Nodemon

Nodemon 是一個非常方便的工具，用於開發 Node.js 應用程式。 它的作用是監視你的專案文件的變化，並在文件發生改變時自動重啟你的 Node 應用程序，這樣你就不必手動停止和重新啟動伺服器。 這在開發過程中可以大大提高效率。

使用 nodemon，你可以專注於程式碼的編寫和調試，而不必在每次更改後都手動重新啟動伺服器。 它極大地簡化了開發流程，特別是在開發大型專案時，頻繁修改程式碼並查看更改的效果是非常常見的。

要開始使用 nodemon，首先確保你已經全域安裝了它（使用 npm install -g nodemon 命令）。 然後，在你的專案目錄下，只需執行 nodemon 命令，它會開始監視你的文件，並在文件變更時自動重新啟動伺服器。

除了基本的檔案監視和自動重新啟動功能外，nodemon 還提供了許多設定選項，可以根據您的需求進行定制，例如指定要監視的檔案類型、忽略特定檔案或目錄等等。

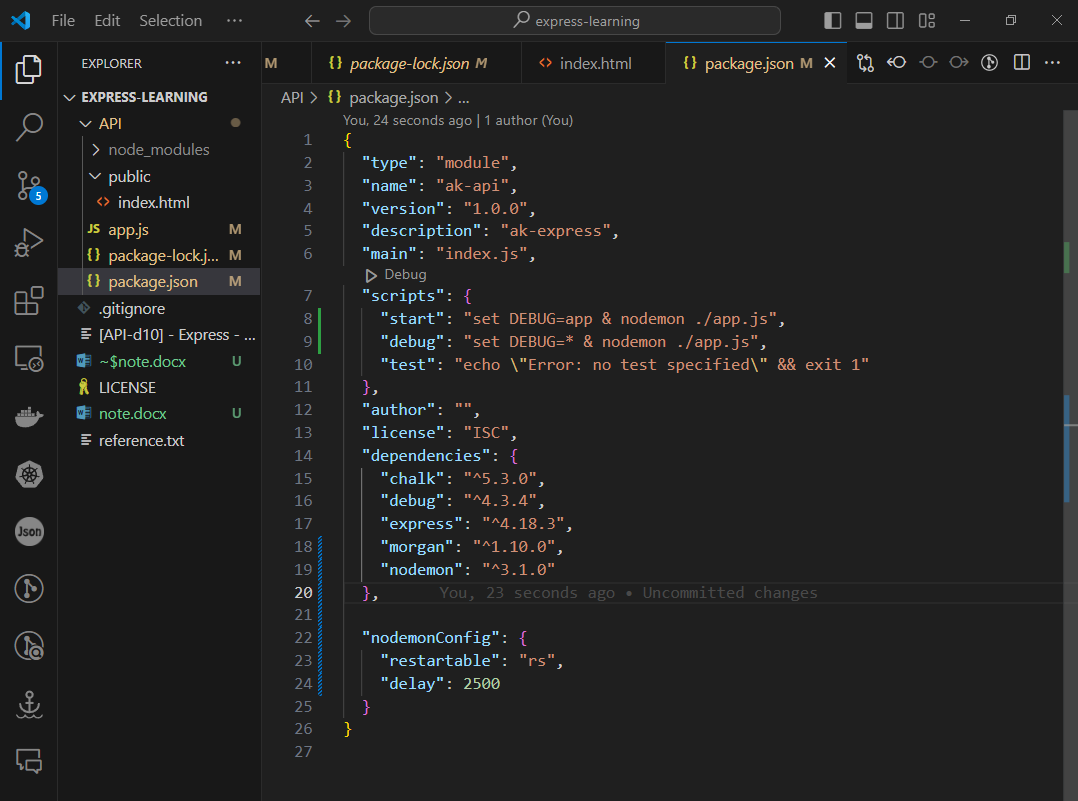
總的來說，nodemon 是 Node.js 開發中一個非常實用的工具，它可以幫助你更輕鬆地進行程式碼開發和調試，節省了大量寶貴的時間。



如何使用:

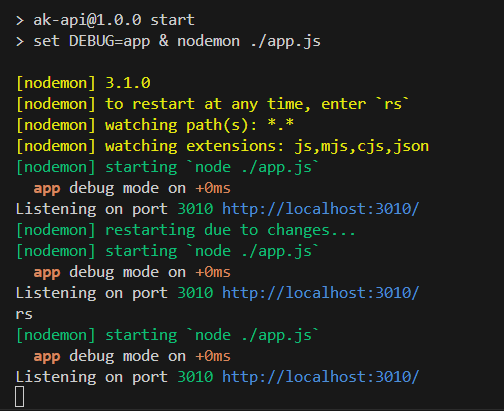
先安裝

$ npm install nodemon

改啟動命令 從node改成nodemon 和新增 nodemon的設定  


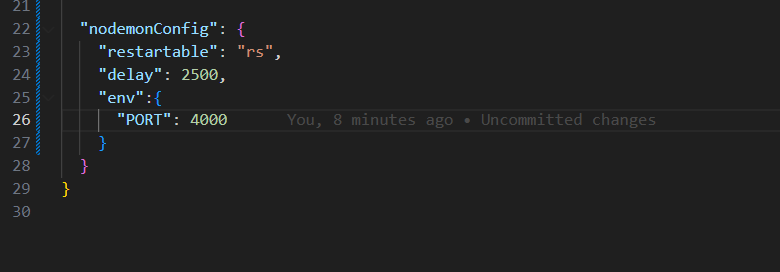
跑node run start 就可以看到使用nodemon來啟動了，他會偵測如果有JS 或 JSON file更新了 就會reload

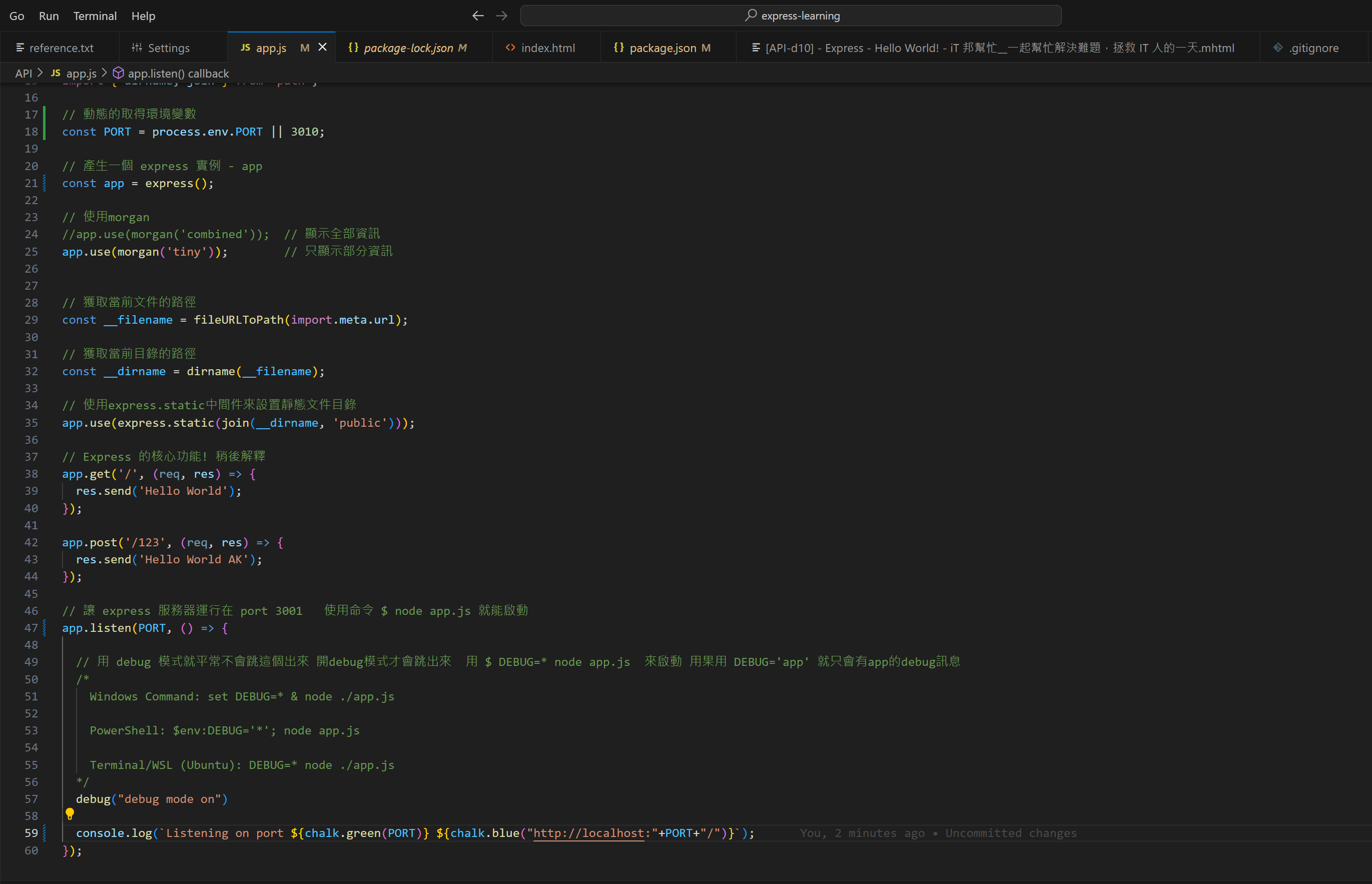
在command line上打 rs 也會restart(我們設定的命令)



再來是要設置環境變數:

先設置環境變數:



再程式中取得環境變數:  


EJS:

先安裝EJS

$ npm install ejs

EJS，全名为Embedded JavaScript templating，是一个流行的模板引擎，用于从数据中生成HTML标记或其他文本。EJS使用简单的JavaScript代码嵌入在模板文件中，这使得它可以很容易地将数据渲染成HTML页面，非常适用于动态内容的网站开发。它既可以在客户端也可以在服务器端使用，但更常见于服务器端，如在Node.js环境中。

EJS的核心优势在于它的简单性和灵活性。使用它，开发人员可以写标准的HTML，并在其中插入简单的JavaScript代码来处理数据。这种方式让页面的动态内容生成变得直接而且高效，不需要复杂的逻辑或结构。

基本语法

EJS模板的基本语法包括几个核心部分：

<% %>：用于执行其中的JavaScript代码，但不输出到HTML中。

<%= %>：会对内部的JavaScript表达式进行计算，并将结果输出到HTML中，结果会被自动转义以防止跨站脚本攻击（XSS）。

<%- %>：和<%= %>相似，但不会对输出进行自动转义。这在输出HTML标记时很有用。

示例

一个简单的EJS模板示例可能看起来像这样：

html

Copy code

<!DOCTYPE html>

<html>

<head>

<title><%= pageTitle %></title>

</head>

<body>

<h1>Welcome, <%= user.name %>!</h1>

<p>We have <%= messages.length %> messages for you.</p>

</body>

</html>

这个模板中，pageTitle、user.name和messages.length都是动态插入到HTML中的数据。使用EJS，你只需要提供这些数据，模板引擎就会渲染出完整的HTML页面。

使用EJS

要开始使用EJS，你可以通过NPM安装它作为一个依赖：

Copy code

npm install ejs

然后在你的Node.js应用中引入并使用它：

javascript

Copy code

const ejs = require('ejs');

const express = require('express');

const app = express();

app.set('view engine', 'ejs');

app.get('/', (req, res) => {

res.render('index', {

pageTitle: 'My EJS Page',

user: { name: 'John Doe' },

messages: ['Hello', 'Hi', 'Welcome']

});

});

app.listen(3000, () => console.log('Server running on port 3000'));

这段代码设置了Express应用使用EJS作为视图引擎，并定义了一个路由来渲染一个名为index的EJS模板，同时传递了一些数据给模板。

EJS是一个强大而灵活的模板引擎，可以使得基于数据的HTML生成变得非常简单。它的语法易于理解，使得开发人员可以快速上手，并在他们的Web应用中高效地使用它。