



JavaScript

David

目錄

建立函式的方法

3



建立函式的方法

第一種：函數宣告(Function Declaration)
為最常用的建立函式方式
在程式內任何地方調用都不會有問題

```
function func1() {  
    console.log('func1');  
};
```



建立函式的方法

第二種：函數運算式(Function Expressions)

用此方式宣告函式要注意的地方為順序

一定要先宣告函式，在去調用它，否則會出現錯誤訊息 →

```
✖ ▶ Uncaught TypeError: fnExpr is not a function  
    at <anonymous>:1:1
```

```
var fnExpr = function () {  
    console.log('fnExpr')  
};
```



建立函式的方法

第三種：函數關鍵字(Function Keyword)

使用 Function 關鍵字建立函數物件，在使用時將參數與函數的內容依序傳入 Function，就可以建立一個函數物件。

此方式也需要先宣告函式在調用

```
var add = new Function('a', 'b', 'return a + b');
```



建立函式的方法

動動腦：

以下程式碼結果會是如何？

```
var globe = "global";
function scope() {
    var scope = "scoped",
        scop = Function('console.log(typeof scope)'),
        glob = Function('console.log(typeof globe)');

    scop();
    glob();
}

scope();
```



建立函式的方法

答案：undefined、string

原因：這個函數中只能存取全域（global）的變數或是存在於該函數內部的變數。



建立函式的方法

讓scop()可以顯示值的更改方式：

```
var globe = "global";
function scope() {
    var scop = "scoped",
        scop = Function('var scop = true; console.log(typeof scop)'),
        glob = Function('console.log(typeof globe)');

    scop();
    glob();
}

scope();
```

```
var globe = "global";
var scop = 10;
function scope() {
    var scop = "scoped",
        scop = Function('console.log(typeof scop)'),
        glob = Function('console.log(typeof globe)');

    scop();
    glob();
}

scope();
```



參考網址

https://developer.mozilla.org/zh-TW/docs/Web/JavaScript/Reference/Global_Objects/Function

<https://blog.gtwang.org/programming/defining-javascript-functions/>



THE END

