Project: HTTP JSON API Node.js Time Server

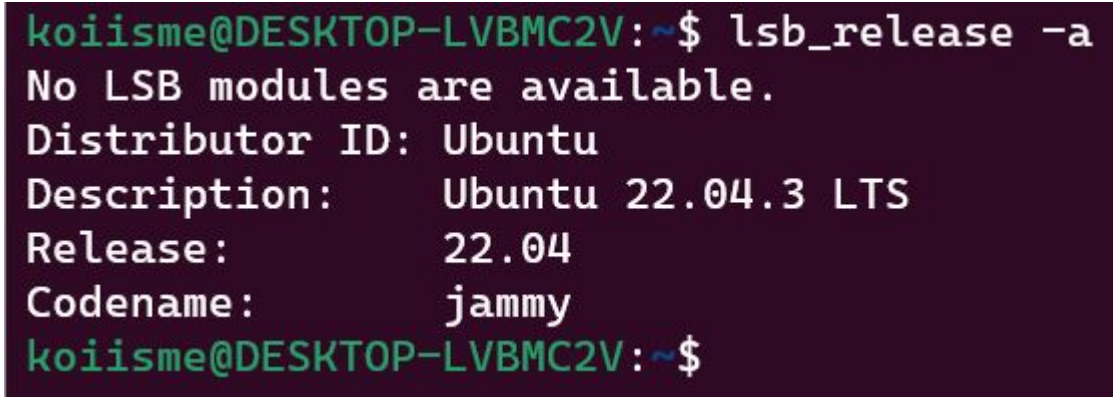
Khoi Duong
Prof. Chang
CS571
1/30/2024

Step 1: Install Node.js on Ubuntu

1. Setup Ubuntu

Refer to Setup Ubuntu - Khoi Duong - 19610

2. How to Check your Ubuntu Version

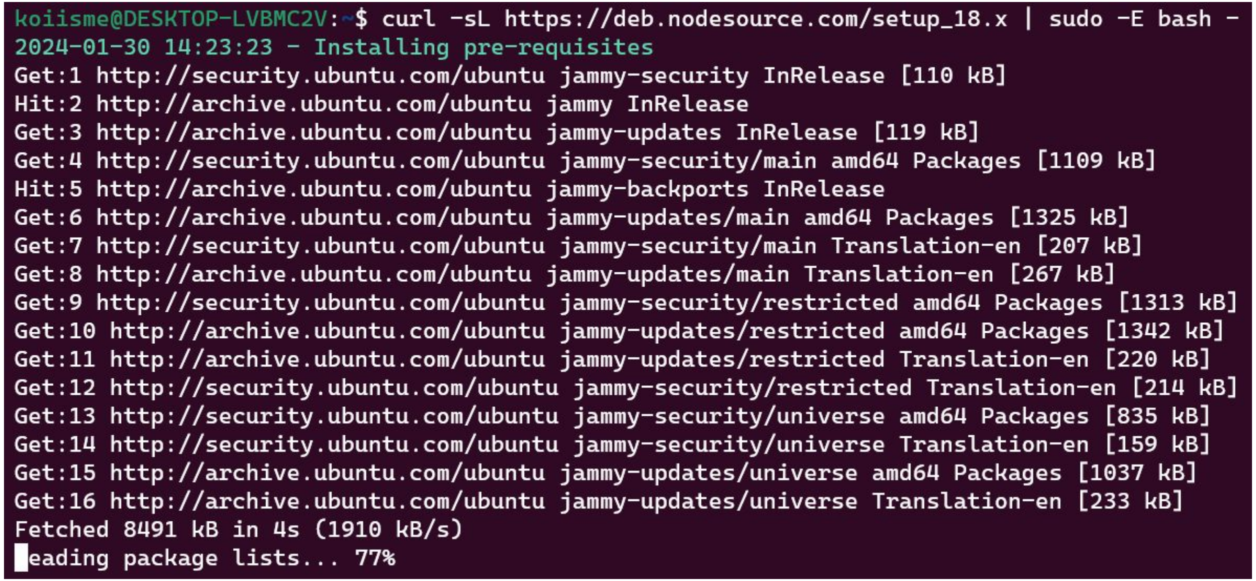


```
koiisme@DESKTOP-LVBMC2V:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 22.04.3 LTS
Release:        22.04
Codename:       jammy
koiisme@DESKTOP-LVBMC2V:~$
```

Step 1 (cont)

3. Enable the NodeSource repository by running the following curl

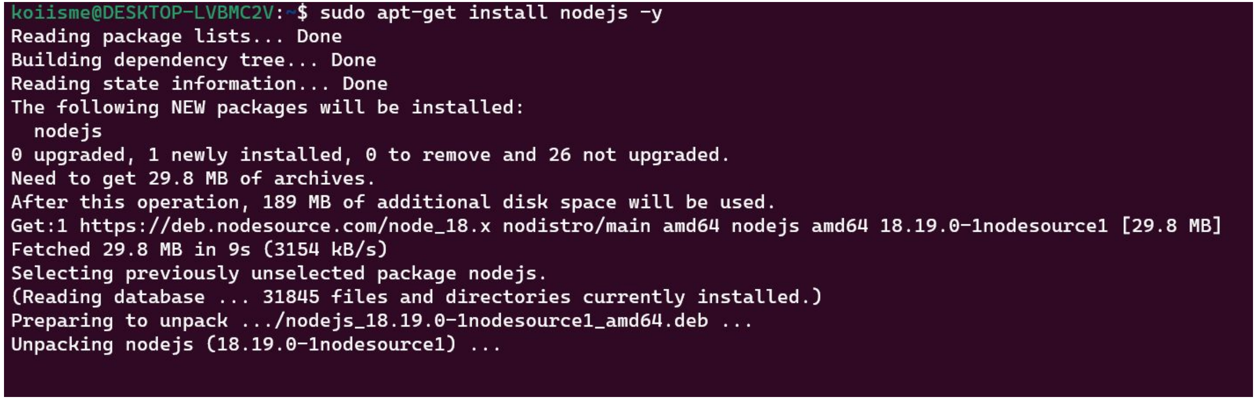
curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash -



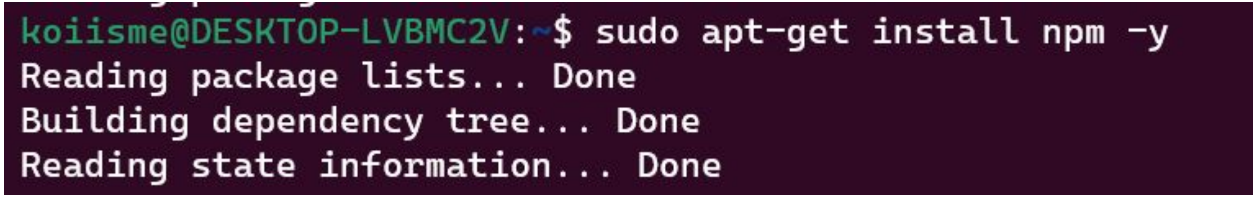
```
koiisme@DESKTOP-LVBMC2V:~$ curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash -
2024-01-30 14:23:23 - Installing pre-requisites
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:2 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1109 kB]
Hit:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1325 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [207 kB]
Get:8 http://archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [267 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1313 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1342 kB]
Get:11 http://archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [220 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [214 kB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [835 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [159 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1037 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [233 kB]
Fetched 8491 kB in 4s (1910 kB/s)
█eading package lists... 77%
```

Step 1 (cont)

4. Install Node.js and npm



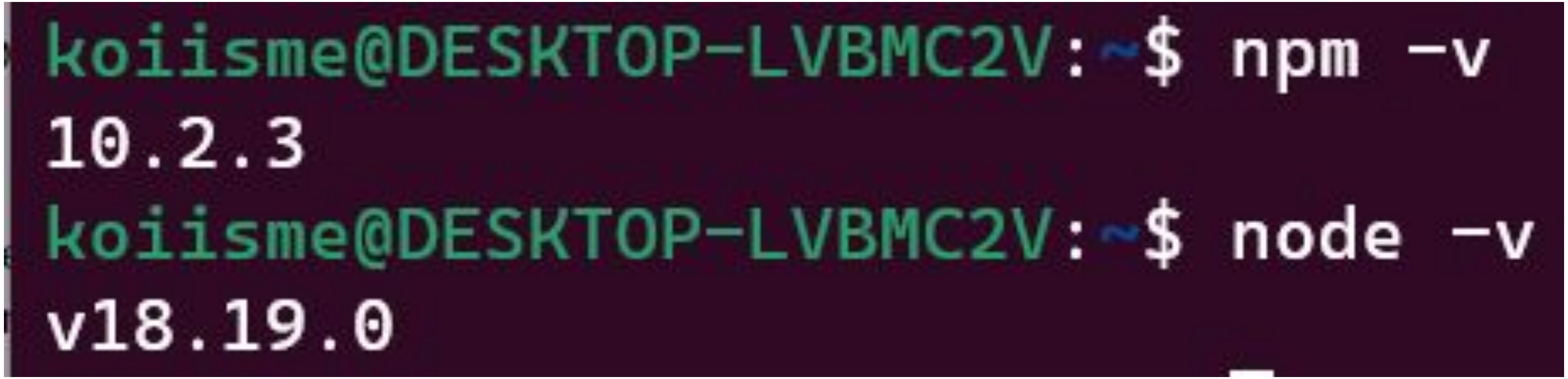
```
koiisme@DESKTOP-LVBMC2V:~$ sudo apt-get install nodejs -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  nodejs
0 upgraded, 1 newly installed, 0 to remove and 26 not upgraded.
Need to get 29.8 MB of archives.
After this operation, 189 MB of additional disk space will be used.
Get:1 https://deb.nodesource.com/node_18.x nodistro/main amd64 nodejs amd64 18.19.0-1nodesource1 [29.8 MB]
Fetched 29.8 MB in 9s (3154 kB/s)
Selecting previously unselected package nodejs.
(Reading database ... 31845 files and directories currently installed.)
Preparing to unpack .../nodejs_18.19.0-1nodesource1_amd64.deb ...
Unpacking nodejs (18.19.0-1nodesource1) ...
```



```
koiisme@DESKTOP-LVBMC2V:~$ sudo apt-get install npm -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

Step 1 (cont)

5. Verify that the Node.js and npm were successfully

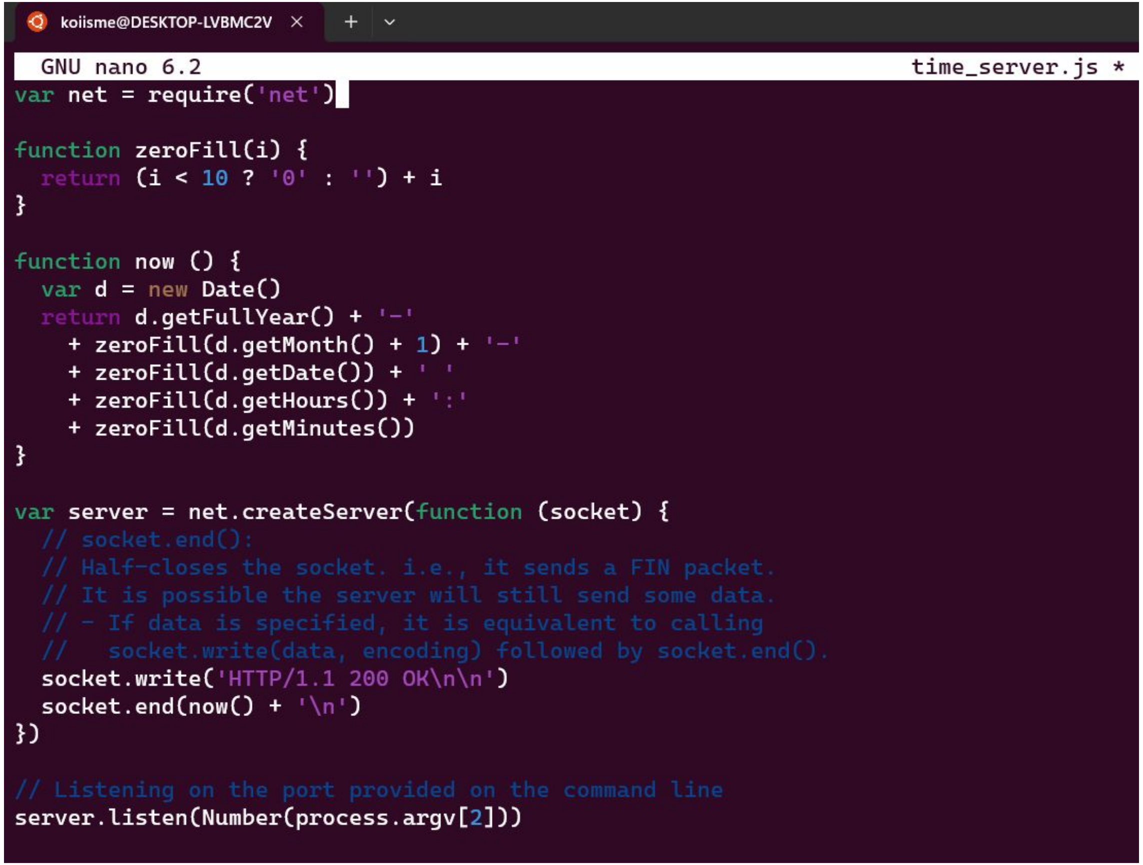


```
koiisme@DESKTOP-LVBMC2V:~$ npm -v
10.2.3
koiisme@DESKTOP-LVBMC2V:~$ node -v
v18.19.0
```

Step 2: Study Time Server

Create a file named ‘time_server.js’

Source code: time_server.js



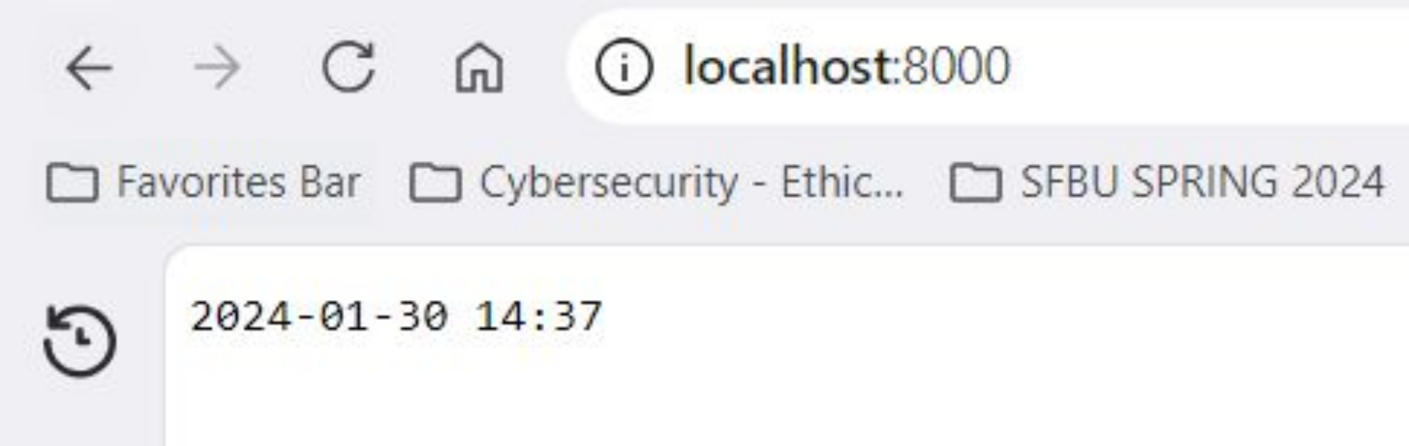
```
  GNU nano 6.2                                                    time_server.js *
var net = require('net')

function zeroFill(i) {
  return (i < 10 ? '0' : '') + i
}

function now () {
  var d = new Date()
  return d.getFullYear() + '-'
    + zeroFill(d.getMonth() + 1) + '-'
    + zeroFill(d.getDate()) + ' '
    + zeroFill(d.getHours()) + ':'
    + zeroFill(d.getMinutes())
}

var server = net.createServer(function (socket) {
  // socket.end():
  // Half-closes the socket. i.e., it sends a FIN packet.
  // It is possible the server will still send some data.
  // - If data is specified, it is equivalent to calling
  //   socket.write(data, encoding) followed by socket.end().
  socket.write('HTTP/1.1 200 OK\n\n')
  socket.end(now() + '\n')
})

// Listening on the port provided on the command line
server.listen(Number(process.argv[2]))
```

Step 2 (cont)

Open Ubuntu terminal and enter:

node time_server.js 8000

Then, open your browser and go to http://localhost:8000, you should see a page with the current time like this



```
2024-01-30 14:37
```

Step 3: Study JSON

Refer to the document about JSON:

https://hc.labnet.sfbu.edu/~henry/npu/classes/javascript/json/slide/index_slide.html

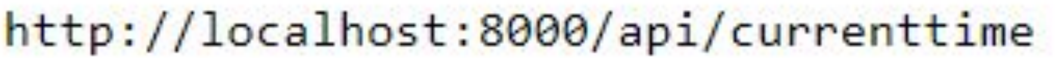
Step 4: Study HTTP JSON API Server

Refer to the document about HTTP JSON API Server:

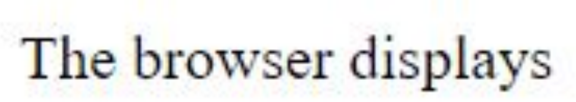
https://hc.labnet.sfbu.edu/~henry/npu/classes/javascript/node_js/course/nodeschool/learnyounode/http_json_api_server.html

Step 5: Modify HTTP JSON API Server

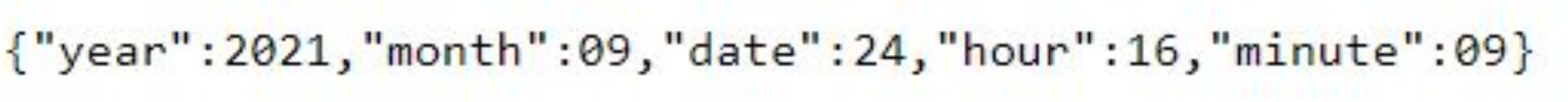
Problem: Step 5: Modify HTTP JSON API Server to support this request from the client side:



```
http://localhost:8000/api/currenttime
```



The browser displays

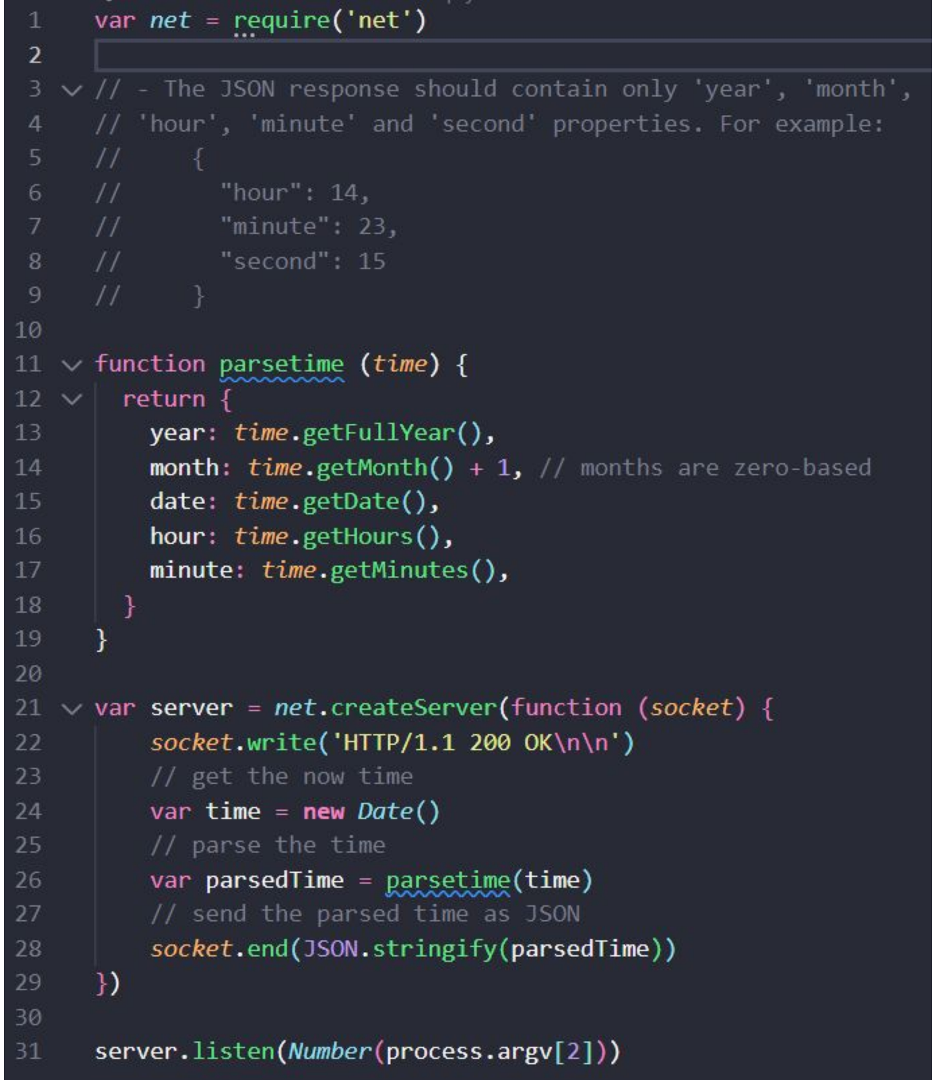


```
{"year":2021,"month":09,"date":24,"hour":16,"minute":09}
```

Step 5 (cont)

Create a file name ‘json_time_server.js’

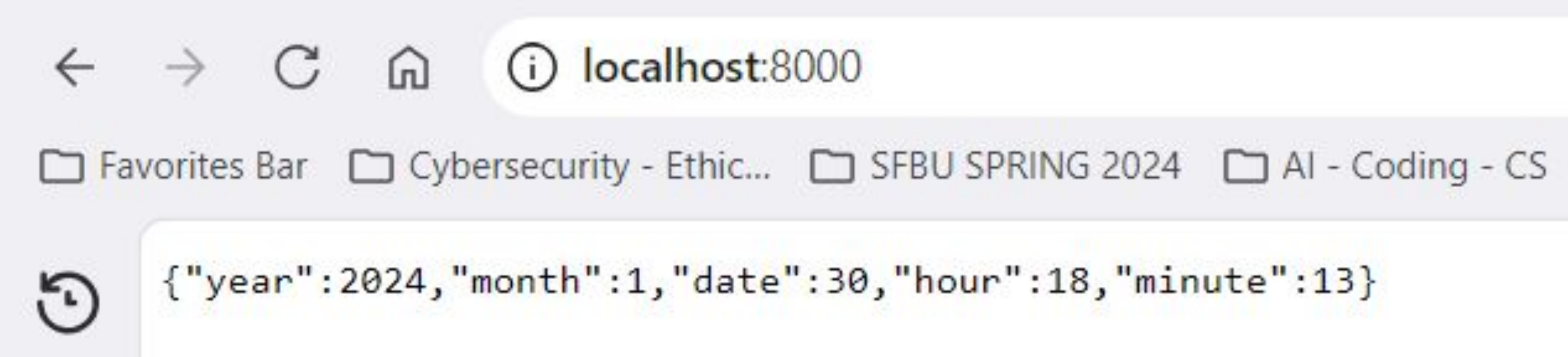
Source code: json_time_server.js



```
var net = require('net')

// - The JSON response should contain only 'year', 'month',
// 'hour', 'minute' and 'second' properties. For example:
//     {
//       "hour": 14,
//       "minute": 23,
//       "second": 15
//     }

function parsetime (time) {
  return {
    year: time.getFullYear(),
    month: time.getMonth() + 1, // months are zero-based
    date: time.getDate(),
    hour: time.getHours(),
    minute: time.getMinutes(),
  }
}

var server = net.createServer(function (socket) {
    socket.write('HTTP/1.1 200 OK\n\n')
    // get the now time
    var time = new Date()
    // parse the time
    var parsedTime = parsetime(time)
    // send the parsed time as JSON
    socket.end(JSON.stringify(parsedTime))
})

server.listen(Number(process.argv[2]))
```

Step 5 (cont)

Now, go to http://localhost:8000/ to check the result



```
{"year":2024,"month":1,"date":30,"hour":18,"minute":13}
```

Source code & references

Problem & guidance:

https://hc.labnet.sfbu.edu/~henry/npu/classes/javascript/node_js/course/nodeschool/learnyounode/exercise_learnyounode.html

GitHub source code:

https://github.com/MynameisKoi/CS571/tree/main/Java/Project/Time%20Server