**途牛火车票供应商接口手册**

V3.8.8

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 修订内容 | 修改人 | 修改日期 |
| V1.0 | 初稿 | 余松涛 | 2015-06-09 |
| V2.0 | 增加 3.9-3.13接口 | 余松涛 | 2015-07-05 |
| V2.1 | 去除3.9批量功能  增加3.14-3.15接口 | 余松涛 | 2015-08-11 |
| V2.2 | 增加必要接口说明  修正3.14序列号不对的问题 | 余松涛 | 2015-09-13 |
| V2.3 | 增加3.16取消抢票功能 | 王文军 | 2015-10-16 |
| V2.4 | 增加3.16的回调地址参数 | 王文军 | 2015-11-17 |
| V2.5 | 修改3.5线上退票，添加refundId（退票请求流水号）字段，标志每次退票请求 | 张岩 | 2015-12-31 |
| V2.6 | 1. 修改联系人相关接口，把identyType改为string类型 2. 删除常用联系人接口增加根据身份证件号删除所需字段 | 徐骥 | 2016-01-08 |
| V2.7 | 1. 增加改签申请，取消改签、确认改签三个接口 2. 对原乘客信息描述，增加改签特有的字段oldTicketNo和newTicketNo | 徐骥 | 2016-02-05 |
| v2.8 | 1. 增加催退款通知接口 2. 抢票接口增加入参：备选车次、备选座等字段。 3. 抢票成功、抢票失败、取消抢票成功增加返回参数：当前订单查询余票次数、余票不为0尝试失败次数、当前订单从抢票开始到抢票结束同时有多少人在抢；抢票入参增加抢票截止时间和相应的错误码 4. 预定下单、确认出票增加可能的错误码 5. 改签接口取消对变更到达站的支持 6. 增加报警接口，实现占位确认等操作已超过的时间，提醒供应商抓紧处理 | 兰伟敏 | 2016-03-09 |
| v2.9 | 1. 预定下单、确认出票、取消订单、线上退票模块的返回参数中code、msg变更 | 兰伟敏 | 2016-03-21 |
| v2.9.1 | 12306常旅相关联系人接口调整（data加密处理） | 徐骥 | 2016-04-04 |
| V3.0 | 增加线下退款进度查询接口 | 张岩 | 2016-04-05 |
| V3.1 | 身份验证接口新增isPass字段返回错误码 | 单伟峰 | 2016-04-11 |
| V3.1.1 | 占位反馈新增3个返回码 | 兰伟敏 | 2016-04-20 |
| V3.2 | 新增查询火车票状态接口 | 单伟峰 | 2016-05-17 |
| V3.2.1 | 催退款通知接口增加2001错误码 | 张岩 | 2016-05-23 |
| V3.2.2 | 修改了文档格式；修改了3.23接口描述 | 崔子龙 | 2016-05-25 |
| V3.2.3 | 抢票次数接口 | 单伟峰 | 2016-05-25 |
| V3.3 | 12306直连新增登陆、验证码、占位相关接口 | 单伟峰 | 2016-06-03 |
| V3.3.1 | 查询火车票状态接口字段修正（vendorOrderId为必传） | 单伟峰 | 2016-06-14 |
| V3.3.2 | 改签新增返回码 | 单伟峰 | 2016-06-14 |
| V3.3.3 | 12306验证码获取、验证码提交、订单查询、12306登录核验接口 | 李阳 | 2016-06-15 |
| V3.3.4 | 12306直连新增前置分单接口、反刷订单号接口 | 李阳 | 2016-06-22 |
| V3.3.5 | 12306直连接口错误码更新 | 李阳 | 2016-06-28 |
| V3.5 | 占位、取消占位、出票、线上退票新增一批返回码 | 余学文 | 2016-10-13 |
| V3.6 | 新增12306验证码获取、12306验证码提交、  12306用户登录核验 | 李阳 | 2016-11-14 |
| V3.7 | 新增12306注册账号校验、12306注册验证码获取、12306注册提交 | 张岩11 | 2016-12-0  6 |
| V3.8 | 新增预定下单12306排队状态推送 | 李阳23 | 2016-12-17 |
| V3.8.1 | 占位接口新增12306选座字段 | 张岩11 | 2016-12-26 |
| V3.8.2 | 常旅查询接口，新增添加时间字段 | 李阳23 | 2017-01-24 |
| V3.8.3 | 修复文档中接口出入参数实例中多空格 | 余学文 | 2017-02-13 |
| V3.8.4 | 常旅增加学生信息、出票反馈增加检票口信息 | 余学文 | 2017-03-13 |
| V3.8.5 | 新增学生票占位必填字段约束 | 李阳23 | 2017-06-07 |
| V3.8.6 | 新增订单状态推送接口 | 李阳23 | 2017-07-24 |
| V3.8.7 | 抢票取消接口异步改同步 | 李阳23 | 2017-08-01 |
| V3.8.8 | 套皮支付更新   1. 预定下单、改签占位新增paymentUrl 2. 新增查询指定订单支付链接接口   3、新增查询指定订单的出票（或改签出票）状态接口 | 余学文 | 2017-09-19 |
| V3.8.9 | 占位和改签占位，添加clearTime字段要求 | 邵冬冬 | 2017-10-26 |

目录

[1. 概述 6](#_Toc488666245)

[1.1 目的 6](#_Toc488666246)

[2015.09 API术语 6](#_Toc488666247)

[2. API调用详解 6](#_Toc488666248)

[2.1 请求流程 6](#_Toc488666249)

[2.2数据格式 6](#_Toc488666250)

[2.3应用环境 7](#_Toc488666251)

[2.4安全机制 8](#_Toc488666252)

[2.5注意事项 9](#_Toc488666253)

[2015. 火车票业务接口 10](#_Toc488666254)

[3.1余位查询（必要） 10](#_Toc488666255)

[3.2占位（必要） 13](#_Toc488666256)

[3.3出票（必要） 19](#_Toc488666257)

[3.4取消占位（必要） 21](#_Toc488666258)

[3.5线上退票（必要） 23](#_Toc488666259)

[3.6线下退款通知（必要） 28](#_Toc488666260)

[3.7查询经停站 30](#_Toc488666261)

[3.8身份证核验 32](#_Toc488666262)

[**3.9 12306账号验证** 34](#_Toc488666263)

[**3.10 查询常用联系人** 35](#_Toc488666264)

[**3.11 增加和修改常用联系人** 38](#_Toc488666265)

[**3.12 删除常用联系人** 41](#_Toc488666266)

[**3.13 抢票** 43](#_Toc488666267)

[**3.14 线下出票** 48](#_Toc488666268)

[**3.15 快递信息回调** 53](#_Toc488666269)

[**3.16 取消抢票** 55](#_Toc488666270)

[3.17 改签 58](#_Toc488666271)

[3.18 取消改签 63](#_Toc488666272)

[3.19 确认改签 66](#_Toc488666273)

[3.20 催退款通知 68](#_Toc488666274)

[3.21 报警通知 70](#_Toc488666275)

[3.22 线下退款进度查询 72](#_Toc488666276)

[3.23 订单跟踪 74](#_Toc488666277)

[3.24抢票次数查询 77](#_Toc488666278)

[3.25 12306验证码获取 78](#_Toc488666279)

[3.26 12306验证码提交 82](#_Toc488666280)

[3.27 12306用户登录核验 85](#_Toc488666281)

[3.28 身份核验信息上传 87](#_Toc488666282)

[3.29 12306注册—账号校验 89](#_Toc488666283)

[3.30 12306注册—信息核验 90](#_Toc488666284)

[3.31 12306注册—提交 92](#_Toc488666285)

[3.32 预定下单12306排队状态推送 94](#_Toc488666286)

[3.33 订单状态推送供应商 95](#_Toc488666287)

[4. 附录 97](#_Toc488666288)

[4.1 API返回码 97](#_Toc488666289)

[4.2完整请求示例 97](#_Toc488666290)

[4.3 自定义数据类型说明 99](#_Toc488666291)

## 概述

### 目的

本文档目的是让合作商可以更方便快捷的接入API系统，同时为开发人员提供帮助手册，了解API接入过程中的注意点，以及如何进行API接入。

### API术语

|  |  |
| --- | --- |
| 名称 | 解释 |
| API | Application Programming Interface应用程序接口 |
| account | 分配给途牛的接入账号 |
| passWord | 分配给途牛的接入密码 |
| key | 分配给途牛的密钥 |
| sign | 签名，用来防止请求被伪造或篡改 |

## API调用详解

### 请求流程

### 2.2数据格式

#### 2.2.1 请求入参格式

API调用参数分为系统参数和业务参数，请求时，系统参数是必传的，否则无法成功请求，业务参数由具体业务接口定义,所有请求内容格式为JSON,编码为utf-8。

所有火车票接口在传入消息里都必需包括这几个属性

|  |  |  |  |
| --- | --- | --- | --- |
| **系统参数** | | | |
| 名称 | 类型 | 必填 | 描述 |
| account | String | 是 | 途牛接入账号 |
| sign | String | 是 | 请求签名，生成规则参见[2.4.3 签名机制](#_2.3.3_签名机制) |
| timestamp | String | 是 | 时间戳，参见[2.4.2 时间戳](#_2.3.2_时间戳) |
| data | Object | 否 | 请求参数内容，此字段中的信息密文加密，见2.4 |

#### 2.2.2异步回调请求格式

同步返回如果是失败，则不需要再异步回调。

API请求返回结果目前支持json格式,消息格式字段如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 返回结果 | | | |
| 名称 | 类型 | 必填 | 描述 |
| account | String | 是 | 途牛接入账号 |
| sign | String | 是 | 请求签名，生成规则参见2.4.3 签名机制 |
| timestamp | String | 是 | 时间戳，参见2.4.2 时间戳 |
| returnCode | int | 是 | 结果码，具体值参见[API返回码](#_API返回码) |
| errorMsg | String | 否 | 异常时错误信息 |
| data | Object | 否 | 此字段中的信息密文加密，见2.4 |

#### 2.2.3 同步返回参数格式

同步返回为明文传输

|  |  |  |  |
| --- | --- | --- | --- |
| 返回结果 | | | |
| 名称 | 类型 | 必填 | 描述 |
| success | Boolean | 否 | 保留字段，固定为true |
| returnCode | int | 是 | 结果码，具体值参见[API返回码](#_API返回码) |
| errorMsg | String | 否 | 异常时错误信息 |
| data | Object | 否 | 返回结果 |

#### 2.2.4 异步回调同步返回格式

异步回调同步返回为明文传输

|  |  |  |  |
| --- | --- | --- | --- |
| 返回结果 | | | |
| 名称 | 类型 | 必填 | 描述 |
| success | boolean | true | 结果标示 |

### 2.3应用环境

#### 2.3.1 正式环境

地址：{API\_Url}

#### 2.3.2测试环境

用于联调时进行接口测试。

地址：{API\_UrlTest}

### 2.4安全机制

#### 2.4.1 时间戳

时间戳，格式为yyyy-MM-ddHH:mm:ss，例如：2015-07-30 12:34:56。请求时会校验时间戳，只允许时间误差在5分钟之内的请求访问。

#### 2.4.2 签名机制

为了防止用户请求被劫持、篡改，我们通过签名手段来确保请求数据的完整性。用户在调用API 时需要对请求参数进行签名验证，请求到达服务器后也会对请求参数进行验证，当生成的签名不匹配时则表示数据被拦截篡改或丢失。由于这个机制，合作商需要保存好自己的密钥，一旦泄露，则表示请求可以被伪造。如发现密钥可能泄露，请联系相关接口人重置密钥。

所有类型的请求，签名生成规则都一样，我们将所有入参按照一定规则进行拼装，再加上密钥，经过加密最终生成一串不可逆的加密字符串。为了方便毕竟，我们将密钥统一转成大写字符。生成签名时，需要注意空值的参数不参与校验，且参数名称和值大小写敏感。签名具体步骤参考附录案例[4.2.1 签名生成流程](#_5.2.1_签名生成流程),业务参数中有加密数据的，签名按照加密之后的数据进行签名生成。

#### 2.4.3请求IP限制

合作商申请API接入时，需要将途牛的对外IP作为account绑定的IP白名单。当请求发生时，如果请求IP不是该account的IP白名单内，则请求失败。该限制的目的是为了防止合作商身份泄露导致的恶意访问。

#### 2.4.4消息加密

针对传输过程的敏感信息进行加密处理，加密格式：BASE64(DES (参数明文, key))

详细说明：

业务参数在传输之前是明文的JSON对象，例如：

“data” :{

“vendorOrderId”: “testVendorOrderId”,

“cheCi”: “K1237”,

“fromStationCode”: “BZH”,

…

“passengers”: [

{

“passengerId”: 33292562,

“ticketNo”: null,

“passengerName”: “张三”,

“passportNo”: “341222197608254405”,

“passportTypeId”: “1”,

“passportTypeName”: “二代身份证”,

“piaoType”: “1”,

…

}

]

}

加密之后整个消息结构如下：

{

“account”: “Ape2hqlBF0sFUUcjbj”,

“timestamp”: “2015-07-3012: 34: 56”,

“sign”: “C81E00F2CCC228DD8C1D3EB6C69B4260”,

“data”: “base64ResultTest”

}

### 2.5注意事项

1. 所有的请求和响应数据编码皆为utf-8格式

2. 生成签名时，空值的参数不参与校验

3. 生成签名时，参数名称和值大小写敏感

## 火车票业务接口

说明：以下所有接口出/入参都只包含相应业务类参数，系统参数不单独给出。参数中所有非基本类型的字段类型，均在附录->自定义类型描述中给出相应解释。

### 3.1余位查询（必要）

#### 3.1.1接口说明

此接口用于查询指定日期的出发站与到达站之间的剩余车票数量。

#### 3.1.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/search

DataType：json

#### 3.1.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| trainDate | string | Y |  | 乘车日期（yyyy-MM-dd） |
| fromStation | string | Y |  | 出发站简码 |
| toStation | string | Y |  | 到达站简码 |
| trainCode | string | N |  | 车次号(提供车次号时仅返回车次号对应的余票信息) |

#### 3.1.4接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| saleDateTime | string | Y |  | 车票开售时间 |
| canBuyNow | string | Y |  | 当前是否可以接受预定（Y:可以，N:不可以） |
| arriveDays | int | Y |  | 列车从出发站到达目的站的运行天数 0：当日到达，1：次日到达，2：三日到达，3：四日到达，依此类推 |
| trainStartDate | string | Y |  | 列车从始发站出发的日期 |
| trainCode | string | Y |  | 车次 |
| accessByIdcard | string | Y |  | 是否可凭二代身份证直接进出站 |
| trainNo | string | Y |  | 列车号 |
| trainType | int | Y |  | 列车类型 0高铁，1城际特快，2动车组，3直达，4特快，5普快，6其他 |
| fromStationName | string | Y |  | 出发车站名 |
| fromStationCode | string | Y |  | 出发车站简码 |
| toStationName | string | Y |  | 到达车站名 |
| toStationCode | string | Y |  | 到达车站简码 |
| startStationName | string | Y |  | 列车始发站名 |
| endStationName | string | Y |  | 列车终到站名 |
| startTime | string | Y |  | 出发时刻 |
| arriveTime | string | Y |  | 到达时刻 |
| runTime | string | Y |  | 历时（从出发站到目的站的列车运行时间） |
| runTimeMinute | int | Y |  | 历时分钟合计 |
| gjrwNum | int | Y |  | 高级软卧余票数量 |
| gjrwPrice | string | Y |  | 高级软卧票价（此返回中所有price字段值含义：空字符串表示无此坐等， 数字表示有此坐） |
| qtxbNum | int | N |  | 其他席别余票数量(此返回中所有数量字段含义: 空字符或—表示无此坐等，数字表示余位数量) |
| qtxbPrice | string | N |  | 其他席别对应的票价 |
| rwNum | int | Y |  | 软卧余票数量 |
| rwPrice | string | Y |  | 软卧票价 |
| rzNum | int | Y |  | 软座的余票数量 |
| rzPrice | string | Y |  | 软座的票价 |
| swzNum | int | Y |  | 商务座的余票数据 |
| swzPrice | string | Y |  | 商务座票价 |
| tdzNum | int | Y |  | 特等座的余票数量 |
| tdzPrice | string | Y |  | 特等座票价 |
| wzNum | int | Y |  | 无座的余票数量 |
| wzPrice | string | Y |  | 无座票价 |
| ywNum | int | Y |  | 硬卧的余票数量 |
| ywPrice | string | Y |  | 硬卧票价 |
| yzNum | int | Y |  | 硬座的余票数量 |
| yzPrice | string | Y |  | 硬座票价 |
| edzNum | int | Y |  | 二等座的余票数量 |
| edzPrice | string | Y |  | 二等座票价 |
| ydzNum | int | Y |  | 一等座的余票数量 |
| ydzPrice | string | Y |  | 一等座票价 |

#### 3.1.5接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“trainDate”: “2015-08-29”,

“fromStation”: “NJH”,

“toStation”: “SHH”,

“trainCode”: “”

}

}

#### 3.1.6接口出参示例

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: [

{

“trainNo”: “1100000K7507”,

“trainCode”: “K75”,

“startStationName”: “CCT”,

“endStationName”: “NGH”,

“fromStationCode”: “NJH”,

“fromStationName”: “南京”,

“toStationCode”: “SNH”,

“toStationName”: “上海南”,

“startTime”: “00:58”,

“arriveTime”: “04:37”,

“arriveDays”: “0”,

“untime”: “03:39”,

“canBuyNow”: “Y”,

“runTimeMinute”: “219”,

“trainStartDate”: “20150827”,

“accessByIdcard”: “Y”,

“saleDateTime”: “1630”,

“gjrwNum”: “—“,

“gjrwPrice”: “”,

“qtxbNum”: “—“,

“qtxbPrice”: “”,

“rwNum”: “0”,

“rwPrice”: “140.5”,

“rzNum”: “—“,

“rzPrice”: “”,

“tdzNum”: “—“,

“tdzPrice”: “”,

“wzNum”: “59”,

“wzPrice”: “46.5”,

“ywNum”: “16”,

“ywPrice”: “92.5”,

“yzNum”: “13”,

“yzPrice”: “46.5”,

“edzNum”: “—“,

“edzPrice”: “”,

“ydzNum”: “”,

“ydzPrice”: “”,

“swzNum”: “”,

“swzPrice”: “”

}

]

}

#### 3.1.7通用输出参数增加可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 201 | 没有符合条件的车次信息 |
| 202 | 查询失败 |

### 3.2占位（必要）

#### 3.2.1接口说明

此接口用于申请分配座位席别信息。

此接口同步返回请直接返回成功，调用结果以异步反馈的结果为准

#### 3.2.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/book

DataType：json

#### 3.2.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderId | String | Y |  | 途牛订单号 |
| cheCi | String | Y |  | 车次 |
| fromStationCode | String | Y |  | 出发站简码 |
| fromStationName | String | Y |  | 出发站名称 |
| toStationCode | String | Y |  | 到达站简码 |
| toStationName | String | Y |  | 到达站名称 |
| trainDate | String | Y |  | 乘车日期 |
| callBackUrl | String | Y |  | 锁票异步回调地址[选填],此属性有值则需要回调此地址 |
| hasSeat | Boolean | Y |  | 是否出无座票  true:不出无座票  false:允许出无座票 |
| passengers | String | Y |  | 乘客信息的json字符串。可以是多个乘客信息，最多5个，如：[{乘客1信息},{乘客2信息},…]，也可以只有一个，[{乘客1信息}]。乘客参数见附注1。重要提示：不能只购买儿童票，如果购买儿童票，必须使用随行成人的成人票证件信息（包括姓名、证件号码）。 |
| Contact | String | Y |  | 联系人姓名 |
| phone | String | Y |  | 联系人手机 |
| userName | String | N |  | 12306用户名 |
| userPassword | String | N |  | 12306密码 |
| insureCode | String | N |  | 保险产品编号（有值，则表示此单购买保险） |
| isChooseSeats | boolean | N | true | 是否选座, true:选, false:非选 |
| chooseSeats | String | N | 1A1D2B2C2F | 选座信息(选座个数要和乘客数量一致) |

#### 3.2.4接口出参

##### 3.2.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | String | 是 | 15538430 | 合作伙伴方订单号 |

##### 3.2.4.2异步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | String |  |  | 合作伙伴方订单号 |
| orderId | String |  |  | 途牛订单号 |
| orderSuccess | Bool |  |  | 订票是否成功 |
| orderAmount | String |  |  | 订单总金额 |
| cheCi | String |  |  | 车次 |
| fromStationCode | String |  |  | 出发站简码 |
| fromStationName | String |  |  | 出发站名称 |
| toStationCode | String |  |  | 到达站简码 |
| toStationName | String |  |  | 到达站名称 |
| trainDate | String |  |  | 乘车日期 |
| startTime | String |  |  | 从出发站开车时间 |
| arriveTime | String |  |  | 抵达目的站的时间 |
| orderNumber | String |  | E010400222 | 12306取票号 |
| passengers | String |  |  | 与输入的一样，订票成功了里面的cxin和ticketid参数会有值。 |
| paymentUrls | PaymentUrl[] | N |  | 支付链接 |
| clearTime | String | N | 2017-10-26 13:00:00 | 清位时间（例2017-10-26 13:00:00） |

**PaymentUrl**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| type | Integer | N |  | 1、支付宝 2、银联 |
| url | String | N |  | 支付链接 |
| validStartTime | String | N |  | 有效的起始时间 |

#### 3.2.5接口入参示例

下单接口需要进行加密，请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数样例如下：

{

"account": "testAccount",

"sign": "testsign",

"timestamp": "2015-08-0300: 00: 00",

"data": {

"vendorOrderId": "testVendorOrderId",

"cheCi": "K1237",

"fromStationCode": "BZH",

"fromStationName": "亳州",

"toStationCode": "RZH",

"toStationName": "温州",

"trainDate": "2015-08-29",

"callBackUrl": "{途牛URL}/train/bookOrderFeedback",

"contact": "张三",

"phone": "13728784623",

"insureCode": "123",

"isChooseSeats": false,

"chooseSeats": "",

"passengers": [

{

"passengerId": 33292562,

"ticketNo": null,

"passengerName": "张三",

"passportNo": "341222197608254405",

"passportTypeId": "1",

"passportTypeName": "二代身份证",

"piaoType": "1",

"piaoTypeName": "成人票",

"zwCode": "1",

"zwName": "硬座",

"cxin": null,

"price": "1234",

"reason": 0,

"provinceCode": null,

"schoolCode": null,

"schoolName": null,

"studentNo": null,

"schoolSystem": null,

"enterYear": null,

"preferenceFromStationName": null,

"preferenceFromStationCode": null,

"preferenceToStationName": null,

"preferenceToStationCode": null

}

],

"paymentUrls": [

{

"type": 1,

"url": "http://10.10.32.160/fb2/t1/group1/M00/05/33/Cgogo1mNJJWIeaMJAAAKe-fantgAAABSQODVM4AAAqT61.html",

"validStartTime": "2017-09-19 16:08:23"

},

{

"type": 2,

"url": "http://10.10.32.160/fb2/t1/group1/M00/05/33/Cgogo1mNJJWIeaMJAAAKe-fantgAAABSQODVM4AAAqT61.html",

"validStartTime": "2017-09-19 16:08:23"

}

]

}

}

#### 3.2.6接口出参示例

##### 3.2.6.1同步反馈

{

“success”: true,

“returnCode”: 301,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”

}

}

##### 3.2.6.2异步反馈

此返回需要加密，反馈请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数示例如下：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”,

“orderSuccess”: true,

“orderAmount”: 305,

“cheCi”: “K1237”,

“fromStationCode”: “BZH”,

“fromStationName”: “亳州”,

“toStationCode”: “RZH”,

“toStationName”: “温州”,

“trainDate”: “2015-08-29”,

“startTime”: “15:57:00”,

“arriveTime”: “11:40:00”,

“orderNumber”: “E010400222”,

“clearTime”:”2017-10-26 15:00:00”

“passengers”: [

{

“reason”: 0,

“price”: “152.5”,

“passengerId”: 33292562,

“ticketNo”: “E0104002221053148”,

“zwCode”: “1”,

“cxin”: “05车厢,无座”,

“passportTypeName”: “二代身份证”,

“passportNo”: “341222197608254405”,

“zwName”: “硬座”,

“piaoType”: “1”,

“passengerName”: “张三”,

“passportTypeId”: “1”,

“piaoTypeName”: “成人票”

}

]

}

}

#### 3.2.7通用输出参数增加可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 110000 | 没有余票 |
| 110001 | 乘客信息有误 |
| 110002 | 行程冲突 |
| 110003 | 乘客身份信息{姓名}{证件号码}未通过验证 |
| 110004 | 距离开车时间太近 |
| 110005 | 乘客身份信息{姓名}{证件号码}被冒用 |
| 110006 | 您的12306账号未进行手机号码核验，请前往12306网站/app进行手机号码核验，核验通过之后即可正常购票，很抱歉给您带来了不便，敬请谅解。 |
| 110008 | 您的身份信息{姓名}{证件号码}正在进行网上核验，您也可持居民身份证原件到车站售票窗口即时办理核验。 |
| 110009 | 夜间无法下单 |
| 110010 | 当前的排队购票人数过多，请稍后重试 |
| 110011 | 该12306账号所添加的常旅数量已达到上限 |
| 110012 | 限制高消费 |
| 110013 | 存在未完成订单 |
| 110014 | 取消次数过多 |
| 110015 | 用户已在其他地点登录 |
| 110016 | 订单正在预定，请勿重复提交 |
| 110017 | 该车次未到起售时间，请稍后再试 |
| 110018 | 车站三字码信息有误 |
| 110019 | 暂不支持{学生票}的车票类别 |
| 110020 | 途牛账户余额不足 |
| 110021 | 12306暂未查询到该车次信息或车次已经停运 |
| 110022 | 12306账号密码错误 |
| 110023 | 12306账号已被锁定，请稍后再试 |
| 110024 | 系统异常,操作失败 |

**注**：注意大括号区域的文本，如果一个订单中包含多个乘客，并且多个乘客身份信息无法购票，那么只需要使用其中一位乘客的信息替换即可（大括号不需要保留）。务必按此格式返回。

### 3.3出票（必要）

#### 3.3.1接口说明

此接口是指途牛得到订票成功（成功分配到了座位席别信息）的反馈后，在规定的时间（比如30分钟）内发出确认出票请求。

此接口同步返回请直接返回成功，调用结果以异步反馈的结果为准

途牛测配置等待异步反馈最大时限（目前是30分钟），超过最大时限供应商未反馈

途牛侧设置订单状态为出票失败，并通过订单状态推送接口将订单状态推送供应商

以便供应商及时更新处理出票结果

#### 3.3.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/confirm

DataType：json

#### 3.3.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| callBackUrl | string | Y |  | 回调URL地址, 此属性有值则需要回调此地址 |
| isProxyPay | Boolean | 是 |  | true：套皮支付 false：非套皮支付 |

#### 3.3.4接口出参

##### 3.3.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |

##### 3.3.4.2异步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | 是 |  | 合作伙伴方订单号 |
| orderId | string | 是 |  | 途牛订单号 |
| ticketEntrances | List<object> | 否 |  | 检票口信息 |

TicketEntrance

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| trainNum | string | 是 |  | 车次 |
| stationName | string | 是 |  | 站点名称 |
| entrance | string | 是 |  | 检票口 |

#### 3.3.5接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”,

“callBackUrl”: “{途牛URL}/train /confirmOrderFeedback”,

“isProxyPay”:false

}

}

#### 3.3.6接口出参示例

##### 3.3.6.1同步接口出参

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AEME3TJS1H00”

}

}

##### 3.3.6.2异步接口出参

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00: 00: 00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AEME3TJS1H00”,

“ticketEntrances”: [

{

“trainNum”: “K99”,

“stationName”: “北京”,

“entrance”: “19号”

}

]

}

}

#### 3.3.7通用输出参数增加可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 120000 | 出票失败 |
| 120001 | 出票成功 |
| 120002 | 支付超时，12306已经自动取消，出票失败 |
| 120003 | 途牛账户余额不足 |
| 120004 | 乘客身份信息{姓名}{证件号码}未通过验证 |
| 120005 | 您已在12306取消订单，出票失败 |
| 120006 | 12306账号密码错误 |
| 120007 | 12306账号已被锁定，请稍后再试 |
| 120008 | 系统异常,操作失败 |

### 3.4取消占位（必要）

#### 3.4.1接口说明

此接口是指途牛得到订票成功的反馈后发出请求，合作方收到请求将此火车票订单取消。

此接口同步返回请直接返回成功，调用结果以异步反馈的结果为准

#### 3.4.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/cancel

DataType：json

#### 3.4.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| callBackUrl | string | Y |  | 回调URL地址, 此属性有值则需要回调此地址 |

#### 3.4.4接口出参

##### 3.4.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |

##### 3.4.4.2异步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |

#### 3.4.5接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”,

“callBackUrl”: “{途牛URL}/train /cancelOrderFeedback”

}

}

#### 3.4.6接口出参示例

##### 3.4.6.1 同步接口出参示例

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “15274218T7650055”

}

}

##### 3.4.6.2异步接口出参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “15274218T7650055”

}

}

#### 3.4.7通用输出参数增加可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 130000 | 订单已出票，不能取消 |
| 130001 | 取消失败 |
| 130002 | 取消成功 |
| 130003 | 订单已经取消成功，请勿重复提交 |
| 130004 | 订单正在取消，请勿重复提交 |
| 130005 | 当前的订单状态不能取消订单 |
| 130006 | 支付超时，12306已经自动取消 |
| 130007 | 乘客身份信息{姓名}{证件号码}未通过验证 |
| 130008 | 12306账号密码错误 |
| 130009 | 12306账号已被锁定，请稍后再试 |
| 130010 | 系统异常,操作失败 |

### 3.5线上退票（必要）

#### 3.5.1接口说明

我方每次可对同一订单的多张车票进行退票处理。即只要所退车票是同一订单内，就可以一次性进行退票处理。根据相关规定，我方只能在不晚于开车前3小时对尚未取票的车票办理网上退票。另外，依据规定，将对每张车票按梯次收取退票手续费，最终退票手续费以铁路局实际收取为准，退票款项将会退到途牛的账户余额当中。

线上退票入参和异步返回需要进行数据加密参见2.4.5消息加密。

#### 3.5.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/return

DataType：json

#### 3.5.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| orderNumber | string | Y |  | 取票单号 |
| userName | String |  |  | 12306用户名 |
| userPassword | String |  |  | 12306密码 |
| callBackUrl | string | N |  | 回调URL地址, 此属性有值则需要回调此地址 |
| tickets | string | Y |  | 车票信息（json字符串数组形式，主要包含车票的乘车人信息，乘车人姓名、乘车人证件类型ID和乘车人证件号码，如：[{“ticketNo”:“E2610890401070051”,“passengerName”:“王二”,“ passportTypeId”:1,“passportNo”:“421116198907143795” }] |
| refundId | string | Y |  | 退票请求流水号 |

Ticket

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| ticketNo | string | Y |  | 车票编号 |
| passengerName | string | Y |  | 乘客姓名 |
| passportTypeId | string | Y |  | 乘客证件类型 |
| passportNo | string | Y |  | 乘客证件号码 |

#### 3.5.4接口出参

##### 3.5.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderNumber | string | Y |  | 取票单号 |

##### 3.5.5.2异步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId |  | Y |  | 合作伙伴方订单号 |
| orderNumber |  | Y |  | 火车票取票单号 |
| returnTickets |  | N |  | 车票退票信息(json字符串数组形式，每张车票包含乘车人信息和退票相关信息，如：  [“ticketNo”:” E2610890401070051”,”passengerName”:”王二”,”passportTypeId”:1,”passportNo”:”421116198907143795”,”returnSuccess”:true,”returnMoney”:”20.05”,”returnTime”:”2015-02-13 15:00:05”,”returnFailId”:””,”returnFailMsg”:””]] |
| returnState |  | Y |  | 退票状态 true:表示成功 false:表示退票失败 |
| returnMoney |  | N |  | 退款金额（成功才有值）线上退票时，此值为退款总额 |
| returnMsg |  | N |  | 退票后消息描述（一般当returnState=false时，会显示退票失败原因等） |
| refundId |  | Y |  | 退票请求流水号（回传请求的值） |

其中的returnTickets中每张票的退票信息如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| ticketNo | string | Y |  | 票号 |
| passengerName | string | Y |  | 乘客姓名 |
| passportTypeId | int | Y |  | 证件类别编号 |
| passportNo | string | Y |  | 证件号码 |
| returnSuccess | bool | Y |  | 是否退票成功（true:成功，false:失败） |
| returnMoney | string | N |  | 退款金额（成功才有值） |
| insureReturnMoney | string | N |  | 保险退款金额（如果购买了保险） |
| returnTime | string | N |  | 退票成功的时间（成功才有值） |
| returnFailId | string | N |  | 退票失败原因编号（失败才有值） |
| returnFailMsg | string | N |  | 退票失败原因信息（失败才有值） |
|  |  |  |  |  |

失败码号及提示信息

|  |  |
| --- | --- |
| returnFailId | returnFailMsg |
| 0 | 退票操作成功[成功时，此值为空] |
| 1 | 未查询到该车票 |
| 2 | 退票操作异常，请与客服联系 |
| 3 | 已改签 |
| 4 | 已退票 |
| 5 | 已出票，只能在窗口办理退票 |
| 6 | 不可退票 |
|  |  |

在提供的回调URL中不发生任何跳转，收到请求后同步返回结果，表明成功处理。

#### 3.5.5接口入参示例

此接口需要加密，请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数示例如下：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“refundId”: “123”,

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”,

“orderNumber”: “E010400222”,

“callBackUrl”: “{途牛URL}/train/returnTicketFeedback”,

“tickets”: [

{

“ticketNo”: “E0104002221053148”,

“passengerName”: “张三”,

“passportTypeId”: “1”,

“passportNo”: “341222197608254405”

}

]

}

}

#### 3.5.6接口出参示例

1.同步反馈

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderNumber”: “E010400222”

}

}

1. 异步反馈

此反馈参数需要加密，请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数示例如下：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“refundId”: “123”,

“vendorOrderId”: “testVendorOrderId”,

“orderNumber”: “E010400222”,

“returnTickets”: [

{

“ticketNo”: “E0104002221053148”,

“passengerName”: “张三”,

“passportTypeId”: 1,

“passportNo”: “341222197608254405”,

“returnSuccess”: “True”,

“returnFailId”: “0”,

“returnTime”: “2015-08-29 10:05:54”,

“returnMoney”: “122.00”

}

],

“returnState”: true,

“returnMoney”: “122.00”,

“returnMsg”: “”

}

}

#### 3.5.7通用输出参数增加可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 140000 | 距离开车时间太近无法退票 |
| 140001 | 退票成功 |
| 140002 | 退票失败 |
| 140003 | 已取纸制票，不能退票 |
| 140004 | 订单已经退票成功，请勿重复提交 |
| 140005 | 订单正在退票，请勿重复提交 |
| 140006 | 当前时间不支持退票业务，请稍后再试 |
| 140007 | 尊敬的旅客，为防止网上囤票倒票，给广大旅客创造一个公平的购票环境，凡通过互联网或手机购买的本次列车车票，如需办理退票、改签和变更到站等变更业务，请持乘车人身份证件原件到就近车站办理，代办时还需持代办人的身份证件原件 |
| 140008 | 乘客身份信息{姓名}{证件号码}未通过验证 |
| 140009 | 12306账号密码错误 |
| 140010 | 12306账号已被锁定，请稍后再试 |
| 140011 | 系统异常,操作失败 |

### 3.6线下退款通知（必要）

#### 3.6.1接口说明

本接口的应用场景是：购票人到车站已经取票，然后拿票到窗口去退票，退票的工作人员会和他说这个票款会退到他们的网银里面。票款信息经过合作方返回给途牛，由途牛退款给客户。

#### 3.6.2接口调用方法

Method：HTTP-POST

Url：途牛指定

DataType：json

#### 3.6.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| returnState | boolean | Y |  | 退票状态 true:表示成功 false:表示退票失败 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| orderNumber | string | Y |  | 取票单号 |
| tickets | string | N |  | 车票退票信息(json字符串数组形式，每张车票包含乘车人信息和退票相关信息，如：  [“ticketNo”:” E2610890401070051”,”passengerName”:”王二”,”passportTypeId”:1,”passportseNo”:”421116198907143795”,”returnSuccess”:true,”returnMoney”:”20.05”,”returnTime”:”2014-02-13 15:00:05”,”returnFailId”:””,”returnFailMsg”:””]]  注：保留字段，当为线下退票时，此值为空 |
| returnMoney | string | Y |  | 退款总额 |
| returnMsg | String | N |  | 退票后消息描述 |

#### 3.6.4接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| success | string | Y |  | 是否成功 |

#### 3.6.5接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“returnState“: true,

“vendorOrderId”: “4HEE0TS42400”,

“orderId”: “5ECE0TS42401”,

“orderNumber”: “E262577575”,

“tickets”: [

{

“ticketNo”: “E967421083110004D”,

“passengerName”: null,

“passportTypeseId”: 0,

“passportNo”: null,

“returnSuccess”: null,

“returnMoney”: null,

“returnTime”: null,

“returnFailId”: null,

“returnFailMsg”: null

}

],

“returnMoney”: “ 20.05”,

“returnMsg”: “线下退款”

}

}

#### 6.6接口出参示例

{

“success”: true

}

#### 3.6.7通用输出参数增加可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 701 | 收到报告信息。 |

### 3.7查询经停站

#### 3.7.1接口说明

查询某车次的经停站信息

#### 3.7.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/queryStations

DataType：json

#### 3.7.3 接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| trainDate | string | Y |  | 乘车日期（yyyy-MM-dd） |
| trainCode | string | Y |  | 车次号 |

#### 3.7.4 接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| trainDate | string | Y |  | 乘车日期（yyyy-MM-dd） |
| trainCode | string | Y |  | 车次号 |
| stations | list | Y |  |  |

stations:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| stationId | int | Y | 220 | 站点ID |
| stationName | string | Y | 南京南 | 站点名称 |
| stationNum | int | Y | 1 | 车站序号 |
| departTime | string | Y | 19:30 | 出站时间 |
| arriveTime | string | Y | 19:20 | 到站时间 |
| stayTime | string | Y | 10分钟 | 停留时间 |

#### 3.7.5 接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“trainDate”: “2015-10-08”,

“trainCode”: “G11”

}

}

#### 3.7.6 接口出参示例

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“trainDate”: “2015-10-08”,

“trainCode”: “G11”,

“stations”: [

{

“stationId”: “220”,

“stationName”: “南京南”,

“stationNum”: “1”,

“departTime”: “19:30”,

“arriveTime”: “----“,

“stayTime”: “始发站”

},

{

“stationId”: “221”,

“stationName”: “镇江”,

“stationNum”: “2”,

“departTime”: “19:50”,

“arriveTime”: “19:40”,

“stayTime”: “10分钟”

},

{

“stationId”: “222”,

“stationName”: “上海虹桥”,

“stationNum”: “3”,

“departTime”: “20:00”,

“arriveTime”: “20:00”,

“stayTime”: “终点站”

}

]

}

}

#### 3.7.7 通用输出参数可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 801 | 查询失败 |
| 802 | 无此车次 |

### 3.8身份证核验

#### 3.8.1接口说明

197046366

身份证验证接口

#### 3.8.2 接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/validate

DataType：json

#### 3.8.3 接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| name | string | Y | 张三 | 姓名 |
| identityType | int | Y | 1 | 证件类型 1、身份证 |
| identityCard | string | Y | 365521000254456642 | 证件号码 |

#### 3.8.4 接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| identityCard | string | Y |  | 证件号码 |
| name | string | Y | 张三 | 姓名 |
| identityType | int | Y |  | 证件类型 |
| isPass | int | Y |  | 是否通过 0 、待核验 1、核验通过2核验失败3冒用4其他失败类型 |
| error | string | N |  | 原因,通过时此值为空字符 |

#### 3.8.5 接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: [

{

“name”: “张三”,

“identityType”: 1,

“identityCard”: “365521000254456642”

}

]

}

#### 3.8.6 接口出参示例

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: [

{

“identityCard”: “365521000254456642”,

“name”: “张三”,

“identityType”: 1,

“isPass”: 1,

“error“: “身份证未通过核验”

}

]

}

#### 3.8.7 通用输出参数可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 901 | 系统异常 |

**3.9 12306账号验证**

**3.9.1接口说明**

12306账号验证接口

**3.9.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/trainAccount/validate

DataType：json

**3.9.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| trainAccount | string | Y | 222 | 12306账号 |
| pass | string | Y | 1 | 密码 |

**3.9.4 接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| isPass | int | Y | 0 | 是否通过 0 、通过 1、不通过 |

**3.9.5 接口入参示例**

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-0300: 00: 00”,

“data”: {

“trainAccount”: “222”,

“pass”: “111”

}

}

**3.9.6 接口出参示例**

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”:

{

“ isPass“: 0

}

}

**3.9.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 901 | 系统异常 |

**备注：**

查询失败了需要在errorMsg备注具体的失败原因

例如

1. 登录名存在等
2. 账号已锁定，请20分钟之后重新登录

**3.10 查询常用联系人**

**3.10.1接口说明**

查询常用联系人接口

入参和出参Data部分都需要加密，整个报文不需要签名

**3.10.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/trainAccount/contact/query

DataType：json

**3.10.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| trainAccount | string | Y | 222 | 12306账号 |
| pass | string | Y | 1 | 密码 |

**3.10.4 接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| id | int | Y |  | 序列号 |
| name | string | Y |  | 姓名 |
| isUserSelf | int | Y |  | 是否是账号本人 0是 1不是 |
| sex | int | N |  | 性别 0 男 1 女 |
| birthday | string | N | 2015-09-05 | 生日 |
| country | string | N | CN | 国家 CN中国 |
| identyType | string | Y | 0 | 证件类型 见附录4.3 |
| identy | string | Y |  | 证件号码 |
| personType | int | Y |  | 旅客类型 0 成人 1儿童 2学生 |
| phone | string | N |  | 手机 |
| tel | string | N |  | 固话 |
| email | string | N |  | 邮件 |
| address | string | N |  | 地址 |
| checkStatus | int | Y |  | 身份核验状态   1. 已通过 2. 未通过 |
| addTime | string | Y |  | 添加时间（yyyy-mm-dd） |
| provinceName | string | N |  | 学校所在省份名称 |
| schoolName | string | N |  | 学校名称 |
| department | string | N |  | 院系 |
| className | string |  |  | 班级 |
| studentNo | string | N |  | 学号 |
| schoolSystem | string | N |  | 学制 |
| enterYear | string | N |  | 入学年份 |
| preferenceNum | string | N |  | 优惠卡号 |
| preferenceFromStationName | string | N |  | 优惠区间起始地名称 |
| preferenceToStationName | string | N |  | 优惠区间到达地名称 |

**3.10.5 接口入参示例**

入参密文如下

{

“account”: “testAccount”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: ‘’Base64ResultTest

}

入参明文如下

{

“account”: “testAccount”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“trainAccount”: “222”,

“pass”:”111”

}

}

**3.10.6 接口出参示例**

出参密文如下

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”:” Base64ResultTest”

}

出参明文如下

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: [

{

“id”: 1,

“name”: “张三”,

“isUserSelf”: 1,

“sex”: 0,

“birthday”: “2015-09-08”,

“country”: “CN”,

“identyType”: “1”,

“identy”: “36542”,

“personType”: 1,

“phone”: “16235555555”,

“tel”: “025-8888888”,

“email”: “test@163.com”,

“address”: “江苏省南京市”,

“checkStatus”: 0,

“addTime”: “2017-01-09”,

“provinceName”: “”,

“schoolName”: “”,

“department”: “”,

“className”: “”,

“studentNo”: “”,

“schoolSystem”: “”,

“enterYear”: “”,

“preferenceNum”: “”,

“preferenceFromStationName”: “”,

“preferenceToStationName”: “”

}

]

}

**3.10.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 1001 | 查询失败 |

**3.11 增加和修改常用联系人**

**3.11.1接口说明**

增加和修改常用联系人接口

入参Data部分需要加密，整个报文不需要签名

**3.11.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/trainAccount/contact/saveOrUpdate

DataType：json

**3.11.4 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| trainAccount | string | Y | 222 | 12306账号 |
| pass | string | Y | 1 | 密码 |
| contacts | Contact | Y |  | 联系人列表 |

Contact:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| id | int | Y |  | 序列号（0表示新增此联系人，>0表示修改此联系人） |
| name | string | Y |  | 姓名 |
| sex | int | N |  | 性别 0 男 1 女 |
| birthday | string | N | 2015-09-05 | 生日 |
| country | string | N | CN | 国家 CN中国 |
| identyType | string | Y | 0 | 证件类型 见附录4.3 |
| identy | string | Y |  | 证件号码 |
| personType | int | Y |  | 旅客类型 0 成人 1儿童 |
| phone | string | N |  | 手机 |
| tel | string | N |  | 固话 |
| email | string | N |  | 邮件 |
| address | string | N |  | 地址 |
|  |  |  |  |  |

**3.11.3 接口出参**

无业务返回参数

**3.11.5 接口入参示例**

入参密文如下

{

“account”: “testAccount”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: “base64ResultTest”

}

入参明文如下

{

“account”: “testAccount”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“trainAccount”: “testAccount”,

“pass”: “222”,

“contacts”: [

{

“id”: 1,

“name”: “张三”,

“sex”: 0,

“birthday”: “2015-09-08”,

“country”: “CN”,

“identyType”: “1”,

“identy”: “36542”,

“personType”: 1,

“phone”: “16235555555”,

“tel”: “025-8888888”,

“email”: “test@163.com”,

“address”: “江苏省南京市”

},

{

“id”: 2,

“name”: “李四”,

“sex”: 0,

“birthday”: “2015-09-09”,

“country”: “CN”,

“identyType”: “1”,

“identy”: “3654235”,

“personType”: 1,

“phone”: “16235555557”,

“tel”: “025-8888888”,

“email”: “test@163.com”,

“address”: “江苏省南京市”

}

]

}

}

**3.11.6 接口出参示例**

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {}

}

**3.11.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 1101 | 新增失败 |
| 1102 | 更新失败 |

**3.12 删除常用联系人**

**3.12.1接口说明**

删除常用联系人接口

入参Data部分需要加密，整个报文不需要签名

**3.12.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/trainAccount/contact/delete

DataType：json

**3.12.4 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| trainAccount | string | Y | 222 | 12306账号 |
| pass | int | Y | 1 | 密码 |
| id | int | N | 1 | 联系人ID |
| identyType | string | Y | 0 | 证件类型 见附录4.3 |
| identy | string | Y |  | 证件号码 |

参数说明

删除联系人既可以依据联系人ID也可以依据身份证件编号，此接口中的联系人ID和证件号码必须有一个非空。如果证件号码非空，则证件类型也不能为空。

**3.12.3 接口出参**

无业务返回参数

**3.12.5 接口入参示例**

入参密文如下

{

“account”: “testAccount”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: “base64ResultTest”

}

入参明文如下

{

“account”: “testAccount”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“trainAccount”: “testsign”,

“pass”: “11”,

“identy”: “3412222134234X”,

“identyType”: “1”

}

}

**3.12.6 接口出参示例**

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

}

}

**3.12.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 1201 | 删除失败 |

**3.13 抢票**

**3.13.1接口说明**

抢票接口

**3.13.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/trainAccount/grab

DataType：json

**3.13.3接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderId | string | Y |  | 途牛订单号 |
| cheCi | string | Y |  | 车次 |
| fromStationCode | string | Y |  | 出发站简码 |
| fromStationName | string | Y |  | 出发站名称 |
| toStationCode | string | Y |  | 到达站简码 |
| toStationName | string | Y |  | 到达站名称 |
| trainDate | string | Y |  | 乘车日期 |
| deadLine | string | Y | 2016-02-28 19:30:00 | 抢票截止时间，  格式yyyy-MM-dd HH:mm:ss |
| reserveCheCi | string[] | N | [“K101”,”K102”] | 备选车次，最多8个，可以为空。但是不能合cheCi字段相同 |
| reserveZwCode | string[] | N | [“1”,”2”] | 备选坐席，可以为空。对所有乘客、所有备选车次有效，但是并不对cheCi字段对应车次有效。 |
| callBackUrl | string | Y |  | 回调地址 |
| hasSeat | boolean | Y |  | 是否出无座票  true:不出无座票  false:允许出无座票 |
| passengers | string | Y |  | 乘客信息的json字符串。可以是多个乘客信息，最多5个，如：[{乘客1信息},{乘客2信息},…]，也可以只有一个，[{乘客1信息}]。乘客参数见附注1。重要提示：不能只购买儿童票，如果购买儿童票，必须使用随行成人的成人票证件信息（包括姓名、证件号码）。 |
| Contact | string | Y |  | 联系人姓名 |
| phone | string | Y |  | 联系人手机 |
| trainAccount | string | N |  | 12306用户名 |
| pass | string | N |  | 12306密码 |
| insureCode | string | N |  | 保险产品编号（有值，则表示此单购买保险） |

**3.13.4接口出参**

**3.13.4.1同步接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y | 15538430 | 合作伙伴方订单号 |

**3.13.4.2异步接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| orderSuccess | bool | Y |  | 订票是否成功 |
| orderAmount | string | Y |  | 订单总金额 |
| cheCi | string | Y |  | 车次 |
| fromStationCode | string | Y |  | 出发站简码 |
| fromStationName | string | Y |  | 出发站名称 |
| toStationCode | string | Y |  | 到达站简码 |
| toStationName | string | Y |  | 到达站名称 |
| trainDate | string | Y |  | 乘车日期 |
| deadLine | string | Y | 2016-02-28 19:30:00 | 抢票截止时间，  格式yyyy-MM-dd HH:mm:ss |
| startTime | string | Y |  | 从出发站开车时间 |
| arriveTime | string | Y |  | 抵达目的站的时间 |
| orderNumber | string | Y | E010400222 | 12306取票号 |
| passengers | string | Y |  | 与输入的一样，订票成功了里面的cxin和ticketid参数会有值。 |
| searchNumber | int | Y |  | 当前订单查询余票次数 |
| grabFailNumber | int | Y |  | 当前订单余票不为0尝试抢票失败次数 |
| concurrentNumber | int | Y |  | 当前订单从抢票开始到抢票结束，同时有多少人在抢 |

**3.13.5接口入参示例**

抢票接口需要进行加密，请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数样例如下：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“cheCi”: “K1237”,

“fromStationCode”: “BZH”,

“fromStationName”: “亳州”,

“toStationCode”: “RZH”,

“toStationName”: “温州”,

“trainDate”: “2015-08-29”,

“sername”: “2015-08-30 20:03:00”,

“callBackUrl”: “{途牛URL}/train /bookOrderFeedback”,

“contact”: “张三”,

“phone”: “13728784623”,

“trainAccount”: “222”,

“pass”: “111”,

“insureCode”: “123”,

“passengers”: [

{

“passengerId”: 33292562,

“ticketNo”: null,

“passengerName”: “张三”,

“passportNo”: “341222197608254405”,

“passportTypeId”: “1”,

“passportTypeName”: “二代身份证”,

“piaoType”: “1”,

“piaoTypeName”: “成人票”,

“zwCode”: “1”,

“zwName”: “硬座”,

“cxin”: null,

“price”: “1234”,

“reason”: 0,

“provinceCode”: null,

“schoolCode”: null,

“schoolName”: null,

“studentNo”: null,

“schoolSystem”: null,

“enterYear”: null,

“preferenceFromStationName”: null,

“preferenceFromStationCode”: null,

“preferenceToStationName”: null,

“preferenceToStationCode”: null

}

]

}

}

**3.13.6接口出参示例**

**3.13.6.1同步反馈**

{

“success”: true,

“returnCode”: 301,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”

}

}

**3.13.6.2异步反馈**

此返回需要加密，反馈请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数示例如下：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”,

“orderSuccess”: true,

“orderAmount”: 305,

“cheCi”: “K1237”,

“fromStationCode”: “BZH”,

“fromStationName”: “亳州”,

“toStationCode”: “RZH”,

“toStationName”: “温州”,

“trainDate”: “2015-08-29”,

“sername”: “2015-08-30 20:03:00”,

“startTime”: “15:57:00”,

“arriveTime”: “11:40:00”,

“orderNumber”: “E010400222”,

“searchNumber”: 20,

“grabFailNumber”: 19,

“concurrentNumber”: 20,

“passengers”: [

{

“reason”: 0,

“price”: “152.5”,

“passengerId”: 33292562,

“ticketNo”: “E0104002221053148”,

“zwCode”: “1”,

“cxin”: “05车厢,无座”,

“passportTypeName”: “二代身份证”,

“passportNo”: “341222197608254405”,

“zwName”: “硬座”,

“piaoType”: “1”,

“passengerName”: “张三”,

“passportTypeId”: “1”,

“piaoTypeName”: “成人票”

}

]

}

}

**3.13.7通用输出参数增加可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 301 | 没有余票 |
| 302 | 乘客信息有误 |
| 303 | 乘客已办理其他订单 |
| 304 | 已经超过未完成订单的授权的数量。此授权的数量在签约时确立。  （未完成订单：已经确定席别但未确认出票的订单） |
| 305 | 乘客已经预订过该车次 |
| 306 | 该车次已经没有此席别的车票了 |
| 307 | 当前提交订单用户过多（服务器繁忙） |
| 308 | 乘客身份信息未通过验证 |
| 309 | 没有足够的票 |
| 310 | 本次购票与其它订单冲突 |
| 311 | 距离开车时间太近 |
| 313 | 抢票已过截止时间 |

**3.14 线下出票**

**3.14.1接口说明**

线下出票接口，此接口中的票要么都成功，要么都失败，不存在部分的情况。

**3.14.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/train/offline/book

DataType：json

**3.14.3接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderId | string | Y |  | 途牛订单号 |
| cheCi | string | Y |  | 车次 |
| fromStationCode | string | Y |  | 出发站简码 |
| fromStationName | string | Y |  | 出发站名称 |
| toStationCode | string | Y |  | 到达站简码 |
| toStationName | string | Y |  | 到达站名称 |
| trainDate | string | Y | 2015-08-09 | 乘车日期 |
| trainStartTime | string | Y | 16:30 | 发车时间 |
| callBackUrl | string | Y |  | 回调地址 |
| seat | seat | Y |  | 坐席（单个坐席） |
| extSeat | seat | N |  | 备选坐席（数组，可以为多个） |
| passengers | passenger | Y |  | 乘客信息 |
| ticketType | int | Y |  | 票类型(0普通,1团体,2下铺,3靠窗,4连坐),注意：目前只有 0和2两种类型可以实现，其他类型为保留类型。 |
| tikcetBakcUp | int | N |  | 当下铺/靠窗/连坐无票时,是否支持非下铺/非靠窗/非连坐(0不接受,1接受)注意：只有下铺类型有效，其他类型此值无效。 |
| minSeatCount | int | N |  | 至少接受下铺/靠窗/连坐数量,注意：只有下铺类型有效，其他类型此值无效 |
| transportName | string | Y |  | 快递收件人姓名 |
| transportPhone | string | Y |  | 快递收件人手机号 |
| transportAddress | string | Y |  | 快递收件人地址 |

Passenger:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| passportNo | string | Y |  | 乘客证件号码 |
| passportTypeId | string | Y |  | 证件类型ID  与名称对应关系:  1:二代身份证，2:一代身份证，C:港澳通行证，G:台湾通行证，B:护照更多参见证件编码对应表 |
| passengerName | string | Y |  | 乘客姓名 |
| ticketType | int | Y |  | 票类型 0成人票 1儿童票 |

Seat:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| seatType | string | Y |  | 见4.3.2 |
| seatName | string | Y |  | 见4.3.2 |
| seatPrice | string | Y |  | 坐席价格（卧铺等传递下铺价格） |

**3.14.4接口出参**

**3.14.4.1同步接口出参**

无业务参数。

**3.14.4.2异步接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| passportNo | string | N |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| failMsg | failMsg[] | N |  | 失败信息 |
| tickets | ticket[] | N |  | 出票信息 |

failMsg

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| passportNo | string | Y |  | 乘客证件号码 |
| passportTypeId | string | Y |  | 证件类型ID  与名称对应关系:  1:二代身份证，2:一代身份证，C:港澳通行证，G:台湾通行证，B:护照更多参见证件编码对应表 |
| passengerName | string | Y |  | 乘客姓名 |
| reason | int | Y |  | 1:身份证待审核2:身份证未通过3:实名制购票路线冲突. |
| preDate | string | N | 2015-08-19 | 冲突的团期，仅在reason为3时有值 |
| preTrainNo | string | N |  | 冲突的车次号，仅在reason为3时有值 |

Ticket:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderNumber | string | Y |  | 12306出票号 |
| seatType | string | Y |  | 见4.3.2 |
| seatNo | string | Y | 13车厢08号 | 车厢车号 |
| price | string |  | 128.00 | 价格 |
| passportNo | string | Y |  | 乘客证件号码 |
| passportTypeId | string | Y |  | 证件类型ID  与名称对应关系:  1:二代身份证，2:一代身份证，C:港澳通行证，G:台湾通行证，B:护照更多参见证件编码对应表 |
| passengerName | string | Y |  | 乘客姓名 |
| ticketType | int | Y |  | 票类型 0成人票 1儿童票 |

**3.14.5接口入参示例**

抢票接口需要进行加密，请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数样例如下：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“orderId”: “testId”,

“cheCi”: “K1237”,

“fromStationCode”: “BZH”,

“fromStationName”: “亳州”,

“toStationCode”: “RZH”,

“toStationName”: “温州”,

“trainDate”: “2015-08-29”,

“trainStartTime”: “16:30”,

“callBackUrl”: “{途牛URL}”,

“seat”: {

“seatType”: “M”,

“seatName”: “一等座”,

“seatPrice”: “200.00”

},

“extSeat”: [

{

“seatType”: “M”,

“seatName”: “一等座”,

“seatPrice”: “200.00”

}

],

“passengers”: [

{

“passportNo”: “222222”,

“passportTypeId”: “1”,

“passengerName”: “16:30”,

“ticketType”: 0

}

],

“ticketType”: 0,

“tikcetBakcUp”: “123”,

“minSeatCount”: 2,

“transportName”: “张三”,

“transportPhone”: “18000”,

“transportAddress”: “南京市XX区”

}

}

**3.14.6接口出参示例**

**3.14.6.1同步反馈**

{

“success”: true,

“returnCode”: 301,

“errorMsg”: “”,

“data”: { }

}

**3.14.6.2异步反馈**

此返回需要加密，反馈请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数示例如下：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“passportNo”: “testId”,

“orderId”: “K1237”,

“failMsg”: [

{

“passportNo”: “1111”,

“passportTypeId”: “1”,

“passengerName”: “李四”,

“reason”: 1,

“sernam”: “2015-09-18”,

“preTrainNo”: “K188”

}

],

“tickets”: [

{

“orderNumber”: “E11582”,

“seatType”: “M”,

“seatNo”: “13车厢0A座”,

“price”: “200.00”,

“passportNo”: “222”,

“passportTypeId”: “1”,

“passengerName”: “张三”,

“ticketType”: 0

}

]

}

}

**3.14.7通用输出参数增加可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 1400 | 其他 |
| 1401 | 所购买的车次坐席已无票 |
| 1402 | 身份证件已经实名制购票，不能再次购买同日期同车次的车票 |
| 1403 | 商户传过来的票价和12306不符 |
| 1404 | 车次数据与12306不一致 |
| 1405 | 乘客信息错误 |
| 1406 | 12306乘客身份信息核验失败 |

**3.15 快递信息回调**

**3.15.1接口说明**

线下出票之后，供应商回调途牛的快递信息接口

**3.15.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/train/delivery/sername

DataType：json

**3.15.3接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderId | string | Y |  | 途牛订单号 |
| transportNo | string | Y |  | 快递单号 |
| transportDate | string | Y | 20150817 | 快递日期 |
| transportCompany | string | Y | 顺丰速递 | 快递公司，默认为顺丰速运 |

**3.15.4接口出参**

**3.15.4.1同步接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| success | string | Y |  | 是否成功 |

**3.15.4.2异步接口出参**

无异步。

**3.15.5接口入参示例**

此接口需要进行加密，请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数示例如下：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“orderId”: “testId”,

“transportNo”: “456213”,

“transportDate”: “20150817”,

“transportCompany”: “顺丰速递”

}

}

**3.15.6接口出参示例**

**3.15.6.1同步反馈**

{

“success”: true

}

**3.15.7通用输出参数增加可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 1500 | 其他 |

**3.16 取消抢票**

**3.16.1接口说明**

取消抢票接口，该接口同步反馈取消结果

**3.16.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/trainAccount/cancelGrab

DataType：json

**3.16.3接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderId | string | Y |  | 途牛订单号 |
| callBackUrl | string | Y |  | 回调地址 |
| userName | string | N |  | 12306用户名 |
| userPassword | string | N |  | 12306账号 |

**3.16.4接口出参**

**3.16.4.1同步接口出参**

见出参示例

**3.16.5接口入参示例**

接口需要进行加密，请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数样例如下：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“orderId”: “111”,

“sername”: null

“userPassword”:null,

“callBackUrl”: “callbackUrl”

}

}

**3.16.6接口出参示例**

**3.16.6.1同步反馈**

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “取消成功”,

“data”: { }

}

**3.16.7通用输出参数增加可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 取消成功 |
| 1600 | 已经抢票成功，不能取消 |
| 1700 | 取消失败(支付中) |

### 3.17 改签

#### 3.17.1接口说明

此接口用于申请改签。

此接口同步返回请直接返回成功，调用结果以异步反馈的结果为准

此接口的data部分需要加密，需要签名

出发站到达站不可变更

#### 3.17.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/change/apply

DataType：json

#### 3.17.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| orderNumber | string | Y | E010400222 | 12306取票号 |
| changeId | string | Y |  | 我方改签请求流水号（后续改签的反馈取消等依据此流水号）一个订单多张票，每张票支持独立的改签请求 |
| changeCheCi | string | Y |  | 改签新车票的车次 |
| changeDateTime | string | Y | 2016-02-23 17:32:00 | 改签新车票的发车时间 |
| changeZwCode | string | Y |  | 改签新车票座位编码（参见4.3.2座位编码） |
| hasSeat | boolean | Y |  | 改签是否接受无座票  true:不改无座票  false:允许改到无座票 |
| oldZwCode | string | Y |  | 原车票的座位编码 |
| userName | string | N |  | 原下单的12306用户名 |
| userPassword | string | N |  | 原下单的12306密码 |
| oldTicketInfos | Json数组 | Y |  | 原车票的信息，格式参见4.3.1乘客信息说明 |
| callBackUrl | string | Y |  | 锁票异步回调地址[选填],此属性有值则需要回调此地址 |

#### 3.17.4接口出参

##### 3.17.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | 是 | 15538430 | 合作伙伴方订单号 |

##### 3.17.4.2异步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| orderNumber | string | Y | E010400222 | 12306取票号 |
| changeId | string | Y |  | 我方改签请求流水号（后续改签的反馈取消等依据此流水号）一个订单多张票，每张票支持独立的改签请求 |
| changeSuccess | boolean | Y |  | 改签是否成功 |
| cheCi | string | N |  | 改签后的车次 |
| fromStationCode | string | N |  | 改签后的出发站简码 |
| fromStationName | string | N |  | 改签后的出发站名称 |
| toStationCode | string | N |  | 改签后的到达站简码 |
| toStationName | string | N |  | 改签后的到达站名称 |
| trainDate | string | N |  | 改签后的乘车日期(格式同占位反馈) |
| startTime | string | N |  | 改签后的从出发站开车时间(格式同占位反馈) |
| arriveTime | string | N |  | 改签后的抵达目的站的时间(格式同占位反馈) |
| priceChangeType | int | Y |  | 改签票价变动类型，改签失败时为0  1 表示新票款高于原票款  2 表示新票款与原票款相等  3 表示新票款低于原票款（产生差额退款） |
| priceChangeTypeInfo | string | N |  | 改签票价变动信息描述，与priceChangeType三个值对应  示例如下  收取新票款:340.0 元,退还原票票款:230.0 元  改签票款差价:0.0 元  退还票款差额:52.0 |
| priceDifference | decimal | N |  | 差价，差价=新票总价-原票总价（成功才有值） |
| diffRate | decimal | N |  | 差额退款费率，默认0，如0,0.05 |
| totalPriceDiff | decimal | N |  | 实际退还差额合计，如7.0 |
| fee | decimal | N |  | 改签手续费(差额退款时有大于0的值) |
| newTicketInfos | Json数组 | Y |  | 改签后新车票的信息，格式参见4.3.1乘客信息说明 |
| paymentUrls | PaymentUrl[] | N |  | 支付链接 |
| clearTime | String | N | 2017-10-26 12:00:00 | 清位时间 |

**PaymentUrl**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| type | Integer | N |  | 1、支付宝 2、银联 |
| url | String | N |  | 支付链接 |
| validStartTime | String | N |  | 有效的起始时间 |

#### 3.17.5接口入参示例

改签申请接口需要进行加密，请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数样例如下：

{

"account": "testAccount",

"sign": "testsign",

"timestamp": "2015-08-0300:00:00",

"data": {

"vendorOrderId": "testVendorOrderId",

"orderId": "AC9WSY4H1H00",

"orderNumber": "E010400222",

"changeId": "10020160203002",

"changeCheCi": "K1239",

"changeDateTime": "2015-08-2916: 00: 00",

"changeZwCode": "1",

"hasSeat": true,

"oldZwCode": "1",

"callBackUrl": "{途牛URL}/train/changeOrderFeedback",

"oldTicketInfos": [

{

"reason": 0,

"price": "152.5",

"passengerId": 33292562,

"oldTicketNo": "E0104002221053148",

"zwCode": "1",

"cxin": "05车厢,10F",

"passportTypeName": "二代身份证",

"passportNo": "341222197608254405",

"zwName": "硬座",

"piaoType": "1",

"passengerName": "张三",

"passportTypeId": "1",

"piaoTypeName": "成人票"

}

]

}

}

#### 3.17.6接口出参示例

##### 3.17.6.1同步反馈

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”

}

}

##### 3.17.6.2异步反馈

此返回需要加密，反馈请求为：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: “Base64ResultTest”

}

为方便理解，此处给出解密后的参数示例如下：

{

"account": "testAccount",

"sign": "testsign",

"timestamp": "2015-08-03 00:00:00",

"returnCode": 231000,

"errorMsg": "",

"data": {

"vendorOrderId": "testVendorOrderId",

"orderId": "AC9WSY4H1H00",

"orderNumber": "E010400222",

"changeId": "10020160203002",

"changeSuccess": true,

"cheCi": "K1239",

"fromStationCode": "BZH",

"fromStationName": "亳州",

"foStationCode": "RZH",

"toStationName": "温州",

"trainDate": "2015-08-29",

"startTime": "16:00:00",

"arriveTime": "11:45:00",

"priceChangeType": 3,

"priceChangeTypeInfo": "改签票款差价:2.5 元",

"priceDifference": -2.5,

"diffRate": 0,

"totalPriceDiff": 2.5,

"fee": 0,

“clearTime”:”2017-10-26 13:00:00”

"newTicketInfos": [

{

"reason": 0,

"price": "150.0",

"passengerId": 33292562,

"oldTicketNo": "E0104002221053148",

"newTicketNo": " E0104002221053159",

"zwCode": "1",

"cxin": "06车厢,13F",

"passportTypeName": "二代身份证",

"passportNo": "341222197608254405",

"zwName": "硬座",

"piaoType": "1",

"passengerName": "张三",

"passportTypeId": "1",

"piaoTypeName": "成人票"

}

],

"paymentUrls": [

{

"type": 1,

"url": "http://10.10.32.160/fb2/t1/group1/M00/05/33/Cgogo1mNJJWIeaMJAAAKe-fantgAAABSQODVM4AAAqT61.html",

"validStartTime": "2017-09-19 16:15:23"

},

{

"type": 2,

"url": "http://10.10.32.160/fb2/t1/group1/M00/05/33/Cgogo1mNJJWIeaMJAAAKe-fantgAAABSQODVM4AAAqT61.html",

"validStartTime": "2017-09-19 16:15:23"

}

]

}

}

#### 3.17.7通用输出参数增加可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 改签占位成功 |
| 1600101 | 请求改签失败 |
| 1600102 | 取消改签次数超过上限，无法继续操作 |
| 1600103 | 距离开车时间太近无法改签 |
| 1600104 | 未找到要改签的车次 |
| 1600105 | 未找到要改签的车票 |
| 1600106 | 账号登陆失败 |
| 1600107 | 已出票，请到车站办理 |
| 1600108 | 已改签，不能再次改签 |
| 1600109 | 已退票，无法改签 |
| 1600110 | 旅游票，请到站办理 |
| 1600111 | 订单状态不正确 |
| 1600112 | 重复提交的改签信息 |
| 1600113 | 批量改签新车票和原车票座位都不能为卧铺 |
| 1600114 | 没有余票 |
| 1600115 | 当前的排队购票人数过多，请稍后重试 |
| 1600116 | 限制高消费 |
| 1600117 | 存在未完成订单 |
| 1600118 | 用户已在其他地点登录 |
| 1600119 | 您好，开车前48小时（不含）以上，可改签预售期内的其它列车；开车前48小时以内且非开车当日，可改签票面日期当日及之前的其它列车；开车当日，可改签票面日期当日的其它列车 |
| 1600120 | 行程冲突 |
| 1600121 | 列车停运 |

### 3.18 取消改签

#### 3.18.1接口说明

此接口是指申请改签异步反馈成功之后，确认改签之前放弃某次改签的接口。

此接口同步返回请直接返回成功，调用结果以异步反馈的结果为准

#### 3.18.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/change/cancel

DataType：json

#### 3.18.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| changeId | string |  |  | 我方改签请求流水号（后续改签的反馈取消等依据此流水号）一个订单多张票，每张票支持独立的改签请求 |
| callBackUrl | string | Y |  | 回调URL地址, 此属性有值则需要回调此地址 |

#### 3.18.4接口出参

##### 3.18.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |

##### 3.18.4.2异步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string |  |  | 订单号 |
| changeId | String |  |  |  |

#### 3.18.5接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”,

“changeId”: “10020160203002”,

“callBackUrl”: “{途牛URL}/train /cancelOrderFeedback”

}

}

#### 3.18.6接口出参示例

##### 3.18.6.1 同步接口出参示例

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “15274218T7650055”,

“changeId”: “10020160203002”,

“orderId”: “AC9WSY4H1H00”

}

}

##### 3.18.6.2异步接口出参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “15274218T7650055”,

}

}

#### 3.18.7通用输出参数增加可能增加的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 取消改签成功 |
| 1600301 | 取消改签失败 |
| 1600302 | 订单已确认改签，不能取消改签 |
| 1600303 | 未找到要取消改签的订单 |
| 1600304 | 取消改签超时 |

### 3.19 确认改签

#### 3.19.1接口说明

此接口是指途牛申请改签成功（成功分配到了新的座位席别信息）之后，在规定的时间（比如30分钟）内发出确认改签请求。

此接口同步返回请直接返回成功，调用结果以异步反馈的结果为准

#### 3.19.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/change/confirm

DataType：json

#### 3.19.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| changeId | string |  |  | 我方改签请求流水号 |
| callBackUrl | string | Y |  | 回调URL地址, 此属性有值则需要回调此地址 |

#### 3.19.4接口出参

##### 3.19.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |

##### 3.19.4.2异步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | 是 |  | 合作伙伴方订单号 |
| orderId | string | 是 |  | 途牛订单号 |
| changeId | string |  |  | 我方改签请求流水号 |

#### 3.19.5接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”,

“changeId”: “10020160203002”,

“callBackUrl”: “{途牛URL}/train /confirmOrderFeedback”

}

}

#### 3.19.6接口出参示例

##### 3.19.6.1同步接口出参

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AEME3TJS1H00”,

}

}

##### 3.19.6.2异步接口出参

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AEME3TJS1H00”,

“changeId”: “10020160203002”,

}

}

#### 3.19.7通用输出参数增加可能增加的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 确认改签成功 |
| 1600201 | 确认改签失败 |
| 1600202 | 确认改签时间已超时，确认改签失败 |
| 1600203 | 未找到订单 |

### 3.20 催退款通知

#### 3.20.1接口说明

此接口是我方用户已线下退票，但是供应商尚未处理或通知的情况下，我方用户主动发起催促供应商处理。供应商在接收到我方催退款通知之后，人工去处理，并变更该订单状态。

此接口同步反馈当前该订单的状态，不需要异步反馈。我方会不定时通过该接口来获取退款状态。

#### 3.20.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/refundNotice

DataType：json

#### 3.20.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| tickets | String[] | Y |  | 需要退款的车票号，下单或改签的车票号 |
| orderNumber | String | Y |  | 12306取票号 |

#### 3.20.4接口出参

##### 3.20.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| orderNumber | String | Y |  | 12306取票号 |
| tickets | string |  |  | 车票退票状态信息(json字符串数组形式，每张车票包含乘车人信息和退票相关信息，如：  [“ticketNo”:” E2610890401070051”,”passengerName”:”王二”,”passportTypeId”:1,”passportseNo”:”421116198907143795”,”returnSuccess”:true,”returnMoney”:”20.05”,”returnTime”:”2014-02-13 15:00:05”,”returnFailId”:””,”returnFailMsg”:””,”returnStatus”:1, “returnMsg”:”退款处理中”]]  注： returnStatus为乘客的退款状态，1退款处理中，2已退款，3退款失败，4其他；returnMsg填写提示文本信息，在退款失败时返回详细的失败信息。returnStatus为4时，returnMsg需要详细的信息。  returnStatus为1、2时returnSuccess为true。  returnStatus为3、4时returnSuccess为false。 |

#### 3.20.5接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00: 00: 00”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”,

“orderNumber”: “E12306”,

“tickets”: [

“E1230600”,

“E12306002”

]

}

}

#### 3.20.6接口出参示例

##### 3.20.6.1同步接口出参

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AEME3TJS1H00”,

“orderNumber”: “E12306”,

“tickets”:[

{ “ticketNo”:”E2610890401070051”,

“passengerName”:”王二”,

“passportTypeId”:1,

“passportseNo”:”421116198907143795”,

“returnSuccess”:true,

“returnMoney”:”20.05”,

“returnTime”:”2014-02-13 15:00:05”,

“returnFailId”:””,

“returnFailMsg”:””,

“returnStatus”:1,

“returnMsg”:”退款处理中”

}

]

}

}

#### 3.20.7通用输出参数增加可能增加的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 210001 | 催退款失败 |
| 231000 | 操作成功 |

#### 3.20.8备注

1、若我方请求的订单中存在多张车票，目前供应商从12306拿到的流水信息不能保证准确

区分每张票的退款状态以及退款金额。此时出参中的车票明细金额可以为空。returnMsg须注 该订单共计退款xxx元

2、returnTime 为供应商接收到12306退款流水时的时间

3、部分供应商取不到 returnTime 此时returnTime为空

### 3.21 报警通知

#### 3.21.1接口说明

此接口是我方在下单、确认出票、取消下单、线上退票等操作超过一定时间范围时，我方主动调用，向供应商发出一个报警，提醒供应商抓紧时间处理。

#### 3.21.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/alarm

DataType：json

#### 3.21.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| function | string | Y |  | 功能类型：book下单，confirm确认出票，return线上退票，cancel取消订单 |
| refundId | String | N |  | 退票流水号。当function为return时，必须传递对应退票请求的流水号，其他情况不传。 |
| Duration | int | Y |  | 从发起请求到目前为止，已停留了多长时间未反馈。单位为秒。 |

#### 3.21.4接口出参

##### 3.21.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| msg | string | Y |  | 报警提醒接收状况描述，纯文本。比如某订单的占位提醒已接收…… |

#### 3.21.5接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”,

“function”: “book”,

“refundId”: “1000001”,

“duration”: 200

}

}

#### 3.21.6接口出参示例

##### 3.21.6.1同步接口出参

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AEME3TJS1H00”,

“msg”: “ AEME3TJS1H00订单的下单报警提醒已接收，我方会尽快处理，请等待。”

}

}

#### 3.21.7通用输出参数增加可能增加的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 2101 | 订单已处理完毕，无须提醒 |
| 2102 | 报警提醒失败 |

### 3.22 线下退款进度查询

#### 3.22.1接口说明

我方客户已线下退票（或线下改签），为了解当前退款进度，主动发起查询，供应商侧应提供当前订单所涉及的线下退款进度。

此接口同步反馈当前该订单的退款状态，不需要异步反馈。我方会不定时通过该接口来获取退款进度。

#### 3.22.2接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/refund/query

DataType：json

#### 3.22.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |

#### 3.22.4接口出参

##### 3.22.4.1同步接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| refundList | Array<RefundInfo > | Y |  | 退款信息 |
|  |  |  |  |  |

RefundInfo：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| ticketNo | string | Y |  | 车票号 |
| returnMoney | string | Y |  | 退款金额 |
| returnTime | string | Y |  | 退款时间（yyyy-MM-dd HH:mm:ss） |
| returnStatus | int | Y |  | 退款状态：  1退款处理中  2已退款  3 退款失败  4其他 |
| returnMsg | string | Y |  | 退款状态描述，对应returnStatus的中文描述，returnStatus为4时需要详细的信息。 |

#### 3.22.5接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“vendorOrderId”: “testVendorOrderId”,

“orderId”: “AC9WSY4H1H00”

}

}

#### 3.22.6接口出参示例

##### 3.22.6.1同步接口出参

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“returnList”: [

{

“ticketNo”: “E2610890401070051”,

“returnMoney”: “20.05”,

“returnTime”: “2014-02-13 15:00:05”,

“returnStatus”: 1,

“returnMsg”: “退款处理中”

}

]

}

}

#### 3.22.7通用输出参数增加可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 2201 | 暂未收到退款，12306按规定会在15个日内退还到账，请耐心等待。 |
|  |  |

### 3.23 订单跟踪

[url:{api}/trainTicket/searchStatusFrom12306](url:%7bapi%7d/trainTicket/searchStatusFrom12306)

Method:post

dataTyep:json

#### 3.23.1入参：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **示例值** | **描述** |
| orderId | String | 是 | 123245698 | 途牛订单id |
| orderNumber | String | 是 | E13156 | 火车票取票号 |
| vendorOrderId | String | 是 | XXXX | 供应商订单号 |
| msg | String | 否 |  | 预留字段 |
| userName | String | 否 | 1111 | 12306账号 |
| password | String | 否 | \*\*\* | 密码 |
| ticket | object | 否 | 0 | 如果没有值则返回orderNumber中所有的票的状态，否则返回指定取票号票的状态 |

Ticket

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **示例值** | **描述** |
| ticketNo | String | 否 | 123245698 | 火车票订单号 |

#### 3.23.2出参：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **示例值** | **描述** |
| orderId | String | 是 | 123245698 | 途牛订单id |
| orderNumber | String | 是 | E13156 | 火车票订单号 |
| vendorOrderId | String | 否 | XXXX | 供应商订单号 |
| ticket | object | 是 |  |  |
| msg | String | 否 |  | 预留字段 |

Ticket

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **示例值** | **描述** |
| ticketNo | String | 是 | 123245698 | 火车票订单号 |
| statusInfo | Object | 是 | E13156 | 火车票对应票的相关状态信息 |

Status

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **示例值** | **描述** |
| operationTime | data | 否 | 2016-05-16 00:00:00 | 状态变更时间 |
| status | String | 是 | 支付成功 | 车票状态：占位成功/支付成功/办理改签/办理退票/制票成功/检票进站/改签中/变更到站中/检票出站/已进站/已变更到站/  其他… |
| channel | String | 否 | 互联网 | 改签、退票、购票、制票渠道：互联网/app/xx售票大厅（与status对应） |
| msg | String | 否 |  | 预留字段（status选择其他时，该字段须说明原因 |

#### 3.23.3入参示例：

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-0300: 00: 00”,

“data”:

{

“username”:”2323”,

“password”:”23NSAN234/3/==”,

“orderId”: “tn16051625698741”,

“orderNumber”: “E123456987”,

“vendorOrderId”: “456321789”,

“ticket”: [

{

“ticketNo”: “E1234569871213654”

}

],

“msg”: “”

}

}

#### 3.23.4出参示例：

{

“success”: true,

“returnCode”: 200002

“errorMsg”: “”,

“data”: null

“msg”:””

}

}

#### 3.23.5可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 操作成功 |
| 200001 | 查询信息失败 |
| 200002 | 订单不存在 |
| 231008 | 参数异常 |

### 3.24抢票次数查询

[url:{api}/train/grabTimes](url:%7bapi%7d/train/grabTimes)

Method:post

dataTyep:json

#### 3.24.1 入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **示例值** | **描述** |
| **orderId** | **String** | **Y** | **test12345679** | **途牛订单号** |
| **vendorOrderId** | **String** | **N** | **CDFFD45662** | **供应商对应途牛的订单号** |
| **reqtime** | **String** | **Y** | yyyyMMddHHmmss | **请求时间** |
| **sign** | **String** | **Y** | ADSASDASDASDDASD | **请求签名(MD5(orderId+ reqtime + vendorOrderId+MD5(key)))key为约定的唯一值** |

#### 3.24.2 出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **示例值** | **描述** |
| **orderId** | **String** | **Y** | **test12345679** | **途牛订单号** |
| number | **Integer** | **N** | **13** | **供应商对应途牛的订单号** |
| success | **boolean** | **Y** | true | **请求是否成功** |
| **code** | **String** | **Y** |  | **请求返回编码** |
| **msg** |  |  |  | **请求返回消息** |

#### 3.24.3 入参示例

{

“**orderId**”:”**test12345679**”,

“**vendorOrderId**”:”**CDFFD45662**”,

“**reqtime**”:”20160525160330”,

“**sign**”: “ADSASDASDASDDASD”

}

#### 3.24.4 出参示例

{

“**orderId**”:”**test12345679**”,

“number”:13,

“success”:true,

“**code**”:”231000”,

“**msg**”:””

}

#### 3.24.5 输出参数可能值

|  |  |  |
| --- | --- | --- |
| success值 | code值 | msg值 |
| true | 231000 | 操作成功 |
| false | 231315 | 操作失败 |
| false | 231999 | 接口异常 |

### 3.25 12306验证码获取

#### 3.25.1接口说明

途牛通过此接口获取12306的验证码,用于后续登录,占位,改签等动作需要的验证码

本接口同步反馈验证码获取结果

#### 3.25.2 接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/captcha/getCaptcha

DataType：json

#### 3.25.3接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| businessType | String | Y | 2 | 业务类型  1：登录  2：占位  3：改签 |
| isRefreshcaptcha | boolean | Y | true | 是否刷新验证码 |
| userName | String | N | oumin | 12306用户名 |
| userPassword | String | N | a3123333 | 12306密码 |
| requestId | String | Y | 7965C4ADF98240C6A9DF525722FD7FC5 | 请求ID（32位）  后续供应商以此标识关联订单信息 |
| busData | object | N |  | 获取验证码附带业务参数(占位、改签) |

busData定义如下：

**登录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| ~~orderId~~ | ~~String~~ | ~~Y~~ |  | ~~途牛订单号~~ |
| cheCi | String | Y |  | 车次 |
| fromStationCode | String | Y |  | 出发站简码 |
| fromStationName | String | Y |  | 出发站名称 |
| toStationCode | String | Y |  | 到达站简码 |
| toStationName | String | Y |  | 到达站名称 |
| trainDate | String | Y |  | 乘车日期 |
| ~~callBackUrl~~ | ~~String~~ | ~~Y~~ |  | ~~锁票异步回调地址[选填],此属性有值则需要回调此地址~~ |
| hasSeat | Boolean | Y |  | 是否出无座票  true:不出无座票  false:允许出无座票 |
| passengers | String | Y |  | 乘客信息的json字符串。可以是多个乘客信息，最多5个，如：[{乘客1信息},{乘客2信息},…]，也可以只有一个，[{乘客1信息}]。乘客参数见附注1。重要提示：不能只购买儿童票，如果购买儿童票，必须使用随行成人的成人票证件信息（包括姓名、证件号码）。 |
| Contact | String | N |  | 联系人姓名 |
| phone | String | N |  | 联系人手机 |
| userName | String | N |  | 12306用户名 |
| userPassword | String | N |  | 12306密码 |
| ~~insureCode~~ | ~~String~~ | ~~N~~ |  | ~~保险产品编号（有值，则表示此单购买保险）~~ |

**改签**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| vendorOrderId | string | Y |  | 合作伙伴方订单号 |
| orderId | string | Y |  | 途牛订单号 |
| orderNumber | string | Y | E010400222 | 12306取票号 |
| ~~changeId~~ | ~~string~~ | ~~Y~~ |  | ~~我方改签请求流水号（后续改签的反馈取消等依据此流水号）一个订单多张票，每张票支持独立的改签请求~~ |
| changeCheCi | string | Y |  | 改签新车票的车次 |
| changeDateTime | string | Y | 2016-02-23 17:32:00 | 改签新车票的发车时间 |
| changeZwCode | string | Y |  | 改签新车票座位编码（参见4.3.2座位编码） |
| hasSeat | boolean | Y |  | 改签是否接受无座票  true:不改无座票  false:允许改到无座票 |
| oldZwCode | string | Y |  | 原车票的座位编码 |
| userName | string | N |  | 原下单的12306用户名 |
| userPassword | string | N |  | 原下单的12306密码 |
| oldTicketInfos | Json数组 | Y |  | 原车票的信息，与改签占位接口格式一致 |
| ~~callBackUrl~~ | ~~string~~ | ~~Y~~ |  | ~~锁票异步回调地址[选填],此属性有值则需要回调此地址~~ |

#### 3.25.4 接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| captchaType | String | Y | occupy | 验证码类型  login 登录,  occupy 占位,  change改签占位 |
| requestId | String | Y |  | 原请求ID |
| file | String | Y |  | 验证码文件（Base64编码） |

#### 3.25.5 接口入参示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-0300: 00: 00”,

“data”: {

“businessType”: “occupy”,

“isRefreshcaptcha”: false,

“sername”: “oumingf”,

“userPassword”: “a32132332”,

“requestId”: “7965C4ADF98240C6A9DF525722FD7FC5”,

“busData”: {

“username”: “aaaa”,

“cheCi”: “Z290”,

“fromStationCode”: “SOH”,

“fromStationName”: “绍兴”,

“toStationCode”: “GIW”,

“toStationName”: “贵阳”,

“trainDate”: “2016-12-01”,

“hasSeat”: true,

“passengers”: [

{

“passengerId”: 1721068,

“ticketNo”: null,

“passengerName”: “李文”,

“passportNo”: “360302186701151512”,

“passportTypeId”: “1”,

“passportTypeName”: “二代身份证”,

“piaoType”: “1”,

“piaoTypeName”: “成人票”,

“zwCode”: “1”,

“zwName”: “硬座”,

“cxin”: null,

“price”: “213.00”,

“reason”: 0,

“provinceCode”: null,

“schoolCode”: null,

“schoolName”: null,

“studentNo”: null,

“schoolSystem”: null,

“enterYear”: null,

“preferenceFromStationName”: null,

“preferenceFromStationCode”: null,

“preferenceToStationName”: null,

“preferenceToStationCode”: null

}

],

“contact”: “李文”,

“phone”: “13802084121”,

“sername”: null,

“userPassword”: null

}

}

}

**3.25.6 接口出参示例**

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“captchaType”: “login”,

“requestId”: “378D18F63C8B46A8A93BE33F231A1081”,

“file”: “”

}

}

**3.25.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 请求成功 |
| 25001 | 12036账号不存在 |
| …… |  |

### 3.26 12306验证码提交

**3.26.1接口说明**

途牛册使用本接口提交12306验证获取验证码、12306用户登录核验这两个接口获取到的验证码坐标，以及本次业务打码是否完成，若验证码是通过12306验证码获取接口得到，供应商依据其接口中的bussinessType字段判断是否完成，若未完成，供应商应反馈下轮打码的图片信息，途牛侧完成新一轮打码后会使用该接口再次提交打码坐标，直至打码完成。若验证码是通过12306用户登录核验获取，表明本次打码业务为登录业务。

本接口同步反馈核验结果

**3.26.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/train/captcha/submitCaptcha

DataType：json

**3.26.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| requestId | String | Y | 378D18F63C8B46A8A93BE33F231A1081 | 请求ID |
| isRefreshcaptcha | boolean | false |  | 是否刷新验证码 |
| randCode | String | N | 112,75,188,67 | 验证码坐标 |

**3.26.4 接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| requestId |  |  | 378D18F63C8B46A8A93BE33F231A1081 | 请求ID |
| isCorrect | Boolean | Y | true | 验证码是否正确 |
| isFinish | Boolean | Y | false | 打码流程是否完成 |
| captchaType | String | N | occupy | 下次打码类型  login 登录,  occupy 占位,  change改签占位 |
| flie | String | N |  | 下一次打码文件（Base64编码） |

**3.26.5 接口入参示例**

data字段数据为加密信息，此处给出明文示例

{  
    “account”:”testAccount”,  
    “sign”:”testsign”,  
    “timestamp”:”2015-08-03 00: 00: 00”,  
    “data”:{  
        “requestId”:”378D18F63C8B46A8A93BE33F231A1081”,  
        “isRefreshcaptcha”:false,  
        “randCode”:”112,75,188,67”  
    }  
}

**3.26.6 接口出参示例**

{  
    “success”:true,  
    “returnCode”:231000,  
    “errorMsg”: “”,  
    “data”:{  
        “requestId”:”378D18F63C8B46A8A93BE33F231A1081”,  
        “isCorrect”:true,  
        “isFinish”:false,  
        “captchaType”:”occupy”,  
        “file”:””  
    }  
}

**3.26.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 23001 | 操作成功 |
| 26001 | 12036账号不存在 |
| 26001 | 12306账号密码错误 |
| 26003 | 用户被锁定，20分钟后重试 |
| 26004 | XXXX乘客身份信息冒用 |
| 26005 | XXXX乘客身份信息未通过核验 |
| 26006 | 该12306账号所添加的常旅数量已达到上限 |
| 26007 | 您的身份信息正在进行网上核验，您也可持居民身份证原件到车站售票窗口即时办理核验 |
| …… |  |

### 3.27 12306用户登录核验

**3.27.1接口说明**

途牛侧通过该接口请求供应商判断指定账号是否已登录，若判断结果为未登录，供应商反馈登录验证码信息

本接口同步反馈结果

**3.27.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/train/captcha/checkLogin

DataType：json

**3.27.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| requestId | String | Y | 324C049AF502405EA370AFD852C923CA | 请求ID |
| userName | String | N | oumin | 12306账号 |
| userPassword | String | N | a31231243 | 12306密码 |

3.27.4 接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| isLogin | boolean | Y | true | 是否已登录 |
| file | String | N |  | 验证码文件(Base64编码) |

**3.27.5 接口入参示例**

data字段数据为加密信息，此处给出明文示例

{  
    “account”:”testAccount”,  
    “sign”:”testsign”,  
    “timestamp”:”2015-08-0300: 00: 00”,  
    “data”:{  
        “requestId”:”324C049AF502405EA370AFD852C923CA”,  
        “sername”:”oumin”,  
        “userPassword”:”a31231243”  
    }  
}

**3.27.6 接口出参示例**

{  
    “account”:”testAccount”,  
    “sign”:”testsign”,  
    “timestamp”:”2015-08-0300: 00: 00”,  
    “data”:{  
        “lsLogin”:false,  
        “file”:””  
    }  
}

**3.27.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 请求成功 |
| 27001 | 12036账号不存在 |
| …… |  |

### 3.28 身份核验信息上传

**3.31.1接口说明**

供应商调用本接口，上传本地库中核验通过的乘客身份信息

乘客身份核验信息变更后，可在点调用该接口进行更新

初期同步数据时，建议不要使用高并发

目前每次调用接口最多传20条数据，超过20条的请求接受，不处理

**3.31.2 接口调用方法**

Method：HTTP-POST

Url：{Tuniu\_Url}/aln/common/identityValidationUpload

DataType：json

**3.31.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| number | Integer | Y | 100 | 一次请求，包含的核验身份信息数量 |
| vendorOrderId | String | Y | 852346461 | 供应商流水号 |
| idCardList | ArrayList | Y |  | 身份 |

idCardList

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| idCard | String | Y | 101541990124514 | 证件号 |
| userName | String | Y | 小武 | 证件持有人姓名 |
| type | String | Y | 1 | 证件类型（**参考4.3.3**） |
| idCardName | String | Y | 身份证 | 证件名称 |
| status | Integer | Y | 1 | 12306核验状态  1：核验通过  3：核验不通过  5:已经上传的白名单删除标识 |
| msg | String | N | 更新 | 备注信息 |

**3.31.4 接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| success | String | Y | success | 是否成功 |

**3.31.4接口入参示例**

{

“account”: “testAccount”,

“sign”: “testsign”,

“returnCode”: 231000,

“timestamp”: “2016-06-22 00: 00: 00”,

“data”: {

“number”:100,

“vendorOrderId”:”xxx-ssss”,

“idCardList”: [

{

“idCard”: “1015411990124514565”,

“sername”: “小武”,

“type”: “1”,

“idCardName”: “身份证”,

“status”: 1,

“msg”: null

}

]

}

}

**3.31.5 接口出参示例**

success

### 3.29 12306注册—账号校验

**3.29.1接口说明**

途牛侧通过该接口请求供应商判断指定账号是否已在12306网站注册，供应商反馈该账号是否可用。本接口同步反馈结果。

**3.29.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/train/register/checkUserName

DataType：json

**3.29.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| userName | String | Y | oumin | 12306账号 |

**3.29.4 接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| isPass | int | Y | 0 | 是否可用  0:可用  1:不可用 |
| msg | String | Y | 用户名可用 | 描述是否可用的信息 |

**3.29.5 接口入参示例**

data字段数据为加密信息，此处给出明文示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“sername”: “a31231243”

}

}

**3.29.6 接口出参示例**

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“isPass”: 0,

“msg”: “用户名可用”

}

}

**3.29.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 请求成功 |
|  |  |

### 3.30 12306注册—信息核验

**3.30.1接口说明**

途牛侧通过该接口请求供应商核验12306注册信息，12306核验通过将发送验证码至指定手机号。本接口同步反馈核验结果。

**3.30.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/train/register/getRandCode

DataType：json

**3.30.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| passengerType | String | Y |  | 乘客类型  成人：1  儿童：2  学生：3  残疾军人、伤残人民警察：4 |
| idTypeCode | String | Y |  | 证件类型  身份证：1  港澳通行证：C  台湾通行证：G  护照：B |
| idNo | String | Y |  | 证件号 |
| confirmPassWord | String | Y |  | 确认密码（与password一致） |
| userName | String | Y |  | 12306账号名 |
| name | String | Y |  | 姓名 |
| mobileNo | String | Y |  | 手机号 |

**3.30.4 接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| isPass | int | Y | 0 | 表明注册信息通过  0:通过  1:不通过 |
| msg | String | Y |  | 原样返回12306提示 |

**3.30.5 接口入参示例**

data字段数据为加密信息，此处给出明文示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“passengerType”: “1”,

“sername”: “yanxinyi111”,

“name”: “海洋”,

“idTypeCode”: “1”,

“idNo”: “610324198902221812”,

“confirmPassWord”: “1231243”,

“mobileNo”: “15921223066”

}

}

**3.30.6 接口出参示例**

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“isPass”: 0,

“msg”: “核验通过”

}

}

**3.30.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 请求成功 |
|  |  |

### 3.31 12306注册—提交

**3.31.1接口说明**

途牛侧通过该接口请求供应商提交12306注册信息。本接口同步反馈注册结果。

**3.31.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/train/register/register

DataType：json

**3.31.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| passengerType | String | Y |  | 乘客类型  成人：1  儿童：2  学生：3  残疾军人、伤残人民警察：4 |
| idTypeCode | String | Y |  | 证件类型  身份证：1  港澳通行证：C  台湾通行证：G  护照：B |
| idNo | String | Y |  | 证件号 |
| confirmPassWord | String | Y |  | 确认密码（与password一致） |
| userName | String | Y |  | 12306账号名 |
| name | String | Y |  | 姓名 |
| mobileNo | String | Y |  | 手机号 |
| randCode | String | Y |  | 用户使用手机输入的数字码 |

**3.31.4 接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| isPass | int | Y | 0 | 表明注册通过  0:通过  1:不通过 |
| msg | String | Y |  | 原样返回12306注册提示 |

**3.31.5 接口入参示例**

data字段数据为加密信息，此处给出明文示例

{

“account”: “testAccount”,

“sign”: “testsign”,

“timestamp”: “2015-08-03 00:00:00”,

“data”: {

“passengerType”: “1”,

“sername”: “yanxinyi111”,

“name”: “海洋”,

“idTypeCode”: “1”,

“idNo”: “610324198902221812”,

“confirmPassWord”: “1231243”,

“mobileNo”: “15921223066”,

“randCode”: “123134”

}

}

**3.31.6 接口出参示例**

{

“success”: true,

“returnCode”: 231000,

“errorMsg”: “”,

“data”: {

“isPass”: 0,

“msg”: “注册成功”

}

}

**3.31.7 通用输出参数可能出现的值**

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231000 | 请求成功 |
|  |  |

### 3.32 预定下单12306排队状态推送

**3.32.1接口说明**

供应商调用本接口，按订单维度推送当前12306占位排队状况

供应商每次查询未完成订单，判断是否占位成功时，若发现此订单尚未占座成功

应主动按此接口规范推送当前实时排队情况给我方

**3.32.2 接口调用方法**

Method：HTTP-POST

Url：{Tuniu\_Url}/aln/common/reserveStatusPush

DataType：json

**3.32.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderId | String | Y | tn16121781547273 | 订单号 |
| vendorOrderId | String | Y | xxxxxxxxxx | 供应商订单号 |
| [queue](http://www.baidu.com/link?url=5cmoR3SX7GN5--6r53eUsSeCLI33nkGVT8h-qCFcTOJ705e-KUHYP_efAqFrNLkN-4SiE0rnOvFKvEM-0sUSAR6HgOF-IvUg8kGhnY2vSee)Number | Integer | N | 28 | 当前排队人数 |
| waitTime | String | N | 34分10秒 | 预计排队时间 |
| msg | String | Y | 未出票，订单排队中…… | 12306完整备注信息 |

**3.32.4 接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| success | String | Y | success | 是否成功 |

**3.32.4接口入参示例**

{

“account”: “testAccount”,

“sign”: “testsign”,

“returnCode”: 231000,

“timestamp”: “2016-06-22 00: 00: 00”,

“data”: {

“orderId”: “ tn16121781547273”,

“vendorOrderId”: “xxxxxxxxxx”,

“[queue](http://www.baidu.com/link?url=5cmoR3SX7GN5--6r53eUsSeCLI33nkGVT8h-qCFcTOJ705e-KUHYP_efAqFrNLkN-4SiE0rnOvFKvEM-0sUSAR6HgOF-IvUg8kGhnY2vSee)Number”: “28”,

“waitTime”: “34分10秒”,

“msg”:”未出票，订单排队中……”

}

}

**3.32.5 接口出参示例**

Success

### 3.33 订单状态推送供应商

**3.33.1接口说明**

途牛测将订单当前最新状态推送至供应商，用于供应商进行状态核对

目前仅推送状态异常订单

**3.33.2 接口调用方法**

Method：HTTP-POST

Url：{API\_Url}/train/pushOrderStatus

DataType：json

**3.33.3 接口入参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderId | String | Y | tn16121781547273 | 订单号 |
| vendorOrderId | String | N | xxxxxxxxxx | 供应商订单号 |
| changeId | String | N |  | 改签流水号（改签通知必填） |
| status | Integer | Y | 1 | 1. 预定占位超时告警 2. 出票超时告警 3. 改签占位超时告警 4. 改签确认超时告警 5. 预定占位失败 6. 出票失败 7. 改签占位失败 8. 改签确认失败 |
| msg | String | Y | 订单已超时，当前状态出票失败 | 备注信息 |
| updateTime | String | Y | 2017-07-26 16:59:12 | 状态变更时间 |

**3.33.4 接口出参**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| success | String | Y | success | 是否成功 |

**3.33.4接口入参示例**

{

"account": "testAccount",

"sign": "testsign",

"returnCode": 231000,

"timestamp": "2016-06-22 00: 00: 00",

"data": {

"orderId": " tn16121781547273",

"vendorOrderId": "xxxxxxxxxx",

"changeId": null,

" status ": 2,

" msg ": "订单出票超时，请10分钟内处理完毕 "

}

}

**3.33.5 接口出参示例**

success

### 3.34 查询指定订单的支付链接

#### 3.34.1接口说明

此接口用于套皮支付时请求供应商查询订单的支付链接

#### 3.34.2 接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/queryPaymentUrl

DataType：json

#### 3.34.3 接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderId | String | 是 | tn16121781547273 | 订单号 |

#### 3.34.4 接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| type | Integer | 是 |  | 1、支付宝 2、银联 |
| url | String | 是 |  | 支付链接 |
| validStartTime | String | 是 |  | 有效的起始时间 |

#### 3.34.5 接口入参示例

入参密文如下

{

"account": "testAccount",

"timestamp": "2015-08-03 00:00:00",

"data":Base64ResultTest

}

入参明文如下

{

"account": "testAccount",

"sign": "testsign",

"timestamp": "2015-08-03 00:00:00",

"data": {

"orderId": "tn16121781547273"

}

}

#### 3.34.6 接口出参示例

出参密文如下

{

"success": true,

"returnCode": 231000,

"errorMsg": "",

"data": " Base64ResultTest"

}

出参明文如下

{

"success": true,

"returnCode": 231000,

"errorMsg": "",

"data": [

{

"type": 1,

"url": "http://10.10.32.160/fb2/t1/group1/M00/05/33/Cgogo1mNJJWIeaMJAAAKe-fantgAAABSQODVM4AAAqT61.html",

"validStartTime": "2017-09-19 16:18:23"

},

{

"type": 2,

"url": "http://10.10.32.160/fb2/t1/group1/M00/05/33/Cgogo1mNJJWIeaMJAAAKe-fantgAAABSQODVM4AAAqT61.html",

"validStartTime": "2017-09-19 16:18:23"

}

]

}

#### 3.34.7 通用输出参数可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231099 | 系统异常 |

### 3.35 查询指定订单的出票（或改签出票）状态

#### 3.35.1接口说明

此接口用于套皮支付时查询订单的出票(或改签出票)状态（包含待支付、出票成功、出票失败）

#### 3.35.2 接口调用方法

Method：HTTP-POST

Url：{API\_Url}/train/ queryConfirmStatus

DataType：json

#### 3.35.3 接口入参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| orderId | String | 是 | tn16121781547273 | 订单号 |
| confirmType | Integer | 是 |  | 出票类型（1、出票 2、改签出票） |

#### 3.35.4 接口出参

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| confirmStatus | Integer | 是 |  | 1、待支付 2、出票成功 3、出票失败 |

#### 3.35.5 接口入参示例

入参密文如下

{

"account": "testAccount",

"timestamp": "2015-08-03 00:00:00",

"data":Base64ResultTest

}

入参明文如下

{

"account": "testAccount",

"sign": "testsign",

"timestamp": "2015-08-03 00:00:00",

"data": {

"orderId": "tn16121781547273",

"confirmType":1

}

}

#### 3.35.6 接口出参示例

出参密文如下

{

"success": true,

"returnCode": 231000,

"errorMsg": "",

"data": " Base64ResultTest"

}

出参明文如下

{

"success": true,

"returnCode": 231000,

"errorMsg": "",

"data": {

" confirmStatus ": 1

}

}

#### 3.35.7 通用输出参数可能出现的值

|  |  |
| --- | --- |
| **返回码** | **错误信息** |
| 231099 | 系统异常 |

## 4. 附录

### 4.1 API返回码

|  |  |  |
| --- | --- | --- |
| **返回码** | **错误信息** | **备注** |
| 231000 | success | 请求成功 |
| 231001 | user not exists | 请求用户不存在 |
| 231002 | No permission to do this operation | 用户没有该接口访问权限 |
| 231003 | Unauthorized client IP address | IP未授权 |
| 231004 | high frequency error | 访问频率过高 |
| 231005 | visit num limit | 访问次数达到限制 |
| 231006 | timestamp error | 时间戳异常 |
| 231007 | signature error | 签名异常 |
| 231008 | param error | 入参异常 |
| 231009 | inter not exist | 接口不存在 |
| 231099 | unknown error | 未知异常 |

### 4.2完整请求示例

#### 4.2.1签名生成流程

**假设用户密钥为wUDSCOdFibEL6pIQGYgF，请求入参如下：**

account：Ape2hqlBF0sFUUcjbj

timestamp：2015-08-15 18:16:32

agencyProductId：12345

info:

planDateStr:test

1. **首先，我们要按照key值首字母排序，并去掉value为空的，得到以下参数：**

agencyProductId：12345

account：Ape2hqlBF0sFUUcjbj

planDateStr:test

timestamp：2015-08-15 18:16:32

1. **将参数名、参数值依次拼接在一起**

agencyProductId12345accountApe2hqlBF0sFUUcjbjplanDateStrtesttimestamp2015-08-15 18:16:32

1. **在开头和结尾分别加上密钥**

wUDSCOdFibEL6pIQGYgFagencyProductId12345accountApe2hqlBF0sFUUcjbjplanDateStrtesttimestamp2015-08-15 18:16:32wUDSCOdFibEL6pIQGYgF

1. **进行MD5加密，并将结果全部转为大写，获得签名值**

C81E00F2CCC228DD8C1D3EB6C69B4260

1. **签名生成结束，请求入参变为**

account：Ape2hqlBF0sFUUcjbj

timestamp：2015-08-15 18:16:32

agencyProductId：12345

info:

planDateStr:test

sign:C81E00F2CCC228DD8C1D3EB6C69B4260

1. **以上只是生成签名，真实请求中还需要将上面的所有入参统一转成json格式**

#### 4.2.2 java版生成签名demo

// 二进制转化为大写的十六进制

public static String byte2hex(byte[] bytes) {

StringBuilder sign = new StringBuilder();

for (inti = 0; i<bytes.length; i++) {

String hex = Integer.toHexString(bytes[i] & 0xFF);

if (hex.length() == 1) {

sign.append("0");

}

sign.append(hex.toUpperCase());

}

returnsign.toString();

}

#### 4.2.3 php版生成签名demo

#### 4.2.4.net版生成签名demo

### 4.3 自定义数据类型说明

#### 4.3.1 passengers（乘客信息）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| passengerId | int | Y |  | 乘客的顺序号 |
| ticketNo | string | N |  | 票号（此票在本订单中的唯一标识，**订票成功后才有值**） |
| oldTicketNo | string | N |  | 改签前的票号（改签申请时必须有值） |
| newTicketNo | string | N |  | 改签后新车票的票号（改签成功后有值） |
| passengerName | string | Y |  | 乘客姓名 |
| insureNumber | string | N |  | 保单号（购买保险情况下有值） |
| insurePrice | string | N |  | 保险价格（购买保险情况下有值） |
| passportNo | string | Y |  | 乘客证件号码 |
| passportTypeId | string | Y |  | 证件类型ID  与名称对应关系:  1:二代身份证，2:一代身份证，C:港澳通行证，G:台湾通行证，B:护照更多参见证件编码对应表 |
| passportTypeName | string | Y |  | 证件类型名称 |
| piaoType | string | Y |  | 票种ID。  与票种名称对应关系：  1:成人票，2:儿童票，3:学生票，4:残军票 |
| piaoTypeName | string | Y |  | 票种名称 |
| zwCode | string | Y |  | 座位编码。  与座位名称对应关系：  9:商务座，P:特等座，M:一等座，O:二等座，6:高级软卧，  4:软卧，3:硬卧，2:软座，1:硬座更多参见座位编码对应表  注意：当最低的一种座位，无票时，购买选择该座位种类，买下的就是无座(也就说买无座的席别编码就是该车次的最低席别的编码)，另外，当最低席别的票卖完了的时候才可以卖无座的票。 |
| zwName | string | N |  | 座位名称 |
| cxin | string | N |  | 几车厢几座  （在订票成功后**或改签成功后**才会有值） |
| price | string | Y |  | 票价 |
| reason | int | N |  | 身份核验状态 0：正常 1：待审核 2：未通过 |
| provinceCode | String | N |  | 省份编号【学生票时必填】 |
| provinceName | String | N |  | 省份名称【学生票时选填】 |
| schoolCode | String | N |  | 学校代号【学生票时选填】 |
| schoolName | String | N |  | 学校名称【学生票时选填】 |
| studentNo | String | N |  | 学号【学生票时必填】 |
| schoolSystem | String | N |  | 学制【学生票时必填】 |
| enterYear | String | N |  | 入学年份:yyyy【学生票时必填】 |
| preferenceFromStationName | String | N |  | 优惠区间起始地名称【学生票时选填】 |
| preferenceFromStationCode | String | N |  | 优惠区间起始地代号【学生票时必填】 |
| preferenceToStationName | String | N |  | 优惠区间到达地名称【学生票时选填】 |
| preferenceToStationCode | String | N |  | 优惠区间到达地代号【学生票时必填】 |

乘客信息说明如下：

A,调用请求占座且车票类型为学生票时蓝色字体字段需要填写，注：蓝色字体字段仅供输入，不用于接口输出

B, oldTicketNo在改签申请和改签异步反馈时必须有值，newTicketNo在改签成功之后才有值，改签时ticketNo可以有值也可无值，车票号依据oldTicketNo和newTicketNo为准

#### 4.3.2zwcode（座位编码）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| 0 |  |  |  | 棚车 |
| 1 |  |  |  | 硬座 |
| 2 |  |  |  | 软座 |
| 3 |  |  |  | 硬卧 |
| 4 |  |  |  | 软卧 |
| 5 |  |  |  | 包厢硬卧 |
| 6 |  |  |  | 高级软卧 |
| 7 |  |  |  | 一等软座 |
| 8 |  |  |  | 二等软座 |
| 9 |  |  |  | 商务座 |
| A |  |  |  | 高级动卧 |
| B |  |  |  | 混编硬座 |
| C |  |  |  | 混编硬卧 |
| D |  |  |  | 包厢软座 |
| E |  |  |  | 特等软座 |
| F |  |  |  | 动卧 |
| G |  |  |  | 二人软包 |
| H |  |  |  | 一人软包 |
| I |  |  |  | 一等双软 |
| J |  |  |  | 二等双软 |
| K |  |  |  | 混编软座 |
| L |  |  |  | 混编软卧 |
| M |  |  |  | 一等座 |
| O |  |  |  | 二等座 |
| P |  |  |  | 特等座 |
| Q |  |  |  | 观光座 |
| S |  |  |  | 一等包座 |

#### 4.3.3passporttypeseid（证件编码）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 示例值 | 描述 |
| 1 |  |  |  | 二代身份证 |
| 2 |  |  |  | 一代身份证 |
| C |  |  |  | 港澳通行证 |
| G |  |  |  | 台湾通行证 |
| B |  |  |  | 护照 |
| H |  |  |  | 外国人居留证 |