

Docker Cheat Sheet

調度指令 ORCHESTRATE

初始化 swarm 模式並監聽節點 IP 位置

```
docker swarm init --advertise-addr  
10.1.0.2
```

加入 swarm 群組並作為管理者節點

```
docker swarm join --token <manager-token>  
10.1.0.2:2377
```

加入 swarm 群組並作為工作節點

```
docker swarm join --token <worker-token>  
10.1.0.2:2377
```

列出參與群組的節點

```
docker node ls
```

使用者定的映像檔建立服務 · 設定對應通訊埠並部屬 3 個服務實體

```
docker service create --replicas 3 -p  
80:80 --name web nginx
```

列出 swarm 群組內的服務

```
docker service ls
```

擴展服務

```
docker service scale web=5
```

列出運行中服務的任務

```
docker service ps web
```

BUILD

在目前資料夾目錄中使用 Dockerfile 建立容器映像檔並設定標籤

```
docker build -t myapp:1.0 .
```

列出所有儲存在本機 Docker 引擎內的容器映像檔

```
docker images
```

刪除儲存在本機 Docker 引擎內的容器映像檔

```
docker rmi alpine:3.4
```

SHIP

從 Registry 取得容器映像檔

```
docker pull alpine:3.4
```

使用新的名稱及標籤重新標註容器映像檔

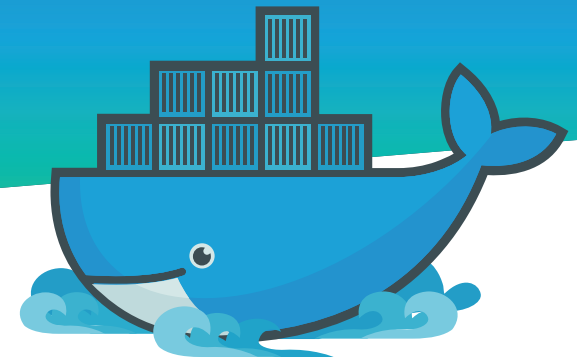
```
docker tag alpine:3.4 myrepo/myalpine:3.4
```

登入指定的 Registry (預設為 Docker Hub)

```
docker login my.registry.com:8000
```

推送容器映像檔至 Registry

```
docker push myrepo/myalpine:3.4
```



RUN

docker run

--rm	關閉容器後自動移除容器
-it	連接至容器至終端機
--name web	命名容器名稱
-p 5000:80	對外公開 5000 埠並映射至 80 埠
-v ~/dev:/code	在容器內建立映射的主機磁碟空間
alpine:3.4	用於實體化容器的容器映像檔
/bin/sh	在容器內運行的指令

發送系統訊號 (SIGTERM) 停止運行中的容器

```
docker stop web
```

發送系統訊號 (SIGKILL) 終止運行中的容器

```
docker kill web
```

建立覆蓋網路並指定網關及網段

```
docker network create --subnet 10.1.0.0/24  
--gateway 10.1.0.1 -d overlay mynet
```

列出網路卡清單

```
docker network ls
```

列出運行中的容器

```
docker ps
```

刪除所有運行中及已停止的容器

```
docker rm -f $(docker ps -aq)
```

在容器內建立 bash 程序並連接至終端機

```
docker exec -it web bash
```

印出最後 100 行的容器 log 訊息

```
docker logs --tail 100 web
```