

HTML DOM

https://www.w3schools.com/js/js_htmldom.asp

2025 Fall
w3schools

What is the HTML DOM?

The HTML DOM is a standard **object** model and **programming interface** for HTML. It defines:

- The HTML elements as **objects**
- The **properties** of all HTML elements
- The **methods** to access all HTML elements
- The **events** for all HTML elements

In other words: **The HTML DOM is a standard for how to get, change, add, or delete HTML elements.**

JavaScript Form Validation

HTML form validation can be done by JavaScript.

If a form field (fname) is empty, this function alerts a message, and returns false, to prevent the form from being submitted:

```
function validateForm() {  
    let x = document.forms["myForm"]["fname"].value;  
    if (x == "") {  
        alert("Name must be filled out");  
        return false;  
    }  
}
```

```
<!DOCTYPE html>
<html>
<head>
<script>
function validateForm() {
    let x = document.forms["myForm"]["fname"].value;
    if (x == "") {
        alert("Name must be filled out");
        return false;
    }
}
</script>
</head>
<body>

<h2>JavaScript Validation</h2>

<form name="myForm" action="/action_page.php" onsubmit="return validateForm()" method="post">
    Name: <input type="text" name="fname">
    <input type="submit" value="Submit">
</form>

</body>
</html>
```

JavaScript Validation

Name: Submit

www.w3schools.com 내용:

Name must be filled out

확인

JavaScript Can Validate Numeric Input

JavaScript is often used to validate numeric input:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Validation</h2>

<p>Please input a number between 1 and 10:</p>

<input id="numb">

<button type="button" onclick="myFunction()">Submit</button>

<p id="demo"></p>
```

https://www.w3schools.com/js/tryit.asp?filename=tryjs_validation_number

```
<script>
function myFunction() {
    // Get the value of the input field with id="numb"
    let x = document.getElementById("numb").value;
    // If x is Not a Number or less than one or greater than 10
    let text;
    if (isNaN(x) || x < 1 || x > 10) {
        text = "Input not valid";
    } else {
        text = "Input OK";
    }
    document.getElementById("demo").innerHTML = text;
}
</script>
```

JavaScript Validation

Please input a number between 1 and 10:

JavaScript Validation

Please input a number between 1 and 10:

Input not valid

Automatic HTML Form Validation

HTML form validation can be performed automatically by the browser:

If a form field (fname) is empty, the `required` attribute prevents this form from being submitted:

https://www.w3schools.com/js/tryit.asp?filename=tryjs_validation_html

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Validation</h2>

<form action="/action_page.php" method="post">
  <input type="text" name="fname" required>
  <input type="submit" value="Submit">
</form>

<p>If you click submit, without filling out the text field,  
your browser will display an error message.</p>
```

JavaScript Validation

 Submit

If you click submit, without filling out the text field, your browser will display an error message.



The HTML DOM allows JavaScript to change the style of HTML elements.

https://www.w3schools.com/js/tryit.asp?filename=tryjs_change_style

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML DOM</h2>
<p>Changing the HTML style:</p>

<p id="p1">Hello World!</p>
<p id="p2">Hello World!</p>

<script>
document.getElementById("p2").style.color = "blue";
document.getElementById("p2").style.fontFamily = "Arial";
document.getElementById("p2").style.fontSize = "larger";
</script>
```

JavaScript HTML DOM

Changing the HTML style:

Hello World!

Hello World!

https://www.w3schools.com/js/tryit.asp?filename=tryjs_dom_color2

```
<!DOCTYPE html>
<html>
<body>

<h1 id="id1">My Heading 1</h1>

<button type="button"
onclick="document.getElementById('id1').style.color = 'red'">
Click Me!</button>

</body>
</html>
```

My Heading 1

Click Me!

Using Events

The HTML DOM allows you to execute code when an event occurs.

My Heading 1

Click Me!

JavaScript HTML DOM Animation

Create an Animation Container

All animations should be relative to a container element.

Example

```
<div id = "container">
  <div id = "animate">My animation will go here</div>
</div>
```

Style the Elements

The container element should be created with style = "position: relative".

The animation element should be created with style = "position: absolute".

https://www.w3schools.com/js/tryit.asp?filename=tryjs_dom_animate_2

```
<!Doctype html>
<html>
<style>
#container {
    width: 400px;
    height: 400px;
    position: relative;
    background: yellow;
}
#animate {
    width: 50px;
    height: 50px;
    position: absolute;
    background: red;
}
</style>
<body>

<h2>My First JavaScript Animation</h2>

<div id="container">
<div id="animate"></div>
</div>
```

My First JavaScript Animation



Animation Code

JavaScript animations are done by programming gradual changes in an element's style.

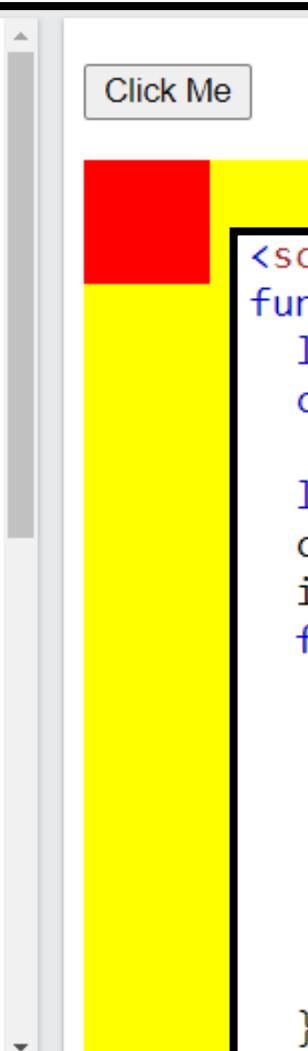
The changes are called by a timer. When the timer interval is small, the animation looks continuous.

https://www.w3schools.com/js/tryit.asp?filename=tryjs_dom_animate_3

```
<!DOCTYPE html>
<html>
<style>
#container {
  width: 400px;
  height: 400px;
  position: relative;
  background: yellow;
}
#animate {
  width: 50px;
  height: 50px;
  position: absolute;
  background-color: red;
}
</style>
<body>

<p><button onclick="myMove()">Click Me</button></p>

<div id ="container">
  <div id ="animate"></div>
</div>
```



```
<script>
function myMove() {
  let id = null;
  const elem = document.getElementById("animate");

  let pos = 0;
  clearInterval(id);
  id = setInterval(frame, 5);
  function frame() {
    if (pos == 350) {
      clearInterval(id);
    } else {
      pos++;
      elem.style.top = pos + "px";
      elem.style.left = pos + "px";
    }
  }
}
</script>
```

Reacting to Events

A JavaScript can be executed when an event occurs, like when a user clicks on an HTML element.

To execute code when a user clicks on an element, add JavaScript code to an HTML event attribute:

```
onclick=JavaScript
```

Examples of HTML events:

- When a user clicks the mouse
- When a web page has loaded
- When an image has been loaded
- When the mouse moves over an element
- When an input field is changed
- When an HTML form is submitted
- When a user strokes a key

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML Events</h2>
<h2 onclick="this.innerHTML='Ooops!'">Click on this text!</h2>

</body>
</html>
```

JavaScript HTML Events

Click on this text!

https://www.w3schools.com/js/tryit.asp?filename=tryjs_event_onclick3

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML Events</h2>
<h2 onclick="changeText(this)">Click on this text!</h2>

<script>
function changeText(id) {
  id.innerHTML = "Ooops!";
}
</script>
```

JavaScript HTML Events

Click on this text!

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML Events</h2>
<p>Click the button to display the date.</p>

<button onclick="displayDate()">The time is?</button>

<script>
function displayDate() {
    document.getElementById("demo").innerHTML = Date();
}
</script>

<p id="demo"></p>
```

JavaScript HTML Events

Click the button to display the date.

The time is?

Assign Events Using the HTML DOM

The HTML DOM allows you to assign events to HTML elements using JavaScript:

https://www.w3schools.com/js/tryit.asp?filename=tryjs_events2

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML Events</h2>
<p>Click "Try it" to execute the displayDate() function.</p>

<button id="myBtn">Try it</button>

<p id="demo"></p>

<script>
document.getElementById("myBtn").onclick = displayDate;

function displayDate() {
  document.getElementById("demo").innerHTML = Date();
}
</script>
```

JavaScript HTML Events

Click "Try it" to execute the displayDate() function.

Try it

The onload and onunload Events

The `onload` and `onunload` events are triggered when the user enters or leaves the page.

The `onload` event can be used to check the visitor's browser type and browser version, and load the proper version of the web page based on the information.

The `onload` and `onunload` events can be used to deal with cookies.

https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_onload

```
<!DOCTYPE html>
<html>
<body onload="checkCookies()>

<h2>JavaScript HTML Events</h2>

<p id="demo"></p>

<script>
function checkCookies() {
    var text = "";
    if (navigator.cookieEnabled == true) {
        text = "Cookies are enabled.";
    } else {
        text = "Cookies are not enabled.";
    }
    document.getElementById("demo").innerHTML = text;
}
</script>
```

JavaScript HTML Events

Cookies are enabled.

The onchange Event

The `onchange` event is often used in combination with validation of input fields.

Below is an example of how to use the onchange. The `toUpperCase()` function will be called when a user changes the content of an input field.

https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_onchange

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML Events</h2>
Enter your name: <input type="text" id="fname"
onchange="upperCase()">
<p>When you leave the input field, a function is triggered which
transforms the input text to upper case.</p>

<script>
function upperCase() {
  const x = document.getElementById("fname");
  x.value = x.value.toUpperCase();
}
</script>
```

JavaScript HTML Events

Enter your name:

When you leave the input field, a function is triggered which transforms the input text to upper case.

```
<h1>JavaScript HTML Events</h1>
<h2>The onchange Attribute</h2>

Enter your name: <input type="text" id="fname"
onchange="toUpperCase()">
<p>When you leave the input field, a function transforms
the input to upper case.</p>

<script>
function upperCase() {
  const x = document.getElementById("fname");
  x.value = x.value.toUpperCase();
}

</script>

</body>
</html>
```

JavaScript HTML Events

The onchange Attribute

Enter your name:

When you leave the input field, a function transforms
to upper case.

```
<h1>JavaScript HTML Events</h1>
<h2>The onchange Attribute</h2>

Enter your name: <input type="text" id="fname"
onchange="toUpperCase()">
<p>When you leave the input field, a function transforms
the input to upper case.</p>

<script>
function upperCase() {
  const x = document.getElementById("fname");
  x.style.backgroundColor = "yellow";
  x.value = x.value.toUpperCase();
}

</script>

</body>
</html>
```

JavaScript HTML Events

The onchange Attribute

Enter your name: DF

When you leave the input field, a function transforms
to upper case.

The onmouseover and onmouseout Events

The `onmouseover` and `onmouseout` events can be used to trigger a function when the user mouses over, or out of, an HTML element:

https://www.w3schools.com/js/tryit.asp?filename=tryjs_events_mouseover

```
<!DOCTYPE html>
<html>
<body>

<div onmouseover="mOver(this)" onmouseout="mOut(this)"
style="background-
color:#D94A38;width:120px;height:20px;padding:40px;">
Mouse Over Me</div>

<script>
function mOver(obj) {
  obj.innerHTML = "Thank You"
}

function mOut(obj) {
  obj.innerHTML = "Mouse Over Me"
}
</script>
```



```
<div onmouseover="mOver(this)" onmouseout="mOut(this)"  
style="background-color:#D94A38;width:120px;height:20px;padding:40px;">  
Mouse Over Me</div>  
<script>  
function mOver(obj) {  
    obj.innerHTML = "Thank You"  
}  
function mOut(obj) {  
    obj.innerHTML = "Mouse Over Me"  
}
```