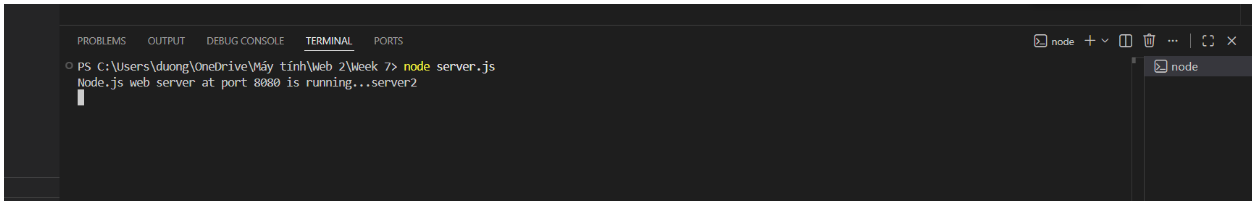
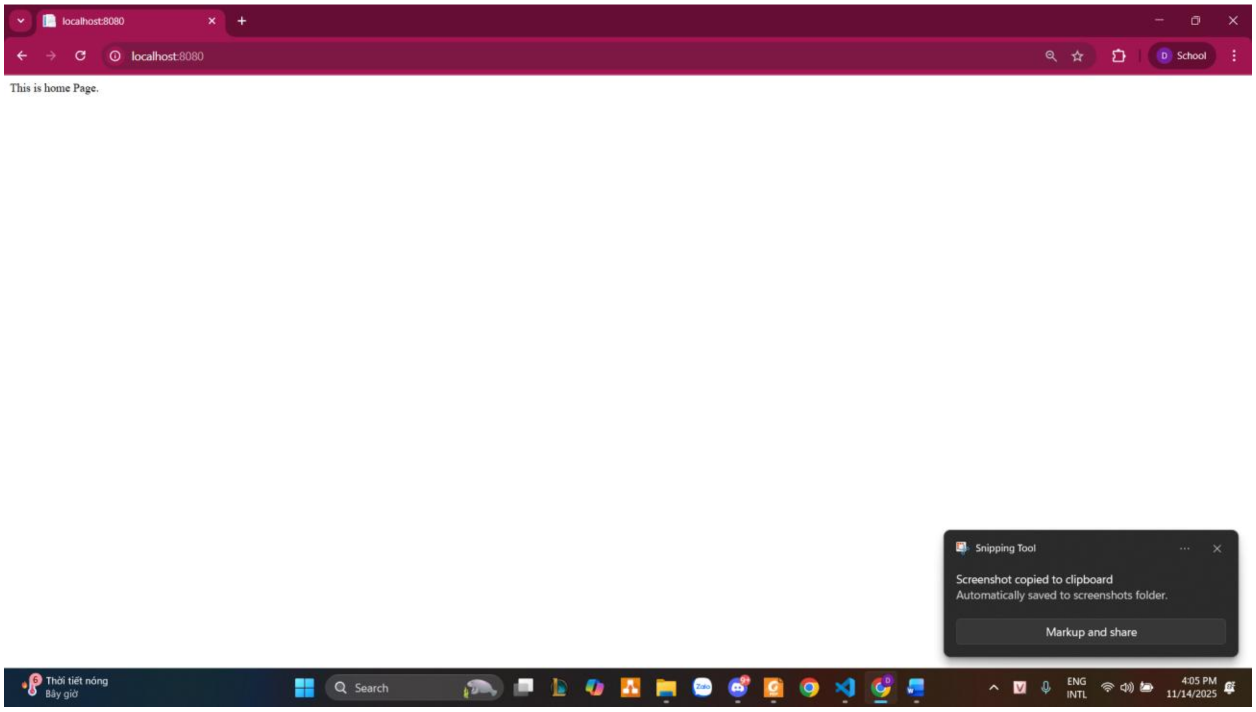
Lab 4

Name: Dương Quốc Anh
ID : GCS230290



```
PS C:\Users\duong\OneDrive\Máy tính\Web 2\Week 7> node server.js
Node.js web server at port 8080 is running...server2
```

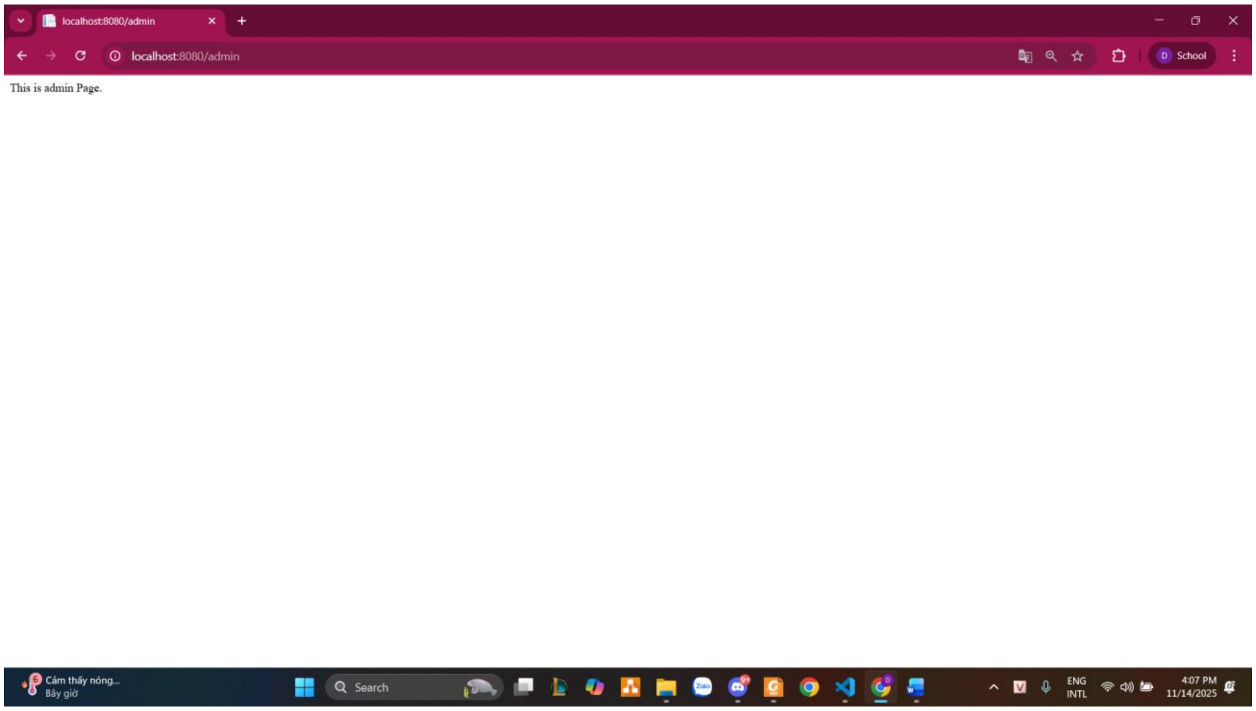
Pic 1.Server.js terminal when successfully Run using node.js data



This is home Page.

Screenshot copied to clipboard
Automatically saved to screenshots folder.

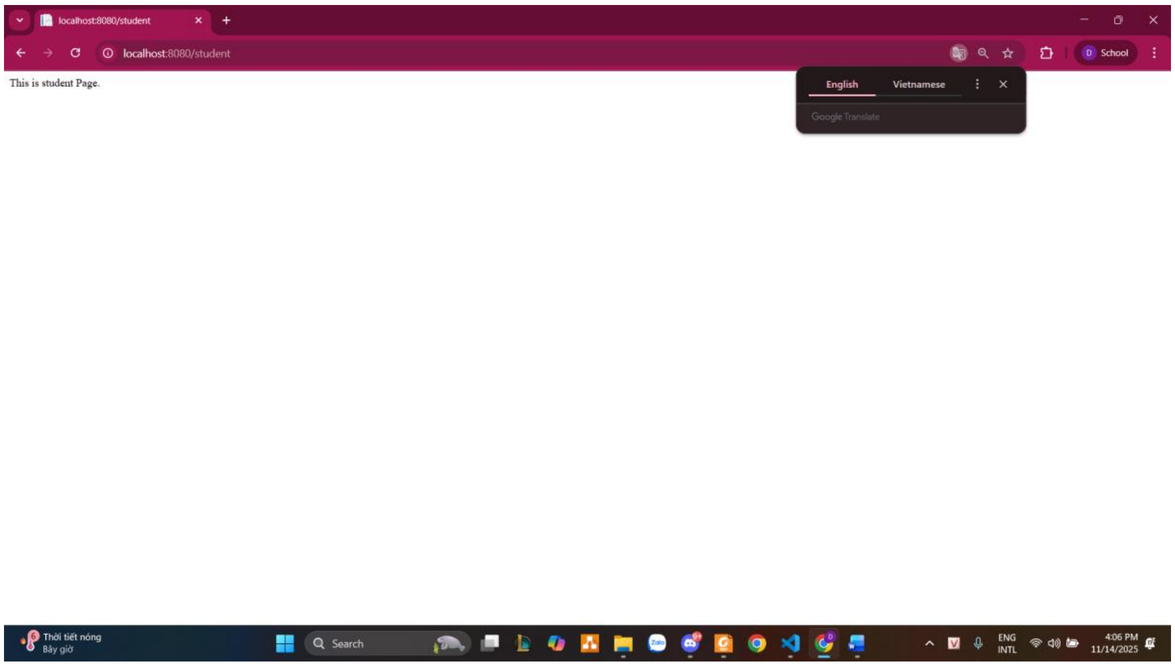
Pic 2.Home Page


This is admin Page.

Pic 3.admin page

Change the address(add “/admin” to show the admin page using (req.url == "/admin"))

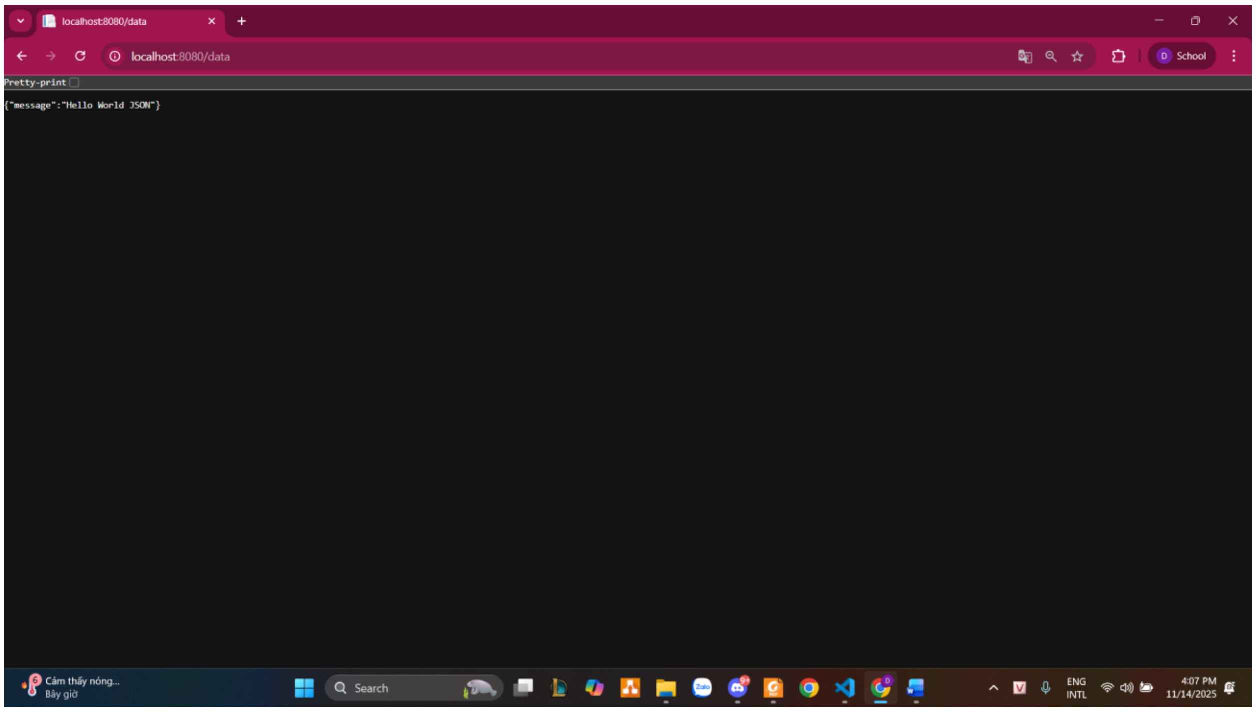
http://localhost:8080 → http://localhost:8080/admin


This is student Page.

Pic 4.Student Page

Change the address (add “/student” to show the admin page using (req.url == "/student"))

http://localhost:8080 → http://localhost:8080/student

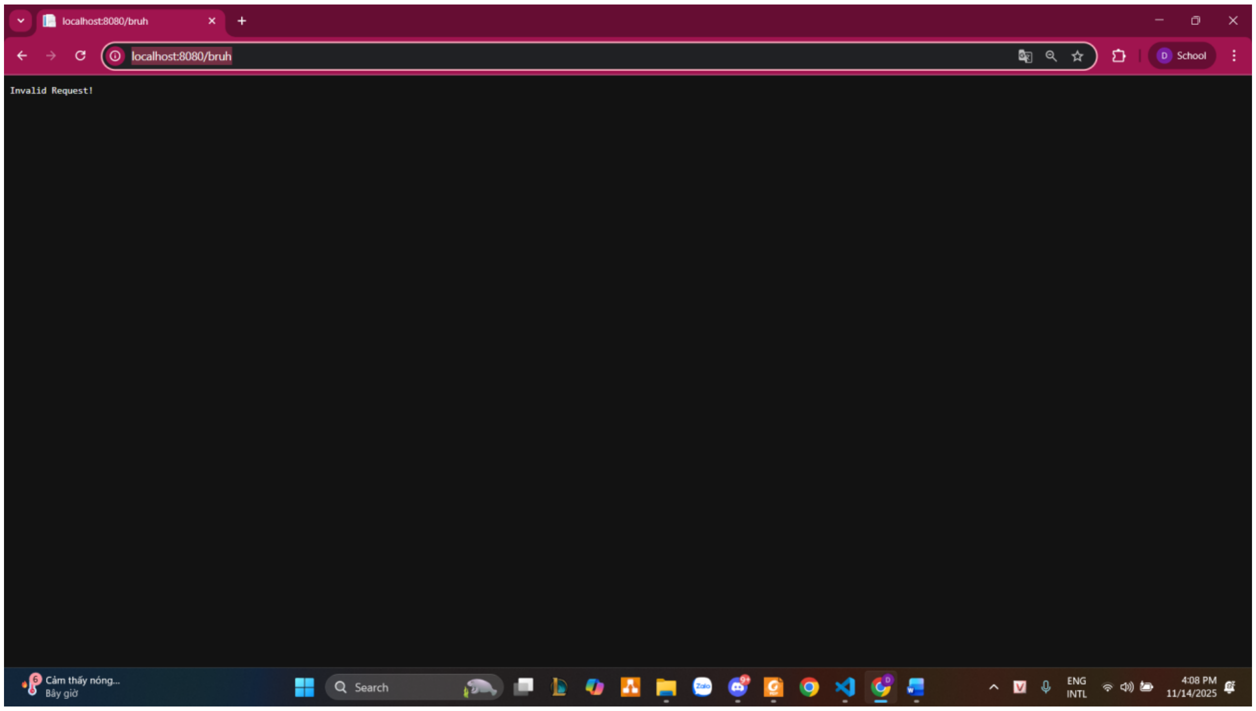


```
{"message":"Hello World JSON"}
```

Pic 5.Data Page

Change the address (add “/data” to show the data page using (req.url == "/data"))

http://localhost:8080 → http://localhost:8080/data



```
Invalid Request!
```

Pic 6.invalid request page

Change the address

(add “/data” to show the data page using (req.url == "/” not in the request URL ”"))

http://localhost:8080 → http://localhost:8080/ “anything except the word that we choose in the Request URL”

My understanding about Node.js

This lesson has taught me that Node.js can act like a web server without using tools like Apache or XAMPP; instead, it uses the http module. While listening on a certain port, say 8080, the server waits for customers to access http://localhost:8080. Node.js checks req.url to decide what to return, be it the home page, student page, admin page, or JSON data. Responses are sent by the server using res.writeHead, res.write, and res.end to return both HTML and JSON. Node.js can provide API-style responses, as with the route called /data. Node.js sends a message to the terminal to confirm that the server is up and running.