**JS EVENTS**

Example 1

window.onload=function(){

const btn = document.querySelector('button');

function random(number) {

return Math.floor(Math.random() \* (number+1));

}

btn.addEventListener('click', () => {

const rndCol = `rgb(${random(255)}, ${random(255)}, ${random(255)})`;

document.body.style.backgroundColor = rndCol;

});

}

*NB: To add code inside a template literal string, use the dollar sign followed by curly brackets (****${}****) and place the code inside the curly brackets.*

**Example 2**

<script>

function countRabbits() {

for(let i=1; i<=3; i++) {

alert("Rabbit number " + i);

}

}

</script>

<input type="button" onclick="countRabbits()" value="Count rabbits!">

Example 3

<input id="elem" type="button" value="Click me"/>

<script>

function handler1() {

alert('Thanks!');

};

function handler2() {

alert('Thanks again!');

}

elem.onclick = () => alert("Hello");

elem.addEventListener("click", handler1); // Thanks!

elem.addEventListener("click", handler2); // Thanks again!

</script>

Example 4

<input type="button" value="Click me" id="elem">

<script>

elem.onclick = function(event) {

// show event type, element and coordinates of the click

alert(event.type + " at " + event.currentTarget);

alert("Coordinates: " + event.clientX + ":" + event.clientY);

};

</script>

Example 5

<input id="textBox" type="text" />

<div id="output"></div>

const textBox = document.querySelector("#textBox");

const output = document.querySelector("#output");

textBox.addEventListener('keydown', (event) => output.textContent = `You pressed "${event.key}".`);

Example 6

<form>

<div>

<label for="fname">First name: </label>

<input id="fname" type="text" />

</div>

<div>

<label for="lname">Last name: </label>

<input id="lname" type="text" />

</div>

<div>

<input id="submit" type="submit" />

</div>

</form>

<p></p>

Now some JavaScript — here we implement a very simple check inside a handler for the [submit](https://developer.mozilla.org/en-US/docs/Web/API/HTMLFormElement/submit_event) event (the submit event is fired on a form when it is submitted) that tests whether the text fields are empty. If they are, we call the [preventDefault()](https://developer.mozilla.org/en-US/docs/Web/API/Event/preventDefault) function on the event object — which stops the form submission — and then display an error message in the paragraph below our form to tell the user what's wrong:

const form = document.querySelector('form');

const fname = document.getElementById('fname');

const lname = document.getElementById('lname');

const para = document.querySelector('p');

form.addEventListener('submit', (e) => {

if (fname.value === '' || lname.value === '') {

e.preventDefault();

para.textContent = 'You need to fill in both names!';

}

});

Example 7

const output = document.querySelector('#output');

function handleClick(e) {

output.textContent += `You clicked on a ${e.currentTarget.tagName} element\n`;

}

const container = document.querySelector('#container');

container.addEventListener('click', handleClick);

“e” is **a short variable reference to an event object provided to the event handlers**. The event object generally offers certain useful methods and properties that the event handlers can utilize.

Example 8

The HTML looks like this:

<button>Display video</button>

<div class="hidden">

<video>

<source

src="https://interactive-examples.mdn.mozilla.net/media/cc0-videos/flower.webm"

type="video/webm" />

<p>

Your browser doesn't support HTML video. Here is a

<a href="rabbit320.mp4">link to the video</a> instead.

</p>

</video>

</div>

It includes:

* a <button> element
* a <div> element which initially has a class="hidden" attribute
* a <video> element nested inside the <div> element.

We're using CSS to hide elements with the "hidden" class set.

The JavaScript looks like this:

const btn = document.querySelector('button');

const box = document.querySelector('div');

const video = document.querySelector('video');

btn.addEventListener('click', () => box.classList.remove('hidden'));

video.addEventListener('click', (event) => {

event.stopPropagation();

video.play();

});

box.addEventListener('click', () => box.classList.add('hidden'));

This adds three 'click' event listeners:

* one on the <button>, which shows the <div> that contains the <video>
* one on the <video>, which starts playing the video
* one on the <div>, which hides the video

Joke genertor

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <script src="functions.js" defer></script>

    <style>

        .hidden{

            display: none;

        }

    </style>

</head>

<body>

    <h1>Random Joke Generator</h1>

<button id="generateJoke">Generate Joke</button>

<p id="jokeDisplay"></p>

    <script>

        document.addEventListener('DOMContentLoaded', function() {

          const jokes = [

            "Why don't scientists trust atoms? Because they make up everything!",

            "What do you call fake spaghetti? An impasta!",

            "Why did the bicycle fall over? Because it was two-tired!",

            "What did one wall say to the other wall? I'll meet you at the corner!",

            "Why did the scarecrow win an award? Because he was outstanding in his field!",

          ];

          const generateJokeButton = document.getElementById('generateJoke');

          const jokeDisplay = document.getElementById('jokeDisplay');

          generateJokeButton.addEventListener('click', function() {

            const randomIndex = Math.floor(Math.random() \* jokes.length);

            const randomJoke = jokes[randomIndex];

            jokeDisplay.textContent = randomJoke;

          });

        });

      </script>

</body>

</html>