本周继续学习docker常用命令，主要学习了容器命令

退出容器，两种退出方式

exit：run进去容器，exit退出，容器停止

ctrl+p+q：run进去容器，ctrl+p+q退出，容器不停止

启动已停止运行的容器：docker start 容器ID或者容器名

重启容器：docker restart 容器ID或者容器名

停止容器：docker stop 容器ID或者容器名

强制停止容器：docker kill 容器ID或容器名

删除已停止的容器：docker rm 容器ID

启动守护式容器(后台服务器)：在大部分的场景下，我们希望 docker的服务是在后台运行的，我们可以过-d指定容器的后台运行模式。docker run -d 容器名。以redis为例

前台交互式启动：docker run -it redis；后台守护式启动：docker run -d redis。

查看容器日志：docker logs 容器ID

查看容器内运行的进程：docker top 容器ID

查看容器内部细节：docker inspect 容器ID

进入正在运行的容器并以命令行交互

docker exec -it 容器ID bashShell（exec 是在容器中打开新的终端，并且可以启动新的进程 用exit退出，不会导致容器的停止。）

docker attach 容器ID（attach 直接进入容器启动命令的终端，不会启动新的进程 用exit退出，会导致容器的停止。）

从容器内拷贝文件到主机上：docker cp 容器ID:容器内路径 目的主机路径

导入和导出容器

export 导出容器的内容留作为一个tar归档文件：docker export 容器ID > 文件名.tar

import 从tar包中的内容创建一个新的文件系统再导入为镜像：cat 文件名.tar | docker import - 镜像用户/镜像名:镜像版本号

Docker镜像底层原理

Docker中的镜像分层，支持通过扩展现有镜像，创建新的镜像。类似Java继承于一个Base基础类，自己再按需扩展。

新镜像是从 base 镜像一层一层叠加生成的。每安装一个软件，就在现有镜像的基础上增加一层

可使用docker commit提交容器副本使之成为一个新的镜像。docker commit -m="提交的描述信息" -a="作者" 容器ID 要创建的目标镜像名:[标签名]