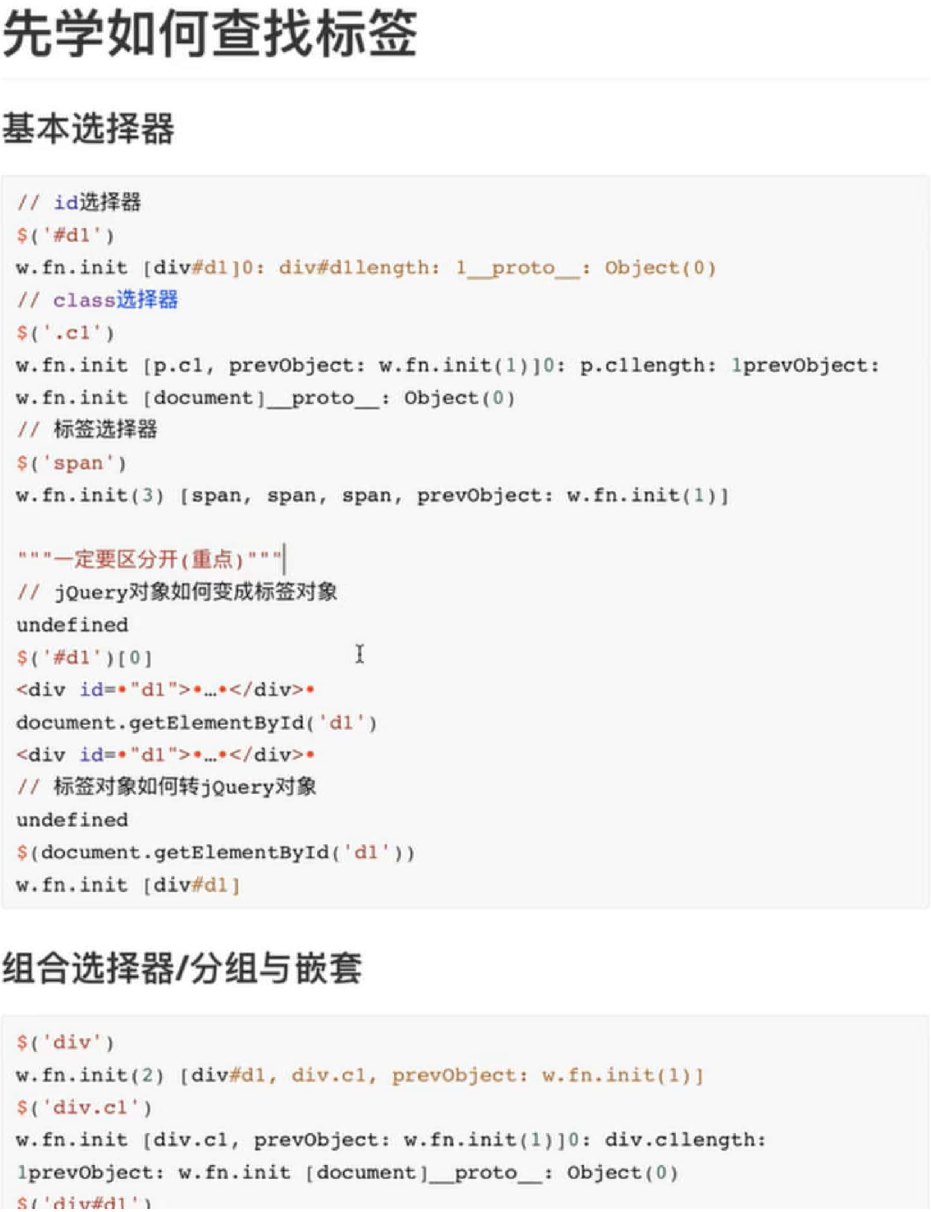
JQuery基础

2020年9月19日 16:56



先学如何查找标签

基本选择器

```
// id选择器
$('#d1')
w.fn.init [div#d1]0: div#d1length: 1__proto__: Object(0)
// class选择器
$('.c1')
w.fn.init [p.c1, prevObject: w.fn.init(1)]0: p.c1length: 1prevObject:
w.fn.init [document]__proto__: Object(0)
// 标签选择器
$('span')
w.fn.init(3) [span, span, span, prevObject: w.fn.init(1)]

"""一定要区分开(重点)"""
// jQuery对象如何变成标签对象
undefined
$('#d1')[0]
<div id=•"d1">•…•</div>•
document.getElementById('d1')
<div id=•"d1">•…•</div>•
// 标签对象如何转jQuery对象
undefined
$(document.getElementById('d1'))
w.fn.init [div#d1]
```

组合选择器/分组与嵌套

```
$('div')
w.fn.init(2) [div#d1, div.c1, prevObject: w.fn.init(1)]
$('div.c1')
w.fn.init [div.c1, prevObject: w.fn.init(1)]0: div.c1length:
1prevObject: w.fn.init [document]__proto__: Object(0)
$('div#d1')
```

组合选择器/分组与嵌套

```
$('div')
w.fn.init(2) [div#d1, div.c1, prevObject: w.fn.init(1)]
$('div.c1')
w.fn.init [div.c1, prevObject: w.fn.init(1)]0: div.c1length:
1prevObject: w.fn.init [document]__proto__: Object(0)
$('div#d1')
w.fn.init [div#d1, prevObject: w.fn.init(1)]
$('*')
w.fn.init(19) [html, head, meta, title, meta, link, script, script,
body, span, span, div#d1, span, p#d2, span, span, div.c1, span, span,
prevObject: w.fn.init(1)]

$('#d1,.c1,p')  # 并列+混用
w.fn.init(3) [div#d1, p#d2, div.c1, prevObject: w.fn.init(1)]

$('div span')  # 后代
w.fn.init(3) [span, span, span, prevObject: w.fn.init(1)]
$('div>span')  # 儿子
w.fn.init(2) [span, span, prevObject: w.fn.init(1)]
$('div+span')  # 毗邻
w.fn.init [span, prevObject: w.fn.init(1)]
$('div~span')  # 弟弟
w.fn.init(2) [span, span, prevObject: w.fn.init(1)]
```

基本筛选器

```
$('ul li')
w.fn.init(10) [li, li, li, li, li, li, li.c1, li, li#d1, li, prevObject:
w.fn.init(1)]

$('ul li:first')  # 大儿子
w.fn.init [li, prevObject: w.fn.init(1)]0: lilength: 1prevObject:
w.fn.init [document]__proto__: Object(0)

$('ul li:last')  # 小儿子
w.fn.init [li, prevObject: w.fn.init(1)]0: lilength: 1prevObject:
w.fn.init [document]__proto__: Object(0)
```

```
$('ul li:eq(2)')    # 放索引
w.fn.init [li, prevObject: w.fn.init(1)]0: lilength: 1prevObject:
w.fn.init [document]__proto__: Object(0)

$('ul li:even')  # 偶数索引 0包含在内
w.fn.init(5) [li, li, li, li.c1, li#d1, prevObject: w.fn.init(1)]0: li1:
li2: li3: li.c14: li#d1length: 5prevObject: w.fn.init
[document]__proto__: Object(0)

$('ul li:odd')  # 奇数索引
w.fn.init(5) [li, li, li, li, li, prevObject: w.fn.init(1)]0: li1: li2:
li3: li4: lilength: 5prevObject: w.fn.init [document]__proto__:
Object(0)

$('ul li:gt(2)')  # 大于索引
w.fn.init(7) [li, li, li, li.c1, li, li#d1, li, prevObject:
w.fn.init(1)]0: li1: li2: li3: li.c14: li5: li#d16: lilength:
7prevObject: w.fn.init [document]__proto__: Object(0)

$('ul li:lt(2)')  # 小于索引
w.fn.init(2) [li, li, prevObject: w.fn.init(1)]0: li1: lilength:
2prevObject: w.fn.init [document]__proto__: Object(0)
```

```
$('ul li:not("#d1")')  # 移除满足条件的标签
w.fn.init(9) [li, li, li, li, li, li, li.c1, li, li, prevObject:
w.fn.init(1)]

$('div')
w.fn.init(2) [div, div, prevObject: w.fn.init(1)]
$('div:has("p")')  # 选取出包含一个或多个标签在内的标签
w.fn.init [div, prevObject: w.fn.init(1)]
```

属性选择器

```
$('[username]')
w.fn.init(3) [input, input, p, prevObject: w.fn.init(1)]
$('[username="jason"]')
w.fn.init [input, prevObject: w.fn.init(1)]
$('p[username="egon"]')
w.fn.init [p, prevObject: w.fn.init(1)]
```

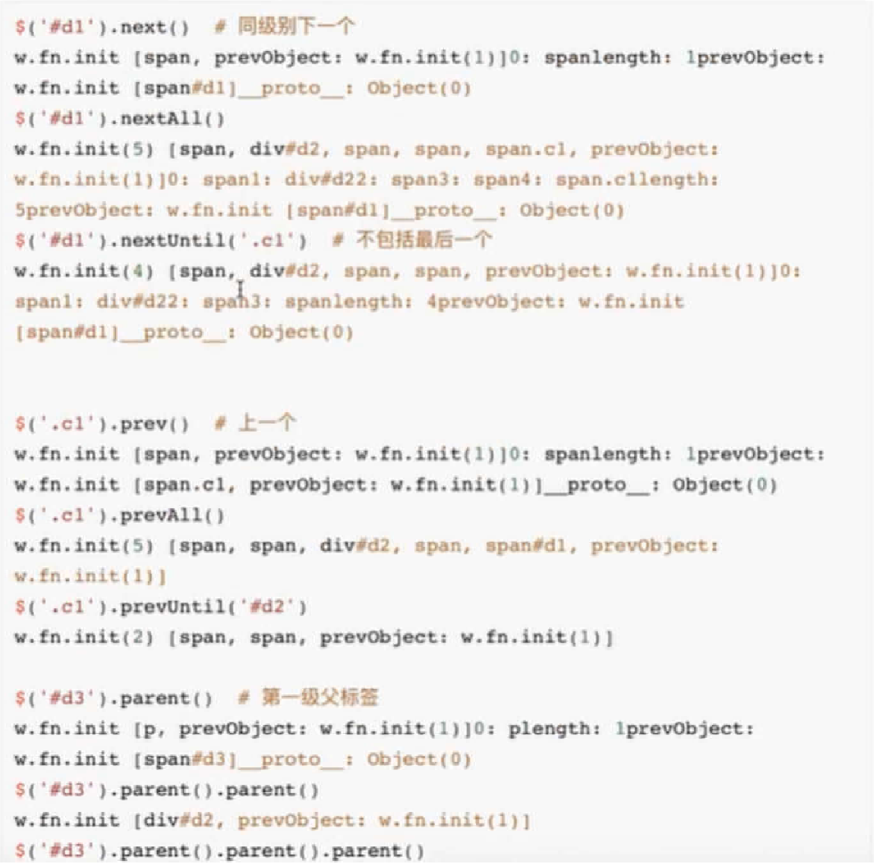
表单筛选器

```
$('input[type="text"]')
w.fn.init [input, prevObject: w.fn.init(1)]0: inputlength: 1prevObject:
w.fn.init [document]__proto__: Object(0)
$('input[type="password"]')
w.fn.init [input, prevObject: w.fn.init(1)]

$(':text')  # 等价于上面第一个
w.fn.init [input, prevObject: w.fn.init(1)]0: inputlength: 1prevObject:
w.fn.init [document]__proto__: Object(0)
$(':password')  # 等价于上面第二个
w.fn.init [input, prevObject: w.fn.init(1)]



:text
:password
:file
:radio
:checkbox

:submit
:reset
:button
```

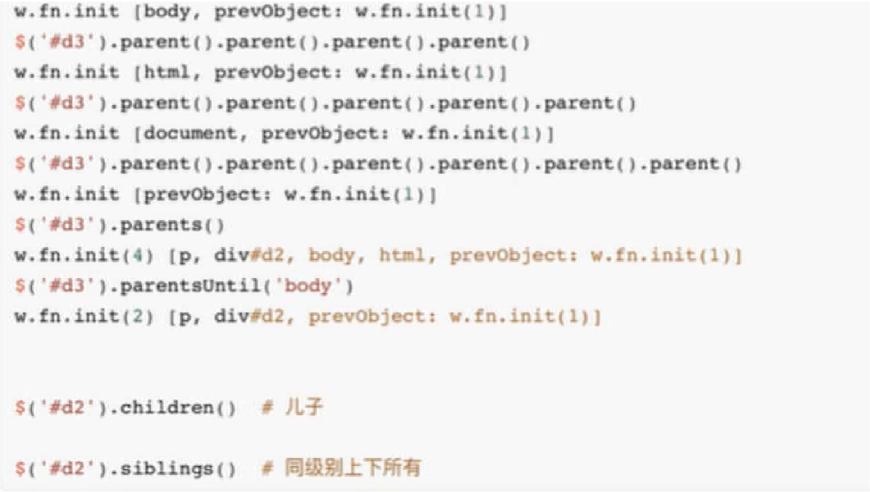
```
表单对象属性
:enabled
:disabled
:checked
:selected
```

```
$(':checked')  # 它会将checked和selected都拿到
w.fn.init(2) [input, option, prevObject: w.fn.init(1)]0: input1:
optionlength: 2prevObject: w.fn.init [document]__proto__: Object(0)
$(':selected')  # 它不会 只拿selected
w.fn.init [option, prevObject: w.fn.init(1)]
$('input:checked')  # 自己加一个限制条件
w.fn.init [input, prevObject: w.fn.init(1)]
```

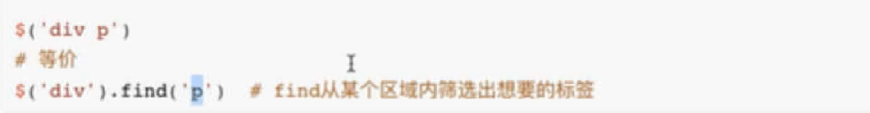
筛选器方法



```
$('#d1').next()  # 同级别下一个
w.fn.init [span, prevObject: w.fn.init(1)]0: spanlength: 1prevObject:
w.fn.init [span#d1]__proto__: Object(0)
$('#d1').nextAll()
w.fn.init(5) [span, div#d2, span, span, span.c1, prevObject:
w.fn.init(1)]0: span1: div#d22: span3: span4: span.c1length:
5prevObject: w.fn.init [span#d1]__proto__: Object(0)
$('#d1').nextUntil('.c1')  # 不包括最后一个
w.fn.init(4) [span, div#d2, span, span, prevObject: w.fn.init(1)]0:
span1: div#d22: span3: spanlength: 4prevObject: w.fn.init
[span#d1]__proto__: Object(0)



$('.c1').prev()  # 上一个
w.fn.init [span, prevObject: w.fn.init(1)]0: spanlength: 1prevObject:
w.fn.init [span.c1, prevObject: w.fn.init(1)]__proto__: Object(0)
$('.c1').prevAll()
w.fn.init(5) [span, span, div#d2, span, span#d1, prevObject:
w.fn.init(1)]
$('.c1').prevUntil('#d2')
w.fn.init(2) [span, span, prevObject: w.fn.init(1)]

$('#d3').parent()  # 第一级父标签
w.fn.init [p, prevObject: w.fn.init(1)]0: plength: 1prevObject:
w.fn.init [span#d3]__proto__: Object(0)
$('#d3').parent().parent()
w.fn.init [div#d2, prevObject: w.fn.init(1)]
$('#d3').parent().parent().parent()
```

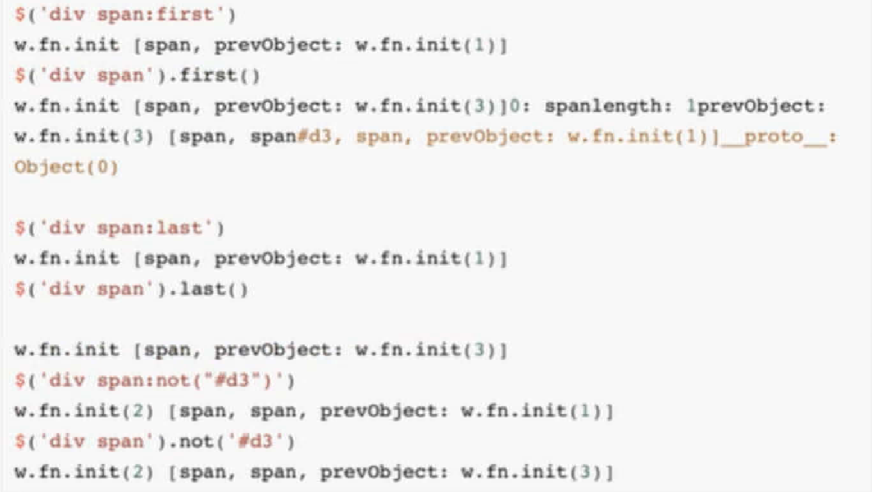


```
w.fn.init [body, prevObject: w.fn.init(1)]
$('#d3').parent().parent().parent().parent()
w.fn.init [html, prevObject: w.fn.init(1)]
$('#d3').parent().parent().parent().parent().parent()
w.fn.init [document, prevObject: w.fn.init(1)]
$('#d3').parent().parent().parent().parent().parent().parent()
w.fn.init [prevObject: w.fn.init(1)]
$('#d3').parents()
w.fn.init(4) [p, div#d2, body, html, prevObject: w.fn.init(1)]
$('#d3').parentsUntil('body')
w.fn.init(2) [p, div#d2, prevObject: w.fn.init(1)]


$('#d2').children()  # 儿子

$('#d2').siblings()  # 同级别上下所有
```



```
$('div p')
# 等价
$('div').find('p')  # find从某个区域内筛选出想要的标签
```

下述两两等价



```
$('div span:first')
w.fn.init [span, prevObject: w.fn.init(1)]
$('div span').first()
w.fn.init [span, prevObject: w.fn.init(3)]0: spanlength: 1prevObject:
w.fn.init(3) [span, span#d3, span, prevObject: w.fn.init(1)]__proto__:
Object(0)

$('div span:last')
w.fn.init [span, prevObject: w.fn.init(1)]
$('div span').last()

w.fn.init [span, prevObject: w.fn.init(3)]
$('div span:not("#d3")')
w.fn.init(2) [span, span, prevObject: w.fn.init(1)]
$('div span').not('#d3')
w.fn.init(2) [span, span, prevObject: w.fn.init(3)]
```