

Assignment submission

Name – Sayali Suresh Chavan

Email – sayalichavan0512@gmail.com

Q1. Contact Form

Code –

*HTML file –

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="style1.css">
  <title>Contact Form</title>
</head>
<body>

<!-- Contact Form -->
<form action="https://www.greatfrontend.com/api/questions/contact-form"
method="post">
  <label for="name">Name:</label>
  <input type="text" id="name" name="name" required>

  <label for="email">Email:</label>
  <input type="email" id="email" name="email" required>

  <label for="message">Message:</label>
  <textarea id="message" name="message" rows="4" required></textarea>

  <button type="submit">Submit</button>
</form>

</body>
</html>
```

*CSS file –

```
body {
```

```
font-family: 'Arial', sans-serif;
background-color: #f4f4f4;
margin: 0;
padding: 0;
display: flex;
justify-content: center;
align-items: center;
height: 100vh;
}

form {
  background-color: #fff;
  padding: 20px;
  border-radius: 8px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  width: 300px;
}

label {
  display: block;
  margin-bottom: 8px;
}

input,
textarea {
  width: 100%;
  padding: 8px;
  margin-bottom: 16px;
  box-sizing: border-box;
  border: 1px solid #ccc;
  border-radius: 4px;
}

button {
  background-color: #007BFF;
  color: #fff;
  padding: 10px;
  border: none;
  border-radius: 4px;
  cursor: pointer;
}

button:hover {
  background-color: #0056b3;
}
```

Q2. Develop tabs

*HTML file –

```
<html lang="en"><head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tab Example</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>

<div class="tab-container">
  <div class="tab active" onclick="showTab('htmlTab')">HTML</div>
  <div class="tab" onclick="showTab('cssTab')">CSS</div>
  <div class="tab" onclick="showTab('jsTab')">JavaScript</div>
</div>

<div id="htmlTab" class="tab-content active">
  <h2>HTML</h2>
  <p>HTML is the standard markup language for creating and structuring web pages. It provides a set of elements or tags that define the structure of content on a webpage.</p>
</div>

<div id="cssTab" class="tab-content">
  <h2>CSS</h2>
  <p>CSS is a stylesheet language used to describe the presentation and style of HTML documents. It enables the separation of content and presentation, allowing for consistent styling across multiple pages.</p>
</div>

<div id="jsTab" class="tab-content">
  <h2>JavaScript</h2>
  <p>JavaScript is a high-level, interpreted programming language that enables dynamic and interactive behavior on webpages. It is commonly used for client-side scripting, allowing manipulation of the Document Object Model (DOM) and handling events.</p>
</div>

<script src="script.js"></script>
</body>
</html>
```

*CSS File –

```
.tab-container {
```

```

display: flex;
}

.tab {
padding: 10px;
cursor: pointer;
border: 1px solid #ccc;
border-radius: 4px 4px 0 0;
background-color: #f4f4f4;
margin-right: 5px;
}

.tab.active {
background-color: #007BFF;
color: #fff;
}

.tab-content {
display: none;
padding: 10px;
border: 1px solid #ccc;
border-radius: 0 0 4px 4px;
background-color: #fff;
}

.tab-content.active {
display: block;
}

```

*JavaScript File –

```

function showTab(tabId) {
// Hide all tab content
var tabs = document.querySelectorAll('.tab-content');
tabs.forEach(function(tab) {
    tab.classList.remove('active');
});

// Remove active class from all tabs
var tabButtons = document.querySelectorAll('.tab');
tabButtons.forEach(function(tabButton) {
    tabButton.classList.remove('active');
});
}

```

```
// Show the selected tab content and make the tab active
document.getElementById(tabId).classList.add('active');
event.currentTarget.classList.add('active');
}
```

Q3.

The screenshot shows the Chrome DevTools interface. The 'Sources' panel is active, displaying a file named 'index.js' with the following code:

```
1  const input = { a: 1, b: 2, c: 3 };
2  const output = [];
3
4  for (const key in input) {
5    if (input.hasOwnProperty(key)) {
6      const newObj = {};
7      newObj[key] = input[key];
8      output.push(newObj);
9    }
10 }
11
12 console.log(output);
```

The cursor is positioned at Line 12, Column 21. Below the code editor, the 'Console' panel shows the output of the log statement:

```
{ } Line 12, Column 21
```

The console output is as follows:

```
New      Issues
> | Filter
< undefined
> output
< ▼ (3) [{...}, {...}, {...}] i
  ▶ 0: {a: 1}
  ▶ 1: {b: 2}
  ▶ 2: {c: 3}
    length: 3
  ▶ [[Prototype]]: Array(0)
```

Q4

The screenshot displays the developer tools interface of a web browser. The top navigation bar includes tabs for Elements, Console, Sources, Network, Performance, Memory, and Application. The Sources panel is active, showing a file named `index.js` with the following JavaScript code:

```
1  const input = [{ a: 1 }, { b: 2 }, { c: 3 }];
2
3  const output = input.reduce((result, obj) => {
4    for (const key in obj) {
5      if (obj.hasOwnProperty(key)) {
6        result[key] = obj[key];
7      }
8    }
9    return result;
10 }, {});
11
12 console.log(output);
13
14
```

Below the code editor, a status bar indicates the cursor is at `{ }` on Line 14, Column 1. The Console panel is also visible, showing a list of messages on the left and the corresponding log output on the right. The messages include 68 messages, 3 user messages, 21 errors, 45 warnings, 2 info messages, and no verbose messages. The log output shows three `undefined` values, followed by an `output` object, and then the object `{a: 1, b: 2, c: 3}`.