Lesson-End Project

Developing a Web-based JavaScript Quiz Application

Project agenda: To develop a web-based JavaScript Quiz Application that tests users' knowledge of JavaScript concepts through interactive and timed multiple-choice questions. The project focuses on the setup of a coding environment using Visual Studio Code, design and implementation of a user-friendly interface with HTML and CSS, and the development of quiz functionality using JavaScript. This initiative aims to enhance programming skills in JavaScript, HTML, and CSS while also providing a practical application that can assess JavaScript knowledge.

Description: As a developer, your current project involves creating an engaging and educational tool for testing JavaScript knowledge. The goal is to build an interactive, web-based quiz application that not only serves as a learning platform but also to solidify your front-end development skills. This project is structured to implement a responsive user interface, manage quiz content dynamically, and handle user interactions effectively. By completing this project, you will improve your capabilities in web development and gain insights into effective JavaScript programming practices.

Tools required: Visual Studio Code

Prerequisites: None

Expected deliverables: A fully functional JavaScript Quiz Application with features like timed questions, answer verification, score calculation, and a final score display.

Steps to be followed:
1. Set up and configure the project
2. Build the quiz interface and implement functionality
3. Launch the application

Step 1: Set up and configure the project

1.1 Create a new project folder named **quiz-app** and open it in VS Code
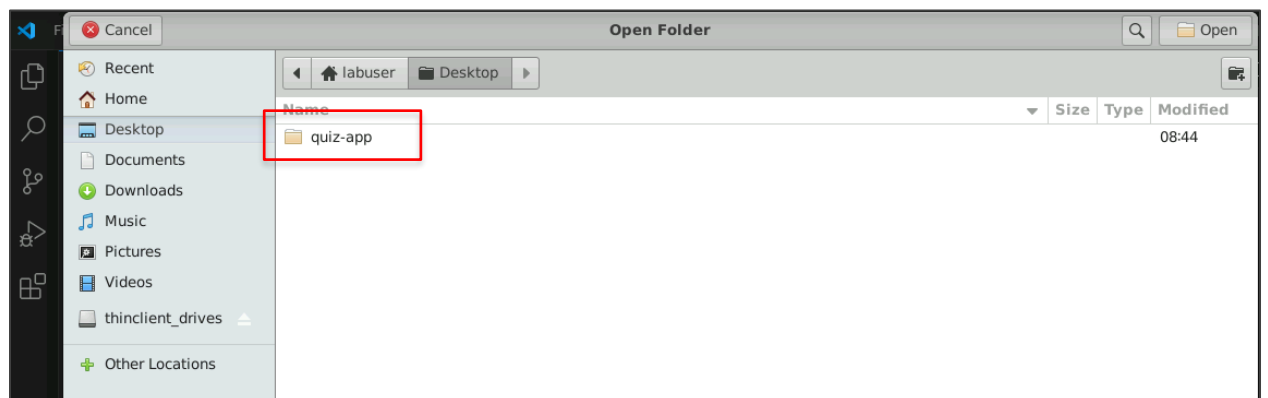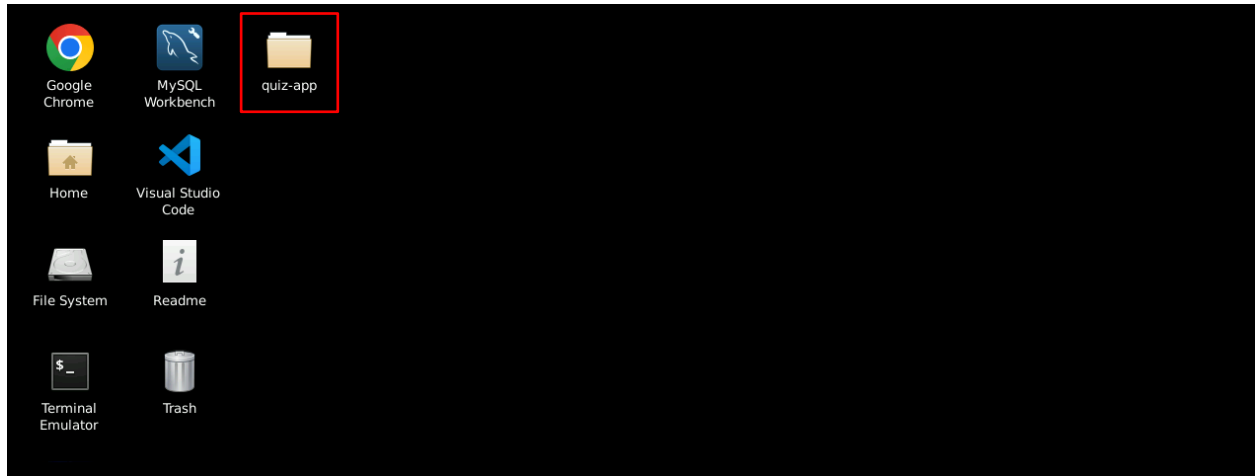
1.2 Create the essential files **index.html** for the main structure, **styles.css** for styling, **questions.js** for storing quiz questions, and **script.js** for quiz logic
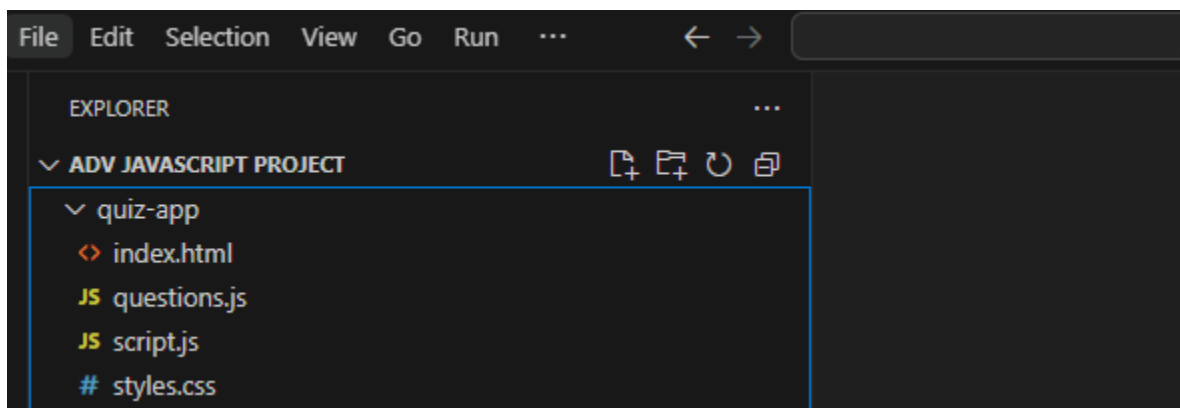
Step 2: Build the quiz interface and implement functionality

2.1 Add the following code in **index.html** file:

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Quiz App</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <div class="container mt-5 text-center">
    <div class="quiz-container p-4">
      <h2 class="mb-4">JavaScript Quiz</h2>
      <div id="question" class="mb-3"></div>
      <div id="options" class="d-grid gap-2"></div>
      <p id="timer" class="mt-3 text-danger">Time Left: 10s</p>
      <button id="next-btn" class="btn btn-primary mt-3" onclick="nextQuestion()">Next</button>
      <p id="score" class="mt-3"></p>
    </div>
  </div>
  <script src="questions.js"></script>
  <script src="script.js"></script>
</body>
</html>
```

```html
index.html ×
quiz-app > <> index.html > ⊘ html > ⊘ body > ⊘ script
  1    <!DOCTYPE html>
  2    <html lang="en">
  3    <head>
  4        <meta charset="UTF-8">
  5        <meta name="viewport" content="width=device-width, initial-scale=1.0">
  6        <title>Quiz App</title>
  7        <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
  8        <link rel="stylesheet" href="styles.css">
  9    </head>
 10    <body>
 11        <div class="container mt-5 text-center">
 12            <div class="quiz-container p-4">
 13                <h2 class="mb-4">JavaScript Quiz</h2>
 14                <div id="question" class="mb-3"></div>
 15                <div id="options" class="d-grid gap-2"></div>
 16                <p id="timer" class="mt-3 text-danger">Time Left: 10s</p>
 17                <button id="next-btn" class="btn btn-primary mt-3" onclick="nextQuestion()">Next</button>
 18                <p id="score" class="mt-3"></p>
 19            </div>
 20        </div>
 21        <script src="questions.js"></script>
 22        <script src="script.js"></script>
 23    </body>
 24    </html>
 25
```

2.2 Add the following code in **style.css** file

```css
body {
    background-color: #f8f9fa;
}
.quiz-container {
    max-width: 600px;
    margin: auto;
    background: white;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
}
button {
    width: 100%;
}
```

```css
# styles.css  ×

quiz-app > # styles.css > ...
    1    body {
    2        background-color: ■#f8f9fa;
    3    }
    4
    5    .quiz-container {
    6        max-width: 600px;
    7        margin: auto;
    8        background: ■white;
    9        padding: 20px;
   10        border-radius: 10px;
   11        box-shadow: 0px 0px 10px □rgba(0, 0, 0, 0.1);
   12    }
   13
   14    button {
   15        width: 100%;
   16    }
   17
```

2.3 Add the following code in **script.js** file

```javascript
let currentQuestionIndex = 0;
let score = 0;
let timer;
let timeLeft = 10;
function startTimer() {
   timeLeft = 10;
   document.getElementById("timer").innerText = `Time Left: ${timeLeft}s`;
   clearInterval(timer);
   timer = setInterval(() => {
      timeLeft--;
      document.getElementById("timer").innerText = `Time Left: ${timeLeft}s`;
      if (timeLeft === 0) {
         clearInterval(timer);
         nextQuestion();
      }
   }, 1000);
}
function loadQuestion() {
   const currentQuestion = questions[currentQuestionIndex];
   document.getElementById("question").innerText = currentQuestion.question;
   const optionsContainer = document.getElementById("options");
   optionsContainer.innerHTML = "";
   currentQuestion.options.forEach(option => {
      const button = document.createElement("button");
```
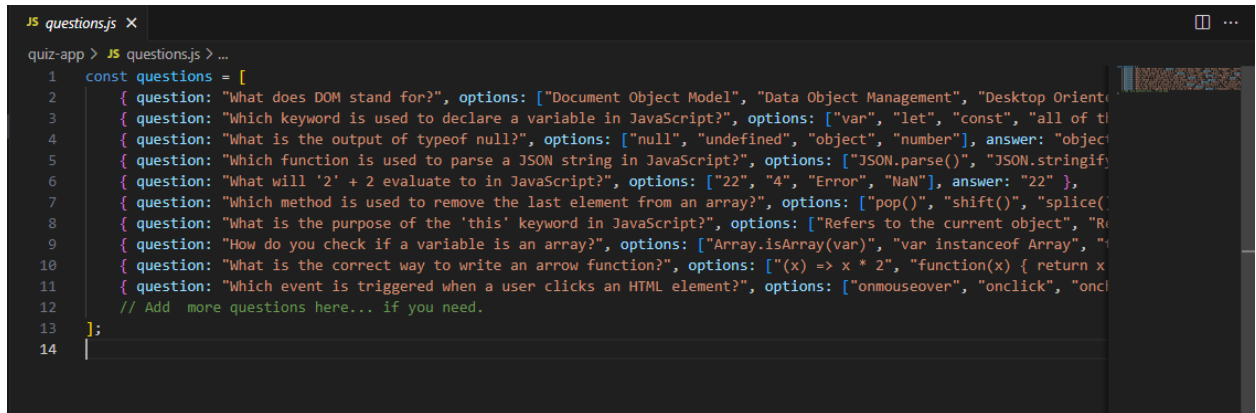
```javascript
        button.className = "btn btn-outline-primary";
        button.innerText = option;
        button.onclick = () => checkAnswer(option);
        optionsContainer.appendChild(button);
    });
    startTimer();
}
function checkAnswer(selected) {
    const correct = questions[currentQuestionIndex].answer;
    if (selected === correct) {
        score++;
    }
    nextQuestion();
}
function nextQuestion() {
    currentQuestionIndex++;
    if (currentQuestionIndex < questions.length) {
        loadQuestion();
    } else {
        endQuiz();
    }
}
function endQuiz() {
    clearInterval(timer);
    document.getElementById("question").innerText = "Quiz Completed!";
    document.getElementById("options").innerHTML = "";
    document.getElementById("score").innerText = `Your Score: ${score} /
${questions.length}`;
    document.getElementById("next-btn").style.display = "none";
}
loadQuestion();
```

```js
JS script.js    ✕

quiz-app ›  JS script.js › ⊕ loadQuestion
  1    let currentQuestionIndex = 0;
  2    let score = 0;
  3    let timer;
  4    let timeLeft = 10;
  5
  6    function startTimer() {
  7        timeLeft  function setInterval(handler: TimerHandler, timeout?: number, ...arguments: any[]): number
  8        document
  9        clearInt  MDN Reference
 10        timer = setInterval(() => {
 11            timeLeft--;
 12            document.getElementById("timer").innerText = `Time Left: ${timeLeft}s`;
 13            if (timeLeft === 0) {
 14                clearInterval(timer);
 15                nextQuestion();
 16            }
 17        }, 1000);
 18    }
 19
 20    function loadQuestion() {
 21        const currentQuestion = questions[currentQuestionIndex];
 22        document.getElementById("question").innerText = currentQuestion.question;
 23        const optionsContainer = document.getElementById("options");
 24        optionsContainer.innerHTML = "";
 25        currentQuestion.options.forEach(option => {
```

2.4 Add the following code in **questions.js** file

const questions = [
**   { question: "What does DOM stand for?", options: ["Document Object Model",**
"Data Object Management", "Desktop Oriented Mode", "Digital Ordinance Model"],
answer: "Document Object Model" },
**   { question: "Which keyword is used to declare a variable in JavaScript?", options:**
["var", "let", "const", "all of the above"], answer: "all of the above" },
**   { question: "What is the output of typeof null?", options: ["null", "undefined",**
"object", "number"], answer: "object" },
**   { question: "Which function is used to parse a JSON string in JavaScript?", options:**
["JSON.parse()", "JSON.stringify()", "JSON.toObject()", "JSON.decode()"], answer:
"JSON.parse()" },
**   { question: "What will '2' + 2 evaluate to in JavaScript?", options: ["22", "4", "Error",**
"NaN"], answer: "22" },
**   { question: "Which method is used to remove the last element from an array?",**
options: ["pop()", "shift()", "splice()", "remove()"], answer: "pop()" },
**   { question: "What is the purpose of the 'this' keyword in JavaScript?", options:**
["Refers to the current object", "Refers to global scope", "Refers to parent function",
"Refers to new instance"], answer: "Refers to the current object" },
**   { question: "How do you check if a variable is an array?", options:**
["Array.isArray(var)", "var instanceof Array", "typeof var == 'array'", "Both A and B"],
answer: "Both A and B" },

{ question: "What is the correct way to write an arrow function?", options: ["(x) => x * 2", "function(x) { return x * 2; }", "x -> x * 2", "(x) -> { x * 2 }"], answer: "(x) => x * 2" },

{ question: "Which event is triggered when a user clicks an HTML element?", options: ["onmouseover", "onclick", "onchange", "onhover"], answer: "onclick" },
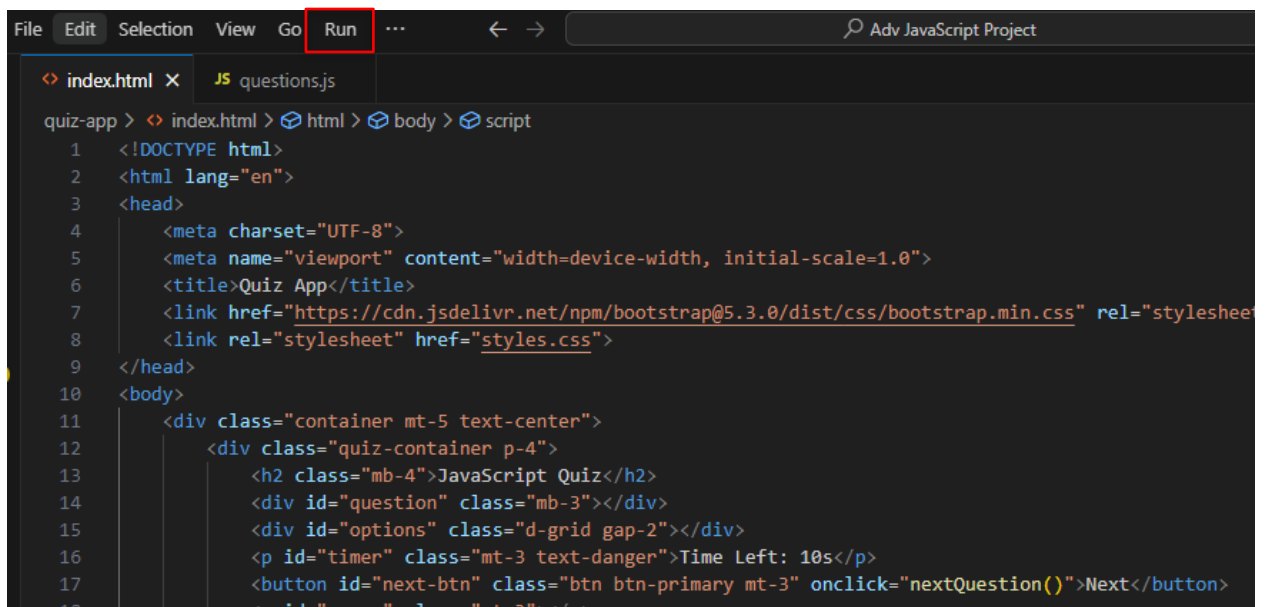
// Add more questions here... if you need.

];

Step 3: Launch the application

3.1 Execute the **index.html** file by clicking **Run** in your development environment or code editor

You will observe the following result:

By following the above steps, you have successfully created and launched a dynamic web-based JavaScript Quiz Application that tests users on various JavaScript concepts, offering an interactive and educational experience.