

山西工程技术学院

实验报告

(2024- 2025 学年第一学期)

课程名称: 数据库系统原理

专业班级:

学 号:

学生姓名:

任课教师:

2024 年 12 月

实验报告 4

实验名称	数据库的视图定义及维护			指导教师	李素萍
实验类型	综合型	实验学时	4	实验时间	
一、实验目的与要求 <ol style="list-style-type: none"> 1、使学生掌握视图的定义与维护操作 2、加深对视图在关系数据库中的作用的理解决。 					
二、实验环境 <p>操作系统：win-7</p> <p>编程环境：Microsoft SQL Server 2008</p>					
三、实验内容和步骤 <ol style="list-style-type: none"> 1、写出创建满足下述要求的视图的 SQL 语句，并执行所写代码。 <ol style="list-style-type: none"> (1) 包含学生的学号、姓名、所在系、年龄，课程号、课程名、课程学分的视图。 <pre>create view v1 as select Student.Sno,Sname,Sdept,Sage,SC.Cno,Cname,Credit from Student,SC,Course where Student.Sno=SC.Sno and Course.Cno=SC.Cno</pre> (2) 学生的学号、姓名、选修的课程名和考试成绩的视图。 <pre>create view v2 as select Student.Sno,Sname,Cname,Grade from Student,SC,Course where Student.Sno=SC.Sno and SC.Cno=Course.Cno</pre> (3) 统计每个学生的选课门数，要求列出学生学号和选课门数的视图。 <pre>create view v3 as select Sno,COUNT(Cno) CourseNum from SC group by Sno</pre> (4) 统计每个学生的选修课总学分，要求列出学生学号和总学分的视图。 <pre>create view v4 as select Sno,SUM(Credit) SumCredit from SC,Course where SC.Cno=Course.Cno and Grade>=60 group by Sno</pre> 2、利用第 1 题建立的视图，完成如下查询。 <ol style="list-style-type: none"> (1) 查询考试成绩大于等于 90 分的学生的姓名、课程名和成绩。 <pre>create view v4 as select Sno,SUM(Credit) SumCredit</pre> 					

```
from SC, Course
where SC.Cno=Course.Cno and Grade>=90 group by Sno
```

(2) 查询选课门数超过 3 门的学生的学号和选课门数。

```
select Sno, CourseNum
from v3
where CourseNum>3
```

(3) 查询计算机系选课门数超过 3 门的学生的姓名和选课门数。

```
select Sname, CourseNum
from v3, Student
where v3.Sno=Student.Sno and CourseNum>3 and Sdept='计算机系'
```

(4) 查询修课总学分超过 10 分的学生的学号、姓名、所在系和修课总学分。
查询年龄大于等于 20 岁的学生中，修课总学分超过 10 分的学生的姓名、年龄、所在系和修课总学分。

```
select Sname, Sage, Sdept, SumCredit from v4, Student where
v4.Sno=Student.Sno and SumCredit>10 and Sage>=20
```

3. 使用 WITH CHECK OPTION，在第 1 题建立的视图（1）的基础上，再创建一个“计算机系”同学的学号、姓名、所在系、年龄，课程号、课程名、课程学分的视图。写出 SQL 语句。

```
create view v5 as
select * from v1 where Sdept='计算机系'
with check option;
```

增加一条记录

(0841101, 张新, 计算机系, 20, C001, 高等数学, 4)

修改这条记录为

(0841101, 张新, 物理系, 20, C001, 高等数学, 4)

以上两个操作是否能够实现？为什么？可以实现的，写出 SQL 语句。

1. **修改系别：**“计算机”这个系别已经存在，并且只是想修改系别为“物理系”，并且修改的对象 Sname 不变，所以更新这条记录只涉及到修

改 Student 表的系别属性。

2. 修改课程和学分：对于 C001 课程和学分（4），这部分是 SC 表中的记录。该表存储了学生选课信息，因此如果要修改课程，应该针对选课记录进行更新。

可以通过 **两个更新操作** 来分别修改学生的系别（在 Student 表中），以及更新课程信息（在 SC 表中）。如果需要在 Student 表和 SC 表之间进行更新，以下是相应的 SQL 语句。

```
UPDATE Student
SET Sdept = '物理系'
WHERE Sno = '0841101' AND Sdept = '计算机系';
```

```
UPDATE SC
SET Grade = 4 -- 如果需要修改成绩
WHERE Sno = '0841101' AND Cno = 'C001';
```

四、实验小结和思考

本实验中我学会了使用和创建视图，视图为数据库提供了灵活性、可维护性和安全性，通过简化查询、控制数据访问和提高代码重用性，帮助开发人员和数据库管理员更高效地工作。

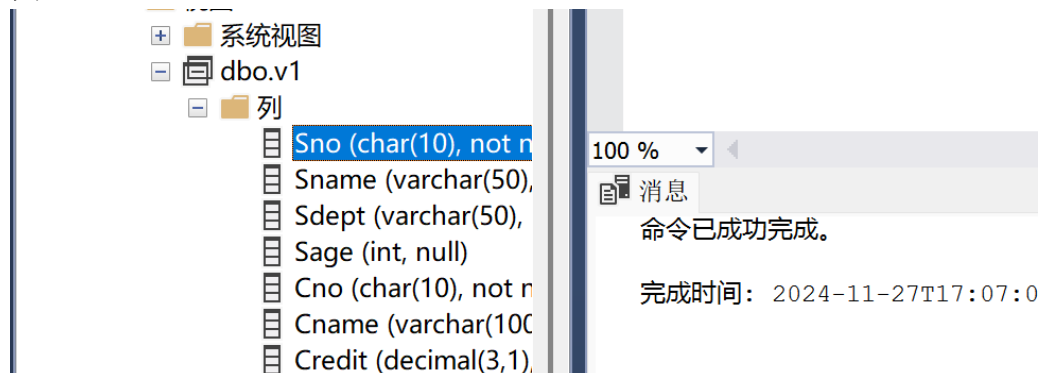
同时也学会和体验了外键的作用和重要性：保护数据一致性和完整性，以及防止插入无效数据

实验成绩		批阅日期		批阅人	
------	--	------	--	-----	--

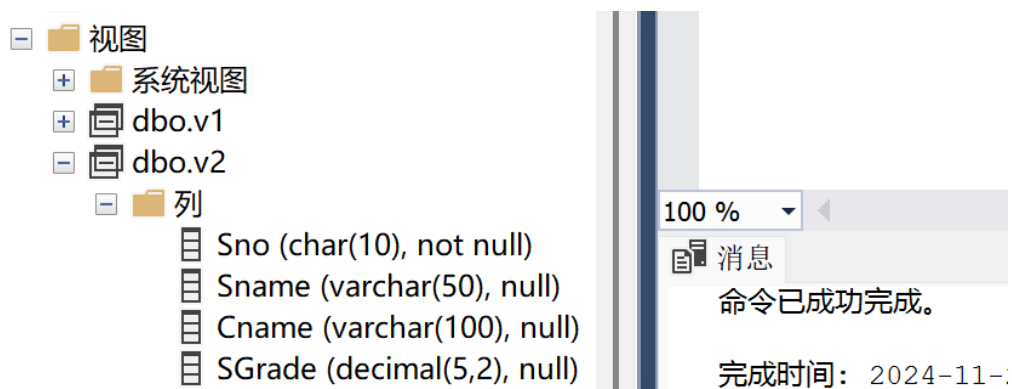
附件（实验结果）

1、写出创建满足下述要求的视图的 SQL 语句，并执行所写代码。

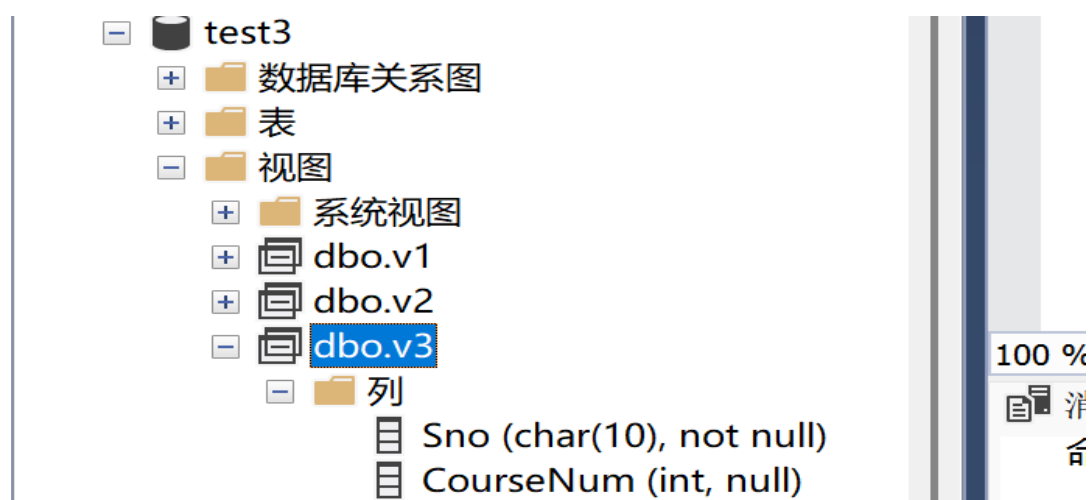
(1) 包含学生的学号、姓名、所在系、年龄，课程号、课程名、课程学分的视图。



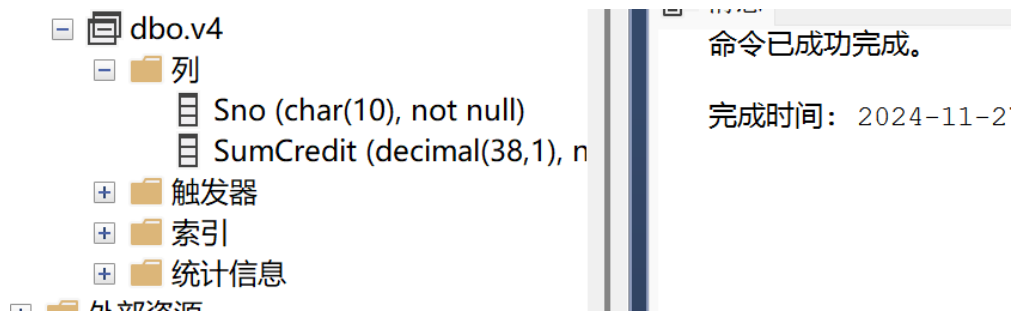
(2) 学生的学号、姓名、选修的课程名和考试成绩的视图。



(3) 统计每个学生的选课门数，要求列出学生学号和选课门数的视图。



(4) 统计每个学生的修课总学分，要求列出学生学号和总学分的视图。



2、利用第 1 题建立的视图，完成如下查询。

(1) 查询考试成绩大于等于 90 分的学生的姓名、课程名和成绩。

	Sname	Cname	Sgrade
1	李四	数据库系统	90.50
2	李四	计算机网络	90.50
3	钱七	数据库系统	92.00
4	钱七	经济学原理	92.00

(2) 查询选课门数超过 3 门的学生的学号和选课门数。

结果		消息	
	Sno	CourseNum	

显然，没有人选择课程数量大于 3 门，数值超出范围了，故我们可以修改选择课程数量 ≥ 2 门的情况

结果		消息	
	Sno	CourseNum	
1	S001	2	
2	S002	2	
3	S003	2	
4	S004	2	
5	S005	2	

(3) 查询计算机系选课门数超过 3 门的学生的姓名和选课门数。

结果 消息	
	Sname CourseNum

查询出信息依然为空的原因如上，更改数据范围即可得出结论：

结果 消息	
	Sname CourseNum
1	张三 2

(4) 查询修课总学分超过 10 分的学生的学号、姓名、所在系和修课总学分。
查询年龄大于等于 20 岁的学生中，修课总学分超过 10 分的学生的姓名、年龄、所在系和修课总学分。

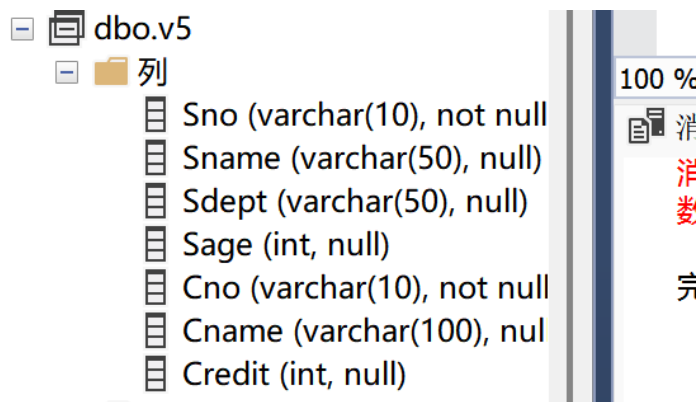
结果 消息	
	Sno Sname Sdept SumCredit

由上可知：10 分依然超过范围，故修改为总学分大于 6.0

100 %	
结果 消息	
	Sno Sname Sdept SumCredit
1	S001 张三 计算机科学与技术 7.0
2	S002 李四 电子信息工程 6.5
3	S004 赵六 法学 7.0

3. 使用 WITH CHECK OPTION, 在第 1 题建立的视图 (1) 的基础上, 再创建一个 “计算机系” 同学的学号、姓名、所在系、年龄, 课程号、课程名、课程学分的视图。写出 SQL 语句。

```
create view v5 as
select * from v1 where Sdept='计算机系'
with check option;
```



增加一条记录
 (0841101, 张新, 计算机系, 20, C001, 高等数学, 4)
 修改这条记录为
 (0841101, 张新, 物理系, 20, C001, 高等数学, 4)
 以上两个操作是否能够实现? 为什么? 可以实现的, 写出 SQL 语句。

	Sno	Sname	Sdept	Sage	Grade
1	0841101	张新	物理系	20	90.00
2	0841102	李华	计算机系	22	85.00
3	0841103	王强	计算机系	21	88.00
4	0841104	赵丽	电子系	23	92.00
5	0841105	钱超	计算机系	20	86.00
6	S001	张三	计算机科学与技术	20	85.50
7	S002	李四	电子信息工程	21	92.00
8	S003	王五	软件工程	22	78.50
9	S004	赵六	数学与应用数学	19	88.00
10	S005	孙七	物理学	23	90.50
11	S006	周八	化学	21	82.75
12	S007	吴九	生物科学	22	74.20
13	S008	郑十	工商管理	20	91.00
14	S009	钱十一	会计学	23	79.80
15	S010	孙十二	法学	24	95.60

UPDATE 语句用于更新表中的记录。你可以指定一个或多个字段进行修改。只要你在 WHERE 子句中明确指定了唯一标识 (在这个例子中是学号 Sno), SQL 就能找到需要修改的行, 并且修改指定的字段 (Sdept)。

由于该操作只修改 Sdept 字段 (系别字段), 其他字段 (如学号、姓名、年龄、课程等) 不会受到影响, 仍然保持原值。因此, 这两个操作 (从 计算机系 到 物理系) 是可以成功完成的。

消息 547, 级别 16, 状态 0, 第 2 行

UPDATE 语句与 REFERENCE 约束"FK__SC__Sno__30F848ED"冲突。该冲突发生于数据库"test4", 表"dbo.SC", column 'Sno'。
语句已终止。

消息 547, 级别 16, 状态 0, 第 7 行

UPDATE 语句与 REFERENCE 约束"FK__SC__Sno__30F848ED"冲突。该冲突发生于数据库"test4", 表"dbo.SC", column 'Sno'。
语句已终止。

消息 547, 级别 16, 状态 0, 第 12 行

UPDATE 语句与 REFERENCE 约束"FK__SC__Sno__30F848ED"冲突。该冲突发生于数据库"test4", 表"dbo.SC", column 'Sno'。
语句已终止。

消息 547, 级别 16, 状态 0, 第 17 行

UPDATE 语句与 REFERENCE 约束"FK__SC__Sno__30F848ED"冲突。该冲突发生于数据库"test4", 表"dbo.SC", column 'Sno'。
语句已终止。

无法直接在 SC 表中插入数据！！

原因：外键约束通常会限制修改该列的值，以保持数据库中的引用完整性。

外键约束的作用：

保持数据的一致性和完整性：外键约束确保子表中的数据值必须在父表中存在，即外键列的值必须引用父表中的有效主键或唯一键值。这就避免了数据库中的孤立数据（例如，子表中的某些记录引用了父表中不存在的记录）。

防止插入无效数据：当插入新记录时，如果外键约束被激活，数据库会检查外键值是否在父表中存在。如果不存在，数据库会拒绝插入操作。