Practical 1B

Write a JavaScript Program to get the user registration data and push to array/local storage with AJAX POST method and data list in new page.

1. The user submits the registration form (index.html).
2. The JavaScript (script.js) collects the data and stores it in localStorage.
3. The user is redirected to the dataDisplay.html page.
4. The data from localStorage is loaded and displayed in a table on dataDisplay.html.

---

1. HTML for Registration Form (`index.html`)

This part is responsible for creating the form where users can enter their details (name and email).

```
<form id="registrationForm">
    <label for="name">Name:</label><br>
    <input type="text" id="name" required><br><br>

    <label for="email">Email:</label><br>
    <input type="email" id="email" required><br><br>

    <button type="submit">Submit</button>
</form>
```

- **`<form>`**: A form where users will type in their name and email.
- **`<input>`**: These are the fields where the user types their name and email.
- **`<button>`**: A button that the user clicks to submit the form.

2. JavaScript for Registration Form Logic (`script.js`)

This part handles the form submission, saves data, and redirects to the next page. Let's break it down:

Getting the Data and Saving it:

```
let usersArray = JSON.parse(localStorage.getItem('users')) || [];
```

- **`localStorage.getItem('users')`**: It checks if there's any user data saved in the browser (in `localStorage`).
- **`JSON.parse()`**: This turns the data back into a usable JavaScript array (since `localStorage` stores data as text).
- **`|| []`**: If no data is found, we start with an empty array (`[]`).

Handling Form Submission:

```
document.getElementById('registrationForm').addEventListener('submit',
function(event) {
```

```
    event.preventDefault();  // Prevents the page from refreshing when
submitting

    const name = document.getElementById('name').value;  // Get the user's name
    const email = document.getElementById('email').value;  // Get the user's
email

    const user = { name, email };  // Create an object with the name and email

    usersArray.push(user);  // Add this user to the usersArray

    localStorage.setItem('users', JSON.stringify(usersArray));  // Save
usersArray to localStorage

    simulateAJAXPost(user);  // Simulate sending the data to a server
    window.location.href = 'dataDisplay.html';  // Redirect to a new page to
show the users
});
```

- **`event.preventDefault()`**: This stops the form from reloading the page after submission.
- **`document.getElementById()`**: This grabs the name and email values that the user typed.
- **`usersArray.push(user)`**: Adds the new user's details to the `usersArray`.
- **`localStorage.setItem()`**: Saves the updated array back to `localStorage`, so it doesn't disappear when you refresh the page.

Simulating AJAX (Sending Data):

```
function simulateAJAXPost(user) {
    setTimeout(() => {
        console.log('Simulating POST request:', user);
    }, 1000);  // Wait 1 second before showing the message
}
```

- **`setTimeout()`**: This simulates a delay, like sending data to a server. After 1 second, we print the user's data to the console (like we're sending it to a server).

3. Displaying Registered Users (`dataDisplay.html`)

On this page, we show the list of users who registered.

```
<table id="userTable">
    <thead>
        <tr>
            <th>Name</th>
            <th>Email</th>
        </tr>
    </thead>
    <tbody></tbody>
</table>
```

- **`<table>`**: We use this to display the list of registered users in rows.

4. JavaScript to Show Users (`dataDisplay.js`)

This JavaScript is responsible for showing the list of users when we load the page.

```
window.onload = function() {
    const usersArray = JSON.parse(localStorage.getItem('users')) || [];  // Get
the users data from localStorage
    const tableBody = document.querySelector('#userTable tbody');  // Get the
table body to add users

    if (usersArray.length === 0) {  // If no users are found
        tableBody.innerHTML = '<tr><td colspan="2">No users registered
yet.</td></tr>';
        return;  // Stop further execution
    }

    usersArray.forEach(user => {
        const row = document.createElement('tr');  // Create a new row for each
user
        row.innerHTML = `
            <td>${user.name}</td>  // Display the user's name
            <td>${user.email}</td>  // Display the user's email
        `;
        tableBody.appendChild(row);  // Add the row to the table
    });
};
```

- **`window.onload`**: This function runs when the page is loaded.
- **`localStorage.getItem('users')`**: Retrieves the stored users from `localStorage`.
- **`usersArray.forEach()`**: Loops through each user in the array and creates a new row in the table to display their details.
- **`<td>`**: Each user's name and email are added to separate columns in the table.

Simple Summary:

- **Registration Form** (`index.html`): Allows users to enter their name and email.
- **JavaScript Logic** (`script.js`): Collects the data, saves it in `localStorage` (so it stays even after the page is refreshed), and simulates sending it to a server using a fake AJAX call. It then redirects to the page that shows the list of users.
- **Displaying Data** (`dataDisplay.html` & `dataDisplay.js`): On the new page, we fetch the stored data from `localStorage` and display it in a table.

So, this code helps:

1. Collect user data.
2. Store it in `localStorage` so the data is not lost after refreshing the page.
3. Display the registered users on another page.