



**K4 CSF3133 REKA BENTUK ANTARA MUKA BERASASKAN WEB**

**SEM 3**

**LAB REPORT 6**

**PREPARED FOR:**

**ASSOC. PROF. TS. DR. WAN NURAL JWAHIR**

**LAB ASSISTANT:**

**MADAM WAN FATIN**

**PREPARED BY:**

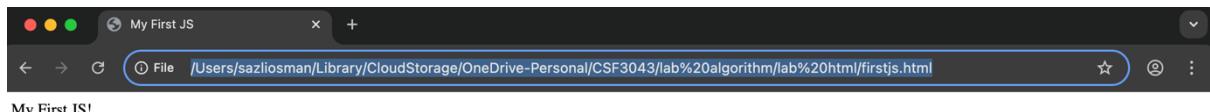
**NUR ALYA ADRIANA BINTI SAZLI**

**NO. MATRIC:**

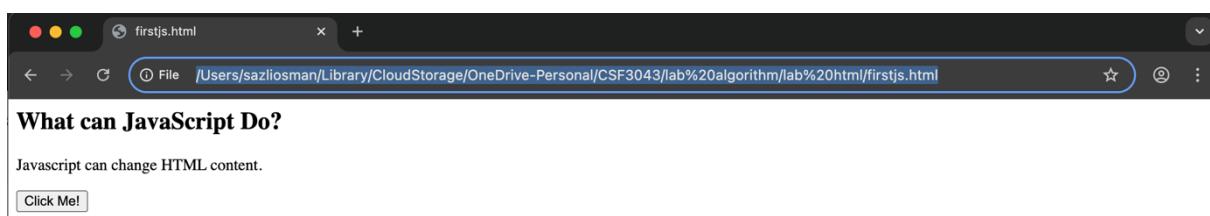
**S75898**

**SUBMIT BY: 6 DEC 2025**

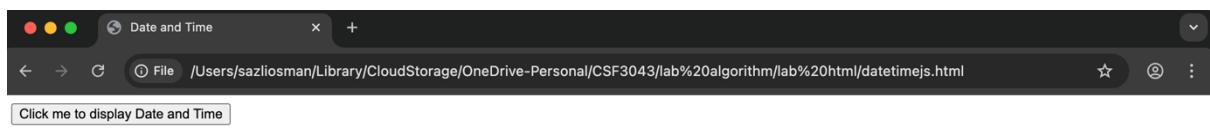
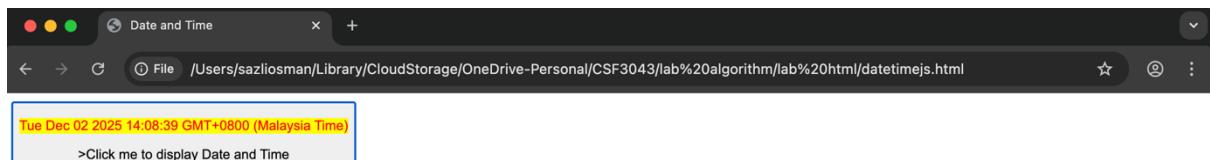
## 6.3 basic Javascript



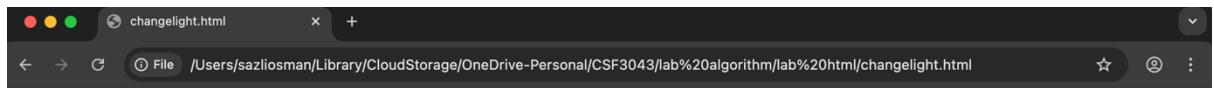
## 6.4 getElementById()



## 6.5 Date and Time



## 6.6 Change HTML Attributes Values



### What Can JavaScript \index{JavaScript} Do?

JavaScript \index{JavaScript} can change HTML attribute values.

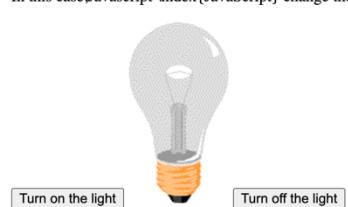
In this case, Javascript \index{JavaScript} change the value of the 'src'(source) attribute of an image.



### What Can JavaScript \index{JavaScript} Do?

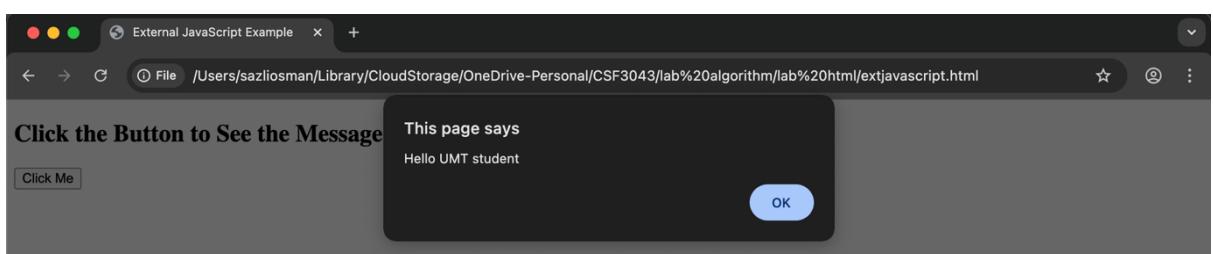
JavaScript \index{JavaScript} can change HTML attribute values.

In this case, Javascript \index{JavaScript} change the value of the 'src'(source) attribute of an image.

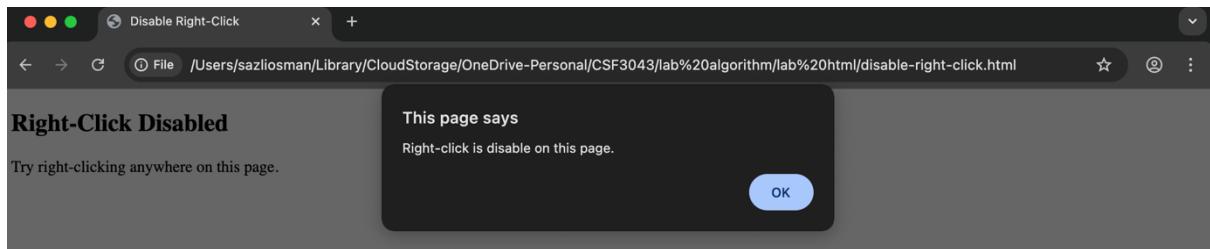


## 6.7 Link to file Javascript

### 6.7.3 External references



## 6.8 Working with events

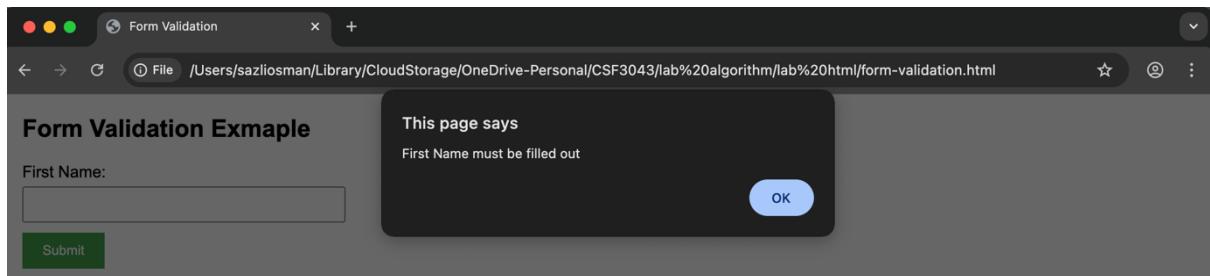


## 6.9 Multiplication Table

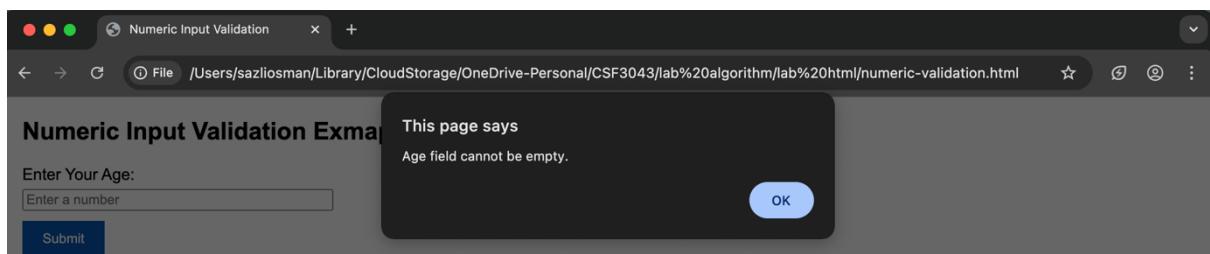


## 6.10 Client-side validation

### 6.10.1 Form Validation

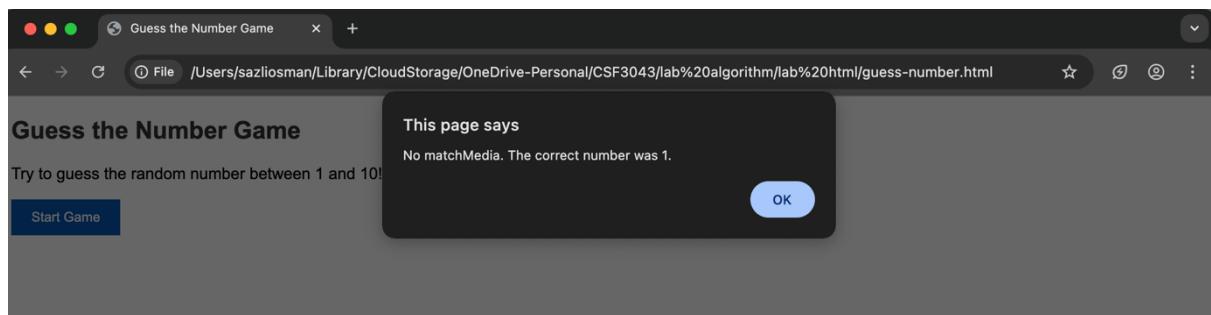
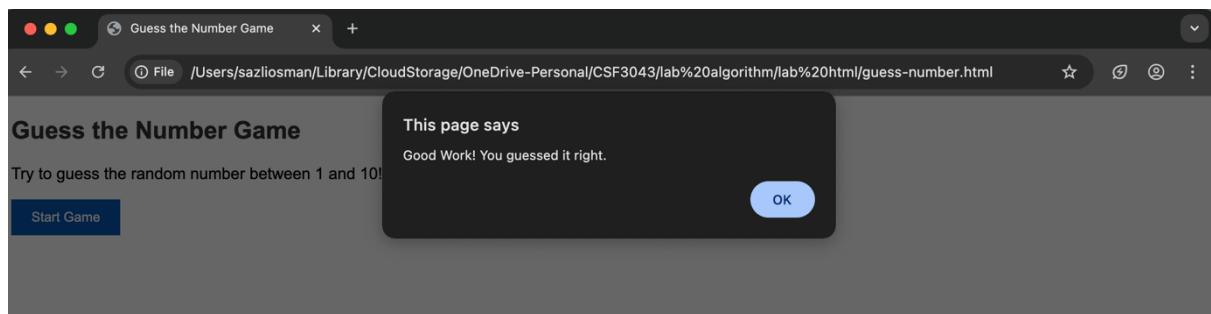
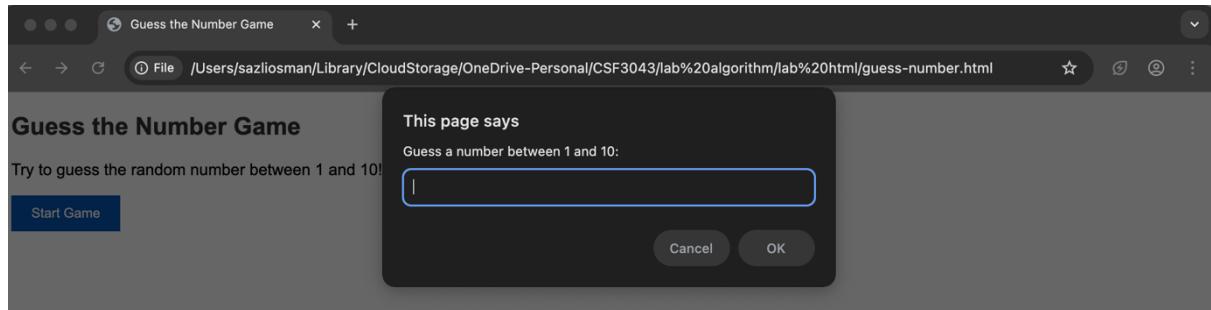


### 6.10.2 Numeric Input Validation

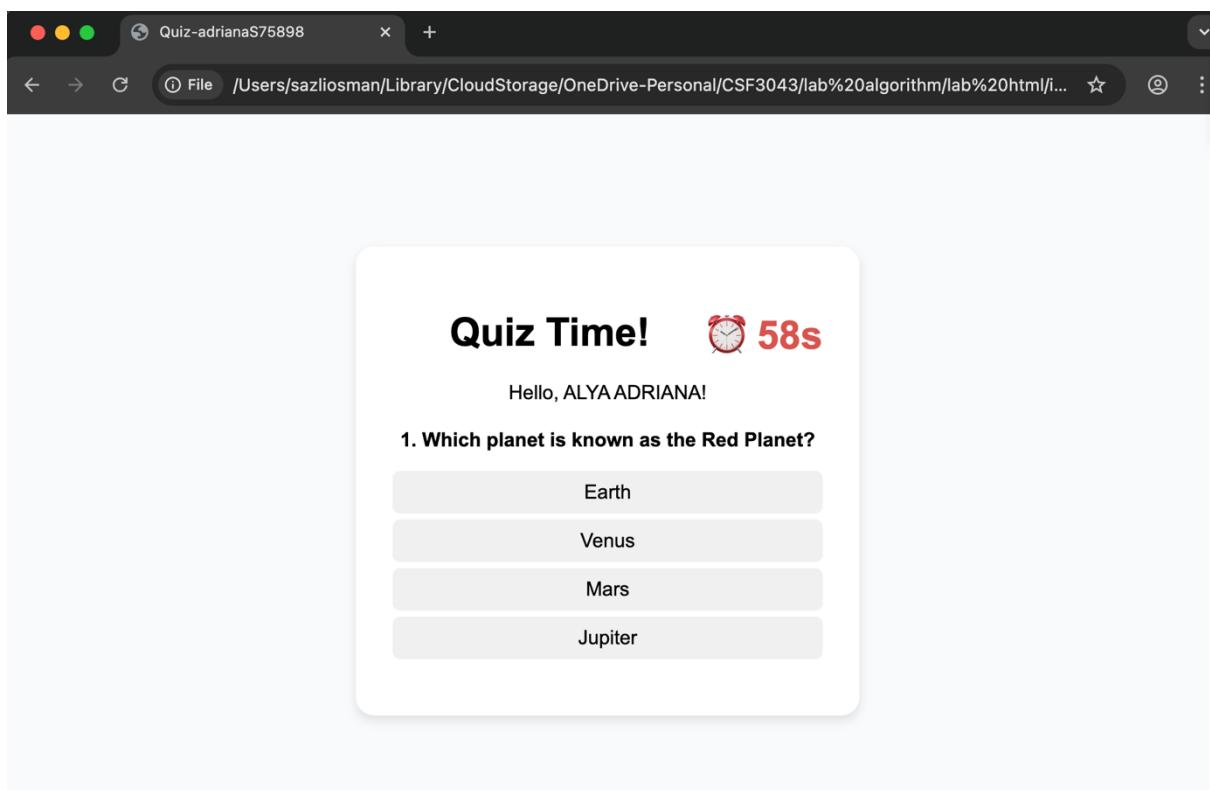
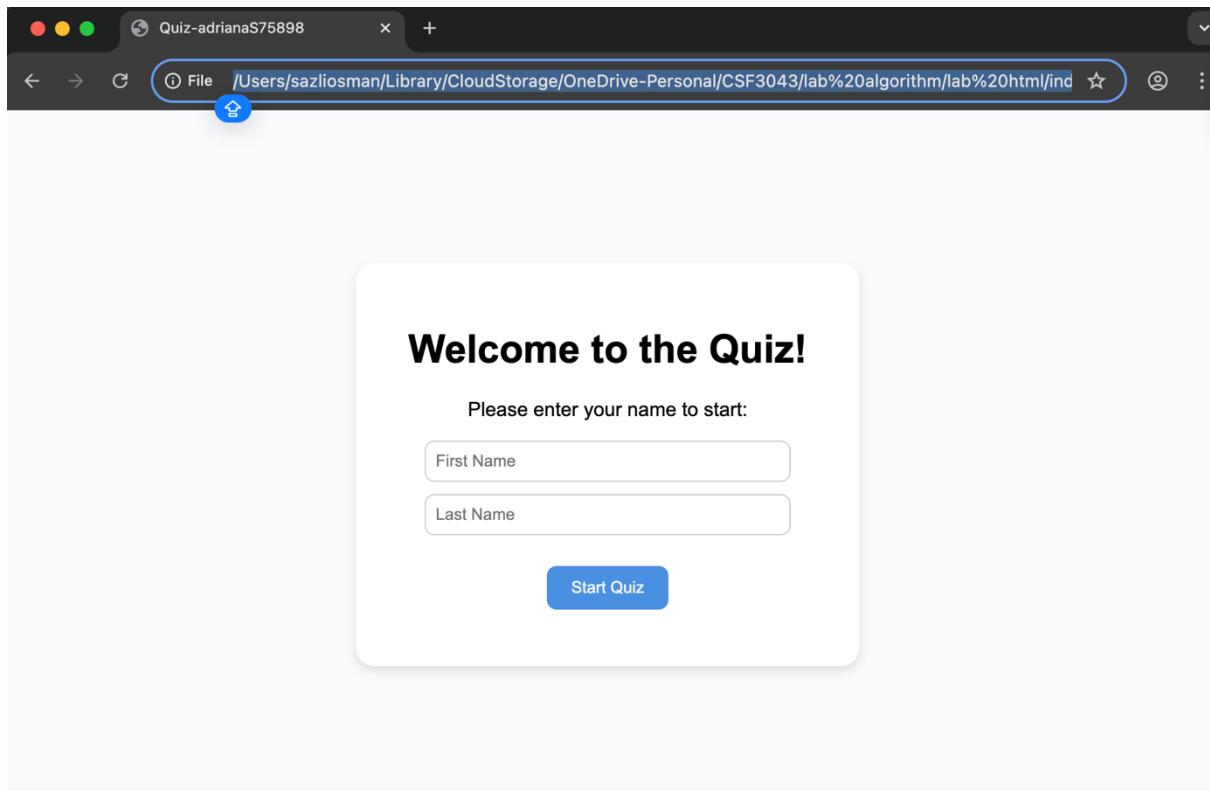


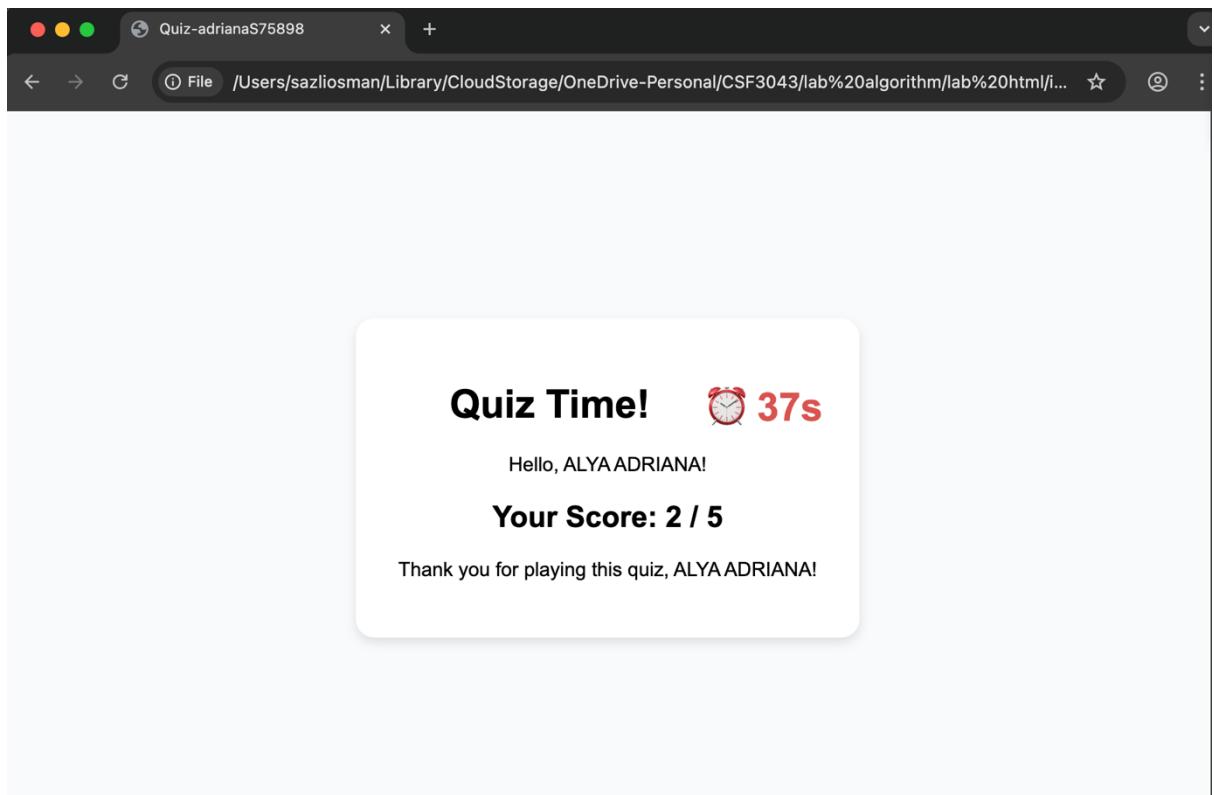
## 6.11 Math Function

### 6.11.1 Guess a Number

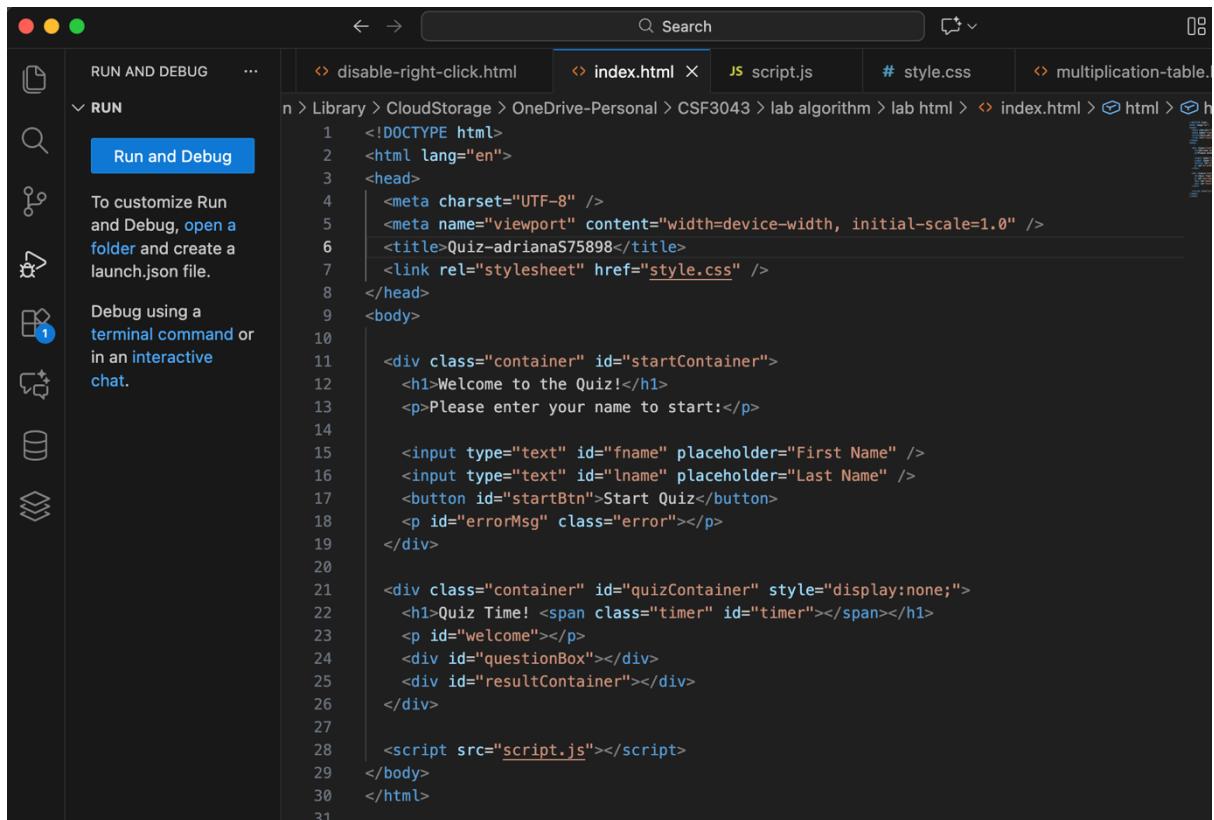


## Lab Exercise 6





## INDEX.HTML



The screenshot shows a code editor interface with a dark theme. On the left is a sidebar with various icons: a folder, a magnifying glass, a gear, a terminal, a file, and a stack of files. The main area has a title bar with tabs: 'disable-right-click.html', 'index.html', 'script.js', 'style.css', and 'multiplication-table.html'. Below the title bar is a search bar with the placeholder 'Search'. The main content area displays the following code:

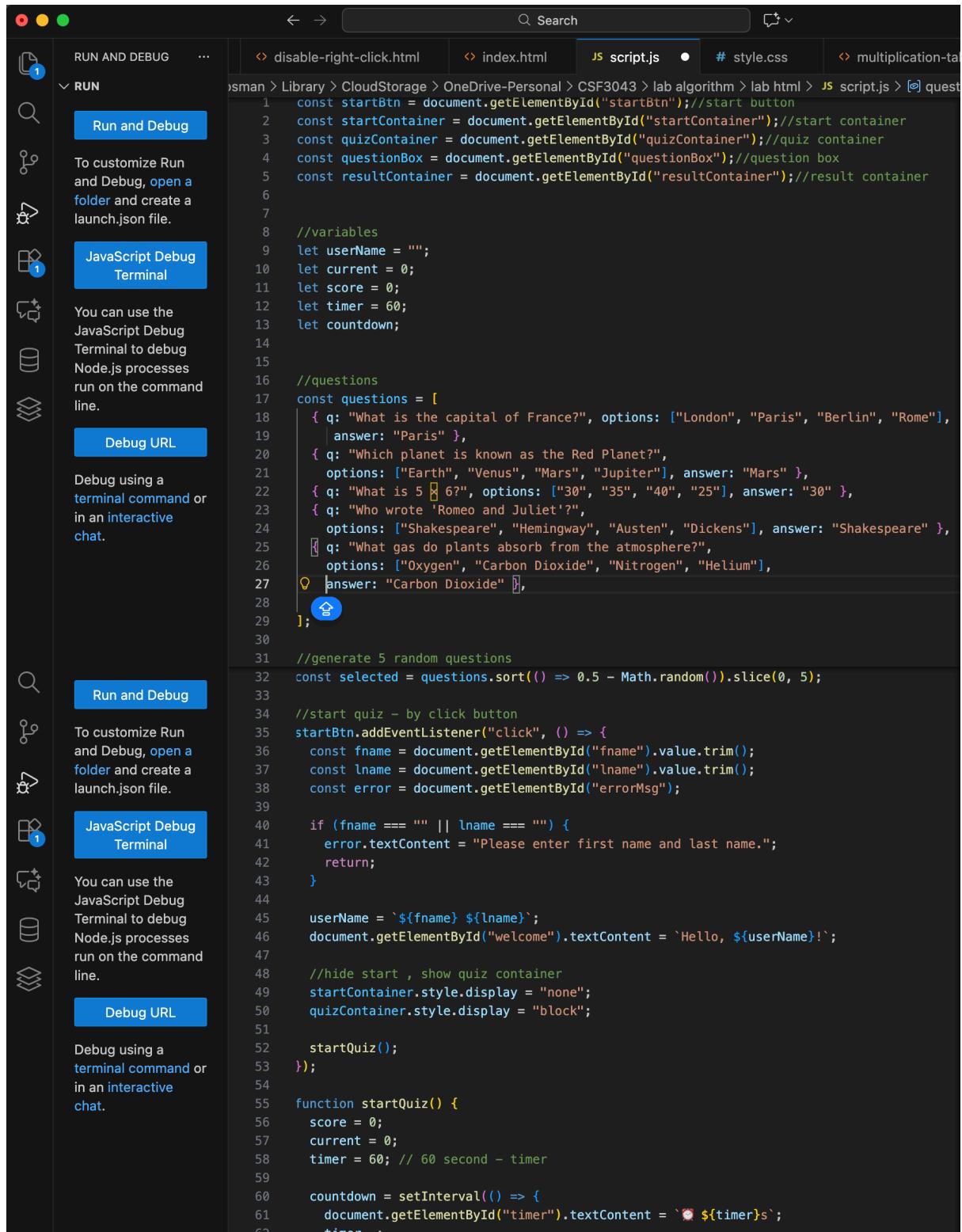
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Quiz-adrianaS75898</title>
<link rel="stylesheet" href="style.css" />
</head>
<body>

<div class="container" id="startContainer">
<h1>Welcome to the Quiz!</h1>
<p>Please enter your name to start:</p>
<input type="text" id="fname" placeholder="First Name" />
<input type="text" id="lname" placeholder="Last Name" />
<button id="startBtn">Start Quiz</button>
<p id="errorMsg" class="error"></p>
</div>

<div class="container" id="quizContainer" style="display:none;">
<h1>Quiz Time! <span class="timer" id="timer"></span></h1>
<p id="welcome"></p>
<div id="questionBox"></div>
<div id="resultContainer"></div>
</div>

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

## SCRIPT.JS



The screenshot shows a code editor interface with a sidebar on the left and a main content area on the right. The sidebar contains several sections with icons and buttons:

- RUN AND DEBUG** (with a gear icon)
- RUN** (with a play icon)
- Run and Debug** (button)
- To customize Run and Debug, open a folder and create a launch.json file.** (text)
- JavaScript Debug Terminal** (button)
- You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line.** (text)
- Debug URL** (button)
- Run and Debug** (button)
- To customize Run and Debug, open a folder and create a launch.json file.** (text)
- JavaScript Debug Terminal** (button)
- You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line.** (text)
- Debug URL** (button)

The main content area displays the `script.js` file with the following code:

```
const startBtn = document.getElementById("startBtn"); //start button
const startContainer = document.getElementById("startContainer"); //start container
const quizContainer = document.getElementById("quizContainer"); //quiz container
const questionBox = document.getElementById("questionBox"); //question box
const resultContainer = document.getElementById("resultContainer"); //result container

//variables
let userName = "";
let current = 0;
let score = 0;
let timer = 60;
let countdown;

//questions
const questions = [
  { q: "What is the capital of France?", options: ["London", "Paris", "Berlin", "Rome"], answer: "Paris" },
  { q: "Which planet is known as the Red Planet?", options: ["Earth", "Venus", "Mars", "Jupiter"], answer: "Mars" },
  { q: "What is 5 * 6?", options: ["30", "35", "40", "25"], answer: "30" },
  { q: "Who wrote 'Romeo and Juliet'?", options: ["Shakespeare", "Hemingway", "Austen", "Dickens"], answer: "Shakespeare" },
  { q: "What gas do plants absorb from the atmosphere?", options: ["Oxygen", "Carbon Dioxide", "Nitrogen", "Helium"], answer: "Carbon Dioxide" }
];
};

//generate 5 random questions
const selected = questions.sort(() => 0.5 - Math.random()).slice(0, 5);

//start quiz - by click button
startBtn.addEventListener("click", () => {
  const fname = document.getElementById("fname").value.trim();
  const lname = document.getElementById("lname").value.trim();
  const error = document.getElementById("errorMsg");

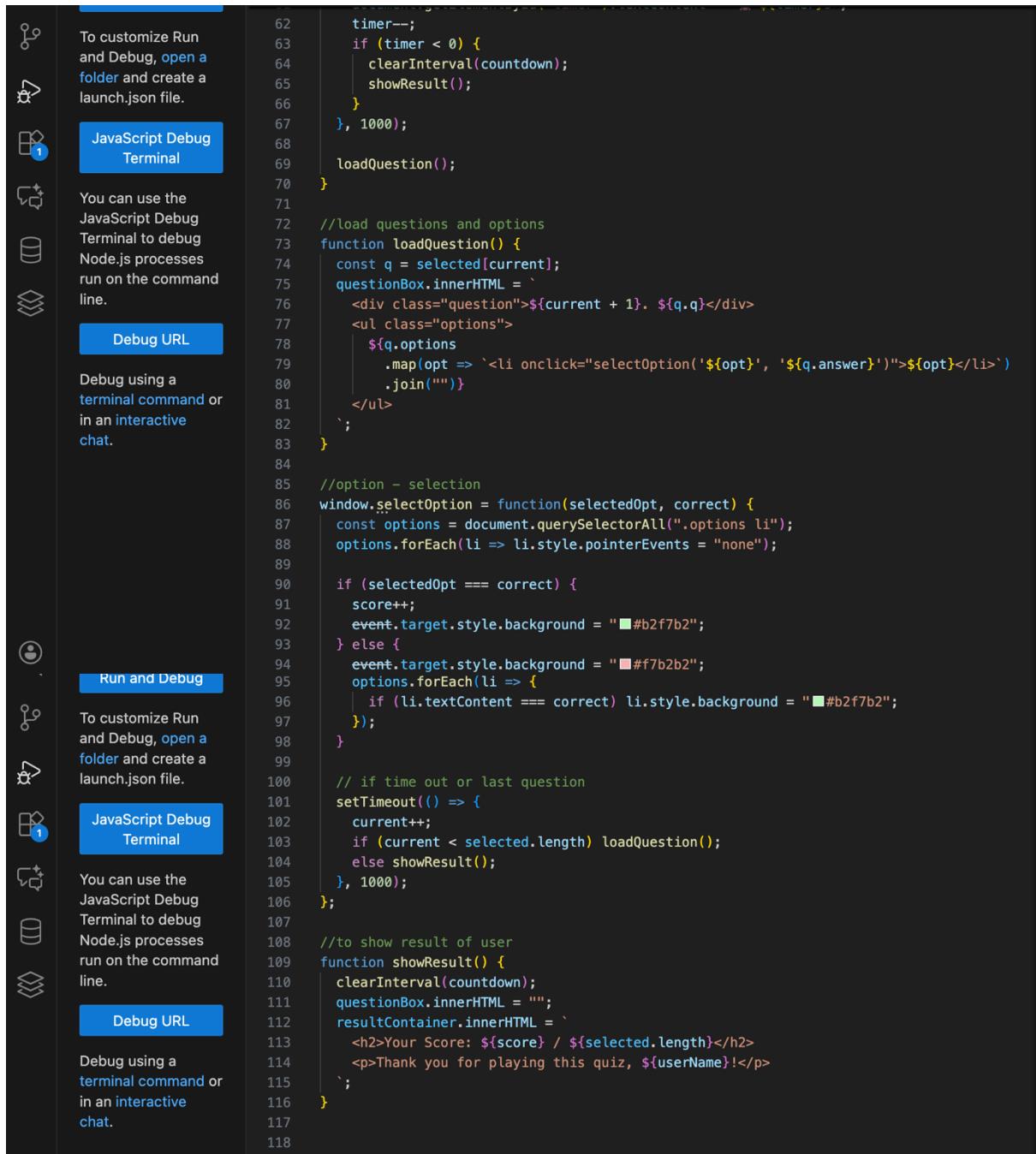
  if (fname === "" || lname === "") {
    error.textContent = "Please enter first name and last name.";
    return;
  }

  userName = `${fname} ${lname}`;
  document.getElementById("welcome").textContent = `Hello, ${userName}!`;

  //hide start, show quiz container
  startContainer.style.display = "none";
  quizContainer.style.display = "block";

  startQuiz();
});

function startQuiz() {
  score = 0;
  current = 0;
  timer = 60; // 60 second - timer
  countdown = setInterval(() => {
    document.getElementById("timer").textContent = `⌚ ${timer}s`;
    timer--;
  }, 1000);
}
```



To customize Run and Debug, open a folder and create a launch.json file.

JavaScript Debug Terminal

```
62 timer--;
63 if (timer < 0) {
64     clearInterval(countdown);
65     showResult();
66 }
67 }, 1000);

68 loadQuestion();
69 }

70 //load questions and options
71 function loadQuestion() {
72     const q = selected[current];
73     questionBox.innerHTML = `
74         <div class="question">${current + 1}. ${q.q}</div>
75         <ul class="options">
76             ${q.options}
77             .map(opt => `<li onclick="selectOption('${opt}', '${q.answer}')">${opt}</li>`)
78             .join('')
79         </ul>
80     `;
81 }
82 }

83 //option - selection
84 window.selectOption = function(selectedOpt, correct) {
85     const options = document.querySelectorAll(".options li");
86     options.forEach(li => li.style.pointerEvents = "none");
87
88     if (selectedOpt === correct) {
89         score++;
90         event.target.style.background = "#b2f7b2";
91     } else {
92         event.target.style.background = "#f7b2b2";
93         options.forEach(li => {
94             if (li.textContent === correct) li.style.background = "#b2f7b2";
95         });
96     }
97 }

98 // if time out or last question
99 setTimeout(() => {
100     current++;
101     if (current < selected.length) loadQuestion();
102     else showResult();
103 }, 1000);
104 }

105 //to show result of user
106 function showResult() {
107     clearInterval(countdown);
108     questionBox.innerHTML = "";
109     resultContainer.innerHTML = `
110         <h2>Your Score: ${score} / ${selected.length}</h2>
111         <p>Thank you for playing this quiz, ${userName}!</p>
112     `;
113 }
114
115 }
116 }

117

118 }
```

You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line.

Debug URL

Debug using a terminal command or in an interactive chat.

Run and Debug

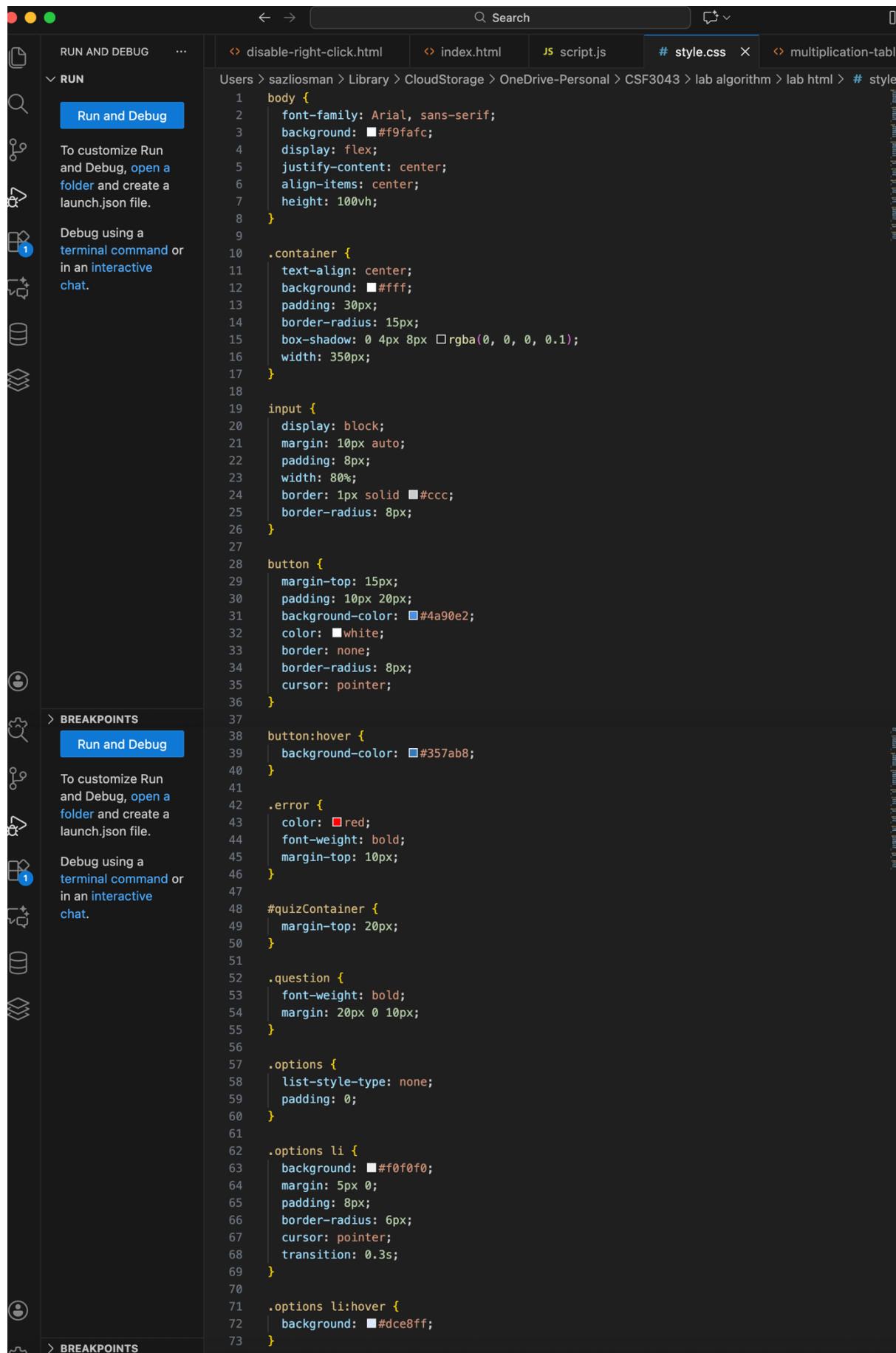
To customize Run and Debug, open a folder and create a launch.json file.

JavaScript Debug Terminal

You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line.

Debug URL

Debug using a terminal command or in an interactive chat.



Users > sazliosman > Library > CloudStorage > OneDrive-Personal > CSF3043 > lab algorithm > lab html > # style

```
1 body {  
2   font-family: Arial, sans-serif;  
3   background: #f9fafc;  
4   display: flex;  
5   justify-content: center;  
6   align-items: center;  
7   height: 100vh;  
8 }  
9  
10 .container {  
11   text-align: center;  
12   background: #ffff;  
13   padding: 30px;  
14   border-radius: 15px;  
15   box-shadow: 0 4px 8px #rgba(0, 0, 0, 0.1);  
16   width: 350px;  
17 }  
18  
19 input {  
20   display: block;  
21   margin: 10px auto;  
22   padding: 8px;  
23   width: 80%;  
24   border: 1px solid #ccc;  
25   border-radius: 8px;  
26 }  
27  
28 button {  
29   margin-top: 15px;  
30   padding: 10px 20px;  
31   background-color: #4a90e2;  
32   color: white;  
33   border: none;  
34   border-radius: 8px;  
35   cursor: pointer;  
36 }  
37  
38 button:hover {  
39   background-color: #357ab8;  
40 }  
41  
42 .error {  
43   color: red;  
44   font-weight: bold;  
45   margin-top: 10px;  
46 }  
47  
48 #quizContainer {  
49   margin-top: 20px;  
50 }  
51  
52 .question {  
53   font-weight: bold;  
54   margin: 20px 0 10px;  
55 }  
56  
57 .options {  
58   list-style-type: none;  
59   padding: 0;  
60 }  
61  
62 .options li {  
63   background: #f0f0f0;  
64   margin: 5px 0;  
65   padding: 8px;  
66   border-radius: 6px;  
67   cursor: pointer;  
68   transition: 0.3s;  
69 }  
70  
71 .options li:hover {  
72   background: #dce8ff;  
73 }
```

```
74
75  .timer {
76    font-weight: bold;
77    color: #d9534f;
78    float: right;
79  }
80
```