

# Programming in HTML5 with JavaScript and CSS3 Chapter 13: Drag and drop

---

Q1. How do you indicate to the browser that an element can be dragged?

- A. Set draggable attribute to true
- B. Set draggable attribute to false
- C. Set draggable attribute to auto
- D. Add the draggable attribute

Answer: A

[Page 509]

Q2. To specify to the browser that an element can be dragged, use the \_\_\_\_\_ attribute.

- A. draggable
- B. allowdrag
- C. drag
- D. data-drag

Answer: A

[Page 509]

Q3. What does it mean when you set the draggable attribute of an element to auto?

- A. The element can be dragged
- B. The element cannot be dragged
- C. The browser will decide whether the element should be draggable
- D. The element can be dragged to user desktop

Answer: C

[Page 509]

For most browsers, the auto setting is the default, which means that the browser decides whether the element should be draggable. For example, the <img> element is usually draggable by default, but a <div> element is not draggable by default]

Q4. How can you control whether the text of an element can be selected?

- A. Using css user-select property
- B. Using css display property
- C. Using JavaScript onselect event handler
- D. Using JavaScript onselect event

handler Answer: A

[Page 509]

The user-select property specifies whether the text of an element can be selected.

In web browsers, if you double-click on some text it will be selected/highlighted. This property can be used to prevent this.

Syntax

user-select: auto|none|text|all;

auto - Default. Text can be selected if the browser allows it

---

none - Prevent text selection

text - The text can be selected by the user all - Text selection is made with one click instead of a double-click Example

Prevent text selection of a <div> element:

```
div {  
  -webkit-user-select: none; /* Safari 3.1+ */  
  -moz-user-select: none; /* Firefox 2+ */  
  -ms-user-select: none; /* IE 10+ */  
  user-select: none; /* Standard syntax */  
}
```

For details <https://developer.mozilla.org/en-US/docs/Web/CSS/user-select>

Q6. Which of the following is not the event of a dragged element?

- A. dragstart
- B. drag
- C. dragend
- D. drop

Answer: D

[Page 510

drop is the event of drop target element]

Q7. Which of the following is not an event of the drop target?

- A. dragstart
- B. dragenter
- C. dragover
- D. dragleave

Answer: A

[Page 510

dragstart is the event of drag source element]

Q8. Which of the following is not an event of the dragged element?

- A. dragstart
- B. drag
- C. drop
- D. dragend

Answer: C

[Page 510]

Q9. When the drag event is triggered?

- A. Triggers when the drag is started
- B. Triggers continuously as the element is being dragged
- C. Triggers continuously as the element is dragged over the drop zone
- D. Triggers when the drag enters a drop zone

Answer: B

[Page 510]

Q10. Which of the following is not an event of the drop target element?

- 
- A. dragenter
  - B. drop
  - C. drag
  - D. dragleave

Answer: C

[Page 511]

Q11. The dragenter and dragover events default to rejecting dragged items. How can you enable dropping?

- A. By adding alldrop attribute to the drop target
- B. By cancelling the default action on these events
- C. By setting css user-select property to none
- D. By returning false from the event handler

method Answer: B

[Page 511]

Default action must be cancelled otherwise you can't drop. you should code like

```
document.getElementById('targetid')
```

```
.ondragenter = (e) =>{  
    e.preventDefault();  
}
```

```
document.getElementById('targetid')
```

```
.ondragover = (e) =>{  
    e.preventDefault();  
    return true;  
}
```

returning true is important in most browsers]

Q12. You are implementing HTML5 drag and drop. You have subscribed to dragenter event of the drop target like below

```
document.getElementById('drop-target').addEventListener('dragenter', function  
    (evt) { evt.preventDefault();  
    }, false);
```

Why do you call the preventDefault() on event object?

- A. To prevent the rejection of the dragged items
- B. To accept the dragged items
- C. To cancel event bubble
- D. None of the above

Answer: A

[Page 512]

The dragenter and dragover events default to rejecting dragged items, which is why you can't currently drop an item. You can enable dropping by cancelling the default action on these events.]

Q13. How can you pass data from the dragstart event to the drop event?

- A. Using the DataTransfer object
- B. Using a global variable
- C. Using a local variable
- D. All of the above

Answer: A

---

[Page 513]

Q14. Which property of the dataTransfer object is used to set the type of drag operation and the cursor type?

- A. dropEffect
- B. effectAllowed
- C. types
- D. cursor

Answer: A

[Page 514]

Q15. Which one is **not** a valid value for the DataTransfer object's effectAllowed property?

- A. copy
- B. move
- C. paste
- D. link

Answer: C

[Page 514]

effectAllowed – type of operation copy/move/link/none]

Q16. Using the DataTransfer object, what kind of data can you pass to the drop event?

- A. Any valid string, number, date/time, or Boolean value
- B. Any URL that is within the same domain as the webpage
- C. Any JSON object
- D. Any object that can be represented as a string or

URL Answer: D

[Page 517]

Q17. You are coding for HTML5 drag and drop in JavaScript. You have handled dragstart and drop event like below: (Line numbers are illustrative purpose only)

```
Line 1:     draggableElement.on('dragstart' function(evt){
Line 2:     var id = evt.target.id;
Line 3:     evt.dataTransfer.setData('text/plain', id);
           }); Line 4:
           dropTarget.on('drop', function(evt){ Line 5:
```

```
Line 6:
```

```
Line 7: Line 8: });    //code
```

You want to capture the data passed in dragstart event at line 2 in drop event. Which code should you add at line 6?

- A. var id = evt.dataTransfer.getData();
- B. var id = evt.dataTransfer.getData('id');
- C. var id = evt.dataTransfer.getData('plain');
- D. var id =

evt.dataTransfer.getData('text/plain');

Answer: D

[Page 515]

Q18. In which event, do you set the desired drag effect for the drag source?

- A. dragstart

- 
- B. drag
  - C. dragend
  - D. dragmove

Answer: A [Page 510]

Q19. Which of the following is **not** a property on the File object?

- A. name
- B. path
- C. type
- D. size

Answer: B  
[Page 517]

Q20. Which one is not a property of the File object?

The File object provides the name, type, and size properties

- A. name
- B. type
- C. size
- D. extension

Answer: D  
[Page 517]

Q21. You want to allow users to drop files on a division element in your web page. The html for the division is like below

```
<div id='target'>  
  <p>Drop files here...</p>  
</div>
```

You want to access the files a user dropped in drop event. You have code like below: (Line numbers are illustrative purpose only)

```
Line 1:      document.getElementById('target')  
Line 2:      .addEventListener('drop', function(evt){  
Line 3:  
Line 4:          //mode code  
Line 4:      }, false);
```

- A. var files = evt.files
- B. var files = evt.target.files
- C. var files = evt.data.files
- D. var files =

evt.dataTransfer.files Answer: D  
[Page 520]

Q22. Which object provides access to the dropped files?

- A. DataTransfer
- B. Data
- C. Target
- D. Window

Answer: A

