Angular

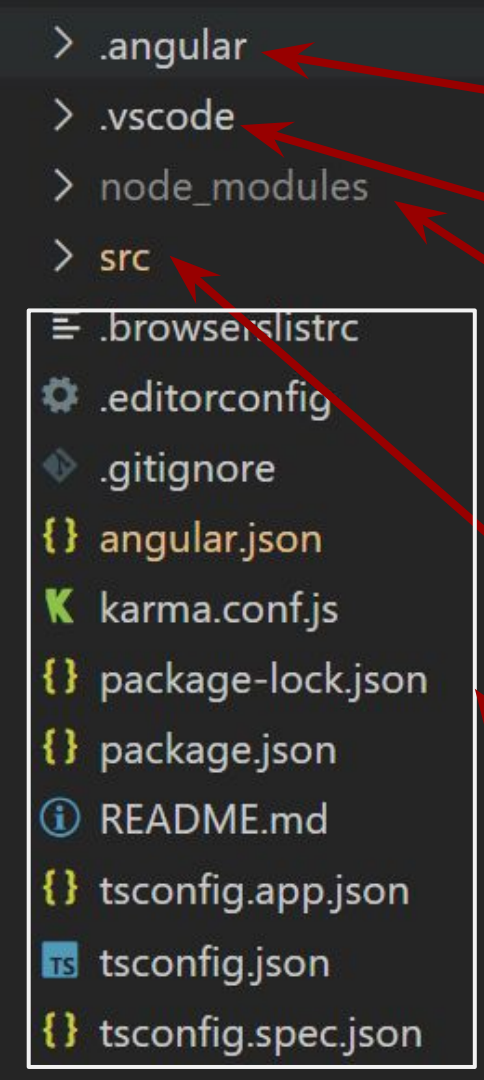
Chapitre 3: Structure et architecture d'un projet Angular

Enseignant: Mouhamed Amar

Licence 3
Université Amadou Hampaté BA de Dakar
Année 2022-2023

Plan

1. Architecture globale
2. src
3. Fichiers de configurations du projet
4. Component (création, structure et utilisation)
5. Module (création, structure et exemple)



l'espace attribué afin de stocker des informations facilement accessibles dans le but d'accélérer certaines tâches

Configurations de Vscode

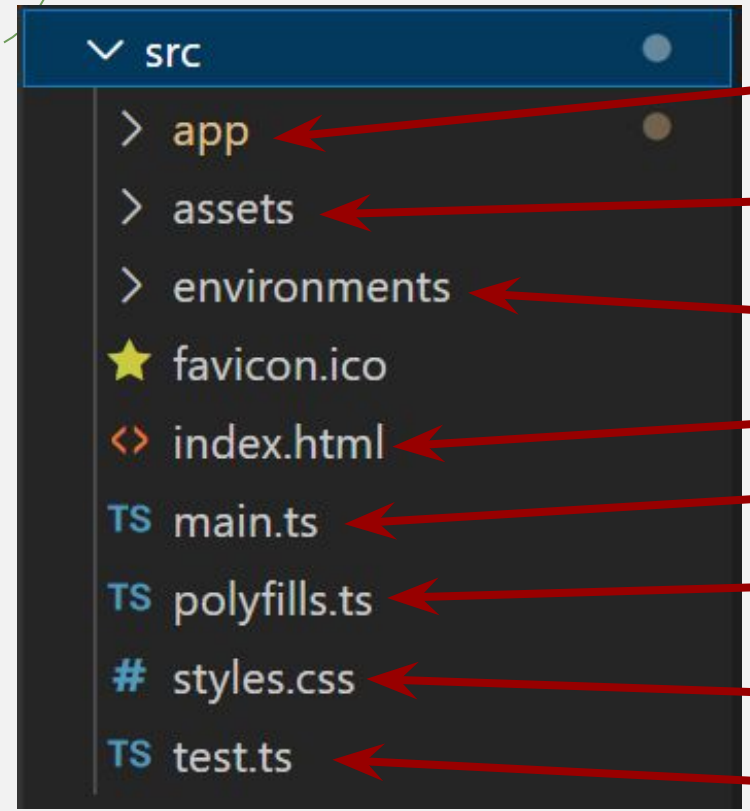
Emplacement des bibliothèques ou dépendances

Le corps de l'application Angular.

Configurations du projet Angular

Architecture globale

src (corps du projet)



Premier composant de l'application, composant parent. Il est directement inclus dans **index.html**

Emplacement des fichiers utilisés dans le projet (images,vidéos …)

Configurations de l'environnement de développement

La page mère de l'application

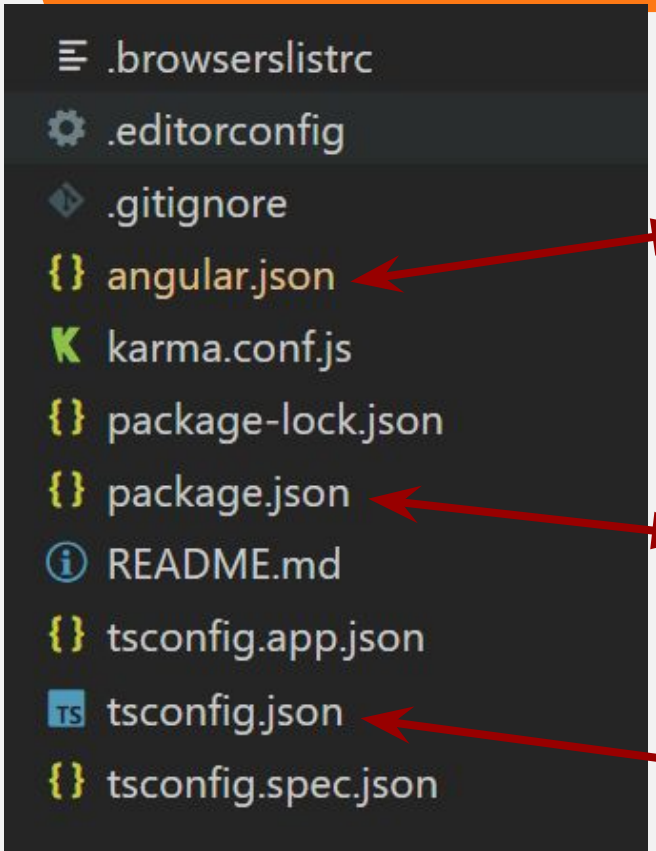
point d'entrée de votre application, compile et démarre l'application

Gère les éventuelles incompatibilités de certains navigateurs

Feuille de style global du projet

Gestions des tests

Fichiers de configurations



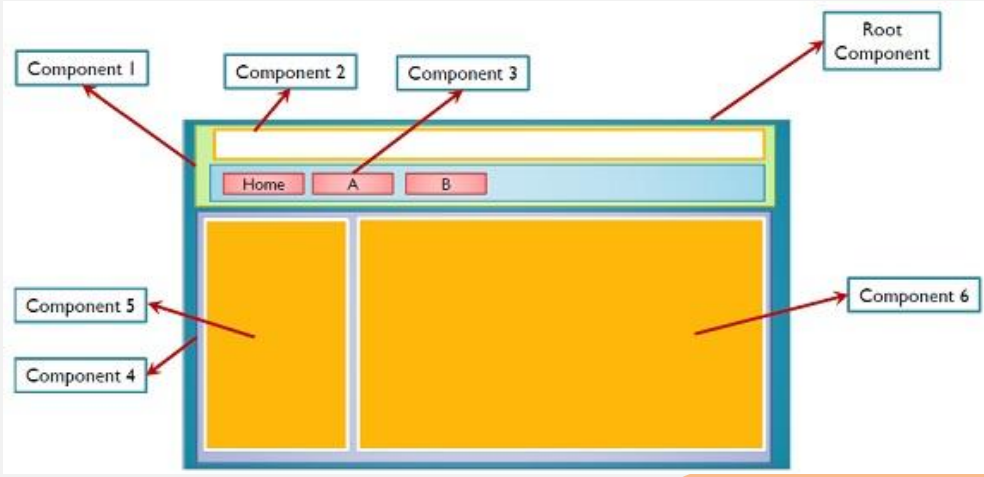
configurations par défaut de l'espace de travail utilisés par les outils de construction et de développement fournis par la Angular CLI

Ce fichier contient l'ensemble des dépendances du projet

spécifie les options de base du compilateur TypeScript et Angular

Composant

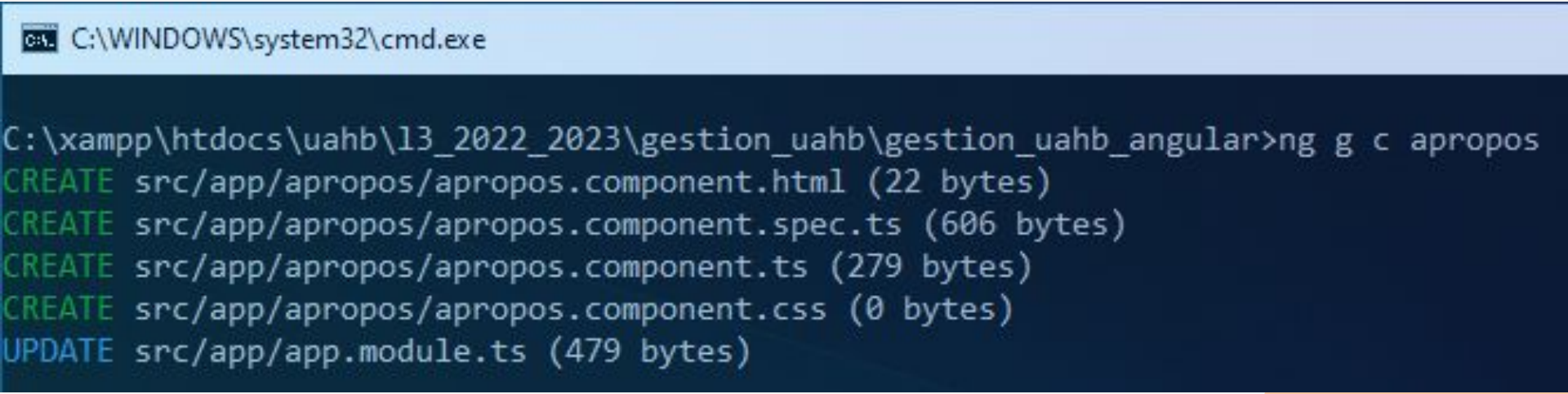
Un composant est une **maison** dans la construction d'une application Angular supposée être le **quartier** ou la **ville** concerné(e). Toutes les zones d'une application Angular sont produites par des composants.



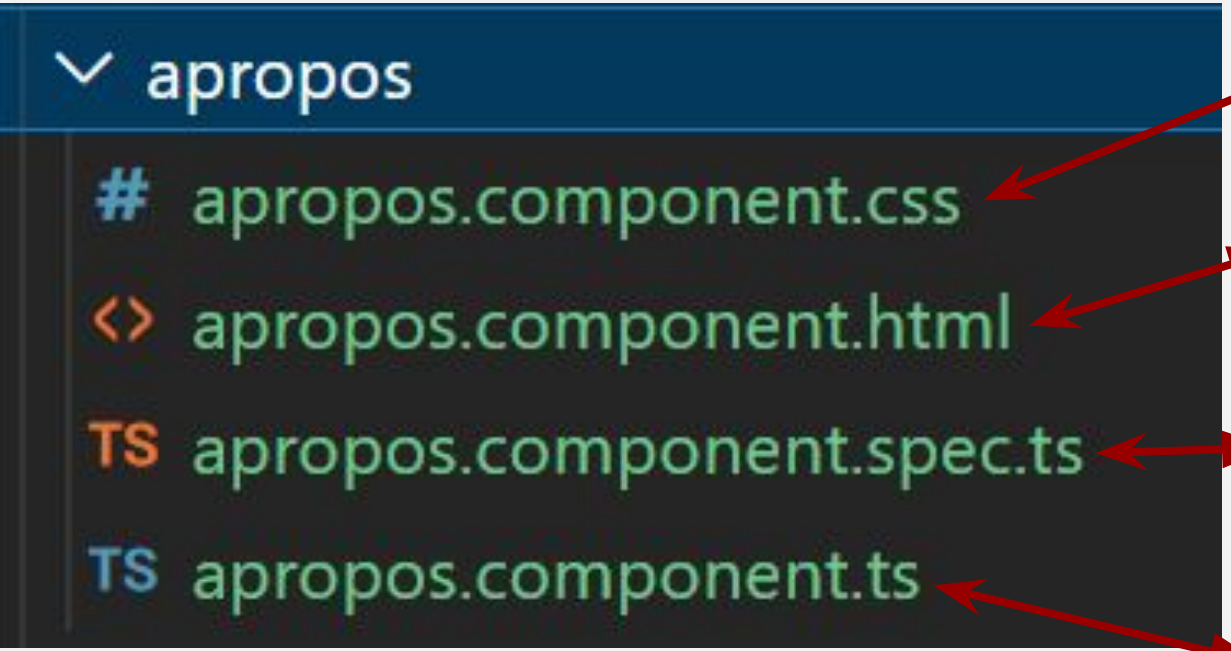
Composant: création

ng generate component /apropos

ng g c /apropos



Composant: structure



Code de **mise en forme** avec css

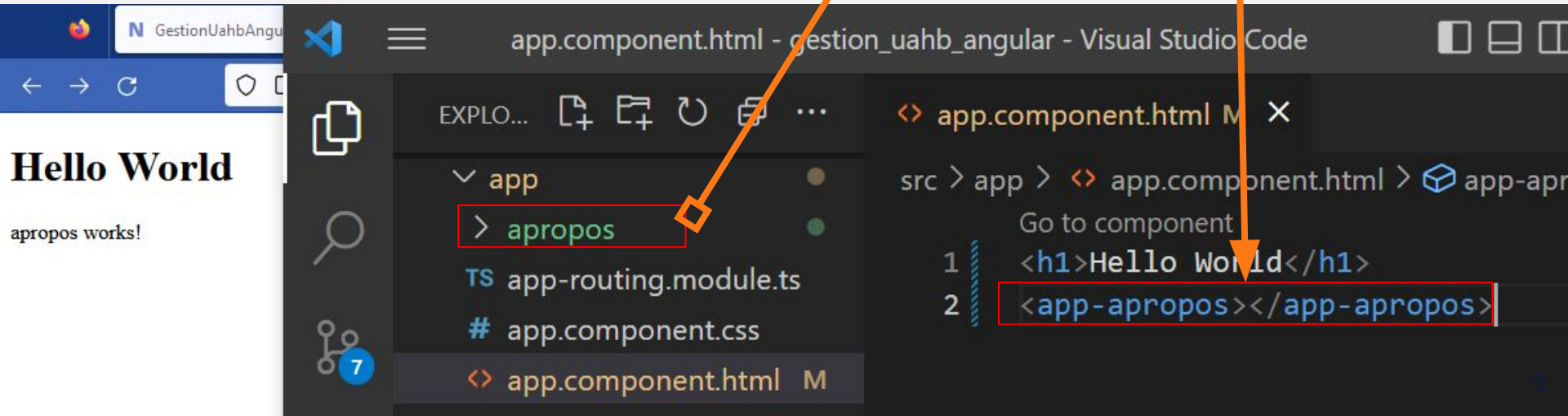
L'**interface visible** du composant

Les configurations pour les **tests**

La logique du composant avec **TypeScript**

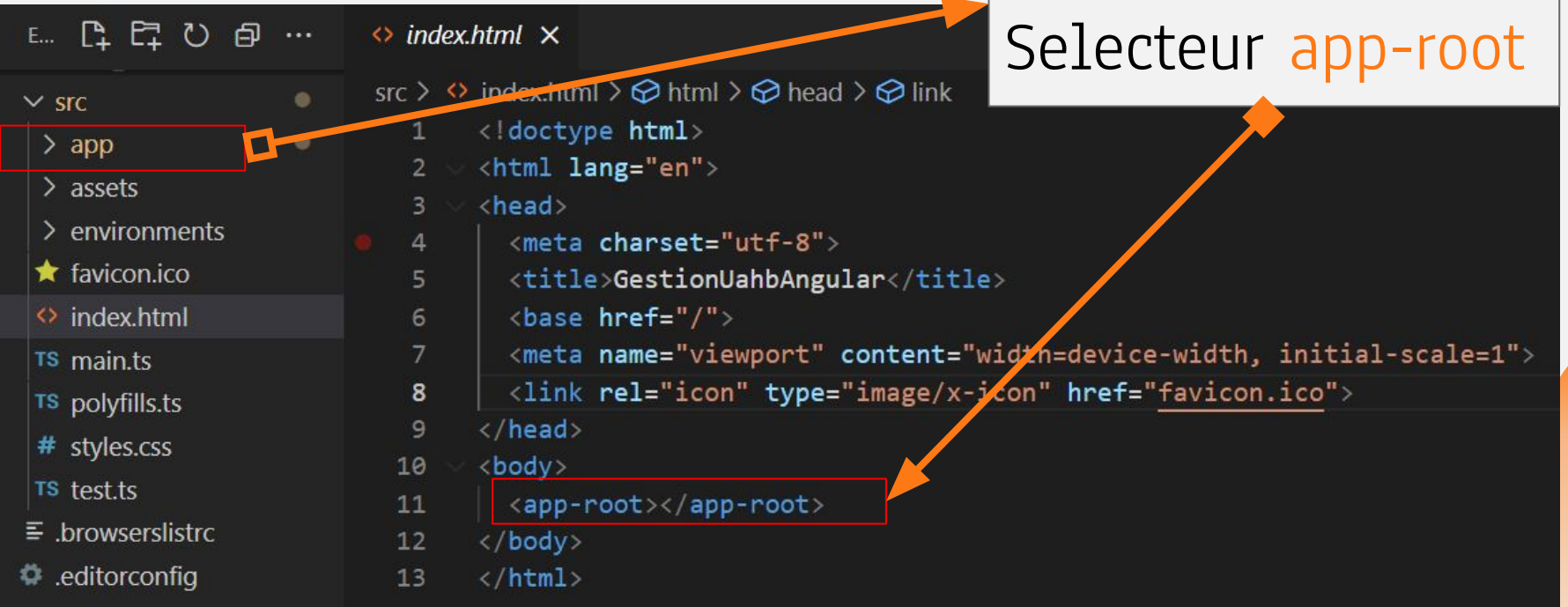
Composant: utilisation

Composant apropos
Selecteur app-apropos



Composant: utilisation

Composant app
Selecteur app-root



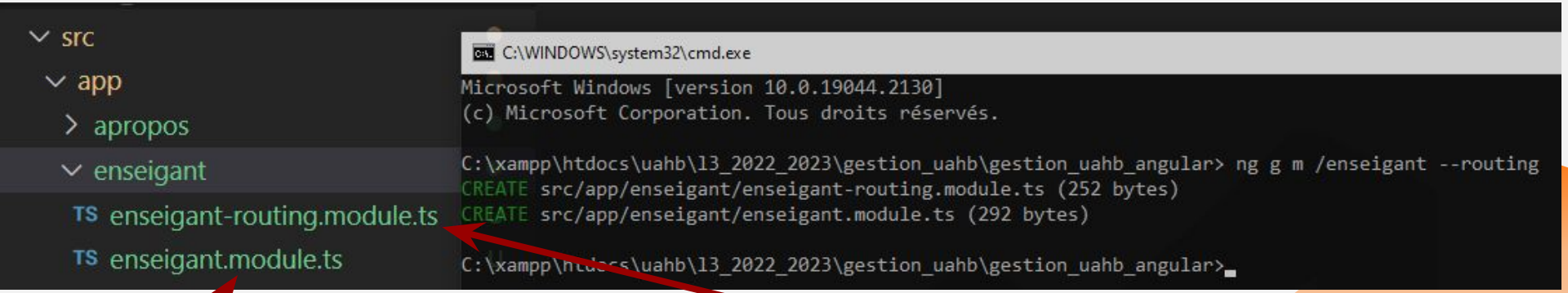
Module

Un **module** est la **mairie** dans la construction d'une application Angular supposée être le **quartier** ou la **ville** concerné(e). C'est un mécanisme permettant de regrouper et d'organiser des **composants**, définir leurs dépendances, leurs visibilités ...

Module création

ng generate **module** /enseignant **--routing**

ng g **m** /enseignant **--routing**



Module **enseigant**

Module de routage **enseigant**

Module: structure

TS enseigant.module.ts U

src > app > enseigant > TS enseigant.module.ts > EnseigantModule

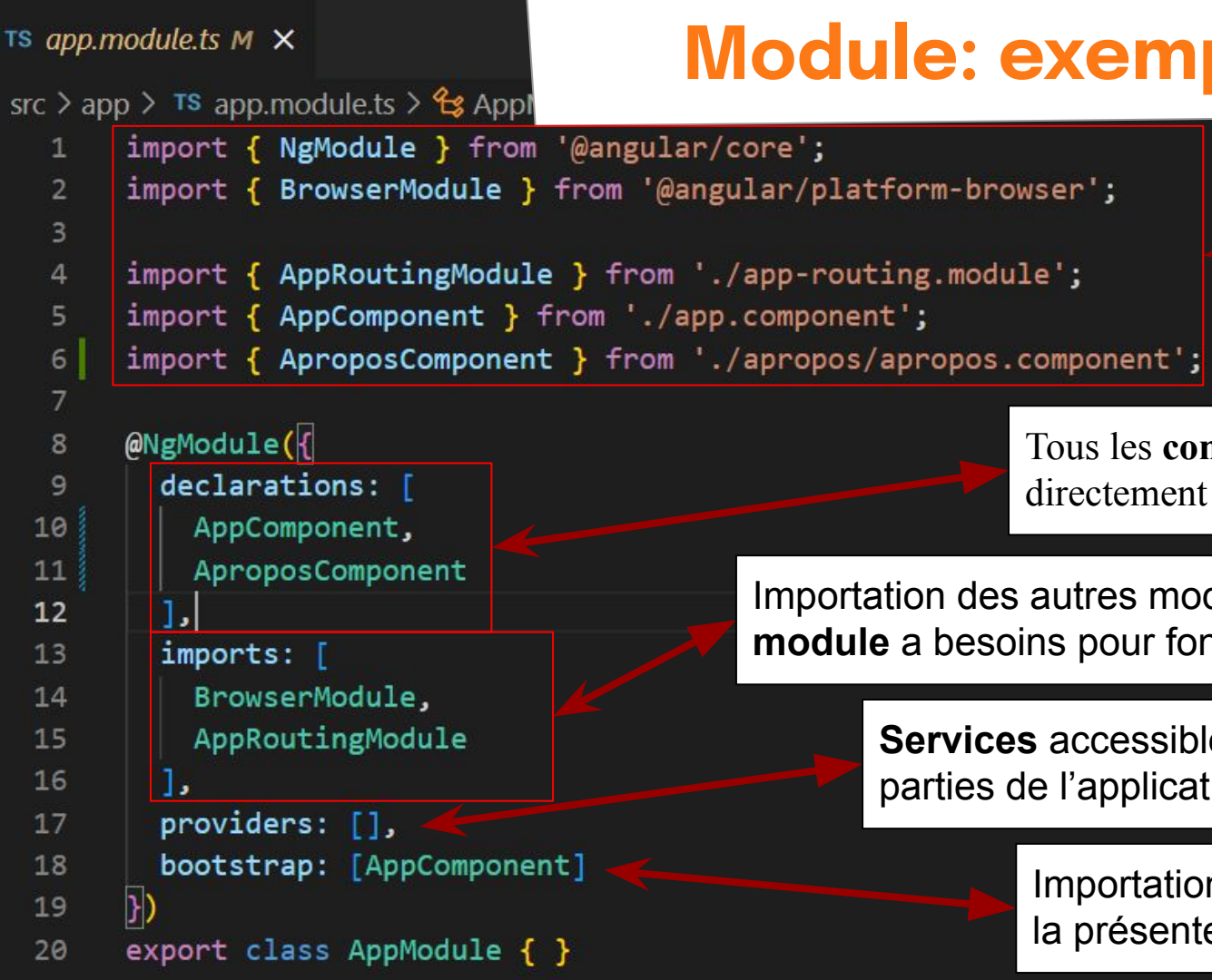
```ts
import { NgModule } from '@angular/core';
import { CommonModule } from '@angular/common';

import { EnseigantRoutingModule } from './enseigant-routing.module';


@NgModule({
  declarations: [],
  imports: [
    CommonModule,
    EnseigantRoutingModule
  ]
})
export class EnseigantModule { }
```

Importations des dépendances

Décorateur qui montre qu'une classe est une **module** et fournit des métadonnées de configuration qui déterminent comment la module doit être traitée, instanciée et utilisée lors de l'exécution.

La classe proprement dite

Module: exemple



Importations des **dépendances**

Tous les **composants** qui dépendent directement de cette présente module

Importation des autres modules dont la présente **module** a besoins pour fonctionner

Services accessibles dans toutes les parties de l'application

Importation des autres modules dont la présente **module** a besoins

FIN

De chapitre

Pour plus de ressources, vous pouvez consulter:
https://angular.io
https://blog.jant.tech