

本書會先帶領讀者學習 PHP 這項程式語言，然後教讀者如何利用 PHP 建立網站，以及如何將網站使用的資料儲存在 MySQL 這類的資料庫裡。

PHP 的設計宗旨是發展一個可以在網頁伺服器上執行的程式語言，目的是當有人請求某個網頁時，伺服器會自動根據請求的內容產生 HTML 頁面，再將產生出來的頁面發送回去給指定的使用者，也就是說伺服器在產生網頁時，可以針對個人請求量身制定出不同的內容。這項需求讓所有使用 PHP 語言的網站允許使用者執行以下這些操作：

- **註冊或登入網站**：因為每位使用者的名稱、電子郵件和個人密碼都不一樣。
- **線上購物**：因為每位顧客的訂單、付款和運送細節都是獨一無二的內容。
- **網站搜尋**：對每位使用者來說，每次的搜尋結果都是針對他們的搜尋條件而量身制定的內容。

PHP 的設計考量之一是搭配像 MySQL 這類的資料庫一起運用，目的是儲存資料，例如，網站顯示的網頁內容、網站販售的產品或是網站會員的詳細資訊。本書將帶讀者學習怎麼使用 PHP 語言建立網頁，讓網站會員透過網頁更新資料庫裡儲存的資料。例如：

- **內容管理系統**：允許網站擁有者利用表單更新網站內容，並且將更新內容顯示給網站訪客，但不需要額外寫新的程式碼。
- **網路商店**：允許商店擁有者針對促銷目的和顧客，列出可以購買的產品。
- **社群網路**：允許訪客註冊帳號和登入網站、建立使用者個人資料、上傳使用者自己的內容，以及讓使用者看到針對個人興趣等等需求條件而量身制定的網頁內容。

由於這些網站顯示的資訊是儲存在資料庫裡，通常會說這些是具有**動態資料庫設計**的網站。

本書透過不同的內頁設計傳達幾種不同類型的資訊，此處先帶讀者看看本書會出現的內頁類型。

### 變數命名

### 純量（基本）資料型態

變數名稱應符合以下條件，而且需要利用以下規則來產生變數名稱：

- 1 以美元符號「\$」開頭。
- 2 由數字、英文字母以及底線「\_」組成。變數名稱中，英文字母與數字間，不可有空格。
- 3 變數名稱中不可出現特殊字元，且變數名稱中不可出現「\_」字元。

PHP 的變數命名規則與 C 語言類似，但 PHP 的變數命名規則更為寬鬆。變數名稱可以包含數字、下劃線、美元符號、以及變數名稱中不可出現特殊字元。變數名稱中不可出現「\_」字元。

變數命名規則：

- 變數名稱必須以美元符號「\$」開頭。
- 變數名稱可以由數字、英文字母、下劃線、美元符號、以及變數名稱中不可出現特殊字元組成。
- 變數名稱中不可出現「\_」字元。

變數命名規則：

- 變數名稱必須以美元符號「\$」開頭。
- 變數名稱可以由數字、英文字母、下劃線、美元符號、以及變數名稱中不可出現特殊字元組成。
- 變數名稱中不可出現「\_」字元。

## 主題資訊

這種以白底設計呈現的內頁是用於介紹各項主題，解釋這些主題的來龍去脈及其使用方法。

### 範例：使用比較運算子

### 範例：使用邏輯運算子

### PHP 發展史

### MYSQL 發展史

## 範例程式碼

這種以米褐色底設計呈現的內頁是用於說明如何套用本書展示的各段程式碼。

### 一個陽春的 PHP 網頁

範例程式碼：

```

<?php
// 設定網頁標題
<title>一個陽春的 PHP 網頁</title>
// 設定網頁內容
<h1>一個陽春的 PHP 網頁</h1>
</pre>

```

## 實作範例

這種內頁會出現在本書前幾章裡，目的是將讀者已經學過的主題結合在一起，展示如何應用這些主題。

## 資訊圖表

這種以暗黑色底設計呈現的內頁是以圖表提供資訊，用來解釋各項觀念。

### 本章重點回顧

#### PHP 基本語法：變數、表達式與運算子

- 變數用於儲存每次執行腳本時會改變的資料。
- 純量資料型態包含文字、非負整數、浮點數和布林值（true 或 false）。
- 陣列屬於變量資料型態，用於儲存一些相關的內容。
- 陣列中的每個項目稱為元素。
- 變數名稱與陣列的索引具有變態和資料值。
- 索引式陣列的索引具有可變態和資料值。
- 字串運算子能結合（串接）多個字串的文字。
- 算術運算子使用數字執行數學運算。
- 比較運算子用於比較兩個值，判斷其中一個值是否相等、大於或小於另一個值。
- 邏輯運算子利用 &&「且」和 |「或」，組合多個表達式的計算結果。

## 重點回顧

這種內頁會出現在每一章的結尾，用意是帶讀者回顧每一章涵蓋的重要主題。

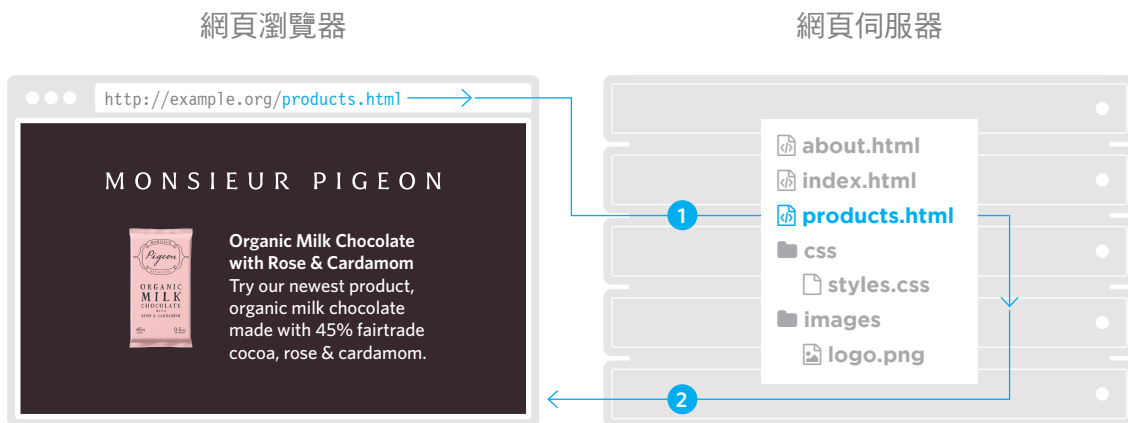
# 靜態網站 vs. 動態網站

只用 HTML 和 CSS 檔案建立而成的網站，每位訪客看到的網站內容都一樣，因為他們全都收到相同的 HTML 和 CSS 檔案。

1. 當使用者從瀏覽器請求網站上的某個網頁，如果這個網站是以 HTML 和 CSS 檔案建構而成，則這項請求會傳送給架設這個網站的**網頁伺服器**。
2. 網頁伺服器收到請求後，會尋找瀏覽器端要求的 HTML 檔案，然後將找到的檔案傳回給瀏覽器。可能還會傳送其他類型的檔案，例如，用來設定網頁風格的 CSS 檔案、媒體（例如，圖像）、JavaScript 和其他網頁會用到的檔案。

由於所有網站訪客都收到相同的 HTML 檔案，他們當然都會看到相同的網頁內容，因此，這種類型的網站就稱為**靜態網站**。

負責更新靜態網站內容的人必須具備 HTML 和 CSS 方面的知識。如果網站內容的擁有者想更新網頁上的文字，必須先手動更新 HTML 程式碼，再將 HTML 檔案上傳到網頁伺服器。



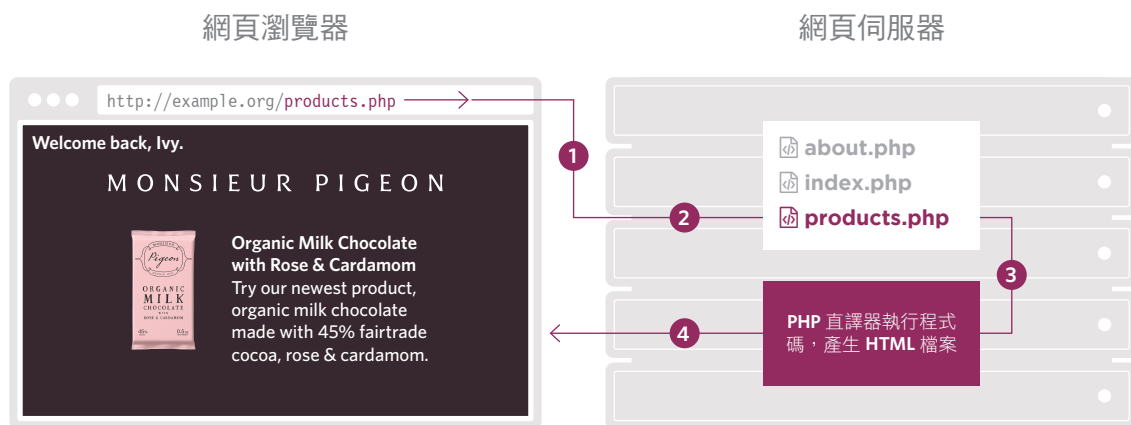
本書預設讀者已具有 HTML 和 CSS 方面的相關知識，若讀者尚未接觸或是不熟悉這方面的知識，建議先閱讀本書姊妹作《HTML&CSS：網站設計建置優化之道》（<http://htmlandcssbook.com>）。

使用 PHP 建置網站時，每位網站訪客都能看到不同的網頁內容，因為 PHP 網頁的做法是製作 HTML 檔案，再將產生出來的檔案發送給網站訪客。

每當使用者訪問 eBay、Facebook、新聞等等網站時，這些網站通常會顯示新的資訊。如果使用者檢視瀏覽器內的網頁原始碼，會看到 HTML 程式碼，但程式設計師並不會在使用者訪問網站期間手動更新程式碼。

這種類型的網站稱為**動態網站**，因為發送給網站訪客的 HTML 網頁是以像 PHP 這種程式語言所寫的指令製作而成。

1. 當使用者從瀏覽器請求網站上的某個網頁，如果這個網站是以 PHP 建構而成，則這項請求會傳送給網頁伺服器。
2. 網頁伺服器收到請求後會先尋找 PHP 檔案。
3. PHP 檔案裡所有的程式碼都會透過一套稱為 PHP 直譯器的軟體執行，為該網站訪客製作 HTML 網頁。
4. 網頁伺服器將這個為網站訪客創造出來的 HTML 網頁發送到訪客使用的瀏覽器。網頁伺服器不會保留這個檔案的副本，因此，當該訪客下次要求 PHP 檔案時，伺服器又會為訪客重新製作一個 HTML 網頁。



網頁伺服器不會將 PHP 程式碼發送給網頁瀏覽器，這個程式碼只會用於製作 HTML 網頁，並且發送給瀏覽器。由於 PHP 程式碼只會在網頁伺服器上執行，所以被視為伺服器端的程式設計。

PHP 可針對每位網站訪客量身制定，製作出個人化的 HTML 網頁，內容可能包含顯示訪客名稱、訪客感興趣的主題或是來自訪客好友的貼文。

# PHP：程式語言和直譯器

PHP 直譯器是一套在網頁伺服器上執行的軟體，使用者利用 PHP 語言撰寫程式碼，告訴直譯器要做什麼。

軟體是幫助人類利用電腦完成特定的工作任務，但無須深入了解電腦究竟是如何達成任務。例如：

- 你不需要了解電腦如何儲存或傳輸電子郵件，只要利用電子郵件程式就能收發電子郵件。
- 你不必得知電腦究竟怎麼處理圖像，只要利用 Photoshop 這套軟體就能編輯圖片。

使用者每次使用一套軟體時，軟體具備的能力是完成相同的工作，但使用者在執行這些工作時可以帶入不同的資料：

- 電子郵件程式能製作、收發和儲存電子郵件，但每封郵件的內容和收件人可以不同。
- Photoshop 這套軟體可以完成的工作任務有：為圖像加上濾鏡、調整或裁剪大小，而且可以對任意圖像完成相同的工作。

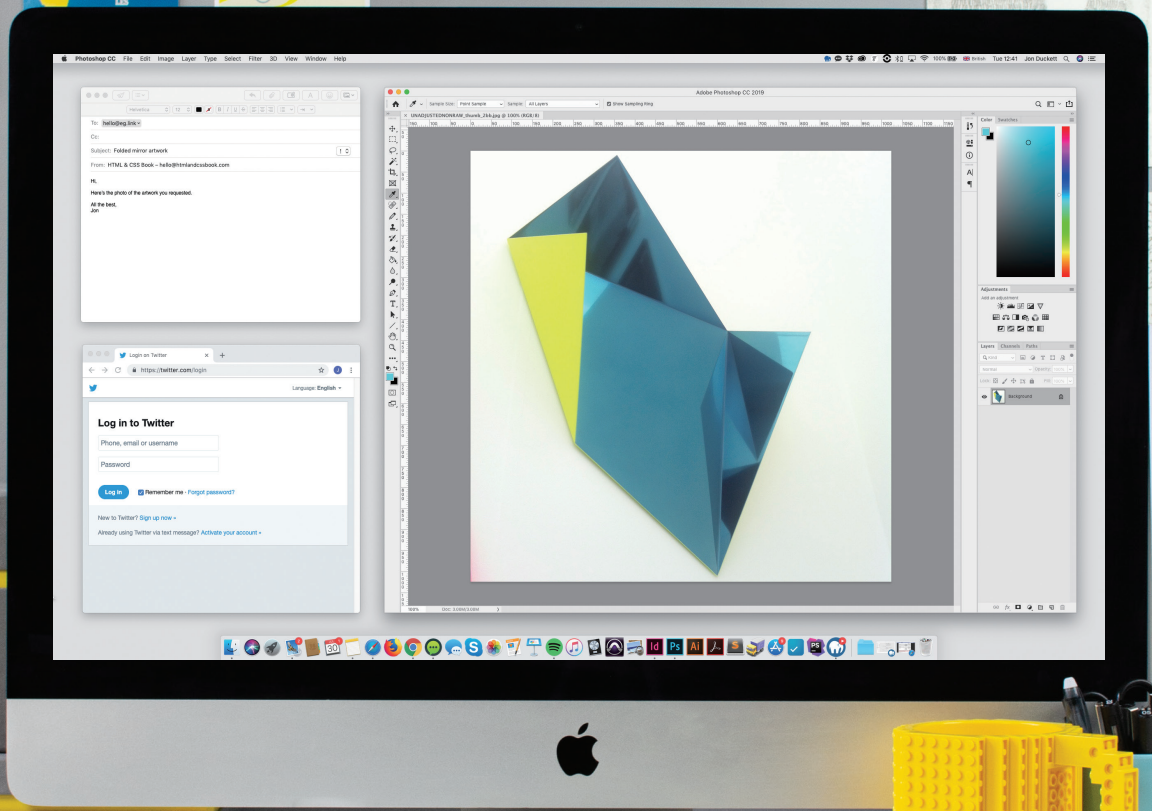
這些軟體都具有圖形化使用者介面，使用者透過介面互動就能完成這些工作。

PHP 直譯器也是一套軟體，這套軟體在網頁伺服器上執行，是伺服器組成的一部分。然而，不同於其他軟體，這套軟體沒有圖形化使用者介面可以使用，當使用者希望 PHP 直譯器做某些事情時，只能利用一項稱為 PHP 的程式語言撰寫程式碼。

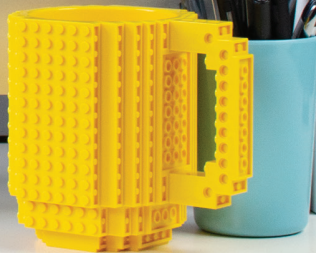
以 PHP 製作網頁時，雖然這個網頁永遠都會執行相同的工作任務，但每次要求網頁時，可以帶入不同的資料來執行這些任務。例如，以 PHP 語言撰寫而成的網站可以完成以下這些工作：

- 即使每位會員用來登入網站的電子郵件帳號和密碼不同，但只要一個登入頁面，就能讓所有網站會員同時登入。
- 即使有數百位不同的使用者同時利用使用者帳號這個頁面，但也只需要一個頁面就能讓所有會員都看到帳號的詳細資訊，而且是只能看到自己帳號底下的資訊。

會出現這樣的可能性是因為對每位使用者來說，雖然執行這些工作需要的規則或指令都一樣，但每位使用者可以提供或看到不同的資料。



Olafur Eliasson — Baroque



# 以不同的資料執行相同的工作

程式語言讓我們創造規則，告訴電腦如何執行一項工作任務，但程式每次執行工作任務時可以帶入不同的資料。

使用任何程式語言都必須對電腦下精準的指令，確實說明你希望電腦做什麼事，這種類型的指令跟你要求人類執行工作時所下的指令大不相同。

請想像一下，現在你想買五根棒棒糖，所以你需要計算購買棒棒糖的總金額，計算總金額的方式是將一根棒棒糖的單價乘上要購買的根數。

以下列這個式子表達前述規則：

$$\text{總金額} = \text{商品單價} \times \text{商品數量}$$

計算棒棒糖的總金額：

- 若一根棒棒糖的單價為 1 美金，則買五根棒棒糖花費的總金額是 5 美金。
- 若一根棒棒糖的單價變成 1.5 美金，計算規則不變，則買五根棒棒糖花費的總金額會是 7.5 美金。
- 若你想買十根單價為 2 美金的棒棒糖，計算規則不變，則花費的總金額會是 20 美金。

計算式裡的**總金額**、**商品單價**和**商品數量**可以用不同的值取代，但用於計算棒棒糖花費總金額的規則會一樣。

以 PHP 製作網頁時，首先必須釐清：

- 你要完成的工作任務
- 每次執行工作任務時會改變的資料

然後提供詳細的指令給 PHP 直譯器，說明完成工作任務的方式，以名稱表示會改變的值。假設你提供以下的值給 PHP 直譯器：

$$\text{商品單價} = 3$$

$$\text{商品數量} = 5$$

使用以下計算規則：

$$\text{總金額} = \text{商品單價} \times \text{商品數量}$$

**總金額**的值會是 15。下次執行網頁時，如果提供不同的值給商品單價或商品數量，網頁會使用相同規則計算出新的總金額值。

因為每次執行程式時，這些值可以改變（或發生變化），程式設計師就將表示這些值的單字稱為**變數**。

總金額 = 商品單價 × 商品數量



$$\$9 = 3 \times 3$$



# PHP 網頁簡介

PHP 網頁通常會由 HTML 和 PHP 程式碼組成，用於發送 HTML 網頁給瀏覽器。

左下方這個 PHP 網頁是由 HTML 和 PHP 程式碼組成。

- 藍色文字是 HTML 程式碼。
- 紫色文字是 PHP 程式碼。

PHP 直譯器開啟這個網頁檔案後：

- 為網站訪客產生 HTML 暫存檔，然後將 HTML 程式碼的部分全部直接複製到這個檔案裡。
- 依照 PHP 程式碼裡寫的指令執行工作（通常是產生 HTML 網頁的內容）。

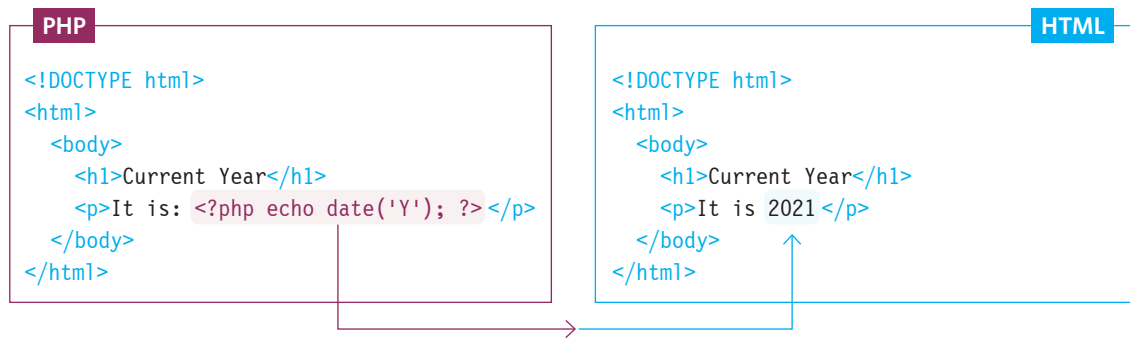
下方的 PHP 程式碼內容是確認目前的年分，然後將年分資料寫在起始標籤 `<p>` 和結束標籤 `</p>` 之間。

PHP 程式碼不僅能執行基本的工作任務，像是算數或取得今天的日期；也能完成更複雜的工作，例如，使用 HTML 表單發送的資訊來更新資料庫裡現存的資料。

PHP 直譯器處理完 PHP 檔案後，會將為網站訪客產生的 HTML 暫存檔發送給訪客使用的瀏覽器，然後刪除這個暫存檔。

PHP 直譯器依照 PHP 程式碼執行工作後，會發送右下方的 HTML 網頁給瀏覽器。

PHP 直譯器確認目前的年分後，將年分資料顯示在直譯器產生出來的 HTML 網頁裡。



PHP 直譯器取得目前的年分，然後將年分資料寫入兩個段落標籤之間。

所有網頁每次收到請求時，通常會執行相同的工作，但可以針對每位網站訪客處理不同的資訊。

PHP 網站是由一組 PHP 網頁組成，每個網頁負責執行特定的工作。例如，網站若要允許會員登入，必須具有：

- 登入頁面：讓會員登入網站。
- 個人資料頁面：顯示會員的個人資料。

每當其中一個頁面收到請求時，都必須針對當前發送請求的會員處理不同的資料，所以這些頁面必須：

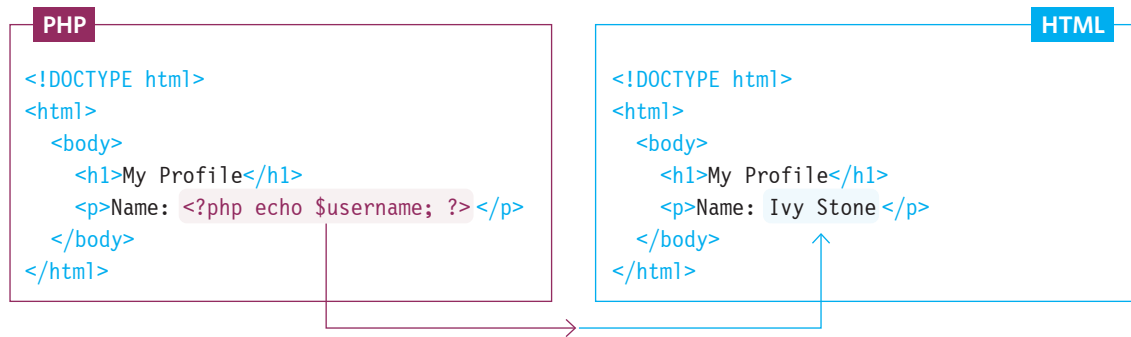
- 含有指令，用以指示該頁面本身應該如何完成工作。
- 為每次請求該頁面時會改變的每個資料命名。

PHP 以變數名稱表示每次請求該頁面時可以改變的值，PHP 程式碼負責告訴 PHP 直譯器：

- 每次請求該頁面時可以改變的特定資料要用什麼變數名稱。
- 本次請求該頁面時要使用什麼值。

PHP 直譯器將產生出來的 HTML 網頁發送給使用者後，就會忘記所有原本儲存在變數裡的值，這樣才能為下一位請求該網頁的使用者，以不同的值執行相同的工作。

如果資料需要保留更久的時間，就要放在第 12 頁會介紹到的 MySQL 資料庫裡。



PHP 直譯器取得變數 `$username` 的儲存值，然後將這個值寫在兩個段落標籤之間。

# MYSQL 簡介

MySQL 是一種資料庫，資料庫以結構化方式儲存資料，方便我們能輕鬆存取和更新資料庫保存的資訊。

Excel 等電子試算表程式是將資訊保存在由欄和列組成的儲存格裡，再將電子試算表儲存的資料用於執行計算或以公式處理資料。

MySQL 是一套軟體，以類似電子試算表的方式，將資訊儲存在也是由欄和列組成的**資料表**裡，再利用 PHP 程式存取和更新資料庫儲存的資訊。

一個資料庫能擁有多個資料表，每個資料表通常保存網站需要儲存的一種資料。以下是範例資料庫中的兩個資料表：

- 網站會員（或使用者）。
- 網站上顯示的文章。

每個資料表中的**欄位名稱**是用來說明每個欄位保存的資料類型：

- 資料表「member」：儲存每位會員的名字、姓氏、電子郵件帳號、密碼、加入會員的日期以及個人頭像。
- 資料表「article」：儲存每篇文章的標題、摘要、內容、建立日期及其他下一頁會討論到的資料值。

每一**列**保存的資料是用來描述構成資料表的其中一個項目：

- 資料表「member」：每一列代表一位會員的資料。
- 資料表「article」：每一列代表一篇文章的資料。

資料表名稱		欄位名稱				欄		
<b>member</b>								
id	forename	surname	email	password	joined	picture		
1	Ivy	Stone	ivy@eg.link	\$2y\$10\$MAdTTCA0Mi0w	2021-01-01 20:28:47	ivy.jpg		
2	Luke	Wood	luke@eg.link	\$2y\$10\$NN5HEAD3atar	2021-01-02 09:17:21	NULL		
3	Emiko	Ito	emi@eg.link	\$2y\$10\$/RpRmiUMStji	2021-01-02 10:42:36	emi.jpg		
<b>article</b>								
id	title	summary	content	created	category_id	member_id	image_id	published
1	Systemic	Brochure	<p>This	2021-01-01	1	2	1	1
2	Polite	Poster	<p>These	2021-01-02	1	1	2	1
3	Swimming	Architect	<p>This	2021-01-02	4	1	3	1

PHP 可以用來：

- **取得資料庫裡的資料**，將獲得的資訊顯示在網頁上。
- **新增資料列**：建立新文章時，需要在資料表「article」裡新增一列資料，並且提供每個資料欄位應該儲存的內容。
- **刪除資料列**：刪除一篇文章時，需要移除代表該篇文章的一整列資料。
- **改變現有資料列的內容**：更新會員電子郵件帳號時，需要從資料表「member」中找出想要更新的那一列資料，然後修改欄位「email」的值。

請注意：這兩個資料表的起始欄位都是 id，資料表中的每一列資料都具有唯一的 id 欄位值（這裡的欄位值是從 1 開始，每一列遞增 1）。利用欄位「id」，你可以告訴資料庫你想處理哪一列資料，例如，你要取得 id 為 2 的會員資料或是 id 為 1 的文章資料。

MySQL 稱為**關聯式資料庫**，因為這種資料庫能針對不同資料表裡儲存的資料類型，解釋它們之間的關係。

例如，在下方的資料表裡，這幾篇文章是由不同的網站會員寫的。資料表「article」裡的「member\_id」欄位值表示這篇文章是由哪個使用者撰寫，這個欄位裡保存的數字可以跟資料表「member」裡的「id」欄位值配對。

第一篇文章是由欄位「id」值為 2 的會員寫的（也就是 Luke Wood），第二篇和第三篇文章是由欄位「id」值為 1 的使用者寫的（也就是 Ivy Stone）。

這些關係是為了：

- **資料結構化**：確保每個資料表只會保存一種特定類型的資料（會員或文章）。
- **節省資料庫空間**：避免資料庫需要在多個資料表裡儲存相同的資料。
- **易於保存最新的資料**，若有會員改變名字，則只需要更新資料表「member」，不需要更新他們寫的每篇文章。

member							
id	forename	surname	email	password	joined	picture	
1	Ivy	Stone	ivy@eg.link	\$2y\$10\$MAdTTCAOMi0w	2021-01-01 20:28:47	ivy.jpg	
2	Luke	Wood	luke@eg.link	\$2y\$10\$NN5HEAD3atar	2021-01-02 09:17:21	NULL	
3	Emiko	Ito	emi@eg.link	\$2y\$10\$/RpRmiUMStji	2021-01-02 10:42:36	emi.jpg	

article								
id	title	summary	content	created	category_id	member_id	image_id	published
1	Systemic Brochure	<p>This	2021-01-01	1	2	1	1	1
2	Polite Poster	<p>These	2021-01-02	1	1	2	1	1
3	Swimming Architect	<p>This	2021-01-02	4	1	3	1	1

# PHP 發展史

PHP、MySQL 和多數軟體一樣，從問世以來不斷推陳出新，至今已經發展出多個版本。每推出一個新版本就會增加新的特性，執行速度也會比舊版本來得更快。

程式設計師 Rasmus Lerdorf 於 1994 年創造出 PHP 這項程式語言，並且於 1995 年公開釋出 PHP 的原始程式碼，鼓勵使用者共同改善這項程式語言。當時，PHP 這三個字母縮寫是表示「個人首頁」(Personal Home Page)，現在則是「PHP：超文字處理器」(PHP: Hypertext Processor)。

現今有 80% 的網站伺服器是使用 PHP 這項程式語言。

Facebook、Etsy、Flickr 和 Wikipedia 等網站最初都是以 PHP 語言開發而成，雖然其中某些網站近年來也使用其他技術。

一些熱門的開放原始碼軟體也都是以 PHP 語言撰寫而成，例如，WordPress、Drupal、Joomla 和 Magento，其中 WordPress 甚至支援全球 35% 以上的網站建置工作，學習 PHP 語言有助於你使用這些軟體。

PHP 語言每發布一個新版本，就會新增幾個特性，本書介紹的特性會包含到 2020 年 11 月釋出的 PHP 8。

.....	PHP 1	1995
.....		1996
.....		1997
.....	PHP 2	1998
.....	PHP 3	1999
.....		2000
.....	PHP 4	2001
.....		2002
.....		2003
.....		2004
.....	PHP 5	2005
.....		2006
.....	PHP 5.1	2006
.....	PHP 5.2	2007
.....		2008
.....		2009
.....	PHP 5.3	2010
.....		2011
.....	PHP 5.4	2012
.....		2013
.....	PHP 5.5	2014
.....		2015
.....	PHP 5.6	2015
.....		2016
.....	PHP 7	2016
.....		2017
.....	PHP 7.1	2017
.....	PHP 7.2	2018
.....		2019
.....	PHP 7.3	2019
.....	PHP 7.4	2020
.....		2021
.....	PHP 8	2021

# MYSQL 發展史

1995	MYSQL 1
1996	
1997	MYSQL 3.2
1998	
1999	PHPMYADMIN
2000	
2001	
2002	
2003	MYSQL 4
2004	
2005	
2006	MYSQL 5
2007	
2008	SUN BUYS MYSQL
2009	MYSQL 5.1
2010	MARIADB ORACLE BUYS SUN
2011	MYSQL 5.5
2012	
2013	MYSQL 5.6
2014	
2015	
2016	MYSQL 5.7
2017	
2018	MYSQL 8
2019	
2020	
2021	

MySQL 問世於 1995 年，SQL（英文發音為 *ess-queue-el* 或 *sequel*）這三個字母縮寫是表示「結構化查詢語言」（Structured Query Language），SQL 語言是用於存取關聯式資料庫裡的資訊。

MySQL 原本是瑞典一家叫 MySQL AB 的公司開發出來、讓大家免費使用的資料庫，其名稱由來是創始人之一的程式設計師 Michael Widenius 以自己女兒的名字「My」命名。

昇陽電腦（Sun Microsystems）於 2008 年 1 月收購 MySQL AB 這家公司，隨後甲骨文（Oracle）於 2010 年收購了昇陽電腦。

當 MySQL 的這些創始人得知甲骨文要收購昇陽電腦，他們擔心甲骨文擁有 MySQL 的所有權後，不會再讓大家免費使用 MySQL，所以轉而創了另一個開放原始碼資料庫「MariaDB」，這次是以同一位創始人的小女兒名字「Maria」命名。

Facebook、YouTube、Twitter、Netflix、Spotify 和 Wordpress 等知名網站都是使用 MySQL 或 MariaDB。

phpMyAdmin 這項免費工具於 1998 年釋出，其功能是用於幫忙管理 MySQL 和 MariaDB 的資料庫。

本書使用的程式碼適用 MySQL 5.5 或 MariaDB 5.5 以上的版本，這些版本都能搭配 phpMyAdmin 一起使用。

MySQL 目前釋出的最新版本是 MySQL 8。（註：MySQL 6 未曾釋出，MySQL 7 則沒有出現在個人電腦上，因為這個版本的設計目的是在伺服器叢集上執行。）

# 本書包含哪些主題？

本書內容共分成四大部分，在每個部分的章節裡，你會學到以下這些主題。

## SECTION A：PHP 程式語言入門指導

第一部分的章節會介紹如何使用 PHP 程式碼，寫出 PHP 直譯器可以理解的指令。你將學到：

- 基本的程式設計指令。
- 根據不同的情況執行不同的程式碼，例如，若使用者已經登入，執行某一組程式碼，未登入則執行另一組程式碼。
- 函式如何將執行特定任務需要的所有程式碼組合在一起。
- 類別和物件如何幫助我們表示身邊周遭的事物，並且組織這些程式碼。

## SECTION B：動態網頁設計

第二部分的章節會介紹一套 PHP 提供的工具，學習如何建立動態網頁。你將學到：

- 獲取網頁瀏覽器發送的資料。
- 檢查使用者是否已經提供網頁端需要的資料，以及資料格式是否正確。
- 運用任何已經發送的資料。
- 處理使用者上傳的檔案。
- PHP 表示日期與時間的方式。
- 將資料暫存於 Cookie 與 Session。
- 排除程式碼產生的問題。

## SECTION C：動態資料庫設計

第三部分的章節會介紹如何獲取來自資料庫的資料、將資料顯示於網頁，以及更新資料庫裡現存的資料。你將學到：

- 將資料儲存於資料庫。
- 利用 SQL 語言獲取或更新資料庫裡的資料。
- 在 PHP 網頁裡顯示資料庫的資料。
- 利用 HTML 表單，讓網站訪客更新資料庫裡現存的資料。

## SECTION D：範例網站的延伸應用

第四部分的章節會介紹 PHP 的實務技巧，包含建構網站、網頁應用以及一個應用範例：具有社群功能的陽春版內容管理系統。你將學到：

- 改良程式碼結構。
- 結合其他程式設計師分享的程式碼。
- 利用 PHP 發送電子郵件。
- 讓網站提供會員註冊與登入功能。
- 針對個別會員產生量身定制的網頁內容。
- 使用對搜尋引擎友善的網址。
- 增加社群功能，像是按讚和留言。