

Document Object Model

-This is a programming Interface that allows Javascript to access HTML element and styles change,update,remove (manipulate them).

1.What is DOM

2.Common Selectors

querySelector() - Selects the first element only.

querySelectorAll() - Selects all the elements with the name.

3.Textcontent

4.Value

5.Click Events.

addEventListener('click')

6.Manipulating styles using DOM.

document.querySelector().property=newvalue

7Classes in DOM

-Removing classes remove()

Accessing Element in Document

-We begin with document object.

-We then find our element with document object.

`document.getElementById(name)`

-Finds Element by ID name.

-By ID we use `#`

`document.getElementsByClassName(name)`

-Finds Element by Class Name.

-By Class we use `.`

`document.getElementsByTagName()`

-Finds Element by Tag Name

`textcontent`

-It used to get the text content written inside the element after find our element with document methods

```
document.getElementById(`name`).textContent
```

-We can assign new text using textcontent

```
document.getElementById(`name`).textContent=`new value`
```

value

-It is used to get or return the value of any attribute

-Attributes Come in name and value pair e.g name: `value`

```
document.getElementById(`name`).value
```

```
document.getElementById(`name`).value=`new value`
```

Handling Click Events

-The onclick executes a certain functionality when a button is clicked.

-We first select our element which is the button.

-Afterwards we add an event listener.

-The event listener takes in the `click` and the function we need.

```
document.querySelector(`selected element`).addEventListener(`click`,()=>)
```

Manipulating Styles using DOM.

-HTML DOM allows Javascript to change the style of HTML elements.

-Here we Select the element we want to change.

-Afterwards we add the style and then the property we want to change.

-Always Specify the property in Camel Case notation if its in two words.

-The value is usually stored in Strings.

Syntax.

```
document.querySelector(`element`).style.property = New Value ;
```

Removing Classes.

- We remove classes of our element by using remove() followed by the name of class we need to Remove.
- We first begin with the property of classlist then Remove

```
classList.remove()
```

Adding Classes.

- We remove classes of our element by using add() followed by the name of class we need to Remove.
- We first begin with the property of classlist then Remove.

```
classList.remove()
```

Key Events

- The event listener function should have the key event we want.
- Our function inside the event should take in a parameter `event`.
- When we wish to know the key pressed we call event.key and log it.

```
document.addEventListener('keyup',(event){  
    console.log(event.key)  
})
```

Key Up

Happens when we lift off our finger from keyboard.

```
document.addEventListener('keyup',(event){  
    console.log(event.key)  
})
```

Key Down

Happens when we once press our finger on keyboard

```
document.addEventListener('keydown',(event){  
    console.log(event.key)  
})
```

Key Press

Happens when we keep our finger on a certain Key.

```
document.addEventListener('keypress',(event){  
    console.log(event.key)  
})
```

DOM Methods.

insertAdjacentHTML()

- This is a method which inserts HTML code into a specified position in the DOM tree.
- The selected element comes first then the method.
- Inside the method is the position, then the html of the variable string to be inserted in the DOM.
- The positions are 4: beforebegin, afterbegin, beforeend, afterend
- afterbegin comes after the beginning of the element which is the first child.
- afterend comes after the element.
- beforebegin: before the element.
- beforeend: before the end of the element which is last child

```
<!-- beforebegin -->  
<p>  
    <!-- afterbegin -->  
    foo  
    <!-- beforeend -->  
</p>  
<!-- afterend -->
```

Syntax

```
selectedElementFromHTML.insertAdjacentHTML(position, html)
```

Example

-Selected Element

```
const h2 = document.getElementById("myH2");
```

-HTML Created

```
let html = "<p>My new paragraph.</p>";
```

-Added insertAdjacentHTML method

```
h2.insertAdjacentHTML("afterend", html);
```

