**Hướng dẫn restfull webservice với**

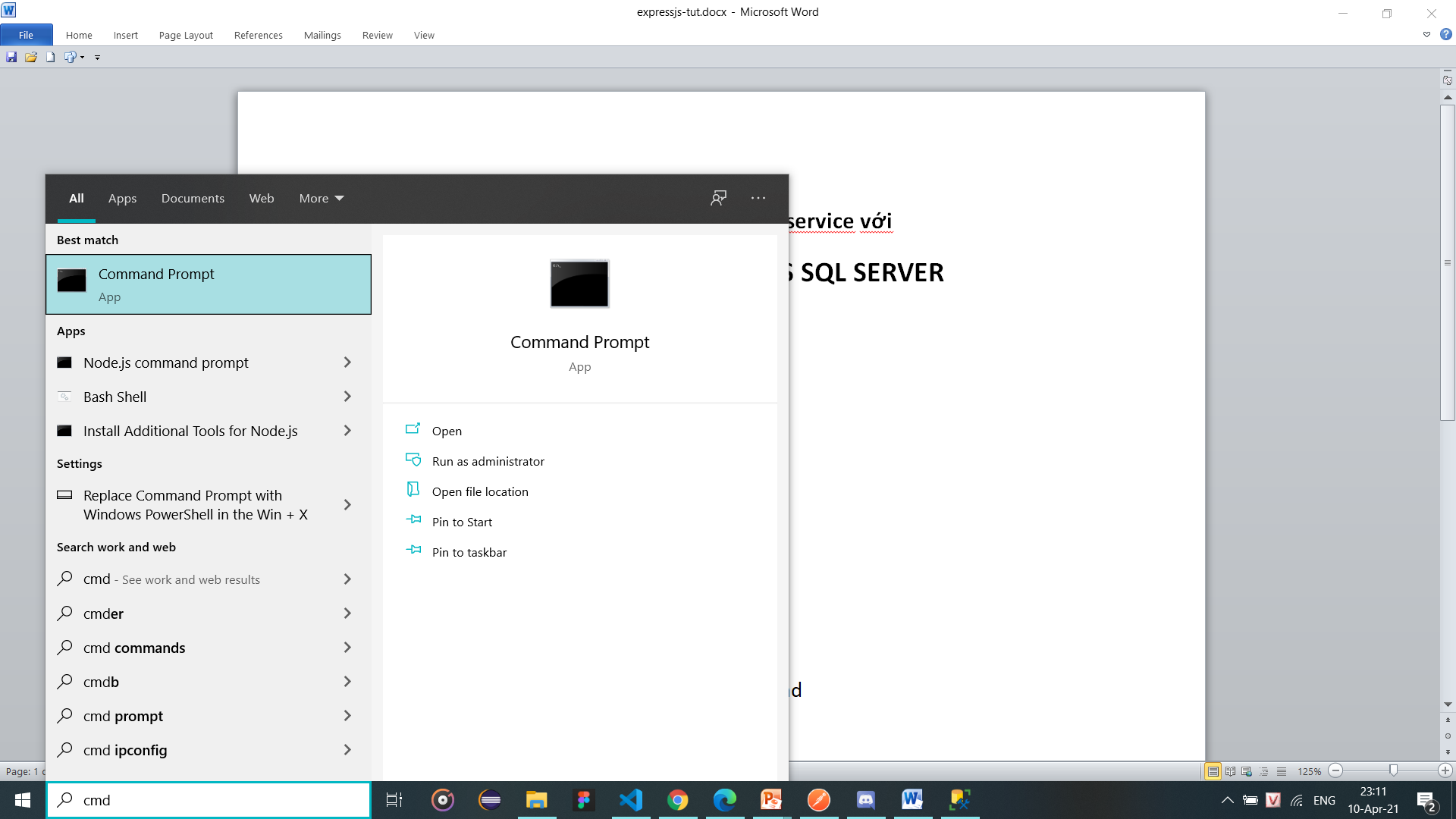
**NODEJS EXPRESSJS + MS SQL SERVER**

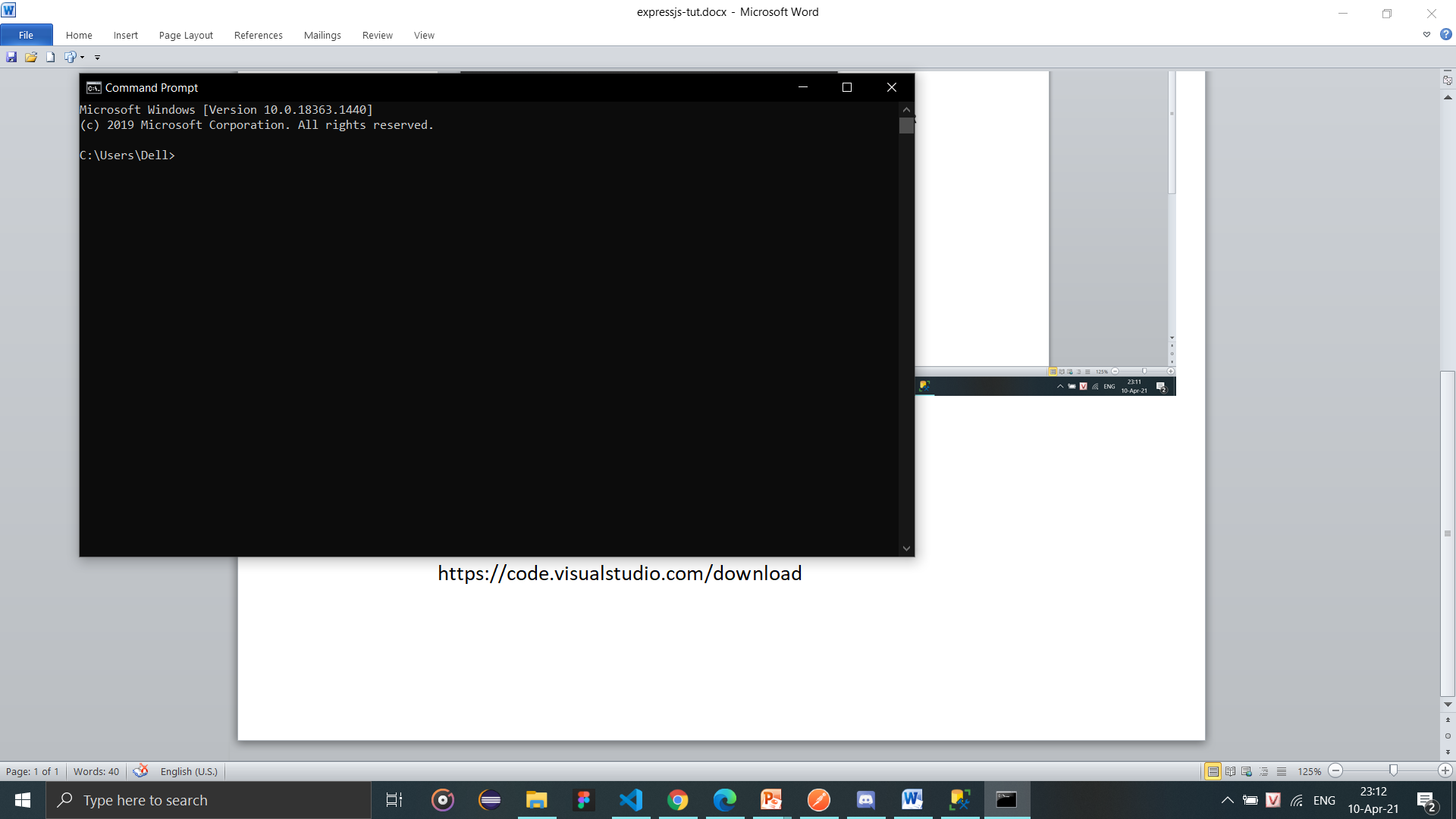
1. Cài đặt công cụ
   1. Cài node js

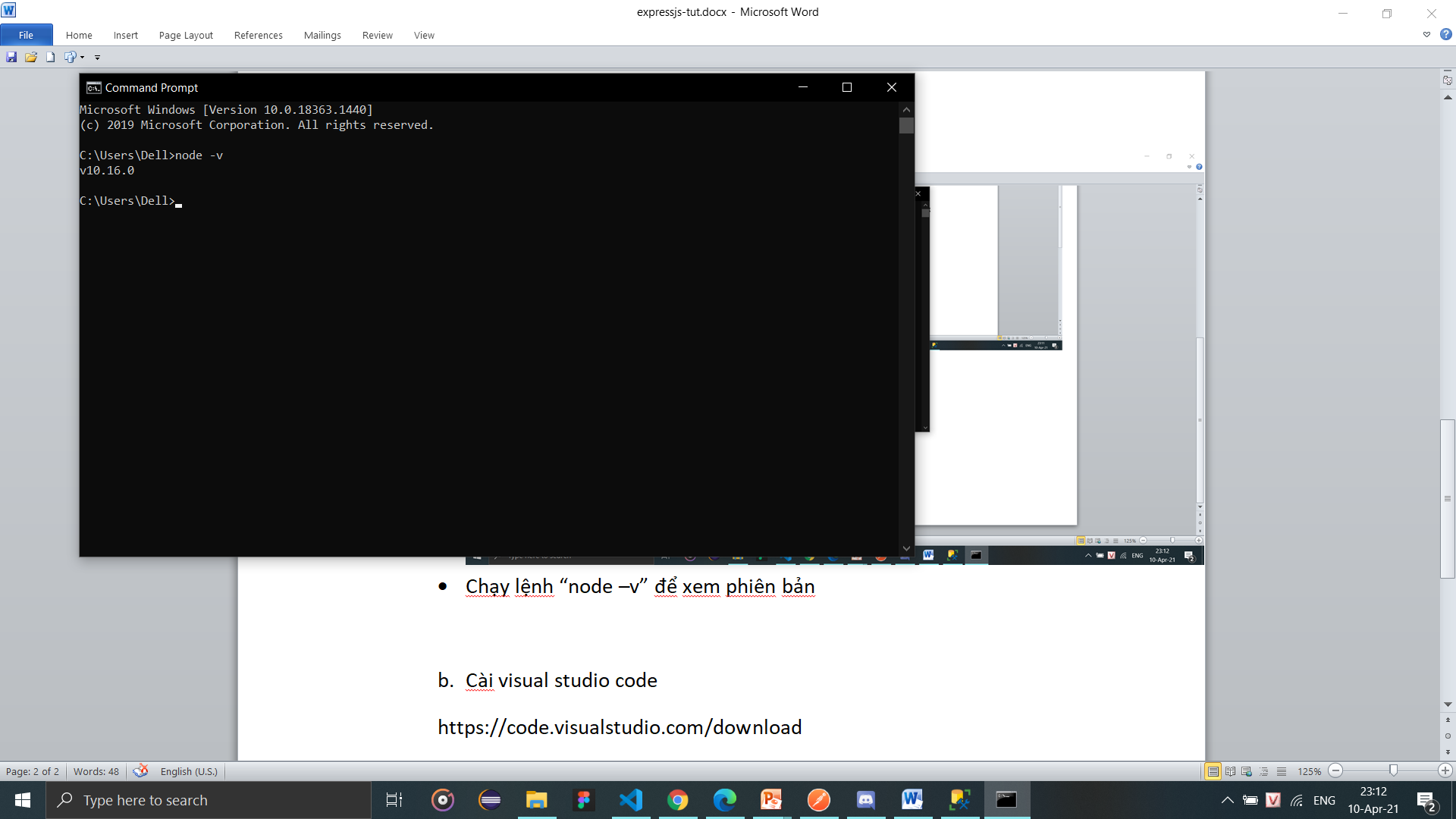
Link cài đặt: <https://nodejs.org/en/download/>

Test cài đặt thành công:

* Mở cửa sổ cmd:



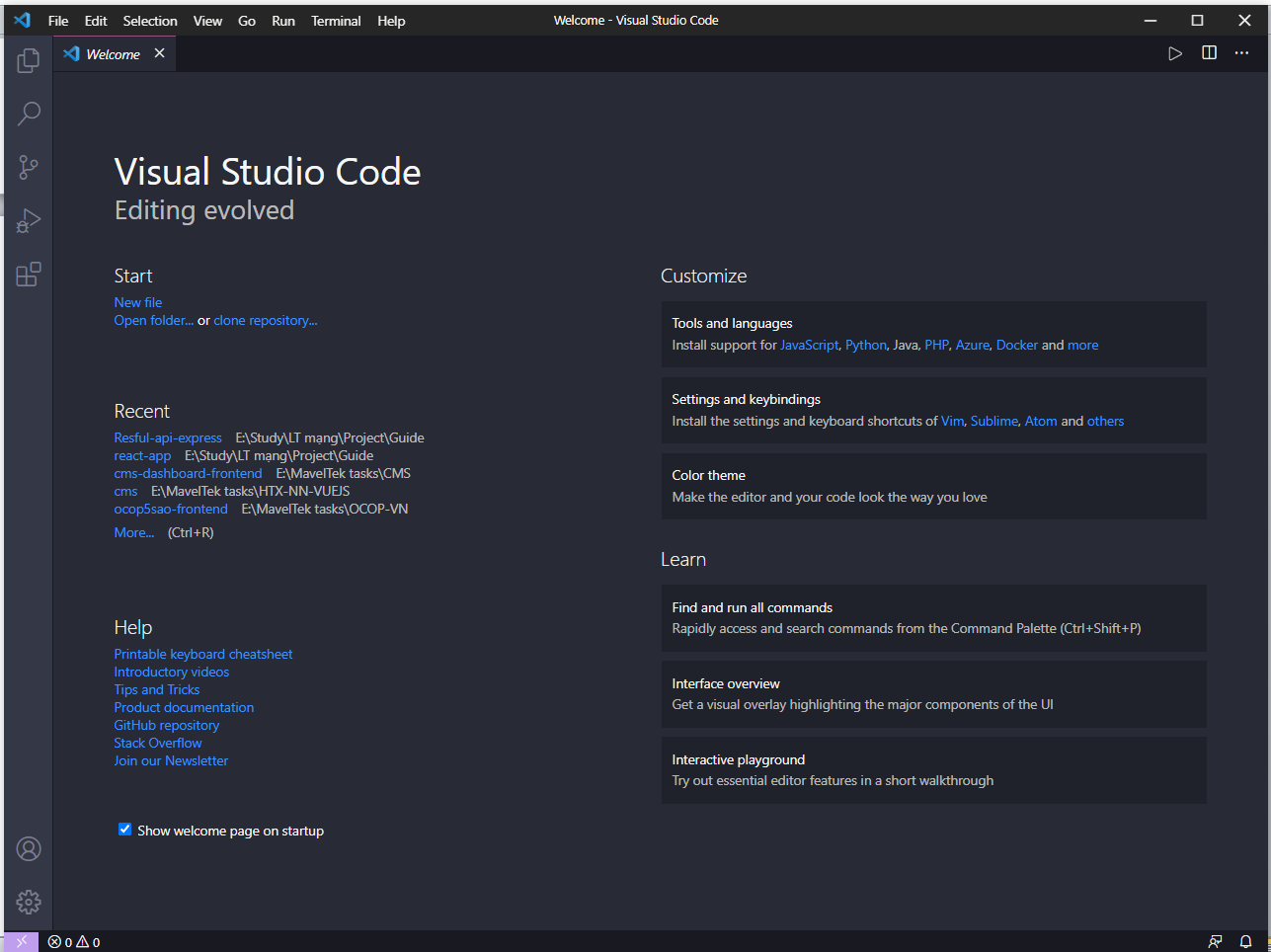
* Enter 
* Chạy lệnh “node –v” để xem phiên bản



* Vậy là cài đặt thành công
  1. Cài visual studio code

<https://code.visualstudio.com/download>

Giao diện Visual studio code

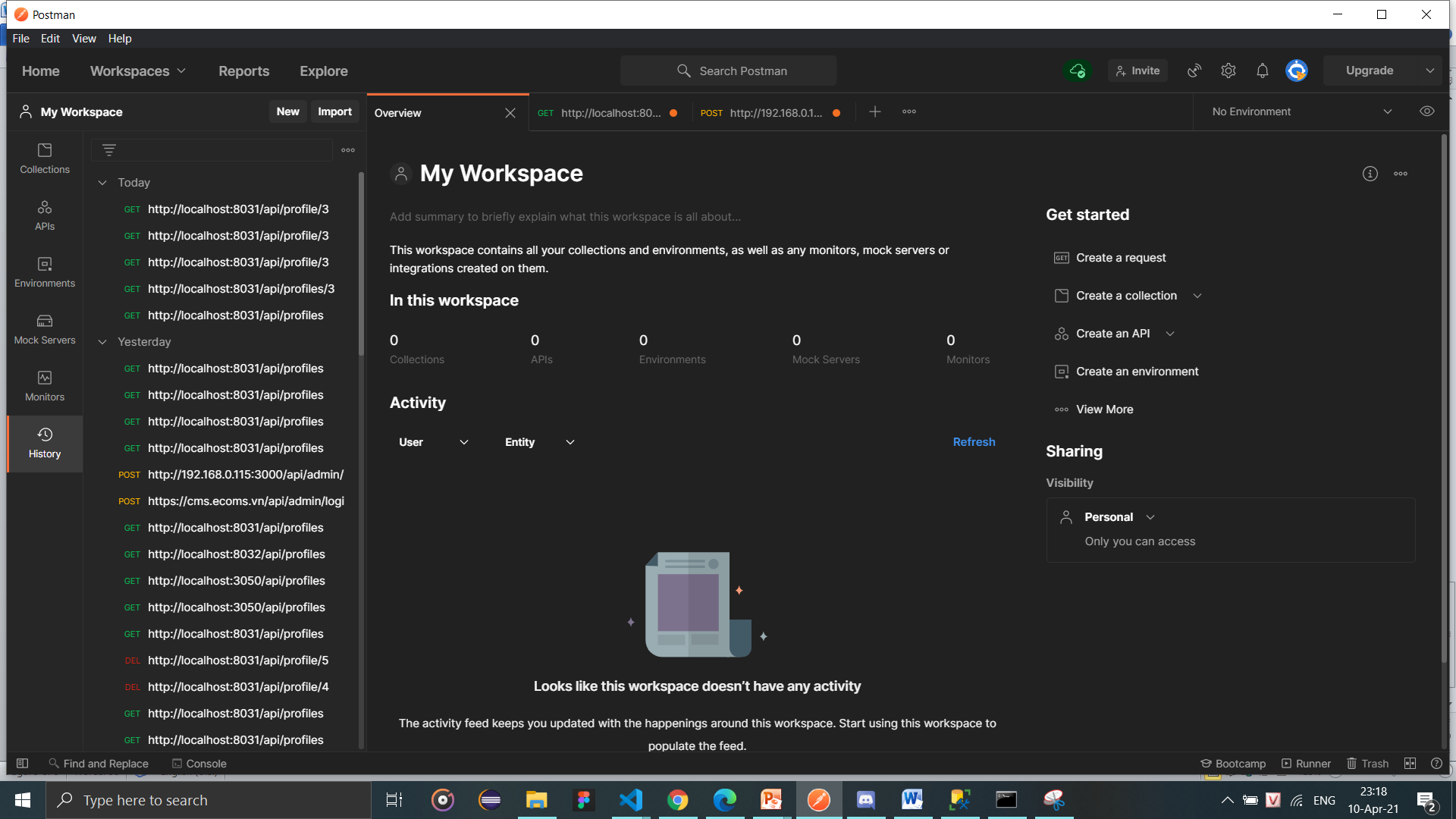


* 1. Cài đặt Microsoft SQL Server. || Cái này tự cài nha =))
  2. Cài đặt Postman để test api

Link cài đặt: <https://www.postman.com/downloads/>

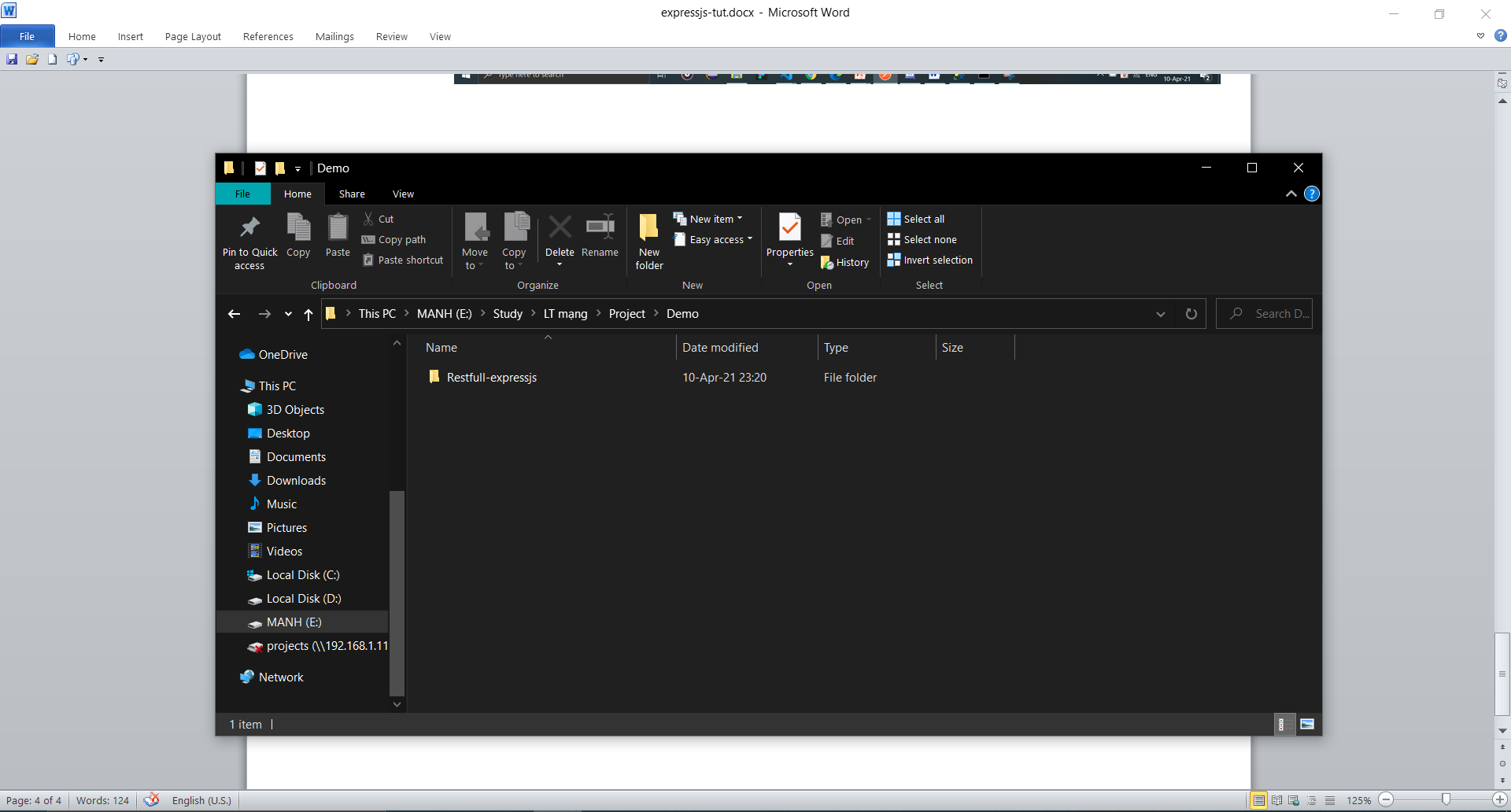
Yêu cầu đăng nhập thì cứ đăng nhập bình thường thôi nha

Giao diện Postman theme dark có thể cài đặt ở phần setting



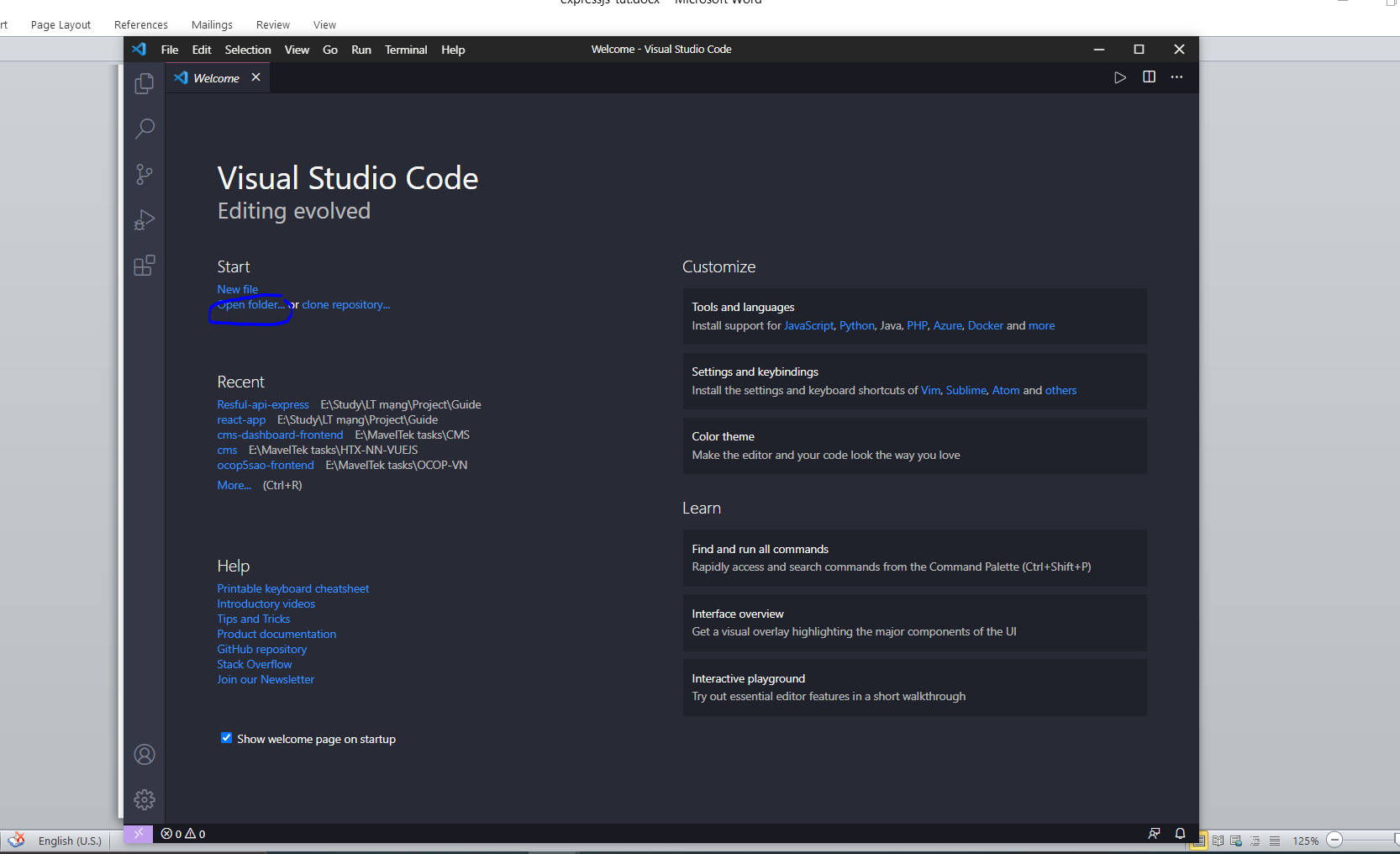
Ok vậy là xong phần công cụ

1. Cài đặt chương trình
2. Khởi tạo project

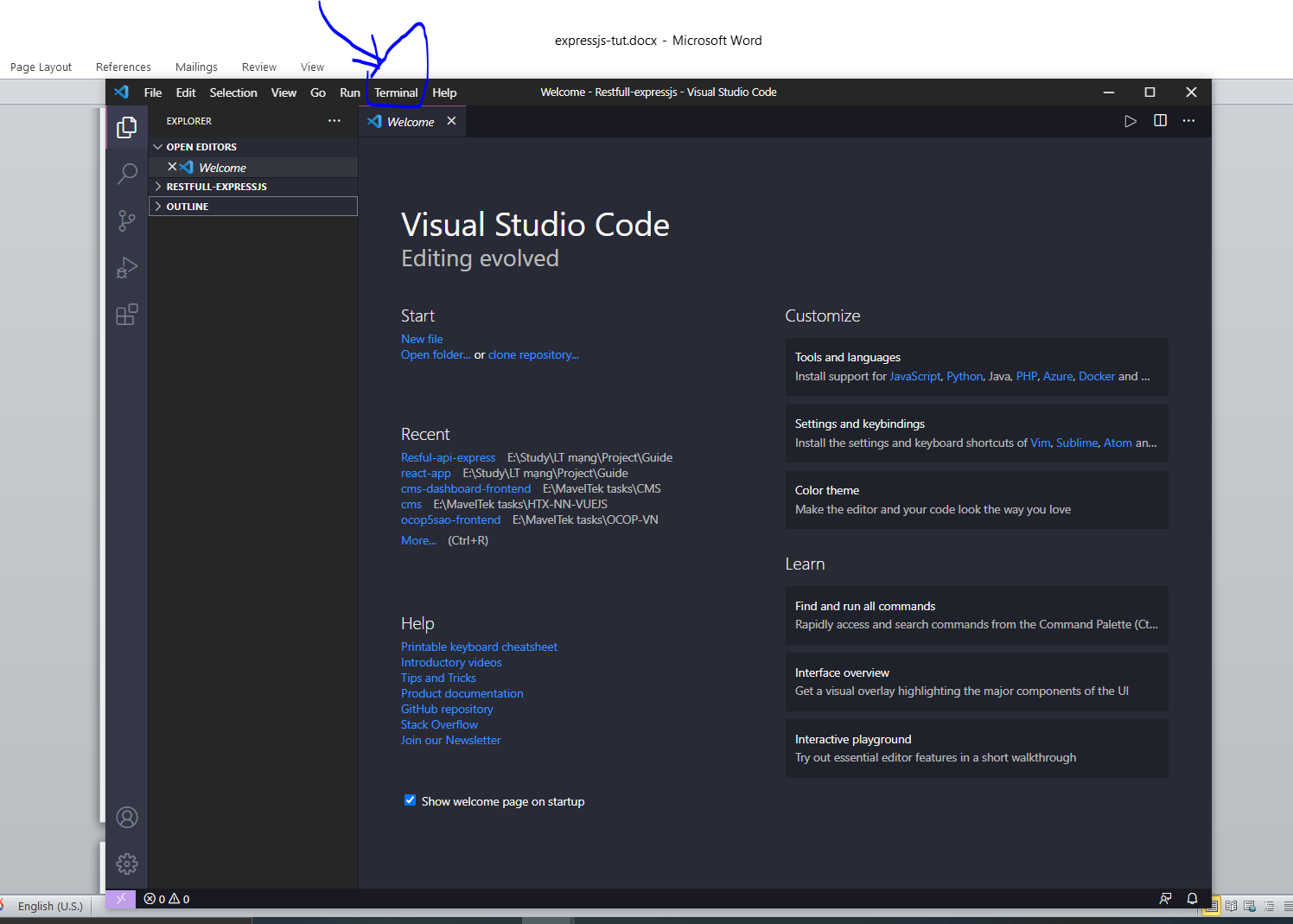


Tạo thư mục rỗng với tên tùy ý như t là Restfull-expressjs

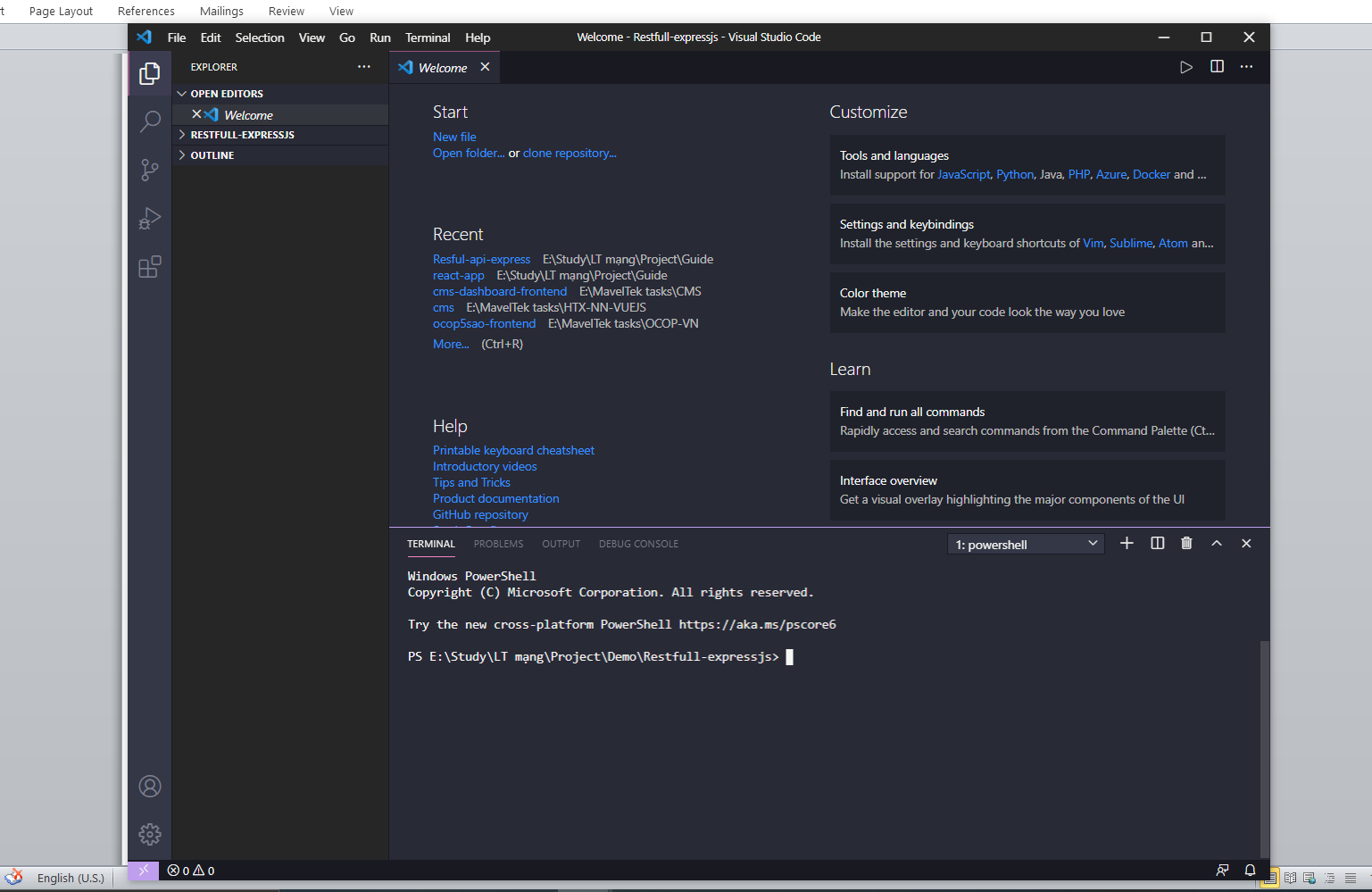
* Khởi tạo project bằng cửa sổ cmd như sau:



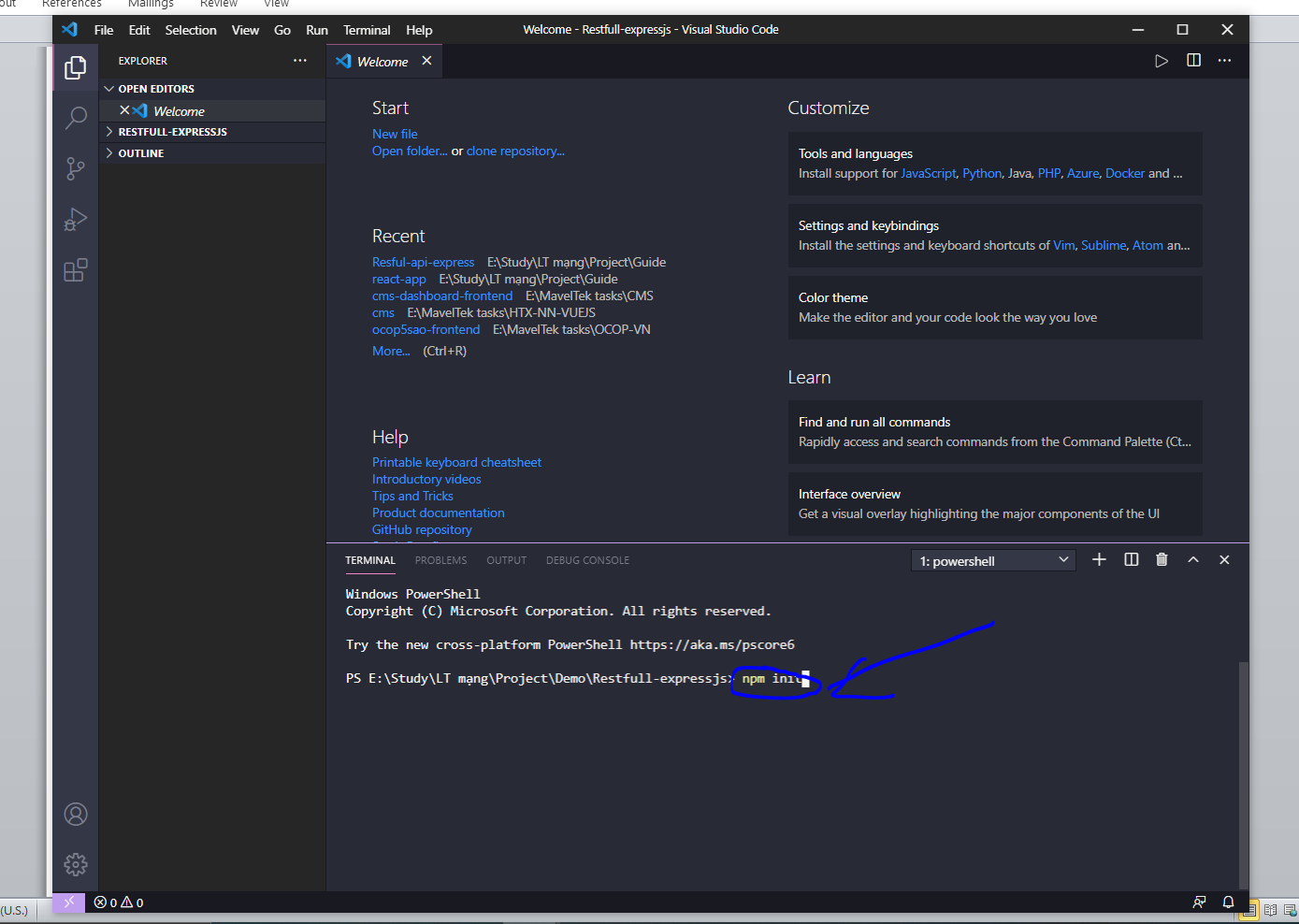
* Open folder rồi chọn folder của mình
* Sau đó mở “Terminal” => “New Terminal”



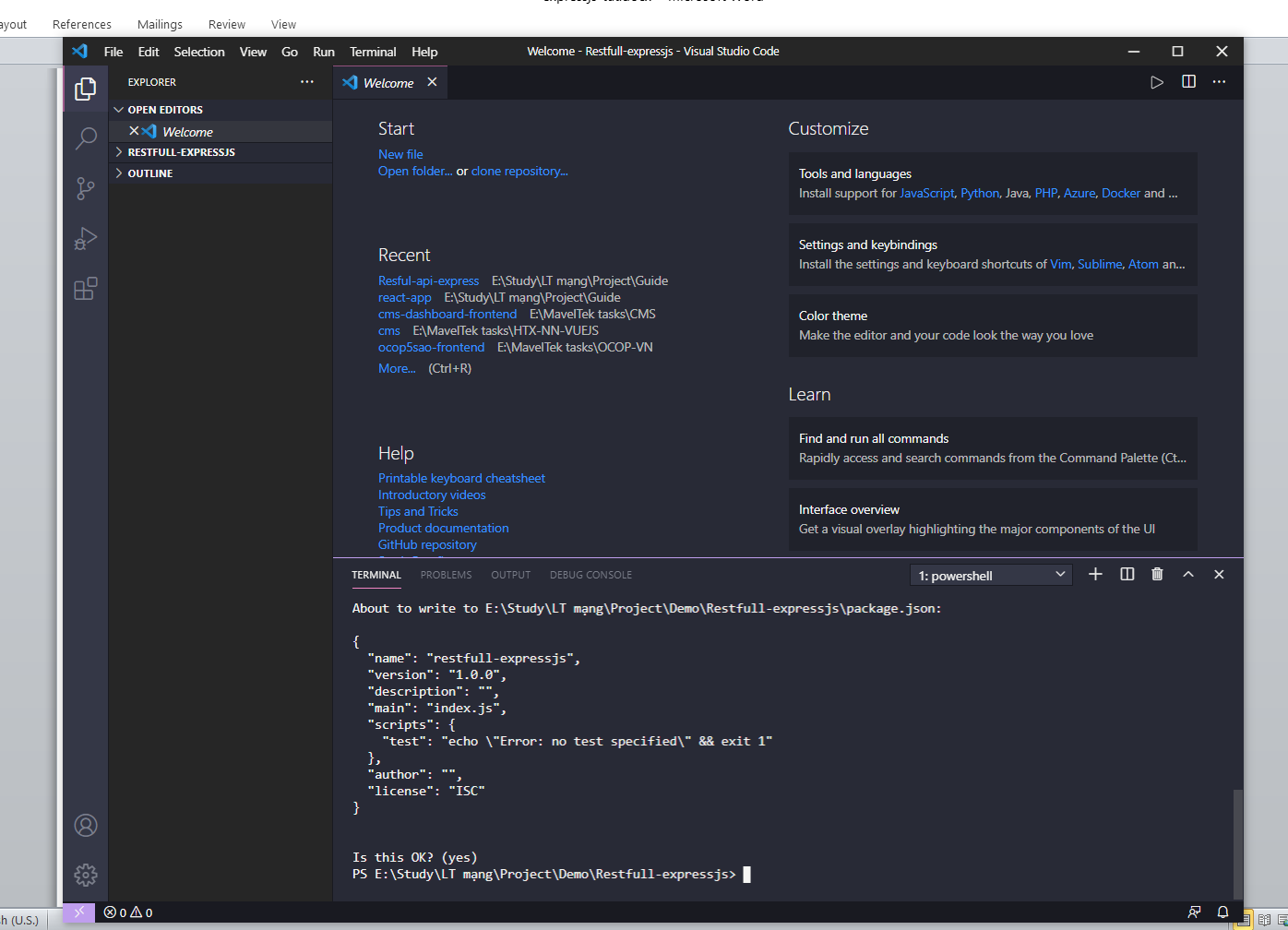
* Được kết quả như sau



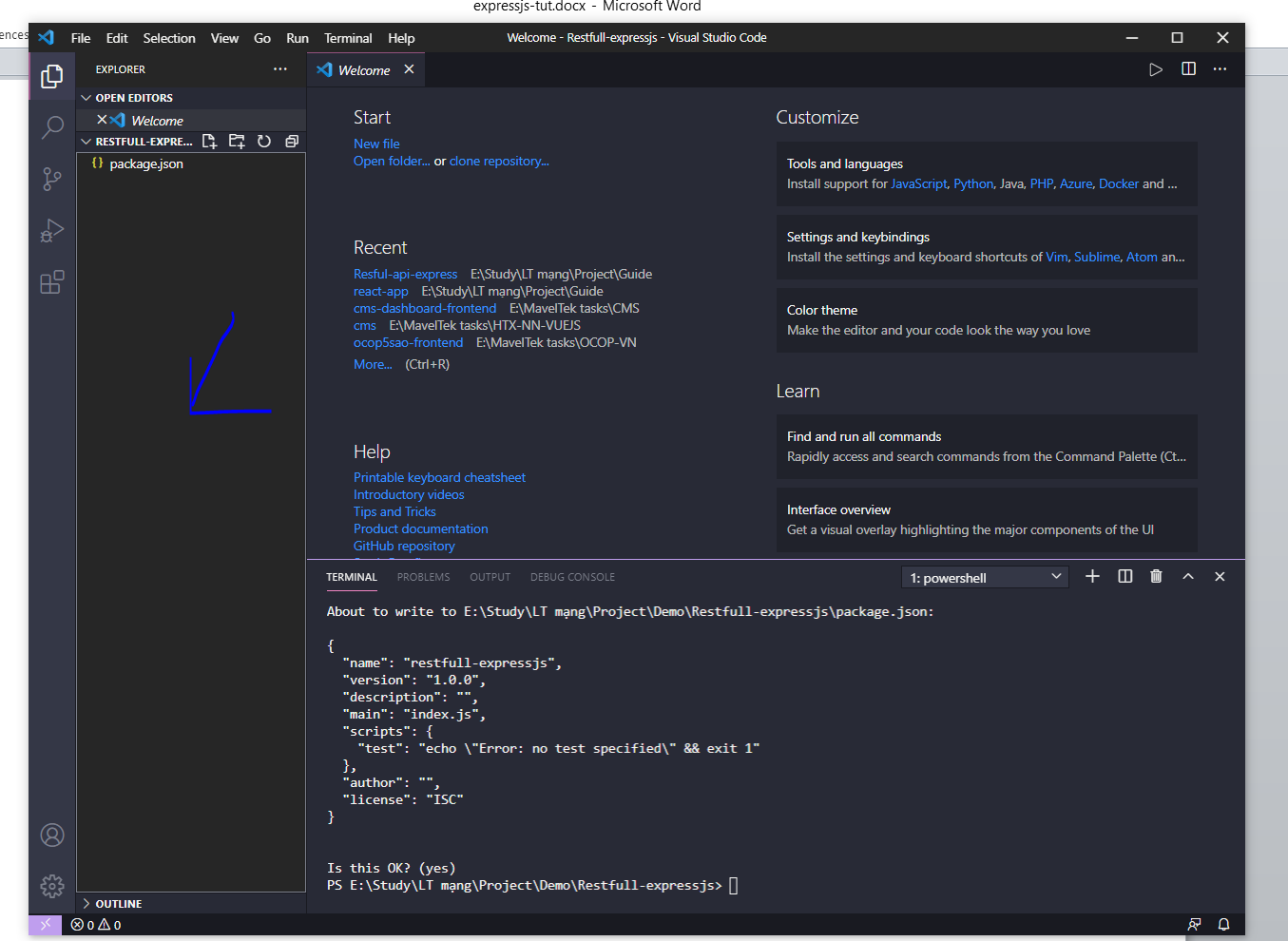
* Sau đó gõ “npm init”



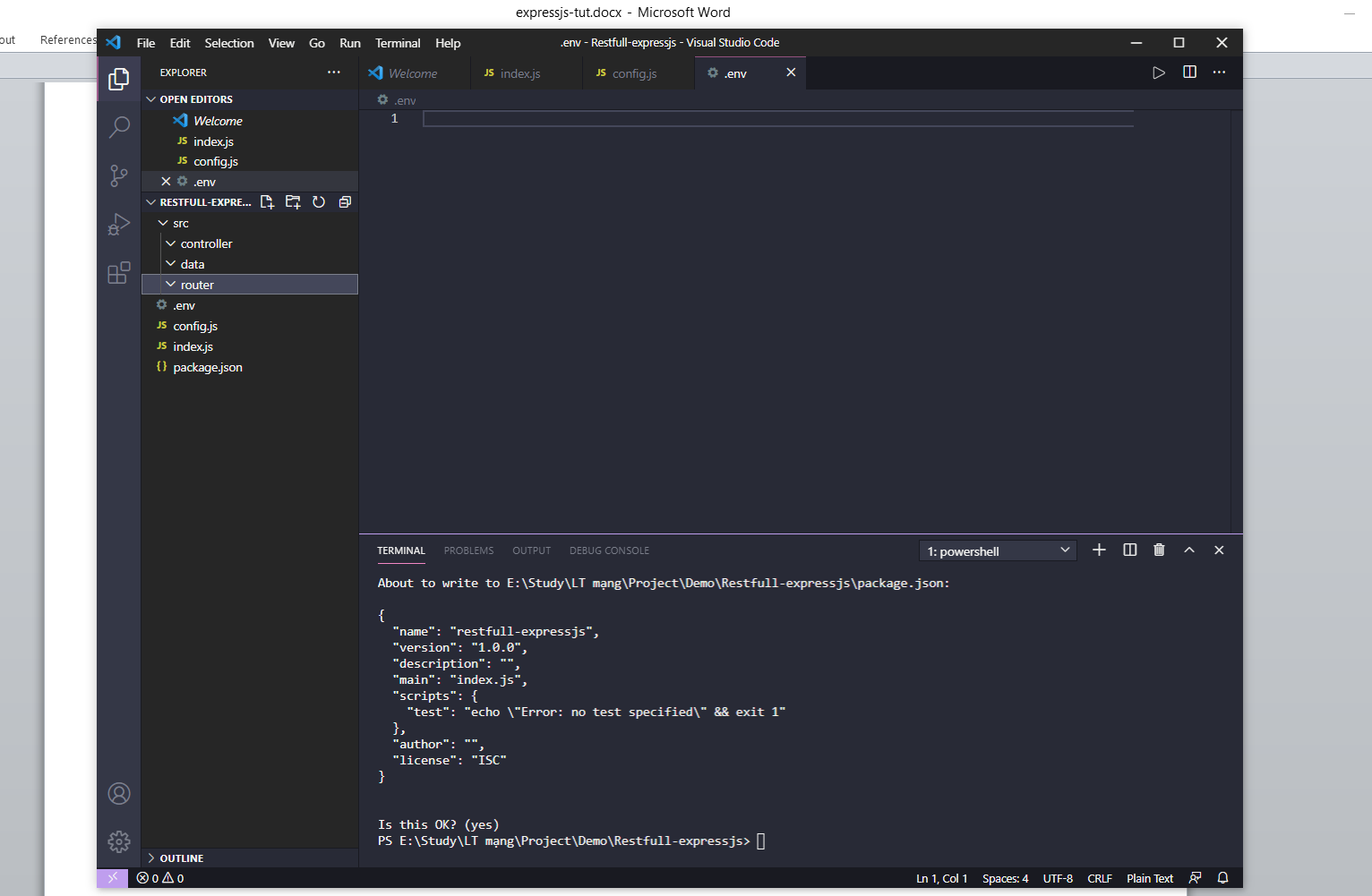
* Sau đó cứ enter cho đến hết



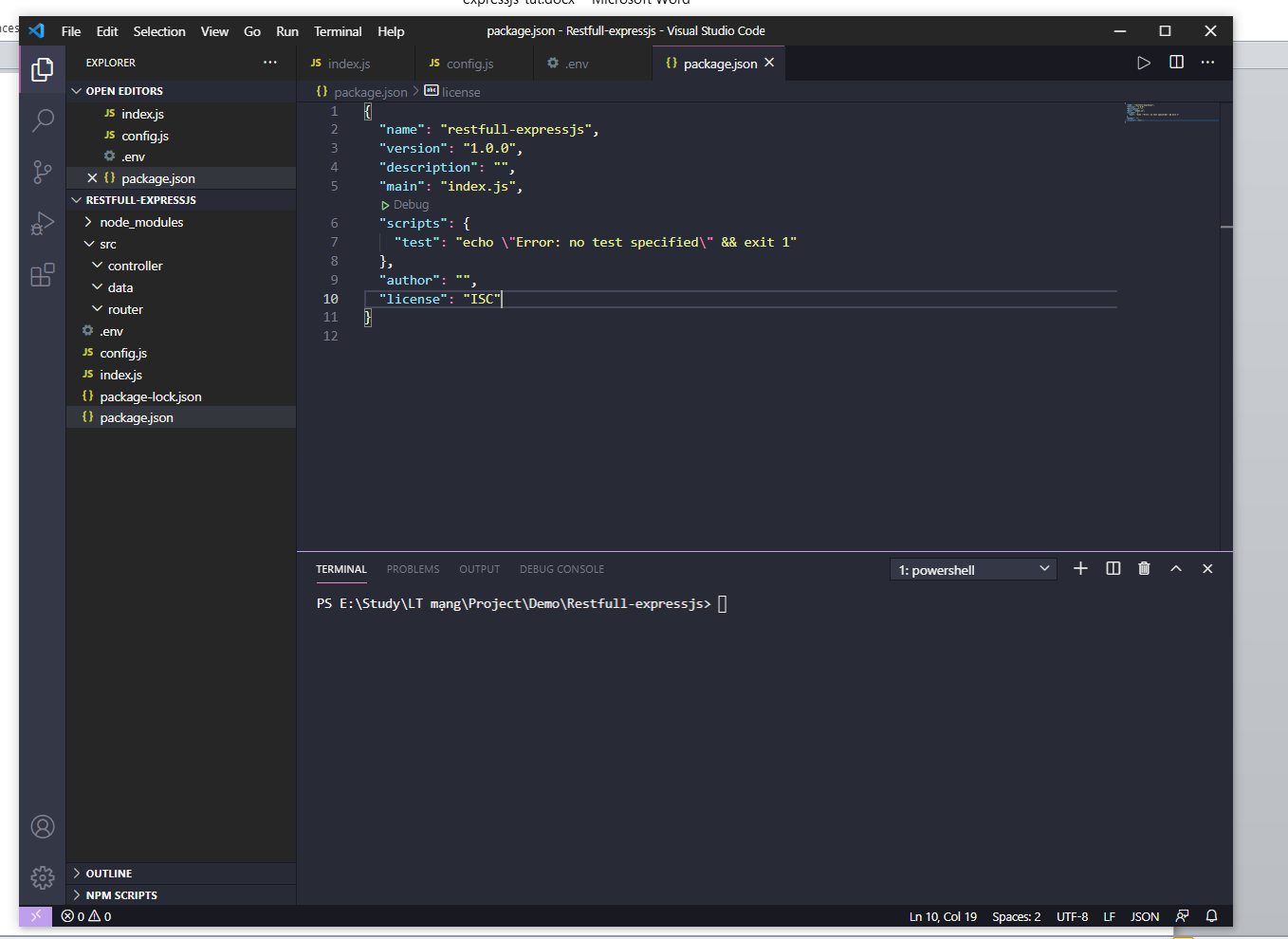
* Như này là oke



Chuột phải vào đây để tạo file và cấu trúc thư mục như sau



* Mở file pakeage.json (Nơi khai bào các thư viện trong project)



Trong “script” thêm đoạn này

    "start": "nodemon index.js"

Và thêm đoạn này vào cuối file

"dependencies": {

    "cors": "^2.8.5",

    "dotenv": "^8.2.0",

    "express": "^4.17.1",

    "mssql": "^6.3.1"

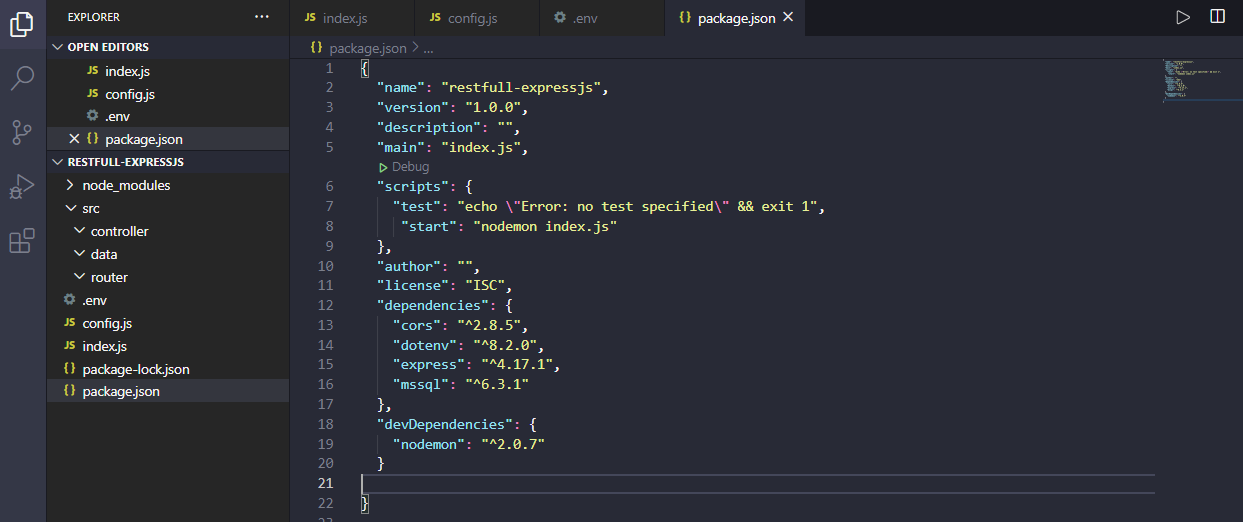
  },

  "devDependencies": {

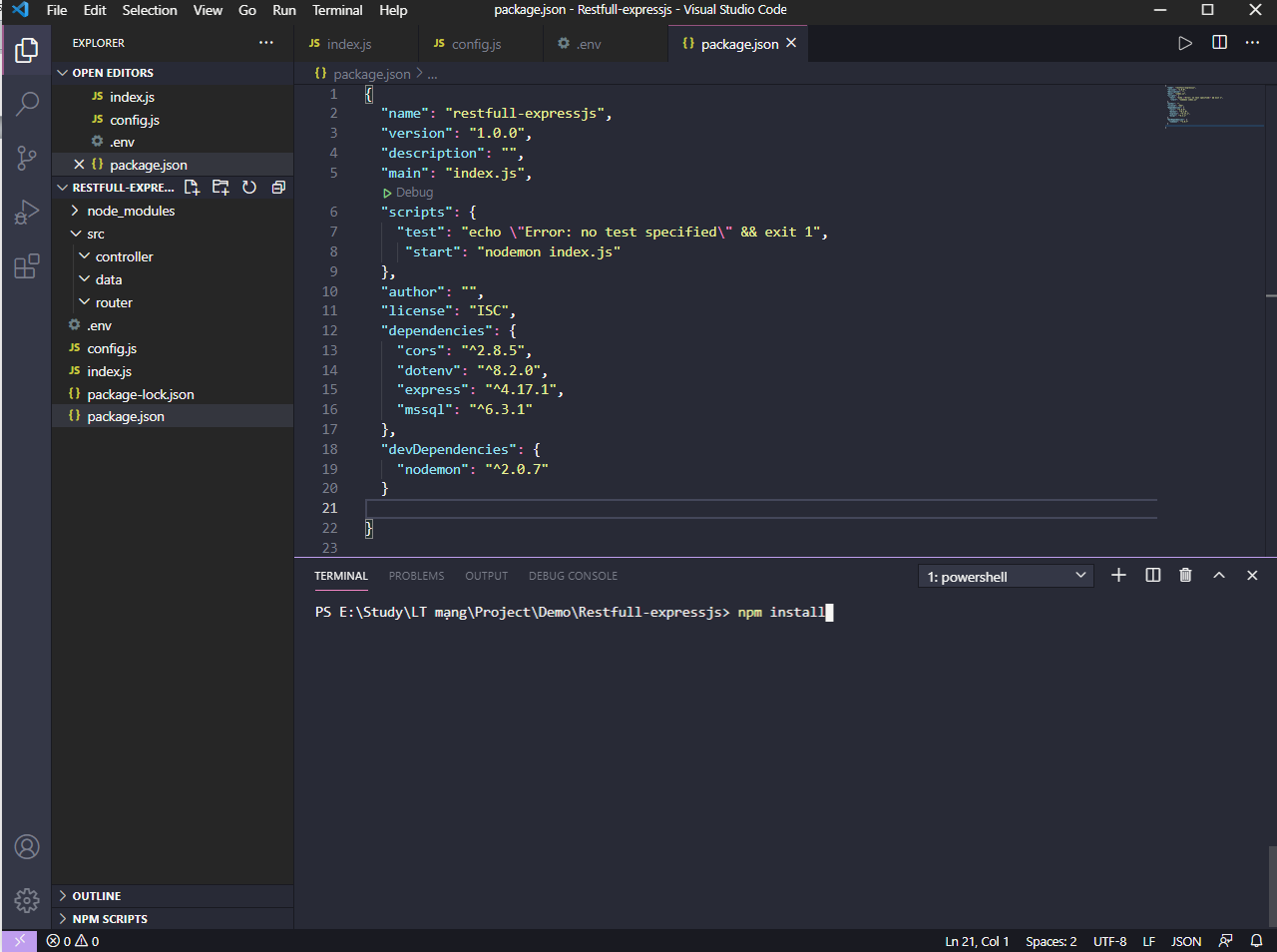
    "nodemon": "^2.0.7"

  }

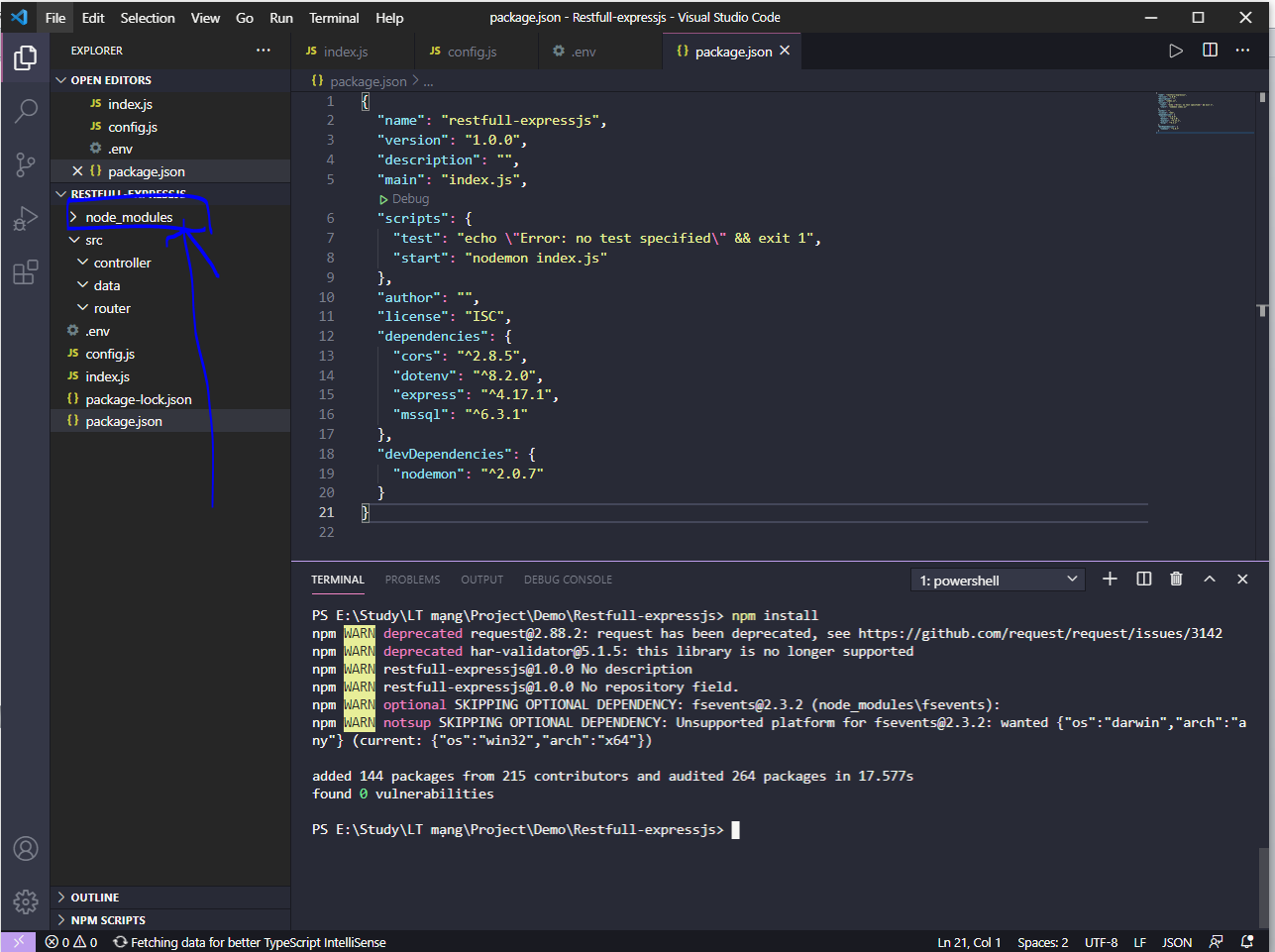
Sau đây là kết quả sau khi thêm (Chú có dấu “,” ngăn cách)



Tại cửa sổ terminal gõ lệnh “npm install” để install toàn bộ thư viện đã thêm



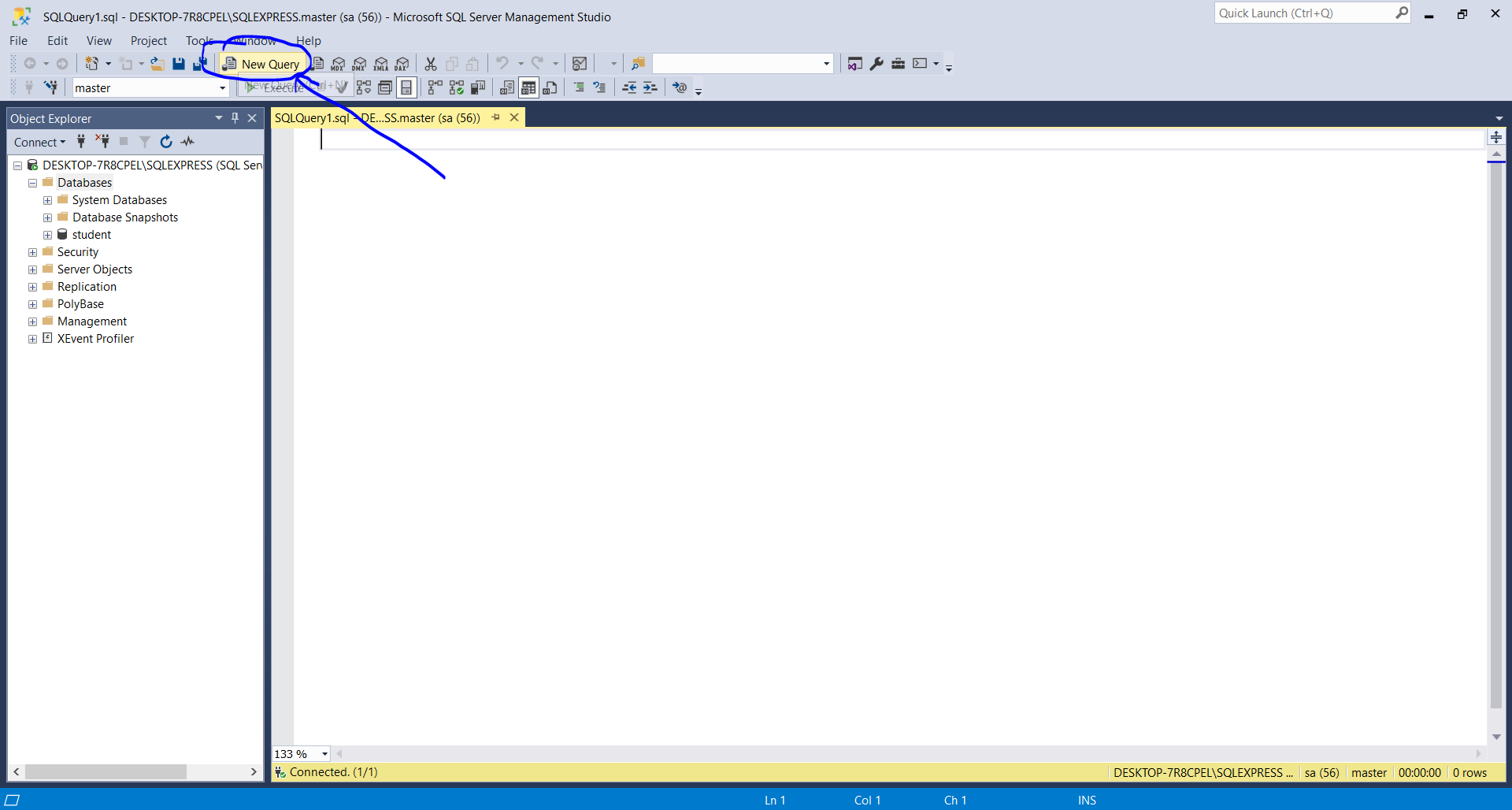
Rồi enter và đợi nó tại các thư viện về



Như này là xong và các thư viện sẽ đc tải về trong thư mục này

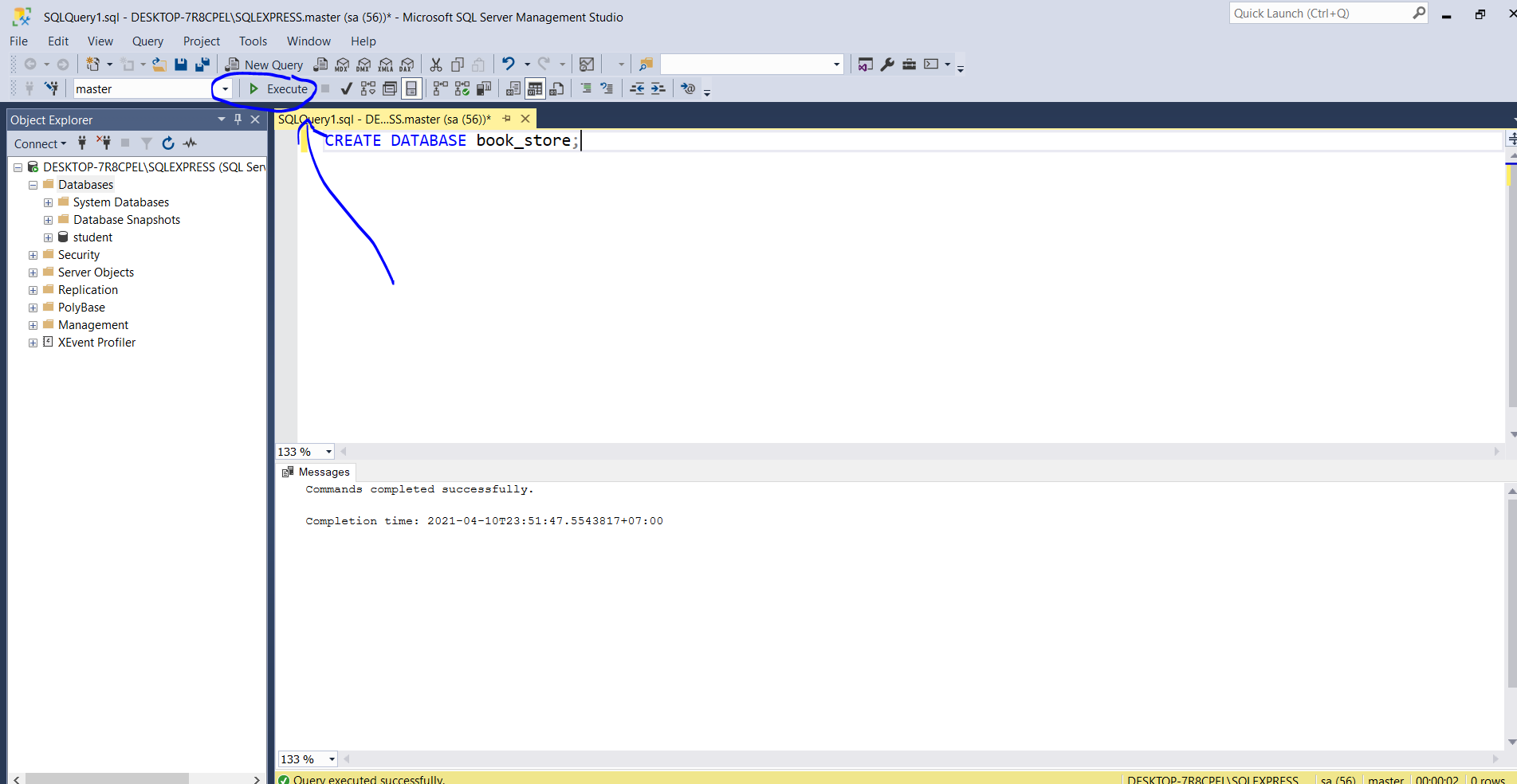
2.Tiếp đến ta sẽ tạo database và bảng trong ms sql server (Nhớ tk và mật khẩu sql server)

Mở sql server, kết nối rồi chọn “new query”

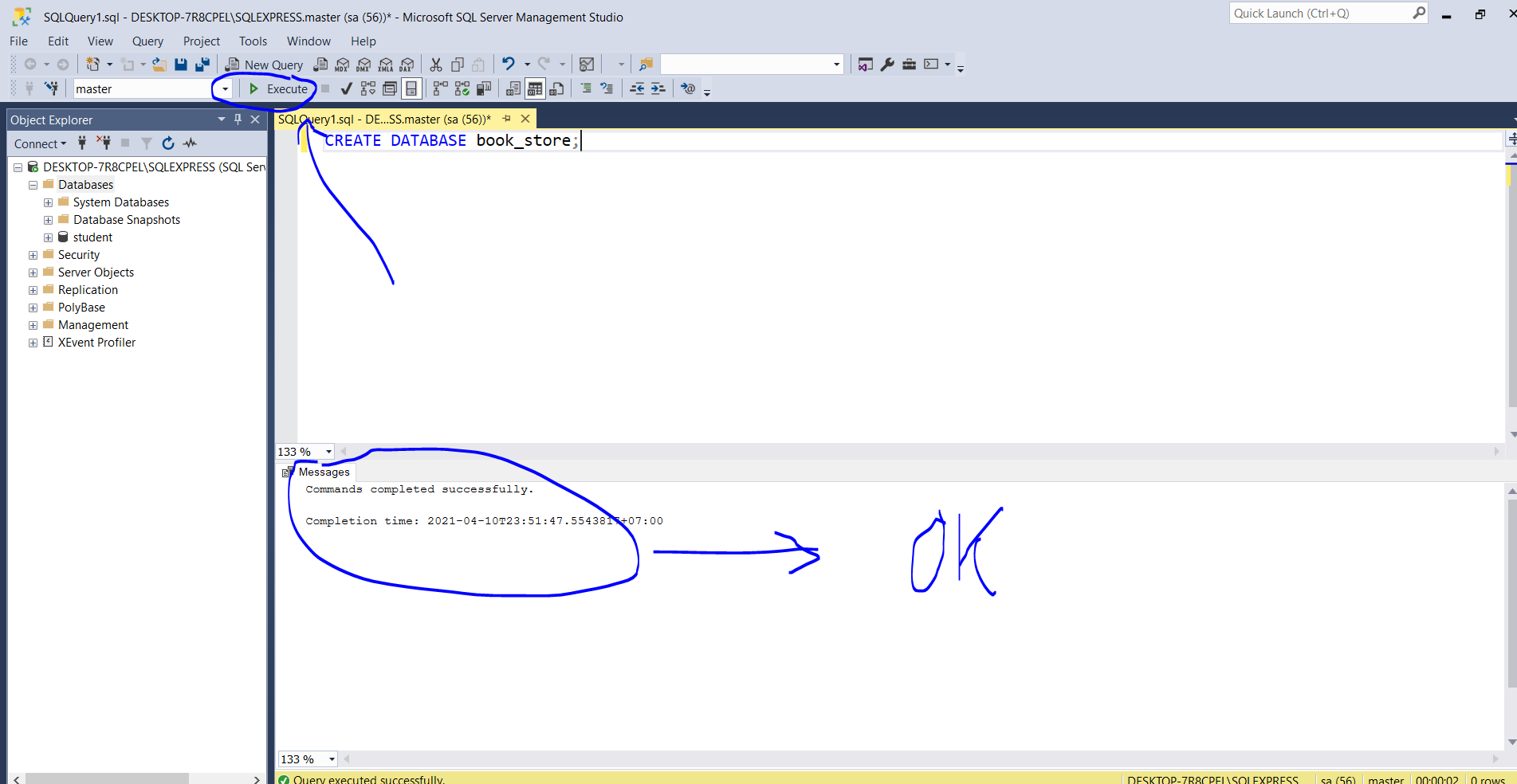


Tạo database bằng lệnh sau:

CREATE DATABASE book\_store;



Chọn execute để chạy lệnh



Tạo bảng bằng lệnh sau:

USE book\_store;

CREATE TABLE book(

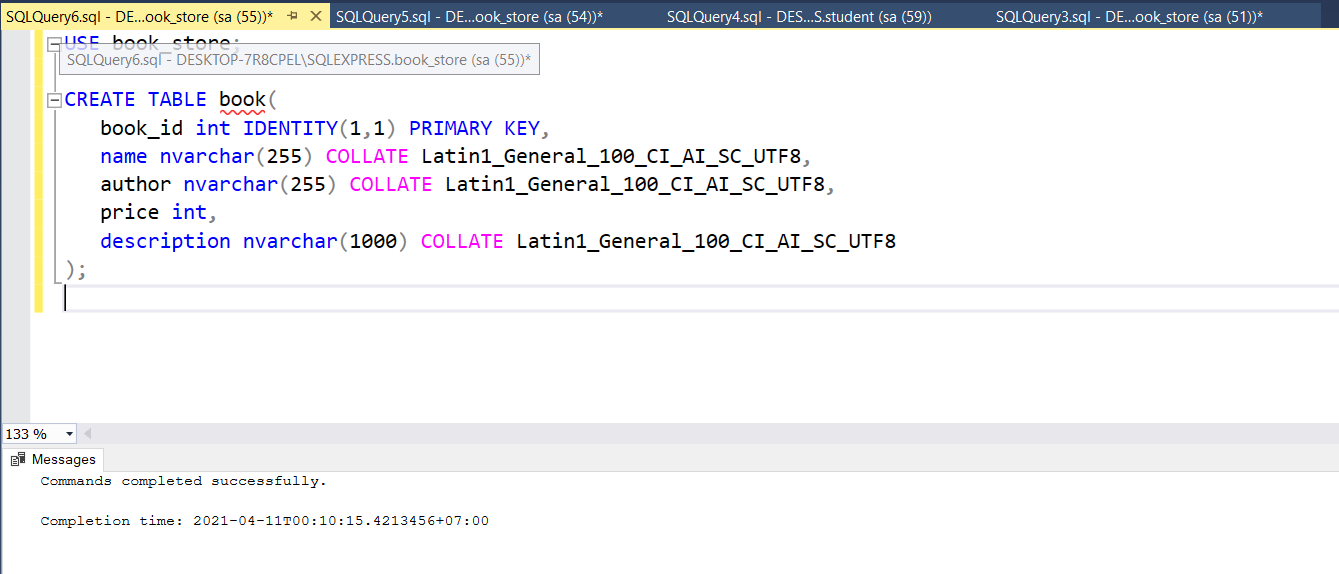
book\_id int IDENTITY(1,1) PRIMARY KEY,

name nvarchar(255) COLLATE Latin1\_General\_100\_CI\_AI\_SC\_UTF8,

author nvarchar(255) COLLATE Latin1\_General\_100\_CI\_AI\_SC\_UTF8,

price int,

description nvarchar(1000) COLLATE Latin1\_General\_100\_CI\_AI\_SC\_UTF8

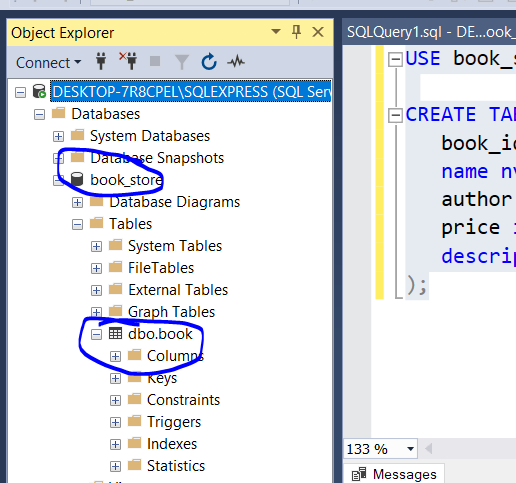
); 

Book\_id là khóa chính tự tăng

255, 1000 là số ký tự

“COLLATE Latin1\_General\_100\_CI\_AI\_SC\_UTF8” để sử dụng được tiếng Việt

Như vậy là oke



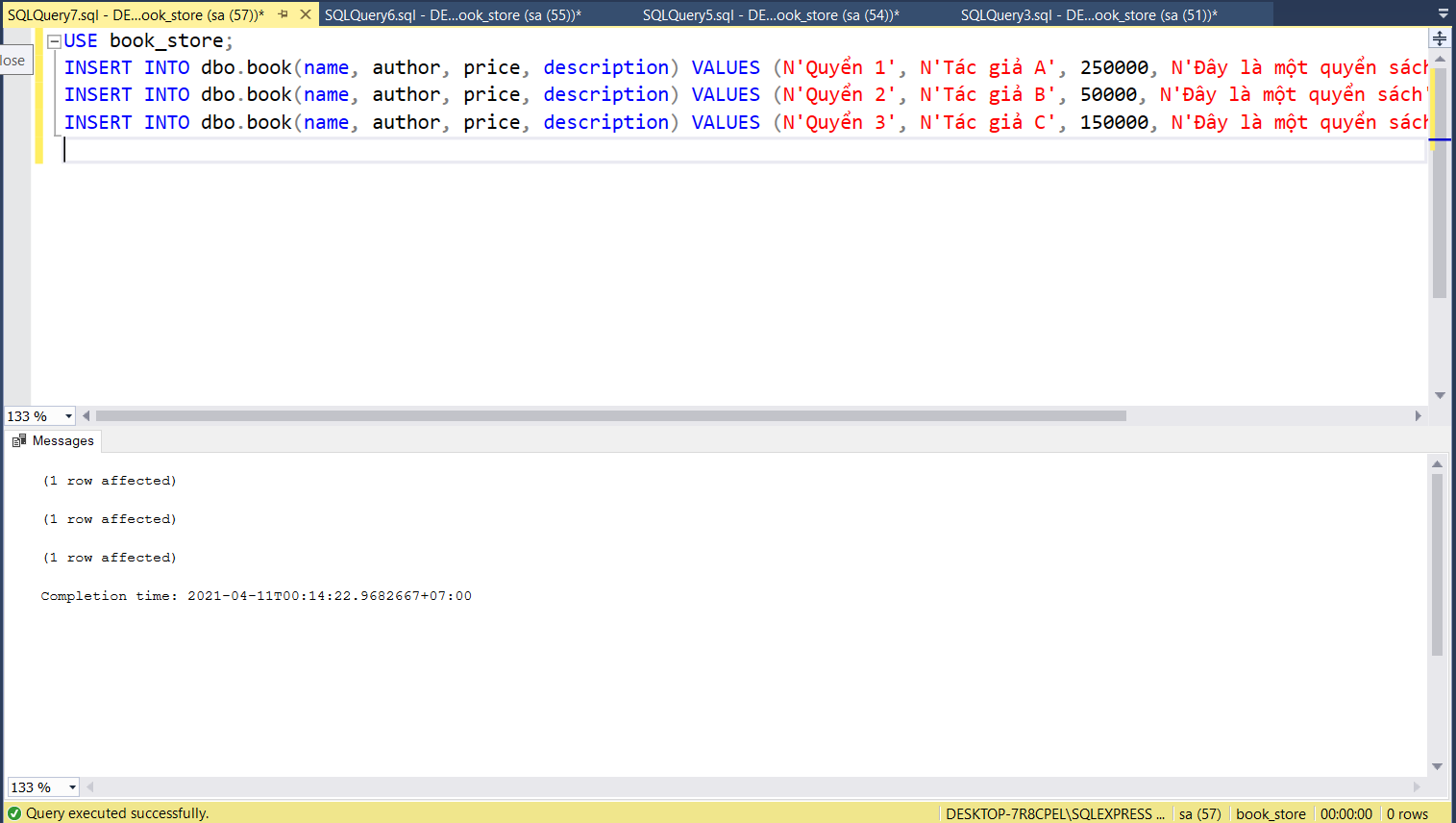
Giờ chúng ta sẽ insert vài quyển vào để có dữ liệu

USE book\_store;

INSERT INTO dbo.book(name, author, price, description) VALUES (N'Quyển 1', N'Tác giả A', 250000, N'Đây là một quyển sách');

INSERT INTO dbo.book(name, author, price, description) VALUES (N'Quyển 2', N'Tác giả B', 50000, N'Đây là một quyển sách');

INSERT INTO dbo.book(name, author, price, description) VALUES (N'Quyển 3', N'Tác giả C', 150000, N'Đây là một quyển sách');

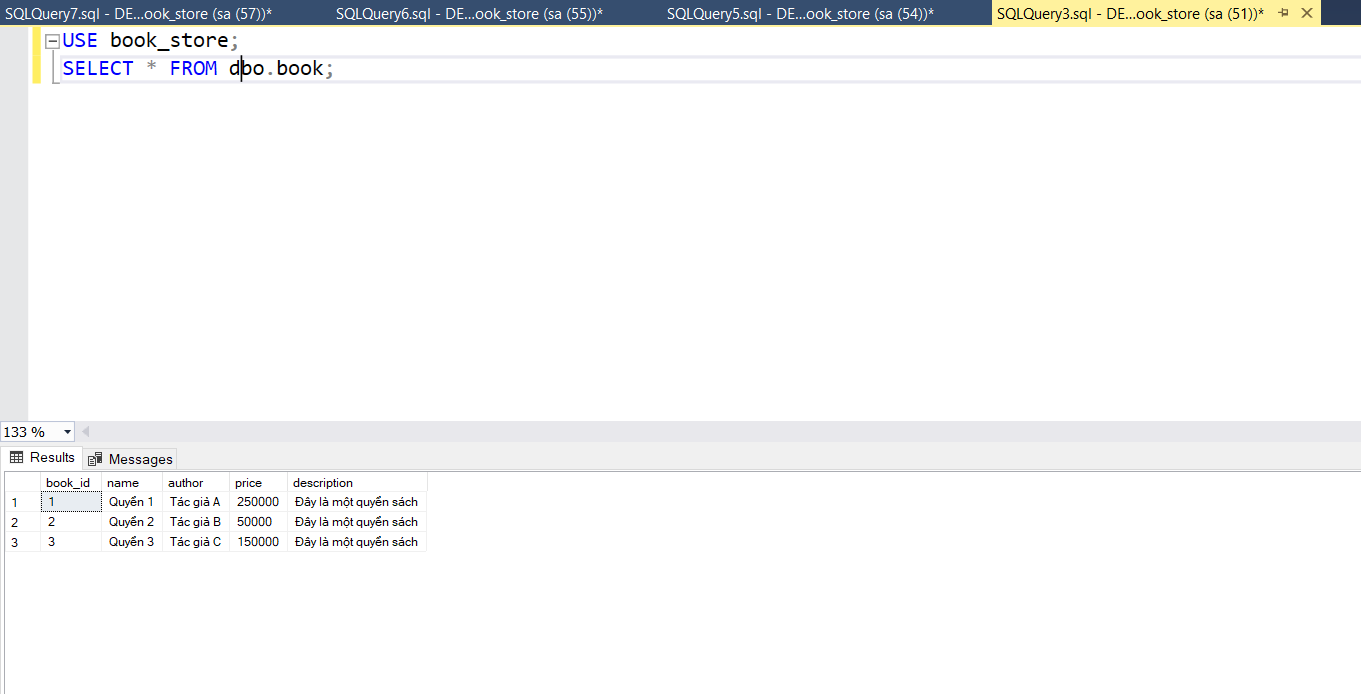


Như vậy là oke

Để xem các bản ghi

USE book\_store;

SELECT \* FROM dbo.book;



Oke vậy là xong phần database

1. Thiết lập server

Mở file .env để thêm các biến môi trường

#express serve config

PORT=8031 // Port mà server chạy

HOST=localhost

HOST\_URL=http://localhost:8031 // đường dẫn

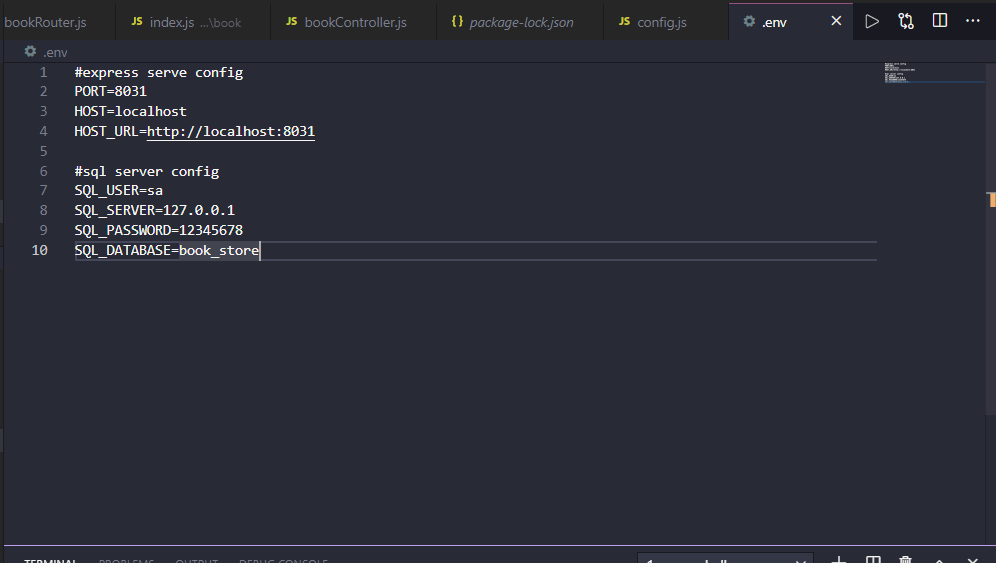
#sql server config

SQL\_USER=sa // Tên đăng nhập vào ms sql server

SQL\_SERVER=127.0.0.1

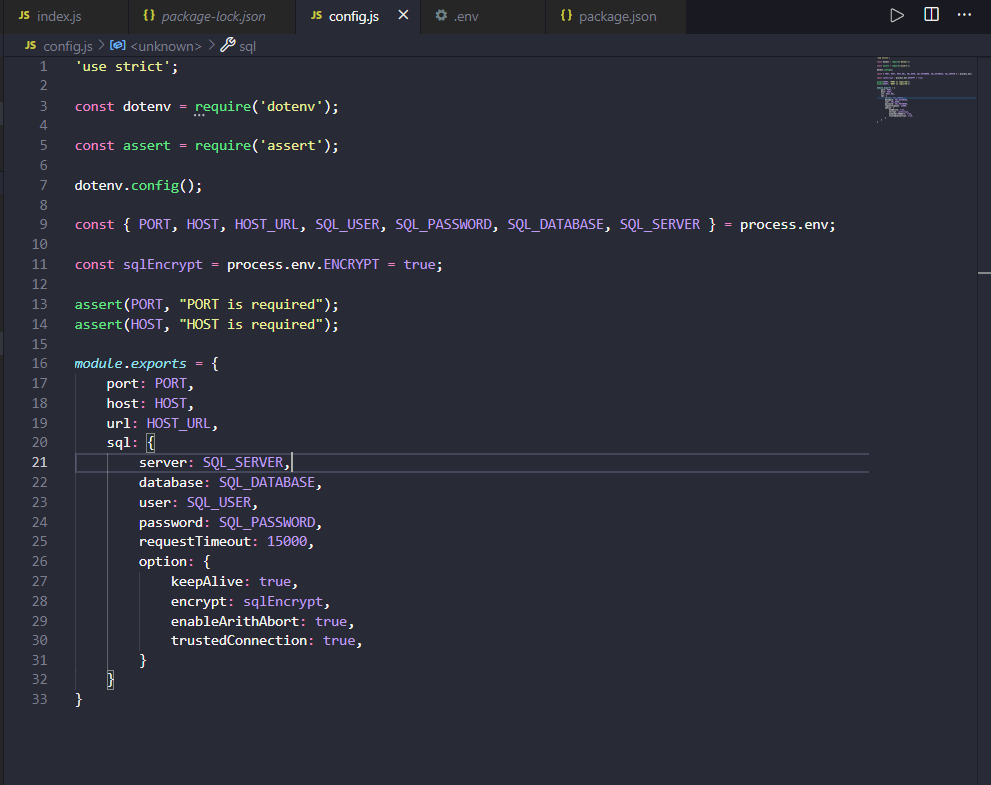
SQL\_PASSWORD=12345678 // Mật khẩu đăng nhập vào

SQL\_DATABASE=book\_store // Tên database

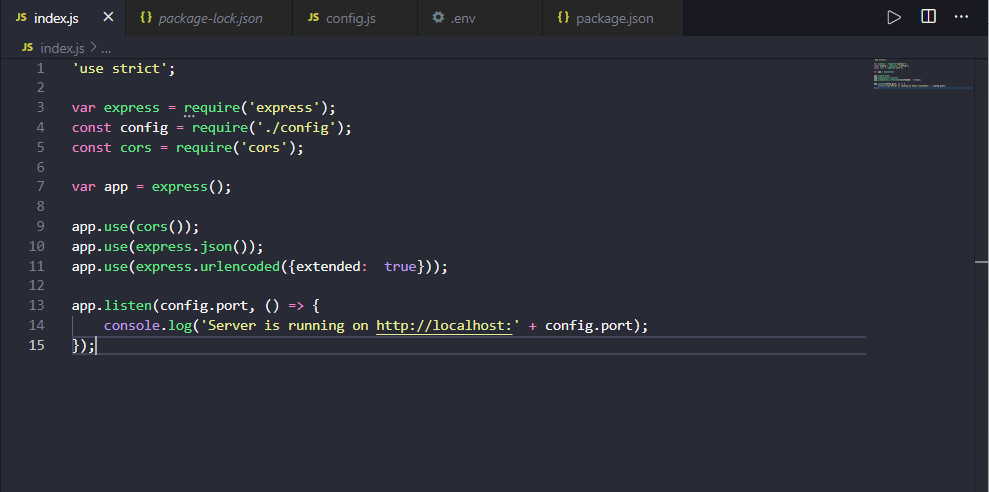


* Thiết lập config trong file config.js
* 'use strict';
* const dotenv = require('dotenv');
* const assert = require('assert');
* dotenv.config();
* const { PORT, HOST, HOST\_URL, SQL\_USER, SQL\_PASSWORD, SQL\_DATABASE, SQL\_SERVER } = process.env;
* const sqlEncrypt = process.env.ENCRYPT = true;
* assert(PORT, "PORT is required");
* assert(HOST, "HOST is required");
* *module*.*exports* = {
* port: PORT,
* host: HOST,
* url: HOST\_URL,
* sql: {
* server: SQL\_SERVER,
* database: SQL\_DATABASE,
* user: SQL\_USER,
* password: SQL\_PASSWORD,
* requestTimeout: 15000,
* option: {
* keepAlive: true,
* encrypt: sqlEncrypt,
* enableArithAbort: true,
* trustedConnection: true,
* }
* }
* }

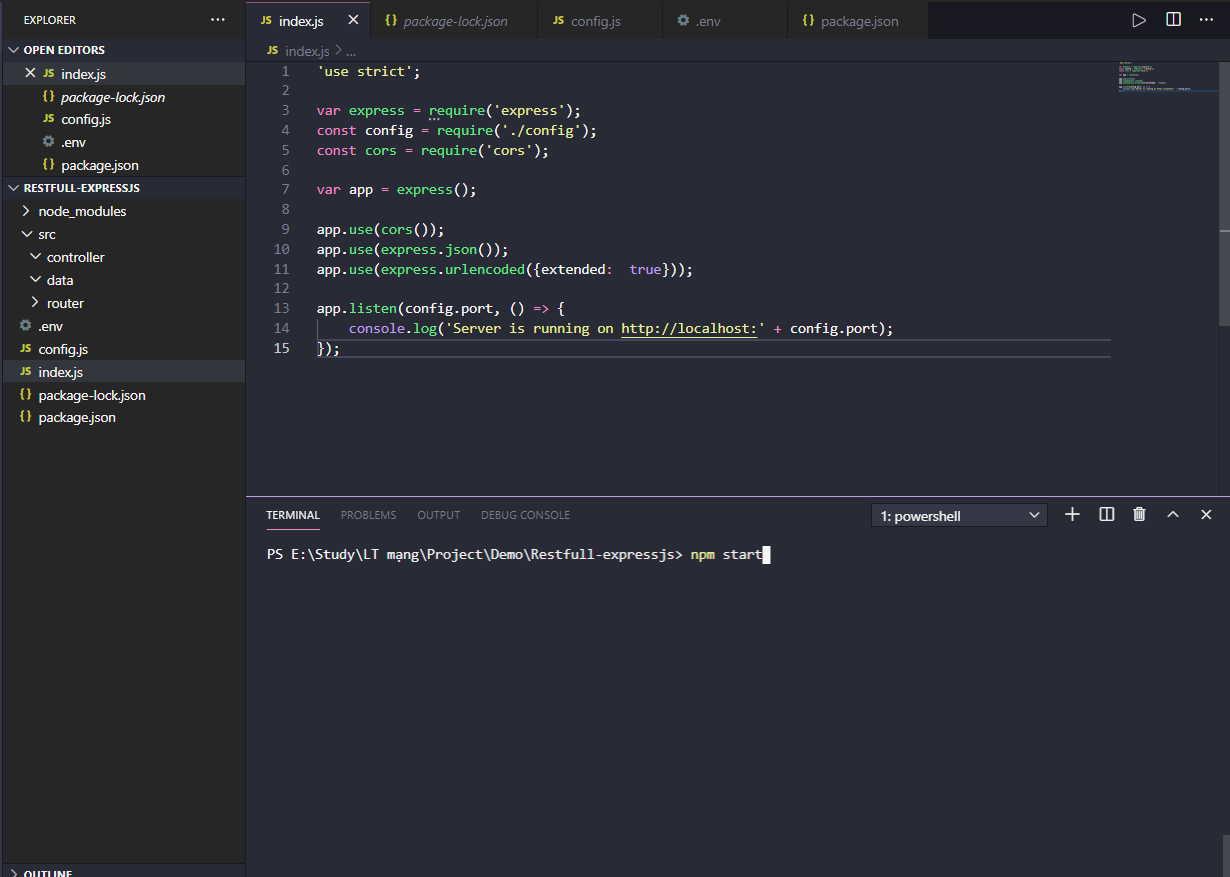
Tạm thời chưa cần hiểu chỉ cần copy paste thôi



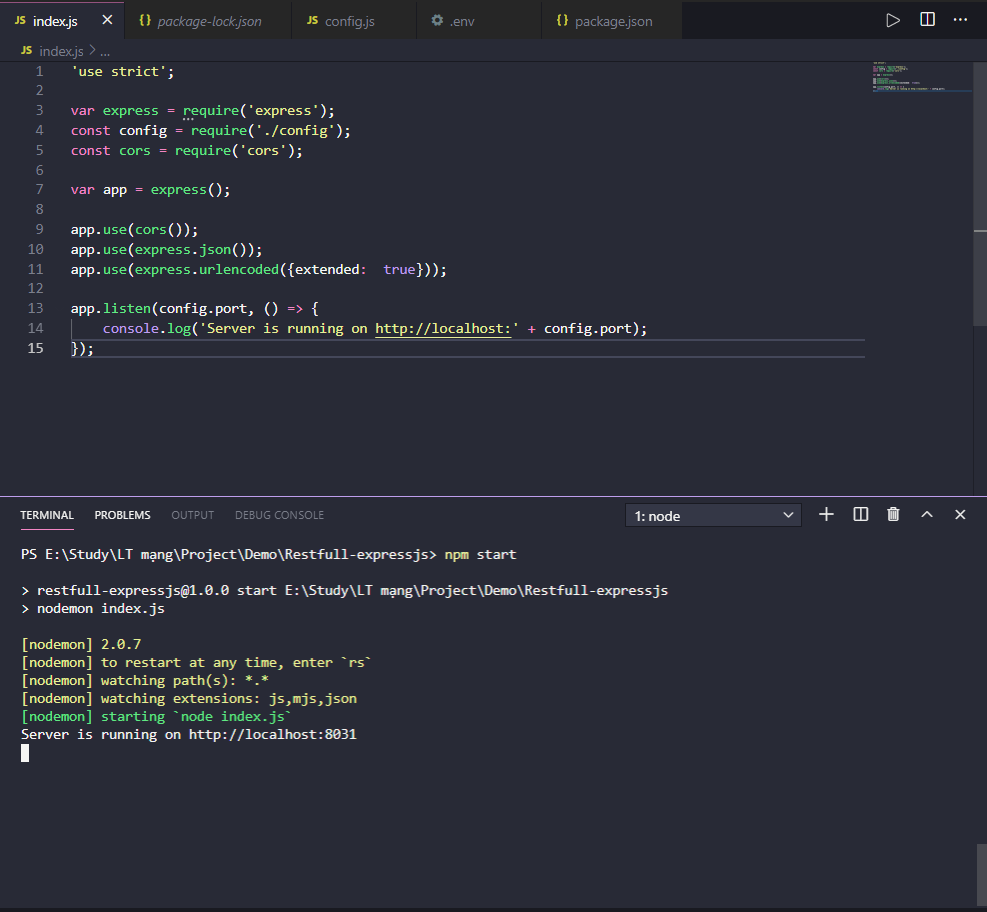
* Thiết lập để chạy server trong file index.js
* 'use strict';
* var express = require('express');
* const config = require('./config'); // để lấy config từ file config
* const cors = require('cors');
* var app = express();
* app.use(cors());
* app.use(express.json()); // để sử dụng json
* app.use(express.urlencoded({extended:  true}));
* app.listen(config.port, () => { //Lắng nghe và chạy server
* //Nếu thành công sẽ in ra dùng ntn trong console.log
* console.log('Server is running on http://localhost:' + config.port);
* });



Oke chạy server thôi



Dùng lệnh npm start để chạy server



Đây là kết quả (Như vậy là server chạy thành công trên <http://localhost:8031>)

Server sẽ tự reset khi có thay đổi code

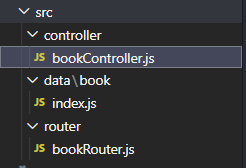
1. Tiến hành code service đâu tiên (Lấy toàn bộ dữ liệu sách trong table book)

* Tạo các file như sau:

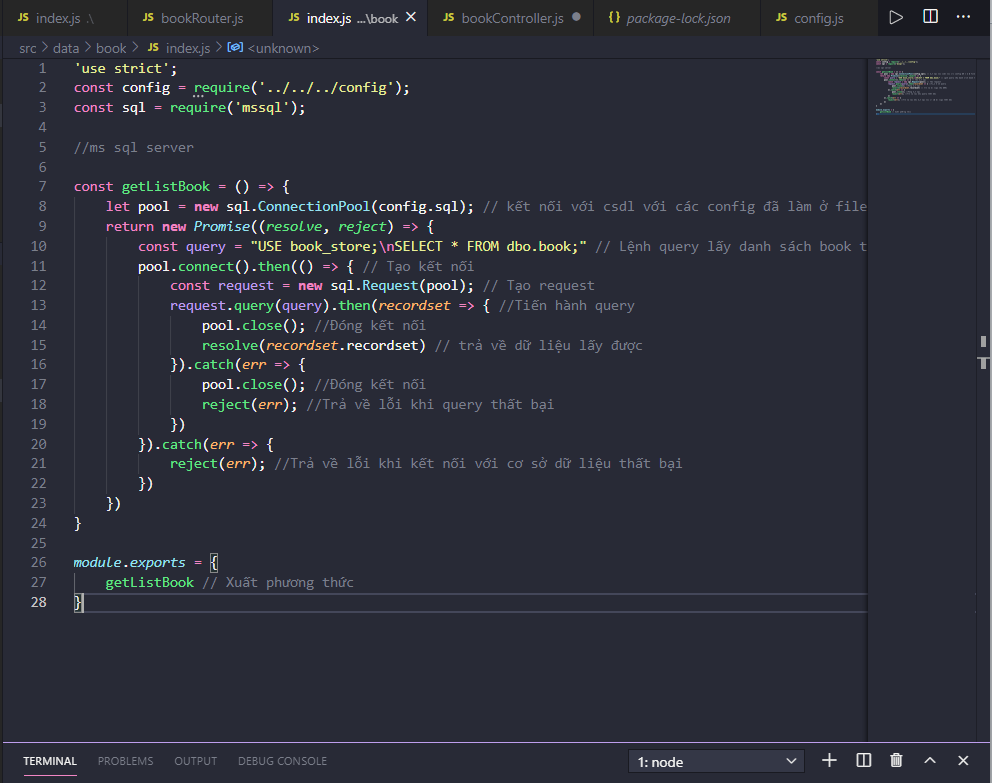
+ Trong thư mục controller tạo file bookController.js (Để xử lý các request GET, POST, PUT, DELETE…)

+ Trong thư mục data tạo thư mục book để quản lý dữ liệu bảng book (kết nối csdl và query). Tiếp đó tạo file index.js để thực hiện

+ Trong thư mục router tạo file bookRouter.js (Để tạo các đường dẫn và định nghĩa các phương thức cho các đường dẫn vs các phương thức ở controller)



* Ở file index.js
* 'use strict';
* const config = require('../../../config');
* const sql = require('mssql');
* //ms sql server
* const getListBook = () => {
* let pool = **new** sql.ConnectionPool(config.sql); // kết nối với csdl với các config đã làm ở file config
* return **new** *Promise*((*resolve*, *reject*) => {
* const query = "USE book\_store;\nSELECT \* FROM dbo.book;" // Lệnh query lấy danh sách book trong csdl
* pool.connect().then(() => { // Tạo kết nối
* const request = **new** sql.Request(pool); // Tạo request
* request.query(query).then(*recordset* => { //Tiến hành query
* pool.close(); //Đóng kết nối
* resolve(*recordset*.recordset) // trả về dữ liệu lấy được
* }).catch(*err* => {
* pool.close(); //Đóng kết nối
* reject(*err*); //Trả về lỗi khi query thất bại
* })
* }).catch(*err* => {
* reject(*err*); //Trả về lỗi khi kết nối với cơ sở dữ liệu thất bại
* })
* })
* }
* *module*.*exports* = {
* getListBook // Xuất phương thức
* }



Giờ đến controller

'use strict';

const bookData = require('../data/book'); // import module kết nối với cơ sở dữ liệu

const getListBook = async (*req*, *res*, *next*) => {

    try{

        const books  = await bookData.getListBook();

*res*.send(books); //Thành công trả về list book

    }catch(err){

*res*.status(400).send(err.message); // trả về lỗi thì lấy dữ liệu thất bại

    }

}

*module*.*exports* = {

    getListBook // Xuất phương thức

}



Đến router

'use strict';

const express = require('express');

const router = express.Router();

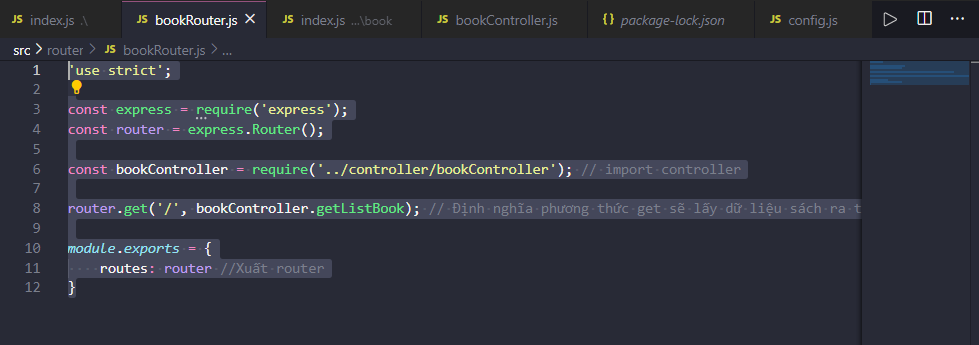
const bookController = require('../controller/bookController'); // import controller

router.get('/', bookController.getListBook); // Định nghĩa phương thức get sẽ lấy dữ liệu sách ra tại "/"

*module*.*exports* = {

    routes: router //Xuất router

}



Cuối cùng là file index.js ở ngoài cùng

Import router

const bookRouter = require('./src/router/bookRouter')

Sử dụng router

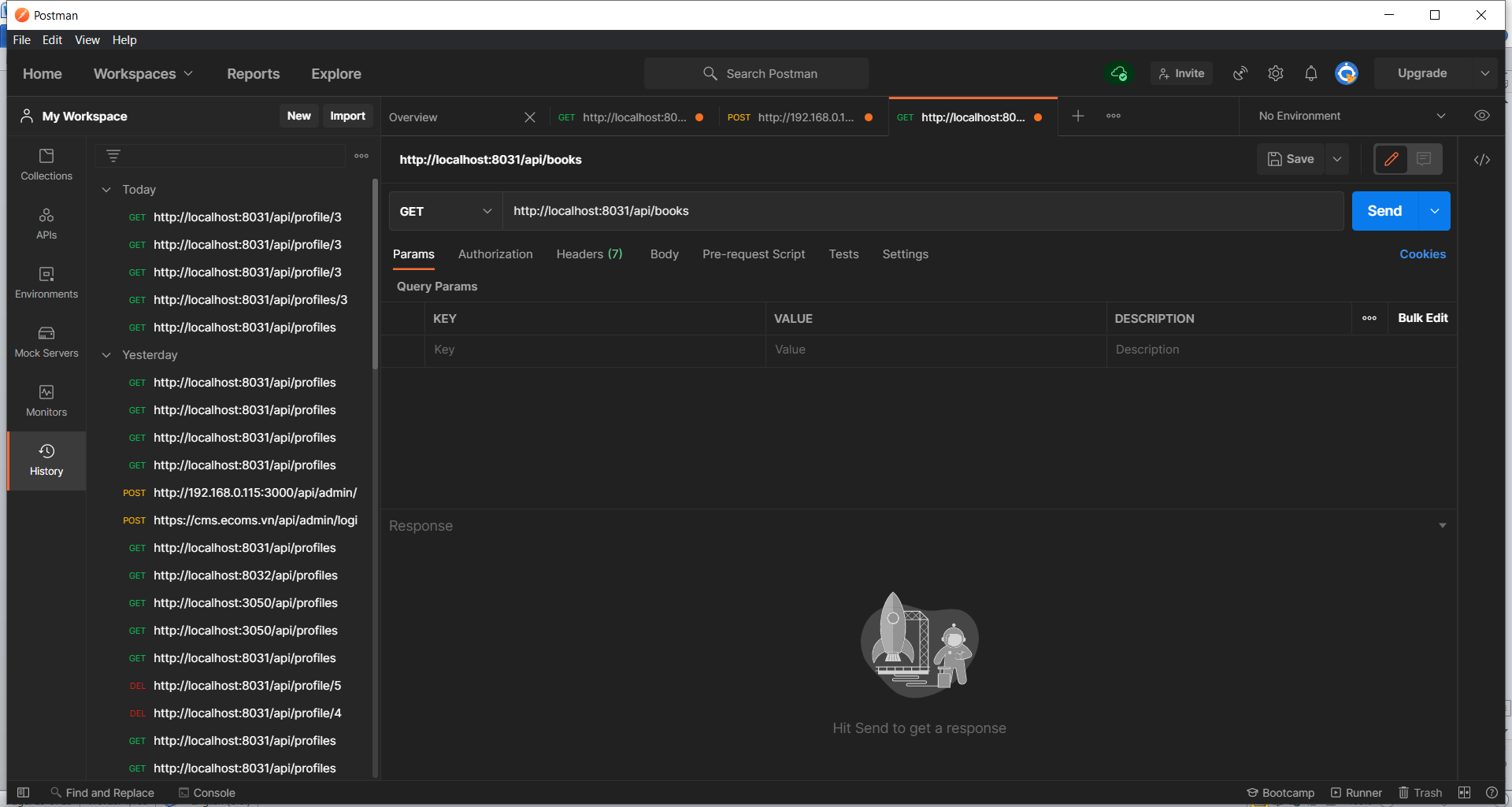
app.use('/api/books', bookRouter.routes); // http://localhost:8031/api/books url để tạo request



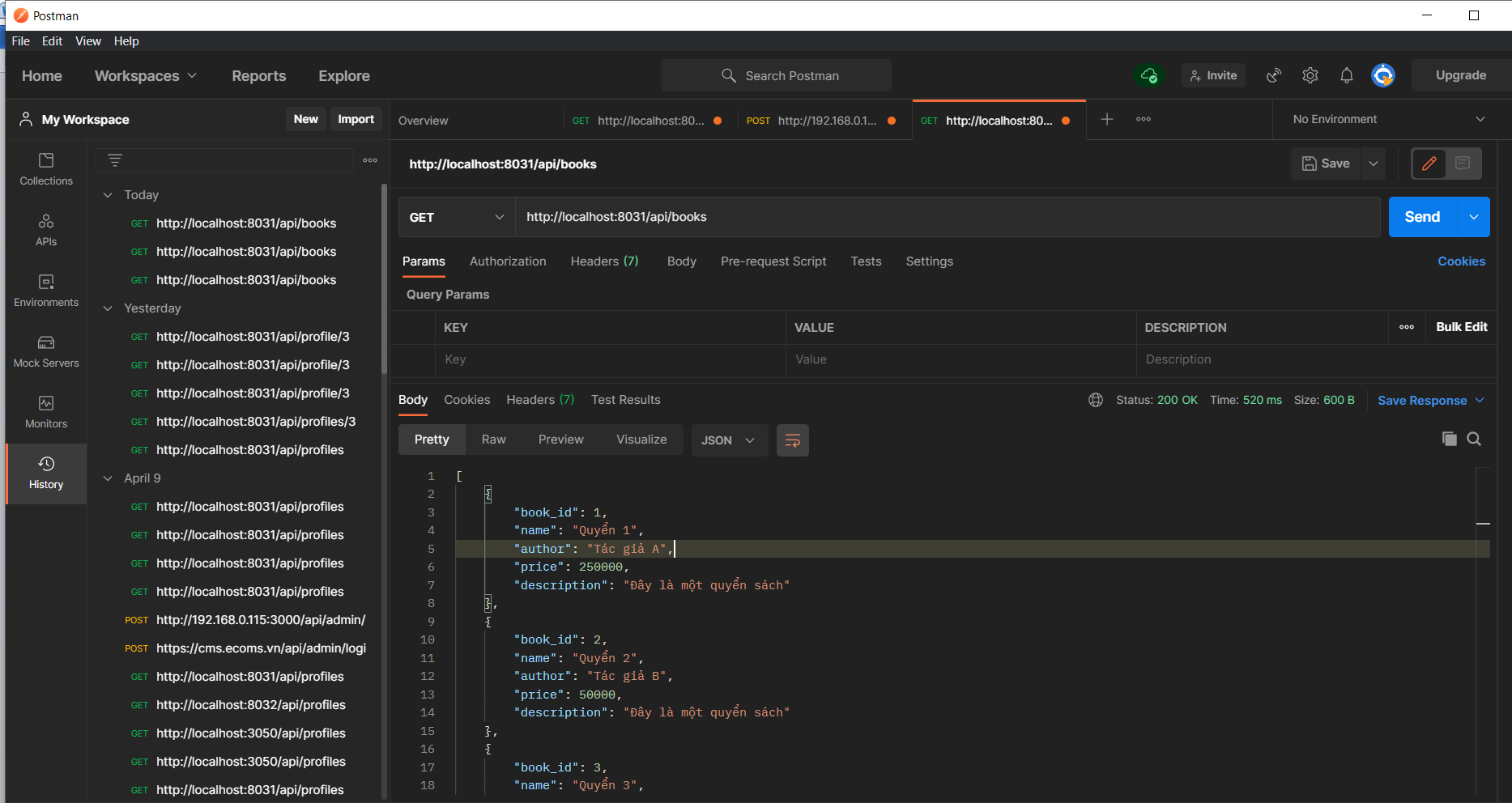
Oke giờ test service đầu tiên thôi

Mở Postman

Dán url: http://localhost:8031/api/books



Sau đó bấm send và đây là kết quả



getById, thêm mới, sửa, xóa thì có trong source code sau:

<https://github.com/RainStrong532/book-resfull-api-expressjs>

Test search api

