

```
1  /*
2
3  =====
4
5  Initial Users Array (Database)
6
7  =====
8
9  */
10
11 // We create an array of objects called 'users'. Each object represents one registered user with name, age and designation.
12
13 // Check if 'users' already exist in localStorage
14
15 const savedUsers = localStorage.getItem("users");
16
17 // If 'users' exists in localStorage, use it, otherwise create an empty array
18
19 // If found → parse and use them, otherwise use default
20
21 const users = savedUsers ? JSON.parse(savedUsers) : [
22
23     { name: "Ajay Suneja", age: 30, designation: "Front End Web Developer" },
24
25     { name: "Anshika Gupta", age: 25, designation: "Back End Web Developer" }
26 ];
27
28 /*
29
30 =====
31
32 Render Users to the DOM
33
34 =====
35
36 */
37
```



```
38 // This function takes the current 'users' array and displays it inside the <div id="users">.
39
40 const renderUsers = () => {
41
42     // Get the 'users' container div from HTML
43
44     const usersDiv = document.getElementById("users");
45
46     // Clear the div before re-rendering (so old content doesn't duplicate)
47
48     usersDiv.innerHTML = "";
49
50     // Loop over each user in the 'users' array
51
52     users.forEach((user) => {
53
54         // Append a user card (using template literals to insert values)
55
56         usersDiv.innerHTML += `<div class="user-card">
57
58             <strong>Name:</strong> ${user.name}<br>
59
60             <strong>Age:</strong> ${user.age}<br>
61
62             <strong>Designation:</strong> ${user.designation}
63
64         </div>`
65     })
66 }
67
68 /*
69
70 =====
71
72 Add User Function (Asynchronous)
73
74 =====
75
```



```
76 */
77
78 /*
79
80 This function simulates saving a new user (like to a database or server). It accepts:
81
82 1) 'newUser' → the new user object to add
83
84 2) 'renderUsers' → a callback function to run AFTER the user is added
85
86 */
87
88 const addUser = (newUser, renderUsers) => {
89
90     // Simulate server delay using setTimeout (2 seconds)
91
92     setTimeout(() => {
93
94         // Add the new user into the 'users' array
95
96         users.push(newUser);
97
98         // Save the updated 'users' array into localStorage
99
100        localStorage.setItem("users", JSON.stringify(users));
101
102        // Call the callback function (refresh the user list on the page)
103
104        renderUsers();
105
106    }, 2000); // 2000ms = 2 seconds
107 }
108
109 /*
110
111 =====
112
113 Initial Render of Users
```



```
114
115 =====
116
117 */
118
119 // Call renderUsers() once when the page loads, so that the initial users appear in the "Registered Users" section.
120
121 renderUsers();
122
123 /*
124
125 =====
126
127 Form Submission Handling
128
129 =====
130
131 */
132
133 // Get the form with id="userForm" and attach an event listener. When user submits the form, run the function below
134
135 document.getElementById("userForm").addEventListener("submit", (e) => {
136
137     // Prevent the form's default behavior (page reload)
138
139     e.preventDefault();
140
141     // Create a new user object with values from the form inputs
142
143     const newUser = {
144
145         name: document.getElementById("name").value,           // name input
146
147         age: document.getElementById("age").value,             // age input
148
149         designation: document.getElementById("designation").value // designation input
150     };
151
```



```
152 // Change button text to show "Adding..."
153
154 e.target.querySelector("button").textContent = "Adding...";
155
156 // Disable the button temporarily (prevent multiple submissions while waiting)
157
158 e.target.querySelector("button").disabled = true;
159
160 /*
161
162 Call addUser() function. Pass:
163
164 1) newUser (the object we just created)
165
166 2) a callback function that runs AFTER user is added
167
168 */
169
170 addUser(newUser, () => {
171
172     // Re-render all users (to show the new one on screen)
173
174     renderUsers();
175
176     // Reset form inputs (clear fields)
177
178     e.target.reset();
179
180     // Change button text back to "Add User"
181
182     e.target.querySelector("button").textContent = "Add User";
183
184     // Enable the button again (so user can add more entries)
185
186     e.target.querySelector("button").disabled = false;
187 });
188 });
```

