Lahti teknillinen yliopisto

School of Software Engineer

Sofware Development Skills

<Hongyue Zhang>, <001219638>

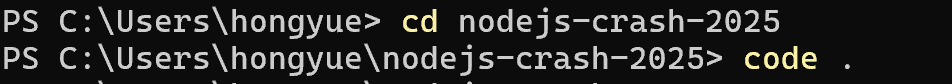
LEARNING DIARY, <FULL-BACK> MODULE

**LEARNING DIARY**

1. 3.2025

I learned what is Node.js and how it works. Node is built on top of the V8 JS Engine and extened the engine to work on the server side. Node is also called non-blocking which means that it doesn’y wait around for to operations.Node also uses event loop which is a mechanism that allows node to perform non-blocking IO operations. During watching the video, I unsderstood what is Node.js used for like APIs, Microservices, and etc.And Node is not good a t CPU-intensive tasks.

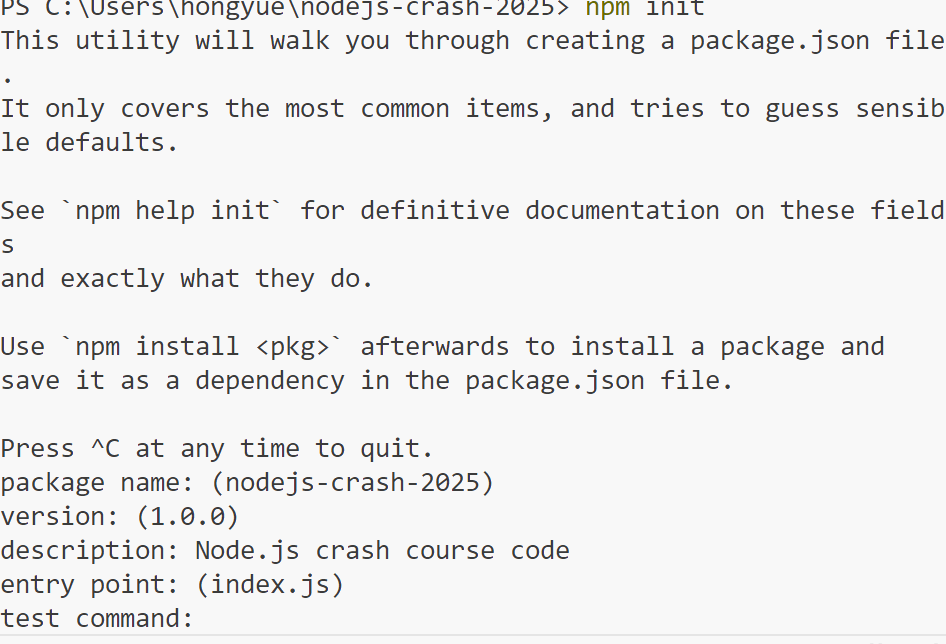
I learned also how to create a node project under command line.



Run PowerShell as an administrator and run the following command in an administrator PowerShell window

Set-ExecutionPolicy RemoteSigned -Scope CurrentUser

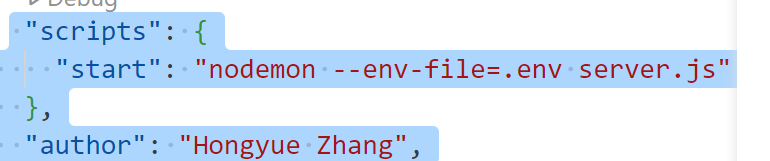
Then I can run the terminal



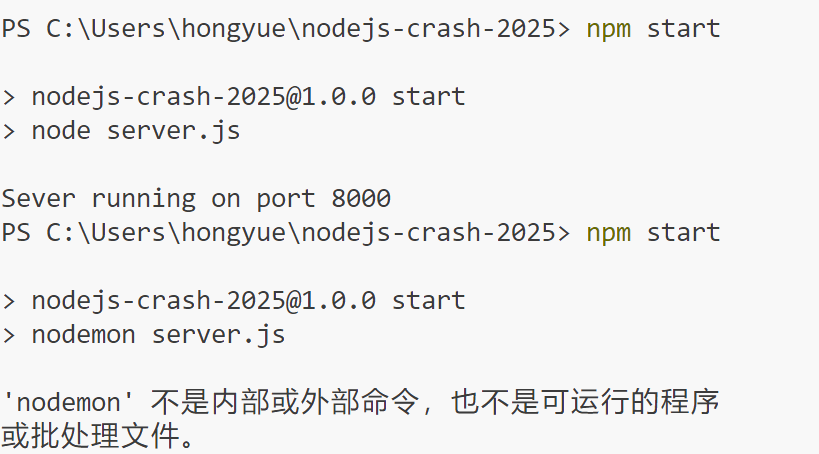
22.3.2025

I learned some basic code about the Node.js. Like, how to export functions.And learned how to create a basic HTTP server, read files asynchronously, and how to handle and return a 404 status when it is not a GET request.And during the development process.

When I chage the json file ,like this



Then it shows:



So I installed the nodemon tool to automatically restart the server.

Then run the command:

npm start

The server can run automayically.

Then I installed the postman. Used to test HTTP requests. I use postman to send a get request to get data, and submit the data by using post method.

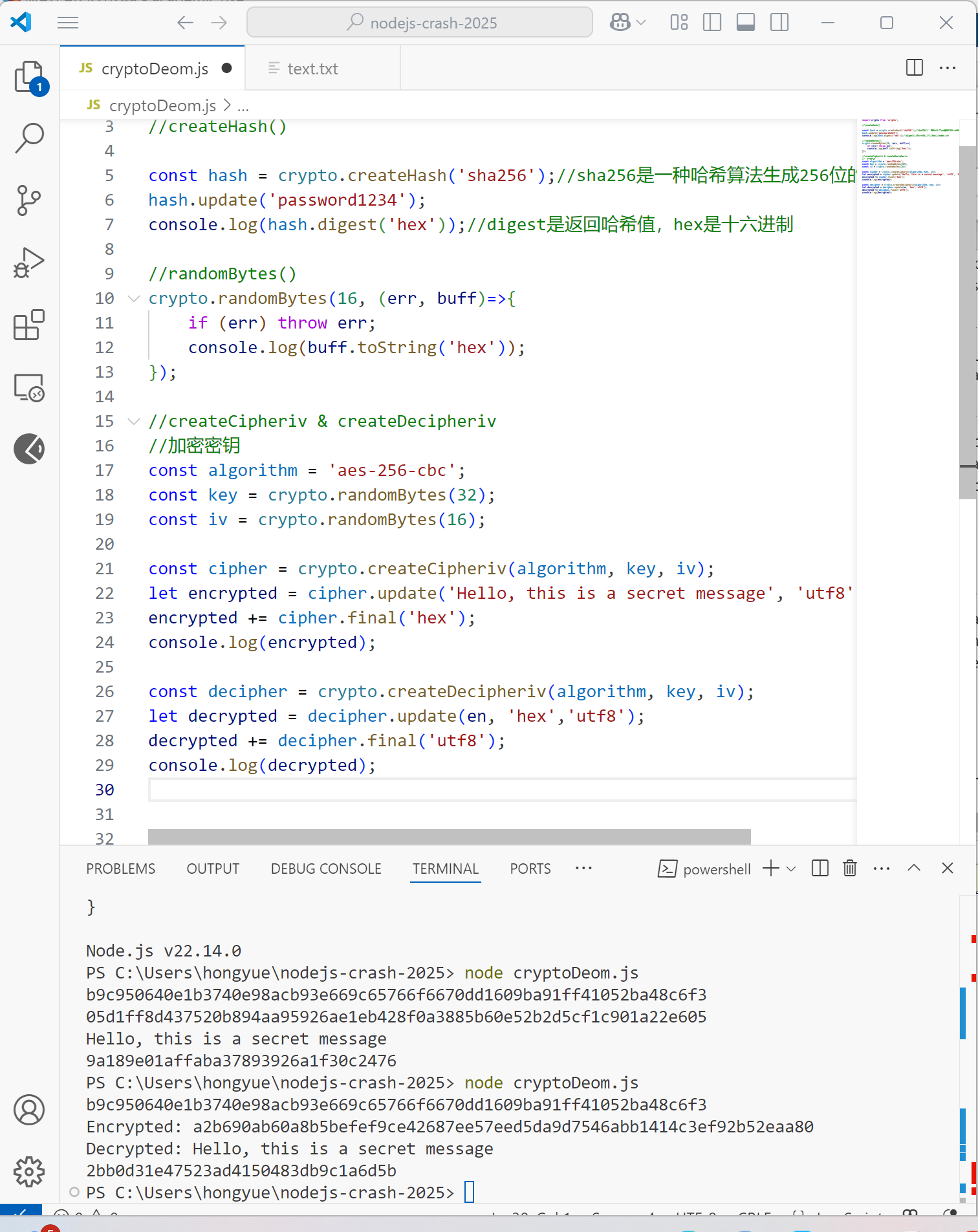
24.03.2025

Today I learned how to handle /api /users and find the user id in the user array that matches the extracted ID.

I also learned about HTTP servers and their middleware. For example, I created a middleware to record the method and url of each request to control the server. I also learned how to handle POST requests. I can use req.on('data') to receive data, and then use req.on('end') to process the data after receiving it, and add the new user data to the user array.

In addition, I also learned the URL, fs, and os modules. In the url module, I understood how to process URL strings, parse query strings in urls, and convert file URLs into paths, etc.

But when I learned crypto module, I have some troubles,like:



The last line of the output should be the decrypted message, but it showed an encrypted value.

I fixed the issue by rewatching the video again. This is because randomByte will randomly generate a 16-byte string, which has nothing to do with the encryption and decryption method, just delete it.

Use process.argv to get command line parameters such as import, -s, etc.

