

今日内容

1. JQuery 基础:

1. 概念
2. 快速入门
3. JQuery对象和JS对象区别与转换
4. 选择器
5. DOM操作
6. 案例

JQuery 基础:

概念： 一个JavaScript框架。简化JS开发

- jQuery是一个快速、简洁的JavaScript框架，是继Prototype之后又一个优秀的JavaScript代码库（或JavaScript框架）。jQuery设计的宗旨是“write Less, Do More”，即倡导写更少的代码，做更多的事情。它封装JavaScript常用的功能代码，提供一种简便的JavaScript设计模式，优化HTML文档操作、事件处理、动画设计和Ajax交互。
- JavaScript框架：本质上就是一些js文件，封装了js的原生代码而已

快速入门

步骤:

下载JQuery

- 目前jQuery有三个大版本：
 - 1.x: 兼容ie678,使用最为广泛的，官方只做BUG维护，功能不再新增。因此一般项目来说，使用1.x版本就可以了，最终版本：1.12.4 (2016年5月20日)
 - 2.x: 不兼容ie678，很少有人使用，官方只做BUG维护，功能不再新增。如果不考虑兼容低版本的浏览器可以使用2.x，最终版本：2.2.4 (2016年5月20日)
 - 3.x: 不兼容ie678，只支持最新的浏览器。除非特殊要求，一般不会使用3.x版本的，很多老的jQuery插件不支持这个版本。目前该版本是官方主要更新维护的版本。最新版本：3.2.1 (2017年3月20日)
- jquery-xxx.js 与 jquery-xxx.min.js区别：
 1. jquery-xxx.js: 开发版本。给程序员看的，有良好的缩进和注释。体积大一些
 2. jquery-xxx.min.js: 生产版本。程序中使用，没有缩进。体积小一些。程序加载更快

导入JQuery的js文件：导入min.js文件

使用

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>JqueryDemo01</title>
  <script src="../../js/jquery-3.3.1.min.js"></script>
</head>
<body>
  <div id="div1">div1...</div>
  <div id="div2">div2...</div>

  <script>
    /*用jq来获取元素对象*/
    var div1 = $("#div1");
    var div2 = $("#div2");
  </script>

</body>
</html>
```

JQuery对象和JS对象区别与转换

JQuery对象在操作时，更加方便。

JQuery对象和js对象方法不通用的。

两者相互转换

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>JqueryDemo01</title>
  <script src="../../js/jquery-3.3.1.min.js"></script>
</head>
<body>
  <div id="div1">div1...</div>
  <div id="div2">div2...</div>

  <script>
    var divs = document.getElementsByTagName("div");
    var $divs = $("div");

    alert(divs);    //[object HTMLCollection], 可以当做数组来使用
    alert($divs);  //[object Object], 也可以当做数组来使用
  </script>
</body>
</html>
```

```

    /*但是使用jq要比js方便很多，比如将所有div文本设置*/
    /*js版设置*/
    for (var i=0;i<divs.length;i++){
        divs[i].innerHTML="1";
    }

    /*jq设置*/
    $divs.html("2");
</script>

</body>
</html>

```

1. jq对象在操作时更加方便。
2. jq对象和js对象方法不通用。
3. 虽然不通用，但是两者可以相互转换
 - jq --> js: jq对象[索引] 或者 jq对象.get(索引)
 - js --> jq: \$(js对象)

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>JqueryDemo01</title>
    <script src="../../js/jquery-3.3.1.min.js"></script>
</head>
<body>
    <div id="div1">div1...</div>
    <div id="div2">div2...</div>

    <script>
        var divs = document.getElementsByTagName("div");
        var $divs = $("div");

        /*js-->jq*/
        for (var i=0;i<divs.length;i++){
            $(divs[i]).html("3");
        }

        /*jq-->js*/
        $divs[0].innerHTML = "4";
        $divs.get(1).innerHTML = "5";
    </script>

</body>

```

```
</html>
```

选择器：筛选具有相似特征的元素(标签)

基本操作学习：

事件绑定

- `$("div").onclick(function () {});`

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>JqueryDemo01</title>
  <script src="../../js/jquery-3.3.1.min.js"></script>
</head>
<body>
  <div id="div1">div1...</div>
  <div id="div2">div2...</div>

  <script>

    $("div").onclick(function () {});

  </script>

</body>
</html>
```

入口函数

- `$(function () {});`

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>JqueryDemo01</title>
  <script src="../../js/jquery-3.3.1.min.js"></script>
</head>
<body>
  <div id="div1">div1...</div>
  <div id="div2">div2...</div>

  <script>
```

```
        $(function () {

        });

    </script>

</body>
</html>
```

- window.onload和\$(function)都可以实现等待页面(dom文档)加载之后才使用script代码的作用
- window.onload 和 \$(function) 区别
 - window.onload 只能定义一次,如果定义多次,后边的会将前边的覆盖掉
 - \$(function)可以定义多次的。

样式控制: css方法

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>JqueryDemo01</title>
    <script src="../../js/jquery-3.3.1.min.js"></script>
</head>
<body>
    <div id="div1">div1...</div>
    <div id="div2">div2...</div>

    <script>

        $(function () {
            /*写法一*/
            $("#div1").css("background-color", "red");
            /*写法二*/
            $("#div2").css("backgroundColor", "blue");
        });

    </script>

</body>
</html>
```

分类

基本选择器

标签选择器 (元素选择器)

- 语法: \$("html标签名") 获得所有匹配标签名称的元素

id选择器

- 语法: `$("#id的属性值")` 获得与指定id属性值匹配的元素

类选择器

- 语法: `$(".class的属性值")` 获得与指定的class属性值匹配的元素

并集选择器:

- 语法: `$("选择器1,选择器2....")` 获取多个选择器选中的所有元素

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>基本选择器</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <script src="../js/jquery-3.3.1.min.js"></script>
    <style type="text/css">
      div,span{
        width: 180px;
        height: 180px;
        margin: 20px;
        background: #9999CC;
        border: #000 1px solid;
        float:left;
        font-size: 17px;
        font-family:Roman;
      }

      div .mini{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
      }

      div .mini01{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
      }

    </style>
    <script type="text/javascript">
```

```

        // <input type="button" value="保存" class="mini" name="ok"
class="mini" />
        $(function () {

            });
        // <input type="button" value="改变 id 为 one 的元素的背景色为 红色"
id="b1"/>
        $(function () {
            $("#b1").click(function () {
                $("#one").css("backgroundColor", "red");
            });
        });
        // <input type="button" value=" 改变元素名为 <div> 的所有元素的背景色为 红色"
id="b2"/>
        $(function () {
            $("#b2").click(function () {
                $("div").css("backgroundColor", "red");
            });
        });
        // <input type="button" value=" 改变 class 为 mini 的所有元素的背景色为 红色"
id="b3"/>
        $(function () {
            $("#b3").click(function () {
                $(".mini").css("backgroundColor", "red");
            });
        });
        // <input type="button" value=" 改变所有的<span>元素和 id 为 two 的元素的背景
色为红色" id="b4"/>
        $(function () {
            $("#b4").click(function () {
                $("span, #two").css("backgroundColor", "red");
            });
        });

</script>

</head>

<body>

        <input type="button" value="保存" class="mini" name="ok" class="mini"
/>

        <input type="button" value="改变 id 为 one 的元素的背景色为 红色"
id="b1"/>
        <input type="button" value=" 改变元素名为 <div> 的所有元素的背景色为 红色"
id="b2"/>
        <input type="button" value=" 改变 class 为 mini 的所有元素的背景色为 红色"
id="b3"/>
        <input type="button" value=" 改变所有的<span>元素和 id 为 two 的元素的背景
色为红色" id="b4"/>

```

```

<h1>有一种奇迹叫坚持</h1>
<h2>自信源于努力</h2>

<div id="one">
  id为one
</div>

<div id="two" class="mini" >
  id为two   class是 mini
  <div class="mini" >class是 mini</div>
</div>

<div class="one" >
  class是 one
  <div class="mini" >class是 mini</div>
  <div class="mini" >class是 mini</div>
</div>
<div class="one" >
  class是 one
  <div class="mini01" >class是 mini01</div>
  <div class="mini" >class是 mini</div>
</div>

<span class="spanone">class为spanone的span元素</span>
<span class="mini">class为mini的span元素</span>

<input type="text" value="zhang" id="username" name="username">

</body>

</html>

```

层级选择器

后代选择器

- 语法：\$("标签A 标签B") 选择A元素内部的所有B元素，包括所有小辈

子选择器

- 语法：\$("标签A > 标签B") 选择A元素内部的所有B子元素，或者说是A元素下面的直接的子元素，只包括儿子

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>层次选择器</title>

```



```

<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<script src="../../js/jquery-3.3.1.min.js"></script>
<style type="text/css">
    div,span{
        width: 180px;
        height: 180px;
        margin: 20px;
        background: #9999CC;
        border: #000 1px solid;
        float:left;
        font-size: 17px;
        font-family:Roman;
    }

    div .mini{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
    }

    div .mini01{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
    }

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

    // <input type="button" value="保存" class="mini" name="ok"
class="mini" />

    // <input type="button" value=" 改变 <body> 内所有 <div> 的背景色为红色"
id="b1"/>
    $(function () {
        $("#b1").click(function () {
            $("body div").css("backgroundColor","red");
        });
    });

    // <input type="button" value=" 改变 <body> 内子 <div> 的背景色为 红色"
id="b2"/>
    $(function () {
        $("#b2").click(function () {
            $("body>div").css("backgroundColor","red");
        });
    });

```

```

    });
</script>

</head>

<body>

    <input type="button" value="保存" class="mini" name="ok" class="mini"
/>
    <input type="button" value=" 改变 <body> 内所有 <div> 的背景色为红色"
id="b1"/>
    <input type="button" value=" 改变 <body> 内子 <div> 的背景色为 红色"
id="b2"/>


    <h1>有一种奇迹叫坚持</h1>
    <h2>自信源于努力</h2>


    <div id="one">
        id为one

    </div>


    <div id="two" class="mini" >
        id为two class是 mini
        <div class="mini" >class是 mini</div>
    </div>


    <div class="one" >
        class是 one
        <div class="mini" >class是 mini</div>
        <div class="mini" >class是 mini</div>
    </div>
    <div class="one">
        class是 one
        <div class="mini01" >class是 mini01</div>
        <div class="mini" >class是 mini</div>
    </div>


    <span class="spanone"> span
</span>


</body>

</html>

```

属性选择器

属性名称选择器

- 语法: `$("标签A[属性名]")` 包含指定属性的选择器

属性选择器

- 语法: `$("标签A[属性名='值']")` 包含指定属性等于指定值的选择器
- 语法: `$("标签A[属性名!='值']")` 包含指定属性不等于指定值的选择器
- 语法: `$("标签A[属性名^='值']")` 包含指定属性以指定值为开头的选择器(正则中以^开始, 以\$结束)
- 语法: `$("标签A[属性名$='值']")` 包含指定属性以指定值为结尾的选择器(正则中以^开始, 以\$结束)
- 语法: `$("标签A[属性名*='值']")` 包含指定属性以包含指定值的选择器

复合属性选择器

- 语法: `$("标签A[属性名='值'][...]...")` 包含多个属性条件的选择器

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>属性过滤选择器</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <script src="../js/jquery-3.3.1.min.js"></script>
    <style type="text/css">
      div,span{
        width: 180px;
        height: 180px;
        margin: 20px;
        background: #9999CC;
        border: #000 1px solid;
        float:left;
        font-size: 17px;
        font-family:Roman;
      }

      div .mini{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
      }

      div .mini01{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
      }
    </style>
  </head>
  <body>
    <div>
      <div class="mini">
        1
      </div>
      <div class="mini01">
        2
      </div>
    </div>
  </body>
</html>
```

```

        div.visible{
            display:none;
        }
    </style>
    <script type="text/javascript">
    // <input type="button" value="保存" class="mini" name="ok" class="mini"
    />

    // <input type="button" value=" 含有属性title 的div元素背景色为红色" id="b1"/>
    $(function () {
        $("#b1").click(function () {
            $("div[title]").css("backgroundColor", "red");
        });
    });

    // <input type="button" value=" 属性title值等于test的div元素背景色为红色"
    id="b2"/>
    $(function () {
        $("#b2").click(function () {
            $("div[title='test']").css("backgroundColor", "red");
        });
    });

    // <input type="button" value=" 属性title值不等于test的div元素 (没有属性title的也
    将被选中) 背景色为红色" id="b3"/>
    $(function () {
        $("#b3").click(function () {
            $("div[title!='test']").css("backgroundColor", "red");
        });
    });

    // <input type="button" value=" 属性title值 以te开始 的div元素背景色为红色"
    id="b4"/>
    $(function () {
        $("#b4").click(function () {
            $("div[title^='te']").css("backgroundColor", "red");
        });
    });

    // <input type="button" value=" 属性title值 以est结束 的div元素背景色为红色"
    id="b5"/>
    $(function () {
        $("#b5").click(function () {
            $("div[title$='est']").css("backgroundColor", "red");
        });
    });

    // <input type="button" value="属性title值 含有es的div元素背景色为红色"
    id="b6"/>
    $(function () {
        $("#b6").click(function () {
            $("div[title*='es']").css("backgroundColor", "red");
        });
    });

```

```

    });
    // <input type="button" value="选取有属性id的div元素，然后在结果中选取属性title值
    含有“es”的 div 元素背景色为红色" id="b7"/>
    $(function () {
        $("#b7").click(function () {
            $("div[id][title*='es']").css("backgroundColor", "red");
        });
    });

</script>

</head>

<body>

    <input type="button" value="保存" class="mini" name="ok" class="mini"
/>
    <input type="button" value=" 含有属性title 的div元素背景色为红色"
id="b1"/>
    <input type="button" value=" 属性title值等于test的div元素背景色为红色"
id="b2"/>
    <input type="button" value=" 属性title值不等于test的div元素 (没有属性title的
也将被选中) 背景色为红色" id="b3"/>
    <input type="button" value=" 属性title值 以te开始 的div元素背景色为红色"
id="b4"/>
    <input type="button" value=" 属性title值 以est结束 的div元素背景色为红色"
id="b5"/>
    <input type="button" value="属性title值 含有es的div元素背景色为红色"
id="b6"/>
    <input type="button" value="选取有属性id的div元素，然后在结果中选取属性title
值含有“es”的 div 元素背景色为红色" id="b7"/>

    <div id="one">
        id为一one div
    </div>

    <div id="two" class="mini" title="test">
        id为two class是 mini div title="test"
        <div class="mini" >class是 mini</div>
    </div>

    <div class="visible" >
        class是 one
        <div class="mini" >class是 mini</div>
        <div class="mini" >class是 mini</div>
    </div>
    <div class="one" title="test02">
        class是 one title="test02"

```

```
        <div class="mini01" >class是 mini01</div>
        <div class="mini" style="margin-top:0px;">class是 mini</div>
    </div>

    <div class="visible" >
        这是隐藏的
    </div>

    <div class="one">

    </div>

    <div id="mover" >
        动画
    </div>

    <input type="text" value="zhang" id="username" name="username">
</body>

</html>
```

过滤选择器

首元素选择器

- 语法： 标签:first 获得选择的元素中的第一个元素

尾元素选择器

- 语法： 标签:last 获得选择的元素中的最后一个元素

非元素选择器

- 语法： 标签 :not(selector) 不包括指定内容的元素

偶数选择器

- 语法： 标签:even 偶数，从 0 开始计数

奇数选择器

- 语法： 标签:odd 奇数，从 0 开始计数

等于索引选择器

- 语法： 标签:eq(index) 指定索引元素

大于索引选择器

- 语法： 标签 :gt(index) 大于指定索引元素

小于索引选择器

- 语法： 标签 :lt(index) 小于指定索引元素

标题选择器

- 语法： 标签:header 获得标题 (h1~h6) 元素，固定写法

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>基本过滤选择器</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <script src="../../js/jquery-3.3.1.min.js"></script>
    <style type="text/css">
      div,span{
        width: 180px;
        height: 180px;
        margin: 20px;
        background: #9999CC;
        border: #000 1px solid;
        float:left;
        font-size: 17px;
        font-family:Roman;
      }

      div .mini{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
      }

      div .mini01{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
      }
    </style>
    <script type="text/javascript">

      // <input type="button" value="保存" class="mini" name="ok"
      class="mini" />

      // <input type="button" value=" 改变第一个 div 元素的背景色为 红色"
      id="b1"/>

      $(function () {
        $("#b1").click(function () {
          $("div:first").css("backgroundColor","red");
        });
      });
    </script>
  </head>
  <body>
    <div>
      <div class="mini">
        <div class="mini01">
          <input type="button" value="保存" class="mini" name="ok" class="mini" />
          <input type="button" value=" 改变第一个 div 元素的背景色为 红色" id="b1" />
        </div>
      </div>
    </div>
  </body>
</html>
```

```
});

// <input type="button" value=" 改变最后一个 div 元素的背景色为 红色"
id="b2"/>
$(function () {
    $("#b2").click(function () {
        $("div:last").css("backgroundColor", "red");
    });
});

// <input type="button" value=" 改变class不为 one 的所有 div 元素的背景色为
红色" id="b3"/>
$(function () {
    $("#b3").click(function () {
        $("div:not(.one)").css("backgroundColor", "red");
    });
});

// <input type="button" value=" 改变索引值为偶数的 div 元素的背景色为 红色"
id="b4"/>
$(function () {
    $("#b4").click(function () {
        $("div:even").css("backgroundColor", "red");
    });
});

// <input type="button" value=" 改变索引值为奇数的 div 元素的背景色为 红色"
id="b5"/>
$(function () {
    $("#b5").click(function () {
        $("div:odd").css("backgroundColor", "red");
    });
});

// <input type="button" value=" 改变索引值为大于 3 的 div 元素的背景色为 红色"
id="b6"/>
$(function () {
    $("#b6").click(function () {
        $("div:gt(3)").css("backgroundColor", "red");
    });
});

// <input type="button" value=" 改变索引值为等于 3 的 div 元素的背景色为 红色"
id="b7"/>
$(function () {
    $("#b7").click(function () {
        $("div:eq(3)").css("backgroundColor", "red");
    });
});
```



```

        // <input type="button" value=" 改变索引值为小于 3 的 div 元素的背景色为 红色"
id="b8"/>
        $(function () {
            $("#b8").click(function () {
                $("div:lt(3)").css("backgroundColor", "red");
            });
        });

        // <input type="button" value=" 改变所有的标题元素的背景色为 红色" id="b9"/>
        $(function () {
            $("#b9").click(function () {
                $(".:header").css("backgroundColor", "red");
            });
        });

</script>
</head>

<body>

        <input type="button" value="保存" class="mini" name="ok" class="mini"
/>

        <input type="button" value=" 改变第一个 div 元素的背景色为 红色" id="b1"/>
        <input type="button" value=" 改变最后一个 div 元素的背景色为 红色"
id="b2"/>
        <input type="button" value=" 改变class不为 one 的所有 div 元素的背景色为 红
色" id="b3"/>
        <input type="button" value=" 改变索引值为偶数的 div 元素的背景色为 红色"
id="b4"/>
        <input type="button" value=" 改变索引值为奇数的 div 元素的背景色为 红色"
id="b5"/>
        <input type="button" value=" 改变索引值为大于 3 的 div 元素的背景色为 红色"
id="b6"/>
        <input type="button" value=" 改变索引值为等于 3 的 div 元素的背景色为 红色"
id="b7"/>
        <input type="button" value=" 改变索引值为小于 3 的 div 元素的背景色为 红色"
id="b8"/>
        <input type="button" value=" 改变所有的标题元素的背景色为 红色" id="b9"/>


        <h1>有一种奇迹叫坚持</h1>
        <h2>自信源于努力</h2>


        <div id="one">
            id为one

        </div>


        <div id="two" class="mini" >
            id为two class是 mini

```

```

        <div class="mini" >class是 mini</div>
    </div>

    <div class="one" >
        class是 one
        <div class="mini" >class是 mini</div>
        <div class="mini" >class是 mini</div>
    </div>
    <div class="one" >
        class是 one
        <div class="mini01" >class是 mini01</div>
        <div class="mini" >class是 mini</div>
    </div>

</body>

</html>

```

表单过滤选择器

可用元素选择器

- 语法：:enabled 获得可用元素

不可用元素选择器

- 语法：:disabled 获得不可用元素

选中选择器

- 语法：:checked 获得单选/复选框选中的元素

选中选择器

- 语法：:selected 获得下拉框选中的元素

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>表单属性过滤选择器</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <script src="../js/jquery-3.3.1.min.js"></script>
    <style type="text/css">
      div,span{
        width: 180px;
        height: 180px;
        margin: 20px;
        background: #9999CC;
        border: #000 1px solid;
        float:left;
        font-size: 17px;

```

```

        font-family:Roman;
    }

    div .mini{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
    }

    div .mini01{
        width: 50px;
        height: 50px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
    }
    #job{
        margin: 20px;
    }
    #edu{
        margin-top:-70px;
    }

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

    // <input type="button" value="保存" class="mini" name="ok"
class="mini" />

    // <input type="button" value=" 利用 jQuery 对象的 val() 方法改变表单内可用
<input> 元素的值" id="b1"/> //其实就是改变输入框中的值
    $(function () {
        $("#b1").click(function () {
            $("input[type='text']:enabled").val("aaa");
        });
    });

    // <input type="button" value=" 利用 jQuery 对象的 val() 方法改变表单内不可用
<input> 元素的值" id="b2"/>
    $(function () {
        $("#b2").click(function () {
            $("input[type='text']:disabled").val("bbb");
        });
    });

    // <input type="button" value=" 利用 jQuery 对象的 length 属性获取复选框选中
的个数" id="b3"/>

```

```

$(function () {
    $("#b3").click(function () {
        alert($("#input[name='items']:checked").length);
    });
});

// <input type="button" value=" 利用 jQuery 对象的 length 属性获取下拉框选中
的个数" id="b4"/>
$(function () {
    $("#b4").click(function () {
        alert($("#select[name='edu'] > option:selected").length);
    });
});

</script>
</head>

<body>

    <input type="button" value="保存" class="mini" name="ok" class="mini"
/>

    <input type="button" value=" 利用 jQuery 对象的 val() 方法改变表单内可用
<input> 元素的值" id="b1"/>
    <input type="button" value=" 利用 jQuery 对象的 val() 方法改变表单内不可用
<input> 元素的值" id="b2"/>
    <input type="button" value=" 利用 jQuery 对象的 length 属性获取复选框选中的
个数" id="b3"/>
    <input type="button" value=" 利用 jQuery 对象的 length 属性获取下拉框选中的
个数" id="b4"/>

    <br><br>

    <input type="text" value="不可用值1" disabled="disabled">
    <input type="text" value="可用值1" >
    <input type="text" value="不可用值2" disabled="disabled">
    <input type="text" value="可用值2" >

    <br><br>
    <input type="checkbox" name="items" value="美容" >美容
    <input type="checkbox" name="items" value="IT" >IT
    <input type="checkbox" name="items" value="金融" >金融
    <input type="checkbox" name="items" value="管理" >管理

    <br><br>

    <input type="radio" name="sex" value="男" >男
    <input type="radio" name="sex" value="女" >女

    <br><br>

```

```

<select name="job" id="job" multiple="multiple" size=4>
  <option>程序员</option>
  <option>中级程序员</option>
  <option>高级程序员</option>
  <option>系统分析师</option>
</select>

<select name="edu" id="edu">
  <option>本科</option>
  <option>博士</option>
  <option>硕士</option>
  <option>大专</option>
</select>

<br/>

<div id="two" class="mini" >
  id为two class是 mini div
  <div class="mini" >class是 mini</div>
</div>

<div class="one" >
  class是 one
  <div class="mini" >class是 mini</div>
  <div class="mini" >class是 mini</div>
</div>
<div class="one" >
  class是 one
  <div class="mini01" >class是 mini01</div>
  <div class="mini" >class是 mini</div>
</div>

</body>

</html>

```

DOM操作

内容操作

html(): 获取/设置元素的标签体内容

```
<a><font>内容</font></a> --> <font>内容
```

text(): 获取/设置元素的标签体纯文本内容

```
<a><font>内容</font></a> --> 内容
```

val(): 获取/设置元素的value属性值

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title></title>
    <script src="../../js/jquery-3.3.1.min.js"></script>
    <script>

      $(function () {
        /*获取myinput 的value值
        * <input id="myinput" type="text" name="username" value="张三" /><br
        */
        var val = $("#myinput").val();

        /*设置myinput 的value值
        * <input id="myinput" type="text" name="username" value="张三" /><br
        */
        $("#myinput").val("李四");

        /*获取mydiv的标签体内容
        * <div id="mydiv"><p><a href="#">标题标签</a></p></div>
        * -->
        * <p><a href="#">标题标签</a></p>
        * */
        var html = $("#mydiv").html();

        /*设置mydiv 的value值
        * <div id="mydiv"><p><a href="#">标题标签</a></p></div>
        * -->
        * <div id="mydiv">123</div>
        * */
        $("#mydiv").html("123");

        /*获取mydiv文本内容
        * <div id="mydiv"><p><a href="#">标题标签</a></p></div>
        * -->
        * 标题标签
        * */
        var text = $("#mydiv").text();
```

```

        /*设置mydiv文本内容
        *<div id="mydiv"><p><a href="#">标题标签</a></p></div>
        * -->
        * <div id="mydiv">123</div>
        * */
        $("#mydiv").text("123");
    });

</script>

</head>
<body>
    <input id="myinput" type="text" name="username" value="张三" /><br />
    <div id="mydiv"><p><a href="#">标题标签</a></p></div>
</body>
</html>

```

属性操作

通用属性操作

attr(): 获取/设置元素的属性

removeAttr(): 删除属性

prop(): 获取/设置元素的属性

removeProp(): 删除属性

- attr和prop区别?
 1. 如果操作的是元素的固有属性，则建议使用prop
 2. 如果操作的是元素自定义的属性，则建议使用attr

如果不遵守以上两种规则，那么可能会有一些属性是获取不到，会是undefined，比如selected

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>获取属性</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <script src="../js/jquery-3.3.1.min.js"></script>

    <style type="text/css">
      div,span{
        width: 140px;
        height: 140px;
        margin: 20px;
        background: #9999CC;
        border: #000 1px solid;
      }
    </style>
  </head>
  <body>
    <div id="mydiv">
      <p><a href="#">标题标签</a></p>
    </div>
  </body>
</html>

```

```

        float:left;
        font-size: 17px;
        font-family:Roman;
    }

    div.mini{
        width: 30px;
        height: 30px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
    }

    div.visible{
        display:none;
    }
</style>

<script type="text/javascript">

    $(function () {
        //获取北京节点的name属性值
        var attr = $("#bj").attr("name");
        //设置北京节点的name属性的值为dabeijing
        $("#bj").attr("name","dabeijing");
        //新增北京节点的discription属性 属性值是didu
        $("#bj").attr("discription","didu");
        //删除北京节点的name属性并检验name属性是否存在
        $("#bj").removeAttr("name");
        //获得hobby的选中状态
        var checked = $("#hobby").prop("checked");

    });

</script>

</head>

<body>

    <ul>
        <li id="bj" name="beijing" xxx="yyy">北京</li>
        <li id="tj" name="tianjin">天津</li>
    </ul>
    <input type="checkbox" id="hobby"/>

</body>
</html>

```


对class属性操作

addClass():添加class属性值

removeClass():删除class属性值

toggleClass():切换class属性

- toggleClass("one"):
- 判断如果元素对象上存在class="one", 则将属性值one删除掉。 如果元素对象上不存在class="one", 则添加

css():

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>样式操作</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
    <script src="../../js/jquery-3.3.1.min.js"></script>
    <style type="text/css">
      .one{
        width: 200px;
        height: 140px;
        margin: 20px;
        background: red;
        border: #000 1px solid;
        float:left;
        font-size: 17px;
        font-family:Roman;
      }

      div,span{
        width: 140px;
        height: 140px;
        margin: 20px;
        background: #9999CC;
        border: #000 1px solid;
        float:left;
        font-size: 17px;
        font-family:Roman;
      }

      div .mini{
        width: 40px;
        height: 40px;
        background: #CC66FF;
```

```

        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
    }
    div .mini01{
        width: 40px;
        height: 40px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
    }

    /*待用的样式*/
    .second{
        width: 300px;
        height: 340px;
        margin: 20px;
        background: yellow;
        border: pink 3px dotted;
        float:left;
        font-size: 22px;
        font-family:Roman;
    }

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

    $(function () {

        //<input type="button" value="采用属性增加样式(改变id=one的样式)"
id="b1"/>

        $("#b1").click(function () {
            $("#one").prop("class", "second");
        });

        //<input type="button" value=" addClass" id="b2"/>
        $("#b2").click(function () {
            $("#one").addClass("second")
        });

        //<input type="button" value="removeClass" id="b3"/>
        $("#b3").click(function () {
            $("#one").removeClass("second");
        });

        //<input type="button" value=" 切换样式" id="b4"/>
        $("#b4").click(function () {

```



```
<div class="one" >
  class是 one
  <div class="mini01" >class是 mini01</div>
  <div class="mini" >class是 mini</div>
</div>

<span class="spanone">    span
</span>

</body>

</html>
```

CRUD操作:

append():父元素将子元素追加到末尾

- 对象1.append(对象2): 对象2-->对象1元素内部, 并且在末尾

prepend():父元素将子元素追加到开头

- 对象1.prepend(对象2):对象2-->对象1元素内部, 并且在开头

appendTo():

- 对象1.appendTo(对象2):对象1-->对象2内部, 并且在末尾

prependTo():

- 对象1.prependTo(对象2):对象1-->对象2内部, 并且在开头

after():添加元素到元素后边

- 对象1.after(对象2): 对象2-->对象1后面。对象1和对象2是兄弟关系

before():添加元素到元素前边

- 对象1.before(对象2): 对象2-->对象1前边。对象1和对象2是兄弟关系

insertAfter()

- 对象1.insertAfter(对象2): 对象1-->对象2后边。对象1和对象2是兄弟关系

insertBefore()

- 对象1.insertBefore(对象2): 对象1-->对象2前边。对象1和对象2是兄弟关系

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

```
<html>
<head>
  <title>内部插入脚本</title>
  <meta http-equiv="content-type" content="text/html; charset=UTF-8">
  <script src="../../js/jquery-3.3.1.min.js"></script>

  <style type="text/css">
    div,span{
      width: 140px;
      height: 140px;
      margin: 20px;
      background: #9999CC;
      border: #000 1px solid;
      float:left;
      font-size: 17px;
      font-family:Roman;
    }

    div .mini{
      width: 30px;
      height: 30px;
      background: #CC66FF;
      border: #000 1px solid;
      font-size: 12px;
      font-family:Roman;
    }

    div.visible{
      display:none;
    }
  </style>

</head>

<body>

<script type="text/javascript">

  $(function () {

    // <input type="button" value="将反恐放置到city的后面" id="b1"/>
    $("#b1").click(function () {
      $("#fk").appendTo("#city");
    });
    // <input type="button" value="将反恐放置到city的最前面" id="b2"/>
    $("#b2").click(function () {
      $("#fk").prependTo("#city");
    });
    // <input type="button" value="将反恐插入到天津后面" id="b3"/>
```

```

        $("#b3").click(function () {
            $("#fk").insertAfter($("#tj"));
        });
        // <input type="button" value="将反恐插入到天津前面" id="b4"/>
        $("#b4").click(function () {
            $("#fk").insertBefore($("#tj"));
        });
    });

</script>

<input type="button" value="将反恐放置到city的后面" id="b1"/>
<input type="button" value="将反恐放置到city的最前面" id="b2"/>
<input type="button" value="将反恐插入到天津后面" id="b3"/>
<input type="button" value="将反恐插入到天津前面" id="b4"/>
<ul id="city">
    <li id="bj" name="beijing">北京</li>
    <li id="tj" name="tianjin">天津</li>
    <li id="cq" name="chongqing">重庆</li>
</ul>

<ul id="love">
    <li id="fk" name="fankong">反恐</li>
    <li id="xj" name="xingji">星际</li>
</ul>

<div id="fool">Hello1</div>

</body>

</html>

```

remove():移除元素

- 对象.remove():将对象删除掉

empty():清空元素的所有后代元素。

- 对象.empty():将对象的后代元素全部清空，但是保留当前对象以及其属性节点

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head>
    <title>删除节点</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">

```

```

<script src="../../js/jquery-3.3.1.min.js"></script>
<style type="text/css">
    div,span{
        width: 140px;
        height: 140px;
        margin: 20px;
        background: #9999CC;
        border: #000 1px solid;
        float:left;
        font-size: 17px;
        font-family:Roman;
    }

    div.mini{
        width: 30px;
        height: 30px;
        background: #CC66FF;
        border: #000 1px solid;
        font-size: 12px;
        font-family:Roman;
    }

    div.visible{
        display:none;
    }
</style>
<script type="text/javascript">

    $(function () {

        // <input type="button" value="删除<li id='bj' name='beijing'>北京
</li>" id="b1"/>
        $("#b1").click(function () {
            $("#bj").remove();
        });
        // <input type="button" value="删除city所有的li节点 清空元素中的所有后
代节点 (不包含属性节点)" id="b2"/>
        $("#b2").click(function () {
            $("#city").empty();
        });
    });

</script>
</head>

<body>

```

```

        <input type="button" value="删除"><li id='bj' name='beijing'>北京</li>"
        id="b1"/>
        <input type="button" value="删除所有的子节点    清空元素中的所有后代节点 (不包含属性节
        点)"    id="b2"/>

        <ul id="city">
            <li id="bj" name="beijing">北京</li>
            <li id="tj" name="tianjin">天津</li>
            <li id="cq" name="chongqing">重庆</li>
        </ul>
        <p class="hello">Hello</p> how are <p>you?</p>
    </body>

</html>

```

案例

```

<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
        <script src="../../js/jquery-3.3.1.min.js"></script>

        <script>
            //需求：将数据行的奇数行背景色设置为 pink，偶数行背景色设置为 yellow
            $(function () {
                $("tr:gt(1):odd").css("backgroundColor", "pink");
                $("tr:gt(1):even").css("backgroundColor", "yellow");
            });
        </script>
    </head>
    <body>
        <table id="tab1" border="1" width="800" align="center" >
            <tr>
                <td colspan="5"><input type="button" value="删除"></td>
            </tr>
            <tr style="background-color: #999999;">
                <th><input type="checkbox"></th>
                <th>分类ID</th>
                <th>分类名称</th>
                <th>分类描述</th>
                <th>操作</th>
            </tr>

```



```

        <tr>
            <td><input type="checkbox"></td>
            <td>1</td>
            <td>手机数码</td>
            <td>手机数码类商品</td>
            <td><a href="">修改</a>|<a href="">删除</a></td>
        </tr>
        <tr>
            <td><input type="checkbox"></td>
            <td>2</td>
            <td>电脑办公</td>
            <td>电脑办公类商品</td>
            <td><a href="">修改</a>|<a href="">删除</a></td>
        </tr>
        <tr>
            <td><input type="checkbox"></td>
            <td>3</td>
            <td>鞋靴箱包</td>
            <td>鞋靴箱包类商品</td>
            <td><a href="">修改</a>|<a href="">删除</a></td>
        </tr>
        <tr>
            <td><input type="checkbox"></td>
            <td>4</td>
            <td>家居饰品</td>
            <td>家居饰品类商品</td>
            <td><a href="">修改</a>|<a href="">删除</a></td>
        </tr>
    </table>
</body>
</html>

```

```

<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
        <script src="../../js/jquery-3.3.1.min.js"></script>

    </head>
    <body>

        <table id="tab1" border="1" width="800" align="center" >
            <tr>
                <td colspan="5"><input type="button" value="删除"></td>
            </tr>

```

```

<tr>
  <th><input type="checkbox" onclick="selectAll(this)" ></th>
  <th>分类ID</th>
  <th>分类名称</th>
  <th>分类描述</th>
  <th>操作</th>
</tr>
<tr>
  <td><input type="checkbox" class="itemSelect"></td>
  <td>1</td>
  <td>手机数码</td>
  <td>手机数码类商品</td>
  <td><a href="">修改</a>|<a href="">删除</a></td>
</tr>
<tr>
  <td><input type="checkbox" class="itemSelect"></td>
  <td>2</td>
  <td>电脑办公</td>
  <td>电脑办公类商品</td>
  <td><a href="">修改</a>|<a href="">删除</a></td>
</tr>
<tr>
  <td><input type="checkbox" class="itemSelect"></td>
  <td>3</td>
  <td>鞋靴箱包</td>
  <td>鞋靴箱包类商品</td>
  <td><a href="">修改</a>|<a href="">删除</a></td>
</tr>
<tr>
  <td><input type="checkbox" class="itemSelect"></td>
  <td>4</td>
  <td>家居饰品</td>
  <td>家居饰品类商品</td>
  <td><a href="">修改</a>|<a href="">删除</a></td>
</tr>
</table>

<script>
  function selectAll(obj) {
    $("input[class=itemSelect]").prop("checked", obj.checked);
  }

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

```

```

<!DOCTYPE html>
<html>
<head>

```

```

<meta charset="UTF-8" />
<title>QQ表情选择</title>
<script src="../../js/jquery-3.3.1.min.js"></script>
<style type="text/css">
*{margin: 0;padding: 0;list-style: none;}

.emoji{margin:50px;}
ul{overflow: hidden;}
li{float: left;width: 48px;height: 48px;cursor: pointer;}
.emoji img{ cursor: pointer; }
</style>
<script>
    //需求: 点击qq表情, 将其追加到发言框中
    $(function () {
        $("ul img").click(function () {
            /*
             * 注意了, this对象是一个js, 不能使用jq的方法, 所以要转换为jq
             * */
            $(this).clone().appendTo($(".p"));
        });
    });

</script>

</head>
<body>
    <div class="emoji">
        <ul>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
            <li></li>
        </ul>
        <p class="word">
            <strong>请发言: </strong>
            
        </p>
    </div>
</body>
</html>

```



```
        <option>李四</option>
        <option>王五</option>
        <option>赵六</option>
    </select>
    <div id="btn">
        <input type="button" id="toRight" value="-->"><br>
        <input type="button" id="toLeft" value="<--">

    </div>

    <select id="rightName" multiple="multiple">
        <option>钱七</option>
    </select>

</div>

</body>
</html>
```