

2-18AJAX实现无刷新分页

2-18AJAX实现无刷新分页	1
1. 课堂案例	2
课堂讲解	2
一、上节回顾	2
什么是ajax.....	2
如何实现ajax.....	2
jquery.from.js的使用.....	2
二、学习目标	2
实现无刷新分页	2
三、教学过程描述	2
1、on	2
2、attr	3
3、html.....	3
4、新闻无刷新分页	4
四、小结	5
2. 课堂练习	5
3. 课后练习	5
4. 资料扩展	5
live.....	5

1. 课堂案例

课堂讲解

一、上节回顾

什么是ajax

如何实现ajax

jquery.from.js的使用

二、学习目标

实现无刷新分页

三、教学过程描述

1、on

定义和用法

on() 方法在被选元素及子元素上添加一个或多个事件处理程序。

语法

`$(selector).on(event,childSelector,data,function,map)`

参数 描述

event: 必需。规定要从被选元素移除的一个或多个事件或命名空间。

由空格分隔多个事件值。必须是有效的事件。

childSelector

: 可选。规定只能添加到指定的子元素上的事件处理程序(且不是选择器本身, 比如已废弃的 `delegate()` 方法)。

data: 可选。规定传递到函数的额外数据。

function: 可选。规

定当事件发生时运行的函数。

map: 规定事件映射 `((event:function, event:function, ...))`, 包含要添加到元素的一个或多个事件, 以及当事件发生时运行的函数。

2、attr

定义和用法

attr() 方法设置或返回被选元素的属性和值。

当该方法用于返回属性值, 则返回第一个匹配元素的值。

当该方法用于设置属性值, 则为匹配元素设置一个或多个属性/值对

语法

返回属性的值:

`$(selector).attr(attribute)`

设置属性和值:

`$(selector).attr(attribute,value)`

使用函数设置属性和值:

`$(selector).attr(attribute,function(index,currentvalue))`

设置多个属性和值:

`$(selector).attr({attribute:value, attribute:value,...})`

参数 描述

attribute 规定属性的名称。

value 规定属性的值。

function(index,currentvalue) 规定要返回属性值到集合的函数

index - 接受集合中元素的 index 位置。

currentvalue - 接受被选元素的当前属性值。

3、html

定义和用法

html() 方法设置或返回被选元素的内容(innerHTML)。

当该方法用于返回内容时, 则返回第一个匹配元素的内容。

当该方法用于设置内容时, 则重写所有匹配元素的内容。

语法

返回内容:

`$(selector).html()`

设置内容:

`$(selector).html(content)`

使用函数设置内容:

`$(selector).html(function(index,currentcontent))`

参数 描述

content 必需。规定被选元素的新内容(可包含 HTML 标签)。

function(index,currentcontent) 可选。规定返回被选元素的新内容的函数。

index - 返回集合中元素的 index 位置。

currentcontent - 返回被选元素的当前 HTML 内容。

4、新闻无刷新分页

先在分页设置一个id

```
<div class="pagination" id="page">
    <?=$page?>
</div> <!-- End .pagination -->
```

在新闻列表设定一个id

```
<tbody id="news">
<?php foreach($result as $key=>$value): ?>
    <tr>
        <td><input type="checkbox"/></td>
        <td><?=$value['id']?></td>
        <td><?=$value['category_name']?></td>
        <td><?=$value['title']?></td>
        <td><?=$value['excerpt']?></td>
        <td><?=$value['time']?></td>
        <td>
            <!-- Icons -->
            <a href="index.php?m=admin&c=news&a=edit&id=<?=$value['id']?>" title="Edit">
                
            </a>
            <a href="index.php?m=admin&c=news&a=delete&id=<?=$value['id']?>" title="Delete">
                
            </a>
        </td>
    </tr>
</tbody>
<?php endforeach?>
```

在使用ajax分页

```

<script type="text/javascript">
$( "#page" ).on( 'click', 'a', function ( e ) {
    var href = $(this).attr( 'href' );
    $.ajax({
        type: 'get',
        url: href,
        dataType: 'json',
        success: function (data) {
            var list = data.result;
            $( '#page' ).html( data.page );
            var news = '';
            for (var i=0; i<list.length; i++){
                if(i%2==0){
                    news += '<tr class="alt-row">';
                }else {
                    news += '<tr>';
                }
                news += '<td><input type="checkbox" /></td>';
                news += '<td>'+list[i].id+'</td>';
                news += '<td>'+list[i].category_name+'</td>';
                news += '<td>'+list[i].title+'</td>';
                news += '<td>'+list[i].excerpt+'</td>';
                news += '<td>'+list[i].time+'</td>';
                news += '<td>';
                news += '<a href="index.php?m=admin&c=news&a=edit&id='+list[i].id+'" title="Edit">';
                news += '';
                news += '</a>';
                news += '<a href="index.php?m=admin&c=news&a=delete&id='+list[i].id+'" title="Delete">';
                news += '';
                news += '</a>';
                news += '</td>';
                news += '</tr>';
            }
            $( '#news' ).html( news );
        }
    })
    return false; //停止往下执行
}
}
</script>

```

四、小结

- 1、jq的字符串拼接
- 2、html的重新赋值

2. 课堂练习

1、完成新闻无刷新分页

3. 课后练习

完成项目中所有分页的无刷新

4. 资料扩展

live

live() 方法在 jQuery 版本 1.7 中被废弃，在版本 1.9 中被移除。请使用 on() 方法代替。

定义和用法

`live()` 方法为被选元素附加一个或多个事件处理程序, 并规定当这些事件发生时运行的函数。

通过 `live()` 方法附加的事件处理程序适用于匹配选择器的当前及未来的元素(比如由脚本创建的新元素)。

语法

```
$(selector).live(event,data,function)
```

参数描述

event

必需。规定附加到元素的一个或多个事件。

由空格分隔多个事件。必须是有效的事件。

data 可选。规定传递到该函数的额外数据。

function必需。规定当事件发生时运行的函数。