

jQuery 入门

jQuery入门：

查看官网

1、jquery的概念

问题：

使用js完成对网页的操作，代码的书写量比较大。js中对网页操作提供的方式过于简单，造成获取和操作HTML元素对象过于麻烦。

解决：

使用jQuery

结论：

其实jQuery就是讲常用的js操作封装了起来。jQuery是js的一个框架。

内容：

jQuery本质上还是js，基本语法和js一致。将js的DOM操作重新封装整合了。

学习：

如何使用jQuery完成网页操作。

2、jquery的特点

jQuery是js的一个框架

jQuery提供了多种多样的选择器

jQuery支持主流的各种版本的浏览器

jQuery使用起来特别简单

3、jQuery的使用

jQuery的选择器

jQuery的DOM操作

jQuery的事件和动画

jQuery的表单验证

jQuery 的基本选择器

基本选择器：

```
<html>
  <head>

    <title>jquery的选择器学习</title>

    <meta charset="UTF-8"/>
    <!--
```

jQuery的选择器学习：

作用：获取要操作的HTML元素对象

使用：

基本选择器：

全部选择器*

ID选择器：

`$("#id名")` 返回的是存储了指定的HTML元素对

象的数组

标签选择器：

`$("标签名")` 返回的是存储了指定标签的数组

类选择器：

`$(".类选择器")` 返回的是使用了相同类选择器

的HTML元素对象

组合选择器：

`$("选择器，选择器，选择器，.....")` 返回的是

所有选择器所指定的HTML元素对象。

层级选择器

简单选择器

可见性选择器

属性选择器

子元素选择器

表单选择器

表单对象属性选择器

注意：

jQuery中的选择器返回的是存储了指定的HTML元素对象的数组。

```
-->
<!--引入jQuery-->
<script src="js/jquery-1.9.1.js" type="text/javascript"
charset="utf-8"></script>
<!--声明js代码域-->
<script type="text/javascript">
    //基本选择器

    //id选择器
    function testById(){
        var un=$("#uname");//
        alert(un.val());
    }

    //标签选择器
    function testByTag(){
        var inps$("input");
        alert(inps.length);
    }

    //类选择器
    function testByClass(){
        var objs$(".commom");
        objs.css("color","red")

    }

    //组合选择器
    function testBySelect(){
        var objs=$("#upwd,#uname,#showdiv")
        objs.css("background-color","aqua");

    }
</script>
<!--声明css代码域-->
```


jQuery 的层级选择器

jQuery的层级选择器:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>层级选择器</title>
    <style type="text/css">
      #showdiv{
        border: solid 1px;
        width: 600px;
        height:300px;
        background-color: orange;
        margin-top: 20px;
        text-align: center;
      }
    </style>
    <!--
```

层级选择器学习:

祖先获取所有后代:

`$("选择器 标签名")` 返回的是选择器下的所有指明的后

代元素。

选择所有子元素

`$("选择器>选择器")` 返回的是所有的子元素

选择紧跟的元素

`$("选择器+选择器")` 返回的是紧跟的元素

选择后面所有的元素

`$("选择器 ~选择器")` 返回所有后面的元素 (了解)

-->

```
<script src="js/jquery-1.9.1.js" type="text/javascript"
```

```
charset="utf-8"></script>
<script type="text/javascript">

//层级选择器

//选择所有的后代元素
function getAllChild(){
    var objs=$("#showdiv img");
    objs.css("border","solid 3px red");
}

//选择子元素
function getChild(){
    var objs=$("#showdiv>img");
    objs.css("padding","5px");
}

//选择紧跟元素
function getNextEle(){
    var objs=$("#img+br+span");
    objs.css("color","red");
}

//案例
$(function(){
    $("#showdiv img").mouseover(function(){
        $(this).css("border","solid 3px red")
    })

    $("#showdiv img").mouseout(function(){
        $(this).css("border","")
    })
})
</script>
</head>
<body>

<h3>层级选择器：</h3>

<input type="button" id="" value="测试层级选择器--选择所有
后代元素" onclick="getAllChild()"/>
```

```

        <input type="button" id="" value="测试层级选择器--选择所有
子元素" onclick="getChild()"/>

        <input type="button" id="" value="测试层级选择器--选择紧跟
元素" onclick="getNextEle()"/>
    <hr />
    <div id="showdiv">
        <br /><span>产品价
格: ¥99.99</span>
        <div id="">
            <br /><span>产
品价格: ¥100.99</span>
        </div>
    </div>
    <br /><span>产品价格: ¥200.99</span>
</body>
</html>

```

jQuery 的简单选择器

jQuery的简单选择器：

```

<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
        <title>简单选择器</title>
        <style type="text/css">
            table{
                border: solid 1px;
            }
            tr{

```

```
        height: 35px;
    }
    td{
        border: solid 1px;
        width: 100px;
    }
</style>
<!--
```

简单选择器：在其他选择器的基础上选择某个或某些指定的HTML元

素(筛选)

选择第一个元素：

`$("选择器:first")` 返回第一个HTML元素

选择最后一个

`$("选择器:last")` 返回第一个HTML元素

去除某个元素：

`$("选择器:not(选择器)")` 去除选择的HTML元素中的某些

元素

奇数偶数

`$("选择器:even")` 返回角标为偶数的HTML元素

`$("选择器:odd")` 返回角标为奇数的HTML元素

指定角标

`$("选择器:eq(角标)")`返回指定角标的HTML元素对象

大于角标

`$("选择器:gt(角标)")`返回大于指定角标的HTML元素

小于角标

`$("选择器:lt(角标)")`返回小于指定角标的HTML元素


```
-->
<script src="js/jquery-1.9.1.js" type="text/javascript"
charset="utf-8"></script>
<script type="text/javascript">

    //简单选择器：

        //选择第一个
        function getFirst(){
            $("tr:first").css("background-color","orange");
        }

        //选择最后一个
        function getLast(){
            $("tr:last").css("background-color","red");
        }

        //去除某个元素
        function getNot(){

            $("tr:not(tr:first)").css("background-color","orangered");
        }

        //获取奇数和偶数--隔行变色
        function getEvenOdd(){

            $("tr:not(tr:first):even").css("background-color","orange")
;

            $("tr:not(tr:first):odd").css("background-color","blueviole
t");
        }

        //按照角标筛选
        function getEq(){
            $("tr:eq(3)").css("background-color","brown");
        }

        //按照大于某值
        function getGt(){
            $("tr:gt(3)").css("background-color","brown");
        }

    }
```



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

jQuery 的内容选择器

jQuery的内容选择器：

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
        <title>内容选择器</title>
        <script src="js/jquery-1.9.1.js" type="text/javascript"
charset="utf-8"></script>
    <!--
```

内容选择器：

包含：

选择器:contains("内容") 返回包含指定内容的元素

内容为空：

选择器:empty 返回内容为空的元素对象

含有指定元素对象

选择器:has(选择器) 返回含有指定的元素的元素对象

不为空的元素

选择器:parent 返回内容不为空的元素

```
-->
<script type="text/javascript">
    //内容选择器：

    //包含--模糊筛选
    function testCon(){
        $("li:contains('水果')").css("color","red");
    }

    //内容为空
    function testEmp(){
        $("li:empty").css("color","royalblue");
    }

    //匹配含有选择器所匹配的元素
    function testHas(){
        $("li:has(a)").css("background-color","#FFA500");
    }

    //匹配含有子元素或者文本的元素
    function testParent(){
        $("li:parent").css("background-color","#FFA500");
    }
</script>
</head>
<body>

    <h3>内容选择器：</h3>

    <input type="button" id="" value="内容选择器---包含"
onclick="testCon()"/>

    <input type="button" id="" value="内容选择器---空选择"
onclick="testEmp()"/>
```

```
<input type="button" id="" value="内容选择器---包含指定的元素" onclick="testHas()"/>
```

```
<input type="button" id="" value="内容选择器---有子元素的元素" onclick="testParent()"/>
```

```
<hr />
<ul>
    <li>水果：苹果</li>
    <li>键盘：双飞燕</li>
    <li>电脑：thinkpid</li>
    <li>水果：香蕉</li>
    <li>篮球：乔丹</li>
    <li>水果：猕猴桃</li>
    <li></li>
    <li><a href="http://www.baidu.com">百度一下</a></li>
</ul>
</body>
</html>
```

jQuery 的可见性选择器

jQuery的可见性选择器：

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
        <title>可见性</title>
        <!--
```

可见性

匹配隐藏

`$("选择器:hidden")` 返回隐藏的HTML元素

匹配可见

`$("选择器:visible")` 返回可见的HTML元素

```
-->
<script src="js/jquery-1.9.1.js" type="text/javascript"
charset="utf-8"></script>
<script type="text/javascript">
    //可见性

    //匹配隐藏
    function testHidden(){
        $("div:hidden").css("display","block");
    }

    //匹配可见的
    function testVisible(){
        $("div:visible").css("display","none");
    }
</script>
<style type="text/css">
    #div01{
        border: solid 1px;
        width: 200px;
        height: 200px;
        background-color: orange;
        margin-bottom: 20px;

        display: none;/*隐藏元素*/

    }
    #div02{
        border: solid 1px;
        width: 200px;
```

```
        height: 200px;
        background-color: aqua;
    }
</style>
</head>
<body>

    <h3>可见性</h3>

    <input type="button" id="" value="可见性--hidden"
onclick="testHidden()"/>

    <input type="button" id="" value="可见性--visible"
onclick="testVisible()"/>
    <hr />
    <div id="div01"></div>
    <div id="div02"></div>
</body>
</html>
```

jQuery 的属性选择器

jQuery的属性选择器：

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>属性选择器</title>
    <!--
```

属性选择器：

属性：

`$("选择器[属性名]")` 返回具备某些属性的HTML元素

属性值：

`$("选择器[属性名=值]")` 返回具备属性且属性值为特定

值的HTML元素

不等于特定值

`$("选择器[属性名!=值]")` 返回具备属性且属性值不等于

特定值的HTML元素

其他：参照API

```
-->
<script src="js/jquery-1.9.1.js" type="text/javascript"
charset="utf-8"></script>
<script type="text/javascript">

    //属性选择器

    //属性选择
    function getEleByAttr(){

        $("input[name]").css("background-color","hotpink");
    }

    //属性为特定值
    function getEleByAttrValue(){

        $("input[type=text]").css("background-color","aqua");
    }

    //属性值不等于特定值
    function getEleByAttrValue2(){

        $("div>input[type!=text]").css("background-color","red");
    }
</script>
<style type="text/css">
    #showdiv{
        border: solid 1px;
        width: 300px;
        height: 240px;
        background-color: orange;
        margin-bottom: 20px;
        padding: 10px;
    }
```

```
        input{
            margin: 10px;
        }
    </style>
</head>
<body>

    <h3>属性选择器</h3>

    <input type="button" id="" value="属性选择器--属性"
onclick="getEleByAttr()"/>

    <input type="button" id="" value="属性选择器--属性值"
onclick="getEleByAttrValue()"/>

    <input type="button" id="" value="属性选择器--不等于特定值"
onclick="getEleByAttrValue2()"/>
    <hr />
    <div id="showdiv">

        用户名:<input type="text" name="uname" id=""
value="" /><br /><br />

        手机号: <input type="text" name="phone" id=""
value="" /><br /><br />

        密码: <input type="password" name="" id="" value=""
/><br /><br />

        邮箱:<input type="text" value="" /><br />
    </div>
</body>
</html>
```

jQuery 的表单选择器

jQuery的表单选择器：

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
```

```
<title>表单选择器</title>

<!--引入css-->
<link rel="stylesheet" type="text/css" href="css/my.css"/>

<!--引入jQuery-->
<script src="js/jquery-1.9.1.js" type="text/javascript"
charset="utf-8"></script>

<!--声明js代码域-->
<script type="text/javascript">
    //表单选择器

    //匹配所有 input, textarea, select 和 button 元素
    function testInput(){
        $(":input").css("background-color","red");
    }

    //匹配所有的单行文本框
    function testText(){
        $(":text").css("background-color","red");
    }
</script>
<!--
    表单操作及表单对象属性参照API即可
-->
</head>
<body>
    <div id="" style="text-align: center;">

        <input type="button" id="" value="表单选择器--input"
onclick="testInput()"/>

        <input type="button" id="" value="表单选择器--text"
onclick="testText()"/>
    </div>
    <hr />
    <form action="" method="post">
    <div id="showdiv">
    <table cellpadding="0" cellspacing="0">
        <tr>
```

```
<td>用户名:</td>
<td>
    <input type="text" name="uname" id="uname"
value="" disabled="disabled"/>
</td>
</tr>
<tr>
    <td width="70px">密码:</td>
    <td width="200px">
        <input type="password" name="pwd" id="pwd"
value="" />
    </td>
</tr>
<tr>
    <td>邮箱:</td>
    <td>
        <input type="text" name="mail" id="mail" value=""
/>
    </td>
</tr>
<tr>
    <td>手机号:</td>
    <td>
        <input type="text" name="phone" id="phone"
value="" />
    </td>
</tr>
<tr>
    <td>性别:</td>
    <td>
        男: <input type="radio" name="sex" id=""
value="1" />
        女: <input type="radio" name="sex" id=""
value="0" />
    </td>
</tr>
</tr>
```

```

        <td>爱好:</td>
        <td>
            唱歌<input type="checkbox" name="fav" id=""
value="1" />
            跳舞<input type="checkbox" name="fav" id=""
value="2" />
            打球<input type="checkbox" name="fav" id=""
value="3" />
        </td>
    </tr>
    <tr>
        <td>籍贯:</td>
        <td>
            <select name="address" id="">
                <option value="1">北京</option>
                <option value="2">上海</option>
                <option value="3">商丘</option>
            </select>
        </td>
    </tr>
    <tr>
        <td>个人介绍:</td>
        <td>
            <textarea name="intro" rows="4"
cols="20"></textarea>
        </td>
    </tr>
    <tr>
        <td colspan="2" align="center">
            <input type="checkbox" name="" id="" value="" />
            是否同意协议
        </td>
    </tr>
    <tr>
        <td colspan="2">

```

```
<td colspan="2" align="center">
    <input type="submit" name="" id="" value="注册" />
</td>
</tr>
</table>
</div>
</form>
</body>
</html>
```

jQuery 的操作元素属性

jQuery操作元素属性：

```
<html>
  <head>

    <title>操作属性</title>
    <meta charset="UTF-8"/>
    <style type="text/css">
      #showdiv{
        border: solid 1px;
        width: 400px;
        height: 100px;
        background-color: orange;
      }
      input[type=text]{
        margin: 10px;
      }
    </style>
    <script src="js/jquery-1.9.1.js" type="text/javascript"
charset="utf-8"></script>
    <!--
```

jQuery获取元素对象(选择器)

jquery操作元素属性

获取：

对象名.attr("属性名") 返回属性值。返回的是默认值

注意：

如果要获取用户数据使用对象名.val()

修改

对象名.attr("属性名", "新的值")

注意：

使用jQuery对象获取紧跟的元素：

对象名.next() //返回的是紧跟的jQuery对象

-->

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

```
    //操作属性
```

```
    //获取
```

```
    function getField(){
```

```
        //获取元素对象
```

```
        var obj=$("#uname");
```

```
        //获取
```

```
        alert(obj.attr("type")+":"+obj.attr("name")+":"+obj.attr("id")+":"+obj.attr("value"));
```

```
        alert(obj.val());
```

```
    }
```

```
    //修改
```

```
    function operField(){
```

```
        //获取元素对象
```

```
        var obj=$("#uname");
```

```
        //修改
```

```
        obj.attr("type", "button")
```

```
    }
```

```
    //案例：用户名校验
```

```
    function checkUname(){
```

```
//获取元素对象

var obj=$("#uname");//用户名

//获取用户名数据
var data=obj.val();

//校验
if(data==" " || data==null){
    obj.next().html("*用户名不能为空");
    obj.next().css("color","red");
}else if(data.length>6&&data.length<=8){
    obj.next().html("*用户名正确
").css("color","green");
}else{
    obj.next().html("*用户名不符合规则");
    obj.next().css("color","red");
}
}
</script>
</head>
<body>

<h3>操作属性</h3>

<input type="button" id="" value="操作属性---获取"
onclick="getField()"/>

<input type="button" id="" value="操作属性---修改"
onclick="operField()"/>
<hr />
<div id="showdiv">

    用户名: <input type="text" name="uname" id="uname"
value="" onblur="checkUname()" /><span></span>
</div>
</body>
</html>
```


jQuery 的操作元素内容

jQuery操作元素内容：

```
<html>
  <head>

    <title>操作元素内容</title>

    <meta charset="UTF-8"/>
    <style type="text/css">
      #showdiv{
        border: solid 1px;
        width: 200px;
        height: 200px;
        background-color: orange;
      }
    </style>

    <!--引入-->

    <script src="js/jquery-1.9.1.js" type="text/javascript"
charset="utf-8"></script>
  <!--
```

jquery获取元素对象

jQuery操作元素内容

获取：

元素对象.html() 返回元素的所有内容包括HTML标签

元素对象.text() 返回所有的文本内容，不包括HTML标签

修改

元素对象.html("新的内容") 会覆盖原有内容，解析HTML

标签

元素对象.text("新的内容") 覆盖原有内容，不解析HTML

标签

注意：

如果是追加则使用

元素对象.html(元素对象.html()+"新的内容")

元素对象.text(元素对象.text()+"新的内容")

-->

<script type="text/javascript">

//操作元素内容

//获取

function getData(){

//获取元素对象

var obj=\$("#showdiv");

//获取元素内容

alert(obj.html());//返回元素的所有内容包括

HTML标签

alert(obj.text());//返回所有的文本内容，不包括

HTML标签

}

//修改

function operData(){

//获取元素对象

var obj=\$("#showdiv");

//修改元素内容

obj.html(obj.html()+"<i>后端知识马上就开始了，

你的人生巅峰就要到了</i>");//会覆盖原有内容，解析HTML标签

obj.text(obj.text()+"<i>后端知识马上就开始了，

你的人生巅峰就要到了</i>");//会覆盖原有内容，不解析HTML标签

```
    }  
  </script>  
</head>  
<body>  
  <h3>操作元素内容</h3>  
  
  <input type="button" id="" value="操作元素内容--获取"  
onclick="getData()"/>  
  
  <input type="button" id="" value="操作元素内容--获取"  
onclick="operData()"/>  
  <hr />  
  <div id="showdiv">  
    <b>别急，前端知识马上就学完啦。</b>  
  </div>  
</body>  
</html>
```

jQuery 操作元素样式

jQuery操作元素样式：

```
<html>  
<head>  
  
  <title>操作样式</title>  
  <meta charset="UTF-8"/>  
  <script src="js/jquery-1.9.1.js" type="text/javascript"  
charset="utf-8"></script>  
  <!--
```

jQuery获取元素对象

jquery操作样式：

添加：

元素对象.css("样式名","样式值");

元素对象.css({"样式名":"样式值","样式名":"样式值"

", "样式名": "样式值",})

修改

元素对象.css("样式名", "新的样式");

注意:

也可以元素对象名.addClass("类选择器名");

```
-->
<script type="text/javascript">
    //操作元素样式

    //添加
    function addCss(){
        //获取元素对象
        var obj=$("#showdiv");
        //添加样式

        //基本写法
        //
        obj.css("border", "solid 1px");
        //
        obj.css("width", "200px");
        //
        obj.css("height", "200px");
        //
        obj.css("background-color", "orange");

        //高级写法
        obj.css({"border": "solid
1px", "width": "200px", "height": "200px", "background-color": "oran
ge"})
    }

    //修改
    function updateCss(){
        //获取元素对象
        var obj=$("#showdiv");
        //修改
```

```
        obj.css("background-color","darkcyan");
    }
    //添加
    function addClass(){
        //获取元素对象
        var obj=$("#showdiv");
        //使用类选择器添加
        obj.addClass("commom");
    }

</script>
<style type="text/css">
    .commom{
        border: solid 1px;
        width: 200px;
        height: 200px;
        background-color: orange;
    }
</style>
</head>
<body>
<h3>操作样式</h3>

    <input type="button" id="" value="操作样式---添加--css()"
onclick="addCss()"/>

    <input type="button" id="" value="操作样式---修改--css()"
onclick="updateCss()"/>

    <input type="button" id="" value="操作样式---修改
--addClass()" onclick="addClass()"/>
    <hr />
    <div id="showdiv">

    </div>
</body>
</html>
```

jQuery 操作元素文档结构

jQuery操作元素文档结构：

```
<html>
  <head>

    <title>操作文档</title>

    <meta charset="UTF-8"/>
    <style type="text/css">
      #div01{
        border: solid 1px;
        width: 400px;
        height: 100px;
        background-color: orange;
        margin-bottom: 30px;
        padding: 10px;
      }

      #div02{
        border: solid 1px;
        width: 400px;
        height: 100px;
        background-color: darkgrey;
        padding: 10px;
      }
    </style>
    <script src="js/jquery-1.9.1.js" type="text/javascript"
charset="utf-8"></script>
  <!--
```

jQuery获取元素对象

操作文档结构：修改，添加，删除

内部插入：

元素对象名.append("HTML代码") 追加(创建新的节点并追加)

元素对象名.appendTo("选择器") 将制定的元素移动并追加到

指定的元素中(将别的节点移动到追加)

元素对象名.prepend("HTML代码") 在前面追加(创建新的节点)

并追加)

元素对象名.prependTo("选择器")	将指定的元素移动并在前面
------------------------	--------------

追加到指定的元素中(将别的节点移动到在前面追加)

外部插入:

元素对象名.after("HTML代码")	在元素之后添加
-----------------------	---------

元素对象名.before("HTML代码")	在元素之前添加
------------------------	---------

元素对象名.insertAfter("选择器")	将指定的节点移动到指定的
--------------------------	--------------

元素后

元素对象名.insertBefore("选择器")	将指定的节点移动到指定的
---------------------------	--------------

元素前

包裹:

参照API

替换:

replace()

删除:

empty()

remove()

detach()

克隆:

clone()

clone(true)

-->

<script type="text/javascript">

//操作文档结构:

//内部插入

//append

function testAppend(){

```
//获取元素对象
var obj=$("#div01");
//内部插入--追加

obj.append("<span>聪明</span>");
//obj.append($("#div02+span"));
}
//appendTo
function testAppendTo(){
    //获取元素对象
    $("#div02+span").appendTo("div");
}
//prepend
function testPrepend(){
    //获取元素对象
    var obj=$("#div01");
    //内部插入--追加

    obj.prepend("<span>帅气</span>");
}

//外部插入
//after
function testAfter(){
    //获取元素对象

    $("#div01").after("<span>大家对他的印
象:</span>");
}

//before
function testBefore(){
    //获取元素对象

    $("#div01").before("<span>请选择印象标
```



```
签:</span>");
    }
    //insertAfter
    function testInsertAfter(){
        //将指定的节点移动到指定的元素后面
        $("#div02+span").insertAfter("#div01");
    }
    //替换
    function testReplace(){
        $("#div01 span").replaceWith("<b>哈哈</b>");
    }
    /*-----*/
    -----*/

    function operSpanCss(){
        //获取元素对象
        $("#div01
span").css("background-color","#FFF5A8");
        $("#div01 span").css("font-size","20px");
        $("#div01 span").css("padding","5px");
    }

    //案例
    function moveSpan(){
        $("#div01 span").click(function(){
            $(this).clone().appendTo("#div02");
        })
    }
</script>

</head>
<body>

    <h3>操作文档结构</h3>

    <input type="button" id="" value="测试操作文档结构--内部插
```

```
入--append" onclick="testAppend()"/>

    <input type="button" id="" value="测试操作文档结构--内部插
入--appendTo" onclick="testAppendTo()"/>

    <input type="button" id="" value="测试操作文档结构--内部插
入--prepend" onclick="testPrepend()"/>

    <input type="button" id="" value="span样式1"
onclick="operSpanCss()"/>
    <input type="button" id="" value="案例-移动span"
onclick="moveSpan()"/>
    <hr />
    <input type="button" id="" value="测试外部插入--after"
onclick="testAfter()"/>
    <input type="button" id="" value="测试外部插入--after"
onclick="testBefore()"/>
    <input type="button" id="" value="测试外部插入
--insertAfter" onclick="testInsertAfter()"/>
    <hr />
    <input type="button" id="" value="替换"
onclick="testReplace()"/>
    <hr />
    <div id="div01">
        <span>情商高</span>
    </div>
    <div id="div02">

    </div>
    <span>666</span>
</body>
</html>
```