

DBA基础

NSD DBA1

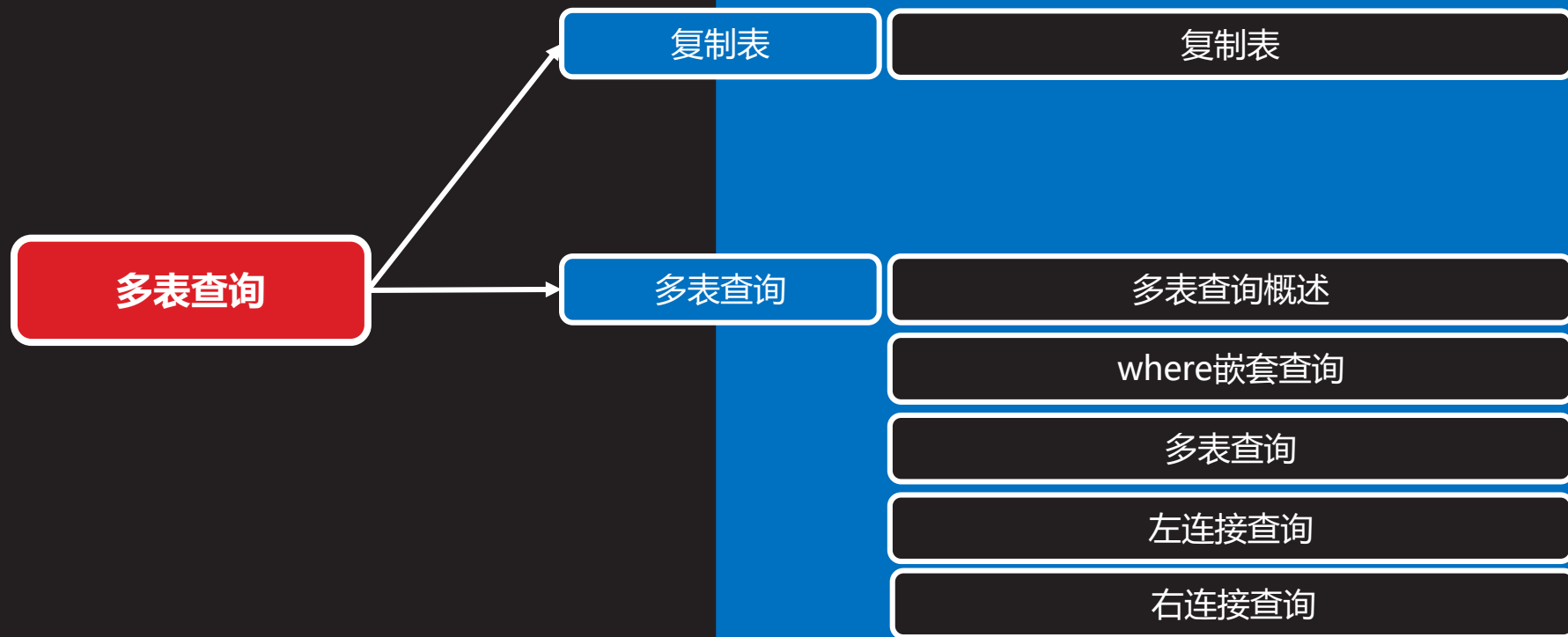
DAY04

内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	多表查询
	10:30 ~ 11:20	
	11:30 ~ 12:00	MySQL管理工具
下午	14:00 ~ 14:50	用户授权及撤销
	15:00 ~ 15:50	
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑



多表查询



复制表

复制表

- 将源表 xxx 复制为新表 yyy
 - CREATE TABLE yyy SELECT * FROM xxx;
- 将指定的查询结果复制为新表 zzz
 - CREATE TABLE zzz SQL查询语句;

```
mysql> CREATE TABLE yyuser SELECT * FROM mysql.user;
Query OK, 4 rows affected (0.02 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> CREATE TABLE zzuser
-> SELECT user,host,password FROM mysql.user;
Query OK, 4 rows affected (0.04 sec)
Records: 4  Duplicates: 0  Warnings: 0
```



复制表（续1）

- 复制源表 xxx 的结构到新表 vv
 - CREATE TABLE vv SELECT * FROM xxx WHERE FALSE;
- 将源表 vv 的名称改为 ww
 - ALTER TABLE vv RENAME TO ww;

```
mysql> CREATE TABLE vvuser SELECT * FROM mysql.user
      -> WHERE FALSE;
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> ALTER TABLE vvuser RENAME TO wwuser;
Query OK, 0 rows affected (0.00 sec)
```



多表查询

多表查询概述

- 多表查询，也称为 连接查询
 - 将2个或2个以上的表 按某个条件连接起来，从中选取需要的数据
 - 当多个表中 存在相同意义的字段（字段名可以不同）时，可以通过该字段连接多个表



多表查询

- 格式1
 - select 字段名列表 from 表a, 表b;
- 格式2
 - select 字段名列表 from 表a,表b where 条件;

以上格式的查询结果叫笛卡尔集
 查询结果的总条目数是 = (表a的记录数 x 表b的记录数)



where子查询

- 使用where子查询
 - 把内层查询结果作为外层查询的查询条件
- 语法格式
 - select 字段名列表 from 表名
where 条件
(select 字段名列表 from 表名 where 条件);
 - 输出年龄小于平均年龄的学生的名字和年龄

```
mysql> select name,age from student where age < (select avg(age)
from student);
```



左连接查询

- 基本用法
 - select 字段名列表 from
表a left join 表b on 条件表达式;

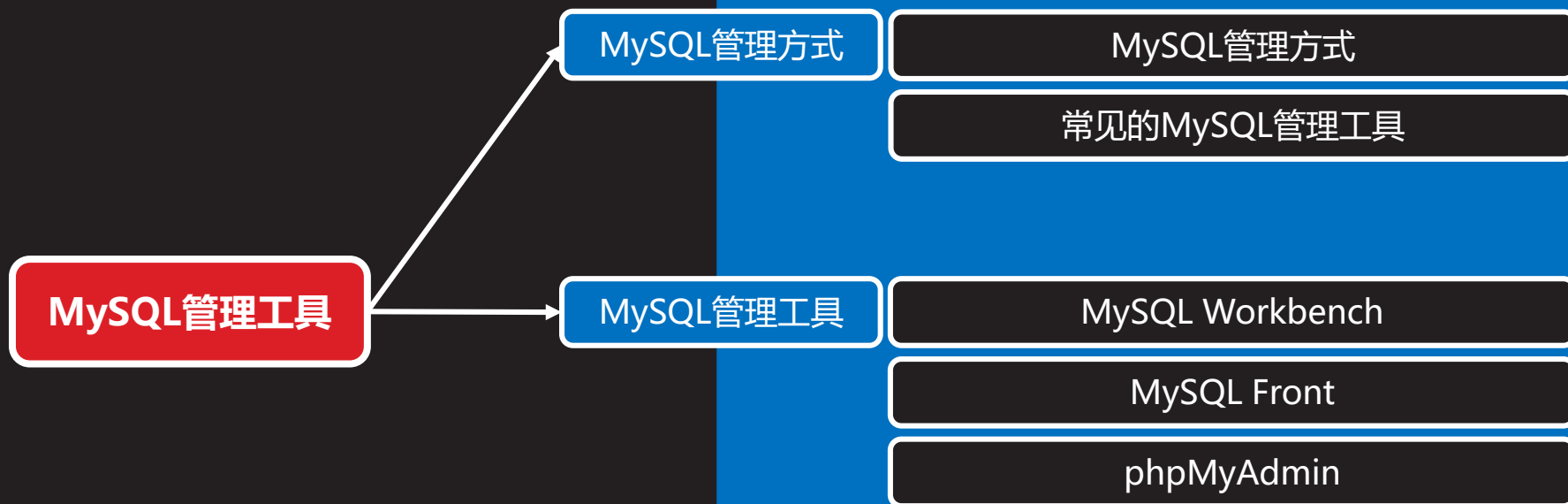


右连接查询

- 基本用法
 - select 字段名列表 from
表a right join 表b on 条件表达式;



MySQL管理工具



MySQL管理方式

MySQL管理方式

- mysql命令的局限性?
- 是否有图形化、更加直观的管理软件?
- 能通过Web界面访问吗?



常见的MySQL管理工具

- 常见的管理工具

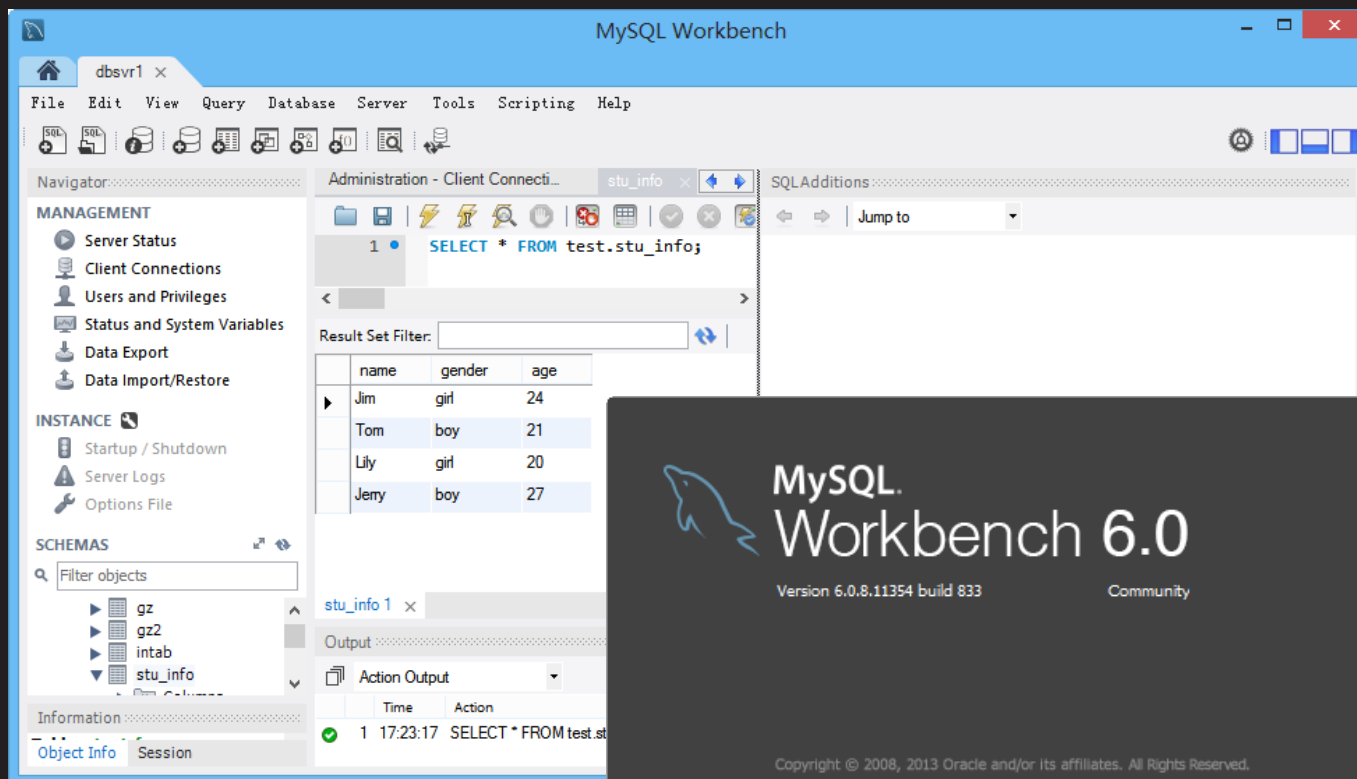
类 型	界 面	操作系统	说 明
mysql	命令行	跨平台	MySQL官方bundle包自带
MySQL-Workbench	图形	跨平台	MySQL官方提供
MySQL-Front	图形	Windows	开源，轻量级客户端软件
phpMyAdmin	浏览器	跨平台	开源，需LAMP平台
Navicat	图形	Windows	专业、功能强大，商业版



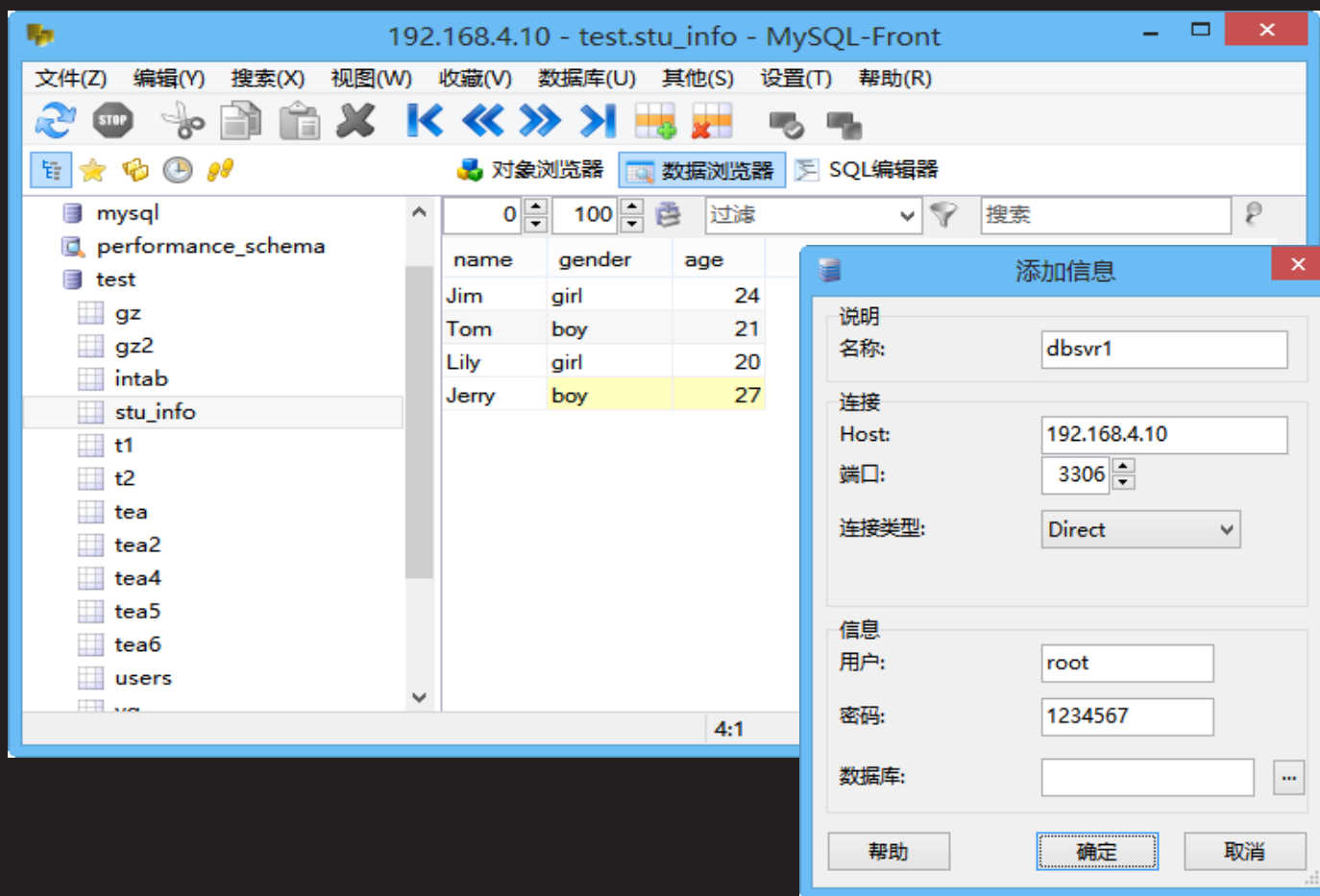
MySQL管理工具



MySQL Workbench



MySQL Front



phpMyAdmin

- 基本思路
 1. 安装httpd、mysql、php-mysql及相关包
 2. 启动httpd服务程序
 3. 解压phpMyAdmin包，部署到网站目录
 4. 配置config.inc.php，指定MySQL主机地址
 5. 创建授权用户
 6. 浏览器访问、登录使用



phpMyAdmin (续1)

```
# yum -y install httpd php php-mysql
# systemctl start httpd ; systemctl enabled httpd
```

```
# tar -zxf phpMyAdmin-2.11.11-all-languages.tar.gz -C /var/www/html/
# cd /var/www/html/
# mv phpMyAdmin-2.11.11-all-languages/ phpmyadmin
# chown -R apache:apache phpmyadmin/
# cp phpmyadmin/config.sample.inc.php phpmyadmin/config.inc.php
```

```
# sed -n '17p;31p' /var/www/html/phpmyadmin/config.inc.php
$cfg['blowfish_secret'] = 'plj123';
$cfg['Servers'][$i]['host'] = 'localhost';
```



phpMyAdmin (续2)

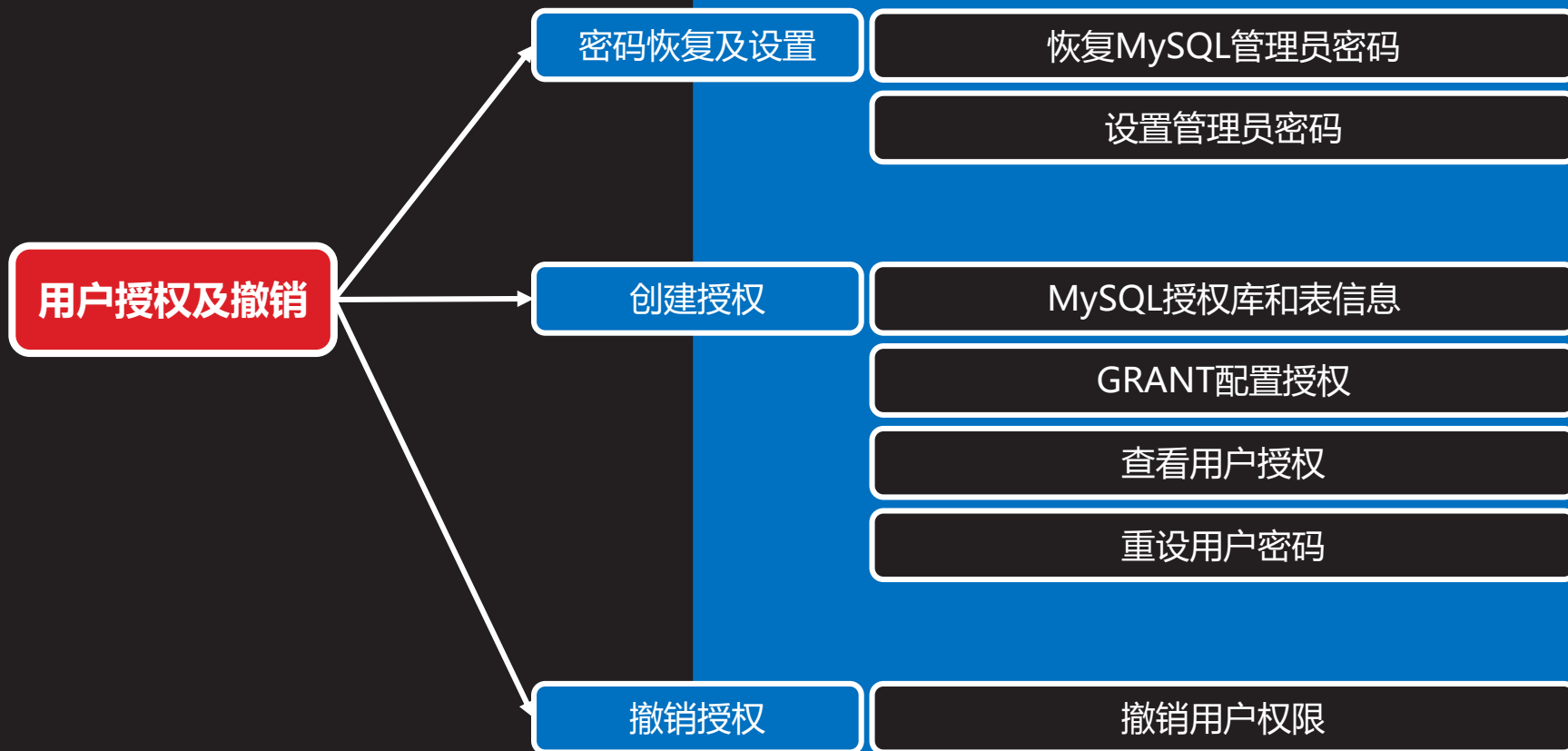


案例1：MySQL管理工具

- 部署LAMP+phpMyAdmin平台



用户授权及权限撤销



密码恢复及设置

恢复MySQL管理密码

- 密码忘了怎么办?

1. 停止MySQL服务程序

2. 跳过授权表启动MySQL服务程序

skip-grant-tables

写入/etc/my.cnf配置文件

3. 重设root密码 (更新user表记录)

4. 以正常方式重启MySQL服务程序



恢复MySQL管理密码（续1）

- 主要操作过程

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

```
.....
```

```
skip-grant-tables
```

```
::wq
```

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

```
[root@localhost ~]# mysql
```

```
[mysql> update mysql.user set authentication_string=password("888888")
```

```
-> where
```

```
-> user="root" and host="localhost";
```

```
mysql> flush privileges;
```

```
mysql> quit
```



恢复MySQL管理密码（续2）

- 主要操作过程

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

```
.....
```

```
#skip-grant-tables
```

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

```
[root@localhost ~]# mysql -uroot -p888888  
[mysql>
```



设置管理员密码

- 查看模块、修改配置文件、重启服务

```
[root@localhost ~]# find / -name "validate_password.so"  
/usr/lib64/mysql/plugin/debug/validate_password.so  
/usr/lib64/mysql/plugin/validate_password.so
```

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

```
plugin-load=validate_password.so
```

```
validate_password=FORCE_PLUS_PERMANENT
```

```
validate_password_policy=0
```

```
validate_password_length=6
```

//加载模块

//永久启用模块

//只验证密码长度

//指定密码长度

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



设置管理员密码（续1）

- 在Shell命令行修改登陆密码
 - 需要验证旧密码
 - 不适用于跳过授权表启动的情况

```
[root@localhost ~]# mysqladmin -uroot -p password "新密码"
```

```
Enter password: //输入旧密码
```

```
mysqladmin: [Warning] Using a password on the command line interface can be insecure.
```

```
Warning: Since password will be sent to server in plain text, use ssl connection to ensure password safety.
```

```
[root@localhost ~]#
```



案例2：密码恢复及设置

1. 恢复MySQL管理密码
2. 正常设置管理密码



创建授权

MySQL授权库和表信息

- 授权库 mysql , 主要的几个表
 - user表, 存储授权用户的访问权限
 - db表, 存储授权用户对数据库的访问权限
 - tables_priv表, 存储授权用户对表的访问权限
 - columns_priv表, 存储授权用户对字段的访问权限



GRANT配置授权

- 基本用法

- GRANT 权限列表... ON 库名.表名

- TO 用户名@'客户端地址'

- IDENTIFIED BY '密码'

设置密码

- [WITH GRANT OPTION];

是否有授权权限

- 注意事项

- 当库名.表名 为 *.* 时，匹配所有库所有表

- 授权设置存放在 mysql 库的 user 表



GRANT配置授权 (续1)

- 权限列表
 - all: 匹配所有权限
 - SELECT,UPDATE,INSERT
 - SELECT,UPDATE (字段1, , 字段N)



GRANT配置授权 (续2)

- 客户端地址
 - %: 匹配所有主机
 - 192.168.1.%: 匹配指定的一个网段
 - 192.168.1.1: 匹配指定IP地址的单个主机
 - %.tarena.com: 匹配一个DNS区域
 - svr1.tarena.com: 匹配指定域名的单个主机



GRANT配置授权 (续3)

- 应用示例
 - 新建用户mydba, 对所有库、表有完全权限
 - 允许从任何地址访问, 密码设为 s1cr2t!
 - 允许该用户为其他用户授权

```
mysql> GRANT all ON *.* TO mydba@'%'
      -> IDENTIFIED BY 's1cr2t!'
      -> WITH GRANT OPTION;
Query OK, 0 rows affected (0.09 sec)
```



GRANT配置授权 (续4)

- 应用示例
 - 授权sqler01用户，允许从192.168.4.0/24网段访问，对userdb库的user表有查询权限，登陆密码888888
 - 授权sqler02用户，允许从本机访问，允许对userdb库的任何表有查询/更新/插入/删除权限，密码为123456

```
mysql> grant select on userdb.user to
sqler01@"192.168.4.0/24" identified by "888888";
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

```
mysql> grant select,insert,update,delete on userdb.* to
sqler02@"localhost" identified by "123456";
Query OK, 0 rows affected, 1 warning (0.00 sec)
```



查看用户授权

- 用户查看自己的权限
 - SHOW GRANTS;

```
mysql> SHOW GRANTS\G
***** 1. row *****
Grants for root@localhost: GRANT ALL PRIVILEGES ON *.* TO 'root'
@'localhost' IDENTIFIED BY PASSWORD '*353C33BC20A4B4B2281F3DAAE9
01DBD0A5224E24' WITH GRANT OPTION
***** 2. row *****
Grants for root@localhost: GRANT PROXY ON ''@'' TO 'root'@'local
host' WITH GRANT OPTION
2 rows in set (0.00 sec)
```



查看用户授权 (续1)

- 管理员可查看其他用户的权限
 - SHOW GRANTS FOR 用户名@'客户端地址';

```
mysql> SHOW GRANTS FOR sqler02@'localhost'\G
***** 1. row *****
Grants for sqler02@localhost: GRANT USAGE ON *.* TO 'sqler02'@'localhost' IDENTIFIED BY PASSWORD '*6BB4837EB74329105EE4568DDA7DC67ED2CA2AD9'
***** 2. row *****
Grants for sqler02@localhost: GRANT SELECT, INSERT, UPDATE, DROP ON `test`.* TO 'sqler02'@'localhost'
2 rows in set (0.00 sec)
```



重设用户密码

- 授权用户连接后修改密码
 - `set password= password(“新密码”);`
- 管理员重置授权用户连接密码
 - `set password`
`for 用户名@'客户端地址'=passowrd(“新密码”);`



撤销授权



撤销用户权限

- 基本用法
 - REVOKE 权限列表 ON 库名.表名
FROM 用户名@"客户端地址";

```
mysql> REVOKE insert,drop ON test.* FROM sqler02@'localhost';
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> SHOW GRANTS FOR sqler02@'localhost'\G
***** 1. row *****
Grants for sqler02@localhost: GRANT USAGE ON *.* TO 'sqler02'@'localhost' IDENTIFIED BY PASSWORD '*6BB4837EB74329105EE4568DDA7DC67ED2CA2AD9'
***** 2. row *****
Grants for sqler02@localhost: GRANT SELECT, UPDATE ON `test`.* TO 'sqler02'@'localhost'
2 rows in set (0.00 sec)
```



案例3：用户授权及撤销

1. 允许root从192.168.4.0/24网段 访问，对所有库/表有完全权限，密码为tarena
2. 添加一个管理账号dba007，完全控制及授权
3. 撤销root从本机访问的权限，然后恢复
4. 允许webuser从任意客户机登录，只对webdb库有完全权限，密码为 888888
5. 撤销webuser的完全权限，改为查询权限



总结和答疑

撤销用户权限

问题现象

故障分析及排除

总结和答疑

撤销用户权限

问题现象

- 无法撤销用户的授权
 - 报错1: ERROR 1147 (42000)
 - 报错2: no such grant defined for user

```
mysql> revoke select on webdb.a from  
'webadmin'@'172.40.51.106';  
ERROR 1147 (42000): There is no such grant defined for user  
'webadmin' on host '172.40.51.106' on table 'a'  
mysql>
```



故障分析及排除

- 原因分析
 - 授权用户对目标没有权限
 - 只有给用户对目标库做过授权才可以撤销权限
- 解决办法
 - 查看用户的授权信息，看用户是否对库有权限

```
[root@dbsvr1 ~]# mysql -uroot -p123  
mysql> show grants for 'webadmin'@'172.40.51.106';
```

