

# Referensi Cepat jQuery

Selector, DOM, event, AJAX, efek, traversal

## Selector

### Selector Dasar

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

### Selector Umum

<code>\$( '*' )</code>	Semua elemen
<code>\$( 'a, span' )</code>	Beberapa selector
<code>\$( ':first' )</code>	Elemen pertama yang cocok
<code>\$( ':last' )</code>	Elemen terakhir yang cocok
<code>\$( ':even' ) / \$( ':odd' )</code>	Elemen dengan indeks genap/ganjil
<code>\$( ':visible' ) / \$( ':hidden' )</code>	Filter visibilitas
<code>\$( ':contains(text)' )</code>	Elemen yang mengandung teks
<code>\$( ':has(selector)' )</code>	Elemen yang mengandung kecocokan selector

## Manipulasi DOM

### Konten & Struktur

```
$( "#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
```

### Sisipkan & Hapus

```
$( "<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
```

## Method

<code>.html()</code>	Get/set inner HTML
<code>.text()</code>	Get/set konten teks
<code>.append()</code> / <code>.prepend()</code>	Tambah anak di akhir/awal
<code>.after()</code> / <code>.before()</code>	Sisipkan sibling setelah/sebelum
<code>.wrap()</code>	Bungkus elemen dengan parent baru
<code>.clone()</code>	Salin elemen secara mendalam
<code>.replaceWith()</code>	Ganti elemen

## Event

### Mengikat Event

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

### Method Shorthand

<code>.click(fn)</code>	Handler klik
<code>.hover(fnIn, fnOut)</code>	Mouse masuk/keluar
<code>.focus()</code> / <code>.blur()</code>	Fokus diperoleh/hilang
<code>.change(fn)</code>	Nilai berubah (input/select)
<code>.submit(fn)</code>	Pengiriman form
<code>.keydown()</code> / <code>.keyup()</code>	Event keyboard

### Document Ready

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

## Efek & Animasi

### Tampilkan / Sembunyikan / 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
```

### Animasi Custom

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

### Method Efek

<code>.fadeIn()</code> / <code>.fadeOut()</code>	Animasikan opacity
<code>.fadeToggle()</code>	Toggle dengan fade
<code>.slideDown()</code> / <code>.slideUp()</code>	Animasikan tinggi
<code>.animate(props, ms)</code>	Animasi CSS custom
<code>.stop()</code>	Hentikan animasi saat ini
<code>.delay(ms)</code>	Tunda efek antrian berikutnya

## AJAX

### Method Shorthand

```
$.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 Lengkap

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

### Opsi AJAX

<code>url</code>	URL request
<code>method</code>	Metode HTTP (GET, POST, ...)
<code>data</code>	Data yang dikirim
<code>dataType</code>	Tipe respons yang diharapkan
<code>contentType</code>	Tipe konten request
<code>success</code> / <code>error</code>	Fungsi callback
<code>headers</code>	Header request custom

## Traversal

### Navigasi Pohon

```
$( "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
```

## Method Traversal

<code>.parent()</code> / <code>.parents()</code>	Parent langsung / semua ancestor
<code>.children()</code> / <code>.find()</code>	Anak langsung / semua keturunan
<code>.siblings()</code>	Semua elemen sibling
<code>.next()</code> / <code>.prev()</code>	Sibling berikutnya/sebelumnya
<code>.closest(sel)</code>	Ancestor terdekat yang cocok dengan selector
<code>.first()</code> / <code>.last()</code>	Pertama/terakhir dalam set yang cocok
<code>.eq(index)</code>	Elemen pada indeks
<code>.filter(sel)</code>	Kurangi ke elemen yang cocok

## Atribut & CSS

### Atribut

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

### CSS & Class

```
$( ".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
```

## Form

### Nilai & Serialisasi

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

### Selector Form

<code>\$( ":input" )</code>	Semua elemen form
<code>\$( ":checked" )</code>	Checkbox/radio yang dicentang
<code>\$( ":selected" )</code>	Elemen option yang dipilih
<code>\$( ":disabled" )</code>	Elemen yang dinonaktifkan
<code>\$( ":text" )</code>	Input teks

## Utilitas

### Helper Array & Objek

```
$.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)
```

### Method Utilitas

<code>\$.each(arr, fn)</code>	Iterasi array atau objek
<code>\$.extend(target, ...)</code>	Gabungkan objek (shallow)
<code>\$.inArray(val, arr)</code>	Indeks nilai (-1 jika tidak ada)
<code>\$.isArray(obj)</code>	Periksa apakah array
<code>\$.trim(str)</code>	Hapus spasi di awal/akhir
<code>\$.parseJSON(str)</code>	Parse string JSON
<code>\$.map(arr, fn)</code>	Transformasi elemen array

## Pola Umum

### Chaining

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

# Referensi Cepat jQuery

---

## Pola Plugin

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

## Resolusi Konflik

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