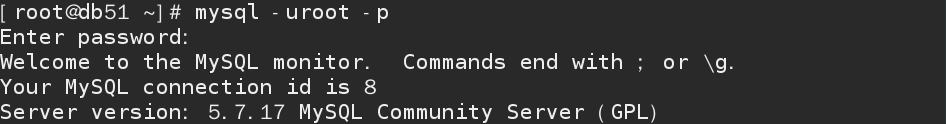
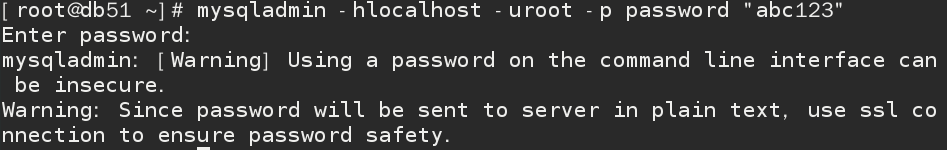
**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*用户授权和权限撤销**

**1.数据库管理员密码的设置**

**1）修改数据库管理员本机登陆密码**

**]# mysqladmin -hlocalhost -uroot -p password “abc123”**

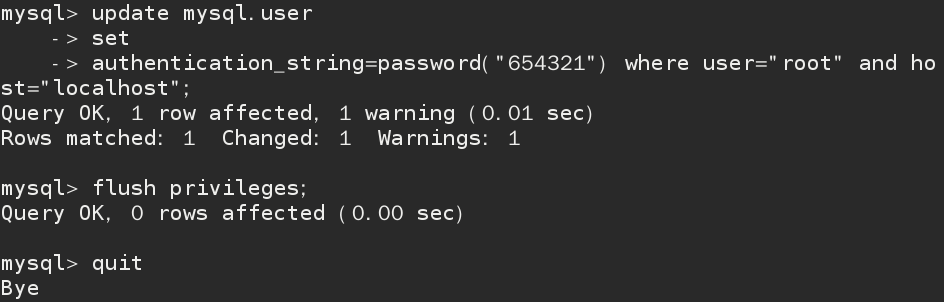
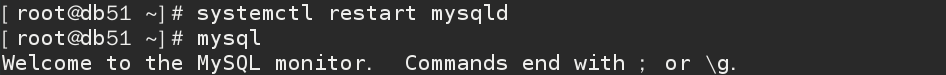
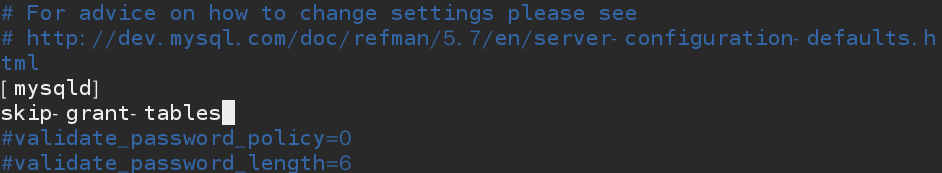


**输入旧密码**

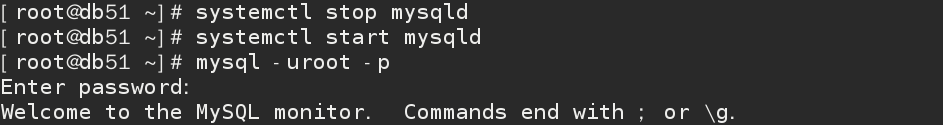
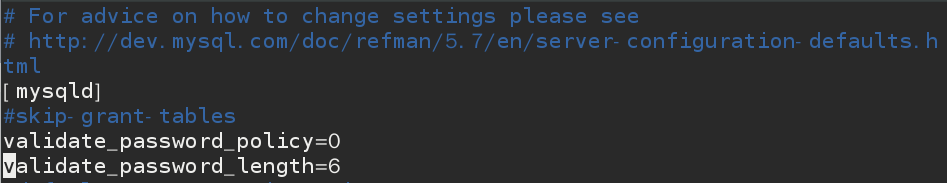
**输入密码：abc123**

**2）恢复数据库管理员本机登陆密码**



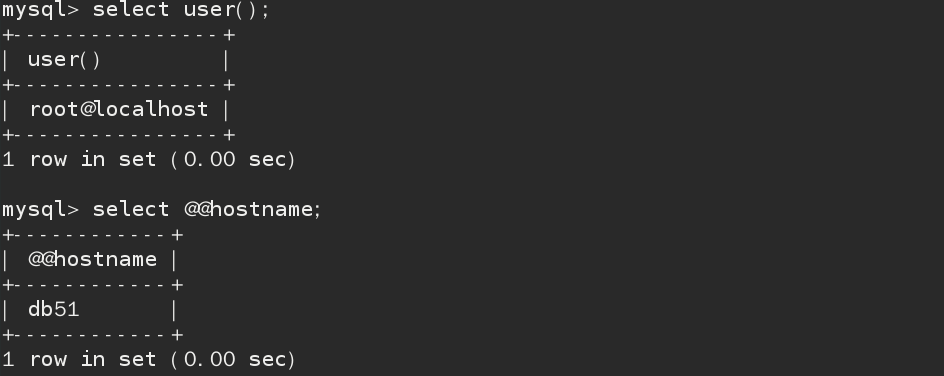






**输入密码：654321**

**3) 查看服务器上已有的授权用户及访问权限**



**查看服务器上已有的授权用户**

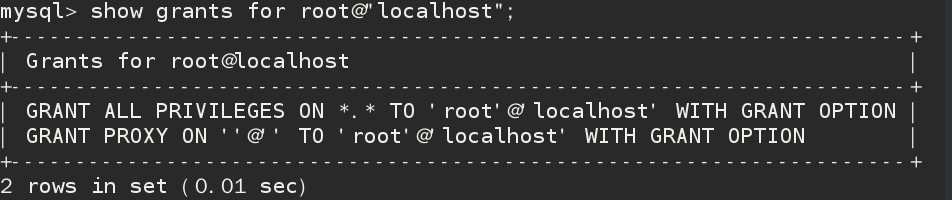
**登陆用户查看自己的访问权限**

**数据库服务器的主机名**

**显示当前登陆的用户信息**

**4）管理员用户查看服务器已有授权用户的访问权限**

***show grants for 用户名@“客户端地址”；***



**5）撤销权限**

***revoke 权限列表 on 数据库名 from 用户名@“客户端地址”；***

*::::mysql> revoke grant option on bbsdb.\* from webadmin@"%";*

**6）删除授权用户**

***drop user 用户名@“客户端地址”；***

**7）修改授权用户的登陆密码**

***set password for 用户名@“客户端地址”=password(“新密码”)；***

**8）授权用户重置自己的登陆密码**

***set password=password(“密码”)；***

***9） 授权信息记录在mysql库下的不同表里***

***use mysql;***

***show tables;***

***columns\_priv 表：记录授权用户对表中字段的访问权限***

***tables\_priv 表： 记录授权用户对库中表的访问权限***

***desc mysql.tables\_priv;***

***select \* from mysql.tables\_priv;***

***select \* from mysql.tables\_priv***

***db表： 记录授权用户服务器上库的访问权限***

***desc db;***

***select \* from db;***

***user表：记录服务器上有哪些授权用户***

**2.用户授权:在数据库服务器上添加新的连接用户**

***grant 权限列表 on 库名 to 用户名@“客户端地址” identified by “密码” with grant option;***

***----------------------------------------------------------------------------------------------***

***grant all on \*.\* to [admin@”192.168.4.52”](mailto:admin@\”192.168.4.254\”) identified by “123456” with grant option;***

***grant select,insert on gamedb.\* to webuser@”%” identified by “123456”;***

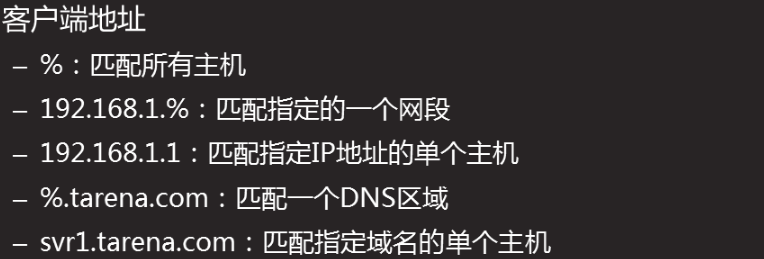
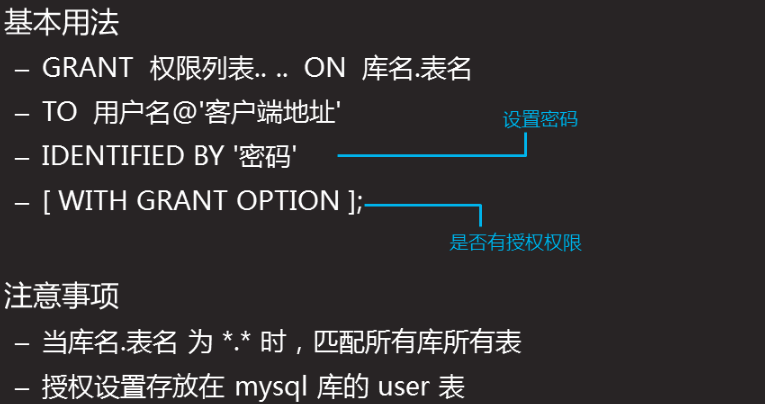
***select user,host from mysql.user;***

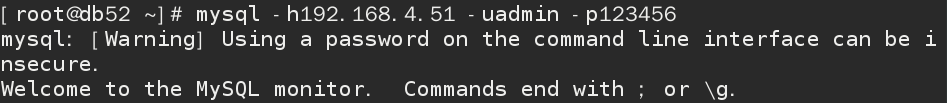
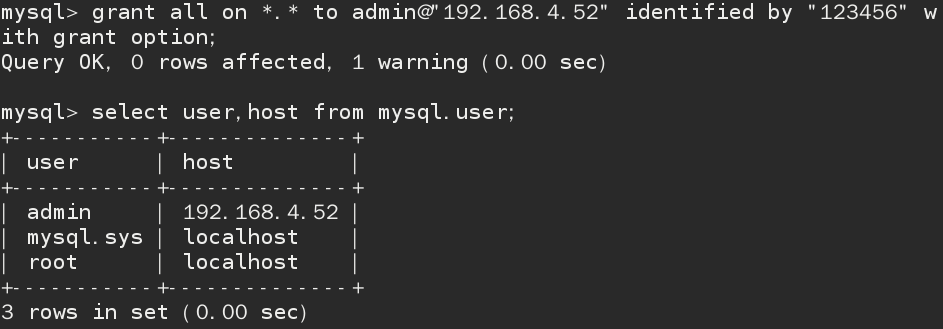
***grant all on \*.\* to root@”%” identified by “123456”;***

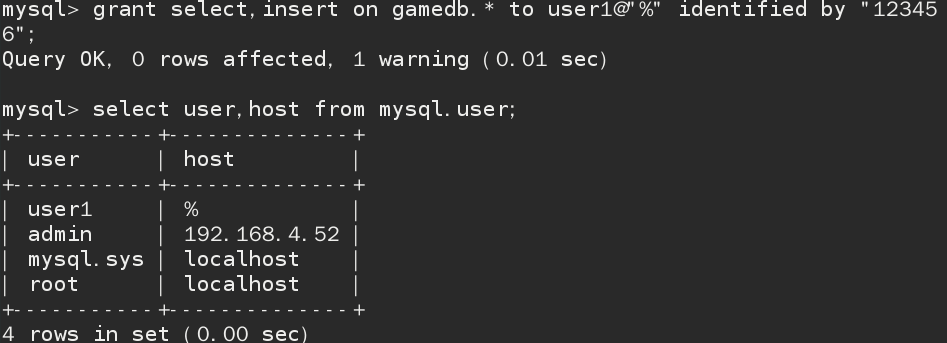
***(在客户端使用授权用户连接数据库)***

***mysql -h192.168.4.51 -uadmin -p123456***

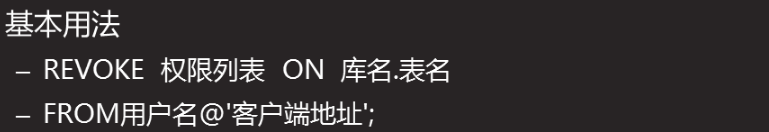
***-----------------------------------------------------------------------------------------------***

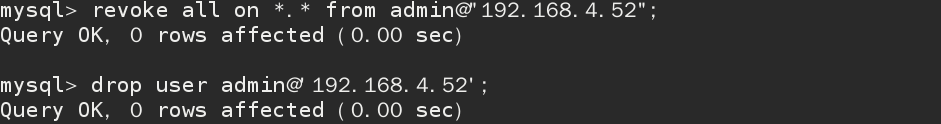




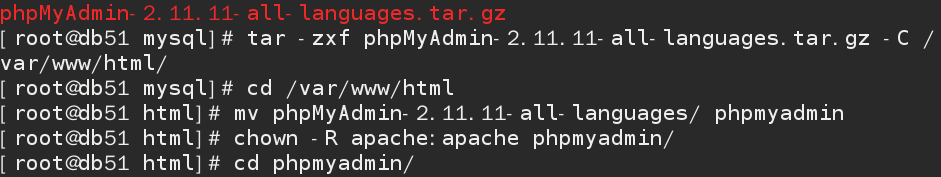
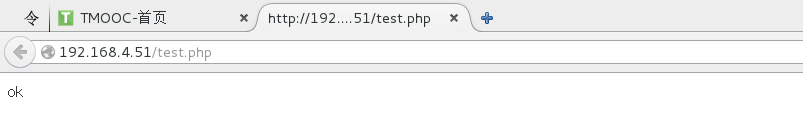
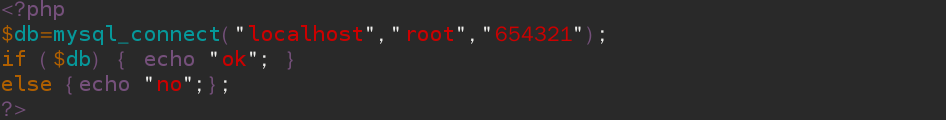
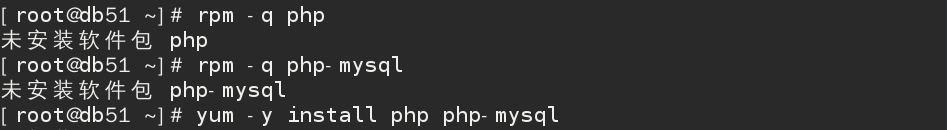


**权限撤销**

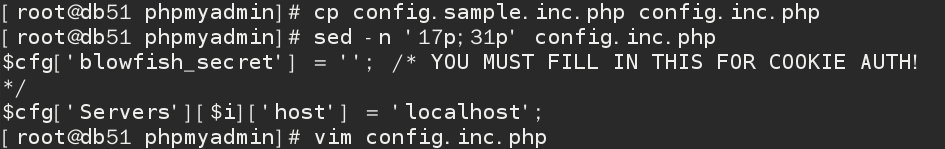


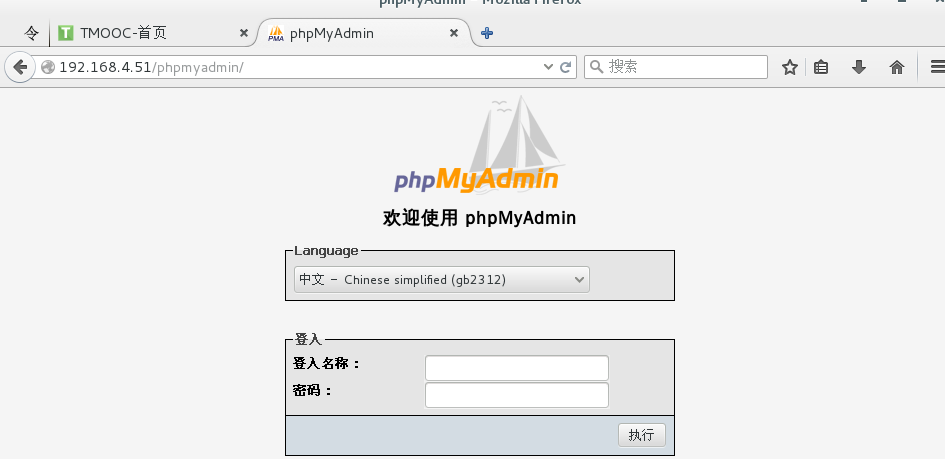
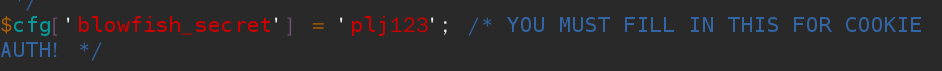


**安装数据库图形管理工具-phpmyadmin**

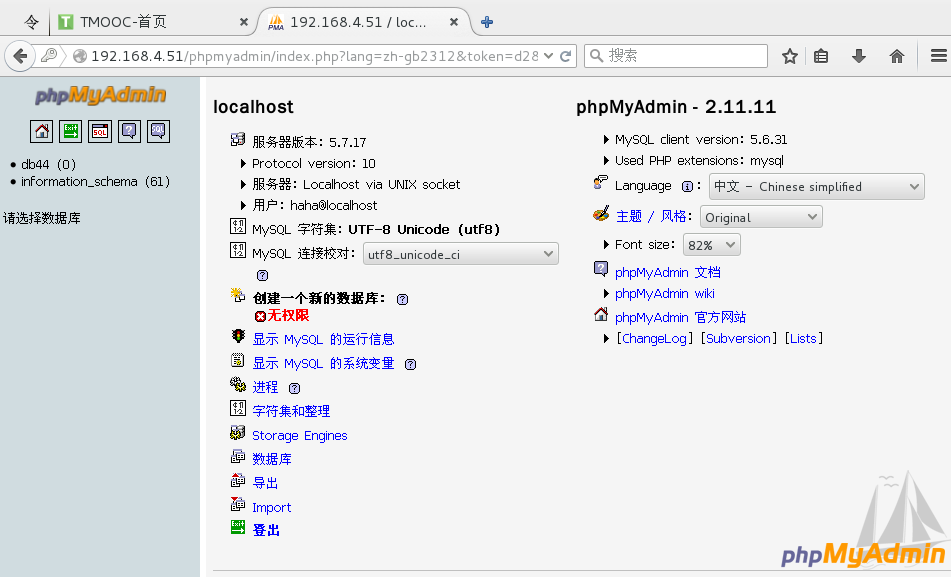
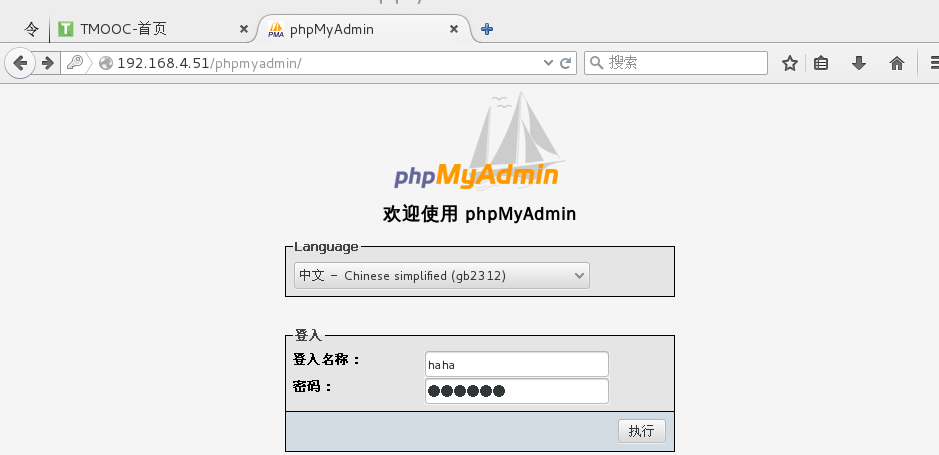
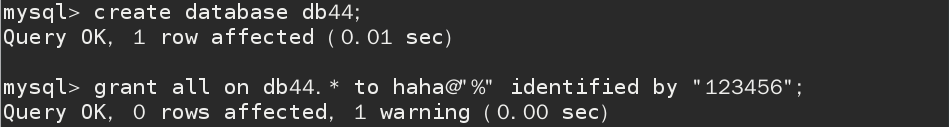
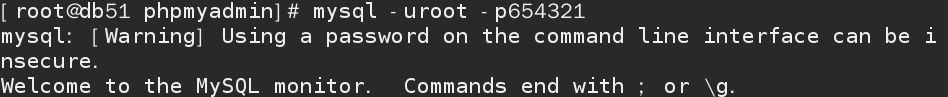


**测试是否可以连通数据库、php服务**





(生产环境下，可以为开发人员创建一个单独的数据库，避免开发人员以root身份登陆)



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**===============================================**

1. **数据备份与恢复**

**1.1备份数据的目的：**

**1.2备份方式：**

**物理备份（直接拷贝库和表对应的文件,适用于数据量小的备份），**

***cp -r /var/lib/mysql/mysql /opt/mysql.bak***

***tar -zcvf /opt/mysql.tar.gz /var/lib/mysql/\****

***rm -rf /var/lib/mysql/mysql***

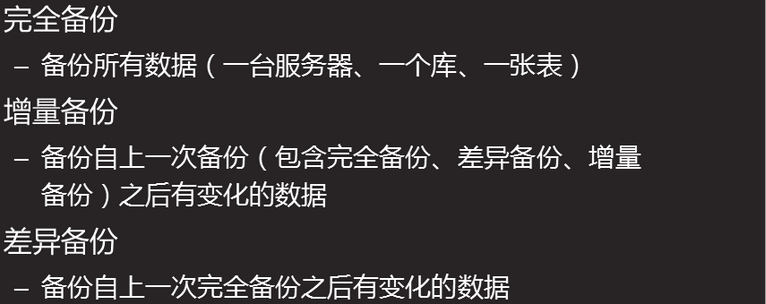
***cp -r /opt/mysql.bak /var/lib/mysql/mysql***

***chown -R mysql:mysql /var/lib/mysql/mysql***

***systemtl restart mysqld***

**逻辑备份（执行备份时根据备份的库或表，生成对应的sql命令，然后把sql命令存储到指定的文件里）**

**1.3备份策略:**



**逻辑备份**

**1.4 备份策略的使用方式**

**完全+差异**

**完全+增量**

**crond 执行备份脚本**

***30 23 \* \* 1 /root/allbak.sh (周一备份所有)***

***50 23 \* \* 2-7 /root/newdata.sh (周二到周日备份新产生的数据)***

**1.5备份数据时要考虑的问题**

**备份策略**

**备份时间**

**数据备份频率**

**备份文件的命名 xxx（有标识性，一般是日期+库名）.sql**

***yyyy-mm-dd-dbname.sql***

**存储空间 可扩展具有容错性（RAID、LV）**

1. **数据备份与恢复-完全备份和恢复**

***#mysqldump -uroot -p654321 数据库名 > 目录名/xxx.sql***

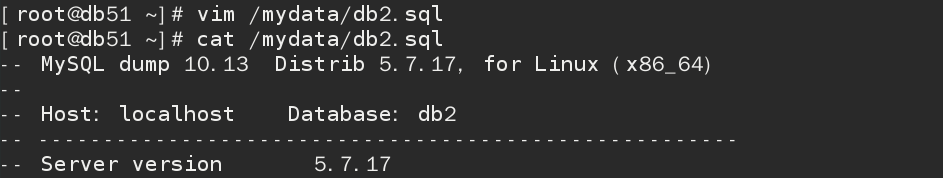
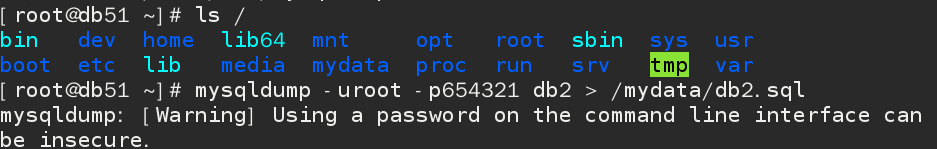
**数据库名的表示方式**

**一台服务器上的所有数据： *--all-databases***

**备份某一个库的所有数据： *库名***

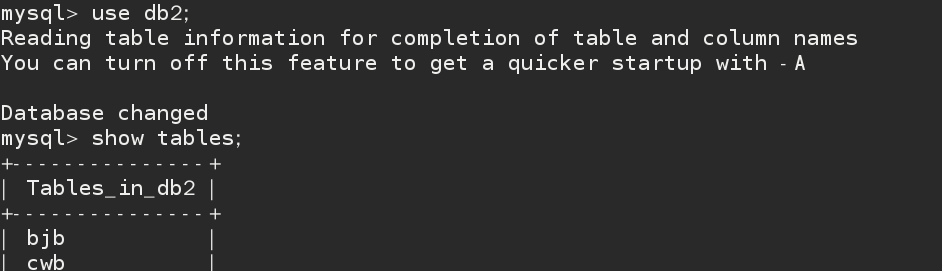
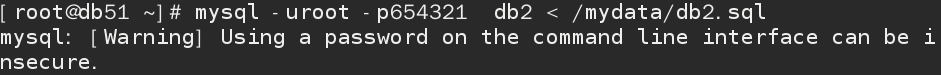
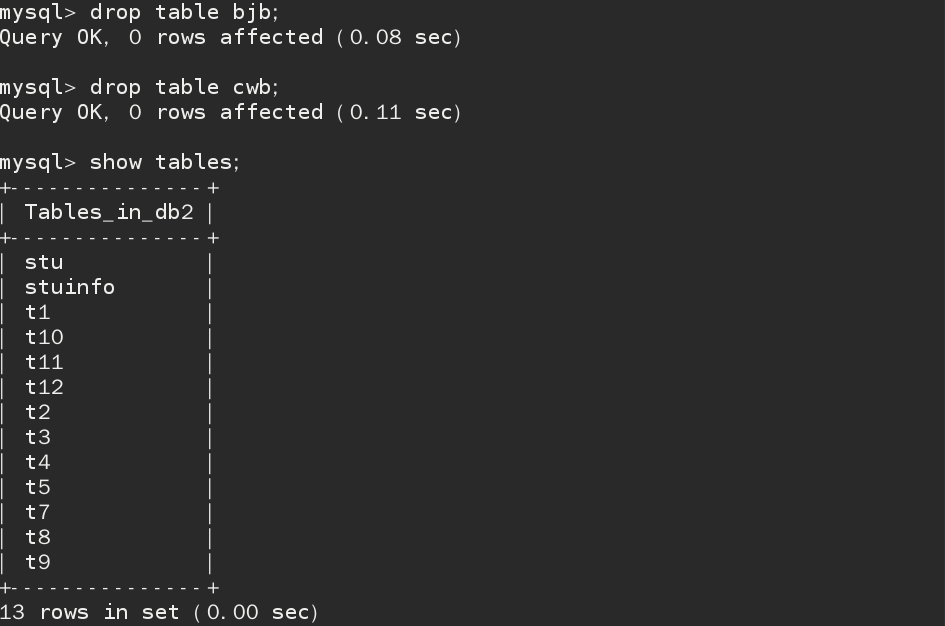
**备份某一张表的所有数据： *库名 表名***

**（少用）把多个库的数据存放到一个文件里： *-B 库名列表***



**恢复**

***]#mysql -uroot -p654321 库名 < xxx.sql***



***vim /root/allbakdb5.sh***

***#！/bin/bash***

*day=$(date +%F)*

*if [ ! –e /mydbbak ] ; then*

*mkdir /mydbbak*

*fi*

*mysqldump -uroot -pabc123 userdb > /mydbbak/$day-userdb.sql*

***]#chmod +x /root/allbakdb5.sh***

***]#crontab -e***

***30 23 \* \* 1 /root/allbakdb5.sh***

**练习：**

***备份服务器上所有数据到/datadir目录名叫all.sql***



***备份服务器上db3库的所有数据到/datadir目录名叫db3.sql***



***备份服务器上db3库下的user的所有数据到/datadir目录名叫db3-user.sql***



***删除db3库下的user表记录 使用备份文件恢复数据***

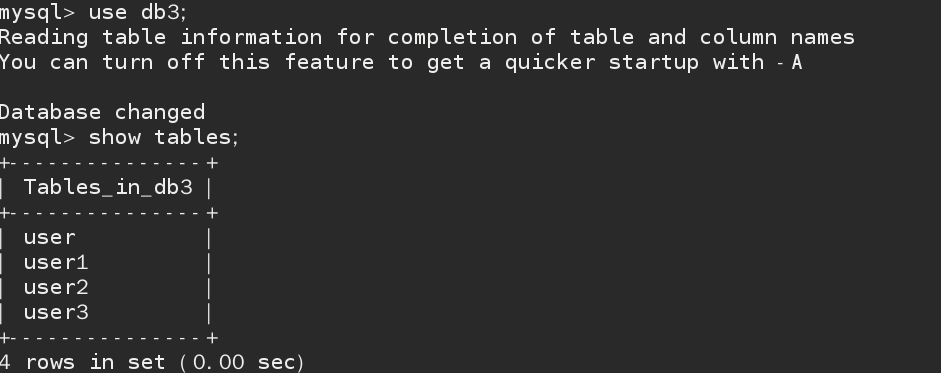
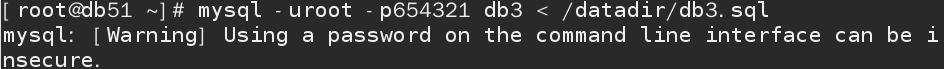
***mysql> delete from db3.user;***



***把db3的所有数据恢复到db55库里***

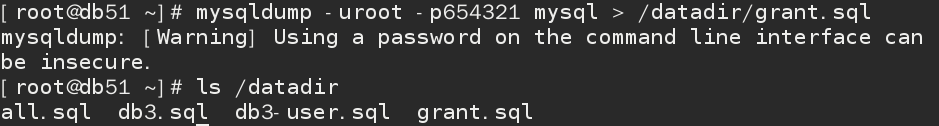


***删除db3库，使用备份文件恢复db3库里的数据***

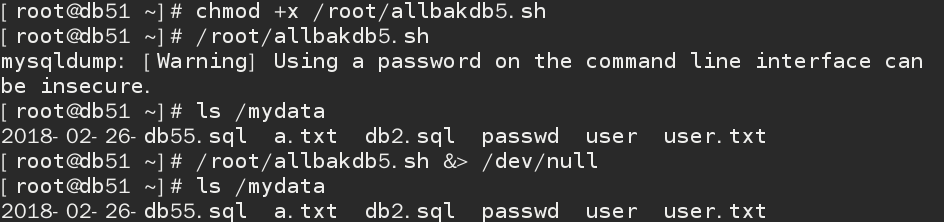


***备份数据库服务的所有授权信息到/datadir/grant.sql文件里***

***#] mysqldump -uroot -p654321 mysql > /datadir/grant.sql***



***编写脚本alldata.sh 功能备份db55库的所有数据到系统的/mydata目录下，备份文件名称要求如下：年-月-日-库名.sql***



**完全备份策略有哪些缺点？**

***备份和恢复时都会锁表（写锁）***

***只能把数据恢复到备份时的状态***

***-----------------------------------------------------------------------------------------------***

**3.数据备份与恢复-增量备份和恢复**

**3.1:启用binlog日志实现数据的增量备份**

**binlog日志的使用：**

1. **binlog日志介绍**

**是mysql数据库服务日志文件类型的一种，又被称为二进制日志。记录除查询之外的sql命令，默认没有启用**

**查询的sql命令包括：select、desc、show**

**非查询的sql命令比如：insert、update、delete、create、drop、grant、revoke**

1. **启用binlog日志**

**如何启用：**

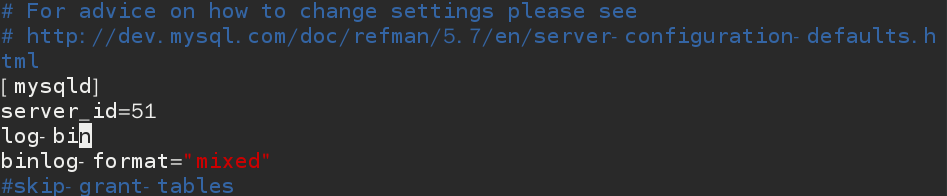
***vim /etc/my.cnf***

***server\_id=51***

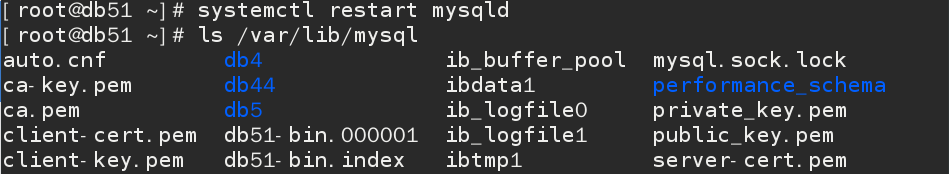
***log-bin***

***binlog-format=”mixed”***

***:wq***



***systemctl restart mysqld***



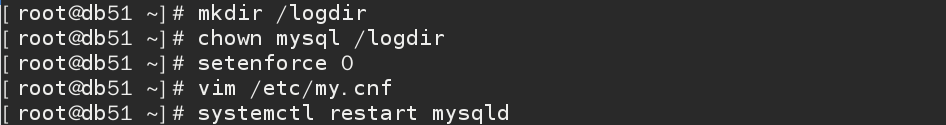
**日志默认存放位置： /var/lib/mysql**

**日志默认名称:**

**主机名-bin.000001 binlog日志文件 > 500M 自动创建第二个**

**主机名-bin.index 索引文件**

**修改日志存储位置和命令**



***mkdir /logdir***

***chown mysql /logdir***

***setenforce 0***

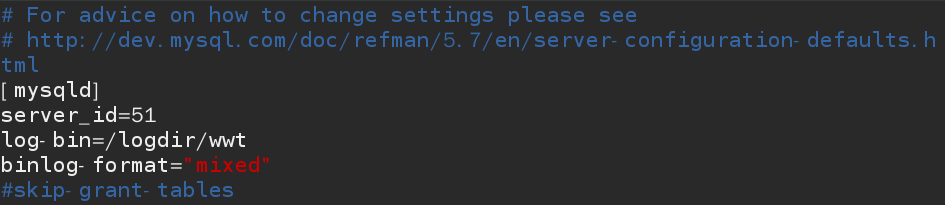
***vim /etc/my.cnf***

***server\_id=51***

***log-bin=/logdir/wwt***

***binlog-format=”mixed”***

***:wq***



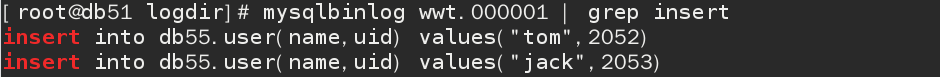
***systemctl restart mysqld***



1. **查看日志文件内容**

***]# mysqlbinlog [选项] binlog日志名***

***]# mysqlbinlog binlog日志名 | grep -i insert***



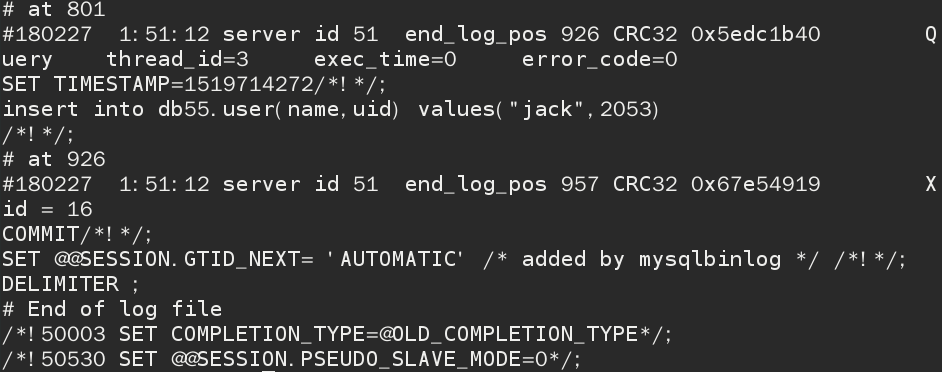
1. **了解日志文件记录sql命令的方式**

**时间点 --start-datetime=”yyyy-mm-dd hh:mm:ss”**

**--stop-datetime=”yyyy-mm-dd hh:mm:ss”**

**偏移量 --start-position=数字**

**--stop-position=数字**

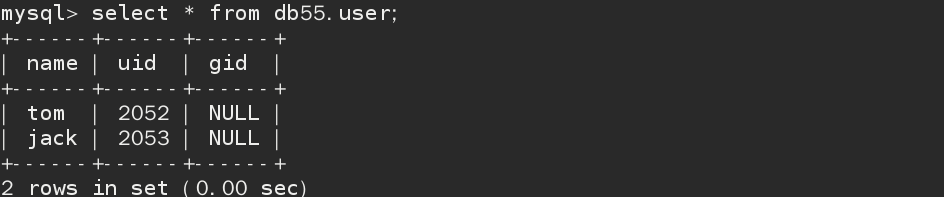
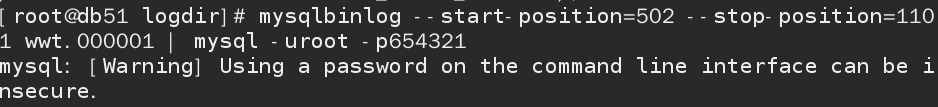
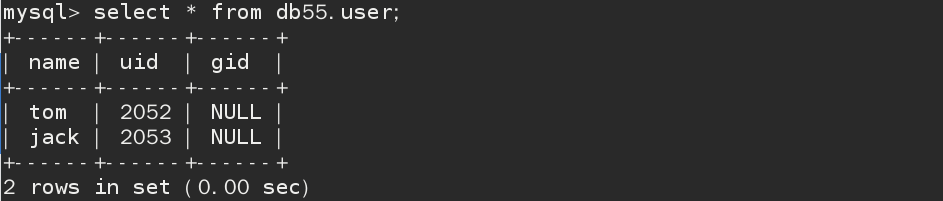
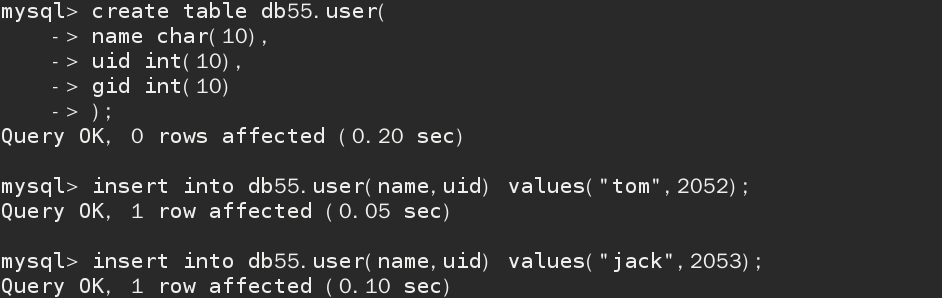


**#时间点**

**#偏移量**

1. **执行日志*文件中的sql命令恢复数据***

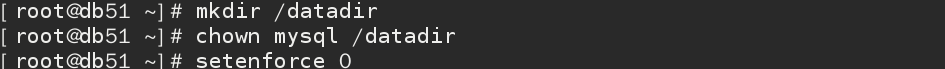
***]# mysqlbinlog [选项] binlog日志名 | mysql -uroot -p654321***



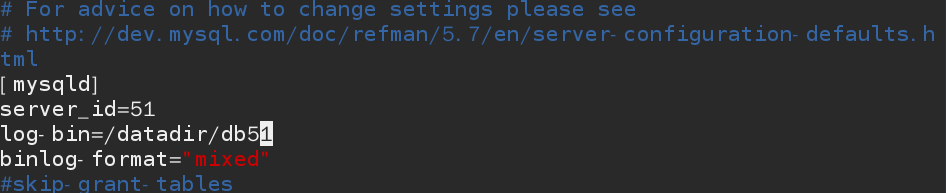
**练习：**

*启用binlog日志要求如下：*

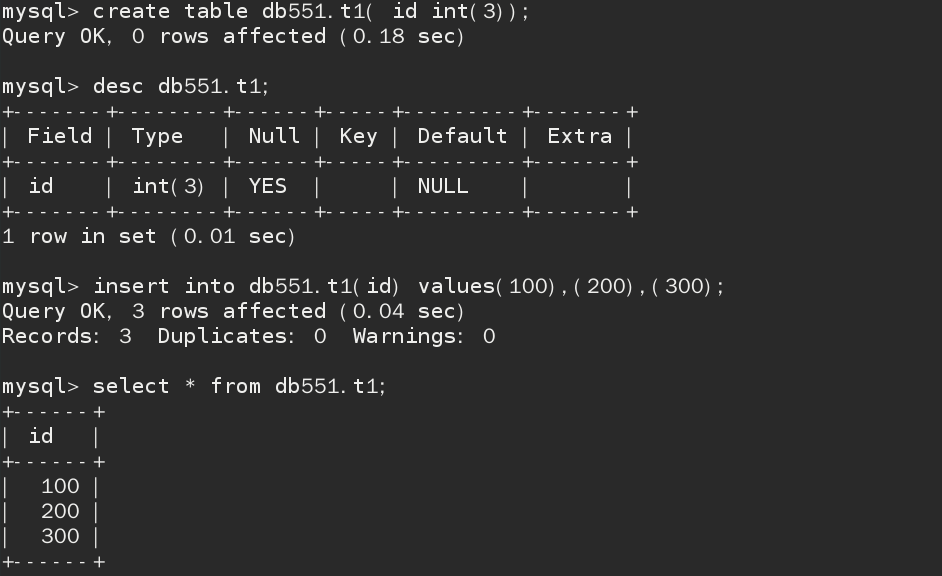
*存储目录/datadir，文件名为db51*



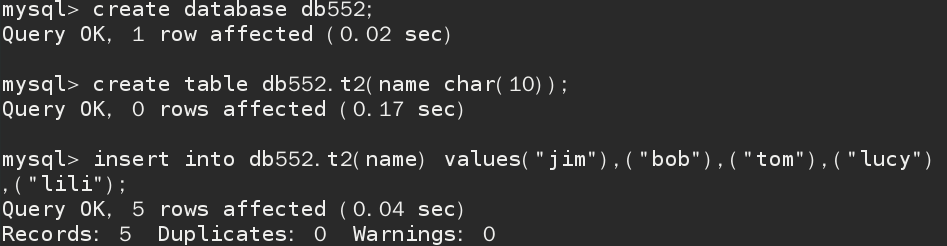




*创建db551库，创建t1表，只有一个名称为id的字段，并插入3条记录值分别是100，200，300*



*创建db552库，创建t2表，只有一个名称为name的字段，并插入5条记录值分别是jim,bob,tom,lucy,lili*



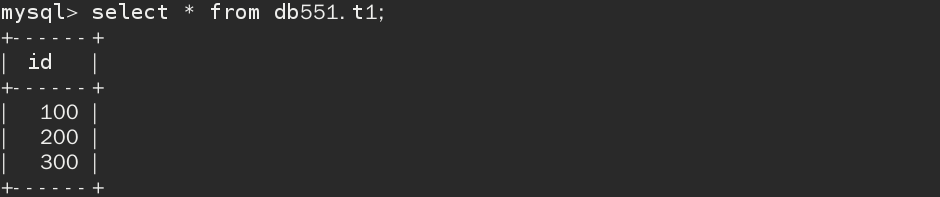
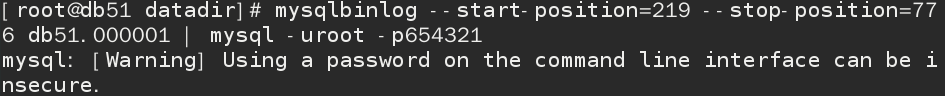
*删除db551库*



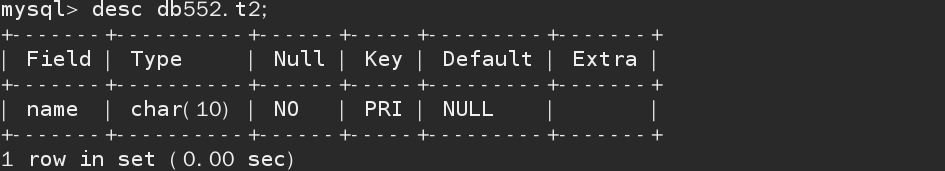
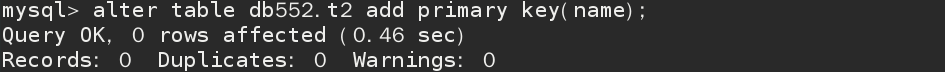
*删除db552库下t2表的后两条记录*



*使用binlog日志恢复db551库的数据*



*使用binlog日志恢复db552库t2表的后两条数据*



*插入数据的时候一条一条的添加，一次性添加的话，恢复的时候会出现重复；*

1. **手动创建新的日志文件（希望某一个日志文件存放某一个数据库操作的sql命令操作过程）**

**默认记录的sql命令是日志文件的容量>500M会自动生成新的日志文件**

**手动创建  *mysql> flush logs;***

**当有多个日志文件，容量没有达到500m时，默认会向编号大的文件里存**

**或：**

***]#mysql -uroot -p654321 -e “flush logs”***

**或：**

***]#mysqldump -uroot -p654321 --flush-logs db5 > /opt/db5.sql***

**或：**

**重启一次数据库服务就会产生一个新的日志文件：*systemctl restart mysqld***

**7）删除日志文件**

**使用sql命令删除**

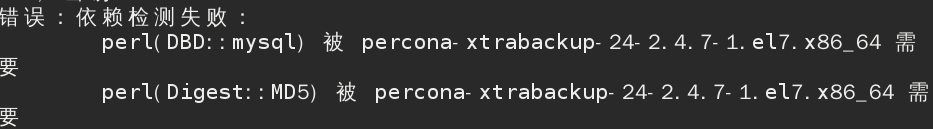
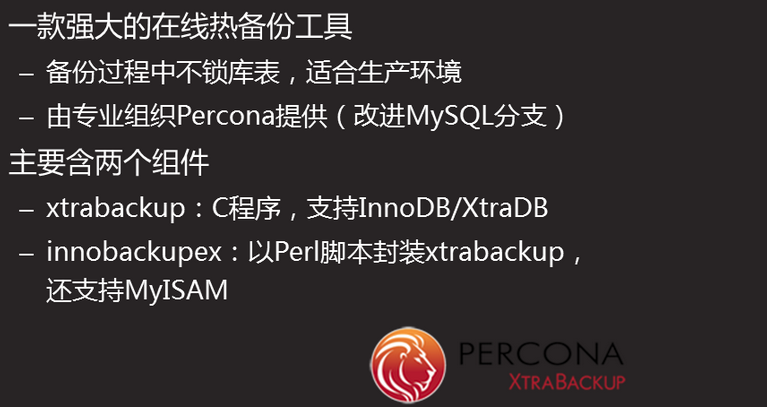
**mysql> reset master; （删除所有，重新创建日志）**

**mysql> purge master logs to “xxx.000005”; (删除000005之前的所有日志文件，同步删除索引文件)**

**使用系统命令删除**

**rm -rf 日志文件名 （缺点：只删除日志文件，索引文件不会同步删除）**

**3.2:安装第三方软件提供命令，对数据做增量**



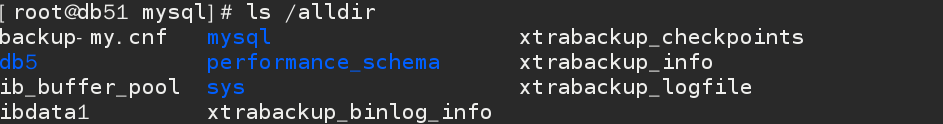
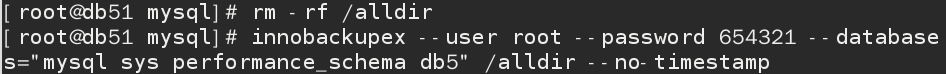
**#要求/var/lib/mysql/下是空才可以恢复**

**命令格式：**

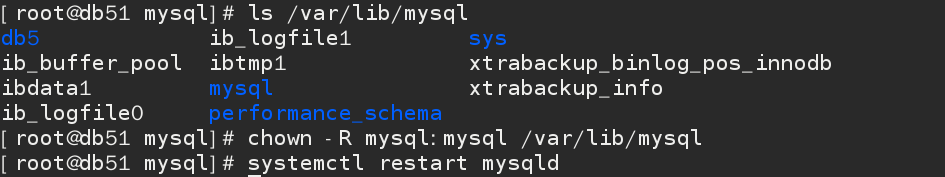
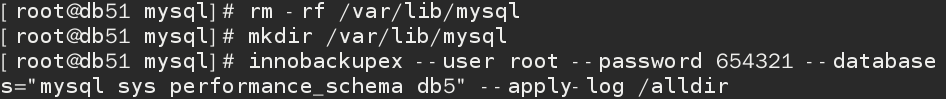
***]# innobackupex [选项]***



**完全备份：**



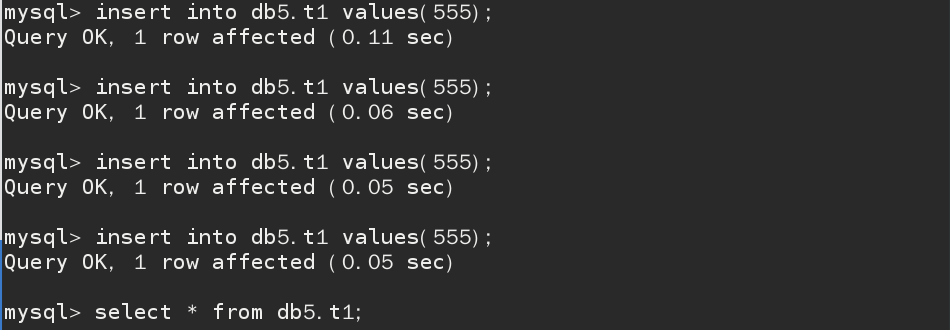
**完全恢复：**



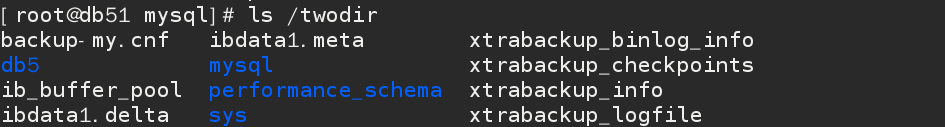
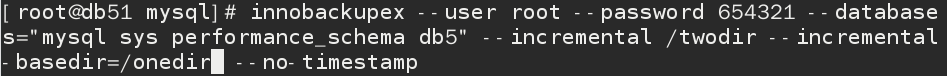
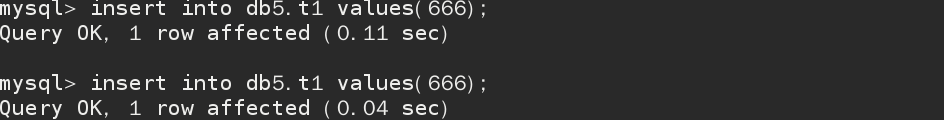
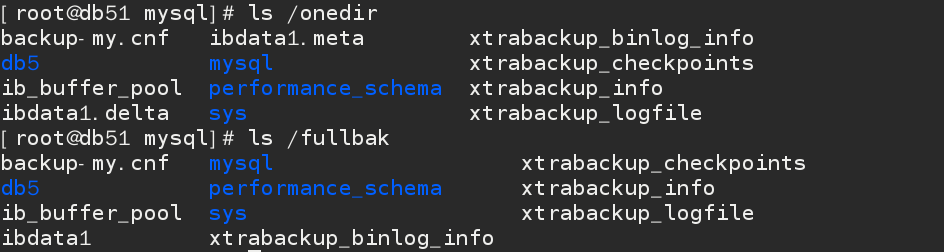
**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**增量备份：**

1. **先完全备份**



1. **再增量备份**



**增量恢复：**

**--apply-log 恢复日志**

**--redo-only 合并日志**

**--incremental-dir=目录名 恢复数据时，指定使用哪个备份目录下的文件恢复数据**

**--copy-back 拷贝数据**

**步骤：**

1. **清空数据库目录**

***]# rm -rf /var/lib/mysql***

***]# mkdir var/lib/mysql***

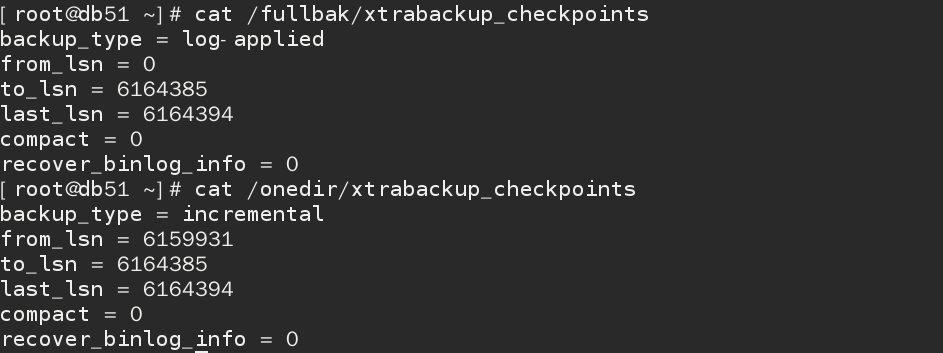
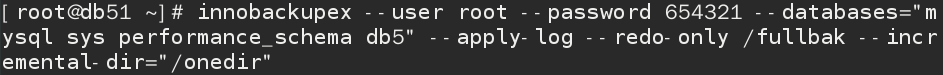


1. **恢复日志**

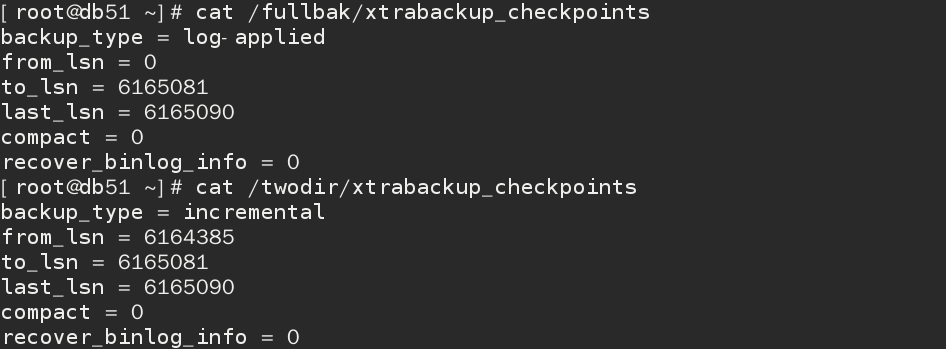
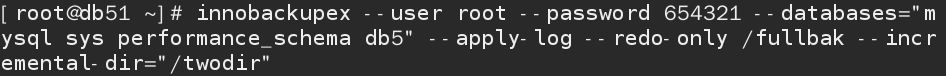
***]# innobackupex --user root --password 654321 --databases="mysql sys performance\_schema db5" --apply-log --redo-only /fullbak***



***]# innobackupex --user root --password 654321 --databases="mysql sys performance\_schema db5" --apply-log --redo-only /fullbak --incremental-dir="/onedir"***



***]# innobackupex --user root --password 654321 --databases="mysql sys performance\_schema db5" --apply-log --redo-only /fullbak --incremental-dir="/twodir"***

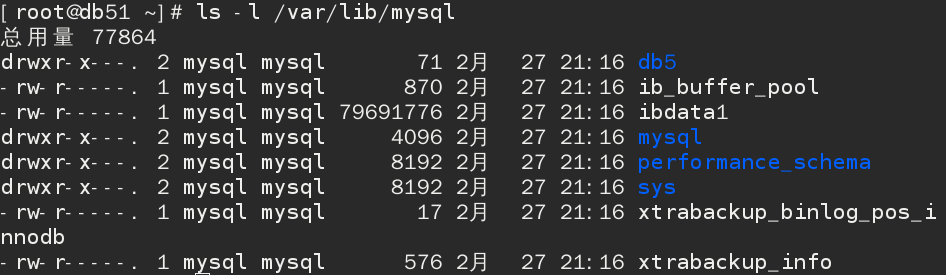
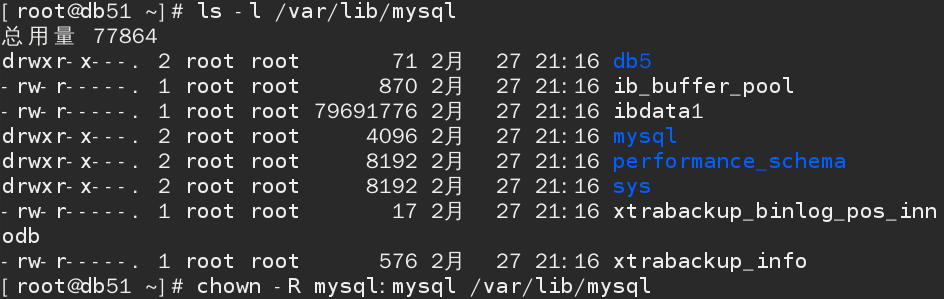


1. **拷贝备份数据到数据库目录下**

***]# innobackupex --user root --password 654321 --databases="mysql sys performance\_schema db5" --copy-back /fullbak***



1. **修改数据库目录的所有者和所属组**



1. **重启数据库服务**



1. **查看数据**

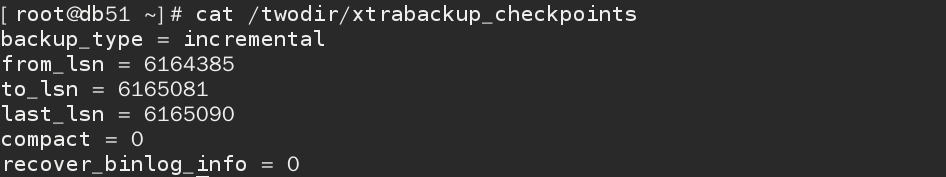
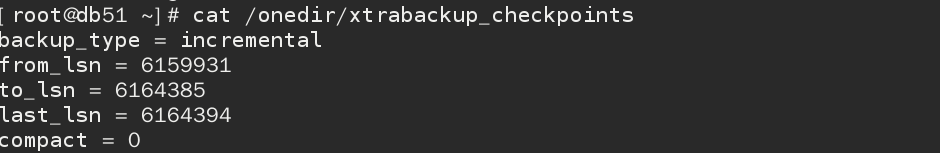
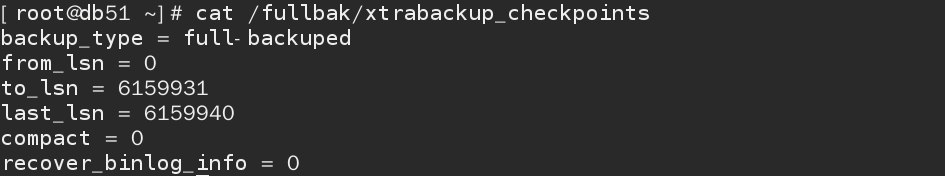


**增量备份的原理：**

**innobackupex 对innodb存储引擎的能够实现增量备份。**

**innodb存储引擎的表有事务日志文件（/var/lib/mysql/ib\_logfile0\ib\_logfile1\**

**ibdata1），记录sql操作和数据信息**

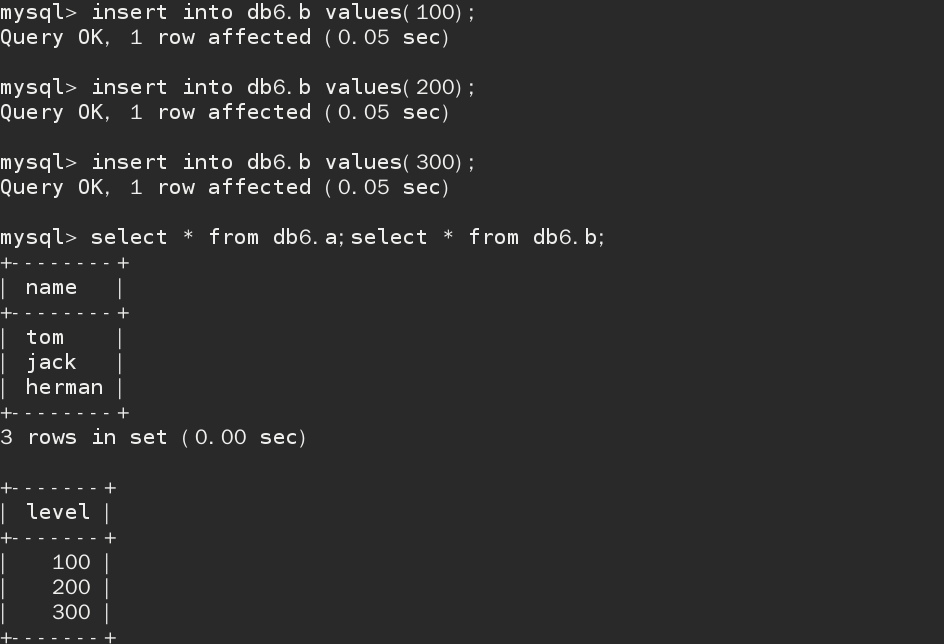
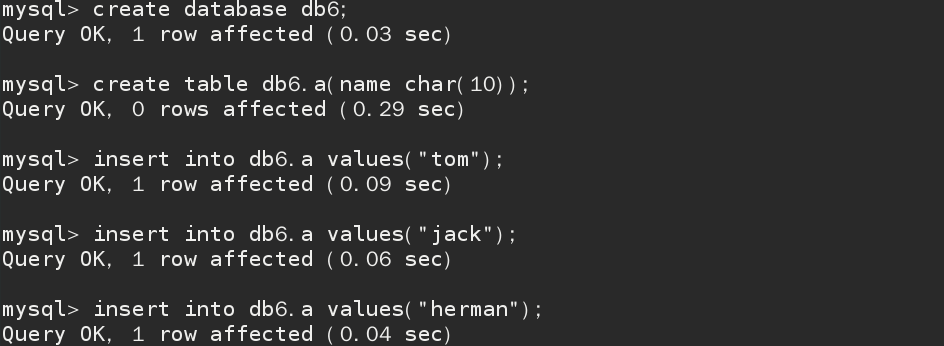


**写脚本做自动增量备份(innobackupex)**

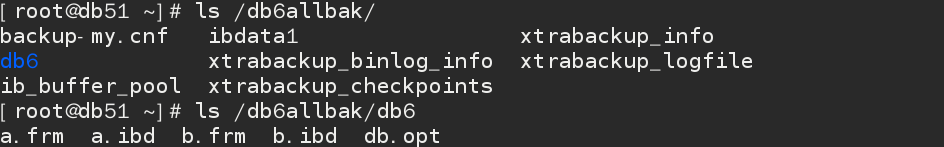
**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**从完全备份文件中恢复某个表的记录**

***环境准备：***



***1）完全备份***



***.ibd 表空间文件（存储表记录）***

***.frm 表结构文件***



1. ***恢复步骤***

***--export 导出表信息***

***import 导入表空间***

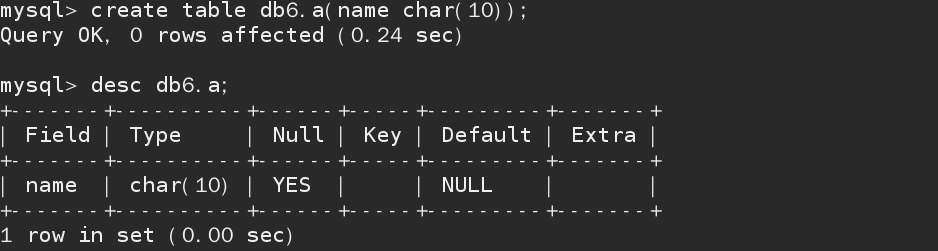
***mysql> alter table 库名.表名 discard tablespace; //删除表空间***

***2.1）导出表信息***

***]# innobackupex --user root --password 654321 --databases="db6" --apply-log --export /db6allbak***



***2.2）建表（表结构必须和删除之前的结构一样）***



***2.3）删除表空间***



***2.4）拷贝表信息文件到数据库目录下，并修改所有者和所属组为mysql***

***]# cp /db6allbak/db6/a.{cfg,exp,ibd} /var/lib/mysql/db6/***

***]# chown mysql:mysql /var/lib/mysql/db6/a.\****



***2.5）导入表空间***



***2.6）查看表记录***

