**docker服务相关命令**

启动docker： systemctl start docker

停止docker：systemctl stop docker

重启docker：systemctl restart docker

查看docker服务状态：systemctl status docker

设置开机启动docker：systemctl enable docker

**Docker镜像相关命令**

docker images 查看镜像文件

Docker search （） 查询某一镜像文件

Docker pull （）加版本号 下载该版本的镜像文件

查看想下载版本号docker是否拥有：

hub.docker.com进入docker官方进行查询

Docker rmi (想要删除的镜像id) 删除镜像文件

Docker image -q 查看所有镜像

**Docker容器相关命令**

创建容器：sudo docker run -it --name=lwh ubuntu /bin/bash

* **-i**: 交互式操作。
* **-t**: 终端。
* **ubuntu**: ubuntu 镜像。
* **/bin/bash**：放在镜像名后的是命令，这里我们希望有个交互式 Shell，因此用的是 /bin/bash。

--name=()对容器进行命名

退出容器指令：exit

**查看正在运行的容器**：docker ps

**查看所有容器**：docker ps -a

启动一个容器：docker start （容器id）

创建一个后台运行的容器：docker run -itd --name=() 容器选择的镜像文件 /bin/bash

在使用 **-d** 参数时，容器创建后会进入后台，使用exec进入容器且退出后容器不会关闭。

进入一个正在后台运行的容器：docker exec -it 容器名 /bin/bash

停止一个容器：docker stop (容器id)

-it创建的容器称为交互式容器

-id创建的容器称为守护式容器

删除一个容器：docker rm （容器id或者名）

查看容器的信息：docker inspect （容器名）（要在非容器内执行）

**Docker容器的数据卷**

数据卷：独立于容器，存在于宿主机里的一个文件或者目录，当容器的目录与数据卷的目录绑定后，其中任意一方对文件的修改，两方都会文件都会改变（铁索连环）。但是当容器被删除后，数据卷中依然会保留容器里的数据。一个数据卷可以被多个容器同时挂载，一个容器也可以挂载多个数据卷

配置一个数据卷sudo docker run -it --name=c3 -v /root/data**:**/root/date\_container ubuntu /bin/bash

-v 宿主机目录文件：容器内目录文件

注意：两文件间的：没有空格

打开root目录步骤

sudo su

Cd /root

Cd /root下文件名

实现两个容器间的互联只需在创建容器时选择相同文件目录的数据卷即可。

**数据卷容器：作为一个媒介。**

**将c1与c2容器挂载到数据卷容器c3上，将c3挂载到数据卷上，此时就相当于c1，c2，c3都挂载到了数据卷上，就算c3，出问题，c1与c2依旧可以联系数据卷。**

创建数据卷容器：sudo docker run -it --name=c3 -v /volume ubuntu /bin/bash

注:/volume 为创建的数据卷容器文件名，可以改。

创建容器c1并连接到c3容器 sudo docker run -it --name=c1 --volumes-from c3 ubuntu

创建容器c2并连接到c3容器 sudo docker run -it --name=c2 --volumes-from c3 ubuntu

**Docker的应用部署**

**Mysql**

端口映射：容器内的网络服务与外部机器的网络服务是不能直接通信的。但是外部机器和宿主机可以直接通信。宿主机与容器间也可以直接通信。因此，当容器中的网络服务需要被外部机器访问时，可以将容器中提供服务的端口映射到宿主机的端口上外部机器就可以通过访问宿主机的服务端口，进而访问容器内的网络服务。