

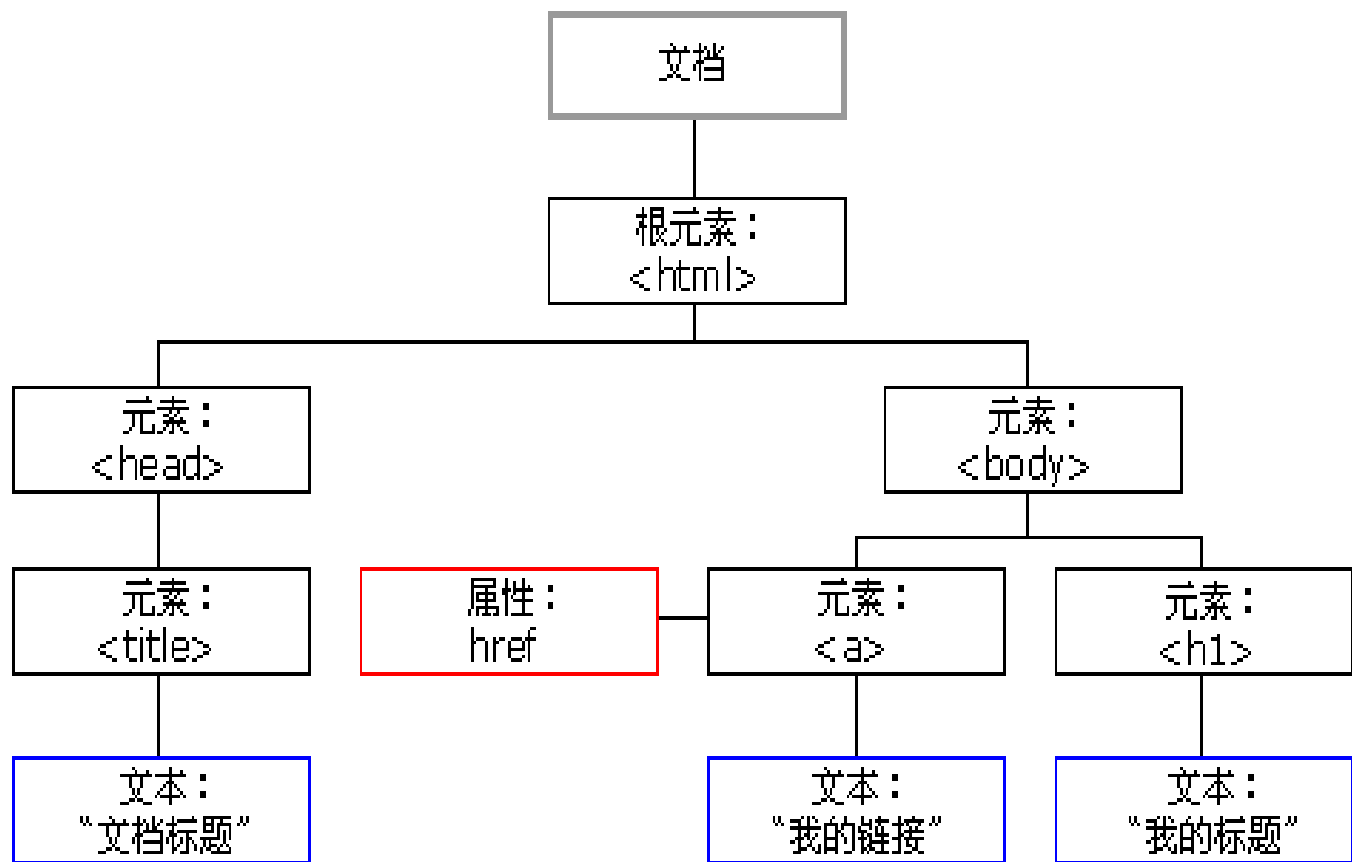
Web开发 (二)

--- 第八章 DOM 模型 (二)



河北师范大学软件学院
Software College of Hebei Normal University

DOM 节点



HTML 文档中的所有
内容都是**节点**：

- ① 整个文档是一个**文档节点**；
- ② 每个 HTML 元素是**元素节点**；
- ③ HTML 元素内的文本是**文本节点**；
- ④ 每个 HTML 属性是**属性节点**；
- ⑤ 注释是**注释节点**；

获取 DOM 节点

```
<body>
  <div>
    <h1>我是标题1</h1>
    <a href="demo.html" id="list">我是超链接</a>
  </div>
</body>
```

- 获取 DOM 节点

- 元素节点 `var list = document.getElementById("list")`
- 属性节点 `var href = list.getAttributeNode("href")`
- 文本节点 `var text = list.firstChild`

内容提纲

- **DOM 节点属性**
- **DOM 节点操作**
- **HTML DOM**
- **实例**



DOM 节点

- DOM 树中每个节点都拥有包含着关于节点某些信息的属性。这些属性是：

nodeType (节点类型)

nodeName (节点名称)

nodeValue (节点值)



nodeType 属性

- nodeType 属性可区分节点的类型
 - 元素节点的 nodeType 返回值为 **1**
 - 属性节点的 nodeType 返回值为 **2**
 - 文本节点的 nodeType 返回值为 **3**
- nodeType 是只读的

nodeName 属性

- nodeName 属性依据节点的类型返回其名称
 - 元素节点的 nodeName 与标签名相同
 - 属性节点的 nodeName 与属性名相同
 - 文本节点的 nodeName 始终是 #text
- nodeName 是只读的
- nodeName 返回标签名为大写标签名

nodeValue 属性

- nodeValue 属性返回或设置当前节点的值
 - 元素节点的 nodeValue 是 undefined 或 null
 - 属性节点的 nodeValue 是属性值
 - 文本节点的 nodeValue 是文本本身
- DOM 处理中的常见错误是希望元素节点包含文本节点
 - 例如 `<title>DOM 教程</title>`，元素节点 `<title>`，包含值为 "DOM 教程" 的文本节点。
 - 可通过节点的 `innerHTML` 属性来访问文本节点的值。

DOM 节点属性

元素类型	nodeType		nodeName	nodeValue
元素	1	Element	元素名	null
属性	2	Attr	属性名称	属性值
文本	3	Text	#text	节点的内容



内容提纲

- DOM 节点属性
- DOM 节点操作
- HTML DOM
- 实例



添加元素/文本节点

- 第一步：生成一个 DOM 节点

- 生成一个元素节点： `document.createElement(Element)`

通过指定标签名创建一个元素节点，返回一个节点对象。

例： `var btn=document.createElement("button");`

创建元素节点 html元素名

- 生成一个文本节点： `document.createTextNode(text)`

创建文本节点，返回文本节点对象。

例： `var text=document.createTextNode("hello!");`

创建文本节点 文本字符串

添加元素/文本节点

- 第二步：把生成的节点作为某一个节点的子节点进行添加

- 向父节点的末尾添加： **node.appendChild(newNode)**

例： `var btn=document.createElement("button");`

`document.body.appendChild(btn);`

父节点

添加子节点

新节点对象

- 在已有子节点前插入一个新的子节点： **node.insertBefore(new, old)**

例： `var btn=document.createElement("button");`

`document.body.insertBefore(btn, null);` ←

父节点

添加子节点

新节点对象

可选，如果未规定，则会在结尾插入newnode。



练习

- 通过创建节点和添加节点给 ul 适合的位置添加

```
<body>
  <ul>
    <li><a href="">我是列表第一项</a></li>
    <li><a href="">我是列表第二项</a></li>
    <li><a href="">我是列表第三项</a></li>
    <li><a href="">我是列表第五项</a></li>
  </ul>
</body>
```

demo 8-7.html

- 创建 7 行 5 列表格，第一列单元格含有按钮

点击我删除行 0				
点击我删除行 1				

demo 8-8.html

删除元素/文本节点

- 删除元素节点或文本节点

- 从子节点列表中删除某个节点: `parentNode.removeChild(childNode)`

例: 通过父节点删除本节点:

myParent.removeChild(mySelfNode)

父节点

要移除的节点对象

通过自己删除本节点:

mySelfNode.parentNode.removeChild(mySelfNode)

通过自身获取到的父节点

要移除的节点对象



练习

- 为按钮添加点击事件，点击按钮删除按钮所在行

点击我删除行 0				
点击我删除行 1				
点击我删除行 2				
点击我删除行 3				
点击我删除行 4				
点击我删除行 5				
点击我删除行 6				

修改元素/文本节点

- 修改一个元素节点，即用新节点替换某个子节点
 - oldNode.parentNode.replaceChild (newNode, oldNode)
父节点 替换子节点 新节点 原有节点
- DOM 节点替换过程
 - 创建新的节点
 - 找到旧的节点
 - 站在父节点的角度，使用 `replaceChild(新,旧)` 方法来替换



修改元素/文本节点

```
<ul id="ls"><li>苹果</li><li>香蕉</li><li>西瓜</li></ul>  
var txtnode=document.createTextNode("菠萝");
```

→ 创建一个新的文本节点

```
var item=document.getElementById("ls").childNodes[0];
```

→ 获取旧的元素节点的父节点

```
var oldtxtnode = item.childNodes[0];
```

→ 获取旧的文本节点

```
item.replaceChild( txtnode , oldtxtnode );
```

↓

```
<ul id="ls"><li>菠萝</li><li>香蕉</li><li>西瓜</li></ul>
```

内容提纲

- DOM 节点属性
- DOM 节点操作
- **HTML DOM**
- 实例



HTML DOM

- HTML DOM

- 定义了访问和操作 HTML 文档的方法
- HTML 元素对象化
- 网页中的每个元素都看作一个对象
 - 元素属性为元素节点对象的属性
 - 元素事件属性为元素节点对象的方法

- 对象

- Image, Table, Form, Input, Select ...

元素节点属性操作

- 核心 DOM 获取节点属性: `node.getAttribute(attrName)`

- `node.id`
- `node.className`

- HTML DOM 获取节点属性: `node.attrName`

```
<a id="aa" class="bb" href="1.html" target="_blank">跳转</a>  
var attrName= document.getElementById("aa");
```

<code>attrName.getAttribute("id")</code>	<code>==</code> ✓	<code>attrName.id</code>
<code>attrName.getAttribute("class")</code>	<code>==</code> ✗	<code>attrName.className</code>
<code>attrName.getAttribute("href")</code>	<code>==</code> ✗	<code>attrName.href</code>
<code>attrName.getAttribute("target")</code>	<code>==</code> ✓	<code>attrName.target</code>



Image 对象

- 创建图片节点对象

- `var img = new Image()`
- 等价于 `document.createElement("img")`
- <https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLImageElement/Image>

- 增删改查图片节点对象属性

- `img.src`
- `img.width`
- `img.height`
- ...



Table 对象

- Table 对象

- 在 HTML 文档中 `<table>` 标签每出现一次，一个 Table 对象就会被创建
- `var table = document.createElement("table")`

- 常用属性和方法

- `table.rows` —— 返回包含表格中所有 `tr` 元素的集合
- `table.createCaption()` —— 为表格创建一个 `caption` 元素
- `table.insertRow(index)` —— 在表格中的指定位置插入一个空的 `tr` 元素
- `table.deleteRow(index)` —— 从表格删除指定位置的行

TableRow 对象

- TableRow 对象

- 在 HTML 文档中出现一个 `<tr>` 标签, 就会创建一个 TableRow 对象
- `var tr = table.insertRow()`

- 常用属性和方法

- `tr.cells` —— 返回表格行中所有 td 或 th 元素的集合
- `tr.insertCell(index)` —— 在一行的指定位置插入一个空的 td 元素
- `tr.deleteCell(index)` —— 删除表格行中的指定 td 元素

TableCell 对象/ TableHeader 对象

- TableCell 对象

- 在 HTML 文档中出现一个 `<td>` 标签, 就会创建一个 TableCell 对象
- `var td = tr.insertCell()`

- TableHeader 对象

- 在 HTML 文档中出现一个 `<th>` 标签, 就会创建一个 TableHeader 对象
- `var th = document.createElement("th");`

DOM

- 核心 DOM
 - 操作元素**节点**，创建，删除，查找元素节点等
- HTML DOM
 - 操作**属性**，如读取或修改属性的值



内容提纲

- DOM 节点属性
- DOM 节点操作
- HTML DOM
- 实例



实例

- 复选框实例



The background of the slide is decorated with numerous overlapping circles in various shades of green and yellow, scattered across the top and right sides.

Thank You!