JavaScript / jQuery DOM Selectors

❮ Previous Next ❯

jQuery vs JavaScript

jQuery was created in 2006 by John Resig. It was designed to handle Browser Incompatibilities and to simplify HTML DOM Manipulation, Event Handling, Animations, and Ajax.

For more than 10 years, jQuery has been the most popular JavaScript library in the world.

However, after JavaScript Version 5 (2009), most of the jQuery utilities can be solved with a few lines of standard JavaScript:

Finding HTML Element by Id

Return the element with id="id01":

jQuery

```
myElement = $("#id01");
```

Try it Yourself »

JavaScript

Finding HTML Elements by Tag Name

Return all <p> elements:

jQuery

```
myElements = $("p");
```

Try it Yourself »

JavaScript

```
myElements = document.getElementsByTagName("p");
```

Try it Yourself »

Finding HTML Elements by Class Name

Return all elements with class="intro".

jQuery

```
myElements = $(".intro");
```

JavaScript

```
myElements = document.getElementsByClassName("intro");
```

Try it Yourself »

Finding HTML Elements by CSS Selectors

Return a list of all <p> elements with class="intro".

jQuery

```
myElements = $("p.intro");
```

Try it Yourself »

JavaScript

```
myElements = document.querySelectorAll("p.intro");
```

Try it Yourself »

❮ Previous Log in to track progress Next ❯

JavaScript / jQuery HTML Elements

❮ Previous Next ❯

jQuery vs JavaScript

jQuery was created in 2006 by John Resig. It was designed to handle Browser Incompatibilities and to simplify HTML DOM Manipulation, Event Handling, Animations, and Ajax.

For more than 10 years, jQuery has been the most popular JavaScript library in the world.

However, after JavaScript Version 5 (2009), most of the jQuery utilities can be solved with a few lines of standard JavaScript:

Set Text Content

Set the inner text of an HTML element:

jQuery

```
myElement.text("Hello Sweden!");
```

Try it Yourself »

JavaScript

Get Text Content

Get the inner text of an HTML element:

jQuery

```
myText = $("#02").text();
```

Try it Yourself »

JavaScript

```
myText = document.getElementById("02").textContent;
```

Try it Yourself »

Set HTML Content

Set the HTML content of an element:

jQuery

```
myElement.html("<p>Hello World</p>");
```

JavaScript

```
myElement.innerHTML = "<p>Hello World</p>";
```

Try it Yourself »

Get HTML Content

Get the HTML content of an element:

jQuery

```
content = myElement.html();
```

Try it Yourself »

JavaScript

```
content = myElement.innerHTML;
```

Try it Yourself »

❮ Previous Log in to track progress Next ❯

JavaScript / jQuery CSS Styles

❮ Previous Next ❯

jQuery vs JavaScript

jQuery was created in 2006 by John Resig. It was designed to handle Browser Incompatibilities and to simplify HTML DOM Manipulation, Event Handling, Animations, and Ajax.

For more than 10 years, jQuery has been the most popular JavaScript library in the world.

However, after JavaScript Version 5 (2009), most of the jQuery utilities can be solved with a few lines of standard JavaScript:

Hiding HTML Elements

Hide an HTML Element:

jQuery

```
myElement.hide();
```

Try it Yourself »

JavaScript

Showing HTML Elements

Show an HTML Element:

jQuery

```
myElement.show();
```

Try it Yourself »

JavaScript

```
myElement.style.display = "";
```

Try it Yourself »

Styling HTML Elements

Change the font size of an HTML element:

jQuery

```
$("#demo").css("font-size","35px");
```

Try it Yourself »

JavaScript / jQuery HTML DOM

❮ Previous Next ❯

jQuery vs JavaScript

jQuery was created in 2006 by John Resig. It was designed to handle Browser Incompatibilities and to simplify HTML DOM Manipulation, Event Handling, Animations, and Ajax.

For more than 10 years, jQuery has been the most popular JavaScript library in the world.

However, after JavaScript Version 5 (2009), most of the jQuery utilities can be solved with a few lines of standard JavaScript:

Removing HTML Elements

Remove an HTML element:

jQuery

```
$("#id02").remove();
```

Try it Yourself »

JavaScript

Get Parent Element

Return the parent of an HTML element:

jQuery

```
myParent = $("#02").parent().prop("nodeName"); ;
```

Try it Yourself »

JavaScript

```
myParent = document.getElementById("02").parentNode.nodeName;
```

Try it Yourself »

❮ Previous Log in to track progress Next ❯

jQuery Traversing

❮ Previous Next ❯

What is Traversing?

jQuery traversing, which means "move through", are used to "find" (or select) HTML elements based on their relation to other elements. Start with one selection and move through that selection until you reach the elements you desire.

The image below illustrates an HTML page as a tree (DOM tree). With jQuery traversing, you can easily move up (ancestors), down (descendants) and sideways (siblings) in the tree, starting from the selected (current) element. This movement is called traversing - or moving through - the DOM tree.

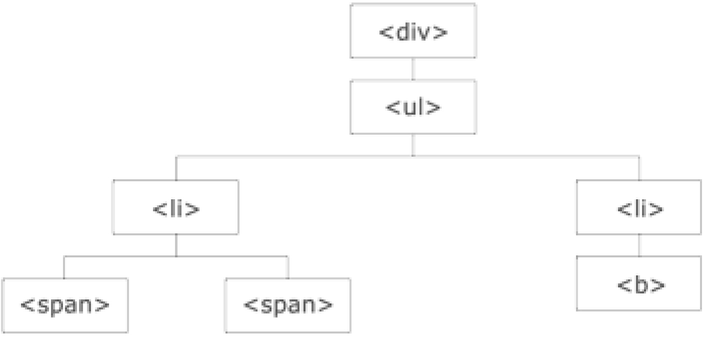

Illustration explained:

- The <div> element is the **parent** of <ul>, and an **ancestor** of everything inside of it
- The <ul> element is the **parent** of both <li> elements, and a **child** of <div>
- The left <li> element is the **parent** of <span>, **child** of <ul> and a **descendant** of <div>
- The <span> element is a **child** of the left <li> and a **descendant** of <ul> and <div>
- The two <li> elements are **siblings** (they share the same parent)

An ancestor is a parent, grandparent, great-grandparent, and so on.
A descendant is a child, grandchild, great-grandchild, and so on.
Siblings share the same parent.

Traversing the DOM

jQuery provides a variety of methods that allow us to traverse the DOM.

The largest category of traversal methods are tree-traversal.

The next chapters will show us how to travel up, down and sideways in the DOM tree.

jQuery Traversing Reference

For a complete overview of all jQuery Traversing methods, please go to our jQuery Traversing Reference.

❮ Previous Log in to track progress Next ❯