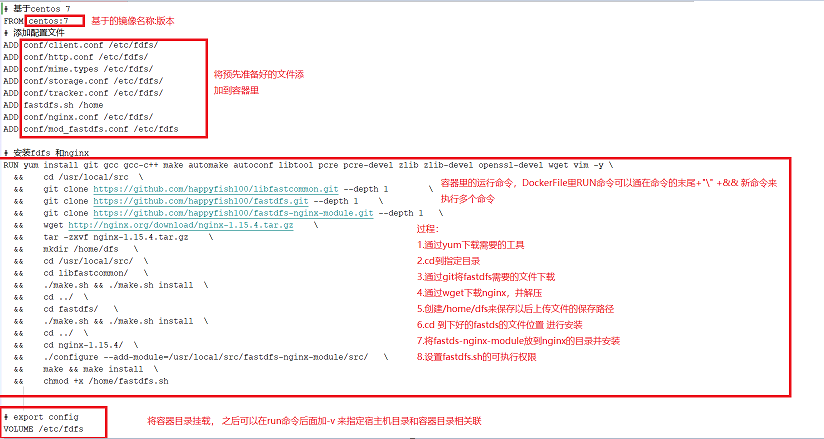
## 生成镜像和运行命令

**一．生成镜像**

1. 先将预先的3个文件放到同一目录下



DockerFile 的配置



DockerFile文件里用的命令，具体可以参考：<https://www.cnblogs.com/panwenbin-logs/p/8007348.html>



1. 运行命令

docker build -t fdfsjdktomcatnginx:v1 . （-t 镜像名称: 版本号）

运行后出现两个Successfully为成功



到现在包含nginx+jdk+tomcat+fdfs的就有了

1. 查看镜像

docker images



1. 将镜像打成tar

docker save -o fdfsjdktomcatnginx.tar fdfsjdktomcatnginx:v1

Docker save -o [tar名称] [镜像:版本]



1. 将tar导入成镜像

docker load -i fdfsjdktomcatnginx.tar

docker load -i [tar名称]

1. 删除镜像

docker rmi -f fdfsjdktomcatnginx:v1

docker rmi -f [镜像名称:版本]

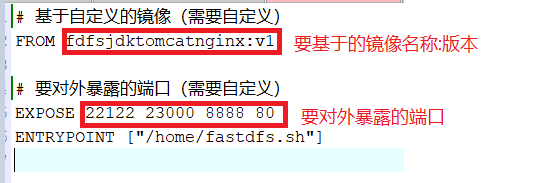
1. 从网上拉起镜像

docker pull nginx

docker pull [镜像名称:版本]

1. **生成fdfs的容器**

将生成fdfs里的文件复制到同一目录，将dockerFile按需求配置，映射镜像，暴露端口，添加的文件等



fastdfs.sh讲解



1. 生成镜像

docker build -t fdfs:v1 .

1. 运行成fdfs容器的命令

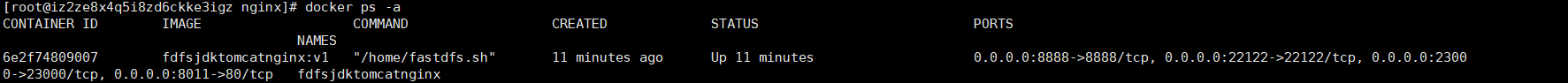
docker run -dit -e FASTDFS\_IPADDR=39.106.199.188 -p 8888:8888 -p 22122:22122 -p 23000:23000 -p 8011:80 --name fdfs fdfs:v1

（其中-e FASTDFS\_IPADDR=xx.xx.xx.xx是宿主机的ip 对应fastdfs.sh文件的配置, -p 后面是要映射的地址 宿主端口:容器端口, -name 是要设置的容器名称，最后的xxx:xx是镜像的名称:版本）

上传的文件都保存在容器的/home/dfs

1. 查看容器

docker ps -a



1. 进入容器

docker exec -it fdfs bash

docker exec -it [容器名称] bash

进入后可以通过exit来退出容器

1. 启动，关闭，重启容器

docker [命令] [容器名称]

docker start fdfs 启动

docker stop fdfs 关闭

docker restart fdfs 重启

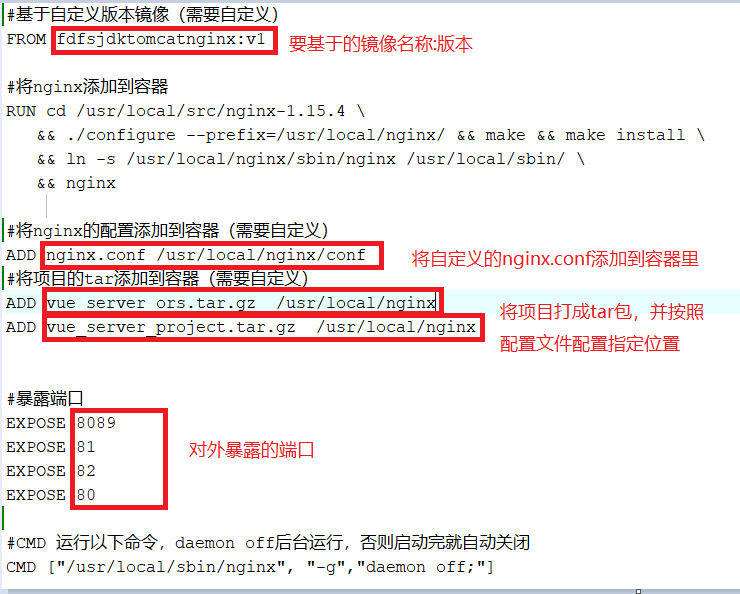
1. 删除容器

docker rm -f fdfs

docker rm -f [容器名称]

1. **生成nginx的容器**

将生成nginx里的文件复制到同一目录，将dockerFile按需求配置，映射镜像，暴露端口，添加的文件



1. 生成镜像

docker build -t nginx:v1 .

1. 运行成fdfs容器的命令

docker run -dit -p 8089:8089 -p 81:81 -p 82:82 -p 8088:80 --name nginx nginx:v1

（其中 -p 后面是要映射的地址 宿主端口:容器端口, -name 是要设置的容器名称，最后的xxx:xx是镜像的名称:版本）

1. 查看容器

docker ps -a

1590850727(1)

1. 进入容器

docker exec -it nginx bash

docker exec -it [容器名称] bash

进入后可以通过exit来退出容器

1. 启动，关闭，重启容器

docker [命令] [容器名称]

docker start nginx 启动

docker stop nginx 关闭

docker restart nginx 重启

1. 删除容器

docker rm -f nginx

docker rm -f [容器名称]

1. **生成tomcat容器**

将生成tomcat里的文件复制到同一目录，将dockerFile按需求配置，映射镜像，暴露端口，添加的文件



1. 生成镜像

docker build -t tomcat:v1 .

1. 运行成fdfs容器的命令

docker run -dit -p 8089:8080 --name tomcat tomcat:v1

（其中 -p 后面是要映射的地址 宿主端口:容器端口, -name 是要设置的容器名称，最后的xxx:xx是镜像的名称:版本）

1. 查看容器

docker ps -a

1590852478(1)

1. 进入容器

docker exec -it tomcat bash

docker exec -it [容器名称] bash

进入后可以通过exit来退出容器

1. 启动，关闭，重启容器

docker [命令] [容器名称]

docker start tomcat 启动

docker stop tomcat 关闭

docker restart tomcat 重启

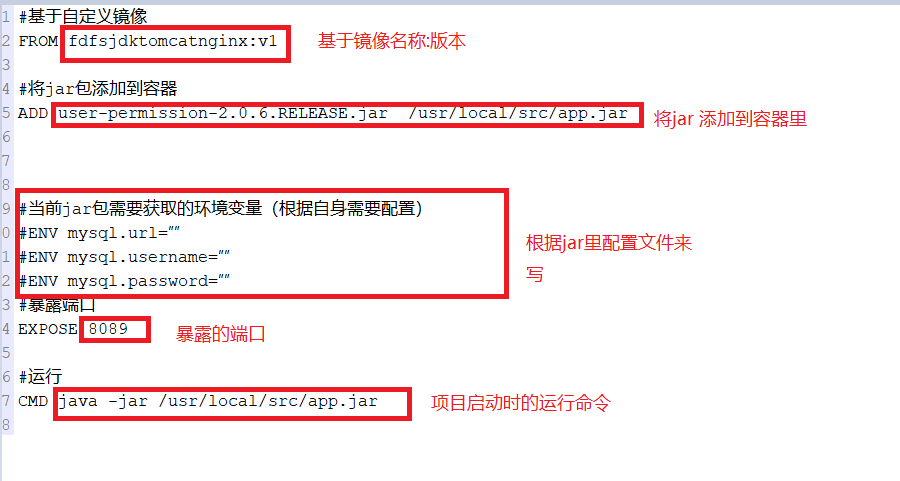
1. 删除容器

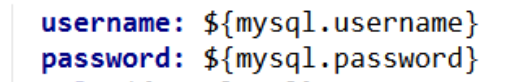
docker rm -f tomcat

docker rm -f [容器名称]

1. **生成jdk容器**

将生成jdk里的文件复制到同一目录，将dockerFile按需求配置，映射镜像，暴露端口，添加的文件



注意 ENV对应项目配置里的 

1. 生成镜像

docker build -t jdk:v1 .

1. 运行成fdfs容器的命令

docker run -dit -p 8089:8089 --env nacos.addr=172.16.1.100:8848 --env mysql.username=root --name jdk jdk:v1

（其中 -p 后面是要映射的地址 宿主端口:容器端口, -env 是对应DockerFiel里的ENV （没有不用写）,-name 是要设置的容器名称，最后的xxx:xx是镜像的名称:版本）

1. 查看容器

docker ps -a



1. 进入容器

docker exec -it jdk bash

docker exec -it [容器名称] bash

进入后可以通过exit来退出容器

1. 启动，关闭，重启容器

docker [命令] [容器名称]

docker start jdk 启动

docker stop jdk 关闭

docker restart jdk 重启

1. 删除容器

docker rm -f jdk

docker rm -f [容器名称]