Skrypt 25

Przygotuj prosty program KLIENT-SERWER typu chat używając WebSocket. Klient łączy się z serwerem, a serwer odpisuje do wszystkich podłączonych klientów w danym momencie.

KLIENT

Podpowiedź:

        // Połączenie z serwerem WebSocket

        const socket = new WebSocket('ws://localhost:3000');

// Obsługa odebranych wiadomości

socket.onmessage = (event) => {

    const blob = event.data; // Odebrane dane to Blob

    // Konwersja Blob do tekstu

    const reader = new FileReader();

    reader.onload = () => {

        const message = document.createElement('p');

        message.textContent = reader.result; // Konwersja do tekstu

        chat.appendChild(message);

    };

    reader.readAsText(blob); // Czytaj dane jako tekst

};

// Wysyłanie wiadomości po kliknięciu przycisku

sendButton.addEventListener('click', () => {

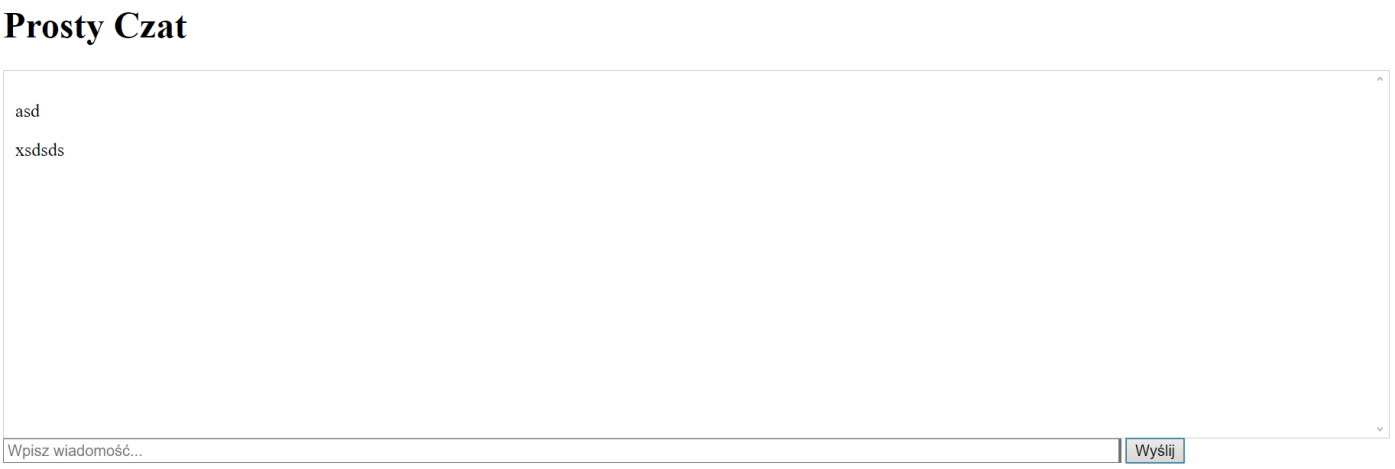
    const message = messageInput.value;

    if (message) {

        socket.send(message); // Wysyłanie danych jako tekst

        messageInput.value = '';

    }



SERWER

/ Tworzenie serwera WebSocket

const server = new WebSocket.Server({ port: 3000 });

console.log('Serwer WebSocket działa na ws://localhost:3000');

// Obsługa połączeń WebSocket

server.on('connection', (socket) => {

    console.log('Nowy klient połączony');

    // Obsługa odebranych wiadomości od klienta

    socket.on('message', (message) => {

        console.log(`Odebrano: ${message}`);

        // Wysyłanie wiadomości do wszystkich klientów

        server.clients.forEach((client) => {

            if (client.readyState === WebSocket.OPEN) {

                client.send(message);

            }

        });

    });