Jar包构建docker镜像脚本

#!/bin/bash

#判断 eurekaserver服务容器ID

containerID=`docker ps -a | grep eurekaserver | awk '{print $1}'`

#停止容器

docker stop $containerID

#删除容器

docker rm $containerID

#判断容器删除情况

if [ $containerID -eq 0 ];

then

echo '>>>删除容器成功'

fi

#判断 eurekaserver服务镜像ID

imagesID=`docker images |grep eurekaserver | awk '{print $3}'`

#删除镜像

docker rmi $imagesID

#判断容器删除情况

if [ $imagesID -eq 0 ];

then

echo '>>>删除镜像成功'

fi

#进入jar包所在目录 --Dockerfile也在同目录下

cd /app

#构建镜像

docker build -t eureka-servertest .

#启动镜像 (在后台启动)

docker run -d -p 8761:8761 eureka-servertest

docker container logs containerID

**docker logs 查看实时日志：**

docker logs -f -t --since=”2018-12-26” --tail=10 nsu-server

--since: 此参数指定了输出日志开始日期，即只输出指定日期之后的日志

-f: 查看实时日志

-t: 查看日志产生的日期

--tail=10 查看最后的10条日志

nsu-server 容器名称

[容器时间跟宿主机时间同步](https://www.cnblogs.com/kevingrace/p/5570597.html)：

<https://www.cnblogs.com/kevingrace/p/5570597.html>

# 微服务Docker化注册中心网络处理:

<http://blog.51cto.com/aaronchou/1965565>

手动配置nginx前端项目到docker中

一。手动配置 步骤如下

首先创建本地挂载目录

mkdir -p /docker/nginx/{conf,conf.d,html,log}

cd /docker/nginx/conf && vim nginx.conf

此处nginx.conf 内容省略，需要特别注意的是以下配置：

try\_files $uri /index.html

location / {

try\_files $uri $uri/ @router; #需要指向下面的@router否则会 出现vue的路由在nginx中刷404

index index.html index.htm;

}

#对应上面的@router，主要原因是路由的路径资源并不是一个真实的路径，所以无法找到具体的文件

#因此>需要rewrite到index.html中，然后交给路由

在处理请求资源

location @router {

rewrite ^.\*$ /index.html last;

}

然后做下面操作：

1.docker pull nginx

2.docker run -d -p 80:80 nginx

3.docker exec -it xxx /bin/bash

4.创建项目目录

mkdir -p /usr/share/nginx/html/dist

mkdir -p /usr/share/nginx/html/maintenance

5.ctrl+p+q 退出容器 8083:7979 8082:7978 8081:8080 8084:80

6.docker stop xx 停止容器

7.docker run --name mynginx -d -p 8080:80 -p 8081:8081 \

-v /docker/nginx/conf/nginx.conf:/etc/nginx/nginx.conf \

-v /docker/nginx/log:/var/log/nginx \

-v /docker/nginx/html:/usr/share/nginx/html \

-v /docker/dist:/usr/share/nginx/html/dist nginx

docker restart containerID 重启容器

**如何在docker容器和宿主机之间复制文件:**

docker cp containerID:container\_path host\_path 从容器复制到主机

docker cp host\_path containerID:container\_path 从主机复制到容器