

BOM

BOM (Browser Object Model)

Bom 就是浏览器的对象模型

BOM的核心对象是window 它是浏览器中最大的一个全局对象, 因此在全局作用域中声明的对象和函数都是window对象的属性和方法

```
window.alert();
var a = 10;
alert(window.a);
function aa()
{
    alert(1);
}
window.aa();
```

系统对话框

警告框: alert(“内容”), 没有返回值

选择框: confirm(“提问的内容”), 返回boolean

输入框: prompt(), 返回字符串或null

window的属性和方法

#screen对象

窗口位置 (了解)

screenLeft screenTop

```
console.log(window.screenLeft+"-"+window.screenTop);
```

location对象

href属性 控制浏览器地址栏的内容

reload()方法 刷新页面

```
btn1.onclick = function(){
    window.location.reload();
    window.location = "http://www.baidu.com";
}
```

#history对象

用户（在浏览器窗口中）访问过的 URL

back() //上一个 forward() //下一个

```
function jumpPage(){
    var btn1 = document.getElementById("btn");
    window.location.href="page2.html";
}
function backPage(){
    window.history.back();
}
function forPage(){
    window.history.forward();
}
```

navigator对象

navigator中有很多浏览器相关的内容，通常判断浏览器类型都是通过这个对象 其中比较常用的内容就是：如何判断各种版本的浏览器

```
console.log("完整的浏览器名称");
console.log(navigator.appName);
console.log("浏览器版本");
console.log(navigator.appVersion);
console.log("浏览器所在的系统平台");
console.log(navigator.platform);
console.log("浏览器的用户代理字符串");
console.log(navigator.userAgent);
```

userAgent 用户代理信息，通过该属性可以获取浏览器及 操作系统信息(重点了解)

常用事件方法

close() 关闭窗口

open() 打开窗口

```

window.onload = function(){
    document.getElementById("newWin").onclick = function(){
        window.open("1.html","", "height=100")
    }
    document.onclick=function(){
        window.close()
    }
}

```

window.open 弹出新窗口的命令；

'进度条.html' 弹出窗口的文件名；

'newwindow' 弹出窗口的名字（不是文件名），非必须，可用空''代替；

height=100 窗口高度；

width=400 窗口宽度；

top=0 窗口距离屏幕上方的象素值；

left=0 窗口距离屏幕左侧的象素值；

toolbar=no 是否显示工具栏，yes为显示；

menubar, 表示菜单栏。

onload 加载事件

onresize 窗口变化事件

定时器

开启定时器 setInterval (function(){alert(1)},time) 间隔执行的定时器

```

var timer,timer2,timer3 ;

timer = setInterval(function(){
    console.log("a")
},2000);
timer2 = setInterval(function(){
    console.log("b")
},2000);
timer3 = setInterval(function(){
    console.log("c")
},2000);

btn.onclick=function(){
    clearInterval(timer2)
    /*清除的时候把定时器序列的数字传进去*/
}

```

setTimeout(function(){alert(1)}, time) 延时执行的定时器

```
var timer,timer2,timer3 ;

timer = setTimeout(function(){
    console.log("a")
},2000);
timer2 = setTimeout(function(){
    console.log("b")
},2000);
timer3 = setTimeout(function(){
    console.log("c")
},2000);
document.onclick=function(){
    clearTimeout(timer2)
    /*清除的时候把定时器序列的数字传进去*/
}
```

//time是定时器间隔时间

清除定时器

clearInterval() clearTimeout()