

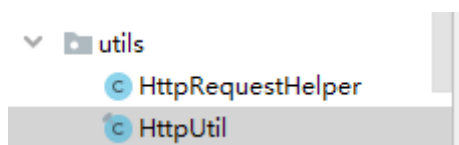
# 一、预约下单

## 1、实现生成订单接口

### (1) service\_orders引入依赖

```
1 <dependencies>
2     <dependency>
3         <groupId>com.atguigu</groupId>
4         <artifactId>service_cmn_client</artifactId>
5         <version>0.0.1-SNAPSHOT</version>
6     </dependency>
7
8     <dependency>
9         <groupId>com.atguigu</groupId>
10        <artifactId>service_hosp_client</artifactId>
11        <version>0.0.1-SNAPSHOT</version>
12    </dependency>
13
14    <dependency>
15        <groupId>com.atguigu</groupId>
16        <artifactId>service_user_client</artifactId>
17        <version>0.0.1-SNAPSHOT</version>
18    </dependency>
19 </dependencies>
```

### (2) 封装工具类



### (3) 实现生成订单

```
1 @Autowired
2 private PatientFeignClient patientFeignClient;
```

```
3
4 @Autowired
5 private HospitalFeignClient hospitalFeignClient;
6
7 //创建订单
8 @Override
9 public Long saveOrder(String scheduleId, Long patientId) {
10     //1 根据scheduleId获取排班数据
11     ScheduleOrderVo scheduleOrderVo = hospitalFeignClient.getScheduleOrderVo(
12
13     //2 根据patientId获取就诊人信息
14     Patient patient = patientFeignClient.getPatientOrder(patientId);
15
16     //3 平台里面 ==> 调用医院订单确认接口,
17     // 3.1 如果医院返回失败, 挂号失败
18     //使用map集合封装需要传过医院数据
19     Map<String, Object> paramMap = new HashMap<>();
20     paramMap.put("hoscode", scheduleOrderVo.getHoscode());
21     paramMap.put("depcode", scheduleOrderVo.getDepcode());
22     paramMap.put("hosScheduleId", scheduleOrderVo.getHosScheduleId());
23     paramMap.put("reserveDate", new DateTime(scheduleOrderVo.getReserveDate()));
24     paramMap.put("reserveTime", scheduleOrderVo.getReserveTime());
25     paramMap.put("amount", scheduleOrderVo.getAmount()); //挂号费用
26     paramMap.put("name", patient.getName());
27     paramMap.put("certificatesType", patient.getCertificatesType());
28     paramMap.put("certificatesNo", patient.getCertificatesNo());
29     paramMap.put("sex", patient.getSex());
30     paramMap.put("birthdate", patient.getBirthdate());
31     paramMap.put("phone", patient.getPhone());
32     paramMap.put("isMarry", patient.getIsMarry());
33     paramMap.put("provinceCode", patient.getProvinceCode());
34     paramMap.put("cityCode", patient.getCityCode());
35     paramMap.put("districtCode", patient.getDistrictCode());
36     paramMap.put("address", patient.getAddress());
37     //联系人
38     paramMap.put("contactsName", patient.getContactsName());
39     paramMap.put("contactsCertificatesType", patient.getContactsCertificatesType());
40     paramMap.put("contactsCertificatesNo", patient.getContactsCertificatesNo());
41     paramMap.put("contactsPhone", patient.getContactsPhone());
42     paramMap.put("timestamp", HttpRequestHelper.getTimestamp());
43     //String sign = HttpRequestHelper.getSign(paramMap, signInfoVo.getSignKey());
44     paramMap.put("sign", "");
45 }
```

```
46 //使用httpClient发送请求，请求医院接口
47 JSONObject result =
48     HttpRequestHelper.sendRequest(paramMap, "http://localhost:9998/order/
49 //根据医院接口返回状态码判断 200 成功
50 if(result.getInteger("code") == 200) { //挂号成功
51     // 3.2 如果返回成功，得到返回其他数据
52     JSONObject jsonObject = result.getJSONObject("data");
53     //预约记录唯一标识（医院预约记录主键）
54     String hosRecordId = jsonObject.getString("hosRecordId");
55     //预约序号
56     Integer number = jsonObject.getInteger("number");
57     //取号时间
58     String fetchTime = jsonObject.getString("fetchTime");
59     //取号地址
60     String fetchAddress = jsonObject.getString("fetchAddress");
61
62     //4 如果医院接口返回成功，添加上面三部分数据到数据库
63     OrderInfo orderInfo = new OrderInfo();
64     //设置添加数据--排班数据
65     BeanUtils.copyProperties(scheduleOrderVo, orderInfo);
66     //设置添加数据--就诊人数据
67     //订单号
68     String outTradeNo = System.currentTimeMillis() + "" + new Random().next
69     orderInfo.setOutTradeNo(outTradeNo);
70     orderInfo.setScheduleId(scheduleId);
71     orderInfo.setUserId(patient.getUserId());
72     orderInfo.setPatientId(patientId);
73     orderInfo.setPatientName(patient.getName());
74     orderInfo.setPatientPhone(patient.getPhone());
75     orderInfo.setOrderStatus(OrderStatusEnum.UNPAID.getStatus());
76
77     //设置添加数据--医院接口返回数据
78     orderInfo.setHosRecordId(hosRecordId);
79     orderInfo.setNumber(number);
80     orderInfo.setFetchTime(fetchTime);
81     orderInfo.setFetchAddress(fetchAddress);
82
83     //调用方法添加
84     baseMapper.insert(orderInfo);
85
86     //TODO 5 根据医院返回数据，更新排班数量
87     //排班可预约数
88     Integer reservedNumber = jsonObject.getInteger("reservedNumber");
```

```
89      //排班剩余预约数
90      Integer availableNumber = jsonObject.getInteger("availableNumber");
91
92      //TODO 6 给就诊人发送短信
93
94      //7 返回订单号
95      return orderInfo.getId();
96  } else { //挂号失败
97      throw new YyghException(20001, "挂号失败");
98  }
99 }
```