

## DOM Manipulation 2

### 🔥 Understanding more events like drag and drop

So drag and drop will be an event in the DOM, because it will be based on user actions. To make an element capable of being dragged from one part to other we have to use the `draggable` attribute. This attribute defines whether the element is `draggable` or not ? It's an enumerated attribute meaning, in whatever HTML tag we are defining this attribute we have to manually give it a `true` or `false` value as a string.

### According to MDN:

**Warning:** This attribute is enumerated and not Boolean. A value of true or false is mandatory, and shorthand like `<img draggable>` is forbidden. The correct usage is `<img draggable="false">`

### Where can we add this draggable in our code ?

```
<section id="non-tier-section">
  <div class="item-container">
    
  </div>
</section>
```

Inside the non-tier-section we are having a lot of divs with class item-container . These divs are going to contain images of our tier-items which needs to be made draggable. So ideally these divs should be given the `draggable = "true"` attribute.

```
<section id="non-tier-section">
  <div class="item-container" draggable="true">
    ...
  </div>
  <div class="item-container" draggable="true">
    ...
  </div>
</section>
```

- But these `item-container` divs we are creating on-the-go using JS, whenever the user fills the form with an image url.
- That means the newly created divs (which are made using JS) should also contain these attributes.

### setAttribute function

This function is available on HTML elements, to set a particular attribute with a value. It takes

two arguments :

- attribute name
- attribute value

```
imageDiv.setAttribute('draggable', 'true');
```

So we can use this `setAttribute` function to set these draggable values.

The div with class `item-container` will be dragged, so we need to add an event listener to this div. If we want to add an event listener to these divs, we need to select them as well. As all of them have got a common class, and we need to attach an event listener to all of these divs, we can use `getElementsByClassName` function.