

## Lab: 建立 docker 映像檔-使用 Dockerfile

透過 `docker build` 指令來建立映像檔

首先觀察目前的路徑

```
# pwd
```

```
/root
```

建立工作目錄

```
# mkdir sakana
```

進入該目錄

```
# cd sakana/
```

建立 Dockerfile

```
# vi Dockerfile
```

```
# 用於註解
```

```
# 後續的流程, 前面都要大寫
```

```
# FROM 基於那個 image
```

```
FROM opensuse:42.1
```

```
# MAINTAINER 維護作者
```

```
MAINTAINER Max Huang < sakana@study-area.org >
```

```
# RUN 要執行的指令
```

```
# 每一行就是一個 image layer, 儘量將同一種行為放同一行 使用 && 來連接
```

```
RUN zypper -n install openssh
```

使用 `docker build` 指令建立 image

這邊 最後面是 Dockerfile 所在目錄, 這邊使用 `.` 目前目錄來指定, 也可以使用絕對路徑

```
# docker build -t testrepo/buildimage .
```

```
Sending build context to Docker daemon 2.048 kB
```

```
Step 1 : FROM opensuse:42.1
```

```
----> 80bd0f661aef
```

```
Step 2 : MAINTAINER Max Huang < sakana@study-area.org >
```

```
----> Running in c4aba3a10ec7
```

```
----> a3f0f611196b
```

```
Removing intermediate container c4aba3a10ec7
```

```
Step 3 : RUN zypper -n install openssh
```

```
----> Running in 702c7f07ba12
```

```
Retrieving repository 'NON-OSS' metadata [.....done]
```

```
Building repository 'NON-OSS' cache [....done]
```

```
Retrieving repository 'OSS' metadata [.....done]
```

```
Building repository 'OSS' cache [....done]
```

```
Retrieving repository 'OSS Update' metadata [.....done]
```

```
Building repository 'OSS Update' cache [....done]
```

```
Retrieving repository 'Update Non-Oss' metadata [...done]
```

```
Building repository 'Update Non-Oss' cache [....done]
```

```
Loading repository data...
```

```
Reading installed packages...
```

```
Resolving package dependencies...
```

```
The following 3 NEW packages are going to be installed:
```

```
libedit0 libncurses6 openssh
```

```
The following 3 packages are recommended, but will not be installed (only required packages will be installed):
```

```
audit openssh-helpers xauth
```

```

3 new packages to install.
Overall download size: 1.2 MiB. Already cached: 0 B. After the operation, additional 5.7 MiB will be used.
Continue? [y/n/? shows all options] (y): y
Retrieving package libncurses6-5.9-53.4.x86_64 (1/3), 349.5 KiB (1020.7 KiB unpacked)
Retrieving: libncurses6-5.9-53.4.x86_64.rpm [.done]
Retrieving package libedit0-3.1.snap20140620-3.2.x86_64 (2/3), 94.9 KiB (247.1 KiB unpacked)
Retrieving: libedit0-3.1.snap20140620-3.2.x86_64.rpm [.done]
Retrieving package openssh-6.6p1-11.1.x86_64 (3/3), 830.7 KiB ( 4.5 MiB unpacked)
Retrieving: openssh-6.6p1-11.1.x86_64.rpm [..done (37.8 KiB/s)]
Checking for file conflicts: [.....done]
(1/3) Installing: libncurses6-5.9-53.4.x86_64 [.....done]
(2/3) Installing: libedit0-3.1.snap20140620-3.2.x86_64 [.....done]
(3/3) Installing: openssh-6.6p1-11.1.x86_64 [.....done]
Additional rpm output:
Updating /etc/sysconfig/ssh...

```

```

---> c823a9e4a1a0
Removing intermediate container 702c7f07ba12
Successfully built c823a9e4a1a0

```

- -t 是 tag 後面為 repository 名稱/image 名稱
- 路徑為 Dockerfile 所在目錄, 跟他上層目錄叫啥名字沒有關係...

觀察映像檔

```

# docker images
REPOSITORY          TAG                IMAGE ID           CREATED            VIRTUAL SIZE
testrepo/buildimage latest            c823a9e4a1a0      3 minutes ago     168.9 MB
testrepo/testimage  vTEST            c840a5f6ade6      12 hours ago      168.9 MB
opensuse             42.1              80bd0f661aef      5 weeks ago       97.74 MB
opensuse             latest            80bd0f661aef      5 weeks ago       97.74 MB
busybox              latest            b41c5284db84      8 weeks ago       1.093 MB

```

Lab: 映像檔管理 - Tag 與刪除映像檔

設定映像檔 Tag

可以透過 `docker tag` 設定映像檔的 TAG

觀察映像檔

```

# docker images
REPOSITORY          TAG                IMAGE ID           CREATED            VIRTUAL SIZE
testrepo/buildimage latest            c823a9e4a1a0      3 minutes ago     168.9 MB
testrepo/testimage  vTEST            c840a5f6ade6      12 hours ago      168.9 MB
opensuse             42.1              80bd0f661aef      5 weeks ago       97.74 MB
opensuse             latest            80bd0f661aef      5 weeks ago       97.74 MB
busybox              latest            b41c5284db84      8 weeks ago       1.093 MB

```

使用 `docker tag` 指令建立 TAG

```
# docker tag c823a9e4a1a0 testrepo/buildimage:tagTest
```

觀察映像檔

```
# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	VIRTUAL SIZE
testrepo/buildimage	latest	c823a9e4a1a0	22 minutes ago	168.9 MB
testrepo/buildimage	tagTest	c823a9e4a1a0	22 minutes ago	168.9 MB
testrepo/testimage	vTEST	c840a5f6ade6	12 hours ago	168.9 MB
opensuse	42.1	80bd0f661aef	5 weeks ago	97.74 MB
opensuse	latest	80bd0f661aef	5 weeks ago	97.74 MB
busybox	latest	b41c5284db84	8 weeks ago	1.093 MB

移除映像檔

如果要移除映像檔, 必須沒有作用中的 container 才可以移除

觀察 container

目前 busybox 沒有 container 使用中, opensuse 有 container 使用中

```
# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

```
# docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
f4b981edb1e5	opensuse	"/bin/bash"	12 hours ago	Exited (0) 26 minutes ago		sleepy_bhabha
3965b5144bdf	opensuse	"/bin/bash"	13 hours ago	Exited (0) 12 hours ago		elegant_brown
1dd047e1ec6b	opensuse	"/bin/bash"	13 hours ago	Exited (0) 13 hours ago		thirsty_archimedes

首先嘗試 移除 busybox 映像檔

```
# docker rmi busybox
```

Untagged: busybox:latest

Deleted: b41c5284db849fbc4d486094a8896ba46a470f9d397ddec08e8a3d5dd4b725cd

Deleted: a5d4c53980c6aec4758bfea6b0cc90938ec253c3758a8b6b345a0d37cc7b35cb

觀察映像檔

```
# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	VIRTUAL SIZE
testrepo/buildimage	latest	c823a9e4a1a0	28 minutes ago	168.9 MB
testrepo/buildimage	tagTest	c823a9e4a1a0	28 minutes ago	168.9 MB
testrepo/testimage	vTEST	c840a5f6ade6	12 hours ago	168.9 MB
opensuse	latest	80bd0f661aef	5 weeks ago	97.74 MB
opensuse	42.1	80bd0f661aef	5 weeks ago	97.74 MB

接下來嘗試刪除 有 container 執行的 image

```
# docker rmi opensuse:42.1
```

Error response from daemon: conflict: unable to remove repository reference

"opensuse:42.1" (must force) - container f4b981edb1e5 is using its referenced image 80bd0f661aef

Error: failed to remove images: [opensuse:42.1]

因為 container 仍然使用中所以不能刪除

觀察 container

```
# docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
f4b981edb1e5	80bd0f661aef	"/bin/bash"	13 hours ago	Exited (0) 34 minutes ago		sleepy_bhabha
3965b5144bdf	80bd0f661aef	"/bin/bash"	13 hours ago	Exited (0) 13 hours ago		elegant_brown
1dd047e1ec6b	80bd0f661aef	"/bin/bash"	13 hours ago	Exited (0) 13 hours ago		thirsty_archimedes

透過 docker rm 來移除 container

```
# docker rm f4b981edb1e5
```

```
f4b981edb1e5
```

觀察 container

```
# docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
3965b5144bdf	80bd0f661aef	"/bin/bash"	13 hours ago	Exited (0) 13 hours ago		elegant_brown
1dd047e1ec6b	80bd0f661aef	"/bin/bash"	13 hours ago	Exited (0) 13 hours ago		thirsty_archimedes

一個一個刪除 container 有時候在實驗環境有點麻煩

所以就把腦筋動到一次刪除所有的 container

docker ps 指令加上 -q 只會列出 container ID

```
# docker ps -aq
```

```
3965b5144bdf
```

```
1dd047e1ec6b
```

但是做這件事的時候要三思而後行

( 會刪除所有的 container, 會刪除所有的 container, 會刪除所有的 container )

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

```
3965b5144bdf
```

```
1dd047e1ec6b
```

觀察 container

```
# docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

接下來就可以將沒有 container 使用中的 image 刪除了