**Example 1**

<div id="div-01">Here is div-01</div>

<div id="div-02">Here is div-02</div>

<li>This is a list item</li>

<li>This is another list item</li>

<div id="div-03">Here is div-03</div>

<script>

let el = document.getElementById("div-03").previousElementSibling;

document.write("<p>Siblings of div-03</p><ol>");

while (el) {

document.write("<li>" + el.nodeName + "</li>");

el = el.previousElementSibling;

}

document.write("</ol>");

</script>

**Example 2**

const div = document.createElement('div');

// creates a new div referenced in the variable 'div'

**Example 3: Adding Inline Style**

div.style.color = 'blue';

// adds the indicated style rule

div.style.cssText = 'color: blue; background: white;';

// adds several style rules

div.setAttribute('style', 'color: blue; background: white;');

// adds several style rules

div.setAttribute('id', 'theDiv');

// if id exists, update it to 'theDiv', else create an id

// with value "theDiv"

div.getAttribute('id');

// returns value of specified attribute, in this case

// "theDiv"

div.removeAttribute('id');

// removes specified attribute

**Example 4: Working with Classes**

div.classList.add('new');

// adds class "new" to your new div

div.classList.remove('new');

// removes "new" class from div

div.classList.toggle('active');

// if div doesn't have class "active" then add it, or if

// it does, then remove it

It is often standard (and cleaner) to toggle a CSS style rather than adding and removing inline CSS.

**Example 5: Adding Text Content**

div.textContent = 'Hello World!'

// creates a text node containing "Hello World!" and

// inserts it in div

**Example 6: Adding HTML Content**

div.innerHTML = '<span>Hello World!</span>';

// renders the HTML inside div

**Example 7**

<html>

<body>

<h1>

THE TITLE OF YOUR WEBPAGE

</h1>

<div id="container"></div>

</body>

// your JavaScript file

const container = document.querySelector('#container');

const content = document.createElement('div');

content.classList.add('content');

content.textContent = 'This is the glorious text-content!';

container.appendChild(content);

Exercise

Add the following elements to the container using ONLY JavaScript and the DOM methods shown above.

1. a <p> with red text that says “Hey I’m red!”
2. an <h3> with blue text that says “I’m a blue h3!”
3. a <div> with a black border and pink background color with the following elements inside of it:
4. another <h1> that says “I’m in a div”
5. a <p> that says “ME TOO!”
6. Hint for this one: after creating the <div> with createElement, append the <h1> and <p> to it before adding it to the container.