

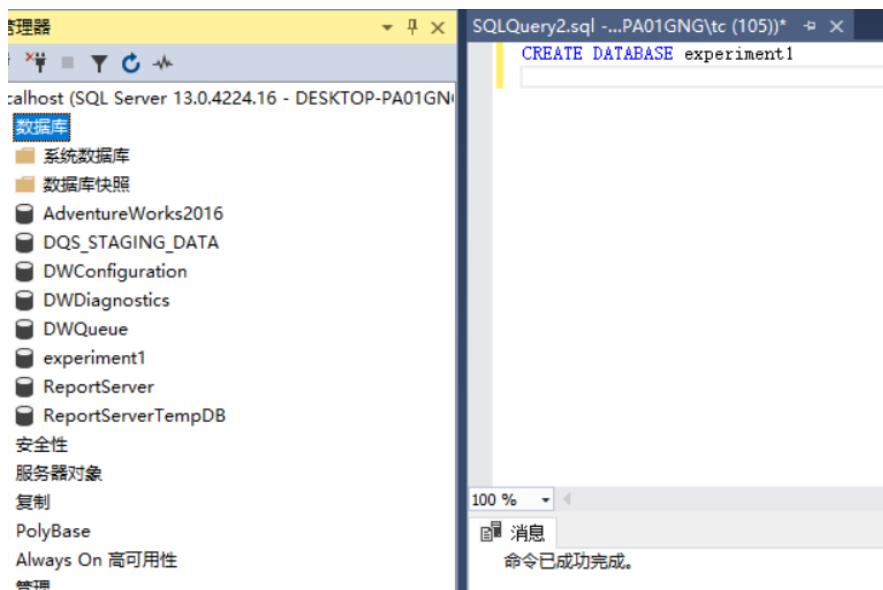
## 一．实验目的：

通过本实验，掌握 SQL Server 安全管理中的登录、数据库服务器角色、用户、数据库角色、特定对象权限等基本概念与安全机制，掌握数据库服务器角色授权、数据库角色授权和特定对象授权的方法与各种方法的差异。

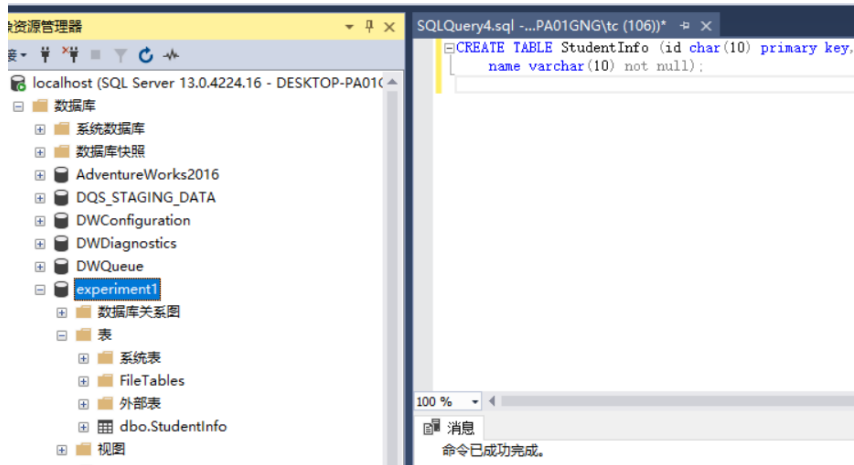
## 二．实验内容：

- (1) 创建一个登录并将创建的数据库设置为该登录的默认数据库。
- (2) 设置登录在默认数据库中的用户映射。
- (3) 尝试将登录加入到两个不同的数据库服务器角色中对默认数据库进行操作（如创建表 Score，查询表 StudentInfo 等）。注意每次只使用一个数据库服务器角色来验证授权。
- (4) 尝试使用两个不同的数据库角色进行相应的授权操作。
- (5) 只使用特定对象授权，完成相应的操作以验证授权的成功和没有授权时发生的错误

## 三．实验过程：

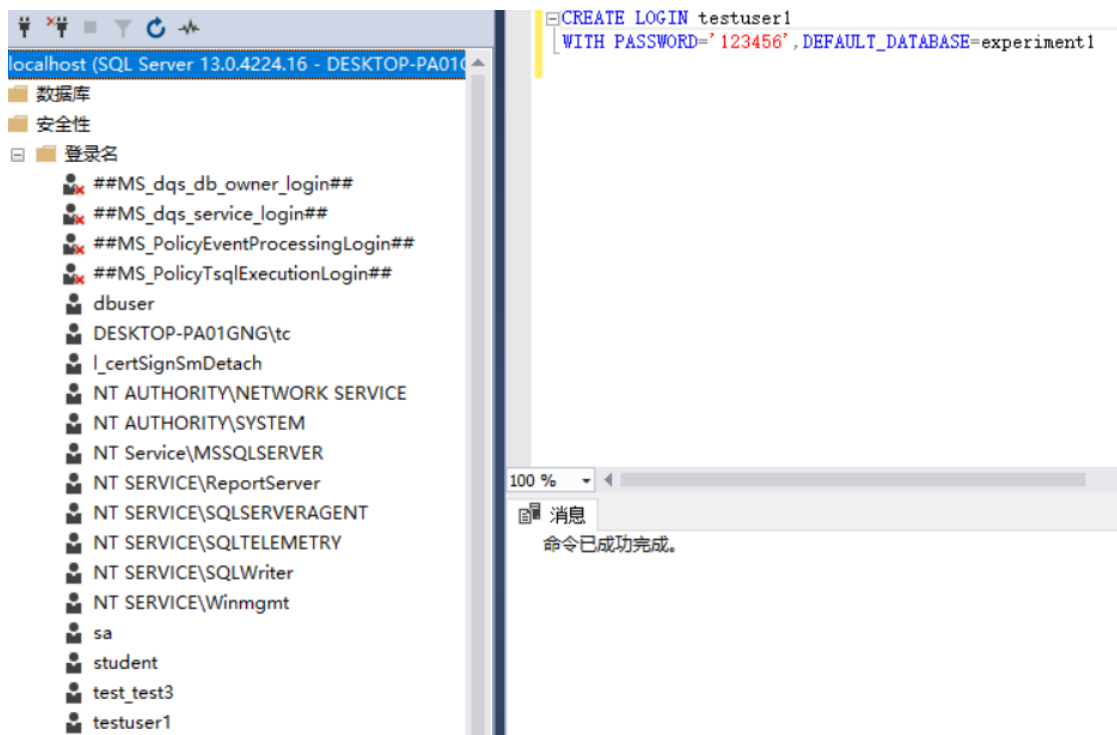


创建一个数据库 experiment1



在该数据库中创建一个表 StudentInfo, 该表有一个主码 id 和非空元素 name

(1)

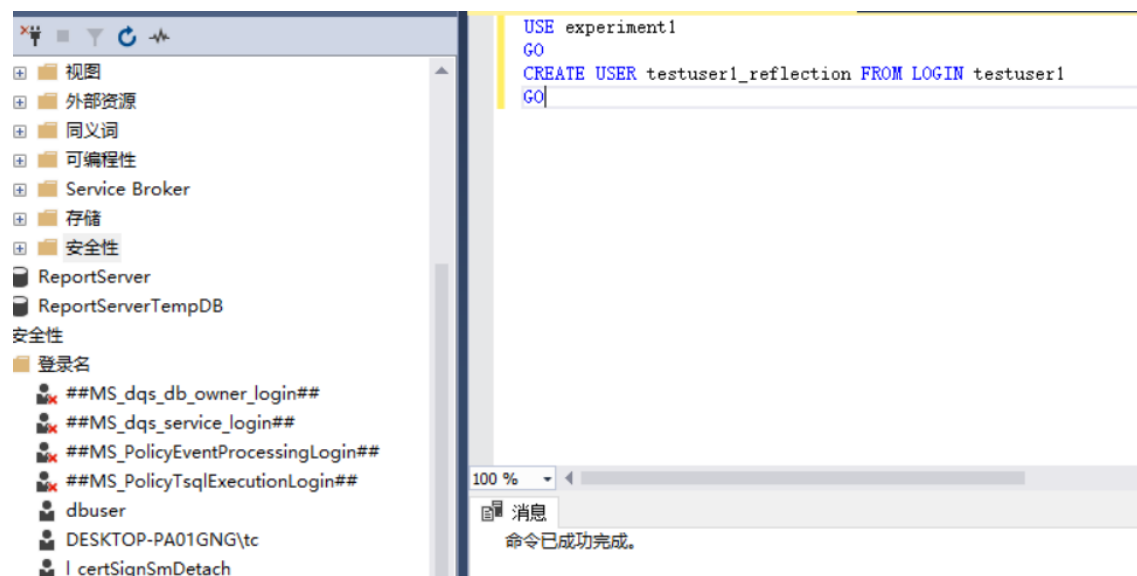


实验方法：创建了一个登陆 testuser1 并设置默认数据库为 experiment1

实验结果：创建成功

结果分析：指令正确

(2)

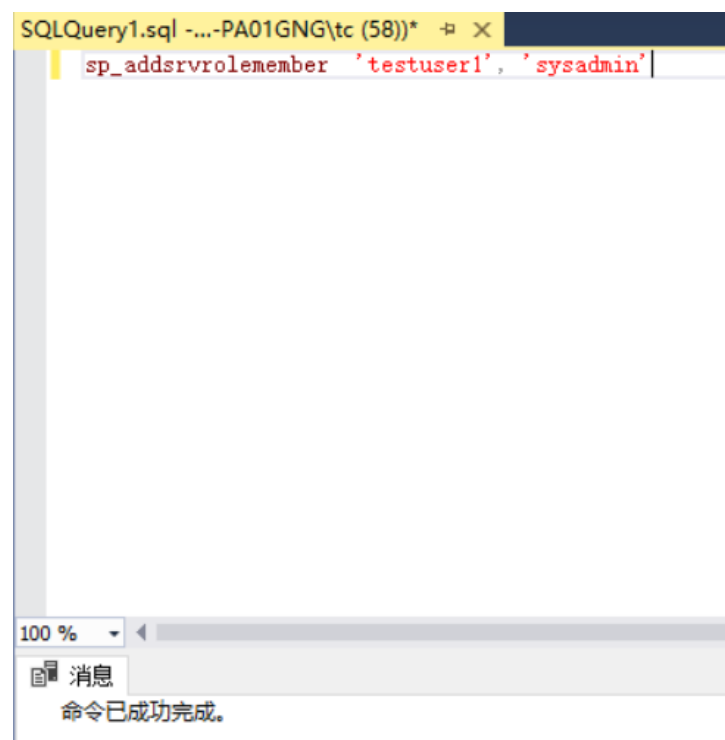


**实验方法：**在数据库 experiment1 中创建一个用户 testuser1\_reflection 并将其作为登陆 testuser1 在该数据库的用户映射。

**实验结果：**创建成功

**实验分析：**指令正确

(3)



SQLQuery5.sql - lo...(testuser1 (109))\*

```
CREATE TABLE ifo
(
    sno int,
    sname char(10)
)
```

100 %

消息

命令已成功完成。

SQLQuery12.sql - l...(testuser1 (113))\*

```
SELECT* FROM StudentInfo
```

100 %

结果 消息

id	name
----	------

SQLQuery2.sql -...PA01GNG\tc (101))\* ✕

```
sp_dropssrvrolemember 'testuser1','sysadmin'
```

100 %

消息

命令已成功完成。

SQLQuery11.sql -...A01GNG\tc (111))\* ✕

```
sp_addssrvrolemember 'testuser1','setupadmin'
```

100 %

消息

命令已成功完成。

SQLQuery10.sql - l...(testuser1 (115))\*

```
CREATE TABLE ifo2
(
    sno int,
    sname char(10)
)
```

100 %

消息

消息 262, 级别 14, 状态 1, 第 1 行  
在数据库 'experiment1' 中拒绝了 CREATE TABLE 权限。

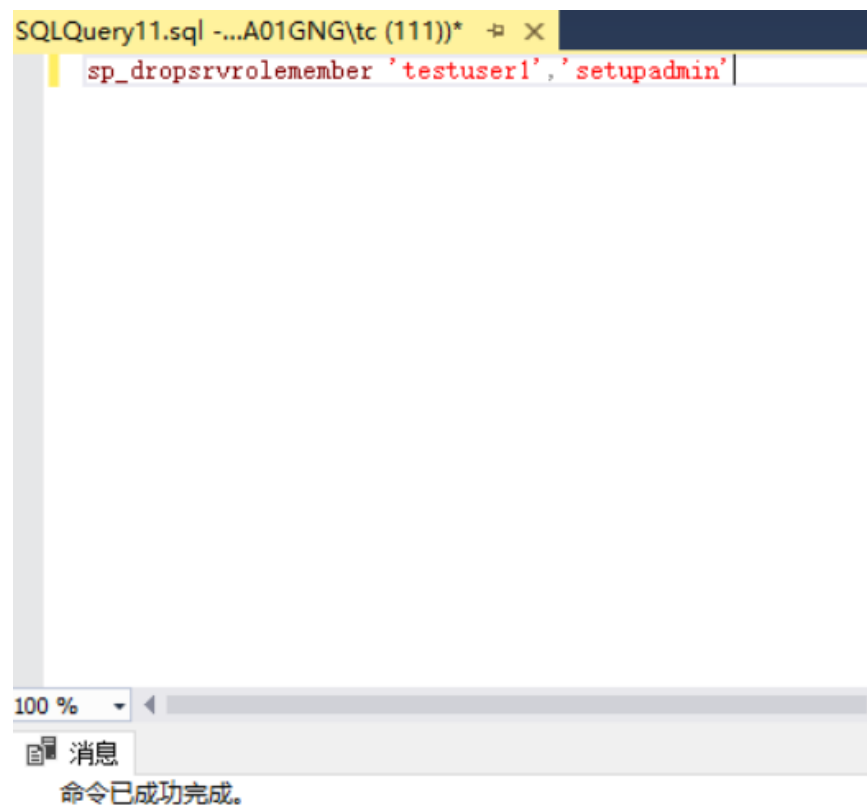
SQLQuery10.sql - l...(testuser1 (115))\*

```
SELECT* FROM StudentInfo
```

100 %

消息

消息 229, 级别 14, 状态 5, 第 1 行  
拒绝了对对象 'StudentInfo' (数据库 'experiment1', 架构 'dbo')的 SELEC



**实验方法：** 先将登陆 testuser1 加入到服务器角色 sysadmin 中，然后创建表 ifo，查询表 StudentInfo,然后将登陆 testuser1 从服务器角色 sysadmin 中删除，然后将登陆 testuser1 加入到服务器角色 setupadmin 中，然后创建表 ifo2，查询表 StudentInfo,然后将登陆 testuser1 从服务器角色 setupadmin 中删除，然后将登陆 testuser1 从服务器角色 setupadmin 中删除。

**实验结果：** 当登陆 testuser1 在服务器角色 sysadmin 中时，创建表 ifo 和查询表 StudentInfo 成功，当登陆 testuser1 在服务器角色 setupadmin 中时，创建表 ifo 和查询表 StudentInfo 失败。

**实验分析：** 服务器角色 sysadmin 有执行所有动作的权限，当登陆 testuser1 在服务器角色 sysadmin 中时，testuser1 拥有了执行所有动作的权限，所以创建表 ifo 和查询表 StudentInfo 成功，而服务器角色 setupadmin 拥有安装复制和管理扩展过程的权限，没有创建表和查询表的权限，故而当登陆 testuser1 在服务器角色 setupadmin 中时，创建 ifo2 和查询表 StudentInfo 失败。

(4)

SQLQuery13.sql -...A01GNG\{tc (116))\*

```
sp_addrolemember 'db_owner', 'testuser1_reflection'
```

100 %

消息

命令已成功完成。

SQLQuery14.sql -...1 (testuser1 (54))\*

```
CREATE TABLE ifo_second  
(  
    sno int,  
    sname char(10)  
)
```

100 %

消息

命令已成功完成。



SQLQuery14.sql -...1 (testuser1 (54))\* ❏ ✕

```
SELECT* FROM StudentInfo
```

100 % ▾

❏ 结果 ❏ 消息

id	name

SQLQuery15.sql -...A01GNG\tc (117))\* ❏ ✕

```
sp_droprolemember 'db_owner','testuser1_reflection'
```

100 % ▾

❏ 消息

命令已成功完成。

SQLQuery15.sql -...A01GNG\tc (117))\* ✕

```
sp_addrolemember 'db_accessadmin','testuser1_reflection'
```

100 %

消息

命令已成功完成。

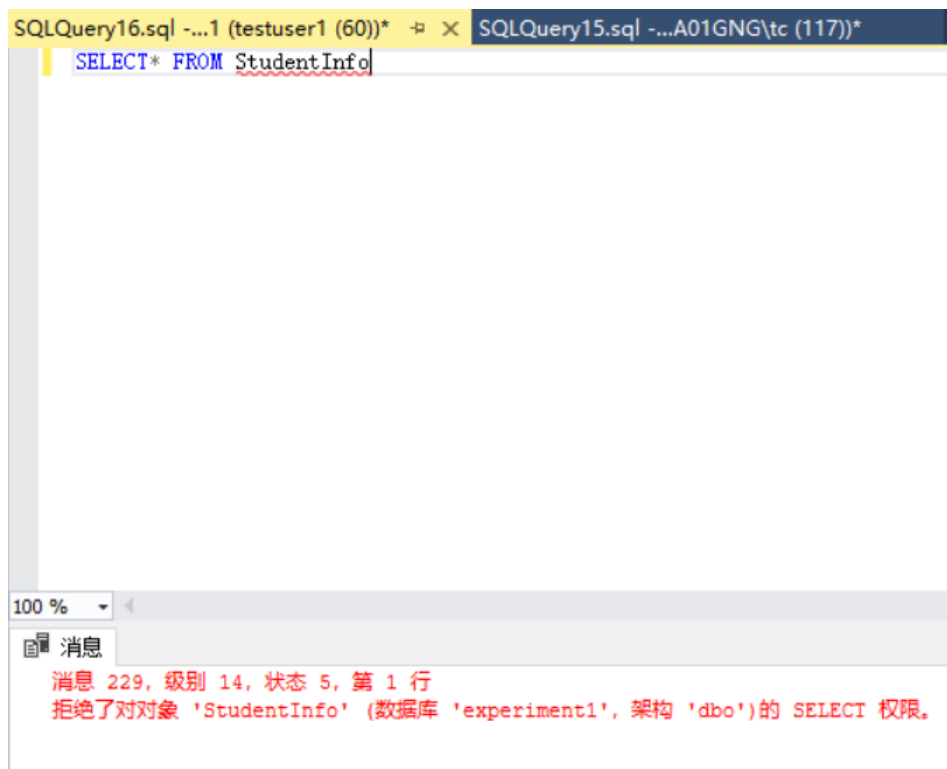
SQLQuery16.sql -...1 (testuser1 (60))\* ✕

```
CREATE TABLE ifo_second_2  
(  
    sno int,  
    sname char(10)  
)
```

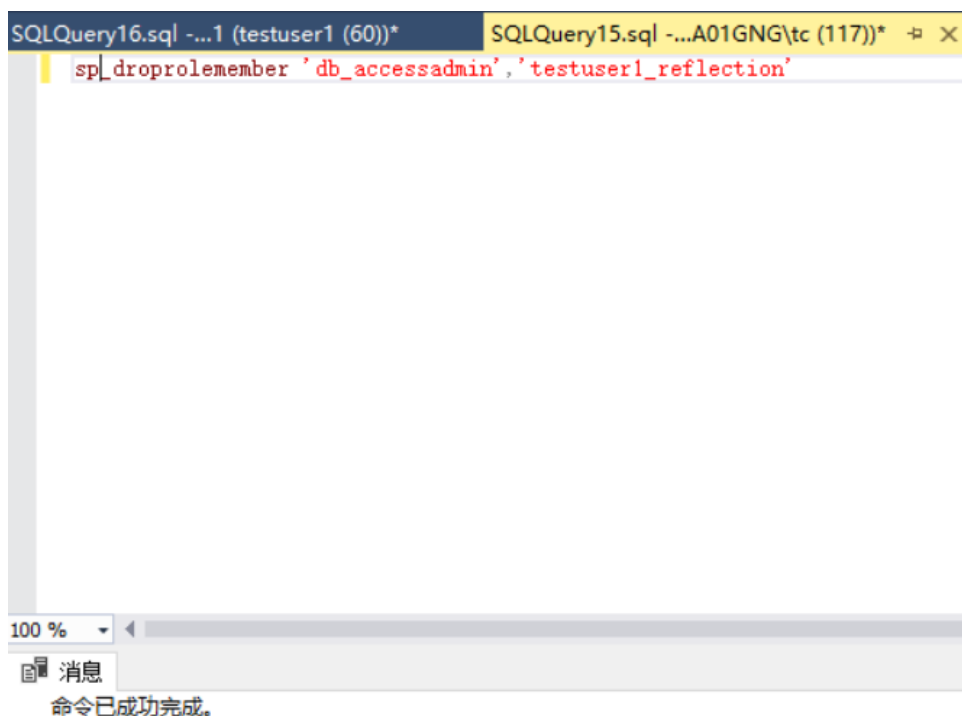
100 %

消息

消息 262, 级别 14, 状态 1, 第 1 行  
在数据库 'experiment1' 中拒绝了 CREATE TABLE 权限。



The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'SQLQuery16.sql -...1 (testuser1 (60))\*' and 'SQLQuery15.sql -...A01GNG\tc (117))\*'. The active tab contains the SQL query `SELECT* FROM StudentInfo`. Below the query editor, a message pane displays an error: '消息 229, 级别 14, 状态 5, 第 1 行 拒绝对对象 'StudentInfo' (数据库 'experiment1', 架构 'dbo')的 SELECT 权限。' (Message 229, Level 14, State 5, Line 1: The SELECT permission was denied for the object 'StudentInfo', database 'experiment1', schema 'dbo').



The screenshot shows the same SQL Server Enterprise Manager window. The active tab now contains the SQL command `sp_droprolemember 'db_accessadmin','testuser1_reflection'`. The message pane below shows the status: '命令已成功完成。' (Command completed successfully).

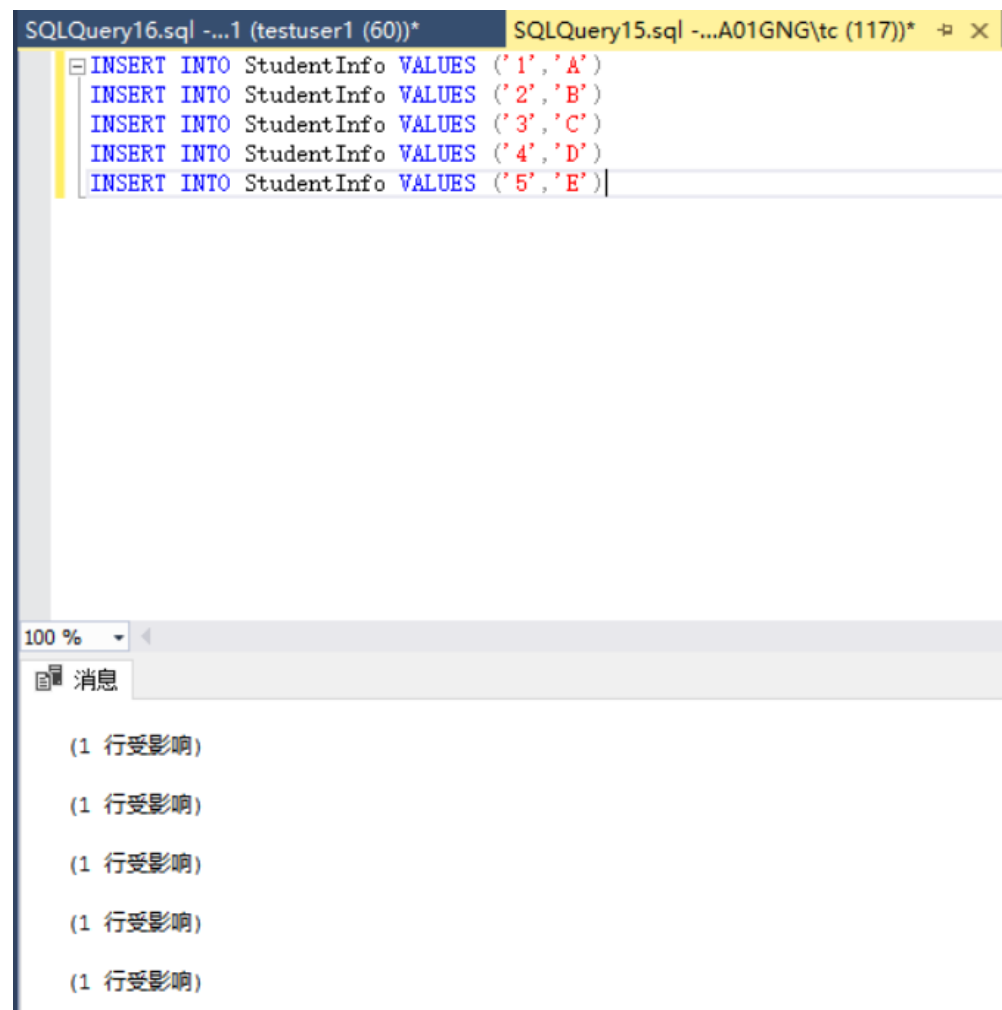
**实验方法：**先将用户 testuser1\_reflection 加入到数据库角色 db\_owner 中，然后创建表 ifo\_second，查询表 StudentInfo,然后将用户 testuser1\_reflection 从数据库角色 db\_owner 中删除，然后将用户 testuser1\_reflection 加入到数据库角色 db\_accessadmin 中，然后创建表 ifo\_second\_2，查询表 StudentInfo,然后将用户 testuser1\_reflection 从数据库角色 db\_accessadmin 中删除。

**实验结果：**当用户 testuser1\_reflection 在数据库角色 db\_owner 中时，创建表 ifo\_second 和查询表 StudentInfo 成功，当用户 testuser1\_reflection 在数据库角色 db\_accessadmin 中时，

创建表 ifo\_second\_2 和查询表 StudentInfo 失败。

**实验分析：**因为登录 testuser1 在数据库 experiment1 中映射到用户 testuser1\_reflection，当用户 testuser1\_reflection 在数据库角色 db\_owner 中时，数据库的角色 db\_owner 具有对数据库的有执行所有动作的操作权限，所以能够查询该表数据库角色 db\_owner testuser1 拥有了执行所有动作的权限，所以创建表 ifo 和查询表 StudentInfo 成功，而数据库角色 db\_accessadmin 只拥有添加和删除用户的权限，没有创建表和查询表的权限，故而当用户 testuser1\_reflection 在数据库角色 db\_accessadmin 中时，创建 ifo2 和查询表 StudentInfo 失败。

(5)



```
SQLQuery16.sql -...1 (testuser1 (60))*  SQLQuery15.sql -...A01GNG\tc (117))*
INSERT INTO StudentInfo VALUES ('1', 'A')
INSERT INTO StudentInfo VALUES ('2', 'B')
INSERT INTO StudentInfo VALUES ('3', 'C')
INSERT INTO StudentInfo VALUES ('4', 'D')
INSERT INTO StudentInfo VALUES ('5', 'E')
```

100 %

消息

- (1 行受影响)
- (1 行受影响)
- (1 行受影响)
- (1 行受影响)
- (1 行受影响)

SQLQuery16.sql -...1 (testuser1 (60))\* SQLQuery15.sql -...A01GNG\tc (117))\*

```
SELECT* FROM StudentInfo
SELECT name FROM StudentInfo
SELECT id FROM StudentInfo
UPDATE StudentInfo SET name='F' WHERE id='1'
INSERT INTO StudentInfo VALUES('6','G')
```

100 %

消息

拒绝了对对象 'StudentInfo' (数据库 'experiment1', 架构 'dbo')的 SELECT 权限。  
消息 229, 级别 14, 状态 5, 第 3 行  
拒绝了对对象 'StudentInfo' (数据库 'experiment1', 架构 'dbo')的 SELECT 权限。  
消息 229, 级别 14, 状态 5, 第 4 行  
拒绝了对对象 'StudentInfo' (数据库 'experiment1', 架构 'dbo')的 SELECT 权限。  
消息 229, 级别 14, 状态 5, 第 4 行  
拒绝了对对象 'StudentInfo' (数据库 'experiment1', 架构 'dbo')的 UPDATE 权限。  
消息 229, 级别 14, 状态 5, 第 5 行  
拒绝了对对象 'StudentInfo' (数据库 'experiment1', 架构 'dbo')的 INSERT 权限。

SQLQuery16.sql -...1 (testuser1 (60))\*

SQLQuery15.sql -...A01GNG\tc (117))\*

```
GRANT SELECT(name) ON StudentInfo TO testuser1_reflection
```

100 %

消息

命令已成功完成。

SQLQuery16.sql -...1 (testuser1 (60))\* × SQLQuery15.sql -...A01GNG\tc (117))\*

```
SELECT* FROM StudentInfo  
SELECT name FROM StudentInfo  
SELECT id FROM StudentInfo  
UPDATE StudentInfo SET name='F' WHERE id='1'  
INSERT INTO StudentInfo VALUES('6','G')
```

100 %

结果 消息

	name
1	A
2	B
3	C
4	D
5	E

SQLQuery16.sql -...1 (testuser1 (60))\* SQLQuery15.sql -...A01GNG\tc (117))\* ×

```
GRANT SELECT(id) ON StudentInfo TO testuser1_reflection
```

100 %

消息

命令已成功完成。

SQLQuery16.sql -...1 (testuser1 (60))\* X SQLQuery15.sql -...A01GNG\tc (117))\*

```
SELECT* FROM StudentInfo
SELECT name FROM StudentInfo
SELECT id FROM StudentInfo
UPDATE StudentInfo SET name='F' WHERE id='1'
INSERT INTO StudentInfo VALUES ('6','G')
```

100 %

结果 消息

	id	name
1	1	A
2	2	B
3	3	C
4	4	D
5	5	E

	name
1	A
2	B
3	C
4	D
5	E

	id
1	1
2	2
3	3
4	4
5	5

SQLQuery16.sql -...1 (testuser1 (60))\* X SQLQuery15.sql -...A01GNG\tc (117))\*

```
SELECT* FROM StudentInfo
SELECT name FROM StudentInfo
SELECT id FROM StudentInfo
UPDATE StudentInfo SET name='F' WHERE id='1'
INSERT INTO StudentInfo VALUES ('6','G')
```

100 %

结果 消息

(5 行受影响)

(5 行受影响)

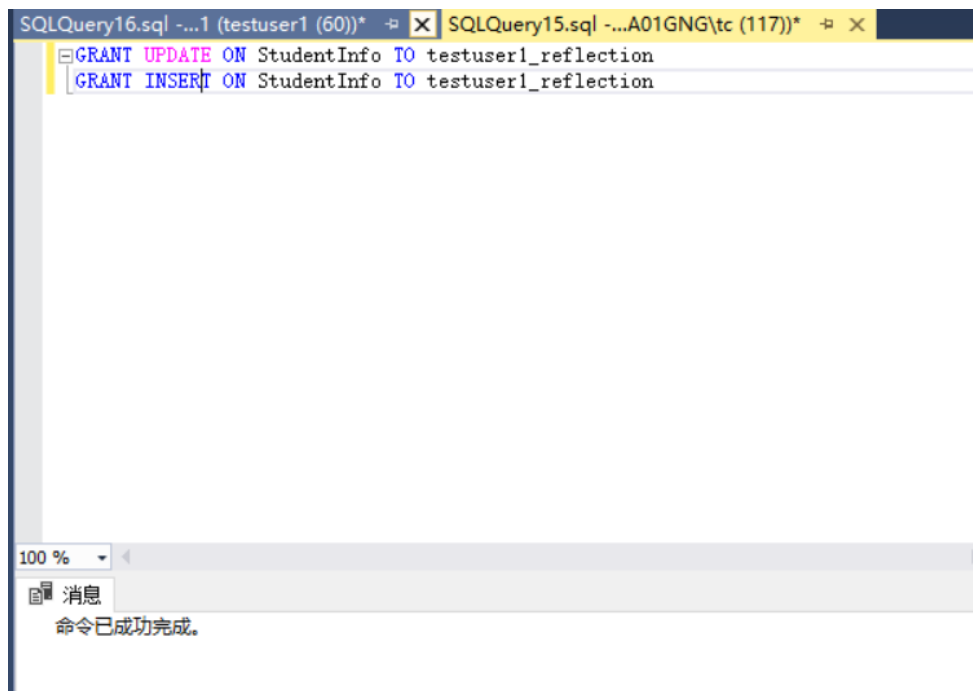
(5 行受影响)

消息 229, 级别 14, 状态 5, 第 4 行

拒绝了对对象 'StudentInfo' (数据库 'experiment1', 架构 'dbo')的 UPDATE 权限。

消息 229, 级别 14, 状态 5, 第 5 行

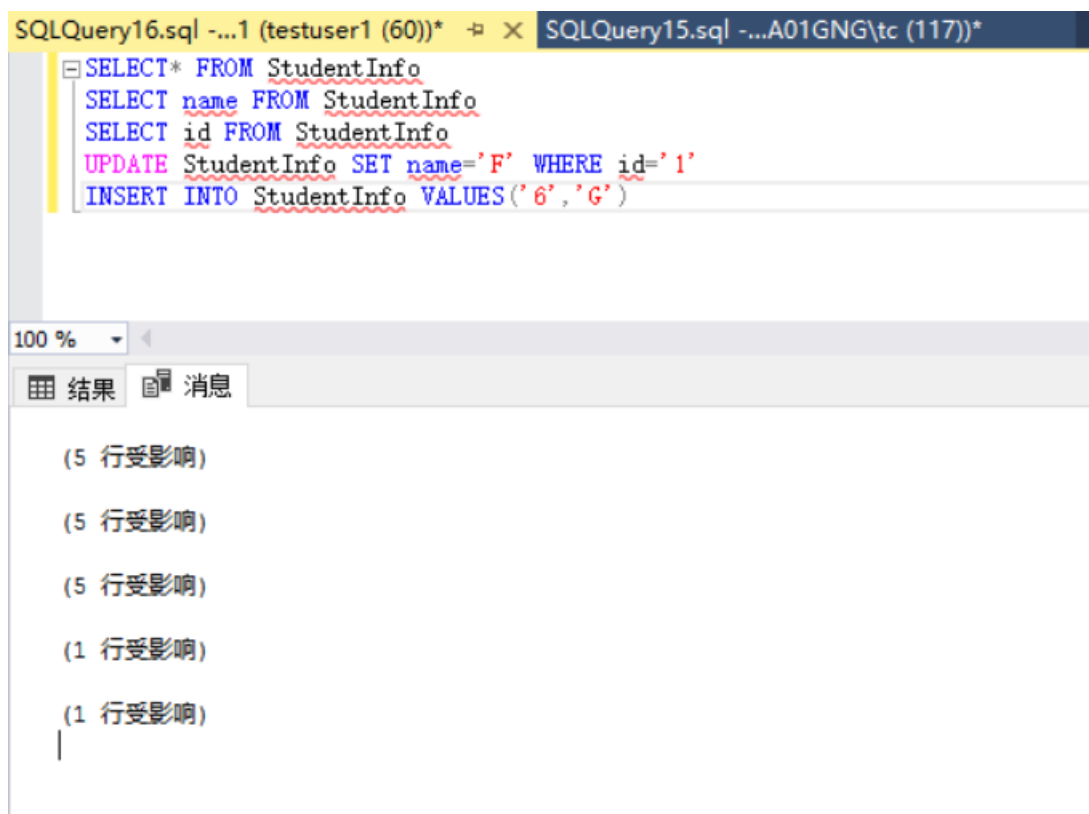
拒绝了对对象 'StudentInfo' (数据库 'experiment1', 架构 'dbo')的 INSERT 权限。



The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'SQLQuery16.sql -...1 (testuser1 (60))\*' and 'SQLQuery15.sql -...A01GNG\tc (117))\*'. The active tab contains the following SQL commands:

```
GRANT UPDATE ON StudentInfo TO testuser1_reflection
GRANT INSERT ON StudentInfo TO testuser1_reflection
```

Below the query editor, a status bar indicates '100 %' zoom and a message box titled '消息' (Message) with the text '命令已成功完成。' (Command completed successfully).



The screenshot shows the same SQL Server Enterprise Manager window with the following SQL commands in the active tab:

```
SELECT* FROM StudentInfo
SELECT name FROM StudentInfo
SELECT id FROM StudentInfo
UPDATE StudentInfo SET name='F' WHERE id='1'
INSERT INTO StudentInfo VALUES ('6','G')
```

Below the query editor, the '结果' (Results) tab is selected, showing the following output:

- (5 行受影响)
- (5 行受影响)
- (5 行受影响)
- (1 行受影响)
- (1 行受影响)

**实验方法：**首先通过 windows 用户登陆向 StudentInfo 表中插入 5 组数据，然后通过新建的登陆 testuser1 查询 StudentInfo 的所以信息和每一列信息，把 id 为 1 的那一行信息的 name 更新为 F，插入一条信息。然后给 testuser1 授权查询 name 的权限，查询 id 的权限，插入和更新的权限，并在每一次授权后执行一次上述的查询、更新、插入操作。

**实验结果：**未授权时，显示拒绝了对对象 StudentInfo 的 select、update、insert 的权限，给 testuser1 授权查询 name 的权限后，结果显示出了 StudentInfo 的 name 列信息，拒绝了对对象 StudentInfo 的 select (id)、update、insert 的权限，给 testuser1 授权查询 id 的权限后，



结果显示出了 StudentInfo 的所有信息，拒绝了对对象 StudentInfo 的 update、insert 的权限，给 testuser1 授权 update、insert 的权限后，结果显示了所有查询结果，并没有拒绝权限。

**实验分析：**未授权的时候，由于 testuser1\_reflection 没有任何操作权限，而登录 testuser1 在数据库 experiment1 中映射到用户 testuser1\_reflection，故而所以查询、更新、插入操作被拒绝。当对 testuser1\_reflection 授与对象 experiment1 的 name 列的查询权限，name 列的查询结果显示出来，而对 experiment1 的 name 列和所有列的查询以及更新和插入操作由于权限不足被拒绝。当对 testuser1\_reflection 授与对象 experiment1 的 id 列的查询权限，由于对象 experiment1 总共只有 id 和 name 两列，故而有了所有列的查询权限，所有显示了所有查询信息，而更新和插入操作由于权限不足被拒绝。当对 testuser1\_reflection 授与对象 experiment1 的更新和插入权限，testuser1 便有了对 experiment1 的查询、更新、插入权限，所有操作都被执行。