

JQuery回顾

```
1  开发步骤
2  1.jquery提供外部js文件
3  2.项目引入jquery
4  3.jquery代码书写，要求：
5  $(function(){
6      //代码处理
7  })
8
9  $()原型：
10 $(document).ready(function(){
11
12 });
13
14 $.ready(function(){
15
16 }
17 $(function(){
18
19 });
20
21 选择器：
22 基本选择器：
23  class选择器 $(".class值")
24  id选择器 $("#id值")
25  通配符选择器 $("*")
26  标签选择器 $("div")
27
28  JQ对象和DOM对象互相转换：*****
29  jq-->DOM $.get(下标):Array<Element>
30          $()[下标]
31
32  DOM-->JQ
33          $(DOM对象)
34
35  掌握：
36      事件注册：click() focus() blur()
37      数组循环方式：each(function(下标, DOM对象){})
38      属性值设置或获取方式：获取 prop(属性名) 设置prop(属性名,属性值)
39      获取双标签内部文本的方式：获取==text() html() 设置==text(string)
          html(string)
```

课程目标

1 选择器 ===== 掌握

2 jquery事件 ===== 掌握

3 DOM方法=====掌握

课程实施

1 选择器

1-1 层级选择器

```
1 1、后代选择器
2     语法: ${“ancestor descendant”}
3     用法: $(“form input”) ;    返回值  集合元素
4     说明: 在给定的祖先元素下匹配所有后代元素.这个要下面讲的”parent > child”区分开.
5 2、子选择器
6     语法: ${“parent > child”}
7     用法: $(“form > input”) ;    返回值  集合元素
8     说明: 在给定的父元素下匹配所有子元素.注意:要区分好后代元素与子元素
9 4、相邻元素选择器
10    语法: ${“prev + next”}
11    用法: $(“h2+dl ” );    返回值  集合元素
12    说明: 匹配所有紧接在 prev 元素后的 next 元素, 代码意思是选取紧邻<h2>元素之后的同
    辈元素<dl>
13 5、同辈元素选择器
14    语法: $(“prev ~ siblings”)
15    用法: $(“form ~ input”) ;    返回值  集合元素
16    说明: 匹配 prev 元素之后的所有 siblings 元素.注意:是匹配之后的元素,不包含该元素
    在内,并且siblings匹配的是和prev同辈的元素,其后辈元素不被匹配
```

课堂案例

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4         <meta charset="utf-8" />
5         <title></title>
6         <script type="text/javascript" src="js/jquery-1.8.3.js" ></script>
7     </head>
8     <script type="text/javascript">
9         $(function(){
10             //获取ul里面春天那个汉字
11             var arrA=$(“ul a”);
12             alert(arrA.html());
13             //获取ul所有的li的个数
14             var arrEles=$(“ul>li”);
15             // alert(arrEles.size());//1.8的jquery支持size(),也可以使用length
16             // alert(arrEles.length);//3.0+jquery不支持size(),使用length替换使用
17
18             //春天后面的紧邻季节是什么季节
19             var arrEless=$(“#li1+li”);
20             // alert(arrEless.text());//jQuery支持隐式迭代
21             //春天的后面还有哪些季节
22             var arr=$(“#li1~li”);
23             // alert(arr.text());////jQuery支持隐式迭代,所以不用自己循环数组
24         })
25     </script>
26     <body>
27         <ul>
28             <li id="li1">
```

```

29         <a href="http://www.spring.io">
30             春天</a></li>
31         <li>夏天</li>
32         <li>秋天</li>
33         <li>冬天</li>
34     </ul>
35     <ol>
36         <li>Spring</li>
37         <li>Summer</li>
38         <li>Autumn</li>
39         <li>Winter</li>
40     </ol>
41 </body>
42 </html>

```

1-2 属性选择器

- 1 1、[attribute]
- 2 举例说明：\$(" [href]")选取含有href属性的元素
- 3 2、[attribute=value]
- 4 举例说明：\$(" [href ='#']")选取href属性值为“#”的元素
- 5 3、[attribute !=value]
- 6 举例说明：\$(" [href !='#']")选取href属性值不为“#”的元素
- 7 4、[attribute^=value]
- 8 举例说明：\$(" [href^='en']")选取href属性值以en开头的元素
- 9 5、[attribute\$=value]
- 10 举例说明：\$(" [href\$='.jpg']")选取href属性值以.jpg结尾的元素
- 11 6、[attribute*=value]
- 12 举例说明：\$(" [href*='txt']")选取href属性值中含有txt的元素
- 13 7、综合应用 [attributeFilter1][attributeFilter2][attributeFilterN]
- 14 举例说明：\$("input[id][name\$='man']")选取所有含有 id 属性,并且它的 name 属性是以 man 结尾的input元素
- 15 备注：这是一个组合属性选择器,需要同时满足多个条件时使用

课堂案例

```

1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="utf-8" />
5          <title></title>
6          <script type="text/javascript" src="js/jquery-1.8.3.js" ></script>
7      </head>
8      <script type="text/javascript">
9          $(function(){
10              $("button").click(function(){
11                  //第一步：获取所有的复选框 js:getElementsByName()
12                  var arrHobbies=$("input[name='hobby']");
13
14                  //          第二步：选中的复选框有一个属性：checked=checked
15                  var content='';
16                  arrHobbies.each(function(i,dom){
17                      if($(dom).prop("checked")){
18                          //val()等价于js中value
19                          content+=$(dom).val();
20                      }

```

```

21         })
22
23         //显示结果
24         $("#div1").text(content);
25
26     });
27 }
28 </script>
29 <body>
30     <input type="checkbox" name="hobby" value="音乐1"/>音乐1
31     <input type="checkbox" name="hobby" value="音乐2"/>音乐2
32     <input type="checkbox" name="hobby" value="音乐3"/>音乐3<br />
33     <input type="checkbox" name="hobby" value="音乐4"/>音乐4
34     <input type="checkbox" name="hobby" value="音乐5"/>音乐5
35     <input type="checkbox" name="hobby" value="音乐6"/>音乐6<br />
36     <input type="checkbox" name="hobby" value="音乐7"/>音乐7
37
38     <button>获取选中的爱好</button>
39     <div id="div1">
40
41     </div>
42 </body>
43 </html>

```

1-3 基本过滤选择器

语法	描述	示例
:eq(index)	选取索引等于 index 的元素 (index 从 0 开始)	\$("li:eq(1)") 选取索引等于 2 的元素
:gt(index)	选取索引大于 index 的元素 (index 从 0 开始)	\$("li:gt(1)") 选取索引大于 1 的元素 (注: 大于 1, 不包括 1)
:lt(index)	选取索引小于 index 的元素 (index 从 0 开始)	\$("li:lt(1)") 选取索引小于 1 的元素 (注: 小于 1, 不包括 1)
:header	选取所有标题元素, 如 h1~h6	\$(":header") 选取网页中所有标题元素
:focus	选取当前获取焦点的元素	\$(":focus") 选取当前获取焦点的元素
:animated	选择所有动画	\$(":animated") 选取当前所有动画元素
语法	描述	示例
:first	选取第一个元素	\$("li:first") 选取所有元素中的第一个元素
:last	选取最后一个元素	\$("li:last") 选取所有元素中的最后一个元素
:not(selector)	选取去除所有与给定选择器匹配的元素	\$("li:not(.three)") 选取 class 不是 three 的元素
:even	选取索引是偶数的所有元素 (index 从 0 开始)	\$("li:even") 选取索引是偶数的所有元素
:odd	选取索引是奇数的所有元素 (index 从 0 开始)	\$("li:odd") 选取索引是奇数的所有元素

课堂案例

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8" />
5     <title></title>
6     <script type="text/javascript" src="js/jquery-3.1.1.min.js" >
7   </script>
8   <script type="text/javascript">
9     $(function(){
10       //获取第一个ul中第三个li的内容
11       var objLi=$("ul:first>li:eq(2)");
12       // alert(objLi.text());
13
14       //获取ol中下标是偶数的li的内容*****
15       var arrEvenLi=$("ol li:even");//even是偶数
16       //设置li的背景色 设置与外观样式相关的属性，都可以使用css() 链式编程
17       arrEvenLi.css("background-
color","pink").css("color","darkblue").css("font-size",12);
18       //获取ol中下标是奇数的li的内容
19       var arrOddLi=$("ol li:odd");//odd是奇数
20       arrOddLi.css("background-color","greenyellow");
21
22       //ul找li, 排除id是li1
23       var arrLi=$("ul li:not('#li1')");
24       alert(arrLi.text());
25     })
26   </script>
27   <body>
28     <ul>
29       <li id="li1">
30         <a href="http://www.spring.io">
31           春天</a></li>
32       <li>夏天</li>
33       <li>秋天</li>
34       <li>冬天</li>
35     </ul>
36     <ol>
37       <li>Spring</li>
38       <li>Summer</li>
39       <li>Autumn</li>
40       <li>Winter</li>
41     </ol>
42   </body>
43 </html>
```

表格奇偶行交替背景色

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title></title>
```

```

6      <script type="text/javascript" src="js/jquery-3.1.1.min.js" >
</script>
7      </head>
8
9      <script type="text/javascript">
10     $(function(){
11         //获取表格的奇数行
12         $("table tbody>tr:odd").css("background-color","pink");
13
14         //表格偶数行
15         $("table tbody>tr:even").css("background-color","lightblue");
16     })
17 </script>
18 <body>
19     <table border="1" cellspacing="0" cellpadding="4">
20         <thead>
21             <tr>
22                 <th>编号</th>
23                 <th>姓名</th>
24                 <th>年龄</th>
25             </tr>
26         </thead>
27         <tbody>
28             <tr>
29                 <td>1</td>
30                 <td>张三</td>
31                 <td>24</td>
32             </tr>
33             <tr>
34                 <td><p id="mini">2</p></td>
35                 <td>jack</td>
36                 <td>29</td>
37             </tr>
38             <tr>
39                 <td>3</td>
40                 <td>lily熊</td>
41                 <td>14</td>
42             </tr>
43             <tr>
44                 <td>3</td>
45                 <td>lily熊</td>
46                 <td>14</td>
47             </tr>
48         </tbody>
49         <tfoot>
50             <tr>
51                 <td colspan="3">
52                     <a href="#">首页</a>
53                     <a href="#">上一页</a>
54                     <a href="#">下一页</a>
55                     <a href="#">尾页</a>
56                 </td>
57             </tr>
58         </tfoot>
59     </table>
60 </body>
61 </html>

```

1-4 内容过滤选择器

```
1 内容过滤选择器的过滤规则主要体现在它所包含的子元素和文本内容上
2 1、:contains(text)
3 用法: $("div:contains('John')") 返回值 集合元素
4 说明: 匹配包含给定文本的元素.这个选择器比较有用,当我们要选择的不是dom标签元素时,它就派上了用场了,它的作用是查找被标签""围起来的文本内容是否符合指定的内容的.
5 2、:empty
6 用法: $("td:empty") 返回值 集合元素
7 说明: 匹配所有不包含子元素或者文本的空元素
8 3、:has(selector)
9 用法: $("div:has('.mini')")返回值 集合元素
10 说明: 匹配含有选择器所匹配的元素元素
11 4、:parent
12 用法: $("td:parent") 返回值 集合元素
13 说明: 匹配含有子元素或者文本的元素.注意:这里是":parent",可不是".parent"!可以理解与上面讲的":empty"形成反义词.
```

课堂案例

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5 <meta charset="UTF-8">
6 <title></title>
7 <script type="text/javascript" src="js/jquery-1.8.3.js"></script>
8 </head>
9 <script type="text/javascript">
10     $(function() {
11         //1.按钮注册click事件监听器
12         $("#sear").click(function(){
13             //1.获取用户的搜索条件
14             var strSearch=$("#find").val();
15             //2.table里面包含搜索条件的td显示
16             //标签: 显示show() 隐藏: hide()
17             //2-1 隐藏所有的数据
18             $("table tr:gt(0)").hide();
19             //2-2 将匹配的行显示
20             $("table tr:contains('"+strSearch+"')").show();
21         });
22
23         //文本框的键盘弹起事件
24         $("#find").keyup(function(){
25             //1.获取用户的搜索条件
26             var strSearch=$(this).val();//($("#find").val());
27             //2.table里面包含搜索条件的td显示
28             //标签: 显示show() 隐藏: hide()
29             //2-1 隐藏所有的数据
30             $("table tr:gt(0)").hide();
31             //2-2 将匹配的行显示
32             if(!strSearch){
33                 //显示所有
34                 $("table tr:gt(0)").show();
35             }else{
36                 $("table tr:contains('"+strSearch+"')").show();
```

```

37         }
38     });
39 });
40 </script>
41
42 <body>
43     <input type="text" id="find" />
44     <input type="button" value="search" id="sear"/>
45     <table border="1" cellspacing="0" cellpadding="4">
46         <thead>
47             <tr>
48                 <th>编号</th>
49                 <th>姓名</th>
50                 <th>年龄</th>
51             </tr>
52         </thead>
53         <tbody>
54             <tr>
55                 <td>1</td>
56                 <td>张三</td>
57                 <td>24</td>
58             </tr>
59             <tr>
60                 <td><p id="mini">2</p></td>
61                 <td>jack</td>
62                 <td>29</td>
63             </tr>
64             <tr>
65                 <td>3</td>
66                 <td>lily熊</td>
67                 <td>14</td>
68             </tr>
69             <tr>
70                 <td>3</td>
71                 <td>lily熊</td>
72                 <td>14</td>
73             </tr>
74         </tbody>
75     </table>
76 </body>
77 </html>

```

1-5 表单选择器【重点】

```

1  表单项: input, 这一类标签选择器语法:
2  $(" :type属性值")
3
4  select:
5      $("select[name='aaa']")
6  textarea:
7      $("textarea")

```


表单属性过滤选择器【重点】

```
1  选中的复选框、选中的单选按钮
2  $(":checkbox:checked")
3  $(":radio:checked")
4
5  获取select中用户选中的option:
6  $("select option:selected")
7
8  理解:
9  选择所有的不可用的表单项
10 $(":disable")
11
12 选择所有的不可用的表单项
13 $(":enable")
```

课堂案例

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="UTF-8">
5          <title></title>
6          <script type="text/javascript" src="js/jquery-1.8.3.js" ></script>
7      </head>
8      <script type="text/javascript">
9          $(function(){
10             $(":button").click(function(){
11                 //1.获取用户名
12                 var usrName=$(":text").val();
13                 //2.密码
14                 var usrPwd=$(":password").val();
15
16                 //3.单选按钮
17                 var sex=$(":radio:checked").val();
18
19                 //4.学历
20                 var level=$("select option:selected").val();
21
22                 //5.爱好
23                 var checkedBox=$(":checkbox:checked");
24
25                 var hobby='';//没有隐式迭代
26                 checkedBox.each(function(i,dom){
27                     hobby+=$(dom).val();
28                 });
29
30                 //6.备注
31                 var remark=$("textarea").val();
32
33                 //7.头像
34                 var face=$(":file").val();
35
36                 //8.隐藏信息 :hidden与可见性选择器重复了,
37                 var info=$("input:hidden").val();
38             });
10             $(":button").click(function(){
11                 //1.获取用户名
12                 var usrName=$(":text").val();
13                 //2.密码
14                 var usrPwd=$(":password").val();
15
16                 //3.单选按钮
17                 var sex=$(":radio:checked").val();
18
19                 //4.学历
20                 var level=$("select option:selected").val();
21
22                 //5.爱好
23                 var checkedBox=$(":checkbox:checked");
24
25                 var hobby='';//没有隐式迭代
26                 checkedBox.each(function(i,dom){
27                     hobby+=$(dom).val();
28                 });
29
30                 //6.备注
31                 var remark=$("textarea").val();
32
33                 //7.头像
34                 var face=$(":file").val();
35
36                 //8.隐藏信息 :hidden与可见性选择器重复了,
37                 var info=$("input:hidden").val();
38             });
```

```

39 alert(usrName+","+usrPwd+","+sex+","+level+","+hobby+","+remark+","+face+","+
+info);
40     });
41     });
42     </script>
43     <body>
44         <form action="http://www.baidu.com" method="get">
45             <!--
46                 表单项中用户选择的或者输入的值统一按照value属性获取
47             -->
48             <p>用户名: <input type="text" name="uname" /></p>
49             <p>密码: <input type="password" name="upwd"/></p>
50             <p>性别: <input type="radio" name="sex" value="男"/>男<input
type="radio" name="sex" value="女"/>女</p>
51             <p>学历: <select name="level">
52                 <option value="本科">本科</option>
53                 <option value="大专">大专</option>
54                 <option value="高中">高中</option>
55                 <option value="文盲">文盲</option>
56             </select>
57             </p>
58             <p>爱好:
59             <input type="checkbox" name="hobby" value="绘画">绘画
60             <input type="checkbox" name="hobby" value="舞蹈">舞蹈
61             <input type="checkbox" name="hobby" value="拳击">拳击
62             <input type="checkbox" name="hobby" value="唱歌">唱歌
63             </p>
64             <p>备注: <textarea cols="50" name="remark" rows="10"></textarea>
65             </p>
66             <p>头像: <input type="file" name="face"/></p>
67             <p>看不见: <input type="hidden" name="hid" value="HelloWorld"/>
68             </p>
69             <p><input type="button" value="注册"/></p>
70         </form>
71     </body>
72 </html>

```

2 事件

- 1 | 引入键盘事件: keyup--键盘弹起 keydown--键盘按下 keypress--键盘按下并释放

2-1 keydown

keydown([[data],fn])	返回值:jQuery
概述	
当键盘或按钮被按下时, 发生 keydown 事件。	
注释: 如果在文档元素上进行设置, 则无论元素是否获得焦点, 该事件都会发生。	
参数	

2-2 keyup

keyup([[data],fn)] 返回值:jQuery

概述

当按钮被松开时, 发生 keyup 事件。它发生在当前获得焦点的元素上。

注释: 如果在文档元素上进行设置, 则无论元素是否获得焦点, 该事件都会发生。

2-3 keypress

keypress([[data],fn)] 返回值:jQuery

概述

当键盘或按钮被按下时, 发生 keypress 事件。

keypress 事件与 keydown 事件类似。当按钮被按下时, 会发生该事件。它发生在当前获得焦点的元素上。不过, 与 keydown 事件不同, 每插入一个字符, 就会发生 keypress 事件。注释: 如果在文档元素上进行设置, 则无论元素是否获得焦点, 该事件都会发生。

编号	姓名	年龄
3	lily熊	14
3	lily熊	14

键盘事件:
keydown
keyup
keypress

3 DOMのCRUD

3-1 创建节点

createElement() createTextNode()

```
1 | $(标签拼接str)
```

3-2 内部添加

特点: 父节点

```
1 | append(标签拼接str或$(标签拼接str))
2 | prepend()
3 | apppenTo()
4 | prependTo()
```

3-3 替换

js:replaceChild()

```
1 | 获取要替换你的节点.replacewith(新节点)
```

3-4 删除

```
1 删除一个：
2 获取要删除的节点.remove()
3 清空：
4 获取要删除的节点.empty();
```

3-5 外部追加

同级元素操作

```
1 after()
2 before()
3 insertAfter()
4 insertBefore()
```

课堂案例

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4         <meta charset="UTF-8">
5         <title></title>
6         <script src="js/jquery-1.8.3.js" type="text/javascript"></script>
7         <script type="text/javascript">
8             $(function(){
9                 //下标指定按钮
10                //1. 获取所有的按钮
11                var arrButtons=$(":button");
12                //2. 获取单个按钮，做click事件注册
13                arrButtons.eq(0);//下标从0开始
14                $(":button:eq(0)").click(function(){
15                    //prepend将新节点放在ul的第一个
16                    $("ul").prepend("<li><input type='checkbox'
17                                />aaaaaa</li>");
18                });
19                $(":button:eq(1)").click(function(){
20                    //append:尾部追加，等价于appendChild()
21                    //ul里面追加一个li，li的内容“aaaaa”
22                    //1. 创建节点
23                    var ob=$("<input type='checkbox' /><li>aaaaaa</li>");
24                    //2. 追加节点
25                    $("ul").append("<li><input type='checkbox'
26                                />aaaaaa</li>");
27                });
28                $(":button:eq(2)").click(function(){
29                    //prependTo:子节点.prependTo(ul)
30                    $("<li><input type='checkbox'
31                                />aaaaaa</li>").prependTo("ul");
32                });
33                $(":button:eq(3)").click(function(){
34                    //appendTo:子节点.appendTo(ul)
35                    $("<li><input type='checkbox'
36                                />aaaaaa</li>").appendTo("ul");
37                });
38            });
39        </script>
40    </head>
41    <body>
42        <ul>
43            <li><input type="checkbox" /> 1</li>
44            <li><input type="checkbox" /> 2</li>
45            <li><input type="checkbox" /> 3</li>
46            <li><input type="checkbox" /> 4</li>
47        </ul>
48    </body>
49 </html>
```

```

34
35     $(".button:eq(4)").click(function(){
36         //after:同级节点.prependTo(u1)
37         //获取沉鱼节点
38         var objLi=$("ul li:nth-child(3)");
39
40         //沉鱼后面添加bbbbbb
41         objLi.after("<li><input type='checkbox' />bbbbbb</li>")
42     });
43
44     $(".button:eq(5)").click(function(){
45         //before:同级节点.prependTo(u1)
46         $(".<li><input type='checkbox'
47 />aaaaaa</li>").prependTo("ul");
48     });
49     $(".button:eq(6)").click(function(){
50         //insertAfter:新增节点.insertAfter(沉鱼)
51         $(".<li><input type='checkbox'
52 />bbbbbb</li>").insertAfter($(".ul li:nth-child(3)"))
53     });
54
55     $(".button:eq(7)").click(function(){
56         //insertBefore:同级节点.prependTo(u1)
57         $(".<li><input type='checkbox'
58 />aaaaaa</li>").prependTo("ul");
59     });
60
61     $(".button:eq(8)").click(function(){
62         //克隆
63     });
64
65     $(".button:eq(9)").click(function(){
66         //替换
67         /*
68          * 要替换的节点.replaceWith(新节点)
69          * 扩展：基于选中的节点实现替换
70          */
71         //1. 获取选中的复选框,所在li替换新的li
72         var obj=$(":checkbox:checked").parent();
73         alert(obj.html());
74         //替换所有匹配的节点
75         obj.replaceWith("<li><input type='checkbox'
76 />cccc</li>")
77     });
78
79     $(".button:eq(10)").click(function(){
80         //删除
81         /*
82          * 删除所有的匹配项
83          * 删除：自己.remove()
84          */
85         $(".checkbox:checked").parent().remove();
86     });
87
88     $(".button:eq(11)").click(function(){
89         //清空 匹配的元素.empty()
90         if(confirm("确认清空吗? ")){

```

```

88         $("ul li").empty();
89     }
90 });
91 });
92 </script>
93 </head>
94 <body>
95     <ul>
96         <li class="li1"><input type="checkbox" />闭月</li>
97         <li><input type="checkbox" />羞花</li>
98         <li class="li1"><input type="checkbox" />沉鱼</li>
99         <li><input type="checkbox" />落雁</li>
100     </ul>
101     <fieldset>
102         <legend>添加节点的方法</legend>
103         <input type="button" name="" id="" value="prepend" />
104         <input type="button" name="" id="" value="append" />
105         <input type="button" name="" id="" value="prependTo" />
106         <input type="button" name="" id="" value="appendTo" />
107         <input type="button" name="" id="" value="after" />
108         <input type="button" name="" id="" value="before" />
109         <input type="button" name="" id="" value="insertAfter" />
110         <input type="button" name="" id="" value="insertBefore" />
111     </fieldset>
112     <input type="button" name="" id="" value="克隆" />
113     <input type="button" name="" id="" value="替换" />
114     <input type="button" name="" id="" value="删除" />
115     <input type="button" name="" id="" value="清空" />
116 </body>
117 </html>

```

课程总结

1 选择器：层级选择器 属性选择器 过滤选择器 表单选择器 以及表单属性过滤选择

2 DOM：创建、追加、插入、删除、清空、替换

预习安排

XML解析：DOM获取xml保存的数据 XML约束===了解

MYSQL:配置版本（手动创建，本质记事本修改后缀名 my.ini）