

**Dockerfile常用关键字**

* FROM  
  指定基础镜像，当前新镜像是基于哪个镜像的。其中，scratch是个空镜像，这个镜像是虚拟的概念,并不实际存在,它表示一个空白的镜像，当前镜像没有依赖于其他镜像  
  FROM scratch
* MAINTAINER  
  镜像维护者的姓名和邮箱地址  
  MAINTAINER Sixah <sixah@163.com>
* RUN  
  容器构建时需要运行的命令  
  RUN echo 'Hello, Docker!'
* EXPOSE  
  当前容器对外暴露出的端口  
  EXPOSE 8080
* WORKDIR  
  指定在创建容器后，终端默认登陆进来的工作目录，一个落脚点  
  WORKDIR /go/src/app
* ENV  
  用来在构建镜像过程中设置环境变量  
  例如，ENV MY\_PATH /usr/mytest  
  这个环境变量可以在后续的任何RUN指令中使用，这就如同在命令前面指定了环境变量前缀一样;当然，也可以在其他指令中直接使用这些环境变量，比如：WORKDIR $MY\_PATH
* ADD  
  将宿主机目录下的文件拷贝进镜像且ADD命令会自动处理URL和解压tar压缩包  
  ADD Linux\_amd64.tar.gz
* COPY  
  类似于ADD，拷贝文件和目录到镜像中，将从构建上下文目录中<源路径>的文件/目录复制到新的一层镜像内的<目标路径>位置
* # 拷贝运行jar包到工作空间
* COPY com.docker.helloworld-1.0-SNAPSHOT.jar /usr/local/com.docker.helloworld-1.0-SNAPSHOT.jar
* # 添加JDK包并ADD到工作空间, 此处会自动解压, 注意解压不会解压zip包,并且路径必须与DockerFile同级  
  COPY . /go/src/app
* VOLUME  
  容器数据卷，用于数据保存和持久化工作  
  VOLUME /data
* CMD  
  指定一个容器启动时要运行的命令。Dockerfile中可以有多个CMD指令，但只有最后一个生效，CMD会被docker run之后的参数替换  
  CMD ["/bin/bash"]
* ENTRYPOINT  
  指定一个容器启动是要运行的命令。ENTRYPOINT的目的和CMD一样，都是在指定容器启动程序及参数
* ONBUILD  
  当构建一个被继承的Dockerfile时运行的命令，父镜像在被子镜像继承后，父镜像的ONBUILD指令被触发