javascript函数一共可分为五类： ·常规函数 ·数组函数 ·日期函数 ·数学函数 ·字符串函数

1.常规函数

javascript常规函数包括以下9个函数：

(1)alert函数：显示一个警告对话框，包括一个OK按钮。

(2)confirm函数：显示一个确认对话框，包括OK、Cancel按钮。

(3)escape函数：将字符转换成Unicode码。

(4)eval函数：计算表达式的结果。

(5)isNaN函数：测试是(true)否(false)不是一个数字。

(6)parseFloat函数：将字符串转换成符点数字形式。

(7)parseInt函数：将符串转换成整数数字形式(可指定几进制)。

(8)prompt函数：显示一个输入对话框，提示等待用户输入。例如：<script language="javascript"> <!-- alert("输入错误"); prompt("请输入您的姓名","姓名");confirm("确定否！"); //--> script>

(9)unescape函数：解码由escape函数编码的字符。

2.数组函数

javascript数组函数包括以下4个函数：

(1)join函数：转换并连接数组中的所有元素为一个字符串。例:

function JoinDemo(){

var a, b;

a = new Array(0,1,2,3,4);

b = a.join("-");//分隔符

return(b);//返回的b=="0-1-2-3-4" }

(2)length函数：返回数组的长度。例：

function LengthDemo(){

var a, l;

a = new Array(0,1,2,3,4);

l = a.length;

return(l);//l==5 }

(3)reverse函数：将数组元素顺序颠倒。例：

function ReverseDemo(){

var a, l;

a = new Array(0,1,2,3,4);

l = a.reverse();

return(l); }

(4)sort函数：将数组元素重新排序。例：

function SortDemo(){

var a, l;

a = new Array("X" ,"y" ,"d", "Z", "v","m","r");

l = a.sort();

return(l); }

3.日期函数

javascript日期函数包括以下20个函数：

(1)getDate函数：返回日期的"日"部分，值为1～31。例：

function DateDemo(){

var d, s = "Today's date is: ";

d = new Date();

s += (d.getMonth() + 1) + "/";

s += d.getDate() + "/";

s += d.getYear();

return(s); }

(2)getDay函数：返回星期几，值为0～6，其中0表示星期日，1表示星期一，...，6表示星期六。例：

function DateDemo(){

var d, day, x, s = "Today is: ";

var x = new Array("Sunday", "Monday", "Tuesday");

var x = x.concat("Wednesday","Thursday", "Friday");

var x = x.concat("Saturday");

d = new Date();

day = d.getDay();

return(s += x[day]); }

(3) getHours函数：返回日期的"小时"部分，值为0～23。例。

function TimeDemo(){

var d, s = "The current local time is: ";

var c = ":";

d = new Date();

s += d.getHours() + c;

s += d.getMinutes() + c;

s += d.getSeconds() + c;

s += d.getMilliseconds();

return(s); }

(4)getMinutes函数：返回日期的"分钟"部分，值为0～59。见上例。

(5)getMonth函数：返回日期的"月"部分，值为0～11。其中0表示1月，2表示3月，...，11表示12月。见前面的例子。

(6)getSeconds函数：返回日期的"秒"部分，值为0～59。见前面的例子。

(7)getTime函数：返回系统时间。

function GetTimeTest(){

var d, s, t;

var MinMilli = 1000 \* 60;

var HrMilli = MinMilli \* 60;

var DyMilli = HrMilli \* 24;

d = new Date();

t = d.getTime();

s = "It's been "

s += Math.round(t / DyMilli) + " days since 1/1/70";

return(s); }

(8)getTimezoneOffset函数：返回此地区的时差(当地时间与GMT格林威治标准时间的地区时差)，单位为分钟。

function TZDemo(){

var d, tz, s = "The current local time is ";

d = new Date();

tz = d.getTimezoneOffset();

if (tz < 0)

s += tz / 60 + " hours before GMT";

else if (tz == 0)

s += "GMT";

else

s += tz / 60 + " hours after GMT";

return(s); }

(9)getYear函数：返回日期的"年"部分。返回值以1900年为基数，例如1999年为99。前面有例子。

(10)parse函数：返回从1970年1月1日零时整算起的毫秒数(当地时间)。

function GetTimeTest(testdate){

var d, s, t;

var MinMilli = 1000 \* 60;

var HrMilli = MinMilli \* 60;

var DyMilli = HrMilli \* 24;

d = new Date();

t = Date.parse(testdate);

s = "There are "；

s += Math.round(Math.abs(t / DyMilli)) + " days "；

s += "between " + testdate + " and 1/1/70";

return(s); }

(11)setDate函数：设定日期的"日"部分，值为1～31。

(12)setHours函数：设定日期的"小时"部分，值为0～23。

(13)setMinutes函数：设定日期的"分钟"部分，值为0～59。

(14)setMonth函数：设定日期的"月"部分，值为0～11。其中0表示1月，...，11表示12月。

(15)setSeconds函数：设定日期的"秒"部分，值为0～59。

(16)setTime函数：设定时间。时间数值为1970年1月1日零时整算起的毫秒数。

(17)setYear函数：设定日期的"年"部分。

(18)toGMTString函数：转换日期成为字符串，为GMT格林威治标准时间。

(19)setLocaleString函数：转换日期成为字符串，为当地时间。

(20)UTC函数：返回从1970年1月1日零时整算起的毫秒数，以GMT格林威治标准时间计算。

4.数学函数

javascript数学函数其实就是Math对象，它包括属性和函数(或称方法)两部分。其中，属性主要有下列内容：

Math.e:e(自然对数)、Math.LN2（2的自然对数)、Math.LN10(10的自然对数)、Math.LOG2E(e的对数，底数为2)、

Math.LOG10E(e的对数，底数为10)、Math.PI(π)、Math.SQRT1\_2(1/2的平方根值)、Math.SQRT2(2的平方根值)。

函数有以下18个：

(1)abs函数：即Math.abs(以下同)，返回一个数字的绝对值。

(2)acos函数：返回一个数字的反余弦值，结果为0～π弧度(radians)。

(3)asin函数：返回一个数字的反正弦值，结果为-π/2～π/2弧度。

(4)atan函数：返回一个数字的反正切值，结果为-π/2～π/2弧度。

(5)atan2函数：返回一个坐标的极坐标角度值。

(6)ceil函数：返回一个数字的最小整数值(大于或等于)。

(7)cos函数：返回一个数字的余弦值，结果为-1～1。

(8)exp函数：返回e(自然对数)的乘方值。

(9)floor函数：返回一个数字的最大整数值(小于或等于)。

(10)log函数：自然对数函数，返回一个数字的自然对数(e)值。

(11)max函数：返回两个数的最大值。

(12)min函数：返回两个数的最小值。

(13)pow函数：返回一个数字的乘方值。

(14)random函数：返回一个0～1的随机数值。

(15)round函数：返回一个数字的四舍五入值，类型是整数。

(16)sin函数：返回一个数字的正弦值，结果为-1～1。

(17)sqrt函数：返回一个数字的平方根值。

(18)tan函数：返回一个数字的正切值。

5.字符串函数

javascript字符串函数完成对字符串的字体大小、颜色、长度和查找等操作，共包括以下20个函数：

(1)anchor函数：产生一个链接点(anchor)以作超级链接用。anchor函数设定的链接点的名称，另一个函数link设定的URL地址。

(2)big函数：将字体加到一号，与...标签结果相同。

(3)blink函数：使字符串闪烁，与...标签结果相同。

(4)bold函数：使字体加粗，与...标签结果相同。

(5)charAt函数：返回字符串中指定的某个字符。

(6)fixed函数：将字体设定为固定宽度字体，与...标签结果相同。

(7)fontcolor函数：设定字体颜色，与标签结果相同。

(8)fontsize函数：设定字体大小，与标签结果相同。

(9)indexOf函数：返回字符串中第一个查找到的下标index，从左边开始查找。

(10)italics函数：使字体成为斜体字，与...标签结果相同。

(11)lastIndexOf函数：返回字符串中第一个查找到的下标index，从右边开始查找。

(12)length函数：返回字符串的长度。(不用带括号)

(13)link函数：产生一个超级链接，相当于设定的URL地址。

(14)small函数：将字体减小一号，与...标签结果相同。

(15)strike函数：在文本的中间加一条横线，与...标签结果相同。

(16)sub函数：显示字符串为下标字(subscript)。

(17)substring函数：返回字符串中指定的几个字符。

(18)sup函数：显示字符串为上标字(superscript)。

(19)toLowerCase函数：将字符串转换为小写。

(20)toUpperCase函数：将字符串转换为大写。

事件源对象 event.srcElement.tagName event.srcElement.type

捕获释放 event.srcElement.setCapture(); event.srcElement.releaseCapture();

事件按键 event.keyCode event.shiftKey event.altKey event.ctrlKey

事件返回值 event.returnValue

鼠标位置 event.x event.y

窗体活动元素 document.activeElement

绑定事件 document.captureEvents(Event.KEYDOWN);

访问窗体元素 document.all("txt").focus(); document.all("txt").select();

窗体命令 document.execCommand

窗体COOKIE document.cookie

菜单事件 document.oncontextmenu

创建元素 document.createElement("SPAN");

根据鼠标获得元素： document.elementFromPoint(event.x,event.y).tagName=="TD document.elementFromPoint(event.x,event.y).appendChild(ms)

窗体图片 document.images[索引]

窗体事件绑定 document.onmousedown=scrollwindow;

元素 document.窗体.elements[索引]

对象绑定事件 document.all.xxx.detachEvent('onclick',a);

插件数目 navigator.plugins

取变量类型 typeof($js\_libpath) == "undefined"

下拉框 下拉框.options[索引] 下拉框.options.length

查找对象 document.getElementsByName("r1"); document.getElementById(id);

定时 timer=setInterval('scrollwindow()',delay); clearInterval(timer);

UNCODE编码 escape() ,unescape

父对象 obj.parentElement(dhtml) obj.parentNode(dom)

交换表的行 TableID.moveRow(2,1) //将第三行与第二行交换

替换CSS document.all.csss.href = "a.css";

并排显示 display:inline

隐藏焦点 hidefocus=true

根据宽度换行 style="word-break:break-all"

自动刷新 <meta HTTP-EQUIV="refresh" CONTENT="8;URL=http://hi.baidu.com">

简单邮件 <a href="mailto:jishu@bj08ay.cn?subject=thisistitle&body=thisisthewordswhatyouwanttowriteforme">

快速转到位置 obj.scrollIntoView(true)

锚 <a name="first"> <a href="#first">anchors</a>

网页传递参数 location.search();

可编辑 obj.contenteditable=true

执行菜单命令 obj.execCommand

双字节字符 /[^\x00-\xff]/

汉字 /[\u4e00-\u9fa5]/

让英文字符串超出表格宽度自动换行 word-wrap: break-word; word-break: break-all;

透明背景 <IFRAME src="1.htm" width=300 height=180 allowtransparency></iframe>

获得style内容 obj.style.cssText

HTML标签 document.documentElement.innerHTML

第一个style标签 document.styleSheets[0]

style标签里的第一个样式 document.styleSheets[0].rules[0]

防止点击空链接时，页面往往重置到页首端 <a href="javascript:function()">word</a>

上一网页源 asp: request.servervariables("HTTP\_REFERER") javascript: document.referrer

释放内存 CollectGarbage();

禁止右键 document.oncontextmenu = function() { return false;}

禁止保存 <noscript><iframe src="\*.htm"></iframe></noscript>

禁止选取 <body>

禁止粘贴 <input type=text>

地址栏图标 <link rel="Shortcut Icon" href="favicon.ico">

favicon.ico 名字最好不变16\*16的16色,放虚拟目录根目录下

收藏栏图标 <link rel="Bookmark" href="favicon.ico">

查看源码 <input type=button value=查看网页源代码>

关闭输入法 <input style="ime-mode:disabled">

自动全选 <input type=text name=text1 value="123">

ENTER键可以让光标移到下一个输入框 <input>

文本框的默认值 <input type=text value="123">

获得时间所代表的微秒 var n1 = new Date("2004-10-10".replace(/-/g, "\/")).getTime()

窗口是否关闭 win.closed checkbox

扁平 <input type=checkbox style="position: absolute; clip:rect(5px 15px 15px 5px)"><br>

获取选中内容 document.selection.createRange().duplicate().text

自动完成功能 <input type=text autocomplete=on>打开该功能 <input type=text autocomplete=off>关闭该功能

窗口最大化 <body>

无关闭按钮IE window.open("aa.htm", "meizz", "fullscreen=7");

统一编码/解码 alert(decodeURIComponent(encodeURIComponent("http://你好.com?as= hehe"))) encodeURIComponent对":"、"/"、";" 和 "?"也编码

表格行指示 <tr>

各种尺寸

s += "\r\n网页可见区域宽："+ document.body.clientWidth;

s += "\r\n网页可见区域高："+ document.body.clientHeight;

s += "\r\n网页可见区域高："+ document.body.offsetWeight +" (包括边线的宽)";

s += "\r\n网页可见区域高："+ document.body.offsetHeight +" (包括边线的宽)";

s += "\r\n网页正文全文宽："+ document.body.scrollWidth;

s += "\r\n网页正文全文高："+ document.body.scrollHeight;

s += "\r\n网页被卷去的 高："+ document.body.scrollTop;

s += "\r\n网页被卷去的左："+ document.body.scrollLeft;

s += "\r\n网页正文部分上："+ window.screenTop;

s += "\r\n网页正文部分左："+ window.screenLeft;

s += "\r\n屏幕分辨率的高："+ window.screen.height;

s += "\r\n屏幕分辨率的宽："+ window.screen.width;

s += "\r\n屏幕可用工作区高度："+ window.screen.availHeight;

s += "\r\n屏幕可用工作区宽度："+ window.screen.availWidth;