

课程回顾

1 三级联动的参考代码

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title></title>
6   </head>
7   <script type="text/javascript">
8     //定义数组保存所有的省
9     var province=["湖北省","河南省","湖南省","河北省"];
10    //使用省份名称为key，存入该省对应的市
11    province["湖北省"]=["武汉市","黄冈市","襄阳市","荆州市"];
12    province["河南省"]=["郑州市","信阳市","洛阳市","驻马店"];
13    province["湖南省"]=["长沙市","郴州市","岳阳市","衡阳市"];
14    province["河北省"]=["石家庄市","邯郸市","秦皇岛市"];
15    //一个格式，实现市和区的对应
16    province["武汉市"]=["洪山区","武昌区","汉口区"];
17    province["襄阳市"]=["襄州区","宜城市","谷城市"];
18    province["长沙市"]=["羊区","人区"];
19
20    //网页加载完毕，就加载所有的省份到select
21    window.onload=function(){
22      //获取省份的select
23      var objParentSelect=document.getElementById("selProvince");
24      //将数组中所有的省份追加到select中
25      /**
26       * 1.创建option createElement()
27       * 2.创建text createTextNode(数据从何而来???)
28       * 3.父.appendChild(子);
29       */
30      for(var i=0;i<province.length;i++){
31        var strProv=province[i];//获取省份名称
32        //获取省份，以option标签的形式，放在select列表显示
33        //1.创建option
34        var objNewOp=document.createElement("option");
35        //2.设置option内部显示的数据
36        var objText=document.createTextNode(strProv);
37        //3.将文本对象设置到option显示
38        objNewOp.appendChild(objText);
39
40        //4.将设置好的option，放到select显示
41
42        objParentSelect.appendChild(objNewOp);
43      }
44    }
45
46    /**
47     * 二级联动:
48     * 1.给省份下拉列表添加onchange
49     * 2.获取用户选择的省份对应的城市
50     * 3.创建option，显示城市，并将创建的option添加到市对应的select
```

```

51      */
52      var objCitySel=document.getElementById("selCity");
53      objParentSelect.onchange=function(){
54
55          //清空，保留第一项
56          var arrSon=objCitySel.getElementsByTagName("option");//获取
    所有的城市
57          while(arrSon.length>1){
58              objCitySel.removeChild(arrSon[1]);
59          }
60
61          //获取用户选择的省份
62          // alert("您选择的省份是："+this.value);//this指事件源 所有的表单项
    获取用户输入的或者选择的都是用value
63          var strProv=this.value;
64          //获取省份对应的城市
65          var arrCities=province[strProv];
66          //排除省没有对应的市的问题
67          if(!arrCities){
68              return;
69          }
70          for(var i=0;i<arrCities.length;i++){
71              var strCity=arrCities[i];//获取省份名称
72              //获取省份，以option标签的形式，放在select列表显示
73              //1.创建option
74              var objNewOp=document.createElement("option");
75              //2.设置option内部显示的数据
76              var objText=document.createTextNode(strCity);
77              //3.将文本对象设置到option显示
78              objNewOp.appendChild(objText);
79
80              //4.将设置好的option，放到select显示
81              objCitySel.appendChild(objNewOp);
82
83          }
84
85          //县区列表数据也要发生变化
86          objCitySel.onChange();//函数调用
87          //省变了，市数据也会变化，县区目前应该在没有有效的市，下拉列表应该是空
    的
88      }
89
90      /**
91      * 三级联动：
92      * 1.给市下拉列表添加onchange
93      * 2.获取选中的市对应的县区
94      * 3.DOM将县区的数据追加县区的下拉列表
95      */
96      objCitySel.onChange=function(){
97          //1-2 父.appendChild(子节点)
98          var objRegionSel=document.getElementById("selRegion");
99          //清空
100         var arrSon=objRegionSel.getElementsByTagName("option");
101         while(arrSon.length>1){
102             objRegionSel.removeChild(arrSon[1]);
103         }
104         //获取用户选择的市 value
105         var strCity=this.value;//获取城市下拉列表用户的选中项

```

```

106
107 //获取县区的数据
108 var arrRegion=province[strCity];
109 //undefined判断
110 if(!arrRegion){
111     return;
112 }
113 //循环，动态DOM加载
114 for(var i=0;i<arrRegion.length;i++){
115     //追加
116     //1-1 创建
117     var objOpt=document.createElement("option");
118     var objText=document.createTextNode(arrRegion[i]);
119     objOpt.appendChild(objText);
120
121     objRegionSel.appendChild(objOpt);
122 }
123 }
124 }
125 </script>
126 <body>
127     省份:
128     <select id="selProvince">
129         <option>====请选择所在的省份====</option>
130     </select><br />
131     市: <select id="selCity">
132         <option>====请选择所在的市====</option>
133     </select><br />
134     区: <select id="selRegion">
135         <option>====请选择所在的县区====</option>
136     </select><br />
137 </body>
138 </html>

```

课程目标

1 RegExp对象

2 表单验证的思路 ===== 掌握

3 innerText 和 innerHTML === 掌握

4 BOM (window/location/history)===掌握

课程实施

1 innerText 和 innerHTML

1-1 innerText 和 innerHTML功能

用在双标签上面，获取开始标签和闭合标签内部的内容

1-2 innerText 和 innerHTML区别

innerText: 设置或获取文本部分, 遇到文本中含有html代码, 不会解析

innerHTML: 设置或获取html代码+文本部分, 遇到文本中含有html代码, 会解析

1-3 innerText和innerHTML的应用案例

```
1  <!DO    CTYPE html>
2  <html>
3      <head>
4          <meta charset="utf-8" />
5          <title></title>
6      </head>
7      <script type="text/javascript">
8          window.onload=function(){
9              //1. 获取id是div1的内部的内容
10             var objDiv1=document.getElementById("div1");
11             //2. 获取innerText innerHTML
12             alert(objDiv1.innerText);//百度
13             alert(objDiv1.innerHTML);//<a href="http://www.baidu.com">百度
14         </a>
15
16         //div2里面追加一个a, sina.com.cn
17         var objDiv2=document.getElementById("div2");
18         var strText=objDiv2.innerHTML;
19         alert(strText);
20         // strText="大家好"+"<a href='http:www.sina.com.cn'>新浪</a>"
21         strText+="<a href='http://www.sina.com.cn'>新浪</a>";
22
23         objDiv2.innerText=strText;//设置,innerText不会接下a的代码
24         objDiv2.innerHTML=strText;//设置,innerHTML显示汉字,且解析a的代码
25     }
26 </script>
27 <body>
28     <!--
29         getElementById():element
30         //获取标签里面的数据
31         value只有表单项具有value。普通标签没有value的属性
32
33         双标签, 开始标签和闭合标签之间的内容: innerText innerHTML
34     -->
35     <div id="div1">
36         <a href="http://www.baidu.com">百度</a>
37     </div>
38     <div id="div2">大家好</div>
39 </body>
40 </html>
```

课后作业: 动态表格的实现代码

```
1  <!DO    CTYPE html>
2  <html>
3      <head>
4          <meta charset="utf-8" />
```

```

5     <title></title>
6 </head>
7 <script type="text/javascript">
8     /**
9     * 简化document.getElementById(id)
10    * 位置：放在window.onload外面
11    * 执行时机，与网页是否加载完毕没有关系
12    * 作用域：
13    */
14    function $(id){
15        return document.getElementById(id);
16    }
17    window.onload=function(){
18        /**
19        * 动态表格实现思路：
20        * 1.按钮注册onclick事件监听器
21        * 2.获取用户输入的行和列值（判断）
22        * 3.拼接字符串： 表格的代码
23        */
24        $("btn").onclick=function(){
25            var strRow=$("txtRow").value;
26            var strCol=$("txtColumn").value;
27
28            //动态添加table
29            var str="<table border='1' cellspacing='0'
cellpadding='4px'>";
30            for(var i=0;i<strRow;i++){
31                str+="<tr>";
32                for(var j=0;j<strCol;j++){
33                    str+="<td>aaaaaa</td>";
34                }
35                str+="</tr>";
36            }
37            str+="</table>";
38            //div2显示table
39            $("div2").innerHTML=str;
40            // alert(str);
41        };
42
43    }
44
45 </script>
46 <body>
47     <p>
48     行: <input type="text" id="txtRow" />
49
50     列: <input type="text" id="txtColumn" />
51     <button id="btn">生成表格</button>
52     </p>
53     <div id="div2">大家好</div>
54 </body>
55 </html>
56

```

2 RegExp对象

应用场景：字符串的格式校验、字符串截取、匹配

引入案例：注册功能

要求用户输入用户名、输入密码、输入qq号码

举例：qq号码：纯数字 用户名：数字、字母、_、# %组成，长度一般要求不少于6位

如何保证用户输入的数据合法呢？

- 1 String提供的方法，某一个格式验证有思考的思路，但是实施起来非常繁琐
- 2 正则表达式

2-1 定义对象

- 1 //字面量方式
- 2 var 对象名=/字符串验证模板格式/验证属性；
- 3 验证属性：
- 4 i:Ignore 忽略大小写
- 5 g:global 全局
- 6 m:multiple 多次重复

2-2 RegExp常用方法

- 1 test():验证字符串是否符合验证格式 true false
- 2 exec():提取与正则格式匹配的字符串 object

课堂案例

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title></title>
6   </head>
7   <script type="text/javascript">
8     //1.字符串
9     var str="i Love you,you Love me~! ! ";
10    //2.判断str是否有love这个词
11    var reg=/love/i;
12    //3.str中有love这个词语吗？
13    alert(reg.test(str)); //str中有没有love，有就是true 没有就是false
14
15    //4.获取匹配的项
16    var result=reg.exec(str);
17    //了解：result包含匹配的字符串，包含字符串出现的下标起始位置、提供验证的字符串的
    对象
18    alert(result+"在"+result.input+"出现，出现的位置是："+result.index);
19  </script>
20  <body>
21  </body>
22 </html>
```

2-3 String中与RegExp对象相关的方法

- 1 `replace()`: 替换字符串
- 2 `search()`: 查找字符串
- 3 `match()`: 雷同exec 提取匹配的字符串

课堂案例

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title></title>
6   </head>
7   <script type="text/javascript">
8     //1.字符串
9     var str="i Love1 you,you Love2 me~LOVE3! love9! ";
10    //2.判断str是否有love这个词
11    //^:指定字符串开始的符号  $:指定字符串结束符号
12    var reg=/love\d/ig;
13    //替换love为*
14    str=str.replace(reg,"*");//不添加g属性, reg匹配love, 匹配一次, 添加g属性
    后, 匹配所有
15    alert(str);//i * you,you * me~*!!
16
17    var index=str.search(reg);//查找匹配的字符串出现的初始位置
18    alert(index);//love第一次出现的下标位置: 2,
19
20    var arr=str.match(reg);//获取正则所有的匹配项
21    alert(arr);//Love1,Love2,LOVE3, love9
22  </script>
23  <body>
24  </body>
25 </html>
26
```

3 表单验证思路

3-1 表单验证的必要性

- 1 用户体验度
- 2 安全性: 非法的数据提交给后台, 会有注入风险
- 3 服务器处理无效的客户端请求, 增加服务器处理的压力

3-2 验证思路

- 1 对用户输入的数据合法性校验
- 2 如果验证不通过, 阻止表单提交给后台
- 3 解释: 验证不通过, 单击submit按钮, 没有反应
- 4
- 5 难点: 表单如何不提交!! submit不需要程序员写任何代码, 自己本身具有提交表单功能!!

课堂案例

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title></title>
6   </head>
7   <script type="text/javascript">
8     window.onload=function(){
9       //给form添加onsubmit的监听器，验证手机号码是否正确
10      document.getElementById("form1").onsubmit=function(){
11        //获取用户输入的手机号
12        var phone=document.getElementById("txtPhone").value;
13        //正则
14        var reg=/^1[35789]\d{9}$/;
15        if(!reg.test(phone)){
16          //submit事件返回值的作用：true--数据提交后台 false--数据不提交
17          return false;
18        }
19        return true;
20      }
21    }
22  </script>
23  <body>
24    <!--
25      阻止submit按钮默认行为：
26      return false: 表单提交时，如果看到提交监听器注册function的返回值是false
27      onsubmit="return 验证结果;"
28    -->
29    <form id="form1" action="http://www.baidu.com" method="post" >
30      <p>
31        手机号码: <input id="txtPhone" type="text" name="phone"/>
32      </p>
33      <p>
34        用户名: <input id="txtUserName" name="username" type="text"/>
35      </p>
36      <p>
37        <!--
38          submit单击之后不是click事件。触发form的submit事件
39        -->
40        <input type="submit" value="注册" />
41      </p>
42    </form>
43  </body>
44 </html>
45
```

补充：手机号码输入完毕之后，离开文本框一瞬间，验证手机号并给出提示

- 1 获取焦点: focus
- 2 失去焦点: blur

参考代码

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="UTF-8">
5          <title></title>
6      </head>
7      <script type="text/javascript">
8
9      </script>
10     <body>
11         <form id="form1" action="http://www.baidu.com" method="post" >
12             <p>
13                 手机号码: <input id="txtPhone" type="text" name="phone"
onblur="alert('焦点离开了')"/>
14             </p>
15             <p>
16                 用户名: <input id="txtUserName" onfocus="alert('获取焦点了')"
type="text"/>
17             </p>
18             <p>
19                 <input type="submit" value="注册" />
20             </p>
21         </form>
22     </body>
23 </html>
24
```

使用onblur完成数据验证的实时提示

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="UTF-8">
5          <title></title>
6      </head>
7      <script type="text/javascript">
8          /**
9           * 验证手机号码
10          */
11          function checkPhone(){
12              //获取用户输入的手机号
13              var objText=document.getElementById("txtPhone");
14              var phone=objText.value;
15              //正则
16              var reg=/^1[35789]\d{9}$/;
17              if(!reg.test(phone)){
18                  //提示
19                  alert("手机号码输入不正确, 请重新输入! ");
20                  document.getElementById("spanPhone").innerText="验证不通过";
21                  //现有手机号码全选
22                  objText.select();
23                  //清空
24                  objText.value='';
25                  return false;
26              }
27          }
28      </script>
29  </html>
```

```

26         document.getElementById("spanPhone").innerText="验证通过";
27         return true;//在表单提交的时候,
28     }
29
30     /**
31     * 验证表单
32     */
33     function doSubmit(){
34
35         return checkPhone();
36     }
37 </script>
38 <body>
39     <!--
40     阻止submit按钮默认行为:
41     return false:表单提交时, 如果看到提交监听器注册function的返回值是false
42     onsubmit="return 验证结果;"
43     -->
44     <form id="form1" action="http://www.baidu.com" method="post"
onsubmit="return doSubmit();">
45         <p>
46             手机号码: <input id="txtPhone" type="text" name="phone"
onblur="checkPhone()" />
47             <span id="spanPhone" style="color: red;"></span>
48         </p>
49         <p>
50             用户名: <input id="txtUserName" type="text" name="username" />
51         </p>
52         <p>
53             <!--
54             submit单击之后不是click事件。触发form的submit事件
55             -->
56             <input type="submit" value="注册" />
57         </p>
58     </form>
59 </body>
60 </html>

```

4 BOM

4-1 BOM概述

BOM: Browser(浏览器) Object Model

bom研究浏览器对象

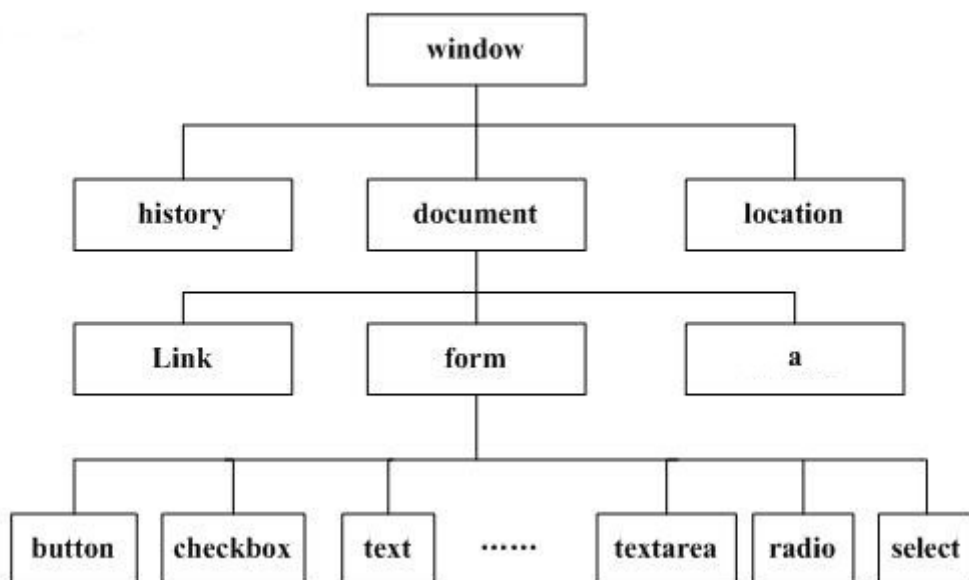
因为浏览器商业竞争的原因, 同一个对象同一属性获取的值会存在兼容性问题

window对象是整个BOM的顶层对象, 处于对象模型的第一层, 对于每个打开的窗口, 系统都会自动将其定义为window对象。

各个对象所处位置及其关系如下图所示:



window是一个顶级对象，一般window的方法或者属性使用时，不需要写window



事件名称	描述
<code>prompt()</code>	显示可提示用户输入的对话框，返回值为输入的内容。
<code>alert()</code>	显示带有一个提示消息和一个确定按钮的警示框。
<code>confirm()***</code>	显示一个带有提示信息、确定和取消按钮的确认框，当点击确认按钮时返回 true 当点击取消按钮时返回 false 。
<code>close()**</code>	关闭浏览器窗口
<code>open()**</code>	打开一个新的浏览器窗口，加载给定 URL 所指定的文档
<code>setTimeout()</code>	在设定的毫秒数后调用函数或计算表达式
<code>setInterval()**</code>	按照设定的周期（以毫秒计）来重复调用函数或表达式
<code>clearInterval()***</code>	取消重复设置，与 <code>setInterval</code> 对应

课堂案例

弹框

```

1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="UTF-8">
5          <title></title>
6      </head>
7      <script type="text/javascript">
8          //提示框：有一个确认按钮=true、取消按钮=false
9          var bool=window.confirm("确认退出吗? ");
10         alert(bool);
11     </script>
12     <body>
13
14     </body>
15 </html>
16

```

open和close

```

1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="UTF-8">
5          <title></title>
6      </head>
7      <body>
8          <button onclick="window.open('http://www.baidu.com', 'new',
9              'width=500,height=300,left=200,top=200,location=no,menubar=no,toolbar=0,resizable=no');">打开百度</button>
10         <button onclick="window.close();">关闭当前窗口</button>
11     </body>
12 </html>

```

定时器

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title></title>
6   </head>
7   <script type="text/javascript">
8     // alert('hello');
9     function showTime(){
10    //     alert('hello');
11     var strDate=new Date().toLocaleString();
12     document.getElementById("div1").innerText="当前时间: "+strDate;
13   }
14   var id;//保存定时器id
15   window.onload=function(){
16     showTime();
17     //showTime不停重复执行
18     id=setInterval(showTime,1000);
19   }
20
21   function stop(){
22     // window.clearInterval("要停止的定时器的id")
23     clearInterval(id);
24   }
25
26   //setInterval(功能,间隔时间):定时器效果
27   // window.setInterval(sayHello,1000);//毫秒
28 </script>
29 <body>
30   <div id="div1">
31     当前时间:
32   </div>
33   <button id="btn" onclick="stop()">停止时间</button>
34 </body>
35 </html>
```

弹框方法

alert():警告确认框，只有一个按钮：确认

java中会频繁使用窗口：confirm()

confirm():弹出一个确认框，有一个取消和确认按钮

prompt(): 带有一个输入框的提示框

4-2 location对象

课程总结

课程总结

1 innerHTML innerText

2 重点 难点：表单验证

3.window常用方法：弹窗方法、open和close方法、定时器方法 (setInterval() clearInterval())

预习安排

jQuery：选择器 DOM