



BOM和DOM

Dr. Zhili Zhao

Email: zhaozhl@lzu.edu.cn
School of Information Science & Engineering
<http://xxxxy.lzu.edu.cn>



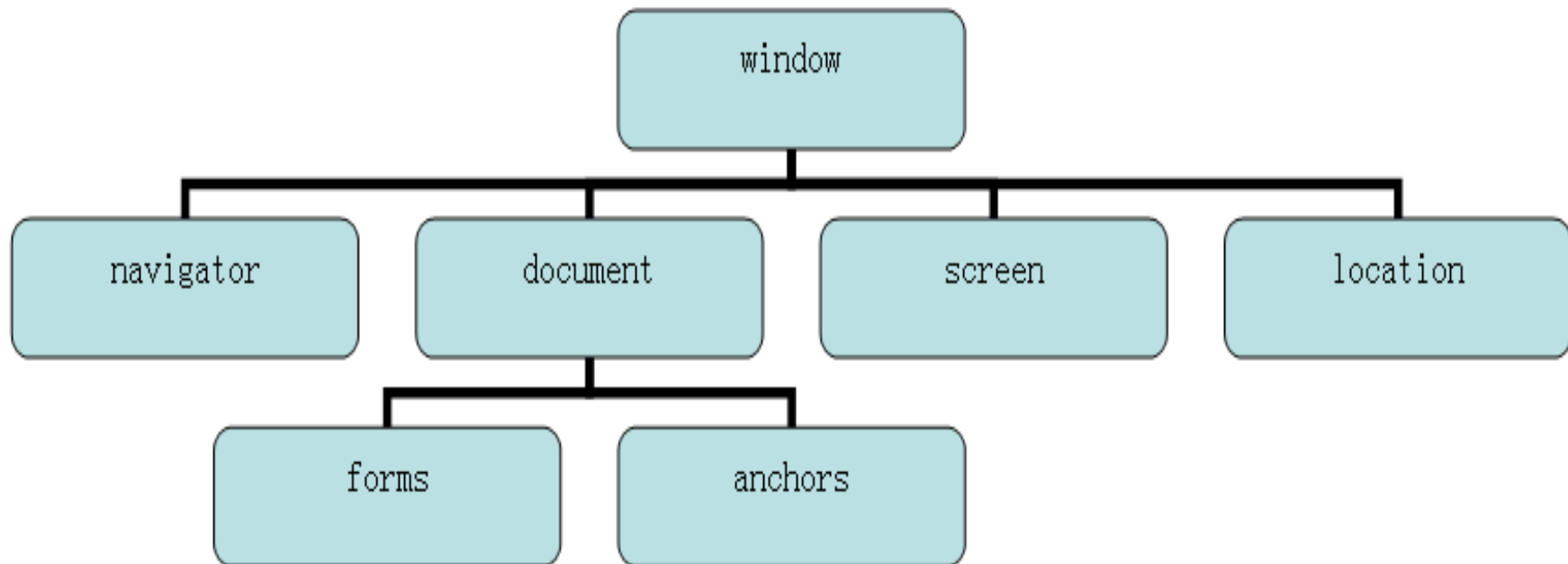
浏览器对象模型（ Browser Object Model , BOM ）和文档对象模型（ Document Object Model , DOM ）为JavaScript编程提供了更多的功能。

在JavaScript中使用BOM提供的专门对象可以与Web浏览器进行交互，而DOM则使得JavaScript对HTML文档有了空前的访问和处理能力。

- 浏览器对象模型
- 文档对象模型及HTML文档树
- 事件及其处理
- 事件的综合处理

1. 浏览器对象模型

BOM提供的专门对象主要包括window、navigator、document、screen和location等，并且这些对象都是预先定义的。



1. 浏览器对象模型



在Web浏览器宿主环境中，**window对象**是JavaScript的**全局对象**，因此**在使用window对象的属性和方法时不需要特别指明window对象**。

例如，在JavaScript中经常使用的alert方法实际上就是window对象的一个方法，调用该方法的完整的JavaScript代码应该是**window.alert**，但通常在JavaScript代码中省略window、而直接写成**alert**。

1.2 window对象的方法



通过window对象调用方法可以直接对Web浏览器窗口进行各种操作，同时实现相应的功能。

1. 生成系统对话框。这类方法包括以下三个方法。

(1) `alert(message)` 方法。该方法用于显示包含一条指定消息和一个“确定”按钮的警告框。其中的参数message是用于指定消息的字符串。

(2) `confirm(message)` 方法。该方法用于显示包含一条指定消息和“确定”及“取消”按钮的确认框。其中的参数message是用于指定消息的字符串。

在确认框中，如果单击“确定”按钮，该方法将返回布尔值`true`。如果单击“取消”按钮，则该方法将返回布尔值`false`。

【例13-2】确认框和警告框。

(3) `prompt(message, defaultText)`方法。该方法用于显示一个提示框，并在其中包含一个文本框，该文本框能够接受用户输入的字符串数据。其中，参数`message`是显示给用户、起提示作用的字符串，参数`defaultText`是文本框中的默认字符串。该方法返回文本框中的字符串。

【例13-3】提示框。

2. 打开新窗口和关闭窗口。这类方法主要包括open和close两个方法。

(1) `open(URL,name,features)`方法。该方法用于打开一个新的Web浏览器窗口。

(2) `close()`方法。该方法用于关闭一个Web浏览器窗口。

3. 定时操作，其目的主要有两个：

- 一个是将某些操作延迟一段时间后再执行。例如打开某些网站的首页时，会同时打开一个临时的广告窗口，该广告窗口会在几十秒钟后自动关闭，相当于关闭该广告窗口的操作延迟几十秒钟后再执行。
- 另一个则是周期性地执行某些操作。例如在网页中显示时钟，需要每隔一秒钟更新一次时间的显示；又如在网页上漂浮的广告图片，就是首先让广告图片在网页上的某个位置静止几毫秒，然后再更新广告图片在网页上的位置，相当于周期性地更新广告图片在Web浏览器窗口中的位置，这样即可产生广告图片在网页上漂浮的动画效果。

利用window对象进行定时操作，可以使用以下几个方法。

(1) **setTimeout**("code",millisec)方法。该方法用于设置定时器，在一段时间之后执行指定的JavaScript代码。

setTimeout方法将为每个定时器分配一个唯一的ID值，且**返回该定时器的ID值**。

【例13-4】 window对象的open、close和setTimeout方法。

(2) **clearTimeout**(id_of_settimeout)方法。该方法用于清除由setTimeout方法设置的定时器。

1.2 window对象的方法



(3) **setInterval**("code",millisec)方法。该方法用于设置定时器，每隔一段时间周期性地执行指定的JavaScript代码。

setInterval方法将为每个定时器分配一个唯一的ID值，且返回该定时器的ID值。

(4) **clearInterval**(id_of_setinterval)方法。该方法用于清除由setInterval方法设置的定时器。

注意：setTimeout和setInterval方法的**区别**在于，前者**只执行**由参数code指定的JavaScript代码**一次**，而后者则**反复地执行**由参数code指定的JavaScript代码，直到使用clearInterval方法清除由setInterval方法设置的定时器或关闭Web浏览器窗口。

2. 文档对象模型及HTML文档树

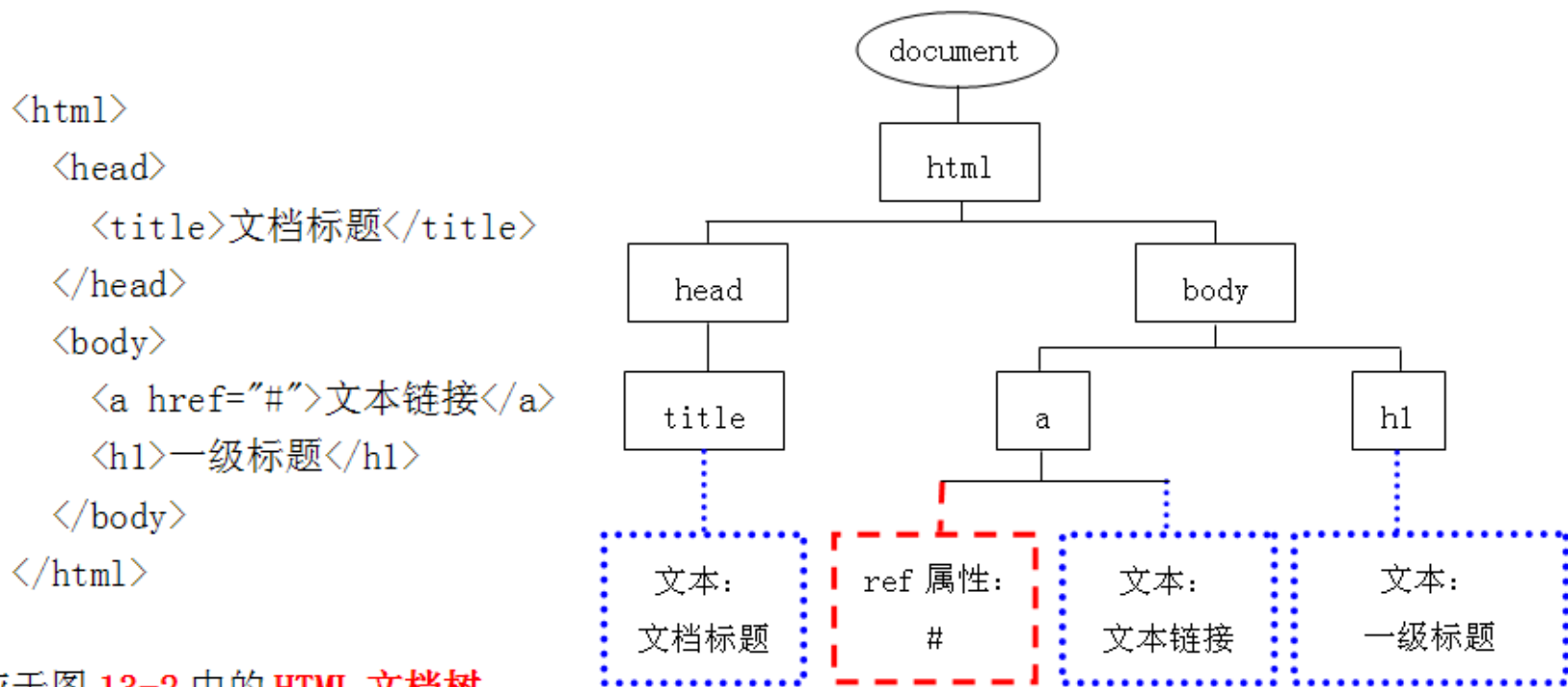


与BOM不同，**文档对象模型**（Document Object Model, **DOM**）定义了访问和处理HTML文档的标准接口和方法。在DOM中，HTML文档中的每个成分都是一个**节点（Node）**。DOM规定：

- （1）整个HTML文档对应一个**文档节点（Document Node）**。
- （2）每个HTML元素是一个**元素节点（Element Node）**。
- （3）HTML元素的每个属性是一个**属性节点（Attribute Node）**。
- （4）包含在HTML元素中的文本称为**文本节点（Text Node）**。
- （5）注释属于**注释节点（Comment Node）**。

在DOM中，HTML文档中的所有节点组成了一个HTML文档树（也称HTML节点树）。

2. 文档对象模型及HTML文档树



对应于图 13-2 中的 HTML 文档树:

图 13-2 HTML 文档树

在JavaScript中，HTML文档树的文档节点document以及每个元素节点都被看作一个对象。文档节点document对应的对象称为document对象，元素节点对应的对象称为元素对象。通过document对象和元素对象可以调用相应的方法。

2.1 使用innerHTML属性访问元素的内容



对应于HTML文档树中的元素节点，**每个元素对象都有innerHTML属性**，该属性值是一个字符串。对于br、img等**空元素**对象，innerHTML属性值是**空字符串**，即""。而对于**非空元素**对象，innerHTML属性值是**位于元素的开始标签和结束标签之间的HTML代码**。例如，对于以下HTML代码

```
<p id="p1">这是一个段落</p>
```

```
<div id="d2">这是一个<strong>块</strong></div>
```

p元素对象的innerHTML属性值就是“这是一个段落”，而div元素对象的innerHTML属性值就应该是“这是一个块”。

2.2 调用getElementById方法访问HTML文档树中的元素节点



在HTML文档中某个**元素的开始标签内**可以使用**id属性及属性值**唯一地标识该元素。因此，对于指定的id属性值，在HTML文档中只可能存在唯一的具有该id属性值的元素。

而通过document对象调用**getElementById方法**，即可根据指定的id属性值返回HTML文档树中唯一的具有该id属性值的元素节点，无论这个元素节点位于HTML文档树中的任何位置。

调用getElementById方法的具体语法为：

```
document.getElementById("id_of_element");
```

2.2 调用getElementById方法访问HTML文档树中的元素节点



【例13-5】调用getElementById方法、并使用innerHTML属性获取和设置元素的内容。

【例13-6】使用setInterval方法实现时钟。

【例13-7】打字效果的字符串（仅显示一次）。

3. 事件及其处理



事件 (Event) 是JavaScript和DOM之间进行交互的桥梁。当某个事件发生时，可以通过对应的**事件处理函数 (Event Handler)** 执行特定的JavaScript代码。

3.1 事件的类型

根据事件的来源或触发事件的原因，可以将与JavaScript编程有关的常见事件分为**窗口事件**、**表单事件**、**鼠标事件**和**键盘事件**四种类型。

事件分类	事件描述	典型的关联元素	相关属性
窗口事件	Web 浏览器加载 HTML 文档	body	onload
	Web 浏览器卸载 HTML 文档	body	onunload
	改变 Web 浏览器窗口大小	body	onresize
表单事件	提交表单	form	onsubmit
	重置表单	form	onreset
	选取元素	input	onselect
	元素值发生改变	input	onchange
	元素获得焦点	input	onfocus
	元素失去焦点	input	onblur

3.1 事件的类型

鼠标事件	按下鼠标	input、div、span 等大多数元素	onmousedown
	松开鼠标	input、div、span 等大多数元素	onmouseup
	单击鼠标	input、div、span 等大多数元素	onclick
	双击鼠标	input、div、span 等大多数元素	ondblclick
	鼠标指针悬停在元素上	input、div、span 等大多数元素	onmouseover
	鼠标指针移出元素	input、div、span 等大多数元素	onmouseout
	移动鼠标指针	input、div、span 等大多数元素	onmousemove
键盘事件	按下键盘	input、div、span 等大多数元素	onkeydown
	松开键盘	input、div、span 等大多数元素	onkeyup
	按下键盘后又松开键盘	input、div、span 等大多数元素	onkeypress

在HTML文档及JavaScript中，一个事件可以与某个HTML元素关联，而且还要与该HTML元素的特定属性联系在一起。

3.2 在HTML元素的开始标签中处理事件



在HTML元素的开始标签中，可以通过相关属性值调用事件处理函数，完成相应的事件处理任务。

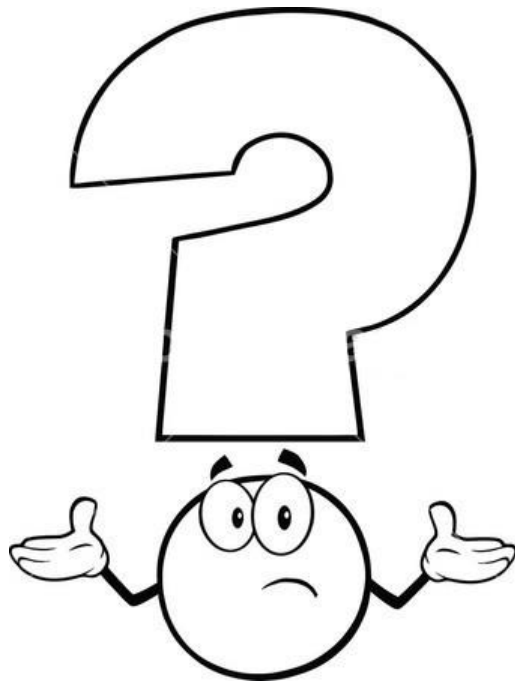
【例13-9】在HTML元素的开始标签中处理事件。

4. 事件的综合处理



在JavaScript中，不仅可以处理窗口事件和鼠标事件，而且能够处理表单事件。在处理表单事件时，有时还需要动态地调整某些控件的工作状态。

【例13-11】 带有开始按钮和停止按钮的计数器。



Questions?