

Update Element

- 1. Element.attributes
- 2. HTMLElement.dataset
- 3. Element.classList
- 4. HTMLElement.style

1. Element.attributes

Common attributes

#	Name	Desc
1	Element.id	the id of the element
2	Element.className	get class string
3	HTMLElement.hidden	indicating the element is hidden or not
4	HTMLElement.title	text appears in a popup box when hover
5

Element specific attribute

- HTMLAnchorElement `<a>`: href, rel, target, ...
- HTMLImageElement ``: src, alt, ...

Custom attributes

#	Name	Desc
1	Element.getAttribute(name)	get value of an attribute by name
2	Element.hasAttribute(name)	check if an attribute is existed
3	Element.setAttribute(name, value)	set value for attribute name
4	Element.removeAttribute(name)	remove an attribute by name
5	Element.toggleAttribute(name)	toggle an attribute
6	Element.attributes()	live collection of all attribute nodes

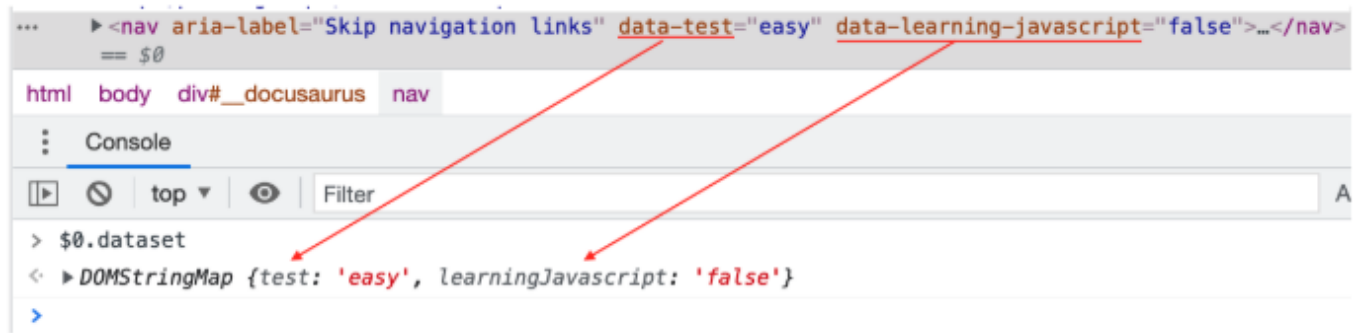
```
const title = document.getElementById('title');
if (title) {
  title.setAttribute('learn', 'javascript');

  title.hasAttribute('learn'); // true
  title.getAttribute('learn'); // javascript
}
```

Source: <https://javascript.info/dom-attributes-and-properties#property-attribute-synchronization>

2. HTMLElement.dataset

- HTMLElement.dataset (read-only) provide read/write access to custom data attributes (data-*)
- In HTML, kebab-case
- In Javascript, camelCase



```
<nav id="topnav" data-test="easy" data-learning-javascript="false">
  <!-- ... -->
</nav>
```

```
const nav = document.getElementById('topnav');
if (nav) {
  // get data attributes
  nav.dataset.test; // easy
  nav.dataset.learningJavaScript; // false
  // set data attribute (it will convert to string)
  nav.dataset.test = 'hard';
  nav.dataset.learningJavaScript = true;
}
```

3. Element.classList

#	Name	Desc
1	Element.classList (read-only)	return a live DOMTokenList collection of the class
2	Element.classList.add(class1, class2, ...)	add class
3	Element.classList.remove(class1, class2, ...)	remove class
4	Element.classList.replace(oldClass, newClass)	replace oldClass with newClass
5	Element.classList.toggle(class)	toggle class

```
function handleToggleClick() {
  const accordion = document.getElementById('accordion');
```

```
    if (!accordion) return;
    accordion.classList.toggle('active');
  }
```

4. HTMLElement.style

#	Name	Desc
1	HTMLElement.style.width	get / set CSS width
2	HTMLElement.style.height	get / set CSS height
3	HTMLElement.style.color	get / set CSS color
4	HTMLElement.style.backgroundColor	get / set CSS background-color
5	HTMLElement.style.fontSize	get / set CSS font-size
6	HTMLElement.style.fontWeight	get / set CSS font-weight

Rule: transform kebab-case to camelCase to use as property in style object.

```
const title = document.getElementById('title');
if (title) {
  // set style
  title.style.color = 'red';
  title.style.fontSize = '24px';
  title.style.fontWeight = 'bold';
  // get style
  const computedStyle = getComputedStyle(title);
  computedStyle.color; // rgb(255, 0, 0)
}
```

Tham khảo

- <https://developer.mozilla.org/en-US/docs/Web/API/Element/attributes>
- <https://developer.mozilla.org/en-US/docs/Web/API/HTMLElement/dataset>
- <https://developer.mozilla.org/en-US/docs/Web/API/Element/classList>
- <https://developer.mozilla.org/en-US/docs/Web/API/HTMLElement/style>