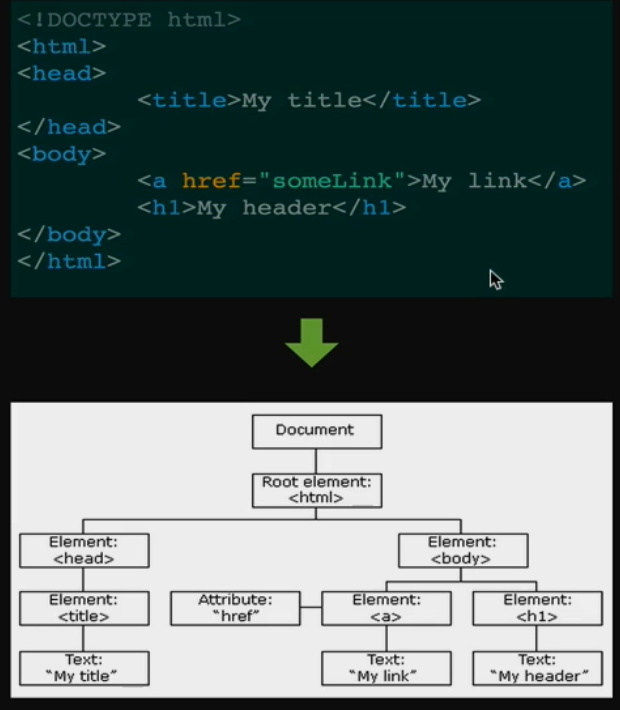
1.Introduction

**DOM = Document Object Model**

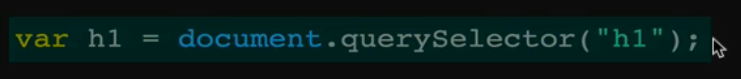
* The browser turns every *HTML* tags into a *JavaScript* object that we can manipulate. Window is the topmost document and it is provided to us by default.



* Inside each object (h1 for example) it has different attribute (color, type of tag, text size etc.)
* Document is the top-level object that everything inside is contained.

2.Select and Manipulate

* Select: For example, we want to select a h1 header.

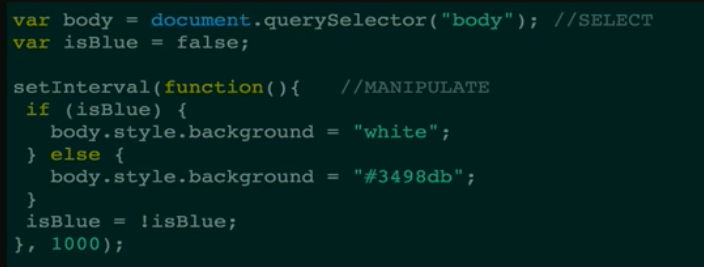


This will store the h1 object in the h1 variable.

* Manipulate: Now that we have the object in h1 variable, we can do all sorts of things. For example, changing the color. Here h1 is the object variable, style is another object which is part of the h1 object and color is one of the variables of style object.

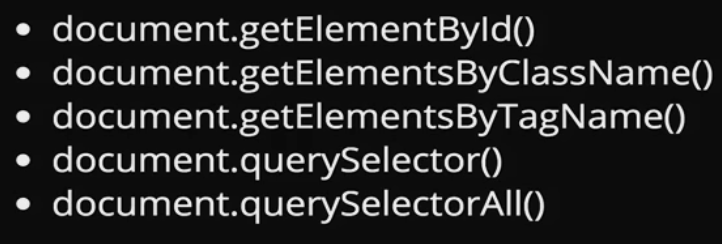


* A simple animation function: Check the conditional logic a bit, its nice trick



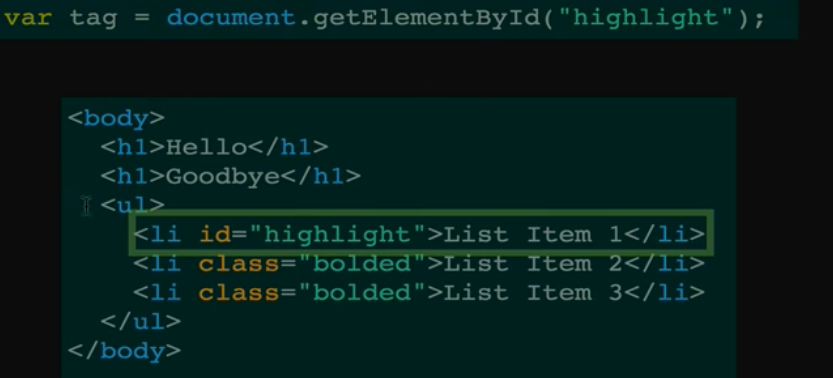
3.Important Select Methods

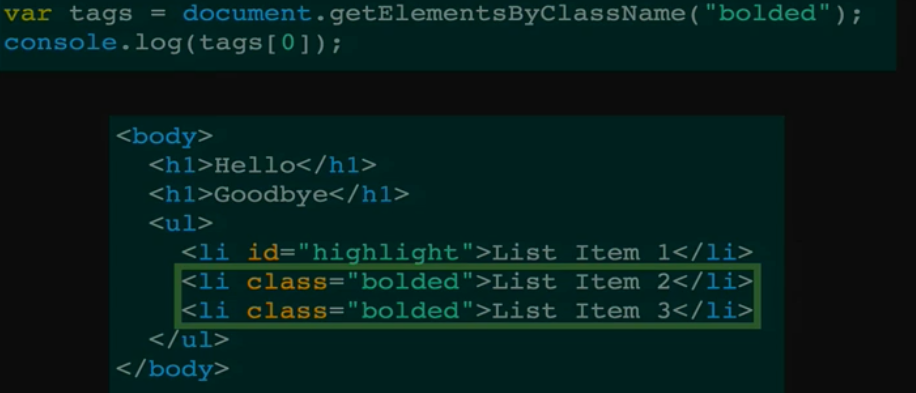
* Five important method of selecting

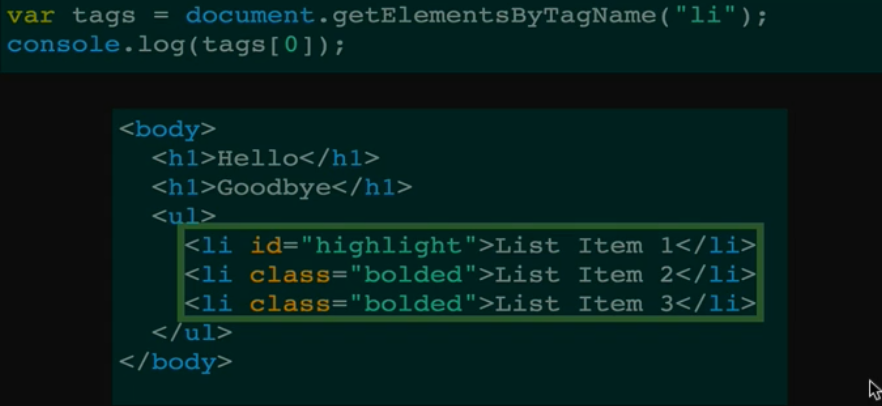


These methods are already in the document object.

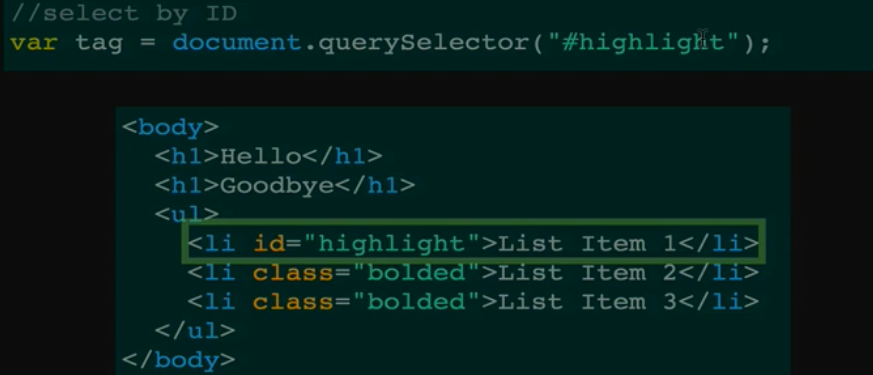
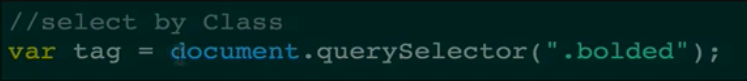
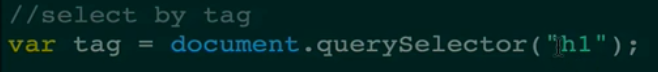
The class name and tag name selectors return a list, not an array. Its important cause we can’t use the array methods for those objects that we get.



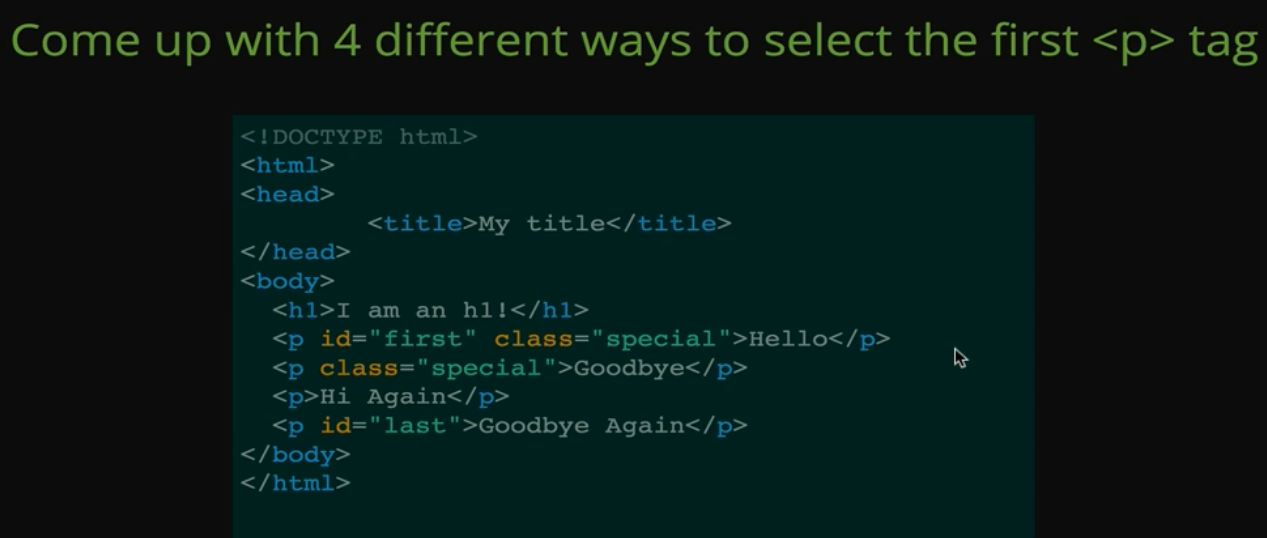




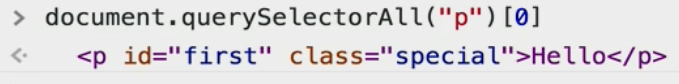
* Query selector is a newer method. We can do everything that the other three selector that we have seen so far does with Query Selector. It takes a CSS style selector (any selector that we can use in a CSS document). **If there is more than one result, it only gives the first element.**

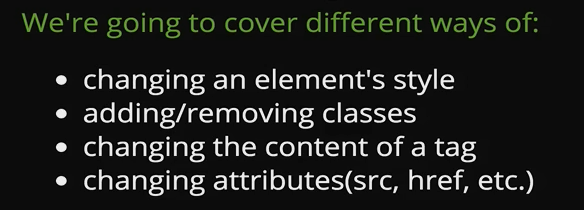
* Query Selector All is same as query selector but instead of returning the first item that it finds; this method returns all elements as a list.
* Exercise



* Solution

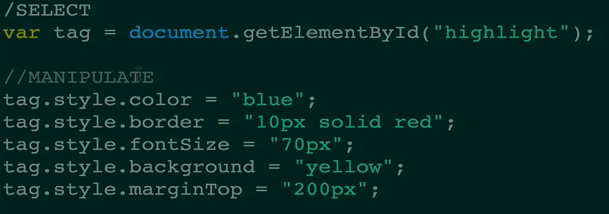
4.Manipulate



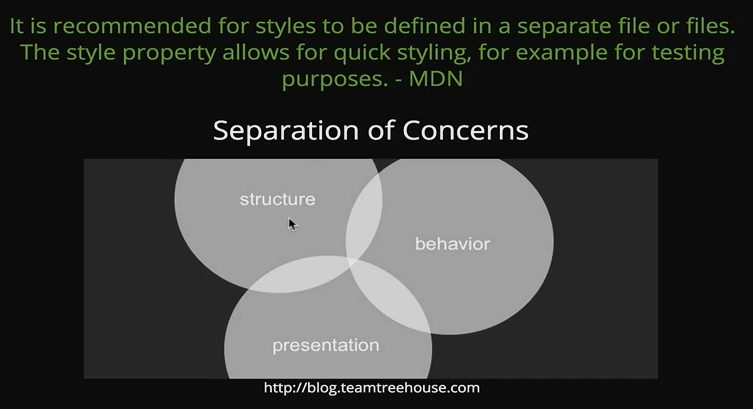
* Style is an object with huge amount of property. We can manipulate all style property of an element with this object. The right side must be a string. So, we

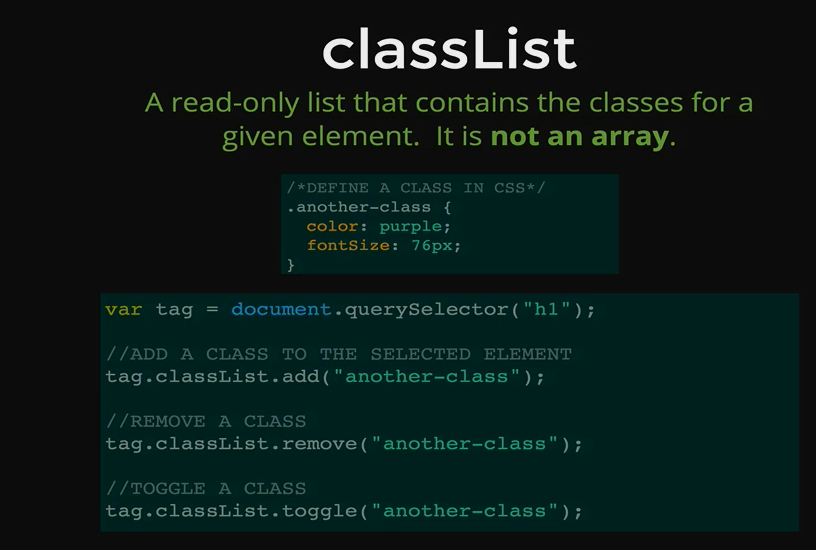
Must keep them between double quote. One exception here is, the css properties that has – is omitted in javascipt dom, instead they are translated into camelCase;

For example: CSS=> **background**-**color** | Dom -> **backgroundColor**



* One important thing:

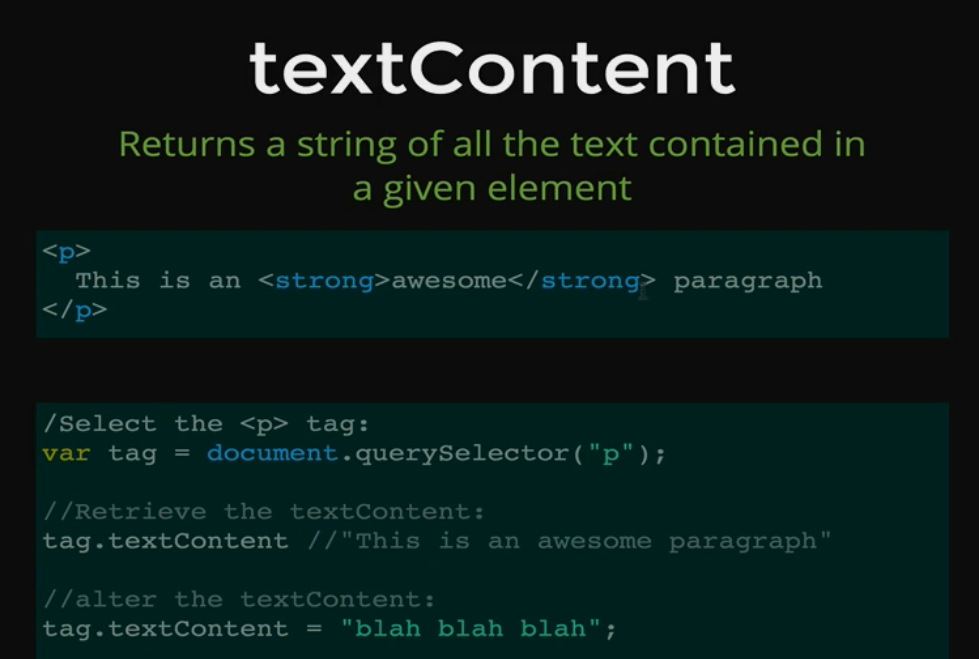


* The alternative solution 
* Classlist 

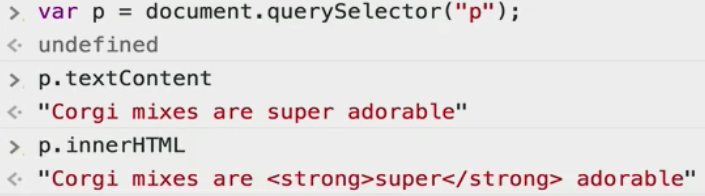
Toggle property is useful, if the element has that class already. It will turn it off. And if the element doesn’t have the class, it will add the class.

5.Manipulate Text and Content

* Text Content: this tag will return a string of all text contained inside the element, **but it will not include the tags that are inside**.

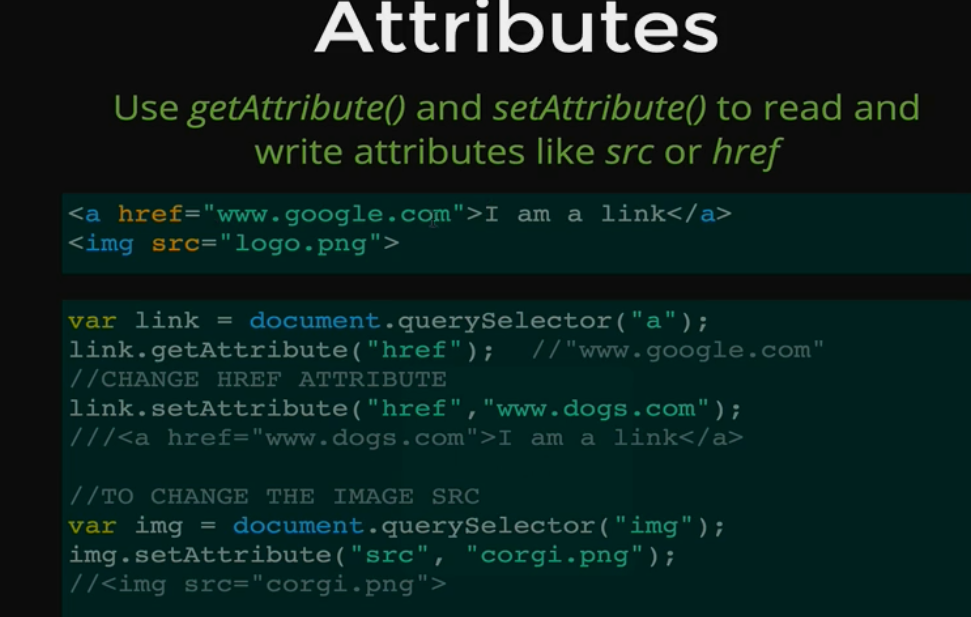


* For that part we can use innerHTML property, TextContent will treat the values inside as string. On the other hand, innerHTML will treat is as html code. So, if we want to write some html code with java script it will need innerHTML.



6.Manipulate Attributes

* Attributes are always like somepropertyname =” something”.



* One quick note, to change the href, we need to add the **Http://** to the link, otherwise it will treat link as local and won’t go to web.