

# Exception Handling in Spring REST

Basic Spring 5.0

# Lesson Objectives



- Exception handling at controller level
- Exception handling at application level





## 8.1 Exception handling at controller level

we will define a method to handle exceptions, and annotate that with `@ExceptionHandler`

```
@RestController
public class CountryController{

    @ResponseStatus(value=HttpStatus.NOT_FOUND,
reason="Country with this id not present")
    @ExceptionHandler({Exception.class})
    public void handleException() {

        }
    }
```



## 8.1 Exception handling at controller level

SpringRESTDemoWithException





## 8.2 Exception handling at application level

**Spring 3.2** brings support for a global `@ExceptionHandler` with the new `@ControllerAdvice` annotation.

```
@ControllerAdvice
public class DemoException {
    @ResponseBody
    @ResponseStatus(value=HttpStatus.NOT_FOUND)
    @ExceptionHandler(value = {Exception.class})
    protected ErrorInfo handleConflict(Exception ex,
    HttpServletRequest req) {
        String bodyOfResponse = ex.getMessage();// "Country
with this id not present";
        String uri = req.getRequestURL().toString();
        return new ErrorInfo(uri,bodyOfResponse);
    }
}
```



## 8.2 Exception handling at application level

SpringRESTDemowithGlobalException



# Summary



We have so far learnt

- Exception handling at controller level
- Exception handling at application level





Lab

Lab3







# Review Question

Question 1: Which of the following annotation is used for handler method?

- Option1 : @Exception
- Option 2: @ExceptionHandler
- Option 3: @HandlerMapping
- Option 4: @HandlerMethod

Question 2: We can handle Custom Exception in Spring REST.

- True
- False

