

# 中山大学计算机学院数据库实验本科生实验报告

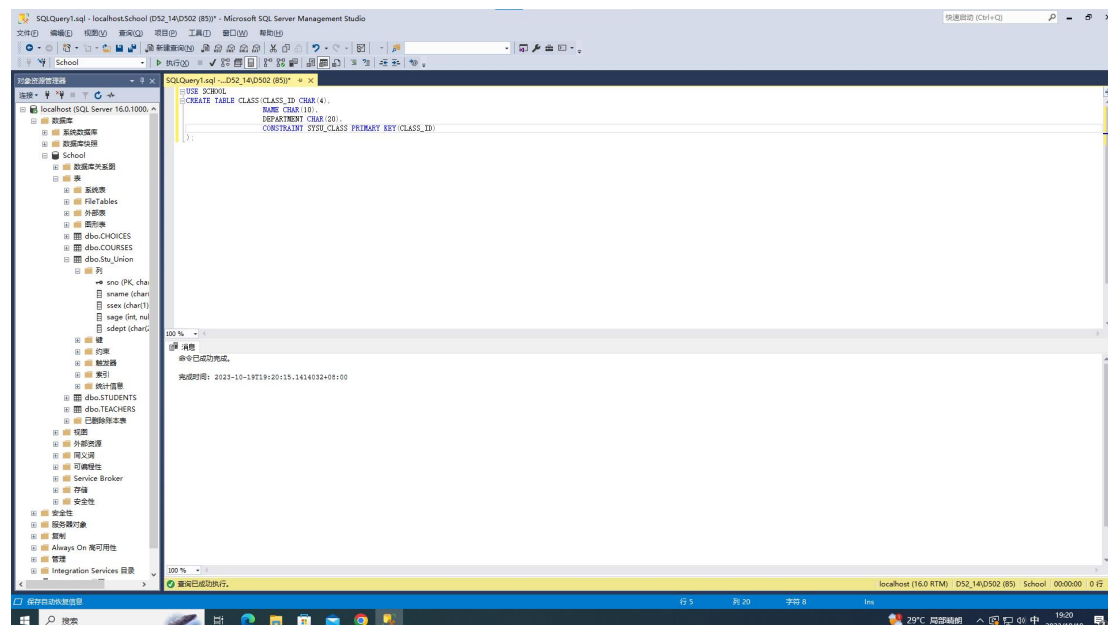
课程名称:数据库实验 教学班级:2 班 学号:21307174 姓名:刘俊杰

## 一、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。

## 二、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。



创建表 CLASS 成功

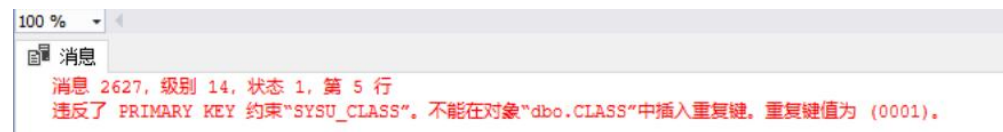
2. 创建事务 T3, 在事务中插入一个元组( '0001', '01CSC', 'CS' ), 并在 T3 中嵌套创建事务 T4, T4 也插入和 T3 一样的

元组，编写代码测试，查看结果。

### ①使用不回滚的事务

```
USE SCHOOL
SET XACT_ABORT OFF
BEGIN TRANSACTION T3
INSERT INTO CLASS VALUES('0001','01CSC','CS');
BEGIN TRANSACTION T4
INSERT INTO CLASS VALUES('0001','01CSC','CS');
```

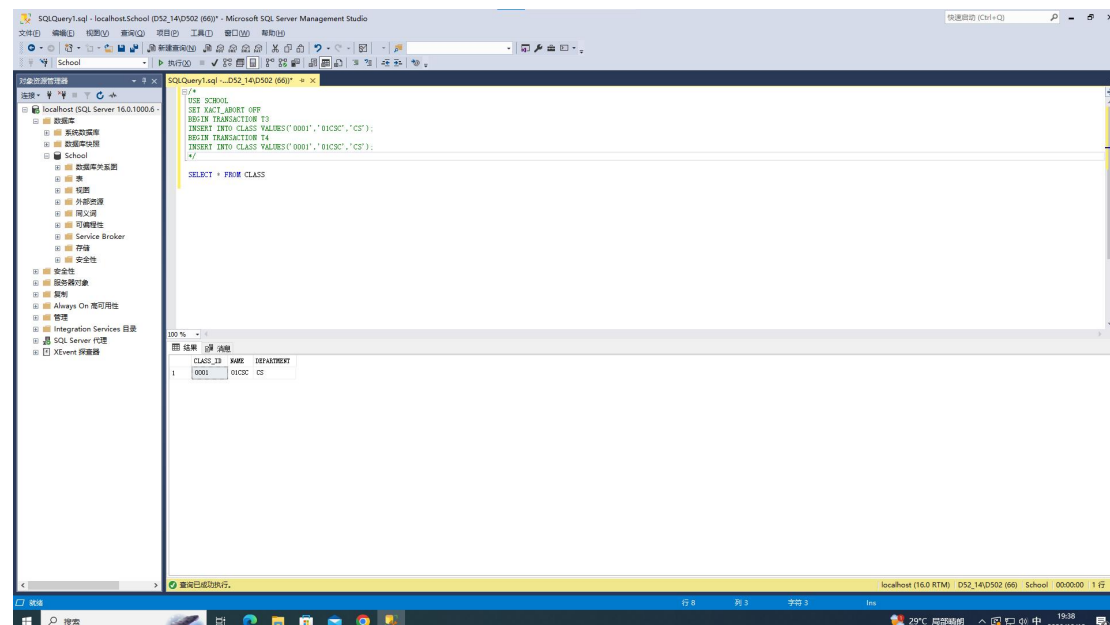
运行后显示报错



经过查询，发现（‘0001’，’01CSC’，’CS’）插入成功，这是因为使用的是不回滚的事务，T3 中的插入成功，而 T4 的插入失败不会影响 T3 的插入回滚，所以 T3 插入的结果依然在表中。

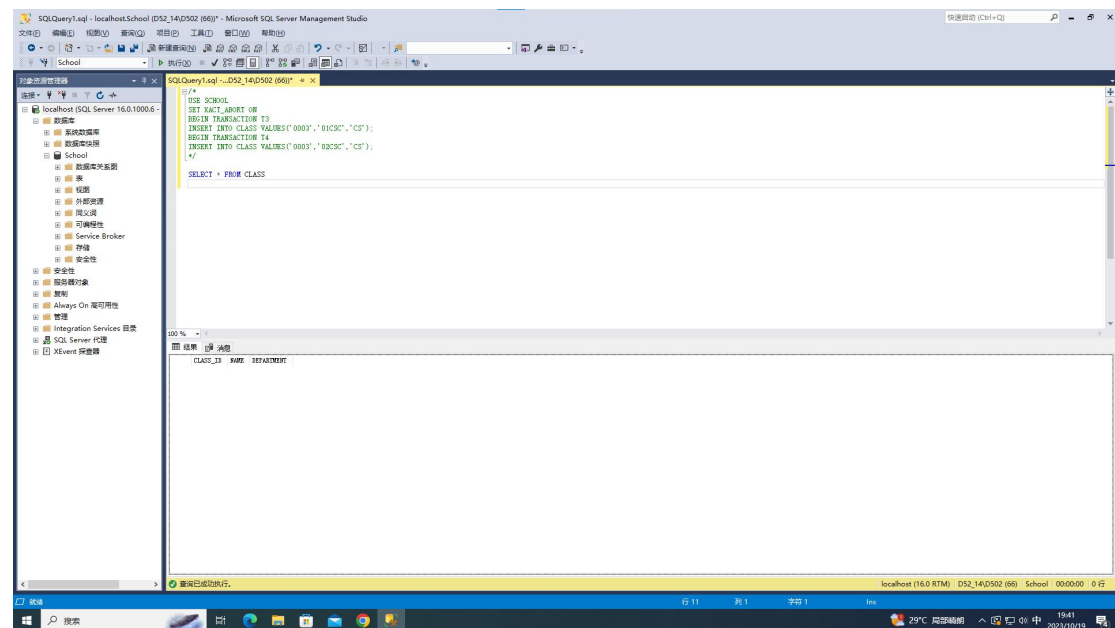
### ②使用会回滚的事务

同样会显示报错



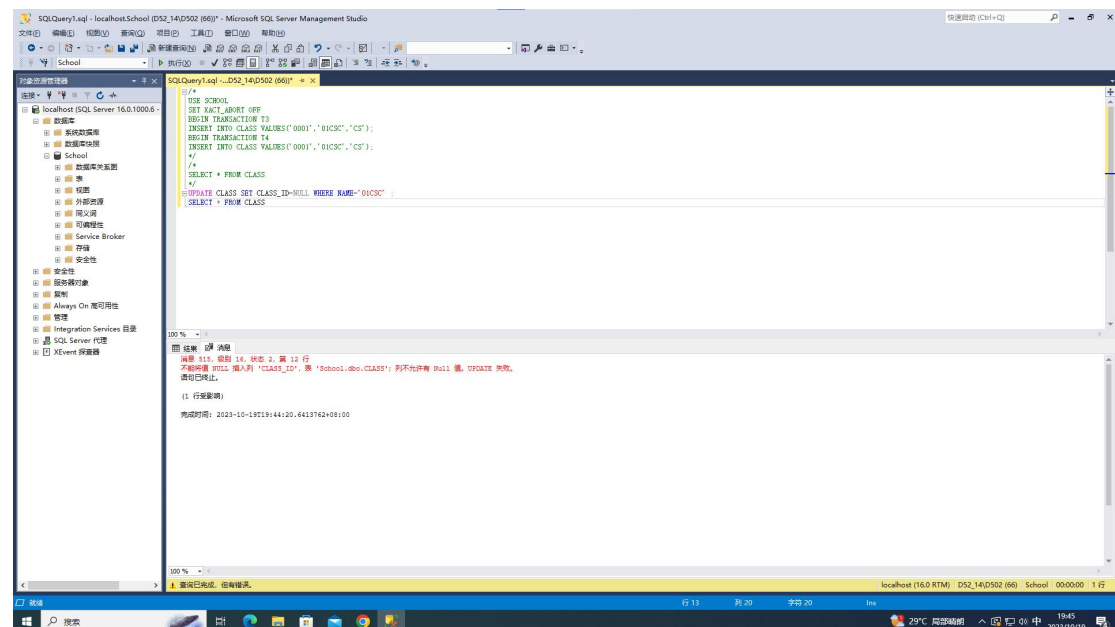
但查询后发现表是空的，这是因为使用的是会回滚的事务，T3 中的插入成功，但 T4 的插入失败会影响 T3 的插入回滚，所以 T3 插入的

结果回滚导致没有插入。



(在后续的实验中，使用练习 2 使用不回滚的事物的 CLASS)

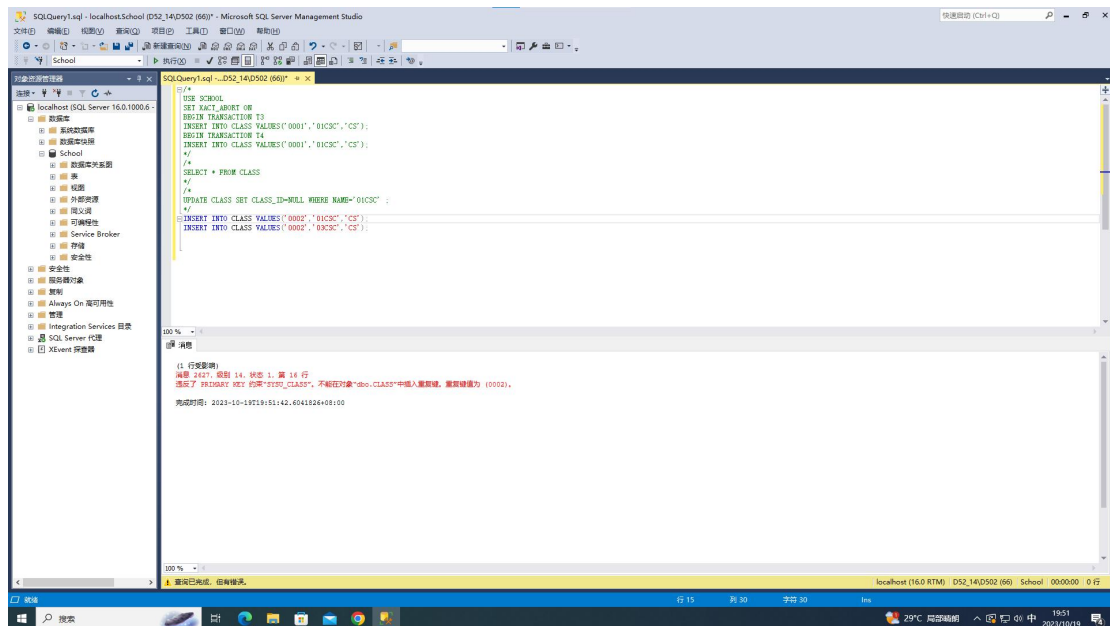
3. 在表class中,尝试设置name='01CSC'的记录class\_id为 NULL，查看结果.



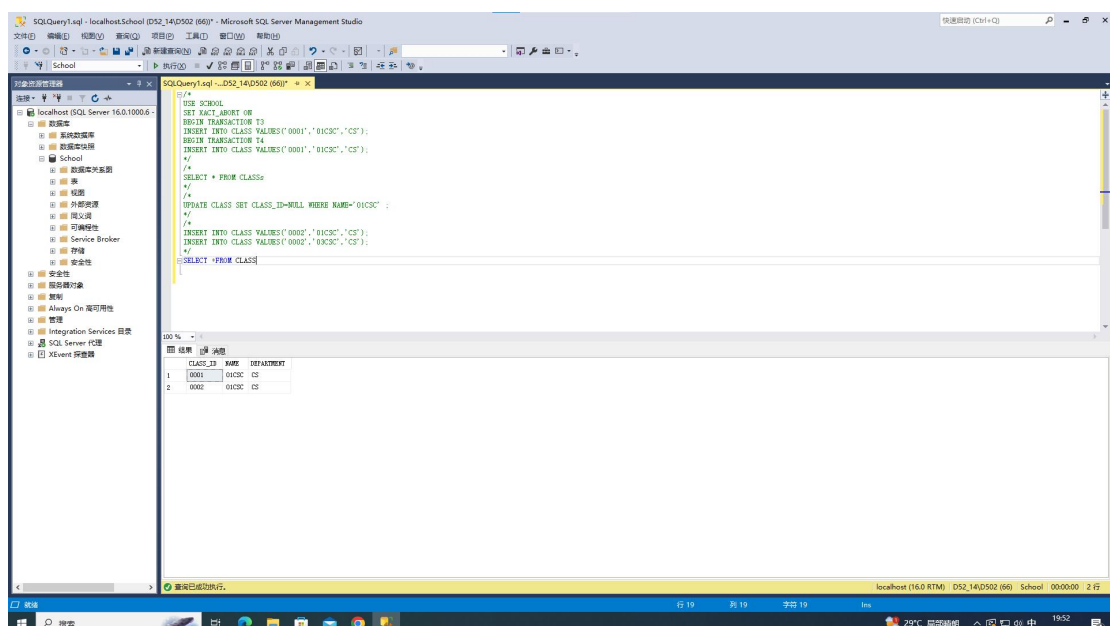
报错显示 CLASS\_ID 不能设置为 NULL，这是因为尽管我们没有在创建表时给 CLASS\_ID 加上不能为 NULL 的约束，但我们设置了 CLASS\_ID

为主键，而主键的值不能为 NULL，所以 CLASS\_ID 不能设置为 NULL。

4. 在表 class 中，不创建事务，插入两个元组（'0002'，'01CSC'，'CS'），（'0002'，'03CSC'，'CS'），然后查看表中有几条记录，为什么？



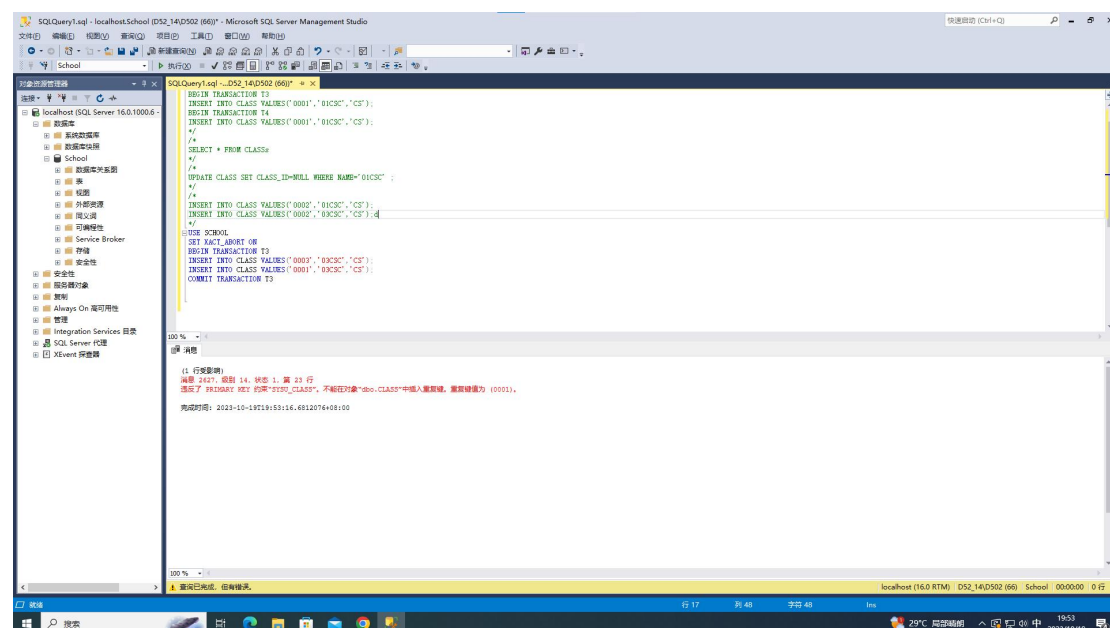
运行插入指令

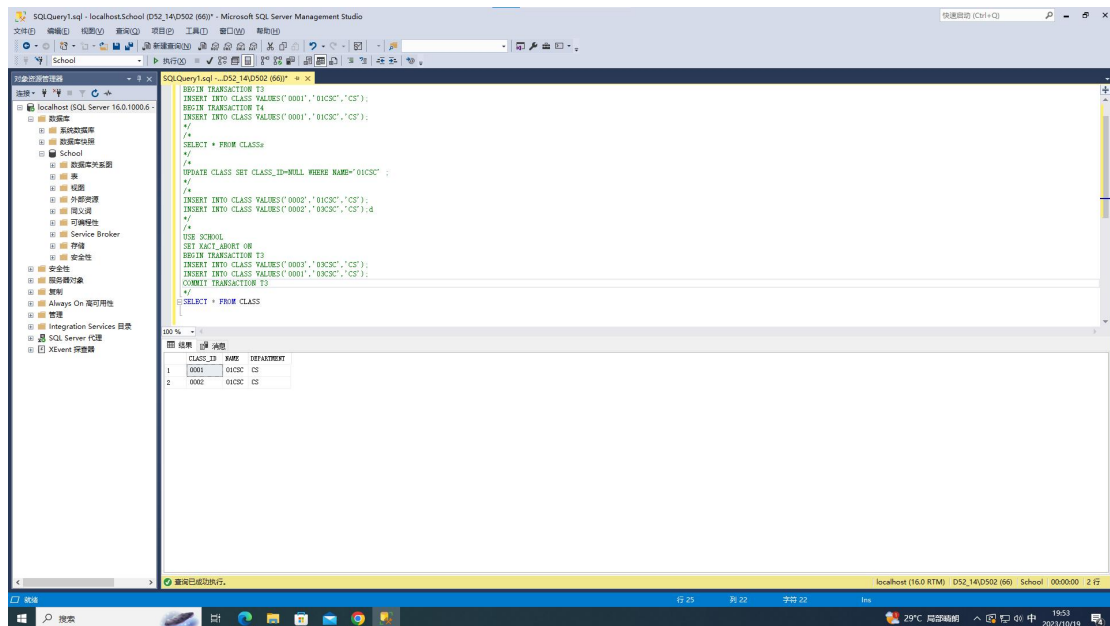


从表中看出（‘0002’，’ 01CSC ‘。 ’ CS ‘）插入成功，而（’ 0002 ‘，’ 03CSC ‘，’ CS ‘）没有插入成功，这是因为第一条插入成功后，执行下一条指令时表中有和其主键相同的元组而导致执行失败，而下一条指令的失败并不会影响之前执行成功的指令。

5. 在表 class 中，创建事务，并设置开启回滚，然后插入两个元组（‘0003’，’ 03CSC ‘。 ’ CS ‘），（’ 0001 ‘，’ 03CSC ‘，’ CS ‘），查看结果，表中有几条记录？

执行事务

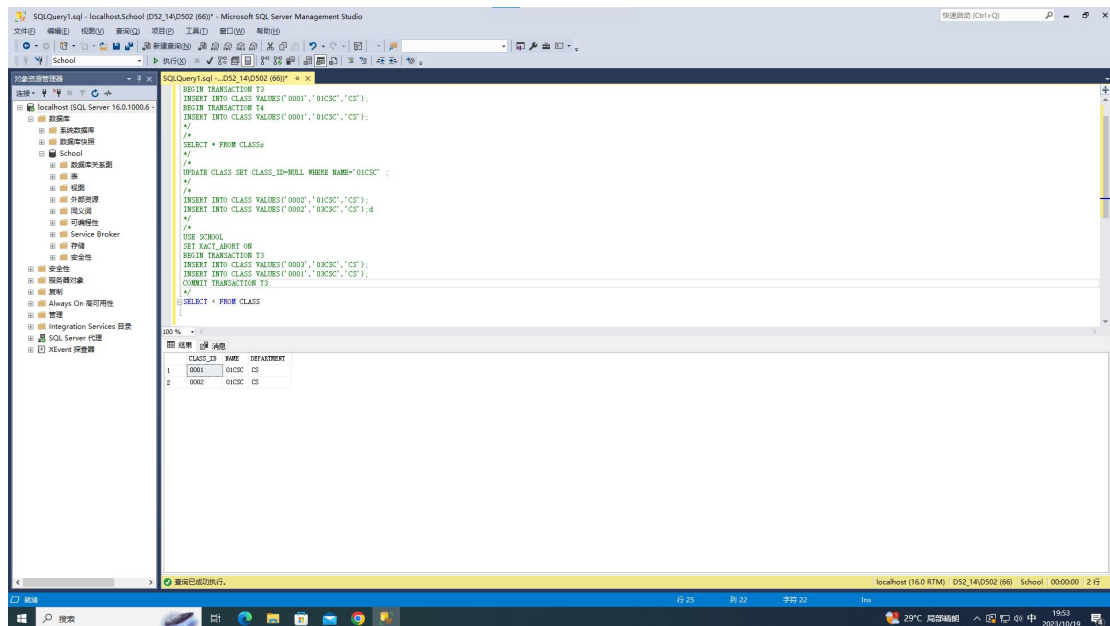




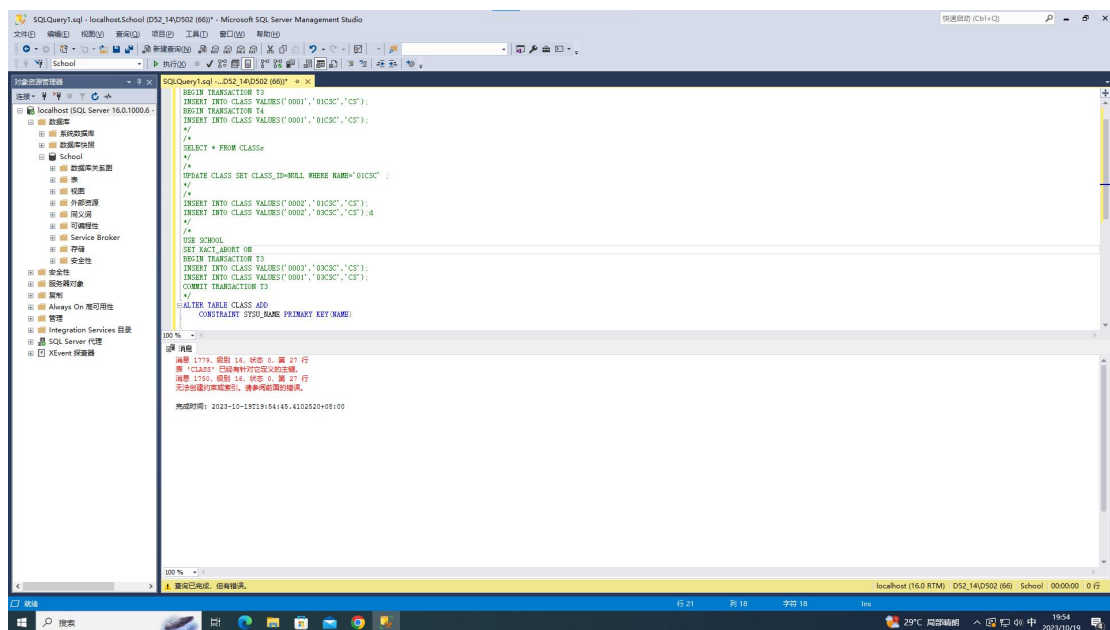
可以看到事务中的 2 条插入指令都没有成功,这是因为这是因为使用的是会回滚的事务, ( '0003' , ' 03CSC '。 ' CS ' ) 的插入成功, 但 ( ' 0001 ' , ' 03CSC ' , ' CS ' ) 因为表中有相同主键的元组而导致插入失败, 它的插入失败会影响前一条的插入回滚, 所以上一条插入的指令相当于没有执行入。

6. 在完成上面几步的前提下, 尝试设置' name '为主键, 看能否成功, 并思考原因。

此时 CLASS 表中的结果:



## 将 NAME 设置为主键失败



因为 CLASS 已有主键 CLASS\_ID，不能再设置主键。