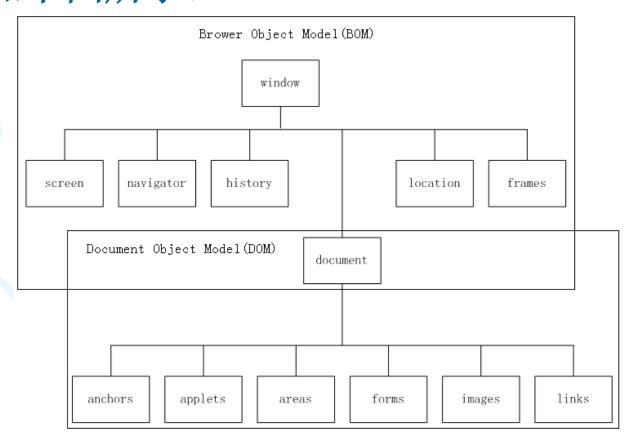
BOM对象

- 1 BOM对象概述
- 2 window对象
- 3 navigator对象
- 4 location对象
- 5 history对象
- 6 screen对象

1 BOM对象概述

• BOM(Browser Object Model),即浏览器对象模型。BOM主要用于管理窗口与窗口之间的通讯。BOM提供了独立于内容的、可以与浏览器窗口进行交互的对象结构。

• BOM由多个对象组成,其中核心对象是window对象,该对象是BOM的顶层对象,代表浏览器打开的窗口,其他对象都是该对象的子对象。BOM结构如下图所示:



2 window对象

- window 对象表示一个浏览器窗口或一个框架,用于访问除navigator 以外任何对象的顶层对象。
- window 对象是全局对象,在同一个窗口中访问其他对象时,可以省略"window"字样,但如果要跨窗口访问,则必须写上相应窗口的名称(或别名)。

1. window对象的常用属性

属性	描述
defaultStatus	设置或返回窗口状态栏的默认信息。主要针对IE,FF
uciauitstatus	和google chorme没有状态栏。
status	设置窗口状态栏的信息。主要针对IE,FF和google
Status	chorme没有状态栏。
document	引用document对象。
history	引用history对象。
location	引用location对象。
navigator	引用navigator对象。
screen	引用screen对象。
name	设置或返回窗口的名称。
opener	返回创建当前窗口的窗口。
self	返回当前窗口,等价于window对象。
top	返回最顶层窗口。
parent	返回当前窗口的父窗口。

2. window对象的常用方法

方法	描述
back()	回到历史记录中的上一网页,相当于IE浏览器的工具栏上单击"后退"按钮。
forward()	加载历史清单中的下一个网址,相当于IE浏览器工具栏上单击"前进"按钮。
blur()	使窗口失去焦点。
focus()	使窗口获得焦点。
close()	关闭窗口。
home()	进入客户端在浏览器上设置的主页
print()	打印当前窗口的内容,相当于在IE浏览器中选择【文件】— 【打印】。
alert(警告信息字符串)	显示警告对话框,用以提示用户注意某些事项
confirm(确认信息字符串)	显示确认对话框,有"确认"和"取消"两个按钮。单击确认 按钮,返回true,单击取消按钮,返回false。
prompt(提示字符串,[默认值])	显示提示输入信息对话框,返回用户输入信息。
open(URL,窗口名称,[窗口规格])	打开新窗口
setTimeout(执行程序,毫秒)	在指定的毫秒数后调用函数或计算表达式。
setInterval(执行程序,毫秒)	按照指定的周期(以毫秒计)来调用函数或计算表达式。
clearTimeout(定时器对象)	取消setTimeout设置的定时器
clearInterval(定时器对象)	取消setInterval设置的定时器

- 3. 访问window对象的属性和方法的方式
- [window(或窗口名称或别名)].属性 [window(或窗口名称或别名)].方法(参数列表) 例如:
 - ▲ Window.alert("警告对话框"); adwin.status="ok";//adwin是窗口名称
- 说明:在实际使用中,"window"也经常使用别 名代替。常用的别名有:
 - opener:表示打开当前窗口的窗口
 - parent:表示当前窗口的上一级窗口
 - -top: 表示最上方的窗口
 - -self:表示当前窗口
 - 例如: self.close();

警告对话框示例

```
<html>
〈head〉
<script>
document.title="警告对话框示例";
 function checkPassword(testObject) {
   if (test0bject. value. length<4) {</pre>
     alert ("密码长度不得小于四")|
     testObject.select();
   }else{
       alert ("密码输入正确!");
</script>
</head>
<body>
请输入密码:<input type="password" onBlur="checkPassword(this)">
</body>
```

确认对话框示例

```
<html>
<head>
<title>确认对话框示例</title>
<script>
 function isConfirm() {
   if (confirm ("你确认删除此信息吗"))
      alert ("信息已成功删除!"):
   else alert ("你取消了删除");
</script>
</head>
<body>
<form method="post" action="">
<input type="button" value="delete" onClick="isConfirm()">
</form>
</body>
/html>
```

信息提示对话框示例

```
⟨script⟩
document.title="提示信息对话框示例";

var name=prompt("请输入你的姓名","NCH");
document.write("你的姓名是: "+name);
⟨/script⟩
```

4. 打开窗口方法

• 语法:

open (URL,窗口名称[,规格参数])

• 语法解释:

- URL:该部分可以是完整的网址,表示打开该网址页面;也可以是以相对路径表示的文件名称,表示打开该文件;此外,其也可以是一个空字符串,此时将新增一个空白窗口。
- 窗口名称:这个名称可以用作标记 <a> 和 <form> 的属性 target 的值。如果该参数指定了一个已经存在的窗口,那么 open()方法就不再创建一个新窗口,而只是返回对指定窗口的引用
- <mark>规格参数:</mark>由许多由逗号隔开的参数字符串所组成,用以制定新窗口的特征,如果省略该参数,新窗口将具有所有标准特征

窗口规格参数:

- 以参数值的类型来分,窗口的规格参数可以分成两类:一类是布尔类型,以0或no来表示关闭,以1或yes来表示显示;另一类则是数值型
- 常用规格参数见下表

常用规格参数

规格参数	用法
directories=yes no 1 0	是否显示连接工具栏。默认为no。
fullscreen=yes no 1 0	是否以全屏显示。默认为no。
location=yes no 1 0	是否显示网址栏。默认为no。
menubar=yes no 1 0	是否显示菜单栏。默认为no。如果打开窗口不显示菜单栏,打 开窗口也将不显示。
resizable=yes no 1 0	是否可以改变窗口尺寸。默认为no。
scrollbars=yes no 1 0	设置如果网页内容超过窗口大小,是否显示滚动条。默认为no。
status=yes no 1 0	是否显示状态栏。默认为no。
titlebar=yes no 1 0	是否显示标题栏。默认为no。
toolbar=yes no 1 0	是否显示工具栏。默认为no。
height=number	设置窗口的高度,以像素为单位。
width=number	设置窗口的宽度,以像素为单位。
left=number	设置窗口左上角相对于显示器左上角的X坐标,以像素为单位。
top=number	设置窗口左上角相对于显示器左上角Y坐标,以像素为单位。

打开窗口示例

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>打开一个新窗口</title>
<script type="text/javascript">
   window.open("http://www.baidu.com","",
   "toolbar=yes, scrollbars=yes, height=200, width=400, resizable=yes, location=yes");
</script>
</head>
<body>
</body>
```

</html>

5.定时器的使用

- · window对象可以提供定时器的功能。定时器的作用: 在规定的时间自动执行某个函数或表达式。
- 定时器类型:
 - -setTimeout
 - -setInterval

- **setTimeout**与**setInterval**定时器的区别: 前者是在指定时间时调用一次函数或计算表达式; 后者则按照指定的周期来重复调用函数或计算表 达式,直到窗口被关闭或执行**clearInterval** 函数关闭**Interval**为止。
- ·注:利用setTimeout()递归调用,setTimeout()也可以作出与setInterval()相同的效果

1) setInterval定时器的创建和清除

• 创建语法:

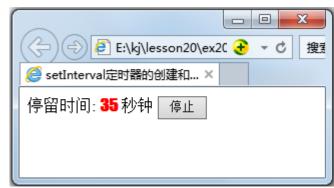
[定时器对象名称=]setInterval(表达式,毫秒)

- 语法解释:
 - 每隔由第二个参数设定的毫秒数,就执行第一个参数指定的操作
- 清除setInterval定时器:

clearInterval(定时器对象名称)

setInterval定时器示例

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>setInterval定时器的创建和清除</title>
<style>
span {
    color: red:
    font-weight:bold;
    font-family: impact;
</style>
<script>
var oSpan=null;
var sec=0;
window.onload=function() {
    oSpan=document.getElementById("num");
    timerID=setInterval("count()",1000);
function count() {
    oSpan.innerHTML=sec++;
</script>
</head>
<body>
 停留时间: <span id="num">0</span> 秒钟
 <input type=button value="停止" onClick="clearInterval(timerID)</pre>
</body>
</html>
```



2) setTimeout定时器的创建与清除

• 设置语法:

[定时器对象名称=]setTimeout(表达式,毫秒)

• 语法解释:

经过第二个参数所设定的时间后,执行一次第一个参数指定的操作。

· 清除setTimeout定时器:

clearTimeout(定时器对象名称)

setTimeout定时器示例

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>setTimeout定时器的创建和取消</title>
<style>
span {
   color:red:
   font-weight:bold;
   font-family: impact;
</style>
<script>
var oSpan=null;
var sec=0;
window.onload=function() {
   oSpan=document.getElementById("num");
   timeCount();
function timeCount() {
   oSpan.innerHTML=sec++;
    //定时递归调用timeCount()方法,同时把定时器名称赋给变量t
     timerID=setTimeout("timeCount()",1000);
</script>
</head>
<body>
 停留时间: <span id="num">0</span> 秒钟
 <input type=button value="停止" onClick="clearTimeout(timerID)</pre>
</body>
</html>
```

3 navigator对象

· navigator对象包含有关浏览器的信息。 navigator对象包含的属性描述了正在使用的 浏览器。navigator对象是window对象的 属性,因而可以使用window.navigator来 引用它,实际使用时一般省略"window"。

• 常用属性如下表所示:

属性	描述
appCodeName	返回浏览器的代码名。
appMinorVersion	返回浏览器的次级版本。
appName	返回浏览器的名称。
appVersion	返回浏览器的平台和版本信息。
browserLanguage	返回当前浏览器的语言。
cookieEnabled	返回指明浏览器中是否启用cookie,如果启用则返回true,否则返回false。
platform	返回运行浏览器的操作系统平台。
systemLanguage	返回OS使用的默认语言。
userAgent	返回由客户机发送服务器的user-agent头部的值。

• 常用方法:

javaEnabled():用于判断是否支持JAVA preference():用于取得浏览器的爱好设置

· 访问navigator对象属性和方法的格式:
navigator.属性
navigator.方法(参数列表)

navigator对象应用示例

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>navigator对象的使用</title>
<script>
if (navigator.userAgent.toLowerCase().indexOf("trident") > -1){
    alert ('你使用的是IE'+', 浏览器的cookie启用了吗?'+navigator.cookieEnabled);
}else if(navigator.userAgent.indexOf('Firefox') >= 0){
    alert('你使用的是Firefox'+',浏览器的cookie启用了吗?'+navigator.cookieEnabled);
}else if(navigator.userAgent.indexOf('Opera') >= 0){
    alert('你使用的是Opera'+', 浏览器的cookie启用了吗?'+navigator.cookieEnabled);
}else if(navigator.userAgent.indexOf("Safari")>0){
    alert('你使用的是Safari'+', 浏览器的cookie启用了吗?'+navigator.cookieEnabled);
}else{
    alert ('你使用的是其他的浏览器浏览网页!');
</script>
</head>
<body>
</body>
</html>
```

4 location对象

· location对象包含了浏览器当前显示的文档的 URL的信息。当location对象调用href属性设置URL时,可使浏览器重定向到该URL。 location对象是window对象的一个对象类型的属性,因而可以使用window.location来引用它,使用时也可以省略"window"。

· 完整的URL组成结构:



• 常用属性如下表所示:

属性	描述
hash	设置或返回从井号(#)开始的URL(锚)
host	设置或返回主机名和当前URL的端口号
hostname	设置或返回当前URL的主机名
href	设置或返回完整的URL
pathname	设置或返回当前URL的路径部分
port	设置或返回当前URL的端口号
protocol	设置或返回当前URL的协议
search	设置或返回从问号(?)开始的URL(查询部分)

• 常用方法:

方法	描述
assign()	加载新的文档。
reload()	重新加载当前文档。
replace()	用新的文档替换当前文档,且无须为它创建一个新的历史记录。

· 访问location对象属性和方法的格式:

location.属性 location.方法(参数列表)

location对象示例

- - X

A → | A ☆

```
<!doctype html>
< ht.ml >
<head>
<meta charset="utf-8">
<title>location对象的使用</title>
<style>
                                          🔷 🞒 E:\kj\lesson20\ex20 🚱 🔻 🖒 搜索...
span {
                                        遵 location对象的使用
    color:red:
                                        五秒钟后自动带你前往:http://www.sise.com.cn/
    font-weight:bold;
                                        倒计时: 3秒钟
</style>
<script>
var sec=5;
var oSpan=null;
function countDown() {
    oSpan=document.getElementById("num");
    if(sec>0) {
       oSpan.innerHTML=sec--;
    else location.href="http://www.sise.com.cn";
</script>
<body onLoad="setInterval('countDown()',1000)">
<h2>五秒钟后自动带你前往:http://www.sise.com.cn/</h2>
<h2>倒计时: <span id="num">5</span>秒钟</h2>
</body>
</html>
```

5 history对象

- history对象包含用户(在浏览器窗口中)访问 过的URL。history对象是window对象的一 个对象类型的属性,可通过window.history 属性对其进行访问,使用时也可以省略 "window"。
- 常用属性:
 - -length: 返回浏览器历史列表中的URL数量

• 常用方法:

方法	描述
back()	加载history列表中的前一个URL。
forward()	加载history列表中的下一个URL。
go(number)	加载history列表中的某个具体页面。参数number是要访问的 URL在history的URL列表中的相对位置,可取正数可负数。在当前页面前面的URL的位置为负数(如在前一个页面的位置为-1),反之则为正数。

· 访问history对象的属性和方法的方式:

history.属性 history.方法(参数1,参数2,...)

histroty对象示例

history.back();//等效单击"后退"按钮 history.forward();//等效单击"前进"按钮 history.go(-1);//等效单击一次后退按钮 history.go(-2);//等效单击两次后退按钮

6 screen对象

·screen对象包含有关客户端显示屏幕的信息。 JavaScript 程序可以利用这些信息来优化输 出,以达到用户的显示要求。Screen对象是 window对象的一个对象类型的属性,可通过 window.screen属性对其进行访问,使用时 也可以省略"window"。

• 常用属性:

属性	描述
availHeight	返回显示屏幕的可用高度,单位为像素,不包括任务栏。
availWidth	返回显示屏幕的可用宽度,单位为像素,不包括任务栏。
height	返回显示屏幕的高度,单位为像素。
width	返回显示屏幕的宽度,单位为像素。
colorDepth	返回当前颜色设置所用的位数 - 1: 黑白; 8: 256色; 16: 增强色; 24/32: 真彩色。

· 访问screen对象属性的方式: screen.属性

screen对象示例

screen.availHeight;//获取屏幕的可用高度 screen.availWidth;//获取屏幕的可用宽度 screen.height;//获取屏幕的高度 screen.width;//获取屏幕的宽度