

# 关于本文档的说明

本文档由太平洋软件资讯栏目制作,版权归太平洋软件资讯站 及文档作者 所有。制作该文档的目的是为了方便广大网友。因此任何人都可以自由复制,或 在网络中散布该文档。但不允许将该文档用于出版或其它商业用途,同时使用该 文档的过程中,未经允许,不得对该文档进行修改及反编译。如有问题请与我们 联系谢谢!

http://shope.taobao.com/



太平洋软件咨询, http://www.pcsoftware.com.cn 在这里你将找到你想要的东西。来看看吧,每天都有大量更新,不会让朋友们 失望的。

> 编者按: JavaScript是由Netscape公司开发并随Navigator导航者一起发布的、介于Java与HTML之间、基于 对象事件驱动的编程语言,正日益受到全球的关注。因它的开发环境简单,不需要Java编译器,而是直接 运行在Web浏览器中,而因倍受Web设计者的所爱。为了帮助我国广大计算机用户尽快了解和掌握这些新技 术,本着先进的技术,要大家分享的原则,电脑网络学校特组织JavaScript技术讲座。其各讲内容为:

■第一讲 JavaScript语言概况

■第二讲 JavaScript基本数据结构

■第三讲 JavaScript程序构成

■第四讲 基于对象的JavaScript语言

第五讲 创建新对象
第六讲 使用 的对象系统
第七讲 窗口及输入 他出
第八讲 WEB页面信息交互
中 实现更复杂交互
中 实现更复杂交互

إمارها مازما إمازها والمارما والمازما إمازها

# 第一讲 JavaScript 高声概况

Internet时代,造就了我们新的工作和生活方式,其互联性、并放性和共享信息的模式,打破了传统信息传播方式的重重壁垒,为我们带来了新的机遇。随着计算机和信息时代的到来,人类社会前进的脚步 在逐渐加快,每一天都有新的事情发生,每一天都在创造着奇迹。随着Internet技术的突飞猛进,各行各业都在加入Internet的行业中来。无论从管理方面,还是从商业角度来看,Lingret都可以带来无限生机。通过Internet,可以实现地区、集体乃至个人的连接,从而达到一种"统"的和谐"。那么怎样把自己的或公司的信息资源加入到 WWW 服务器,是广大用户日益关心的问题。采用超链技术(超文本和超媒体 技术)是实现这个目标最简单的、最快速的手段和途径。具体实现这种手段的支持环境,那就是HTML超文 本标识语言。通过它们可制作所需的Web网页。

通过超文本(Hyper Text)和超媒体(Hyper Media)技术结合超链接(Hyper link)的链接功能将各种信息 组织成网络结构(web),构成网络文档(Document),实现Internet上的"漫游"。通过HTML符号的描述就可 以实现文字、表格、声音、图像、动画等多媒体信息的检索。

然而采用这种超链技术存在有一定的缺陷,那就是它只能提供一种静态的信息资源,缺少动态的客户 端与服务器端的交互。虽然可通过CGI(Common Gateway Interface)通用网关接口实现一定的交互,但由 于该方法编程较为复杂,因而在一段时间防碍了Internet技术的发展。而JavaScript的出现,无凝为 Internet网上用户带来了一线生机。可以这样说, JavaScript的出现是时代的需求, 是当今的信息时代造就 了JavaScript。

JavaScript的出现,它可以使得信息和用户之间不仅只是一种显示和浏览的关系,而是实现了一种实 时的、动态的、可交式的表达能力。从而基于CGI静态的HTML页面将被可提供动态实时信息,并对客户操作 进行反应的Web页面的取代。JavaScript脚本正是满足这种需求而产生的语言。它深受广泛用户的喜爱的欢 迎。它是众多脚本语言中较为优秀的一种,它与WWW的结合有效地实现了网络计算和网络计算机的蓝图。无 凝Java家族将占领Internet网络的主导地位。因此,尽快掌握JavaScript脚本语言编程方法是我国广大用 户日益关心的。

### 一、什么是JavaScript

JavaScript是一种基于对象(Object)和事件驱动(Event Driven)并具有安全性能的脚本语言。使用它

This document is 所會特定學所提起了本來語語。ofaCHMAPEE Pilota不是多2-起实现在一个Web页面中链接多个对 象,与Web客户交互作用。从而可以开发客户端的应用程序等。它是通过嵌入或调入在标准的HTML语言中实 现的。它的出现弥补了HTML语言的缺陷,它是Java与HTML折衷的选择,具有以下几个基本特点:

#### 1. 是一种脚本编写语言

JavaScript是一种脚本语言,它采用小程序段的方式实现编程。像其它脚本语言一样, JavaScript 同样已是一种解释性语言,它提供了一个易的开发过程。

它的基本结构形式与C、C++、VB、Delphi十分类似。但它不像这些语言一样,需要先编译,而是在 程序运行过程中被逐行地解释。它与HTML标识结合在一起,从而方便用户的使用操作。

2. 基于对象的语言。

JavaScript是一种基于对象的语言,同时以可以看作一种面向对象的。这意味着它能运用自己已经 创建的对象。因此,许多功能可以来自于脚本环境中对象的方法与脚本的相互作用。

3. 简单性

JavaScript的简单性主要体现在: 首先它是一种基于Java基本语句和控制流之上的简单而紧凑的设 计,从而对于学习Java是一种非常好的过渡。其次它的变量类型是采用弱类型,并未使用严格的数 据类型。

4. 安全性

JavaScript是一种安全性语言,它不允许访问本地的硬盘,并不能将数据存入到服务器上,不允许 对网络文档进行修改和删除,只能通过浏览器实现信息浏览或动态交互。从而有效地防止数据的丢 失。

5. 动态性的

JavaScript是动态的,它可以直接对用户或客户输入做出响应,无须经过Web服务程序。它对用户 的反映响应,是采用以事件驱动的方式进行的。所谓事件驱动,就是指在主页(Home Page)中执行 了某种操作所产生的动作,就称为"事件"(Event)。比如按下鼠标、移动窗口、选择菜单等都可 以视为事件。当事件发生后,可能会引起相应的事件响应。

6. 跨平台性

跨平台性
JavaScript是依赖子浏览器本身,与操作环境无关,只要能运行浏览器的计算机,并支持
JavaScript的浏览器就可正确执行。从而实现了"编写一次,走遍天下"的梦想。

实际上JavaScript最杰出之处在了以用很小的程序做大量的事。无须有高性能的电脑,软件仅需一个字处理软件及一浏览器,无须wein的务器通道,通过自己的电脑即可完成所有的事情。

综合所述JavaScript 是一种新作品述语言,它可以被箝入到 HTML 的文件之中。 JavaScript语言可以做到回应使用者的需求事件(如: four 的输入),而不用任何的网路来回传输资料,所以当一位使用者输入一项资料时,它不用多述传统伺服端(server)处理,再传回来的过程,而直接可以被客户端(client)的应用程式所处理。

JavaScript 和 Java 很类似,但到底心。语言,而 JavaScript 则是相当容易了解的语言。 Javacon所以许多 Java 的特性在 Java Script 中并不支援 JavaScript 和 Java 很类似,但到底并不一步! Java 是一种比 JavaScript 更复杂许多的程式语言,而 JavaScript 则是相当容易了解的语言。 JavaScript 创作者可以不那麼注重程式技巧,

#### 二、JavaScript和Java的区别

虽然JavaScript与Java有紧密的联系,但却是两个公司开发的不同的两个产品。Java是SUN公司推出的 新一代面向对象的程序设计语言,特别适合于Internet应用程序开发;而 a aScript是Netscape公司的产品,其目的是为了扩展Netscape Navigator功能,而开发的一种可以嵌入Web而每中的基于对象和事件驱动的解释性语言,它的前身是Live Script;而Java的前身是Oak语言。下面对两种语言间的异同作如下比 较:

#### 1. 基于对象和面向对象

Java是一种真正的面向对象的语言,即使是开发简单的程序,必须设计对象。

JavaScript是种脚本语言,它可以用来制作与网络无关的,与用户交互作用的复杂软件。它是一种 基于对象(Object Based)和事件驱动(Event Driver)的编程语言。因而它本身提供了非常丰富 的内部对象供设计人员使用。

2. 解释和编译

两种语言在其浏览器中所执行的方式不一样。Java的源代码在传递到客户端执行之前,必须经过编 译,因而客户端上必须具有相应平台上的仿真器或解释器,它可以通过编译器或解释器实现独立于 某个特定的平台编译代码的束缚。

JavaScript是一种解释性编程语言,其源代码在发往客户端执行之前不需经过编译,而是将文本格 式的字符代码发送给客户编由浏览器解释执行。

3. 强变量和弱变量

两种语言所采取的变量是不一样的。

Java采用强类型变量检查,即所有变量在编译之前必须作声明。如:

Integer x:

String y;

x=1234;

x=4321;

其中X=1234说明是一个整数,Y=4321说明是一个字符串。 JavaScript中变量声明,采用其弱类型。即变量在使用前不需作声明,而是解释器在运行时检查其 x=1234;

y = "4321";

前者说明x为其数值型变量,而后者说明y为字符型变量。

# 4. 代码格式不一样

Java是一种与HTML无关的格式,必须通过像HTML中引用外媒体那么进行装载,其代码以字节代码的 形式保存在独立的文档中。

JavaScript的代码是一种文本字符格式,可以直接嵌入HTML文档中,并且可动态装载。编写HTML文档就像编辑文本文件一样方便。

- 5. 嵌入方式不一样
  - 在HTML文档中,两种编程语言的标识不同,JavaScript使用〈Script〉...〈/Script〉来标识,而Java使用〈applet〉...〈/applet〉来标识。
- 6. 静态联编和动态联编

Java采用静态联编,即Java的对象引用必须在编译时的进行,以使编译器能够实现强类型检查。 JavaScript采用动态联编,即JavaScript的对象引用在运行时进行检查,如不经编译则就无法实现 对象引用的检查。



# 四、编写第一个JavaScript程序

下面我们通过一个例子,编写第一个JavaScript程序。通过它可说明JavaScript的脚本是怎样被嵌入到HTML文档中的。

test1.html文档:

<html>

<head>

<Script Language ="JavaScript">

// JavaScript Appears here.

alert("这是第一个JavaScript例子!");

alert("欢迎你进入JavaScript世界!");

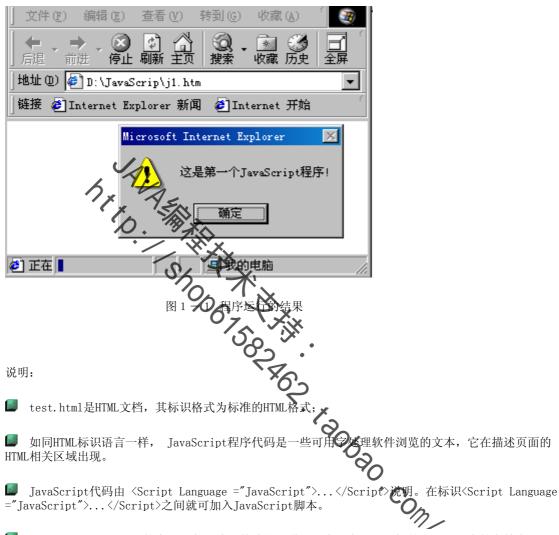
alert("今后我们将共同学习JavaScript知识!");

</Script>

</Head>

</Html>

在Internet Explore5.0中运行行后的结果见图 1-1 所示。



- alert()是JavaScript的窗口对象方法,其功能是弹出一个具有OK对话框并显示()中的字符串。
- 通过<!-- ... //-->标识说明:若不认识JavaScript代码的浏览器,则所有在其中的标识均被忽略;若认识,则执行其结果。使用注释这是一个好的编程习惯,它使其他人可以读懂你的语言。
- JavaScript 以〈/Script〉标签结束。

从上面的实例分析中我们可以看出,编写一个JavaScript程序确实非常容易的。



太平洋软件咨询, http://www.pcsoftware.com.cn 在这里你将找到你想要的东西。来看看吧,每天都有大量更新,不会让朋友们 失望的。

# 第二讲 JavaScript基本数据结构

JavaScript提供脚本语言的编程与C++非常相似,它只是去掉了C语言中有关指针等容易产生的错误,并提供了功能强大的类库。对于已经具备C++或C语言的人来说,学习JavaScript脚本语言是一件非常轻松愉快的事。

# 一、JavaScript代码的加入

JavaScript的脚本包括在HTML中,它成为HTML文档的一部分。与HTML标识相结合,构成了一个功能强大的Internet网上编程语言。可以直接将JavaScript脚本加入文档:

# <Script Language = "JavaScript">



在浏览器的窗口中调用test2.html,则显示"这是电脑报网络学校"字串。见图2所示。

This document is created the light state. ated with trial version ₩2PDF Pilot 2.15.82.



图2

#### 说明:

■ Document. write()是文档对象的输出函数,其功能是将括号中的字符或变量值输出到窗口; document. close()是将输出关闭。

可将〈Scribty. .</Script>标识放入head>..</Head>或<Body> ...</Body>之间。将 JavaScrict标识放着《Head》... 〈/Head〉在头部之间,使之在主页和其余部分代码之前装载,从而可使代码的功能更强大,可以将JavaScript标识放置在〈Body〉... 〈/Body〉主体之间以实现某些部 分动态地创建文档。

分动态地创建发档。

二、基本数据类型

JavaScript脚本语言同其它语言 有 有 本框架结构。JavaScript提供了四种基本的数据表式则可以完成较复杂的信息处理 表达式则可以完成较复杂的信息处理。

#### 1、基本数据类型

在JavaScript中四种基本的数据类型:数值(整数和实数)、字符串型(用""号或''括起来的字符或数值)、布尔型(使True或False表示)和空值。在JavaScript的基本类型中的数据可以是常量,也可以变量。由于JavaScript采用弱类型的形式,因而一个数据的变量或常量不必首先作声明,而是在使用或赋值时确定其数据的类型的。当然也可以先声明该数据的类型,是通过在赋值时自动说明其数据类型的 SO COM 的。

# 2、常量

#### ■ 整型常量

JavaScript的常量通常又称字面常量,它是不能改变的数据。其整型常量可以使用十六进制、八进 制和十进制表示其值。

# ■ 实型常量

实型常量是由整数部分加小数部分表示,如12.32、193.98。可以使用科学或标准方法表示: 5E7、4e5等。

# ■ 布尔值

布尔常量只有两种状态: True或False。 它主要用来说明或代表一种状态或标志,以说明操作流 程。它与C++是不一样的, C++可以用1或0表示其状态, 而JavaScript只能用True或False表 示其状态。

### ■ 字符型常量

使用单引号(')或双引号(")括起来的一个或几个字符。如 "This is a book of JavaScript 、"3245"、"ewrt234234" 等。

#### ■ 空値

This document is created with trial version of CHM2PDF Pilot 2.15.82
JavaScript中有一个空值null,表示什么也没有。如试图引用没有定义的变量,则返回一个Null值。

#### ■ 特殊字符

同C语言一样,JavaScript中同样以有些以反斜杠(/)开头的不可显示的特殊字符。通常称为控

# 3、变量

变量的主要作用是存取数据、提供存放信息的容器。对于变量必须明确变量的命名、变量的类型、变 量的声明及其变量的作用域。

### ■ 变量的命名

JavaScript中的变量命名同其计算机语言非常相似,这里要注意以下两点:

- A、必须是一个有效的变量,即变量以字母开头,中间可以出现数字如test1、text2等。除下划线 (一)作为连字符外,变量名称不能有空格、(+)、(一)、(,)或其它符号。
- B、不能使用JavaScript中的关键字作为变量。

4 0 多个类键字,这些关键是JavaScript内部使用的,不能作为变量的名 在JavaScript true不能作为变量的名称。

在对变量命名时人员 与其代表的意思对应起来,以免出现错误。

变量的类型

在JavaScript中,

var mytest;

国命令Var作声明. 该例子定义了一个mytest变量。

Var mytest=" This is a book'

该例子定义了一个mytest变量,同时赋予了它的值

如:

x = 100

y="125"

xy= True

cost=19.5等。

其中x整数, y为字符串, xy为布尔型, cost为实型。

# ■ 变量的声明及其作用域

JavaScript变量可以在使用前先作声明,并可赋值。通过使用var关键字对变量作声明。对变量作 声明的最大好处就是能及时发现代码中的错误;因为JavaScript是采用动态编译的,而动态编译是 不易发现代码中的错误,特别是变量命名的方面。

对于变量还有一个重要性——那就是变量的作用域。在JavaScript中同样有全局变量和局部变量。 全局变量是定义在所有函数体之外,其作用范围是整个函数;而局部变量是定义在函数体之内,只 对其该函数是可见的,而对其它函数则是不可见的。

# 三、表达式和运算符

在定义完变量后,就可以对它们进行赋值、改变、计算等一系列操作,这一过程通常又叫称一个叫表 达式来完成,可以说它是变量、常量、布尔及运算符的集合,因此表达式可以分为算术表述式、字串表达 式、赋值表达式以及布尔表达式等。

# 2、运算符

运算符完成操作的一系列符号,在JavaScript中有算术运算符,如+、-、\*、/等;有比较运算符如!=、==等;有逻辑布尔运算符如!(取反)、|、||;有字串运算如+、+=等。 在JavaScript主要有双目运算符和单目运算符。其双目运算符由下列组成:

#### 操作数1 运算符 操作数2

即由两个操作数和一个运算符组成。如50+40、"This"+"that"等。单目运算符,只需一个操作数,其 运算符可在前或后。

#### (1)算术运算符

JavaScript中的算术运算符有单目运算符和双目运算符。

双目运算符:

\*(乘)、/(除)、%(取模)、|(按位或)、&(按位与)、<< (右移)、 >>>(右移,零填充)。

单目运算符:

(递加1)、--(递减1)。

# (2)比较运算符

数进行比较,尔后再返回一个true或False值,有8 比较运算符它的基本操作过程是 个比较运算符:

=(等于)、!=(不等于)。 〈(小于)、〉(大于)、〈=(小于等于)、〉=(大

#### (3)布尔逻辑运算符

在JavaScript中增加了几个布尔逻辑运算符:

#运算符

ipt中增加了几个布尔逻辑运算符:

! (取反)、&= (与之后赋值)、 & (逻辑与)、 |= 0 戊之后赋值)、 | (逻辑或)、 ^= (异或之后赋值)、 ^ (逻辑异或)、 ?: (三目操作符)、 || (或)、==(等于)、 |=(不等于)、 !(取反)、&=(与之后赋值)、 &(逻辑与) 于)。

其中三目操作符主要格式如下:

# 操作数?结果1:结果2

若操作数的结果为真,则表述式的结果为结果1,否则为结果2。

#### 四、范例

下面是一个跑马灯效果的JavaScript文档。

 $Test2_1.html$ 

<html>

<head>

<script Language="JavaScript">

var msg="这是一个跑马灯效果的JavaScript文档";

var interval = 100;

```
var spacelen = 120;
var space10="";
var seq=0;
function Scroll() {
            len = msg.length;
            window.status = msg.substring(0, seq+1);
            seq++;
            if ( seq \ge len ) {
            seq = spacelen;
            window.setTimeout("Scrol12();", interval );
          function Scroll2() {
                                      Sens. Acts aobao com
                                              •i++) out +=
            space10;
            out = out + msg;
            len=out.length;
            window.status=out.substring(seq, len);
            seg++;
            if ( seq >= len ) { seq = 0; };
            window.setTimeout("Scrol12();", interval );
Scroll();
</script>
<body>
</body>
</html>
```

本讲介绍了JavaScript脚本是如何加入Web页面,并学习了JavaScript语言中的基本数据类型、变量、常量、操作运算符等。从本讲中的内容中可以看出,对于已经掌握C++语言的人来说,学习JavaScript真是一件非常轻松愉快的事。



太平洋软件咨询,http://www.pcsoftware.com.cn 在这里你将找到你想要的东西。来看看吧,每天都有大量更新,不会让朋友们 失望的。

# 第三讲 JavaScript程序构成

JavaScript脚本语言的基本构成是由控制语句、函数、对象、方法、属性等,来实现编程的。

### 一、程序控制流

在任何一种语言中,程序控制流是必须的,它能使得整个程序减小混乱,使之顺利按其一定的 方式执行。下面是JavaScript常用的程序控制流结构及语句:

# 1、if条件语句

# ■ 基本格式



■ 功能: 若表达式为true,则势

# ■ 说明:

• if -else 语句是JavaScript中最基本的控制语句 通过它可以改变语句的执行顺序。

, con

- 表达式中必须使用关系语句,来实现判断,它是作为 布尔值来估算的。 它终季和非素的教分别转化或falso和true
- 它将零和非零的数分别转化成false和true。
- 若if后的语句有多行,则必须使用花括号将其括起来。

#### 4. if语句的嵌套

if (布尔值)语句1;

else (布尔值) 语句 2;

else if (布尔值)语句3;

.....

else 语句4;

在这种情况下,每一级的布尔表述式都会被计算,若为真,则执行其相应的语句,否则执行 else后的语句。

for (初始化;条件;增量)

语句集;

- 功能:实现条件循环,当条件成立时,执行语句集,否则跳出循环体。
- 说明:
- 初始化参数告诉循环的开始位置,必须赋予变量的初值;
- 条件: 是用于判别循环停止时的条件。若条件满足,则执行循环体, 否则 跳出。
- 增量: 主要定义循环控制变量在每次循环时按什么方式变化。
- 三个主要语句之间,必须使用逗号分隔。

#### 3、while循环

■ 基本格式

クメリカ语句集

该语句(For语》),,当条件为真时,重复循环,否则退出循环。

■ For与while语句

两种语句都是循环语句,使用tor语句在处理有关数字时更易看懂,也较紧凑;而while循环对复杂的语句效果更特别。

4、break和continue语句

与C++语言相同,使用break语句使得循环,为r或while中跳出,continue使得跳过循环内剩余的语句而进入下一次循环。

#### 二、函数

函数为程序设计人员提供了一个丰常方便的能力。通常在设定一个复杂的程序设计时,总是根据所要完成的功能,将程序划分为一些相对独立的部分,每部分编写一个函数。从而,使各部分充分独立,任务单一,程序清晰,易懂、易读、易维护。JavaScript 的数可以封装那些在程序中可能要多次用到的模块。并可作为事件驱动的结果而调用的程序。从而实现一个函数把它与事件驱动相关联。这是与其它语言不样的地方。

# 1、JavaScript函数定义

Function 函数名 (参数,变元) {

函数体;.

Return 表达式;

说明:

}

- 当调用函数时, 所用变量或字面量均可作为变元传递。
- 函数由关键字Function定义。
- 函数名:定义自己函数的名字。
- 参数表,是传递给函数使用或操作的值,其值可以是常量 , 变量或其它表达式。

- 通过指定函数名(实参)来调用一个函数。
- 必须使用Return将值返回。
- 函数名对大小写是敏感的。

# 2、函数中的形式参数:

在函数的定义中,我们看到函数名后有参数表,这些参数变量可能是一个或几个。那么怎样才 能确定参数变量的个数呢?在JavaScript中可通过arguments.Length来检查参数的个数。例:

Function function\_Name(exp1, exp2, exp3, exp4) Number =function \_Name . arguments .length; (Number)

cument.write(e...

f (Number>2)

document.write(exp3);

f(Number>3)

write(exp4);

JavaScript是基于对象(object-based)的语言。这 Saya不同, Java是面向对象的语言。而基于对象的基本特征,就是采用事件驱动(event-driven)。 是在用形界面的环境下,使得一切输入变化简单化。通常鼠标或热键的动作我们称之为事件(Event),而由鼠标或热键引发的一连 串程序的动作,称之为事件驱动(Event Driver)。而对事件进行处理程序或函数,我们称之为 事件处理程序(Event Handler)。

### 2、事件处理程序

在JavaScript中对象事件的处理通常由函数(Function)担任。其基本格式与函数全部一样, 可以将前面所介绍的所有函数作为事件处理程序。格式如下:

Function 事件处理名(参数表){

事件处理语句集;

# 3、事件驱动

JavaScript事件驱动中的事件是通过鼠标或热键的动作引发的。它主要有以下几个事件:

# (1) 单击事件onClick

当用户单击鼠标按钮时,产生onClick事件。同时onClick指定的事件处理程序或代码将被调 用执行。通常在下列基本对象中产生:

- checkbox (复选框)或(检查列表框)
- radio (单选钮)
- reset buttons (重要按钮)
- submit buttons (提交按钮)

例:可通过下列按钮激活change()文件:

<Form>

<Input type="button" Value= " " onClick="change()">

</Form>

在onClick等号后,可以使用自己编写的函数作为事件处理程序,也可以使用JavaScript中 内部的函数。还可以直接使用JavaScript的代码等。例:

<Input type="button" value=" " onclick=alert("这是一个例子");</pre>

(2) onChange改变事件

当利用text或texturea元素输入字符值改变时发该事件,同时当在select表格项中一个选项

name= Test va.

一被加亮后,对发该事件。 value="Test" onCharge="check('this.test)"> <Input type=</pre>

</Form>

(3) 选中事件onSelect

当Text或Textarea对象中的文

(4) 获得焦点事件onFocus

当用户单击Text或textarea以及select对象时, 产生该事件。此时该对象成为前台对象。

(5) 失去焦点onBlur

当text对象或textarea对象以及select对象不再拥有焦点。而退到后台时,引发该文件,他 与onFocas事件是一个对应的关系。

(6) 载入文件onLoad

当文档载入时,产生该事件。onLoad一个作用就是在首次载入一个文档时检测cookie的值, 并用一个变量为其赋值, 使它可以被源代码使用。

(7) 卸载文件onUnload

当Web页面退出时引发onUnload事件,并可更新Cookie的状态。

四、范例

范例1: 下例程序是一个自动装载和自动卸载的例子。即当装入HTML文档时调用loadform()函 数,而退出该文档进入另一HTML文档时则首先调用unloadform()函数,确认后方可进入。

 $test3_1.htm$ 

<HTML>

<HEAD>

<script Language="JavaScript">

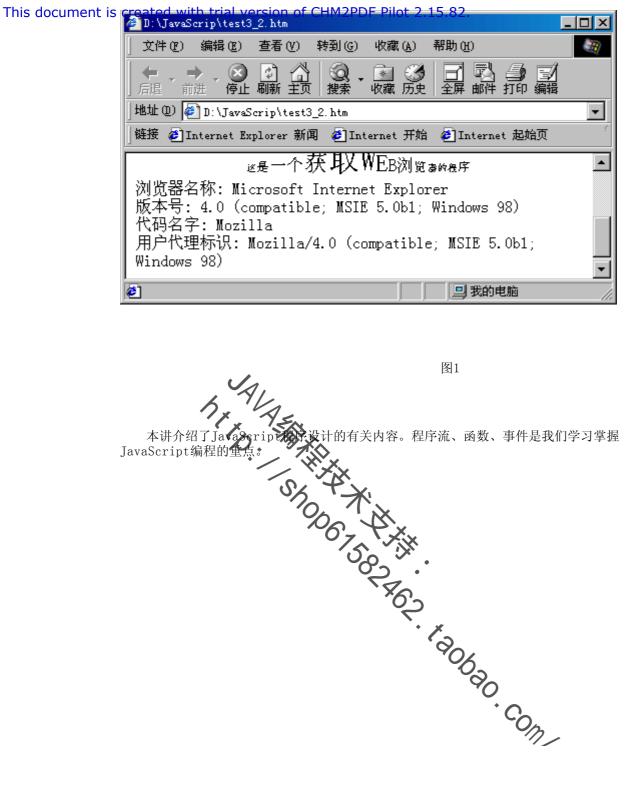
```
function loadform() {
                    alert("这是一个自动装载例子!");
       function unloadform() {
                    alert("这是一个卸载例子!");
       //-->
       </Script>
《HEAD》
《BODY OnLoad="loadform"、
《a href="test_htm"開用《/a》
《/BODY》
《/HTML》

范例2: 这是一个获取浏览器版本等研究。 家庭产首先显示一个波浪一提示信息。之后显示浏览器的版本号有关信息。
test3_2.htm
《html》

**TavaScript"》
《!--
       </HEAD>
       this.length=n
       return this
       function hexfromdec(num) {
       hex=new makeArray(1);
       var hexstring="";
       var shifthex=16;
       var temp1=num;
       for (x=1; x>=0; x--) {
       hex[x]=Math.round(temp1/shifthex - .5);
```

```
temp1=hex[x-1];
shifthex /= 16;
for (x=1; x>=0; x--) { hexstring+=getletter(hex[x]); }
return (hexstring);
}
function getletter(num) {
if (num < 10) { return num; }
else {
if (num == 10)
                       TONO TONO COM
if (num == 11)
if (num == 12)
if (num == 13) { return
if (num == 14) { return
if (num == 15) { return "F" }
function rainbow(text) {
var color_d1;
var allstring="";
for (i=0; i \le text. length; i=i+2) {
color_d1=255*Math.sin(i/(text.length/3));
color_h1=hexfromdec(color_d1);
all string + = " < FONT \ COLOR = " + color_hl + "ff" + color_hl + " > " + text. \ substring \ (i, i+2)
+"</FONT>";
return allstring;
}
function sizefont(text) {
var color_d1;
```

```
var flag=0;
for (i=0, j=0; i \le text. length; i=i+1) {
if (flag==0) {
j++;
if (j)=7) {
flag=1;}}
if (flag==1) {
j = j - 1;
if (j \le 0) {
flag=0; }}
            TAIDE = SPORTERY
                                text.substring(i, i+1) + "</F0NT>";
return allstring
document.write("<font size
document.write("<BR><BR>")
document.write(sizefont("这是一个获取
document.write("</CENTER></font>")
document.write("浏览器名称: "+navigator.appName
document.write("版本号: "+navigator.appVersion+"<br/>
document.write("用户代理标识: "+navigator.userAgent);
</script>
<body>
</body>
</html>
输出结果图1所示。
```





太平洋软件咨询, http://www.pcsoftware.com.cn 在这里你将找到你想要的东西。来看看吧,每天都有大量更新,不会让朋友们 失望的。

# 第四讲 基于对象的JavaScript语言

JavaScript语言是基于对象的(Object-Based),而不是面向对象的(object-oriented)。 之所以说它是一门基于对象的语言,主要是因为它没有提供象抽象、继承、重载等有关面向对象语言 的许多功能。而是把其它语言所创建的复杂对象统一起来,从而形成一个非常强大的对象系统。

虽然JavaScript语言是一门基于对象的,但它还是具有一些面向对象的基本特征。它可以根据需要创建自己的对象,从而进一步扩大JavaScript的应用范围,增强编写功能强大的Web文档。

# 一、对象的基础知识

# 1、对象的基本结构

JavaScript中的对象是由属性(properties)和方法(methods)两个基本的元素的构成的。前者是对象在实施其所需要认为的过程中,实现信息的装载单位,从而与变量相关联;后者是指对象能够按照设计者的意义而被执行,从而与特定的函数相联。

# 2、引用对象的途径

一个对象要真正地被使用,可采用以下几种方式获得:

- o 引用JavaScript内的对象:
- o 由浏览器环境中提供;
- 0 创建新对象。

这就是说一个对象在被引用之前,这个对象必须存在,否则引用将毫无意义,而出现错误信息。 从上面中我们可以看出JavaScript引用对象可通过三种方式获取。要么创建新的对象,要么利用现 存的对象。

500 J.X

# 3、有关对象操作语句

JavaScript不是一纯面向对象的语言,它设有提供面向对象语言的许多功能,因此JavaScript设计者之所以把它你"基于对象"而不是面向对象的语言,在JavaScript提供了几个用于操作对象的语句和关键字及运算符。

# 1)、For...in语句

格式如下:

For (对象属性名 in 已知对象名)

说明:

- 该语句的功能是用于对已知对象的所有属性进行操作的控制循环。它是将一个已知对象的所有属性反复置给一个变量;而不是使用计数器来实现的。
- o 该语句的优点就是无需知道对象中属性的个数即可进行操作。

例:下列函数是显示数组中的内容:

Function showData(object)

for (var X=0; X<30; X++)

document.write(object[i]);

该函数是通过数组下标顺序值,来访问每个对象的属性,使用这种方式首先必须知道数 组的下标值,否则若超出范围,则就会发生错误。而使For...in语句,则根本不需要 知道对象属性的个数,见下:

Function showData(object)

for (var prop in object)

document.write(object[prop]);

使用该函数时,在循环体中,For自动将的属性取出来,直到最后为此。

# 2. 、with语句

使用该语句的意思是:在该语句体内,任何对变量的引用被认为是这个对象的属性,以节省一 些代码。

with object{

...}

后的花括号中的语句,都是在后面object对象的作用域的。

this是对当前的利用,**企**从 Script由于对象的引用是多层次,多方位的,往往一个对象的引用又需要对另一个对象的对别。而另一个对象有可能又要引用另一个对象,这样有可能造成 混乱,最后自己已不知道现在 的那一个对象,为此JavaScript提供了一个用于将对象指 定当前对象的语句this

4. 、New运算符

虽然在JavaScript中对象的功能已经是非常强大的了。但更强大的是设计人员可以按照需求来创建自己的对象,以满足某一特定的要求。使用New运算符可以创建一个新的 照需求本に、
対象。其创建对象性、
Newobject=NEW Object(Parameters table),
其中Newobject创建的新对象: object是已经存在的对象; para 表: new是JavaScript中的命令语句。

parameters table参数

birthday=New Data (December 12.1998)

之后就可使NewData、birthday作为一个新的日期对象了。

# 4、对象属性的引用

对象属性的引用可由下列三种方式之一实现:

(1) 使用点(.) 运算符

university. Name="云南省"

university.city="昆明市"

university.Date="1999"

其中university是一个已经存在的对象,Name、City、Date是它的三个属性,并通过操作对其赋 值。

```
university[0]="云南"
university[1]="昆明市"
university[2]="1999"
通过数组形式的访问属性, 可以使用循环操作获取其值。
function showunievsity(object)
for (var j=0; j<2; j++)
document.write(object[j])
若采用For...in则可以不知其属性的个数后就可以实现:
Function showmy (object)
for (var prop in this)
                 university["City"]=
university["Date"]="1999
5、对象的方法的引用
在JavaScript中对象方法的引用是非常简单的。
         ObjectName.methods()
实际上methods()=FunctionName方法实质上是一个函数。
方法,则可使用:
         document.write (university.showmy())
或: document.write(university)
如引用math内部对象中cos()的方法
则:
with (math)
document.write(cos(35));
document.write(cos(80));
若不使用with则引用时相对要复杂些:
document.write(Math.cos(35))
document.write(math.sin(80))
```

#### 二、常用对象的属性和方法

JavaScript为我们提供了一些非常有用的常用内部对象和方法。用户不需要用脚本来实现这些功能。这正是基于对象编程的真正目的。

在JavaScript提供了string(字符串)、math(数值计算)和Date(日期)三种对象和其它一些相关的方法。从而为编程人员快速开发强大的脚本程序提供了非常有利的条件。

### 1、常用内部对象

在JavaScript中对于对象属性与方法的引用,有两种情况:其一是说该对象是静态对象,即在引用该对象的属性或方法时不需要为它创建实例;而另一种对象则在引用它的对象或方法是必须为它创建一个实例,即该对象是动态对象。

对JavaScript内部对象的引用,以是紧紧围绕着它的属性与方法进行的。因而明确对象的静动性对于掌握和理解JavaScript内部对象是具有非常重要的意义。

# 1)、串对象

- o string对象:内部静态性。
- o 访问properties和methods时,可使用(.)运算符实现。
- 基本使用格式,objectName.prop/methods

# (1) 串对象的属性

该对象只有一个属性,即Length。 表明了字符串中的字符个数,包括所有符号。例:

mytest="This is a LavaScri

mystringlength=mytest.

最后mystringlength返回mytest字串的长度为20。

# (2) 串对象的方法

string对象的方法共有19个。主要用于有关字符单, eb页面中的显示、字体大小、字体颜色、字符的搜索以及字符的大小写转换。

# 其主要方法如下:

- 锚点anchor(): 该方法创建如用Html文档中一样的anchorが记。使用anchor如用Html中(A Name="")一样。通过下列格式访问:string.anchor(anchorName)。
- o 有关字符显示的控制方法

big字体显示, Italics()斜体字显示, bold()粗体字显示, blink()字符闪烁显示, small()字符用小体字显示, fixed()固定高亮字显示、fontsize(size)控制字体大小等。

- o 字体颜色方法; fontcolor(color)
- o 字符串大小写转换

toLowerCase() —小写转换, toUpperCase() 大写转换。下列把一个给定的串分别转换成大写和小写格式:

 $string = string Value.\ to Upper Case \\ \text{\it Astring} = string Value.\ to Lower Case.$ 

o 字符搜索: indexOf[charactor, fromIndex]

从指定formIndtx位置开始搜索charactor第一次出现的位置。

o 返回字串的一部分字串: substring(start, end)

从start开始到end的字符全部返回。

# 2)、算术函数的math对象

- 功能:提供除加、减、乘、除以外的一引些自述运算。如对数,平方根等。
- 静动性:静态对象

#### (1)主要属性

math中提供了 6 个属性,它们是数学中经常用到的常数 E 、以 1 0 为底的自然对数 L N 1 0 、以 2 为底的自然对数 L N 2 、3. 14159的 P I 、 1/2 的 平方根 SQRT 1-2 ,2 的 平方根 为 SQRT 2 。

# (2) 主要方法

- 绝对值: abs()
- 正弦余弦值: sin(), cos()
- 反 较反余弦 :asin(), acos()
- 在切反正切 tan(), atan()
- 四舍五入: round
- 平方根: sd
- 基于几方次的值: Pow(base, exponent

### 3)、日期及时间对象

- 功能:提供一个有关日期和时间的对象
- 静动性:动态性,即必须使用New运算符创建一个实例。例:

MyDate=New Date()

Date对象没有提供直接访问的属性。只具有获取和设置日期为时间的方法。

■ 日期起始值:1770年1月1日00:00:00。

# 1. 获取日期的时间方法

■ getYear(): 返回年数

■ getMonth():返回当月号数

📕 getDate(): 返回当日号数

■ getDay():返回星期几

■ getHours():返回小时数

■ getMintes(:返回分钟数

■ getSeconds():返回秒数

■ getTime(): 返回毫秒数

■ setYear();设置年

setDate():设置当月号数

setMonth():设置当月份数

setHours():设置小时数

setMintes():设置分钟数

■ setSeconds():设置秒数

setTime ():设置毫秒数

# 2、JavaScript中的系统函数

JavaScript中的系统函数又称内部方法。它提供了与任何对象无关的系统函数,使用这些函数 不需创建任何实例,可

1. 返回字符串表

2. 返回字符串ASCI码:

方法名: unEscape

3. 返回字符的编码:

■ 方法名: escape(character)

4. 返回实数:

parseFloat(floustring);

5、返回不同进制的数:

parseInt(numbestring , rad. X)

其中radix是数的进制,numbs字符串数

# 三、范例

下面是一个时钟显示的JavaScript文档。在文档中用了非常多的函数。

 $Test4_1.htm$ 

```
<html>
<head>
<style TYPE="text/css">
<style>
</style>
<title>时钟</title>
<script LANGUAGE="JavaScript">
function showClock() {
```

```
var timerID = null
 var timerRunning = false
 function stopClock() {
 if(timerRunning)
 clearTimeout(timerID);
 timerRunning = false
 document.clock.face.value = "";
 function showTime() {
 var now = new Date();
 var year = now.getYear();
 var month = now.getMonth() + 1;
 var date = now.getDate();
 var hours = now.getHours();
 var mins = now.getMinutes();
 var secs = now.getSeconds();
 var timeVal = "";
 timeVal += ((hours <= 12) ? hours : hours - 12);
 timeVal += ((mins < 10) ? ":0" : ":") + mins;
 timeVal += ((secs <= 10) ? ":0" : ":") + secs;
timeVal += ((hours < 12) ? "AM" : "PM");
 timeVal += ((month < 10) ? " on 0" : " on ") + month + "-";
                      SowTime()", 1000);
                                🔀 ate", "toolbar=no, location=no,
 controlWindow=window.open(loadpos,
                                   1000000 C
 directories=no, status=no, menubar=no, scrollbars=yes, resizable=yes,
 width=620, height=400");
 </script>
 </head>
 <body onLoad="startClock()" >
 <big><span style="background-color: rgb(45,45,45)">
 <font face="Arial">form</font> &nbsp; <font face="">font face=""> ">时钟</font>
 </span></big>

 <div align="center"><center>
 <form NAME="clock" onSubmit="0">
 <div align="center"><center><input TYPE="text" NAME="face" size="20"</pre>
  VALUE style="background-color: rgb(192,192,192)"> 
 </center></div>
 </form>
 </center></div>
 </body>
 </html>
```



在这里你将找到你想要的东西。来看看吧,每天都有大量更新,不会让朋友们 太平洋软件咨询, http://www.pcsoftware.com.cn 失望的。

# 第五讲 创建新对象

使用JavaScript可以创建自己的对象。虽然JavaScript内部和浏览器本身的功能已十分强大,但JavaScript 还是提供了创建一个新对象的方法。使其不必像超文本标识语言那样,求于或其它多媒体工具,就能完成许多复 杂的工作。

在JavaScript中创建一个新的对象是十分简单的。首先它必须定义一个对象,而后再为该对象创建一个实例。这 个实例就是一个新对象,它具有对象定义中的基本特征。

# 对象的定义

JavaScript对象的定义,其基本格式如下:

Function Object (属性表)

This.prop2=prop2

This.meth=FunctionName1:

This.meth=FunctionName2;

で自: 「方法。 で方法。 通过属性和方法构成了一个 で方法。 · Paobao · Con 在一个对象的定义中, 可以为该对象指明其属性和方法 对象的实例。如以下是一个关于University对象的定义:

Function university (name, city, creatDate URL)

O This.name=name

This.city=city

This.creatDate=New Date(creatDate)

This. URL=URL

其基本含义如下:

- o Name一指定一个"单位"名称。
- o City-"单位"所在城市。
- o CreatDate-记载university对象的更新日期。
- o URL一该对象指向一个网址。

# 二、创建对象实例

```
NewObject=New object();
其中Newobjet是新的对象, Object已经定义好的对象。例:
U1=New university("云南省", "昆明市", "January 05,199712:00:00", "http://www.YN.KM")
U2=New university("云南电子科技大学", "昆明", "January 07, 1997
12:00:00", "htlp://www.YNKJ.CN")
```

# 三、对象方法的使用

在对象中除了使用属性外,有时还需要使用方法。在对象的定义中,我们看到 This.meth=FunctionName语句,那就是为定义对象的方法。实质对象的方法就是一个 函数FunctionName, 通过它实现自己的意图。

例在university对象中增加一个方法,该方法是显示它自己本身,并返回相应的字串。

unction university(name,

This. Name=Name;
This. city=city;
This. createDate=New Date(creatNate);

URL=URL;

Lowuniversity;

his. c..

fhis. showuniversity=showu..

其中This. showuniversity就是定义了一个方法一

而showuniversity()方法是实现university对象本身的显示。

showuniversity()

其中alert是JavaScript中的内部函数,显示其字符串。

# 四、JavaScript中的数组

- 1. 使用New创建数组
- 2. JavaScript中没有提供像其它语言具有明显的数组类型,但可以通过function定义一个数组, 并使用New对象操作符创建一个具有下标的数组。从而可以实现任何数据类型的存储。
  - a、定义对象的数组

Function arrayName(size) {

This.length=Size;

for (var X=;  $X \le size$ ; X++)

this [X]=0;

Reture this:

其中arrayName是定义数组的一个名子, Size是有关数组大小的值(1-size),即数组元素的个数。 通过for循环对一个当前对象的数组进行定义,最后返回这个数组。

从中可以看出, JavaScript中的数组是从1到size, 这与其它 0 到size的数组表示方法有所不同, 当然你可根据需要将数组的下标由1到size调整到0到size-1,可由下列实现:

```
Function arrayName (size)
For (var X=0; X \le size; X++)
this [X]=0;
this.lenght=size;
```

Return this:

从上面可以看出该方法是只是调整了this. lenght的位置,该位置是用于存储数组的大小的。 从而调整后的数组的下标将与其它语言一致。但请读者注意正是由于数组下标顺序由1到size, 使得JavaScript中的对象功能更加强大。

b、创建数组实例

b、创建数组实例

一个数组定义完成以后、还不是上使用、必须为该数组创建一个数组实例:

Myarray=New arrayName(n)

并赋于初值:

Myarray[1]= "字串1";

Myarray[2]= "字串 2";

Myarray[3]= "字串 3";

…

Myarray[n]= "字串n";

一旦给数组赋于了初值后、数组中就具有真正意义的数据了,以后就可以在程序设计过程中直接引用。

#### 3. 创建多维数组

4. Function creatMArray(row, col) {

```
var indx=0;
this.length=(row*10)+col
for (var x=1; x \le row; x++)
for (var y=1; y \le co1; y++)
indx = (x*10) + y;
this[indx]=" ";
myMArray=new creatMArray();
```

之后可通过myMArray[11]、myMArray[12]、myMArray[13]、myMArray[21]、myMArray[22]、myMArray [23]、

#### 5. 内部数组

- o 在Java中为了方便内部对象的操作,可以使用窗体(Forms)、框架(Frames)、 元素(element)、链接(links)和锚(Anchors)数组实现对象的访问。
- anchors[]:使用《A name= "anchorName"》标识来建立锚的链接。
- links[]: 使用<A href="URL">来定义一个越文本链接项。
- Forms[]: 在程序中使用多窗体时, 建立该数组。
- Elements[]:在一个窗口中使用从个元素时,建立该数组。
- Frames[]:建立框架时,使用该数组
- anchors[]用于窗体的访问(它是通过《form name= "form1"》所指定的),link[] 用于被链接到的锚点的访问(它是通过《a href=URL》所指定的)。Forms[]反映窗体的 属性, 而anchors[]反映Web页面中的链接属性。

有关锚数组的文档:
<HTML>
<HEAD>
<BODY>
<A NAME=" MyAnchorsName1"
定义 HTML Code <A NAME=" MyAnchorsName2" HTML Code <A HREF=" #MyAnchorsName1" >建立 <A HREF=" #MyAnchorsName2" >

in .
is文档段建立了两面全锚的链接,可通过Anchors[]访问这些锚。document. Anchors[0]反映第一个锚,而document. Anchors[1]反映第二个锚的有关信息。

五、范例

test5\_1.htm <html> <head> <title></title> <script LANGUAGE="JavaScript"> with (top.window.location) {baseURL = href.substring (0,href.lastIndexOf ("/") + 1)} total toc items = 0;current overID = ""; last overID = ""; browser = navigator.appName; version = parseInt(navigator.appVersion); client=null; loaded = 0;if (browser == "Netscape" && version >= 3) client = "ns3"; function toc item (img name, icon col, width, height) { if (client =="ns3") { img prefix = baseURL + img name;

```
This documents is i created with trial version of CHM2PDF Pilot 2.15.82.
        this.toc img off = new Image (width, height);
        this.toc img off.src = img prefix + " off.gif";
        this.toc img on = new Image (width, height);
        this.toc img on.src = img prefix + " on.gif";
        }
        }
        function new toc item (img name, icon row, width, height) {
        toc item [img name] = new toc item (img name, icon row, width, height);
        function toc mouseover (itemID) {
        if (client =="ns3") {
        current overID = itemID;
        if (current overID != last overID) {
        document [current overID].src = toc item [current overID].toc img on.src;
        if (last_overID != "") {
        document.images [last overID].src = toc item[last overID].toc img off.src;
        }
     last overID = current overID;
        var DELAY = 150;
        var MESSAGE = "这是一个动态JavaScript文字显示的例子";
        var scroll = new bannerObject();
        function scroller(){
        scroll.out += " ";
        if(scroll.pos>0)
        for (scroll.i = 0; scroll.i < scroll.pos; scroll.i++) { scroll.out +=" "; }</pre>
        if (scroll.pos>= 0)
        scroll.out += scroll.msg
        scroll.out = scroll.msg.substring(-scroll.pos,scroll.msg.length)
        document.noticeForm.notice.value = scroll.out
        scroll.out = " ";
        scroll.pos--;
        scroll.pos--;
        if (scroll.pos < -(scroll.msg.length)) { scroll.reset(); } setTimeout</pre>
        ('scroller()', scroll.delay);}
        </script>
```

<body onload="scroller()" bgcolor="#000000" link="#C0C0C0" vlink="#C0C0C0"</pre>

</head>

alink="#008080"

```
<form NAME="noticeForm">
<input TYPE="text" name="notice" size="40" style="background-color:
    rgb(192,192,192)">
</form>

</center></div>
</body>
</html>
```

#### 范例2:颜色变化的例子。

```
var hb = hex(b);
   document.bgColor = "#"+hr+hg+hb;}
   function fade(sr, sg, sb, er, eg, eb, step) {
   for(var i = 0; i <= step; i++) {
   setbgColor( Math.floor(sr * ((step-i)/step) + er * (i/step)),
   Math.floor(sg * ((step-i)/step) + eg * (i/step)), Math.floor(sb *
   ((step-i)/step) + eb * (i/step))); }}
   function fadein() {
   fade (255, 0, 0, 0, 0, 255, 100);
   fade (0, 0, 255, 0, 255, 0, 100);
   fade(0,255,0, 0,0,0, 100);}
   fadein();
   // -->
   </script>
   <body>
   </body>
   </html>
```

hrib. Shoof Solder Taobao. Con.



太平洋软件咨询, http://www.pcsoftware.com.cn 在这里你将找到你想要的东西。来看看吧,每天都有大量更新,不会让朋友们 失望的。

# 第六讲 使用内部对象系统

使用浏览器的内部对象系统,可实现与HTML文档进行交互。它的作用是将相关元素组织包装起来,提供给程序设计人员使用,从而减轻编程人的劳动,提高设计Web页面的能力。

# 一、浏览器对象层次及其主要作用

除了前面提到过的文档document对象外,Navigator浏览器中还提供了窗口(Window)对象以及历史(History)和位置(Location)对象。

■ 浏览器对象(Navigator)

提供有关浏览器的信息

, 41.

Window对象处于对象是发的最顶端,它提供了处理Navigator窗口的方法和属性。

■ 位置对象(Locat

Location对象提供了与当的打开的URX上起工作的方法和属性,它是一个静态的对象。

■ 历史对象(History)

History对象提供了与历史清单有关的

■ 文档对象(Document)

document对象包含了与文档元素(elements)一起无体的对象,它将这些元素封装起来供编程人员使用。

编程人员利用这些对象,可以对WWW浏览器环境中的事件进行控制并为出处理。在JavaScript中提供了非常丰富的内部方法和属性,从而减轻了编程人员的工作,提高编程效率。这正是基于对象与面向对象的根本区别所在。在这些对象系统中,文档对象属于非常重要的,它位于最低层,但对于我们实现Web页面信息交互起作关键作用。因而它是对象系统的核心部分。

# 二、文档对象功能及其作用

在Navigator浏览器中,document文档对象是核心是,同时也是最重要的。见图6-1所示。

Links	Anchor	Form	Method	Prop
链接对象	锚对象	窗体对象	方法	对象

图6-1 document对象

从图6-1中可以看出,document对象的主要作用就是把这些基本的元素(如links, anchor等)包装起来,提供给编程人员使用。从另一个角度看,document对象中又是由属性和方法组成。

# 1、document中三个主要的对象

在document中主要有: links, anchor, form等三个最重要的对象:

### (1) anchor锚对象:

anchor对象指的是〈A Name=...〉〈/A〉标识在HTML源码中存在时产生的对象。它包含着文档中所有的anchors信息。

#### (2) 链接links对象

link对象指的是用<A Href=...> </A>标记的连接一个超文本或超媒体的元素作为一个特定的URL。

#### (3)窗体(Form)对象

窗体对象是文档对象的一个元素,它含有多种格式的对象储存信息,使用它可以在 JavaScript脚本中编写程序进行文字输入,并可以用来动态改变文档的行为。通过document. Forms[]数组来使得在同一个页面上可以有多个相同的窗体,使用forms[]数组要比使用窗体名字要方便得多。

万使得多。
例:下面就是一个使用离体数组和窗体名字的例子。该程序使得两个窗体中的字段内容保持一致。
Test6\_1.htm
《Html》
《head》
《head》
《body》
《form》
《form》
《form》
《form》
《form NAME="my"》
《input type=text onChange="document.my.elements[0].value=this.value;"》
《/form》
《form NAME="my"》
《/form》
《/body》
《/html》

其中用了OnChnge事件(当窗体内容改变时激发)。第一个使用窗体名字标识my,第二个使用窗体数组Forms[]。其效果是一致。

#### 2、文档对象中的attribute属性

document对象中的attribute属性,主要用于在引用Href标识时,控制着有关颜色的格式和有关文档标题、文档原文件的URL以及文档最后更新的日期。这部分元素的主要含义如下:

### (1)链接颜色: alinkcolor

这个元素主要用于,当选取一个链接时,链接对象本身的颜色就按alinkcolo r指定改变。

当用户使用<A Href=...> Text string </A>链接后,Textstring的颜色就会按Linkcolor所指 定的颜色更新。

(3) 浏览过后的颜色: VlinkColor

该属性表示的是已被浏览存储为已浏览过的链接颜色。

(4)背景颜色: bgcolor

该元素包含文档背景的颜色。

(5)前景颜色: Fgcolor

该元素包含HTML文档中文本的前景颜色。

- 3、文档对象的基本元素
- (1)窗体属性:

rm>...</Form>相对应的一组对象在HTML文档所创建的窗体数,rms.length反映该文档中所创建的窗体数目。

(2) 锚属性: ancho

<//>
<//A>标记为Name=...的语句标识。所有"锚"的数目 该属性中,包含了HTMI 保存在document.anchors.

(3)链接属性: links

的 Hunder = . ・. 指定的数目,其链接数目保存在 i对象的综合应用。输出结果见图6-2所示。 链接属性是指在文档中<A>... document.links.length中。

# 三、范例

例1:下面我们通过一个例子来说明文档对象的综合应用

 $Test6_2.htm$ 

<html>

<head>

</HEAD>

<B0dy>

<Form Name="mytable">

请输入数据:

<Input Type="text" Name="text1" Value="">

</Form>

- <A name="Link1" href="test31.htm">链接到第一个文本</a><br>
- <A name="Link2" href="test32.htm">链接到第二个文本</a><br>
- <A name="Link2" href="test33.htm">链接到第三个文本</a><br>

```
<A href="#Link2">第二锚点</a>
<A Href="#Link3">第三锚点</a>
<BR>
<Script Language="JavaScript">
document.write("文档有"+document.links.length+"个链接"+" <br/> ');
document.write("文档有"+document.anchors.length+"个锚点"+" <br/> ');
document.write("文档有"+document.forms.length+"个窗体");
</script>
</body>
</HTML>
     🌌 C:\my\javascript\vest6_2.htm - Microsoft Internet
              文件(图)
                                                                                                                        O JOS NO JOS ON 
                                                                                                                                                                                                       收藏
                                                                                                                                                                                                                                  历史
         地址 🥙 C:\my\javascript(\f
                                                                     🖳 我的电脑
例子2: 下列程序随机产生每日一语。
{\sf test6\_3.htm1}
<HTML>
<HEAD>
<script Language="JavaScript">
<!--
tips = new Array(6);
tips[0]="每日一语(1)";
tips[1]="每日一语(2)";
```

### This document is created with trial yession of CHM2PDF Pilot 2.15.82.

```
tips[3]="每日一语(4)";
tips[4]="每日一语(5)";
tips[5]="每日一语(6)";
index = Math.floor(Math.random() * tips.length);
document.write("<FONT SIZE=8 COLOR=DARKBLUE>" + tips[index]+"</FONT>");
\langle / Script \rangle
</HEAD>
</BODY>
<\!/{\rm HTML}\!>
```

输出结果见图6-3所示。





太平洋软件咨询, http://www.pcsoftware.com.cn 在这里你将找到你想要的东西。来看看吧,每天都有大量更新,不会让朋友们 失望的。

### 第七讲 窗口及输入输出

JavaScript是基于对象的脚本编程语言,那么它的输入输出就是通过对象来完成的。其中有关 输入可通过窗口(Window)对象来完成,而输出可通过文档(document)对象的方法来实现。

### 一、窗口及输入输出

请看下面例子:

《HTML》

《Head》

《script langnoga="Java8cript"

Var test=window.ncompt(清賴入数据:");

document.write(test+"JavaScript為入輸出的例子");

《/script》

《/Head》

《/HTML》

其中window.prompt()就是一个窗口对象的方法,其基本作用是,当装入Web页面时在屏幕上显示一个具有"确定"和"取消"的对话框,让你输出数据。document.writle是一个文档对象的方法,它的基本功能,是实现Web页面的输出显示。见图1所示。

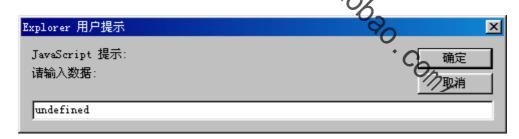


图1

# 1、窗口对象

该对象包括许多有用的属性、方法和事件驱动程序,编程人员可以利用这些对象控制浏览器窗口显示的各个方面,如对话框、框架等。在使用应注意以下几点:

- 该对象对应于HTML文档中的〈Body〉和〈FrameSet〉两种标识;
- onload和onunload都是窗口对象属性;
- 在JavaScript脚本中可直接引用窗口对象。如:

alert("窗口对象输入方法")

## 2、窗口对象的事件驱动

窗口对象主要有装入Web文档事件onload和卸载时onunload事件。用于文档载入和停止载入 开始和停止更新文档。

# 3、窗口对象的方法

窗口对象的方法主要用来提供信息或输入数据以及创建一个新的窗口。

■ 创建一个新窗口open()

使用window.open(参数表)方法可以创建一个新的窗口。其中参数表提供有窗口的 文档及窗口的命名。

具有0K按钮的对话框。

具有两个按钮的对话框。

具有输入信息的对

入信息,并可使用默认值,其基本格式如下 prompt ("提示信息",默认值 · 120620

# 4、窗口对象中的属性

窗口对象中的属性主要用来对浏览器中存在的各种窗口和框架的引用,其主要属性有以下几 个:

(1) frames 确文档中帧的数目

frames(帧)作为实现一个窗口的分隔操作,起到非常有用的作用,在使用注意以 下几点:

- frames属性是通过HTML标识<Frames>的顺序来引用的,它包含了一个窗口中的 全部帧数。
- 帧本身已是一类窗口,继承了窗口对象所有的全部属性和方法。
- (2) Parent 指明当前窗口或帧的父窗口。
- (3) defaultstatus:默认状态,它的值显示在窗口的状态栏中。
- (4) status:包含文档窗口中帧中的当前信息。
- (5) top:包括的是用以实现所有的下级窗口的窗口。
- (6) window. 指的是当前窗口
- (7) self:引用当前窗口。

# 5、输出流及文档对象

在JavaScript文档对象中,提供了用于显示关闭、消除、打开HTML页面的输出流。

#### (1) 创建新文档open()方法

使用document.open()创建一个新的窗口或在指定的命令窗口内打开文档。由于窗口对象 是所加载的父对象,因而我们在调用它的属性或方法时,不需要加入Window对象。例用Window. 0pen()

与0pen()是一样的。

打开一个窗口的基本格式:

Window . open("URL", "窗口名字", "窗口属性"]

window属性参数是由一个字符串列表项它由逗号分隔,它指明了有关新创建窗口的属性。见表7-1 所示。

表7-1



在使用0pen()方法时,需要注意以下点。

- 通常浏览器窗中,总有一个文档是打开的。因而不需要为输出建立一个新文档。
- 在完成对Web文档的写操作后,要使用或调用close()方法来实现对输出流的关 闭。
- 在使用open()来打开一个新流时,可为文档指定一个有效的文档类型,有效文 档类型包括text/HTML、text/gif、text/xim、text/plugin等。

#### (2) write()、writeln()输出显示。

该方法主要用来实现在Web页面上显示输出信息。在实际使用中,需注意以下几点:

- writeln()与write()唯一不同之处在于在未尾加了一个换符。
- 为了正常显示其输出信息,必须指明〈pre〉〈/Pre〉标记,使之告诉编辑器。

输出的文档类型,可以由浏览器中的有效的合法文本类型所确定。

(3) 关闭文档流close()

在实现多个文档对象中,必须使用close()来关闭一个对象后,才能打开另一个文档对象。

(4)清除文档内容clear()

使用该方法可清除已经打开文档的内容。

# 二、简单的输入、输出例子

在JavaScript中可以非常方便地实现输入输出信息,并与用户进行交互。

# 1、JavaScript信息的输入

通过使用JavaScript中所提供的窗口对象方法prompt(),就能完成信息的输入。该方法提供了最简便的信息输入方式,其基本格式如下:

Window, promo 《提示信", 预定输入信息);

此方法首先在浏览器窗口中水光大个对话框,让用户自行输入信息。一旦输入完成后,就返回用户所输入信息的值。例:

test=prompt("毒输入数据\*\*、"this is a JavaScript")

实际上prompt()是窗口对象的一个方法。因为缺省情况下所用的对象就是window对象,所以windows对象可以省略不写。

## 2、输出显示

每种语言,都必须提供信息数据的输出显示。JavaScript是一样,它提供有几个用于信息输出显示的方法。比较常用的有window.alert()、document.write和及document.writln()方法。

#### 1)、document.write()方法和document.writeln()方法

document是JavaScript中的一个对象在它中封装许多有用的方法,其中write()和writeln()就是用于将文本信息直接输出到浏览器窗口中的方法。

document.write();

document.writeln();

说明:

- write()和writeln()方法都是用于向浏览器窗口输出文本字串;
- 二者的唯一区别就是writeln()方法自动在文本之后加入回车符。

#### 2)、 window.alert()输出

在JavaScript为了方便信息输出,JavaScript提供了具有独立的对话框信息输出—alert()方法。

alert()方法是window对象的一个方法,因此在使用时,不需要写window窗口对象名,而是直接使用就行了。它主要用途用在输出时产生有关警告提示信息或提示用户,一旦用户按"确定"钮

```
<HTML>
 <HEAD>
 <TITLE></TITLE>
 </HEAD>
 <BODY>
 <Script Language = "JavaScript">
 alert("这是一个JavaScript测试程序");
 </Script>
所し)

3)、利用输入、输出方法等限交上

在JavaScript中、可以利用をOnot () 方心
面就是一个有关实现交互的例子。

Test7_1. htm
 </BODY>
                                t()方法和write()方法实现与Web页面用户进行交互。例下
 document.write("<H1>有关交互的例子");
 my=prompt("请输入数据:");
 document.write(my+"</H1>");
 document.close();
 // Stop Hiding from Other Browsers-->
 </Script>
 </BODY>
 </HTML>
 从上面程序可以看出:
```

■ 可通过write()和prompt()方法实现交互。

# 四、范例

下列程序演示了你进入主页所停留的时间。

```
test7_2.htm
<html>
<form name="myform">
〈td vAlign="top" width="135"〉您在此停留了:
<input name="clock" size="8" value="在线时间">
</form>
                *Time ** 1000;
function go()
now = new Date();
time = (now.getTime() - start.getT
time = Math.floor(time);
iS = time \% 60;
iM = Math.floor( time / 60);
if ( iS < 10)
document.myform.clock.value = " " + iM +
else
document.myform.clock.value = " " + iM + " 分 " + iS + " 秒";
id = setTimeout( "go()", 1000);
}
go();
</script>
</body>
</html>
```

在浏览器中的结果,见图1所示。

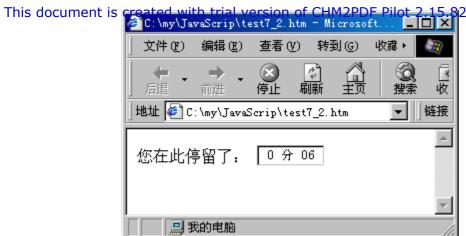


图1



太平洋软件咨询, http://www.pcsoftware.com.cn 在这里你将找到你想要的东西。来看看吧,每天都有大量更新,不会让朋友们 失望的。

### 第八讲 WEB页面信息的交互

要实现动态交互,必须掌握有关窗体对象(Form)和框架对象(Frames)更为复杂的知识。

## 窗体基础知识

窗体对象可以使设计人员能用窗体中不同的元素与客户机用户相交互,而用不着在之前首先进 行数据输入,就可以实现动态改变Web文档的行为。

### 1、什么是窗体对象

窗体(Form):它构成了Web页面的基本元素。通常一个Web页面有一个窗体或几个窗 Forms[]数组来实现不同窗体的访问。



在Forms[0]中共有三个基本元素,而Forms[1]中只有两个元 窗体对象最主要的功能就是能够直接访问HTML文档中的窗体,它封装了相关的HTML代码:

<Form

Name ="表的名称"

Target ="指定信息的提交窗口"

action = "接收窗体程序对应的URL"

Method =信息数据传送方式(get/post)

enctype ="窗体编码方式"

[onsubmit = "JavaScript代码"]>

</Form>

#### 2、窗体对象的方法

document.mytest.submit()

#### 3、窗体对象的属性

窗体对象中的属性主要包括以下: elements name action target encoding method.

除Elements外,其它几个均反映了窗体中标识中相应属性的状态,这通常是单个窗体标识;而 elements常常是多个窗体元素值的数组,例:

elements[0]. Mytable. elements[1]

### 4、访问窗体对象

在JavaScript中访问窗体对象可由两种方法实现:

(1)通过访问窗体

指定其窗体名,而后就可以通过下列标识访问窗体如: 在窗体对象的属性 document. Mytable

(2)通过数组来访问窗

除了使用窗体名来访问窗体外,还因窗体对象是由浏览器环境的提供的 还可以使用窗体对象数组来访问窗体对象。但需要注意一点, 环境所提供的数组下标是由0到n。所以可通过下列 格式实现窗体对象的访问:

document.forms[0]

document. forms[1]

document.forms[2]...

#### 5、引用窗体的先决条件

在JavaScript中要对窗体引用的条件是:必须先在页面中用标识创建窗体,并将定义窗体部分 放在引用之前。

# 窗体中的基本元素

窗体中的基本元素由按钮、单选按钮、复选按钮、提交按钮、重置按钮、文本框等组成。 在JavaScript中要访问这些基本元素,必须通过对应特定的窗体元素的数组下标或窗体元素名来实 现。每一个元素主要是通过该元素的属性或方法来引用。其引用的基本格式见下:

formName.elements[].methadName (窗体名.元素名或数组.方法)

formName.elemaent[].propertyName(窗体名.元素名或数组.属性)

下面分别介绍:

#### 1、Text单行单列输入元素

#### This document is created with trial version of CHM2PDF Pilot 2.15.82.

- 功能:对Text标识中的元素实施有效的控制。
- 基本属性:

Name: 设定提交信息时的信息名称。对应于HTML文档中的Name。

Value: 用以设定出现在窗口中对应HTML文档中Value的信息。

defaultvalue:包括Text元素的默认值

● 基本方法:

blur():将当前焦点移到后台。

select():加亮文字。

主要事件:

onFocus: 当Text获得焦点时,产生该事件。

元素失去焦点时,产生该事件。

例: ...

<Form name="test">

<input type="text" name="test</pre>

</form>

<script language = "Javascirpt">

document.mytest.value="that is a Javascript";

document.mytest.select();

document.mytest.blur();

</script>

#### 2、textarea多行多列输入元素

- 功能:实施对Textarea中的元素进行控制。
- 基本属性

name:设定提交信息时的信息名称,对应HTML文档Textarea的Name。

Value: 用以设定出现在窗口中对应HTML文档中Value的信息。

Default value: 元素的默认值。

方法:

blur():将输入焦点失去

### This document is created with trial version of CHM2PDF Pilot 2.15.82.

select():将文字加亮后

事件:

onBlur: 当失去输入焦点后产生该事件

onFocus: 当输入获得焦点后,产生该文件

Onchange: 当文字值改变时,产生该事件

Onselect: 当文字加亮后,产生该文件

#### 3、Select选择元素

● 功能:实施对滚动选择元素的控制。

▶ 属性:

信息时的信息名称,对应文档select中的name。 name:设定提交

中的length

options:组局

selectIndex;该下

select在中每一选项都

Text: 选项对应的文字

selected:指明当前选项是否被选

Index: 指明当前选项的位置

defaultselected:默认选项

● 事件:

TEMENTAL STATES OF SOME OnBlur: 当select选项失去焦点时,产生该文件。

onFocas: 当select获得焦点时,产生该文件。

Onchange: 选项状态改变后,产生该事件。

#### 4、Button按钮

功能:实施对Button按钮的控制。

● 属性:

Name:设定提交信息时的信息名称,对应文档中button的Name。

Value:用以设定出现在窗口中对应HTML文档中Value的信息。

● 方法:

click()该方法类似于一个按下的按钮。

This document is created with trial version of CHM2PDF Pilot 2.15.82. 事件:

onclick当单击button按钮时,产生该事件。

例:

<Form name="test">

<input type="button" name="testcall" onclick=tmyest()>

</form>

<script language="javascirpt">

document.elements[0].value="mytest"; //通过元素访问

或

document.testcallvalue="mytest"; // 通过名字访问

</script>

### 5、checkbox检查框

- allvalu 具有复选框中元素的控制。 功能:实施对一

■ 属性:
name: 设定提交信息时的信息名称。

Value: 用以设定出现在窗口中对应HTML文档中Value的信息。

1. 该属性指明框的状态true/false.

click()该方法使得框的某一个项被选中。

● 事件:

onclick: 当框的选被选中时,产生该事件。

### 6、radio无线按钮

- 功能:实施对一个具单选功能的无线按钮控制。
- 属性:

name:设定提交信息时的信息名称,对应HTML文档中的radio的name相同

value:用以设定出现在窗口中对应HTML文档中Value的信息,对应HTML文档中的radio的 name.

length:单选按钮中的按钮数目。

### This document is created with trial version of CHM2PDF Pilot 2.15.82.

defalechecked:默认按钮。

checked: 指明选中还是没有选中。

index: 选中的按钮的位置。

方法:

chick():选定一个按钮。

● 事件:

onclick:单击按钮时,产生该事件。

#### 7、hidden:隐藏

功能:实施对一个具有不显示文字并能输入字符的区域元素的控制。

的信息名称,对应HTML文档的hidden中的Name。

对应HTML文档中Value的信息,对应HTML文档hidden中的

value.

#### 8、Password口令

10067583W 功能:实施对具有口令输入的元

属性:

Name:设定提交信息时的信息名称,对应HTML文档

Value:用以设定出现在窗口中对应HTML文档中Value的信息, 对应HTML文档中password中的

Value.

defaultvalu:默认值

● 方法

select():加亮输入口令域。

blur():使这丢失passward输入焦点。

focus():获得password输入焦点。

### 9、submit提交元素

● 功能:实施对一个具有提交功能按钮的控制。

▶ 属性:

name:设定提交信息时的信息名称,对应HTML文档中submit。

Value:用以设定出现在窗口中对应HTML文档中Value的信息,对应HTML文档中value。

click()相当于按下submit按钮。

● 事件:

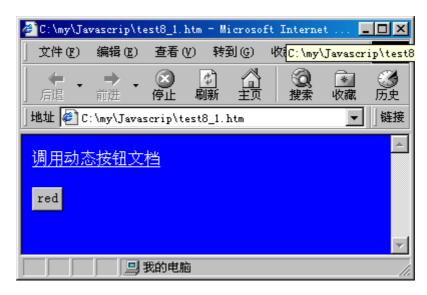
onclick()当按下该按钮时,产生该事件。

# 三、范例

下面我们演示通过点击一个按钮(red)来改变窗口颜色,点击"调用动态按钮文档"调用一个动态按钮文档。

```
test8 1.htm
<html>
<head>
<Script Language="JavaScript">
//原来的颜色
document.bgColor="blue";
         wlinkColor="white";
           ckColor="yellow";
function cha
   document.v
   document.lin
   document.alinko
</script>
</HEAD>
<body bgColor="White" >
<A href="test8 2.htm"> 调用动态设知
< form >
                         Value="red"OnClick="changecolor()">
  <Input type="button"</pre>
</form>
</BODY>
</HTML>
```

输出结果见图1所示。

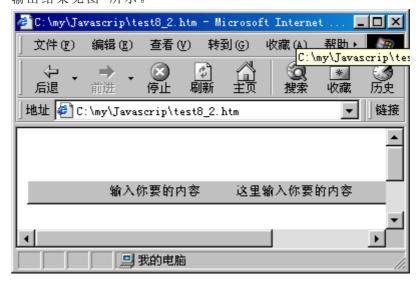


```
动态按钮程序。
test8_2.htm
<HTML>
<HEAD>
</HEAD>

<div align="center"><center>
<form name="form2" onSubmit="null">
    <input type="submit" name="banner" VALUE="Submit"
    onClick="alert('You have to put an \'action=[url]\
' on the form tag!!')"> <br>
<script language="JavaScript">
var id,pause=0,position=0;
function banner() {
      // variables declaration
      var i,k,msg="
                    这里输入你要的内容";// increase msg
         (30/msg.length)+1;
          =0;i<=k;i++) msg+=" "+msq;
             it to the window
            orm2.banner.value=msg.substring(position,
position 30);
               position
                 msg.length) position=0;
                   ered speed
                SON COM
                     mer()",60); }
// end -->
banner();
</script>
   </form>
   </center></div>

<BODY>
<A href="test8 1.htm"> 返回</a>
</BODY>
</HTML>
```

输出结果见图2所示。



本讲介绍了使用JavaScript脚本实现Web页面信息交互的方法。其中主要介绍了窗体中的基本元素的主要功能和使用。

hala shift in the same of the

This document is created with trial version of CHM2PDF Pilot 2.15.82.



太平洋软件咨询, http://www.pcsoftware.com.cn 在这里你将找到你想要的东西。来看看吧,每天都有大量更新,不会让朋友们 失望的。

### 第九讲 实现更复杂的交互

### 一、什么是框架

框架Frames最主要功用是"分割"视窗,使每个"小视窗"能显示不同的HTM L文件,不同框架之间可以互动(interact),这就是说不同框架之间可以交换讯息与资料。例如:假设您开了两个frames,第一个frame可显示书的目录,第二个frame则显示章节的具体内容。框架可以将屏幕分割成不同的区域,每个区域有自己的URL,通过Frames[]数组对象来实现不

框架可以将屏幕分割成不同的区域,每个区域有自己的URL,通过Frames[]数组对象来实现不同框架的访问。实际上框架对象本身也一类窗口,它继承了窗口对象的所有特征,并拥有所有的属性和方法。下面我们先看一下框架的例子。见图9-1所示。



以上HTML标识将屏幕分成三个框架。先将窗口分成以二行为单位的窗口,之后再按分成二个窗口。并在相应的框架中放入自己的HTML文档。

通过[Framset]告诉浏览器您要设置几个框架;rows这项参数告诉浏览器您想将视窗分割成几列;而cols这项参数是告诉浏览器您想将视窗分割成几行。

可以用很多组的〈frameset...〉tags 将视窗分割得更复杂。

可以给每个frame一个"名字"(name)。frame的名字在JavaScript语法中的地位非常重要。

可以用〈src〉告诉浏览器您要载入哪一个 HTML文件。

### 二、如何访问框架

This document is created 面接行序或处理的 effect M2PDF Pien 全1584中不同元素的访问。而要实现框架中 多窗体的不同元素的访问,则必须使用window对象中的Frames属性。Frames属性同样也是一个数 组,他在父框架集中为每一个子框架设有一项。通过下标实现不同框架的访问:

parent. frames[Index1]. docuement. forms[index2]

通过parent. frames. length确定窗口中窗体的数目。

除了使用数组下标来访问窗体外还可以使用框架名和窗体名来实现各元素的访:

parent. frames Name. decument. form Names. element Name. (m/p)

### 三、范例

下面我们通过一个具体的实例,来说明利用JavaScript脚本在WEB中实现更为复杂的信息交 互。该例子是在一个多窗口中实现窗体信息的动态交互, 在程序中首先在浏览器窗口中制作三个用 于窗体交互的窗口,每个窗体窗口实现不同信息的动态交互。

tset9. html 为主调用文档它首先将窗口划分为具有二行的窗体, 尔后再将第二行的窗体划分 为具有二列的窗体

```
</HTML>
```

#### 2. 第一个框架

主要作用是显示标题文档。

### Test9\_1.htm

```
<HTML>
<HEAD>
</HEAD>
<H2>使用框架实现WEB交互</H2>
</HTML>
```

#### 3. 第二个框架

主要作用是实现交互。可以通过JavaScript脚本将所示的"云南省"和"四川省"分别改 为"昆明市"和"成都市"。

```
<HTML>
      <HEAD>
      </HEAD>
      <Body>
      <Form name="test9 1">
      请选择城市:〈BR〉
      <Select name="select1" Multiple>
      <0ption>云南省
      <0ption>四川省
      <0ption>贵州省
      <0ption>山东省
      <0ption>江苏省
      <0ption>浙江省
      <0ption>安徽省
      <0ption>河南省
      </select><BR>
      <HR>
       Type-m>

ipt language=",document.best9_1.e.
document.test9_1.ele.

//script>
/pre>
dy>

BR>

Con

** 息(BI

** < P
     <Input Type="Submit" name="" value="提交">
<Input Type="reset" name="" value="复位">
      </Form>
      <script language="JavaScript">
       </script
      </Body>
      </HTML>
  4. 第三个框架
      主要作用是实现交互。
Test9_3.htm
<HTML>
<HEAD>
</HEAD>
<Body>
<Form name="test9_2">
请输入用户名:
<Input Type="text" name="text1" Value="" Size=20><BR</pre>
<HR>
请选择:
<Input Type="Checkbox" name="checkbox1" Value="qb">全部信息<BR>
<Input Type="Checkbox" name="checkbox2" Value="bf">部分信息<BR>
<Input Type="Checkbox" name="checkbox3" Value="sy">所有城市<br>
<Input Type="Submit" name="" value="提交">
<Input Type="reset" name="" value="复位">
<BR>
</Form>
<script language="JavaScript">
   document.test9_2.elements[0].value="劳动和社会保障";
   document.test9_2.elements[1].checked=true;
   document.test9_2.elements[2].checked=true;
   document.test9_2.elements[3].checked=false;
</script>
</Body>
</HTML>
```

在浏览器中的结果见图9-2所示。

This document is created with trial version of CHM2PDF Pilot 2.15.82. C:\my\Javascrip\efo^7. htm - Microsoft Internet Explorer 编辑(E) 查看(V) 转到(G) 帮助(H) 文件(图) 收藏(4) 製頻道 0 **多** 历史 4 \* 읦 対 搜索 前进 収藏 停止 地址 🥝 C:\my\Javascrip\efo~7.htm • 链接 <u>.</u> • 请输入用户名: 请选择城市: 昆明市 🔺 劳动和社会保障 成都市 贵州省 请选择: 🗹 全部信息 山东省 ▼ ☑部分信息 口所有城市 提交| 复位 复位 提交|

💹 我的电脑

图 9-2 在浏览器中结果

本讲介绍框架中的基本元素和企要功能和使用,利用JavaScript脚本可以非常方便、灵活地实现Web页面更为复杂的信息交流,这是HTML标识语言所不能具备的。从中可以看出JavaScript是多么的吸引众多的Web设计。