

Broadcasting

The next goal is for us to emit the event from the server to the rest of the users.

In order to send an event to everyone, Socket.IO gives us the `io.emit()` method.

```
io.emit('some event', { someProperty: 'some value', otherProperty: 'other value' }); // This
```

If you want to send a message to everyone except for a certain emitting socket, we have the `broadcast` flag for emitting from that socket:

```
io.on('connection', (socket) => {
  socket.broadcast.emit('hi');
});
```

In this case, for the sake of simplicity we'll send the message to everyone, including the sender.

```
io.on('connection', (socket) => {
  socket.on('chat message', (msg) => {
    io.emit('chat message', msg);
  });
});
```

And on the client side when we capture a `chat message` event we'll include it in the page. The *total* client-side JavaScript code now amounts to:

```
<script src="/socket.io/socket.io.js"></script>
<script>
  var socket = io();

  var messages = document.getElementById('messages');
  var form = document.getElementById('form');
  var input = document.getElementById('input');

  form.addEventListener('submit', function(e) {
    e.preventDefault();
    if (input.value) {
      socket.emit('chat message', input.value);
      input.value = '';
    }
  });

  socket.on('chat message', function(msg) {
    var item = document.createElement('li');
    item.textContent = msg;
    messages.appendChild(item);
    window.scrollTo(0, document.body.scrollHeight);
  });
</script>
```

And that completes our chat application, in about 20 lines of code! This is what it looks like:

Homework

Here are some ideas to improve the application:

Getting this example

You can find it on [GitHub here](#).

```
git clone https://github.com/socketio/chat-example.git
```

Copyright © 2024 Socket.IO