





Geeks México

BLOG DE PROGRAMACIÓN EN ESPAÑOL SOBRE JAVA,
FRAMEWORKS, BASES DE DATOS, CÓMPUTO EN LA NUBE, ETC.
EN ESPAÑOL Y EN INGLÉS.

[TUTORIALES EN ESPAÑOL](#)[TUTORIALS IN ENGLISH](#)[ABOUT](#)[CONTACT](#)

Anuncios

 	12GA DC 6V 1...	1Pcs I2C NAN...	Micro 130 DC...	1x DC3V-6V D...
	AliExpr...	AliExpr...	AliExpress	AliExpr...
	\$2.76 motor m...	\$1.02 module...	\$0.65 mini itx t...	\$0.66 dc gear...

Crear componentes en

Angular 4

📅 [HACE 2 SEMANAS](#) 💬 [DEJA UN COMENTARIO](#)

Como se explicó en el primer post de Angular 4, la forma en la que se crea una aplicación es a través de componentes. En este post se explicará como crear componentes manualmente y haciendo uso de la CLI(Command line interface). Se tomará como base el post [Configura Bootstrap en Angular 4](https://geeks-mexico.com/2017/09/12/configura-bootstrap-en-angular-4/) (<https://geeks-mexico.com/2017/09/12/configura-bootstrap-en-angular-4/>).

Una vez que se tiene un proyecto construido veremos un componente llamado **app.component** que será utilizado para cargar el contenido al archivo index.html. A continuación veremos como crear componentes adicionales a nuestra aplicación.

Creación de componentes manualmente

A continuación crearemos un componente nuevo llamado posts siguiendo los siguientes pasos:

- Crear un folder con el nombre del componente dentro del folder src/app/ , en este caso nuestro folder sería **/src/app/posts**.
- Dentro del folder /src/app/posts crearemos los siguientes archivos:
 - **posts.component.html** : Contendrá el HTML que se desplegará donde se inserte el componente
 - **posts.component.ts** : Contendrá una clase typescript que definirá el componente.

- Dentro del folder /src/app editaremos el archivo **app.modules.ts** para registrar el nuevo componente.

Paso 1 Editar el archivo app.component.html

El primer paso para este ejemplo será editar el archivo app.component.html con el siguiente código:

```
1 <div align="center">
2 <h1>Geeks México</h1>
3 <h3>Base component</h3>
4 
2 <h3>Post List</h3>
3 <div>
4 <table class="table">
5 <tr>
6 <th>Author Twitter</th>
7 <th>Post name</th>
8 <th>Post date</th>
9 </tr>
10 <tr>
11 <td>@raidentrance</td>
12 <td>Angular from scratch</td>
13 <td>10/10/2017</td>
14 </tr>
15 </table>
16 </div>
17 </div>
```

3.- Crear el archivo **posts.components.ts** dentro del folder `/src/app/posts`.

```
1 import { Component } from '@angular/core'
2 @Component({
3   selector: 'app-posts',
4   templateUrl: './posts.component.html'
5 })
6 export class PostsComponent {
7
8 }
```

Paso 3: Incluir el nuevo componente en el archivo `app.modules.ts`

Una vez que ya se creó el componente el siguiente paso es registrarlo dentro del archivo **app.modules.ts**. Para hacerlo debemos editar el archivo como se muestra a continuación:

```
1 import { BrowserModule } from '@angular/platform-browser';
2 import { NgModule } from '@angular/core';
3
4 import { AppComponent } from './app.component';
5 import { PostsComponent } from './posts.component';
6
7 @NgModule({
8   declarations: [
9     AppComponent, PostsComponent
```

```
10 ],
11 imports: [
12   BrowserModule
13 ],
14 providers: [],
15 bootstrap: [AppComponent]
16 })
17 export class AppModule { }
```

Las modificaciones que se harán al archivo

app.modules.ts son las siguientes:

1. Incluir la sentencia `import` para `PostsComponent` desde `./posts/posts.component`
2. Incluir `PostsComponent` en el objeto `declarations`

Paso 4: Incluir el nuevo componente en el componente principal

Para hacer esto lo único que debemos hacer es editar el archivo **app.components.html** Como se muestra a continuación:

```
1 <div align="center">
2 <h1>Geeks México</h1>
3 <h3>Base component</h3>
4 <img class="img-fluid" src="https://geeksjc
5 <app-posts></app-posts>
```

Como se puede observar, la única modificación fue agregar al final el selector de nuestro componente llamado .

Una vez concluido lo anterior la página se verá como se muestra a continuación:

Geeks México

Base component



Post List

Author Twitter	Post name	Post date
@raidentrance	Angular from scratch	10/10/2017

Creando un componente utilizando la CLI

Crear un componente utilizando la línea de comandos es mucho más sencillo, solo debes ejecutar el siguiente comando:

```
1 | ng generate component posts
```

Una vez hecho esto verás que todos los archivos necesarios fueron creados automáticamente y que lo único que resta es editar los archivos para tener todo listo. Si tienes alguna pregunta sobre como hacerlo tu mismo, agrégala a los comentarios.

Autor: Alejandro Agapito Bautista

Twitter: @raidentrance

Contacto:raidentrance@gmail.com

Anuncios

