

Meru Price Freeze API v4.7

Updated: 09/18/23

There are three simple steps to use the Meru Price Freeze API outlined below.

1. **Setup:** Register with Meru.
2. **Check Price Freeze:** Check the cost to freeze the price of a given ticket
3. **Freeze Ticket:** Purchase one of our listed freeze options

Setup: Post to our `/v4/flights/setup` to setup your account. If you are a new customer, please contact support@usemeru.com to get started and get your Authorization Token for our API.

Auth: Authorization Bearer Token: Contact support@usemeru.com to get authorized to use our API

Note: We provide separate authorization tokens for development and production environments. Authorization tokens used in the development environment will start with "test_" and can be used by all the API endpoints without charging your card.

Inputs: None.

Outputs:

- `HTTP statusCode: 401` - Unauthorized
- `HTTP statusCode: 400 - Bad Request`
 - Refer to the *Error Codes* section of this document for description of error code.
- `HTTP statusCode: 500 - Internal Server Error`
 - Refer to the *Error Codes* section of this document for description of error code.
- `HTTP statusCode: 200` - Request accepted
 - `err_code (int) : 0` - Success
 - `url (str)` : If `err_code == 0`, the API will output a stripe link, for you to enter your card information for stripe to store in file for future payments.

```
curl -X POST "https://api.usemeru.com/v4/flights/setup" \  
-H "Content-Type: application/json" \  

```

```
-H "Authorization: Bearer Token"
```

Check Price Freeze: POST to our `/v4/flights/check` endpoint with your flight information for us to. ***The prices listed through this API are valid for only 15 minutes!***

Auth: Authorization Bearer Token: Contact support@usemeru.com to get authorized to use our API

Note: We provide separate authorization tokens for development and production environments. Authorization tokens used in the development environment will start with "test_" and can be used by all the API endpoints without charging your card.

Inputs:

Required:

- `cabinCode (str)` : Cabin class code for trip
- `numAdults (int)` : Number of adult travelers on trip
- `numChildren (int)` : Number of child travelers on trip
- `numInfants (int)` : Number of infant travelers on trip
- `OriginDestinationInformation (dict)` : Itinerary information used to retrieve and revalidate itinerary. Follows the same format as the `OriginDestinationInformation` object in Sabre Revalidate Itinerary Request. Refer to https://developer.sabre.com/docs/rest_apis/air/search/revalidate_itinerary/reference-documentation#/default/CreateRevalidateItinerary.

Outputs:

- `HTTP statusCode: 401` - Unauthorized
- `HTTP statusCode: 400` - Bad Request
 - Refer to the *Error Codes* section of this document for description of error code.
- `HTTP statusCode: 500` - Internal Server Error
 - Refer to the *Error Codes* section of this document for description of error code.
- `HTTP statusCode: 200` - Request accepted
 - `err_code (int)` : 0 - Success
 - `freeze_pricing (dict)` : Output dict that contains price freezing offerings

- `id (str)`: ID used to refer to freeze in future when purchasing
- `expires (str)`: Date-time string with expiry time of offering in UTC
- `freeze_details (list)`: List of dicts of price freeze offering details
 - `duration (string)`: Duration price freeze is valid for
 - `pricePerAdult (float)`: Average price freeze cost per adult in USD
 - `totalPrice (float)`: Total price freeze cost for itinerary in USD

```
curl -X POST "https://api.usemeru.com/v4/flights/check" \
-H "Content-Type: application/json" \
-H "Authorization: Bearer Token" \
-d '{
  "cabinCode": "Y",
  "numAdults": 1,
  "numChildren": 1,
  "numInfants": 1,
  "OriginDestinationInformation": [
    {
      "DepartureDateTime": "2023-09-21T00:00:00",
      "DestinationLocation": {
        "LocationCode": "BLR"
      },
      "OriginLocation": {
        "LocationCode": "SFO"
      },
      "TPA_Extensions": {
        "SegmentType": {
          "Code": "0"
        }
      },
      "Flight": [
        {
          "Number": 1216,
          "DepartureDateTime": "2023-09-17T00:24:00",
          "ArrivalDateTime": "2023-09-21T08:53:00",
          "ClassOfService": "K",
          "OriginLocation": {
            "LocationCode": "SFO"
          },
          "DestinationLocation": {
            "LocationCode": "JFK"
          },
          "Airline": {
            "Operating": "B6",
            "Marketing": "B6"
          }
        }
      ],
      {
        "Number": 102,
```

```

      "DepartureDateTime": "2023-09-21T12:30:00",
      "ArrivalDateTime": "2023-09-22T12:00:00",
      "ClassOfService": "K",
      "OriginLocation": {
        "LocationCode": "JFK"
      },
      "DestinationLocation": {
        "LocationCode": "DEL"
      },
      "Airline": {
        "Operating": "AI",
        "Marketing": "AI"
      }
    },
    {
      "Number": 807,
      "DepartureDateTime": "2023-09-22T17:15:00",
      "ArrivalDateTime": "2023-09-22T20:15:00",
      "ClassOfService": "K",
      "OriginLocation": {
        "LocationCode": "DEL"
      },
      "DestinationLocation": {
        "LocationCode": "BLR"
      },
      "Airline": {
        "Operating": "AI",
        "Marketing": "AI"
      }
    }
  ]
}
}
}'

```

Freeze Ticket Price: POST to our `/v4/flights/confirm` endpoint to purchase the price freeze.

The price freeze values retrieved through the `/v4/flights/check` endpoint are valid only until ***the expiration date-time returned by the API call.***

Auth: Authorization Bearer Token: Contact support@usemeru.com to get authorized to use our API

Note: We provide separate authorization tokens for development and production environments. Authorization tokens used in the development environment will start with "test_" and can be used by all the API endpoints without charging your card.

Inputs:

Required:

- `id (str)` : ID of price to freeze
- `duration (string)` : Duration to freeze - only accepts values that the selected freeze id supports
- `buyer (dict)` : Customer information used to later search up for frozen ticket when purchasing. Contains field specified below:
 - `firstname (str)` : First name of customer
 - `lastname (str)` : Last name of customer
 - `dob (str)` : Date of birth of customer in MM-DD-YYYY format.
 - `email (str)` : Email address of customer
 - `phone (str)` : Phone number of customer

Outputs:

- `HTTP statusCode: 401` - Unauthorized
- `HTTP statusCode: 400` - Bad Request
 - Refer to the *Error Codes* section of this document for description of error code.
- `HTTP statusCode: 500` - Internal Server Error
 - Refer to the *Error Codes* section of this document for description of error code.
- `HTTP statusCode: 200` - Request accepted
 - `err_code (int)` : 0 - Success
 - `id (str)` : ID of price freeze purchase

```
curl -X POST "https://api.usemeru.com/v4/flights/confirm" \  
-H "Content-Type: application/json" \  
-H "Authorization: Bearer Token" \  
-d '{  
  "id": id,  
  "duration": "2d",  
  "buyer": {  
    "firstname": "Rohan",  
    "lastname": "Kumar",  
    "dob": "01-01-1980",  
    "email": "rohankumar@arora.com",
```

```
    "phone": "555-555-5555"  
  }  
}'
```

Error Codes:

Listed below are the possible request error codes and their meanings. These error codes are in reference to the `err_code` field returned with your request on `HTTP Success Code != 200` (request failed).

```
SUCCESS = 0  
INVALID_SCHEMA = 1  
INVALID_REQUEST = 2  
INVALID_CUSTOMER = 3  
INTERNAL_SERVER_ERROR = 4  
RETRIEVE_ITINERARY_FAILED = 5  
ITINERARY_NOT_FOUND = 6  
FREEZE_FAILED = 7  
FREEZE_ID_INVALID = 8  
FREEZE_ID_EXPIRED = 9  
CUSTOMER_NOT_READY = 10  
ALREADY_PURCHASED = 11  
FREEZE_DURATION_INVALID = 12  
PAYMENT_FAILED = 13
```