

Get started

To embed jQuery in your project, download source code from <https://jquery.com/download/> or include jQuery CDN in the head of html file.

Selectors `$()`

In addition to similar CSS selectors, jQuery selectors include:

- index selectors: `:eq(index)`, `:lt(index)`, `:gt(index)`
`$("li:lt(3)")` /*Selects elements with index 0, 1, 2*/
- content selectors: `:contains(text)`, `:has(selector)`, `:empty`, `:parent`
`$("p:contains(hello)")` /*Selects <p> elements that contain text "hello"*/
`$("p:has(button)")` /*Selects all <p> elements that contain <button>*/
- form selectors `type(:button, :file, :input, etc)/status(:selected, :checked, :disabled, :enabled, :focus)`
`$(":button")` /*Select all <button> elements and elements of type button*/

Traversing `$(selector).method([param])`

In addition to methods shown in slides, jQuery traversal methods include:

- **Ancestors**
 - `.closest(selector [, context])`
`$("div#selected").closest("div")` /*The closest <div> ancestor (including itself) of <div> element with id "selected".*/
 - `.offsetParent()`
`$("p").offsetParent()` /*The closest ancestor of <p> elements that have a position attribute*/
- **Filtering**
 - `is(selector)` returns true, false
`$("#button").parent().is("div")` /*True if the parent of the element with id "button" is a <div> element; otherwise, false.*/
 - `.not(selector)`
`$("p").not(".test")` /*All <p> elements that do not have a class "test"*/
 - `.slice(start_index [, end_index])`
`$("li").slice(0, 3)` /*The first three elements*/

CSS/HTML Manipulation

Structure Manipulation

Get	Update	Add	Remove
<code>.text()</code>	<code>.text(new_text)</code>	<code>.append(content)</code>	<code>.remove(filter)</code>
<code>.html()</code>	<code>.html(new_html)</code>	<code>.prepend(content)</code>	<code>.detach(filter)</code>
<code>.val()</code>	<code>.val(new_val)</code>	<code>.after(content)</code>	<code>.empty()</code>
<code>.attr(attr_name)</code>	<code>.attr(attr_name, new_value)</code>	<code>.before(content)</code>	<code>.removeAttr(attr_name)</code>
		<code>.wrap(wrapper)</code>	<code>.unwrap()</code>

Style Manipulation

Class	Style	Dimension (get or set)
<code>.hasClass(class_name)</code>	<code>.css(property_name)</code>	<code>.width()</code> <code>.height(400)</code>
<code>.addClass(class_names)</code> <code>.addClass("first second")</code>	<code>.css(property_name, value)</code>	<code>.innerWidth()</code> <code>.innerHeight()</code>
<code>.removeClass(class_names)</code>	<code>.css(property_object)</code> <code>.css({backgroundColor:"red", ... })</code>	<code>.outerWidth()</code> <code>.outerHeight()</code>
<code>.toggleClass(class_names)</code>		<code>.outerWidth(true)</code> <code>.outerHeight(true)</code>

Event & Animation

- jQ Event
 - Properties: `.target` / `.pageX` / `.pageY` / `.which`
 - Create an Event
 - `var e = $.Event("click");` //Same as `jQuery.Event("click")`
 - `var e = $.Event("keydown", {keyCode: 64})`
 - Trigger an Event: `$("body").trigger(e);`
 - Add/Remove Listener
 - `$("selector").eventName(function);`
 - `$("selector").on("eventName", function);`
 - `$("selector").off(["eventName", handler]);`
 - eventName example: click, keypress, mouseenter
- Effect and Animation
 - Effect
 - e.g. `$("selector").fadeIn([speed, callback]);`
 - Toggle: If element currently shown, execute the effect to hide element; vise-versa
 - effect list: show/hide, fadeIn/fadeOut/fadeToggle/fadeTo, slideDown/slideUp/slideToggle
 - Animation
 - `$("selector").animate({CSSparameters} [, speed, callback]);`
 - Stop Animation: `$("selector").stop([stopAll, goToEnd]);`

AJAX

Helper	Core
<code>\$.get(url [,payload, success, dataType])</code>	<code>\$.ajax({ url: "server/location" })</code>
<code>\$.post(url [,payload, success, dataType])</code>	<code>\$.ajax().done(function(data){ })</code> <code>\$.ajax().fail(function(xhr, status, error){ })</code> <code>\$.ajax().always(function(xhr, status){ })</code>
<code>\$(selector).load(url [, payload, success])</code>	