# Docker的基本知识

本次Docker的安装环境是CentOS7.6，Docker必须在CentOS7版本及以上才能运行。

Docker官网：<https://www.docker.com/>

Docker公共镜像仓库：<https://hub.docker.com/>

镜像(Mirroring)：是一种文件存储形式，一个磁盘上的数据在另一个磁盘上存在一个完全相同的副本即为镜像。

容器(Container)：泛指任何可以用于容纳其他物品的工具，可以部分或完全封闭，被用于容纳、储存、运输物品。在云原生技术中，容器就是将软件(镜像)打包成标准化单元，以用于开发、交付和部署。

# 自动安装Docker

# 使用官方安装脚本

curl -fsSL https://get.docker.com | bash -s docker --mirror Aliyun

# 手动安装Docker

# 1.卸载Docker旧版本

sudo yum remove docker \

docker-client \

docker-client-latest \

docker-common \

docker-latest \

docker-latest-logrotate \

docker-logrotate \

docker-engine

# 2.安装需要的安装包

sudo yum install -y yum-utils \

device-mapper-persistent-data \

lvm2

# 3.设置镜像的仓库，这个是国外的(不推荐，很慢)

sudo yum-config-manager --add-repo <https://download.docker.com/linux/centos/docker-ce.repo>

# 4.设置镜像的仓库，这个是阿里云的(推荐)

sudo yum-config-manager \

--add-repo <http://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo>

# 5.安装Docker引擎、容器、组件

sudo yum install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

# 6.启动Docker，如果这一步报错了，执行sudo service docker start手动启动服务

sudo systemctl start docker

# 7.查看Docker的运行状态

sudo systemctl status docker

# 8.简单运行一个镜像来检测Docker引擎的成功安装

sudo docker run hello-world

补充：在使用run命令运行一个镜像时，Docker会首先在本机寻找镜像，找到就运行。否则默认会去Docker Hub上找，找到后会下载到本机并运行，否则就报错-找不到镜像。在设置了新的镜像仓库后，比如阿里云镜像仓库，则将访问Docker Hub改为访问阿里云镜像仓库。

# 卸载Docker

# 1.卸载Docker引擎、客户端、包

sudo yum remove docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin docker-ce-rootless-extras

# 2.删除镜像、容器、卷

sudo rm -rf /var/lib/docker

sudo rm -rf /var/lib/containerd

# Docker镜像命令

# 1.查看本机已有的镜像

docker images

可选项：

-a # 列出所有镜像

-q # 只显示镜像的id



REPOSITORY：表示镜像的仓库源

TAG：镜像的标签

IMAGE ID：镜像ID

CREATED：镜像创建时间

SIZE：镜像大小

同一仓库源可以有多个 TAG，代表这个仓库源的不同个版本，区分不同的镜像是使用REPOSITORY:TAG，即REPOSITORY+TAG。

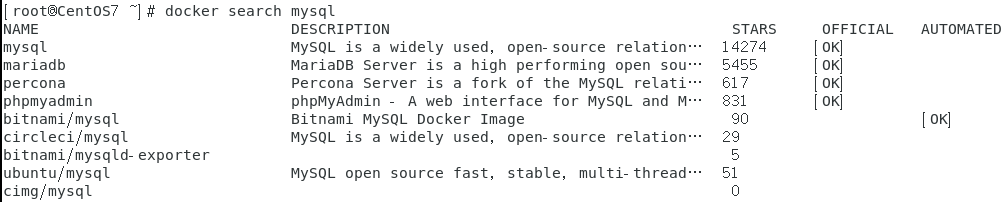
# 2.搜索镜像

docker search 镜像仓库源名称

可选项：

--filter # 根据提供的条件过滤镜像仓库源。

如：docker search mysql --filter=STARS=3000 # 查看STARS大于等于3000的mysql镜像



NAME: 镜像仓库源的名称

DESCRIPTION: 镜像的描述

STARS: 类似 Github 里面的 star，表示点赞、喜欢的意思。

OFFICIAL: 是否 docker 官方发布

AUTOMATED: 自动构建。

# 3.拉取符合镜像名称且tag为latest的镜像

docker pull镜像仓库源名称

拉取指定tag的镜像的示例：

docker pull mysql:5.7 # 拉取tag为5.7的mysql



# 4.根据镜像名称删除镜像

docker rmi 镜像仓库源名称

根据镜像id删除镜像的示例(镜像id可以写多个，用空格隔开)：

docker rmi -f 91b53e2624b4 2be84dd575ee # 镜像id是91b53e2624b4、2be84dd575ee

删除本机所有镜像的示例：

docker rmi -f $(docker images -aq)

# 5.创建镜像

# Docker容器命令

# 1.从仓库拉取一个镜像，这里拉取的是centos

docker pull centos

# 2.运行镜像

docker run [可选项] 镜像名称

可选项说明：

--name=”容器名称” 给容器起名以区分容器

-d 表示以后台方式运行

-it i表示交互式操作，t表示终端，它俩也可以拆开用

-P(大写) 指定端口

-P 主机端口:容器端口

-P 容器端口

容器端口

-p(小写) 表示随机指定端口

以下命令使用 centos 镜像启动一个容器，参数为以命令行模式进入该容器：

docker run -it centos /bin/bash # /bin/bash表示用交互式shell进入容器

# 退出容器，同时关闭容器

exit

# 退出容器，但是不关闭容器

ctrl + p + q

补充：docker容器是由docker镜像创建的运行实例。

# 3.查看正在运行的容器

docker ps [可选项]

可选项：

-a # 查看正在运行的容器，包括曾经运行过的

-n=x # 查看最近x个运行的容器

-q # 只显示容器id

# 4.根据id删除容器

docker rm 容器id # 此命令不能删除正在运行的容器

docker rm -f 容器id # 强制删除容器，正在运行的也可以

docker rm -f $(docker ps -aq) # 强制删除所有容器

docker ps -a -q|xargs docker rm # 删除所有容器

# 5.启动容器

docker start 容器id

# 6.重启容器

docker restart 容器id

# 7.关闭容器，stop会给容器内应用10s时间去停止服务

docker stop 容器id

# 8.强制关闭容器，跟stop不同，kill会立即停止服务

docker kill 容器id

# Docker普通命令

# 1.查看Docker版本信息

docker version

# 2.显示Docker系统信息，包括镜像和容器数

docker info

# 3.查看该命令的解释，包含可选项

docker 命令 --help

# 4.后台启动容器

docker run -d 镜像名称 # 这种后台启动方式会让容器启动后自动关闭，因为容器内没有服

务，所以容器后台启动的方式没有意义，关闭是为了节省资源。

# 5.查看日志

docker logs