



Proyecto Angular

06/03/2022

Álvaro López Álvarez

IES Ruiz Gijón - Utrera

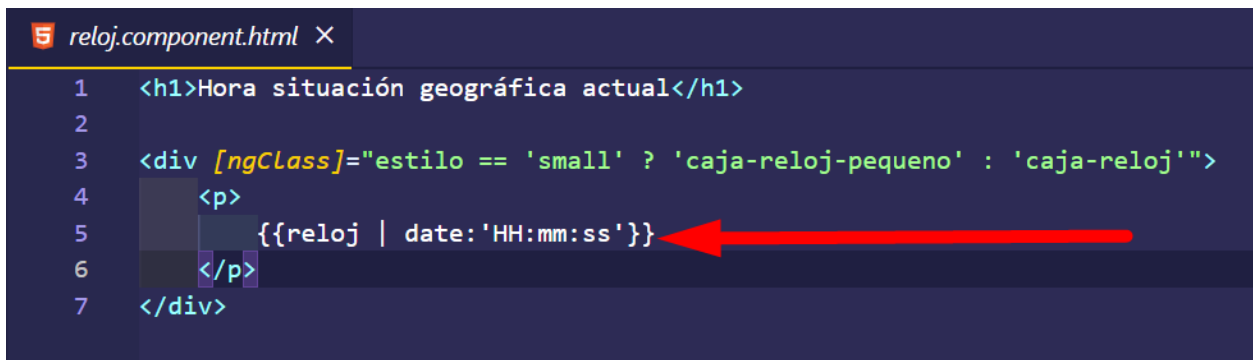
Ficheros y documentos a entregar:

- Carpeta comprimida del proyecto Angular sin contener la carpeta de dependencias (node_modules)
- Fichero sql con la estructura y datos de la BD
- Despliegue en máquina ruix
- Documento que contenga la url a tu proyecto desplegado en ruix y con capturas del código donde expliques en qué parte del código has implementado los requisitos exigidos (pipes, routing, validación personalizada, etc.)

<https://ruizgijon.ddns.net/lopeza/angular/proyecto-angular/>

-Pipes:

Formateo de fecha y hora.

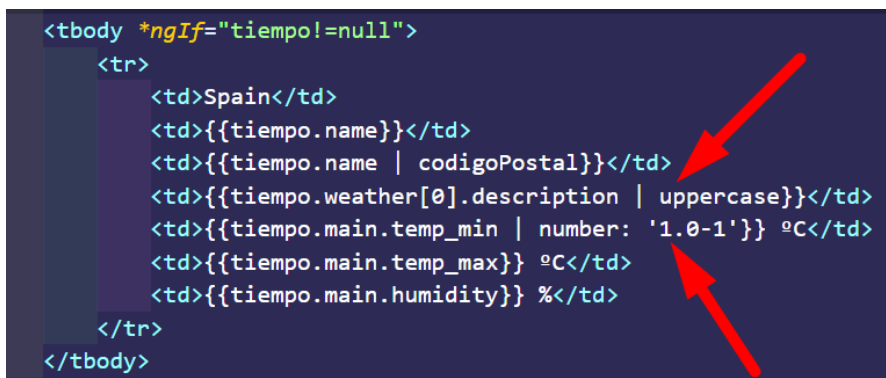


```

1 <h1>Hora situación geográfica actual</h1>
2
3 <div [ngClass]="estilo == 'small' ? 'caja-reloj-pequeno' : 'caja-reloj'">
4   <p>
5     {{reloj | date:'HH:mm:ss'}}
6   </p>
7 </div>

```

Formateo de decimales y mayúsculas.



```

<tbody *ngIf="tiempo!=null">
  <tr>
    <td>Spain</td>
    <td>{{tiempo.name}}</td>
    <td>{{tiempo.name | codigoPostal}}</td>
    <td>{{tiempo.weather[0].description | uppercase}}</td>
    <td>{{tiempo.main.temp_min | number: '1.0-1'}} °C</td>
    <td>{{tiempo.main.temp_max}} °C</td>
    <td>{{tiempo.main.humidity}} %</td>
  </tr>
</tbody>

```

-Pipe personalizada que saca cod postal Utrera.

```

A codigo-postal.pipe.ts X
1  import { Pipe, PipeTransform } from '@angular/core';
2
3  @Pipe({
4    name: 'codigoPostal'
5  })
6  export class CodigoPostalPipe implements PipeTransform {
7
8    transform(lugar: string, ...args: any[]): number {
9
10     let codigoPostal = 0;
11
12     if(lugar == 'Utrera') {
13       codigoPostal = 41710;
14     }
15
16
17     return codigoPostal;
18
19   }
20
21 }
22

```

-Routing y angular material con navbar:

```

5 barra-lateral.component.html X
1  <mat-sidenav-container class="sidenav-container">
2    <mat-sidenav #drawer class="sidenav" fixedInViewport [attr.role]="(isHandset$ | async) ? 'dialog' : 'navigation'"
3      [mode]="(isHandset$ | async) ? 'over' : 'side'" [opened]="(isHandset$ | async) === false">
4      <mat-toolbar>Menu</mat-toolbar>
5      <mat-nav-list>
6        <a mat-list-item routerLink="tiempo">Tiempo Utrera</a>
7        <a mat-list-item routerLink="reloj">Reloj</a>
8        <a mat-list-item routerLink="Formulario">Formulario</a>
9
10       <div style="position: absolute; bottom: 0;">
11         <app-reloj [estilo]="small" (multiplo10)="actualizar($event)"></app-reloj>
12         <h3 style="margin: 1em;">{{mensaje}} (se actualiza cada 10 segundos y se resetea cada minuto)</h3>
13       </div>
14     </mat-nav-list>
15   </mat-sidenav>
16   <mat-sidenav-content>
17     <mat-toolbar color="primary">
18       <button type="button" aria-label="Toggle sidenav" mat-icon-button (click)="drawer.toggle()"
19         *ngIf="isHandset$ | async">
20         <mat-icon aria-label="Side nav toggle icon">menu</mat-icon>
21       </button>
22       <span>Poyecto Álvaro López Álvarez</span>
23     </mat-toolbar>
24     <!-- Add Content Here -->
25     <router-outlet></router-outlet>
26   </mat-sidenav-content>
27 </mat-sidenav-container>

```

```
app-routing.module.ts X
1 import { NgModule } from '@angular/core';
2 import { RouterModule, Routes } from '@angular/router';
3 import { TiempoComponent } from '../tiempo/tiempo.component';
4 import { RelojComponent } from '../reloj/reloj.component';
5 import { FormularioComponent } from '../formulario/formulario.component';
6
7 const routes: Routes = [
8   {
9     path: 'tiempo',
10    component: TiempoComponent
11  },
12  {
13    path: 'reloj',
14    component: RelojComponent
15  },
16  {
17    path: 'formulario',
18    component: FormularioComponent
19  }
20 ];
21
22 @NgModule({
23   imports: [RouterModule.forRoot(routes)],
24   exports: [RouterModule]
25 })
26 export class AppRoutingModule { }
27
```

Ahí indicamos el nombre de las routes y el componente al que asociamos.

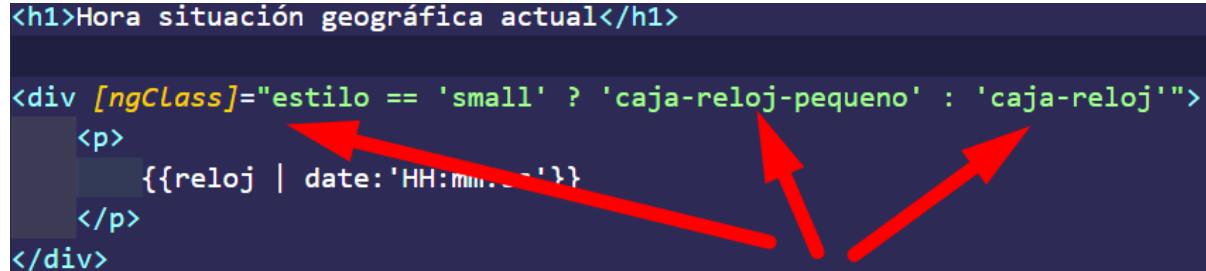
- validación personalizada:

Valida el código postal con una expresión regular.

```
TS validaciones-propias.ts X
1 import { AbstractControl, FormControl, ValidationErrors } from '@angular/forms';
2
3 export class ValidacionesPropias {
4
5     static codPostal(control: AbstractControl): ValidationErrors | null {
6         let post: string = control.value;
7         Test Regex...
8         let patron = /\d{5}$/;
9         if (patron.test(post) || post == "") {
10             return null;
11         } else {
12             return { codPos: true }
13         }
14     }
15 }
16
17
18
19
```

- property binding: Cambia la clase y según el valor. Cada clase tiene distintos estilos.

```
<h1>Hora situación geográfica actual</h1>
<div [ngClass]="estilo == 'small' ? 'caja-reloj-pequeno' : 'caja-reloj'">
  <p>
    {{reloj | date:'HH:mm:ss'}}
  </p>
</div>
```



- Debe tener un componente donde inyectes un servicio que realice una petición al API usado en tu práctica de JQuery:

Toma los datos del json que devuelve la api "tiempo.datos".

```

5 tiempo.component.html X
1 <h1>Tiempo Utrera</h1>
2 <table id="ciudades">
3   <thead>
4     <th>Country</th>
5     <th>City</th>
6     <th>Postal Code</th>
7     <th>Sky</th>
8     <th>Temp. Min</th>
9     <th>Temp. Max</th>
10    <th>Humidity</th>
11  </thead>
12  <tbody *ngIf="tiempo!=null">
13    <tr>
14      <td>Spain</td>
15      <td>{{tiempo.name}}</td>
16      <td>{{tiempo.name | codigoPostal}}</td>
17      <td>{{tiempo.weather[0].description | uppercase}}</td>
18      <td>{{tiempo.main.temp_min | number: '1.0-1'}} °C</td>
19      <td>{{tiempo.main.temp_max}} °C</td>
20      <td>{{tiempo.main.humidity}} %</td>
21    </tr>
22  </tbody>
23 </table>

```

Esa es la API con su clave y los dos valores que le pasamos.

```

servicio-tiempo.service.ts X
1 import { Injectable } from '@angular/core';
2 import { HttpClient } from '@angular/common/http';
3
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class ServicioTiempoService {
8
9   constructor(private http: HttpClient) {}
10
11   retornar() {
12     return this.http.get("http://api.openweathermap.org/data/2.5/weather?q=utrera,pais&units=metric&appid=fff70737bb887cd2adb9bd7e2a052d1c");
13   }
14 }
15

```

Clave de mi API

Importante importar esto o da fallo:

```
import { HttpClientModule } from '@angular/common/http';
```