

■ 필터링 메소드

- 선택자에 의해 선택된 요소들의 일부분을 추출
- 이미 선택된 요소 집합에서 새로운 조건을 적용하여 새로운 집합 생성
- 기본 형태
 - `$(selector).filter(selector)`
 - `$(selector).filter(function(index) {})`
 - `$(selector).filter(element)`
 - `$(selector).filter(jquery object)`

■ 필터링 메소드

● h3 중 짝수 요소만 선택

```
<h3>Header-0</h3>
<h3>Header-1</h3>
<h3>Header-2</h3>
<h3>Header-3</h3>
<h3>Header-4</h3>
<h3>Header-5</h3>
<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
    $('h3:even').css({'color' : 'white', 'background-color' : 'black'});
</script>
```

```
<h3>Header-0</h3>
<h3>Header-1</h3>
<h3>Header-2</h3>
<h3>Header-3</h3>
<h3>Header-4</h3>
<h3>Header-5</h3>
<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
    $('h3').filter(':even').css({'color' : 'white', 'background-color' : 'black'});
</script>
```

Header-0

Header-1

Header-2

Header-3

Header-4

Header-5

■ 필터링 메소드

- title 클래스를 가지는 요소와 p 태그를 자식으로 가지는 요소 선택

```
<h3>Header-0</h3>
<h3 class="title">Header-1</h3>
<h3>Header-2</h3>
<h3>Header-3</h3>
<h3><p>Header-4</p></h3>
<h3>Header-5</h3>

<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
    $("h3").filter(".title").css("background", "gray");
    $("h3").filter(":has(p)").css("background", "blue");
</script>
```

Header-0

Header-1

Header-2

Header-3

Header-4

Header-5

■ 필터링 메소드

● h3 중 index가 3의 배수인 요소 선택

```
<h3>Header-0</h3>
<h3>Header-1</h3>
<h3>Header-2</h3>
<h3>Header-3</h3>
<h3>Header-4</h3>
<h3>Header-5</h3>

<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
    $('h3').filter(function(idx) {
        return idx % 3 == 0;
    }).css({'color' : 'white', 'background-color' : 'black'});
</script>
```

Header-0

Header-1

Header-2

Header-3

Header-4

Header-5

■ 필터링 메소드

● li 중 자식 span 요소를 하나만 가지는 요소 선택

```
<ul>
  <li><span>item 1</span> <span>item 1-1</span></li>
  <li><span>item 2</span></li>
  <li><span>item 3</span></li>
  <li><span>item 4</span> <span>item 4-1</span></li>
  <li><span>item 5</span></li>
</ul>
```

- item 1 item 1-1
- item 2
- item 3
- item 4 item 4-1
- item 5

```
<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
  $('li').filter(function(idx) {
    var len = $('span', this).length;
    return len == 1;
  }).css('background-color', 'yellow');
</script>
```

■ 필터링 메소드

메소드	설명
eq()	특정 위치에 존재하는 요소 선택
first()	첫번째에 위치하는 요소 선택
last()	마지막에 위치하는 요소 선택
prev()	이전 형제 요소 선택
next()	다음 형제 요소 선택
children()	자식 요소 선택
find()	자식 요소 중 선택자에 의한 요소 선택
parent()	부모 요소 선택
siblings()	형제 요소 중 선택자에 의한 요소 선택
is()	요소 특징을 판별하여 요소 선택

■ 필터링 메소드

● eq(), first(), last()

```
<h3>Header-0</h3>
```

```
<h3>Header-1</h3>
```

```
<h3>Header-2</h3>
```

```
<h3>Header-3</h3>
```

```
<h3>Header-4</h3>
```

```
<h3>Header-5</h3>
```

```
<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
```

```
<script>
```

```
    $("h3").first().css("background", "red");
```

```
    $("h3").eq(2).css("background", "green");
```

```
    $("h3").last().css("background", "blue");
```

```
</script>
```

Header-0

Header-1

Header-2

Header-3

Header-4

Header-5

■ 필터링 메소드

● prev(), next()

```
<h3>Header-0</h3>
<h3>Header-1</h3>
<h3 id='select'>Header-2</h3>
<h3>Header-3</h3>
<h3>Header-4</h3>
<h3>Header-5</h3>

<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
    $("#select").prev().css("background", "red");
    $("#select").next().css("background", "green");
</script>
```

Header-0

Header-1

Header-2

Header-3

Header-4

Header-5

■ 필터링 메소드

● children()

```
<div>
  <p>1번</p>
  <p>2번</p>
  <p>3번</p>
  <h3>4번</h3>
  <h3>5번</h3>
  <h3>6번</h3>
</div>

<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
  $('div').children().css('background', 'cyan');
  $('div').children('p').css('background', 'yellow');
</script>
```

1번

2번

3번

4번

5번

6번

■ 필터링 메소드

● find()

```
<ul>
  <li><p>1 번</p></li>
  <li><span>2 번</span></li>
  <li><ol><li>3 번</li></ol></li>
</ul>

<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
  $('ul').find('span').css('color', 'red');
</script>
```

- 1 번
- 2 번
- 1. 3 번

■ 필터링 메소드

● parent(), parents()

```
<ul>
  <li><p>1 번</p></li>
  <li><span>2 번</span></li>
  <li><ol><li>3 번</li></ol></li>
</ul>
```

- 1 번
- 2 번
- 1. 3 번

```
<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
  $('span').parent().css('background', 'cyan');
  $('li li').parent().parent().css('background', 'orange');
  $('li li').parents('ul').css('border', '1px dashed blue');
</script>
```

■ 필터링 메소드

● siblings()

```
<h1>Header-0</h1>
<h2>Header-1</h2>
<h3>Header-2</h3>
<h4>Header-3</h4>
<h5>Header-4</h5>
<h6>Header-5</h6>
```

```
<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>
<script>
    $('h4').siblings('h1').css('border', '1px dashed blue');
    $('h4').siblings('h6').css('border', '1px dashed blue');
</script>
```

Header-0

Header-1

Header-2

Header-3

Header-4

Header-5

■ 필터링 메소드

● is() - 요소 특징 판별

- 매개변수로 선택자 입력
- 선택한 객체가 선택자와 일치하는지 판별해 true / false 반환

```
<h3 class="select">Header-0</h3>  
<h3>Header-1</h3>  
<h3>Header-2</h3>  
<h3>Header-3</h3>  
<h3 class="select">Header-4</h3>  
<h3>Header-5</h3>
```

```
<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>  
<script>  
  var elements = $('h3');  
  for(var i = 0; i < elements.length; i++) {  
    var element = elements[i];  
    var isSelect = $(element).is('.select');  
    if(isSelect) {  
      $(element).css('background', 'yellow');  
    }  
  }  
</script>
```

Header-0

Header-1

Header-2

Header-3

Header-4

Header-5

■ 필터링 메소드

● is() - 요소 특징 판별

- 매개변수로 선택자 입력
- 선택한 객체가 선택자와 일치하는지 판별해 true / false 반환

```
<h3 class="select">Header-0</h3>  
<h3>Header-1</h3>  
<h3>Header-2</h3>  
<h3>Header-3</h3>  
<h3 class="select">Header-4</h3>  
<h3>Header-5</h3>
```

```
<script src="http://code.jquery.com/jquery-3.1.1.min.js"></script>  
<script>  
  var elements = $('h3');  
  for(var i = 0; i < elements.length; i++) {  
    var element = elements[i];  
    var isSelect = $(element).is('h3:eq(2)');  
    if(isSelect) {  
      $(element).css('background', 'yellow');  
    }  
  }  
</script>
```

Header-0

Header-1

Header-2

Header-3

Header-4

Header-5