

## 第 19 章 JavaScript 的進階語法

資科系  
林偉川

### 函數的應用-函數

- JavaScript 定義函數的語法如下：

```
function 函數名稱 (參數串列) {  
    ..... //程式碼  
    return 回傳值  
}
```

## 函數的應用-函數

- 下面是語法各部份的說明。
  - 函數名稱：定義函數的名稱。
  - 參數串列：函數被呼叫時，接收傳入資料的參數。參數的宣告的語法如下：  
    型別 參數1, 型別 參數2, ……
  - 回傳值：將函數的運算結果傳出。如果函數沒有傳出值，則此敘述可省略。

3

## 函數的應用-函數

- 函數可將程式中重複的動作或運算，獨立成一個子程式。當需要執行這些動作時，將以呼叫的方式執行它們，比方：您在程式中，常常需要計算一個物體的體積與表面積時，您就可以把這些計算動作撰寫成函數，然後在需要的時候呼叫它。

```
function CircleArea (Radius){  
    var PI = 3.1415926 //宣告PI為圓周率常數  
    return PI * Radius * Radius //圓面積計算  
}
```

4

## 函數的應用-函數

- 函數(方法)的呼叫

當欲呼叫函數時，只要在程式中運用以下的語法即可呼叫。

函數名稱 (參數1, 參數2, ..., 參數N)

下面的語法將呼叫前面所提及的圓面積函數，並計算半徑為5cm之圓面積。

CircleArea(5)

下面的語法，將把12與15傳入GCD函數，以求取兩數的最大公因數。

GCD(12, 15)

5

## 函數的應用-全域變數與區域變數

- 當在網頁主程式中，宣告一個變數後，該變數便是一個全域變數 (Global Variable)，這類變數的可使用範圍，為整個網頁中的程式碼部份，並包含網頁中函數內的程式敘述。
- 在函數內宣告的變數，稱之為區域變數 (Local Variable)，它的可見用範圍僅及函數內的程式敘述，在宣告變數的函數外，無法使用該變數。

6

## 函數的應用-全域變數與區域變數

- 全域變數與區域變數的宣告

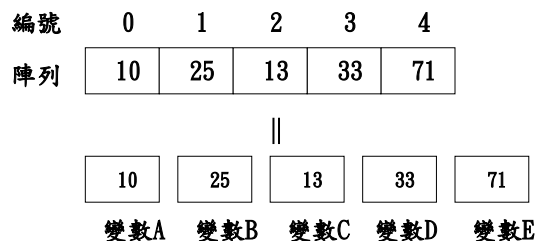
下表為全域變數與區域變數的差異：

全域變數	區域變數
適用於整個網頁主程式的宣告，其有效範例為文件中所有<Script>標籤。	僅適用於函數中宣告，所以變數有效範圍僅止於所宣告的函數。
不一定要使用var宣告變數	一定要使用var宣告變數

7

## 陣列的宣告 -什麼是陣列？

- 陣列是由好幾個陣列元素所組成。而每一個單獨的陣列元素，就相當於一個變數，可用於儲存資料。陣列中的每個元素都有一個編號，用於存取陣列的元素，這個編號的起始值為0。因此，陣列的第一個元素索引值為0，依此類推。



8

## 陣列的宣告 - 陣列的宣告與使用

- 陣列的宣告

陣列變數的宣告語法如下：

```
陣列名稱 = new Array ( )
```

整個語法將利用new運算子，與Array函數建立出一個陣列物件。下面敘述將宣告一個名稱為aryName的陣列變數。aryName = new Array()

如果在宣告陣列時，已經知道陣列中元素的值，您可以運用以下的語法，宣告陣列變數。

```
陣列名稱 = new Array (陣列元素一, 陣列元素二……)
```

下面的敘述將宣告一個名為aryName的陣列，該陣列中將有三個變數值，分別為10、20、30。

```
aryName = new Array(10, 20, 30)
```

9

## 陣列的宣告 - 陣列的宣告與使用

- 陣列元素

當我們欲取得陣列變數中某一個元素時，就必須靠陣列的索引值。語法如下：

```
陣列名稱[索引值]
```

下面的程式敘述將宣告一個aryVal陣列變數，並將陣列中第5個元素之值設為10。

```
varAry = new Array ( )
```

```
varAry[4] = 10
```

使用陣列時，通常會與for迴圈配合，下面的敘述將設定陣列中每個元素的起始值為0。

```
for (i=0; i<=9; i++){ aryVal(i) = 0 }
```

10