in28minutes / spring-microservices Public

Code Issues 3 Pull requests 4 Actions Projects Wiki Security Insi

master

spring-microservices / 03.microservices / step17.md

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

0 contributors

Complete Code Example

/currency-exchange-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/XMLSch
	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>currency-exchange-service</artifactId>
	<version>0.0.1-SNAPSHOT</version>
	<packaging>jar</packaging>

	<name>currency-exchange-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>
```

```xml
			<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-starter-data-jpa</artifactId>
		</dependency>
		<dependency>
			<groupId>com.h2database</groupId>
			<artifactId>h2</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>
```

/currency-exchange-service/src/main/java/com/in28minutes/microservices/currencyexchangeservice/CurrencyExchangeController.java

```
package com.in28minutes.microservices.currencyexchangeservice;

import java.math.BigDecimal;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.core.env.Environment;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class CurrencyExchangeController {

        @Autowired
        private Environment environment;

        @Autowired
        private ExchangeValueRepository repository;

        @GetMapping("/currency-exchange/from/{from}/to/{to}")
        public ExchangeValue retrieveExchangeValue
                (@PathVariable String from, @PathVariable String to){

                ExchangeValue exchangeValue =
                                repository.findByFromAndTo(from, to);

                exchangeValue.setPort(
                                Integer.parseInt(environment.getProperty("local.server.port

                return exchangeValue;
        }
}
```

/currency-exchange-service/src/main/java/com/in28minutes/microservices/currencyexchangeservice/CurrencyExchangeServiceApplication.java

```
package com.in28minutes.microservices.currencyexchangeservice;

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

@SpringBootApplication
public class CurrencyExchangeServiceApplication {

        public static void main(String[] args) {
                SpringApplication.run(CurrencyExchangeServiceApplication.class, args);
        }
}
```

/currency-exchange-service/src/main/java/com/in28minutes/microservices/currencyexchangeservice/ExchangeValue.java

```
package com.in28minutes.microservices.currencyexchangeservice;
```

```java
import java.math.BigDecimal;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;

@Entity
public class ExchangeValue {

        @Id
        private Long id;

        @Column(name="currency_from")
        private String from;

        @Column(name="currency_to")
        private String to;

        private BigDecimal conversionMultiple;
        private int port;

        public ExchangeValue() {

        }


        public ExchangeValue(Long id, String from, String to, BigDecimal conversionMultiple
                super();
                this.id = id;
                this.from = from;
                this.to = to;
                this.conversionMultiple = conversionMultiple;
        }

        public Long getId() {
                return id;
        }

        public String getFrom() {
                return from;
        }

        public String getTo() {
                return to;
        }

        public BigDecimal getConversionMultiple() {
                return conversionMultiple;
        }

        public int getPort() {
                return port;
        }

        public void setPort(int port) {
                this.port = port;
        }

}
```

/currency-exchange-service/src/main/java/com/in28minutes/microservices/currencyexchangeservice/ExchangeValueRepository.java

```
package com.in28minutes.microservices.currencyexchangeservice;

import org.springframework.data.jpa.repository.JpaRepository;

public interface ExchangeValueRepository extends
                JpaRepository<ExchangeValue, Long>{
        ExchangeValue findByFromAndTo(String from, String to);
}
```

/currency-exchange-service/src/main/resources/application.properties

```
spring.application.name=currency-exchange-service
server.port=8000

spring.jpa.show-sql=true
spring.h2.console.enabled=true
```

/currency-exchange-service/src/main/resources/data.sql

```
insert into exchange_value(id,currency_from,currency_to,conversion_multiple,port)
values(10001,'USD','INR',65,0);
insert into exchange_value(id,currency_from,currency_to,conversion_multiple,port)
values(10002,'EUR','INR',75,0);
insert into exchange_value(id,currency_from,currency_to,conversion_multiple,port)
values(10003,'AUD','INR',25,0);
```

/currency-exchange-service/src/test/java/com/in28minutes/microservices/currencyexchangeservice/CurrencyExchangeServiceApplicationTests.java

```
package com.in28minutes.microservices.currencyexchangeservice;
```

```java
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 CurrencyExchangeServiceApplicationTests {

        @Test
        public void contextLoads() {
        }

}
```

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

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

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

```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

```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;
```

```java
	}

	public int getMaximum() {
		return maximum;
	}

}
```

/limits-service/src/main/java/com/in28minutes/microservices/limitsservice/LimitsConfigurationController.java

```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/LimitsServiceApplication.java

```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/bootstrap.properties

```
spring.application.name=limits-service
spring.cloud.config.uri=http://localhost:8888
spring.profiles.active=qa
```

/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/XMLSch
	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>
```

```
        <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 {

        public static void main(String[] args) {
                SpringApplication.run(SpringCloudConfigServerApplication.class, args);
        }
}
```

/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

```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() {
        }

}
```