

# AIOT 智慧物聯網學習馬拉松

## ► 使用 Docker 安裝 MongoDB

陪跑專家：王柏鈞



## 重要知識點

## 重要知識點

- 安裝 docker
- 學習 Docker
- 使用 Docker file

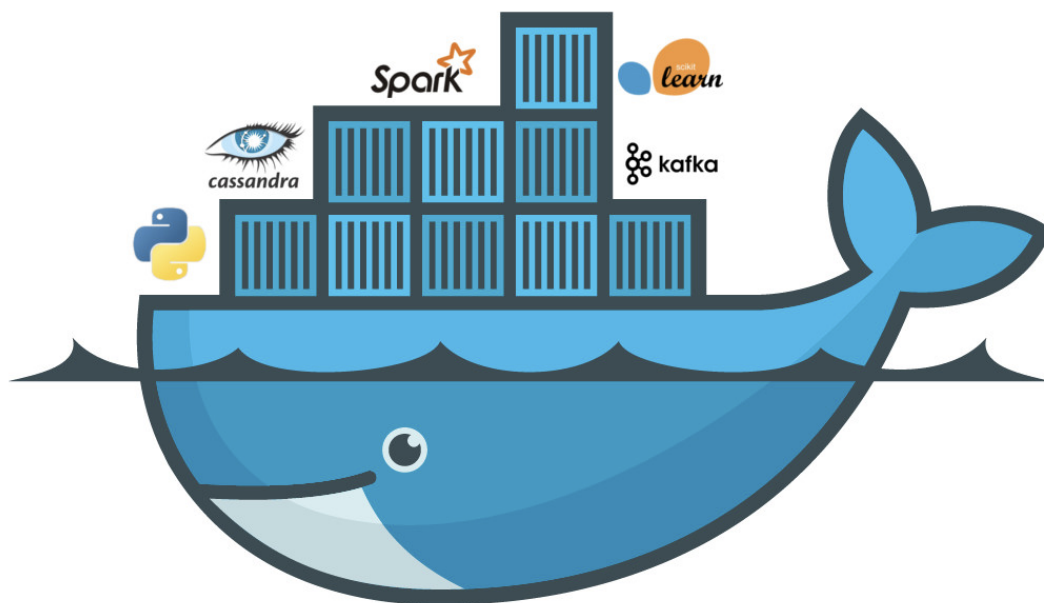
## 何謂 Docker ?

---

Docker 它可以安裝在不同的作業系統，它關鍵的技術是 **容器 (container)**。

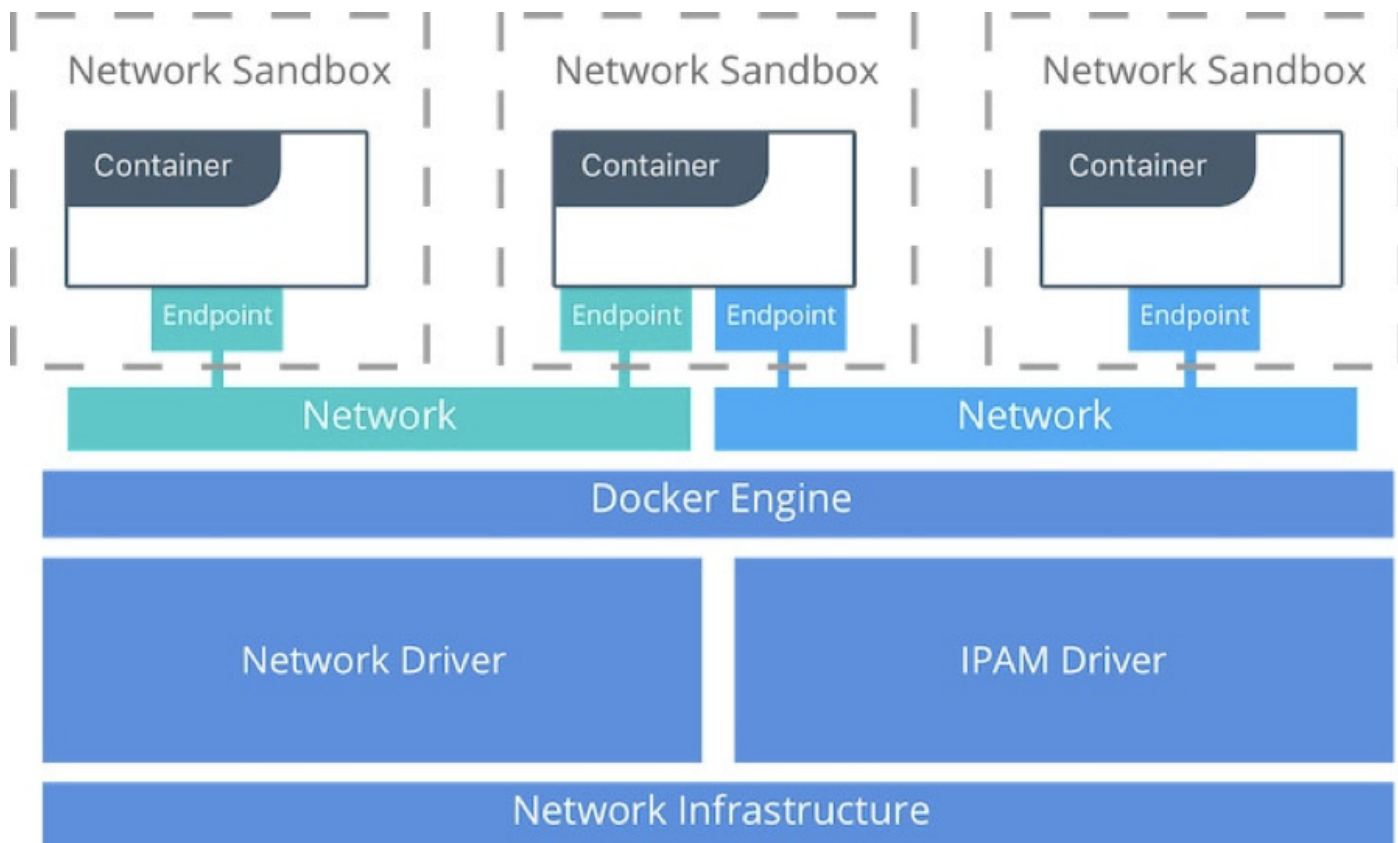
當容器初次建立時會載入某個映象檔(image)，映象檔打包著相關程式碼、函式庫、環境配置檔，然後容器就會執行在一個沙箱的執行環境。

這技術跟模擬整個作業系統的虛擬機器不同，容器是更小的執行環境。



## Docker 的主成分

---



## 架設 Docker

依照自己作業系統安裝

<https://docs.docker.com/get-docker/>

## Get Docker

*Estimated reading time: 1 minute*

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.

You can download and install Docker on multiple platforms. Refer to the following section and choose the best installation path for you.



### Docker Desktop for Mac

A native application using the macOS sandbox security model which delivers all Docker tools to your Mac.



### Docker Desktop for Windows

A native Windows application which delivers all Docker tools to your Windows computer.



### Docker for Linux

Install Docker on a computer which already has a Linux distribution installed.

以 Windows 為例：

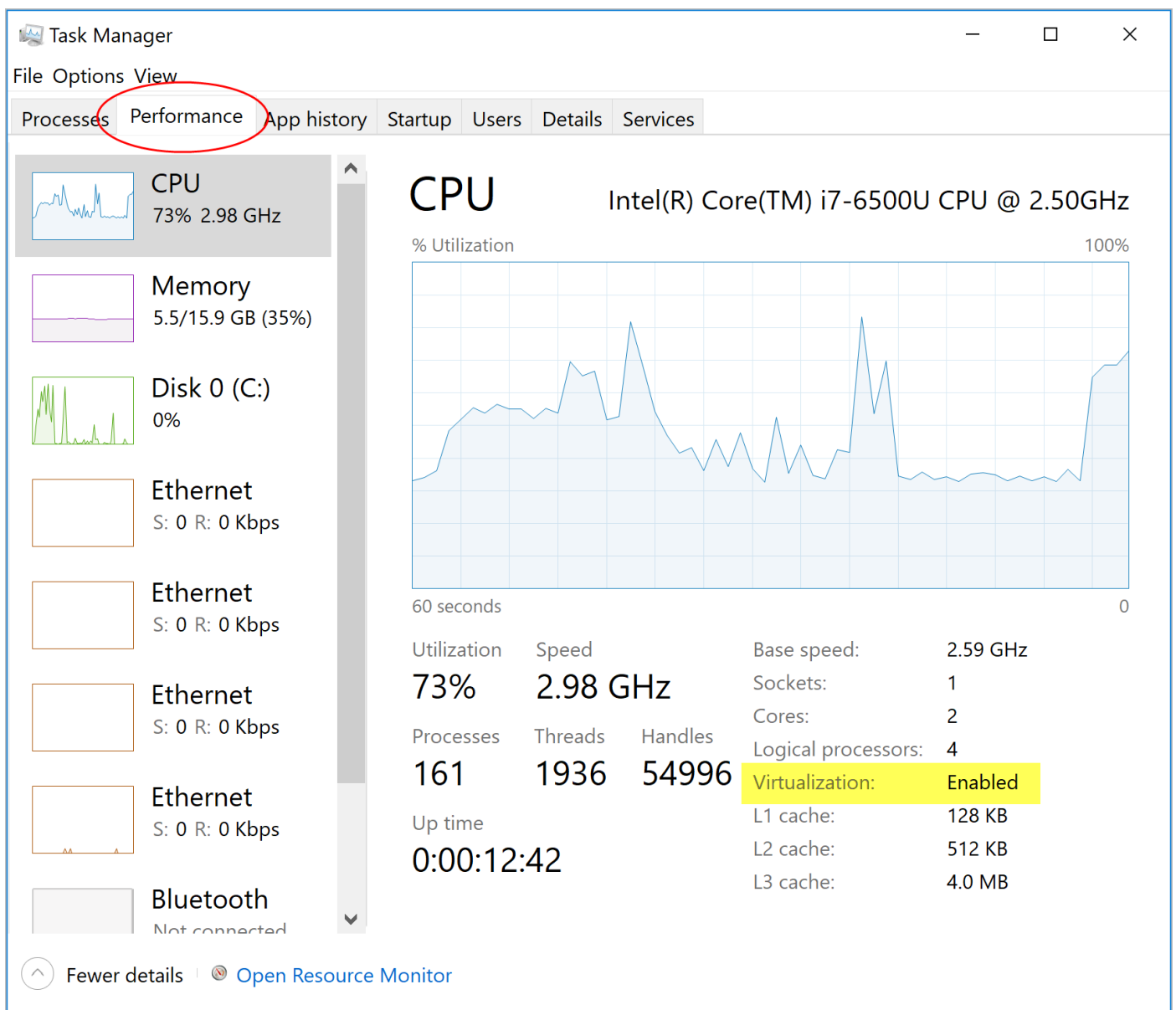
系統要求

Windows 10 64位：專業版，企業版或教育版（內部版本 15063 或更高版本）。

必須啟用 Hyper-V 和 Containers Windows 功能。

要在 Windows 10 上成功運行 Client Hyper-V，需要滿足以下硬件先決條件：

1. 具有 64 位處理器
2. 4GB 系統內存
3. 必須在 BIOS 設置中啟用 BIOS 級硬件虛擬化支持。有關更多信息，請參見 [虛擬](#)





## Docker Desktop for Windows

By [Docker](#)

The fastest and easiest way to get started with Docker on Windows

Edition

Windows

x86-64

### Get Docker Desktop for Windows

Docker Desktop for Windows is available for free.

Requires Microsoft Windows 10 Professional or Enterprise 64-bit. For previous versions get [Docker Toolbox](#).

By downloading this, you agree to the terms of the [Docker Software End User License Agreement](#) and the [Docker Data Processing Agreement \(DPA\)](#).



Get Docker

## 測試 Docker 版本

```
$ docker --version
Docker version 19.03.5, build 633a0ea
```

## 測試 Docker 安裝

```
$ docker run hello-world

Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
ca4f61b1923c: Pull complete
Digest: sha256:ca0eeb6fb05351dfc8759c20733c91def84cb8007aa89a5bf606bc8b315b9fc7
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.
...
```

## Docker 中刪除 Images 鏡像 及 Containers

- 刪除 Docker Image

Docker 提供了 `rmi` 參數，可以用作刪除 images，但在執行前，我們需要知道 image id，先用以下指令找出 image id：

```
$ docker images
REPOSITORY          TAG         IMAGE ID      CREATED       SIZE
mongo               latest      ba0c2ff8d362  3 days ago   492MB
```

找到要刪除的 image 的 image id 後，便可以執行以下指令刪除：

```
$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
mongo                latest             ba0c2ff8d362       3 days ago         492MB
```

@LAPTOP-7HPI0KOB MINGW64 /c/Program Files/Docker Toolbox

```
$ docker rmi ba0c2ff8d362
```

- 刪除 Docker Containers

要刪除 containers，跟上面的做法差不多，只要將參數換成 rm，但在執行前，要先找出 container id

## 何謂 Dockerfile

---

Dockerfile 就可以對應一個可以執行的環境(Image)，而 Dockerfile 就是建置 Docker Image 的腳本

創建 Docker file，預設存在 \$HOME/project/dockerws/nodejs 目錄下

```
FROM google/nodejs
```

```
# Define working directory.
```

```
WORKDIR /data
```

```
#Define default command.
```

```
EXPOSE 3000
```

```
CMD []
```

```
ENTRYPOINT ["/nodejs/bin/npm", "start"]
```

```
1 FROM google/nodejs
2
3 # Define working directory.
4
5 WORKDIR /data
6
7 #Define default command.
8
9 EXPOSE 3000
10
11 CMD []
12
```

- **FROM** : 指定 base image
- **WORKDIR** : 指定 docker 執行起來時候的預設目錄位置
- **EXPOSE** : 指定所有發布的 port
- **CMD** : 指定 Instance 啟動後所要執行的指令
- **ENTRYPOINT** : 指令 Instance 啟動後，程式的進入點

---

## 檢視範例

FROM google/debian:wheezy

RUN apt-get update -y && apt-get install --no-install-recommends -y -q curl python  
build-essential git ca-certificates

RUN mkdir /nodejs && curl <http://nodejs.org/dist/v0.10.33/node-v0.10.33-linux-x64.tar.gz> | tar xvfz - -C /nodejs --strip-components=1

ENV PATH \$PATH:/nodejs/bin

- **RUN**: 指定build過程中所要執行的指令與安裝動作
  - **ENV**: 指令啟動後的環境變數
-

建置 docker file

切換到 Dockerfile 的目錄 (\$HOME/project/dockerws/nodejs)

執行 docker build

這個 build 動作完成後會回覆一個 image id，代表這個 image 建置完成

## Build with TAG

要 commit 到 Docker Hub 的話，會需要針對該 Docker Image 製作一個 TAG，之後使用這個 TAG 做 push... 透過"-t"可以指定該 build 動作到一個 TAG 上

**docker build --tag bulletinboard:1.0 .**

執行以下指令以基於新映像啟動容器：

**docker run --publish 8000:8080 --detach --name bb bulletinboard:1.0**

※ 正確命名映射才能在 Docker Hub 上共用。

具體地說，您必須為影像命名，如 。<Your Docker ID>/<Repository Name>:<tag>





---

## Push 到 Docker Hub

最後透過 docker push，即可將該 image push 到 docker hub，供大家使用

- 確認是否有 Docker ID
  - ※ Set up your Docker Hub account
- 訪問 [Docker 中心註冊](#) 頁面。
- 填寫表單並提交以建立 Docker ID。
- 驗證您的電子郵件地址以完成註冊過程。
- 點選工具列或系統匣中的 Docker 圖示，然後按下「登入/建立 **Docker ID**」。
- 填寫新的 Docker ID 和密碼。成功驗證後，Docker ID 將顯示在 Docker 桌面功能表中，以代替您剛才使用的「登錄」選項。
- 還可以通過鍵入“docker login”從命令行登錄到 Docker 中心。



https://hub.docker.com/signup

## Create a Docker ID.

Already have an account? [Sign In](#)



☐ Send me occasional product updates and announcements.



我不是機器人



reCAPTCHA  
隱私權 · 條款

Sign Up

By creating an account, you agree to the [Terms of Service](#),  
[Privacy Policy](#), and [Data Processing Terms](#).

## 建立 Docker 內式儲存函式庫並推送映像

現在，讓我們創建第一個存儲庫，然後將公告板映射推送到 Docker Hub。

按下選單列中的 Docker 圖示，然後瀏覽到儲存函式庫 > 建立。您將被重定向到 Docker 中心上的"創建儲存庫" 頁面。

將儲存庫名”bulletinboard”稱鍵入，然後單擊頁面底部的“創建”。現在不要填寫任何其他詳細資訊。

請確保您在終端或 PowerShell 的目錄中，然後執行：**node-bulletin-board/bulletin-board-app**

```
docker tag bulletinboard:1.0 <Your Docker ID>/bulletinboard:1.0
```

最後，將映射推送到 Docker 中心：

```
docker push <Your Docker ID>/bulletinboard:1.0
```

The screenshot shows the Docker Hub 'Create Repository' interface. The repository name is 'MongoShare'. The visibility is set to 'Public'. The page includes a 'Pro tip' section with CLI commands for tagging and pushing the image. At the bottom, there are buttons for 'Cancel', 'Create', and 'Create & Build'.

**Create Repository**

**MongoShare**

**Pro tip**  
You may push a new image to this repository using the CLI:

```
docker tag local-image:tagname new-repo:tagname  
docker push new-repo:tagname
```

Make sure to change *tagname* with your desired image repository tag.

**Build Settings (optional)**  
Autobuild triggers a new build with every **git push** to your source code repository. [Learn More](#).

**Connected** **Disconnected**

**Cancel** **Create** **Create & Build**

## 從容器外部訪問

#利用 Dockerfile (參考GITHUB)

根據 [Dockerfile](#) 的最後幾行，默認情況下，數據庫在端口“27017”上可用：

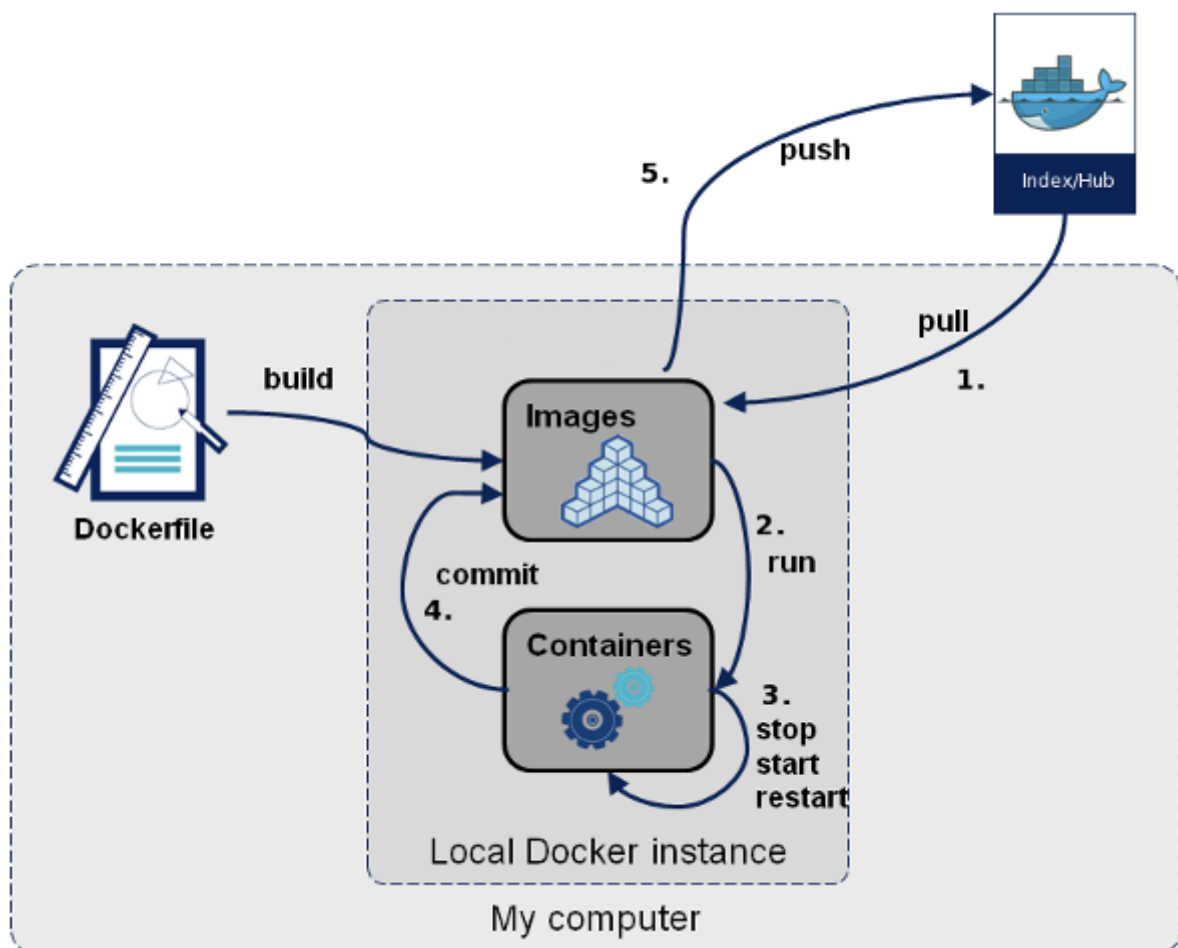
## EXPOSE 27017 CMD ["mongod"]

#打開容器，映射端口，從容器外部訪問Mongo

```
docker run -p 27017:27017 --name my-local-db -d mongo
```

#如果您有節點服務器，在端口3000上運行的服務器將可以通過 HTTP（端口80）使用。  
首先是容器端口，然後是圖像端口。

```
docker run -p 80:3000 my-node-app
```



## 知識點回顧

利用 Docker 分享你的成果

Google 搜尋一些你感興趣的映像檔，甚至自己寫一個 Dockerfile 將你目前的專案打包起來跟別人分享但只要有一個 Dockerfile，你就能使用 `docker build` 來輕鬆建立一個自給自足的 Docker 映像檔

```
FROM python:3.6.3
MAINTAINER your name "email"
COPY ./requirements.txt /app/requirements.txt
WORKDIR /app
RUN pip install -r requirements.txt
COPY . /app
ENTRYPOINT [ "python3" ]
CMD ["app.py"]
```

## 延伸閱讀

---

### Docker Hub

網站：[philipzheng.gitbook](https://philipzheng.gitbook.io/docker-hub/)

### Docker 簡介

本章將帶領你進入 Docker 的世界。

什麼是 Docker？

用它會帶來什麼樣的好處？

- 建立並登陸 Docker Hub，以及目標網站；
- 在目標網站中連接帳戶到 Docker Hub；
- 在 Docker Hub 中 設定一個自動建立；
- 選取一個目標網站中的專案（需要含 Dockerfile）和分支；
- 指定 Dockerfile 的位置，並提交建立。
- 可以在 Docker Hub 的 [自動建立頁面](#) 中跟蹤每次建立的狀態。

《一》	《Docker —— 從入門到實踐》正體中文版	🔍
前言		
Docker 簡介		>
基本概念		>
安裝		>
映像檔		>
容器		>
倉庫		▼
Docker Hub		
私有倉庫		
設定檔案		
資料管理		>
使用網路		>
進階網路設定		>
實戰案例		>
安全		>
Dockerfile		>
底層實作		>
附錄一：命令查詢		
附錄二：常見倉庫介紹		>
附錄三：資源連結		

## Docker Hub

目前 Docker 官方維護了一個公共倉庫 [Docker Hub](#)，其中已經包括了超過 15,000 的映像檔。大部分需求，都可以透過在 Docker Hub 中直接下載映像檔來實作。

### 登錄

可以透過執行 `docker login` 命令來輸入使用者名稱、密碼和電子信箱來完成註冊和登錄。註冊成功後，本地使用者目錄的 `.dockercfg` 中將保存使用者的認證訊息。

### 基本操作

使用者無需登錄即可透過 `docker search` 命令來查詢官方倉庫中的映像檔，並利用 `docker pull` 命令來將它下載到本地。

例如以 centos 為關鍵字進行搜索：

```
1 $ sudo docker search centos
2 NAME                                DESCRIPTION
3 centos                              The official build of CentOS.
4 tianon/centos                       CentOS 5 and 6, created using rinse
5 blalor/centos                       Bare-bones base CentOS 6.5 image
6 saltstack/centos-6-minimal
7 tutum/centos-6.4                    DEPRECATED. Use tutum/centos:6.4 in
8 ...
```

MongoDB 的 document 有以下特性：

- 物件資料：一個 collection 可放入很多 document，且 document 的資料可以是 JSON 或 BSON (Binary JSON)。
- 物件格式：collection 不用預先定義 schema，也就是 document 想塞什麼物件都可以。

插入資料

修改 `POST /api/echo`，利用 **Immediately-Invoked Function Expressions (IIFE)** 立刻執行 `async function()`，寫入完成後才回應 request