运行test\_1容器，并在容器内创建一个路径为/yjl的数据卷

docker run -d -p 801:80 --name mynginx2 -v /yjl docker.io/nginx

进入容器：

docker exec -it mynginx2 /bin/bash

运行test\_2容器，加载宿主机的/root/yjl目录到容器的/data\_2目录作为数据卷

docker run -d -p 802:80 --name mynginx3 -v /root/yjl:/data\_2 docker.io/nginx

进入容器：

docker exec -it mynginx3 /bin/bash

使用docker volume命令创建一个别名为wordpress的数据卷

docker volume create --name wordpress

查看数据卷信息

[root@localhost www]# docker volume list

DRIVER VOLUME NAME

local 56c755c429d28e3a26d202b9d43196216203fdf96045c22f2d6e4c277265c731

local wordpress

[root@localhost www]# docker volume inspect wordpress

[

{

"Driver": "local",

"Labels": {},

"Mountpoint": "/var/lib/docker/volumes/wordpress/\_data",

"Name": "wordpress",

"Options": {},

"Scope": "local"

}

]

请注意此目录是宿主机的数据卷，

而不是容器的数据卷。

运行mynginx4容器，挂载wordpress数据卷到容器的/wordpress目录作为数据卷

docker run -d -p 804:80 --name mynginx4 -v wordpress:/wordpress docker.io/nginx

docker volume rm <参数：数据卷的VOLUME NAME>

删除数据卷

请注意如果数据卷没有取别名，

则名称是64位的全名，

不支持模糊匹配，

所以在创建数据卷时最好取别名。

docker rm -v <参数：容器名称>

随容器删除数据卷，

只针对随容器创建时而创建的数据卷