

**长沙理工大学**

计算机与通信工程学院

实验报告

**课程名称**： **云计算 \_\_\_\_\_\_**

**年级**： **大三 班级**： **网络2102班**

**学号**： **202108060918 姓名**： **王俊\_\_\_\_**

**完成日期**： **2024 年 5 月**  **16 日**

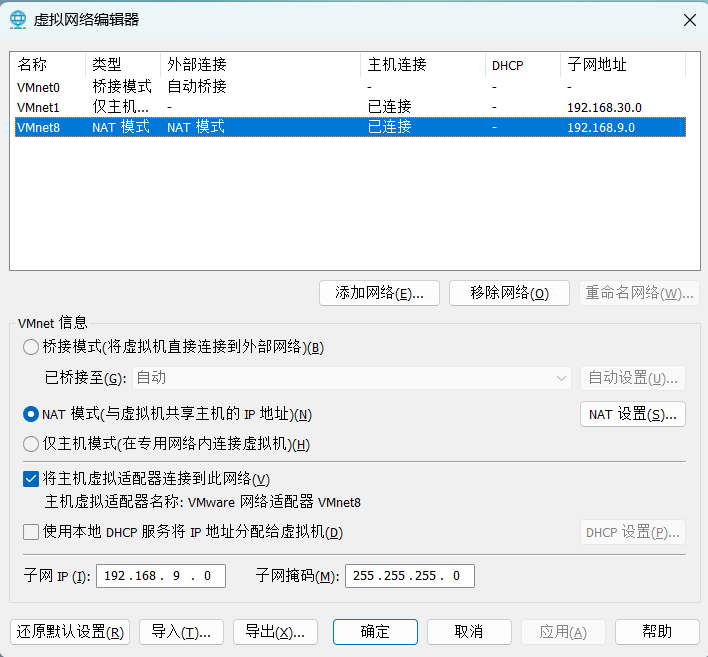
**授课老师： 廖年冬\_\_\_\_\_\_\_\_\_\_\_\_\_**

1.实训环境准备

（1）VMWare网络设置

打开VMware workstation，在菜单中选“编辑”→“虚拟网络编辑器”。

在方框1中选定VMnet8，在方框2中输入子网IP和子网掩码。



（2）虚拟主机准备

从虚拟机centos2009快照“命令行”克隆一个虚拟机：

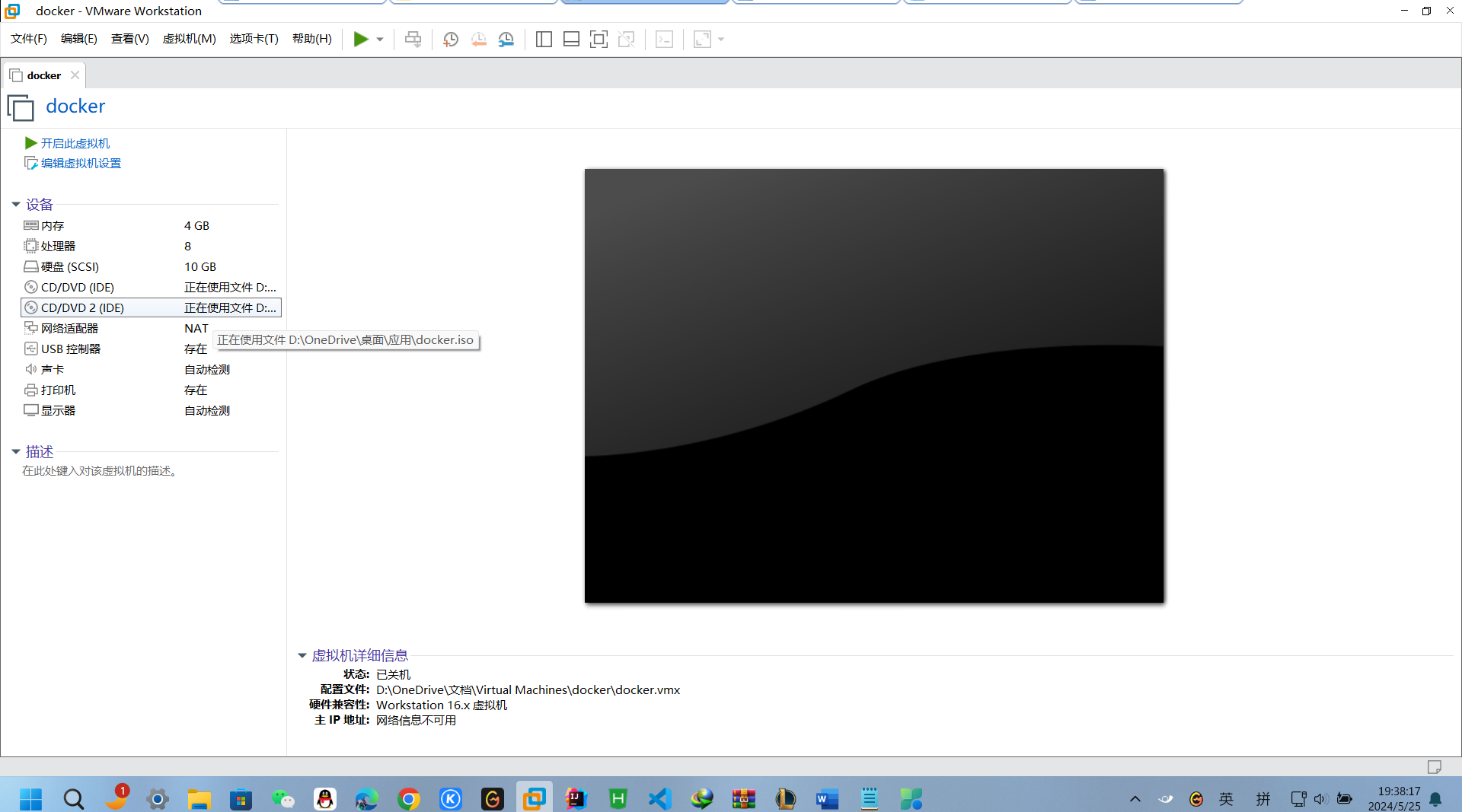
主机名称：docker

内存：8G

CD/DVD：增加一个CD/DVD

CD/DVD1：使用docker.iso为虚拟光驱

CD/DVD2：使用CentOS-7-x86\_64-DVD-2009.iso为虚拟光驱



2.载入镜像

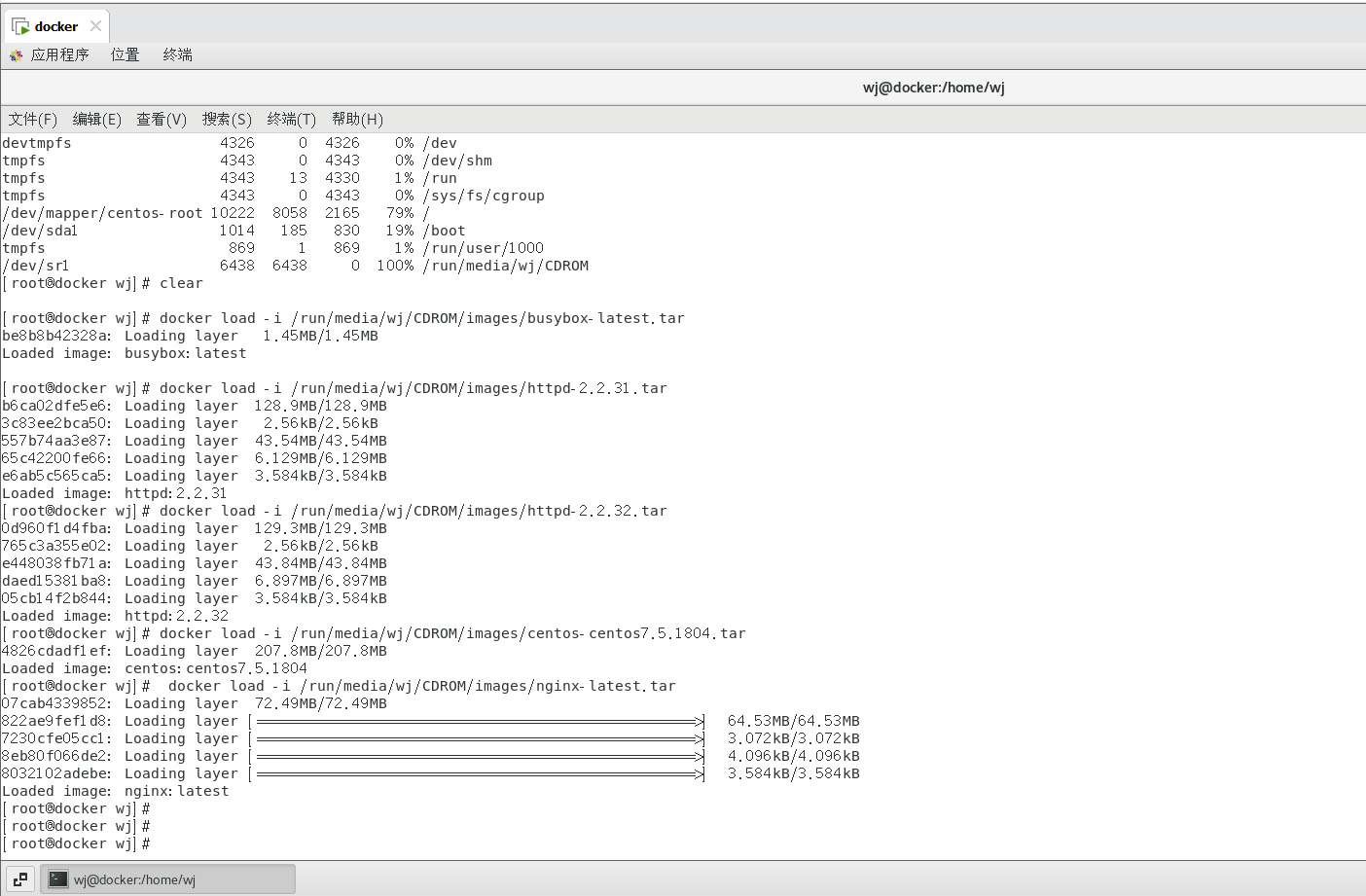
**# docker load -i /mnt/docker/images/busybox-latest.tar**

**# docker load -i /mnt/docker/images/httpd-2.2.31.tar**

**# docker load -i /mnt/docker/images/httpd-2.2.32.tar**

**# docker load -i /mnt/docker/images/centos-centos7.5.1804.tar**

**# docker load -i /mnt/docker/images/nginx-latest.tar**



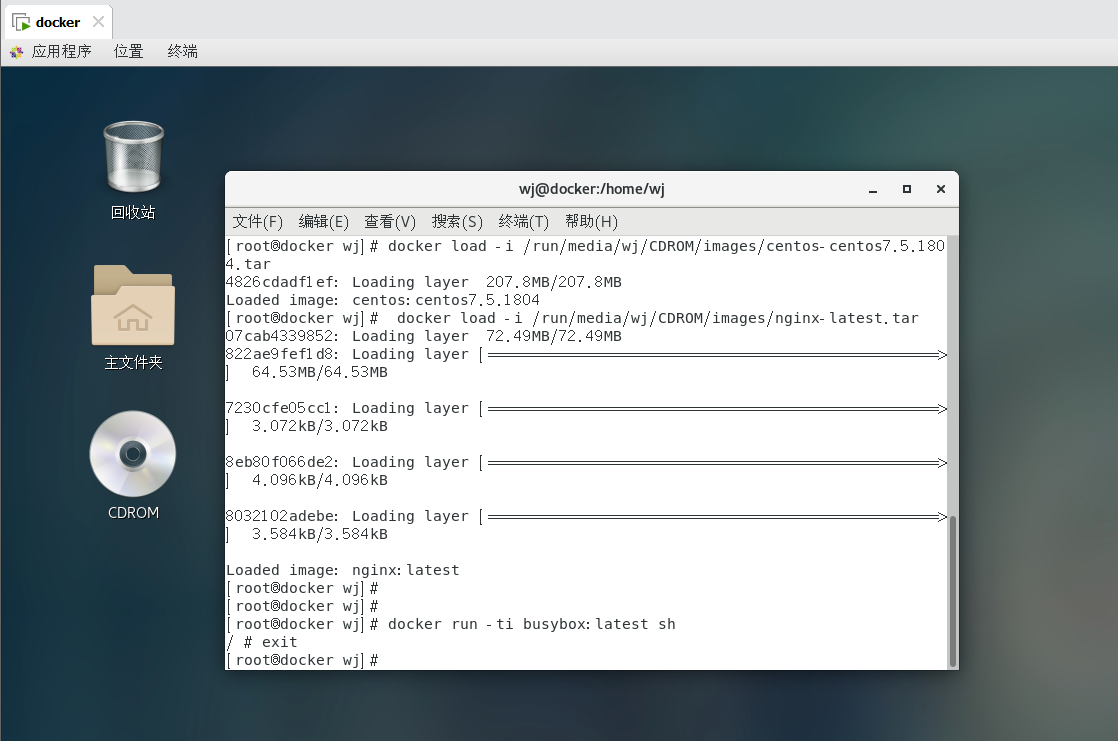
3.容器的创建

（1）用docker run创建容器

用-ti选项运行容器

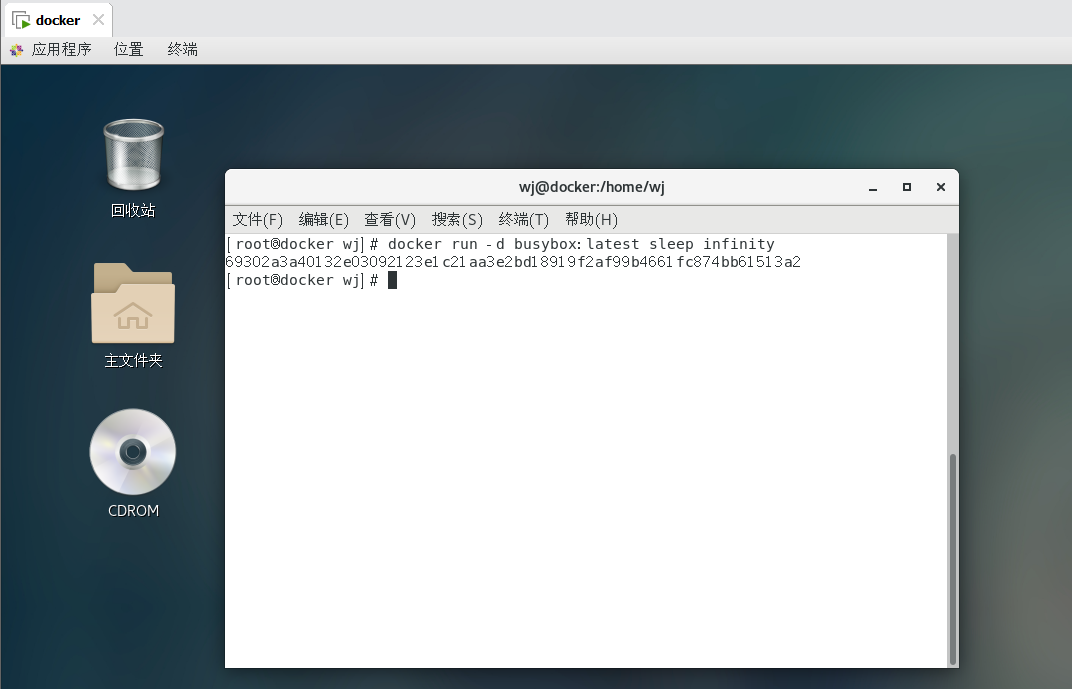
**# docker run -ti busybox:latest sh**

**/ # exit**



用-d在后台运行容器

**# docker run -d busybox:latest sleep infinity**



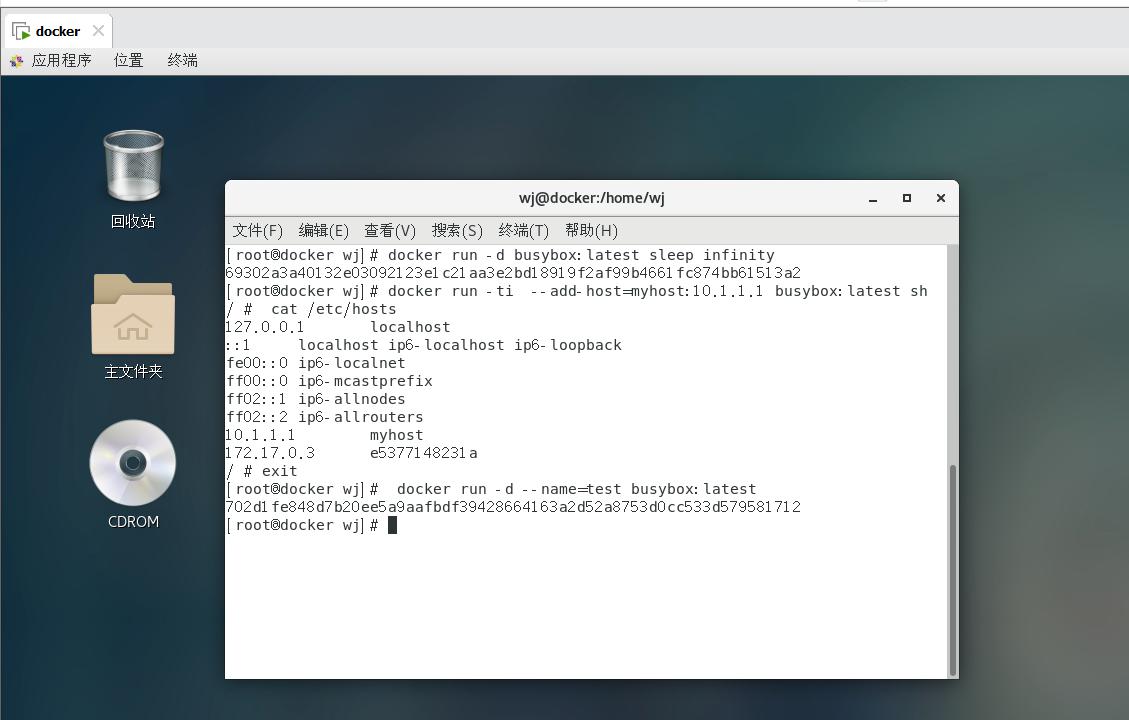
使用docker run的其它选项

**# docker run -ti --add-host=myhost:10.1.1.1 busybox:latest sh**

**/ # cat /etc/hosts**

**/ # exit**

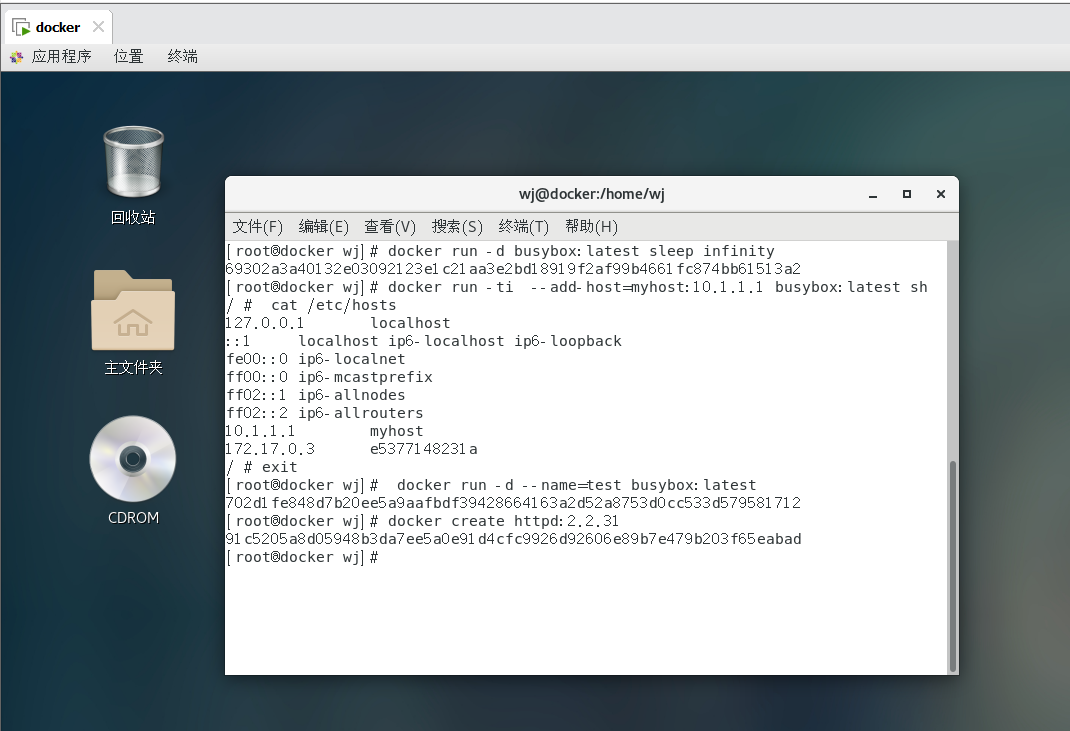
**# docker run -d --name=test busybox:latest**



（2）分步创建容器

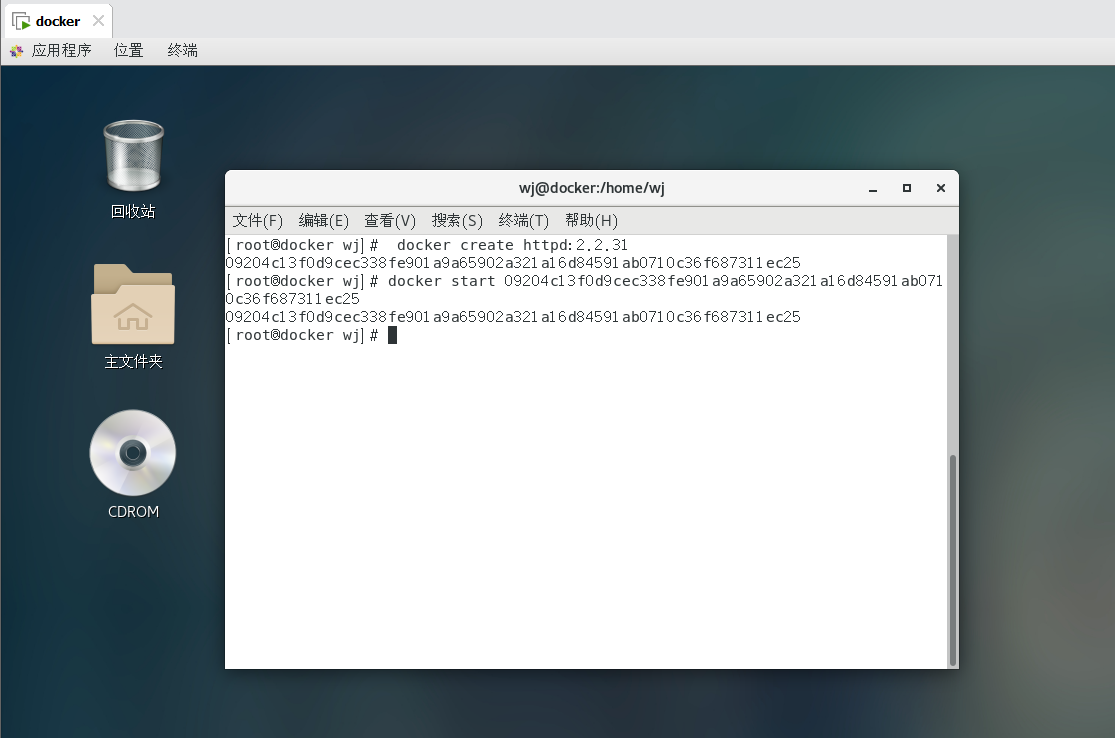
先使用docker create命令创建的容器。

**# docker create httpd:2.2.31**

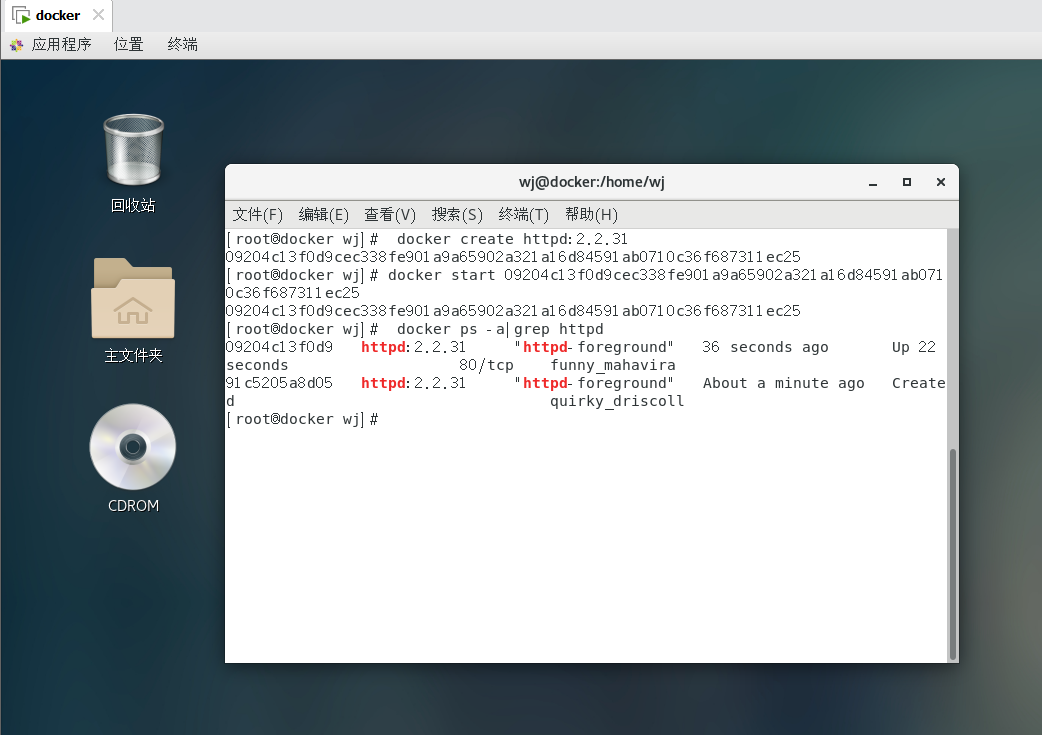


然后使用docker start命令创建的容器。

**# docker start 34c3b78c029ff765**

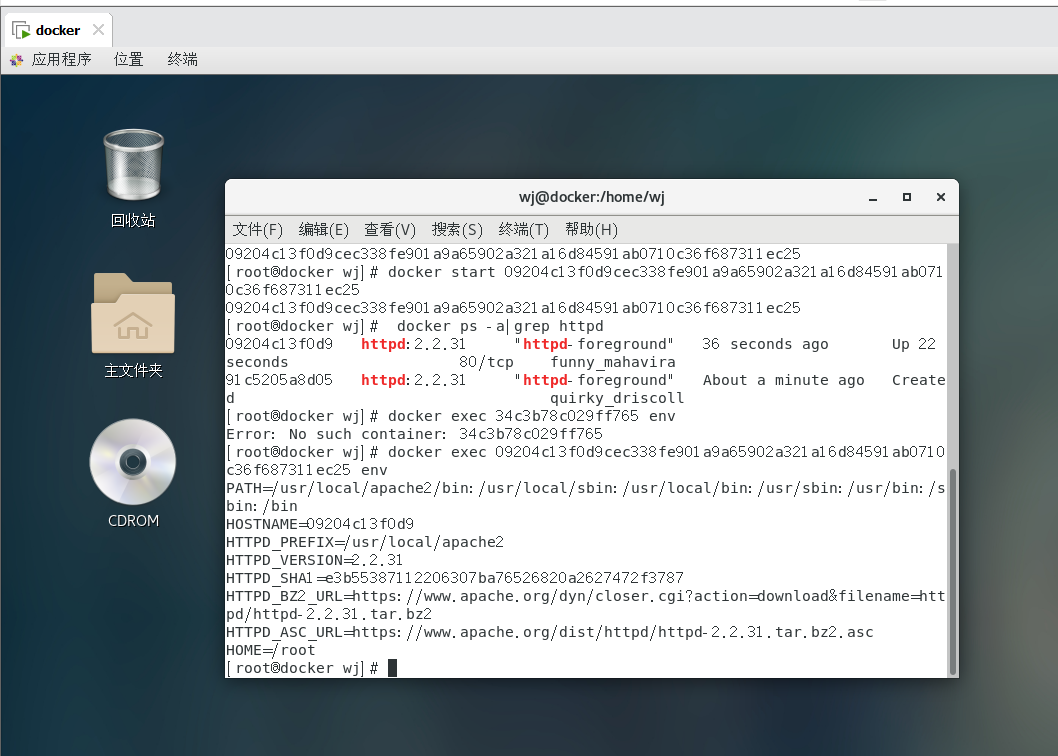


**# docker ps -a|grep httpd**



（3）在容器中执行命令env

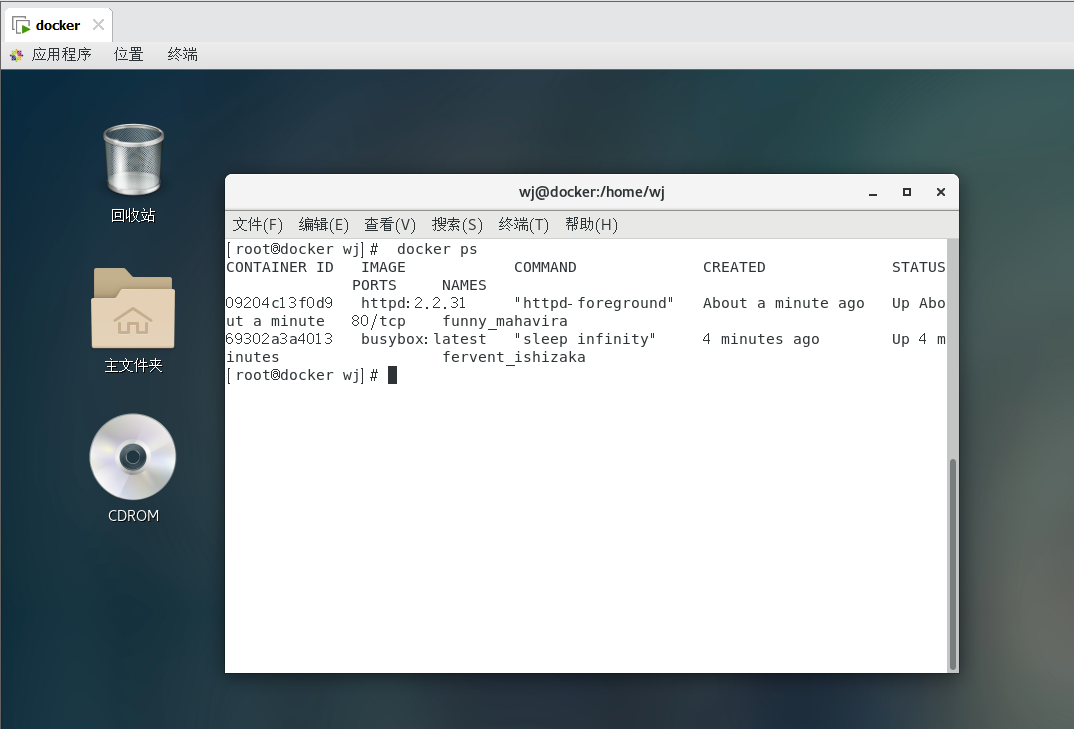
**# docker exec 34c3b78c029ff765 env**



4.查询容器

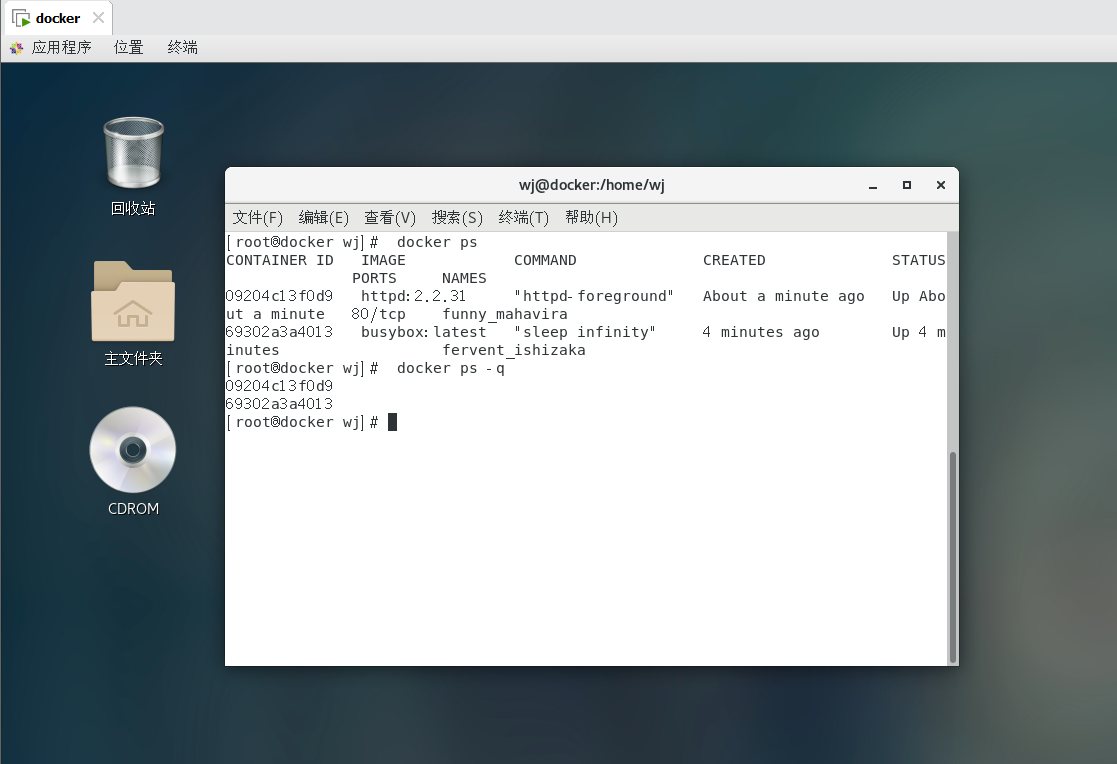
（1）查询运行中的容器。

**# docker ps**



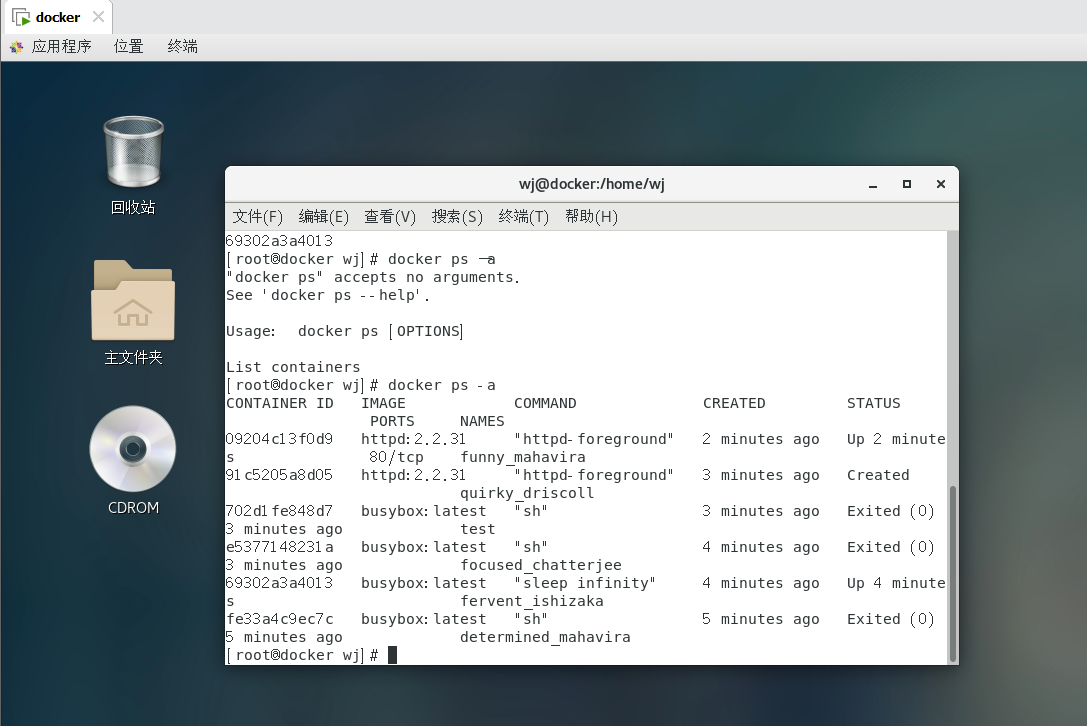
（2）选项-q只显示容器的ID

**# docker ps -q**



（3）查询所有状态的容器

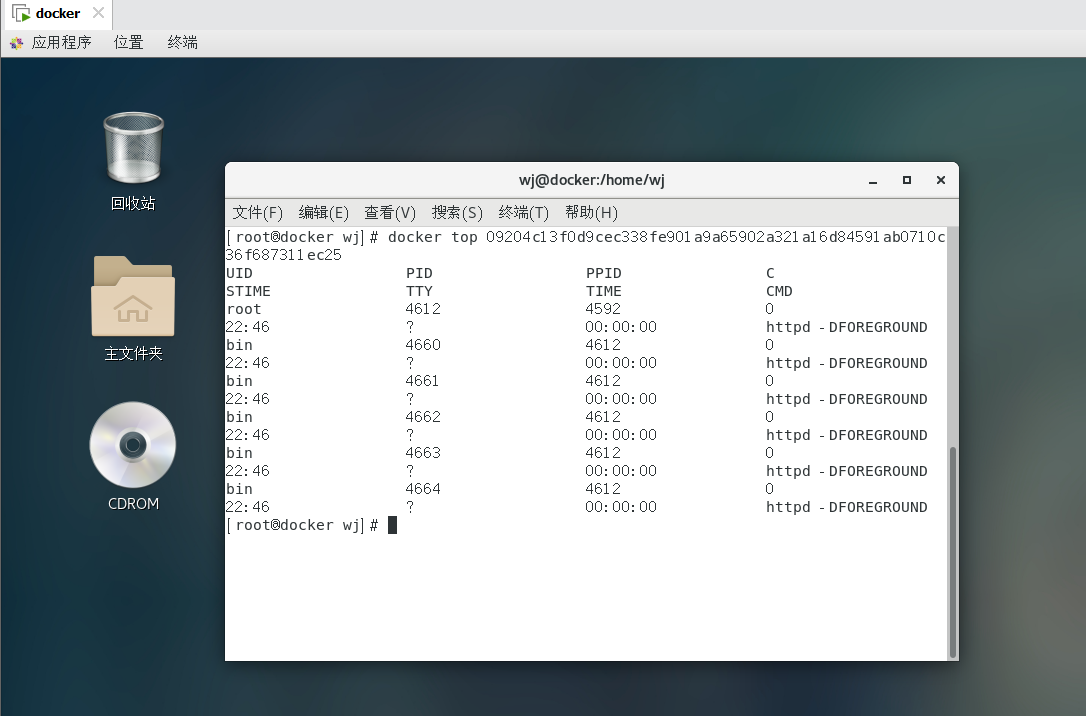
**#docker ps –a**



5.查询容器内进程

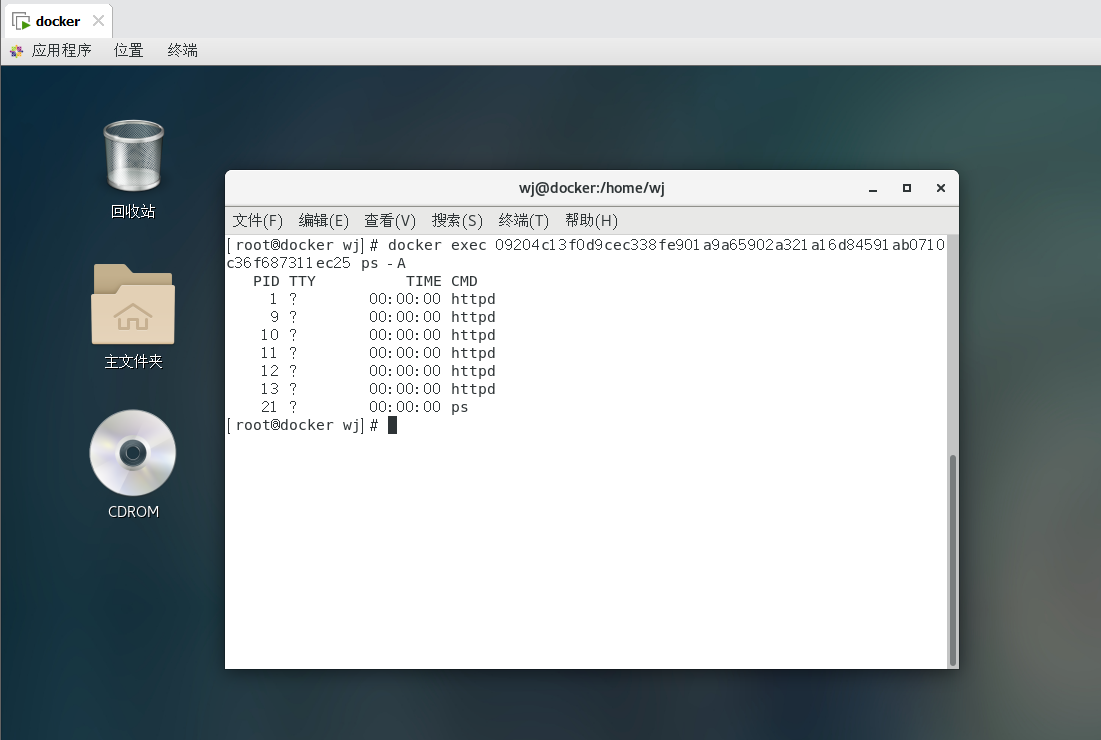
（1）在容器外部查询容器的进程

**# docker top 34c3b78c029ff765**



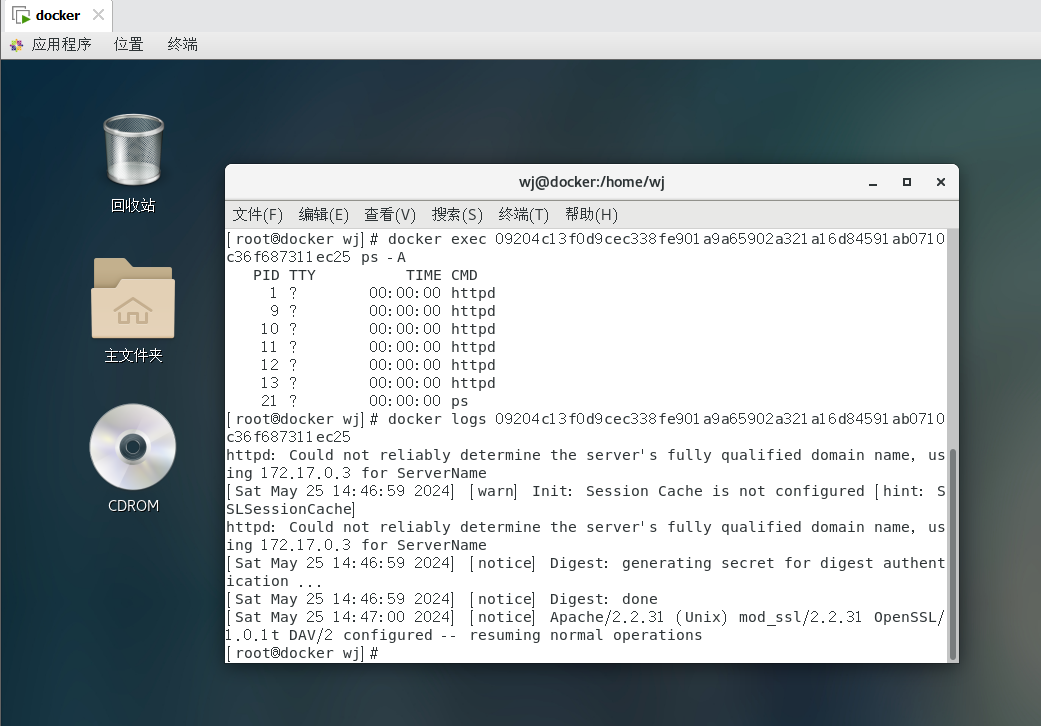
（2）在容器内部查询容器的进程。条件：容器内有ps命令

**# docker exec 34c3b78c029ff765 ps -A**



6.查询容器日志

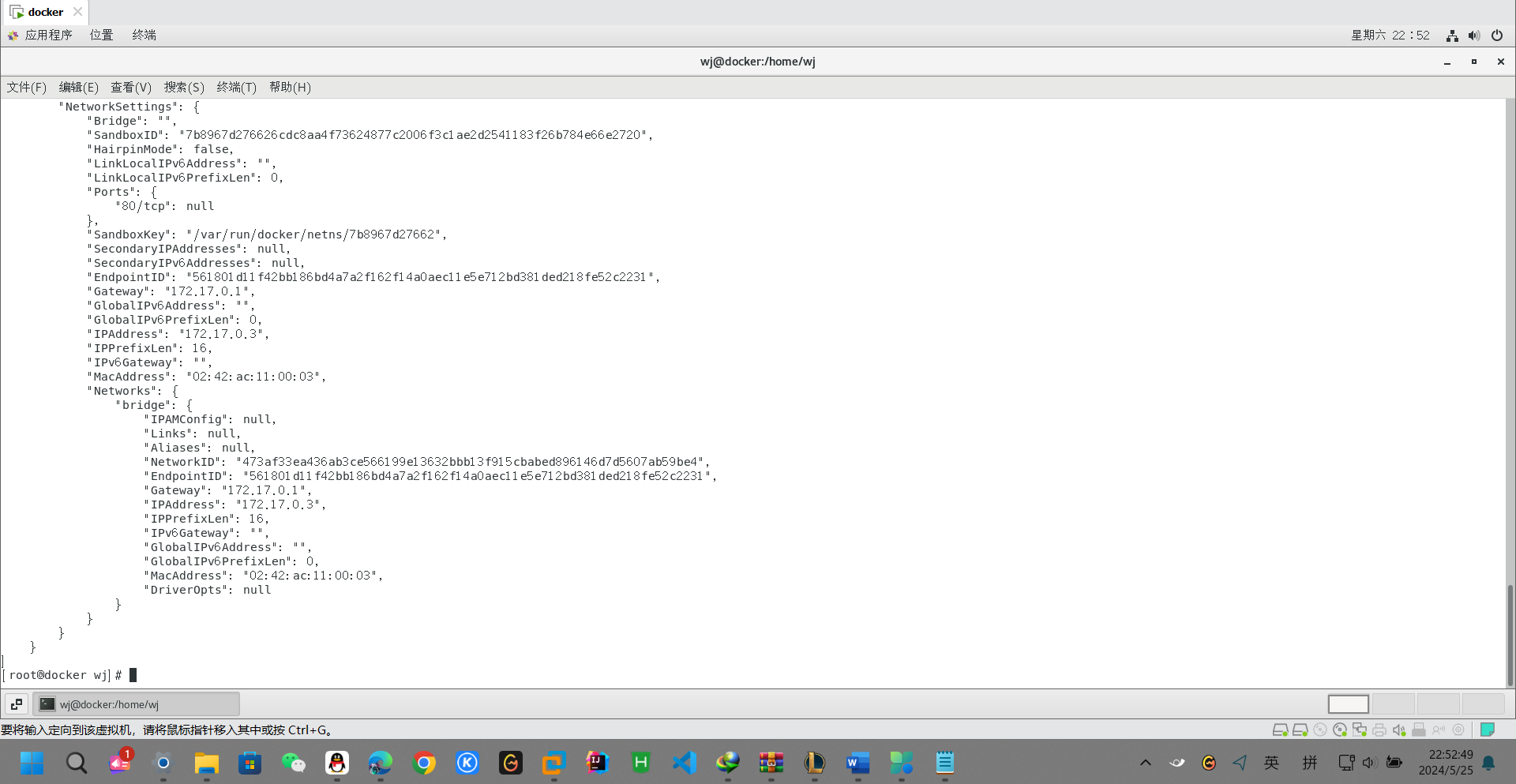
**# docker logs 34c3b78c029ff765**



7.查询容器的详细信息

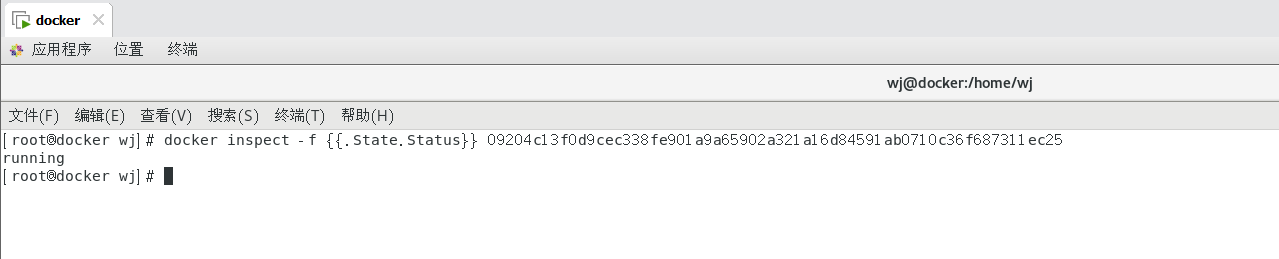
（1）查询所有信息

**# docker inspect 34c3b78c029ff765**



（2）查询指定信息

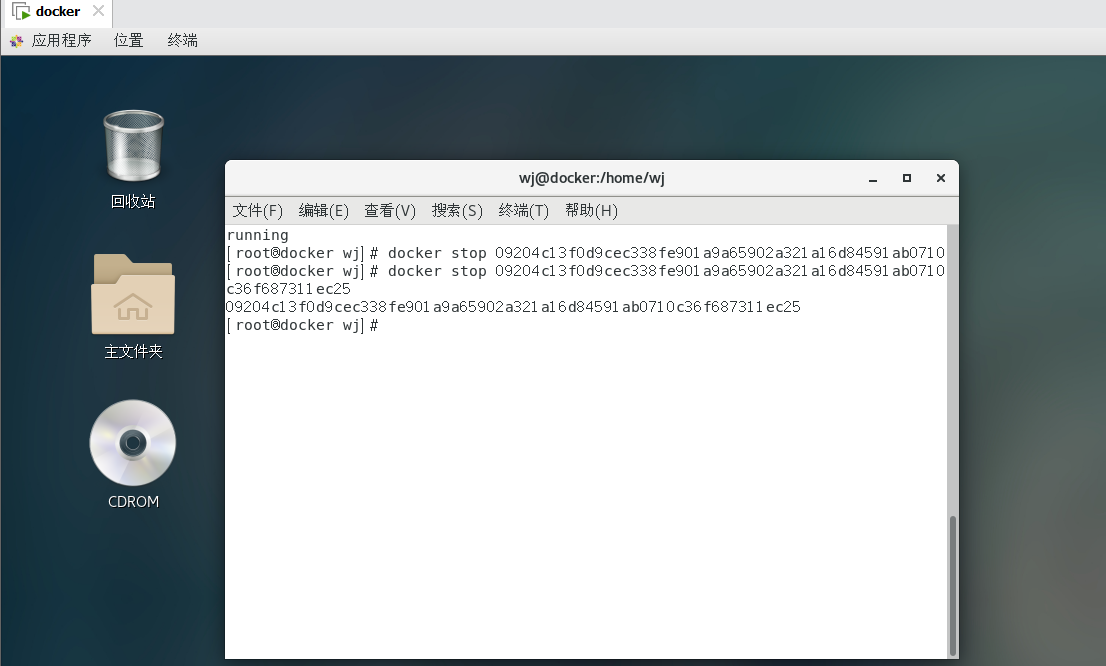
**# docker inspect -f {{.State.Status}} 34c3b78c029ff765**



8.停止和删除容器

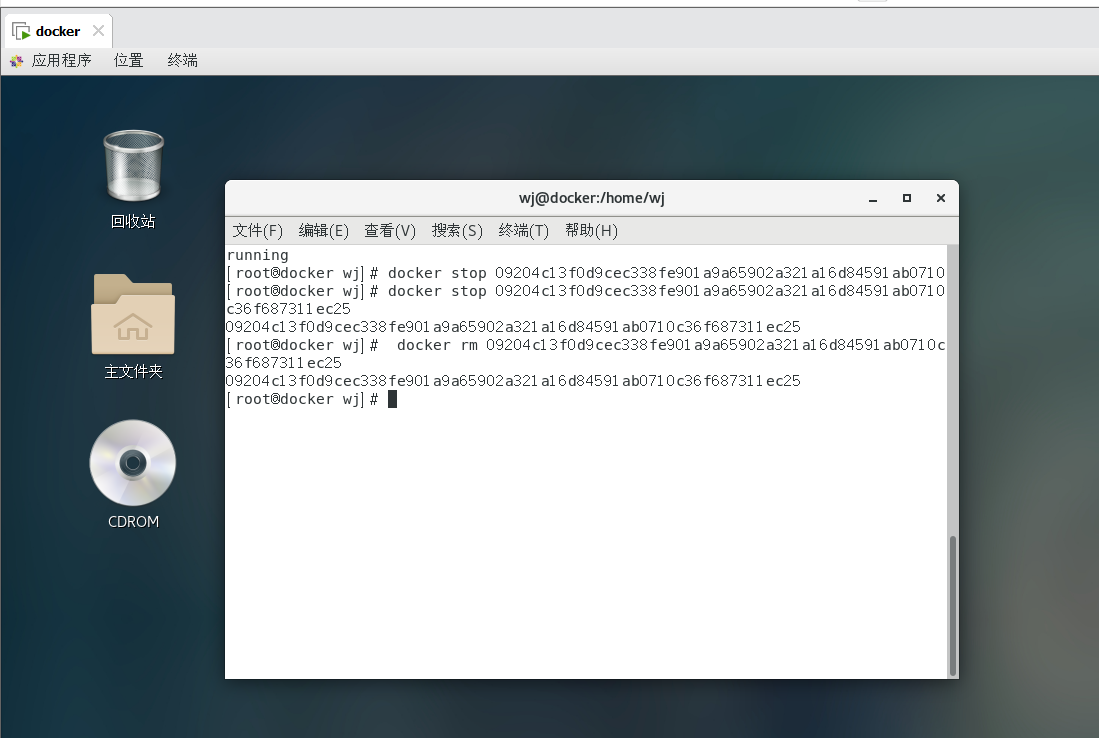
（1）停止容器

**# docker stop 34c3b78c029ff765**



（2）删除容器

**# docker rm 34c3b78c029ff765**



（3）删除一个正在运行中的容器

**# docker rm -f 34c3b78c029ff765**

