

Name: \_\_Racy Jhanna Ledres\_\_\_\_Date: \_\_November. 08, 2025\_\_  
Course and Section: \_\_BSIS-3A\_\_\_\_Score: \_\_\_\_\_

## Laboratory Activity 8:

### Instructions:

1. Use JavaScript to perform input/output and interactive quizzes.
  2. Manipulate HTML elements and respond to user events.
  3. Create an interactive quiz that asks the user a question and gives feedback based on their answer.
- 

### HTML (main.html):

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!--This is the title of the web page displayed in the browser tab-->
  <title>SheCodes</title>

  <!-- Links an external CSS file for styling the webpage-->
  <link rel="stylesheet" href="stylesheet.css">
</head>
<body>
  <!--Logo Section-->
  <div class="logo">
    SheCodes
  </div>

  <!-- Main container for the quiz application. this is the white part sa html page-->
  <div class="app">
    <h1>Web Development General Quiz</h1>
    <div class="quiz"> <!-- This container holds the question and answers -->
```

```

<h2 id="question">Question goes here</h2> <!-- Placeholder for displaying current questions-->
<div id="answer-buttons"> <!-- Container for answer buttons-->
  <!-- Each button represents the possible answer choice-->
  <button class="btn">Answer 1</button>
  <button class="btn">Answer 2</button>
  <button class="btn">Answer 3</button>
  <button class="btn">Answer 4</button>
</div>

<!--This is the button to move to the next question after answering-->
<button id="next-btn">Next</button>
</div>
</div>

<!--This is the background container. The gridline pattern background -->
<div class="bgcontainer"></div>
<script src="quizscript.js"></script>
</body>
</html>

```

## JavaScript (quizscript.js)

```

// This is the Quiz Question and Answer Data
// each questions has an array of possible answers
// with a 'correct' property to indicate the right answer
// Each answer contains the answer text and whether it is correct or not
const questions = [
  {
    question: "What does HTML stand for?",
    answers: [
      { text: "Hyper Text Markup Language", correct: true },
      { text: "High Text Machine Language", correct: false },
      { text: "Hyperlink Text Mark Language", correct: false },
    ]
  }
]

```

```
{ text: "Home Tool Markup Language", correct: false }  
]  
},
```

```
{  
  question: "Which language is used for styling web pages?",  
  answers: [  
    { text: "HTML", correct: false },  
    { text: "jQuery", correct: false },  
    { text: "CSS", correct: true },  
    { text: "Python", correct: false }  
  ]  
},
```

```
{  
  question: "Which is a JavaScript framework?",  
  answers: [  
    { text: "React", correct: true },  
    { text: "Laravel", correct: false },  
    { text: "Django", correct: false },  
    { text: "Flask", correct: false }  
  ]  
},
```

```
{  
  question: "Which method is used to log messages to the console in JavaScript?",  
  answers: [  
    { text: "console.log()", correct: true },  
    { text: "print()", correct: false },  
    { text: "echo()", correct: false },  
    { text: "write()", correct: false }  
  ]  
}
```

```
]
},

{
  question: "Which symbol is used for comments in JavaScript?",
  answers: [
    { text: "//", correct: true },
    { text: "/*", correct: false },
    { text: "#", correct: false },
    { text: "<!-- -->", correct: false }
  ]
},
```

```
{
  question: "What does CSS stand for?",
  answers: [
    { text: "Cascading Style Sheets", correct: true },
    { text: "Colorful Style Sheets", correct: false },
    { text: "Creative Style System", correct: false },
    { text: "Computer Style Sheets", correct: false }
  ]
}
```

```
];
```

```
//the DOM Element References
```

```
//Grabs HTML elements by their IDs to manipulate them later
const questionElement = document.getElementById("question");
const answerButtonsElement = document.getElementById("answer-buttons");
const nextButton = document.getElementById("next-btn");
```

```
//Quiz state variables
```

```
//Tracks which question the user is on and their score
```

```
let currentQuestionIndex = 0;
```

```
let score = 0;
```

```
//Start Quiz function. This function initializes the quiz
```

```
//by resetting the question index and score, updating the next button text,
```

```
//and displaying the first question.
```

```
function startQuiz() {
```

```
    currentQuestionIndex = 0;
```

```
    score = 0;
```

```
    nextButton.innerHTML = "Next";
```

```
    showQuestion();
```

```
};
```

```
// Display Question function. This function displays the current question
```

```
// and its possible answers. It resets the state, sets the question text,
```

```
// creates buttons for each answer, and adds event listeners to handle answer selection.
```

```
function showQuestion() {
```

```
    //Clear previous question and answers
```

```
    resetState();
```

```
    //Get current question based on the index
```

```
    let currentQuestion = questions[currentQuestionIndex];
```

```
    //Display question number and text
```

```
    let questionNo = currentQuestionIndex + 1;
```

```
    questionElement.innerHTML = questionNo + ". " + currentQuestion.question;
```

```

//Create and displays buttons for each answer
currentQuestion.answers.forEach(answer => {
  const button = document.createElement("button");
  button.innerHTML = answer.text; // Set button text to answer text
  button.classList.add("btn"); // Add CSS class for styling
  answerButtonsElement.appendChild(button); // Add button to the answer buttons container

  // Store whether the answer is correct in a data attribute
  if (answer.correct) {
    button.dataset.correct = answer.correct;
  }

  // Add click event listener to check the answer when clicked
  button.addEventListener("click", selectAnswer);
});

};

//Reset State function. This function clears the previous question and answers
function resetState() {
  nextButton.style.display = "none"; // Hide the next button
  //Remove all answer buttons
  while (answerButtonsElement.firstChild) {
    answerButtonsElement.removeChild(answerButtonsElement.firstChild);
  }
}

//Select Answer function. This function handles the logic when an answer is selected
function selectAnswer(e){
  const selectedButton = e.target; // the button that was clicked

```

```
const isCorrect =selectedButton.dataset.correct === "true"; // eto yung checker if the answer is correct or not
```

```
// This handles the visual feedback and score updating based on the selected answer
```

```
if (isCorrect) {  
  selectedButton.classList.add("correct");  
  score++; // So if the answer is correct, increase the score by 1  
} else {  
  selectedButton.classList.add("wrong");  
}
```

```
//Highlight the correct answer and disable all buttons after an answer is selected
```

```
Array.from(answerButtonsElement.children).forEach(button => {  
  if (button.dataset.correct === "true") {  
    button.classList.add("correct");  
  }  
  button.disabled = true; // this disables all buttons and prevent further clicks  
});
```

```
//Show the next button after an answer is selected
```

```
nextButton.style.display = "block";  
}
```

```
//Show Score function. This function displays the user's final score
```

```
function showScore() {  
  resetState();  
  questionElement.innerHTML = `You scored ${score} out of ${questions.length}!`;   
  nextButton.innerHTML = "Play Again"; // Changes the next button text to "Play Again"  
  nextButton.style.display = "block"; // Shows the "Play Again" button  
}
```

```
// Handle Next Button function. This function manages the transition to the next question or the end of the quiz
```

```
function handleNextButton() {
  currentQuestionIndex++;
  if (currentQuestionIndex < questions.length) {
    // Show the next question if there are more questions
    showQuestion();
  } else { // there's no more questions, then it will display the final score
    showScore();
  }
}
```

// next Button Event Listener

// This listens for clicks on the next button and either moves to the next question or restarts the quiz

```
nextButton.addEventListener("click", () => {
  if(currentQuestionIndex < questions.length) {
    handleNextButton();
  } else {
    startQuiz();
  }
});
```

//Start the quiz when the page loads

```
startQuiz();
```

**Be guided with the following Rubrics:**

Criteria	Points	Notes	Score
<b>Functionality</b>	20	Quiz works correctly	
<b>Use of JS Syntax</b>	20	Proper use of variables and events	
<b>DOM Manipulation</b>	20	Feedback updates dynamically	
<b>Creativity</b>	20	Engaging question and design	
<b>Code Comments &amp; Clarity</b>	20	Well commented and organized code	