Obtaining Information about Elements

The DOM can also provide information on the items you've just selected. Let's find out how!

We'll cover the following

- HTML Content
- Textual Content
- Attributes
- Classes

HTML Content

The `innerHTML` property will retrieve the HTML content of your DOM element.

Output

JavaScript

HTML

```
// The HTML content of the DOM element with ID "content"
console.log(document.getElementById("content").innerHTML);

```

Console

Clear

Textual Content

The `textContent` property returns all the text content of a DOM element, without any HTML markup.

Output

JavaScript

HTML

```
// The textual content of the DOM element with ID "content"
console.log(document.getElementById("content").textContent);
```

Console

Clear

Attributes

The `getAttribute()` method can be applied to a DOM element and will return the value of a given attribute.

Output

JavaScript

HTML

```
// Show href attribute of the first link
console.log(document.querySelector("a").getAttribute("href"));
```

Console

Clear

Some attributes are directly accessible as properties. This is true for the `id`

Some attributes are directly accessible as properties. This is true for the `id`, `href`, and `value` attributes.

Output

JavaScript

HTML

```
// Show ID attribute of the first list
console.log(document.querySelector("ul").id);

// Show href attribute of the first link
console.log(document.querySelector("a").href);
```

Console

Clear

You can check for the existence of an attribute using the `hasAttribute()` method as seen in the example below.

Output

JavaScript

HTML

```
if (document.querySelector("a").hasAttribute("target")) {
  console.log("The first link has a target attribute.");
} else {
  console.log("The first link does not have a target attribute."); // Will be shown
}
```

Console

Clear

Classes

In a web page, a tag can have multiple classes. The `classList` property retrieves a DOM element’s list of classes.

retrieves a DOM element's list of classes.

Output

JavaScript

HTML

```
// List of classes of the element identified by "ancient"
const classes = document.getElementById("ancient").classList;
console.log(classes.length); // 1 (since the element only has one class)
console.log(classes[0]);     // "wonders"
```

Console

⊘ Clear

You also have the opportunity to test the presence of a class on an element by calling the `contains()` on the class list, passing the class to test as a parameter.

Output

JavaScript

HTML

```
// Check whether "ancient" element contains the "wonders" class
console.log(document.getElementById("ancient").classList.contains("wonders"));
```

Console

⊘ Clear

This is only a part of the DOM traversal API. For more details, check the Mozilla Developer Network.