

```
1 package com.chuwa.redbook.exception;
2
3 import org.springframework.http.HttpStatus;
4 import org.springframework.web.bind.annotation.ResponseStatus;
5
6 public class ResourceAlreadyExistsException extends RuntimeException { 4 usages
7     public ResourceAlreadyExistsException(String resource, String field, Object value) { 1 usage
8         super(String.format(
9             "%s already exists with %s = '%s'",
10            resource, field, value
11        ));
12    }
13 }
14
15 @ExceptionHandler(ResourceAlreadyExistsException.class) no usages  ↗ xx247
16 public ResponseEntity<ErrorDetails> handleResourceAlreadyExistsException(
17     ResourceAlreadyExistsException exception,
18     WebRequest webRequest) {
19
20     ErrorDetails errorDetails = new ErrorDetails(
21         new Date(),
22         exception.getMessage(),
23         webRequest.getDescription(includeClientInfo: false)
24     );
25
26     return new ResponseEntity<>(errorDetails, HttpStatus.CONFLICT);
27 }
28
29 @Override 1 usage  ↗ xx247
30 public PostDto createPost(PostDto postDto) {
31     boolean exists = postRepository.existsByTitle(postDto.getTitle());
32     if (exists) {
33         throw new ResourceAlreadyExistsException(
34             "Post",
35             "title",
36             postDto.getTitle()
37         );
38     }
39
40     // convert DTO to Entity
41     Post post = mapToEntity(postDto);
42     Post post = modelMapper.map(postDto, Post.class);
43 }
```

POST http://localhost:8080/api/v1/posts

Docs Params Authorization Headers (8) Body raw binary GraphQL JSON Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "title": "Spring Boot",
3   "description": "Duplicate title",
4   "content": "Test"
5 }
```

Body Cookies Headers (5) Test Results 409 Conflict • 16 ms • 304 B •

{ } JSON Preview Debug with AI

```
1 {
2   "timestamp": "2026-01-24T00:40:46.464+00:00",
3   "message": "Post already exists with title = 'Spring Boot'",
4   "details": "uri=/api/v1/posts"
5 }
```