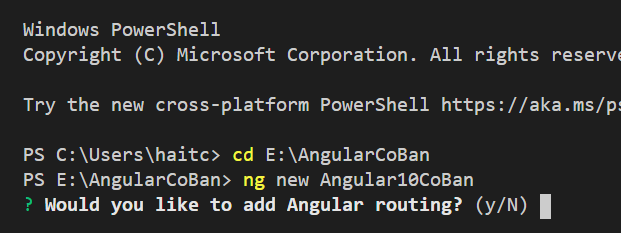
# Bài 2 Tạo project Angular 10 đầu tiên

## 1. Tạo project

Để tạo câu project Angular ta chạy câu lệnh **ng new <tên project>** trong cửa sổ Terminal của IDE VS Code (Để mở terminal ấn Ctrl+`).

Như ví dụ ở hình dưới, ta tạo 1 app tên là Angular10CoBan. Vì thế, câu lệnh tạo project sẽ là **ng new Angular10CoBan** các bạn nhớ chuyển đến thư mục mình muốn bằng lênh **cd <đường dẫn thư mục>**  trước khi tạo project.



Đợi vài phút, quá trình khởi tạo diễn ra, bạn sẽ nhận được kết quả là thư mục smart-note. Trong này có một số thư mục và file:  
- **e2e** (end to end): thư mục sử dụng để test.  
- **node\_modules**: các package được cài đặt sử dụng trong project  
- **src**: code của project sẽ nằm ở đây. Phần dưới chúng ta sẽ tìm hiểu sâu hơn về thư mục này.  
- **.editorconfig**: file chứa cài đặt cho editor  
- **angular.json**: chứa các thông tin của project.  
- **.gitignore**: file chứa danh sách các file, folder không nằm trong danh mục commit. Bạn có thể tìm hiểu thêm về gitignore tại đây (https://git-scm.com/docs/gitignore)  
- **karma.conf.js**- file chứa các thông tin yêu cầu của project sử dụng cho unit testing.   
- **package.json:** file chứa thông tin các thư viện sử dụng trong project

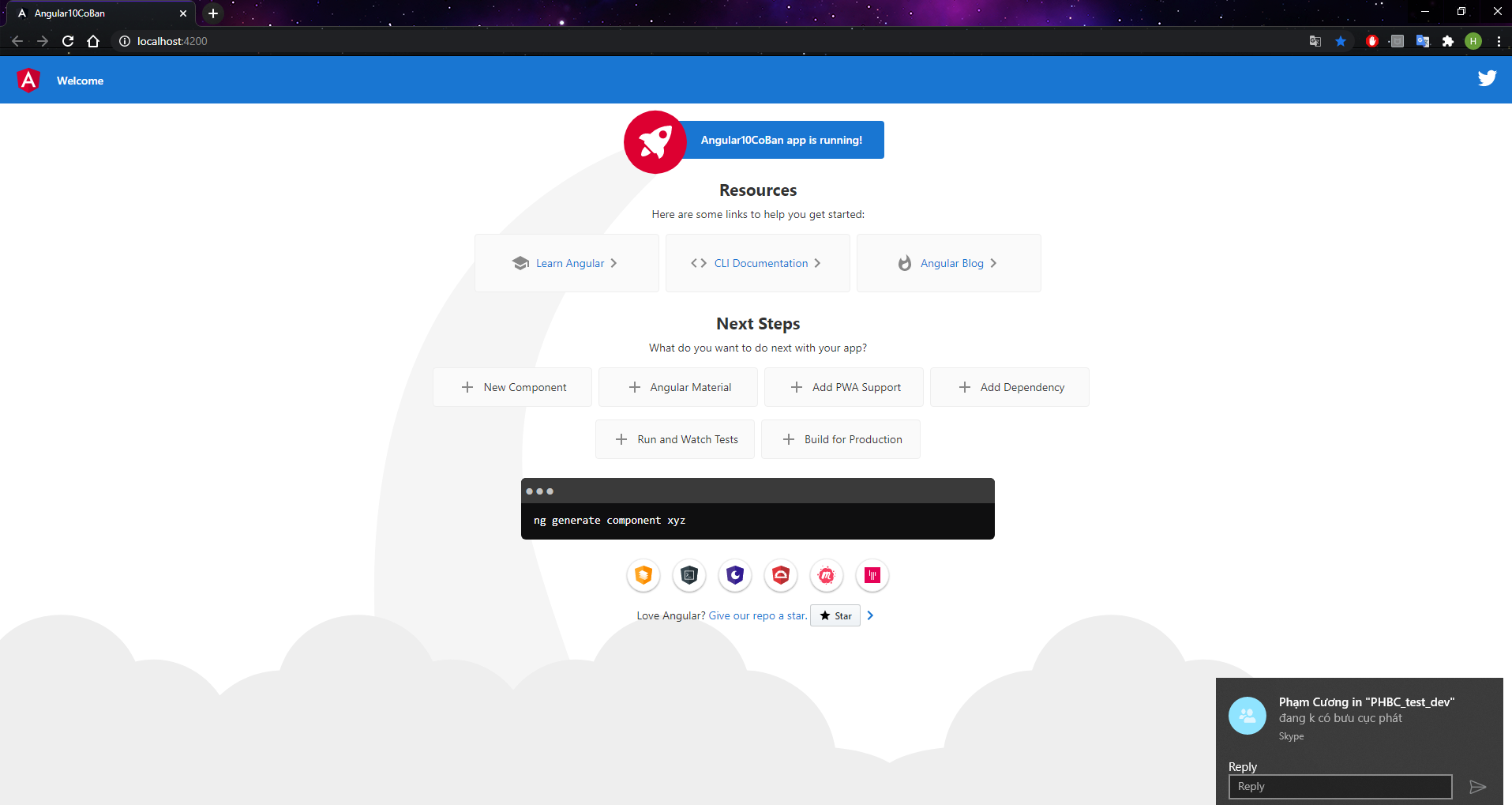
- **tsconfig.json:** chứa các quy tắc phụ khi project được dịch như dịch vào đâu (outDir), đường dẫn cơ bản là gì (baseUrl), phiên bản đích (target) là gì…

- **.tslint.json**(TypeScript lint): lintlà công cụ phân tích source code để đánh dấu các lỗi, bạn có thể đọc thêm về lint [ở đây](https://en.wikipedia.org/wiki/Lint_(software)). tslint là TypeScript lint - công cụ phân tích source code để đánh dấu lỗi cho typescript

## 2. Chạy project

Để chạy project, bạn chạy lệnh **ng serve** cũng có thể viết tắt **ng s**

Bạn có thể them –open để tự động mở trình duyệt sau khi biên dịch xong.



- Project sẽ được chạy mặc định tại <http://localhost:4200/>

- Khi bạn chạy lệnh ng serve, hệ thống sẽ biên dịch code ứng dụng của bạn (biên dịch từ TypeScript về Javascript) đồng thời tạo 1 server ở cổng 4200 (cổng mặc định)

- Để chạy ứng dụng ở 1 cổng tùy chọn, bạn thêm tham số --port <port\_number> vào sau ng serve. Ví dụ chạy ứng dụng ở cổng 3006, bạn chạy lệnh ng serve --port 3006

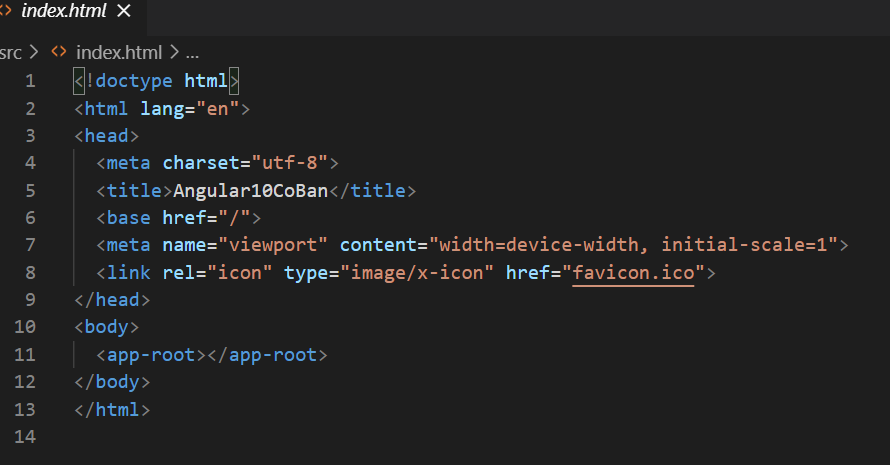
Dừng debug bằng Ctrl+c trong Termincal.

## 3. Cấu trúc thư mục src

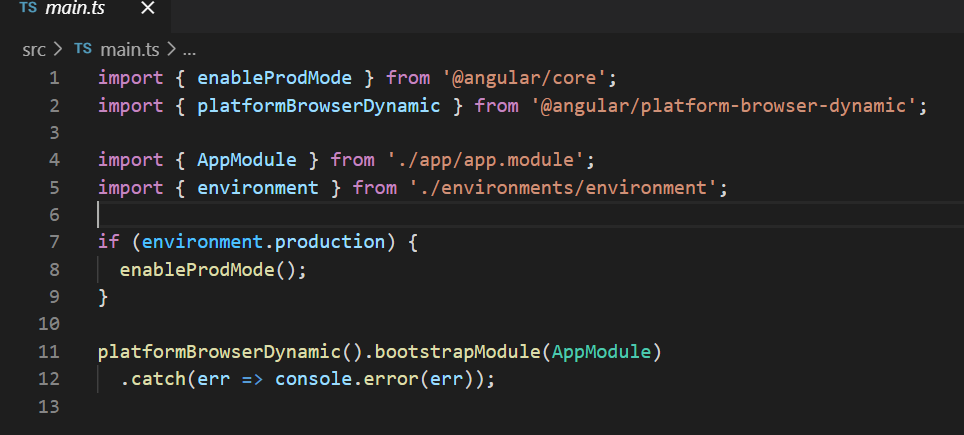
Thư mục src chứa mã nguồn của ứng dụng và các cài đặt liên quan. Cấu trúc thư mục bao gồm:

* **app**: thư mục chứa module chính của ứng dụng. Chúng ta sẽ tìm hiểu chi tiết về thư mục này trong bài sau.
* **assets**: thư mục chứa các file tài nguyên như css, js, hình ảnh
* **environment**: thư mục chứa các file cài đặt môi trường.
* **fav.ico**: file icon, hiển thị biểu tượng (logo) của website trên thanh tiêu đề của trình duyệt.
* **index.html**: file chính của chương trình.

Cùng nhìn sơ qua vào nội dung file **index.html**này:



Bên trong thẻ <body> có chứa thẻ <app-root></app-root>. Đây là selector được khai báo trong file **app.component.ts**, sử dụng để hiển thị dữ liệu từ file **app.component.html**. Bạn có thể hiểu đơn giản: ta định nghĩa thẻ <app-root> "đại diện" cho file **app.component.html**. Như vậy thẻ <app-root> đặt ở đâu thì nội dung html của file mà thẻ <app-root> này "đại diện" sẽ hiển thị ở đó.

* **main.ts**​: là file đầu tiên được chạy khi ứng dụng của bạn chạy. Trong này import các module cần thiết cho project. Bạn có thể thấy angular/core, angular/platform-browser-dynamic, app.module và environment được import mặc định khi ta tạo project bằng Angular CLI
* 

Khi dòng lệnh

platformBrowserDynamic().bootstrapModule(AppModule)

  .catch(err => console.error(err));

được thực thi sẽ gọi đến AppModule. AppModule được khai báo trong file **app.module**. Trong file **app.module** có sử dụng AppComponent trong file **app.component**. Trong **app.component** lại khai báo selector <app-root>. Thẻ <app-root> được gọi ở file **index.html**. Vậy file **index.html** được thực thi ở trình duyệt

Như vậy ta có luồng thực thi của ứng dụng như sau:

main.ts -> app.module.ts -> app.component.ts -> index.html

Tiếp tục là các file:

* **polyfill.ts**: file sử dụng cho việc tương thích ngược.
* **style.css**: file style cho ứng dụng
* **test.ts**: file chứa các test case cho project
* **tsconfig.app.json**: file chứa các cấu hình biên dịch, "hướng dẫn" cho trình dịch biên dịch code.
* **typings.d.ts**: file quản lý các định nghĩa trong Typescript.

Nhìn tổng quan, project của chúng ta có cấu trúc như sau:

