

RandomVariable / August 14, 2012 10:32PM

[Browser 的 JavaScript - Event 及其觸發源頭的取得](#)

我們都知道 IE 和 FireFox 對「事件」取得的方式 不一樣

(1) IE 可利用全域變數 window.event

(2) FireFox 可利用傳遞參數的方法來取得 event

，或者如果沒有傳遞參數，則系統會自動將「事件」放在第一個參數 arguments[0]

我試了一段程式如下，該程式修改自 <http://www.yongfa365.com/Item/9fdc446493798d53.html>

```
=====
function Get_srcElement() { e = arguments.callee.caller.arguments[0] || window.event; a = e.target || e.srcElement;
var srcElement="" srcElement = srcElement + "\n" + "event.srcElement.id : " + a.id srcElement = srcElement + "\n"
+ "event.srcElement.tagName : " + a.tagName srcElement = srcElement + "\n" + "event.srcElement.type : " +
a.type srcElement = srcElement + "\n" + "event.srcElement.value : " + a.value srcElement = srcElement + "\n" +
"event.srcElement.name : " + a.name srcElement = srcElement + "\n" + "event.srcElement.className : " +
a.className srcElement = srcElement + "\n" + "event.srcElement.parentElement.id : " + a.parentElement.id
srcElement = srcElement + "\n" + "event.srcElement.getAttribute : " + a.getAttribute alert(srcElement) }
=====
```

這個程式在 各下列瀏覽器都可正確執行：

IE9

FireFox 14.0.1

Google Chrome 21.0.1180.77

Opera 12.01

Safari 5.1.7

不過，有一些特別的地方

(1) 如果

e = arguments.callee.caller.arguments[0] || window.event;

寫成

e = arguments[0] || window.event;

FireFox 在按下button後不會有反應，而且「錯誤主控台」出現「TypeError: e is undefined」的錯誤訊息

也就是 arguments[0]前面一定要加 arguments.callee.caller

(2) 如果這兩句

e = arguments.callee.caller.arguments[0] || window.event;

a = e.target || e.srcElement;

合併成一句也會出錯

(2-1) 如果合併成

var a = arguments.callee.caller.arguments[0].target || window.event.srcElement;

則 FireFox不會出錯，因為FireFox的正確式寫在前面

IE 則會出現「無法取得屬性'target'的值:物件為null或未經定義」的錯誤

(2-2) 如果合併成

```
var a = window.event.srcElement; | arguments.callee.caller.arguments[0].target
```

IE 不會錯，因為這次是IE的正確式寫在前面

FireFox 就會出現「TypeError: window.event is undefined」的錯誤

(2-3)也就是一定要先得得到正確表示的「事件」，再利用該「事件」來讀取「觸發事件的源頭」，一定要分兩步，不能合成一步

javascript 真是個複雜混亂的世界

Edited 7 time(s). Last edit at 08/14/2012 11:53PM by RandomVariable.
