

Validating Spring MVC Request Mapping Method parameters

AUGUST 29, 2015

This short post demonstrates how to set up and use JSR-303 Validation for the arguments in Spring MVC request mapping methods for path variable and request parameter arguments. I am using Spring Boot v1.2.5 with Maven 3.

I. MethodValidationPostProcessor

The only configuration needed is adding the MethodValidationPostProcessor ([javadoc \(http://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/validation/beanvalidation/MethodValidationPostProcessor.html\)](http://docs.spring.io/spring/docs/current/javadoc-api/org/springframework/validation/beanvalidation/MethodValidationPostProcessor.html)) bean to the Spring configuration, e.g.

```
@Bean
public MethodValidationPostProcessor methodValidationPostProcessor()
    return new MethodValidationPostProcessor();
}
```

II Add validation to controller request mapping method

First, add the @Validate annotation to the controller class as follows:

```
@RestController
@Validated
public class HelloController {
    ...
}
```

Then add any JSR-303 validation annotation to a request mapping method arguments:

```
@RequestMapping("/hi/{name}")
public String sayHi(@Size(max = 10, min = 3, message = "name should h
    return "Hi " + name;
}
```

The example codes above shows how to validate a value in the request path marked by @PathVariable. You can do the same with @RequestParam

III Validation exception handling

A ConstraintViolationException will be thrown if the size of the path variable is not within 3 to 10 characters. You may need to catch this exception and process it using a Spring MVC exception handler, for example to return the error messages in the response:

```

@ExceptionHandler(value = { ConstraintViolationException.class })
@ResponseStatus(value = HttpStatus.BAD_REQUEST)
public String handleResourceNotFoundException(ConstraintViolationExce
    Set<ConstraintViolation<?>> violations = e.getConstraintViolatio
    StringBuilder strBuilder = new StringBuilder();
    for (ConstraintViolation<?> violation : violations ) {
        strBuilder.append(violation.getMessage() + "\n");
    }
    return strBuilder.toString();
}

```

Advertisements

[Samsung XE303C12-A01U...](#)

Samsung XE303C12-A01US
 Samsung Exynos 5250 1.7 GHz
 2 GB 16 GB SSD 11.6", Silver,...
 \$104.99
BUY NOW

[WD Red 6TB NAS Hard Di...](#)

[Samsung XE303C12-A01U...](#)

Samsung XE303C12-A01US
 Samsung Exynos 5250 1.7 GHz
 2 GB 16 GB SSD 11.6", Silver,...
 \$104.99
BUY NOW

[WD Red 4TB NAS Hard Di...](#)

FILED UNDER [J2EE](#), [JAVA](#), [SPRING](#) TAGGED WITH [@PATHVARIABLE](#), [@VALIDATE](#), [JSR-303](#),
[METHODVALIDATIONPOSTPROCESSOR](#), [SPRING MVC](#), [VALIDATION](#)

About Raymond Lee

Professional Java/EE Developer, software development technology enthusiast.

Comments are closed.

Blog at WordPress.com.