

同程机票运价直连文档

Version No.1.4

版本修改历史

版本	发布日期	修改内容
V1.0	2017.06.26	新建
V1.0.1	2017.08.03	更新 search 接口样例,生单接口错误的属性
V1.1	2017.09.30	新增订单取消接口
V1.2	2017.12.12	规则说明 5,6 增加 base64 编码相关. 新增 java 测试 demo 一个
V1.3	2018.01.05	更新 search,verify,order,paycheck,ticket 接口对应请求响应
V1.4	2018.02.06	规范相关描述

目 录

- 一、 规则说明6
- 二、 搜索接口(SEARCH)8
 - 1. 请求参数8
 - 2. 请求样例8
 - 3. 返回参数9
 - 4. 返回样例11
- 三、 价格校验接口(VERIFY).....13
 - 1. 请求参数13
 - 2. 请求样例13
 - 3. 返回参数15
 - 4. 返回样例15
- 四、 生单接口(ORDER)17
 - 1. 请求参数17
 - 2. 请求样例18
 - 3. 返回参数19
 - 4. 返回样例20
- 四、 订单取消接口 (CANCELORDER)21

1. 请求参数.....	21
2. 请求样例.....	21
3. 返回参数.....	21
4. 返回样例.....	22
五、支付校验接口(PAYCHECK)	22
1. 请求参数说明	22
2. 请求样例	22
3. 返回参数说明	24
4. 返回样例	24
六、出票确认接口(TICKET)	25
1. 请求参数说明	25
2. 请求样例	25
3. 返回参数说明	26
4. 返回样例	26
5. 其他说明	26
七、票号通知接口(SETORDERISSUED)	26
1. 接口说明	26
2. 请求参数说明	26
3. 请求样例	27
4. 返回参数说明	27

5. 返回样例	27
八、 附录	28
附录一（ 返回错误码 ）	28
附录二(AES 加密 DEMO).....	29
附录三(接入查询 DEMO).....	34

一、规则说明

- 1. 提供接口 http 访问 url , 统一字符编码 : UTF-8 ;
- 2. http 接口请求 content-Type="text/plain" ;
- 3. 接口通过 http post 请求接口 url 并发送符合 JSON 规范的 body ;
- 4. 生单传输数据须做 AES 加密后进行 BASE64 转码 , 密钥满足 128 位。生产环境密钥使用 PID 安全码 ;
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 :
8Z3dZzqn05FmiuBLOWExK0CAbs4TY2GorC2dDPVIsn/tP+VuJGePqIMv1uSaVErr
- 5. 接口入参用 GZIP 压缩 params,再用 base64 编码节点 ;
- 6. 接口出参为 base64 编码后再压缩的 json 字符串,客户端需要先解码 base64,再解压缩 ;
- 7. 接口地址描述
地址 : <http://tcflightopenapi.17uoft.com/flightdistributeapi/dispatcher/api>
注 : PID 和安全码单独提供。

8. 请求公共参数及安全验证 :

参数名	名称	元素描述	类型	必填	备注
serviceCode	业务码	用于确定本次接口访问需要完成的业务	String	yes	eg:get_interbenefit_pnrcontent
pid	合作伙伴标示	用于区分合作伙伴	String	yes	由同程提供
sign	校验码	用于保证数据通信安全	String	yes	使用方式参见“概述”中说明
requestID	请求 ID	请求唯一码	String	yes	
timestamp	时间戳		String	yes	时间戳 , 格式为 yyyy-MM-dd HH:mm:ss , 时区为 GMT+8 , 例如 : 2016-01-01 12:00:00

同程国际机票运价直连文档

businessType	业务类型		string	yes	0 国内 1 国际, 默认为 0
params	请求参数内容		string	yes	

安全验证 : Sign 加 IP 白名单, 其中 sign 值构建如下 :

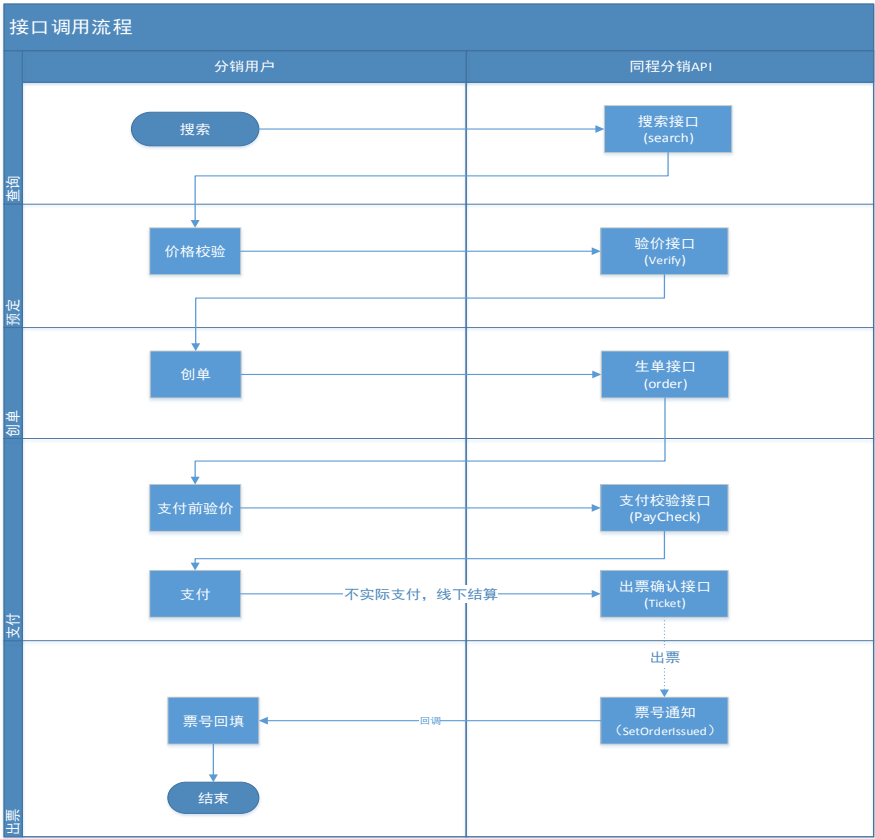
Sign=Md5[Params 字符串+ timestamp + PID 安全码+Pid]

- 1) params 是请求参数中的 params 压缩前 json 字符串
- 2) 参数值签名计算的时候保证传递的参数和签名计算大小写一致
- 2) 安全码在使用时候, 不能进行大小写转换
- 3) 时间戳 : yyyy-MM-dd HH:mm:ss, 时区为 GMT+8eg:

MD5(paramsjsonstr2017-03-18

12:30:338a62a930f473481b8c22f2c9727673e50008605511503185427)

9. 接口调用流程, 如下 :



二、 搜索接口(Search)

1. 请求参数

参数	必须项	类型	详细说明
productType	yes	int	1 优选 2 特惠
tripType	yes	string	行程类型, 1: 单程; 2: 往返;
fromCity	yes	string	出发地城市 IATA 三字码代码
toCity	yes	string	目的地城市 IATA 三字码代码
fromDate	yes	string	出发日期, 格式为 yyyyMMdd
retDate	no	string	回程日期, 格式为 yyyyMMdd (留空表示单程)
adultNumber	no	int	1. 没有该字段时, 默认乘客类型为成人, 乘机总人数为 1 (即: 1 个成人) 2. 有该字段时, 代表乘客类型为成人的乘机总人数
childNumber	no	int	1. 没有该字段时, 默认为 0 个儿童 2. 有该字段时, 代表乘客类型为儿童的乘机总人数
carrier	no	string	指定航司进行查询
cabinGrade	no	string	头等: F, 商务: C, 超经: S, 经济: Y

2. 请求样例

```
{
  "serviceCode": "Search",
  "pid": 1234567,
  "sign": "(根据规则说明计算)",
  "requestID": 123,
  "timestamp": "2017-07-14 12:12:30",
  "businessType": 1,
  "params": {
    "adultNumber": 1,
    "childNumber": 0,
    "fromCity": "SHA",
    "fromDate": "20180212",
    "productType": "1",
    "retDate": "",
    "toCity": "HKG",
    "tripType": "1",
```



```

    "carrier": "MU",
    "cabinGrade": "Y"
  }
}

```

3. 返回参数

参数	类型	说明
code	string	LY000000, 成功；其他，失败。
msg	string	提示信息，长度小于 64。
 routings	object	报价信息，参考下面的 Routing Element
Routing Element		
data	string	可保存必要信息，之后生单按原值回传。最大 1000 个字符
adultPrice	int	成人单价（以 CNY 为单位），不含税或不含基建燃油
adultTax	int	成人税费（以 CNY 为单位）
childPrice	int	儿童单价（以 CNY 为单位），不含税或不含基建燃油
childTax	int	儿童税费（以 CNY 为单位）
rule	object	退改签信息，参考 Rule Element
fromSegments	array	去程航段按顺序，参考下面的 Segment Element
retSegments	array	回程航段按顺序，参考下面的 Segment Element（单程搜索为空）
validatingCarrier	string	出票航司
Rule Element		
baggageInfoList	array	行李额规定。参考下方 Baggage Element，
refundInfoList	array	退票规定。参考下面的 Refund Element
changesInfoList	array	改期规定。参考下面的 Changes Element
note	string	备注信息，最大 300 个字符
Baggage Element		
adultBaggage	string	成人行李额，格式重量+单位 如 15KG，默认值为空字符串
childBaggage	string	儿童行李额，格式重量+单位 如 15KG，默认值为空字符串
segmentNo	int	航 段 序 号 ： 从 1 至 (fromSegments.Count+retSegments.Count)的数值
Refund Element		
refundType	int	退票类型 0：客票全部未使用； 1：客票部分使用；
refundStatus	string	退票标识 T：不可退 H：有条件退 F：免费退

refundFee	decimal	当 refundStatus =H, 必须赋值
currency	string	退票费币种, 当 refundStatus =H, 必须赋值。IATA 标准币种编码, 例如人民币: CNY
cnRemark	string	当 RefundStatus=H 时, 中文说明
enRemark	string	当 RefundStatus=H 时, 英文说明
Changes Element		
changesType	int	改期类型 0: 起飞前改期; 1: 起飞后改期; 注: 上述改期同时包含 Revalidation 和 Reissue 部分
changesStatus	string	改期标识 T: 不可改期 H: 有条件改期 F: 免费改期
changesFee	decimal	H 有条件改时改期费 当 changesStatus =H, 必须赋值
currency	string	改期费币种, 当 changesStatus =H, 必须赋值, IATA 标准币种编码, 例如人民币: CNY
cnRemark	string	当 changesStatus =N 时, 中文说明
enRemark	string	当 changesStatus =N 时, 英文说明
Segment Element		
segmentNo	int	航段序号
flightType	String	去程: GO 返回: BACK
carrier	string	航司 IATA 二字码, 必须与 flightNumber 航司相同
flightNumber	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	string	出发机场 IATA 三字码
depTerminal	string	出发航站楼, 使用简写, 例如 T1
depTime	string	起飞日期时间, 格式: YYYYMMDDHHMM 例: 201703100300 表示 2017 年 3 月 10 日 3 时 0 分
arrAirport	string	到达机场 IATA 三字码
arrTerminal	string	抵达航站楼, 使用简写, 例如 T1
arrTime	string	起飞日期时间, 格式: YYYYMMDDHHMM 例: 201703101305 表示 2017 年 3 月 10 日 13 时 5 分
codeShare	boolean	代码共享标识 (true 代码共享/false 非代码共享)
cabin	string	舱位
cabinGrade	string	舱等, 头等: F, 商务: C, 超经: S, 经济: Y
aircraftCode	string	机型, IATA 标准 3 字码, 并过滤下列机型运价信息 BUS ICE LCH LMO MTL RFS TGV THS THT TRN TSL
duration	int	飞行时长, 单位为分钟, 通过时差转换后的结果
stopTime	int	经停总时长

stops	List<string>	经停信息, 经停信息,只包含机场三字码, 默认为 null
operatingCarrier	String	实际承运航司
operatingFlightNumber	String	实际承运航班

4. 返回样例

返回成功样例

```
{
  "code": "LY000000",
  "msg": "请求成功",
  "routings": [{
    "adultPrice": 2680,
    "adultTax": 618,
    "childPrice": 0,
    "childTax": 0,
    "data":
"a9b46954-029d-4e5e-984c-ec5016d1e50a_5c2febd5-d943-4e9f-9d3c-7808994c8a31;10W,1,,,S1,a9b9915a-3a39-4f81-ae
dd-f7878ef43b10S1SHAGMPICNHKG",
    "fromSegments": [{
      "aircraftCode": "333",
      "arrAirport": "GMP",
      "arrTerminal": "",
      "arrTime": "201802121100",
      "cabin": "S",
      "cabinGrade": "Y",
      "carrier": "KE",
      "codeShare": true,
      "depAirport": "SHA",
      "depTerminal": "T1",
      "depTime": "201802120830",
      "duration": 90,
      "flightNumber": "KE5862",
      "flightType": "GO",
      "operatingCarrier": "MU",
      "operatingFlightNumber": "MU511",
      "segmentNo": 1,
      "stopTime": 0,
      "stops": []
    ]
  }
}
```

```

    },
    {
      "aircraftCode": "77W",
      "arrAirport": "HKG",
      "arrTerminal": "1",
      "arrTime": "201802122255",
      "cabin": "S",
      "cabinGrade": "Y",
      "carrier": "KE",
      "codeShare": false,
      "depAirport": "ICN",
      "depTerminal": "2",
      "depTime": "201802122000",
      "duration": 235,
      "flightNumber": "KE607",
      "flightType": "GO",
      "operatingCarrier": "KE",
      "operatingFlightNumber": "KE607",
      "segmentNo": 2,
      "stopTime": 0,
      "stops": []
    }
  ],
  "rule": {
    "baggageInfoList": [
      {
        "adultBaggage": "一共 2 件",
        "childBaggage": "一共 2 件",
        "segmentNo": 1
      },
      {
        "adultBaggage": "一共 2 件",
        "childBaggage": "一共 2 件",
        "segmentNo": 2
      }
    ],
    "changesInfoList": [
      {
        "changesFee": "0.0",
        "changesStatus": "H",
        "changesType": "0",
        "cnRemark": "起飞前 允许改期 收取 400.0CNY;起飞后 允许改期 收取 400.0CNY;改期另需

```

```
加收服务费 210.00 元人民币;";
    },
    "note": "起飞前 允许误机 免费;起飞后 允许误机 免费;",
    "refundInfoList": [{
        "cnRemark": "起飞前 允许退票 收取 650.0CNY;起飞后 允许退票 收取 650.0CNY;退票另需
加收服务费 210.00 元人民币;";
        "refundFee": "0.0",
        "refundStatus": "H",
        "refundType": "0"
    }]
},
"validatingCarrier": "KE"
},{.....}]
}
```

三、 价格校验接口(Verify)

1. 请求参数

参数	必须项	类型	详细说明
tripType	yes	int	行程类型，1：单程；2：往返；
adultNumber	no	int	1. 没有该字段时，默认乘客类型为成人，乘机总人数为 1 （即：1 个成人） 2. 有该字段时，代表乘客类型为成人的乘机总人数
childNumber	no	int	1. 没有该字段时，默认为0个儿童 2. 有该字段时，代表乘客类型为儿童的乘机总人数
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

2. 请求样例

```
{
  "serviceCode": "Verify",
  "pid": 1234567,
  "sign": "value",
}
```

```

"requestID": 123,
"timestamp": "2017-07-14 12:12:30",
"businessType": 1,
"params": {
  "adultNumber": 1,
  "childNumber": 0,
  "routing": {
    "data":
"a9b46954-029d-4e5e-984c-ec5016d1e50a_5c2febd5-d943-4e9f-9d3c-7808994c8a31;10W,1,,,S1,a9b9915a-3a39-4f81-ae
dd-f7878ef43b10S1SHAGMPICNHKG",
    "fromSegments": [{
      "aircraftCode": "333",
      "arrAirport": "GMP",
      "arrTerminal": "",
      "arrTime": "201802121100",
      "cabin": "S",
      "cabinGrade": "Y",
      "carrier": "KE",
      "codeShare": true,
      "depAirport": "SHA",
      "depTerminal": "T1",
      "depTime": "201802120830",
      "duration": 90,
      "flightNumber": "KE5862",
      "flightType": "GO",
      "operatingCarrier": "MU",
      "operatingFlightNumber": "MU511",
      "segmentNo": 1,
      "stopTime": 0,
      "stops": []
    },
    {
      "aircraftCode": "77W",
      "arrAirport": "HKG",
      "arrTerminal": "1",
      "arrTime": "201802122255",
      "cabin": "S",
      "cabinGrade": "Y",

```

```
        "carrier": "KE",
        "codeShare": false,
        "depAirport": "ICN",
        "depTerminal": "2",
        "depTime": "201802122000",
        "duration": 235,
        "flightNumber": "KE607",
        "flightType": "GO",
        "operatingCarrier": "KE",
        "operatingFlightNumber": "KE607",
        "segmentNo": 2,
        "stopTime": 0,
        "stops": []
      }
    },
    "tripType": "1"
  }
}
```

3. 返回参数

参数	类型	说明
code	string	LY000000, 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 20 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。
rule	object	退改签信息，参考搜索返回结果中的 Rule Element

4. 返回样例

返回成功样例

```
{
  "code": "LY000000",
  "maxSeats": 9,
  "msg": "请求成功",
  "routing": {
    "adultPrice": 2680,
    "adultTax": 618,
    "childPrice": 0,
```

```

"childTax": 0,

"data":

"a9b46954-029d-4e5e-984c-ec5016d1e50a_5c2febd5-d943-4e9f-9d3c-7808994c8a31;10W,1,,,S1,a9b9915a-3a39-4f81-ae
dd-f7878ef43b10S1SHAGMPICNHKG",

"fromSegments": [{

  "aircraftCode": "333",

  "arrAirport": "GMP",

  "arrTerminal": "",

  "arrTime": "201802121100",

  "cabin": "S",

  "cabinGrade": "Y",

  "carrier": "KE",

  "codeShare": true,

  "depAirport": "SHA",

  "depTerminal": "T1",

  "depTime": "201802120830",

  "duration": 90,

  "flightNumber": "KE5862",

  "flightType": "GO",

  "operatingCarrier": "MU",

  "operatingFlightNumber": "MU511",

  "segmentNo": 1,

  "stopTime": 0,

  "stops": []

}],

{

  "aircraftCode": "77W",

  "arrAirport": "HKG",

  "arrTerminal": "1",

  "arrTime": "201802122255",

  "cabin": "S",

  "cabinGrade": "Y",

  "carrier": "KE",

  "codeShare": false,

  "depAirport": "ICN",

  "depTerminal": "2",

  "depTime": "201802122000",

  "duration": 235,

```



```
        "flightNumber": "KE607",
        "flightType": "GO",
        "operatingCarrier": "KE",
        "operatingFlightNumber": "KE607",
        "segmentNo": 2,
        "stopTime": 0,
        "stops": []
    }
},
"sessionId": "abe210d5-5c6e-4de3-b91e-621d13623ce8"
}
```

四、 生单接口(Order)

1. 请求参数

参数	必须项	类型	详细说明
tripType	yes	string	行程类型，1：单程；2：往返；
sessionId	yes	string	会话标识：标记服务接口调用的唯一标识，会将价格校验接口中的 sessionId 原值返回。数字或字母，长度小于 20 个字符
routing	yes	string	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
passengers	yes	array	乘机人信息，参考下面的 Passenger Element
contract	yes	object	联系人信息，参考下面的 Contract Element
Passengers Element 乘机人信息			
name	yes	string	国际：LastName/FirstName MiddleName，姓/名，如：ZHANG/SAN
passengerType	yes	string	乘客类型，ADT 成人/CHD 儿童
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 18 个字符
cardType	yes	string	证件类型： ID - 身份证（国内） 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 QT - 其它 注：国内只能使用 ID/PP/QT
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
mobile	yes	string	乘客手机号
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
mobile	no	string	联系人手机号

2. 请求样例

```
{
  "serviceCode": "Order",
  "pid": 1234567,
  "sign": "value",
  "requestID": 123,
  "timestamp": "2017-07-14 12:12:30",
  "businessType": 1,
  "params": {
    "contract": {
      "address": "成都市九眼桥",
      "email": "123123@163.com",
      "mobile": "18382479999",
      "name": "温温",
      "postcode": "30033"
    },
    "passengers": [{
      "name": "will/liu",
      "birthday": "19700323",
      "cardExpired": "20201010",
      "cardIssuePlace": "AB",
      "cardNum": "14010919931",
      "cardType": "NI",
      "gender": "M",
      "mobile": "13438169774",
```

```
    "nationality": "CA",
    "passengerType": "ADT"
  }},
  "routing": {
    "data":
"954b1ede-c5ac-4197-9e25-5d53dd4d621e_04a955d1-d996-4cbc-8af6-b6f82364650d;10W,1,,,S1,92258a77-8e15-41ea-a4
c4-4785e8a61087S1PVGHKG",
    "fromSegments": [{
      "aircraftCode": "33V",
      "arrAirport": "HKG",
      "arrTerminal": "1",
      "arrTime": "201802121945",
      "cabin": "E",
      "cabinGrade": "Y",
      "carrier": "MU",
      "codeShare": true,
      "depAirport": "PVG",
      "depTerminal": "T2",
      "depTime": "201802121645",
      "duration": 180,
      "flightNumber": "MU8991",
      "flightType": "GO",
      "operatingCarrier": "HX",
      "operatingFlightNumber": "HX241",
      "segmentNo": 1,
      "stopTime": 0,
      "stops": []
    ]
  },
  "sessionId": "1d5a40b1-c8d7-4666-b7a1-d774dd1b29ff",
  "tripType": "1"
}
```

3. 返回参数

参数	类型	说明
code	string	LY000000, 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，将请求中的 sessionId 原值返回。数字或字母，长度小于 20 个字符
orderNo	string	订单号，最大 100 个字符
pnrCode	string	PNR 编码，最大 6 个字符
routing	array	报价信息，参考搜索返回结果中的 Routing Elements

4. 返回样例

```
{
  "code": "LY000000",
  "msg": "请求成功",
  "orderNo": "TI1801051127402UK10W",
  "pnrCode": "HY96ZX",
  "routing": {
    "adultPrice": 2200,
    "adultTax": 124,
    "childPrice": 0,
    "childTax": 0,
    "data":
"954b1ede-c5ac-4197-9e25-5d53dd4d621e_04a955d1-d996-4cbc-8af6-b6f82364650d;10W,1,,,S1,92258a77-8e15-41ea-a4c4-4785e8a61087S1PVGHKG",
    "fromSegments": [{
      "aircraftCode": "33V",
      "arrAirport": "HKG",
      "arrTerminal": "1",
      "arrTime": "201802121945",
      "cabin": "E",
      "cabinGrade": "Y",
      "carrier": "MU",
      "codeShare": true,
      "depAirport": "PVG",
      "depTerminal": "T2",
      "depTime": "201802121645",
      "duration": 180,
      "flightNumber": "MU8991",
      "flightType": "GO",
      "operatingCarrier": "HX",
      "operatingFlightNumber": "HX241",
```

```
        "segmentNo": 1,
        "stopTime": 0,
        "stops": []
    }
},
"sessionId": "1d5a40b1-c8d7-4666-b7a1-d774dd1b29ff"
}
```

五、订单取消接口 (CancelOrder)

1 . 请求参数

可以主动发起订单请求，如果超时未支付，同程平台会自动取消，超时时间为 30min。

参数	必须项	类型	详细说明
orderNo	yes	string	生成订单成功时返回的订单号
Pnr	yes	string	编码
reason	No	string	订单取消原因

2 . 请求样例

```
{
  "serviceCode": " CancelOrder ",
  "pid": 1234567,
  "sign": "value",
  "requestID": 123,
  "timestamp": "2017-07-14 12:12:30",
  "businessType": 0,
  "params": {
    "orderNo": "B20170727134653QE102",
    "pnr": "HKSELC",
    "reason": "取消原因"
  }
}
```

3 . 返回参数

参数	类型	说明
code	string	LY000000, 成功；其他，失败。
msg	string	提示信息
orderNo	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	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。

4 . 返回样例

```
{
  "code": "LY000000",
  "msg": "success",
  "orderNo": "B20170727134653QE102",
  "pnrCode": "HKSDRE",
}
```

六、支付校验接口(PayCheck)

1. 请求参数说明

参数	必须项	类型	详细说明
tripType	yes	string	行程类型，1：单程； 2：往返；
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 字符，且不能为空
orderNo	yes	string	生成订单成功时返回的订单号
externalOrderNo	no	string	外部订单号(代理商自己产生的订单号)
pnrCode	no	string	生成订单成功时返回的 PNR 编码
routing	yes	object	报价信息，参考生单返回结果中的 Routing Elements。 <u>只含航班信息，不含价格信息</u>

2. 请求样例

```
{
```

```

"serviceCode": "PayCheck",

"pid": 1234567,

"sign": "value",

"requestID": 123,

"timestamp": "2017-07-14 12:12:30",

"businessType": 1,

"params": {

  "externalOrderNo": "2018127051127444075",

  "orderNo": "TI1801051127402UK10W",

  "pnrCode": "HY96ZX",

  "routing": {

    "data":

"954b1ede-c5ac-4197-9e25-5d53dd4d621e_04a955d1-d996-4cbc-8af6-b6f82364650d;10W,1,,,S1,92258a77-8e15-41ea-a4

c4-4785e8a61087S1PVGHKG",

    "fromSegments": [{

      "aircraftCode": "33V",

      "arrAirport": "HKG",

      "arrTerminal": "1",

      "arrTime": "201802121945",

      "cabin": "E",

      "cabinGrade": "Y",

      "carrier": "MU",

      "codeShare": true,

      "depAirport": "PVG",

      "depTerminal": "T2",

      "depTime": "201802121645",

      "duration": 180,

      "flightNumber": "MU8991",

      "flightType": "GO",

      "operatingCarrier": "HX",

      "operatingFlightNumber": "HX241",

      "segmentNo": 1,

      "stopTime": 0,

      "stops": []

    ]

  },

  "sessionId": "1d5a40b1-c8d7-4666-b7a1-d774dd1b29ff",

  "tripType": "1"

```

```
}  
}
```

3. 返回参数说明

参数	类型	说明
code	string	LY000000, 成功；其他，失败。
msg	string	提示信息
sessionId	string	会话标识：标记服务接口调用的唯一标识，将请求中的 sessionId 原值返回。数字或字母，长度小于 5 个字符，且不能为空
orderNo	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	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	object	报价信息，参考生单返回结果中的 Routing Elements。

4. 返回样例

返回成功样例：

```
{  
  "code": "LY000000",  
  "msg": "请求成功",  
  "orderNo": "TI1801051127402UK10W",  
  "pnrCode": "HY96ZX",  
  "routing": {  
    "adultPrice": 2200,  
    "adultTax": 124,  
    "childPrice": 0,  
    "childTax": 0,  
    "data":  
    "954b1ede-c5ac-4197-9e25-5d53dd4d621e_04a955d1-d996-4cbc-8af6-b6f82364650d;10W,1,,,S1,92258a77-8e15-41ea-a4c4-4785e8a61087S1PVGHKG"  
  },  
  "sessionId": "1d5a40b1-c8d7-4666-b7a1-d774dd1b29ff"  
}
```


七、出票确认接口(Ticket)

此接口适用于线下与同程结算用户，一旦调用此接口表示确认需要出票，操作请谨慎。

1. 请求参数说明

参数	必须项	类型	详细说明
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 字符，且不能为空
data	yes	string	可保存必要信息，之后生单按原值回传。最大1000 个字符
orderNo	yes	string	平台订单号
externalOrderNo	no	string	外部订单号(代理商自己产生的订单号)
pnrCode	yes	string	PNR 编码
status	yes	int	3：已成单 待出票 4：已取消
price	yes	Decimal	订单票面价总和
tax	yes	Decimal	订单税费总和/基建+燃油

批注 [同程网络-杨洪波1]: 去掉

2. 请求样例

```
{
  "serviceCode": "Ticket",
  "pid": 1234567,
  "sign": "value",
  "requestID": 123,
  "timestamp": "2017-07-14 12:12:30",
  "businessType": 1,
  "params": {
    "data":
      "954b1ede-c5ac-4197-9e25-5d53dd4d621e_04a955d1-d996-4cbc-8af6-b6f82364650d;10W,1,,,S1,92258a77-8e15-41ea-a4c4-4785e8a61087S1PVGHKG",
    "externalOrderNo": "2018127051127444075",
    "orderNo": "T11801051127402UK10W",
  }
}
```

```
{
  "pnrCode": "HY96ZX",
  "price": "2200",
  "status": "2",
  "sessionId": "1d5a40b1-c8d7-4666-b7a1-d774dd1b29ff",
  "tax": "124"
}
```

批注 [同程网络-杨洪波2]: 去掉

3. 返回参数说明

参数	类型	说明
code	string	LY000000, 成功；其他，失败。
msg	string	提示信息

4. 返回样例

```
{
  "code": "LY000000",
  "msg": "请求成功"
}
```

5. 其他说明

超时或者请求失败，建议重试，不超过 5 次，如果收到成功信息后，则停止发送请求。

八、票号通知接口(SetOrderIssued)

1. 接口说明

- 1. 此接口为同程回调分销渠道，告知分销渠道出票信息；
- 2. 由分销商提供接收出票信息 URL 地址，同程将以 POST 方式进行回调，编码 UTF-8。

2. 请求参数说明

参数	必须项	类型	详细说明
externalOrderNo	no	string	外部订单号(代理商自己产生的订单号)

orderNo	yes	string	订单号
pnrCode	yes	string	PNR 编码，最大 6 个字符
ticketNoItems	yes	array	客票列表，参考下面的 TicketNoItems Element
TicketNoItems Element 客票信息			
psrName	yes	string	乘机人姓名
tktno	yes	string	票号（票号格式：结算码+“-”+票号；多个票号以“;”分割）
idNo	yes	string	证件号

3. 请求样例

```
{
  "serviceCode": "SetOrderIssued",
  "pid": 1234567,
  "sign": "value",
  "requestID": 123,
  "timestamp": "2017-07-14 12:12:30",
  "businessType": 1,
  "params": {
    "orderNo": 201706132541,
    "externalOrderNo": 123456789,
    "pnrCode": 123456,
    "ticketNoItems": [
      {
        "psrName": "张三",
        "tktno": 1234567890123,
        "idNo": 140109199303235100
      }
    ]
  }
}
```

4. 返回参数说明

参数	类型	说明
code	string	LY000000, 成功；其他，失败。
msg	string	提示信息

5. 返回样例

```
{
  "code": "LY000000",
  "msg": "success"
}
```

九、 附录

附录一（返回错误码）

```
错误码样例
{
  "code": "LYxxxxxx",
  "msg": "xxxxxx"
}
```

模块	code	code 描述	message
公共	LY000000	成功	请求成功
	LY000001	请求参数解析错误	参数解析错误
	LY000002	身份验证	非法 IP 请求无效/sign 验签不通过
	LY000003	接口权限验证	无该接口访问权限
	LY100001	不存在	pid 不存在
	LY100002	无效	pid 无效
	LY100003	产品类型无效	产品类型无效
	LY200001	系统异常	系统异常
搜索	LY300001	参数有效性	参数有效性校验不通过-tripType 不能为空 /fromCity 不能为空
	LY400001	查询无结果	查询数据无结果
	LY400002	查询数据异常	查询数据异常
	LY400003	单儿童查询验证	无成人陪伴儿童不允许查询
验价	LY300001	参数有效性	请求实体不能为空/data 不能为空/tripType 类型不支持—tripType:{0}
	LY300002	与上一步返回信息不匹配	传入信息与上步返回数据不匹配
	LY300003	航程信息与航程类型对比	itinerarys 与 tripType 传入不匹配—sessionId:{0}
	LY410001	验价失败	验价失败—sessionId:{0}
	LY410002	单儿童验价验证	无成人陪伴儿童不允许验价
下单	LY300001	参数有效性	请求实体不能为空/sessionId 不能为空/data

			不能为空
	LY300002	与上一步返回信息不匹配	传入信息与上步返回数据不匹配
	LY300003	航程信息与航程类型对比	itinerarys 与 tripType 传入不匹配—sessionId:{0}
	LY420001	下单失败	预定失败—sessionId:{0}
	LY420002	单儿童下单验证	无成人陪伴儿童不允许下单
	LY420003	乘客姓名重复验证	存在与{0}姓名、拼音或证件号相同的乘客
	LY420004	乘客数量验证	订单人数不得多于 9 人
	LY420005	乘客重复预定错误	乘客重复预定，导致预定失败—sessionId:{0}
支付前校验	LY300001	参数有效性	请求实体不能为空/sessionId 不能为空 /prices 不能为空
	LY300002	与上一步返回信息不匹配	传入信息与上步返回数据不匹配
	LY430001	校验失败	支付前校验异常，sessionId:{0},orderNo:{1}
	LY430002	订单状态	订单状态异常，sessionId:{0},orderNo:{1},orderStatus:{2}
	LY430003	支付超时	支付时间超时异常，sessionId:{0},orderNo:{1}
出票通知	LY300001	参数有效性	请求实体不能为空/sessionId 不能为空 /orderNo 不能为空
	LY440001	通知失败	打款通知异常，sessionId:{0},orderNo:{1}
	LY440002	订单状态	订单状态异常，sessionId:{0},orderNo:{1},orderStatus:{2}

附录二(AES 加密 Demo)

(一) 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 keySpec = new SecretKeySpec( raw, "AES" );
        javax.crypto.Cipher cipher = Cipher.getInstance( "AES/CBC/PKCS5Padding" );
        cipher.init( Cipher.ENCRYPT_MODE, keySpec, 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;
    }
}

/**
 * 解密
 *
 * @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 keySpec = new SecretKeySpec( raw, "AES" );

        Cipher cipher = Cipher.getInstance( "AES/CBC/PKCS5Padding" );
        cipher.init( Cipher.DECRYPT_MODE, keySpec, 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);

}

```

(二) 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;
    }

```


(三) 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 $text;  
    }  
    if (strpos($text, $text{mb_strlen($text) - 1}, mb_strlen($text) - $pad) != $pad) {  
        return $text;  
    }  
    return mb_substr($text, 0, -1 * $pad);  
}  
}
```

附录三(接入查询 Demo)



javaDemoApi.zip