

# 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

```
$( "*" ) // すべての要素
$( "a, span" ) // 複数セレクター
$( ":first" ) // 最初にマッチした要素
$( ":last" ) // 最後にマッチした要素
$( ":even" ) / $( ":odd" ) // 偶数 / 奇数インデックスの要素
```

```
$( ":visible" ) / $( ":hidden" ) // 表示 / 非表示フィルター
$( ":contains(text)" ) // テキストを含む要素
$( ":has(selector)" ) // 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( "<div>Bold</div>" ); // 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>Hi</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

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

```
$.click(fn) // クリックハンドラー
$.hover(fnIn, fnOut) // マウスエンター / リーブ
$.focus() / $.blur() // フォーカス取得 / 喪失
$.change(fn) // 値変更 (input/select)
$.submit(fn) // フォーム送信
$.keydown() / $.keyup() // 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

```
$.fadeIn() / $.fadeOut() // 不透明度をアニメーション
$.fadeToggle() // フェードで切り替え
$.slideDown() / $.slideUp() // 高さをアニメーション
$.animate(props, ms) // カスタム CSS アニメーション
$.stop() // 現在のアニメーションを停止
$.delay(ms) // 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

```
url // リクエスト URL
method // HTTP メソッド (GET, POST など)
data // 送信するデータ
dataType // 期待するレスポンスタイプ
```

```
contentType // リクエストのコンテンツタイプ
success / error // コールバック関数
headers // 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

```
$.parent() / $.parents() // 直接の親 / すべての祖先
$.children() / $.find() // 直接の子 / すべての子孫
$.siblings() // すべての兄弟要素
$.next() / $.prev() // 次 / 前の兄弟
$.closest(sel) // セレクターにマッチする最も近い祖先
```

```
$.first() / $.last() // マッチセットの最初 / 最後
$.eq(index) // インデックスの要素
$.filter(sel) // 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

```
$( ":input" ) // すべてのフォーム要素
$( ":checked" ) // チェックされたチェックボックス / ラジオ
$( ":selected" ) // 選択された option 要素
$( ":disabled" ) // 無効化された要素
$( ":text" ) // 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

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

## 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
});
```