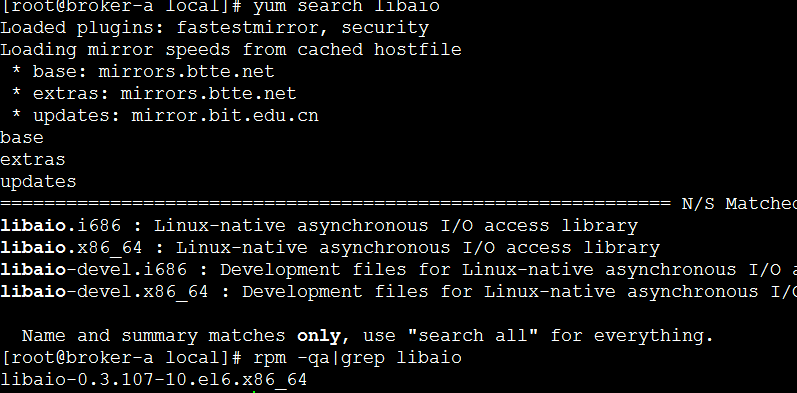
# Mysql安装部署

## 卸载原有mysql

* 检查当前mysql安装
* Rpm -qa|grep mysql
* 删除mysql
* Rpm –e --nodeps xxxx

## 检查依赖包

* Yum search libaio
* Yum install libaio



## 创建用户

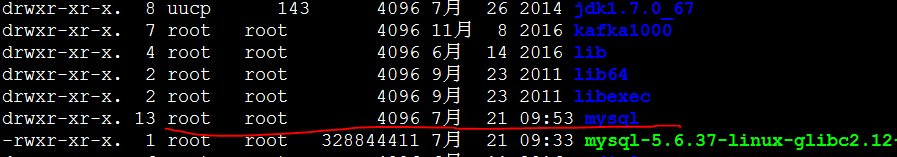
* 创建用户组及用户



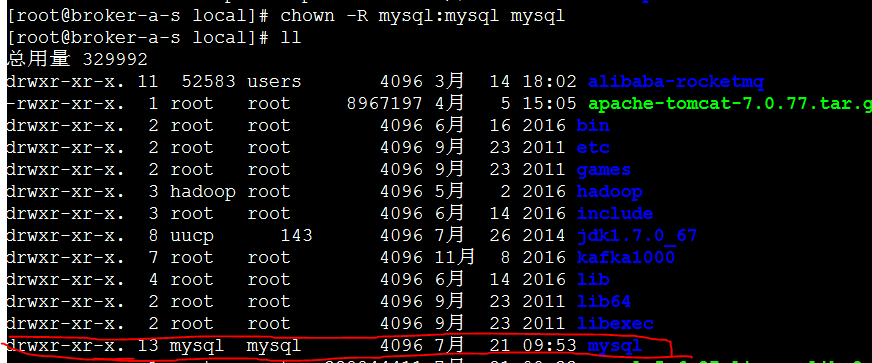
-r是指系统用户，-g指定组或gid，-s默认登陆shell

## 改变目录拥有者

* 使用root创建用户，安装mysql。将来使用mysql操作数据库。所以授权mysql能够操作相关文件权限。改变前mysql5637(版本号)文件拥有者如下：



* 使用chown命令将mysql的拥有者给mysql用户



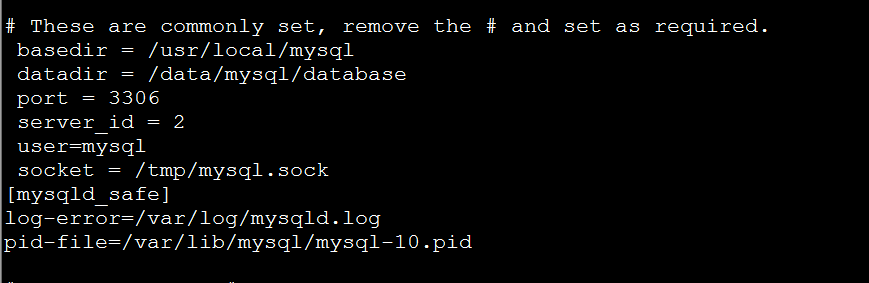
* -R是指处理当前目录及其子目录下的所有文件生效

## 配置执行文件

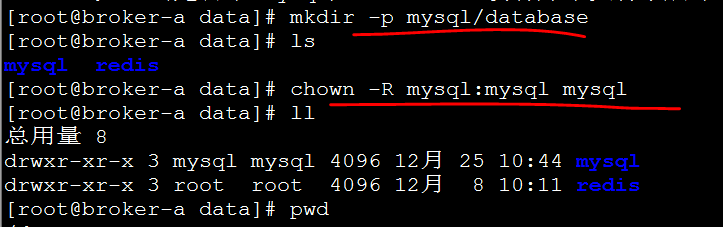
* 在/etc目录下配置mysql核心配置文件my.cnf



* 配置部分信息

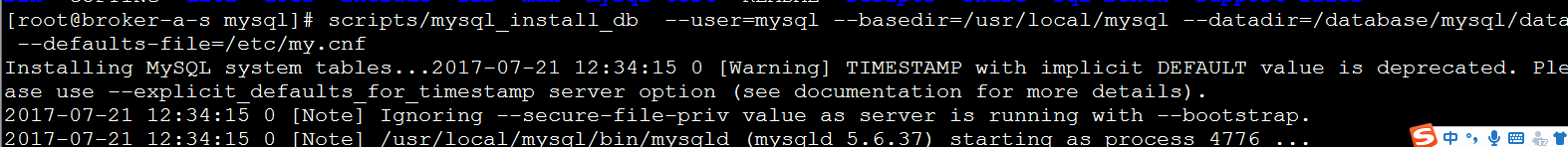


* 改变datadir执行权限,其他目录同样道理



## 初始化mysql

* 执行scripts/mysql\_install\_db --defaults-file=/etc/my.cnf --user=mysql --basedir=/usr/local/mysql --datadir=/data/mysql/database

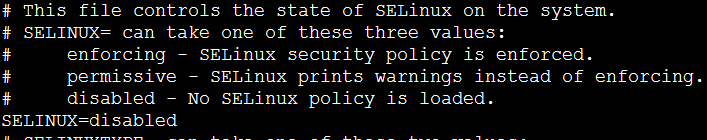


## 设置root用户密码

* 这个是设置mysql的root用户密码

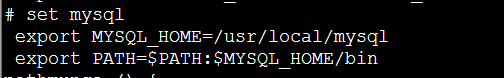


## 修改/etc/selinux/config



* 将enforcing修改为disabled

## 设置环境变量



* **Shell> source /etc/profile**

## 启动mysql

### Mysqld\_safe启动

* bin/mysqld\_safe --defaults-file=/etc/my.cnf --user=mysql &

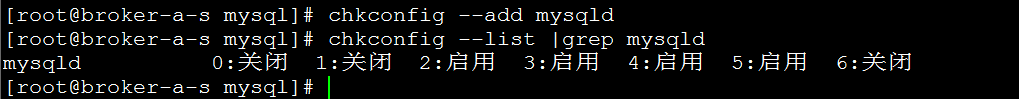


### 开机自动启动

* 开机自动启动调用mysqld进程。需要将supports/mysql.server复制到linux系统/etc/init.d目录下面：
* **cp support-files/mysql.server /etc/init.d/mysqld**
* **该命令就是复制/etc/init.d目录下mysqld文件**



* 设置可执行权限(非必须，具体查看一下)
* Chmod +x /etc/init.d/mysqld
* 添加开机启动，只要2-5为开启即可



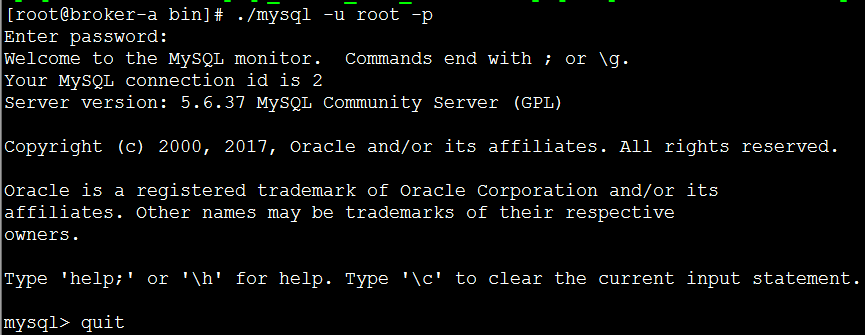
## 停止mysql服务

* 使用mysqladmin停止mysql服务，命令如下
* Shell> mysqladmin –u root –p shutdown
* Shell> Enter password:



## 登陆mysql

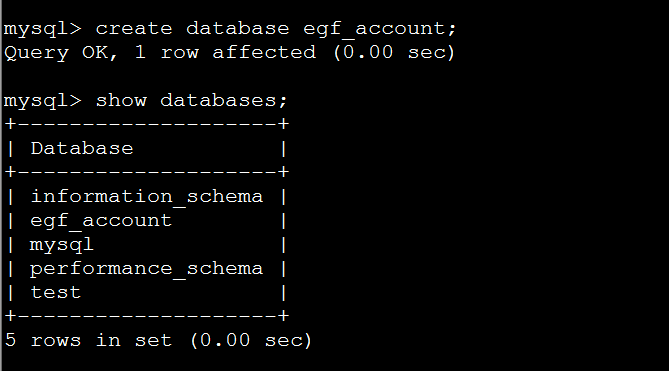
* Shell> mysql –u root(userName) -p
* 退出mysql数据库，直接quit即可



# Mysql基本操作

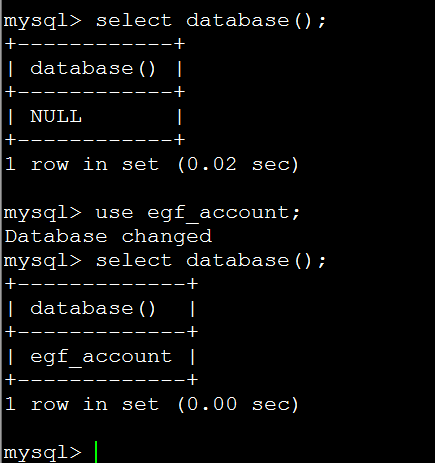
## Create databases

* Create database database\_Name



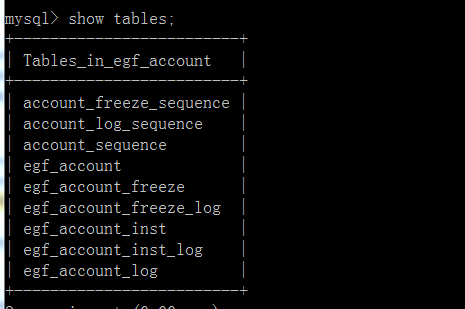
## 查询当前使用db

* Select database()



## 查看当前db中表

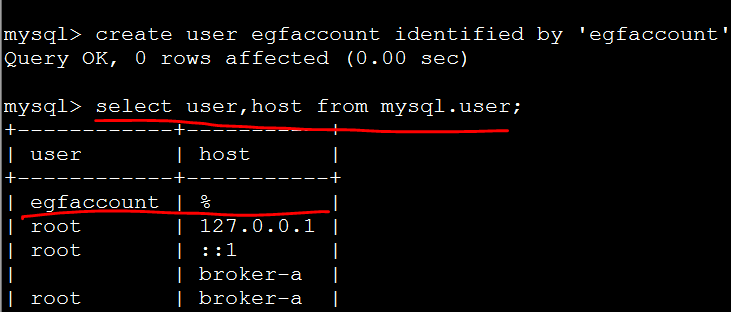
* Show tables



## 创建用户

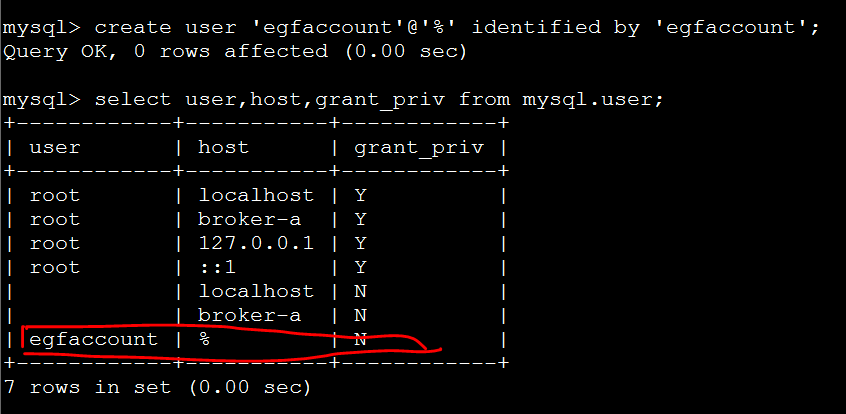
### 默认创建

* **Create use**r username  **identified by** ‘password’



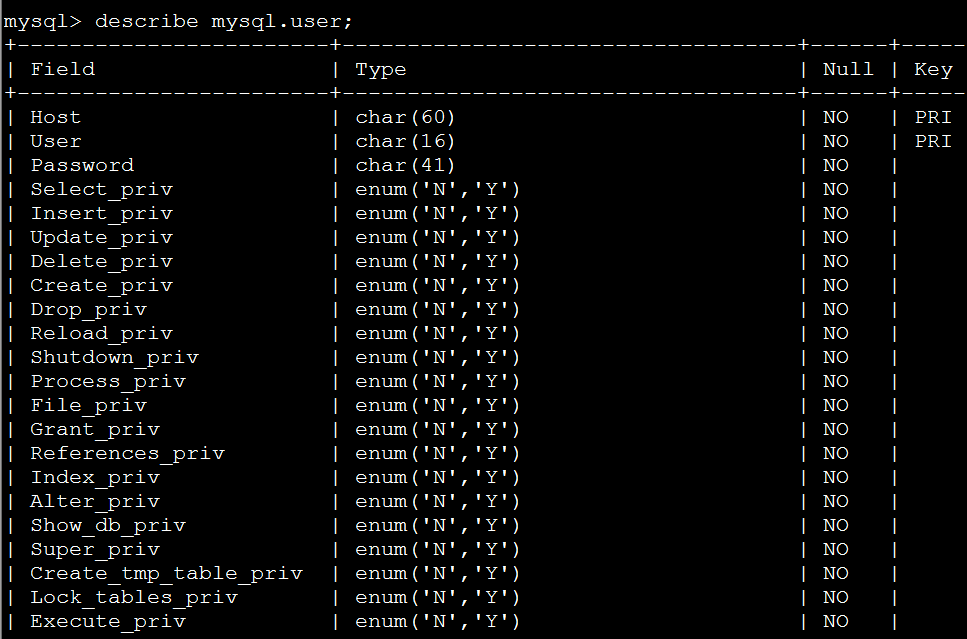
### 带有参数创建

* 命令格式：create user ‘userName’@’pattern’ identified by ‘password’
* Pattern有如下几种类型：
* Localhost：从当前服务服务器上去连接mysql
* %: 可以从任何一台服务器去连接mysql



## 查询当前用户权限

* 创建完用户之后，可以使用管理员用户在mysql.user表中查看当前用户的权限。该表中记录了当前用户的拥有的各种权限。
* 下图中是展现的部分字段



## 删除用户

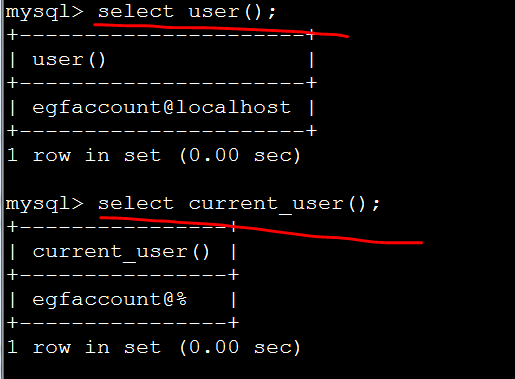
* 命令格式：drop user ‘userName’@’loclhost’

## 更改用户密码

* Update mysql.user set password = password(‘newPassword’) Where user =’userName’; flush privileges;

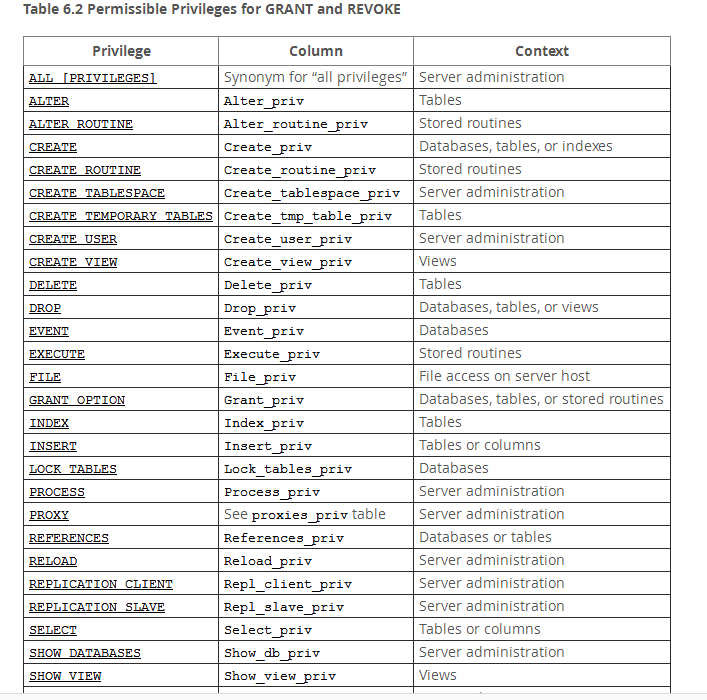
## 查询当前登录用户

* 查询当前登录用户有两种方式：select user() 或 select current\_user()



## Mysql权限表

* Mysql提供的用户权限表如下：



* 具体可以参见如下连接：
* https://dev.mysql.com/doc/refman/5.6/en/privileges-provided.html

## 给用户授权

### 官方命令格式如下：





### 常用命令格式：

* **grant privilegeCode on priv\_level to ‘user’@’host’ [auth\_operation ]**

**[with grant opriton] ;**

* **flush privileges;**
* **其中auth\_operation最好不要省略。**
* **常用的权限privilegeCode：**

**All privileges ：所有权限**

**Select\_priv ： 查询权限**

**Update\_priv：更新权限**

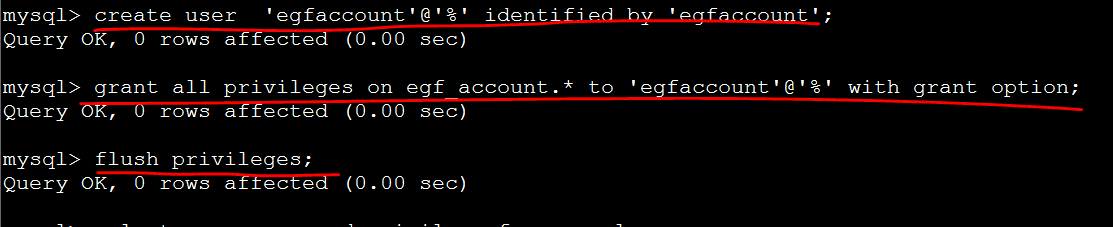
**Drop ： 删除数据库，表的权限**

**Create\_tablespace ：创建表空间的权限**

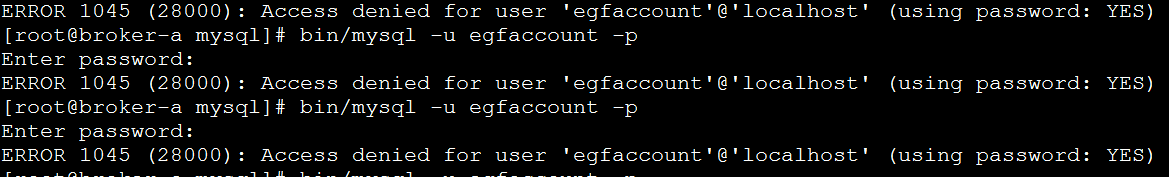
* **Priv\_level**
* \*.\* : 授权数据库服务器所有数据库的权限
* Db\_name**.**\* : 授予用户db\_name所有权限
* Db\_name.db\_tablename 授予用户当前表的权限
* Table\_name : 授权所有表的权限
* user@host中的host如下方式
* localhost : 只允许用户在本地登陆
* % : 允许用户在任何一台服务器登陆
* Hostip : 指定用户在某一台IP地址登陆
* **auth\_operation**
* IDENTIFIED BY 'auth\_string'
* IDENTIFIED BY PASSWORD 'hash\_string
* 在授权的时候，最好加上这部分功能。

### 举例说明

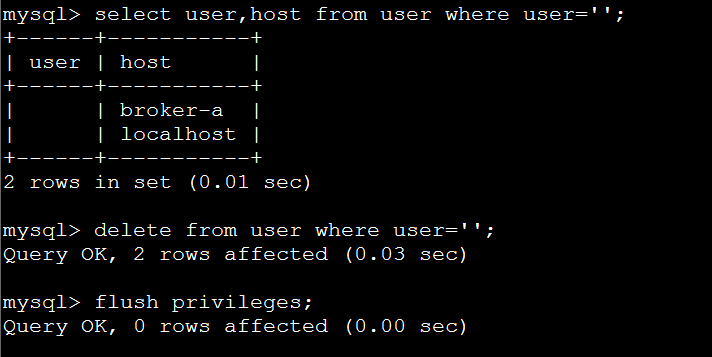
* 创建用户及授权的例子



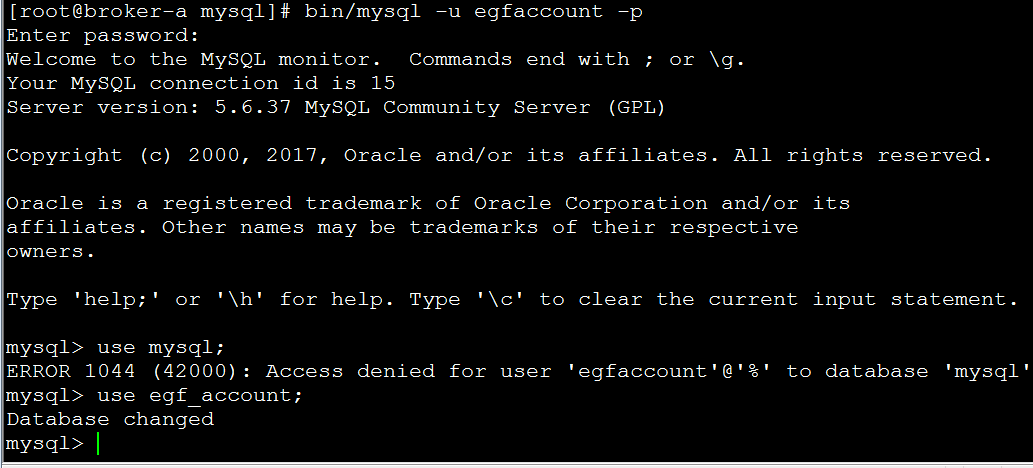
* 换作新用户egfaccount登录，发现一只报如下错误



* 搜查了一下，是因为mysql.user中有一个用户名为空，host为localhost的用户在搞鬼。删除该用户即可。



* 再用新用户重新登录一下



## 回收权限

* 官方撤销权限命令格式：

REVOKE

*priv\_type* [(*column\_list*)]

[, *priv\_type* [(*column\_list*)]] ...

ON [*object\_type*] *priv\_level*

FROM *user* [, *user*] ...

REVOKE ALL [PRIVILEGES], GRANT OPTION

FROM *user* [, *user*] ...

REVOKE PROXY ON *user*

FROM *user* [, *user*] ...

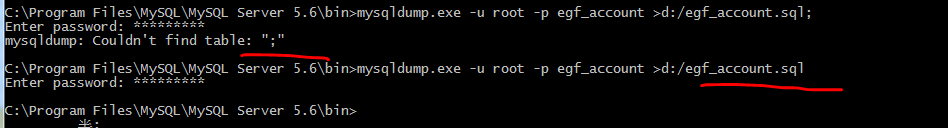
# mysql导入导出

## 导出数据

在mysql中使用mysqldump进行数据备份。

### Windows环境

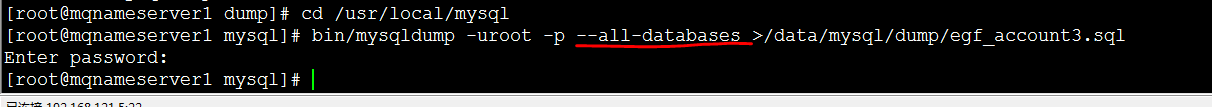
* 进入到Mysql安装bin目录下，执行如下脚本就可以将数据库导出。
* **Mysqldump –u username –p databaseName >filename**



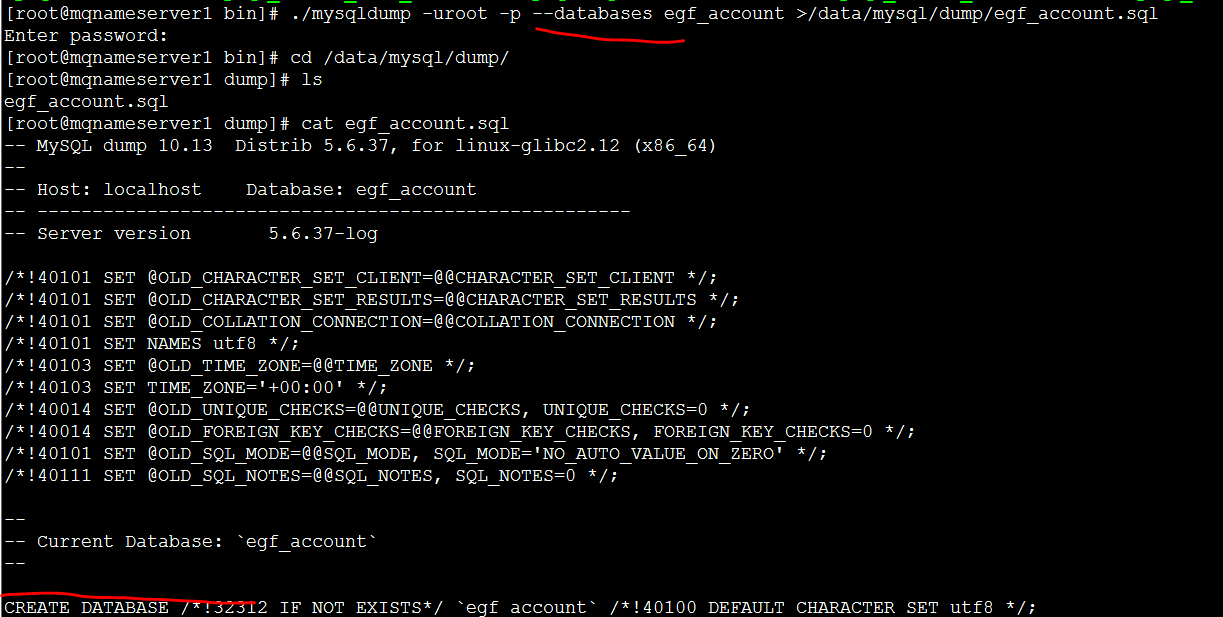
注意导出语句后面不能带分号 ;

### Linux环境

* 命令格式：Mysqldump -u root –p [options] db1,db2 >filename
* Root用户是指mysql数据库的root用户，不是OS的root用户。
* [options]常用的有：
* **--all-databases**：将所有的数据库都导出到fileName文件中。使用该选项则省略db1,db2这些参数。



* **--databases** : 指定导出数据库的数据及格式。



* 使用mysqldump复制的时候，如果带有参数 - -databases或者- -all—databases时，备份脚本里面是含有create database语句 。

# 事务常用操作

## 设置非自动提交

Set autocommit = 0/1; 0代表非自动提交，1代表自动提交。

## 隔离级别操作

可重复读: repeatable-read

读已提交：read-committed

读未提交：read-uncommitted

串行化：serializable

### 显示隔离级别

显示当前会话：Select @@tx\_isolation;

显示全局隔离：select @@global.tx\_isolation

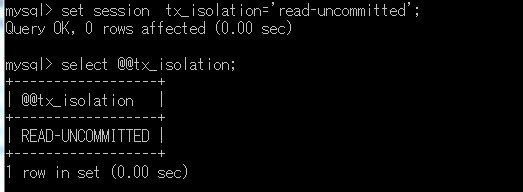
### 设置隔离级别

#### Set transaction syntx

1. **Set transaction isolation level level\_value**:
2. **Level\_value:read-committed/read-uncommitted/repeatable-read/serializable**

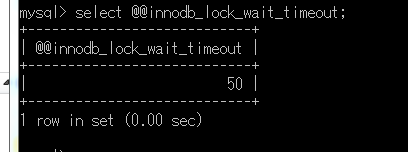
#### Set isolation

Set [session | global] tx\_isolation=’level-value’

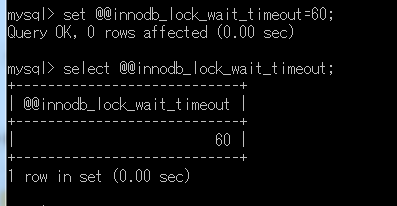


## 锁等待超时时间设置

1. Innodb\_lock\_wait\_timeout用来设置锁等待超时时间设置，默认为50秒，该参数可以动态设置。



1. 动态设置锁超时时间：



## 开启事物

在命令行可以显示开启一个事物，通过begin, start transaction或set autocommitted=0开启，后续需要commit 【work】命令提交

# 性能调优

## CPU

对于多核的CPU而言，可以调整innodb\_read\_Io\_threads和innodb\_read\_io\_threads大小，增加I/O处理能力。

## 内存

* 在Innodb引擎中，内存大小直接反应了数据库的性能。Innodb引擎即缓存了数据又缓存索引，并且将他们至于很大的缓冲池中，即innodb buffer pool中。
* 一般情况下，内存的大小和数据文件大小相当时，性能达到最佳状态。通常情况下，通过比较物理磁盘的读取和内存读取的比例来判断缓存命中率。

## 硬盘

### 机械硬盘