SpringBoot集成RabbitMQ 实现WebStomp接受消息

### 一．用MQ发送消息

## 1:pom.xml

<dependency>

<groupId>org.springframework.boot</groupId>

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

</dependency>

2:application.properties

server.port=8000

#服务器配置

spring.application.name=rabbitmq-hello-sending

#rabbitmq连接参数

spring.rabbitmq.host=localhost

spring.rabbitmq.port=5672

spring.rabbitmq.username=root

spring.rabbitmq.password=ilovelaohan

3:配置消息队列

@Configuration

public class RabbitConfig {

@Bean

public Queue QueueA() {

return new Queue("hello");

}

}

4：发送消息

@RestController

public class SendController {

public int count = 0;

@Resource

private AmqpTemplate amqpTemplate;

@RequestMapping("/send")

public String send() {

count++;

String message = "message is "+ new Date();

amqpTemplate.convertAndSend("hello",message);

System.out.println("发送=================>"+count);

return message;

}

}

5：Java接受消息

@Component

@RabbitListener(queues="hello")

public class Reveice {

public int count = 0;

@RabbitHandler

public void receive(String msg) {

count++;

System.out.println("接受=================>"+count);

System.out.println("您有新的短消息"+msg);

}

}

6：Stomp接受消息

<!DOCTYPE HTML>

<html>

<head>

<title>My WebSocket</title>

</head>

<body>

Welcome<br/>

<input id="text" type="text" />

<div id="message">

</div>

</body>

<script src="https://cdn.bootcss.com/stomp.js/2.3.3/stomp.js"></script>

<script src="https://cdn.bootcss.com/sockjs-client/1.1.4/sockjs.js"></script>

<script type="text/javascript">

if (typeof WebSocket == 'undefined') {

console.log('不支持websocket')

}

// 初始化 ws 对象

var ws = new WebSocket('ws://127.0.0.1:15674/ws');

// 获得Stomp client对象

var client = Stomp.over(ws);

// SockJS does not support heart-beat: disable heart-beats

//client.heartbeat.outgoing = 0;

//client.heartbeat.incoming = 0;

//client.debug = pipe('#second');

// 定义连接成功回调函数

var on\_connect = function(x) {

//data.body是接收到的数据

client.subscribe("/queue/hello", function(data) {

var msg = data.body;

alert("收到数据：" + msg);

});

};

// 定义错误时回调函数

var on\_error = function() {

console.log('error');

};

// 连接RabbitMQ

client.connect('guest', 'guest', on\_connect, on\_error, '/');

console.log(">>>连接上http://localhost:15674");

</script>

</html>