第二章-jQuery框架使用准备

2.1 jQuery框架和JavaScript加载模式对比

jQuery框架的加载模式

```
<script>
    window.onload = function () {
        console.log("window.onload--1")
    };

    window.onload = function () {
        console.log("window.onload--2")
    }
</script>
```

打印结果：window.onload——2

JavaScript的加载模式

```
<script src="jquery-3.2.1.js"></script>
<script>
    $(document).ready(function () {
        console.log("$().ready--1")
    });

    $(document).ready(function () {
        console.log("ready--2")
    })
</script>
```

打印结果：ready——1 ready——2

两种加载模式对比

① 执行时机不相同

```
window.onload方法需要等所有的资源（CSS\JS\图片等）都加载完毕后执行包裹代码。
jQuery框架的ready方法等DOM结构加载完毕后执行包裹代码。
```

② 执行次数不相同

```
window.onload方法，N次只会执行一次，后面的会把前面的覆盖。
jQuery框架的ready方法，N次会执行N次，不存在覆盖的问题。
```

③ 简写方式不相同

```
jQuery框架中的加载方式可以简写为:
$().ready(function () {})
$(function () {})
```

2.2 jQuery框架解决冲突

在实际的开发中，我们的项目中可能需要用到并引入多个第三方框架，如果这些框架本身在设计的时候，没有命名空间的约束，那么库与库之间发生冲突将在所难免。

jQuery框架在设计的使用，使用闭包的形式在所有的代码都封装到一个**立即调用函数**中，对外仅仅提供了美元符号和jQuery作为框架的入口。

jQuery当中所有的操作都是使用**美元符号**或者是**jQuery对象**进行的。假如，我们在使用jQuery框架之前已经在页面的代码中用到了美元符号，那么这种情况下，我们再按照常规的方式使用jQuery就可能会发生错误。

为了避免这种情况的发生，jQuery框架使用noConflict方法,可以在使用之前把美元符号替换成其它的标识符，相当于是给jQuery对象换个其他的名字。

代码示例

```
<script src="jquery-3.1.1.js"></script>
<script>
    var $ = "文顶顶";
    $(function () {
        console.log("DOM加载完毕");
    })
</script>
```

代码说明

如果直接执行上面的代码，那么会报错。

报错信息： `Uncaught TypeError: $ is not a function`

报错原因：美元符号被定义为字符串，jQuery框架中美元符号被覆盖。

解决冲突（给jQuery框架安排新的代言人）

```
<script>
    //在$符号被定义之前使用noConflict方法来重新设置名称
    var jq = $.noConflict();

    // $符号可能被定义为字符串或其他数据
    var $ = "文顶顶";

    //解决冲突之后,使用新设置的名称来操作
    jq(function () {
        console.log("DOM加载完毕");
    })
</script>
```

2.3 jQuery对象和DOM对象的相互转换

DOM对象：每个HTML页面都是一个 DOM对象(Document Object Model ，文本对象模型)，通过传统的JavaScript方法访问页面中的元素，就是访问 DOM 对象。

jQuery对象：在 jQuery框架中，通过本身自带的方法获取页面元素的对象，称为 jQuery 对象；ps: 其实jQuery本身只是个工厂函数，我们通常意义上所说的jQuery实例对象其实是jQuery的原型对象上面的init方法创建出来的实例对象。即 jQuery对象 = new jQuery.prototype.init() ，只是因为init方法和jQuery构造函数共用相同的原型对象，所以我们才会称init构造函数创建出来的对象为jQuery实例对象 。

代码示例

```
<body>
<div class="box">我是div</div>
<script src="jquery-3.2.1.js"></script>
<script>
    //通过DOM提供的api获取页面中所有class为box的标签
    var oDiv = document.getElementsByClassName("box");
    console.log(oDiv);

    //通过jQuery提供的方法获取页面中所有class为box的标签
    var ojQueryObj = $("div");
    console.log(ojQueryObj);
</script>
</body>
```

DOM对象和jQuery对象的转换

DOM对象可以理解为标签对象，我们在操作这些标签的时候，有很多标签自带的方法可以使用，如innerHTML、innerText属性，或者是appendChild方法等。

jQuery对象可以理解为jQuery初始化方法这个构造函数创建的实例化对象，因为它的原型对象为jQuery.prototype，因此所有的jQuery实例对象都可以访问jQuery原型对象上面的成员[属性或方法]，如html、text方法等。

注意： DOM对象不能直接访问jQuery原型对象上面的成员，jQuery对象也不能直接访问标签对象上面的成员，如需访问则应该先进行转换。

DOM标签对象 -> jQuery实例对象 $(DOM标签对象)
jQuery实例对象 -> DOM标签对象 jQuery对象.get(index) | jQuery对象[index]

代码示例

```
<body>
<div class="box">我是div</div>
<script src="jquery-3.1.1.js"></script>
<script>
    //通过DOM提供的api获取页面中所有class为box的标签
    var oDiv = document.getElementsByClassName("box")[0];
```

```
    //通过jQuery提供的方法获取页面中所有class为box的标签
    var ojQueryObj = $("div");

    //DOM -> jQuery
    console.log($(oDiv).html());
    //jQuery -> DOM
    console.log(ojQueryObj.get(0));
    console.log(ojQueryObj[0]);
</script>
</body>
```

- Posted by 博客园·文顶顶 ~ 文顶顶的个人博客_花田半亩
- 联系作者 简书·文顶顶 新浪微博·Coder_文顶顶
-