博硕文化

李展谋 编著

DHTML 实例应用

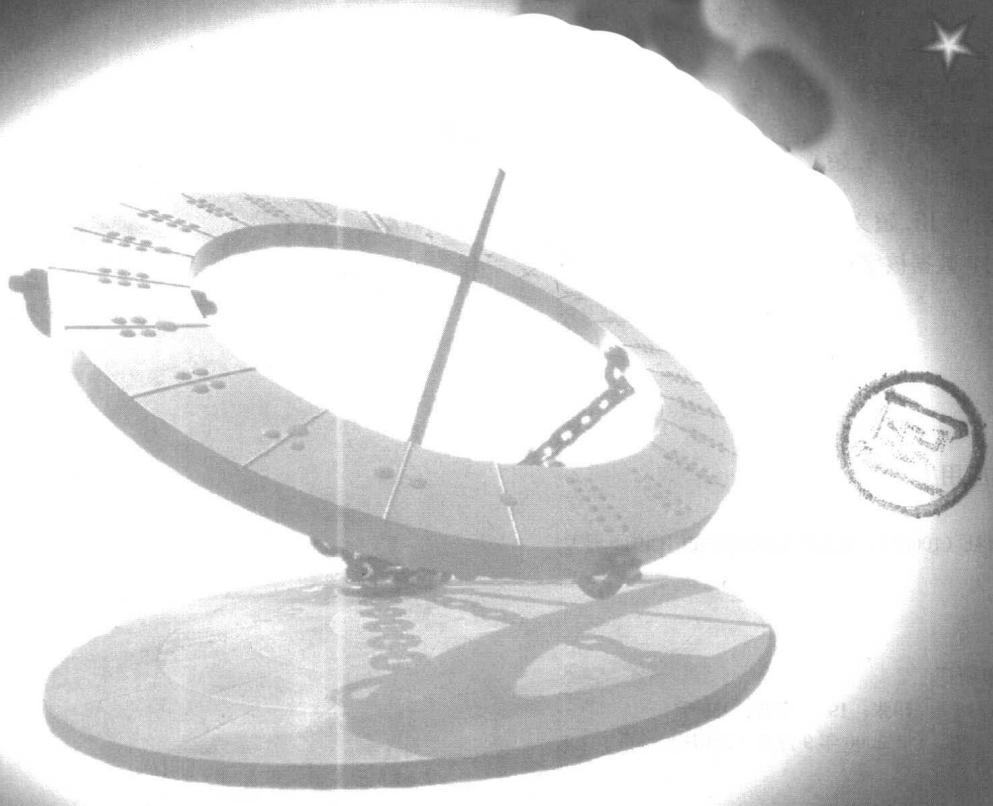
内附光盘

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

DHTML
实例应用
李展谋 编著
博硕文化

DHTML 实例应用

李展强 编著



中国铁道出版社

（京）新登字 063 号

北京市版权局著作权合同登记号 01-1999-1309 号

版 权 声 明

图书在版编目（CIP）数据

DHTML 实例应用/李展谋编著.—北京：中国铁道出版社，2000.9

ISBN 7-113-03869-7

Ⅰ.D… Ⅱ.李… Ⅲ.超文本标记语言，DHTML—程序设计 Ⅳ.P312

中国版本图书馆 CIP 数据核字(2000)第 43491 号

书　　名：DHTML 实例应用
作　　者：李展谋
出版发行：中国铁道出版社（100054，北京市宣武区右安门西街 8 号）
责任编辑：严晓舟
特邀编辑：刘　静
封面设计：冯龙彬
印　　刷：北京市燕山印刷厂
开　　本：787×1092　1/16　　印张：19　字数：459 千
版　　本：2000 年 9 月第 1 版　　2000 年 9 月第 1 次印刷
印　　数：1~5000 册
书　　号：ISBN7-113-03869-7/TP・470
定　　价：41.00 元

出版说明

自从 HTML4 的标准制定后，各家浏览器纷纷开始支持 Dynamic HTML，以强化客户端浏览器的动态效果，希望在不增加网络带宽的负担下，开发出更友善、更具亲和力的客户端网页。

本书详细地讲解 DHTML 的基本概念、层叠样式表、视觉滤镜效果、转换滤镜效果、与客户互动的界面、互动式网页、客户端的动态数据表及动态网页等内容，并且配有各种应用范例及剖析说明，能帮助读者迅速掌握其基本原理、技术方法和开发应用。

本书由台湾博硕文化有限公司出版提供版权，中国铁道出版社计算机图书项目中心审选；孙国瑞、李霞、王占清、鲁胜辉、宁夕等完成整稿工作；孟丽花、颜耳顺、肖志军、廖康良完成了本书的排版工作。

中国铁道出版社

2000．8

目　录

第 1 章　DHTML 基本概念

为什么要学 DHTML？您曾访问过 http://www.geocities.com/？在它还没和 yahoo 合作前，曾经有一阵子当使用者点击其下的网站时，会在该网站的右上角出现一个很有特色的小写“g”，它是一个水印(使用者可以看到其背后的文件内容)，并且无论网页怎么卷动，都一直停留在右上角的位置，更炫的是，如果有使用者去点击这个水印，它还提供超级链接的功能到 GEOCITIES 的首页，而当时 DHTML 的功能刚被提出来不久。

如果您无缘见到这个当时蛮具吸引力的设计或印象不很深刻，没关系，请参考程序 CH1-1.HTM，这个例子提供类似的设计，将有一枝花停留在屏幕的左上角，因为这是例子，所以点击后将出现一个信息提示窗口，而不是超级链接到哪个网址，请您先自书附光盘上执行看看，程序的画面将会像图 1-1。

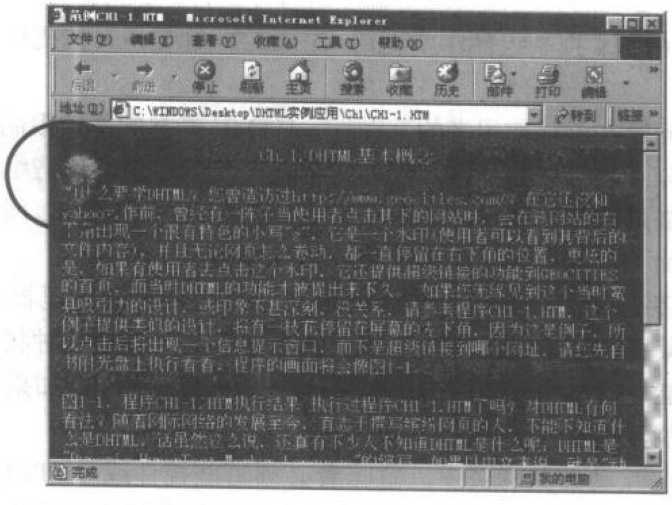


图 1-1　程序 CH1-1.HTM 执行结果，请注意左上角的那枝花

执行过程序 CH1-1.HTM 了吗？对 DHTML 有何看法？随着互联网的发展至今，有志于撰写缤纷网页的人，不能不知道什么是 DHTML，话虽然这么说，还真有不少人不知道 DHTML 是什么呢；DHTML 是“Dynamic HyperText Markup Language”的缩写，如果以中文来说，就是“动态超文本标示语言”，在开始第一节的详细说明之前，先请各位读者建立一个概念，“DHTML 是一个形容词”，而不是哪一种标准或什么产品的名称。

什么人需要学 DHTML 呢？如果您希望您制作的网页能与使用者有更佳的互动性，而且不准备自己写 Active X 控制组件或 APPLET，那么 DHTML 绝对值得您花点时间去了解；因为这种动态 HTML 所强调的就是在客户端与使用者做实时的互动，而且也因为其本质上还是 HTML，所以在传送网页的速度上，比含有 Active X 控制组件或 APPLET 的网页快上许多；当然，只要您高兴，您也还是可以在 DHTML 的网页中，使用 Active X 控制组件或 APPLET 来让您的网页有不一样的效果。

有了这基本概念后，就可以进入 DHTML 的世界了，让我们先来看看它的发展过程。

1-1 DHTML 的族谱

在本章一开始，我们请读者们建立了“DHTML 是个形容词”的印象，原因在于 DHTML 是种“动态的”HTML；而所谓的动态，是指网页可以立即直接与使用者产生实时的互动，不像传统静态的 HTML，只能单纯地显示信息，而且除了被动地响应使用者的操作外，DHTML 还可以自主地让 HTML 真正地动起来，例如在网页上跑来跑去。

经此介绍您可能已经想到“HTML 原来是静态的”，没错，最初发展互联网的目的，是在设计出一个信息交流的途径，以便让全世界相关的研究机构，都能分享彼此的研究成果或是做学术交流，而这些研究成果基本上以类似书面的方式显示即可，如果需要互相联络讨论，再通过电子邮件即可。

也就是因为初期的网页数据只要以类似印刷品的方式显示就够了，因此全球信息网协会（World Wide Web Consortium，W3C）在前几版推荐的 HTML 规格中，并未对 HTML 的整体结构做大幅度地修正；随着互联网的商业化，与相关应用的增加，人们不再满足于书面式单向传递的网页，开始要求网页应该表现得像应用程序一样，要能够与使用者有互动的关系，此时，APPLET 与 Active X 控制组件，就利用<OBJECT>标记，开始提供网页与使用者间的互动。

传统网页最大的问题在于与使用者的互动性。HTML 静态的本质限制了互动性的发展，虽然还是可以通过某些互动组件的使用，但是无论是产生或再使用这些组件都不太容易；而网页制作者面临的挑战是如何让他设计的网页有更好的互动性，HTML 网页静态的本质限制了他们创作的选择，而且互动组件也不容易制作或重新使用。

换句话说，使用专有的扩充设计意味着使用者必须用特定的浏览器才可以看到这个网页，这背后的意义是某些使用者无法体验您网页的奥妙，为了避免这种状况，通常设计人员不是放弃这些扩充功能，就是为不同的浏览器制作个别的网页；因此如果您要在单纯 HTML 的环境中提供互动功能，往往代表着工程浩大和所费不低。

因为在以往要加些互动性到网页中，意味着必须写些控件或 APPLET，并且要试着用 SCRIPT 将这些东西结合在网页中，虽然这些组件多可以做些特殊的工作，但实际上许多网页的制作者认为这比撰写 HTML 或 SCRIPT 难多了。因此，DHTML 的出现，让使用者可以利用此技术充分发挥网页应用程序的特性，并且可以重复使用以 DHTML 为设计基础的网页内容。

DHTML 是 HTML4.0 推出后，浏览器的新兴标准；就 HTML4.0 来说，这个规格定义了一种用于互联网出版的语言，若与前一版 HTML 比较，除了增加文字、多媒体、与超级链接的特色外，HTML4.0 支持更多的多媒体选项、脚本(SCRIPT)语言、样式表、更好的打印控制、让残疾人士更容易存取的文件，也让文件的国际化迈进了一大步，使得网络更加的全球化。

HTML4.0 是所谓“标准一般化标示语言”(Standard Generalized Markup Language，SGML）的应用，并受国际标准 ISO8879 所认可。但是 DHTML 则不同，还记得开始介绍的一个概念吗？DHTML 是一个形容词，它所指的是浏览器一种提供“动态”网页的方式，因此大体上各家浏览器虽然都支持 HTML4.0 的标准，可是在提供“动态”网页的方式上，却有相当不同的做法，这种差异尤其在目前市场占有率最高的前两大浏览器(IE 和 Navigator)上最为明显。

由于 DHTML 的做法也将因浏览器而异，在许多因素的衡量下，我们书中的例子都将以

微软的 IE5 为客户端的浏览器，仅在第 2 章介绍对象模型的地方用到几次 Navigator 来示范；如果您能掌握第二节所介绍 DHTML 的组成，相信应该很容易将相同的概念应用到不同的浏览器上，毕竟互联网最吸引使用者的地方，就是提供了一个能让文件轻易交换的标准格式。

1-2 DHTML 的组成

在探讨 DHTML 之前，我们要先回顾程序 CH1-1.HTM，因为在这个程序中，恰好具备 DHTML 的三个组成要素，那就是 HTML、SCRIPT 和浏览器的对象；如果您对 HTML 已经有些基础，您应该可以轻易认出在程序 CH1-1.HTM 中的以下片段，就是标准的 HTML：

```
<html>
<head>
<script LANGUAGE="VBScript">
.....
</script>
<title>范例 CH1-1.HTM</title>
</head>
<body bgcolor="#000000" text="#FFFFFF" onLoad="RightThere()" onScroll="RightThere()"
onResize="RightThere()">
<div ID="flower" Style="position:absolute; Left:-40; z-index:1">
<dd><img src="test.gif" width="53" height="73"> </dd>
</div>
<p align="center"><font size="4">Ch.1.DHTML 基本概念</font></p>
<p>为什么要学 DHTML？您曾造访过 http://www.geocities.com/？在它还没和 yahoo 合作前，曾经有一阵子当使用者点击其下的网站时，会在该网站的右上角出现一个很有特色的小写&amp;quot;g&amp;quot;，它是一个水印(使用者可以看到其背后的文件内容)，并且无论网页怎么卷动，都一直停留在右上角的位置，更炫的是，如果有使用者去点击这个水印，它还提供超级链接的功能到 GEOCITIES 的首页，而当时 DHTML 的功能才被提出来不久。
如果您无缘见到这个当时蛮具吸引力的设计、或印象不很深刻，没关系，请参考程序 CH1-1.HTM，这个例子提供类似的设计，将有一枝花停留在屏幕的左上角，因为这是例子，所以点击后将出现一个信息提示窗口，而不是超级链接到哪个网址，请您先自书附光盘上执行看看，程序的画面将会像图 1-1。
</p>
<p>图 1-1、程序 CH1-1.HTM 执行结果 执行过程序 CH1-1.HTM 了吗？对 DHTML 有何看法？随着互联网的发展至今，有志于撰写缤纷网页的人，不能不知道什么是 DHTML，话虽然这么说，还真有不少人不知道 DHTML 是什么呢；DHTML 是&amp;quot;Dynamic HyperText Markup Language&amp;quot;的缩写，如果以中文来说，就是&amp;quot;动态超文本标示语言&amp;quot;，在开始第一节的详细说明之前，先请各位读者建立一个概念，&amp;quot;DHTML 是一个形容词&amp;quot;，而不是哪一种标准或什么产品的名称。
</p>
```

```
<p>什么人需要学 DHTML 呢？如果您希望您制作的网页能与使用者有更佳的互动性，而且不准备自己
写 ACTIVE X 控制组件或 APPLET，那么 DHTML 绝对值得您花点时间去了解；因为这种动态 HTML
所强调的就是在客户端与使用者做实时的互动，而且也因为其本质上还是 HTML，所以在传送网页的
速度上，比含有 ACTIVE X 控制组件或 APPLET 的网页快上许多；当然，只要您高兴，您也还是可以
在 DHTML 的网页中，使用 ACTIVE X 控制组件或 APPLET 来让您的网页有不一样的演出。
有了这基本概念后，就可以进入 DHTML 的世界了，让我们先来看看它的发展过程。
</p>
</body>
</html>
```

程序 CH1-1.HTM 中，标准 HTML 部分的程序代码

标准的 HTML 是由标记(TAG)和文件的内容构成；在我们的例子里内容的部分就像程序行表中与本章开头相同的那些文字，当然也包括在图 1-1 的那枝花的图案，只是代表花的这个内容，是被包含在<IMG>标记内的 URL()，而非我们直觉所想的文字内容。

至于 HTML 标记的部分就很容易辨识了，可以说在程序 CH1-1.HTM 里所有用一组角括号“<”与“>”括着的，就是 HTML 标记，而且字母大小写不限，因此您可以在程序中看到如<HTML>、<TITLE>、<BODY>等都是由于本书的重点在介绍如何写出 DHTML 的网页程序，所以对标准 HTML 只再提醒两点，其他关于标准 HTML 的疑义，请参考相关书籍。

第一个要注意的地方是 HTML 标记不一定都成对出现；所谓成对出现指的是一个起始标记搭配一个结束标记，例如<TITLE>搭配</TITLE>，这代表的意义是在这组标记范围内的内容，在浏览器决定如何显示时，都将受这组标记的影响，例如<TITLE>这组标记的文字，就会在网页显示时，被浏览器当成是此网页的标题，而被放在浏览器窗口名称栏的最左方；在此强调的是，并非所有 HTML 标记都会成对出现，典型的例子如
标记，它的作用就是强制文字在该处换行，有时也会看到只用一个<P>代表自此以后都是同一段落的这种用法，因此在使用 HTML 标记时，最好弄清楚这些标记的写法。

另一个要提醒的主题，是关于 HTML 标记属性的写法；就像先前介绍一枝花的图形是用<IMG>这个标记所标示的，但是实际上存储此图形的文件(也就是先前所说的 URL)究竟在哪里，就必须靠此标记的“SRC”属性来决定，因此您可以从程序中看到这一行“src="test.gif" width="53" height="73"”，聪明的你是否能推想到在“SRC”属性后面跟着的“WIDTH”和“HEIGHT”也都是<IMG>标记的属性呢？没错，它们就代表这枝花图形的宽度与高度；在程序 CH1-1.HTM 中，除了<IMG>标记以外，<SCRIPT>、<BODY>、<P>、<FONT>等标记都有带些属性在其中，您可以试着将这些属性辨识出来，请记得，适度运用 HTML 标记属性，将使网页更精确生动。

在上一个程序片段中，我们省略了接下来要介绍“脚本(SCRIPT)”程序的部分，这个部分可以说是 DHTML 的精神所在，从程序的角度来看，动态网页之所以能“动”，就是因为背后有这些程序的关系；也因为本书的例子中，几乎全部都有“脚本(SCRIPT)”程序的踪影，因此我们都称这些例子为“程序”，而非一般的“网页”。

关于“SCRIPT”，有许多种译法，我们比较喜欢用“脚本”来代表它，因为它的工作，

就是在网页上完成设计师精心设计的逻辑程序，和我们看的电视、电影上演员都照剧本演出的情形十分类似，因此我们把“SCRIPT”称为“网页剧本（脚本）”；以下这个片段，就包含了两段写在程序 CH1-1.HTM 里的“脚本”，请参考。

```
<script LANGUAGE="VBScript">
<!--//
Sub Flower_onClick()
   Call window.alert("您点到程序 CH1-1.HTM 的花了")
End sub

Sub RightThere()
   dim where_y
   where_y = document.body.scrollTop + document.body.clientHeight - document.all.flower.offsetHeight
   flower.style.top = where_y
End Sub
//-->
</script>
```

程序 CH1-1.HTM 中，“脚本（script）”部分的程序代码

在 HTML 中，“脚本(SCRIPT)”程序通常被括在<SCRIPT>与</SCRIPT>标记组中，<SCRIPT>这个标记里最重要的一个属性，就是标示其内这些“脚本(SCRIPT)”程序代码所用的是哪种语言，以程序 CH1-1.HTM 为例，我们就将它标示成“LANGUAGE=VBSCRIPT”，代表其中的“脚本(SCRIPT)”程序是用 VBSCRIPT 这个语言写成的。

请注意先前介绍的这个特性，因为既然我们可以用“LANGUAGE”这个属性标示<SCRIPT>与</SCRIPT>标记组内的程序代码是用哪一种语言来撰写的，那么是否意味着同一份网页，可用多种不同的程序语言来撰写“脚本(SCRIPT)”程序？答案是肯定的，因为我们可以视需要而将<SCRIPT>与</SCRIPT>标记组放在网页中的任何地方，且各组“脚本(SCRIPT)”使用不同的语言，换句话说，您可以让网页程序加载时，先用 JAVA SCRIPT 处理某些工作，然后再用 VB SCRIPT 使网页上个别组件与使用者互动。

事实上，目前两大浏览器的各自支持不同的“脚本(SCRIPT)”语言，前一段语句中谈到的 VB SCRIPT 与 JAVA SCRIPT，就恰巧分别是微软的 IE 和网景的 Navigator 所支持的语言，其中网景的 Navigator 至今还是只支持 JAVA SCRIPT，而且需是纯种的 JAVA SCRIPT，之所以如此介绍的原因，在于微软也推出微软版的 JAVA SCRIPT，而且与所谓的纯种 JAVA SCRIPT 略有不同，所以当您用相同的 JAVA SCRIPT 程序在 IE 可以顺利执行，而在 Navigator 产生意外时，很可能就是这个原因。

至于 VB SCRIPT 虽然只有微软的 IE 支持，但由于 VISUAL BASIC 所打下的基础与微软免费浏览器的策略，声势也直逼曾经独霸的 JAVA SCRIPT，甚至有凌驾其上的趋势，因此在本书的例子中，除了稍后的 CH1-2.HTM 与第 2 章的部分例子外，其他都是以 VB SCRIPT 为主；当然，语言本身并无优劣的问题，我们的考虑主要着眼于普遍性，因此若您需要开发放诸四海皆准的 DHTML 网页程序，除了本书的概念外，还需要研习如何使用 JAVA SCRIPT

去操作 Navigator 的对象。

我们用程序 CH1-2.HTM 来看看这两种“脚本(SCRIPT)”语言有何不同，和它们在程序中一起出现的做法；类似程序 CH1-1.HTM 的做法，这个例子中，我们提供左右上角各一枝花，无论使用者如何滚动网页，这两枝花都会坚守岗位，停留在左右上角的位置上，而控制这个操作的程序就是用 JAVA SCRIPT 完成的，请参考以下这段程序代码，并与先前程序 CH1-1.HTM 所列“脚本(SCRIPT)”部分的片段做一比较。

```
<script LANGUAGE="JavaScript">
<!--//
function RightThere(){
   var where_y = document.body.scrollTop + document.body.clientHeight -
document.all.flower2.offsetHeight;
   flower2.style.top = where_y;
   flower4.style.top = where_y;
}
.....
//-->
</script>
```

程序 CH1-2.HTM 中，将两朵花定位部分的程序代码

程序 CH1-2.HTM 执行的结果就像图 1-1 一样，不过右上角多了一枝花，因此就不再特别列出，如果好奇的话，请自己执行一下光盘片上的程序，或是想一下在图 1-2 或图 1-3 中，将画面中的输入对话框除去就是了。以下是与左边那支花被点击后相关的程序代码，请参阅：

```
<script LANGUAGE="VBScript">
<!--//
Sub flower2_onClick()
   dim temp
   temp = inputbox("您点到左下角的花了","这是 VB SCRIPT 的输入对话框","左下角的花")
End Sub
//-->
</script>
```

程序 CH1-2.HTM 中，左边花被点击后相关的程序代码

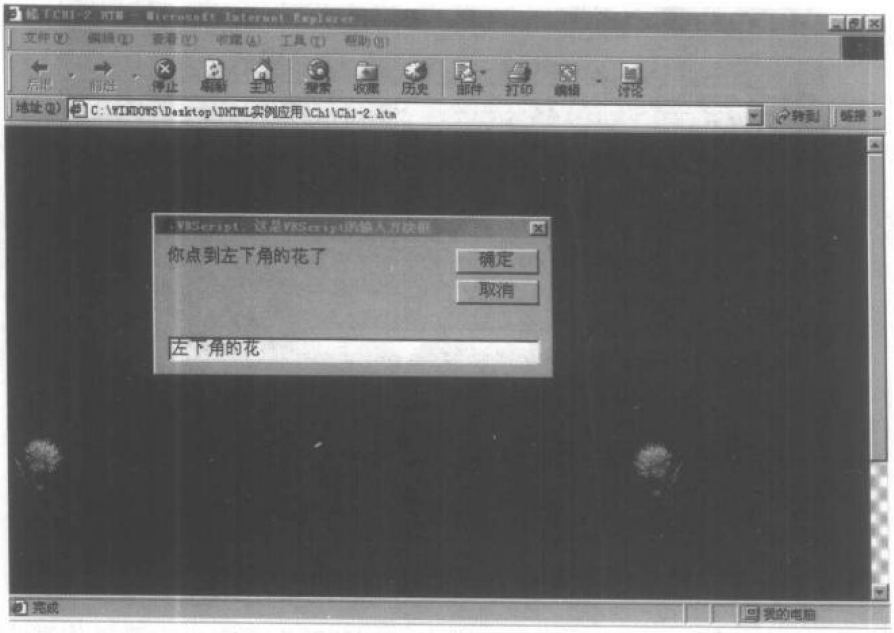


图 1-2 由 VB SCRIPT 产生的输入对话框

若使用者点击左下角的花，画面上将会出现一个 VB SCRIPT 专属的输入对话框让使用者输入数据，这个画面就在图 1-2；反之，如果使用者点击右下角的花，画面上会弹出一个 JAVA SCRIPT 的输入对话框，其功能与图 1-2 那个输入对话框一样，我们将这个画面显示在图 1-3。补充说明一点，图 1-3 的输入对话框可以在 Navigator 中使用，而图 1-2 的则否，原因在稍前已经提过了，至于这些输入对话框在第 7 章会有进一步的探讨。在此还是先参考右边那枝花被点击后相关的程序代码，并且建议您和左边那枝花对应的程序(在图 1-2 之前)做个比较，请参阅：

```
<script LANGUAGE="JavaScript">
<!--//
.....
function flower4_onClick(){
   var temp = prompt("您点到右下角的花了","右下角的花");
}
//-->
</script>
```

程序 CH1-2.HTM 中，右边花被点击后相关的程序代码列表

组成 DHTML 的第三大部分，是浏览器的“对象”；如果您写过其他的窗口应用程序，相信您对“对象”的概念应该有相当的了解，所以您还需要知道的，应该是关于浏览器究竟提供哪些对象与这些对象又有哪些属性、事件和可用的方法；万一您对对象的概念仍然模糊，也不必过于紧张，因为浏览器提供的对象，简单来说就是构成网页的那些东西，如最基本的窗口(WINDOW)、文件(DOCUMENT)等。

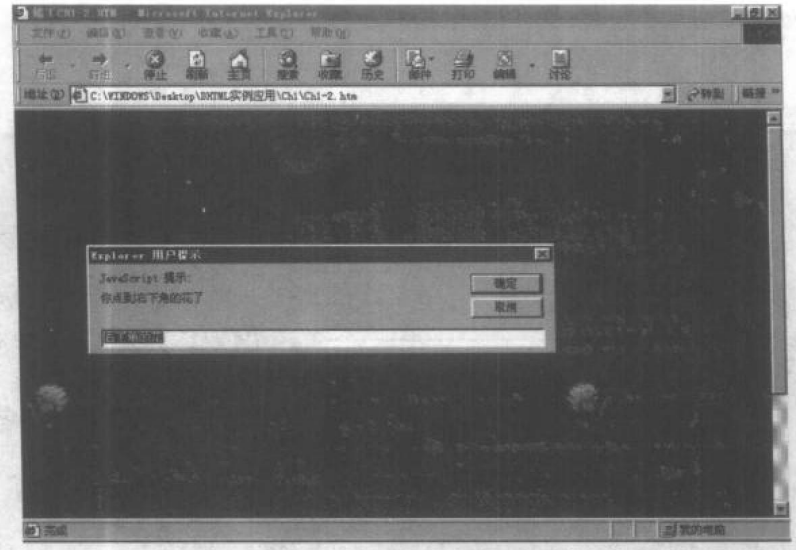


图 1-3　由 JAVA SCRIPT 产生的输入对话框

一个好的 DHTML 网页，必然需要在“脚本(SCRIPT)”程序内大量操作浏览器的对象，可能是设置其“属性(property)”，也可能是运用其“方法(method)”，因此对浏览器对象了解的程度，将直接影响到 DHTML 网页的品质，所以我们在第 2 章就分别有系统地介绍 IE 和 Navigator 的对象模型，不论您对“对象”的了解如何，应该都能很快的掌握要点，并对撰写生动的 DHTML 网页程序有相当的帮助。

1-3　在 Navigator 与 IE 中的异同

由于 DHTML 本质上还是 HTML，那么它神奇的互动功能怎么来的呢？答案就在浏览器上，原来从 HTML4 的标准推出后，微软与网景都相继推出支持此版本的浏览器，分别是 IE 与 Navigator，但是问题就出在这两款浏览器对“动态的”支持不尽相同，而本书所有的范例都基于 IE5，所以利用这一节做个简单地比较，让您在熟悉本书的例子后，可以将 DHTML 的概念应用到 Navigator 上，以拓展您精心设计网页的势力范围。

在程序 CH1-2.HTM 中，我们已经粗略地看过 VB SCRIPT 和 JAVA SCRIPT 的不同，但是除了“脚本(SCRIPT)”程序不尽相同外，IE 和 Navigator 另一个重要的差异在第 2 章要介绍的对象模型上。

解决“脚本(SCRIPT)”程序不尽相同的办法之一是采用差异较小的 JAVA SCRIPT，或是针对不同的浏览器撰写不同的“脚本(SCRIPT)”程序，但是对象模型的不同，则迫使设计师在做某些精巧的设计时，非得针对不同的浏览器撰写不同的“脚本(SCRIPT)”程序；举例来说，Navigator 支持以“图层(LAYER)”显示的做法，而 IE 则否，如果您的程序必须用到“图层(LAYER)”，则无论您是否使用微软版的 JAVA SCRIPT，在 IE 中都势必需要改写方能适用。

虽然浏览器对“脚本(SCRIPT)”程序或对象模型并非百分之百的支持，但由于全球信息网(WORLD WIDE WEB)的基本精神是要共享信息，或许是这个原因，两大浏览器的差异才没有像操作系统那么多，而且彼此都还有替代方法能支持，像先前提到“图层(LAYER)”的问题，还是可以考虑使用“DIV”标记来取代，尽管这两个标记在使用上仍有许多差异存在。

程序 CH1-3.HTM 与程序 CH1-4.HTM 就是分别以 LAYER 和 DIV 的做法，让网页上的四朵花自动从四个角落移动到中心，再从中心回到四个角落；程序的基本逻辑并不困难，但是您需要分别知道 LAYER 和 DIV 这两个网页组件的特性与用法，无论是在 HTML 的语句上还是在 SCRIPT 程序中；请参考下面的程序行表：

```
<html>

<head><script Language="javascript">
var concentrate
var separate
var xmove = 30;
var ymove = 13;

function showNow(){
   concentrate = setInterval("ToConcentrate()",100)
}

function ToConcentrate(){
   document.layers["flower1"].offset( xmove, ymove);
   document.layers["flower2"].offset( xmove, -ymove);
   document.layers["flower3"].offset( -xmove, ymove);
   document.layers["flower4"].offset( -xmove, -ymove);
   if (document.layers["flower1"].top >100 ) {
      clearInterval( concentrate );
      separate = setInterval("ToSeparate()",100);
   }
}

function ToSeparate(){
   document.layers["flower1"].offset( -xmove, -ymove);
   document.layers["flower2"].offset( -xmove, ymove);
   document.layers["flower3"].offset( xmove, -ymove);
   document.layers["flower4"].offset( xmove, ymove);
   if (document.layers["flower1"].top <0 ) {
      clearInterval( separate );
      concentrate = setInterval("ToConcentrate()",100);
   }
}
</script>
```