More on JavaScript Events | JavaScript Tutorial In Hindi #18

Overview Q&A Downloads Announcements

More on JavaScript Events | JavaScript Tutorial In Hindi #18

As we have studied before, the JavaScript's interaction with HTML is handled through events that occur when the user or the browser manipulates a page. When the user clicks a button, it is called an event. Events are a part of the Document Object Model(DOM), and every HTML element contains a set of events that can trigger the JavaScript Code.

In the previous lecture, we have studied different events like onclick, onsubmit, etc. Now let's explore more events available in JavaScript. There are three types of events, mouse events, input events, and keyboard events. Here are some important events, along with the example.

Dblclick:-

The purpose of a double click event is to triggers after two clicks on the same element within a short timeframe. This event is rarely used nowadays. Even when we use it, we should *never* use both an onclick and a dblclick event handler on the same HTML element. When the user double-clicks on an element, a one-click event takes place before the dblclick. This will cause a problem on the webpage. So, keep the clicks and dblclicks well separated to avoid complications.

```
const myclick = document.querySelector('aside');
myclick.addEventListener('dblclick', function (e) {
console.log(“Double click event occur”);});
```

Mousemove:-

The mousemove event takes place at an element when a mouse is moved while the cursor's hotspot is inside it. This event works fine, but we should know that it may take quite some system time to process all mousemove events. If we move the mouse one pixel, the mousemove event fires.

```
const test= document.getElementById('test');
test.addEventListener('mousemove', function (e) {
```

```
console.log(“ Mousemove event occur”);});
```

Mouseover and mouseout:-

For an element when a mouse or other pointing device is used to move the cursor onto the element or one of its child elements, mouseover event is fired. Whereas, for an element when a mouse or other pointing device is used to move the cursor so that it is no longer contained within the element or one of its children, the **mouseout** event is fired.

Example of Mouseover:-

```
const test= document.getElementById('test');
test.addEventListener("mouseover", function( event ) { event.target.style.color = "red";});
```

Example of Mouseout:-

```
const test= document.getElementById('test');
test.addEventListener("mouseout", function( event ) {
event.target.style.color = "red";});
```

Mouseenter and mouseleave:-

The mouseenter event is fired at an element when a mouse initially moved so that its hotspot is within the element at which the event was fired. Whereas the **mouseleave** event is fired at an element when the mouse or other pointing device is moved out of it. These events are similar, but the difference is that the mouseleave event is fired when the pointer has exited the element *and* all of its descendants. In contrast, the mouseout event is fired when the pointer leaves the element *or* leaves one of the elements' descendants even if the pointer is still within the element.

Example of Mouseenter:-

```
const mouseTarget = document.getElementById('mouseTarget');
mouseTarget.addEventListener('mouseenter', function(e) {
mouseTarget.style.border = '5px dotted blue';});
```

Example of Mouseleave:-

```
const mouseTarget = document.getElementById('mouseTarget');
mouseTarget.addEventListener('mouseleave', function(e){
mouseTarget.style.border = '1px solid red'; });
```

Mousedown and mouseup:-

The **mousedown** event is fired at an element when the pointing device like mouse button is pressed while the pointer is inside the element. Whereas, the **mouseup** event is fired at an element when a button on a pointing device is released while the pointer is located inside it.

There is a difference between click and mousedown event. As we know, that click event occurs after a full click action; that is, the mouse button is pressed and released while the pointer remains inside the same element. Whereas, the mousedown event is fired the moment the button is initially pressed.

Example of mousedown:

```
myevent.addEventListener('mousedown', function(e) {
console.log("Mousedown event occur")});
```

Example of mouseup:

```
myevent.addEventListener('mouseup', function(e) {
console.log("Mouseup event occur")});
```

Website.html code as described/written in the video

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
</head>
<body>
    <div class="container">
        <h1 id="heading"> Welcome to Code With Harry</h1>
        <div id="myfirst" class="child red good" id="first">child 1

            <ul class="this">
                <li class="childul">this</li>
                <li class="childul">is</li>
                <li class="childul">a</li>
                <li class="childul">list </li>
                <li class="childul">of my dreams</li>
            </ul>
        </div>
        <div class="child">child 2</div>
        <div class="child red">child 3</div>
        <div class="child">child 4</div>
        <form action="none.html" method="post">
            <a href="//codewithharry.com">Go to Code With Harry</a>
            <br>
            <br>
            Search this website: <input type="text" name="Hello" id="">
            <input type="button" value="submit">
        </form>
    </div>
    <br>
    <div class="no">this is a dummy div1</div>
    <div class="no">this is a dummy div2</div>
    <div class="no">this is a dummy div3</div>
</body>
<!-- <script src="js/tut12.js"></script> -->
<!-- <script src="js/tut14.js"></script> -->
<script src="js/tut15.js"></script>
</html>
```

JavaScript code as described/written in the video

```
console.log('This is tut 18');
// let btn = document.getElementById('btn');
// btn.addEventListener('click', func1);
// // btn.addEventListener('mousedown', func3);
// btn.addEventListener('dblclick', func2);

// function func1(e) {
//     console.log("Thanks", e);
//     e.preventDefault();
// }

// function func2(e) {
//     console.log("Thanks its a double click", e);
//     e.preventDefault();
// }
// function func3(e) {
//     console.log("Thanks its a mouse down ", e);
//     e.preventDefault();
// }
// document.querySelector('.no').addEventListener('mouseenter', function(){


//     console.log('You entered no')
// })

// document.querySelector('.no').addEventListener('mouseleave', function(){
//     console.log('You exited no')
// })

document.querySelector('.container').addEventListener('mousemove', function(e){
    console.log(e.offsetX, e.offsetY);
    document.body.style.backgroundColor = `rgb(${e.offsetX}, ${e.offsetX},154)`;
    console.log('You triggered mouse move event')
})
```

Previous

Next