**JQUERY**

**Mục lục**

[1. Khái niệm 3](#_Toc84422569)

[2. Syntax 3](#_Toc84422570)

[3. jQuery Selectors 3](#_Toc84422571)

[4. Xử lý HTML jQuery 4](#_Toc84422572)

[4.1 add() 4](#_Toc84422573)

[.add(bộ chọn) 4](#_Toc84422574)

[.add(html) 4](#_Toc84422575)

[.add(Object) 5](#_Toc84422576)

[.add(bộ chọn,context) 5](#_Toc84422577)

[4.2 .remove() 6](#_Toc84422578)

[. remove() – Loại bỏ tất cả các thành phần được chọn 6](#_Toc84422579)

[.remove(‘Bộ chọn’) – Loại bỏ các thành phần được xác định 6](#_Toc84422580)

[4.3 .addClass() 6](#_Toc84422581)

[.addClass(tênClass) 6](#_Toc84422582)

[.addClass(function(index){…}) 7](#_Toc84422583)

[4.4 .removeClass() 7](#_Toc84422584)

[.removeClass() 7](#_Toc84422585)

[.removeClass(‘Tên Class’) 8](#_Toc84422586)

[4.5 .after() 8](#_Toc84422587)

[.after(nội dung) 8](#_Toc84422588)

[.after(function(index){…}) 9](#_Toc84422589)

[4.6 .before() 9](#_Toc84422590)

[4.7 .insertAfter() 9](#_Toc84422591)

[.insertAfter(bộ chọn) 10](#_Toc84422592)

[4.8 .insertBefore() 10](#_Toc84422593)

[4.9 .append() 10](#_Toc84422594)

[.append(nội dung) 11](#_Toc84422595)

[.append(function(index){…}) 11](#_Toc84422596)

[4.10 .appendTo() 12](#_Toc84422597)

[.appendTo(Mục tiêu) 12](#_Toc84422598)

[4.11 .prepend() 12](#_Toc84422599)

[.prepend(nội dung) 12](#_Toc84422600)

[.prepend(function(index){…}) 13](#_Toc84422601)

[4.12 .prependTo() 14](#_Toc84422602)

[.appendTo(Mục tiêu) 14](#_Toc84422603)

[4.13 .attr() 14](#_Toc84422604)

[.attr(tên thuộc tính) 14](#_Toc84422605)

[.attr(tên thuộc tính, giá trị thuộc tính) 15](#_Toc84422606)

[4.14 .html() 15](#_Toc84422607)

[.html() 15](#_Toc84422608)

[.html(‘Nội dung HTML’) 15](#_Toc84422609)

[.html(function(){…}) 16](#_Toc84422610)

[4.15 .text() 16](#_Toc84422611)

[4.16 .val() 16](#_Toc84422612)

[4.17 .change() 16](#_Toc84422613)

[4.18 .clone() 16](#_Toc84422614)

[4.19 .height() 16](#_Toc84422615)

[4.20 .width() 16](#_Toc84422616)

[5. jQuery HTML 16](#_Toc84422617)

[6. jQuery Traversing 16](#_Toc84422618)

1. Khái niệm

* Thư viện của JavaScript
* Viết ngắn gọn hơn JS
* Thư viện jQuery có các tính năng :
  + Thao tác với HTML/DOM
  + Thao tác CSS
  + Hiệu ứng và hình ảnh động
  + AJAX
  + Tiện ích

1. Syntax

* Cách viết câu lệnh jQuery cơ bản tất cả các xử lý đều nằm trong khai báo ban đầu:

$(document).ready(function(){  
  
  *// jQuery methods go here...*  
  
});

* Bên trong lấy các selector để định nghĩa các sự kiện cho nó

**$(*selector*).*action*()**

**Ví dụ:**

$(document).ready(function(){  
  $("p").click(function(){  
    $(this).hide();  
  });  
});

1. jQuery Selectors

* Cho phép chọn và thao tác với các phần tử của HTML

Ví dụ:

* [$('.class')](https://hocwebchuan.com/tutorial/jquery/selector-class.php) Chọn thành phần theo class cụ thể.
* [$('#id')](https://hocwebchuan.com/tutorial/jquery/selector-id.php) Chọn thành phần theo id cụ thể.
* [$('selector1, selector2, selectorN')](https://hocwebchuan.com/tutorial/jquery/selector-multiple.php) Chọn nhiều thành phần cùng lúc.
* [$('tag:eq(n)')](https://hocwebchuan.com/tutorial/jquery/selector-eq.php) Chọn thành phần thứ n.
* [$('tag:gt(n)')](https://hocwebchuan.com/tutorial/jquery/selector-gt.php) chọn các thành phần với chỉ số lớn hơn n.
* [$('tag:lt(n)')](https://hocwebchuan.com/tutorial/jquery/selector-lt.php) Chọn các thành phần với chỉ số nhỏ hơn n.
* [$('tag:even')](https://hocwebchuan.com/tutorial/jquery/selector-even.php) Chọn các thành phần ở vị trí lẻ.
* [$('tag:odd')](https://hocwebchuan.com/tutorial/jquery/selector-odd.php) Chọn các thành phần ở vị trí chẵn.
* [$('tag:first-child')](https://hocwebchuan.com/tutorial/jquery/selector-first-child.php) Chọn các thành phần con ở vị trí đầu tiên.
* [$('tag:last-child')](https://hocwebchuan.com/tutorial/jquery/selector-last-child.php) Chọn các thành phần con ở vị trí cuối cùng.
* [$('tag:nth-child(n)')](https://hocwebchuan.com/tutorial/jquery/selector-nth-child.php) Chọn thành phần thứ n trong thành phần cha.
* [$('tag[attribute="value"]')](https://hocwebchuan.com/tutorial/jquery/selector-attribute.php) Chọn thành phần có thuộc tính với giá trị xác định.

<https://hocwebchuan.com/tutorial/jquery/>

1. Xử lý HTML jQuery

## .add()

* Thêm thành phần

### .add(bộ chọn)

<div>Thành phần div</div>

<p>Thành phần p</p>

<script>

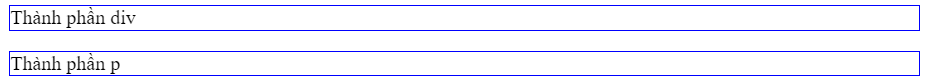
$(function(){

$('div').add('p').css('border','1px solid blue');

});

</script>

=>



### .add(html)

<div>Thành phần div</div>

<script>

$(function(){

$('div').add('<p class="test">tag p có class test</p>').appendTo('body');

});

</script>

=>

Text

Description automatically generated

### .add(Object)

<div>Thành phần divdiv</div>

<p>Thành phần p</p>

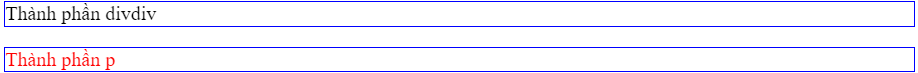
<script>

$(function(){ $('div').add($('p').css('color','red')).css('border','1px solid blue');

});

</script>

=>



### .add(bộ chọn,context)

<div>

<p>Thành phần p nằm trong div</p>

</div>

<p>Thành phần p</p>

<style>

.test {

background-color: blue;

}

</style>

<script>

$(function(){

$('div').click(function() {

$('div').add('p',this).addClass('test');

});

});

</script>

=>Tìm thẻ p trong div và add thêm class test đã CSS

=> Sự kiện click chuôt vào div sẽ có css của class test



## .remove()

Loại bỏ thành phần được xác định

### . remove() – Loại bỏ tất cả các thành phần được chọn

<div class="test">Thành phần div có class test</div>

<div>Thành phần div</div>

<script>

$(function(){

$('.test').remove();

});

</script>

=>

Shape

Description automatically generated with medium confidence

### .remove(‘Bộ chọn’) – Loại bỏ các thành phần được xác định

<div class="test">Thành phần div có class test</div>

<div>Thành phần div</div>

<script>

$(function(){

$('div').remove(".test");

});

</script>

=>

A picture containing text

Description automatically generated

* 1. .addClass()
* Thêm class cho thành phần

### .addClass(tênClass)

<div>Thành phần div</div>

<style>

.test {

background-color: yellow;

}

</style>

<script>

$(function(){

$('div').addClass('test');

});

</script>

* + Sau khi addClass test vào html có cấu trúc như sau
  + <div class="test">Thành phần div</div>



### .addClass(function(index){…})

<ul>

<li>Thành phần li 0</li>

<li>Thành phần li 1</li>

<li>Thành phần li 2</li>

<li>Thành phần li 3</li>

<li>Thành phần li 4</li>

</ul>

<script>

$(function(){

$('ul li').addClass(function(index){

return "list-" + index;

});

});

</script>

<style>

.list-2 {

background-color: yellow;

}

</style>

* + Sau khi add Class vào thẻ li trong ul
  + <li class="list-0">Thành phần li 0</li>…

A picture containing shape

Description automatically generated

* 1. .removeClass()
* Xoá class của thành phần

### .removeClass()

– Loại bỏ tất cả các thuộc tính có trong thành phần

<div id="remove" class="test01 test02 test03">Thành phần div</div>

<script>

$(function(){

$('div').removeClass();

});

</script>

* + <div id="remove" class="">Thành phần div</div>

A picture containing text

Description automatically generated

### .removeClass(‘Tên Class’)

– Loại bỏ tên class có trong thành phần

<div id="remove" class="test01 test02 test03">Thành phần div</div>

<script>

$(function(){

$('div').removeClass('test02');

});

</script>

* + <div id="remove" class="test01 test03">Thành phần div</div>



* 1. .after()
* Chèn nội dung vào ngay sau mỗi thành phần đã có

### .after(nội dung)

<div>Thành phần div</div>

<script>

$(function(){

$('div').after('<p>nội dung thêm vào</p>');

});

</script>

* + <div>Thành phần div</div>  
    <p>nội dung thêm vào</p>

Text

Description automatically generated with medium confidence

VD2:

<div>

<h3>thành phần h3</h3>

<p>thành phần p</p>

</div>

<script>

$(function(){

$('div').after($('h3'));

});

</script>

* + <div>  
     <p>thành phần p</p>  
    </div>  
    <h3>thành phần h3</h3>

Text

Description automatically generated with medium confidence

### .after(function(index){…})

<div class="test01">thành phần div 01</div>

<div class="test02">thành phần div 02</div>

<script>

$(function(){

$('div').after(function() {

return '<p>' + this.className + '</p>';

});

});

</script>

* + <div class="test01">thành phần div 01</div>  
    <p>test01</p>  
    <div class="test02">thành phần div 02</div>  
    <p>test02</p>

Graphical user interface, text, application

Description automatically generated

* 1. [.before()](https://hocwebchuan.com/tutorial/jquery/before.php)
* Chèn nội dung vào ngay trước mỗi thành phần đã có

Ngược lại với after

* 1. [.insertAfter()](https://hocwebchuan.com/tutorial/jquery/insertafter.php)
* Chèn hoặc di chuyển một thành phần vào ngay sau mục tiêu được chọn.

### .insertAfter(bộ chọn)

<div class="test">Thành phần div</div>

<div class="test">Thành phần div</div>

<script>

$(function(){

$('<p>Thành phần p được chèn</p>').insertAfter('div.test');

});

</script>

* + <div class="test">Thành phần div</div>  
    <p>Thành phần p được chèn</p>  
    <div class="test">Thành phần div</div>  
    <p>Thành phần p được chèn</p>

Text

Description automatically generated

VD2

<h2>Thành phần h2</h2>

<p>Thành phần p</p>

<script>

$(function(){

$('h2').insertAfter('div');

});

</script>

* + <div>  
     <p>Thành phần p</p>  
    </div>  
    <h2>Thành phần h2</h2>

Graphical user interface, text

Description automatically generated

* 1. .insertBefore()
* Chèn hoặc di chuyển một thành phần vào ngay trước mục tiêu được chọn.

Ngược lại với .insertAfter

* 1. .append()
* Chèn nội dung vào trong thành phần đã có, sắp xếp ở vị trí ban đầu

### .append(nội dung)

<div>Thành phần div</div>

<script>

$(function(){

$('div').append('<p>nội dung thêm vào</p>');

});

</script>

* + <div>Thành phần div  
     <p>nội dung thêm vào</p>  
    </div>

Text

Description automatically generated with low confidence

VD2:

<div>

<h3>thành phần h3</h3>

<p>thành phần p</p>

</div>

<script>

$(function(){

$('div').append($('h3'));

});

</script>

* + <div>  
     <p>thành phần p</p>  
     <h3>thành phần h3</h3>  
    </div>

Text

Description automatically generated with medium confidence

### .append(function(index){…})

<div class="test01">thành phần div 01</div>

<div class="test02">thành phần div 02</div>

<script>

$(function(){

$('div').append(function() {

return '<p>' + this.className + '</p>';

});

});

</script>

* + <div class="test01">thành phần div 01  
     <p>test01</p>  
    </div>  
    <div class="test02">thành phần div 02  
     <p>test02</p>  
    </div>

Graphical user interface, application

Description automatically generated

* 1. .appendTo()
* Chèn nội dung, di chuyển thành phần vào trong thành phần khác, nội dung này thường sắp xếp ở vị trí sau cùng
* Ứng dụng vào tạo nội dung tooltip, sắp xếp thành phần,…

### .appendTo(Mục tiêu)

<div class="test">Thành phần div</div>

<script>

$(function(){

$('<p>Thành phần p</p>').appendTo('div.test');

});

</script>

* + <div class="test">Thành phần div  
     <p>Thành phần p</p>  
    </div>

Text

Description automatically generated

* 1. .prepend()
* Chèn nội dung vào trong thành phần ở vị trí đầu tiên

### .prepend(nội dung)

<div>Thành phần div</div>

<div>Thành phần div</div>

<script>

$(function(){

$('div').prepend('<p>nội dung thêm vào</p>');

});

</script>

* + <div>

<p>nội dung thêm vào</p>  
 Thành phần div  
</div>  
<div>

<p>nội dung thêm vào</p>  
 Thành phần div  
 </div>

Text

Description automatically generated with low confidence

VD2:

<div>

<p>thành phần p</p>

<h3>thành phần h3</h3>

</div>

<script>

$(function(){

$('div').prepend($('h3'));

});

</script>

* + <div>  
     <h3>thành phần h3</h3>  
     <p>thành phần p</p>  
    </div>

Text

Description automatically generated with medium confidence

### .prepend(function(index){…})

<div class="test01">thành phần div 01</div>

<div class="test02">thành phần div 02</div>

<script>

$(function(){

$('div').prepend(function() {

return '<p>' + this.className + '</p>';

});

});

</script>

* + <div class="test01">  
     <p>test01</p>  
     thành phần div 01

</div>  
<div class="test02">  
 <p>test02</p>  
 thành phần div 02

</div>

Graphical user interface, text, application

Description automatically generated

* 1. .prependTo()
* Chèn nội dung vào thành phần đã có, thường được sắp xếp ở vị trí đầu tiên

### .appendTo(Mục tiêu)

<div class="test">Thành phần div</div>

<script>

$(function(){

$('<p>Thành phần p</p>').prependTo('div.test');

});

</script>

* + <div class="test">

<p>Thành phần p</p>  
 Thành phần div  
</div>

Text

Description automatically generated

* 1. .attr()
* Lấy giá trị hoặc thêm thuộc tính cho thành phần
* .attr() để lấy giá trị như id, class, chúng ta có thể lấy title, type, hay bất cứ attr nào khác

### .attr(tên thuộc tính)

<p>Thành phần p có <a href="http://www.hocwebchuan.com">link a</a></p>

<div>Bạn đang xem nội dung tại <span class="test"></span></div>

<script>

$(function(){

var hrefAttr = $('a').attr('href');

$('.test').html(hrefAttr);

});

</script>

Text

Description automatically generated with low confidence

### .attr(tên thuộc tính, giá trị thuộc tính)

<p>Thành phần p có <a href="http://www.google.com">link a</a></p>

<script>

$(function(){

$('a').attr('href','http://www.hocwebchuan.com');

});

</script>

* + <p>Thành phần p có <a href="http://www.hocwebchuan.com">link a</a></p>

Text

Description automatically generated with medium confidence

* 1. .html()
* Lấy nội dung hoặc gán giá trị HTML cho thành phần
* Sử dụng trong popup, lightbox, thickbox,…

### .html()

<div>

<p>Thành phần <span>span</span> bên trong thành phần cha</p>

</div>

<button>Click</button>

<script>

$(function(){

var divHtml = $('div').html();

$('button').click(function(){

alert(divHtml);

});

});

</script>

=>

Text

Description automatically generated with medium confidenceGraphical user interface, text, application

Description automatically generated

### .html(‘Nội dung HTML’)

<div></div>

<script>

$(function(){

$('div').html('<p>Thành phần được thêm</p>');

});

</script>

Logo

Description automatically generated with medium confidence

### .html(function(){…})

<p>Thành phần p 01</p>

<p>Thành phần p 02</p>

<div></div>

<script>

$(function(){

$('div').html(function(){

var txt = '<em>' + $('p').length + ' đoạn văn bản.</em>';

return '<p>Tất cả nội dung gồm ' + txt + '<p>';

});

});

</script>

Text, application

Description automatically generated with medium confidence

* 1. .text()
* Lấy hoặc thay đổi hoặc gán nội dung text cho thành phần

### .text()

<p>Thành phần <span>span</span> bên trong thành phần cha</p>

<button>Click</button>

<script>

$(function(){

var txt = $('p').text();

$('button').click(function(){

alert(txt);

});

});

</script>

Text

Description automatically generated with medium confidenceGraphical user interface, text, application

Description automatically generated

### .text(‘Nội dung HTML’)

<div></div>

<script>

$(function(){

$('div').text('Đoạn text được thêm vào');

});

</script>



* 1. .val()
* .val() lấy giá trị hiện tại của thành phần, hoặc thay đổi giá trị cho thành phần
* .val() lấy giá trị đầu tiên nếu thành phần chọn là một danh sách
* Thông thường giá trị là các thành phần của form

### .val()

<input type="text" value="12">

<button>Click</button>

<script>

$(function(){

var inputVal = $('input').val();

$('button').click(function(){

alert(inputVal);

});

});

</script>

Graphical user interface, text, application

Description automatically generated

.val(‘Giá trị’)

<input type="text" value="">

<button>CSS</button>

<button>HTML</button>

<button>JQUERY</button>

<script>

$(function(){

$("button").click(function () {

var text = $(this).text();

$("input").val(text);

});

});

</script>

* + Khi click vào button sẽ lấy text trong button ra để vào input

Text

Description automatically generated

* 1. .change()
* .change() xác định một thành phần đã được thay đổi
* .change() thường được sử dụng để xử lý các thành phần của form như input, textarea, button

### .change(function(){})

<input class="textField" type="text" value="Text"><br>

<span></span>

<script>

$(function(){

$('input').change(function() {

$('span').text('Trường nhập đã được thay đổi');

});

});

</script>

A picture containing rectangle

Description automatically generated

* + Khi thay đổi nội dung trong input, đưa chuột ra ngoài và click sẽ thêm text vào span
  + <span>Trường nhập đã được thay đổi</span>

A picture containing text

Description automatically generated

* 1. .clone()
* .clone() tạo một bản sao (copy) cho thành phần
* .clone() thường dùng kết hợp với .after() hoặc .before() để tạo trình diễn hình ảnh trong slide
* .clone() còn được ứng dụng trong xử lý thông tin form, như tạo thêm giá trị add more,…

### .clone()

<span>Học Web Chuẩn</span>

<div>Hello </div>

<script>

$(function(){

$('span').clone().appendTo('div');

});

</script>

* + <span>Học Web Chuẩn</span>  
    <div>Hello <span>Học Web Chuẩn</span></div>

Text

Description automatically generated

* 1. .height()
* .height() xác định hoặc gán giá trị chiều cao(height) cho thành phần
* .height() được ứng dụng nhiều, xác định chiều cao của window, hay thành phần, hoặc chỉnh chiều cao 2 thành phần cho bằng nhau,…

### .height()

<div>

<p>Thành phần p con của div</p>

</div>

<span></span>

<script>

$(function(){

var hDiv = $('div').height();

$('span').text('Chiều cao của thành phần div: ' + hDiv+'px');

});

</script>

<style>

p { height: 50px; }

</style>

Graphical user interface, text, application

Description automatically generated

.height(val)

<div>

<p>Thành phần p con của div</p>

</div>

<script>

$(function(){

$('div').height(100);

});

</script>

<style>

div { background-color: yellow; }

</style>

Shape, rectangle

Description automatically generated

* 1. .width()
* Giống như .height()

1. jQuery Events
2. jQuery Traversing
   1. Ancestors – Quan hệ cha selector trong Jquery

$("span").parent().css({ … })

* + Lấy một selector cha gần nhất của “span”

$("span").parents().css({ … })

* + Lấy tất cả các selector cha của “span”

$("span").parents("ul").css({ … })

* + Lấy tất cả các selector “ul” cha “span”

$("span").parentsUntil("div").css({ … })

* + Lấy tất cả các selector giữa “span” và “div”

## Descendants – Quan hệ con selector trong Jquery

$("div").children("p.first").css({ … })

* + Lấy tất cả “p.first” con của div

$("div").children().css({ … })

* + Lấy một selector con của div

$("div").find("span").css({ … })

* + Lấy tất cả “span” là con của div

$("div").find("\*").css({ … })

* + Lấy tất selector là con của div

## Siblings – Quan hệ cùng cấp

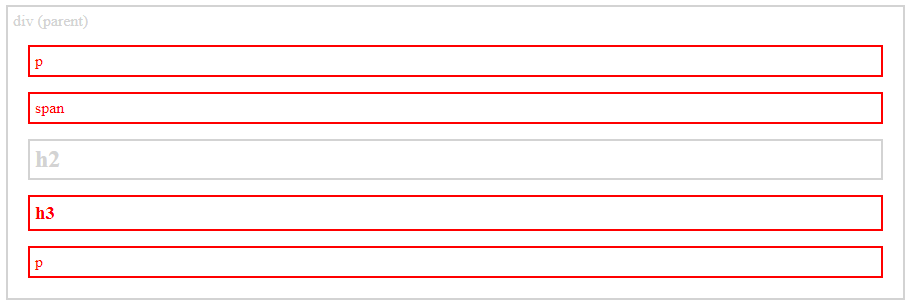
$("h2").siblings().css({ … })

$(document).ready(function(){

$("h2").siblings().css({"color": "red", "border": "2px solid red"});

});

Bỏ qua không CSS cho <h2>



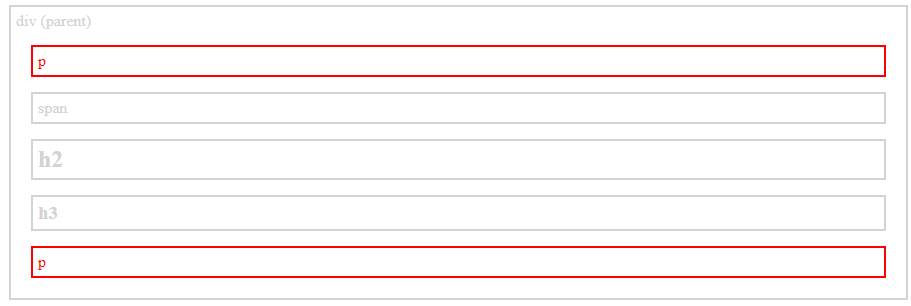
$("h2").siblings("p").css({ … })

$(document).ready(function(){

$("h2").siblings("p").css({"color": "red", "border": "2px solid red"});

});

Bỏ qua không CSS giữa <h2> và <p>



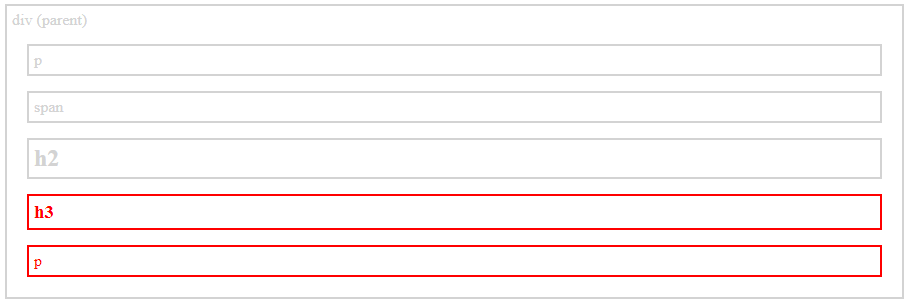
$("h2").nextAll().css({ … })

$(document).ready(function(){

$("h2").nextAll().css({"color": "red", "border": "2px solid red"});

});

CSS tất cả các element sau <h2>



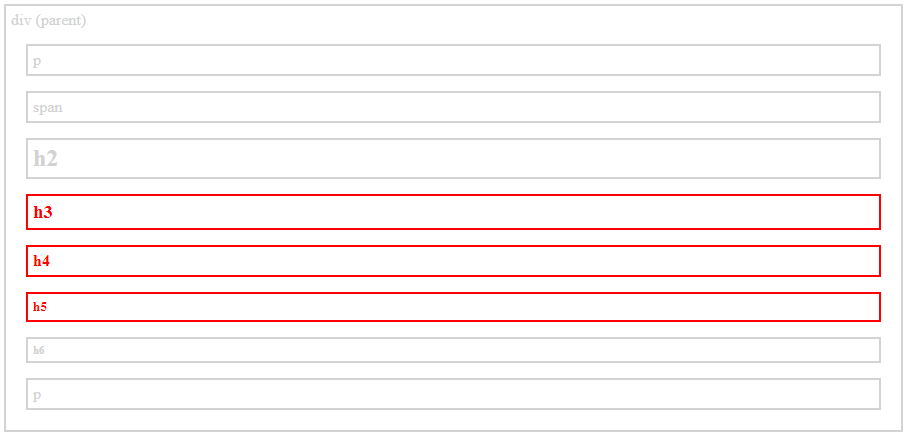
$("h2").nextUntil("h6").css({ … })

$(document).ready(function(){

$("h2").nextUntil("h6").css({"color": "red", "border": "2px solid red"});

});

CSS giữa 2 element <h2> và <h6>



## Filter selector

$("div").first().css(“”, “”)

* + Lấy phần tử <div> đầu tiên

$("div").last().css(“”, “”)

* + Lấy phần tử <div> cuối

$("p").eq(2).css(“”, “”)

* + Lấy phần tử <p> thứ i

$("p").filter(".intro").css(“”, “”)

* + Lấy phần tử <p> có tên class là “intro”

$("p").not(".intro").css(“”, “”)

* + Lấy tất cả các phần tử <p> mà không có tên class là “intro”