



intomylife

码龄3年

暂无认证

112

1万+

9174

91万+



原创

周排名

总排名

访问

等级

6440

513

555

148

1495

积分

粉丝

获赞

评论

收藏

私信

关注

搜博文文章

博主相关

个人主页:

https://www.zouwenc

GitHub:

https://github.com/intomylife

Docker Hub:

://hub.docker.com/u/intomylife

热门文章

SpringCloud之服务网关Gateway

190872

在vue中使用rules对表单字段进行验证

54235

SpringCloud之熔断器Hystrix

53154

SpringBoot整合MongoDB，使用mongoTemplate实现高级查询

46671

设置type='date'的input标签不能选择大于当前时间的日期

29822

分类专栏

 初学者商城

24篇

 SpringBoot

10篇

 SpringCloud

12篇

 Vue

12篇

 Docker

13篇

 CentOS7

15篇

最新评论

使用mongoTemplate实现多条件加分组查询

publicman_: 感谢分享

SpringCloudAlibaba之配置中心Nacos

wen972368: config-order-service明明配了3个，为什么会多出两个；还有权重到底...

我的「SpringCloud」专栏总篇

偷吃布丁的恶魔: 已三连，感谢博主分享

SpringCloud之服务网关Gateway

Tisfy: 这让我想起了先贤的一句话：满目山河增感慨，一时风景寄遨游。

SpringCloud之熔断器Hystrix

TS六道轮回: 好

最新文章

CentOS7中使用docker-compose部署Nacos集群

SpringCloudAlibaba之配置中心Nacos

SpringCloudAlibaba之服务注册与发现Nacos

2020年 24篇

2019年 62篇

2018年 26篇

目录

前言

源码

环境

开发工具

正文

commons 工程 - POM 文件

commons 工程 - system.properties

commons 工程 - 项目结构

service 工程

Provider（提供者）

service 工程 - base-service - base-s...

service 工程 - base-service - 启动项目

Consumer（消费者）

Kafka

SpringBoot整合Kafka

原创intomylife2019-04-21 16:26:2812470收藏8

分类专栏:SpringBoot文章标签:SpringBootKafka

前言

kafka 也作为消息中间件的一员，与其他消息中间件相比，它的优点在于拥有极高的吞吐量，ms 级的延迟，是一个高性能，分布式的系统。

源码

GitHub地址：https://github.com/intomylife/SpringBoot

环境

- JDK 1.8.0 +
- Maven 3.0 +
- SpringBoot 2.0.3
- ZooKeeper-3.4.5
- kafka_2.12-2.2.0

开发工具

- IntelliJ IDEA

正文

commons 工程 - POM 文件

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/
3     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.or
4     <modelVersion>4.0.0</modelVersion>
5
6     <!-- 三坐标 -->
7     <groupId>com.zwc</groupId>
8     <artifactId>springboot-kafka-commons</artifactId>
9     <version>0.0.1-SNAPSHOT</version>
10
11     <!-- 工程名称和描述 -->
12     <name>springboot-kafka-commons</name>
13     <description>公用工程</description>
14
15     <!-- 打包方式 -->
16     <packaging>jar</packaging>
17
18     <!-- 在properties下声明相应的版本信息，然后在dependency下引用的时候用${spring-version}就可
19     <properties>
20         <!-- 编码 -->
21         <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
22         <!-- jdk -->
23         <java.version>1.8</java.version>
24
25         <!-- springboot -->
26         <platform-bom.version>Cairo-SR3</platform-bom.version>
27
28         <!-- ali json -->
29         <fastjson.version>1.2.47</fastjson.version>
30         <jackson.mapper.asl.version>1.9.9</jackson.mapper.asl.version>
31     </properties>
32
33     <!-- 加入依赖 -->
34     <dependencies>
35         <!-- ali json依赖 -->
36         <dependency>
37             <groupId>com.alibaba</groupId>
38             <artifactId>fastjson</artifactId>
39             <version>${fastjson.version}</version>
40         </dependency>
41         <dependency>
42             <groupId>org.codehaus.jackson</groupId>
43             <artifactId>jackson-mapper-asl</artifactId>
44             <version>${jackson.mapper.asl.version}</version>
45         </dependency>
46
47         <!-- kafka 依赖 -->
48         <dependency>
49             <groupId>org.springframework.kafka</groupId>
50             <artifactId>spring-kafka</artifactId>
51         </dependency>
52     </dependencies>
53
54     <!-- 依赖 jar 包版本管理的管理器 -->
55     <!-- 如果 dependencies 里的 dependency 自己没有声明 version 元素，那么 maven 就此处来找
56     <!-- 如果有，就会继承它；如果没有就会报错，告诉你没有版本信息 -->
57     <!-- 优先级：如果 dependencies 里的 dependency 已经声明了版本信息，就不会生效此处的版本信息
58     <dependencyManagement>
59         <dependencies>
60             <!-- SpringBoot -->
61             <dependency>
62                 <groupId>io.spring.platform</groupId>
63                 <artifactId>platform-bom</artifactId>
64                 <version>${platform-bom.version}</version>
65                 <type>pom</type>
66                 <scope>import</scope>
67             </dependency>
68         </dependencies>
69     </dependencyManagement>
70
71     <!-- 插件依赖 -->
72     <build>
73         <plugins>
74             <plugin>
75                 <groupId>org.springframework.boot</groupId>
76                 <artifactId>spring-boot-maven-plugin</artifactId>
77             </plugin>
78         </plugins>
79     </build>
80
81
82 </project>
```

- 配置一些共用依赖，其中包括 spring-kafka 依赖来整合 Kafka

commons 工程 - system.properties

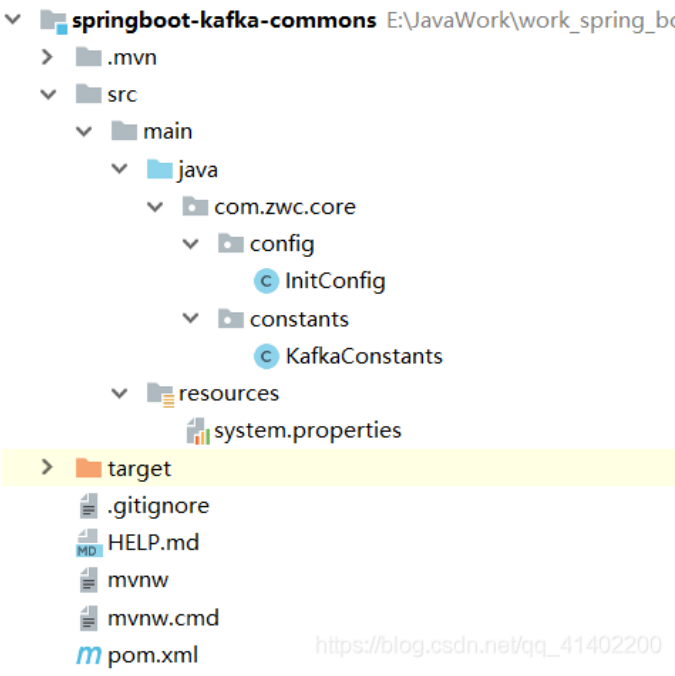
```
1 # kafka 配置
2
3 ## kafka 服务地址
4 spring.kafka.bootstrap-servers=127.0.0.1:9092
5
6 ## producer 提供者
7 ### 如果该值大于零时，表示启用重试失败的发送次数
8 spring.kafka.producer.retries=0
9 ### 每次批量发送消息的数量
10 spring.kafka.producer.batch-size=16384
```



```
13 | spring.kafka.producer.key-serializer=org.apache.kafka.common.serialization.StringSeri
14 | spring.kafka.producer.value-serializer=org.apache.kafka.common.serialization.StringSe
15 |
16 | ## consumer 消费者
17 | ### 指定默认消费者group id
18 | spring.kafka.consumer.group-id=springboot-consumer-group
19 | ### 当Kafka中没有初始偏移量或者服务器上不再存在当前偏移量时该怎么办，默认值为latest，表示自动将偏移
20 | spring.kafka.consumer.auto-offset-reset=earliest
21 | ### 如果为true，则消费者的偏移量将在后台定期提交，默认值为true
22 | spring.kafka.consumer.enable-auto-commit=false
23 | ### 如果 'enable.auto.commit' 为true，则消费者偏移自动提交给Kafka的频率（以毫秒为单位），默认值为5
24 | spring.kafka.consumer.auto-commit-interval=100
25 | ### 指定消息key和消息体的编解码方式
26 | spring.kafka.consumer.key-deserializer=org.apache.kafka.common.serialization.StringDe
27 | spring.kafka.consumer.value-deserializer=org.apache.kafka.common.serialization.String
```

- 一些共用配置，不经常修改的，或者是可以统一修改的
- 比如还可以配置 OSS 的配置信息，Redis 的配置信息，MongoDB 的配置信息等等.

commons 工程 - 项目结构



service 工程

service 工程是一个父工程，里面包含 基础模块，用户模块，每个模块中又会分为 core 和 api

此工程中 base-service 作为 Provider（提供者），user-service 作为 Consumer（消费者）

Provider（提供者）

service 工程 - base-service - base-service-core

```
1 | package com.zwc.base.kafka;
2 |
3 | import com.zwc.core.constants.KafkaConstants;
4 | import org.springframework.beans.factory.annotation.Autowired;
5 | import org.springframework.kafka.core.KafkaTemplate;
6 | import org.springframework.stereotype.Service;
7 |
8 | /**
9 |  * @ClassName KafkaProducer
10 |  * @Desc TODO Kafka 提供者
11 |  * @Date 2019/4/20 19:41
12 |  * @Version 1.0
13 |  */
14 | @Service
15 | public class KafkaProducer {
16 |
17 |     @Autowired
18 |     private KafkaTemplate kafkaTemplate;
19 |
20 |     /*
21 |      * @ClassName KafkaProducer
22 |      * @Desc TODO 发送订阅者消息
23 |      * @Date 2019/4/20 19:46
24 |      * @Version 1.0
25 |      */
26 |     public void sendTopicMessage(){
27 |         kafkaTemplate.send(KafkaConstants.KAFKA_TOPIC_NAME,"EN: From springboot-kafka
28 |     }
29 |
30 | }
```

- 直接使用 @Autowired 注解把 KafkaTemplate 装配进来
- 使用 kafkaTemplate 对象的 send() 方法发送消息
- 记得使用 @Service 注解把此 Bean 交给 spring 管理

service 工程 - base-service - 启动项目

注：项目启动前需要启动 zkServer 和 kafka（在文末处会单独说一下 windows 环境下安装启动 kafka 主要

事项）

1. 端口：8081（具体可以根据自己的喜好，在 application.properties 配置文件中配置 server.port）
2. 发送消息接口：<http://localhost:8081/kafka/send>（调用成功会在页面显示 success）

Consumer（消费者）

service 工程 - user-service - user-service-core

消费者一号

```
1 | package com.zwc.user.kafka;
2 |
3 | import com.zwc.core.constants.KafkaConstants;
4 | import org.springframework.kafka.annotation.KafkaListener;
5 | import org.springframework.stereotype.Service;
6 |
7 | /**
8 |  * @ClassName KafkaConsumer
9 |  * @Desc TODO Kafka 消费者 - 一号
10 |  * @Date 2019/4/20 19:47
11 |  * @Version 1.0
12 |  */
13 | @Service
14 | public class KafkaConsumerOne {
15 |
16 |     /*
17 |      * @ClassName KafkaConsumer
18 |      * @Desc TODO 接收订阅者消息
19 |      * @Date 2019/4/20 19:50
20 |      * @Version 1.0
21 |      */
22 |     @KafkaListener(topics = KafkaConstants.KAFKA_TOPIC_NAME , groupId = KafkaConstant
23 |     public void receiveTopicMessage(String message){
24 |         System.out.println("KafkaConsumerOne ----> receiveTopicMessage: 接收订阅者模式发送
25 |     }
26 |
27 | }
```

- 使用 @KafkaListener 注解开始监听消息任务



命令语法不正确。
错误：找不到或无法加载主类 kafka.Kafka

d) 后来发现下载错了压缩包，我点击下载的是

2.2.0

Released Mar 22, 2019

Release Notes

Source download: kafka-2.2.0-src.tgz (asc, sha512)

Binary downloads:

Scala 2.11 - kafka-2.11-2.2.0.tgz (asc, sha512)

Scala 2.12 - kafka-2.12-2.2.0.tgz (asc, sha512)

We build for multiple versions of Scala. This only matters if you are using Scala and you want a version built for the same Scala version you use. Otherwise any version should work (2.12 is recommended).

e) 正确的下载入口是

2.2.0

Released Mar 22, 2019

Release Notes

Source download: kafka-2.2.0-src.tgz (asc, sha512)

Binary downloads:

Scala 2.11 - kafka-2.11-2.2.0.tgz (asc, sha512)

Scala 2.12 - kafka-2.12-2.2.0.tgz (asc, sha512)

We build for multiple versions of Scala. This only matters if you are using Scala and you want a version built for the same Scala version you use. Otherwise any version should work (2.12 is recommended).

f) 点击 kafka_2.12-2.2.0.tgz 后跳转到一个新的页面，点击最上面的建议下载地址就可以了

We suggest the following mirror site for your download:

http://mirrors.tuna.tsinghua.edu.cn/apache/kafka/2.2.0/kafka_2.12-2.2.0.tgz

Other mirror sites are suggested below.

It is essential that you verify the integrity of the downloaded file using the PGP signature (.asc file) or a hash (GPG or SHA* file).

Please only use the backup mirrors to download KEYS, PGP and MD5 sigs/shashes or if no other mirrors are working.

注：路径中不要有空格！！

2. 启动

a) 前提：jdk 版本最低为 1.8.0，并且要配置到环境变量中；必须先启动 zkServer

b) 有的版本需要如下改动，打开 kafka_2.12-2.2.0\bin\windows 文件夹，使用文本编辑器打开

kafka-run-class.bat, 把 179 行中 %CLASSPATH% 使用双引号包起来

```
179 set CLASSPATH=%CLASSPATH% "%KAFKA_HOME%\bin\windows\kafka-run-class.bat"
180 set KAFKA_HOME=%KAFKA_HOME%
181 set KAFKA_VERSION=%KAFKA_VERSION%
182 set KAFKA_HOME=%KAFKA_HOME%
183 set KAFKA_HOME=%KAFKA_HOME%
```

c) 假如把 kafka_2.12-2.2.0 文件解压到了 D 盘根目录

- ① 在 kafka_2.12-2.2.0 文件夹里新建 kafka-logs 文件夹
- ② 打开 kafka_2.12-2.2.0\config 文件夹，
- ③ 使用文本编辑器打开 server.properties，把 60 行中日志文件路径修改一如下：

D:\kafka_2.12-2.2.0\kafka-logs

```
56 ##### Log Basics #####
57
58 # A comma separated list of directories under which to store log files
59 log.dirs=D:\kafka_2.12-2.2.0\kafka-logs
60
61
```

d) 打开 cmd，进入到 kafka_2.12-2.2.0 文件夹下 bin\windows 目录

使用指令 kafka-server-start.bat ..\..\config\server.properties 启动 kafka

使用指定 kafka-topics.bat --list --zookeeper localhost:2181 查看消息列表

结语

到此 SpringBoot 整合 Kafka 就结束了，注意几个关键的点，多多尝试，一定会成功的!

希望能够帮助到你

over

SpringBoot整合kafka04-22

SpringBoot整合kafka代码，两个工程 一个消费者 一个生产者，利用定时任务。很简洁代码。欢迎大家下载哦。可参考这篇...

Spring Boot + Kafka的使用Black1499的博客 2万+

一、快速了解Kafka 在把Kafka集成到spring之前，我们首先要了解Kafka是什么？由什么东西组成？主要的使用场景是哪些...

优质评论可以帮助作者获得更高权重

抢沙发

评论

相关推荐

- SpringBoot集成kafka全面实战_Felix4-1
- 本文是SpringBoot+Kafka的实战讲解,如果对kafka的架构原理还不了解的读者,建议先看一下《大白话kafka架构原理》、《...
- spring boot整合kafka+注解方式_嘎嘎的博客4-2
- spring boot自动配置方式整合 spring boot具有许多自动化配置,对于kafka的自动化配置当然也包含在内,基于spring boot自动...
- SpringBoot和Kafka整合贾红平 2万+
- 今天简单通过代码演示一下,如何使用springboot来整合kafka或者RabbitMQ,其实非常简单,直接使用别人已经封装好的组件...
- SpringBoot2.0之整合Kafkaart_code的博客 1683
- SpringBoot2.0之整合Kafka maven依赖: <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3....
- springboot2.x整合kafka【详细】_喵王叭的博客3-29
- kafka-zookeeper对应版本(最新可参考官网:http://kafka.apache.org/downloads): springboot-kafka客户端对应版本(https://sp...
- Springboot1.5和kafka的整合lichuangcsdn的博客 3431
- 详细参考http://blog.csdn.net/gebitan505/article/details/702420161、引入依赖（具体spring-kafka的版本由spring boot的当...
- SpringBoot集成kafka全面实战Felix 4万+
- 本文是SpringBoot+Kafka的实战讲解，如果对kafka的架构原理还不了解的读者，建议先看一下《大白话kafka架构原理》、...
- Springboot集成kafka 简单易懂weixin_43914685的博客 778
- kafka集群环境搭建地址见之前博客: https://blog.csdn.net/weixin_43914685/article/details/113802894 开始springboot项目...
- SpringBoot Kafka 整合使用http://www.54tianzhisheng.cn/ 5万+
- 前提 假设你了解过 SpringBoot 和 Kafka。 1、SpringBoot 如果对 SpringBoot 不了解的话，建议去看看 DD 大佬 和 纯洁的...
- springboot+kafka整合(环境: IntelliJ IDEA)ZJE 4212
- (一) kafka producer: 生产者，负责发布消息到kafka cluster(kafka集群)中。生产者可以是web前端产生的page view，或...
- SpringBoot (十二)： SpringBoot整合KafkaSaytime 3万+
- 一、准备工作 提前说明: 如果你运行出问题，请检查Kafka的版本与SpringBoot的版本是否与我文中的一致，本文中的环境...
- springboot2整合oauth2qq_37170583的博客 4万+
- 1.背景 项目由springboot1.5.X升级到springboot2.0.0后，导致各组件API以及依赖包发生了变化。完整项目demo: https://g...
- springboot2整合kafkaqq_37170583的博客 9224
- 一、下载和启动kafka 启动kafka服务之前要先启动zookeeper，目前kafka的压缩包中是包含zookeeper的 1、从官网下载kaf...
- spring 集成kafka消费者zpzkit的专栏 2345
- 本文主要是说明spring集成kafka的消费者功能和过程中遇到的各种小问题。 环境: spring-kafka-2.1.11.RELEASE; spring-...
- springboot整合kafka实现logback收集日志DeanKano的博客 45万+
- springboot中实现logback收集日志输出到kafka 异常容错机制,如果kafka服务宕机,输出到本地文件,可用其他方式重新加载lo...
- _42739473的博客 3039

1.kafka的简单介绍 Apache **kafka**官网上对**kafka**的简单别成为 a distributed streaming platform 即一个分布式流平台 主要有...

springboot整合kafka congge 4万+
业务中需要使用到异步消息队列，为了快速搭建一个消息中间件，这里选了**kafka**，一方面是**kafka**搭建比较简单，而且这个...

SpringBoot 整合 Kafka zhanghe_zhi的博客 87
pom.xml依赖: <dependency> <groupId>org.springframework.**kafka**</groupId> <artifactId>spring-**kafka**</artifactId> </depe...

Springboot整合Kafka消息队列 普通还不自信的程序猿 33
启动**kafka** 1. 启动zookeeper 2. 启动**kafka** 打开黑窗口，输入 .bin\windows**kafka**-server-start.bat .\config\server.properties ...

Spring 整合 kafka qq_41112238的博客 127
在 **kafka** 官网下载，然后解压缩。 首先配置 zookeeper，打开 config 里面的 zookeeper.properties 配置数据存放路径，自...

