

# jQuery

---

## 一、选取DOM节点

### 1. 基础选择器

id选择器 `$("#id名")`  
全选选择器 `$("*")`  
类选择器 `$(".class名")`  
标签选择器 `$("div")`  
并集选择器 `$(div,p,li)`

### 2. 层级选择器

子代选择器 `$("ul>li")`  
后代选择器 `$("ul li")`

### 3. 筛选选择器

`$("ul li:first")` // 获取第一个li元素  
`$("ul li:last")` // 获取最后一个li元素  
`$("ul li:eq(2)")` // 获取到的li元素中，选择索引号为2的元素，索引号从0开始  
`$("ul li:odd")` // 获取到的li元素中，选择索引号为奇数的元素  
`$("ul li:even")` // 获取到的li元素中，选择索引号为偶数的元素

### 4. 筛选方法

`$("li").parent()` // 查找父级  
`$("ul").children("li")` // 相当于 `$("ul>li")` 最近一级亲儿子  
`$("ul").find("li")` // 相当于 `$("ul li")` 后代选择器  
`$(.first).siblings("li")` // 查找兄弟节点，不包括自己本身  
`$(.first).nextAll()` // 查找当前元素之后的所有同辈元素  
`$(.last).prevAll()` // 查找当前元素之前的所有同辈元素  
`$("div").hasClass("protected")` // 检查当前元素是否含有某个特定的类，如果有返回true  
`$("li").eq(2) === $("li:eq(2)")`;  
例1：  
html:  
`<div class="yeye">`  
`<div class="father">`  
`<div class="son">儿子</div>`  
`</div>`  
`</div>`

```

<div class="nav">
<p>我是p1</p>
<div>
<p>我是p2</p>
</div>
</div>
jQuery:
$(function(){
console.log($(".son").parent());//返回的是最近父级元素father
$(".nav").children("p").css("color","red");//我是p1变红
$(".nav").find("p").css("color","red");//我是p1我是p2都变红
}) 例2：html:
<ol>
<li>我是ol的li</li>
<li>我是ol的li</li>
<li class="item">我是ol的li</li>
<li>我是ol的li</li>
</ol>
<ul>
<li>我是ol的li</li>
<li>我是ol的li</li>
<li>我是ol的li</li>
<li>我是ol的li</li>
</ul>
jQuery:
$(function(){
$(".ol .item").siblings("li").css("color","red");//item的兄弟变红
$(".ul li:eq(2)").css("color","blue");//让ul中第三个li标签变红
$(".ul li").eq(2).css("color","blue");
var index = 2;
$(".ul li:eq("+index+"").css("color","blue");//选择器
$(".ul li").eq(index).css("color","blue");//选择方法
})
例3：html:
<div class="current"></div>
jQuery:
console.log($(".div").hasClass("current"));//true

```

## 二、实现简单动画

### 1. 显示与隐藏

hide/show([speed],[easing],[fn])

切换: toggle([speed],[easing],[fn])

注(1)参数都可以省略，无动画直接显示

(2)speed:三种预定速度之一的字符串("slow","normal"or "fast")或表示动画时长的毫秒数值1000，按

照某个速度来显示或隐藏

(3)easing:用来指定切换效果，默认是“swing”（慢快慢）可用参数“linear”（匀速）

(4)fn:回调函数，在动画完成时执行的函数，每个元素执行一次\

html:

```
<button>显示</button>
```

```
<button>隐藏</button>
```

```
<button>切换</button>
```

```
<div></div>
```

```
jQuery:$(function(){
  $("button").eq(1).click(function(){
    $("div").hide(1000,function(){
      alert(1);//动画消失后弹出对话框
    });
  });
  $("button").eq(0).click(function(){
    $("div").hide(1000,function(){
      alert(1);//动画消失后弹出对话框
    });
  });
  $("button").eq(2).click(function(){
    $("div").toggle(1000,function(){
      alert(1);//动画消失后弹出对话框
    });
  });
});
```

## 2. 滑动效果

下拉slideDown([speed],[easing],[fn]])

上拉slideUp([speed],[easing],[fn]])

切换slideToggle([speed],[easing],[fn]])

html:

```
<button>下拉滑动</button>
```

```
<button>上拉滑动</button>
```

```
<button>切换滑动</button>
```

```
<div></div>
```

css:

```
div{
  width:150px;
  height:300px;
  background-color:pink;
  display:none;
}
```

jQuery:

```
$(function(){
  $("button").eq(0).click(function(){
```

```

$("div").slideDown();
})
$("button").eq(1).click(function(){
    $("div").slideUp();
})
$("button").eq(2).click(function(){
    $("div").slideToggle(500);
})
})

```

### 3. 事件切换

```

hover([over],out)
over:鼠标移到元素上要触发的函数=mouseenter
out:鼠标移出元素要触发的函数=mouseleave
<ul class="nav">
<li>
<a href="#">微博</a>
<ul>
<li>
<a href="#">私信</a>
</li>
<li>
<a href="#">评论</a>
</li>
<li>
<a href="#">@我</a>
</li>
</ul>
jQuery:
$(function(){
    $(".nav>li").hover(function(){//鼠标移到元素上要触发的函数
        $(this).children("ul").slideDown(200);
    },function(){//鼠标移出元素要触发的函数
        $(this).children("ul").slideUp(200);
    })
})

```

### 4. 停止动画排队

动画或效果队列:动画或者效果一旦触发就会执行，如果多次触发，就会造成多个动画或者效果排队执行

```

$(".nav>li").hover(function(){
    $(this).children("ul").stop().slideToggle();
})

```

事件切换hover如果只写一个函数，那么鼠标经过和鼠标离开都会触发这个函数，经过时下拉出来，离开时上拉上去

stop()不管触发多少次，先把上一次动画停止，不管走没走完，先执行当前slideToggle

**stop方法必须写在动画前面**

## 5. 淡入淡出

fadeIn([speed],[easing],[fn]])

fadeOut([speed],[easing],[fn]])

fadeToggle([speed],[easing],[fn]])

fadeTo([[speed],opacity,[easing],[fn]])

opacity:透明度，必须写，取值0到1之间speed必须写

html:

<button>淡入</button>

<button>淡出</button>

<button>切换</button>

<button>修改透明度</button>

css:

div{

width:150px;

height:300px;

background-color:pink;

display:none;

}

jQuery:

\$(function(){

\$("#button").eq(0).click(function(){

\$("#div").fadeIn(1000);

})

\$("#button").eq(1).click(function(){

\$("#div").fadeOut(1000);

})

\$("#button").eq(2).click(function(){

\$("#div").fadeToggle(1000);

})

\$("#button").eq(3).click(function(){

\$("#div").fadeTo(1000,0.5);//变成0.5的透明度

});

});

## 6. 自定义动画

animate(params,[speed],[easing],[fn])

params:想要修改的样式属性，以**对象形式**传递必须写属性名，可以不用带引号，如果是复合属性，则需要采取**驼峰命名法**borderLeft，其余参数都可以省略

```

html:
<button>动起来</button>
<div></div>
css:
div{
width:200px;
height:200px;
background-color:pink;
position:absolute;
}
jQuery:
$(function(){
$("button").click(function(){
$("div").animate({//直接调用函数
left:200px;
top:300px;
opacity:0.4;
width:500px;
},500);//半秒内
})
})

```

### 三、设置或获取元素固有属性值prop()

所谓元素固有属性，就是元素本身自带的属性，比如a标签元素里面的href比如input元素里面的type

html:

```
<a href="#" title="都挺好">都挺好</a>
```

jQuery:

```

$(function(){
console.log($(".a").prop("href"));//获取属性值
$(".a").prop("title","我们都挺好");//修改属性值
})

```

HTML :

```
<input type="checkbox" checked> //checked有两个值true（选中）和false（未选中）
```

jQuery:

```

$("input").change(function(){
console.log($(this).prop("checked"))
})

```

### 四、设置或获取元素自定义属性值attr()(index,data-index,class)

html :

```
<div index="1">
```

```

console.log($(".div").attr("index")); //获取属性值
$(".div").attr("index",4); //设置属性值

```

## 五、数据缓存data()

data()方法可以在指定的元素上存取数据，**并不会修改DOM元素结构**，一旦页面刷新，之前存放的数据，都将被移除

html：

```
<span>123</span>
```

```
$("span").data("uname","andy");
```

```
console.log($("span").data("uname"));//andy
```

获取data-index h5自定义属性，不用写data-，而且返回的是数字型（色不一样）

```
console.log($("div").data("index"));
```

## 六、遍历元素

隐式迭代是对同一类元素做了同样的操作，如果想要给同一类元素做不同操作，就需要用到遍历

(1) \$("div").each(function(index,domEle){语句；})

(i)each()方法遍历匹配的每个元素，主要用DOM处理

(ii)里面的回调函数有2个参数，index是每个元素的索引号；domEle是每个DOM对象，不是jQuery对象

(iii)要想使用jQuery方法，需要把这个dom元素转换为jQuery对象\$(domEle)

```
$("div").each(function(index,domEle){
```

```
console.log(index);
```

//回调函数第一个参数一定是索引号可以自己指定索引号名称

```
console.log(domEle);
```

//回调函数，第二个参数一定是dom元素对象可以自己命名

```
domEle.css("color");//报错，dom对象无css方法
```

```
$(domEle).css("color","blue");
```

```
})
```

html:

```
<div style="color:blue;">1</div>
```

```
<div style="color:blue;">2</div>
```

```
<div style="color:blue;">3</div>
```

jQuery://三个div不同颜色

```
var arr = ["red","green","blue"];
```

```
var sum = 0;
```

```
$("div").each(function(i,domEle){
```

```
$(domEle).css("color",arr[i]);
```

```
sum += $(domEle).text();
```

```
})
```

```
console.log(sum);//0123字符型
```

```
$("div").each(function(i,domEle){
```

```
$(domEle).css("color",arr[i]);
```

```
sum += parseInt($(domEle).text());
```

```
})
```

```
console.log(sum);//6(变成数字型再相加)
```

(2)\$.each()方法遍历元素，主要用于遍历数据，处理数据

```
$.each($("div"),function(i,ele){
```

```

console.log(i);
console.log(ele);
})
$.each(arr,function(i,ele){
console.log(i);
console.log(ele);
})
//0
red
1
green
2
blue
$.each({
name:"andy",
age:18
},function(i,ele){
console.log(i);//输出属性名
console.log(ele);//输出属性值
})//
name
andy
age
18

```

## 七、创建添加和添加删除元素

### (1)创建元素

```

var li = $("
我是后来创建的li
");

```

### (2)添加元素

(i)内部添加，并且放到内容最后面

```

$("ul").append(li);

```

(ii)内部添加，并且放到内部最前面

```

$("ul").prepend(li);

```

(iii)外部添加，并且放在兄弟后面

```

$(".test").after(div);

```

(iv)外部添加，并且放在兄弟前面

```

$(".test").before(div);

```

### (3)删除元素

```

$("ul").remove();//删除匹配的元素本身

```

```

$("ul").empty();//删除匹配的元素集合中所有的子节点

```

```

$("ul").html("");//删除匹配的元素集合中所有的子节点

```



## 八、工厂函数

创建一个新的div元素，并通过对象字面量的方式设置属性

```
$("#<element>",{属性: 值})
```

```
var p = $("#<p>",{  
  text:"This is a paragraph."  
  style:"color:blue;font-size:14px;"  
})
```