



# Void Api DOCUMENTATION

© Copyright 2023, Tripjack (Powered By Atlas Travels Online).

This document, which contains confidential material, is private and confidential and is the property and copyright of Tripjack. No part of this document may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, chemical, photocopy, recording or otherwise without the prior written permission of Tripjack.

## Revision History

Version	Publish date	Revisions
1.0	Jan 18, 2024	Document Creation

## 1. Amendment API

This method is used to add the VOIDED status to a booking. It consists of three APIs:

1. **Get Amendment Charges (Optional)** -To obtain the charges associated with a void booking. It will not apply amendments to the booking.
2. **Submit Amendment** - To make changes to a booking, you must use this API. It will return an Amendment ID, which may be used to check/poll the status of the amendment.
3. **Get Amendment Details** - To obtain the status and other records of an applied amendment.

### 1.1 Get Amendment Charges:

This is an optional API request that will return the VOID amendment fees. For any modifications, the booking must be in the SUCCESS state. It will offer amendment fees and refundable amounts for each passenger in the booking, or if the charges are not available, it will respond with "Please contact the support team to know the amendment charges".

**URL :** [{{host}}](#)/oms/v1/air/amendment/amendment-charges

**Method:** POST

**Sample Payloads:** [Link](#)

## Request Structure :

- **bookingId**
- **Type**
- **Trips** - ArrayList of trips for which amendment is to be raised (required only If a certain journey is void)
  - ◆ **src**: departure airport code
  - ◆ **dest**: arrival airport code
  - ◆ **departureDate**: Departure travel date (YYYY-MM-DD)
  - ◆ **travelers**: list of travelers from a particular trip in a booking for which amendment is to be raised.
    - **fn** - first name of traveler
    - **ln** - last name of traveler

**1.1.1 Request amendment charges for the whole booking:** Submitting just the booking id will result in prices for all pax categories for the whole booked journeys.

```
▼ object {2}
  bookingId : {{bookingId}}
  type : VOIDED
```

**1.1.2 Request amendment charges for specific travelers:** Submit traveler information along with trip information. Charges will only be provided for the specified passenger types and trips.

```
▼ object {3}
  bookingId : {{bookingId}}
  type : VOIDED
  ▼ trips [2]
    ▼ 0 {4}
      src : DEL
      dest : SIN
      departureDate : 2024-03-22
      ▼ travellers [1]
        ▼ 0 {2}
          fn : P1
          ln : P1
    ▼ 1 {4}
      src : SIN
      dest : DEL
      departureDate : 2024-02-24
      ▼ travellers [2]
        ▼ 0 {2}
          fn : P1
          ln : P1
        ▼ 1 {2}
          fn : P2
          ln : P2
  ▼ object {3}
    bookingId : {{bookingId}}
    type : VOIDED
    ▼ trips [1]
      ▼ 0 {4}
        src : SIN
        dest : DEL
        departureDate : 2024-02-24
        ▼ travellers [2]
          ▼ 0 {2}
            fn : P1
            ln : P1
          ▼ 1 {2}
            fn : P2
            ln : P2
  ▼ object {3}
    bookingId : {{bookingId}}
    type : VOIDED
    ▼ trips [1]
      ▼ 0 {4}
        src : DEL
        dest : SIN
        departureDate : 2024-03-22
        ▼ travellers [1]
          ▼ 0 {2}
            fn : P1
            ln : P1
```

**1.1.3 Request amendment charges for a particular trip:** If charges are required for all passengers in a trip, provide trip data without traveler details.

```
▼ object {3}
  bookingId : {{bookingId}}
  type : VOIDED
  ▼ trips [1]
    ▼ 0 {3}
      src : DEL
      dest : SIN
      departureDate : 2024-03-22
```

### Response Structure :

- **bookingId**
- **Trips**
  - ◆ **Src** - source airport code
  - ◆ **dest** - destination airport code
  - ◆ **departureDate** - Departure travel date(YYYY-MM-DD)
  - ◆ **flightNumbers** - flight numbers list (includes connecting flights flight numbers)
  - ◆ **airlines** - airlines code
  - ◆ **amendmentInfo** - (Map of PaxType,PaxAmendmentCharges)
    - **amendmentCharges** - Amendment Charges for Particular paxtype
    - **refundAmount** - Refundable Amount for Particular paxtype
    - **totalFare** - Total Fare for Particular paxtype

## Sample Response :

```
▼ object {3}
  bookingId : TJS11780000077091
  ▼ trips [1]
    ▼ 0 {6}
      src : DEL
      dest : SIN
      departureDate : 2024-03-22T21:50
      ▼ flightNumbers [1]
        0 : 403
      ▼ airlines [1]
        0 : SQ
      ▼ amendmentInfo {2}
        ▼ CHILD {3}
          amendmentCharges : 950
          refundAmount : 7824
          totalFare : 8774
        ▼ ADULT {3}
          amendmentCharges : 950
          refundAmount : 10307
          totalFare : 11257
      ▼ status {2}
        success : true
        httpStatus : 200
```

**Note:** total fare provided in the response will be (total fare) / (number of pax of a particular type in the journey).

## 1.2 Submit Amendment (VOID) :

This is used to apply an amendment to a booking. After submitting the request, you will receive an "amendmentId" that may be used to check/poll the status of the amendment.

URL: [{{host}}]({{host}}/oms/v1/air/amendment/submit-amendment)

Method: POST

**Sample Payloads:** [Link](#)

**Request Structure :**

- **bookingId**
- **Type**
- **Remarks** - Comment regarding the amendment.
- **Trips** - ArrayList of trips for which amendment is to be raised (required only If a certain journey is void)
  - ◆ **src**: departure airport code
  - ◆ **dest** : arrival airport code
  - ◆ **departureDate**: Departure travel date (YYYY-MM-DD)
  - ◆ **travelers**: list of travelers from a particular trip in a booking for which amendment is to be raised.
    - **fn** - first name of traveler
    - **ln** - last name of traveler

## Sample Request :

```
▼ object {4}
  bookingId : {{booking id}}
  ▼ trips [1]
    ▼ 0 {4}
      src : BOM
      dest : DEL
      departureDate : 2020-10-19
      ▼ travellers [4]
        ▼ 0 {2}
          fn : TESTA
          ln : TESTA
        ▼ 1 {2}
          fn : TESTD
          ln : TESTD
        ▼ 2 {2}
          fn : TESTB
          ln : TESTB
        ▼ 3 {2}
          fn : TESTc
          ln : TESTc
      type : VOIDED
      remarks : this is test void
```

**Note : The request structure will remain the same only adding one Key remarks.**

1.1.1 Request amendment charges for the whole booking : Submitting just the booking id will result in prices for all pax categories for the whole booked journeys.

1.1.2 Request amendment charges for specific travelers : Submit traveler information along with trip information. Charges will only be provided for the specified passenger types and trips.

1.1.3 Request amendment charges for a particular trip : If charges are required for all passengers in a trip, provide trip data without traveler details.

## **Response Structure :**

- **bookingId**
- **amendmentId** - Should be used for polling current amendment status.

## **Sample Response :**

```
▼ object {3}
  bookingId : bookingId
  amendmentId : amendmentId
  ▼ status {2}
    success :  true
    httpStatus : 200
```

## **1.3 Get Amendment Details :**

This request is used to check the status of amendment, amendment, and refund amount. An amendmentId must be returned with the submit amendment response. If the amendment status is REQUESTED, polling must be done 4-5 times with a 10-second interval to obtain the status. (If the amendment state is still REQUESTED, contact the support staff to find out more about its status).

**URL:** [`{{host}}/oms/v1/air/amendment/amendment-details`](#)

**Method:** POST

**Sample Payloads:** [Link](#)

**Request Sample :**

```
▼ object {1}
  amendmentId : 800000009202
```

#### Request Structure :

→ **amendmentId** - AmendmentId (from submit amendment response)

#### Response Structure :

```
▼ object {6}
  bookingId : TJ11310000001169
  amendmentId : 520000009115
  amendmentStatus : SUCCESS
  refundableAmount : 59504
  ▼ trips [1]
    ▼ 0 {6}
      src : DEL
      dest : SIN
      departureDate : 2024-02-17T09:55
      ▼ flightNumbers [1]
        0 : 401
      ▼ airlines [1]
        0 : SQ
      ▼ travellers [2]
        ▼ 0 {5}
          fn : t1
          ln : t1
          amendmentCharges : 350
          refundableamount : 29752
          totalFare : 30102
        ▼ 1 {5}
          fn : t2
          ln : t2
          amendmentCharges : 350
          refundableamount : 29752
          totalFare : 30102
  ▼ status {2}
    success : true
    httpStatus : 200
```

**Response Structure :** It consists of the amendment details

- **bookingId** - Booking Id
- **amendmentId** - Amendment Id
- **amendmentStatus** - Status of amendment
  - ◆ REQUESTED - Amendment is in requested state
  - ◆ REJECTED - Amendment is rejected.
  - ◆ SUCCESS - Amendment modification has been successful with all changes.
  - ◆ PENDING - Amendment has been in progress.
- **refundableAmount** - Total Refund Amount of selected trips
- **totalFare** - Total fare of Booking
- **Trips** - It is a list consisting of the tripwise amendment details.
  - ◆ **src** - source airport code
  - ◆ **dest** - destination airport code
  - ◆ **departureDate** - Departure travel date(YYYY-MM-DD)
  - ◆ **flightNumbers** - flight numbers list (includes connecting flights flight numbers)
  - ◆ **airlines** - airlines list
  - ◆ **Travelers** - list of traveler with amendment charges details
    - **fn** - traveler first name
    - **ln** - traveler last name
    - **amendmentCharges** - traveler amendment charges for particular trip
    - **refundAmount** - traveler refund amount for particular trip
    - **totalFare** - traveler total fare for particular trip

**{{host}}**

Testenv <https://apitest.tripjack.com>

Prodenv <https://tripjack.com>