**Ex. No : 2 B**

**Date : 06.02.2025**

**CREATE A WEBPAGE TO HANDLE EVENTS AND OBJECT USING JAVASCRIPT**

**AIM:**

To create a webpage to implement JavaScript Objects.

**PROCEDURE:**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **JavaScript Object and Functions** | **DESCRIPTION** |
| **1** | **Objects** | **Used to group related data and functions together (e.g., Forum class managing questions).** |
| **2** | **Arrow Function** | **Concise syntax for defining functions, automatically binding this.** |
| **3** | **Getters** | **Accesses properties and methods of objects (e.g., question.text).** |
| **4** | **Setters** | **Enables controlled updates to object properties (e.g., updateQuestions** |
| **5** | **Find() Method** | **Locates an element in an array that satisfies a given condition (e.g., find a question by id).** |
| **6.** | **Map() Method** | **Iterates over an array and transforms elements (e.g., displaying answers).** |
| **7** | **Filter() Method** | **Creates a new array excluding elements that meet a condition (e.g., deleting a question).** |

**SOURCE CODE :**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Mini Stack Overflow</title>

<style>

body { font-family: Arial, sans-serif; max-width: 600px; margin: auto; }

h2 { color: #0073aa; }

input, textarea { width: 100%; padding: 8px; margin: 5px 0; }

button { background: #0073aa; color: white; padding: 8px; border: none; cursor: pointer; }

.question { border-bottom: 1px solid #ccc; padding: 10px 0; }

.answer { background: #f9f9f9; padding: 5px; margin-left: 20px; }

.delete { background: red; margin-left: 10px; }

</style>

</head>

<body>

<h2>Ask a Question</h2>

<input type="text" id="questionInput" placeholder="Type your question here...">

<button onclick="forum.addQuestion()">Post Question</button>

<h2>Questions</h2>

<div id="questionsContainer"></div>

<script>

class Forum {

constructor() {

this.questions = JSON.parse(localStorage.getItem('questions')) || [];

}

// Getter to return question count

get questionCount() {

return this.questions.length;

}

// Setter to set questions (useful for restoring from localStorage)

set updateQuestions(newQuestions) {

this.questions = newQuestions;

localStorage.setItem('questions', JSON.stringify(this.questions));

this.renderQuestions();

}

addQuestion = () => {

let questionText = document.getElementById("questionInput").value;

if (!questionText) return alert("Please enter a question!");

let newQuestion = {

id: Date.now(),

text: questionText,

answers: []

};

this.questions.push(newQuestion);

this.updateQuestions = this.questions;

document.getElementById("questionInput").value = "";

};

addAnswer = (id) => {

let answerText = prompt("Enter your answer:");

if (!answerText) return;

let question = this.questions.find(q => q.id === id);

question.answers.push(answerText);

this.updateQuestions = this.questions;

};

deleteQuestion = (id) => {

this.questions = this.questions.filter(q => q.id !== id);

this.updateQuestions = this.questions;

};

renderQuestions = () => {

let container = document.getElementById("questionsContainer");

container.innerHTML = "";

this.questions.forEach(q => {

let questionDiv = document.createElement("div");

questionDiv.classList.add("question");

questionDiv.innerHTML = `

<strong>${q.text}</strong>

<button onclick="forum.addAnswer(${q.id})">Answer</button>

<button class="delete" onclick="forum.deleteQuestion(${q.id})">Delete</button>

<div>${q.answers.map(a => `<p class="answer">${a}</p>`).join("")}</div>

`;

container.appendChild(questionDiv);

});

};

}

// Instantiate Forum object

let forum = new Forum();

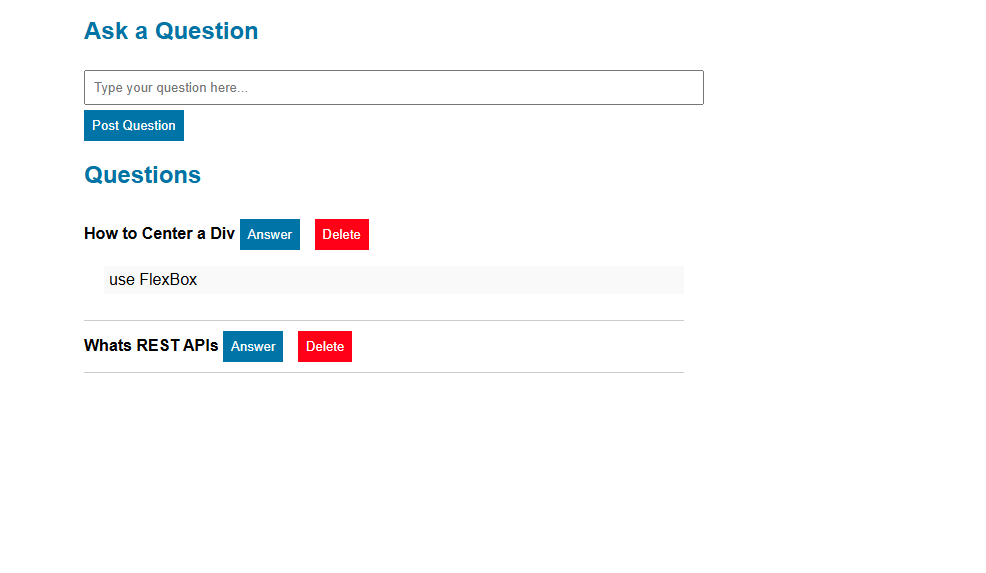
forum.renderQuestions();

</script>

</body>

</html>

**OUTPUT 1 :**



**RESULT :**

The program has been successfully executed and the output is verified.