使用 JavaScript 完成注册功能

1. 编写注册相关 html 元素

- (5) 新建一个普通的 HTML 页面,命名为 test.html。
- (6) 编写蒙板 div。在</body>前输入以下内容:</div id="mask"></div>
- 该 div 将作为注册功能的容器,并且将通过给该 div 应用 css 样式使得其具有蒙板效果。
- 4. 编写注册功能 html form(表单)。在第二步输入的 div 内部输入以下内容:

<form id="registerFrm" action="register.asp" method="post" onsubmit="return validLogin();"> </form>

html form(表单)是一个提交单位。当表单中的某个sumbit或者按钮被单击时,表单的提 交发生。这时窗体的onsubmit事件被触发。上述html代码生成了一个表单,action属性值为 register.asp表明了窗体如果被成功提交,将由register.asp页面在服务器端处理登录。onsubmit 属性值是一个函数调用,return validlogin()表示调用validLogin函数,并且将函数执行的返回 值作为onsubmit属性值。onsubmit属性值如果为false,则提交行为被取消;否则将提交给action 属性所指定的url处理(在本例中即为register.asp)。

- 5. 设计表单元素。在第二步设计的form内部输入以下内容:
 - <fieldset>

<legend>注册新用户</legend>

<input type="submit" value="注册" />

```
<input id="btnClose" type="button" value="关闭" />
```

</fieldset>

第一个input是提交按钮,该按钮单击时将触发表单提交事件。第二个input是一个按钮, 该按钮的作用是单击关闭蒙板。效果如图8-1所示:

> 注册新用户 注册 关闭

图8-1 添加了2个input的效果图

6. 测试表单提交行为。修改"注册"和"关闭"的click事件处理编写javascript代码如

下:

<input type="submit" value="注册" onclick="alert('验证将要开始');" />

<input id="btnClose" type="button" value="关闭" onclick="alert('关闭蒙板');return false;"/>

在</head>前编写javascript代码如下:

<script language"=javascript" type="text/javascript">

function validLogin() {

alert("验证通过");

```
return true;
```

}

</script>

上述代码为第二个input按钮添加了click事件处理,由于返回值为false,因此表单的 提交行为不会发生,因此validLogin函数不会被调用,"验证通过"提示信息不会出现。 效果如图8-2所示:

注册新用户 注册 [美闭]



图8-2 单击"关闭"按钮时效果

标题为"注册"的提交按钮在其click事件处理中,没有显式的返回false,因此表单的提 交事件会被触发。同时由于在validLogin函数中返回了true,因此该表单会被提交到register.asp 处理。其效果图如图8-3至图8-5所示:

注册新用户 [注册] 关词

来自网页	的消息
4	验证将要开始
	确定

图8-3 提交按钮单击提示1



无法提供此类型的页。 **很明:**由于已稍确单止所请求的页类型,无法对读类型的页码供服务,扩展名 "asp"可能不正确。 请检查以下的 URL并确保某种写正确。 **诸余的 URL:** /注册/regater.asp

图8-5 提交按钮单击提示3

7. 完成注册表单的所有内容,表单完整代码如下:

<div id="mask">

```
<form id="registerFrm" action="register.asp" onsubmit="return validLogin();">
<fieldset>
  真实姓名
         <input name="rName" type="text" />
     用户名
        <input name="uName" type="text" />
     密码
        <input name="pwd" type="password" />
     确认密码
        <input name="pwd2" type="password" />
     电子邮件
        <input name="email" type="text" />
     身份证
```

其效果如图8-6所示:

真实姓名	
用户名	
容码	
确认密码	
电子邮件	
身份证	
注册 关闭	

图8-6 注册表单初稿效果

2. 通过 DHTML 生成蒙板效果

(1) 设置蒙板 div 的背景, 边框和定位。在</head>前输入以下内容

<style type="text/css">

#mask {

> position:absolute; background:black; border:solid 1px;

}

</style>

postion 属性为 absolute 将使得蒙板 div 及其内部元素从正常文档流中分离,背景色为全 黑色是为了下面的透明效果做准备。效果如图 8-7 所示:



图 8-7 设置背景,边框和定位后的蒙板 div

(2) 设置透明效果。将 id 为 mask 的 div 样式改为

#mask {

position:absolute; background:black; border:solid 1px; <u>filter: alpha(opacity:50);</u> <u>opacity: 0.50;</u> }

其中粗体加下划线部分为在步骤 1 基础上新添加的内容。filter 和 opacity 属性都可以用 来设置背景的透明度。fiter 属性为 IE 所支持,其他某些浏览器不支持该属性,而 opacity 是 css 标准属性,但是 IE 部分版本的浏览器不支持该属性。通过同时设置 filter 和 opacity 可以 解决浏览器兼容问题。效果如图 8-8 所示:

注册	关闭

图 8-8 设置透明属性后的效果

(3) 编写被遮挡的测试内容。为了显示蒙板效果,在本步骤中向文档添加正常元素,在 <body>后输入下面内容:

<div id='content'>

蒙板下的内容

</div>

效果如图 8-9 所示:

图 8-9 向文档添加正常元素后效果

可以看到蒙板并没有遮挡住正常文档内容,该问题可以通过下面的步骤来解决。

(4) 设置蒙板的左上角位置。为蒙板添加 left 和 top 属性,代码如下:

#mask

{

position: absolute; background: black; border: solid 1px; filter: alpha(opacity:50); opacity: 0.50; <u>left:0px;</u> <u>top:0px;</u>

}

其中粗体加下划线代码是新添加内容。通过设置 left 和 top,可以看到文档中的超链接 被遮挡且不能被单击。效果如图 8-11 所示:

蒙板下的	内容	
注册	关闭	

图 8-11 为蒙板添加 left 和 top 属性后的效果

(5) 设置文档正常内容宽带和高度来测试蒙板的遮挡效果。修改 id 为 content 的 div 为: <div id='content' style='width:600px;height:800px;background:red'>

蒙板下的内容

</div>

设置后的效果如图 8-12 所示:



图 8-12 设置文档正常内容宽带和高度后的遮挡效果

可以看到正常文档元素并没有被蒙板完全遮挡。需要设置蒙板的宽度和高度来解决此问题。

```
(6) 设置蒙板的高度和宽度。修改蒙板 div 的样式为:
```

#mask {

}

position: absolute; background: black; border: solid 1px; filter: alpha(opacity:50); opacity: 0.50; left:0px; top:0px; <u>width:1000px;</u> <u>height:2000px;</u>

其中粗体加下划线代码是新添加内容。设置后效果如图 8-13 所示:





(7) 设置蒙板的字体颜色和 z-index。修改蒙板 div 的样式为:

```
#mask
{
```

}

position: absolute; background: black; border: solid 1px; filter: alpha(opacity:50); opacity: 0.50; left: 0px; top: 0px; width: 1000px; height: 2000px; <u>z-index: 10000;</u> color: White;

其中粗体加下划线代码是新添加内容。z-index 属性(默认值为 0)表明了在 z 方向上元素 的顺序, z-index 属性值最大的在最上面。设置后效果如图 8-14 所示:

真实姓名	
用户名	
密码	
确认密码	
电子邮件	
身份证	
注册 关闭	

图 8-14 设置蒙板的字体颜色和 z-index 后的效果

(8) 隐藏蒙板。蒙板一般在初始状态下是隐藏的,当用户进行某个操作时被显示(例如,单击某个按钮)。为了在初始状态下隐藏蒙板,为蒙板 div 编写属性如下:

#mask {

> position: absolute; background: black; border: solid 1px; filter: alpha(opacity:50); opacity: 0.50; left: 0px; top: 0px; width: 1000px; height: 2000px; z-index: 10000; color: White; **display: none;**

}

display:none 是新添加内容,通过设置此属性值为 none 可以达到隐藏元素的目的。页面 效果如图 8-15 所示:



图 8-15 设置 display 属性后的效果

可以看到蒙板完全被隐藏了。

(9) 编写 javascript 使得超链接被单击时蒙板显示。为"蒙板下的内容" 超链接设置单击 事件处理代码如下:

蒙板下的内容

超链接的单击事件处理代码如果返回 false,将不会跳转到超链接的 href 属性所指定的 地址。编写 display 函数,在</script>前输入代码如下:

function display() {

var maskDiv = document.getElementById('mask'); maskDiv.style.display = 'block'; return false;

}

该段代码先找到 id 为 mask 的 div, 然后设置其 css style 的 display 属性值为 block, 这样将使得蒙板 div 显示出来,效果如图 8-16 所示:

真实姓名	
用户名	
密码	
确认密码 2010年1月1日 1月1日 1月1日 1月1日 1月1日 1月1日 1月1日 1月1日	
电子邮件	
身份证	
注册 关闭	

图 8-16 单击超链接显示蒙板

(10) 调整蒙板大小使得其恰好占满页面。修改 display 函数代码如下:

function display() {

var maskDiv = document.getElementById('mask');

var w = document.body.clientWidth;

 $var \ h = document.body.clientHeight;$

maskDiv.style.width = w+"px";

maskDiv.style.height = h+"px";

maskDiv.style.display = 'block';

return false;

}

上述代码将蒙板 div 的宽度和高度设置为文档中 body 的客户区宽度和高度,达到了占 满页面的目的(请思考蒙板 div 样式中的 height 和 width 属性是否还有意义,并亲手实验)。 其效果如图 8-17 所示:



图 8-17 调整蒙板大小后的效果

(11) 通过 javascript 关闭蒙板。蒙板弹出后,单击蒙板中的"关闭"按钮可以关闭蒙板, 修改"关闭"按钮事件处理代码如下:

<input id="btnClose" type="button" value="关闭" onclick="return hide();" />

编写 hide 函数。在</script>前输入代码如下:

function hide() {

var maskDiv = document.getElementById('mask'); maskDiv.style.display = 'none';

return false;

}

通过将蒙板 div 的 display 样式设置为'none',达到了隐藏蒙板的目的。

(2) 通过 css 样式单使得注册表单更加美观。在</style>前输入 css 代码如下: #mask fieldset

{

border:inset 2px; background-color: #dfe8f7; padding: 30px; width:300px; height:300px;

```
}
设置表格样式。在</style>前输入如下内容:
  #mask fieldset table
  {
    width: 100%;
    border: 1px solid #a0a0a0;
    border-collapse: collapse;
    color: black;
   }
修改表格内容如下:
  真实姓名
         <input name="rName" id='rName' type="text" />
         用户名
         <input name="uName" id='uName' type="text" />
         密码
         <
         <input name="pwd" id='pwd' type="password" />
         确认密码
         <input name="pwd2" id='pwd2' type="password" />
```

```
电子邮件
             <input name="email" id='email' type="text" />
             身份证
             <input name="identity" id='identity' type="text" />
             设置表格内元素样式。在</style>前添加样式内容:
    #mask td.td35r
    {
      font-weight: bold;
      color: #111111;
      text-align: right;
      padding: 3px;
      width: 35%;
      border: 1px solid #a0a0a0;
    }
    #mask td.td651
    {
      text-align: left;
      padding: 3px;
      width: 65%;
      border: 1px solid #a0a0a0;
      height: 30px;
    }
    input[type=text], input[type=password]
    {
      width: 150px;
      border: #ffffff outset;
      font-size: 14px;
```

border-width: 0px 0px 1px 0px;

background-color: #dfe8f7;

```
text-align: left;
```

```
}
```

```
设置按钮样式。使得按钮居中。在</style>前输入如下样式代码:
#div1
{
    position: relative;
    top: 20px;
    left: 100px;
    }
    #btnClose
    {
        margin-left: 30px;
    }

    其中 1: 1. 并未的你田县收包密亚色拉印的书。中国大师中在巴西白
```

其中 div1 样式的作用是将包容两个按钮的 div 相对于原来位置向右和下方移动 100px 和 20px,使得 2 个按钮开起来基本居中显示,并且与上面的元素拉开一小段距离,更加美观。然后修改测试 div 的样式为:

<div id='content' style='width: 600px; height: 800px;'>

蒙板下的内容

</div>

设置后效果如图 8-18 所示:

家板下的内容	
真实姓名	
用户名	
密码	
确认密码	
电子邮件	
身份证	
	注册 关闭

图 8-18 设置 css 样式后的注册表单效果

(3) 通过 javascript 控制注册内容显示在页面的正中间。为蒙板下的 div 添加样式如下: #mask fieldset

{

```
border: inset 2px;
background-color: #dfe8f7;
padding: 30px;
width: 300px;
height: 300px;
position:relative;
```

}

其中,position:relative;是新添加内容。该样式使得蒙板下的 fieldset 采用相对定位。采 用了相对对位的元素可以使用 top 和 left 属性控制其相对于原来位置向下和右方做移动。为 了方便访问 fieldset 元素,为其添加 id 属性值'registerContent',然后修改 display 函数如下:

function display() {

var maskDiv = document.getElementById('mask');

var w = document.body.clientWidth

```
var h = document.body.clientHeight;
```

```
maskDiv.style.width = w + "px";
```

```
maskDiv.style.height = h + "px";
```

```
maskDiv.style.display = 'block';
```

```
var fieldset = document.getElementById('registerContent');
fieldset.style.top = (h / 2 - 150) + 'px';
fieldset.style.left = (w / 2 - 150) + 'px';
```

return false;

}

其中, 粗斜体加下划线代码是新添加内容。修改后效果如图 8-19 所示:



图 8-19 注册表单居中效果

```
(4) 完成验证函数 validLogin。修改 validLogin 函数代码为:
        function validLogin() {
          var rName = document.getElementById('rName');
          var result = true;
          if (!rName.value) {
             alert('真实姓名不能为空');
             result = false;
          }
          var uName = document.getElementById('uName');
          if (!uName.value) {
             alert('用户名不能为空');
             result = false;
          }
          var pwd = document.getElementById('pwd');
          if (!pwd.value) {
             alert('密码不能为空');
             result = false;
          }
          var pwd2 = document.getElementById('pwd2');
          if (pwd2.value != pwd.value) {
             alert('两次密码输入不一致');
             result = false;
          }
          var email = document.getElementById('email').value;
          //为空或者不存在@字符
          if (!email || email.indexOf('@') == -1) {
             alert('电子邮件格式不正确');
             result= false;
          }
          return result;
       }
    上述代码验证规则为:真实姓名,用户名,密码不能为空;二次密码必须和密码一致;
电子邮件不能为空且必须包含'@'符号。为身份证的验证定义函数 validIndentity 如下:
      /**
      **身份证验证函数,验证规则为:待验证字符串17位数字开头,最后一位是任意ASCII字符
      **接受一个参数v,表示待验证的字符串
      **返回一个bool值,返回值表明了参数v所代表的字符串是否符合身份证验证规则
      **如果v是一个身份证,则返回true,否则返回false
      */
      function validIdentity(v) {
          //17位数字+1位任意ASCII字符
```

```
var reg = /^\d{17}\w$/;
return reg.test(v);
}
修改 validLogin, 在 return false; 前添加如下代码:
var identity = document.getElementById('identity').value;
if (!validIdentity(identity)) {
    alert('身份证不正确');
    result = false;
    }
(5) (法件) 新公民 自份证号现按照 CD11642 1000
```

(5) (选做)新公民身份证号码按照 GB11643—1999《公民身份证号码》国家标准编制, 由 18 位数字组成:前 6 位为行政区划分代码,第 7 位至 14 位为出生日期码,第 15 位至 17 位为顺序码,第 18 位为校验码。请考虑第 7-14 位,完善 validIdentity 函数。

(6)(选做)在本实验的第14步中,采用 alert 函数来进行验证提示,请考虑使用一个隐藏 div 来实现验证提示。当验证未通过时,该 div 显示在表单底部,并且在该 div 中显示验证提示。

- (7) 新建一个 asp 页面,保存为 register.asp,修改<body></body>内容如下:<body>注册成功</body>
- (8) 打开 ex9 文件夹下的 register.html 页面,页面效果如图 8-20 所示:



图 8-20 register.html 效果

参照本实验的前述步骤,完成"技术支持"主菜单项下的二级菜单项"注册"对应功能。