



DRAG AND DROP



Drag and Drop

- A feature that allows users to easily move elements on a webpage
- Drag and drop is a very common feature. It is when you "grab" an object and drag it to a different location
- Enhances user experience by making interfaces more interactive and intuitive.
- Enables easy reordering of items, such as in to-do lists or image galleries.
- Simplifies complex actions, like file uploads or rearranging content.



- **Drag and Drop Events:**

- **dragstart:** Triggered when the user starts dragging an element.
- **dragover:** Triggered when a draggable element is dragged over a drop target.
- **drop:** Triggered when the user drops the draggable element onto a drop target.

- **Working Together:**

- **dragstart:** Sets the data to be transferred and defines the drag effect.
- **dragover:** Specifies where the dragged element can be dropped.
- **drop:** Handles the drop operation

dragstart Event



- Initiates the drag operation and specifies the data being dragged.

```
<div id="dragElement" draggable="true" ondragstart="drag(event)">Drag me!</div>  
    <script>  
        function drag(event) {  
            event.dataTransfer.setData("text", event.target.id);  
        }  
    </script>
```

When the dragstart event is triggered, the setData method is used to set the data type ("text") and the data value (the ID of the dragged element).

dragover Event



- Specifies where the dragged element can be dropped.

```
<div id="dropTarget" ondragover="allowDrop(event)" ondrop="drop(event)">Drop  
    here</div>  
    <script>  
    function allowDrop(event) {  
        event.preventDefault();  
    }  
    </script>
```

The **allowDrop** function is used to prevent the default behavior of the browser, which is to not allow dropping. This function is necessary for the **drop** event to work properly.

drop Event



- Handles the drop operation and updates the UI accordingly.

```
<div id="dropTarget" ondragover="allowDrop(event)" ondrop="drop(event)">Drop  
    here</div>  
    <script>  
        function drop(event) {  
            event.preventDefault();  
            var data = event.dataTransfer.getData("text");  
            event.target.appendChild(document.getElementById(data));  
        }  
    </script>
```

The **drop** event is triggered when the user drops the dragged element onto the drop target. The **getData** method is used to retrieve the data that was set in the **dragstart** event, and the dropped element is appended to the drop target's children.