1. Mysql需要 在MySQLServer8.0文件夹中自己编写配置文件my.ini

**（在cmd的mysql命令中，不区分大小写，但是需要用“;”结束命令）**

1. **本机用户名：**MySQL3306

**端口**：3306

**IP地址：**192.168.1.105**（会变化）**

**密码：**1230re0321re

1. 安装mysql命令到系统：

cmd管理员模式，进入mysql server的bin目录，

命令：mysql –install

1. 查看系统用户目录信息:

SELECT DISTINCT CONCAT('User: ''',user,'''@''',host,''';') AS query FROM mysql.user;

或者：

select host,user,plugin from user;

**（注：**plugin---查看密码加密方式）

1. **常用命令：**

**对mysql服务器的操作命令(cmd使用管理员模式运行)**

初始化数据库：mysqld --initialize-insecure --user=mysql

开启服务器：net start mysql （若无法启动，则到服务中手动开启）

关闭服务器：net stop mysql

**登陆的操作命令**

windows登陆本地用户：mysql –u root –p

本地远程登陆：mysql -h **服务器IP** -P 3306**(端口)** -u root**(用户名)** -p

创建用户（由root权限----host=%）：

create user ‘chris’identified by '1230re0321re';

删除用户：Delete FROM user Where User='test' and Host='localhost';

修改用户名：update user set user="新用户名" where user="旧用户名";

flush privileges;

查看当前登陆的用户名：select user();

**对数据库的操作命令**

显示本地mysql所有的数据库：show databases;

选择使用的数据库：use mysql

显示当前数据库中的表格：show tables;

**对表格中的数据操作命令**

显示当前表格中的数据（显示两行）：select \* from bookinfo limit 2;

删除某张表中的所有数据：delete from 表格名字;

查找表格中有多少条数据：select count(\*) from 表格名字;

条件查找：select count(\*) from 表格名字 where （条件，eg：id>300）;

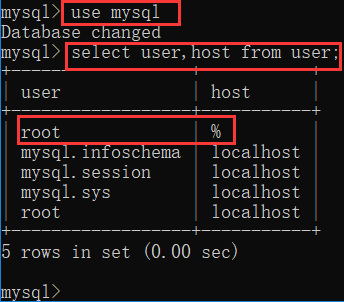
1. **本地远程登陆：（cmd使用管理员权限打开）——root用户远程登陆**

登陆本数据库的root用户（注意root用户的权限应为root，即：host=%）

注：**%：**代表由远程权限

**localhost：**代表本地

(**电脑的IP地址是会变化的**，有时候需要使用**ipconfig查看本机IP**地址，再使用对应的IP地址登陆MySQL)

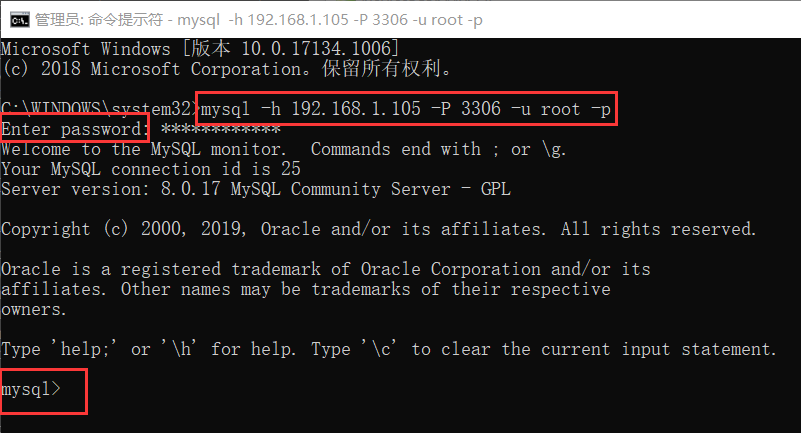


**命令**：本地远程登陆：mysql -h **服务器ip地址** -P 3306 -u **root** -p

**eg**：本地远程登陆：mysql -h **192.168.1.105** -P 3306 -u **root** -p

（注：**本机IP获取**：cmd中用ipconfig，得到网卡的ipv4地址就是。）

**登陆成功的样子：**



1. **本地远程登陆：（自己创建的用户）**

root登陆mysql之后：

**命令：**

**（注：mysql8.0将创建用户、权限设定的命令分离了，所以需要步骤1-2）**

1. **创建用户：**

create user '用户名'@'访问主机' identified by '密码';

**eg:** create user 'test'@'%' identified by '1230re0321re';

1. **赋予权限：（账户可以使用的数据库）**

grant 权限列表 on 数据库 to '用户名'@'访问主机' ;

**eg:** grant all privileges on \*.\* to 'test'@'%' ;

(修改权限时在后面加with grant option)

**（步骤3-4：修改密码加密方式）**

**（远程登陆的账号其密码加密方式为：mysql\_native\_password）**

1. alter user '用户名'@'%' identified by '密码' password expire never;

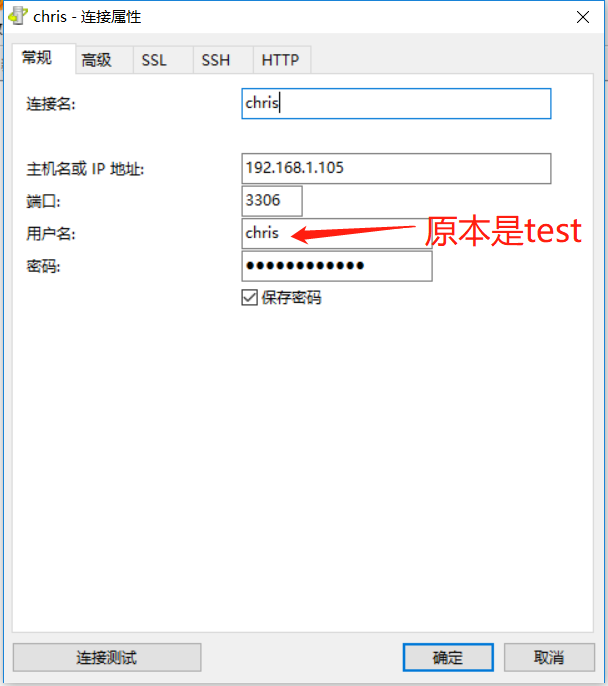
**eg：**alter user 'test'@'%' identified by '1230re0321re' password expire never;

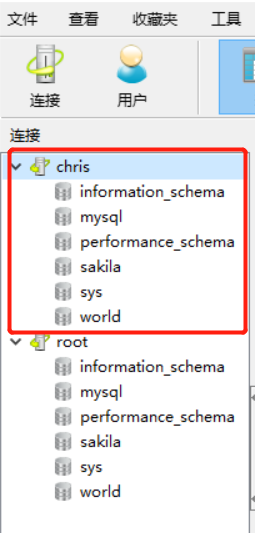
1. alter user '用户名'@'%' identified by with mysql\_native\_password by '密码';

**eg：**alter user 'test'@'%' identified with mysql\_native\_password by '1230re0321re';

1. flush privileges; （修改内容，立即生效-----刷新权限）

**登陆成功的样子：**





1. **windows10启用端口3306的教程：**

<https://blog.csdn.net/yj19961126/article/details/81224107>

1. **关于MySQL的重要说明：**

* MySQL中表格名字、数据库名，都不区分大小写
* 表格中的参数变量名，区分大小写
* 在进行模糊查询的时候，可以使用like，like需要结合%使用。
* '%a' ：以a结尾的数据 （%：表示匹配任意个字符）
* 'a%' ：以a开头的数据
* '%a%' ：含有a的数据
* '\_a\_' ：三位且中间字母是a的 （\_：表示匹配一个字符）
* '\_a' ：两位且结尾字母是a的
* 'a\_' ：两位且开头字母是a的

**eg：**

String sql = SELECT \* from table WHERE author LIKE '%chris'

* **SQL注入：**

SQL注入，就是通过把SQL命令插入到Web表单递交或输入域名或页面请求的查询字符串，最终达到欺骗服务器执行恶意的SQL命令。

我们永远不要信任用户的输入，我们必须认定用户输入的数据都是不安全的，需要**对用户输入的数据进行过滤处理**。

**eg：**

过滤用户输入的名字，规定只能使用数字、字母、下划线，长度在8-20

if (preg\_match("/^\w{8,20}$/", $\_GET['username'], $**matches**))

{

$result = mysqli\_query($conn, "SELECT \* FROM users

WHERE username=$**matches**[0]");

}

else

{

echo "username 输入异常";

}



# 常用的SQL语句

1. **创建表：**

/\*

创建商品信息表

\*/

**create table** **`product\_info`** ( **--注意：表格名，属性名用的``，中文用’’**

**`product\_id`** varchar(32) not null **AUTO\_INCREMENT**, --**自增**

**`product\_price`** decimal(8,2) not null comment **'单价'**,

**`product\_stock`** int not null comment '库存', **--值：不能为空**

**`create\_time`** **timestamp** not null **default** current\_timestamp comment '创建时间',  **--时间戳、默认值**

**`update\_time`** **timestamp** not null **default** current\_timestamp **on update** current\_timestamp comment '修改时间',

**`type`** mediumint(1) **NOT NULL**,

**primary key** (**`product\_id`**) **--创建主键：id**

**KEY** **`index\_type`** (**`type`**) **--创建索引检索名：index\_type**

) **comment** '商品表'; **--表属性中注释名：商品表**

1. **primary key、unique key、foreign key、index的区别：**

（除了index之外，**其他的key都具有约束、索引的作用，只要是key都会在对应的key上建立一个index**）

1. primary key：主键，一张表只能有一个，且不能为null
2. unique key：唯一键，规范数据的唯一性，建立唯一索引，一张表可以有多个索引
3. foreign key：外部键，规范数据引用完整性，在key上建立一个index
4. index：索引，辅助查询，创建index时会在innodb表空间中创建一个类似目录的结构存储。不会约束索引字段的行为。

**eg：创建index索引**

create table t(id int,**unique index** inx\_tx\_id (id)); **index当作key使用**