***CLIENT***

<script>

var socket = io();

socket.on('message', function (data) {

console.log(data);

});

var movement = {

up: false,

down: false,

left: false,

right: false

}

document.addEventListener('keydown', function (event) {

switch (event.keyCode) {

case 65: // A

movement.left = true;

break;

//…

}

});

document.addEventListener('keyup', function (event) {

switch (event.keyCode) {

case 65: // A

movement.left = false;

break;

//…

}

});

socket.emit('new player');

setInterval(function () {

socket.emit('movement', movement);

}, 1000 / 60);

var canvas = document.getElementById('canvas');

canvas.width = 800;

canvas.height = 600;

var context = canvas.getContext('2d');

socket.on('state', function (players) {

context.clearRect(0, 0, 800, 600);

context.fillStyle = 'green';

for (var id in players) {

var player = players[id];

context.beginPath();

context.arc(player.x, player.y, 10, 0, 2 \* Math.PI);

context.fill();

}

});

</script>

***SERVER***

'use strict';

//https://hackernoon.com/how-to-build-a-multiplayer-browser-game-4a793818c29b

// Dependencies

var express = require('express');

var http = require('http');

var path = require('path');

var socketIO = require('socket.io');

var app = express();

var server = http.Server(app);

var io = socketIO(server);

app.set('port', 5000);

app.use('/static', express.static(\_\_dirname + '/static'));

// Routing

app.get('/', function (request, response) {

response.sendFile(path.join(\_\_dirname, 'index.html'));

});

// Starts the server.

server.listen(5000, function () {

console.log('Starting server on port 5000');

});

// Add the WebSocket handlers

io.on('connection', function (socket) {

});

setInterval(function () {

io.sockets.emit('message', 'hi!');

}, 1000);

var players = {};

io.on('connection', function (socket) {

socket.on('new player', function () {

players[socket.id] = {

x: 300,

y: 300

};

});

socket.on('movement', function (data) {

var player = players[socket.id] || {};

if (data.left) {

player.x -= 5;

}

//…

});

});

setInterval(function () {

io.sockets.emit('state', players);

}, 1000 / 60);