

DOM is Document object model

- With JavaScript, you select and then manipulate to change.
- `Var h1 = document.querySelector("h1");` this is the selection.
 - `H1.style.color = pink;` this is manipulating.

Document methods

- `Document.getElementById()`
 - Search a part of HTML by the name of ID which is specific. These are JS objects and not HTML (returned value).
- `Document.getElementsByClassName()`
 - Those elements which share the same class,
 - This is a little like an array, has some similar functionality, but it is an object.
 - The `forEach` loop will not work.
- `Document.getElementsByTagName()`
 - Tag names e.g. "h1"
 - Again these are objects.
- `Document.querySelector()`
 - Takes a CSS style selector.
 - e.g. for ID `"#highlight"` for class `".bolded"`.
 - For class it will list only the first.
- `Document.querySelectorAll()`
 - Works the same way but will return all.
 - Will work if only one object also.

How to manipulate styles?

- the assignment requires `" "`.
- `.style` allows you to change the many style elements available.
- However there is a lot of repetitive code.
- To avoid this, on the CSS file you can create a class in which all these styles are mentioned.
- Thereafter you can use JS to fetch this class and apply the changes.
- `classList.add` will add the class features.
- `.remove` will remove

- .toggle will see if it already has, if not, then will add.
- E.g .some-class{
Font-size : 20;
Border : 5px solid red;
Color : blue;}

Manipulating the text content

- .textContent will display current text in a tag. However this does not take into account of any tags within the tag, e.g. if there is some bold words. This will treat the text as text
- .innerHTML this will do the same job, but will reveal the nested tags. This will treat the text as html.
- You can't use the above to overwrite as it will completely overwrite it.
- You don't need to make a variable each time. Start from document.

Manipulating Attributes

- getAttribute()
 - setAttribute(), this takes two parameters, old and new.
 - First the tag is searched for which has the attribute
- E.g. var img = document.querySelector("img");
img.setAttribute(src, new);

If you have a lot of links to change the attribute of, then you will need to use a for loop.

DOM Events

Making a webpage interactive.

- .addEventListener("action", function(){});
- If there is many grouped and you want to change only one. E.g. a group of links in tag. Method is to use a for loop.
- For(var i = 0; i < list.length; i++){
List[i].addEventListener("click", function(){});
- Clicked, when clicked will change
- Change, when any change happens.

