

HTMX 빠른 참조

속성, 트리거, 스왑, 이벤트, 확장

기본
설치
<pre><script src="https://unpkg.com/htmx.org?></script> <!-- Or via npm --> npm install htmx.org</pre>
동작 방식
HTML-driven HTML 속성으로 AJAX 발행, JS 불필요
Swap content 서버가 HTML 조각 반환, htmx가 교체
Progressive 표준 HTML 항상, 점진적 지원
REST-friendly 모든 서버 사이드 프레임워크와 호환
속성
핵심 요청 속성
hx-get="/url" URL로 GET 요청 발행
hx-post="/url" URL로 POST 요청 발행
hx-put="/url" URL로 PUT 요청 발행
hx-patch="/url" URL로 PATCH 요청 발행
hx-delete="/url" URL로 DELETE 요청 발행
동작 속성
hx-trigger 요청을 트리거하는 이벤트
hx-target 콘텐츠를 교체할 요소
hx-swap 콘텐츠 콘텐츠를 교체하는 방법
hx-select 콘텐츠 HTML의 일부 선택
hx-vals 요청에 추가 값 포함
hx-confirm 요청 전 확인 대화상자 표시
hx-disable 요소의 htmx 처리 비활성화
트리거
트리거 문법
<pre><!-- Default: natural event (click for buttons) --> <button hx-get="/data">Load</button> <!-- Custom trigger event --> <div hx-get="/news" hx-trigger="every 5s">Feed</div> <!-- Multiple triggers --> <input hx-get="/search" hx-trigger="keyup changed delay:300ms" /></pre>
트리거 수정자
changed 값이 변경된 경우에만 실행
delay:Ns N초 대기 후 실행
throttle:Nms Nms당 한 번으로 제한
once 처음 한 번만 트리거
from:selector 다른 요소에서 이벤트 수신
every Ns N초마다 폴링
load 요소 로드 시 실행
revealed 요소가 뷰포트에 나타날 때 실행
디렉트
타겟 선택
<pre><!-- Target another element --> <button hx-get="/data" hx-target="#result">Load</button> <div id="result"></div> <!-- Target closest ancestor --> <button hx-get="/row" hx-target="closest tr">Update</button> <!-- Target with CSS selector --> <button hx-get="/info" hx-target="next .output">Go</button></pre>
타겟 키워드
this 요소 자체 (기본값)
closest <sel> 선택자와 일치하는 가장 가까운 조상
find <sel> 선택자와 일치하는 첫 번째 자손
next <sel> 선택자와 일치하는 다음 형제
previous <sel> 선택자와 일치하는 이전 형제
스왑
스왑 전략
innerHTML 내부 콘텐츠 교체 (기본값)
outerHTML 타겟 요소 전체 교체
afterbegin 타겟 인 맨 앞에 삽입
beforeend 타겟 인 맨 뒤에 추가
beforebegin 타겟 맨 앞에 삽입
afterend 타겟 뒤에 삽입
delete 타겟 요소 제거
none 교체 없이 이벤트만 발생
스왑 수정자
<pre><!-- Swap with transition delay --> <div hx-get="/data" hx-swap="innerHTML swap:300ms"> <!-- Settle delay for CSS transitions --> <div hx-get="/data" hx-swap="innerHTML settle:500ms"> <!-- Scroll to top after swap --> <div hx-get="/page" hx-swap="innerHTML scroll:top"></pre>

헤더

요청 헤더 (htmx가 전송)
HX-Request htmx 요청이면 항상 "true"
HX-Target 타겟 요소의 ID
HX-Trigger 트리거된 요소의 ID
HX-Trigger-Name 트리거된 요소의 name
HX-Current-URL 브라우저 현재 URL
HX-Prompt hx-prompt에 대한 사용자 응답
응답 헤더 (서버가 전송)
HX-Redirect URL로 클라이언트 사이트 리다이렉트
HX-Refresh "true"이면 전체 페이지 새로고침
HX-Retarget CSS 선택자로 hx-target 재정의

HX-Reswap hx-swap 전략 재정의
HX-Trigger 클라이언트 사이드 이벤트 트리거
HX-Push-Url 브라우저 히스토리에 URL 추가

이벤트

생명 주기 이벤트
htmx:configRequest 요청 전, 파라미터/헤더 수정
htmx:beforeRequest AJAX 호출 직전
htmx:afterRequest 요청 완료 후
htmx:beforeSwap 콘텐츠 교체 전
htmx:afterSwap 콘텐츠 교체 후
htmx:afterSettle 스왑 후 DOM이 안정화된 후
htmx:responseError 서버가 에러 상태 반환

이벤트 수신

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

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

확장

확장 사용

```
<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>
```

주요 확장

json-enc	요청 본문을 JSON으로 인코딩
loading-states	로딩 상태 CSS 클래스 관리
head-support	응답의 <head> 태그 병합
preload	mousedown/hover 시 링크 미리 로드
sse	Server-Sent Events 지원
ws	WebSocket 지원
response-targets	에러 응답에 다른 타겟 지정

인디케이터

로딩 인디케이터

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

인디케이터 CSS

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

htmx는 요청 중 요소에 htmx-request 클래스를 추가

요청 중 비활성화

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

hx-disabled-elt는 요청 중 disabled 속성 추가

일단 페턴

실시간 검색

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

무한 스크롤

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

확인 후 삭제

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

인라인 편집

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