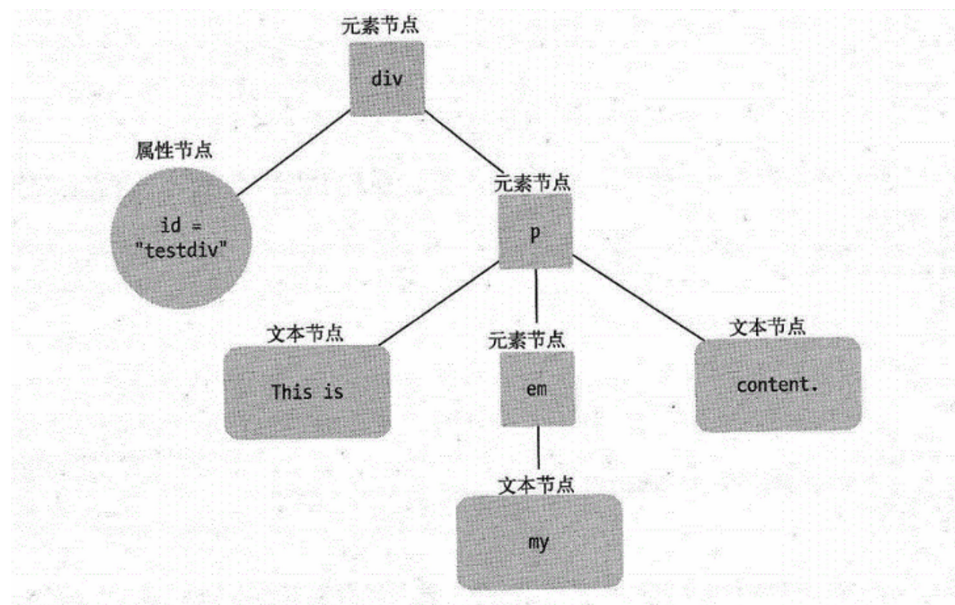


利用DOM方法在web浏览器中往文档添加标记时

- 传统方法
1. Document.write.--方便快捷地把字符串插入到文档中
缺点： 违背行为与表现分离原则
 2. innerHTML属性 -- 可用来读写某给定元素的HTML内容。
无细节。

```
<div id="testdiv">  
<p>This is <em>my</em> content.</p>  
</div>
```



```
window.onload = function() {  
    var testdiv = document.getElementById("testdiv");  
    alert(testdiv.innerHTML);  
}
```



DOM方法

1. createElement -- 创建一个新的元素
`document.createElement("element")`
2. appendChild -- 使新创建的元素成为html文档中已有元素的一个子节点
`var.appendChild(element)`
3. createTextNode -- 将文本放入新创建的元素中
`document.createTextNode("txt")`
4. insertBefore() -- 将把一个新元素插入到一个现有元素前面。

- (1) 创建一个 p 元素节点。
- (2) 把这个 p 元素节点追加到 test.html 文档中的一个元素节点上。
- (3) 创建一个文本节点。
- (4) 把这个文本节点追加到刚才创建的那个 p 元素节点上。

```
<div id="testdiv">  
</div>
```

```
window.onload = function() {  
    var para = document.createElement("p");  
    var testdiv = document.getElementById("testdiv");  
    testdiv.appendChild(para);  
    var txt = document.createTextNode("hello world");  
    para.appendChild(txt);  
}
```

insertBefore()

parentElement.insertBefore(newElement, targetElement)

。在调用此方法时，你必须告诉它三件事。

- (1) 新元素：你想插入的元素 (*newElement*)。
- (2) 目标元素：你想把这个新元素插入到哪个元素 (*targetElement*) 之前。
- (3) 父元素：目标元素的父元素 (*parentElement*)。

```

window.onload = function() {
    var para = document.createElement("p");
    var testdiv = document.getElementById("testdiv");
    testdiv.appendChild(para);
    var txt = document.createTextNode("hello world");
    para.appendChild(txt);
}

```

- (1) 创建一个 p 元素节点。
- (2) 创建一个文本节点。
- (3) 把这个文本节点追加到第 1 步创建的 p 元素节点上。
- (4) 把这个 p 元素节点追加到 test.html 文档中的一个元素节点上。

```

window.onload = function() {
    var para = document.createElement("p");
    var txt = document.createTextNode("hello world");
    para.appendChild(txt);
    var testdiv = document.getElementById("testdiv");
    testdiv.appendChild(para);
}

```

- | | |
|-----------------------------------|---|
| (1) 创建一个 p 元素节点并把它赋给变量 para。 | (6) 把 txt2 追加到 emphasis 上。 |
| (2) 创建一个文本节点并把它赋给变量 txt1。 | (7) 把 emphasis 追加到 para 上。 |
| (3) 把 txt1 追加到 para 上。 | (8) 创建一个文本节点并把它赋值给变量 txt3。 |
| (4) 创建一个 em 元素节点并把它赋给变量 emphasis。 | (9) 把 txt3 追加到 para 上。 |
| (5) 创建一个文本节点并把它赋值给变量 txt2。 | (10) 把 para 追加到 test.html 文档中的 testdiv 元素上。 |

```

window.onload = function() {
    var para = document.createElement("p");
    var txt1 = document.createTextNode("this is ");
    para.appendChild(txt1);
    var emphasis = document.createElement("em");
    var txt2 = document.createTextNode("my");
    emphasis.appendChild(txt2);
    para.appendChild(emphasis);
    var txt3 = document.createTextNode(" content");
    para.appendChild(txt3);
    (local var) testdiv: HTMLElement, Id("testdiv");
    testdiv.appendChild(para);
}

```

- (1) 创建一个 img 元素节点。
- (2) 设置这个节点的 id 属性。
- (3) 设置这个节点的 src 属性。
- (4) 设置这个节点的 alt 属性。
- (5) 创建一个 p 元素节点。
- (6) 设置这个节点的 id 属性。
- (7) 创建一个文本节点。
- (8) 把这个文本节点追加到 p 元素上。
- (9) 把 p 元素和 img 元素插入到 gallery.html 文档。

```

<p id="description">Choose an image.</p>
```

```
var placeholder = document.createElement("img");
placeholder.setAttribute("id", "placeholder");
placeholder.setAttribute("src", "images/1.jpeg");
placeholder.setAttribute("alt", "my IG");
var des = document.createElement("p");
des.setAttribute("id", des);
var destxt = document.createTextNode("choose an image");
des.appendChild(destxt);
document.body.appendChild(placeholder);
document.body.appendChild(des);
```