

FlightsApi

Web services Saop Et Rest



Installation

Table des matières

[1. Installer Python 3](#_Toc68261498)

[2. Installer les modules complémentaires 4](#_Toc68261499)

[3. Installer l’api flights python 5](#_Toc68261500)

[4. Lancer l’api flights python 6](#_Toc68261501)

[5. Web Services Rest 7](#_Toc68261502)

[5.1 Rechercher un vol par son numéro 7](#_Toc68261503)

[5.2 Recherche un vol par ville de départ, d’arrivée et par date 7](#_Toc68261504)

[5.3 Prendre une réservation 7](#_Toc68261505)

[5.4 Rechercher une réservation par le nom du client 8](#_Toc68261506)

[5.5 Rechercher une réservation par numéro 8](#_Toc68261507)

[5.6 Supprimer une réservation par numéro 8](#_Toc68261508)

[5.7 Supprimer toutes les réservations 8](#_Toc68261509)

1. Installer Python

https://www.python.org/ftp/python/3.9.2/python-3.9.2-amd64.exe

Une image contenant texte

Description générée automatiquement

1. Installer les modules complémentaires

Ouvrir l'invite de commande windows

***pip3 install spyne***

***pip3 install lxml***

1. Installer l’api flights python

Créer le dossier ***C:\Formation***

Télécharger le fichier ***PyFlightsAPI.zip*** et le déplacer dans ce dossier

Décompresser le fichier vers ***PyFlightsAPI***

Une image contenant table

Description générée automatiquement

1. Lancer l’api flights python

Ouvrir l'invite de commande windows

Changer de répertoire pour aller dans la répertoire ***C:\Formation\ PyFlightsAPI***

***Cd C:\Formation\ PyFlightsAPI***

Pour lancer l’api soap

***python flight\_saop.py***

*listening to http://127.0.0.1:8000*

*wsdl is at:* [*http://localhost:8000/?wsdl*](http://localhost:8000/?wsdl)

*Pour lancer l’api rest*

***python flight\_rest.py***

1. Web Services Rest

URL : <http://localhost:5000/flights_api/v1/resources>

* 1. Rechercher un vol par son numéro

|  |  |
| --- | --- |
| Ressource | /Flights//{FlightNumber} |
| Méthode | GET |
| Paramètre | **Aucun** |
| Retour | [{"Airline": "SR", "ArrivalCity": "Denver", "ArrivalTime": "12:43 PM", "DepartureCity": "Paris", "DepartureTime": "11:02 AM", "FlightNumber": 16939, "Price": 168.2, "SeatsAvailable": 250, "DayOfWeek": "Wednesday"},  …] |

* 1. Recherche un vol par ville de départ, d’arrivée et par date

|  |  |
| --- | --- |
| Ressource | /Flights |
| Méthode | GET |
| Paramètre | **DepartureCity**=Paris&**ArrivalCity**=Denver&**Date**=2021-12-08 |
| Retour | {"Airline": "TWA", "ArrivalCity": "San Francisco", "ArrivalTime": "12:43 PM", "DepartureCity": "Seattle", "DepartureTime": "11:02 AM", "FlightNumber": 1060, "Price": 168.2, "SeatsAvailable": 250, "DayOfWeek": "Wednesday"} |

* 1. Prendre une réservation

|  |  |
| --- | --- |
| Ressource | /FlightOrders |
| Méthode | POST |
| Paramètre | {  "Class" : "Economy",  "CustomerName" : "IMHAH",  "DepartureDate" : "2021-12-08",  "FlightNumber" : "1060",  "NumberOfTickets" : "2"  } |
| Retour | {  "OrderNumber": 107,  "TotalPrice": 336.4  } |

* 1. Rechercher une réservation par le nom du client

|  |  |
| --- | --- |
| Ressource | /FlightOrders |
| Méthode | GET |
| Paramètre | CustomerName={{customer}} |
| Retour | [{"Class": "Economy", "CustomerName": "IMHAH", "DepartureDate": "2021-12-8 11:02", "FlightNumber": 1060, "NumberOfTickets": 2, "OrderNumber": 105}] |

* 1. Rechercher une réservation par numéro

|  |  |
| --- | --- |
| Ressource | /FlightOrders/{order} |
| Méthode | GET |
| Paramètre | Aucun |
| Retour | [{"Class": "Economy", "CustomerName": "IMHAH", "DepartureDate": "2021-12-8 11:02", "FlightNumber": 1060, "NumberOfTickets": 2, "OrderNumber": 105}] |

* 1. Supprimer une réservation par numéro

|  |  |
| --- | --- |
| Ressource | /FlightOrders/{order} |
| Méthode | DELETE |
| Paramètre |  |
| Retour | {      "true": "Oder deleted 105 "  } |

* 1. Supprimer toutes les réservations

|  |  |
| --- | --- |
| Ressource | /FlightOrders |
| Méthode | DELETE |
| Paramètre |  |
| Retour | {"result":"ok"} |