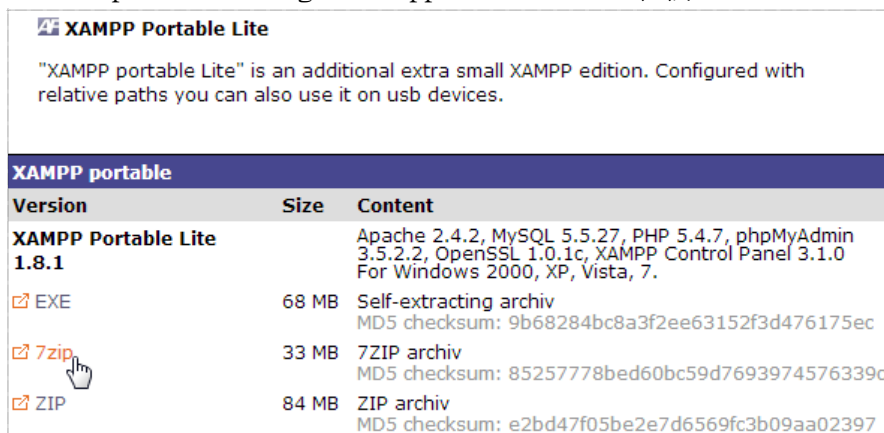


## ➤ 1 執行環境

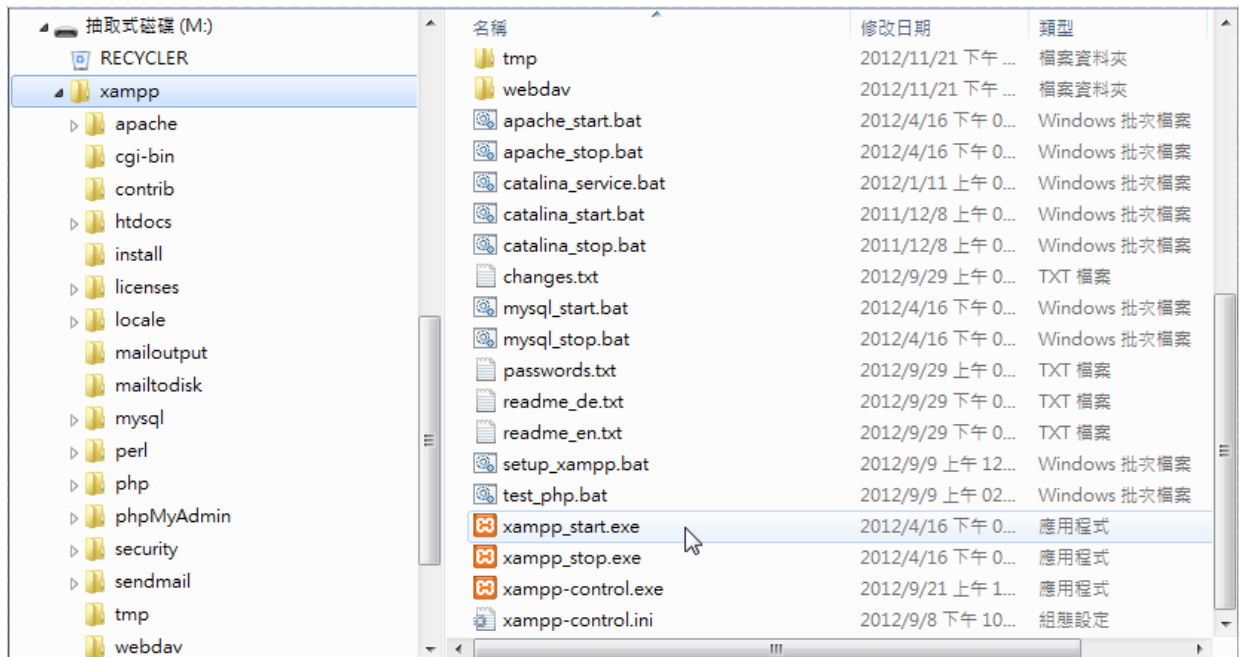
- ◎ Apache + MySQL + PHP (Apache module) , 個別安裝曠日廢時。
- ◎ 開發環境的選擇
  - XAMPP (http://www.apachefriends.org)
  - WAMP (http://www.wampserver.com/en/)
  - AppServ (http://www.appservnetwork.com)
  - MAMP (http://www.mamp.info)

### ➤ 1-1 XAMPP 可攜式安裝

- ◎ 到 <http://www.apachefriends.org/en/xampp-windows.html> 下載 XAMPP Portable Lite 版。



- ◎ 下載 7zip 或 ZIP 檔之後，解壓縮檔案並放在硬碟或隨身碟的根目錄。



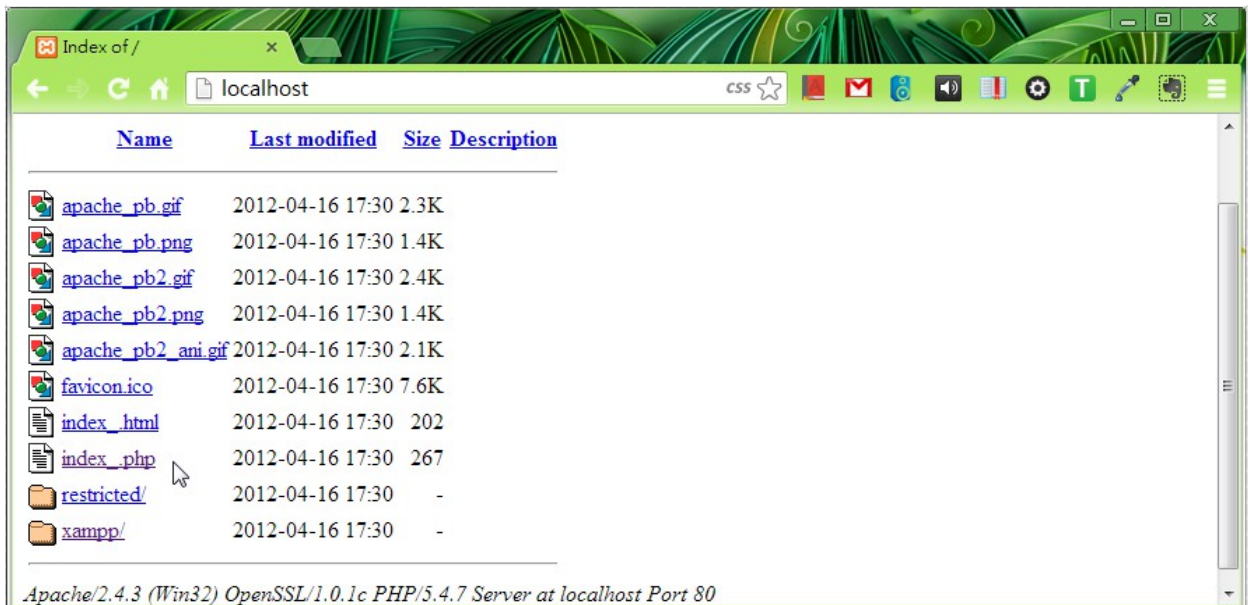
- ◎ 在啟動 XAMPP 之前，先確認 Skype、迅雷等，佔用 80 port 的軟體已閉關。雙擊 xampp/xampp\_start.exe 啟動 XAMPP，自動開啟的 Command Prompt 視窗請勿關閉。當出現「Windows 安全警訊」時，請點選「允許存取」，避免防火牆封鎖服務。關閉

Command Prompt 視窗即關閉 XAMPP。

- ◎ XAMPP 啟動後，會啟動 2 個服務：Apache（預設 port 80）和 MySQL（預設 port 3306）。開啟瀏覽器，在網址列輸入「localhost」或「127.0.0.1」，可以看到預設的首頁。



- ◎ 為了後續開發方便，將預設首頁更名。將網站根目錄（Document Root）「xampp/htdocs」內的預設首頁 index.html 和 index.php 改名為「index\_.html」和「index\_.php」，瀏覽時可以看到檔案列表（預設為開啟）。



- ◎ 用文字編輯器將下列程式存成 M:\xampp\htdocs\phpinfo.php。

```
<?php  
phpinfo();
```

- ◎ 用瀏覽器查看「http://localhost/phpinfo.php」。找出以下的設定值：

```
Loaded Configuration File  
Server Root  
MYSQL_HOME  
DOCUMENT_ROOT
```

## ➤ 1-2 編輯環境

- ◎ 課程使用的開發工具
  - Dreamweaver CS # ( 整合開發環境、視覺編輯)
- ◎ 其他編輯器
  - Notepad++ ( <http://notepad-plus-plus.org/> )
  - Sublime Text 2 ( <http://www.sublimetext.com/2> )
  - Netbeans ( <http://netbeans.org> )

## ➤ 1-3 Apache 的設定

- ◎ Apache 的設定檔位置：

```
M:\xampp\apache\conf\httpd.conf
```

- ◎ 取消自動顯示目錄內容，移除其中的 Indexes。

```
<Directory "/xampp/htdocs">
    Options Indexes FollowSymLinks Includes ExecCGI
    AllowOverride All
    Require all granted
</Directory>
```

- ◎ 設定目錄別名，參考：

```
M:\xampp\apache\conf\extra\httpd-xampp.conf
```

```
Alias /chihlee_digi "C:/Users/shinder/Documents/chihlee_digi"
<Directory "C:/Users/shinder/Documents/chihlee_digi">
    Options Indexes FollowSymLinks Includes ExecCGI
    AllowOverride All
    Order allow,deny
    Allow from all
</Directory>
```

## ➤ 1-4 PHP 錯誤訊息等級設定

- ◎ PHP 的設定檔位置：

```
M:\xampp\php\php.ini
```

- ◎ 搜尋 **error\_reporting**

```
E_ALL & ~E_NOTICE (所有錯誤訊息，提示等級訊息除外)
E_ALL & ~E_NOTICE & ~E_STRICT (所有錯誤訊息，提示和警告等級訊息除外)
E_ALL & ~E_NOTICE & ~E_STRICT & ~E_DEPRECATED
```

## ➤ 1-5 下載 PHP 說明文件

- ◎ 到以下的網址下載「HTML Help file (with user notes)」chm 格式的文件檔。

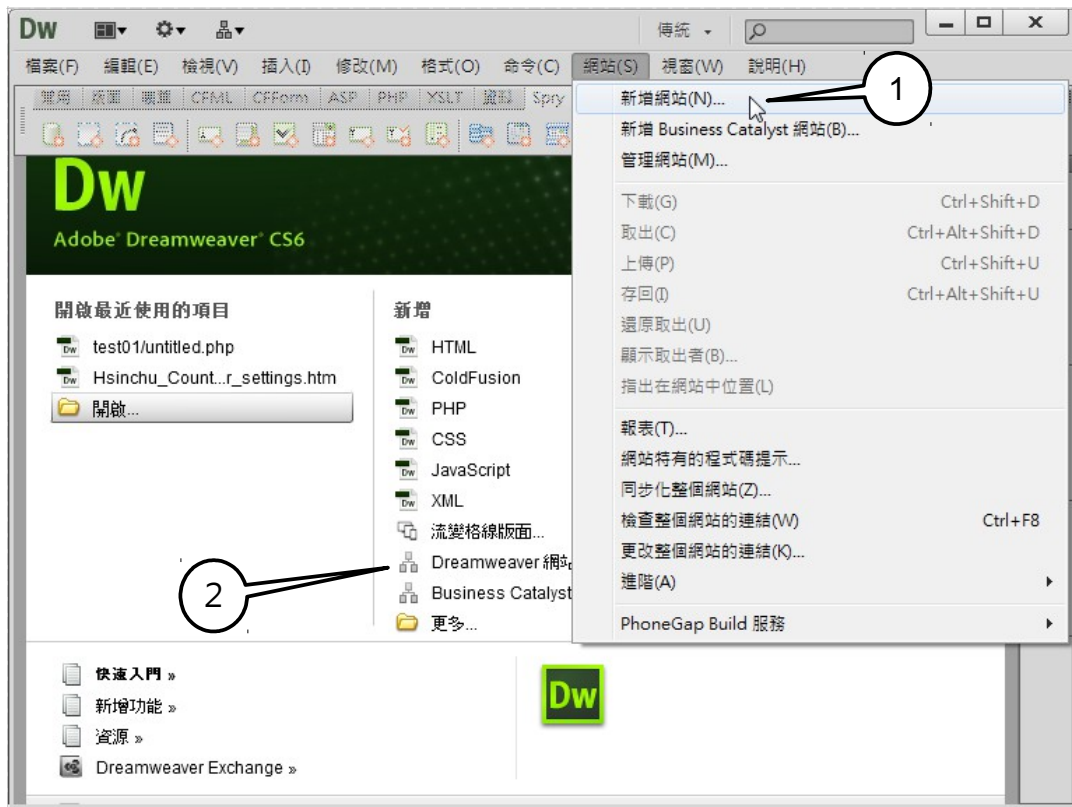
```
http://www.php.net/download-docs.php
```

## ➤ 2 以 Dreamweaver 建立專案

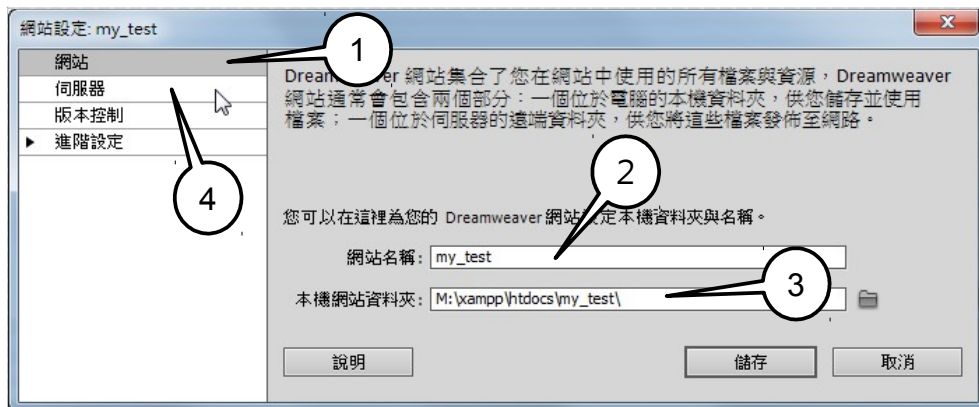
### ➤ 2-1 新增網站專案

◎ 在網站的根目錄（DocumentRoot）新增網站專案所要使用的目錄（my\_test）。目錄名稱請勿使用中文名稱，也避免使用 dash（減號）。

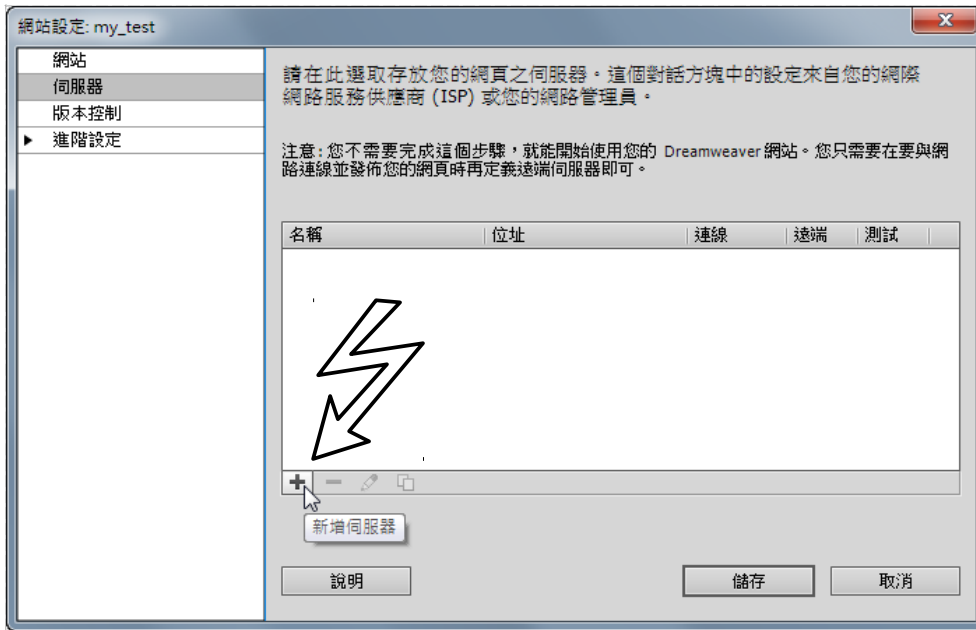
◎ 點選「網站」>「新增網站」①，或在歡迎畫面的「新增」欄上點選「Dreamweaver 網站」②。



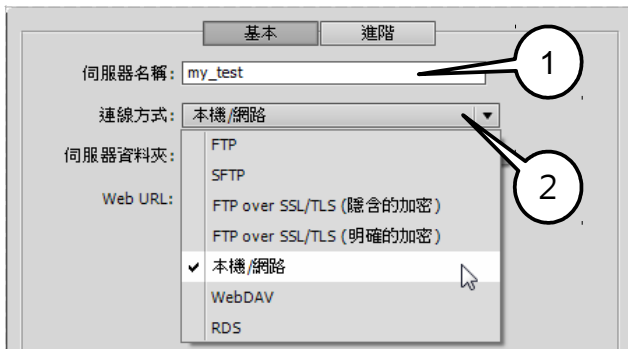
◎ 於「網站設定」視窗的「網站」頁，「網站名稱」欄填入剛剛新建的目錄名稱，也就是以目錄名稱為網站名稱，以方便辨識。「本機網站資料夾」欄填入（或選取）目錄的實體路徑。填寫完成後，點選左側的「伺服器」。



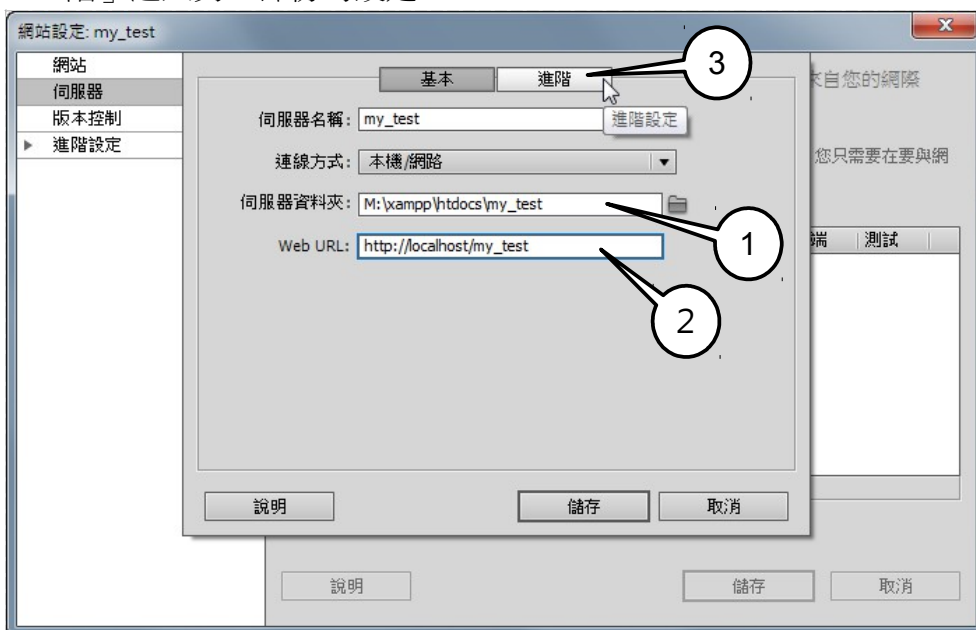
◎ 於「網站設定」視窗的「伺服器」頁，點選「新增伺服器」。



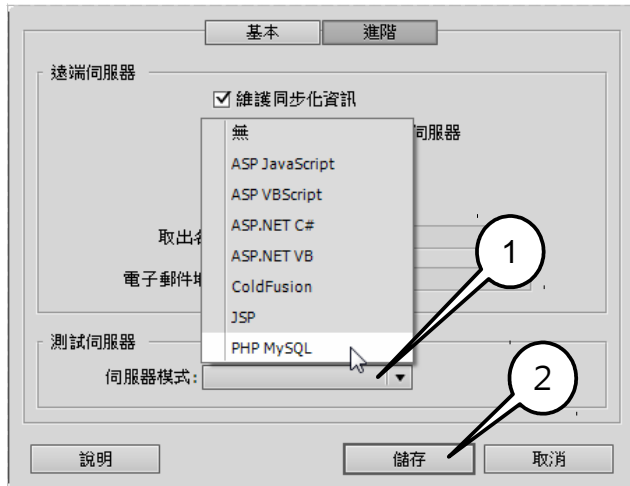
- ◎ 在跳出的面板填入「伺服器名稱」，同網站名稱即可。「連線方式」請選擇「本機/網路」。



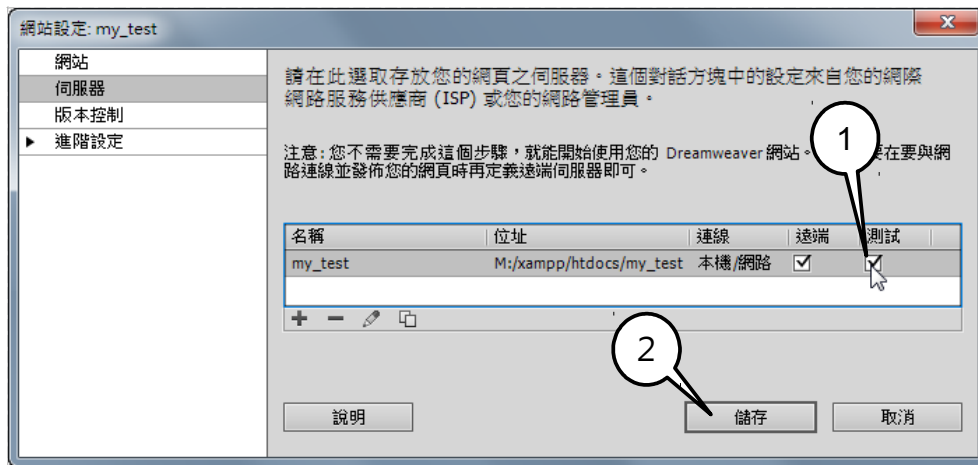
- ◎ 「伺服器資料夾」填入網站目錄的實體路徑；「Web URL」填入「http://localhost/my\_test」。請注意兩欄使用的斜線符號不同方向。接著點選「進階」進入另一部份的設定。



- ◎ 「伺服器模式」選擇「PHP MySQL」，接著點按「儲存」。



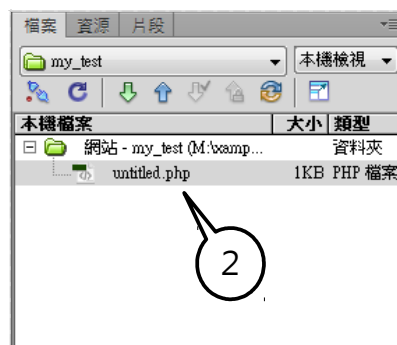
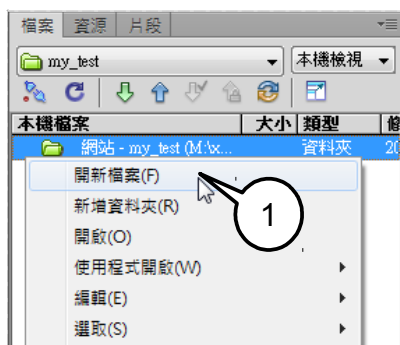
- ◎ 最後勾選「測試」選項，，點按「儲存」後完成設定。



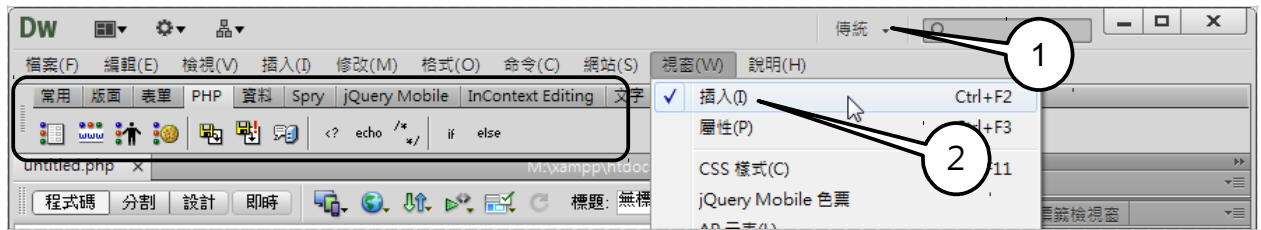
- ◎ 設定完成後，若發現設定有問題，可以透過 Dreamweaver 命令選單上的「網站」>「管理網站」，選擇網站後按「編輯」修改設定內容。

## ➤ 2-2 在 Deamweaver 中編寫 PHP

- ◎ 建立網站後，在檔案面板下的網站圖示按滑鼠右鍵，在跳出的選單上點選「開新檔案」。檢查新開啟的檔案副檔名是否為「.php」；若否，表示「伺服器模式」未選擇「PHP MySQL」。



- ◎ 選擇「傳統」版面配置，也可以依喜好選擇不同的版面配置。傳統配置會開啟「插入」面板組，然後點選 PHP 子面板。若沒有看到「插入」面板組，請勾選「視窗」>「插入」。



- ◎ 點按 PHP 子面板中的「echo」圖示後，可以在鍵盤游標所在位置插入 PHP 標籤及 echo 指令（echo 是將字串輸出至 HTML 頁面）。也可以手動輸入，PHP 子面板只是輔助撰寫程式。



- ◎ 若為了顯示到 HTML 中，PHP 的程式碼通常會放在 <body> 起始標籤和 </body> 結尾標籤之間。寫第一支 PHP 程式，內容只要包含下面一行，就可以看到 php 環境及設定資料。

```
<?php phpinfo();
```

- ◎ 在 dreamweaver，直接按「F12」功能鍵就可以執行並查看網頁！
- ◎ 注意，當 php 程式之後沒有其它 HTML 內容時，php 程式的結束標記 ?> 可以省略。

## ➤ 3 PHP 的運算

- ◎ PHP 一開始就是 WWW 伺服端的技術，用來撰寫動態網頁，其善長處理「文字」資料。HTML 內容本來就是文字，許多資料在輸出到頁面時也會自動轉換成文字。PHP 是 C-like 程式語言，程式裡放置空白（換行）相當有彈性，應確實標上分號（程式敘述結束符號）及遵守縮排規則，以利程式撰寫及除錯。

### ➤ 3-1 常數

- ◎ 所謂常數是不可變更的值。PHP 的常數可以分成三類：一般常數（例：17）、系統常數（例：PHP\_VERSION）及自訂常數。
- ◎ **一般常數**就是一些數值：0、-9、2.5、“abc”等。
- ◎ **系統常數**為 PHP 預設的常數，常用的有：PHP\_VERSION（查看 PHP 版本）、\_\_FILE\_\_（PHP 檔的實體路徑）、TRUE 和 FALSE（布林值的 true 和 false）。系統常數大部份可以由 get\_defined\_constants() 函式取得（取得的資料是陣列類型）。
- ◎ **自訂常數**是程式開發者所定義的常數，可以使用 define() 函式定義，語法為：

```
define(常數名稱, 常數內容, 常數名稱是否不區分大小寫);
```

- 常數名稱通常使用全部大寫的英文字，複合字時中間使用 underscore（\_）分開。
- 常數內容則可以是數值或字串。
- 常數名稱大都使用全大寫，「是否不區分大小寫」一般是不使用的。

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<?php
    define("MY_CONSTANT", "常數 - 值在設定後不可變更");
    echo MY_CONSTANT;
```

- ◎ PHP 程式的每一句敘述請標上分號（;），分號表示程式敘述的結束。只有在最後一行的程式敘述（?> 之前）可以不用標分號，其餘都應該要標示。

### ➤ 3-2 變數

- ◎ PHP 的變數在使用前可以不用宣告，其為**自動型別**，不需要限定變數只能使用某個型別。PHP 變數有個特色，就是在變數前必須要有個美金符號（\$），而且**變數名稱是有大小寫的區別的**。
- ◎ 自訂的常數前不使用\$，而且通常所有字母為大寫。變數前必須要有\$，有區分字母大小寫。
- ◎ PHP 變數的命名規則和一般程式語言一樣，第一個字母不可以是數字。

```
<?php
    $my_var = 66;
    $b = "22";
    $c = "abc";
    echo $my_var + $b;    // 88
    echo "<br />";
    echo $my_var + $c;    // 66
?>
```



- ◎ PHP 的變數有個特點就是可以使用「以變數值為名的變數」（variable variables）。

```
<?php
```

```
    $n = "name";
    $$n = "shinder";
    echo $name;
```

```
?>
```

- ◎ 變數在設值之後，若不想再使用可用 `unset()` 刪除。

### ➤ 3-3 字串

- ◎ PHP 的字串標示方式使用一對雙引號 (") 或一對單引號 (')，不過兩者在使用上有所不同，雙引號標示者裡面若包含變數，會顯示變數值；單引號標示者則否。

```
<?php
```

```
    $a = "Victor";
    echo "Hello, $a";           // Hello, Victor
    echo '<br />';
    echo 'Hello, $a';         // Hello, $a
```

```
?>
```

- ◎ 雙引號標示中，有時輸出變數內容情況會比較複雜，此時可以使用大括號解決。

```
<?php
```

```
    $a = "Victor";
    echo "Hello, $a123 <br/>";           // Hello,
    echo "Hello, {$a}123 <br/>";       // Hello, Victor123
    echo "Hello, ${a}123 <br/>";       // Hello, Victor123
    echo $a . '<br/>';
```

```
?>
```

- ◎ 點符號 (.) 在 PHP 用來做字串串接，不是使用加號 (+)，這點要特別注意。
- ◎ 雙引號和單引號在使用脫逸字元時，亦不太一樣（執行後請查看頁面原始內容）。

```
<?php
```

```
    $a = "xyz\nabc\"def\'ghi\\\"";
    $b = 'xyz\nabc\"def\'ghi\\\"';
    echo $a;
    echo $b;
```

```
?>
```

- ◎ PHP 的字串還有個好處，就是可以換行標示，所以方便定義較長且包含換行的字串。若文字內容也包含雙引號時，可以使用「即為文件」（Heredoc）表示法。Heredoc 表示法是以 `<<<<` 為開頭，緊接「標示名」（可自訂，通常為大寫字母），加換行。Heredoc 結束時，在新的一行使用「標示名」即可。

```

<?php
    $a = "PHP - MySQL
    是好朋友!";
    $b = <<<HDOC
    <br>
    <h1>PHP - </h1>
    <div style="color:#F00;">MySQL</div>
HDOC;
    echo $a;
    echo $b;
?>

```

### ➤ 3-4 運算子

- ◎ PHP 的運算子和其它 C-like 的程式語言大部份是相同的，在此我們來複習一下。算術運算子除了加 + 減 - 乘 \* 除 / 四則運算外，還包括了求餘數 %。這裡的四則運算和數學的四則運算是相同的。

運算子	語法	說明	\$a=7, \$b=2 的運算結果
+	\$a + \$b	\$a 加 \$b	9
-	\$a - \$b	\$a 減 \$b	5
*	\$a * \$b	\$a 乘 \$b	14
/	\$a / \$b	\$a 除以 \$b	3.5
%	\$a % \$b	\$a 除以 \$b 的餘數	1

- ◎ 「\$a = \$a+1」有遞增運算子可以簡化為「\$a++」。 「\$a = \$a + 3」也有替代運算子，稱為算術指定運算子。

算術指定運算子	一般算術並指定的運算式	簡化後的運算式
+=	\$a = \$a + \$b	\$a += \$b
-=	\$a = \$a - \$b	\$a -= \$b
*=	\$a = \$a * \$b	\$a *= \$b
/=	\$a = \$a / \$b	\$a /= \$b
%=	\$a = \$a % \$b	\$a %= \$b

- ◎ 「關係運算子」(relational operator) 或稱為比較運算子。

關係運算子	說明	範例	範例結果
>	是否大於	5 > 2	true
>=	是否大於等於	5 >= 2	true
<	是否小於	5 < 2	false
<=	是否小於等於	5 <= 2	false
==	是否等於	5 == 2	false
!=	是否不等於	5 != 2	true
===	是否等於 (型別必須相同)	5 === "5"	false
!==	是否不等於	5 !== "5"	true

- ◎ 條件運算子「?:」是唯一的三元運算子，其語法如下：

判斷值 ? 真時選擇值 : 假時選擇值

第一個運算元（判斷值）應該為布林值，或是結果為布林值的運算式。當判斷值為 true 時，整個運算式的結果取真時選擇值；反之，判斷值為 false 時，取假時選擇值。通常我們會將整個條件運算結果，存放到某個變數內，所以會如下式：

變數 = 判斷值 ? 真時選擇值 : 假時選擇值;

- ◎ 邏輯運算式和關係運算式有一個共同點，兩者的運算結果都是布林值，不是 true 就是 false。然而，邏輯運算式中的運算對象（運算元），也是布林值。

運算子	意義	語法
&&	且，and	\$a && \$b
and	且，and	\$a and \$b
	或，or	\$a    \$b
or	或，or	\$a or \$b
!	非，not	! \$a

- ◎ 運算子依性質不同分類，優先權高的類別會被先列出來，越往下優先權越低：

- ① 單元運算子
- ② 算術運算子
- ③ 關係運算子
- ④ 邏輯運算子
- ⑤ 指定運算子

- ◎ 除了上述的運算子，分隔符號中的括號「()」、逗號「,」也和運算式有關係。括號可以提升運算子和運算元的運算關係，以括號包起來的運算式會優先運算。逗號可用於分隔宣告變數（或變數宣告的指定式），並不會影響任何值。

### ➤ 3-5 URL 變數 (GET 參數)

- ◎ 網址內容裡 ? 之後為 GET 參數，其格式稱為 URL-encoded。PHP 可以使用 \$\_GET 預設變數取得其內容。

```
<?php
    echo $_GET['a'] + $_GET['b'];
```

- ◎ 測試網頁之後，在網址列輸入下式 ? 之後的內容。

http://localhost/my\_test/untitled1.php?a=12&b=24

## ➤ 4 PHP 的流程控制

- ◎ 流程控制通常可以分成兩類：選擇敘述和迴圈敘述。選擇敘述包含 if/else 和 switch，迴圈敘述包含 for、while、do/while、及 foreach/as。

### ➤ 4-1 if/else 敘述基本用法

- ◎ 簡單地講，if 敘述是選擇執行或不執行，if/else 敘述則是二選一執行。以下是 if/else 選擇結構的語法：

```
if (條件式) {  
    // 區塊敘述一  
} else {  
    // 區塊敘述二  
}
```

if/else 語法意義為：若條件式為 true 時，執行區塊敘述一，否則（條件式為 false 時）執行區塊敘述二。視需要也可以省略 else 區塊。

```
<?php  
    $a = 0;  
    if ( isset($_GET['a']) ) {  
        $a = $_GET['a'];  
    }  
    echo $a;  
?>
```

- ◎ 一般的想法是 PHP 的程式碼是嵌在 HTML 的，其實剛好是相反的，我們可以把 HTML 的內容看成是透過 echo 輸出的。

```
<body>  
<?php  
    if(isset($_GET['name']) and  
        $_GET['name'] == 'shinder') {  
?>  
    <p>您好! Shinder<br />  
    歡迎光臨~  
    </p>  
<?php  
    } else {  
?>  
    <p>您好! 陌生人<br />  
    您無權使用管理系統哦!  
    </p>  
<?php  
    }  
?>  
</body>
```

## ➤ 4-2 if/else 多層巢狀結構

- ◎ if 和 if/else 的基本使用方式是，選或不選，及二選一。如果現在需要一個轉換成績的程式，要把 0~100 分別轉換成 A~F 級。90 分及其以上為 A，80~89 分為 B，70~79 分為 C，60~69 分為 D，59 分及其以下為 F。現在可不是選不選、或二選一的問題，而是要以某個數值成績的範圍來決定成績的等級，這也是個多選一的問題。

```
<?php
    $n = $_GET['n'];
    if ($n >=90) {
        $g = 'A';
    } else {
        if ($n >=80) {
            $g = 'B';
        } else {
            if ($n >=70) {
                $g = 'C';
            } else {
                if ($n >=60) {
                    $g = 'D';
                } else {
                    $g = 'F';
                }
            }
        }
    }
    echo $g;
```

- ◎ 省去 else 後 block 的大括弧

```
<?php
    $n = $_GET['n'];
    if ($n >=90) {
        $g = 'A';
    } else if ($n >=80) {
        $g = 'B';
    } else if ($n >=70) {
        $g = 'C';
    } else if ($n >=60) {
        $g = 'D';
    } else {
        $g = 'F';
    }
    echo $g;
```

- ◎ else if 也可以黏在一起，寫成 elseif。

## ➤ 4-3 switch 選擇敘述

- ◎ if/else 的巢狀結構可以解決多重選擇的問題，在這則是介紹專門設計給多選一使用的 switch 敘述。switch 並不使用條件式來決定執行的區段敘述，而是使用鍵值（通常為變數）。

```
switch (鍵值) {
    case 條件值 1:
        // 區段 1 敘述
        break;
    case 條件值 2:
        // 區段 2 敘述
        break;
    .....
    case 條件值 N:
        // 區段 N 敘述
        break;
    default:
        // default 區段敘述
}
```

執行 switch 敘述時，會先求得鍵值，接著將鍵值和條件值 1 做比對。如果兩值相等，則執行區段 1 敘述，接著遇到 break 之後跳離 switch 敘述；如果兩值不相等，則再往下比對條件值 2，以此類推。如果鍵值和所有的條件值都不相等時，則執行「default:」下的 default 區段敘述。default 部份在 switch 敘述中是選擇性的，可有可無，不一定要放在最後面。

```
<?php
$page = '';
if( isset($_GET['page']) )
    $page = $_GET['page'];

switch( $page ) {
    default :
    case 'main':
        echo 'main page';
        break;
    case 'content':
        echo 'content page';
        break;
    case 'about':
        echo 'about page';
        break;
}
```

## ➤ 4-4 for 迴圈

◎ for 迴圈的語法如下：

```
for (起始式; 條件式; 步進式) {
    //區塊內敘述
}
```

**起始式**—進入迴圈時，一開始執行的程式運算式，只執行一次。通常起始式，多為設定控制迴圈執行變數的起始值。

**條件式**—判斷是否執行迴圈內區塊的依據。如果條件式的結果為 true，則繼續執行；如果為 false，則跳離迴圈。

**步進式**—每經過一次迴圈，就會執行一次的運算式。通常步進式，多為設定控制迴圈執行變數的遞增或遞減運算式。

```
<table border="1" width="400px" cellpadding="0" cellspacing="0">
  <tr>
  <?php
  for($i=0; $i<10; $i++) {
    echo '<td>' . $i . '</td>';
  }
  ?>
  </tr>
</table>
```

## ➤ 4-5 while 迴圈

◎ while 迴圈和 for 迴圈很像，也可以說是 for 迴圈的簡化版，基本上 while 迴圈標頭只有用以判斷的條件式，而沒有起始式和步進式。

```
//起始式
while (條件式) {
    // 區塊內敘述
    // 步進式
}
```

條件式若為 true，則執行大括弧內的區塊內敘述，執行完之後，回到前面再判斷條件式，若還是為 true 則再執行一次區段敘述，直到條件式為 false 時才跳離迴圈。

使用迴圈的時候都必須注意無窮迴圈，尤其是 while 迴圈。while 的控制變數並不像 for 可以直接放在敘述的標頭，所以可能會忘了加步進式，這個時候就會形成無窮迴圈。

◎ 條件式直接使用 true 是在「不知道要跑幾次迴圈」時使用，不過還是可以搭配跳離迴圈的關鍵字 break 來使用，讓它不至於無窮的執行下去。

```
while(true) {
    // 區塊內敘述
}
```

◎ 從 MySQL 的資料集 (Recordset) 取出資料就常使用 while 迴圈，資料集取完後會得到 NULL (空值)，所以不需要控制變數就可以讓迴圈正常停止。

## ➤ 4-6 do/while 迴圈

◎ do/while 迴圈結構和之前介紹的兩種不太相同。

```
do {
    // 區塊內敘述
} while (條件式);
```

do/while 結構以 do 關鍵字為開頭，接著是大括弧包起來的區塊內敘述，最後是 while 關鍵字和小括弧包著的條件式。

請注意，for 和 while 迴圈都是以右大括弧結尾，所以不需要再加分號 (;) 標示敘述的結束。但是，do/while 結尾是右小括弧，並不能代表一個區塊的結束，因此要加上分號，以標示迴圈結尾。

若拿 while 迴圈和 do/while 迴圈做比較，最明顯的不同就是條件式的位置。do/while 的條件式擺在結構的最後面，也就表示**先執行區塊敘述一次，再做條件式判斷**，這也可以說是 do/while 的特色。

## ➤ 4-7 break 與 continue

- ◎ 關鍵字 break 和 continue，可以改變迴圈的流程。break 曾經在使用 switch 敘述時看到，不過在這將介紹 break 和迴圈的關係。
- ◎ 在 switch 選擇敘述中，我們使用過 break，讓流程跳離 switch 敘述區塊。而在 for、while、do/while 迴圈中，break 的作用也是相同的，都是跳離敘述區塊（跳離迴圈的大括號範圍）。一般使用 break 是為了在特殊情況下提早跳離重複結構。
- ◎ continue 的功能是跳到迴圈的起始處。

```
<?php
for ($j = 1; $j<7; $j++) {
    if ($j == 4)
        continue;
    echo $j;
}
```

## ➤ 5 PHP 的陣列

- ◎ 何謂陣列呢？簡單地講，陣列就是「多個擁有相同名稱的變數集合」。PHP 的陣列又可細分成「索引式陣列」（Indexed Array）和「關聯式陣列」（Associated Array）。在 PHP 索引式和關聯式是可以混用的，不過依它們的特性我們先分開來討論。

### ➤ 5-1 索引式陣列

- ◎ 元素都有相同的名稱，為了區別每個陣列元素，它們都有各自的位置編號。使用陣列時，一般用 array() 函式建立函式的實體方式如下：

```
$ar = array(3, 2, 2, 0, 4, 1);
```

- ◎ 為了指出是哪個陣列元素時，必須以陣列名稱，後接一對中括弧 []，中括弧內放入位置編號。位置編號一般也稱為索引，**索引由 0 開始**，接著是 1、2、3... 至元素個數減一。陣列的索引應該為大於或等於 0 的整數，或者結果為整數的運算式。

```
echo $ar[1];           // 2
```



- ◎ PHP 的陣列是動態的，可以動態加入元素或移除元素，而且元素沒有限定類型。用 [] 直接「建立陣列並放入第一個元素值」或「推入一個元素值」：

```
<?php
    $br[] = 3;
    $br[] = 2;
    $br[] = 2;
    $br[] = 0;
    $br[] = 4;
    $br[] = 1;
    print_r($br);
?>
```

- ◎ `print_r()` 和 `var_dump()` 常用來查看資料內容和除錯。
- ◎ `count()` 函式能取得陣列裡元素的個數。

```
<?php
    $ar = array(9,5,2,'abc', true);
    for($i=0; $i<count($ar); $i++) {
        echo " {$ar[$i]} <br />";
    }
?>
```

- ◎ PHP 的內建函式名稱，雖然大部份會用有意思的名稱，有些則用縮寫容易讓人混淆。而且由於發展的歷史，大部份的函式是全域的，而不是透過物件導向的方式歸類在某些類別下。建議初學者常查看 PHP 手冊，或在 php.net 的「search for」進行查詢。
- ◎ 若要不理會索引值是否連續，依序取得陣列的元素值，可以使用 `foreach/as` 迴圈。

```
foreach (陣列變數 as 列舉變數) {
    //迴圈主體內容
}
```

列舉變數是自訂變數，由開發者自行決定。列舉變數會依迴圈執行的圈數，依序取得元素值。

```
<?php
    $ar = array(1,2,3);
    $ar[9] = 7;

    foreach($ar as $v) {
        echo "$v <br />";
    }
?>
```

## ➤ 5-2 關聯式陣列

- ◎ PHP 的關聯式陣列是由一個字串 key 對應到一個 value。
- ◎ 關聯式陣列可以看成是 hash table（雜湊表）的實作，只要給一個 key（屬性名稱）就可以得到對應的 value（屬性值）。
- ◎ PHP 的關聯式陣列同樣用 `array()` 函式建立，key 和 value 之間用胖箭頭「`=>`」來設定。「`=>`」由一個等號和大於符號組成，注意中間沒有空白。

```
<?php
    $ar = array(
        3      => 'abc',
        "3"    => 'def',
        'name' => 'shinder',
        "pw"   => "pass" );

    echo $ar[3] . "<br/>";
    echo $ar["3"] . "<br/>";
    echo $ar['name'] . "<br/>";
```

注意，使用關聯式陣列時，通常 key 不使用數值，使用數值會變成使用索引式陣列。key 若可以轉換成數值，即使用雙引號標示，同樣視為相同，所以「3」和「" 3"」視為相同的 key，key 是唯一的。

- ◎ 關聯式陣列也可以用 `foreach/as` 迴圈來取值，其語法如下：

```
foreach(陣列變數 as 鍵變數 => 值變數) {
    //迴圈主體內容
}
```

「鍵變數」和「值變數」同樣使用「`=>`」指明其關係。

```
<?php
    $ar = array(
        3 => 'abc',
        "3" => 'def',
        'name' => 'shinder',
        "pw" => "pass" );

    foreach($ar as $k=>$v) {
        echo "$k : $v<br />";
    }
```

## ➤ 5-3 陣列的「`=`」設定

- ◎ PHP 陣列在使用「`=`」做設定運算時，並不是設定參照（reference），而是做實體的複製。若要設定參照，必須使用 `&` 符號。

```

<pre>
<?php
    $ar = array(
        "num" => 17,
        'name' => 'shinder',
    );

    $br = $ar;
    $cr = &$ar;
    $ar['num'] = 26;

    print_r($ar);
    print_r($br);
    print_r($cr);

?>
</pre>

```

## ➤ 5-4 \$\_GET 和 \$\_POST

- ◎ PHP 用來取出客戶端傳來資料的 \$\_GET 和 \$\_POST 是以陣列的形式存在。以下的例子在執行時，試著在網址後接著輸入「?a=123&bb=yes」，a 和 bb 會是 \$\_GET 的 key，而 123 和 yes 分別是 \$\_GET 的 value。

```

<pre>
<?php
    print_r($_GET);

?>
</pre>

```

- ◎ 測試登入表單。

```

<!DOCTYPE html>
<html>
<head>
    <title>example</title>
    <meta http-equiv="content-type" content="text/html; charset=utf-8">
</head>
<style type="text/css">.titleCol { float:left; width:100px; }</style>
<body>
    <?php
        if( isset($_POST['email']) ) {
            echo '<pre>' . print_r($_POST, true) . '</pre>';
        } else {
            ?>
            <form name="form1" method="post" action="">
                <div class="titleCol">Email:</div>
                <input type="text" id="email" name="email" />
                <br />
                <div class="titleCol">Password:</div>
                <input type="password" id="password" name="password" />
                <br />
                <div class="titleCol">&nbsp;</div>
            </form>
        }
    </?php>

```

```
        <input type="submit" value="登入" />
    </form>
    <?php
        }
    ?>
</body>
```

- ◎ \$\_REQUEST 的功能是？方便，但是...
- ◎ 如何用 Chrome 觀察表單的 get 和 post 方法？

## ➤ 6 PHP 的函數

- ◎ 函式 (function) 就是有某種功能的小程式，可以看成是一個神奇的盒子，輸進一些資料後，可以得到處理過後的結果。一個程式是為了解決某個大問題，要一下子解決一個大問題並不容易著手，把大問題分開成許多小問題，再個個擊破，就比較容易。使用函式就是把問題分開個別解決。
- ◎ 之前使用過的 `phpinfo()`、`array()` 和 `count()` 都是 PHP 內訂的函式。

### ➤ 6-1 自訂函式

- ◎ 自訂函式的語法以 `function` 關鍵字為開頭：

```
function 函式名(參數 1, 參數 2, ...) {
    // 函式的內容敘述
    return 回傳值;
}
```

```
<?php
    function multi($a, $b=10) {
        return $a * $b;
    }
    echo multi(6, 7) . '<br />'; // 42
    echo multi(8) . '<br />'; // 80
```

- ◎ 以參照為參數。

```
<?php
    function swap(&$a, &$b) {
        $c = $a;
        $a = $b;
        $b = $c;
    }
    $m = 100;
    $n = 'abc';
    swap($m, $n);
    echo "$m $n";
```

## ➤ 6-2 函式外的變數

- ◎ 在 PHP，函式內不能直接使用函式外設定的變數，若要使用必須在函式內宣告變數為**全域變數**，才能使用。這樣的做法是，儘量避免在函式內使用外面的變數，可以增加函式的獨立性和再用性。

```
<?php
    $g1 = 1000;
    $g2 = 2000;

    function fun() {
        global $g2;
        echo ">{$g1}< >{$g2}<";
    }
    fun();
```

## ➤ 7 PHP 的類別

- ◎ 類別可以看成是設計圖，依照設計圖實作出來的就是物件。類別也可以想成是使用者定義的資料類型，而物件是以類別宣告出來的變數。

### ➤ 7-1 簡單類別

- ◎ 若先不考慮繼承，類別的定義語法如下：

```
class 類別名稱 //類別定義的標頭
{
    // 屬性宣告
    // 建構函式定義
    // 方法定義
}
```

- ◎ 屬性就是物件的變數。屬性的宣告可以用修飾字 var、public 和 private。var 和 public 的功能是相同的，都是定義可以公開使用的屬性；private 則是定義私有屬性，只能在類別裡使用。下式用來說明 public 和 private 的區別。

```
<?php
class Person {
    var $name;
    public $mobile;
    private $sno = 'secret';
}

$p = new Person;
$p->name = "Shinder Lin";
$p->mobile = "0918123456";
// $p->sno = "123"; // 取用私有屬性將發生錯誤
print_r($p);
echo $p->name;
```

- ◎ 一個 PHP 檔可以定義許多類別。
- ◎ PHP 5.3 以前的類別沒有 package 之分，需要時 include 或 require 進來即可。

## ➤ 7-2 方法

- ◎ 「方法」其實就是函式，主要區別是：方法屬於物件，函式則是全域的。在修飾字方面，使用 private 表示私有方法，否則為公開方法。類別定義中，除了「宣告並指定屬性的敘述」可以直接放在類別內，其餘敘述都應該放在方法內。

```
<?php
class Person {
    private $name;
    function setName($n) {
        $this->name = $n;
    }
    function getName() {
        return $this->name;
    }
}
$p = new Person;
$p->setName("Victor");
echo $p->getName();
```

## ➤ 7-3 建構函式及解構函式

- ◎ 建構函式是物件在使用 new 建立之後，會自動被呼叫的函式，用來做物件的初始設定之用。只要定義一個名稱和類別相同的 function 就是建構函式，然而在 PHP5 建議使用「\_\_construct」。PHP 5 在物件被刪除前會呼叫解構函式「\_\_destruct」，有時資料庫的操作在物件刪除後就要關閉，即可在解構函式中進行關閉動作。

```
<?php
class Person {
    var $name;
    function __construct($n) {
        $this->name = $n;
        echo $this->name . '建立<br />';
    }

    function __destruct() {
        echo $this->name . '解構<br />';
    }
}
$p = new Person('p');
unset($p);
$q = new Person('q');
```

- ◎ unset() 刪除物件前，解構函式會被呼叫；程式結束前（刪除所有物件前）所有物件的解構函式都會被呼叫。

## ➤ 7-4 類別方法

- ◎ 有些方法沒有用到 `$this`，而且其目的是用來直接呼叫的，稱為類別方法。類別方法定義方式和一般方法一樣，不同處在於呼叫方式，使用「`::`」。

```
<?php
    class MyMath {
        function distance($x1, $y1, $x2, $y2) {
            $dx = $x1 - $x2;
            $dy = $y1 - $y2;
            return sqrt($dx * $dx + $dy * $dy);
        }
    }
    echo MyMath::distance(1, 1, 11, 11);
```

## ➤ 8 Cookie

- ◎ Cookie 是網站透過瀏覽器將資料存放在硬碟裡的機制。基於安全性，Cookie 有幾個特性：
  1. **以網域為單位**：例如 pchome.com.tw 不可讀取 yahoo.com.tw 所設定的 Cookie。
  2. **大小受限**：不同的瀏覽器有不同的大小限制，一般為 4096 bytes。
  3. **客戶端可能可以看得到**：IE 的 Cookie 會放在「C:\Documents and Settings\使用者\Local Settings\Temporary Internet Files」。
  4. **客戶端可以設定**：瀏覽器可以設定是否使用 Cookie，一般設定為使用，筆者也建議使用。
  5. **過期**：Cookie 是有期限的，過期就會失效。
  6. **自動傳給主機**：瀏覽器每次做 request 時，都會將有效的 Cookie 放在 HTTP Header 裡，傳給 server。

- ◎ Cookie 的內容應該要簡短，而且不應存放敏感性資料（例如：密碼）。Cookie 是好的，可以幫助網站做客製化的設定，大部份的 Session 機制也是建立在 Cookie 之上。

- ◎ PHP 設定 Cookie 使用 `setcookie()` 函式：

```
setcookie("Cookie 變數名稱", "Cookie 數值", "期限", "路徑", "網域", "安全");
```

通常我們只會用到前三個參數：「Cookie 名稱」、「Cookie 值」和「期限」。「期限」用的是 Unix 系統上的時間戳記（Unix timestamp），單位為秒，用 `time()` 函式可以取得當下的時間戳記。若不使用「期限」參數，瀏覽器的所有視窗關閉時為過期時間。

- ◎ PHP 讀取 Cookie 使用 `$_COOKIE` 預設的陣列變數。

```
<?php
    if(! isset($_COOKIE['mycookie'])) {
        setcookie("mycookie", "10", time()+30);
```

```

        } else {
            $c = $_COOKIE['mycookie'];
            setcookie("mycookie", $c+1, time()+30);
        }
    ?>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Cookie</title>
</head>

<body>
<?php
    if(! isset( $_COOKIE['mycookie'] ) ){
        echo '第一次設定 Cookie, 或重新設定';
    } else {
        echo $_COOKIE['mycookie'];
    }
?>
</body>
</html>

```

- ◎ 第一次用 `setcookie()` 設值時，用 `$_COOKIE` 是讀不到的，因為 server 是要告訴 client 設定 Cookie，並未從 client 端得到 Cookie。
- ◎ Cookie 的設定（使用 `setcookie` 函式）是放在 HTTP 的 Header（表頭）裡，應該在 HTML 內容還沒有出現時設定。若 PHP 的設定「`output_buffering`」為「On」時，表示 HTTP 內容有做緩衝，或許不需要放在最前面。不過避免「`output_buffering`」為「Off」，或者緩衝區已滿的情況下，`setcookie` 函式應該在 HTML 內容還沒有出現時設定。

## ➤ 9 Session

一般講「會談」（session）是指客戶還在瀏覽該站，或者還在使用該站的服務。Session 是伺服器為了知道客戶端是否還在瀏覽該站的機制。為了不佔用連線資源，HTTP 是個溝通完就斷線的協定，所以伺服器無法得知客戶端是否還在該站瀏覽。因此，就必須訂定一些規則來實現這個機制。

首先，server 必須先分辨 client，一般有兩種做法。第一種，在網站的所有連結中都置入一個 GET 參數「客戶識別碼」（Session ID），這種方式實做起來較為麻煩，而且一不小心就會置換「客戶識別碼」。

第二種，是利用 Cookie 存放「Session ID」，只要在 client 第一次拜訪時將 Session ID 存入 Cookie 即可。這種是常用也是比較方便的做法。此時 Session 的存活就會依賴 Cookie 是否有效而定。

若 client 的瀏覽器停在某個網頁，使用者可能某些原因（例如：去洗澡）久久未再拜訪該站，或者根本就已離開該站。此時會依 Session 的存活時間，決定 Session 是否有效。當然，server 是以 client 最後一次拜訪開始計時的；若 client 在 Session 存活時間內，持續訪問該站，Session 就會一直有效。一般 Session 的存活時間會設定為 5~20 分鐘，流量越大的網站，Session 存活時間設得越短越節省主機資源。



有了 Session ID 之後，server 會在主機記憶體為每個 Session ID 建立一個對應的 Session 物件，資料就存在 Session 物件裡。有效的 Session 越多，使用的記憶體就會越多。

- ◎ 常用操作 Session 的函式有：session\_start() 啟用 Session、session\_destroy() 清除 Session 物件。讀取和設定 Session 使用 \$\_SESSION 預設變數。session\_name 函式可以讀取或設定 Session ID 名稱，預設為 PHPSESSID。

```
<?php
    session_start();
    if(isset($_POST['name']) and isset($_POST['password'])) {
        if($_POST['name']=='shinder' and $_POST['password']=='12345') {
            $_SESSION['login'] = $_POST['name'];
        }
    }
?>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <title>login test</title>
</head>
<body>
<?php
    if( isset($_SESSION['login']) ) {
        echo $_SESSION['login'] . ', 您好!';
        echo '<br /><a href="session_logout.php">登出</a>';
    } else {
?>
    <form id="form1" name="form1" method="post" action="">
        <label style="display:block">
            名稱<input type="text" name="name" />
        </label>
        <label style="display:block">
            密碼<input type="password" name="password" />
        </label>
        <label>
            <input type="submit" name="button" value="Submit" />
        </label>
    </form>
<?php
    }
?>
</body>
</html>
```

- ◎ 登出時，只要用 session\_destroy() 函式即可。

```
<?php
session_start();
session_destroy();
header("Location: session_login.php");
```

## ➤ 10 用 phpMyAdmin 建立資料庫

- ◎ MySQL 的管理工具，最基本的是 mysql client（在 Windows 上是 mysql.exe），不過由於它是文字指令介面，不是圖形介面。使用時要熟悉指令，這對一開始學習 MySQL 的程式設計師而言增加了門檻。
- ◎ phpMyAdmin 是 PHP 撰寫而成的管理工具、圖形網頁介面、開放軟體、費免使用。可以執行 PHP 的環境就可以使用，而且大部份 PHP/MySQL 虛擬主機商都提供 phpMyAdmin 介面，讓使用者自行管理。

### ➤ 10-1 建立資料庫

- ◎ 透過 <http://localhost/xampp/> 左側的工具，打開 phpMyAdmin 主頁。然後在「建立新資料庫」的欄位填入資料庫的名稱，「校對」欄選「utf\_general\_ci」，接著按「建立」。



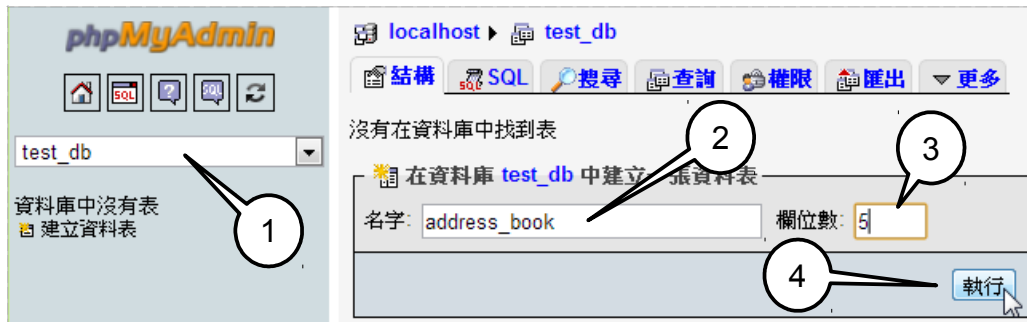
- ① 點選首頁（主目錄）。
- ② 點選「資料庫」分頁。
- ③ 填入資料庫名稱「test\_db」。
- ④ 點選「utf8\_general\_ci」。
- ⑤ 點按「建立」。

按下「建立」之後，頁面會顯示建立成功，有些版本會秀出建立資料庫的 SQL 語法：

```
CREATE DATABASE `test_db` DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
```

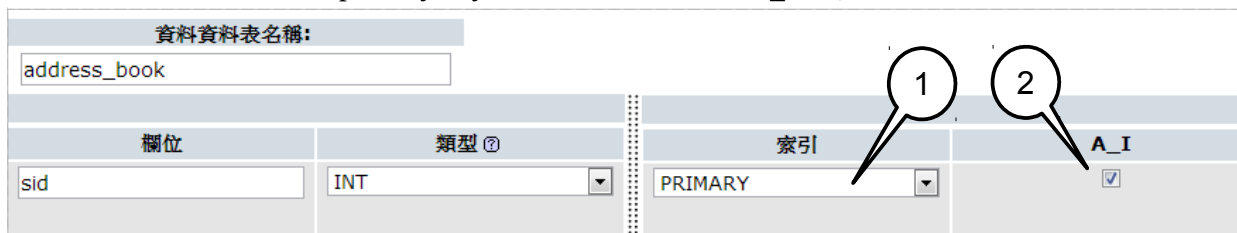
### ➤ 10-2 建立資料表

- ◎ 建立資料表之前，先規劃好用途及 ER mapping（Entity-relation mapping）。ER mapping 是所謂實體關係對應。以建立通訊錄為例，通訊錄（簡化後）預計存放四項資料：姓名、email、手機號碼和地址，這個資料表至少要有 4 欄，外加一個流水號當主鍵，應該要有 5 欄。選定資料庫，並建立新資料表，輸入資料表名稱及欄位數目。



- ① 點選剛剛建立的資料庫。
- ② 輸入所要建立的資料表名稱「address\_book」。
- ③ 填入欄位數為5。
- ④ 點按「執行」。

- ◎ 第一個欄位為流水號名稱設定為「sid」，型態為INT整數，在「索引鍵 INDEX」欄設定為PRIMARY（primary key，主鍵），並勾選A\_I的核選盒。



- ① 設定為PRIMARY。
- ② 勾選A\_I的核選盒。

- ◎ 主鍵的意思是該欄位為識別資料的主要欄位，所以該欄位的值不會重複。主鍵會編列索引以加快資料的查詢，一張資料表只會有一欄為主鍵。若其它欄也要有索引的功能，可設定為索引鍵。設定為索引鍵可加快查詢，同時也會較耗資源。
- ◎ 勾選A\_I核選盒的意思是該欄為Auto Increment（自動加號）。新增資料時，sid欄不用設定，該值會自動累加。
- ◎ 資料型態可以使用的有許多種，常用的有，INT（整數）、FLOAT（浮點數）、VARCHAR（不定長度的字串）、TEXT（大量的文字）、DATE（日期）、DATETIME（日期及時間）、TIMESTAMP（時間戳記）。
- ◎ 其它欄位名稱分別為：name、email、mobile和address，型態皆使用VARCHAR。VARCHAR的意思是不定長度的字串，使用VARCHAR要記得「一定要設定長度」，沒設長度在建立資料表時會發生錯誤。設定長度是限定字串的最大長度，所以在設定時應該寬裕一點，以免到時候資料放不下而被截斷。
- ◎ 欄位設定好之後，儲存引擎使用「MyISAM」，校對使用「utf8\_general\_ci」，按「儲存」按鈕建立。

建立資料表

資料資料表名稱:  
address\_book

欄位	類型	長度/值1	預設2
sid	INT		無
name	VARCHAR	255	無
email	VARCHAR	255	無
mobile	VARCHAR	255	無
address	VARCHAR	255	無

表註釋:

分區定義:

儲存引擎: MyISAM

排序規則: utf8\_general\_ci

儲存 或 新增 1 個欄位 執行

- ① 填入各欄位名稱。
  - ② 選擇適當的資料類型。
  - ③ 設定資料長度，尤其是 VARCHAR。
  - ④ 選擇儲存引擎，通常使用「MyISAM」。
  - ⑤ 排序規則（校對編碼）選擇「utf8\_general\_ci」。
  - ⑥ 點按「儲存」。
- ◎ MySQL 的設計是儲存引擎是低耦合的，可以選擇使用不同的儲存引擎（每個資料庫在選定儲存引擎後就不能更換）。常用的儲存引擎為「MyISAM」，其特色是快速，一般網站使用這個就可以了。若要有 transaction（交易）的功能，可以使用「InnoDB」。
  - ◎ 所謂「校對編碼」，是資料以什麼編碼來看待。校對編碼從頭到尾都統一用 utf8 以避免不必要的亂碼發生。每個資料欄都有「校對」的設定，只要設定資料表的校對即可。
  - ◎ 欄位名稱可不可以使用中文？可以，資料庫名稱、資料表名稱和欄位名稱都可以是中文，不過不建議使用中文，以避免造成不必要的困擾。
  - ◎ 如果在建立資料表的過程，發現當初設想的欄位數太少，可以在「新增？個欄位」下填入要增加的欄位數，並按「執行」。若欄位數太多，沒用到的只要欄位名稱留白即可。

### ➤ 10-3 新增資料

- ◎ 點選上方選單的「新增」按鈕，進入新增資料的頁面，並填入所欲新增的資料，再按「執行」按鈕新增。sid 欄位的值可以不用填，因為我們已經設定它為 Auto Increment（自動增加）。

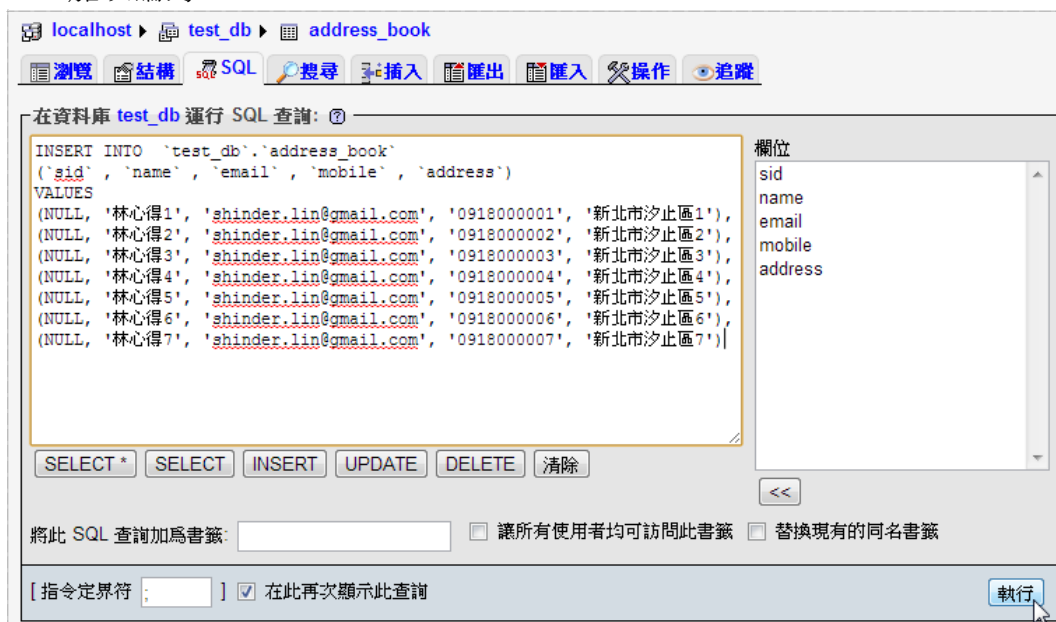


- ① 點選資料表。
- ② 點選「插入」分頁。
- ③ 填入各欄的值。
- ④ 點按「執行」。

- ◎ 點按「執行」後，將顯示插入狀況的訊息。完成新增資料，同時會顯示其 SQL 語法。

```
INSERT INTO `test_db`.`address_book`
(`sid`, `name`, `email`, `mobile`, `address`)
VALUES
(NULL, '林心得', 'shinder.lin@gmail.com', '0918000000', '新北市汐止區');
```

- ◎ 請注意，上述的欄位名稱都使用「`」包裹，字串的值用單引號「'」包裹。phpMyAdmin 的新增頁面每次可以新增 2 筆資料，如果有多筆資料，用其新增頁面就顯得較為不方便。可以使用文字編輯器，將之前新增資料的 SQL 敘述複製貼上，再加以編寫。



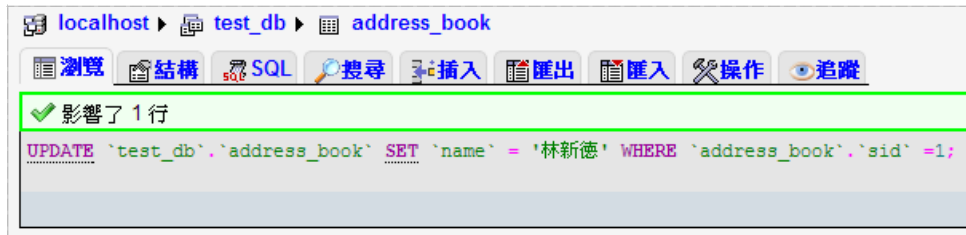
### ➤ 10-4 修改資料

◎ 新增一些資料後，點按資料表名稱。再按某筆資料前的編輯圖示以進行修改。



- ① 點選「瀏覽」分頁。
- ② 點按「編輯」。
- ③ 或點按「快速編輯」。

◎ 修改 name 欄位的資料後，按「執行」。



```
UPDATE `test_db`.`address_book` SET `name` = '林新德'  
WHERE `address_book`.`sid` = 1;
```

◎ 在使用時會先指定資料庫，所以資料庫名稱其實是可以省略的。同樣，只有使用一張表，WHERE 後面的資料表名稱可以省略，通常寫成：

```
UPDATE `address_book` SET `name` = '林新德' WHERE `sid` = 1;
```

### ➤ 11 phpMyAdmin 的安全性設定

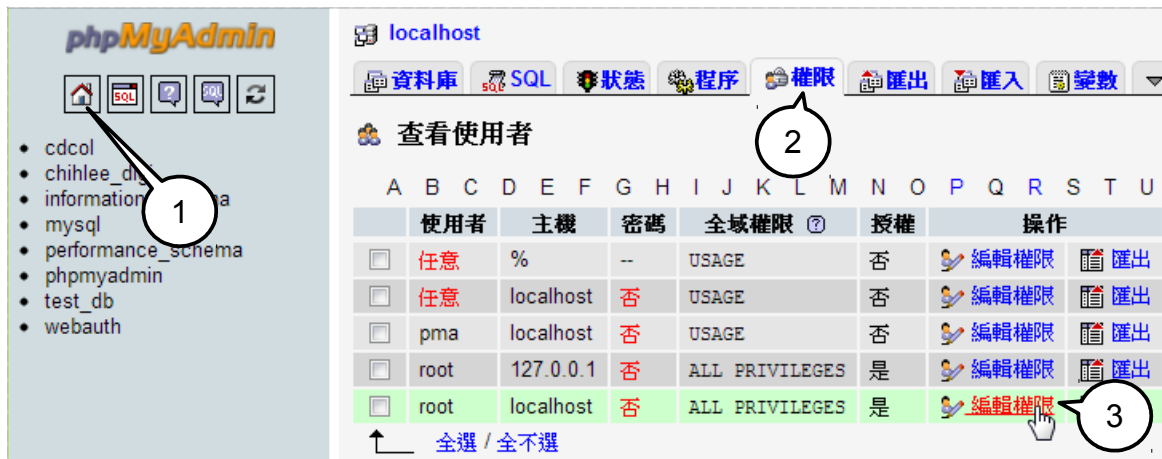
◎ MySQL 預設的使用者是 root，沒有密碼。在使用 phpMyAdmin 時，連接的方式是用 config，直接以 root 為使用者，無密碼連接，因此我們不需要額外的設定就能用 phpMyAdmin 操作 MySQL 了。然而進到 phpMyAdmin 後，會看到警告訊息。

您設定檔案中的設定 (空密碼的 root) 與 MySQL 預設管理員帳號對應。您的 MySQL 服務器使用預設值運行當然沒有問題，不過這樣的話，被入侵的可能性會很大，我們強烈建議您應該立即給 'root' 使用者設定一個密碼來補上這個安全漏洞

#### ➤ 11-1 設定使用者及密碼

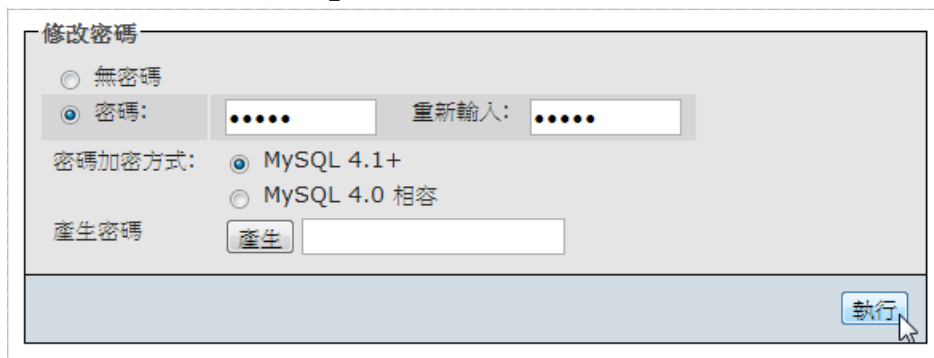
◎ 在開發時，由於是測試環境，使用 root 及無密碼是沒問題的。如果有隱私問題，或者要直接用發佈時使用的帳號及密碼，就必須設定使用者和密碼。

- ◎ 在第一次變更密碼時，最好用紙筆記下密碼，倘若忘記密碼就可能要重新安裝 XAMPP。
- ◎ 先到 phpMyAdmin 的主目錄頁面，點選「權限」，此時可以看到「使用者一覽」裡有兩個授權的使用者都是 root，可是「從哪登入」的主機是不同的。只要主機 IP 或網域名稱看起來不一樣，就視為不同的主機（不管它們最後是不是指向相同的主機）。



- ① 點選「首頁」（主目錄）。
- ② 點按「權限」分頁。
- ③ 點按「編輯權限」。

- ◎ 點選「編輯權限」圖示，進入權限編輯及密碼設定頁面，找到密碼設定的表單（權限全開不要變動），輸入密碼兩次（在此密碼為 admin，範例！要上線的 Service 勿用），點按「執行」。



- ◎ 變更密碼後，可以看到成功更改的字樣及對應的 SQL 敘述，重新載入頁面後就會發現「拒絕存取」（Access denied）。



- ◎ MySQL 使用者「[root@localhost](#)」的密碼更改後，phpMyAdmin 還是使用無密碼登入，所以被拒絕存取。我們必須修改 phpMyAdmin 的設定檔，位於「D:\xampp\phpMyAdmin」的「config.inc.php」。注意，另一個檔名有點雷同的「config.sample.inc.php」，就如其名是個 sample，參考用，改這個是無效的。

- ◎ 用文字編輯器打開「config.inc.php」之後，在密碼設定行加入密碼：

```
$cfg['Servers'][$i]['user'] = 'root';  
$cfg['Servers'][$i]['password'] = 'admin';
```

- ◎ config.inc.php 存檔後，phpMyAdmin 重新載入就可以正常操作 MySQL 了。忘記密碼怎麼辦呢？重新安裝應該是最快的方式。

## ➤ 11-2 不將密碼放在 config.inc.php

- ◎ 某些情況您可能不希望直接開 phpMyAdmin 就可以操作 MySQL，希望有個登入的管控。此時可以改用不同的 auth\_type，原來是使用「config」的方式。在 config.inc.php，我們可以找到這行：

```
$cfg['Servers'][$i]['auth_type'] = 'config';
```

- ◎ 除了 config，我們還可以使用 http 或 cookie。把上行的「config」改成「http」就是使用 http 的方式，使用者名稱和密碼並不放在 config.inc.php 裡，而是登入時輸入。

- ◎ 若使用 cookie，必須多設定一個 blowfish 加密的字串：

```
$cfg['blowfish_secret'] = 'just_keyin_something';  
$cfg['Servers'][$i]['auth_type'] = 'cookie';
```

## ➤ 11-3 從其它電腦操作 phpMyAdmin

- ◎ 如果 XAMPP（包含 MySQL 和 phpMyAdmin）安裝在 A 電腦主機，要從網際網路上 B 電腦拜訪 A 電腦主機的 phpMyAdmin，需要注意以下幾點：
  1. A 電腦的防火牆是否有開放 Apache HTTP 的服務。避免發生 Apache 有啟動，但是外面的訪問被防火牆擋住進不去。注意，防毒軟體的防火牆。
  2. 設定檔 D:\xampp\apache\conf\extra\httpd-xampp.conf（被 Apache httpd.conf 包含）是否有修改為「Allow from all」或允許外部某台電腦拜訪，並取消 Indexes（自動顯示目錄內容）。
  3. phpMyAdmin 對 MySQL 的認證方式使用 http 或 cookie 的方式。

若要改成某些 IP 可拜訪，D:\xampp\apache\conf\extra\httpd-xampp.conf 須修改：

```
# Close XAMPP sites here  
<LocationMatch "^/(?:i:(?:xampp|licenses|phpmyadmin|webalizer|server-status|server-info))">  
    Order deny,allow  
    Deny from all  
    Allow from ::1 127.0.0.0/8 192.168.0.100  
    ErrorDocument 403 /error/HTTP_XAMPP_FORBIDDEN.html.var  
</LocationMatch>
```



## ➤ 12 MySQL 資料備份

◎ 在此介紹兩種方便備份的方式：用 phpMyAdmin 的輸出，及直接複製相關檔案。

### ➤ 12-1 使用 phpMyAdmin 做資料備份

◎ 以 phpMyAdmin 的輸出做備份時，依範圍大小分成以「資料庫」為單位，或者以「資料表」為單位。

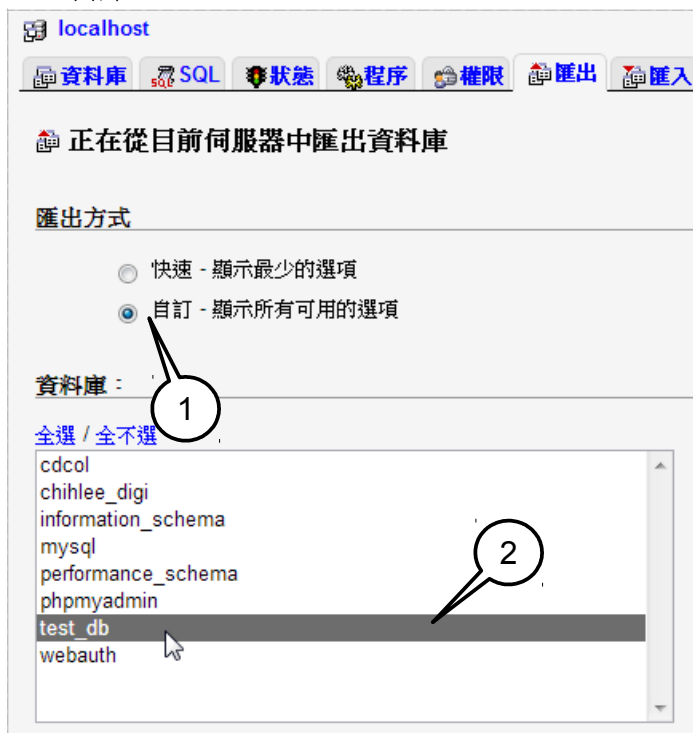
◎ 以「資料庫」為單位備份時，先進入主目錄，再點選「輸出」選項。



① 點選「首頁」（主目錄）。

② 點按「匯出」分頁。

◎ 以「資料庫」為單位備份時，快速選項可匯出所有資料庫。自訂選項可選擇某些資料庫。

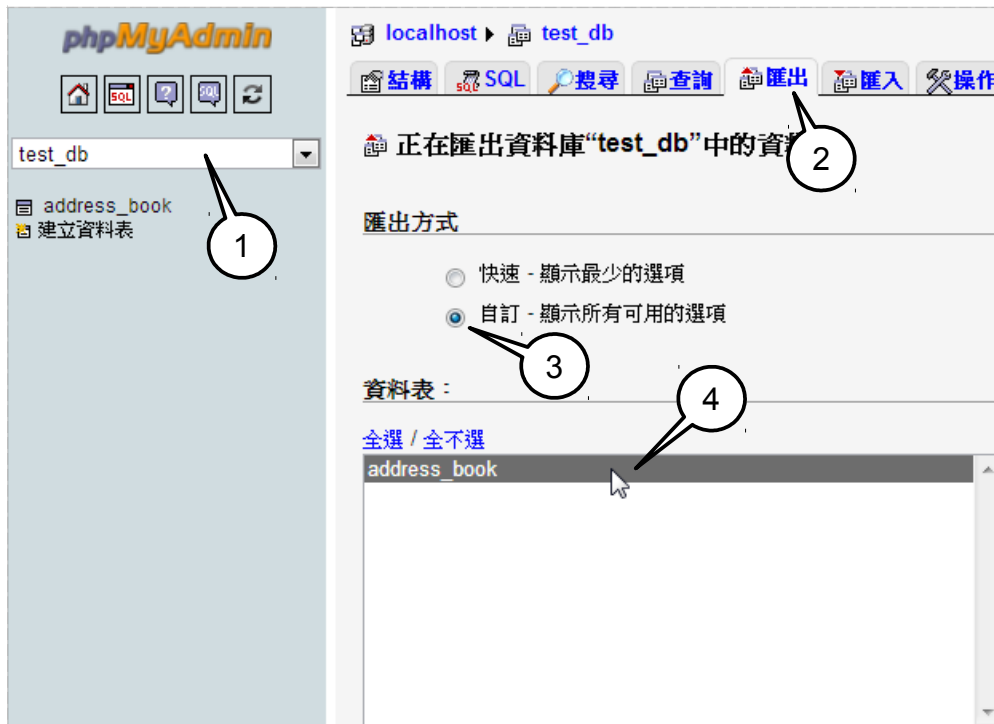


① 點選「自訂」。

② 選取要匯出的資料庫。

③ 點按「執行」按鈕匯出（其餘項目皆不須變更）。

- ◎ 以「資料表」為單位備份時，先點選左側資料庫，再點選「輸出」選項。接著在「檢視資料庫的備份概要」的左側就可以選擇要輸出的資料表。



- ① 點選欲匯出的資料庫。
- ② 點選「匯出」分頁。
- ③ 點選「自訂」。
- ④ 選取要匯出的資料表。
- ⑤ 點按「執行」按鈕匯出。

- ◎ 單一「資料表」備份，作法類似。下圖是顯示為文字的方式。



- ① 點選欲匯出的資料表。

- ② 點選「匯出」分頁。
- ③ 點選「自訂」。
- ④ 點選「直接顯示為文字」。
- ⑤ 點按「執行」按鈕匯出到頁面上。

- ◎ 以下方框內容為匯出的 SQL 內容，很明顯的備份的方式是輸出 SQL 敘述（其中一為註解），使用的語法不外乎「CREATE」和「INSERT」。注意，在預設的情況，建立資料表時會先判斷是否已存在同名稱的資料表，若不存在才建立。

```
-- phpMyAdmin SQL Dump
-- version 3.4.5
-- http://www.phpmyadmin.net
--
-- 主機: localhost
-- 產生日期: 2012 年 12 月 04 日 14:38
-- 伺服器版本: 5.5.16
-- PHP 版本: 5.3.8

SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

--
-- 資料庫: `test_db`
--

-----

--
-- 表的結構 `address_book`
--

CREATE TABLE IF NOT EXISTS `address_book` (
  `sid` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(255) NOT NULL,
  `email` varchar(255) NOT NULL,
  `mobile` varchar(255) NOT NULL,
  `address` varchar(255) NOT NULL,
  PRIMARY KEY (`sid`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=9 ;

--
-- 轉存資料表中的資料 `address_book`
--

INSERT INTO `address_book` (`sid`, `name`, `email`, `mobile`, `address`) VALUES
(1, '林新德', 'shinder.lin@gmail.com', '0918000000', '新北市汐止區'),
(2, '林心得 1', 'shinder.lin@gmail.com', '0918000001', '新北市汐止區 1'),
(3, '林心得 2', 'shinder.lin@gmail.com', '0918000002', '新北市汐止區 2'),
(4, '林心得 3', 'shinder.lin@gmail.com', '0918000003', '新北市汐止區 3'),
(5, '林心得 4', 'shinder.lin@gmail.com', '0918000004', '新北市汐止區 4'),
(6, '林心得 5', 'shinder.lin@gmail.com', '0918000005', '新北市汐止區 5'),
(7, '林心得 6', 'shinder.lin@gmail.com', '0918000006', '新北市汐止區 6'),
(8, '林心得 7', 'shinder.lin@gmail.com', '0918000007', '新北市汐止區 7');
```

- ◎ 除了以 SQL 格式做輸出，也可以使用其它格式。不過其它格式大多是為了將資料轉移到不同的軟體（如 Excel）。以資料庫備份的角度應該使用 SQL 格式。

## ➤ 12-2 使用 phpMyAdmin 做資料還原

- ◎ 在 phpMyAdmin 做匯入時，同樣依範圍大小分成以「資料庫」為單位，或者以「資料表」為單位。
- ◎ 以「資料庫」為單位匯入時，先進入主目錄，再點選「匯入」選項。
- ◎ 以「資料表」為單位匯入時，先點選左側資料庫，再點選「匯入」選項。



- ① 選取標的資料庫。
- ② 點選「匯入」分頁。
- ③ 點按「選擇檔案」，選擇所要匯入的 SQL 檔。
- ④ 點按「執行」按鈕匯入。

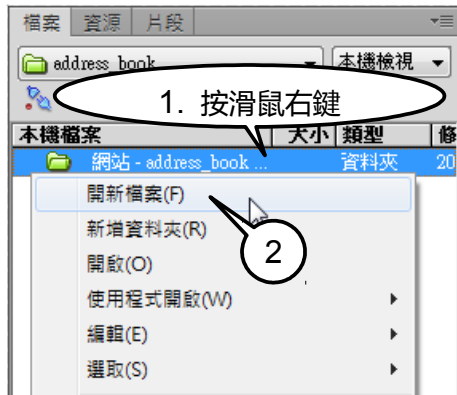
- ◎ 載入的 SQL 檔應該小於 2M（或 8M）。如果大於 2M 時，可以用文字編輯器將檔案分割成幾個小檔案；或者調整伺服器 PHP 可上傳的檔案大小。若資料備份時檔案遠大於 2M 時，可以使用 BigDump（<http://www.ozeroov.de/bigdump/>）。

## ➤ 12-3 複製檔案

- ◎ 如果伺服器主機是我們有權限可以操作檔案的，那就好辦了直接複製檔案即可。複製資料時應該先將 MySQL 的服務關閉。若要所有資料庫都備份，請把整個「D:\xampp\mysql\data」目錄複製起來。
- ◎ 若資料庫裡的資料表皆使用 MyISAM 引擎，可以將單一個資料庫做備份，或者單一個資料表做備份。資料庫會以目錄為單位，而 MyISAM 的資料表都會以資料表名稱為檔名，包含三個檔案，副檔名分別為：frm、MYD 和 MYI。複製和貼上檔案是相當簡單的工作，操作前請務必記得檢查資料的新舊，以免發生舊資料庫（表）取代新資料庫（表）的悲劇。

## ➤ 13 建立資料庫連線

- ◎ 建立目錄「D:\xampp\htdocs\address\_book」，接著如「2 以 Dreamweaver 建立專案」介紹的方式建立新網站，名稱設定為「address\_book」。在網站上開新檔案，並開啟該 PHP 檔。

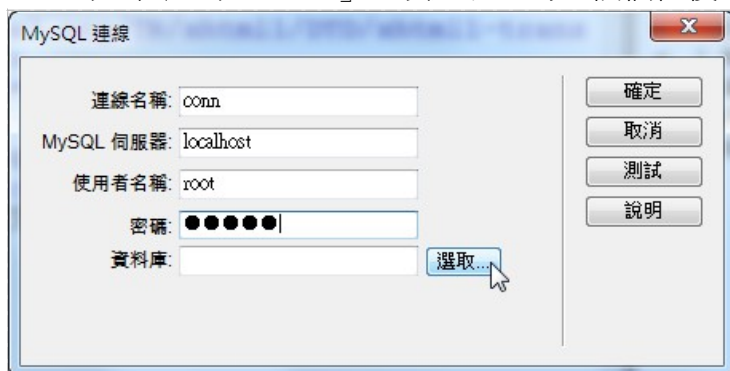


- ◎ 打開「資料庫」面板，點選「視窗」>「資料庫」開啟。點選加號按鈕「+」後再點選「MySQL 連線」。

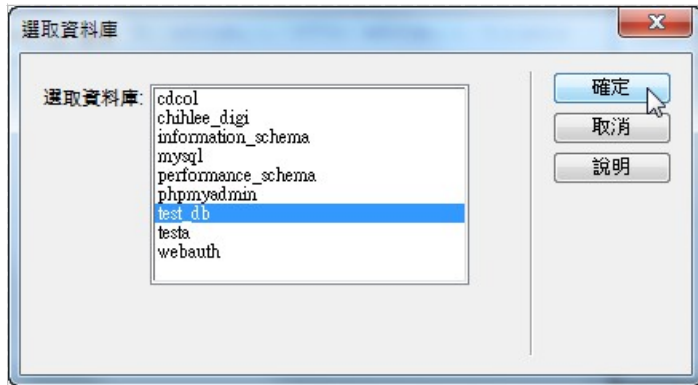


- ① 點選加號按鈕「+」。
- ② 點選「MySQL 連線」。

- ◎ 接著出現「MySQL 連線」面板。「連線名稱」欄位填入自訂的名稱，請注意在此不要使用中文和空白，這個名稱到時候會變成變數名稱。「MySQL 伺服器」欄位填入「localhost」；「使用者名稱」欄位填入「root」；「密碼」欄位填入之前設定的密碼，在此為「admin」。填入以上的四個欄位後，點按「資料庫」欄後的「選取」。



- ◎ 於開啟「選取資料庫」面板，選取「test\_db」資料庫後，按「確定」。

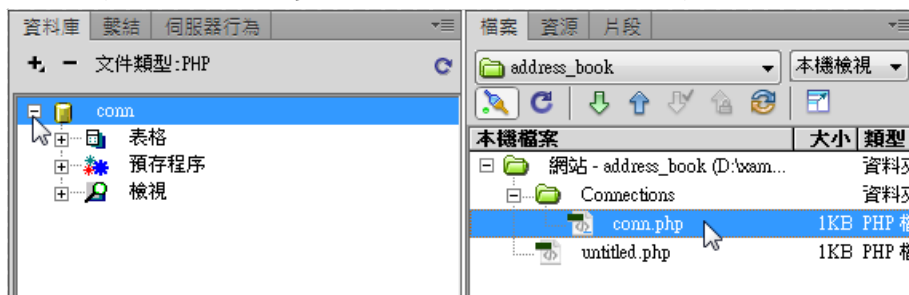


- ◎ 按「MySQL 連線」面板上的「確定」完成連線設定。



- ◎ 開啟「選取資料庫」面板時若發生錯誤，請先檢查 MySQL 服務是否正常啟動，接著再檢查密碼是否正確。

- ◎ 連線設定完成後，看一下 Dreamweaver 的「資料庫」及「檔案」面板。「資料庫」面板內多了一個 conn 連線圖示，由該圖示可樹狀下拉查看表格的結構。由「檔案」面板，可以看到多了一個 Connections 目錄，裡面有一個 conn.php 檔。



- ◎ 每個連線都針對一個資料庫，若要使用多個資料庫就必須建立多個連線。一般來講，一個網站只使用一個資料庫，以方便管理。複雜的網站或許需要多個資料庫連線。

- ◎ conn.php 內容：

```
<?php
# FileName="Connection_php_mysql.htm"
# Type="MYSQL"
# HTTP="true"
```

```

$hostname_conn = "localhost";
$database_conn = "test_db";
$username_conn = "root";
$password_conn = "admin";
$conn = mysql_pconnect($hostname_conn, $username_conn, $password_conn)
or
trigger_error(mysql_error(),E_USER_ERROR);
?>

```

- ◎ # 也是 PHP 的單行註解符號。由變數名稱可知，這些東西都是在設定連線時給的內容。比較特別的是使用 mysql\_pconnect() 函式建立連線，並設定連線的控制編號給 \$conn 變數。如果無法建立資料庫連線（mysql\_pconnect 函式結果為 false），則執行 trigger\_error 函式（將錯誤訊息秀出來）。
- ◎ or 是個快捷運算子。以「A or B」為例說明（A 和 B 是運算式或者函式呼叫），當 A 的結果為 true 時，B 結果不論是 true 或 false，「A or B」的結果都是 true，為了效率 B 就不會被執行。所以，當 A 的結果為 false 時，B 才會被執行。
- ◎ 請在 conn.php 最後（建立連線之後）加入一行程式，以避免亂碼問題。

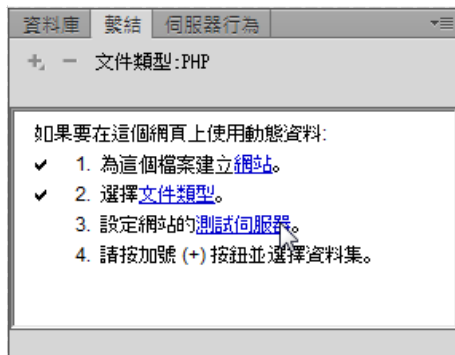
```
mysql_query("SET NAMES utf8");
```

## ➤ 14 顯示資料及分頁

- ◎ 上一節講建立資料庫連線，接著應該是「建立新增資料的表單」並新增資料。之前在學習 phpMyAdmin 時，已經在資料表中新增了一些資料，我們先來看如何透過 PHP 顯示資料。

### ➤ 14-1 建立資料集繫結

- ◎ 如果我們尚未測試伺服器，「繫結」和「伺服器行為」面板的新增「+」可能無法使用，此時測試一下伺服器，在「繫結」或「伺服器行為」面板點按「測試伺服器」。如果設定沒有問題點按「測試伺服器」面板的「確定」即可。



- ◎ 倘若「+」還無法使用，關閉並重啟 Dreamweaver。
- ◎ 我們需要一個網頁來顯示資料，因此先開新檔案，並命名為 show\_all.php。在工作區

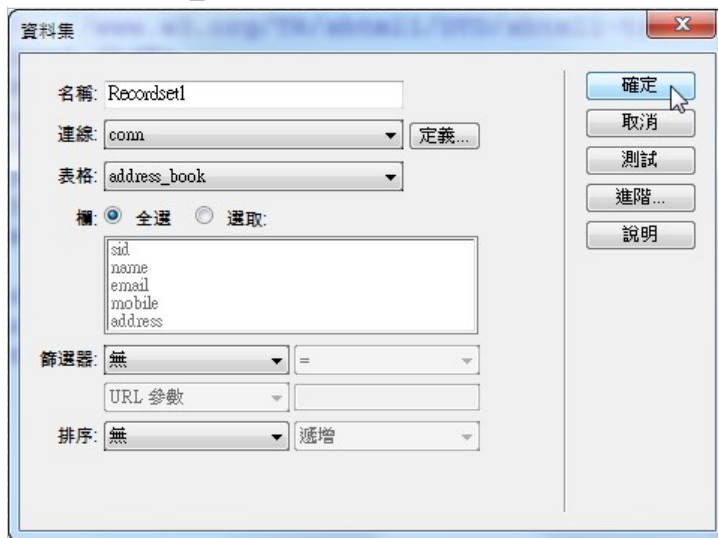
開啟 show\_all.php，這個動作很簡單卻很重要，因為後續的動作都會加諸於當下工作區開啟的檔案。

◎ 點擊「繫結」面板裡的加號「+」，接著在跳出的選單上點選「資料集(查詢)」。



- ① 開啟新建的 show\_all.php。
- ② 點擊「繫結」面板裡的加號「+」。
- ③ 點選「資料集(查詢)」。

◎ 然後出現「資料集」面板。資料集名稱可以自訂，它會變成變數名，所以取名時請遵守變數命名規則，在此我們用預設的「Recordset1」。連線用「conn」，表格用「address\_book」，按「確定」完成。



◎ 完成新增資料集後，可以在「繫結」面板看到「資料集(Recordset1)」圖示。





◎ 此時，Dreamweaver 幫我們加入一些程式，最前面的部份：

```
<?php require_once('Connections/conn.php'); ?>
<?php
if (!function_exists("GetSQLValueString")) {
function GetSQLValueString($theValue, $theType, $theDefinedValue = "",
$theNotDefinedValue = "")
{
    if (PHP_VERSION < 6) {
        $theValue = get_magic_quotes_gpc() ? stripslashes($theValue) : $theValue;
    }

    $theValue = function_exists("mysql_real_escape_string") ?
mysql_real_escape_string($theValue) : mysql_escape_string($theValue);

    switch ($theType) {
        case "text":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "long":
        case "int":
            $theValue = ($theValue != "") ? intval($theValue) : "NULL";
            break;
        case "double":
            $theValue = ($theValue != "") ? doubleval($theValue) : "NULL";
            break;
        case "date":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "defined":
            $theValue = ($theValue != "") ? $theDefinedValue : $theNotDefinedValue;
            break;
    }
    return $theValue;
}

mysql_select_db($database_conn, $conn);
$query_Recordset1 = "SELECT * FROM address_book";
$Recordset1 = mysql_query($query_Recordset1, $conn) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);
$totalRows_Recordset1 = mysql_num_rows($Recordset1);
?>
```

- ☆ require\_once() 只包含（引入）一次，若之前有包含過該檔案，則不再包含。
- ☆ GetSQLValueString() 的功能在於防止資料隱碼（SQL injection）攻擊。
- ☆ mysql\_select\_db() 選擇使用的資料庫。
- ☆ mysql\_query() 執行 SQL 命令。
- ☆ die() 中斷程式並顯示訊息。
- ☆ mysql\_fetch\_assoc() 從資料集中取出一筆資料，以關聯式陣列回傳。
- ☆ mysql\_num\_rows() 取得資料集中的資料筆數。
- ☆ mysql\_free\_result() 釋出資料集所佔用的資源。

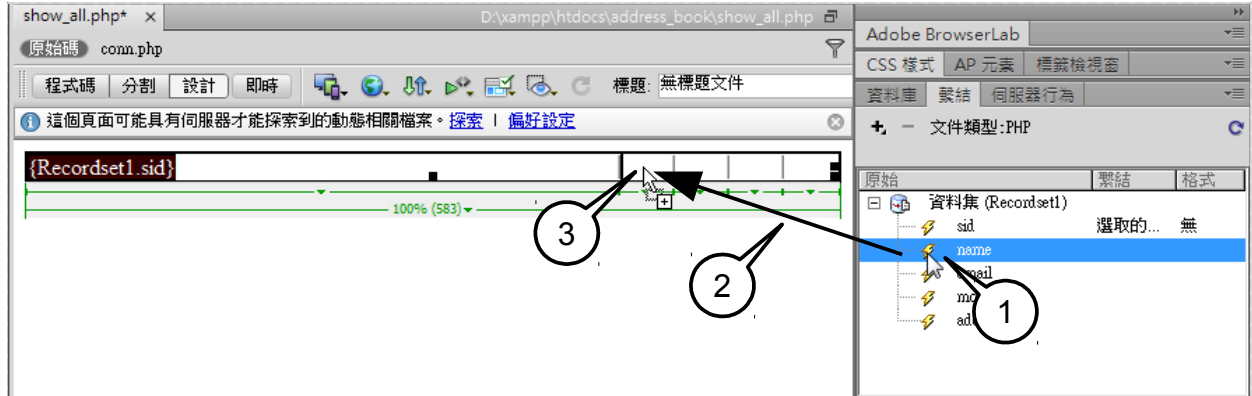
## ➤ 14-2 顯示一筆資料

◎ 在 show\_all.php 上製作 1 列 5 欄的表格。



- ① 頁面切換為「設計」模式。
- ② 點擊「表格」圖示，建立表格。
- ③ 設定列數及欄數。
- ④ 點擊「確定」。

◎ 在「繫結」面板裡，資料集下會有五個項目（分別是資料表的欄位名）。在 sid 上按滑鼠左鍵拖曳至表格的第一格後放開，如此就建立資料欄位的繫結。依序分別將不同欄位插入至不同的格子內。



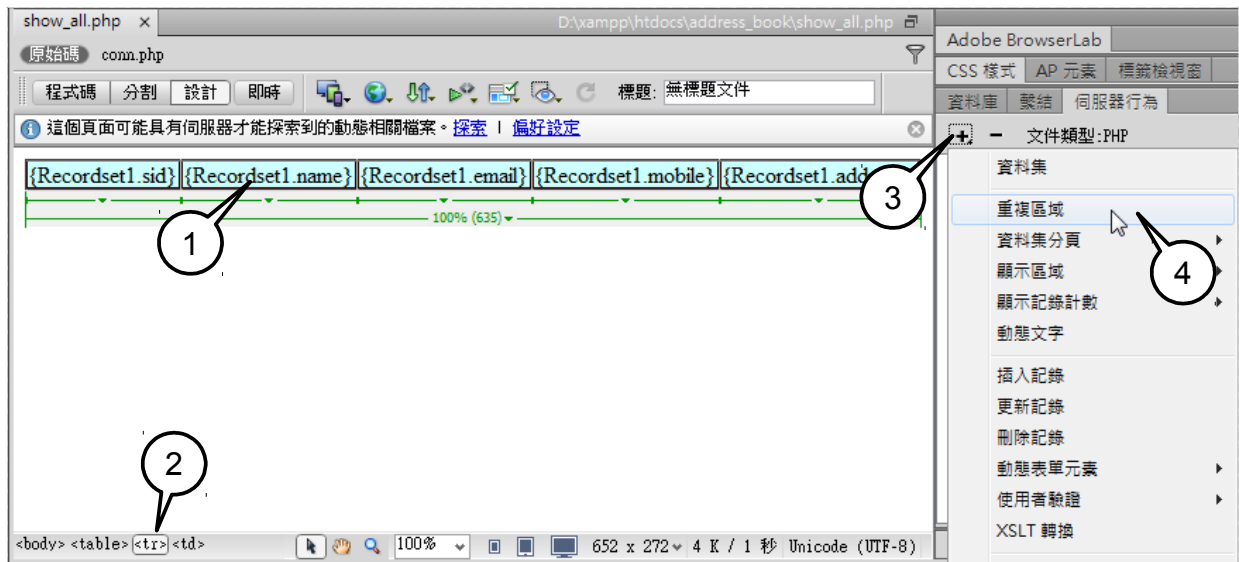
- ① 選定項目按住滑鼠左鍵。
- ② 拖曳。
- ③ 在適當位置放開滑鼠。

◎ 存檔後測試，應該看到一筆資料。若中文字無法顯示，可能未在 conn.php 中設定校對編碼為 utf8。

```
mysql_query("SET NAMES utf8");
```

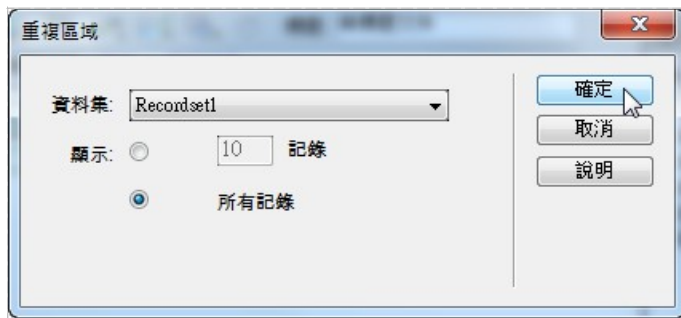
## ➤ 14-3 顯示所有資料

◎ 可以顯示一筆資料之後，我們將以「伺服器行為」的「重複區域」顯示所有資料。show\_all.php 用滑鼠點選表格中的某一格，點選<tr>元素（表格整列）。然後，在「伺服器行為」面板上點按加號「+」，於快顯選單上點選「重複區域」。



- ① 點選表格中的某一格。
- ② 點選 <tr> 元素。
- ③ 「伺服器行為」面板上點按加號「+」。
- ④ 點選「重複區域」。

◎ 接著在跳出的「重複區域」面板上，點選「所有記錄」，再點按「確定」。



◎ 完成操作後，存檔並按「F12」功能鍵於瀏覽器中預覽，可以看到資料表中的每一筆資料。

◎ 以下是從顯示一筆資料開始所新增的程式碼：

```
<table width="100%" border="1" cellspacing="0" cellpadding="0">
  <?php do { ?>
    <tr>
      <td><?php echo $row_Recordset1['sid']; ?></td>
      <td><?php echo $row_Recordset1['name']; ?></td>
      <td><?php echo $row_Recordset1['email']; ?></td>
      <td><?php echo $row_Recordset1['mobile']; ?></td>
      <td><?php echo $row_Recordset1['address']; ?></td>
    </tr>
  <?php } while ($row_Recordset1 = mysql_fetch_assoc($Recordset1)); ?>
</table>
```

☆這邊使用 do/while 迴圈，主要原因是資料集一開始就會做一次「\$row\_Recordset1 = mysql\_fetch\_assoc(\$Recordset1)」從資料集取出一筆資料，所以還沒有使用迴圈，

就可以顯示單一筆資料。因此，才使用判斷式放於末端的 do/while 迴圈。

☆ 「`$row_Recordset1 = mysql_fetch_assoc($Recordset1)`」為迴圈的條件式，當資料集未被讀完時，`$row_Recordset1` 會得到一陣列（視為 true）；若資料集被讀完了，`$row_Recordset1` 會得到 NULL（視為 false）。

◎ 除了 `mysql_fetch_assoc()` 函式，常用來讀取一筆資料的還有 `mysql_fetch_row()` 和 `mysql_fetch_array()` 函式。`mysql_fetch_row()` 函式會將一筆資料轉換成索引式陣列的格式，這個時候就得清楚欄位前後的順序。`mysql_fetch_array()` 函式回傳同時包含索引式及關聯式的陣列，意即可以用索引式也可以用關聯式。

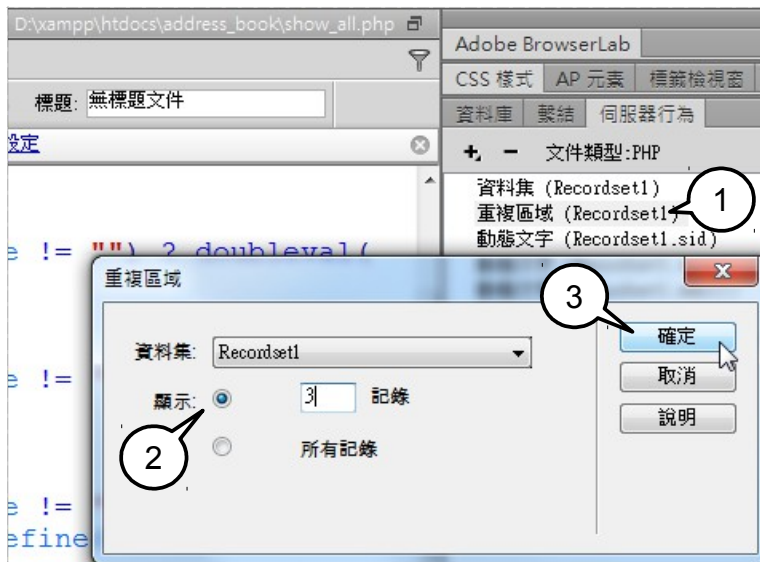
◎ 資料集內部包含一指標變數（the internal row pointer），該指標變數一開始為 0，表示指向第一筆資料。在使用 `mysql_fetch_assoc()` 等函式去由資料集讀取一筆資料時，會將指標變數所指向的該筆資料讀出，並指向下一筆。使用 `mysql_data_seek()` 函式可以設定指標變數的值。例如

「`mysql_data_seek($Recordset1, 0)`」可以讓資料集回恢到未讀取的狀態。

## ➤ 14-4 分頁顯示

◎ 上一小節是將所有資料表內的資料顯示出來。如果資料有幾百筆，顯示在一個頁面並不是好主意，所以我們要將資料做分頁顯示。

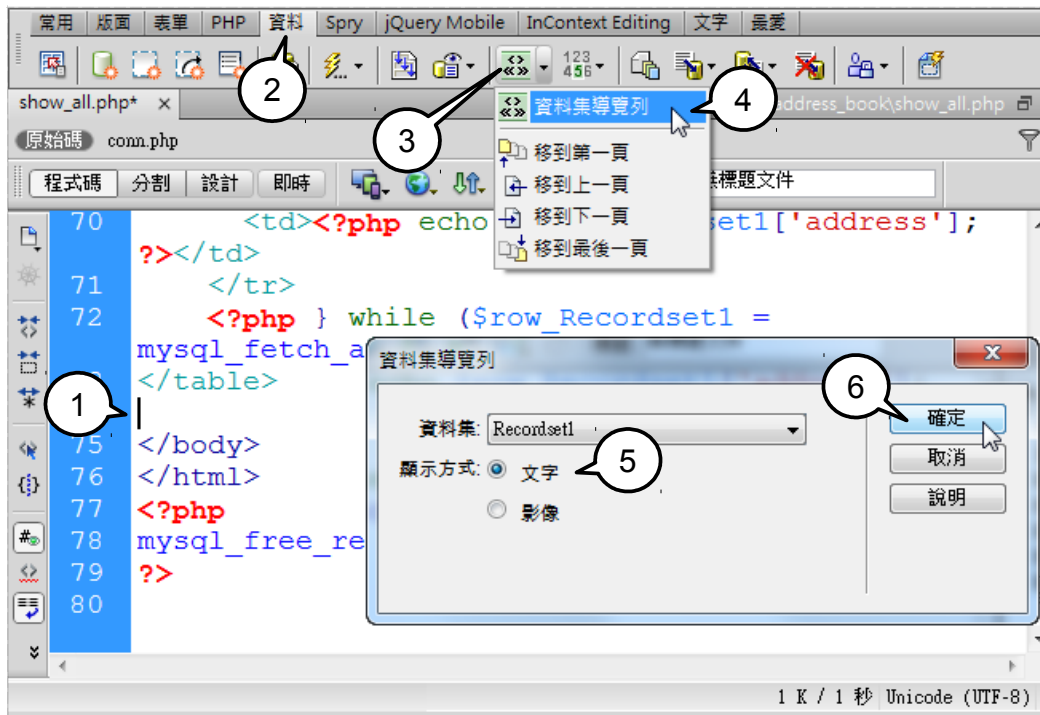
◎ 繼續上一小節的例子，於「伺服器行為」的「重複區域」項點按兩下，以編輯「重複區域」面板，修改為「顯示 3 記錄」，並按「確定」。在此只是做分頁的測試，所以將每頁要顯示的筆數改為較小的數量，資料的總數應該要超出每頁的顯示數。



- ① 雙擊「重複區域」。
- ② 修改為「顯示 3 記錄」。
- ③ 點按「確定」。

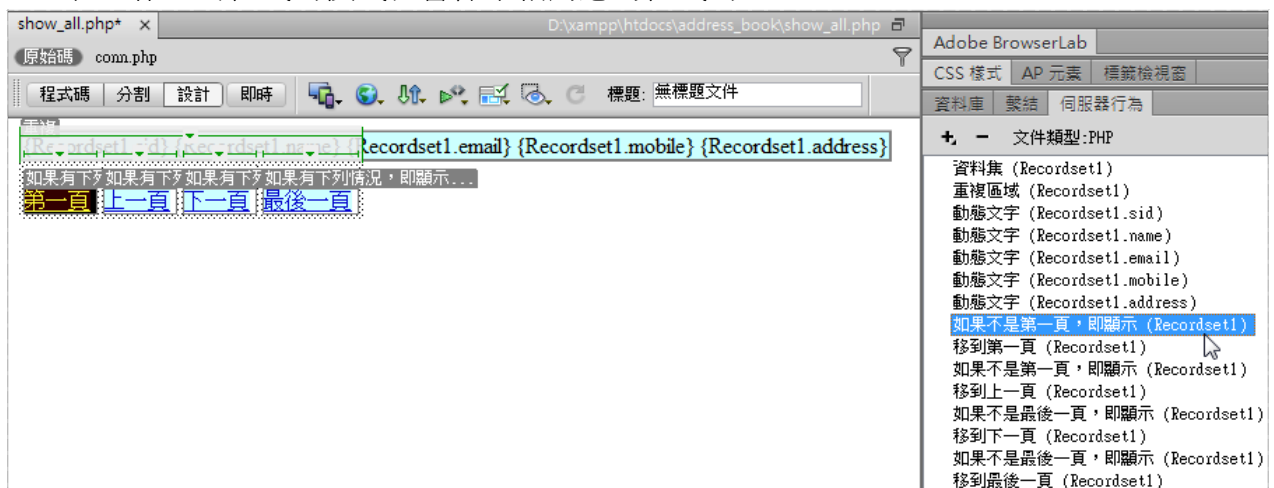
◎ 存檔後，在瀏覽器中預覽可以看到只顯示 3 筆資料。接著我們要加入「導覽列」（換頁的連結），Dreamweaver 在此可以用插入工具快速完成。工作區切到程式碼模

式，鍵盤游標放在 </table> 和 </body> 之間。點選「插入」面板的「資料」頁，再點選「資料集導覽列」圖示（或於快顯選單點選「資料集導覽列」）。



- ① 用滑鼠點擊，讓鍵盤游標放在 </table> 和 </body> 之間。
- ② 點選「插入」面板的「資料」頁。
- ③ 點選「資料集導覽列」圖示。
- ④ 點選「資料集導覽列」。
- ⑤ 於「資料集導覽列」面板，設定顯示方式為「文字」。
- ⑥ 點按「確定」。

◎ 完成後，在「伺服器行為」面板裡，多了一些和分頁相關的項目。點選這些項目，在工作區（程式碼模式）會標示相對應的程式碼。



◎ 在瀏覽器中預覽，請注意網址列多了 GET 參數「pageNum\_Recordset1」和「totalRows\_Recordset1」。要顯示哪一頁是由「pageNum\_Recordset1」決定，顯示第一頁時，其值為 0。「totalRows\_Recordset1」是資料表的總筆數。

4	林心得3	shinder.lin@gmail.com	0918000003
5	林心得4	shinder.lin@gmail.com	0918000004
6	林心得5	shinder.lin@gmail.com	0918000005

第一頁 上一頁 下一頁 最後一頁

◎ 分頁相關的程式碼：

```

$maxRows_Recordset1 = 3;
$pageNum_Recordset1 = 0;
if (isset($_GET['pageNum_Recordset1'])) {
    $pageNum_Recordset1 = $_GET['pageNum_Recordset1'];
}
$startRow_Recordset1 = $pageNum_Recordset1 * $maxRows_Recordset1;

mysql_select_db($database_conn, $conn);
$query_Recordset1 = "SELECT * FROM address_book";
$query_limit_Recordset1 = sprintf("%s LIMIT %d, %d",
    $query_Recordset1, $startRow_Recordset1, $maxRows_Recordset1);
$Recordset1 = mysql_query($query_limit_Recordset1, $conn) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

if (isset($_GET['totalRows_Recordset1'])) {
    $totalRows_Recordset1 = $_GET['totalRows_Recordset1'];
} else {
    $all_Recordset1 = mysql_query($query_Recordset1);
    $totalRows_Recordset1 = mysql_num_rows($all_Recordset1);
}
$totalPages_Recordset1 = ceil($totalRows_Recordset1/$maxRows_Recordset1)-1;

```

- ☆ \$maxRows\_Recordset1 為每頁顯示的資料筆數（最大筆數）。
- ☆ \$pageNum\_Recordset1 為顯示第幾頁，預設為 0（第一頁）。
- ☆ 若有設定 \$\_GET['pageNum\_Recordset1']，則以 GET 參數為主。
- ☆ 計算要由哪一筆開始顯示，設定於變數 \$startRow\_Recordset1。
- ☆ sprintf() 函式的功能是格式化文字。第一個參數是包含置換符號「%s」（字串置換符號）或「%d」（數字置換符號）的字串。sprintf() 函式的第二個參數會放到第一個置換符號的位置，第三個參數會放到第二個置換符號的位置，依此類推。
- ☆ LIMIT 為 MySQL 的 SQL 語法，「... LIMIT 2, 5」表示從第三筆開始（包含第三筆）取得五筆資料。
- ☆ 如果有 totalRows\_Recordset1 的 GET 參數，則以該值為 \$totalRows\_Recordset1 的值。否則，計算出資料的總筆數設定給 \$totalRows\_Recordset1。
- ☆ 計算頁數，並設定給變數 \$totalPages\_Recordset1。

◎ 以下的變數或函式的功能為何？

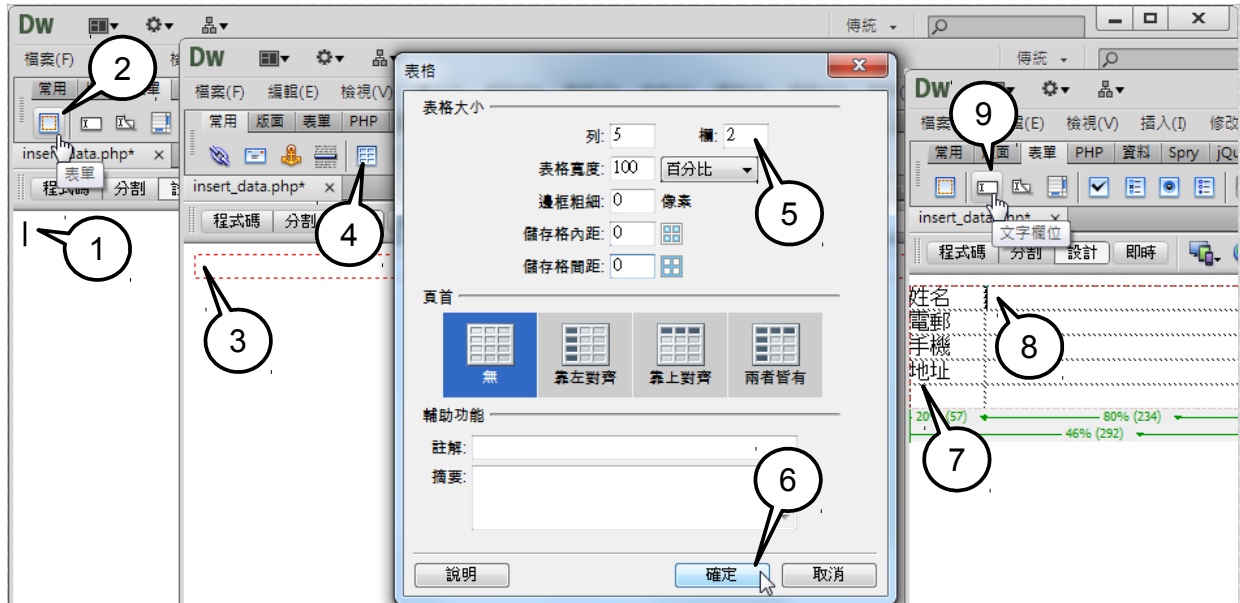
```

$_SERVER['QUERY_STRING']
empty(), explode(), htmlentities(), implode()

```

## ➤ 15 建立新增資料的表單

- ◎ 新建檔案 insert\_data.php，並在裡面建立一個表單，包含我們要使用者輸入的四個欄位：姓名、電郵、手機及地址。



- ① 在設計模式，點選要放表單的位置。
- ② 點選「表單」圖示建立表單。
- ③ 滑鼠點擊紅色虛線包圍起來的區域，將鍵盤游標放至表單內。
- ④ 點選「表格」圖示建立表格。
- ⑤ 設定為5列2欄。
- ⑥ 點按「確定」完成表格建立。
- ⑦ 在表格第一欄填入文字。
- ⑧ 點選表格第二欄第一列。
- ⑨ 點按「文字欄位」，在表單內建立輸入文字欄。

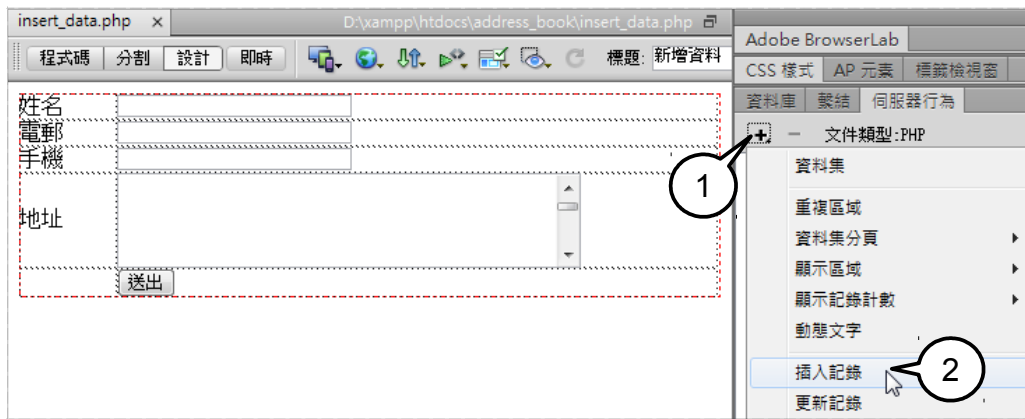
- ◎ 表單內的輸入欄位名稱，對應到資料表的欄位名稱。

姓名	<input type="text"/>
電郵	<input type="text"/>
手機	<input type="text"/>
地址	<input type="text"/>
<input type="button" value="送出"/>	

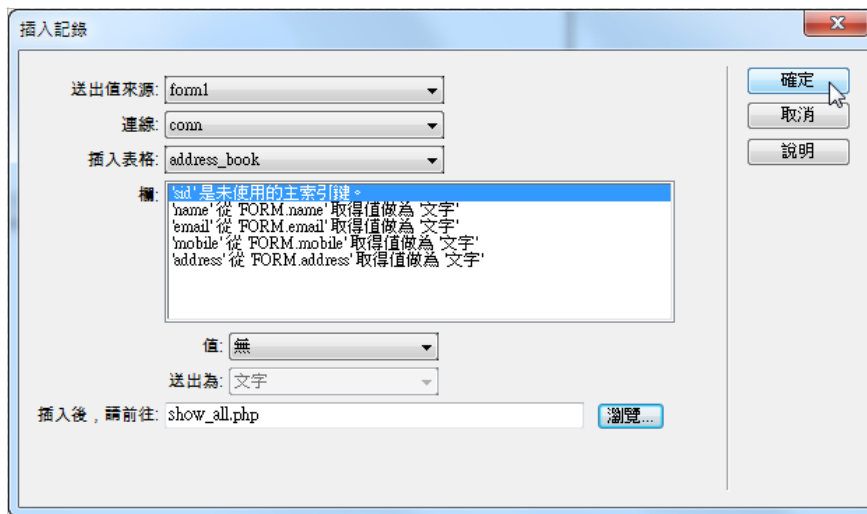
```
<form id="form1" name="form1" method="post" action="">
  <table width="500" border="0" cellspacing="0" cellpadding="0">
    <tr>
      <td width="70">姓名</td>
      <td><input type="text" name="name" id="name" /></td>
    </tr>
    <tr>
      <td>電郵</td>
      <td><input type="text" name="email" id="email" /></td>
    </tr>
  </table>
```

```
<tr>
  <td>手機</td>
  <td><input type="text" name="mobile" id="mobile" /></td>
</tr>
<tr>
  <td>地址</td>
  <td><textarea name="address" id="address" cols="45" rows="5">
    </textarea></td>
</tr>
<tr>
  <td>&nbsp;</td>
  <td><input type="submit" name="button" id="button" value="送出" /></td>
</tr>
</table>
</form>
```

◎ 在「伺服器行為」面板點按「+」，並在快顯選單中點選「插入記錄」，開啟「插入記錄」面板。



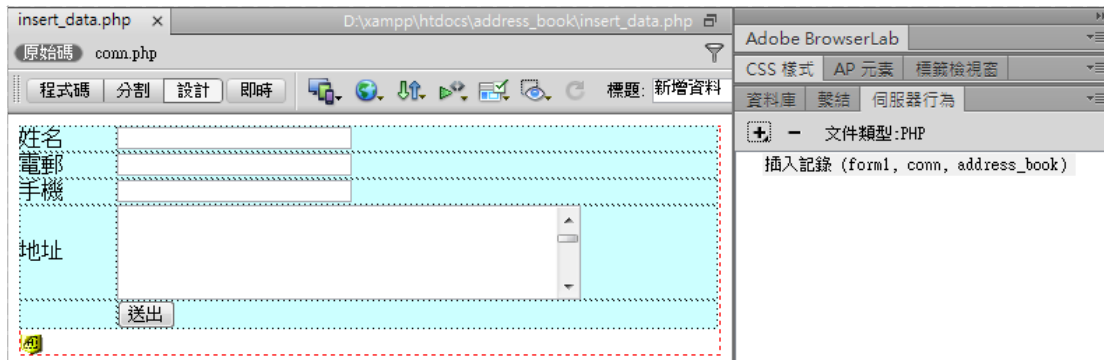
◎ 在「插入記錄」面板，「送出值來源」為送出值的表單，在 insert\_data.php 只有 form1 這個表單。「連線」欄，選擇所要用的連線，在此也只有一個 conn。「插入表格」欄，指的是資料要新增到資料庫裡的哪一張表，這裡使用 address\_book。



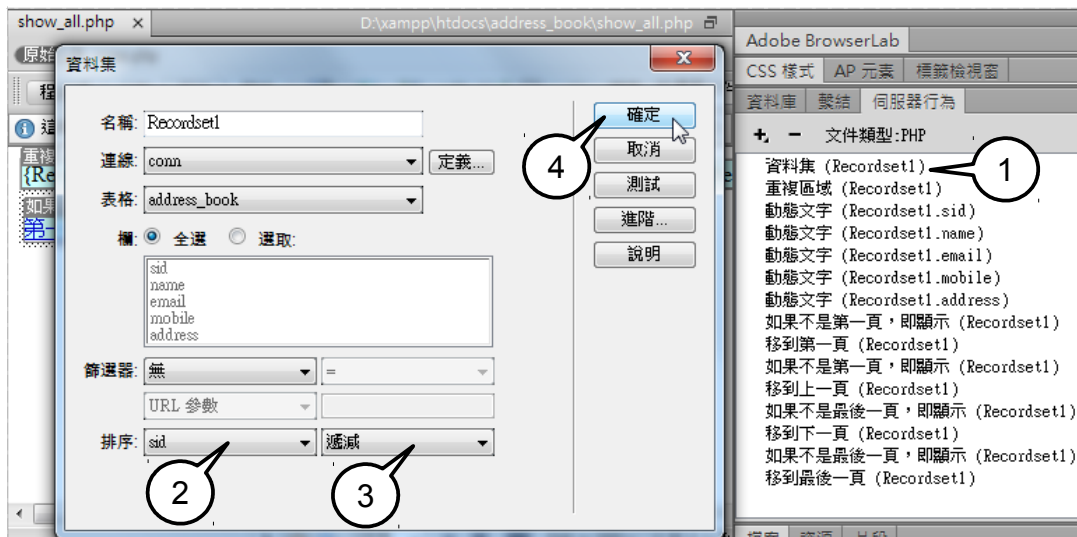
◎ 「欄」欄裡面會列出資料表所有的欄位名稱，以及它們所對應的表單項目。表單項目名稱和資料表的欄位名稱相同時，Dreamweaver 會自動做配對的動作。



- ◎ 若有項目沒有配對，或者配對錯誤，可以點選「欄」裡的項目，再到「值」挑選欲對應的表單項目。「送出為」那一欄可以設定送出的是文字、整數、浮點數等。
- ◎ 表單項目和資料表欄位的對應設定好了之後，點選「插入後，請前往」後的「瀏覽」按鈕。並在「選取重新導向檔案」面板，點選 show\_all.php。意即，完成新增資料後，頁面會導向 show\_all.php，讓使用者可以查看到新增的資料。
- ◎ 點按「插入記錄」面板的「確定」後完成設定，存檔並預覽網頁。



- ◎ 在表單中輸入資料，按「送出」後會導向 show\_all.php，到最後一頁可以看到新增的一筆資料。
- ◎ 可以更改 show\_all.php 讓最新的資料放在第一頁。在「伺服器行為」面板，雙擊「資料集(Recordset1)」項目，並在秀出的「資料集」面板，設定「排序」欄為「sid」、「遞減」，讓最後新增的資料顯示在最前面。



- ① 雙擊「資料集(Recordset1)」項目。
- ② 設定「排序」欄為「sid」。
- ③ 設定「排序」方式為「遞減」。
- ④ 點選「確定」。

- ◎ insert\_data.php 程式碼：

```

<?php
# 略 ###
$editFormAction = $_SERVER['PHP_SELF'];
if (isset($_SERVER['QUERY_STRING'])) {
    $editFormAction .= "?" . htmlentities($_SERVER['QUERY_STRING']);
}

if ((isset($_POST["MM_insert"])) && ($_POST["MM_insert"] == "form1")) {
    $insertSQL = sprintf("INSERT INTO address_book (name, email, mobile, address)
VALUES (%s, %s, %s, %s)",
        GetSQLValueString($_POST['name'], "text"),
        GetSQLValueString($_POST['email'], "text"),
        GetSQLValueString($_POST['mobile'], "text"),
        GetSQLValueString($_POST['address'], "text"));

    mysql_select_db($database_conn, $conn);
    $Result1 = mysql_query($insertSQL, $conn) or die(mysql_error());

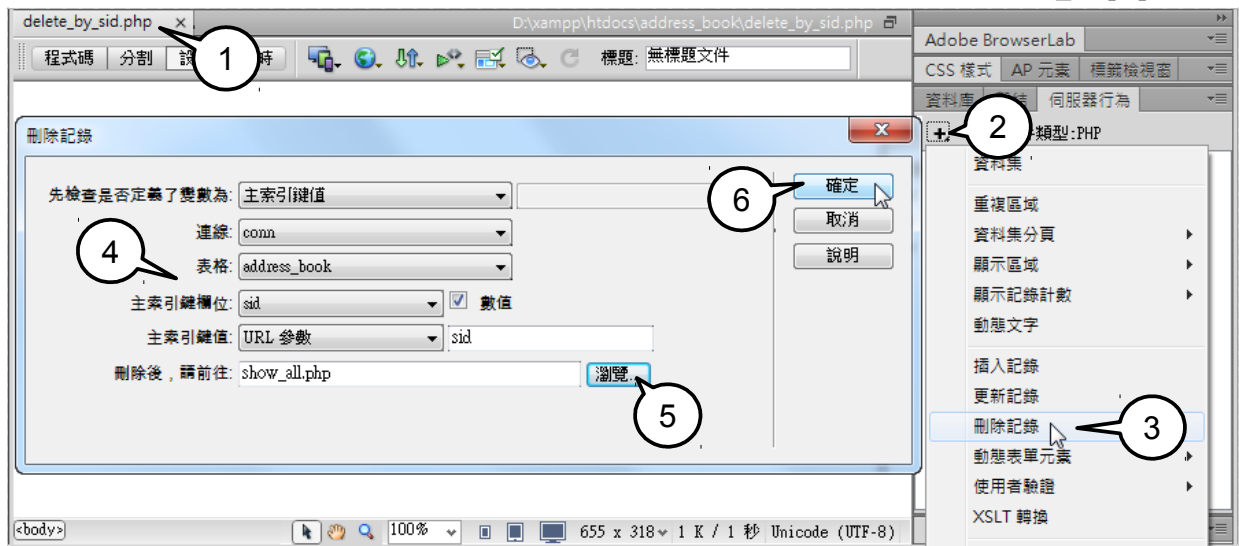
    $insertGoTo = "show_all.php";
    if (isset($_SERVER['QUERY_STRING'])) {
        $insertGoTo .= (strpos($insertGoTo, '?')) ? "&" : "?";
        $insertGoTo .= $_SERVER['QUERY_STRING'];
    }
    header(sprintf("Location: %s", $insertGoTo));
}
?>
<!-- 略 -->
<body>
<form id="form1" name="form1" method="POST"
        action="<?php echo $editFormAction; ?>">
    <table width="500" border="0" cellspacing="0" cellpadding="0">
        <!-- 略 -->
    </table>
    <input type="hidden" name="MM_insert" value="form1" />
</form>
</body>
</html>

```

- ☆ `$_SERVER['PHP_SELF']` 可以得到目前的 PHP 檔名稱（包含路徑），設定給 `$editFormAction`。
- ☆ 若有 GET 變數，經過 HTML 編碼後，接在 `$editFormAction` 之後。  
`$_SERVER['QUERY_STRING']` 可以取得要求字串，就是網址?之後的字串內容。  
`htmlentities()` 函式將字串經由 HTML 編碼，用這個函式的原因是到時候 `$editFormAction` 是放在 HTML 裡。
- ☆ 若有名為「MM\_insert」的 POST 參數，而且其值為「form1」，則將 POST 參數 name、email、mobile 和 address 插入資料表。「MM\_insert」是 Dreamweaver 加入的，為的是判斷是否為使用者輸入表單並送出。
- ☆ 設定導向目標為 show\_all.php。保留原來的 GET 參數。
- ☆ POST 的標的網頁為 `$editFormAction` 的值，其實就是自身網頁。
- ☆ Dreamweaver 加入了隱藏欄位「MM\_insert」，值為表單 ID「form1」。

## ➤ 16 刪除資料

- ◎ 有了新增和瀏覽資料的功能之後，我們要加上刪除的功能。先新建檔案 delete\_by\_sid.php，然後在「伺服器行為」面板點選「+」，於快顯選單上點選「刪除記錄」，接著會顯示「刪除記錄」面板。
- ◎ 在「刪除記錄」面板，「先檢查是否定義了變數為」欄使用「主索引鍵值」，「連線」和「表格」不變。「主索引鍵欄位」使用「sid」並勾選數值；「主索引鍵值」使用「URL 參數」的 sid，其實就是 GET 參數的 sid。刪除後前往「show\_all.php」。



- ① 新建檔案 delete\_by\_sid.php。
- ② 在「伺服器行為」面板點選「+」。
- ③ 點選「刪除記錄」。
- ④ 如圖預設。
- ⑤ 點按「瀏覽」，接著選「show\_all.php」。
- ⑥ 點按「確定」完成。

- ◎ delete\_by\_sid.php 重要內容：

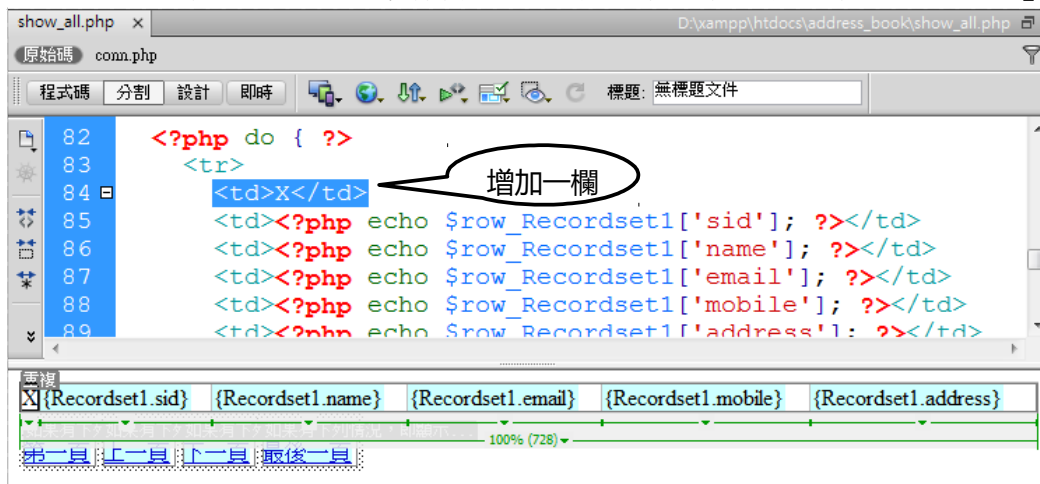
```
<?php
# 略 ###
if ((isset($_GET['sid'])) && ($_GET['sid'] != "")) {
    $deleteSQL = sprintf("DELETE FROM address_book WHERE sid=%s",
        GetSQLValueString($_GET['sid'], "int"));

    mysql_select_db($database_conn, $conn);
    $Result1 = mysql_query($deleteSQL, $conn) or die(mysql_error());

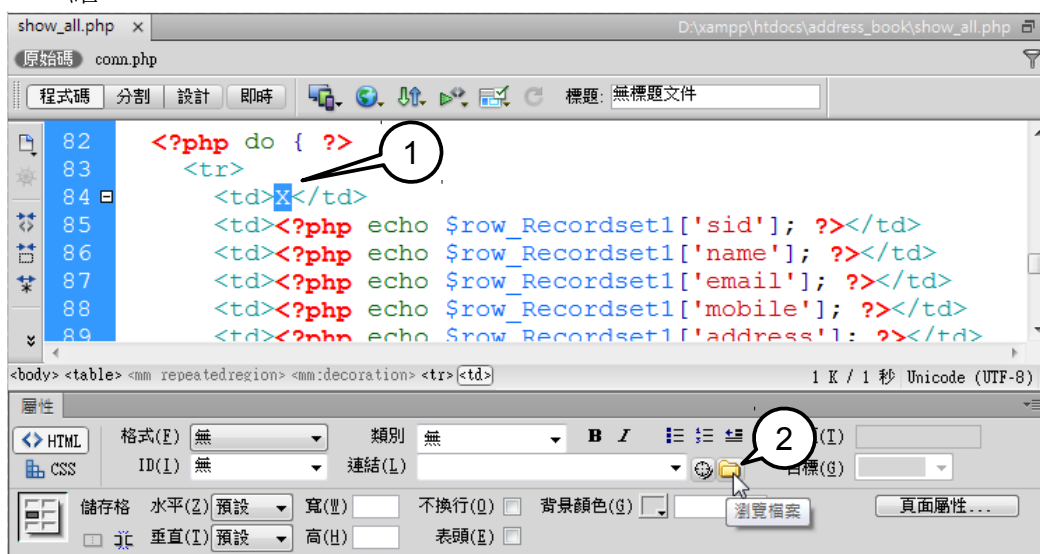
    $deleteGoTo = "show_all.php";
    if (isset($_SERVER['QUERY_STRING'])) {
        $deleteGoTo .= (strpos($deleteGoTo, '?')) ? "&" : "?";
        $deleteGoTo .= $_SERVER['QUERY_STRING'];
    }
    header(sprintf("Location: %s", $deleteGoTo));
}
?>
```

- ☆ 判斷是否有 GET 參數 sid，而且其值不能為空字串。
- ☆ 以 GET 參數 sid 為條件，刪除該筆資料。
- ☆ 網頁導向 show\_all.php。

◎ 包含刪除功能的網頁已經有了，我們還要在 show\_all.php 裡加上，讓使用者可以用點選的方式選擇欲刪除的資料項目。在其表格左側增加一欄，放入「X」。

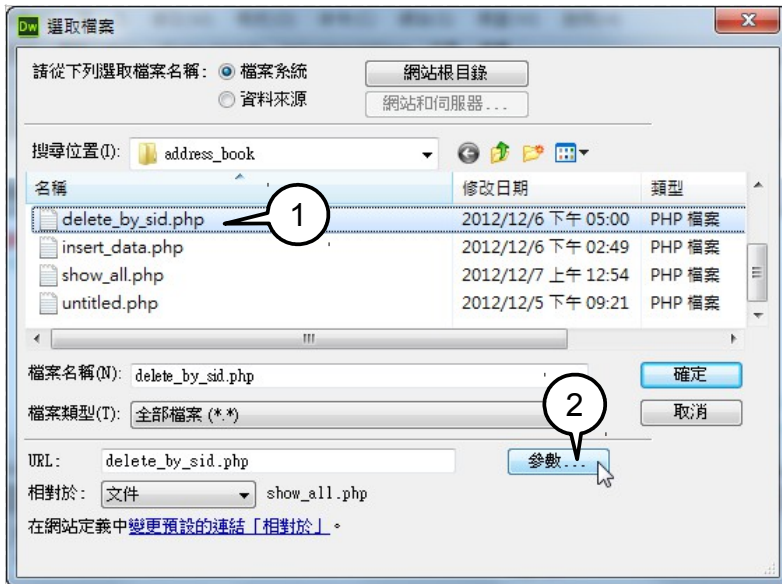


◎ 標記「X」，並點選「屬性」面板裡「連結」欄後的「瀏覽檔案」圖示，以加入連結。



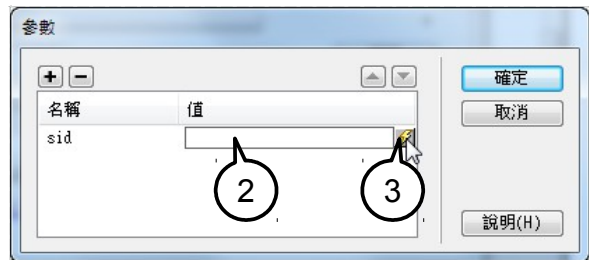
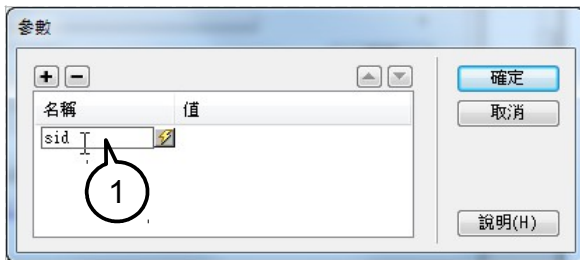
- ① 標記「X」。
- ② 點選「屬性」面板裡「連結」欄後的「瀏覽檔案」圖示。

◎ 選取「delete\_by\_sid.php」，點按 URL 欄後的「參數」按鈕以，以在連結中加入 GET 參數。



- ① 選取「delete\_by\_sid.php」。
- ② 點按 URL 欄後的「參數」按鈕。

◎ 在「參數」面板設定參數。



- ① 點選「名稱」欄，並輸入名稱為「sid」。
- ② 點選「值」欄。
- ③ 點選閃電圖示，以加入動態資料。



- ① 在「動態資料」面板，點選「資料集(Recordset1)」中的「sid」。
  - ② 點按「確定」按鈕。
- ※ 於「參數」面板按確定，回到「選取檔案」面板，再按「確定」按鈕完成設定。

- ◎ show\_all.php 存檔後，以瀏覽器預覽，點選某筆資料前的「X」，資料會立刻刪除。但是這種方式似乎沒有反悔的機會，萬一按錯，資料也找不回來。
- ◎ 修改 show\_all.php，利用 JavaScript 的 confirm() 函式來要求確認。

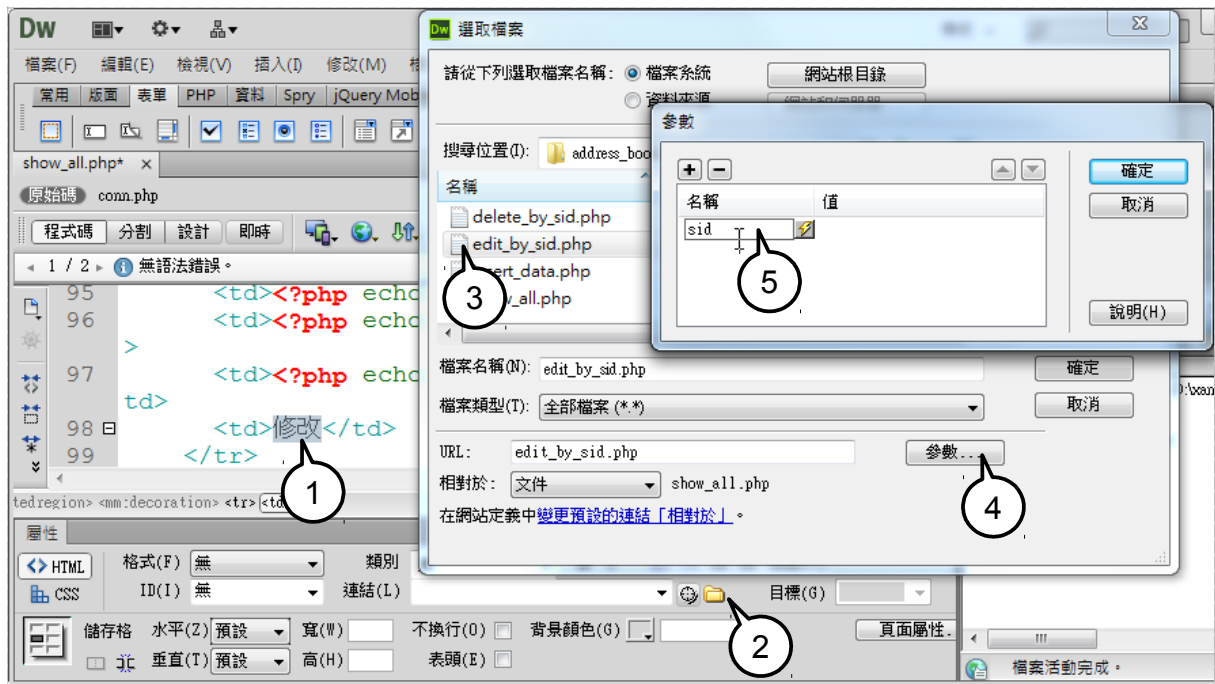
```
<!-- 略 -->
<script type="text/javascript">
    function delete_it(sid) {
        if( confirm("是否刪除流水號為" + sid + "的資料") ) {
            location = "delete_by_sid.php?sid=" + sid;
        }
    }
</script>
</head>

<body>
<table width="100%" border="1" cellspacing="0" cellpadding="0">
  <?php do { ?>
    <tr>
      <td><a href="javascript:delete_it(<?php
          echo $row_Recordset1['sid']; ?>)">X</a></td>
      <td><?php echo $row_Recordset1['sid']; ?></td>
      <td><?php echo $row_Recordset1['name']; ?></td>
    <!-- 略 -->
```

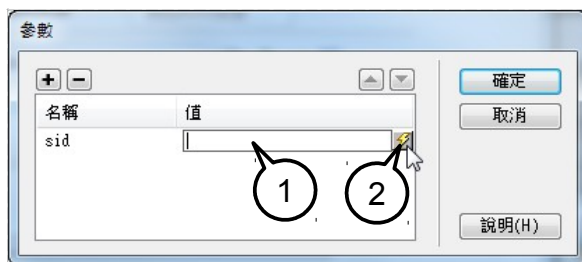
- ◎ 用 <script> 標示 JavaScript 的內容，定義 delete\_it() 函式。delete\_it() 接收 sid 參數，接著以 confirm() 讓使用者確認是否要刪除資料，若為是，則導向 delete\_by\_sid.php 並設定 GET 參數 sid 的值。
- ◎ X 的連結處，由資料庫取得的 sid 值，直接設定為呼叫 delete\_it() 函式的參數。預覽網頁，並試著刪除看看，會看到確認的對話框，此時是有機會反悔的。
- ◎ 在 <a> 標籤中的 href 屬性呼叫 JavaScript 程式時，前面要標上「**javascript:**」。

## ➤ 17 更新資料 ( 修改資料 )

- ◎ 更新資料的做法類似刪除資料，在資料的呈現頁面 ( show\_all.php )，加上修改的連結，使用者挑選要修改的資料項目，再進入單項資料修改頁面 ( edit\_by\_sid.php )。
- ◎ 建立一個新檔 edit\_by\_sid.php，先不需要在裡面做編輯。在 show\_all.php 呈現資料的表格增加一欄，並填入文字「修改」，標記「修改」後，點按增加「連結」的「瀏覽檔案」圖示。
- ◎ 同上一節類似的做法，挑選連結檔案「edit\_by\_sid.php」，點按「參數」按鈕加入動態資料，參數名稱為「sid」，值為資料庫而來的 sid 欄位動態資料。



- ① 選取「修改」。
- ② 瀏覽檔案。
- ③ 選取「edit\_by\_sid.php」。
- ④ 點按「參數」。
- ⑤ 點按「名稱」欄，輸入「sid」。
- ⑥ 點按「確定」完成。



- ① 點按「值」欄。
- ② 點選閃電圖示，以加入動態資料。
- ③ 在「動態資料」面板，點選「資料集(Recordset1)」中的「sid」。
- ④ 點按「確定」按鈕。

◎ show\_all.php 存檔後，預覽並點按「修改」，應該會連往 edit\_by\_sid.php。目前 edit\_by\_sid.php 沒有功能，不過可以在網址列檢查一下 edit\_data.php 送過來的 GET 參數是否正確。

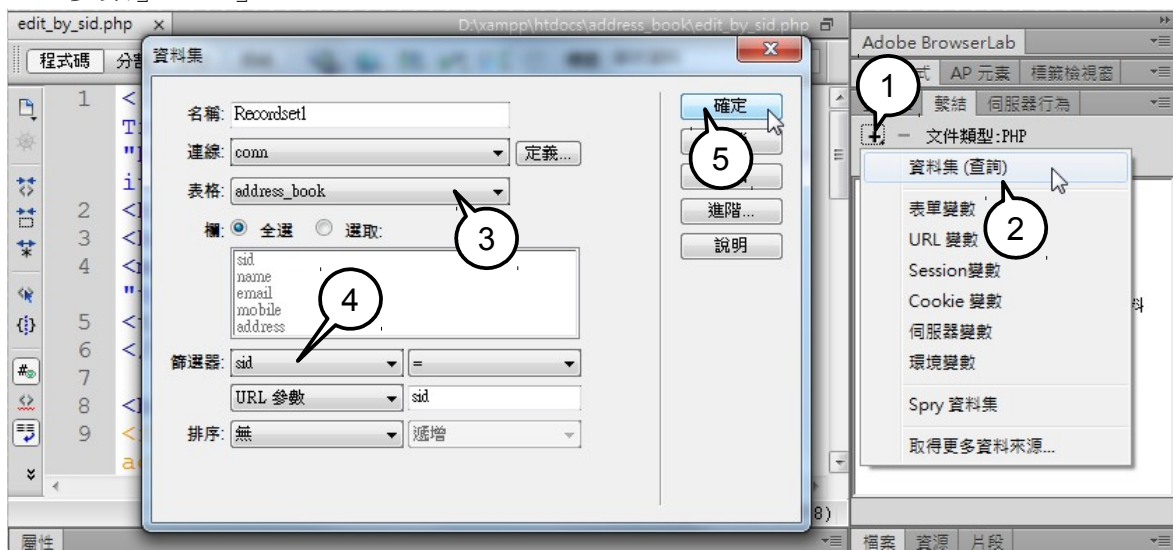
◎ 在 edit\_by\_sid.php 裡建立表單，包含有：編號、姓名、電郵、手機和地址欄。可以直接複製 insert\_data.php 裡的表單。

```
9 <form id="form1" name="form1" method="post" action="">
10 <table width="500" border="0" cellspacing="0" cellpadding="0">
11 <tr>
12 <td width="70">編號</td>
13 <td>&nbsp;</td>
14 </tr>
15 <tr>
16 <td width="70">姓名</td>
17 <td><input type="text" name="name" id="name" /></td>
18 </tr>
19 <tr>
20 <td>電郵</td>
21 <td><input type="text" name="email" id="email" /></td>
22 </tr>
23 <tr>
24 <td>手機</td>
25 <td><input type="text" name="mobile" id="mobile" /></td>
26 </tr>
27 <tr>
28 <td>地址</td>
29 <td><textarea name="address" id="address" cols="45" rows="5"></textarea></td>
30 </tr>
31 <tr>
32 <td>&nbsp;</td>
33 <td><input type="submit" value="送出" /></td>
34 </tr>
35 </table>
36 <input type="hidden" name="sid" value="" />
37 </form>
```

- ① 刪除原有的 action 屬性值。
- ② 將隱藏欄位的名稱改為「sid」。
- ③ 刪掉原有的 value 屬性值。
- ④ 表格增加一列。

◎ 表單建立好之後，先把要修改的當筆資料放入表單，讓使用者查看原有的資料。點按「繫結」的加號「+」，並在快顯選單上點選「資料集」，以顯示「資料集」面板。資料集「名稱」用預設的就可以了，「連線」使用「conn」，「表格」使用「address\_book」，資料欄位全選。

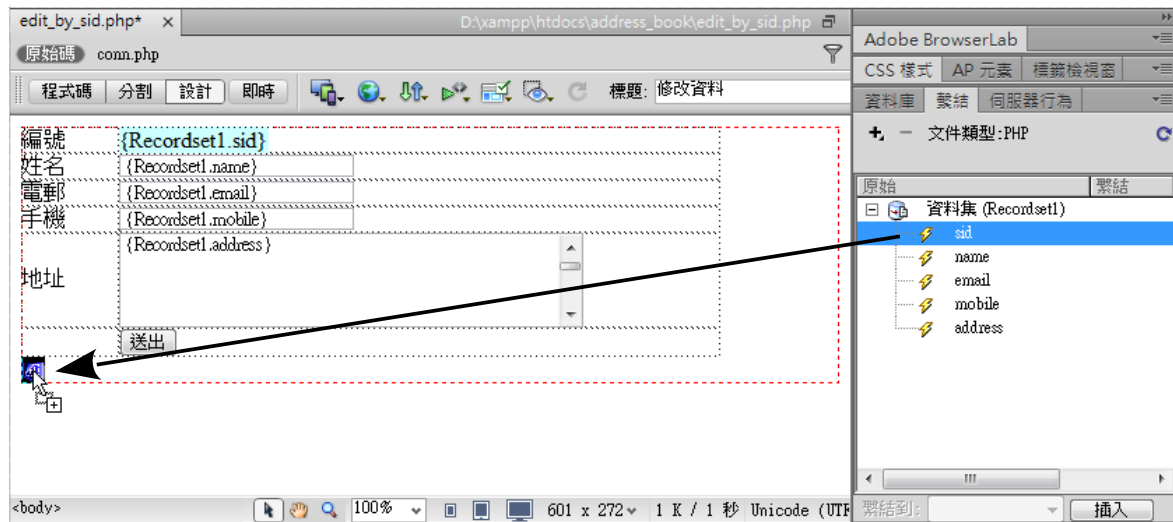
◎ 在 show\_all.php 挑定資料項之後，會傳一個 GET 參數 sid 給 edit\_by\_sid.php，以此參數為篩選資料項的條件。在篩選器選用資料表的「sid」欄位等於「=」傳入的「URL 參數」「sid」。



- ① 點按「繫結」的加號「+」。



- ② 在快顯選單上點選「資料集」。
  - ③ 「表格」使用「address\_book」。
  - ④ 篩選器選用資料表的「sid」欄位，等於「=」傳入的「URL 參數」 「sid」。
  - ⑤ 按「確定」完成。
- ◎ 完成之前的動作後，在「繫結」面板可以看到「資料集(Recordset1)」，將其下的項目：sid、name、email、mobile 和 address 分別拖曳放入表單欄位。繫結的 sid 也拖曳到隱藏欄位。



- ◎ 因為 sid 是用來識別資料項目的，避免使用者更動，所以使用隱藏欄位。設定好之後，從 edit\_data.php 開始預覽，點擊項目中的「修改」測試 edit\_by\_sid.php 是否正確顯示選定項目的資料。以下是目前 edit\_by\_sid.php 重要部份的程式：

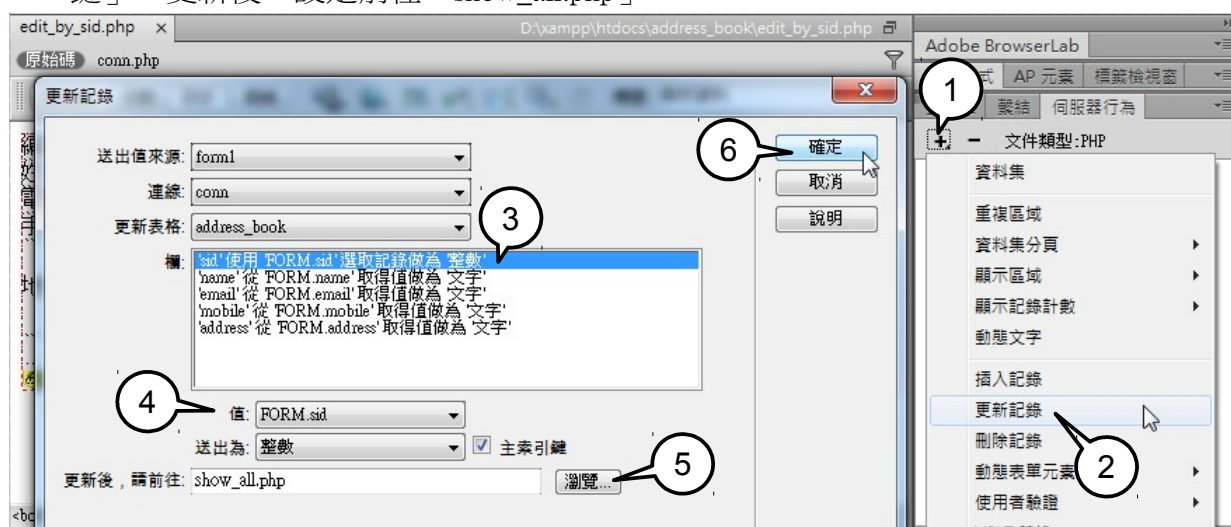
```
<?php
// 略
$colname_Recordset1 = "-1";
if (isset($_GET['sid'])) {
    $colname_Recordset1 = $_GET['sid'];
}
mysql_select_db($database_conn, $conn);
$query_Recordset1 = sprintf("SELECT * FROM address_book WHERE sid = %s",
GetSQLValueString($colname_Recordset1, "int"));
$Recordset1 = mysql_query($query_Recordset1, $conn) or die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);
$totalRows_Recordset1 = mysql_num_rows($Recordset1);
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>修改資料</title>
</head>
<body>
<form id="form1" name="form1" method="post" action="">
    <table width="500" border="0" cellspacing="0" cellpadding="0">
```

```

<tr>
  <td width="70">編號</td>
  <td><?php echo $row_Recordset1['sid']; ?></td>
</tr>
<tr>
  <td width="70">姓名</td>
  <td><input name="name" type="text" id="name" value="<?php echo
$row_Recordset1['name']; ?>" /></td>
</tr>
<!-- 略 -->

```

- \$colname\_Recordset1 是用來接收 URL 參數 sid 的變數，預設其值為 -1。
- SQL 語法使用 WHERE 子句來做條件過濾，注意這裡是用一個等號「=」做是否相同的判斷。
- 原則上應該只有一筆資料會被查詢到，以關聯式陣列取得資料並設定給 \$row\_Recordset1。
- ◎ 目前只能查看資料，我們將加入更新資料的功能。點擊「伺服器行為」面板裡的加號「+」，於快顯選單中選取「更新記錄」。
- ◎ 在「更新記錄」面板，所有設定幾乎都幫我們搞定，檢查一下資料表欄位和表單欄位的對應。最重要的是第一欄，值「FORM.sid」，送出為「整數」，為「主索引鍵」。更新後，設定前往「show\_all.php」。



- ① 點按「伺服器行為」的加號「+」。
  - ② 在快顯選單上點選「更新記錄」。
  - ③ 檢查表單送出的值寫入的欄位名稱。
  - ④ 值「FORM.sid」，送出為「整數」，為「主索引鍵」。
  - ⑤ 更新後前往「show\_all.php」。
  - ⑥ 按「確定」完成。
- ◎ 雖然已經設定更新記錄後，前往「show\_all.php」，到該頁面時，由於分頁顯示的關係只會顯示第一頁，常常不能直接看到更新的該筆資料。可以將前一個頁面的網址（包含 GET 參數）記住在 Cookie，之後再取出前往。手動加入的程式標為粗體字。

```
// 略
$http_refer = $_SERVER['HTTP_REFERER'];
if(strpos($http_refer, $_SERVER['PHP_SELF'])===false) {
    setcookie("http_refer", $http_refer);
}

$editFormAction = $_SERVER['PHP_SELF'];
if (isset($_SERVER['QUERY_STRING'])) {
    $editFormAction .= "?" . htmlentities($_SERVER['QUERY_STRING']);
}

if ((isset($_POST["MM_update"])) && ($_POST["MM_update"] == "form1")) {
    $updateSQL = sprintf("UPDATE address_book SET name=%s, email=%s, mobile=%s,
address=%s WHERE sid=%s",
        GetSQLValueString($_POST['name'], "text"),
        GetSQLValueString($_POST['email'], "text"),
        GetSQLValueString($_POST['mobile'], "text"),
        GetSQLValueString($_POST['address'], "text"),
        GetSQLValueString($_POST['sid'], "int"));

    mysql_select_db($database_conn, $conn);
    $Result1 = mysql_query($updateSQL, $conn) or die(mysql_error());

    $updateGoTo = "show_all.php";
    if (isset($_SERVER['QUERY_STRING'])) {
        $updateGoTo .= (strpos($updateGoTo, '?')) ? "&" : "?";
        $updateGoTo .= $_SERVER['QUERY_STRING'];
    }
    // header(sprintf("Location: %s", $updateGoTo));
    header(sprintf("Location: %s", $_COOKIE['http_refer']));
}
// 略
```

- 取得上一網頁的參照 (URL)，並設定給變數 \$http\_refer。
- 檢查上一網頁 (連結過來的網頁) 是否為本頁。若非本頁，則將 URL 記錄在 Cookie 裡。為什麼要判斷是不是本頁呢？主要是按「修改」時，資料會 POST 給自己 (edit\_by\_sid.php)。
- 取得本頁的網址並保留 GET 參數，設定給 \$editFormAction 變數。
- 若有名為 MM\_update 的 POST 參數，而且其值為「form1」則設定 SQL 程式依 POST 參數 sid 的值為條件，更新記錄。
- 原有前往「show\_all.php」的設定可以刪除。
- 從 Cookie 取 http\_refer 的值並前往該網址。

## ➤ 18 PHP 的 MySQL 函式

- ◎ PHP5 對 MySQL 資料庫的操作可以透過兩組函式（extension）mysql 及 mysqli。mysqli 是較新的版本且速度也較快，但 Dreamweaver 裡面為了相容性依然使用 mysql，而沒有選用 mysqli，因此就不對 mysqli 多做介紹了。下表為 mysql 在 Dreamweaver 中常出現及常用的函式簡介。

mysql 函式名稱	簡單說明
mysql_close()	關閉資料庫連線。用 mysql_connect 開啟的連線會在網頁程式結束後自動關閉。
mysql_connect()	建立資料庫連線。
mysql_data_seek()	移動資料集內部的指標。
mysql_errno()	當錯誤發生時，取得 MySQL 操作的錯誤碼。
mysql_error()	當錯誤發生時，取得 MySQL 操作的錯誤訊息。
mysql_escape_string()	對字串做 SQL 脫逸。
mysql_fetch_array()	從資料集中讀取一筆資料，轉換為關聯式、索引式、陣列或者同時具有兩者功能。
mysql_fetch_assoc()	從資料集中讀取一筆資料，轉換為關聯式陣列。
mysql_fetch_field()	從資料集中取得欄位表頭的資料，並轉換為物件。
mysql_fetch_object()	從資料集中讀取一筆資料，並轉換為物件。
mysql_fetch_row()	從資料集中讀取一筆資料，並轉換為索引式陣列。
mysql_free_result()	釋放資料集所佔用的記憶體。
mysql_insert_id()	取得最近新增資料的主鍵值。
mysql_num_rows()	計算資料集中的資料筆數。
mysql_pconnect()	建立持續性的資料庫連線。
mysql_query()	執行 SQL 語法。
mysql_select_db()	選擇使用的資料庫。
mysql_set_charset()	設定校對編碼。