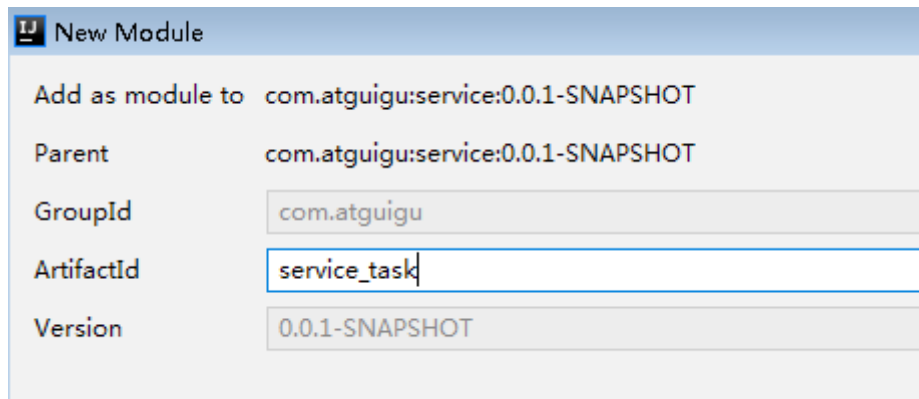


# 一、就医提醒

我们通过定时任务，每天8点执行，提醒就诊

## 1、搭建定时任务模块



Add as module to	com.atguigu:service:0.0.1-SNAPSHOT
Parent	com.atguigu:service:0.0.1-SNAPSHOT
GroupId	com.atguigu
ArtifactId	service_task
Version	0.0.1-SNAPSHOT

### (1) 添加依赖

```
1 <dependencies>
2     <dependency>
3         <groupId>com.atguigu</groupId>
4         <artifactId>rabbit_util</artifactId>
5         <version>0.0.1-SNAPSHOT</version>
6     </dependency>
7 </dependencies>
```

### (2) 添加配置文件

```
1 # 服务端口
2 server.port=8208
3 # 服务名
4 spring.application.name=service-task
5 # 环境设置: dev、test、prod
6 spring.profiles.active=dev
7
8 # nacos服务地址
```

```
9 spring.cloud.nacos.discovery.server-addr=127.0.0.1:8848
10
11 #rabbitmq地址
12 spring.rabbitmq.host=192.168.44.165
13 spring.rabbitmq.port=5672
14 spring.rabbitmq.username=guest
15 spring.rabbitmq.password=guest
```

### (3) 添加启动类

```
1 @SpringBootApplication(exclude = DataSourceAutoConfiguration.class)
2 @EnableDiscoveryClient
3 public class ServiceTaskApplication {
4     public static void main(String[] args) {
5         SpringApplication.run(ServiceTaskApplication.class, args);
6     }
7 }
```

### (4) 添加常量配置

在rabbit-util模块MqConst类添加

```
1 //定时任务
2 public static final String EXCHANGE_DIRECT_TASK = "exchange.direct.task";
3 public static final String ROUTING_TASK_8 = "task.8";
4 //队列
5 public static final String QUEUE_TASK_8 = "queue.task.8";
```

### (5) 添加定时任务

cron表达式

秒
分钟
**小时**
日
月
周
年

☐ 小时 允许的通配符[, - \* /]

☐ 周期从 0 - 2 小时

☐ 从 0 小时开始,每 1 小时执行一次

☒ 指定

AM:
☐ 00
☐ 01
☐ 02
☐ 03
☐ 04
☐ 05
☐ 06
☐ 07
☒ 08
☐ 09
☐ 10
☐ 11

PM:
☐ 12
☐ 13
☐ 14
☐ 15
☐ 16
☐ 17
☐ 18
☐ 19
☐ 20
☐ 21
☐ 22
☐ 23

表达式

	秒	分钟	小时	日	月
表达式字段:	0	0	8	*	*
Cron 表达式:	0 0 8 * * ?				

```

1 @Component
2 @EnableScheduling
3 public class ScheduledTask {
4
5     @Autowired
6     private RabbitService rabbitService;
7
8     /**
9      * 每天8点执行 提醒就诊
10     */
11     //@Scheduled(cron = "0 0 8 * * ?")
12     @Scheduled(cron = "0/50 * * * * ?")
13     public void task1() {
14         System.out.println(new Date().toLocaleString());
15         rabbitService.sendMessage(MqConst.EXCHANGE_DIRECT_TASK, MqConst.ROUTING_TASK_
16     }
17 }

```

## 2、添加就医提醒处理

操作模块service-order

### (1) 添加OrderService接口和实现

```

1  /**
2   * 就诊提醒
3   */
4  void patientTips();
5
6  //实现方法
7  @Override
8  public void patientTips() {
9      QueryWrapper<OrderInfo> queryWrapper = new QueryWrapper<>();
10     queryWrapper.eq("reserve_date", new DateTime().toString("yyyy-MM-dd"));
11     List<OrderInfo> orderInfoList = baseMapper.selectList(queryWrapper);
12     for(OrderInfo orderInfo : orderInfoList) {
13         //短信提示
14         MsmVo msmVo = new MsmVo();
15         msmVo.setPhone(orderInfo.getPatientPhone());
16         String reserveDate = new DateTime(orderInfo.getReserveDate()).toString("yyyy-
17         Map<String,Object> param = new HashMap<String,Object>(){
18             put("title", orderInfo.getHosname()+"|"+orderInfo.getDepname()+"|"+orderI
19             put("reserveDate", reserveDate);
20             put("name", orderInfo.getPatientName());
21         });
22         msmVo.setParam(param);
23         rabbitService.sendMessage(MqConst.EXCHANGE_DIRECT_MSM, MqConst.ROUTING_MSM_IT
24     }
25 }

```

## (2) 添加mq监听

```

1  @Component
2  public class OrderReceiver {
3
4      @Autowired
5      private OrderService orderService;
6
7      @RabbitListener(bindings = @QueueBinding(
8          value = @Queue(value = MqConst.QUEUE_TASK_8, durable = "true"),
9          exchange = @Exchange(value = MqConst.EXCHANGE_DIRECT_TASK),
10         key = {MqConst.ROUTING_TASK_8}
11     ))
12     public void patientTips(Message message, Channel channel) throws IOException {

```

```
13         orderService.patientTips();  
14     }  
15 }
```