

# REFERENCIA RÁPIDA DE HTMX

Atributos, triggers, intercambio, eventos, extensiones

## Fundamentos

### Instalación

```
<script src="https://unpkg.com/htmx.org?></script>
```

```
<!-- 0 con npm -->
npm install htmx.org
```

### Cómo funciona

**HTML-driven** Lanzar AJAX desde atributos HTML, sin JS

**Intercambio de contenido** El servidor devuelve fragmentos HTML, htmx los intercambia

**Progresivo** Mejora HTML estándar; funciona sin JavaScript

**REST-friendly** Funciona con cualquier framework del servidor

## Atributos

### Atributos de solicitud principales

**hx-get="/url"** Enviar solicitud GET a la URL

**hx-post="/url"** Enviar solicitud POST a la URL

**hx-put="/url"** Enviar solicitud PUT a la URL

**hx-patch="/url"** Enviar solicitud PATCH a la URL

**hx-delete="/url"** Enviar solicitud DELETE a la URL

### Atributos de comportamiento

**hx-trigger** Evento que activa la solicitud

**hx-target** Elemento donde se intercambia el contenido

**hx-swap** Cómo intercambiar el contenido de la respuesta

**hx-select** Seleccionar un subconjunto del HTML de la respuesta

**hx-vals** Agregar valores adicionales a la solicitud

**hx-confirm** Mostrar diálogo de confirmación antes de la solicitud

**hx-disable** Deshabilitar el procesamiento htmx en el elemento

## Triggers

### Sintaxis de trigger

```
<!-- Por defecto: evento natural (click para botones) -->
<button hx-get="/data">Load</button>
```

```
<!-- Evento trigger personalizado -->
<div hx-get="/news" hx-trigger="every 5s">Feed</div>
```

```
<!-- Múltiples triggers -->
<input hx-get="/search" hx-trigger="keyup changed delay:300ms" />
```

### Modificadores de trigger

**changed** Solo activar si el valor cambió

**delay:Ns** Esperar N segundos antes de activar

**throttle:Nms** Limitar a una vez por N ms

**once** Activar solo la primera vez

**from:selector** Escuchar en un elemento diferente

**every Ns** Sondar cada N segundos

**load** Activar al cargar el elemento

**revealed** Activar cuando el elemento entra en la vista

## Targets

### Selección de target

```
<!-- Apuntar a otro elemento -->
<button hx-get="/data" hx-target="#result">Load</button>
<div id="result"></div>
```

```
<!-- Apuntar al ancestro más cercano -->
<button hx-get="/row" hx-target="closest tr">Update</button>
```

```
<!-- Apuntar con selector CSS -->
<button hx-get="/info" hx-target="next .output">Go</button>
```

### Palabras clave de target

**this** El propio elemento (por defecto)

**closest <sel>** Ancestro más cercano que coincide con el selector

**find <sel>** Primer descendiente que coincide con el selector

**next <sel>** Hermano siguiente que coincide con el selector

**previous <sel>** Hermano anterior que coincide con el selector

## Intercambio

### Estrategias de intercambio

**innerHTML** Reemplazar contenido interno (por defecto)

**outerHTML** Reemplazar el elemento target completo

**afterbegin** Insertar al inicio dentro del target

**beforeend** Insertar al final dentro del target

**beforebegin** Insertar antes del target

**afterend** Insertar después del target

**delete** Eliminar el elemento target

**none** Sin intercambio, solo disparar eventos

### Modificadores de intercambio

```
<!-- Intercambio con retraso de transición -->
<div hx-get="/data" hx-swap="innerHTML swap:300ms">
```

```
<!-- Retraso de settle para transiciones CSS -->
<div hx-get="/data" hx-swap="innerHTML settle:500ms">
```

```
<!-- Desplazar al inicio tras el intercambio -->
<div hx-get="/page" hx-swap="innerHTML scroll:top">
```

## Cabeceras

### Cabeceras de solicitud (enviadas por htmx)

**HX-Request** Siempre "true" para solicitudes htmx

**HX-Target** ID del elemento target

**HX-Trigger** ID del elemento que disparó la solicitud

**HX-Trigger-Name** Nombre del elemento que disparó la solicitud

**HX-Current-URL** URL actual del navegador

**HX-Prompt** Respuesta del usuario desde hx-prompt

### Cabeceras de respuesta (enviadas por el servidor)

**HX-Redirect** Redirección del lado del cliente a la URL

**HX-Refresh** Recarga completa de página si es "true"

**HX-Retarget** Anular hx-target con selector CSS

**HX-Reswap** Anular estrategia hx-swap

**HX-Trigger** Disparar eventos del lado del cliente

**HX-Push-Url** Insertar URL en el historial del navegador

## Eventos

### Eventos del ciclo de vida

**htmx:configRequest** Antes de la solicitud; modificar params/cabeceras

**htmx:beforeRequest** Justo antes de que se llame a AJAX

**htmx:afterRequest** Después de que la solicitud se completa

**htmx:beforeSwap** Antes de intercambiar el contenido

**htmx:afterSwap** Después de intercambiar el contenido

**htmx:afterSettle** Después de que el DOM se estabiliza tras el intercambio

**htmx:responseError** El servidor devolvió estado de error

### Escuchar eventos

```
document.body.addEventListener("htmx:afterSwap", (e) => {
  console.log("Swapped:", e.detail.target);
});
```

```
// Cancelar una solicitud
document.body.addEventListener("htmx:configRequest", (e) => {
  if (!confirm("Proceed?")) e.preventDefault();
});
```

## Extensiones

### Usar extensiones

```
<script src="https://unpkg.com/htmx-ext-json-enc"></script>
```

```
<div hx-ext="json-enc">
  <form hx-post="/api/data">
    <input name="title" />
    <button>Submit as JSON</button>
  </form>
</div>
```

### Extensiones populares

**json-enc** Codificar el cuerpo de la solicitud como JSON

**loading-states** Gestionar clases CSS de estado de carga

**head-support** Fusionar etiquetas <head> de las respuestas

**preload** Precargar enlaces al pulsar/pasar el ratón

**sse** Soporte de Server-Sent Events

**ws** Soporte de WebSocket

**response-targets** Diferentes targets para respuestas de error

## Indicadores

### Indicador de carga

```
<button hx-get="/slow" hx-indicator="#spinner">
  Load Data
</button>
<span id="spinner" class="htmx-indicator">Loading...</span>
```

### CSS del indicador

```
.htmx-indicator { display: none; }
.htmx-request .htmx-indicator { display: inline; }
.htmx-request .htmx-indicator { display: inline; }
```

htmx agrega la clase htmx-request al elemento durante las solicitudes

### Deshabilitar durante la solicitud

```
<button hx-post="/submit" hx-disabled-elt="this">
  Submit
</button>
```

hx-disabled-elt agrega el atributo disabled durante la solicitud

## Patrones comunes

### Búsqueda activa

```
<input type="search" name="q"
  hx-get="/search"
  hx-trigger="keyup changed delay:300ms"
  hx-target="#results" />
<div id="results"></div>
```

### Scroll infinito

```
<tr hx-get="/rows?page=2"
  hx-trigger="revealed"
  hx-swap="afterend">
  <td>Loading more...</td>
</tr>
```

### Eliminar con confirmación

```
<button hx-delete="/item/42"
  hx-confirm="Delete this item?"
  hx-target="closest .item"
  hx-swap="outerHTML">
  Delete
</button>
```

### Edición en línea

```
<div hx-get="/edit/1" hx-trigger="click"
  hx-swap="outerHTML">
  Click to edit this text
</div>
```