

Referencia Rápida de jQuery

Selectores, DOM, eventos, AJAX, efectos, recorrido

Selectores

Selectores básicos

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

Selectores comunes

<code>\$('.*')</code>	Todos los elementos
<code>\$('.a, span')</code>	Selectores múltiples
<code>\$('.:first')</code>	Primer elemento coincidente
<code>\$('.:last')</code>	Último elemento coincidente
<code>\$('.:even')</code> / <code>\$('.:odd')</code>	Elementos con índice par/impar
<code>\$('.:visible')</code> / <code>\$('.:hidden')</code>	Filtros de visibilidad
<code>\$('.:contains(text)')</code>	Elementos que contienen texto
<code>\$('.:has(selector)')</code>	Elementos que contienen una coincidencia del selector

Manipulación del DOM

Contenido y estructura

```
$("#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
```

Insertar y eliminar

```
$("#<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>	Obtener/establecer HTML interno
<code>.text()</code>	Obtener/establecer contenido de texto
<code>.append()</code> / <code>.prepend()</code>	Agregar hijo al final/inicio
<code>.after()</code> / <code>.before()</code>	Insertar hermano después/antes
<code>.wrap()</code>	Envolver el elemento con un nuevo padre
<code>.clone()</code>	Copia profunda del elemento
<code>.replaceWith()</code>	Reemplazar el elemento

Eventos

Vincular 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>	Manejador de clic
<code>.hover(fnIn, fnOut)</code>	Entrada/salida del ratón
<code>.focus()</code> / <code>.blur()</code>	Foco ganado/perdido
<code>.change(fn)</code>	Valor cambiado (input/select)
<code>.submit(fn)</code>	Envío del formulario
<code>.keydown()</code> / <code>.keyup()</code>	Eventos de teclado

Document ready

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

Efectos y animación

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
```

Animación personalizada

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

Métodos de efecto

<code>.fadeIn()</code> / <code>.fadeOut()</code>	Animar opacidad
<code>.fadeToggle()</code>	Alternar con fundido
<code>.slideDown()</code> / <code>.slideUp()</code>	Animar altura
<code>.animate(props, ms)</code>	Animación CSS personalizada
<code>.stop()</code>	Detener animación actual
<code>.delay(ms)</code>	Retrasar el siguiente efecto en cola

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)
});
```

Opciones de AJAX

<code>url</code>	URL de la petición
<code>method</code>	Método HTTP (GET, POST, ...)
<code>data</code>	Datos a enviar
<code>dataType</code>	Tipo de respuesta esperado
<code>contentType</code>	Tipo de contenido de la petición
<code>success</code> / <code>error</code>	Funciones de callback
<code>headers</code>	Cabeceras personalizadas de la petición

Recorrido del DOM

Navegación por el árbol

```
$("#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 recorrido

<code>.parent()</code> / <code>.parents()</code>	Padre directo / todos los ancestros
<code>.children()</code> / <code>.find()</code>	Hijos directos / todos los descendientes
<code>.siblings()</code>	Todos los elementos hermanos
<code>.next()</code> / <code>.prev()</code>	Hermano siguiente/anterior
<code>.closest(sel)</code>	Ancstro más cercano que coincide con el selector
<code>.first()</code> / <code>.last()</code>	Primero/último del conjunto coincidente
<code>.eq(index)</code>	Elemento en el índice
<code>.filter(sel)</code>	Reducir a los elementos coincidentes

Atributos y 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 y clases

```
$(".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
```

Formularios

Valores y serialización

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

Selectores de formulario

<code>\$("#:input")</code>	Todos los elementos de formulario
<code>\$("#:checked")</code>	Checkboxes/radios marcados
<code>\$("#:selected")</code>	Opciones seleccionadas
<code>\$("#:disabled")</code>	Elementos deshabilitados
<code>\$("#:text")</code>	Inputs de texto

Utilidades

Helpers de array y 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 de utilidad

<code>\$.each(arr, fn)</code>	Iterar array u objeto
<code>\$.extend(target, ...)</code>	Fusionar objetos (superficial)
<code>\$.inArray(val, arr)</code>	Índice del valor (-1 si no existe)
<code>\$.isArray(obj)</code>	Verificar si es array
<code>\$.trim(str)</code>	Eliminar espacios al inicio/final
<code>\$.parseJSON(str)</code>	Parsear cadena JSON
<code>\$.map(arr, fn)</code>	Transformar elementos del array

Referencia Rápida de jQuery

Patrones comunes

Encadenamiento

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

Patrón de plugin

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

Resolución de conflictos

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