

Corendon Airlines API Documentation

API Integration Guide

Table of Content

INTRODUCTION	3
ENVIRONMENTS	3
AUTHENTICATION TOKEN	4
METHODS	6
<i>Get Flights</i>	<i>6</i>
<i>Flight Search</i>	<i>7</i>
<i>Pricing</i>	<i>14</i>
<i>Pre Book</i>	<i>19</i>
<i>Booking</i>	<i>21</i>
<i>Booked Tickets By Pnr</i>	<i>24</i>
<i>Booking Details</i>	<i>25</i>
<i>Cancel Booking</i>	<i>29</i>
<i>Cancel Ticket</i>	<i>30</i>
<i>Ticket Report</i>	<i>31</i>

Introduction

This document is designed to give a brief overview of Corendon Airlines API methods (Requests and Responses), parameters and to provide some JSON examples about the data structure.

You can contact us via e-mail for technical issues at api@corendonairlines.com

Environments

In this document {domainname} represents;

- “api.corendonairlines.com” for production environment.
- “testapi.corendonairlines.com” for test environment.

Authentication Token

Authentication in API is based on OAuth 2.0 specifications. Before calling any API methods, access token should be acquired.

URI address for authorization is in <https://{domainname}/oauth2/token> format.

Following parameters are required for authentication, which will be provided:

- ClientId
- ClientSecret
- Username
- Password

Request should be sent with POST method to token URI.

Example request:

```
POST /oauth2/token HTTP/1.1
Host: api.corendonairlines.com
Authorization: Basic eHZ6MWV2R ... o4OERSZHlPZw==
Content-Type: application/x-www-form-urlencoded; charset=UTF-8

grant_type=password&username=XXUSERNAMEXX&password=XXPASSWORDXX
```

- Authorization: Basic XXKEYXX header should be added to request header, where XXKEYXX is “ClientId:ClientSecret” (replace with provided values) string in Base64 format
- In body of request replace XXUSERNAMEXX and XXPASSWORDXX with provided values respectively.

Example response:

```
HTTP/1.1 200 OK
Status: 200 OK
Content-Type: application/json; charset=utf-8

{"access_token":"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9..Wjum-Dh0","token_type":"bearer","expires_in":3599}
```

- “access_token” field is token required for further access to API methods
- “expires_in” field shows duration in seconds, how long this token is valid for.

For further access to API requests require http header Bearer with access token as value which acquired from authentication.

Example Request:

```
POST /api/flight/search HTTP/1.1
Host: api.corendonairlines.com
Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9...Wjum-Dh0
Content-Type: application/json; charset=utf-8
```

Methods

Get Flights

- Description: This method used to get all available flights to update your local database. You should use this method once every 24 hours.
- URL: `https://{domainname}/api/flight/list`
- Method: GET
- Response JSON Formula:

```
{
  "Flights": [
    {
      "DepartureAirportCode": "AYT",
      "DepartureAirportName": "Antalya Airport",
      "DepartureCityName": "Antalya",
      "ArrivalAirportCode": "AMS",
      "ArrivalAirportName": "Amsterdam Airport Schiphol",
      "ArrivalCityName": "Amsterdam",
      "FlightNumber": "020",
      "DepartureDate": "2016-04-09T10:00:00",
      "ArriveDate": "2016-04-09T13:30:00",
      "AirlineCode": "CAI",
      "AirlineName": "Corendon Airlines",
    },
    .
    .
    .
  ]
}
```

Flight Search

- Description: The FlightSearch method gives you an overview about all the flights that are available through Corendon Airlines according to a given parameters.
- URL: <https://{domainname}/api/flight/search>
- Method: POST
- Request JSON Formula:

```
{
  "TripType": "2",
  "PassengerCounts": {"AdultCount" : "2", "ChildCount" : "1", "InfantCount" : "1"},
  "CurrencyCode": "EUR",
  "Flights":
  [
    {
      "FlightDate": "2016-06-09",
      "DeparturePointType": "0",
      "DeparturePointCode": "AMS",
      "ArrivalPointType": "0",
      "ArrivalPointCode": "AYT"
    },
    {
      "FlightDate": "2016-06-10",
      "DeparturePointType": "0",
      "DeparturePointCode": "AYT",
      "ArrivalPointType": "0",
      "ArrivalPointCode": "AMS"
    }
  ]
}
```

- In Parameters:

Parameter Name	Mandatory or Optional	Explanation	Example
TripType	Mandatory	One Way or Round Trip	1
AdultCount	Mandatory	Adults	2
ChildCount	Mandatory	Children	1
InfantCount	Mandatory	Infants	1
CurrencyCode	Mandatory	Currency Code	EUR
FlightDate	Mandatory	Departure Date	2016-06-09
DeparturePointType	Mandatory	Type of Departure Place	0
DeparturePointCode	Mandatory	Departure Airport	AMS
ArrivalPointType	Mandatory	Type of Destination Place	0
ArrivalPointCode	Mandatory	Destination Airport	AYT

* **TripType** values:

- OneWay = 1
- RoundTrip = 2

* **PointType** values:

- Airport = 0
- City = 1

* When you choose RoundTrip you have to provide at least two Flights.

- Response JSON Formula:

```
{
  "Currency": "EUR",
  "PassengerCounts": {
    "AdultCount": 2,
    "ChildCount": 1,
    "InfantCount": 1
  },
  "TotalServiceCharge": 0.0,
  "Flights": [
    {
      "Key": {
        "FlightDate": "2016-06-09",
        "DeparturePointType": 0,
        "DeparturePointCode": "AMS",
        "ArrivalPointType": 0,
        "ArrivalPointCode": "AYT"
      },
      "Value": [
        {
          "Key": {
            "FlightRouteID": 7250,
            "DepartureDate": "2016-06-09T16:30",
            "ArriveDate": "2016-06-09T21:30",
            "DepartureAirportCode": "AMS",
            "DepartureAirportName": "Amsterdam Airport Schiphol",
            "DepartureCityName": "Amsterdam",
            "ArrivalAirportCode": "AYT",
            "ArrivalAirportName": "Antalya Airport",
            "ArrivalCityName": "Antalya",
            "FlightNumber": "021",
            "AirlineCode": "CAI",
            "Via": "Direct",
            "FreePax": 9
          },
          "Value": [
            {
              "FlightKey": "eyJGQSI6MTYzNTIsIkFGIjoyLCJGRiI6MjE5trDFSLCJGUCl6OSwiRiI6NTcyODR9",
              "RestPax": 5,
              "PriceBandCode": "W",
              "AirFareCode": "ECO",
              "AirFareName": "Economy",
              "CabinClass": 0,
              "Baggage": 20,
              "HandLuggage": 8,
              "Fare": [
                {

```

```

        "TravellerType": 1,
        "Price": 129.00
    },
    {
        "TravellerType": 2,
        "Price": 129.00
    },
    {
        "TravellerType": 3,
        "Price": 20.0
    }
],
"SubTotal": 407.00
},
{
    "FlightKey": "eyJGQSI6MTYzNTIsIkFGIjozLCJGdsRiI6MjE5LCJGUUCI6fsda564I6NTcyODR9",
    "RestPax": 9,
    "PriceBandCode": "D",
    "AirFareCode": "PREM",
    "AirFareName": "Premium",
    "CabinClass": 0,
    "Baggage": 30,
    "HandLuggage": 8,
    "Fare": [
        {
            "TravellerType": 1,
            "Price": 169.00
        },
        {
            "TravellerType": 2,
            "Price": 169.00
        },
        {
            "TravellerType": 3,
            "Price": 20.0
        }
    ],
    "SubTotal": 527.00
}
]
}
]
},
{
    "Key": {
        "FlightDate": "2016-06-10",
        "DeparturePointType": 0,
        "DeparturePointCode": "AYT",
        "ArrivalPointType": 0,
        "ArrivalPointCode": "AMS"
    }
}

```

```

},
"Value": [
  {
    "Key": {
      "FlightRouteID": 7251,
      "DepartureDate": "2016-06-10T22:25",
      "ArriveDate": "2016-06-11T01:45",
      "DepartureAirportCode": "AMS",
      "DepartureAirportName": "Amsterdam Airport Schiphol",
      "DepartureCityName": "Amsterdam",
      "ArrivalAirportCode": "AYT",
      "ArrivalAirportName": "Antalya Airport",
      "ArrivalCityName": "Antalya",
      "FlightNumber": "021",
      "AirlineCode": "CAI",
      "Via": "Direct",
      "FreePax": 9
    },
    "Value": [
      {
        "FlightKey": "reFASDDI6MTYzNTIsIkFGIjldsdFfDREQSMjE5LCJGUCl6OSwiRiI6NTfdREA",
        "RestPax": 5,
        "PriceBandCode": "W",
        "AirFareCode": "ECO",
        "AirFareName": "Economy",
        "CabinClass": 0,
        "Baggage": 20,
        "HandLuggage": 8,
        "Fare": [
          {
            "TravellerType": 1,
            "Price": 129.00
          },
          {
            "TravellerType": 2,
            "Price": 129.00
          },
          {
            "TravellerType": 3,
            "Price": 20.0
          }
        ],
        "SubTotal": 407.00
      },
      {
        "FlightKey": "eyJGQSI6MTYzNTIsIkFGIjoZLCJGRiI66MjE5LCJGUCl6NywiRiI6NTcyODR9",
        "RestPax": 9,
        "PriceBandCode": "D",
        "AirFareCode": "PREM",
        "AirFareName": "Premium",

```

```

    "CabinClass": 0,
    "Baggage": 30,
    "HandLuggage": 8,
    "Fare": [
      {
        "TravellerType": 1,
        "Price": 169.00
      },
      {
        "TravellerType": 2,
        "Price": 169.00
      },
      {
        "TravellerType": 3,
        "Price": 20.0
      }
    ],
    "SubTotal": 527.00
  }
]
}

```

* **TravellerType** values:

- Adult = 1, Child = 2, Infant = 3

* **CabinClass** values:

- Economy = 0, PremiumEconomy = 1, Business = 2, First = 3

- Response Structure

- Currency
- PassengersCounts
- TotalServiceCharge
- Flights **(array)**
 - o Key
 - FlightDate
 - DeparturePointType
 - DeparturePointCode
 - ArrivalPointType
 - ArrivalPointCode
 - o Value **(array)**
 - Key
 - FlightRouteID
 - DepartureDate
 - ArriveDate
 - DepartureAirportCode
 - DepartureAirportName
 - DepartureCityName
 - ArrivalAirportCode
 - ArrivalAirportName
 - ArrivalCityName
 - FlightNumber
 - AirlineCode
 - Via
 - FreePax
 - Value **(array)**
 - FlightKey
 - RestPax
 - PriceBandCode
 - AirFareCode
 - AirFareName
 - CabinClass
 - Baggage
 - HandLuggage
 - Fare **(array)**
 - o TravellerType
 - o Price (per person)
 - SubTotal

Pricing

- Description: The Pricing method gives a complete prices by key for selected flights.
- URL: <https://{domainname}/api/flight/price>
- Method: POST
- Request JSON Formula:

```
{
  "TripType": "2",
  "CurrencyCode": "EUR",
  "PassengerCounts": {
    "AdultCount": "2",
    "ChildCount": "1",
    "InfantCount": "1"
  },
  "FlightKeys": [
    "eyJGQSI6MTYzNTIsIkFGIjoyLCJGRiI6MjE5rtrDFSJCJGUCI6OSwiRiI6NTcyODR9",
    "reFASDDI6MTYzNTIsIkFGIj1dsdfFdREQSMjE5LCJGUCI6OSwiRiI6NTfdREA"
  ]
}
```

- In Parameters:

Parameter Name	Mandatory or Optional	Explanation	Example
TripType	Mandatory	One Way or Round Trip	1
AdultCount	Mandatory	Adults	2
ChildCount	Mandatory	Childs	1
InfantCount	Mandatory	Infants	1
CurrencyCode	Mandatory	Currency Code	EUR
FlightKeys	Mandatory	Flight Keys From Search	

- * **TripType** values:

- o OneWay = 1, RoundTrip = 2

- Response JSON Formula:

```
{
  "Currency": "EUR",
  "TotalAmount": "814.00",
  "PassengerCounts":
  {
    "AdultCount": 2,
    "ChildCount": 1,
    "InfantCount": 1
  },
  "TotalServiceCharge": 0.0,
  "Flights": [
    {
      "Key": {
        "FlightDate": "2016-06-09",
        "DeparturePointType": 0,
        "DeparturePointCode": "AMS",
        "ArrivalPointType": 0,
        "ArrivalPointCode": "AYT"
      },
      "Value": [
        {
          "Key": {
            "DepartureDate": "2016-06-09T16:30",
            "ArriveDate": "2016-06-09T21:30",
            "DepartureAirportCode": "AMS",
            "DepartureAirportName": "Amsterdam Airport Schiphol",
            "DepartureCityName": "Amsterdam",
            "ArrivalAirportCode": "AYT",
            "ArrivalAirportName": "Antalya Airport",
            "ArrivalCityName": "Antalya",
            "FlightNumber": "021",
            "AirlineCode": "CAI"
          },
          "Value": [
            {
              "FlightKey": "eyJGQSI6MTYzNTIsIkFGIjoyLCJGRiI6MjE5rtrDFSLCJGUCl6OSwiRiI6NTcyODR9",
              "PriceBandCode": "W",
              "AirFareCode": "ECO",
              "AirFareName": "Economy",
              "CabinClass": 0,
              "Baggage": 20,
              "HandLuggage": 8,
              "Fare": [
                {

```

```

        "TravellerType": 1,
        "Price": 258.00
    },
    {
        "TravellerType": 2,
        "Price": 129.00
    },
    {
        "TravellerType": 3,
        "Price": 20.0
    }
],
"SubTotal": 407.00
}
]
}
],
{
    "Key": {
        "FlightDate": "2016-06-10",
        "DeparturePointType": 0,
        "DeparturePointCode": "AYT",
        "ArrivalPointType": 0,
        "ArrivalPointCode": "AMS"
    },
    "Value": [
        {
            "Key": {
                "DepartureDate": "2016-06-10T22:25",
                "ArriveDate": "2016-06-11T01:45",
                "DepartureAirportCode": "AMS",
                "DepartureAirportName": "Amsterdam Airport Schiphol",
                "DepartureCityName": "Amsterdam",
                "ArrivalAirportCode": "AYT",
                "ArrivalAirportName": "Antalya Airport",
                "ArrivalCityName": "Antalya",
                "FlightNumber": "021",
                "AirlineCode": "CAI",
            },
            "Value": [
                {
                    "FlightKey": "reFASDDI6MTYzNTIsIkFGIjldsdFfFdREQSMjE5LCJGUCl6OSwiRiI6NTfdREA",
                    "PriceBandCode": "W",
                    "AirFareCode": "ECO",
                    "AirFareName": "Economy",
                    "CabinClass": 0,
                    "Baggage": 20,
                    "HandLuggage": 8,
                    "Fare": [

```



```

    {
      "TravellerType": 1,
      "Price": 258.00
    },
    {
      "TravellerType": 2,
      "Price": 129.00
    },
    {
      "TravellerType": 3,
      "Price": 20.0
    }
  ],
  "SubTotal": 407.00
}
]
}
]
}
}

```

* **TravellerType** values:

- Adult = 1, Child = 2, Infant = 3

* **CabinClass** values:

- Economy = 0, PremiumEconomy = 1, Business = 2, First = 3

- Response Structure

- TotalAmount
- Currency
- PassengersCounts
- TotalServiceCharge
- Flights **(array)**
 - o Key
 - FlightDate
 - DeparturePointType
 - DeparturePointCode
 - ArrivalPointType
 - ArrivalPointCode
 - o Value **(array)**
 - Key
 - DepartureDate
 - ArriveDate
 - DepartureAirportCode
 - DepartureAirportName
 - DepartureCityName
 - ArrivalAirportCode
 - ArrivalAirportName
 - ArrivalCityName
 - FlightNumber
 - AirlineCode
 - Value **(array)**
 - FlightKey
 - PriceBandCode
 - AirFareCode
 - AirFareName
 - CabinClass
 - Baggage
 - HandLuggage
 - Fare **(array)**
 - o TravellerType
 - o Price (Sum of Traveller Types)
 - SubTotal

Pre Book

- Description: This method creates a basket which is providing availability to book. This is always mandatory before booking.
- URL: <https://{domainname}/api/flight/basket/create>
- Method: POST
- Request JSON Formula:

```
{
  "Passengers": [
    {
      "TravellerType": "1",
      "Title": "MR",
      "FirstName": "John",
      "LastName": "Doe",
      "Birthdate": "1984-06-04"
    },
    {
      "TravellerType": "1",
      "Title": "MRS",
      "FirstName": "Emma",
      "LastName": "Doe",
      "Birthdate": "1986-07-09"
    },
    {
      "TravellerType": "2",
      "Title": "Mstr",
      "FirstName": "Michel",
      "LastName": "Doe",
      "Birthdate": "2013-03-05"
    },
    {
      "TravellerType": "3",
      "Title": "INF",
      "FirstName": "Sara",
      "LastName": "Doe",
      "Birthdate": "12.04.2015"
    }
  ],
  "TripType": "2",
  "PassengerCounts": {
    "AdultCount": "2",
    "ChildCount": "1",
    "InfantCount": "1"
  }
},
```

```

    "CurrencyCode": "EUR",
    "FlightKeys": [
      "eyJGQSI6MTYzNTIsIkFGIjoyLCJGRI6MjE5LCJGU6OSwiRiI6NTcyODR9",
      "eyJGQSI6MTYzNTYzNTIsIkFGIjoyLCJGRI6MjE5LCJGU6OSwiRiI6NTcyOTd9"
    ]
  }

```

- In Parameters:

Parameter Name	Mandatory or Optional	Explanation	Example
TravellerType	Mandatory	Traveller Type	1
Title	Mandatory	Specifying Gender	MR
FirstName	Mandatory	First Name	John
LastName	Mandatory	Last Name	Doe
BirthDate	Mandatory	Birth Date	1984-06-04
TripType	Mandatory	One Way or Round Trip	1
AdultCount	Mandatory	Adults	2
ChildCount	Mandatory	Childs	1
InfantCount	Mandatory	Infants	1
CurrencyCode	Mandatory	Currency Code	EUR
FlightKeys	Mandatory	Flight Keys From Search	

* **TripType** values:

- OneWay = 1, RoundTrip = 2

* **TravellerType** values:

- Adult = 1, Child = 2, Infant = 3

* **Title** values:

- Adult Male = “MR”, Adult Female = “MRS”, Child Male = “Mstr”, Child Female = “Miss”, Infant = “INF”

- Response JSON Formula:

```

{
  "BasketKey": "6JNKR4Q93"
}

```

Insert Service Charge / Extra Price

- Description: This method is optional before booking. If you wish / should insert your service charge which is taken extra money from customer.
- URL: <http://{domainname}/api/basket/comission/updateflight>
- Method: POST
- Request JSON Formula:

```
{  
  "BasketKey": "6JNKRF4Q93",  
  "BenefitAmount": "20.00"  
}
```

- Response JSON Formula:

```
{  
  "IsOk": true,  
  "Message": ""  
}
```

Booking

- Description: This method creates a booking on the system. You should provide valid credit card and contact information with basket key from pre-book and total amount which is latest one you have. The Payment Method will be provided by Corendon Airlines for test and production environment. You don't have to send credit card fields if Payment Method Code is "C" (Credit) or "PP" (PrePayment). Otherwise it should be filled.
- URL: <https://{domainname}/api/booking/create>
- Method: POST
- Request JSON Formula:

```
{
  "BasketKey": "6JNKRF4Q93",
  "BookingLanguageCode": "EN",
  "SalesAmount": "1014.00",
  "Payment": {
    "CreditCard": {
      "CardType": 1,
      "CardNumber": "1234-1234-1234-1234",
      "ExpireMonth": 12,
      "ExpireYear": 17,
      "Cvc": "123",
      "FullName": "John Edward"
    },
    "CurrencyCode": "EUR",
    "PaymentMethodCode": "C",
  },
  "ContactInformation": {
    "Title": "MR",
    "FirstName": "John",
    "LastName": "Edward",
    "Phone": "+3465597449254",
    "Email": "John.Edward@infoinfo.com",
    "BirthDate": "1990-01-01"
  }
}
```

- In Parameters:

Parameter Name	Mandatory or Optional	Explanation	Example
BasketKey	Mandatory	Basket Key from PreBook	6JNKRF4Q93
BookingLanguageCode	Optional (Default: EN)	Booking Language	TR
SalesAmount	Mandatory	Total Amount from Last Pricing	1014.00
FirstName	Mandatory	Customer First Name	John
LastName	Mandatory	Customer Last Name	Edward
Email	Mandatory	Customer Email	john@edward.com
Phone	Mandatory	Customer Phone	+905431234567
CurrencyCode	Mandatory	Currency Code	EUR
PaymentMethodCode	Mandatory	Payment Method of Booking	C
CardType	Depends on the Payment Method	Visa or MasterCard	1
CardNumber	Depends on the Payment Method	Credit Card Number	1234-1234-5678-4567
ExpireMonth	Depends on the Payment Method	Expire Month of Credit Card	12
ExpireYear	Depends on the Payment Method	Last 2 Digit of Expire Year of Credit Card	17
Cvc	Depends on the Payment Method	3 Digit Key of Credit Card	123
FullName	Depends on the Payment Method	Full Name of Credit Card	John Edward

* **CardType** values:

- Visa = 1, MasterCard= 2

- Response JSON Formula:

```
{
  "Pnr": "OJDSNV",
  "HasProblem": false,
  "Message": ""
}
```

Booked Tickets By Pnr

- Description: This method used to check travelers with ticket number in the booking which is provided by given Pnr.
- URL: <https://{domainname}/api/flight/booking/ticket/{pnr}>
- Method: GET
- Response JSON Formula:

```
{
  "Passengers": [
    {
      "Id": "45666",
      "TicketNumber": "395 00000012312",
      "FirstName": "John",
      "LastName": "Doe",
      "BirthDate": "1980-01-01",
      "Title": "MR",
      "TravellerType": 1
    },
    {
      "Id": "45667",
      "TicketNumber": "395 00000012313",
      "FirstName": "Emma",
      "LastName": "Doe",
      "BirthDate": "1980-01-01",
      "Title": "MRS",
      "TravellerType": 1
    }
  ]
}
```

- Response Structure

```
- Passengers (array)
  o Id
  o TicketNumber
  o FirstName
  o LastName
  o BirthDate
  o Title
  o TravellerType
```


Booking Details

- Description: This method used to get booking detail which is provided by given Pnr.
- URL: <https://{domainname}/api/flight/booking/{pnr}>
- Method: GET
- Response JSON Formula:

```
{
  "Pnr": "HM518H",
  "BookingTravellers": [
    {
      "Id": 20612,
      "FirstName": "John",
      "LastName": "John",
      "BirthDate": "1986-04-06T00:00:00",
      "Title": "MR",
      "TravellerType": 1
    },
    {
      "Id": 20613,
      "FirstName": "Michel",
      "LastName": "Michel",
      "BirthDate": "2013-04-06T00:00:00",
      "Title": "CHD",
      "TravellerType": 2
    }
  ],
  "BookingProducts": [
    {
      "BookingProductId": 264,
      "BookingProductDetails": [
        {
          "FareClass": {
            "Code": "FLEX",
            "Name": "Flexi"
          },
          "Departure": {
            "Code": "AMS",
            "Name": "Amsterdam Airport Schiphol"
          },
          "Arrival": {
            "Code": "AYT",
            "Name": "Antalya Airport"
          }
        }
      ]
    }
  ]
}
```

```

    "Airline": {
      "Code": "CAI",
      "Name": "Corendon Airlines"
    },
    "FlightNumber": "021",
    "ProductTravellers": [
      {
        "TicketNumber": "100 0000000115",
        "Id": 20612,
        "FirstName": "John",
        "LastName": "John",
        "BirthDate": "1986-04-06T00:00:00",
        "Title": "MR",
        "TravellerType": 1
      },
      {
        "TicketNumber": "100 0000000116",
        "Id": 20613,
        "FirstName": "Michel",
        "LastName": "Michel",
        "BirthDate": "2013-04-06T00:00:00",
        "Title": "CHD",
        "TravellerType": 2
      }
    ],
    "AdultCount": 1,
    "ChildCount": 1,
    "InfantCount": 0,
    "BeginDate": "2016-04-07T14:30:00",
    "EndDate": "2016-04-07T19:30:00",
    "Prices": [
      {
        "BookingProductRelationId": 1027,
        "BookingProductId": 264,
        "Amount": 129,
        "Explain": "MR John John (06.04.1986) (07.04.2016 021 (E Class))",
        "PriceCode": 201,
        "PriceLang": "FLIGHT_PRICE_ADULT"
      },
      {
        "BookingProductRelationId": 1028,
        "BookingProductId": 264,
        "Amount": 129,
        "Explain": "CHD Michel Michel (06.04.2013) (07.04.2016 021 (E Class))",
        "PriceCode": 202,
        "PriceLang": "FLIGHT_PRICE_CHILD"
      }
    ],
    "SubTotal": 258
  }

```

```

    ]
  }
],
"Currency": {
  "Id": 8,
  "Code": "EUR",
  "Name": "Euro",
  "Symbol": "€",
  "ShowText": "€ EUR"
},
"TravelDate": "2016-04-07T14:30:00",
"BookingDate": "2016-04-06T08:49:39.177",
"ModificationDate": "2016-04-06T08:49:41.17",
"TotalSalesAmount": 258
}

```

- Response Structure

- Pnr
- BookingTravellers (**array**)
 - o Id
 - o FirstName
 - o LastName
 - o BirthDate
 - o Title
 - o TravellerType
- BookingProducts (**array**)
 - o BookingProductId
 - o BookingProductDetails (**array**)
 - FareClass
 - Code
 - Name
 - Departure
 - Code
 - Name
 - Arrival
 - Code
 - Name

- Airline
 - Code
 - Name
 - FlightNumber
 - ProductTravellers (**array**)
 - Ticket Number
 - Id
 - FirstName
 - LastName
 - BirthDate
 - Title
 - TravellerType
 - AdultCount
 - ChildCount
 - InfantCount
 - BeginDate (Departure Date)
 - EndDate (Arrive Date)
 - Prices (**array**)
 - BookingProductRelationId
 - BookingProductId
 - Amount
 - Explain
 - PriceCode
 - PriceLang
- Currency
 - o Id
 - o Code
 - o Name
 - o Symbol
 - o ShowText
- TravelDate
- BookingDate
- ModificationDate
- TotalSalesAmount

Cancel Booking

- Description: This method cancels a booking on the system. You should provide Pnr to cancel booking.
- URL: <https://{domainname}/api/booking/modify>
- Method: POST
- Request JSON Formula:

```
{  
  "Pnr": "OJDSNV"  
}
```

- In Parameters:

Parameter Name	Mandatory or Optional	Explanation	Example
Pnr	Mandatory	Pnr of Booking	OJDSNV

- Response JSON Formula:

```
{  
  "Pnr": "OJDSNV",  
  "HasProblem": false,  
  "Message": ""  
}
```

Cancel Ticket

- Description: This method cancels tickets on the booking. You should provide Pnr and Ticket Numbers to cancel tickets.
- URL: <https://{domainname}/api/flight/booking/cancel/ticket>
- Method: POST
- Request JSON Formula:

```
{
  "Pnr": "OJDSNV",
  "TicketNumbers": {
    "395 00000012312",
    "395 00000012313"
  }
}
```

- In Parameters:

Parameter Name	Mandatory or Optional	Explanation	Example
Pnr	Mandatory	Pnr of Booking	OJDSNV
TicketNumbers	Mandatory	Ticket Numbers to Cancel	395 00000012312

- Response JSON Formula:

```
{
  "Pnr": "OJDSNV",
  "HasProblem": false,
  "Message": ""
}
```

Ticket Report

- Description: This method lists tickets booked between given dates. You should provide BookingDateBegin.
- URL: <https://{domainname}/api/flight/booking/ticket/report>
- Method: POST
- Request JSON Formula:

```
{
  "BookingDateBegin": "2016-09-01"
}
```

- In Parameters:

Parameter Name	Mandatory or Optional	Explanation	Example
BookingDateBegin	Mandatory	Begin of the Booking Date Range	2016-06-01
BookingDateEnd	Optional (Default: BookingDateBegin+1)	End of the Booking Date Range	2016-06-02

- Response Structure

- **(array)**
 - o PNR
 - o BookingDate
 - o BookingStatus
 - o TicketNumber
 - o Passenger
 - o TravellerType
 - o TripType
 - o OutbountFlightInfo
 - o InbountFlightInfo
 - o SaleCurrencyCode
 - o OutbountFlightPrice
 - o InbountFlightPrice
 - o ServiceChargePrice
 - o TicketPrice

- Response JSON Formula:

```
[
  {
    "PNR": "ENTNV8",
    "BookingDate": "01.09.2016",
    "BookingStatus": "Confirmed",
    "TicketNumber": "395 0000000262",
    "Passenger": "MR\\Doe.John",
    "TravellerType": 1,
    "TripType": "RoundTrip",
    "OutBountFlightInfo": "31.10.2016 | AYT-AMS | 020 | 10:00-13:30",
    "InBountFlightInfo": "31.11.2016 | AMS-AYT | 021 | 10:00-13:30",
    "SaleCurrencyCode": "EUR",
    "OutBountFlightPrice": 219,
    "InBountFlightPrice": 99,
    "ServiceChargePrice": 0,
    "TicketPrice": 318
  },
  {
    "PNR": "VB6LCJ",
    "BookingDate": "01.09.2016",
    "BookingStatus": "Confirmed",
    "TicketNumber": "395 0000000263",
    "Passenger": "MRS\\Doe.Emma",
    "TravellerType": 1,
    "TripType": "OneWay",
    "OutBountFlightInfo": "03.11.2016 | AMS-AYT | 021 | 14:30-19:30",
    "InBountFlightInfo": "",
    "SaleCurrencyCode": "EUR",
    "OutBountFlightPrice": 99,
    "InBountFlightPrice": 0,
    "ServiceChargePrice": 0,
    "TicketPrice": 99
  }
]
```