

Socket Assignment

Server

```
var net = require('net')
var HOST = '127.0.0.1' //กำหนด host เป็น 127.0.0.1
var PORT = 5000 //ใช้ port 5000
var server = new net.createServer() //สร้าง server
var pIDs = []
var hp = {}
var msg = {}

let optionList = 'Select Options:\n'           //ตัวเลือกที่จะให้userเลือก
                + '1 Shoot Person\n'
                + '2 Give Up'

var createID = () => {           //สร้างid และhpให้กับuser
  let id = 0
  while (true) {
    if (!pIDs.includes(id)) {
      hp[id] = 100
      msg[id] = ''
      pIDs.push(id)
      return id
    }
    id++
  }
}

var removeID = (id) => {        //ลบข้อมูล id ของ user
  delete hp[id]
  delete msg[id]
  const index = array.indexOf(id)
  if (index > -1) {
    pIDs.splice(index, 1)
  }
}

var playerDestroy = (pID, sock) => { //ทำลายsocket และid ของ user
  removeID(pID)
  sock.destroy()
}

var msgPrint = async (id, sock) => { //แสดงข้อความกรณีที่ user ตาย
  if (!pIDs.indexOf(id)) return
  setTimeout(function() {
    if (msg[id] !== '') {
      if (msg[id] == 'die') {
        sock.write('You died!')
        playerDestroy(id, sock)
      } else {
```

```

        sock.write(msg[id])
        msg[id] = ''
    }
}
msgPrint(id, sock)
}, 1000)
}

server.listen(PORT, HOST)
server.on('connection', function (sock) { //server ทำการเชื่อมต่อกับ client
    let state = 0
    let pID = createID()

    console.log('CONNECTED: ' + sock.remoteAddress + ':' + sock.remotePort)

    msgPrint(pID, sock)
    sock.write('Welcome to shooting game. Your ID: ' + pID + '\n' + optionList) //แสดง id และ
option list ของ user
    state = 1

    sock.on('data', function (data) { //server เชื่อมต่อกับ client ถ้าสำเร็จ
        switch (state) {
            case 1: // รอการตอบรับจาก user
                if (data == 1) { // shoot person
                    let res = ''
                    pIDs.forEach(id => {
                        res += id + ' '
                    });
                    sock.write('Choose targeted IDs:\n' + res)
                    state = 2
                } else {
                    msg[pID] = 'die'
                    state = 1
                }
                break
            case 2: // รอรับ target (targetedID) จาก user
                let res = ''
                hp[data] -= 20
                sock.write(optionList)
                state = 1
                break
            default: // กรณีที่ตัวเลือกที่ไม่อยู่ใน case ทั้ง 2 case ให้แสดง game crashed
                sock.write('Game crashed with state ' + state + ' and data ' + data)
                playerDestroy(pID, sock)
                break
        }
    });
});

```

```

sock.on('close', function (data) { // ลบ id ของ user และแสดงข้อมูลปิด
    removeID(pID)
    console.log('CLOSED: ' + sock.remoteAddress + ' ' + sock.remotePort)
});
});

```

Client

```

const { Socket } = require('dgram');
var net = require('net')
var rl = require('readline')

var HOST = '127.0.0.1' //กำหนด host เป็น 127.0.0.1
var PORT = 5000 //ใช้ port 5000

var client = new net.Socket() //สร้าง socket ของ client
var readline = rl.createInterface(process.stdin, process.stdout) //กำหนดให้ตัวแปร readline เป็นตัวแปรรับค่าจาก user
var state = 0

readline._normalWrite = function(b) { //กำหนดการทำงานของ readline
    if(b == undefined) {
        return;
    }
    if(!this._line_buffer) {
        this._line_buffer = '';
    }
    this._line_buffer += b.toString();
    if(this._line_buffer.indexOf('\n') !== -1) {
        var lines = this._line_buffer.split('\n');
        // either '' or the unfinished portion of the next line
        this._line_buffer = lines.pop();
        lines.forEach(function(line) {
            this._onLine(line + '\n');
        }, this);
    }
};

client.connect(PORT, HOST, function(){ //client เชื่อมต่อกับ server
    console.log('Connected');
})

client.on('data', function(data){ //client เชื่อมต่อกับ server สำเร็จ และทำงานขึ้นตอนถัดไป
    console.log('"' + data) //แสดง id ที่ user ได้จาก server
    switch(state){

        case 0: //หลังจาก user ได้ id แล้วทำ state ถัดไป
            state = 1;
            break
    }
}

```

```

case 1: //เลือกที่จะเล่นหรือไม่
    readline.on('line', function (line) {
        console.log(`${state} Got ${line}!`)
        client.write("" + line)
        if(line == 1) state = 2
    })
    break

case 2: //เลือกที่จะบิงใคร
    readline.on('line', function (line) {
        console.log(`${state} Got ${line}!`)
        client.write("" + line)
        state = 1
    })
    break

default: //กรณีที่เกิดข้อผิดพลาดขึ้น
    console.log('Game error: state'+ state + 'data' + data) //แสดงstate และ data
    client.destroy() //สั่งทำลาย socketของclient
    break
}
})

client.on('close',function(){ //แสดงการปิดของ client
    console.log('GoodBye')
})

```

ผลการrun

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Dell> cd ..
PS C:\Users> cd ..
PS C:\> cd D:\work\y3\2term\Clientserver
PS D:\work\y3\2term\Clientserver> node server.js

```

การrun Server.js ยังไม่มี client

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Dell> cd ..
PS C:\Users> cd D:\work\y3\2term\Clientserver
PS D:\work\y3\2term\Clientserver> node client.js
Connected
Welcome to shooting game. Your ID: 0
Select Options:
1 Shoot Person
2 Give Up
1
1
1
1
1
1
1
1

```

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS C:\Users\Dell> cd ..
PS C:\Users> cd ..
PS C:\> cd D:\work\y3\2term\Clientserver
PS D:\work\y3\2term\Clientserver> node client.js
Connected
Welcome to shooting game. Your ID: 1
Select Options:
1 Shoot Person
2 Give Up
1
1
1
1
1
0

```

การ run client.js โดยมี 2 client คือ ID 0,1

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Dell> cd ..
PS C:\Users> cd ..
PS C:\> cd D:\work\y3\2term\Clientserver
PS D:\work\y3\2term\Clientserver> node server.js
CONNECTED: 127.0.0.1:57780
CONNECTED: 127.0.0.1:57785

```

Server.js หลังจากที่มีการ run client.js ทั้ง 2 client