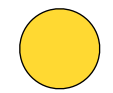


IMPORTANT DOM VOCABULARY

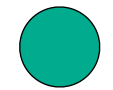
DOM Node

Every element, attribute, and piece of text in the HTML is represented by its own DOM node



Document Node

When you access any element, attribute, or text node - you navigate to it via the document node (the starting point for all visits in the DOM tree)



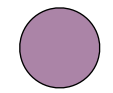
Element Nodes

To access the DOM tree, you start by looking for elements - once you find the element you want *then* you can access its text and attribute nodes, should you wish



Attribute Nodes

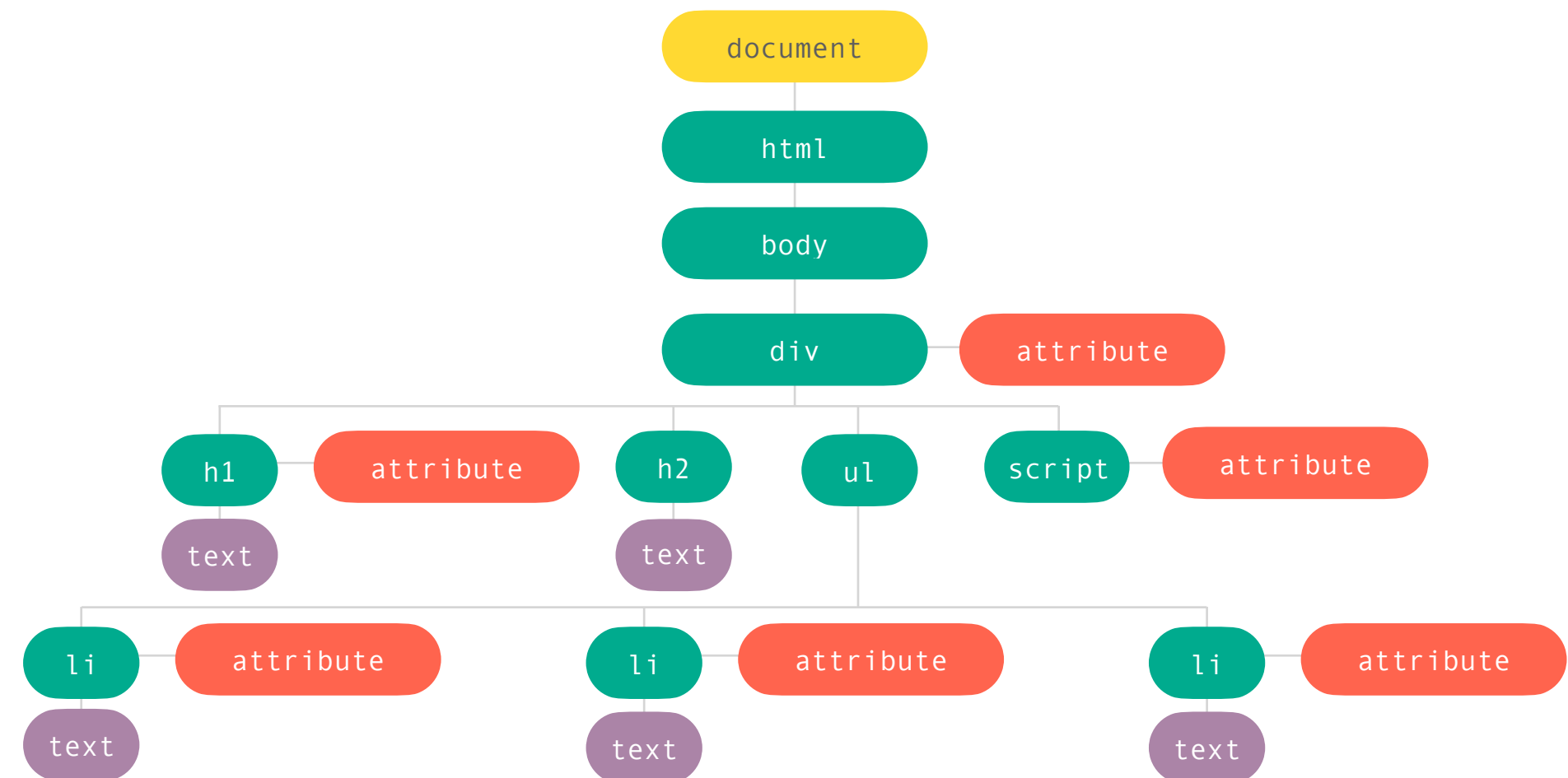
Attribute nodes are part of the element that carries them, not children of that element - once you access an element there are specific JavaScript methods and properties to read or change that element's attributes (example: change the value of class attributes to trigger new CSS rules)



Text Nodes

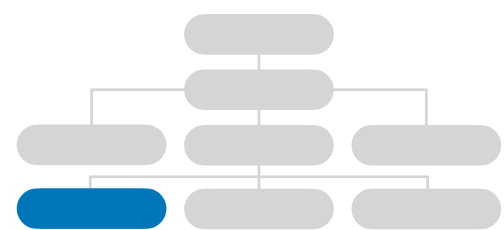
Once you access an element node, you can then reach the text within that element (stored within its own text node) - text nodes cannot have children, they're always a new branch of the DOM tree and no further branches can come off of it

SAMPLE DOM TREE



STEP 1 ACCESSING ELEMENTS

SELECT INDIVIDUAL ELEMENT NODE



getElementById()

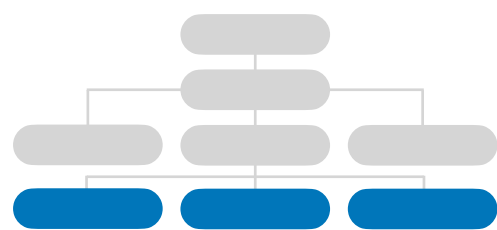
Uses the value of an element's id attribute (which should be unique within the page)

querySelector()

Uses CSS selector and returns the first matching element

You can also select individual elements by **traversing from one element to another** within the DOM tree (refer to third column)

SELECT MULTIPLE ELEMENTS (NODELIST)



getElementByClassName()

Selects all elements that have a specific value for their class attribute

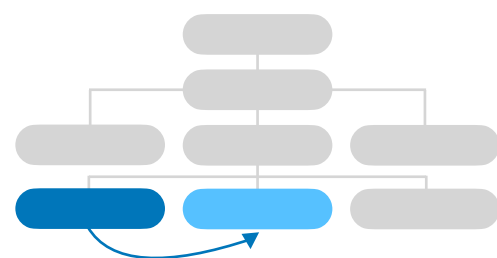
getElementsByTagName()

Selects all elements that have the specified tag name

querySelectorAll()

Uses a CSS selector to select all matching elements

TRAVERSING BETWEEN ELEMENT NODES



parentNode

Selects the parent of the current element node (which will return only one element)

previousSibling / nextSibling

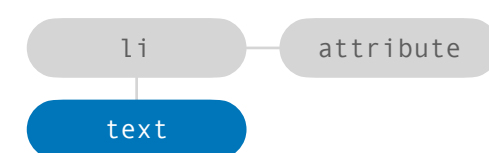
Selects the previous or next sibling from the DOM tree

firstChild / lastChild

Selects the first or last child of the current element

STEP 2 WORKING WITH ELEMENTS

ACCESS / UPDATE TEXT NODES



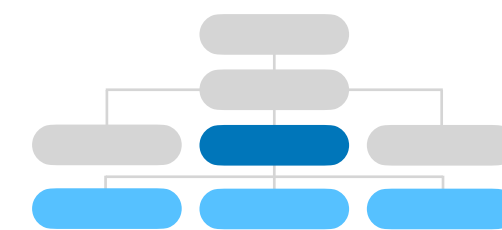
The text inside any element is stored inside a **text node**. To access the text node above:

1. Select the `` element
2. Use the **firstChild** property to get the text node
3. Use the text node's only property, **nodeValue**, to get text from the element

nodeValue

This property lets you access or update contents of a text node (does not include text in child elements)

WORK WITH HTML CONTENT



innerHTML

Allows access to the child elements and text content

textContent

Allows access to just the text content

createElement()
createTextNode()
appendChild()
removeChild()

These methods let you create new nodes, add nodes to a tree, and remove nodes from a tree - called **DOM manipulation**

ACCESS / UPDATE ATTRIBUTE VALUES



className / id

Allows you to get or update the value of the class and id attributes

hasAttribute()
getAttribute()
setAttribute()
removeAttribute()

1. The first checks if an attribute exists
2. The second gets its value
3. The third updates the value
4. The fourth removes the attribute

