-15 专业信息插入查询界面展示图

7、实验总结

7.1 遇到的问题 and 解决的办法

问题一：不知道如何连接数据库

通过去 csdn 论坛和百度等搜索引擎查找相关信息，阅读理解之后，采用了连接字符串的方法来连接数据库；在期间找不到连接字符串，后来发现了从数据库的属性进去就可以找到，最后解决了这个问题。通过对示例代码的阅读和理解，最后完成了数据库的连接。

问题二：查询结果不显示更新？

发现是在显示函数编写的时候显示的内容没有重新刷新，最后补上了显示更新语句，重新编译显示最后更新成功。

问题三：如何建立学生表，教师表和课程表之间的关系？

通过成绩表存在的学号、教师号、课程号作为外键连接起三个表。

7.2 系统设计的不足

因为时间比较紧促，还未按照不同身份来进行不同权限的设置与编写。

7.3 进一步改进思路和体会

思路目前有两种，

第一种就是从登录界面增加一项身份的 textbox，输入身份，通过数据库中的 user 表比对来确认身份，再添加一个判断功能，身份不同展示不同的主界面，进行的操作不同。

第二种就是从内部进行修改，使用的是同一套主界面，那就是从函数运行的内部进行修改，怎加判断语句，如果设置每个身份的人不同的权限即可。

通过此次实验，让我对数据库有了更加深的认知，数据库不仅是用来存储数据的，还可以大幅度的优化查询和管理的效率，同时我也认识到了数据库系统建立的不容易，需要设计数据库考虑需求的同时，还需要构思并编写数据库的前台应用程序，需要让两者相得益彰，同时需要考虑到数据的逻辑性等等，这次实践让我对数据库的认知更加深入了，也让我习得了 c# 的部分语法，学到了不少的新知识。