实验名称

一、 实验目的

1. 熟悉通过 SQL 进行数据安全性控制的方法。

二、 实验环境

1. 数据库管理系统：MySQL

三、 实验流程

1. 创建用户

登录 root 用户。如图 3.1.1 所示，查看用户，可以看见除了系统预设的用户外，没有普通用户。

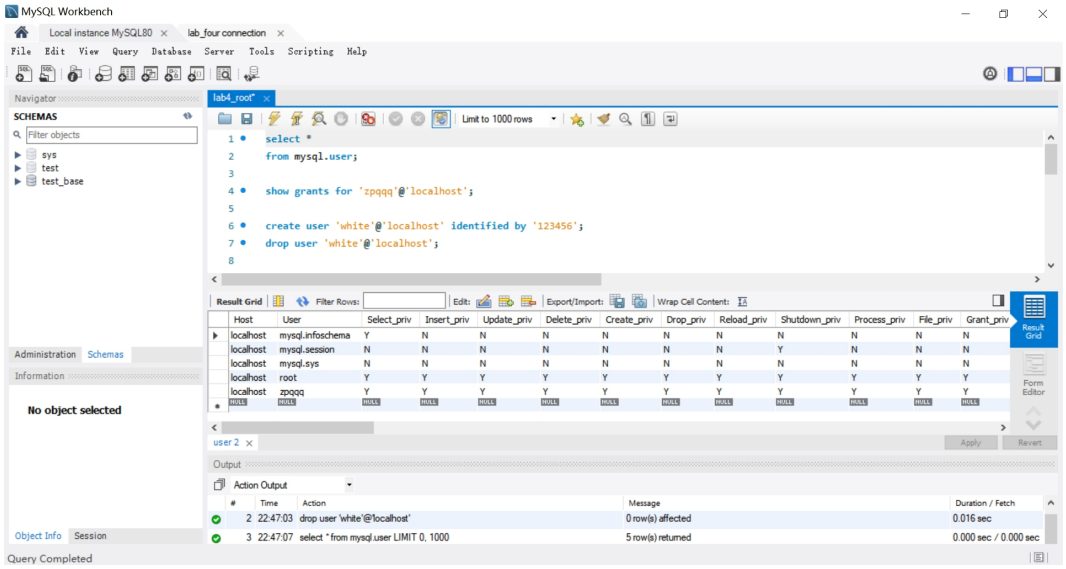

图 3.1.1 查看用户

然后通过语句创建一个名为 white 的普通用户。创建后查看用户，可以看见已经多出了一个名为 white 的普通用户，如图 3.1.2。

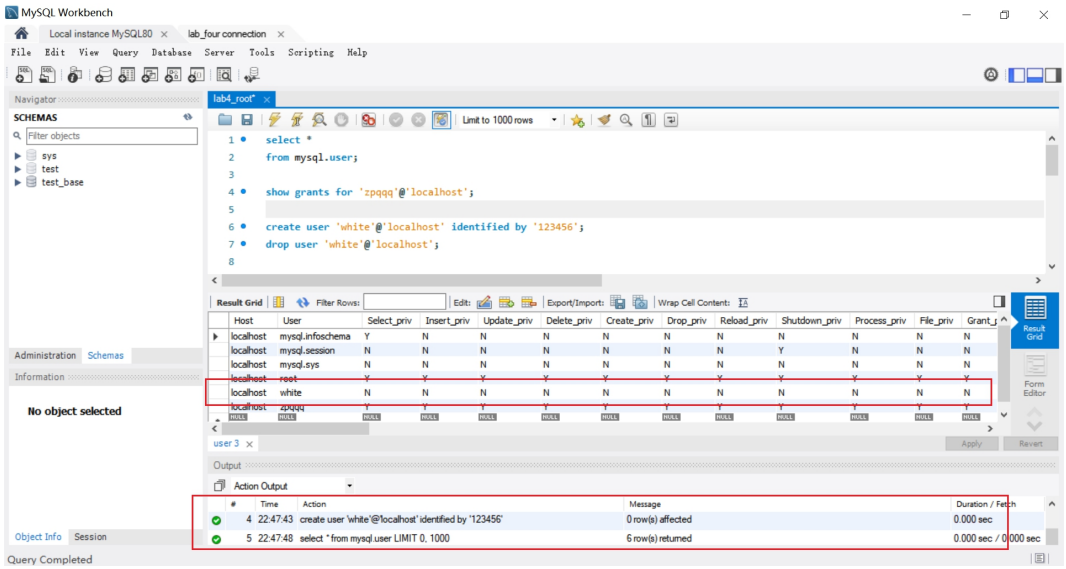

图 3.1.2 创建用户

然后登录该用户，只需要在主页点击加号，输入用户名与密码即可。

2. 考察普通用户权限

此处实验基本采用上一次实验创建的数据库 test_base，其中有数据表 employee 和 baclub。如图 3.2.1，在普通用户的连接内使用数据库，或者查看数据表，均提示操作被拒绝（denied）。

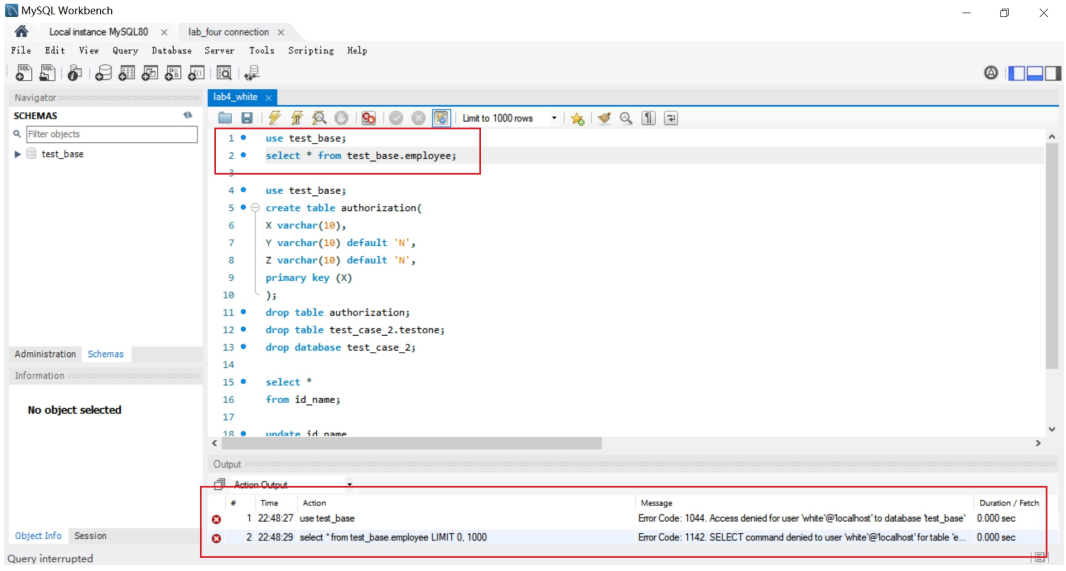

图 3.2.1 操作被拒绝

向 white 用户授予在数据库 test_base 上创建、删除的权限后，查看用户 white 的权限，可见其有了相应的权限记录，如图 3.2.2。

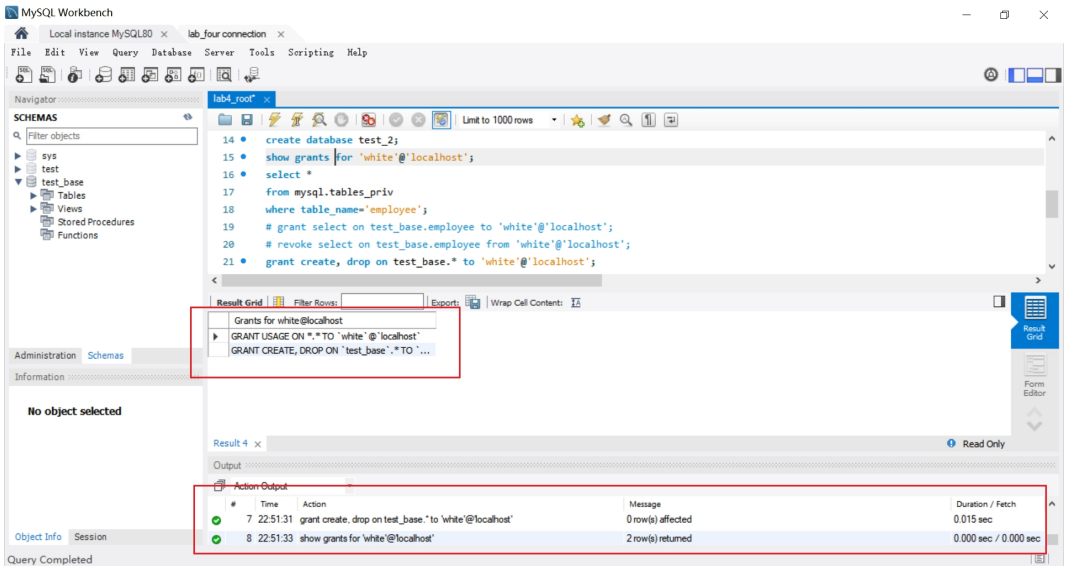

图 3.2.2 查看授权

此时查看表的权限记录，却发现向 white 用户授予在数据库 test_base 上创建、删除的权限后，用户对 test_base 内两个数据表的权限没有记录在此，如图 3.2.3。

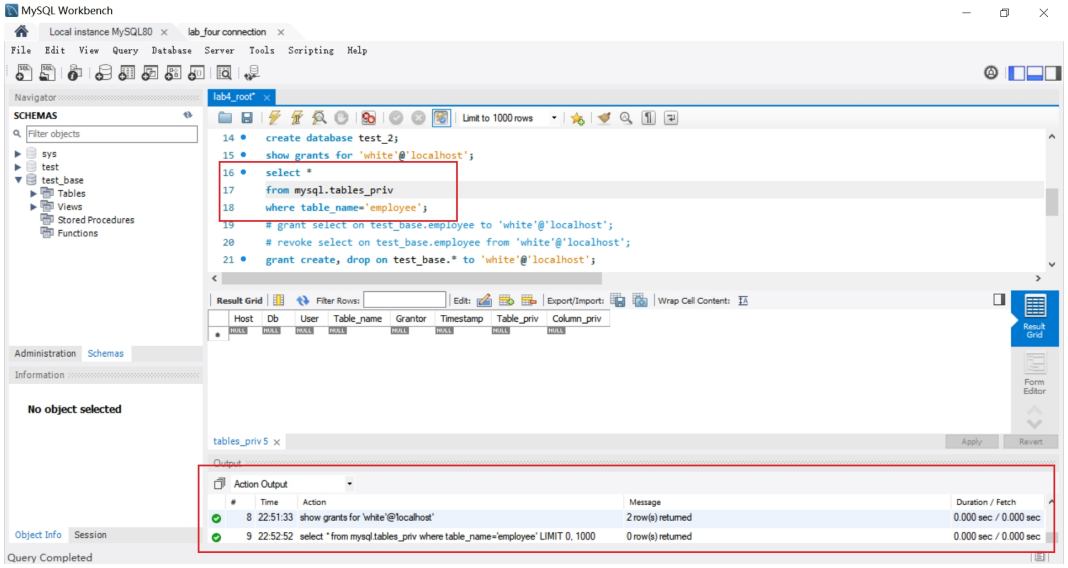

图 3.2.3 表的权限未记录

作为对比，如图 3.2.4，再向用户授予查询某一特定表的权利。这时再查看表的权限记录，可见 mysql.tables_priv 内记录了相应的信息。

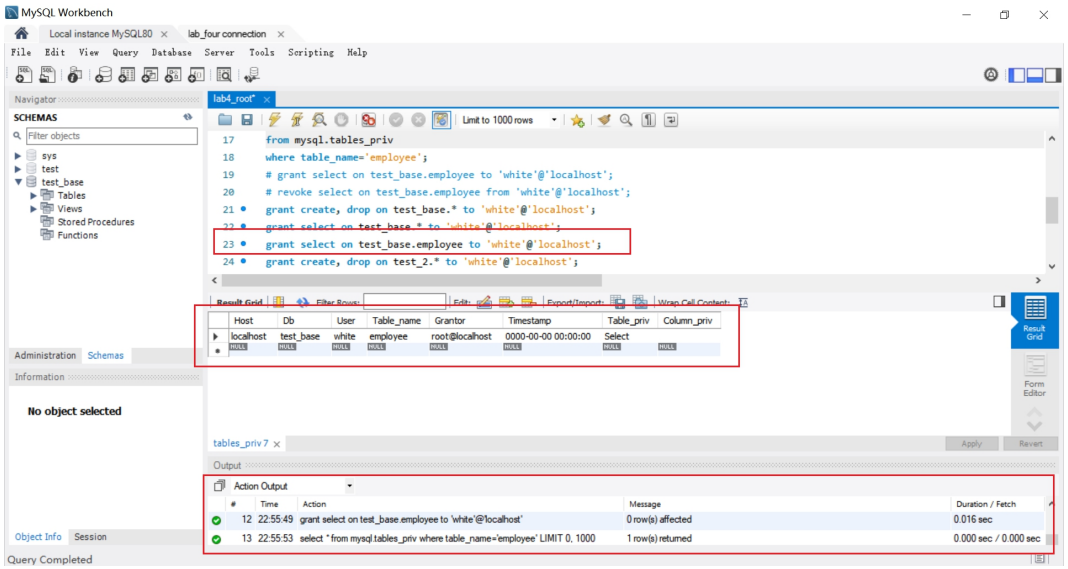

图 3.2.4 授予对表的权利

回到普通用户的连接内，再次执行一开始被拒绝的两条指令，可见两条指令都能被执行了，并且普通用户在被授权后成功查询到了相关数据，如图 3.2.5。

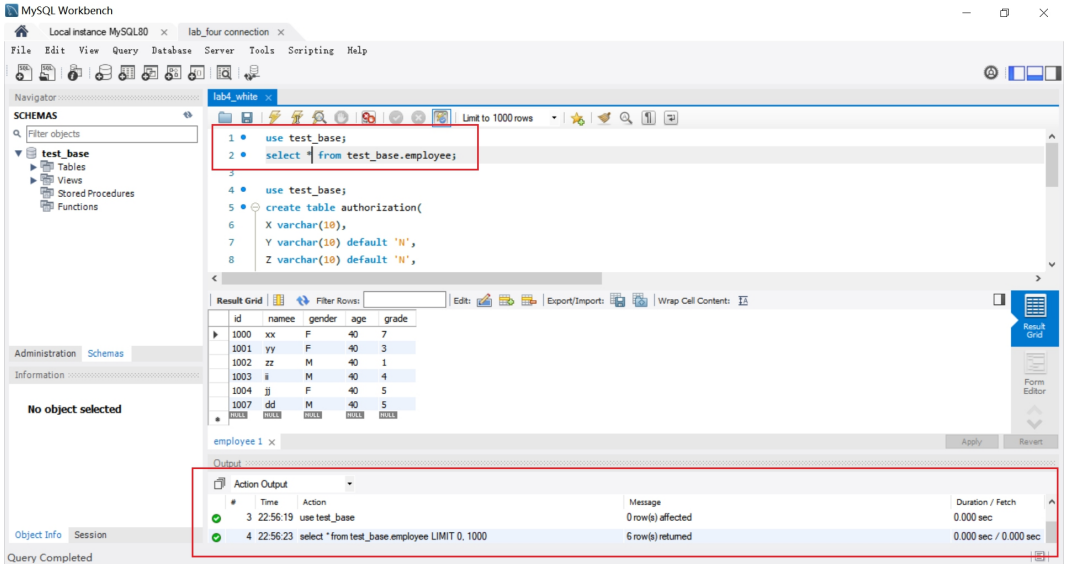

图 3.2.5 授权后查询

回到 root 用户的连接内，收回授予用户 white 的对于数据库 test_base 的权限。此时查询数据表的权限，发现用户 white 对 test_base 内数据表 employee 查询的权限还在，如图 3.2.5。

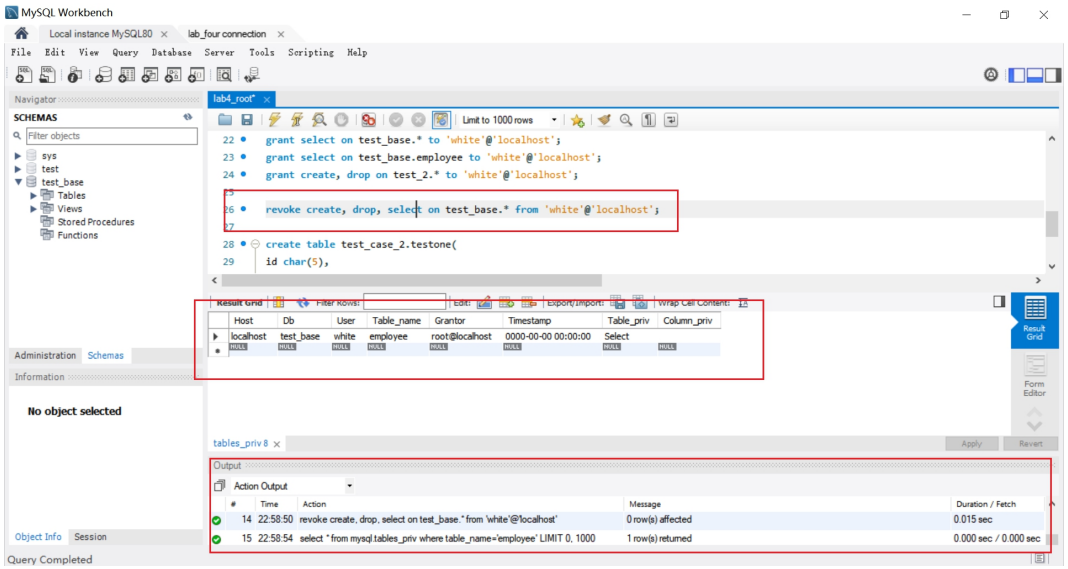

图 3.2.6 撤回对数据库的权限

回到普通用户的连接内，再次执行头两条指令，发现两条指令依然能被执行，如图 3.2.7。

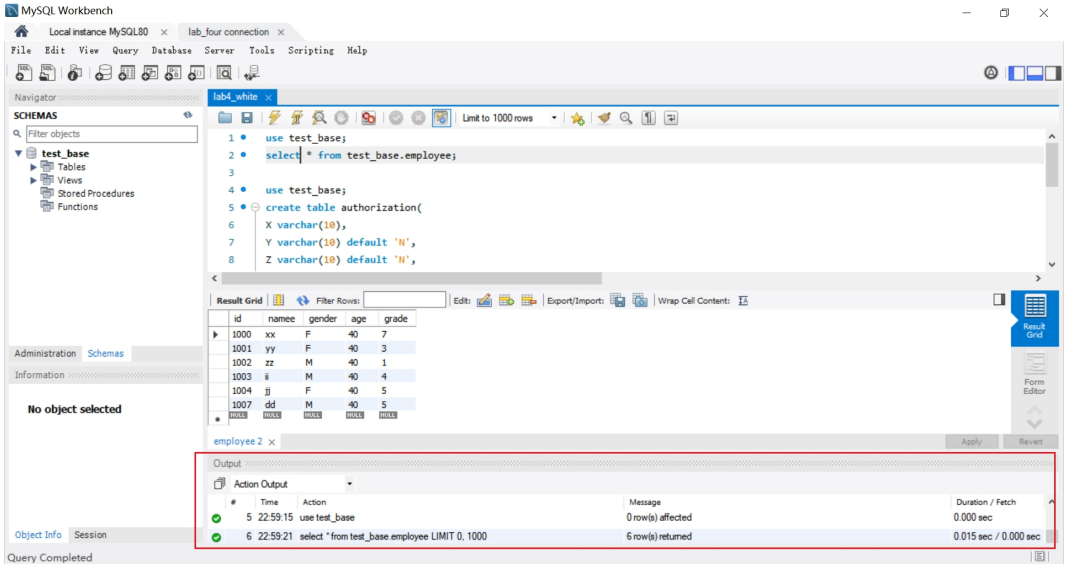

图 3.2.7 未撤回对表的权限

回到 root 用户的连接内，收回授予用户 white 查询数据表的权限，如图 3.2.8。

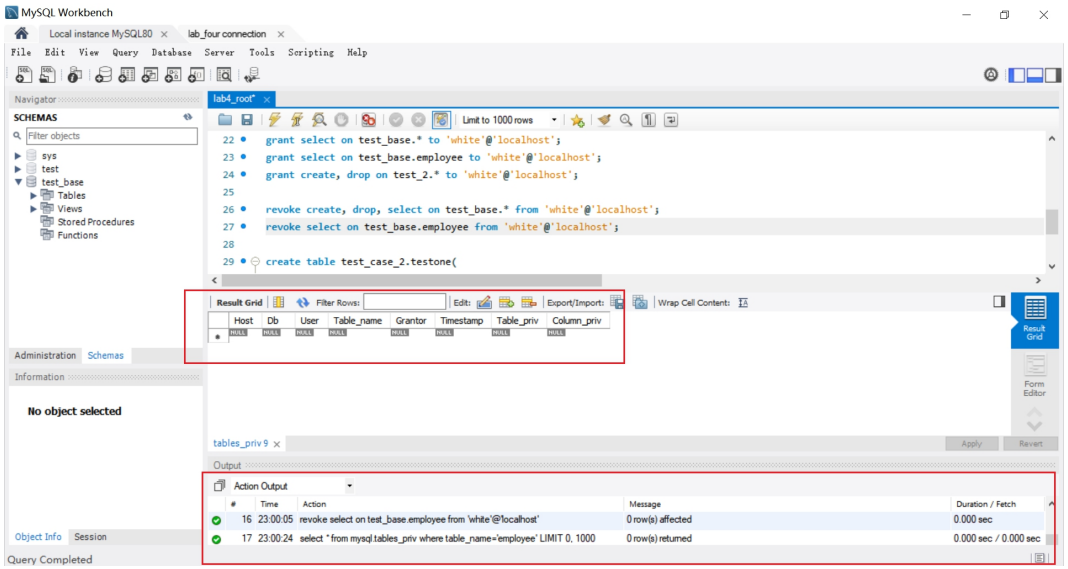

图 3.2.8 撤回查询数据表的权限

回到普通用户的连接内，再次执行头两条指令，发现两条指令被拒绝执行，如图 3.2.9。

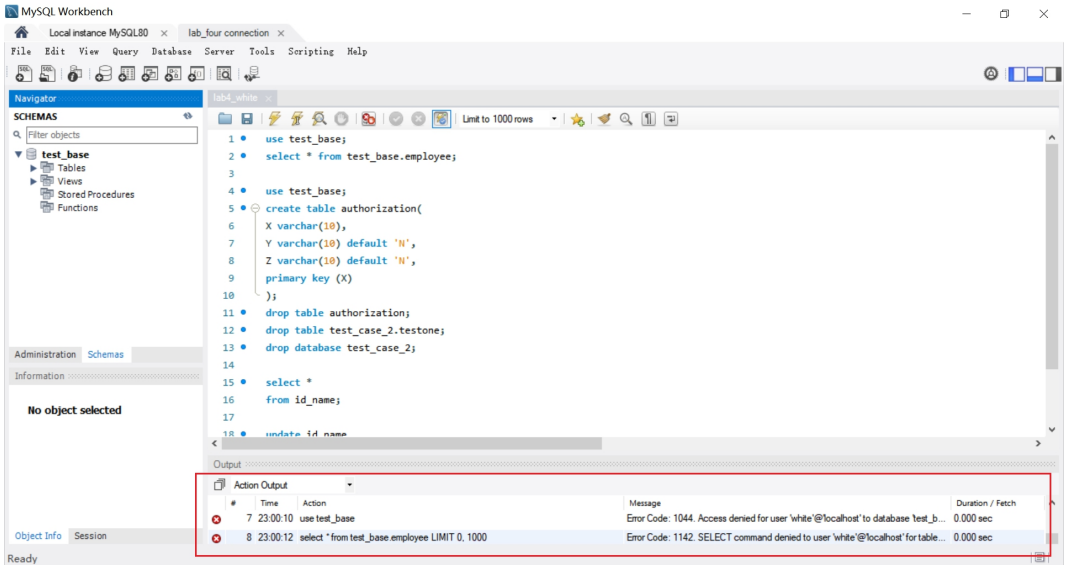

图 3.2.9 指令被拒绝执行

3. 考察表的生成者拥有该表的哪些权限

如图 3.3.1，首先用户 white 授权。

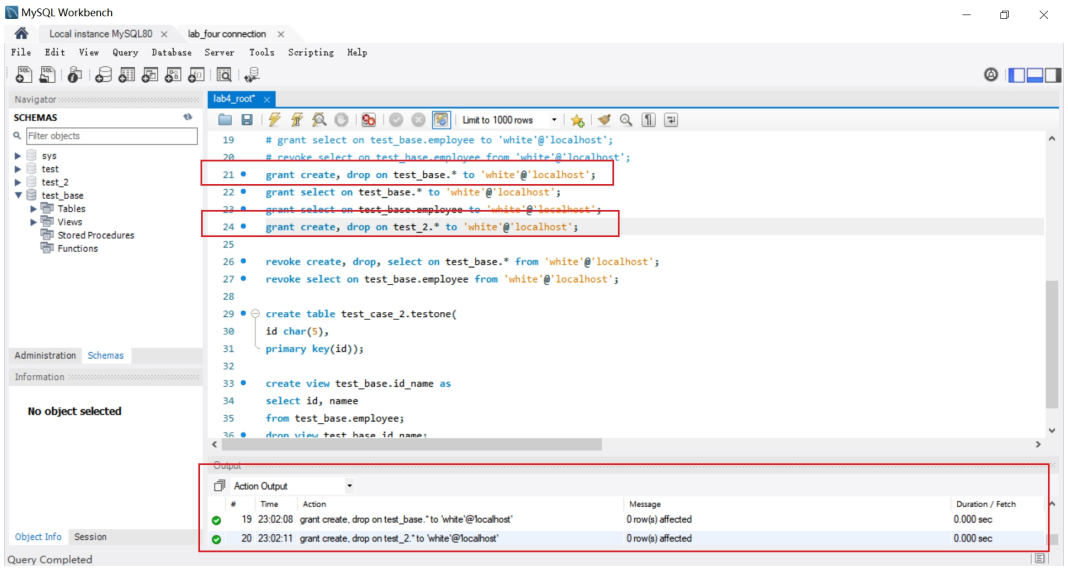

图 3.3.1 授权

此后，在普通用户的连接内创建数据表，如图 3.3.2。

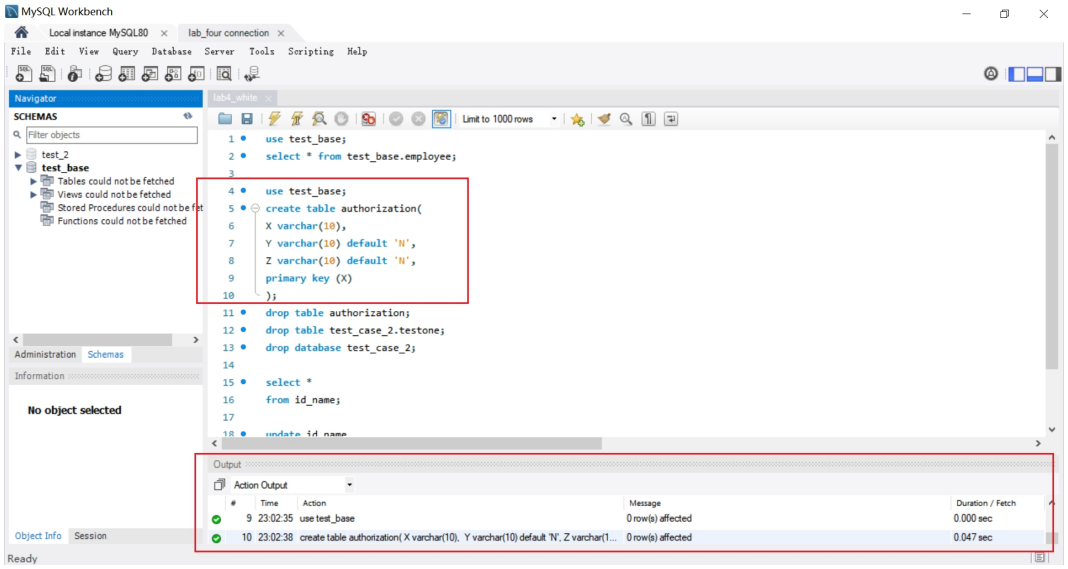

图 3.3.2 创建数据表

回到 root 用户的连接内，刷新后可在左侧看到新创建的数据表，如图 3.3.3。

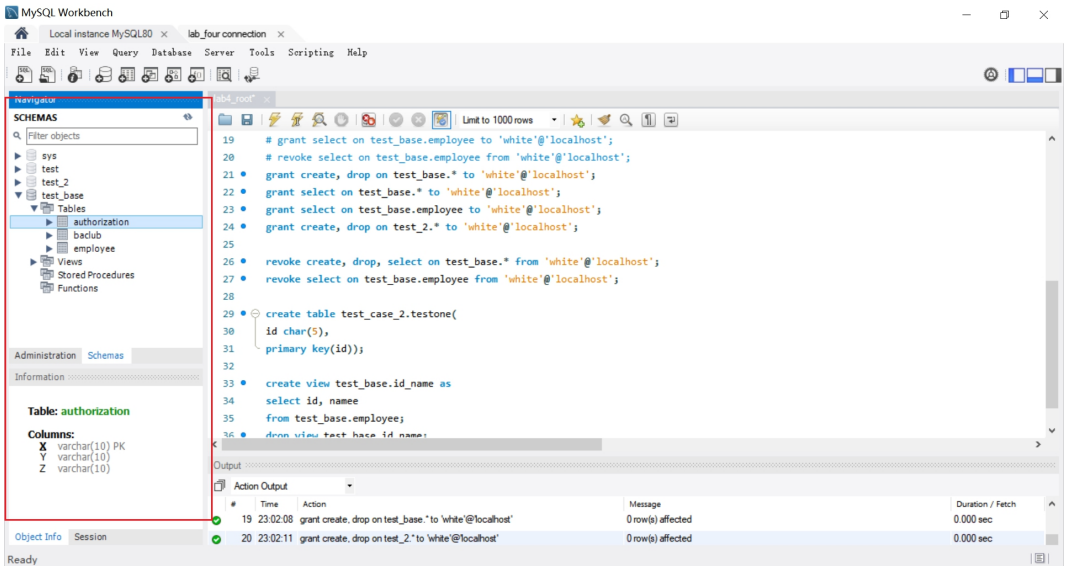

图 3.3.3 查看数据表

此时，虽然用户 white 是表的创建者，但是其没有查询表的权利，如图 3.3.4，系统提示查询被拒绝。事实上，由于授权，用户 white 只有创建和删除的权限，其余权限均由 root 用户保留。这一点也可以通过查询数据表的授权以及用户的权限来确认。

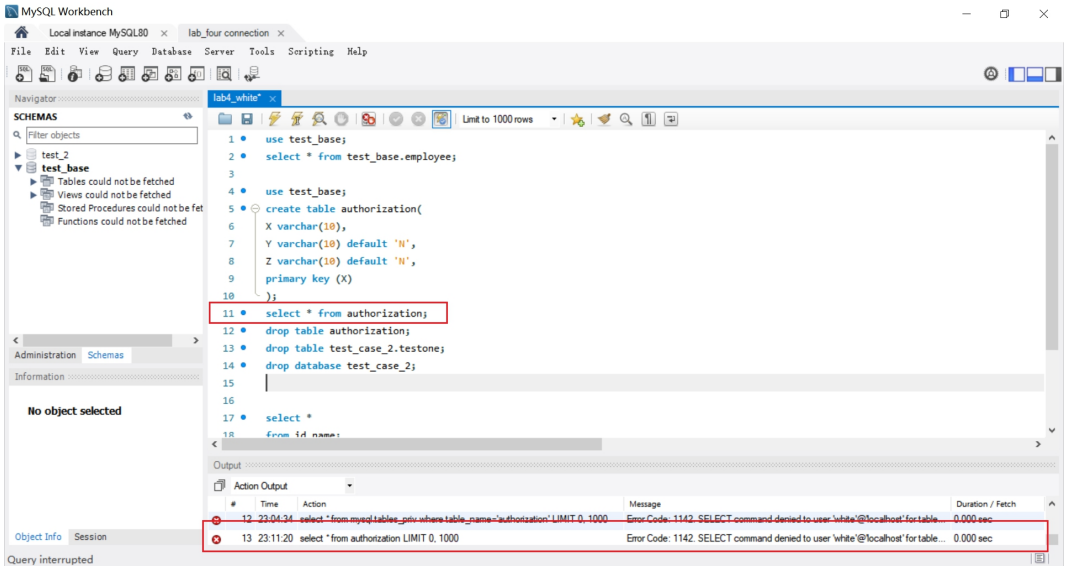

图 3.3.4 查询被拒绝

此外，授权时的“.*”意味着对整个数据库的权限。这一点可以通过删除数据库来确认，如图 3.3.5，用户 white 可以直接删除数据库 test_2。

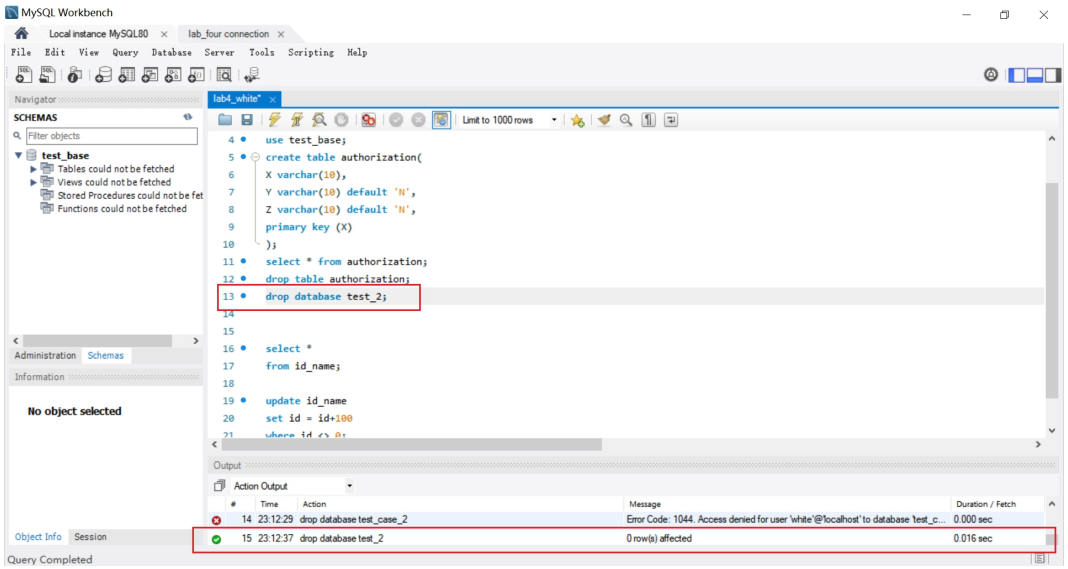

图 3.3.5 删除数据库

4. 视图权限

如图 3.4.1，在数据表 employee 上创建一个关于 id 与姓名的视图。并把查询与更新的权限授予用户 white。

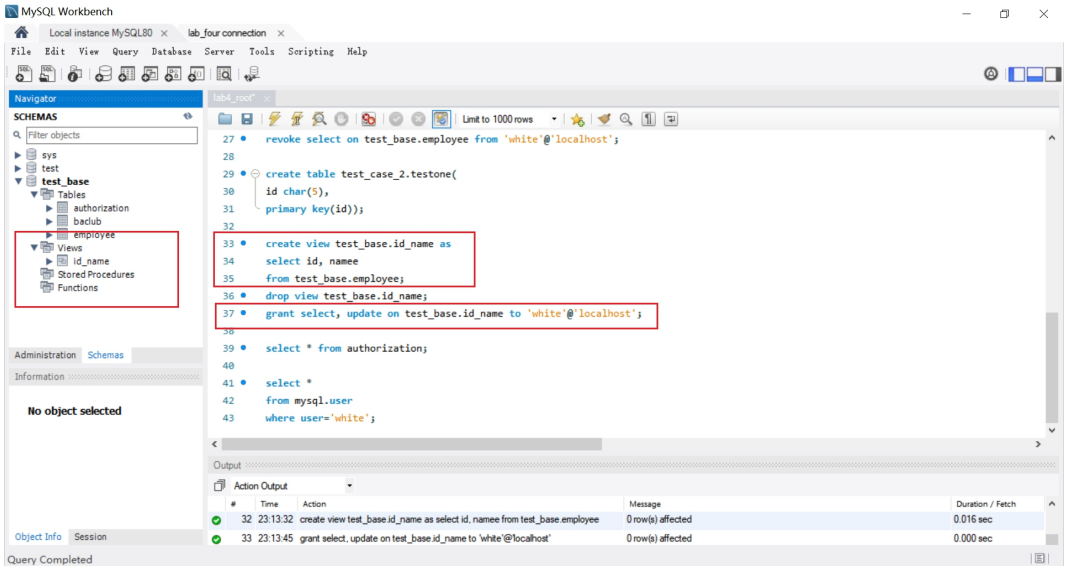

图 3.4.1 创建视图

如图 3.4.2，在普通用户的连接内利用视图进行查询，可见查询能正常查询到结果。

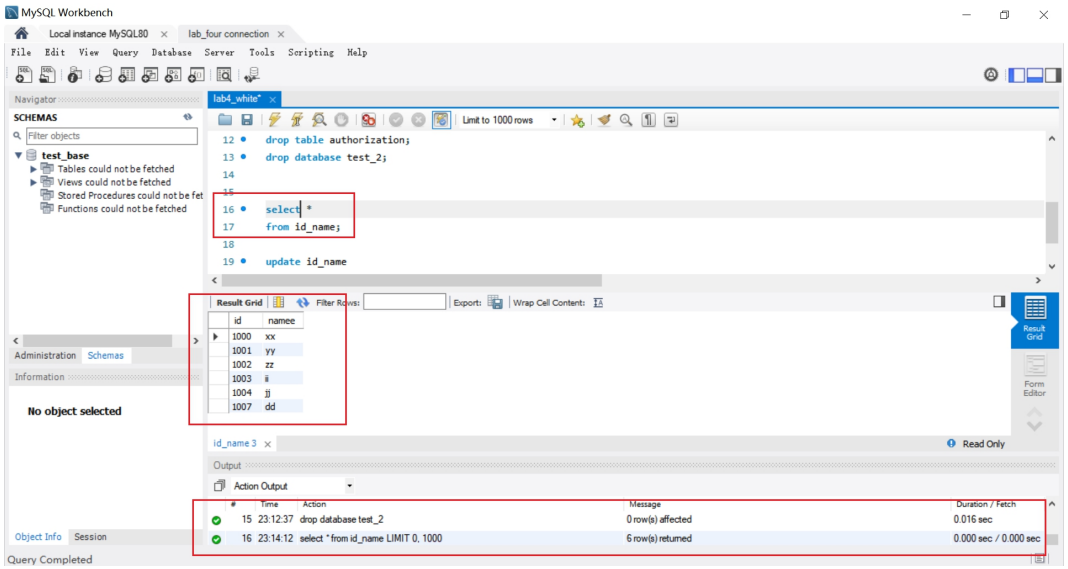

图 3.4.2 视图查询

如图 3.4.3，在普通用户的连接内利用视图进行数据更新，可见数据表的数据被正常更新了，id 为 1003 的行，id 变成了 1103。

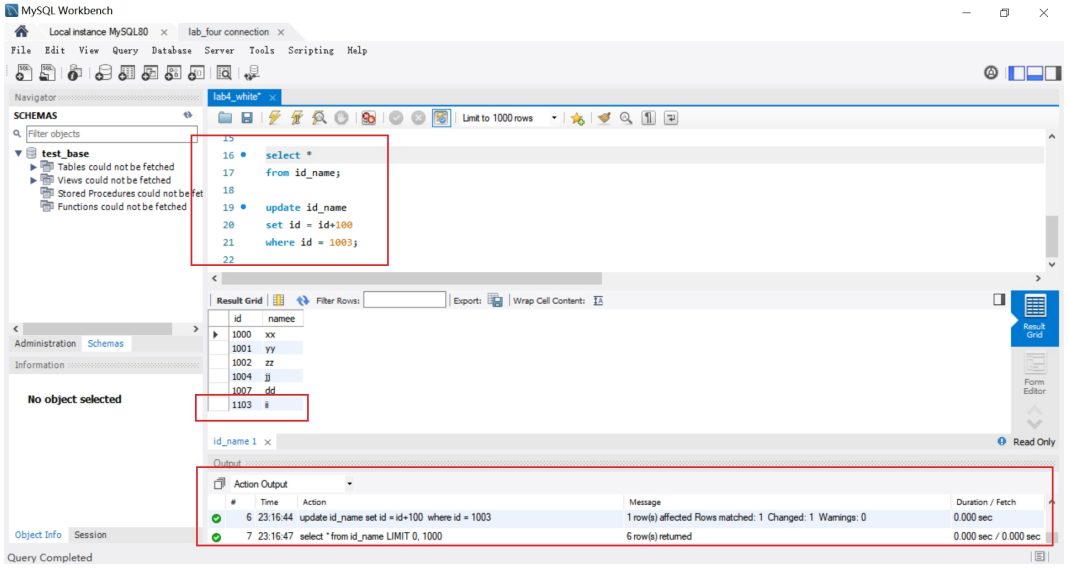

图 3.4.3 视图更新

四、 遇到的问题及解决方法

1. 关于数据库与数据表的授权

通过上述实验，可以看出 MySQL 应该是将对表的授权以及对数据库的授权分开记录。当我撤回了对数据库的权限后，普通用户依然可以对数据表进行查询。

五、 总结

此次实验让我了解了数据库权限的设计，这种设计极大地方便了数据库的管理。