# 第5天 jQuery基础(上)

【学习目标】

## 1 了解jQuery技术

a)了解jQuery的优点

b)了解jQuery的下载途径

c)掌握jQuery的引入

## 2掌握jQuery对象和DOM对象的转换

## 3掌握jQuery中常见的选择器

## 4会使用jQuery为元素绑定事件

## 5掌握jQuery中常用的方法

a)掌握jQuery中操作元素属性的方法

b)掌握jQuery中操作元素样式的方法

c)掌握jQuery中动画效果的方法

# 案例一:重写弹出广告

## 案例分析

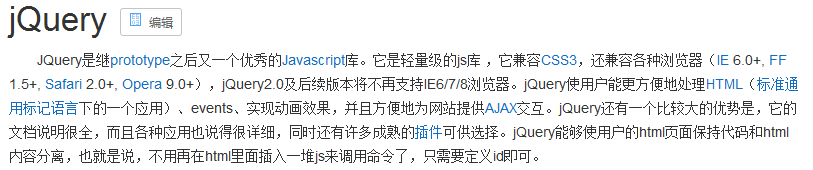
在首页中(logo的上方)显示一个广告图片(页面加载后，间隔3秒弹出广告图片，再间隔3秒后隐藏广告图片)。【使用jQuery实现】



## 技术分析

* + 1. jQuery简介

**1.什么是jQuery**

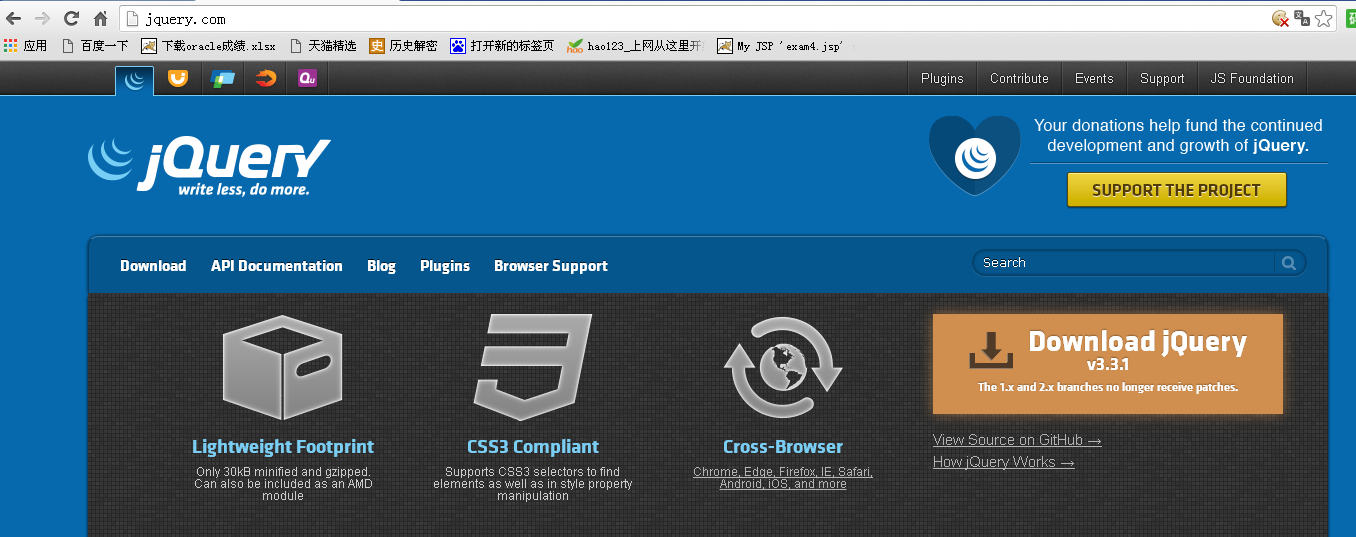
**** jQuery并不是一门新的语言，而是对javascript进行封装的一个轻量级框架。它提供了很多方便的选择器，供开发人员迅速定位到需要操作的元素，并且提供很多便捷的方法。

**2.JavaScript优点:**

1. 轻巧方便
2. 强大的选择器
3. 出色的DOM操作的封装
4. 可靠的事件处理机制
5. 出色浏览器兼容性
6. 不污染顶级变量
7. 完善的AJAX
8. 开源

**3.jQuery下载**

jQuery官网：<http://www.jquery.com.在该网站下载jquery最新版本jQuery>文件。

****

**4.jQuery的引入方式**

jQuery它是一个库(框架)，要想使用它，必须先引入！

jQuery-1.8.3.js：一般用于学习阶段。

实际项目中根据具体情况而定，一般使用压缩版（.min.js）

引入语法：<script type=”text/javascript” src=”js/jquery-1.8.3.js”></script>

**5.jquery入门案例**

|  |
| --- |
| <script type="text/javascript" src="../../js/jquery-1.8.3.js" ></script>  <script>  $(function(){  alert("hello jQuery!");  });    </script> |

* + 1. jQuery的页面加载

jQuery提供ready()函数，用于页面成功加载后执行。与window.onload函数类似。

|  |
| --- |
| <script type="text/javascript" src="../../js/jquery-1.8.3.js" ></script>  <script>  //传统的页面加载  //多次写onload会出现覆盖的问题，整个页面只能执行一次window.onload事件    window.onload = function(){  alert("张三");  }  //传统JS比JQ慢（JS页面加载是当整个页面的所有元素<其他的页面元素，如图片，视频...>都加载完毕才执行）  window.onload = function(){  alert("李四");  }    //JQ页面加载方式比JS页面加载更快（jQuery是在页面整个DOM树结构加载完就执行）  jQuery(document).ready(function(){    alert("王五");  });    //JQ页面加载不会出现覆盖问题，并且顺序执行  $(document).ready(function(){  alert("马六");  });    //最简化的写法  $(function(){  alert("小七");  });    </script> |

window.onload和jquery的ready函数的主要区别：

1.onload事件只能执行一个函数。如果多次绑定onload事件，则后面绑定的会覆盖之前的。因此只能执行一个函数；而ready函数可以调用多次，执行多个函数。

2.onload是当页面所有内容都加载完毕时才调用。如果页面中存在大的图片或者视频，则也必须等到加载完毕才能执行调用的函数；而ready是当页面主要元素加在完毕后执行，大的图片和视频等元素则未必加载完毕，这样用户体验度会更好一些，不用等待过长时间。

* + 1. jQuery对象与DOM对象转换

jQuery对象和DOM对象是相互独立的，可以相互转换，但是两个对象的函数不能混淆使用（这点一定要谨记）。

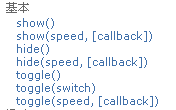
将DOM对象转换为jQuery对象，语法：jQuery(dom对象)或者$(dom对象)

将jQuery对象转换为DOM对象，语法：$jquery对象名[index]或者$jquery对象名.get(index)。

|  |
| --- |
| <html>  <head>  <meta charset="UTF-8">  <title>JS的DOM与JQ操作的区别</title>  <script type="text/javascript" src="../../js/jquery-1.8.3.js" ></script>  <script>  function writeJS(){  //获取页面的元素，操作属性，写入内容  //document.getElementById("span1").innerHTML="真强大！";  //DOM对象无法操作JQ中的属性和方法。  //document.getElementById("span1").html("是一个框架");    //将DOM对象转换为JQ对象  var spanEle = document.getElementById("span1");  $(spanEle).html("转成JQ对象");    }    //JQ的方式  $(function(){  $("#btn").click(function(){  //JQ对象无法操作DOM对象的属性和方法！！！！  //$("#span1").innerHTML="JQ真好玩儿！";    //JQ对象操作JQ的属性和方法  $("#span1").html("JQ真好玩儿！");    //JQ对象向DOM对象转换,方式一  //$("#span1").get(0).innerHTML = "转换成DOM对象";  //JQ对象向DOM对象转换,方式二  $("#span1")[0].innerHTML = "转换成DOM对象";    });      })    </script>  </head>  <body>  <input type="button" value="JS写入" onclick="writeJS()" />  <input type="button" value="JQ写入" id="btn" /><br>  jQuery:<span id="span1">真有趣！</span>  </body>  </html> |

* + 1. jQuery效果

**1.基本显示和隐藏效果**



show(speed,fn):显示某元素

参数speed:显示速度，单位毫秒。也可使用固定字符串：slow,normal,fast

参数fn:显示成功之后的回调函数

hide(speed,fn):隐藏某元素

参数同上

toggle(speed,fn):切换显示和隐藏效果，即如果元素隐藏，则使用该方法后则显示，反之则隐藏。

**2.滑动效果**



slideDown():元素显示，高度逐渐变高，直到设置值。

slideUp():元素隐藏，高度逐渐变小，直到高度为0,元素隐藏。

slideToggle():切换隐藏和显示效果

**3.淡入淡出效果**



fadeIn():元素显示。从透明一直到完全不透明。

fadeout():元素隐藏。从完全不透明到完全透明再隐藏。

fadeToggle():切换显示隐藏效果

fadeTo(speed,opacity,[fn]):指定透明度

第二个参数opacity取值0-1之间数值;

## 步骤分析

第一步：引入jQuery相关的文件

第二步：书写页面加载函数

第三步：在页面加载函数中，获取显示广告图片的元素。

第四步：设置定时操作(显示广告图片的函数)

第五步：在显示广告图片的函数中，使用jQuery的方法让广告图片显示(show())

第六步：清除显示广告图片的定时操作

第七步：设置定时操作(隐藏广告图片的函数)

第八步：在隐藏广告图片的函数中，使用jQuery的方法让广告图片隐藏(hide())

第九步：清除隐藏广告图片的定时操作

## 案例实现

|  |
| --- |
| <script type="text/javascript" src="../js/jquery-1.8.3.js" ></script>  <script>  //1.页面加载函数  $(function(){  //2.设置定时器，显示图片  time = setInterval("showAd()",3000);  });    //3.编写显示广告图片的函数  function showAd(){  //4.获取页面广告图片元素,并使其显示  //$("#showAdv").show(2000);    //$("#showAdv").slideDown(2000);    $("#showAdv").fadeIn(2000);    //5.清除显示的定时器  clearInterval(time);  //6.设置隐藏的定时器，隐藏广告  time = setInterval("hiddenAd()",4000);    }    function hiddenAd(){  //7.获取页面广告图片元素，并使其隐藏  //$("#showAdv").hide(2000);    //$("#showAdv").slideUp(2000);    $("#showAdv").fadeOut(2000);      //8.清除隐藏的定时器  clearInterval(time);  }  </script> |
| 页面：  <img id="showAdv" src="../img/f001a62f-a49d-4a4d-b56f-2b6908a0002c\_g.jpg" width="100%" style="display: none;"> |

## 案例总结(选择器)

* + 1. jQuery基本选择器

id选择器：$(“#id名称”);

元素选择器：$(“元素名称”);

类选择器：$(“.类名”);

通配符：\*

多个选择器共用(并集)

案例代码：

|  |
| --- |
| <html>  <head>  <meta charset="UTF-8">  <title>基本选择器</title>  <link rel="stylesheet"href="../../css/style.css"type="text/css"/>  <script type="text/javascript"src="../../js/jquery-1.8.3.js"></script>  <script>  $(function(){  $("#btn1").click(function(){  $("#one").css("background-color","pink");  });  $("#btn2").click(function(){  $(".mini").css("background-color","pink");  });  $("#btn3").click(function(){  $("div").css("background-color","pink");  });  $("#btn4").click(function(){  $("\*").css("background-color","pink");  });  $("#btn5").click(function(){  $("#two,.mini").css("background-color","pink");  });  });  </script>  </head>  <body>  <input type="button"id="btn1"value="选择为one的元素"/>  <input type="button"id="btn2"value="选择样式为mini的元素"/>  <input type="button"id="btn3"value="选择所有的div元素"/>  <input type="button"id="btn4"value="选择所有元素"/>  <input type="button"id="btn5"value="选择id为two并且样式为mini的元素"/>  <hr/>  <div id="one">  <div class="mini">  111  </div>  </div>    <div id="two">  <div class="mini">  222  </div>  <div class="mini">  333  </div>  </div>    <div id="three">  <div class="mini">  444  </div>  <div class="mini">  555  </div>  <div class="mini">  666  </div>  </div>    <span id="four">    </span>  </body>  </html> |

* + 1. 层级选择器

****

ancestor descendant: 在给定的祖先元素下匹配所有的后代元素(儿子、孙子、重孙子)

parent > child : 在给定的父元素下匹配所有的子元素(儿子)

prev + next: 匹配所有紧接在 prev 元素后的 next 元素(紧挨着的，同桌)

prev ~ siblings: 匹配 prev 元素之后的所有 siblings 元素(兄弟)

案例代码：

|  |
| --- |
| <html>  <head>  <meta charset="UTF-8">  <title>层级选择器</title>  <link rel="stylesheet"href="../../css/style.css"/>  <script type="text/javascript"src="../../js/jquery-1.8.3.js"></script>  <script>  $(function(){  $("#btn1").click(function(){  $("body div").css("background-color","pink");  });  $("#btn2").click(function(){  $("body>div").css("background-color","pink");  });  $("#btn3").click(function(){  $("#two+div").css("background-color","pink");  });  $("#btn4").click(function(){  $("#one~div").css("background-color","pink");  });  });    </script>      </head>  <body>  <input type="button"id="btn1"value="选择body中的所有的div元素"/>  <input type="button"id="btn2"value="选择body中的第一级的孩子"/>  <input type="button"id="btn3"value="选择id为two的元素的下一个元素"/>  <input type="button"id="btn4"value="选择id为one的所有的兄弟元素"/>    <hr/>  <div id="one">  <div class="mini">  111  </div>  </div>    <div id="two">  <div class="mini">  222  </div>  <div class="mini">  333  </div>  </div>    <div id="three">  <div class="mini">  444  </div>  <div class="mini">  555  </div>  <div class="mini">  666  </div>  </div>    <span id="four">    </span>  </body>  </html> |

* + 1. 基本过滤选择器

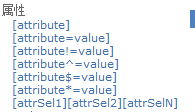
$('li').first() 等价于：$(“li:first”)



案例代码：

|  |
| --- |
| <html>  <head>  <meta charset="UTF-8">  <title>基本过滤选择器</title>  <link rel="stylesheet"href="../../css/style.css"type="text/css"/>  <script type="text/javascript"src="../../js/jquery-1.8.3.js"></script>  <script>  $(function(){  $("#btn1").click(function(){  $("div:first").css("background-color","pink");  });  $("#btn2").click(function(){  $("div:last").css("background-color","pink");  });  $("#btn3").click(function(){  $("div:odd").css("background-color","pink");  });  $("#btn4").click(function(){  $("div:even").css("background-color","pink");  });  });  </script>    </head>  <body>  <input type="button"id="btn1"value="body中的第一个div元素"/>  <input type="button"id="btn2"value="body中的最后一个div元素"/>  <input type="button"id="btn3"value="选择body中的奇数的div"/>  <input type="button"id="btn4"value="选择body中的偶数的div"/>    <hr/>  <div id="one">  <div class="mini">  111  </div>  </div>    <div id="two">  <div class="mini">  222  </div>  <div class="mini">  333  </div>  </div>    <div id="three">  <div class="mini">  444  </div>  <div class="mini">  555  </div>  <div class="mini">  666  </div>  </div>    <span id="four">    </span>  </body>  </html> |

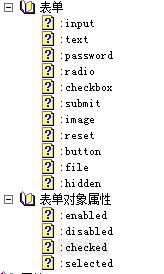
* + 1. 属性选择器

****

案例代码：

|  |
| --- |
| <html>  <head>  <meta charset="UTF-8">  <title>属性选择器</title>  <link rel="stylesheet"href="../../css/style.css"type="text/css"/>  <script type="text/javascript"src="../../js/jquery-1.8.3.js"></script>  <script>  $(function(){  $("#btn1").click(function(){  $("div[id]").css("background-color","pink");  });  $("#btn2").click(function(){  $("div[id='two']").css("background-color","pink");  });  });    </script>  </head>  <body>  <input type="button"id="btn1"value="选择有id属性的div"/>  <input type="button"id="btn2"value="选择有id属性的值为two的div"/>    <hr/>  <div id="one">  <div class="mini">  111  </div>  </div>    <div id="two">  <div class="mini">  222  </div>  <div class="mini">  333  </div>  </div>    <div id="three">  <div class="mini">  444  </div>  <div class="mini">  555  </div>  <div class="mini">  666  </div>  </div>    <span id="four">    </span>  </body>  </html> |

* + 1. 表单选择器

****

案例代码：

|  |
| --- |
| <html>  <head>  <meta charset="UTF-8">  <title>表单选择器</title>  <link rel="stylesheet"type="text/css"href="../../css/style.css"/>  <script type="text/javascript"src="../../js/jquery-1.8.3.js"></script>  <script>  $(function(){  $("#btn1").click(function(){  $(":input").css("background-color","pink");  });  $("#btn2").click(function(){  $(":text").css("background-color","pink");  });  });  </script>  </head>  <body>  <input type="button"id="btn1"value="选择所有input元素"/>  <input type="button"id="btn2"value="选择文本框"/>  <br/>  <form>  <input type="text"/><br />  <input type="checkbox"/><br />  <input type="radio"/><br />  <input type="image"/><br />  <input type="file"/><br />  <input type="submit"/>  <input type="reset"/><br />  <input type="password"/><br />  <input type="button"/><br />  <select><option/></select><br />  <textarea></textarea><br />  <button></button>  </form>  </body>  </html> |

# 案例二:使用jQuery完成表格隔行换色

## 需求分析

重写JavaScript案例中的表格隔行换色效果。

在页面加载的时候让显示用户的表格进行隔行换色显示，效果如下：使用JQ实现！



## 案例相关技术

* + 1. class操作



addClass():给指定元素追加class属性样式

removeClass():将指定元素的class属性移除

toggleClass():切换class属性样式，即没有该class属性则添加，有该class属性则移除

## 案例分析

步骤分析

第一步：引入jquery的类库

第二步：直接写页面加载函数

第三步：直接使用jquery的选择器(组合选择)拿到需要操作的元素(奇数行和偶数行)

第四步：分别使用CSS的方法(css(name,value))对奇数行和偶数行设置背景颜色。

## 代码实现

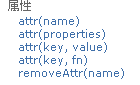
|  |
| --- |
| <link rel="stylesheet" href="../css/style.css" />  <script type="text/javascript" src="../js/jquery-1.8.3.js" ></script>  <script>  //1.页面加载函数  $(function(){  /\*//2.表格的tbody中的tr的偶数行，改变背景色  $("tbody>tr:even").css("background-color","orange");  //3.表格的tbody中的tr的奇数行，改变背景色  $("tbody>tr:odd").css("background-color","yellow");\*/  //追加类样式addClass  $("tbody>tr:even").addClass("even");    $("tbody>tr:even").removeClass("even");    $("tbody>tr:odd").addClass("odd");  });  </script> |

# 案例三:使用jQuery完成全选和全不选

* 1. 需求分析

在系统后台进行人员管理时，进行批量删除，使用jQuery完成全选和全不选效果

* 1. 相关技术分析
     1. 属性操作：prop()



prop()操作元素的特性。获取一些第一次分配undefined属性值的标签，如果抛异常，将忽略浏览器生成的任何错误。

removeProp():移除标签的特性。

* 1. 步骤分析

第一步：引入jquery文件

第二步：书写页面加载函数

第三步：为上面的复选框绑定单击事件

第四步：将下面所有的复选框的选中状态设置成跟上面的一致！

* 1. 代码实现

|  |
| --- |
| <script type="text/javascript" src="../js/jquery-1.8.3.js" ></script>  <!--<script type="text/javascript" src="../js/jquery-1.11.0.js" ></script>-->  <script>  $(function(){  //为编号前复选框绑定事件  $("#select").click(function(){  //获取下面的所有复选框，并设置checked的属性  //下面的所有复选框状态与编号前的复选框状态一致  //注意：attr函数与JQ版本有关，在1.8.3版本及以下可用  //$("tbody input:checkbox").attr("checked",this.checked);  //版本一般是向下兼容  $("tbody input:checkbox").prop("checked",this.checked);  });  });  </script> |

注意：attr在jquery1.11版本不适用，采用prop()来替代(在各个版本都适用)。

# 今日总结

今天主要介绍了jQuery基础，包括jQuery引入方式、jQuery获取页面元素方法、jQuery对象与DOM对象的转换、jQuery中的特效函数、jQuery中的选择器等内容。希望同学根据讲的案例，结合技术文档对知识点进行巩固。