**How to create a Node server ?**

* First of all, if we want to create a node server than we need to install node in our system
* After then, open cmd and write the command 🡺 node –v

And it will see the version of node installed in our system

* Then we need to know about the command line of the system like – cd , mkdir, dir etc
* Then create a js file and type some code and run it(**ctrl+alt+n**) to the terminal
* Then we need to know about some module of node
* There are some kind of module we will use in node :

1. Local module (which we create and use it of our own)
2. Built-in module(node js own module) -> http,fs,url,path (no need to install)
3. External module (managed by npm) (npm-> node package manager)

* **Local module**= which module we create in other js file and export it and which we import in our file
* **Build-in module**= which module we don’t need to install or import form other file. It is build in module in node. http,fs,url,path.

**fs(File System)**:

fs module is the build in module of node

which file we want to create, read, update, overwrite,delete

is done by fs module.

|  |
| --- |
| writeFile()  --  কোনো ফাইল থাকলে সেখানে নতুন লেখা যুক্ত করা অথবা নতুন ফাইল                   ক্রিয়েট করে ওইখানে লেখা যুক্ত করা  appendFile() -- লেখা overWrite না করে ওইখানে  readFile()  -- txt ফাইল এ কি আছে সেটা বলে দেওয়া -- eikhane obossoi utf-8         ব্যাবহার করতে হবে  rename()   -- txt file rename kora  unlink()   -- file delete korar jnno  exists()   -- কোনো file আছে কিনা সেটা খুজে বের করা |

how to use fs module of node js====>>

//declare the fs module ==>

let s=require('fs')

//writeFile()==>

s.writeFile("fileName.txt","what u want to write",(err)=>{

        if(err)

        {

            console.log("show error if there is an error")

        }

        else{

            console.log("successfully inserted")

        }

})

//appendFile()==>

s.appendFile("fileName.txt","what u want to overwrite in the txt file ",(err)=>{

        if(err)

        {

            console.log("show error if there is an error")

        }

        else{

            console.log("successfully overWrited")

        }

})

//rename()==>

s.rename("fileName.txt","renamedFile.txt",(err)=>{

    if(err){

        console.log("if file is not renamed")

    }

     else {

        console.log("file renamed succesfully")

     }

})

//readFile()==>

s.readFile("fileName.txt","utf-8",(err,data)=>{

    if(err)

    {

        console.log("error is found")

    }

    else{

        console.log(data)

    }

})

exists()==>

s.exists("fileName.txt",(found)=>{

    if(found)

    {

        console.log("file is found")

    }

     else{

        console.log("file is not found")

     }

})

\*/

let s=require('fs')

s.readFile("demo.txt","utf-8",(err,data)=>{

        if(err)

        {

            console.log("error is found")

        }

        else{

            console.log(data)

        }

})

**http**:

http is the build in module of node to create a server in our system.

How to create a server using http:

|  |
| --- |
| // how to create a server    let http=require('http') //initialize http module  let hostname='127.0.0.1'  let port=2000  let server=http.createServer((req,res)=>{          res.end("<h1>How are u? i am Fine</h1> ")  })  server.listen(port,hostname,()=>{      console.log(`i am corrently in https://${hostname}:${port}`)  }) |

If we run the terminal using **ctrl+alt+n** , then the server starts

And if we go to the bowser and type hostname:port (127.0.0.1:2000)

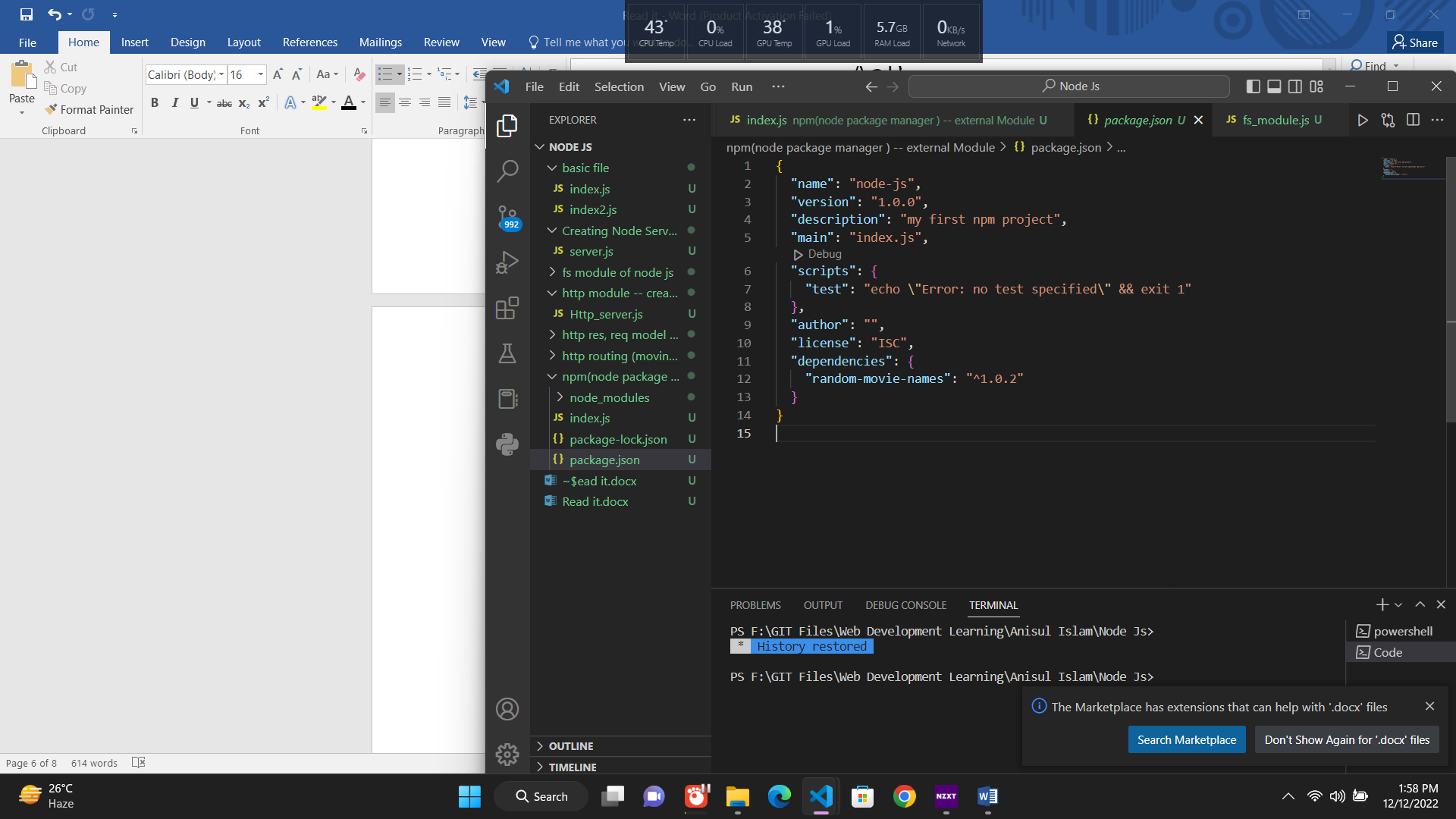
Then we can see the server

* External Module(managed by npm) (npm=node package manager):

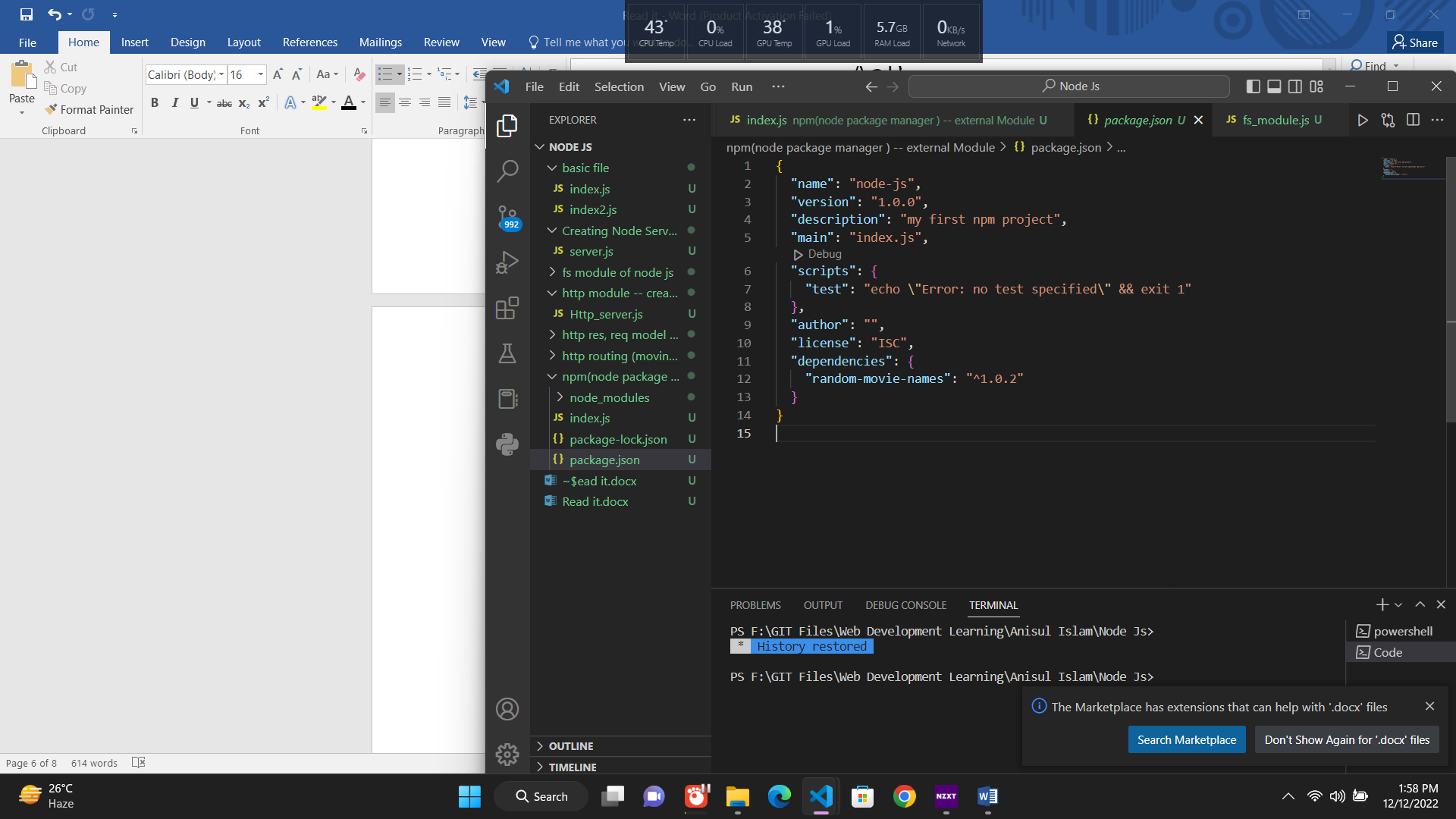
External modules are the modules which we need to install in our project before using it

In order to using external module in our project first we need to do something in our project=>

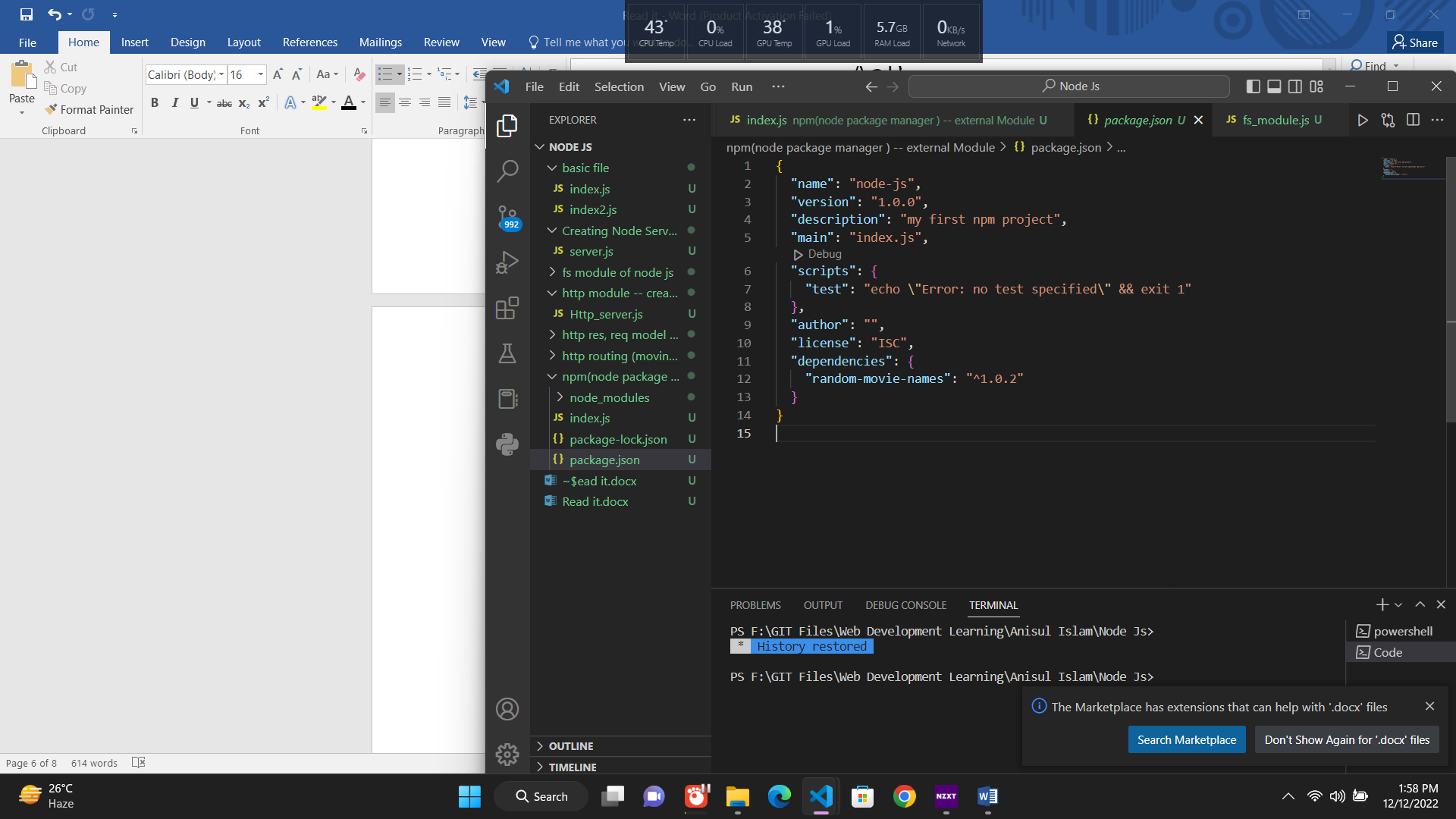
* In the command line , type— **npm init** and press enter and then it will give u some instruction one by one, follow the instruction and type y to initialize successfully. If you want to skip the instruction the just type **npm init –y**  . and then package.json installed



* After initializing , after that what npm  we want to use just search in the npm website and then go to our terminal and write -- **npm install file\_name** (npm install randomMovieName). And then package-lock.json installed



* after installing the external package we can see in package.json and in the dependency: file is installed and its version



**Which external module we want to install :**

**nodemon** npm package

**Why we use nodemon package in our server ?**

=

when we create a server and host it and if we want to change something than we need to stop the server and again start the server .

To avoid this kind of things we need to install nodemon package in our server so that the server can automatically update itself