

網頁
設計

講師：田甜甜

Email: rita.design@gmail.com

RITA TEACHING

JavaScript v.15.0

ENTER

<http://tutorial.jumpdesign.tw/>

JavaScript基本概念

JavaScript 傳輸方式：



JavaScript 網頁用法：

1. MIME型態：`<script type="text/javascript"> </script>`
2. HTML5型態：`<script>` 其他可以省略，因為預設就是javascript。
//加入外部連結：`<script src="myscript.js">`
3. `<script>` 可以放在網頁中的任何位置，多放置於`<head>` 檔頭區 或 網頁程式碼最下方 (`</body>` 之上)。
4. javascript 語法有區分英文大小寫。
5. 分號「`;`」可作為單一程式敘述結束符號。
6. 大刮號「`{`」`}`」可作為程式區塊，由單一程式或多個程式敘述組成。
7. javascript 註解寫法：
// 單行註解內容
/* 多行註解內容 */
8. javascript 語法不會區分「單引號」或「雙引號」。
9. 語法使用「駝峰式拼寫法」。

JavaScript 基本功能：

1. 互動性動態文件內容網頁
2. 重新定義網頁內容、html標籤及屬性樣式
3. 表單驗證和送出
4. 處理網頁或html標籤事件
5. 建立web應用程式

JavaScript Basic\

JavaScript基本概念

RITATEACHING

JavaScript可以“顯示”數據的方式：

JavaScript沒有任何內置的列印或顯示功能。

- 1、使用寫入HTML輸出的`document.write()`。
- 2、寫入到一個HTML元素，使用的`innerHTML`。
- 3、寫入瀏覽器控制台，使用的`console.log()`。
- 4、寫入一個警告框，使用`window.alert()`。

第一份Javascript：document.write()

```
<!doctype html>
<html>
<head>
<script type="text/javascript">
```

```
//document物件(代表整份HTML文件)的write方式輸出內容至瀏覽器顯示
document.write("第一份JavaScript程式");
```

```
</script>
```

*/*document.write() 可於html文件中寫入字串，還可以包含 HTML 的標籤，例如換行或顯示圖片*/*

```
</head>
<body>
</body>
</html>
```

JavaScript基本概念

重新定義網頁內容、html標籤及屬性樣式

- 重新定義HTML網頁內容：innerHTML

```
document.getElementById("demo").innerHTML = "Hello JavaScript";
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p id="demo">JavaScript 可以重新定義HTML網頁內容 </p>
```

```
<button type="button" onclick="document.getElementById('demo').innerHTML  
= 'Hello JavaScript!' ">按我 </button>
```

```
</body>
```

```
</html>
```

-
- 重新定義HTML元素屬性：

```
document.getElementById('myImage').src='xxx.png'
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<!-- 改變src屬性 -->
```

```
<button onclick="document.getElementById('myImage').src='tvopen.png'">  
打開電視 </button>
```

```

```

```
<button onclick="document.getElementById('myImage').src='tvclose.png'">  
關閉電視 </button>
```

```
</body>
```

```
</html>
```

JavaScript Basic\

JavaScript基本概念

RITATEACHING

重新定義網頁內容、html標籤及屬性樣式

- 重新定義CSS樣式：style

```
document.getElementById('demo').style.color = '#f00';
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p id="demo">JavaScript 可以重新定義CSS網頁樣式</p>
```

```
<button type="button" onclick="document.getElementById('demo').style  
.color = '#f00' ">按我</button>
```

```
</body>
```

```
</html>
```

Javascript語法功能：

【敘述】

指定值、事物相加、計算、
javascript函式庫

```
var cup = 96.3;  
var cup = 4;  
var cup = true;  
var cup = "Rita" ;  
cup = (Cup-32)*5/9;  
cup = cup+ "加油！" ;  
var cup = Math.random();
```

【迴圈】

重複執行敘述

```
while 迴圈  
for 迴圈
```

【條件式】

根據程式狀態決定執行事項

```
if ( 條件 ) {  
    成立時執行指令  
} else {  
    不成立時執行指令  
}
```

JavaScript變數

變數是用來儲存數據值的容器。

JavaScript 變數基本型態 (Primitive type) :

此資料型態所佔用的記憶體大小是固定的 (除了字串，其大小不固定，因字數而定)，使用傳值(pass by value)的方式，指派或傳遞傳遞複本。

"String" ('字串') 單引號/雙引號	Number (數字) 整數 / 浮點數	Boolean (布林) true / false	Null (空值)	undefined (未定義)
---------------------------------	----------------------------	---------------------------------	--------------	--------------------

JavaScript 變數參考型態 (Reference type) :

此資料型態所佔用大小是不固定的，使用傳址(pass by reference)的方式，指派或傳遞傳遞複本。

陣列 (Array)	物件 (Object)	函式 (Function)
---------------	----------------	------------------

*JavaScript使用「**var**」變數關鍵字建立一個容器，用來儲存資料和進行運算的基本程式功能，我們必須為變數指定一個合適的名稱，才能進一步操作

JavaScript變數

JavaScript 變數命名注意事項：

變數命名

1. 不可使用 JavaScript 語法保留字
2. 建立多個詞彙時，建議使用「駝峰式拼寫法」
3. 命名開頭字，除底線符號「_」、「\$」以外，不可使用數字、句號或任何符號，但仍不建議使用符號。
4. 命名中間，不可使用「空格」或「+」、「-」符號；可使用其他符號或數字

宣告變數

1. 同時宣告多個變數，使用逗號「**,**」分隔連續宣告
2. 宣告變數時，可不指定變數值：`var strName,intBalance;`
亦可同時指定變數值：`var strName = "Rita田"`
3. 「**+**」可用於數字相加，也可以是聯字串的運算子
4. 當變數未指定變數的值時，變數值為 `undefined` (未定義)
5. 當變數沒有值時，可指定變數值為 `null` (沒有值)
6. 變數範圍(Scope)可區分為「**全域變數**」(global) 和「**區域變數**」(local)
7. JavaScript中允許省略「**var**」這個關鍵字來給定變數初始值
8. 所有省略「**var**」指令建立的變數皆為全域變數(不建議省略)
9. 具有**動態型別**，就算不指定型別，程式碼執行時也會自動找出該使用的型別

/* JavaScript中會先解析var，所以即使程式後面才出現宣告，前面就先使用了也不算是未宣告，算是宣告而未給初始值：

```
console.log(m); // undefined  
var m = 1;    */
```

JavaScript變數基本語法練習

宣告變數指定值：等號(=)是一個“轉讓”操作符，不是“等於”操作符

/* 「+」可用於數字相加，也可以是聯字串的運算子

cup = cup - 1; cup(變數) =(指定) cup-1(運算式); */

1. 浮點數：`var cup = 96.3;`
2. 整數：`var cup = 4;`
3. 布林值：`var cup = true;`
4. 字串：`var cup = "Rita";`
5. 運算式：`cup = 5*5+3;`
`cup = (cup-32)*5/9;`
6. 事物相加：`cup = cup + "加油!";`
7. 內部函式庫：`var cup = Math.random();`

習題：`console.log();`

● `var tempA = 3;`
`var tempB = (10 / 5) + tempA + 50;`

tempB的答案是？
`console.log(tempA,tempB);`

● `tempC = "Rita"+"1"+"2";`
`tempD = 15+5+"Rita";`
`tempE = "Rita"+15+5;`

tempD,tempE的答案是？
`console.log(tempC,tempD,tempE);`

● `tempF = "Rita" + (15+5);`

tempF的答案是？
`console.log(tempF);`

● `tempM = "3" + "4";`

結果等於：？
`console.log(tempM);`

● `tempN = "3" * "4";`

結果等於：？
`console.log(tempN);`

● `var a = 10;`
`var b = 5;`
`tempG = (a > b);`
`tempH = (a < b);`
`tempI = (a == 10);`
`tempJ = (b == 10);`

結果等於：(true/false)？
`console.log(tempG,tempH,tempI,tempJ);`

● `myColor = "blue";`
`myColor != "pink";`

當myColor為"blue"結果等於：(true/false)？
`console.log(myColor != "pink");`

● `tempK = 3;`
`tempL = (2 * Math.PI) * tempK // (Math.PI =3.1415.....)`

結果等於：？
`console.log(tempL);`

JavaScript變數基本語法練習

宣告變數指定值 : document.write()

```
<!doctype html>
<html>
<body>

<script type="text/javascript">
  //宣告變數
  var Name = "Rita田";
  var Year = 18;
  var Sex = true;
  //顯示變數內容
  document.write( "姓名：" + Name + "<br>" );
  document.write( "性別：" + Sex + "<br>" );
  document.write( "年齡：" + Year );
</script>

</body>
</html>
```



```
// var Name = "Rita田",Year = 18,Sex = true;
```

宣告變數不指定值 : document.write()

```
<script type="text/javascript">
  //宣告變數
  var Name , Year , Sex ;
  //指定變數值
  Name = "Rita田" ;
  Year = 18 ;
  Sex = true ;
</script>
```

JavaScript迴圈

while 迴圈

for 迴圈

while 迴圈

while迴圈 + alert警告視窗：window.alert()

```
<!doctype html>
<html>
<head>

<script type="text/javascript">

    var cup = 10;

    while ( cup > 0 ) {
        alert ( "還有多餘的咖啡" );
        cup = cup - 1 ;
    }

    alert ( "沒有咖啡人生是黑白的" );

</script>

</head>
<body>
</body>
</html>
```



//while迴圈以某個條件不確定數量時，作為控制條件。

JavaScript迴圈

while 迴圈

for 迴圈

for 迴圈

for迴圈 : window.alert()

```
<!doctype html>
<html>
<head>
```

```
<script type="text/javascript">
```

```
    for ( cup = 10 ; cup > 0 ; cup -- ) {
```

```
        alert ( "還有多餘的咖啡" );
    }
```

```
    alert ( "沒有咖啡人生是黑白的" );
```

```
</script>
```

```
</head>
<body>
</body>
</html>
```

陳述式

方法

for

alert

//for迴圈以固定數量，作為控制條件。

JavaScript If/Else \

JavaScript判斷式

RITATEACHING

if 判斷式+confirm()函式：window.confirm()

```
<!doctype html>
<html>
<body>
  <h2>confirm訊問視窗練習</h2>

  <script type="text/javascript">

    x=confirm("是否訂閱電子報？");
    if (x) {
      alert("感謝您已訂閱電子報！");
    } else {
      alert("您尚未訂閱電子報！");
    }
  </script>

  <h3>先執行script才執行後續內容</h3>

</body>
</html>
```

條件陳述式

方法

if / else

alert

JavaScript If/Else \

JavaScript判斷式

RITATEACHING

if 判斷式-判斷大小 : window.alert()

```
<!doctype html>
<html>
<head>

<script type="text/javascript">
    for ( cup = 10 ; cup > 0 ; cup -- ) {

        if ( cup < 3 ) {
            alert ( "請補充咖啡" );
        } else if ( cup < 5 ) {
            alert ( "剩沒幾杯咖啡了" );
        } else {
            alert ( "還有多餘的咖啡" );
        }
    }
</script>

</head>
<body>
</body>
</html>
```

陳述式	for
條件陳述式	if / else
方法	alert

JavaScript Function\

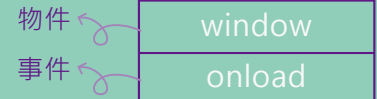
JavaScript函式值

RITATEACHING

```
<script type="text/javascript">
  function init () {
    alert ( " 自訂函數 " );
  }
  window.onload = init ;
</script>
```

簡化

```
<script type="text/javascript">
  window.onload = function() {
    alert ( " 自訂函數 " );
  }
</script>
```



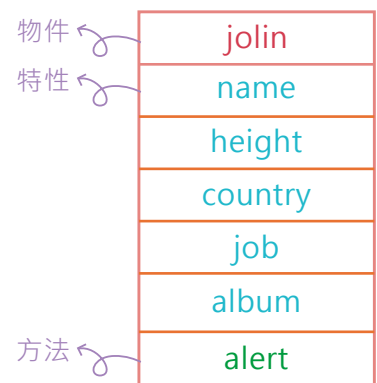
/* window物件的onload事件可以指定函式，onload的功能為，當瀏覽器完成整份HTML文件的解析後，如果onload事件有設定函式，就會予以呼叫 */

「物件+函式方法」：

/* 需要重覆調用的語句，可設定為function函式 */

```
function myAlert() {
  alert("你好");
}
myAlert();
```

```
var jolin = {
  name: "蔡依林",
  height: 158,
  country: "Taiwan",
  job: ["歌手", "音樂總監", "翻糖蛋糕師"],
  album: function () {
    alert ( " 我呸 " );
  }
}
jolin.album();
```



JavaScript物件

/* 可將參數傳送到function函式裡，用function函式得到數據，再利用return屬性回傳值 */

js- 「物件+函式方法」指定參數Parameter 練習：

```
function myAdd(a,b){  
    return(a+b);  
}
```

```
document.getElementById("demo").innerHTML=myAdd(5,8);
```

```
// 宣告變數=tempAge(指定參數給tempAge)  
var KonanAge = tempAge ( "柯南" , 7 );  
alert ( KonanAge );  
  
var KogoroAge = tempAge ( "毛利小五郎" , 38 );  
alert ( KogoroAge );  
  
function tempAge ( Name , Age ){  
    if ( Age < 18 ){  
        return Name + "未成年！";  
    } else {  
        return Name + " 成年人！";  
    }  
}
```

JavaScript Prompt\

JavaScript提示

RITATEACHING

ScriptTag.js

```
window.onload = function() {  
  
    var welcome = document.getElementById("demo");  
    welcome.innerHTML = "Hello!" + prompt ("Input your name")+" ! ";  
  
};
```



js-外部連結練習：

```
<!doctype html>  
<html>  
<head>  
<meta charset="UTF-8">  
<title>js-外部連結練習</title>  
<script type="text/javascript" src="js/ScriptTag.js"> </script>  
</head>  
<body>  
    <span id="welcome"> </span>  
</body>  
</html>
```


JavaScript陣列

/* 可分配連續值給記憶體來儲存，利用元素索引計算對應的儲存位址 */

//1

```
var myArray = new Array();
```

```
myArray [0] = "Apple";
```

```
myArray [1] = "Orange";
```

//2

```
myOtherArray = ["dog","cat"];
```

//3

```
myOtherArray .push("bear"); //增加新的址
```

//4

```
myOtherArray .splice( 0,1 ); //指定的位置,刪除幾個址
```

//5 指定的位置,刪除幾個址,要加入的址 (可加入多個址)

```
myOtherArray .splice( 2,0,"fox","bird");
```

//2

```
console.log(myOtherArray ); //查看瀏覽器記錄
```

//6

```
console.log(myOtherArray .length); //總共有幾個址
```

//1

```
document.getElementById("myBtn").onclick=function(){
```

```
    document.getElementById("demo").innerHTML=myArray[0];
```

```
}
```

JavaScript陣列

js-array+外部連結練習練習：

```

<!doctype html>
<html>
<head>
<meta charset="UTF-8">
<title>array+外部連結練習</title>
<script type="text/javascript" src="js/ScriptArray.js"></script>
</head>
<body>
  <h1>溫度紀錄</h1>
  <ul>
    <li id="temp0"></li>
    <li id="temp1"></li>
    <li id="temp2"></li>
    <li id="temp3"></li>
    <li id="temp4"></li>
  </ul>
</body>
</html>

```

ScriptArray.js

```

function showTemps ( ) {

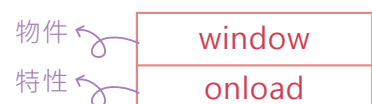
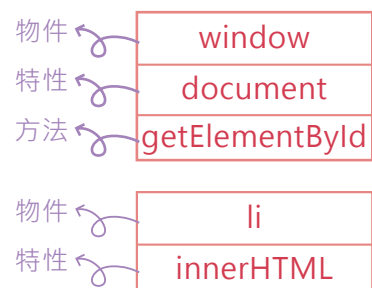
var tempByHour = new Array();

tempByHour [ 0 ] = 38.5 ;
tempByHour [ 1 ] = 37.8 ;
tempByHour [ 2 ] = 38 ;
tempByHour [ 3 ] = 37.6 ;
tempByHour [ 4 ] = 37 ;
// 陣列亦可縮寫為「實字陣列」： var tempByHour = [38.5,37.8,38,37.6,37];

for ( var i = 0 ; i < tempByHour . length ; i ++ ){
  var theTemp = tempByHour [ i ] ;
  var id = "temp" + i ;
  var li = document.getElementById ( id );
  if ( i ==0 ){
    li . innerHTML = "現在的溫度： " + theTemp;
  } else {
    li . innerHTML = i +"小時前的溫度： " + the-
Temp;
  }
}
}

window.onload = showTemps;

```



END.