

第 24 章 事件入门

学习要点:

- 1.事件介绍
- 2.内联模型
- 3.脚本模型
- 4.事件处理函数

主讲教师: 李炎恢

合作网站: <http://www.ibeifeng.com>

讲师博客: <http://hi.baidu.com/李炎恢>

JavaScript 事件是由访问 Web 页面的用户引起的一系列操作, 例如: 用户点击。当用户执行某些操作的时候, 再去执行一系列代码。

一. 事件介绍

事件一般是用于浏览器和用户操作进行交互。最早是 IE 和 Netscape Navigator 中出现, 作为分担服务器端运算负载的一种手段。直到几乎所有的浏览器都支持事件处理。而 DOM2 级规范开始尝试以一种复合逻辑的方式标准化 DOM 事件。IE9、Firefox、Opera、Safari 和 Chrome 全都已经实现了“DOM2 级事件”模块的核心部分。IE8 之前浏览器仍然使用其专有事件模型。

JavaScript 有三种事件模型: 内联模型、脚本模型和 DOM2 模型。

二. 内联模型

这种模型是最传统接单的一种处理事件的方法。在内联模型中, 事件处理函数是 HTML 标签的一个属性, 用于处理指定事件。虽然内联在早期使用较多, 但它是和 HTML 混写的, 并没有与 HTML 分离。

```
//在 HTML 中把事件处理函数作为属性执行 JS 代码  
<input type="button" value="按钮" onclick="alert('Lee');" /> //注意单双引号
```

```
//在 HTML 中把事件处理函数作为属性执行 JS 函数  
<input type="button" value="按钮" onclick="box();" /> //执行 JS 的函数
```

PS: 函数不得放到 window.onload 里面, 这样就看不见了。

三. 脚本模型

由于内联模型违反了 HTML 与 JavaScript 代码层次分离的原则。为了解决这个问题, 我们可以在 JavaScript 中处理事件。这种处理方式就是脚本模型。

```
var input = document.getElementsByTagName('input')[0]; //得到 input 对象  
input.onclick = function () { //匿名函数执行  
    alert('Lee');  
};
```

PS: 通过匿名函数, 可以直接触发对应的代码。也可以通过指定的函数名赋值的方式来执行函数(赋值的函数名不要跟着括号)。

```
input.onclick = box; //把函数名赋值给事件处理函数
```

四. 事件处理函数

JavaScript 可以处理的事件类型为: 鼠标事件、键盘事件、HTML 事件。

JavaScript 事件处理函数及其使用列表

事件处理函数	影响的元素	何时发生
onabort	图像	当图像加载被中断时
onblur	窗口、框架、所有表单对象	当焦点从对象上移开时
onchange	输入框, 选择框和文本区域	当改变一个元素的值且失去焦点时
onclick	链接、按钮、表单对象、图像映射区域	当用户单击对象时
ondblclick	链接、按钮、表单对象	当用户双击对象时
ondragdrop	窗口	当用户将一个对象拖放到浏览器窗口时
onError	脚本	当脚本中发生语法错误时
onfocus	窗口、框架、所有表单对象	当单击鼠标或者将鼠标移动聚焦到窗口或框架时
onkeydown	文档、图像、链接、表单	当按键被按下时
onkeypress	文档、图像、链接、表单	当按键被按下然后松开时
onkeyup	文档、图像、链接、表单	当按键被松开时
onload	主题、框架集、图像	文档或图像加载后
onunload	主体、框架集	文档或框架集卸载后
onmouseout	链接	当图标移除链接时
onmouseover	链接	当鼠标移到链接时
onmove	窗口	当浏览器窗口移动时
onreset	表单复位按钮	单击表单的 reset 按钮
onresize	窗口	当选择一个表单对象时
onselect	表单元素	当选择一个表单对象时
onsubmit	表单	当发送表格到服务器时

PS: 所有的事件处理函数都会都有两个部分组成, on + 事件名称, 例如 click 事件的事

件处理函数就是：`onclick`。在这里，我们主要谈论脚本模型的方式来构建事件，违反分离原则的内联模式，我们忽略掉。

对于每一个事件，它都有自己的触发范围和方式，如果超出了触发范围和方式，事件处理将失效。

1.鼠标事件，页面所有元素都可触发

`click`：当用户单击鼠标按钮或按下回车键时触发。

```
input.onclick = function () {  
    alert('Lee');  
};
```

`dblclick`：当用户双击主鼠标按钮时触发。

```
input.ondblclick = function () {  
    alert('Lee');  
};
```

`mousedown`：当用户按下了鼠标还未弹起时触发。

```
input.onmousedown = function () {  
    alert('Lee');  
};
```

`mouseup`：当用户释放鼠标按钮时触发。

```
input.onmouseup = function () {  
    alert('Lee');  
};
```

`mouseover`：当鼠标移到某个元素上方时触发。

```
input.onmouseover = function () {  
    alert('Lee');  
};
```

`mouseout`：当鼠标移出某个元素上方时触发。

```
input.onmouseout = function () {  
    alert('Lee');  
};
```

`mousemove`：当鼠标指针在元素上移动时触发。

```
input.onmousemove = function () {  
    alert('Lee');  
};
```

2.键盘事件

`keydown`：当用户按下键盘上任意键触发，如果按住不放，会重复触发。

```
onkeydown = function () {  
    alert('Lee');  
};
```

keypress: 当用户按下键盘上的字符键触发, 如果按住不放, 会重复触发。

```
onkeypress = function () {  
    alert('Lee');  
};
```

keyup: 当用户释放键盘上的键触发。

```
onkeyup = function () {  
    alert('Lee');  
};
```

3.HTML 事件

load: 当页面完全加载后在 window 上面触发, 或当框架集加载完毕后在框架集上触发。

```
window.onload = function () {  
    alert('Lee');  
};
```

unload: 当页面完全卸载后在 window 上面触发, 或当框架集卸载后在框架集上触发。

```
window.onunload = function () {  
    alert('Lee');  
};
```

select: 当用户选择文本框(input 或 textarea)中的一个或多个字符触发。

```
input.onselect = function () {  
    alert('Lee');  
};
```

change: 当文本框(input 或 textarea)内容改变且失去焦点后触发。

```
input.onchange = function () {  
    alert('Lee');  
};
```

focus: 当页面或者元素获得焦点时在 window 及相关元素上面触发。

```
input.onfocus = function () {  
    alert('Lee');  
};
```

blur: 当页面或元素失去焦点时在 window 及相关元素上触发。

```
input.onblur = function () {  
    alert('Lee');  
};
```

submit: 当用户点击提交按钮在<form>元素上触发。

```
form.onsubmit = function () {  
    alert('Lee');  
};
```

reset: 当用户点击重置按钮在<form>元素上触发。

```
form.onreset = function () {  
    alert('Lee');  
};
```

resize: 当窗口或框架的大小变化时在 window 或框架上触发。

```
window.onresize = function () {  
    alert('Lee');  
};
```

scroll: 当用户滚动带滚动条的元素时触发。

```
window.onscroll = function () {  
    alert('Lee');  
};
```

感谢收看本次教程!

本课程是由北风网(ibeifeng.com)

瓢城 **Web** 俱乐部(yc60.com)联合提供:

本次主讲老师: 李炎恢

我的博客: hi.baidu.com/李炎恢/

我的邮件: yc60.com@gmail.com