服务器环境搭建顺序建议：git - mysql -Nginx-升级git-代码-PHP

sequel可视化数据库

ssh:

ssh localhost

sudo systemsetup -setremotelogin on

打开数据库/Applications/MAMP/Library/bin/mysql -u root –p

显示数据库: show databases 执行命令\g或\G或;

选择数据库:use databaseName

显示表:show tables

增加字段: alter table test add column name text

添加管理员账户grant all on \*.\* to ly@localhost identified by "lypwd"最后是密码

查询时间: select now() 查询用户: select user() 查询数据库版本: select version() 查询数据库: select database()

导入数据库:source:路径

复制表结构:create table test2 select \* from test where 1<>1;其中1<>1是false，这里是为了只复制结构而不拷贝数据

对表重新命名: alter table test2 rename as table2;

修改列类型: alter table table2 modify name int;

修改列名及其类型: alter table table2 change name sid int unsigned;

创建索引: alter table table2 add index ind\_id(sid);或create index ind\_id on table2(sid); 创建唯一索引: create unique index ind\_id on table2(sid);

删除索引:drop index idx\_id on table2;或alter table table2 drop index ind\_id;

拼接字符或者多个列: select concat(sid,1,name,'=') from table2;

连接: SELECT a.runoob\_id, a.runoob\_author, b.runoob\_count FROM runoob\_tbl a INNER JOIN tcount\_tbl b ON a.runoob\_author = b.runoob\_author;

判断是否为NULL：SELECT \* FROM tcount\_tbl WHERE runoob\_count = NULL;

事务: <?php $handler=mysqli\_connect("localhost","root","root"); mysqli\_select\_db($handler,"test\_ly"); mysqli\_query($handler,"SET AUTOCOMMIT=0");//设置为不自动提交，因为MYSQL默认立即执行

mysqli\_query($handler,"BEGIN");//开始事务定义

if(!mysqli\_query($handler,"insert into articles (name) values('2')")){

mysqli\_query($handler,"ROLLBACK");//判断当执行失败时回滚 } mysqli\_query($handler,"COMMIT");//执行事务 mysqli\_close($handler); ?>

指定字段添加的位置：ALTER TABLE testalter\_tbl ADD i INT AFTER c;

代码中PHP7获取数据库列表：$db\_list = mysqli\_query($con,"SHOW DATABASES");while ($db = mysqli\_fetch\_object($db\_list)){echo $db->Database . "<br />";}

代码中获取最后插入数据的ID（AUTO\_INCREMENT）： mysql\_insert\_id ($conn\_id);

转义防止注入echo mysqli\_real\_escape\_string($con,$es);

导出数据并格式化：SELECT \* FROM runoob\_tbl INTO OUTFILE '/Users/luoyuan/Documents/myResource/runoob.txt' FIELDS TERMINATED BY ',' ENCLOSED BY '"' LINES TERMINATED BY '\r\n';

操作服务器：(小于CentOS7的环境)

安装数据库：yum install mysql

ssh root@ip报错问题：REMOTE HOST IDENTIFICATION HAS CHANGE，进入提示文件，删除该ip下的信息

mysqld未被识别的服务的解决：yum -y install mysql-server

退出数据库exit

找不到mysql命令时： vim ~/.bash\_profile； export PATH=$PATH:/usr/local/mysql/bin； source ~/.bash\_profile；

sequel连接数据库时denied for user解决(如果只有初始密码：use mysql; SET PASSWORD FOR 'root'@'localhost' = PASSWORD('newpass');)：改密码：mysql> UPDATE user SET Password=PASSWORD('newpassword') where USER='root'; mysql> FLUSH PRIVILEGES; mysql> quit 如果ssh命令mysql时再出现该错误则mysql -u root –p <http://www.jb51.net/article/87580.htm>

ssh长久不操作断开解决：ClientAliveInterval 60 ClientAliveCountMax 86400

ServerAliveInterval 60 或ssh -o ServerAliveInterval=60 user@sshserver

用git部署代码：

<http://www.cnblogs.com/shaohuixia/p/5503521.html>

步骤：

进入服务器：

1添加用户useradd -m deployuser; 密码passwd deployuser

2新建部署代码的根目录mkdir /var/www/html/deploy

3将这个目录的属主和属组都改为上面新建的用户cd /var/www/html

chown deployuser:deployuser deploy

4 切换到部署代码的专用用户su deployuser

5 进入项目根目录，初始化为git仓库cd deploy git init

6 让仓库接受代码提交git config receive.denyCurrentBranch ignore

进入客户端:

1 通过 git clone 或 git pull 从 github 仓库上将代码获取到本地

2 将服务器添加到远程仓库列表git remote add testing ssh://deployuser@121.42.12.142/var/www/html/deploy/.git

3 将本地代码提交到测试服务器上面git push testing master

回到服务器端:

1 更新服务端 git 仓库状态并检出文件cd /var/www/html/deploy

git update-server-info git checkout –f

2 设置服务器端更新钩子cd .git/hooks mv post-receive.sample post-receive  
vim post-receive 添加如下内容:

#!/bin/sh

unset GIT\_DIR

cd ..

git checkout –f

配置nginx代码：<http://www.poluoluo.com/server/201505/361490.html>

1.用git上传代码,假设目录是codedir

2.下载nginx：将nginx安装包上传

3. cd nginx-1.11.12

4.安装依赖yum -y install gcc pcre-devel openssl openssl-devel

5. ./configure

6. make 和 make install

7. 启动nginx服务：/usr/local/nginx/sbin/nginx

8. 重启nginx服务：/usr/local/nginx/sbin/nginx -s reload 或sudo service nginx restart

9. vi /usr/local/nginx/conf/nginx.conf其中将server下location中的root改成codedir，index也改成codedir中的index.html，重启nginx

卸载nginx配置：删除/usr/local/nginx

杀死nginx进程的命令：killall -9 nginx

linux终端命令: cat -n model.html > model1.html把model的文档内容加上行号后输入model1这个文档里

cat /dev/null > model2.html清空model2文档的内容

chattr +i model.html防止系统中某个关键文件被修改 lsattr /etc/resolv.conf

查看保护属性

git将本地的状态还原和远程一样(远程没有的内容不会被删除)：git fetch origin, git pull

git不追踪某些文件：在.gitignore中添加相对路径(最好是在项目刚开始时就添加)。如果无效git rm -r --cached . 再重新添加

chgrp -v root html将html这个文件的所属组设置为root

权限：chmod ug+w,o-w file1.txt file2.txt 其中，u文件的拥有者；g与该文件的拥有者属于同个群组；o其他以外的人，a全部。+增加权限；-取消权限；=唯一设定权限。r读；w写；x执行

chmod -R a+r \*将目前目录下的所有文件与子目录皆设为任何人可读取

chmod 777 file其中777依次表示User、Group和Other的权限r=4,w=2,x=1

chown root:root html将html的拥有者设为root群体的使用者root

chown -R deployuser:deployuser \*将目前目录下的所有文件与子目录的拥有者皆设为 deployuser群体的使用者 deployuser

cksum model.html检查文件的CRC是否正确,输出344058071 1819 model.html，344058071是校验码，1819是字节数

比较两个文件是否相同cmp model1.html model.html,返回第一个不同

比较文件差异diff model.html model1.html -y -W 50其中-y表示并排显示，-W 50表示在使用-y参数时指定栏宽，其他参数见<http://www.runoob.com/linux/linux-comm-diff.html>

查看文件中的内容：cat deploy/model.html

显示文件类型：file -i model.html其中-i显示MIME类别，-b不显示文件名称

查找：find . -name "\*.c"将延伸档名是c的文件列出；find . -ctime -20将20天内更新过的文件列出; find . -type f列出普通文件

who|cut -b 3提取输出内容每一行的第三个字节

ln -s model.html model\_launch为model.html创建软链接(类似快捷方式)model\_launch; 创建硬链接(类似副本)ln model1.html model1\_launch

删除非空目录rm -fr [www.hnbwqr.xyz](http://www.hnbwqr.xyz)

复制：cp php-7.1.3.tar.bz2 ../

安装配置PHP(完成其他后最后来进行这一步)：

1.官网下载安装包，上传到服务器

2. yum install libxml2 yum install libxml2-devel -y：<http://www.jsjtt.com/xitongyingyong/linux/17.html> 否则提示libxml2错误

cd到解压出来的目录 ./configure --prefix=/usr/local/server/php --with-config-file-path=/etc --with-mysql=/usr/local/server/mysql --with-mysqli=/usr/local/server/mysql/bin/mysql\_config --with-mysql-sock=/server/mysql/mysql.sock --with-iconv --with-zlib --with-libxml-dir --enable-xml --disable-debug --disable-rpath --enable-discard-path --enable-bcmath --enable-shmop --enable-sysvsem --enable-inline-optimization --with-curl --with-curlwrappers --enable-mbregex --enable-fastcgi --enable-fpm --enable-mbstring --enable-ftp --with-openssl --enable-pcntl --enable-sockets --with-xmlrpc --enable-zip --enable-soap --without-pear --with-gettext --enable-session --with-mcrypt --enable-force-cgi-redirect

make 和 make install

3.启动php：/usr/local/sbin/php-fpm

出现already in use错误解决： netstat -lntup | grep 9000 killall php-fpm

找不到该文件就通过php-fpm.default复制一个。如果提示Nothing matches the include pattern '/usr/local/php/etc/php-fpm.d/\*.conf' from … cd到php-fpm.d目录同理复制一个。 vim [www.conf将user和group改为非root用户](http://www.conf将user和group改为非root用户), 再启动php-fpm

4. vi /usr/local/nginx/conf/nginx.conf

作如下改动：set $root\_path '/var/vhosts/www.hnbwqr.xyz/public'; root $root\_path; location / { index index.html index.htm index.php; try\_files $uri $uri/ /index.php?$query\_string; }

location ~ \.php$ {try\_files $uri =404; fastcgi\_pass 127.0.0.1:9000; fastcgi\_index index.php; fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name; include fastcgi.conf;}

5.重启nginx

服务器上git clone报错The requested URL returned error: 401 Unauthorized，解决(升级git)：<http://blog.csdn.net/z_dianjun/article/details/50819908>

yum install curl-devel expat-devel gettext-devel openssl-devel zlib-d

yum install gcc perl-ExtUtils-MakeMaker

wget <http://ftp.gnu.org/pub/gnu/libiconv/libiconv-1.14.tar.gz>

tar zxvf libiconv-1.14.tar.gz

cd libiconv-1.14

./configure --prefix=/usr/local/libiconv

make && make install

yum remove git

wget <https://github.com/git/git/archive/v2.2.1.tar.gz>

tar zxvf v2.2.1.tar.gz

cd git-2.2.1

make configure

./configure --prefix=/usr/local/git --with-iconv=/usr/local/libiconv

make all doc

make install install-doc install-html

echo "export PATH=$PATH:/usr/local/git/bin" >> /etc/bashrc

source /etc/bashrc

laravel访问报错Warning: require(/vendor/autoload.php) …

laravel下缺少vender这个依赖包:

先安装composer: 回到主机根目录curl -sS https://getcomposer.org/installer | php

如果报错openssl missing，--with-openssl

卸载php：yum remove php

另外rpm -qa | grep php 从下往上依次rpm -e php-fpm-5.3.3-22.el6.x86\_64或者 yum remove php-common.x86\_64

查找字符串：grep -rnR phpinfo /Users/luoyuan/Documents;

LNMP集成安装 <https://lnmp.org/install.html>

Linux连接数据库: mysql -uroot –p

phpMyAdmin配置：把下载的phpMyAdmin文件夹放在根目录(或laravel的public目录),将config.sample.inc.php改名为config.inc.php，在config.inc.php中

$cfg['servers'][$i]['auth\_type']='cookie'; $cfg['blowfish\_secret']='任意字符串做验证'

网址输入<http://localhost:8000/phpMyAdmin>：输入数据库用户名密码

创建多级目录: mkdir -p jai/contact/src/ 创建文件touch

mac装mysql：

1.官网用迅雷下载，不然很慢

2.记住安装后的密码

3.登陆进去后出现You must reset your password：SET PASSWORD = PASSWORD('hnb314159'); FLUSH PRIVILEGES; exit

defaults write com.apple.finder AppleShowAllFiles Yes && killall Finder //显示隐藏文件

laravel项目在服务器上只能访问欢迎页面不能访问其他页面，解决：

try\_files $uri $uri/ @rewrite; location @rewrite { rewrite ^/(.\*)$ /index.php?\_url=/$1; }

git撤销某个commit： git reset HEAD filename

squel连接不上远程服务器数据库：GRANT ALL PRIVILEGES ON \*.\* TO 'myuser'@'%' IDENTIFIED BY 'mypassword' WITH GRANT OPTION;

FLUSH PRIVILEGES;

只能子域名访问，或配置多个应用：.conf文件写多个server，删掉default\_server这句。如果用laravel，还要修改.env