Topic: Dom – setAttribute, getAttribute, classlist, event listeners and event handlers

How to change attribute values by using setAttribute

Syntax:

This is some text and have id myDiv but it will changed to demo

var a=document.getElementById("myDiv").**setAttribute**("id","demo");

console.log(document)//can inspect and check weather it was changed or not

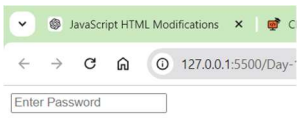
we can change the attribute by using .setAttribute.(“attribute name”,”attribute value”)

//output

Example-1:

```
var a=document.createElement("form");
      var b=document.createElement("input");
      b.setAttribute("type","password");
      b.setAttribute("placeholder","Enter Password");
      a.appendChild(b);
      document.body.appendChild(a);
```

Output:

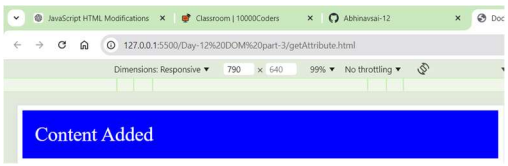


Example-2: img

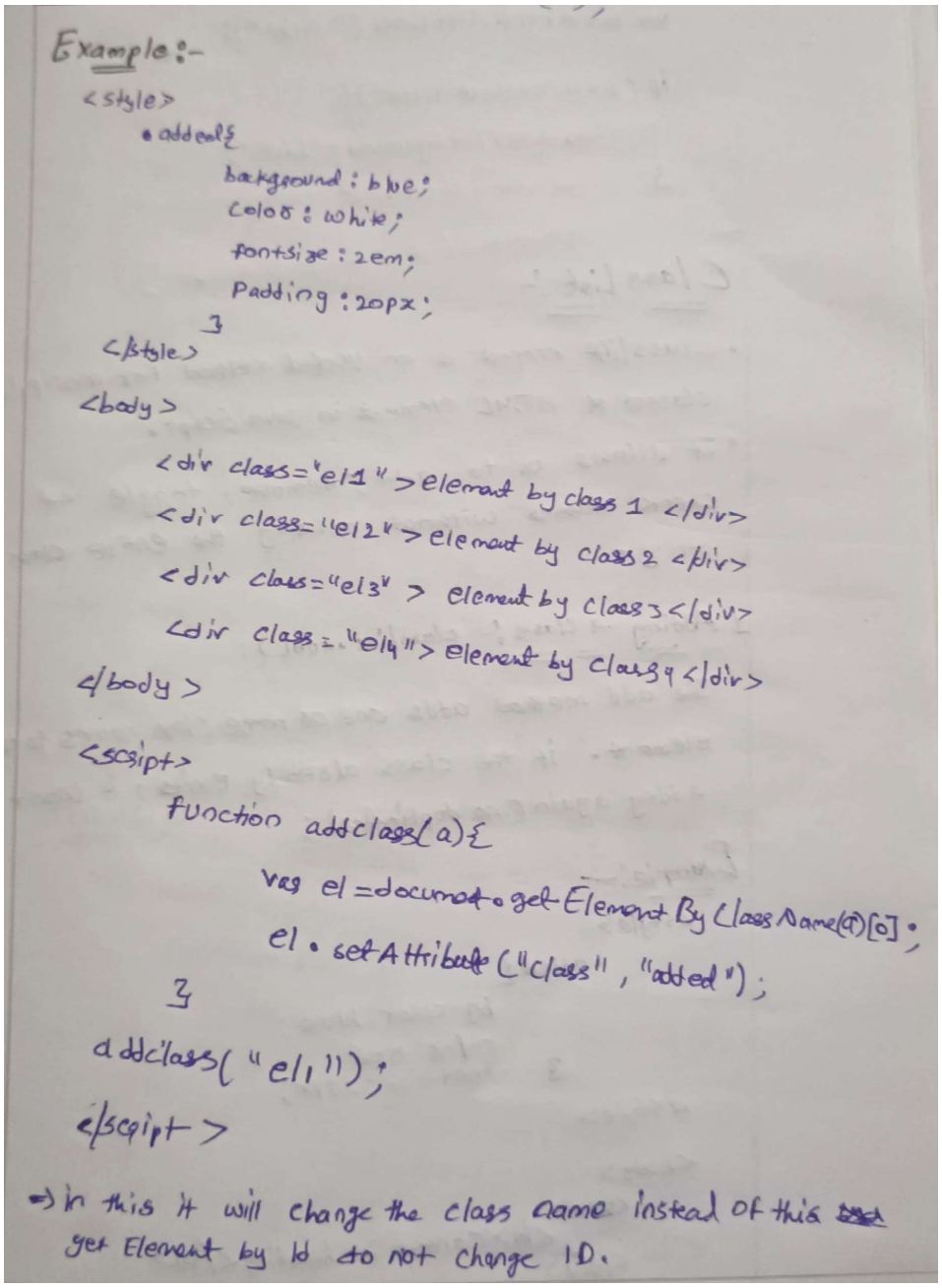
```
var a=document.createElement("img");
       img.setAttribute("src","a1.jpge");
       document.body.appendChild(a);
```

Example-3: How to set class name by using setAtribute and use css of class name

```
<style>
      .adder{
         background-color:blue;
         color: white;
         font-size:2em;
         padding:20px;
      }
   </style>
</head>
<body>
   <script>
      var a=document.createElement("div");
      a.innerHTML="Content Added";
      a.setAttribute("class","adder");
      document.body.appendChild(a);
   </script>
```



Example-4:



How to get attribute

We can get the element attribute by using get attribute method in dom

Example:

```
<div class="el1" title="ct">Content Added</div>
<script>
   var a=document.getElementsByClassName("el1")[0];
   var b=a.getAttribute("title");
   if(b=="ct"){
      a.style.background="blue";
   }
</script>
```

Classlist add and remove

- The classList property is an incredibly useful method for manipulating the classes of HTML elements in JavaScript.
- It allows you to add, remove, toggle, and check classes without altering the entire className string. Here's an in-depth explanation of how classList.add() and classList.remove() work.

1. **What is classList?**

• classList is a property that returns a live DOMTokenList collection of the classes of an element.

• You can think of classList as a way to interact with the classes applied to an HTML element, allowing you to add, remove, or toggle CSS classes dynamically.

2. Syntax: The syntax for accessing classList is as follows:

element.classList
classList methods include:
.add(), .remove(), .toggle()

3. **Adding a Class: classList.add()**

The add() method adds one or more class names to the element. If the class already exists, it won't be added again (no duplicates).

Syntax: element.classList.add(className1, className2, ..., classNameN);

Example:

```
<style>
      .added{
         background-color:blue;
         color: white;
         font-size:2em;
         padding:20px;
      }
   </style>
</head>
<body>
   <div>
      <h1>Content in heading</h1>
   </div>
   <button onclick="fun()">click here</button>
   <script>
      function fun(){
         var div=document.querySelector("div");
         div.classList.add("added");
      }
   </script>
</body>
```

4. **Removing a Class: classList.remove()**

The remove() method removes one or more class names from the element. If the class does not exist, nothing happens.

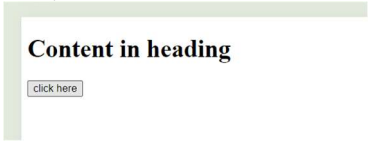
Syntax: element.classList.remove(className1, className2, ..., classNameN);

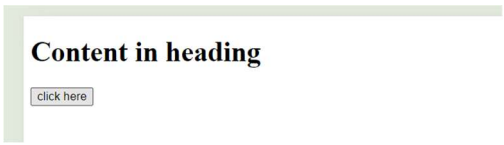
Example-1:

```
function fun(){
        var div=document.querySelector("div");
        div.classList.remove("added");
    }
```

Example-2:

```
var count=true;
    function fun(){
      var div=document.querySelector("div");
      if(count){
        div.classList.add("added");
        count=false;
      }
      else{
        div.classList.remove("added");
        count=true;
      }
    }
```





5. **Common Use Cases**

A. Toggling Classes (With classList.toggle())

• Sometimes, you may want to add a class if it's not present or remove it if it is. This can be done with the toggle() method.

Syntax: element.classList.toggle('className');

Example:

```
<div>
    <h1>Content in heading</h1>
  </div>
  <button onclick="fun()">click here</button>
  <script>
    var count=true;
    function fun(){
      var div=document.querySelector("div");
      div.classList.toggle("added");
    }
```

B. Checking If an Element Has a Class (classList.contains())

• To check if an element has a certain class, use the contains() method.

Syntax: if (element.classList.contains('className')) { // do something }

Types of Events

Mouse Events:

click: Occurs when a mouse button is clicked.

dblclick: Occurs when a mouse button is double-clicked.

mouseover: Occurs when the mouse pointer enters the area of an element.

mouseout: Occurs when the mouse pointer leaves the area of an element.

mousemove: Occurs when the mouse pointer is moved over an element.

Keyboard Events:

keydown: Occurs when a keyboard key is pressed down.

Keyup: Occurs when a keyboard key is released.

keypress: Occurs when a keyboard key is pressed and released.

Form Events:

submit: Occurs when a form is submitted.

change: Occurs when the value of an input element changes.

focus: Occurs when an element receives focus.

blur: Occurs when an element loses focus.

Window Events:

load: Occurs when a resource and its dependent resources have finished loading.

resize: Occurs when the browser window is resized.

scroll: Occurs when the user scrolls through a webpage.

Event handlers:

Event handlers are functions in JavaScript that are responsible for handling specific types of events. They define what should happen when a particular event occurs. Event handlers are associated with HTML elements and are triggered when the corresponding event takes place.

1.Inline Event Handlers: Inline event handlers are defined directly within the HTML markup using the on attribute followed by the event name.

<button onclick="myFunction()">Click me</button>

Example:

```
<div>
      <h1>Content in heading</h1>
   </div>
   <button onclick="fun()">click here</button>
   <script>
      var count=true;
      function fun(){
         var div=document.querySelector("div");
         div.classList.toggle("added")
      }
   </script>
```

2.DOM Event Handlers: DOM event handlers are assigned to HTML elements using JavaScript code.

You can attach event handlers using methods like addEventListener()

const button = document.getElementById('myButton');
button.addEventListener('click', myFunction);

Event listeners:

Event listeners in JavaScript are functions that wait for a specific event to occur and then execute code in response to that event.

Using addEventListener() Method: The addEventListener() method attaches an event listener to an HTML element.

It takes three parameters: the event name, the function to be executed when the event occurs, and an optional boolean value indicating whether to use capturing or bubbling (default is false, indicating bubbling).

```
const button = document.getElementById('myButton');
button.addEventListener('click', function() {
    console.log('Button clicked!');
});
```

Removing Event Listeners: You can remove event listeners using the removeEventListener() method. It requires the same parameters as addEventListener().

```
function handleClick() {
   console.log('Button clicked!');
}
const button = document.getElementById('myButton');
button.addEventListener('click', handleClick);
// Later, if you want to remove the event listener
button.removeEventListener('click', handleClick);
```

==========

Mouse Events Examples:

1. click: Occurs when a mouse button is clicked.

```
<div>
      <h1>Content in heading</h1>
   </div>
   <button>click here</button>
   <script>
      var btn=document.querySelector("button");
      var div=document.querySelector("div");
      btn.addEventListener("click",fun);
            function fun(){
```

```
        var div=document.querySelector("div");
        div.style.color="red";
        div.style.background="blue";
    }
</script>
```

2. **dblclick:** Occurs when a mouse button is double-clicked.

```
<div>
    <h1>Content in heading</h1>
</div>
<button>click here</button>
<script>
    var btn=document.querySelector("button");
    var div=document.querySelector("div");
    btn.addEventListener("dblclick",fun2);

// dblclick
    function fun2(){
        var div=document.querySelector("div");
        div.style.color="white";
        div.style.background="blue";
    }
</script>
```

3. **mouseover:** Occurs when the mouse pointer enters the area of an element.

```
<div>
        <h1>Content in heading</h1>
</div>
<button>click here</button>
<script>
    var btn=document.querySelector("button");
    var div=document.querySelector("div");
    div.addEventListener("mouseover",fun3);

    // mouseover
    function fun3(){
        var div=document.querySelector("div");
        div.style.color="white";
        div.style.background="red";
    }
<script>
```

4. mousemove: Occurs when the mouse pointer is moved over an element.

```
<div>
    <h1>Content in heading</h1>
</div>
<button>click here</button>
<script>
    var btn=document.querySelector("button");
    var div=document.querySelector("div");
    div.addEventListener("mousemove",fun4);

    //mousemove
    function fun4(){
        var div=document.querySelector("div");
        div.style.color="red";
        div.style.background="green";
    }
</script>
```

5. mouseout: Occurs when the mouse pointer leaves the area of an element.

```
<div>
    <h1>Content in heading</h1>
</div>
<button>click here</button>
<script>
```

```
    var btn=document.querySelector("button");
    var div=document.querySelector("div");
    div.addEventListener("mouseout",fun5);
    // mouseout
    function fun5(){
        var div=document.querySelector("div");
        div.style.color="red";
        div.style.background="grey";
    }
</script>
```

Keyboard Events:

1. keydown: Occurs when a keyboard key is pressed down.

```
<script>
    document.addEventListener("keydown",function(event){
        console.log(event.key);
    })
</script>
```

2. Keyup: Occurs when a keyboard key is released.

```
<script>
    document.addEventListener("keyup",function(event){
        console.log(event.key);
    })
</script>
```

3. keypress: Occurs when a keyboard key is pressed and released.

```
<script>
    document.addEventListener("keypress",function(event){
        console.log(event.key);
    })
</script>
```

Tasks:

1. **Modify element attributes: Select an image element and change its src and alt attributes.**
   Code:

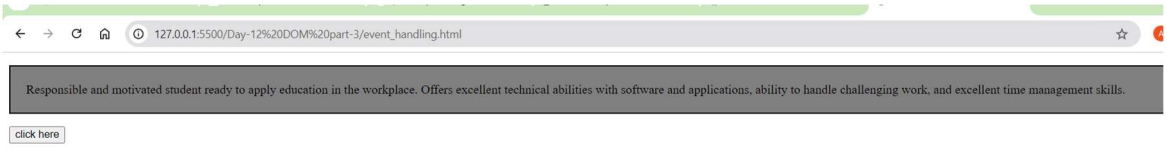
```
<img id="js" src="https://training.epam.com/static/news/496/JavaScriptforatestautomationengineerfeaturesadvantagestools_014111.png"
alt="myimg" height="350px" width="350px">
    <script>
        var img=document.getElementById("js");
        img.src="https://miro.medium.com/v2/resize:fit:800/1*PEzOBf4AkvDoE4VME4kq4Q.jpeg";
        img.alt="new js img";
        console.log(img);
    </script>
```

2. **Event handling:** Write a script that adds a click event listener to a button that changes the text of a paragraph when clicked.
   Code:

```
    <p>Event handling: Write a script that adds a click event listener
        to a button that changes the text of a paragraph when clicked.
    </p>
    <button>click here</button>

    <script>
        var btn=document.querySelector("button");
        btn.addEventListener("click",fun);
    function fun(){
            var p=document.querySelector("p");
            p.innerText="Responsible and motivated student ready to apply education in the workplace. Offers excellent technical abilities with
software and applications, ability to handle challenging work, and excellent time management skills.";
            p.style.background="grey";
            p.style.padding="20px";
            p.style.border="2px solid black";
```

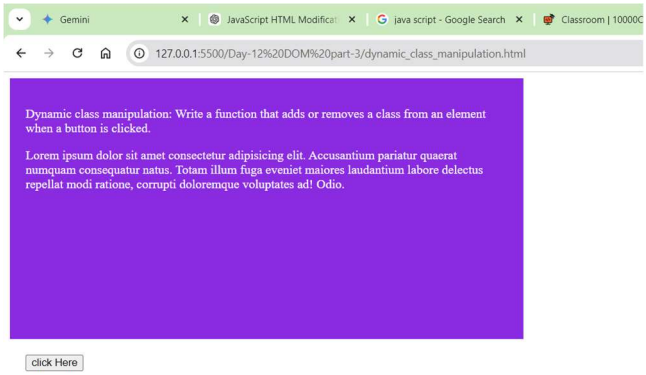
```
    }
  </script>
```

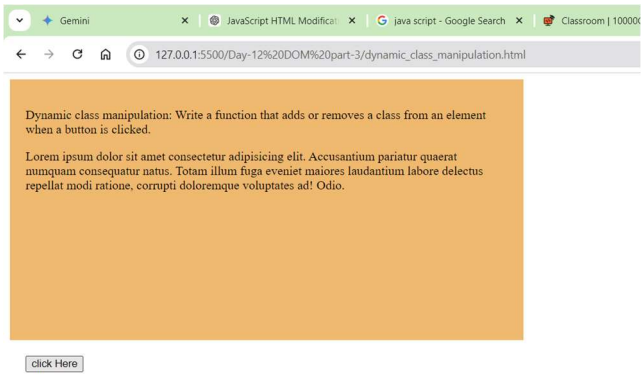


3. Dynamic class manipulation: Write a function that adds or removes a class from an element when a button is clicked.

Code:

```
<style>
    .div11{
      background-color:blueviolet;
      color: white;
      padding:20px;
      height:40vh;
      width: 40vw;
    }
    .adder{
      background-color: rgb(240, 185, 112);
      color: black;
      padding:20px;
      height:40vh;
      width: 40vw;
    }
    button{
      margin: 20px;
    }
  </style>
</head>
<body>
  <!--
    Dynamic class manipulation: Write a function that adds or removes
    a class from an element when a button is clicked.
  -->
  <div id="div11" class="div11">
    <p>Dynamic class manipulation: Write a function that adds or removes
      a class from an element when a button is clicked.
    </p>
    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Accusantium
      pariatur quaerat numquam consequatur natus. Totam illum fuga eveniet
      maiores laudantium labore delectus repellat modi ratione, corrupti doloremque voluptates ad! Odio.
    </p>
  </div>
  <button onclick="fun()">click Here</button>
  <script>
    function fun(){
      var div=document.getElementById("div11");
      div.classList.toggle('adder');
    }
  </script>
</body>
```

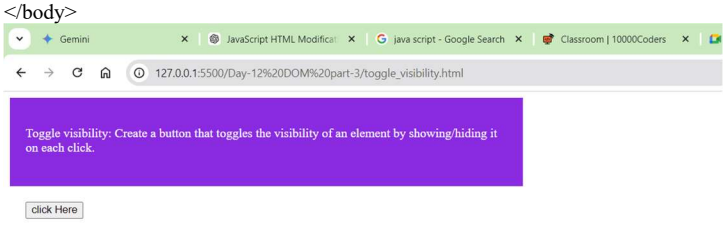


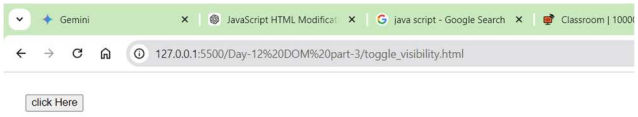


4. Toggle visibility: Create a button that toggles the visibility of an element by showing/hiding it on each click.

Code:

```
<style>
        .div1{
            background-color:blueviolet;
            color: white;
            padding:20px;
            height:10vh;
            width: 40vw;
        }
        .adder{
        display: none;
        }
        button{
            margin: 20px;
        }
    </style>
</head>
<body>
    <div id="div1" class="div1">
        <p>Toggle visibility: Create a button that toggles the visibility of
            an element by showing/hiding it on each click.
        </p>
    </div>
    <button onclick="fun()">click Here</button>
    <script>
        function fun(){
            var div=document.getElementById("div1");
            div.classList.toggle('adder');
        }
    </script>
</body>
```





5. Task:5 => by clicking the button display an image

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style>
        .div1{
            background-color:blueviolet;
            color: white;
            padding:20px;
            height:40vh;
            width: 40vw;
        }
        .html{
            background-image: url("https://media.licdn.com/dms/image/D4D12AQEaTK5_cv5Fmg/article-cover_image-
shrink_720_1280/0/1677082453584?e=2147483647&v=beta&t=hy4wcIl2vOpp5lr5k74obYVN63IT-U4lydNOIo3hHGo");
            background-repeat: no-repeat;
            background-size: cover;
            color: white;
            padding:20px;
            height:40vh;
            width: 40vw;
        }
        .css{
            background-image: url("https://ksra.eu/wp-content/uploads/2021/05/Vp9WvV7YKdH4k8sKRePcE8-1200-80.jpeg");
            background-repeat: no-repeat;
            background-size: cover;
            color: white;
            padding:20px;
            height:40vh;
            width: 40vw;
        }
        .boot{
            background-image: url("https://www.drupal.org/files/project-images/b5-new-logo.png");
            background-repeat: no-repeat;
            background-size: cover;
            color: white;
            padding:20px;
            height:40vh;
            width: 40vw;
        }
        .js{
            background-image: url("https://cyberhoot.com/wp-content/uploads/2020/07/Free-Courses-to-learn-JavaScript-1024x576.jpg");
            background-repeat: no-repeat;
            background-size: cover;
            color: white;
            padding:20px;
            height:40vh;
            width: 40vw;
        }
        button{
            margin: 20px;
            margin-left:60px;
        }
    </style>
</head>
<body>
    <div id="div1" class="div1">

    </div>
    <button onclick="html()">HTML</button>
    <button onclick="css()">CSS</button>
    <button onclick="boot()">Bootstrap</button>
    <button onclick="js()">Java Script</button>

    <script>
```

```
        function html(){
            var div=document.getElementById("div1");
            div.classList.toggle('html');
        }

        function css(){
            var div=document.getElementById("div1");
            div.classList.toggle('css');
        }

        function boot(){
            var div=document.getElementById("div1");
            div.classList.toggle('boot');
        }

        function js(){
            var div=document.getElementById("div1");
            div.classList.toggle('js');
        }
    </script>

</body>
</html>
```