ONE Order Sandbox Testing Using Postman

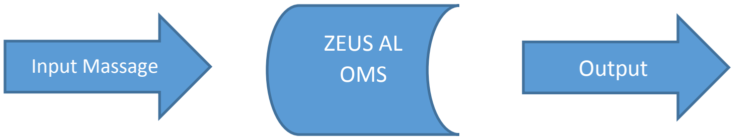
How-to Guide

V1.0

Executive Summary:

i. *Purpose of document:* the purpose of this guide is to provide a step-by-step approach to using the ONE Order Sandbox: ZEUS Airlines.

ii. *Overview:* The ONE Order team along with JR Technology created a ONE Order dedicated Sandbox for the purpose of testing and demonstrating the ONE Order messages in an NDC shopping flow. ZEUS Airlines represents a fictitious airline whereby the sandbox is ZEUS Airlines Order Management System (OMS) containing the airline's complete inventory and assets as described in the first part of a dedicated ZEUS Airlines Brochure. To test the NDC and ONE Order chain of messages, the OMS interacts with the various messages it receives and from that point onwards, testers, developers and users will have the opportunity to test multiple use cases and scenarios.



iii. *High level How to test:* Along with the ZEUS Airline brochure, which describes the airline's inventory, a set of sample ONE Order messages (aka. collection) are uploaded to an online repository called Github. The purpose of those samples, which are prepopulated with key input data, is to provide the users with an initial feel of the interaction with the sandbox in a controlled manner and predictable behavior/output. To start working on the sandbox, you can follow the below high-level steps that will be discussed in detail in the below sections:

1- Download the ZEUS Airlines Brochure
(http://ndc.developer.iata.org/docs/ONE_Order_Sandbox_Brochure_ZEUS_Airlines_Z9.pdf)

2- Access github to obtain the latest NDC V17.1 and ONE Order Alpha V2.1 samples
(https://github.com/iata-ndc/ONEOrder)

3- Register onto Mashery to get your personal API Key
(http://iata.mashery.com/member/register)

4- Download Postman (A Google Application) onto your Chrome browser
(https://www.getpostman.com/)

5- Set up Postman:

   a. Set Globals

    b. Import collection (`ONE Order Samples (NDC 17.1 and OO 2.1) 21 04 2017.postman_collection v2`)

6- Start sending messages to the ONE Order Sandbox, either predefined samples or introducing your own modifications.

Step One: ZEUS Airlines Brochure

A dedicated brochure describing the Airline's inventory (destinations, fares, services, classes, Miscellaneous…) has been created. The purpose of this document is to enable the user to experiment with the messages by manipulating the input data such as: origin/destination data.

You can download the ZEUS Airlines Brochure by referring to this link

`http://ndc.developer.iata.org/docs/ONE_Order_Sandbox_Brochure_ZEUS_Airlines_Z9.pdf`

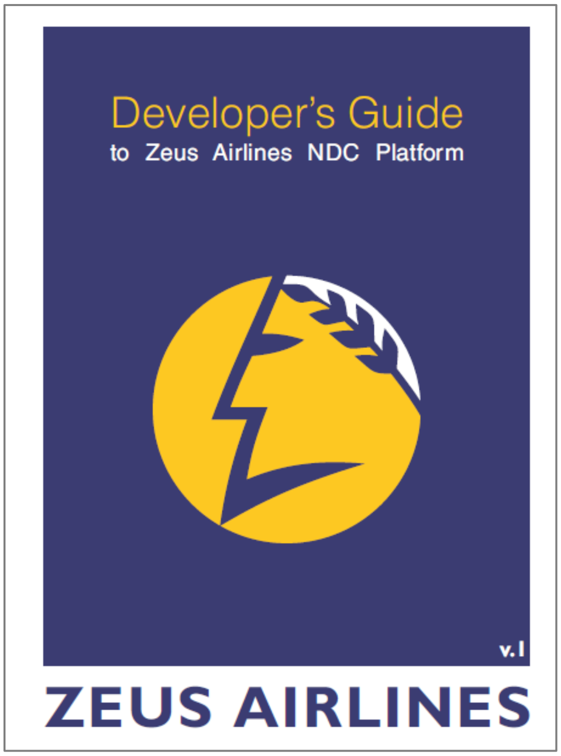
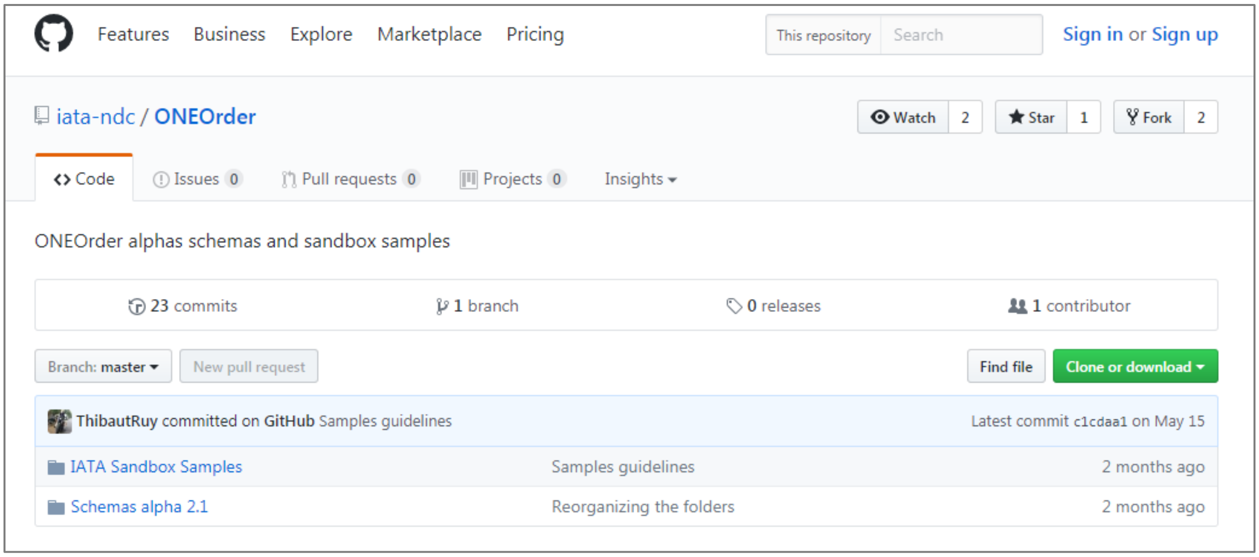


Figure 1 A Snapshot of the front page of ZEUS Airlines Brochure

Step Two: github Repository

A shared Github repository enables you to gain access to the latest NDC and ONE Order samples. You can gain access to the samples by referring to this link https://github.com/iata-ndc/ONEOrder



ONEOrder alphas schemas and sandbox samples

ThibautRuy committed on GitHub Samples guidelines — Latest commit c1cdaa1 on May 15

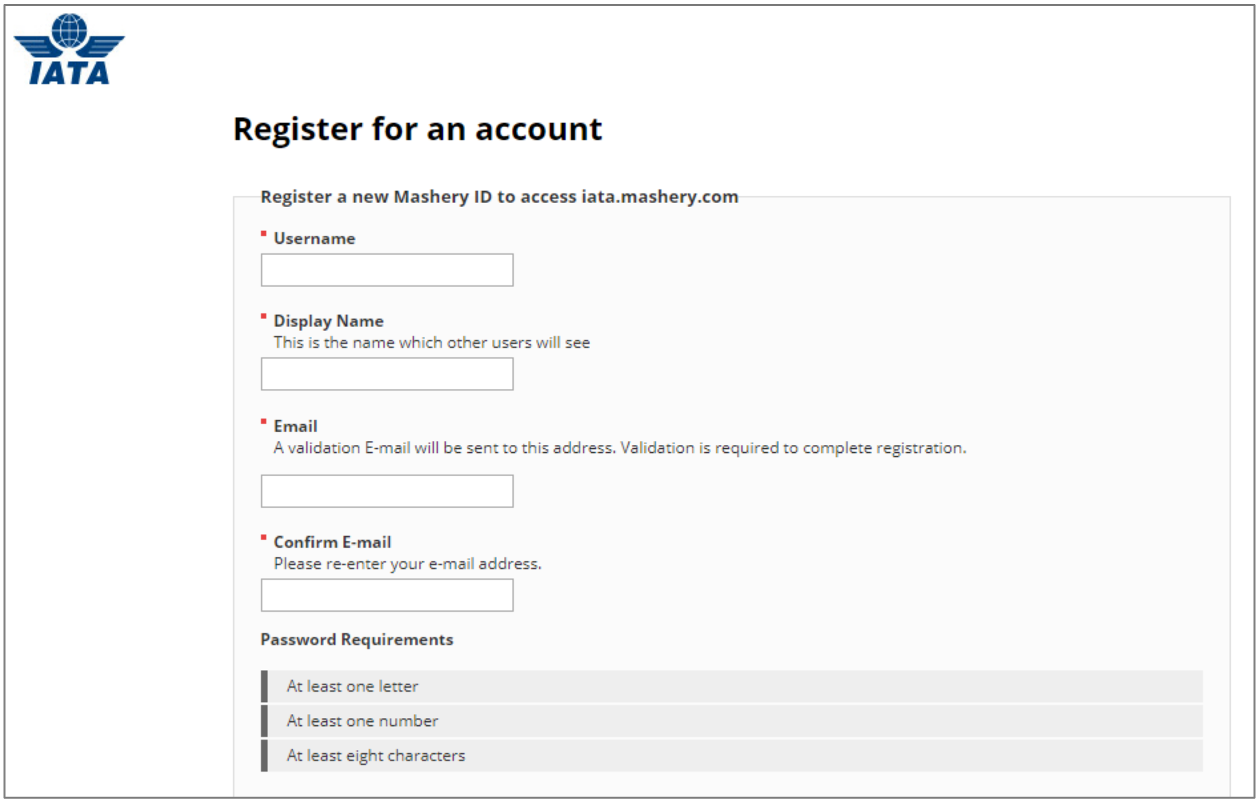
IATA Sandbox Samples	Samples guidelines	2 months ago
Schemas alpha 2.1	Reorganizing the folders	2 months ago

Figure 2 github repository for ONE Order Samples

Step Three: Registration and API Key

In order to start off with the ONE Order sandbox, you'll need to register on the IATA Developer Portal (powered by Mashery). To start this process, please follow the below link: (If you are already registered, simply log in or follow the normal forgot password process)

http://iata.mashery.com/member/register



Register for an account

Register a new Mashery ID to access iata.mashery.com

* Username

* Display Name
This is the name which other users will see

* Email
A validation E-mail will be sent to this address. Validation is required to complete registration.

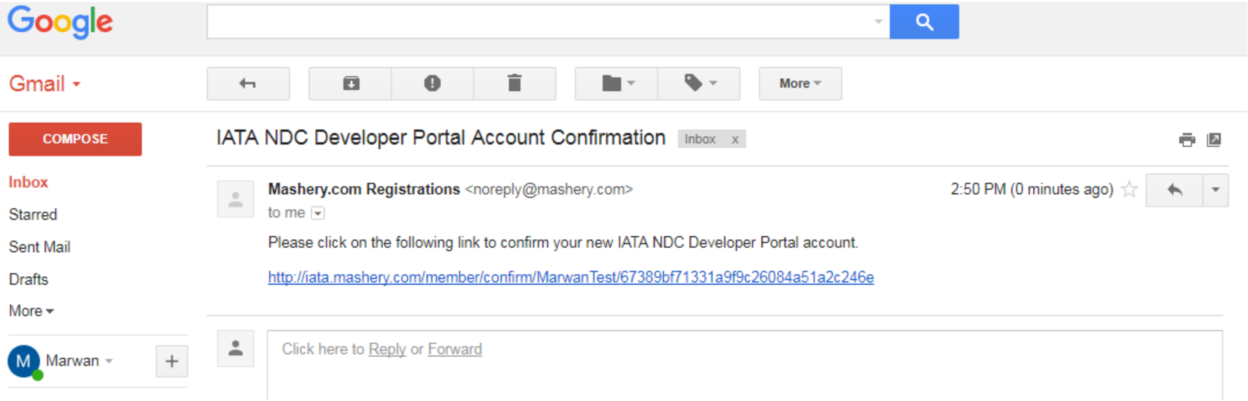
* Confirm E-mail
Please re-enter your e-mail address.

Password Requirements

- At least one letter
- At least one number
- At least eight characters

Figure 3 A snapshot of registration page

Once you have completed the registration, you will need to check your e-mail account for a confirmation e-mail that will enable you to activate your account:



IATA NDC Developer Portal Account Confirmation Inbox x

Mashery.com Registrations <noreply@mashery.com> 2:50 PM (0 minutes ago)
to me

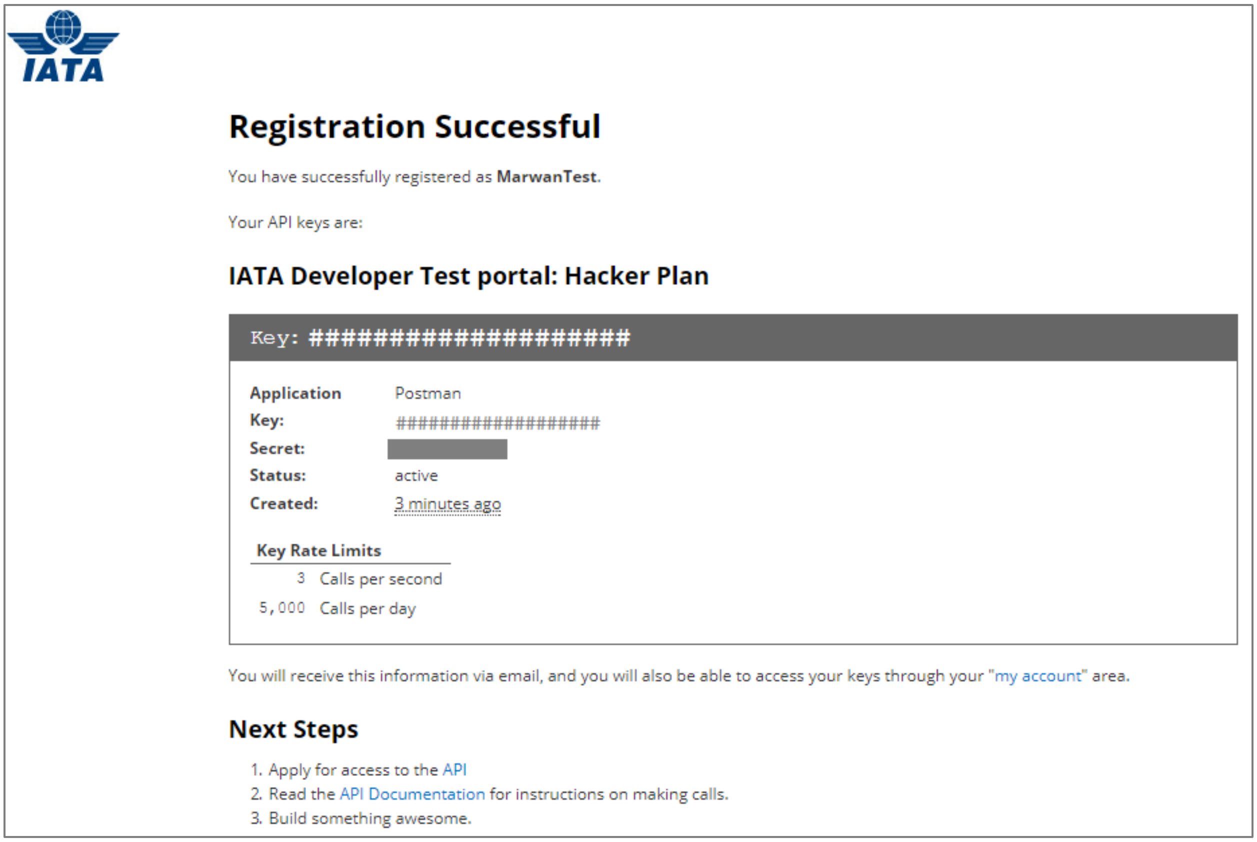
Please click on the following link to confirm your new IATA NDC Developer Portal account.

http://iata.mashery.com/member/confirm/MarwanTest/67389bf71331a9f9c26084a51a2c246e

Click here to Reply or Forward

Figure 4 Confirmation e-mail

After you have clicked you on the link confirming your registration, you will be taken to your details page which includes most importantly, your **API Key**.



IATA

Registration Successful

You have successfully registered as **MarwanTest**.

Your API keys are:

IATA Developer Test portal: Hacker Plan

Key: ####################

Application	Postman
Key:	###################
Secret:	[redacted]
Status:	active
Created:	3 minutes ago

	Key Rate Limits
3	Calls per second
5,000	Calls per day

You will receive this information via email, and you will also be able to access your keys through your "my account" area.

Next Steps

1. Apply for access to the API
2. Read the API Documentation for instructions on making calls.
3. Build something awesome.

Figure 5 Successful Registration with API Key

Step Four: Download Postman

Postman a tool for API development; it is a medium enabling you to send and receive messages from the ONE Order Sandbox. You can download a FREE version of Postman by going to this link:

https://www.getpostman.com/

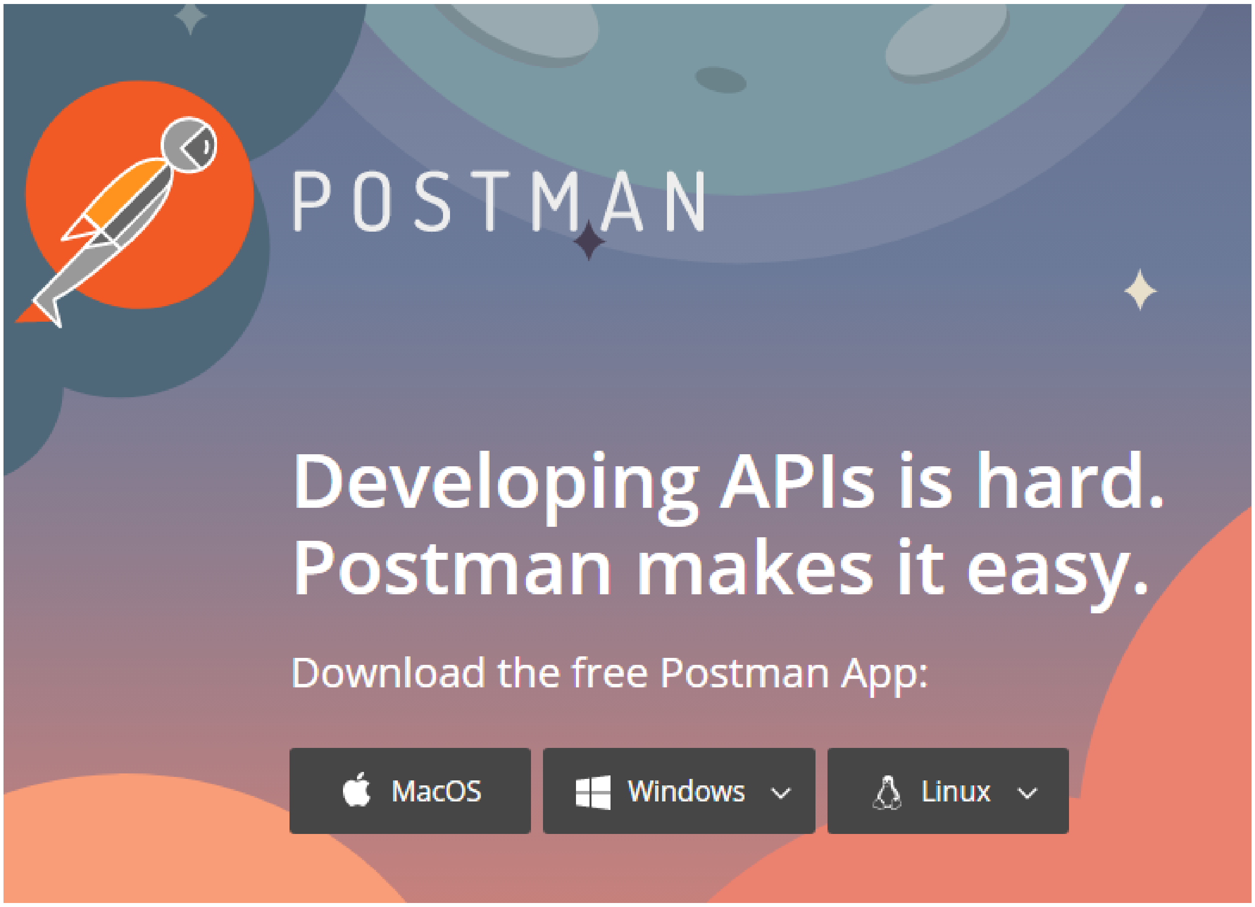


Figure 6 Postman landing page

Once Postman has been downloaded, you will have to sign up:

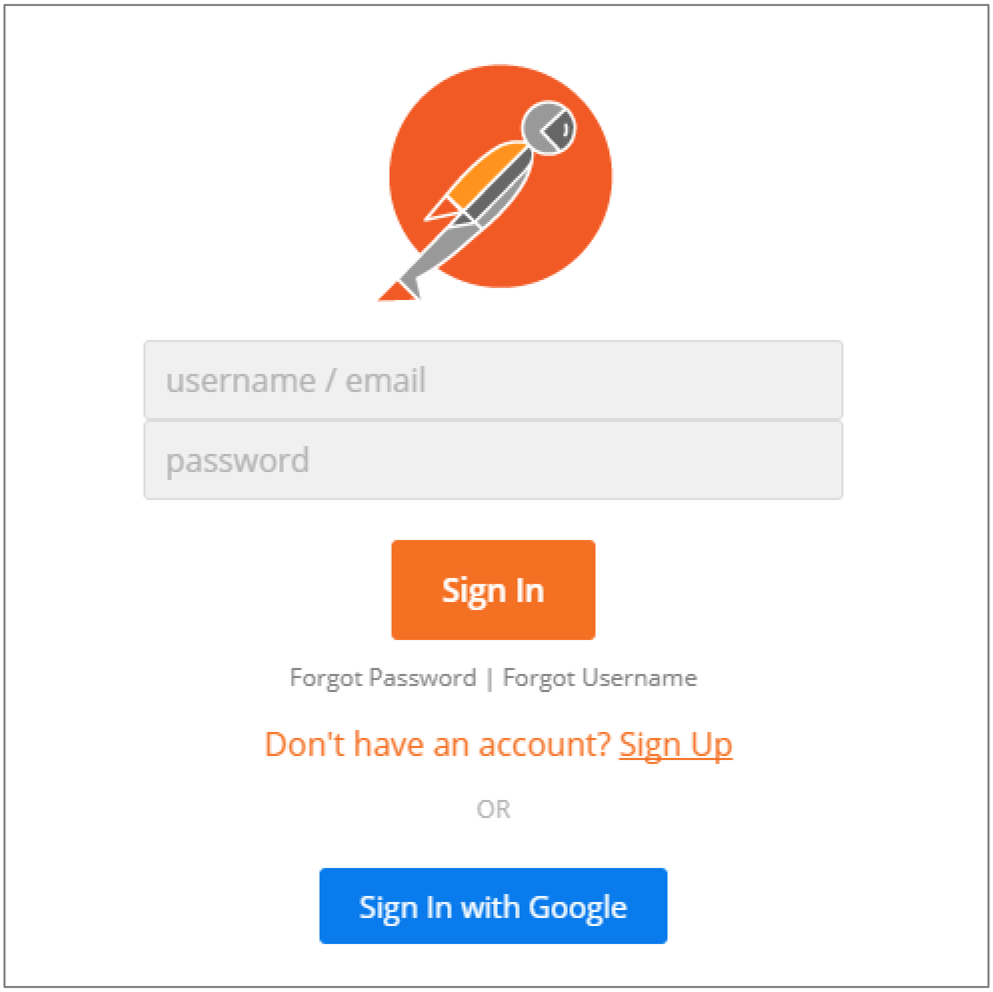
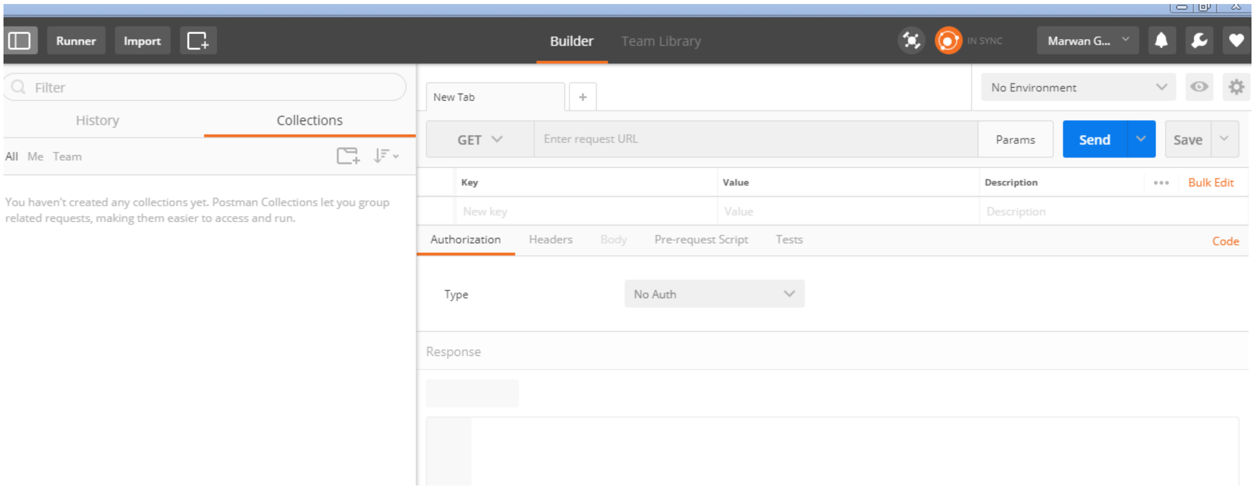

username / email
password
Sign In
Forgot Password | Forgot Username
Don't have an account? Sign Up
OR
Sign In with Google

Figure 7 Sign up page of Postman

If this is the first time you are using Postman, then you'd need to sign up with the option of signing up with you Google details (if you have Gmail or a Google account). If you are familiar with Postman then you can sign in.



You haven't created any collections yet. Postman Collections let you group related requests, making them easier to access and run.

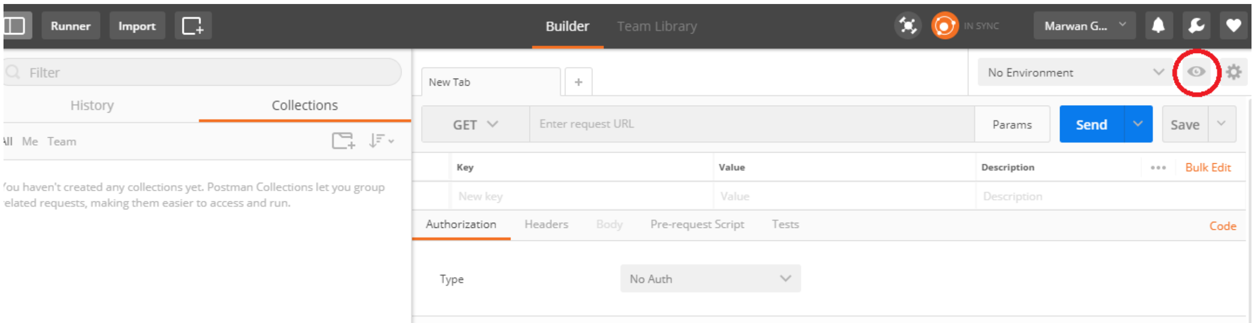
Figure 8 Postman blank landing page

Step Five: Setting up Postman

This section describes in details how to properly set up your Postman prior to testing.

a. Setting *Globals*: this is a general setting that you have to do only once to enable you to test.

   i. On the landing page of Postman go to the Globals icon which looks like an eye on the upper right corner of the Postman window (Fig 9).



'ou haven't created any collections yet. Postman Collections let you group elated requests, making them easier to access and run.

Figure 9 Globals Icon on the upper right corner

   ii. Press the Global item and select Edit (Fig 10)

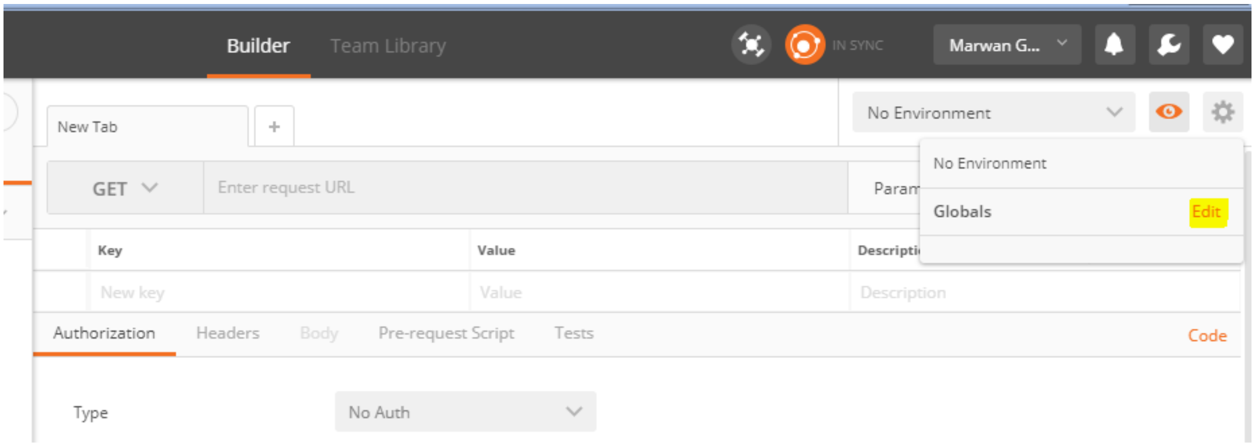
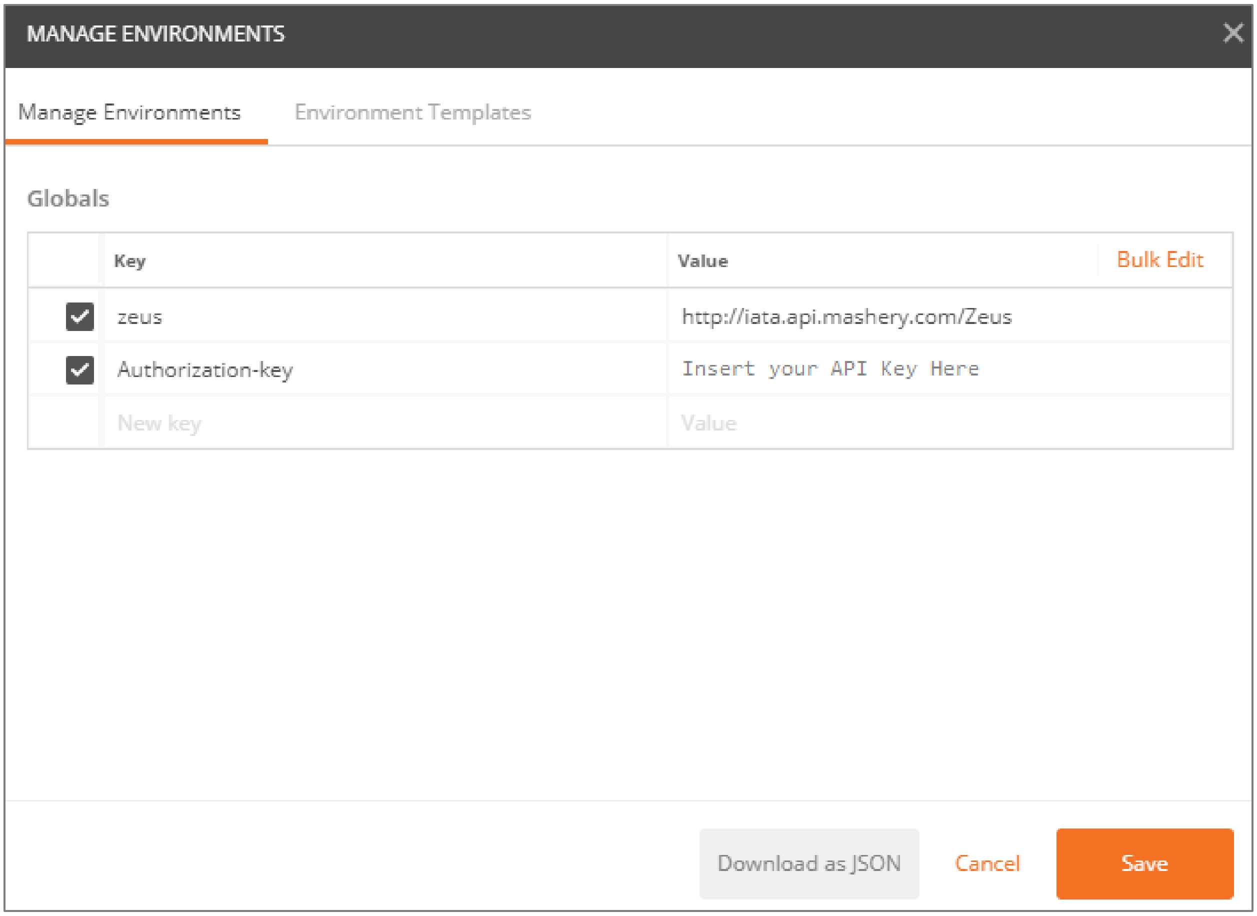


Figure 10 Press edit

iii. Refer to the below table and screenshot (Fig 11) and input the following variables (Case Sensitive); Click save when you are done:

Key	**Value**
zeus	`http://iata.api.mashery.com/Zeus`
Authorization-key	Insert your personal API key here (refer to step 3)



Globals

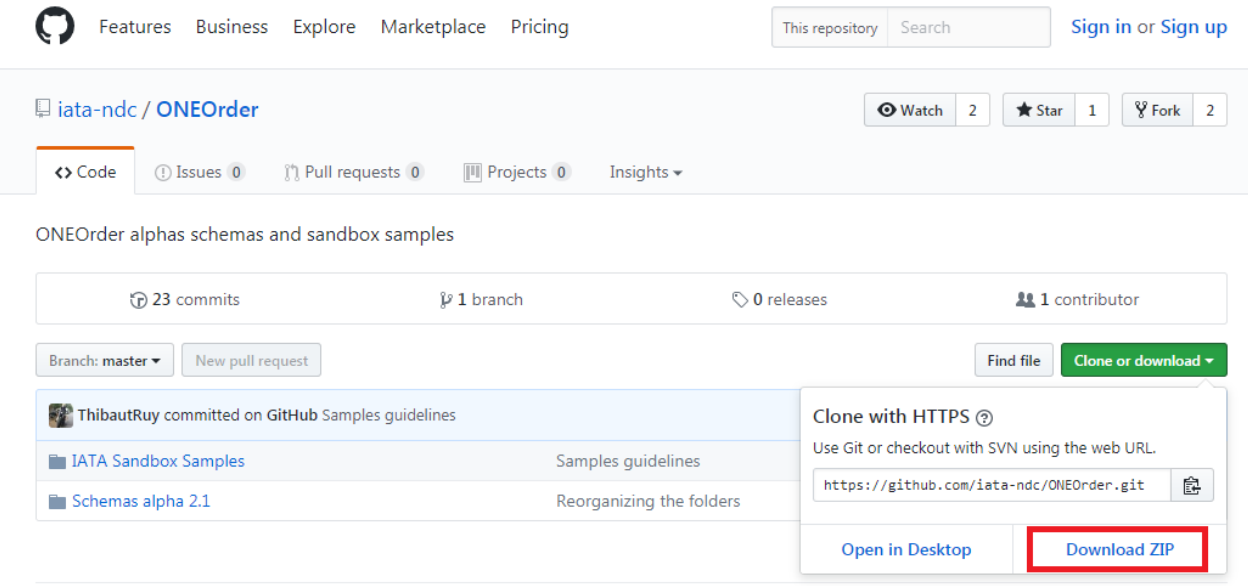
	Key	Value
☑	zeus	http://iata.api.mashery.com/Zeus
☑	Authorization-key	Insert your API Key Here
	New key	Value

Figure 11 Setting up Globals

b. Import collection (ONE Order Samples (NDC 17.1 and OO 2.1) 21 04 2017.postman_collection v2)

   i. Go to the Github main page: `https://github.com/iata-ndc/ONEOrder`

   ii. Click on Clone or download on the right hand of the window (Fig 12) and download ZIP on your local drive. Unzip the contents of the Zip File.



ONEOrder alphas schemas and sandbox samples

ThibautRuy committed on GitHub Samples guidelines

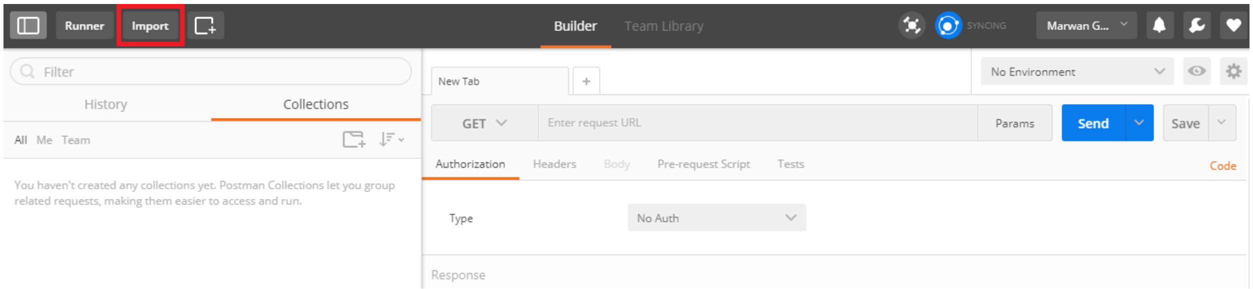
IATA Sandbox Samples	Samples guidelines
Schemas alpha 2.1	Reorganizing the folders

Use Git or checkout with SVN using the web URL.

```
https://github.com/iata-ndc/ONEOrder.git
```

Figure 12 Github main page “Clone or download”

iii. Back on Postman, click Import on the upper left corner of the window as shown in (Fig 13)

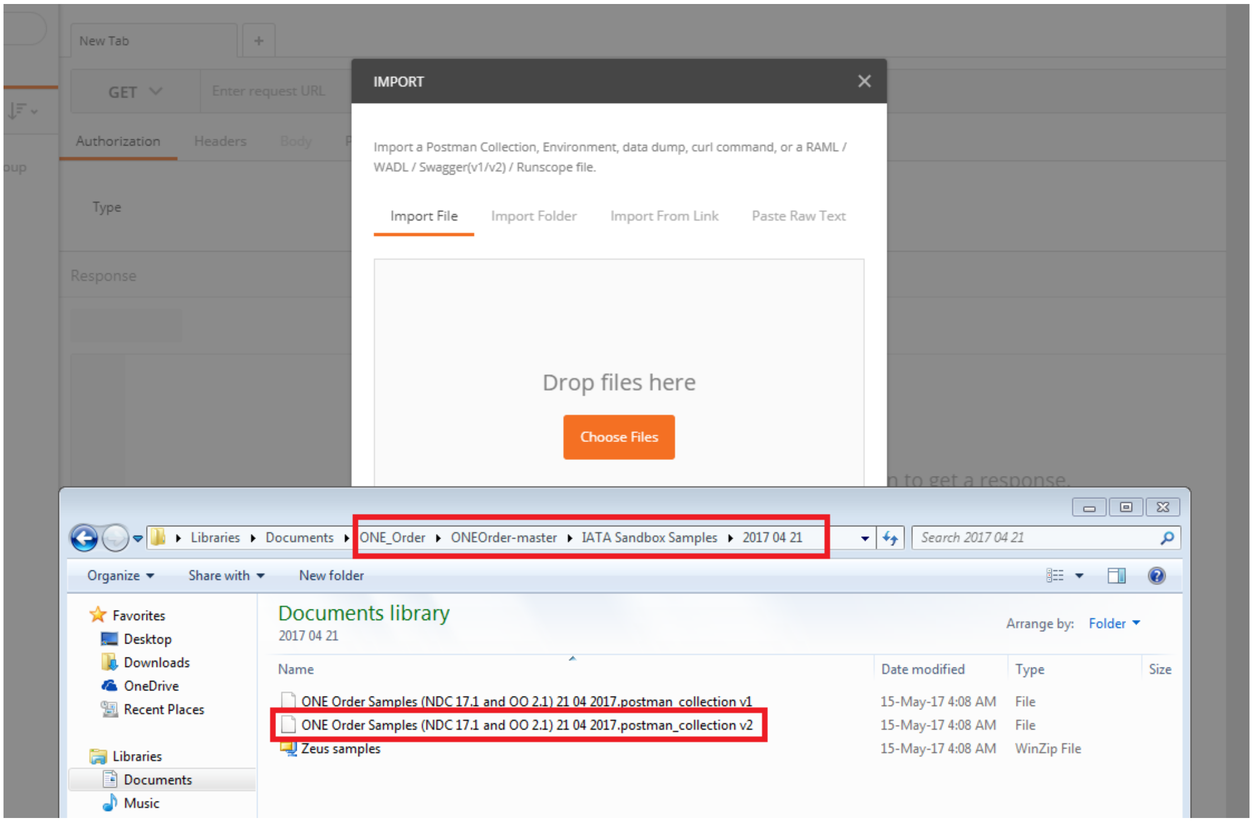


You haven't created any collections yet. Postman Collections let you group related requests, making them easier to access and run.

Figure 13 Import Collection on Postman

iv. Drop `ONE Order Samples (NDC 17.1 and OO 2.1) 21 04 2017.postman_collection v2` to Postman. You can find the collection v2 by following this path:

- `ONEOrder-master`
    - `IATA Sandbox Samples`
        - `2017 04 21`
            - `ONE Order Samples (NDC 17.1 and OO 2.1) 21 04 2017.postman_collection v2`



Import a Postman Collection, Environment, data dump, curl command, or a RAML / WADL / Swagger(v1/v2) / Runscope file.

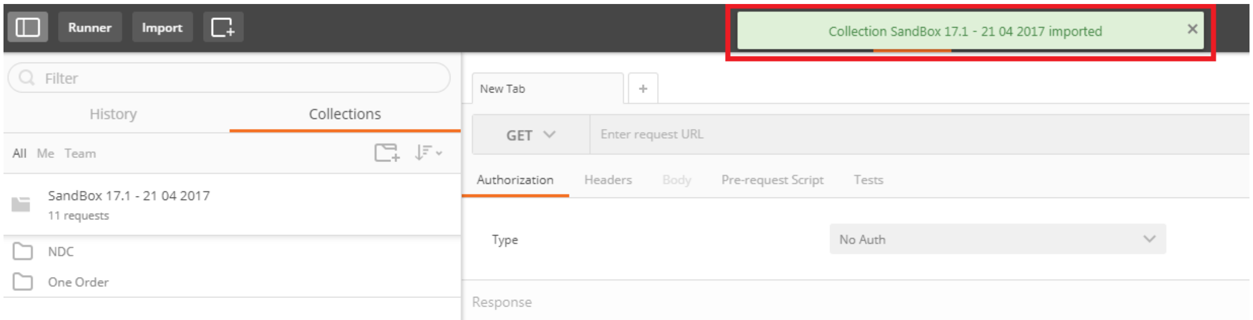
Libraries ▸ Documents ▸ ONE_Order ▸ ONEOrder-master ▸ IATA Sandbox Samples ▸ 2017 04 21

Documents library
2017 04 21

Name	Date modified	Type	Size
ONE Order Samples (NDC 17.1 and OO 2.1) 21 04 2017.postman_collection v1	15-May-17 4:08 AM	File	
ONE Order Samples (NDC 17.1 and OO 2.1) 21 04 2017.postman_collection v2	15-May-17 4:08 AM	File	
Zeus samples	15-May-17 4:08 AM	WinZip File	

Figure 14 Drag and drop ONE Order collection to Postman

v. A green pop-up notification at the top page of postman will appear to confirm that the collection has been imported successfully to Postman (Fig 15)



Collection SandBox 17.1 - 21 04 2017 imported

SandBox 17.1 - 21 04 2017
11 requests
NDC
One Order

Figure 15 Confirmation message

vi. Your collection is now imported (the left part of the Postman window) and you are now ready to interact with the ONE Order Sandbox.

Step Six: Using Postman to send messages to

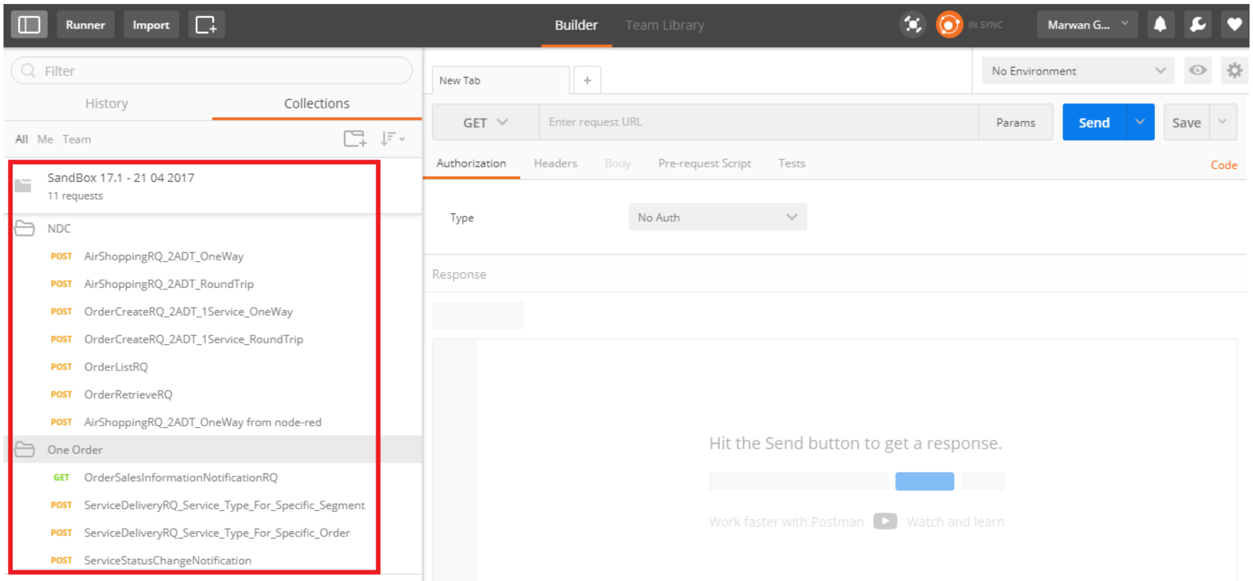
After you have fulfilled the prerequisites described in the above sections, you are now ready to interact with the ONE Order Sandbox: ZEUS Airlines through Postman.

Scenario: the below section explains a logical flow composed of 3 main messages:

Message Name	**Simple Business Meaning**
AirShoppingRQ_2ADT_OneWay	Passenger booking one way ticket in an NDC environment.
OrderCreateRQ_2ADT_1Service_OneWay	After the booking is done, an Order is created.
OrderSalesInformationNotificationRQ	This message is sent to retrieve information about the Order.

Of course you are able to experiment with your own flow according to your interests with different messages and combinations but for the sake of presenting how the sending and receiving via Postman works, we will explore those 3 messages.

i. Expand the NDC and ONE Order folders by simply clicking on them as shown in Fig



SandBox 17.1 - 21 04 2017
11 requests

NDC
- POST AirShoppingRQ_2ADT_OneWay
- POST AirShoppingRQ_2ADT_RoundTrip
- POST OrderCreateRQ_2ADT_1Service_OneWay
- POST OrderCreateRQ_2ADT_1Service_RoundTrip
- POST OrderListRQ
- POST OrderRetrieveRQ
- POST AirShoppingRQ_2ADT_OneWay from node-red

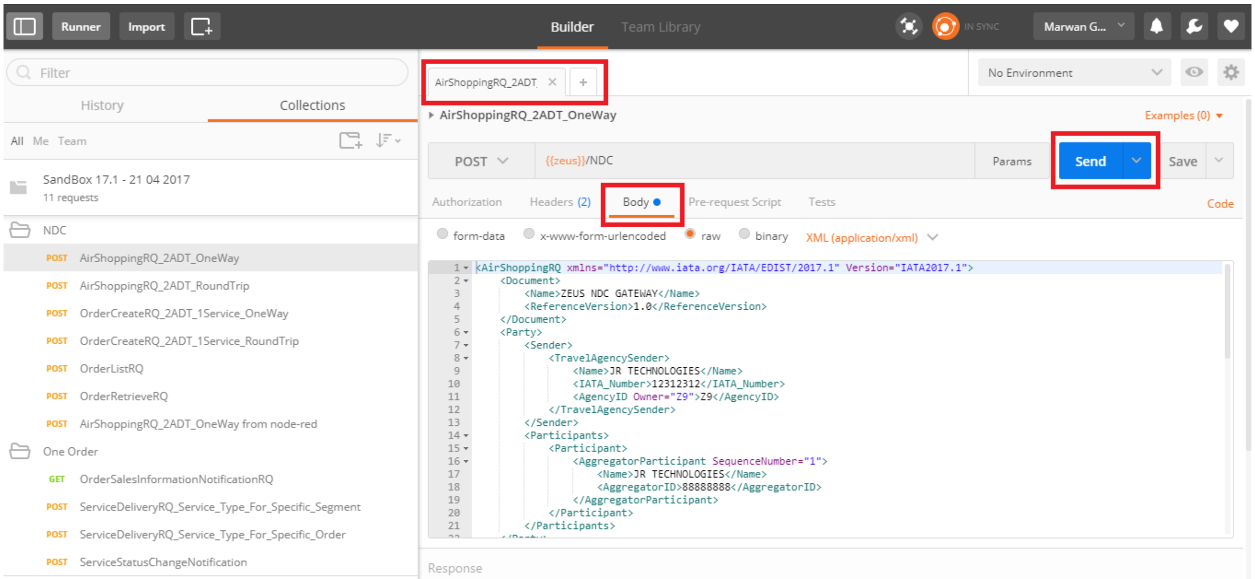
One Order
- GET OrderSalesInformationNotificationRQ
- POST ServiceDeliveryRQ_Service_Type_For_Specific_Segment
- POST ServiceDeliveryRQ_Service_Type_For_Specific_Order
- POST ServiceStatusChangeNotification

Figure 16 Expanded NDC and ONE Order collection folders

ii. *AirShoppingRQ_2ADT_OneWay:* Click on the AirShoppingRQ_OneWay message and the lay out of the page will appear like Fig 17 – you will notice the following observations:

- A new tap appears named AirShoppingRQ_2ADT_OneWay

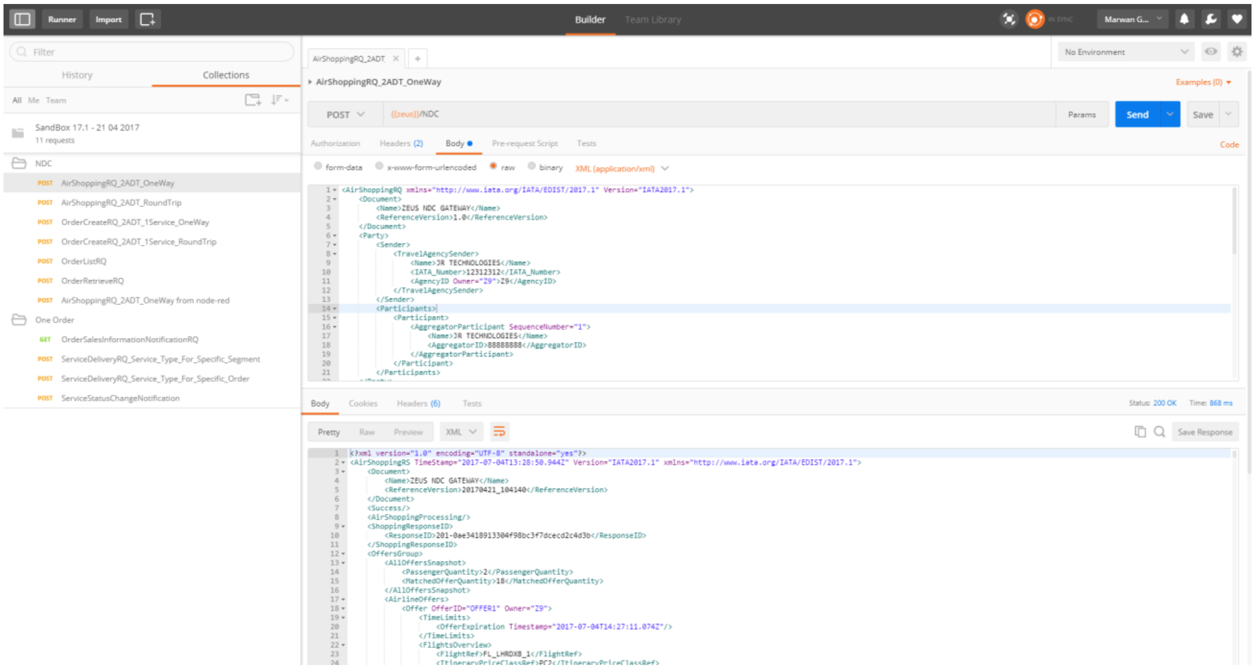
- The HTTP Method for this particular message is POST (Submits data to be processed to a specified resource)
- If you click on Body, you can examine the body of the messages. You will realize that this message is already prefilled with certain information about the one way trip such as Origin and Destination Airports along with the desired date of travel.
- Click send to **send** message to the ONE Order Sandbox. The response appears under the body of the message.


Runner
Import
Builder
Team Library
IN SYNC
Marwan G...
Filter
History
Collections
All Me Team
SandBox 17.1 - 21 04 2017
11 requests
NDC
POST AirShoppingRQ_2ADT_OneWay
POST AirShoppingRQ_2ADT_RoundTrip
POST OrderCreateRQ_2ADT_1Service_OneWay
POST OrderCreateRQ_2ADT_1Service_RoundTrip
POST OrderListRQ
POST OrderRetrieveRQ
POST AirShoppingRQ_2ADT_OneWay from node-red
One Order
GET OrderSalesInformationNotificationRQ
POST ServiceDeliveryRQ_Service_Type_For_Specific_Segment
POST ServiceDeliveryRQ_Service_Type_For_Specific_Order
POST ServiceStatusChangeNotification
AirShoppingRQ_2ADT ×
+
No Environment
AirShoppingRQ_2ADT_OneWay
Examples (0)
Params
Send
Save
Authorization
Headers (2)
Body
Pre-request Script
Tests
Code
○ form-data
○ x-www-form-urlencoded
● raw
○ binary
XML (application/xml)
Response

POST {{zeus}}/NDC

```
<AirShoppingRQ xmlns="http://www.iata.org/IATA/EDIST/2017.1" Version="IATA2017.1">
    <Document>
        <Name>ZEUS NDC GATEWAY</Name>
        <ReferenceVersion>1.0</ReferenceVersion>
    </Document>
    <Party>
        <Sender>
            <TravelAgencySender>
                <Name>JR TECHNOLOGIES</Name>
                <IATA_Number>12312312</IATA_Number>
                <AgencyID Owner="Z9">Z9</AgencyID>
            </TravelAgencySender>
        </Sender>
        <Participants>
            <Participant>
                <AggregatorParticipant SequenceNumber="1">
                    <Name>JR TECHNOLOGIES</Name>
                    <AggregatorID>88888888</AggregatorID>
                </AggregatorParticipant>
            </Participant>
        </Participants>
```

Figure 17 AirShoppingRQ_OneWay POST



Runner
Import
Builder
Team Library
Marwan G...
Filter
History
Collections
All Me Team
SandBox 17.1 - 21 04 2017
11 requests
NDC
POST AirShoppingRQ_2ADT_OneWay
POST AirShoppingRQ_2ADT_RoundTrip
POST OrderCreateRQ_2ADT_1Service_OneWay
POST OrderCreateRQ_2ADT_1Service_RoundTrip
POST OrderListRQ
POST OrderRetrieveRQ
POST AirShoppingRQ_2ADT_OneWay from node-red
One Order
GET OrderSalesInformationNotificationRQ
POST ServiceDeliveryRQ_Service_Type_For_Specific_Segment
POST ServiceDeliveryRQ_Service_Type_For_Specific_Order
POST ServiceStatusChangeNotification
AirShoppingRQ_2ADT
No Environment
AirShoppingRQ_2ADT_OneWay
Examples (0)
Params
Send
Save
Authorization
Headers (2)
Body
Pre-request Script
Tests
Code
Cookies
Headers (6)
Pretty
Raw
Preview
XML
Save Response

POST {{zeus}}/NDC

○ form-data ○ x-www-form-urlencoded ● raw ○ binary XML (application/xml)

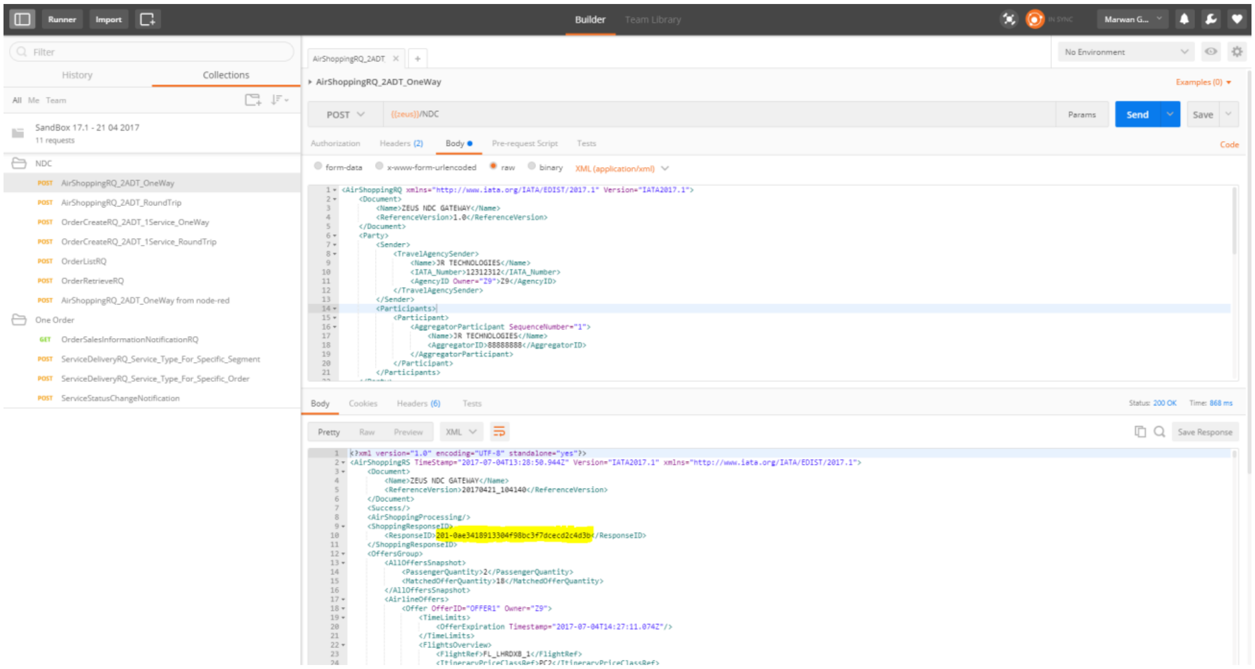
```
<AirShoppingRQ xmlns="http://www.iata.org/IATA/EDIST/2017.1" Version="IATA2017.1">
    <Document>
        <Name>ZEUS NDC GATEWAY</Name>
        <ReferenceVersion>1.0</ReferenceVersion>
    </Document>
    <Party>
        <Sender>
            <TravelAgencySender>
                <Name>JR TECHNOLOGIES</Name>
                <IATA_Number>12312312</IATA_Number>
                <AgencyID Owner="Z9">Z9</AgencyID>
            </TravelAgencySender>
        </Sender>
        <Participants>
            <Participant>
                <AggregatorParticipant SequenceNumber="1">
                    <Name>JR TECHNOLOGIES</Name>
                    <AggregatorID>88888888</AggregatorID>
                </AggregatorParticipant>
            </Participant>
        </Participants>
    </Party>
```

Status: 200 OK Time: 868 ms

```
1  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2  <AirShoppingRS TimeStamp="2017-07-04T13:28:50.944Z" Version="IATA2017.1" xmlns="http://www.iata.org/IATA/EDIST/2017.1">
3      <Document>
4          <Name>ZEUS NDC GATEWAY</Name>
5          <ReferenceVersion>20170421_104140</ReferenceVersion>
6      </Document>
7      <Success/>
8      <AirShoppingProcessing/>
9      <ShoppingResponseID>
10         <ResponseID>201-0ae3418913304f98bc3f7dcecd2c4d3b</ResponseID>
11     </ShoppingResponseID>
12     <OffersGroup>
13         <AllOffersSnapshot>
14             <PassengerQuantity>2</PassengerQuantity>
15             <MatchedOfferQuantity>18</MatchedOfferQuantity>
16         </AllOffersSnapshot>
17         <AirlineOffers>
18             <Offer OfferID="OFFER1" Owner="Z9">
19                 <TimeLimits>
20                     <OfferExpiration Timestamp="2017-07-04T14:27:11.074Z"/>
21                 </TimeLimits>
22                 <FlightsOverview>
23                     <FlightRef>FL_LHRDXB_1</FlightRef>
24                     <ItineraryPriceClassRef>PC2</ItineraryPriceClassRef>
```

Figure 18 AirShoppingRQ_OneWay Response

- A note of the response message: this message contains the ResponseID (line 10) to be used for the following message OrderCreateRQ_2ADT_1Service_OneWay (Fig 19)



POST {{zeus}}/NDC

```
<AirShoppingRQ xmlns="http://www.iata.org/IATA/EDIST/2017.1" Version="IATA2017.1">
    <Document>
        <Name>ZEUS NDC GATEWAY</Name>
        <ReferenceVersion>1.0</ReferenceVersion>
    </Document>
    <Party>
        <Sender>
            <TravelAgencySender>
                <Name>JR TECHNOLOGIES</Name>
                <IATA_Number>12312312</IATA_Number>
                <AgencyID Owner="Z9">Z9</AgencyID>
            </TravelAgencySender>
        </Sender>
        <Participants>
            <Participant>
                <AggregatorParticipant SequenceNumber="1">
                    <Name>JR TECHNOLOGIES</Name>
                    <AggregatorID>88888888</AggregatorID>
                </AggregatorParticipant>
            </Participant>
        </Participants>
```

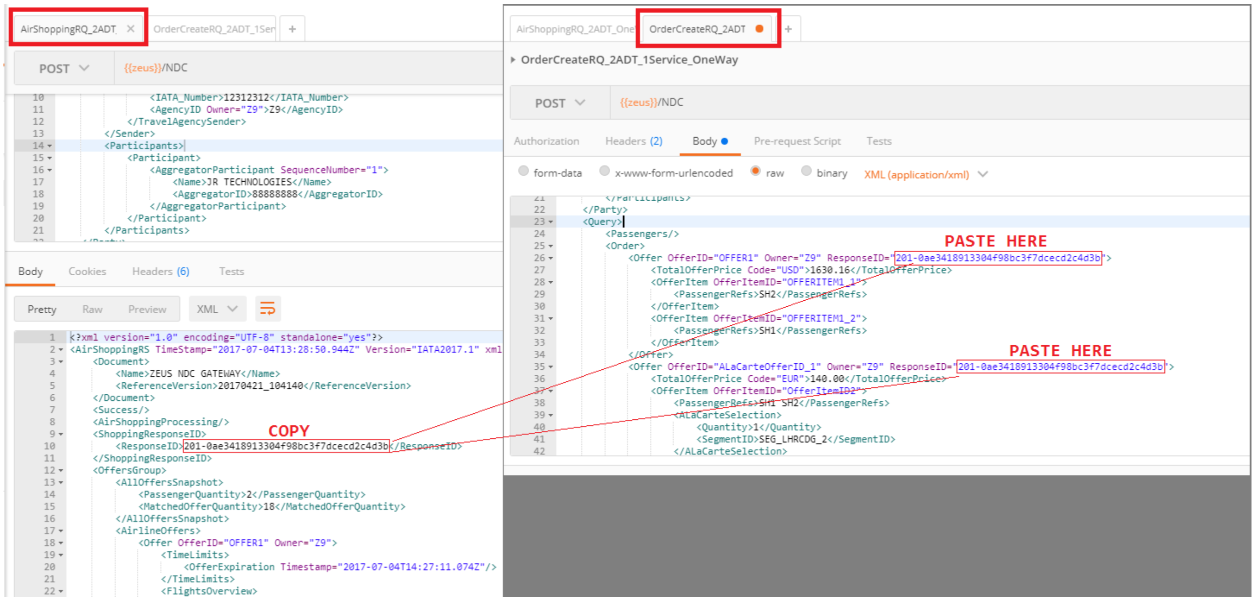
Status: 200 OK Time: 868 ms

```
1  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2  <AirShoppingRS TimeStamp="2017-07-04T13:28:50.944Z" Version="IATA2017.1" xmlns="http://www.iata.org/IATA/EDIST/2017.1">
3      <Document>
4          <Name>ZEUS NDC GATEWAY</Name>
5          <ReferenceVersion>20170421_104140</ReferenceVersion>
6      </Document>
7      <Success/>
8      <AirShoppingProcessing/>
9      <ShoppingResponseID>
10         <ResponseID>201-0ae3418913304f98bc3f7dcecd2c4d3b</ResponseID>
11     </ShoppingResponseID>
12     <OffersGroup>
13         <AllOffersSnapshot>
14             <PassengerQuantity>2</PassengerQuantity>
15             <MatchedOfferQuantity>18</MatchedOfferQuantity>
16         </AllOffersSnapshot>
17         <AirlineOffers>
18             <Offer OfferID="OFFER1" Owner="Z9">
19                 <TimeLimits>
20                     <OfferExpiration Timestamp="2017-07-04T14:27:11.074Z"/>
21                 </TimeLimits>
22                 <FlightsOverview>
23                     <FlightRef>FL_LHRDXB_1</FlightRef>
24                     <ItineraryPriceClassRef>PC2</ItineraryPriceClassRef>
```

Figure 19 ResponseID highlighted

iii. *OrderCreateRQ_2ADT_1Service_OneWay:*

- Click on the *OrderCreateRQ_2ADT_1Service_OneWay* message and select Body to check the message contents.
- Retrieve the ResponseID from the AirShoppingRQ_2ADT_OneWay Response message, and insert as an attribute in the elements OrderCreateRQ_2ADT_1Service_OneWay as shown in Fig 20.



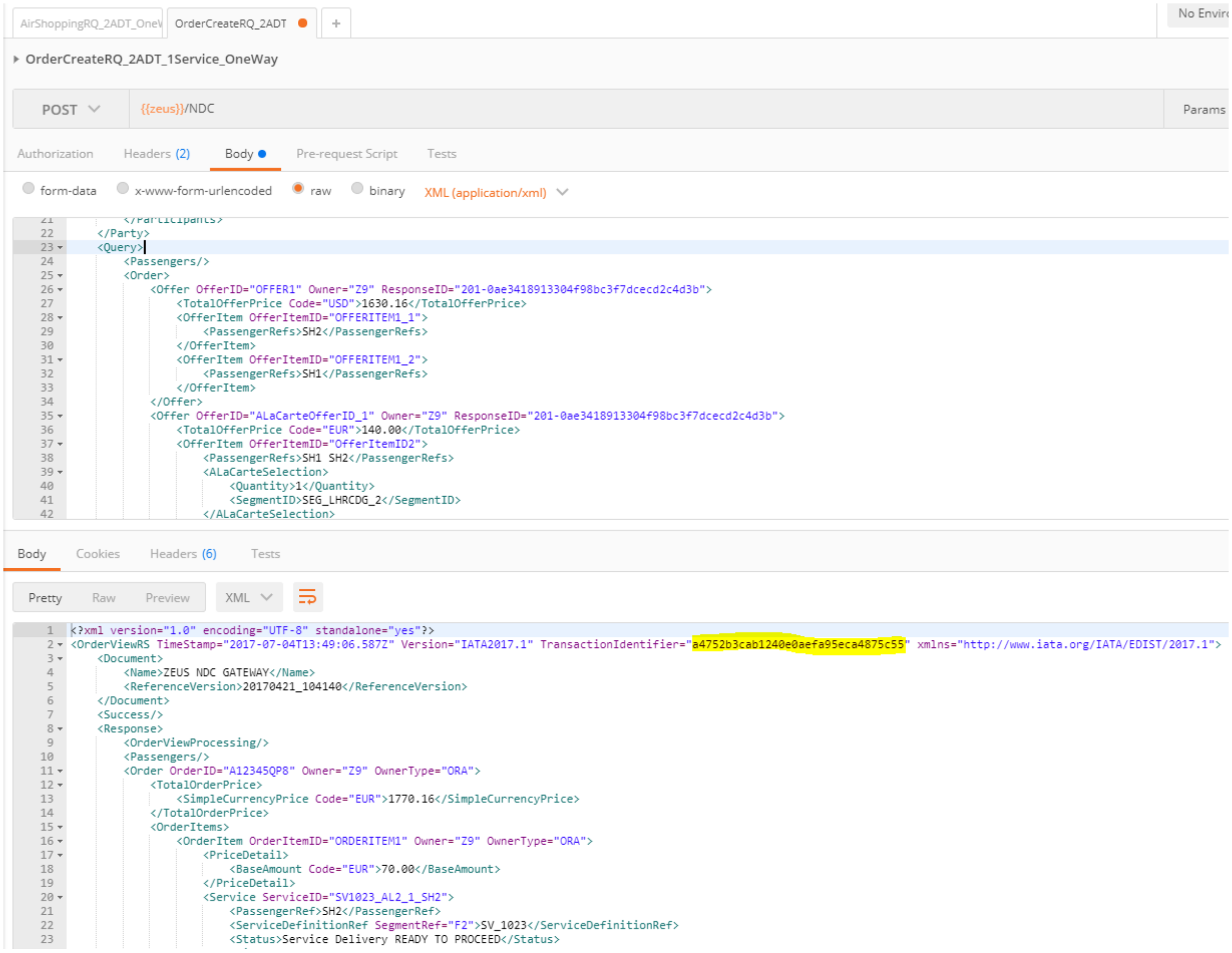
```
                <IATA_Number>12312312</IATA_Number>
                <AgencyID Owner="Z9">Z9</AgencyID>
            </TravelAgencySender>
        </Sender>
        <Participants>
            <Participant>
                <AggregatorParticipant SequenceNumber="1">
                    <Name>JR TECHNOLOGIES</Name>
                    <AggregatorID>88888888</AggregatorID>
                </AggregatorParticipant>
            </Participant>
        </Participants>
    </Party>
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<AirShoppingRS TimeStamp="2017-07-04T13:28:50.944Z" Version="IATA2017.1" xml
    <Document>
        <Name>ZEUS NDC GATEWAY</Name>
        <ReferenceVersion>20170421_104140</ReferenceVersion>
    </Document>
    <Success/>
    <AirShoppingProcessing/>
    <ShoppingResponseID>
        <ResponseID>201-0ae3418913304f98bc3f7dcecd2c4d3b</ResponseID>
    </ShoppingResponseID>
    <OffersGroup>
        <AllOffersSnapshot>
            <PassengerQuantity>2</PassengerQuantity>
            <MatchedOfferQuantity>18</MatchedOfferQuantity>
        </AllOffersSnapshot>
        <AirlineOffers>
            <Offer OfferID="OFFER1" Owner="Z9">
                <TimeLimits>
                    <OfferExpiration Timestamp="2017-07-04T14:27:11.074Z"/>
                </TimeLimits>
                <FlightsOverview>
```

```
        </Participants>
    </Party>
    <Query>
        <Passengers/>
        <Order>
            <Offer OfferID="OFFER1" Owner="Z9" ResponseID="201-0ae3418913304f98bc3f7dcecd2c4d3b">
                <TotalOfferPrice Code="USD">1630.16</TotalOfferPrice>
                <OfferItem OfferItemID="OFFERITEM1_1">
                    <PassengerRefs>SH2</PassengerRefs>
                </OfferItem>
                <OfferItem OfferItemID="OFFERITEM1_2">
                    <PassengerRefs>SH1</PassengerRefs>
                </OfferItem>
            </Offer>
            <Offer OfferID="ALaCarteOfferID_1" Owner="Z9" ResponseID="201-0ae3418913304f98bc3f7dcecd2c4d3b">
                <TotalOfferPrice Code="EUR">140.00</TotalOfferPrice>
                <OfferItem OfferItemID="OfferItemID2">
                    <PassengerRefs>SH1 SH2</PassengerRefs>
                    <ALaCarteSelection>
                        <Quantity>1</Quantity>
                        <SegmentID>SEG_LHRCDG_2</SegmentID>
                    </ALaCarteSelection>
```

Figure 20 ResponseID in OrderCreateRQ message

- Press **send** to get a response; kindly note the TransactionIdentifier to be used in the following message as highlighted in Fig 21



```xml
        </Participants>
    </Party>
    <Query>
        <Passengers/>
        <Order>
            <Offer OfferID="OFFER1" Owner="Z9" ResponseID="201-0ae3418913304f98bc3f7dcecd2c4d3b">
                <TotalOfferPrice Code="USD">1630.16</TotalOfferPrice>
                <OfferItem OfferItemID="OFFERITEM1_1">
                    <PassengerRefs>SH2</PassengerRefs>
                </OfferItem>
                <OfferItem OfferItemID="OFFERITEM1_2">
                    <PassengerRefs>SH1</PassengerRefs>
                </OfferItem>
            </Offer>
            <Offer OfferID="ALaCarteOfferID_1" Owner="Z9" ResponseID="201-0ae3418913304f98bc3f7dcecd2c4d3b">
                <TotalOfferPrice Code="EUR">140.00</TotalOfferPrice>
                <OfferItem OfferItemID="OfferItemID2">
                    <PassengerRefs>SH1 SH2</PassengerRefs>
                    <ALaCarteSelection>
                        <Quantity>1</Quantity>
                        <SegmentID>SEG_LHRCDG_2</SegmentID>
                    </ALaCarteSelection>
```

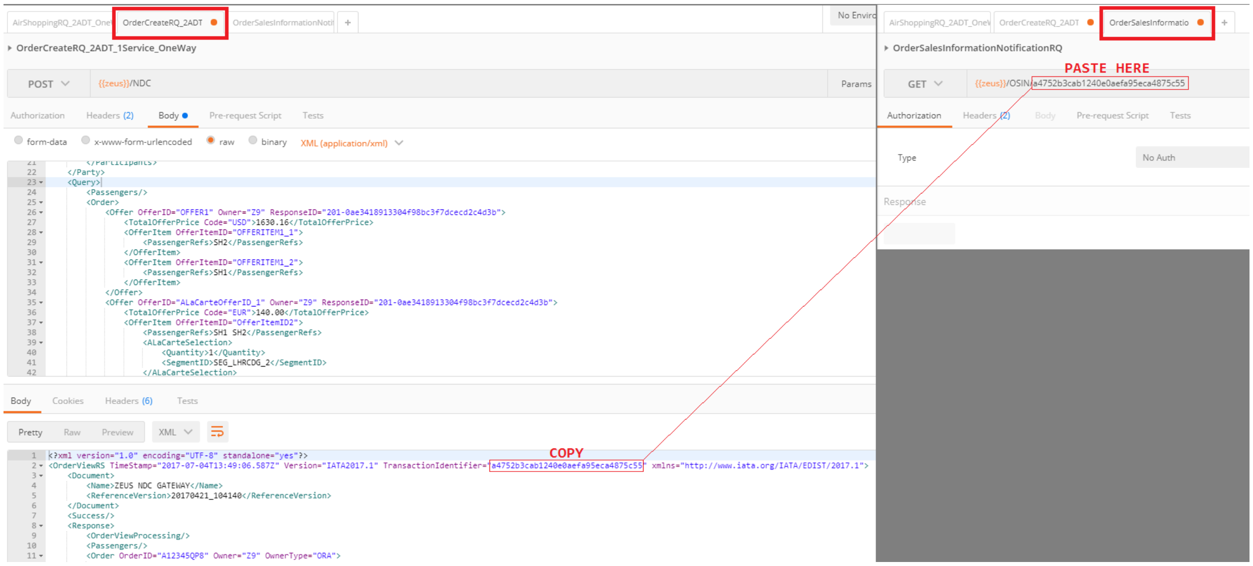
```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<OrderViewRS TimeStamp="2017-07-04T13:49:06.587Z" Version="IATA2017.1" TransactionIdentifier="a4752b3cab1240e0aefa95eca4875c55" xmlns="http://www.iata.org/IATA/EDIST/2017.1">
    <Document>
        <Name>ZEUS NDC GATEWAY</Name>
        <ReferenceVersion>20170421_104140</ReferenceVersion>
    </Document>
    <Success/>
    <Response>
        <OrderViewProcessing/>
        <Passengers/>
        <Order OrderID="A12345QP8" Owner="Z9" OwnerType="ORA">
            <TotalOrderPrice>
                <SimpleCurrencyPrice Code="EUR">1770.16</SimpleCurrencyPrice>
            </TotalOrderPrice>
            <OrderItems>
                <OrderItem OrderItemID="ORDERITEM1" Owner="Z9" OwnerType="ORA">
                    <PriceDetail>
                        <BaseAmount Code="EUR">70.00</BaseAmount>
                    </PriceDetail>
                    <Service ServiceID="SV1023_AL2_1_SH2">
                        <PassengerRef>SH2</PassengerRef>
                        <ServiceDefinitionRef SegmentRef="F2">SV_1023</ServiceDefinitionRef>
                        <Status>Service Delivery READY TO PROCEED</Status>
```

Figure 21 OrderCreateRQ Response Message with TransactionIdentifier highlighted

iv. *OrderSalesInformationNotificationRQ*

You will notice that the HTTP method for this message is GET (Requests data from a specified resource); this message is part of the ONE Order specialty messages.

- Click on the message OrderSalesInformationNotificationRQ under the ONE Order folder.
- Retrieve the TransactionIdentifier from OrderCreateRQ_2ADT_1Service_OneWay response message (OrderViewRS) and insert in the API endpoint as shown in Fig 22



```
        </Participants>
    </Party>
    <Query>
        <Passengers/>
        <Order>
            <Offer OfferID="OFFER1" Owner="Z9" ResponseID="201-0ae3418913304f98bc3f7dcecd2c4d3b">
                <TotalOfferPrice Code="USD">1630.16</TotalOfferPrice>
                <OfferItem OfferItemID="OFFERITEM1_1">
                    <PassengerRefs>SH2</PassengerRefs>
                </OfferItem>
                <OfferItem OfferItemID="OFFERITEM1_2">
                    <PassengerRefs>SH1</PassengerRefs>
                </OfferItem>
            </Offer>
            <Offer OfferID="ALaCarteOfferID_1" Owner="Z9" ResponseID="201-0ae3418913304f98bc3f7dcecd2c4d3b">
                <TotalOfferPrice Code="EUR">140.00</TotalOfferPrice>
                <OfferItem OfferItemID="OfferItemID2">
                    <PassengerRefs>SH1 SH2</PassengerRefs>
                    <ALaCarteSelection>
                        <Quantity>1</Quantity>
                        <SegmentID>SEG_LHRCDG_2</SegmentID>
                    </ALaCarteSelection>
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<OrderViewRS TimeStamp="2017-07-04T13:49:06.587Z" Version="IATA2017.1" TransactionIdentifier="a4752b3cab1240e0aefa95eca4875c55" xmlns="http://www.iata.org/IATA/EDIST/2017.1">
    <Document>
        <Name>ZEUS NDC GATEWAY</Name>
        <ReferenceVersion>20170421_104140</ReferenceVersion>
    </Document>
    <Success/>
    <Response>
        <OrderViewProcessing/>
        <Passengers/>
        <Order OrderID="A12345QP8" Owner="Z9" OwnerType="ORA">
```

```
{{zeus}}/OSIN/a4752b3cab1240e0aefa95eca4875c55
```

Type: No Auth

Figure 22 TransactionIdentifier in the API Endpoint of OSIN Message

- Press **send** and analyze the information contained in the response.