



Chapter 06

DOM物件模型

中原大學 資訊管理學系

賴錦慧 老師

chlai@cycu.edu.tw

大綱

- DOM物件模型的基礎
- 取得HTML元素
- DOM的節點瀏覽
- HTML集合物件
- 存取HTML標籤的屬性
- DOM的節點操作

HTML DOM - 參考資料



- ❑ http://www.w3schools.com/js/js_htmldom.asp
- ❑ <http://pydoing.blogspot.tw/2011/08/javascript-htmldom-overview.html>
- ❑ <http://blog.kkbruce.net/2012/02/javascript-w3c-dom.html#.VmkVRr8WWsY>
- ❑ <http://www.wibibi.com/info.php?tid=379>



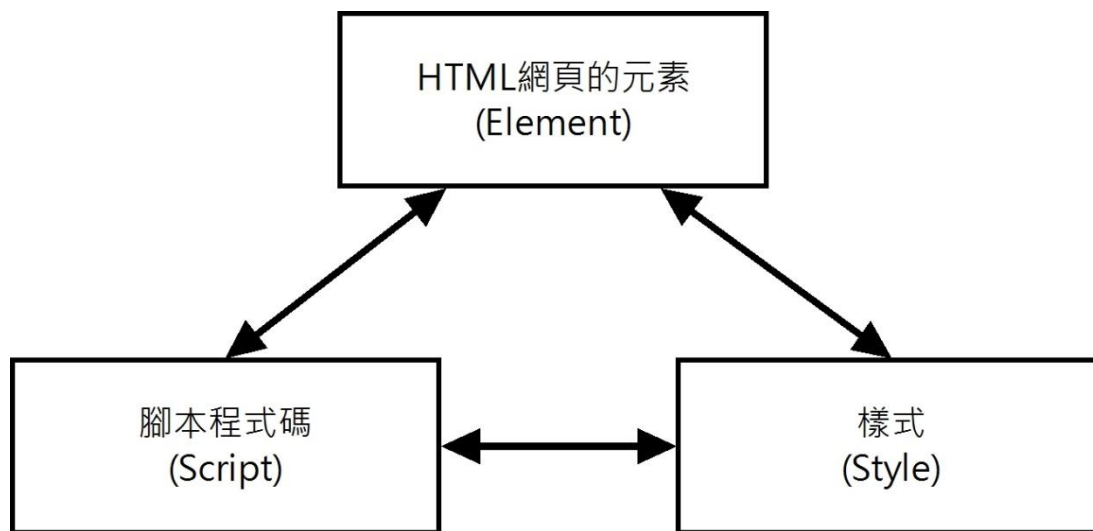
DOM物件模型的基礎

- 物件模型與DOM
- DOM基礎的HTML網頁內容

物件模型與DOM-說明

□ DOM (Document Object Model)

- 文件物件化
- 規範如何存取HTML元素、樣式或程式碼的機制
- 將HTML元素、樣式和程式碼視為物件，和定義之間的關係



物件模型與DOM-簡介



- DOM物件模型提供一組標準程式介面，用來存取物件的屬性和方法
 - 瀏覽HTML網頁或XML文件
 - 新增、刪除和修改節點資料

- DOM主要是由兩大部分組成：
 - **DOM Core**: 提供HTML網頁或XML文件瀏覽、處理和維護階層架構
 - 瀏覽 (Navigator) : 能夠在網頁的樹狀架構中走訪節點。
 - 參考 (Reference) : 能夠存取節點的集合物件。
 - **DOM HTML** : HTML網頁專屬的DOM API介面
 - 將網頁元素都視為是一個個物件，以便讓JavaScript等程式語言存取元素來建立動態網頁內容。

物件模型與DOM-規格



- DOM Level 0：這並不是真正存在的規格，它是建立DOM Level 1規格的基礎，目前瀏覽器支援Level 0的主要目的只是為了和舊版相容。
- DOM Level 1：在1998年成為W3C（World Wide Web Consortium）的建議規格，提供HTML網頁和XML文件通用的瀏覽和處理元素內容的物件模型，不過，在Level 1並沒有提供事件處理機制。
- DOM Level 2：在2000年成為W3C的建議規格，除了DOM2 Core和DOM2 HTML外，還支援事件處理機制的DOM2 Events等6項規格。
- DOM Level 3：在2004年成為W3C的建議規格，Level 3包含5種規格，即DOM3 Core、Load and Save、Validation、Events和XPath。

物件模型與DOM-優點



□ 提供跨平台和程式語言的程式介面

- DOM提供應用程式標準的程式處理介面，這是一種HTML和XML文件的標準API。

□ 支援多種文件

■ DOM支援HTML網頁和XML文件

- DOM Core可以使用在HTML網頁和XML文件
- DOM HTML是針對HTML網頁。

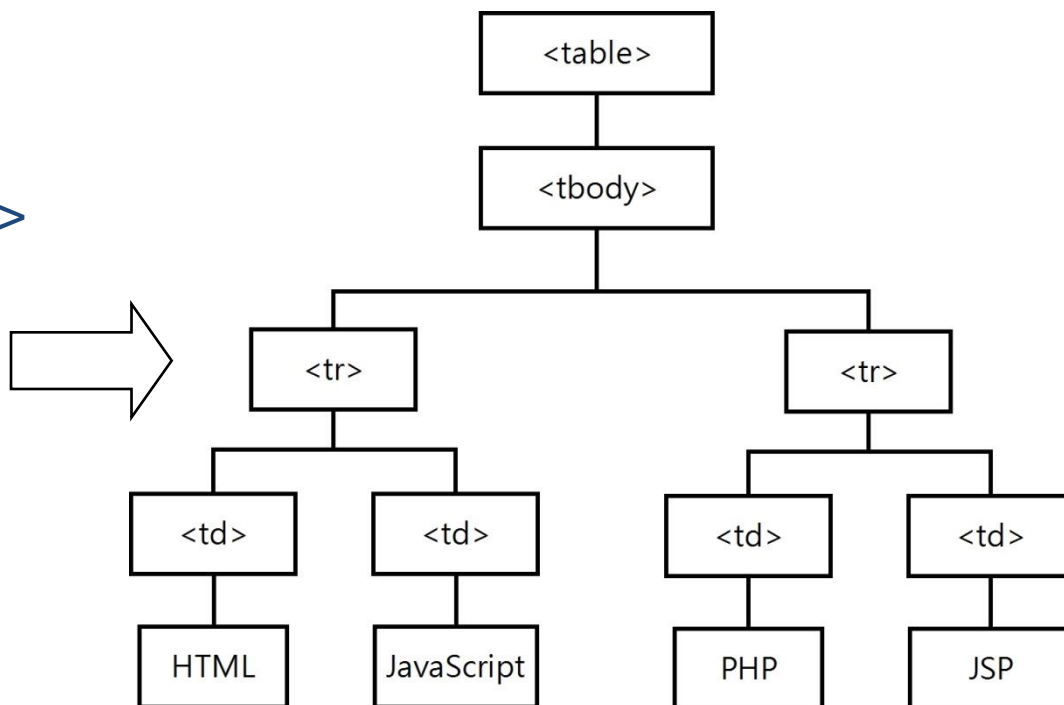
□ 支援多種程式語言

- DOM支援多種程式語言和腳本語言，例如：Java、PHP、.NET語言和ECMAScript（即JavaScript）等。

DOM基礎的HTML網頁內容

- DOM是一種結構化文件內容的物件模型，可以將HTML網頁的標籤和文字內容視為「節點」(Node)，依各節點之間的關係連接成樹狀結構
- 例如：<table>表格標籤片斷

```
<table>
  <tbody>
    <tr>
      <td>HTML</td>
      <td>JavaScript</td>
    </tr>
    <tr>
      <td>PHP</td>
      <td>JSP</td>
    </tr>
  </tbody>
</table>
```





取得HTML元素節點

- 使用**Id屬性**取得元素節點
- 使用**標籤名稱**取得元素節點
- 取得與更改元素內容
- 存取HTML元素的尺寸與位置

使用Id屬性取得元素節點

getElementById()方法

□ 取出HTML網頁指定的HTML元素，和傳回節點物件的參考

□ 它是使用參數的id屬性值來取得指定的元素

```
var a = document.getElementById("google");
```

■ 程式碼的document可以取得HTML網頁的DOM

■ 方法的參數是<a>標籤的id屬性值

➔ 就是取回此a元素的節點物件

```
<p><a id="google"
```

```
href="http://www.google.com.tw">Google</a></p>
```

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8"/>
    <title>Ch6_2_1.html</title>
    <script>
      function showAElement() {
        var a = document.getElementById("google");
        alert(a.nodeName + " - " + a.href);
      }
      function showButtonElement() {
        var btn = document.getElementById("test");
        alert(btn.tagName + " - " + btn.type);
      }
    </script>
  </head>
  <body>
    <h2>使用Id屬性取得元素節點</h2>
    <hr/>
    <p id="content">使用Id屬性取得元素節點</p>
    <p><a id="google"
      href="http://www.google.com.tw">Google</a> </p>
    <input id="test" type="button"
      onclick="showAElement()" value="選取<a>元素">
    <input type="button"
      onclick="showButtonElement()"
      value="選取<input>元素">
  </body>
</html>

```

使用Id屬性取得元素節點

使用Id屬性取得元素節點

Google

選取<a>元素

選取<input>元素



使用標籤名稱取得元素節點



nodeName與tagName屬性

- 在取得指定節點物件後，可以使用**nodeName**或**tagName**屬性取得節點的標籤名稱，傳回的是大寫的標籤名稱字串

```
alert(a.nodeName + " - " + a.href);
```

```
alert(btn.tagName + " - " + btn.type);
```

- href和type屬性就是HTML標籤的原生屬性，在取得指定節點物件後，就可以存取這些屬性值

使用標籤名稱取得元素節點-方法



getElementsByTagName()方法

- 傳回指定標籤名稱的節點陣列或清單，它是一個NodeList物件

```
var p = document.getElementsByTagName("p");
```

- 程式碼取回HTML網頁中所有<p>標籤的節點
- 可能有多個同名標籤，所以傳回的不是單一節點物件，而是NodeList物件，可以使用length屬性取得共有幾個節點物件

```
alert("P元素有: " + p.length + "個");
```

使用標籤名稱取得元素節點-item()

使用item()方法取得指定的節點物件

□ 使用item()方法來取得指定的節點物件

```
var f = document.getElementsByTagName("input");
for (var i = 0; i < f.length; i++) {
    objEle = f.item(i);
    strTags += objEle.type + " ";
}
```

使用標籤名稱取得元素節點-索引



使用陣列索引方式取得節點物件

- 可以直接將**NodeList**物件當成陣列來取出指定索引值的節點物件

```
var f=document.getElementsByTagName("input");  
for (var i = 0; i < f.length; i++)  
    strTags += f[i].type + " ";
```


Example



HTML元素H2

HTML元素P

HTML元素P

HTML元素H2

☐ ☐

取得P元素

取得H2元素

顯示INPUT元素

顯示INPUT元素

P元素有: 2個

確定

text radio checkbox button button button button

☐ 避免此頁面產生更多對話框

確定

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8"/>
<title>Ch6_2_2.html</title>
<script>
function getEle1() {
    var p = document.getElementsByTagName("p");
    alert("P元素有: " + p.length + "個");
}
function getEle2() {
    var h = document.getElementsByTagName("h2");
    alert("H2元素有: " + h.length + "個");
}
function showItems1() {
    var strTags = "";
    var f = document.getElementsByTagName("input");
    for (var i = 0; i < f.length; i++) {
        objEle = f.item(i);
        strTags += objEle.type + " ";
    }
    alert(strTags);
}
function showItems2() {
    var strTags = "";
    var f = document.getElementsByTagName("input");
    for (var i = 0; i < f.length; i++)
        strTags += f[i].type + " ";
    alert(strTags);
}
</script>
```

```
</head>
<body>
<h2>HTML元素H2</h2>
<hr/>
<p>HTML元素P</p>
<p>HTML元素P</p>
<h2>HTML元素H2</h2>
<form>
    <input type="text"/>
    <input type="radio"/>
    <input type="checkbox"/>
</form>
<input type="button" onclick="getEle1()" value="取得P元素">
<input type="button" onclick="getEle2()" value="取得H2元素">
<input type="button" onclick="showItems1()" value="顯示INPUT元素">
<input type="button" onclick="showItems2()" value="顯示INPUT元素">
</body>
</html>
```

練習1

請在ch06_ex1.html加入JS程式碼

(1)請使用id取得段落P的節點名稱與節點內容

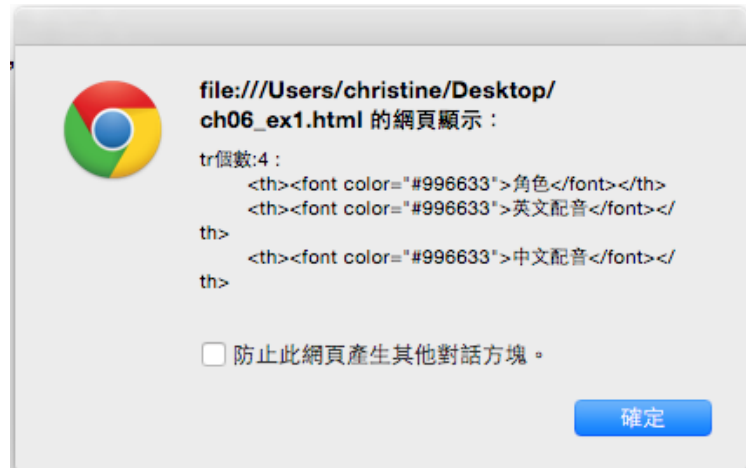
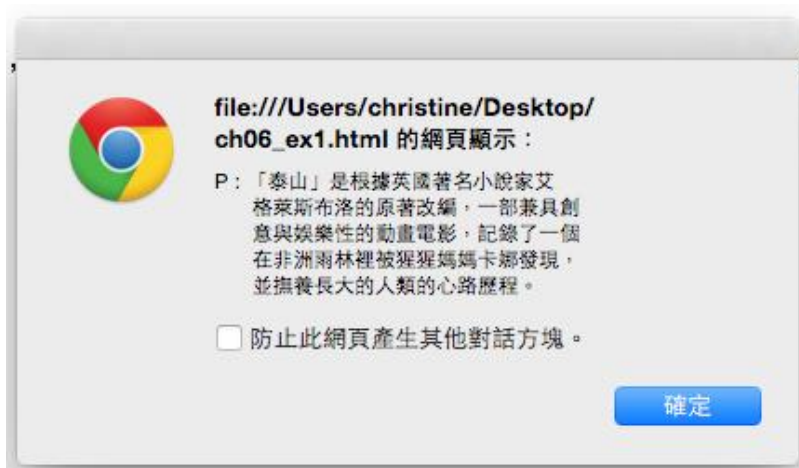
(2)請使用tag取得"tr"標籤的個數、與第1個"tr"標籤的內容

角色	英文配音	中文配音
泰山	東尼高德溫(第六感生死戀)	金城武
珍妮	敏妮卓芙(心靈捕手)	楊采妮
卡娜	葛倫克蘿絲(101真狗)	張艾嘉

「泰山」是根據英國著名小說家艾 格萊斯布洛的原著改編，一部兼具創 意與娛樂性的動畫電影，記錄了一個 在非洲雨林裡被猩猩媽媽卡娜發現， 並撫養長大的人類的心路歷程。

使用id取得p節點

使用TagName取得tr



取得與更改元素內容

- 在取得指定的DOM節點物件後，可以使用 **innerHTML** 屬性存取HTML元素的文字內容(HTML 5版支援的屬性)

```
var c = document.getElementById("content");  
alert(c.innerHTML);  
c.innerHTML = "JavaScript和DOM"; //更改內容
```

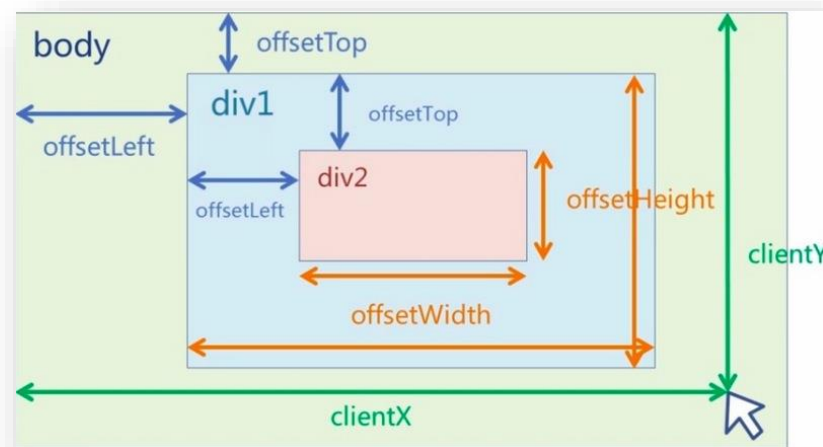
存取HTML元素的尺寸與位置

□ DOM節點物件提供屬性來取得HTML標籤的尺寸和位置

- offsetLeft屬性：節點物件距離左方邊界的距離。
- offsetTop屬性：節點物件距離上方邊界的距離。
- offsetHeight屬性：節點物件的高。
- offsetWidth屬性：節點物件的寬。
- offsetParent屬性：取得節點物件的上一層物件。

• 節點物件還提供方法來自動捲動瀏覽器視窗

- scrollToView()方法：如果瀏覽器視窗看不到指定HTML元素，就自動捲動視窗顯示此元素。



Example



存取HTML元素的位置和尺寸

存取HTML元素的位置和尺寸

元素位置

元素尺寸

顯示父元素

顯示段落

存取HTML元素的位置和尺寸1

存取HTML元素的位置和尺寸2

存取HTML元素的位置和尺寸3

存取HTML元素的位置和尺寸4

存取HTML元素的位置和尺寸5

left:8 top:88

確定

20 X 1712

☐ 避免此頁面產生更多對話框

確定

Example

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8"/>
<title>Ch6_2_4.html</title>
<script>
function showPosition() {
    var p = document.getElementById("myP");
    alert("left:" + p.offsetLeft + " top:" + p.offsetTop);
}
function showSize() {
    var p = document.getElementById("myP");
    alert(p.offsetHeight + " X " + p.offsetWidth);
}
function showParent() {
    var p = document.getElementById("myP");
    alert(p.offsetParent.tagName);
}
function viewElement() {
    var objEle = document.getElementsByTagName("p");
    if (objEle.length > 4 )
        objEle[3].scrollIntoView(true);
}
</script>
</head>
```

```
<body id="myBody">
<h2>存取HTML元素的位置和尺寸</h2>
<hr/>
<p id="myP">存取HTML元素的位置和尺寸</p>
<form>
<input type="button" onclick="showPosition()"
value="元素位置">
<input type="button" onclick="showSize()" value="
元素尺寸">
<input type="button" onclick="showParent()"
value="顯示父元素">
<input type="button" onclick="viewElement()"
value="顯示段落">
</form>
<p>存取HTML元素的位置和尺寸1</p>
<p>存取HTML元素的位置和尺寸2</p>
<p>存取HTML元素的位置和尺寸3</p>
<p>存取HTML元素的位置和尺寸4</p>
<p>存取HTML元素的位置和尺寸5</p>
</body>
</html>
```



DOM的節點瀏覽

- DOM瀏覽節點的相關屬性
- 瀏覽父節點
- 瀏覽兄弟節點
- 瀏覽子節點與子元素

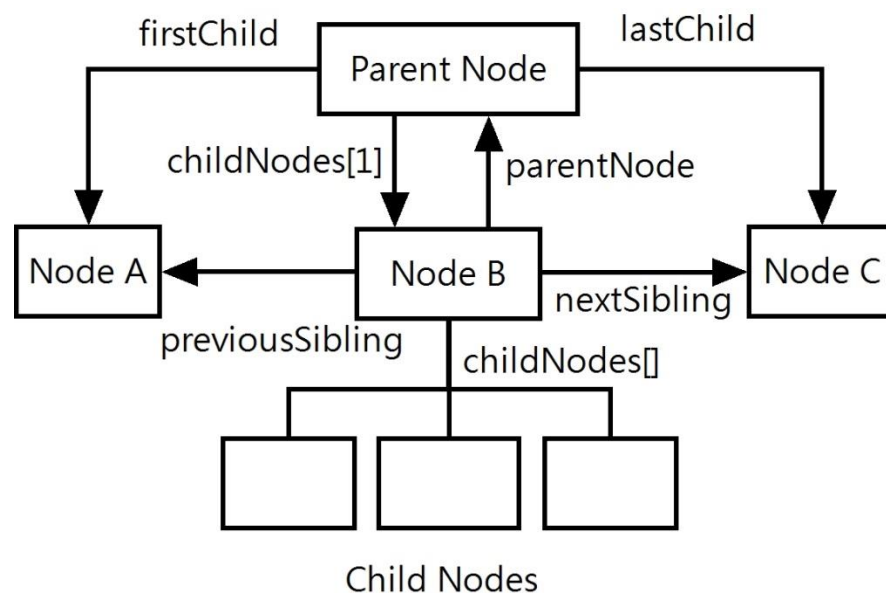
DOM瀏覽節點的相關屬性-瀏覽



屬性	說明
firstChild	傳回第1個childNodes集合物件的子節點，包含此節點下的所有子節點
lastChild	傳回最後1個childNodes集合物件的子節點，包含此節點下的所有子節點
parentNode	傳回父節點的物件
nextSibling	傳回下一個兄弟節點的物件
previousSibling	傳回前一個兄弟節點的物件
nodeName	傳回節點的HTML標籤名稱，名稱是英文大寫字母，例如：P和TABLE
nodeType	傳回節點種類，1為標籤；2為屬性；3為文字節點
specified	傳回HTML標籤是否擁有屬性的布林值

DOM瀏覽節點的相關屬性-圖例

- Parent Node使用**firstChild**屬性取得第1個子節點Node A
- **childNodes[1]**為第2個子節點Node B，**childNodes**是所有子節點的集物件，
- **lastChild**屬性可以取得最後一個子節點Node C。
- 在Node B使用**previousSibling**屬性可以取得前一個兄弟節點Node A；**nextSibling**屬性取得下一個兄弟節點Node C
- 取得上一層父節點是使用**parentNode**屬性。



DOM瀏覽節點的相關屬性-存取



□ DOM提供存取屬性來存取文字節點的內容

■ **data**屬性：存取文字節點的內容，其他種類的節點傳回undefined。

■ **nodeValue**屬性：存取文字節點的內容，其他種類的節點傳回null。

■ **Note:** 文字節點是葉節點, 不能有子節點

DOM瀏覽節點的相關屬性-集合物件



□ DOM的集合物件可以取得下一層子節點和節點屬性

- **attributes**屬性：節點屬性的集合物件，可以直接使用名稱來存取。
- **childNodes**屬性：子節點的集合物件，這是一個陣列物件，可以使用從0開始的索引值進行存取
 - `childNodes[0];`
- **children**屬性：子元素的集合物件，不包含文字節點的子節點，這也是一個陣列物件，可以使用從0開始的索引值進行存取
 - `children[0];`

瀏覽父節點

- 在DOM取得父節點是使用parentNode屬性
- 例如：項目符號的HTML標籤和，li的父節點是ul：

```
var objLi = document.getElementById("myLI");  
var objNode = objLi.parentNode;  
alert(objNode.nodeName);
```

瀏覽兄弟節點

- DOM可以使用previousSibling和nextSibling屬性取得兄弟節點
- 使用清單項目的HTML標籤和為例:

```
var objLi = document.getElementById("myLI");  
var objNode = objLi.previousSibling;  
alert(objNode.firstChild.data);
```

 - 程式碼的myLI是標籤物件，可以取得前一個兄弟節點，然後使用firstChild屬性取得下一層文字節點的data屬性（也可以使用innerHTML屬性取代）。

Example:

https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_node_previoussibling

瀏覽子節點與子元素-取得

□ DOM節點可以使用`childNodes`屬性取得子節點的集合物件，這是一個陣列物件，可以使用從0開始的索引值進行存取

- 取得父節點的節點物件

```
var objUL = document.getElementById("myUL");
```

- 使用`childNodes`屬性取得所有子節點

```
var objChilds = objUL.childNodes;
```

- 可以取得ul元素的所有子節點，包含li元素和文字節點。

瀏覽子節點與子元素-瀏覽



- 可以使用for迴圈——取得每一個節點物件且將節點名稱顯示出來

```
var strTags = "";
for (var i = 0; i < objChlds.length; i++) {
    objEle = objChlds.item(i);
    strTags += objEle.nodeName + "\n";
}
```

- 如果使用children屬性，取得的子節點就只有li子元素，而沒有文字節點。

HTML集合物件



□ DOM提供一些屬性來取得常用HTML元素的集合物件

- document.anchors屬性：包含HTML網頁所有擁有name屬性的<a>標籤。
- document.forms屬性：包含HTML網頁所有<form>標籤。
- document.images屬性：包含HTML網頁所有標籤。
- document.links屬性：包含HTML網頁所有擁有href屬性的<a>標籤。

DOM Document object

https://www.w3schools.com/jsref/dom_obj_document.asp

存取HTML標籤的屬性



- 在取得HTML網頁的節點物件後，不只可以存取元素內容，也可以存取標籤屬性

```
var p = document.getElementsByTagName("p");  
alert(p.item(0).align);
```

- 程式碼取得所有p元素後，就可以取得第1個<p>標籤的align屬性值
 - 可以存取HTML標籤的其他屬性。
-
- 在DOM節點物件提供3種方法來處理屬性：
 - `getAttribute(attribute)`方法：取得傳入參數attribute屬性的屬性值。
 - `setAttribute(attribute, value)`方法：將傳入參數attribute屬性設定為value值。
 - `removeAttribute(attribute)`方法：刪除傳入參數的attribute屬性。

練習2

- 請在ch06_ex2.html中加入JS程式碼, 使網頁可以瀏覽節點與新增節點
 - 請列出"HTML"的父節點
 - 請列出"JavaScript"的下一個兄弟節點
 - 請列出所有的子節點與文字節點內容

瀏覽節點與新增節點

- HTML
- JavaScript
- VBScript
- PHP
- XML



DOM的節點操作

- 插入和新增節點
- 刪除節點

插入和新增節點-步驟一



步驟一：建立節點物件

- 使用Document物件的createElement()方法建立節點物件
- 如果標籤擁有文字節點，還需要使用createTextNode()方法建立文字節點物件

```
var objNewNode = document.createElement("P");
```

```
var objTextNode = document.createTextNode('段落四');
```

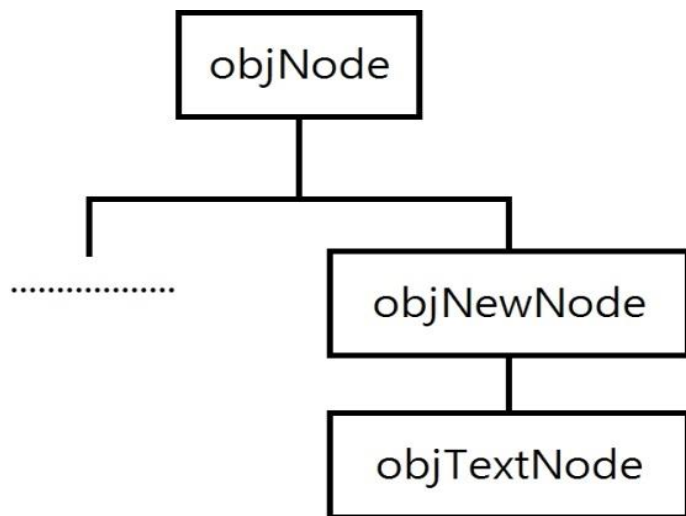
- 程式碼建立<p>標籤的節點，參數的標籤名稱字串為大寫
- 在<p>標籤擁有文字內容，所以建立文字節點，參數為文字節點的內容。

插入和新增節點-步驟二

步驟二：將節點物件新增或插入DOM節點樹

- 在建立節點物件後，就可以將節點物件新增或插入DOM節點樹

```
objNode.appendChild(objNewNode);  
objNewNode.appendChild(objTextNode);
```

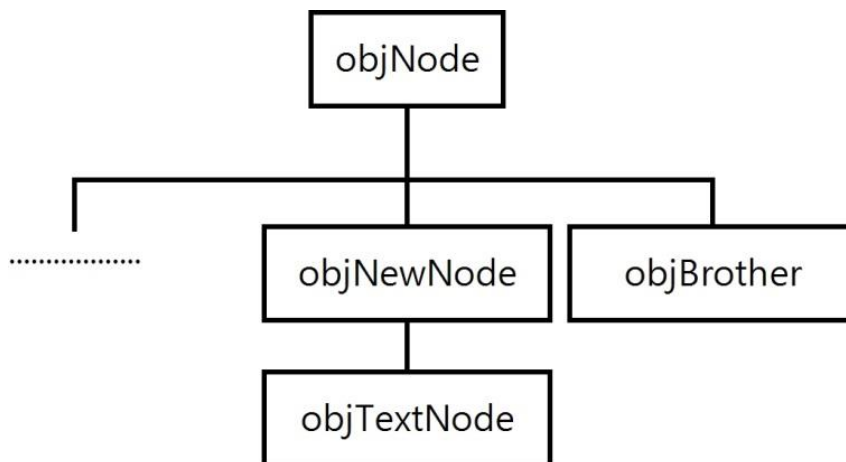


HTML DOM appendChild() Method

https://www.w3schools.com/jsref/met_node_appendchild.asp

插入和新增節點-插入節點

- 可以將節點物件插入節點樹中指定兄弟節點之前
`objNode.insertBefore(objNewNode, objBrother);`
`objNewNode.appendChild(objTextNode);`



HTML DOM insertBefore() Method

https://www.w3schools.com/jsref/met_node_insertbefore.asp

Example- 插入與新增節點

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8"/>
<title>Ch6_6_1.html</title>
<script>
function appendNode() {
    var objNode = document.getElementById("myDiv");
    var strText = "段落四";
    var objNewNode = document.createElement("P");
    var objTextNode = document.createTextNode(strText);
    objNode.appendChild(objNewNode);
    objNewNode.appendChild(objTextNode);
}
function insertNode(objNode, objBrother, strText) {
    var objNode = document.getElementById("myDiv");
    var objBrother = objNode.children[1];
    var strText = '段落一.二';
    var objNewNode = document.createElement("P");
    var objTextNode = document.createTextNode(strText);
    objNode.insertBefore(objNewNode, objBrother);
    objNewNode.appendChild(objTextNode);
}
</script>
</head>
```

```
<body id="myBody">
<h2>在DOM節點樹插入和新增節點</h2>
<hr/>
<div id="myDiv">
<p onclick="appendNode()">段落一</p>
<p onclick="insertNode()">段落二</p>
<p>段落三</p>
</div>
</body>
</html>
```

在DOM節點樹插入和新增節點

段落一

段落一.二

段落二

段落三

段落四

刪除節點-刪除

- 使用DOM節點物件的removeChild()方法刪除節點樹中指定的子節點

objParent.removeChild(objNode);

- objParent是父節點物件，removeChild()方法可以刪除參數的子節點，即objNode。

HTML DOM removeChild() Method

https://www.w3schools.com/jsref/met_node_removechild.asp

刪除節點-顯示子節點

- Ch6_6_2.html範例是使用printChilds()函數來顯示指定節點的所有子節點
- if條件是使用hasChildNodes()方法檢查節點是否擁有子節點

```
if (objNode.hasChildNodes()) {  
    var nodeCount = objNode.children.length;  
    strMsg += "子元素數 = " + nodeCount + "\n";  
    for (var i = 0; i < nodeCount; i++)  
        strMsg += "標籤名稱 = " +  
            objNode.children[i].nodeName + "\n";  
    alert(strMsg);  
}
```

練習3

- 延續練習2的程式碼(ch06_ex2.html)
 - 請在XML之前插入一個節點-“CSS”
 - 請在XML之後新增一個節點-”ActionScript”

瀏覽節點與新增節點

- HTML
- JavaScript
- VBScript
- PHP
- XML