

BOM

window对象

window对象，浏览器实例

两个身份：

- ES中Global对象
- 浏览器窗口的JavaScript接口

Global 作用域

var声明的所有全局变量和函数都会成为window对象的属性和方法

窗口关系

- **top** 指向最上层（最外层）窗口
- 如果最上层window不是window.open()打开的，name就不包含值
- **parent**指向当前窗口的父窗口
- **self**对象，始终指向window
- 如果当前窗口就是最外层窗口，self，top，parent对象都相等（都是window）

窗口位置与像素比

window对象的位置

- screenLeft、screenTop属性，窗口相对屏幕左侧和顶部的属性 返回css像素
- moveTo(), x、y绝对坐标
- moveBy(), x、y相对坐标

像素比

css像素是Web开发使用的统一像素单位 0.0213°

为了在不同设备上统一标准

不同像素密度屏幕有不同的缩放系数，把物理像素（分辨率）转换成css像素（浏览器虚拟分辨率）

缩放系数：window.devicePixelRatio 物理像素与逻辑像素缩放比例

DPI(dots per inch) 每英寸像素数，表示单位像素密度

窗口大小

浏览器支持4个属性

- innerWidth 窗口页面视口 宽 不包含浏览器边框和工具栏，但包括**滚动条**
- innerHeight 窗口页面视口 高 不包含浏览器边框和工具栏，但包括**滚动条**

- outerWidth 窗口自身 宽
- outerHeight 窗口自身 高

低版本IE浏览器[<IE9]不支持 innerWidth、innerHeight

替代方案:

- document.documentElement.clientWidth 不包含浏览器边框、工具栏、滚动条
- document.documentElement.clientHeight 不包含浏览器边框、工具栏、滚动条

```
let pageWidth = window.innerWidth,
    pageHeight = window.innerHeight;
if (typeof pageWidth !== "number") {
    // 检验页面是否处于标准模式
    if (document.compatMode === "CSS1Compat") {
        pageWidth = document.documentElement.clientWidth;
        pageHeight = document.documentElement.clientHeight;
    } else {
        // 非标准则用布局视口
        pageWidth = document.body.clientWidth;
        pageHeight = document.body.clientHeight;
    }
}
```

移动设备

- Mobile Internet Explorer
支持 window.innerWidth, window.innerHeight
document.documentElement.clientWidth、document.documentElement.clientHeight在放大缩小页面时，值会发生变化
- 其他移动浏览器
document.documentElement.clientWidth 和 document.documentElement.clientHeight返回布局视口大小，即渲染页面的实际大小

布局视口信息document.body.clientWidth 和 document.body.clientHeight

- resizeTo() 调整窗口大小，x、y绝对像素
- resizeBy() 调整窗口大小，x、y相对像素

很多浏览器禁用了上述缩放方法，即便能用也只有最上层的window对象能用

视口位置

度量文档相对于视口滚动举例的属性有两对

window.pageXOffset/window.scrollX和window.pageYOffset/window.scrollY

- scroll()
- scrollTo()
- scrollBy()

scroll()与scrollTo() 实际上是一样的

接收x、y参数

```
// 相对于当前视口向下滚动 100 像素
window.scrollBy(0, 100);
// 相对于当前视口向右滚动 40 像素
window.scrollBy(40, 0);
// 滚动到页面左上角
window.scrollTo(0, 0);
// 滚动到距离屏幕左边及顶边各 100 像素的位置
window.scrollTo(100, 100);
```

接收ScrollToOptions字典

```
// 正常滚动
window.scrollTo({
  left: 100,
  top: 100,
  behavior: 'auto'
});
// 平滑滚动
window.scrollTo({
  left: 100,
  top: 100,
  behavior: 'smooth'
});
```

导航与打开新窗口

window.open()

- 导航到指定URL
- 打开新浏览器窗口

参数：

- 要加载的URL
- 目标窗口 一个已经存在的窗口/窗格的名字
如果有这个名字的窗口则在这个窗口打开，否则就新建窗口并命名
特殊窗口：_self, _parent, _top, _blank
- 特性字符串 用来配置新窗口，如果是旧窗口则忽视
- 新窗口在浏览记录中是否替代当前加载页面（boolean）

```
// 与<a href="http://www.wrox.com" target="topFrame"/>相同
window.open("http://www.wrox.com/", "topFrame");
```

弹出窗口

window.open()如果第二个参数不是已有窗口，则会打开一个新窗口/标签页
第三个参数，特性字符串 用来配置新窗口，旧窗口则忽视，不传表默认

设 置	值	说 明
fullscreen	"yes"或"no"	表示新窗口是否最大化。仅限 IE 支持
height	数值	新窗口高度。这个值不能小于 100
left	数值	新窗口的 x 轴坐标。这个值不能是负值
location	"yes"或"no"	表示是否显示地址栏。不同浏览器的默认值也不一样。在设置为"no"时，地址栏可能隐藏或禁用（取决于浏览器）
Menubar	"yes"或"no"	表示是否显示菜单栏。默认为"no"
resizable	"yes"或"no"	表示是否可以拖动改变新窗口大小。默认为"no"
scrollbars	"yes"或"no"	表示是否可以在内容过长时滚动。默认为"no"
status	"yes"或"no"	表示是否显示状态栏。不同浏览器的默认值也不一样
toolbar	"yes"或"no"	表示是否显示工具栏。默认为"no"
top	数值	新窗口的 y 轴坐标。这个值不能是负值
width	数值	新窗口的宽度。这个值不能小于 100

用逗号隔开，不能出现空格

```
window.open("http://www.wrox.com/",
    "wroxWindow",
    "height=400,width=400,top=10,left=10,resizable=yes");
```

window.open()返回新建窗口的引用。这个对象与普通window对象没有区别，只是为了控制新窗口提供方便（有些浏览器不允许缩放或者移动主窗口，但是允许window.open()创建的窗口）

新建的窗口window对象有opener属性，指向打开它的窗口。只在弹出窗口的最上层window对象（top）有定义

安全限制

为了区分广告与系统对话框
多重安全限制

- 不能隐藏状态栏
- 不能隐藏地址栏
- 不允许出现在系统对话框的位置
- 在用户操作下才允许创建弹窗
- 网页加载中调用window.open()没有效果

弹窗屏蔽程序

window.open()返回值是null或者报错

测试是否内置弹窗屏蔽，需要考虑报错情况使用try...catch

```
let blocked = false;
try {
    let wroxWin = window.open("http://www.wrox.com", "_blank");
    if (wroxWin == null){
        blocked = true;
    }
} catch (ex){
    blocked = true;
}
```

```
if (blocked){  
    alert("The popup was blocked!");  
}
```

定时器

JS在浏览器单线程执行，允许使用定时器

严格来说定时器不是全局方法（不是Global）而是浏览器方法（window方法）

- setTimeout(callback,time)
返回值，一个表示该超时排期的数值 ID
唯一标识，用来取消任务 clearTimeout(ID)
- setInterval()
与setTimeout()使用方法一样
clearInterval(ID)

定时器的时间并不精确，而是在等待时间后加入队列（因为单线程）

系统对话框

系统对话框：

- 与网页无关
- 不包含html
- 无法使用css设置

阻塞的系统对话框

- alert()
只接受一个参数，显示的字符串（不是字符串回调用toString()）
- confirm()
确认框，返回布尔值，表示ok或者calcel
- prompt()
两个参数，显示字符串，文本框默认值
返回值，如果ok返回文本框内的值，若calcel/关闭，返回null
要显示给用户的文本，以及文本框的默认值(可以是空字符串)。调用 prompt("What is your name?", "Jake") 会显示图 12-3 所示的提示框。



非阻塞系统对话框

- find()
- print()

没有返回值，所以很难利用。对话框计数器不会涉及它们，禁用对话框也没有影响