

# Referência Rápida de jQuery

Seletores, DOM, eventos, AJAX, efeitos, navegação

## Seletores

### Seletores Básicos

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

### Seletores Comuns

<code>\$('.*')</code>	Todos os elementos
<code>\$('.a, span')</code>	Múltiplos seletores
<code>\$('.:first')</code>	Primeiro elemento correspondente
<code>\$('.:last')</code>	Último elemento correspondente
<code>\$('.:even')</code> / <code>\$('.:odd')</code>	Elementos de índice par/ímpar
<code>\$('.:visible')</code> / <code>\$('.:hidden')</code>	Filtros de visibilidade
<code>\$('.:contains(text)')</code>	Elementos que contém texto
<code>\$('.:has(selector)')</code>	Elementos que contém correspondência do seletor

## Manipulação do DOM

### Conteúdo e Estrutura

```
$("#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
```

### Inserir e Remover

```
$("#<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
```

### Métodos

<code>.html()</code>	Obter/definir HTML interno
<code>.text()</code>	Obter/definir conteúdo de texto
<code>.append()</code> / <code>.prepend()</code>	Adicionar filho no final/início
<code>.after()</code> / <code>.before()</code>	Inserir irmão depois/antes
<code>.wrap()</code>	Envolver elemento com novo pai
<code>.clone()</code>	Cópia profunda do elemento
<code>.replaceWith()</code>	Substituir elemento

## Eventos

### Vinculando Eventos

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

### Métodos Abreviados

<code>.click(fn)</code>	Manipulador de clique
<code>.hover(fnIn, fnOut)</code>	Entrada/saída do mouse
<code>.focus()</code> / <code>.blur()</code>	Foco obtido/perdido
<code>.change(fn)</code>	Valor alterado (input/select)
<code>.submit(fn)</code>	Envio de formulário
<code>.keydown()</code> / <code>.keyup()</code>	Eventos de teclado

### Documento Pronto

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

## Efeitos e Animação

### Mostrar / Ocultar / Alternar

```
$("#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
```

### Animação Personalizada

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

### Métodos de Efeito

<code>.fadeIn()</code> / <code>.fadeOut()</code>	Animar opacidade
<code>.fadeToggle()</code>	Alternar com fade
<code>.slideDown()</code> / <code>.slideUp()</code>	Animar altura
<code>.animate(props, ms)</code>	Animação CSS personalizada
<code>.stop()</code>	Parar animação atual
<code>.delay(ms)</code>	Atrasar próximo efeito na fila

## AJAX

### Métodos Abreviados

```
$.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
```

### \$.ajax Completo

```
$.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)
});
```

### Opções de AJAX

<code>url</code>	URL da requisição
<code>method</code>	Método HTTP (GET, POST, ...)
<code>data</code>	Dados a enviar
<code>dataType</code>	Tipo de resposta esperada
<code>contentType</code>	Tipo de conteúdo da requisição
<code>success</code> / <code>error</code>	Funções de callback
<code>headers</code>	Cabeçalhos de requisição personalizados

## Navegação

### Navegação na Árvore

```
$("#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
```

## Métodos de Navegação

<code>.parent()</code> / <code>.parents()</code>	Pai direto / todos os ancestrais
<code>.children()</code> / <code>.find()</code>	Filhos diretos / todos os descendentes
<code>.siblings()</code>	Todos os elementos irmãos
<code>.next()</code> / <code>.prev()</code>	Irmão próximo/anterior
<code>.closest(sel)</code>	Ancestral mais próximo que corresponde ao seletor
<code>.first()</code> / <code>.last()</code>	Primeiro/último no conjunto correspondente
<code>.eq(index)</code>	Elemento no índice
<code>.filter(sel)</code>	Reduzir aos elementos correspondentes

## Atributos e CSS

### Atributos

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

### CSS e 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
```

## Formulários

### Valores e Serialização

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

### Seletores de Formulário

<code>\$("#:input")</code>	Todos os elementos de formulário
<code>\$("#:checked")</code>	Checkboxes/rádios marcados
<code>\$("#:selected")</code>	Elementos de opção selecionados
<code>\$("#:disabled")</code>	Elementos desabilitados
<code>\$("#:text")</code>	Inputs de texto

## Utilitários

### Auxiliares de Array e Objeto

```
$.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)
```

### Métodos Utilitários

<code>\$.each(arr, fn)</code>	Iterar array ou objeto
<code>\$.extend(target, ...)</code>	Mesclar objetos (superficial)
<code>\$.inArray(val, arr)</code>	Índice do valor (-1 se ausente)
<code>\$.isArray(obj)</code>	Verificar se é array
<code>\$.trim(str)</code>	Remover espaços no início/fim
<code>\$.parseJSON(str)</code>	Analisar string JSON
<code>\$.map(arr, fn)</code>	Transformar elementos do array

# Referência Rápida de jQuery

---

## Padrões Comuns

### Encadeamento

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

### Padrão de Plugin

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

### Resolução de Conflitos

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