

A decorative graphic in the top-left corner consisting of two overlapping parallelograms. The front one is blue and the back one is light green. They are positioned diagonally, with the blue one partially covering the green one.

Drag and drop

## Significado

- Arrastrar un elemento y luego soltarlo sobre otro elemento
- Introducida en la especificación de HTML5
- Cercana a las aplicaciones de escritorio

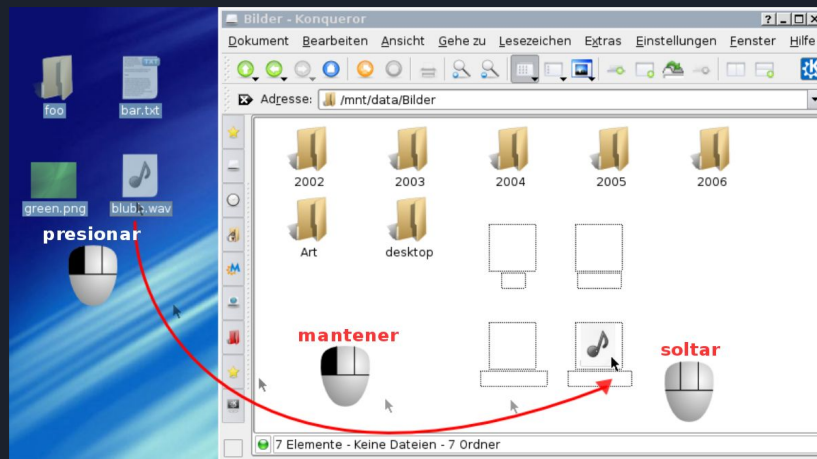
Item 1

Item 2

Item 3

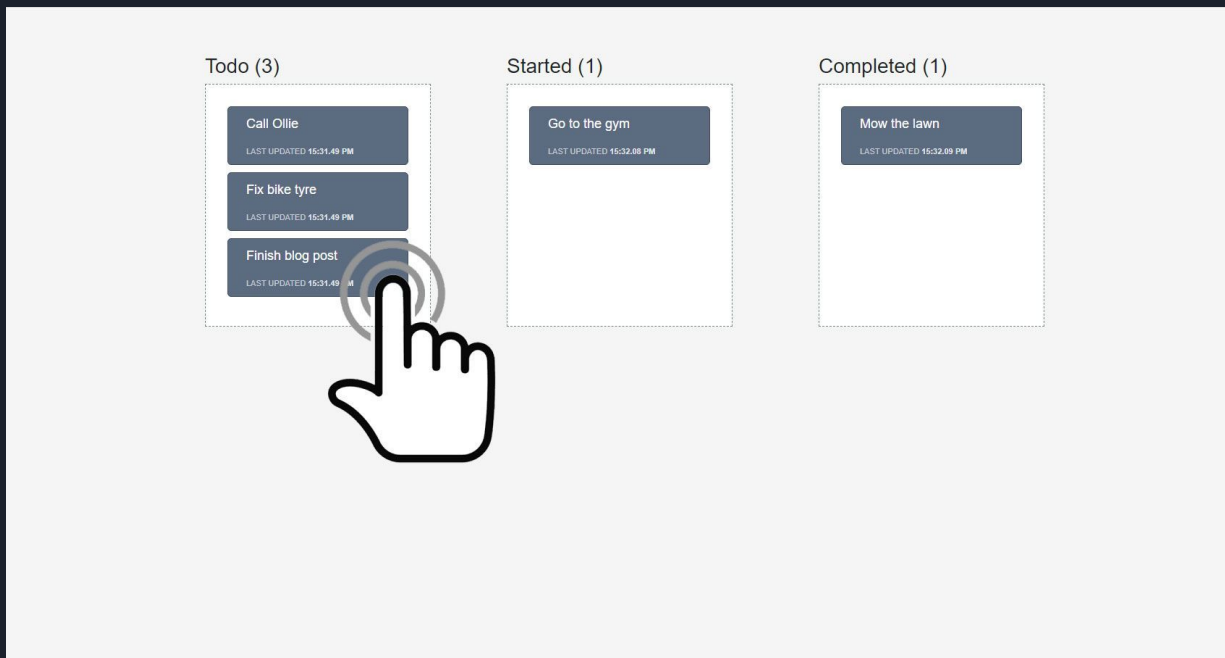
Item 4

Item 5



## Elementos

- Origen (elemento siendo arrastrado)
- Destino (elemento sobre el que se suelta el origen)





## Eventos

- Tenemos eventos distintos para el destino y origen
- Origen
  - dragstart (comienza el arrastre)
  - drag (durante el arrastre)
  - dragend (finaliza el arrastre)
- Destino
  - dragenter (el puntero entra dentro del área destino)
  - dragover (el puntero se encuentra sobre el área destino)
  - dragleave (el puntero sale del área destino)
  - drop (soltamos el elemento origen en el elemento destino)



## Eventos

- Todos los navegadores tienen eventos predefinidos para el drag and drop, por lo que deberemos prevenirlos y personalizarlos en base a nuestras necesidades
- En líneas generales, siempre realizaremos el mismo proceso
  - Prevenimos dragenter
  - Prevenimos dragover
  - Prevenimos y personalizamos drop



## Eventos - DataTransfer

- Objeto que contiene la información relacionada con la operación de arrastrar y soltar
- Este objeto contiene una serie de métodos y atributos que se detallarán a continuación



## Eventos - DataTransfer: Atributos

- **types:** array con los tipos de datos
- **files:** array con información sobre los archivos siendo arrastrados
- **dropEffect:** tipo de operación permitida en destino (none, copy, link y move)
- **effectAllowed:** tipo de operación permitida en origen ( none, copy, copyLink, copyMove, link, linkMove, move, all y uninitialized)



## Eventos - DataTransfer: Métodos

- **setData(tipo, dato):** Declara los datos a ser enviados y su tipo
- **getData(tipo):** Devuelve datos enviados por el origen del tipo especificado
- **clearData(tipo):** Elimina los datos del tipo especificado
- **setDragImage(elemento, x, y):** imagen mostrada durante el drag and drop junto al puntero del ratón. Los valores x e y son relativos al cursor





## Draggable

- Atributo que debemos colocar a cualquier elemento HTML que queramos poder arrastrar

## Ejemplo completo

```
<div>
  <p id="source" ondragstart="dragstart_handler(event);"
draggable="true">
    Elemento que será arrastrado
  </p>
</div>
```

```
<div id="target" ondrop="drop_handler(event);"
ondragover="dragover_handler(event);">Zona de arrastre
</div>
```

```
function dragover_handler(ev) {
  ev.preventDefault();
}
```

```
function drop_handler(ev) {
  ev.preventDefault();
  var data = ev.dataTransfer.getData("text");

  ev.target.appendChild(document.getElementById(data));
}
```

```
function dragstart_handler(ev) {
  ev.dataTransfer.setData("text/plain", ev.target.id);

  var img = new Image();
  img.src = 'example.gif';
  ev.dataTransfer.setDragImage(img, 10, 10);
}
```