

## Práctica 4.5: Usar API Swagger: Read

Rama git: **p\_read**

Rama resuelta: **p\_read\_resuelta**

En **edificios.component.html**:

1. Añadir un desplegable (**mat-select**) que obtenga todas las provincias ( de **getProvincias()** ).
2. Añadir un botón “Filtrar” que ejecute el método **onFiltrar()**

En **edificios.component.ts**:

3. Añadir las propiedades necesarias para guardar las provincias y la provincia seleccionada
4. En **onFiltrar()** recuperar los edificios de la provincia seleccionada en el desplegable

### SOLUCIÓN

En **edificios.component.html**:

1. Creamos un desplegable **<mat-select>** vinculado a **provinciaSeleccionada**:

```
<mat-select [(value)]="provinciaSeleccionada" placeholder="Filtrar provincia">
```

2. En el **<mat-option>** metemos un **\*ngFor** que recorra el contenido de provincias mostrando el nombre (**provincia.provincia**) y que guarde **provincia.id**:

```
<mat-option *ngFor="let provincia of provincias" [value]="provincia.id">
  {{ provincia.provincia }} </mat-option>
```

3. Añadimos un botón vinculado al método **onFiltrar()**:

```
<button type="button" (click)="onFiltrar()" mat-raised-button>Filtrar</button>
```

Así quedaría todo junto:

```

6 | <mat-form-field>
7 |   <!-- Práctica 4.5: Usar API Swagger: Read
8 |       Alguien debería añadir un desplegable <mat-select> que muestre todas las provincias
9 |       y que vincule el valor seleccionado a provinciaSeleccionada -->
10 |   <mat-select [(value)]="provinciaSeleccionada" placeholder="Filtrar provincia">
11 |     <mat-option *ngFor="let provincia of provincias" [value]="provincia.id">
12 |       {{ provincia.provincia }}
13 |     </mat-option>
14 |   </mat-select>
15 | </mat-form-field>
16 | <!-- Práctica 4.5: Usar API Swagger: Read
17 |       Alguien debería añadir un botón Filtrar vinculado al método onFiltrar() -->
18 | <button type="button" (click)="onFiltrar()" mat-raised-button>Filtrar</button>

```

En **edificios.component.ts**:

4. declaramos las propiedades:
  - **provincias** (tipo Provincia) para recoger las provincias del servicio
  - **provinciaSeleccionada** (tipo number) para recoger el **id** de la provincia seleccionada

```

provincias: Provincia [] = [];
provinciaSeleccionada: number = null;

```

```

19 | export class EdificiosComponent implements OnInit, AfterViewInit {
20 |   edificios: Edificio [] = [];
21 |   edificioSeleccionado: Edificio=null;
22 |   // Práctica 4.5: Usar API Swagger: Read
23 |   // Alguien debería declarar las propiedades:
24 |   // - provincias (tipo Provincia) para recoger las provincias del
25 |   // - provinciaSeleccionada (tipo number) para recoger el id de la
26 |   provincias: Provincia [] = [];
27 |   provinciaSeleccionada: number = null;

```

5. En el **constructor** nos suscribimos al servicio **getProvincias()** que proporciona todas las provincias, y asignamos la respuesta a la propiedad provincias

```

this.defaultService.getProvincias().subscribe(      response => {
    this.provincias = response;  }  );

```

```

33 | constructor(private edificioMockService: EdificioMockService, private defaultService: DefaultService,
34 |             // Práctica 4.5: Usar API Swagger: Read
35 |             // Alguien debería suscribirse al servicio que proporciona todas las provincias
36 |             // y asignar la respuesta a la propiedad provincias:
37 |             this.defaultService.getProvincias().subscribe(
38 |               response => {
39 |                 this.provincias = response;
40 |               }
41 |             );
42 | }

```

6. En el método **onFiltrar** nos suscribimos al servicio **getEdificios** que proporciona todos los edificios (indicando que sólo queremos los de la **provinciaSeleccionada**, para ello se lo pasamos

como **segundo parámetro**)

y asignamos la respuesta al datasource.data:

```
this.defaultService.getEdificios('',this.provinciaSeleccionada).subscribe(  
data => { this.dataSource.data = data; } )
```

Lo metemos todo en un **if** por si dan al botón sin haber seleccionado ninguna provincia:

```
109 onFiltrar(){  
110     // Práctica 4.5: Usar API Swagger: Read  
111     // Alguien debería suscribirse al servicio que proporciona todos los edificios  
112     // (indicando que sólo quiere los de la provinciaSeleccionada)  
113     // y asignar la respuesta al datasource.data:  
114     if (this.provinciaSeleccionada){  
115         this.defaultService.getEdificios('',this.provinciaSeleccionada).subscribe(  
116             data => {  
117                 this.dataSource.data = data;  
118             }  
119         )  
120     }  
121 }
```