

数据库实验lab5

曹广杰

15352015 数据科学与计算机

授课教师：刘玉葆

2017/10/20

Content

数据库实验lab5

Content

实验目的

句法汇总

实验内容

用户授权

test1

test2

基于view的授权模式以及授权的传播

test3

test4

传递授权

test5

test6

权限回收

test7

test8

关于传递授权的架构分析

实验目的

1. 熟悉SQL的数据控制操作
2. 能够使用SQL语句对用户进行授予和收回权限

句法汇总

1. 权限的分享[with grant option]——test3
2. grant对于某些属性的授权[on view]——test3
3. 权限回收——test6
4. 权限回收级联——关于传授权限的架构分析

- 重复出现的授权语法

```
grant select
on students
to public
```

- 重复出现的回收权限的语法

```
revoke select
on teachers
from user1
```

实验内容

用户授权

test1

授予所有用户对表 STUDENTS的查询权限

```
grant select
on students
to public
```

思路比较简单，就不做分析了。

句法之用户授权：

```
grant 公开的权限
on 授权所在的关系
to 被授权用户
```

【public表示所有属性为public的用户】

test2

授予所有用户对表 COURSES的查询和更新权限

```
grant select, insert, update
on courses
to public
```

考察方式同上一题，理解简单，句法同上。

句法：多个权限之间使用 `,` 隔离开来。

基于view的授权模式以及授权的传播

test3

1. 授予USER1对表TEACHERS的查询，且允许USER1可以传播这些权限

```
grant select
on teachers
to user1
with grant option
```

句法：with grant option 允许其分享这种权限。一旦该用户获得这种权限，该用户可以将其所拥有的权限分享给其他的用户。

2. 以及更新工资的权限，且允许USER1可以传播这些权限

```
create view teachers_salary
as select salary
from TEACHERS
---
grant update
on teachers_salary
to user1
with grant option
```

注：

- grant可以向某一用户授权，但是该权限一定是基于view或者是table，而不能是查询语句，即使只对一个属性进行授权，也需要将该属性制成一个view，放在on的参数位置上；
- 创建view的语句必须要单独运行，运行之后清空query界面，再继续之后的授权工作；

test4

授予USER2对表CHOICES的查询,更新成绩的权限

```
grant select
on choices
to user2
-- create view shall run alone
create view Choices_Score
as
    select score
    from CHOICES

grant update
on Choices_Score
to user2
```

基于视图的查询与上文一样，句法也别无二致，删除掉传播权限即可。

传递授权

test5

由USER1授予USER2对表TEACHERS的查询权限和传播的此项权限的权利

```
grant select
on teachers
to user2
with grant option
```

题意分析：

- 之前笔者已经在管理员中对user1赋予了极大的权力，现在需要分享user1的权限，则进入user1的账户，对以上权限进行相同的操作即可。

test6

由USER2授予USER3对表TEACHERS的查询权限,和传播的此项权限的权利

```
grant select
on teachers
to user3
with grant option
```

基于视图的查询与上文一样，句法也别无二致，将被授权用户改为user3即可。

此时，如果在user3向user2授权依然有效。

权限回收

test7

取消USER1对表STUDENTS的查询权限

```
revoke select
on teachers
from user1
```

题意分析：

- 这种事情只有管理员有权限可以做，所以进入管理员账户执行以上语句

句法之权限回收：

```
revoke 公开的权限
on 授权所在的关系
from 被夺权用户
```

【public表示所有属性为public的用户】

test8

取消USER1和USER2的关于表COURSES的权限

```
revoke select
on courses
from user1, user2
```

在管理员账户中执行以上语句。

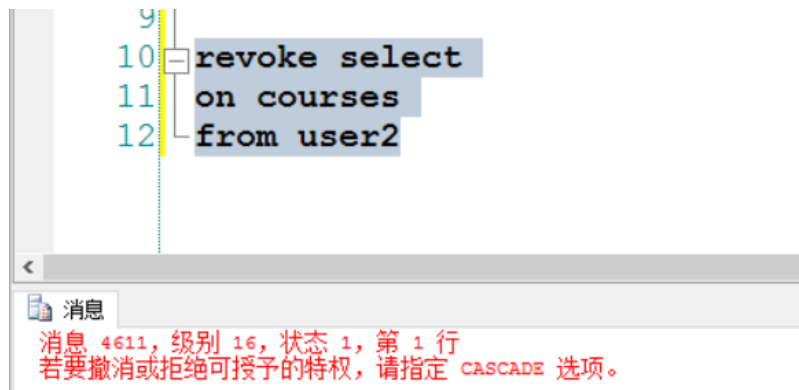
关于传递授权的架构分析

1. 如果sa授予user1的权限以及分享权限，user1授予user2权限

- 而sa继而回收了user1的权限，则权限依然在二者之间保存

因为不级联回收权限的话，这两个用户之间的权限是可以互相分享的。事实上，笔者的计算机上对于非级联回收权限的行为是判断为错误的——即或者不回收，回收请级联，不存在中间状态。

如果是user1向user2进行授权，那么，只要使用了 `with grant option`，回收的时候就要搭配上 `cascade`，否则是没有意义的。



```
9
10 revoke select
11    on courses
12    from user2
```

消息 4611，级别 16，状态 1，第 1 行
若要撤消或拒绝可授予的特权，请指定 CASCADE 选项。

2. 权限的回传分享：

在user1授予user2分享权限之后，user2依然可以回传给user1，这就使得sa在回收权限的时候，如果不使用级联是不能回收彻底的。

综上：

- 对于权限结构，sa（管理员）掌控着所有权限的管理，把握生杀予夺的大权，但是处理的时候常常只能批量化处理
- 对于权限本身，是可以重复授权的，存在被授权者向授权者授权的情况，所以权限更像是一个布尔数，可以无限地被叠加，反正只存在“yes”和“No”的形态。