

# jQuery Quick Reference

Selectors, DOM, events, AJAX, effects, traversal

## Selectors

### Basic Selectors

```
$( "p" ) // all <p> elements
$( ".btn" ) // class selector
$( "#main" ) // ID selector
$( "ul > li" ) // direct children
$( "input[type='text']" ) // attribute selector
```

### Common Selectors

<code>\$( '*' )</code>	All elements
<code>\$( 'a, span' )</code>	Multiple selectors
<code>\$( ':first' )</code>	First matched element
<code>\$( ':last' )</code>	Last matched element
<code>\$( ':even' ) / \$( ':odd' )</code>	Even/odd indexed elements
<code>\$( ':visible' ) / \$( ':hidden' )</code>	Visibility filters
<code>\$( ':contains(text)' )</code>	Elements containing text
<code>\$( ':has(selector)' )</code>	Elements containing selector match

## DOM Manipulation

### Content & Structure

```
$( "#el" ).html("<b>Bold</b>"); // set inner HTML
$( "#el" ).text("Plain text"); // set text content
$( "ul" ).append("<li>New</li>"); // add at end
$( "ul" ).prepend("<li>First</li>"); // add at start
```

### Insert & Remove

```
$( "<p>Hello</p>" ).insertAfter("#el"); // insert after
$( "<p>Hi</p>" ).insertBefore("#el"); // insert before
$( "#el" ).remove(); // remove element from DOM
$( "#el" ).empty(); // remove all children
```

### Methods

<code>.html()</code>	Get/set inner HTML
<code>.text()</code>	Get/set text content
<code>.append()</code> / <code>.prepend()</code>	Add child at end/start
<code>.after()</code> / <code>.before()</code>	Insert sibling after/before
<code>.wrap()</code>	Wrap element with new parent
<code>.clone()</code>	Deep copy element
<code>.replaceWith()</code>	Replace element

## Events

### Binding Events

```
$( ".btn" ).on("click", function () {
    $(this).toggleClass("active");
});
$(document).on("click", ".dynamic", handler); // delegation
$( ".btn" ).off("click"); // remove handler
```

### Shorthand Methods

<code>.click(fn)</code>	Click handler
<code>.hover(fnIn, fnOut)</code>	Mouse enter/leave
<code>.focus()</code> / <code>.blur()</code>	Focus gained/lost
<code>.change(fn)</code>	Value changed (input/select)
<code>.submit(fn)</code>	Form submission
<code>.keydown()</code> / <code>.keyup()</code>	Keyboard events

### Document Ready

```
$(document).ready(function () { /* DOM loaded */ });
$(function () { /* shorthand */ });
```

## Effects & Animation

### Show / Hide / Toggle

```
$( "#el" ).hide(400); // hide with animation
$( "#el" ).show(400); // show with animation
$( "#el" ).toggle(); // toggle visibility
$( "#el" ).fadeIn(600); // fade in
$( "#el" ).slideDown(400); // slide down
```

### Custom Animation

```
$( "#el" ).animate({
    opacity: 0.5,
    left: "+=50px",
    height: "toggle"
}, 1000, function () { /* complete */ });
```

### Effect Methods

<code>.fadeIn()</code> / <code>.fadeOut()</code>	Animate opacity
<code>.fadeToggle()</code>	Toggle with fade
<code>.slideDown()</code> / <code>.slideUp()</code>	Animate height
<code>.animate(props, ms)</code>	Custom CSS animation
<code>.stop()</code>	Stop current animation
<code>.delay(ms)</code>	Delay next queued effect

## AJAX

### Shorthand Methods

```
$.get("/api/data", function (data) { console.log(data); });
$.post("/api/data", { name: "Alice" }, function (res) {});
$.getJSON("/api/items", function (json) {});
$( "#el" ).load("/fragment.html"); // load HTML into element
```

### Full \$.ajax

```
$.ajax({
    url: "/api/data",
    method: "POST",
    data: JSON.stringify({ key: "val" }),
    contentType: "application/json",
    success: (res) => console.log(res),
    error: (xhr) => console.error(xhr.status)
});
```

### AJAX Options

<b>url</b>	Request URL
<b>method</b>	HTTP method (GET, POST, ...)
<b>data</b>	Data to send
<b>dataType</b>	Expected response type
<b>contentType</b>	Request content type
<b>success / error</b>	Callback functions
<b>headers</b>	Custom request headers

## Traversal

### Tree Navigation

```
$( "li" ).parent(); // direct parent
$( "li" ).parents("ul"); // all matching ancestors
$( "ul" ).children("li"); // direct children
$( "ul" ).find(".active"); // all matching descendants
$( "li" ).siblings(); // sibling elements
```

## Traversal Methods

<code>.parent()</code> / <code>.parents()</code>	Direct parent / all ancestors
<code>.children()</code> / <code>.find()</code>	Direct children / all descendants
<code>.siblings()</code>	All sibling elements
<code>.next()</code> / <code>.prev()</code>	Next/previous sibling
<code>.closest(sel)</code>	Nearest ancestor matching selector
<code>.first()</code> / <code>.last()</code>	First/last in matched set
<code>.eq(index)</code>	Element at index
<code>.filter(sel)</code>	Reduce to matching elements

## Attributes & CSS

### Attributes

```
$( "img" ).attr("src"); // get attribute
$( "img" ).attr("src", "new.jpg"); // set attribute
$( "img" ).removeAttr("title"); // remove attribute
$( "input" ).prop("checked", true); // set property
```

### CSS & Classes

```
$( ".box" ).css("color", "red"); // set single style
$( ".box" ).css({ color: "red", fontSize: "14px" }); // multi
$( "p" ).addClass("highlight");
$( "p" ).removeClass("highlight");
$( "p" ).toggleClass("active");
$( "p" ).hasClass("active"); // returns boolean
```

## Forms

### Values & Serialization

```
$( "input" ).val(); // get value
$( "input" ).val("new value"); // set value
$( "form" ).serialize(); // "name=Alice&age=30"
$( "form" ).serializeArray(); // [{name, value}, ...]
```

### Form Selectors

<code>\$( ":input" )</code>	All form elements
<code>\$( ":checked" )</code>	Checked checkboxes/radios
<code>\$( ":selected" )</code>	Selected option elements
<code>\$( ":disabled" )</code>	Disabled elements
<code>\$( ":text" )</code>	Text inputs

## Utilities

### Array & Object Helpers

```
$.each([1, 2, 3], function (i, val) { /* iterate */ });
$.each(obj, function (key, val) { /* iterate object */ });
$.extend({}, defaults, options); // merge objects
$.inArray("b", ["a", "b", "c"]); // returns 1 (index)
```

### Utility Methods

<code>\$.each(arr, fn)</code>	Iterate array or object
<code>\$.extend(target, ...)</code>	Merge objects (shallow)
<code>\$.inArray(val, arr)</code>	Index of value (-1 if missing)
<code>\$.isArray(obj)</code>	Check if array
<code>\$.trim(str)</code>	Remove leading/trailing whitespace
<code>\$.parseJSON(str)</code>	Parse JSON string
<code>\$.map(arr, fn)</code>	Transform array elements

## Common Patterns

### Chaining

```
$( "ul" )
    .find("li")
    .addClass("item")
    .first()
    .css("font-weight", "bold");
```

# jQuery Quick Reference

---

## Plugin Pattern

```
$.fn.highlight = function (color) {  
  return this.each(function () {  
    $(this).css("background", color || "yellow");  
  });  
};  
$("p").highlight("pink");
```

## Conflict Resolution

```
// Avoid $ conflicts with other libraries  
jQuery.noConflict();  
jQuery(document).ready(function ($) {  
  $(".btn").click(handler); // $ safe inside  
});
```