# DOM

## DOM事件

DOM0级事件

如 document.onclick=funA;

document.onclick=funB;

为DOM0级事件，funB会覆盖funA;

DOM2级事件

addEventListener() IE9不兼容

则会添加多个事件方法。

兼容所有的解决方案：g0001

### 剪贴板事件

beforecopy：在发生复制操作前触发；

copy：在发生复制操作的时候触发；

beforecut：在发生剪切操作前触发；

cut：在发生剪切操作的时候触发；

beforepaste：在发生粘贴操作前触发；

paste：在发生粘贴操作的时候触发。

#### 访问剪贴板数据

clipboardData对象

在IE中，clipboardData对象是window对象的属性；

而在Chrome、Safari和Firefox 4+中，clipboardData对象是相应event对象的属性。

clipboardData对象方法

getData( dataType )

在IE中，为text | URL

其他中： 是一个MIME类型。text可以代表‘text/plain’

setData(dataType , data)

不同的是这里的dataType不能识别text

clearData()

clipboardData对象属性

dropEffect none

effectAllowed uninitialized

Files 粘贴为空List

items 剪切板中各数据

types 剪切板中的数据类型

## 文档碎片

DocumentFragment:

没有父级文件的最小文档对象，不是真实DOM树的一部分，它本身变化不会引起DOM树的重新渲染。不会导致性能问题。可以将其插入文档树（插入的不是DocumentFragment，而是其子孙节点，使得DocumentFragment成为有用的占位符，可以暂存节点。）

### 生成DocumentFragment

document.createDocumentFragment()

## DOM属性

classList

value：class值

方法：

add(string,[string])

remove(string, [string])

item(num,[num])

返回值

toggle(string, [string]) :兼容性问题

contains(string)

返回是否存在

ajax

function ajax(){

var xmlhttp;

if(window.XMLHttpRequest){

xmlhttp = new XMLHttpRequest();

}else{

// code for IE6, IE5

xmlhttp = ActiveXObject("Microsoft.XMLHTTP");

}

//判定执行状态

xmlhttp.onreadystatechange = function(){

/\*

readyState

0: 请求未初始化

1: 服务器连接已建立

2: 请求已接收

3: 请求处理中

4: 请求已完成，且响应已就绪

status

200:请求成功

404:未找到

500:服务器内部错误

\*/

if (xmlhttp.readyState==4 && xmlhttp.status==200){

document.getElementById("myDiv").innerHTML=xmlhttp.responseText;//获得字符串形式的响应数据

}

}

xmlhttp.open("Get","url",true);

//设置头信息

xmlhttp.setRequestHeader("Content-type","application/x-www-form-urlencoded");

//将信息发送到服务器

xmlhttp.send();

}