

[首页](#) [资讯](#) [精华](#) [论坛](#) [问答](#) [博客](#) [专栏](#) [群组](#) [更多 ▼](#)
[您还未登录！登录](#)

fp-moon

- [博客](#)
- [微博](#)
- [相册](#)
- [收藏](#)
- [留言](#)
- [关于我](#)

JavaScript模态窗口和非模态窗口

javascript 窗口

JavaScript模态窗口和非模态窗口

JavaScript中弹出的窗口有模态窗口和非模态窗口。模态窗口就是打开一个子窗口，如果这个子窗口不关闭，就不能操作它的父窗口，原来程序暂停执行，直到这个模态窗口关闭

后才回到原来程序继续。非模态的就是直接显示出来，然后原来的程序继续执行下面的语句，而且其他窗口也呈可用状态。模态窗口独占了用户的输入，当一个模态窗口打开时，

用户只能与该窗口进行交互，而其他用户界面对象收不到输入信息。通常浏览器中window.open或超链接弹出的新窗口就是非模式窗口，而模式窗口是类似alert那种必须关闭才能

响应其他事件的窗口。

常见的方法

```
window.showModalDialog( "http://www.javachen.com" , " newwin" , " dialogHeight: 200px;  
dialogWidth: 150px; dialogTop: 458px; dialogLeft: 166px; edge: Raised;
```

```
center: Yes; help: Yes; resizable: Yes; status: Yes;" );
```

window.showModelessDialog() //用来创建一个显示HTML内容的非模态对话框。

window.returnValue=" javachen" ; //设置模态窗口返回值

var str=window.showModalDialog("http://www.javachen.com"); //获得模态窗口返回值

window.close(); //关闭窗口

window.focus(); //窗口聚焦

使用方法：

参数说明：sURL–必选参数，类型：字符串。用来指定对话框要显示的文档的URL。vArguments–可选参数，类型：变体。用来向对话框传递参数。传递的参数类型不限，包括数

组等。对话框通过window.dialogArguments来取得传递进来的参数。

sFeatures-可选参数，类型：字符串。用来描述对话框的外观等信息，可以使用以下的一个或几个，用分号“;”隔开。

1.dialogHeight :对话框高度，不小于100px，IE4中dialogHeight 和 dialogWidth 默认的单位是em，而IE5中是px，为方便起见，在定义modal方式的对话框时，用px做单位。

2.dialogWidth: 对话框宽度。

3.dialogLeft: 离屏幕左的距离。

4.dialogTop: 离屏幕上的距离。

5.center: {yes | no | 1 | 0} : 窗口是否居中，默认yes，但仍可以指定高度和宽度。

6.help: {yes | no | 1 | 0} : 是否显示帮助按钮，默认yes。

7.resizable: {yes | no | 1 | 0} [IE5 +] : 是否可被改变大小。默认no。

8.status: {yes | no | 1 | 0} [IE5+] : 是否显示状态栏。默认为yes[Modeless]或no[Modal]。

9.scroll:{ yes | no | 1 | 0 | on | off } : 指明对话框是否显示滚动条。默认为yes。

下面几个属性是用在HTA中的，在一般的网页中一般不使用。

10.dialogHide:{ yes | no | 1 | 0 | on | off } : 在打印或者打印预览时对话框是否隐藏。默认为no。

11.edge:{ sunken | raised } : 指明对话框的边框样式。默认为raised。

12.unadorned:{ yes | no | 1 | 0 | on | off } : 默认为no。

```
function showTDYT()
{
    //打开模态子窗体,并获取返回值
    var result=showModalDialog('土地储备-用途.htm','这里随便命名','dialogWidth:320px;dialogHeight:530px;center:yes;help:no;resizable:no;status:no');
    $("#txtYongTu").val(result);
}
```

子页面的方法和返回值：

```
function tc() //参数分别为id,name和password
{
    var getCC=$("#cc").text(); //获取想要的值
    window.returnValue= getCC; //返回 结果
    window.close(); //firefox不支持window.close()
}
```

使用窗口时的常见问题：

vReturnValue = window.showModalDialog(sURL[, vArguments][, sFeatures]); //模态

vReturnValue = window.showModelessDialog(sURL[, vArguments][, sFeatures]); //非模态

window.open(pageURL,name,parameters) //非模态 （pageURL 为子窗口路径；name 为子窗口句柄 parameters；为窗口参数(各参数用逗号分隔)）

非模态时使用open()方法比较好（因为各浏览器都基本对它兼容）

1、使用脚本关闭窗口时防止出现确认框

在做网页的时候，有时候在完成某个功能的时候需要关闭当前窗口，但现在的新版的浏览器中在使用脚本window.close() 方法来关闭窗口时，总是会出现一个确认框，这带来非

常不好的体验。

可以防止出现确认框：

```
window.opener = null;  
window.open("", "_self");  
window.close();
```

关闭窗口兼容写法实例：

```
function doLoad()  
{  
    var sUstrAgent=navigator.userAgent;  
    var isIE=sUstrAgent.indexOf("MSIE")!=-1;  
    var isFF=sUstrAgent.indexOf("Firefox")!=-1;  
    var isOP=sUstrAgent.indexOf("Opera")!=-1;  
    var isSF=sUstrAgent.indexOf("Safari")!=-1&& sUstrAgent.indexOf("Chrome")!=-1;  
    var isCH=sUstrAgent.indexOf("Chrome")!=-1;  
    if((isIE == 1 || isOP == 1 || isSF == 1 || isCH == 1)  
    {  
        window.opener=null;  
        window.open('','_self');  
        setTimeout("window.close()",3000);  
    }  
    else if(isFF == 1)  
    {  
        window.open('','_parent','');  
        setTimeout("window.close()",3000);  
    }  
    else  
    {  
        window.opener=null;  
        window.open('','_self');  
        setTimeout("window.close()",3000);  
    }  
}
```

2、防止 showModalDialog 方法打开网页时总是显示缓存

showModalDialog和showModelessDialog在第一次打开页面时会默认缓存该页面，如果再次打开相同URL的页面的话，他们会直接调用缓存中的页面，而不是从服务器返回，要不使

用缓存可进行如下配置：

<title>被打开的页面</title>

<meta http-equiv="pragram" content="no-cache"> //禁止浏览器从本地缓存中调阅页面，网页不保存在缓存中，每次访问都刷新页面。

<meta http-equiv="cache-control" content="no-cache, must-revalidate"> //同上面意思差不多，必须重新加载页面

<meta http-equiv="expires" content="0"> //网页在缓存中的过期时间为0，一旦网页过期，必须从服务器上重新订

上面的配置不一定有效果，所以不推荐使用，最好的办法是在URL后加上一个时间戳，如下：

```
url = url + "&time=" + new Date();
```

3.参数传递：

```
vReturnValue = window.showModalDialog(sURL[, vArguments][, sFeatures]);
```

```
vReturnValue = window.showModelessDialog(sURL[, vArguments][, sFeatures]);
```

i.要想对话框传递参数，是通过vArguments来进行传递的。类型不限制，对于字符串类型，最大为4096个字符。也可以传递对象，例如：

```
-----
parent.htm
<script>
var obj = new Object();
obj.name="51js";
window.showModalDialog("modal.htm",obj,"dialogWidth=200px;dialogHeight=100px");
</script>
```

```
modal.htm
<script>
var obj = window.dialogArguments || opener;
alert("您传递的参数为：" + obj.name)
</script>
```

ii.可以通过window.returnValue向打开对话框的窗口返回信息，当然也可以是对象。例如：

```
-----
parent.htm
<script>
str =window.showModalDialog("modal.htm",,"dialogWidth=200px;dialogHeight=100px");
alert(str);
</script>
```

```
modal.htm
<script>
window.returnValue="http://www.51js.com";
</script>
```

用window.showModalDialog 或者window.showModelessDialog打开一个模式窗口后，和父窗口的一些交互问题。

要进行交互操作的前提，在调用showModalDialog或者showModelessDialog方法的时候，第二个参数传window，如：

```
window.showModelessDialog('filename.htm',window,'dialogWidth=200px;dialogHeight=250px;')
```

接下来，就是取得父窗口的一些数据和方法，这是经常会用的，父窗口取子窗口的参数一般通过returnValue就可以搞定了

```
//取得父窗口的JS变量 var
window.dialogArguments.var;
//获得父窗口的对象和属性
window.dialogArguments.form1.name.value ;
//调用父窗口的方法 fun
window.dialogArguments.fun() ;
```

JS模态窗口返回值兼容问题解决方案

i、打开弹出窗口时把 window 作为第二个参数传入。

```
var result = window.showModalDialog(url, window, "dialogWidth=" + width +
"px;dialogHeight=" + height + "px;resizable:yes;")

if (typeof (result) == 'undefined') { //条件也可以改为：result不是设置的值时重新获取值 result =
window.ReturnValue

    result = window.ReturnValue;

}

return result;
```

ii、在弹出窗口中，执行如下JS，以接收传入的window

```
if (typeof (window.opener) == 'undefined') window.opener = window.dialogArguments;

或者：var obj = window.dialogArguments || opener;
    alert("您传递的参数为：" + obj.name)
```

iii、弹出窗口关闭前，调用如下JS赋返回值

```
window.returnValue = vals;

if (window.opener && window.opener != null){

    window.opener.ReturnValue = vals;

}



window.close();
```

4.修改窗口标题：

```
test.htm
=====
<script>
window.showModalDialog( "a.htm ",window)
</script>
```

```
a.htm
=====
<html>
<title> myOldTitle </title>
<body>
```

```
</body>
<script>
sx();
function sx()
{
    document.title= "MyNewTitle ";
}
</script>
```

分享到:  

[linux fork函数的精辟解说](#) | [HTML字符实体，转义字符串--转](#)

- 2012-02-27 17:53
- 浏览 10217
- [评论\(0\)](#)
- 分类: [Web前端](#)
- [查看更多](#)

评论

发表评论



[您还没有登录,请您登录后再发表评论](#)

相关资源推荐

[C++非模态窗口实现](#)

[模态窗口和非模态窗口](#)

[模态、非模态对话框、window.open](#)

[用javascript打开模态页面并得到返回值](#)

[js模态对话框showModalDialog与非模态对话...](#)

[JavaScript 模态窗口的使用及传值: window.sh...](#)

[模态窗口与非模态窗口](#)

[对话框1----建立模态对话框和非模态对话框](#)

[C++ 模态对话框与非模态对话框](#)

[PyQt编程之模态与非模态对话框 \(二\)](#)

[模态窗口与非模态窗口的区别](#)

[透明非模态窗口作图测试](#)

[MFC模态窗口和非模态窗口](#)

[javascript模态窗口](#)

[WPF中模态窗口和非模态窗口](#)

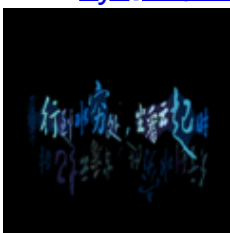
[模态与非模态对话框在实现原理上的区别](#)

[Winform窗体text模态和非模态传值](#)

[MFC--非模态对话框的创建](#)

[c#中的模态对话框和非模态对话框](#)

[模态型和非模态型子窗口的使用及区别](#)



fp_moon

- 浏览: 769266 次