**Worksheet 3**

**Student Name: Sonu Munda UID: 23MCA20616**

**Branch: MCA Section: 4B**

**Semester: 3 Date : 22-08-2024**

**Subject Name : Backend Technologies Lab Subject Code: 23CAH-705**

**Aim/Overview of The Practical:**

Create a Signup page and store the input data by user in file.

**CODE :**

Server Code

const express = require("express");

const app = express();

const fs = require("fs");

const path = require("path");

const cors = require("cors");

app.use(express.json());

app.use(cors());

app.get("/", (req, res) => {

  res.send("Hello World");

});

app.post("/api/signup", async (req, res) => {

  const { name, username, email, password } = req.body;

  const filePath = path.join(\_\_dirname, "db", "users.json");

  try {

    let userData = [];

    if (fs.existsSync(filePath)) {

      const fileContent = fs.readFileSync(filePath, "utf-8");

      userData = JSON.parse(fileContent);

    }

    const newUser = { name, username, email, password };

    userData.push(newUser);

    fs.writeFileSync(filePath, JSON.stringify(userData, null, 2), "utf-8");

    res.status(201).json({ message: "User signed up successfully!" });

  } catch (error) {

    console.error("Error during signup:", error);

    res.status(500).json({ error: "Internal Server Error" });

  }

});

app.get("/api/users", (req, res) => {

  const filePath = path.join(\_\_dirname, "db", "users.json");

  try {

    if (fs.existsSync(filePath)) {

      const fileContent = fs.readFileSync(filePath, "utf-8");

      const userData = JSON.parse(fileContent);

      res.json(userData);

    } else {

      res.json([]);

    }

  } catch (error) {

    console.error("Error fetching users:", error);

    res.status(500).json({ error: "Internal Server Error" });

  }

});

const PORT = 3000;

app.listen(PORT, () => {

  console.log("Server is runnig at port: ", PORT);

});

Index.html (Client Code) :

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

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

    <title>Signup Form</title>

  </head>

  <body>

    <div class="container">

      <form id="signupForm">

        <h1>Signup Form</h1>

        <div class="form-group">

          <label for="name">Name:</label>

          <input type="text" id="name" name="name" required />

        </div>

        <div class="form-group">

          <label for="username">Username:</label>

          <input type="text" id="username" name="username" required />

        </div>

        <div class="form-group">

          <label for="email">Email:</label>

          <input type="email" id="email" name="email" required />

        </div>

        <div class="form-group">

          <label for="password">Password:</label>

          <input type="password" id="password" name="password" required />

        </div>

        <button type="submit">Sign Up</button>

      </form>

      <div class="registerd-users">

        <h2>Registered Users</h2>

        <ul id="userList" class="user-list"></ul>

      </div>

    </div>

    <script>

      document.addEventListener("DOMContentLoaded", () => {

        const form = document.getElementById("signupForm");

        const userList = document.getElementById("userList");

        form.addEventListener("submit", async (event) => {

          event.preventDefault();

          const formData = new FormData(form);

          const data = {

            name: formData.get("name"),

            username: formData.get("username"),

            email: formData.get("email"),

            password: formData.get("password"),

          };

          try {

            const response = await fetch("http://localhost:3000/api/signup", {

              method: "POST",

              headers: {

                "Content-Type": "application/json",

              },

              body: JSON.stringify(data),

            });

            if (response.ok) {

              alert("User signed up successfully!");

              form.reset();

              fetchUsers();

            } else {

              alert("Failed to sign up user.");

            }

          } catch (error) {

            console.error("Error during signup:", error);

            alert("An error occurred.");

          }

        });

        async function fetchUsers() {

          try {

            const response = await fetch("http://localhost:3000/api/users");

            const users = await response.json();

            userList.innerHTML = users

              .map(

                (user) => `

                        <li>

                          <strong>Name:</strong> ${user.name}<br>

                          <strong>Username:</strong> ${user.username}<br>

                          <strong>Email:</strong> ${user.email}

                        </li>

                        `

              )

              .join("");

          } catch (error) {

            console.error("Error fetching users:", error);

          }

        }

        fetchUsers();

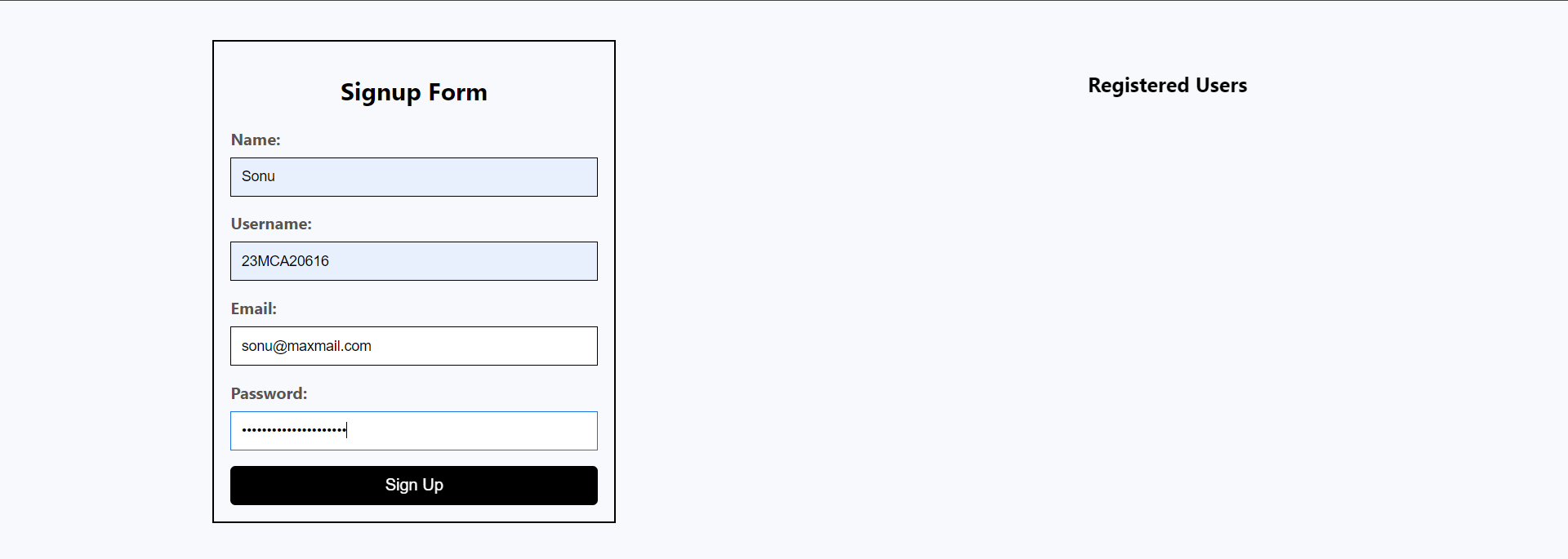
      });

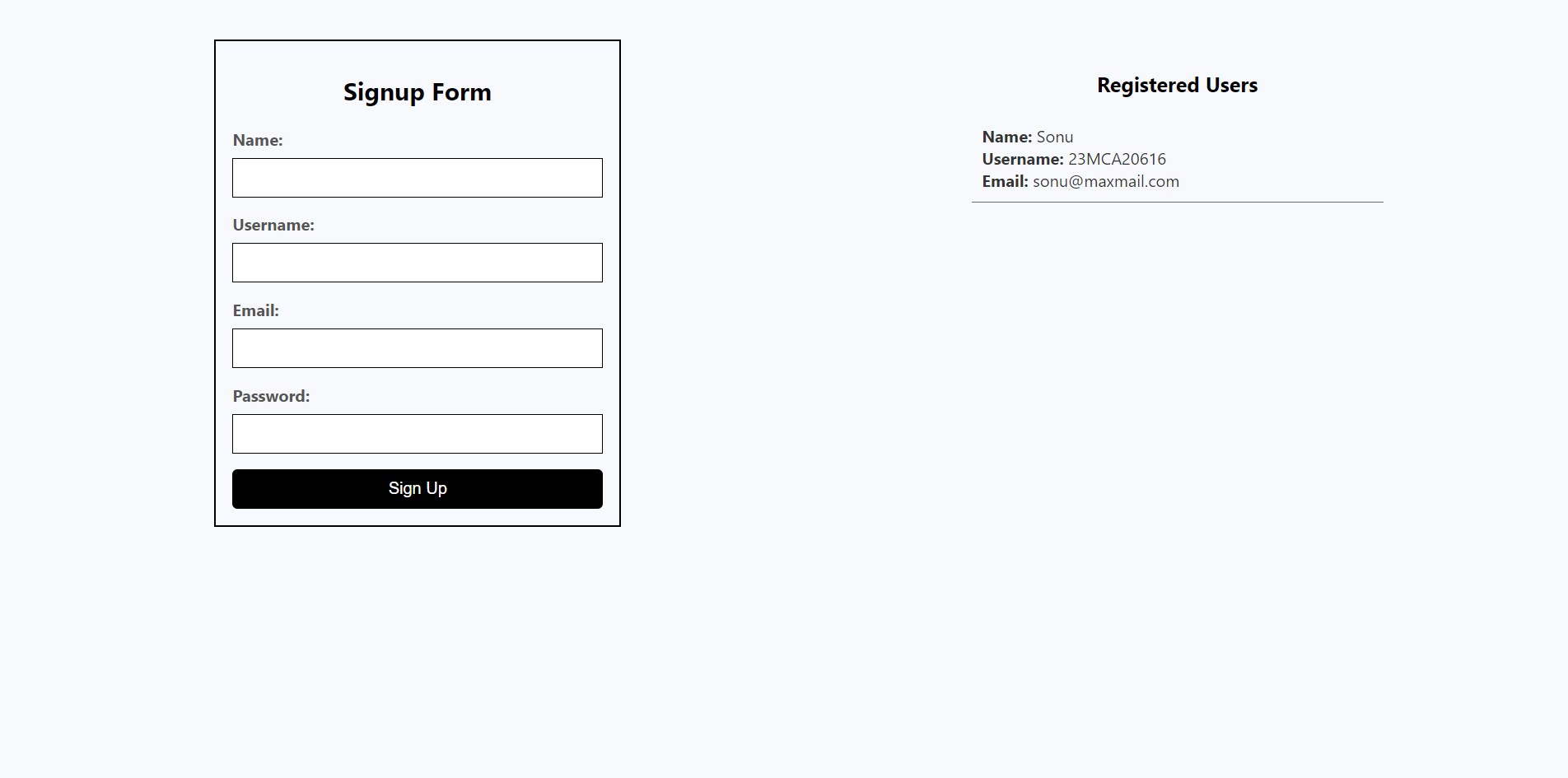
    </script>

  </body>

</html>

**OUTPUT**

****

****

**Learning & Outcomes:**

1. Learned about fs module.
2. Learned to store data in JSON from front end.
3. Learned about path module