

1 jQuery

一、jQuery概述

1、什么是jQuery(了解)

jQuery: JavaScript Query JS查询
 jQuery是一个轻量的、免费开源的JS函数库
 轻量的: 少(小)、简单易用
 免费开源: 不收费、源码对外开放
 函数库: 在JS底层的很多属性, 在jQuery中都变为了函数

可以极大的简化JS代码

2、jQuery的优势(了解)

既然jQuery是JS的框架, 也就是说jQuery能实现的功能, JS也可以实现, 那为什么要学习jQuery?

- (1) 可以极大的简化JS代码
- (2) 可以像CSS选择器一样非常方便的获取元素


```
document.getElementById("div1");
$("#div1")
$(".c1")
$("div")
```
- (3) 可以通过操作CSS属性控制页面的效果


```
$("#div").css("background", "red");
$("#div").css("border", "2px solid red");
$("#div").css("font-size", "28px");
```
- (4) 可以兼容常用的浏览器

JS中的某些方法或者属性在某些浏览器中的不兼容的
 innerText、replaceNode()、removeNode() 等

常用浏览器: 谷歌/火狐/欧朋/苹果/

...

3、jQuery引入

由于jQuery底层其实都是JS代码, 所以要引入的jQuery库文件, 其实就是一个JS文件.

```
<script src="js文件的路径"></script>
```

4、\$符号简介(了解)

\$ == jQuery
 调用\$() 等价于调用jQuery() 函数

\$("#div"): 获取所有的div元素, 返回值是一个jQuery对象
 这个jQuery对象可以理解为一个集合(包含了所匹配的元素)

```
$("#div").css("xx", "xx");
$("#div").remove();
$("#div").replaceWith("<span>我是来替换的!</span>")
```

5、文档就绪事件函数

什么时候需要用到文档就绪事件函数?

比如在获取元素时, 获取元素的代码执行的时机, 比元素加载的时机还要早, 像这种情况, 必然是获取不到元素的!

此时就可以让获取元素的代码放在一个文档就绪事件函数中, 等整个html文档加载完之后再获取元素, 就没有问题了!!

在jQuery中提供了一个文档就绪事件函数:

```
$(function() {
    //在文档加载完之后立即触发执行...
});
等价于window.onload事件: 在整个html文档加载完成之后立即触发
window.onload = function() {
    //在文档加载完之后立即触发执行...
}
```

示例:

```
<script>
/* 文档就绪事件函数: 这个函数会在浏览器将
整个html文档加载完(即加载完最后一个标签)
之后才会执行!!! */
/*window.onload = function() {
    //获取h1标签内的内容
    //1. 获取id为h1的标签
    var h1 = document.getElementById("h1");
    //2. 获取h1元素的内容
    alert( h1.innerHTML );
}*/

/* jQuery也提供了文档就绪事件函数 */
$(function() {
    //获取h1标签内的内容
    //1. 获取id为h1的标签
    var h1 = document.getElementById("h1");
    //2. 获取h1元素的内容
    alert( h1.innerHTML );
});
</script>
...
<body>
    <h1 id="h1">引入jQuery示例...</h1>
</body>
```

6、JS对象转换为jQuery对象

JS对象只能调用JS提供的属性或方法, 不能调用jQuery提供的属性或方法

比如: 在获取h1元素的内容例子中

```
var h1 = document.getElementById("h1");//返回的是一个JS对象
alert( h1.innerHTML );//正确!!!
```

```
//html();//jQuery提供的函数, 必须由jQuery对象调用才可以
//alert( h1.html() );//这是错的!!!
```

```
alert( $(h1).html() );//正确!!!
```

二、jQuery选择器(重点!!!)

在html网页中帮我们选中元素的一门技术

1、基本选择器

(1) 标签名(元素名)选择器

`$("div")` -- 匹配所有的div元素

(2) class选择器

`$(".mini")` -- 匹配所有class值为mini的元素

`$("div.mini")` -- 匹配所有class值为mini的div元素

(3) ID选择器

`$("#one")` -- 匹配id值为one的元素

2、层级选择器

`$("div span")` -- 匹配div内部的所有span后代元素

`$("div>span")` -- 匹配div内部的所有span子元素

`$("div+span")` -- 匹配div后面紧邻的span兄弟元素

`$("div~span")` -- 匹配div后面所有的span兄弟元素

`$("div").find("span");` -- 匹配div内部的所有span后代元素(等价于 `$("div span")`)

`$("div").parents();` -- 匹配div所有的祖先元素们

`$("div").parent();` -- 匹配div的父元素

```
<body>
    <table>
        <tr>
```

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

\$("#div").next("span") -- 匹配div后面紧邻的span兄弟元素
\$("#div").nextAll("span") -- 匹配div后面所有的span兄弟元素
\$("#div").prev("span") -- 匹配div前面紧邻的span兄弟元素
\$("#div").prevAll("span") -- 匹配div前面所有的span兄弟元素
\$("#div").siblings("span") -- 匹配div前后所有的span兄弟元素

3、基本过滤选择器

\$("#div:first") -- 匹配所有div中的第一个
\$("#div:eq(0)") -- 匹配所有div中的第一个
\$("#div:last") -- 匹配所有div中的最后一个
\$("#div:eq(n)") -- 匹配所有div中第n+1个

\$(".one") -- 匹配所有class为one的元素
\$(".:not(.one)") -- 匹配所有class不为one的元素
\$("#div:not(.one)") -- 匹配所有class不为one的div元素

三、jQuery案例

2 AJAX

3 JSON
