

## 0.安装

```
1 #安装 docker
2 yum install docker -y
3
4 systemctl start docker.service
```

### 1.docker 状态操作

启动 `systemctl start docker`

重启 `systemctl restart docker`

停止 `systemctl stop docker`

### 2.docker查看相关

查看镜像: `docker images`

查看容器: `docker ps` 正在运行的容器 `docker ps -a` 查看所有的容器

### 3.docker删除相关

先删除容器: `docker rm ID`

再删除镜像: `docker rmi ID`

### 4.docker 镜像操作 (以mysql为例)

搜索 `docker search mysql`

拉取指定的 `docker pull mysql`

### 5.docker 启动

命令格式: `docker run [OPTIONS] IMAGE [COMMAND] [ARG...]`

Usage: Run a command in a new container

中文意思为: 通过run命令创建一个新的容器 (container)

用镜像创建一个新的容器

例如:

```
1 sudo docker run --name dockermysql -p 3307:3306 -e MYSQL_ROOT_PASSWORD=Z20seppdsit -d
  mysql:5.7
```

--name 为mysql的实例设置别名。 -p 3307为对外暴露的端口。3306是内部端口

-e MYSQL\_ROOT\_PASSWORD 设置mysql登录密码 -d 以守护进程运行 (后台运行)

## 最后的mysql是镜像名称

```
docker run -d --restart=unless-stopped -p 80:80 -p 443:443
```

```
10.2.33.182:8800/rancher/rancher:v2.4.2
```

### • 常用选项说明

- `-d, --detach=false`
- 指定容器运行于前台还是后台，默认为false
- `-i, --interactive=false`
- 打开STDIN，用于控制台交互
- `-t, --tty=false`
- 分配tty设备，该可以支持终端登录，默认为false
- `-a, --attach=[]`
- 登录容器（必须是以docker run -d启动的容器）
- `-w, --workdir=""`
- 指定容器的工作目录
- `-e, --env=[]`
- 指定环境变量，容器中可以使用该环境变量
- `-p, --publish=[]`
- 指定容器暴露的端口
- `-h, --hostname=""`
- 指定容器的主机名
- `-v, --volume=[]`
- 给容器挂载存储卷，挂载到容器的某个目录
- `--volumes-from=[]`
- 给容器挂载其他容器上的卷，挂载到容器的某个目录
- `--env-file=[]`
- 指定环境变量文件，文件格式为每行一个环境变量
- `--expose=[]`
- 指定容器暴露的端口，即修改镜像的暴露端口
- `--name=""`
- 指定容器名字，后续可以通过名字进行容器管理，links特性需要使用名字
- `--restart="no"`
- 指定容器停止后的重启策略：
  - no：容器退出时不重启
  - on-failure：容器故障退出（返回值非零）时重启
  - always：容器退出时总是重启
- `--rm=false`
- 指定容器停止后自动删除容器(不支持以docker run -d启动的容器)

## 6. 查看docker内部信息

查看内部信息 `docker inspect dockername`

直接查看ip信息 `docker inspect --format='{{.NetworkSettings.IPAddress}}' dockername`