实验十一 网站的动态数据开发技术Javascript对象

实验目的：

 了解Java对象的构成，掌握其基本用法

实验内容：

1.

**【例3-10】“书”对象的定义。**

**function print( )**

**{//方法成员定义，输出各属性成员值**

**document.write("书名为"+this.name+"<br>");**

**document.write("作者为"+this.author+"<br>");**

**document.write("出版社为"+this.publisher+"<br>");**

**document.write("出版时间为"+this.date+"<br>");**

**document.write("印数为"+this.num+"<br>");**

**}**

**function book(name,author,publisher,date,num)**

**{//构造函数**

**this.name=name; //书名，属性成员**

**this.author=author; //作者，属性成员**

**this.publisher=publisher;**

**//出版社，属性成员**

**this.date=date; //出版时间，属性成员**

**this.num=num; //印数，属性成员**

**this.print=print; //方法成员**

**}**

**var book1=new book("语文","集体编","人民教育出版社","1999",10000);**

知识点：

对象的定义：先定义成员方法，后定义对象；

对象的引用

2.

**【例3-11】下列函数Show显示其参数对象各属性的值。**

**function Show(obj) //定义通用函数Show**

**{ var prop;**

**for (prop in obj)**

**document.write(obj[prop]+" ");**

**document.write("<br>");**

**}**

知识点：

For…in…和with语句及其在对象中的应用

3.

###### 【例5-8】下列函数Show显示其参数对象各属性的值，它可作为一个通用函数使用。

<html><body><script language= "JavaScript">

function person(name,age) //定义对象person

{ this.name=name;

this.age=age;

}

function book(title,author,publisher,price) //定义对象book

{ this.title=title;

this.author=author;

this.publisher=publisher;

this.price=price;

}

function Show(obj) //定义通用函数Show

{ var prop;

for (prop in obj)

document.write(obj[prop]+" ");

document.write("<br>");

}

var obj1=new person("Mary",20);

var obj2=new book("语文","集体编","人民教育出版社",5.5);

Show(obj1);

Show(obj2);

</script></body></html>

知识点：

function Show(obj)中，为什么不直接写obj的成员名，而是用for语句？

4.

###### 【例5-9】一个Array对象的应用示例。

<html><head><title>数组对象</title>

<script language="JavaScript">

function updateInfo(WhichBook) //对象book的方法成员，修改对象属性值

{ document.BookForm.currbook.value=WhichBook;

 document.BookForm.BookTitle.value=this.Title;

  document.BookForm.BookPublisher.value=this.Publisher;

 document.BookForm.BookAmount.value=this.Amount;

}

function Book(title,publisher,amount) //对象book的构造函数

{ this.Title=title;

 this.Publisher=publisher;

 this.Amount=amount;

 this.UpdateInfo=updateInfo;

}

</script></head>

<body><script language="JavaScript">

var Books=new Array(); //创建数组，数组元素是book对象

//为数组各元素赋值

Books[0]=new Book("语文","少年儿童出版社",10000);

Books[1]=new Book("数学","高等教育出版社",5000);

Books[2]=new Book("普通物理","高等教育出版社",3000);

Books[3]=new Book("计算机基础","清华大学出版社",2000);

</script>

<h2 align=center>共有四本书，可选择查看其信息</h2>

<form name="BookForm">

选择当前所显示的书：&nbsp;&nbsp;

<input type=button value=A书 onClick="Books[0].UpdateInfo('A书')">

<input type=button value=B书 onClick="Books[1].UpdateInfo('B书')">

<input type=button value=C书 onClick="Books[2].UpdateInfo('C书')">

<input type=button value=D书 onClick="Books[3].UpdateInfo('D书')"><br><br>

当前书：<input type="text" name="currbook" value="A书"><br><br>

书名：<input type="text" name="BookTitle" value="语文"><br><br>

出版社：<input type="text" name="BookPublisher" value="少年儿童出版社"><br><br>

印数：<input type="text" name="BookAmount" value="10000"></form></body></html>

5.

###### 【例5-10】一个有关Date对象的应用例子。该HTML文件在浏览器窗口显示一个不断刷新的数字时钟。

<html><head><title>数字钟</title>

<style>

form { font-size:22px; }

input { font-size:24px;

color:red;

width:180;height:40;}

</style>

<script language="JavaScript">

function aClock( ){

var now=new Date( );

var hour=now.getHours( );

var min=now.getMinutes( );

var sec=now.getSeconds( );

var timeStr=" "+hour;

timeStr+=((min<10)?":0":":")+min;

timeStr+=((sec<10)?":0":":")+sec;

timeStr+=(hour>=12)?" P.M.":" A.M.";

document.clock\_form.clock\_text.value=timeStr;

clockId=setTimeout("aClock( )",1000);

}

</script></head>

<body onLoad="aClock( )">

<br><br><br>

<form name="clock\_form">

&nbsp;&nbsp;当前时间是：

<input type="text" name="clock\_text" value="">

</form></body></html>

6.

###### 【例5-11】根据浏览器的类型显示不同的页面，在Chrome和IE中的结果如图5-11所示。

<html><head><title>Navigator对象</title></head>

<body><center>

<font face="隶书" color=red size=6>欢迎您来访</font></center>

<script language="JavaScript">

if (navigator.appName=="Netscape")

document.write ('<hr width=100%>');

else { document.write('<font face="隶书" color=darkgreen size=4>');

document.write('<marquee border="0">您好！欢迎您来到我的主页

</marquee></font>');

}

document.write("<br><font size=4 color=blue>您使用的浏览器是:<br>");

document.write(navigator.userAgent);

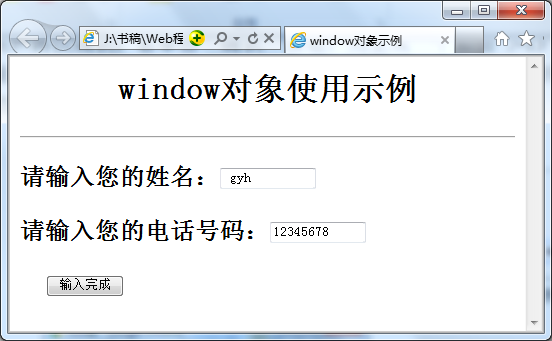
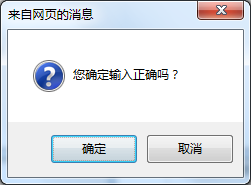
document.write(navigator.appName);

document.write("</font>");

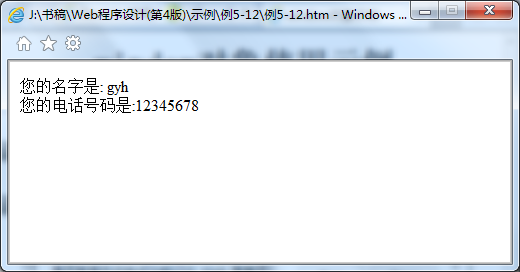
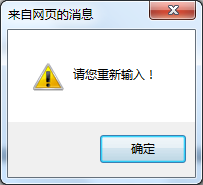
</script></body></html>

7.

###### 【例5-12】在浏览器中显示一个如图5-13（a）所示的初始用户输入界面，接收用户输入的姓名和电话号码，用户单击“输入完成”按钮后，将弹出如图5-13（b）所示的对话框，要求用户再次确认，若单击“确认”按钮，则生成一个如图5-13（c）所示的新窗口，显示用户输入的姓名和电话号码。若单击图5-13（b）中的“取消”按钮，则弹出如图5-13（d）所示的警告框。

(a) 初始用户输入界面 (b) 单击“输入完成”按钮后的对话框

(c) 单击“确定”按钮生成的新窗口 (d) 单击“取消”按钮弹出的对话框

图5-13 Window对象使用示例

<html><head><title>window对象示例</title>

<script language= "JavaScript">

function confSubmit( ){

if ((ok=confirm("您确定输入正确吗？"))==true){

var nw=open("a.htm","nwin","width=500,height=200,toolbar=1");

nw.focus( );

nw.document.write("您的名字是:"+parent.document.input\_form.nm.value);

nw.document.write("<br>");

nw.document.write("您的电话号码是:" + parent.document.input\_form.phone.value);

}

else

alert("请您重新输入！");

}

</script></head>

<body>

<script>defaultStatus="这是一个window对象使用示例。";</script>

<center><h1>window对象使用示例一</h1></center><hr><br>

<form name="input\_form"><h2>

请输入您的姓名：<input type=text name="nm" size=12 value=" "><br><br>

请输入您的电话号码：<input type=text name="phone" size=12><br><br>

&nbsp;&nbsp;<input type=submit value="确定" onClick="confSubmit( )"></h2>

</form></body></html>

7.

###### 【例5-18】Form对象属性的引用方法示例。设计如图5-18所示的页面，放置两个表单，包含文本框和按钮，通过Form对象的属性和方法，在页面中显示表单元素。

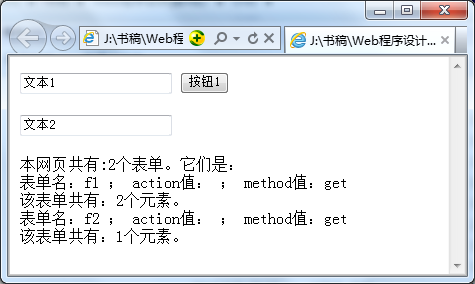


图5-18 Form对象属性引用方法示例

<html><body><form name="f1">

<input type="text" name="t1" value="文本1">

<input type="button" value="按钮1"></form>

<form name="f2"><input type="text" name="t2" value="文本2"></form>

<script language="JavaScript">

document.write("本网页共有:"+document.forms.length+"个表单。它们是：<br>");

for (var i=0;i<document.forms.length;i++){

document.write("表单名："+document.forms[i].name+" ； ");

document.write("action值："+document.forms[i].action+" ； ");

document.write("method值："+document.forms[i].method+"<br>");

document.write("该表单共有："+document.forms[i].length+"个元素。<br>");

}

</script></body></html>

###### 【例5-19】 用户输入数据合法性检查示例。设计如图5-19所示的页面，首先显示用户输入电话号码的界面，当用户输入了电话号码并单击“提交”按钮后，将执行合法性检查程序。若用户输入的电话号码值不符合要求（必须是数字，且数字位数为11位或8位），则提示出错信息，并且不提交服务器；否则将用户输入的电话号码数据提交给服务器。

<html><head><script language="JavaScript">

function Verify( ){

var Tel=document.TelForm.TelNo.value;

if ((Tel.length==8) | (Tel.length==11)){

if (parseInt(Tel)!=0){

NewWin=window.open("","","width=200,height=200");

NewWin.document.open("text/html");

NewWin.document.write("<H3>号码已经成功提交！</H3>");

NewWin.document.close( );

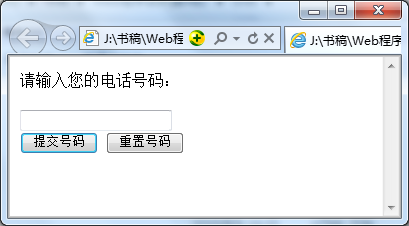


图5-19 数据合法性检查示例

}

else{

alert("号码输入不正确！");

return false;

}

}

else{

alert("号码输入不正确！");

return false;

}

}

</script></head>

<body>请输入您的电话号码：<br>

<form name="TelForm" onSubmit="Verify( )">

<input type="text" name="TelNo" value=""><br>

<input type="submit" value="提交号码" width=100>

<input type="reset" value="重置号码" width=100></form>

</body></html>

扩展：

【例5-13】 通过JavaScript设置页面的颜色和文字等属性。

<html><head><title>通过Document对象设置页面属性</title></head>

<body><script language= "JavaScript">

document.bgColor="#DDEEFF";

document.fgColor="darkred";

document.linkColor="#0088FF";

document.alinkColor="#0088FF";

document.vlinkColor="#0088FF";

document.write("<h1>通过Document对象设置页面属性示例</h1>");

document.write("<hr>");

document.write("<a href='a.htm'>去页面A</a>");

document.write("<br><br>本HTML文件名是："+document.URL);

document.write("<br><br>本HTML文件最后被修改的时间是："+document.lastModified);

</script></body></html>

###### 【例5-14】 设计如图5-15所示的框架网页。窗口划分为左、右两个框架，文件anchor1.htm显示于左框架中，它包含一系列按钮，每个按钮对应右框架中显示的页面锚点；文件anchor2.htm显示于右框架中，它定义了被命名为“0”，“1”，“2”和“3”的4个锚点。当单击左框架中的某按钮时，右框架中显示被定位到的指定目标。主文件名为anch.htm。

文件anch.htm的内容：

<html><head><title>anchors数组示例</title></head>

<frameset cols="30%,\*">

<frame src="anchor1.htm">

<frame src="anchor2.htm" name="anchors2">

</frameset></html>

文件anchor1.htm的内容：

<html><head><title>anchors:frame 1</title></head>

<body><script language= "JavaScript">

function linkToAnchor(num){

if (parent.anchors2.document.anchors.length>=num)

parent.anchors2.location.hash=num

else

alert("目标不存在！");

}

</script>

单击以下按钮可跳转到需要的目标：<br><br>

<form><input type="button" value="目标0"

name="anch0" onclick="linkToAnchor(0)"><br>

<input type="button" value="目标1" name="anch1" onclick="linkToAnchor(1)"><br>

<input type="button" value="目标2" name="anch2" onclick="linkToAnchor(2)"><br>

<input type="button" value="目标3" name="anch3" onclick="linkToAnchor(3)">

</form></body></html>

文件anchor2.htm：

<html><head><title>anchors:frame 2</title></head>

<body><p><a name="0"><b>Some persons</b></a>

<ol><li>Li Ming<li>Wang Lin<li>Zhang Shan<li>Zhu Yun<li>Ma Li<li>Lei Feng

</ol></p>

<p><a name="1"><b>Some countries</b></a>

<ul><li>China<li>Canada<li>U.S.A<li>Japan<li>Brazil</ul></p>

<p><a name="2"><b>Some colors</b></a>

<ul><li>red<li>bule<li>green<li>yellow<li>gray</ul></p>

<p><a name="3"><b>Some courses</b></a><ol><li>Data Structure and Programming

<li>Operating System<li>Computer Networks<li>Artificial Intelligence</ol></p>

</body></html>

###### 【例5-15】实现一个简单的小动画，交替显示三幅图像。

<html><head><script language="JavaScript">

var ImageNum=1;

function Begin( ){

document.MyImage.src=ImageArray[ImageNum].src;

ImageNum++;

if (ImageNum>3)

ImageNum=1;

}

</script></head>

<body>

<img name="MyImage" src="images/alvbull1.gif" onLoad= "setTimeout('Begin( )',100)">

<script language="JavaScript">

var ImageArray=new Array( );

for (i=1;i<=3;i++)

{ ImageArray[i]=new Image( );

ImageArray[i].src="images/alvbull"+i+".gif";

}

</script></body></html>

###### 【例5-20】Form子对象的用法示例。设计如图5-20所示的页面，填写和选择个人信息，由系统进行检查和显示。

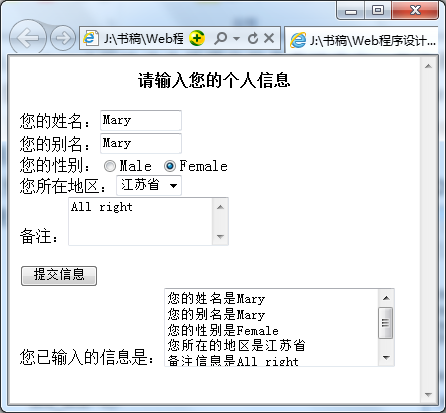


图5-20 Form子对象的用法示例

<html><head><script language="JavaScript">

sex=new Array( );

sex[0]="Male";

sex[1]="Female";

sele=0;

sex\_sele=0;

function VerifyAndChgText( ){

var Length=document.forms[0].length;

var Type,Empty=false;

for (var i=0;i<Length;i++){

Type=document.forms[0].elements[i].type;

if (Type=="text")

if (document.forms[0].elements[i].value=="")

Empty=true;

}

if (!Empty ){

name="您的姓名是"+document.forms[0].NameText.value+"\n";

alias="您的别名是"+document.forms[0].AliasText.value+"\n";

sex\_1="您的性别是"+sex[sex\_sele]+"\n";

area="您所在的地区是"+document.forms[0].area.options[sele].text+"\n";

exp="备注信息是"+document.forms[0].exp.value+"\n";

document.forms[0].info.value=name+alias+sex\_1+area+exp;

}

else

alert("您未输入完全！");

return Empty;

}

</script></head>

<body><h4 align=center>请输入您的个人信息</h4><form>

您的姓名：<input type=text name="NameText" size=10><br>

您的别名：<input type=text name="AliasText" size=10><br>

您的性别：<input type="radio" name="sex" onClick="sex\_sele=0">Male

<input type="radio" name="sex" onClick="sex\_sele=1">Female<br>

您所在地区：<select name="area" onChange="sele=this.selectedIndex">

<option value="1" selected>江苏省

<option value="2" >北京市

<option value="3" >上海市

<option value="4">天津市

<option value="5">浙江省

</select><br>

备注：<textarea name="exp" rows=3 cols=20></textarea><br><br>

<input type=button value="提交信息" onClick="VerifyAndChgText( )">

<br>您已输入的信息是：<textarea name="info" rows=5 cols=30></textarea><br><br>

</form></body></html>

###### 【例5-21】在HTML文件中实现页面的前进和后退。

<html><head><title>History对象示例</title></head>

<body><center><h2>History对象使用示例</h2></center><hr><br>

<form><input type=button value="往前翻 " onClick="history.go(-1)">

<input type=button value="重载当前页" onClick="history.go(0)">

<input type=button value="往后翻 " onClick="history.go(1)"></form>

</body></html>

【例5-22】 框架之间的通信示例。

① 文件frame通信.htm，定义框架结构。

<html>

<frameset rows="15%,\*">

<frame src="frame通信1.htm" name="Win1">

<frame src="frame通信2.htm" name="Win2">

</frameset>

</html>

② 文件frame通信1.htm，是主要文件。

<html><head>

<script language="JavaScript">

var Questions=new Array("您认为LAN指什么?","您知道TCP/IP吗?","您对计算机网络的认识?");

var CurrentQuestion=0;

var Answers=new Array( );

function PutQuestions( ){ //在Win2中输出问题并创建表单，接收用户输入的答案

with (parent.Win2.document){

open( );

write("<html><body>");

write("第",CurrentQuestion+1,"个问题: <font color=red>");

write(Questions[CurrentQuestion]+"</font><br>");

write("<form name=\'QuestionForm\'>");

write("<textarea name=\'AnswerText\' cols=30 rows=5></textarea><br>");

write("<input type=button value=\'提交答案\' onClick=\'parent.Win1.NextQ( )\'>");

write("</form></body></html>");

close( );

}

}

function PrintAnswers( ){ //在Win2中输出问题和用户输入的答案

with (parent.Win2.document){

open( );

write("<html><body>");

for (var i=0;i<Questions.length;i++){

write("问题",i+1,":",Questions[i]);

write("答案：",Answers[i],"<br>");

}

write("</body></html>");

}

}

function NextQ( ){ //若问题未处理完，则调用PutQuestions，否则，调用PrintAnswers

if (CurrentQuestion<Questions.length){

Answers[CurrentQuestion]=parent.Win2.document.QuestionForm.AnswerText.value;

CurrentQuestion++;

if (CurrentQuestion<Questions.length) PutQuestions( );

else { Answers[CurrentQuestion]=

parent.Win2.document.QuestionForm.AnswerText.value;

PrintAnswers( );

}}

}

</script></head>

<body onLoad="setTimeout('PutQuestions( )',1000)">

<h2 align=center>请您回答以下问题</h2></body></html>

③ 文件frame通信2.htm，可以为空，但为了浏览器的识别，写入了一些内容。

<html><head><title>文件frame通信2.htm </title></head>

<body>This is a frame</body></html>

###### 【例5-23】设计如图5-23所示的用户注册页面。在单击“发送”按钮后，对输入框中输入的数据进行合法性检查。

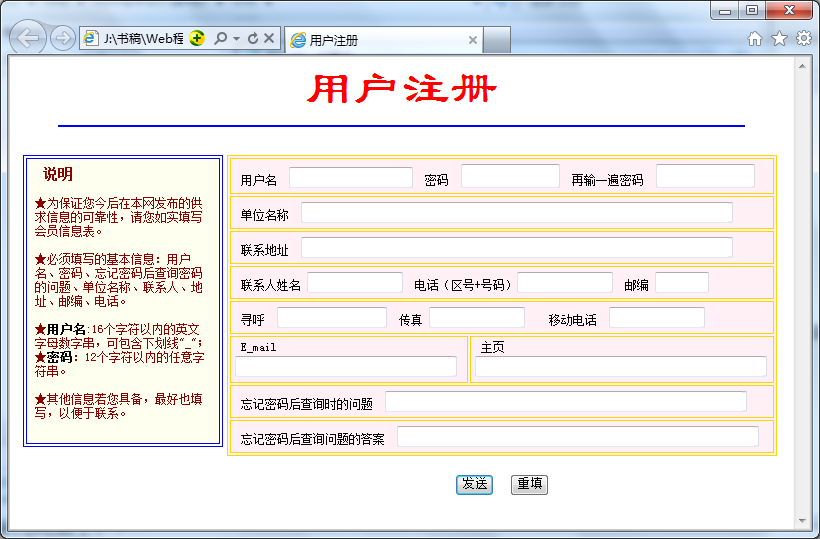


图5-23 利用JavaScript进行客户端输入数据检查

源代码如下：

<html>

<script language="JavaScript">

function init( )

{ document.reg\_form.usrname.focus( ); //初始将光标定位在用户名输入框

}

function Verify( ) //校验用户输入

{ if (VerifyUsrName( )==false) return false; //校验用户名

if (VerifyPasswd( )==false) return false; //校验密码

if (VerifyDepart( )==false) return false; //校验单位名称

if (VerifyAddr( )==false) return false; //校验地址

if (VerifyPersonName( )==false) return false; //校验联系人姓名

if (VerifyPhone( )==false) return false; //校验电话号码

if (VerifyZip( )==false) return false; //校验邮编

if (VerifyBp( )==false) return false; //校验Bp号码

if (VerifyFax( )==false) return false; //校验传真号

if (VerifyHand( )==false) return false; //校验手机号

if (VerifyEmail( )==false) return false; //校验电子邮件地址

if (VerifyHomepage( )==false) return false; //校验主页地址

if (VerifyQuest( )==false) return false; //校验忘记密码时所提问题

if (VerifyAnsw( )==false) return false; //校验问题答案

return true;

}

function VerifyUsrName( )

{ if (document.reg\_form.usrname.value.length==0)

{ alert(" 用户名不能为空!请见左边的说明，输入合法的用户名。");

return false;

}

if (validOfUsrName( )==false)

{ alert("您输入的用户名中包含了不合法的字符!请见左边的说明，重新输入。");

return false;

}

return true;

}

function validOfUsrName( ) //检验用户输入的用户名中是否包含非法字符

{ valid=true;

for (var i=0;i<document.reg\_form.usrname.value.length;i++)

{ var ch=document.reg\_form.usrname.value.charAt(i);

if (!((ch>="0") && (ch<="9")) && !((ch>="a") && (ch<="z")) && !((ch>="A") && (ch<="Z")) && !(ch=="\_"))

valid=false;

if (!valid)

break;

}

return valid;

}

function VerifyPasswd( )

{ if (document.reg\_form.pass.value.length==0)

{ alert(" 密码不能为空!请见左边的说明，输入您的密码。");

return false;

}

if (document.reg\_form.pass.value!=document.reg\_form.pass2.value)

{ alert("您两次输入的密码不相同!请重新输入密码。");

return false;

}

return true;

}

function VerifyDepart( )

{ if (document.reg\_form.dname.value.length==0)

{ alert("单位名称不能为空!请见左边的说明进行输入。");

return false;

}

return true;

}

//说明：校验联系人姓名、地址、邮编等与校验单位名称的正确性的程序相似，读者可自行补齐，其代码略去。

</script>

<body topmargin=6>

<center><font face="隶书" size=7 color=red>用户注册</font></center>

<hr width=90% color=blue><br>

<table border=0 aligen=center onLoad="init( )">

<tr><td valign=top><!—对用户填写表单数据的说明-->

<table border=1 align=left bordercolor=blue bgcolor=ivory width=200

style="fontsize:14px;align:left;" cellpadding=8><tr><td>

<div style="color:darkred;font-size:12px;text-align:left;font-family:'宋体'">

<span style="color:darkred;font-size:14px"><b>&nbsp;说明</b></span><br><br>

<div>★为保证您今后在本网发布的供求信息的可靠性，请您如实填写会员信息表。

</div><br>

<div>★必须填写的基本信息：用户名、密码、忘记密码后查询密码

的问题、单位名称、联系人、地址、邮编、电话。</div><br>

<div>★<font color=black><b>用户名</b></font>:16个字符以内的英

文字母数字串，可包含下划线"\_"；</br>

★<font color=black><b>密码</b></font>：12个字符以内的任意字符串。</div><br>

<div>★其他信息若您具备，最好也填写，以便于联系。</div><br>

</div></td></tr></table></td>

<td valign=top align=center><!—表单输入区定义-->

<form action="reg\_handle.asp" method="get" name="reg\_form"

onSubmit="return Verify( )">

<table border=1 bordercolor=gold bgcolor=lavenderblush width=550

style="font-size:12px;align:left;" cellpadding=4>

<tr><td width=550 ailgn=left colspan=2 >

&nbsp;用户名&nbsp; <input type=text size=16 name="usrname">

&nbsp;密码&nbsp; <input type=password size=12 name="pass">

&nbsp;再输一遍密码&nbsp; <input type=password size=12 name="pass2">

</td></tr>

<tr><td width=500 align=left colspan=2 >

&nbsp;单位名称 &nbsp;<input type=text size=60 name="dname"></td></tr>

<tr><td width=550 colspan=2 align=left>

&nbsp;联系地址 &nbsp;<input type=text size=60 name="address"></td></tr>

<tr><td width=550 align=left colspan=2>

&nbsp;联系人姓名&nbsp;<input type=text size=12 name="person\_name">

&nbsp;电话（区号+号码）<input type=text size=12 name="tel">

&nbsp;邮编 <input type=text size=6 name="postcode"></td></tr>

<tr><td width=500 align=left colspan=2>

&nbsp;寻呼&nbsp; <input type=text size=14 name="bpcode">

&nbsp;传真&nbsp;<input type=text size=12 name="fax">

&nbsp;&nbsp; 移动电话&nbsp; <input type=text size=12 name="hand"></td></tr>

<tr><td width=250 ailgn=left>

&nbsp;E\_mail <input type=text size=30 name="email"></td>

<td width=300 align=left>&nbsp;主页<input type=text size=40 name="homepg">

</td></tr>

<tr><td width=550 colspan=2 align=left>

&nbsp;忘记密码后查询时的问题 &nbsp;<input type=text size=50 name="quest">

</td></tr>

<tr><td width=550 colspan=2 align=left>

&nbsp;忘记密码后查询问题的答案 &nbsp;<input type=text size=50 name="answ">

</td></tr></table><br>

<input type=submit value="发送">&nbsp;&nbsp;<input type=reset value="重填">

</form></td></tr></table></body></html>

###### 【例5-24】读者一定很熟悉用扑克牌计算24点的游戏吧，下面就用JavaScript来设计一个在浏览器中计算24点的游戏程序。游戏开始后，系统随机给出4个1~9之间的整数，用户给出这4个数的运算式，运算式中可包括加（+）、减（-）、乘（\*）、除（/）运算符；若用户给出的算式的值为24，则系统弹出一个对话框告之结果正确，否则告之结果错误。

本例共设计如下5个文件：

① game.htm——“开始新游戏”界面，显示游戏规则，提供一个“新游戏”功能按钮，用户单击该按钮即可生成一个新窗口，在该窗口中加载文件poker.htm文件，开始一次新游戏的过程。

② poker.htm——主文件，它首先产生4个1~9之间的随机数，并将输入焦点定于结果算式输入框，然后等待用户输入；用户输入的时间限制为1 min（60 000 ms），每隔1 ms poker.htm在状态栏提示用户所剩余的时间；若用户在规定的时间内在文本框中输入算式并单击“提交答案”按钮，poker.htm将检查该答案，若计算正确，则弹出新窗口，其中加载文件yes.htm，提示用户答案正确；若计算错误或算式不合法，也弹出新窗口，在其中加载文件no.htm，提示用户答错了；若计时时间满了，而用户还未输入算式，也将弹出“超时”窗口，在其中加载文件timeout.htm，提示用户超时，同时结束本次游戏。在该文件的init( )函数中，利用Math对象的数学函数random( )产生一个0~1之间的随机数；函数ceil(number)将返回一个大于或等于number的最小整数，因此表达式Math.ceil(Math.random( )\*9)将返回一个1～9随机之间的整数。该函数还利用setTimeout( )方法设置用户答题时间限制和状态栏提示信息的更新时间。文件是其他函数的功能以及HTML语句的作用在下面的源文件中都给出了详细注解，在此不再赘述。

③ yes.htm——当用户输入正确答案后弹出的提示窗口中要加载的文件。

④ no.htm——当用户输入错误答案后弹出的提示窗口中要加载的文件。

⑤ timeout.htm——超时后弹出的提示窗口中要加载的文件。

各文件的内容如下。

（1）文件game.htm

<html><head><title>扑克牌游戏</title>

<style type="text/css">

body {margin-top:16;}

h3{ background-color:#ffddcc;color:darkred;width:16%;height:60px;

vertical-align:50%;}

</style></head><body><h3><br>&nbsp;&nbsp;游戏规则：</h3><br>

<p><h4>规则1. 要进行新游戏，请按“新游戏”按钮，此时将生成一个新窗口，其中给出4个1~9之间的整数。

<br>规则2. 在新窗口的输入框输入四则运算式，可以使用括号，运算数为给出的4个值，每个数只能使用1次，结果需等于24。

<br>规则3. 若您认为给出的4个数的运算结果不可能为24，则可按“放弃这局”按钮。

</h4></p>

<form>

<input type="button" value="新游戏"

onClick="open('poker.htm','newWin','width=500,height=400,status=1')">

</form></body></html>

（2）文件poker.htm

<html><head>

<script language="JavaScript">

//全局变量定义

var card = new Array( ); //存储4张牌

for (k=0;k<4;k++)

card[k]=0; //赋初值

var cardUsed = new Array( ); //每张牌是否被使用过的标记

for (k=0;k<4;k++)

cardUsed[k] = false; //赋初值

var TimeID,StatID; //时间和状态标记

var count = 60; //计时器

function Init( ) { //初始化函数，生成4个随机数，并设置时间和状态标记

var i;

status = "您有1min的时间考虑与输入答案！";

for (i=0;i<4;i++) //用随机函数产生4张牌

card[i] = Math.ceil(Math.random( )\*9);

StatID = setTimeout("ChangeStatus( )",1000);

TimeID = setTimeout("open('timeout.htm','timeoutWin',

'width=200,height=100');close( )",60000);

} //End of Init

//状态栏刷新函数定义

function ChangeStatus( ) { //每隔1s刷新1次状态栏显示

clearTimeout(StatID); //清除状态标记

count--; //剩余时间减少1s

status = "剩余时间为："+count+"s";

setTimeout("ChangeStatus( )",1000); //每隔1s调用1次ChangeStatus

}

//输入合法性判断函数，若算式合法，返回true，否则返回false

function IsValid( ) { //判定用户输入的算式是否正确

var exp = document.expForm.expText.value; //取用户输入的算式

var expLen = exp.length; //算式长度

var i,j;

var numberUsed = 0; //算式中使用的运算数的个数

for (i=0;i<expLen-1;i++){

var ch = exp.charAt(i); //取第i个字符

if (ch>='0' && ch<='9') { //当前处理的是数值字符

for (j=0;j<4;j++)

if ((ch == card[j]) && (!cardUsed[j])) {

//该数字是否是给出的4个数之一且未被使用过

numberUsed++;

cardUsed[j] = true; //置数字已被使用过标记

}

}

else { //当前处理的是运算符

if ((ch!="+") && (ch!="-") && (ch!="\*") && (ch!="/") && (ch!="(")

&& (ch!=")"))

{ alert("您输入的算式是非法的!");

return false;

}

}

}

if (numberUsed!=4) { //算式中未使用全部4个数字

alert("您输入的算式是非法的!");

return false;

}

return true;

}

function calResult( ) { //计算算式结果函数

clearTimeout(TimeID); //清除计时标记

if (IsValid( )) { //算式合法

if (eval(document.expForm.expText.value)==24) //若算式的值等于24

{ //弹出新窗口，提示结果正确

winid = open("yes.htm","nwin1","width=200,height=100");

close( );

return;

}

}

//算式的值不等于24或算式不合法，弹出新窗口，提示结果错误

winid = open("no.htm","nwin2","width=200,height=100");

close( );

return;

}

</script></head>

<!—将输入焦点设置在答案输入框中-->

<body onLoad="document.expForm.expText.focus()">

<script language="JavaScript">

Init( ); //生成四张牌

document.write("您可以使用的四张牌是：<br>");

for (i=0;i<4;i++)

document.write(card[i]+" ");

</script>

<br><form name="expForm">请在右边的输入框中输入您的答案：&nbsp;&nbsp;

<input type=text name="expText" size=12 value=" "><br><br><br>&nbsp;&nbsp;

<input type=button value="提交答案" onClick="calResult( )">

<input type=button value="放弃该局" onClick="self.close( )"></form></body></html>

（3）文件yes.htm

<html>

<body onLoad="setTimeout('close()',60000);return true">恭喜您，您答对了！</body>

</html>

（4）文件no.htm

<html>

<body onLoad="setTimeout('close()',60000);return true">对不起，您答错了！</body>

</html>

（5）文件timeout.htm

<html>

<body onLoad="setTimeout('close()',60000);return true">

对不起，您超时了！请重新开始。</body></html>

###### 【例5-25】产生表格的HTML文件示例。DOM将该文件作为如图5-25所示的树形结构。

<table>

<tbody>

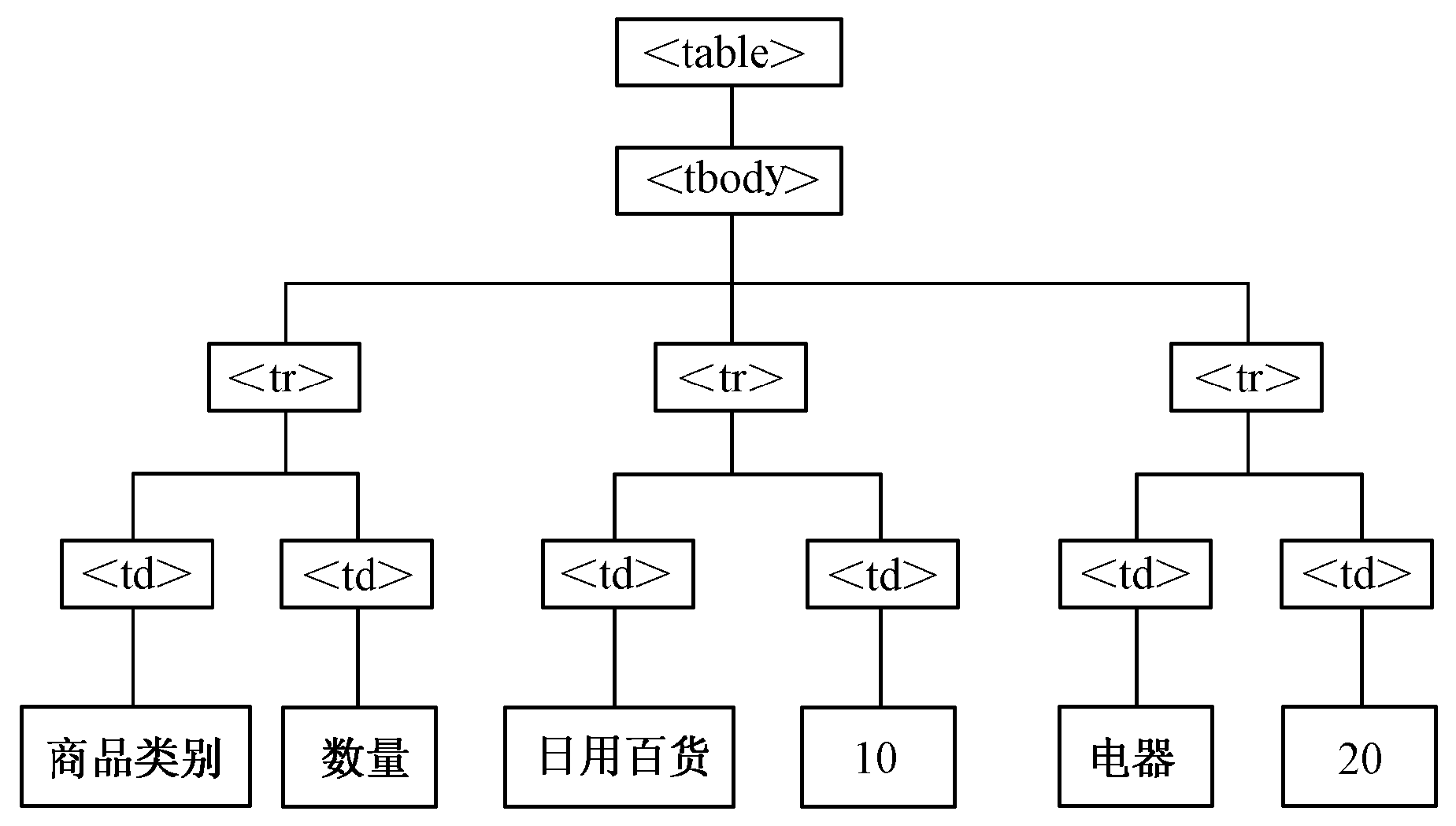


图5-25 例5-25对应的DOM树形结构

<tr>

<td>商品类别</td>

<td>数量</td>

</tr>

<tr>

<td>日用百货</td>

<td>10</td>

</tr>

<tr>

<td>电器</td>

<td>20</td>

</tr>

</tbody>

</table>

【例5-26】在下列JavaScript程序中，分别用parentNode和previousSibling获得父节点和左兄弟节点，用nodeName属性输出节点的标记名。本例在浏览器中的运行结果如图5-26（a）、（b）所示。

<html>

<head><title>DOM属性示例</title>

<script language="JavaScript">

function Access( )

{ ShowParentNode( );

ShowLeftSiblingNode( );

}

function ShowParentNode( )

{ var pnode=p1.parentNode;

alert(pnode.nodeName);

}

function ShowLeftSiblingNode( )

{ var prenode=p1.previousSibling;

alert(prenode.nodeName);

}

</script>

</head>

<body>

<h2>这是一个简单的DOM示例</h2>

<p id="p1" onClick="Access( )">

单击这里将弹出二个对话框，<br>

分别显示P标记的父节点和左兄弟节点的标记名称<br>

</p>

</body>

</html>

###### 【例5-27】 使用DOM的方法生成一个表格。它在浏览器中的运行结果如图5-27（a）、（b）所示。

<html>

<head><title>使用DOM生成表格</title>

<script language="JavaScript">

function genTable(pNode)

{

var i,j;

var contents=new Array(3);

for (i=0;i<3;i++)

contents[i]=new Array(2);

contents[0][0]="商品类别";

contents[0][1]="数量";

contents[1][0]="日用百货";

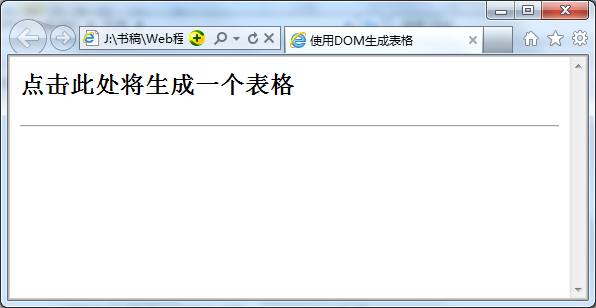
contents[1][1]="10";

contents[2][0]="电器";

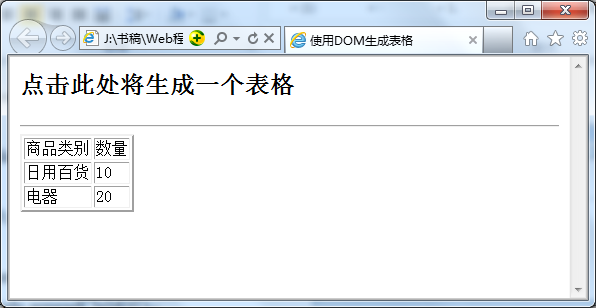
contents[2][1]="20";

var tableNode=document.createElement("TABLE");

var tBodyNode=document.createElement("TBODY");



（a）初始显示



（b）单击文字后，生成一表格

图5-27 例5-27的显示结果

var t1,t2;

for (i=0;i<3;i++)

{

t1=document.createElement("TR");

tBodyNode.appendChild(t1);

for (j=0;j<2;j++)

{

t1=document.createElement("TD");

t2=document.createTextNode(contents[i][j]);

t1.appendChild(t2);

tBodyNode.childNodes[i].appendChild(t1);

}

}

pNode.appendChild(tableNode);

tableNode.id="test";

tableNode.border=2;

tableNode.appendChild(tBodyNode);

}

</script>

</head>

<body id="tableTest">

<h2 onClick="genTable(tableTest)">单击此处将生成一个表格</h2><hr>

</body>

</html>