

Check user online Socket.io

How to check if a user is online or not in socket.io using map?

To track online users in **Socket.io** using a `Map`, you can follow this approach:

Steps to Implement

1. Maintain a `Map` of connected users

- The key will be the `userId`, and the value will be the `socket.id`.

2. Track when a user connects

- Store their `socket.id` in the `Map`.

3. Track when a user disconnects

- Remove their entry from the `Map`.

4. Check if a user is online

- If the `userId` exists in the `Map`, the user is online.
-

Code Implementation

Server-side (Node.js with Socket.io)

javascript

```
const io = require("socket.io")(3000, {
  cors: {
    origin: "*",
  },
});

const onlineUsers = new Map();
```

```

io.on("connection", (socket) => {
  console.log("A user connected:", socket.id);

  // Listen for "user-connected" event from the client
  socket.on("user-connected", (userId) => {
    onlineUsers.set(userId, socket.id);
    console.log(`User ${userId} is online`);
  });

  // Listen for user disconnecting
  socket.on("disconnect", () => {
    const userId = [...onlineUsers.entries()].find(([key, value]) => value ===
socket.id)?.[0];
    if (userId) {
      onlineUsers.delete(userId);
      console.log(`User ${userId} is offline`);
    }
  });

  // Check if a user is online
  socket.on("check-user-online", (userId, callback) => {
    const isOnline = onlineUsers.has(userId);
    callback(isOnline);
  });
});

```

Client-side (React/Vanilla JavaScript)

```

javascript

import { io } from "socket.io-client";

const socket = io("http://localhost:3000");

// Emit user-connected event when the user logs in
const userId = "user123"; // Replace with actual user ID
socket.emit("user-connected", userId);

// Check if a specific user is online

```

```
const checkUserOnline = (userId) => {
  socket.emit("check-user-online", userId, (isOnline) => {
    console.log(`User ${userId} is ${isOnline ? "online" : "offline"}`);
  });
};

// Call the function to check if a user is online
checkUserOnline("user123");
```

Explanation

1. When a user connects:

- The server listens for "user-connected" and stores `userId -> socket.id` in the `Map`.

2. When a user disconnects:

- The server removes their entry from the `Map` using `socket.id`.

3. Checking if a user is online:

- The client emits "check-user-online", and the server responds with a boolean.

Advantages of Using `Map`

- **Efficient Lookups:** `Map.has(userId)` is $O(1)$, faster than arrays.
- **Avoids Duplicate Entries:** Each `userId` is unique.
- **Flexible for Additional Features:** Can store more metadata if needed.

Would you like me to extend this with **database persistence** or **multi-device support**? 🚀