



Chapter 09

jQuery選擇器與CSS 和DOM

中原大學 資訊管理學系

賴錦慧 老師

chlai@cycu.edu.tw

大綱

- jQuery選擇器與包裝者
- 基本CSS選擇器
- 篩選選擇器
- 屬性選擇器
- jQuery與CSS
- jQuery與DOM處理

jQuery選擇器與包裝者



- jQuery包裝者

- jQuery選擇器

jQuery包裝者-說明



- jQuery函數庫提供一個名為jQuery的物件，即包裝者物件
 - 包裝其他物件來提供這些物件擴充方法的新功能。
- 包裝者物件（ Wrapper Object ）是一種物件導向設計樣式
 - 目的是讓包裝者物件提供全新介面
 - 所有操作都是使用jQuery包裝者的方法
 - 呼叫包裝者方法來處理被它包裝的物件。

jQuery包裝者-新增DOM元素



- jQuery物件可以包裝各種不同類型的物件，至於jQuery物件可以作什麼需視其包裝的物件而定
- 例如：包裝一個HTML標籤片斷，且新增至HTML網頁的最後：

```
$('#<p>建立DOM元素</p>').appendTo('body');
```

 - 程式碼使用jQuery的`appendTo()`方法將此片斷新增至body元素的最後一個子元素。

jQuery包裝者-更改DOM元素



- ❑ 除了在HTML網頁新增DOM元素外，需要處理存在的DOM元素
- ❑ jQuery包裝者允許包裝存在的網頁元素，使用的是`$()`函數參數的選擇器
- ❑ 例如：基本CSS選擇器
 - `$('p').addClass('blue');`
 - 選擇器可以包裝網頁所有<p>元素，並且套用.blue類別的CSS樣式。

jQuery選擇器-組成元素



- jQuery選擇器主要有三種基本組成元素
 - 標籤名稱 (Tag Name)
 - id屬性
 - class屬性
- 可以任意組合這些元素來建立選擇器
- 選擇器與CSS之間的關係說明:

選擇器	CSS	jQuery	說明
標籤名稱	p { }	\$('p')	選擇所有p元素
id屬性	#list { }	\$('#list')	選擇id屬性為list的元素
class屬性	.item { }	\$('.item')	選擇class屬性為item的元素

jQuery選擇器-種類

□ 基本CSS選擇器 (Basic CSS Selectors)

- 稱為找尋選擇器 (Find Selectors)，可以在HTML網頁的DOM之中，找出所需的DOM元素

□ 位置選擇器 (Positional Selectors)

- 基於元素之間位置關係的選擇器，可以新增至基本CSS選擇器來依據位置進行元素篩選

□ jQuery選擇器 (Custom jQuery Selectors)

- jQuery本身提供的選擇器，如同位置選擇器，它也是新增至基本CSS選擇器來進行元素篩選

□ 位置選擇器, jQuery選擇器 ==> 篩選選擇器(Filter selectors)

基本CSS選擇器



- 使用CSS選擇器選擇元素
- 使用多個類別名稱來選擇元素
- 父子關係選擇器
- 同時選擇多種不同類型的元素

使用CSS選擇器選擇元素- 選擇所有元素



- 如果你需要選擇所有DOM元素，就可以使用「*」星號選擇器：

```
$('*').addClass('line');
```

- 程式碼在選擇所有元素後，呼叫addClass()方法替選擇元素加上CSS樣式類別line。



使用CSS選擇器選擇元素- 使用HTML標籤名稱選擇元素

□ 在HTML網頁可以使用HTML標籤名稱來選擇DOM元素

■ 例如：p、div、span、a和h2等：

```
$('p').addClass('blue');
```

■ 程式碼使用<p>標籤名稱'p'，可以選擇網頁中所有HTML標籤名稱為<p>的DOM元素

■ 相當於執行JavaScript的`getElementsByTagName()`方法：
`document.getElementsByTagName('p');`

使用CSS選擇器選擇元素-

使用id屬性選擇元素

- 對於擁有id屬性的HTML標籤來說，可以使用id屬性選擇元素，即「#」選擇器加上屬性值：

```
$('#list').addClass('red');
```

- 程式碼選擇id屬性值為list的DOM元素，相當於是執行JavaScript的getElementById()方法：

```
document.getElementById('list');
```

使用CSS選擇器選擇元素-

使用class屬性選擇元素



- 對於擁有class屬性的HTML標籤來說，**可以使用class屬性選擇元素**，即「**.**」選擇器加上屬性值：

```
$('.item').addClass('green');
```

- 程式碼選擇class屬性值為item的DOM元素，相當於是執行JavaScript的**getElementsByClassName()**方法：

```
document.getElementsByClassName('item');
```

Example

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8"/>
<title>Ch9_2_1.html</title>
<style type="text/css">
.blue { color: blue; }
.red { color: red; }
.green { color: green; }
.line { border: 1px solid #333; }
</style>
<script src="jquery-
2.1.1.min.js"> </script>
<script>
$(document).ready(function() {
    $('*').addClass('line');
    $('p').addClass('blue');
    $('#list').addClass('red');
    $('.item').addClass('green');
});
</script>
</head>
```

```
<body>
<p>JavaScript網頁設計</p>
<p>jQuery網頁設計</p>
<ol id="list">
    <li>jQuery</li>
    <li class='item'>Prototype</li>
    <li>MooTools</li>
    <li class='item'>DOJO</li>
</ol>
</body>
</html>
```

JavaScript網頁設計

jQuery網頁設計

1. jQuery
2. Prototype
3. MooTools
4. DOJO

使用多個類別名稱來選擇元素

- 同一個HTML標籤的class屬性可以套用多個類別名稱
- 使用class屬性選擇元素時，可以同時使用多個類別名稱：
`$('.item.inactive').addClass('hide');`
 - 選擇器需要符合class屬性值有item和inactive時才符合條件。

Example

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8"/>
<title>Ch9_2_2.html</title>
<style type="text/css">
.hide { display: none; }
</style>
<script src="jquery-
2.1.1.min.js"> </script>
<script>
$(document).ready(function() {
    $('.item.inactive').addClass('hide');
});
</script>
</head>
```

```
<body>
<ul id="list">
  <li class='item inactive'>jQuery</li>
  <li class='item active'>Prototype</li>
  <li class='item inactive'>MooTools</li>
  <li class='item active'>DOJO</li>
  <li class='item active'>YUI</li>
</ul>
</body>
</html>
```

- Prototype
- DOJO
- YUI

父子關係選擇器

- 父子關係選擇器就是第7章的CSS巢狀選擇器，
可以選擇HTML標籤之下的特定HTML元素：

父子關係選擇器	jQuery範例	範例說明
element element	<code>\$('div p')</code>	選擇所有父元素是div元素，p是其 <u>子孫元素</u>
element>element	<code>\$('div > p')</code>	選擇所有父元素是div元素的 <u>子元素</u>
element+element	<code>\$('div + span')</code>	選擇所有緊接著div元素之後的span <u>兄弟元素</u>

Example

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8"/>
<title>Ch9_2_3.html</title>
<style type="text/css">
.blue { color: blue; }
.red { color: red; }
.green { color: green; }
</style>
<script src="jquery-
2.1.1.min.js"> </script>
<script>
$(document).ready(function() {
    $('div p').addClass('blue');
    $('div > p').addClass('red');
    $('div + span').addClass('green');
});
</script>
</head>
```

```
<body>
<div>
    <p>JavaScript</p>
</div>
<span> <p>jQuery</p> </span>
<span> <p>Flash</p> </span>
<div>
    <p>ASP.NET</p>
    <b> <p>ASP</p> </b>
    <span> <p>PHP</p> </span>
    <span> <p>JSP</p> </span>
</div>
</body>
</html>
```

JavaScript

jQuery

Flash

ASP.NET

ASP

PHP

JSP

同時選擇多種不同類型的元素

- 如果需要同時選擇多種不同類型的元素，我們可以同時使用標籤名稱、id屬性、class屬性等多個不同的選擇器：

```
$('.item-3, .item-5, #book6, p').addClass('blue');
```

- 程式碼使用多種選擇器，只需使用「,」逗號分隔就可以選擇多種不同種類的元素

- 如果HTML標籤擁有id或class屬性，可以組合標籤名稱、id屬性、class屬性來選擇特定的DOM元素：

```
$('#li#book1').addClass('red');
```

```
$('#li.item-2').addClass('green');
```

- 選擇器是選擇id屬性值為book1的li元素；class屬性值為item-2的li元素。



篩選選擇器

- 使用EVEN和ODD篩選選擇器
- 使用FIRST和LAST篩選選擇器
- 篩選空元素和特定元素
- 篩選包含特定內容的元素

篩選選擇器

- 篩選 (Filter) 可以讓我們進一步過濾選取的DOM元素
- jQuery提供數十種篩選，包含CSS 2和3的篩選屬性（詳見第7章），其基本語法如下所示：

E:filter

- E是我們選擇的元素，在「:」符號之後就是篩選屬性。

EVEN和ODD篩選選擇器



- EVEN和ODD篩選屬性可以在表格加上斑馬紋效果，替奇數或偶數列套用不同的樣式：

```
$('tr:even').addClass('even');
```

```
$('tr:odd').addClass('odd');
```

- 程式碼使用:even替偶數的tr元素套用even樣式類別；:odd是替奇數的tr元素套用odd樣式類別。

FIRST和LAST篩選選擇器

- **FIRST和LAST篩選選擇器**可以從選擇的DOM元素集合中，取出:**first**的第1個元素，和:**last**取出最後1個元素，篩選只會傳回1個元素：

```
$('ol li:first').addClass('line');
```

```
$('ol li:last').addClass('line');
```

- 程式碼使用:**first**取得第1個li元素；**:last**取得最後1個li元素。

篩選空元素和特定元素

- 如果需要篩選出元素之中擁有特定元素時，可以使用`has()`篩選：

```
$('.header:has(p)').addClass('line');
```

- 程式碼可以篩選出`class`屬性值為`header`，而且之中擁有`p`元素，只需擁有即可，不一定需要是直接子元素

- **篩選空元素**：即元素沒有子元素或文字內容

```
$('.error:empty').addClass('hide');
```

- 程式碼使用：**empty篩選**，可以篩選出`class`屬性值為`error`且沒有內容的空元素

Example

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8"/>
<title>Ch9_3_3.html</title>
<style type="text/css">
.hide { display: none; }
.line { border: 1px solid #333; }
</style>
<script src="jquery-
2.1.1.min.js"></script>
<script>
$(document).ready(function() {
    $('.header:has(p)').addClass('line');
    $('.error:empty').addClass('hide');
});
</script>
</head>
```

```
<body>
<div>
    <div class="header">
        <p>JavaScript網頁設計</p>
    </div>
    <div class="header">
        <span>jQuery網頁設計</span>
    </div>
    <div class="header">
        <span><p>Google Chrome<p></span>
    </div>
    <div class="error"></div>
    <div class="error">顯示錯誤訊息...</div>
</div>
</body>
</html>
```

JavaScript網頁設計

jQuery網頁設計

Google Chrome

顯示錯誤訊息...



篩選包含特定內容的元素

- 如果需要篩選包含特定內容的元素，可以使用 **contains()** 篩選：

```
$("li:contains('ASP')").addClass('line');
```

- 程式碼可以篩選出li元素包含'ASP'字串內容。



屬性選擇器

- 選擇包含指定網址的超連結
- 選擇id屬性值是特定開頭或結尾的元素

選擇包含指定網址的超連結

- 屬性選擇器是使用正規運算式 (Regular Expressions) 的符號來表示包含字串 (*)、開頭字串 (^) 和結尾字串 (\$) 等。
- 使用「 * 」屬性選擇器在HTML元素的屬性值中找出包含特定內容。
- 例如：台灣網址最後有 ".tw" :
`$('ol li a[href*=".tw"]').addClass('line');`
 - `[href*=".tw"]` 的屬性選擇器是使用 `*`，表示 `href` 屬性值只需包含等號之後的值就符合條件的HTML元素。

選擇id屬性值是特定開頭或結尾的元素

- 使用「^」和「\$」屬性選擇器，找出屬性值是特定值開頭或結尾：

```
$('div[id^="red"]').addClass('line');
```

- [id^="red"]的屬性選擇器是使用^=，表示id屬性值只需是等號之後的"red"開頭就符合條件，例如：red-pen、red-bird和red-book等。

```
$('div[id$="bird"]').addClass('blue');
```

- 屬性選擇器「\$=」是指id屬性值是"bird"結尾，例如：red-bird和blue-bird等。

Example: Ch9_4_2.html

練習1

□ 請使用ch09-filter.html, 寫出適合的選擇器內容

例如: 找出id值為 bigbang 的元素 → \$('#bigbang')

- 1) 找出超連結網址開頭為CSS的元素
- 2) 找出所有有id值的元素
- 3) 找出被選取的服務 (多選)
- 4) 找出包含有超連結的清單

jQuery與CSS

- 存取CSS

- 存取CSS樣式類別

存取CSS-讀取

- 使用 **css()** 方法取出指定元素的樣式屬性：

```
var fonSize = $('p.title').css('font-size');  
alert(fonSize);
```

- 程式碼使用 **css()** 方法取得 **p** 元素的字型尺寸，參數 **font-size** 是 CSS 樣式屬性名稱，然後呼叫 **JavaScript** 函數 **alert()** 顯示取得的字型尺寸。

https://www.w3schools.com/jquery/jquery_css.asp

存取CSS-設定

- 設定CSS樣式屬性也是使用`css()`方法：

```
$('h2').css('background-color', 'red');
```

- 程式碼是使用`css()`方法設定h2元素的樣式，第1個參數是樣式屬性名稱，第2個是屬性值。

- 設定多個樣式屬性&屬性值(使用JavaScript的“物件文字值(Object Literal)”)

```
$('div').css({  
    'background-color': '#dddddd',  
    'color': '#666666',  
    'font-size': '12pt',  
    'line-height': '2.5em'  
});
```

存取CSS樣式類別-新增

- 在選擇元素可以使用`addClass()`方法新增樣式類別：

```
$('h2').addClass('littlered');
```

- 程式碼是在h2元素新增CSS樣式類別littlered。

存取CSS樣式類別-移除

- 對於HTML元素已經套用的CSS樣式類別，可能有很多個，我們可以使用`removeClass()`方法來移除元素的樣式類別
- 方法沒有參數，表示移除div元素的所有樣式類別
`$('.div').removeClass();`
- 移除特定樣式類別，要使用參數指明：
`$('.div').removeClass('littlered');`
- 可以同時移除多個樣式類別：
`$('.div').removeClass('littlered product red');`
 - 程式碼共移除div元素的littlered、product和red三個樣式類別，只需使用空白字元分隔即可。

存取CSS樣式類別-新增或刪除

- 在建立動態網頁內容時，可能需要切換元素的樣式類別，此時就可以使用toggleClass()方法：

`$('#p').toggleClass('littlegreen');`

- 程式碼是當p元素沒有套用樣式類別，就套用，反之，如果有，就移除。

Example:

https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_dom_toggleclass

練習2

□ 請開啟ch09-ex2.html, 利用jQuery產生下圖結果

jQuery introduction

- jQuery supports
 - CSS1
 - CSS2
 - CSS3
 - Basic XPath
- jQuery also supports
 - Custom selectors
 - Form selectors

- 1) 設定為紅色
- 2) 加上外框
- 3) 前3個項目為斜體

設定為藍色

jQuery與DOM處理



- 在網頁新增和刪除DOM元素
- 取得jQuery包裝類別的DOM元素

在網頁新增和刪除DOM元素-方法



□ 在網頁新增和刪除DOM元素的相關方法說：

方法	說明
html()	即JavaScript的innerHTML屬性，可以將參數的HTML標籤取代DOM元素的內容
text()	即JavaScript的innerText屬性，可以將參數的文字內容取代DOM元素的內容
prepend()	將參數的HTML標籤新增成為其第1個子元素
append()	將參數的HTML標籤新增成為其最後1個子元素
before()	將參數的HTML標籤插入至元素之前
after()	將參數的HTML標籤插入至元素之後
remove()	從DOM之中刪除此元素

在網頁新增和刪除DOM元素- 將HTML標籤和文字新增至DOM元素



- 對於選取的DOM元素，可以使用html()和text()方法取代其內容
 - html()方法是取代成另一個HTML子元素
 - text()方法是取代文字內容

```
$('.content1').html('<div class="main">jQuery網頁設計</div>');
```

```
$('.content2').text('HTML網頁設計');
```

- 程式碼是將class屬性值content1的內容取代成參數的div元素；content2的內容取代成參數的文字內容。

在網頁新增和刪除DOM元素- 新增第1個或最後1個子元素



- 如果不是取代子元素或文字內容，可以使用 **prepend()** 或 **append()** 方法來新增第1個或最後1個子元素：

```
$('.content2').append('<p>PHP網頁設計</p>');
```

- 程式碼在class屬性值content2的元素中，新增最後一個p子元素；如果使用prepend()方法就是新增成為第1個p子元素。

在網頁新增和刪除DOM元素-插入



- 使用 **before()** 和 **after()** 方法在選取DOM元素之前或之後來插入新元素：

`$('.content2').after('<p>JSP網頁設計</p>');`

- 程式碼在class屬性值content2的元素之後插入p兄弟元素；如果使用before()方法就是在之前插入p兄弟元素。

在網頁新增和刪除DOM元素-刪除



- 在選取後，使用`remove()`方法來刪除不需要的DOM元素：

`$('.content3').remove();`

- 程式碼刪除class屬性值content3的元素。

Example- jQuery & DOM

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8"/>
<title>Ch9_6_1.html</title>
<script src="jquery-3.1.1.min.js"></script>
<script>
$(document).ready(function() {
    $('.content1').html('<div class="main">jQuery網頁設計</div>');
    $('.content2').text('HTML網頁設計');
    $('.content2').append('<p>PHP網頁設計</p>');
    $('.content2').after('<p>JSP網頁設計</p>');
    $('.content3').remove();
});
</script>
</head>
<body>
<div class="content1">
    <p>JavaScript網頁設計</p>
</div>
<div class='content2'></div>
<div class='content3'>ASP.NET網頁設計</div>
</body>
</html>
```

jQuery網頁設計
HTML網頁設計

PHP網頁設計

JSP網頁設計

取得jQuery包裝類別的DOM元素-取得DOM元素數

- jQuery物件可以使用length屬性取得包裝的DOM元素數：

```
var length = $("li:contains('ASP')").length;
```

- 程式碼使用length屬性取得共有多少個DOM元素被包裝在jQuery物件。

取得jQuery包裝類別的DOM元素-直接存取DOM元素

- 對於取回的jQuery物件，因為被包裝的DOM元素可能不只一個，使用`get()`方法來取出指定的DOM元素

```
var myTag1 = $('#list').get(0).tagName;
```

- 程式碼取得第1個元素，即參數值0（索引值是從0開始），然後就可以使用`tagName`屬性取得此DOM元素的標籤名稱

- 為了方便取得DOM元素，可以使用類似陣列索引方式來取得DOM元素：

```
var myTag2 = $('ol, p')[1].tagName;
```

練習3

□ 開啟ch09-ex3.html

- 1) 在List清單中“Sheldon”節點之前與後各新增1個節點
- 2) 移除“test”節點

<原始檔>

the big bang

- lenoard
- Sheldon
- Raj
- Leslie
- ~~test~~



<處理後>

the big bang

- lenoard
- Penny
- Sheldon
- Howard
- Raj
- Leslie