EcomLabs

GUÍA PASO A PASO

Popover API: menús y tooltips sin JavaScript

El código del menú de cuenta, el tooltip que se abre con clic y el aviso que solo se cierra con su botón, la animación con CSS y una chuleta con todos los atributos, métodos y eventos de la Popover API. Con la checklist.

PARA	Tiendas y sitios en HTML, Liquid, React o cualquier framework que escriba HTML
PROBADO CON	Playwright 1.62 · Chromium 151, Firefox 155 y WebKit 26.5 · 19 de noviembre de 2026
ACTUALIZADA	19 de noviembre de 2026
DOCUMENTACIÓN	developer.mozilla.org/en-US/docs/Web/API/Popover_API
ARTÍCULO	ecomlabs.dev/g/popover
AUTOR	Luis Morales · @luismoralesdev

Índice

Antes de empezar	Soporte y pruebas
Paso 01	El menú de cuenta
Paso 02	El tooltip con clic
Paso 03	El aviso que no se cierra solo
Paso 04	La animación
Chuleta	Atributos, métodos y eventos
Al terminar	Checklist

ANTES DE EMPEZAR

Soporte y pruebas

El popover básico es Baseline desde el 27 de enero de 2025 (Chrome y Edge 114, Firefox 125, Safari 17, iOS 18.3). Todo el código de esta guía corrió en Chromium 151, Firefox 155 y WebKit 26.5.

PASO 01

El menú de cuenta

```
<button type="button" popovertarget="menu">Mi cuenta</button>
<nav id="menu" popover aria-label="Mi cuenta">
  <ul>
    <li><a href="/cuenta/pedidos">Mis pedidos</a></li>
    <li><a href="/cuenta/direcciones">Direcciones</a></li>
    <li><a href="/cuenta/salir">Cerrar sesión</a></li>
  </ul>
</nav>
```

Para que salga debajo del botón y no centrado en la ventana, usa anchor positioning. `popovertarget` ya hace del botón el ancla implícita, así que basta con `position-area`:

```
/* The menu sits under its button (anchor positioning). */
#menu {
  position-area: bottom span-right;
  margin: 0.25rem 0 0;
}
```

`bottom span-right` lo alinea con el borde izquierdo del botón; el margen quita el `margin: auto` de fábrica y deja 4 px de aire. Sin soporte de `position-area`, el menú sale centrado: funciona igual.

PASO 02

El tooltip con clic

```
<p>Envío exprés
  <button type="button" popovertarget="ayuda-envio" aria-label="Qué es el envío exprés">?</button>
</p>
<p id="ayuda-envio" popover>Llega en un día hábil dentro de la ciudad.</p>
```

- El botón solo muestra `?`: el `aria-label` es obligatorio para que tenga nombre.
- Es `auto`: se cierra afuera o con Escape, y abrirlo cierra el menú.
- Se abre con clic o toque, no al pasar el mouse. `popover="hint"` está pensado para tooltips, pero Safari estable aún no lo tiene y lo trata como `manual`.

PASO 03

El aviso que no se cierra solo

```
<button type="button" popovertarget="aviso" popovertargetaction="show">Ver aviso de envíos</button>
<div id="aviso" popover="manual">
  <p>Los pedidos de hoy salen el lunes.</p>
  <button type="button" popovertarget="aviso" popovertargetaction="hide">Cerrar aviso</button>
</div>
```

`manual` ignora el clic afuera y Escape, y no lo cierra abrir un menú. El primer botón solo abre (`show`), el de dentro solo cierra (`hide`). Para mostrarlo al cargar la página hace falta una línea de JavaScript:

```
document.getElementById("aviso").showPopover();
```

Mostrar un popover no mueve el foco. Si el aviso tiene que oírse con lector de pantalla, pon el texto también en una región viva (`role="status"`) que ya exista en la página.

PASO 04

La animación

```
/* Fade in and out. overlay keeps it in the top layer while it closes. */
[popover] {
  opacity: 0;
  translate: 0 -0.5rem;
  transition:
    opacity 0.15s,
    translate 0.15s,
    overlay 0.15s allow-discrete,
    display 0.15s allow-discrete;
}
[popover]:popover-open {
  opacity: 1;
  translate: 0 0;
}
@starting-style {
  [popover]:popover-open {
    opacity: 0;
    translate: 0 -0.5rem;
  }
}

/* No movement for people who asked for less of it. */
@media (prefers-reduced-motion: reduce) {
  [popover] {
    transition: none;
```

```
  }
}
```

> **CUIDADO**
>
> El orden importa: `@starting-style` va después de `:popover-open`. La entrada se animó en los tres motores; la salida solo en Chromium, porque `overlay` solo existe en Chrome y Edge. En Firefox y Safari el popover se cierra de golpe, sin errores.

CHULETA

Atributos, métodos y eventos

PIEZA	VALORES O USO	QUÉ HACE
`popover`	`auto` (o vacío)	Cierra con clic afuera y Escape; uno abierto a la vez
`popover`	`hint`	Para tooltips; no cierra los `auto`. Sin soporte, cuenta como `manual`
`popover`	`manual`	Solo lo cierras tú; varios a la vez
`popovertarget`	el `id`	En `<button>` o `<input>` de botón: controla el popover
`popovertargetaction`	`toggle`, `show`, `hide`	Qué hace el botón; `toggle` por defecto
`commandfor` + `command`	`toggle-popover`, `show-popover`, `hide-popover`	Lo mismo con Invoker Commands (Baseline desde diciembre de 2025)
`showPopover()`, `hidePopover()`, `togglePopover()`	JavaScript	Abrir, cerrar o alternar desde código
`beforetoggle`	evento	Antes del cambio; `preventDefault()` impide abrir
`toggle`	evento	Después del cambio; trae `oldState` y `newState`
`:popover-open`	CSS	Estilo del estado abierto
`::backdrop`	CSS	Capa de pantalla completa detrás del popover
`position-area`	CSS	Lo coloca junto a su botón (ancla implícita)

AL TERMINAR

Checklist

- [ ] Cada control es un `<button type="button">` con el `id` correcto en `popovertarget`.
- [ ] Menús y tooltips son `auto`; los avisos son `manual` con botón de cerrar.
- [ ] El botón con `?` tiene `aria-label`.
- [ ] El menú sale debajo del botón con `position-area` y se ve bien si sale centrado.
- [ ] No quedó ningún `z-index` ni `role="menu"` de más.
- [ ] La animación respeta `prefers-reduced-motion` y el cierre de golpe se ve bien.
- [ ] Teclado: Enter abre, Tab entra al popover, Escape cierra y regresa al botón.
- [ ] Lo que debe bloquear la página usa `<dialog>` con `showModal()`, no un popover.