WAY2LEARN

LAB- Consuming Restful Services

In this lab, you will be working on **01-consuming-rest-start** project under **02-consuming-webservices** section

In this lab you will understand

a) How to consume a restful webservice

b) How to extract uri params from http request and pass uri params to a restful service

STEP 1

1)Execute **rest.bat** given to you in lab-docs folder . This will start a rest service which starts listening on port **7070** and base path is **/rest**

2) Give requests to http://localhost:7070/rest/products and observe that you get all products in Json format

Give requests to **http://localhost:7070/rest/products/Mac** will give give you products whose name contains Mac. Try the same for

http://localhost:7070/rest/products/Hp and **http://localhost:7070/rest/products/Moto**

Open POSTMAN and give a POST request to http://localhost:7070/rest/products and pass the following product Json in the body :

```
{
    "name": "Sony   VAIO",
    "description": "SONY Laptop ",
    "originalPrice": 4000,
    "offer": {
      "offerPrice": 2000,
      "offerValidUntil": 1466098094993
    },
    "brandName": "Sony",
    "images": [
      "image15.jpeg",
      "image16.jpeg",
      "image17.jpeg"
    ]
  }
```

Observe that the product is created successfully.

Test by giving GET request to http://localhost:7070/rest/products. You should see 7 products

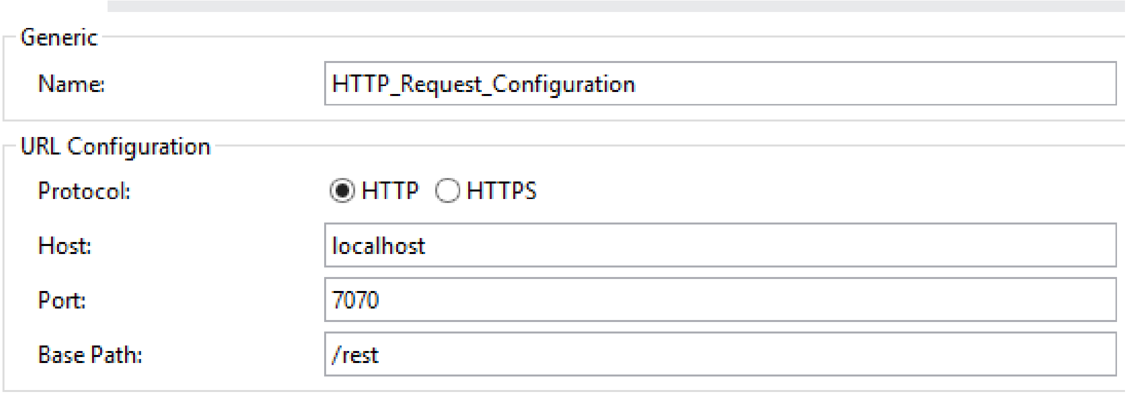
5) Now we want to consume this rest service

open 01-consuming-rest-start.xml in src/main/app

Configure a flow with http listener at port 8081 and path /products

Drap Http endpoint. Observe that this is Outbound (Http Requestor)

Configure the Http Request connector configuration as shown below :



Generic

Name: HTTP_Request_Configuration

URL Configuration

Protocol: ● HTTP ○ HTTPS

Host: localhost

Port: 7070

Base Path: /rest

Configure the path of http outbound endpoint as /products

Now run the application and give a request to http://localhost:8081/products. You should see all the products

6) Now modify the path of http outbound endpoint such that it will get the products whose name matches the name passed as **query** parameter **"productname"**

[Hint : use the url as /products/#[message.inboundProperties.'http.query.params'.productname] **]**

Deploy the application give request to **http://localhost:8081/products/?productname=Mac** and observe that you will get all the product whose name contains Mac

7) Now Drag a variable transformer between the Http endpoints and configure a flow variable with name "pname" value as /#[message.inboundProperties.'http.query.params'.productname]

8) Modify the URL of http outbound endpoint such that it picks the product name from flow variable "pname"

[Hint : use the url as products/#[flowVars.pname] **]**

Deploy the application give request to **http://localhost:8081/products/?productname=Mac** and observe that you will get all the product whose name contains Mac

9) Modify the path of inbound endpoint as /products/{productname} . In this path {productname} is the uri parameter

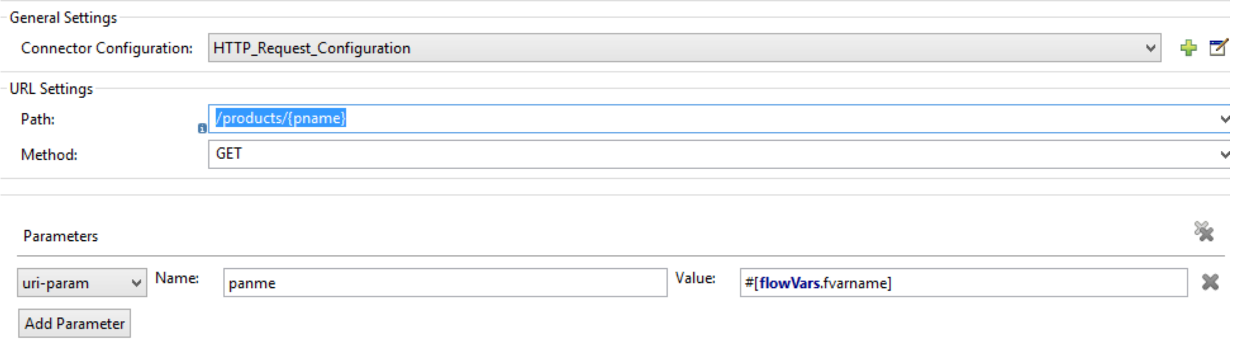
Now modify the flow variable value such that it extracts the product name from URI param

[Hint : #[message.inboundProperties.'http.uri.params'.productname]]

Deploy the application give request to **http://localhost:8081/products/Mac** and observe that you will get all the product whose name contains Mac

10)Now modify the outbount endpoint url as /products/{pname} .

Pass the value for pname by clicking on add parameter button as shown below :



General Settings

Connector Configuration: HTTP_Request_Configuration

URL Settings

Path: /products/{pname}

Method: GET

Parameters

uri-param Name: panme Value: #[flowVars.fvarname]

Deploy the application give request to **http://localhost:8081/products/Mac** and observe that you will get all the product whose name contains Mac

This is the end of the Exercise