

第6章 函数和window对象



本章目标

- ✓使用函数实现代码的重用
- ✓使用window对象的open()方法制作广告窗口
- ✓使用window对象的setinterval()方法实现时钟特效

什么是函数

- 函数的含义：类似于Java中的方法，是完成特定任务的代码语句块

- 使用更简单：不用定义属于某个类，直接



提问 使用 如果希望单击某个按钮后在页面显示“HelloWorld”，并且能够控制语句显示的次数，怎么办？

演示示例：调用函数输出“HelloWorld”



JavaScript核心语法

- 系统函数
- 自定义函数
 - 有参函数
 - 无参函数
 - 函数的调用



常用系统函数

○ parseInt ("字符串")

将字符串转换为整型数字

如: `parseInt ("86")` 将字符串 “86” 转换为整型值 86

○ parseFloat ("字符串")

将字符串转换为浮点型数字

如: `parseFloat ("34.45")` 将字符串 “34.45” 转换为浮点值 34.45

○ isNaN()

用于检查其参数是否是非数字

演示示例: [类型转换函数](#)

自定义函数

■ 创建函数

■ 无参函数

```
function 函数名()  
{  
    JavaScript代码;  
}
```

■ 有参函数

```
function 函数名(参数1,参数2, ... )  
{  
    JavaScript代码;  
}
```

○ 调用函数

- 函数调用一般和表单元素的事件一起使用，调用格式：

事件名 = “函数名()” ;

调用无参函数

■ 调用无参函数，输出 10次

```
function showHello()  
{  
  for(var i=0;i<5;i++)  
  {  
    document.write("<h2>Hello World</h2>");  
  }  
}
```

单击此按钮时，调用函数showHello()，执行函数体中的代码

```
<input name="btn" type="button" value="显示10次HelloWorld"  
onclick="showHello()" />
```

演示示例：调用无参函数输出“HelloWorld”

调用有参函数

- 根据输入的次数，调用有参函数，显

```
function showHello(count)
{
    for(var i=0;i<count;i++)
    {
        document.write("<h2>Hello World</h2>");
    }
}
```

单击此按钮时，调用函数showHello(count)，执行函数体中的代码

```
<input name="btn" type="button" value="请输入显示HelloWorld的次数"
onclick="showHello(prompt('请输入显示HelloWorld的次数：',''))"/>
```

演示示例：调用有参函数输出“HelloWorld”

变量的作用域

全局变量

```
<script language="javascript" type="text/javascript">  
var num;  
function showHello(){  
    for(var i=0;i<num;i++)  
    {  
        document.write("<h2>Hello World</h2>");  
    }  
}  
function counts(){  
    num=prompt("请输入显示HelloWorld的次数: ","");  
    showHello();  
}  
</script>
```

局部变量

```
<input name="btn" type="button" value="请输入显示HelloWorld的次数"  
onclick="counts()"/>
```

演示示例：变量的作用范围



小结

- 如何创建一个有参函数并实现函数调用？
- 全局变量和局部变量的区别是什么？



Window对象

- 常用属性
- 常用方法
- 常用事件



Window对象的常用属性

○ 常用的属性

名称	说明
screen	有关客户端的屏幕和显示性能的信息
history	有关客户访问过的URL的信息
location	有关当前 URL 的信息



Window对象的常用方法

常用的方法

名称	说明
prompt	显示可提示用户输入的对话框
alert	显示带有一个提示信息和一个确定按钮的警示框
confirm	显示一个带有提示信息、确定和取消按钮的对话框
close	关闭浏览器窗口
open	打开一个新的浏览器窗口，加载给定 URL 所指定的文档
setTimeout	在指定的毫秒数后调用函数或计算表达式
setInterval	按照指定的周期（以毫秒计）来调用函数或表达式



confirm()方法

语法

confirm("对话框中显示的纯文本")

```
<script language="javascript" type="text/javascript">
var flag=confirm("确认要删除此条信息吗? ");
if(flag==true){
    alert("删除成功! ");
}
else{
    alert("你取消了删除");
}
</script>
```

- **confirm()** 与 **alert()** 、
prompt()区别

open()方法

语法

窗口特征

名称	说明
height、width	窗口文档显示区的高度、宽度。以像素计。
left、top	窗口的x坐标、y坐标。以像素计
toolbar=yes no 1 0	是否显示浏览器的工具栏。默认是yes。
scrollbars=yes no 1 0	是否显示滚动条。默认是yes。
location=yes no 1 0	是否显示地址地段。默认是yes。
status=yes no 1 0	是否添加状态栏。默认是yes。
menubar=yes no 1 0	是否显示菜单栏。默认是yes。
resizable=yes no 1 0	窗口是否可调节尺寸。默认是yes。
titlebar=yes no 1 0	是否显示标题栏。默认是yes。
fullscreen=yes no 1 0	是否使用全屏模式显示浏览器。默认是no。处于全屏模式的窗口必须同时处于剧院模式。



Window对象的常用事件

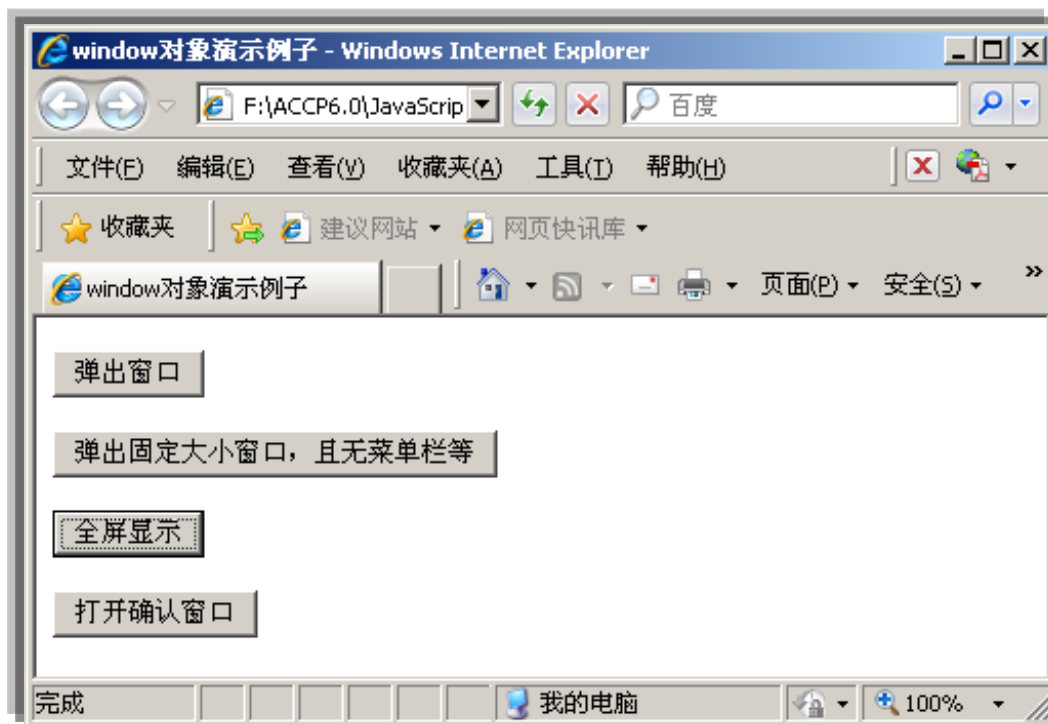
○ 常用的事件

名称	说明
onload	一个页面或一幅图像完成加载
onmouseover	鼠标移到某元素之上
onclick	当用户单击某个对象时调用的事件句柄
onkeydown	某个键盘按键被 按下
onchange	域的内容被改变

如何使用window对象-1

示例

- 弹出窗口、弹出固定大小且无菜单栏的窗口
- 当前页面全屏显示
- 弹出确认消息框





如何使用window对象-2

示例

示例完整代码

```
<script type="text/javascript">
function open_adv(){
    window.open("adv.html");
}
function open_fix_adv(){
    window.open("adv.html","", "height=380,width=320,toolbar=0,
scrollbars=0,location=0,status=0,menubar=0,resizable=0");
}
function fullscreen(){
    window.open("plan.html","", "fullscreen=yes");
}
function confirm_msg(){
    if(confirm("你相信自己是最棒的吗? "))
        { alert("有信心必定会赢， 没信息一定会输! "); }
}
</script>
```



如何使用window对象-2

示例

示例完整代码

```
<input name="open1" type="button" value="弹出窗口" onclick="open_adv()" />  
<input name="open2" type="button" value="弹出固定大小窗口，且无菜单栏等"  
onclick="open_fix_adv()"/>  
<input name="full" type="button" value="全屏显示" onclick="fullscreen()"/>  
input name="con" type="button" value="打开确认窗口"  
onclick="confirm_msg()"/>
```



匿名调用函数

语法

Function 函数名() {JavaScript代码}

调用函数: **onclick="函数名() "**

○ 网上常见匿名写法

- 方式一: 事件名=function(){...}
- 方式二: (function () { JavaScript代码; }())

○ 直接运行函数

- window.onload=function(){...}
- <body onload= "事件名() " >

演示时钟特效

问题

如何实现如演示示例中所示的时钟效果？

演示示例：[时钟效果演示](#)

分析

由于涉及到时间的显示问题，所以要用到日期对象Date，还有时间在不停地走，因此需要不断地调用函数，所以要用到Windows的定时器setInterval()方法。

Date对象

语法

var 日期对象=new Date(参数)

参数格式: MM DD,YYYY,hh:mm:ss

示例

```
var today=new Date(); //返回当前日期和时间  
var tdate=new Date("september 1,2009,14:58:12");
```

Date方法分组	说明
setXxx	这些方法用于设置时间和日期值
getXxx	这些方法用于获取时间和日期值



Date对象的方法

■ Date 对象的方法

值	整数
Seconds 和 Minutes	0 至 59
Hours	0 至 23
Day	0 至 6（星期几）
Date	1 至 31（月份中的天数）
Months	0 至 11（一月至十二月）

制作时钟特效-1

示例

```
<script language="javascript" type="text/javascript">
function disptime()
{ var today = new Date(); //获得当前时间
  var hh = today.getHours(); //获得小时、分钟、秒
  var mm = today.getMinutes();
  var ss = today.getSeconds();
  document.getElementById("myclock").innerHTML=hh+":"+mm
  +":"+ss;}
</script>

.....

<body onload="disptime()">
<div id="myclock"></div>

.....
```

示例完整代码



制作时钟特效-2

问题

制作的时钟特效示例中，时间为什么不改变？

分析

由于时间在不停地走，所以应该每隔1秒调用显示时间的方法，如何解决？

使用`setTimeout()`方法或`setInterval()`

定时函数

- **setTimeout()用法**

- **setTimeout** ("调用的函数", "指定的时间后")

- **setInterval()方法**

- **setInterval** ("调用的函数", "指定的时间间隔")

```
var myTime=setTimeout("disptime( ) ", 1000 );
```

```
var myTime=setInterval("disptime( ) ", 1000 );
```

每隔1秒(1000毫秒)执行函数disptime()一次

setTimeout()只执行disptime()一次，如果要多次调用使用setInterval()或者让disptime()自身再次调用setTimeout()

时钟显示特效



总结

提问

- **window**对象有哪些常用的方法及其含义？
- 请列举**Date**对象有哪些方法？
- 请解释**setTimeout()**方法与**setInterval()**方法的区别，及各自适用场合？
- 请举例说明事件**onload**和**onclick**的用法