

jQuery 篩選元素 (Traversing)

先前談到的選取元素 (selectors) 是直接從根元素去尋找其下的後代子元素，而我們這裡要探討如何去進一步「篩選」，jQuery 的 Traversing 能幫你作這些過濾、查訪元素的動作。

Traversing也就是從某元素進一步去找出跟它有關係 (關連) 的元素。

過濾元素 (Filtering)

jQuery 有提供一些函數幫助我們方便的「濾出」我們要的目標元素：

取得第 **index** 個元素 (**index**從0開始)

eq(index)

例：取得匹配的第3個元素

```
$( "p" ).eq( 2 );
```

相較於get(index)得到的是DOM物件；eq(index)則是jQuery物件。

找出所有符合表達式條件的元素 (可用逗號「,」分開多個 **expr**)

filter(expr)

例：取得類別為highlight的所有段落元素

```
$( "p" ).filter( ".highlight" );
```

刪除所有符合表達式條件的元素 (**expr** 為一個表達式或一個(組) **DOM** 元素)

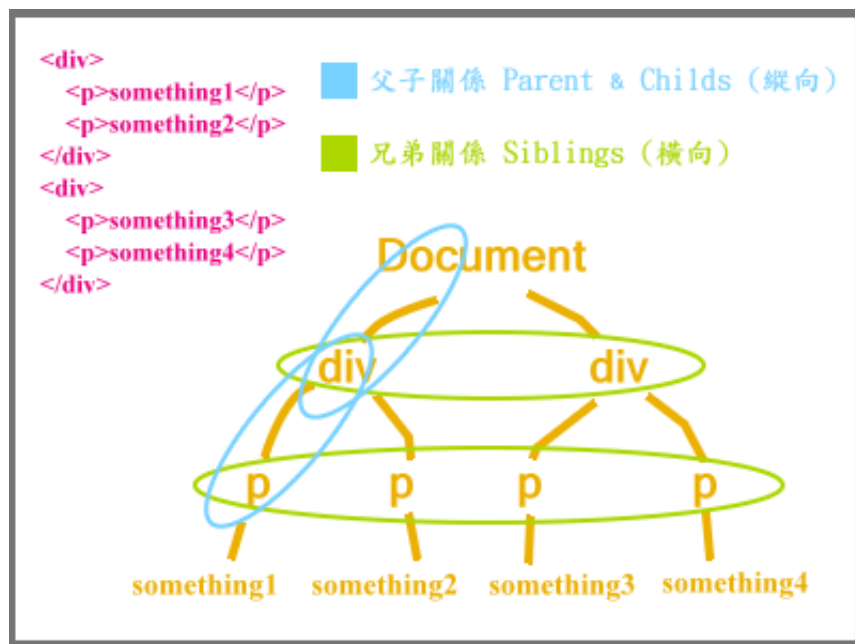
not(expr)

例：刪除掉類別為green的及id為blueone的元素

```
$( "div" ).not( ".green, #blueone" );
```

元素（節點）間位置的相互關係

在談如何查訪元素之前，我先用一張簡單的示意圖來說明元素節點間相互的空間關係：



依縱向關係來查訪 (Finding)

```
/* 取得上一階層的父元素 */
$(expression).parent([expr]);

/* 取得全部的父元素集合 (祖先元素) */
$(expression).parents([expr]);

/* 取得(僅)下一階層的所有子元素之集合 (不含text nodes) */
$(expression).children([expr]);

/* 取得全部的子元素 (含text nodes)
   若selectors是個iframe，則contents會取得其document物件 */
$(expression).contents([expr]);
```

依橫向關係來查訪 (Finding)

```
/* 取得其後緊鄰的兄弟元素（同輩元素） */
$(expression).next([expr]);

/* 取得從下一個直到最後一個同輩元素 */
$(expression).nextAll([expr]);

/* 前一個同輩元素 */
$(expression).prev([expr]);

/* 從前一個直到最開頭的同輩元素 */
$(expression).prevAll([expr]);

/* 取得其所有同輩元素的集合 */
$(expression).siblings([expr]);
```

依表達式條件來查訪元素 (Finding)

```
find(expr)
```

例如我們想取得段落下的span元素：

```
/* HTML */<p><span>Hello</span> World</p> /* jQuery codes */
$("p").find("span");
/* 得到的結果 */
[<span>Hello</span>]
```

你也可以寫成下面這樣的型式，其意思同find()。其中，第一個參數為查訪條件式；第二個參數則為jQuery或DOM物件：

```
$(expression,[context]);
```