



WEB前端

科技官网头部知识点介绍





jQuery遍历



jQuery制作下拉列表



move.js的使用



animate.css的使用

该颜色字体为演示案例

01

jQuery遍历



jQuery遍历



jQuery 遍历：用于根据其相对于其他元素的关系来“查找”（或选取）HTML 元素。以某项选择开始，并沿着这个选择移动，直到抵达期望的元素为止。

遍历例子见外附遍历文件夹



向上遍历方法(查找元素的祖先)：

遍历方法	描述	例子
parent()	返回被选元素的直接父元素 (只会向下一级对 DOM 树进行遍历)	<code>\$("span").parent();</code>
parents()	返回被选元素的所有祖先元素，它一路向上直到文档的根元素 (<html>)	<code>\$("span").parents();</code> <code>\$("span").parents("ul")</code> 返回所有 元素的所有祖先，并且它是 元素
parentsUntil()	返回介于两个给定元素之间的所有祖先元素	<code>\$("span").parentsUntil("div");</code>



jQuery遍历

向下遍历方法（查找元素后代）：

遍历方法	描述	例子
children()	返回被选元素的所有直接子元素(只会向下一级对 DOM 树进行遍历)	<code>\$("div").children()</code> <code>\$("div").children("p.1")</code> 返回类名为 "1" 的所有 <code><p></code> 元素，并且它们是 <code><div></code> 的直接子元素
find()	返回被选元素的后代元素，一路向下直到最后一个后代	<code>\$("div").find("span");</code> <code>\$("div").find("*");</code> 返回 <code><div></code> 的所有后代



jQuery遍历

水平遍历方法(元素同胞元素)：

遍历方法	描述	例子
siblings()	返回被选元素的所有同胞元素	<code>\$("h2").siblings();</code> <code>\$("h2").siblings("p");</code> 返回属于 <code><h2></code> 的同胞元素的所有 <code><p></code> 元素
next() prev()	返回被选元素的下一个/上一个同胞元素	<code>\$("h2").next();</code>
nextAll() prevAll()	返回被选元素的所有跟随的同胞（前面同胞）元素	<code>\$("h2").nextAll();</code>
nextUntil() prevUntil()	返回介于两个给定参数之间的所有跟随的同胞（前面同胞）元素	<code>\$("h2").nextUntil("h6");</code>



jQuery遍历

过滤：

遍历方法	描述	例子
first()	返回被选元素的首个元素	<code>\$("div p").first();</code>
last()	返回被选元素的最后一个元素	<code>\$("div p").last();</code>
eq()	返回被选元素中带有指定索引号的元素	<code>\$("p").eq(1);</code>
filter()	不匹配这个标准的元素会被从集合中删除，匹配的元素会被返回	<code>\$("p").filter(".intro");</code>
not()	not() 方法返回不匹配标准的所有元素 not() 方法与 filter() 相反。	<code>\$("p").not(".intro");</code>



jQuery遍历

each遍历：

each()方法规定为每个匹配元素规定运行的函数

语法：\$(selector).each(function(index,element))

- index - 选择器的 index 位置
- element - 当前的元素 (也可使用 "this" 选择器)

例：

```
$("#button").click(function(){  
    $("#li").each(function(){  
        alert($(this).text())  
    });  
});
```

02

jQuery制作下拉列表



下拉列表

三级下拉列表（竖向）：

单击一级菜单时展开二级菜单，单击二级菜单时展开三级菜单，当单击另一个一级菜单时同级其它菜单将收起。例图

我是一级菜单

我是二级菜单

我是三级菜单

我是三级菜单

我是三级菜单

我是二级菜单

我是二级菜单

我是一级菜单

我是一级菜单

单机展开下拉列表



下拉列表

多级下拉列表（横向）：

鼠标经过一级菜单时展开二级菜单，经过二级菜单时展开三级菜单，依次类推

我的网站 ▶	我的网站 ▶	我的网站 ▶	我的网站 ▶
2级分类 ▶	3级分类 ▶	4级分类 ▶	5级分类
2级分类 ▶	3级分类	4级分类	5级分类
2级分类 ▶	3级分类	4级分类	5级分类
2级分类 ▶	3级分类	4级分类	5级分类

多级下拉列表导航菜单（横向）

03

move.js的使用



move.js

move.js 是个极小的 JavaScript 库，支持 CSS3 的动画效果，非常简单美观。

引用方法：<script src="move.js"></script>

move.js中有编写了多种效果

如图：

```
easeIn: function(x,t, b, c, d){ //加速曲线
    return c*(t/=d)*t + b;
},
easeOut: function(x,t, b, c, d){ //减速曲线
    return -c *(t/=d)*(t-2) + b;
},
easeBoth: function(x,t, b, c, d){ //加速减速曲线
```



move.js

应用效果：

`$(this).find("ul").stop().slideDown(600, 'elasticOut');`

我们只需在这更换效果名称即可：如 `easeIn`, `easeOut` 等。

```
easeIn: function(x,t, b, c, d){ //加速曲线
    return c*(t/=d)*t + b;
},
easeOut: function(x,t, b, c, d){ //减速曲线
    return -c *(t/=d)*(t-2) + b;
},
easeBoth: function(x,t, b, c, d){ //加速减速曲线
```

move.js

04

animate.css的使用



animate.css的使用

animate.css是由Dan Eden的Daniel Eden使用CSS3的animation制作的动画效果的CSS集合, 很酷, 很有趣, 而且很炫又能跨浏览器的动画效果样式集锦, 可直接使用到项目中

- 引入文件: `<link href="animate.css" type="text/css" />`
- HTML及使用:

```
<div class= "animated bounce" ></div>
```

刷新页面就可看到动画效果



animate.css的使用

例1：animate的使用

```
<div class="box1">  
  <h1 class="animated bounce">Animate.css</h1>  
</div>
```

animated 类似于全局变量，它定义了动画的持续时间，动画特效须配合此标记才可执行

动画特效名称，可换任意效果，如果动画是无限播放的，可以添加 class infinite



animate.css的使用

例1：animate的使用

```
<div class="box1">  
  <h1 class="animated bounce">Animate.css</h1>  
</div>
```

animated 类似于全局变量，它定义了动画的持续时间，动画特效须配合此标记才可执行

动画特效名称，可换任意效果，如果动画是无限播放的，可以添加 class infinite



animate.css

例2：可以通过JavaScript或者jQuery给元素添加这些class

```
<div class="box1">  
  <h1 class="animated bounce">Animate.css</h1>  
</div>
```

animated 类似于全局变量，它定义了动画的持续时间，动画特效须配合此标记才可执行

动画特效名称，可换任意效果，如果动画是无限播放的，可以添加 class infinite

animate的使用



练习

- 侧边导航收缩伸展效果
- 标签切换

谢 谢