

Spring Data REST: Proyecciones

Elementos avanzados en tu API REST con Spring Boot

Proyección

- Por defecto, Spring Data REST exporta los objetos de dominio tal cual están definidos.
 - Los datos primitivos, se exportan tal cual
 - Para los datos que asocian con otro modelo
 - Si hay un repositorio exportado, como una URI (también objeto anidado *embedded*)
 - Si no hay repositorio exportado, como un objeto anidado.

Proyección

- Nos puede interesar tener algunos endpoints en los cuales los datos que se obtengan del objeto no sean todos, sino solamente unos pocos.
- Por ejemplo, ciudad sin país ni coordenadas

Proyecciones con Spring Data REST

- Se definen a través de una interfaz
- Uso de la anotación `@Projection`
 - Se indica el nombre de la proyección y la clase sobre la que se define.
- Un método getter por cada dato que queremos exportar.

```
@Projection(name = "ciudadSinUbicacion", types = { Ciudad.class })  
public interface CiudadProj {  
    String getNombre();  
}
```

Uso de proyecciones

- Si ejecutamos ahora la aplicación podemos consumir la url de una ciudad. Por ejemplo <http://localhost:8080/api/ciudades/29>
- Podemos ver que aparece una nueva entrada en el JSON

```
"ciudad": {  
  "href": "http://localhost:8080/api/ciudades/29{?projection}",  
  "templated": true  
},
```

- Nos dice que se puede pasar un parámetro con el nombre de la proyección. Por ejemplo <http://localhost:8080/api/ciudades/29?projection=ciudadSinUbicacion>

Proyecciones más complejas

- Podemos definir proyecciones más complejas
- Uso de @Value y SpEL
- Podemos obtener datos de asociaciones o construir datos derivados.
- Por ejemplo, incluir el nombre del país, no el objeto completo.
- Incluir también las coordenadas en formato *lat, lng*.