

# 浙江大学

## 数据库系统实验报告

作业名称:	SQL 安全性
姓 名:	秦嘉俊
学 号:	3210106182
电子邮箱:	HobbitQia@zju.edu.cn
联系电话:	18084011903
指导老师:	孙建伶

2023 年 3 月 27 日

# 实验名称

## 一、实验目的

熟悉通过 SQL 进行数据完整性控制的方法。

## 二、实验环境

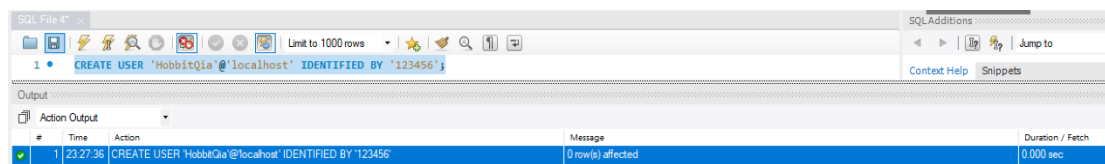
操作系统: Windows 10

数据库管理系统: MySQL 8.0.32.0

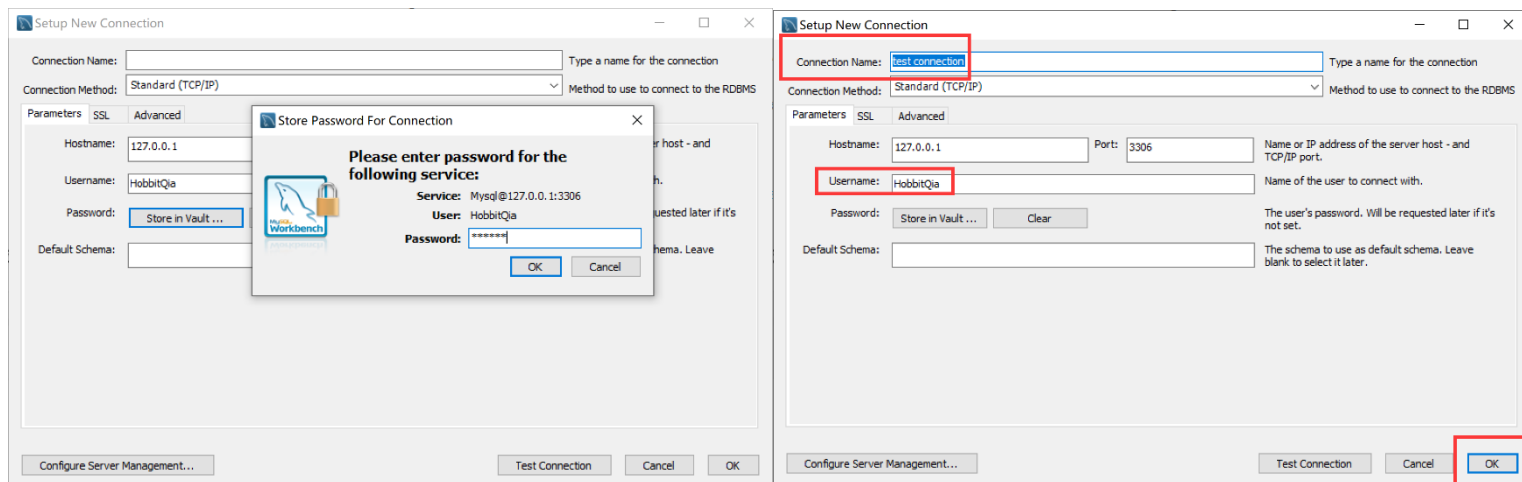
工具: MySQL Workbench

## 三、实验流程

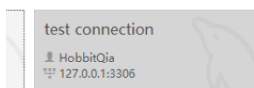
首先,用 root 身份登录 MySQL,并通过 `CREATE USER 'HobbitQia'@'localhost' IDENTIFIED BY '123456';` 新建普通用户 HobbitQia。



现在用这个新的用户身份登录。

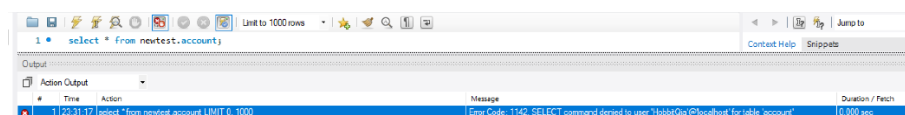


⑧

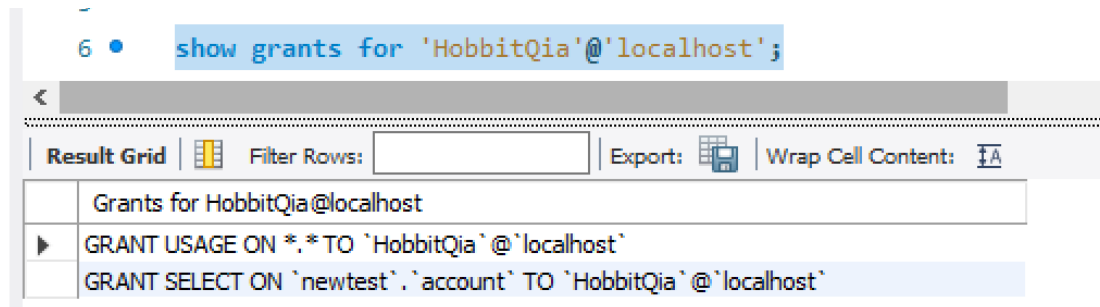


左侧的图表明我们已经登录成功。

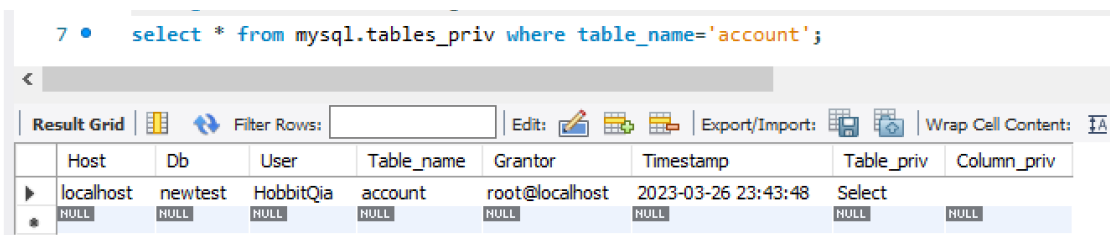
这个新建的用户不能直接访问 root 用户建立的表。我们尝试通过新建的用户访问数据库, (`select * from newtest.account;`) 却得到报错  
`select * from newtest.account LIMIT 0, 1000`  
Error Code: 1142. SELECT command denied to user 'HobbitQia'@'localhost' for table 'account'; 0.000 sec



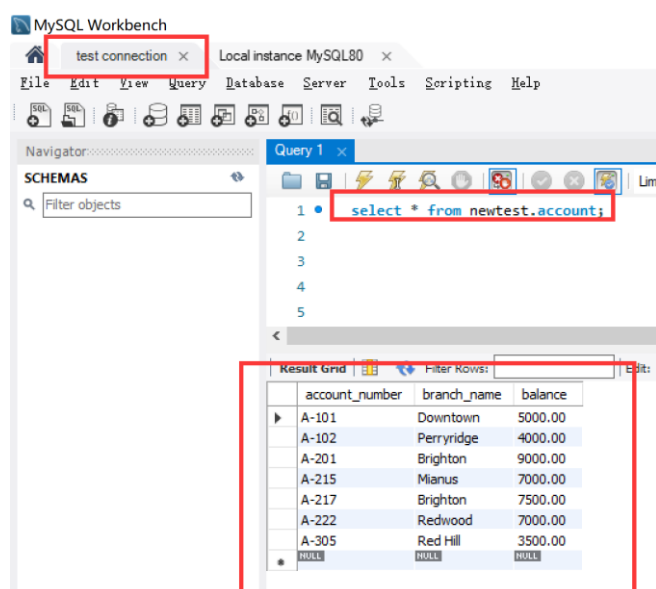
为此，需要回到 root 身份，并为该用户授权。输入语句 `GRANT select ON newtest.account TO 'HobbitQia'@'localhost';` 即可赋予给用户 select 的权限。授权后我们通过 `show grants for 'HobbitQia'@'localhost';`



还可以通过 `select * from mysql.tables_priv where table_name='account';` 查看哪些用户对 account 表有哪些权限。



可以看到用户已经有了对 account 表进行 select 的权限。于是我们回到用户 HobbitQia，成功通过 `select * from newtest.account;` 访问。



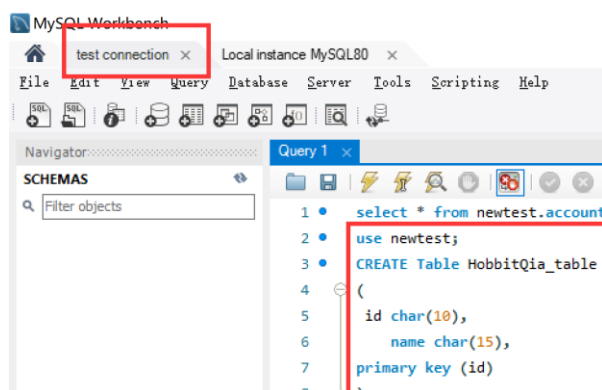
回到 root，通过 `REVOKE select ON newtest.account FROM 'HobbitQia'@'localhost';` 即可将刚刚授予的权限回收。

## 1. 建立表，考察表的生成成者拥有该表的哪些权限。

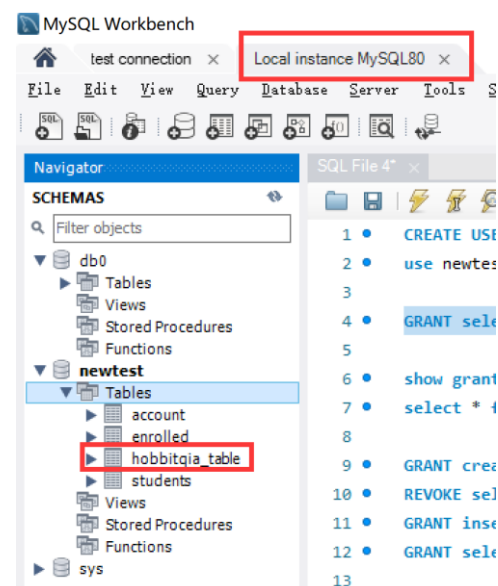
为了以普通用户身份建表，首先需要在 root 身份为 testuser1 授予在某个数据库上建表的权限，例如：

`GRANT create ON newtest.* TO 'HobbitQia'@'localhost';`

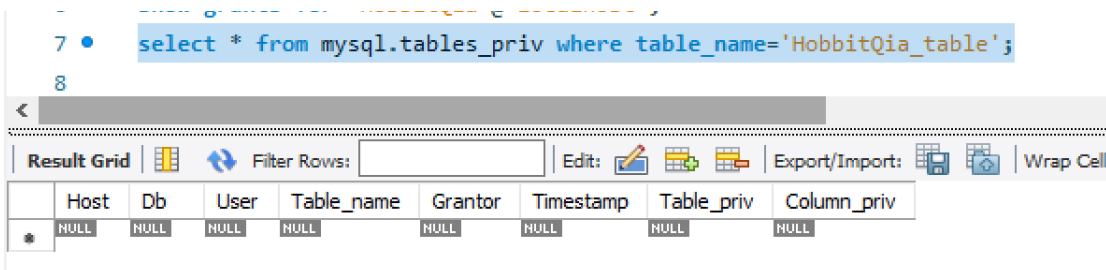
随后即可以 HobbitQia 的身份建表 HobbitQia\_table。



在 root 身份刷新数据库，看到表 HobbitQia\_table 确实建立成功。



此时我们查看哪些用户对 HobbitQis\_table 有哪些权限，发现 HobbitQia 对这个表并没有任何权限。

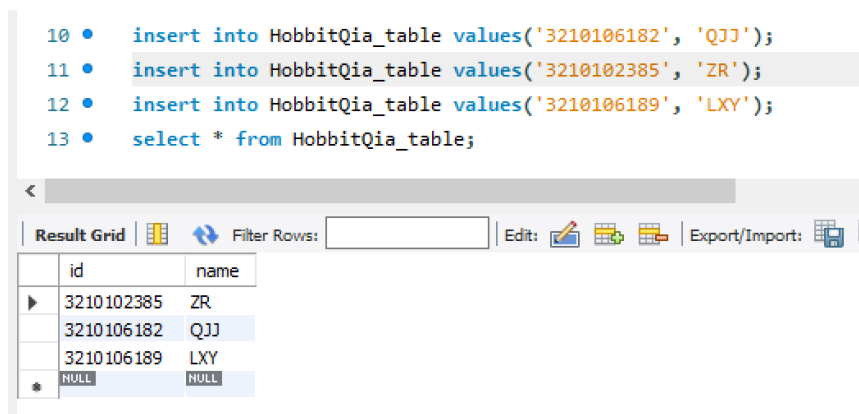


2. 使用 SQL 的 grant 和 revoke 命令对其他用户进行授权和权力回收，考察相应的作用。

回到 root 用户，授予 HobbitQia 对 HobbitQia\_table 插入和 select 的权限。

```
11 • GRANT insert ON newtest.hobbitqia_table TO 'HobbitQia'@'localhost';
12 • GRANT select ON newtest.hobbitqia_table TO 'HobbitQia'@'localhost';
```

随后以 HobbitQia 的身份插入数据，并查看表。



最后我们用 Revoke 语句收回权限，此时再次运行刚才 insert 和 select 的语句会得到报错信息，提示我们缺少权限。

```
14 • REVOKE insert ON newtest.hobbitqia_table FROM 'HobbitQia'@'localhost';
15 • REVOKE select ON newtest.hobbitqia_table FROM 'HobbitQia'@'localhost';
```

### 3. 建立视图，并把该视图的查询权限授予其他用户，考察通过视图进行权限控制的作用。

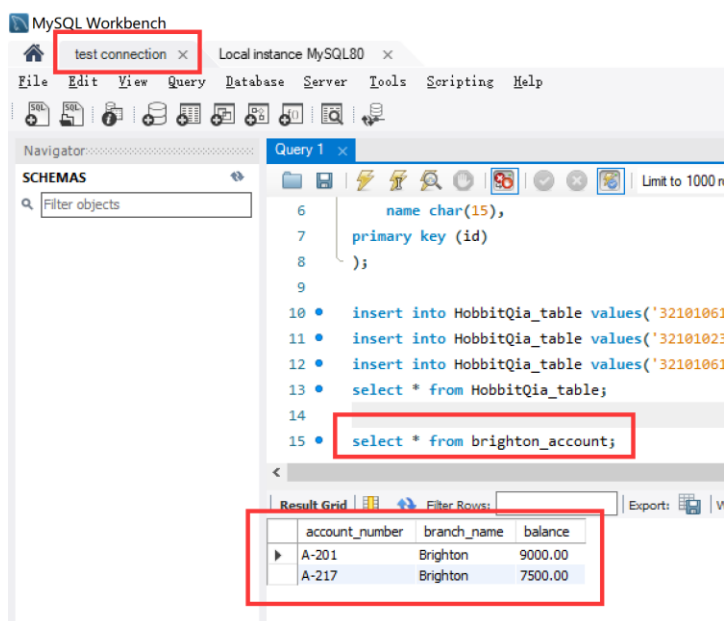
利用上次实验的内容，通过 root 用户建立一张视图 brighton\_account.

现在通过下面的语句把该视图的查询权限授予 testuser1:

```
GRANT select ON newtest.brighton_account TO 'HobbitQia'@'localhost';
```

```
18 • create view brighton_account as select * from account where account.branch_name = 'brighton';
19 • GRANT select ON newtest.brighton_account TO 'HobbitQia'@'localhost';
```

用 HobbitQia 身份查询该视图:



## 四、遇到的问题及解决方法

在以 HobbitQia 身份创建表 HobbitQia\_table 时，发现始终运行失败，

```
23:51:57 CREATE Table HobbitQia_table ( id char(10), name char(10), primary
key (id) ) Error Code: 1046. No database selected Select the default DB to be
used by double-clicking its name in the SCHEMAS list in the sidebar. 0.000
sec
```

后来经过上网搜索报错信息发现，是没有未选定默认的数据库。于是我加上 use newtests; 后即可运行成功。

## 五、总结

在本次实验中通过建立表，考察了表的生成者拥有该表的哪些权限。使用 SQL 的 grant 和 revoke 命令对其他用户进行授权和权力回收，考察了相应的作用。最后通过建立视图，并

把该视图的查询权限授予其他用户，考察了通过视图进行权限控制的作用。