

# DOM Manipulation

# DOM Manipulation

We're going to cover different ways of:

- changing an element's style
- adding/removing classes
- changing the content of a tag
- changing attributes(src, href, etc.)

# Style

The style property is one way to manipulate an element's style

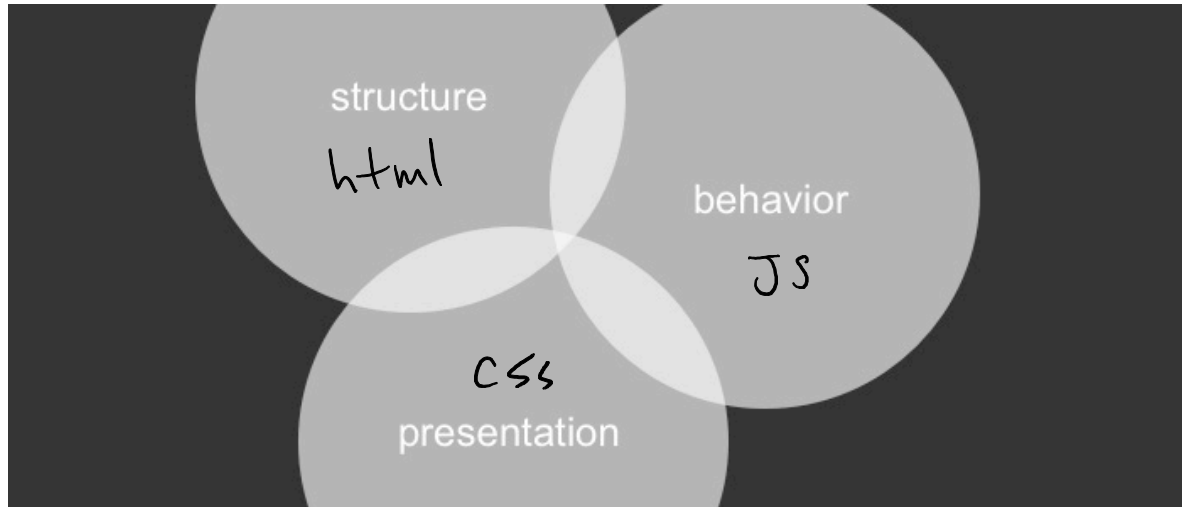
```
/SELECT
var tag = document.getElementById("highlight");

//MANIPULATE
tag.style.color = "blue";
tag.style.border = "10px solid red";
tag.style.fontSize = "70px";
tag.style.background = "yellow";
tag.style.marginTop = "200px";
```

# Is This a Bad Idea?

It is recommended for styles to be defined in a separate file or files.  
The style property allows for quick styling, for example for testing purposes. - MDN

## Separation of Concerns



<http://blog.teamtreehouse.com>

# An Alternative

Rather than directly manipulating style with JS, we can define a CSS class and then toggle it on or off with JS

```
//INSTEAD OF THIS:
```

```
var tag = document.getElementById("highlight");  
tag.style.color = "blue";  
tag.style.border = "10px solid red";
```

```
/*DEFINE A CLASS IN CSS*/  
.some-class {  
  color: blue;  
  border: 10px solid red;  
}
```

```
var tag = document.getElementById("highlight");  
//ADD THE NEW CLASS TO THE SELECTED ELEMENT  
tag.classList.add("some-class");
```

# classList

A read-only list that contains the classes for a given element. It is **not an array**.

```
/*DEFINE A CLASS IN CSS*/
.another-class {
  color: purple;
  fontSize: 76px;
}
```

```
var tag = document.querySelector("h1");
```

```
//ADD A CLASS TO THE SELECTED ELEMENT
tag.classList.add("another-class");
```

```
//REMOVE A CLASS
tag.classList.remove("another-class");    class name
```

```
//TOGGLE A CLASS
tag.classList.toggle("another-class");
```

# textContent

Returns a string of all the text contained in  
a given element

```
<p>  
  This is an <strong>awesome</strong> paragraph  
</p>
```

```
/Select the <p> tag:  
var tag = document.querySelector("p");
```

```
//Retrieve the textContent:  
tag.textContent //"This is an awesome paragraph"
```

```
all the text no matter of the other tags inside  
//alter the textContent:  
tag.textContent = "blah blah blah";
```

overwrite everything without preserving the other formatting tags  
everything is just text without identifying tags inside

# innerHTML

Similar to `textContent`, except it returns a string of all the HTML contained in a given element

```
<p>  
  This is an <strong>awesome</strong> paragraph  
</p>
```

```
//Select the <p> tag:  
var tag = document.querySelector("p");  
  
tag.innerHTML  
//"This is an <strong>awesome</strong> paragraph"
```



# Attributes

Use *getAttribute()* and *setAttribute()* to read and write attributes like *src* or *href*

```
<a href="www.google.com">I am a link</a>  

```

```
var link = document.querySelector("a");  
link.getAttribute("href"); // "www.google.com"  
//CHANGE HREF ATTRIBUTE http://www.dogs.com  
link.setAttribute("href", "www.dogs.com");  
///
```

```
//TO CHANGE THE IMAGE SRC  
var img = document.querySelector("img");  
img.setAttribute("src", "corgi.png");  
//
```