# jQuery

## Basics

It is a JavaScript library specialized for changing web page documents on the fly.

jQuery uses selector engine.

|  |  |
| --- | --- |
| **JavaScript** | **jQuery** |
| document.getElementByTagName(“p”)  [0].innerHTML = “Change”; | $(“p”).html(“Change”) |
| function init(){} window.onload = init; | $(document).ready(function(){}); |

The JavaScript interpreter doesn’t change the original HTML and CSS files. It makes changes to the DOM’s representation of the page in the browser’s memory.

The dollar sign “$” with the parentheses is the shorter name of the jQuery function. This shortcut saves us from writing jQuery(). The jQuery function is also often referred to as the **jQuery wrapper**.

jQuery() 🡪 $()

$() can have css selector, JavaScript Object, HTML

If we put a **CSS selector** here, jQuery will return us the **set of elements** that match that selector

If we put a string of **HTML** in here, we can add DOM elements to the browser page.

**JavaScript Object** can also be used as a selector.

### CSS Selector

CSS selectors select elements to add style to those elements; jQuery selectors select elements to add behavior to those elements.

### Element Selector

h1{

text-align: left;

}

### Class Selector

.my\_class{

position: absolute;

}

### ID Selector

#my\_id{

color: #3300FF

}

### jQuery methods

.hide() 🡪 Hides the element

.show() 🡪 Shows the element

.toggle() 🡪 Hide and Show the element

.slideUp() 🡪 It changes the height property of the element until its 0

.slideDown() 🡪 It changes the height property of the element from 0

.slideToggle() 🡪 Slides up and down the element

.fadeIn() 🡪 Fades in the element

.fadeOut() 🡪 Fades out the element

.fadeTo() 🡪 Fades to the element

<!DOCTYPE html>

<**html**>

<**head**>

<**title**> jQuery </**title**>

<**meta** charset = "UTF-8"/>

<**script** src = "../Required/jquery.js"></**script**>

<**script** src = "basics.js"></**script**>

</**head**>

<**body**>

<**p** id = "text"> Hide and Show </**p**>

<**button** id = "hide">Hide</**button**>

<**button** id = "show">Show</**button**>

<**button** id = "toggle">Toggle</**button**><**br**/><**br**/>

<**button** id = "slide-up">Slide Up</**button**>

<**button** id = "slide-down">Slide Down</**button**>

<**button** id = "slide-toggle">Slide Toggle</**button**><**br**/><**br**/>

<**button** id = "fade-in">Fade In</**button**>

<**button** id = "fade-out">Fade Out</**button**>

</**body**>

</**html**>

$(document).ready(**function**(){

$("#hide").click(**function**(){

$("#text").hide(2000);

});

$("#show").click(**function**(){

$("#text").show(2000);

});

$("#toggle").click(**function**(){

$("#text").toggle(2000);

});

$("#slide-up").click(**function**(){

$("#text").slideUp(2000);

});

$("#slide-down").click(**function**(){

$("#text").slideDown(2000);

});

$("#slide-toggle").click(**function**(){

$("#text").slideToggle(2000);

});

$("#fade-in").click(**function**(){

$("#text").fadeIn(2000);

});

$("#fade-out").click(**function**(){

$("#text").fadeOut(2000);

});

});

**Code 1: basics.js**

**$(this)**, this selector gives us an easy way to point to the current element. It’s important to think $(this) as **context-dependent.**

### Descendant Selectors

html

↑

body

↑

div

↑

div div p id = “good”

↑ ↑

img p

$(“div p#good”)

$(“div div”)

$(“div p”)

$(“div div img”)

## Binding an Event

* **$(“#myElement”).click(function(){})**
* **$(“#myElement”).bind(“click”,function(){})**

The first method is simply a shortcut for second method, but only **when the DOM elements exist already**

**Selector + Event + Function = Complex Interaction**

|  |  |
| --- | --- |
| **Events** | **Description** |
| click | Mouse Events 🡪 Click |
| dblclick | Mouse Events 🡪 Double Click |
| focusin | Mouse Events 🡪 Focus In |
| focusout | Mouse Events 🡪 Focus Out |
| hover | Mouse Events 🡪 Hover |
| mousedown | Mouse Events 🡪 Mouse Down |
| mouseenter | Mouse Events 🡪 Mouse Enter |
| mouseleave | Mouse Events 🡪 Mouse Leave |
| mousemove | Mouse Events 🡪 Mouse Move |
| mouseout | Mouse Events 🡪 Mouse Out |
| mouseover | Mouse Events 🡪 Mouse Over |
| mouseup | Mouse Events 🡪 Mouse Up |
| toggle | Mouse Events 🡪 Toggle |
| keydown | Keyboard Events 🡪 Press Down Arrow |
| keypress | Keyboard Events 🡪 Press Enter |
| keypup | Keyboard Events 🡪 Press Up Arrow |
| blur | Form Events 🡪 Focus out of the field box |
| change | Form Events 🡪 Change in the field Text |
| focus | Form Events 🡪 Focusing in the field box |
| select | Form Events 🡪 Selecting the field box |
| submit | Form Events 🡪 Submit Clicked |
| load | Document Events 🡪 Page Fully Loaded |
| ready | Document Events 🡪 Page is ready |
| unload | Document Events 🡪 Page unloaded |
| error | Browser Events 🡪 Error in Browser |
| resize | Browser Events 🡪 Resize the Browser |
| scroll | Browser Events 🡪 Scroll through the Browser |

### Removing Events

$(“#myElement”).unbind(“click”) 🡪 Removes the click event from myElement

$(“#myElement”).unbind() 🡪 Removes all the events from myElement

### Going through Elements

$(“.nav-items”).each(function(){});

<!DOCTYPE html>

<**html**>

<**head**>

<**title**> Selectors</**title**>

<**meta** charset = "UTF-8"/>

<**script** src = "../Required/jquery.js"></**script**>

<**script** src = "selectors.js"></**script**>

</**head**>

<**body**>

<**div** id = "image-container">

<**img** src = "images/desktop.ico" class = "picture"/>

<**img** src = "images/hdd.ico" class = "picture"/>

<**img** src = "images/trash.ico" class = "picture"/>

<**img** src = "images/folder.ico" class = "picture"/>

</**div**>

</**body**>

</**html**>

$(document).ready(**function**(){

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

$(**this**).slideUp();

**var** discount = Math.floor((Math.random()\*5 + 5));

**var** message = "<p> Your discount price is: " + discount + "%</p>";

$("#image-container").append(message);

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

$(**this**).unbind("click");

});

$(".picture").hide(5000);

});

});

**Code 2: selectors.js**

### Checking Element

**$.contains(document.getElementById(“parent”), document.getElementById(“child”))**

This method check if the first parameter (parent) contains the second parameter(child)

<!DOCTYPE html>

<**html**>

<**head**>

<**title**>Check</**title**>

<**meta** charset = "UTF-8"/>

<**script** src = "../Required/jquery.js"></**script**>

<**script** src = "check.js"></**script**>

</**head**>

<**body**>

<**div** id = "container">

<**button** id = "check"> Check </**button**>

</**div**>

<**button** id = "outsider"> Outsider</**button**>

</**body**>

</**html**>

$(document).ready(**function**(){

$("#check").click(**function**(){

console.log($.contains(document.getElementById("container"), **this**));

**var** outsider = document.getElementById("outsider");

console.log($.contains(document.getElementById("container"), outsider));

});

});

**Code 3: check.js**

### CSS Hover

<!DOCTYPE html>

<**html**>

<**head**>

<**title**>Check</**title**>

<**meta** charset = "UTF-8"/>

<**script** src = "../Required/jquery.js"></**script**>

<**script** src = "hover.js"></**script**>

</**head**>

<**body**>

<**div** id = "container">

<**button** id = "hover"> Hover </**button**>

</**div**>

</**body**>

<**style**>

#hover{

**border**: **none**;

**padding**: 5px;

transition: 0.5s;

}

**.hover-on**{

**color**: rgba(140, 234, 12, 0.5);

**background-color**: rgba(0, 0, 0, 0.6);

transition: 0.5s;

}

</**style**>

</**html**>

$(document).ready(**function**(){

$("#hover").bind("mouseover",**function**(){

**if**($(**this**).hasClass("hover-on")){}

**else**{$(**this**).addClass("hover-on");}

});

$("#hover").bind("mouseleave", **function**(){

**if**($(**this**).hasClass("hover-on")){

$(**this**).removeClass("hover-on");

}

});

});

**Code 4: hover.js**

## Modify the Document

**$(“img#thumbnail”).remove()**

This removes the method drops the element out of the DOM

**$(“img#thumbnail”).detach()**

This detach method takes the selected elements(s) but holds on to it so that it can be reattached later.

**$(“.fish”).parent()**

It selects all the elements in the fish class and then get the element above those element

**$(“.menu\_list”).children()**

It selects all the elements in the menu\_list get all the elements below it

**$(“.fish”).prev()**

Selects all the element of fish class and get the immediate left sibling

**$(“.fish”).next()**

Selects all the element of fish class and get the immediate right sibling

**$(“h3”).replaceWith(“<h1>My Menu</h1>”);**

Selects all the h3 elements, replace with selected elements with the contents in parenthesis.

It works well when we have a one-to-one replacement.

**$(“.meat”).before(“<li>Tofu</li>”);**

before inserts content before the selected element

**$(“.meat”).after(“<li>Tofu</li>”);**

after inserts content after the selected element

**$(“.menu\_list”).children().first();**

The first method will filter out everything but the first element in a selected set of elements

**$(“.menu\_list”).children().last();**

The last method will filter out everything bu the last element in a selected set of elements.

**$(“.menu\_list”).children().eq(0);**

The **eq** method will filter out everything out the element whose index number is equal to what we put in the parenthesis in a selected set of elements.

**$(“.menu\_list”).children().slice(1,3);**

The slice method will filter out everything but elements with an index between the index number we put it parenthesis.

**$(“.menu\_list”).children().filter(“.organic”);**

The filter method will filter out everything but elements that match the selector we put in its parenthesesis

**$(“.menu\_list”).children().not(“.local”);**

The not method will filter out everything that does not match the selector we place in the parentheses.

### Storing the Variable

**$f = $(“.fish”).parent().parent().detach();**

jQuery variable

**$f[2] = 15;**

jQuery array, **“$”** it is just a coding convention