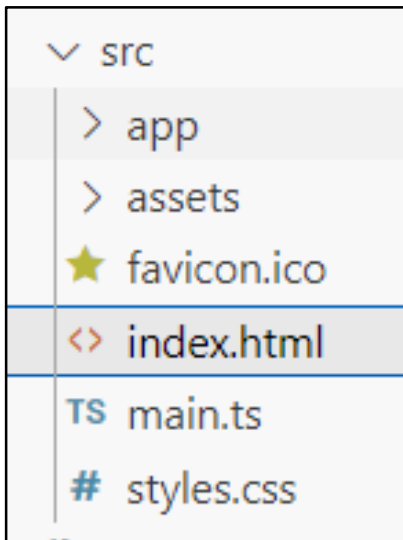


Angular

1.3. Cambios en los componentes

1. app.component.ts

- Hemos visto en el ejemplo anterior el contenido del fichero **index.html** que está en la carpeta **src**.



```
<> index.html X
src > <> index.html > ...
2   <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <title>Proyecto1</title>
6     <base href="/">
7     <meta name="viewport" content="width=device-width,
8       initial-scale=1">
9     <link rel="icon" type="image/x-icon" href="favicon.
10      ico">
11   </head>
12   <body>
13     <app-root></app-root>
14   </body>
15 </html>
```

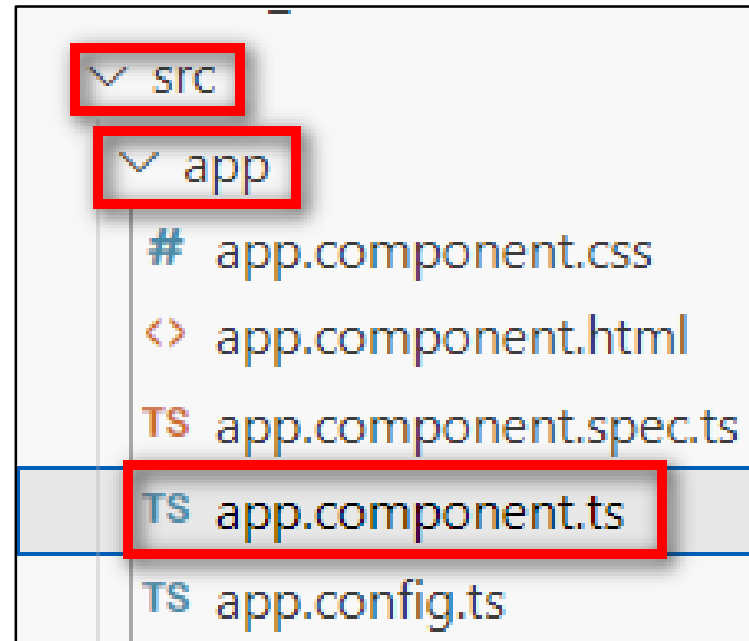
1. app.component.ts

- El fichero index.html carga una etiqueta llamada **<app-root>**

```
<> index.html X
src > <> index.html > ...
2   <html lang="en">
3   <head>
4     <meta charset="utf-8">
5     <title>Proyecto1</title>
6     <base href="/">
7     <meta name="viewport" content="width=device-width,
8       initial-scale=1">
9     <link rel="icon" type="image/x-icon" href="favicon.ico">
10  </head>
11  <body>
12    <app-root></app-root>
13  </body>
14  </html>
```

1. app.component.ts

- Dentro de la carpeta **src/app** tenemos un fichero TypeScript llamado **app.component.ts**.



1. app.component.ts

- Observamos el contenido del fiche ***app.component.ts***

```
TS app.component.ts X
src > app > TS app.component.ts > ...
1  import { Component } from '@angular/core';
2  import { CommonModule } from '@angular/common';
3  import { RouterOutlet } from '@angular/router';
4
5  @Component({
6    selector: 'app-root',
7    standalone: true,
8    imports: [CommonModule, RouterOutlet],
9    templateUrl: './app.component.html',
10   styleUrls: ['./app.component.css']
11 })
12 export class AppComponent {
13   title = 'proyecto1';
14 }
```

1. app.component.ts

- En el decorador **@Component** tenemos:


```
@Component({  
  selector: 'app-root',  
  standalone: true,  
  imports: [CommonModule, RouterOutlet],  
  templateUrl: './app.component.html',  
  styleUrls: ['./app.component.css']  
})
```

→ Esta será la etiqueta con la que se conocerá en HTML

1. app.component.ts

- Especificamos también la plantilla HTML y CSS que pertenecerán a mi componente.

```
@Component({  
  selector: 'app-root',  
  standalone: true,  
  imports: [CommonModule, RouterOutlet],  
  templateUrl: './app.component.html',  
  styleUrls: ['./app.component.css']  
})
```



Fichero HTML del componente

Fichero CSS del componente

1. app.component.ts

- El fichero también tendrá la clase TypeScript **AppComponent** donde desarrollaremos el código de este componente.

```
export class AppComponent {  
  title = 'proyecto1';  
}
```


1. app.component.ts

- Si modificamos el texto de **title** podremos observar los cambios en la aplicación.

```
export class AppComponent {  
  title = 'Esto es un ejemplo de Angular';  
}
```



localhost:4200

Hello, Esto es un ejemplo de Angular

Congratulations! Your app is running. 🎉