

Technical Test JAVA

RMSFlights - Problem to be solved

Background:

RMSFlights is a flight search solution aggregating flight results initially from 2 different airlines (Tunisair and Nouvelair). A future iteration (not part of the test) may add more airlines.

What is required:

Use this GitHub repository as a base to implement the RMS Flights service that should produce an aggregated result from both Tunisair and Nouvelair. The result should be a JSON response **that contains a list of flights ordered by fare** which has the following attributes:

RMS Flights API

Request

Name	Description
origin	3 letter IATA code(eg. TUN, FRA)
destination	3 letter IATA code(eg. GER, ESP)
departureDate	ISO_LOCAL_DATE format
returnDate	ISO_LOCAL_DATE format
numberOfPassengers	Maximum 4 passengers

Response

Name	Description
airline	Name of Airline
fare	Total price rounded to 2 decimals
departureAirportCode	3 letter IATA code(eg. TUN, FRA)
destinationAirportCode	3 letter IATA code(eg. TUN, FRA)
departureDate	ISO_DATE_TIME format
arrivalDate	ISO_DATE_TIME format

The service should connect to the both the suppliers using HTTP.

Tunisair API

Request

Name	Description
origin	3 letter IATA code(eg. TUN, FRA)
destination	3 letter IATA code(eg. TUN, FRA)
departureDate	ISO_LOCAL_DATE format
returnDate	ISO_LOCAL_DATE format
passengerCount	Number of passengers

Response

Name	Description
airline	Name of the airline
price	Total price
cabinclass	E for Economy and B for Business
departureAirportCode	Eg: TUN
destinationAirportCode	Eg: FRA
departureDate	ISO_LOCAL_DATE_TIME format
arrivalDate	ISO_LOCAL_DATE_TIME format

Nouvelair API

Request

Name	Description
from	3 letter IATA code(eg. TUN, FRA)
to	3 letter IATA code(eg. GER, ESP)
outboundDate	ISO_LOCAL_DATE format
inboundDate	ISO_LOCAL_DATE format
numberOfAdults	Number of passengers

Response

Name	Description
carrier	Name of the Airline
basePrice	Price without tax(doesn't include discount)
tax	Tax which needs to be charged along with the price
discount	Discount which needs to be applied on the price(in percentage)
departureAirportName	3 letter IATA code(eg. LHR, AMS)
arrivalAirportName	3 letter IATA code(eg. LHR, AMS)
outboundDateTime	ISO_INSTANT format
inboundDateTime	ISO_INSTANT format

What you need to provide:

- A solution that meets the above requirements.
- The implementation should be made as close to 'production ready' as possible within the time constraints.
- use Validator, transformer, ExceptionHandler, CompletableFutur (supplyAsync)...etc
- use a **unit test** for the different methods
- swagger for API documentation
- Dockerizing this application using docker file, if possible run each API in different containers using docker-compose

It is fine to change any of the supplied application code if you choose to do so please add comments to indicate what has changed and why.

Formulas

$\text{fare} = (\text{basePrice} * \text{discount} / 100) + \text{tax}$

Note

Please clone this project and then create your own repository from it.

Examples

First API (Tunisair) :

GET

<http://localhost:8080/tunisair/flights?origin=FRA&destination=TUN&departureDate=2022-07-08&returnDate=2022-07-09>

```
[ {
  "airline": "tunisair",
  "price": 100.21,
  "cabinclass": "E",
  "departureAirportCode": "FRA",
  "destinationAirportCode": "TUN",
  "departureDate": "2022-07-08T11:44:44.797",
  "arrivalDate": "2022-07-09T11:44:44.797"
},
{
  "airline": "tunisair",
  "price": 90.21,
  "cabinclass": "B",
  "departureAirportCode": "FRA",
  "destinationAirportCode": "TUN",
  "departureDate": "2022-07-08T11:44:44.797",
  "arrivalDate": "2022-07-09T11:44:44.797"
}]
```

Second API (Nouvelair) :

GET

http://localhost:8080/nouvelair/flights?from=FR&to=TUN&inboundDate=2022-07-08&outboundDate=2022-07-09

```
[ {
  "carrier": "nouvelair",
  "basePrice": 200.21,
  "tax": 100.32,
  "discount": 20.0,
  "departureAirportName": "TUN",
  "arrivalAirportName": "FRA",
  "outboundDateTime": "2022-07-08T09:28:27.141Z",
  "inboundDateTime": "2022-07-09T09:28:27.141Z"
},
{
  "carrier": "nouvelair",
  "basePrice": 250.21,
  "tax": 110.32,
  "discount": 15.0,
  "departureAirportName": "TUN",
  "arrivalAirportName": "FRA",
  "outboundDateTime": "2022-07-08T09:28:27.141Z",
  "inboundDateTime": "2022-07-09T09:28:27.141Z"
}]
```

Third API (RMSflights)

GET

http://localhost:8080/rmsflights/flights?origin=TUN&destination=FR&departureDate=2022-07-08&returnDate=2022-07-09

```
[{
  "airline": "tunisair",
  "fare": 90.21,
  "departureAirportCode": "TUN",
  "destinationAirportCode": "FRA",
  "departureDate": "2022-07-08T11:44:44.797",
```

```
"arrivalDate": "2022-07-09T11:44:44.797"
},
{
  "airline": "tunisair",
  "fare": 100.21,
  "departureAirportCode": "TUN",
  "destinationAirportCode": "FRA",
  "departureDate": "2022-07-08T11:44:44.797",
  "arrivalDate": "2022-07-09T11:44:44.797"
},
{
  "airline": "nouvelair",
  "fare": 260.488,
  "departureAirportCode": "TUN",
  "destinationAirportCode": "FRA",
  "departureDate": "2022-07-08T09:28:27.141Z",
  "arrivalDate": "2022-07-09T09:28:27.141Z"
},
{
  "airline": "nouvelair",
  "fare": 310.5,
  "departureAirportCode": "TUN",
  "destinationAirportCode": "FRA",
  "departureDate": "2022-07-08T09:28:27.141Z",
  "arrivalDate": "2022-07-09T09:28:27.141Z"
}]
```