



链滴

一次性删除 Mysql 数据库中所有表的数据， 保留表结构

作者: [liqing455](#)

原文链接: <https://ld246.com/article/1567075196606>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

1、前言

常见的删除数据库表中数据的方法是通过delete或者truncate的方法进行删除操作，如果删除的是表某一条或者部分数据的话适合用delete操作进行删除，如果要删除表中所有的数据的话，适合是用truncate进行删除操作。

那么问题来了，如果某一个数据库中有很多张表，此时我想将该数据库中所有表中的数据全部删掉，如何操作呢？

解决这个问题主要分两种情况，如果不需要保留数据库中所有表的结构，那么答案很简单，执行命令dro database 数据库名 即可达到目的。但是如果需要保留该数据库中所有表的结构，只想删除所有表的数据，又该怎么解决呢？有人会说可以多执行几次truncate操作就OK啦，没错，多执行几次truncate确实可以达到目的，但是，如果要删除的数据库中有很多张表，几十张上百张表，执行上百次truncate操作显然不是好的办法？至此，就是本文要说讲到的办法了。

2、删除方法

删除的办法其实还是执行truncate方法，只是不需要每次手动的输入truncate命令进行删除。通过sql命令的方式生成所有的truncate语句并写入到.sql脚本文件中，然后执行脚本即可完成删除操作，并且留了表结构。

生成truncate命令的sql语句为：

```
SELECT CONCAT('TRUNCATE TABLE ',TABLE_NAME,';') FROM information_schema.TABLES WHERE TABLE_SCHEMA='test' into outfile '/tmp/truncate_test.sql';
```

然后将生成的.sql脚本拷贝到当前文件夹下面：

```
mv /tmp/truncate_test.sql $current_dir/
```

然后执行.sql脚本将数据库中所有表中数据删除：

```
source $current_dir/truncate_test.sql
```

3、说明：

在进行select...into outfile.....操作时，默认只能将文件写入到tmp路径下，可以不用将tmp文件夹下的.sql脚本移动到当前文件夹下，直接在tmp路径下执行.sql脚本即可。如果要写入到其他路径下，需给mysql的守护进程赋写操作的权限，最简单的办法是将/etc/selinux路径下的config配置文件中的SELINUX修改成disabled可以实现写入其他路径下，这里不展开说明。

4、运行效果

结合上一博文mysql数据库的备份和恢复，下面贴出一个小脚本的源码和运行截图。使用该脚本进行数据备份和恢复时，源和目的数据库中的所有表结构需要一致，否则执行可能出错。该脚本主要实现mysql数据库的备份和恢复，以及一次删除整个数据库中所有表数据并保留数据表结构。

运行结果截图说明：

```
#####
Quick Backup and Recovery
Source and Destination Mysql Must Be Same
#####
1) Back up database test
2) Recovery database testnet/li_qinging
3) Clear data in database test
q) Quit
Your Option:█
```

1. 备份test数据库中的所有数据
2. 恢复数据
3. 删除数据库test中所有表中的所有数据，并保留表结构

```
#####
Quick Backup and Recovery
Source and Destination Mysql Must Be Same
#####
1) Back up database test
2) Recovery database test
3) Clear data in database test
q) Quit
Your Option:1
Database test already backup...
#####
Quick Backup and Recovery
Source and Destination Mysql Must Be Same
#####
1) Back up database test
2) Recovery database test
3) Clear data in database test
q) Quit
Your Option:3
Clear test of admin successful...
#####
Quick Backup and Recovery
Source and Destination Mysql Must Be Same
#####
1) Back up database test
2) Recovery database test
3) Clear data in database test
q) Quit
Your Option:2
Database test already recovery...
#####
Quick Backup and Recovery
Source and Destination Mysql Must Be Same
#####
1) Back up database test
2) Recovery database test
3) Clear data in database test
q) Quit
Your Option:█
```

5、脚本源码：

```
#!/bin/bash
while [ 1 ]
do
echo " #####"
echo " Quick Backup and Recovery"
echo " Source and Destination Mysql Must Be Same"
echo " #####"
echo " 1) Back up database test"
echo " 2) Recovery database test"
echo " 3) Clear data in database test"
echo " q) Quit"
echo -n " Your Option:"
read option_char
```

```

current_dir=$(pwd)
case ${option_char} in
"1")
    rm -f $current_dir/test_bk.sql
    mysqldump -t -c -uroot -proot test > $current_dir/test_backup.sql
    echo " Database test already backup..."
    ;;
"2")
    mysql -u root --password='root' -e "
use test
source $current_dir/test_backup.sql"
    echo " Database test already recovery..."
    ;;
"3")
    rm -f $current_dir/truncate_test.sql
    chmod 777 $current_dir
    mysql -u root --password='root' -e "
SELECT CONCAT('TRUNCATE TABLE ',TABLE_NAME,';') FROM information_schema.TAB
ES WHERE TABLE_SCHEMA='test' into outfile '/tmp/truncate_test.sql';"
    mv /tmp/truncate_test.sql $current_dir/
    mysql -u root --password='root' -e "
use test
source $current_dir/truncate_test.sql"
    echo " Clear data of test successful..."
    ;;
"q"|"quit"|"exit"|"Q"|"QUIT"|"Quit")
    break
    ;;
*)
    echo "your option is invalid, please input again..."
    ;;
esac
done

```