



事件



# 学习目标

- 1、掌握什么是事件
- 2、掌握事件流
- 3、掌握DOM事件流与IE事件处理程序
- 4、掌握跨浏览器的事件处理程序
- 5、掌握event对象的常用属性及方法
- 6、掌握常用的事件类型

# 关于事件

**什么是事件？**

**事件是可以被 JavaScript 侦测到的行为，通俗的讲就是当用户与Web页面进行某些交互时，解释器就会创建响应的 event 对象以描述事件信息。**

# 关于事件

**常见的事件有：**

**用户点击页面上的某项内容**

**鼠标经过特定的元素**

**用户按下键盘上的某个按键**

**用户滚动窗口或改变窗口大小**

**页面元素加载完成或加载失败**

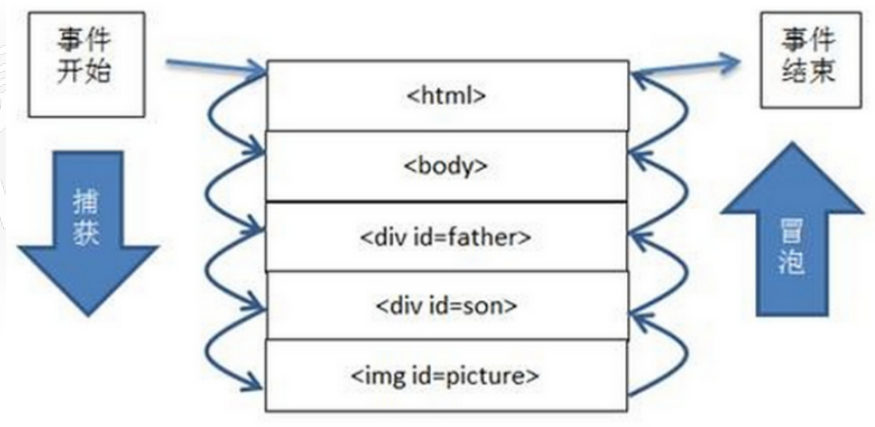
# 事件周期

解释器创建一个event对象后，会按如下过程将其在HTML元素间进行传播

第一阶段：事件捕获，事件对象沿DOM树向上传播

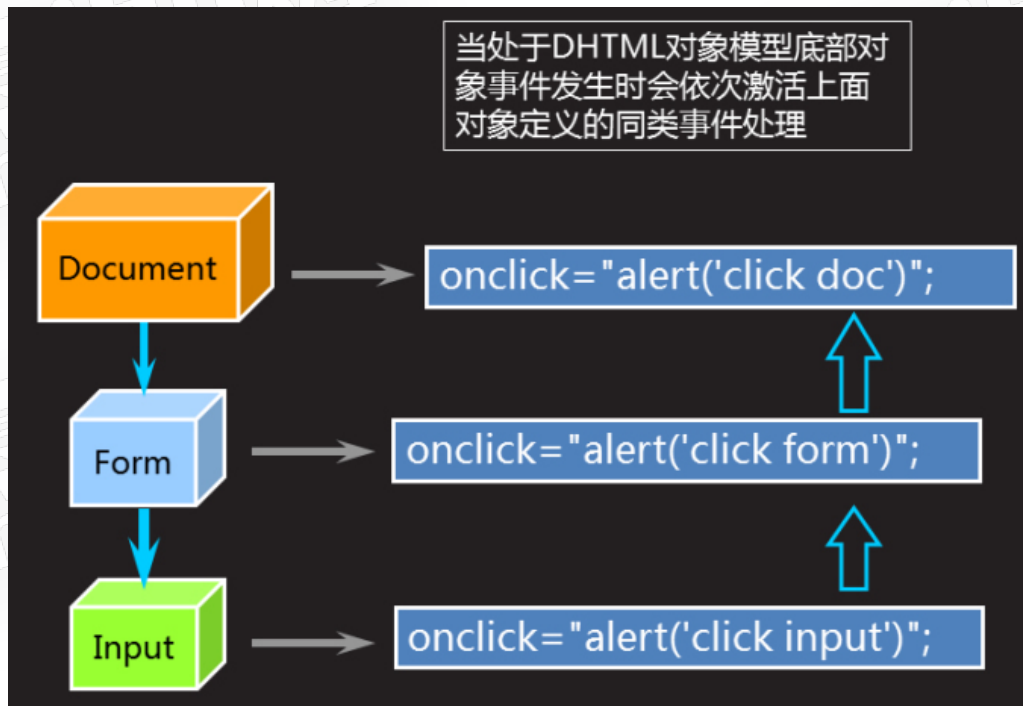
第二阶段：目标触发，运行事件监听函数

第三阶段：事件冒泡，事件沿DOM树向上传播



IE的事件模型  
中没有“事件  
捕获”阶段

# 事件的冒泡处理机制



# 事件句柄

## 什么是事件句柄？

事件句柄（又称事件处理函数、事件监听函数），指用于响应某个事件而调用的函数。每一个事件均对应一个事件句柄，在程序执行时，将相应的函数或语句指定给事件句柄，则在该事件发生时，浏览器便执行指定的函数或语句。

# 事件定义

为特定事件定义监听函数有三种方式：

(1)、直接在HTML中定义元素的事件相关属性

`<button onclick="alert('hello')">按钮</button>`

`<body onload="init()">...</body>`

缺点：违反了“内容与行为相分离”的原则，应尽可能少用。



# 事件定义

## ( 2 )、DOM0级事件

在JavaScript中为元素的事件相关属性赋值：

```
document.getElementById("btn").onclick = function(){ // ...}
```

```
document.body.onload = init;
```

```
function init(){ // ...}
```

缺点：此语法实现了“内容与行为相分离”，但元素只能绑定一个监听函数。

# 事件定义

## ( 3 )、DOM2级事件

高级事件处理方式，一个事件可以绑定多个监听函数：

```
btn.addEventListener("click",function(){}),false); // DOM
```

```
btn.attachEvent("onclick",function(){}); // IE
```

```
document.body.addEventListener("load",init);
```

```
document.body.attachEvent("onload",init);
```

```
function init(){ //... }
```

此语法可以为一个元素绑定多个监听函数，但需要注意浏览器兼容性问题

# DOM事件流

**DOM0级与DOM2级区别：**

- 1、DOM2支持同一dom元素注册多个同种事件。**
- 2、DOM2新增了捕获和冒泡的概念。**

# DOM事件流

**添加事件：**

**addEventListener()**

**语法：**element.addEventListener(event, function, useCapture)

**功能：**用于向指定元素添加事件句柄

**参数：**

**event：**必须。字符串，指定事件名。

**function：**必须。指定要事件触发时执行的函数。

**useCapture：**可选。布尔值，指定事件是否在捕获或冒泡阶段执行。

# DOM事件流

**移除事件：**

**removeEventListener()**

**语法：**element.removeEventListener(event, function, useCapture)

**功能：**移除 addEventListener() 方法添加的事件句柄。

**参数：**

**event：**必须。字符串，要移除的事件名称。

**function：**必须。指定要移除的函数。

**useCapture：**可选。布尔值，指定移除事件句柄的阶段。

## IE事件流（IE事件处理程序）

**添加事件：**

**attachEvent()**

**语法：**element.attachEvent(event, function)

**功能：**用于向指定元素添加事件句柄

**参数：**

**event：**必须。字符串，指定事件名，必须加“on”前缀。

**function：**必须。指定要事件触发时执行的函数。

# IE事件流

**移除事件：**

**detachEvent()**

**语法：**element.detachEvent(event, function)

**功能：**移除 attachEvent() 方法添加的事件句柄。

**参数：**

**event：**必须。字符串，要移除的事件名称。

**function：**必须。指定要移除的函数。

## event对象常用属性和方法

**type** : 事件的类型;

**srcElement/target** : 事件源 , 就是发生事件的元素 ;

**cancelBubble** : 布尔属性 , 设为true的时候 , 将停止事件进一步起泡到包容层次的元素 ; (e.cancelBubble = true; 相当于 e.stopPropagation());;

**returnValue** : 布尔属性 , 设置为false的时候可以组织浏览器执行默认的事件动作 ; (e.returnValue = false; 相当于 e.preventDefault());;



# event对象常用属性和方法

**clientX/clientY** : 事件发生的时候, 鼠标相对于浏览器窗口可视文档区域的左上角的位置;

**offsetX,offsetY** : 事件发生的时候, 鼠标相对于事件源元素左上角的位置;

**说明** : *event*对象的所有属性列表可以在浏览器控制台中输出查看。

## 常用事件类型

**onclick:**当用户点击某个对象时调用的事件句柄。

**onfocus:**元素获得焦点。

**onblur:**元素失去焦点。

**onmouseover:**鼠标移到某元素之上。

**onmouseout:**鼠标从某元素移开。

**onmousedown:**鼠标按钮被按下。

**onmousemove:**鼠标被移动。

**onmouseup:**鼠标按键被松开。

## 常用事件类型

**onresize:**窗口或框架被重新调整大小。

**onsubmit:**表单提交。

**onkeydown:**某个键盘按键被按下。

**onkeypress:**某个键盘按键被按下并松开。

**onkeyup:**某个键盘按键被松开。

## 常用事件类型

**touchstart:**手指触摸屏幕。

**touchmove:**手指在屏幕上滑动。

**touchend:**手指从屏幕上移开。

# 小项目

**利用所学知识，实现一个非常小的小项目，方便同学们巩固所学的知识点。**