



# INNOVATION HUB TRAINING

COURSE THREE HOMEWORK EXERCISES

APRIL 2024

## Contents

Introduction.....	3
What if You Have Questions?.....	3
Key Objectives .....	3
Training Exercise.....	4
Overview.....	4
Exercise One.....	5
Exercise Two.....	6
You've Finished the Exercises.....	7

## Introduction

Welcome to the InfoPlugin configuration homework for the MyWave Innovation Hub. This assignment aims to complement your knowledge from the 'Master the Wave' course and provide practical experience in using InfoPlugins to connect Wave Processes with systems like SAP Business One. To access the "Master the Wave" course, please email [training@mywave.ai](mailto:training@mywave.ai).

## What if You Have Questions?

If at any point you have questions or hit a dead end, please email us on [support@myWavesupport.zendesk.com](mailto:support@myWavesupport.zendesk.com) or view our FAQ's [here](#).

## Key Objectives

By completing this homework, you will:

1. **Understand the InfoPlugin Architecture:**
  - Learn how InfoPlugins (both HTTP and JAR) extend and enable Wave Process Connectivity.
2. **API Authentication Tokens:**
  - Configure the getSapAccessTokenStep InfoPlugin to authenticate and retrieve a session token for accessing the SAP Business One Service Layer API.
3. **Apply Variables in InfoPlugins:**
  - Utilise variables and parameters effectively within InfoPlugins.

# Training Exercise

Configuring InfoPlugins to Update Manufacturer Code in SAP Business One.

## Overview

Access the Innovation Hub Training Environment by clicking [here](#) to access the Innovation Hub Training Environment. Use login credentials sent to you in your welcome email.

If you haven't received your login credentials, please contact [training@mywave.ai](mailto:training@mywave.ai)

Once logged in:

- Navigate to your designated workspace using the dropdown menu on the left-hand side.
- If not already selected, click on **InfoPlugins** from the left-hand menu.

## Your Objective

Your goal is to learn how to configure and execute InfoPlugins to update the manufacturer code for an item in SAP Business One using specific variables.

## Background

This process involves using an InfoPlugin to obtain an access token and then updating the manufacturer code in SAP Business One. We will utilise the **Generic HTTP API** template and a premade JAR file for this homework.

## Prerequisites

- Basic understanding of HTTP methods (GET, PATCH).
- Familiarity with SAP Business One Service Layer API.

## Exercise One

In this exercise, we will configure an HTTP API InfoPlugin.

### Key Steps

- Navigate to InfoPlugins on the left-hand side of the Innovation Hub.
- Click **Create new InfoPlugin**, select **HTTP API** from the Available Implementations dropdown, and click **Next**.

### Get Access Token

1. Click on the **Generic HTTP API** template located in the “User Defined” Category in the API Library on the left of the screen. The template will appear in the Builder area after you have clicked.
2. Configure the template as follows:
  - **Template Name:** getSapAccessTokenStep. Note, the template name is different from the InfoPlugin name. The template name is the text in blue above the newly added template.
  - **baseUrl:** \${config.innovation\_hub\_integration\_service\_base\_url}
  - **path:** \${config.innovation\_hub\_integration\_service\_get\_access\_token\_url\_path}
  - **httpMethod:** GET
  - **Headers:** {"accept": "application/json", "customHeader": "customHeaderValue", "my-wave-account-id": "\${parameters.myWaveAccountId}"}
3. Any parameters referred to in the header need to be set up as Input Parameters. Add an Input Parameter called “myWaveAccountId” and set its Type to “string”.
4. The API call you have configured will retrieve an access token that will be used in the next step.

### Get Item Code:

1. Click again on the **Generic HTTP API** template in the API section. It will appear beneath your configured **getSapAccessTokenStep**.
2. Configure the second template as follows:
  - **Template Name:** getItemsMasterDataStep
  - **baseUrl:** \${config.sap\_b1\_base\_url}
  - **path:** \${config.sap\_b1\_get\_items\_url\_path}('\${parameters.itemCode}')
  - **httpMethod:** GET
  - Set headers as follows (remember to add { } where appropriate):
    - "accept ": "application/json"
    - "Authorization": "Bearer  
\${response.getSapAccessTokenStep.body.accessToken}"
    - "x-b1-companyid": "\${response.getSapAccessTokenStep.body.companyId}"
  - Note that the header "\${response.getSapAccessTokenStep.body.companyId}" refers to the response returned by the Get Access Token you configured above. This is how data may be passed between the APIs in an InfoPlugin.

3. To create your InfoPlugin, you need to give it a Name. Call it “GetItemsMasterData” in the InfoPlugin Name textbox in the top left of the screen. The Create button should then be enabled. Click Create. You will be taken back to the InfoPlugin Overview page.

Now that you have created your InfoPlugin, it will be available for use in conversations by any Conversation Creator. Refer to Course 2 or the Technical User Manual for how to do this.

## Exercise Two

In this exercise, we will configure a JAR-based InfoPlugin.

### Uploading the JAR

1. Note the JAR file included with this exercise, innovation-hub-infoplugin-2.0.0-SNAPSHOT.jar. This JAR contains four defined InfoPlugins. You can view their details in the infoPluginDescriptors.yml file inside the JAR. We need to upload this JAR to this system. This will create the four InfoPlugins.
2. Navigate to the **Repository** menu item in the Developer Tools menu on the left-hand side of the Innovation Hub.
3. Click **Upload New File**.
4. Drag and drop the innovation-hub-infoplugin-2.0.0-SNAPSHOT.jar into the upload area of the screen. You will see a success message, and the file name should appear on the Repository page.

### Configuring an InfoPlugin from the JAR

You can now configure the InfoPlugins contained in the JAR for use in the system.

1. Click on the InfoPlugins menu item in the Developer Tools Menu.
2. Click the Create New InfoPlugin button.
3. In the Available Implementations dropdown, you will see the four InfoPlugins that were in the JAR you just uploaded.
4. Click the first one in the list, “GetMasterDataItemPrefixInfoPlugin”.
5. Click Next.
6. You will be taken to the JAVA InfoPlugin configuration screen where you can configure this InfoPlugin.
7. Give it a name of your choice.
8. Add Response Properties of your choice and add a Constructor Arg of your choice.
9. Click Create.
10. On the InfoPlugin Overview screen, expand the JAR menu item and verify that the InfoPlugin you just created is visible.
11. If you now return to the HTTP API builder screen, you will see that the InfoPlugin you just created can now also be used in a chain of InfoPlugins. Feel free to experiment with this to get a feel for how this can work.

Now that you have created your InfoPlugin, it will be available for use in conversations by the Conversation Creator. Refer to Course 2 or the Technical User Manual for how to do this.

## You've Finished the Exercises

You have finished your homework exercises, well done!

So, what is next?

1. Once you have completed your Homework exercises, please email [training@myWave.ai](mailto:training@myWave.ai) with your name and Organization name to let us know that it is ready for review.
2. A MyWave expert will review your work and provided feedback and guidance on any areas to improve.

Thank you for joining us on the Training Course 'Master the Wave'.