**.add()**

*Create a new jQuery object with elements added to the set of matched elements.*

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>add demo</title>

<style>

div {

width: 60px;

height: 60px;

margin: 10px;

float: left;

}

p {

clear: left;

font-weight: bold;

font-size: 16px;

color: blue;

margin: 0 10px;

padding: 2px;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<div></div>

<div></div>

<div></div>

<div></div>

<div></div>

<div></div>

<p>Added this... (notice no border)</p>

<script>

$( "div" ).css( "border", "2px solid red" )

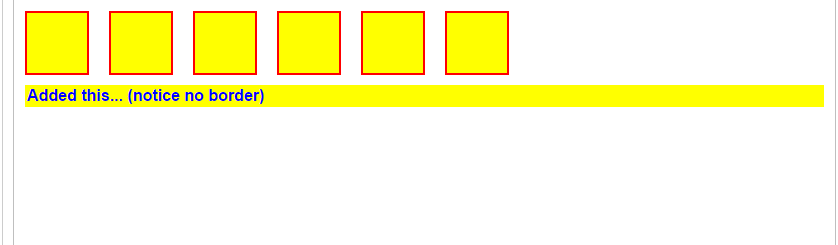
.add( "p" )

.css( "background", "yellow" );

</script>

</body>

</html>



Adds more elements, matched by the given expression, to the set of matched elements.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>add demo</title>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<p>Hello</p>

<span>Hello Again</span>

<script>

$( "p" ).add( "span" ).css( "background", "yellow" );

</script>

</body>

</html>

Hello

Hello Again

Adds more elements, created on the fly, to the set of matched elements.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>add demo</title>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<p>Hello</p>

<script>

$( "p" ).clone().add( "<span>Again</span>" ).appendTo( document.body );

</script>

</body>

</html>

Hello

Hello

Again

Adds one or more Elements to the set of matched elements.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>add demo</title>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<p>Hello</p>

<span id="a">Hello Again</span>

<script>

$( "p" ).add( document.getElementById( "a" ) ).css( "background", "yellow" );

</script>

</body>

Hello

Hello Again

Demonstrates how to add (or push) elements to an existing collection

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>add demo</title>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<p>Hello</p>

<span id="a">Hello Again</span>

<script>

**var** collection = $( "p" );

*// Capture the new collection*

collection = collection.add( document.getElementById( "a" ) );

collection.css( "background", "yellow" );

</script>

</body>

</html>

Hello

Hello Again

# .addBack()

*Add the previous set of elements on the stack to the current set, optionally filtered by a selector.*

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>addBack demo</title>

<style>

p, div {

margin: 5px;

padding: 5px;

}

**.border** {

border: 2px solid red;

}

**.background** {

background: yellow;

}

**.left**, **.right** {

width: 45%;

float: left;

}

**.right** {

margin-left: 3%;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<div class="left">

<p><strong>Before <code>addBack()</code></strong></p>

<div class="before-addback">

<p>First Paragraph</p>

<p>Second Paragraph</p>

</div>

</div>

<div class="right">

<p><strong>After <code>addBack()</code></strong></p>

<div class="after-addback">

<p>First Paragraph</p>

<p>Second Paragraph</p>

</div>

</div>

<script>

$( "div.left, div.right" ).find( "div, div > p" ).addClass( "border" );

*// First Example*

$( "div.before-addback" ).find( "p" ).addClass( "background" );

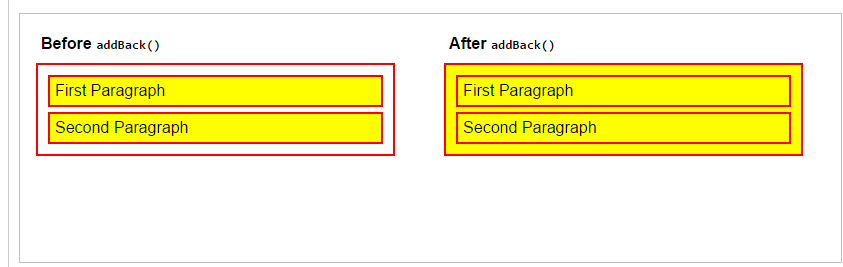
*// Second Example*

$( "div.after-addback" ).find( "p" ).addBack().addClass( "background" );

</script>

</body>

</html>



# .children()

# *Get the children of each element in the set of matched elements, optionally filtered by a selector.*

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>children demo</title>

<style>

body {

font-size: 16px;

font-weight: bolder;

}

div {

width: 130px;

height: 82px;

margin: 10px;

float: left;

border: 1px solid blue;

padding: 4px;

}

**#container** {

width: auto;

height: 105px;

margin: 0;

float: none;

border: none;

}

**.hilite** {

border-color: red;

}

**#results** {

display: block;

color: red;

}

p, span, em, a, b, button {

border: 1px solid transparent;

}

p {

margin: 10px;

}

span {

color: blue;

}

input {

width: 100px;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<div id="container">

<div>

<p>This <span>is the <em>way</em> we</span>

write <em>the</em> demo,</p>

</div>

<div>

<a href="#"><b>w</b>rit<b>e</b></a> the <span>demo,</span> <button>write

the</button> demo,

</div>

<div>

This <span>the way we <em>write</em> the <em>demo</em> so</span>

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

</div>

<p>

<span>t</span>he <span>m</span>orning.

<span id="results">Found <span>0</span> children in <span>TAG</span>.</span>

</p>

</div>

<script>

$( "#container" ).click(**function** ( event ) {

$( "\*" ).removeClass( "hilite" );

**var** kids = $( event.target ).children();

**var** len = kids.addClass( "hilite" ).length;

$( "#results span:first" ).text( len );

$( "#results span:last" ).text( event.target.tagName );

event.preventDefault();

});

</script>

</body>

</html>

# 

# Find all children of each div.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>children demo</title>

<style>

body {

font-size: 16px;

font-weight: bolder;

}

span {

color: blue;

}

p {

margin: 5px 0;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<p>Hello (this is a paragraph)</p>

<div><span>Hello Again (this span is a child of the a div)</span></div>

<p>And <span>Again</span> (in another paragraph)</p>

<div>And One Last <span>Time</span> (most text directly in a div)</div>

<script>

$( "div" ).children().css( "border-bottom", "3px double red" );

</script>

</body>

</html>

**Hello (this is a paragraph)**

**Hello Again (this span is a child of the a div)**

**And Again (in another paragraph)**

**And One Last Time (most text directly in a div)**

# Find all children with a class "selected" of each div.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>children demo</title>

<style>

body {

font-size: 16px;

font-weight: bolder;

}

p {

margin: 5px 0;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<div>

<span>Hello</span>

<p class="selected">Hello Again</p>

<div class="selected">And Again</div>

<p>And One Last Time</p>

</div>

<script>

$( "div" ).children( ".selected" ).css( "color", "blue" );

</script>

</body>

</html>

**Hello**

**Hello Again**

**And Again**

**And One Last Time**

# .closest()

*For each element in the set, get the first element that matches the selector by testing the element itself and traversing up through its ancestors in the DOM tree.*

The [.parents()](http://api.jquery.com/parents/) and .closest() methods are similar in that they both traverse up the DOM tree. The differences between the two, though subtle, are significant:

| **.closest()** | [**.parents()**](http://api.jquery.com/parents/) |
| --- | --- |
| Begins with the current element | Begins with the parent element |
| Travels up the DOM tree until it finds a match for the supplied selector | Travels up the DOM tree to the document's root element, adding each ancestor element to a temporary collection; it then filters that collection based on a selector if one is supplied |
| The returned jQuery object contains zero or one element for each element in the original set, in document order | The returned jQuery object contains zero or more elements for each element in the original set, in reverse document order |

Show how event delegation can be done with closest. The closest list element toggles a yellow background when it or its descendent is clicked.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>closest demo</title>

<style>

li {

margin: 3px;

padding: 3px;

background: #EEEEEE;

}

li**.hilight** {

background: yellow;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<ul>

<li><b>Click me!</b></li>

<li>You can also <b>Click me!</b></li>

</ul>

<script>

$( document ).on( "click", **function**( event ) {

$( event.target ).closest( "li" ).toggleClass( "hilight" );

});

</script>

</body>

</html>

* **Click me!**
* You can also **Click me!**

Pass a jQuery object to closest. The closest list element toggles a yellow background when it or its descendent is clicked.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>closest demo</title>

<style>

li {

margin: 3px;

padding: 3px;

background: #EEEEEE;

}

li**.hilight** {

background: yellow;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<ul>

<li><b>Click me!</b></li>

<li>You can also <b>Click me!</b></li>

</ul>

<script>

**var** listElements = $( "li" ).css( "color", "blue" );

$( document ).on( "click", **function**( event ) {

$( event.target ).closest( listElements ).toggleClass( "hilight" );

});

</script>

</body>

</html>

* **Click me!**
* You can also **Click me!**

# .each()

# *Iterate over a jQuery object, executing a function for each matched element.*

# Iterate over three divs and sets their color property.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>each demo</title>

<style>

div {

color: red;

text-align: center;

cursor: pointer;

font-weight: bolder;

width: 300px;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<div>Click here</div>

<div>to iterate through</div>

<div>these divs.</div>

<script>

$( document.body ).click(**function**() {

$( "div" ).each(**function**( i ) {

**if** ( **this**.style.color !== "blue" ) {

**this**.style.color = "blue";

} **else** {

**this**.style.color = "";

}

});

});

</script>

</body>

</html>

**Click here**

**to iterate through**

**these divs.**

# To access a jQuery object instead of the regular DOM element, use $( this ).

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>each demo</title>

<style>

ul {

font-size: 18px;

margin: 0;

}

span {

color: blue;

text-decoration: underline;

cursor: pointer;

}

**.example** {

font-style: italic;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

To do list: <span>(click here to change)</span>

<ul>

<li>Eat</li>

<li>Sleep</li>

<li>Be merry</li>

</ul>

<script>

$( "span" ).click(**function**() {

$( "li" ).each(**function**() {

$( **this** ).toggleClass( "example" );

});

});

</script>

</body>

</html>

To do list: (click here to change)

* Eat
* Sleep
* Be merry

Use return false to break out of each() loops early.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>each demo</title>

<style>

div {

width: 40px;

height: 40px;

margin: 5px;

float: left;

border: 2px blue solid;

text-align: center;

}

span {

color: red;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<button>Change colors</button>

<span></span>

<div></div>

<div></div>

<div></div>

<div></div>

<div id="stop">Stop here</div>

<div></div>

<div></div>

<div></div>

<script>

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

$( "div" ).each(**function**( index, element ) {

*// element == this*

$( element ).css( "backgroundColor", "yellow" );

**if** ( $( **this** ).is( "#stop" ) ) {

$( "span" ).text( "Stopped at div index #" + index );

**return** false;

}

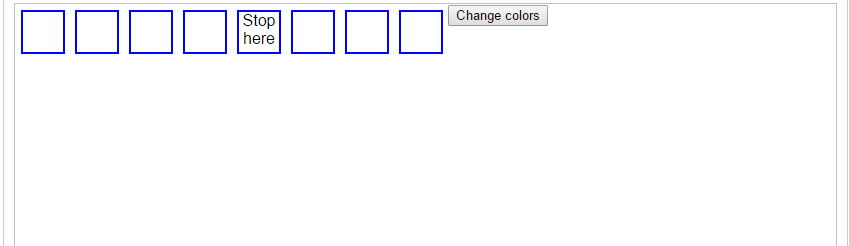
});

});

</script>

</body>

</html>



# .eq()

# *Reduce the set of matched elements to the one at the specified index.*

# Turn the div with index 2 blue by adding an appropriate class.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>eq demo</title>

<style>

div {

width: 60px;

height: 60px;

margin: 10px;

float: left;

border: 2px solid blue;

}

**.blue** {

background: blue;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<div></div>

<div></div>

<div></div>

<div></div>

<div></div>

<div></div>

<script>

$( "body" ).find( "div" ).eq( 2 ).addClass( "blue" );

</script>

</body>

</html>

# 

# .find()

*Get the descendants of each element in the current set of matched elements, filtered by a selector, jQuery object, or element.*

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>find demo</title>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<p><span>Hello</span>, how are you?</p>

<p>Me? I'm <span>good</span>.</p>

<script>

$( "p" ).find( "span" ).css( "color", "red" );

</script>

</body>

</html>

Hello, how are you?

Me? I'm good.

A selection using a jQuery collection of all span tags. Only spans within p tags are changed to red while others are left blue.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>find demo</title>

<style>

span {

color: blue;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<p><span>Hello</span>, how are you?</p>

<p>Me? I'm <span>good</span>.</p>

<div>Did you <span>eat</span> yet?</div>

<script>

**var** spans = $( "span" );

$( "p" ).find( spans ).css( "color", "red" );

</script>

</body>

</html>

Hello, how are you?

Me? I'm good.

Did you eat yet?

Add spans around each word then add a hover and italicize words with the letter **t**.

**<!doctype html>**

<html lang="en">

<head>

<meta charset="utf-8">

<title>find demo</title>

<style>

p {

font-size: 20px;

width: 200px;

color: blue;

font-weight: bold;

margin: 0 10px;

}

**.hilite** {

background: yellow;

}

</style>

<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

</head>

<body>

<p>

When the day is short

find that which matters to you

or stop believing

</p>

<script>

**var** newText = $( "p" ).text().split( " " ).join( "</span> <span>" );

newText = "<span>" + newText + "</span>";

$( "p" )

.html( newText )

.find( "span" )

.hover(**function**() {

$( **this** ).addClass( "hilite" );

}, **function**() {

$( **this** ).removeClass( "hilite" );

})

.end()

.find( ":contains('t')" )

.css({

"font-style": "italic",

"font-weight": "bolder"

});

</script>

</body>

</html>

**When *the* day is**

***short*find *that***

**which *matters* *to***

**you or *stop* believing**