

Sets the value of an attribute on the specified element. If the attribute already exists, the value is updated; otherwise a new attribute is added with the specified name and value.

To get the current value of an attribute, use `getAttribute()`; to remove an attribute, call `removeAttribute()`.

Syntax

```
Element.setAttribute(name, value);
```

Parameters

name

A `DOMString` specifying the name of the attribute whose value is to be set. The attribute name is automatically converted to all lower-case when `setAttribute()` is called on an HTML element in an HTML document.

value

A `DOMString` containing the value to assign to the attribute. Any non-string value specified is converted automatically into a string.

Boolean attributes are considered to be `true` if they're present on the element at all, regardless of their actual value; as a rule, you should specify the empty string (`"`) in `value` (some people use the attribute's name; this works but is non-standard). See the [example](#) below for a practical demonstration.

Since the specified `value` gets converted into a string, specifying `null` doesn't necessarily do what you expect. Instead of removing the attribute or setting its value to be `null`, it instead sets the attribute's value to the string `"null"`. If you wish to remove an attribute, call `removeAttribute()`.

Return value

undefined.

Exceptions

InvalidCharacterError

The specified attribute name contains one or more characters which are not valid in attribute names.

Example

In the following example, `setAttribute()` is used to set attributes on a `<button>`.

HTML

```
<button>Hello World</button>
```

JavaScript

```
var b = document.querySelector("button");

b.setAttribute("name", "helloButton");
b.setAttribute("disabled", "");
```

This demonstrates two things:

- The first call to `setAttribute()` above shows changing the `name` attribute's value to `"helloButton"`. You can see this using your browser's page inspector (Chrome, Edge, Firefox, Safari).
- To set the value of a Boolean attribute, such as `disabled`, you can specify any value. An empty string or the name of the attribute are recommended values. All that matters is that if the attribute is present at all, *regardless of its actual value*, its value is considered to be `true`. The absence of the attribute means its value is `false`. By setting the value of the `disabled` attribute to the empty string (`""`), we are setting `disabled` to `true`, which results in the button being disabled.