Private Room chat

Step 1: Set Up Your Project in VS Code

Open VS Code and create a new folder for your project.

Inside the folder, create the following files:

index.html (Main HTML file)

style.css (For styling)

script.js (For JavaScript logic)

Step 2: Write the HTML Structure

Create and open the index.html file, then add the following code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Private Room Chat</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div class="chat-container">
        <h2>Private Chat Room</h2>
        <div id="chat-box"></div>
        <input type="text" id="message-input" placeholder="Type a message...">
        <button onclick="sendMessage()">Send</button>
    </div>

    <script src="script.js"></script>
</body>
</html>
```

Step 3: Style the Chat Room

Open style.css and add styling for a simple chat UI:


```
body {

    font-family: Arial, sans-serif;

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    background-color: #f4f4f4;

    margin: 0;

}


.chat-container {

    width: 350px;

    background: white;

    padding: 20px;

    border-radius: 10px;

    box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);

    text-align: center;

}


#chat-box {

    height: 300px;

    overflow-y: auto;

    border: 1px solid #ddd;

    padding: 10px;

    margin-bottom: 10px;

    background: #fafafa;

}
```

```css
#message-input {

    width: 70%;

    padding: 8px;

    margin-right: 5px;

    border: 1px solid #ccc;

    border-radius: 5px;

}


button {

    padding: 8px 15px;

    background: blue;

    color: white;

    border: none;

    border-radius: 5px;

    cursor: pointer;

}


button:hover {

    background: darkblue;

}
```

Step 4: Add JavaScript Logic for Messaging

Now, open script.js and add the logic to send messages:

```javascript
function sendMessage() {

    let inputField = document.getElementById("message-input");

    let message = inputField.value.trim();


    if (message !== "") {

        let chatBox = document.getElementById("chat-box");

        let messageElement = document.createElement("p");

        messageElement.textContent = message;
```

```
        messageElement.style.background = "#d1e7fd";

        messageElement.style.padding = "8px";

        messageElement.style.borderRadius = "5px";

        messageElement.style.marginBottom = "5px";

        messageElement.style.textAlign = "right";


        chatBox.appendChild(messageElement);

        inputField.value = "";

        chatBox.scrollTop = chatBox.scrollHeight;

    }

}
```

Step 5: Test the Chat Interface

Open the index.html file in a browser (double-click it or use Live Server in VS Code).

Type a message and click "Send."

The message should appear in the chat box.

Step 6: Add Real-Time Communication (Optional)

To make this a real-time chat, you need WebSockets using Socket.io and a Node.js backend.

Would you like to continue with that?