候补场景api定义文档

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 修改人 | 修改时间 | 变更记录 |
| 0.1 | 徐平 | 2016/7/25 | 创建 |
| 0.2 | 杜潇 | 2016/7/26 | 修订 |
| 0.3 | 杜潇 | 2016/8/2 | 修订 |
| 0.4 | 杜潇 | 2016/8/3 | 修订命名规范和功能块划分 |
| 0.5 | 徐平 | 2016/8/3 | 修改了部分方法名和字段名  增加了新的方法 |
| 0.6 | 杜潇 | 2016/8/4 | 合并修订 |
| 0.7 | 徐平 | 2016/8/4 | 1.增加了航班查询的参数以方便客户端过滤  2.修改离港旅客清除及旅客清除两个方法的 |
| 0.8 | 杜潇 | 2016/8/8 | 1.修订航班、锁定、解锁操作的提交数据结构和返回数据结构。  2.修订各个操作的传入值，从前端传入操作人信息，返回信息改为操作状态码和描述。  3.后补登记和查询航班允许一次填写多个出发地、目的地和航班号。 |
| 0.9 | 杜潇 | 2016/8/9 | 修订string和string[] |
| 1.0 | 杜潇 | 2016/8/9 | 增加REST接口访问标识 |
| 1.1 | 杜潇 | 2016/8/10 | 1.特殊服务字段类型改为字符串数组  2.增加通过刷证件获取旅客信息的接口 |
| 1.2 | 徐平、杜潇 | 2016/8/18 | 合并修订 |

目录

[候补场景api定义文档 1](#_Toc459296834)

[1. 航班 3](#_Toc459296835)

[航班查询 3](#_Toc459296836)

[单个航班查询 4](#_Toc459296837)

[航班锁定 5](#_Toc459296838)

[取消锁定 6](#_Toc459296839)

[操作项 6](#_Toc459296840)

[2. 升舱 7](#_Toc459296841)

[升舱信息查询 7](#_Toc459296842)

[升仓 8](#_Toc459296843)

[取消升仓 8](#_Toc459296844)

[操作项 9](#_Toc459296845)

[3. 志愿者 9](#_Toc459296846)

[志愿者信息查询 9](#_Toc459296847)

[志愿者选择 10](#_Toc459296848)

[取消选择 11](#_Toc459296849)

[操作项 11](#_Toc459296850)

[4. 获取客票信息 12](#_Toc459296851)

[5. 获取旅客信息 12](#_Toc459296852)

[6. 候补 13](#_Toc459296853)

[候补登记 13](#_Toc459296854)

[航班候补信息查询 14](#_Toc459296855)

[发送候补通知 15](#_Toc459296856)

[操作项 16](#_Toc459296857)

[候补旅客列表 16](#_Toc459296858)

[离港旅客清除 17](#_Toc459296859)

[旅客清除 17](#_Toc459296860)

[操作项 18](#_Toc459296861)

# 航班

## 航班查询

方法名: queryFlights()

Http Method: Get

REST: /guics/Flight/queryFlights/?requestData={JSON对象}

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| date | string | 日期  可选参数，不填则返回24小时内航班 | 2016-08-06 |
| fromAirportCode | string | 出发机场代码列表 |  |
| toAirportCode | string | 到达机场代码列表(可选) |  |
| flights | string | 航班号列表(可选)  我关注的和搜索通过指定此参数 |  |
| flightStatus | string | 状态列表(可选)  异常航班需要指定此参数 |  |
| startTime | string | 开始时间（可选）  时间段过滤 |  |
| endTime | string | 结束时间（可选）  时间段过滤 |  |

返回数据：

返回值为航班列表，单个航班的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| flightNo | string | 航班号 | MU5159 |
| date | string | 航班日期 |  |
| fromAirportCode | string | 出发机场代码 | SHA |
| toAirportCode | string | 到达机场代码 | PEK |
| departureTime | string | 起飞时间 | 13：15 |
| landingTime | string | 降落时间 | 15：15 |
| flightStatus | string | 航班状态(或状态码) | 正常（或状态代码:延误、取消） |
| isOverSold | bool | 是否超售 | True/false |
| acceptanceStatus | string | 值机状态(或状态码) | 尚未开始  正在值机  值机结束  已登机  已起飞 |
| boardingStatus | string | 登机状态 | 正在登机 |
| isLocked | bool | 是否锁定 | true/false |
| lockedByUser | string | 锁定人 |  |
| gate | string | 登机口 |  |
| candidatesCount | int | 候补人数 |  |

调用样例：

queryFlights('SHA','2016-08-06');

## 单个航班查询

方法名:getFlight()

Http Method: Get

REST: /guics/Flight/getFlight/?requestData={JSON对象}

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| flightNo | string | 航班号 | MU5159 |
| date | String | 航班日期 | 2016-08-04 |

返回数据：

返回值为航班列表，单个航班的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| flightNo | string | 航班号 | MU5159 |
| date | String | 航班日期 | 2016-08-04 |
| fromAirportCode | string | 出发机场代码 | SHA |
| toAirportCode | string | 到达机场代码 | PEK |
| departureTime | string | 起飞时间 | 13：15 |
| landingTime | string | 降落时间 | 15：15 |
| flightStatus | string | 航班状态(或状态码) | 正常（或状态代码:延误、取消） |
| isOverSold | bool | 是否超售 | True/false |
| acceptanceStatus | string | 值机状态(或状态码) | 尚未开始  正在值机  值机结束  已登机  已起飞 |
| isLocked | bool | 是否锁定 | true/false |
| lockedByUser | string | 锁定人 |  |
| gate | string | 登机口 |  |
| candidatesCount | int | 候补人数 |  |
| remainSeats | Object | 各仓位剩余座位数 |  |

调用样例：

## 航班锁定

方法名:lockFlight()

Http Method: Post requestData={JSON对象}

REST: /guics/Flight/lockFlight/

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| flightNo | string | 航班号 | MU5159 |
| date | string | 日期 | 2016-08-06 |

返回数据：

返回值为航班列表，单个航班的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| isLocked | bool | 是否锁定 | true/false |
| lockedByUser | string | 锁定人 |  |
| flightNo | string | 航班号 | MU5159 |

调用样例：

lockFlight('MU5159','2016-08-06')

## 取消锁定

方法名:unlockFlight()

Http Method: Post requestData={JSON对象}

REST: /guics/Flight/unlockFlight/

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| flightNo | string | 航班号 | MU5159 |
| date | string | 日期 | 2016-08-06 |

返回数据：

返回值为航班列表，单个航班的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| isLocked | bool | 是否锁定 | true/false |
| flightNo | string | 航班号 | MU5159 |
| date | string | 日期 | 2016-08-06 |

调用样例：

unlockFlight('MU5159','2016-08-06')

备注：

1.注意考虑锁定人在锁定航班后意外退出系统后怎样解锁的问题

2.操作人信息从访问标识中获取

## 操作项

筛选和排序功能在前端完成，是否关注存在localstorge里，所以查询只传时间查询48小时内的数据即可。以下字段在前端处理：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| fromAirportCode | string | 出发机场 | SHA |
| toAirportCode | string | 到达机场 | PEK |
| flightNo | string | 航班号 | MU5159 |
| date | string | 日期 | 2016-08-06 |
| departureTime | string | 起飞时间 | 13：15 |
| isNormal | bool | 是否异常，根据航班状态判断 | true/false |
| isFollowed | bool | 是否关注，用户点击关注时暂存在客户端 | true/false |

# 升舱

## 升舱信息查询

方法名: queryFlightPassengers()

Http Method: Get

REST: /guics/Alternate/queryFlightPassengers/?requestData={JSON对象}

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| flightNo | string | 航班号 | MU5159 |
| date | string | 日期 | 2016-08-06 |

返回数据：

返回值为候补客户列表，单个航班的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| valueWeight | string | 客户价值 | 6 |
| name | string | 姓名 | 张三 |
| frequentPassengerClass | string | 常客等级 | 5 |
| ticketSeatClass | string | 客票舱位 | X |
| newSeatClass | string | UP舱位 | F |
| specialService | string[] | 特殊服务 |  |
| fellows | string | 同行(同行状态、人员名单，信息链接) | [{sequence :'A'，passengerId:'1',passengerName:'张三'}，{ sequence :'A'，passengerId : '2',passengerName:'李四'}] |
| checkInBoardingStatus | string | 值机登机状态 |  |
| availableSeatClass | string | 可用升级舱位 | [U,F,J,W,Y] |
| comment | string | 备注 |  |

调用样例：

queryFlightPassengers ('MU5159','2016-08-06')

## 升仓

方法名:changeSeatClass ()

Http Method: Post requestData={JSON对象}

REST: /guics/Alternate/ changeSeatClass /

请求参数：

参数为旅客列表，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| date | string | 出发日期 | 2016-08-06 |
| flightNo | string | 航班号 | MU5159 |
| newSeatClass | string | 拟调整仓位代码 | [U,F,J,W,Y] |
| comment | string | 备注文本 | 经济舱全满 |

返回数据：

返回值为调整后状态列表，单个状态的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| seatClass | string | 新仓位代码 | U |

调用样例：

changeSeatClass ("[{'310XXXX0033','张三','2016-08-06','MU5159','F','经济舱全满'},{'310XXXX0034','李四','2016-08-06','MU5159', 'F','经济舱全满'}]")

## 取消升仓

方法名:cancelSeatClassChange ()

Http Method: Post requestData={JSON对象}

REST: /guics/Alternate/ cancelSeatClassChange /

请求参数：

参数为旅客列表，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| date | string | 出发日期 |  |
| flightNo | string | 航班号 | MU5159 |
| comment | string | 备注文本 | 经济舱全满 |

返回数据：

返回值为调整后状态列表，单个状态的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| seatClass | string | 原仓位代码 | F |

调用样例：

cancelSeatClassChange ("[{'310XXXX0033','张三','2016-08-06','MU5159'},{'310XXXX0034','李四','2016-08-06','MU5159'}]")

## 操作项

筛选和排序功能在前端完成。以下字段在前端处理：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| ticketSeatClass | string | 客票舱位 | F |
| checkInBoardingStatus | string | 值机登机状态 |  |
| changeSeatClassStatus | string | 升舱状态 |  |
| frequentPassengerClass | string | 价值排序 |  |
| fellows | string | 同行排序 | 3A |

# 志愿者

## 志愿者信息查询

方法名: queryFlightVolunteers ()

Http Method: Get

REST: /guics/ Volunteer / queryFlightVolunteers /?requestData={JSON对象}

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| flightNo | string | 航班号 | MU5159 |
| date | string | 日期 | 2016-08-06 |

返回数据：

返回值为候补客户列表，单个航班的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| valueWeight | string | 客户价值 | 6 |
| name | string | 姓名 | 张三 |
| frequentPassengerClass | string | 常客等级 | 5 |
| ticketSeatClass | string | 客票舱位 | F |
| specialService | string[] | 特殊服务 |  |
| fellows | object | 同行(同行状态、人员名单，信息链接) | [{sequence :'A'，passengerId:'1',passengerName:'张三'}，{ sequence :'A'，passengerId : '2',passengerName:'李四'}] |
| checkInBoardingStatus | string | 值机登机状态 |  |
| isVolunteer | bool | 志愿者选择状态 | true/false |
| comment | string | 备注 |  |

调用样例：

queryFlightVolunteers ( 'MU5159','2016-08-06')

## 志愿者选择

方法名:chooseVolunteers ()

Http Method: Post requestData={JSON对象}

REST: /guics/ Volunteer / chooseVolunteers /

请求参数：

参数为旅客列表，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| date | string | 出发日期 |  |
| flightNo | string | 航班号 | MU5159 |
| comment | string | 备注文本 | 经济舱全满 |

返回数据：

返回值为调整后状态列表，单个状态的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| isVolunteer | bool | 是否被选为志愿者 | true/false |

调用样例：

chooseVolunteers ("[{'310XXXX0033','张三','2016-08-06','MU5159','经济舱全满'},{'310XXXX0034','李四','2016-08-06','MU5159','经济舱全满'}]")

## 取消选择

方法名:cancelVolunteers ()

Http Method: Post requestData={JSON对象}

REST: /guics/ Volunteer / cancelVolunteers /

请求参数：

参数为旅客列表，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| date | string | 出发日期 |  |
| flightNo | string | 航班号 | MU5159 |
| comment | string | 备注文本 | 经济舱全满 |

返回数据：

返回值为调整后状态列表，单个状态的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| isVolunteer | bool | 是否被选为志愿者 | false |

调用样例：

cancelVolunteers ("[{'310XXXX0033','张三','2016-08-06','MU5159','经济舱全满'},{'310XXXX0034','李四','2016-08-06','MU5159','经济舱全满'}]")

## 操作项

筛选和排序功能在前端完成。以下字段在前端处理：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| ticketSeatClass | string | 客票舱位 | F |
| checkInBoardingStatus | string | 常客等级 | 6 |
| isVolunteer | bool | 志愿者选择状态 | true/false |
| passengerValue | string | 价值排序 | 5 |
| fellows | string | 同行排序 | 3A |

# 获取客票信息

方法名:queryTickets ()

Http Method: Get

REST: /guics/ RegisterCandidates / queryTickets /? requestData={JSON对象}

请求参数：

参数为旅客信息，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| fromAirportCode | string | 出发机场代码 (可选) | SHA |
| toAirportCode | string | 到达机场代码（可选） | PEK |

返回值为客票列表

Tickets字段定义

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| ticketNo | string | 票号 |  |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| seatClass | string | 仓位等级 |  |
| ticketStatus | string | 客票状态 |  |
| fromAirportCode | string | 出发机场代码 | SHA |
| toAirportCode | string | 到达机场代码 | PEK |

调用样例:

queryTickets ('310XXXX0033','张三',' SHA ',' PEK ')

# 获取旅客信息

方法名:queryPassenger ()

Http Method: Get

REST: /guics/ RegisterCandidates / queryPassengerInfo /? requestData={JSON对象}

请求参数：

参数为旅客信息，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客证件号 | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |

返回值为客票列表

Passenger字段定义

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| identityType | string | 证件类型(或类型码) |  |
| name | string | 旅客姓名 | 张三 |
| contact | string | 联系方式 |  |
| ageGroup |  |  |  |
| frequentPassengerClass |  |  |  |
| frequentPassengerCardNo |  |  |  |
| allianceCardNo |  |  |  |
| ticketInfos | object[] | 客票信息列表 |  |

调用样例:

queryPassenger ('310XXXX0033')

# 候补

## 候补登记

方法名: registerCandidates()

Http Method: Post requestData={JSON对象}

REST: /guics/ RegisterCandidates / registerCandidates /

请求参数：

参数为旅客列表，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| date | string | 日期  可选参数，不填则返回24小时内航班 | 2016-08-06 |
| fromAirportCode | string | 出发机场代码列表 | [“SHA”,“PVG”] |
| toAirportCode | string | 到达机场代码列表 | ["PEK"] |
| requiredSeatClass | string[] | 候补舱位列表 | [F,J,X] |
| requiredFlightNos | string[] | 指定航班号列表（可以为空） | ["MU5159"," MU51609"] |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| identityType | string | 证件类型(或类型码) |  |
| name | string | 旅客姓名 | 张三 |
| contact | string | 联系方式 |  |
| ageGroup | string | 年龄段(旅客属性) |  |
| frequentPassengerClass | string | 常旅客等级 |  |
| frequentPassengerCardNo | String | 常旅客卡号 |  |
| fellowIdentityNos | string[] | 关联旅客证件号 |  |
| **allianceCardNo** | string | 联盟卡号 |  |
| comment | string | 备注文本 | 此旅客特别着急 |
| ticketNo | string | 已选的票号 |  |

返回数据：

返回值为操作状态，单个状态定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| resultCode | string | 结果状态码 |  |
| resultDescription | string | 结果状态描述 |  |

调用样例：

registerCandidates("[{passengerInfo:…},{passengerInfo:…}]")

## 航班候补信息查询

方法名: queryFlightWaitingList()

Http Method: Get

REST: /guics/Alternate/queryFlightWaitingList/? requestData={JSON对象}

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| flightNo | string | 航班号 | MU5159 |
| date | string | 日期 | 2016-08-06 |

返回数据：

返回值为候补客户列表，单个航班的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| flightNo | string | 航班号 |  |
| date | string | 航班日期 |  |
| valueWeight | string | 客户价值 | 6 |
| name | string | 姓名 | 张三 |
| frequentPassengerClass | string | 常客等级 | 5 |
| frequentPassengerCardNo | string | 东航卡号 |  |
| requiredSeatClass | string | 候补舱位 | [F,J,X] |
| **allianceCardNo** | string | 联盟票 |  |
| ticketSeatClass | string | 客票舱位 | F |
| specialService | string[] | 特殊服务 |  |
| fellows | string | 同行(同行状态、人员名单，信息链接) | [{sequence :'A'，passengerId:'1',passengerName:'张三'}，{ sequence :'A'，passengerId : '2',passengerName:'李四'}] |
| isNoticed | boolean | 通知状态 |  |
| comment | string | 备注 |  |

调用样例：queryAlternateInfo('MU5159','2016-08-06')

## 发送候补通知

方法名: sendCandidateNotification()

Http Method: Post requestData={JSON对象}

REST: /guics/Alternate/sendCandidateNotification/

请求参数：

参数为旅客列表，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| date | string | 出发日期 | 2016-08-06 |
| flightNo | string | 航班号 | MU5159 |
| comment | string | 备注文本 | 此旅客特别着急 |

返回数据：

返回值为通知状态列表，单个通知状态的字段定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |
| flightNo |  |  |  |
| date |  |  |  |
| isNoticed | boolean | 通知状态(或状态码) | 已通知或Y |

调用样例：

sendCandidateNotification("[{'310XXXX0033','张三','2016-08-06','MU5159','此旅客特别着急'},{'310XXXX0034','李四','2016-08-06','MU5159','此旅客特别着急'}]")

## 操作项

筛选和排序功能在前端完成。以下字段在前端处理：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| requiredSeatClass | string | 候补舱位 | [F,J,X] |
| frequentPassengerClass | string | 常客等级 | 5 |
| notificationStatus | string | 通知状态 |  |
| passengerValue | string | 价值排序 | 2 |
| fellows | string | 同行排序 | 3A |

# 旅客列表

## 候补旅客列表

方法名: queryWaitingList()

Http Method: get

REST: /guics/ PassengerViewList / queryWaitingList /? requestData={JSON对象}

请求参数：

字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| fromAirportCode | string | 出发机场代码 | SHA |
| toAirportCode | string | 到达机场代码 | PEK |
| date | string | 日期 |  |

返回数据：

返回值为旅客列表，定义如下

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| valueWeight | string | 价值排序 | 1 |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| identityNoType | string | 证件类型(或类型码) |  |
| name | string | 旅客姓名 | 张三 |
| contact | string | 联系方式 |  |
| ageGroup | string | 年龄段(旅客属性) |  |
| frequentPassengerClass | string | 常旅客等级 |  |
| fromAirportCode | string | 出发机场代码 | SHA |
| toAirportCode | string | 到达机场代码 | PEK |
| requiredFlights | string[] | 候补航班列表 |  |
| requiredSeatClass | string[] | 可候补仓位 |  |
| noticedFlights | string[] | 已通知航班列表 |  |

调用样例：

queryWaitingList ('SHA',' PEK ')

## 离港旅客清除

方法名: clearDepartedPassengers ()

Http Method: Post requestData={JSON对象}

REST: /guics/ PassengerViewList / clearDepartedPassengers /

请求参数：

参数为全部旅客列表，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |

返回数据：

返回值为已离港旅客列表，单个字段定义如下

返回字段同旅客列表

调用样例：

clearDepartedPassengers ("[{'310XXXX0033','张三'},{'310XXXX0034','李四'}]")

## 旅客清除

方法名:clearPassengers ()

Http Method: Post

请求参数：

参数为待清除旅客列表，字段定义如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| identityNo | string | 旅客id号(或证件号) | 310XXXX0033 |
| name | string | 旅客姓名 | 张三 |

返回数据：

返回值同候补旅客列表

调用样例：

clearPassengers (‘310XXXX0033’,‘张三’)

## 操作项

筛选和排序功能在前端完成。以下字段在前端处理：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **值类型** | **备注** | **样例** |
| passengerValue | string | 客户价值 | 6 |
| passengerName | string | 姓名 | 张三 |
| alternateSeatClass | string | 候补舱位 | [F,J,X] |
| notificationStatus | string | 通知状态 |  |
| notificationTimes | Int | 通知次数 | 3 |