

实验 4、数据视图

1.1 实验目的

熟悉 SQL 支持的有关视图的操作，能够熟练使用 SQL 语句来创建需要的视图，对视图进行查询和删除视图。

1.2 实验内容

- (1) 定义常见的视图形式，包括：
 - 行列子集视图
 - WITH CHECK OPTION 的视图
 - 基于多个基表的视图
 - 基于视图的视图
 - 带表达式的视图
 - 分组视图
- (2) 通过实验考察 WITH CHECK OPTION 这一语句在视图定义后产生的影响，包括对修改操作、删除操作、插入操作的影响。
- (3) 讨论视图的数据更新情况，对子行列视图进行数据更新。
- (4) 使用 DROP 语句删除一个视图，由该视图导出的其他视图定义仍在数据字典中，但已不能使用，必须显式删除。同样的原因，删除基表时，由该基表导出的所有视图定义都必须显式删除。

1.3 实验步骤

- (1) 创建一个行列子集视图 **chenht_CS_View**，给出选课成绩合格的学生的编号、教师编号、所选课程号和该课程成绩。
- (2) 创建基于多个基本表的视图 **chenht_SCT_View**，这个视图由学生姓名和他所选修的课程名及讲授该课程的教师姓名构成。
- (3) 创建带表达式的视图 **chenht_EXP_View**，由学生姓名及所选课程名和所有课程成绩都比原来多 5 分这几个属性组成。
- (4) 创建分组视图 **chenht_Group_View**，将学生的学号及他的平均成绩定义为一个视图。
- (5) 创建一个基于视图的视图，基于 (1) 中建立的视图，定义一个包括学生编号、学生所选课程数目和平均成绩的视图 **chenht_VV_View**。
- (6) 查询所有选修课程“数据库原理及其应用”的学生姓名。
- (7) 插入元组 (S52,T02,C02,59) 到视图 **chenht_CS_View** 中。若是在视图的定义中存在 WITH CHECK OPTION 字句对插入操作由什么影响。
- (8) 将视图 **chenht_CS_View** (包括定义 WITH CHECK OPTION) 中，所

有课程编号为 **C01** 的课程的成绩都减去 5 分。这个操作数据库是否会正确执行，为什么？如果加上 5 分（原来 95 分以上的不变）呢？

(9) 在视图 chenht_CS_View（包括定义 WITH CHECK OPTION）删除编号 **S03** 学生的记录，会产生什么结果？

(10) 删除视图 chenht_SCT_View 和视图 chenht_CS_View

1.4 实验结果

2) 创建基于多个基本表的视图 **chenht_SCT_View**，这个视图由学生姓名和他所选修的课程名及讲授该课程的教师姓名构成。

SQL 语句

```
create view chenht_SCT_View as
select cht_sname, cht_cname, cht_tname from chenht_Students,
chenht_Reports, chenht_Courses, chenht_Teachers
where chenht_Students.cht_sno = chenht_Reports.cht_sno
and chenht_Courses.cht_cno = chenht_Reports.cht_cno
and chenht_Teachers.cht_tno = chenht_Reports.cht_tno
```

实验结果

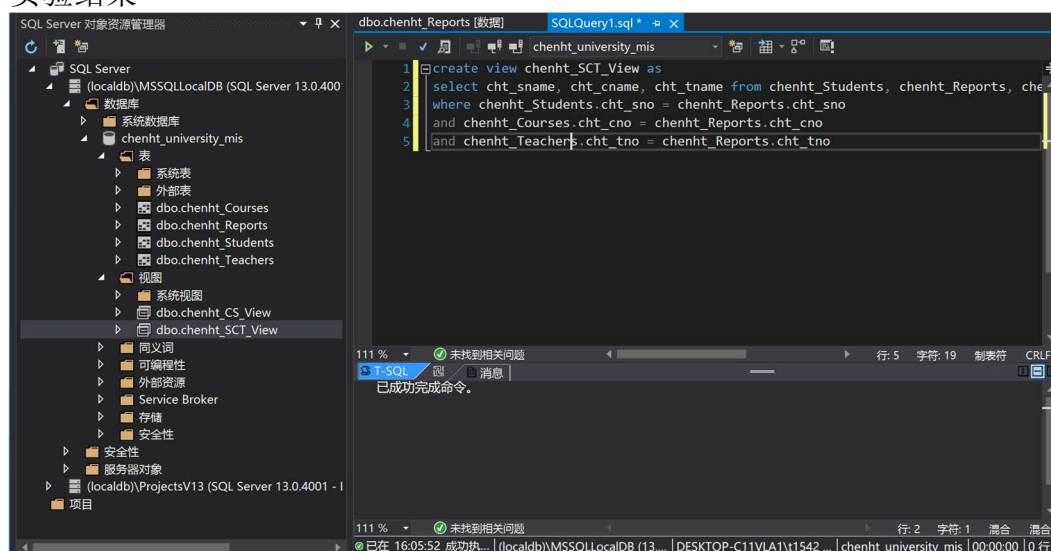
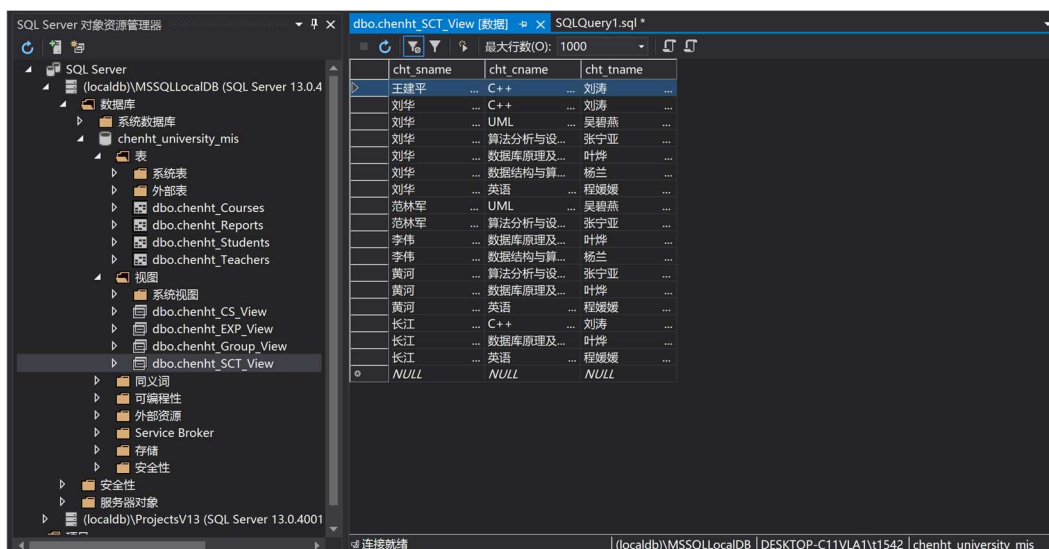


图 3-1 创建 chenht_SCT_View 的实验结果



The screenshot shows the SQL Server Enterprise Manager interface. On the left, the database structure is displayed, including the 'chenht_university_mis' database and its tables. The main window shows the results of the view 'dbo.chenht_SCT_View'. The results are displayed in a table with three columns: 'cht_sname', 'cht_cname', and 'cht_tname'. The data includes various students and their scores in different courses.

cht_sname	cht_cname	cht_tname
王建平	C++	刘涛
刘华	C++	刘涛
刘华	UML	吴碧燕
刘华	算法分析与设...	张宁亚
刘华	数据库原理及...	叶梓
刘华	数据结构与算...	杨兰
刘华	英语	程媛媛
范林军	UML	吴碧燕
范林军	算法分析与设...	张宁亚
李伟	数据库原理及...	叶梓
李伟	数据结构与算...	杨兰
黄河	算法分析与设...	张宁亚
黄河	数据库原理及...	叶梓
黄河	英语	程媛媛
长江	C++	刘涛
长江	数据库原理及...	叶梓
长江	英语	程媛媛
NULL	NULL	NULL

图 3-2 视图 chenht_SCT_View 的结果

3) 创建带表达式的视图 **chenht_EXP_View**, 由学生姓名及所选课程名和所有课程成绩都比原来多 5 分这几个属性组成。

SQL 语句

```
create view chenht_EXP_View as
select cht_sname, cht_cname, (cht_score+5)cht_score_2
from chenht_Students, chenht_Reports, chenht_Courses
where chenht_Students.cht_sno = chenht_Reports.cht_sno
and chenht_Courses.cht_cno = chenht_Reports.cht_cno
```

实验结果

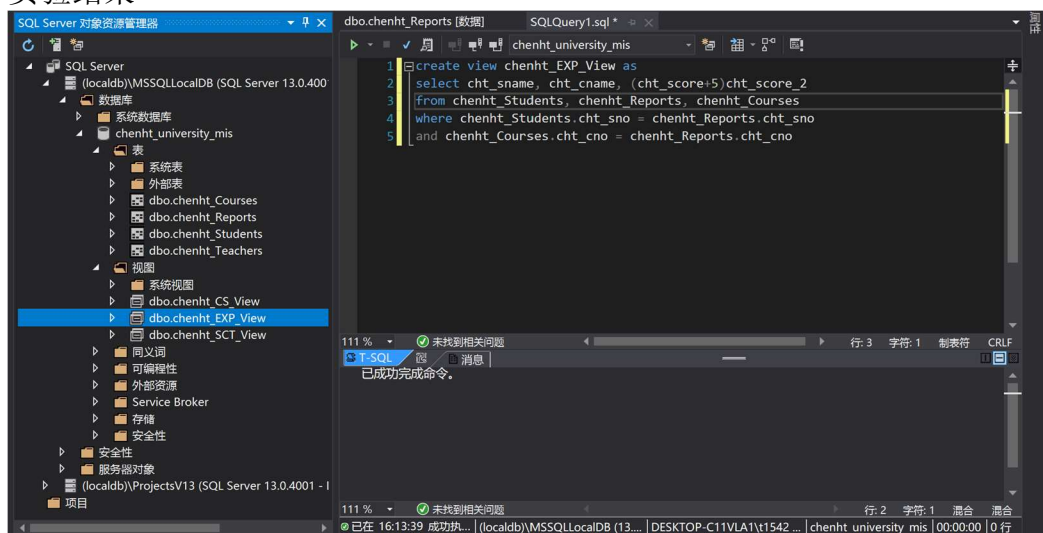
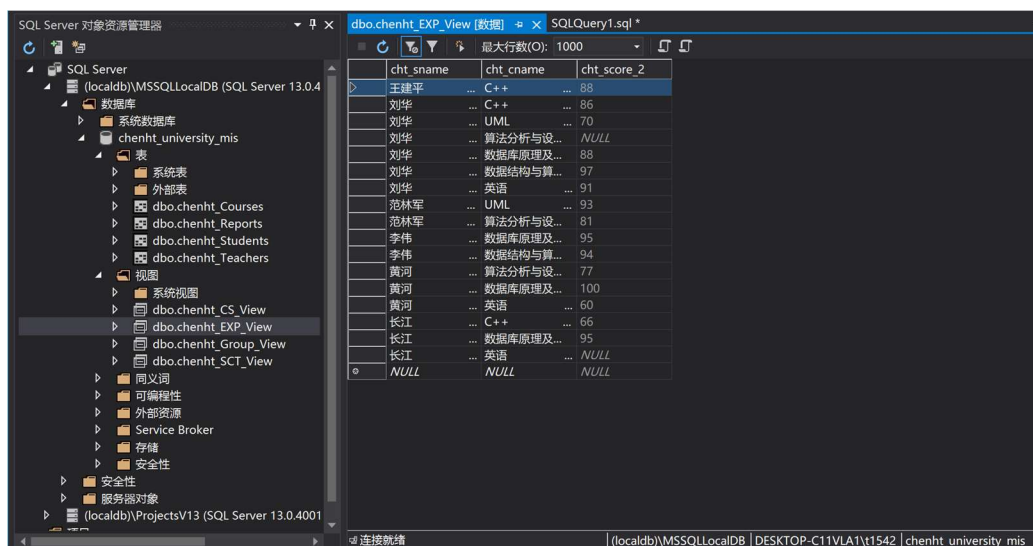


图 3-3 创建 chenht_Exp_View 实验结果



cht_sname	cht_cname	cht_score_2
王建平	C++	88
刘华	C++	86
刘华	UML	70
刘华	算法分析与设	NULL
刘华	数据库原理及	88
刘华	数据结构与算	97
刘华	英语	91
范林军	UML	93
范林军	算法分析与设	81
李伟	数据库原理及	95
李伟	数据结构与算	94
黄河	算法分析与设	77
黄河	数据库原理及	100
黄河	英语	60
长江	C++	66
长江	数据库原理及	95
长江	英语	NULL
NULL	NULL	NULL

图 3-4 视图 chenht_Exp_View 的结果

4) 创建分组视图 chenht_Group_View, 将学生的学号及他的平均成绩定义为一个视图。

SQL 语句

```
create view chenht_Group_View as
select chenht_Students.cht_sno, avg(cht_score)avg_score
from chenht_Students, chenht_Reports
where chenht_Students.cht_sno = chenht_Reports.cht_sno
group by chenht_Students.cht_sno
```

实验结果

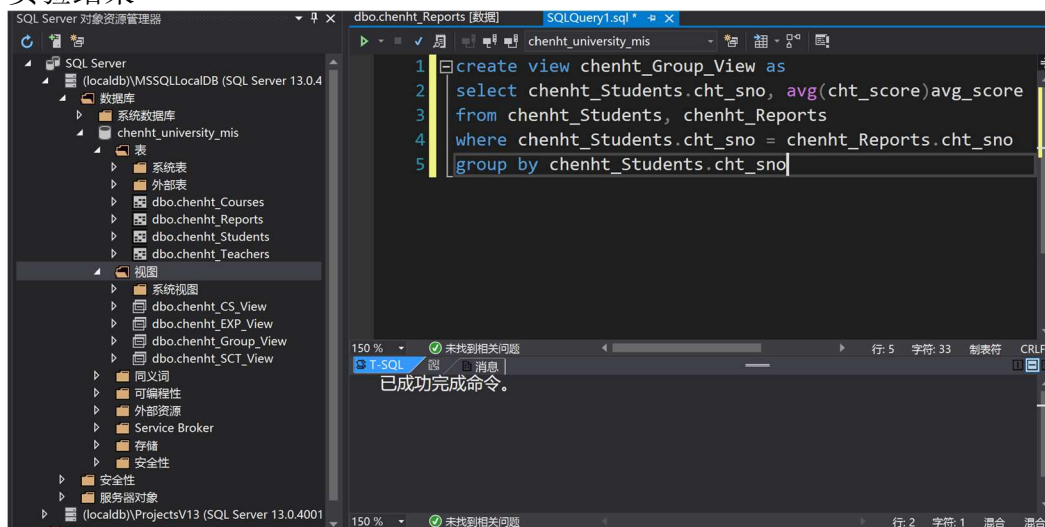
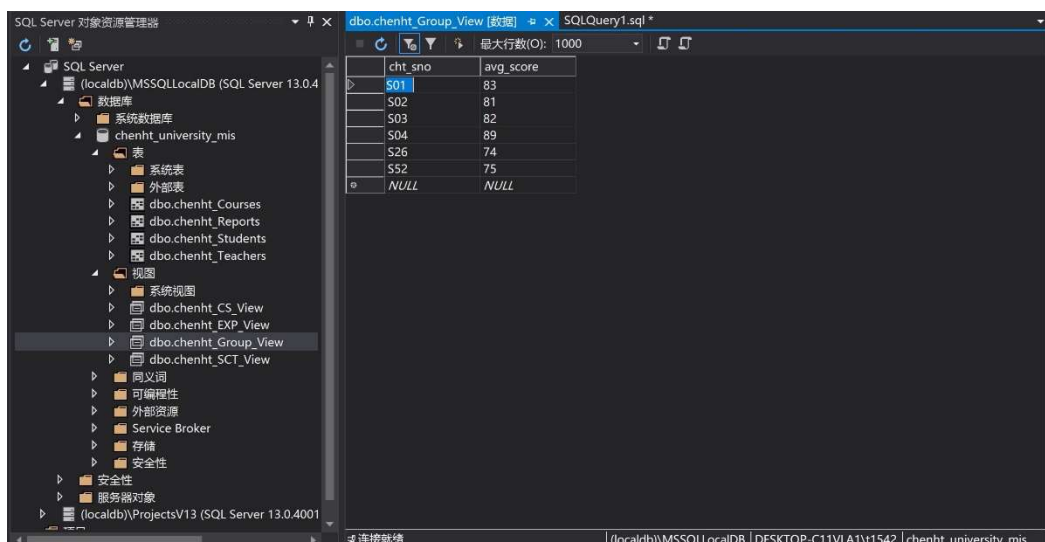


图 3-5 创建 chenht_Group_View 实验结果



cht_sno	avg_score
S01	83
S02	81
S03	82
S04	89
S26	74
S52	75
NULL	NULL

图 3-6 视图 chenht_Group_View 结果

7) 插入元组 (S52,T02,C02,59) 到视图 chenht_CS_View 中。若是在视图的定义中存在 WITH CHECK OPTION 字句对插入操作由什么影响。

SQL 代码:

a. 没有 WITH CHECK OPTION 选项 (默认)

```
insert into chenht_CS_View
values ('S52','T02','C02',59)
```

b. 有 WITH CHECK OPTION 选项

```
create or alter view chenht_CS_View as
select * from chenht_Reports
where cht_score >= 60
with check option
```

```
delete from chenht_Reports
where cht_sno = 'S52' and cht_cno = 'C02' and cht_tno = 'T02'

insert into chenht_CS_View
values ('S52','C02','T02',59)
create or alter view chenht_CS_View as
select * from chenht_Reports
where cht_score >= 60
with check option
```

实验结果

a. 没有 WITH CHECK OPTION 选项 (默认)

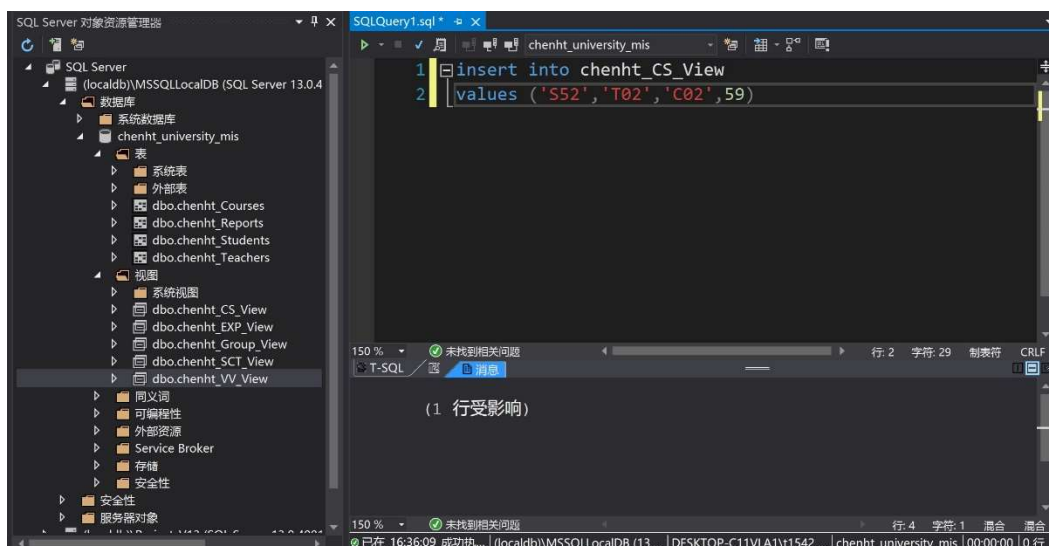


图 3-7 向 chen_CS_View 插入数据的结果(1)

b. 有 WITH CHECK OPTION 选项

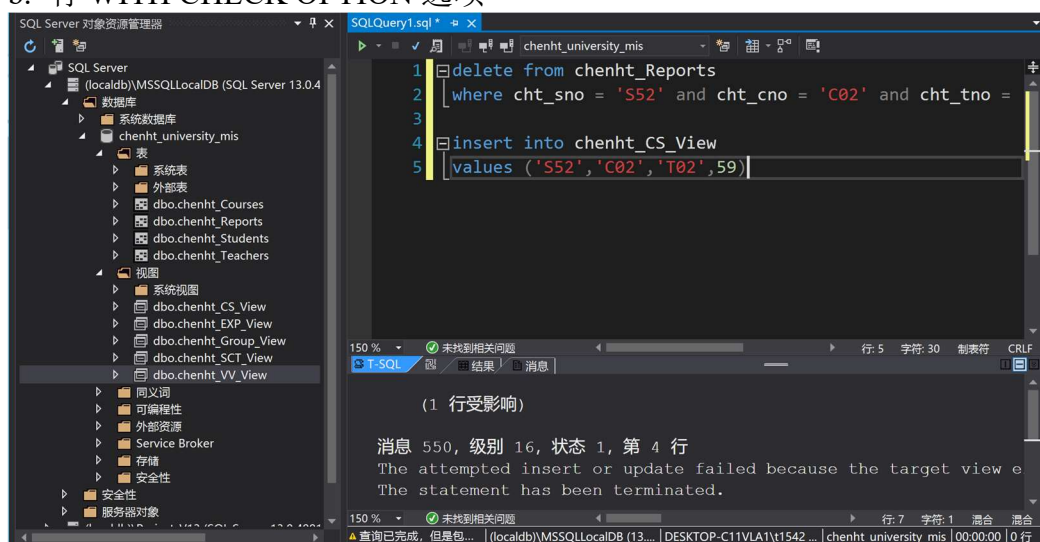


图 3-8 向 chen_CS_View 插入数据的结果(2)

8) 将视图 chenht_CS_View (包括定义 WITH CHECK OPTION) 中, 所有课程编号为 **C01** 的课程的成绩都减去 5 分。这个操作数据库是否会正确执行, 为什么? 如果加上 5 分 (原来 95 分以上的不变) 呢?

预测: 减去 5 分不会正确执行, 原因是元组 ('S52', 'T01', 'C01', 61) 在执行过后会不满足 WITH CHECK OPTION, 导致异常。而减去 5 分则会正确执行。

SQL 语句:

a. 减去 5 分

```
update chenht_CS_View
set cht_score = cht_score - 5
where cht_cno = 'C01'
```

b. 加上 5 分

```
update chenht_CS_View
set cht_score = cht_score + 5
where cht_cno = 'C01' and cht_score <= 95
```

实验结果:

a. 减去 5 分

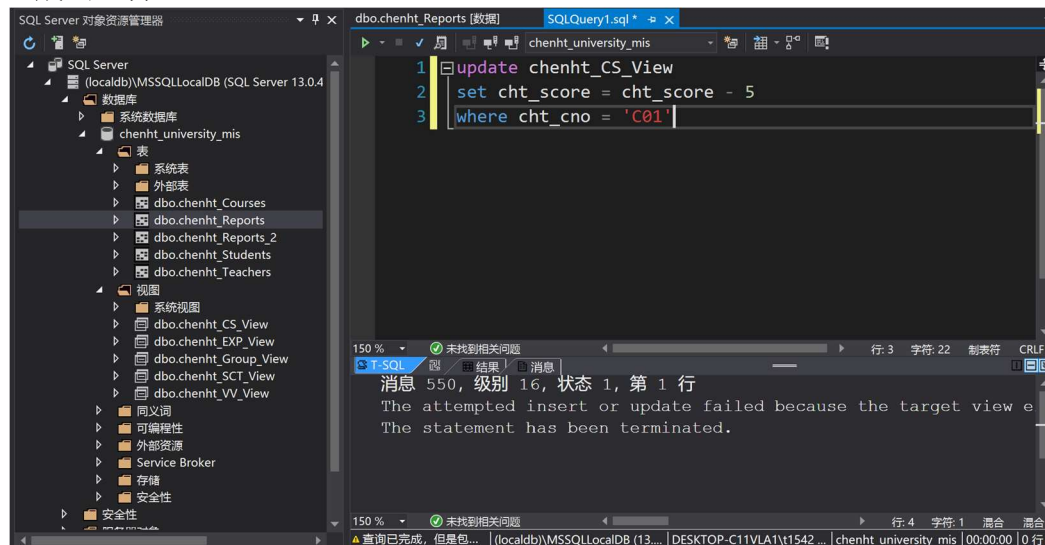


图 3-9 更新 chenht_CS_View 结果 (1)

b. 加上 5 分

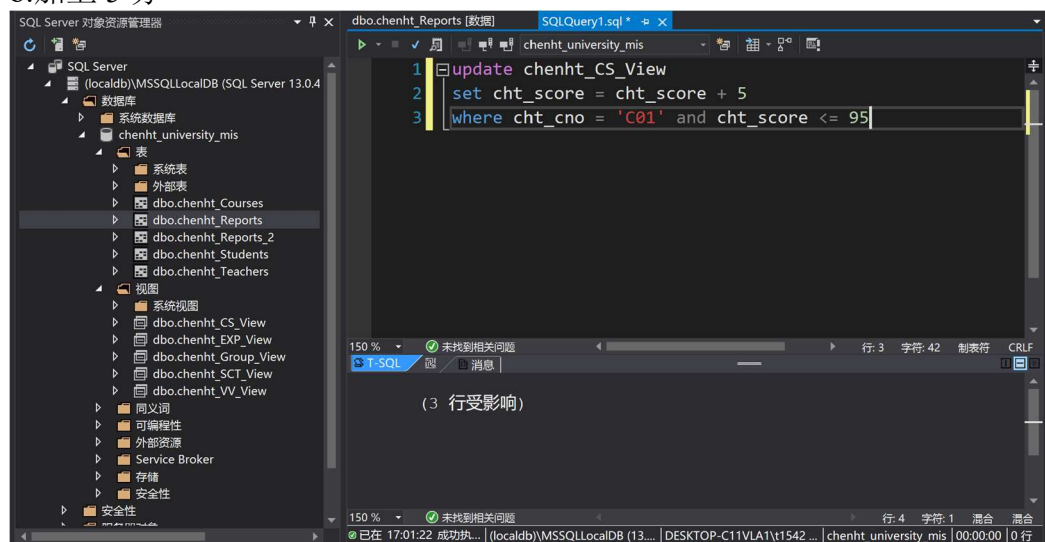


图 3-10 更新 chenht_CS_View 结果 (2)

1.5 实验体会

1) 实验反思

- 1、创建的视图并不是存储数据，而是存储模式的映射。当基本表的数据进行变更时，视图相关联的数据也会跟着变更。
- 2、只能对单一基本表（没有分组）导出的视图，即行例子集视图进行更新数据的操作，因此在其他类型的视图中 WITH CHECK OPTION 是没有意义的。
- 3、WITH CHECK OPTION 时，更新的数据项在更新之后仍然需要满足 WHERE 条件，并且只要有一条数据是不满足 WITH CHECK OPTION，所有的数据项均不会进行更新。因此需不需要使用 WITH CHECK OPTION 还需要根据实际应用决定。

2) 实验收获

对视图的定义、更新、删除、WITH CHECK OPTION 有了更加深入的了解。能够通过定义视图从而简化后续的 SELECT 查询语句。

对 WITH CHECK OPTION 的关键概念有了更加深入的了解。

