

## 03-jQuery

讲师：王振国

### 今日任务

- 1.jQuery Hello world
- 2.jQuery 选择器
- 3.jQuery 过滤器
- 4.jQuery 元素筛选

### 课堂笔记

## 1、jQuery 介绍

#### 什么是 jQuery ?

jQuery，顾名思义，也就是 JavaScript 和查询（Query），它就是辅助 JavaScript 开发的 js 类库。

#### jQuery 核心思想!!!

它的核心思想是 write less, do more (写得更少, 做得更多)，所以它实现了很多浏览器的兼容问题。

#### jQuery 流行程度

jQuery 现在已经成为最流行的 JavaScript 库，在世界前 10000 个访问最多的网站中，有超过 55% 在使用 jQuery。

#### jQuery 好处!!!

jQuery 是免费、开源的，jQuery 的语法设计可以使开发更加便捷，例如操作 [文档](#) 对象、选择 [DOM](#) 元素、制作动画效果、事件处理、使用 [Ajax](#) 以及其他功能

## 2、jQuery 的初体验!!!

需求：使用 jQuery 给一个按钮绑定单击事件？

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Insert title here</title>
  <script type="text/javascript" src="../script/jquery-1.7.2.js"></script>
  <script type="text/javascript">
    // window.onload = function () {
    //     var btnObj = document.getElementById("btnId");
    //     // alert(btnObj);//[object HTMLButtonElement] ===>>> dom 对象
    //     btnObj.onclick = function () {
    //         alert("js 原生的单击事件");
    //     }
    // }

    $(function () { // 表示页面加载完成 之后, 相当 window.onload = function () {}
        var $btnObj = $("#btnId"); // 表示按 id 查询标签对象
        $btnObj.click(function () { // 绑定单击事件
            alert("jQuery 的单击事件");
        });
    });

  </script>
</head>
<body>

  <button id="btnId">SayHello</button>

</body>
</html>
```

常见问题？

- 1、使用 jQuery 一定要引入 jQuery 库吗？  
答案： 是，必须
- 2、jQuery 中的 \$ 到底是什么？  
答案： 它是一个函数
- 3、怎么为按钮添加点击响应函数的？  
答案：
  - 1、使用 jQuery 查询到标签对象
  - 2、使用标签对象.click( function(){} );

## 3、jQuery 核心函数

\$ 是 jQuery 的核心函数，能完成 jQuery 的很多功能。\$()就是调用\$这个函数

- 1、传入参数为 [ 函数 ] 时：  
表示页面加载完成之后。相当于 `window.onload = function(){}`
- 2、传入参数为 [ HTML 字符串 ] 时：  
会对我们创建这个 html 标签对象
- 3、传入参数为 [ 选择器字符串 ] 时：  
`$("#id 属性值");` id 选择器，根据 id 查询标签对象  
`$("标签名");` 标签名选择器，根据指定的标签名查询标签对象  
`$(".class 属性值");` 类型选择器，可以根据 class 属性查询标签对象
- 4、传入参数为 [ DOM 对象 ] 时：  
会把这个 dom 对象转换为 jQuery 对象

## 4、jQuery 对象和 dom 对象区分

### 4.1、什么是 jQuery 对象，什么是 dom 对象

#### Dom 对象

- 1.通过 `getElementById()` 查询出来的标签对象是 Dom 对象
- 2.通过 `getElementsByName()` 查询出来的标签对象是 Dom 对象
- 3.通过 `getElementsByTagName()` 查询出来的标签对象是 Dom 对象
- 4.通过 `createElement()` 方法创建的对象，是 Dom 对象

DOM 对象 Alert 出来的效果是： `[object HTML 标签名 Element]`

#### jQuery 对象

- 5.通过 JQuery 提供的 API 创建的对象，是 JQuery 对象
- 6.通过 JQuery 包装的 Dom 对象，也是 JQuery 对象
- 7.通过 JQuery 提供的 API 查询到的对象，是 JQuery 对象

jQuery 对象 Alert 出来的效果是： `[object Object]`

## 4.2、问题：jQuery 对象的本质是什么？

jQuery 对象是 dom 对象的数组 + jQuery 提供的一系列功能函数。

## 4.3、jQuery 对象和 Dom 对象使用区别

jQuery 对象不能使用 DOM 对象的属性和方法

DOM 对象也不能使用 jQuery 对象的属性和方法

## 4.4、Dom 对象和 jQuery 对象互转

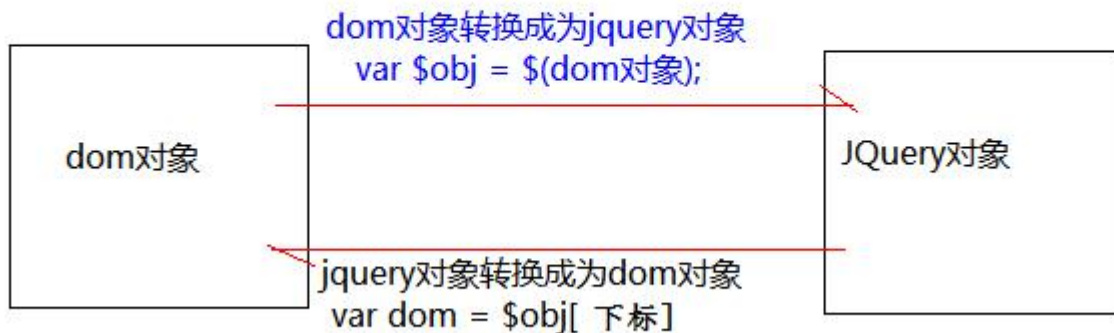
### 1、dom 对象转化为 jQuery 对象（\*重点）

- 1、先有 DOM 对象
- 2、\$( DOM 对象 ) 就可以转换成为 jQuery 对象

### 2、jQuery 对象转为 dom 对象（\*重点）

- 1、先有 jQuery 对象
- 2、jQuery 对象[下标]取出相应的 DOM 对象

### Jquery对象和Dom对象相互转换



## 5、jQuery 选择器 (\*\*\*\*\*重点)

### 5.1、基本选择器 (\*\*\*\*重点)



**#ID** 选择器：根据 id 查找标签对象

**.class** 选择器：根据 class 查找标签对象

**element** 选择器：根据标签名查找标签对象

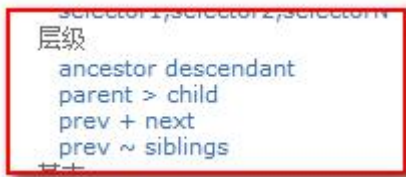
**\*** 选择器：表示任意的，所有的元素

**selector1, selector2** 组合选择器：合并选择器 1，选择器 2 的结果并返回

p.myClass

表示标签名必须是 p 标签，而且 class 类型还要是 myClass

### 5.2、层级选择器 (\*\*\*\*重点)



**ancestor descendant** 后代选择器：在给定的祖先元素下匹配所有的后代元素

**parent > child** 子元素选择器：在给定的父元素下匹配所有的子元素

**prev + next** 相邻元素选择器：匹配所有紧接在 prev 元素后的 next 元素

**prev ~ siblings** 之后的兄弟元素选择器：匹配 prev 元素之后的所有 siblings 元素

## 5.3、过滤选择器

基本过滤器：



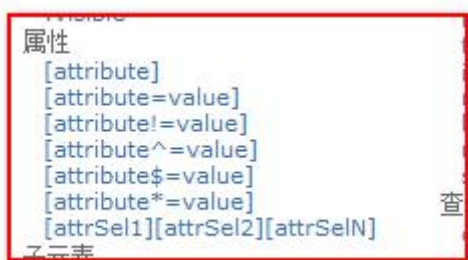
<b>:first</b>	获取第一个元素
<b>:last</b>	获取最后一个元素
<b>:not(selector)</b>	去除所有与给定选择器匹配的元素
<b>:even</b>	匹配所有索引值为偶数的元素，从 0 开始计数
<b>:odd</b>	匹配所有索引值为奇数的元素，从 0 开始计数
<b>:eq(index)</b>	匹配一个给定索引值的元素
<b>:gt(index)</b>	匹配所有大于给定索引值的元素
<b>:lt(index)</b>	匹配所有小于给定索引值的元素
<b>:header</b>	匹配如 h1, h2, h3 之类的标题元素
<b>:animated</b>	匹配所有正在执行动画效果的元素

内容过滤器：



<b>:contains(text)</b>	匹配包含给定文本的元素
<b>:empty</b>	匹配所有不包含子元素或者文本的空元素
<b>:parent</b>	匹配含有子元素或者文本的元素
<b>:has(selector)</b>	匹配含有选择器所匹配的元素

## 属性过滤器:



**[attribute]**

匹配包含给定属性的元素。

**[attribute=value]**

匹配给定的属性是某个特定值的元素

**[attribute!=value]**

匹配所有不含有指定的属性，或者属性不等于特定值的元素。

**[attribute^=value]**

匹配给定的属性是以某些值开始的元素

**[attribute\$=value]**

匹配给定的属性是以某些值结尾的元素

**[attribute\*=value]**

匹配给定的属性是以包含某些值的元素

**[attrSel1][attrSel2][attrSelN]**

复合属性选择器，需要同时满足多个条件时使用。

## 表单过滤器:



**:input**

匹配所有 input, textarea, select 和 button 元素

**:text**

匹配所有 文本输入框

**:password**

匹配所有的密码输入框

**:radio**

匹配所有的单选框

**:checkbox**

匹配所有的复选框

**:submit**

匹配所有提交按钮

**:image**

匹配所有 img 标签

**:reset**

匹配所有重置按钮

**:button**

匹配所有 input type=button <button>按钮

**:file**

匹配所有 input type=file 文件上传

**:hidden**

匹配所有不可见元素 display:none 或 input type=hidden



## 表单对象属性过滤器：

表单对象属性  
:enabled  
:disabled  
:checked  
:selected

:enabled 匹配所有可用元素  
:disabled 匹配所有不可用元素  
:checked 匹配所有选中的单选，复选，和下拉列表中选中的 option 标签对象  
:selected 匹配所有选中的 option

## 6、jQuery 元素筛选

### 筛选

#### 过滤

eq(index|-index)  
first()  
last()  
hasClass(class)  
filter(expr|obj|ele|fn)  
is(expr|obj|ele|fn)<sup>1.6\*</sup>  
map(callback)  
has(expr|ele)  
not(expr|ele|fn)  
slice(start,[end])

#### 查找

children([expr])  
closest(expr,[con]|obj|ele)<sup>1.6\*</sup>  
find(expr|obj|ele)  
next([expr])  
nextall([expr])  
nextUntil([exp|ele],[fil])<sup>1.6\*</sup>  
offsetParent()  
parent([expr])  
parents([expr])  
parentsUntil([exp|ele],[fil])<sup>1.6\*</sup>  
prev([expr])  
prevall([expr])  
prevUntil([exp|ele],[fil])<sup>1.6\*</sup>  
siblings([expr])

eq()	获取给定索引的元素	功能跟 :eq() 一样
first()	获取第一个元素	功能跟 :first 一样
last()	获取最后一个元素	功能跟 :last 一样
filter(exp)	留下匹配的元素	
is(exp)	判断是否匹配给定的选择器，只要有一个匹配就返回，true	
has(exp)	返回包含有匹配选择器的元素的元素	功能跟 :has 一样
not(exp)	删除匹配选择器的元素	功能跟 :not 一样
children(exp)	返回匹配给定选择器的子元素	功能跟 parent>child 一样
find(exp)	返回匹配给定选择器的后代元素	功能跟 ancestor descendant 一样



next()	返回当前元素的下一个兄弟元素	功能跟 prev + next 功能一样
nextAll()	返回当前元素后面所有的兄弟元素	功能跟 prev ~ siblings 功能一样
nextUntil()	返回当前元素到指定匹配的元素为止的后面元素	
parent()	返回父元素	
prev(exp)	返回当前元素的上一个兄弟元素	
prevAll()	返回当前元素前面所有的兄弟元素	
prevUnit(exp)	返回当前元素到指定匹配的元素为止的前面元素	
siblings(exp)	返回所有兄弟元素	
add()	把 add 匹配的选择器的元素添加到当前 jquery 对象中	



