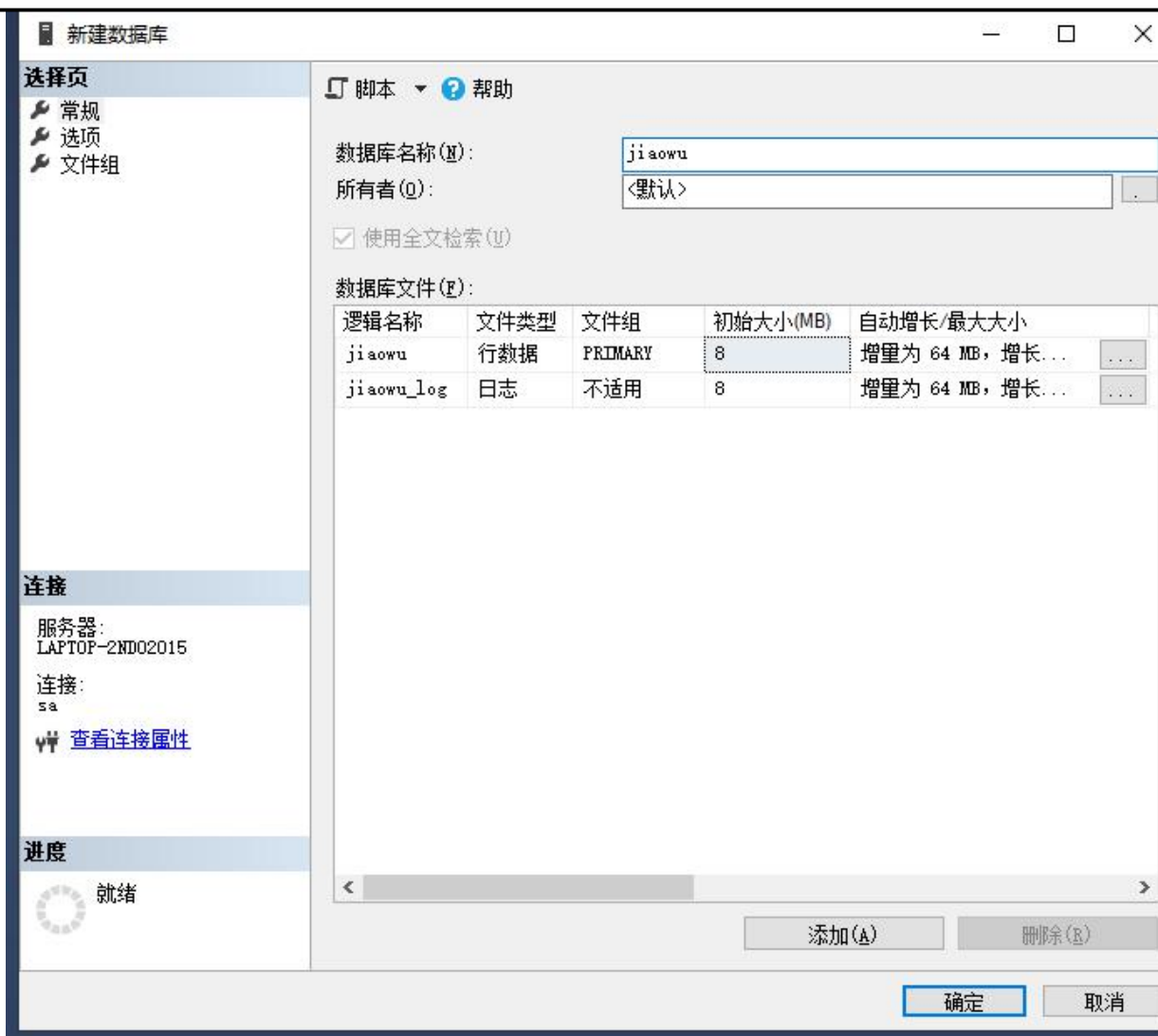


实 验 报 告

学 号	1902001 1038	姓 名	岳宇轩	专业班级	19 级计算机科学与技术慧与班
课程名称	《数据库系统》			学期	2022 年春季学期
任课教师	刘洁	完成日期	2022/3/14	实验课时间	2022/3/14
实 验 名 称	实验二：使用商用数据库 SQL server 或 MySQL 实现教务系统的数据库设计				
<p>一、实验要求（10%）</p> <p>在教务系统的数据库设计时，完成以下内容：</p> <ol style="list-style-type: none"> 1.创建教务系统各数据库表，包括:Student Course, SC(数据库逻辑结构设计参考课本 Page 79)。设置主键和外键，以及用户定义的完整性约束条件。 2.输入数据，体验实体完整性、参照完整性、以及用户定义的完整性约束。 3.实现简单的数据浏览以及数据增删改。 4.使用实验报告模板，完成实验报告。 <p>二、实验内容及步骤（80%）</p> <ol style="list-style-type: none"> 1. 首先完成 SQL Sever 的安装 安装步骤直接参考博客 https://blog.csdn.net/CHQC388/article/details/104565370 完成 SQL Sever 2019 的安装。 2. 完成对 SQL Sever 的配置 配置步骤直接参考博客 https://blog.csdn.net/CHQC388/article/details/104565370 完成对 SQL Sever 2019 的配置 3. 创建数据库 右键数据库，选择新建数据库，输入数据库名称为 jiaowu，点击确定 					

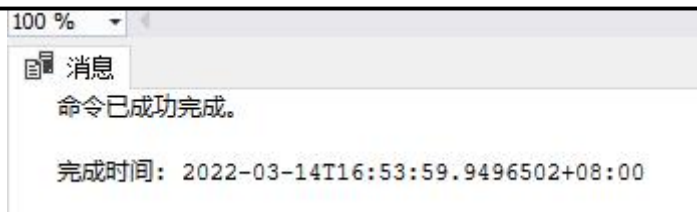


4. 完成 Student 表的设计

- ① 点击新建查询
- ② 输入如下 SQL 语句:

```
use jiaowu --使用教务数据库
go
CREATE TABLE Student --创建表，表名为Student
( Sno CHAR(9) PRIMARY KEY, /*列级完整性约束，Sno是主码*/
  Sname CHAR(20) UNIQUE, /*Sname取唯一值*/
  Ssex CHAR(2),
  Sage SMALLINT,
  Sdept CHAR(20)
);
```

- ③ 点击执行
- ④ 显示如下消息



⑤ 可以看到成功建立了学生表

列名	数据类型	允许 Null 值
Sno	char(9)	<input type="checkbox"/>
Sname	char(20)	<input checked="" type="checkbox"/>
Ssex	char(2)	<input checked="" type="checkbox"/>
Sage	smallint	<input checked="" type="checkbox"/>
Sdept	char(20)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

- 表的主键是学号，学号为 9 位字符。主键不能取空值；
- 姓名不能重复，为 20 位字符；
- 性别设置为两个字符；
- 年龄为 smallint 类型，数据范围是从 -2^{15} (-32,768) 到 $2^{15} - 1$ (32,767) 的整型数据。存储大小为 2 个字节。足够表示年龄；
- 院系设置为 20 字符长度；

5. 完成 Course 表的设计

仿照学生表建立的方法，编写并执行如下 SQL 语句：

```
use jiaowu
go
CREATE TABLE Course
(Cno CHAR(4) PRIMARY KEY,
Cname CHAR(40) NOT NULL,
Cpno CHAR(4),
Ccredit SMALLINT,
FOREIGN KEY (Cpno) REFERENCES Course (Cno)
);
```

可以看到成功建立了课程表，如下：

列名	数据类型	允许 Null 值
Cno	char(4)	<input type="checkbox"/>
Cname	char(40)	<input type="checkbox"/>
Cpno	char(4)	<input checked="" type="checkbox"/>
Ccredit	smallint	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

课程号是主码，为 4 个字符，不允许空值；

课程名称不允许空值；

关键语句：FOREIGN KEY (Cpno) REFERENCES Course (Cno) 的含义是建立表级完整性约束，即参

照完整性规则，先修课号要么取空值，要么就必须与 Cno 列中的某一值对应。

6. 完成 SC 表的设计

编写并执行如下 SQL 语句：

```
SQLQuery1.sql - L...5.jiaowu (sa (51))* X LAPTOP-2ND0201...wu
use jiaowu
go
CREATE TABLE SC
(Sno CHAR(9),
Cno CHAR(4),
Grade SMALLINT,
PRIMARY KEY (Sno,Cno),
FOREIGN KEY (Sno) REFERENCES Student (Sno),
FOREIGN KEY (Cno) REFERENCES Course (Cno)
);
```

可以看到成功建立的选课关系的表

	列名	数据类型	允许 Null 值
🔑	Sno	char(9)	<input type="checkbox"/>
🔑	Cno	char(4)	<input type="checkbox"/>
	Grade	smallint	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

主码由两个属性构成，是 Sno 和 Cno

Sno 参照 Student 表的 Sno

Cno 参照 Course 表的 Cno

7. 成功建立了数据库和三张表：



8. 验证实体完整性:

当我尝试加入数据如下时:

LAPTOP-2ND02015...u - dbo.Student						SQLQuery1.sql - L...5.jiaowu (sa (52))*	
	Sno	Sname	Ssex	Sage	Sdept		
	123456789	岳宇轩	男	20	计算机		
	123456789	NULL	NULL	NULL	NULL		

系统报错:



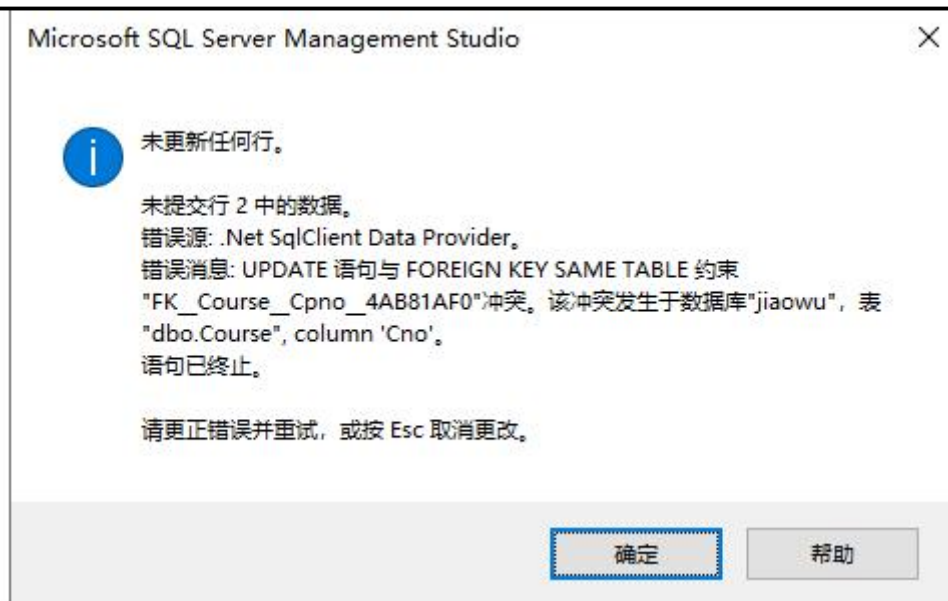
这是因为违反了实体完整性规则: 学生关系中, 学号是主码, 不能重复, 而我输入了两个学号都是 123456789 的学生, 因此操作无法完成。

9. 验证参照完整性:

当我尝试建立课程, 课程号是 1111, 没有先修课 (Cpno 取值为 NULL) 时, 可以成功, 这验证了参照完整性规则的第一条

LAPTOP-2ND0201...wu - dbo.Course				LAPTOP-2ND02015...u - dbo.St	
	Cno	Cname	Cpno	Ccredit	
	1111	数据库系统	NULL	4	
	2222	高级数据库	4567	10	

当我尝试建立课程, 课程号是 2222, 先修课的课程号是 4567 (Course 表中 Cno 列没有此值) 时, 系统提示:



这验证了参照完整性规则的第二条。
当我将其 Cpno 修改为已存在的 1111 后，建立成功：

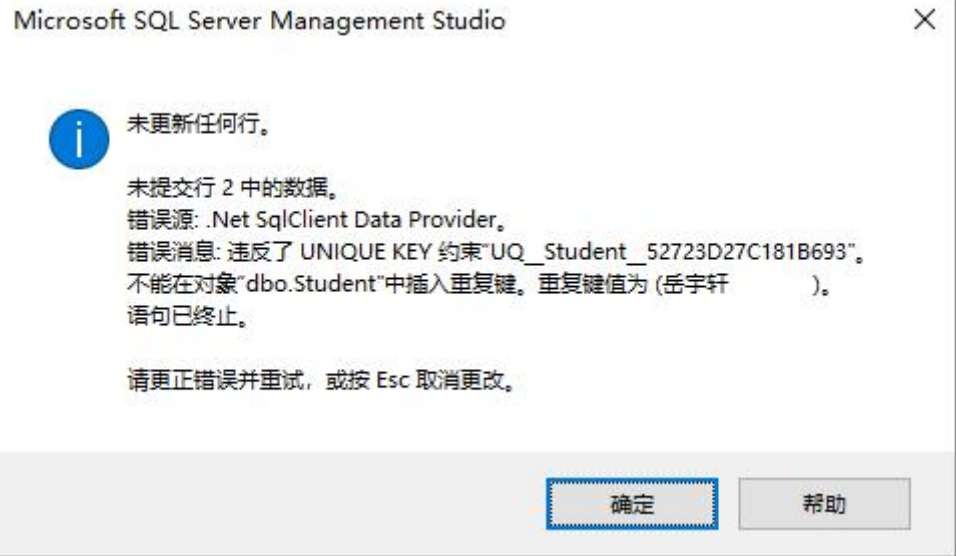
LAPTOP-2ND0201...wu - dbo.Course					LAPTOP-2ND02015...u - dbo.St				
	Cno	Cname		Cpno	Ccredit				
	1111	数据库系统 ...		NULL	4				
	2222	高级数据库 ...		1111	10				

10. 验证用户定义完整性

我们定义了在学生关系中，姓名不能重复，属于用户定义的完整性约束。

LAPTOP-2ND0201...wu - dbo.Course				LAPTOP-2ND02015...u - dbo.Student			
	Sno	Sname	Ssex	Sage	Sdept		
	123456789	岳宇轩 ...	男	20	计算机 ...		
	987654321	岳宇轩	男	21	物理		

可以看到，重复的学生姓名是不允许的，因为违反了用户定义完整性约束。
系统报错如下：



11. 对数据库表进行浏览，增删改：

浏览：

```
SQLQuery2.sql - L...5.jiaowu (sa (52))*
use jiaowu
go
select * from Student
```

结果：

100 %					
结果 消息					
	Sno	Sname	Ssex	Sage	Sdept
1	123456782	NULL	NULL	NULL	NULL
2	123456789	岳宇轩	男	20	计算机
3	987654321	岳宇轩2	男	21	物理

增：

```
SQLQuery2.sql - L...5.jiaowu (sa (52))*
use jiaowu
go
--select * from Student
Insert into Student values('111111111', '小小红', '女', 19, '计算机')
```

结果：

100 %

	Sno	Sname	Ssex	Sage	Sdept
1	111111111	小小红	女	19	计算机
2	123456782	NULL	NULL	NULL	NULL
3	123456789	岳宇轩	男	20	计算机
4	987654321	岳宇轩2	男	21	物理

删:

```
SQLQuery2.sql - L...5.jiaowu (sa (52))*
use jiaowu
go
--Insert into Student values('111111111', '小小红', '女', 19, '计算机')
delete from Student where Sno = '111111111'
select * from Student
```

结果:

100 %

	Sno	Sname	Ssex	Sage	Sdept
1	123456782	NULL	NULL	NULL	NULL
2	123456789	岳宇轩	男	20	计算机
3	987654321	岳宇轩2	男	21	物理

改:

```
SQLQuery2.sql - L...5.jiaowu (sa (52))*
use jiaowu
go
--Insert into Student values('111111111', '小小红', '女', 19, '计算机')
--delete from Student where Sno = '111111111'
update Student set Sname='岳更新', Sage='120' where Sno='987654321'
select * from Student
```

结果:

100 %

	Sno	Sname	Ssex	Sage	Sdept
1	123456782	NULL	NULL	NULL	NULL
2	123456789	岳宇轩	男	20	计算机
3	987654321	岳更新	男	120	物理

三、心得总结（写出自己在完成实验过程中遇到的问题、解决方法，以及体会、收获等）（10%）

实验一开始，我先对照网上的教程自主安装了 SQL Sever 2019.

在创建表的过程中，我一开始是使用可视化窗口进行操作的，但是这样并不能直观的显示参照完整性约束和用户定义完整性约束，因此我又从网上找到了使用 SQL 语句创建表的方法。使用 SQL 语句创建表的好处是可以避免可视化界面操作不规范。在创建表的过程中，在 .sql 文件的开始要先输入 use jiaowu，表示要使用的数据库是 jiaowu，接着在下一行输入 go，再下一行才输入 SQL 语句。

通过实例操作，我也更加清楚了实体完整性，参照完整性和用户定义完整性。实体完整性要求主码不能为空值。参照完整性要求外码要么取空，要么取所参照的某个元组的主码的值。比如在课程关系中，先修课号要么取空值，表示该课程无先修课，要么填写一个已经存在的课程号。用户定义完整性是用户自己设定的对表的约束，比如学生关系中设置学生姓名 Sname 为 UNIQUE，表示不能重复，后来实践中插入重复姓名的学生会报错，也验证了这一点。

对于浏览表中的数据，我是用 select * from TABLENAME 的方法，执行之后可以看到表中的数据。对于增、删、改，我分别使用 Insert, delete, update 方法，搭配 set, where 等关键字，实现简单的数据操作。

总体来说，通过此次实验，我对于 SQL Sever 有了初步的认知和一定的操作熟练度，为以后更好的学习数据库打下基础，同时我也复习了关系的三个完整性约束，获益匪浅。

