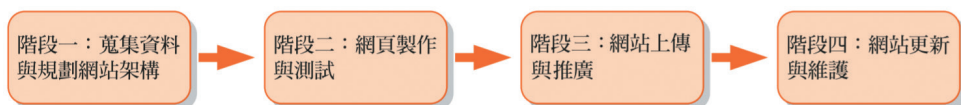


## 1.1 網站建置流程

網站建置流程大致上可以分成如下圖的四個階段，以下就為您做說明。



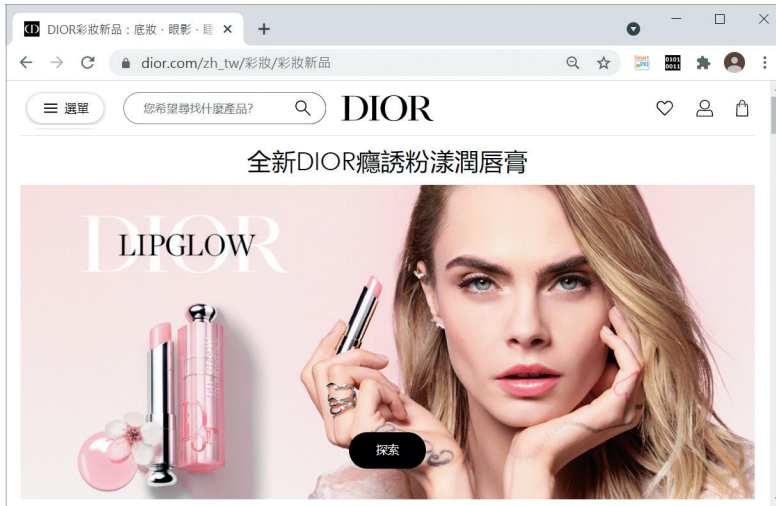
### 1.1.1 階段一：蒐集資料與規劃網站架構

階段一的工作是蒐集資料與規劃網站架構，除了釐清網站所要傳達的內容，更重要的是確立網站的目的、功能與目標族群，然後規劃出組成網站的網頁（裡面可能包括文字、圖形、聲音與視訊），並根據目的、功能與目標族群決定網頁的呈現方式。

下面幾個問題值得您深思：

- ✓ 網站的目的是為了銷售產品或服務？塑造並宣傳企業形象？還是方便業務聯繫或客戶服務？抑或技術交流或資訊分享？若網站本身具有商業用途，那麼您還需要進一步瞭解其行業背景，包括產品類型、企業文化、品牌理念、競爭對手等。
- ✓ 網站的建置與經營需要投入多少時間、預算與資源？您打算如何行銷網站？有哪些管道及相關的費用？
- ✓ 網站將提供哪些資訊或服務給哪些對象？若是個人的話，那麼其統計資料為何？包括年齡層分佈、男性與女性的比例、教育程度、職業、婚姻狀況、居住地區、上網的頻率與時數、使用哪些裝置上網等；若是公司的話，那麼其統計資料為何？包括公司的規模、營業項目與預算。

關於這些對象，他們有哪些共同的特徵或需求呢？舉例來說，彩妝網站的使用者可能鎖定為時尚愛美的女性，所以首頁往往呈現出豔麗的視覺效果，而購物網站的使用者比較廣泛，所以首頁通常展示出琳瑯滿目的商品。



彩妝網站的首頁往往呈現出豔麗的視覺效果



購物網站的首頁通常展示出琳瑯滿目的商品

- ✓ 網站的獲利模式為何？例如銷售產品或服務、廣告贊助、手續費或其它。
- ✓ 網路上是否已經有相同類型的網站？如何讓自己的網站比這些網站更吸引目標族群？因為人們往往只記得第一名的網站，卻分不清楚第二名之後的網站，所以定位清楚且內容專業將是網站勝出的關鍵，光是一味的模仿，只會讓網站流於平庸化。

## 1.1.2 階段二：網頁製作與測試

階段二的工作是製作並測試階段一所規劃的網頁，包括：

### ① 網站視覺設計、版面配置與版型設計

首先，由**視覺設計師** (Visual Designer) 設計網站的視覺風格；接著，針對 PC、平板或手機等目標裝置設計網頁的版面配置；最後，設計首頁與內頁版型，試著將圖文資料編排到首頁與內頁版型，如有問題，就進行修正。

### ② 前端程式設計

由**前端工程師** (Front-End Engineer) 根據視覺設計師所設計的版型進行「切版與組版」，舉例來說，版型可能是使用 Photoshop 所設計的 PSD 設計檔，而前端工程師必須使用 HTML、CSS 或 JavaScript 重新切割與組裝，將圖文資料編排成網頁。

切版與組版需要專業的知識才能兼顧網頁的外觀與效能，例如哪些動畫、陰影或框線可以使用 CSS 來取代？哪些素材可以使用輪播、超大螢幕、標籤頁等效果來呈現？響應式網頁的斷點設定在多少像素？圖文資料編排成網頁以後的內容是否正確等。

此外，前端工程師還要負責將後端工程師所撰寫的功能整合到網站，例如資料庫存取功能、後端管理系統等，確保網站能夠順利運作。

### ③ 後端程式設計

相較於前端工程師負責處理與使用者接觸的部分，例如網站的架構、外觀、瀏覽動線等，**後端工程師** (Back-End Engineer) 則是負責撰寫網站在伺服器端運作的資料處理、商業邏輯等功能，然後提供給前端工程師使用。

### ④ 網頁品質測試

由**品質保證工程師** (Quality Assurance Engineer) 檢查前端工程師所整合出來的網站，包含使用正確的開發方法與流程，校對網站的內容，測試網站的功能等，確保軟體的品質，如有問題，就讓相關的工程師進行修正。

## 1.2 開發適用於不同裝置的網頁

隨著無線網路與行動通訊的蓬勃發展，行動上網的比例已經大幅超越 PC 上網，這意味著傳統以 PC 為主要考量的網頁設計思維必須要改變，因為行動瀏覽器雖然能夠顯示大部分的 PC 網頁，卻經常會遇到下面幾種情況：

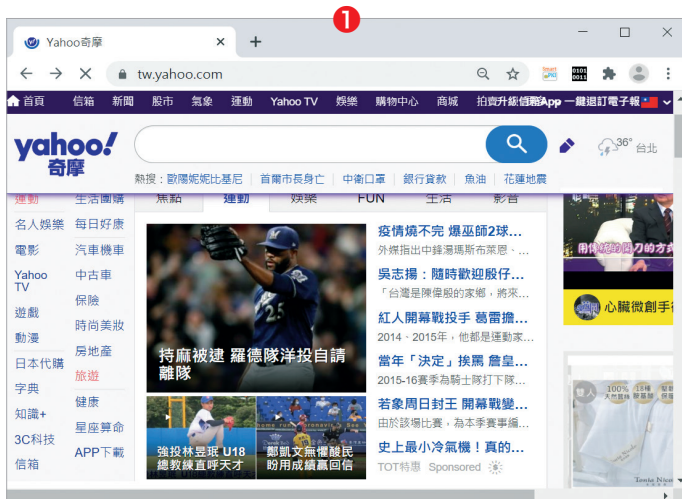
- ✔ 行動裝置的螢幕較小，使用者往往得透過頻繁的拉近、拉遠、捲動，才能閱讀網頁的資訊，相當不方便。
- ✔ 行動裝置的執行速度較慢、上網頻寬較小，若網頁包含太大的圖片或影片，可能耗時過久無法順利顯示。
- ✔ 行動裝置的操作方式是以觸控為主，不再是傳統的滑鼠或鍵盤，因此，PC 網頁到了行動裝置可能會變得不好操作，例如網頁尺寸較大，超連結層次較多，按鈕太小不易觸控或沒有觸控回饋效果，以致於使用者重複點按。
- ✔ 行動裝置不支援 Flash 動畫，但相對的，行動瀏覽器對於 HTML5 與 CSS3 的支援程度則比 PC 瀏覽器更好。

為此，愈來愈多人希望開發適用於不同裝置的網頁，常見的做法有兩種，分別是「針對不同裝置開發不同網站」和「響應式網頁設計」。

### 1.2.1 針對不同裝置開發不同網站

為了因應行動上網的趨勢，有些網站會針對 PC 開發一種版本的網站，稱為「PC 網站」，同時亦針對行動裝置開發另一種版本的網站，稱為「行動網站」，兩者的網址不同，網頁內容也不盡相同，當使用者連線到網站時，會根據上網裝置自動轉址到 PC 網站或行動網站。

舉例來說，左下圖是 Yahoo! 奇摩的 PC 網站 (<https://tw.yahoo.com/>)，而右下圖是 Yahoo! 奇摩的行動網站 (<https://tw.mobi.yahoo.com/>)，對於 Yahoo! 奇摩這種資訊瀏覽類型的網站來說，其行動網站除了著重執行效能，資訊的分類與動線的設計更是重要，才能帶給行動裝置的使用者直覺流暢的操作經驗。



① Yahoo! 奇摩的 PC 網站      ② Yahoo! 奇摩的行動網站

這種做法最主要的優點是可以針對不同裝置量身訂做最適合的網站，不必因為要適用於不同裝置而有所妥協，例如可以保留 PC 網頁所使用的一些動畫或功能，可以發揮行動裝置的特點，同時網頁的程式碼比較簡潔。

雖然有著前述優點，而且也不乏大型的商業網站採取這種做法，不過，這會面臨下列問題：

### ✓ 開發與維護成本隨著網站規模遞增

當網站規模愈來愈大時，光是針對 PC、平板電腦、智慧型手機等不同裝置開發專屬的網站就是日益沉重的工作，一旦資料需要更新，還得一一更新個別的網站，不僅耗費時間與人力，經年累月下來也容易導致資料不同步。

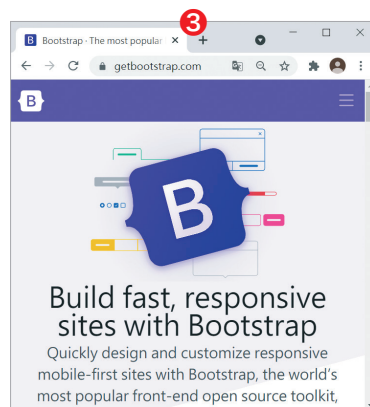
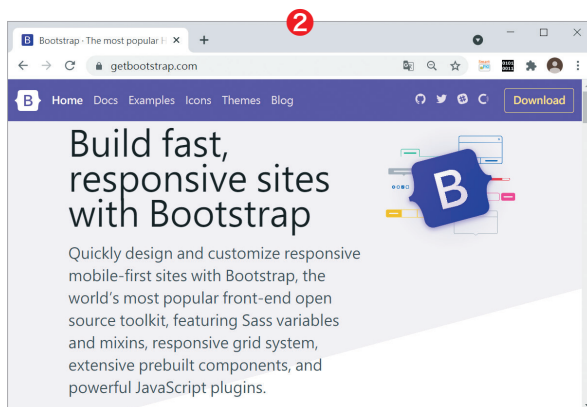
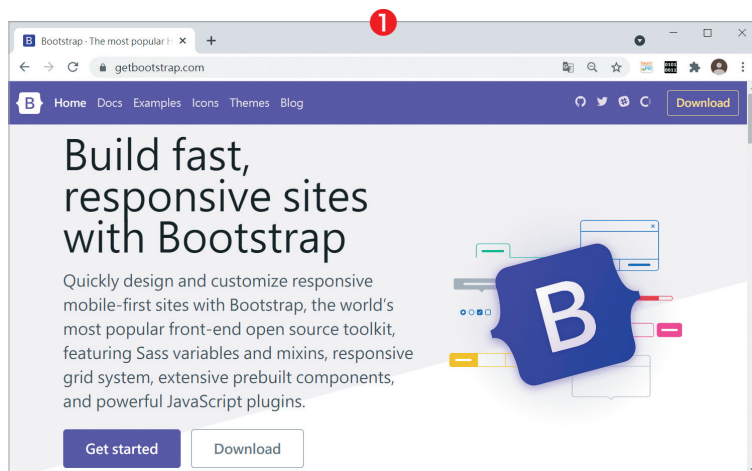
### ✓ 不同裝置的網站有各自的網址

以前面舉的 Yahoo! 奇摩為例，其 PC 網站的網址為 <https://tw.yahoo.com/>，而其行動網站的網址為 <https://tw.mobi.yahoo.com/>，多個網址可能不利於搜尋引擎為網站建立索引，影響自然排序名次；或者，當自動轉址程式無法正確判斷使用者的上網裝置時，可能會開啟不適合該裝置的網站。

## 1.2.2 響應式網頁設計

**響應式網頁設計** (RWD, Responsive Web Design) 指的是一種網頁設計方式，目的是根據使用者的瀏覽器環境（例如寬度或行動裝置的方向等），自動調整網頁的版面配置，以提供最佳的顯示結果，換句話說，只要設計單一版本的網頁，就能完整顯示在 PC、平板電腦、智慧型手機等裝置。

以 Bootstrap 網站 (<https://getbootstrap.com/>) 為例，它會隨著瀏覽器的寬度自動調整版面配置，當寬度夠大時，會顯示如下圖①，隨著寬度縮小，就會按比例縮小，如下圖②，最後變成單欄版面，如下圖③，這就是響應式網頁設計的基本精神，不僅網頁的內容只有一種，網頁的網址也只有一個。





## 響應式網頁設計的優點

### ✓ 網頁內容只有一種

由於響應式網頁是同一份 HTML 文件透過 CSS 的技巧，以根據瀏覽器的寬度自動調整版面配置，因此，一旦資料需要更新，只要更新同一份 HTML 文件即可，這樣就不用擔心費時費力和資料不同步的問題。

### ✓ 網址只有一個

由於響應式網頁的網址只有一個，所以不會影響網站被搜尋引擎找到的自然排序名次，也不會發生自動轉址程式誤判上網裝置的情況，達到 One Web One URL（單一網站單一網址）的目標。

### ✓ 技術門檻較低

響應式網頁只要透過 HTML 和 CSS 就能夠達成，不像自動轉址程式必須使用 JavaScript 或 PHP 來撰寫。

## 響應式網頁設計的缺點

### ✓ 舊版的瀏覽器不支援

響應式網頁需要使用 HTML5 的部分功能與 CSS3 的媒體查詢功能，舊版的瀏覽器可能不支援。

### ✓ 開發時間較長

由於響應式網頁要同時兼顧不同裝置，所以需要花費較多時間在不同裝置進行模擬操作與測試。

### ✓ 無法充分發揮裝置的特點

為了要適用於不同裝置，響應式網頁的功能必須有所妥協，例如一些在 PC 廣泛使用的動畫或功能可能無法在行動裝置執行，而必須放棄不用；無法針對行動裝置的觸控、螢幕可旋轉、照相功能等特點開發專屬的操作介面。

## 響應式網頁設計的主要技術

響應式網頁設計主要會使用到下面三種技術：

### ✓ 媒體查詢 (media query)

透過 CSS3 新增的媒體查詢功能以針對媒體類型量身訂做樣式表，例如根據瀏覽器的寬度自動調整版面配置。

### ✓ 流動圖片 (fluid image)

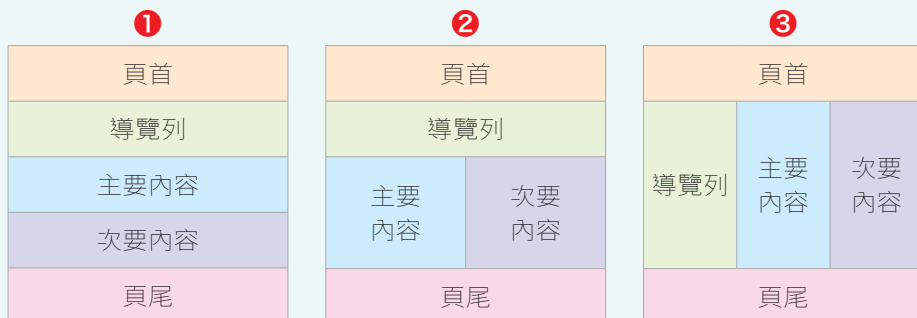
流動圖片指的是在設定圖片或物件等元素的大小時，根據其容器的大小比例做縮放，而不要設定絕對大小，如此一來，當螢幕的大小改變時，元素的大小也會自動按比例縮放，以同時適用於 PC 和行動裝置。

### ✓ 流動網格 (fluid grid)

流動網格包含**網格設計** (grid design) 與**液態版面** (liquid layout)，前者指的是利用固定的格子分割版面來設計布局，將內容排列整齊，而後者指的是根據瀏覽器的寬度自由縮放網頁上的元素。



隨著響應式網頁設計逐漸成為主流，許多網站開始導入「多欄式版面」，行動網頁通常採取如圖 ① 的單欄式，而 PC 網頁因為寬度較大，可以採取如圖 ② 的兩欄式或如圖 ③ 的三欄式。





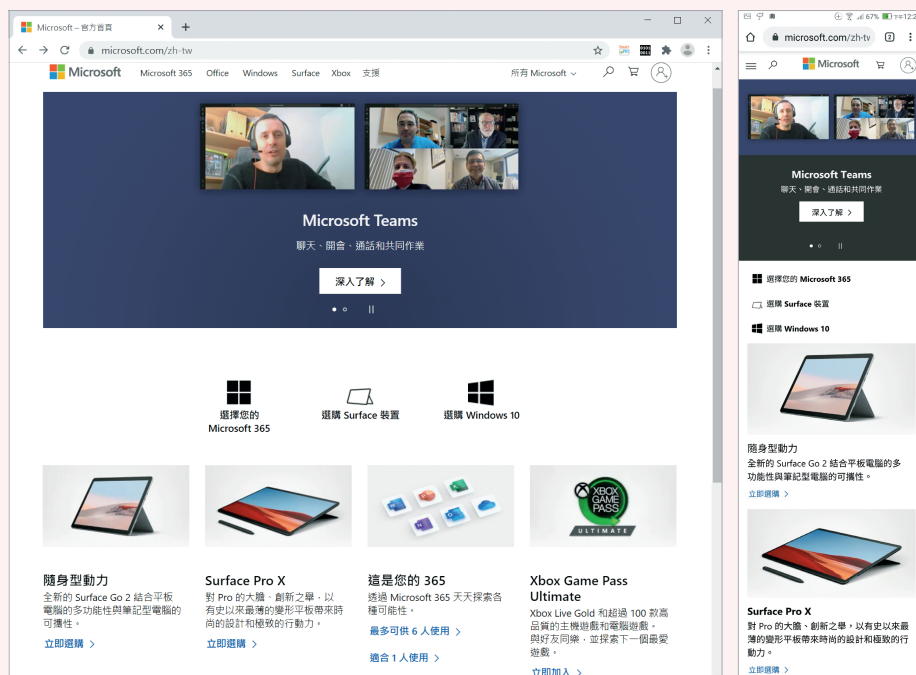


## 行動優先

**行動優先** (mobile first) 指的是在設計網站時應以優化行動裝置體驗為主要考量，其它裝置次之，但這並不是說要從行動網站開始設計，而是在設計網站的過程中優先考量網頁在行動裝置上的操作性與可讀性，不能將傳統的 PC 網頁直接移植到行動裝置，畢竟 PC 和行動裝置的特點不同。

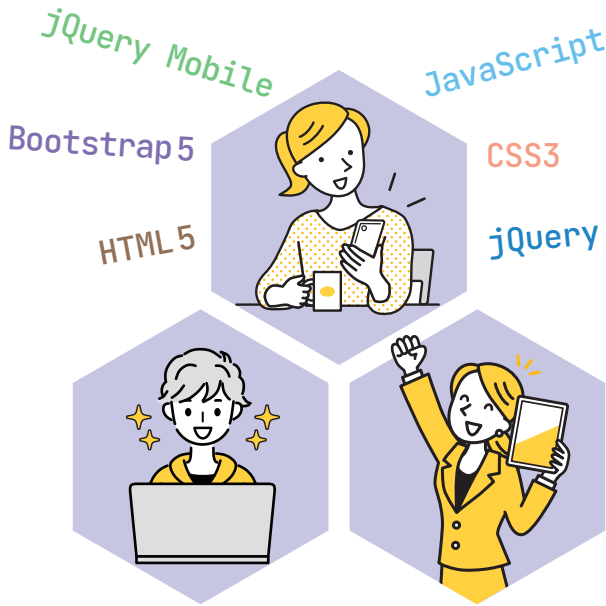
事實上，在開發響應式網頁時，優先考量如何設計行動網頁是比較有效率的做法，畢竟手機的限制比較多，先想好要在行動網頁放置哪些必要的內容，再來想 PC 網頁可以加上哪些選擇性的內容並逐步加強功能。

目前已經有不少網站導入行動優先的概念，以下圖的微軟網站 (<https://www.microsoft.com/zh-tw>) 為例，無論是 PC、平板電腦或手機的使用者都可以透過單一網址瀏覽網站，網頁會根據瀏覽器的寬度自動調整欄位的數目與順序。



# 04

## 圖片與表格



- 4.1 嵌入圖片 —  
<img> 元素
- 4.2 標註 —  
<figure>、<figcaption> 元素
- 4.3 建立表格 —  
<table>、<tr>、<td>、<th> 元素
- 4.4 表格標題 —  
<caption> 元素
- 4.5 表格的表頭、主體與表尾 —  
<thead>、<tbody>、<tfoot> 元素
- 4.6 直行式表格 —  
<colgroup>、<col> 元素

## 4.1 嵌入圖片 — <img> 元素

除了文字之外，HTML 文件還可以包含圖片、聲音、影片或其它 HTML 文件，本節的討論是以圖片為主。我們可以使用 **<img>** 元素在 HTML 文件中嵌入圖片，該元素沒有結束標籤，常見的屬性如下：

- ✓ **src="url"**：設定圖片的網址。
- ✓ **width="n"**：設定圖片的寬度 ( $n$  為像素數或容器寬度比例)。
- ✓ **height="n"**：設定圖片的高度 ( $n$  為像素數或容器高度比例)。
- ✓ **alt="..."**：設定圖片的替代顯示文字。
- ✓ **ismap**：設定圖片為伺服器端影像地圖。
- ✓ **usemap="url"**：設定所要使用的影像地圖。
- ✓ 第 2.1 節所介紹的全域屬性。



網頁上的圖檔格式通常是以 JPEG、GIF、PNG 為主，若圖片是由點與線的幾何圖形所組成，亦可考慮使用 SVG 向量格式，其優點是檔案較小，適合縮放。

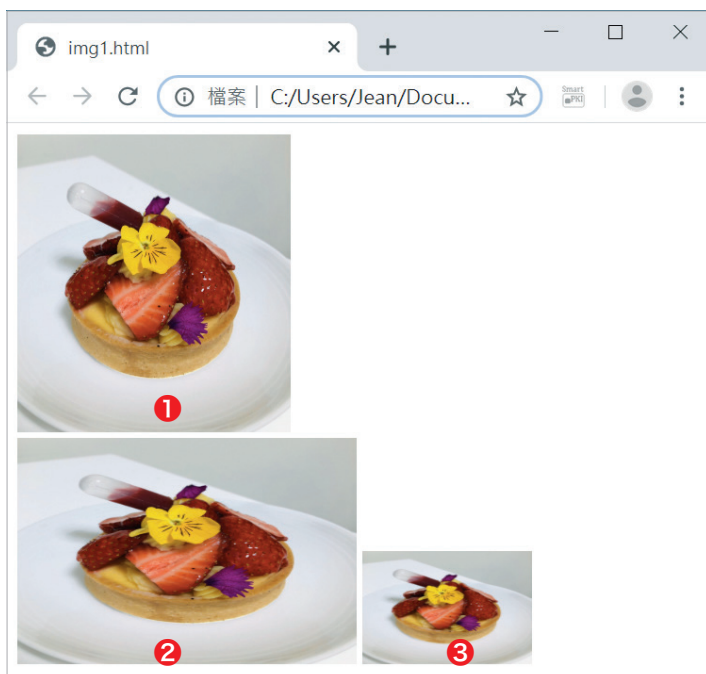
	JPEG	GIF	PNG
色彩數目	全彩	256 色	全彩
透明度	無	有	有
動畫	無	有	無 (可以透過擴充規格 APNG 製作動態效果)
適用時機	照片、漸層圖片	簡單圖片、需要去背或動態效果的圖片	照片、漸層圖片、簡單圖片、需要去背的圖片、動態貼圖

### 4.1.1 圖片的寬度與高度

當我們使用 `<img>` 元素在 HTML 文件中嵌入圖片時，除了可以透過 `src="url"` 屬性設定圖片的網址，還可以透過 `width="n"` 和 `height="n"` 屬性設定圖片的寬度與高度， $n$  為像素數或容器寬度與高度比例。若沒有設定寬度與高度，瀏覽器會以圖片的原始大小來顯示，下面是一個例子。

\Ch04\img1.html

```
<body>  
  <br>  
    
    
</body>
```



- ❶ 圖片的寬度為網頁寬度的 40%（高度則依照同比例縮放）
- ❷ 圖片的寬度為 240 像素、高度為 160 像素（圖片有點變形）
- ❸ 圖片的寬度為 120 像素、高度為 80 像素

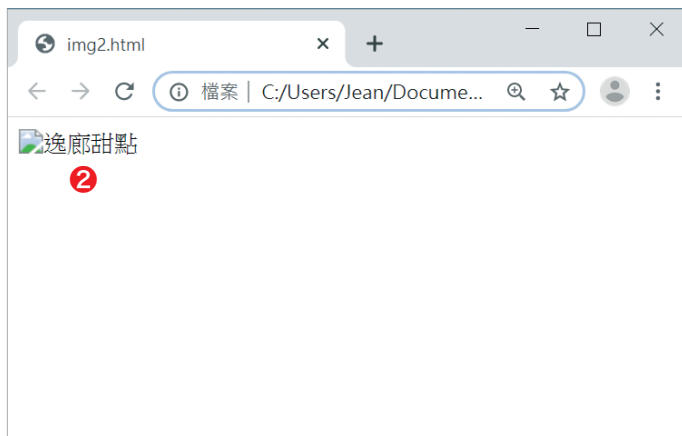
### 4.1.2 圖片的替代顯示文字

為了避免圖片因為取消下載、連線錯誤或找不到檔案等情況而無法顯示，我們通常會使用 `<img>` 元素的 `alt="..."` 屬性設定替代顯示文字來描述圖片，而且此舉將有助於搜尋引擎優化 (SEO)，提高圖片及網頁被搜尋引擎找到的機率。

下面是一個例子，由於找不到 `<img>` 元素所設定的 `cake.jpg` 圖片，所以會在圖片的位置顯示替代顯示文字。

\Ch04\img2.html

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



- ① 設定圖片的替代顯示文字
- ② 當無法順利顯示圖片時，會顯示替代顯示文字

## 4.2 標註 — <figure>、<figcaption> 元素

我們可以使用 HTML5 新增的 <figure> 元素將圖片、表格、程式碼等能夠從主要內容抽離的區塊標註出來，同時可以使用 <figcaption> 元素針對 <figure> 元素的內容設定說明，這兩個元素的屬性有第 2.1 節所介紹的全域屬性。

<figure> 元素所標註的區塊不會影響主要內容的閱讀動線，而且可以移到附錄、網頁的一側或其它專屬的網頁。

下面是一個例子，它會使用 <figure> 元素標註兩張照片，並使用 <figcaption> 元素設定照片的說明。

\Ch04\figure.html

```
<body>
  <figure>
    
    
    <figcaption> 逸廊甜點 </figcaption>
  </figure>
</body>
```



## 4.3 建立表格 — <table>、<tr>、<td>、<th> 元素

當我們要陳列資料時，表格是很常見的形式，能夠讓讀者一目了然。在本節中，我們會說明如何使用 <table>、<tr>、<td>、<th> 等元素建立表格，以及示範一些常用的技巧。

### <table> 元素

<table> 元素用來標示表格，其屬性如下：

- ✓ **border**="n"：設定表格的框線大小 (n 為像素數)。
- ✓ 第 2.1 節所介紹的全域屬性。

### <tr> 元素

<tr> 元素用來在表格中標示一行 (row)，其屬性如下：

- ✓ 第 2.1 節所介紹的全域屬性。

### <th> 元素

<th> 元素用來在一列中標示一個標題儲存格，其屬性如下：

- ✓ **colspan**="n"：設定標題儲存格是由幾行合併而成 (n 為行數)。
- ✓ **rowspan**="n"：設定標題儲存格是由幾列合併而成 (n 為列數)。
- ✓ **headers**="..."：設定標題儲存格的標題。
- ✓ **abbr**="..."：根據儲存格的內容設定一個縮寫。
- ✓ **scope**="{row,col,rowgroup,colgroup,auto}"：設定標題儲存格是一列、一行、一組列或一組行的標題，省略不寫的話，表示預設值 auto (自動)。
- ✓ 第 2.1 節所介紹的全域屬性。



## <td> 元素

<td> 元素用來在一列中標示一個儲存格，其屬性如下：

- ✓ **colspan**="n"：設定儲存格是由幾行合併而成 (n 為行數)。
- ✓ **rowspan**="n"：設定儲存格是由幾列合併而成 (n 為列數)。
- ✓ **headers**="..."：設定與儲存格關聯的標題儲存格。
- ✓ 第 2.1 節所介紹的全域屬性。

下面是一個例子，它要製作如下圖的 4×3 表格 (4 列 3 行)，操作步驟如下：

人物素描	角色	介紹
	喬巴	身份船醫，夢想成為能治百病的神醫。
	索隆	主角魯夫的夥伴，夢想成為世界第一的劍士。
	佛朗基	傳說中的船匠－湯姆的弟子，打造了千陽號。

- 1 首先要標示表格，請在 HTML 文件的 <body> 元素裡面加入 <table> 元素，同時將表格框線設定為 1 像素，若沒有設定表格框線，就不會顯示框線。

```

<body>
  <table border="1">
    </table>
</body>

```

- ② 接著要標示表格的列數，請在 `<table>` 元素裡面加入 4 個 `<tr>` 元素。

```
<table border="1">  
  <tr></tr>  
  <tr></tr>  
  <tr></tr>  
  <tr></tr>  
</table>
```

- ③ 繼續要在表格的每一列中標示各個儲存格，由於表格有 3 行，而且第一列為標題列，所以在第一個 `<tr>` 元素裡面加入 3 個 `<th>` 元素，其餘各列則分別加入 3 個 `<td>` 元素，表示每一列有 3 行。

```
<table border="1">  
  <tr>  
    <th></th>  
    <th></th>  
    <th></th>  
  </tr>  
  <tr>  
    <td></td>  
    <td></td>  
    <td></td>  
  </tr>  
  <tr>  
    <td></td>  
    <td></td>  
    <td></td>  
  </tr>  
  <tr>  
    <td></td>  
    <td></td>  
    <td></td>  
  </tr>  
</table>
```

- 4 最後在每個 <th> 和 <td> 元素裡面輸入各個儲存格的內容，就大功告成了。您可以在儲存格內嵌入圖片或輸入文字，同時可以設定圖片或文字的格式，有需要的話，還可以設定超連結。

\Ch04\piece.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title> 航海王 </title>
  </head>
  <body>
    <table border="1">
      <tr>
        <th> 人物素描 </th>
        <th> 角色 </th>
        <th> 介紹 </th>
      </tr>
      <tr>
        <td></td>
        <td> 喬巴 </td>
        <td> 身份船醫，夢想成為能治百病的神醫。</td>
      </tr>
      <tr>
        <td></td>
        <td> 索隆 </td>
        <td> 主角魯夫的夥伴，夢想成為世界第一的劍士。</td>
      </tr>
      <tr>
        <td></td>
        <td> 佛朗基 </td>
        <td> 傳說中的船匠－湯姆的弟子，打造了千陽號。</td>
      </tr>
    </table>
  </body>
</html>
```

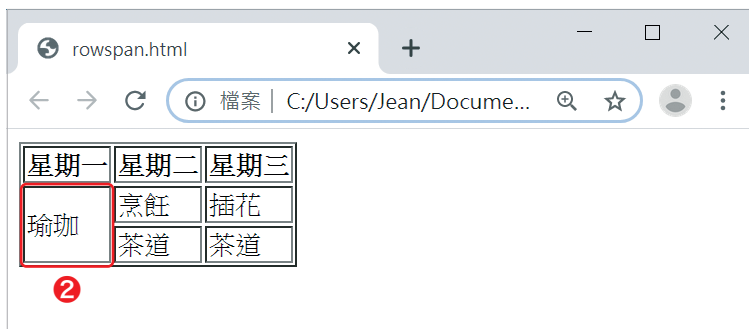
### 4.3.1 跨列合併儲存格

有時我們需要將某幾列的儲存格合併成一個儲存格，達到跨列的效果，此時可以使用 `<td>` 或 `<th>` 元素的 `rowspan="n"` 屬性，其中  $n$  為要合併的列數。

下面是一個例子，它在第二列的第一個儲存格加上 `rowspan="2"` 屬性，表示該儲存格是由兩個儲存格跨列合併而成，於是得到如下圖的瀏覽結果。

\Ch04\rowspan.html

```
<table border="1">
  <tr>
    <th> 星期一 </th>
    <th> 星期二 </th>
    <th> 星期三 </th>
  </tr>
  <tr>
    <td rowspan="2"> 瑜珈 </td>
    <td> 烹飪 </td>
    <td> 插花 </td>
  </tr>
  <tr>
    <td> 茶道 </td>
    <td> 茶道 </td>
  </tr>
</table>
```



❶ 在此儲存格加上 `rowspan="2"` 屬性    ❷ 跨列合併儲存格的結果

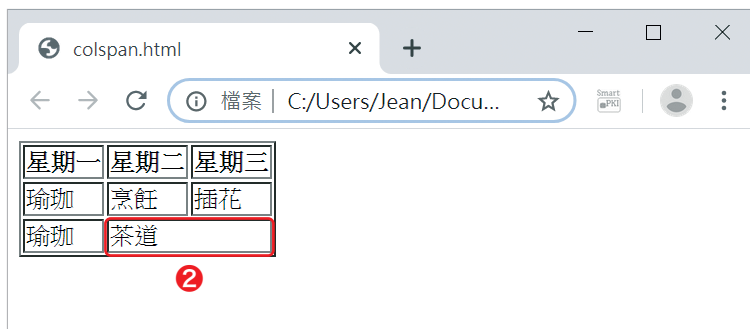
### 4.3.2 跨行合併儲存格

有時我們需要將某幾行的儲存格合併成一個儲存格，達到跨行的效果，此時可以使用 `<td>` 或 `<th>` 元素的 `colspan="n"` 屬性，其中  $n$  為要合併的行數。

下面是一個例子，它在第三列的第二個儲存格加上 `colspan="2"` 屬性，表示該儲存格是由兩個儲存格跨行合併而成，於是得到如下圖的瀏覽結果。

\Ch04\colspan.html

```
<table border="1">
  <tr>
    <th> 星期一 </th>
    <th> 星期二 </th>
    <th> 星期三 </th>
  </tr>
  <tr>
    <td> 瑜珈 </td>
    <td> 烹飪 </td>
    <td> 插花 </td>
  </tr>
  <tr>
    <td> 瑜珈 </td>
    <td colspan="2"> 茶道 </td>
  </tr>
</table>
```



❶ 在此儲存格加上 `colspan="2"` 屬性    ❷ 跨行合併儲存格的結果