|  |
| --- |
| **《数据库系统SSD7》**  **实验报告**  专业班级 软件工程2306班  学 号 8209230614  姓 名 李兴  **实验成绩：**  **批阅教师：**  2025年 4 月 11日 |

# 实验1《数据库与表的基本操作》

实验学时： 2 实验地点： 信息楼508 实验日期： 2025/4/11

**一、实验目的**

1．熟练掌握一种DBMS的使用方法，完成数据库的创建、删除和连接；数据表的建立、删除；表结构的修改。

2. 加深对表的实体完整性、参照完整性和用户自定义完整性的理解。

**二、实验内容**

在以下实验中，使用学生-课程数据库，它描述了学生的基本信息、课程的基本信息及学生选修课程的信息。用SQL语句分别完成。

（1）创建学生-课程数据库，该数据库名为student\_xxxxxx，xxxxxx表示班级学号。2201班12号，数据库名为student\_220112，默认字符集设置为utf8。

（2）创建学生关系表S ：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 学号 | 姓名 | 性别 | 年龄 | 所在系 |
| Sno | Sname | Ssex | Sage | sdept |

（3）创建课程关系表C ：

|  |  |  |  |
| --- | --- | --- | --- |
| 课程号 | 课程名 | 先行课 | 学分 |
| Cno | Cname | Cpno | ccredit |

（4）创建学生-课程表SC ：

|  |  |  |
| --- | --- | --- |
| 学号 | 课程号 | 成绩 |
| Sno | Cno | grade |

（5）将以上创建表S、C、SC的SQL命令以 .SQL文件的形式保存在磁盘上。在表中加入至少4个元组，第一个为本人信息。

（6）在表S上增加“出生日期”与“身高”属性列。

（7）删除表S的“身高” 属性列。

（8）备份数据库，再还原。

**三、实验方法与实验步骤**

**实验方法**

1.使用MySQL作为数据库管理系统

2.使用Navicat Premium作为数据库管理工具

3.使用SQL语言完成数据库和表的创建、修改和备份操作

**实验步骤**

1.创建学生-课程数据库

2.创建学生关系表S

3.创建课程关系表C

4.创建学生-课程表SC

5.将创建表的SQL命令保存为.SQL文件，在表中插入至少4个元组，第一个为本人信息

6.在表S上增加“出生日期”与“身高”属性列

7.删除表S的“身高”属性列

8.备份数据库，再还原

**四、实验结果**

**源代码：**

//创建数据库

CREATE DATABASE student\_230614 CHARACTER SET utf8;

//创建学生关系表S

CREATE TABLE S (

Sno VARCHAR(10) PRIMARY KEY,

Sname VARCHAR(50) NOT NULL,

Ssex CHAR(1) ,

Sage INT,

sdept VARCHAR(50)

);

//创建课程关系表C

CREATE TABLE C (

Cno VARCHAR(10) PRIMARY KEY,

Cname VARCHAR(50) NOT NULL,

Cpno VARCHAR(10),

ccredit INT

);

//创建学生-课程表SC

CREATE TABLE SC (

Sno VARCHAR(10),

Cno VARCHAR(10),

grade FLOAT,

PRIMARY KEY (Sno, Cno),

FOREIGN KEY (Sno) REFERENCES S(Sno),

FOREIGN KEY (Cno) REFERENCES C(Cno)

);

//在表中加入至少4个元组，第一个为本人信息

INSERT INTO S (Sno, Sname, Ssex, Sage, sdept) VALUES

('230614', '李兴', 'M', 20, '计算机'),

('2306101', '小明', 'F', 22, '语文'),

('2306102', '小红', 'M', 21, '数学'),

('2306103', '小刚', 'F', 23, '英语');

INSERT INTO C (Cno, Cname, Cpno, ccredit) VALUES

('C001', '高等数学', NULL, 5),

('C002', '大学英语', NULL, 2),

('C003', '数据结构', 'C001', 3),

('C004', '操作系统', 'C003', 3);

INSERT INTO SC (Sno, Cno, grade) VALUES

('230614', 'C001', 100),

('2306101', 'C003', 55),

('2306102', 'C002', 80),

('2306103', 'C004', 95);

//在表S上增加“出生日期”与“身高”属性列

ALTER TABLE S

ADD COLUMN birthdate DATE,

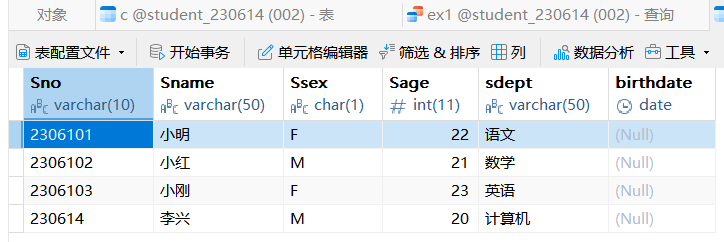
ADD COLUMN height FLOAT;

//删除表S的“身高” 属性列

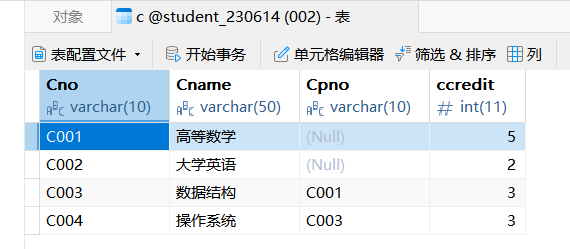
ALTER TABLE S

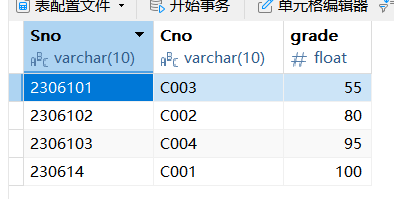
DROP COLUMN height;

**数据库截图**

学生关系表S：

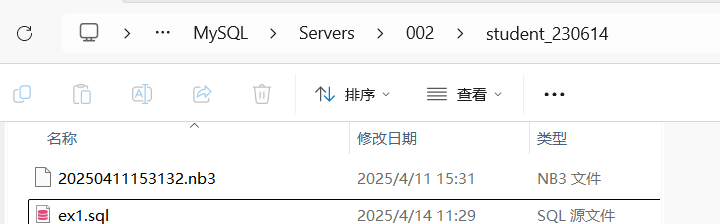
课程关系表C：

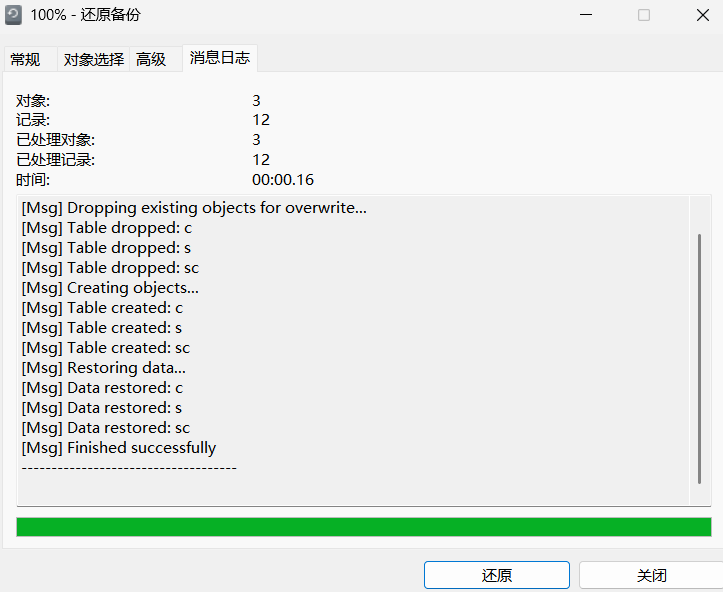
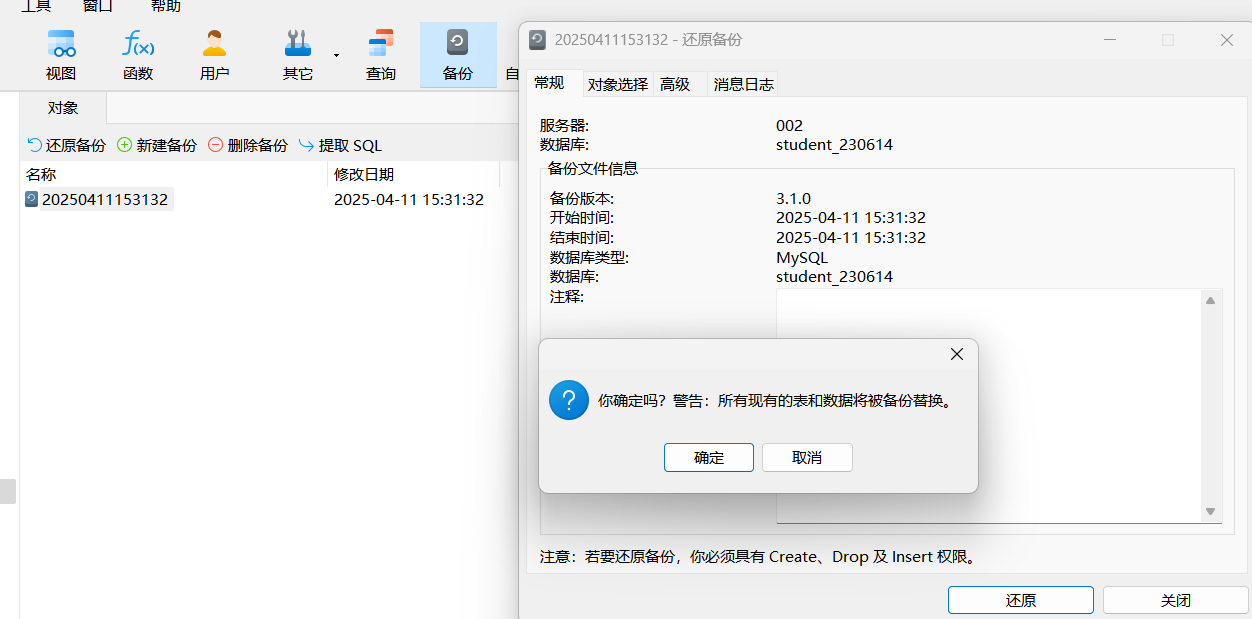


学生-课程表SC ：

**保存备份还原**

将创建表S、C、SC的SQL命令以 .SQL文件的形式保存在磁盘上。备份数据库，再还原。这两步操作可以使用Navicat Premium中的GUI实现。





**五、实验小结**

实验中遇到的问题：

问题：SQL语句写的不正确，比如在创建表时，不同的属性要用逗号隔开，而不是分号。

解决：先分析报错原因，再看ppt中的代码是怎么写的，对照着修改。

问题：初次创建表时，未注意主键和外键，导致了一些小问题

解决：再次仔细分析表的结构，找出主键和外键，参考ppt修改代码

问题：不会保存sql文件和数据库备份和还原

解决：放弃使用命令行，改为使用Navicat Premium自带的UI进行可视化操作，方便简单

通过本次实验，我初步学会了数据库管理系统的使用方法，掌握了数据库和数据表的创建、修改、删除等基本操作，进一步加深了对表的实体完整性、参照完整性和用户自定义完整性的理解。对于数据库与数据表的操作，我学会了如何使用SQL语句创建数据库和数据表，明确了在创建过程中需要设置的字符集、字段类型、主键等关键要素。例如，在创建学生-课程数据库时，正确设置了默认字符集为utf8，以确保数据的正确存储和显示。对于修改表结构，我掌握了如何向表中添加和删除属性列。通过在表S上增加“出生日期”与“身高”属性列，以及后续删除“身高”属性列的操作，熟悉了ALTER TABLE语句的使用方法。对于理解数据完整性，我通过创建学生、课程和学生-课程表，明确了各表之间的关系以及完整性约束的重要性。例如，学生表的学号（Sno）和课程表的课程号（Cno）作为主键，确保了数据的唯一性；学生-课程表SC中的学号（Sno）和课程号（Cno）作为外键，体现了参照完整性，保证了数据之间的正确关联。当然，我也会了进行数据库备份和恢复，对Navicat Premium的基本操作的掌握进一步加深。