**版本历史**

|  |  |  |  |
| --- | --- | --- | --- |
| **文档版本** | **撰写时间** | **变更人** | **属性** |
| v1.0 |  |  | 新建 |
| V2.0 | 2014-8-20 | 王旭芳 | 1、运价直连支持多程报价功能。  搜索、价格校验、生单及支付校验请求接口**增加行程类型字段（请确保json解析标准，避免多输出一个字段解析出错的情况）**。  如果为多程的情况，将所有返回行程信息放在fromsegment中，其他保持不变。  2、已经接入的OTA，如果不需要接入多程，可不做任何调整，升级后的系统支持原有功能，有需要接入多程的OTA，，告知运价直连产品负责人张璐，开通权限即可。  系统会向开通权限的代理人发送搜索请求，否则不会发送。 |
| V2.1 | 2014-8-27 | 王旭芳 | 运价直连搜索接口返回参数中，增加退改签规则信息（rule） |
| V2.2 | 2015-3-10 | 李博 | 1、生单接口请求样例中,agetType修改为ageType（去掉t）。  2、价格校验、生单和支付三个接口的**请求样例中**，codeShare改为codeshare（s小写），返回样例的codeShare保持大写不变。 |
| V2.3 | 2015-3-19 | 李博 | 价格校验、生单和支付三个接口的sessionid长度限制由20调整为50，且不能为空。 |
| V2.4 | 2015-3-27 | 付彬桐 | 搜索，价格校验接口返回退改签部分字段变更，规范退改签字段值。 |
| V2.5 | 2015-6-16 | 张璐 | 在routings字段中增加报销凭证类型“ticketInvoiceType”字段，根据verify接口中ticketInvoiceType的值判断报销凭证的类型，0为行程单，1为旅行发票。 |
| V2.6 | 2015-6-16 | 李博 | 运价直连支持投放限制国籍和年龄产品，搜索和价格校验接口的Routing Element增加国籍和年龄限制录入项。 |
| V2.7 | 2015-8-21 | 解开妮 | 接口的Routing Element增加出票速度“**ticketTimeLimit**”字段录入项，表示用户支付后的最晚出票时限。单位为分钟，可输入1-10080内的正整数。超过该范围数值不生效，读取服务平台-出票速度全局设置-预定产品设置数值。请勿输入字符串。 |
| V2.8 | 2015-9-15 | 李文婷 | * 接口segement Element中增加字段“cabinClass”, 为非必填项，可输入1、2、3，分别代表1-economy，2-business，3-first |
| V2.81 | 2015-09-25 | 任洋洋 | * 更正json数据的格式 |
| V2.82 | 2015-09-30 | 马同辉 | * 更正json数据的格式 |
| V2.9 | 2015-10-25 | 刘宇超 | * 运价直连支持外币输入报价新增币种字段“currency”，为非必填项，可输入人民币或外币对应的3位合法字母，大小写任意，输入完成后，相应的价格字段自动依据指定的汇率进行人民币-外币转换，并返回人民币报价。 |
| V3.0 | 2017-01-01 | 张泽阳 | Search、verify接口请求参数，增加成人数、儿童数。 |
| V3.1 | 2017-07-10 | 张泽阳 | 细化Search、verify接口格式化行李额字段传输样例。 |
| V3.2 | 2017-07-26 | 陈志雄 | Search返回参数中增加字段：combineIndexs，标记拼接点，用户单程、多程和往返 |

Qunar 国际运价直连接口开发规范

# 规范说明

1. 提供 4 个接口：搜索、价格校验、生单、支付校验。
2. 提供接口 http 访问 url，统一字符编码：UTF-8 。
3. http 接口请求 content-Type="text/plain" 。
4. 接口通过 http post 请求接口 url 并发送符合 JSON 规范的 body。
5. 生单、支付校验接口传输数据须做 AES 加密后进行 BASE64 转码，密钥满足128 位。生产环境密钥在测试完成后由双方共同协商确定。

|  |
| --- |
| 加密模式：AES/CBC/PKCS5Padding  加密初始化向量：长度为 16 的空字节数组  测试用密钥：1234567890123456  加密样例如下  原文：  abcdefghigklmnopqrstuvwxyz0123456789  加密后：  8Z3dZzqn05FmiuBLowExK0CAbs4TY2GorC2dDPVlsn/tP+VuJGePqIMv1uSaVErr |

1. 接口使用 cid 身份识别及 IP 白名单验证机制。

cid: test

密钥：1234567890123456

# 搜索接口（search）

**搜索接口请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必须项 | 类型 | 详细说明 |
| cid | Yes | string | 接口身份标识用户名（渠道唯一标识） |
| tripType | Yes | String | 行程类型，  1：单程；  2：往返；  3：多程；  4：往返中推荐多程（消费者往返的请求，我们会推荐多程的结果信息，推荐的原则为行程的出发或到达出现在第二程中，比如消费者搜索A-B的往返，我们会推荐A-B,B-C或者A-B,C-A,不推荐A-B,C-D） |
| adultNum | Yes | int | 1. 该字段为1~9时，代表乘客类型为成人的乘机总人数 2. 该字段为-1时，代表不确定成人人数。   代理商可按1成人处理。  3．生单时成人人数可能与此字段不一致 |
| childNum | Yes | int | 1. 该字段为0~8时，代表乘客类型为儿童的乘机总人数 2. 该字段为-1时，代表不确定儿童人数。   代理商可按1儿童处理。  3．该字段为1~8时，返回的儿童价不能为0。  4．生单时儿童人数可能与此字段不一致 |
| fromCity | Yes | string | 出发地IATA三字码代码  如果为多程，会按照PEK/HKG，HKG/SHA格式请求，第一程为北京->香港，第二程为香港->上海 |
| toCity | Yes | string | 目的地城市 、到达城市IATA 三字码代码 |
| fromDate | Yes | string | 去程，格式为 YYYYMMDD，YYYYMMDD  如果为多程，20130729, 20130804方式传输数据 |
| retDate | No | string | 回程日期，格式为 YYYYMMDD（留空表示单程） |

**请求样例：**

|  |
| --- |
| 单程：  {  "cid": "test",  "tripTpye":"1",  "adultNum": 2,  "childNum": 1,  "fromCity": "BJS",  "toCity": "HKG",  "fromDate": "20130729",  "retDate": ""  }  往返：  {  "cid": "test",  "tripTpye":"2",  "adultNum": 2,  "childNum": 1,  "fromCity": "SHA",  "toCity": "HKG",  "fromDate": "20130729",  "retDate": "20130804"  }  多程：  {  "cid": "test",  "tripTpye":"3",  "adultNum": 2,  "childNum": 1,  "fromCity": "PEK/HKG,HKG/SHA",  "toCity":null,  "fromDate": "20130729, 20130804",  "retDate": ""  } |

**搜索接口返回参数：**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | 必须项 | 类型 | | 说明 | |
| status | Yes | int | | 0,成功；其他，失败 | |
| msg | Yes | string | | 提示信息，长度小于64 | |
| routings | Yes | array | | 报价信息，参考下面的Routing Element | |
| Routing Element | | | | | |
| data | Yes | string | 可保存必要信息，之后生单按原值回转。最大1000个字符 | | |
| currency | No | string | 1：没有该字段时，默认为RMB：currency：""  2：有currency字段时，需输入币种所对应的合法3位字母（currency："RMB|CNY", “USD”, “ EUR”, 或“ IDR”等，大小写任意） | | |
| adultPrice | Yes | int | 1：成人单价（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2：当currency字段输入不合法，或输入不为支付中心支持的币种符号，不返回人民币报价 | | |
| adultTax | Yes | int | 1：成人税费（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2：当currency字段输入不合法，或输入不为支付中心支持的币种符号，不返回人民币报价 | | |
| childPrice | Yes | int | 1：儿童单价（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2：当currency字段输入不合法，或输入不为支付中心支持的币种符号，不返回人民币报价 | | |
| childTax | Yes | int | 1：儿童税费（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2：当currency字段输入不合法，或输入不为支付中心支持的币种符号，不返回人民币报价 | | |
| nationalityType | Yes | int | 乘客国籍类型:0全部/1适用/2不适用 | | |
| nationality | Yes | string | 乘客国籍，可以为空，若输入则为国家二字码，多个用,分隔，最多支持五个国家或地区 | | |
| suitAge | Yes | string | 适用乘客年龄，可以为空，输入格式为12~59，表示适用于12~59岁的乘客预订，年龄限制范围为12~99，仅支持录入一个年龄段 | | |
| priceType | Yes | int | 报价类型：0 普通价 / 1留学生价 | | |
| applyType | Yes | int | 报价类型：0 预定价 / 1申请价/2预约 | | |
| adultTaxType | Yes | int | 成人税费类型：0 未含税 / 1已含税 | | |
| childTaxType | Yes | int | 儿童税费类型：0 未含税 / 1已含税 | | |
| ticketTimeLimit | Yes | int | 出票速度：[1~10080] （以分钟为单位） | | |
| ticketInvoiceType | Yes | int | 报销凭证类型：0行程单/1旅行发票  如果该字段没有，或者为空，默认为行程单类型 | | |
| **rule** | Yes | **object** | **退改签信息，参考 Rule Element** | | |
| fromSegments | Yes | array | 去程航段按顺序，参考下面的Segment Element  如果为多程第一程、第二程的信息，全输出到此节点。 | | |
| retSegments | Yes | array | 回程航段按顺序，参考下面的Segment Element（单程搜索为空） | | |
| combineIndexs | No | array | 分段出票报价的拼接点，嵌套array，每一程是一个array，最后放在一个array里。 如航段为A-B-C-D，D-C-B-A，拼接点为[[1,2],[1]]表示，去程的B，C为拼接点，返程的C为拼接点。如果不是分段出票报价可不填，某一程没有可填[ ]，如[[],[1]]表示去程没有拼接点，返程的第一点为拼接点。 若上述例子中的往返返回[[1]],则默认去程第一个拼节点B为拼节点。 | | |
| Rule Element | | | | | |
| hasRefund | Yes | int | | | 是否允许退票：0表示不允许，1表示允许 |
| refund | Yes | string | | | 可输入格式如：200-72-300-48-1000-0-\*，代表72小时前退票手续费200（单位以输入的币种为准）；48小时到72小时之间，退票手续费300（单位以输入的币种为准）；飞机起飞不足48小时，手续费1000（单位以输入的币种为准）；起飞后不予退票（输入\*），为空表示按航司规定 |
| partRefund | Yes | int | | | 部分退票规则，0表示不允许，1表示允许；单程直飞为0 |
| partRefundPrice | Yes | int | | | 1：部分退票费用，为空或正整数，为空表示按航司规定执行（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2： 当currency字段输入不合法，或输入不为支付中心支持的币种符号，该字段不返回结果 |
| hasEndorse | Yes | int | | | 是否允许改期：0表示不允许，1表示允许 |
| endorse | Yes | string | | | 可输入格式如：200-72-300-48-1000-0-\*，代表72小时前改期手续费200（单位以输入的币种为准）；48小时到72小时之间，改期手续费300（单位以输入的币种为准）；飞机起飞不足48小时，手续费1000（单位以输入的币种为准）；起飞后不予改期（输入\*），为空表示按航司规定 |
| partEndorse | Yes | int | | | 部分改期规则，0表示不允许，1表示允许；单程直飞为0 |
| partEndorsePrice | Yes | int | | | 1：部分改期费用，为空或正整数，为空表示按航司规定执行（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2： 当currency字段输入不合法，或输入不为支付中心支持的币种符号，该字段不返回结果 |
| endorsement | Yes | int | | | 签转规则，0表示不支持，1表示支持 |
| hasBaggage | Yes | int | | | 是否提供免费行李额：0表示不提供，1表示提供 |
| baggage | Yes | string | | | 1. 单程直飞时，样例：1-23，表示1PC，23kg。 2. 单程直飞时，样例：1-，表示1PC，kg数不限。 3. 单程直飞时，样例：-23，表示23kg，件数不限。 4. 单程中转一次和往返直飞录入1-23;1-23，中间用分号隔开。往返各中转两次录入1-23;1-23;1-23;1-23;1-23;1-23。 5. hasBaggage 为0时，须传“-;-;-;-”，表示四段均无行李（其中分号不可缺少）。 6. hasBaggage 为1时，传“-;-;-;-”，表示四段均按航司规定执行（其中分号不可缺少）。 |
| hasNoShow | Yes | int | | | 是否有noshow规则，0表示没有规则限制，1表示有规则限制 |
| noShowLimitTime | Yes | int | | | Noshow时限，输入正整数 |
| penalty | Yes | int | | | 1：Noshow罚金，可为空，若输入则为正整数。输入为空表示按照航司规定执行（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2： 当currency字段输入不合法，或输入不为支付中心支持的币种符号，该字段不返回结果 |
| specialNoShow | Yes | int | | | Noshow特殊规则：0（无） 1（Noshow时限内不允许退票） 2（Noshow时限内不允许改期） 3（Noshow时限内不允许退票和改期）， |
| other | Yes | string | | | 其他说明，最大 300 个字符 |
| Segment Element | | | | | |
| carrier | Yes | string | | | 航司IATA二字码，必须与flightNumber航司相同 |
| flightNumber | Yes | string | | | 航班号，如：CA123。航班号数字前若有多余的数字0，必须去掉，如CZ006需返回CZ6 |
| depAirport | Yes | string | | | 出发机场IATA三字码 |
| depTime | Yes | string | | | 起飞日期时间，格式：YYYYMMDDHHMM例：201203100300表示2012年3月10日3时0分 |
| arrAirport | Yes | string | | | 到达机场IATA三字码 |
| arrTime | Yes | string | | | 起飞日期时间，格式：YYYYMMDDHHMM例：201203101305表示2012年3月10日13时5分 |
| stopCities | Yes | string | | | 经停地，/分隔城市三字码 |
| codeShare | Yes | boolean | | | 代码共享标识（true代码共享/false非代码共享） |
| cabin | Yes | string | | | 舱位 |
| aircraftCode | Yes | string | | | 机型 |
| cabinClass | Yes | Int | | | 舱位等级，1代表经济舱，2代表商务舱，3代表头等舱， |

**返回成功样例：**

|  |
| --- |
| {  "status": 0,  "msg": "success",  "routings": [  {  "data": "3da0a93eba26c6e8f28955fe65f426fadbec03d9",  "currency"："RMB"  "adultPrice": 800,  "adultTax": 66,  "childPrice": 150,  "childTax": 35,  "nationalityType":1,  "nationality":"CN,HK,TW",  "suitAge":"12~59",  "adultTaxType": 0,  "childTaxType": 0,  "ticketTimeLimit": 60,  "priceType": 0,  "applyType": 0,  "rule": {  "hasRefund": 1,  "refund": "200-72-300-48-1000-0-\*",  "partRefund": 1,  "partRefundPrice": 500,  "hasEndorse": 1,  "endorse": "200-72-300-48-1000-\*",  "partEndorse":1,  "partEndorsePrice": 300,  "endorsement": 0,  "hasBaggage": 1,  "baggage": "1-23;2-30",  "hasNoShow": 1,  "noShowLimitTime": 48,  "penalty": 500,  "specialNoShow": 0,  "other": "退改签需要提前三个工作日联系客服"  },  "fromSegments": [  {  "carrier": "AA",  "depAirport": "LAX",  "depTime": "201203140140",  "arrAirport": "PEK",  "arrTime": "201203150530",  "stopCities": "",  "codeShare": false,  "cabin": "E",  "aircraftCode": "Boeing 777",  "flightNumber": "AA89",  “cabinClass”:2  },  {  "carrier": "CA",  "depAirport": "PEK",  "depTime": "201203151400",  "arrAirport": "TPE",  "arrTime": "201203151715",  "stopCities": "HKG/TYO/CTU",  "codeShare": false,  "cabin": "E",  "aircraftCode": "Boeing 737",  "flightNumber": "CA189",  “cabinClass”:2  }  ],  "retSegments": [],  "combineIndexs": [  [  1  ]  ]  }  ]  } |
|  |
|  |

**返回失败样例：**

|  |
| --- |
| {  "status": 999,  "msg": "ERROR MESSAGE"  } |

# 价格校验接口(verify)

**价格校验接口请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必须项 | 类型 | 详细说明 |
| cid | Yes | string | 接口身份标识用户名（渠道唯一标识） |
| tripType | Yes | string | 行程类型，  1：单程；  2：往返；  3：多程； |
| adultNum | Yes | int | 1. 该字段为1~9时，代表乘客类型为成人的乘机总人数 2. 该字段为-1时，代表不确定成人人数。   代理商可按1成人处理。  3．生单时成人人数可能与此字段不一致 |
| childNum | Yes | int | 1. 该字段为0~8时，代表乘客类型为儿童的乘机总人数 2. 该字段为-1时，代表不确定儿童人数   代理商可按1儿童处理  3．该字段为1~8时，返回的儿童价不能为0  4．生单时儿童人数可能与此字段不一致 |
| routing | Yes | object | 报价信息，参考搜索返回结果中的 Routing Elements。  **只含航班信息，不含价格信息 不包含cabinclass** |

**请求样例：**

|  |
| --- |
| {  "cid": "test",  "tripTpye": "1",  "adultNum": 2,  "childNum": 1,  "routing": {  "data": "3da0a93eba26c6e8f28955fe65f426fadbec03d9",  "fromSegments": [  {  "carrier": "AA",  "depAirport": "PEK",  "depTime": "201203140140",  "arrAirport": "HKG",  "arrTime": "201203150530",  "stopCities": "",  "codeshare": false,  "cabin": "E",  "aircraftCode": "Boeing 777",  "flightNumber": "AA89"  },  {  "carrier": "CA",  "depAirport": "HKG",  "depTime": "201204151400",  "arrAirport": "TPE",  "arrTime": "201204151715",  "stopCities": "",  "codeshare": false,  "cabin": "E",  "aircraftCode": "Boeing 737",  "flightNumber": "CA189",  }  ],  "retSegments": []  }  } |

**价格校验接口返回参数：**

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 说明 |
| status | int | 0,成功；其他，失败 |
| msg | string | 提示信息，长度小于64 |
| sessionId | string | 会话标识：标记服务接口调用的唯一标识，相应的调用结果中会原值返回。数字或字母，长度小于50个字符,且不能为空。 |
| maxSeats | int | 可预订的座位数，最大为 9 |
| routing | object | 报价信息，参考搜索返回结果中的 Routing Elements。（注意：combineIndexs 一定要和搜索保持一致） |
| rule | object | 退改签信息，参考 Rule Element |
| Rule Element | | |
| hasRefund | int | 是否允许退票：0表示不允许，1表示允许 |
| currency | string | 1：没有该字段时，默认为RMB：currency：""  2：有currency字段时，需输入币种所对应的合法3位字母（currency："RMB|CNY", “USD”, “ EUR”, 或“ IDR”等，大小写任意） |
| refund | string | １：可输入格式如：200-72-300-48-1000-0-\*，代表72小时前退票手续费200（单位以所输入的币种为准）；48小时到72小时之间，退票手续费300（单位以所输入的币种为准）；飞机起飞不足48小时，手续费1000（单位以所输入的币种为准）；起飞后不予退票（输入\*），为空表示按航司规定 |
| partRefund | int | 部分退票规则，0表示不允许，1表示允许；单程直飞为0 |
| partRefundPrice | int | １：部分退票费用，为空或正整数，为空表示按航司规定执行（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2： 当currency字段输入不合法，或输入不为支付中心支持的币种符号，该字段不返回结果 |
| hasEndorse | int | 是否允许改期：0表示不允许，1表示允许 |
| endorse | string | 可输入格式如：200-72-300-48-1000-0-\*，代表72小时前改期手续费200（单位以所输入的币种为准）；48小时到72小时之间，改期手续费300（单位以所输入的币种为准）；飞机起飞不足48小时，手续费1000（单位以所输入的币种为准）；起飞后不予改期（输入\*），为空表示按航司规定 |
| partEndorse | int | 部分改期规则，0表示不允许，1表示允许；单程直飞为0 |
| partEndorsePrice | int | １：部分改期费用，为空或正整数，为空表示按航司规定执行（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2： 当currency字段输入不合法，或输入不为支付中心支持的币种符号，该字段不返回结果 |
| endorsement | int | 签转规则，0表示不支持，1表示支持 |
| hasBaggage | int | 是否提供免费行李额：0表示不提供，1表示提供 |
| baggage | string | 1. 单程直飞时，样例：1-23，表示1PC，23kg。 2. 单程直飞时，样例：1-，表示1PC，kg数不限。 3. 单程直飞时，样例：-23，表示23kg，件数不限。 4. 单程中转一次和往返直飞录入1-23;1-23，中间用分号隔开。往返各中转两次录入1-23;1-23;1-23;1-23;1-23;1-23。 5. hasBaggage 为0时，须传“-;-;-;-”，表示四段均无行李（其中分号不可缺少）。 6. hasBaggage 为1时，传“-;-;-;-”，表示四段均按航司规定执行（其中分号不可缺少）。 |
| hasNoShow | int | 是否有noshow规则，0表示没有规则限制，1表示有规则限制 |
| noShowLimitTime | int | Noshow时限，输入正整数 |
| penalty | int | １：Noshow罚金，可为空，若输入则为正整数。输入为空表示按照航司规定执行（单位以所输入的币种为准，如：USD, EUR, 或IDR等）  2： 当currency字段输入不合法，或输入不为支付中心支持的币种符号，该字段不返回结果 |
| specialNoShow | int | Noshow特殊规则：0（无） 1（Noshow时限内不允许退票） 2（Noshow时限内不允许改期） 3（Noshow时限内不允许退票和改期）， |
| other | string | 其他说明，最大 300 个字符 |

**返回成功样例：**

|  |
| --- |
| {  "status": 0,  "msg": "success",  "sessionId": "S12345",  "maxSeats": 9,  "routing": {  "data": "3da0a93eba26c6e8f28955fe65f426fadbec03d9",  "currency"："RMB"  "adultPrice": 800,  "adultTax": 66,  "childPrice": 150,  "childTax": 35,  "nationalityType":1,  "nationality":"CN,HK,TW",  "suitAge":"12~59",  "adultTaxType": 0,  "childTaxType": 0,  "ticketTimeLimit": 60,  "priceType": 0,  "applyType": 0,  "rule": {  "hasRefund": 1,  "refund": "200-72-300-48-1000-0-\*",  "partRefund": 1,  "partRefundPrice": 500,  "hasEndorse": 1,  "endorse": "200-72-300-48-1000-\*",  "partEndorse":1,  "partEndorsePrice": 300,  "endorsement": 0,  "hasBaggage": 1,  "baggage": "1-23;2-30",  "hasNoShow": 1,  "noShowLimitTime": 48,  "penalty": 500,  "specialNoShow": 0,  "other": "退改签需要提前三个工作日联系客服"  },  "fromSegments": [  {  "carrier": "AA",  "depAirport": "LAX",  "depTime": "201203140140",  "arrAirport": "PEK",  "arrTime": "201203150530",  "stopCities": "",  "codeShare": false,  "cabin": "E",  "aircraftCode": "Boeing 777",  "flightNumber": "AA89",  “cabinClass”:2  },  {  "carrier": "CA",  "depAirport": "PEK",  "depTime": "201203151400",  "arrAirport": "TPE",  "arrTime": "201203151715",  "stopCities": "HKG/TYO/CTU",  "codeShare": false,  "cabin": "E",  "aircraftCode": "Boeing 737",  "flightNumber": "CA189",  “cabinClass”:2  }  ],  "retSegments": [],  "combineIndexs": [  [  1  ]  ]  },  "rule": {  "hasRefund": 1,  "refund": "200-72-300-48-1000-0-\*",  "partRefund": 1,  "partRefundPrice": 500,  "hasEndorse": 1,  "endorse": "200-72-300-48-1000-\*",  "partEndorse":1,  "partEndorsePrice": 300,  "endorsement": 0,  "hasBaggage": 1,  "baggage": "1-23;2-30",  "hasNoShow": 1,  "noShowLimitTime": 48,  "penalty": 500,  "specialNoShow": 0,  "other": "退改签需要提前三个工作日联系客服"  }  } |
|  |

**返回失败样例：**

|  |
| --- |
| {  "status": 505,  "msg": "ERROR MESSAGE"  } |

# 生单接口（order）

**生单接口请求参数：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 必须项 | | 类型 | 详细说明 |
| cid | Yes | | string | 接口身份标识用户名（渠道唯一标识） |
| orderSource | No | | int | 订单来源：“1”为抢单池订单 |
| tripType | Yes | | string | 行程类型，  1：单程；  2：往返；  3：多程； |
| sessionId | Yes | | string | 会话标识：标记服务接口调用的唯一标识，相应的调用结果中会原值返回。数字或字母，长度小于50个字符，且不能为空。 |
| routing | Yes | | object | 报价信息，参考搜索返回结果中的 Routing Elements。  **只含航班信息，不含价格信息.不包含CabinClass** |
| Passengers | Yes | | array | 乘机人信息，参考下面的Passenger Element |
| contact | Yes | | object | 联系人信息，参考下面的Contact Element |
| Passengers Element 乘机人信息 | | | | |
| name | | Yes | string | LastName/FirstName MiddleName，姓/名 |
| ageType | | Yes | int | 乘客类型，0 成人/1 儿童/-1 留学生 |
| birthday | | Yes | string | 生日，格式：YYYYMMDD |
| gender | | Yes | string | 乘客性别，M 男 / F 女 |
| cardNum | | Yes | string | 证件号码，最大 15 个字符 |
| cardType | | Yes | string | 证件类型：  PP - 护照  GA - 港澳通行证  TW - 台湾通行证  TB - 台胞证  HX -回乡证  HY - 国际海员证 |
| cardIssuePlace | | Yes | string | 证件发行国家，国家二字码 |
| cardExpired | | Yes | string | 证件有效时间，格式：YYYYMMDD |
| nationality | | Yes | string | 乘客国籍，国家二字码 |
| Contact Element 联系人信息 | | | | |
| name | | Yes | string | 联系人姓名，不单独区分姓和名 |
| address | | No | string | 详细地址 |
| postcode | | No | string | 邮编 |
| email | | No | string | 联系人邮箱（留空） |
| mobile | | No | string | 联系人手机号，接口方提供固定手机号，不使用用户手机号 |

**请求样例：**

|  |
| --- |
| {  "cid": "test",  "orderSource": 1,  "tripTpye": "1",  "sessionId": "S12345",  "routing": {  "data": "3da0a93eba26c6e8f28955fe65f426fadbec03d9",  "fromSegments": [  {  "carrier": "AA",  "depAirport": "PEK",  "depTime": "201203140140",  "arrAirport": "HKG",  "arrTime": "201203150530",  "stopCities": "",  "codeshare": false,  "cabin": "E",  "aircraftCode": "Boeing 777",  "flightNumber": "AA89",  },  {  "carrier": "CA",  "depAirport": "HKG",  "depTime": "201204151400",  "arrAirport": "TPE",  "arrTime": "201204151715",  "stopCities": "",  "codeshare": false,  "cabin": "E",  "aircraftCode": "Boeing 737",  "flightNumber": "CA189",  }  ],  "retSegments": []  },  "passengers": [  {  "name": " Marcus/Tom",  "ageType": 0,  "birthday": "19740225",  "gender": "M",  "cardType": "PP",  "cardNum": "G756789",  "cardExpired": "20200101",  "cardIssuePlace": "CN",  "nationality": "CN"  },  {  "name": " Marcus/Jack",  "ageType": 1,  "birthday": "20060225",  "gender": "M",  "cardType": "PP",  "cardNum": "G756789",  "cardExpired": "20200101",  "cardIssuePlace": "CN",  "nationality": "CN"  }  ],  "contact": {  "name": " Marcus/Tom ",  "address": "xxxx spring valley road",  "postcode": "12345",  "email": "",  "mobile": "13800000000"  }  } |

**生单成功返回参数：**

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 说明 |
| status | int | 0,成功；其他，失败 |
| msg | string | 提示信息，长度小于64 |
| sessionId | string | 会话标识：标记服务接口调用的唯一标识，相应的调用结果中会原值返回。数字或字母，长度小于50个字符，且不能为空。 |
| orderNo | string | 订单号，最大 100 个字符 |
| pnrCode | string | PNR 编码，最大 30 个字符 |
| maxSeats | int | 可预订的座位数，最大为 9 |
| routing | object | 报价信息，参考搜索返回结果中的 Routing Elements。 |
| currency | string | 1：没有该字段时，默认为RMB：currency：""  2：有currency字段时，需输入币种所对应的合法3位字母（currency："RMB|CNY", “USD”, “ EUR”, 或“ IDR”等，大小写任意） |

**返回成功样例：**

|  |
| --- |
| {  "status": 0,  "msg": "success",  "sessionId": "S12345",  "orderNo": "Ticket-AAA-2013031209-0001",  "pnrCode": "HUICDE",  "maxSeats": 9,  "routing": {  "data": "3da0a93eba26c6e8f28955fe65f426fadbec03d9",  "currency"："RMB"  "adultPrice": 800,  "adultTax": 66,  "childPrice": 0,  "childTax": 0,  "adultTaxType": 0,  "childTaxType": 0,  "ticketTimeLimit": 60,  "priceType": 0,  "applyType": 0,  "fromSegments": [  {  "carrier": "AA",  "depAirport": "PEK",  "depTime": "201203140140",  "arrAirport": "HKG",  "arrTime": "201203150530",  "stopCities": "",  "codeShare": false,  "cabin": "E",  "aircraftCode": "Boeing 777",  "flightNumber": "AA89",  “cabinClass”:2  },  {  "carrier": "CA",  "depAirport": "HKG",  "depTime": "201204151400",  "arrAirport": "TPE",  "arrTime": "201204151715",  "stopCities": "",  "codeShare": false,  "cabin": "E",  "aircraftCode": "Boeing 737",  "flightNumber": "CA189",  “cabinClass”:2  }  ],  "retSegments": []  }  } |

**返回失败样例：**

|  |
| --- |
| {  "status": 999,  "msg": "ERROR MESSAGE"  } |

# 支付校验接口(pay)

**支付校验接口请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必须项 | 类型 | 详细说明 |
| cid | Yes | string | 接口身份标识用户名（渠道唯一标识） |
| tripType | Yes | string | 行程类型，  1：单程；  2：往返；  3：多程； |
| sessionId | Yes | string | 会话标识：标记服务接口调用的唯一标识，相应的调用结果中原值返回。数字或字母，长度小于50个字符，且不能为空。 |
| orderNo | Yes | string | 生成订单成功时返回的订单号 |
| pnrCode | Yes | string | 生成订单成功时返回的 PNR 编码 |
| routing | Yes | object | 报价信息，参考搜索返回结果中的 Routing Elements。  **只含航班信息，不含价格信息.不包含CabinClass** |

**接口请求样例：**

|  |
| --- |
| {  "cid": "test",  "tripTpye": "1",  "sessionId": "S12345",  "orderNo": "Ticket-AAA-2013031209-0001",  "pnrCode": "XXXXXX",  "routing": {  "data": "3da0a93eba26c6e8f28955fe65f426fadbec03d9",  "fromSegments": [  {  "carrier": "AA",  "depAirport": "LAX",  "depTime": "201203140140",  "arrAirport": "PEK",  "arrTime": "201203150530",  "stopCities": "",  "codeshare": false,  "cabin": "E",  "aircraftCode": "Boeing 777",  "flightNumber": "AA89",  },  {  "carrier": "CA",  "depAirport": "PEK",  "depTime": "201203151400",  "arrAirport": "TPE",  "arrTime": "201203151715",  "stopCities": "HKG/TYO/CTU",  "codeshare": false,  "cabin": "E",  "aircraftCode": "Boeing 737",  "flightNumber": "CA189" ,  }  ],  "retSegments": []  }  } |

**支付校验接口返回参数：**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必填项 | 类型 | 说明 |
| status | Yes | int | 0,成功；其他，失败 |
| msg | Yes | string | 提示信息 |
| sessionId | Yes | string | 会话标识：标记服务接口调用的唯一标识，相应的调用结果中会原值返回。数字或字母，长度小于50个字符，且不能为空。 |
| orderNo | Yes | string | 请求支付校验时提交的订单号。返回必须与请求参数一致。 |
| pnrCode | Yes | string | |  | | --- | | 请求支付校验时提交的PNR编码。返回必须与请求参数一致。 | |
| routing | Yes | object | 报价信息，参考搜索返回结果中的 Routing Elements。 |
| currency | No | string | 1：没有该字段时，默认为RMB：currency：""  2：有currency字段时，需输入币种所对应的合法3位字母（currency："RMB|CNY", “USD”, “ EUR”, 或“ IDR”等，大小写任意） |

**返回成功样例：**

|  |
| --- |
| {  "status": 0,  "msg": "success",  "sessionId": "S12345",  "orderNo": "Ticket-AAA-2013031209-0001",  "pnrCode": "XXXXXX",  "routing": {  "data": "3da0a93eba26c6e8f28955fe65f426fadbec03d9",  "currency"："RMB"  "adultPrice": 800,  "adultTax": 66,  "childPrice": 0,  "childTax": 0,  "adultTaxType": 0,  "childTaxType": 0,  "ticketTimeLimit": 60,  "priceType": 0,  "applyType": 0,  "fromSegments": [  {  "carrier": "AA",  "depAirport": "LAX",  "depTime": "201203140140",  "arrAirport": "PEK",  "arrTime": "201203150530",  "stopCities": "",  "codeShare": false,  "cabin": "E",  "aircraftCode": "Boeing 777",  "flightNumber": "AA89",  “cabinClass”:2  },  {  "carrier": "CA",  "depAirport": "PEK",  "depTime": "201203151400",  "arrAirport": "TPE",  "arrTime": "201203151715",  "stopCities": "HKG/TYO/CTU",  "codeShare": false,  "cabin": "E",  "aircraftCode": "Boeing 737",  "flightNumber": "CA189",  “cabinClass”:2  }  ],  "retSegments": []  }  } |

**返回失败样例：**

|  |
| --- |
| {  "status": -1,  "msg": "ERROR MESSAGE"  } |

# 附录(加解密)

* 1. **java**

|  |
| --- |
| */\*\*  \* 加密  \*   \** ***@param*** *content 需要加密的内容  \** ***@param*** *password 加密密码  \** ***@return*** *\*/* public static String encrypt(String content, String password) throws Exception{  try {  byte[] raw = password.getBytes(Charsets.*UTF\_8*);  if (raw.length != 16) {  throw new IllegalArgumentException("Invalid key size. " + password + ", 密钥token长度不是16位");  }   javax.crypto.spec.SecretKeySpec skeySpec = new SecretKeySpec(raw, "AES");  javax.crypto.Cipher cipher = Cipher.*getInstance*("AES/CBC/PKCS5Padding");  cipher.init(Cipher.*ENCRYPT\_MODE*, skeySpec, new javax.crypto.spec.IvParameterSpec(new byte[16])); // zero IV  byte[] final = cipher.doFinal(content.getBytes(Charsets.*UTF\_8*));  return new String(org.apache.commons.codec.binary.Base64.encodeBase64(final));  } catch (Exception e) {  *logger*.error(e.getMessage(), e);  throw e;  } }  */\*\*  \* 加密  \*   \** ***@param*** *content 需要解密的内容  \** ***@param*** *password 加密密码  \** ***@return*** *\*/*  public static String decrypt(String content, String password) throws Exception{  try {  byte[] raw = password.getBytes(Charsets.*UTF\_8*);  if (raw.length != 16) {  throw new IllegalArgumentException("Invalid key size. " + password + ", 密钥token长度不是16位");  }  SecretKeySpec skeySpec = new SecretKeySpec(raw, "AES");   Cipher cipher = Cipher.*getInstance*("AES/CBC/PKCS5Padding");  cipher.init(Cipher.*DECRYPT\_MODE*, skeySpec, new IvParameterSpec(new byte[16]));  byte []toDecrypt = org.apache.commons.codec.binary.Base64.*decodeBase64*(content.getBytes())  byte[] original = cipher.doFinal(toDecrypt);  return new String(original, Charsets.*UTF\_8*);  } catch (IllegalArgumentException e) {  *logger*.error(e.getMessage(), e);  throw e;  } catch (Exception e){  *logger*.error(e.getMessage(), e);  }  return null; } |

* 1. **c#**

|  |
| --- |
| /// <summary>  /// AES加密  /// </summary>  /// <param name="plainStr">明文字符串</param>  /// <returns>密文</returns>  public static string AESEncrypt(string plainStr, string Key)  {  byte[] IV = new byte[16];  byte[] bKey = Encoding.UTF8.GetBytes(Key);  byte[] byteArray = Encoding.UTF8.GetBytes(plainStr);  string encrypt = null;  Rijndael aes = Rijndael.Create();  try  {  using (MemoryStream mStream = new MemoryStream())  {  using (CryptoStream cStream = new CryptoStream(mStream, aes.CreateEncryptor(bKey, IV), CryptoStreamMode.Write))  {  cStream.Write(byteArray, 0, byteArray.Length);  cStream.FlushFinalBlock();  encrypt = Convert.ToBase64String(mStream.ToArray());  }  }  }  catch { }  aes.Clear();  return encrypt;  }  /// <summary>  /// AES解密  /// </summary>  /// <param name="encryptStr">密文字符串</param>  /// <returns>明文</returns>  public static string AESDecrypt(string encryptStr,string Key)  {  byte[] IV = new byte[16];  byte[] bKey = Encoding.UTF8.GetBytes(Key);  byte[] byteArray = Convert.FromBase64String(encryptStr);  string decrypt = null;  Rijndael aes = Rijndael.Create();  try  {  using (MemoryStream mStream = new MemoryStream())  {  using (CryptoStream cStream = new CryptoStream(mStream, aes.CreateDecryptor(bKey, IV), CryptoStreamMode.Write))  {  cStream.Write(byteArray, 0, byteArray.Length);  cStream.FlushFinalBlock();  decrypt = Encoding.UTF8.GetString(mStream.ToArray());  }  }  }  catch { }  aes.Clear();  return decrypt;  } |

* 1. **php**

|  |
| --- |
| class EncryptDecryptHelper {  const TOKEN = '1234567890123456';  public static $key\_usage = self::TOKEN;  public static function encryptAES($jsonResponse){  $iv = pack('H\*', "00000000000000000000000000000000");  $pkcs5Padded = self::pkcs5Pad($jsonResponse);  $encrypted = mcrypt\_encrypt(MCRYPT\_RIJNDAEL\_128, self::$key\_usage, $pkcs5Padded, MCRYPT\_MODE\_CBC, $iv);  $encoded = base64\_encode($encrypted);  return $encoded;  }  public static function decryptAES($text) {  $iv = pack('H\*', "00000000000000000000000000000000");  $key = self::$key\_usage;  $toDecrypt = base64\_decode($text);  $decoded = mcrypt\_decrypt(MCRYPT\_RIJNDAEL\_128, $key, $toDecrypt, MCRYPT\_MODE\_CBC, $iv);  $pkcs5Padded = self::pkcs5Unpad($decoded);  return $pkcs5Padded ;  }  private static function ordutf8($string, &$offset)  {  $code = ord(substr($string, $offset, 1));  if ($code >= 128) { //otherwise 0xxxxxxx  if ($code < 224) $bytesnumber = 2; //110xxxxx  else if ($code < 240) $bytesnumber = 3; //1110xxxx  else if ($code < 248) $bytesnumber = 4; //11110xxx  $codetemp = $code - 192 - ($bytesnumber > 2 ? 32 : 0) - ($bytesnumber > 3 ? 16 : 0);  for ($i = 2; $i <= $bytesnumber; $i++) {  $offset++;  $code2 = ord(substr($string, $offset, 1)) - 128; //10xxxxxx  $codetemp = $codetemp \* 64 + $code2;  }  $code = $codetemp;  }  $offset += 1;  if ($offset >= strlen($string)) $offset = -1;  return $code;  }  private static function pkcs5Pad($text)  {  $size = mcrypt\_get\_block\_size(MCRYPT\_RIJNDAEL\_128, MCRYPT\_MODE\_CBC);  $pad = $size - (mb\_strlen($text) % $size);  return $text . str\_repeat(chr($pad), $pad);  }  private static function pkcs5Unpad($text)  {  $offset = mb\_strlen($text) - 1;  $pad = self::ordutf8($text, $offset);  if ($pad > mb\_strlen($text)) {  return $text;  }  if (strspn($text, $text{mb\_strlen($text) - 1}, mb\_strlen($text) - $pad) != $pad) {  return $text;  }  return mb\_substr($text, 0, -1 \* $pad);  }  } |