BOM编程

全称 Browser Object Model，浏览器对象模型。

JavaScript是由浏览器中内置的javascript脚本解释器程序来执行javascript脚本语言的。

为了便于对浏览器的操作，javascript封装了对浏览器的各个对象使得开发者可以方便的操作浏览器。

Window 对象是 JavaScript 层级中的顶层对象。

Window 对象代表一个浏览器窗口或一个框架。

Window 对象会在 <body> 或 <frameset> 每次出现时被自动创建。

浏览器对象模型中把浏览器的各个部分都是用了一个对象进行描述，如果我们要操作浏览器的一些属性，我就可以通过浏览器对象模型 的对象进行操作。

window 代表了一个新开的窗口

location 代表了地址栏对象。

screen 代表了整个屏幕的对象

**1.window 对象**

window对象常用的方法：

open() 打开一个新的窗口。

resizeTo() 将窗口的大小更改为指定的宽度和高度值。

moveTo() 将窗口左上角的屏幕位置移动到指定的 x 和 y 位置。

moveBy() 相对于原来的窗口移动指定的x、y值。

setInterval() 每经过指定毫秒值后就会执行指定的代码。

clearInterval() 根据一个任务的ID取消的定时任务。

setTimeout() 经过指定毫秒值后执行指定 的代码一次。

注意： 使用window对象的任何属性与方法都可以省略window对象不写的。

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="UTF-8">  <title></title>  </head>  <body>    <input type="button" value="open" onclick="Testopen()" />  <input type="button" value="resizeTo" onclick="TestresizeTo()" />    <input type="button" value="moveTo" onclick="TestmoveTo()" />  <input type="button" value="moveBy" onclick="TestmoveBy()" />    <input type="button" value="clearInterval" onclick="TestclearInterval()" />    </body>      <script>  function Testopen(){  window.open("ad.html","\_blank","location=no,toolbar=no");  }    function TestresizeTo(){  window.resizeTo("400","400");  }    function TestmoveTo(){  window.moveTo("500","200");  }    function TestmoveBy(){  for(var i=0;i<100;i++){  window.moveBy(50,0);//右移  window.moveBy(0,50);//下移  window.moveBy(-50,0);//左移  window.moveBy(0,-50);//上移  }  }    //var id = window.setInterval("Testopen()",3000);    function TestclearInterval(){  window.clearInterval(id);  }    //window.setTimeout("Testopen()",1000);    </script>  </html> |
|  |

ad.html

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="UTF-8">  <title></title>  </head>  <body>  <img src="img/ad.png" /><font color="blue" size="7">广告来了....</font>  </body>    </html> |

**2.常用的事件**

鼠标点击相关：

onclick 在用户用鼠标左键单击对象时触发。

ondblclick 当用户双击对象时触发。

onmousedown 当用户用任何鼠标按钮单击对象时触发。

onmouseup 当用户在鼠标位于对象之上时释放鼠标按钮时触发。

鼠标移动相关：

onmouseout 当用户将鼠标指针移出对象边界时触发。

onmousemove 当用户将鼠标划过对象时触发。

焦点相关的：

onblur 在对象失去输入焦点时触发。

onfocus 当对象获得焦点时触发。

其他：

onchange 当对象或选中区的内容改变时触发。

onload 在浏览器完成对象的装载后立即触发。

onsubmit 当表单将要被提交时触发。

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="UTF-8">  <title></title>  </head>  <body <!--onload="showPage()"-->>    <input type="button" value="单击" onclick="Testonclick()" />    <input type="button" value="双击" ondblclick="Testondblclick()" />    <input type="button" value="按下鼠标" onmousedown="Testonmousedown()" />    <input type="button" value="释放鼠标" onmouseup="Testonmouseup()" />    <input type="button" value="显示时间" onmousemove="Testonmousemove()" onmouseout="Testonmouseout()" />  <span id="time"></span>    <hr>    用户名:<input type="text" id="username" name="username" onfocus="Testonfocus()" onblur="Testonblur()" />  <span id="user"></span>  密码:<input type="password" name="password" />    <select onchange="Testonchange()">  <option>上海</option>  <option>北京</option>  <option>深圳</option>  </select>    <form action="ad.html" onsubmit="Testonsubmit()">  <input type="submit" />  </form>    </body>    <script>    function Testonsubmit(){  alert("表单将要被提交..")  }    function Testonchange(){  alert("城市变了..")  }    function Testonblur(){  var usernameNode = document.getElementById("username");  var username = usernameNode.value;  if(username.length<=6){  document.getElementById("user").innerHTML = "用户名长度不足六位".fontcolor("red");  }  }    function Testonfocus(){  document.getElementById("user").innerHTML = "用户名长度必须大于六位".fontcolor("green");  }      function showPage(){  document.write("page");  }    function Testonclick(){  alert("单击");  }    function Testondblclick(){  alert("双击");  }    function Testonmousedown(){  alert("按下鼠标");  }    function Testonmouseup(){  alert("释放鼠标");  }    function Testonmousemove(){  document.getElementById("time").innerHTML = new Date().toLocaleString();  }    function Testonmouseout(){  document.getElementById("time").innerHTML = "";  }    </script>  </html> |

**3.location 对象**

Location 对象是由 JavaScript runtime engine 自动创建的，包含有关当前 URL 的信息。

location中的重要方法：

href属性 设置或获取整个 URL 为字符串。

reload() 重新装入当前页面

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="UTF-8">  <title></title>  </head>  <body>    <input type="button" value="新浪网" onclick="writeSina()" />    <input type="button" value="刷新" onclick="refresh()" />  </body>    <script>    document.writeln(location.href+"<br>");  document.writeln(location.host+"<br>");  document.writeln(location.hostname+"<br>");  document.writeln(location.port+"<br>");  document.writeln(location.protocol+"<br>");      function writeSina(){  location.href = "http://www.sina.com.cn";  }    function refresh(){  location.reload();  }    </script>  </html> |

**4.screen 对象**

Screen 对象是由 JavaScript runtime engine 自动创建的，包含有关客户机显示屏幕的信息。

Screen(屏幕)对象

availHeight 获取系统屏幕的工作区域高度，排除 Windows 任务栏。

availWidth 获取系统屏幕的工作区域宽度，排除 Windows 任务栏。

height 获取屏幕的垂直分辨率。

width 获取屏幕的水平分辨率。

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="UTF-8">  <title></title>  </head>  <body>  </body>    <script type="text/javascript">  document.write("获取系统屏幕的工作区域高度：" + screen.availHeight + "<br/>");  document.write("获取系统屏幕的工作区域宽度：" + screen.availWidth + "<br/>");  document.write("获取屏幕的垂直分辨率：" + screen.height + "<br/>");  document.write("获取屏幕的水平分辨率：" + screen.width + "<br/>");    </script>  </html> |

**5.随机点餐**

|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"  "http://www.w3.org/TR/html4/strict.dtd">  <html xmlns="http://www.w3.org/1999/xhtml" lang="en">  <head>  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />  <title></title>  </head>  <body onload="eat()">  <center>  <span id="food"></span>  <hr />    <input type="button" style="font-size: 50px;" value="暂停" onclick="stop()"/>    <input type="button" style="font-size: 50px;" value="继续" onclick="star()"/>    </center>  </body>    <script type="text/javascript">  function eat() {  var arr = ["盖浇饭", "油泼面", "米皮","兰州拉面"];  //随机获取下标  var index = Math.floor(Math.random() \* arr.length);  //获取数组中具体的数据  var code = arr[index];    var spanNode = document.getElementById("food");  spanNode.innerHTML = code;  //操作css样式  spanNode.style.fontSize = "124px";  spanNode.style.color = "red";  spanNode.style.backgroundColor = "yellow";  }    var id = setInterval("eat()",100);    function stop(){  clearInterval(id);  }    function star(){  location.reload();  }    </script>  </html> |