Drag and Drop

[DragEvent](https://developer.mozilla.org/en-US/docs/Web/API/DragEvent)

* It was added du to html5, we don't add anything by javascript, you can activate drag and drop option through html, and when you drag and drop elemnts some events happens in our web page. These events can be received by javascript to do some work on them.
* Drag and drop **Handlers**
* First: add **draggable attribute** to the items

<div class="list-item" draggable="true"></div>

**List of drag events**

* **drag** // This event is fired when an element or text selection is being dragged.
* **dragend** // This event is fired when a drag operation is being ended (by releasing a mouse button or hitting the escape key).
* **dragenter** // This event is fired when a dragged element or text selection enters a valid drop target.
* **dragleave** // This event is fired when a dragged element or text selection leaves a valid drop target.
* **dragover** // This event is fired continuously when an element or text selection is being dragged and the mouse pointer is over a valid drop target (every 50 ms WHEN mouse is not moving ELSE much faster between 5 ms (slow movement) and 1ms (fast movement) approximately. This firing pattern is different than mouseover ).
* **dragstart** // This event is fired when the user starts dragging an element or text selection.
* **drop** // This event is fired when an element or text selection is dropped on a valid drop target.

element.addEventListener('dragstart' , this.handleDragStart.bind(this))

element.addEventListener('drageneter' , this.handleDragEneter.bind(this))

element.addEventListener('dragover' , this.handleDragOver.bind(this))

element.addEventListener('dragleave' , this.handleDragLeave.bind(this))

element.addEventListener('drop' , this.handleDragDrop.bind(this))

element.addEventListener('dragend' , this.handleDragEnd.bind(this)) handleDragStart(e) {

console.log(this);

console.log('drag start' , e.target);

}

handleDragEneter(e) {

console.log('drag enter' , e.target);

}

handleDragOver(e) {

console.log('drag over' , e.target);

}

handleDragLeave(e) {

console.log('drag leave' , e.target);

}

handleDragDrop(e) {

console.log('drag drop' , e.target);

}

handleDragEnd(e) {

console.log('drag end' , e.target);

}

**How to implement**

* DragEvent.dataTransfer // To get data and transfer to somewhere else. It has two methods:
* **e.dataTransfer.setData(type, data)**
* **e.dataTransfer.getData()**

handleDragStart(e) {

e.dataTransfer.setData("text/html", e.target.outerHTML)

}

* e.dataTransfer.**effectAllowed** // to define the type we can use. Can be used for other things too. Like copying, move-copying ... [effectAllowed](https://developer.mozilla.org/en-US/docs/Web/API/DataTransfer/effectAllowed)

handleDragStart(e) {

e.dataTransfer.effectAllowed = "move"

}