Weekly Quiz

spring

SPRING BOOT HAZELCAST EXAMPLE

Blog

Categories

Hire Team

Guest Post

About Us

Donate us

```
Table of Contents [hide]
 1. Overview
 2. Example
   2.1 pom.xml
   2.2 application.properties
   2.3 SpringBootConfig
   2.4 HazelcastConfiguration
   2.5 HazelcastController
   2.6 Demo
   Start Instance1
   2. Start instance2
   3. Write Data in instance1
   4. Read Data from instance2
 3. References
 4. Source code
   Was this post helpful?
```

Spring boot hazelcast example

Let's try to understand the requirement of hazelcast, For example, two microservices are running separately, Obviously in separate JVM. Now a requirement is like, value write by one microservice and

configuration in spring boot.

1. Overview

read by another microservice which are running independent JVM, To achieve memory sharing hazelcase will help us. Hazelcast supports Map, List, Set and many other data structures, Here we are using Map in our example.

Spring boot microservice architecture requires sharing cache between two micro-services which are

running independently so here we are explaining **Spring boot hazelcast example** and **hazelcast**

2. Example Here is an example of spring boot hazel case configuration: rpring-boot-Hazelcast-example [spring-boot-demos] idea .idea

▼ Image javadeveloperzone

▼ **Controller**

11.

12.

13.

14. 15.

16. 17.

18.

19.

20.

21.

22.

23.

24.

25.

26. 27.

28. 29.

30.

31.

32.

33.

34. 35.

36.

37.

38. 39.

40.

41. 42.

43.

44.

45.

9. 10.

16. 17.

10.

11.

12.

13.

14.

15. 16.

17.

18.

23. 24.

2.

9. 10. 11.

13.

15. 16. 17.

18.

19.

@RestController

@Autowired

<parent>

</parent>

properties>

</properties> <dependencies>

<dependency>

</dependency> <dependency>

</dependency>

</dependency>

</dependencies>

<plugins>

</plugins>

package com.javadeveloperzone;

import org.springframework.boot.SpringApplication;

* Created by JavaDeveloperZone on 19-07-2017.

import org.springframework.boot.autoconfigure.SpringBootApplication;

SpringApplication.run(SpringBootConfig.class, args);

import org.springframework.context.annotation.ComponentScan;

</plugin>

<build>

<dependency>

C 🕒 HazelcastController

src

main

▼ **i**java

▼ 🗖 com

```
► C • HazelcastConfiguration
                                       🕝 🕒 SpringBootConfig
                             resources
                                 a favicon.ico
                          target
                        m pom.xml
                        spring-boot-demos.iml
                  Spring boot hazelcast example
2.1 pom.xml
To configure hazelcast and hazelcast-spring must be required in CLASSPATH.
       <?xml version="1.0" encoding="UTF-8"?>
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   3.
               xsi:schemaLocation="http://maven.apache.org/POM/4.0.0"
   4.
      http://maven.apache.org/xsd/maven-4.0.0.xsd">
           <modelVersion>4.0.0</modelVersion>
           <groupId>spring-boot-example
           <artifactId>spring-boot-hazelcast-example</artifactId>
   8.
           <version>1.0-SNAPSHOT
   9.
   10.
           <description>Spring boot hazelcast example</description>
```

<!-- Inherit defaults from Spring Boot -->

<version>1.5.4.RELEASE

<groupId>org.springframework.boot

<artifactId>spring-boot-starter-parent</artifactId>

<maven.compiler.source>1.8</maven.compiler.source>

<maven.compiler.target>1.8</maven.compiler.target>

<groupId>org.springframework.boot

<groupId>com.hazelcast <artifactId>hazelcast</artifactId>

<groupId>com.hazelcast

<!-- Package as an executable jar -->

<artifactId>hazelcast-spring</artifactId>

<groupId>org.springframework.boot

<artifactId>spring-boot-maven-plugin</artifactId>

<artifactId>spring-boot-starter-web</artifactId>

2.2 application.properties server.port=8282

2.3 SpringBootConfig

</build>

</project>

```
without needing to specify a basePackage attribute
     public class SpringBootConfig {
         public static void main(String[] args) throws Exception {
14.
```

application

@SpringBootApplication

```
2.4 HazelcastConfiguration
Here we have created a bean for hazelcast which has configuration related to hazel cast. In bellow code
comments has been given to each line which is self-explanatory.
        package com.javadeveloperzone;
        import com.hazelcast.config.*;
        import org.springframework.context.annotation.Bean;
        import org.springframework.context.annotation.Configuration;
    5.
         * Created by JavaDeveloperZone on 08-01-2018.
        @Configuration
```

.setName("configuration")

.setMaxSizeConfig(new MaxSizeConfig(200,

.setEvictionPolicy(EvictionPolicy.LRU)

12. @ComponentScan // Using a root package also allows the @ComponentScan annotation to be used

// it wil start

// hazel case instance name

// create map

.setTimeToLiveSeconds(-1)); // cache will be available

// autowire hazel cast instance

19. 20. until it will remove manually. less then 0 means never expired. 21. return config; 22.

2.5 HazelcastController

@Bean

public class HazelcastConfiguration {

public Config hazelCastConfig() {

Config config = new Config();

.addMapConfig(

MaxSizeConfig.MaxSizePolicy.FREE HEAP SIZE))

shared between different JVM of the same hazelcast cluster.

package com.javadeveloperzone.controller;

import com.hazelcast.core.HazelcastInstance;

import org.springframework.stereotype.Controller;

* Created by JavaDeveloperZone on 19-07-2017.

private HazelcastInstance instance;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.RequestMapping; import org.springframework.web.bind.annotation.RequestParam; import org.springframework.web.bind.annotation.RestController;

import com.hazelcast.core.Hazelcast;

public class HazelcastController {

config.setInstanceName("hazelcast-instance")

new MapConfig()

The controller is doing the job for simple read and write value to hazelcast Map. This map will be

```
@RequestMapping("/write")
   20.
           public String write(@RequestParam("values")String values)
   21.
                java.util.Map<String,String> stringStringMap = instance.getMap("configuration");
       // get map from hazel cast
   23.
                stringStringMap.put("data", values);
                                                                // write value, This value will
       be accessible from another jvm also
                return "Value has been write to Hazelcast";
   24.
   25.
            @RequestMapping("/read")
   26.
            public String read() {
            java.util.Map<String,String> stringStringMap = instance.getMap("configuration");
       // get map from hazel cast
               return "Hazelcast values is :" +stringStringMap.get("data");
       read value
   30.
   31. | }
2.6 Demo
For demo, we will start the same application with the different port means two instances of same
application will be running on different where JVM data will be written by instance1 and read by
instance2.
 1. Start Instance1
java -jar spring-boot-hazelcast-example-1.0-SNAPSHOT.jar --server.port=8181
      1. 2018-01-13 10:24:34.240 INFO 134680 --- [
                                                                main]
         c.h.i.cluster.impl.MulticastJoiner : [192.168
         .0.103]:5701 [dev] [3.7.7]
```

Member [192.168.0.103]:5701 - c6f63ad1-1988-46e0-bc90-cbb534c40570 this

Member [192.168.0.103]:5701 - c6f63ad1-1988-46e0-bc90-cbb534c40570

Member [192.168.0.103]:5702 - d6e539a0-2835-4504-be66-67903d6922f0 this

2. Start instance2 java -jar spring-boot-hazelcast-example-1.0-SNAPSHOT.jar --server.port=8282

Members [1] {

2. 0.103]:5702 [dev] [3.7.7]

Value has been write using 8181 port instance

4. Members [2] {

3. Write Data in instance1

```
← → C 🗘 🛈 localhost:8181/write?values=Java%20Developer%20Zone
Value has been write to Hazelcast
```

1. 2018-01-13 10:25:49.214 INFO 59140 --- [ration.thread-0] c.h.internal.cluster.ClusterService : [192.168.

```
Spring boot hazelcast example write value
4. Read Data from instance2
Value has been read using 8282 port instance
               C ☆ O localhost:8282/read
```

NO

More

26

in LinkedIn

spring-boot-microservice-caching

JavaDeveloperZone is a group of innovative software developers.

Spring boot CSS, JS and Image

Example

Your email address will not be published. Required fields are marked *

November 4, 2017

13

We

3. References hazelcast document

Spring boot hazelcast document

4. Source code

YES

Hazelcast values is :Java Developer Zone

Spring boot hazelcast example read value

Was this post helpful? Let us know if you liked the post. That's the only way we can improve.

Spring Boot Hazelcast example (170 KB)

Share via: **Twitter** Facebook

hazelcast

Related Articles

Spring Boot View build

② November 11, 2017

Leave a Reply

Comment *

information

spring-boot

are experienced in,	Java
Software Development	Java
web development	• Big
Data development	Data
analytics	•
Artificial Intelligence Development	Our
contributions will help Java developers and make development journey easy. Feel fr	ee to ask
any questions and suggestions. Always have space for improvement! Feel free to Contact us	
for any software development services.	

Java Developer Zone

https://javadeveloperzone.com

Java 8 Documentation

Solr Documentation

Spring boot JPA Multi Tenancy

example

② September 16, 2018

Email * Website

• Hadoop & Spark Repository

Java Repository

Name *

I'm not a robot **POST COMMENT**

```
Git Repository
                                           Popular Tutorial
                                                                                       Favorites Links

    Spring boot tutorial

                                                                                       Spring MVC documentation
  Hadoop Repository

    Machine Learning Repository

    Java tutorial

                                                                                       Spring security documentation

    Solr tutorial

  Spring Repository
  • Spring Boot Microservice
                                                                                       Spring Boot Documentation
                                              • Elastic search tutorial
  • Elastic Search Repository

    Hadoop tutorial

                                                                                       Java 9 Documentation
                                              • Java 9 tutorial
  • Hadoop & Spark Repository
```

Donate VISA VISA DERECT

Your donation will help us to improve

our content, site maintenance, and

community improvement. So we like

to request that if you can donate a

small amount then will be more

valuable to us. Thank you

Contact Us

Q

• Spring Data JPA delete query • Spring data JPA where clause

Today's Popular Post

• Spring JPA query IN clause

example

- Java Convert int to byte array
- Spring JPA dynamic query example
- Spring Boot Liquibase Example • Spring cache clear all cache • Spring boot index page example
- Spring Jackson custom date format

• Java Find duplicate objects in list

Last 30 Day's Popular Post

• Spring JPA query IN clause

• Spring JPA dynamic query

example

example • Spring Data JPA delete query

- Spring Boot Liquibase Example • Java Convert int to byte array
- Spring data JPA where clause • How to run spring boot using
- Maven & Gradle • Java Find duplicate objects in list
- Spring cache clear all cache • NodeJS change server port
- **Recent Posts**
 - shaping present and future technology May 22, 2020

> Spring Boot Global Exception

> 25 years on! Here's how Java is

Difference between var, let and const in Nodejs

Handling

May 22, 2020

- NodeJs Async and Await example May 17, 2020 > Promise in Nodejs with example May 17, 2020
- May 17, 2020 > Spring bean scope

> Spring Declarative Transaction

Management

March 27, 2020

- May 17, 2020

May 17, 2020

- > Nodejs Stream Example May 17, 2020 > Function callback in Nodejs May 17, 2020 > Different ways to clone or copy an object in NodeJs