



```
命令格式: docker [OPTIONS] COMMAND
```

**OPTIONS:** 

```
-D, --debug
               Enable debug mode
              Daemon socket(s) to connect to
-H, --host list
-l, --log-level string Set the logging level ("debug"|"info"|"warn"|"error"|"fatal") (default "info")
               Print version information and quit
-v, --version
管理类命令
container 管理容器
        管理镜像
image
        管理容器网路
network
        管理swarm模式下的节点
node
       管理docker service (容器副本集)
service
       管理docker service组合
stack
        管理docker swarm集群
 swarm
volume
        管理存储volume
执行指令:
       将本地STDIN,STDOUT,STDERR接入容器
attach
build
       通过dockerfile构建容器镜像
commit 将一个运行中的容器构建成image
       在容器和宿主机之间传递问价
ср
       让运行中的容器执行命令
exec
history
       查看镜像构建历史
        列出镜像列表
images
info
       查看系统信息
inspect 查看docker对象的详细信息
      kill一个container
       查看容器STDOUT、STDERR的日志信息
logs
        挂起容器
pause
       查看容器列表
ps
       拉去镜像
pull
       推送镜像
push
        检索镜像
search
       显示容器实时资源使用情况
stats
      创建镜像tag
 tag
       显示容器进程信息
top
```

## 运行容器命令

-v, --volume list

-w, --workdir string

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

```
Options:
                       CPU权重,按比例分配
-c, --cpu-shares int
                       指定可执行的CPU
  --cpuset-cpus string
  --device-read-bps list
                        限制读吞吐
  --device-read-iops list
                        限制读次数
  --device-write-bps list
                       限制写吞吐
  --device-write-iops list
                       限制写次数
  --entrypoint string
                       覆盖ENTERPOINT
                    设置环境变量
-e, --env list
                    指定环境变量文件
  --env-file list
  --expose list
                     指定监听端口列表
  --health-cmd string
                        健康检查命令
  --health-interval duration
                         健康检查周期
  --health-retries int
                      健康检查重复次数
  --health-start-period duration 等待检查时间
  --health-timeout duration
                         健康检查超时时间
                     保持STDIN是打开的,即使未被接入
-i, --interactive
                         内存限制
-m, --memory bytes
                         总内存限制
  --memory-swap bytes
                       挂载目录
  --mount mount
  --name string
                      指定容器名
  --network network
                        连接至指定容器网络
  --privileged
                     获取容器特权
-p, --publish list
                     发布端口
-P, --publish-all
                  发布全部端口
                   自动产出生命周期结束的容器
  --rm
-t, --tty
                  分配一个伪终端
```

挂载volume

指定容器工作目录