

# Web應用程式開發

## Web Application Development

網路簡介  
Introduction to Web

授課教師：林器弘

1

## Course rules

- 請假原則遵照學校規定，記得也mail通知我
- 中途不下課，下列事項不須報備，完成後記得回來
  - 上廁所
  - 吃東西
  - 接電話
  - 要事外找等

2

## 計分方式

2025/9/4

3

3

## 計分方式

- |           |     |
|-----------|-----|
| • 期中考     | 30% |
| • 作業      | 10% |
| • 出席與課堂表現 | 10% |

2025/9/4

4

4

## Outline

- 網路原理
- Web 入門
- 網站發佈上線

5

## 網路原理



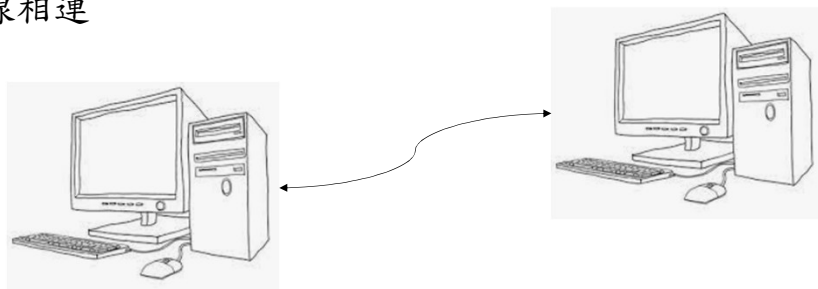
連結裝置  
分享檔案  
分工合作

為何要連結裝置?

6

## 如何傳遞資料

- 電線相連

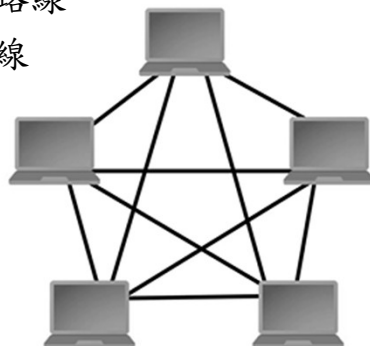


7

## 如何傳遞資料

五台電腦相連需要10條網路線

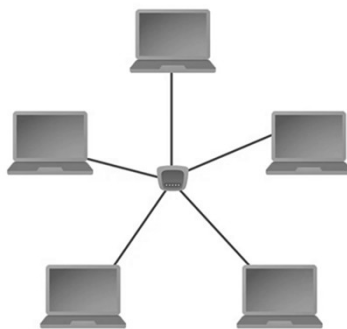
五十台就需要1225條網路線



8

## 如何傳遞資料

- 路由器Router
- 交換器Switch
- 大幅降低所需線路



9

## 如何傳遞資料

- 網路



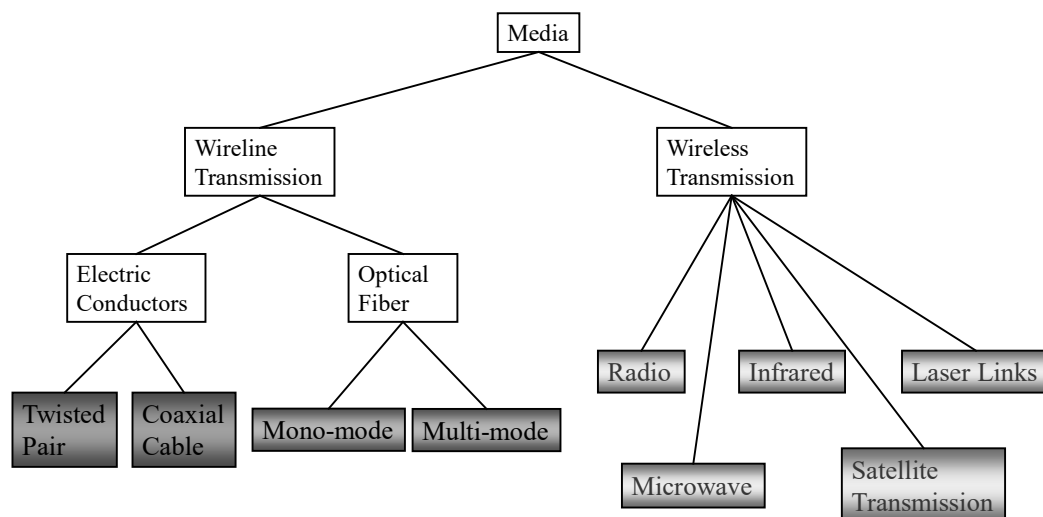
任兩台裝置相連，即形成網路。



世界上可以存在無限多完全不相連的網路。

10

## 網路傳輸介質



11

## 網路設備

- LAN/Internet設備
  - Repeater
  - Hub
  - Bridge
  - Switch
  - Router
  - Gateway
- WAN設備
  - Modem
  - Repeater
  - ADM (Add-Drop Multiplexer)
  - Cross-Connect
  - Switch
  - Multiplexer
  - Concentrator

12

## OSI 7層參考模型

- OSI Model

- OSI - Open Systems Interconnection
- 由國際標準組織ISO (International Standard Organization)制定之網路分層架構標準。
- OSI參考模型，從實體傳輸介質至上層網路應用，由低而高共包括七個層級
- OSI之網路標準，依循OSI參考模型之七層級制定，以便清楚完整規範網路應有之功能元件以及必須遵守的準則。

13

## OSI Model

7	Application	7: 應用層
6	Presentation	6: 表現(展示)層
5	Session	5: 交談(會議)層
4	Transport	4: 傳輸層
3	Network	3: 網路層
2	Data Link	2: 資料鏈結層
1	Physical	1: 實體層

14

## OSI Model

### • 第1層 實體層

- 實體層（Physical Layer）在區域網路上傳送資料框（Data Frame），它負責管理電腦通訊裝置和網路媒體之間的互通。包括了針腳、電壓、線纜規範、集線器、中繼器、網卡、主機介面卡等。

### • 第2層 資料連結層

- 資料連結層（Data Link Layer）負責網路尋址、錯誤偵測和改錯。當表頭和表尾被加至數據包時，會形成資訊框（Data Frame）。數據連結串列頭（DLH）是包含了實體位址和錯誤偵測及改錯的方法。數據連結串列尾（DLT）是一串指示數據包末端的字串。例如乙太網、無線區域網路（Wi-Fi）和通用分組無線服務（GPRS）等。
- 分為兩個子層：邏輯鏈路控制（logical link control, LLC）子層和媒介存取控制（Media access control, MAC）子層。

15

## OSI Model

### • 第3層 網路層

- 網路層（Network Layer）決定數據的路徑選擇和轉寄，將網路表頭（NH）加至數據包，以形成封包。網路表頭包含了網路資料。例如：網際網路協定（IP）等。

### • 第4層 傳輸層

- 傳輸層（Transport Layer）把傳輸表頭（TH）加至資料以形成封包。傳輸表頭包含了所使用的協定等傳送資訊。例如：傳輸控制協定（TCP）等。

16



## OSI Model

- **第5層 會議層**

- 會議層（Session Layer）負責在數據傳輸中設定和維護電腦網路中兩台電腦之間的通訊連接。

- **第6層 表達層**

- 表達層（Presentation Layer）把數據轉換為能與接收者的系統格式相容並適合傳輸的格式。

- **第7層 應用層**

- 應用層（Application Layer）提供為應用軟體而設計的介面，以設定與另一應用軟體之間的通訊。例如：HTTP、HTTPS、FTP、Telnet、SSH、SMTP、POP3等。

17

## Web 入門

18

18

## Web 入門

- 將運用相關工具建構簡易網頁並發布自己的程式碼。
- 替你的第一個網站說故事
- 建立個人網站需要很多功夫。如果你才剛開始接觸網頁設計，我們建議大家可以先從小地方著手。不是要你立刻就寫出跟「Facebook」一樣規模的網站，但自己架一個上線的網站一點都不難，現在就開始吧！

19

19

## Web 入門

- 安裝基本軟體
  - 現有許多工具可建構網站。如果你剛起步，你可能不知從何選擇程式碼編輯器、框架、測試工具等等。我們將透過〈安裝基本軟體〉逐步引領你安裝基本的網頁開發軟體。
- 你的網站看起來會是什麼樣子？
  - 在開始為自己的網站寫程式碼之前，你應該先規劃要呈現哪些資訊？要採用哪種字體與顏色？你可依照〈你的網站看起來會是什麼樣子？〉所提供的簡易方法，照著來規劃網站的內容與設計。
- 與各式各樣檔案打交道
  - 一個網站包含許多檔案：文字內容、程式碼、樣式表、多媒體內容等等。當建立網站時，你需要將這些檔案組合成清晰的架構，並確保它們能彼此互動溝通。〈與各式各樣檔案打交道〉將引領你安排合理的檔案架構，以及你應該注意的問題。

20

## 安裝基本軟體

在本文中，你會知道有哪些 Web 開發的簡易工具，以及正確的安裝方式。

### • 專家都用哪些工具？

- **電腦**，這東西對某些人來說再合理不過了，但可能有人用自己的手機，或在圖書館電腦上讀這篇文章。若要真正著手開發 Web，最好還是弄一台桌電或筆電（Windows、Mac、Linux 都行）。
- **文字編輯器**，用來寫程式碼。可以是文字編輯器（如 Visual Studio Code、Notepad++、Sublime Text、Atom、GNU Emacs、VIM）、也可以是混合式編輯器（如 Dreamweaver 或 WebStorm）。辦公室軟體的文件編輯器並不適合，因為它們依賴許多會干擾瀏覽器排版引擎的隱藏元素。
- **Web 瀏覽器**，可測試寫好的程式碼。目前最常用的瀏覽器有 Firefox、Chrome、Opera、Safari、Internet Explorer 與 Microsoft Edge。你也可以測試自己網站在行動裝置上呈現的效果，若目標讀者分佈甚廣，你可能必須用舊版瀏覽器（如 IE 6-8）測試。
- **圖像編輯器**，如 GIMP、Paint.NET、Photoshop 都可用來為自己的網站添增圖案。

## 安裝基本軟體

- **版本控制系統**，可讓團隊用以協作某個專案、分享程式碼與外部檔案，更可避免編輯作業發生衝突、還可以管理伺服器的檔案。目前 Git 是最常見的版本控制工具；GitHub 與 GitLab 則是最常用的程式碼分享服務。
- **FTP 程式**，可將網頁上傳至比較老的網路託管伺服器，以利使用者檢視——前述的 Git 近年來漸漸取代了 FTP——可用的 (S)FTP 程式有 Cyberduck、Fetch、FileZilla。
- **自動化系統**，如 Grunt 或 Gulp 可自動執行重複性的作業，例如縮減程式碼並執行測試作業。
- 範本、函式庫、框架……等等，都能加速寫出一般功能。
- 及其他更多工具！

## 安裝基本軟體

我實際需要、立刻需要哪些工具？

- 上面一長串看起來好像很嚇人，但其實剛接觸 Web 開發時，不需了解所有的東西。我們先幫你設定最低限度的必要工具：文字編輯器和幾款主流瀏覽器。
- 安裝文字編輯器
- 你的電腦裡很可能已經提供基本的文字編輯器了。Windows 本身就有 記事本；OS X 已提供 文字編輯；Linux 各版本不太一樣：例如 Ubuntu 就有 gedit。
- 而針對 Web 開發，其實有著比 Notepad 或 TextEdit 好很多的工具。推薦使用 PyCharm & Visual Studio Code，因為這個自由編輯器有提供即時預覽、以及程式碼提示。

23

## 安裝基本軟體

Internet Explorer 與當今的 web 不相容，可能會讓專案跑不動。

### 安裝常用瀏覽器

- 目前我們會安裝數款 Web 瀏覽器的桌面版，以利測試我們所寫的程式碼。先在下方找到自己所用的作業系統，再點擊你愛用的瀏覽器連結：
  - Linux：Firefox、Chrome、Opera、Brave。
  - Windows：Firefox、Chrome、Opera、Internet Explorer、Microsoft Edge、Brave。（Edge 是 Windows 10 的預設瀏覽器；如果你用的是 Windows 8 或之後版本，可以安裝 IE 10 或更高版本；否則應安裝其他瀏覽器替代）
  - macOS：Firefox、Chrome、Opera、Safari、Brave。（Safari 是 iOS 與 OS X 的預設瀏覽器）
- 在著手開發之前，應先安裝至少兩款瀏覽器以利後續測試。

24

## 你的網站看起來會是什麼樣子？

- 你的網站看起來會是什麼樣子？會說明你該為網站做的規劃與設計，決定自己的網站該提供哪些資訊？、「該使用哪些字型與色彩？」，以及「網站又該達到哪些目的？」

### 首要：規劃

- 做任何事之前都需要一些想法。你的網站要達到哪種目的？任何 1 個網站均具備基本作用，但首先你應該保持簡單。我們要先寫出包含了標題、圖像，以及數段文字的簡單網頁。
- 開始之前，請你先回答下列問題：
  1. 你網站的主題為何？你喜歡貓貓狗狗？想寫城市遊記？或你愛打電動呢？
  2. 你對某個主題所提供的資訊到哪种程度？寫個標題加上幾段文字，或是在網頁上加個圖片。
  3. 簡單的問：你網站的外觀為何？背景顏色？哪種字型比較合適？一般字型？卡通化的字型？又粗又大的字體？還是要纖細淡化的文字呢？

25

## 你的網站看起來會是什麼樣子？

### 從頭設計

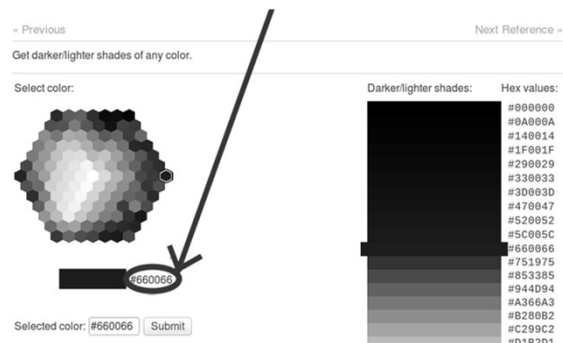
- 接著就是拿紙筆，概略畫出你理想的網站外觀。對自己的第一個簡易網頁，不需考量太多要素或畫得太仔細，應該先養成這個勾勒草圖的習慣。不是要你變成梵谷或畢卡索，但對你以後絕對有幫助！
- 注意：實際的複雜網站，設計團隊也都會先在紙上勾勒草圖，再接著透過圖檔編輯器或 Web 技術弄出數位版的草稿。
- Web 團隊往往編制了圖像設計師與使用者經驗 (User Experience; UX) 設計師各 1 名。圖像設計師當然是負責整合網站的視覺效果；較屬抽象角色的 UX 設計師則要確定使用者與網站的互動方式。

26

## 你的網站看起來會是什麼樣子？

### 主題色彩

- 可透過選色工具找到你喜歡的顏色。當你點擊其中一種顏色，就會看到如下圖「#660066」的 6 位數字，此即代表該顏色的十六進位 (Hexadecimal) 色碼。請另外找個地方記錄此色碼。



27

## 你的網站看起來會是什麼樣子？

- **色彩選擇工具**
- 這是個可以讓人新增、調整和測試網頁顏色的工具。藉由這個工具讓人可以輕鬆將顏色轉換成各種 CSS 支援的格式，包含：HEXA、RGB (Red/Green/Blue) 和 HSL (Hue/Saturation/Lightness)。在 RGB (rgba) 和 HSL (hsla) 格式也支援控制阿爾法通道 (alpha channel)
- 隨著調整每個顏色都會顯示成三種 CSS 的標準格式；此外，基於目前選擇的顏色，一個調色板 HSL 和 HSV、還有 alpha, 被創建。滴管風格的色彩選擇器對話框可以在 HSL 或 HSV 格式中切換。

- [https://developer.mozilla.org/zh-TW/docs/Web/CSS/CSS\\_Colors/Color\\_picker\\_tool](https://developer.mozilla.org/zh-TW/docs/Web/CSS/CSS_Colors/Color_picker_tool)

28

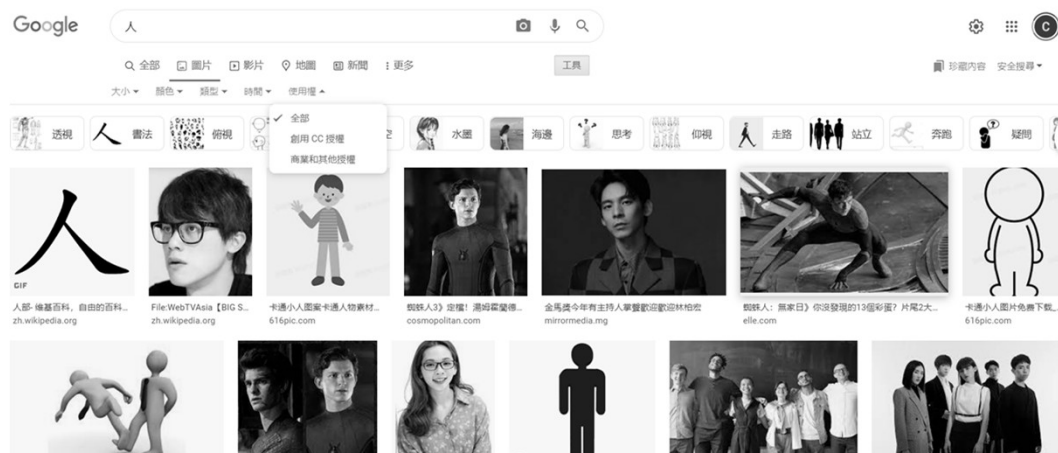
## 你的網站看起來會是什麼樣子？

### 圖片

- 可透過 Google 的圖片搜尋 找到適合的圖片。
  1. 找到你想要的圖片時，點選該圖片。
  2. 按下「查看圖片」按鈕。
  3. 接著在圖片上點擊滑鼠右鍵 (Mac 則是 Ctrl + click)，選擇「將圖片另存新檔...」，將圖片儲存到你熟悉的位置。或是可複製瀏覽器網址列中的網址，以待往後使用。

29

## 你的網站看起來會是什麼樣子？



30

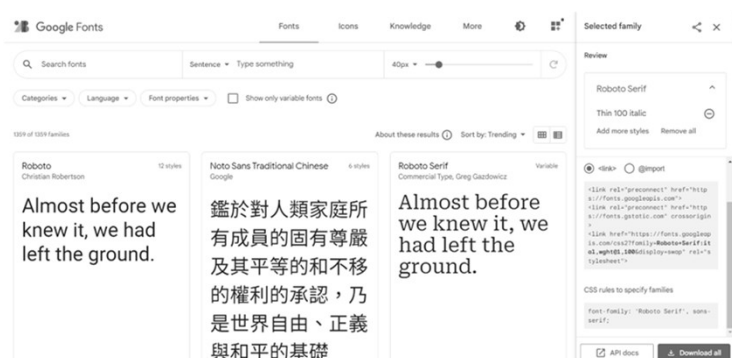
## 你的網站看起來會是什麼樣子？

### 字型

- 選用自己喜歡的字型：

Google Fonts 上捲動清單，直找到你喜歡的字型。你也可以使用左邊的控制功能先行篩選。

- <https://fonts.google.com/>



31

## 與各式各樣檔案打交道

- 一個網站會包含許多檔案: 文字內容、程式碼、樣式表、影音內容.....等。每當你建立一個網站時，你需要將這些檔案在你的電腦上合理架構好。以確保它們能夠互相溝通，並讓內容正常顯示。然後你接著才能將你的網站發佈上線。本篇文章將探討你應該注意的一些議題，以便讓你能夠為你的網站設定好合理的檔案架構。

### 你的網站在哪裡？

- 當你正在你自己的電腦上編輯你的網站時，你應該將所有相關的檔案放在同一個資料夾中，這反映到未來在伺服器上的檔案架構。這個資料夾可以放在任何地方，但你應該會放在一個容易找到的地方，對吧！像是你的桌面，你的家目錄，或是你硬碟的根目錄。
  1. 找到一個地方來存放你的網站專案。首先，建立一個新資料夾並命名為 web-projects (或類似名稱)。這裡將存放你的各種網站專案。
  2. 在上述資料夾底下，建立另一個資料夾來存放你的第一個網站，請將這個資料夾命名為 test-site (或其他有創意的名稱)。

32



## 與各式各樣檔案打交道

### 留意大小寫與空格

- 你將注意到在本篇文章中，我們會要求你在命名檔案或是資料夾時，只使用小寫並且避免使用空格。這是因為：
  1. 許多的電腦尤其是網路伺服器，是大小寫區分的(case-sensitive)。所以假設你放了圖片在你的網站上而路徑是 test-site/MyImage.jpg，然後另一個檔案你想放在 test-site/myimage.jpg，這可能是無法運作的。
  2. 瀏覽器、伺服器、以及各種程式語言對於空格的處理並不是一致的。舉例來說，如果你在檔名中使用了空格，有些系統會將其視為兩個檔名，有些伺服器會將空格替換成 "%20" (這是空格在 URIs 中的表示法)，並破壞了你的連結。我們建議使用底線(underscores)與破折號(dashes)來隔開單字。例如：my-file.html 或是 my\_file.html。
- 也因為這些原因，你應該盡量在命名資料夾與檔案時使用小寫並避免使用空格，這樣一來將能夠減少一些不必要的錯誤。

33

## 與各式各樣檔案打交道

### 你的網站架構應該如何？

- 我們要看看我們的測試網站應該具有什麼樣的架構。我們的網站專案最常見的東西就是一個HTML檔案與專門放圖片、樣式檔案、腳本檔案的資料夾們。讓我們來看看下面：
- index.html: 這個檔案會包含你的首頁內容，也就是別人一進到你的網站時所看到的文字與圖片。使用你的文字編輯器，建立一個新檔案命名為 index.html，並將它存到 test-site 這個資料夾下。
- images folder: 這個資料夾包含了所有網站會用到的圖片，建立一個新資料夾命名為 images，並將它存到 test-site 這個資料夾下。
- styles folder: 這個資料夾包含了能夠設計你的網站的CSS碼(例如：設定文字與背景顏色)，建立一個資料夾命名為 styles，並將它存到 test-site 這個資料夾下。
- scripts folder: 這個資料夾將包含能夠使網站具有互動性的JavaScript程式碼。(例如：按下按鈕後會載入資料)。建立一個資料夾命名 scripts，並將它存到 test-site 這個資料夾下。

Note: 在 Windows 的電腦上，你可能在設定副檔名上會遇到一些困難。因為 Windows 預設會將已知的檔案類型名稱隱藏。一般來說你可以將這項設定關掉，只需要去檔案總管，選擇「資料夾選項」並取消選取「隱藏已知檔案類型的副檔名」，並點選 OK。有關不同版本的 Windows 的設定方法，請利用搜尋引擎搜尋。

34

## 與各式各樣檔案打交道

### • 檔案路徑

要讓一個檔案能夠與另一個檔案"溝通"，你需要提供一個他們之間的相對檔案路徑以讓檔案能夠找到另一個檔案在哪裡。為了要展示，我們將插入一小段的HTML到我們的 index.html 檔案中

1. 複製你選的圖片並放到 images 資料夾中。
2. 打開你的 index.html，並寫下面這段code。

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>My test page</title>
  </head>
  <body>
    <img src="" alt="My test image">
  </body>
</html>
```

35

## 與各式各樣檔案打交道

3. `<img src="" alt="My test image">` 這行是一段將圖片插入到頁面中的 HTML code，我們必須告訴HTML圖片在哪。我們知道圖片在 images 資料夾中，而 images 資料夾就跟 index.html 在同一目錄下。為了要在檔案系統結構中從 index.html 走到我們的圖片，我們需要將檔案路徑設為images/your-image-filename. 舉例來說，我們的圖片命名為 firefox-icon.png，所以這裡的檔案路徑即為 images/firefox-icon.png.
4. 將檔案路徑貼到你的 HTML code 中的 `src=""` 的雙引號之間。
5. 將你的 HTML 檔案存檔，並且滑鼠雙擊HTML檔案來打開它，你應該會看到一個新的網頁並展示著你的圖片！

36

## 與各式各樣檔案打交道

一些有關路徑的規則：

- 要連結一個目標檔案到在同一個目錄下的HTML檔案，只需要檔案名稱就可以了。例如 my-image.jpg。
- 為了要參照到在子目錄下的檔案，在路徑前面加上該目錄的名稱，並接著加上一個斜線(/)。例如： subdirectory/my-image.jpg。
- 為了要連結到一個在 HTML file 的上層目錄的檔案，你需要使用兩個點點(..)。例如：index.html是在test-site 這個目錄的一個子目錄下，而 my-image.png 是在 test-site 這個目錄下，HTML file 要參照 my-image.png 必須使用 ../my-image.png。
- 你可以混和著使用以上的規則，例如 ../subdirectory/another-subdirectory/my-image.png。

Note: Windows的檔案系統會傾向使用反斜線(\)，而非斜線(/)。例如 C:\windows. 這並沒有關係，即使你是在Windows上開發網站，你仍然應該在程式碼中使用斜線(/)。

37

## 網頁三要素

- HTML (骨架)
- CSS (外觀)
- JavaScript (動作)

38

## HTML 基礎

- HTML (Hypertext Markup Language)，中文全名為「超文字標示語言」，是一種用來組織架構並呈現網頁內容的程式語言。網頁內容的組成，可能包含了段落、清單、圖片或表格...等。透過這篇文章，希望能幫助大家對 HTML 及其功能有基本的認識。

39

## HTML 基礎

- HTML 到底是什麼？
- HTML 是一種標記語言 (markup language)，而非一般熟知的程式設計語言；它會告訴瀏覽器該如何呈現你的網頁——單純簡易或是極其複雜的頁面都沒問題。HTML 包含了一系列的元素 (elements)，而元素包含了標籤 (tags) 與內容 (content)，我們用標籤來控制內容的呈現樣貌，例如字體大小、斜體粗體、在文字或圖片設置超連結等。舉例來說，請看看以下這個句子：

```
My cat is very grumpy
```

- 如果我們想讓這個句子自成一個段落，那麼可以在它前後分別加上段落標籤 (<p>)，它就變成一個段落元素了：

```
<p>My cat is very grumpy</p>
```

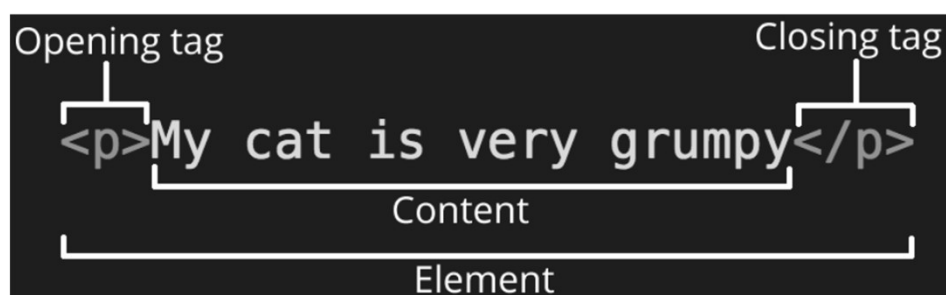
40

## HTML 基礎

- 我們可以看到基本的架構：
- 起始標籤（The opening tag）：先打角括弧，也就是大於、小於的符號「<>」，裡面再放入元素名稱，如上面的例子「<p>」。起始標籤代表這個元素從這裡開始。
- 結束標籤（The closing tag）：與起始標籤一樣，只是在元素名稱前面多了個前置斜線「/」。很容易理解地，內容的最後加上結束標籤，代表這個元素的尾端。在寫HTML時，很容易忘了最後的結束標籤，提醒大家要多注意唷！
- 內容（The content）：這個元素的內容，以上面的例子來說，內容就是這句文字。
- 元素（The element）：由起始標籤、結束標籤、內容所組成。

41

## HTML 基礎



42

## HTML 基礎

- 元素還可以有「屬性 (Attribute)」
- 屬性的組成包含：
  1. 在元素名稱和屬性之間有一個空格（如果有多個屬性，屬性之間也需要有空格）
  2. 屬性名稱後面接著等於符號「=」
  3. 屬性包在起始標籤裡面，如範例所示

Attribute

```
<p class="editor-note">My cat is very grumpy</p>
```

屬性能提供更多的資訊（當然，這個資訊是幫助我們更有效及方便編輯，不會呈現在網頁上），屬性包含了屬性名稱與值，你可以利用屬性設定這個元素的色彩、對齊方式、圖表的格線等等。

43

## HTML 基礎

- 巢狀元素
- 元素裡面可以在放進元素，我們稱之為「巢狀元素 (nesting element)」。  
例如這個句子：「我的貓有夠無敵臭臉」，若你想強調「有夠無敵」，我們就可以把「有夠無敵」這四個字自成一個顯示為粗體的元素 `<strong>`：

```
<p>My cat is <strong>very</strong> grumpy.</p>
```

- 要注意的是，每個元素都有自己的起始和結束標籤，一層一層的包覆。所以最外層是 `<p>`，接著 `<strong>`；先結束 `strong` 元素，所以先寫 `</strong>`，最外面才是 `</p>`。

```
<p>My cat is <strong>very</p></strong> grumpy.</p>
```



44

## HTML 基礎

- 有些元素沒有內容，我們稱為「空元素（empty elements）」。  
以這個圖片元素 `<img>` 為例：

```

```

- 它有兩個屬性，但是沒有結束標籤，也沒有裡面的內容。因為圖片元素是直接把圖檔嵌在 HTML 網頁上。

45

## HTML 基礎

- 讓我們來看看一個完整的 HTML 頁面它所包含的要素
- `<!DOCTYPE html>` — 文件類型（doctype）。在 HTML 發展初期（約莫 1991/2 的年代）文件類型是用來連結一些應遵守的規則，有點像自動校正的功能。然而，現在大家其實不太管文件類型，它就是個必須放在程式碼中的東西，現階段大家只需要知道這點就夠了。
- `<html></html>` — `<html>` 元素，又被視為根元素（root element），包含了所有顯示在這個頁面上的內容。
- `<head></head>` — `<head>` 元素，裡面放的是你想涵括的重要資訊，但不會顯示於網頁瀏覽者眼前的。例如，顯示於搜尋結果的關鍵字、頁面說明、CSS、字元實體集...等。

46

## HTML 基礎

- `<body></body>` — `<body>` 元素，包含了所有會顯示於網頁瀏覽者眼前的內容。無論是文字、圖片、影面、互動遊戲...等。
- `<meta charset="utf-8">` — 這個元素指定了你的文件使用utf-8這種字元編碼，建議大家都要使用這個元素，它會幫助你免去許多文字無法正確呈現的煩惱。
- `<title></title>` — 呈現於網頁瀏覽者眼前的網頁標題。

47

## HTML 基礎

- HTML 文件的架構
- 讓一個完整的HTML頁面它所包含的要素

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>My test page</title>
  </head>
  <body>
    
  </body>
</html>
```

48



## HTML 基礎

- 圖片元素：

- 

```

```

- 如同我們前面提到，圖片元素是直接把圖檔嵌在HTML網頁上，它是透過圖片來源（src，source）這個屬性，提供了連到圖片檔案的路徑。
- 我們也可以加上alt (alternative) 這個屬性。在網頁瀏覽者無法正確看到圖片時，你希望對他們呈現什麼樣的說明文字。這種狀況會發生通常是因為：
- 許多視能障礙的網頁瀏覽者，會使用「Screen Readers」這樣的工具，利用說明文字（alt text）來了解網頁要呈現的圖片內容。
- 就是有些東西出錯了。例如，你誤植了圖片來源的路徑，你可能就會看到類似以下的文字：

49

## HTML 基礎 -標記文字

- 這個段落將為大家介紹如何標記文字（讓這些文字如何呈現）。
- 文件標題 (heading)
- 讓你呈現這些內容的主題，就像一本書有書名、章節名稱和副標題，一份HTML文件也有類似的概念。HTML最多可以有六層的heading，<h1>—<h6>，雖然通常我們只使用3至4層：

```
<h1>My main title</h1>
<h2>My top level heading</h2>
<h3>My subheading</h3>
<h4>My sub-subheading</h4>
```

50

## HTML 基礎 – 段落

- 如上面介紹過的，`<p>` elements 包含文字段落，在呈現一般文字時，這是我們最常用到的。

```
<p>This is a single paragraph</p>
```

51

## HTML 基礎 - 清單

- 清單至少會包含兩個元素，以下是最常見的清單類，無順序性與有順序性的：
  - 無順序性清單（Unordered lists）代表這些項目的順序改變，不影響任何是，例如購物清單。項目會包含在 `<ul>` 裡面。
  - 有順序性清單（Ordered lists）代表這些項目的順序是有意義的，例如食譜裡的製作步驟。項目會包含在 `<ol>` 裡面。
- 每個項目則分別放在 `<li>` (list item) element 裡面。
- 例如，我們想把以下這段文字變成清單：

```
<p>At Mozilla, we' re a global community of technologists, thinkers, and builders working together ... </p>
```

52

## HTML 基礎

- 寫法如下：

```
<p>At Mozilla, we're a global community of</p>

<ul>
  <li>technologists</li>
  <li>thinkers</li>
  <li>builders</li>
</ul>

<p>working together ... </p>
```

53

## HTML 基礎

- 連結 (link)

- 連結對於網頁來說是非常重要的。要加上連結，我們需要用到這個元素 — `<a>` — `a` 代表了「anchor」。要讓文字變成連結的步驟如下：

1. 選擇一些文字，例如「Mozilla Manifesto」。
2. 把他們包在這個`<a>` 元素裡：

```
<a>Mozilla Manifesto</a>
```

54

## HTML 基礎

3. 在<a> element 中加上href attribute這個屬性：

```
<a href="">Mozilla Manifesto</a>
```

4. 屬性質就是你要連結網址：

```
<a href="https://www.mozilla.org/zh-TW/about/manifesto/">Mozilla Manifesto</a>
```

網址的開頭使用https:// 或 http://（網路文字傳送標準的不同）可能會給你不一樣的結果。因此，在寫連結時，請自己先點擊過，確認無誤。

55

## CSS 基本

- 階層樣式表 (Cascading Stylesheets；CSS) 可用以塑造網站的特殊風格。例如這段文字要用一般的黑色，或是改用紅色標明重點？某段重要內容應該置於畫面的何處？想用什麼背景圖片及顏色裝飾你的網站？〈CSS 基本概念〉帶你入門。
- CSS到底是什麼？
- 跟 HTML 一樣，CSS 既非標準程式語言，也不是標記語言，而是一種風格頁面語言 (*style sheet language*)：它能讓你在 HTML 文件中的元素 (element) 上套用不同的頁面樣式 (style)。例如，當想要將 HTML 頁面上所有段落元素 (paragraph elements) 裡的文字全部轉換成紅色，你會在CSS裡寫：

56

## CSS 基本

- 試看看在你的編輯器上建立新的檔案 style.css 並貼上這三行 CSS 程式碼，並存到你的 styles 目錄。
  - 但我們還需要把 CSS 套用在 HTML 文件上。否則 CSS 的樣式效果，不會在瀏覽器的 HTML 檔案顯示。（如果你還未跟上我們的專案，請閱讀 [Dealing with files](#) 和 [HTML basics](#) 以找出你需要什麼）
1. 打開 index.html 文件，然後將下面一行貼到 head，也就是 `<head>` 和 `</head>` 標籤之間。

```
<link href="styles/style.css" rel="stylesheet" type="text/css">
```

57

## CSS 基本

- 存檔 index.html 並且在瀏覽器載入。你應該可以看到下面的頁面。
- 解析 CSS ruleset
- 讓我們深入解析下列的 CSS：



58

## CSS 基本

- 整個架構我們稱為規則集 (rule set)，或是簡稱為規則 (rule) 也可以。  
(也注意名字裡面的單獨部分)
- 選擇器 (Selector)
  - 在這個規則的最前頭為 HTML 的元素名。它將決定你 HTML 裡什麼元素將被你接下來的設定影響 (在這個範例中,就是段落元素 p)。若要改變欲影響的元素，只要更改選擇器就行了。
- 宣告 (Declaration)
  - 單一個規則，例如 color: red; 指定了這個元素的呈現樣貌。
- 屬性 (Properties)
  - 修改屬性是改變你HTML元素的一種方法。(在這範例中,color 是段落 (p) 元素的一種屬性。) 在CSS中, 你可以選擇哪些屬性用來影響 rule.
- 屬性值 (Property value)
  - 屬性值就是位於屬性右邊，在冒號 (:) 之後，從眾多的可能樣式選出一個給予屬性 (範例中就是從眾多的 color 樣式中選出 red)

59

## CSS 基本

- 注意語法其他重要的部分：
- 每一個規則當中，除了選擇器名稱以外，其他都必須被大括號 ( { } ) 給包住。
- 在每一個宣告裡面，屬性跟屬性值之間必須用冒號 (:) 做區分。
- 在每一個規則裡面可以包含有許多宣告，但不同的宣告之間必須使用分號 (;) 來區分。

60

## CSS 基本

- 下面是一個簡單的CSS規則範例。注意每個宣告都是以冒號(:)來指定屬性值，並且宣告之間都是以分號做區分(;)。

```
p {  
  color: red;  
  width: 500px;  
  border: 1px solid black;  
}
```

- 選擇多個元素
- 你可以選擇數種元素（elements）並同時用在同一個 rule set 上。可以用逗號(,) 包含數個選擇器，如：

```
p,li,h1 {  
  color: red;  
}
```

61

## JavaScript 基礎

- JavaScript 是一個可以幫您在網站裡加入互動功能的程式語言（舉例來說，一個遊戲可能會在按鈕按下或資料被輸入表單內時回應、動態更改樣式、以及展示動畫等）。這篇文章會幫助您踏上學習這個令人興奮的語言的旅程，並展示她可以實現的所有可能。

62

## JavaScript 基礎

- 所以 JavaScript 到底是什麼？

- JavaScript 是一個成熟的動態程式語言，應用於 HTML 文件（document）上時，就可以為網頁提供動態的互動功能。JavaScript 是由 Mozilla project、Mozilla Foundation 和 Mozilla Corporation 的創辦人 Brendan Eich 所發明的。
- 你可以用 JavaScript 實現許多事情。你可以先從簡單的特性開始，如跑馬燈、相簿、動態版型、回應按鈕點擊等。在你熟悉的這個程式語言以後，甚至可以製作遊戲、2D 平面以及立體的圖像、資料庫系統等等的應用！

63

## JavaScript 基礎

- JavaScript 本身非常的簡潔，卻也充滿彈性，開發者們已經以 JavaScript 核心為基礎為她撰寫了相當多的工具，讓各位可以感到事半功倍。這些工具包括：
  - 內建在瀏覽器內的應用程式介面（Application Programming Interfaces，APIs）提供了多樣化的功能，像是動態產生 HTML 以及設定 CSS 樣式、擷取以及處理從使用者的網路攝影機錄下的影像、製作立體圖形或是聲音樣本。
  - 第三方 API 允許開發者將他們的網頁與其他如 Twitter 或 Facebook 提供的內容合併在一起。
  - 第三方框架和函式庫允許您將這些元件套用在您的 HTML 文件內，讓您可以迅速地建立網頁或應用程式。

64



## JavaScript 基礎

- 前面所述的功能聽起來令人興奮 JavaScript 是眾多令人感到興奮的網路科技之一，您會因為選擇利用她來製作網頁而進入一個嶄新且充滿創意及力量的次元。
- 但無論如何，要讓 JavaScript 跟 HTML 和 CSS 合作無間的話，可能還要費一些功夫。現在您將會從一些細小的地方開始著手，接著一步步地往前進。首先，我們將會向您展示如何將一些基本的 JavaScript 給加入您的頁面中，並且打造一個「*hello world!*」的範例。

65

## JavaScript 基礎

1. 首先，進入您測試網頁的資料夾中，並建立一個名為 main.js 的檔案，再將她存放於 scripts 資料夾內。
2. 接著，開啟 index.html 檔案，並在 `</body>` 這個結束標籤之前的位置，使用一行新的空間來輸入以下的元素：

```
<script src="scripts/main.js"></script>
```

66

## JavaScript 基礎

3. 我們做的事情，基本上跟新增一個 CSS 的 `<link>` 元素是相同的概念 — 我們將 JavaScript 給導入這個頁面中，讓她來影響 HTML（以及 CSS、還有任何頁面上的東西）。
4. 再來我們把以下的程式碼新增到 `main.js` 檔案內：

```
var myHeading = document.querySelector('h1');  
myHeading.textContent = 'Hello world!';
```

5. 現在請您將修改過的 HTML 和 JavaScript 給存檔，再用瀏覽器讀取 `index.html`。您應該會看到以下的內容：

67

網站發佈上線

68

## 將你的網站發佈上線

- 當你完成你的網頁程式碼後，你需要把它放到網路上，這樣人們才可以搜尋得到。這個章節將介紹如何快速的把你的程式碼放到網路上。

### 有哪些選項？

- 發佈網站並不是一個幾句話就能說得完的課題，主要是因為有太多方法能夠發佈網站。在這個章節中，我們不會介紹所有可能的方法，但是我們會簡單講解三個概念，並從初學者的角度分別說明它們的優缺點，然後一步一步帶你用一個你現階段有辦法完成的方法發佈網站。

### 取得主機（hosting）和網域名稱（domain name）

- 如果想要完全掌控你發佈的網站，那你可能需要花錢買：
  - 主機：跟主機租借商（hosting company）的網路伺服器（web server）租一個放置檔案的空間。你把你建置的網頁檔案放到這個空間中，然後想要連結到網頁的人就能透過網頁伺服器連結到你的網站。
  - 網域名稱(domain name)：人們可以透過這個獨特的網址來尋找你的網站，像是 <http://www.mozilla.org> 或 <http://www.bbc.co.uk>。你需要向網域名稱註冊商(domain registrar)租借網域名稱。

69

## 將你的網站發佈上線

- 除此之外，你還會需要一個 File Transfer Protocol (FTP) 程式（點選 [How much does it cost: software](#) 來取得更多資訊），這樣才能真正的把你建置的網頁檔案傳達給伺服器。FTP 程式很廣泛，但一般來說，你可以用你公司提供的資訊，像是使用者名稱、密碼以及 host name 來登入你的網頁伺服器，它就會以兩個視窗的形式分別顯示你電腦裡的檔案和你網頁伺服器上的檔案，然後你就可以移動你的檔案。

70

## 將你的網站發佈上線

### 租借主機和網域的方法

- 在這裡我們不介紹特定的主機租借商(hosting companies)或網域名稱註冊商(domain name registrars)，你只要搜尋「web hosting」與「domain names」就能找到。所有的註冊商都會提供方法，讓你檢查你想要的網域名稱可否使用、或有沒有被註冊過。
- 你的網路供應商可能會提供有限制的主機，這可以让你發佈一些比較簡易的網站，雖然會有些限制，但嘗試使用它來發佈你的第一個網站也是很不錯的——聯絡並詢問他們！
- 有些公司提供免費的服務，像是 Neocities、Blogger 和 WordPress，但是我要強調，你付出多少得到多少，不過使用它們來發佈你的第一個網站何嘗不是一個理想的方法？而且免費的服務大部分不需要FTP程序來上傳檔案，你只需要用它們提供的介面來移動檔案。
- 有些公司同時提供主機(hosting)和網域(domains)的服務。

71

## 將你的網站發佈上線

- 使用線上工具，像是 GitHub 或 Google App Engine
- 使用工具來發佈網站：
  - GitHub 可以交流程式的平台，它提供你一個空間來存放程式碼，這個空間是基於 Git 的**版本控制系統**，你能夠透過系統共同編輯平台上的程式專案，而這個系統是開放資源，也就是說全世界的人都可以找到你的 GitHub code，包括使用它、從中學習並將它改得更好。GitHub 提供一個非常實用的工具——GitHub Pages，它能讓你發佈網站。
  - Google App Engine 是一個強大的平台，不管是要從頭建置 multi-tiered web 程式還是託管靜態網站，它都能讓你在 Google 的基礎下建置和運行應用程式。點選 How do you host your website on Google App Engine? 以獲得更多資訊。
- 這類工具和託管不同，通常他們都是免費的，不過功能當然也會受限。

72

## 將你的網站發佈上線

- 透過 **GitHub** 發布
- 來看看把網站用 Github Pages 發佈多簡單。
  1. 首先註冊 [GitHub](#) 並驗證電子郵件。
  2. 接著針對要上傳的檔案建立一個 repository。
  3. 在頁面的 Repository name 標籤輸入 username.github.io，username 是指你的用戶名。例如我們的好朋友 bobsmith 就會輸入 bobsmith.github.io。  
另外，請勾選 Initialize this repository with a README 後點選 Create repository。

73

## 將你的網站發佈上線

4. 之後，把網站內容拖曳到 repository 目錄，並勾選 Commit changes。請確定目錄內有 index.html 檔案。
  5. 現在讓瀏覽器連到 username.github.io 來看看你的網站。例如你的用戶名字是 chrisdavidmills，就連到 chrisdavidmills.github.io。
- 請參考 [GitHub Pages Help](#).

74

## 何謂網路伺服器？

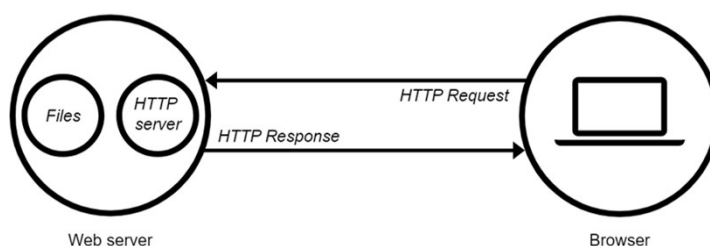
「網路伺服器」（web server）可以指軟體、也可以指硬體、還可以指它們共同運作的狀態。

- 以硬體來說，web server 是存放網路伺服器軟體、還有網站檔案（如 HTML 文件、圖片、CSS 樣式表、JavaScript 檔案）的電腦。它會連上網際網路（Internet）並能和其他連上網的設備做物理數據交換。
- 以軟體來說，web server 包含了一連串控制網路用戶如何訪問託管檔案——至少有 HTTP 伺服器——的檔案。HTTP 伺服器是其中一個部份，它理解 URLs（網路地址）與 HTTP（瀏覽器用來觀察網頁的協議）。它能透過域名（domain name）訪問託管的網站（如 mozilla.org）、並將其內容遞送到終端用戶（end-user）的設備上。

75

## 何謂網路伺服器？

- 以最基本的層面來說，如果瀏覽器需要網路伺服器所託管的檔案，它就需要透過 HTTP 發送對該檔案的請求。如果請求已經傳送到正確的（硬體）網路伺服器，那 HTTP（軟體）伺服器就會接受請求、找出所請求的文件（假若不是接著回傳 404 頁面）、再透過 HTTP 回傳給瀏覽器。



76

## 何謂網路伺服器？

- 要發布網站，你需要一個靜態或動態的網路伺服器。
  - **靜態網路伺服器**（static web server）、或是 stack，由（硬體的）電腦和（軟體的）HTTP 伺服器組成。之所以稱為「靜態」是因為伺服器只會給你的瀏覽器，傳送「事先寫好的」（as-is）檔案。
  - **動態網路伺服器**（dynamic web server）除了靜態網路伺服器以外、還附加了一些軟體：通常是應用伺服器（application server）與資料庫（database）之所以稱為「動態」是因為：應用伺服器會在託管檔案，透過 HTTP 伺服器傳送到瀏覽器之前更新之。
  - 要生成瀏覽器看到的最終網頁，應用伺服器會使用從資料庫讀取資料的 HTML 模板（HTML template）填補之。像 MDN 或維基百科（Wikipedia）這樣的網站也有上千個網頁；但它們全都不是「真的」HTML 文件，而是少數的 HTML 模板、還有龐大的資料庫。如此一來，要維護並傳送資料、都會變得很容易。

77

## 何謂網路伺服器？

### • 託管檔案

- 網路伺服器要儲存網站檔案，也就是所有 HTML 文件、和附屬的 asset：asset 包含了圖片、CSS 樣式表、JavaScript 檔案、字型檔、還有影片。
- 技術上來說，你可以把它們都放在自己的電腦裡面，但放在網路伺服器上面會方便許多，理由是伺服器：
  - 永遠開機並運轉
  - 永遠連上網際網路
  - 永遠都有相同的 IP 地址（不是所有網際網路提供者都給家庭用戶提供固定的 IP 地址）
  - 由第三方提供者維護
- 因此，找到優秀的託管提供者，是建立網站的重點之一。好好探索各大公司提供的服務、並選擇一個符合需求、預算也能負擔的方案（服務的價格從免費到上千美元都有）。你可以在這篇文章找到更多資訊。
- 一旦找到適合的網絡託管解決方案，你只要把文件上傳到網路伺服器就行了。

78

## 何謂網路伺服器？

- 透過 HTTP 溝通
- 網路伺服器會支援 HTTP (Hypertext Transfer Protocol, 超文本傳輸協議)。顧名思義, HTTP 會指定兩台電腦之間, 該如何傳送超文本 (例如 linked web document) 。
  - 協議 (Protocol) 是一套兩台電腦間該如何溝通的規則。HTTP 是文本性、無狀態的協議。
  - 文本性 (Textual)
    - 所有指令都是純文字、人類也容易理解。
  - 無狀態 (Stateless)
    - 無論伺服器還是瀏覽器, 都不會記得他們上一次的溝通。像是伺服器, 如果只依賴 HTTP 的話, 它就不會記得你輸入的帳號密碼、或是在交易中採取了哪些步驟。要完成這樣的任務, 你需要應用伺服器 (我們將在其他文章中介紹這種技術)。

79

## 何謂網路伺服器？

- HTTP 提供了用戶端與伺服器端, 該如何溝通的明確規則。我們將在之後的技術文章內講解 HTTP 本身。目前, 我們會先聚焦在：
  - 只有用戶端能發出 HTTP 請求, 也只有伺服器能接受它。相對地, 也只有伺服器能回應用戶端的 HTTP 請求。
  - 如果透過 HTTP 請求檔案, 用戶端必須提供檔案的 URL。
  - 網路伺服器**必須回應**所有的 HTTP 請求, 最起碼需要回應錯誤訊息。

80



## 何謂網路伺服器？

- 在網路伺服器裡面，HTTP 伺服器負責處理和回答傳入的請求。
  1. HTTP 伺服器接收請求後，會先檢查請求的 URL 是否匹配現有文件。
  2. 如果匹配，網路伺服器會把檔案內容回傳給瀏覽器。不然，應用伺服器會建立需要的檔案。
  3. 如果都沒有用的話，網路伺服器會回傳錯誤訊息給瀏覽器，最常見的就是「404 Not Found」。（這個錯誤很常見，所以許多網頁設計師花了相當多的心力設計 404 錯誤頁面。）

### Not Found

We're sorry, we couldn't find what you were looking for.  
You can try a search or start over on the home page.  
If you were following a document link, you can sign in and create the page. Otherwise, please fix a bug. Thank!

[Sign in with Persona](#)

Powered by Mozilla Firefox 4.0



TikTok

Log in

4 😊 4

Couldn't find this page

Check out more trending videos on TikTok

▶ Watch now

81

## 何謂網路伺服器？

### • 靜態與動態內容

- 大略上來說，伺服器能儲存動態與靜態的內容。「靜態」是指「提供事先寫好的」。靜態網站設定上最簡單，所以我們建議選擇靜態內容，作為你的第一個網站。
- 「動態」指的是伺服器處理內容、甚至從資料庫即時產生。這個解決方案提供了更多靈活性，但技術會變得難以駕馭、令網站明顯複雜許多。
- 以你目前閱讀的頁面為例。在託管的伺服器裡面，有個應用伺服器會從資料庫取得內容、規範化、再把它塞進某些 HTML 模板裡面。這裡的應用伺服器，是以 Python 語言的 Django 框架為基礎，所組建的 Kuma。Mozilla 團隊基於 MDN 的特殊需求開發了 Kuma，不過也有很多相似、但使用其他技術的應用程式。
- 從海量的應用伺服器裡面選一個推薦，是個大難題。有些應用程式會迎合特定的類別，如部落格、百科、電子商務。其他還有更通用的 CMS（content management systems，內容管理系統）。如果要建立動態網站，花點時間找個符合需求的工具。除非想學習伺服器端程式設計，否則不用建立屬於自己的應用伺服器。

82

**Q & A**