

JAX-Y-Demo



Source Download : https://sourceforge.net/projects/jax-y/?source=typ_redirect

Demo :

I) Download :

- 1) download jax-y-demo.zip from the link above
- 2) unzip jax-y-demo.zip

II) Install demo Database :

- 1) `chmod +x demo/db-script/db-planes.sh`
- 2) `./demo/db-script/db-planes.sh`

III) Run The executable Jar which deploy an endpoint at the URL :

<http://localhost:8080/rest/resources/>

- 1) `java -jar jax-y-swarm.jar`

IV) Run the GUI Client :

- 1) `java -jar GUI/jaxy-ui.jar`

V) Tests (Public service) :

- 1) Invoke the serviceDiscovery :
`http://localhost:8080/rest/resources/infoServices`
- 2) Invoke the service planes :
`http://localhost:8080/rest/resources/planes (XML / JSON)`
- 3) Filter on total_pssengers > 300 : `total_passengers=>_300`
- 4) Sort Only model : `model`
- 5) Sort Only model + distance_km : `model - distance_km`

VI) Add new Secured Service (customSignOn authentication) :

- 1) Stop the server
- 2) Uncomment vip_planes service
- 3) Uncomment customSignOn authentication in the serviceConf.yaml file
- 4) restart the server
- 5) Invoke the serviceDiscovery :
http://localhost:8080/rest/resources/infoServices
- 6) Invoke the service vip_planes :
http://localhost:8080/rest/resources/vip_planes
(XML / JSON / XML-ENCRYPTED / JSON-ENCRYPTED)
- 7) Test authentication by changing login / password / timeStamp
Test timeOut // test SQL type inference capacity
- 8) Decrypt data locally
- 9) Filter on total_pssengers > 300
total_passengers=>_300
total_passengers=>_300&model='Airbus A340-500'
total_passengers=>_300&model=_not_'Airbus A340-500'
- 10) Sort Only model : model
- 11) Sort Only model + distance_km : model - distance_km
- 12) Change Tags using "AS" in SQL Queries

VII) Test SSO authentication with KeyCloack (should works with HTTPS) :

- 1) Start KeyCloack SERVER (127.0.0.1:8180)
- 2) Stop the jaxy server
- 3) Comment customSignOn authentication in the serviceConf.yaml file
- 4) Uncomment SSO authentication in the serviceConf.yaml file
- 5) restart the server
- 6) Go to the SSO panel
- 7) Invoke the serviceDiscovery :
http://localhost:8080/rest/resources/infoServices
- 8) Invoke the service vip_planes :
http://localhost:8080/rest/resources/vip_planes
(XML / JSON / XML-ENCRYPTED / JSON-ENCRYPTED)
- 9) Test authentication by changing login / password / clientID / secretID

```
10 ) Filter on total_pssengers > 300 : total_passengers=>_300

11 ) Sort Only model : model

12 ) Sort Only model + distance_km : model - distance_km

13 ) Check logs in KeyCloack server for the user admin
```

VIII) Generate Shell-script for automation

```
1 ) Generate script

2 ) test Script
```

Upcoming Features :

```
* Global Configuration Supports ( Thread pool size, nb of threads by service,
  data queue size ... authentication server, HTTPS ... )

* Swagger-Angular-Client intergration

* GUI Client Https supports

* Runtime Algo Choice for Encryption ( AES - DES ... )

* Randomize IV ( Initialization Vector ) for AES ..

* Add support of NoSQL databases ??

* Tests Tests & Tests
```