#### Spring Boot

##### 之JMS消息模型

问题描述

Spring Boot 2.2.5

JDK 8

Java内的消息传递有两种实现方案：

* JVM内部消息：使用JMS消息模型
* 外部消息：使用MQ消息队列（Kafka、ActiveMQ等）

日出东方，唯我不败。Spring Framework为我们提供了Java Web一站式解决方案。其中就包括通过注解封装的JMS消息模型

解决

1. 定义消息体

@Getter

@Setter

@Accessors(chain = true)

public class GenericAnimalEvent extends ApplicationEvent {

private String title;

private String content;

public GenericAnimalEvent(Object source) {

super(source);

}

}

1. 实现、注册消息监听器（Listener）
   1. 实现监听器

public class FeedAnimalListener {

@EventListener

public void handleEvent(GenericAnimalEvent genericAnimalEvent) {

System.out.println(genericAnimalEvent.getSource());

System.out.println("Feed animal done.");

}

}

* 1. 注册监听器

@Configuration

@ComponentScan

public class ApplicationBeanConfig {

@Bean

public FeedAnimalListener getFeedAnimalListener() {

return new FeedAnimalListener();

}

}

1. 发布消息

@Service

public class AnimalQueryService implements ApplicationEventPublisherAware {

private ApplicationEventPublisher applicationEventPublisher;

public void sendMessage() {

this.applicationEventPublisher.publishEvent(

new GenericAnimalEvent(this)

.setContent("It s time to feed your animal.")

.setTitle("Tip")

);

}

@Override

public void setApplicationEventPublisher(ApplicationEventPublisher applicationEventPublisher) {

this.applicationEventPublisher = applicationEventPublisher;

}

}

笔记

1. 消息体中的字段使用final关键字定义更佳
2. 使用注解@EventListener声明消息监听器（Listener）
3. 实现ApplicationEventPublisherAware接口（隶属Aware接口组）发布消息
4. Spring5使用全注解配置
   1. applicationContext.xml替换为@Bean注解
   2. Java Bean声明、初始化更为灵活

□