

自我介紹--小安 (Andy)

1. 15 年系統工程師
2. 擅長專案開發與優化
3. 多專案技術負責人

經歷

1. 交通部運研所 GIS 系統
2. 教育部社教博識網（含APP）
3. 知名大學多個大型系統建置，含蓋媒體宣傳平台、學生、教師、職員歷程系統、校園實名制管理、資訊安全管理系統（ISMS），多款 APP 開發。

證照

1. AZ-104
2. AZ-204
3. ISO/IEC 27001:2013
4. ISO/IEC 27001:2022

專業領域

1. ASP.NET FRAMEWORK、ASP.NET CORE、Python、PHP 網頁開發技術。
2. Bootstrap、Jquery、Vue 前端框架。
3. SQL SERVER 與 mysql 資料庫開發建置。
4. IIS、Apache、NGINX 伺服器建置管理。

專業領域

- 5. API 開發與撰寫
- 6. 多行動應用程式開發技術，涵蓋 Java、Kotlin、Objective-C、Swift、Flutter，以及 Microsoft Power Apps。
- 7. 多次現有系統上雲 (Azure)、系統優化、成本控管的實務經驗。

章節三 網站開發基礎 (15小時)

5/13-5/27

1. 網頁技術基礎

- HTML、CSS、Javascript

2. 基礎網站安全

- 網站安全基本概念
- XSS、CSRF

章節四 Web 應用程式安全入門 (12小時)

5/29-6/10

維廷老師

1. OWASP Top 10 基礎概念
2. 基本的漏洞掃描
3. 網站防火牆基礎

章節五 資料庫基礎 (9小時) 6/12-6/19

1. 資料庫基本概念
2. 基礎語法
3. 基本的資料庫管理

章節六 資料庫安全入門 (9小時)

6/24-7/1

1. SQL 注入基礎防護
2. 基本的資料庫備份
3. 簡單的權限管理

網頁技術需要知道的基礎知識

1. 瀏覽器 **Client** : FireFox、Chrome、Edge
2. 網站伺服器 **Server** : IIS、Apache
3. 前端 **Client** 與後端 **Server**
4. 靜態 **無互動** 和動態 **有互動** 網頁
5. 網頁技術基礎架構 : **HTML** 、 **CSS** 、 **JAVASCRIPT**

網頁技術需要知道的基礎知識

6. 後端技術：PHP、Python、ASP.NET (Core)

7. 資料庫：MySQL、MS SQL SERVER、SQLite

8. 資訊安全：XSS、CSRF、SQL Injection

網頁技術基礎架構

1. HTML

- Html and Html5

2. CSS

- CSS and CSS3
- Bootstrap

3. Javascript

- JQuery
- Datatables

網站 = 房子

1. 基本結構：HTML
2. 裝飾風格：CSS
3. 智慧型系統：Javascript

[Search: "learn the foundations of html css javascript"](#)

1. 不載入 Javascript

- F12 > Settings > 停用 Javascript

2. 不載入 CSS

- index.css
- sys_index.css

開發工具

1. **Visual Studio Code**
2. Visual Studio (適用 ASP.NET (2017) 或 Core (2022) 開發)
3. **WebStorm**
4. Atom

開發工具

1. 安裝 Visual Studio Code 主程式

2. 套件

- 中文套件
- Live Server
- 程式碼提示工具
- 美化及輔助閱讀
- AI 套件 (Copilot、Codeium)

什麼是 HTML

1. 用來告訴瀏覽器如何呈現網頁的標籤語言
2. 目前版本：HTML5
3. 構成：`<h1 style='font-size:14;'>這是一個標題</h1>`
 - `<h1>` 開始標籤， `</h1>` 結束標籤
 - `style` 屬性
 - `這是一個標題` 內容

實作：寫一個 HTML

1. 新增 hello.html

```
<!DOCTYPE html>
<html lang="zh-Hant-TW">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body><h1>Hello Html</h1></body>
</html>
```

實作：用 Emmet 的方式寫 HTML

1. 清除 hello.html
2. 輸入快速鍵 `!`
3. 輸入快速鍵 `h1`
4. 輸入 `Hello Html`
5. 啟動 Live Server

標籤 `<html>`

1. 根元素 (Root Element)，包含了整個文件的內容。
2. `<html lang="en">` `lang` 是一個屬性，用來指定 Html 的語言，例如 `en` 代表英文，`ja` 代表日文、`zh-Hant-TW` 台灣使用的正體中文。
3. `<html>` 標籤通常包含 `<head>` 和 `<body>` 標籤。
4. 在 HTML5 中，`<!DOCTYPE html>` 用來告訴瀏覽器這是一個 HTML5 文件。

標籤 `<head>`

1. HTML 文件的 Head Element。
2. 瀏覽器不會將 head 的內容顯示在畫面上。
3. Head 的內容通常是：
 - `<title>`: 瀏覽器的頁籤名稱
 - `<meta>`: Meta 資料，例如文字編碼、關鍵字、描述等
 - `<link>`: CSS 連結
 - `<script>`: JS 連結

<meta name='viewport'>

1. `initial-scale` : 初始縮放比例
2. `maximum-scale` 與 `minimum-scale` : 預設 1.0，最大（小）縮放比例
3. `user-scalable` : 允許使用者縮放，預設 `yes`
4. `width` 與 `height`

```
<meta name="viewport"  
content=  
"width=device-width, initial-scale=1.0, minimum-scale=0.5,user-scalable=no">
```

IE 相容

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">
```

1. 現代渲染模式：強制 Internet Explorer 使用最新的渲染標準和引擎來顯示網頁，以獲得最佳的兼容性和性能。
2. 避免顯示錯誤：有些舊版的 Internet Explorer 可能會以慣性自動選擇兼容模式來顯示網站，可能導致顯示問題或 JavaScript 錯誤。此標籤協助避免這類問題。
3. 效果？
4. `content`：IE=9；IE=8；IE=7

標籤 `<body>`

1. HTML 文件的 Body Element，它包含了 Html 文件主體內容。
2. 一個 html 僅有一個 Body 元素。
3. Body 元素可以包含多種標籤，例如 `<p>`，``，``，`<table>` 等。

標籤 `<h1>` ~ `<h6>`

1. `<h1>` 到 `<h6>` HTML 元素代表六個級別的章節標題。
2. `<h1>` 是最高的章節級別
3. `<h6>` 是最低的
4. 預設的情況下，顯示由大到小的順序為 `<h1>` 到 `<h6>`。

實作：撰寫一個 h1-h6 的 Html

1. 新增檔案 h1-h6.html
2. 在 `<body></body>` 插入 `<h1></h1>` 到 `<h6></h6>` 標籤，並輸入內容
3. 啟動 Live Server 觀看結果

實作：使用 Emmet 撰寫一個 h1-h6 的 Html

1. 新增檔案 h1-h6.html
2. 在 `<body></body>` 插入 `h$*6>lorem`
3. 在 `<body></body>` 插入 `h$*6{hello $}`
4. 啟動 Live Server 觀看結果

Block and Inline Level Content

1. Inline Level Content 是指元素在一行內顯示，不會佔據整行。
2. Block level Content 是指元素會佔據整行，並且會在前後添加換行。

Block and Inline Level Content

1. Inline Level Content

```
123456789
```

2. Block level Content 是指元素會佔據整行，並且會在前後添加換行。

```
123

456

789
```

區塊 Block Level Content

1. `<div>`: 用於組織其他元素的區塊容器。
2. `<p>`: 段落，用於包含文字的區塊。
3. `<h1>` - `<h6>`: 標題元素，表示章節標題，`<h1>`。
4. ``、`` 和 ``: 無序清單、有序清單和清單項目。
5. `<table>`: 表格，用於顯示表格資料。

區塊 Html5 Block Level Content

1. `<header>`: HTML5 新增，用於頁面或區段的頭部內容。
2. `<footer>`: HTML5 新增，用於頁面或區段的底部內容。
3. `<section>`: HTML5 新增，用於頁面或文件中的區段。
4. `<article>`: HTML5 新增，用於獨立的内容塊。
5. `<main>`: HTML5 新增，用於主要的內容塊。

行內 Inline level Content

1. ``: 用於行內內容的通用容器。
2. `<a>`: 超連結，用於建立鏈接。
3. ``: 表示強調，通常以斜體顯示。
4. ``: 表示重要性，通常以粗體顯示。
5. ``: 圖像，用於嵌入圖片。
6. `<input>`: 用於用戶輸入的元素。
7. `<label>`: 用於表單元素的標籤。
8. `<button>`: 用於觸發操作的按鈕。

實作：撰寫 Block and Inline 的 Html

1. 新增 BlockInline.html
2. 在 `<body></body>` 插入 `<div></div>`
3. 在 `<div></div>` 插入數個 `<p></p>` 並插入 `` 及內容
4. 啟動 Live Server 觀看結果

實作：使用 Emmet 撰寫 Block and Inline 的 Html

1. 新增 BlockInline.html
2. 在 `<body></body>` 插入 `div>p*6{hello p $}`
3. 插入 `div>span*6{hello span $}`
4. 啟動 Live Server 觀看結果

標籤 `
` 與 `<hr />`

1. `
` : 換行
2. `<hr />` : 水平分隔線
3. 實作 brhr.html 並觀看結果

```
123<br />456<hr />789
```

標籤 `<a>`

1. 超連結、Inline Level

2. 屬性：

- `href`：連結到的網址
- `target`：
 - `_self`：預設值，在相同的分頁中開啟連結
 - `_blank`：在新的分頁中開啟連結

3. 實作 a.html 並觀看結果

標籤 ``

1. 圖片

2. 常用格式：`jpg`、`png`、`gif`、`svg`、`base64`

3. 屬性：

- `src`：圖片的網址
- `alt` and `title`：無障礙重要元素，提供圖片的文字描述
- `width` and `height`：圖片的寬高

4. 實作 `img.html` 並觀看結果

5. How to display base64 images in html?

Html 使用 base64 圖片特性

1. 可嵌入於 HTML 文件，減少 HTTP 請求次數。
2. 適用於小型圖片，避免過度增加文件大小。
3. 提供即時顯示，減少載入延遲。
4. 不易被外部鏈接影響，增加圖片穩定性。
5. 支援多種圖像格式，如 PNG、JPEG。

文字樣式的 Html 標籤

1. ``: 粗體
2. `<i>`: 斜體
3. `<u>`: 底線
4. ``: 強調重要性，通常以粗體顯示
5. ``: 表示強調，通常以斜體顯示
6. `<mark>`: 標籤或突顯
7. `<small>`: 顯示小號文字
8. ``: 表示被刪除的文字

清單標籤

1. ``：無序清單，例如：Menu、目錄、購物清單等。
2. ``：有序清單，例如：排行榜、步驟。
3. `<dl><dt><dd>`：定義清單，例如：詞彙解釋、FAQ 等。

標籤 ``

1. `style` 屬性 `list-style-type` :

- `disc` : 實心黑點
- `circle` : 空心黑點
- `square` : 實心黑方塊
- `none` : 不使用
- 其他 :

2. 實作

屬性 `list-style-type` :

1. `decimal` 十進位數字
2. `lower-roman` 小寫羅馬數字
3. `upper-roman` 大寫羅馬數字
4. `lower-greek` 小寫希臘字母
5. `trad-chinese-informal` 中文一二三
6. `'-'` 使用自訂符號「-」（單引號內為自訂符號）

標籤 ``

1. 屬性 `type` :

- `a` : 小寫拉丁字母
- `A` : 大寫拉丁字母
- `i` : 小寫羅馬數字
- `I` : 大寫羅馬數字
- `1` : 阿拉伯數字

2. 實作

標籤 `<table>`

1. `<table>`：表格的根元素。
2. `<caption>`：表格的標題。
3. `<thead>`、`<tbody>`、`<tfoot>`：表格的頭部、主體和尾部。
4. `<tr>`：表格的行。
5. `<th>`、`<td>`：表格的欄位，`<th>` 會加粗字體。
6. 屬性：`colspan`、`rowspan`
7. 實作
8. 快速鍵：`table>caption+tr>th+th^tr>th+td`

標籤 `<form>`

1. 提供 User 輸入資料使用的元素容器，將輸入資料傳送到伺服器端
2. 新 Javascript 前端框架（vue、angular、react 等）不需直接使用
3. `id`：欄位 id
4. `name`：欄位名稱
5. `action`：目標網址
6. `method`：GET 或 POST
7. `enctype`：
 - `application/x-www-form-urlencoded`：預設
 - `multipart/form-data`：上傳檔案

標籤 `<input>`

1. 輸入

2. 可根據不同的 type 切換不同的輸入型態

- `type` : 預設，文字輸入
- `password` : 密碼輸入，文字不顯示
- `submit` : 送出按鈕，點擊後 Form 會送出
- `hidden` : 隱藏的輸入，不存在於畫面上
- `radio` : 選項按鈕，點擊後會被選擇
- `checkbox` : 核取方塊，點擊後會被選擇

Html5 新增元素

1. `<header>` 頁首元素
2. `<nav>` 導航連結元素
3. `<main>` 主要內容元素
4. `<section>` 章節元素
5. `<article>` 文章元素
6. `<aside>` 側邊元素
7. `<figure>` 引用元素
8. `<footer>` 頁底元素

使用 AI 產生 Html5 的文件

1. 新增 html5.html
2. 快速鍵 **!**
3. 請AI生成將以下標籤 `<header>` 、 `<nav>` 、 `<main>` 、 `<section>` 、 `<article>` 、 `<aside>` 、 `<figure>` 、 `<footer>` 加入 `<body>` 中，並產生內容，