

Lecture 9, Fall 2017/2018

数据库系统实验

Yubao Liu (刘玉葆)

School of Data and Computer Science

Sun Yat-sen University

- 本节课提纲

- 实验目的
- 实验内容
- 实验示例
- 练习

- 实验目的

学习实体完整性的建立，以及实践违反实体完整性的结果。

- ## 实验内容

(1)在数据库 school中建立表 Stu_Union,进行主键约束,在没有违反实体完整性的前提下插入并更新一条记录。

(2)演示违反实体完整性的插入操作。

(3)演示违反实体完整性的更新操作。

(4)演示事务的处理,包括事务的建立、处理,以及出错时的事务回退。

(5)通过建立 Scholarship表,插入数据,演示当与现有的数据环境不符时,无法建立实体完整性和参照完整性。

- 实验示例

以 school数据库为例，在该数据库中存在4张表格，分别为

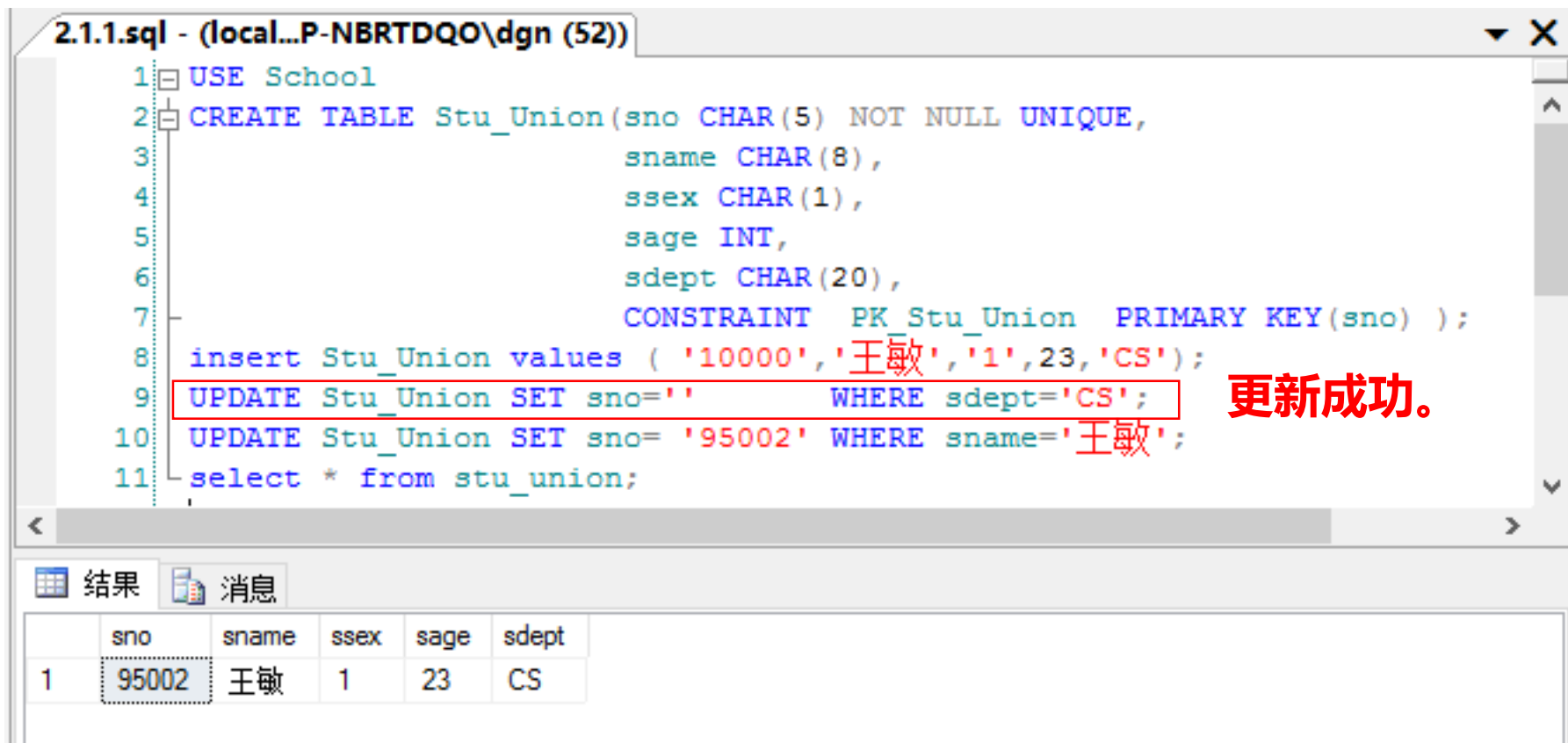
- students (sid, sname , email , grade)
- teachers (tid, tname , email , salary)
- courses (cid, cname , hour)
- choices (no, sid , tid , cid , score)

在数据库中，存在这样的关系，学生可以选择课程。一个课程对应一个教师。

在CHOICES表中保存学生的选课记录。

实验示例

1.在数据库 school中建立表 Stu_Union,进行主键约束,在没有违反实体完整性的前提下插入并更新一条记录。



```
2.1.1.sql - (local...P-NBRTDQO\dgn (52))
1 USE School
2 CREATE TABLE Stu_Union(sno CHAR(5) NOT NULL UNIQUE,
3                           sname CHAR(8),
4                           ssex CHAR(1),
5                           sage INT,
6                           sdept CHAR(20),
7                           CONSTRAINT PK_Stu_Union PRIMARY KEY(sno) );
8 insert Stu_Union values ( '10000', '王敏', '1', 23, 'CS');
9 UPDATE Stu_Union SET sno='' WHERE sdept='CS';
10 UPDATE Stu_Union SET sno= '95002' WHERE sname='王敏';
11 select * from stu_union;
```

更新成功。

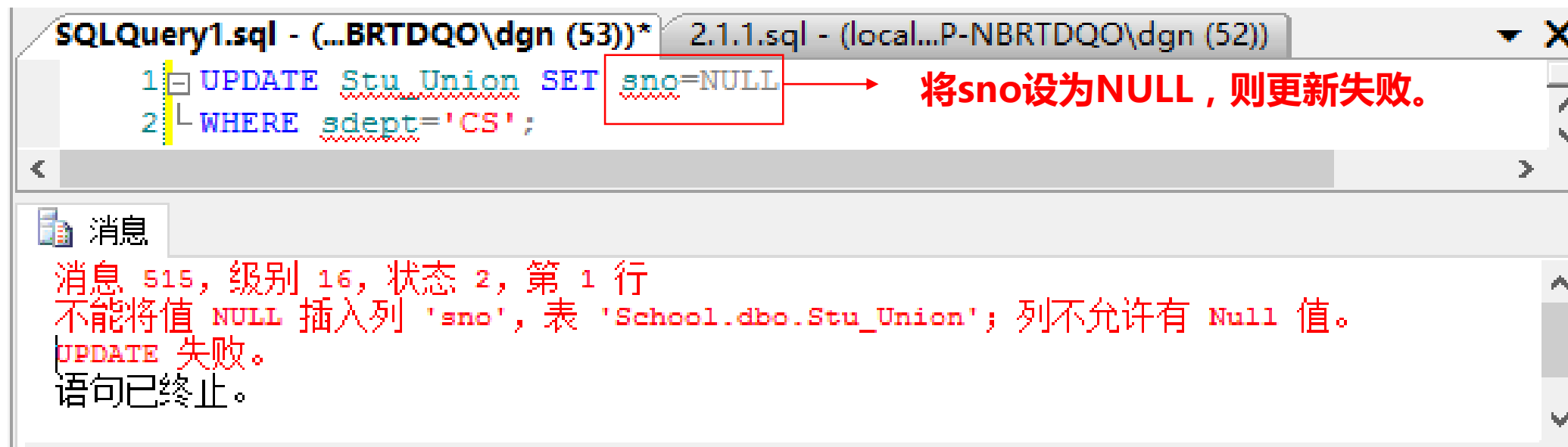
	sno	sname	ssex	sage	sdept
1	95002	王敏	1	23	CS

思考：为什么把sno设置为'' 没有违反NOT NULL约束呢？

• 实验示例

因为： ' ' （空格）不等于NULL。NULL不是0，也不是空格，而是一个“不知道”或“不确定”的数据值。

验证：将sno设为NULL，则违反主体完整性，更新失败。



The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows a query window with the following SQL statement:

```
1 UPDATE Stu_Union SET sno=NULL
2 WHERE sdept='CS';
```

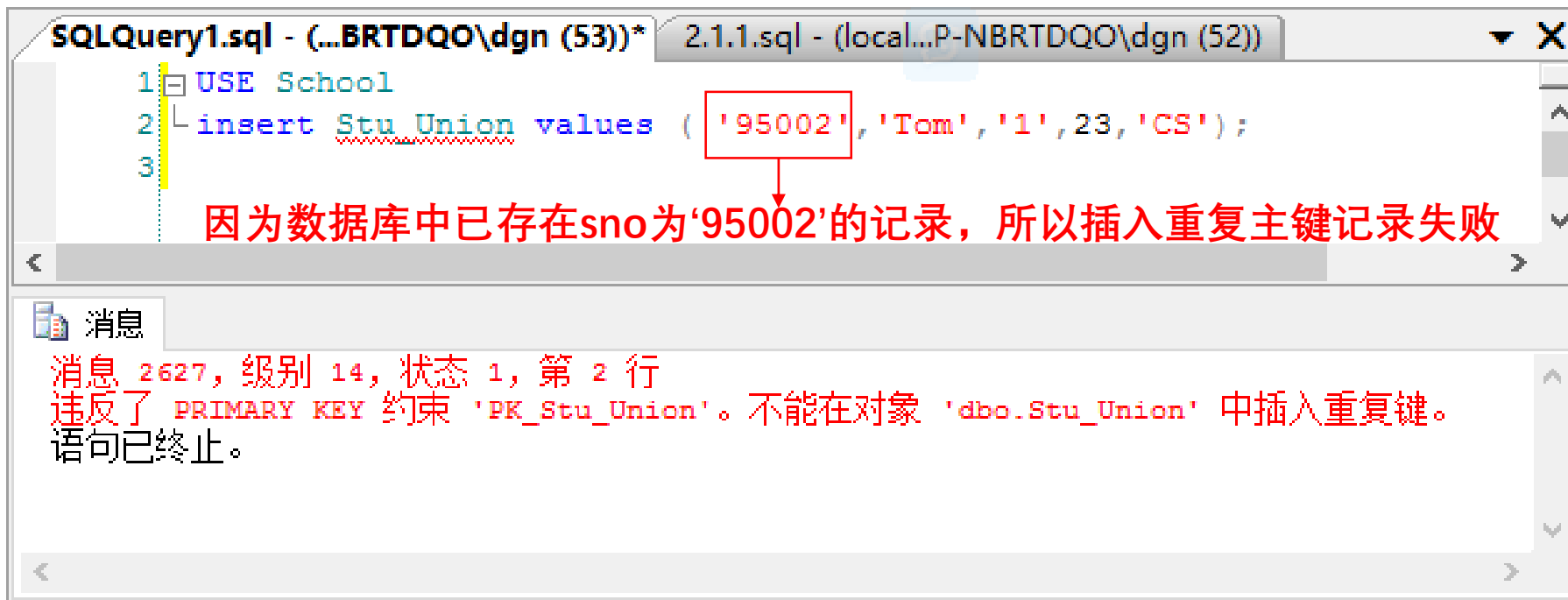
A red box highlights the `sno=NULL` assignment, with a red arrow pointing to the text **将sno设为NULL，则更新失败。** (Setting sno to NULL, then update fails).

The bottom pane, titled "消息" (Messages), displays the following error message in red text:

```
消息 515，级别 16，状态 2，第 1 行
不能将值 NULL 插入列 'sno'，表 'School.dbo.Stu_Union'；列不允许有 Null 值。
UPDATE 失败。
语句已终止。
```

• 实验示例

2.演示违反实体完整性的插入操作。



违反了主键的唯一性属性，将破坏实体完整性，所以插入失败。

• 实验示例

3.演示违反实体完整性的更新操作。

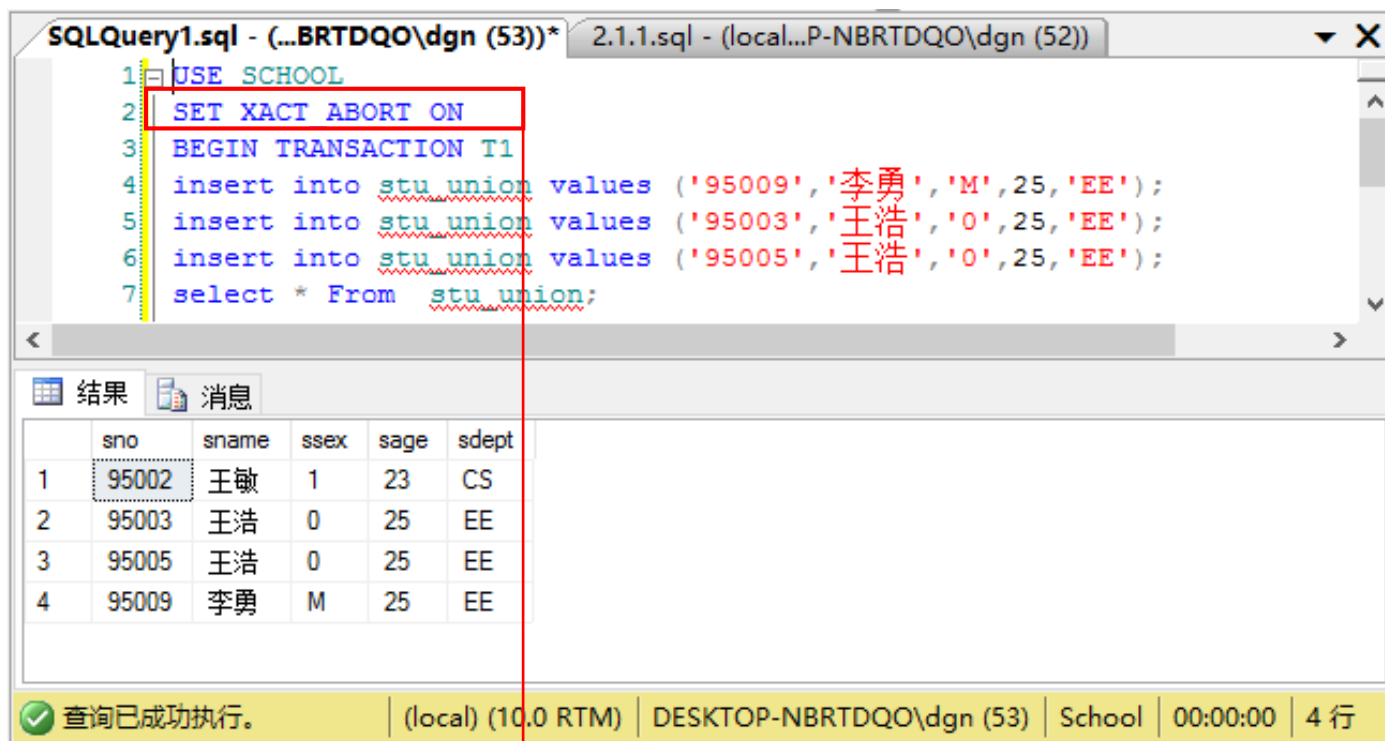


违反主键的NOT NULL属性，将破坏实体完整性，所以更新失败。

实验示例

4.演示事务的处理,包括事务的建立、处理,以及出错时的事务回退。

先建立事务T1：



The screenshot shows a SQL query window with two tabs. The active tab is 'SQLQuery1.sql - (...BRTDQO\dgn (53))'. The query text is as follows:

```
1 USE SCHOOL
2 SET XACT ABORT ON
3 BEGIN TRANSACTION T1
4 insert into stu_union values ('95009', '李勇', 'M', 25, 'EE');
5 insert into stu_union values ('95003', '王浩', '0', 25, 'EE');
6 insert into stu_union values ('95005', '王浩', '0', 25, 'EE');
7 select * From stu_union;
```

The second tab is '2.1.1.sql - (local...P-NBRTDQO\dgn (52))'. Below the query text, there are two tabs: '结果' (Results) and '消息' (Messages). The '结果' tab is active, displaying a table with the following data:

	sno	sname	ssex	sage	sdept
1	95002	王敏	1	23	CS
2	95003	王浩	0	25	EE
3	95005	王浩	0	25	EE
4	95009	李勇	M	25	EE

At the bottom of the window, a status bar shows a green checkmark icon and the text '查询已成功执行。' (Query executed successfully). To the right of this text, the status bar displays '(local) (10.0 RTM) | DESKTOP-NBRTDQO\dgn (53) | School | 00:00:00 | 4 行'.

注：此句表明当SQL语句运行错误时，整个事物将终止并回滚。若为OFF，则只回滚产生错误的SQL语句。

实验示例

然后建立事务T2：

The screenshot displays a SQL query window with the following code:

```
1 USE SCHOOL
2 SET XACT_ABORT ON
3 BEGIN TRANSACTION T2
4 insert into stu_union values ('95007','李明','M',25,'EE');
5 select * From stu_union;
6 insert into stu_union values ('95009','李进','F',22,'CS');
7 COMMIT TRANSACTION T2
```

Red boxes highlight the first and second insert statements. A red arrow points from the second insert statement to the error message below. The error message states:

表中已存在主键为'95009'的记录，
而插入主键重复键，违反了实体完整性，插入数据失败，事务回滚。

Below the error message, the status bar indicates: (1 行受影响) (5 行受影响) 消息 2627, 级别 14, 状态 1, 第 6 行 违反了 PRIMARY KEY 约束 'PK_Stu_Union'。不能在对象 'dbo.Stu_Union' 中插入重复键。

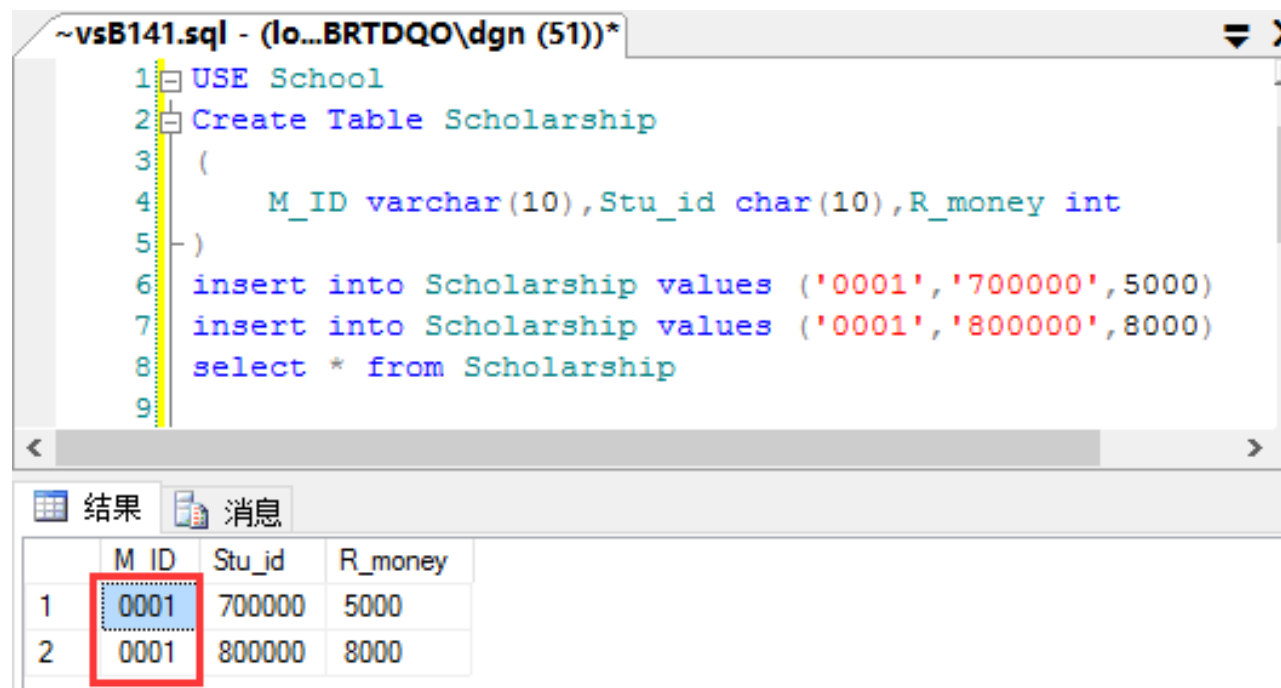
The status bar at the bottom shows: 查询已完成，但有错误。 (local) (10.0 RTM) DESKTOP-NBRTDQO\dgn (53) School 00:00:00 5 行

虽然此句没有错误，但由于作为一个事务，T2中操作要么一起成功，要么一起失败，所以当T2失败时，此句也无法执行成功。整个事务回滚到初始状态。同学们可自行查询表stu_union验证。

• 实验示例

5.通过建立 Scholarship表,插入数据,演示当与现有的数据环境不等时,无法建立实体完整性以及参照完整性。

首先建立表scholarship,并插入两条M_ID相同的数据。

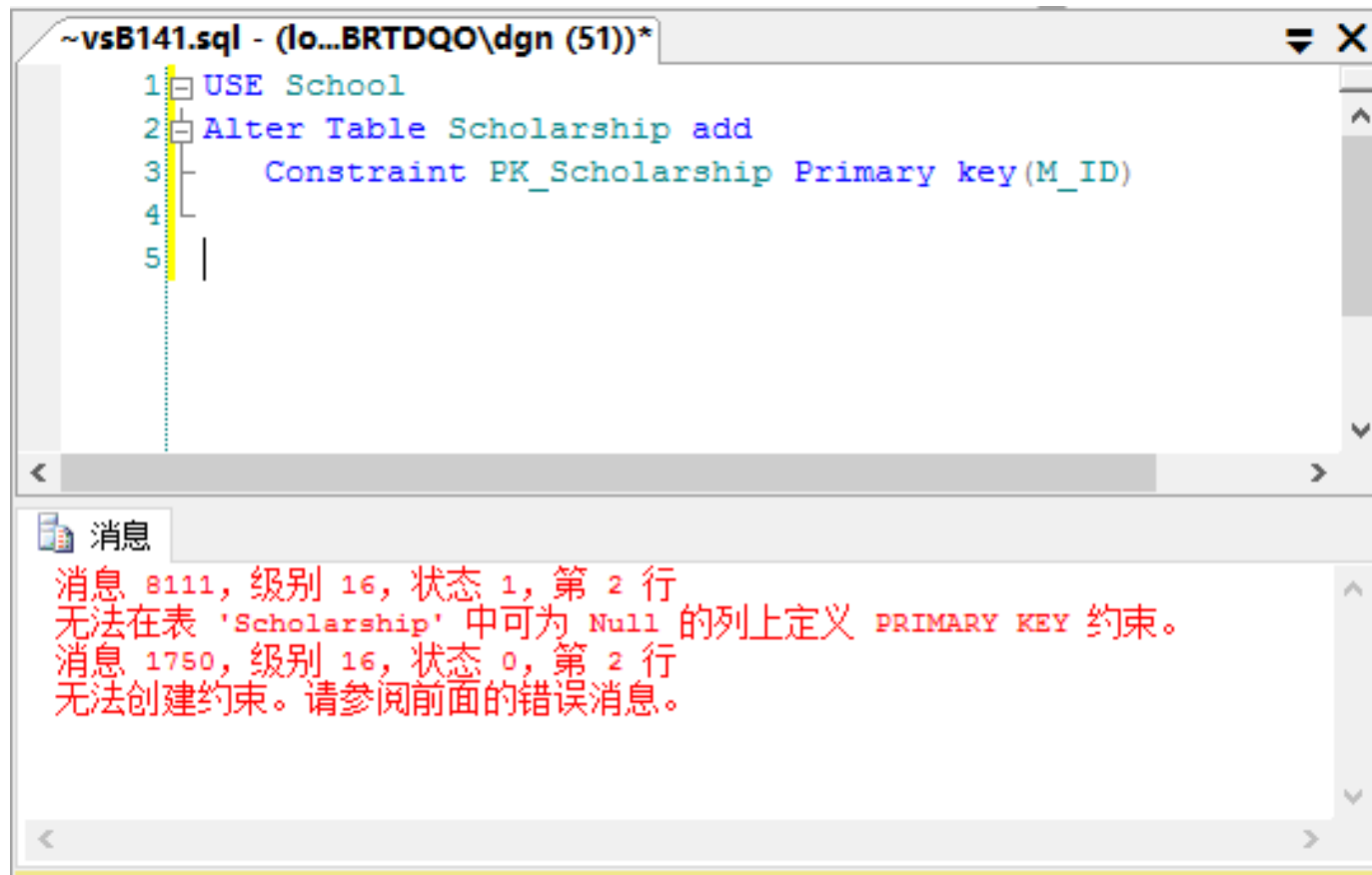


```
~vsB141.sql - (lo...BRTDQO\dgn (51))*
1 USE School
2 Create Table Scholarship
3 (
4     M_ID varchar(10),Stu_id char(10),R_money int
5 )
6 insert into Scholarship values ('0001','700000',5000)
7 insert into Scholarship values ('0001','800000',8000)
8 select * from Scholarship
9
```

	M_ID	Stu_id	R_money
1	0001	700000	5000
2	0001	800000	8000

- 实验示例

然后尝试将M_ID设为主键，操作失败。因为数据列M_ID不满足实体完整性。



The screenshot shows a SQL IDE window titled '~vsB141.sql - (lo...BRTDQO\dgn (51))*'. The SQL editor contains the following code:

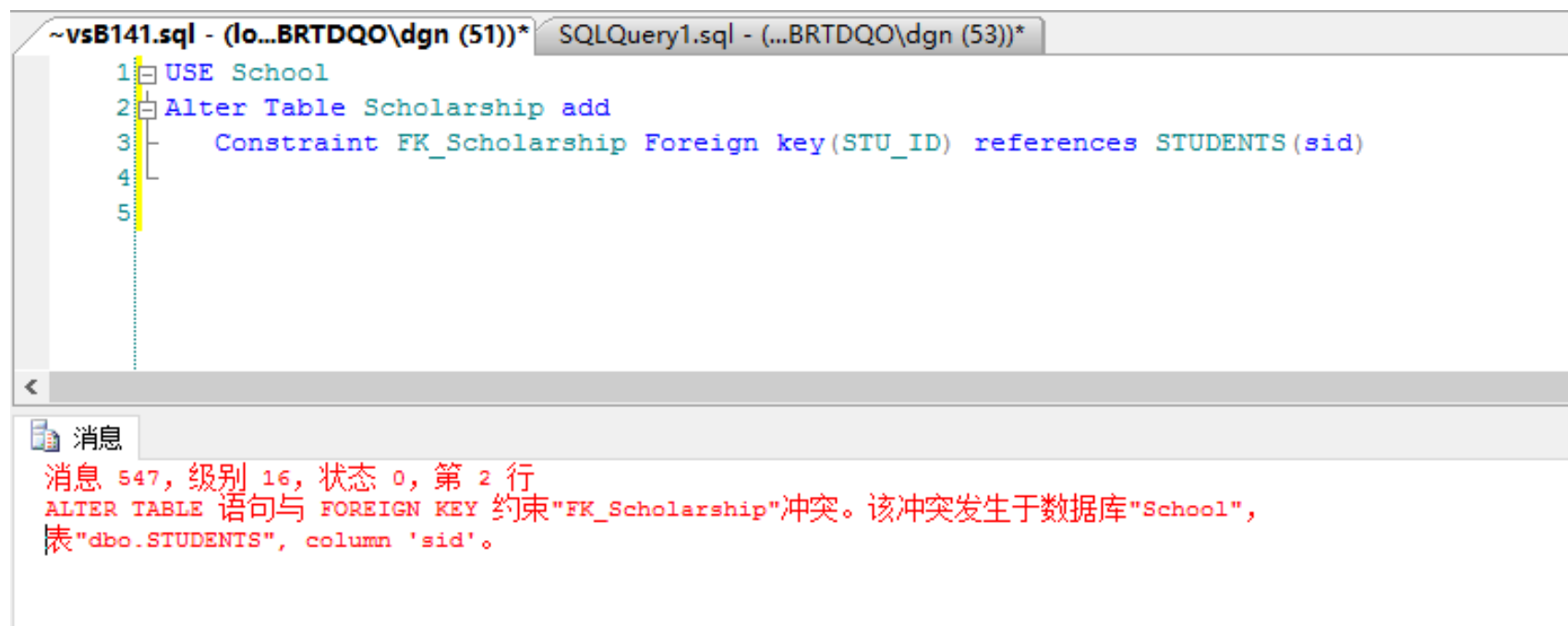
```
1 USE School
2 Alter Table Scholarship add
3     Constraint PK_Scholarship Primary key (M_ID)
4
5
```

Below the editor is a '消息' (Messages) pane. It displays two error messages in red text:

- 消息 8111, 级别 16, 状态 1, 第 2 行
无法在表 'Scholarship' 中可为 Null 的列上定义 PRIMARY KEY 约束。
- 消息 1750, 级别 16, 状态 0, 第 2 行
无法创建约束。请参阅前面的错误消息。

• 实验示例

6.尝试在scholarship表中增加students(id)的外键。但操作失败，因为scholarship中的数据不满足students表中sid的对应性。所以创建参照完整性失败。



The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows a script window with the following SQL code:

```
1 USE School
2 Alter Table Scholarship add
3     Constraint FK_Scholarship Foreign key(STU_ID) references STUDENTS(sid)
4
5
```

The bottom pane shows a message window with the following error text:

消息 547, 级别 16, 状态 0, 第 2 行
ALTER TABLE 语句与 FOREIGN KEY 约束"FK_Scholarship"冲突。该冲突发生于数据库"School",
表"dbo.STUDENTS", column 'sid'。

- 练习

- (1) 在school数据库中建立一张新表class , 包括class_id(varchar(4)), name(varchar(10)), department(varchar(20))三个列 , 并约束class_id为主键。
- (2) 创建事务T3 , 在事务中插入一个元组 ('00001' , ' 01CSC' , ' CS') , 并在T3中嵌套创建事务T4 , T4也插入和T3一样的元组 , 编写代码测试 , 查看结果。