Amadeus Quick-Connect

Integration Guide

Version 6.6

29.03.2022

Document control				
Security level	Restricted			
Company	Amadeus IT Group SA			
Department	API Consultancy Services			
Author	Karla Paris / Andreas Bonk			
Reviewed by			**Date**	
Approved by			**Date**	
Version	**Date**	**Change**	**Comment**	**By**
1.0	24/10/2019	Document launch		Karla PARIS
1.1	18/11/2019	Update	Flow Diagrams	Karla PARIS
1.2	03/12/2019	Update	B2B Wallet	Andreas BONK
1.3	03/12/2019	Update	Glossary	Karla PARIS
1.4	12/12/2019	Update	Seat map	Rabia Burcu Ekinci
1.5	17/12/2019	Update	Several items	Karla Paris/ Andreas Bonk
2.0	22/05/2020	Update	New Functionality	Karla Paris
3.0	18/01/2021	Update	New Functionality	Karla Paris
3.0	15/03/2021	Update	PATCH example	Andreas Bonk
4.0	22/03/2021	Update	Branded Fares Upsell	Sylvia L. Quiñones
5.0	19/05/2021	Update	Pricing form cache information, Branded Fares and Availability.	Andreas Bonk/ Karla Paris
6.0	30/07/2021	Update	B2B Wallet	Andreas Bonk
6.1	02/09/2021	Update	Enhancements	Karla Paris
6.2	21/10/2021	Update	Commissions, errors & warnings	Karla Paris/ Andreas Bonk
6.3	20/12/2021	Update	Search Parameters/objects to consider, Technical Parameter and additional information payment	Karla Paris/ Andreas Bonk
6.4		Update	errors & warnings	Andreas Bonk
6.4	25/01/2022	Update	Baggage Flow	Rabia Burcu Ekinci
6.5	28/02/2022	Update	Title of travelers	Andreas Bonk
6.6	29/03/2022	Update	TRX/s, B2BWallet Additional information, FOP additional information’s	Andreas Bonk

			as a note, Note to PATCH	

Index

1. Introduction

This document describes the implementation process for Amadeus Quick Connect.

2. Concepts/Background

Amadeus Quick Connect is a set of REST/JSON business webservices aimed at non-GDS experts to implement an air booking engine. It based on REST/JSON technology and offers a simplified connectivity to customers. The feature scope is focused on the booking flow (from Offers to Orders).

3. Pre-requisites

Internal to Amadeus

- The Amadeus Account Manager or Delivery Manager oversees the activation process for the office ID and all relevant activation (IE: Pyton, B2B, MPTB etc..).
- API Consultant is responsible for the request of the Dedicated Access. Please follow link for detailed process on Dedicated Access: https://rndwww.nce.amadeus.net/confluence/display/WSC/REST+Dedicated+Access+-+Request+process

3.1 Office ID Settings

Internal to Amadeus

The following OID setup must be setup before starting Amadeus Quick Connect implementation:

- Office settings:
  BAC = Y => retrieve baggage allowance in TST
  DSD = Y => retrieve cabin information
  MPU = Y => is MasterPricer user
  PBC = Y => Pricing By Carrier

- Other settings:
  Switch R100 = Y => return the last ticketing date at Search & Price steps

  MPT & MPC Data Profiles => All Data profiles for MasterPricer TravelBoard and MasterPricer Calendar (if necessary).

  LSS user role - PNR_MOD => needed to book
- Mandatory:
  CODESHARE CARRIER 1 => if marketing carrier unknown, log it in MP traces
  SWEEPER 20 => check 20 first recommendations availability
  FB ALLOWANCE => retrieve baggage allowance
  REMOVE BLACKLIST CXR => remove EU blacklisted carrier recommendations
  EFT PER FLIGHT => to return each flight segment duration in search response

- Optional:
  FARE FAMILY INFO => to get Fare family info in search response
  COMPLEX ITINERARY => to be able to search for up to 6 itineraries (3 max otherwise)
  ANCILLARY SERVICES => to get price of first additional bag in search response
  SEARCH BY FBA 1 => to be able to search recommendations with free luggage included
  INCLUDE TLA (EOS with carrier codes / issuance allowed for the OfficeId) => to get LTC content in search response
  NO COMMERCIAL DUPE => to remove duplicate code share recommendations
  GENERATE MINI RULES => to get mini rules in search response (Warning: may be chargeable on certain markets)

3.2 Amadeus Quick Connect Configuration

The new configuration Questionnaire contains a summary of various mandatory parameters that covers multiple functional areas: Pyton connectivity, Booking preferences, Travel preferences, Pricing options, Queue's information, Static remarks, and Agency & Invoice information.

Customers will be asked to update this questionnaire to create the specific setup for their account. The configuration is done per user and per Office ID as each office ID can have a different setup. Additional parameters exist with pre-defined values and will be used when not specified in the service requests. They can be updated upon request. For a list of default values please contact the Amadeus Delivery Manager/Account Manager.

Internal to Amadeus - AQC Configuration
https://rndwww.nce.amadeus.net/confluence/pages/viewpage.action?pageId=2021819226

3.3 Technical Parameters on Client Side

3.3.1 Time Out

We strongly recommend to setup request/data timeout to the following values:

- Flight Offers Search: 31s
- Flight Orders Create: 91s
- All other APIs: 61s

3.3.2 Transaction per second

We strongly recommend to setup:

- Max number of stateless transactions per second: 10

The customer has to ensure he won't send more than 10 transaction per second and, if they need more, then it has to be requested via the Amadeus Account manager and it has to go through validation process.

4. Authentication

Authentication is done using a Dedicated Access which is prepared by the API Consultant once the project has started.

4.1 Dedicated Access

Dedicated Access is an access that uses a unique USAP which is used exclusively by each customer. It has specific configurations, and it is attached to a customer's own office ID(s). It also contains the API services that will be used by the application. Once the Dedicated Access has been created it will be available in the application.

4.2 API Key/API Secret

To initiate a connection to Amadeus, customer will need to use an API Key and API Secret. This information in the Developers Portal at application level. These keys are necessary to generate a security token which is used to establish a connection to Amadeus. The token validity period is 30 minutes. Development can be done to regenerate this token as needed. For more information on generating the token please see technical documentation available in the Developer's Portal:

https://developers.amadeus.com/enterprise/technical-doc/read/10049

4.3 Session management

REST is stateless by nature. Below please find additional details regarding the session management:

- An **Ama-Request-Id** is automatically generated by the Amadeus Quick Connect for each API call to track all underlying calls to providers (for internal issue analysis purpose)
- An **Ama-Client-Ref** must be generated by the customer for each end user session and be passed in the header of all requests for the entire booking flow (Search, Price and Book...Etc.). This allows for easier issue analysis by the support teams.

Example:

The generation of the Ama-Client-Ref illustrated below was done using Postman":

```
Content-Type: application/json
Authorization: ***CONCEALED***
Ama-Client-Ref: 123e4567-e89b-12d3-a456-426655440000
User-Agent: PostmanRuntime/7.21.0
Accept: */*
Cache-Control: no-cache
Postman-Token: 12345678-1234-12ad-12ce-ce1234f123d4
```

For example, you can find a true uuid generator on GitHub with the link below:

https://github.com/uuidjs/uuid

4.4 Endpoints

- **PDT (Test Environment):** "https://**test.**travel.api.amadeus.com/v?/API service name
  e.g.: "https://test.travel.api.amadeus.com/v2/shopping/flight-offers

- **PRD (Production environment):** https://travel.api.amadeus.com/v?/API service name
  e.g.: "https://travel.api.amadeus.com/v2/shopping/flight-offers

5. Content

The following air content is available via Amadeus Quick Connect (AQC):

- GDS/FSC: support for all FSC airlines (except SSR ePay ones)
- GDS/LTC: support only for EasyJet and Transavia
- GDS/EAC: support Amadeus LCC content from different sources.
  (Please see ”AQC EAC supplementary guide” available in the Developers Portal)
- Pyton: support all LCC content airlines from one source (Pyton) (API and/or screen scrapping)

General notes on Content:

- Not all GDS features/options are exposed through this API since the main purpose is to hide complexities and offer a simple way to access the content while exposing the most used features.
- Some features are not available or may not be applied fully for LCC content (ex: free luggage allowance, open jaws, …). The API does its best to hide any differences however, when it’s not possible, the response will raise some warnings to inform the user that some limitations apply.

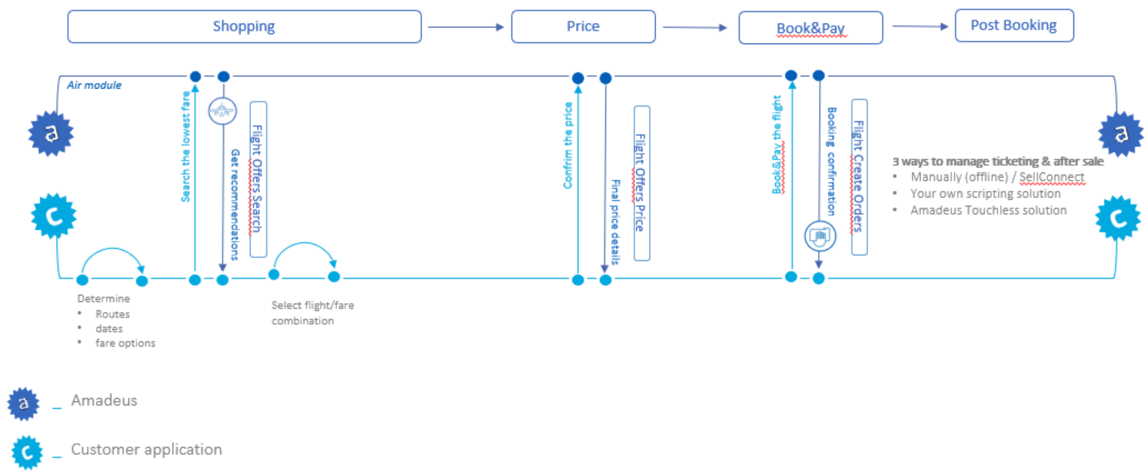
6. Functional Flow Diagrams

AQC uses three main REST API services to complete a booking, from offers to book.

6.1 Prime Booking Flow

The prime booking flow uses three main services to complete a booking FlightOffersSearch, FlightOffersPrice and FlightCreateOrders. In this flow diagram, no additional services are booked:

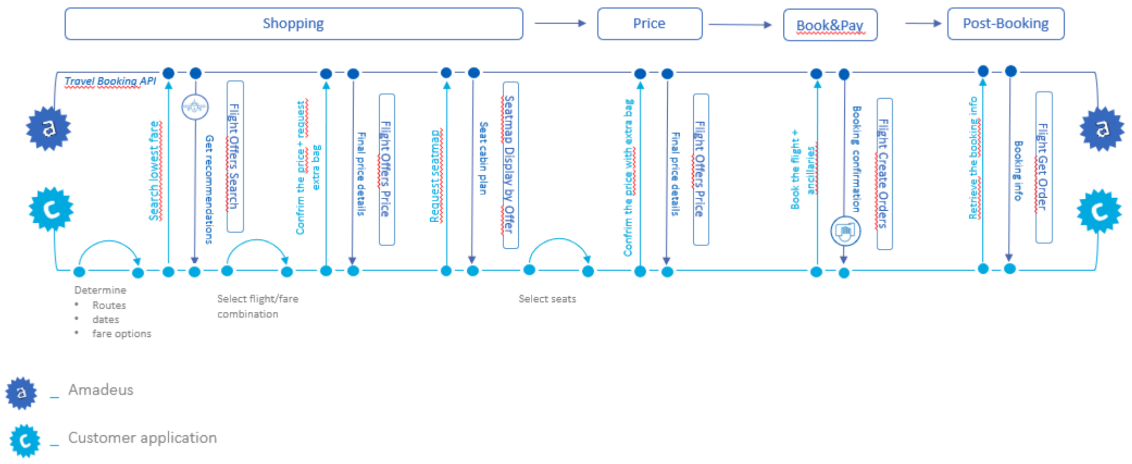
Prime Booking flow



6.2 Booking Flow with Additional Services (baggage and seats) and Retrieve Flight Order

Additional API services and operations can be added to the prime booking flow to add bag and seats. In addition, post-booking API services, Flight Get Order, can be sent to retrieve the flight order id as described in the diagram below:

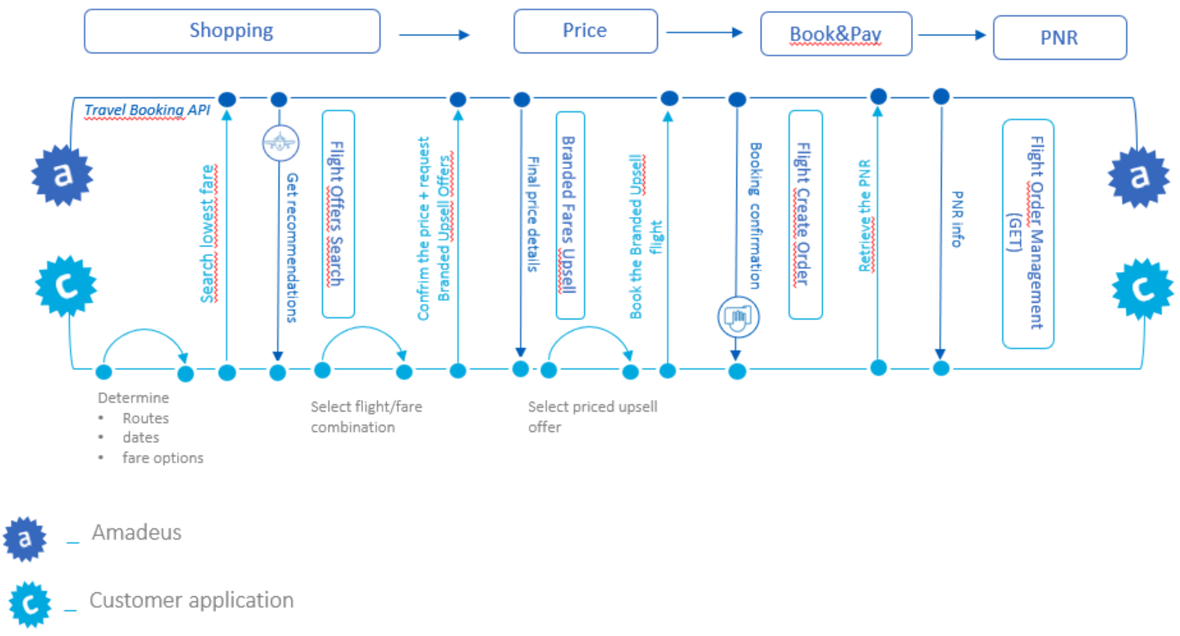
Booking Flow with ancillaries bags and seats + retrieve your booking information



6.2.1. Booking Flow with Branded Upsell Fares and Retrieve Flight Order

The Branded Fares Upsell provides the ability to upsell offers proposed by the carriers.

Search & Book Branded Upsell flight and get your PNR information



6.2.2. Functionality available per source:

Source	Mini Rules	B2B wallet	Bags	Seats	Upsell
GDS	Y	Y	Y	Y	Y
LTC	N	Y	Y	Y	N
Pyton	N	Y	Y	N	N
EAC	N	Y	Y	N	N

Note: Information on these specific functionalities is available in next chapters

7. API Details

API services described in the flow diagrams above:

7.1 Flight Offers Search

The FlightOffersSearch retrieves the cheapest flight recommendations operating for any flight with at least one seat available for sale on a requested route and travel dates.

Parameters can be used to optimize the request and can be automated in the configuration. For a list of these parameters and their default values or to request an update please contact the Amadeus Delivery Manager/Account Manager.

General notes on the service:

- Open Jaw and Multi-destination (+2 itineraries) are not supported with Pyton only search.
- Complex itineraries (three or more origin and destinations) is supported for GDS content only.

Margins (Margin Manager):

Markups/Fees/Discounts sent via Margin Manager are integrated and displayed in dedicated fields in AQC services replies (GDS only).

- In the Flight Offers Search, name and amount can be located in the travelerPricings object:

```
    "travelerPricings": [
                   {
                       "travelerId": "1",
                       "fareOption": "STANDARD",
                       "travelerType": "ADULT",
                       "price": {
                           "currency": "GBP",
                           "total": "122.12",
                           "base": "53.00",
                           "margins": [
                               {
                                   "amount": "2.65",
                                   "name": "AGENT_FEE"
                               }
                           ]
                       },
```

- The margin fees are not included in the any total amounts of the offer.
- Pyton / EAC / LTC content are excluded from Margin Manager scope.

A standard Flight Offers Search request must contain at least:

- One Origin and Destination
- One source
- One traveler’s ID and type
- One fare pricing option (Configurable)

A Standard Flight Offers Response will contain:

- Flight Offer – Applies to all passengers in the request
- Price – Total amount including taxes and fees.
- Pricing Option – Fare type
- Validating Carrier
- Traveler Pricings – Price per traveler including fare basis.

**Note: Complete information of mandatory/optional element is available in the swagger specs (Developers Portal - https://developers.amadeus.com/)

Search Parameters/objects to consider:

Object	Parameter	Description	Default Value
Search Criteria			
	addOneWayOffers	Allows to activate the one-way combinable (OWC) feature. At the search step the OWC option works with only 2 itineraries.	FALSE
	Maximum number of recommendation (maxFlightOffers)	Maximum number of flights offers returned. Highly recommended to use the maximum recommendation of 250.	250
	Maximum different price (maxPrice)	Post filtering on the AQC side and corresponds to the number of the different "total amount" in the response. Example: If in the search response we have 3 recommendations at 100.00EUR and 2 recommendations at 120.00EUR --> The number of different prices is 2 (100.00 and 120.00)	100
	allowAlternativeFareOptions	The allowAlternativeFareOptions means that if the requested fareOptions is not available, AQC will try to return Offers with Standard PTC. This also works for Pyton because only Standard PTC are managed by Pyton.	TRUE
	dateWindow	Either 1, 2 or 3 extra days before and after the specified date (MasterPricer Calendar)	Not Configurable
	timeWindow	1 to 12 hours around (both +and -) the local time.	Not Configurable
	additionalInformation		
	chargeableCheckedBags	If ”True” returns the price of the first chargeable bag for adults only when the airline participates in the ”Amadeus Ancillary Service”.	FALSE
	fareRules	If ”True” returns the fare rules before departure for each flight offer when available.	FALSE
	brandedFares	If ”True” returns the Airline Fare families names in LowFareSearch response	FALSE
	pricingOptions		
	fareType	Target type of fare: NEGOTIATED PUBLISHED CORPORATE	NEGOTIATED: FALSE PUBLISHED: TRUE CORPORATE; FALSE

	corporateCodes	List of corporate codes (max 6)	0
	Return Free Baggage Allowance Only (includedCheckedBagsOnly)	If ”TRUE” will display fares with free baggage allowance only. Caution when setting this parameter to TRUE, as normally the cheapest fares do not include free baggage allowance and will not be displayed.	FALSE
	refundableFare	If true, returns the flight-offers with refundable fares only.	Not configurable
	flightFilters		
	returnToDepartureAirport	If ”TRUE” it will ensure that departure and arrival airports of the journey will be the same (for example CDG -> NYC -> CDG and not CDG -> NYC -> ORY).	FALSE
	railSegmentAllowed	This flag enable/disable filtering of rail segment (TGV AIR, RAIL). (GDS only) Only Rail services returned by MasterPricer are available, e.g. Eurostar.	FALSE
	busSegmentAllowed	This flag enables/disables filtering on bus segments (GDS only). Only Rail services returned by MasterPricer are available.	FALSE
	Maximum Elapsed Flying Time (maxFlightTime)	This option allows to modify the value for the Elapsed Flying Time (EFT) MasterPricer option. It restricts flight within the specified percentage: - If no value is specified, then AQC will not filter on the Elapsed Flying Time. For example: flights lasting 12hrs can be shown if the price is low. - It is impossible to specify less than 100% as it will mean less than the shortest flight. - If a value of 999 is entered, then AQC will consider only scheduled flights with EFT of 9.99hrs maximum.	0
	carrierRestrictions		
	includedCarrierCodes	Ensures that the system will only consider the specified airlines. Example: if AF is requested only AF recommendations forced: one query is sent for AF as mandatory carrier.	None

	excludedCarrierCodes	Ensures that the system will not consider these airlines. Example: If AF is excluded, the query is sent on all airlines except AF.	None
travelers			
	TravelerType	To ensure the correct fares are returned, PTCs must be present in the query (Adult, Child, Infants etc.) when applicable. For a complete list of possible values and their limitations please check the swagger specifications. **Note: A complete list is available in the swagger specs (Developers Portal)	
		SEATED_INFANT – Not supported with Pyton content.	Not configurable
		HELD_INFANT – Ensure the to use the “associatedAdultId” option that corresponds to the adult traveler’s id who will share the seat.	Not configurable
	fareOptions	(up to 3 fareOptions) per traveler. **Note: A complete list is available in the swagger specs (Developers Portal)	
		STANDARD – Default if no fareOption is included in the request. **Note: If other fareOptions are included in the request and does not include STANDARD the system will not return these types of fares. Parameter “allowAlternativeFareOptions” to TRUE	Not configurable
		INCLUSIVE_TOUR – Type of negotiated fares (IIT)	Not configurable
		SPANISH_CANARY_RESIDENT - specific discount for Spanish residents valid only	Not configurable

7.2 Flight Offers Price

This service confirms fare, availability and the expected price (flights and additional services, if any). When requested, the standardized bag catalogue is returned for GDS, Pyton bag catalogue, Pyton booking options or credit card fees. By default, the system performs a best informative pricing (on GDS content).

General notes on the service:

- In Flight Offer Price and Flight Order Create the "fareOptions" are part of "travelerPricing" and therefore per "travelerId". This means that it mandatory for every traveler in the request.
  In the Flight Offers Search we need to specify fareOptions for each of the travelers in the request. For example, if there are two passengers and only one of them has an applicable fareOption e.g: "AIR_FRANCE_COMBINED_DISCOUNT_PASS" separate search requests will need to be sent, one for passenger with 'AIR_FRANCE_COMBINED_DISCOUNT_PASS', and one with 'STANDARD'.

Margins (Margin Manager):

- If the Office ID has been configured with Margin Manager, Margin (fees) will be included in the response of the API services end to end flow for informational purposes only.
- The margin fees are not included in the any total amounts of the offer.

Payment

- When using B2B Wallet or any other Virtual Card solution it is highly recommended to use corresponding payment card brands (MASTERCARD_IXARIS, VISA_IXARIS, MASTERCARD_AIRPLUS, UATP_AIRPLUS) to maximize price coherency between price & book calls.
- The payment card BIN number can be included to maximizes the Credit card fees reliability (if any) for GDS only:

```
"payments": [{
   "brand": "VISA_IXARIS",
   "binNumber": "444433",
   "flightOfferIds": [ “1” ]
}]
```

- Document requirements are returned in the response of the FlightOffersPrice:

```
{
          "travelerId": "t1",
          "genderRequired": true,
          "documentRequired": true,
          "dateOfBirthRequired": true,
          "redressRequiredIfAny": true,
          "residenceRequired": true
        },
```

Price Parameters/objects to consider:

Object	Parameter	Description	Default Value
URL Parameters			
	"forceClass=true"	To enforce the requested booking classes, added to the URL parameter. true, to for pricing with the specified booking class false, to get the best available price	Not configurable
	"bags=true"	To get extra bag options	Not configurable
	"credit-card-fees=true"	To obtain a list of credit card fees added to the URL parameter	Not configurable
	"detailed-fare-rules=true"	To obtain detailed fare rules added to the URL parameter	Not configurable
	"other-services=true"	To get services options	Not configurable
payments			
	”brand”	Credit card brand can be included in the request to ensure all credit card fees are included (if any).	Not configurable
	”binNumber”	The first 6 digits of the credit card (only GDS)	Not configurable

7.2.1. Direct Pricing from Cache Flight Information

Flight Offers Price also allows to price an offer from cache flight information (not from Flight Offers Search) by constructing a minimum pricing request with the specified flight information. For more details on creating the request please see the JSON examples section within this integration guide.

7.3 Flight Create Orders

This service creates an order based on a list of flight offers passed in the input. One single request will manage the booking and optionally the payment & issuance (for GDS content).

General notes on the service:

- Taxes are optional (and therefore not checked), however total amount is mandatory.
- Some remarks are automatically added in the order (i.e., the pricing query).
- The book without pricing functionality allows records to be created without TST by adding parameter "disablePricing": true in the request.

- Airlines sometimes add a more restrictive date for the ticket time limit. An SSR for the "ADVANCE TICKET TIME LIMIT" is inserted automatically into the order by the airlines who subscribed to the corresponding Amadeus product to manage their Ticketing Time Limits. AQC only reads this information if present in the PNR:

  Example:
  SR ADTK TO KL BY 22FEB 1200 PAR OTHERWISE WILL BE XLD.
- Up to 9 passengers can be issued in one Flight Order.
- The titles (MR, MRS) are added automatically in AQC by taking the "gender" of each traveler.
- User can specify the issue=true parameter on the URL in order to trigger the issuance for GDS content. For more information, please refer to the section "DOCUMENT ISSUANCE" within this integration guide.
- In case of issuance error, the Flight Order is created however, a warning is displayed in the response advising issuance was not possible. In this case, please handle offline.
- The usage "TEST" as names for travelers are not allowed by some airlines/providers, therefor you should not use it.

Form of payment:

- Credit card payment is limited to B2B wallet, corporate card or any Virtual card solution.
- For GDS content, a form of payment is not mandatory when creating an order. However, it is mandatory for issuance. If payment was not added during the creation of the order it will need to be added offline prior to issuing the ticket.
- For LCC content, payment information is mandatory during the creation of an order. The form of payment can be B2B wallet, corporate card or any virtual card solution. Using personal traveler cards will fail due to the strong authentication (3DSecure/PSD2) process. Enhancements are ongoing to allow traveler cards to be used for some specific content, for more details, please consult with your Amadeus Delivery Segment.
- Amadeus Quick-Connect does not support a form of payment with a service fee.
- For all countries (based on the Country Code of the OID) the AQC the formOfPayment "CashPayment" will be added as FP element (Form of Payment Element) 'CA' in the PNR. If the FOP element must be 'FP CASH' instead of 'FP CA' for a country, it must be added with a new rule! This must be requested by your account manager.
- NONREFUNDABLE payment method is meant to insert the NONREF FOP that can be used in some market for BSP payment. It may not work in all market depending on FOP table.
- PYTON payment method can be used only for PYTON bookings in order to delegate the payment to Pyton system, using the customer payment configuration defined there (instead of AQC managing the payment). This has been implemented primarily to ease the migration of Pyton direct customers to AQC.

Margins (Margin Manager):

- Margin (fees) will be included in the response of the Flight Offers Search and Price and can also be included in the request of the Flight Create Order. Office ID will need to be configured with rules Margin Manager.
- The margin fees are not included in the any total amounts of the offer.

Parameter		Description	Default Value
Travelers			
	Add Date of Birth In PNR	If true, add date of birth in the PNR's NM element for all pax types / if false, only add it when mandatory (infant)	FALSE
	Add CTCE and CTCM in PNR	If mobile number (CTCM) and e-mail (CTCE) are given in the contact at traveler level the SR CTCM and CTCE will be added for each traveler.	TRUE ** NOTE: CTCE and CTCM cannot be entered in the request. They can only be configured to be entered automatically)
	documents	Document requirements can be found in the response of the Flight Offers Price. Please see Flight Offers Price section for more information. Management of the document in AQC: • In PriceRS\BookingRequirements the attribute documentRequired ="true" if one city of the trip concerns these countries: "AE","AF","CA","CN","CU","IL","IN","IR","JP","KP","MX","PK","US","YE" • It's just an informative flag, AQC will not block the booking if the Document is not added in the Book Request. • Document information such as passport is mandatory for ticketing when itinerary concerns any of the countries specified above.	
		○ When ”documentRequired” is TRUE, passport is required. ○ When ”redressRequired” is TRUE, it can be included (if traveler has one) but not mandatory as normally passport is sufficient. ○ When ”residenceRequired” is TRUE, this is required for PYTON orders. Information can be included in the general contacts	

Parameter		Description	Default Value
		object or it can be configured internally (configuration questionnaire) in the agency contact information. ○ SSR DOCS are automatically created (for APIS purpose) even when a book request specifies at least the first/last name, gender and DOB of a passenger while not providing any documents (passport, id card) ○ SSR DOCO is automatically created (for APIS purpose) when a book request specifies at least a REDRESS number. ○ SSR DOCA is automatically created when "address" information is included at traveler level. ○	
contacts		Information in this object corresponds to the agency information	
	Address emailAddress phone		The values configured will be used to enter billing address (AB), agency e-mail address (APE) and agency phone number (AP).
	Purpose	The options "STANDARD" and "STANDARD_WITHOUT_TRANSMISSION" correspond to a CTCR in the PNR. The option "STANDARD_WITHOUT_TRANSMISSION" the information is not sent to the airline.	Not Configurable
remarks			
	general	These remarks are used to add additional comments or information concerning an order. There are different types of general remarks: **GENERAL_MISCELLANEOUS**– add general information or comments "RM".	The values configured will be used to enter the remarks automatically.

Parameter		Description	Default Value
		CONFIDENTIAL – add a confidential remark ”RC”. **INVOICE** – add a remarks that will show in an invoice ”RIR”. **TICKETING_MISCELLANEOUS**– add a miscellaneous ticketing information for accounting purposes, ”FS”. **TOUR_CODE** – add a tour code as required by carrier. A tour code is also used for corporate loyalty cards.	
	airline	These remarks are used to add additional comments or information regarding a passenger and is sent to the airline(s) in the flight offer: **Other Service Information** – sends additional information regarding a passenger(s) to the airline(s) in the flight offer request (OSI element in the PNR) **Other Services** - is only for PRIORITY_BOARDING and AIRPORT_CHECKIN for PYTON offers. **Keyword** – adds special information about an order that can be used by the airline. Please contact the carrier for the supported code as not all airlines support this element. (SK element in the PNR).	Not Configurable
formOfPayments			
		If form of payment is configured it will be entered in the PNR automatically. - account - cash - check - non-refundable If configured, this will be used in case b2bWallet payment fails. *Note: Credit cards cannot be configured in the admin	None
Objects to Consider			

Parameter		Description	Default Value
	queuingOfficeId	If an office ID is entered this will change the Queuing Office in the RP field of the PNR.	
	ownerOfficeId	If an office ID is entered this will transfer responsibility of the PNR to the specified office ID.	
TicketingAgreementOption			
	CONFIRM	This will enter a TKOK in the ticketing element.	
	DELAY_TO_QUEUE	This will enter a TKTL and send it to the specified or configured Queue. In case no value is entered it will be sent to Queue 8C0	
	DELAY_TO_CANCEL	This will enter a TKXL and allows the system to do an automatic cancellation on the specified date.	
Configurable parameters			
	Price Margin	The price margin allows the specified percentage to be allowed in case the fare increases or decreases. It is highly recommended to set a value for LCC bookings (even more for screen scrapped airlines via Pyton) to avoid booking rejection due to price discrepancies.	0
	Name truncation	It is used only for the traveler's/name part (corresponding to the NM element of the PNR) and enables a truncation	TRUE
	Stop on Service Booking failure	Stop book process if ancillaries booking fails	TRUE
	Allow Booking if Lower Price	Allow booking process to continue if the price drops below the price margin	TRUE
	Add MIS Segment	Adds a MIS segment in the PNR to prevent the PNR purge for 1 year after its creation (RU MISC)	FALSE

7.4 Flight Order Management

The Flight Order Management is an open API that allows to retrieve, delete and manipulate a flight order previously created.

List of supported operation by resource:

7.4.1. DELETE

This operation only applies to flight order containing non-issued GDS flight offers.

```
DELETE  https://test.travel.api.amadeus.com/v1/booking/flight-order/eJzTd9cPNfX09DUFAAs-GAkQ=
```

General notes on the service:

- To DELETE orders that contains 2 GDS records (OWC):
  - Use the GetOrderByRecLoc- this API allows to retrieve a flightorder id that matches a single GDS PNR via its RECLOC (that can be found in the OrderCreate response).
  - Use the flightorder id to DELETE.
- When a DELETE order has been successful a standard JSON empty list is returned as AQC cannot access that specific REST resource. Example:

```
{
 "meta": {
 "count": 0,
 "links": {
 "self": "https://test.travel.api.amadeus.com/v1/booking/flight-orders/by-reference?reference=MCWXXX&originSystemCode=GDS";
 }
 },
 "data": { }
}
```

7.4.2. GET

This operation retrieves an existing flight order. It contains flight information, fare detail, ancillary services, traveler information and ticket details.

```
GET  https://test.travel.api.amadeus.com/v1/booking/flight-orders/eJzTd9f38Q039XcBAAtXAl4=
```

Response:

New additional information is now displayed in the response of the GET Order:
Segment Status:

```
{
    "meta": {
        "count": 1,
        "links": {
            "self": "https://test.travel.api.amadeus.com/v1/booking/flight-orders/eJzTd9cPMo0Id44CAAtrAnk%3D"
        }
    },
    "data": {
        "type": "flight-order",
        "id": "eJzTd9cPMo0Id44CAAtrAnk%3D",
        "queuingOfficeId": "PARKU2104",
        "associatedRecords": [
            {
                "reference": "R5XWCZ",
                "originSystemCode": "QR",
                "flightOfferId": "1"
            },
            {
                "reference": "R5XWCZ",
                "creationDate": "2021-07-28T15:29:00",
                "originSystemCode": "GDS",
                "flightOfferId": "1"
            }
        ],
        "flightOffers": [
            {
                "type": "flight-offer",
                "id": "1",
                "source": "GDS",
                "nonHomogeneous": false,
                "lastTicketingDate": "2021-07-29",
                "itineraries": [
                    {
                        "segments": [
                            {
```

```
                "departure": {
                    "iataCode": "CDG",
                    "terminal": "2E",
                    "at": "2021-11-15T15:05:00"
                },
                "arrival": {
                    "iataCode": "DOH",
                    "at": "2021-11-15T23:30:00"
                },
                "carrierCode": "QR",
                "number": "40",
                "aircraft": {
                    "code": "77W"
                },
                "operating": {},
                "bookingStatus": "CONFIRMED",
                "segmentType": "ACTIVE",
                "isFlown": false,
                "id": "1",
                "numberOfStops": 0
            },
```

Included Meals (if offered by carrier):

```
             "fareDetailsBySegment": [
{
"segmentId": "1",
"cabin": "ECONOMY",
"fareBasis": "ODNNEOB2",
"class": "O",
"includedCheckedBags": {
"quantity": 0
},
"mealServices": [
{
"label": "Food and beverages for purchase"
}
]
```

Note: The flightOrderId should be URL- encoded before putting it in the endpoints. IDs can sometimes contain "/" so it is recommended to URL-encoded (replace the "/" with %2F%) before putting them in an endpoint.

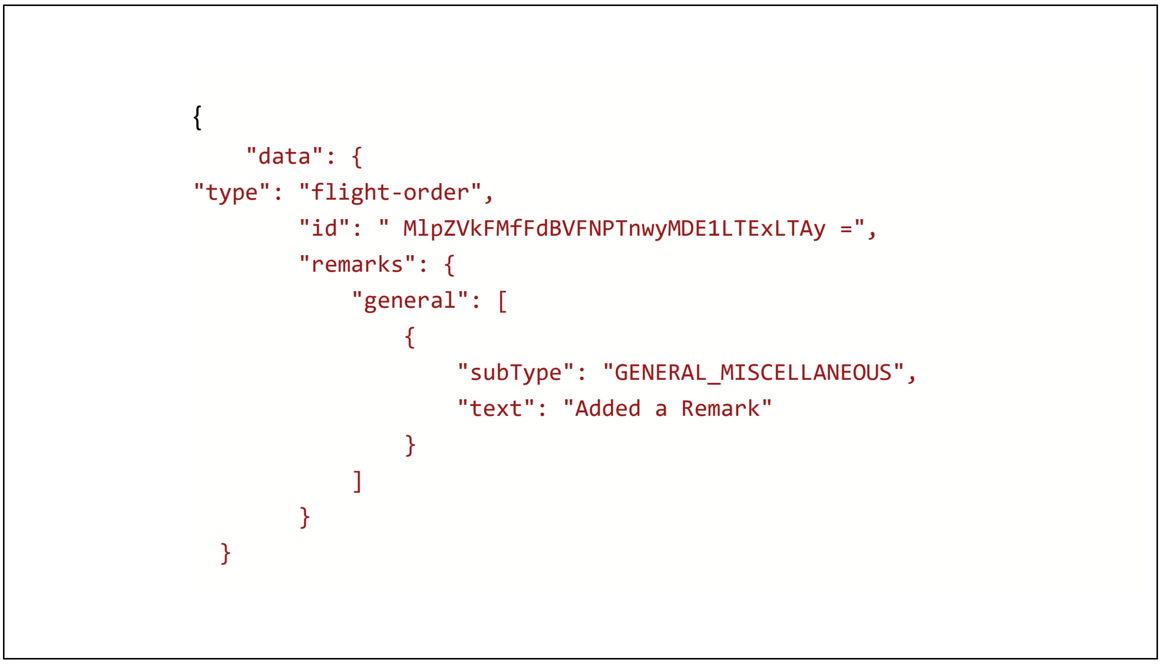
7.4.3.PATCH

This operation currently allows to modify an existing Flight Order for GDS content only. It is part of guidelines and recommendation to perform a GET before issuing a PATCH. The PATCH must be applied to the payload of the GET response, not to the payload of the response from the Create Order.

Example:

```
PATCH  https://test.travel.api.amadeus.com/v1/booking/flight-order/MlpZVkFMfFdBVFNPTnwyMDE1LTExLTAy
```

Response:



```
{
  "data": {
"type": "flight-order",
    "id": " MlpZVkFMfFdBVFNPTnwyMDE1LTExLTAy =",
    "remarks": {
      "general": [
        {
          "subType": "GENERAL_MISCELLANEOUS",
          "text": "Added a Remark"
        }
      ]
    }
 }
```

General notes on the service:

The possible additions allowed on an orderId are:

- Seats - This service requires activation, please contact your Amadeus Account Manager for more information. A seat map integration guide is available in the Developers Portal: https://developers.amadeus.com/enterprise/implementation-guides?page=1&count=10&filter=title,seatmap&sorting=title,asc
- Documents
- Remarks
- Contacts
- Queue Place
- Commissions (FM)
  Current limitations - Not possible to add commissions in Prime Booking Flow.

Example request to add automatedProcess, documents, contacts, remarks and commissions (The element Commissions is used to record commissions earned from the sale of a ticket, as percentage or amount. Associated to FM fare element in PNR):

```
{
    "data": {
        "type": "flight-order",
        "id": "eJzTd9cP8nAM8TEHAAtUAlg%3D",
        "automatedProcess": [
            {
                "code": "IMMEDIATE",
                "queue": {
                    "number": "2",
                    "category": "0"
                },
                "officeId": "MIA1A08QC"
            }
        ],
        "documents": [
                    {
                        "documentType": "PASSPORT",
                        "number": "567AN34546",
                        "issuanceLocation": "FRANCE",
                        "issuanceDate": "2019-09-25",
                        "expiryDate": "2025-09-25",
                        "issuanceCountry": "FR",
                        "nationality": "FR",
                        "holder": true
                    }
                ],
                 "contacts": [
            {
                "addresseeName": {
                    "firstName": "TRAVELLAB",
                    "lastName": "TRAVELLAB"
                },
                "companyName": "Travel Lab SAS",
                "purpose": "STANDARD",
                "phones": [
                    {
                        "deviceType": "LANDLINE",
                        "countryCallingCode": "33",
                        "number": "888812345"
                    }
                ],
                "address": {
                    "lines": [
                        "AMADEUS ROAD"
```

```
          ],
          "postalCode": "99999",
          "cityName": "NICE",
          "countryCode": "FR"
        }
      }
    ],
     "remarks": {
      "general": [
        {
          "subType": "GENERAL_MISCELLANEOUS",
          "text": "CUSTOMER WANTS AN UPGRADE"
        }
      ]
    },
     "commissions": [
      {
        "controls": [
          "MANUAL"
        ],
        "values": [
          {
            "commissionType": "NEW",
            "amount": "123.45"   ← can be an amount
          },
          {
            "commissionType": "VAT_ON_NEW"
          }
        ],
        "travelerIds": [
          "2"
        ]
      },
      {
        "controls": [
          "MANUAL"
        ],
        "values": [
          {
            "commissionType": "NEW",
            "percentage": "15"   ← or it can be percentage
          }
        ],
        "travelerIds": [
          "1"
        ]
      }
    ]
  }
}
```

Patch also allows the modification of:

- Ticketing element (TK)
- To PATCH orders that contains 2 GDS records (OWC):
    - Use the GetOrderByRecLoc- this API allows to retrieve a flightorder id that matches a single GDS PNR via its RECLOC (that can be found in the OrderCreate response).
    - Use the flightorder id to PATCH.

Note: PATCH can only add new elements to the flight order that aren't present yet. Nothing can be modified!

7.5 Get Flight Order By Record Locator

The "Get Flight Order By Record Locator" allows you to retrieve booking information using the Record Locator (PNR) in the request.

```
https://test.travel.api.amadeus.com/v1/booking/flight-orders/by-reference?reference=
WRWVCJ&originSystemCode=GDS
```

Response:

```
{
    "meta": {
        "count": 1,
        "links": {
            "self": "https://test.travel.api.amadeus.com/v1/booking/flight-orders/by-
reference?reference=WRWVCJ&originSystemCode=GDS"
        }
    },
    "data": [
        {
            "type": "flight-order",
            "id": "eJzTd9cPDwoPc%2FYCAAwDAok%3D",
            "queuingOfficeId": "NCE1A01AP",
            "associatedRecords": [
                {
                    "reference": "RECLOC",
                    "originSystemCode": "AA",
                    "flightOfferId": "1"
                },
                {
                    "reference": "WRWVCJ",
                    "creationDate": "2021-08-31T16:58:00",
                    "originSystemCode": "GDS",
                    "flightOfferId": "1"
                }
            ],
            "flightOffers": [
                {
                    "type": "flight-offer",
                    "id": "1",
                    "source": "GDS",
```

General notes on the service:

- This service is mandatory to DELETE orders that contains 2 GDS records (OWC). For More information please see “Flight Order Management” section above.
- The response will include an order “id” which can be used for post-booking transactions.

7.6 Flight Order Issue

Flight Order Issue allows you to issue tickets for Full-Service Carriers from an existing orderId.

```
GET: https://test.travel.api.amadeus.com/v1/booking/flight-orders/eJzTd9cPjLBwDnE-FAAtlAmM%3D/issuance
```

The ticket number will be available in the response:

```
 "tickets": [
      {
        "documentType": "ETICKET",
        "documentNumber": "057-3322491266",
        "documentStatus": "ISSUED",
        "travelerId": "1",
        "segmentIds": [
          "1",
          "2"
        ]
      },
      {
        "documentType": "EMD",
        "documentNumber": "057-1858808477",
        "documentStatus": "ISSUED",
        "travelerId": "1",
        "segmentIds": [
          "1"
        ]
```

General notes on the service:

- Form of payment is mandatory for issuance. If payment was not added during the creation of the order it will need to be added offline for issuance purposes.
- This service is used to issue both tickets and EMDs when present in the order.
- EMD standalone is not available.

7.7 Branded Fares Upsell

The Branded Fares Upsell REST/JSON service enables you to get upgraded Branded Fares flight recommendations available from a flight offer. Branded Fares are airline offers in which airline bundles its airfares with options and features, such as refundability and miles accrual, or a pre-reserved seat, baggage, and meal. It is usually used after the Flight Offers Search service.

The option to trigger Branded Fares Upsell is to send the upselling parameter in the POST URL right after the shopping request.

The response will return the priced offer from the Shopping response along with the additional branded upsell proposals. The first offer is the original priced offer. The other offers are the upsells.

The branded fare information is returned under the fareDetailsBySegment as shown in the example below. The amenities section provides the description details for each branded fare. It will also return the following Boolean advising if the amenity is included or excluded. For example:

In the grey section below, the amenityType is for Baggage which is a carry on up to 40 LI 101 LCM. The isChargeable returned “False” which means the carry-on baggage is included under this branded fare.

The second example is highlighted in blue below. The “isChargeable” returned “True”. This indicates it is not included with this branded fare. In order to have these amenities, the customer will need to upgrade to a higher upsell offer.

Example:

POST /shopping/flight-offers/upselling

Return a list of upsell Flight Offers based on given Flight Offers. The response will return all of the possible upsell branded fares available along with the amenities.

```
{
    "meta": {
        "count": 4
    },
    "data": [
        {
…
            "type": "flight-offer",
                    "fareDetailsBySegment": [
                        {
                            "segmentId": "8",
                            "cabin": "ECONOMY",
                            "fareBasis": "QVALZNN3",
                            "brandedFare": "MAIN",
                            "class": "Q",
                            "includedCheckedBags": {
                                "quantity": 0
                            },
                            "amenities": [
                                {
                                    "code": "0MM",
                                    "description": "CARRY ON UP TO 40 LI 101 LCM",
                                    "isChargeable": false,
                                    "amenityType": "BAGGAGE"
                                },
                                {
                                    "code": "0MU",
                                    "description": "CARRY ON UP TO 45 LI 115 LCM",
                                    "isChargeable": false,
                                    "amenityType": "BAGGAGE"
                                },
                                {
                                    "code": "0BV",
                                    "description": "SAME DAY STANDBY",
                                    "isChargeable": false,
                                    "amenityType": "STANDBY"
                                },
                                {
                                    "code": "059",
                                    "description": "CHANGEABLE TICKET",
                                    "isChargeable": false,
                                    "amenityType": "BRANDED_FARES"
                                },
                                {
                                    "code": "057",
                                    "description": "AADVANTAGE MILES",
                                    "isChargeable": false,
                                    "amenityType": "BRANDED_FARES"
                                },
                                {
                                    "code": "0B1",
                                    "description": "IN FLIGHT ENTERTAINMENT",
                                    "isChargeable": false,
                                    "amenityType": "ENTERTAINMENT"
                                },
                                {
                                    "code": "032",
                                    "description": "STREAMING VIDEO",
                                    "isChargeable": false,
                                    "amenityType": "ENTERTAINMENT"
                                },
                                {
                                    "code": "050",
                                    "description": "BASIC SEAT",
                                    "isChargeable": false,
                                    "amenityType": "BRANDED_FARES"
                                },
                                {
                                    "code": "PSA",
                                    "description": "PRE RESERVED SEATS",
                                    "isChargeable": false,
                                    "amenityType": "PRE_RESERVED_SEAT"
                                },
                                {
                                    "code": "SBO",
                                    "description": "STANDARD BOARDING",
                                    "isChargeable": false,
                                    "amenityType": "TRAVEL_SERVICES"
                                },
                                {
                                    "code": "0CC",
                                    "description": "FIRST CHECKED BAG",
                                    "isChargeable": true,
                                    "amenityType": "BAGGAGE"
                                },
                                {
                                    "code": "0CD",
                                    "description": "SECOND CHECKED BAG",
                                    "isChargeable": true,
                                    "amenityType": "BAGGAGE"
                                },
                                {
                                    "code": "0CE",
                                    "description": "THIRD CHECKED BAG",
                                    "isChargeable": true,
                                    "amenityType": "BAGGAGE"
                                },
                                {
                                    "code": "0CP",
                                    "description": "SAME DAY FLIGHT CHANGE",
                                    "isChargeable": true,
                                    "amenityType": "STANDBY"
                                },
                                {
                                    "code": "03P",
                                    "description": "PRIORITY CHECKIN",
                                    "isChargeable": true,
                                    "amenityType": "TRAVEL_SERVICES"
                                },
                                {
                                    "code": "0BX",
                                    "description": "ADMIRALS CLUB",
                                    "isChargeable": true,
                                    "amenityType": "LOUNGE"
                                },
                                {
                                    "code": "0AV",
                                    "description": "PREMIUM BEVERAGE",
                                    "isChargeable": true,
                                    "amenityType": "MEAL"
                                },
                                {
                                    "code": "0AT",
                                    "description": "MEAL OR SNACK",
                                    "isChargeable": true,
                                    "amenityType": "MEAL"
                                },
                                {
                                    "code": "0CL",
                                    "description": "WI FI",
                                    "isChargeable": true,
                                    "amenityType": "ENTERTAINMENT"
                                },
                                {
                                    "code": "MCE",
                                    "description": "MAIN CABIN EXTRA",
                                    "isChargeable": true,
                                    "amenityType": "PRE_RESERVED_SEAT"
                                },
                                {
                                    "code": "05Z",
                                    "description": "EXTRA LEG ROOM",
                                    "isChargeable": true,
                                    "amenityType": "BRANDED_FARES"
                                },
                                {
                                    "code": "0G6",
                                    "description": " PRIORITY BOARDING",
                                    "isChargeable": true,
                                    "amenityType": "TRAVEL_SERVICES"
                                }
                            ]
                        },
```

General notes on the service:

- Branded Fares Upsell verb is only available on a Flight-Offer containing the "brandedFares" information
- Parameter “homogeneous branded fares” allows to choose if upsell can be done in homogeneous branded fares (Same branded fares in all segments of one Offer).
- The upselling is based on the fare families and not on the class of service or cabin. For example, for a specific flight X where we have:

Class (ordered by cheapest)	Fare Family	Cabin
A	BASIC	ECONOMY
B	BASIC	ECONOMY
C	ECOPLUS	ECONOMY
D	ECOPLUS	ECONOMY
E	BUSINESS	BUSINESS
F	BUSINESS_PLUS	BUSINESS

7.8 Queue Management

7.8.1. Queue List

This API will provide list of PNR and related information in the response based on the queue number provided in the input.

Request:
The queue number is mandatory in the request. In the example below the request is for a list of PNRs in Queue 5:

```
GET:   https://test.travel.api.amadeus.com/v1/office/queues/5
```

Response:
1 PNR was found in Queue 5 and relevant information is displayed:

```
{
    "meta": {
        "count": 1,
        "links": {
            "self": "{host}/office/queues/{queue-id}"
        }
    },
    "data": [
        {
            "reference": "QPCW3U",
            "creation": {
                "dateTime": "10-5-2021"
            },
            "travelers": [
                {
                    "names": [
                        {
                            "lastName": "VERMEIL"
                        }
                    ]
                }
            ],
            "products": [
                {
                    "airSegment": {
                        "segment": {
                            "arrival": {
                                "iataCode": " "
                            },
                            "departure": {
                                "iataCode": "PAR",
                                "localDateTime": "06JUN21"
                            },
                            "carrierCode": "HTL"
                        }
                    }
                }
            ],
            "type": "PNR"
        }
    ]
}
```

Parameters can be used to display specific information from the specified queue:

Request:

```
GET:   https://test.api.amadeus.com/v1/office/queues/5?officeId=PAR1A015U&cate-
gory=0&max=50
```

- Office ID: A guest office ID can be entered providing that all EOS agreements are in place to access the guest office ID queues. If no office ID is entered in the request the default office ID queues will be displayed.
- Category: A category number can be specified. If not entered it will default to 0.
- Max: These parameters allows you to control the amount of PNRs displayed.

7.8.2. Queue Place PNR

PNRs can be place in a specific Queue in the request of the “Flight Create Order” object “automatedProcess”:

- Immediate: Places the PNR in Queue as soon as it is created.
- Delayed: Places a PNR in a Queues delaying for the specified amount of days.
- Error: Places PNR in specified Queue in case of errors (ticketing, EMDs, etc.)

```
"automatedProcess": [
      {
        "code": "IMMEDIATE",
        "queue": {
          "number": "90"
        },
        "officeId": "LON1A0111"
      },
      {
        "code": "DELAYED",
        "queue": {
          "number": "92"
        },
        "delay": "1D",
        "officeId": "LON1A0111"
      },
      {
        "code": "ERROR",
        "queue": {
          "number": "93"
        },
        "officeId": "LON1A0111"
      }
    ]
```

These options can also be configured in the AQC administration therefore PNRs will be placed automatically in the specified Queue(s). To override any configured value, ensure to specify a new value in the request and it will override the configured one (please see example below).

7.9 Flight Availabilities Search

Flight Availabilities Search is an open search service that enables you to get “flight-availabilities" containing air availability, schedules and timetable information. The API provides a list of flight availabilities from a given origin (city or airport), for a given date (or date range) and for a given list of passengers.

This service is useful for Schedule Driven booking flows.

Two new attributes are added on the existing model SearchCriteria:

- includeClosedContent: Boolean – This attribute will allow to include or not the closed booking classes, departed flights and cancelled flights in the response.
- class: String – This attribute allows to include only the requested booking class in the response.

Note: PYTON is not available for this search.

POST: **/v1/shopping/availability/flight-availabilities**

```
{
    "originDestinations": [
        {
            "id": 1,
            "originLocationCode": "PAR",
            "destinationLocationCode": "NYC",
            "departureDateTime": {
                "date": "2021-09-22",
                "time": "10:30:00"
            }
        }
    ],
    "travelers": [
        {
            "id": 1,
            "travelerType": "ADULT"
        }
    ],
    "sources": [
        "GDS"
    ],
  "searchCriteria": {
        "excludeAllotments": false,
        "flightFilters": {
            "cabinRestrictions": [
                {
                    "cabin": "PREMIUM_ECONOMY",
                    "originDestinationIds": [
                        1
                    ]
                }
            ],
             "connectionRestriction": {
                "maxNumberOfConnections": 2,
                "airportChangeAllowed": false,
                "technicalStopsAllowed": true
            }
        },
        "includeClosedContent": false,
        "class": "A"
    }
}
```

General notes on the service:

- "**tourAllotment"** information is retuned in the response when negotiated space is applicable.

8. Focus on specific items

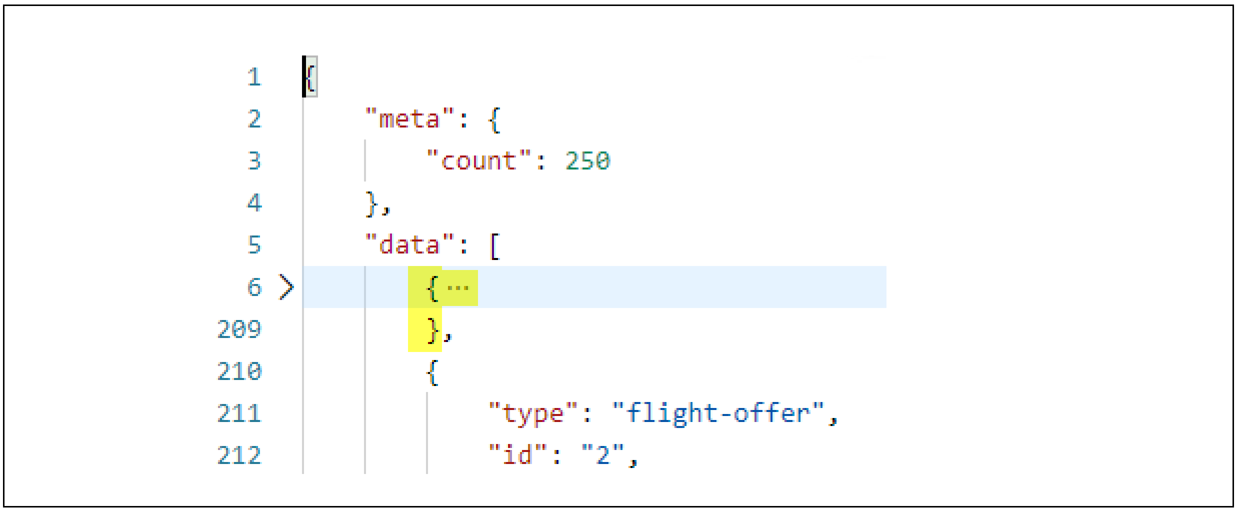
8.1 Selecting a recommendation (Object reuse)

The Amadeus Quick Connect offers the option to reuse an object between APIs in subsequent APIs to ensure the correct information is sent. It is **strongly recommended** to use this option to avoid errors at booking time as it transfers all mandatory objects into the subsequent calls.

Postman is used in this guide only to illustrate this concept

1. From the FlightOffersSearch response collapse the information of the selected recommendation and copy:

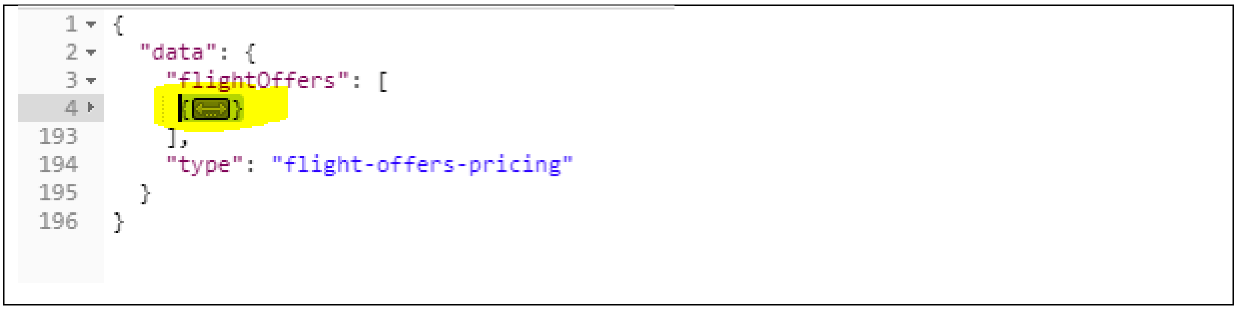
Example:



```
{
    "meta": {
        "count": 250
    },
    "data": [
        {⋯
        },
        {
            "type": "flight-offer",
            "id": "2",
```

2. Paste into the FlightOffersPrice request located right below the flight offers object:

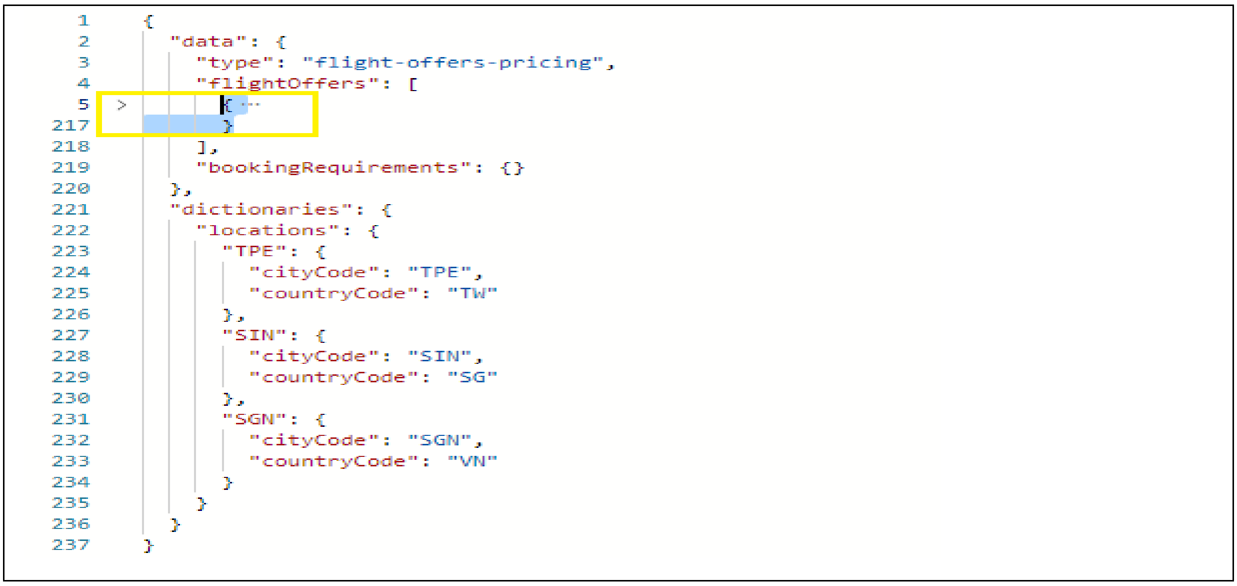
Example:



```
{
  "data": {
    "flightOffers": [
      {⟷}
    ],
    "type": "flight-offers-pricing"
  }
}
```

3. From the FlightOffersPrice response, collapse the flight offer returned in the response with the confirmed pricing.

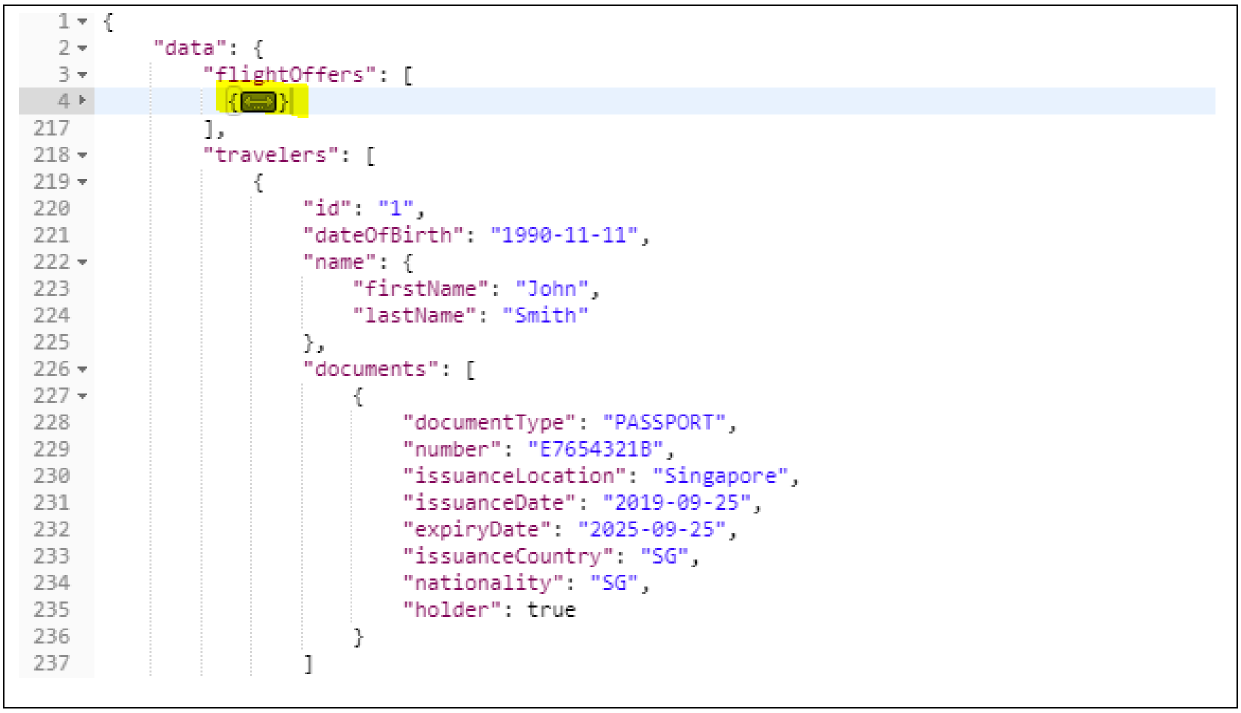
Example:



```
{
  "data": {
    "type": "flight-offers-pricing",
    "flightOffers": [
>     { ··
      }
    ],
    "bookingRequirements": {}
  },
  "dictionaries": {
    "locations": {
      "TPE": {
        "cityCode": "TPE",
        "countryCode": "TW"
      },
      "SIN": {
        "cityCode": "SIN",
        "countryCode": "SG"
      },
      "SGN": {
        "cityCode": "SGN",
        "countryCode": "VN"
      }
    }
  }
}
```

4. Paste it into the FlightCreateOrders located right below the flight Offers section and include the rest of the needed information to complete the booking.

Example:



```
{
    "data": {
        "flightOffers": [
            {↔}
        ],
        "travelers": [
            {
                "id": "1",
                "dateOfBirth": "1990-11-11",
                "name": {
                    "firstName": "John",
                    "lastName": "Smith"
                },
                "documents": [
                    {
                        "documentType": "PASSPORT",
                        "number": "E7654321B",
                        "issuanceLocation": "Singapore",
                        "issuanceDate": "2019-09-25",
                        "expiryDate": "2025-09-25",
                        "issuanceCountry": "SG",
                        "nationality": "SG",
                        "holder": true
                    }
                ]
```

8.2 B2B wallet

The Amadeus Quick Connect allows a more flexible and complete usage of B2B Wallet.

In the FlightOffersPrice, payment information can be entered to obtain all applicable credit card fees.

Example:

```
"payments": [
   {
      "brand": "MASTERCARD_IXARIS",
      "flightOfferIds": [ “1” ]
   }
]
```

In the FlightCreateOrders, the standard option to use B2B wallet is to use the default/optimizer mode when selecting a recommendation. This default/optimizer will select the default card or the best available card for this Flight offer. Please ensure configuration has been setup.

Example:

```
"formOfPayments": [
   {
      "b2bWallet": {                     ⍰  B2B wallet option
         "flightOfferIds": [
            "1"                          ⍰  first recommendation
         ]
      }
   }
],
```

In the reply the external reference of the generated card, the selected brand and the amount returned is included.

Example:

```
"formOfPayments": [
   {
      "b2bWallet": {
         "cardId": "xxxxxxxx",            ←  external card reference
         "virtualCreditCardDetails": {
            "brand": "MASTERCARD",
            "amount": "235.67",
            "currencyCode": "EUR"
         },
         "flightOfferIds": [
            "1"                           ←  used for the first recommendation
         ]
      }
   }
],
```

The cardUsageName field can be used to target a specific card from the B2BWallet portfolio as configured specifically for each customer. There are three default names that may be used if the corresponding cards are available for the customer:

- PREPAID: to select the MASTERCARD Ixaris
- CREDIT: to select the MASTERCARD Airplus
- DEBIT: to select the VISA debit Ixaris

Some additional customer defined names may also be used once defined in the B2BWallet configuration.

Example request with cardUsageName:

```
"formOfPayments": [
   {
      "b2bWallet": {
         "cardUsageName": "PREPAID",
         "flightOfferIds": [
            "1"
         ]
      }
   }
],
```

In the reply the external reference of the generated card, the selected brand and the amount returned is included.

Example response with cardUsageName:

```
"formOfPayments": [
   {
      "b2bWallet": {
         "cardId": "xxxxxxxx",
         "cardUsageName": "PREPAID",
         "virtualCreditCardDetails": {
            "brand": "MASTERCARD",
            "amount": "235.67",
            "currencyCode": "EUR"
         },
         "flightOfferIds": [
            "1"
         ]
      }
   }
],
```

The cardFriendlyName field can be used to set a customer defined unique name to the generated card.

Example request with cardFriendlyName:

```
        "formOfPayments":[
           {
                     "b2bWallet":
                       {
                         "cardFriendlyName": "MYIXARIS",
                            "flightOfferIds": [
                                "10"
                            ]
                       }
                    }
                ]
```

In the reply the external reference of the generated card, the selected brand and the amount returned is included.

Example response with cardFriendlyName:

```
        "formOfPayments": [
            {
                "b2bWallet": {
                    "cardId": "22xxxxxx",
                    "cardFriendlyName": "MYIXARIS",
                    "virtualCreditCardDetails": {
                        "brand": "MASTERCARD",
                        "amount": "382.56",
                        "currencyCode": "GBP"
                    },
                    "flightOfferIds": [
                        "10"
                    ]
                }
            }
        ],
```

Additional information can be included in the "`formOfPayments`" section of the FlightCreateOrders. This additional information is also included in the regular report generated by the provider and sent to the customer in front of the virtual card number. This is useful information for reconciliation. The name is customizable as well as, the value. The reporting data sent in the request will be used on top of the ones already configured on B2B Wallet or if you are sending a new set of the already configured ones, this data will be overwritten.

We recommend configuring static reporting data into the B2B Wallet configuration and only send in the request the dynamic ones (ex: order number). Please be aware that there is a maximum length for the whole `reportingData` array of 88 signs and for "name" and "value" only alphanumerical signs are allowed! Please have also in mind that there is a maximum number of reporting Data, which was configured during the initial set-up time of your B2B Wallet. The additional info will be reported as:

- PASSENGERNAME: John Doe
- OrderID: 123456
- Trip: Paris

Example:

```
"formOfPayments":[
   {
      "b2bWallet": {
         "reportingData": [
            {
               "name": "PASSENGERNAME",
               "value": "JohnDoe"
            },
            {
               "name": "OrderID",
               "value": "123456"
            },
            {
               "name": "Trip",
               "value": "Paris"
            }
         ],
         "flightOfferIds": [
            "1" ← first recommendation
         ]
      }
   }
],
```

Note:

- In reportingData for value, no spaces and special characters are allowed for example dashes (-) or underscore (_).
- When adding reporting data with AirPlus provider (B2BWallet), specific DBI reporting keys must be used as “name” in order for the information to be successfully processed by AirPlus (else they would be ignored)

In the reply the set information back is returned:

```
"formOfPayments": [
   {
      "b2bWallet": {
         "cardId": "xxxxxxxx",
         "reportingData": [
            {
               "name": "Trip",
               "value": "Paris"
            },
            {
               "name": "PASSENGERNAME",
              "value": "JohnDoe"
            },
            {
               "name": "Trip",
               "value": "Paris"
            }
         ],
         "virtualCreditCardDetails": {
            "brand": "MASTERCARD",
            "amount": "235.67",
            "currencyCode": "EUR"
         },
         "flightOfferIds": [
            "1"
         ]
      }
   }
],
```

In case of a failure due to a Virtual card not accepted by carrier, please review some alternative options below:

- Amadeus B2B Wallet team can force an alternative Mastercard (called fallback card) to be generated for all bookings when necessary.
- The fallback card is usually much more accepted than the default Ixaris Mastercard for some LCC carriers.
- Please contact your Amadeus IT account manager if more information is needed.
- Another virtual card solution can be used.

8.3 EasyPay payment method

The IATA EasyPay payment method is available and can be used as described below:

Example:

```
"formOfPayments": [
            {
                "creditCard": {
                    "brand": "EASYPAY",
                    "number": "5164700100000005",
                    "expiryDate": "2022-01",
                    "flightOfferIds": [
                        "1"
                    ]
                }
            }
        ]
```

Note:

The “number” is to be generated by the customer on IATA side and then pass it to AQC in the create order request as in the example above.

8.4 Baggage Flow

General notes on the baggage flow:

- Ancillary bags are available in Amadeus Quick Connect.
- To retrieve the baggage information (free or chargeable bags as ancillaries), activation is required. Please contact your Amadeus Account Manager for more information.
- Baggage information (free or chargeable bags as ancillaries) can be triggered in the Flight Offers Search, Flight Offers Price and consequently added to the Flight Create Orders.
- Airlines can file free and chargeable baggage information by weight or piece concept.
- “CheckedBags” is used throughout the services to reference free baggage or chargeable bags. It is important to pay close attention to the prefix used before and after “CheckedBags” as this indicates if a bag is free or chargeable:
    - **includedCheckedBags** – Refers to the free bags that are included in the fare and do not incur in extra cost.
    - **chargeableCheckedBags** – Refers to the additional bags that are chargeable, not included in the fare.
- Amadeus Quick-Connect does not support “No Bags” search.

Flight Offer Search:
Several options can be included in the request of the FlightOffersSearch to obtain information on free and chargeable bags:

- **chargeableCheckedBags:**

The chargeable checked bags can be requested in the object “additionalInformation” by using the option “chargeableCheckedBags”

If chargeableCheckedBags = TRUE, the response will include the price of the first additional bag (chargeable) for adult passenger only, when the airline is an AAS (Amadeus Ancillaries Services) member.

If chargeableCheckedBags = FALSE, the response will not include chargeable bag information.

Request example:

```
“searchCriteria”: {
"additionalInformation": {
   "chargeableCheckedBags": true
      }
}
```

- **includedCheckedBagsOnly:**

“includedCheckedBagsOnly” option can be requested in the object “pricingOptions”.

If includedCheckedBagsOnly = TRUE, the response will return recommendations with at least one free baggage only.

****Note:** This option can cause the request to bring zero recommendations in case there are no fares with free baggage included.

If includedCheckedBagsOnly = FALSE, the response will return recommendations with both free baggage (if applicable) and recommendations without free baggage.

Request example:

```
“searchCriteria”: {
  “pricingOptions”: {
    "includedCheckedBagsOnly": true
  }
}
```

****Note:** These settings can also be saved in the administrative console, so it does not have to be included in the query. Please let your Account Manager to set it up.

First Flight Offers Price

In the booking flow with free and chargeable (ancillary) bags, the FlightOffersPrice is called **twice**.

The first flightOffersPrice is called to trigger the catalogue of standardized chargeable bags. The parameter “`include=bags`” is to be included in the POST URL:

```
https://test.travel.api.amadeus.com/v1/shopping/flight-offers/pricing?include=bags
```

The response will include information for both free baggage information and chargeable bags if applicable.

The free baggage allowance (FBA) can be found under the “`fareDetailsBySegment`”. If no FBA is allowed the quantity will be zero.

Response Example:

```
"fareDetailsBySegment": [
   {
      "segmentId": "89",
      "cabin": "ECONOMY",
      "fareBasis": "TWKWGB",
      "brandedFare": "LIGHT2",
      "class": "T",
      "includedCheckedBags": {             🡐 Free Baggage allowance (FBA)
         "quantity": 1    🡐= 1 Piece
      }
   },
…
```

** **Note**: FBA information can be return in piece or weight concept as filed by the carrier**

The chargeable(ancillary) bag information can be found under “`included`” section.

Response Example:

```
"included": {
   "bags": {
      "1": {                           ←  Option 1
         ”quantity": 2,                ←  Two bags
         "name": "CHECKED_BAG",
         "price": {
            "amount": "75.00",         ←  Price for the two bags offered in this option
            "currencyCode": "EUR"
         },
         "bookableByItinerary": true,  ←  if true then the price applies by itinerary
         "segmentIds": [               ←  Price and Information applies to the segments
            "89",                         listed below
            "90"
         ],
         "travelerIds": [
            "1",
            "2"
         ]
      },
      "2": {                           ←  Option 2
         "quantity": 1,                ←  One bag
         "name": "CHECKED_BAG",
         "price": {
            "amount": "30.00”,         ←  Price for one bag
            "currencyCode": "EUR"
         },
         "bookableByItinerary": true,  ←  if true then the price applies by itinerary
         "segmentIds": [               ←  Price and Information applies to the segments
            "89",                         listed below
            "90"
         ]
      },
   }
}
```

The response may include different options with different prices. Customer can select an option for all the passengers in the record or for specific ones. The price shown is for per passenger and per itinerary or the full journey depending on “bookableByItinerary” value.

If bookableByItinerary = FALSE

- The price is given for **full journey** and per traveler
- Must specify the "additionalBags" to **all segments** in the journey, not possible to apply only selected itineraries f.e just inbound or outbound for a round trip.

If bookableByItinerary = TRUE

- The price given is by **per itinerary** and per traveler. So, it means e.g., for a round trip, you have the liberty to book for the outbound itinerary or inbound itinerary only.
- Must specify the "additionalBags" to desired itinerary or itineraries in the journey.

**** Note:** Bags are bookable by itinerary or for the full journey, never by segment. To distinguish the difference between the segments, itineraries and the full journey, please have a look at the following case examples:

- **If an itinerary has no connecting flights, then:**
  segment = itinerary

  Example:
  segment 1- LHRNCE = **itinerary 1**
  segment 2- NCELHR = **itinerary 2**

- **If an itinerary has connecting flights, then:**
  segment 1 and segment 2 = **itinerary 1**
  segment 3 and segment 4 = **itinerary 2**

  Example:
  segment 1- LHRNCE
  segment 2- NCEAMS = **itinerary 1**
  segment 3- NCEAMS
  Segment 4- AMSNCE = **itinerary 2**

- **A full journey with connecting flights:**
  segment 1 and segment 2 = **itinerary 1**
  segment 3 and segment 4 = **itinerary 2**

Example:
segment 1- LHRNCE
segment 2- NCEAMS = **itinerary 1**
segment 3- AMSNCE
Segment 4- NCELHR = **itinerary 2**
So, itinerary 1 and itinerary 2 (LHR-NCE-AMS-NCE-LHR) = Full journey with connecting flights

- **A full journey with NO connecting flights:**

segment 1 = itinerary 1
segment 2 = itinerary 2

Example:

segment 1- LHRNCE = **itinerary 1**
segment 2- NCELHR = **itinerary 2**
So, itinerary 1 and itinerary 2 (LHR-NCE-LHR) = Full Journey

Second Flight Offer Price:

Once the baggage option is selected, the second FlightOffersPrice request should be sent with the chargeable baggage option selected by the passenger. This will reconfirm all fees and fares. AQC does not offer the possibility to book heterogeneous bag options. If an option of 1PC bag is selected for the outbound the same option for 1PC should be selected for the inbound per passenger.

The "grandTotal" should include the total bag fees based on the amount returned in the response of Flight Offer Price. This amount should be manually calculated and updated as it is not done automatically. If an incorrect amount is entered a pricing warning will be displayed. In addition to updating the grandTotal" the baggage information needs to be added per applicable traveler in the "travelerPricings" object. The baggage information needs to be consistent with the information in the response of the flightOffersPrice.

Request Example:

```
"price": {
   "currency": "EUR",
   "total": "263.16",
   "base": "160.00",
   "fees": [
      {
         "amount": "0.00",
         "type": "SUPPLIER"
      },
      {
         "amount": "4.72",
         "type": "TICKETING"
      },
      {
         "amount": "0.00",
         "type": "FORM_OF_PAYMENT"
      }
   ],
   "grandTotal": "327.88",            ←  Price include all fees and checked bags
   "billingCurrency": "EUR",
   "additionalServices": [
      {
         "amount": "60.00",
         "type": "CHECKED_BAGS"
      }
```

```
             ],
}
"travelerPricings": [
          {
            "travelerId": "1",
            "fareOption": "STANDARD",
            "travelerType": "ADULT",
            "price": {
              "currency": "EUR",
              "total": "62.11",
              "base": "16.00",

"taxes": [
                {
                  "amount": "16.33",
                  "code": "FR"
                },
                {
                  "amount": "1.13",
                  "code": "IZ"
                },
                {
                  "amount": "1.50",
                  "code": "O4"
                },
                {
                  "amount": "10.95",
                  "code": "QW"
                },
                {
                  "amount": "4.20",
                  "code": "UI"
                },
                {
                  "amount": "12.00",
                  "code": "YQ"
                }]},
            "fareDetailsBySegment": [
              {
                "segmentId": "5",
                "cabin": "ECONOMY",
                "fareBasis": "VS58BALG",
                "brandedFare": "LIGHT",
                "class": "V",
                "includedCheckedBags": {
                  "quantity": 0
                },
                 "additionalServices": {
                  "chargeableCheckedBags": {
                    "quantity": 1     ←  1 x chargeable baggage added to ADT pax for
                                        segment 5
                  }
                }
              }
            ]
          },
```

In this example we are adding the baggage information in the “additionalServices” object in piece concept based on the response of the flightOffersPrice.

Flight Create Orders:
By reusing the flight offers object from the response of the second Flight Offers Price, no additional data is required to be included in the Flight Create Orders request to book the order with bags. For the confirmation of extra bags booked, travelerPricings/fareDetailsBySegment includes the “additionalServices” information for per passenger in the response of Flight Create Orders.

8.5 Ancillary Seats

For ancillary seats integration please check the “REST API Integration Guide Display Seat Map” available in the” implementation guide” section in Developers Portal.

8.6 One Way Combinable flow (OWC)

One way combinable allows the application to offer and book two one ways in order to target better prices. The traveller can choose two one-way offers instead of a round trip flight that could be more expensive. OWC gives the option to select a combination of Pyton and GDS one-way offers.
At booking time two bookings will be created under a single AQC order. In case of failure of one of the one-way offers the following criteria is followed:

- In a case of a Pyton and GDS one way offers, the GDS one-way offer will be booked first and then secondly the Pyton one-way offer. If the Pyton one-way offer fails AQC cancels the GDS booking and process is stopped. If the GDS one-way offer fails all process is stopped.
- In the case of GDS one-way offers, if one of the offers fails AQC will cancel the successful GDS booking and process is stopped.
- In the case of Pyton one-way offers, if one of the offers fails agency will need to contact the carrier to request the cancellation. Agency could also try to book the failed offer in a new request however keep in mind offers will be retrieved in separate order IDs.

OWC is triggered in the FlightOffersSearch by option “`addOneWayOffers`” set to TRUE.

Example:

```
"searchCriteria": {
   "maxFlightOffers": 50,
   "addOneWayOffers": true,
   "pricingOptions": {
      "fareType": [
         "PUBLISHED",
         "NEGOTIATED"
…
```

In the response of the FlightOffersSearch the one-way recommendations will be returned.

```
{
    "meta": {
        "count": 11,
        "oneWayCombinations": [
            {
                "originDestinationId": "1",← list of outbound offers
                "flightOfferIds": [
                    "6",
                    "7",
                    "8"
                ]
            },
            {
                "originDestinationId": "2", ← list of inbound offers
                "flightOfferIds": [
                    "9",
                    "10",
                    "11"
                ]
            }
        ]
    },
```

The “OneWay” indicator is set to TRUE in each offer.

Example:

```
{
   "type": "flight-offer",
   "id": "6",
   "source": "GDS",
   "instantTicketingRequired": false,
   "nonHomogeneous": false,
   "oneWay": true,                        ← One way recommendation
   "lastTicketingDate": "2019-12-24",
…
```

In both, Flight Offer Price and Flight Create Order, there should be two Flight-Offer objects groups.

Example:

```
One-Way Flight outbound:

{
   "type": "flight-offer",
   "id": "2",
   "source": "GDS",
   "instantTicketingRequired": false,
   "nonHomogeneous": false,
   "oneWay": false,
   "lastTicketingDate": "2019-12-24",
   "numberOfBookableSeats": 9,
   "itineraries": [
      {
         "duration": "PT14H",
         "segments": [
            {
               "departure": {
                  "iataCode": "SIN",
                  "terminal": "3",
                  "at": "2019-12-24T00:15:00"
               },
               "arrival": {
                  "iataCode": "CDG",
                  "terminal": "1",
                  "at": "2019-12-24T07:15:00"
               },
               "carrierCode": "SQ",
               "number": "336",
               "aircraft": {
                  "code": "77W"
               },
               "operating": {
                  "carrierCode": "SQ"
               },
               "id": "1",
               "numberOfStops": 0,
               "blacklistedInEU": false
…

One-Way Flight Inbound:
{
   "type": "flight-offer",
   "id": "12",
   "source": "GDS",
   "instantTicketingRequired": false,
   "nonHomogeneous": false,
   "oneWay": true,
   "lastTicketingDate": "2020-01-02",
   "numberOfBookableSeats": 9,
   "itineraries": [
      {
         "duration": "PT12H45M",
         "segments": [
            {
               "departure": {
                  "iataCode": "CDG",
                  "terminal": "2E",
                  "at": "2020-01-02T20:50:00"
                },
                "arrival": {
                  "iataCode": "SIN",
                  "terminal": "1",
                  "at": "2020-01-03T16:35:00"
                },
                "carrierCode": "AF",
                "number": "256",
                "aircraft": {
                  "code": "77W"
                },
                "operating": {
                  "carrierCode": "AF"
                },
                "id": "3",
                "numberOfStops": 0,
                "blacklistedInEU": false

```

In the Book request, the application must specify the payment method for each way even if the same payment type is used for both recommendations.

Example:

```
"formOfPayments": [
   {
      "other": {
         "method": "CASH",
         "flightOfferIds": [ "12" ]
      }
   },
   {
      "other": {
         "method": "CASH",
         "flightOfferIds": [ "9" ]
      }
   },
…
```

OWC results in two records (1 for Outbound and 1 for Inbound).

Example:

```
"associatedRecords": [
   {
      "reference": "NWPA3G",                    ←  First record
      "creationDate": "2019-11-07T05:31:00.000",
      "originSystemCode": "GDS",
      "flightOfferId": "9"
   },
   {
      "reference": "NWQIQI",                    ←  Second record
      "creationDate": "2019-11-07T05:31:00.000",
      "originSystemCode": "GDS",
      "flightOfferId": "12"
   }
```

8.7 Low Cost Carriers

Low Cost content (LCC) can be booked via three sources:

- **Pyton** – LCC provider that offers low cost carrier content. When booking Pyton content a record is created on Pyton side. Optionally, an Amadeus record with ghost segments can also be created (to be configured/chargeable).

- **LTC** – Content is available through Amadeus. Three carriers are available:
  - U2 - EASYJET
  - TO - TRANSAVIA FRANCE
  - HV - TRANSAVIA AIRLINES

- **Extended Air Choice (EAC)** – Amadeus platform which allows to distribute in an integrated standard Amadeus booking flow Low cost content. Please see ”AQC – Extended Air Choice (EAC) quick card” in the ”Integration Guides” section of the Developers Portal.

Note: For all LCC sources, Virtual Card payment or corporate cards are mandatory. This can be B2B wallet or customer’s own solution. For more information please refer to B2B section in this guide.

General notes on the service:

- For all LCC orders, instant ticketing is done on the provider side.
- Open Jaw and Multi-destination (+2 itineraries) are not supported with Pyton only search.
- When booking a PYTON offer, customer should always check the reference sent back in the create order response:
  - If the booking finally succeeded, nothing more to be done.
  - If a booking is failed, please retry again.
  - If a booking is PENDING, it means that the status of the booking is unknown (DO NOT consider it failed!). The customer can therefore call the Get Order API using the order id provided in the Order Create response in order to check if the status of the booking changes afterwards.
  - If after a couple of minutes, the status remains PENDING then the customer must check its support e-mail address or contact its call center if any booking was really made:
    - If so, the booking is succeeded but is likely to remain PENDING forever on Amadeus side
    - If not, the customer can try to perform the booking again using the exact same Order Create request which was initially sent or select a new offer.
- When booking PYTON offer, Flight Order Management – GET operation is mandatory in case of a failure or pending status.
- When pricing a Pyton offer only the BRAND card is necessary. Pyton does not support BIN option.
- Pyton offers do not return tax details.

- HELD_INFANT passenger type is fully supported by Pyton however SEATED_INFANT is not supported.
- AQC support only STANDARD FareOption with Pyton. A search with another FareOption with the source "PYTON" will return a warning or an error without any Pyton recommendations.
- Pyton offers a conversion (exchange) rate module. When a specific carrier does not support the set currency in the office configuration or the currency sent in the request, the conversion rate module will convert the currency to the one supported by the carrier.   We strongly recommend the use either a virtual payment card or corporate/business payments cards in order to work around any 3D Secure/PSD2 strong authentication that is not currently supported on all contents. Amadeus B2B Wallet is a product which can be used, and it is fully integrated into AQC. For more information please refer to the B2B wallet section in this guide.

8.8 Document Issuance

Amadeus Quick-Connect offers issuance capability for GDS Flight Offers. Two different uses cases are available:

- **Immediate issuance** is available in the FlightCreateOrders. This operation is used to issue tickets and EMDs associate to Ancillary services and it is limited to GDS Flight Offers.

** **Note**: The ticket number will not be returned in the response. Ticket number(s) will be available by retrieving the order (GET).**

The supported operation is available in POST by adding **`issue=true`** to the FlightCreateOrders URL:

```
POST  https://test.travel.api.amadeus.com/v1/booking/flight-orders?issue=true
```

- **Standalone (Post Booking) Issuance** is also available in Amadeus Quick-Connect with the API service “*Flight Order Issue*”. It allows to issue tickets for Full-Service Carriers from an existing orderId. The operation uses the flight order ID that is returned in the response of the FlightCreateOrder.

The supported operation is available in POST using the below endpoint:

```
POST  https://test.travel.api.amadeus.com/v1/booking/flight-orders/{{book_id}}/issuance
```

The ticket number will be available in the response:

```
"tickets": [
     {
       "documentType": "ETICKET",
       "documentNumber": "057-3322491266",
       "documentStatus": "ISSUED",
       "travelerId": "1",
       "segmentIds": [
         "1",
         "2"
       ]
     },
     {
       "documentType": "EMD",
       "documentNumber": "057-1858808477",
       "documentStatus": "ISSUED",
       "travelerId": "1",
       "segmentIds": [
         "1"
       ]
     }
```

- **Direct Ticketing** - Direct Distribution is when an airline allows travel agencies to issue tickets directly on their stock, either because there is no local BSP on that market or the airline is not part of the local BSP. This option requires specific configuration in the Office Profile.

  No specific option/parameter is required as system will perform internal setting of the Office profile configuration.

  The supported operation is available in POST using the below endpoint (same as the standalone).

```
POST  https://test.travel.api.amadeus.com/v1/booking/flight-orders/{{book_id}}/issuance
```

  The ticket number will be available in the response:

```
  "tickets": [
     {
       "documentType": "ETICKET",
       "documentNumber": "172-2400048600",
       "documentStatus": "ISSUED",
       "travelerId": "1",
       "segmentIds": [
         "1"
       ]
     }
   ]
```

8.9 Rules (Mini Rules – Detailed Rules)

Mini-Rules and detailed fares are available in AQC.

Mini Rules provides a short summary of the most important restrictions and fare rules applicable to a specific fare. Mini Rules requires activation, please contact your Amadeus Account Manager for more information. This service is available at Flight Offers Search and Price.

Detailed fare Rules provides the conditions and regulations applicable to a specific fare. This service is only available at Flight Offers Price.

Both Mini Rules and Detailed Fare Rules are available for Full-Service Carriers (GDS) only.

- **Mini Rules -** In the Flight Offers Search request, it is triggered by **“fareRules**” set to TRUE in the “searchCriteria”

Example:

```
"searchCriteria": {
       "additionalInformation": {
           "chargeableCheckedBags": true,
           "fareRules": true
       }
   }
```

The response will include fare rule information in the recommendations:

```
"fareRules": {

              "rules": [
                  {
                      "category": "EXCHANGE",
                      "maxPenaltyAmount": "0.00"
                  },
                  {
                      "category": "REFUND",
                      "notApplicable": true
                  },
                  {
                      "category": "REVALIDATION",
                      "notApplicable": true
                  }
              ]
          }
```

In the Flight Offers Price it is triggered “additionalInformation”, “fareRules” set to TRUE

Example:

```
  "additionalInformation": {
            "fareRules": true
        }
```

The response will include fare rule information in the recommendations:

```
"fareRules": {
                    "rules": [
                        {
                            "category": "REVALIDATION",
                            "notApplicable": true
                        },
                        {
                            "category": "EXCHANGE",
                            "maxPenaltyAmount": "0.00"
                        },
                        {
                            "category": "REFUND",
                            "notApplicable": true
                        }
                    ]
                }
```

- **detailed-fare-rules -** In the Flight Offers Price request, it is triggered by adding parameter “include-detailed-fare-rules in the POST url:

```
https://test.travel.api.amadeus.com/v1/shopping/flight-offers/pricing?include=bags,credit-card-fees,detailed-fare-rules
```

In the Flight Offers Price response the detailed information is returned in the “included” object:

```
 "included": {
        "detailed-fare-rules": {
            "1": {
                "fareBasis": "GS55PBLG",
                "name": "ECONOMY RT UNBUNDLED",
                "fareNotes": {
                    "descriptions": [
                        {
                            "descriptionType": "GENERAL INFORMATION",
                            "text": "RU.RULE APPLICATION FOR ECONOMY RT UNBUN-
DLED FARES KLM -- LIGHT FARES -- HAND BAGGAGE ONLY -ZERO BAGGAGE ALLOWANCE MUST BE DIS-
CLOSED TO CUSTOMER APPLICATION CLASS OF SERVICE THESE FARES APPLY FOR ECONOMY CLASS SER-
VICE. TYPES OF TRANSPORTATION FARES GOVERNED BY THIS RULE CAN BE USED TO CREATE ONE-
WAY/ROUND-TRIP/CIRCLE-TRIP/OPEN
```

```
OM LAST STOPOVER MUST COMMENCE NO EARLIFTER DEPARTURE FROM FARE ORIGIN OR - WHEN DEPAR-
TURE FROM FARE ORIGIN IS SAT THEN TRAVEL FROM LAST STOPOVER MUST COMMENCE NO EAR-
LIER THAN 2 HOURS AFTER DEPARTURE FROM FARE ORIGIN. MX.MAX STAY FOR ECONOMY RT UNBUN-
DLED FARES \nTRAVEL FROM LAST STOPOVER MUST COMMENCE NO LATER THAN 6 MONTHS AFTER DEPAR-
TURE FROM FARE ORIGIN. SR.SALES RESTRICT \nTICKETS MUST BE ISSUED ON/AF-
TER 15APR 20. \nSALE IS RESTRICTED TO SPECIFIC AGENTS. \nSALE IS RESTRICTED TO SPE-
CIFIC AGENTS. \nSALE IS RESTRICTED TO SPECIFIC AGENTS. \nSALE IS RESTRICTED TO SPE-
CIFIC AGENTS. AND - SALE IS RESTRICTED TO SPECIFIC AGENTS. \nSALE IS RESTRICTED TO SPE-
CIFIC AGENTS. AND - SALE IS RESTRICTED TO SPECIFIC AGENTS. AND - SALE IS RE-
STRICTED TO SPECIFIC AGENTS. AND - SALE IS RESTRICTED TO
```

9. JSON Examples

It is important to note that the examples given are only illustrations and are meant to provide the basis for a better understanding on which fields are mandatory for basic operation utilization. It is not a full explanation of every field that can be utilized for the operation, but rather a guideline to its use.

*** Note: The REST/JSON examples are available in the same ZIP file as this implementation guide ***

8.5 Offers, Price and Book, FSC Published Fares - Family

This sample shows return flights booking for multiple PAX type on a Full-Service Carrier with Published fares.

- FlightOffersSearch – Request is for 2 ADTS, 1CHD and 1INF, Round trip, Published Fares:

```
    "travelers": [
           {
               "id": "1",
               "travelerType": "ADULT",
               "fareOptions": [
                   "STANDARD"
               ]
           },
           {
               "id": "2",
               "travelerType": "ADULT",
               "fareOptions": [
                   "STANDARD"
               ]
           },
           {
               "id": "3",
               "travelerType": "CHILD",
               "fareOptions": [
                   "STANDARD"
               ]
           },
           {
               "id": "4",
               "travelerType": "HELD_INFANT",
               "fareOptions": [
                   "STANDARD"
               ],
               "associatedAdultId": "2"
           }
       ],
```

- FlightOffersPrice – Request include Form of payment with credit card brand.
- FlightCreateOrders – Request includes:
    - Travelers contacts – email and Phone number (necessary for the creation of CTCM and CTCE)
    - Document information – Passport
    - Form of payment – Corporate credit card
    - Ticketing agreement ”DELAY_TO_CANCEL” (TKX),
    - Automated Process – (Queue Place) Queue PNR to Q7C2
    - Agency Contact Information

Example in ZIP file: 01 - Booking Flow - FSC - Published - Family

8.6 Offers, Price and Book, FSC- Published Fares – Family - Chargeable Bags

This sample shows return flights booking for multiple PAX type on a Full-Service Carrier, Published fares and chargeable bags.

- FlightOffersSearch – Request includes searchCriteria, additionalInformation option ”chargeableCheckedBags": true (configurable – please see section Amadeus Quick Connect Configuration):

```
    "searchCriteria": {
          "maxFlightOffers": 250,
          "additionalInformation": {
              "chargeableCheckedBags": true
          }
      }
```

- FlightOffersSearch – Response includes in the ”additionalServices” object the price of the 1st chargable bag offered by the carriers when applicable:

```
      "grandTotal": "426.56",
                    "additionalServices": [
                        {
                            "amount": "25.00",
                            "type": "CHECKED_BAGS"
                        }
                    ]
```

In this flow, there are two FlightOffersPrice:

- 1st FlightOffersPrice - Request include parameters for ancillary services (chargeable bags) to trigger the service catalogue:

```
https://test.travel.api.amadeus.com/v2/shopping/flight-offers/pricing?include=bags
```

- 1st FlightOffersPrice - Response includes the chargeable baggage option offered by the carrier for this specific offer in the ”included” object:

```
        "included": {
               "bags": {
                   "1": {    ← Option 1
                       "quantity": 2,
                       "name": "CHECKED_BAG",
                       "price": {
                           "amount": "70.00",
                           "currencyCode": "EUR"
                       },
                       "bookableByItinerary": true,
                       "segmentIds": [
                           "97"
                       ],
                       "travelerIds": [
                           "1",
                           "2"
                       ]
                   },
                   "2": {     ← Option 2
                       "quantity": 1,
                       "name": "CHECKED_BAG",
                       "price": {
                           "amount": "30.00",
                           "currencyCode": "EUR"
                       },
                       "bookableByItinerary": true,
                       "segmentIds": [
                           "61",
                           "62"
                       ],
                       "travelerIds": [
                           "1",
                           "2"
                       ]
                   },
                   "3": {    🡐 Option 3
                       "quantity": 1,
                       "name": "CHECKED_BAG",
                       "price": {
                           "amount": "25.00",
                           "currencyCode": "EUR"
                       },
                       "bookableByItinerary": true,
                       "segmentIds": [
                           "97"
                       ],
                       "travelerIds": [
                           "1",
                           "2"
                       ]
                   },
                   "4": {    🡐 Option 4
                       "quantity": 2,
                       "name": "CHECKED_BAG",
                       "price": {
                           "amount": "75.00",
                           "currencyCode": "EUR"
                       },
                       "bookableByItinerary": true,
                       "segmentIds": [
                           "61",
                           "62"
                       ],
                       "travelerIds": [
                           "1",
                           "2"
                       ]
                   }
               }
```

- 2nd FlightOffersPrice - Request includes the selected baggage option (per passenger) in the "travelerPricings", "fareDetailsBySegment", "additionalServices" object. Information requested is in quantity as the airline chargeable baggage options are offered in quantity (piece) concept:

```
"fareDetailsBySegment": [
                            {
                                "segmentId": "61",
                                "cabin": "ECONOMY",
                                "fareBasis": "GS56PALG",
                                "brandedFare": "LIGHT1",
                                "class": "G",
                                "includedCheckedBags": {
                                    "quantity": 0
                                },
                                "additionalServices": {
                                    "chargeableCheckedBags": {
                                        "quantity": 1
                                    }
                                }
                            },
```

The ”grandTotal” has also been updated to include the fees of the chargeable bags, original ”grandTotal” was "total": "426.56". We add the chargeable "amount": "110.00" to equal "grandTotal": "536.56":

```
"grandTotal": "536.56",
                   "billingCurrency": "EUR",
                   "additionalServices": [
                       {
                           "amount": "110.00",
                           "type": "CHECKED_BAGS"
                       }
                   ]
```

- FlightCreateOrders – Request includes:
    - Travelers contacts – e-mail and phone number (necessary for the creation of CTCM and CTCE)
    - Document information – Passport
    - Form of payment – Corporate credit card
    - Ticketing agreement ”DELAY_TO_CANCEL” (TKX), automated process (Queue_Place).
    - Automated Process – Queue PNR to Q7C2
    - Agency Contact Information

Example in ZIP file: 02 - Booking Flow - FSC - Published - Family - Chargeable Bags

8.7 Offers, Price and Book, FSC Published Fares – Family - Chargeable Bags – Mini-Rules and Detailed Fare Rules - Standalone Issuance

This sample shows return flights booking for multiple PAX type, GDS content, Published fares, chargeable bags -mini-rules and detailed fare rules – CASH form of payment – standalone issuance.

- FlightOffersSearch – Request includes ”fareRules”: ”TRUE”:

```
      "searchCriteria": {
              "additionalInformation": {
                  "chargeableCheckedBags": true,
                  "fareRules": true
              }
          }
```

- 1st Flight Offers Price request includes parameters:

```
https://test.travel.api.amadeus.com/v1/shopping/flight-offers/pricing?include=bags,credit-card-fees,detailed-fare-rules
```

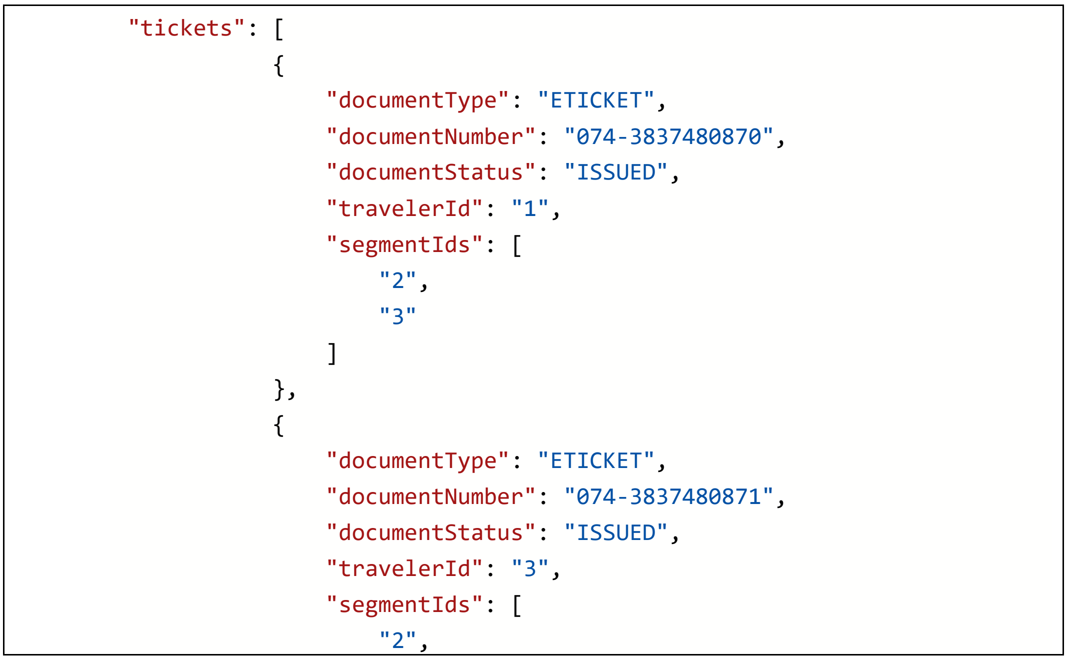
  - Include bags – Triggers service catalogue for chargeable baggage options
  - Credit card fees – Returns credit card fees as filed by the carrier
  - Detailed fare rules – Returns full text of fare conditions and regulations of the selected fare.

- FlightCreateOrders – Request includes:
  - Travelers contacts – e-mail and phone number (necessary for the creation of CTCM and CTCE)
  - Document information – Passport
  - Form of payment – CASH
  - Ticketing agreement ”DELAY_TO_CANCEL” (TKX), automated process (Queue_Place).
  - Automated Process – Queue PNR to Q7C2
  - Agency Contact Information

- Flight Order Issue – Request includes the order ID:

```
https://test.travel.api.amadeus.com/v1/booking/flight-orders/eJzTd9cPNzU1CTQBAAqKAiA%3D/issuance
```

- Flight Order Issue – Response returns ticket number and EMDs (baggage) for all the passengers in the order:



```
        "tickets": [
                  {
                      "documentType": "ETICKET",
                      "documentNumber": "074-3837480870",
                      "documentStatus": "ISSUED",
                      "travelerId": "1",
                      "segmentIds": [
                          "2",
                          "3"
                      ]
                  },
                  {
                      "documentType": "ETICKET",
                      "documentNumber": "074-3837480871",
                      "documentStatus": "ISSUED",
                      "travelerId": "3",
                      "segmentIds": [
                          "2",
                          "3"
                      ]
                  },
                  {
                      "documentType": "ETICKET",
                      "documentNumber": "074-3837480872",
                      "documentStatus": "ISSUED",
                      "travelerId": "4",
                      "segmentIds": [
                          "2",
                          "3"
                      ]
                  },
                  {
                      "documentType": "ETICKET",
                      "documentNumber": "074-3837480873",
                      "documentStatus": "ISSUED",
                      "travelerId": "2",
                      "segmentIds": [
                          "2",
                          "3"
                      ]
                  },
                  {
                      "documentType": "EMD",
                      "documentNumber": "074-1867465964",
                      "documentStatus": "ISSUED",
                      "travelerId": "1",
                      "segmentIds": [
                          "2"
                      ]
                  },
                  {
                      "documentType": "EMD",
                      "documentNumber": "074-1867465965",
                      "documentStatus": "ISSUED",
                      "travelerId": "3",
                      "segmentIds": [
                          "2"
                      ]
                  },
                  {
                      "documentType": "EMD",
                      "documentNumber": "074-1867465966",
                      "documentStatus": "ISSUED",
                      "travelerId": "4",
                      "segmentIds": [
                          "2"
                      ]
                  }
```

```
                  ],
```

Example in ZIP file: 03 - Booking Flow - FSC Published - FF - CH Bags – Mini & Fare Rules - Standalone Issuance

8.8 Offers, Price, Book Pyton Published Fares – Family - Chargeable Bags - B2B Wallet

This sample shows return flights booking for multiple PAX type, Pyton content, published fares, chargeable bags and B2B wallet as form of payment.

- FlightOffersSearch – Request includes ”source”: ”Pyton”:

```
        "sources": [
              "PYTON"
           ],
```

- FlightOffersSearch – Response includes in the ”additionalServices” object the price of the 1st chargeable bag offered by the carriers when applicable:

```
        "grandTotal": "311.15",
                      "additionalServices": [
                          {
                              "amount": "31.48",
                              "type": "CHECKED_BAGS"
                          }
                      ]
```

In this flow, there are two FlightOffersPrice:

- 1st FlightOffersPrice - Request include parameters for ancillary services (chargeable bags) to trigger the service catalogue:

```
https://test.travel.api.amadeus.com/v1/shopping/flight-
offers/pricing?include=bags
```

”payment” information is sent in the request to calculate any fees:

```
  "payments": [
           {
               "brand": "MASTERCARD",
               "flightOfferIds": [
                   "10"
               ]
           }
       ]
```

- 1st FlightOffersPrice - this response generated a warning as supplier fees and a form of payment fee are returned (information comes directly from carrier) since ”payment” information was included:

```
        "warnings": [
              {
                  "code": 0,
                    "title": "PricingOrFareBasisDiscrepancyWarning",
                    "detail": "Actual price and/or fare basis for some passen-
        gers is different from requested ones",
                    "status": 200
                }

Price includes fees due to the “payment information sent in the request:

         "price": {
                            "currency": "GBP",
                            "total": "162.36",
                            "fees": [
                                {
                                    "amount": "140.20",
                                    "type": "SUPPLIER"
                                },
                                {
                                    "amount": "0.00",
                                    "type": "TICKETING"
                                },
                                {
                                    "amount": "4.54",
                                    "type": "FORM_OF_PAYMENT"
```

Response also includes the chargeable baggage option offered by the carrier for this specific offer in the ”included” object. Please note the chargeable bags are offered by quantity and weight concept (information comes directly from carrier):

```
"included": {
        "bags": {
            "1": {
                "quantity": 2,
                "weight": 20,
                "weightUnit": "KG",
                "name": "CHECKED_BAG",
                "price": {
                    "amount": "40.22",
                    "currencyCode": "GBP"
                },
                "bookableByItinerary": false,
                "segmentIds": [
                    "2",
                    "8"
                ],
                "travelerIds": [
                    "1",
                    "2",
                    "3"
                ]
            },
            "2": {
                "quantity": 2,
                "weight": 15,
                "weightUnit": "KG",
                "name": "CHECKED_BAG",
                "price": {
                    "amount": "31.48",
                    "currencyCode": "GBP"
                },
                "bookableByItinerary": false,
                "segmentIds": [
                    "2",
                    "8"
                ],
                "travelerIds": [
                    "1",
                    "2",
                    "3"
                ]
            },
            "3": {
                "quantity": 1,
                "weight": 25,
                "weightUnit": "KG",
                "name": "CHECKED_BAG",
                "price": {
                    "amount": "48.96",
                    "currencyCode": "GBP"
                },
                "bookableByItinerary": false,
                "segmentIds": [
                    "2",
                    "8"
                ],
                "travelerIds": [
                    "1",
                    "2",
                    "3"
                ]
            },
```

The ”grandTotal” has also been updated to include the fees of the chargeable bags, original ”grandTotal” was "total": "489.56". We add the chargeable "amount": "112.12" to equal "grandTotal": "600.70":

```
"grandTotal": "600.70",
                   "billingCurrency": "EUR",
                   "additionalServices": [
                       {
                           "amount": "111.12",
                           "type": "CHECKED_BAGS"
                       }
                   ]
```

- 2nd FlightOffersPrice - Request includes the selected baggage option (per passenger) in the "travelerPricings", "fareDetailsBySegment", "additionalServices" object. Information requested is in quantity and weight as the airline chargeable baggage options are offered in quantity (piece) and weight concept:

```
{
                               "segmentId": "3",
                               "cabin": "ECONOMY",
                               "fareBasis": "WEBECONOMY",
                               "includedCheckedBags": {
                                   "quantity": 0
                               },
                               "additionalServices": {
                                   "chargeableCheckedBags": {
                                       "quantity": 1,
                                       "weight": 15,
                                       "weightUnit": "KG"
                                   }
                               }
                           },
```

- 2nd FlightOffersPrice - Response confirmed all prices and fees as no discrepancy warning is returned:

```
      {
          "data": {
              "type": "flight-offers-pricing",
              "flightOffers": [
                  {
                      "type": "flight-offer",
                      "id": "10",
                      "source": "PYTON",
```

- FlightCreateOrders – Request includes:
    - Travelers contacts – email and Phone number (necessary for the creation of CTCM and CTCE)
    - Document information – Passport
    - Form of payment – B2B wallet
    - ticketing agreement ”DELAY_TO_CANCEL” (TKX),
    - Automated Process – (Queue Place) Queue PNR to Q7C2
    - Agency Contact Information

Example in ZIP file: 04 - Booking Flow - Pyton - Family - CH bags - B2B wallet

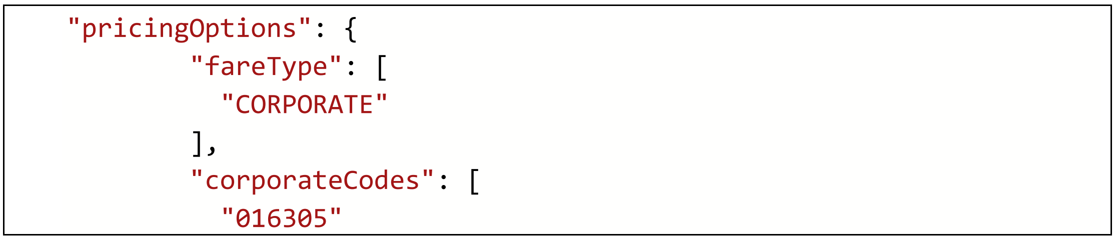
8.9 Offer, Price and Book FSC Inclusive Tour fares

This sample shows return flights booking for two PAX with option ”One way combinable” which allows the application to book two one ways as a return.

- FlightOffersSearch – Request is for 1 ADTS, Corporate fares:

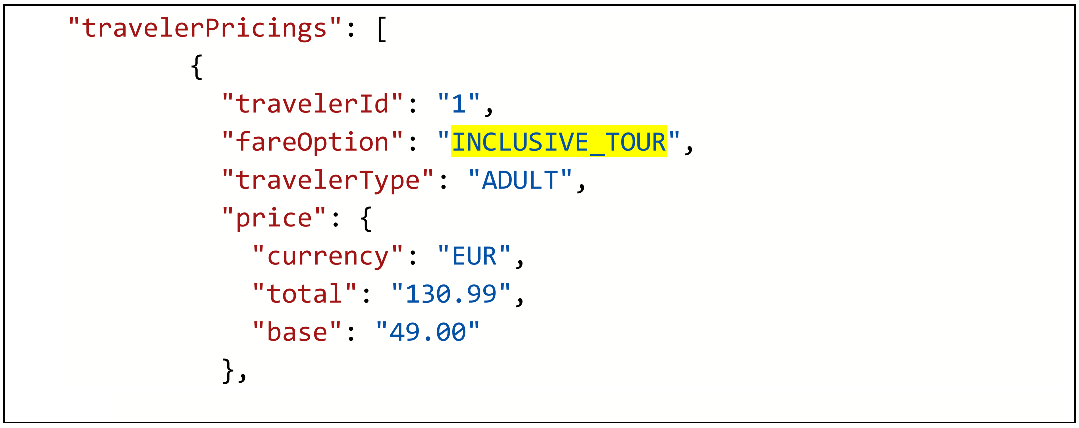
```
      "pricingOptions": {
           "fareType": [
             "CORPORATE"
           ],
           "corporateCodes": [
             "016305"
           ]
         }
```

- FlightOffersSearch – Response the fare type is displayed:



```
    "pricingOptions": {
            "fareType": [
              "CORPORATE"
            ],
            "corporateCodes": [
              "016305"
```

And the fare option:



```
    "travelerPricings": [
            {
              "travelerId": "1",
              "fareOption": "INCLUSIVE_TOUR",
              "travelerType": "ADULT",
              "price": {
                "currency": "EUR",
                "total": "130.99",
                "base": "49.00"
              },
```

- Proceed to follow the booking flow with the Flight Offers Price.
- FlightCreateOrders – Request includes:
    - Travelers contacts – email and Phone number (necessary for the creation of CTCM and CTCE)
    - Company Contact Information (mailing address)

Example in ZIP file: 05 - Booking Flow - FSC Inclusive Tour fares

8.10 Offer, Price and Book with Spanish Residents Discount

This sample shows a proposition to book Spanish Resident discount. It contains return flights booking for one pax type on a full-service carrier with Spanish Resident discount. The flight offers price with Spanish Resident fare option should only be sent once traveler confirms and provides the necessary information for the discount eligibility.

Spanish Resident discounts are only priceable in Spanish Office Ids only.

** Please adhere to local authorities' regulations when booking Spanish Resident Discount**

- In Flight Offers Price request, travelerPricings include the type of discount in the "fareOption" in order to trigger the correct discount:

```
  "travelerPricings": [
                    {
                        "travelerId": "1",
                       "fareOption": "SPANISH_BALEARIC_RESIDENT",
                        "travelerType": "ADULT",
```

- In Flight Create order include the card number applicable to each passenger:

```
   -   "discountEligibility": [
   -                          {
   -                              "subType": "SPANISH_RESIDENT",
   -                              "cityName": "PALMA_DE_MALLORCA",
   -                              "travelerType": "SPANISH_CITIZEN",
   -                              "cardNumber": "12345678Z"
   -                          }
   -                      ]
   -
```

AQC will automatically add FD and FZ elements in the PNR for Spanish Resident PNRs.

Example in ZIP file: 06 - Booking Flow - Spanish Resident

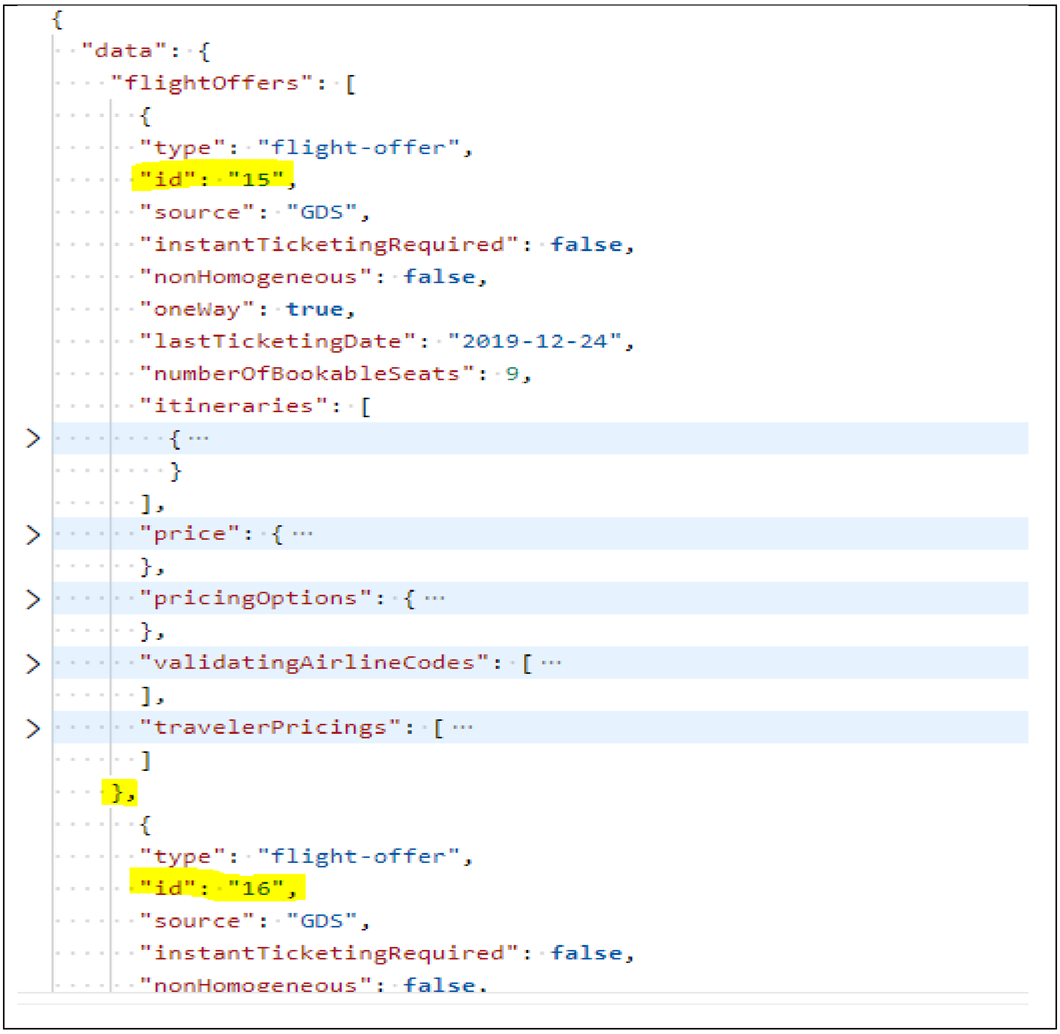
8.11 Offers, Price and book One Way combinable (OWC)

This sample shows return flights booking for two PAX with option ”One way combinable” which allows the application to book two one ways as a return.

- FlightOffersSearch – Request is for 2 ADTS, Published and Negotiated fares:

```
"searchCriteria": {
   "addOneWayOffers": true,
   "pricingOptions": {
     "fareType": [
       "PUBLISHED",
       "NEGOTIATED"
     ]
   }
```

- In both, FlightOffersPrice and FlightCreateOrders, there should be two FlightOffer Groups, one the outbound flights and one for the inbound flights:



```
{
  "data": {
    "flightOffers": [
      {
      "type": "flight-offer",
      "id": "15",
      "source": "GDS",
      "instantTicketingRequired": false,
      "nonHomogeneous": false,
      "oneWay": true,
      "lastTicketingDate": "2019-12-24",
      "numberOfBookableSeats": 9,
      "itineraries": [
        { ···
        }
      ],
      "price": { ···
      },
      "pricingOptions": { ···
      },
      "validatingAirlineCodes": [ ···
      ],
      "travelerPricings": [ ···
      ]
    },
      {
      "type": "flight-offer",
      "id": "16",
      "source": "GDS",
      "instantTicketingRequired": false,
      "nonHomogeneous": false,
```

- In the FlightCreateOrders request, the application must specify the payment method for each way even if the same payment type is used for both recommendations.:

```
  "formOfPayments": [
     {
       "other": {
         "method": "CASH",
         "flightOfferIds": [
           "12"
         ]
       }
     },
     {
       "other": {
         "method": "CASH",
         "flightOfferIds": [
           "9"
         ]
       }
     }
   ]
```

- OWC results in two records (1 for Outbound and 1 for Inbound).

```
{
  "data": {
    "type": "flight-order",
    "id": "eJzTd9cPdjaNDDbRBzHMgkMsASsfBKI=",
    "queuingOfficeId": "1ONXXXXXX",
    "associatedRecords": [
      {
        "reference": "SC5YS4",
        "creationDate": "2019-12-03T15:08:00.000",
        "originSystemCode": "GDS",
        "flightOfferId": "12"
      },
      {
        "reference": "SC6ST9",
        "creationDate": "2019-12-03T15:08:00.000",
        "originSystemCode": "GDS",
        "flightOfferId": "9"
      }
```

Example in ZIP file: 07 - Booking flow - One Way combinable (OWC)

8.12 Offers, Price, Book, Direct ticketing

This sample shows return flights booking for one PAX on a Full-Service Carrier with Published fares and Direct Ticketing (market specific).

- FlightOffersSearch – Request is for 2 ADTS, 1CHD and 1INF, Round trip, Published Fares.
- FlightOffersPrice – Request include Form of payment with credit card brand.
- FlightCreateOrders – Request includes:
    - Travelers contacts – Phone number (necessary for the creation of CTCM)
    - Document information – Passport
    - Form of payment – Cash
    - Ticketing agreement ”DELAY_TO_CANCEL” (TKXL),
    - Agency Contact Information

- Flight Order Issue – The request for Direct Ticketing is the same as the standard ticketing, no additional information is required in the Flight Order Issue. This feature is controlled by the Office ID settings. For more information please contact your Amadeus Account Manager.
- Flight Order Issue – Response returns ticket number and EMDs (baggage) for all the passengers in the order:

```
        "tickets": [
          {
            "documentType": "ETICKET",
            "documentNumber": "172-2400048600",
            "documentStatus": "ISSUED",
            "travelerId": "1",
            "segmentIds": [
              "1"
            ]
          }
```

Example in ZIP file: 08 - Booking flow - Direct Ticketing

8.13 Branded Fares Upsell

This sample returns the branded fares in the pricing call for a family on a Full-Service Carrier with Published fares.

- Flight Offers Search – Request is for 2 ADTS, 1CHD and 1INF, Round trip, Published Fares.
- Branded Fares Upsell – Request is sent with the offer selected from the Flight Offers Search.
- Branded Fares Upsell – Response. The first offer is the original priced offer and the other offers are the upsells offers. The branded fare information is returned in the price offer under the fareDetailsBySegment:

```
"fareDetailsBySegment": [
                        {
                            "segmentId": "10",
                            "cabin": "ECONOMY",
                            "fareBasis": "GS55PBST",
                            "brandedFare": "STANDARD",
                            "class": "G",
                            "includedCheckedBags": {
                                "quantity": 1
                            },
```

Response also returns different specificities in the ”amenities” object:

```
 "amenities": [
                                {
                                    "code": "0C3",
                                    "descrip-
tion": "CHECKED BAG 1PC OF 23KG 158CM",
                                    "isChargeable": false,
                                    "amenityType": "BAGGAGE"
                                },
                                {
                                    "code": "0MR",
                                    "description": "CABIN BAG-
GAGE 12KG 1PC 115CM",
                                    "isChargeable": false,
                                    "amenityType": "BAGGAGE"
                                },
                                {
                                    "code": "0AT",
                                    "description": "SNACK",
                                    "isChargeable": false,
                                    "amenityType": "MEAL"
                                },
                                {
                                    "code": "0AX",
                                    "description": "BEVERAGE",
                                    "isChargeable": false,
                                    "amenityType": "MEAL"
                                },
```

- 1st Flight Offers Price – Request is optional. Would be required in case more options for chargeable bags are presented to traveler.

- 2nd Flight Offers Price – Request to reconfirm total price (as per baggage flow)

- FlightCreateOrders – Request includes:
    - Chargeable bags for each applicable passenger
    - Travelers contacts – email and Phone number (necessary for the creation of CTCM and CTCE)
    - Document information – Passport
    - Remark
    - Form of payment – Corporate credit card
    - Ticketing agreement ”DELAY_TO_CANCEL” (TKXL),
    - Automated Process – (Queue Place) Queue PNR to Q7C2

Example in ZIP file: 09 - Booking Flow - Branded Fares Upsell

8.14 Flight Order Management

- **GET**: Retrieve an Order ID

- **PATCH**: Add the following to an existing order ID
  - Add Passport information to passenger one only
  - Add a phone number to passenger one only
  - Add an email address to passenger one only
  - Place PNR on Queue 2 category 0
  - Add a general remark

- **DELETE**: Cancel an existing Order ID (not issued)
  - Request – Include order ID in the DELETE URL:

```
DELETE: https://test.travel.api.amadeus.com/v1/booking/flight-orders/eJzTd9cPCwt18TEBAAt%2FAls%3D
```

  - Response: No body is returned but status 204 NO CONTENT

```
204 NO CONTENT

The server successfully processed the request but is not returning any content.
```

Example in ZIP file: 10 - Flight Order Management

8.15 Direct Pricing from Cached Flight Information

This sample shows Direct Pricing from Cache Flight Information for (1 ADT, 1 CHD, 1 INF)

- FlightOffersPrice – Request contains minimum pricing objects with specific flight information coming from Cache Information (not Flight Offers Search).

Example in ZIP file: 11 - Direct Pricing from Cache Flight Info

8.16 GetFlightOrderByRecordLocator

This sample shows the retrieval (GET) of a PNR with a passive segment (PK. (Retrieve) by Record Locator. This API can be used for PNRs created outside AQC. An Order ID will be returned

- GetFlightOrderByRecordLocator: request includes GDS reference (PNR):

```
https://test.travel.api.amadeus.com/v1/booking/flight-orders/by-reference?reference=WRWVCJ&originSystemCode=GDS
```

- GetFlightOrderByRecordLocator: response will include an Order ID which is needed for any post booking actions. In this example, the retrieved order is for a passive segment (PK) as seen in the ”segmentType": "PASSIVE" (for more information on segment type please check the swagger spec):

```
{
    "meta": {
        "count": 1,
        "links": {
            "self": "https://test.travel.api.amadeus.com/v1/booking/flight-orders/by-reference?reference=WRWVCJ&originSystemCode=GDS"
        }
    },
    "data": [
        {
            "type": "flight-order",
            "id": "eJzTd9cPDwoPc%2FYCAAwDAok%3D",
            "queuingOfficeId": "NCE1A01AP",
            "associatedRecords": [
                {
                    "reference": "RECLOC",
                    "originSystemCode": "AA",
                    "flightOfferId": "1"
                },
                {
                    "reference": "WRWVCJ",
                    "creationDate": "2021-08-31T16:58:00",
                    "originSystemCode": "GDS",
                    "flightOfferId": "1"
                }
            ],
            "flightOffers": [
                {
```

```
"type": "flight-offer",
"id": "1",
"source": "GDS",
"nonHomogeneous": false,
"itineraries": [
    {
        "segments": [
            {
                "departure": {
                    "iataCode": "LHR",
                    "terminal": "3",
                    "at": "2021-11-24T10:35:00"
                },
                "arrival": {
                    "iataCode": "MIA",
                    "at": "2021-11-24T15:35:00"
                },
                "carrierCode": "AA",
                "number": "39",
                "aircraft": {
                    "code": "772"
                },
                "operating": {},
                "bookingStatus": "CONFIRMED",
                "segmentType": "PASSIVE",
                "isFlown": false,
                "id": "4",
                "numberOfStops": 0
            }
        ]
```

Example in ZIP file: 12 - GetFlightOrderByRecordLocator - Passive segment

10. Certification cases

This section contains the Test cases (adapt according to your specific use cases) that must be validated for your application to be successfully certified. If there is an error in any of these areas, your application will fail the certification. During the certification, your web service implementation consultant will validate all the applicable test cases.

Test Case	Implementation Type/Content Source	Scenario
1	Booking flow – Main FSC	Search, Price and Book – FSC content with Published Fares for a family PNR (2ADTs, 1CHD, 1INF)
2	Booking flow – Main Pyton	Search, Price and Book –Pyton content with Published Fares for a family PNR (2ADTs, 1CHD, 1INF)
	Booking flow – Negotiated/Corporate Fares	Search, Price and Book – FSC with Negotiated/Corporate Fares for a family PNR (2ADTs, 1CHD, 1INF)
3	Booking flow – B2B wallet	Search, Price and Book – Pyton content with Published Fares for a family PNR (2ADTs, 1CHD, 1INF) using B2B wallet as form of payment.
4	Booking flow – Ancillary Bags (chargeable)	Search, Price, Price and Book – FSC with Published Fares for a family PNR (2ADTs, 1CHD, 1INF) including chargeable bags for all qualifying passengers.
5	Booking flow – Ancillary Seats (chargeable)	Search, Price and Book – FSC with Published Fares for a family PNR (2ADTs, 1CHD, 1INF) including chargeable seats for all qualifying passengers.
6	Booking Flow - Issuance	Search, Price and Book – FSC with Published Fares for a family PNR (2ADTs, 1CHD, 1INF) with immediate issuance
7	Booking flow – Standalone Issuance	Search, Price, Book, GET and Issue – FSC with Published Fares for a family PNR (2ADTs, 1CHD, 1INF)
8	Flight Order Management	Search, Price and Book – GET, CANCEL and PATCH an order id.
9	Branded Fares Upsell	Search, Price and Book – FSC with Published Fares for a family PNR (2ADTs, 1CHD, 1INF) with Upsell.
10	Flight Availabilities Search	Availabilities, Price and Book – FSC with Published Fares for a family PNR (2ADTs, 1CHD, 1INF)

11. Certification Checklist

This section contains the list of items that need to be validated to be successfully certified. If there is an Error in any of these areas, then your application will fail the certification. Please ensure all these items are correct by the time your API Consultant begins to certify your application.

GENERAL		
Dedicated Access	Development and Certification of the application must only be performed against your Dedicated Access. Development or Certification cannot be performed against the Fast Access.	Critical
Ama-Client-Ref	An Ama-Client-Ref must be generated by the customer for each end user session and be passed in the header of all requests of a given booking flow. This allows for easier issue analysis by the support team. The Ama-Client-Ref the Ama-Client-Ref should be used for one unique client session (one booking).	Critical
Ama-Request-ID	An Ama-Request-ID is generated for each API sent. Ensure this information is collected as both Ama-Request-ID and Ama-Request-Ref will be needed to log cases in the Amadeus Service Hub (ASH).	Critical

FLIGHT OFFERS SEARCH		
searchCriteria	It is not necessary to include search criteria that has been set in the admin configuration unless the value will be overwritten.	Warning
Travelers	The application must correctly handle passenger types in the Flight Offers Search request.	Critical
fareOptions	To ensure the agent is only provided with fares valid for their agency, ensure the correct fare types are entered in the request.	Warning
FLIGHT OFFERS PRICE		
flightOffers	When selecting a FlightOffer ensure all information is entered correctly. It is recommended to reuse the object of the selected recommendation to avoid discrepancies.	Critical
pricingOptions	To ensure correct pricing, the same pricing options used in the FlightOffersSearch should be used in the subsequent request of the FlightOffersPrice.	Critical
Travelers	The application must correctly handle passenger types in the FlightOffersPrice request. To ensure correct pricing, the same TravelerType that was used in the FlightOffersSearch should be used in the subsequent request of the FlightOffersPrice.	Critical

payments	For LTC, EAC and Pyton payment information is mandatory. Minimum information that can be entered is the credit card brand.	Critical
Payments – B2B Wallet	When using B2B Wallet or any other Virtual Card solution it is highly recommended to use corresponding payment card brands (MASTERCARD_IXARIS, VISA_IXARIS, MASTERCARD_AIRPLUS, UATP_AIRPLUS) to maximize price coherency between price & book calls.	Warning
additionalServices (Bags)	In the booking flow with bags, the FlightOffersPrice is called twice. The first FlightOffersPrice is called to trigger the catalogue of standardized chargeable bags. Once option is selected, the second FlightOffersPrice request should be sent to reconfirm all fees and fares.	Critical
additionalServices (Bags)	The option INCLUEDE BAGS needs to be included in the URL of the first FlightOffersPrice to trigger the service catalogue. However, this option does not need to be included in the second FlightOffersPrice.	Critical

additionalServices (Bags)	The “GrandTotal” should include the total bag fees based on the response of first price information. This amount should be manually calculated (added to the grandTotal) as it is not done automatically.	Critical
additionalServices (Seats)	The “GrandTotal” should include the total seat fees based on the response of Seatmap display. This amount should be manually calculated (added to the grandTotal) as it is not done automatically.	Critical
additionalServices (Seats and bags)	The seats and bags should be added per traveler in the fareDetailsBySegment.	Critical
addOneWayOffers	When pricing one-way combinable recommendations two separate flightOffers should be entered, one for the outbound and one for the inbound.	Critical
addOneWayOffers	When pricing one-way combinable recommendations payment information should be entered for each flightOffer.	Critical
FLIGHT CREATE ORDER		
pricingOptions	To ensure correct pricing during the create order, the same pricing options used in the FlightOffersSearch and FlightOffersPrice should be used in the request.	Critical

Travelers	The application must correctly handle passenger types in the FlightCreateOrders request. To ensure correct pricing, the same TravelerType that was used in the FlightOffersSearch and FlightOffersPrice should be used in the request.	Critical
bookingRequirements	The booking requirements from the FlightOffersPrice response (for instance dateOfBirth needed or document(s) needed) should be respected in the booking input. If not followed issuance may fail due to the missing information.	Critical
payments	For LTC and Pyton payment detailed credit card payment information is needed.	Critical
addOneWayOffers	When creating an order with One-way combinable recommendations two separate flightOffers should be entered, one for the outbound and one for the inbound.	Critical
addOneWayOffers	When creating an offer with One-way combinable recommendations two separate payments should be included. They can be the same for both flightOffers.	Critical
FLIGHT ORDER MANAGEMENT		
GetPNR	The flightOrderId should be URL-encode before putting it in the endpoints. IDs can sometimes contain “/” so it is recommended to URL-encoded	Warning

	(replace the "/" with %2F%) before putting them in an endpoint.	

12. Common errors or warnings

Error Code	Error Title	Error Detail	Explanation	Action
0	ContactInfoSsrNotAddedToBookingWarning	Unable to book the SSR Remark	AQC can automatically create the SR CTCE and CTCM if configured in the admin side. This warning can happen when the needed information is not present to create this information. This error can also be returned when booking Light Ticketing carriers as we cannot send the CTCE and CTCM.	Please ensure the email or mobile number is included for all travelers.
0	IncompleteSearchWarning	Search with the source PYTON could not be done completely	This warning is returned when an option used in the request cannot be executed with PYTON. Example: - IIT fares cannot be executed for Pyton content. - Spanish resident Discount is not available with Pyton content. - SEATED_INFANT is not supported with PYTON content.	Please remove options that are not executable by Pyton.
0	IssuanceFailureBookingWarning	Ticket issuance failed for the following flight offers: x	This error may occur for example on airline Uzbekistan Airways (HY) because of a missing SSRFOID element.	Unfortunately, currently only the SSR DOCS are managed by AQC via the element "document" in the request. It's not possible to add SSR FOID. It is needed to add manually via SellCo or other the SSR FOID before the issuance.

Error Code	Error Title	Error Detail	Explanation	Action
0	OptionDiscardedSearchWarning	Some options have been discarded during the search: RETURN_TO_DEPARTURE_AIRPORT	Some ”flightFilters” are not supported by Pyton content. Therefore, have been ignored in the search.	Please remove options that are not executable by Pyton.
0	OptionDiscardedSearchWarning	At complex itinerary search: Some options have been discarded during the search: MULTIPLE_RECORDS, TICKETABILITY_PRECHECK	This warning informs that due to the complex itinerary search some option (MP option) are disable: • The split PNR (MULTIPLE_RECORDS - NPS option in MP) • The TICKETABILITY_PRECHECK (TAC option in MP)	
0	PRICE DISCREPANCY: Current price exceeds tolerance”		This is a common Pyton error mostly in the test environment. They detect a change of price at the very final step of the booking flow.	Booking flow will need to start over.
0	UnconfirmedBookingWarning	Status of the Booking Unknown - Payment Data sent - Please check with the Airline for confirmation	This error happens when there is a problem with a Pyton booking.	Please see ”Low Cost Carrier” section for the handling of unconfirmed orders with Pyton.
0	UnusedSearchParameterWarning	Unused parameters with the source PYTON - Parameter: DATE_TIME_RANGE	This error can happen when requesting PYTON content with a “date” and “time” option.	Do not include a time option when requesting PYTON content.
141	SYSTEM ERROR HAS OCCURRED	Unknown error	For example, it can happen during pricing, when backend can not process the request.	For example. check the flight segments if they are correct.
477	INVALID FORMAT	JSON cannot be parsed	This error can happen when the request is not properly formatted.	Please check the request.
477	INVALID FORMAT	Number of characters of 'firstName' and 'lastName' must be less than 20 characters	The mailing address name only supports 20 characters	
701	WRONG AUTHENTICATION CREDENTIALS		This error occurs when shooting a specific service using the generated	The configuration of the application needs to be checked. Please open an Incident for investigation.

Error Code	Error Title	Error Detail	Explanation	Action
4926	"INVALID DATA RECEIVED	flightOfferId 1 not existing	This error can happen when the FlightOfferIDs number is not matching in all objects of the request.	Ensure all flight Offer Ids are matching correctly within the entire request
4926	INVALID DATA RECEIVED	type value is not in the allowed enumeration	This error can happen when the additional information does not match the passenger reference.	Ensure all passenger reference Ids are used correctly within the entire request.
4926	INVALID DATA RECEIVED	Can't cancel some elements of this flightOrder	This error is returned when trying to cancel an OrderId. This is because the PNR contains OPW (Option Warning Element – notifies until what day and time the ticket must be issued) and/or OPC (Option Cancellation Element – indicates the day and time when the booking will be automatically canceled if no ticket is issued), it is not possible to delete this PNR with AQC.	In these cases, the PNR must be canceled offline.
4926	INVALID DATA RECEIVED	The form of payment is inconsistent", "source":{"pointer":"/data/formOfPayments[0] /creditCard"	This error can happen when the credit card holder name contains a special character.	Check if the holder name contains a special character. The holder name must be the same as written on the card.
4926	INVALID DATA RECEIVED",	"No fare applicable"	This error can happen when the selected cabin is fully booked at time of price.	Select a new recommendation.
4926	INVALID DATA RECEIVED	associatedAdultId only necessary for HELD_INFANT	HELD_INFANT is the only PTC that requires association to an ADULT.	Please verify your request and ensure the correct passenger type is being associated to and ADULT.
4926	INVALID DATA RECEIVED	Can't update already issued flightOrder	This error can happen when trying to modify an order id that has been issued.	Patch is not available on issued bookings.
4926	INVALID DATA RECEIVED	There are not enough adults to accompany infants and children	Only one Infant per adult is allowed (HELD or SEATED)	Ensure there are enough adults for the specified INFANTS (HELD or SEATED)

Error Code	Error Title	Error Detail	Explanation	Action
9112	Ticketing – TKT	Ticket issuance failed for record XXXXXX (creationDateTime=2020-0X-XX)"	This error can happen for Light Ticketing carriers. At time of ticketing the credit card used is rejected by the carrier (LCCs or Pyton)	Please try a different card or contact the carrier for more details.
32171	MANDATORY DATA MISSING	No Corporate codes defined	This error happens when the FlightOffersSearch request fareType CORPORATE and do not include a corporate number.	Please include a corporate number(s). Note: Corporate numbers can be configured in the admin.
32171	MANDATORY DATA MISSING	Type of resource needed	This error can happen when the resource type in the query is missing, e.g. "type": "flight-order"	Please ensure query is formatted correctly.
32171	MANDATORY DATA MISSING		This error happens when mandatory information is missing in the request	If “Documents” are included in the request, ensure all mandatory items are entered.

32171	MANDATORY DATA MISSING	MAILING_ADDRESS_REQUIRED	MAILING_ADDRESS_REQUIRED means, that an address in contacts has to be added.	Example:
				"contacts": [
				{
				"address": {
				"countryCode": "FR",
				"cityName": "Paris",
				"postalCode": "06902",
				"lines": [
				"99 Rue de Paris"
				]
				}
				}
				]

34651	SEGMENT SELL ERROR		This error can happen when the seat being booked is the as seat and has been booked by someone else.	Select a new recommendation.

Error Code	Error Title	Error Detail	Explanation	Action
37200	PRICE DISCREPANCY	Current grandTotal price (111.00) is different from request one (100.01)	This error is returned in the FlightOffersPrice step, the price per pax for ADT & CHD is the same for both inbound & outbound whereas in the search response, the amount seems more realistic. This is because Pyton may not get the fare breakdown at price step, in this case the new total amount is divided by the number of pax (ADT+CHD) & number of itineraries which means the same amount will show per pax for each segment.	This occurs mostly in the test environment. If this error happens in production, please report it for investigation.
38189	Internal error	An internal error occurred, please contact your administrator	There is some error at internal applications while processing the request. It can also be that the backend did not send the error response.	Please open an incident for this issue via Amadeus Service Hub for analyzing it.
38190	Invalid access token	The access token provided in the Authorization header is invalid	The requests API is not available on the application.	Amadeus has to check the application configuration.

13. Glossary

Terminology	Meaning	Comment
Dedicated Access	Unique Access with specific settings per customer	This access will be created by the API Consultant in charge of the project.
API Key	This is a set of characters used to obtain a token	This can be found in the Developers Portal in the application.
API Secret	This is a set of characters used to obtain a token	This can be found in the Developers Portal in the application.
ama-client-ref	Set of alpha numeric characters that should be generated and used in a booking flow	This reference is used internally for troubleshooting purposes
Pyton	Source for low Cost Carriers	Requires activation - Account Manager/Delivery Manager will request prior to start of project.
addOneWayOffers	Option to get One-way recommendations	FlightOffersSearch
instantTicketingRequired	Recommendation requires immediate tickets	Pyton and Light Ticketing will be ticketed right away.
CHECKED_BAGS	Is used to reference any type of bags that will be checked in at the counter.	This can be free baggage that are included in the fare or any chargeable bag.
includedCheckedBagsOnly	Refers to Free baggage allowance	In the FlightOffersSearch query – If set to TRUE, the response will only include recommendations with free baggage allowance.
includedCheckedBags	Free Baggage Allowance	In the FlightOffersPrice – If set to TRUE, recommendation includes free baggage allowance.
chargeableCheckedBag	Refers to Chargeable baggage	FlightOfferPrice – If sent as TRUE will contain recommendations with first bag price

bookableByItinerary	Specify if the service is bookable by itinerary or mandatory for all itineraries	FlightOfferPrice – If set to TRUE, the chargeable bag can be booked by itinerary and price is by itinerary. If set to FALSE, the chargeable bag must be booked for the whole journey and price is for the whole journey
B2B Wallet	Amadeus solution to generate virtual cards/	
“id”	Order ID number	FlightCreateOrder response – Is the ID number created.
“reference”	Flight Order Reference	Record Locator

14. References

Amadeus for Developers Portal: https://developers.amadeus.com/