Mysql

启动：net start mysql（服务名称）

停止：net stop mysql

提示错误：拒绝访问

用管理员身份启动cmd

# Linux安装MySQL

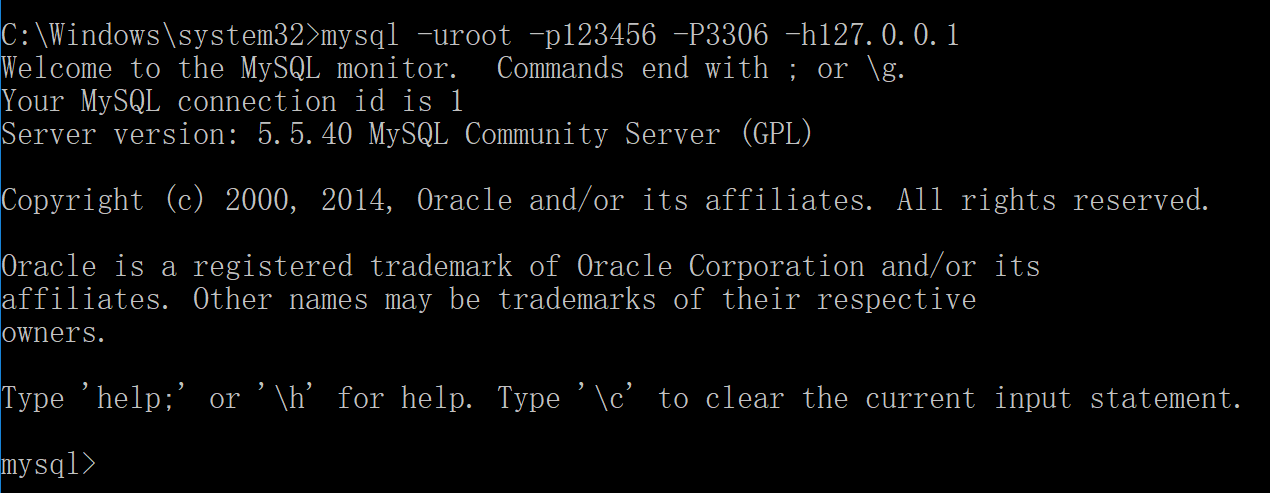
<https://www.jianshu.com/p/276d59cbc529>

# 登录、退出

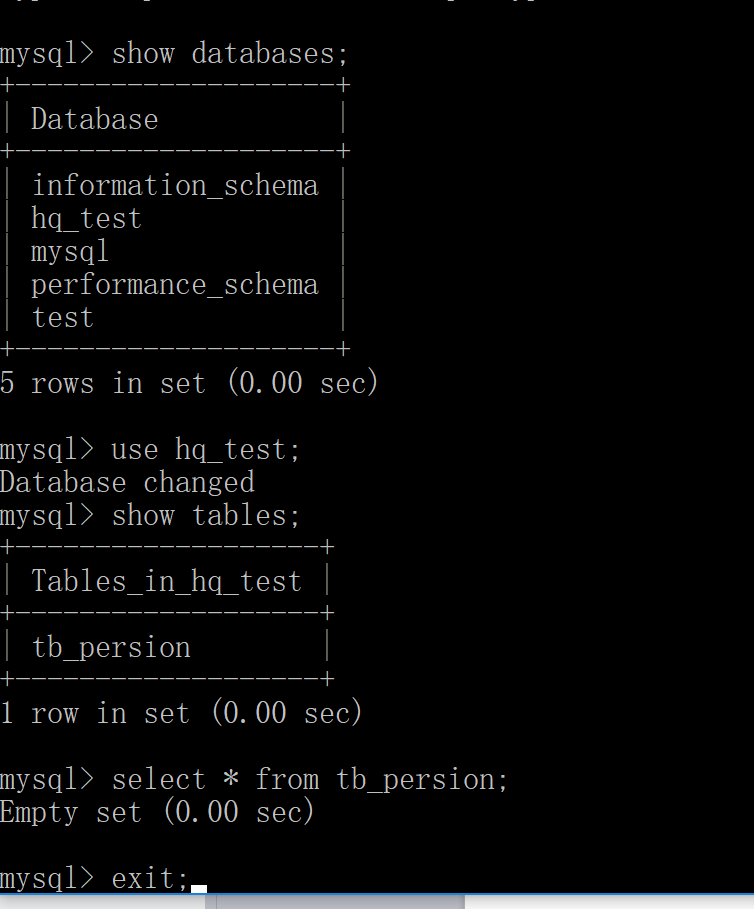


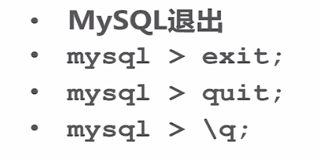
登录u:用户名 p:密码

P(大写P)端口号 -h:ip地址



退出





# 修改前面的提示符





# 常用命令



# 规范



# 创建数据库



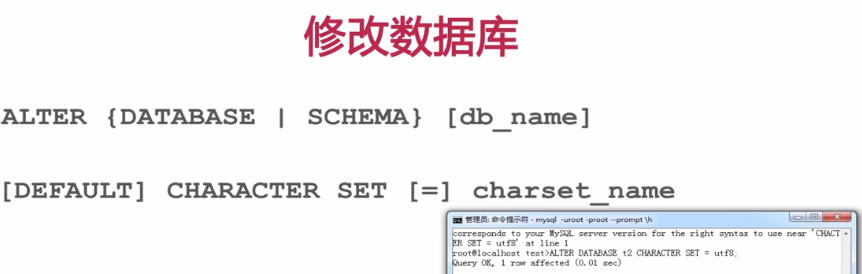
# 查看数据库



# 修改数据库编码方式

查看数据库编码方式：

show create database hq\_test;



# 删除数据库



# 数据类型

# 整型



# 浮点型



# 时间类型

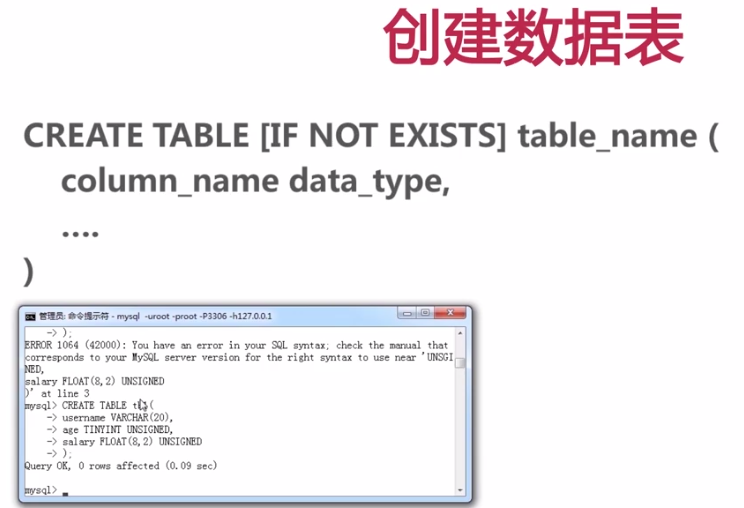


一般用时间戳来存储

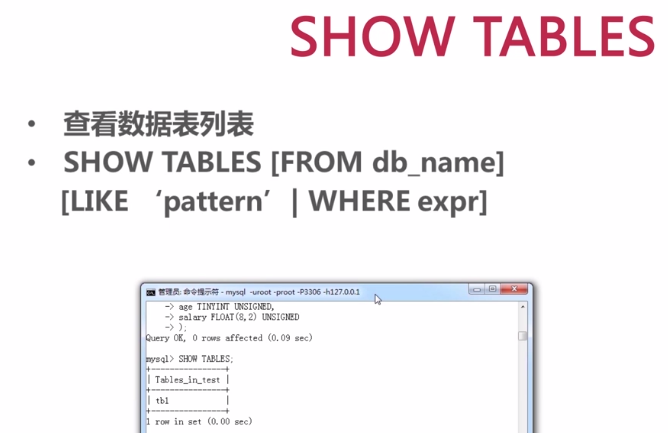
# 字符型



# 创建表



# 查看数据库表

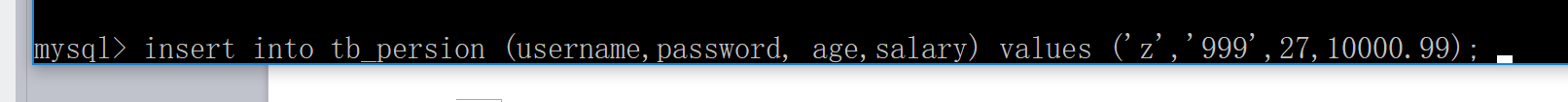


# 查看表结构



# 插入数据





省略字段名的话，那么所有字段都需要赋值。

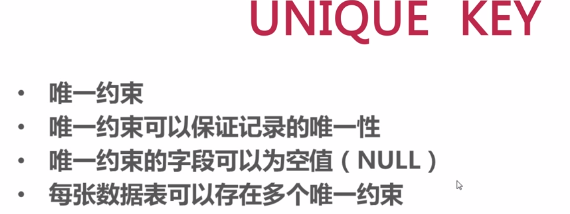
# 查询语句



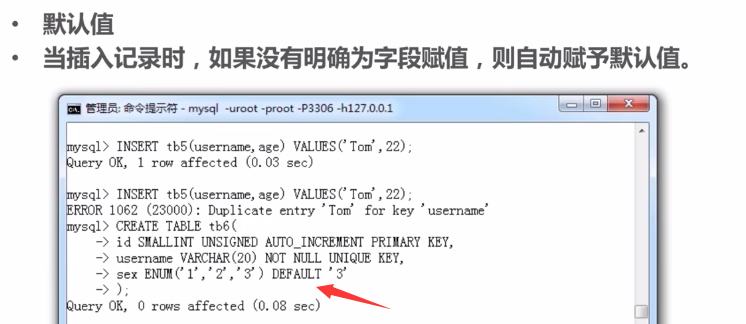
# 主键primary key



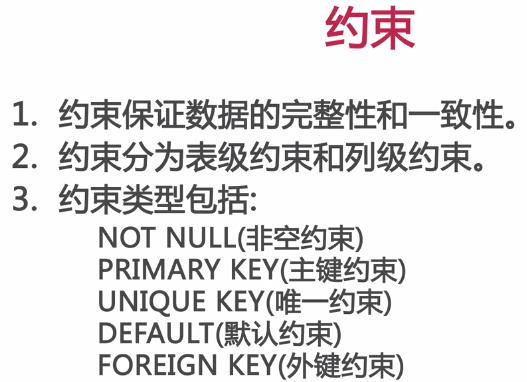
# 唯一约束unique key



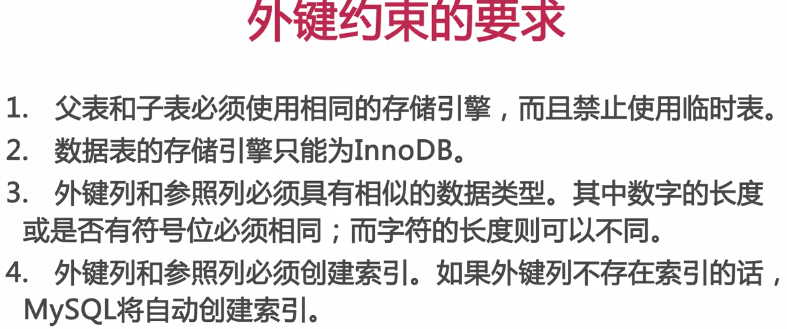
# 默认值

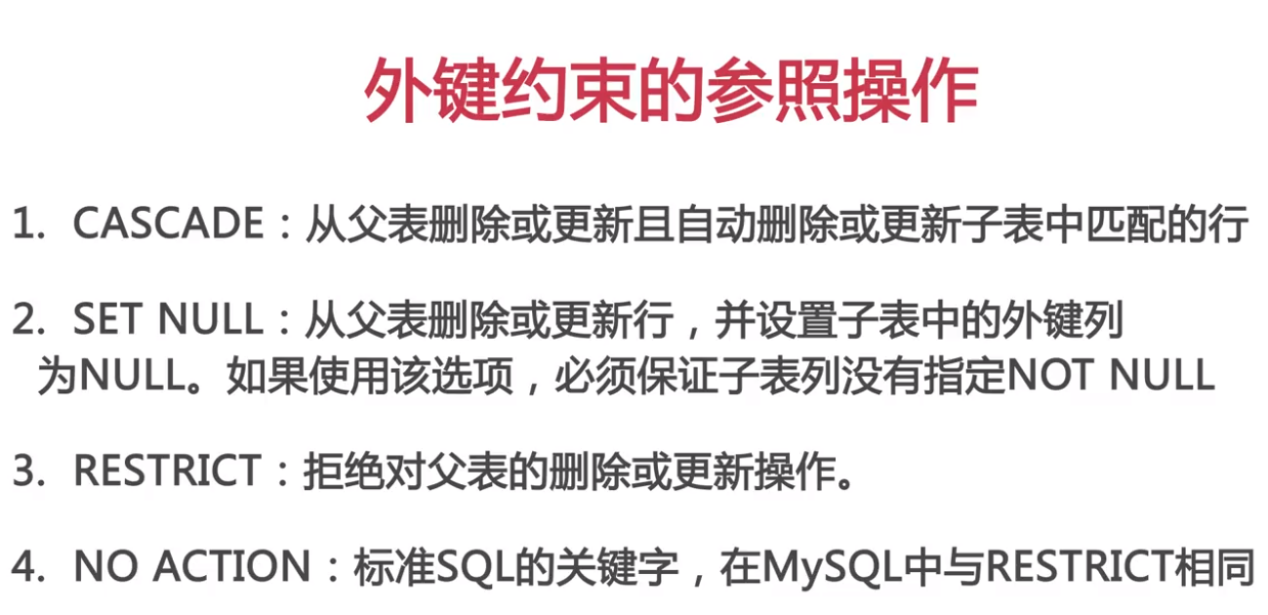


# 约束



# 外键约束(存储引擎为InnoDB)





# 修改存储引擎



# 修改数据库表结构

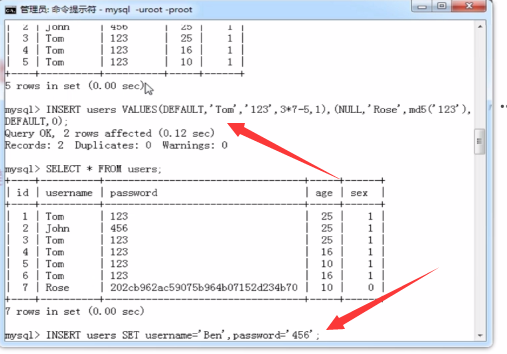
增加



删除一个或多个



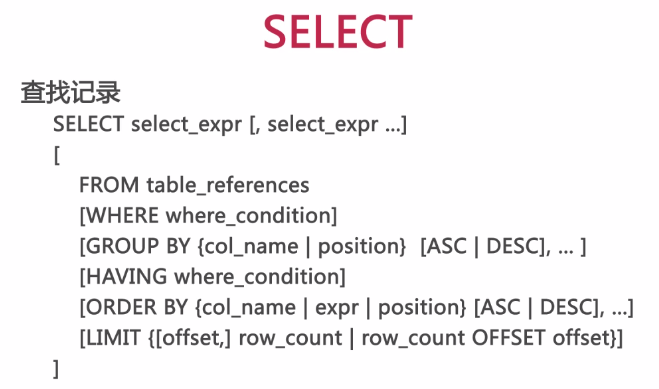
# 插入(两种方式)



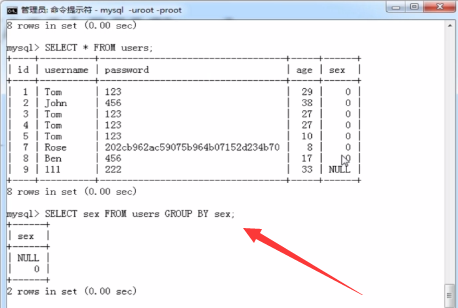
# 更新数据



# 查询数据



# Group by（会将字段数据去重）



对于选择最大值最小值等操作的字段必须是字符型，不能是varchar等类型。

# Having (接在group by后面使用)聚合函数

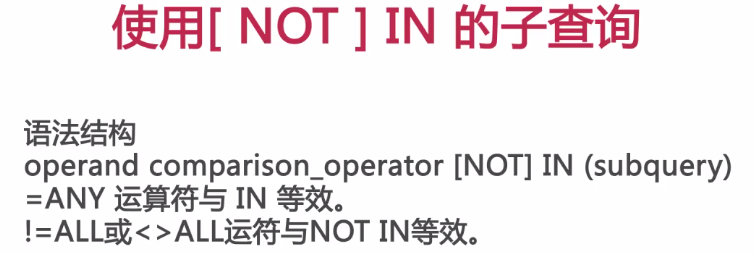
# 子查询 ANY SOME ALL关键字



用法： select \* from tb\_persion where id > ANY (select age from tb\_user);

解释：大于子查询多个返回值中的最小值(ANY)

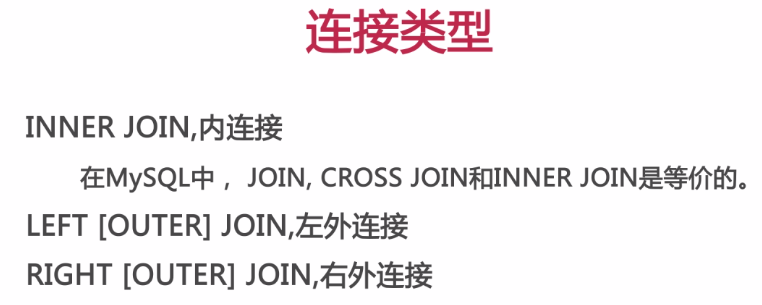
# IN的用法

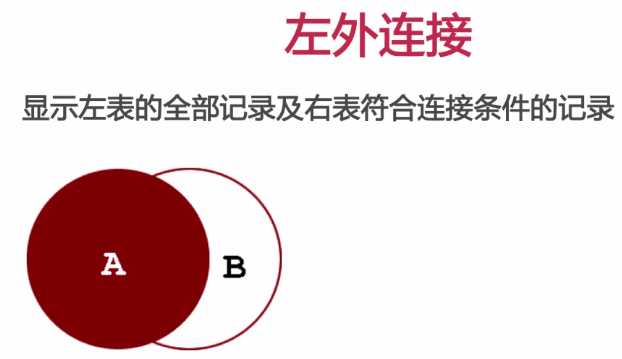
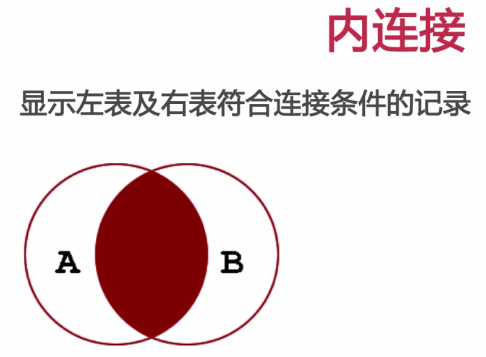


SELECT \* FROM tb\_persion WHERE id in (SELECT id FROM tb\_persion);

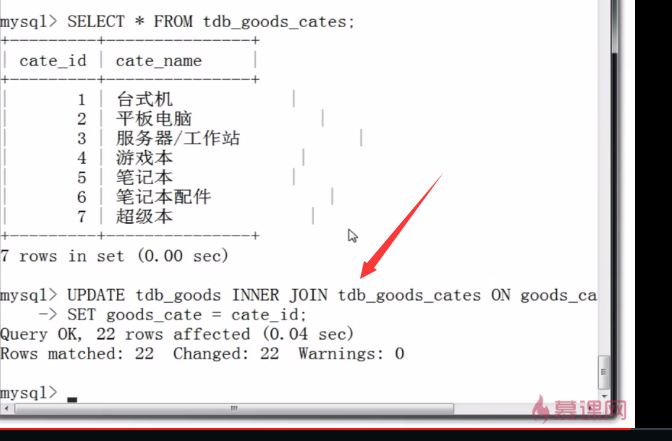
SELECT \* FROM tb\_persion WHERE id in (1,3);

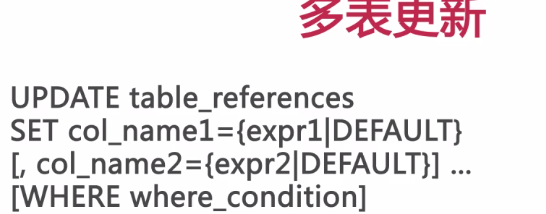
# 连接类型(左、右、内连接)





# 多表更新





多表更新语句

UPDATE tb\_persion\_bk2 a INNER JOIN tb\_info b ON a.bk\_id=b.id

SET a.username=b.address,a.password=b.hobby;

假设5表有六条数据，b表有2条数据。

内连接：放回的数据为最小数据行数。

SELECT a.age,b.hobby FROM tb\_persion a INNER JOIN tb\_info b ON a.id=b.id;



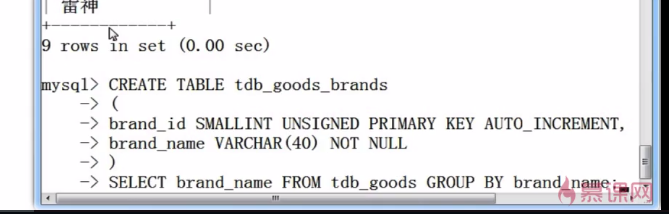
左连接：(那么左表为父表，它有多少条数据就返回多少条数据，如果关联的表比它少，与null显示)

SELECT a.age,b.hobby FROM tb\_persion a LEFT JOIN tb\_info b ON a.id=b.id;



右连接则与左连接正好相反

# 关联创建表



创建表时，将另一张表的数据填充进去

CREATE TABLE tb\_persion\_bk (

bk\_id INT UNSIGNED PRIMARY KEY AUTO\_INCREMENT,

username varchar(255),

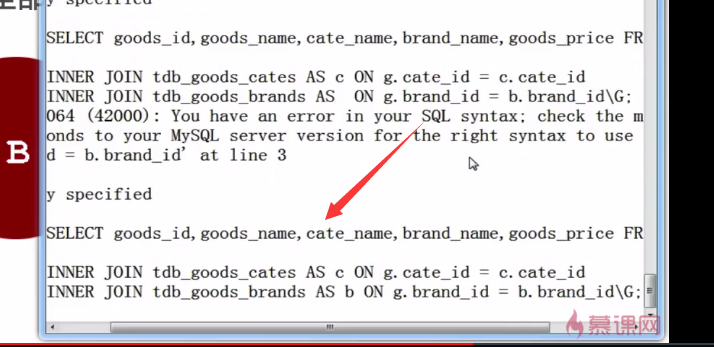
password varchar(255)

)

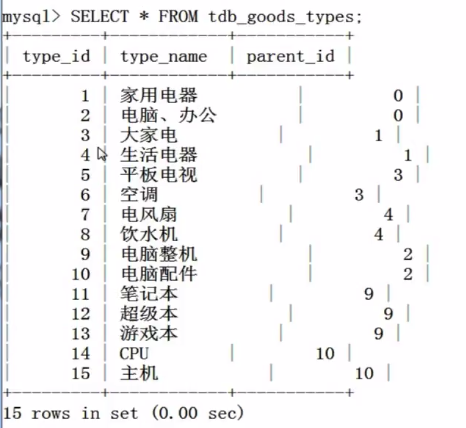
select username,password FROM tb\_persion;

注意事项：查询出来的字段类型要与创建时的类型一致。

# 三张表联合查询



# 无限级分类设计



根据parent\_id关联type\_id找到子类。

比如【大家电】属于【家用电器】

查询方式采用【自身连接】

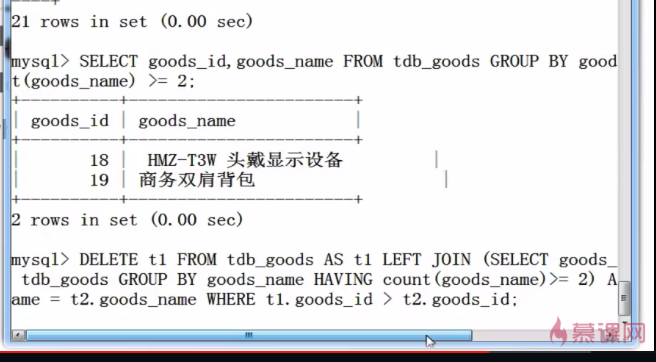
如何查询：思路，假设tdb\_goods\_types的右边还有一张tdb\_goods\_types表，要找出左边父表的type\_name属于哪个父类(一级分类)

Select a.type\_id , a.type\_name,b.type\_name AS parent from tdb\_goods\_types a LEFT JOIN

tdb\_goods\_types b ON a.parent\_id=b.type\_id;

# 自身连接

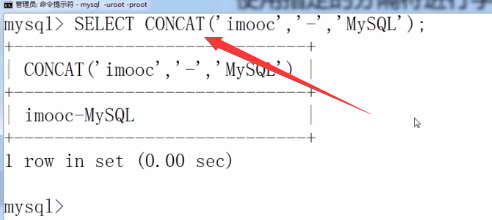
# 多表删除

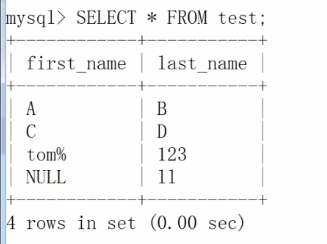


# 字符函数



Concat()函数使用：



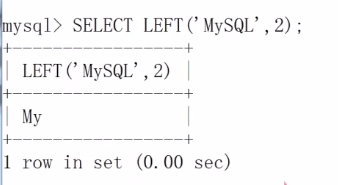


Select CONCAT(first\_name,last\_name) AS full\_name FROM test;

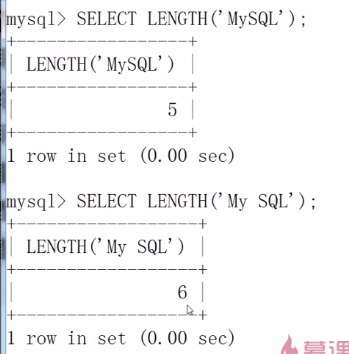
Format()数值格式：

Format(age,2);保留两位小数，四舍五入。

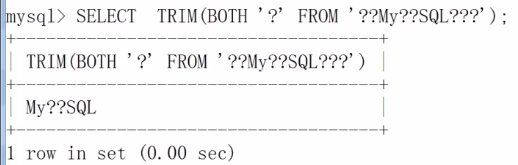
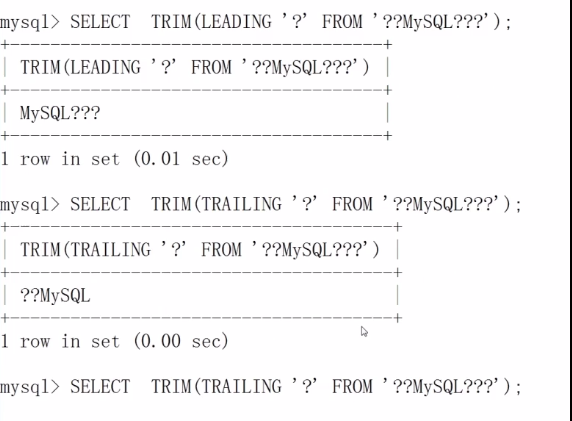
Left用法：第二个参数为位数



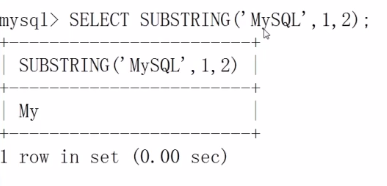


空格也算长度

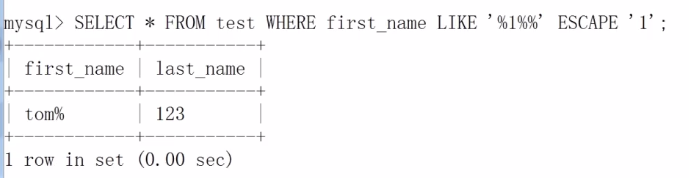
前导与后导







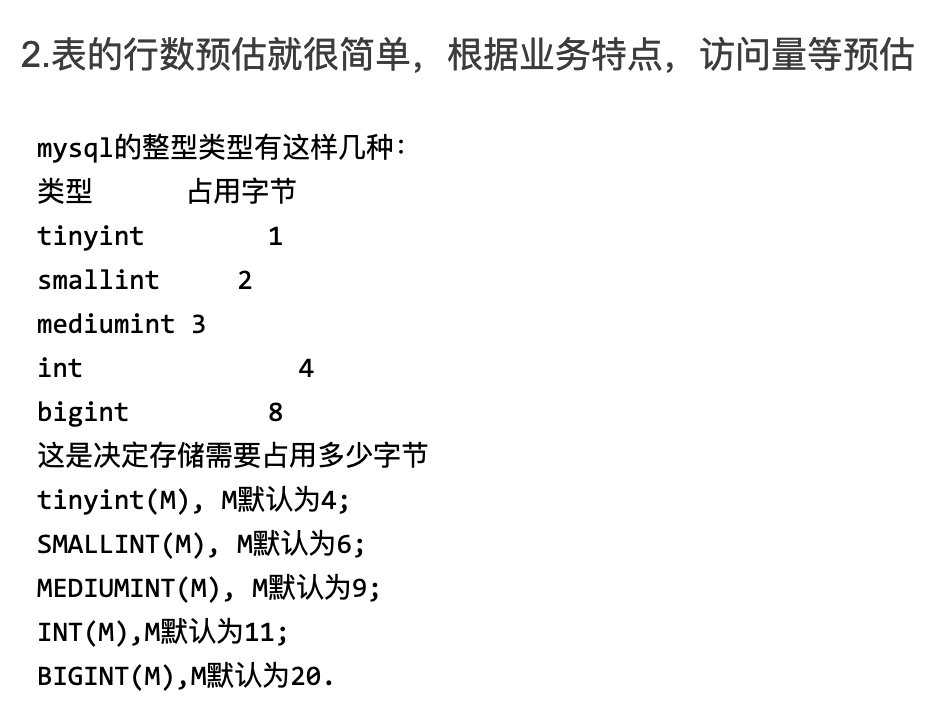
含有%但需要通配符



# MySQL各个数据类型默认值

参考博客：

<https://blog.csdn.net/weixin_41926301/article/details/94509640>



# linux安装MySQL

参考博客

Linux下安装mysql-5.7.24：

<https://www.jianshu.com/p/276d59cbc529>

启动的方式参考这个：

<https://blog.csdn.net/qq_37598011/article/details/93489404>

授权方式写的不一样：

<https://www.cnblogs.com/wendy-0901/p/12673705.html>

## 安装前检查是否已经安装过

一定要卸载干净，删除干净

|  |
| --- |
| [root@localhost local]# rpm -qa | grep mysql  [root@localhost local]# |

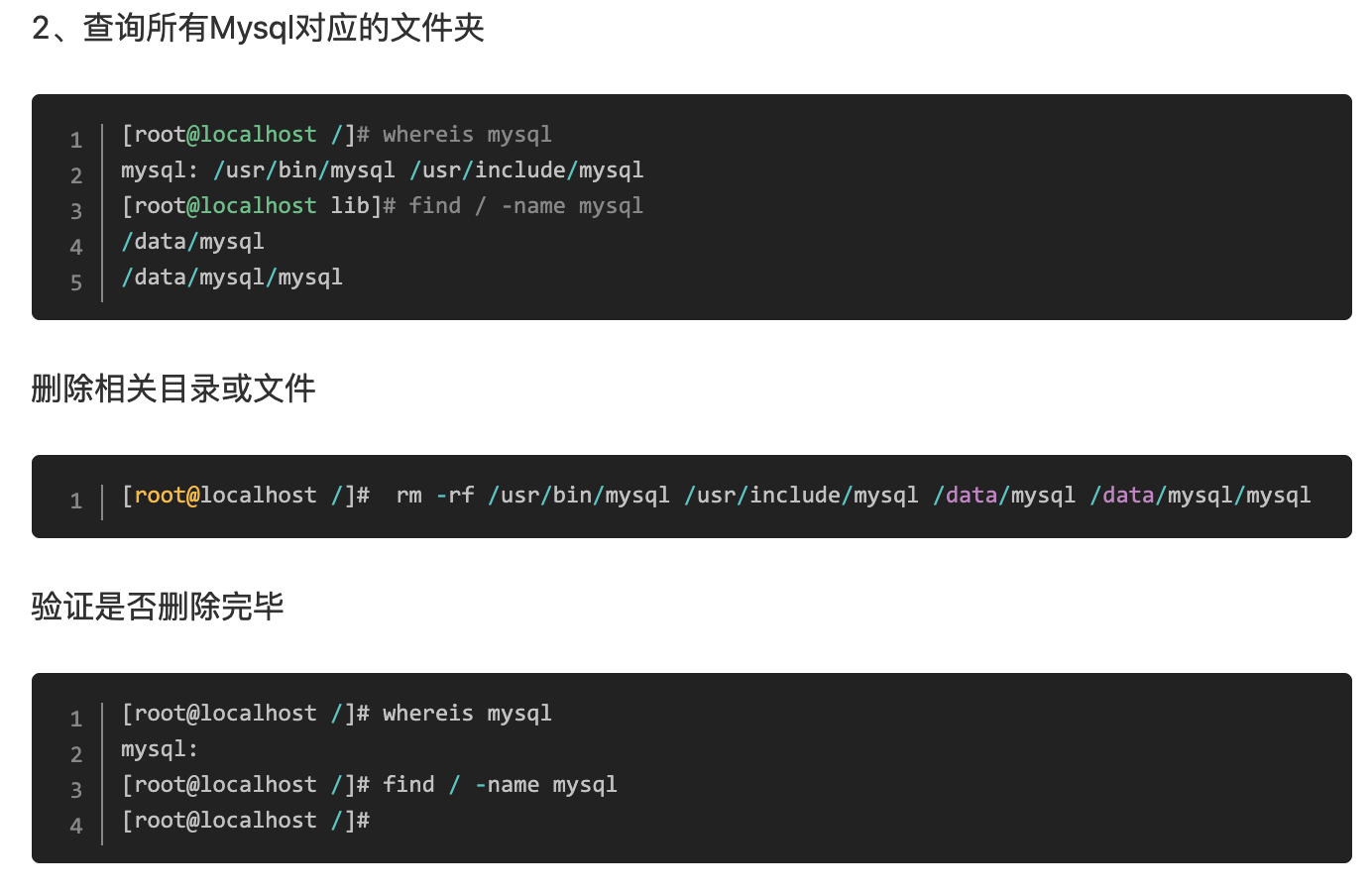
这样就表示没有，如果已经存在，就需要删除再重新安装。

查看mysql文件夹是否存在：

whereis mysql

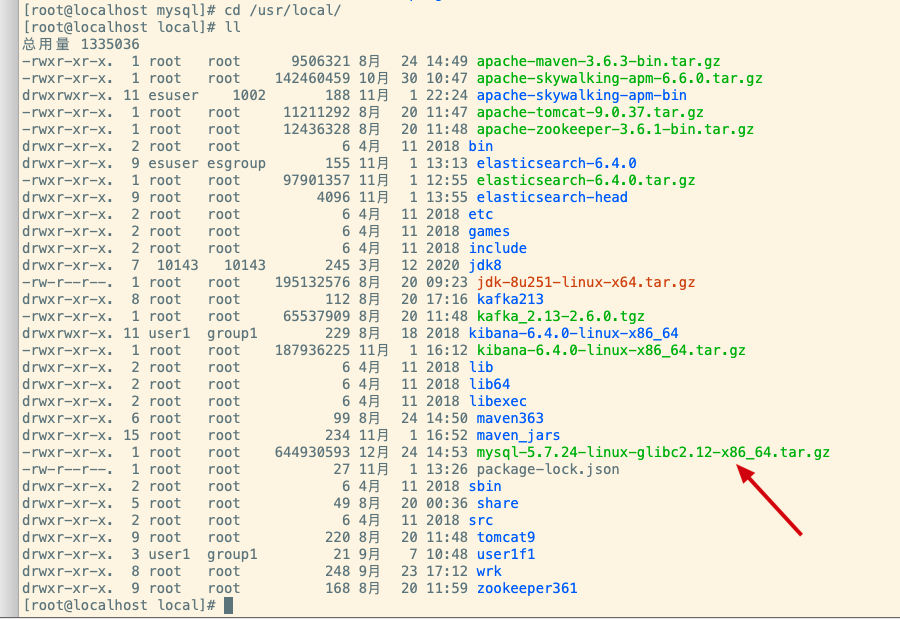
存在就删除

find / -name mysql



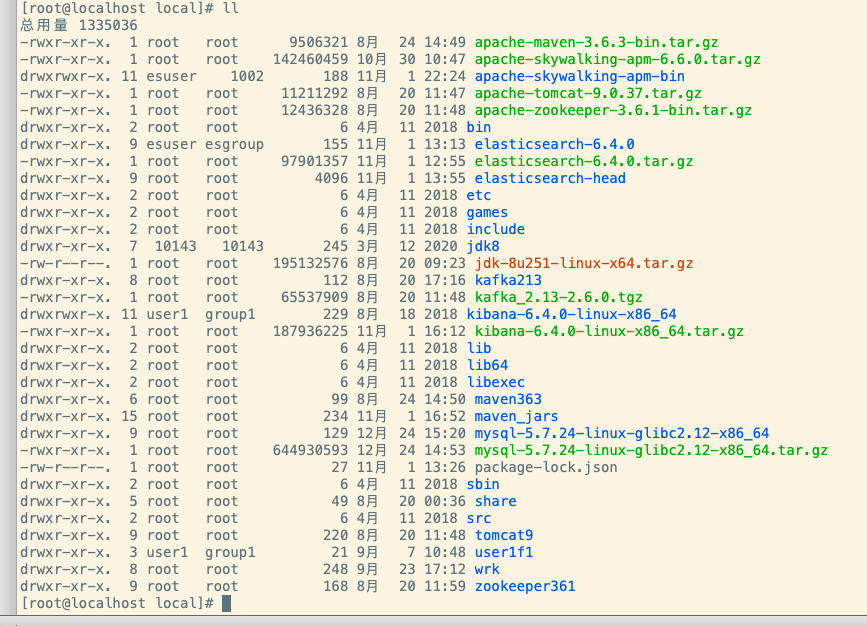
## 上传并解压

目录为： /usr/local/下



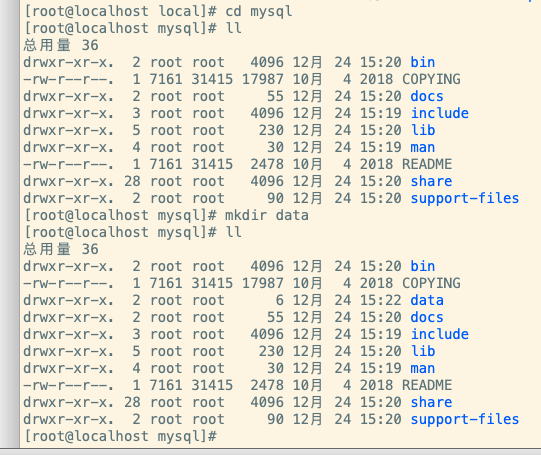
解压：

tar -zxvf



重新命名为mysql

## 在mysql目录下创建data目录



## mysql用户和用户组的创建

检查mysql用户组和用户是否存在，如果没有，则创建





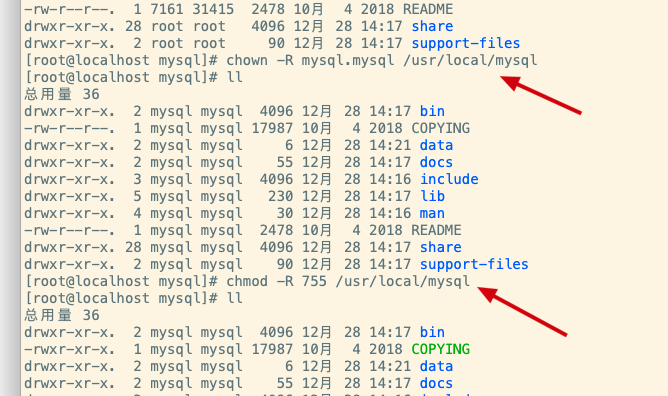




#useradd -r参数表示mysql用户是系统用户，不可用于登录系统

可以为后续操作使用

## 为文件夹赋权



chown -R mysql.mysql /usr/local/mysql

chmod -R 755 /usr/local/mysql

## 编译安装并初始化mysql

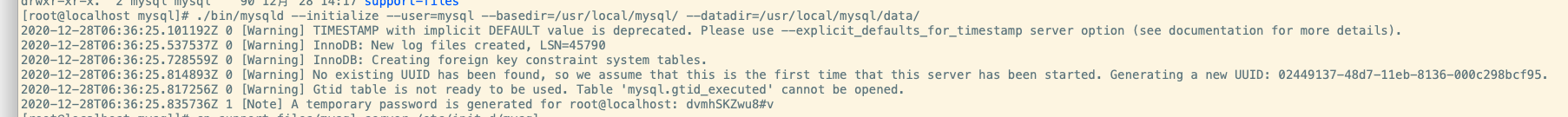
务必记住初始化输出日志末尾的密码（数据库管理员临时密码）

进入bin目录

cd /usr/local/mysql/bin

初始化mysql:

|  |
| --- |
| [root@localhost /]# cd /usr/local/mysql/bin  [root@localhost bin]# ./mysqld --initialize --user=mysql --datadir=/usr/local/mysql/data/ --basedir=/usr/local/mysql/ |



最后几位是密码。

## 编辑配置文件my.cnf

vim /etc/my.cnf

|  |
| --- |
| [mysqld]  #设置mysql的安装目录  basedir =/usr/local/mysql  #设置mysql数据库的数据存放目录  datadir = /usr/local/mysql/data  #设置端口  port = 3306  socket = /tmp/mysql.sock  #设置字符集  character-set-server=utf8  max\_connections=600  innodb\_file\_per\_table=1  lower\_case\_table\_names=1  symbolic-links=0  #日志存放目录  [mysqld\_safe]  log-error = /usr/local/mysql/data/mysqld.log  pid-file = /usr/local/mysql/data/mysqld.pid  #允许时间类型的数据为零(去掉NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE)  sql\_mode=ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_AUTO\_CREATE\_USER,NO\_ENGINE\_SUBSTITUTION  #ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_AUTO\_CREATE\_USER,NO\_ENGINE\_SUBSTITUTION |

lower\_case\_table\_names：是否区分大小写，1表示存储时表名为小写，操作时不区分大小写；0表示区分大小写；不能动态设置，修改后，必须重启才能生效：

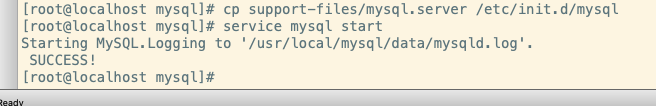
character\_set\_server：设置数据库默认字符集，如果不设置默认为latin1

innodb\_file\_per\_table：是否将每个表的数据单独存储，1表示单独存储；0表示关闭独立表空间，可以通过查看数据目录，查看文件结构的区别；

## 测试启动mysql服务器

先将mysql.server放置到/etc/init.d/mysql中 【把启动脚本放到开机初始化目录】

|  |
| --- |
| cp /usr/local/mysql/support-files/mysql.server /etc/init.d/mysql |

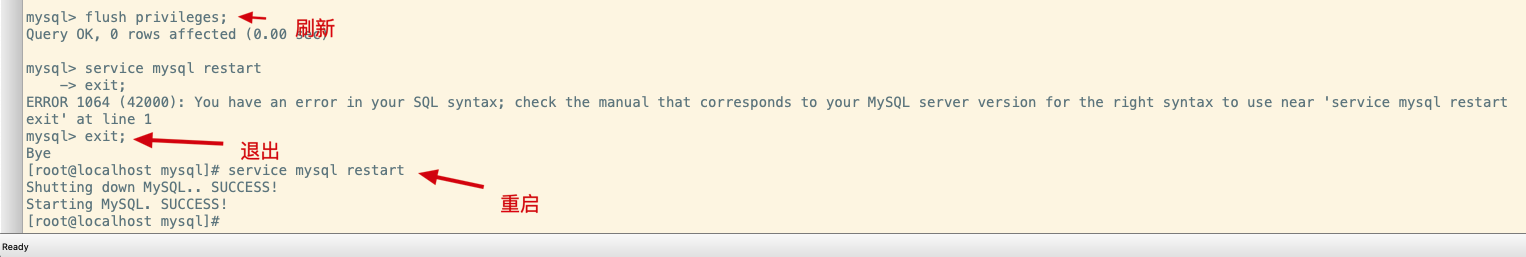


启动： service mysql start

修改密码：

|  |
| --- |
| 进入mysql并更改密码  cd /usr/local/mysql-5.7.28  ./bin/mysql -u root -p  mysql> set password=password('123456');  mysql> grant all privileges on \*.\* to root@'%' identified by '123456';  mysql> flush privileges;  添加远程访问权限  mysql> use mysql;  mysql> update user set host='%' where user = 'root';  mysql> flush privileges;  如果更改时报错：ERROR 1062 (23000): Duplicate entry '%-root' for key 'PRIMARY'，就先查询一下是否已更改，最后执行刷新。    重启mysql生效  service mysql restart |





## 两种启动方式:

1,没有加入开机自启，需要进入到bin目录中去启动

[root@localhost /]# /usr/local/mysql/support-files/mysql.server start

2，已经加入了开机自启

service mysql start

# 使用终端进行用户操作

用root账号登录服务器，以39为例

## 登录：

mysql -u root -p

密码: mysql@dev

## 创建数据库：

创建一个数据库名为 spdjlyy的数据库。

|  |
| --- |
| CREATE DATABASE `spdjlyy` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4\_general\_ci; |

## 为数据库分配其他用户

创建一个用户【省略】

查看我们已经有什么用户，把这个用户绑定到我们的数据库上。

use mysql;

show tables;

select \* from user;

比如你的user表中有个bkgxapp的账号。

## 分配账号

grant all privileges on spdjlyy.\* to 'bkgxapp'@'%' identified by '2020#all^APP';

改动用户名和密码即可

flush privileges;

刷新权限

grant create,alter,drop,select,insert,update,delete on spdjlyy.\* to 'bkgxapp'@'%';

为用分配数据库操作的增删改查权限

flush privileges;

刷新权限