

Getting elements

<code>document.getElementById(id)</code>	Gets a single element based on its id attribute.
<code>document.querySelector(cssSelector):</code>	Gets a single element based on a CSS selector. If multiple **elements match the selector, returns the first one.
<code>document.querySelectorAll(cssSelector)</code>	Gets all elements matching a CSS selector as a NodeList.
<code>document.getElementsByTagName(tag-Name)</code>	Gets all elements with a specific HTML tag as an HTMLCollection.
<code>document.getElementsByClassName(className)</code>	Gets all elements with a specific class as an HTMLCollection.

Sizes and Scrolling

<code>window.innerWidth</code>	The width of the browser window
<code>window.innerHeight</code>	The height of the browser window
<code>window.getComputedStyle(el)</code>	Gets styles as they are currently rendered on the page, converted to pixels.
<code>el.clientHeight</code>	The height of visible content and padding
<code>el.offsetHeight</code>	The height of visible content, padding, borders and scrollbars
<code>el.scrollHeight</code>	The height of all content and padding, including content scrolled out of view
<code>el.offsetTop</code>	The distance from the outer top border of the element to the inner top border of the nearest positioned parent
<code>el.scrollIntoView()</code>	Scrolls the container so the element is in view
<code>el.scrollTo(optionsObj)</code>	Scrolls the element to a specified top value in the options object. Additionally, behavior: 'smooth' will create a smooth transition.

Adding And Removing Elements

<code>document.createElement(tag)</code>	Creates a new HTML element.
<code>document.createTextNode(text)</code>	Creates a text node as an alternative to setting textContent.
<code>document.createDocumentFragment()</code>	Creates a document fragment, which is useful for appending multiple elements at once after a loop.
<code>el.appendChild(el)</code>	Appends an element to the end of the contents of another element.
<code>el.append(node1, ...)</code>	Appends 1 or more nodes (elements or text) to the end of the contents of another element. * <code>element.prepend(node1, node2, ...)</code> : Prepends 1 or more nodes (elements or text) to the beginning of the contents of another element.
<code>el.remove()</code>	Removes the element from the DOM



By **aprox**
cheatography.com/aprox/

Not published yet.
Last updated 14th July, 2022.
Page 1 of 2.

Sponsored by **CrosswordCheats.com**
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>

Setting Attributes

<code>el.style.property</code>	Sets a CSS property using inline styles, although CSS classes should usually be preferred. The style object will only contain inline styles, not those set with CSS.
<code>el.setAttribute('attribute', 'val')</code>	Sets an HTML attribute to a specific value.
<code>el.textContent</code>	The text content of an element, including that of any children. Note: this is slightly different from <code>element.innerText</code> , which only gets text that is actually rendered and <code>element.innerHTML</code> which gets the entire HTML code as a string.
<code>el.attribute</code>	An alternative to the <code>setAttribute</code> function, attributes can be directly edited via their property name. For example, <code>element.value</code> would get the value attribute of the element.
<code>el.classList</code>	An object for updating CSS classes. Specifically, this contains 3 primary functions: <code>add(className)</code> , <code>remove(className)</code> - and <code>toggle(className)</code> .



By **aprox**
cheatography.com/aprox/

Not published yet.
Last updated 14th July, 2022.
Page 2 of 2.

Sponsored by **CrosswordCheats.com**
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>