**DOM Selectors. Modifying Elements**

In the DOM, all HTML elements are defined as **objects**. The document object is the owner of all other objects in your web page.



As a result, we can access/change/add/delete HTML elements through DOM methods and properties.

1. **Finding HTML Elements**

|  |  |
| --- | --- |
| **Methods** | **Description** |
| document.getElementById(*id*) | Find an element by element id |
| document.getElementsByTagName(*tagname*) | Find elements by tag name (returns live HTML collection) |
| document.getElementsByClassName(class*name*) | Find elements by class name (returns live HTML collection) |
| document.querySelector(‘*selectors’*) | Returns the **first element** within the document that matches the selector, or group of selectors |
| document.querySelectorAll(‘*selectors’*) | Returns **all elements** (in **static** [Nodelist](https://developer.mozilla.org/en-US/docs/Web/API/Document/querySelectorAll#:~:text=static%20(not%20live)%20NodeList)) that matches the selector, or group of selectors  We can run array functions on Nodelist |
| document.forms[*form\_id*] | [Collect elements within the form](https://www.w3schools.com/js/tryit.asp?filename=tryjs_dom_form_elements) |

To work on an element, we first need to find (access) it.

We always start with accessing the document object. Then, we select the element through id, tagname, classname, CSS selector, HTML object collections, etc. A more detailed list methods to find HTML elements can be accessed [here](https://developer.mozilla.org/en-US/docs/Web/API/Document/querySelectorAll#:~:text=xmlVersion-,Methods,-Events).

1. **Changing HTML Elements**

After having access to the element, we can use properties/methods to change it. We can change the inner HTML value, attribute value, or style of the element through its [properties and/or methods](https://developer.mozilla.org/en-US/docs/Web/API/Element).

|  |  |
| --- | --- |
| **Properties** | **Description** |
| *element*.innerHTML =  *new html content* | Change the inner HTML of an element |
| *element*.*attribute = new value* | Change the attribute value of an HTML element |
| *element*.style.*property = new property value* | Change the style of an HTML element |
| **Methods** | **Description** |
| *element*.setAttribute*(attribute, value)* | Change the attribute value of an HTML element |

*element*.textContent 🡪 All the text within the element, disregarding style.

*element*.innerText 🡪 Text within the element, but paying attention to style (will not show display:none content)

Graphical user interface, text, application, website

Description automatically generated*element*.innerHTML 🡪 Does not change the element itself. It replaces the inner content of the element with the new nested HTML content.

🡪

**Note:** for style properties, we must use HTML form, not CSS form. e.g., instead of border-bottom, we must use borderBottom.

Note: while using querySelector, we can get the last item/nth item through indexing querySelectorAll. However, we can also do ‘<selector> : last-child’ or ‘<selector> : nth-child (index)’