

实验八 Ajax 与 jQuery 编程实验

一、教学课时

4 学时。

二、实验目的和要求

1. 使用 JavaScript 代码实现 Ajax 请求。
2. 掌握 jQuery 中常见选择器使用。
3. 掌握 jQuery 中的 DOM 操作。
4. 掌握 jQuery 中的事件与动画。

三、实验内容

第一阶段：程序验证

1. 指导 1 使用 JavaScript 代码实现一个简单的 Ajax 请求

知识点：

- XMLHttpRequest 对象的使用
- open()方法， send()方法
- 回调函数的处理

(1) 问题：我们学习了 ajax 的原理，在这儿我们编写 JS 代码实现 Ajax 请求，接受服务端的返回并输出对应的数据。

(2) 分析：当我们点击按钮后，从服务端返回数据，然后在页面显示出来。
思路如下：

- 1) 创建 XMLHttpRequest 对象
- 2) 向服务器发送异步请求
- 3) 接收服务器端的响应并处理

(3) 解决方案：

- a) 在网页中编写如下代码。

```

<script>
function ajax(){
    // 声明一个控队想用来装入XMLHttpRequest
    var xmlhttpReq = null;
    // IE5 IE6是以ActiveObject的方式引入XMLHttpRequest的
    if(window.ActiveXObject){
        xmlhttpReq = new ActiveXObject("Microsoft.XMLHTTP");
    }
    // 除IE5 IE6以外的浏览器XMLHttpRequest是window的子对象
    else if(window.XMLHttpRequest){
        xmlhttpReq = new XMLHttpRequest(); // 实例化一个XMLHttpRequest
    }

    if(xmlhttpReq != null){ // 如果对对象实例化成功
        xmlhttpReq.open("GET", "https://jsonplaceholder.typicode.com/todos/1", true); // 调用open()方法并采用异步方式
        xmlhttpReq.onreadystatechange = doResult; // 设置回调函数
        xmlhttpReq.send(null) // 因为使用get方式提交，所以可以使用null参数调用
    }

    function doResult(){ // 一旦readyState值改变，将会调用这个函数，readyState=4表示完成响应
        if(xmlhttpReq.readyState === 4){
            if(xmlhttpReq.status === 200){
                // 将xmlhttpReq.responseText的值赋给ID为resText的元素
                document.getElementById("resText").innerHTML = xmlhttpReq.responseText;
            }
        }
    }
}
</script>
</head>
<body>
<input type="button" id="" value="Ajax请求页面" onclick="ajax();"/>
<div id="resText"></div>
</body>

```

- (4) 在浏览器中运行这个网页，然后点击按钮，可以在网页中看到后端传过来的数据

2. 指导 2 实现手风琴效果

知识点：

- jQuery 选择器
- hide()方法，show()方法

- (1) 问题：实现手风琴效果，初始化页面如下所示：



- (2) 分析：首先用 span 元素创建章节的标题，然后在每个 span 元素的后面添加一个层，用于显示标题对应的内容，该层内添加四个标记
- (3) 解决方案：
- a) 创建网页，编写基本结构代码。

```

<style>
    #menu{
        width: 300px;
    }
    #menu span{
        background: gray;
        color: #fff;
        cursor: pointer;
        display: block;
    }
    #menu .hidden{
        display: none;
    }
</style>

<body>
    <div id="menu">
        <span id="m1">第1章</span>
        <div id="div_m1">
            <a href="#">A</a></br>
            <a href="#">B</a></br>
            <a href="#">C</a></br>
            <a href="#">D</a></br>
        </div>
        <span id="m2">第2章</span>
        <div id="div_m2" class="hidden">
            <a href="#">1</a></br>
            <a href="#">2</a></br>
            <a href="#">3</a></br>
            <a href="#">4</a></br>
        </div>
        <span id="m3">第3章</span>
        <div id="div_m3" class="hidden">
            <a href="#">!</a></br>
            <a href="#">@</a></br>
            <a href="#">#</a></br>
            <a href="#">$</a></br>
        </div>
    </div>
    <script>
    |
    </script>
</body>

```

b) 引入jQuery 库。

```

<!-- 通过CDN方式引入，可以自己在网上选择合适的版本 -->
<script src="http://code.jquery.com/jquery-2.1.4.min.js"></script>

```

c) 在网页正文<body></body>标签内编写 JavaScript 如下代码。

```
<script>
    $(function(){
        // 为标题添加单击事件
        $("#menu>span").click(function(){
            // 隐藏所有内容
            $("#menu>div").hide();
            // 显示选中内容
            $("#div_" + this.id).show();
        })
    })
</script>
```

(4) 在浏览器中运行此网页，如图上机 2 所示。



图上机 2

3. 指导 3 网页选项卡的应用

知识点:

- jQuery 选择器
- jQuery DOM 操作
- jQuery 简单动画

(1) 问题：实现选项卡效果，如下图：

(2) 分析：网页选项卡实质是通过隐藏和显示来切换不同的内容

(3) 解决方案：

a) 创建 HTML 代码。

```

<div class="tab">
  <div class="tab_menu">
    <ul>
      <li class="selected">新闻</li>
      <li>财经</li>
      <li>视频</li>
    </ul>
  </div>
</div>
<div class="tab_box">
  <div>新闻信息</div>
  <div class="hide">财经信息</div>
  <div class="hide">视频信息</div>
</div>

```

b) 为页面元素添加样式:

```

<style>
  body{
    font: 12/19px Arial, Helvetica, sans-serif;
    color: #666;
  }
  .tab{width: 240px;margin: 50px;}
  .tab_menu{clear: both;}
  .tab_menu li{
    float: left;
    text-align: center;
    cursor: pointer;
    list-style: none;
    padding: 1px 6px;
    margin-right: 4px;
    background: #F1F1F1;
    border: 1px solid #898989;
    border-bottom: none;
  }
  .tab_menu li.hover{background: #DFDFDF;}
  .tab_menu li.selected{color: #FFF;background: #6D84B4;}
  .tab_box{
    clear: both;
    border: 1px solid #898989;
    width: 300px;
    height: 100px;
  }
  .hide{display: none;}
</style>

```

c) 编写如下的jQuery 代码:

```

<script>
$(function(){
    var $div_li=$("div.tab_menu ul li");
    $div_li.click(function(){
        $(this).addClass("selected") // 当前<li>元素高亮
        .siblings().removeClass("selected"); // 去掉其它同辈<li>元素的高亮

        // 获取当前点击的<li>元素在全部li元素中的索引
        var index = $div_li.index(this);
        // 选取子节点。不选取子节点的话，会引起错误。如果里面还有div
        $("div.tab_box>div")
            .eq(index).show() // 显示<li>元素对应的<div>元素
            .siblings().hide(); // 隐藏其它几个同辈的<div>元素
    }).hover(function(){
        $(this).addClass("hover");
    }, function(){
        $(this).removeClass("hover");
    })
})
</script>

```

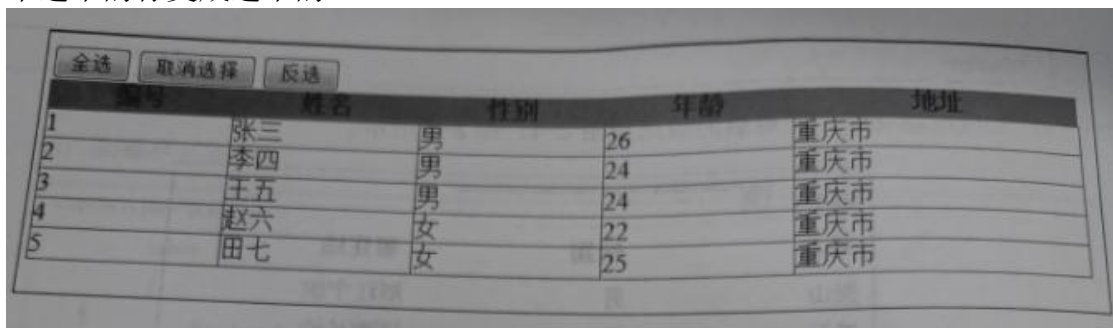
(4) 在浏览器中运行此网页，如图上机 3 所示。



图上机 3

第二阶段：程序编写

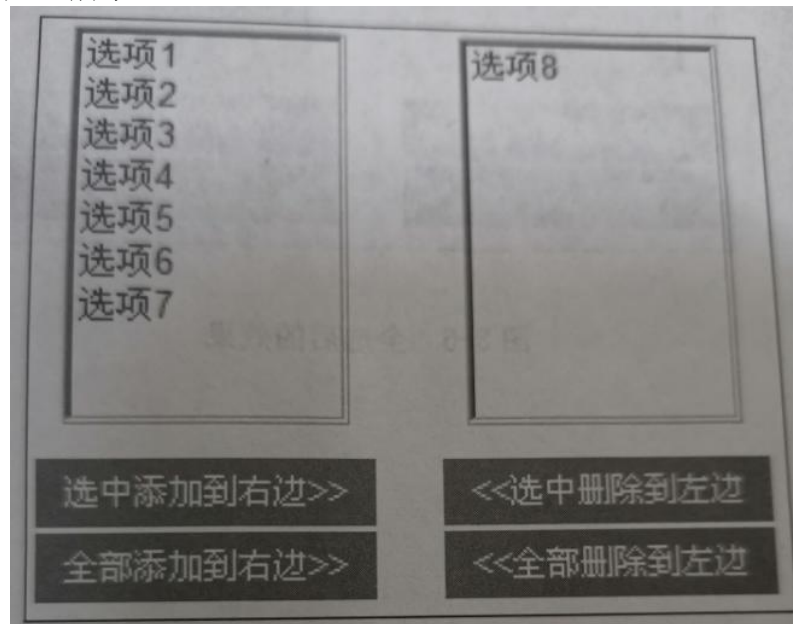
1. （选择器练习 1）利用 jQuery 选择器实现表格的全选，取消选择，单击更改选中状态，反选等功能。如图所示。点击全选按钮，选中所有行，点击取消按钮，所有行为未选中状态，单击反选按钮，则选中的行变成未选中的；未选中的行变成选中的。



图练习 1

提示：行点击事件可以使用 toggleClass() 方法，如果样式不存在，则添加，如果存在，则移除。

2. (DOM 操作练习 2) 点击“选中添加到右边”讲选中项移动到右侧下拉框，如图 2-2 所示：



图练习 2

3. (jQuery 动画练习 3) 修改练习 1 中的表格，为表格添加光棒效果，当鼠标移入某一行时，对应行出现光棒效果（高亮鼠标移入行）。