

1.1.變數與值

JavaScript 啊!

你可以印出點什麼嗎?

在小括號 () 中放入要印的內容即可。雙引號 "" 代表一串文字

```
1 console.log("Hello");
```



得到

```
1 Hello
```

var1.js



掃我線上看code

現在，模擬錢包試試

錢包有 10 元，再加個 20 得到 30 元

```
1 console.log(10+20);
```

```
2 console.log(10+20-5);
```

現在花掉 5 元，但...怎麼過去的 30 元卻要重新計算一次，才能再減 5 元呢？雖然最後的結果是正確的



得到

```
1 30
```

```
2 25
```

能否像錢包一樣

保存我最後還有多少錢嗎？

var2.js



掃我線上看code

當然可以，

只要我們使用 **變數** 即可~

使用 `let` 宣告一個 `x` 變數作為錢包

```
let x = 10;
```

再使用 `=` 將 10 元放入其中

存入 20 元時，只要先從 `x` 變數
拿出原本的 10 元，再加上
20 元，得到 30 元

```
x = x + 20;
```

再透過 `=` 將 30 放回 `x` 變數中保存

```
1 console.log(x);
```

拿出來看有多少錢，確實
變成 30 元了~

再來試試花 5 元。拿出錢包裡的
錢，減 5 元，在放回 `x` 變數中

```
x = x - 5;
```

```
2 console.log(x);
```

確實剩 25 元了~



得到

1 **30**

2 **25**

var3.js



掃我線上看code

*需牢記的知識點

變數與值

變數 Variable 的功能就是計憶，沒別的功能。

操作記憶的方式就是：

- 1.存入值
- 2.取出值

在 JavaScript 中，所有的資料，都可以統稱為 值 Value。

1.2.變數與值

再來一次

```
----- 從 x 拿出 10 元，再加上 "20" -----  
let x = 10;  
-----  
x = x + "20";  
-----  
1 console.log(x);  
-----
```

↓ 得到

1 1020

發生了什麼事？
怎麼是 1020 ？

value1.js



掃我線上看code

來個檢查

使用 typeof 來看看 "20"
會是什麼樣類型的資料

```
-----  
1 console(typeof "20");  
-----  
果然，加了 "" 後就  
不是數值而是  
string 字串
```

↓ 得到

1 string

value2.js



掃我線上看code

這印證了 JavaScript 的規則，所有值遇到 string，都會變成 string 後相加，成為一個新的 string 字串

```
x = 10;  
x = x + "20";  
console.log(x);
```

難怪得到 1020

看來寫程式

真的要注意資料的類型才是~

但若是想跟字串

進行數值上的相加

可以怎麼做？

沒問題

使用 **parseInt()** 函式~

```
x = 10;  
x = x + parseInt("20");  
1 console.log(x);
```



得到

1 **30**

讚~ 果然值的型別對了
運算的結果才會正確~

value3.js



掃我線上看code

*需牢記的知識點

值與型別

在 JavaScript 中，值是有分類型的，而類型在程式中的術語即是型別 Type。型別也決定了值的運算方式。

基本的型別有：

1. 數值 number

包含整數與有小數的浮點數

如：100, 100.123

2. 字串 string、str

如："Hello"

3. 布林 boolean、bool

如：true 或 false

關鍵知識問答

Q: 要在程式中寫註解可以怎麼做？

使用雙斜線 // 即可，如下：

```
//這是註解，下列程式會印出1
```

```
console.log(1);
```

q1.js



掃我線上看code

Q: JavaScript 好像不加分號；，也可以換下一行程式？

沒錯，每行程式可以不需要分號；做換下一行程式，這時它是看換行字元決定換下一行程式，如下：

```
console.log(1)
```

```
console.log(2)
```

q2.js



掃我線上看code

Q:變數的名稱可以怎麼取？

變數必須使用：

1. 字母 A-Z, a-z
 2. 下底線 _
 3. 錢號 \$
- 作為開頭。

開頭後面可以包含數字 0-9 。

例如：

```
let abc123 = 0; //正確
```

```
let _123 = 1; //正確
```

```
let $123 = 2; //正確
```

```
let 1abc = 0; //錯誤
```

q3.js



掃我線上看code

Q: 若想要變數中的值不被更換，可以怎麼做好？

使用 `const` 宣告變數。如：

```
const name = "Mr. X"; //宣告並給值
```

```
name = "J"; //變更錯誤，因禁止變更
```

q4.js



掃我線上看code

Q: 變數可以宣告就好，不給值嗎？

可以，JavaScript 會預先放 `undefined` 的值，`undefined` 是一個特殊的值，用來表示值未被定義，注意，是值未被定義，而不是變數不存在。下列程式將印出 `undefined`：

```
let x;
```

```
console.log(x); //undefined
```

q5.js



掃我線上看code

Q: 若想先有變數，除了不給值得到 `undefined` 外，如何明確的表示變數中目前沒有值？

使用 `null`，`null`是一個特殊值，專門用來表示空值，如：

```
let x = null;  
console.log(x); //null
```

q6.js



掃我線上看code

Q: `var`也可以宣告變數對吧？

是的，但建議停用，在ES6後，加入了區塊作用域(Block Scope)中，但`var`宣告的變數會忽略區塊作用域，導致潛在的Bug問題。這方面會在第4章秘笈-流程控制中作詳細的說明。

Q: 不使用 `let`、`const`、`var` 也可以宣告變數嗎？

可以，它會以 `var` 的方式宣告變數，所以建議停用，如下：

```
x = 0; //以var方式宣告x變數  
console.log(x); //0
```

q7_1.js



掃我線上看code

嚴格模式下則不行，下列程式會發生錯誤，會出現找不到 x 變數的狀況，如下：

'use strict';	//使用嚴格模式
x = 0;	//以var方式宣告x變數
console.log(x);	//error，x變數未定義

q7_2.js



掃我線上看code

Q: 字串宣告的格式有哪些？

使用：

1. 雙引號 ""
2. 單引號 ''

如下：

```
let x = "Hello";
```

```
let y = "你好";
```

```
console.log(x); //Hello
```

```
console.log(y); //你好
```

q8.js



掃我線上看code

Q: 不同型別的資料可以 + - * / 嗎?

可以，基本上除了遇到與字串做相加，會先轉為字串做加的動作外，其他型別都是進行數值上的 + - * / 運算，規則如下表：

運算遇到	+	-	*	/
字串 string	轉為 string	轉為 number	轉為 number	轉為 number
數值 number	轉為 number	轉為 number	轉為 number	轉為 number
布林 boolean	轉為 number	轉為 number	轉為 number	轉為 number

+ 遇到其中一個值是字串，都會轉為字串與字串相加：

```
console.log('hello'+'abc'); //helloabc
console.log(true+'abc'); //trueabc
console.log(1+'abc'); //1abc
```

q9_1.js



掃我線上看code

其餘運算都會將值轉為數值進行 + - * / ，布林值true會轉為1，false會轉為0，字串也會轉為數值：

```
console.log(1 + True); //2
console.log(True + False); //1
console.log(True - True); //0
console.log(4 * '2'); //8
console.log(4 / '2'); //2
```

q9_2.js



掃我線上看code

Q: 數值若 + - * / 失敗會得到什麼?

得到 NaN，它是一個特殊值，Not a Number 的意思，表示不為一個數值，如下：

```
console.log(10 / 'Hello'); //NaN
```

q10.js



掃我線上看code

Q: 數值除以0會得到什麼?

得到Infinity，它是一個特殊值，表示無窮大，如下：

```
console.log(1 / 0); //Infinity
```

```
console.log(-1 / 0); //-Infinity
```

q11.js



掃我線上看code

Q: 變數有型別嗎?

沒有，想放什麼就放什麼，如下：

```
let x = 10; //放入數值
```

```
x = true; //放入布林值
```

```
x = "Hello"; //放入字串
```

```
console.log(x); //Hello
```

q12.js



掃我線上看code

Q: 字串可以變成整數或浮點數嗎？

可以，使用下列函式轉換即可：

`parseInt()`，轉為整數。

`parseFloat()`，轉為浮點數即有小數的數。

關於函式，我們會在第5章秘笈中作說明。示範如下：

```
let x1 = parseInt("888");  
let x2 = parseInt("888.123");  
let x3 = parseFloat("888.123");  
let x4 = parseFloat("888.123");  
console.log(x1);           //888  
console.log(x2);           //888  
console.log(x3);           //888.0  
console.log(x4);           //888.123
```

q13.js



掃我線上看code