

JS Browser

ΜΑΘΗΜΑ 2.5
DRAG & DROP

ΠΕΡΙΕΧΟΜΕΝΑ:

1. dragstart & dragend
2. dragover & drop
3. dragleave
4. drag και dragenter
5. Αλλαγή εικόνας drag
6. Παράδειγμα: File Upload

Events Drag & Drop:

- Τα παρακάτω events χρησιμεύουν για να κάνουμε drag & drop:

event	Επεξήγηση
dragstart	Το drag αρχίζει
dragend	Το drag τελειώνει

- Στοιχεία που μπορούν να γίνουν drag:
 - Πρέπει να έχουν το draggable="true" στα χαρακτηριστικά τους

Σημείωση 1:

- Με τα παραπάνω event απλά ενεργοποιούμε τη δυνατότητα κάποιων στοιχείων να γίνουν drag: Στη συνέχεια θα δούμε πως μπορούν να γίνουν drop σε άλλα στοιχεία.

Σημείωση 2:

- Χρησιμοποιείται ο εξής τρόπος για να μεταφερθούν πληροφορίες για το στοιχείο που γίνεται drag, όταν γίνει το drop (επόμενη διαφάνεια)
- Το event.dataTransfer είναι ένα αντικείμενο στο οποίο δίνουμε πληροφορίες για το drop:
 - Χρησιμοποιούμε τη μέθοδο setData(format, data), όπου format είναι έγκυρος τύπος MIME και data τα πραγματικά δεδομένα
 - (συνήθως μεταφέρουμε το id του στοιχείου που γίνεται drag, ώστε να ανασυρθεί στο drop)

Παράδειγμα 1: events-drag-and-drop.html

```
<p>Drag the item and observe the console for events.</p>
<div class="item" id="item1" draggable="true">Item 1</div>
<div class="item" id="item2" draggable="true">Item 2</div>
```

```
const items = document.querySelectorAll('.item');

for (let item of items) {
  // Handle dragstart event
  item.addEventListener('dragstart', (event) => {
    event.dataTransfer.setData('text/plain', event.target.id);
    console.log('Drag started: ', event.target.textContent);
  });

  // Handle dragend event
  item.addEventListener('dragend', (event) => {
    console.log('Drag ended: ', event.target.textContent);
  });
}
```

Drag the item and observe the console for events.

Item 1

Item 2

Events Drag & Drop:

- Τα παρακάτω events χρησιμεύουν για να κάνουμε drag & drop:

event	Επεξήγηση
dragover	Όταν ένα στοιχείο που γίνεται drag είναι πάνω από έναν έγκυρο στόχο drop
drop	Όταν γίνεται το drop (ενός στοιχείου που γίνεται drag σε έναν έγκυρο στόχο drop)

Σημειώσεις:

- Και τα δύο παραπάνω events έχουν default συμπεριφορά:
- Το dragover, όταν ένα στοιχείο γίνεται drag, δεν επιτρέπεται να γίνει drop και εμφανίζεται ένα εικονίδιο (απαγορευτικό)
 - Με το **preventDefault()** επί του στοιχείου που πυροδοτήθηκε το dragover, επιτρέπεται να γίνει drop στο στοιχείο.
- Το drop, έχει διαφορετικές συμπεριφορές ανάλογα με το στοιχείο που γίνεται drop:
 - π.χ. αν γίνεται drop κείμενο, μπορεί να εισαχθεί στον προορισμό, αν είναι σύνδεσμος να ακολουθηθεί ο σύνδεσμος κ.λπ.
 - Με το **preventDefault()** επί του στοιχείου που έγινε το drop, μπορούμε να προγραμματίσουμε τη δική μας, επιθυμητή συμπεριφορά.
- Από το **event.dataTransfer** ανασύρουμε με τη μέθοδο **getData(format)**, το στοιχείο (π.χ. το id) που αποθηκεύσαμε με το drop.

Παράδειγμα 2: events-drag-and-drop-2.html

```
<p>Drag the item and observe the console for events.</p>
<div class="item" id="item1" draggable="true">Item 1</div>
<div class="item" id="item2" draggable="true">Item 2</div>

<div class="drop-target" id="drop-target">Drop Here</div>
```

```
// Handle dragover event on the drop target to allow drop

const dropTarget = document.getElementById('drop-target');

dropTarget.addEventListener('dragover', (event) => {
  event.preventDefault(); // Necessary to allow dropping
  console.log('Dragging over drop target');
});

// Handle drop event to move the dragged item into the drop target
dropTarget.addEventListener('drop', (event) => {
  event.preventDefault(); // Prevent default to enable drop
  const draggedItemId = event.dataTransfer.getData('text/plain');
  const draggedItem = document.getElementById(draggedItemId);
  dropTarget.appendChild(draggedItem); // Add the item to the drop target
  console.log('Dropped:', draggedItem.textContent);
});
```

Drag the item and observe the console for events.

Item 2

Drop Here

Item 1

Item 2

Events Drag & Drop:

- Τα παρακάτω events χρησιμεύουν για να κάνουμε drag & drop:

event	Επεξήγηση
dragleave	Όταν γίνεται drag, ήταν πάνω από έναν έγκυρο drop στόχο, και φεύγει από τον έγκυρο drop στόχο.

Σημείωση:

- Το παραπάνω event χρησιμοποιείται σε συνδυασμό με το dragover για να προσθέσουμε στυλ
- Π.χ.
 - Όταν πυροδοτείται το dragover, προστίθεται μία κλάση με στυλ στο στοιχείο - drop
 - Όταν το drag φεύγει από το στοιχείο - drop, πυροδοτείται το event dragleave, οπότε μπορούμε να αφαιρέσουμε την κλάση από το στοιχείο - drop.

Παράδειγμα 3: events-drag-and-drop-3.html

```
.dragover {
  background-color: #e0ffe0;
  border-color: #008000;
}
```

```
<p>Drag the item and observe the console for events.</p>
<div class="item" id="item1" draggable="true">Item 1</div>
<div class="item" id="item2" draggable="true">Item 2</div>

<div class="drop-target" id="drop-target">Drop Here</div>
```

```
dropTarget.addEventListener('dragover', (event) => {
  event.preventDefault(); // Necessary to allow dropping
  dropTarget.classList.add('dragover'); // Highlight drop area
  console.log('Dragging over drop target');
});

// Handle drop event to move the dragged item into the drop target
dropTarget.addEventListener('drop', (event) => {
  event.preventDefault(); // Prevent default to enable drop
  const draggedItemId = event.dataTransfer.getData('text/plain');
  const draggedItem = document.getElementById(draggedItemId);
  dropTarget.appendChild(draggedItem); // Add the item to the drop target
  dropTarget.classList.remove('dragover'); // Clear highlight
  console.log('Dropped:', draggedItem.textContent);
});

// Handle dragleave event when dragged element leaves the drop target
dropTarget.addEventListener('dragleave', () => {
  dropTarget.classList.remove('dragover');
  console.log('Drag left drop target');
});
```



Events Drag & Drop:

- Τα παρακάτω events χρησιμεύουν για να κάνουμε drag & drop:

event	Επεξήγηση
drag	Πυροδοτείται (συνεχώς) όσο λαμβάνει χώρα το drag (event του dragged item)
dragenter	Πυροδοτείται τη στιγμή που το αντικείμενο dragged, εισέρχεται στον έγκυρο drop-στόχο (event του έγκυρου drop-στόχου)

Παράδειγμα 4: events-drag-and-drop-4.html

```
.dragenter {
  border-color: #0000ff;
  background-color: #d0e0ff;
}
.dragging {
  transform: scale(0.95);
  border-color: #0000ff;
  background-color: #d0e0ff;
  opacity: 0.7;
}
```



```
<p>Drag the item and observe the console for events.</p>
<div class="item" id="item1" draggable="true">Item 1</div>
<div class="item" id="item2" draggable="true">Item 2</div>

<div class="drop-target" id="drop-target">Drop Here</div>
```



```
const items = document.querySelectorAll('.item');
for (let item of items) {
  item.addEventListener('dragstart', (event) => {
    ...
    event.target.classList.add('dragging'); // Add a class to scale and fade the item
  });
  item.addEventListener('dragend', (event) => {
    ...
    event.target.classList.remove('dragging'); // Remove visual dragging effect
  });
  item.addEventListener('drag', () => {
    console.log('Dragging...');
  });
}

const dropTarget = document.getElementById('drop-target');
dropTarget.addEventListener('drop', (event) => {
  ...
  dropTarget.classList.remove('dragenter');
});
dropTarget.addEventListener('dragleave', () => {
  dropTarget.classList.remove('dragenter');
});
...
dropTarget.addEventListener('dragenter', (event) => {
  dropTarget.classList.add('dragenter'); // Change border color to indicate entry
});
...
});
```



Drag the item and observe the console for events.

Item 1
Item 2



Αλλαγή Εικόνας Drag:

- By default, η εικόνα που εμφανίζεται να γίνεται drag, είναι μία σκιά του αντικείμενου.
- Αυτό μπορούμε να το αλλάξουμε με τη μέθοδο του `dataTransfer` (του αντικειμένου event):

Μέθοδος	Επεξήγηση
<code>setDragImage</code> (element, x, y)	Το element εμφανίζεται ως σκιά όταν γίνεται drag. (x,y) είναι οι συντεταγμένες από την άνω-αριστερή κορυφή όπου θα εμφανίζεται ο cursor.

Παράδειγμα 5: drag-and-drop-dragged-item.html

```
.custom-drag-image {
  position: absolute;
  top: -9999px; /* Position off-screen */
  left: -9999px;

  padding: 5px;
  border: 1px solid #000;
  background-color: #f1f1f1;
  color: #000;
  font-size: 12px;
  pointer-events: none;
}
```

```
<div class="item" id="draggable-item" draggable="true">Item 5HTML
<!-- Custom drag image element that will replace the default shadow -->
<div id="custom-drag-image" class="custom-drag-image">Dragging...</div>
```

```
const draggableItem = document.getElementById('draggable-item');
const customDragImage = document.getElementById('custom-drag-image');

// Handle dragstart event to customize the drag image
draggableItem.addEventListener('dragstart', (event) => {
  // Use setDragImage() to replace the default shadow with a custom element
  event.dataTransfer.setDragImage(customDragImage, 10, 10);
  event.dataTransfer.setData('text/plain', event.target.id);
  console.log('Drag started with custom drag image');
});
```

Drag the item to see the custom drag image

Item 1

Dragging...

ΜΑΘΗΜΑ 2.5: Drag & Drop

6. Παράδειγμα: File Upload

Παράδειγμα 6: file-upload.html

```
<h1>Drag and Drop File Upload</h1>
<div class="drop-zone" id="dropZone">Drag an image here to upload</div>
<div class="image-preview" id="imagePreview"></div>
```



```
...
.drop-zone.dragover {
  background-color: #e0ffe0;
  border-color: #008000;
}
.image-preview {
  margin-top: 20px;
  max-width: 300px;
}
```



```
const dropZone = document.getElementById('dropZone');
const imagePreview = document.getElementById('imagePreview');
```



```
// Add dragover event listener to change appearance when a file is
dragged over
```

```
dropZone.addEventListener('dragover', (event) => {
  event.preventDefault(); // Prevent default to allow drop
  dropZone.classList.add('dragover'); // Add visual feedback
});
```

```
// Remove visual feedback when drag leaves the drop zone
```

```
dropZone.addEventListener('dragleave', () => {
  dropZone.classList.remove('dragover');
});
```

```
// Handle the drop event to process the dropped file
dropZone.addEventListener('drop', (event) => {
  event.preventDefault(); // Prevent default behavior (like opening the file)
  dropZone.classList.remove('dragover'); // Remove visual feedback
```

```
// Retrieve the files from the event
const files = event.dataTransfer.files;
if (files.length > 0) {
  const file = files[0];
  // Check if the file is an image
  if (file.type.startsWith('image/')) {
    displayImage(file);
  } else {
    alert('Please upload an image file.');
```

```
};

// Function to display the image
function displayImage(file) {
  const reader = new FileReader();
  reader.onload = function (e) {
    // Create an img element and set its src to the file content
    const img = document.createElement('img');
    img.src = e.target.result;
    img.alt = 'Uploaded Image';
    img.style.maxWidth = '100%';
    imagePreview.innerHTML = ''; // Clear any existing images
    imagePreview.appendChild(img); // Add the new image
  };
  reader.readAsDataURL(file); // Read the file as a data URL
}
```

Drag and Drop File Upload

Drag an image here to upload

