

REST API - Read Only



REST API

I don't want that!
I want the REST API as READ-ONLY

- Spring Data REST will expose ALL these endpoints for ~~free!~~

HTTP Method	CRUD Action
X POST /products	Create a new product
✓ GET /products	Read a list of products
✓ GET /products/{id}	Read a single product
X PUT /products/{id}	Update an existing product
X DELETE /products/{id}	Delete an existing product

REST API: PATCH

- Spring Data REST also exposes an endpoint to accept **PATCH** requests
- HTTP method: **PATCH** is used to perform partial updates on a resource
 - Provide the resource id in the URL path
 - List the fields to "patch"
 - Will update the "patch" fields, all other fields remain the same

Need to disable PATCH also

HTTP PATCH Example

The screenshot shows the Postman interface with the following details:

- Method:** PATCH
- URL:** http://localhost:8080/api/products/2
- Headers:** (9 items)
- Body:** (Raw JSON)

```
1 {  
2   "name" : "VB Fundamentals",  
3   "description" : "Learn Visual Basic"  
4 }
```

Annotations on the screenshot:

- A green callout points to the URL field with the text "resource id: 2".
- A green callout points to the JSON body with the text "Fields to patch".
- An orange callout at the bottom right contains the text "Need to disable PATCH also".

Possible Solutions

1. Option 1: Don't use Spring Data REST
 1. Manually create our own `@RestController`
 2. Manually define methods for access: `@GetMapping`
 3. But we lose the Spring Data REST support for paging, sorting etc :-(
2. Option 2: Use Spring Data REST  **Choose this one!**
1. Configure to disable certain HTTP methods: POST, DELETE etc...

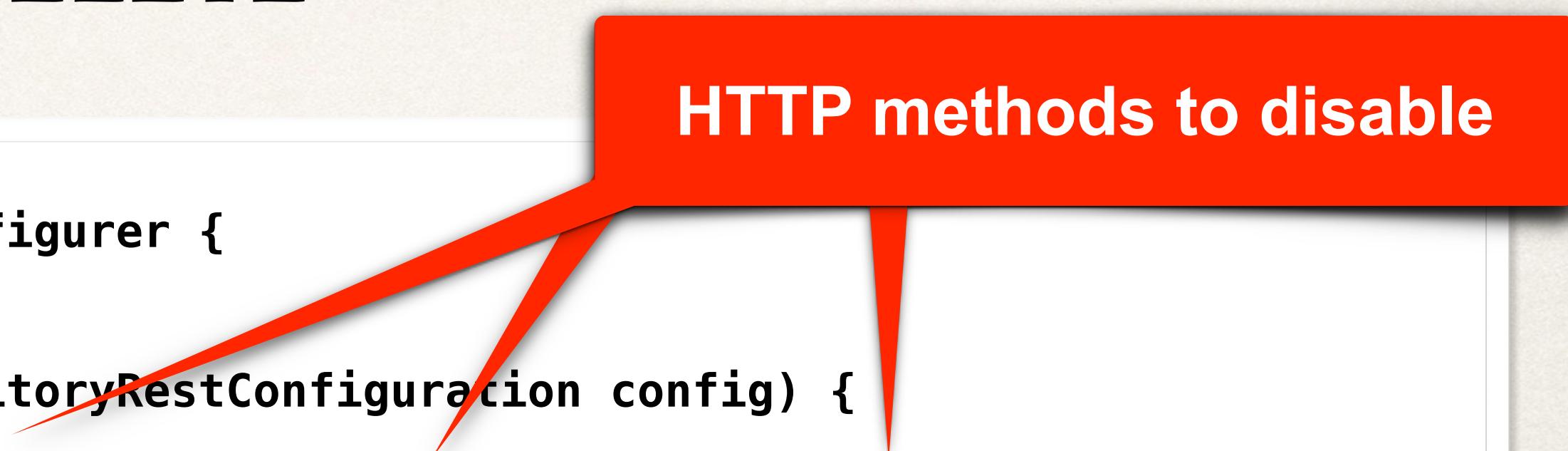
We will cover
REST API Security
later in the course



Spring Data REST Configuration

- Disable HTTP methods: POST, PUT, DELETE

```
@Configuration  
public class MyDataRestConfig implements RepositoryRestConfigurer {  
  
    @Override  
    public void configureRepositoryRestConfiguration(RepositoryRestConfiguration config) {  
  
        HttpMethod[] theUnsupportedActions = {HttpMethod.PUT, HttpMethod.POST, HttpMethod.DELETE};  
  
        // disable HTTP methods for Product: PUT, POST and DELETE  
        config.getExposureConfiguration()  
            .forDomainType(Product.class)  
            .withItemExposure((metadata, httpMethods) -> httpMethods.disable(theUnsupportedActions))  
            .withCollectionExposure((metadata, httpMethods) -> httpMethods.disable(theUnsupportedActions));  
  
    }  
}
```



HTTP methods to disable

Spring Data REST Configuration

- Disable HTTP methods: POST, PUT, DELETE

```
@Configuration  
public class MyDataRestConfig implements RepositoryRestConfigurer {  
  
    @Override  
    public void configureRepositoryRestConfiguration(RestConfiguration config) {  
  
        HttpMethod[] theUnsupportedActions = {HttpMethod.PUT, HttpMethod.POST, HttpMethod.DELETE};  
  
        // disable HTTP methods for Product: PUT, POST and DELETE  
        config.getExposureConfiguration()  
            .forDomainType(Product.class)  
            .withItemExposure((metadata, httpMethods) -> httpMethods.disable(theUnsupportedActions))  
            .withCollectionExposure((metadata, httpMethods) -> httpMethods.disable(theUnsupportedActions));  
    }  
}
```

Single item

Collection

Spring Data REST Configuration

- Disable HTTP methods: POST, PUT, DELETE

```
@Configuration
public class MyDataRestConfig implements RepositoryRestConfigurer {

    @Override
    public void configureRepositoryRestConfiguration(RepositoryRestConfiguration config) {

        HttpMethod[] theUnsupportedActions = {HttpMethod.PUT, HttpMethod.POST, HttpMethod.DELETE};

        // disable HTTP methods for Product: PUT, POST and DELETE
        config.getExposureConfiguration()
            .forDomainType(Product.class)
            .withItemExposure((metadata, httpMethods) -> httpMethods.disable(theUnsupportedActions))
            .withCollectionExposure((metadata, httpMethods) -> httpMethods.disable(theUnsupportedActions));
    }
}
```

Single item

Collection

Disable