

# 一、搭建 MySQL 数据库服务器

## P1 构建 MySQL 服务

环境准备

1. 创建CentOS系统虚拟机1台
2. 关闭firewalld
3. 禁用SELinux
4. 配置yum 源
5. 配置IP地址192.168.4.50
6. 软件 mysql-5.7.17.tar

官网地址 <http://dev.mysql.com/downloads/mysql>

更改主机名为 host50

```
[root@web1 ~]# hostname host50
[root@web1 ~]# echo host50 > /etc/hostname
```

安装软件包

```
[root@host50 ~]# tar -xvf mysql-5.7.17.tar
[root@host50 ~]# yum -y install mysql-community-*.rpm
[root@host50 ~]# systemctl restart mysqld      #启动服务
[root@host50 ~]# systemctl enable mysqld
[root@host50 ~]# ss -antlp | grep 3306
```

连接数据库，使用初始密码登录

```
[root@host50 ~]# grep password /var/log/mysqld.log
[root@host50 ~]# mysql -uroot -p'qg1wpZ;G+deg'
mysql> show databases;      #报错，需要重置密码
mysql> alter user root@"localhost" identified by "123qqq...A";
mysql> show databases;      #成功
mysql> exit
Bye
[root@host50 ~]# mysql -uroot -p123qqq...A
```

修改密码策略

查找密码相关的策略

```
mysql> show variables like "%password%";      #过滤密码字样的，%匹配 0 个或者多个
...
| validate_password_policy                      | MEDIUM |
```

策略名称	验证方式
LOW (0)	长度
<b>MEDIUM (1) 默认</b>	长度; 数字, 小写/大写, 和特殊字符
STRONG (2)	长度; 数字, 小写/大写和特殊字符; 字典文件

```
mysql> set global validate_password_policy=0;      #修改密码策略
```

```
mysql> set global validate_password_length=6;      #修改密码长度
```

```
mysql> alter user root@"localhost" identified by "tarena";
```

永久修改密码策略

```
[root@host50 ~]# vim /etc/my.cnf
```

```
[mysqld]
```

```
validate_password_policy=0
```

```
validate_password_length=6
```

```
[root@host50 ~]# systemctl restart mysqld
```

```
[root@host50 ~]# mysql -uroot -ptarena
```

```
mysql> alter user root@"localhost" identified by "123456";
```

```
mysql> exit
```

```
[root@host50 ~]# mysql -uroot -p123456      #成功登陆
```

## 二、数据库基础

### P1 连接 MySQL 服务

#### 连接方式

- 客户端连接MySQL服务的方法

- ✓ 命令行
- ✓ web页面
- ✓ 安装图形软件
- ✓ 编写脚本 (php、Java、python ……)

- 使用 mysql 命令

- mysql -h服务器IP -u用户名 -p密码 [数据库名]
- quit 或 exit 退出

```
[root@host50 ~]# mysql -uroot -p123456 mysql
```

```
mysql> select database();      #此命令可以查看当前处于哪个数据库下
```

## 数据存储流程

---

- 客户端把数据存储到数据库服务器上的步骤

- ✓ 连接数据库服务器
- ✓ 建库 //类似于文件夹
- ✓ 建表 //类似于文件
- ✓ 插入记录 //类似于文件内容
- ✓ 断开连接

## MySQL管理环境

---

- SQL命令使用规则

- ✓ SQL命令不区分字母大小写（密码、变量值除外）
- ✓ 每条SQL命令以；结束
- ✓ 默认命令不支持Tab键自动补齐
- ✓ \c 终止sql命令

## SQL命令分类

---

- 常用的SQL命令分类

- ✓ 管理数据库使用SQL（结构化查询语言）
- ✓ DDL 数据定义语言 如：create、alter、drop
- ✓ DML 数据操作语言 如：insert、update、delete
- ✓ DCL 数据控制语言 如：grant、revoke
- ✓ DTL 数据事务语言 如：commit、rollback、savepoint

## P2 MySQL 基本操作

## • 库 类似于文件夹，用来存储表

- ✓ 可以创建多个库，通过库名区分
- ✓ show databases; //显示已有的库
- ✓ select user(); //显示连接用户
- ✓ use 库名; //切换库
- ✓ select database(); //显示当前所在的库
- ✓ create database 库名; //创建新库
- ✓ show tables; //显示已有的表
- ✓ drop database 库名; //删除库

## • 库名 命名规则

- ✓ 仅可以使用数字、字母、下划线、不能纯数字
- ✓ 区分字母大小写，具有唯一性
- ✓ 不可使用指令关键字、特殊字符

```
mysql> create database bbsdb;      #创建库
mysql> create database BBSDB;
mysql> show databases;           #查看库
mysql> drop database BBSDB;       #删除库

mysql> use bbsdb;                #切换库
mysql> select database();         #显示当前所在库
mysql> select user();             #显示连接用户
mysql> show tables;              #显示已有的表

创建表
mysql> create table bbsdb.user(name char(10), age int, homedir char(20));
mysql> show tables;
mysql> desc user;                #查看表结构

存储数据
mysql> insert into bbsdb.user values("bob", 19, "USA");
mysql> insert into bbsdb.user values("tom", 21, "beijing");

查看数据
mysql> select * from bbsdb.user;  #查看所有数据
mysql> select name from bbsdb.user; #只查看 name 字段
mysql> select name,age from bbsdb.user; #只查看 name 和 age 字段
```

### 更改数据

```
mysql> update bbsdb.user set homedir="china";
```

```
mysql> select * from bbsdb.user;
```

### 删除数据

```
mysql> delete from bbsdb.user;
```

```
mysql> show tables; #表依旧存在，但是数据没有
```

```
mysql> drop table bbsdb.user; #删除表
```

```
mysql> show tables;
```

### 表中插入中文字符

```
mysql> show create table bbsdb.t1 \G; #查看建表命令,\G 竖着来显示数据 ###默认表的字符集 CHARSET 为 latin1，不能在表中插入中文
```

```
mysql> create table 学生表(姓名 char(15), 地址 varchar(50)) DEFAULT CHARSET=utf8; #设置字符集
```

```
mysql> desc 学生表;
```

```
mysql> insert into 学生表 values("张三丰", "武当山");
```

```
mysql> select * from 学生表;
```

### 课后练习：

创建一个数据库名为 mydb

在 mydb 库下创建一个前任信息表（表头个数，内容及类型自己考虑）

在 mydb 库下创建一个存储游戏任务信息表（表头个数，内容及类型自己考虑）