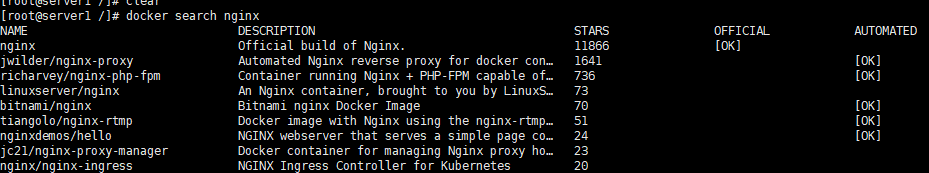
# Docker 安装 Nginx

### **docker pull nginx 命令安装**

查找 [Docker Hub](https://hub.docker.com/r/library/nginx/" \t "https://www.runoob.com/docker/_blank) 上的 nginx 镜像

[root@server1 /]# docker search nginx



这里我们拉取官方的镜像

docker pull nginx

等待下载完成后，我们就可以在本地镜像列表里查到 REPOSITORY 为 nginx 的镜像.

docker images nginx

以下命令使用 NGINX 默认的配置来启动一个 Nginx 容器实例：

[root@server1 /]# docker run --name nginxtest -p 8081:80 -d nginx

nginxtest为 容器名称。

the -d设置容器在在后台一直运行。

the -p 端口进行映射，将本地 8081 端口映射到容器内部的 80 端口。

执行以上命令会生成一串字符串，类似 **6e89d15a26fc73c397ea7173f2b223ddc4af36e6b925301a8389b62d496ce627**，这个表示容器的 ID，一般可作为日志的文件名

我们可以使用 docker ps 命令查看容器是否有在运行

[root@server1 /]# docker ps



PORTS 部分表示端口映射，本地的 8081 端口映射到容器内部的 80 端口。

在浏览器中打开 **http://192.168.224.11:8081/**，效果如下：



## nginx 部署

首先，创建目录 nginx, 用于存放后面的相关东西。

[root@server1 ~]# mkdir -p ~/nginx/www ~/nginx/logs ~/nginx/conf

拷贝容器内 Nginx 默认配置文件到本地当前目录下的 conf 目录，容器 ID 可以查看 **docker ps** 命令输入中的第一列

[root@server1 ~]# docker cp 6e89d15a26fc:/etc/nginx/nginx.conf ~/nginx/conf

* **www**: 目录将映射为 nginx 容器配置的虚拟目录。
* **logs**: 目录将映射为 nginx 容器的日志目录。
* **conf**: 目录里的配置文件将映射为 nginx 容器的配置文件。

[root@server1 conf]# docker run -d -p 8082:80 --name nginx-look-web -v ~/nginx/www:/usr/share/nginx/html -v ~/nginx/conf/nginx.conf:/etc/nginx/nginx.conf -v ~/nginx/logs:/var/log/nginx nginx

命令说明：

**-p 8082:80：** 将容器的 80 端口映射到主机的 8082 端口。

**--name nginx-look-web：**将容器命名为 **nginx-look-web**。

**-v ~/nginx/www:/usr/share/nginx/html：**将我们自己创建的 www 目录挂载到容器的 /usr/share/nginx/html。

**-v ~/nginx/conf/nginx.conf:/etc/nginx/nginx.conf：**将我们自己创建的 nginx.conf 挂载到容器的 /etc/nginx/nginx.conf。

**-v ~/nginx/logs:/var/log/nginx：**将我们自己创建的 logs 挂载到容器的 /var/log/nginx

启动以上命令后进入 ~/nginx/www 目录：

cd ~/nginx/www/

创建 index.html 文件，内容如下

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>docker(学习)</title>

</head>

<body>

<h1>我的第一个标题</h1>

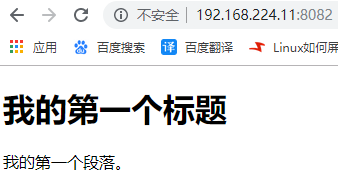
<p>我的第一个段落。</p>

</body>

</html>



输出结果为：



### **相关命令**

如果要重新载入 NGINX 可以使用以下命令发送 HUP 信号到容器：

docker kill -s HUP container-name

Container-name 为容器名称

[root@server1 www]# docker kill -s HUP nginx-look-web

重启 NGINX 容器命令：

$ docker restart container-name

[root@server1 www]# docker restart nginx-look-web

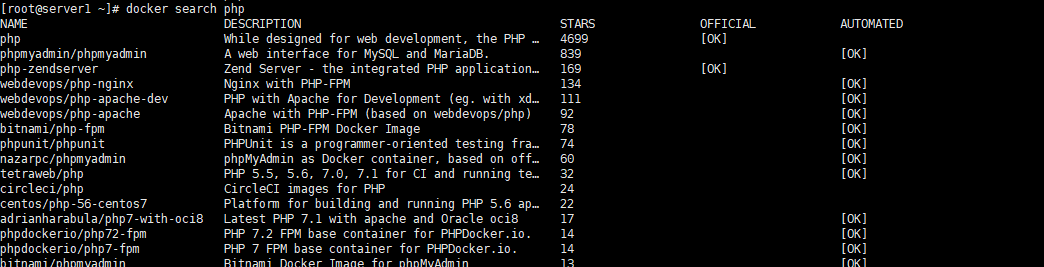
# Docker安装PHP

## 安装PHP镜像

### **方法一，docker pull php**

查找Docker Hub上的php镜像

[root@server1 ~]# docker search php



这里我们拉取官方的镜像,标签为5.6-fpm

[root@server1 ~]# docker pull php:5.6-fpm

等待下载完成后，我们就可以在本地镜像列表里查到REPOSITORY为php,标签为5.6-fpm的镜像。

## Nginx + PHP 部署

Nginx 部署可以查看：[Docker 安装 Nginx](https://www.runoob.com/docker/docker-install-nginx.html" \t "https://www.runoob.com/docker/_blank)，一些 Nginx 的配置参考这篇文章。

启动 PHP：

[root@server1 ~]# docker run -d --name myphp-fpm -v ~/nginx/www:/www php:5.6-fpm

命令说明：

**--name myphp-fpm** : 将容器命名为 myphp-fpm。

**-v ~/nginx/www:/www** : 将主机中项目的目录 www 挂载到容器的 /www

创建 ~/nginx/conf/conf.d 目录：

[root@server1 ~]# mkdir ~/nginx/conf/conf.d

在该目录下添加 **~/nginx/conf/conf.d/look-test-php.conf** 文件，内容如下：

[root@server1 ~]# cd ~/nginx/conf/conf.d/

vim ~/nginx/conf/conf.d/look-test-php.conf

server {

listen 80;

server\_name localhost;

location / {

root /usr/share/nginx/html;

index index.html index.htm index.php;

}

error\_page 500 502 503 504 /50x.html;

location = /50x.html {

root /usr/share/nginx/html; }

location ~ \.php$ {

fastcgi\_pass php:9000;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME /www/$fastcgi\_script\_name;

include fastcgi\_params;

}

}

配置文件说明：

* **php:9000**: 表示 php-fpm 服务的 URL，下面会具体说明。
* **/www/**: 是 **myphp-fpm** 中 php 文件的存储路径，映射到本地的 ~/nginx/www 目录。

启动 nginx：

[root@server1 conf.d]# docker run -d --name look-php-nginx -p 8083:80 -v ~/nginx/www:/usr/share/nginx/html:ro -v ~/nginx/conf/conf.d/:/etc/nginx/conf.d:ro --link myphp-fpm:php nginx

* **-p 8083:80**: 端口映射，把 **nginx** 中的 80 映射到本地的 8083 端口。
* **~/nginx/www**: 是本地 html 文件的存储目录，/usr/share/nginx/html 是容器内 html 文件的存储目录。
* **~/nginx/conf/conf.d**: 是本地 nginx 配置文件的存储目录，/etc/nginx/conf.d 是容器内 nginx 配置文件的存储目录。
* **--link myphp-fpm:php**: 把 **myphp-fpm** 的网络并入 nginx，并通过修改 **nginx** 的 /etc/hosts，把域名 **php** 映射成 127.0.0.1，让 nginx 通过 php:9000 访问 php-fpm。

接下来我们在 ~/nginx/www 目录下创建 index.php，代码如下：

cd ~/nginx/www

vim ~/nginx/www/index.php

<?php

echo phpinfo();

?>

浏览器打开 **http://127.0.0.1:8083/index.php**，显示如下：

