

master ▾

...

[spring-microservices](#) / [03.microservices](#) / [step08.md](#)



Ranga Rao Karanam I'm too lazy to put in a comment

History

0 contributors

## Complete Code Example

### /git-localconfig-repo/limits-service-dev.properties

```
limits-service.minimum=1
limits-service.maximum=111
```

### /git-localconfig-repo/limits-service-qa.properties

```
limits-service.minimum=2
limits-service.maximum=222
```

### /git-localconfig-repo/limits-service.properties

```
limits-service.minimum=8
limits-service.maximum=888
```

### /limits-service/pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/m
  <modelVersion>4.0.0</modelVersion>
```

```
<groupId>com.in28minutes.microservices</groupId>
<artifactId>limits-service</artifactId>
<version>0.0.1-SNAPSHOT</version>
<packaging>jar</packaging>

<name>limits-service</name>
<description>Demo project for Spring Boot</description>

<parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.0.0.RELEASE</version>
    <relativePath/> <!-- lookup parent from repository -->
</parent>

<properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
    <java.version>1.8</java.version>
    <spring-cloud.version>Finchley.M8</spring-cloud.version>
</properties>

<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-actuator</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework.cloud</groupId>
        <artifactId>spring-cloud-starter-config</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>
    </dependency>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-devtools</artifactId>
        <scope>runtime</scope>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>
        <scope>test</scope>
    </dependency>
</dependencies>

<dependencyManagement>
    <dependencies>
        <dependency>
            <groupId>org.springframework.cloud</groupId>
            <artifactId>spring-cloud-dependencies</artifactId>
            <version>${spring-cloud.version}</version>
            <type>pom</type>
```

```

        <scope>import</scope>
    </dependency>
</dependencies>
</dependencyManagement>

<build>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
    </plugins>
</build>

<repositories>
    <repository>
        <id>spring-snapshots</id>
        <name>Spring Snapshots</name>
        <url>https://repo.spring.io/snapshot</url>
        <snapshots>
            <enabled>true</enabled>
        </snapshots>
    </repository>
    <repository>
        <id>spring-milestones</id>
        <name>Spring Milestones</name>
        <url>https://repo.spring.io/milestone</url>
        <snapshots>
            <enabled>false</enabled>
        </snapshots>
    </repository>
</repositories>

<pluginRepositories>
    <pluginRepository>
        <id>spring-snapshots</id>
        <name>Spring Snapshots</name>
        <url>https://repo.spring.io/snapshot</url>
        <snapshots>
            <enabled>true</enabled>
        </snapshots>
    </pluginRepository>
    <pluginRepository>
        <id>spring-milestones</id>
        <name>Spring Milestones</name>
        <url>https://repo.spring.io/milestone</url>
        <snapshots>
            <enabled>false</enabled>
        </snapshots>
    </pluginRepository>
</pluginRepositories>

</project>

```

/limits-

service/src/main/java/com/in28minutes/microservices/limitsservice/bean/LimitConfiguration.java

```
package com.in28minutes.microservices.limitsservice.bean;

public class LimitConfiguration {
    private int maximum;
    private int minimum;

    protected LimitConfiguration() {

    }

    public LimitConfiguration(int maximum, int minimum) {
        super();
        this.maximum = maximum;
        this.minimum = minimum;
    }

    public int getMaximum() {
        return maximum;
    }

    public int getMinimum() {
        return minimum;
    }
}
```

/limits-

service/src/main/java/com/in28minutes/microservices/limitsservice/Configuration.java

```
package com.in28minutes.microservices.limitsservice;

import org.springframework.boot.context.properties.ConfigurationProperties;
import org.springframework.stereotype.Component;

@Component
@ConfigurationProperties("limits-service")
public class Configuration {

    private int minimum;
    private int maximum;

    public void setMinimum(int minimum) {
        this.minimum = minimum;
    }
}
```

```

        public void setMaximum(int maximum) {
            this.maximum = maximum;
        }

        public int getMinimum() {
            return minimum;
        }

        public int getMaximum() {
            return maximum;
        }
    }
}

```

## /limits- service/src/main/java/com/in28minutes/microservices/limitsservice/LimitsC onfigurationController.java

```

package com.in28minutes.microservices.limitsservice;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

import com.in28minutes.microservices.limitsservice.bean.LimitConfiguration;

@RestController
public class LimitsConfigurationController {

    @Autowired
    private Configuration configuration;

    @GetMapping("/limits")
    public LimitConfiguration retrieveLimitsFromConfigurations() {
        return new LimitConfiguration(configuration.getMaximum(),
            configuration.getMinimum());
    }
}

```

## /limits- service/src/main/java/com/in28minutes/microservices/limitsservice/LimitsS erviceApplication.java

```

package com.in28minutes.microservices.limitsservice;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

```

```

@SpringBootApplication
public class LimitsServiceApplication {
    public static void main(String[] args) {
        SpringApplication.run(LimitsServiceApplication.class, args);
    }
}

```

## /limits-service/src/main/resources/application.properties

```

spring.application.name=limits-service

limits-service.minimum=9
limits-service.maximum=999

```

## /limits-service/src/test/java/com/in28minutes/microservices/limitsservice/LimitsServiceApplicationTests.java

```

package com.in28minutes.microservices.limitsservice;

import org.junit.Test;
import org.junit.runner.RunWith;
import org.springframework.boot.test.context.SpringBootTest;
import org.springframework.test.context.junit4.SpringRunner;

@RunWith(SpringRunner.class)
@SpringBootTest
public class LimitsServiceApplicationTests {

    @Test
    public void contextLoads() {
    }

}

```

## /spring-cloud-config-server/pom.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/m
    <modelVersion>4.0.0</modelVersion>

    <groupId>com.in28minutes.microservices</groupId>
    <artifactId>spring-cloud-config-server</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <packaging>jar</packaging>

```

```
<name>spring-cloud-config-server</name>
<description>Demo project for Spring Boot</description>

<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>2.0.0.RELEASE</version>
  <relativePath/> <!-- lookup parent from repository -->
</parent>

<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
  <java.version>1.8</java.version>
  <spring-cloud.version>Finchley.M8</spring-cloud.version>
</properties>

<dependencies>
  <dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-config-server</artifactId>
  </dependency>

  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-devtools</artifactId>
    <scope>runtime</scope>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
  </dependency>
</dependencies>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.springframework.cloud</groupId>
      <artifactId>spring-cloud-dependencies</artifactId>
      <version>${spring-cloud.version}</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>

<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
    </plugin>
  </plugins>
</build>
```

```

<repositories>
  <repository>
    <id>spring-snapshots</id>
    <name>Spring Snapshots</name>
    <url>https://repo.spring.io/snapshot</url>
    <snapshots>
      <enabled>true</enabled>
    </snapshots>
  </repository>
  <repository>
    <id>spring-milestones</id>
    <name>Spring Milestones</name>
    <url>https://repo.spring.io/milestone</url>
    <snapshots>
      <enabled>false</enabled>
    </snapshots>
  </repository>
</repositories>

<pluginRepositories>
  <pluginRepository>
    <id>spring-snapshots</id>
    <name>Spring Snapshots</name>
    <url>https://repo.spring.io/snapshot</url>
    <snapshots>
      <enabled>true</enabled>
    </snapshots>
  </pluginRepository>
  <pluginRepository>
    <id>spring-milestones</id>
    <name>Spring Milestones</name>
    <url>https://repo.spring.io/milestone</url>
    <snapshots>
      <enabled>false</enabled>
    </snapshots>
  </pluginRepository>
</pluginRepositories>

```

```

</project>

```



/spring-cloud-config-server/src/main/java/com/in28minutes/microservices/springcloudconfigserver/SpringCloudConfigServerApplication.java

```

package com.in28minutes.microservices.springcloudconfigserver;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.cloud.config.server.EnableConfigServer;

```



```
@EnableConfigServer
@SpringBootApplication
public class SpringCloudConfigServerApplication {
```

☰ 452 lines (363 sloc) | 10.9 KB

...

```
    }
}
```

## /spring-cloud-config-server/src/main/resources/application.properties

```
spring.application.name=spring-cloud-config-server
server.port=8888
spring.cloud.config.server.git.uri=file:///in28Minutes/git/spring-micro-services/03.microse
```



## /spring-cloud-config-server/src/test/java/com/in28minutes/microservices/springcloudconfigserver/SpringCloudConfigServerApplicationTests.java

```
package com.in28minutes.microservices.springcloudconfigserver;

import org.junit.Test;
import org.junit.runner.RunWith;
import org.springframework.boot.test.context.SpringBootTest;
import org.springframework.test.context.junit4.SpringRunner;

@RunWith(SpringRunner.class)
@SpringBootTest
public class SpringCloudConfigServerApplicationTests {

    @Test
    public void contextLoads() {
    }

}
```