

第五章-jQuery框架动画特效

5.1 jQuery动画特效说明

jQuery框架中为我们封装了众多的动画和特效方法，只需要调用对应的动画方法传递合适的参数，就能够方便的实现一些炫酷的效果，而且jQuery框架还支持自定义各种动画效果。

jQuery中的动画效果主要有以下方法

- ① 显示和隐藏动画
- ② 展开和收起动画
- ③ 淡入和淡出动画
- ④ 自定义动画效果

5.2 显示和隐藏动画

jQuery框架中为我们提供了专门的方法来控制让标签显示或者是隐藏。标签的显示和隐藏在开发中相对来说是比较常见的操作，如果使用原生的JavaScript代码来控制标签的显示或者是隐藏，那么我们主要通过控制该标签的display属性值来实现。

jQuery中控制标签显示和隐藏的动画方法

- ① `show()` 控制让标签显示的动画方法
- ② `hide()` 控制让标签隐藏的动画方法
- ③ `toggle()` 控制让标签显示|隐藏效果切换的动画方法

ex：以原生的方法来控制标签的显示或隐藏

```
<!DOCTYPE html>
<html lang="en">
<head>
  <!-- .... -->
  <style>
    #demoID{
      width: 200px;
      height: 50px;
      background: red;
    }
  </style>
  <script src="jquery-3.1.1.js"></script>
</head>
<body>
<div id="demoID"></div>
<script>
  //控制标签显示或隐藏的方式(1)
  //document.getElementById("demoID").style.display = "none";
```

```
//document.getElementById("demoID").style.display = "block";

//控制标签显示或隐藏的方式(2)
//$("#demoID").css("display","none");
$("#demoID").css("display","block");
</script>
</body>
</html>
```

显示和隐藏动画方法语法

`$("#selector").show(speed,callback)`

`$("#selector").hide(speed,callback)`

参数说明：

第一个参数：可选的参数。speed表示执行动画的速度，该速度可以使用系统默认提供的值，也可以自己以数字的形式传入。

系统默认提供的值有：“slow”、“normal”、“fast”，对应的速度分别为0.6秒、0.4秒和0.2秒。

自己以数字的形式传递则：传递如1000|3000类似的值，单位为毫秒，如果传递1000表示动画的执行速度为1秒。

第二个参数：可选的参数。callback为动画完成时执行的回调函数，该函数每个元素执行一次。

切换动画方法语法

调用方式[1] => `$("#selector").toggle()`

调用方式[2] => `$("#selector").toggle(speed,callback)`

方法参数说明：

(1) 调用方式[1]，不传递参数。切换当前元素（标签）的可见状态，如果当前元素的状态为不可见，则切换为可见，如果当前元素的状态为可见，则切换为不可见。

(2) 调用方式[2]，第一个可选的参数为动画执行的速度，第二个可选的参数为动画执行完后执行的回调函数。

总结

show方法和hide方法用来控制标签的显示或者是隐藏，内部的实现逻辑是同时改变标签的宽度、高度和透明度。

5.3 展开和收起动画

jQuery框架中，为我们提供实现滑动效果的方法，slideDown和slideUp方法分别可以用来控制标签展开和收起。

展开和收起动画方法语法

```
slideDown(speed, callBack);
slideUp(speed, callBack)
slideToggle(speed,callBack)
```

方法解释

slideDown 方法的功能是设置让指定标签的高度从顶部向底部增加，以呈现出一种展开的动画效果，元素的其他属性不会发生任何变化。

sldeUp 方法的功能是设置让指定标签的高度从底部向顶部减小，仅仅改变标签的高度，其他的属性并不会改变。

slideToggle 方法用来切换所选择元素的高度，如果当前标签的高度不为0，那么调用该方法后就会把标签的高度过渡为0，实现收起的效果，否则就实现展开的效果。

注意：无论是展开还是收起的动画方法，他们的动画效果都仅仅只会改变（增加或减小）标签的高度。

参数说明

第一个参数：动画执行的时间，同show方法。

第二个参数：动画执行完毕的回调函数，可以省略，同show方法。

代码示例

```
<body>
<div>我是div</div>
<button>展开</button>
<button>收起</button>
<button>切换</button>
<script>
    $(function () {
        $("button").eq(0).click(function () {
            $("div").slideDown(2000,function () {
                console.log("展开");
            });
        })

        $("button").eq(1).click(function () {
            $("div").slideUp(2000);
        })

        $("button").eq(2).click(function () {
            $("div").slideToggle(1000,function () {
                alert("切换动画指定完毕")
            });
        })
    })
</script>
</body>
```

HTML

5.4 淡入和淡出动画

jQuery框架中还提供了淡入和淡出的动画方法，这两个方法分别是fadeIn和fadeOut。

淡入和淡出动画方法语法

```
fadeIn(speed, callback);  
fadeOut(speed, callback)  
fadeToggle(speed, callback)  
fadeTo(speed, opacity, callback)
```

方法说明：

淡入淡出方法的内部实现原理是控制标签的透明度，通过改变选中标签的透明度来实现淡入和淡出的动画效果，并不修改其他的属性。

fadeIn()方法 的作用是淡入动画，即改变标签的透明度让标签慢慢的显示出来。

fadeOut()方法 的作用是淡出动画，即改变标签的透明度让标签慢慢的消失（透明度为0）。

fadeToggle()方法 的作用是切换动画，如果当前标签的透明度不为0，那么就执行淡出动画，否则就执行淡入动画。

fadeTo()方法 的作用是淡入到，即使用动画效果实现将标签的透明度设置为指定的值。

备注：标签透明度的取值范围为0.0~1.0。

参数说明：

speed参数：动画执行的速度，单位为毫秒。

callback参数：动画执行完毕的回调函数，可选。

opacity参数：指定的标签透明度（0.0~1.0）。

代码示例

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <title>Title</title>  
  <script src="JS/jquery-3.2.1.js"></script>  
  <style>  
    div{  
      width: 400px;  
      height: 200px;  
      background: red;  
    }  
  </style>  
</head>  
<body>  
<div>我是div</div>  
<button>淡入</button>  
<button>淡出</button>  
<button>切换</button>  
<button>惦记我</button>
```

HTML

```

<script>

//fadeIn:淡入
//fadeOut:淡出
//fadeToggle切换
//fadeTo:淡入淡出到...0.5

$(function () {
    //01 获取页面中指定的按钮,添加点击事件
    $("button:eq(0)").click(function () {
        //02 点击按钮后,获取页面中的div标签,设置动画
        //第一个参数:速度
        //第二个参数:回调
        $("div").fadeIn(2000,function () {
            alert("显示完成");
        });
    })

    $("button:eq(1)").click(function () {
        $("div").fadeOut(1000,function () {
            alert("淡出")
        });
    })

    $("button:eq(2)").click(function () {
        $("div").fadeToggle();
    })

    $("button:eq(3)").click(function () {
        //第一个参数:速度
        //第二个参数:目标值
        //第三个参数:回调
        $("div").fadeTo(1000,0.5,function () {
            alert("执行动画完毕")
        });
    })

})

</script>

</body>
</html>

```

5.5 自定义动画

jQuery框架中本身已经为我们封装好了一些简单的控制标签宽高、透明度相关的方法，如显示和隐藏、展开和收起、淡入和淡出，除了这些方法之外，jQuery还为我们提供了 `animate ()` 方法，允许我们自定义动画效果，通过一些设置我们可以实现更加复杂的动画效果，

自定义动画的语法

`animate(params,speed,easing,callBack)`

参数说明：

第一个参数：params是一个对象。在该对象中以键值对的方式来要控制的属性样式和对应的值表示。

第二个参数：speed速度，可以是默认字符串中的某个（“slow” “normal” “fast”）或者是自定义的数字。

第三个参数：easing为动画插件使用的可选参数，用来控制动画的表现效果，通常有linear和swing等固定值。

第四个参数：动画执行完毕后的回调函数。

代码示例

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
  <style>
    div{
      width: 200px;
      height: 50px;
      background: red;
    }
  </style>
</head>
<body>
<!--
  animate({},timer,fn)
  第一个参数: 目标对象
  第二个参数: 速度
  第三个参数: 回调函数
-->

<div>我是个好人</div>
<button>
  喜欢我就点我吧
</button>
<button>
  喜欢我就点我吧2
</button>
<script src="js/jquery-3.2.1.js"></script>
<script>
  $(function () {
    $("button:eq(0)").click(function () {
      $("div").animate({
        width:"+=50",
        height:"+=100"
      },1000);
    })
    $("button:eq(1)").click(function () {
      //01 直接设置目标值: 速度和回调函数可以被省略
      //      $("div").animate({
      //        width:400,
      //        height:100
      //      },2000,function () {
```

```

//            alert("执行完毕");
//        })

//02 第二种用法

//            $("div").animate({
//                width:"+=50",
//                height:"+=100"
//            },1000);

//03 动画切换(如果有值那么就设置为0,否则就恢复)
$("div").animate({
    width:"toggle",
    height:"+=100"
},1000);
    })
})
</script>
</body>
</html>

```

动画队列、动画停止和动画延迟

动画队列：如果某个标签身上要调用多个动画相关方法，即需要展示多个动画效果，那么所有的这些动画效果并不会同一时刻发生，而是需要在队列中排队，然后按照队列中动画效果的顺序依次展现。

动画停止：在执行动画的时候，可以通过stop（）方法来停止动画。停止动画的语法为：

stop (clearQueue,gotoEnd)

方法和参数说明：

stop方法的功能是停止指定标签中正在执行的动画，其中第一个参数clearQueue为可选参数，传递一个布尔类型的值，表示是否停止正在执行的动画。第二个参数gotoEnd也是可选参数，传递一个布尔类型的值，表示是否立即完成正在执行的动画。

动画延迟：设置一个延时的值来推迟后续队列中动画的执行，可以传递延迟的具体时间，单位为毫秒。

代码示例01

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <style>
        div{
            width: 200px;
            height: 200px;
            background: yellow;
        }
    </style>
</head>
<body>
<div>

```

HTML

```

</div>
<button>按钮</button>
<script src="js/jquery-3.2.1.js"></script>
<script>
    $(function () {
        $("button").click(function () {
            $("div").animate({
                width: "50px"
            })
                .animate({
                    height: 50
                })
                .delay(2000)
                .animate({
                    height: 300
                })
                .animate({
                    width: 400
                })
        })
    })
</script>
</body>
</html>

```

代码示例02

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">

    <title>Title</title>
    <style>
        div{
            width: 50px;
            height: 50px;
            background: green;
        }
    </style>
</head>
<body>
<div>
</div>
<button>按钮</button>
<script src="js/jquery-3.2.1.js"></script>
<script>
    $(function () {

        //01 当页面加载完毕就执行动画效果
        $("div").animate({
            width: "400px"
        }, 2000)
        .animate({
            height: 200

```

HTML


```

        },1000)
        .delay(1000)
        .animate({
            height:50
        })
        .animate({
            width:50
        })

//02 当点击按钮的时候停止动画
$("button").click(function () {

    //01 没有传递参数:结束当前的动画,继续执行后面的动画
    //$("#div").stop();

    //02 传递1参数:
    // true:所有的动画全部结束
    // false:结束当前的动画,继续执行后面的动画
    //$("#div").stop(true);
    //$("#div").stop(false);

    //03 传递2参数:
    //true true : 立刻结束到达当前动画的终点,后面的不再执行
    //true false:所有的动画全部结束
    //false false:结束当前的动画,继续执行后面的动画
    //false true : 立刻结束到达当前动画的终点,后面的继续执行
    //$("#div").stop(true,true);
    //$("#div").stop(true,false);
    //$("#div").stop(false,false);
    $("#div").stop(false,true);

})
})
</script>
</body>
</html>

```

- Posted by 博客园·文顶顶 | 花田半亩
- 联系作者 简书·文顶顶 新浪微博·Coder_文顶顶
- 原创文章，版权声明： 自由转载-非商用-非衍生-保持署名 | 文顶顶