

实验七 视图与访问控制

实验目的

- 了解视图的含义和作用
- 掌握创建视图和删除视图的方法；熟悉如何查看视图
- 了解数据库用户；熟悉创建、更改、删除用户的方法；了解查询用户的方法
- 了解权限、角色的不同点以及它们之间的关系；熟练掌握对角色的管理
- 了解数据库的不同权限；掌握为用户分配权限的方法

实验内容

- 创建视图：分别在单表上和多表上创建；查看视图及详细信息；删除视图
- 组角色管理：创建、查看、修改、删除组角色
- 账户管理：创建、更改、删除用户
- 组角色和用户权限管理：对组角色和用户授权及收回
- 数据库权限管理

实验作业

- 在 `EmployDB` 数据库中进行以下视图操作
 - 在 `employee` 表中创建收入 2000 以上的视图 `emp_view`
 - 输入如下代码：

```
CREATE VIEW emp_view AS
SELECT * from employee
WHERE e_salary > 2000;
```

即可创建视图;

- 查看 `emp_view` 中的所有员工信息
 - 输入代码：

```
SELECT * FROM emp_view;
```

员工信息如下：

	e_no integer	e_name character varying (20)	e_gender character (20)	dept_no integer	e_job character varying (20)	e_salary smallint	hiredate date
1	1004	JONES	m	20	MANAGER	2975	1998-05-18
2	1006	BLAKE	f	30	MANAGER	2850	1997-02-15
3	1007	CLARK	m	10	MANAGER	2450	2002-09-12
4	1008	SCOTT	m	20	ANALYST	3000	2003-05-12
5	1009	KING	f	10	PRESIDENT	5000	1995-01-01

- 创建部门 30(d_no=30) 的员工的薪资视图，视图中包含员工的 `id`、姓名和薪资
 - 使用此代码创建薪资视图 `salary_view`，代码如下：

```
CREATE VIEW salary_view AS
SELECT e_no,e_name,e_salary from employee
WHERE dept_no=30;
```

- 视图内容如下：

	e_no integer	e_name character varying (20)	e_salary smallint
1	1002	ALLEN	1600
2	1003	WARD	1250
3	1005	MARTIN	1250
4	1006	BLAKE	2850
5	1012	JAMES	950
6	1010	TURNER	1500

- 删除创建的视图 emp_view

输入代码如下，即可实现删除

```
DROP VIEW emp_view;
```

- 创建一个数据库 test01，进行以下操作
 - 选择 test01 数据库为当前数据库。在该数据库下创建数据表 mystudent，字段包括 numb varchar(20) 主键、name varchar(30) 非空、score int 非空

Step1.建立数据库表



- Step2.在该数据库下创建数据表 mystudent，字段包括 numb varchar(20) 主键、name varchar(30) 非空、score int 非空

- 在 test01 下右击表，点击创建



- 名称输入 `mystudent`

创建-表

General

列

高级

约束

分区

参数

安全

SQL

名称

mystudent

所有者

postgres

架构

public

×

表空间

选择一项...

分区表?

- 点击列，输入字段名称及数据类型；点击保存即可

创建表

General

列

高级

约束

分区

参数

安全

SQL

继承自表

选择要从其继承...

列

名称

数据类型

长度/精度

规模

不为 NUL...

主键?

默认值

<div><div></div><div></div></div> numb	<div><div></div><div></div></div> character varying	<div><div></div><div></div></div> 20	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>
<div><div></div><div></div></div> name	<div><div></div><div></div></div> character varying	<div><div></div><div></div></div> 30	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>
<div><div></div><div></div></div> score	<div><div></div><div></div></div> integer	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>	<div><div></div><div></div></div>

</

- 创建一个新账户，用户名称为 `admin01`，密码为 `pw123`

- 输入以下代码，创建用户

```
CREATE USER admin01 PASSWORD 'pw123';
```

- 创建一个新账户，用户名称为 `admin02`，密码为 `pw456`

- 输入以下代码，创建用户

```
CREATE USER admin02 PASSWORD 'pw456';
```

- 将数据库 `test01` 的所有者修改为 `admin01`，并在【对象浏览器】中查看 `test01` 的属性

- 输入以下代码

```
ALTER DATABASE test01 OWNER TO admin01;
```

- 右击数据库，查看 `test01` 的属性，发现所有者已经更改：

test01

General 定义 安全 参数 默认权限 高级 SQL

数据库

test01

所有者

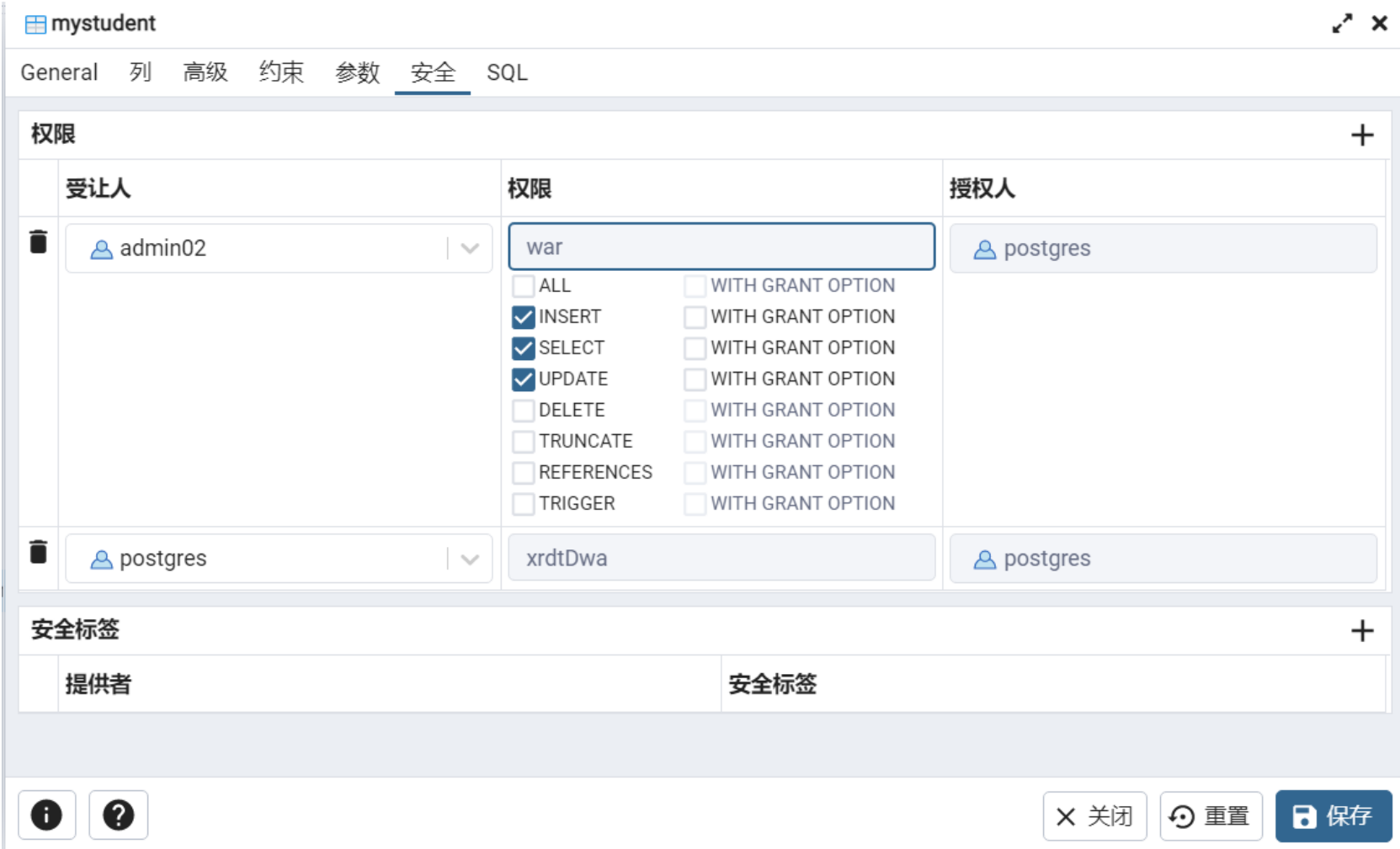
 admin01

- 允许用户 `admin02` 可以对数据表 `mystudent` 进行查询、插入和更新操作

- 输入以下代码，给予 `admin02` 权限

```
GRANT SELECT, INSERT, UPDATE
ON mystudent
TO admin02;
```

- 右击表，查看属性，发现授予权限成功；



- 删除 admin02 的账户信息

- 不能直接删除，报错为

```
ERROR:  role "admin02" cannot be dropped because some objects depend on it
DETAIL:  privileges for table mystudent
SQL 状态: 2BP01
```

因此要先解除权限；

- 输入以下代码，解除 admin02 对 mystudent 的权限

```
REVOKE SELECT, INSERT, UPDATE
ON mystudent
FROM admin02;
```

- 输入以下代码，删除账户

```
DROP USER admin02;
```

实验结果

见 实验作业 中截图；

源代码

全部代码见 实验作业 中代码；

思考题

(1) PostgreSQL 中视图和表的区别和联系是什么？

区别：

- 视图是已经编译好的SQL语句，而表不是；
- 表可以即时进行修改，但是视图只能由创建语句修改；

- 表是内容，视图是窗口
- 表占用物理空间，但视图不占用；
- 表是内模式，视图为外模式
- 表是全局模式中的表，为实表；而视图是局部模式的表，是虚表；
- 视图的建立和删除只影响视图本身，不会影响基表；

联系：

- 视图可以看作是建立在基表基础上的表；
- 一个视图可以对应一或多个基表；
- 视图是基表的抽象和在逻辑意义上建立的新关系；

(2) 如何撤销用户对数据表的操作权限？

使用 REVOKE 语句可以撤销权限；

语句的格式为

```
REVOKE <权限>[, <权限>] ...  
[ON <对象类型><对象名>]  
FROM <角色>[, <角色>] ... ;
```

(3) 思考组角色和用户的区别

创建用户默认具有登录（login）权限，而创建角色默认没有登录权限。