**Day 1 Node Basics**

Installing Node in Operating System

* **Step 1: Download the Installer**

Download the Node installer from the official site.

A screenshot of a computer

Description automatically generated

* **Step 2: Run the installer.**

Run the installer and click Next until the setup wizard is complete.

A screenshot of a computer

Description automatically generated

A screenshot of a software license agreement

Description automatically generated

A screenshot of a computer

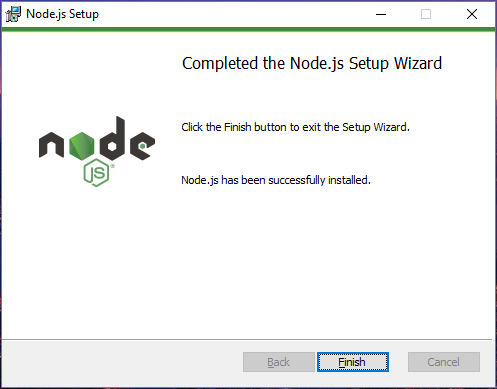
Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated



* **Step 3: Verify the installation**

Open command prompt or PowerShell and run the following command. node -v It should return the node version.

A screenshot of a computer

Description automatically generated

* **Step 4: Update the NPM**

**Day 2 Building Web Server**

Build a simple Web application that prints Hello World on the

browser upon user's request.

A screen shot of a computer program

Description automatically generated

A screenshot of a chat

Description automatically generated

A screen shot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer screen

Description automatically generated

**Day 3 Node modules**

We can install node modules using node package manager or called npm.

You can also install the module and write it in your package.json file

using the following command.

A screen shot of a computer

Description automatically generated

You can create your own node module as well. All you need to do is

create a function and export it for the reusability.

For example, consider this code which can act as a node module.

A screen shot of a computer

Description automatically generated

A screenshot of a computer

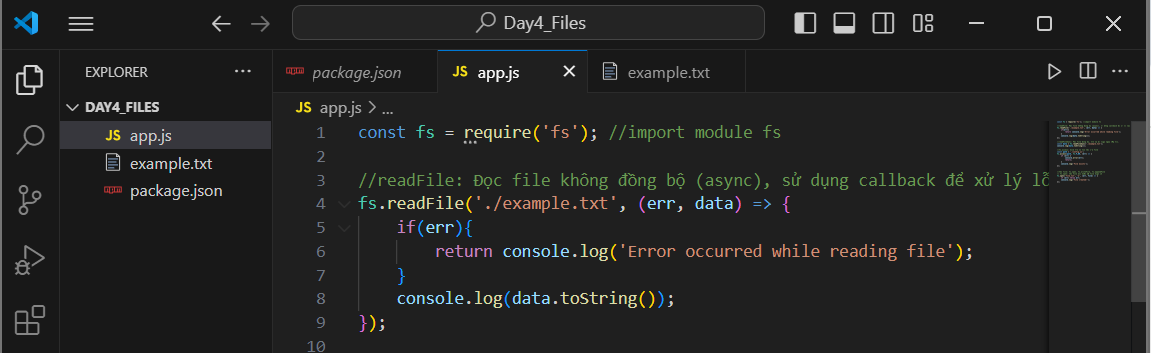
Description automatically generated

**Day 4: Files**

1. **Read File**

You can use fs.readFile() or fs.readFileSync() method to read files

* **readFile**: Read files asynchronously (async), using callbacks to handle errors and data.



* **readFile**: Read files asynchronously (async), using callbacks to handle errors and data.

A screen shot of a computer

Description automatically generated

1. **Check the existence of the file**

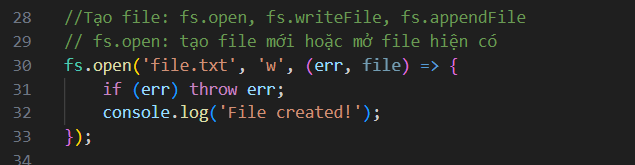
Use fs.access to check if a file exists without taking up much memory**.**

**A screen shot of a computer program

Description automatically generated**

1. **Create file (fs.open, fs.writeFile, fs.appendFile)**

* **fs.open:** Create a new file or open an existing file.

****

* **fs.writeFile:** Write data to the file. If the file does not exist, a new one will be created.

**A black screen with text

Description automatically generated**

* **fs.appendFile:** Add data to the end of the file. If the file does not exist, a new one will be created**.**

**A black screen with text

Description automatically generated**

1. **Delete file (fs.unlink)**

**A computer screen shot of text

Description automatically generated**

1. **Rename file (fs.rename)**

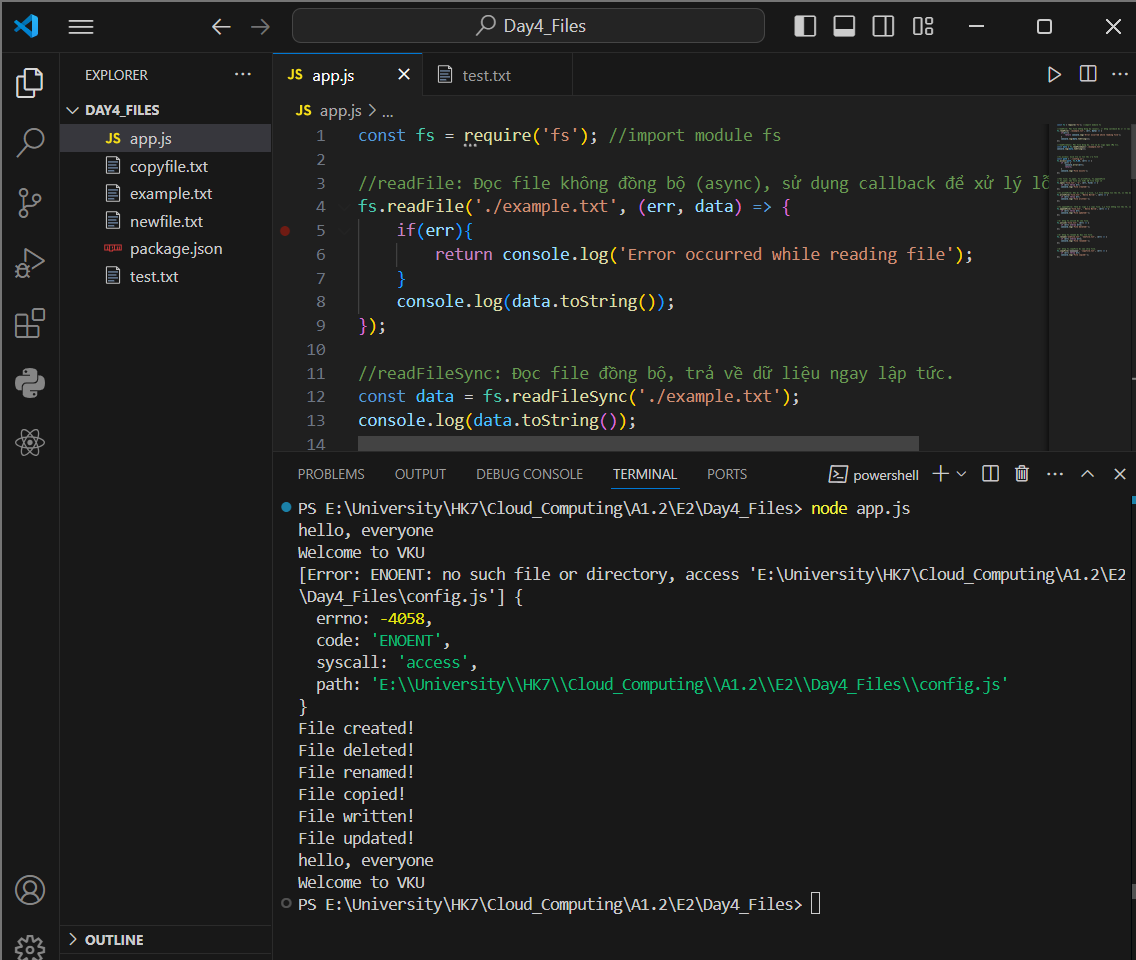
**A computer screen with text

Description automatically generated**

1. **Copy file (fs.copyFile)**

**A screen shot of a computer code

Description automatically generated**

****