

B2B续租对接文档

环境

分销续租对接接口

续租接单成功事件:103

续租接单失败事件:104

续租退费事件:142（新事件需要对接）

续租退费成功回调事件:270

退费失败回调事件:271

推送账单：

环境

http协议地址：<http://pre-gw.api.taobao.com/top/router/rest>，https协议地址：<https://pre-gw.api.taobao.com/top/router/rest>

奇门请求地址:

环境	HTTP地址	HTTPS地址
正式环境	http://qimen.api.taobao.com/top/router/qm	https://qimen.api.taobao.com/top/router/qm
沙箱环境	http://qimen.api.taobao.com/top/router/qmtest	https://qimen.api.taobao.com/top/router/qmtest

TOP请求地址:

环境	HTTP地址	HTTPS地址
正式环境	http://gw.api.taobao.com/top/router/rest	https://gw.api.taobao.com/top/router/rest
沙箱环境	http://pre-gw.api.taobao.com/top/router/rest	https://pre-gw.api.taobao.com/top/router/rest

分销续租对接接口

名称	接口	描述	文档链接
续租报价查询	alitrip.rentcar.btb.rentaldetail.query	根据订单详情查询是否支持续租发起续租报价查询	https://open.taobao.com/api.htm?docId=69026&docType=2
续租下单接口	alitrip.rentcar.btb.rental.submit	续租续租下单，同步返回飞猪续租单，异步返回接单成功、失败事件 (103：成功，104：失败) 需要对接：103：成功，104：失败接单事件	https://open.taobao.com/api.htm?docId=69025&docType=2
续租预取消接口	alitrip.rentcar.btb.rental.checkcancel	续租预取消，同步返回能否预取消，能：费用、收费原因，不能：错误码、错误信息	https://open.taobao.com/api.htm?docId=69024&docType=2
续租取消接口	alitrip.rentcar.btb.rental.order.cancel	续租取消，同步返回能否取消订单，能：费用、收费原因、取消订单，不能：错误码、错误信息	https://open.taobao.com/api.htm?docId=69023&docType=2
订单详情查询接口	alitrip.rentcar.btb.order.query	用主单查询此订单是否支持续租， service_type中新增3，标识能否发起续租	https://open.taobao.com/api.htm?docId=65892&docType=2

客服退款	qimen.alitrip.rentcar.tborder.notify	两种模式： 1.主单退款模式：账单需要用主单 2.续租单(子单)退款模式：账单需要用主单 需要对接续租退费新事件（142）	https://open.taobao.com/api.htm?docId=65822&docType=2
分销对账单数据同步接口	alitrip.rentcar.btb.bill.notify	续租单交易记录以主单订单进退对账， order_id:传飞猪主订单号/out_order_id:传分销主订单号 具体看：推送账单部分	https://open.taobao.com/api.htm?docId=66523&docType=2

续租接单成功事件:103

```
1  /**
2   * 接单成功:103,接单成功事件payload
3   * {
4   * "channelId":"80001",
5   * "orderId":"3187983531024266527",
6   * "outOrderId":"761448761000000016108099",
7   * "eventType":103,
8   * "eventData":{
9   *     "renewalOrderId":"3187983531024266527",
10  *     "createdTime":"2022-10-22 10:22:00"
11  * }
12  * }
13  */
14  @Data
15  public static class SubmitRenewalOrderSuccessEventData implements Platform
    CallbackEvent{
16      /**
17       * 增加交易流水号
18       * payTradeNo:"3187983531024266527_132_1",
19       */
20      //private String payTradeNo;
21
22      /**
23       * 飞猪续租订单号
24       * */
25      private Long renewalOrderId;
26
27      /**
28       * 订单创建时间
29       */
30      private Date createdTime;
31
32      /**
33       * @return
34       */
35      @Override
36      public Integer getEventType() {
37          return B2bPlatformCallbackEventTypeEnum.RENEWAL_ACCEPT_ORDER_SUCCE
            SS.getCode();
38      }
39  }
40
41
```

续租接单失败事件:104

```
1  /**
2   * 接单失败:104,接单失败事件payload
3   * {
4   *   "channelId":"80001",
5   *   "orderId":"3104050285204362836",
6   *   "outOrderId":"RC202212191604667539969978368",
7   *   "eventType":104,
8   *   "eventData":{
9   *     "renewalOrderId":3187983531024266522,
10  *     "errorCode":"71008",
11  *     "errorMessage":"商家接单失败"
12  *   },
13  *
14  * }
15  */
16  @Data
17  public static class SubmitRenewalOrderFailureEventData implements Platform
  CallbackEvent {
18
19      /**
20       * 飞猪续租订单号
21       */
22      private Long renewalOrderId;
23
24      /**
25       * errorCode
26       */
27      private String errorCode;
28
29      /**
30       * errorMessage
31       */
32      private String errorMessage;
33
34      /**
35       * @return
36       */
37      @Override
38      public Integer getEventType() {
39          return B2bPlatformCallbackEventTypeEnum.RENEWAL_ACCEPT_ORDER_FAILU
  RE.getCode();
40      }
41  }
```

续租退费事件:142（新事件需要对接）

需要回调退费事件(270,271)

```
1  /**
2   * 142:退费事件
3   * {
4   *   "channelId":"80001",
5   *   "orderId":"3104050285204362836",
6   *   "outOrderId":"RC202212191604667539969978368",
7   *   "eventType":141,
8   *   "eventData":{
9   *     "renewalOrderId":"3187983531024266527",
10    *     "renewalOutOrderId":"RC202212191604667539969978361",
11    *     "refundType":2,
12    *     "amount":"200.98",
13    *     "reason":"油费退款",
14    *     "payTradeNo":"202212191604667539969978368"
15    *   }
16  * }
17  *
18  * 需要回调退费事件
19  * @see RentCarB2bDistributionCallbackReq.RenewalRefundFeeSuccessCallbackEvent
20  * @see RentCarB2bDistributionCallbackReq.RenewalRefundFeeFailureCallbackEvent
21  */
22 @Data
23 @Accessors(chain = true)
24 public static class RenewalRefundFeeEventData implements PlatformCallbackEvent {
25     /**
26      * 续租单退费飞猪续租订单号:
27      * 1:客服退费具体续租单
28      */
29     private Long renewalOrderId;
30
31     /**
32      * 续租单退费分销续租订单号:
33      * 1:客服退费具体续租单
34      */
35     private String renewalOutOrderId;
36
37     /**
38      * 取消类型:
39      * 1:平台取消(用户通过客服取消)
40      * 2:商家取消(用户通过商家取消), 暂不支持
41      */
42     private String canceller;
```

```

43
44     /**
45      * 退款类型
46      * 1, 提前还车退款,
47      * 2, 油费退款,
48      * 3, 车损扣费退款,
49      * 4, 违章扣费退款,
50      * 5, 取消违约金退款(用于取消费用多扣用户不认同退费),
51      * -1, 其他
52      */
53     private Integer refundType;
54     /**
55      * 退款金额
56      * 单位:元, 200.98
57      */
58     private String amount;
59
60     /**
61      * 退款原因
62      * 分销商要求字段不能为空
63      */
64     @NotNull
65     private String reason;
66
67     /**
68      * 当前退款交易流水号
69      * payTradeNo:"2022120122001192491422107773",
70      */
71     private String payTradeNo;
72
73     /**
74      * 扣款交易流水号
75      * 仅用3,4请求唯一流水号传此值
76      * @see DepositDeductEventData#idempotentId
77      * @see DepositUnfreezeEventData#idempotentId
78      */
79     private String deductPayTradeNo;
80
81     /**
82      * @return
83      */
84     @Override
85     public Integer getEventType() {
86         return B2bPlatformCallbackEventTypeEnum.REFUND_FEE.getCode();
87     }
88 }

```

续租退费成功回调事件:270

```

1  /**
2   * 270:续租退费成功回调事件
3   * {
4   *   "channelId":"80001",
5   *   "orderId":"3104050285204362836",
6   *   "outOrderId":"RC202212191604667539969978368",
7   *   "eventType":260,
8   *   "eventData":{
9   *     "renewalOrderId":"3187983531024266527",
10   *     "renewalOutOrderId":"RC202212191604667539969978361",
11   *     "refundType":2,
12   *     "amount":"200.98",
13   *     "reason":"油费退款",
14   *     "payTradeNo":"202212191604667539969978368"
15   *   }
16   * }
17   */
18 @Data
19 @Accessors(chain = true)
20 public static class RenewalRefundFeeSuccessCallbackEventData implements DistributionCallbackEvent {
21     /**
22      * 续租单退费飞猪续租订单号:
23      * 1:客服退费具体续租单
24      */
25     @NotNull
26     private Long renewalOrderId;
27
28     /**
29      * 续租单退费分销续租订单号:
30      * 1:客服退费具体续租单
31      */
32     @NotNull
33     private String renewalOutOrderId;
34
35     /**
36      * 退款类型
37      * 1, 提前还车退款,
38      * 2, 油费退款,
39      * 3, 车损扣费退款,
40      * 4, 违章扣费退款,
41      * 5, 取消违约金退款(用于取消费用多扣用户不认同退费),
42      * -1, 其他
43      */
44     @NotNull

```

```

45     private Integer refundType;
46     /**
47      * 退款金额
48      * 单位:元, 200.98
49      */
50     private String amount;
51
52     /**
53      * 退款原因
54      */
55     private String reason;
56
57     /**
58      * 必传, 小二退费场景需要根据回调的交易流水号来更新退款记录表的状态
59      * 当前退款交易流水号
60      * 订单_事件类型_子类型[_批次号]
61      * e.g:202301021122903_142_1
62      * e.g:202301021122903_142_2
63      * e.g:202301021122903_142_1_1
64      * e.g:202301021122903_142_1_2
65      * payTradeNo:"202301021122903_142_1_2",
66      */
67     @NotNull
68     private String payTradeNo;
69
70     @Override
71     public Integer getEventType() {
72         return B2bDistributionCallbackEventTypeEnum.RENEWAL_REFUND_FEE_SUC
73             CESS_CALLBACK.getCode();
74     }
75 }

```

退费失败回调事件:271

```
1  /**
2   * 271:退费失败回调事件
3   * {
4   *   "channelId":"80001",
5   *   "orderId":"3104050285204362836",
6   *   "outOrderId":"RC202212191604667539969978368",
7   *   "eventType":261,
8   *   "eventData":{
9   *     "renewalOrderId":"3187983531024266527",
10    *     "renewalOutOrderId":"RC202212191604667539969978361",
11    *     "refundType":2,
12    *     "amount":"200.98",
13    *     "reason":"油费退款",
14    *     "payTradeNo":"202212191604667539969978368"
15    *   }
16  * }
17  */
18  @Data
19  @Accessors(chain = true)
20  public static class RenewalRefundFeeFailureCallbackEventData implements DistributionCallbackEvent {
21    /**
22     * 续租单退费飞猪续租订单号:
23     * 1:客服退费具体续租单
24     */
25    @NotNull
26    private Long renewalOrderId;
27
28    /**
29     * 续租单退费分销续租订单号:
30     * 1:客服退费具体续租单
31     */
32    @NotNull
33    private String renewalOutOrderId;
34
35    /**
36     * 退款类型
37     * 1, 提前还车退款,
38     * 2, 油费退款,
39     * 3, 车损扣费退款,
40     * 4, 违章扣费退款,
41     * 5, 取消违约金退款(用于取消费用多扣用户不认同退费),
42     * -1, 其他
43     */
44    private Integer refundType;
```

```

45     /**
46      * 退款金额
47      * 单位:元, 200.98
48      */
49     private String amount;
50
51     /**
52      * 退款原因
53      */
54     private String reason;
55
56     /**
57      * 必传, 小二退费场景需要根据回调的交易流水号来更新退款记录表的状态
58      * 交易流水号
59      * 订单_事件类型_子类型[_批次号]
60      * e.g:202301021122903_142_1
61      * e.g:202301021122903_142_2
62      * e.g:202301021122903_142_1_1
63      * e.g:202301021122903_142_1_2
64      * payTradeNo:"202301021122903_142_1_2",
65      */
66     @NotNull
67     private String payTradeNo;
68
69     @Override
70     public Integer getEventType() {
71         return B2bDistributionCallbackEventTypeEnum.RENEWAL_REFUND_FEE_FAIR
72             LURE_CALLBACK.getCode();
73     }
74 }

```

推送账单:

bill_detail_list

pay_channel_amount

String

false

0

支付通道费, 单位元

ext

String

false

{"key":"value"}

扩展信息

user_real_pay_amount

String

false

100

用户实际支付金额、用户有损支付金额, 单位元

biz_type

Number

true

10

业务类型, 收入交易类型: 10: 订单费用; 11: 车损押金扣款; 12: 违章押金扣款; 13: 押金扣款; 14: 有损取消 (违约金收入) ; 退款交易类型: 20: 订单费用; 21: 车损押金扣款的退款; 22: 违章押金扣款的退款; 23: 押金扣款的退款; 24: 有损取消的退款 (违约金收入的退款) ; 若无法区分是违章押金还是车损押金, 统一传13(押金扣款)或23(押金退款)

real_consume_amount

String

false

100

实际消费金额(线上费用+ 线下费用)

order_id

String

true

fz2939293

飞猪订单id

pay_time

Date

true

2023-02-03 19:23:32

打款时间, yyyy-MM-dd HH:mm:ss

discount_amount

String

false

0

飞猪平台优惠金额, 单位元

order_status

String

false

1

订单状态 (分销商侧的订单状态)

pay_trade_no

String

true

1122334411

交易流水号

distributor_discount_amount

String

false

0

分销商平台优惠金额, 单位元

order_amount

String

true

100

当前订单的金额变动, 单位元, 此值为正数, 采用 tradeType 或 bizType 来区分收入还是支出

pay_type

Number

false

1

用户支付方式, 1: 支付宝 2: 微信 3: 滴滴支付

out_order_id

String

true

fx123111

分销商订单号

trade_type

Number

true

1

交易类型, 1: 收入 2: 退款

commission_amount

String

false

10

佣金金额,单位元

公共参数

请求参数

响应参数

请求示例

响应示例

异常示例

错误码解释

API 工具

如何获得此API

续租账单, 传飞猪主单

续租账单, 传分销主单

续租账单, 传飞猪主单

续租账单, 传分销主单

