5)Create a Login *Screen?*

Creating a login screen involves both frontend design and backend logic. Here's a basic example using HTML, CSS, and JavaScript for the frontend, and a simple backend logic explanation using Node.js with Express for illustration.

Frontend (HTML and CSS)

HTML (login.html)

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

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

  <title>Login</title>

  <link rel="stylesheet" href="styles.css">

</head>

<body>

  <div class="login-container">

    <h2>Login</h2>

    <form id="loginForm">

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

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

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

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

      <button type="submit">Login</button>

    </form>

    <div id="message"></div>

  </div>

  <script src="script.js"></script>

</body>
```

```
</html>
```

CSS (styles.css)

```
body {
    font-family: Arial, sans-serif;
    background-color: #f0f0f0;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    margin: 0;
}

.login-container {
    background-color: #fff;
    padding: 20px;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    text-align: center;
}

.login-container h2 {
    margin-bottom: 20px;
}

.login-container label {
    display: block;
    margin: 10px 0 5px;
}
```

```
.login-container input {
   width: 100%;
   padding: 10px;
   margin-bottom: 10px;
   border: 1px solid #ccc;
   border-radius: 4px;
}

.login-container button {
   padding: 10px 20px;
   background-color: #007bff;
   color: #fff;
   border: none;
   border-radius: 4px;
   cursor: pointer;
}

.login-container button:hover {
   background-color: #0056b3;
}

#message {
   margin-top: 10px;
   color: red;
}
```

JavaScript (script.js)

```
document.getElementById('loginForm').addEventListener('submit', function(e) {
   e.preventDefault();
```

```
  const username = document.getElementById('username').value;
  const password = document.getElementById('password').value;

  fetch('/login', {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json'
    },
    body: JSON.stringify({ username, password })
  })
  .then(response => response.json())
  .then(data => {
    if (data.success) {
      window.location.href = '/dashboard';
    } else {
      document.getElementById('message').innerText = data.message;
    }
  })
  .catch(error => {
    console.error('Error:', error);
    document.getElementById('message').innerText = 'An error occurred. Please try again.';
  });
});
```

Backend (Node.js and Express)

Node.js and Express Setup

1. *Install Node.js and npm*: Ensure Node.js and npm are installed on your system.

2. *Create a new project*: Initialize a new Node.js project using `npm init`.
3. *Install Express*: Install Express and other necessary dependencies using `npm install express body-parser`.

Server Code (server.js)

```
const express = require('express');
const bodyParser = require('body-parser');

const app = express();
const port = 3000;

// Middleware
app.use(bodyParser.json());
app.use(express.static('public'));

// Dummy user data
const users = {
  user1: 'password123',
  user2: 'password456'
};

// Login endpoint
app.post('/login', (req, res) => {
  const { username, password } = req.body;

  if (users[username] && users[username] === password) {
    res.json({ success: true });
  } else {
    res.json({ success: false, message: 'Invalid username or password' });
  }
});

// Start the server
app.listen(port, () => {
  console.log(`Server running at http://localhost:${port}/`);
});
```

Folder Structure

```
project-folder/
│
├── public/
│   ├── login.html
│   ├── styles.css
│   └── script.js
│
├── server.js └── package.json
```

Steps to Run

4. Save the HTML, CSS, and JavaScript files in the `public` directory.
5. Run `npm install express body-parser` in the project folder.
6. Run the server using `node server.js`.
7. Open a web browser and navigate to `http://localhost:3000/login.html`.