

Name >> Muhammad Huzaifa
Roll No >> 04

Assignment 6
20th August 2023

Module 2 : Front End Development (JS_Web)

Submission Requirements: Please upload your codes in PDF File on Google Classroom in the relevant Assignment section.

Note: Plagiarism is a serious violation. Zero marks will be awarded in case plagiarism is found. You are required to perform the following tasks.

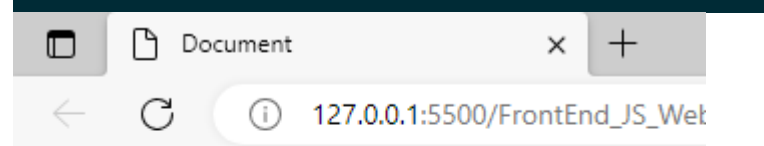
Class Questions:

- Remove direct text placed inside a div.

Solution:

```
<!-- Class Questions -->
<div id="one">
  Kaleem
  <p>Mujtaba</p>
  <h1>Ahmad</h1>
</div>

// Class Task One
const one = document.getElementById('one');
one.removeChild(one.firstChild);
```



Mujtaba

Ahmad

- Add/Remove text in an element based on hover and blur on button respectively.

Solution:

Part 1:

- Use `document.getElementById(id)` to find an element with a specific ID and store it in a variable.

Solution:

```
<!-- Task_1 -->
<div id="first">
  Lorem ipsum dolor sit amet consectetur adipisicing elit. Ipsum doloribus laborum dolores eos! Repudiandae sunt, exercitationem ea ipsam sequi a magnam esse. Deleniti dolore sequi dolorem sint recusandae pariatur corrupti.
</div>
```

```
// task_1
const first = document.getElementById('first');
console.log(first);
```

- Utilize `document.getElementsByTagName(name)` to find all elements with a specific tag name and log them to the console.

Solution:

```
<!-- Task_2 -->
<h1>Ahmad</h1>
<h1>Muhtaba</h1>
<h1>Kainat</h1>
```

```
// task_2
console.log(document.getElementsByTagName('h1'));
```

- Use `document.getElementsByClassName(name)` to find all elements with a specific class name and apply a CSS style to each of them.

Solution:

```
<!-- Task_3 -->
<div>
  <p class="para">Lorem ipsum dolor sit, amet consectetur adipisicing elit. Est voluptate illo perferendis nihil nulla totam maxime dolorem, dolorum eos beatae, hic, ipsam laudantium natus sapiente dolore. Mollitia magni minus labore.</p>
  <h1 class="para">Muhammad Huzaifa</h1>
  <section class="para">Lorem ipsum dolor sit amet consectetur adipisicing elit. Facere sequi repellendus modi magni neque accusantium tempore, aperiam repellat, molestias nulla delectus assumenda tenetur sed! Eveniet animi natus iste fugit commodi.</section>
</div>
```

```
// task_3
const task_3 = document.getElementsByClassName('para');
// console.log(task_3[0]);
task_3[1].style.color = "red";
```

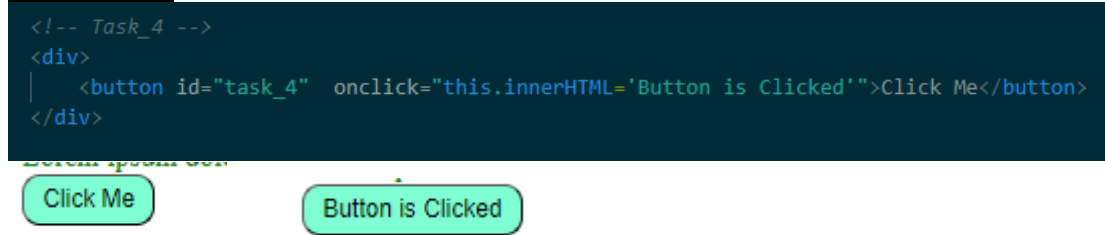
Lorem ipsum dolor sit, amet consectetur adipisicing elit. Est voluptate illo perferendis nihil nulla totam maxime dolorem, dolorum eos beatae, hic, ipsam laudantium natus sapiente dolore. Mollitia magni minus labore.

Muhammad Huzaifa

Lorem ipsum dolor sit amet consectetur adipisicing elit. Facere sequi repellendus modi magni neque accusantium tempore, aperiam repellat, molestias nulla delectus assumenda tenetur sed! Eveniet animi natus iste fugit commodi.

- Change the inner HTML of an element with `element.innerHTML` to display new content when a button is clicked.

Solution:

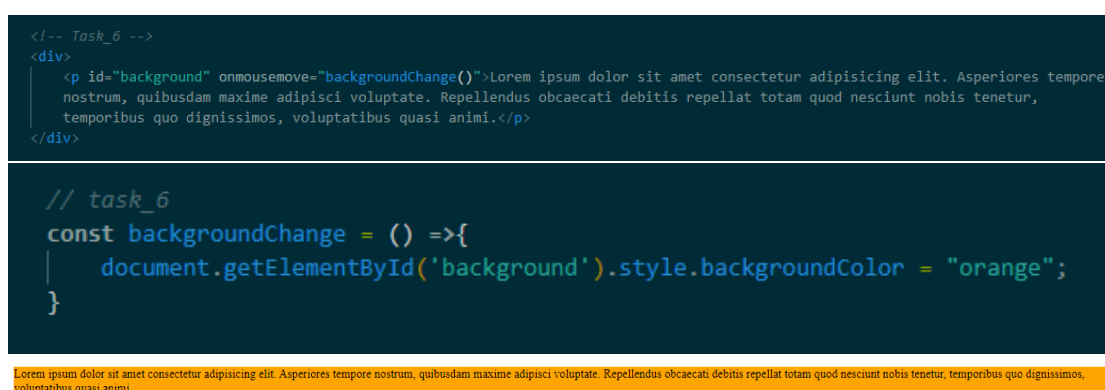


- Update the value of an attribute in an HTML element using the `element.attribute` property based on user input.

Solution:

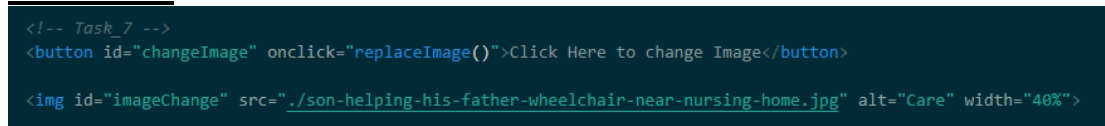
- Create a function that changes the background color of an element using the `element.style.property`.

Solution:



- Create a button that, when clicked, uses `element.setAttribute(attribute, value)` to modify the 'src' attribute of an image.

Solution:



```
// task_7
const replaceImage = () =>{
  const element = document.getElementById('imageChange');
  element.setAttribute('src', './ruben-leija-jY_knL-TVvA-unsplash.jpg')
}
```



[Click Here to change Image](#)



[Click Here to change Image](#)

- Develop a form where users can input their name and a message. On submission, use `element.innerHTML` to display a personalized message.

Solution:

```

<!-- Task_8 -->
<form>
  <br><br><br><br>
  <h1 id="customMessage"></h1>
  <br><br><br><br>

  <label for="name">Name: </label>
  <input type="text" id="name" class="name">

  <br><br>

  <label for="message">Any Message</label>
  <textarea name="message" id="message" cols="50" rows="5"></textarea>

  <button onclick="display()">Submit</button>

</form>

```

```

//task_8
const display =() =>{
  const elem = document.getElementById('customMessage');
  elem.innerText = 'Thanks for submitting your Response';
}

```

Thanks for submitting your Response

Name:

Any Message

- Build a list of items with checkboxes. When a checkbox is clicked, use `element.style.property` to strike through the item's text.

Solution:

```

<!-- Task_9 -->
<label id="book" for="bookreading">Book Reading</label>
<input type="checkbox" name="bookreading" onclick="strike()"><br>

<label id="cric" for="cricket">Cricket</label>
<input type="checkbox" name="cricket" onclick="strike1()"><br>

<label id="foot" for="football">Football</label>
<input type="checkbox" name="football" onclick="strike2()"><br>

<label id="photo" for="photography">Photography</label>
<input type="checkbox" name="photography" onclick="strike3()"><br>

<label id="movie" for="movies">Movies</label>
<input type="checkbox" name="movies" onclick="strike4()"><br>

```

```

// task_9
const strike = ()=>{
  const eleme = document.getElementById('book');
  eleme.style.border='3px solid black';
}
const strike1 = ()=>{
  const eleme = document.getElementById('cric');
  eleme.style.border='3px solid black';
}
const strike2 = ()=>{
  const eleme = document.getElementById('foot');
  eleme.style.border='3px solid black';
}
const strike3 = ()=>{
  const eleme = document.getElementById('photo');
  eleme.style.border='3px solid black';
}
const strike4 = ()=>{
  const eleme = document.getElementById('movie');
  eleme.style.border='3px solid black';
}

```

Book Reading ☐

Cricket ☒

Football ☐

Photography ☒

Movies ☐

- Use `document.createElement(element)` to create a new paragraph element and append it to an existing div.

Solution:

```
<!-- Task_10 -->
<div id="exist">
  |   Already Existing Div.
</div>
```

```
// task_10
const newElement = document.createElement('p').innerHTML = 'Appending new Paragraph';
document.getElementById('exist').append(newElement);
```

Already Existing Div. Appending new Paragraph

- Create a button that, when clicked, removes a specific list item using `document.removeChild(element)`.

Solution:

```
<!-- Task_11 -->
<ul id="cities">
  <li>Lahore</li>
  <li>Islamabad</li>
  <li>Karachi</li>
  <li>Sukkur</li>
</ul>
<button id="remove" onclick="removeElement()">Remove Item</button>
```

```
// task_11
const removeElement={()=>{
  const rem = document.getElementById('cities');
  rem.removeChild(rem.lastChild)
}}
```

- Lahore
- Islamabad
- Karachi
- Sukkur

- Lahore
- Islamabad
- Karachi

Remove Item

Remove Item

- Implement a feature that adds a new list item to an existing unordered list using `document.appendChild(element)`.

Solution:

```
<!-- Task_12 -->
<ul id="add_Item">
  <li>Waqas</li>
  <li>Shaani</li>
  <li>Butt</li>
</ul>
```

```
// task_12
const new_Element = document.createElement('li');
new_Element.innerHTML = 'Huzaifa Added';
document.getElementById('add_Item').appendChild(new_Element);
```

- Waqas
 - Shaani
 - Butt
 - Huzaifa Added
- Replace an existing image on a webpage with a new image by using document.replaceChild(new, old).

Solution:

```
<!-- Task_13 -->
<div id="parent">
  
</div>
```

```
// task_13
const parent = document.getElementById("parent");
const oldChild = document.getElementById("oldChild");
const repChild = document.createElement('img');
repChild.src = './nature-landscape-hd-usqznq19dscdjkf8.jpg';
repChild.style.width = '35%';
parent.replaceChild(repChild, oldChild);
```




- Design a page with two sections. Upon clicking a button, swap the positions of these sections using the `document.replaceChild(new, old)` method.

Solution:

```
<!-- Task_14 -->
<div id="sectionSwap">
  <section id="sec_one">
    |   Left Side
  </section>
  <section id="sec_two">
    |   Right Side
  </section>
</div>

<button id="swapBoxes" onclick="swap()">Swap</button>
```

```
// Task_14
const swap={()=>{
  const parent = document.getElementById('sectionSwap');

  const sec1 = document.getElementById('sec_one');
  const sec2 = document.getElementById('sec_two');

  parent.replaceChild(sec2, sec1);
  parent.replaceChild(sec1, sec2);
}}
```

Left Side

Right Side

Left Side

Swap

- Create a simple website layout with placeholders. Use JavaScript to replace these placeholders with actual content using `document.replaceChild(new, old)`.

Solution:

```
<!-- Task_15 -->
<form class="FormParent">
  <label for="name">Name: </label>
  <input type="text" placeholder="EnterName" id="name" class="name">

  <label for="email">Email: </label>
  <input type="email" placeholder="EnterMail" id="email" class="email">

  <label for="age">Age: </label>
  <input type="number" placeholder="EnterAge" id="age" class="age">
</form>
```

- Create a button on a webpage. Attach an event listener using `addEventListener` to change the button's text to "Clicked!" when it's clicked.

Solution:

```

<!-- Task_16 -->
<div>
  <h1 class="heading">Task 16</h1>
  <h3>Create a button on a webpage. Attach an event listener using addEventListener to change the button's text to "Clicked!" when it's clicked</h3>
  <div id="listener">
    <button id="text">Click Me!</button>
  </div>
  <script>
    const listener = document.getElementById('listener');
    listener.addEventListener('click', function(event){
      const text = document.getElementById('text');
      text.innerHTML = 'Changed Text'
    });
  </script>
</div>

```

Click Me!

Changed Text

- Design an image that changes its source to another image when hovered over by the mouse pointer. Use the mouseover and mouseout events to achieve this effect.

Solution:

```

<!-- Task_16 -->
<div>
  
</div>

```

```

// Task_16
const magic={()=>{
  const prevImg = document.getElementById('prevImg');
  prevImg.src = './omar-ram-xH6zaxF3k6Q-unsplash.jpg'
}}
const magicBack={()=>{
  const prevImg = document.getElementById('prevImg');
  prevImg.src = './jannes-glas-0NaQQsLWLkA-unsplash.jpg';
}}

```



- Build a registration form with input fields for name, email, and password. Use the submit event on the form to validate that the password is at least 8 characters long. Display an error message if the validation fails.

Solution:

```

<!-- Task_18 -->
<div>
  <h1 class="heading">Task 18</h1>
  <h3>Build a registration form with input fields for name, email, and password. Use the submit event on the form to validate that the password is at least 8 characters long. Display an error message if the validation fails</h3>
  <div>
    <form id="regForm">
      <label for="name">Enter your name <input type="text" name="name" id="getName" placeholder="name here"></label>
      <label for="email">Enter your email <input type="email" name="email" id="getemail" placeholder="email here"></label>
      <label for="password">Enter password <input type="password" name="password" id="getpass" placeholder="password here"></label>
      <input type="submit" value="Register">
      <span id="error"></span>
    </form>
    <script>
      const regForm = document.getElementById('regForm');
      const password = document.getElementById('getpass');
      const error = document.getElementById('error');

      regForm.addEventListener('submit', function(event) {
        const passwordValue = password.value;
        if (passwordValue.length < 8) {
          const error = document.getElementById('error');
          error.innerHTML = 'Password must be atleast 8 char!';
          event.preventDefault();
        }
      });
    </script>
  </div>
</div>

```

Enter your name Enter your email Enter password Password must be atleast 8 char!

- Create a text input field. Use the keydown event to count and display in real-time the number of characters entered in the input field. Update the count as the user types.

Solution: