



























1. [案例1：数据导入导出](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/RDBMS1/DAY03/CASE/01/index.html#case1)
2. [案例2：管理表记录](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/RDBMS1/DAY03/CASE/01/index.html#case2)
3. [案例3： 匹配条件](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/RDBMS1/DAY03/CASE/01/index.html#case3)
4. [案例4： MySQL管理工具](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/RDBMS1/DAY03/CASE/01/index.html#case4)

**1 案例1：数据导入导出**

**1.1 问题**

* 修改检索目录为/myload
* 将/etc/passwd文件导入db3库的user表里，并添加行号字段。
* 将db3库user表所有记录导出, 存到/myload/user.txt 文件里。

**步骤一：修改检索目录为/myload**

1）修改配置文件，重启服务

1. **]**# mkdir **/**myload
2. **]**# chown mysql **/**myload
3. **]**# vim **/**etc**/**my**.**cnf
4. **[**mysqld**]**
5. secure\_file\_priv**=**"/myload”
6. :wq
7. ]# systemctl restart mysqld
8. mysql> show variables like “secure\_file\_priv”; //查看
9. +------------------+-----------------------+
10. | Variable\_name | Value |
11. +------------------+-----------------------+
12. | secure\_file\_priv | /myload/ |
13. +------------------+-----------------------+
14. Mysql>

2）新建db3库、user表

1. **[**root@dbsvr1 **~]**# mysql **-**u root –p123456
2. mysql**>** CREATE DATABASE db3**;**
3. create table db3**.**user**(**
4. name char**(50),**
5. password char**(1),**
6. uid int**,**
7. gid int**,**
8. comment char**(150),**
9. homedir char**(50),**
10. shell char**(50)**
11. **);**
12. Query OK**,** **0** rows affected **(0.70** sec**)**
13. Mysql**>**

**步骤二：将/etc/passwd文件导入db3库的user表里，并添加行号字段。**

1）拷贝文件到检索目录下

1. **[**root@dbsvr1 **~]**#
2. **[**root@dbsvr1 **~]**# cp **/**etc**/**passwd **/**myload**/**

2）导入数据

1. **[**root@dbsvr1 **~]**# mysql –uroot –ptarena
2. mysql**>** load data infile "/myload/passwd" into table db3**.**user
3. fields terminated by ":" lines terminated by "**\n**" **;** //导入数据
4. mysql**>** select **\*** from db3**.**user**;** //查看表记录
5. mysql**>** alter table db3**.**user
6. **->** add
7. **->** id int primary key auto\_increment first**;** //添加行号id 字段
8. mysql**>** select **\*** from db3**.**user**;** //查看表记录

**步骤三：将db3库user表所有记录导出, 存到/myload/user.txt 文件里。**

1）查询要导出的数据

1. mysql**>** select **\*** from db3**.**user ；

2）导出数据

1. mysql**>** select **\*** from db3**.**user into outfile "/myload/user1.txt"**;**

3）查看文件内容

1. **]**# cat **/**myload**/**user1**.**txt

**2 案例2：管理表记录**

**2.1 问题**

练习表记录的操作

1. 练习插入表记录
2. 练习更新表记录
3. 练习查询表记录
4. 练习删除表记录

**2.2 步骤**

实现此案例需要按照如下步骤进行。

**步骤一：练习插入表记录**

1）插入记录时，指定记录的每一个字段的值

这种情况下，不需要明确指出字段，但每条记录的值的顺序、类型都必须与表格结构向一致，否则可能无法正确插入记录。

比如，以下操作将向stu\_info表插入3条表记录：

1. mysql**>** INSERT stu\_info VALUES
2. **->** **(**'Jim'**,**'girl'**,24),**
3. **->** **(**'Tom'**,**'boy'**,21),**
4. **->** **(**'Lily'**,**'girl'**,20);**
5. Query OK**,** **3** rows affected **(0.15** sec**)**
6. Records**:** **3** Duplicates**:** **0** Warnings**:** **0**

完成插入后确认表记录：

1. mysql**>** SELECT **\*** FROM stu\_info**;**
2. **+------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Tom **|** boy **|** **21** **|**
7. **|** Lily **|** girl **|** **20** **|**
8. **+------+--------+-----+**
9. **3** rows **in** set **(0.00** sec**)**

**2）插入记录时，只指定记录的部分字段的值**

这种情况下，必须指出各项值所对应的字段；而且，未赋值的字段应设置有默认值或者有自增填充属性或者允许为空，否则插入操作将会失败。

比如，向stu\_info表插入Jerry的年龄信息，性别为默认的“boy”，自动编号，相关操作如下：

1. mysql**>** INSERT INTO stu\_info**(**name**,**age**)**
2. **->** VALUES**(**'Jerry'**,27);**
3. Query OK**,** **1** row affected **(0.04** sec**)**

类似的，再插入用户Mike的年龄信息：

1. mysql**>** INSERT INTO stu\_info**(**name**,**age**)**
2. **->** VALUES**(**'Mike'**,21);**
3. Query OK**,** **1** row affected **(0.05** sec**)**

确认目前stu\_info表的所有记录：

1. mysql**>** SELECT **\*** FROM stu\_info**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Tom **|** boy **|** **21** **|**
7. **|** Lily **|** girl **|** **20** **|**
8. **|** Jerry **|** boy **|** **27** **|**
9. **|** Mike **|** boy **|** **21** **|**
10. **+-------+--------+-----+**
11. **5** rows **in** set **(0.00** sec**)**

3）更新表记录时，若未限制条件，则适用于所有记录

将stu\_info表中所有记录的age设置为10：

1. mysql**>** UPDATE stu\_info SET age**=10;**
2. Query OK**,** **5** rows affected **(0.04** sec**)**
3. Rows matched**:** **5** Changed**:** **5** Warnings**:** **0**

确认更新结果：

1. mysql**>** SELECT **\*** FROM stu\_info**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jim **|** girl **|** **10** **|**
6. **|** Tom **|** boy **|** **10** **|**
7. **|** Lily **|** girl **|** **10** **|**
8. **|** Jerry **|** boy **|** **10** **|**
9. **|** Mike **|** boy **|** **10** **|**
10. **+-------+--------+-----+**
11. **5** rows **in** set **(0.00** sec**)**

4）更新表记录时，可以限制条件，只对符合条件的记录有效

将stu\_info表中所有性别为“boy”的记录的age设置为20：

1. mysql**>** UPDATE stu\_info SET age**=20**
2. **->** WHERE gender**=**'boy'**;**
3. Query OK**,** **3** rows affected **(0.04** sec**)**
4. Rows matched**:** **3** Changed**:** **3** Warnings**:** **0**

确认更新结果：

1. mysql**>** SELECT **\*** FROM stu\_info**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jim **|** girl **|** **10** **|**
6. **|** Tom **|** boy **|** **20** **|**
7. **|** Lily **|** girl **|** **10** **|**
8. **|** Jerry **|** boy **|** **20** **|**
9. **|** Mike **|** boy **|** **20** **|**
10. **+-------+--------+-----+**
11. **5** rows **in** set **(0.00** sec**)**

5）删除表记录时，可以限制条件，只删除符合条件的记录

删除stu\_info表中年龄小于18的记录：

1. mysql**>** DELETE FROM stu\_info WHERE age **<** **18;**
2. Query OK**,** **2** rows affected **(0.03** sec**)**

确认删除结果：

1. mysql**>** SELECT **\*** FROM stu\_info**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Tom **|** boy **|** **20** **|**
6. **|** Jerry **|** boy **|** **20** **|**
7. **|** Mike **|** boy **|** **20** **|**
8. **+-------+--------+-----+**
9. **3** rows **in** set **(0.00** sec**)**

6）删除表记录时，如果未限制条件，则会删除所有的表记录

删除stu\_info表的所有记录：

1. mysql**>** DELETE FROM stu\_info**;**
2. Query OK**,** **3** rows affected **(0.00** sec**)**

确认删除结果：

1. mysql**>** SELECT **\*** FROM stu\_info**;**
2. Empty set **(0.00** sec**)**

**3 案例3： 匹配条件**

**3.1 问题**

具体要求如下：

* 练习数值比较的使用
* 练习字符比较的使用
* 练习逻辑比较的使用
* 练习模糊匹配的使用
* 练习正则匹配的使用
* 练习查询结果分组、排序、过滤、限制显示记录行数
* 练习聚集函数的使用
* 练习四则运算的使用步骤

实现此案例需要按照如下步骤进行。

**3.2 步骤一：匹配条件练习**

1）常用的表记录统计函数

查询stu\_info表一共有多少条记录（本例中为5条）：

1. mysql**>** SELECT count**(\*)** FROM stu\_info**;**
2. **+----------+**
3. **|** count**(\*)** **|**
4. **+----------+**
5. **|** **5** **|**
6. **+----------+**
7. **1** row **in** set **(0.00** sec**)**

计算stu\_info表中各学员的平均年龄、最大年龄、最小年龄：

1. mysql**>** SELECT avg**(**age**),**max**(**age**),**min**(**age**)** FROM stu\_info**;**
2. **+----------+----------+----------+**
3. **|** avg**(**age**)** **|** max**(**age**)** **|** min**(**age**)** **|**
4. **+----------+----------+----------+**
5. **|** **22.6000** **|** **27** **|** **20** **|**
6. **+----------+----------+----------+**
7. **1** row **in** set **(0.00** sec**)**

计算stu\_info表中男学员的个数：

1. mysql**>** SELECT count**(**gender**)** FROM stu\_info WHERE gender**=**'boy'**;**
2. **+---------------+**
3. **|** count**(**gender**)** **|**
4. **+---------------+**
5. **|** **3** **|**
6. **+---------------+**
7. **1** row **in** set **(0.00** sec**)**

2）字段值的数值比较

列出stu\_info表中年龄为21岁的学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE age**=21;**
2. **+------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+------+--------+-----+**
5. **|** Tom **|** boy **|** **21** **|**
6. **|** Mike **|** boy **|** **21** **|**
7. **+------+--------+-----+**
8. **2** rows **in** set **(0.00** sec**)**

列出stu\_info表中年龄超过21岁的学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE age**>21;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Jerry **|** boy **|** **27** **|**
7. **+-------+--------+-----+**
8. **2** rows **in** set **(0.00** sec**)**

列出stu\_info表中年龄大于或等于21岁的学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE age**>=21;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Tom **|** boy **|** **21** **|**
7. **|** Jerry **|** boy **|** **27** **|**
8. **|** Mike **|** boy **|** **21** **|**
9. **+-------+--------+-----+**
10. **4** rows **in** set **(0.00** sec**)**

列出stu\_info表中年龄在20岁和24岁之间的学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE age BETWEEN **20** and **24;**
2. **+------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Tom **|** boy **|** **21** **|**
7. **|** Lily **|** girl **|** **20** **|**
8. **|** Mike **|** boy **|** **21** **|**
9. **+------+--------+-----+**
10. **4** rows **in** set **(0.00** sec**)**

3）多个条件的组合

列出stu\_info表中年龄小于23岁的女学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE age **<** **23** AND gender**=**'girl'**;**
2. **+------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+------+--------+-----+**
5. **|** Lily **|** girl **|** **20** **|**
6. **+------+--------+-----+**
7. **1** row **in** set **(0.00** sec**)**

列出stu\_info表中年龄小于23岁的学员，或者女学员的记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE age **<** **23** OR gender**=**'girl'**;**
2. **+------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Tom **|** boy **|** **21** **|**
7. **|** Lily **|** girl **|** **20** **|**
8. **|** Mike **|** boy **|** **21** **|**
9. **+------+--------+-----+**
10. **4** rows **in** set **(0.00** sec**)**

如果某个记录的姓名属于指定范围内的一个，则将其列出：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE name IN
2. **->** **(**'Jim'**,**'Tom'**,**'Mickey'**,**'Minnie'**);**
3. **+------+--------+-----+**
4. **|** name **|** gender **|** age **|**
5. **+------+--------+-----+**
6. **|** Jim **|** girl **|** **24** **|**
7. **|** Tom **|** boy **|** **21** **|**
8. **+------+--------+-----+**
9. **2** rows **in** set **(0.00** sec**)**

4）使用SELECT做数学计算

计算1234与5678的和：

1. mysql**>** SELECT **1234+5678;**
2. **+-----------+**
3. **|** **1234+5678** **|**
4. **+-----------+**
5. **|** **6912** **|**
6. **+-----------+**
7. **1** row **in** set **(0.00** sec**)**

计算1234与5678的乘积：

1. mysql**>** SELECT **1234\*5678;**
2. **+-----------+**
3. **|** **1234\*5678** **|**
4. **+-----------+**
5. **|** **7006652** **|**
6. **+-----------+**
7. **1** row **in** set **(0.00** sec**)**

计算1.23456789除以3的结果：

1. mysql**>** SELECT **1.23456789/3;**
2. **+----------------+**
3. **|** **1.23456789/3** **|**
4. **+----------------+**
5. **|** **0.411522630000** **|**
6. **+----------------+**
7. **1** row **in** set **(0.00** sec**)**

输出stu\_info表各学员的姓名、15年后的年龄：

1. mysql**>** SELECT name**,**age**+15** FROM stu\_info**;**
2. **+-------+--------+**
3. **|** name **|** age**+15** **|**
4. **+-------+--------+**
5. **|** Jim **|** **39** **|**
6. **|** Tom **|** **36** **|**
7. **|** Lily **|** **35** **|**
8. **|** Jerry **|** **42** **|**
9. **|** Mike **|** **36** **|**
10. **+-------+--------+**
11. **5** rows **in** set **(0.00** sec**)**

5）使用模糊查询，LIKE

以下划线 \_ 匹配单个字符，% 可匹配任意多个字符。

列出stu\_info表中姓名以“J”开头的学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE name LIKE 'J%'**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Jerry **|** boy **|** **27** **|**
7. **+-------+--------+-----+**
8. **2** rows **in** set **(0.00** sec**)**

列出stu\_info表中姓名以“J”开头且只有3个字母的学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE name LIKE 'J\_\_'**;**
2. **+------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **+------+--------+-----+**
7. **1** row **in** set **(0.00** sec**)**

6）使用正则表达式，REGEXP

列出stu\_info表中姓名以“J”开头且以“y”结尾的学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE name REGEXP '^J.\*y$'**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jerry **|** boy **|** **27** **|**
6. **+-------+--------+-----+**
7. **1** row **in** set **(0.00** sec**)**

效果等同于：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE name Like 'J%y'**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jerry **|** boy **|** **27** **|**
6. **+-------+--------+-----+**
7. **1** row **in** set **(0.00** sec**)**

列出stu\_info表中姓名以“J”开头或者以“y”结尾的学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE name REGEXP '^J|y$'**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Lily **|** girl **|** **20** **|**
7. **|** Jerry **|** boy **|** **27** **|**
8. **+-------+--------+-----+**
9. **3** rows **in** set **(0.00** sec**)**

效果等同于：

1. mysql**>** SELECT **\*** FROM stu\_info WHERE name Like 'J%' OR name Like '%y'**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Lily **|** girl **|** **20** **|**
7. **|** Jerry **|** boy **|** **27** **|**
8. **+-------+--------+-----+**
9. **3** rows **in** set **(0.00** sec**)**

7）按指定的字段排序，ORDER BY

列出stu\_info表的所有记录，按年龄排序：

1. mysql**>** SELECT **\*** FROM stu\_info ORDER BY age**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Lily **|** girl **|** **20** **|**
6. **|** Tom **|** boy **|** **21** **|**
7. **|** Jim **|** girl **|** **24** **|**
8. **|** Jerry **|** boy **|** **27** **|**
9. **+-------+--------+-----+**
10. **4** rows **in** set **(0.00** sec**)**

因默认为升序（Ascend）排列，所以上述操作等效于：

1. mysql**>** SELECT **\*** FROM stu\_info ORDER BY age ASC**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Lily **|** girl **|** **20** **|**
6. **|** Tom **|** boy **|** **21** **|**
7. **|** Jim **|** girl **|** **24** **|**
8. **|** Jerry **|** boy **|** **27** **|**
9. **+-------+--------+-----+**
10. **4** rows **in** set **(0.00** sec**)**

若要按降序（Descend）排列，则将ASC改为DESC即可：

1. mysql**>** SELECT **\*** FROM stu\_info ORDER BY age DESC**;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jerry **|** boy **|** **27** **|**
6. **|** Jim **|** girl **|** **24** **|**
7. **|** Tom **|** boy **|** **21** **|**
8. **|** Lily **|** girl **|** **20** **|**
9. **+-------+--------+-----+**
10. **4** rows **in** set **(0.00** sec**)**

8）限制查询结果的输出条数，LIMIT

查询stu\_info表的所有记录，只列出前3条：

1. mysql**>** SELECT **\*** FROM stu\_info LIMIT **3;**
2. **+------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+------+--------+-----+**
5. **|** Jim **|** girl **|** **24** **|**
6. **|** Tom **|** boy **|** **21** **|**
7. **|** Lily **|** girl **|** **20** **|**
8. **+------+--------+-----+**
9. **3** rows **in** set **(0.00** sec**)**

列出stu\_info表中年龄最大的3条学员记录：

1. mysql**>** SELECT **\*** FROM stu\_info GROUP BY age DESC LIMIT **3;**
2. **+-------+--------+-----+**
3. **|** name **|** gender **|** age **|**
4. **+-------+--------+-----+**
5. **|** Jerry **|** boy **|** **27** **|**
6. **|** Jim **|** girl **|** **24** **|**
7. **|** Tom **|** boy **|** **21** **|**
8. **+-------+--------+-----+**
9. **3** rows **in** set **(0.00** sec**)**

9）分组查询结果，GROUP BY

针对stu\_info表，按性别分组，分别统计出男、女学员的人数：

1. mysql**>** SELECT gender**,**count**(**gender**)** FROM stu\_info GROUP BY gender**;**
2. **+--------+---------------+**
3. **|** gender **|** count**(**gender**)** **|**
4. **+--------+---------------+**
5. **|** boy **|** **3** **|**
6. **|** girl **|** **2** **|**
7. **+--------+---------------+**
8. **2** rows **in** set **(0.00** sec**)**

列出查询字段时，可以通过AS关键字来指定显示别名，比如上述操作可改为：

1. mysql**>** SELECT gender AS '性别'**,**count**(**gender**)** AS '人数'
2. **->** FROM stu\_info GROUP BY gender**;**
3. **+--------+--------+**
4. **|** 性别 **|** 人数 **|**
5. **+--------+--------+**
6. **|** boy **|** **3** **|**
7. **|** girl **|** **2** **|**
8. **+--------+--------+**
9. **2** rows **in** set **(0.00** sec**)**

**4 案例4： MySQL管理工具**

**4.1 问题**

部署LAMP+phpMyAdmin平台

**4.2 方案**

* 安装httpd、mysql、php-mysql及相关包
* 启动httpd服务程序
* 解压phpMyAdmin包，部署到网站目录
* 配置config.inc.php，指定MySQL主机地址
* 浏览器访问、登录使用

**4.3 步骤**

实现此案例需要按照如下步骤进行。

**步骤一：准备软件的运行环境 lamp**

1. **[**root@mysql6**~]**# rpm **-**q httpd php php**-**mysql //检测是否安装软件包
2. 未安装软件包 httpd
3. 未安装软件包 php
4. 未安装软件包 php**-**mysql
5. **[**root@mysql6**~]**# yum **-**y install httpd php php**-**mysql //装包
6. **[**root@mysql6**~]**# systemctl start httpd //启动服务
7. **[**root@mysql6**~]**# systemctl enable httpd //设置开机自启
8. Created symlink from **/**etc**/**systemd**/**system**/**multi**-**user**.**target**.**wants**/**httpd**.**service to **/**usr**/**lib**/**systemd**/**system**/**httpd**.**service**.**

**步骤二：测试运行环境**

1. **[**root@mysql6**~]**# vim **/var**/www/html**/**test**.**php //编辑页面测试文件
2. **[**root@mysql6**~]**# cat **/var**/www/html**/**test**.**php //查看页面测试文件
3. **<?**php
4. $x**=**mysql\_connect**(**"localhost"**,**"root"**,**"123456"**);**
5. **if(**$x**){** echo "ok"**;** **}else{** echo "no"**;** **};**
6. **?>**
7. **[**root@mysql6**~]**# yum **-**y install elinks //安装测试网页工具
8. **[**root@mysql6**~]**# elinks **--**dump http**:**//localhost/test.php
9. Ok //验证测试页面成功

**步骤三：安装软件包**

1）物理机传输解压包给虚拟机192.168.4.6

1. **[**root@room9pc桌面**]**# scp phpMyAdmin**-2.11.11-**all**-**languages**.**tar**.**gz **192.168.4.6:**/root/
2. root@**192.168.4.6**'s password:
3. phpMyAdmin-2.11.11-a 100% 4218KB 122.5MB/s 00:00

2）虚拟机192.168.4.6解压phpMyAdmin-2.11.11-all-languages.tar.gz压缩包

1. **[**root@mysql6**~]**# tar **-**zxf phpMyAdmin**-2.11.11-**all**-**languages**.**tar**.**gz **-**C **/var**/www/html**/** //-C 表示改变至目录
2. **[**root@mysql6**~]**# cd **/var**/www/html**/**
3. **[**root@mysql6**~]**# mv phpMyAdmin**-2.11.11-**all**-**languages phpmyadmin //改变目录名
4. **[**root@mysql6**~]**# chown **-**R apache**:**apache phpmyadmin**/** //改变phpmyadmin目录权限

**步骤四：修改软件的配置文件定义管理的数据库服务器**

切换到部署后的phpmyadmin程序目录，拷贝配置文件，并修改配置以正确指定MySQL服务器的地址

1. **[**root@mysql6html**]**# cd phpmyadmin
2. **[**root@mysql6 phpmyadmin**]**# cp config**.**sample**.**inc**.**php config**.**inc**.**php
3. //备份主配置文件
4. **[**root@mysql6 phpmyadmin**]**# vim config**.**inc**.**php //编辑主配置文件
5. **17** $cfg**[**'blowfish\_secret'**]** **=** 'plj123'**;** //给cookie做认证的值，可以随便填写
6. **31** $cfg**[**'Servers'**][**$i**][**'host'**]** **=** 'localhost'**;** //指定主机名，定义连接哪台服务器
7. **:**wq

**步骤五：在客户端访问软件 管理数据库服务器**

1）在客户端访问软件,打开浏览器输入http://192.168.4.6/phpmyadmin(数据库服务器地址) 访问软件，如图-1所示，用户名是root，密码是123456



图-1

2）登入成功后，如图-2示，即可在授权范围内对MySQL数据库进行管理。

/

图-2