



Intern Name: Fatima Wajid

Submitted to: Ali Hyder

Internship Role: Web Development

1. Introduction

Today, we worked on a mini project to practically apply the JavaScript concepts we learned earlier in the internship.

I decided to build a Quiz App that uses LocalStorage to store high scores.

The project helped me revise key topics such as:

- HTML & CSS structure
- JavaScript event handling
- JSON data storage
- LocalStorage API

As an alternative, I also reviewed how a Note App could be created using LocalStorage.

2. Objectives of the Project

- To create an interactive quiz with multiple-choice questions.
- To store and retrieve data from LocalStorage.

- To improve my problem-solving and coding skills.
- To prepare for real-world scenarios where user data needs to be saved locally.

3. Tools & Technologies Used

- HTML → for structure
- CSS → for styling
- JavaScript (ES6+) → for logic
- LocalStorage → for storing scores
- JSON → for data handling

4. Project Features (Quiz App)

1. Displays one question at a time.
2. Allows the user to select an answer.
3. Calculates the score at the end.
4. Saves the highest score in LocalStorage.
5. Displays saved high score when the app is reopened.

HTML Structure:

```
<div id="quiz-container">
```

```
  <h2 id="question"></h2>
```

```
  <div id="options"></div>
```

```
  <button id="next-btn">Next</button>
```

```
<p id="score"></p>
```

```
</div>
```

JavaScript Logic:

```
let questions = [  
  { question: "What is the capital of Pakistan?", options: ["Lahore", "Islamabad", "Karachi"],  
    answer: "Islamabad" },  
  { question: "2 + 2 = ?", options: ["3", "4", "5"], answer: "4" }  
];  
  
let currentQuestion = 0;  
  
let score = 0;  
  
function loadQuestion() {  
  let q = questions[currentQuestion];  
  
  document.getElementById("question").innerText = q.question;  
  
  let optionsHtml = q.options.map(opt => `<button  
onclick="checkAnswer('${opt}')">${opt}</button>`).join("");  
  
  document.getElementById("options").innerHTML = optionsHtml;  
}
```

```
function checkAnswer(selected) {  
    if (selected === questions[currentQuestion].answer) {  
        score++;  
    }  
    currentQuestion++;  
    if (currentQuestion < questions.length) {  
        loadQuestion();  
    } else {  
        let highScore = localStorage.getItem("highScore") || 0;  
        if (score > highScore) {  
            localStorage.setItem("highScore", score);  
        }  
        document.getElementById("score").innerText = `Your score: ${score} | High score: ${localStorage.getItem("highScore")}`;  
    }  
}  
  
loadQuestion();
```

Full Html Code:

```
<!DOCTYPE html>

<html>

<head>

  <meta charset="UTF-8">

  <title>Web Development Quiz</title>

  <style>

    body {

      font-family: Arial, sans-serif;

      background: #f5f7fa;

      display: flex;

      justify-content: center;

      align-items: center;

      height: 100vh;

      margin: 0;

    }

    #quiz-container {

      background: #ffffff;
```

```
padding: 20px;
border-radius: 10px;
width: 450px;
box-shadow: 0px 4px 15px rgba(0,0,0,0.1);
text-align: center;
}
h2 {
    color: #2c3e50;
}
.option-btn {
    display: block;
    width: 100%;
    padding: 10px;
    margin: 5px 0;
    cursor: pointer;
    background-color: #dfe6e9;
    color: #2d3436;
    border: none;
```

```
border-radius: 5px;
transition: background 0.3s;
font-size: 14px;
}
.option-btn:hover {
background-color: #b2bec3;
}
.nav-btn {
padding: 8px 15px;
margin: 10px 5px;
cursor: pointer;
background-color: #74b9ff;
color: white;
border: none;
border-radius: 5px;
transition: background 0.3s;
}
.nav-btn:hover {
```

```
        background-color: #0984e3;
    }
    #restart-btn {
        background-color: #ff7675;
    }
    #restart-btn:hover {
        background-color: #d63031;
    }
    #score {
        margin-top: 15px;
        font-weight: bold;
        color: #2d3436;
    }
</style>
</head>
<body>

<div id="quiz-container">
```



```
<h2 id="question"></h2>
```

```
<div id="options"></div>
```

```
<div>
```

```
  <button class="nav-btn" onclick="prevQuestion()">Previous</button>
```

```
  <button class="nav-btn" onclick="nextQuestion()">Next</button>
```

```
</div>
```

```
  <button id="restart-btn" class="nav-btn"  
onclick="restartQuiz()">Restart</button>
```

```
  <p id="score"></p>
```

```
</div>
```

```
<script>
```

```
let questions = [
```

```
  { question: "What does JSON stand for?", options: ["JavaScript Object  
Notation", "Java Syntax Object Name", "JavaScript Output Name"], answer:  
"JavaScript Object Notation" },
```

```
  { question: "Which method stores data permanently in the browser?", options:  
["sessionStorage", "localStorage", "tempStorage"], answer: "localStorage" },
```

```
  { question: "In ES6, which symbol is used for template literals?", options:  
["Single quotes", "Double quotes", "Backticks"], answer: "Backticks" },
```

```
{ question: "What does the spread operator (...) do in JavaScript?", options:
["Copies or expands elements", "Pauses execution", "Merges functions"],
answer: "Copies or expands elements" },
```

```
{ question: "Which function is used to run code after a delay?", options:
["setTimeout()", "setInterval()", "delayFunction()"], answer: "setTimeout()" },
```

```
{ question: "Which HTML tag is used to link an external CSS file?", options:
["<link>", "<style>", "<css>"], answer: "<link>" },
```

```
{ question: "Which CSS property changes the text color?", options: ["font-
color", "color", "text-color"], answer: "color" },
```

```
{ question: "In HTML, which attribute is used for unique element
identification?", options: ["class", "id", "name"], answer: "id" },
```

```
{ question: "Which JavaScript method converts an object to JSON string?",
options: ["JSON.parse()", "JSON.stringify()", "toJSON()"], answer:
"JSON.stringify()" },
```

```
{ question: "Which asynchronous feature handles future values in JavaScript?",
options: ["Callbacks", "Promises", "Loops"], answer: "Promises" }
```

```
];
```

```
let currentQuestion = 0;
```

```
let score = 0;
```

```
let userAnswers = Array(questions.length).fill(null);
```

```
function loadQuestion() {  
    let q = questions[currentQuestion];  
  
    document.getElementById("question").innerText = `${currentQuestion + 1}.  
    ${q.question}`;  
  
    let optionsHtml = q.options.map(opt =>  
        `onclick="selectAnswer('${opt}')">${opt}</button>`  
    ).join("");  
  
    document.getElementById("options").innerHTML = optionsHtml;  
}
```

```
function selectAnswer(selected) {  
    userAnswers[currentQuestion] = selected;  
}
```

```
function nextQuestion() {  
    if (currentQuestion < questions.length - 1) {  
        currentQuestion++;  
    }
```

```
    loadQuestion();  
  } else {  
    calculateScore();  
  }  
}
```

```
function prevQuestion() {  
  if (currentQuestion > 0) {  
    currentQuestion--;  
    loadQuestion();  
  }  
}
```

```
function calculateScore() {  
  score = 0;  
  for (let i = 0; i < questions.length; i++) {  
    if (userAnswers[i] === questions[i].answer) {  
      score++;  
    }  
  }  
}
```

```
}  
}
```

```
let highScore = localStorage.getItem("highScore") || 0;  
if (score > highScore) {  
  localStorage.setItem("highScore", score);  
}
```

```
document.getElementById("score").innerText =  
  `Your score: ${score}/${questions.length} | High score:  
  ${localStorage.getItem("highScore")}`;   
document.getElementById("question").innerText = "Quiz Finished!";  
document.getElementById("options").innerHTML = "";  
}
```

```
function restartQuiz() {  
  currentQuestion = 0;  
  score = 0;
```

```
userAnswers.fill(null);  
document.getElementById("score").innerText = "";  
loadQuestion();  
}
```

```
loadQuestion();
```

```
</script>
```

```
</body>
```

```
</html>
```

