**事 件**

**一、事件基础知识**

1. ele.on+type=function (event){} 兼容性很好，但是一个元素只能绑定一个处理函数，基本等同于写在HTML行间上

var div=document.getElementsByTagName(‘div’);  
(1)div.onclick=function (){console.log(‘a’);};

(2)<div onclick=’console.log(‘a’)’>句柄写法

1. obj.addEventListener(‘type’,function(){},false):可以为一个事件绑定多个处理程序（IE9及以下不兼容），false改true时为捕捉事件（只有Chrome浏览器实现捕捉功能）
2. obj.attachEvent(‘on’+type,function(){}):IE独有，一个事件可以绑定多个处理程序
3. 封装兼容性方法：

function addEvent(elem,type,func){  
 if(elem.addEventListener){

elem.addEventListener(type,func,false);}  
else if(elem.attachEvent){

elem.attachEvent(‘on’+type,function(){func.call(elem);}) ;}

elese{

elem[‘on’+type]=func;}

}

1. 解除事件：
2. removeEvenListener(type,func,false) func不能写成匿名函数，否则无法解除事件绑定
3. detachEvent(‘on’+type,func)
4. elem[‘on’+type]=null/false/’’
5. 事件处理模型：
6. 事件冒泡：结构上嵌套关系的元素，会存在事件冒泡的功能，即同一事件，自子元素冒泡向父元素（自下而上）
7. 事件捕获：结构上嵌套关系的元素，会存在事件捕获的功能，即同一事件，自父元素捕获向子元素（自上而下）

//IE没有捕获事件功能，只有Chrome实现了捕获功能

1. 触发顺序，先捕获，后冒泡
2. focus、blur、change、submit、reset、select等事件不冒泡
3. 事件对象：绑定事件时，系统会自动将事件对象传到形参上

elem.onclick=function(e){conosle.log(e);} //MouseEvent{}

IE不行，IE用window.event

1. 取消冒泡功能：

W3C标准：event.stopPropagation()

IE独有：event.stopBubble=true

封装取消冒泡的函数：

elem.onclick=function(e){  
 var e=e||window.event;  
 stopPropagation(e);  
 }

function stopBubble(event){  
 if(event.stopPropagation){  
 event.stopPropagation();}  
 else{  
 event.stopBubble=true;}  
 }

9、阻止默认事件：

（1）return false;以对象属性的方式注册的事件才生效

（2）event.preventDefault()：W3C标准，IE9以下不兼容

（3）event.returnValue=false;兼容IE

10、封装阻止默认事件函数cancelHandler(event)：

eg：右键点击出菜单默认事件

document.oncontextmenu=function(){  
 return false;} //兼容性最好的写法

elem.oncontextmenu()=function(e){  
 var e=e||window.event;

cancelHandler(e);

stopBubble(e);

}  
function cancelHandler(event){

if(event.preventDefault){  
 event.preventDefault();}  
 else{  
 event.returnValue=false;}

}

1. <a href=’javascript:void(0/false)’>demo</a>相当于return false，取消默认事件
2. 事件源对象：

event.target 火狐只有这个

event.srcElement IE只有这个

Chrome两者都有

13、事件源对象用例：ul下多个li，要求点击出现li中内容

var ul=document.getElementsByTagName('ul')[0];

ul.onclick=function(e){

var event=e||window.event;

var target=e.target||e.srcElement;

console.log(target);

if(target!=ul) //防止点击ul出现不需要的内容

console.log(target.innerHTML);

}

1. **事件类型**
2. 鼠标事件

click、mousedown、mousemove、mouseup、contextmenu、mouseover、mouseout、mouseenter、mouseleave

（1）用button来区分鼠标的按键，0/1/2

（2）DOM3标准规定：click事件只监听左键，只能通过mousedown和mouseup来判断鼠标键

（3）如何解决mousedown和click的冲突

1. 键盘类事件（移动端：touchstart,touchend,touchmove）

keydown、keypress、keyup

keydown：可以响应任意键盘按键，e.which对应108个按键（不是按照ASCⅡ码排的，无法区分大小写）

keypress：只能响应字符类键盘按键，返回ASCⅡ码，可以转换成相应字符，String.fromCharCode(e.charCode)可以将ASCⅡ码转换成字符

1. 文本操作事件
2. input.onchange:失去焦点，内容发生变化触发
3. input.oninput:输入就触发
4. input.onfocus:聚焦触发事件
5. input.onblur:失去焦点触发事件
6. 窗体操作类（window上的事件）：
7. window.onscroll：IE6没有fixed定位
8. window.onload：加载最慢，不推荐使用，整个页面完全加载后 才执行，效率最低，可以提醒页面加载完毕时刻， 推出广告时

html解析时生成domTree和cssTree---->rederTree

（3）扫雷（扩散算法），贪吃蛇