

```
<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width,initial-scale=1" />

<title>Real-time Chat — Examples</title>

<style>

:root{font-family:system-ui,-apple-system,Segoe UI,Roboto,'Helvetica Neue',Arial}

body{max-width:900px;margin:24px auto;padding:16px}

header{display:flex;align-items:center;gap:12px}

.card{border:1px solid #ddd;padding:12px;border-radius:8px;margin-top:12px}

.chat{height:320px;overflow:auto;border:1px solid #eee;padding:12px;border-
radius:6px;background:#fafafa}

.message{margin-bottom:8px}

.meta{font-size:12px;color:#666}

.controls{display:flex;gap:8px;margin-top:8px}

input[type=text]{flex:1;padding:8px;border-radius:6px;border:1px solid #ccc}

button{padding:8px 12px;border-radius:6px;border:1px solid
#2b7;background:#2b7;color:white}

small.note{color:#444}

pre.server{background:#111;color:#e6e6e6;padding:12px;border-
radius:6px;overflow:auto}

</style>

</head>

<body>

<header>

<h1>Real-time Chat — Examples</h1>
```

</header>

<section class="card">

<h2>A) WebSocket (Node.js server + HTML client)</h2>

<p class="note">Run the Node.js server (code included below) on port 3000, then open this page in the browser. The client connects to ws://localhost:3000.</p>

<div class="card">

<h3>Client (this page): WebSocket chat</h3>

<div id="ws-ui">

<div style="margin-bottom:8px">

<label>Display name: <input id="ws-name" type="text" placeholder="Your name" value="Guest"/></label>

</div>

<div id="ws-chat" class="chat" aria-live="polite"></div>

<div class="controls">

<input id="ws-input" type="text" placeholder="Type message and press Send" />

<button id="ws-send">Send</button>

</div>

<small class="note">WebSocket status: disconnected</small>

</div>

</div>

<!DOCTYPE html>

```
<html lang="en">

<head>

  <meta charset="UTF-8">

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

  <title>Real-Time Chat App</title>

  <style>

    body {

      font-family: Arial, sans-serif;

      background: #f3f4f6;

      display: flex;

      justify-content: center;

      align-items: center;

      height: 100vh;

    }


    .chat-container {

      width: 350px;

      background: white;

      border-radius: 10px;

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

      padding: 20px;

    }


    .chat-box {

      border: 1px solid #ddd;

      border-radius: 5px;
```

```
height: 300px;  
overflow-y: auto;  
padding: 10px;  
background: #fafafa;  
margin-bottom: 10px;  
}
```

```
.message {  
background: #e1f5fe;  
margin: 5px 0;  
padding: 8px;  
border-radius: 6px;  
}
```

```
.user-info, .input-area {  
display: flex;  
gap: 5px;  
margin-bottom: 10px;  
}
```

```
input[type="text"] {  
flex: 1;  
padding: 8px;  
border: 1px solid #ccc;  
border-radius: 5px;  
}
```

```
button {  
  background: #007bff;  
  color: white;  
  border: none;  
  padding: 8px 12px;  
  border-radius: 5px;  
  cursor: pointer;  
}
```

```
button:hover {  
  background: #0056b3;  
}
```

```
#status span {  
  color: red;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="chat-container">
```

```
<h2>💬 Real-Time Chat</h2>
```

```
<div class="user-info">
```

```
<label>Your Name:</label>
```

```
<input type="text" id="name" placeholder="Enter your name" value="Guest">
```

</div>

<div id="chat-box" class="chat-box"></div>

<div class="input-area">

<input type="text" id="message" placeholder="Type your message...">

<button id="sendBtn">Send</button>

</div>

<p id="status">Status: Disconnected</p>

</div>

<script>

const wsUrl = "ws://localhost:3000"; // WebSocket server address

let socket;

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

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

const messageInput = document.getElementById("message");

const nameInput = document.getElementById("name");

const statusText = document.querySelector("#status span");

function connectWebSocket() {

socket = new WebSocket(wsUrl);

socket.onopen = () => {

```
statusText.textContent = "Connected";

statusText.style.color = "green";

};


socket.onclose = () => {

    statusText.textContent = "Disconnected";

    statusText.style.color = "red";

};


socket.onmessage = (event) => {

    const msg = JSON.parse(event.data);

    displayMessage(` ${msg.name}: ${msg.text} `);

};

}


function displayMessage(message) {

    const div = document.createElement("div");

    div.classList.add("message");

    div.textContent = message;

    chatBox.appendChild(div);

    chatBox.scrollTop = chatBox.scrollHeight;

}


function sendMessage() {

    const name = nameInput.value.trim() || "Guest";

    const text = messageInput.value.trim();
```

```
if (text && socket.readyState === WebSocket.OPEN) {  
    const data = { name, text };  
    socket.send(JSON.stringify(data));  
    messageInput.value = "";  
}  
}
```

sen