

jQuery クイックリファレンス

セレクター、DOM、イベント、AJAX、エフェクト、トラバーサル

セレクター

基本セレクター

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

よく使うセレクター

<code>\$('*')</code>	すべての要素
<code>\$('a, span')</code>	複数セレクター
<code>\$(':first')</code>	最初にマッチした要素
<code>\$(':last')</code>	最後にマッチした要素
<code>\$(':even') / \$(':odd')</code>	偶数 / 奇数インデックスの要素
<code>\$(':visible') / \$(':hidden')</code>	表示 / 非表示フィルター
<code>\$(':contains(text)')</code>	テキストを含む要素
<code>\$(':has(selector)')</code>	セレクターにマッチする子孫を持つ要素

DOM 操作

コンテンツと構造

```
$( "#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
```

挿入と削除

```
$( "<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
```

メソッド

<code>.html()</code>	内部 HTML を取得 / 設定
<code>.text()</code>	テキストコンテンツを取得 / 設定
<code>.append()</code> / <code>.prepend()</code>	末尾 / 先頭に子要素を追加
<code>.after()</code> / <code>.before()</code>	後 / 前に兄弟要素を挿入
<code>.wrap()</code>	要素を新しい親要素で包む
<code>.clone()</code>	要素を深くコピー
<code>.replaceWith()</code>	要素を置換

イベント

イベントのバインド

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

ショートハンドメソッド

<code>.click(fn)</code>	クリックハンドラー
<code>.hover(fnIn, fnOut)</code>	マウスエンター / リーブ
<code>.focus()</code> / <code>.blur()</code>	フォーカス取得 / 喪失
<code>.change(fn)</code>	値変更 (input/select)
<code>.submit(fn)</code>	フォーム送信
<code>.keydown()</code> / <code>.keyup()</code>	キーボードイベント

ドキュメント準備完了

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

エフェクトとアニメーション

表示 / 非表示 / 切り替え

```
$( "#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
```

カスタムアニメーション

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

エフェクトメソッド

<code>.fadeIn()</code> / <code>.fadeOut()</code>	不透明度をアニメーション
<code>.fadeToggle()</code>	フェードで切り替え
<code>.slideDown()</code> / <code>.slideUp()</code>	高さをアニメーション
<code>.animate(props, ms)</code>	カスタム CSS アニメーション
<code>.stop()</code>	現在のアニメーションを停止
<code>.delay(ms)</code>	次のキューに入ったエフェクトを遅延

AJAX

ショートハンドメソッド

```
$.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 完全形

```
$.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 オプション

<code>url</code>	リクエスト URL
<code>method</code>	HTTP メソッド (GET、POST など)
<code>data</code>	送信するデータ
<code>dataType</code>	期待するレスポンスタイプ
<code>contentType</code>	リクエストのコンテンツタイプ
<code>success / error</code>	コールバック関数
<code>headers</code>	カスタムリクエストヘッダー

トラバーサル

ツリーのナビゲーション

```
$( "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
```

トラバーサルメソッド

<code>.parent()</code> / <code>.parents()</code>	直接の親 / すべての祖先
<code>.children()</code> / <code>.find()</code>	直接の子 / すべての子孫
<code>.siblings()</code>	すべての兄弟要素
<code>.next()</code> / <code>.prev()</code>	次 / 前の兄弟
<code>.closest(sel)</code>	セレクターにマッチする最も近い祖先
<code>.first()</code> / <code>.last()</code>	マッチセットの最初 / 最後
<code>.eq(index)</code>	インデックスの要素
<code>.filter(sel)</code>	マッチする要素に絞り込む

属性と CSS

属性

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

CSS とクラス

```
$( ".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
```

フォーム

値とシリアライズ

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

フォームセレクター

<code>\$(":input")</code>	すべてのフォーム要素
<code>\$(":checked")</code>	チェックされたチェックボックス / ラジオ
<code>\$(":selected")</code>	選択された option 要素
<code>\$(":disabled")</code>	無効化された要素
<code>\$(":text")</code>	テキスト入力

ユーティリティ

配列とオブジェクトのヘルパー

```
$.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)
```

ユーティリティメソッド

<code>\$.each(arr, fn)</code>	配列またはオブジェクトを反復
<code>\$.extend(target, ...)</code>	オブジェクトをマージ (浅いコピー)
<code>\$.inArray(val, arr)</code>	値のインデックス (見つからない場合 -1)
<code>\$.isArray(obj)</code>	配列かどうかをチェック
<code>\$.trim(str)</code>	前後の空白を除去
<code>\$.parseJSON(str)</code>	JSON 文字列をパース
<code>\$.map(arr, fn)</code>	配列要素を変換

jQuery クイックリファレンス

よく使うパターン

チェーン

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

プラグインパターン

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

競合解消

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