

# Cloud Application Integration Services for Developers

## onDemand Lab Guide

Version: CAIS-R42-DEV-OD-202308



Informatica™  
University

(This page intentionally left blank)

Cloud Application Integration Services for Developers (onDemand)

Version: CAIS-R42-DEV-OD-202308

August 2023

Copyright (c) 1998–2023 Informatica LLC. All rights reserved.

A current list of Informatica trademarks is available on the web at <https://www.informatica.com/trademarks.html>.

This educational service, documentation and related software (collectively, "Materials") are provided under a separate license agreement containing restrictions on use and disclosure. No part of the Materials may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC. A current list of Informatica trademarks is available on the web at <https://www.informatica.com/trademarks.html>. Portions of Materials are subject to copyright held by third parties. Required third party notices are included with the Materials. The Materials are protected by patents as detailed at <https://www.informatica.com/legal/patents.html>. The information in these Materials is subject to change without notice. If you find any problems in these Materials, please report them to us in writing at Informatica LLC 2100 Seaport Blvd. Redwood City, CA 94063. Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA LLC PROVIDES THE INFORMATION IN THESE MATERIALS "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Use, duplication, or disclosure of the related software by the U.S. Government is subject to the restrictions set forth in the applicable software license agreement and as provided in DFARS 227.7202-1(a) and 227.7702-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (OCT 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14 (ALT III), as applicable.

The information in this educational service, materials, and documentation is subject to change without notice. If you find any problems in this educational service, materials, or documentation, please report them to us in writing.

Informatica, Informatica Platform, Informatica Data Services, PowerCenter, PowerCenterRT, PowerCenter Connect, PowerCenter Data Analyzer, PowerExchange, PowerMart, Metadata Manager, Informatica Data Quality, Informatica Data Explorer, Informatica B2B Data Transformation, Informatica B2B Data Exchange, Informatica On Demand, Informatica Identity Resolution, Informatica Application Information Lifecycle Management, Informatica Complex Event Processing, Ultra Messaging, and Informatica Master Data Management are trademarks or registered trademarks of Informatica LLC in the United States and in jurisdictions throughout the world. All other company and product names may be trade names or trademarks of their respective owners.

Portions of this educational service, materials, and/or documentation are subject to copyright held by third parties, including without limitation: Copyright © Adobe Systems Incorporated. All rights reserved. Copyright © Microsoft. All rights reserved. Copyright © Oracle. All rights reserved. Copyright @ the CentOS Project.

This Software is protected by U.S. Patent Numbers 5,794,246; 6,014,670; 6,016,501; 6,029,178; 6,032,158; 6,035,307; 6,044,374; 6,092,086; 6,208,990; 6,339,775; 6,640,226; 6,789,096; 6,820,077; 6,823,373; 6,850,947; 6,895,471; 7,117,215; 7,162,643; 7,243,110; 7,254,590; 7,281,001; 7,421,458; 7,496,588; 7,523,121; 7,584,422; 7,720,842; 7,721,270; and 7,774,791, international Patents and other Patents Pending.

DISCLAIMER: Informatica LLC provides this educational services, materials, and documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of non-infringement, merchantability, or use for a particular purpose. Informatica LLC does not warrant that this educational service, materials, documentation, or related software is error free. The information provided in this educational service, materials, documentation, and related software may include technical inaccuracies or typographical errors. The information in this educational service, materials, documentation and related software is subject to change at any time without notice.

## Document Conventions

This guide uses the following formatting conventions:

| If you see...          | It means...  | Example  |
|------------------------|--|--|
| >                      | Indicates a sub menu to navigate to.   | Click Repository > Connect.<br>In this example, you should click the Repository menu or button and choose Connect. |
| <b>boldfaced text</b>  | Indicates text you need to type or enter.                                    | Click the Rename button and name the new source definition <b>S_EMPLOYEE</b> .                                     |
| UPPERCASE              | Database tables and column names are shown in all UPPERCASE.                 | T_ITEM_SUMMARY   |
| <i>italicized text</i> | Indicates a variable you must replace with specific information.             | Connect to the Repository using the assigned <i>login_id</i> .   |
| <b>Note:</b>           | The following paragraph provides additional facts.                           | <b>Note:</b> You can select multiple objects to import by using the Ctrl key.                                      |
| <b>Tip:</b>            | The following paragraph provides suggested uses or a Velocity best practice. | <b>Tip:</b> The m_ prefix for a mapping name is...   |

## **Other Informatica Resources**

In addition to the student and lab guides, Informatica provides these other resources:

- Documentation and Knowledge Base
- Global Customer Support
- Professional Certification

### **Accessing Documentation and Knowledge Base**

To get the latest documentation and Knowledge Base for your product, go to

<https://network.informatica.com>

### **Contacting Global Customer Support**

You can contact a Customer Support Center by telephone or through the Online Support. Online Support requires a username and password. You can request a username and password at

<https://www.informatica.com/services-and-training/support-services/contact-us.html>

### **Obtaining Informatica Professional Certification**

You can take and pass exams provided by Informatica to obtain Informatica Professional Certification. For more information, go to

<https://www.informatica.com/services-and-training/certification.html>

# Table of Contents

## **Pre-requisite Labs**

|  |    |
|--|----|
| Lab 0-1: Create an IICS Account .....        | 1  |
| Lab 0-2: Install IICS Secure Agent.....      | 8  |
| Lab 0-3: Configure SQL Server Database ..... | 29 |
| Lab 0-4: Configure Kafka Setup .....         | 36 |

## **Module 3: Create a Basic Process**

|   |    |
|---|----|
| Lab 3-1: Create a Basic Process to Display User Input ..... | 48 |
|---|----|

## **Module 4: Process Objects**

|                                       |    |
|---------------------------------------|----|
| Lab 4-1: Create Process Objects ..... | 61 |
|---------------------------------------|----|

## **Module 5: Service Connectors**

|  |     |
|--|-----|
| Lab 5-1: Create a Service Connector.....                     | 70  |
| Lab 5-2: Create a Service Connector from WSDL .....          | 86  |
| Lab 5-3: Create and Test a Swagger File .....                | 95  |
| Lab 5-4: Create a Service Connector from a Swagger File..... | 106 |
| Lab 5-5: Create a Service Connector from OpenAPI 3.0 .....   | 114 |

## **Module 6: Data Access Service Connectors**

|  |     |
|--|-----|
| Lab 6-1: Create a Data Access Service Connector Using Form ..... | 122 |
|--|-----|

## **Module 7: App Connections**

|   |     |
|---|-----|
| Lab 7-1: Create an App Connection Using Service Connector ..... | 131 |
| Lab 7-2: Create a JDBC Connection.....                          | 135 |
| Lab 7-3: Create a Customer and Order Details Process.....       | 140 |
| Lab 7-4: Create a File Connection .....                         | 155 |
| Lab 7-5: Create a Salesforce Connection .....                   | 163 |
| Lab 7-6: Create a Kafka Connection .....                        | 171 |

## **Module 8: Managing and Migrating Assets**

|                               |     |
|-------------------------------|-----|
| Lab 8-1: Export an Asset..... | 197 |
|-------------------------------|-----|

## **Module 9: Using XQuery in CAI**

|   |     |
|---|-----|
| Lab 9-1: Create a Process Using Lists Function of XQuery..... | 202 |
|---|-----|

## **Module 10: Adding Web Services to a Process**

|   |     |
|---|-----|
| Lab 10-1: Invoke a Synchronous Web Service Call.....    | 235 |
| Lab 10-2: Invoke an Asynchronous Web Service Call ..... | 260 |

**Module 11: Fault Handling**

|  |     |
|--|-----|
| Lab 11-1: Handle Sub-process Failures..... | 286 |
| Lab 11-2: Handle Credential Failures.....  | 293 |

**Module 12: Introduction to Guides Designer**

|                                      |     |
|--------------------------------------|-----|
| Lab 12-1: Create a Simple Guide..... | 304 |
|--------------------------------------|-----|

**Module 13: Monitoring**

|   |     |
|---|-----|
| Lab 13-1: Create and Use URN Mapping. ....                                    | 318 |
| Lab 13-2: Create a New Process Schedule. ....                                 | 329 |
| Lab 13-3: Start and Stop an Event Source in a Listener-based Connection. .... | 335 |

**Module 14: API Management**

|                             |     |
|-----------------------------|-----|
| Lab 14-1: API Manager. .... | 346 |
|-----------------------------|-----|

**Module 15: CAI and CDI Integration**

|   |     |
|---|-----|
| Lab 15-1: Create a Flat File Connection.....              | 349 |
| Lab 15-2: Create a Basic Mapping and a Mapping Task. .... | 354 |
| Lab 15-3: Invoke a Mapping in CAI. ....                   | 362 |

**Module 17: CAI Live Project Walk-Through**

|  |     |
|--|-----|
| Lab 17-1: CAI Live Project- Order Management. .... | 367 |
|--|-----|

# Pre-requisite Labs

## Lab 0-1: Create an IICS Account

### Overview:

To use the different services that are available in the IICS platform, users must have a valid IICS account. The trial account is valid for 90 days.

This document lists the steps to create an IICS trial account to use in the onDemand training course. The course content and Labs are activated at the time of registration and are available for 90 days or until all lab hours are consumed.

### Objective:

- Create an IICS trial account
- Copy Lab Prep Files

### Important:

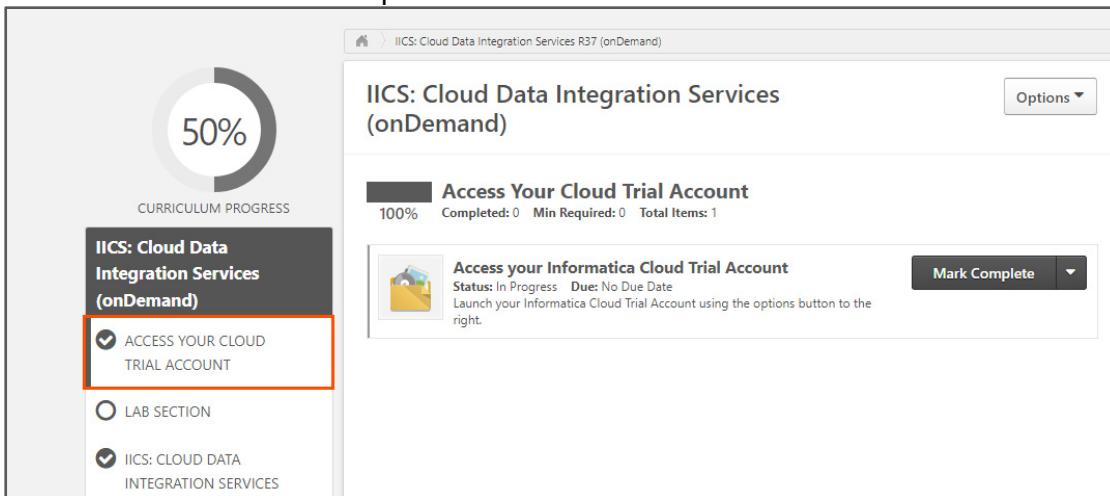
The screenshots used in this lab guide is for Cloud Data Integration course. The name of course may differ depending upon the enrolled course. However, the steps for creating the IICS account will remain the same.

---

## Tasks

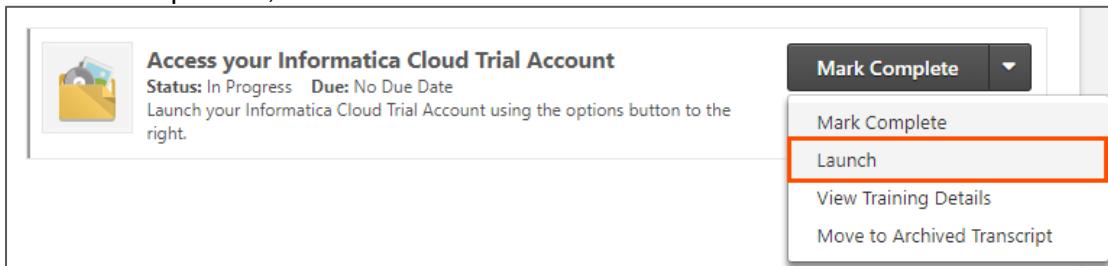
### Creating an IICS account

1. From CSOD (the page where you accessed the course), click on the **ACCESS YOUR CLOUD TRIAL ACCOUNT** option.



The screenshot shows the CSOD interface for the 'IICS: Cloud Data Integration Services (onDemand)' course. On the left, there's a curriculum progress bar at 50% completion. Below it, a sidebar lists three options: 'IICS: Cloud Data Integration Services (onDemand)', 'LAB SECTION', and 'IICS: CLOUD DATA INTEGRATION SERVICES'. The first option is highlighted with a red box around the 'ACCESS YOUR CLOUD TRIAL ACCOUNT' link. To the right, the main content area displays the title 'IICS: Cloud Data Integration Services (onDemand)' and a section titled 'Access Your Cloud Trial Account' with a status of '100%' and 'Completed: 0 Min Required: 0 Total Items: 1'. Below this is a task card for 'Access your Informatica Cloud Trial Account' with a status of 'In Progress' and 'Due: No Due Date'. A 'Mark Complete' button is visible to the right of the task card.

- From the drop-down, click **Launch**.

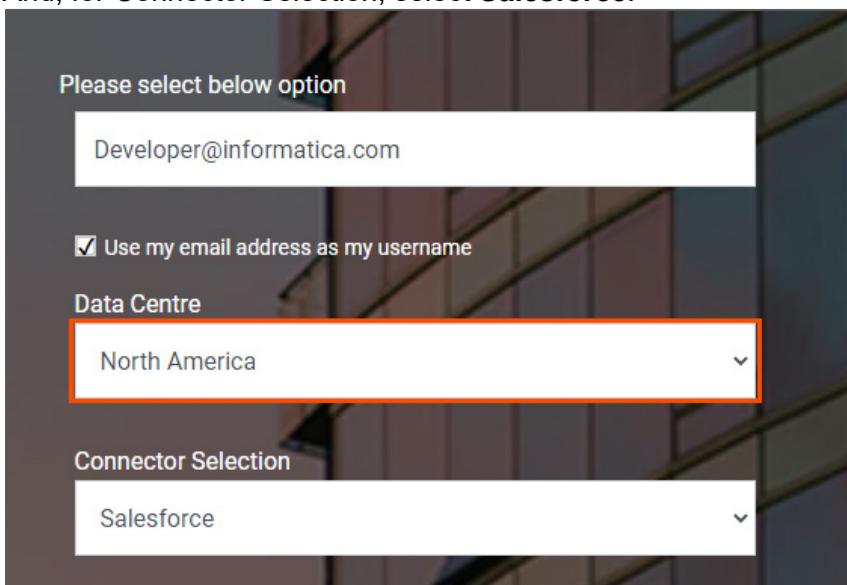


**Note:** If the option menu is labeled anything other than **Launch**, click the down arrow to the right and click **Launch**.

- A new University Trial window opens in a new tab. Use the new window to sign up for an IICS trial account.

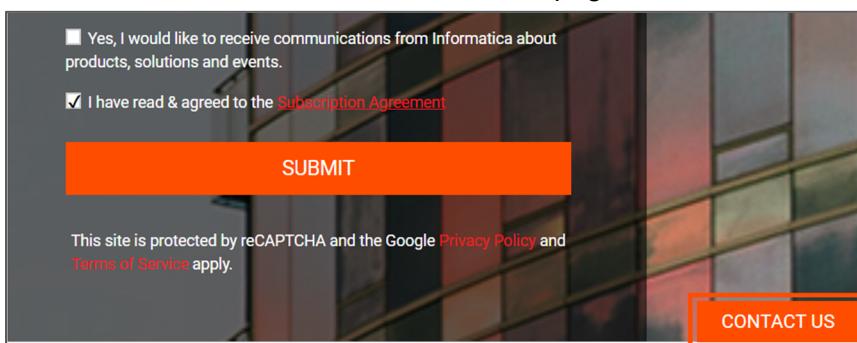
- On the signup page, enter a valid email id and other required information.

**Note:** During account creation, you must select the Data Centre as **North America**. And, for Connector Selection, select **Salesforce**.

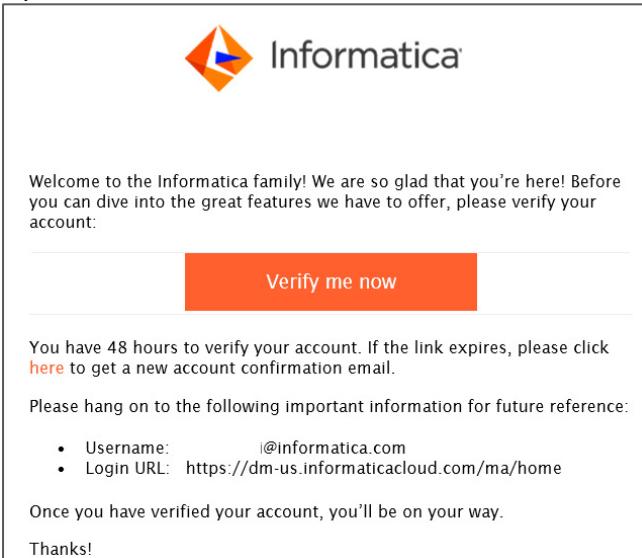


- Once the required information is entered, select the "**I have read & agreed to the Subscription Agreement**" checkbox and click **SUBMIT**.

**Note:** If you are unable to register your email-id while creating the account, use the **CONTACT US** button at the bottom of the page to contact Informatica Support.

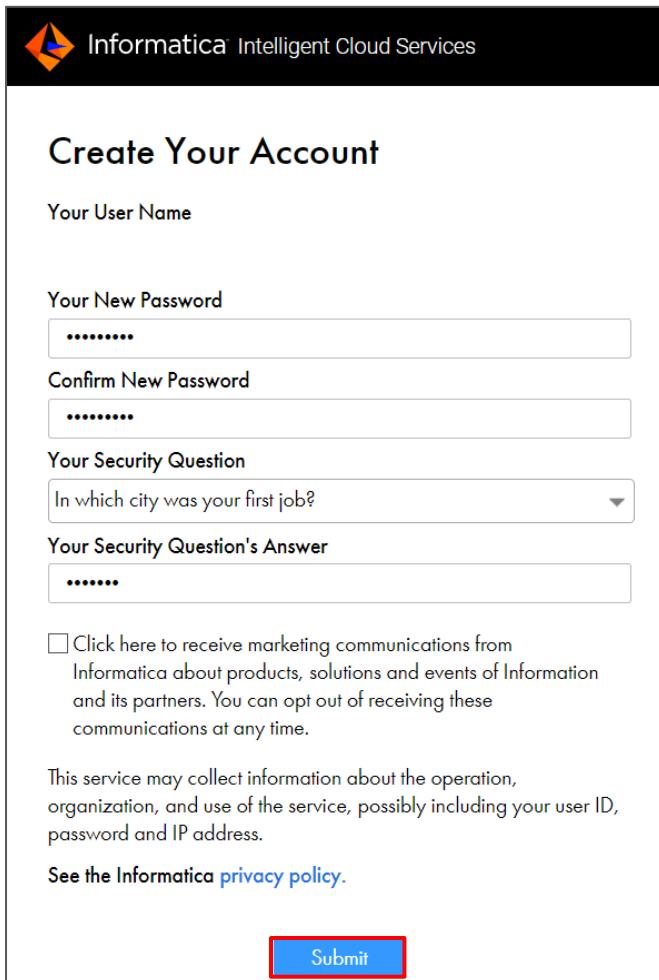


6. After you submit the request, in a few seconds you will get an email from [admin@informaticacloud.com](mailto:admin@informaticacloud.com) in your registered mailbox.
7. Open the email and to confirm the account creation, click on **Verify me now** link.

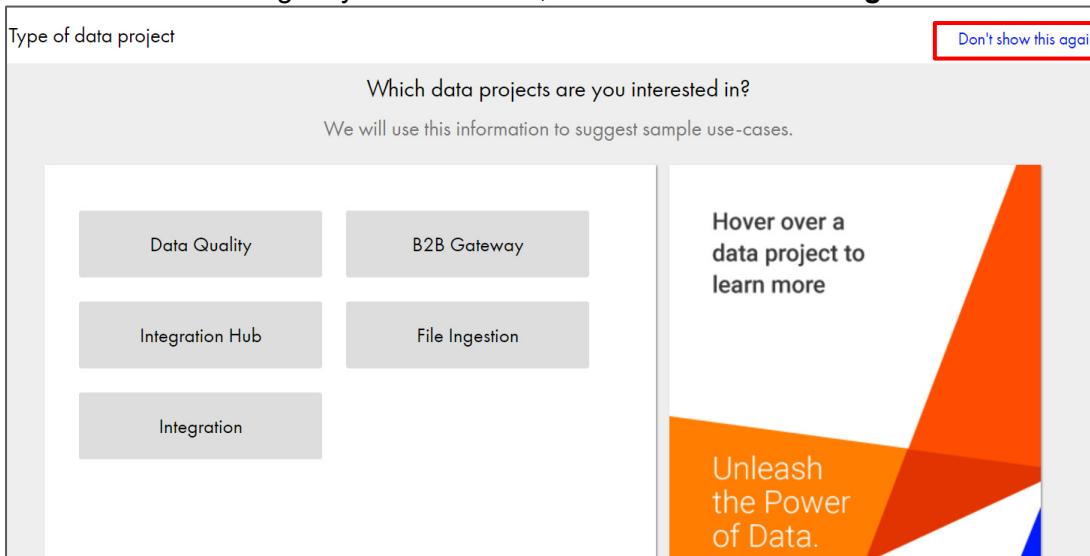


**Note:** You can bookmark the URL link to access the IICS login page for future access.

8. In the password setup page, enter a suitable password and a security question for the account.

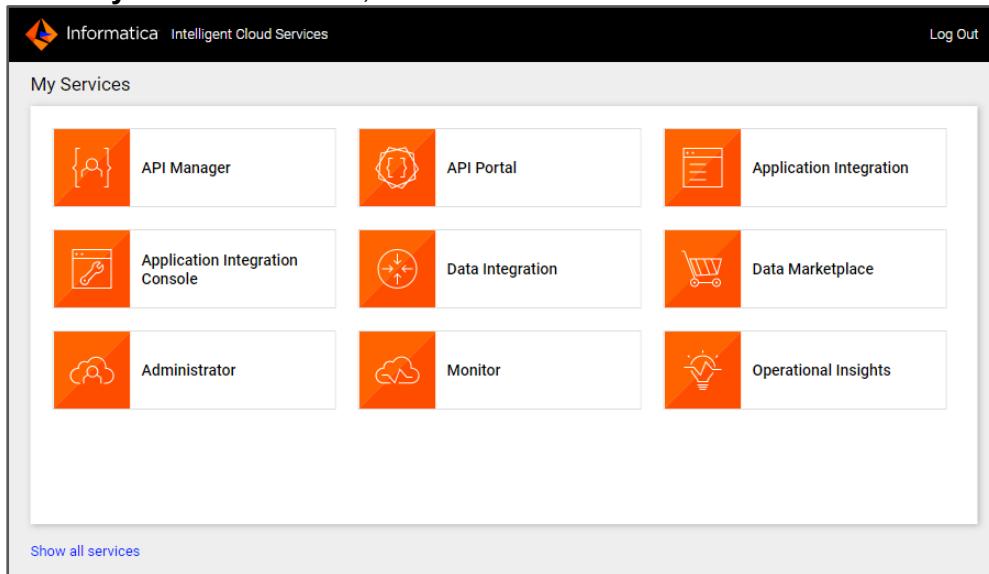
9. Click **Submit**.

The screenshot shows the 'Create Your Account' page. It includes fields for 'Your User Name' (with placeholder '\*\*\*\*\*'), 'Your New Password' (with placeholder '\*\*\*\*\*'), 'Confirm New Password' (with placeholder '\*\*\*\*\*'), 'Your Security Question' (dropdown menu showing 'In which city was your first job?'), 'Your Security Question's Answer' (text input field with placeholder '\*\*\*\*\*'), and a checkbox for marketing communications. Below the checkbox is a note about data collection and a link to the privacy policy. A red box highlights the 'Submit' button at the bottom.

10. In the Source and target systems window, click **Don't show this again**.

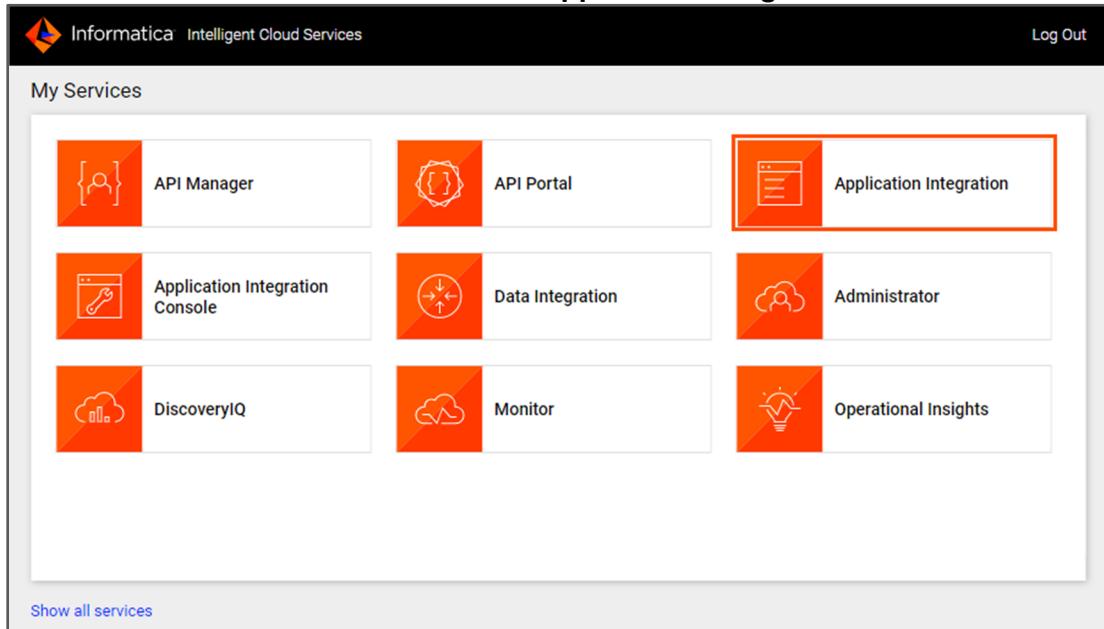
The screenshot shows a window titled 'Type of data project'. It asks 'Which data projects are you interested in?' and explains that it will use this information to suggest sample use-cases. It lists several data project categories: Data Quality, B2B Gateway, Integration Hub, File Ingestion, and Integration. A red box highlights the 'Don't show this again' button in the top right corner. To the right, there is a graphic with text: 'Hover over a data project to learn more' and 'Unleash the Power of Data.'

11. In the **My Services** window, notice the list of available services.



Show all services

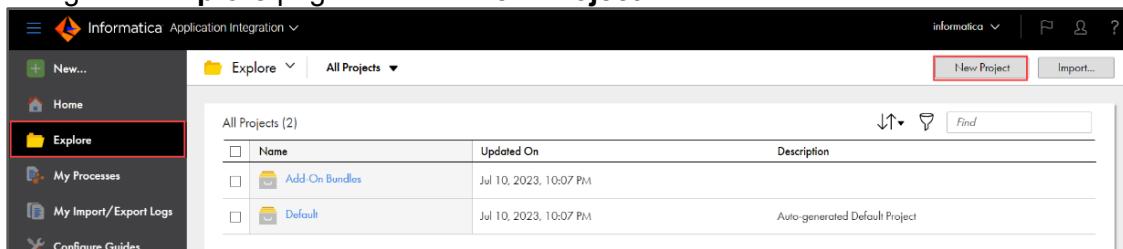
12. From the list of available services, select **Application Integration**.



Show all services

The **Application Integration Home** page appears.

13. Navigate to **Explore** page and click **New Project**.



| Name           | Updated On             | Description                    |
|----------------|------------------------|--------------------------------|
| Add-On Bundles | Jul 10, 2023, 10:07 PM |                                |
| Default        | Jul 10, 2023, 10:07 PM | Auto-generated Default Project |

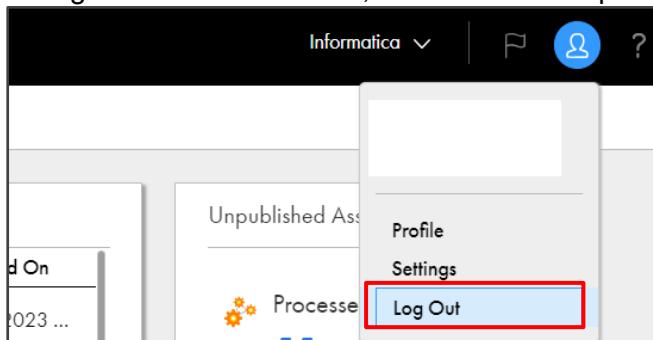
14. In the Name field, enter **S\_XX\_FirstName**.

**Note:** Here, replace XX with your initials (or with your Date of Birth). For example, if your name is John Smith, replace XX with JS.

15. Click **Save**.



16. To log out from the account, click on the user profile and then click **Log Out**.



### Copy Lab Prep Files

17. Download the Prep-Files folder provided to you and paste the folder on your desktop.  
Rename it to **CAI Lab Prep Files**.

### Create Folder

18. In the C drive, create a new folder and name it as **CAI**. Inside the CAI folder, create a new folder and name it as **DataFiles**.

## Important

1. For every asset that you create, make sure the location of the asset is set to your working folder to avoid saving assets to multiple projects.
  2. The assets in this training are given a specific naming convention that has 'XX' and 'Firstname' in them. Here, replace XX with your name initials or date of birth and Firstname with your first name. Note that these are not mandatory naming conventions and this is just to maintain consistency across the training and assets.
  3. If the IICS Org is idle for 30 minutes, you will be logged out of the Org. Hence, you must save your work from time to time.
  4. The Secure Agent can shut down unexpectedly while performing the labs. In this case, navigate to the Windows **Start** option of the system and select **Services**. Scroll down to Informatica Cloud Secure Agent and click **Restart**.
  5. If at any time, the browser hangs and the desired output is not obtained (asset not getting published, windows do not open, and so on), try closing the browser window and opening a fresh one.
  6. If you do not view any of the previously created assets, log out of IICS, and clear the browser cache. After clearing the browser cache, login back into IICS and perform the steps again to select the asset.
- 

*This concludes the lab.*

## Pre-requisite Labs

### Lab 0-2: Install IICS Secure Agent

#### Overview:

The secure agent is a lightweight, self-upgrading program that runs inside your network. It is responsible for moving data from the source to a target.

IICS secure agent runs all tasks and enables a secure communication between your organization and Informatica Cloud.

There are certain system requirements that must be met to install the secure agent. This document lists the system requirements and the installation steps.

#### Objective:

- List the system requirements for installing IICS secure agent
- Install the secure agent
- Rename the secure agent
- Discuss common troubleshooting techniques

#### Duration:

60 minutes

---

#### System requirements for installing IICS secure agent:

1. You must have a multi-core CPU system (at least 2 cores and 4 CPUs/Logical Processors).
2. It is recommended that your system has a minimum of 16 GB RAM.
3. Your system must have at least 100 GB disk space to run tasks and store caches and logs with success.
4. Your system must have a web browser.  
**Note:** Supported browsers are – Internet Explorer version 11, Google Chrome version 62.0 and higher, and Mozilla Firefox (64-bit) version 56.0 and higher.
5. You can install the secure agent on a Windows 64-bit system or on a Linux 64-bit system.
6. The secure agent uses the HTTPS port 443 to connect to the internet. You must configure the firewall to allow traffic to pass over port 443.
7. Do not install the secure agent on the same system that is running Informatica PowerCenter Domain server, as it might cause LIB/PATH conflicts and performance issues.
8. Do not install more than one secure agent on the same system, as this can cause agent ID conflicts during its registration under Informatica Cloud Point of Deployments (PODs).
9. If your company uses a proxy server, you must configure proxy server settings when you install the secure agent for the first time.

**Note:** If you face any issues in installing secure agent, refer to the **Troubleshooting**

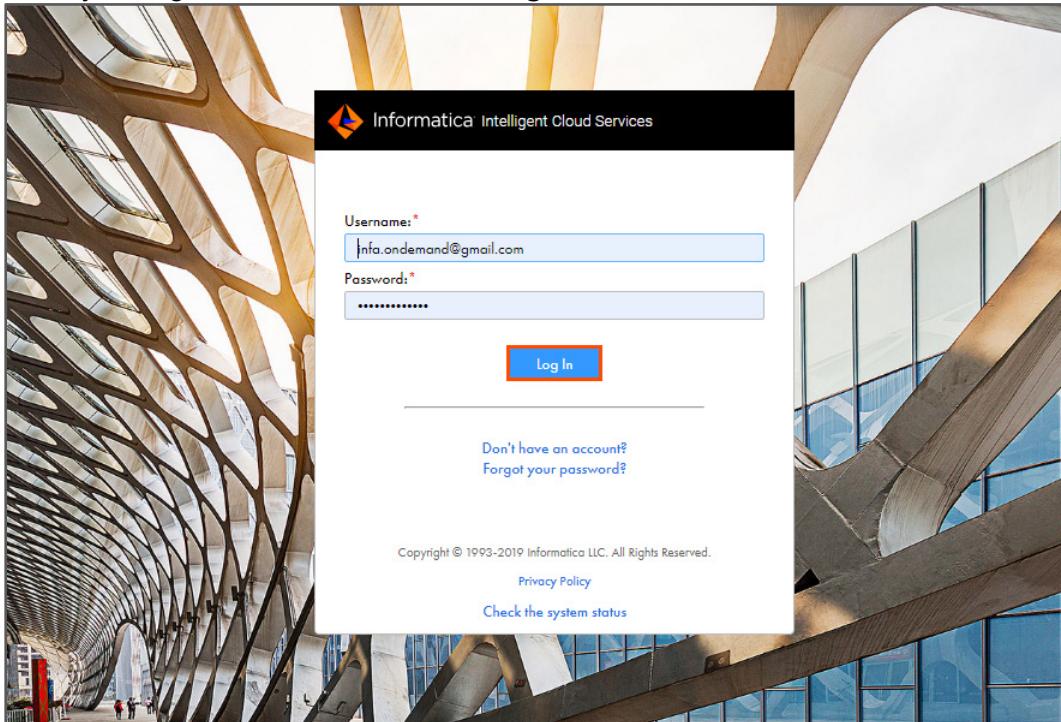
**Prerequisite Installations** document provided with this lab. If the issue is not listed in the document, contact Informatica Support team on [support@informatica.com](mailto:support@informatica.com) and include your organization ID and contact information.

## Download the IICS Secure Agent:

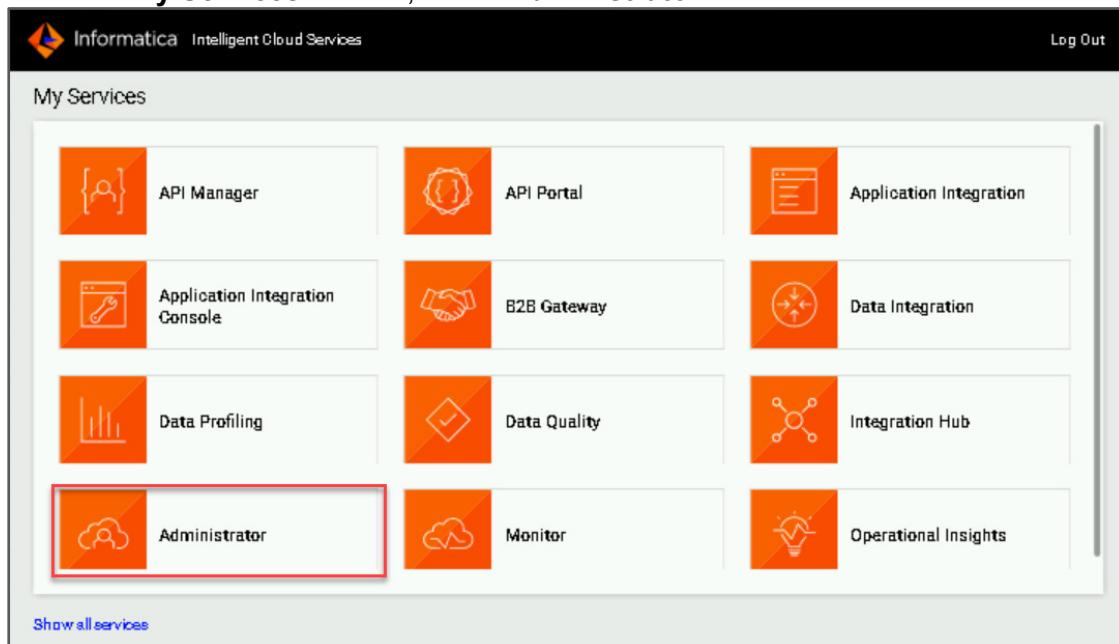
1. In your browser, open the following link to log into IICS.  
<https://dm-us.informaticacloud.com/identity-service/home>

**Note:** If you miss out on bookmarking this link earlier, bookmark it now for the further use.

2. Enter your login credentials and click **Log In**.

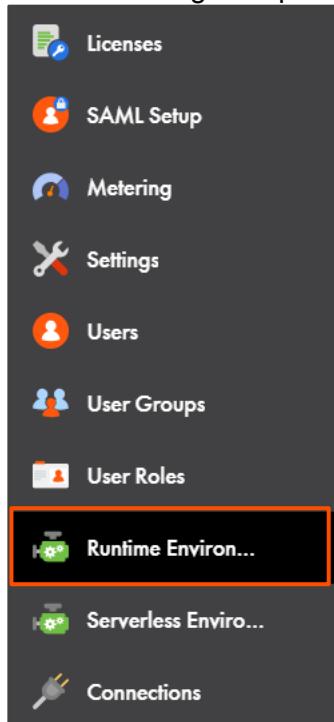


3. From the **My Services** window, select **Administrator**.



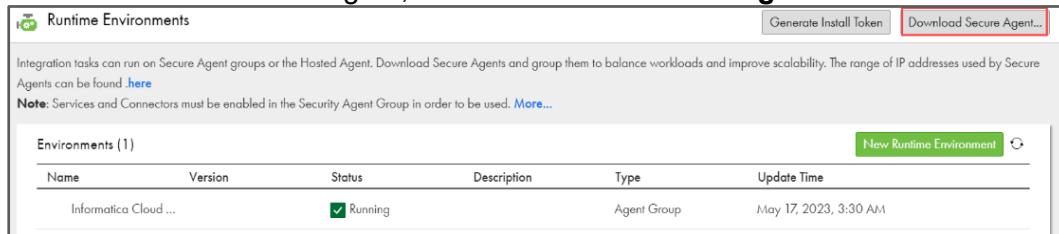
The Organization page appears.

4. From the Navigation pane, select **Runtime Environments**.



**Note:** If you get a pop-up message that states, "To begin using Integration At Scale,...", close the pop-up message.

5. To download the secure agent, click **Download Secure Agent**.



Runtime Environments

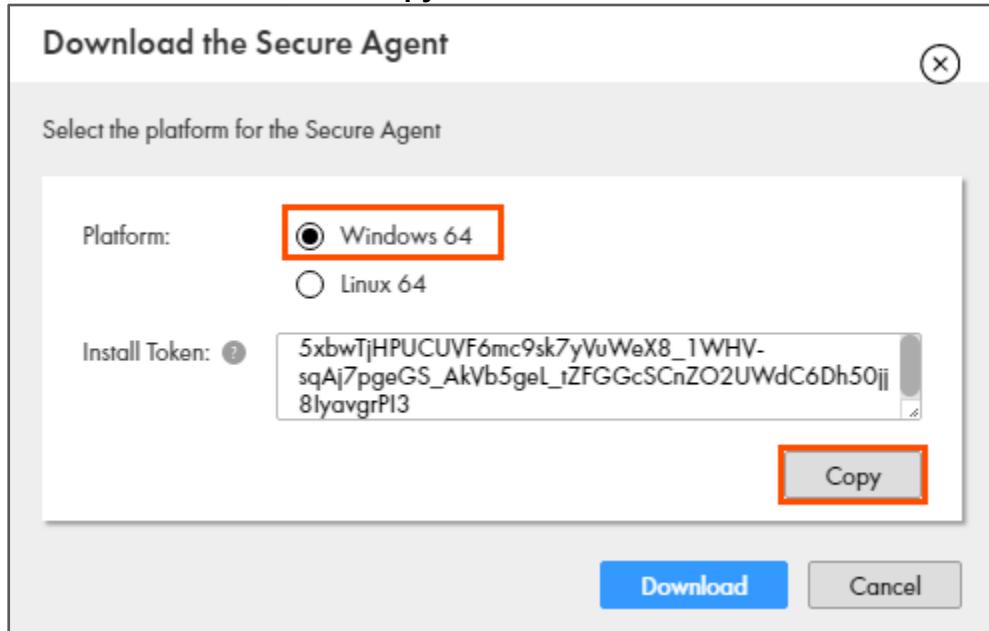
Integration tasks can run on Secure Agent groups or the Hosted Agent. Download Secure Agents and group them to balance workloads and improve scalability. The range of IP addresses used by Secure Agents can be found [here](#).

**Note:** Services and Connectors must be enabled in the Security Agent Group in order to be used. [More...](#)

| Environments (1)      |         |   |             |             |                       |
|-----------------------|---------|---|-------------|-------------|-----------------------|
| Name                  | Version | Status                                      | Description | Type        | Update Time           |
| Informatica Cloud ... |         | <input checked="" type="checkbox"/> Running |             | Agent Group | May 17, 2023, 3:30 AM |

**Generate Install Token** **Download Secure Agent...**

6. From the **Download the Secure agent** window, select the desired platform, and from the Install Token field, click **Copy**.



**Download the Secure Agent**

Select the platform for the Secure Agent

Platform:  Windows 64  Linux 64

Install Token: [?](#) 5xbwTjHPUCUVF6mc9sk7yVuWeX8\_1WHV-sqAj7pgeGS\_AkVb5geL\_tZFGGcSCnZO2UWdC6Dh50jj8IyavgrPI3

**Copy**

**Download** **Cancel**

7. Paste the copied **Install Token** in a text file.

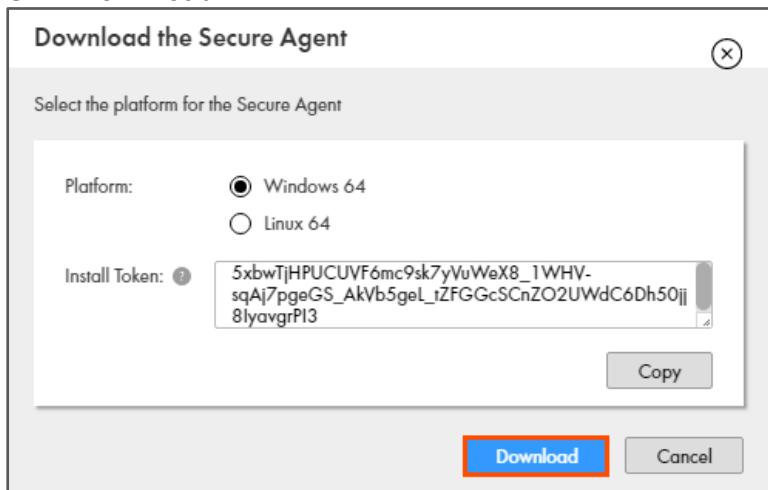


Untitled - Notepad

File Edit Format View Help

5xbwTjHPUCUVF6mc9sk7yVuWeX8\_1WHV-sqAj7pgeGS\_AkVb5geL\_tZFGGcSCnZO2UWdC6Dh50jj8IyavgrPI3

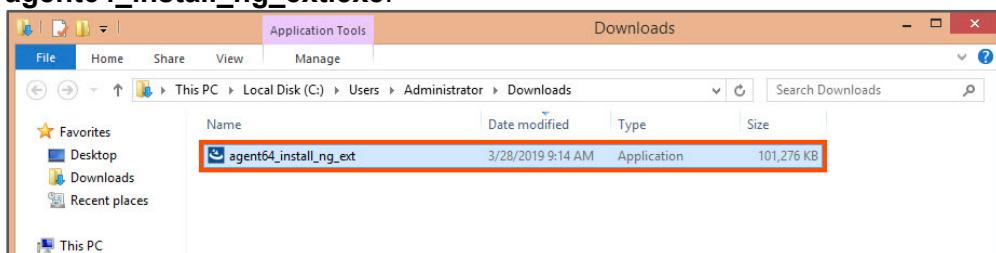
## 8. Click Download.



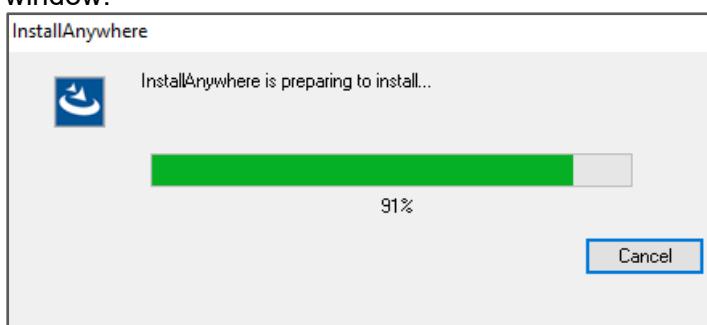
**Note:** Informatica Cloud Secure Agent for the selected platform will download on your machine. In this lab, you will install the secure agent on a Windows platform.

**Install the Secure Agent on your local machine:**

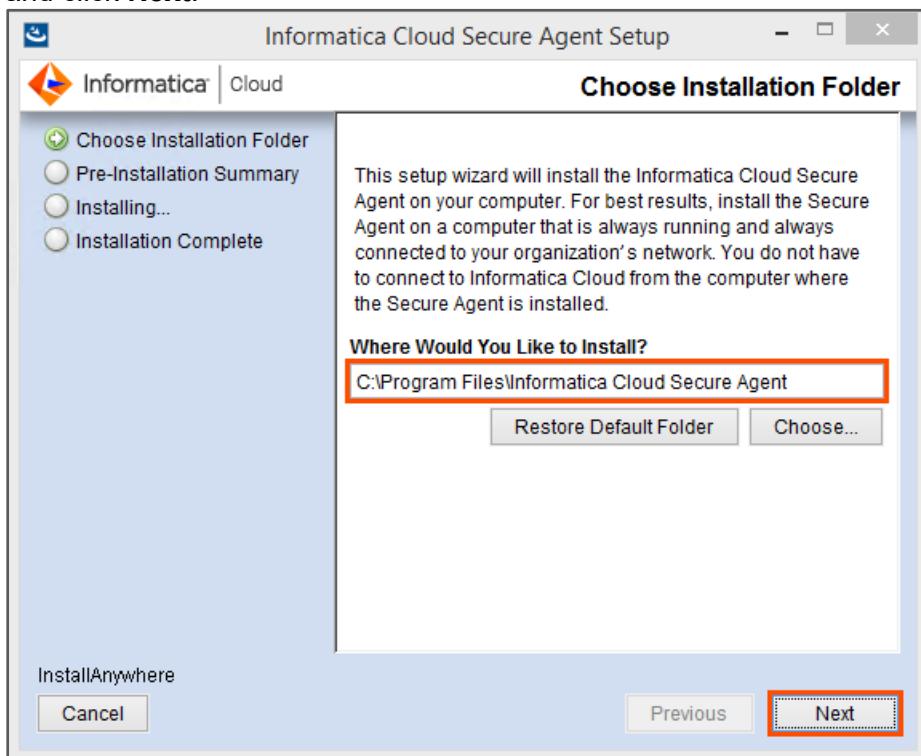
9. Navigate to the Downloads directory on your local machine and locate the agent installation file.
10. To install the secure agent, double-click the executable file **agent64\_install\_ng\_ext.exe**.



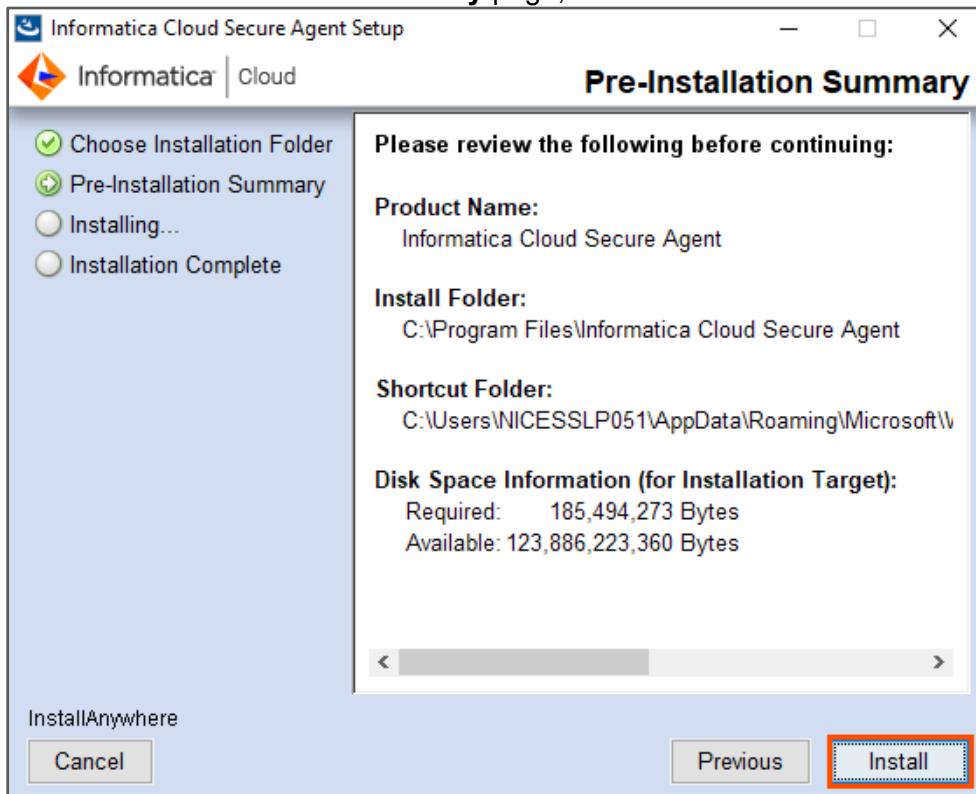
**Note:** When the secure agent initiates the installation, it displays the **InstallAnywhere** window.



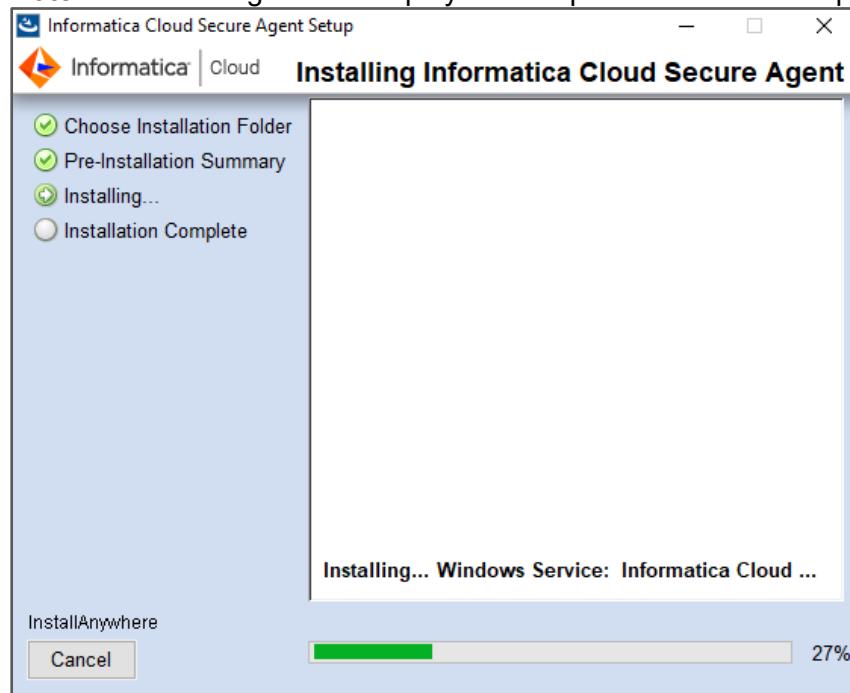
11. From the **Choose Installation Folder** window, retain the default installation location, and click **Next**.



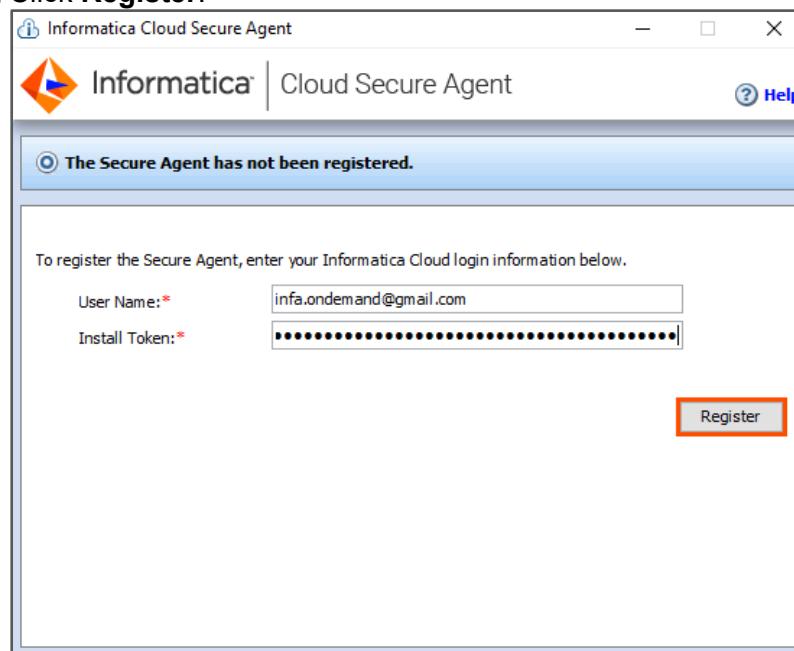
12. From the **Pre-Installation Summary** page, click **Install**.



**Note:** The following screen displays the steps in the installation process.

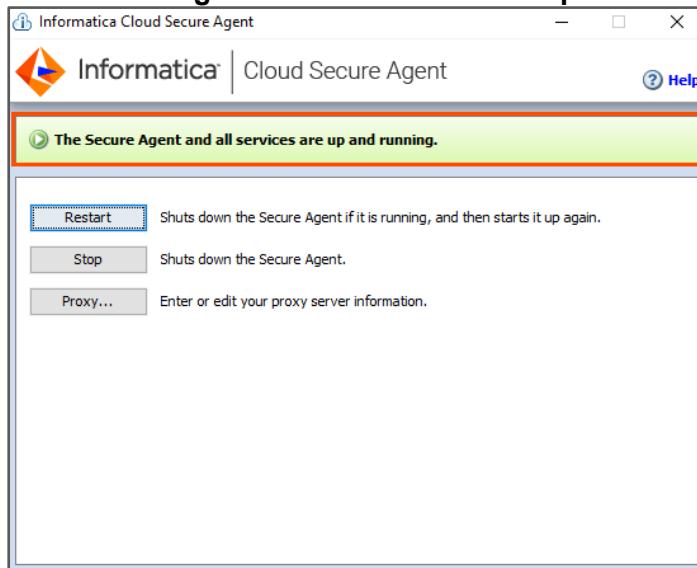


13. After the installation process is complete, the secure agent registration page appears with the message **The Secure Agent has not been registered**.  
**Note:** If the secure agent window doesn't open, from the desktop, right click the Informatica Cloud Secure Agent shortcut and click '**Run as Administrator**'.
14. To register your secure agent, enter the IICS username and paste the copied Install Token in the Install Token field.
15. Click **Register**.



**Note:** After successful registration, the secure agent will download the necessary files for the connectors. This process will take 20 to 30 minutes.

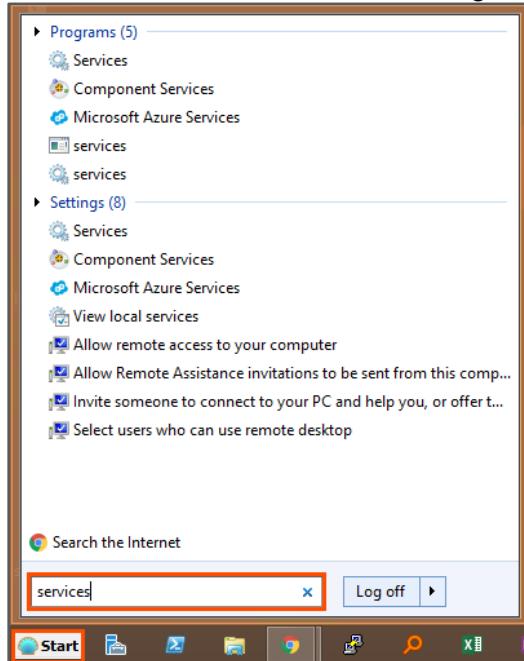
16. When the configuration of the secure agent services is in progress, it displays **Secure Agent Core is running. (Not all services are ready.)** message.
17. After the secure agent configures all the services, the agent status message changes to **The Secure Agent and all services are up and running.**



**Note:** You must provide Administrator rights to your secure agent.

#### Setting Administrative rights for Secure Agent:

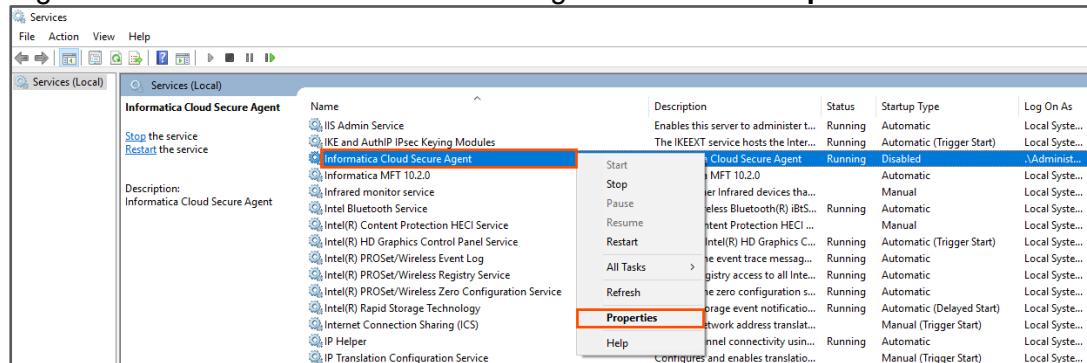
18. From the windows **Start** menu, navigate to **services**.



**Note:** The services window appears.

19. From the services page, select **Informatica Cloud Secure Agent**.

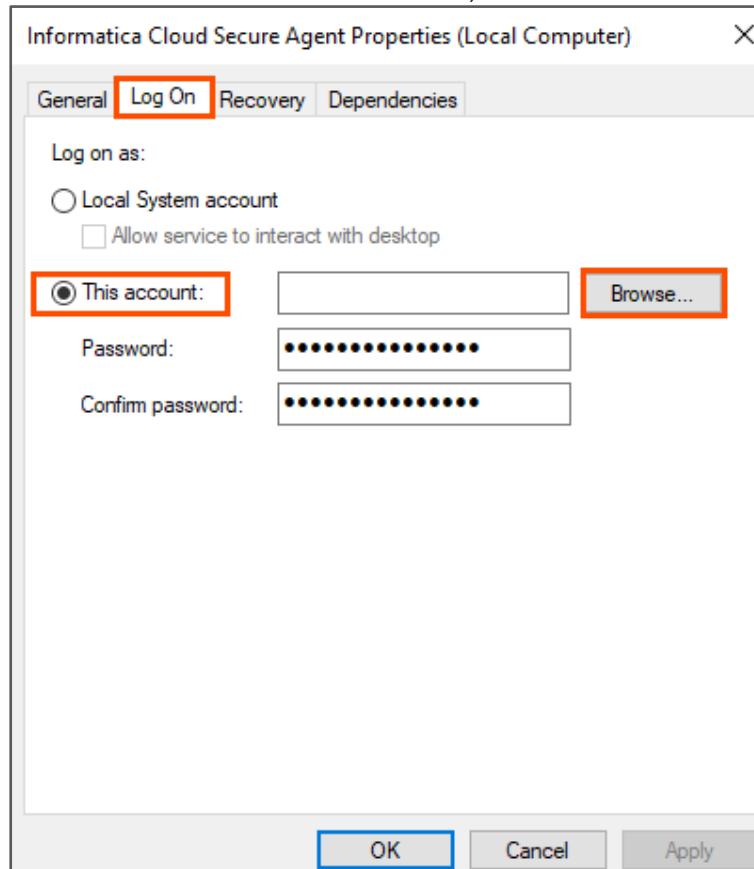
20. Right-click on Informatica Cloud Secure Agent and select **Properties**.



**Note:** A new properties window appears.

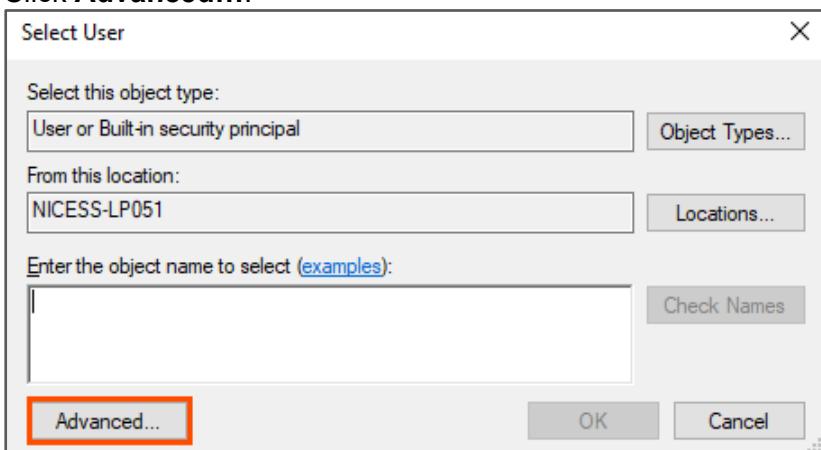
21. Navigate to the **Log On** tab and select **This account**.

22. To select the Administrator account, click **Browse**.



**Note:** The Select User window appears.

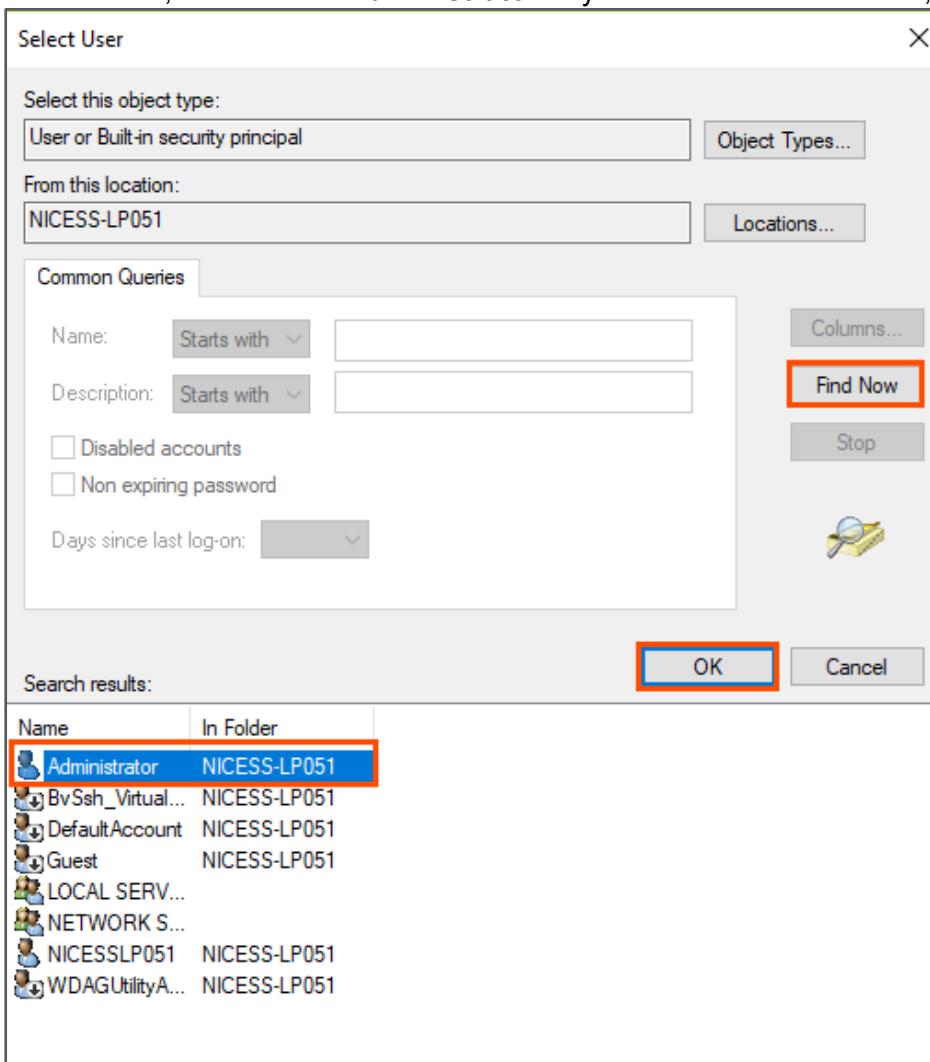
23. Click **Advanced....**



24. From the Select User window, select **Find Now**.

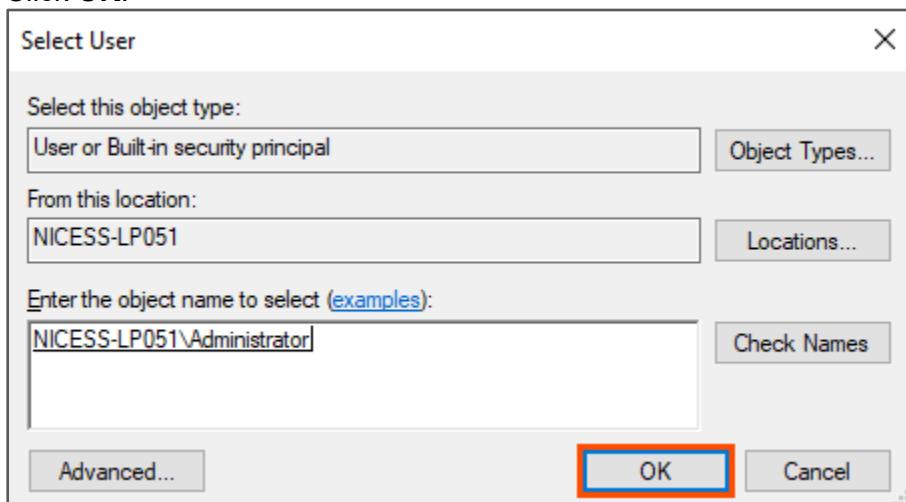
**Note:** A list of available users account appears. Indicative user account selected here for reference.

25. From the list, select either **Administrator** or your current windows user, and click **OK**.

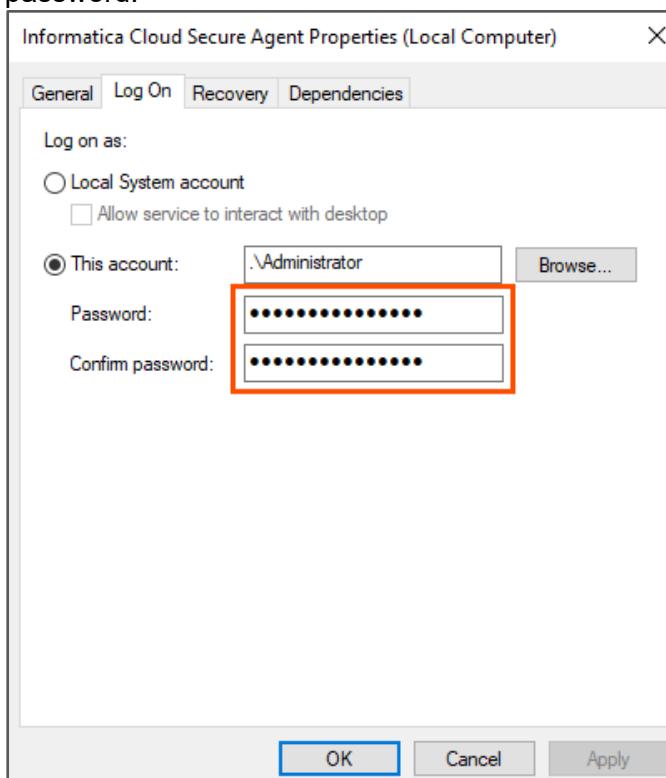


**Note:** You must select your system admin user in the above step. The user name varies from person to person.

26. Click **OK**.

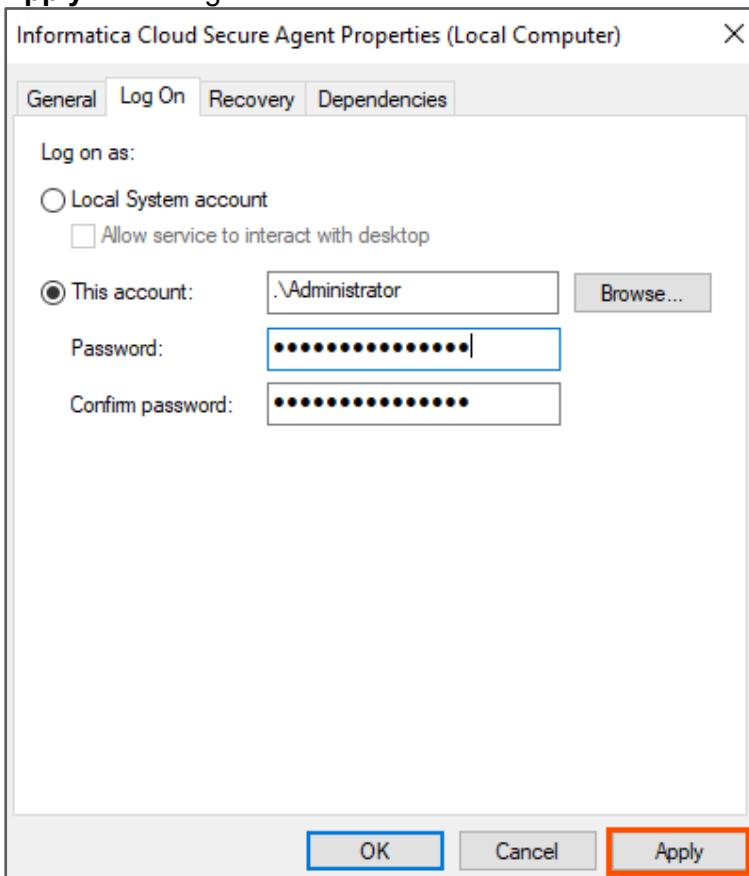


27. In the Password and Confirm Password fields, enter your system's or Administrator user password.



**Note:** Here, enter your system/window user's login password.

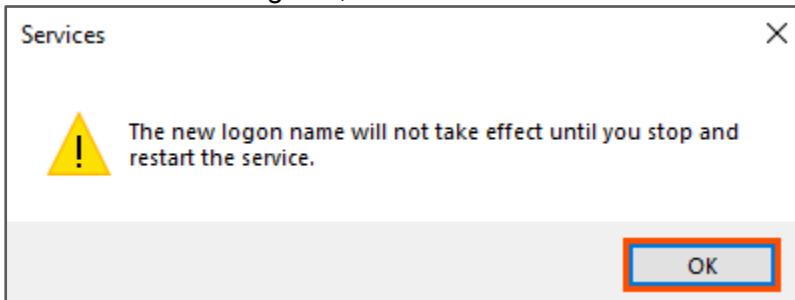
## 28. Apply the changes.



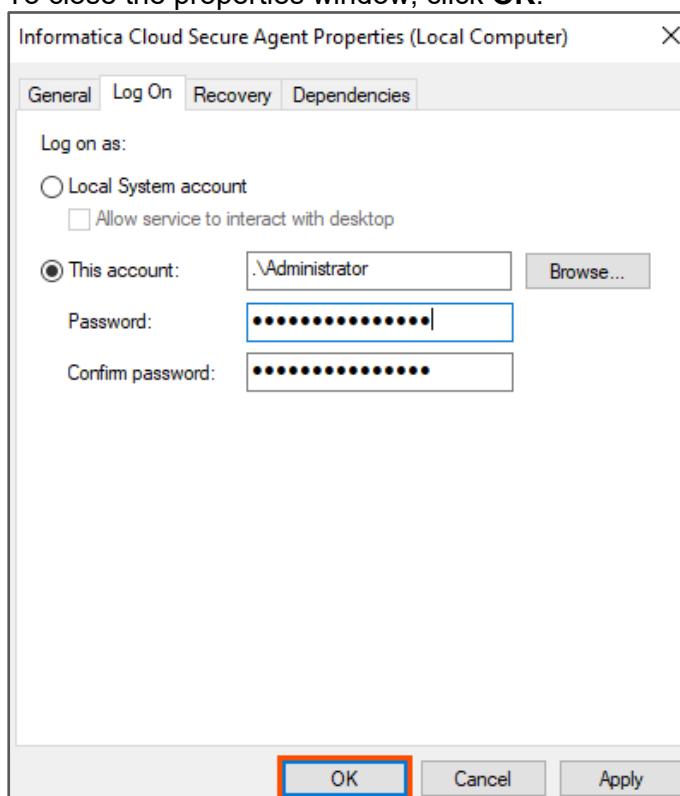
## 29. If you are prompted with the following Services window tab, click OK.



## 30. In the Services dialog box, click OK.



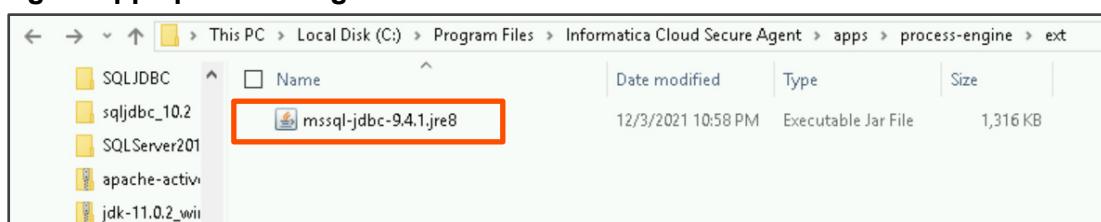
31. To close the properties window, click **OK**.



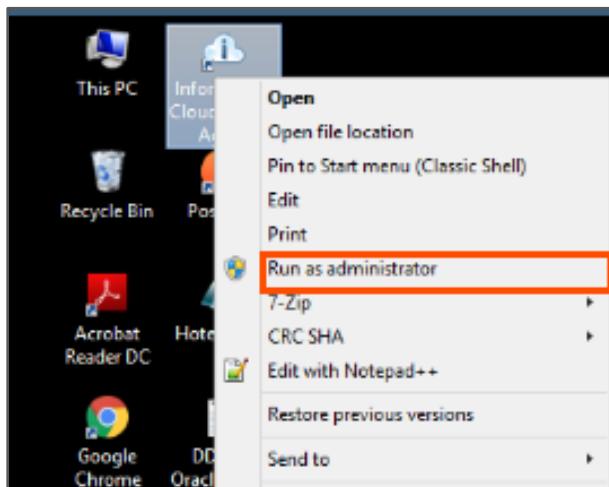
**Note:** After this, you must restart the secure agent and your system on which the secure agent is installed.

### Copy Jar File

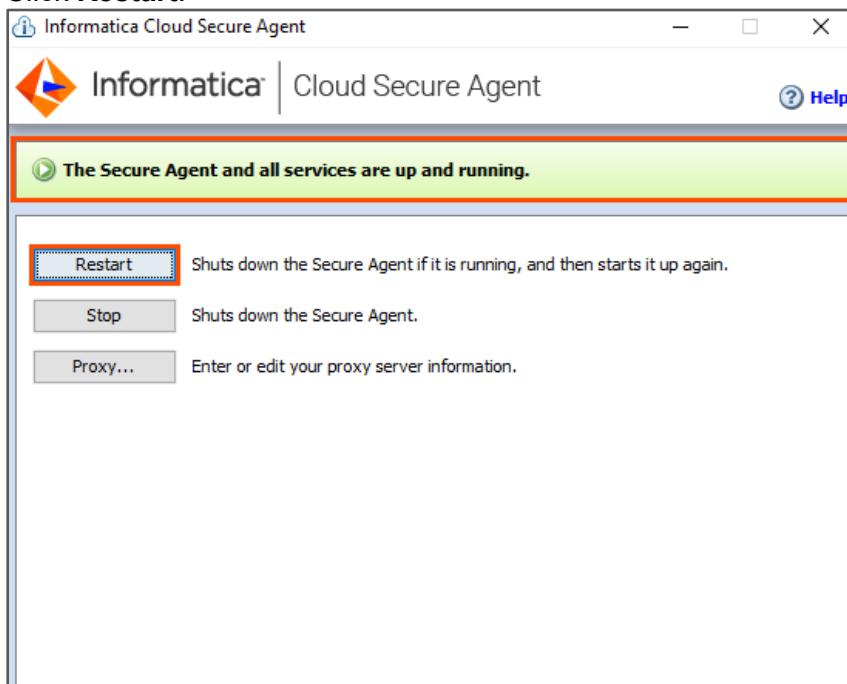
32. Navigate to the **CAI Lab Prep Files** folder on the Desktop. Copy the **mssql-jdbc-9.4.1.jre8** jar file and paste it to **C:\Program Files\Informatica Cloud Secure Agent\apps\process-engine\ext**.



33. To open the secure agent, right-click the secure agent from the desktop, and select **Run as administrator**.



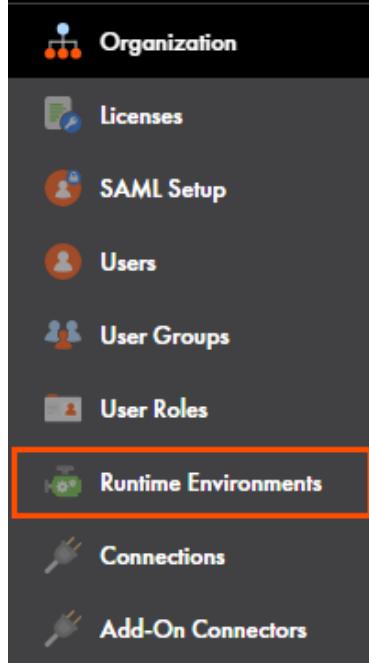
34. Click **Restart**.



**Note:** You must wait for the secure agent to get back to the running state. This may take up to 20 to 30 minutes.

## View and Rename the Secure agent in IICS

35. From the Administrator service navigation pane, select **Runtime Environments**.



**Note:** By default, the secure agent takes the name of the computer it is installed on. It takes some time for the secure agent to appear under **Runtime Environments** page. Refresh the page, if needed.

36. To rename the secure agent group, hover over the drop-down, and select **Rename Secure Agent Group**.



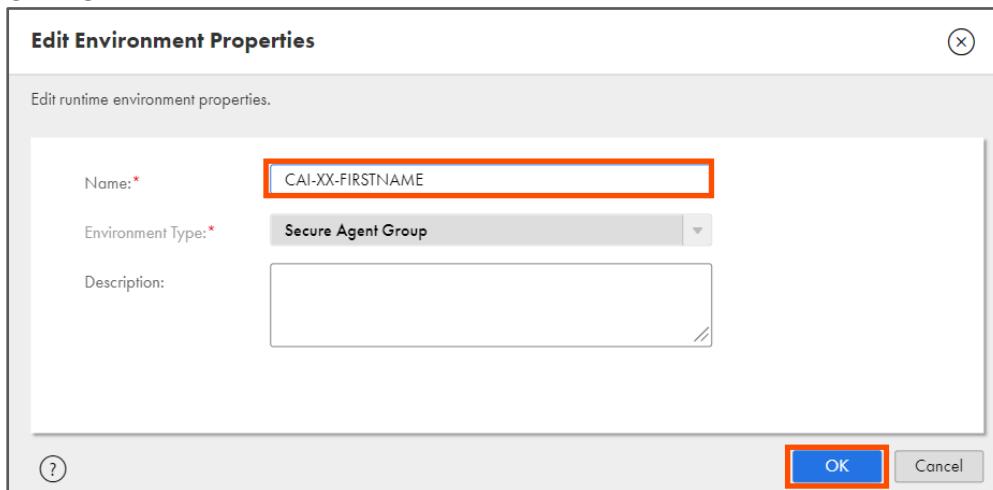
| Name                 | Status           | Agent Group | Last Modified          |
|----------------------|------------------|-------------|------------------------|
| CAI-XX-FIRSTNAME (1) | Running          | Agent Group | Apr 18, 2023, 10:50 PM |
| CAI-XX-FIRSTNAME (1) | Not Secure Agent | Agent Group | Dec 1, 2021, 10:1      |
| CAI-XX-FIRSTNAME (1) | Agent            | Agent Group | Dec 8, 2022, 3:2       |
| CAI-XX-FIRSTNAME (1) | Agent            | Agent Group | Feb 6, 2023, 10:1      |
| CAI-XX-FIRSTNAME (1) | Agent            | Agent Group | Feb 6, 2023, 3:31      |
| CAI-XX-FIRSTNAME (1) | Agent            | Agent Group | Oct 27, 2022, 11:18 PM |

The Rename Secure Agent Group window appears.

37. In the Name field, enter **CAI-XX-FIRSTNAME**, where XX can be your initials or your birth date, and replace FirstName with your first name.

**Note:** **CAI-XX-FIRSTNAME** is an indicative name, you can provide a name of your choice and remember the name.

38. Click **OK**.



39. To rename the secure agent, for the secure agent, hover over the drop-down and select **Edit Secure Agent**.



40. In the Agent Name field, enter **CAI-XX-FIRSTNAME**.



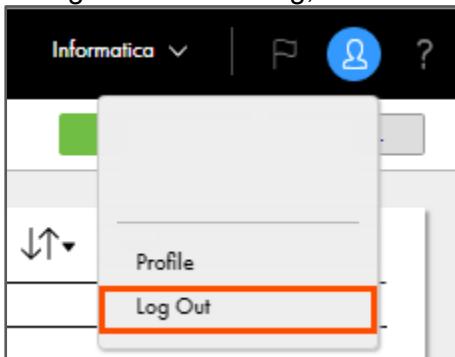
41. On the top right corner, click **Save**.

42. To return to the Runtime Environments page, click .

43. Verify that the secure agent group name and the secure agent's name are now set to **CAI-XX-FirstName**. Also, verify that the status of the secure agent is **Running**.

| Environment Name               | Status   | Enabled Services   |
|--------------------------------|--|--|
| Informatica Cloud Hosted Agent | <input checked="" type="checkbox"/> Up and Running | Data Integration Server  |
| CAI-XX-FIRSTNAME (1)           | <input checked="" type="checkbox"/> Up and Running | File Integration Service, Process Server, B2B Processor, Data Integration Server, Mass Ingestion |

44. To log out from the Org, click **User** and select **Log Out**.



**Note:** You must not close the secure agent window as it is the main component to establish connectivity between your on-premises data and IICS services.

## MOST COMMON TROUBLESHOOTING TECHNIQUES

### Issue: Secure Agent Service status is disabled

| Name                                  | Description                            | Status    | Startup Type              | Log On As      |
|---------------------------------------|--|-----------|---------------------------|----------------|
| IIS Admin Service                     | Enables this server to administer t... | Running   | Automatic                 | Local Syst...  |
| IKE and AuthIP IPsec Keying Modules   | The IKEEXT service hosts the Inter...  | Running   | Automatic (Trigger Start) | Local Syst...  |
| <b>Informatica Cloud Secure Agent</b> | Informatica Cloud Secure Agent         | Disabled  |                           | \Administrator |
| Informatica MFT 10.2.0                | Informatica MFT 10.2.0                 | Automatic |                           | Local Syst...  |
| Infrared monitor service              | Detects other Infrared devices tha...  | Manual    |                           | Local Syst...  |
| Intel Bluetooth Service               | Intel(R) Wireless Bluetooth(R) iBtS... | Running   | Automatic                 | Local Syst...  |

**Solution:** The disabled status of the Informatica Cloud Secure Agent indicates an issue in installation and that the files required for starting the services are missing. You must uninstall and re-install the secure agent to fix this issue. To uninstall the Secure Agent, refer to the section on **Uninstalling the Secure Agent**.

### Issue: ‘Not all the services are running.’

**Solution:** There are many reasons why the agent does not come up. The best method to find the reason is to check the agent **audit logs** and **agentcore.log** files.

- To view the audit logs for a secure agent, navigate to **Administrator > Runtime Environments > Secure Agent > Audit Log**.
- The **agentcore.log** file is located in the following location: **C:\Program Files\Informatica Cloud Secure Agent\apps\agentcore**

You will then have to go through the log files and look for the errors. A few commonly seen errors are listed below.

## Commonly seen errors in agentcore.log file

- Following error message can be seen in the agentcore.log while registering the secure agent with proxy information.

**2018-08-11 11:58:05,127 MST ERROR**

[com.informatica.saas.rest.client.spring.RestTemplateExtended] - HTTP request to URL https://dm-us.informaticacloud.com/ma/api/v3/InternalLogin failed with HTTP status [PROXY\_AUTHENTICATION\_REQUIRED AuthenticationRequired] and response body []

**2018-08-11 11:58:05,127 MST ERROR**

[com.informatica.saas.infaagent.agentcore.AgentCoreStateMachine] - Unable to create metadata file due to the following [407 AuthenticationRequired]."

**Solution:** Navigate to file **proxy.ini** located at **C:\Program Files\Informatica Cloud Secure Agent\apps\agentcore\conf** and provide the proxy details. Save the file and restart the agent.

Sample entry in **proxy.ini**:

```
InfaAgent.ProxyHost=ProxyHostName
InfaAgent.ProxyPort=3128
InfaAgent.ProxyUser=testuser
InfaAgent.ProxyPassword=infa12345
```

- Secure agent keeps restarting or the Data Integration service is getting restarted. The following message is seen in agentcore.log:

**[LCM\_10012] Administrator-1.0.1 is restarting due to an error encountered while running: [Administrator-1.0.1]**

or its Data Integration Server:

**[LCM\_10012] Data\_Integration\_Server-52.0.3 is restarting due to an error encountered while running: [Data\_Integration\_Server-52.0.3]**

**Solution:** This can have different causes. The most likely are:

- Missing licenses i.e. the SDK license: Check with the Informatica support team ([support@informatica.com](mailto:support@informatica.com)) regarding the licenses.
  - Corrupted Secure Agent files: Re-install the secure agent.
- Secure agent status is stuck at "Not all the services are running". The following message is seen in the log file.

**2018-09-25 06:01:40,438 EDT INFO**

[com.informatica.saas.lcm.lcmnative.ProcessSupervisor] ->>> OutOfMemoryError: Metaspace

**2018-09-25 06:01:41,342 EDT INFO**

[com.informatica.saas.lcm.lcmnative.ProcessSupervisor] - ProcessMonitor [CommandSpec command[E:\csapps\Informatica Secure

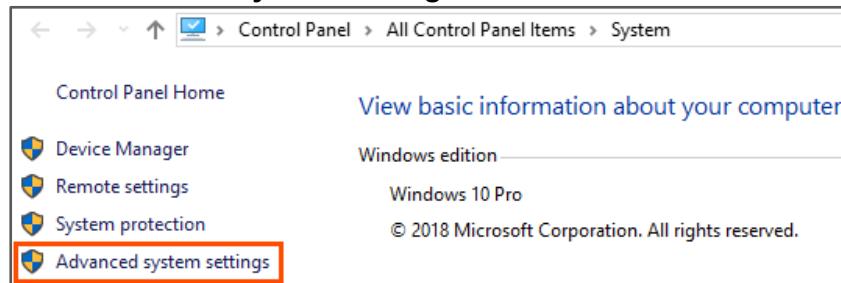
```

Agent\apps\Data_Integration_Server\52.0.3\lcm\stop-wrapper.bat] args[]
dir[E:\csapps\Informatica Secure Agent\apps\Data_Integration_Server\52.0.3]
timeout[600000] waitForever[false]] state[EXITED] tmpFile[E:\csapps\Informatica
Secure
Agent\apps\agentcore\logs\temp\ProcessSupervisor4001252731309672350.tmp]
completed exit code: 1 output:
2018-09-25 06:01:41,342 EDT INFO
[com.informatica.saas.lcm.lcmnative.ProcessSupervisor] - >>>
E:\csapps\Informatica Secure
Agent\apps\Data_Integration_Server\52.0.3>ics_stop.bat

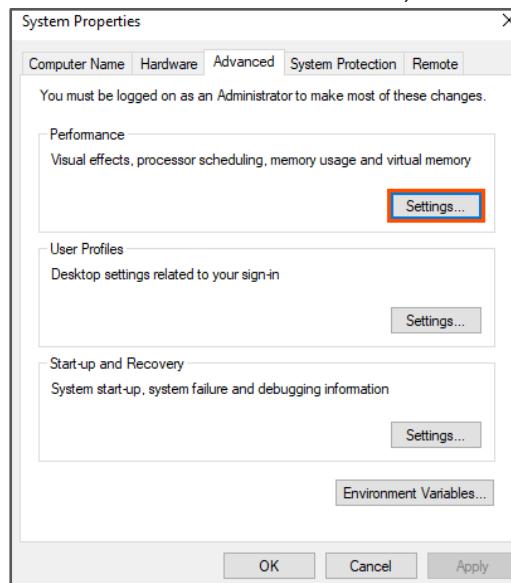
```

**Solution:** To fix the outOfMemory issue, make the following changes in the server where the secure agent is installed:

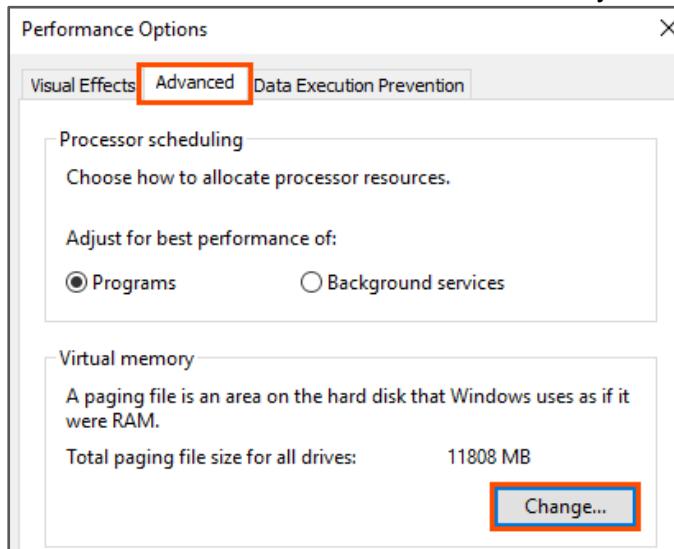
- Right-click on **My computer > Properties.**
- Go to **Advanced system settings.**



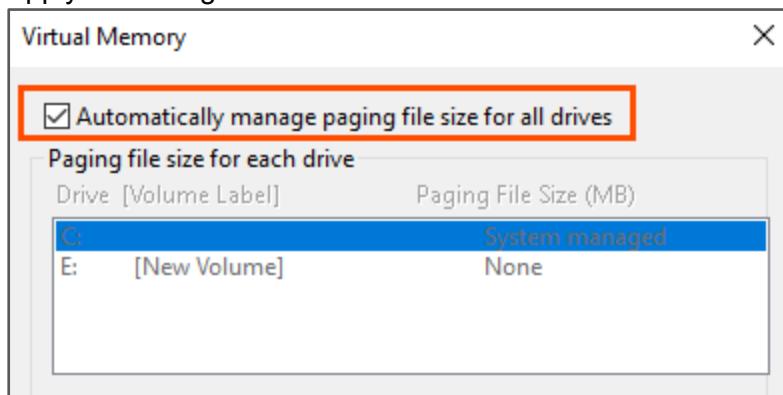
- From the Performance section, click **Settings.**



- d. Click the **Advanced** tab. In the Virtual Memory section, select **Change**.



- e. Select **Automatically managed paging file size for all Drives**.  
 f. Apply the changes and restart the server.

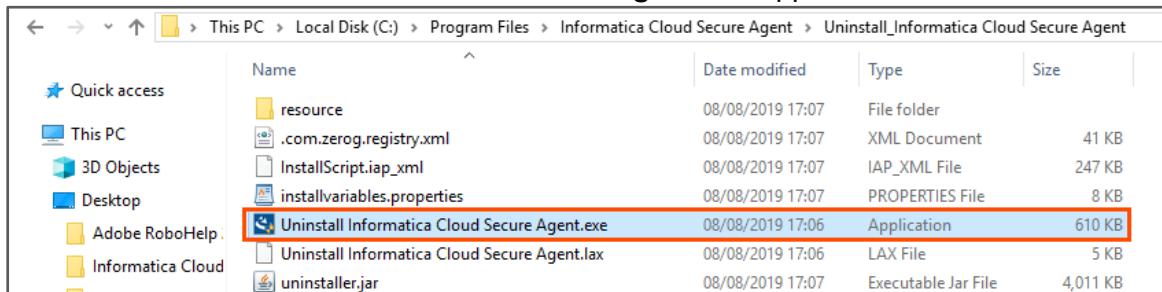


## Uninstalling the Secure Agent

To re-install the Secure Agent, you must first uninstall it. Perform the following steps to uninstall the Secure Agent.

- a. From the File Explorer, navigate to **C:\Program Files\Informatica Cloud Secure Agent\Uninstall\_Informatica Cloud Secure Agent**.  
**Note:** The agent installation path may vary from user to user.

- b. Run the **Uninstall Informatica Cloud Secure Agent.exe** application.

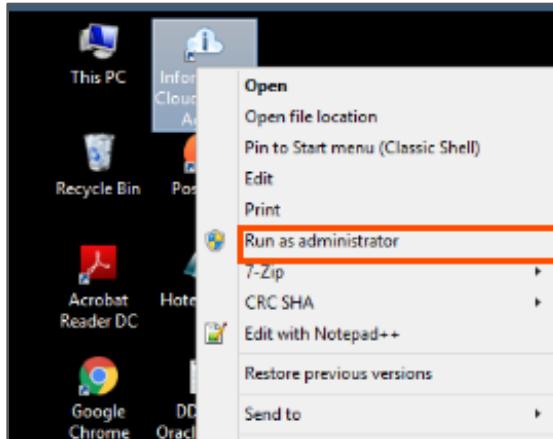


| Name  | Date modified    | Type                | Size     |
|---|------------------|---------------------|----------|
| resource  | 08/08/2019 17:07 | File folder         |          |
| .com.zerog.registry.xml                             | 08/08/2019 17:07 | XML Document        | 41 KB    |
| InstallScript.iap_xml                               | 08/08/2019 17:07 | IAP_XML File        | 247 KB   |
| installvariables.properties                         | 08/08/2019 17:07 | PROPERTIES File     | 8 KB     |
| <b>Uninstall Informatica Cloud Secure Agent.exe</b> | 08/08/2019 17:06 | Application         | 610 KB   |
| Uninstall Informatica Cloud Secure Agent.lax        | 08/08/2019 17:06 | LAX File            | 5 KB     |
| uninstaller.jar                                     | 08/08/2019 17:07 | Executable Jar File | 4,011 KB |

- c. Verify that the **Informatica Cloud Secure Agent** folder is removed from the **C:\Program Files** location. If not, manually delete all the remaining files in the installation directory.  
d. After the Informatica Cloud Secure Agent folder is removed, re-install the secure agent.

## Important Points for Secure Agent Installation

1. Always use the latest secure agent installer downloaded from the IICS Org.
2. Always run the secure agent as an ‘Administrator’.



3. After providing the org details and the install token in the secure agent registration window, the secure agent downloads the necessary files for the connectors. This process takes 20-30 minutes (depending upon the internet speed). You must not stop or restart the secure agent until the secure agent status changes to “**Up and running**”.
4. You must provide Administrator rights to your secure agent.

*This concludes the lab.*

## Pre-requisite Labs

### Lab 0-3: Configure SQL Server Database

#### Overview:

MS SQL Server is a relational database management system. It is commonly used for running online transaction processing (OLTP), data warehousing (DW), and mixed (OLTP & DW) database workloads.

This document lists the steps to configure the SQL Server database for performing lab exercises for this course.

#### Objective:

- Create User
- Create Tables

#### Duration:

30 minutes

---

#### Important

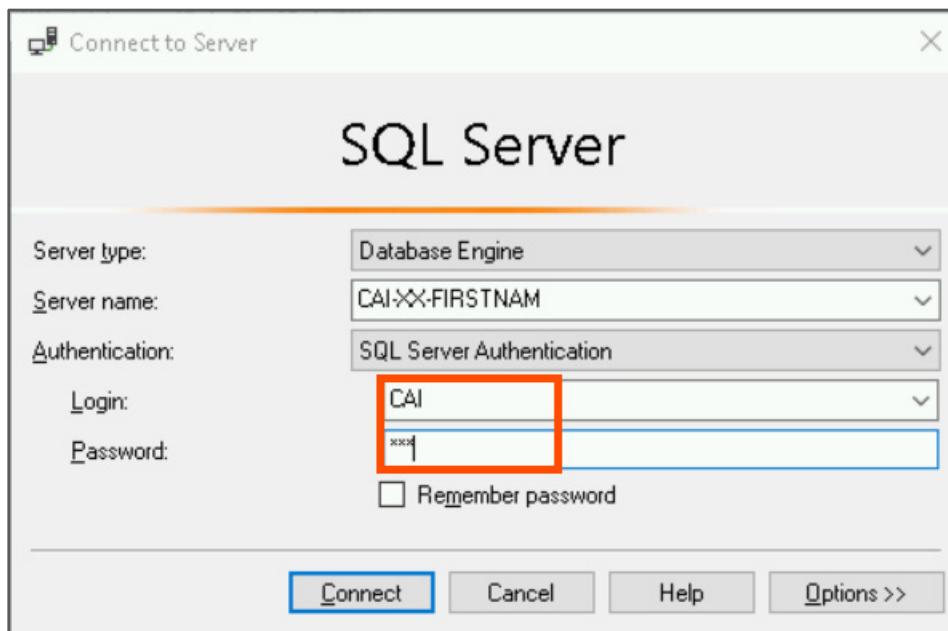
1. It is recommended that your system has a minimum of 8 GB RAM.
2. In the lab exercises for this course, the following lab guides are designed and documented using SQL Server database. We therefore recommend that you have SQL Server database installed on your system (Secure Agent machine) to execute the labs.

| <b>List of SQL Server Database Dependent Labs</b>          |
|--|
| Lab 6-1: Create a Data Access Service Connector Using Form |
| Lab 7-2: Create a JDBC Connection                          |
| Lab 10-1: Invoke a Synchronous Web Service Call            |
| Lab 10-2: Invoke an Asynchronous Web Service Call          |
| Lab 11-1: Handle Sub-process Failures                      |
| Lab 14-1: API Manager                                      |

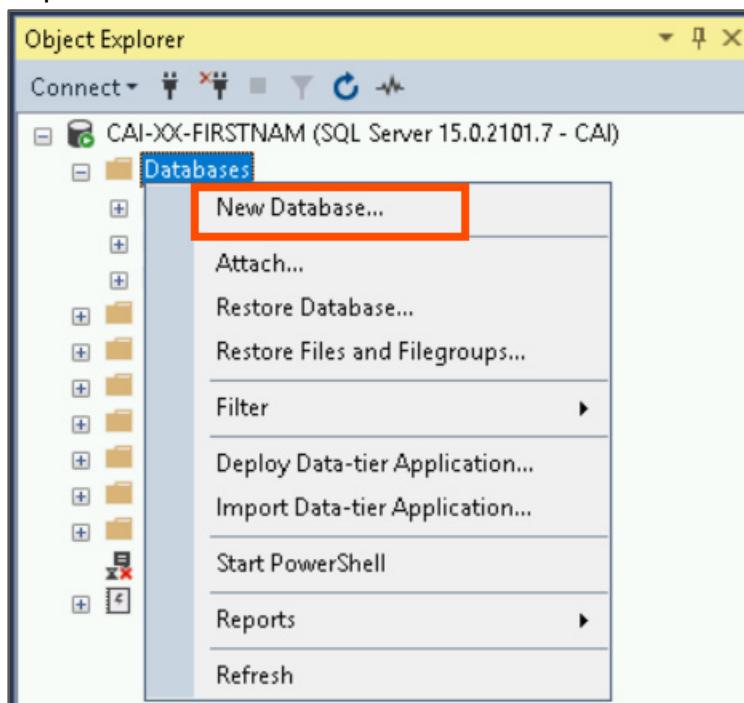
**Note:** If you do not have SQL Server installed on your system, you can skip the labs that are SQL Server database dependent and continue with the rest of the course and lab exercises. When you come across these labs, an alternate way of performing few SQL Server database dependent labs is specified in the course.

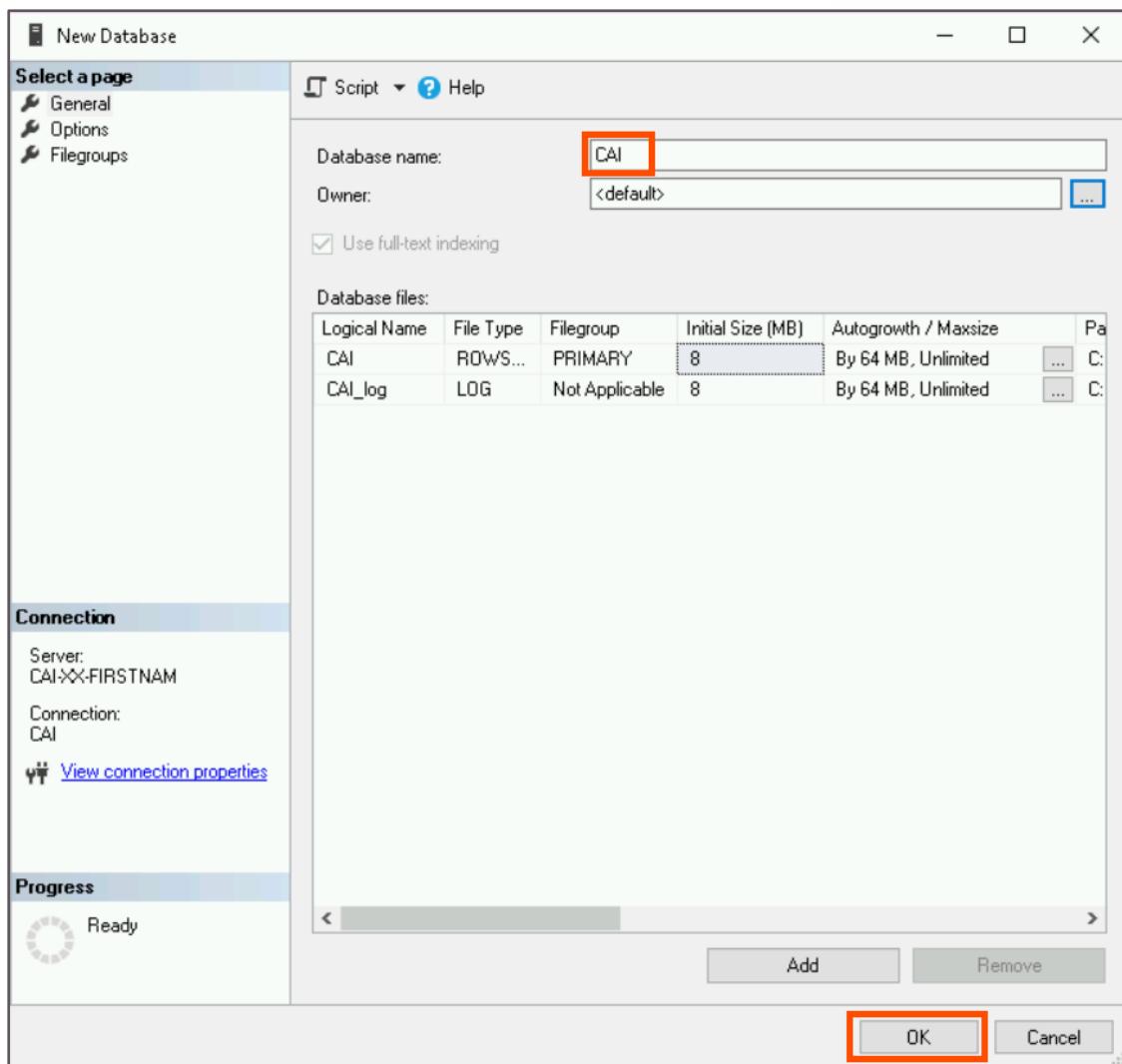
3. The labs for this course are created on **SQL Server 2019** and using the client **Microsoft SQL Server Management Studio** version 19.0.

4. Get in touch with your Organization's IT team and in the SQL Server Management Studio tool, create an **SQL Server Authentication** user with login username as **CAI** and password as **CAI**.



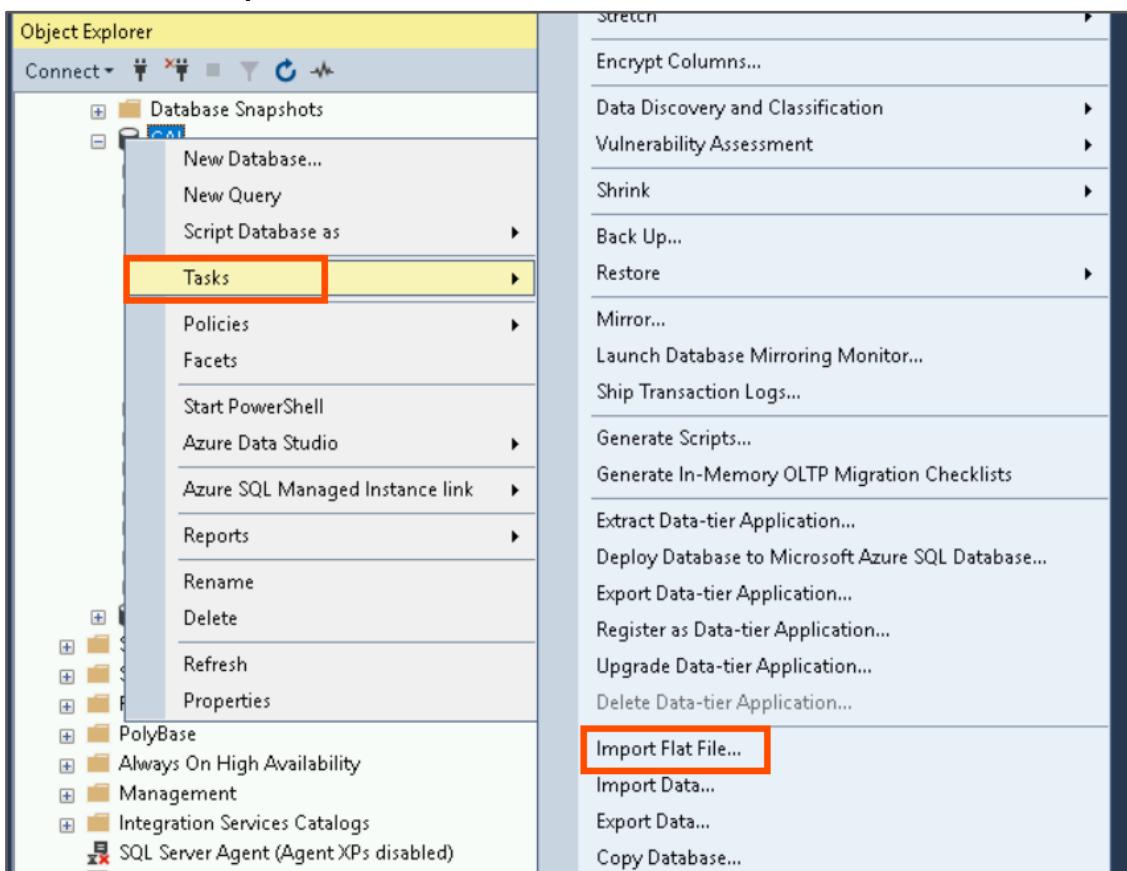
5. In MS SQL Server Management Studio, connect to your using the login credentials **CAI/CAI**.
6. Expand Databases and create a database named **CAI**.



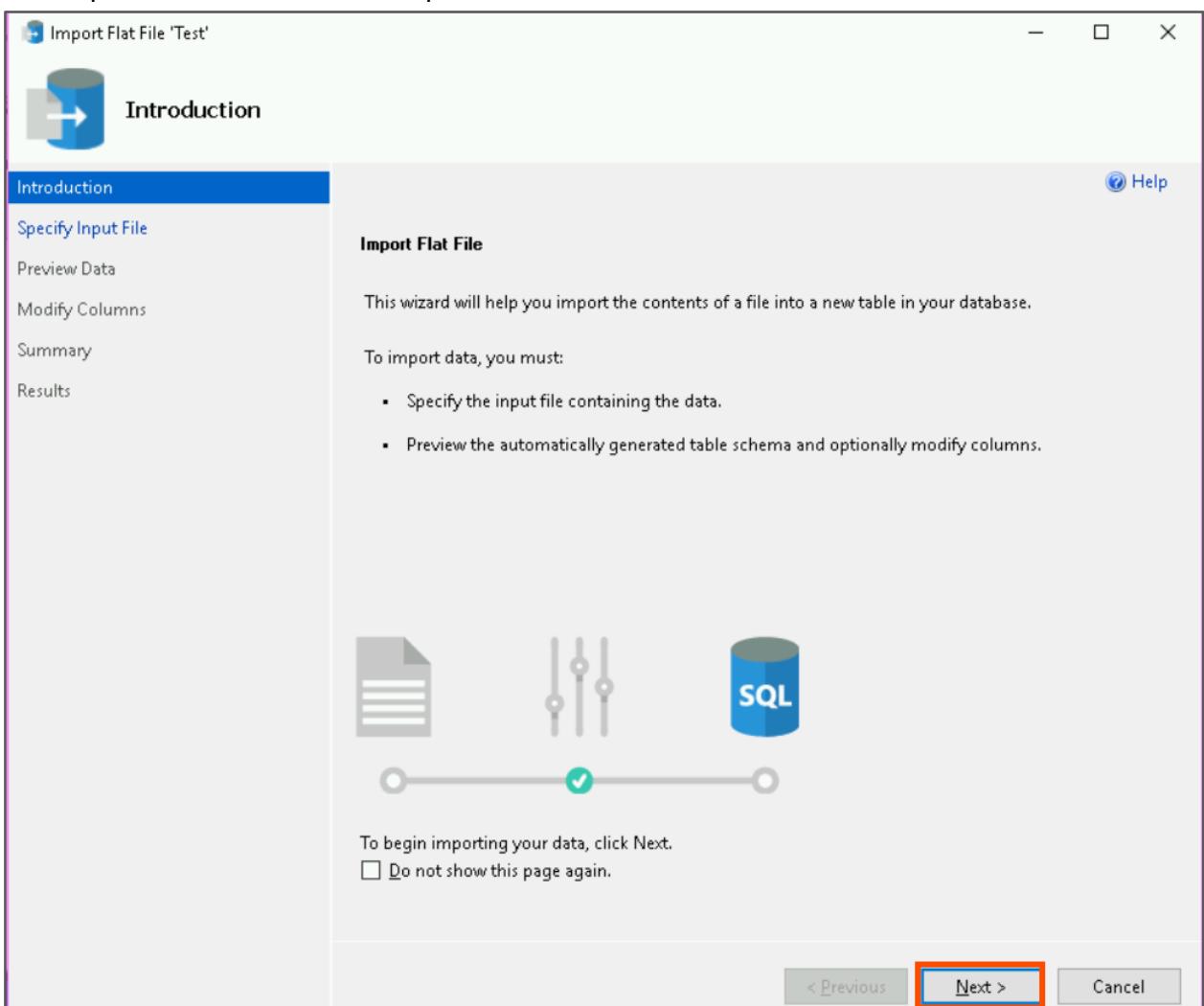


## Create Tables

7. In the CAI Lab Prep Files folder, you will have flat file name CURRENCY\_EXCHANGE\_SVC. You will create a new table in the CAI database using this flat file.
8. Navigate to the SQL Server client tool and right-click the **CAI** database.
9. Select **Tasks > Import Flat File**.

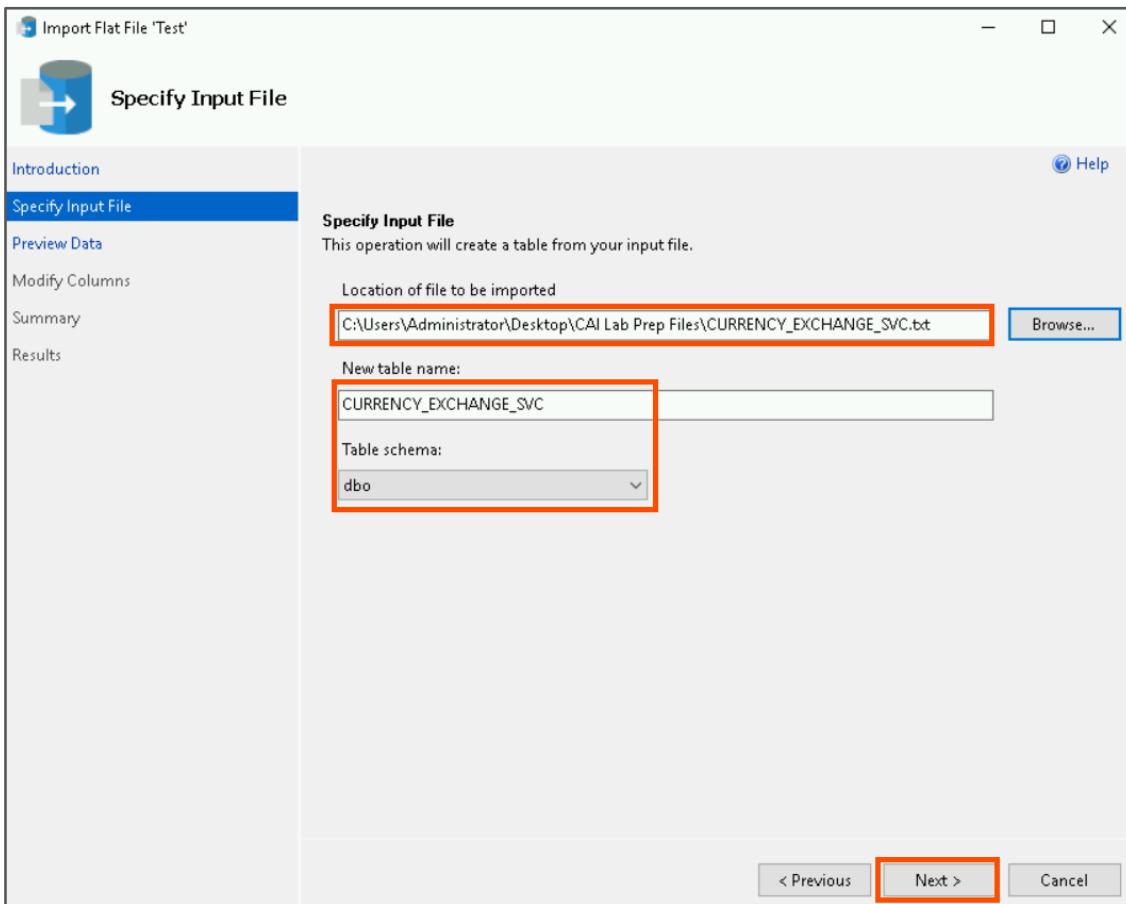


10. The Import Flat File 'CAI' wizard opens. Click **Next**.



11. In the Specify Input File tab, click **Browse** and navigate to the CAI Lab Prep Files folder. Select the flat file **CURRENCY\_EXCHANGE\_SVC.txt**.

12. Retain the table name as **CURRENCY\_EXCHANGE\_SVC** and the Table Schema to **dbo**.



13. Click **Next**.

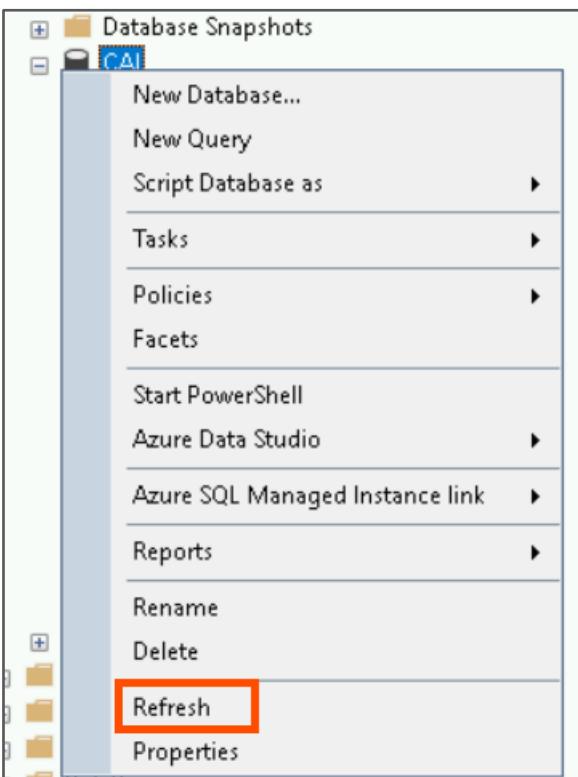
14. In the Preview Data tab, view the data, and click **Next**.

15. In the Modify Columns tab, click **Next**.

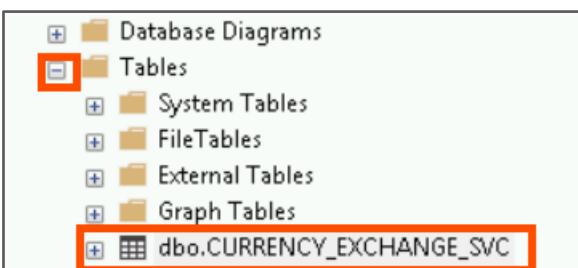
16. In the Summary tab, click **Finish**. This will complete the data import operation.

17. Click **Close** to close the data import wizard.

18. To populate the newly added database table, right-click the database **CAI** and click **Refresh**.



19. Expand **CAI > Tables** and observe that the table is successfully created in the CAI database.



20. Similarly, create the **employees** table in the **CAI** database.

21. Close the SQL Server client tool.

Thus, you have successfully created databases and database tables in SQL Server.

*This concludes the lab.*

## Pre-requisite Labs

### Lab 0-4: Configure Kafka Setup

**Overview:**

Kafka is a distributed publish-subscribe messaging system that is fast, scalable, and durable. It stores messages in the form of topics. Producers write data to a topic and consumers read data from the topic. Kafka topics are partitioned and replicated across multiple nodes thereby allowing distributed processing. Kafka uses ZooKeeper to manage the Kafka cluster.

In this lab, you will install Apache ZooKeeper and Kafka in your machine and configure them to use the Kafka connector in a process to read and write data.

**Objectives:**

- Install and configure Java
- Create folders for the Kafka directory
- Download Apache ZooKeeper and Kafka for windows
- Set Environment Variables
- Modify the configuration files

**Notes:**

- Ensure that you have the **7-Zip** setup installed in your machine.

**Duration:**

30 Minutes

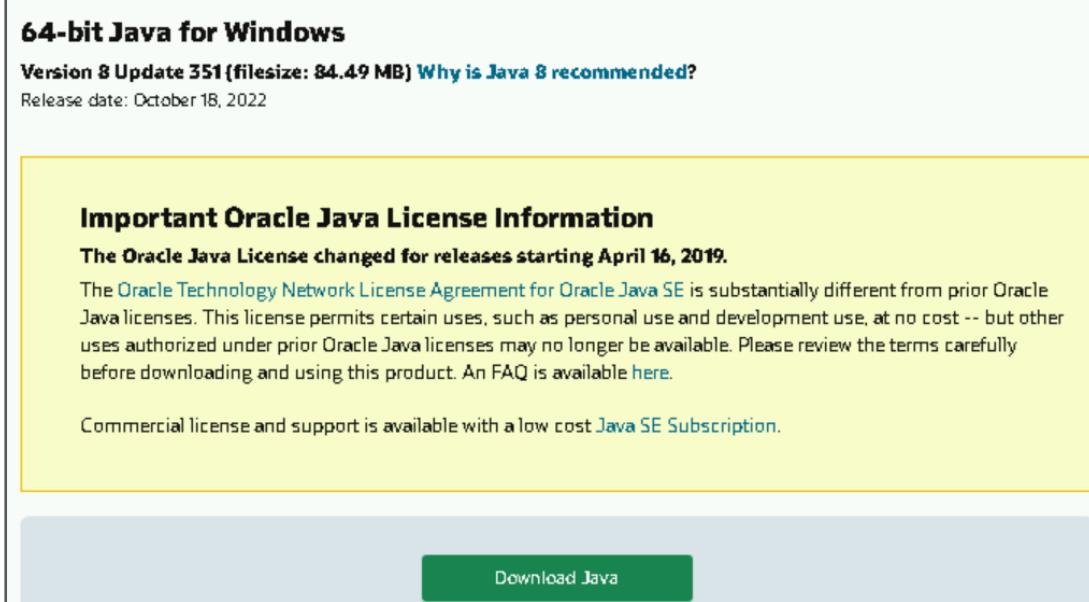
---

**Tasks**

**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

## Install and Configure Java

1. In a web browser, enter the following URL:  
<https://java.com/en/download/>
2. Click Download Java.



**64-bit Java for Windows**

**Version 8 Update 351 (filesize: 84.49 MB) Why is Java 8 recommended?**  
 Release date: October 18, 2022

**Important Oracle Java License Information**

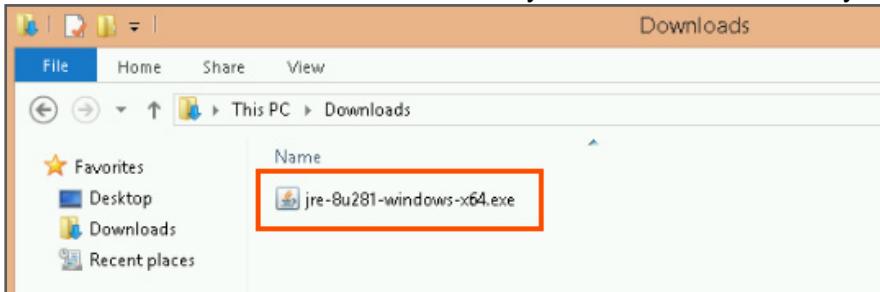
The Oracle Java License changed for releases starting April 16, 2019.

The Oracle Technology Network License Agreement for Oracle Java SE is substantially different from prior Oracle Java licenses. This license permits certain uses, such as personal use and development use, at no cost -- but other uses authorized under prior Oracle Java licenses may no longer be available. Please review the terms carefully before downloading and using this product. An FAQ is available [here](#).

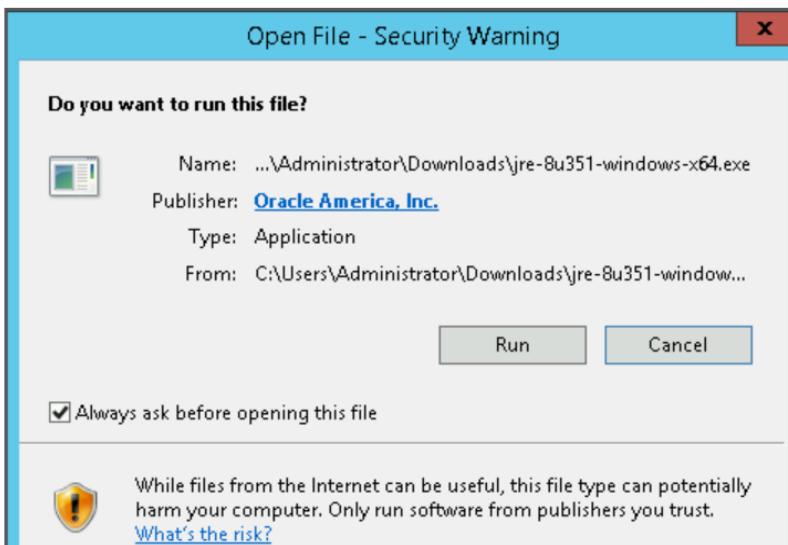
Commercial license and support is available with a low cost [Java SE Subscription](#).

[Download Java](#)

3. Run the downloaded Java installer from your Downloads directory.



4. Click Run.



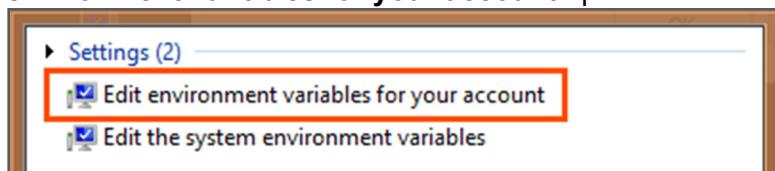
5. In the Java Setup window, click **Install**.



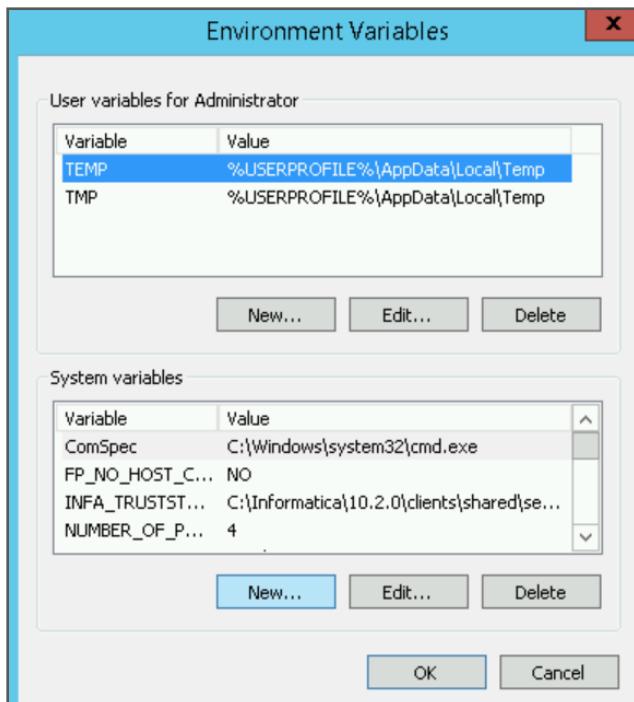
6. Once the Java is installed, you will see the following message. Click **Close**.



7. Now you must setup the **JAVA\_HOME** environment variable.
8. In the Windows search menu, search for Environment Variables, and select the **Edit environment variables for your account** option.



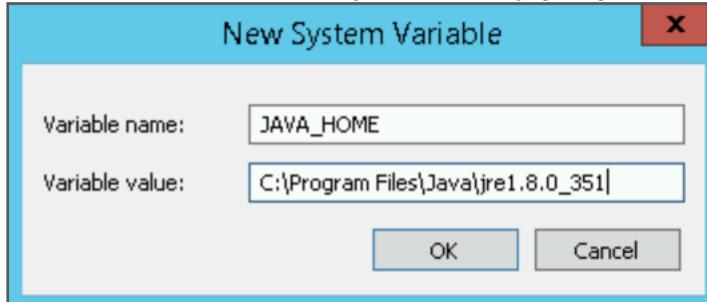
9. In the Environment Variables window, under **System Variables**, click **New**.



10. In the Variable Name field, enter **JAVA\_HOME**.

11. In the Variable Value field, enter the path to the **jre<version>** directory.

**Note:** You can confirm the jre version by going to the path: **C:\Program Files\Java**.

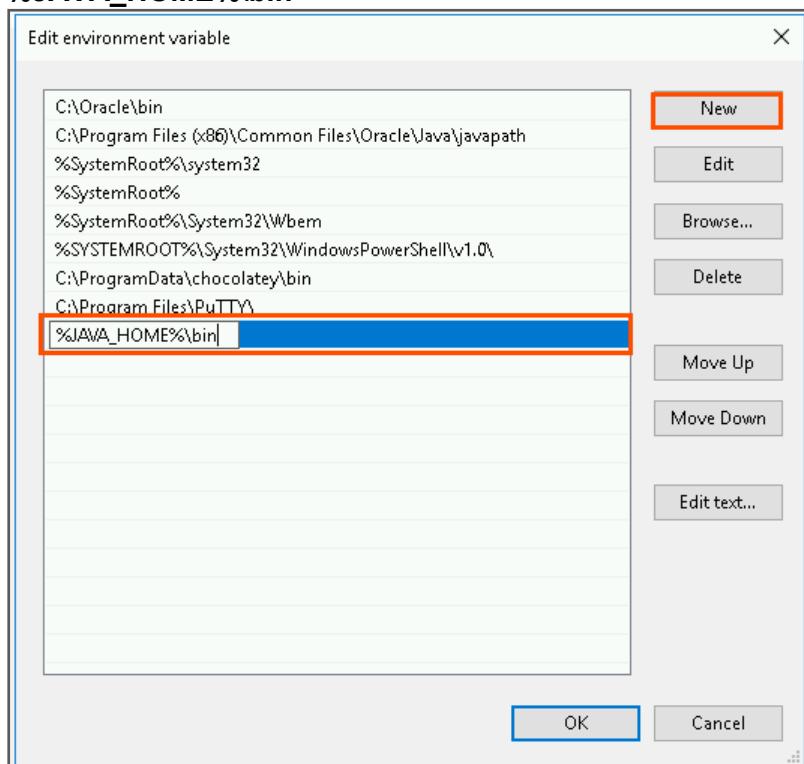


12. Click **OK**.

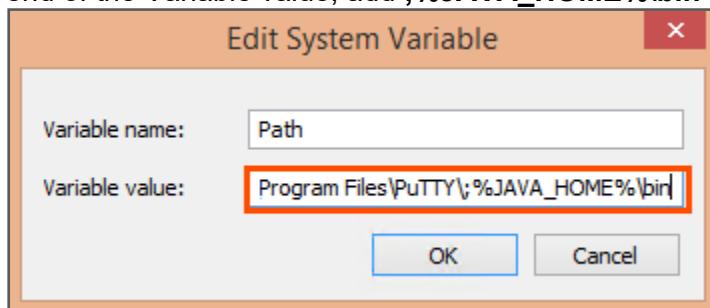
13. Under the System Variables section, select the **Path** environment variable, and click **Edit**.

14. Click **New** and add the following value:

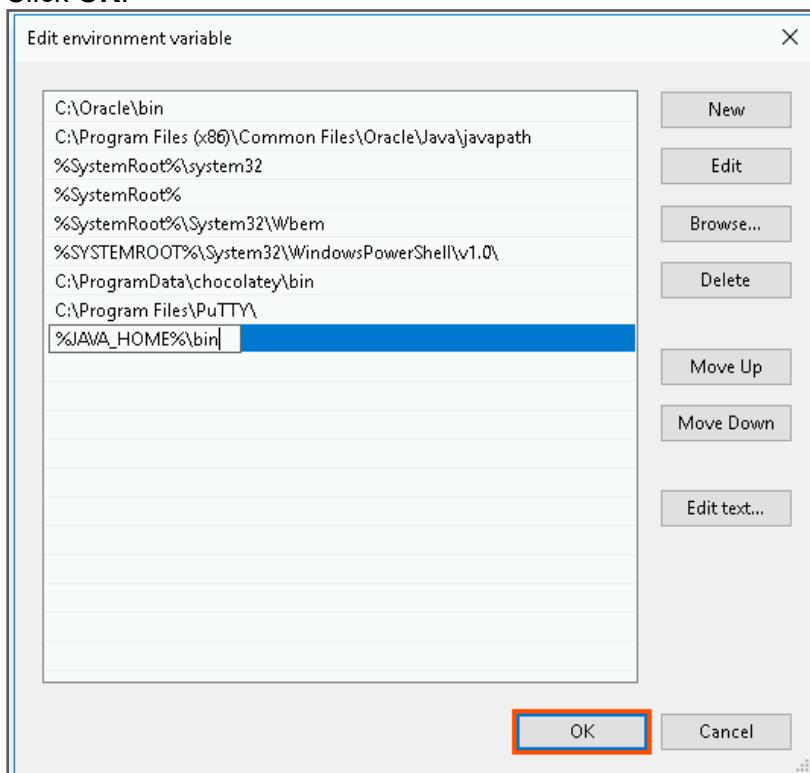
**%JAVA\_HOME%\bin**



**Note:** If you see the Edit System Variable window as shown in the image below, at the end of the Variable value, add ;%JAVA\_HOME%\bin



15. Click **OK**.



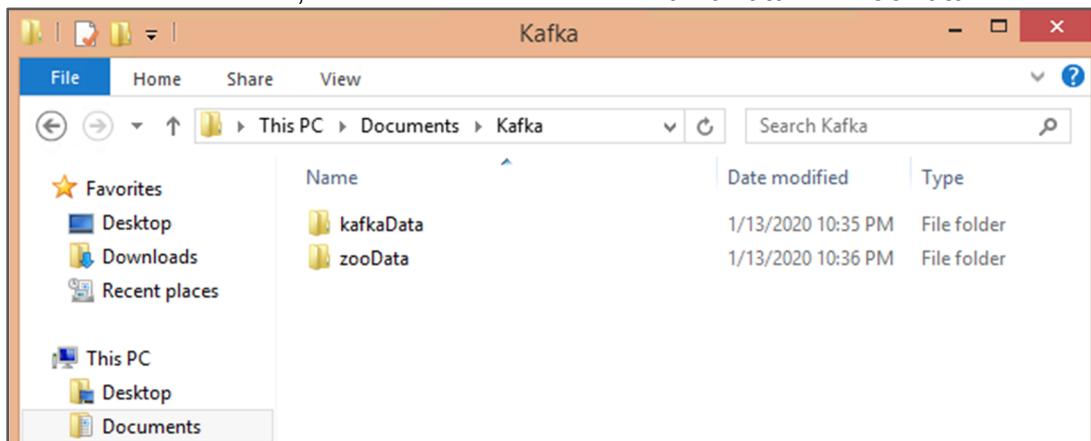
16. For the Environment Variable window, click **OK**.

### Create folders for the Kafka directory

17. In your machine, in the **Documents** folder, create a folder named **Kafka**.

Folder Path:<User Directory>\Documents\Kafka

18. Inside the Kafka folder, create two folders named **kafkaData** and **zooData**.



## Download Apache Zookeeper for Windows

19. In a web browser, enter the following URL:

<https://zookeeper.apache.org/releases.html>

20. Click on the first source release link.



The screenshot shows the Apache ZooKeeper Releases page. At the top, there is a cartoon character holding a shovel and a pickaxe, with the text "Apache ZooKeeper™". To the right is a menu icon. Below the header, the title "Apache ZooKeeper™ Releases" is displayed in large, bold letters. A brief description follows: "The Apache ZooKeeper system for distributed coordination is a high-performance service for building distributed applications." Below this, a list of links includes "Download", "Verifying Hashes and Signatures", "Release Notes", and "News". A section titled "Download" highlights "Apache ZooKeeper 3.7.0" as the latest stable release. The download link "Apache ZooKeeper 3.7.0 Source Release asc, sha512" is shown in green text and is enclosed in a red rectangular box.

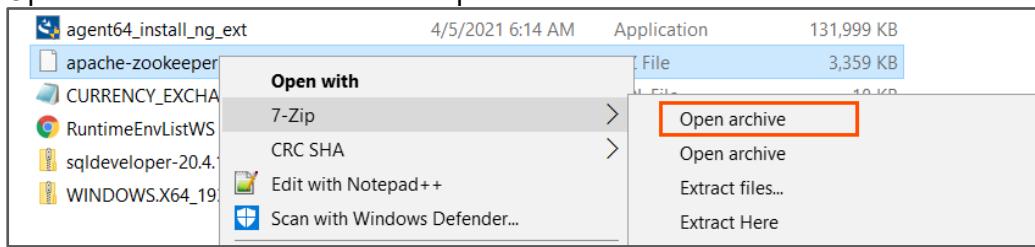
21. Click on the first mirror site.



The screenshot shows the Apache Software Foundation homepage. The navigation bar at the top includes links for News, About, Make a Donation, The Apache Way, Join Us, Downloads, and a search icon. The main header features the Apache logo and the text "COMMUNITY-LED DEVELOPMENT" and "THE APACHE WAY". Below the header, there are links for Projects, People, Community, License, and Sponsors. A sidebar on the left suggests a mirror site: "We suggest the following mirror site for your download: <https://apache.claz.org/zookeeper/zookeeper-3.7.0/apache-zookeeper-3.7.0.tar.gz>". Other mirror sites listed include "https://apache.osuosl.org/zookeeper/zookeeper-3.7.0/apache-zookeeper-3.7.0.tar.gz" and "https://ftp.wayne.edu/apache/zookeeper/zookeeper-3.7.0/apache-zookeeper-3.7.0.tar.gz".

**Note:** This downloads the Apache Zookeeper setup file.

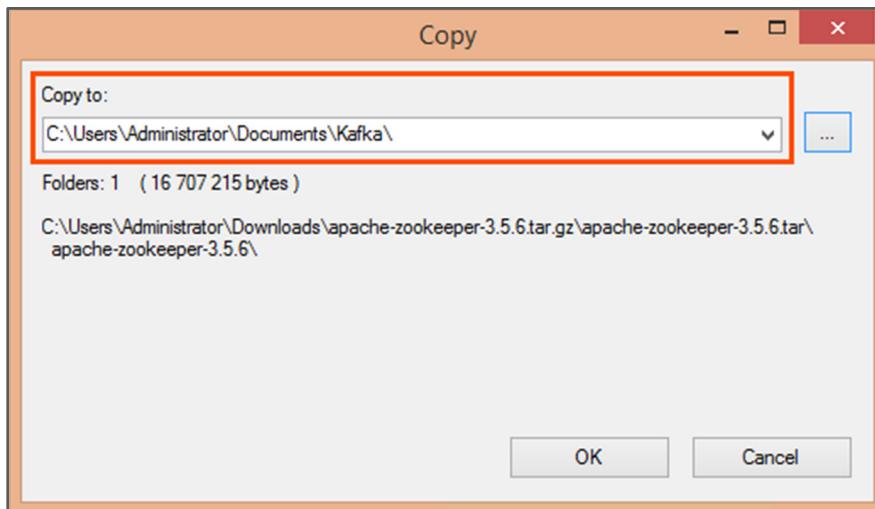
22. Open the downloaded file with 7-Zip.



23. In the 7-Zip window, double click the **apache-zookeeper** file.

**Note:** This extracts the apache-zookeeper folder.

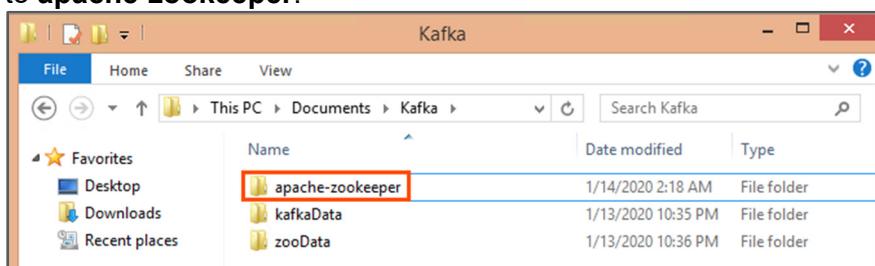
24. Right click and copy the extracted folder to the Kafka directory created in the Documents folder.



25. Click **OK**.

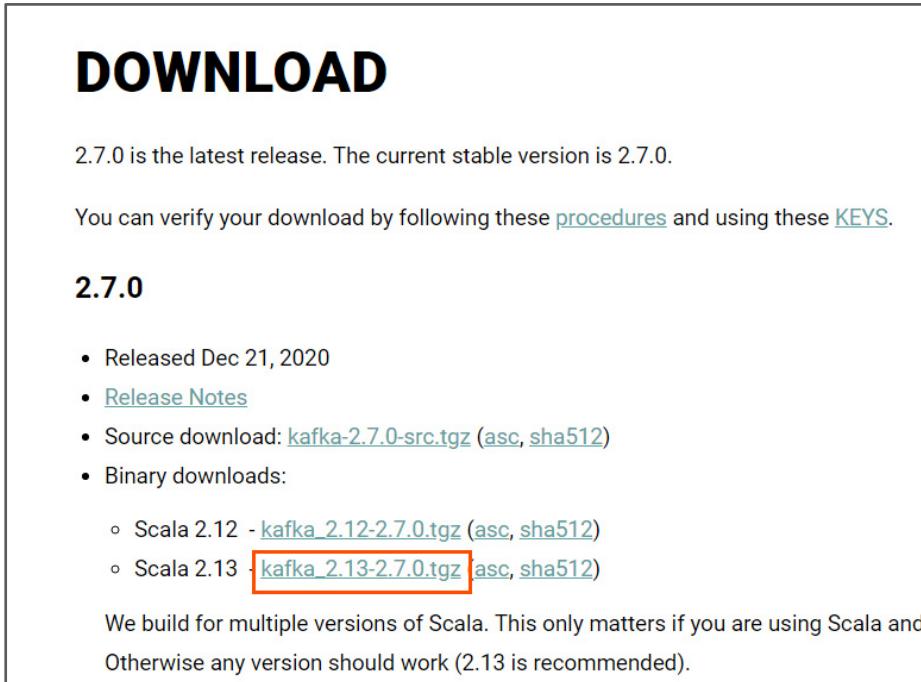
**Note:** Confirm that the folder is copied to the Kafka directory.

26. In the Documents folder, in the Kafka directory, rename the apache-zookeeper-3.5.6 file to **apache-zookeeper**.



## Download Kafka for Windows

27. In a web browser, enter the following URL:  
<https://kafka.apache.org/downloads>
28. In the Binary downloads section, scroll down to the latest version, and click the **Scala** link.



**DOWNLOAD**

2.7.0 is the latest release. The current stable version is 2.7.0.

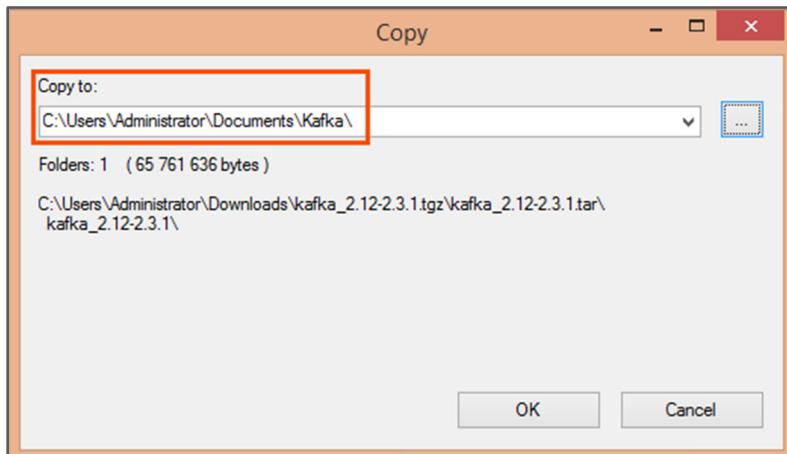
You can verify your download by following these [procedures](#) and using these [KEYS](#).

### 2.7.0

- Released Dec 21, 2020
- [Release Notes](#)
- Source download: [kafka-2.7.0-src.tgz \(asc, sha512\)](#)
- Binary downloads:
  - Scala 2.12 - [kafka\\_2.12-2.7.0.tgz \(asc, sha512\)](#)
  - Scala 2.13 - [kafka\\_2.13-2.7.0.tgz \(asc, sha512\)](#)

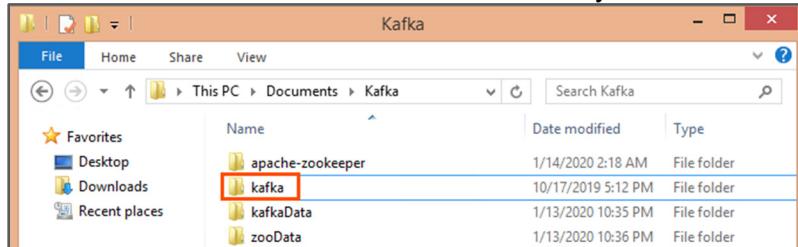
We build for multiple versions of Scala. This only matters if you are using Scala and Otherwise any version should work (2.13 is recommended).

- Note:** This downloads the Kafka setup file.
29. Open the downloaded file with 7-Zip.
  30. In the 7-Zip window, double click the **kafka** file.  
**Note:** This extracts the kafka\_2.13-3.3.1 folder. (Actual version may vary from the screenshot).
  31. Right click and copy the extracted folder to the Kafka directory created in the Documents folder.



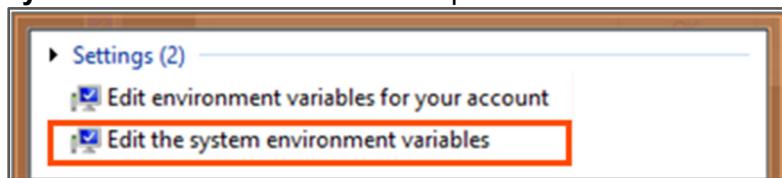
32. Click **OK**.  
**Note:** Confirm that the folder is copied to the Kafka directory.

33. In the Documents folder, in the Kafka directory, rename the extracted kafka file to **kafka**.



## Set Environment Variable for Apache Zookeeper

34. In the Windows search menu, search for Environment Variables, and select the **Edit the system environment variables** option.



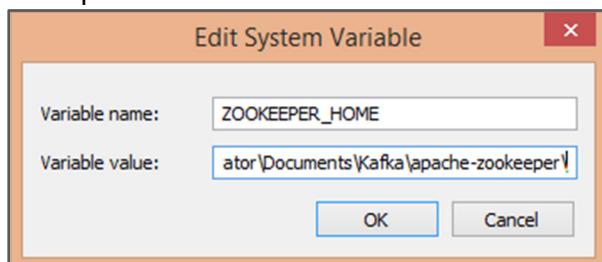
35. In the Environment Variables window, under **System Variables**, click **New**.

36. In the Variable Name field, enter **ZOOKEEPER\_HOME**.

37. In the Variable Value field, enter the following Apache Zookeeper folder path:

<User Directory>\Documents\Kafka\apache-zookeeper

Example: C:\Users\Administrator\Documents\Kafka\apache-zookeeper\



38. Click **OK**.

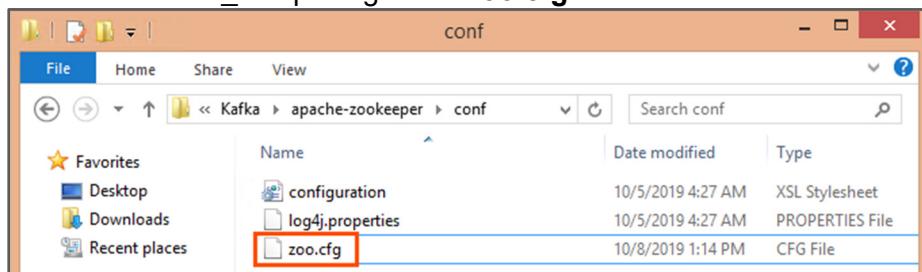
39. For the Environment Variable window, click **OK**.

## Modify the configuration files

40. In the following location, locate the **zoo\_sample.cfg** file.

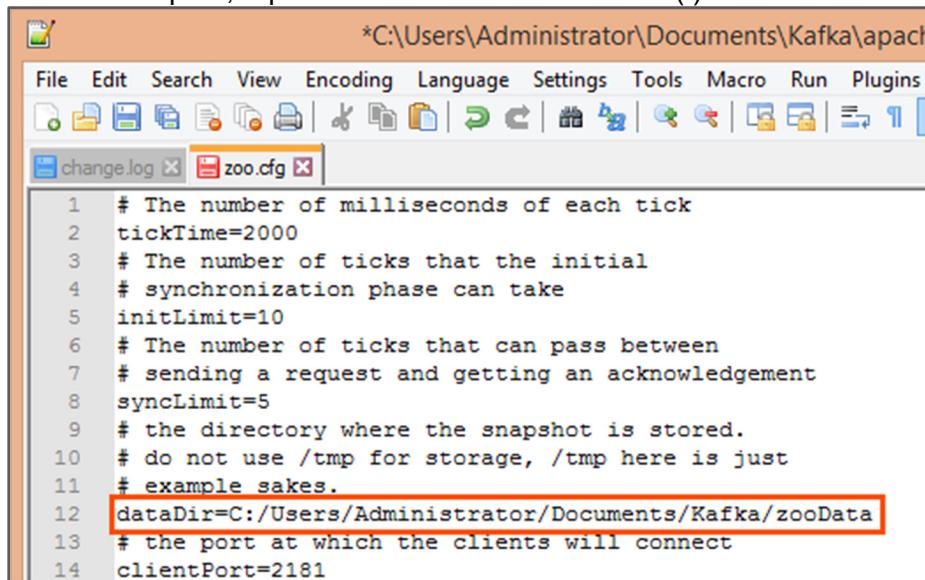
<User Directory>\Documents\Kafka\apache-zookeeper\conf

41. Rename the **zoo\_sample.cfg** file to **zoo.cfg**.



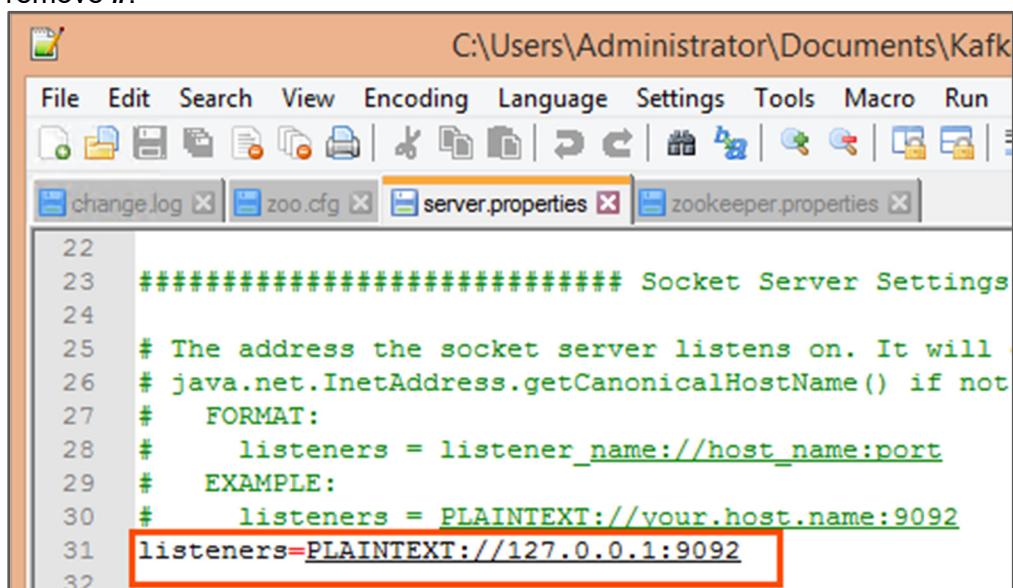
42. Open the **zoo.cfg** file using notepad ++.

43. In the dataDir field, enter the path of the zooData folder present in the Kafka Directory:  
**<User Directory>\Documents\Kafka\zooData**  
**Note:** This is the same Kafka directory that you created in the Documents folder.
44. In the folder path, replace all the backward slashes (\) with the forward slashes (/).



```
*C:\Users\Administrator\Documents\Kafka\apache-kafka-2.1.0\config>
File Edit Search View Encoding Language Settings Tools Macro Run Plugins
change.log x zoo.cfg x
1 # The number of milliseconds of each tick
2 tickTime=2000
3 # The number of ticks that the initial
4 # synchronization phase can take
5 initLimit=10
6 # The number of ticks that can pass between
7 # sending a request and getting an acknowledgement
8 syncLimit=5
9 # the directory where the snapshot is stored.
10 # do not use /tmp for storage, /tmp here is just
11 # example sakes.
12 dataDir=C:/Users/Administrator/Documents/Kafka/zooData
13 # the port at which the clients will connect
14 clientPort=2181
```

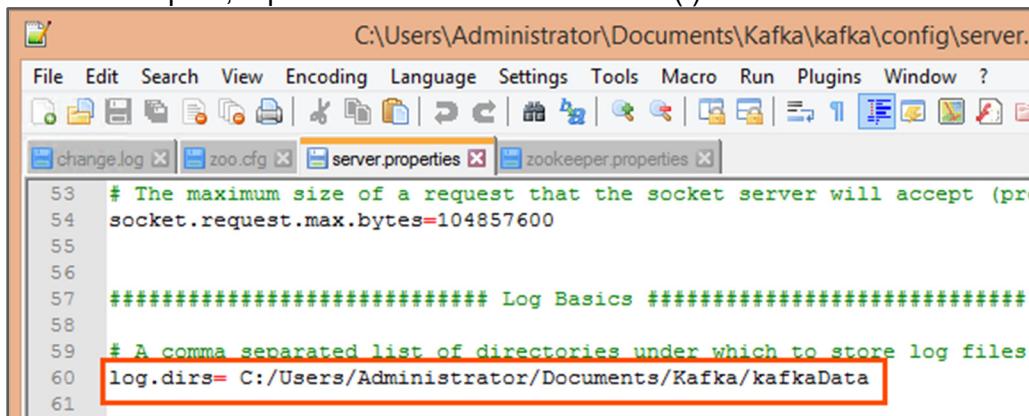
45. Save the file.
46. In the following location, open the **server.properties** file:  
**<User Directory>\Documents\Kafka\kafka\config**
47. In the #listeners field, enter the value as **#listeners=PLAINTEXT://127.0.0.1:9092**, and remove #.



```
C:\Users\Administrator\Documents\Kafka\kafka\config>
File Edit Search View Encoding Language Settings Tools Macro Run Plugins
change.log x zoo.cfg x server.properties x zookeeper.properties x
22 #####
23 ##### Socket Server Settings
24
25 # The address the socket server listens on. It will
26 # java.net.InetAddress.getCanonicalHostName() if not
27 #   FORMAT:
28 #     listeners = listener_name://host_name:port
29 #   EXAMPLE:
30 #     listeners = PLAINTEXT://your.host.name:9092
31 listeners=PLAINTEXT://127.0.0.1:9092
32
```

48. Scroll down and in the log.dirs field, enter the path of the kafkaData folder present in the Kafka Directory:  
**<User Directory>\Documents\Kafka\kafkaData**  
**Note:** This is the same Kafka directory that you created in the Documents folder.

49. In the folder path, replace all the backward slashes (\) with the forward slashes (/).



```
C:\Users\Administrator\Documents\Kafka\kafka\config\server.properties

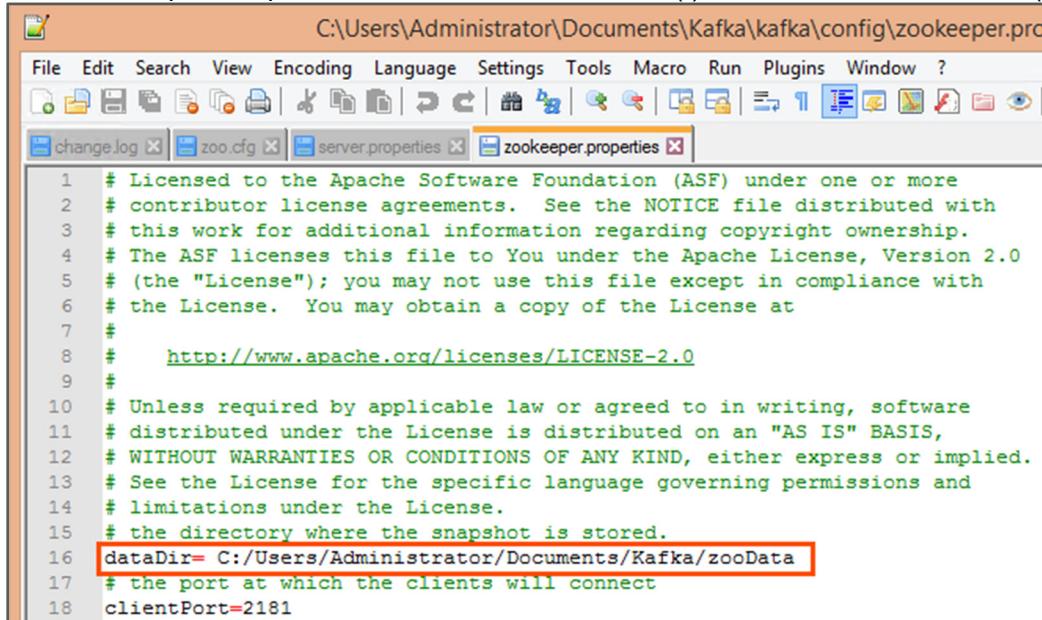
53 # The maximum size of a request that the socket server will accept (pr
54 socket.request.max.bytes=104857600
55
56
57 ##### Log Basics #####
58
59 # A comma separated list of directories under which to store log files
60 log.dirs=C:/Users/Administrator/Documents/Kafka/kafkaData
61
```

50. Save the file.

51. In the following location, open the **zookeeper.properties** file:  
**<User Directory>\Documents\Kafka\kafka\config**

52. In the dataDir field, enter the path of the zooData folder present in the Kafka Directory:  
**<User Directory>\Documents\Kafka\zooData**

53. In the folder path, replace all the backward slashes (\) with the forward slashes (/).



```
C:\Users\Administrator\Documents\Kafka\kafka\config\zookeeper.properties

1 # Licensed to the Apache Software Foundation (ASF) under one or more
2 # contributor license agreements. See the NOTICE file distributed with
3 # this work for additional information regarding copyright ownership.
4 # The ASF licenses this file to You under the Apache License, Version 2.0
5 # (the "License"); you may not use this file except in compliance with
6 # the License. You may obtain a copy of the License at
7 #
8 #     http://www.apache.org/licenses/LICENSE-2.0
9 #
10 # Unless required by applicable law or agreed to in writing, software
11 # distributed under the License is distributed on an "AS IS" BASIS,
12 # WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 # See the License for the specific language governing permissions and
14 # limitations under the License.
15 # the directory where the snapshot is stored.
16 dataDir=C:/Users/Administrator/Documents/Kafka/zooData
17 # the port at which the clients will connect
18 clientPort=2181
```

54. Save the file.

*This concludes the lab.*

## Module 3: Create a Basic Process

### Lab 3-1: Create a Basic Process to Display User Input

#### Overview:

A Process Designer is a suite of tools for building cloud-based processes.

In this lab, you will learn to use the process designer to create a basic process that accepts text as an input and returns the same text as an output.

#### Objectives:

- Create a basic Process
- Invoke the Process

#### Duration:

15 Minutes

---

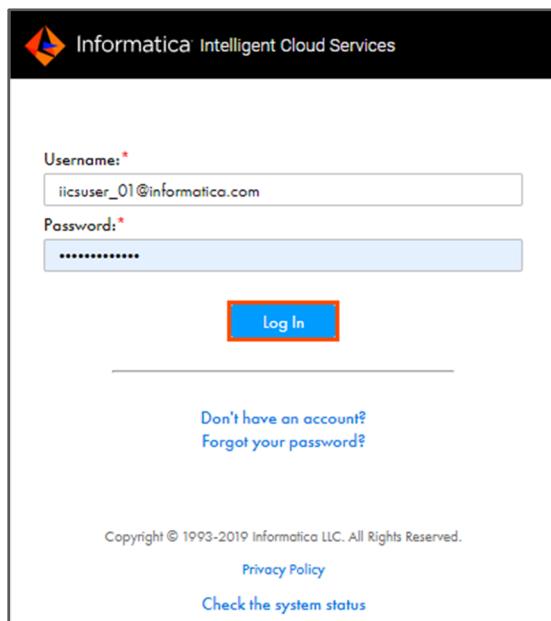
#### Tasks:

**Note:** The screen in your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

#### Create a Process:

1. In the Brower, open the **IICS Org** link, and login into it using your credentials.

**Note:** Use the link that you've bookmarked as **Informatica Cloud** while executing the Getting Started labs. To login into IICS, use the login credentials that you've created as part of the getting started lab guide – ‘Create an IICS Account’.



Informatica Intelligent Cloud Services

Username: \*  
iicsuser\_01@informatica.com

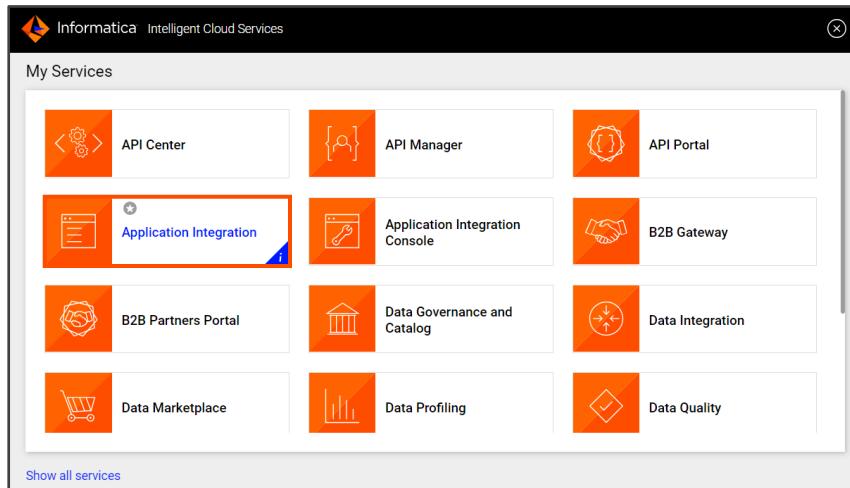
Password: \*  
\*\*\*\*\*

Log In

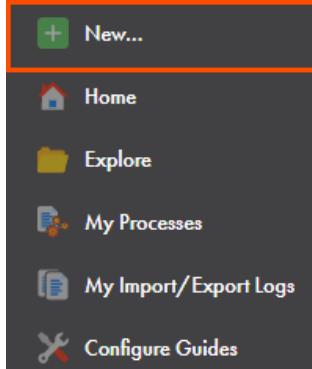
Don't have an account?  
Forgot your password?

Copyright © 1993-2019 Informatica LLC. All Rights Reserved.  
[Privacy Policy](#)  
[Check the system status](#)

- From the My Services window, select **Application Integration**.

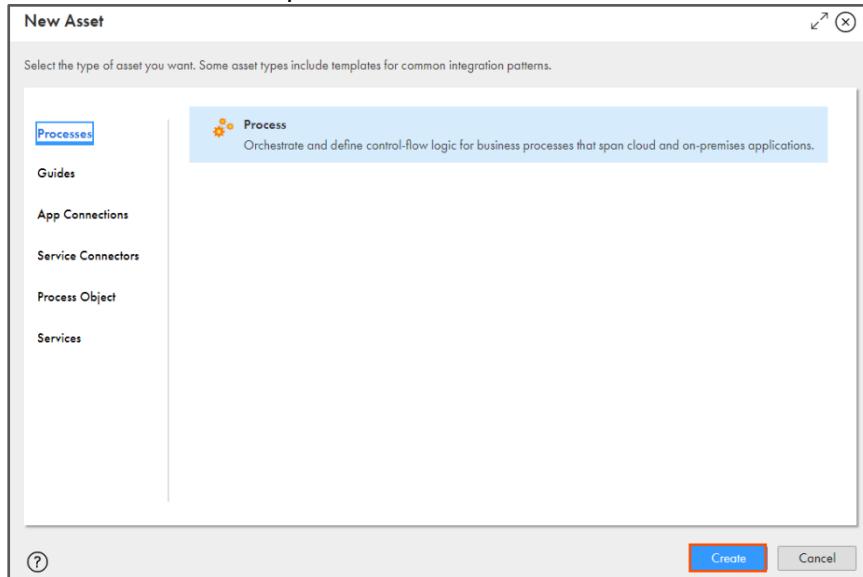


- From the navigation pane, select **New**.



**Recommendation:** Before you create a new asset, from the Explorer page, open your project, and click **New**. This sets the asset location to your working project by default.

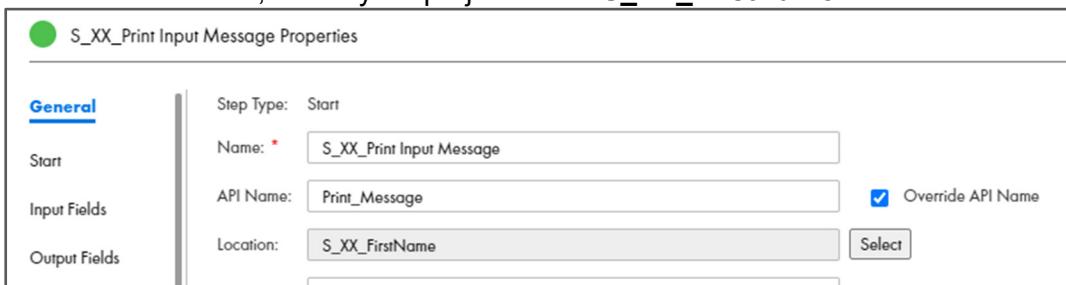
- Retain the **Process** option and click **Create**.



- In the General properties tab, in the Name field, enter **S\_XX\_Print Input Message**.

**Provide a customized API name for the process:**

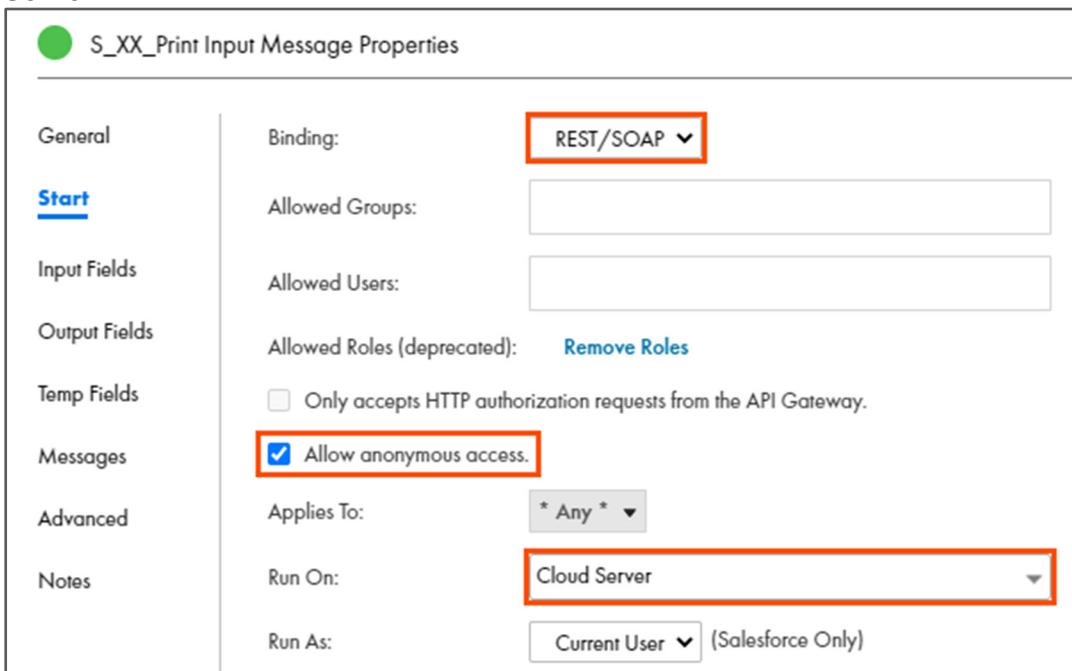
6. To provide a customized API Name, select the **Override API Name** option and specify the name. For example, provide **Print\_Message**.  
**Note:** The only allowed characters for the API Name are alphanumeric, underscore, and hyphen.
7. In the Location field, select your project folder **S\_XX\_FirstName**.



The screenshot shows the 'General' tab of the 'S\_XX\_Print Input Message Properties' dialog. The 'Step Type' is set to 'Start'. The 'Name' field contains 'S\_XX\_Print Input Message'. The 'API Name' field contains 'Print\_Message'. The 'Location' field contains 'S\_XX\_FirstName'. The 'Override API Name' checkbox is checked. A 'Select' button is also visible.

**Note:** This ensures that the process is stored in your project folder.

8. In the Start tab, retain the Binding option as **REST/SOAP**.
9. Select the **Allow anonymous access** option and retain the Run On option as **Cloud Server**.



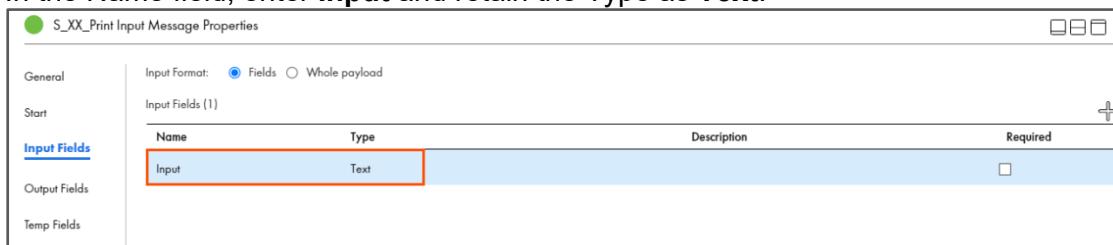
The screenshot shows the 'Start' tab of the 'S\_XX\_Print Input Message Properties' dialog. The 'Binding' dropdown is set to 'REST/SOAP'. Under 'Messages', the 'Allow anonymous access' checkbox is checked. Under 'Notes', the 'Run On' dropdown is set to 'Cloud Server'.

**Note:** The Allow anonymous access option enables all users to access the process.

Now, you need to create an input field of type text and assign a value (the user input) to it using the Assignment step. You will also create an output field that takes the input value from the Assignment step and displays it as an output. The output field must have the same type as the input field.

10. In the Input Fields tab, to add a new field, click .

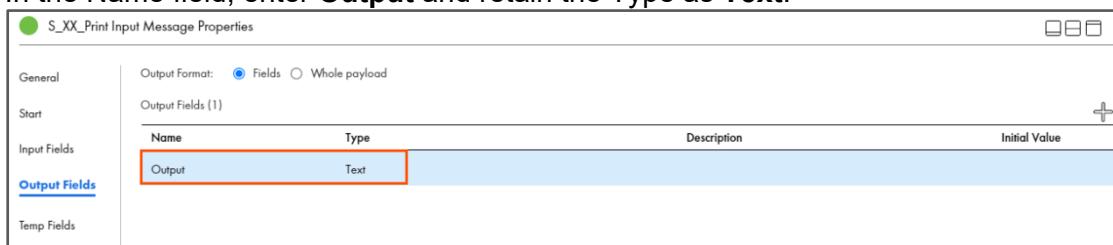
11. In the Name field, enter **Input** and retain the Type as **Text**.



| Name  | Type | Description | Required                 |
|-------|------|-------------|--------------------------|
| Input | Text |             | <input type="checkbox"/> |

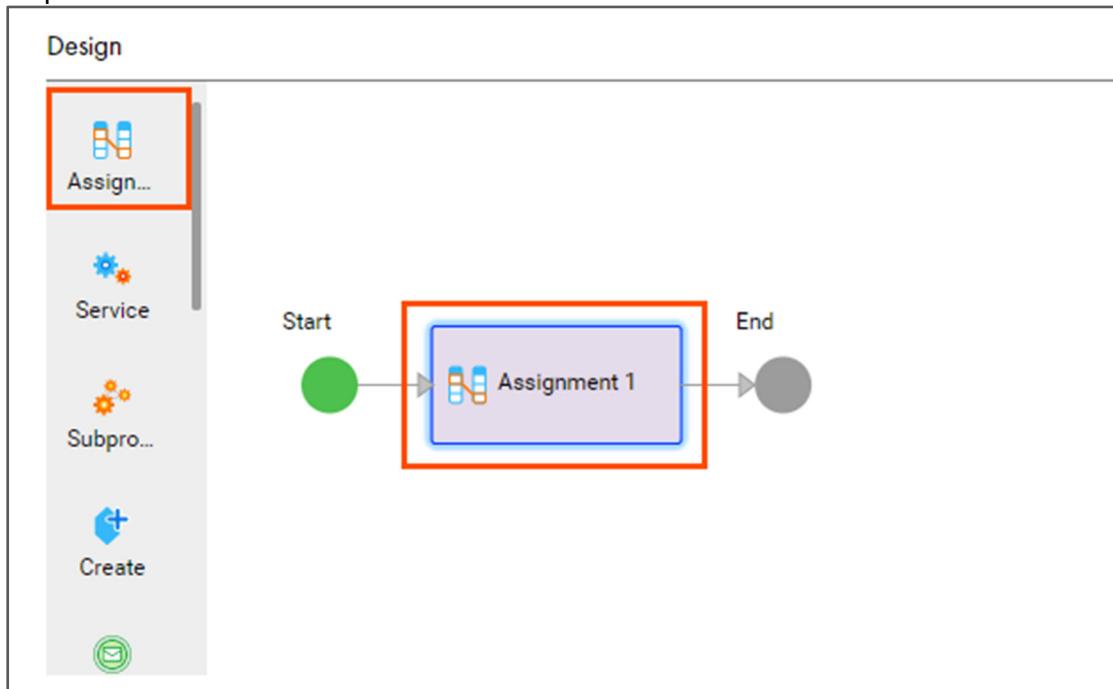
12. In the Output Fields tab, to add a new output field, click .

13. In the Name field, enter **Output** and retain the Type as **Text**.



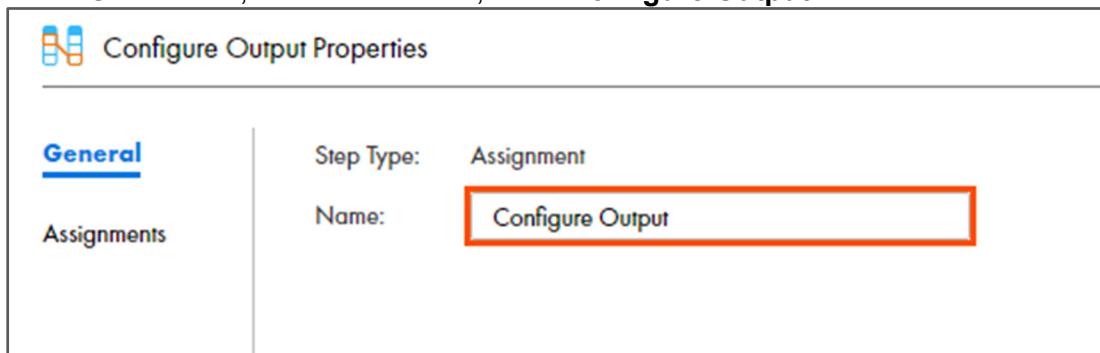
| Name   | Type | Description | Initial Value |
|--------|------|-------------|---------------|
| Output | Text |             |               |

14. From the palette, drag and drop an **Assignment** step between the Start and the End step on the canvas.



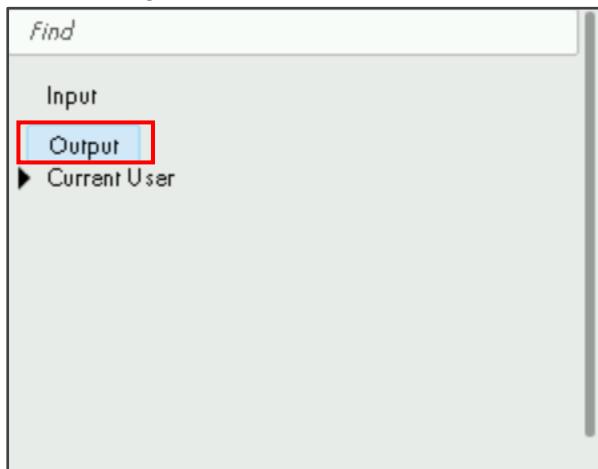
15. Select the **Assignment** step.

16. In the General tab, in the Name field, enter **Configure Output**.



The screenshot shows the 'Configure Output Properties' dialog with the 'General' tab selected. On the left is a sidebar with 'Assignments'. The main area shows 'Step Type: Assignment' and a 'Name:' field containing 'Configure Output', which is highlighted with a red border.

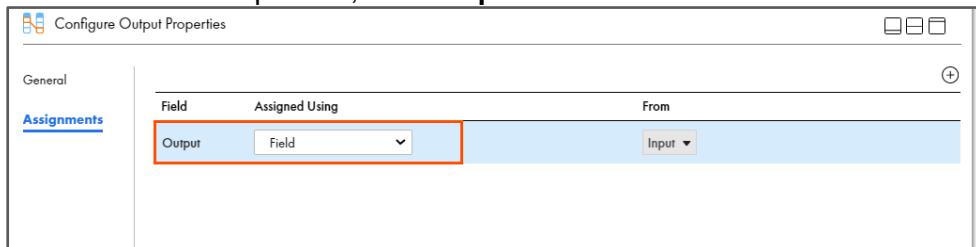
17. In the Assignments tab, click the plus (+) icon, and select **Output**.



**Note:** This field takes values from the input field and displays them as an output.

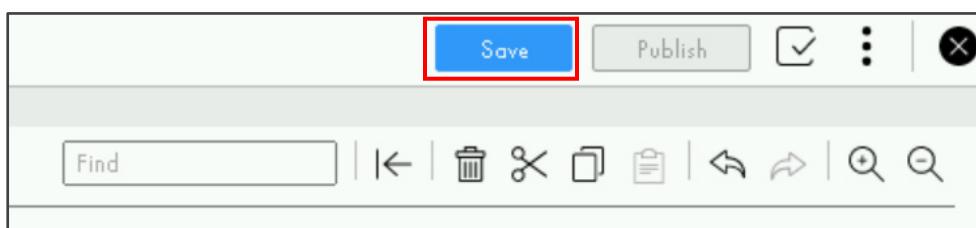
18. From the **Assigned Using** drop-down, select **Field**.

19. From the **From** drop-down, select **Input**.



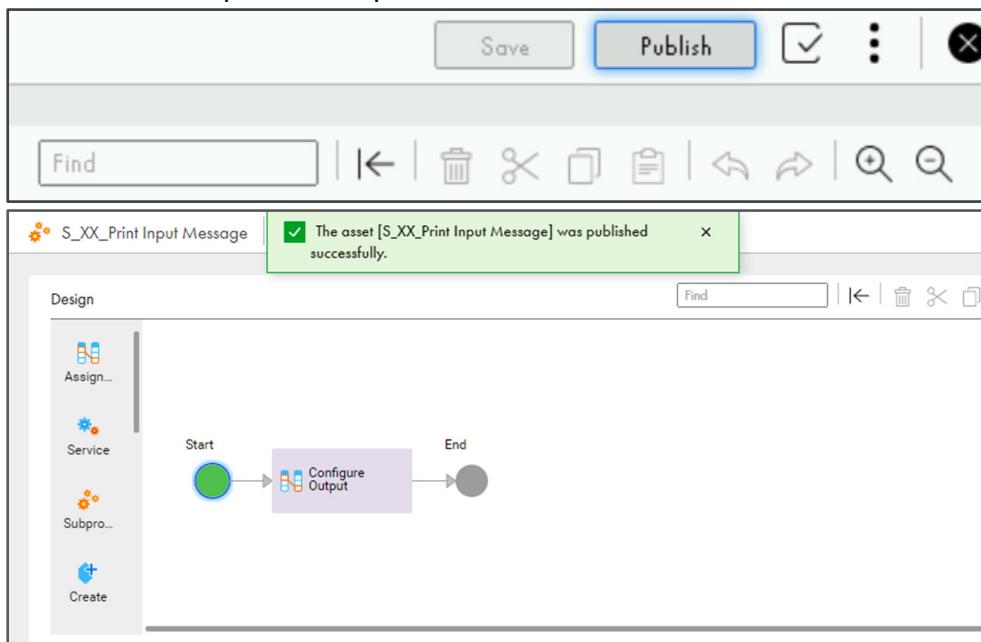
The screenshot shows the 'Assignments' tab with a table. The first row has columns 'Field' (containing 'Output'), 'Assigned Using' (containing 'Field', highlighted with a red border), and 'From' (containing 'Input').

20. Click **Save**.



The screenshot shows the bottom toolbar of the dialog. It includes a 'Save' button (highlighted with a red border), a 'Publish' button, a checked checkbox, a vertical ellipsis, and a close button. Below the toolbar is a navigation bar with icons for Find, Back, Forward, and Search.

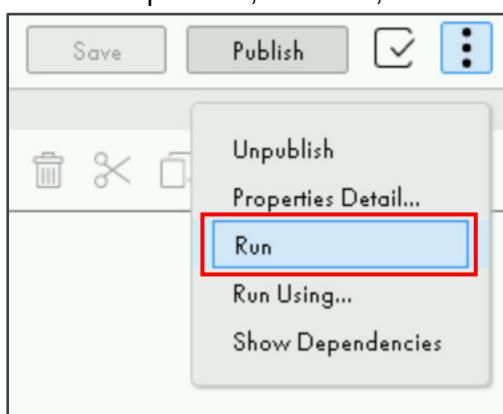
21. Click **Publish** to publish the process.



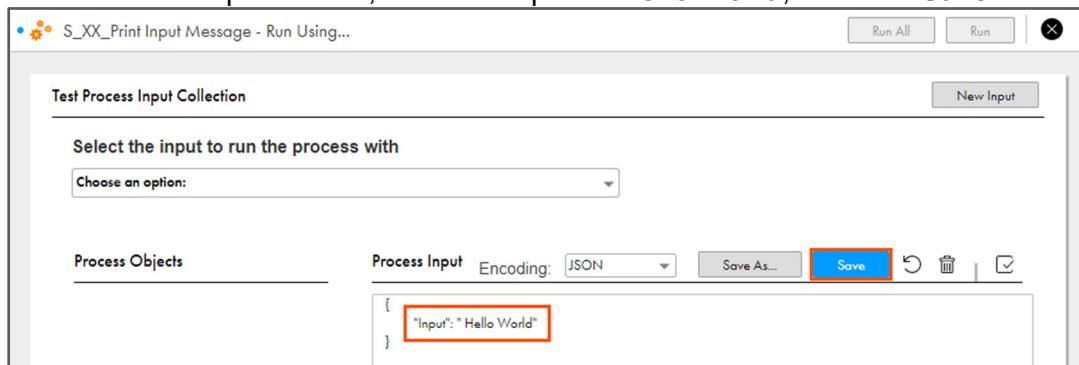
The process is published successfully.

### Invoke the Process:

22. To run the process, click , and select **Run**.



23. In the Process Input section, enter the input as **Hello World**, and click **Save**.



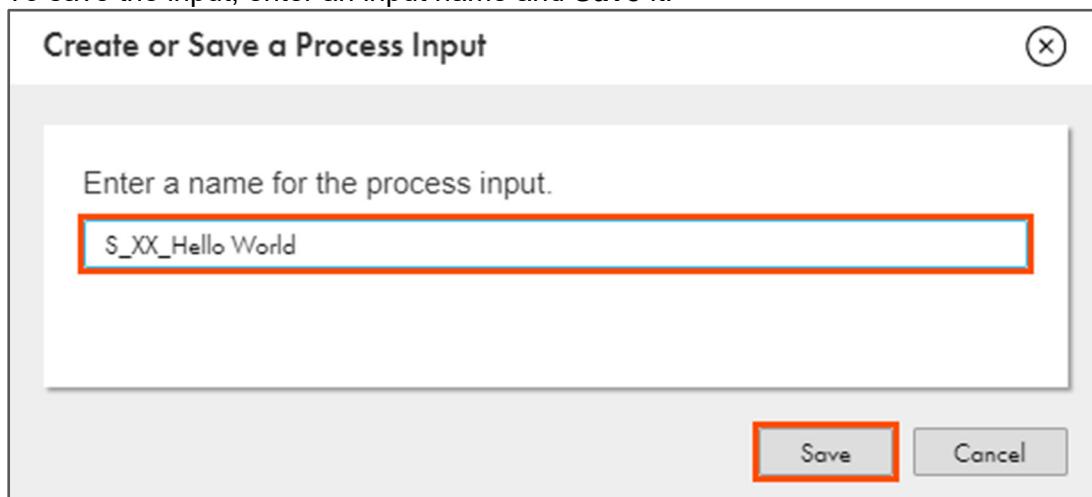
The screenshot shows the 'Test Process Input Collection' interface. In the 'Process Input' section, the 'Encoding' is set to 'JSON'. A JSON object is displayed in a text area:

```
{
  "Input": "Hello World"
}
```

The 'Save' button is highlighted with a red box.

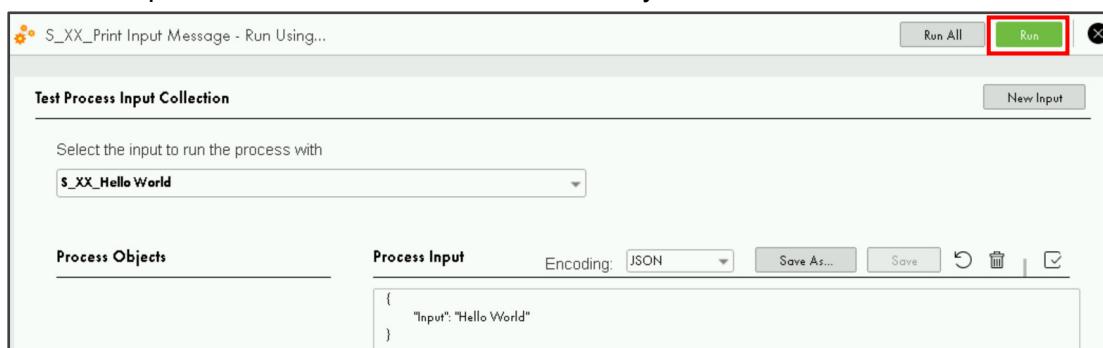
**Note:** This input will be available as the output.

24. To save the input, enter an input name and **Save** it.



The dialog box is titled 'Create or Save a Process Input'. It contains a text input field labeled 'Enter a name for the process input.' with the value 'S\_XX\_Hello World' highlighted by a red box. At the bottom are 'Save' and 'Cancel' buttons, with 'Save' highlighted by a red box.

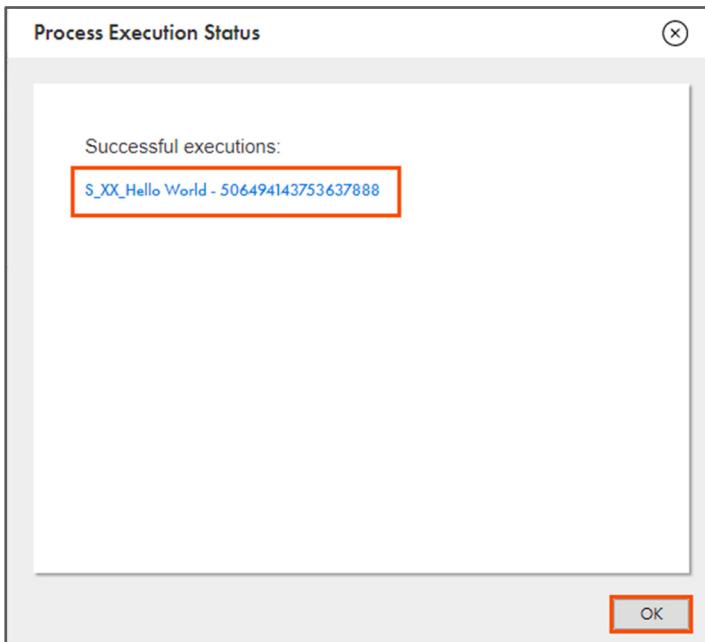
25. After the input is added and validated successfully, click **Run**.



The screenshot shows the 'Test Process Input Collection' interface again. The 'Run' button is highlighted with a green box. The 'Process Input' section shows the same JSON object as before:

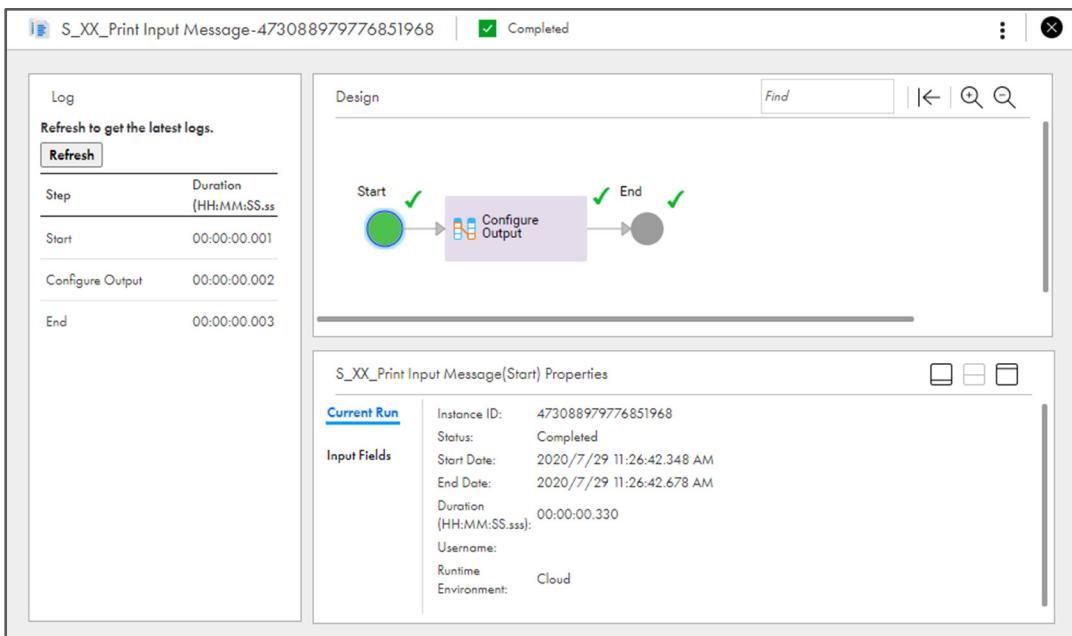
```
{
  "Input": "Hello World"
}
```

26. To check the Process Execution Status, click the **Process Execution** link, and then click **OK**.



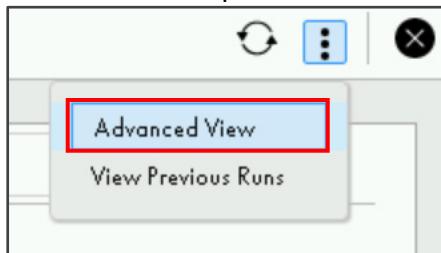
**Note:** If you do not wish to check the status, click **OK**.

27. The execution details are as shown below:



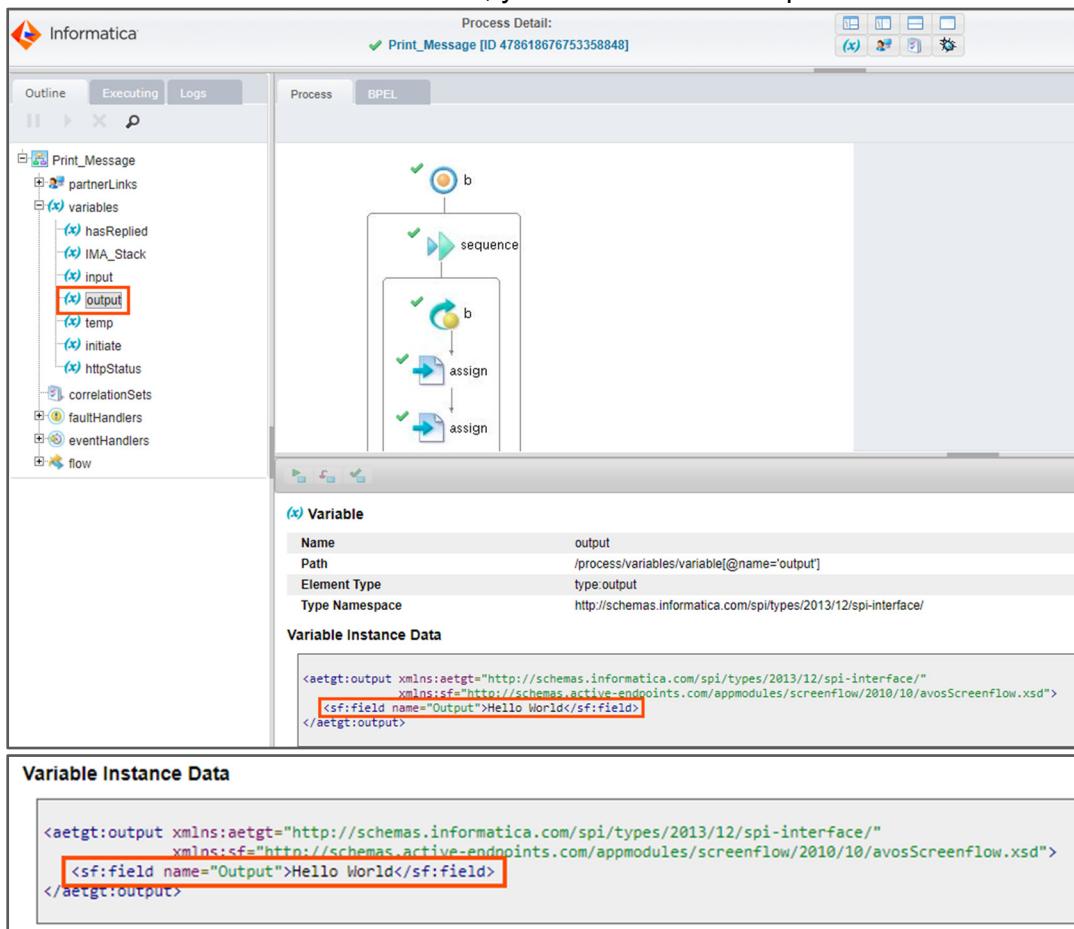
**Note:** If you are not able to see this screen, click **Refresh**.

28. To check the output, from the Actions menu, select **Advanced View**.



29. Expand the **variables** section and select **output**.

In the Variable Instance Data section, you can check the output.



| Name   | Path  | Element Type | Type Namespace  |
|--------|---|--------------|---|
| output | /process/variables/variable[@name='output'] | type:output  | http://schemas.informatica.com/spi/types/2013/12/spi-interface/ |

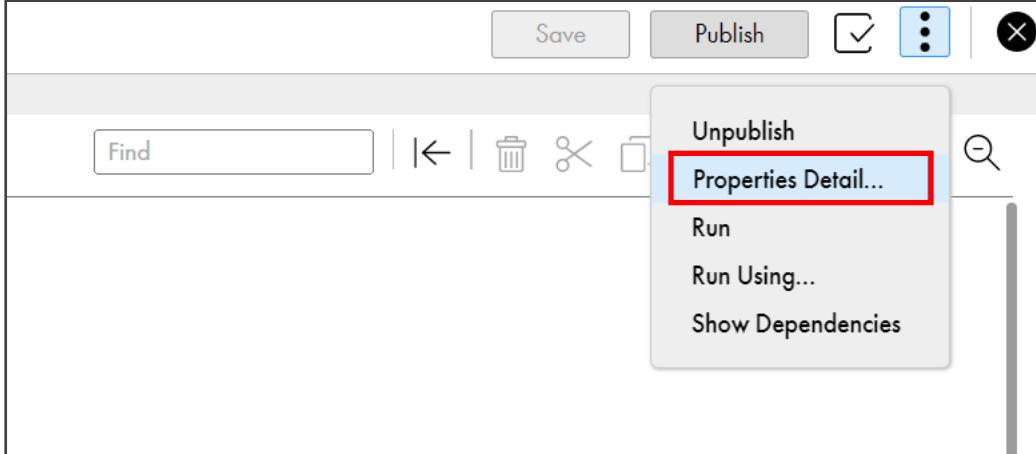
| Variable Instance Data   |  |
|--|--|
| <pre>&lt;aetgt:output xmlns:aetgt="http://schemas.informatica.com/spi/types/2013/12/spi-interface/"<br/>    xmlns:sf="http://schemas.active-endpoints.com/appmodules/screenflow/2010/10/avosScreenflow.xsd"&gt;<br/>    &lt;sf:field name="Output"&gt;Hello World&lt;/sf:field&gt;<br/>&lt;/aetgt:output&gt;</pre> |  |

30. Close all the instances of the process and the Advanced View tab.

You have successfully created a basic process and invoked it from the interface.

## Invoke the Process Using Service URL

31. In CAI, navigate back to the **S\_XX\_Print Input Message** process, and click .
32. Select **Properties Detail**.



33. From the Properties Detail page, to copy the Service URL, click **Copy**.

Properties Detail for S\_XX\_Print Input Message (x)

|                     |   |
|---------------------|---|
| Unique Name:        | Print_Message                                 |
| Location:           | S_XX_FirstName                                |
| Publication Status: | <input checked="" type="checkbox"/> Published |
| Published On:       | 2023-02-07 10:21                              |
| Published By:       |   |
| Applies To:         | * Any *                                       |

---

Endpoints

|                   |   |   |
|-------------------|---|---|
| Service URL:      | <a href="https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfIEqfZgh7aVGifF/Print_Message">https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfIEqfZgh7aVGifF/Print_Message</a>     |  |
| SOAP Service URL: | <a href="https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/soap/hnLOSoRfIEqfZgh7aVGifF/Print_Message">https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/soap/hnLOSoRfIEqfZgh7aVGifF/Print_Message</a> |  |

[View Swagger File](#) [View OpenAPI 3.0 File](#) [View WSDL File](#)

 Close

34. Open the Notepad and enter the copied Service URL.

35. In a web browser, enter the following URL and press Enter:

**<Service URL>?Input=Hello World**

**Note:** Replace <Service URL> with the URL that you have copied in Notepad. **DO NOT CLOSE THE NOTE PAD.**

For Example: [https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRiEqfZgh7aWGiF/Print\\_Message?Input=Hello%20World](https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRiEqfZgh7aWGiF/Print_Message?Input=Hello%20World)

36. Observe that you get the output as shown below:

```
{"Output": "Hello World"}
```

37. Close the browser.

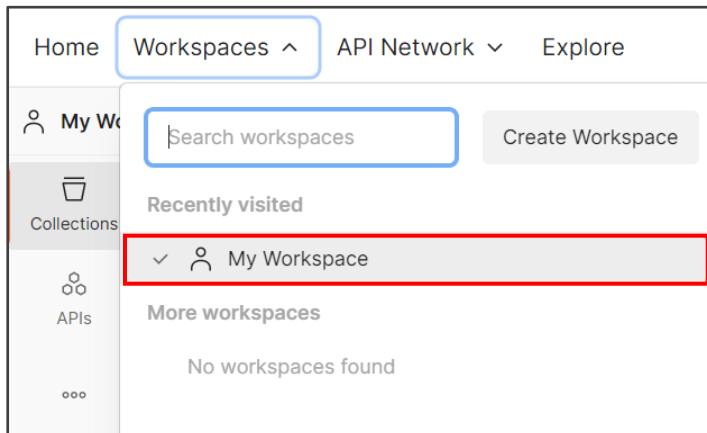
### Invoke the Process Using Postman

38. Open a web browser and enter the following link to open Postman:

<https://identity.getpostman.com/>

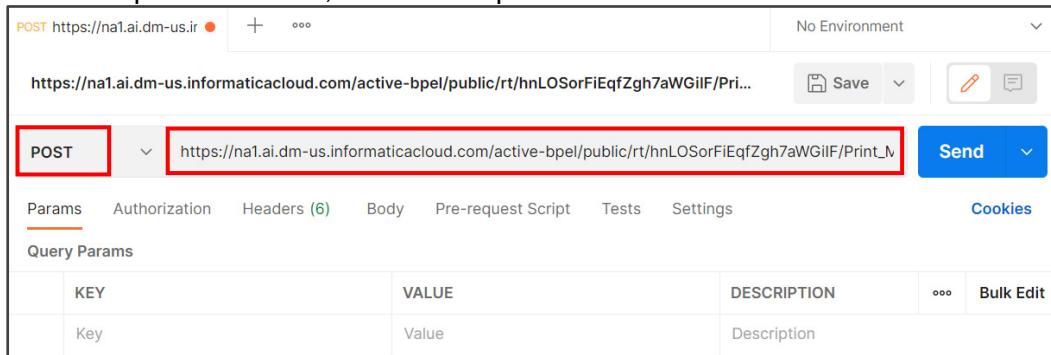
39. Log in to Postman using your credentials.

40. From the Workspace drop-down, select **My Workspace**.



41. In the Workspace section, from the drop-down, select **POST**.

42. In the Request URL field, enter the copied Service URL.



The screenshot shows the Postman interface with the following details:

- Method: POST
- URL: https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfEqfZgh7aWGiF/Print\_M
- Params tab selected
- Query Params table:
 

| KEY | VALUE | DESCRIPTION | ... | Bulk Edit |
|-----|-------|-------------|-----|-----------|
| Key | Value | Description | ... |           |

43. Select the **Authorization** tab.

44. From the TYPE drop-down, select **No Auth**.



The screenshot shows the Postman interface with the following details:

- Method: POST
- URL: https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfEqfZgh7aWGiF/Print\_M
- Authorization tab selected
- Type dropdown set to "No Auth"

45. Select the **Headers** tab.

46. Enter Key and Value as shown in the table below:

| Key          | Value            |
|--------------|------------------|
| Content-Type | application/json |



The screenshot shows the Postman interface with the following details:

- Method: POST
- URL: https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfEqfZgh7aWGiF/Print\_M
- Headers tab selected
- Headers table:
 

| KEY  | VALUE            | DESCRIPTION | ... | Bulk Edit | Presets |
|--|------------------|-------------|-----|-----------|---------|
| <input checked="" type="checkbox"/> Content-Type | application/json |             | ... |           | Presets |
| Key  | Value            | Description | ... |           |         |

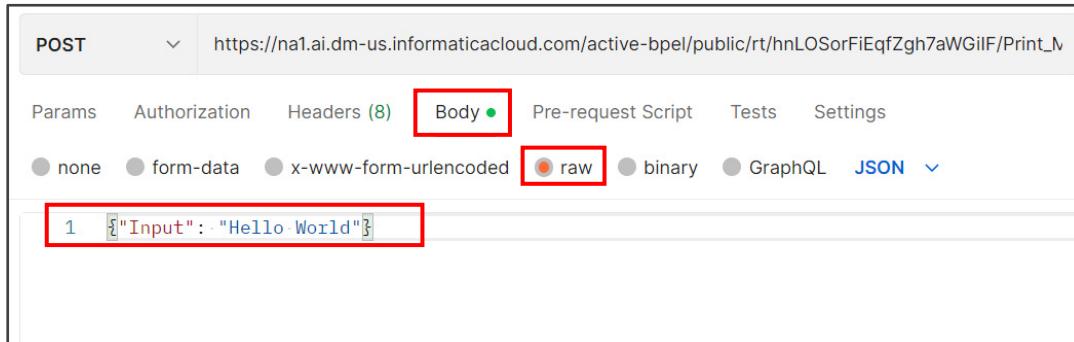
47. Select the **Body** tab and select **raw**.

48. Enter the following text body:

{"Input": "Hello World"}

OR

From the Desktop, navigate to the Lab Prep Files folder and open the notepad file named **01\_LabGuide\_CAI\_CreateABasicProcess\_3-1**. Copy the command mentioned in the notepad and paste it in the body field.



The screenshot shows the Postman application interface. A POST request is being made to the URL <https://na1.ai.d...>. The 'Body' tab is selected, and the 'raw' option is chosen under the dropdown. The JSON input field contains the following code:

```
1 {"Input": "Hello World"}
```

49. To view the response, click **Send**.

50. Verify that the status of the response is **200 OK**.

51. Examine the details in the response.



The screenshot shows the Postman interface after sending the request. The status bar at the top right indicates a **200 OK** response. In the 'Body' tab, the response is displayed in JSON format:

```
1 {  
2   "Output": "Hello World"  
3 }
```

---

*This concludes the lab.*

## Module 4: Process Objects

### Lab 4-1: Create Process Objects

#### **Overview:**

Process objects combine sets of structured data into groups called objects. These objects can be used in processes. The named process objects help you identify the inputs and outputs of a Process.

This is **part 1** of the set of three labs (4-1, 7-2, and 10-1). The primary objective of this exercise is to obtain the conversion rates of specific currencies such as United States Dollars (USD), Australian Dollars (AUD), Great Britain Pounds (GBP), and Canadian Dollars (CAD).

In this lab, you will create four Process Objects to collect input values and input types to request conversion and get a response output. You will use these Process Objects in the ExchangeRate process which will be explained in the subsequent modules.

**Note:** This lab is continued in **Lab 7-2: Create a JDBC Connection** of Module 7 and **Lab 10-1: Invoke a Synchronous Web Service Call** of Module 10.

#### **Objectives:**

- Create a Process Object – BaseRatePO
- Create a Process Object – ConversionRatePO
- Create a Process Object – ConversionRequestPO
- Create a Process Object – ConversionResponsePO

#### **Duration:**

15 Minutes

---

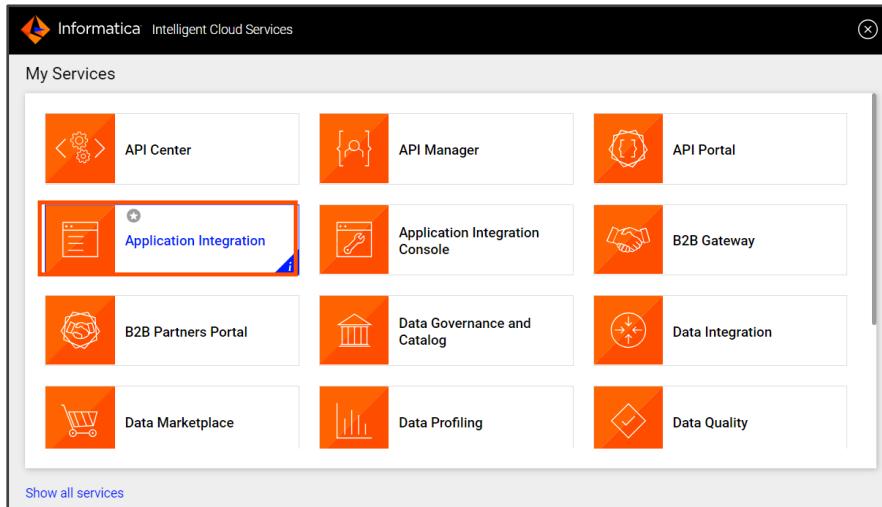
#### **Tasks:**

**Note:** The screen in your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

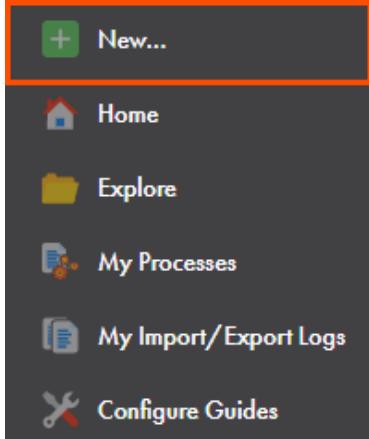
## Create a Process Object – BaseRatePO

This process object includes the type and value of the currency. It will be used as a reference later in the ConversionRequestPO and ConversionResponsePO process objects.

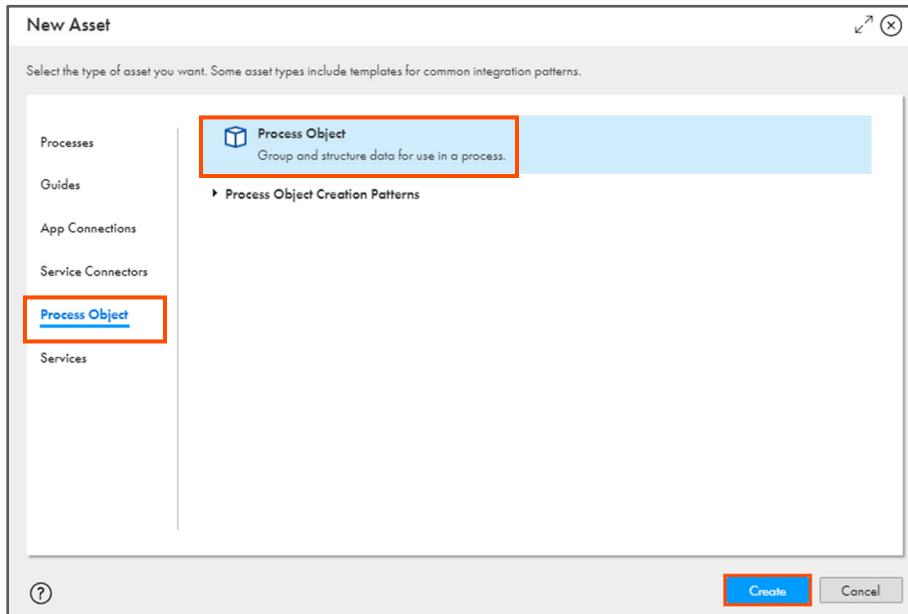
1. Login into IICS and navigate to the **Application Integration** service.



2. From the Navigation pane, click **New**.



- In the New Asset window, from the Process Object tab, select **Process Object**, and click **Create**.



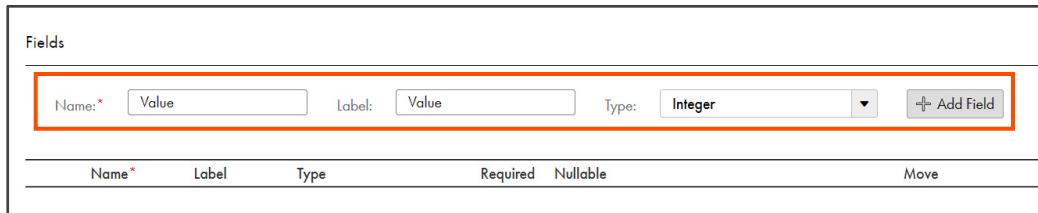
- In the Process Object Properties page, in the Name field, enter **BaseRatePO**.
- In the Location field, select your project folder **S\_XX\_FirstName**.



The screenshot shows the 'Properties' dialog box for the 'BaseRatePO' process object. It has three fields: 'Name:' with value 'BaseRatePO', 'Location:' with value 'S\_XX\_FirstName' (which is highlighted with a red box), and 'Description:' which is empty. There is also a 'Browse' button next to the Location field.

For BaseRatePO Process two fields are required – **Value** and **Type**.

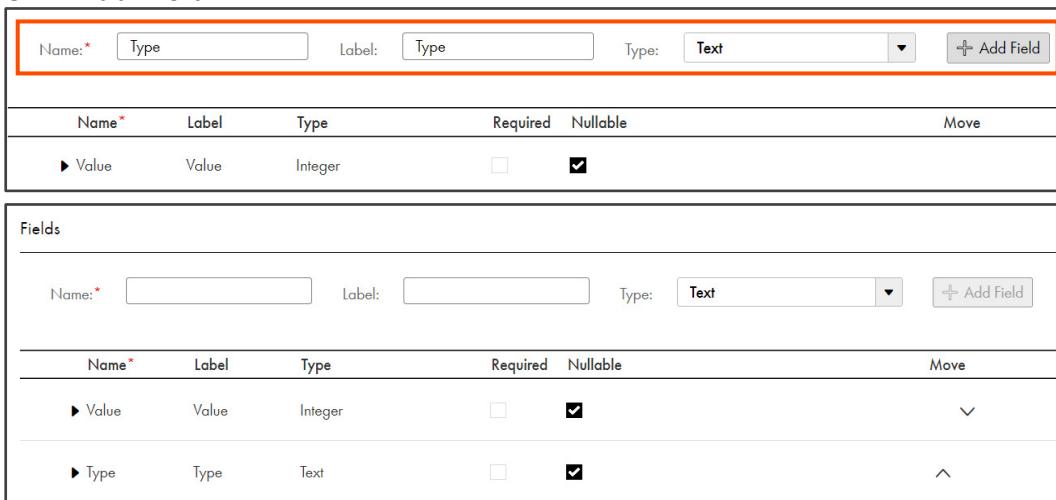
- For the first field, enter Name and Label as **Value**.
- From the Type drop-down, select **Integer**.
- Click **Add Field**.



The screenshot shows the 'Fields' dialog box. At the top, there's a row with 'Name:' (Value), 'Label:' (Value), 'Type:' (Integer), and a 'Add Field' button (which is highlighted with a red box). Below this is a table with columns: Name\*, Label, Type, Required, Nullable, and Move. A new row is being added, indicated by a red border around the first column.

- For the second field, enter Name and Label as **Type**.
- From the Type drop-down, select **Text**.

**11. Click Add Field.**



The screenshot shows the Informatica interface for adding fields to a process object. At the top, there's a header with 'Name:', 'Type', 'Label', 'Type', and a dropdown set to 'Text'. Below this is a 'Move' button with a plus sign. A table below lists fields: 'Value' (Type: Integer, Required: unchecked, Nullable: checked) and 'Type' (Type: Text, Required: unchecked, Nullable: checked). The 'Type' field has a downward arrow icon to its right. The bottom section is titled 'Fields' and contains another 'Add Field' dialog with similar fields: 'Name:', 'Label', 'Type', 'Required', 'Nullable', and a 'Move' button. The 'Type' field here also has a downward arrow icon.

**12. Save the Process Object.**



The screenshot shows the 'Save' dialog for the process object 'BaseRatePO1'. It indicates the object is 'Valid'. The main area contains fields for 'Name:' (BaseRatePO), 'Location:' (S\_XX\_FirstName), and 'Description:' (empty). A 'Save' button is at the top right.

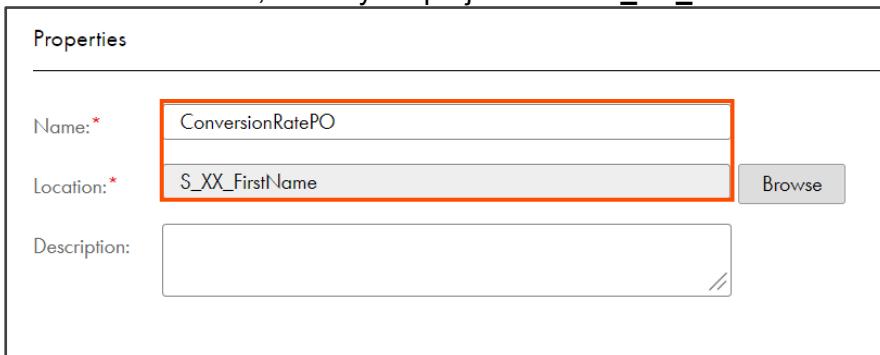
You have successfully designed a Process Object. Similarly, you will design the other three Process Objects.

### Create a Process Object – ConversionRatePO

This process object is used to display the conversion rates as compared to the base rates. It includes the fields – Value, Rate, and RateType. This process object will also be used as a reference later in the ConversionRequestPO and ConversionResponsePO process objects.

**13. Create another process object and in the Name field, enter **ConversionRatePO**.**

**14. In the Location field, select your project folder **S\_XX\_FirstName**.**



The screenshot shows the 'Properties' dialog for creating a new process object. It has sections for 'Name:' (ConversionRatePO), 'Location:' (S\_XX\_FirstName), and 'Description:' (empty). The 'Name:' field is highlighted with a red border. A 'Browse' button is next to the 'Location:' field.

For this Process Object, three fields are required – **Value**, **Rate**, and **RateType**.

15. Add the first field with the Name and Label as **Value** and the Type as **Number**.
16. For the second field, enter Name and Label as **Rate** and select the Type as **Text**.
17. For the third field, enter Name and Label as **RateType** and select the Type as **Text**.

| Fields     |          |        |                          |                                     |      |
|------------|----------|--------|--------------------------|-------------------------------------|------|
| Name:      | Label:   | Type:  | Required                 | Nullable                            | Move |
| ▶ Value    | Value    | Number | <input type="checkbox"/> | <input checked="" type="checkbox"/> | ▼    |
| ▶ Rate     | Rate     | Text   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | ▼ ^  |
| ▶ RateType | RateType | Text   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | ^    |

18. Save the Process Object.

### Create a Process Object – ConversionRequestPO

This process object uses BaseRatePO and ConversionRatePO to obtain the base rate and conversion rates of the requested currency. This will be used as an input into the process.

19. Create another process object and in the Name field, enter **ConversionRequestPO**.
20. In the Location field, select your project folder **S\_XX\_FirstName**.

| Properties   |                     |        |
|--------------|---------------------|--------|
| Name:        | ConversionRequestPO |        |
| Location:    | S_XX_FirstName      | Browse |
| Description: |                     |        |

You will now add two object fields that refer to the two previously created Process Objects – BaseRatePO and ConversionRatePO.

21. Add the first field with the Name and Label as **BaseRate** and the Type as **Reference**.

22. Click **Add Field**.

| Fields  |          |        |          |       |           |
|---|----------|--------|----------|-------|-----------|
| Name:   | BaseRate | Label: | BaseRate | Type: | Reference |
| <input style="border: 2px solid red;" type="button" value="Add Field"/> |          |        |          |       |           |

A new field is created and in the Type field, a new option **Reference** is available.

| Name *                                   | Label    | Type      | Required                            | Nullable                 |
|--|----------|-----------|-------------------------------------|--------------------------|
| BaseRate                                 | BaseRate | Reference | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| <input type="button" value="Select..."/> |          |           |                                     |                          |

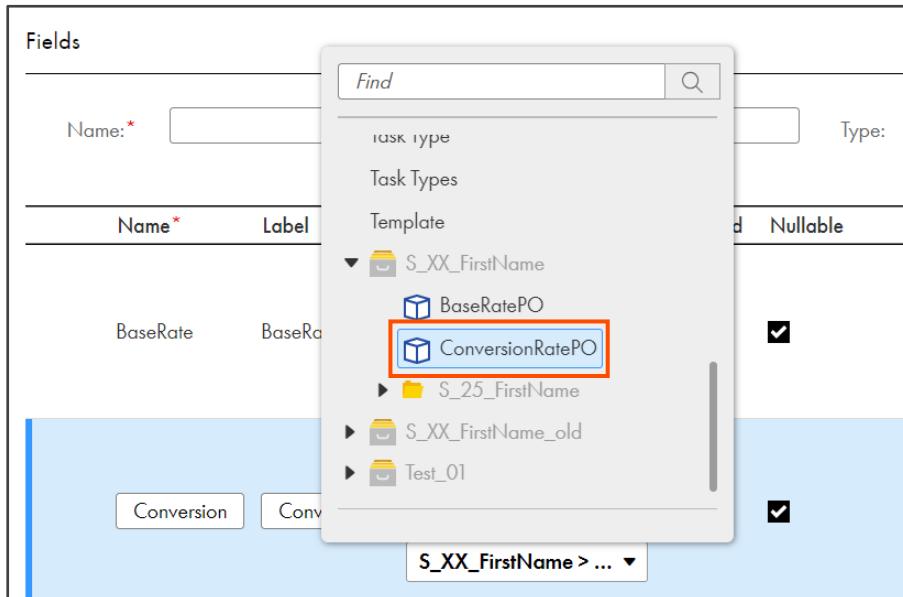
23. Under the Type field, from the drop-down, expand your project folder **S\_XX\_FirstName**, and select **BaseRatePO**.

| Name *   | Label    | Type   | Required                            | Nullable                 |
|--|----------|--|-------------------------------------|--------------------------|
| BaseRate   | BaseRate | <input type="button" value="Select..."/><br><span style="border: 2px solid red; padding: 2px;">BaseRatePO</span> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| <input type="button" value="S_XX_FirstName &gt; ..."/> |          |  |                                     |                          |

24. Add the second field with the Name and Label as **Conversion** and the Type as **Reference**.

| Fields                                   |            |        |            |       |           |
|--|------------|--------|------------|-------|-----------|
| Name:                                    | Conversion | Label: | Conversion | Type: | Reference |
| <input type="button" value="Add Field"/> |            |        |            |       |           |

25. Under the Type field, from the drop-down, expand your project folder **S\_XX\_FirstName**, and select **ConversionRatePO**.



**Note:** ConversionRatePO might take time to appear in the drop-down. If you can't see ConversionRatePO in the drop-down, log out from IICS and log in again. If you still can't see it in the drop-down, clear the browser cache.

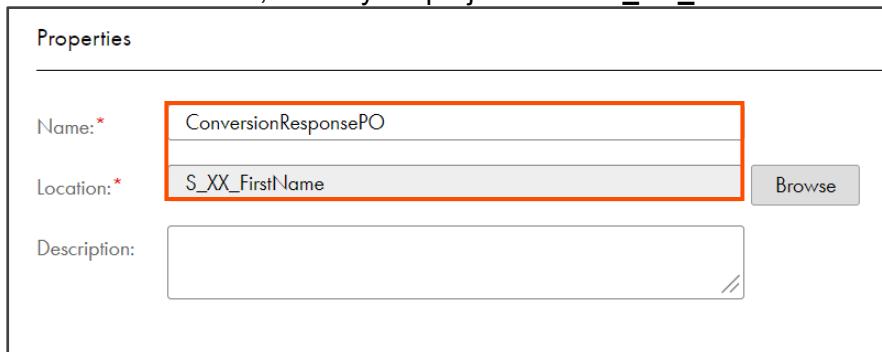
26. Save the Process Object.

### Create a Process Object – ConversionResponsePO

This process object is used to display the conversion rates of the previous request.

27. Create another process object and in the Name field, enter **ConversionResponsePO**.

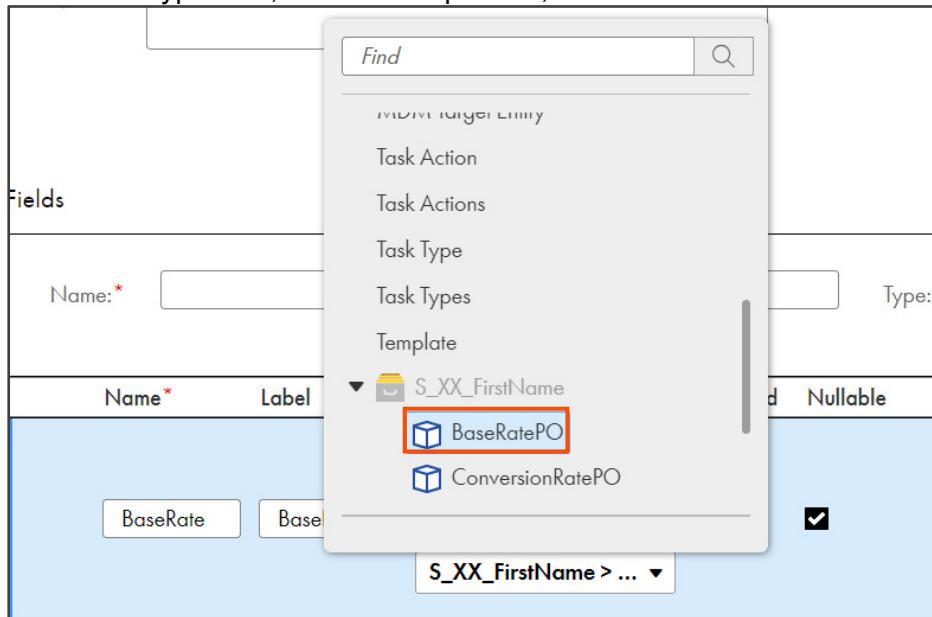
28. In the Location field, select your project folder **S\_XX\_FirstName**.



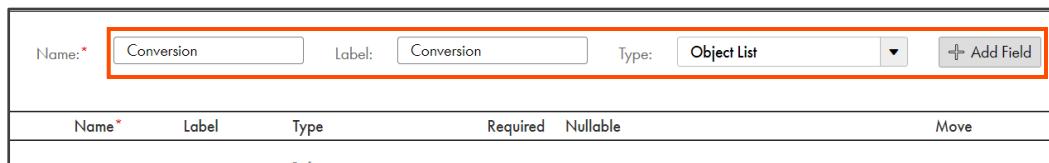
| Properties   |                      |
|--------------|----------------------|
| Name:*       | ConversionResponsePO |
| Location:*   | S_XX_FirstName       |
| Description: |                      |

29. Add the first field with the Name and Label as **BaseRate** and the Type as **Reference**.

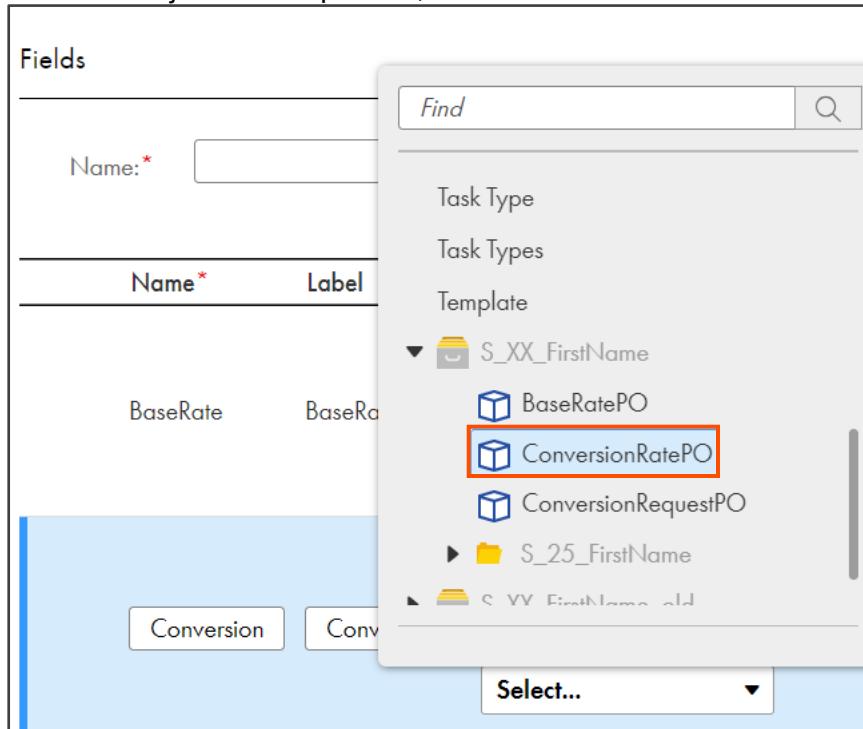
30. Under the Type field, from the drop-down, select **BaseRatePO**.



31. Add the second field with the Name and Label as **Conversion** and the Type as **Object List**.



32. From the Object List drop-down, select **ConversionRatePO**.



33. **Save** the Process Object.

---

*This concludes the lab.*

## Module 5: Service Connectors

### Lab 5-1: Create a Service Connector

#### **Overview:**

A service connector allows you to connect to third-party services such as the Thomas Bayer service.

This lab is **part 1** of a set of three labs. The overall goal of the exercise is to obtain customer and order details such as customer name and order total. The process includes – creating a service connector, creating a connection using the service connector, and then creating a process to obtain the desired output.

In this lab, you will create a Service Connector from a publicly available REST Web Service.

**Note:** This lab is continued in **Lab 7-1: Create an App Connection using Service Connector** and **Lab 7-3: Create a Customer and Order Details Process** of Module 7.

#### **Objectives:**

- Create a new Service Connector
- Define Service Connector input parameters
- Define a Service Connector Action and Binding
- Test the Service Connector

#### **Duration:**

25 Minutes

---

## Tasks

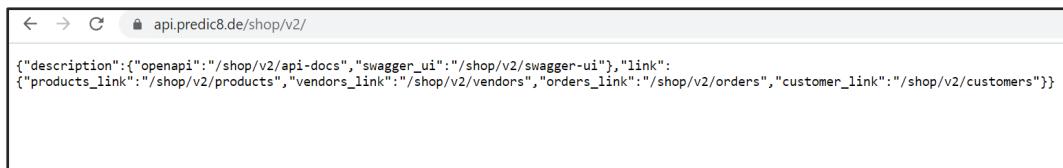
### Examine the web service:

- To examine the list of top-level resources available in the web service, enter the following URL in the browser:

<https://api.predic8.de/shop/v2/>

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **03\_LabGuide\_CAI\_Create a Service Connector\_5-1**. Copy the command mentioned under **Step A** and paste it in the browser.



```
{"description": {"openapi": "/shop/v2/api-docs", "swagger_ui": "/shop/v2/swagger-ui"}, "link": {"products_link": "/shop/v2/products", "vendors_link": "/shop/v2/vendors", "orders_link": "/shop/v2/orders", "customer_link": "/shop/v2/customers"}}
```

### Examine the Customer resource:

- To examine the Customer resource, enter the following URL in a new tab on browser:

<https://api.predic8.de/shop/v2/customers/>

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **03\_LabGuide\_CAI\_Create a Service Connector\_5-1**. Copy the command mentioned under **Step B** and paste it in the browser.



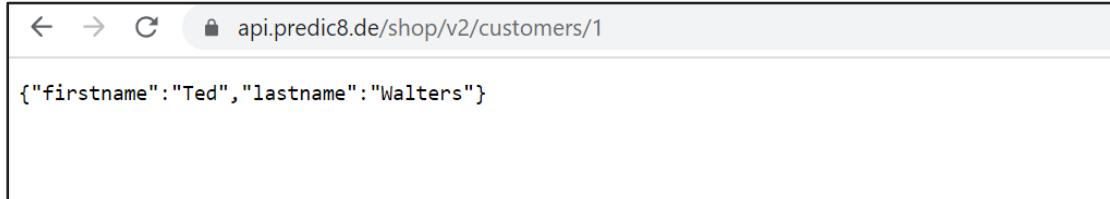
```
{"meta": {"count": 6, "start": 1, "limit": 10}, "customers": [{"id": 6, "name": "Pedro Santiago", "self_link": "/shop/v2/customers/6"}, {"id": 5, "name": "Carlos Sánchez", "self_link": "/shop/v2/customers/5"}, {"id": 4, "name": "Garrick Barnett", "self_link": "/shop/v2/customers/4"}, {"id": 3, "name": "Larry Lawson", "self_link": "/shop/v2/customers/3"}, {"id": 2, "name": "Helen Andrus", "self_link": "/shop/v2/customers/2"}, {"id": 1, "name": "Ted Walters", "self_link": "/shop/v2/customers/1"}]}
```

- To examine the details of a customer, enter the following URL in a new tab on browser:

<https://api.predic8.de/shop/v2/customers/1>

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **03\_LabGuide\_CAI\_Create a Service Connector\_5-1**. Copy the command mentioned under **Step C** and paste it in the browser.

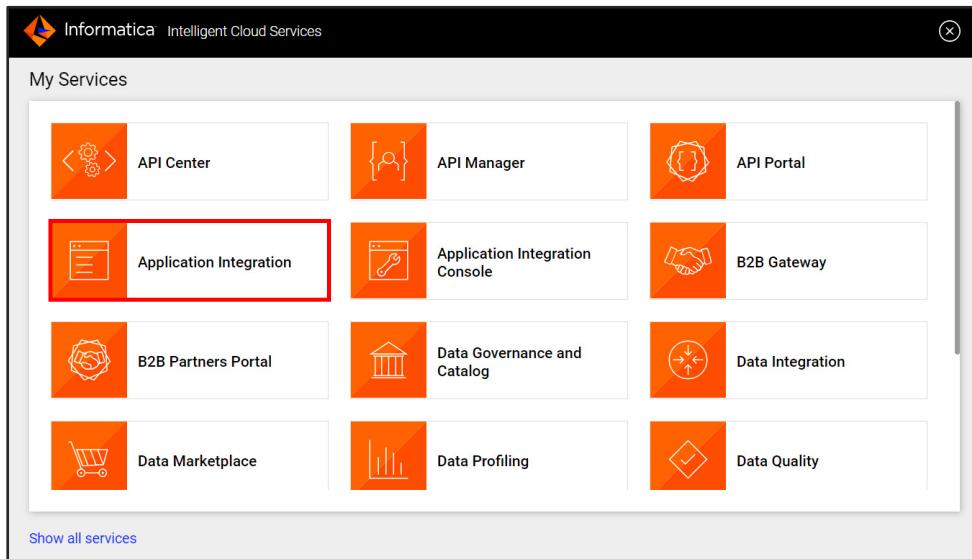


```
{"firstname": "Ted", "lastname": "Walters"}
```

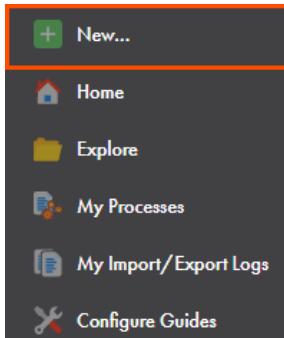
**Note:** The updated URL passes the Order id. A response returns the information about the customer with **Order id=1**. The API responds with two fields: firstname and lastname.

### Create a new Service Connector:

- Log into IICS and access the Application Integration service.

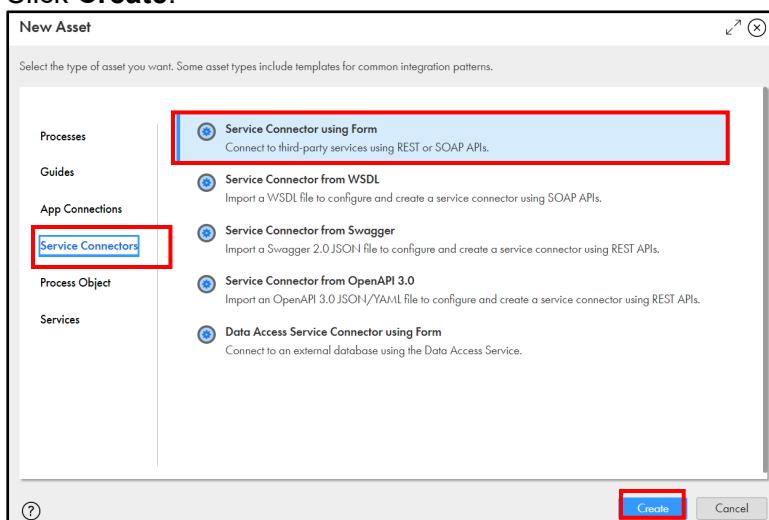


- To create a new Service Connector, from the navigation pane, click **New**.



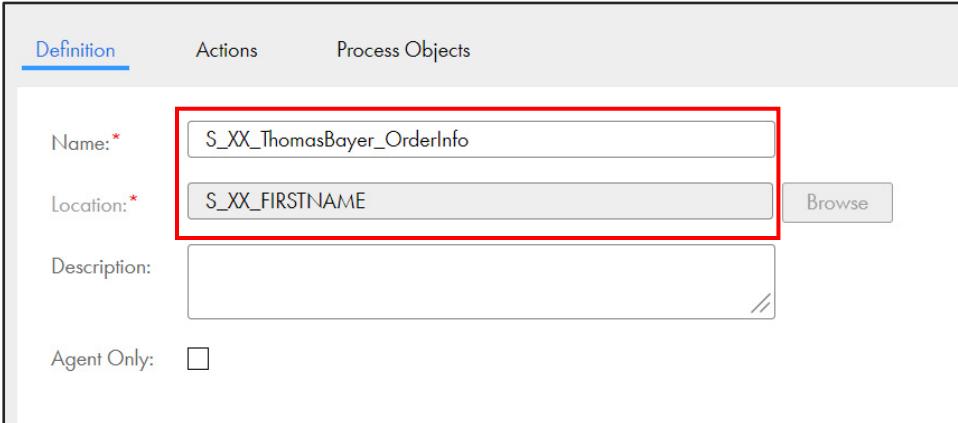
- From the Service Connectors tab, select **Service Connector using Form**.

- Click **Create**.



**Note:** A new Service Connector page appears.

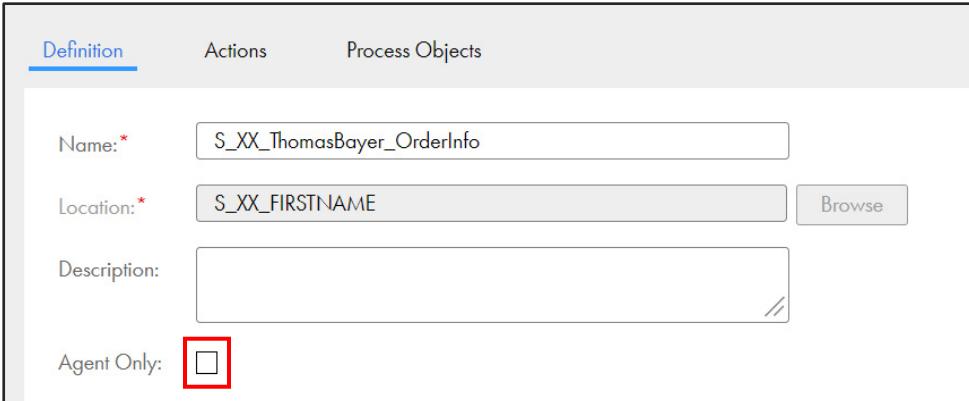
8. In the service connector Name field, enter **S\_XX\_ThomasBayer\_OrderInfo**.  
**Note:** Here, XX refers to your initials.
9. In the Location field, select your project folder **S\_XX\_FirstName**.



The screenshot shows the 'Definition' tab of a service connector configuration page. The 'Name:' field contains 'S\_XX\_ThomasBayer\_OrderInfo' and the 'Location:' field contains 'S\_XX\_FIRSTNAME'. Both of these fields are highlighted with a red rectangular border. The 'Actions' and 'Process Objects' tabs are visible at the top. Below the fields are sections for 'Description:' and 'Agent Only:' with a checkbox.

**Note:** Here, XX is the unique student number.

10. Verify that the **Agent Only** checkbox is cleared.



The screenshot shows the same 'Definition' tab as the previous one, but the 'Agent Only' checkbox is now highlighted with a red rectangular border. All other fields and sections remain the same as in the first screenshot.

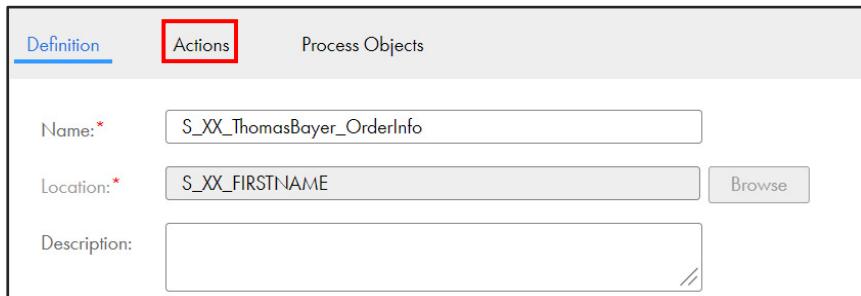
**Note:** If you select Agent Only, then you can run it only on the secure agent machine and not on the cloud server.

### Create Actions:

In this section, you will create three actions – OrderDetails, CustomerDetails, and ItemDetails. These actions will be used in the later labs to extract the corresponding information from the web service.

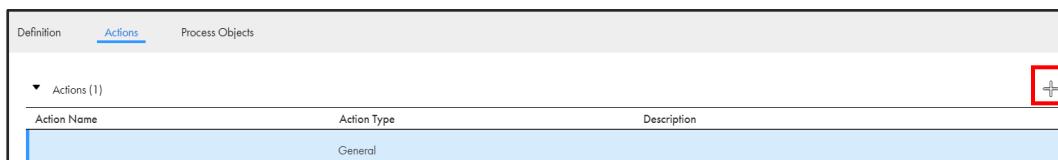
#### Create the OrderDetails Action:

11. Click the **Actions** tab.



The screenshot shows the 'Actions' tab selected in the top navigation bar. Below it, there are three input fields: 'Name:' with value 'S\_XX\_ThomasBayer\_OrderInfo', 'Location:' with value 'S\_XX\_FIRSTNAME' and a 'Browse' button, and 'Description:' with a large empty text area.

12. To add an Action, click .



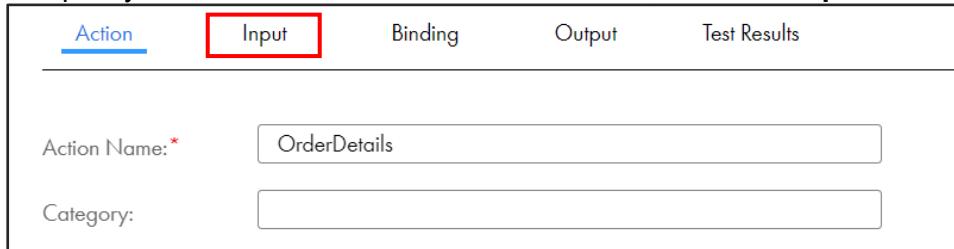
The screenshot shows a table titled 'Actions (1)'. It has columns for 'Action Name' (containing 'General'), 'Action Type' (empty), and 'Description' (empty). In the top right corner of the table, there is a blue plus icon.

13. In the Action Name field, enter **OrderDetails**.



The screenshot shows the 'Action' tab selected. The 'Action Name:' field is highlighted with a red box and contains the value 'OrderDetails'. The 'Category:' field is also present below it.

14. To specify the data that will be sent to the service, select the **Input** tab.



The screenshot shows the 'Input' tab selected. The 'Action Name:' field is highlighted with a red box and contains the value 'OrderDetails'. The 'Category:' field is also present below it.

15. Click  to add an input field.

16. Create the input field as shown in the table below:

| Name     | Label    | Type | Required | Parameter | Test With |
|----------|----------|------|----------|-----------|-----------|
| order_id | order_id | Text | Yes      | No        | 1         |

| Action   | <u>Input</u> | Binding   | Output                              | Test Results             |           |           |
|----------|--------------|---|-------------------------------------|--------------------------|-----------|-----------|
| Name*    | Label        | Type  | Required                            | Description              | Parameter | Test with |
| order_id | order_id     | Text<br><input type="checkbox"/> This is a list | <input checked="" type="checkbox"/> | <input type="checkbox"/> | 1         |           |

**Note:** '1' is just a sample number that will be used to test the service connector.

17. To specify the mode in which the data will be sent, select the **Binding** tab.

| Action           | <u>Input</u> | <u>Binding</u> | Output   | Test Results |  |
|------------------|--------------|----------------|----------|--------------|--|
| Input Fields (1) |              |                |          |              |  |
| Name*            | Label        | Type           | Required |              |  |
|                  |              |                |          |              |  |

18. In the URL field, enter the following URL:

[https://api.predic8.de/shop/v2/orders/{\\$order\\_id}](https://api.predic8.de/shop/v2/orders/{$order_id})

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **03\_LabGuide\_CAI\_Create a Service Connector\_5-1**. Copy the command mentioned under **Step D** and paste it in the URL field.

| Action | Input   | <u>Binding</u> | Output | Test Results  |   |
|--------|---|----------------|--------|---|---|
| URL:*  | <input type="text" value="https://api.predic8.de/shop/v2/orders/{ \$order_id }"/> |                |        |  |  |
| Verb:  | GET   |                |        |  |   |

19. To specify the output fields, select the **Output** tab.

| Action | Input   | <u>Binding</u> | <u>Output</u> | Test Results  |   |
|--------|---|----------------|---------------|---|---|
| URL:*  | <input type="text" value="https://api.predic8.de/shop/v2/orders/{ \$order_id }"/> |                |               |  |  |

20. To create the output fields, click .

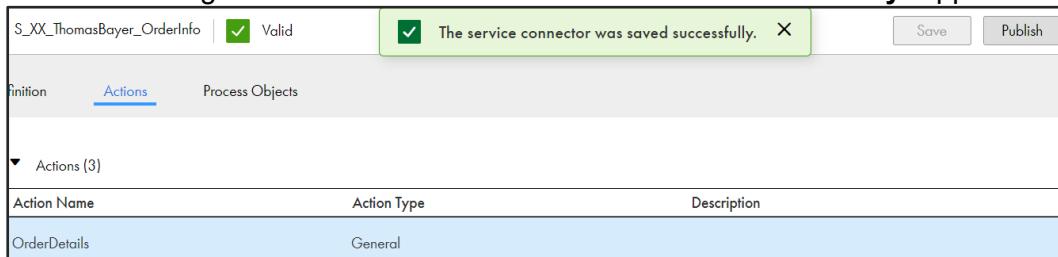
21. Create the Output Fields as shown in the table below:

| Name          | Type | Get From                 |
|---------------|------|--------------------------|
| total         | Text | Property-> total         |
| customer_link | Text | Property-> customer_link |
| Items_link    | Text | Property-> items_link    |

| Action        | Input   | Binding | Output | Test Results              |
|---------------|---|---------|--------|---------------------------|
| total         | Text<br><input type="checkbox"/> This is a list |         |        | Property<br>total         |
| customer_link | Text<br><input type="checkbox"/> This is a list |         |        | Property<br>customer_link |
| Items_link    | Text<br><input type="checkbox"/> This is a list |         |        | Property<br>items_link    |

22. To save the connector, click **Save**.

**Note:** A message “**The service connector was saved successfully**” appears.



S\_XX\_ThomasBayer\_OrderInfo |  Valid |  The service connector was saved successfully. X | Save | Publish

Initiation Actions Process Objects

▼ Actions (3)

| Action Name  | Action Type | Description |
|--------------|-------------|-------------|
| OrderDetails | General     |             |

23. To test the service connector, click the **Test Results** tab.

24. From the Test Server drop-down, retain **Test on Cloud Server**.

25. Click **Test**.

| Action        | Input   | Binding | Output | Test Results                        |
|---------------|---|---------|--------|-------------------------------------|
| Test Server:* | <input type="button" value="Test on Cloud Server"/> |         |        | <input type="button" value="Test"/> |

26. In the Test Results tab, verify that the result of the test is **Successful**.

**Note:** The URL mentions that the order\_id field requested Information for id=1.

| Action         | Input   | Binding | Output | Test Results |
|----------------|---|---------|--------|--------------|
| Test Server: * | Test on Cloud Server                            |         |        |              |
| Result:        | <input checked="" type="checkbox"/> Successful. |         |        |              |
| HTTP Status:   | 200   |         |        |              |
| URL:           | https://api.predic8.de/shop/v2/orders/1         |         |        |              |

27. From the Output Fields section, observe the output.

| Action  | Input                   | Binding | Output | Test Results |       |               |                      |       |                         |  |
|---|-------------------------|---------|--------|--------------|-------|---------------|----------------------|-------|-------------------------|--|
| <hr/>   |                         |         |        |              |       |               |                      |       |                         |  |
| Output  |                         |         |        | Value        |       |               |                      |       |                         |  |
| <table border="1"> <tr> <td>total</td> <td>22.11</td> </tr> <tr> <td>customer_link</td> <td>/shop/v2/customers/1</td> </tr> <tr> <td>items</td> <td>/shop/v2/orders/1/items</td> </tr> </table> |                         |         |        | total        | 22.11 | customer_link | /shop/v2/customers/1 | items | /shop/v2/orders/1/items |  |
| total   | 22.11                   |         |        |              |       |               |                      |       |                         |  |
| customer_link   | /shop/v2/customers/1    |         |        |              |       |               |                      |       |                         |  |
| items   | /shop/v2/orders/1/items |         |        |              |       |               |                      |       |                         |  |
| <span style="color: #0070C0;">▶</span> Response Payload   |                         |         |        |              |       |               |                      |       |                         |  |

**Note:** The output value in the image is subject to change based on the live data available in the API.

28. Expand the **Response Payload** field and observe the output.

| Action | Input   | Binding | Output | Test Results |
|--------|---|---------|--------|--------------|
| <hr/>  |   |         |        |              |
| 1 v    | <root>  |         |        |              |
| 2      | <total>22.11</total>                                |         |        |              |
| 3      | <items_link>/shop/v2/orders/1/items</items_link>    |         |        |              |
| 4      | <id>1</id>  |         |        |              |
| 5      | <state>CREATED</state>                              |         |        |              |
| 6      | <customer_link>/shop/v2/customers/1</customer_link> |         |        |              |
| 7 v    | <actions>   |         |        |              |
| 8 v    | <cancel>  |         |        |              |

**Note:** Response payload displays the response received from the service. Remember, you have entered the Test With value as “1”. The service has responded with the total, items\_link, and customer\_link of order with id “1”.

### Create the CustomerDetails Action:

29. Select the **Actions** tab.
30. To add an action, click .
31. In the Action tab of the service connector, enter Action Name as **CustomerDetails**.
32. Retain the remaining settings to default.

| Action  | Input   | Binding | Output | Test Results |
|---|---|---------|--------|--------------|
| Action Name: * <input type="text" value="CustomerDetails"/> |   |         |        |              |
| Category:   |   |         |        |              |
| Fail on HTTP error: <input checked="" type="checkbox"/>     |   |         |        |              |
| Max Redirects:  | 100   |         |        |              |
| Response Encoding:  | <input type="text" value="Default from response header"/> |         |        |              |

33. To specify the data that will be sent to the service, select the **Input** tab.
34. To add an input field, click .

| Action  | Input | Binding | Output   | Test Results |
|---|-------|---------|----------|--------------|
| Input Fields (0)  |       |         |          |              |
| Name*   | Label | Type    | Required | Description  |
|  Nothing to display. |       |         |          |              |

35. Create the input Fields as shown in the table below:

| Name        | Label       | Type | Required | Parameter | Test With            |
|-------------|-------------|------|----------|-----------|----------------------|
| customer_id | customer_id | Text | Yes      | Yes       | /shop/v2/customers/1 |

| Action  | Input       | Binding | Output                              | Test Results                            |
|---|-------------|---------|-------------------------------------|---|
| Input Fields (1)                                  |             |         |                                     |   |
| Name*   | Label       | Type    | Required                            | Description                             |
| customer_id                                       | customer_id | Text    | <input checked="" type="checkbox"/> | <input type="checkbox"/> This is a list |
| <input type="text" value="/shop/v2/customers/1"/> |             |         |                                     |   |

36. To specify the mode in which the data will be sent, select the **Binding** tab.

37. In the URL field, enter the following URL:

[https://api.predic8.de/{\\$customer\\_id}](https://api.predic8.de/{$customer_id})

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **03\_LabGuide\_CAI\_Create a Service Connector\_5-1**. Copy the command mentioned under **Step E** and paste it in the URL field.

| Action                       | Input   | Binding | Output | Test Results                      |
|------------------------------|---|---------|--------|-----------------------------------|
| URL: *                       | <input type="text" value="https://api.predic8.de/{\$customer_id}"/> |         |        | <input type="button" value="fx"/> |
| Verb:                        | <input type="text" value="GET"/>                                    |         |        | <input type="button" value="▼"/>  |
| Multi Using:                 | <input type="text" value="Semicolon Separated"/>                    |         |        | <input type="button" value="▼"/>  |
| Authentication Type:         | <input type="text" value="Custom"/>                                 |         |        | <input type="button" value="▼"/>  |
| HTTPS/Mutual Authentication: | <input type="checkbox"/>  |         |        |                                   |

38. To define the output format of the response, select the **Output** tab.

39. To create an output field, click .

| Action  | Input | Binding     | Output   | Test Results |
|---|-------|-------------|----------|--------------|
| Output Fields (0)   |       |             |          |              |
| <input type="button" value="+"/>  |       |             |          |              |
| Name *  | Type  | Description | Get From |              |
|  Nothing to display. |       |             |          |              |

40. Create the Output Fields, as shown in the table below:

| Name      | Type | Get From             |
|-----------|------|----------------------|
| firstname | Text | Property-> firstname |
| lastname  | Text | Property-> lastname  |

| Action            | Input                                   | Binding     | Output                | Test Results |
|-------------------|---|-------------|-----------------------|--------------|
| Output Fields (2) |   |             |                       |              |
| Name *            | Type                                    | Description | Get From              |              |
| firstname         | Text                                    |             | Property<br>firstname |              |
|                   | <input type="checkbox"/> This is a list |             |                       |              |
| lastname          | Text                                    |             | Property<br>lastname  |              |
|                   | <input type="checkbox"/> This is a list |             |                       |              |

41. To test the service connector, select the **Test Results** tab.

42. From the Test Server field, retain **Test on Cloud Server**.

43. Click **Test**.

| Action        | Input                | Binding | Output | Test Results |
|---------------|----------------------|---------|--------|--------------|
| Test Server:* | Test on Cloud Server |         |        | <b>Test</b>  |

44. Verify that the result of the test is **Successful**.

| Action        | Input   | Binding | Output | Test Results |
|---------------|---|---------|--------|--------------|
| Test Server:* | Test on Cloud Server  |         |        | <b>Test</b>  |
| Result:       | <input checked="" type="checkbox"/> Successful.                                     |         |        |              |
| HTTP Status:  | 200   |         |        |              |
| URL:          | https://api.predic8.de/shop/v2/customers/1?customer_id=%2Fshop%2Fv2%2Fcustomers%2F1 |         |        |              |

45. Observe the **Output Fields** section. It lists the output fields with the output values defined in the Service Connector action.

| Action               | Input | Binding | Output | Test Results |
|----------------------|-------|---------|--------|--------------|
| <b>Output Fields</b> |       |         |        |              |
| Output               |       |         |        | Value        |
| firstname            |       |         |        | Ted          |
| lastname             |       |         |        | Walters      |

**Note:** The output value in the image is subject to change based on the live data available in the API.

46. Expand the **Response Payload** field and observe the output.

| Action  | Input | Binding | Output | Test Results        |
|---|-------|---------|--------|---------------------|
|   |       |         |        | <u>Test Results</u> |
| lastname<br><div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <span style="border: 1px solid #0070C0; padding: 2px;">▼</span> Response Payload       </div> <pre> 1 &lt;root&gt; 2   &lt;firstname&gt;Ted&lt;/firstname&gt; 3   &lt;lastname&gt;Walters&lt;/lastname&gt; 4 &lt;/root&gt;       </pre> |       |         |        |                     |

#### Create the ItemDetails Action:

47. Select the **Actions** tab.

48. To add the action, click .

49. In the Action tab of the service connector, enter Action Name as **ItemDetails**.

50. In the Action Name field, enter **ItemDetails**.

| Action              | Input                               | Binding | Output | Test Results |
|---------------------|-------------------------------------|---------|--------|--------------|
| Action Name: *      | ItemDetails                         |         |        |              |
| Category:           |                                     |         |        |              |
| Fail on HTTP error: | <input checked="" type="checkbox"/> |         |        |              |
| Max Redirects:      | 100                                 |         |        |              |

51. To specify the data that will be sent to the service, select the **Input** tab.

52. To add the input field, click .

| Action   | Input | Binding | Output   | Test Results |           |           |      |          |             |           |           |   |  |  |  |  |  |  |                     |  |  |  |  |  |  |
|--|-------|---------|----------|--------------|-----------|-----------|------|----------|-------------|-----------|-----------|---|--|--|--|--|--|--|---------------------|--|--|--|--|--|--|
| Input Fields (0) <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Name *</th> <th>Label</th> <th>Type</th> <th>Required</th> <th>Description</th> <th>Parameter</th> <th>Test with</th> </tr> </thead> <tbody> <tr> <td colspan="7" style="text-align: right; padding-right: 10px;"></td> </tr> <tr> <td colspan="7" style="text-align: center; padding-top: 10px;">Nothing to display.</td> </tr> </tbody> </table> |       |         |          |              | Name *    | Label     | Type | Required | Description | Parameter | Test with |  |  |  |  |  |  |  | Nothing to display. |  |  |  |  |  |  |
| Name *   | Label | Type    | Required | Description  | Parameter | Test with |      |          |             |           |           |   |  |  |  |  |  |  |                     |  |  |  |  |  |  |
|   |       |         |          |              |           |           |      |          |             |           |           |   |  |  |  |  |  |  |                     |  |  |  |  |  |  |
| Nothing to display.  |       |         |          |              |           |           |      |          |             |           |           |   |  |  |  |  |  |  |                     |  |  |  |  |  |  |

53. Create the Input Fields as shown in the table below:

| Name     | Label    | Type | Required | Parameter | Test With               |
|----------|----------|------|----------|-----------|-------------------------|
| items_id | items_id | Text | Yes      | No        | /shop/v2/orders/1/items |

| Action           | Input    | Binding | Output   | Test Results                                     |
|------------------|----------|---------|--|--|
| Input Fields (1) |          |         |  |  |
| Name*            | Label    | Type    | Required   | Description                                      |
| items_id         | items_id | Text    | <input checked="" type="checkbox"/> This is a list | <input type="checkbox"/> /shop/v2/orders/1/items |

54. To specify the mode in which the data will be sent, select the **Binding** tab.

55. In the URL field, enter the following URL:

[https://api.predic8.de/{\\$items\\_id}](https://api.predic8.de/{$items_id})

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **03\_LabGuide\_CAI\_Create a Service Connector\_5-1**. Copy the command mentioned under **Step F** and paste it in the URL field.

| Action       | Input  | Binding | Output | Test Results  |
|--------------|--|---------|--------|---|
| URL:*        | <input type="text" value="https://api.predic8.de/{\$items_id}"/> |         |        |  |
| Verb:        | <input type="text" value="GET"/>                                 |         |        |  |
| Multi Using: | <input type="text" value="Semicolon Separated"/>                 |         |        |  |

56. To define the output format of the response, select the **Output** tab.

57. To create the output field, click .

| Action  | Input | Binding     | Output   | Test Results  |
|---|-------|-------------|----------|---|
| Output Fields (0)   |       |             |          |   |
| Name*   | Type  | Description | Get From |  |
|  Nothing to display. |       |             |          |   |

58. Create the Output Field, as shown in the table below:

| Name          | Type | Get From        |
|---------------|------|-----------------|
| items details | Text | Entire Response |

| Action                   | Input   | Binding            | <u>Output</u>   | Test Results |
|--------------------------|---|--------------------|-----------------|--------------|
| <b>Output Fields (1)</b> |   |                    |                 |              |
| <b>Name *</b>            | <b>Type</b>                                     | <b>Description</b> | <b>Get From</b> |              |
| items details            | Text<br><input type="checkbox"/> This is a list |                    | Entire Response |              |

**Note:** You can select the “Entire Response” option from the **Get From** drop-down.

| Action  | Input   | Binding | <u>Output</u>                                   | Test Results   |        |       |         |               |              |               |   |  |   |  |
|---|---|---------|---|--|--------|-------|---------|---------------|--------------|---------------|---|--|---|--|
| ItemDetails General   |   |         |   |  |        |       |         |               |              |               |   |  |   |  |
| <table border="1"> <thead> <tr> <th>Action</th> <th>Input</th> <th>Binding</th> <th><u>Output</u></th> <th>Test Results</th> </tr> </thead> <tbody> <tr> <td>items details</td> <td>Text<br/><input type="checkbox"/> This is a list</td> <td></td> <td>Text<br/><input type="checkbox"/> This is a list</td> <td> <div style="border: 1px solid #ccc; padding: 5px;">           Property<br/>           Expression<br/>           HTTP Response Status Code<br/>           HTTP Response Header<br/> <b>Entire Response</b><br/>           Simplified XML<br/>           Entire Response As Attachment<br/>           Attachments<br/>           Property         </div> </td> </tr> </tbody> </table> |   |         |   |  | Action | Input | Binding | <u>Output</u> | Test Results | items details | Text<br><input type="checkbox"/> This is a list |  | Text<br><input type="checkbox"/> This is a list | <div style="border: 1px solid #ccc; padding: 5px;">           Property<br/>           Expression<br/>           HTTP Response Status Code<br/>           HTTP Response Header<br/> <b>Entire Response</b><br/>           Simplified XML<br/>           Entire Response As Attachment<br/>           Attachments<br/>           Property         </div> |
| Action  | Input   | Binding | <u>Output</u>                                   | Test Results   |        |       |         |               |              |               |   |  |   |  |
| items details   | Text<br><input type="checkbox"/> This is a list |         | Text<br><input type="checkbox"/> This is a list | <div style="border: 1px solid #ccc; padding: 5px;">           Property<br/>           Expression<br/>           HTTP Response Status Code<br/>           HTTP Response Header<br/> <b>Entire Response</b><br/>           Simplified XML<br/>           Entire Response As Attachment<br/>           Attachments<br/>           Property         </div> |        |       |         |               |              |               |   |  |   |  |

59. To test the service connector, select the **Test Results** tab.

60. From the Test Server field, retain **Test on Cloud Server**.

61. Click **Test**.

| Action  | Input | Binding | Output | <u>Test Results</u> |
|---|-------|---------|--------|---------------------|
| Test Server: * <b>Test on Cloud Server</b> <span style="border: 1px solid red; padding: 2px;"> </span> <span style="background-color: green; color: white; border: 1px solid green; padding: 2px; cursor: pointer;">Test</span> |       |         |        |                     |

62. Verify that the result of the test is **Successful**.

| Action   | Input   | Binding | <u>Output</u> | <u>Test Results</u> |
|--|---|---------|---------------|---------------------|
| Test Server: * <b>Test on Cloud Server</b> <span style="border: 1px solid green; background-color: #e0f2e0; padding: 2px;"> </span> <span style="background-color: green; color: white; border: 1px solid green; padding: 2px; cursor: pointer;">Test</span> |   |         |               |                     |
| Result:  | <input checked="" type="checkbox"/> Successful. |         |               |                     |
| HTTP Status:   | 200   |         |               |                     |
| URL:   | https://api.predic8.de/shop/v2/orders/1/items   |         |               |                     |

63. Observe the **Output Fields** section. You can see the entire string of ordered items for order\_id “1”.

**Note:** The output value in the image is subject to change based on the live data available in the API.

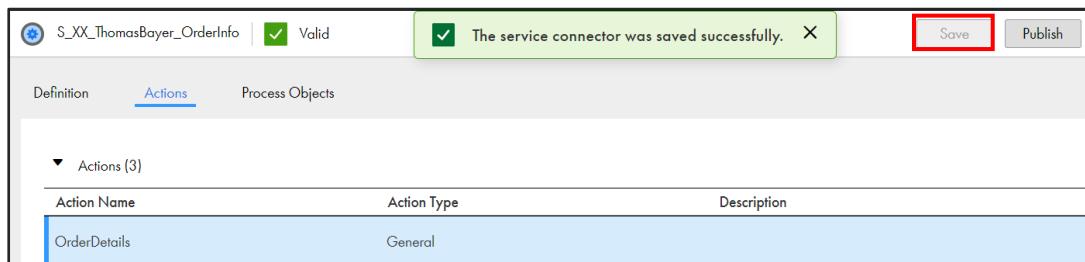
| Action   | Input | Binding | Output | Test Results |
|--|-------|---------|--------|--------------|
| Output Fields  |       |         |        |              |
| Output Value<br><pre>&lt;root xmlns:aelt="http://schemas.informatica.com/socrates/data-services/2014/05/business-connector-model.xsd" xmlns:bconn="http://schemas.informatica.com/socrates/data-services/2014/05/business-connector-model.xsd"&gt;&lt;order_link&gt;/shop/v2/orders/1&lt;/order_link&gt;&lt;items&gt;&lt;item&gt;&lt;product&gt;1&lt;/product&gt;&lt;quantity&gt;21&lt;/quantity&gt;&lt;/item&gt;&lt;item&gt;&lt;product&gt;2&lt;/product&gt;&lt;quantity&gt;11&lt;/quantity&gt;&lt;/item&gt;&lt;/items&gt;&lt;/root&gt;</pre> |       |         |        |              |
| <input type="button" value="Generate Process Objects"/>  |       |         |        |              |

64. Expand the **Response Payload** field and observe the output.

| Action  | Input | Binding | Output | Test Results |
|---|-------|---------|--------|--------------|
| <pre> 1 &lt;root&gt; 2   &lt;order_link&gt;/shop/v2/orders/1&lt;/order_link&gt; 3   &lt;items&gt; 4     &lt;item&gt; 5       &lt;product&gt;1&lt;/product&gt; 6       &lt;quantity&gt;21&lt;/quantity&gt; 7     &lt;/item&gt; 8   &lt;items&gt; 9     &lt;item&gt; 10    &lt;product&gt;2&lt;/product&gt; 11    &lt;quantity&gt;11&lt;/quantity&gt; 12  &lt;/items&gt; </pre> |       |         |        |              |

**Note:** Response payload displays the product numbers along with their quantities.

65. Click **Save** to save the connection.

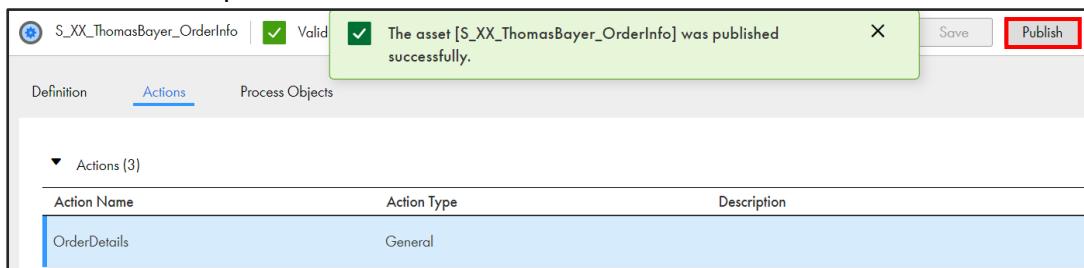


The screenshot shows the 'Actions' tab of a service connector configuration. At the top, there is a green success message: 'The service connector was saved successfully.' Below the message are tabs for 'Definition', 'Actions' (which is selected), and 'Process Objects'. Under the 'Actions' tab, there is a table with one row:

| Action Name  | Action Type | Description |
|--------------|-------------|-------------|
| OrderDetails | General     |             |

A red box highlights the 'Save' button at the top right of the window.

66. Click **Publish** to publish the connection.



**Note:** A message **The asset was published successfully** appears.

---

*This concludes the lab.*

## Module 5: Service Connectors

### Lab 5-2: Create a Service Connector from WSDL

**Overview:**

In this lab, you will create a Service Connector from a WSDL file.

**Objectives:**

- Import a WSDL File
- Modify an existing Action
- Test the Service Connector

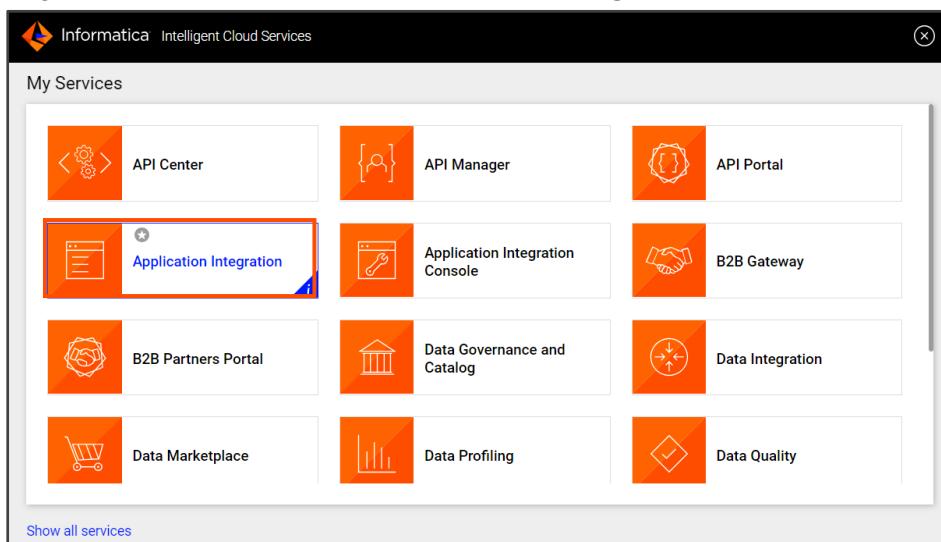
**Duration:**

20 Minutes

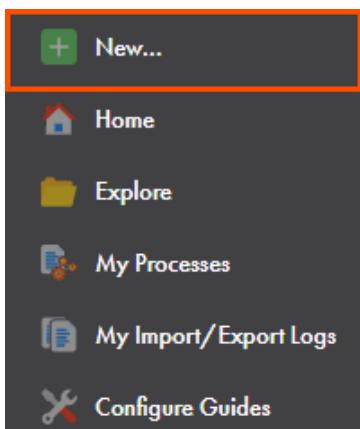
---

**Tasks****Import a WSDL File**

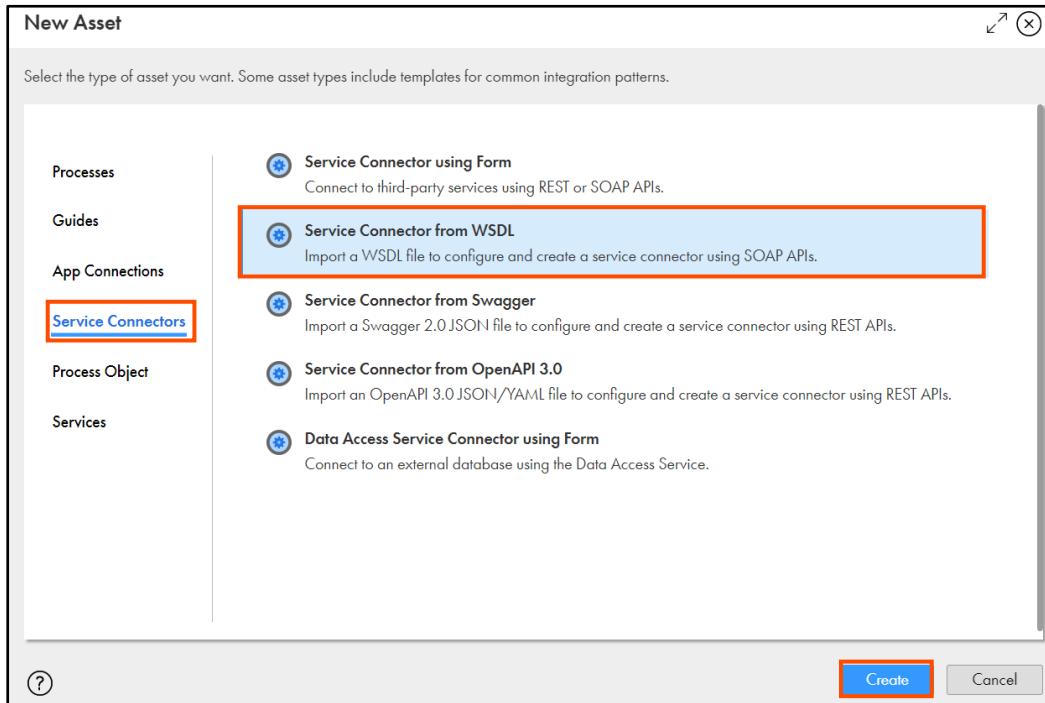
1. Log into IICS and access the **Application Integration** service.



2. To create a new Service Connector, from the navigation pane, click **New**.

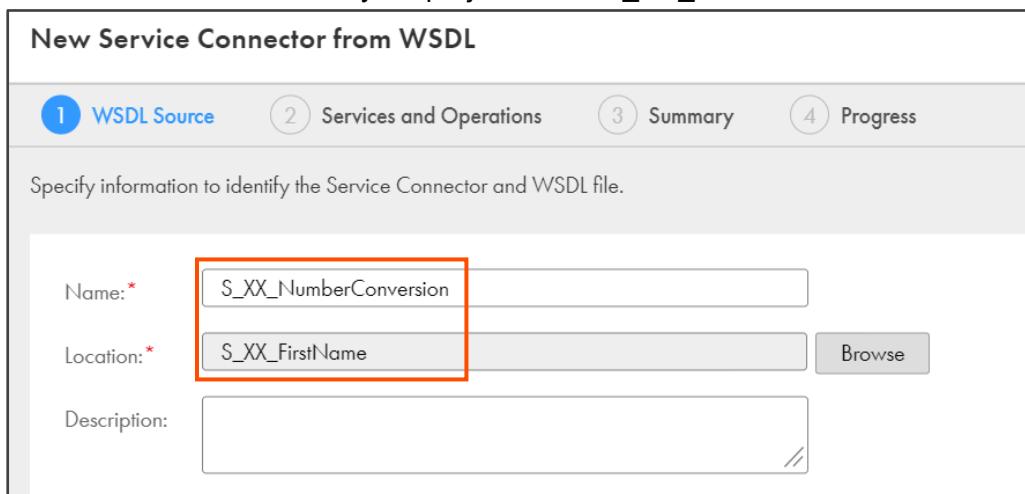


3. From the Service Connectors tab, select **Service Connector from WSDL**.
4. Click **Create**.



A new Service Connector page appears.

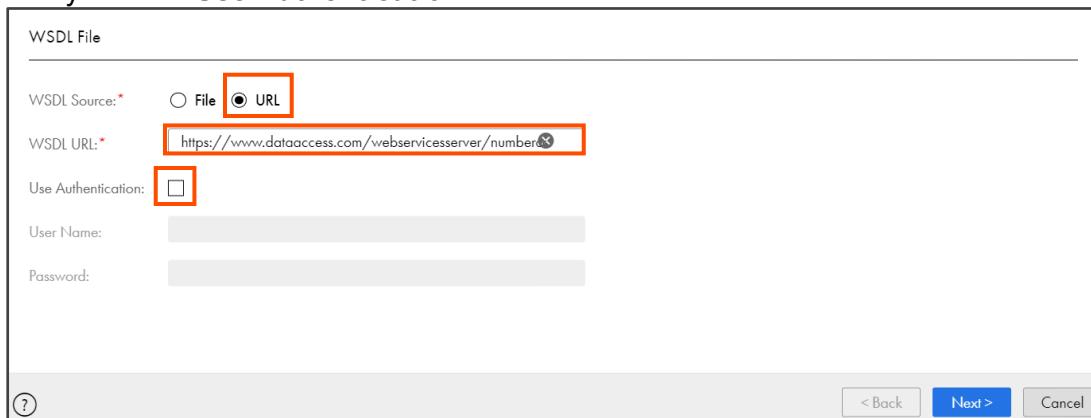
5. In the service connector Name field, enter **S\_XX\_NumberConversion**.
6. In the Location field, select your project folder **S\_XX\_FirstName**.



The screenshot shows the 'New Service Connector from WSDL' configuration page. It has four tabs at the top: 1 WSDL Source (selected), 2 Services and Operations, 3 Summary, and 4 Progress. The main area is titled 'Specify information to identify the Service Connector and WSDL file.' It has three input fields: 'Name:' with value 'S\_XX\_NumberConversion', 'Location:' with value 'S\_XX\_FirstName' (which is highlighted with a red box), and 'Description:' with an empty text area. Below the input fields is a 'Browse' button.

7. Retain WSDL Source as **URL**.
8. In the WSDL URL field, enter the following URL:  
<https://www.dataaccess.com/webservicesserver/numberconversion.wso?WSDL>

9. Verify that the **Use Authentication** checkbox is cleared.



WSDL File

WSDL Source:  File  URL

WSDL URL:

Use Authentication:

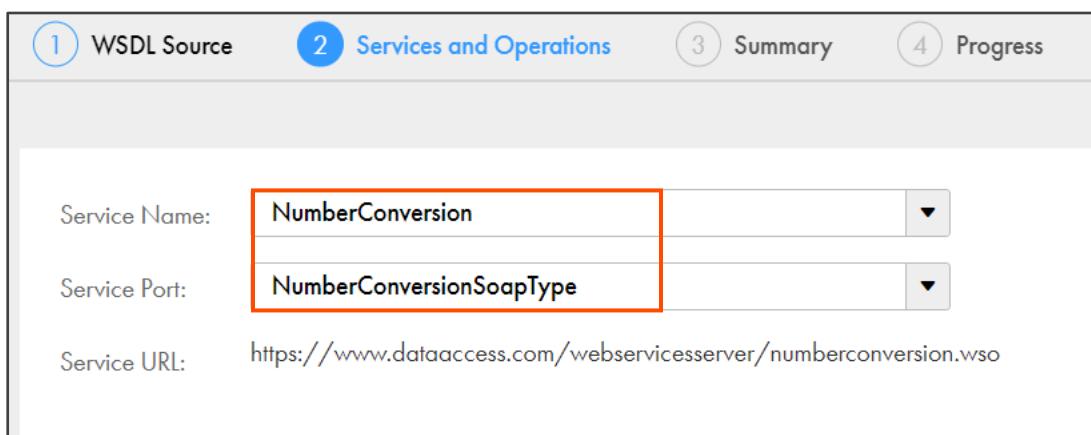
User Name:

Password:

? < Back Next > Cancel

10. Click **Next**.

11. Retain the values in Service Name and Service Port fields.



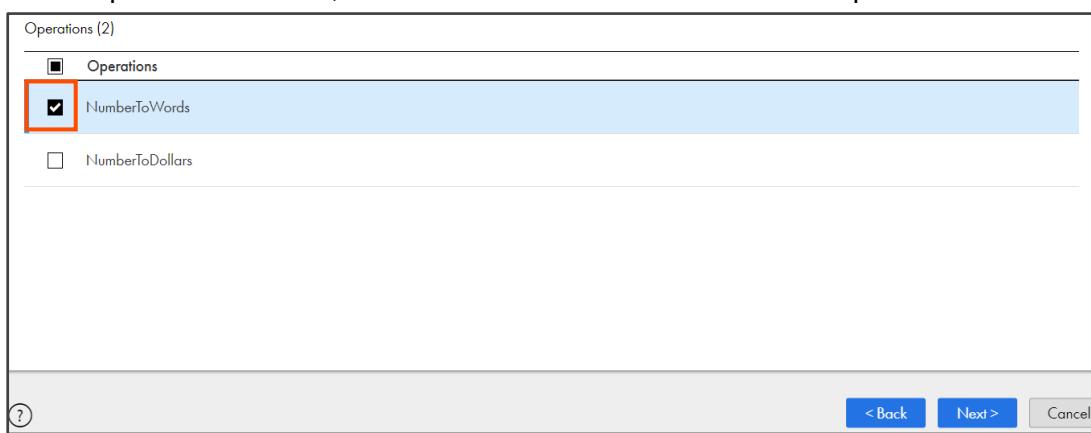
1 WSDL Source 2 Services and Operations 3 Summary 4 Progress

Service Name: NumberConversion

Service Port: NumberConversionSoapType

Service URL: https://www.dataaccess.com/webservicesserver/numberconversion.wsdl

12. In the Operations section, make sure that the **NumberToWords** option is checked.



Operations (2)

Operations

NumberToWords

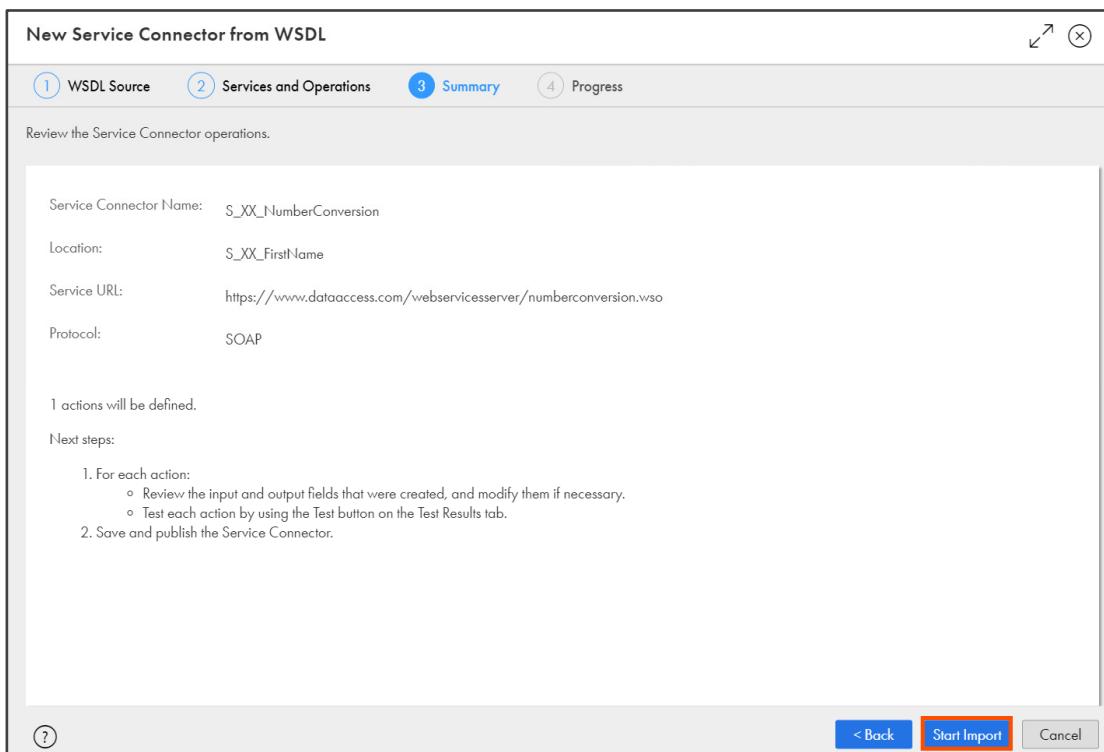
NumberToDollars

? < Back Next > Cancel

**Note:** Ensure that the NumberToDollars option is unchecked.

13. Click **Next**.

**14. Click Start Import.**



New Service Connector from WSDL

① WSDL Source   ② Services and Operations   ③ **Summary**   ④ Progress

Review the Service Connector operations.

Service Connector Name: S\_XX\_NumberConversion

Location: S\_XX\_FirstName

Service URL: <https://www.dataaccess.com/webservicesserver/numberconversion.wso>

Protocol: SOAP

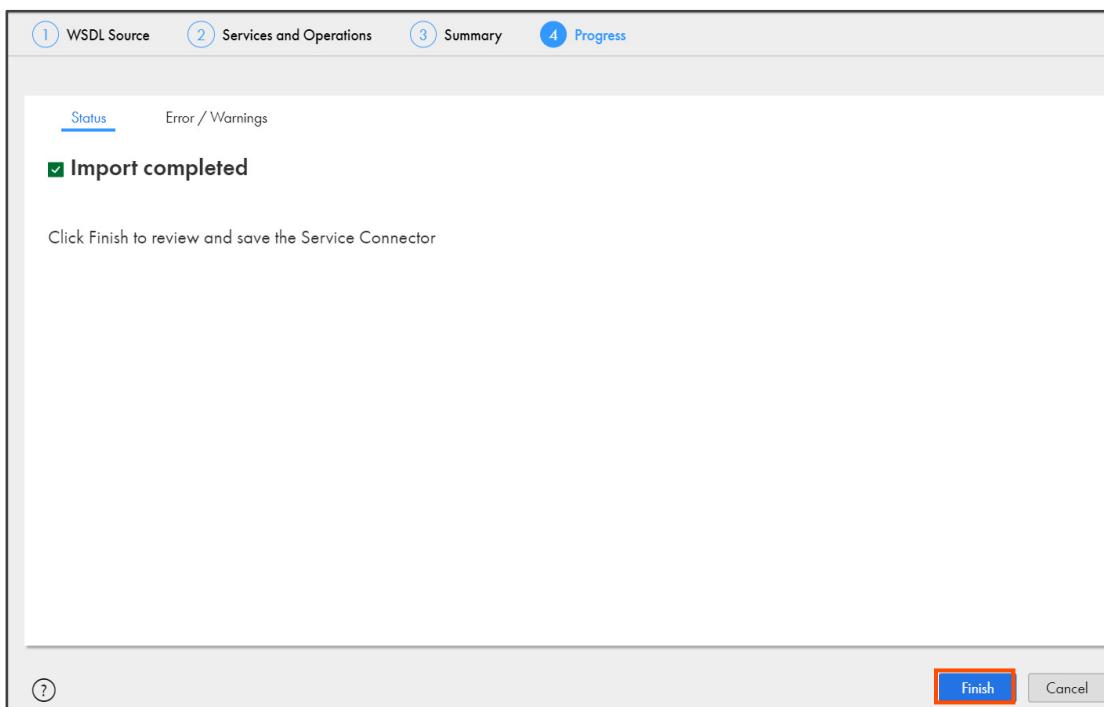
1 actions will be defined.

Next steps:

1. For each action:
  - Review the input and output fields that were created, and modify them if necessary.
  - Test each action by using the Test button on the Test Results tab.
2. Save and publish the Service Connector.

< Back   **Start Import**   Cancel

**15. Click Finish.**



① WSDL Source   ② Services and Operations   ③ Summary   ④ **Progress**

Status   Error / Warnings

**✓ Import completed**

Click Finish to review and save the Service Connector

< Back   **Finish**   Cancel

## Modify the NumberToWords Action

16. Click the **Actions** tab.
17. Select the **NumberToWords** Action.

| Definition  | Actions     | Process Objects     |             |             |             |            |          |  |                      |  |                     |
|---|-------------|---------------------|-------------|-------------|-------------|------------|----------|--|----------------------|--|---------------------|
| <b>▼ Actions (2)</b> <table border="1"> <thead> <tr> <th>Action Name</th> <th>Action Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>BaseAction</td> <td>Abstract</td> <td></td> </tr> <tr> <td><b>NumberToWords</b></td> <td></td> <td>Inherit: BaseAction</td> </tr> </tbody> </table> |             |                     | Action Name | Action Type | Description | BaseAction | Abstract |  | <b>NumberToWords</b> |  | Inherit: BaseAction |
| Action Name   | Action Type | Description         |             |             |             |            |          |  |                      |  |                     |
| BaseAction  | Abstract    |                     |             |             |             |            |          |  |                      |  |                     |
| <b>NumberToWords</b>  |             | Inherit: BaseAction |             |             |             |            |          |  |                      |  |                     |

18. Select the **Input** tab.

| Action   | Input | Binding  | Output                   | Test Results                        |           |           |      |          |             |           |           |               |  |  |                          |                                     |  |  |
|--|-------|--|--------------------------|-------------------------------------|-----------|-----------|------|----------|-------------|-----------|-----------|---------------|--|--|--------------------------|-------------------------------------|--|--|
| <table border="1"> <thead> <tr> <th>Name*</th> <th>Label</th> <th>Type</th> <th>Required</th> <th>Description</th> <th>Parameter</th> <th>Test with</th> </tr> </thead> <tbody> <tr> <td>NumberToWords</td> <td></td> <td>Reference<br/>Process Object<br/><input type="text" value="NumberToWords_AT"/></td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td></td> <td></td> </tr> </tbody> </table> |       |  |                          |                                     | Name*     | Label     | Type | Required | Description | Parameter | Test with | NumberToWords |  | Reference<br>Process Object<br><input type="text" value="NumberToWords_AT"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |  |  |
| Name*  | Label | Type   | Required                 | Description                         | Parameter | Test with |      |          |             |           |           |               |  |  |                          |                                     |  |  |
| NumberToWords  |       | Reference<br>Process Object<br><input type="text" value="NumberToWords_AT"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |           |           |      |          |             |           |           |               |  |  |                          |                                     |  |  |

19. Click the **NumberToWords** row and in the Test With field, enter **123**.

| Action   | Input | Binding  | Output                   | Test Results                        |                                  |           |      |          |             |           |           |               |  |  |                          |                                     |                                  |  |
|--|-------|--|--------------------------|-------------------------------------|----------------------------------|-----------|------|----------|-------------|-----------|-----------|---------------|--|--|--------------------------|-------------------------------------|----------------------------------|--|
| <table border="1"> <thead> <tr> <th>Name*</th> <th>Label</th> <th>Type</th> <th>Required</th> <th>Description</th> <th>Parameter</th> <th>Test with</th> </tr> </thead> <tbody> <tr> <td>NumberToWords</td> <td></td> <td>Reference<br/>Process Object<br/><input type="text" value="NumberToWords_AT"/></td> <td><input type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td><input type="text" value="123"/></td> <td></td> </tr> </tbody> </table> |       |  |                          |                                     | Name*                            | Label     | Type | Required | Description | Parameter | Test with | NumberToWords |  | Reference<br>Process Object<br><input type="text" value="NumberToWords_AT"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="text" value="123"/> |  |
| Name*  | Label | Type   | Required                 | Description                         | Parameter                        | Test with |      |          |             |           |           |               |  |  |                          |                                     |                                  |  |
| NumberToWords  |       | Reference<br>Process Object<br><input type="text" value="NumberToWords_AT"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="text" value="123"/> |           |      |          |             |           |           |               |  |  |                          |                                     |                                  |  |

**Note:** You can replace this value with any number which you want to convert to words.

20. Select **Binding**.

| Action  | Input  | Binding                              | Output | Test Results |       |  |                                      |       |                                   |  |  |              |  |  |  |                      |                                     |  |  |
|---|--|--------------------------------------|--------|--------------|-------|--|--------------------------------------|-------|-----------------------------------|--|--|--------------|--|--|--|----------------------|-------------------------------------|--|--|
| <table border="1"> <tr> <td>URL:*</td> <td><input type="text" value="https://www.dataaccess.com/webservicesserver/"/></td> <td><input type="text" value="inherit"/></td> </tr> <tr> <td>Verb:</td> <td colspan="3"><input type="text" value="POST"/></td> </tr> <tr> <td>Multi Using:</td> <td colspan="3"><input type="text" value="Semicolon Separated"/></td> </tr> <tr> <td>Authentication Type:</td> <td colspan="3"><input type="text" value="Custom"/></td> </tr> </table> |  |                                      |        |              | URL:* | <input type="text" value="https://www.dataaccess.com/webservicesserver/"/> | <input type="text" value="inherit"/> | Verb: | <input type="text" value="POST"/> |  |  | Multi Using: | <input type="text" value="Semicolon Separated"/> |  |  | Authentication Type: | <input type="text" value="Custom"/> |  |  |
| URL:*   | <input type="text" value="https://www.dataaccess.com/webservicesserver/"/> | <input type="text" value="inherit"/> |        |              |       |  |                                      |       |                                   |  |  |              |  |  |  |                      |                                     |  |  |
| Verb:   | <input type="text" value="POST"/>  |                                      |        |              |       |  |                                      |       |                                   |  |  |              |  |  |  |                      |                                     |  |  |
| Multi Using:  | <input type="text" value="Semicolon Separated"/>                           |                                      |        |              |       |  |                                      |       |                                   |  |  |              |  |  |  |                      |                                     |  |  |
| Authentication Type:  | <input type="text" value="Custom"/>  |                                      |        |              |       |  |                                      |       |                                   |  |  |              |  |  |  |                      |                                     |  |  |

21. Scroll down and from the Binding Type drop-down, select **overwrite**.

| Action        | Input  | <u>Binding</u> | Output  | Test Results |
|---------------|--|----------------|---------|--------------|
|               |  |                |         |              |
| Binding Type: | Custom   |                | inherit |              |
| Body:         | <pre>&lt;soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope"&gt;   &lt;soap:Header/&gt;</pre> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <b>inherit</b><br/> <span style="border: 2px solid red; padding: 2px;">overwrite</span> </div> |                |         |              |

22. Scroll down and make sure that the **overwrite** option is selected in the drop-down.

Body:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Header/>
<soap:Body>
```

overwrite ▾

23. Click  $f(x)$ .

Binding      Output      Test Results

inherit

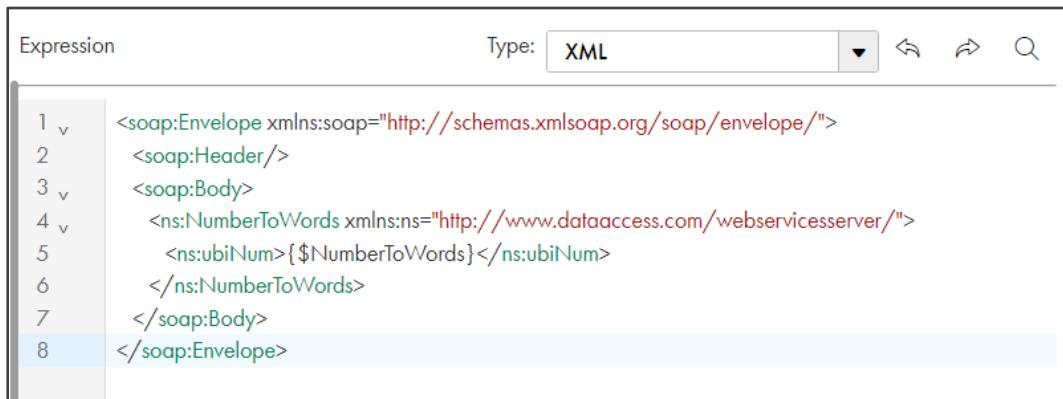
ns="http://www.dataaccess.com/webservicesserver/">  
Words/ubit/lum/text()|</ns:ubit/lum>

24. In the Expression section, replace the text with the following:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body>
    <ns:NumberToWords
      xmlns:ns="http://www.dataaccess.com/webservicesserver/">
      <ns:ubiNum>{$NumberToWords}</ns:ubiNum>
    </ns:NumberToWords>
  </soap:Body>
</soap:Envelope>
```

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **04\_LabGuide\_CAI\_Create a Service Connector from WSDL\_5-2**. Copy the command mentioned under **Step A** and paste it in the body field.



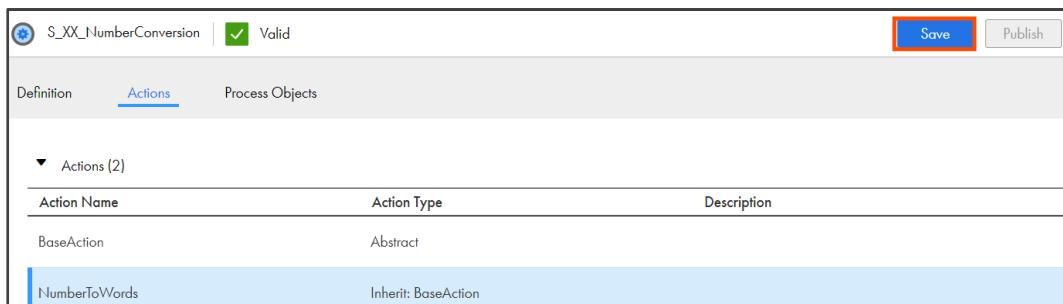
```

1 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
2   <soap:Header/>
3   <soap:Body>
4     <ns:NumberToWords xmlns:ns="http://www.dataaccess.com/webservicesserver/">
5       <ns:ubiNum>{$NumberToWords}</ns:ubiNum>
6     </ns:NumberToWords>
7   </soap:Body>
8 </soap:Envelope>

```

25. Click **OK**.

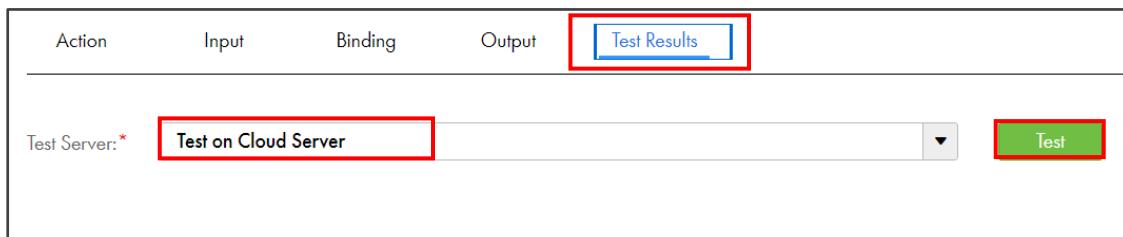
26. Click **Save**.



## Test Action

27. Select **Test Results**.

28. From the Test Server drop-down, retain **Test on Cloud Server** and click **Test**.



| Action         | Input                | Binding | Output | Test Results |
|----------------|----------------------|---------|--------|--------------|
| Test Server: * | Test on Cloud Server |         |        | Test         |

29. In the Test Results tab, verify that the result of the test is **Successful**.

**Note:** Note that HTTPS Status is 200.

| Action               | Input   | Binding | Output | <u>Test Results</u>                 |
|----------------------|---|---------|--------|-------------------------------------|
| Test Server:*        | Test on Cloud Server  |         |        | <input type="button" value="Test"/> |
| Result:              | <input checked="" type="checkbox"/> Successful.                   |         |        |                                     |
| HTTP Status:         | 200   |         |        |                                     |
| URL:                 | https://www.dataaccess.com/webservicesserver/numberconversion.wso |         |        |                                     |
| <b>Output Fields</b> |   |         |        |                                     |

30. From the Output Fields section, scroll down and expand **Response Payload**.

31. Observe the output.

| Action  | Input | Binding | Output | <u>Test Results</u> |
|---|-------|---------|--------|---------------------|
| <div style="border: 1px solid #ccc; padding: 5px;"> <span style="border: 1px solid #ccc; padding: 2px;">▼</span> Response Payload         </div> <pre> 1 &lt;soap:Envelope xmlns:rest="http://schemas.activebpel.org/REST/2007/12/01/aeREST.xsd" 2   xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" 3   &lt;soap:Body&gt; 4     &lt;m:NumberToWordsResponse xmlns:m="http://www.dataaccess.com/webservicesserver/"&gt; 5       &lt;m:NumberToWordsResult&gt;one hundred and twenty three&lt;/m:NumberToWordsResult&gt; 6     &lt;/m:NumberToWordsResponse&gt; 7   &lt;/soap:Body&gt; </pre> |       |         |        |                     |

**Note:** Response payload displays the response received from the service. Remember, you have entered the Test With value as "123". The service has converted the number "123" to words "one hundred and twenty three".

32. Click **Publish**.

S\_XX\_NumberConversion  The asset [S\_XX\_NumberConversion] was published successfully. x

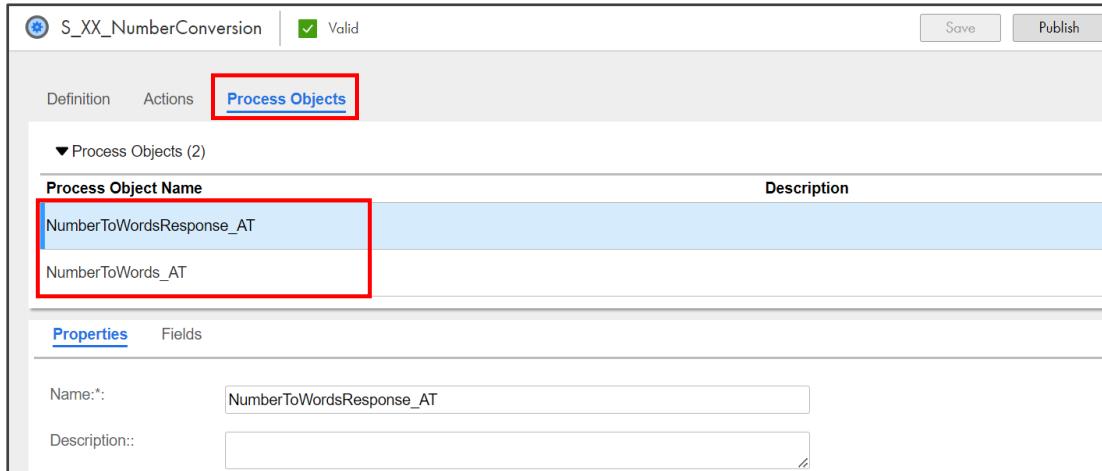
| Definition   | Actions              | Process Objects |             |             |             |            |          |  |               |                      |  |
|--|----------------------|-----------------|-------------|-------------|-------------|------------|----------|--|---------------|----------------------|--|
| <div style="border: 1px solid #ccc; padding: 5px;"> <span style="border: 1px solid #ccc; padding: 2px;">▼ Actions (2)</span> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Action Name</th> <th>Action Type</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>BaseAction</td> <td>abstract</td> <td></td> </tr> <tr> <td>NumberToWords</td> <td>Inherit : BaseAction</td> <td></td> </tr> </tbody> </table> </div> |                      |                 | Action Name | Action Type | Description | BaseAction | abstract |  | NumberToWords | Inherit : BaseAction |  |
| Action Name  | Action Type          | Description     |             |             |             |            |          |  |               |                      |  |
| BaseAction   | abstract             |                 |             |             |             |            |          |  |               |                      |  |
| NumberToWords  | Inherit : BaseAction |                 |             |             |             |            |          |  |               |                      |  |

A message "**The asset [asset\_name] was published successfully**" appears.

## Service Connector Process Objects

A service connector process object groups data. The WSDL URL you imported to generate a service connector generates two process objects.

33. To view service connector process objects, click the **Process Objects** tab.



The screenshot shows the 'Process Objects' tab selected in a service connector configuration interface. The tab displays two process objects:

| Process Object Name      | Description |
|--------------------------|-------------|
| NumberToWordsResponse_AT |             |
| NumberToWords_AT         |             |

Below the table, there are 'Properties' and 'Fields' sections. The 'Properties' section shows 'Name:' set to 'NumberToWordsResponse\_AT' and 'Description:' as empty. The 'Fields' section is currently empty.

**Note:** You can observe two Process Objects: **NumberToWordsResponse\_AT** and **NumberToWords\_AT**.

Process Object passes the value to the Expression.

34. Close the Service Connector.

---

*This concludes the lab.*

# Module 5: Service Connectors

## Lab 5-3: Create and Test a Swagger File

### Overview:

Informatica Intelligent Cloud Services supports Swagger specification version 2.0. When you generate a Swagger file in Informatica Intelligent Cloud Services, you send an API call to the service using a sample request. If you do not have permissions to send an API call to the service, you can generate a Swagger file using a sample request and a sample response without submitting an API call.

You cannot modify a Swagger file once it is created. If you want to make changes in a Swagger file, create a new Swagger file.

**Note:** The Swagger file generation functionality is available for the convenience of the REST V2 customers. Informatica does not warranty the compatibility of the Swagger file for all customer scenarios.

### Objectives:

- Sign up to Access OpenWeatherAPI
- Create Swagger files
- Test the Swagger file

### Duration:

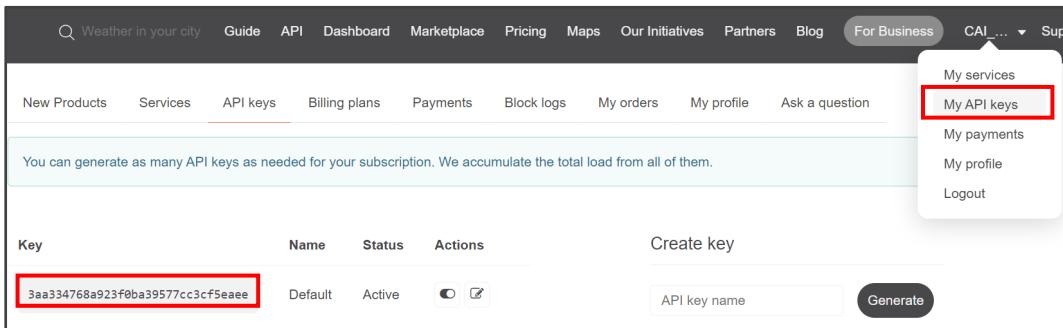
15 Minutes

---

### Tasks

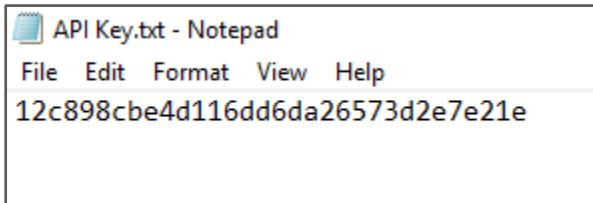
#### Sign up to Access OpenWeatherAPI

1. Open a web browser and enter the following URL in a new tab:  
[https://home.openweathermap.org/users/sign\\_up](https://home.openweathermap.org/users/sign_up)
- Note:** You can bookmark this link for future use.
2. Create a new account in OpenWeather using the link provided.  
**Note:** After you create an account, you can access your unique API key.
3. To access your unique API Key, click on your user ID, and select **MY API Keys**.
4. Copy the API key from the Key field.



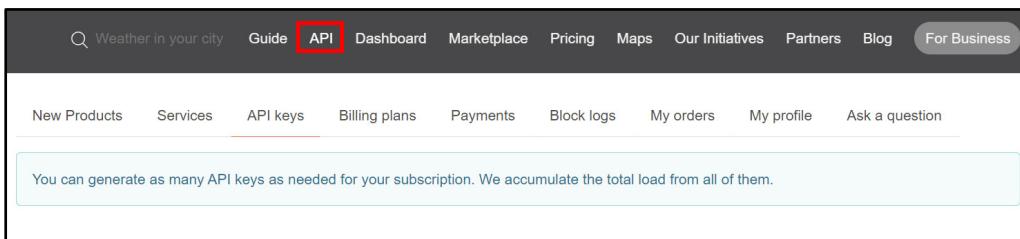
The screenshot shows the OpenWeather API keys management interface. At the top, there's a navigation bar with links like Weather in your city, Guide, API, Dashboard, Marketplace, Pricing, Maps, Our Initiatives, Partners, Blog, For Business, CAI ..., and Sup. Below the navigation bar, there's a main menu with options: New Products, Services, API keys (which is underlined), Billing plans, Payments, Block logs, My orders, My profile, and Ask a question. A tooltip appears over the 'CAI ...' button, listing 'My services', 'My API keys' (which is highlighted with a red box), 'My payments', 'My profile', and 'Logout'. The main content area displays a table for managing API keys. The table has columns: Key, Name, Status, Actions, and Create key. One row is shown with the key value '3aa334768a923f0ba39577cc3cf5eaee'. The 'Actions' column for this row contains two icons: a pencil and a delete symbol. Below the table, there's a text input field labeled 'API key name' and a 'Generate' button.

5. Paste the API key in a notepad and save the notepad file on your system.



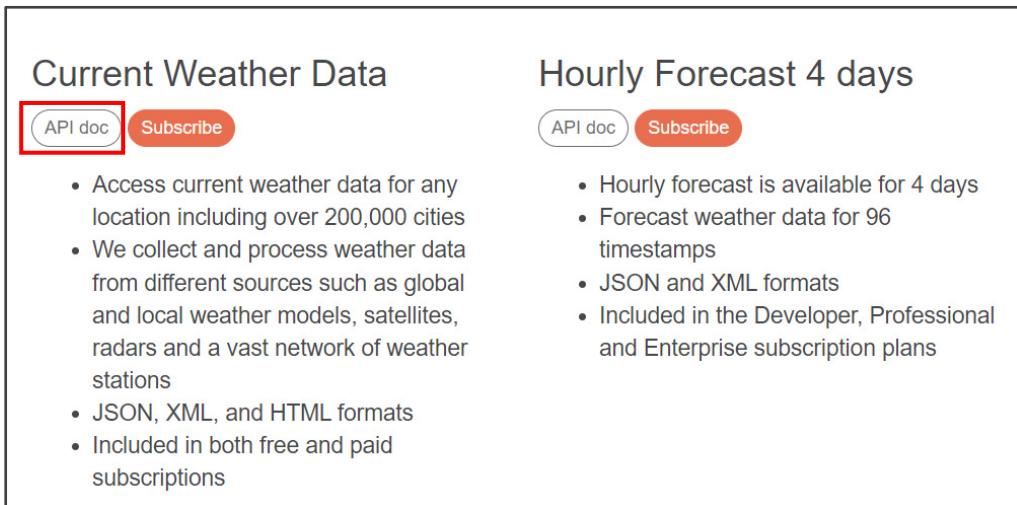
```
12c898cbe4d116dd6da26573d2e7e21e
```

6. Click **API**.



You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them.

7. Scroll down and locate **Current Weather Data**. Click **API doc**.



**Current Weather Data**

[API doc](#) [Subscribe](#)

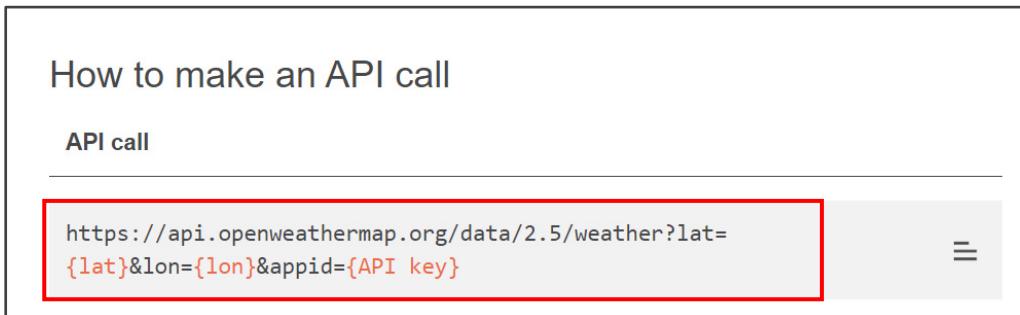
- Access current weather data for any location including over 200,000 cities
- We collect and process weather data from different sources such as global and local weather models, satellites, radars and a vast network of weather stations
- JSON, XML, and HTML formats
- Included in both free and paid subscriptions

**Hourly Forecast 4 days**

[API doc](#) [Subscribe](#)

- Hourly forecast is available for 4 days
- Forecast weather data for 96 timestamps
- JSON and XML formats
- Included in the Developer, Professional and Enterprise subscription plans

8. From the API Call section copy the URL and save it in a notepad file.



**How to make an API call**

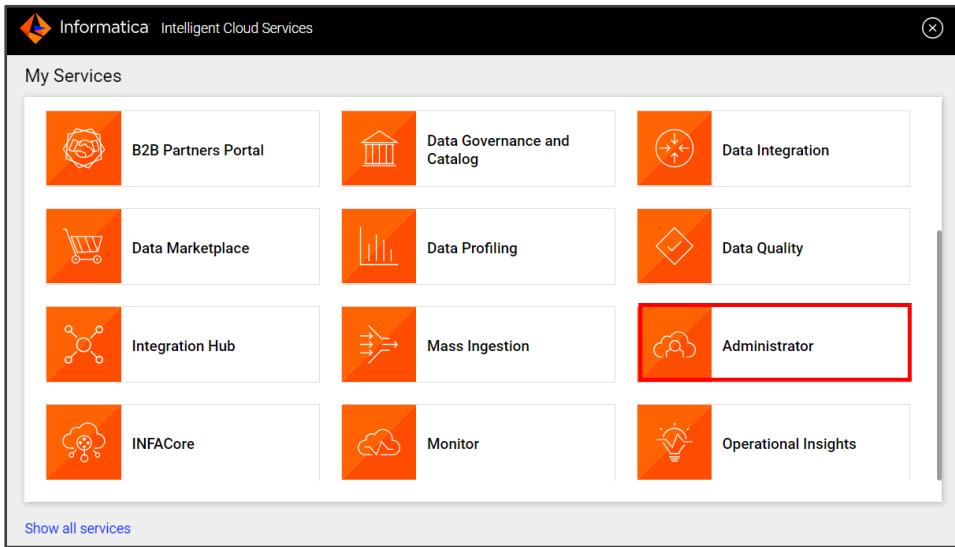
**API call**

```
https://api.openweathermap.org/data/2.5/weather?lat={lat}&lon={lon}&appid={API key}
```

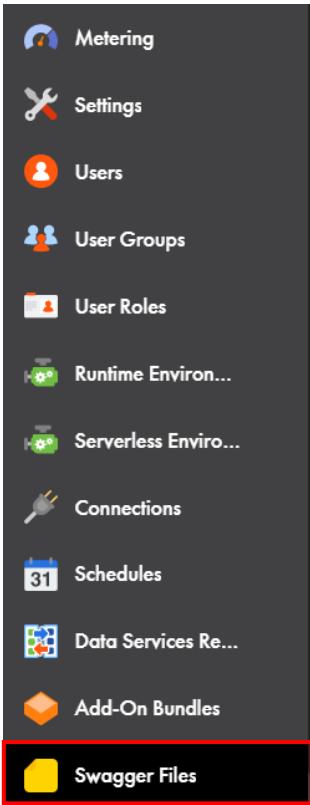
**Note:** We will be using this URL while generating the Swagger file.

## Create the CurrentForecastWeather Swagger File

9. Log into IICS and access the **Administrator** service.



10. To create a Swagger file, from the navigation pane, click **Swagger Files**.



11. Click **New**.

12. In the Name filed, enter **S\_XX\_CurrentForecastWeather**.
13. In the Runtime Environment filed, select your secure agent.
14. In the URL filed, enter the following URL: <https://api.openweathermap.org>
15. In the Verb filed, select **GET**.

16. In Authentication Type field, retain **None**.
17. In the API Base Path filed, enter **/data/2.5**
18. In the API Path field, enter **/weather/**

**Swagger File Details**

|                       |                                |
|-----------------------|--------------------------------|
| Name:*                | S_XX_CurrentForecastWeather    |
| Description:          |                                |
| Runtime Environment:* | CAI-XX-FIRSTNAME               |
| URL:*                 | https://api.openweathermap.org |
| Verb:                 | GET                            |
| Authentication Type:  | None                           |
| API Base Path:        | /data/2.5                      |
| API Path:             | /weather/                      |
| Username:             |                                |
| Password:             |                                |
| Token:                |                                |
| Token Secret:         |                                |
| Consumer Key:         |                                |

19. In the Accept field, select **application/json**.
20. In the Query Params filed, enter the following parameter:  
**{"lat":"<latitude>","lon":"<longitude>","appid":"<API Key>"}**  
**Note:** Replace latitude and longitude with the digits of your choice and API Key with the key you copied to the notepad.

**For example:**

```
{"lat":"44.34","lon":"10.99", "appid": "3aa334768a923f0ba39577cc3cf5eaee"}
```

**Note:** Sometimes, copy-pasting adds in some special characters and may not consider double quotes ("") in a proper format. In this case, the swagger file fails to save in step 22. If the file saving fails, manually re-enter all the **double quotes** present in Query Params field.

21. In the Operation Id field, enter **CurrentForecastWeather**.

|                            |   |
|----------------------------|---|
| Accept:                    | application/json  |
| Headers:                   |   |
| Query Params:              | [{"lat": "44.34", "lon": "10.99", "appid": "3aa334768a923f0ba39577cc3cf5eaee"}] |
| Operation Id: <sup>*</sup> | CurrentForecastWeather  |
| Content Type:              | application/json  |
| Raw Body:                  |   |
| JSON Response File:        | <input type="button" value="Upload..."/> ?                                      |

22. Click **Save**.

**Note:** A message **Swagger saved successfully** appears. Click the **Swagger Files** tab to check the file.

| Actions   | Name▲                       | Swagger JSON File           |
|---|-----------------------------|-----------------------------|
|   | S_XX_CurrentForecastWeather | CurrentForecastWeather.json |

23. Download S\_XX\_CurrentForecastWeather and save it on your machine.

| Actions   | Name▲                       | Swagger JSON File           |
|---|-----------------------------|-----------------------------|
|   | S_XX_CurrentForecastWeather | CurrentForecastWeather.json |

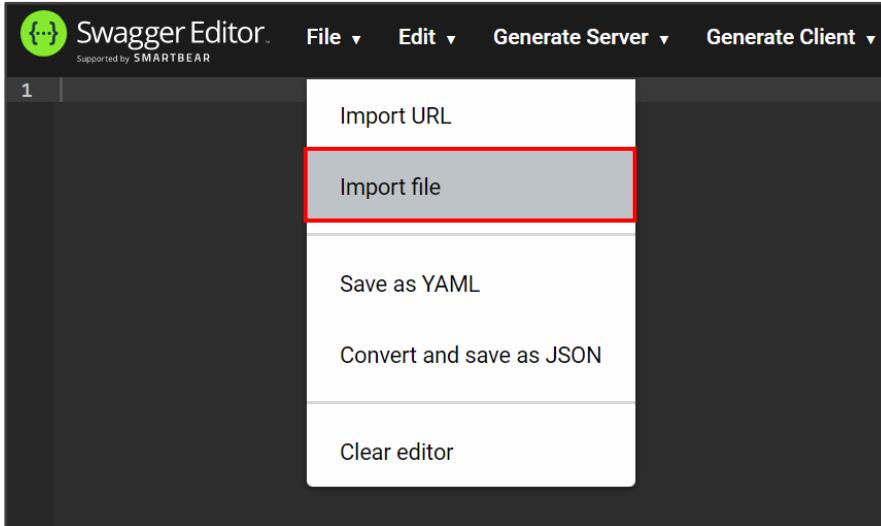
## Test the CurrentForecastWeather Swagger File

In the last step, you have created the S\_XX\_CurrentForecastWeather Swagger file. Now, you will test the correctness of the S\_XX\_CurrentForecastWeather Swagger file.

24. Open your web browser and enter the following URL:

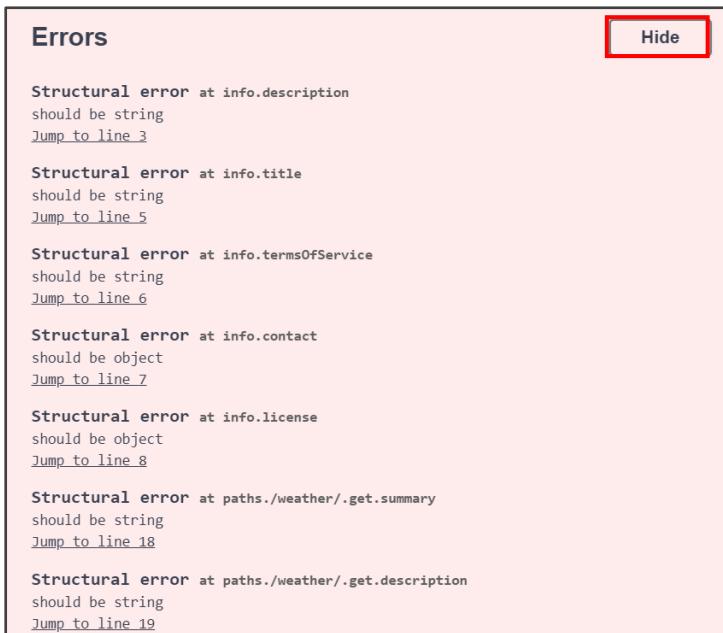
<https://editor.swagger.io/>

25. From the main menu, select **File > Import file**.



26. Select the downloaded **CurrentForecastWeather.json** file.

27. You can see a list of errors. Click **Hide** to ignore the errors.

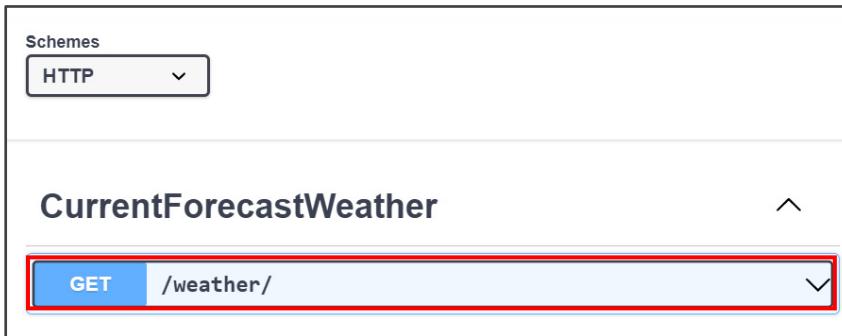


The 'Errors' panel displays a list of validation errors:

- Structural error at info.description  
should be string  
[Jump to line 3](#)
- Structural error at info.title  
should be string  
[Jump to line 5](#)
- Structural error at info.termsOfService  
should be string  
[Jump to line 6](#)
- Structural error at info.contact  
should be object  
[Jump to line 7](#)
- Structural error at info.license  
should be object  
[Jump to line 8](#)
- Structural error at paths./weather/.get.summary  
should be string  
[Jump to line 18](#)
- Structural error at paths./weather/.get.description  
should be string  
[Jump to line 19](#)

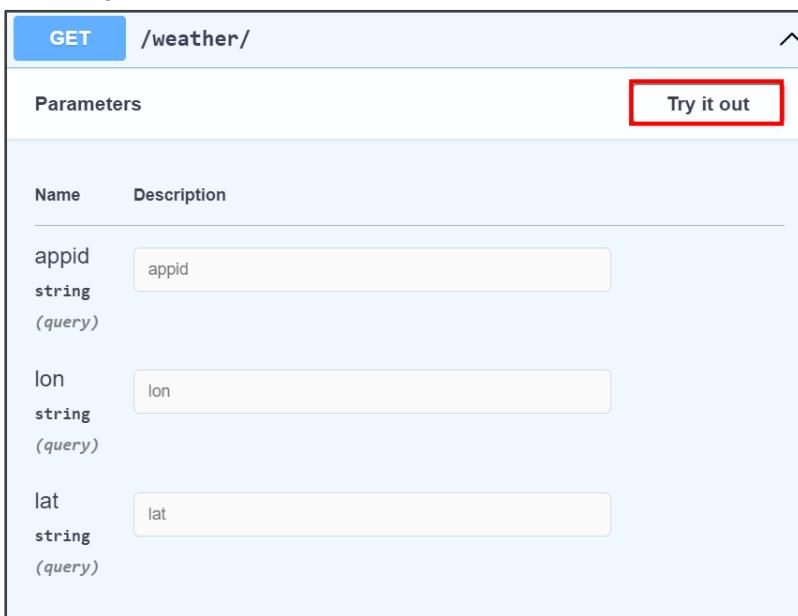
A red box highlights the 'Hide' button in the top right corner of the panel.

28. Click the **GET/weather/** field to expand it.



The screenshot shows a REST API endpoint configuration. At the top, there is a dropdown menu labeled "Schemes" with "HTTP" selected. Below this, the endpoint name "CurrentForecastWeather" is displayed. Underneath the endpoint name, there is a row containing a blue button labeled "GET" and a text input field containing "/weather/". The entire "/weather/" field is highlighted with a red rectangular border.

29. Click **Try it out**.



The screenshot shows the "Try it out" interface for the "/weather/" endpoint. At the top, there is a row with a blue button labeled "GET" and a text input field containing "/weather/". Below this, there is a section titled "Parameters" with three fields: "appid", "lon", and "lat". To the right of the "Parameters" section is a red rectangular box highlighting the "Try it out" button. The "Parameters" section has a table header with columns "Name" and "Description". The three parameters are listed as rows in the table.

| Name  | Description |
|-------|-------------|
| appid | appid       |
| lon   | lon         |
| lat   | lat         |

30. In the appid filed, enter the API Key that you have copied to the notepad.

31. In the lon filed, enter **10.99**.

32. In the lat field, enter **44.34**.

33. Click **Execute**.

| Parameters        |                                  | Cancel |
|-------------------|----------------------------------|--------|
| Name              | Description                      |        |
| appid             | 3aa334768a923f0ba39577cc3cf5eaee |        |
| string<br>(query) |                                  |        |
| lon               | 10.99                            |        |
| string<br>(query) |                                  |        |
| lat               | 44.34                            |        |
| string<br>(query) |                                  |        |
| <b>Execute</b>    |                                  |        |

34. In the Response body section, observes the latitude and longitude values. You can see the same values which you have passed in the previous step.

Response body

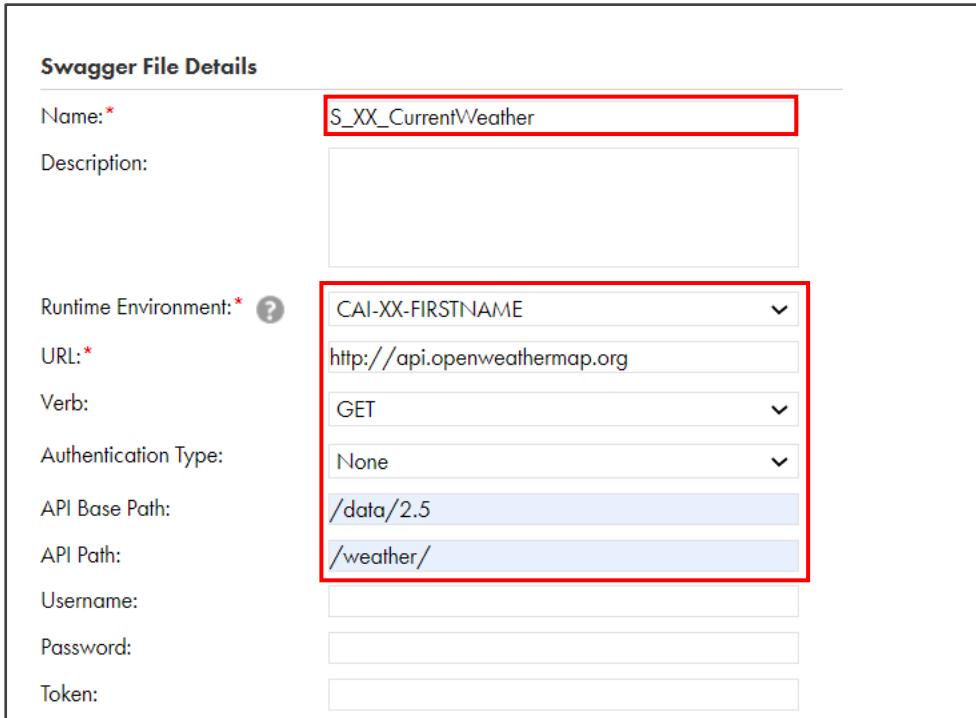
```
{
  "coord": {
    "lon": 10.99,
    "lat": 44.34
  },
  "weather": [
    {
      "id": 800,
      "main": "Clear",
      "description": "clear sky",
      "icon": "01d"
    }
  ],
  "base": "stations",
  "main": {
    "temp": 282.68,
    "feels_like": 282.68,
    "temp_min": 276.56,
    "temp_max": 286.53,
    "pressure": 1029,
    "humidity": 47,
    "sea_level": 1029,
    "grnd_level": 941
  }
}
```

## Skill Application

### Create a Swagger File

In this section of lab, create a new Swagger file and test it using the <https://editor.swagger.io/> URL. Use the following parameters to create and test the Swagger file.

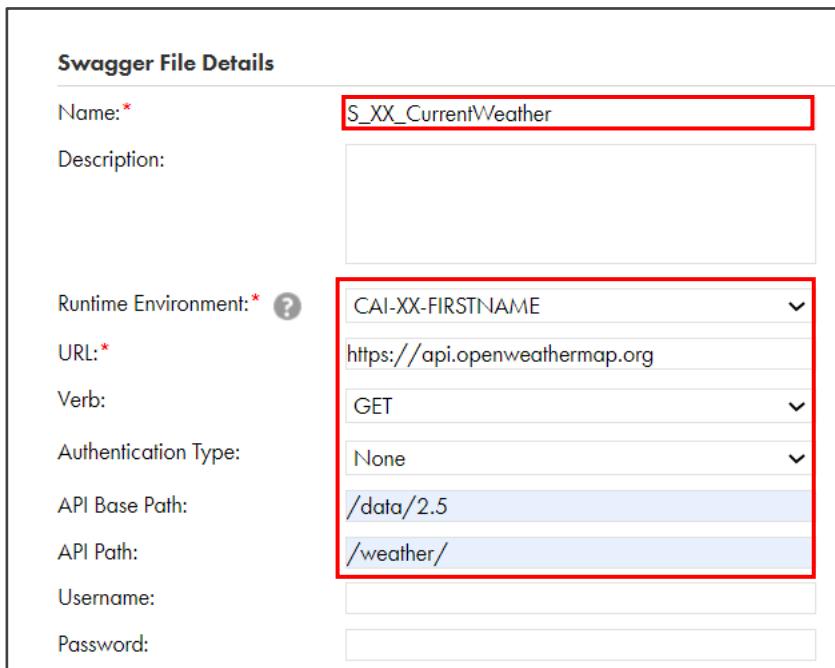
35. In the Name filed, enter **S\_XX\_CurrentWeather**.
36. In the Runtime Environment filed, select your secure agent.
37. In the URL filed, enter the following URL: <https://api.openweathermap.org>
38. In the Verb filed, select **Get**.
39. In Authentication Type filed, retain **None**.
40. In the API Base Path filed, enter **/data/2.5**
41. In the API Path field, enter **/weather/**



| Swagger File Details   |                               |
|------------------------|-------------------------------|
| Name: *                | S_XX_CurrentWeather           |
| Description:           | <input type="text"/>          |
| Runtime Environment: * | CAI-XX-FIRSTNAME              |
| URL: *                 | http://api.openweathermap.org |
| Verb:                  | GET                           |
| Authentication Type:   | None                          |
| API Base Path:         | /data/2.5                     |
| API Path:              | /weather/                     |
| Username:              | <input type="text"/>          |
| Password:              | <input type="text"/>          |
| Token:                 | <input type="text"/>          |

42. In the Accept field, select **application/json**.
43. In the Query Params filed, enter the following parameter:  
`{"q":<City Name>","AppID":<API Key>}`  
**Note:** Replace City Name with the city of your choice and API Key with the key you copied on the notepad.  
For example:  
`{"q":"delhi","AppID":"3aa334768a923f0ba39577cc3cf5eaee"}`

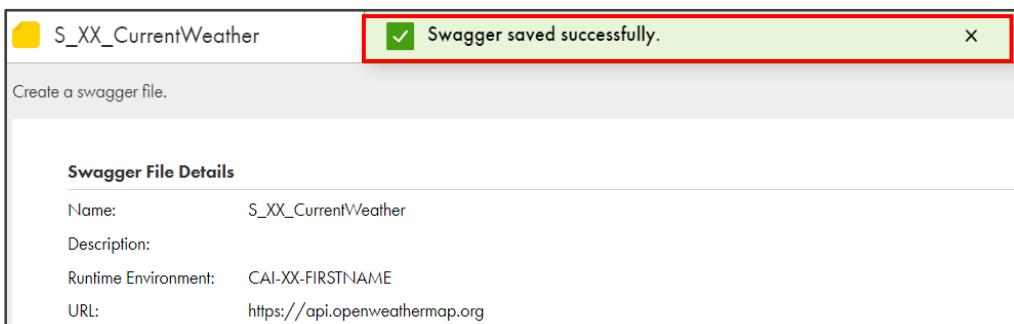
44. In the Operation Id field, enter **CityWeatherReport**.



**Swagger File Details**

|                       |                                |
|-----------------------|--------------------------------|
| Name:*                | S_XX_CurrentWeather            |
| Description:          |                                |
| Runtime Environment:* | CAI-XX-FIRSTNAME               |
| URL:*                 | https://api.openweathermap.org |
| Verb:                 | GET                            |
| Authentication Type:  | None                           |
| API Base Path:        | /data/2.5                      |
| API Path:             | /weather/                      |
| Username:             |                                |
| Password:             |                                |

45. Save and download the Swagger file.



S\_XX\_CurrentWeather

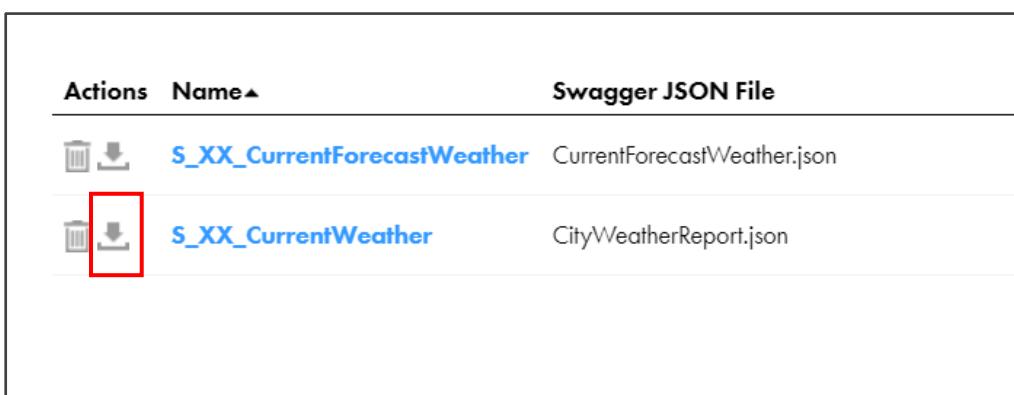
Swagger saved successfully.

Create a swagger file.

**Swagger File Details**

|                      |                                |
|----------------------|--------------------------------|
| Name:                | S_XX_CurrentWeather            |
| Description:         |                                |
| Runtime Environment: | CAI-XX-FIRSTNAME               |
| URL:                 | https://api.openweathermap.org |

**Note:** A message **Swagger saved successfully** appears. Click the **Swagger Files** tab to check the file.



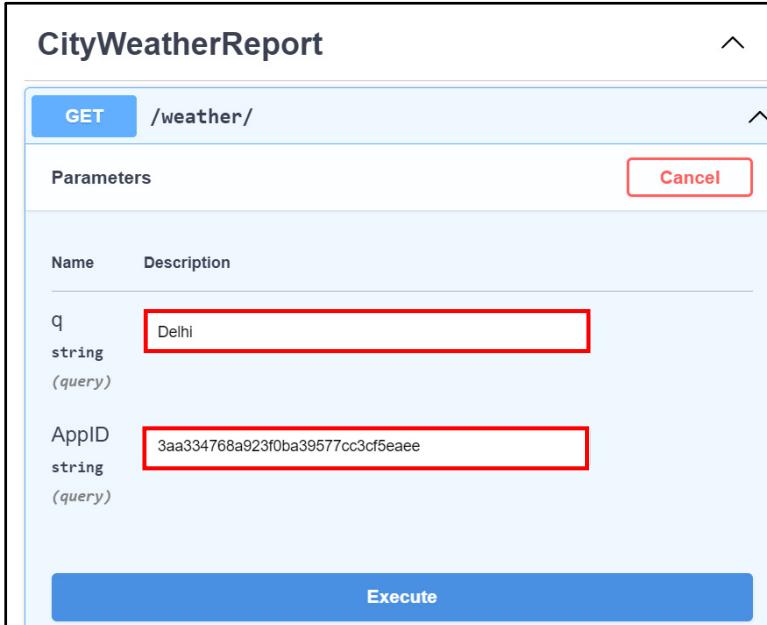
| Actions | Name▲                                       | Swagger JSON File           |
|---------|---|-----------------------------|
|         | <a href="#">S_XX_CurrentForecastWeather</a> | CurrentForecastWeather.json |
|         | <a href="#">S_XX_CurrentWeather</a>         | CityWeatherReport.json      |

46. Test the Swagger file using the <https://editor.swagger.io/> URL.

47. In the q filed, enter **Delhi**.

**Note:** Instead of Delhi, you can enter a different city in the q field.

48. In the appid filed, enter the API Key that you have copied to the notepad.



49. In the Response body field, you will get the current weather report of the city you entered (here, it is Delhi).



```

Response body
{
  "weather": [
    {
      "id": 711,
      "main": "Smoke",
      "description": "smoke",
      "icon": "50d"
    }
  ],
  "base": "stations",
  "main": {
    "temp": 299.2,
    "feels_like": 299.2,
    "temp_min": 299.2,
    "temp_max": 299.2
  },
  "type": 1,
  "id": 9165,
  "country": "IN",
  "sunrise": 1676510961,
  "sunset": 1676551284
},
"timezone": 19800,
"id": 1273294,
"name": "Delhi",
"cod": 200
}

```

The screenshot shows the response body of the API call. It contains JSON data describing the weather in Delhi. The "weather" array has one item with id 711, main "Smoke", description "smoke", and icon "50d". The "base" is "stations". The "main" object has temp 299.2, feels\_like 299.2, temp\_min 299.2, and temp\_max 299.2. The "type" is 1, "id" is 9165, "country" is "IN", "sunrise" is 1676510961, and "sunset" is 1676551284. The "timezon" is 19800, "id" is 1273294, "name" is "Delhi", and "cod" is 200. A red box highlights the "weather" array.

---

*This concludes the lab.*

## Module 5: Service Connectors

### Lab 5-4: Create a Service Connector from a Swagger File

#### Overview:

You can import a Swagger JSON file to easily create a service connector with multiple operations. Use the New Service Connector from Swagger wizard to import a Swagger JSON file and create a service connector. After you specify the Swagger JSON file, Application Integration parses the file and loads the operations.

#### Objectives:

- Import a Swagger JSON File
- Modify an existing Action
- Test the Service Connector

#### Duration:

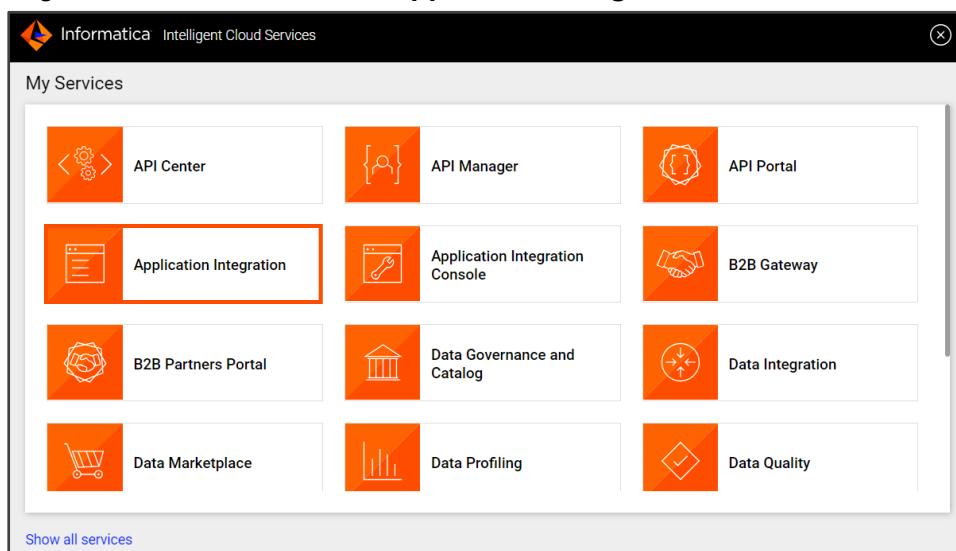
15 Minutes

---

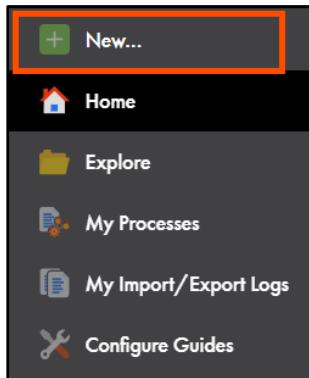
## Tasks

### Import a Swagger File

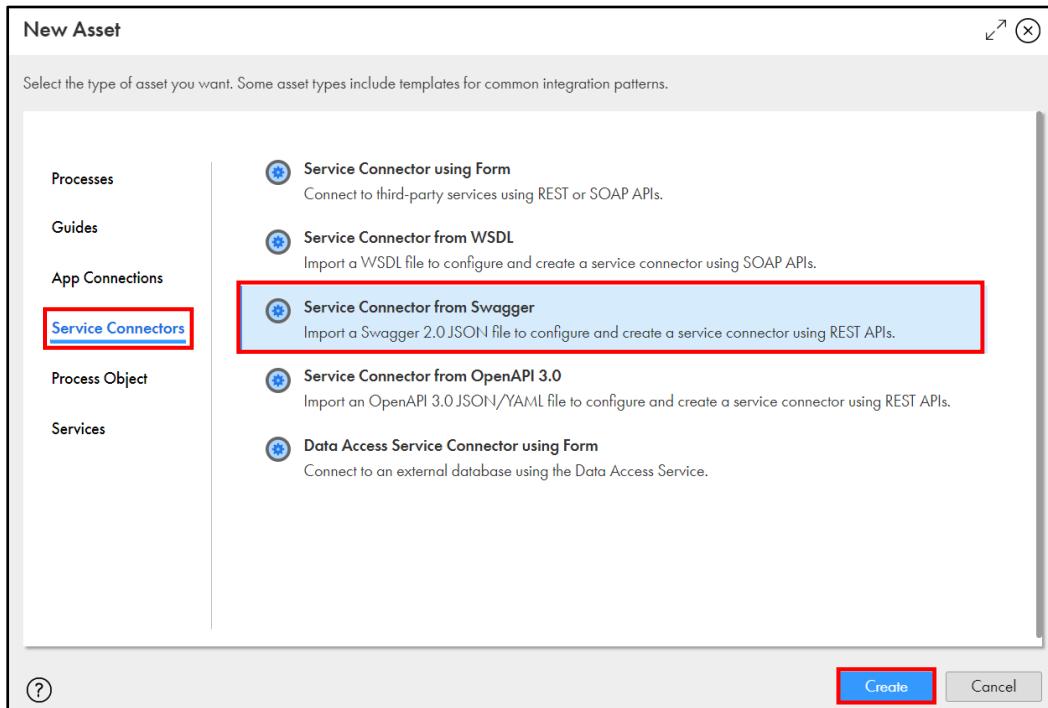
1. Log into IICS and access the **Application Integration** service.



- To create a new Service Connector, from the navigation pane, click **New**.



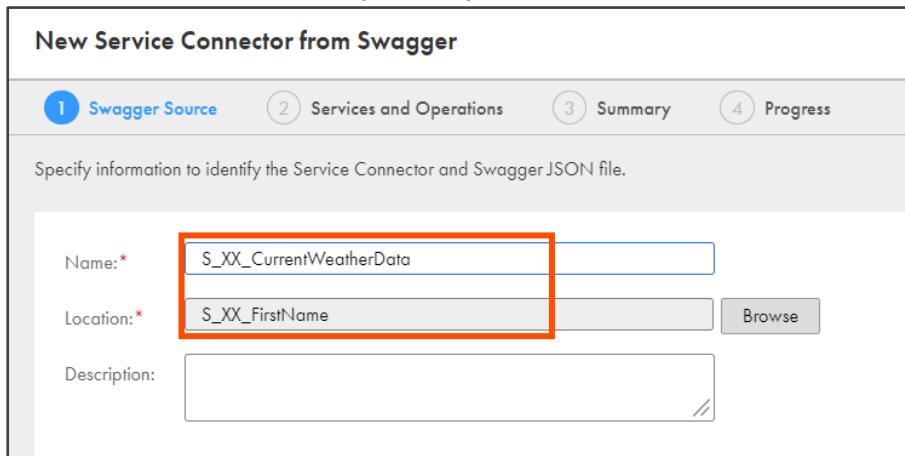
- From the Service Connectors tab, select **Service Connector from Swagger**.
- Click **Create**.



A new Service Connector page appears.

- In the service connector Name field, enter **S\_XX\_CurrentWeatherData**.

6. In the Location field, select your project folder **S\_XX\_FirstName**.



New Service Connector from Swagger

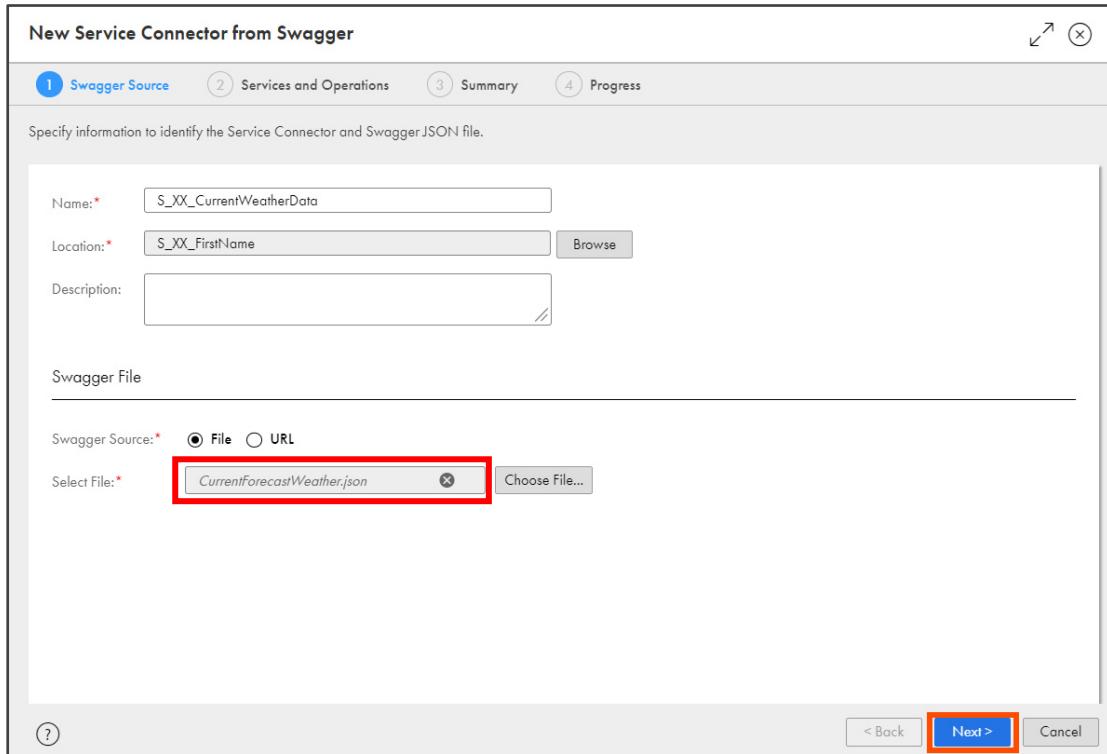
Specify information to identify the Service Connector and Swagger JSON file.

|              |                         |
|--------------|-------------------------|
| Name:*       | S_XX_CurrentWeatherData |
| Location:*   | S_XX_FirstName          |
| Description: |                         |

7. From Swagger Source, select **File**.  
 8. To select the source file, click **Choose File**, and select the **CurrentForecastWeather.json** file.

**Note:** You have created the CurrentForecastWeather.json file in the previous lab.

9. Click **Next**.



New Service Connector from Swagger

Specify information to identify the Service Connector and Swagger JSON file.

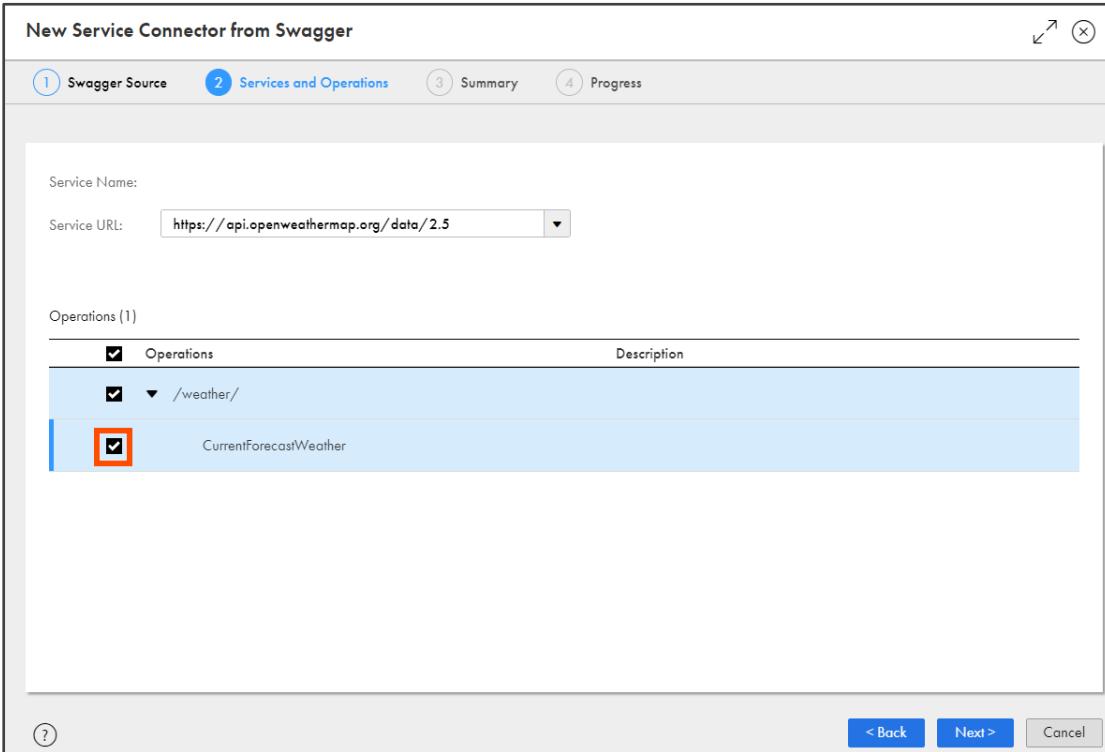
|              |                         |
|--------------|-------------------------|
| Name:*       | S_XX_CurrentWeatherData |
| Location:*   | S_XX_FirstName          |
| Description: |                         |

Swagger File

Swagger Source:  File  URL

Select File: \*

10. Under Operations, expand and verify that the **CurrentForecastWeather** checkbox is selected.



New Service Connector from Swagger

1 Swagger Source    2 Services and Operations    3 Summary    4 Progress

Service Name:

Service URL:

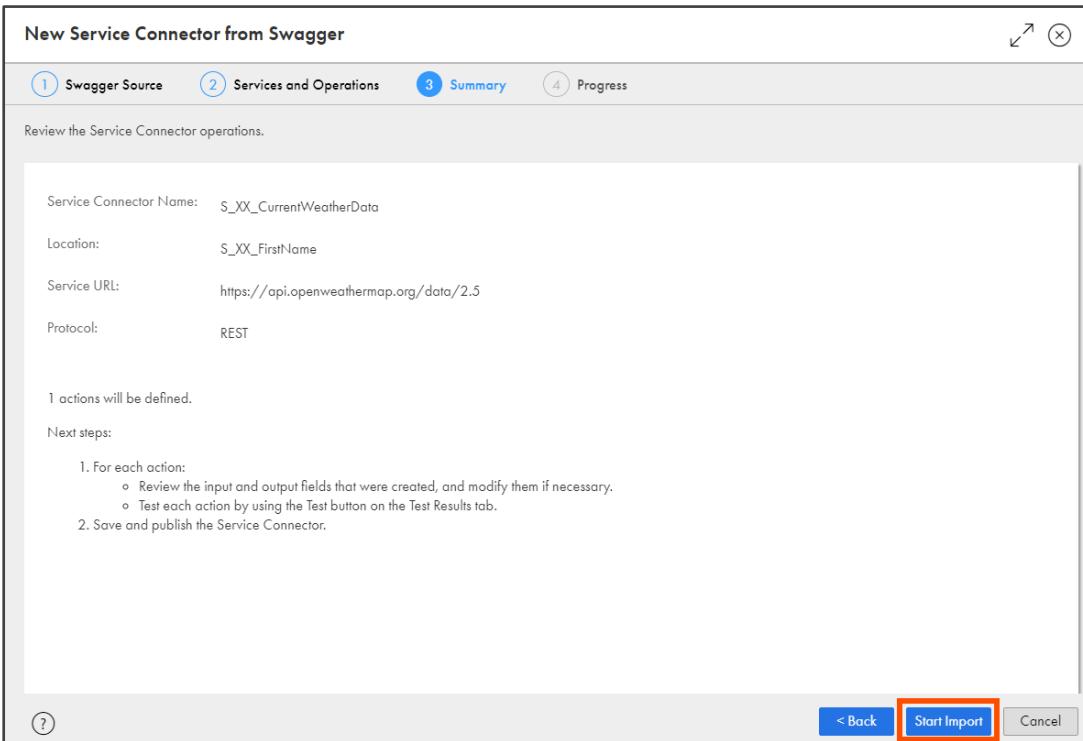
Operations (1)

| Operations   | Description |
|--|-------------|
| <input checked="" type="checkbox"/> /weather/              |             |
| <input checked="" type="checkbox"/> CurrentForecastWeather |             |

< Back    Next >    Cancel

11. Click **Next**.

12. Click **Start Import**.



New Service Connector from Swagger

1 Swagger Source    2 Services and Operations    3 Summary    4 Progress

Review the Service Connector operations.

Service Connector Name: S\_XX\_CurrentWeatherData

Location: S\_XX\_FirstName

Service URL: https://api.openweathermap.org/data/2.5

Protocol: REST

1 actions will be defined.

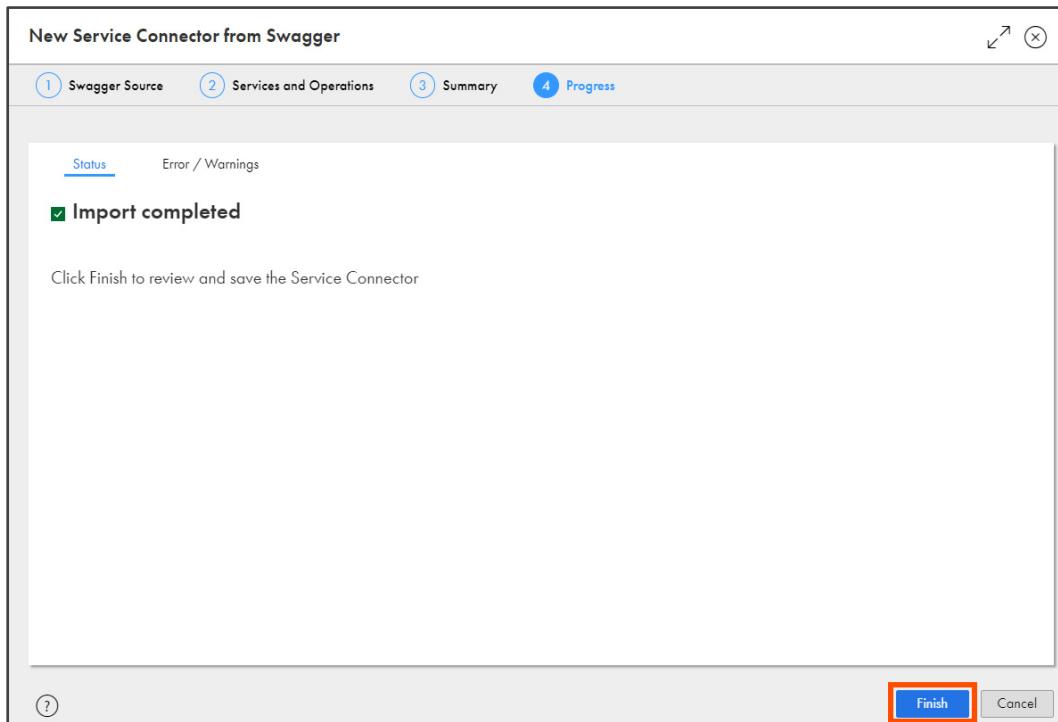
Next steps:

1. For each action:
  - Review the input and output fields that were created, and modify them if necessary.
  - Test each action by using the Test button on the Test Results tab.
2. Save and publish the Service Connector.

< Back    Start Import    Cancel

13. Click **Finish**.

**Note:** Ideally, it will take a few seconds to finish the Import operation. If the Importing process takes longer than 5 minutes, click on the Summary tab (the previous tab), and start the import again.

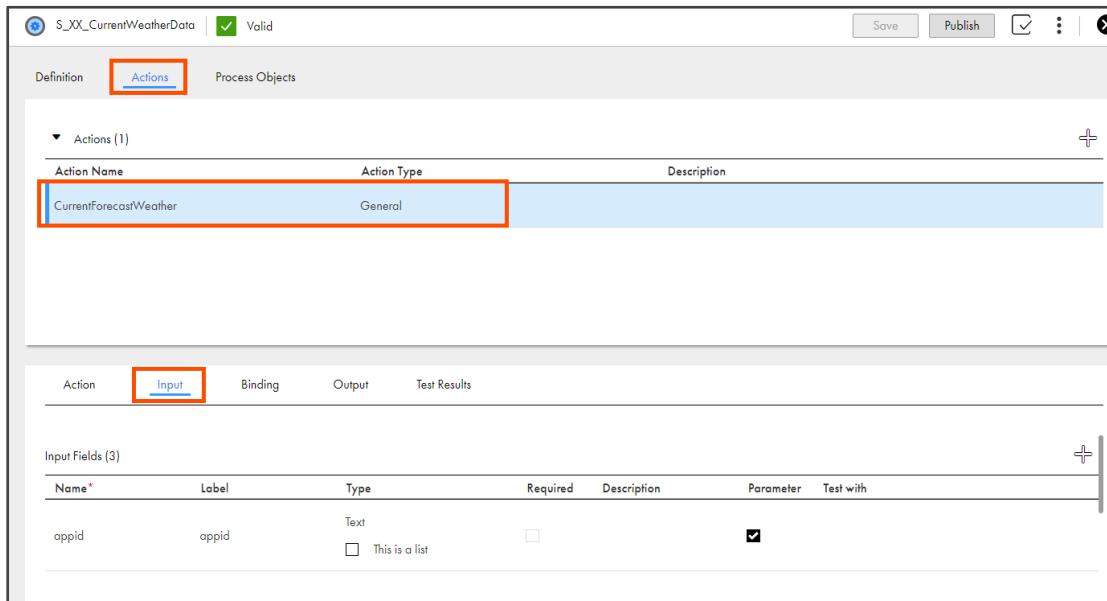
14. Click **Save**.

A message **The service connector was saved successfully** appears.

**Add Parameters to the Service Connector**15. Click **Actions** and select the Input tab.

16. From the Action Name, select **CurrentForecastWeather**.

**Note:** There is only one action in the Action Name section.



| Action Name            | Action Type | Description |
|------------------------|-------------|-------------|
| CurrentForecastWeather | General     |             |

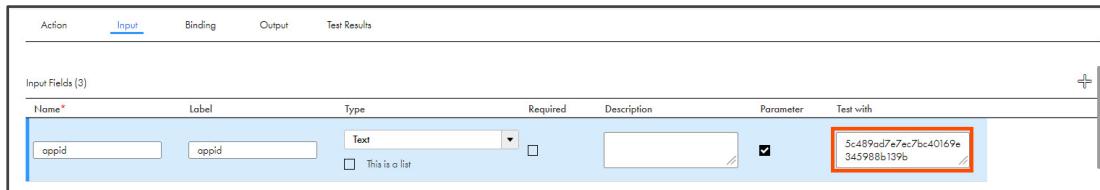
  

| Action | Input | Binding | Output | Test Results |
|--------|-------|---------|--------|--------------|
|        |       |         |        |              |

| Input Fields (3) |       |      |                          |  |                                     |                                      |
|------------------|-------|------|--------------------------|--|-------------------------------------|--------------------------------------|
| Name*            | Label | Type | Required                 | Description  | Parameter                           | Test with                            |
| appid            | appid | Text | <input type="checkbox"/> | <input checked="" type="checkbox"/> This is a list | <input checked="" type="checkbox"/> | Sc499ad7e7ec7bc40169e<br>34598861396 |

17. Click inside the Input Field **appid** and in the **Test with** field, enter the API Key which you have saved in notepad file.

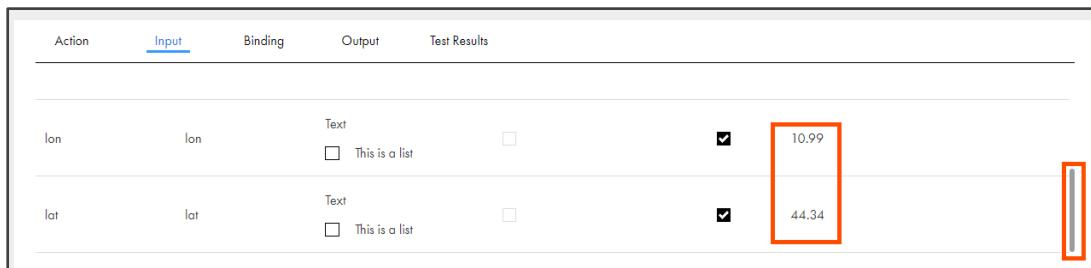


| Name* | Label | Type | Required                 | Description  | Parameter                           | Test with                            |
|-------|-------|------|--------------------------|--|-------------------------------------|--------------------------------------|
| appid | appid | Text | <input type="checkbox"/> | <input checked="" type="checkbox"/> This is a list | <input checked="" type="checkbox"/> | Sc499ad7e7ec7bc40169e<br>34598861396 |

18. In the lon Parameter field, enter **10.99**.

19. In the lat Parameter field, enter **44.34**.

**Tip:** Use the scroll bar present towards the right of the screen to locate the rest of the Input Fields.



| Action | Input | Binding | Output   | Test Results                        |       |
|--------|-------|---------|--|-------------------------------------|-------|
| lon    | lon   | Text    | <input type="checkbox"/><br><input checked="" type="checkbox"/> This is a list | <input checked="" type="checkbox"/> | 10.99 |
| lat    | lat   | Text    | <input type="checkbox"/><br><input checked="" type="checkbox"/> This is a list | <input checked="" type="checkbox"/> | 44.34 |

20. Click **Save**.

## Test Action

### 21. Select Test Results.

| Action   | Input | Binding | Output | Test Results |
|--|-------|---------|--------|--------------|
|  |       |         |        | <b>Test</b>  |
| Test Server: * <input type="button" value="Test on Cloud Server"/> |       |         |        | ▼            |

### 22. From the Test Server drop-down, retain **Test on Cloud Server** and click **Test**.

| Action   | Input | Binding | Output | Test Results |
|--|-------|---------|--------|--------------|
|  |       |         |        | <b>Test</b>  |
| Test Server: * <input type="button" value="Test on Cloud Server"/> |       |         |        | ▼            |

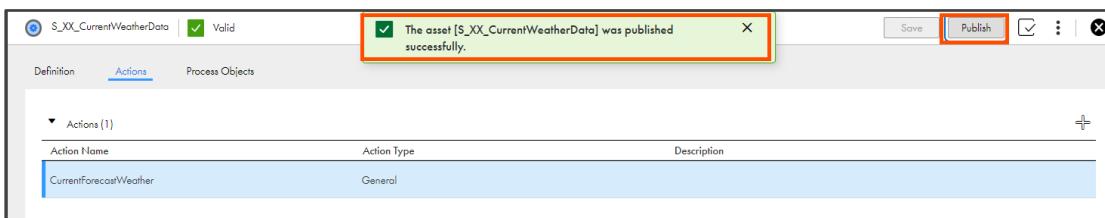
### 23. In the Test Results tab, verify that the result of the test is **Successful**. Observe that the HTTPS Status is **200**.

| Action   | Input | Binding | Output | Test Results |
|--|-------|---------|--------|--------------|
|  |       |         |        | <b>Test</b>  |
| Test Server: * <input type="button" value="Test on Cloud Server"/>   |       |         |        | ▼            |
| Result: <input checked="" type="checkbox"/> Successful.  |       |         |        | <b>Test</b>  |
| HTTP Status: <b>200</b>  |       |         |        |              |
| URL: <a href="https://api.openweathermap.org/data/2.5/weather/?appid=5c489ad7e7ec7bc40169e345988b139b&amp;lon=10.99&amp;lat=44.34">https://api.openweathermap.org/data/2.5/weather/?appid=5c489ad7e7ec7bc40169e345988b139b&amp;lon=10.99&amp;lat=44.34</a> |       |         |        |              |

### 24. Scroll down and from the Output Fields section, expand **Response Payload** section and observe the output.

| Action   | Input | Binding | Output | Test Results |
|--|-------|---------|--------|--------------|
|  |       |         |        | <b>Test</b>  |
| <p>▼ Response Payload</p> <pre> 1 v &lt;root&gt; 2   &lt;visibility&gt;10000&lt;/visibility&gt; 3   &lt;timezone&gt;7200&lt;/timezone&gt; 4 v   &lt;main&gt; 5     &lt;temp&gt;304.48&lt;/temp&gt; 6     &lt;temp_min&gt;301.56&lt;/temp_min&gt; 7     &lt;grnd_level&gt;936&lt;/grnd_level&gt; </pre> |       |         |        | ▼            |

## 25. Click Publish.



A message "**The asset [S\_XX\_CurrentWeatherData] was published successfully**" appears on the screen.

---

*This concludes the lab.*

## Module 5: Service Connectors

### Lab 5-5: Create a Service Connector from OpenAPI 3.0

#### Overview:

You can import an OpenAPI 3.0 JSON or YAML file to easily create a service connector with multiple operations. Use the New Service Connector from OpenAPI 3.0 wizard to import a OpenAPI 3.0 file and create a service connector. After you specify the OpenAPI 3.0 file, Application Integration parses the file and loads the operations.

In this lab, you will create a Service Connector from OpenAPI 3.0. To do so, you will use the OpenAPI 3.0 URL that you have created in the **Lab 3-1: Create a Basic Process to Display User Input**.

#### Objectives:

- Obtain the OpenAPI 3.0 URL
- Create a Service Connector from OpenAPI 3.0
- Add Parameter to an existing Action
- Test the Service Connector

#### Duration:

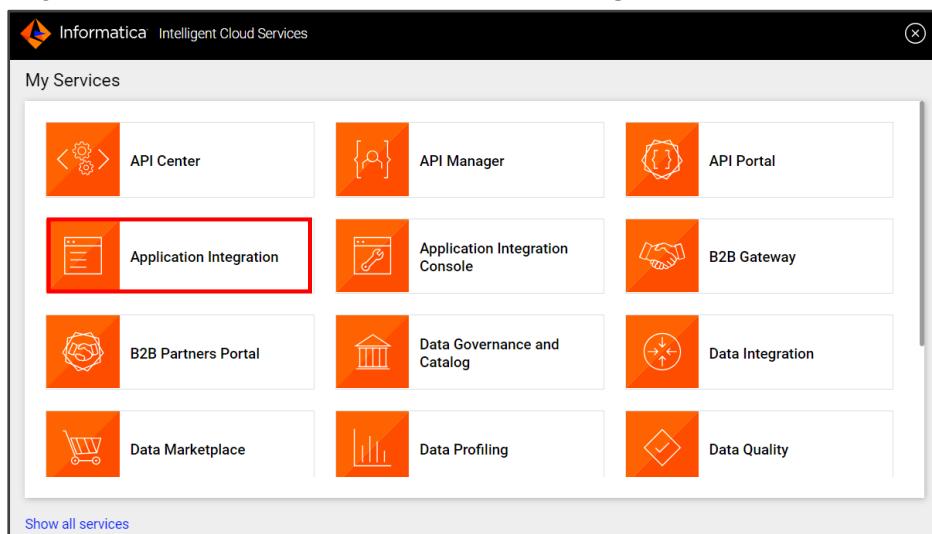
10 Minutes

---

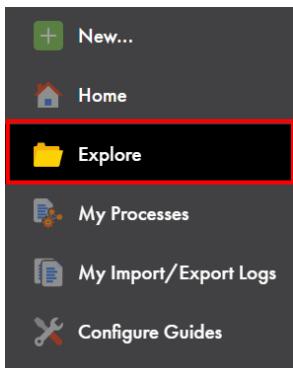
#### Tasks

##### Obtain the OpenAPI 3.0 URL

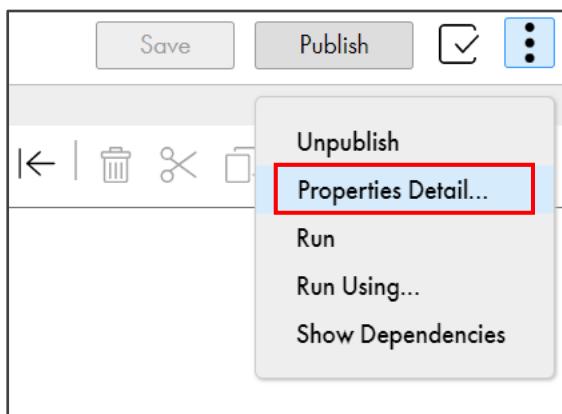
1. Log into IICS and access the **Application Integration** service.



2. Click **Explore**.



3. Navigate to your project folder and open the **S\_XX\_Print Input Message** process.  
 4. From the Action menu, select **Properties Detail**.



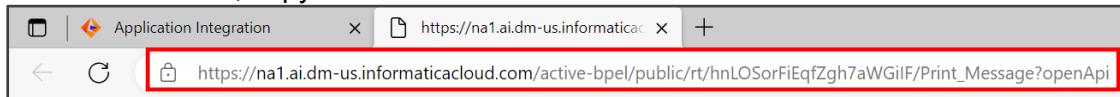
5. From Service URL, click **View OpenAPI 3.0 File**.



|   |                                     |
|---|-------------------------------------|
| Service URL: <a href="https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfEqfZgh7aWGiF/Print_Message">https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfEqfZgh7aWGiF/Print_Message</a>          | <input type="button" value="Copy"/> |
| <a href="#">"View Swagger File"</a> <a href="#">"View OpenAPI 3.0 File"</a>   |                                     |
| SOAP Service URL: <a href="https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/soap/hnLOSoRfEqfZgh7aWGiF/Print_Message">https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/soap/hnLOSoRfEqfZgh7aWGiF/Print_Message</a> | <input type="button" value="Copy"/> |
| <a href="#">"View WSDL File"</a>  |                                     |

The URL will open in a new browser.

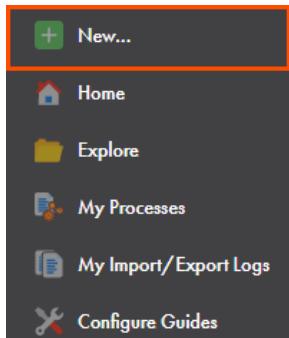
6. From the browser, copy the service URL.



7. Open a notepad file and paste the URL.  
 8. Close the Properties Details window.

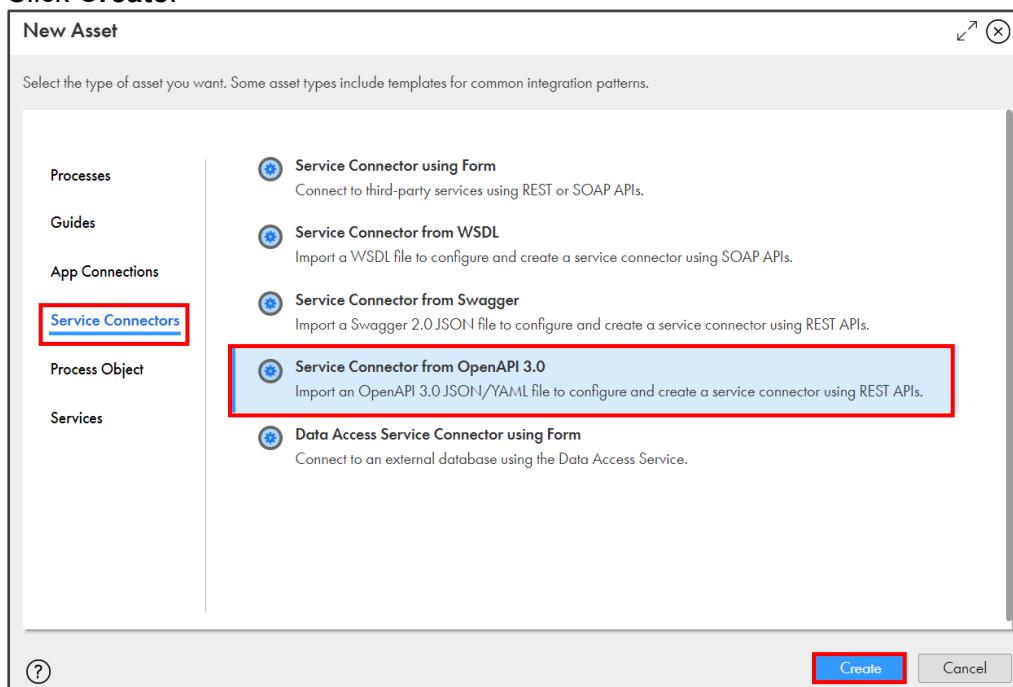
## Create a Service Connector from OpenAPI 3.0

9. Go back to the IICS Application Integration service page and from the navigation pane, click **New**.



10. From the Service Connectors tab, select **Service Connector from OpenAPI 3.0**.

11. Click **Create**.



A new Service Connector page appears.

12. In the service connector Name field, enter **S\_XX\_OpenAPI Print Message**.

13. In the Location field, select your project folder **S\_XX\_FirstName**.

New Service Connector from OpenAPI 3.0

1 OpenAPI 3.0 Source    2 Services and Operations    3 Summary    4 Progress

Specify information to identify the Service Connector and OpenAPI 3.0 JSON/YAML file.

|              |                            |
|--------------|----------------------------|
| Name: *      | S_XX_OpenAPI Print Message |
| Location: *  | S_XX_FirstName             |
| Description: |                            |

14. From OpenAPI 3.0 Source, make sure that **URL** is selected.

15. Copy the URL from the notepad and paste it in the OpenAPI 3.0 URL field.

16. Verify that the **Use Authentication** checkbox is cleared.

OpenAPI 3.0 File

OpenAPI 3.0 Source: \*  File  URL

OpenAPI 3.0 URL: \*

Use Authentication:

User Name:

< Back    Next >    Cancel

17. Click **Next**.

18. Expand **/Print\_Message** and verify that **Print\_MessageOperation** checkbox is selected.

Service Name: Print\_Message

Service URL: https://na1.ai.dm-us.informaticacloud.com:443/active...

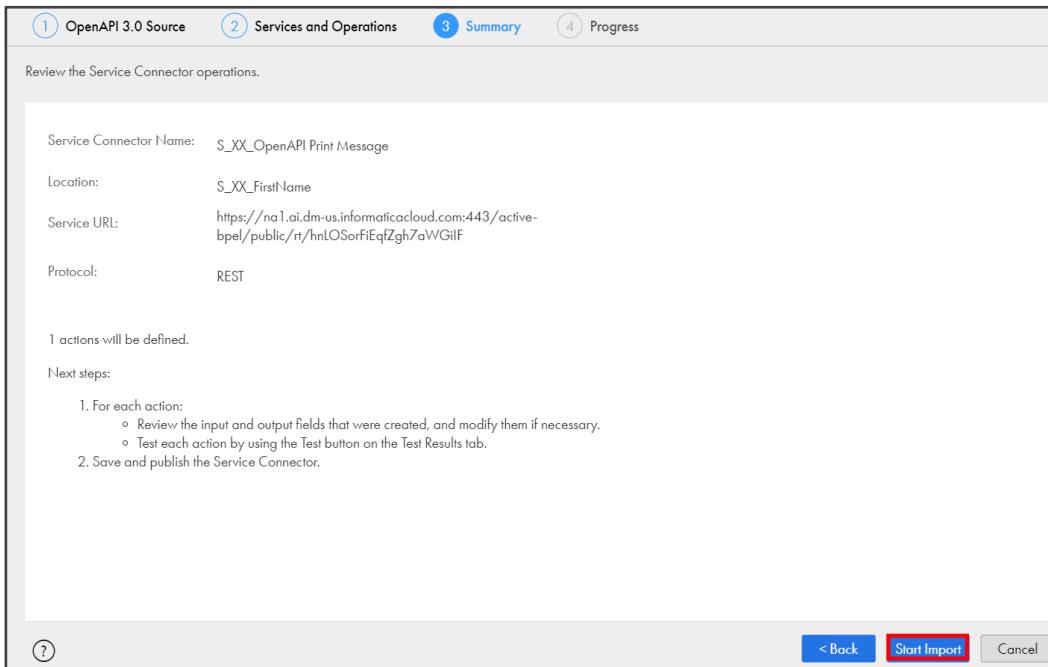
Operations (1)

| Operations   | Description |
|--|-------------|
| <input checked="" type="checkbox"/> /Print_Message         |             |
| <input checked="" type="checkbox"/> Print_MessageOperation |             |

< Back    Next >    Cancel

19. Click **Next**.

20. Click **Start Import**.



Review the Service Connector operations.

Service Connector Name: S\_XX\_OpenAPI Print Message

Location: S\_XX\_FirstName

Service URL: <https://na1.ai.dm-us.informaticacloud.com:443/active-bpel/public/r/hnLOSoRfEqfZgh7aWGifF>

Protocol: REST

1 actions will be defined.

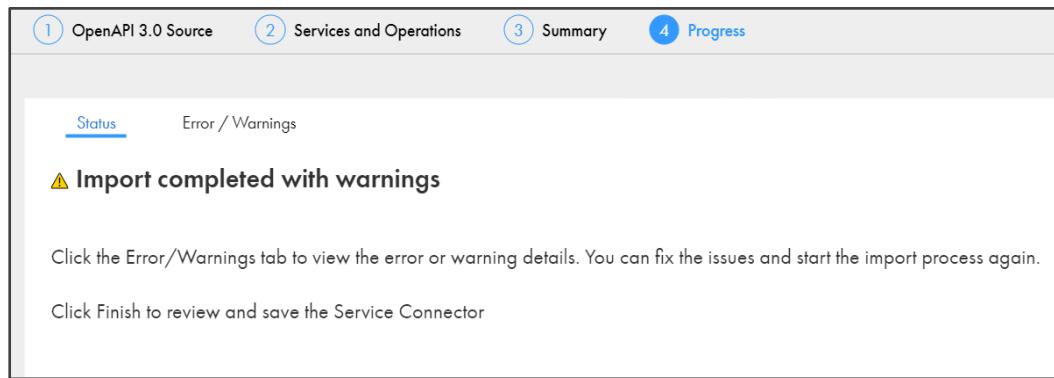
Next steps:

1. For each action:
  - o Review the input and output fields that were created, and modify them if necessary.
  - o Test each action by using the Test button on the Test Results tab.
2. Save and publish the Service Connector.

< Back Start Import Cancel

**Notes:**

- a. You can see a message that **Import Completed with warnings**. You can ignore this message as it will not stop you to move further.



Status Error / Warnings

⚠ Import completed with warnings

Click the Error/Warnings tab to view the error or warning details. You can fix the issues and start the import process again.

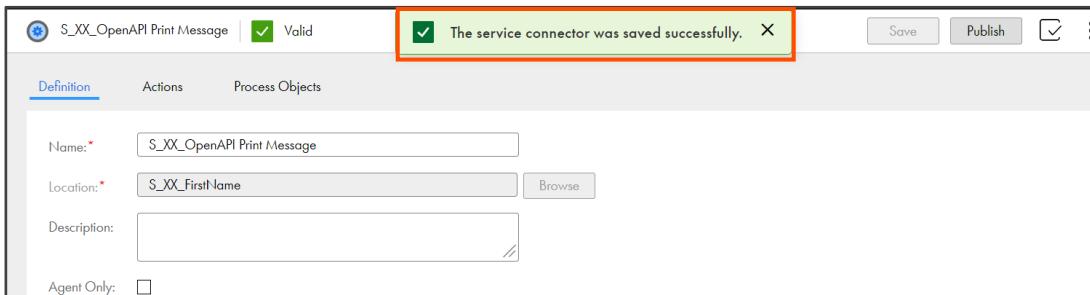
Click Finish to review and save the Service Connector

- b. If an error occurs during import, the wizard displays the error under Error/Warnings on the Progress tab. You can take corrective actions and import the OpenAPI 3.0 file again.

21. Click **Finish**.

22. Click **Save**.

A message “The service connector was saved successfully” appears.



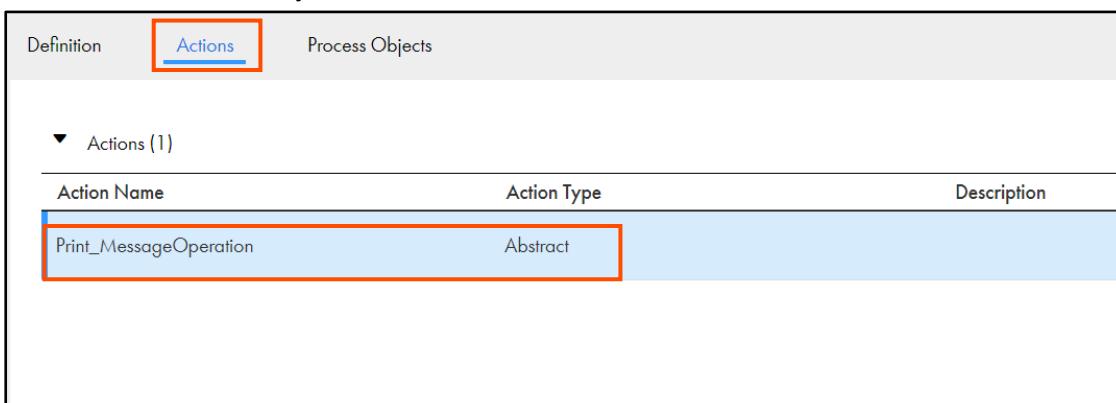
The screenshot shows the 'Definition' tab of a service connector named 'S\_XX\_OpenAPI Print Message'. The 'Actions' tab is selected. A green notification bar at the top right says 'The service connector was saved successfully.' with a close button. Below it, the 'Actions' tab has a single row: 'Action Name: Print\_MessageOperation' and 'Action Type: Abstract'. There are also tabs for 'Process Objects' and 'Save/Publish' buttons.

### Add Parameters to the Action

23. Click the **Actions** tab.

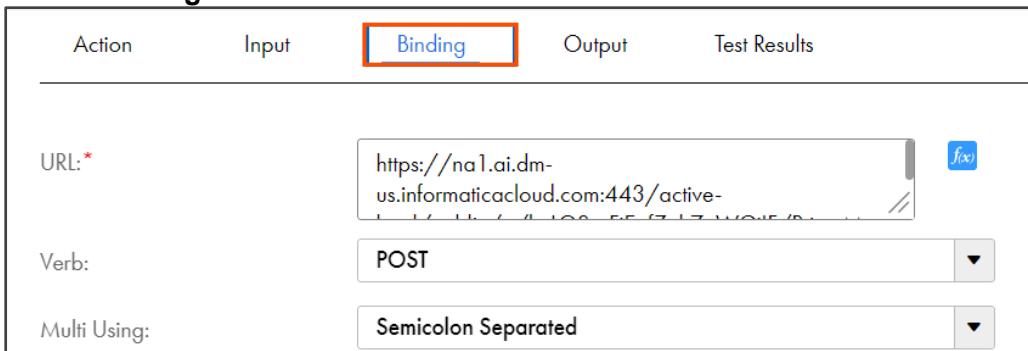
24. Make sure that the **Print\_MessageOperation** Action is selected.

**Note:** You can see only one action in the Action Name section.



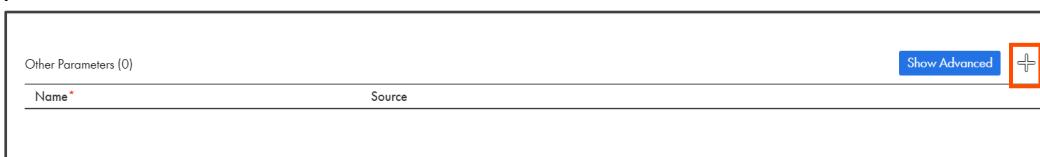
The screenshot shows the 'Actions' tab with a single row: 'Action Name: Print\_MessageOperation' and 'Action Type: Abstract'. The 'Actions' tab is highlighted with a red box.

25. Select **Binding**.



The screenshot shows the 'Binding' tab of the configuration interface. It includes fields for 'URL:' containing 'https://na1.ai.dmn.us.informaticacloud.com:443/active...', 'Verb:' set to 'POST', and 'Multi Using:' set to 'Semicolon Separated'.

26. Scroll down and from the Other Parameters section, click the '+' icon to add a parameter.



The screenshot shows the 'Other Parameters' section with a 'Name\*' field and a 'Source' field. A blue '+' icon with a red border is highlighted next to the 'Show Advanced' button.

27. In the Name field, enter **Input**.

| Other Parameters (1)               |  |
|------------------------------------|--|
| Name*                              | Source                                       |
| <input type="text" value="Input"/> | <input type="text" value="Hello World"/> ... |

28. In the Source field, click .

29. In the Expression field, enter **Hello World**.

| Expression                                 | Type: | Content |
|--|-------|---------|
| 1 <input type="text" value="Hello World"/> |       |         |

**Note:** In the Source field, make sure to copy and paste the text that you want to enter. Sometimes, it does not accept the entire text that you want to type.

30. Click **OK**.

**Note:** In the source field, make sure that entire text has entered.

| Action | Input | Binding                                  | Output | Test Results |
|--------|-------|--|--------|--------------|
|        |       |  |        |              |
| Name*  |       | Source                                   |        |              |
| Input  |       | <input type="text" value="Hello World"/> |        |              |

31. Click **Save**.

| S_XX_OpenAPI Print Message |             | Valid           | Save | Publish | <input checked="" type="checkbox"/> |
|----------------------------|-------------|-----------------|------|---------|-------------------------------------|
| Definition                 | Actions     | Process Objects |      |         |                                     |
| Actions (1)                |             |                 |      |         |                                     |
| Action Name                | Action Type | Description     |      |         |                                     |
| Print_MessageOperation     | Abstract    |                 |      |         |                                     |

## Test Action

32. Select **Test Results**.

33. From the Test Server drop-down, retain **Test on Cloud Server** and click **Test**.

| Action        | Input | Binding   | Output | Test Results                        |
|---------------|-------|---|--------|-------------------------------------|
|               |       |   |        |                                     |
| Test Server:* |       | <input type="text" value="Test on Cloud Server"/> |        | <input type="button" value="Test"/> |

34. In the Test Results tab, verify that the result of the test is **Successful**. Observe that the HTTPS Status is **200**.

| Action        | Input   | Binding | Output | <b>Test Results</b> |
|---------------|---|---------|--------|---------------------|
| Test Server:* | Test on Cloud Server  |         |        | <b>Test</b>         |
| Result:       | <input checked="" type="checkbox"/> Successful.   |         |        |                     |
| HTTP Status:  | 200   |         |        |                     |
| URL:          | <a href="https://na1.ai.dm-us.informaticacloud.com:443/active-bpel/public/rt/hnLOSoRiEqfZgh7aWGILF/Print_Message">https://na1.ai.dm-us.informaticacloud.com:443/active-bpel/public/rt/hnLOSoRiEqfZgh7aWGILF/Print_Message</a> |         |        |                     |

35. From the Output Fields section, scroll down to **Response Payload** field and observe the output.

| Action   | Input  | Binding | Output | <b>Test Results</b> |
|--|--|---------|--------|---------------------|
| print_MessageResponse  | xmlns:bconn="http://schemas.informatica.com/socrates/data-services/2014/05/business-connector-model.xsd"><Output>Hello World</Output></root> |         |        |                     |
| <b>▼ Response Payload:</b><br><pre>1 &lt;root&gt; 2 &lt;output&gt;Hello World&lt;/output&gt; 3 &lt;/root&gt;</pre> |  |         |        |                     |

36. Click **Publish**.

|                             |   |                 |                |
|-----------------------------|---|-----------------|----------------|
| S _XX_OpenAPI Print Message | <input checked="" type="checkbox"/> The asset [S _XX_OpenAPI Print Message] was published successfully. | Save            | <b>Publish</b> |
| Definition                  | <b>Actions</b>  | Process Objects |                |
| <b>▼ Actions (1)</b>        |   |                 |                |

A message “**The asset [S \_XX\_OpenAPI Print Message] was published successfully**” appears.

*This concludes the lab.*

## Module 6: Data Access Service Connectors

### Lab 6-1: Create a Data Access Service Connector Using Form

#### Overview:

You can create a Data Access Service Connector on a Secure Agent to access data directly from a database. The databases include IBM DB2, Microsoft SQL Server, MySQL, Oracle, and

You can create a Data Access Service Connector by defining the database properties and specifying variables to interact directly with the database. Using this service, you can create an invoke activity that executes SQL CRUD operations, such as select, insert, update, or delete on a specified data source and receives responses based on the columns in the database table, expression, result set, and attachments from the data source.

In this lab, you will create a Data Access Service Connector to fetch the employee records from the database.

#### Objectives:

- Create a Data Access Service Connector
- Define properties and actions for the Data Access Service Connector
- Test the Data Access Service Connector

#### Duration:

15 Minutes

---

#### Tasks

**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

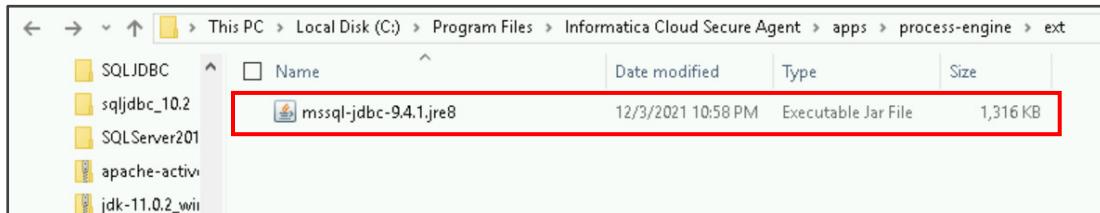
**Important:** Before creating a Data Access Service Connector, you must copy the required JDBC JAR files in the Secure Agent Installation directory and restart the Secure Agent.

#### Copy the JDBC JAR File

1. On the desktop, from the **CAI Lab Prep Files** folder, copy the **mssql-jdbc-9.4.1.jre8** file.

2. Paste the jar file in the following location:

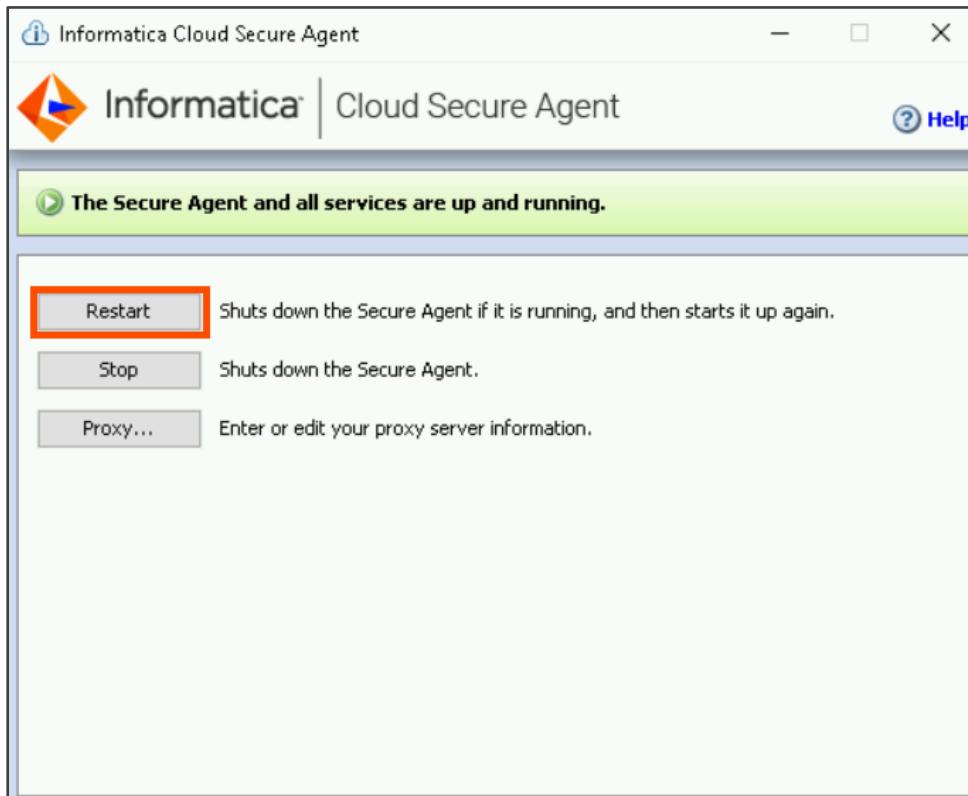
C:\Program Files\Informatica Cloud Secure Agent\apps\process-engine\ext



**Note:** C:\Program Files\Informatica Cloud Secure Agent\apps\process-engine\ext is the default Secure Agent installation directory. If you had installed the agent in a different directory, you must place the jar file in the appropriate agent folder.

After you paste the jar file in the Secure Agent installation directory, you must restart the Secure Agent and wait for it to come up and running.

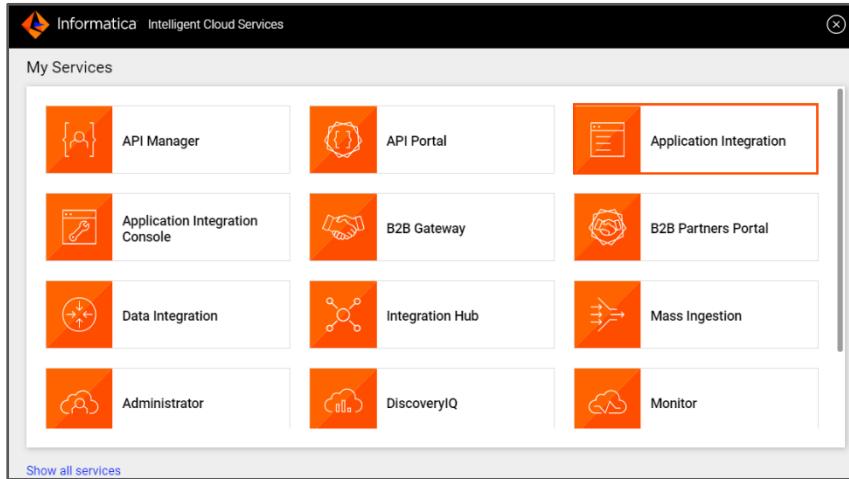
3. Open the Informatica Cloud Secure Agent window and click **Restart**.



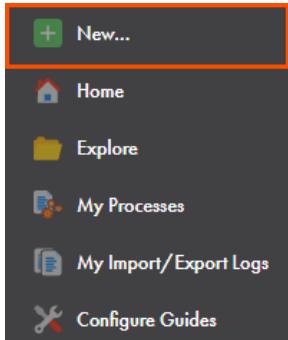
4. Wait for the agent to completely restart and you must see that all the services are up and running. It may take 5 to 10 mins for the agent to restart.

## Create a Data Access Service Connector

5. Log into IICS and access the **Application Integration** service.

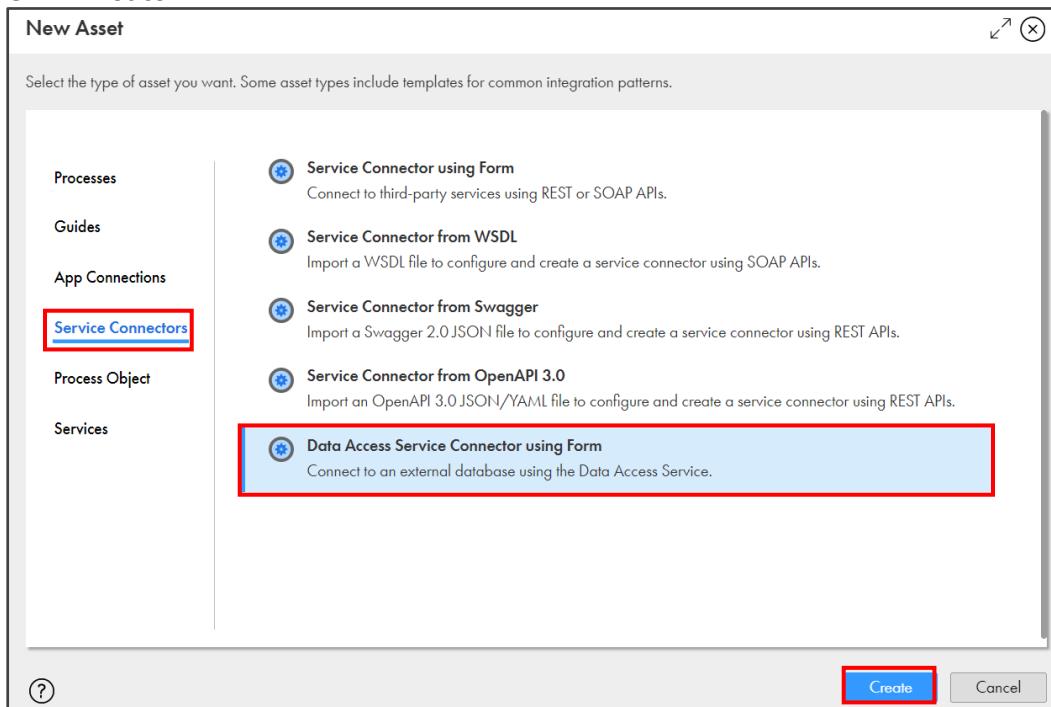


6. To create a new Service Connector, from the navigation pane, click **New**.



7. From the Service Connectors tab, select **Data Access Service Connector using Form**.

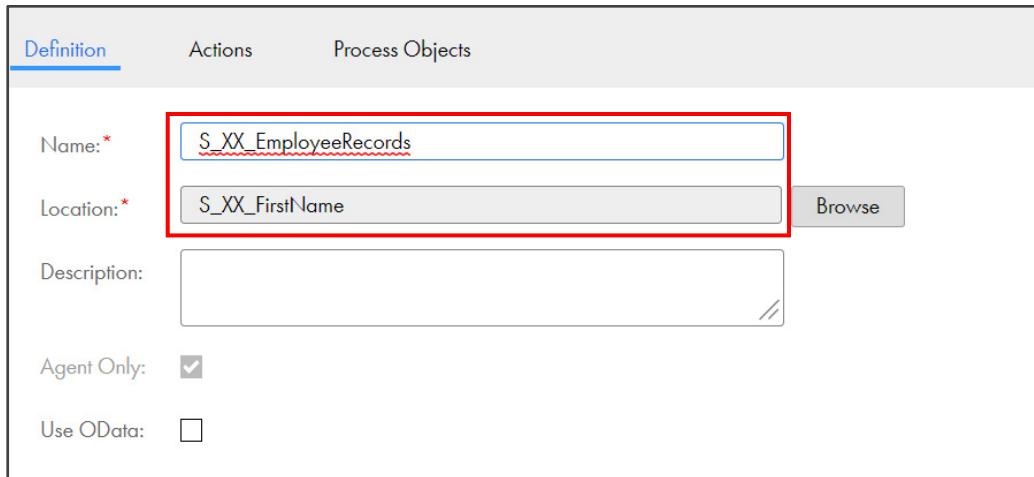
8. Click **Create**.



A new Service Connector page appears.

### Define Properties

9. In the service connector Name field, enter **S\_XX\_EmployeeRecords**.
10. In the Location field, select your project folder **S\_XX\_FirstName**.



| Definition   |   | Actions                               | Process Objects |
|--------------|---|---------------------------------------|-----------------|
| Name: *      | <input type="text" value="S_XX_EmployeeRecords"/> |                                       |                 |
| Location: *  | <input type="text" value="S_XX_FirstName"/>       | <input type="button" value="Browse"/> |                 |
| Description: | <input type="text"/>                              |                                       |                 |
| Agent Only:  | <input checked="" type="checkbox"/>               |                                       |                 |
| Use OData:   | <input type="checkbox"/>                          |                                       |                 |

11. In the JDBC Driver field, enter the following:  
**com.microsoft.sqlserver.jdbc.SQLServerDriver**
12. In the JDBC URL field, enter the following URL:  
**jdbc:sqlserver://localhost:1433;databaseName=CAI;default=master**

| Connection Properties (22) |   |   |                                     |
|----------------------------|---|---|-------------------------------------|
| Name                       | Description   | Test With   | Required                            |
| <b>General</b>             |   |   |                                     |
| JDBC Driver: *             | The fully qualified Java class name of the JDBC driver. | com.microsoft.sqlserver.jdbc.SQLServerDriver                    | <input checked="" type="checkbox"/> |
| JDBC Connection URL: *     | The JDBC connection URL to connect to the database.     | jdbc:sqlserver://localhost:1433;databaseName=CAI;default=master | <input checked="" type="checkbox"/> |

13. In the Username and Password fields, enter **CAI**.

|              |  |       |                                     |                                     |
|--------------|--|-------|-------------------------------------|-------------------------------------|
| User Name: * | The user name to log in to the database. | CAI   | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| Password: *  | The password to log in to the database.  | • • • | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

14. Click **Save**.

A message **The service connector was saved successfully** appears.

S\_XX\_EmployeeRecords
 Valid
 The service connector was saved successfully. X
Save
Publish

| Definition   | Actions | Process Objects |
|--|---------|-----------------|
| Name: * <input type="text" value="S_XX_EmployeeRecords"/><br>Location: * <input type="text" value="S_XX_FirstName"/> <span>Browse</span><br>Description: <input type="text" value=""/><br>Agent Only: <input type="checkbox"/> |         |                 |

## Define Actions

15. Click the **Actions** tab.
16. Click  to add a new action.

Definition
Actions
Process Objects

| ▼ Actions (0) <div style="float: right;"></div> |             |             |
|--|-------------|-------------|
| Action Name  | Action Type | Description |
| Nothing to display.  |             |             |

17. In the Action Name field, enter **Select Employee Records**.

| Action         | Input                   | SQL Binding | Output | Test Results |
|----------------|-------------------------|-------------|--------|--------------|
| Action Name: * | Select Employee Records |             |        |              |
| Category:      |                         |             |        |              |

18. Click the **Input** tab and click the '+' icon to add the input fields.

| Action           | <b>Input</b> | SQL Binding | Output   | Test Results |
|------------------|--------------|-------------|----------|--------------|
| Input Fields (0) |              |             |          |              |
| Name *           | Label        | Type        | Required | Description  |
| Parameter        |              |             |          | Test with    |
| <b>+</b>         |              |             |          |              |

19. In the **Input Fields** section, enter the following values:

| Property  | Value    |
|-----------|----------|
| Name      | EmpName  |
| Label     | EmpName  |
| Type      | Text     |
| Test with | Harpreet |

| Action   | <b>Input</b> | SQL Binding | Output                                  | Test Results |                                     |         |      |   |          |                                     |
|--|--------------|-------------|---|--------------|-------------------------------------|---------|------|---|----------|-------------------------------------|
|  |              |             |   |              |                                     |         |      |   |          |                                     |
| <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;">EmpName</td> <td style="padding: 5px;">EmpName</td> <td style="padding: 5px; text-align: center;">Text</td> <td style="padding: 5px;"><input type="checkbox"/> This is a list</td> <td style="padding: 5px;">Harpreet</td> <td style="padding: 5px;"><input checked="" type="checkbox"/></td> </tr> </table> |              |             |   |              | EmpName                             | EmpName | Text | <input type="checkbox"/> This is a list | Harpreet | <input checked="" type="checkbox"/> |
| EmpName  | EmpName      | Text        | <input type="checkbox"/> This is a list | Harpreet     | <input checked="" type="checkbox"/> |         |      |   |          |                                     |

20. Click the **SQL Binding** tab and in the SQL field, enter the following query:

**select \* from dbo.employee where emp\_fname='{\$EmpName}'**

**OR**

From the CAI Lab Prep Files folder, open the notepad file named **08\_LabGuide\_CAI\_Create Data Access Service Connector Using Form\_6-1**. Copy the command mentioned in the notepad and paste it in the SQL field.

| Action | Input | SQL Binding  | Output | Test Results |
|--------|-------|--|--------|--------------|
|        |       | select * from dbo.employee where emp_fname='{\$EmpName}' |        |              |

21. Click the **Output** tab and click the '+' icon to add the output fields.
  22. In the **Output Fields** section, enter the following values:

| Name                | Type | Get From | Value in Get From |
|---------------------|------|----------|-------------------|
| Employee_Number     | Text | Column   | emp_no            |
| Employee_FirstName  | Text | Column   | emp_fname         |
| Employee_LastName   | Text | Column   | emp_lname         |
| Employee_Department | Text | Column   | dept_no           |

| Action             | Input   | SQL Binding | <u>Output</u>       | Test Results |
|--------------------|---|-------------|---------------------|--------------|
| Name*              | Type  | Description | Get From            |              |
| Employee_Number    | Text<br><input type="checkbox"/> This is a list |             | Column<br>emp_no    |              |
| Employee_FirstName | Text<br><input type="checkbox"/> This is a list |             | Column<br>emp_fname |              |

| Action              | Input   | SQL Binding | <u>Output</u>       | Test Results |
|---------------------|---|-------------|---------------------|--------------|
| Employee_LastName   | Text<br><input type="checkbox"/> This is a list |             | Column<br>emp_lname |              |
| Employee_Department | Text<br><input type="checkbox"/> This is a list |             | Column<br>dept_no   |              |

- 23. Click Save.**

## Test Action

24. Select **Test Results**.

25. From the Test Server drop-down, select your Secure Agent group and click **Test**.

| Action        | Input            | SQL Binding | Output | Test Results |
|---------------|------------------|-------------|--------|--------------|
| Test Server:* | CAI-XX-FIRSTNAME |             |        | <b>Test</b>  |

26. In the Output Fields section, you should see the following values:

| Output              | Value    |
|---------------------|----------|
| Employee_Number     | 1        |
| Employee_FirstName  | Harpreet |
| Employee_LastName   | Bhatia   |
| Employee_Department | 1234     |

| Action              | Input | SQL Binding | Output | Test Results |
|---------------------|-------|-------------|--------|--------------|
| <b>Output</b>       |       |             |        |              |
| Employee_Number     |       |             |        | Value        |
|                     |       |             |        | 1            |
| Employee_FirstName  |       |             |        | Harpreet     |
| Employee_LastName   |       |             |        | Bhatia       |
| Employee_Department |       |             |        | 1234         |

27. From the Output Fields section, scroll down to **Response Payload** field and expand it to observe the output.

▼ Response Payload

```

1 <dataAccessResponse statementId="a0175759-34f9-447c-9d42-dd4a9e746b7e">
2   <row>
3     <id>1</id>
4     <emp_no>1</emp_no>
5     <emp_fname>Harpreet</emp_fname>
6     <emp_lname>Bhatia</emp_lname>
7     <dept_no>1234</dept_no>

```

## 28. Click Publish.



A message “**The asset was published successfully**” appears.

---

*This concludes the lab.*

# Module 7: App Connections

## Lab 7-1: Create an App Connection Using Service Connector

### Overview:

An App Connection defines a link to a data source, a database, or a Service Connector. In CAI, an App Connection can use any published out-of-the-box connector or Service Connector.

This is **part 2** of the set of three labs. In this lab, you will create an app connection that utilizes the service connector created in the **Lab 5-1: Create a Service Connector** exercise.

### Objectives:

- Create an App Connection using Service Connector
- Publish the App Connection

### Duration:

10 Minutes

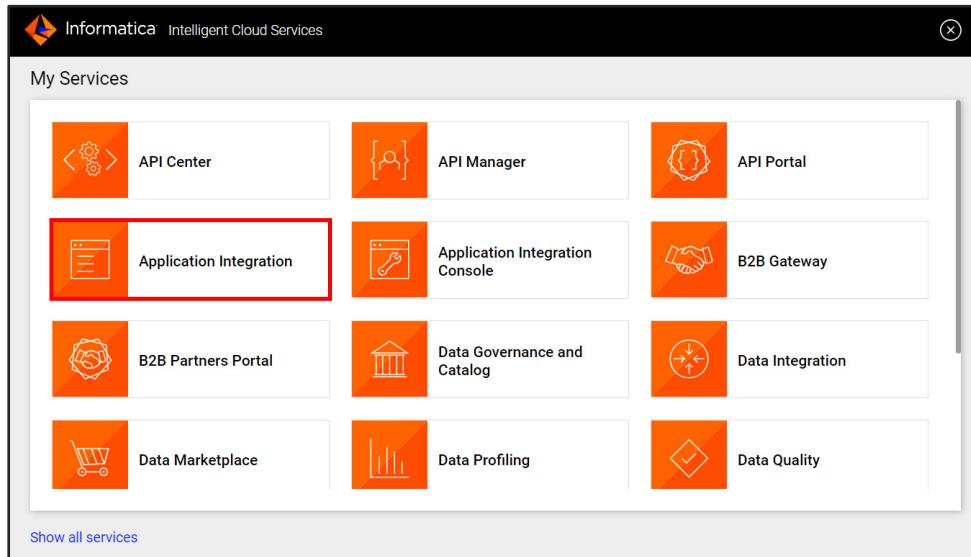
---

### Tasks

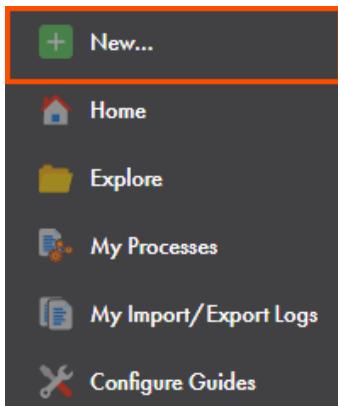
**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

#### Create a connection:

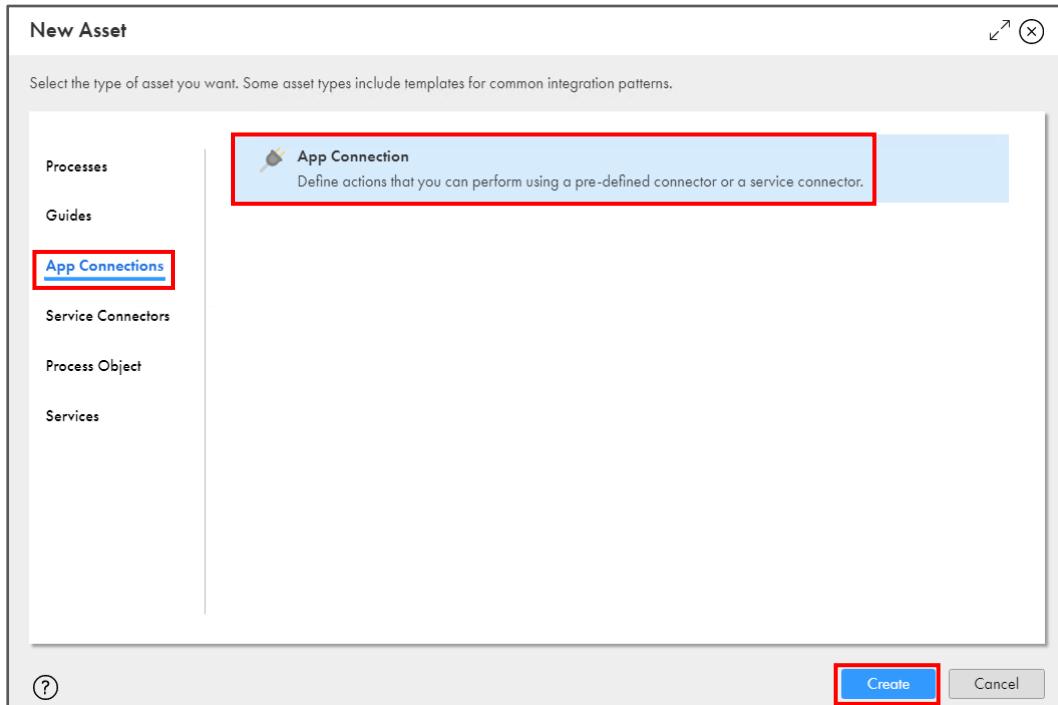
1. Log into IICS and access the **Application Integration** service.



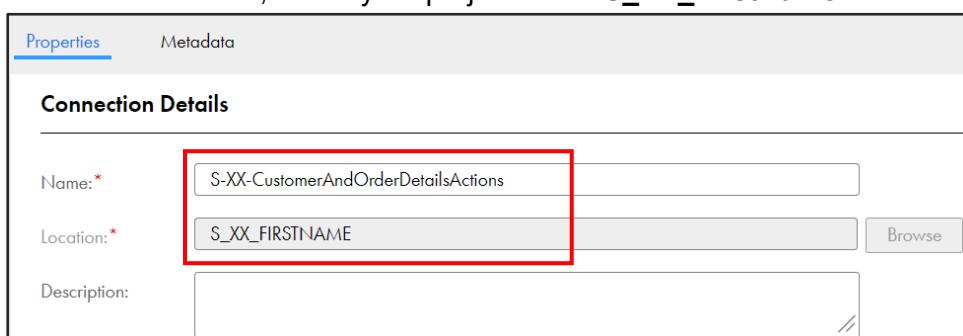
2. From the Navigation pane, click **New**.



3. In the New Asset window, from the App Connections tab, select **App Connection**, and click **Create**.



4. In the Connection Details page, enter the connection name as **S-XX-CustomerAndOrderDetailsActions**.  
 5. In the Location field, select your project folder **S\_XX\_FirstName**.



| Properties                |                                     | Metadata |
|---------------------------|-------------------------------------|----------|
| <b>Connection Details</b> |                                     |          |
| Name: *                   | S-XX-CustomerAndOrderDetailsActions |          |
| Location: *               | S_XX_FIRSTNAME                      |          |
| Description:              |                                     |          |

6. From the Type drop-down, expand your project folder, **S\_XX\_FirstName**, and select **S\_XX\_ThomasBayer\_OrderInfo**.
7. From the Run On drop-down, retain the **Cloud Server or any Secure Agent** option.

**Connection Details**

|                  |  |
|------------------|--|
| Name:*           | S-XX-CustomerAndOrderDetailsActions  |
| Location:*       | S_XX_FIRSTNAME <input type="button" value="Browse"/>                                   |
| Description:     | <input type="text"/>   |
| Type:*           | <b>S_XX_FIRSTNAME &gt; S_XX_ThomasBayer_OrderInfo</b> <input type="button" value="▼"/> |
| Run On:*         | <b>Cloud Server or any Secure Agent</b> <input type="button" value="▼"/>               |
| Connection Test: | Not Supported  |
| OData-Enabled:   | Not Supported  |

8. Click **Save**.

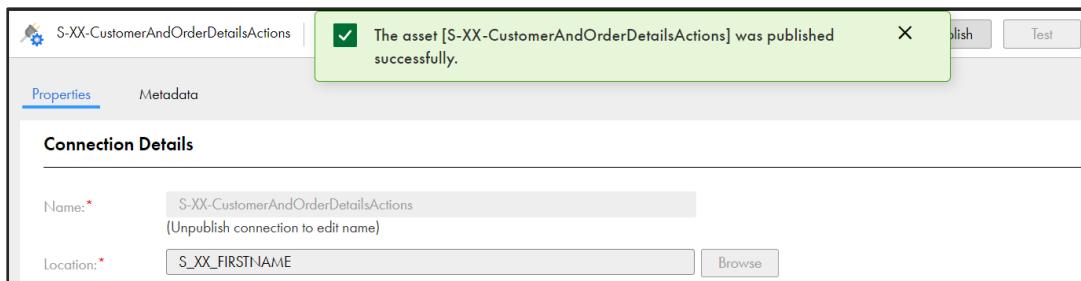
S-XX-CustomerAndOrderDetailsAction |  Valid

|                            |  |
|----------------------------|--|
| <a href="#">Properties</a> | <a href="#">Metadata</a>                             |
| <b>Connection Details</b>  |  |
| Name:*                     | S-XX-CustomerAndOrderDetailsActions                  |
| Location:*                 | S_XX_FIRSTNAME <input type="button" value="Browse"/> |
| Description:               | <input type="text"/>                                 |

9. Click **Publish**.

S-XX-CustomerAndOrderDetailsActions |  Valid  The connection was saved successfully

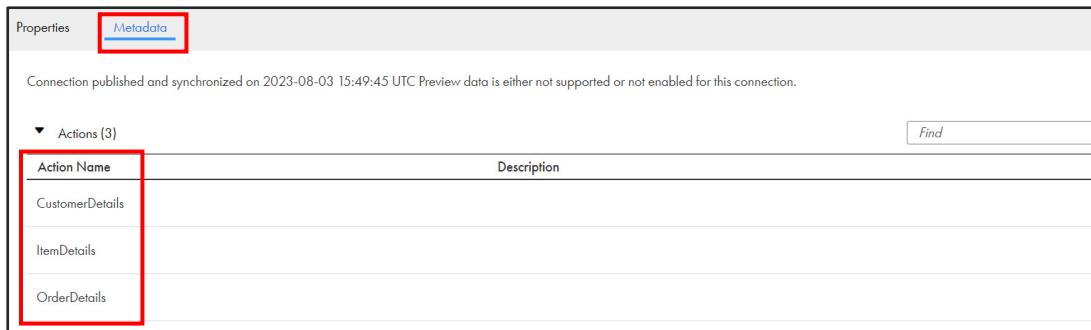
|                            |  |
|----------------------------|--|
| <a href="#">Properties</a> | <a href="#">Metadata</a>                             |
| <b>Connection Details</b>  |  |
| Name:*                     | S-XX-CustomerAndOrderDetailsActions                  |
| Location:*                 | S_XX_FIRSTNAME <input type="button" value="Browse"/> |
| Description:               | <input type="text"/>                                 |



The asset [S-XX-CustomerAndOrderDetailsActions] was published successfully.

You get a notification that “**The asset S-XX-CustomerAndOrderDetailsActions was published successfully.**”

10. In the Metadata tab, observe that the Actions from the Service Connector is added to the Actions section.



Connection published and synchronized on 2023-08-03 15:49:45 UTC Preview data is either not supported or not enabled for this connection.

▼ Actions (3)

| Action Name     | Description |
|-----------------|-------------|
| CustomerDetails |             |
| ItemDetails     |             |
| OrderDetails    |             |

**Note:** If you can't see the Actions in the metadata tab, refresh your browser. If you still can't see the Actions, then unpublish the connection and publish it once again.

*This concludes the lab.*

# Module 7: App Connections

## Lab 7-2: Create a JDBC Connection

### Overview:

Informatica CAI allows you to connect to the databases with the help of the JDBC connector. The databases include IBM DB2, Microsoft SQL Server, MySQL, Oracle, and PostgreSQL.

In this lab, you will create a connection using the built-in JDBC Connector and use that connection in the ExchangeRate process to fetch data from the database table and perform conversion based in the user input values.

**Note:** This lab is the **part 2** of a set of three labs. Earlier, you created process objects (Lab 4-1: Create Process Objects). This lab is continued in **Lab 10-1: Invoke a Synchronous Web Service Call** of Module 10.

### Objectives:

- Create a JDBC connection
- Publish the connection

### Duration:

15 Minutes

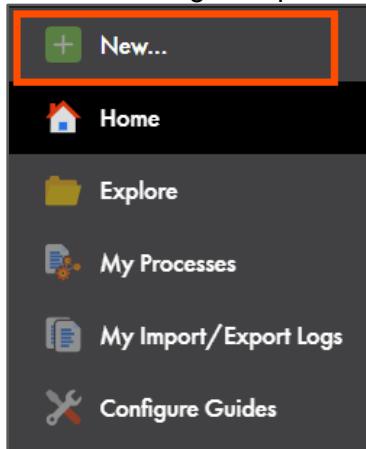
---

### Tasks:

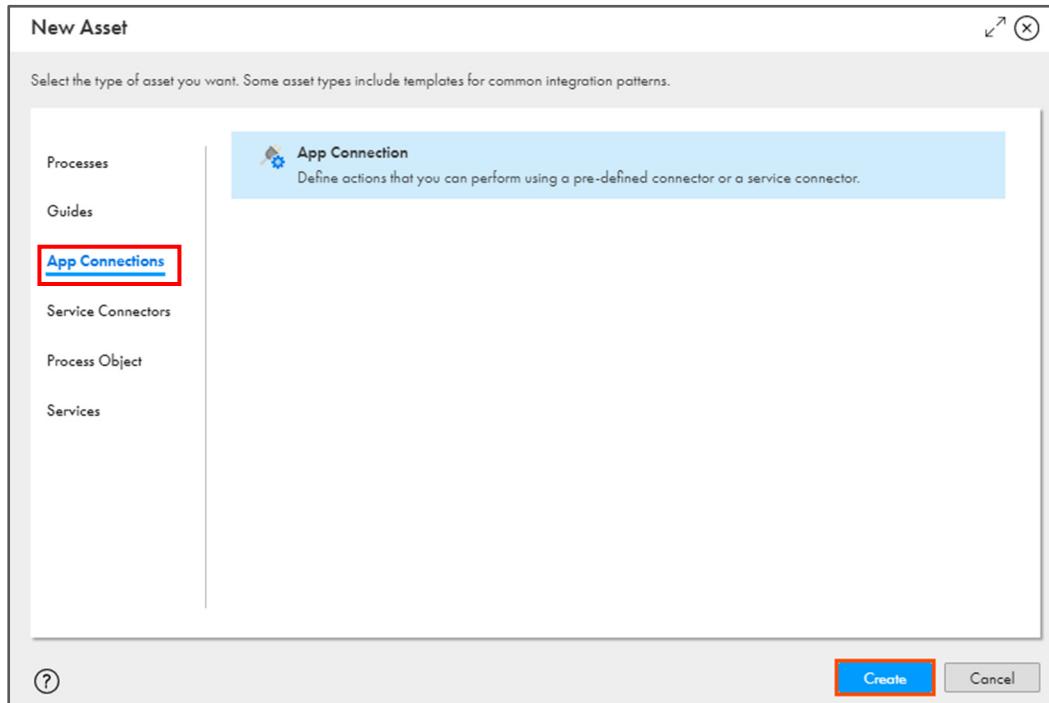
**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

### Create a Connection

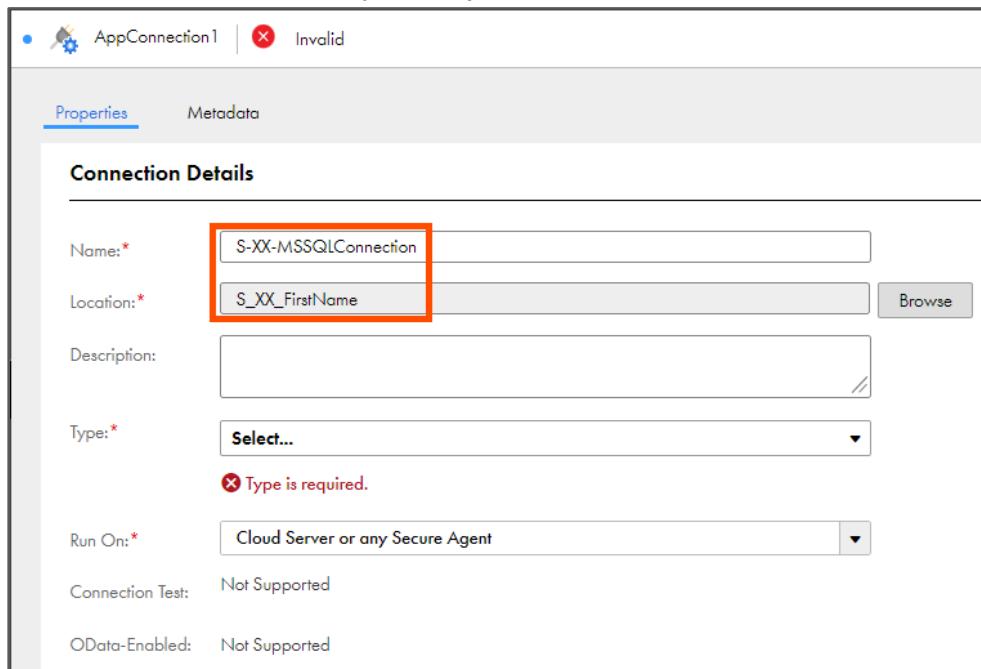
1. Log in to IICS and access the **Application Integration** service.
2. From the Navigation pane, click **New**.



3. In the New Asset window, from the App Connections tab, select **App Connection**, and click **Create**.



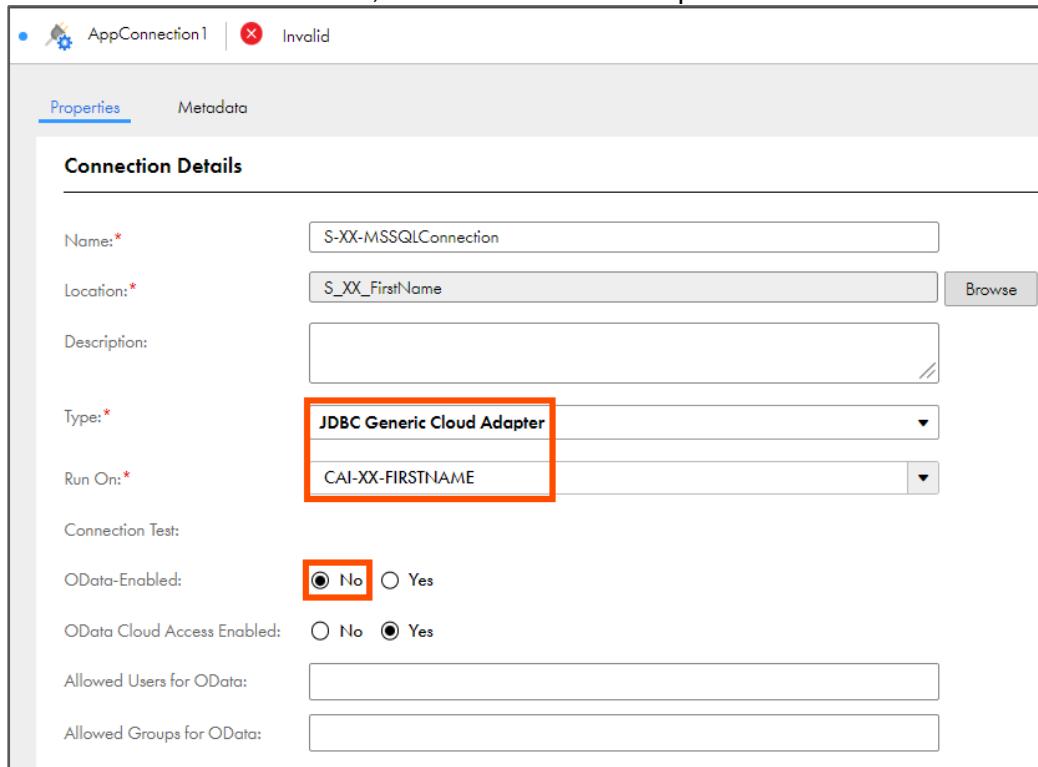
4. In the Connection Details page, enter the connection name as **S-XX-MSSQLConnection**.  
 5. In the Location field, select your project folder **S\_XX\_FirstName**.



The screenshot shows the 'Connection Details' page of the Informatica interface. It has tabs for 'Properties' and 'Metadata'. Under 'Properties', there are several fields:  
 - 'Name:' is set to 'S-XX-MSSQLConnection' (highlighted with a red box).  
 - 'Location:' is set to 'S\_XX\_FirstName' (highlighted with a red box).  
 - 'Description:' is empty.  
 - 'Type:' has a dropdown menu with 'Select...' and a note below it: 'Type is required.'  
 - 'Run On:' is set to 'Cloud Server or any Secure Agent'.  
 - 'Connection Test:' is listed as 'Not Supported'.  
 - 'OData-Enabled:' is also listed as 'Not Supported'.  
 The 'Properties' tab is currently active, indicated by a blue underline.

6. From the Type drop-down, select **JDBC Generic Cloud Adapter**.  
 7. From the Run On drop-down, select your Secure Agent.

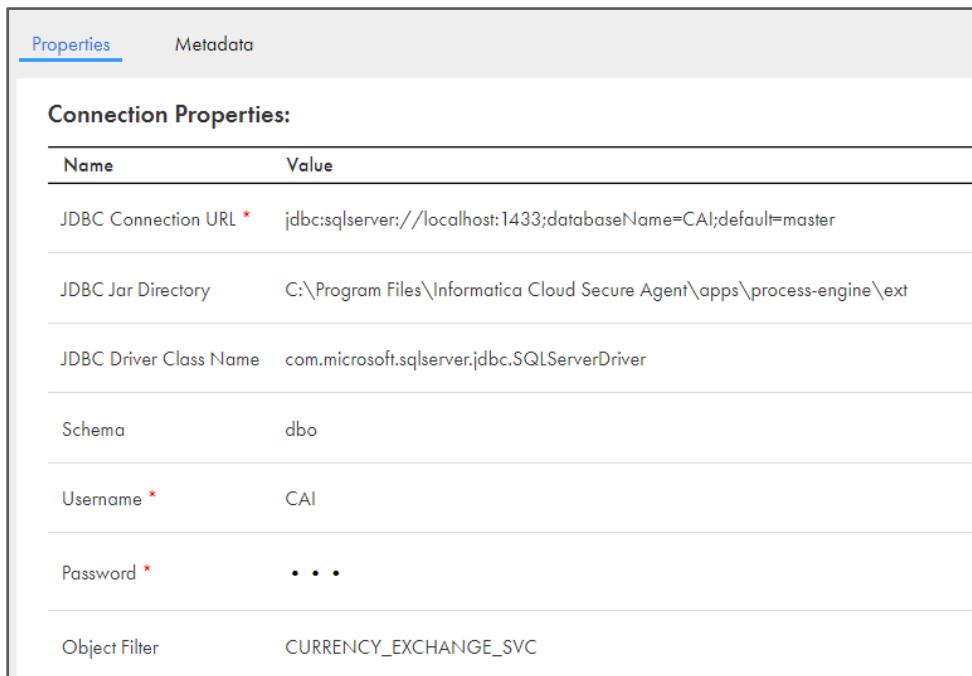
8. For the OData-Enabled field, retain the default **No** option.



The screenshot shows the 'Properties' tab of an app connection named 'AppConnection1'. The 'Connection Details' section is displayed. The 'Type:' dropdown is set to 'JDBC Generic Cloud Adapter', which is highlighted with a red box. The 'Run On:' dropdown contains 'CAI-XX-FIRSTNAME', also highlighted with a red box. Under 'OData-Enabled', the radio button for 'No' is selected, also highlighted with a red box. The 'Metadata' tab is visible at the top but is not active.

9. To configure JDBC connection properties, click in the JDBC Connection URL field and enter **jdbc:sqlserver://localhost:1433;databaseName=CAI;default=master**
10. In the JDBC Jar Directory field, enter  
**C:\Program Files\Informatica Cloud Secure Agent\apps\process-engine\ext**
11. In the JDBC Driver Class Name field, enter  
**com.microsoft.sqlserver.jdbc.SQLServerDriver**
12. In the Schema field, enter **dbo**.
13. Enter **CAI** in the Username and Password fields.

14. In the Object Filter field, enter **CURRENCY\_EXCHANGE\_SVC**.



The screenshot shows the 'Properties' tab of the Connection Properties dialog. The 'Object Filter' field is highlighted and contains the value 'CURRENCY\_EXCHANGE\_SVC'.

| Name                   | Value   |
|------------------------|---|
| JDBC Connection URL *  | jdbc:sqlserver://localhost:1433;databaseName=CAI;default=master         |
| JDBC Jar Directory     | C:\Program Files\Informatica Cloud Secure Agent\apps\process-engine\ext |
| JDBC Driver Class Name | com.microsoft.sqlserver.jdbc.SQLServerDriver                            |
| Schema                 | dbo   |
| Username *             | CAI   |
| Password *             | • • •   |
| Object Filter          | CURRENCY_EXCHANGE_SVC   |

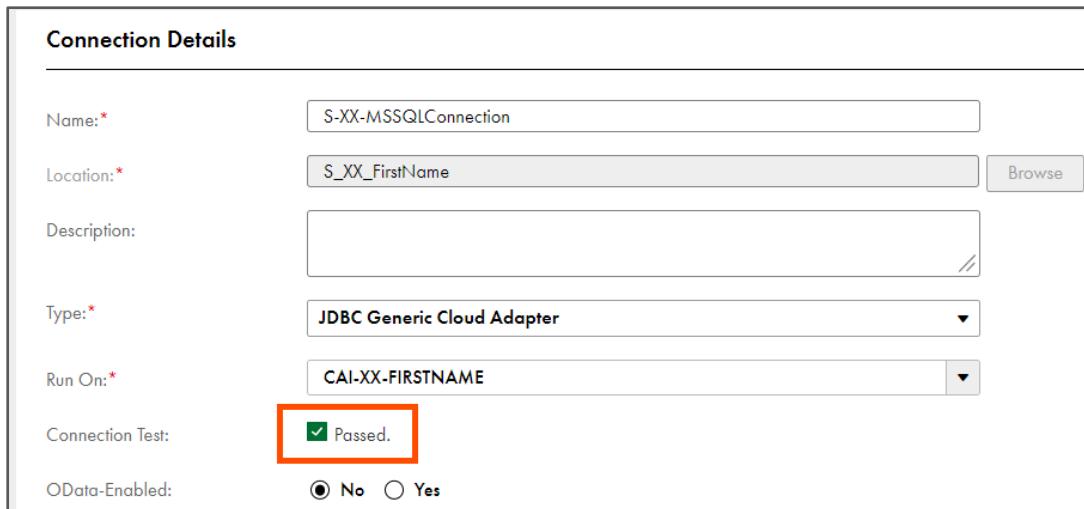
15. **Save** the connection.

16. **Test** the connection.



The screenshot shows the connection test results. A green message box indicates 'The connection was saved successfully'. The 'Test' button is highlighted with a red box.

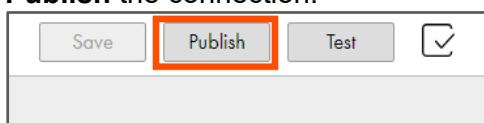
The Connection Test field mentions **Passed**.



The screenshot shows the 'Connection Details' dialog. The 'Connection Test' field is highlighted with a red box and contains the value 'Passed.'

| Connection Details |  |
|--------------------|--|
| Name:*             | S-XX-MSSQLConnection                               |
| Location:*         | S_XX_FirstName                                     |
| Description:       |  |
| Type:*             | JDBC Generic Cloud Adapter                         |
| Run On:*           | CAI-XX-FIRSTNAME                                   |
| Connection Test:   | <input checked="" type="checkbox"/> Passed.        |
| OData-Enabled:     | <input type="radio"/> No <input type="radio"/> Yes |

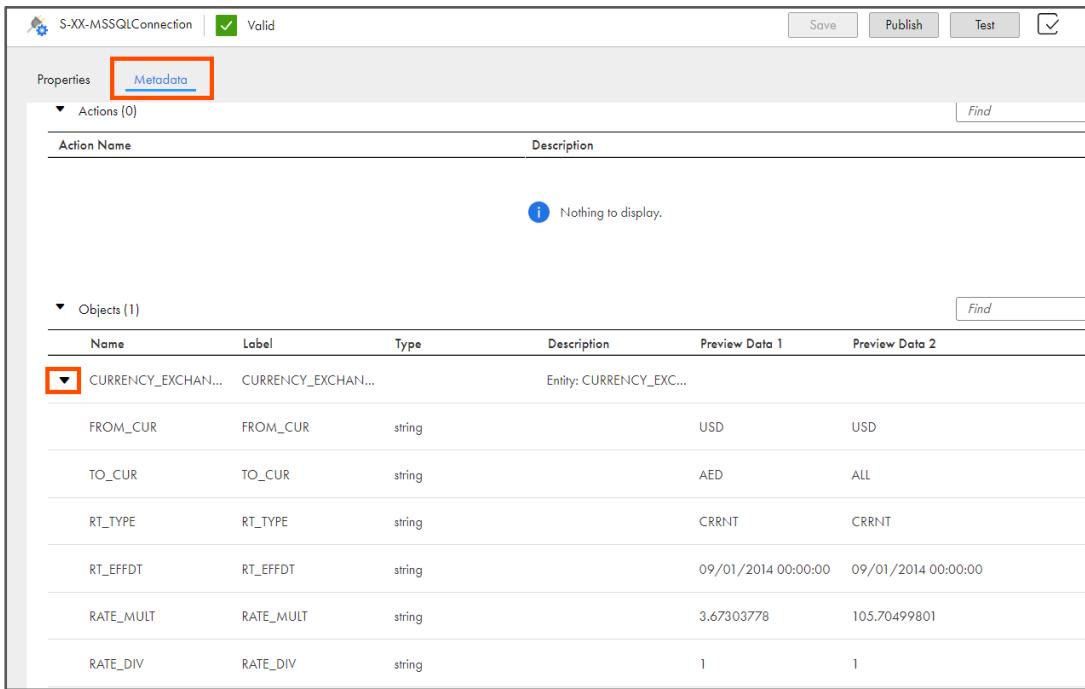
17. **Publish** the connection.



The screenshot shows the 'Publish' dialog. The 'Publish' button is highlighted with a red box.

You get a notification that '**The asset [S-XX-MSSQLConnection] was published successfully**'.

18. In the Metadata tab, observe that the Objects section displays the data preview for the **CURRENCY\_EXCHANGE\_SVC** table.



| Name               | Label              | Type   | Description             | Preview Data 1      | Preview Data 2 |
|--------------------|--------------------|--------|-------------------------|---------------------|----------------|
| CURRENCY_EXCHAN... | CURRENCY_EXCHAN... |        | Entity: CURRENCY_EXC... |                     |                |
| FROM_CUR           | FROM_CUR           | string | USD                     | USD                 |                |
| TO_CUR             | TO_CUR             | string | AED                     | ALL                 |                |
| RT_TYPE            | RT_TYPE            | string | CRRNT                   | CRRNT               |                |
| RT_EFFDT           | RT_EFFDT           | string | 09/01/2014 00:00:00     | 09/01/2014 00:00:00 |                |
| RATE_MULT          | RATE_MULT          | string | 3.67303778              | 105.70499801        |                |
| RATE_DIV           | RATE_DIV           | string | 1                       | 1                   |                |

**Note:** Refresh the page if you can't see the data preview.

You have successfully established a connection between Informatica CAI and the database.

*This concludes the lab.*

## Module 7: App Connections

### Lab 7-3: Create a Customer and Order Details Process

#### Overview:

This is **part 3** of the set of three labs. In Lab 5-1: Create a Service Connector, you created a service connector from a publicly available REST Web Service. You also created an app connection (Lab 7-1: Create an App Connection using Service Connector) using the same service connector.

In this lab, you will create a process to obtain the customer and order details through the Order id, using the third-party OData service.

#### Objectives:

- Create a new Process
- Add and configure service steps
- Publish and execute the Process

#### Duration:

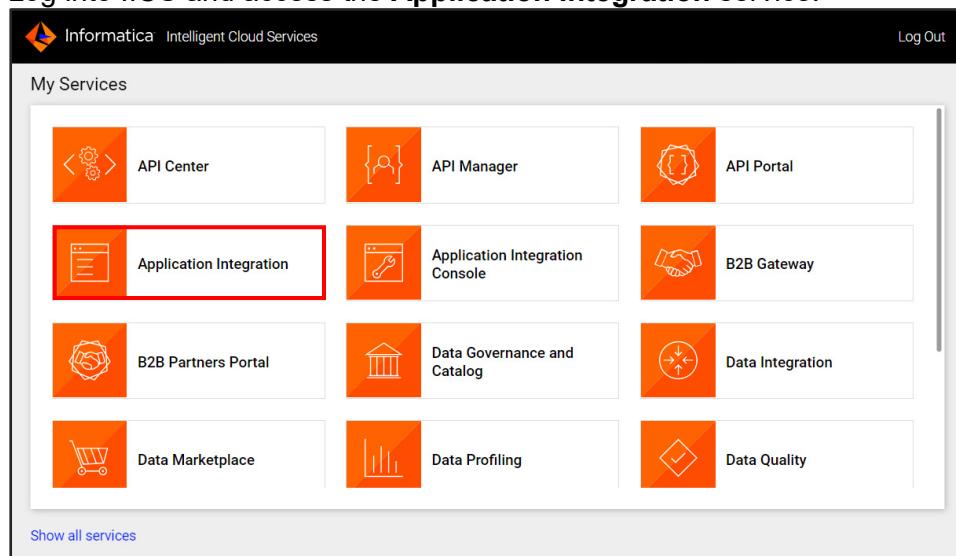
30 Minutes

---

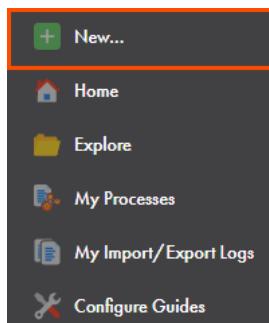
## Tasks

### Create a new process:

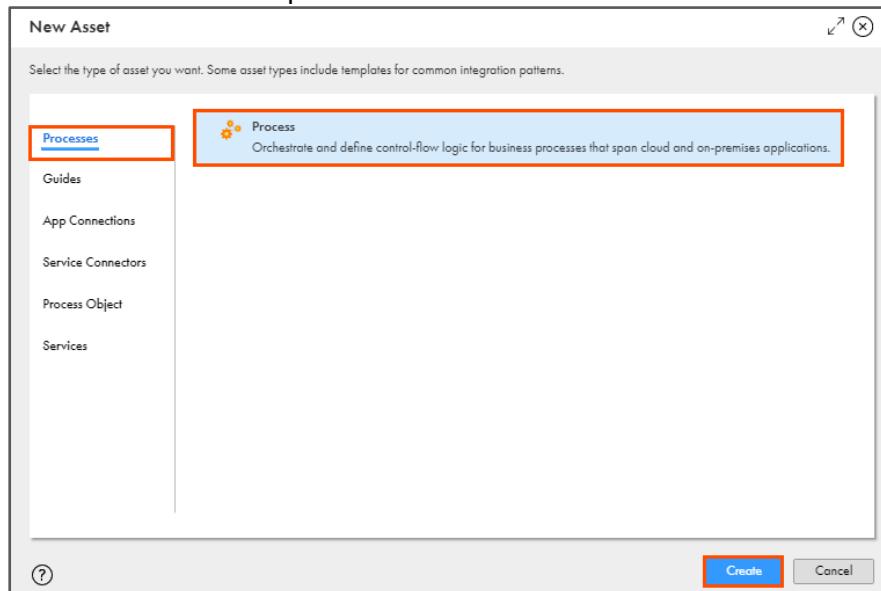
1. Log into IICS and access the **Application Integration** service.



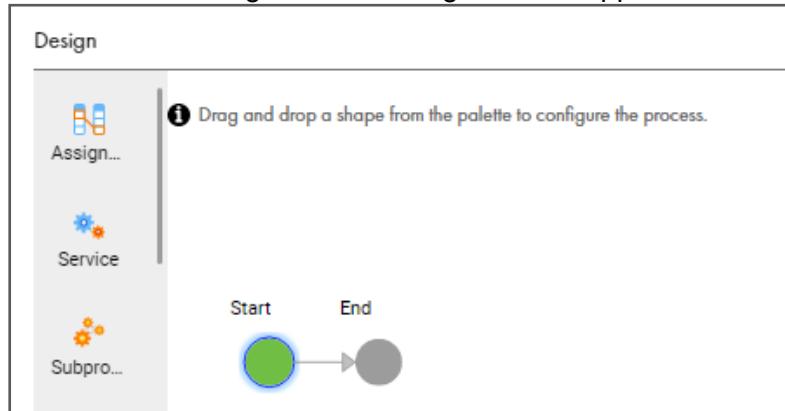
2. To create a new process, from the navigation pane, select **New**.



3. Retain the **Process** option and click **Create**.

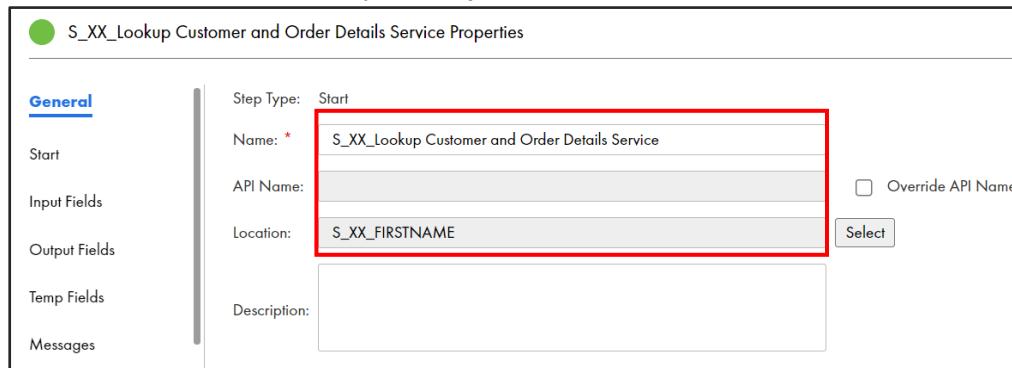


**Note:** The following Process Design canvas appears with the Start and End steps.



4. From the process canvas, select the **Start** step.  
 5. In the General tab, enter the name as **S\_XX\_Lookup Customer and Order Details Service**.  
**Note:** Replace XX with your student number.  
 6. Skip the API Name option.

7. In the Location field, select your project folder **S\_XX\_FirstName**.



S\_XX\_Lookup Customer and Order Details Service Properties

**General**

Step Type: Start  
 Name: \* S\_XX\_Lookup Customer and Order Details Service  
 API Name:  
 Location: S\_XX\_FIRSTNAME  
 Override API Name

Start  
 Input Fields  
 Output Fields  
 Temp Fields  
 Messages

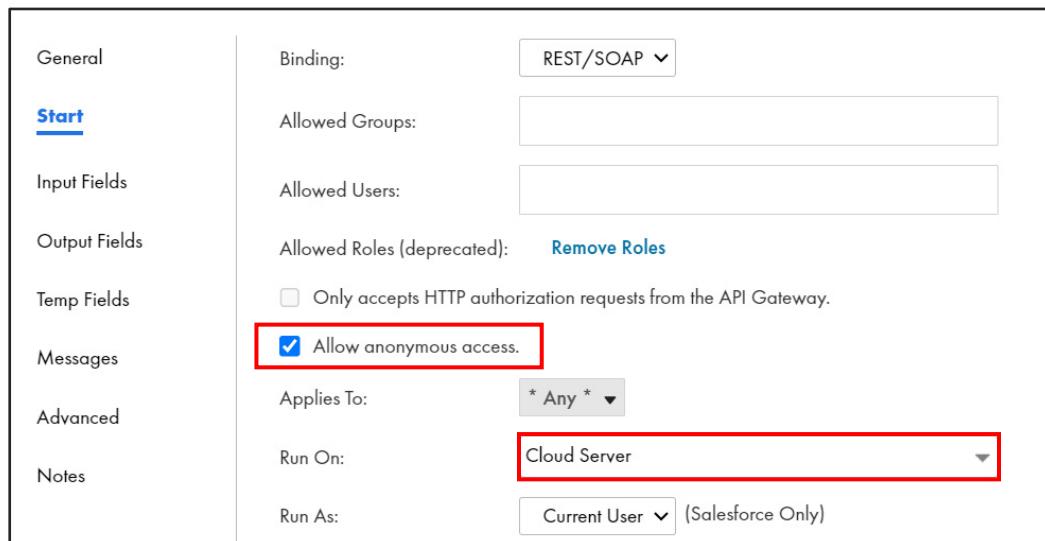
Description:

8. From the properties pane, select **Start**.

9. Select the **Allow anonymous access** option.

10. From the Run On drop-down, retain **Cloud Server**.

**Note:** You can select either Cloud Server or your secure agent. In this case, you are selecting Cloud Server as the previous service connector was tested and published on cloud.

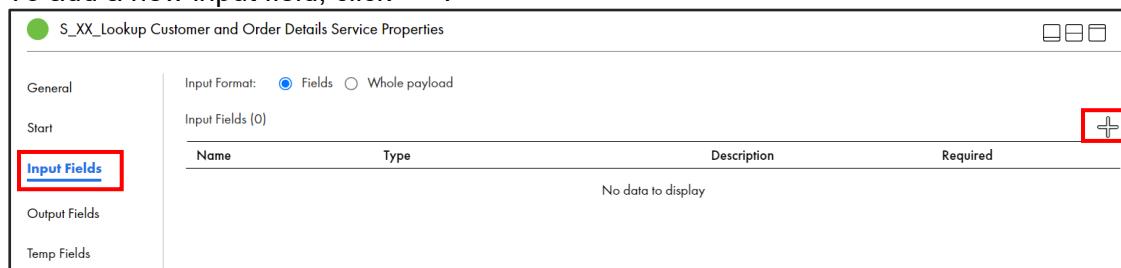


|               |   |
|---------------|---|
| General       | Binding: REST/SOAP  |
| <b>Start</b>  | Allowed Groups:<br>Allowed Users:<br>Allowed Roles (deprecated): <a href="#">Remove Roles</a> |
| Input Fields  | <input type="checkbox"/> Only accepts HTTP authorization requests from the API Gateway.       |
| Output Fields | <input checked="" type="checkbox"/> Allow anonymous access.                                   |
| Temp Fields   |   |
| Messages      | Applies To: * Any *   |
| Advanced      | Run On: Cloud Server  |
| Notes         | Run As: Current User (Salesforce Only)  |

As this process involves accepting an Order id from the user and displaying the relevant details for that order, you need to create an input field that will accept the Order id from the user.

11. From the properties pane, select **Input Fields**.

12. To add a new input field, click .



S\_XX\_Lookup Customer and Order Details Service Properties

General      Input Format:  Fields  Whole payload

Start      Input Fields (0)

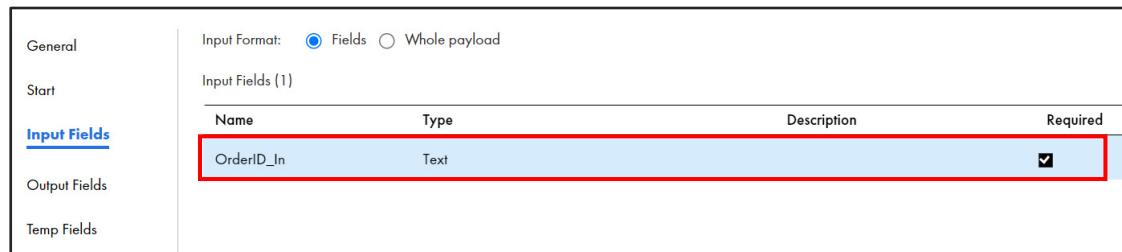
| Name               | Type | Description | Required |
|--------------------|------|-------------|----------|
| No data to display |      |             |          |

**Input Fields**

Output Fields  
Temp Fields

13. Create the input field, as shown in the table below:

| Name       | Type | Required |
|------------|------|----------|
| OrderID_In | Text | Yes      |

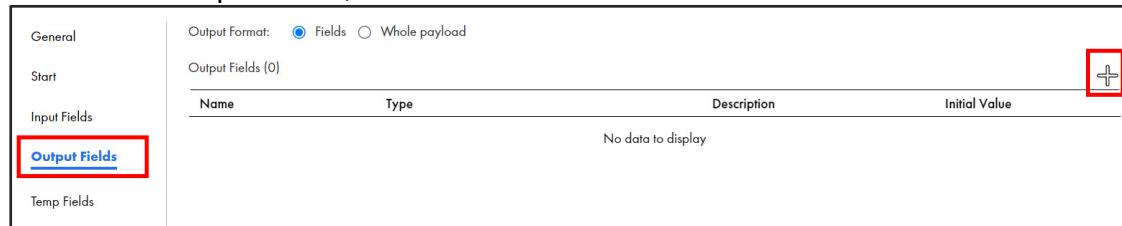


The screenshot shows the Informatica properties pane with the 'Input Fields' tab selected. Under 'Input Fields (1)', there is a table with columns: Name, Type, Description, and Required. The row for 'OrderID\_In' has 'Text' in the Type column and a checked checkbox in the Required column, both highlighted with a red box.

Next, you will create the output fields that will contain the details of the specified order.

14. From the properties pane, select **Output Fields**.

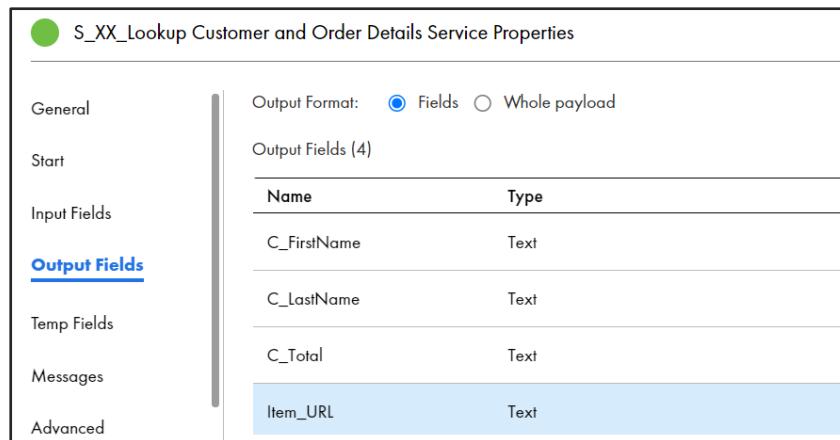
15. To add new Output Fields, click .



The screenshot shows the Informatica properties pane with the 'Output Fields' tab selected. Under 'Output Fields (0)', there is a table with columns: Name, Type, Description, and Initial Value. The 'Initial Value' column header contains a red box and an add button icon.

16. Create the Output Fields, as shown in the table below:

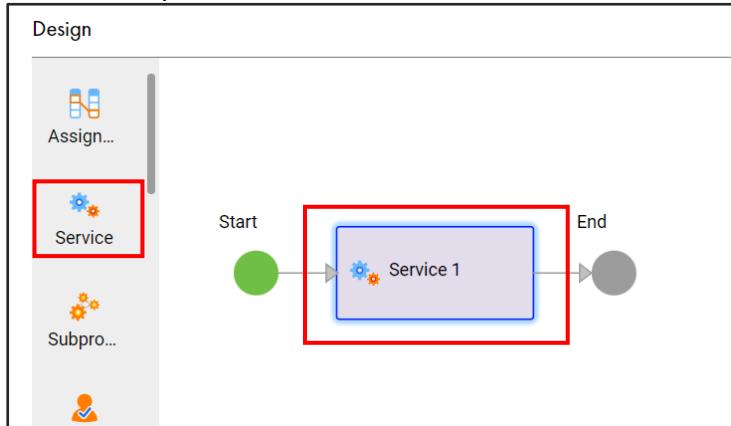
| Name        | Type |
|-------------|------|
| C_FirstName | Text |
| C_LastName  | Text |
| C_Total     | Text |
| Item_URL    | Text |



The screenshot shows the Informatica service properties pane with the 'Output Fields' tab selected. Under 'Output Fields (4)', there is a table with columns: Name and Type. The rows for 'C\_FirstName', 'C\_LastName', 'C\_Total', and 'Item\_URL' are all highlighted with a red box.

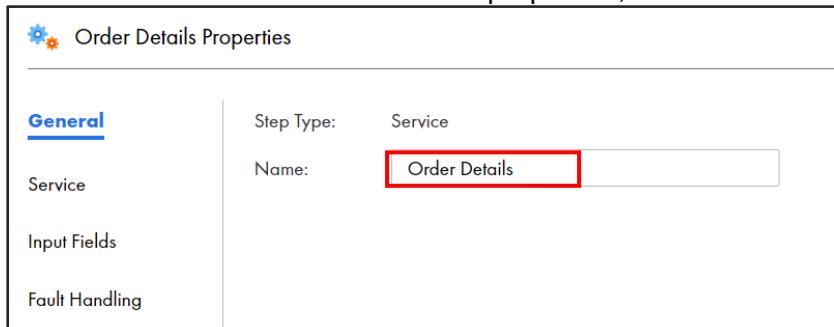
Next, you will add a service step that connects to the third-party service using the connection created in Lab 7-1: Create an App Connection using Service Connector.

17. From the Design palette, drag and drop a **Service** step on the link between the Start and End steps.



18. To configure the service, from the process canvas, select the **Service** step.

19. In the General section of the Service properties, enter the Name as **Order Details**.



| Order Details Properties |  |
|--------------------------|--|
| <b>General</b>           | Step Type: Service<br>Name: <b>Order Details</b> |
| Service                  |  |
| Input Fields             |  |
| Fault Handling           |  |

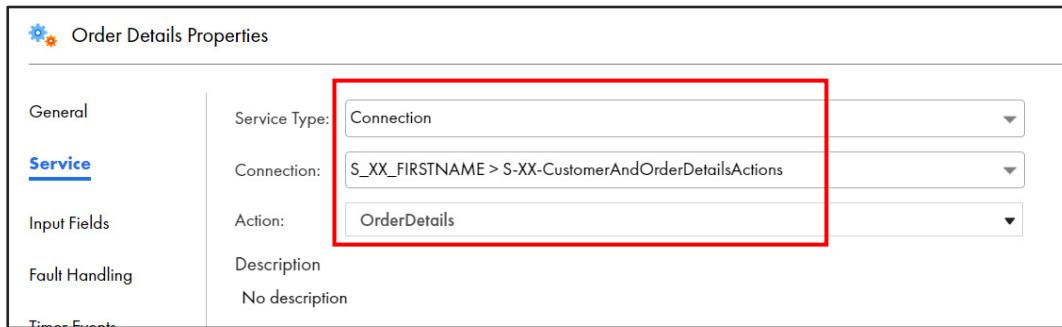
20. From the properties pane, select **Service**.

21. From the Service Type drop-down, select **Connection**.

22. From the Connection drop-down, expand your project folder, and select **S-XX-CustomerAndOrderDetailsActions**.

23. From the Action drop-down, select **OrderDetails**.

**Note:** This action is obtained from the service connector.



| Order Details Properties |  |
|--------------------------|--|
| <b>General</b>           | Service Type: Connection   |
| <b>Service</b>           | Connection: S_XX_FIRSTNAME > S-XX-CustomerAndOrderDetailsActions |
| Input Fields             | Action: <b>OrderDetails</b>                                      |
| Fault Handling           | Description<br>No description                                    |
| Timer Events             |  |

24. From the properties pane, select **Input Fields**.

**Note:** If you do not find the **order\_id** field already present, click the **+** icon, and select **order\_id**.

25. From the Value drop-down, select **Field**.

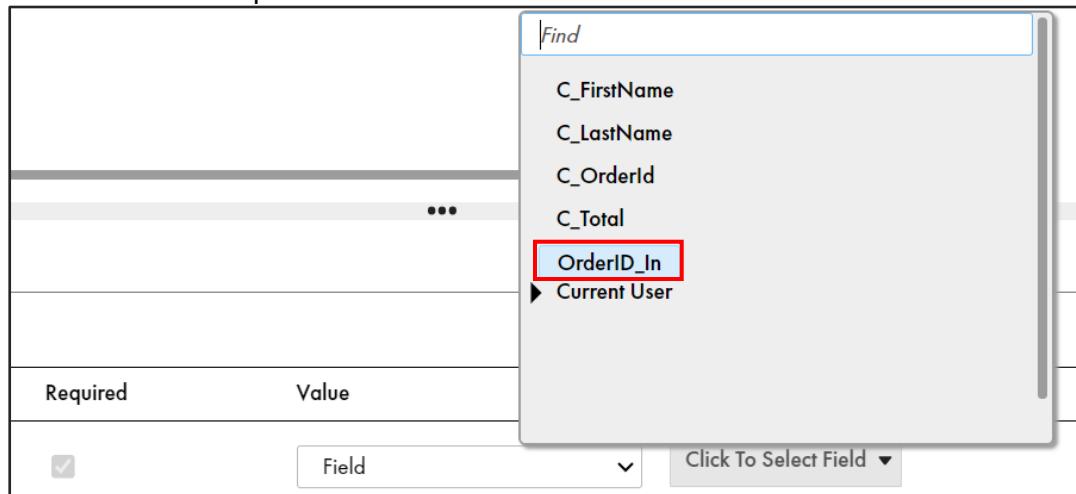
26. Click **Click To Select Field**.



| Input Fields (1) |                                     |  |
|------------------|-------------------------------------|--|
| Name             | Required                            | Value  |
| order_id         | <input checked="" type="checkbox"/> | <input type="button" value="Field"/> <span style="border: 1px solid red; padding: 2px;">Click To Select Field ▾</span> |

27. From the list, select **OrderID\_In**.

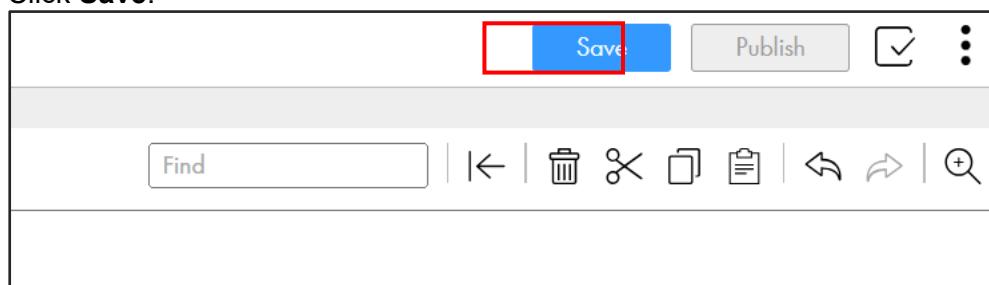
**Note:** This is the input field that will be used in the connection.



Find

- C\_FirstName
- C\_LastName
- C\_OrderId
- C\_Total
- OrderID\_In
- ▶ Current User

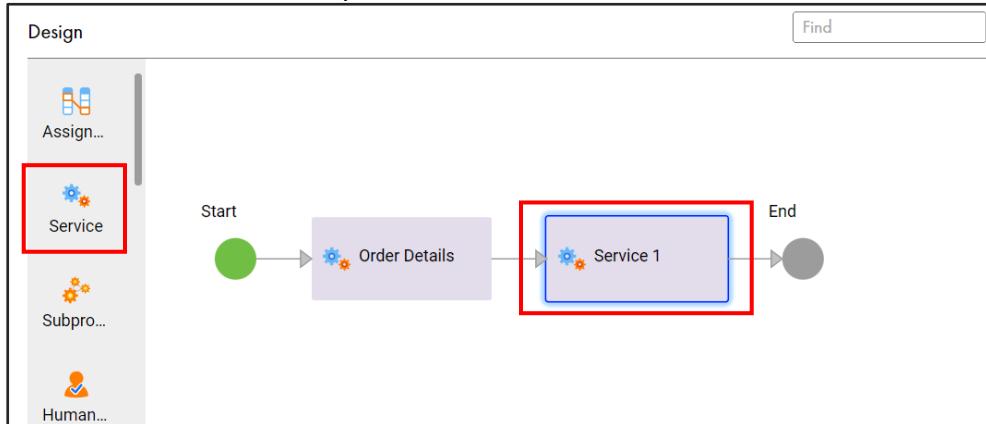
28. Click **Save**.



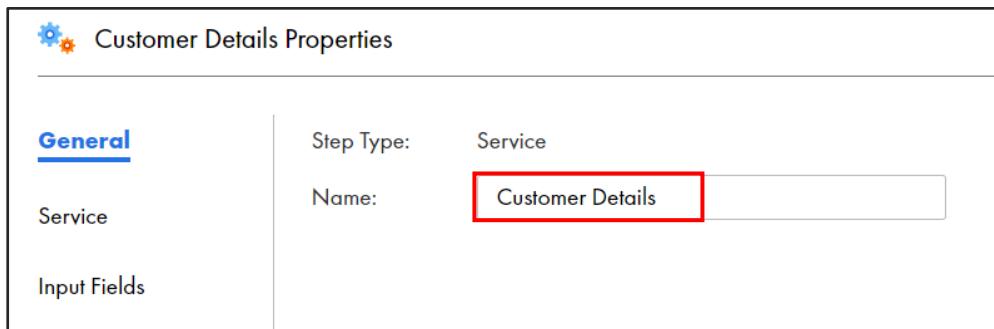
Save Publish  ⋮

Next, you need to add a service step that will retrieve the customer details of the specified order.

29. From the Design palette, drag and drop a **Service** step on the link between the Order Details and End steps.

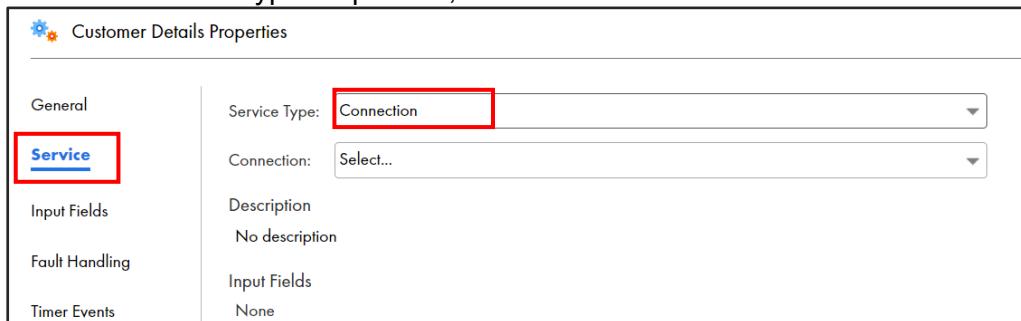


30. To configure the service, from the process canvas, select the **Service** step.  
 31. In the General section of the Service properties, enter the Name as **Customer Details**.



| Customer Details Properties |   |
|-----------------------------|---|
| <b>General</b>              | Step Type: Service<br>Name: <b>Customer Details</b> |
| Service                     |   |
| Input Fields                |   |

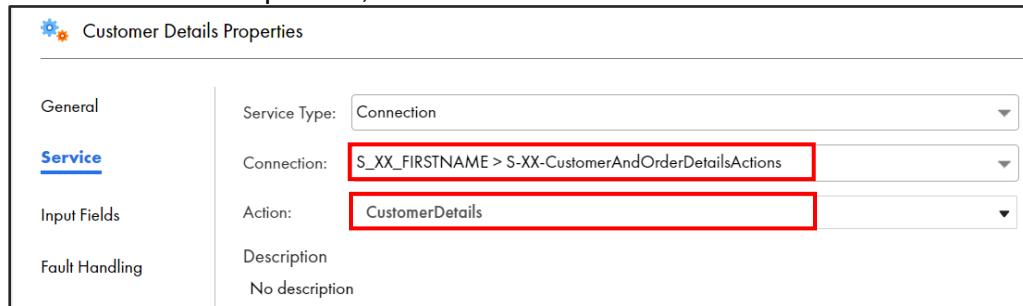
32. From the properties pane, select **Service**.  
 33. From the Service Type drop-down, select **Connection**.



| Customer Details Properties |                                 |
|-----------------------------|---------------------------------|
| General                     | Service Type: <b>Connection</b> |
| <b>Service</b>              | Connection: Select...           |
| Input Fields                | Description<br>No description   |
| Fault Handling              | Input Fields<br>None            |
| Timer Events                |                                 |

34. From the Connection drop-down, expand **your project folder**, and select **S-XX-CustomerAndOrderDetailsActions**.

35. From the Action drop-down, select **CustomerDetails**.



| Customer Details Properties |  |
|-----------------------------|--|
| General                     | Service Type: Connection   |
| <b>Service</b>              | Connection: S_XX_FIRSTNAME > S-XX-CustomerAndOrderDetailsActions |
| Input Fields                | Action: CustomerDetails  |
| Fault Handling              | Description<br>No description                                    |

**Note:** This action is obtained from the service connector.

36. From the properties pane, select **Input Fields**.

**Note:** If you do not find the customer\_id field already present, click the + icon, and select **customer\_id**.

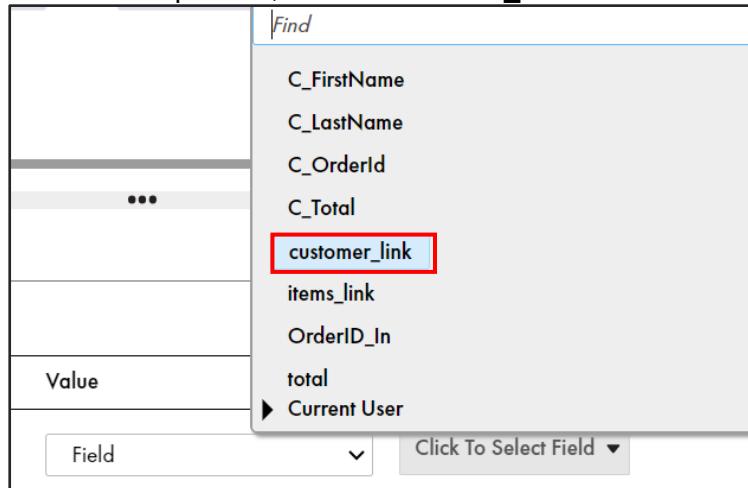
37. From the Value drop-down, select **Field**.

38. Click on **Click To Select Field**.



| Customer Details Properties |                         |                                     |       |
|-----------------------------|-------------------------|-------------------------------------|-------|
| General                     | Input Fields (1)        |                                     |       |
| Service                     | Name                    | Required                            | Value |
| <b>Input Fields</b>         | customer_id             | <input checked="" type="checkbox"/> | Field |
| Fault Handling              | Click To Select Field ▾ |                                     |       |

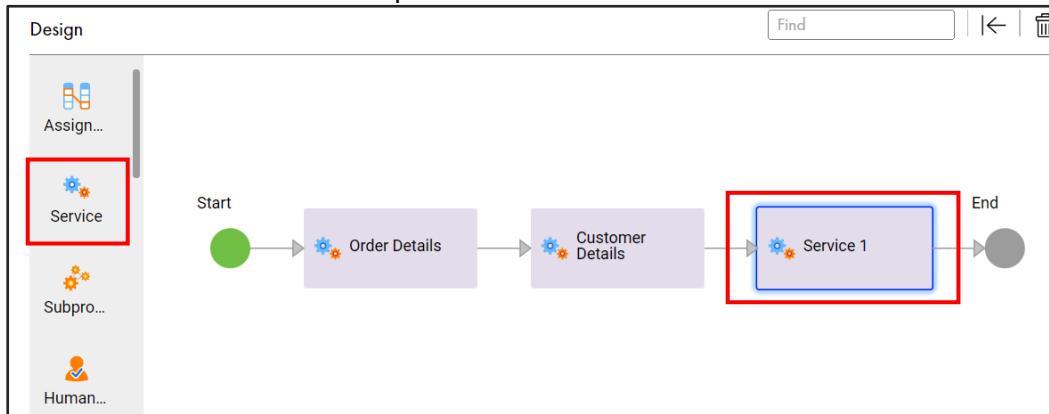
39. From the drop-down, select **customer\_link**.



| Find                 |                         |
|----------------------|-------------------------|
| C_FirstName          |                         |
| C_LastName           |                         |
| C_OrderId            |                         |
| C_Total              |                         |
| <b>customer_link</b> |                         |
| items_link           |                         |
| OrderID_In           |                         |
| total                |                         |
| ▶ Current User       |                         |
| Field                | Click To Select Field ▾ |

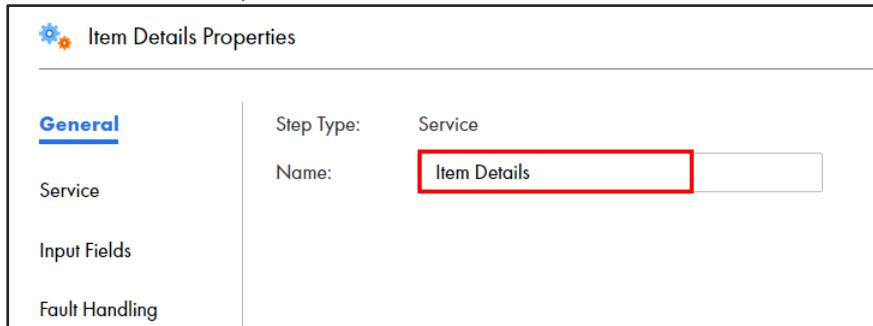
Now, you will add a service step to extract the item details of the specified order.

40. From the Design palette, drag and drop a **Service** step on the link between Customer Details and End steps.



41. To configure the service, from the process canvas, select the **Service** step.

42. In the General tab, enter the Name as **Item Details**.



| Item Details Properties |   |
|-------------------------|---|
| <b>General</b>          | Step Type: Service<br>Name: <b>Item Details</b> |
| Service                 |   |
| Input Fields            |   |
| Fault Handling          |   |

43. From the properties pane, select **Service**.

44. From the Service Type drop-down, select **Connection**.



| Properties     |                                 |
|----------------|---------------------------------|
| General        | Service Type: <b>Connection</b> |
| <b>Service</b> | Connection: Select...           |
| Input Fields   |                                 |
| Description    |                                 |

45. From the Connection drop-down, expand your project folder, and select **S-XX-CustomerAndOrderDetailsActions**.

46. From the Action drop-down, select **ItemDetails**.



| ItemDetails Properties        |  |
|-------------------------------|--|
| General                       | Service Type: Connection   |
| <b>Service</b>                | Connection: <b>S_XX_FIRSTNAME &gt; S-XX-CustomerAndOrderDetailsActions</b><br>Action: <b>ItemDetails</b> |
| Input Fields                  |  |
| Fault Handling                |  |
| Description<br>No description |  |

**Note:** This action is obtained from the service connector.

47. From the properties pane, select **Input Fields**.

**Note:** If you do not find the items\_id field already present, click the + icon, and select **items\_id**.

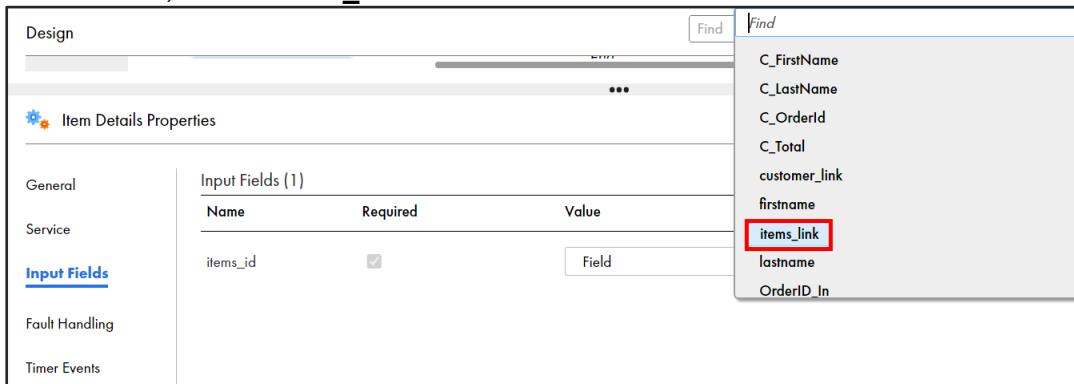
48. From the Value drop-down, select **Field**.

49. Click on **Click To Select Field**.



| Input Fields (1) |                                     |       |
|------------------|-------------------------------------|-------|
| Name             | Required                            | Value |
| items_id         | <input checked="" type="checkbox"/> | Field |

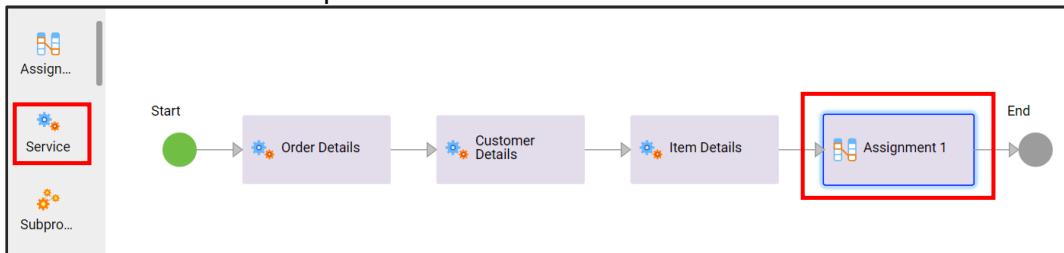
50. From the list, select **items\_link**.



| Input Fields (1) |                                     |       |
|------------------|-------------------------------------|-------|
| Name             | Required                            | Value |
| items_id         | <input checked="" type="checkbox"/> | Field |

Now that the information is extracted, it needs to be displayed as an output. For this, you will use an Assignment step.

51. From the Design palette, drag and drop an **Assignment** step on the link between the Item Details and End steps.



52. To configure the step, from the process canvas, select the **Assignment** step.

53. In the General section of Assignment properties, enter the Name as **Assignment to Output Fields**.



Assignment to Output Fields Properties

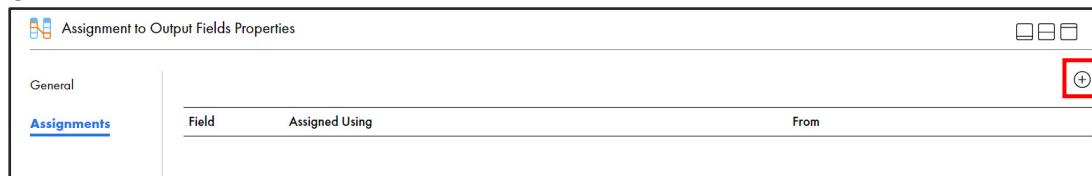
**General**

Step Type: Assignment

Name: **Assignment to Output Fields**

54. From the properties pane, select **Assignments**.

55. Click the **+** icon.



Assignment to Output Fields Properties

**General**

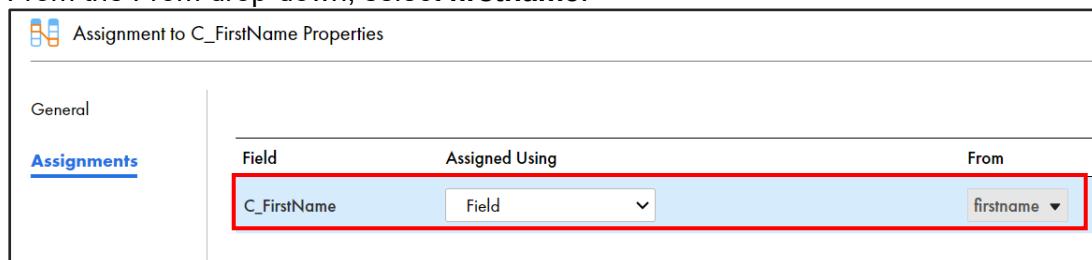
**Assignments**

| Field       | Assigned Using | From      |
|-------------|----------------|-----------|
| C_FirstName | Field          | firstname |

56. From the drop-down, select **C\_FirstName**.

57. From the Assigned Using drop-down, select **Field**.

58. From the From drop-down, select **firstname**.



Assignment to C\_FirstName Properties

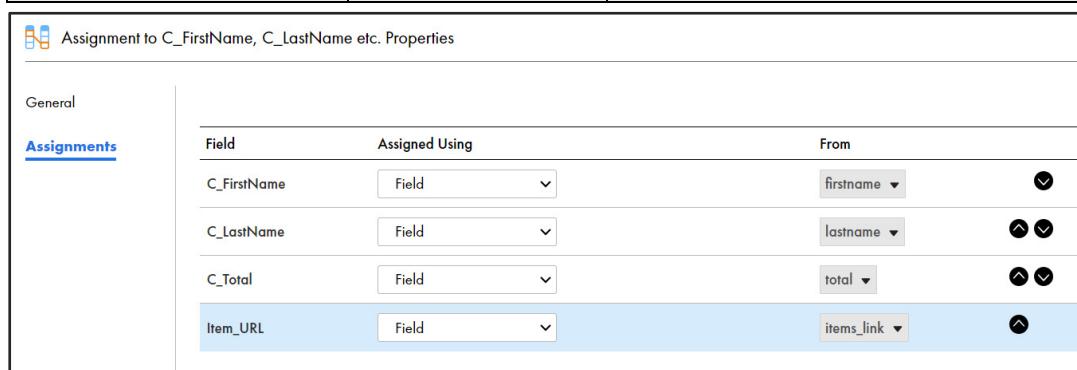
**General**

**Assignments**

| Field       | Assigned Using | From      |
|-------------|----------------|-----------|
| C_FirstName | Field          | firstname |

59. Similarly, add the additional fields listed in the table below:

| Add Field  | Assigned Using | From       |
|------------|----------------|------------|
| C_LastName | Field          | lastname   |
| C_Total    | Field          | total      |
| Item_URL   | Field          | Items_link |



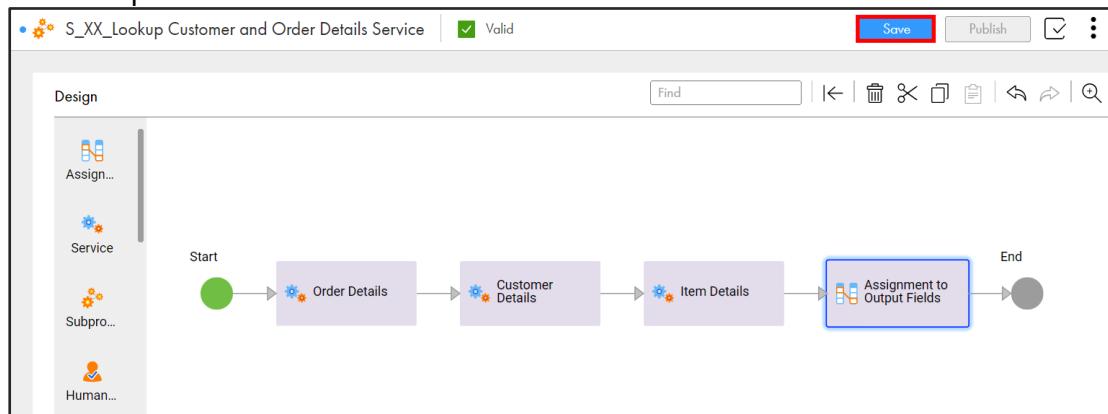
Assignment to C\_FirstName, C\_LastName etc. Properties

**General**

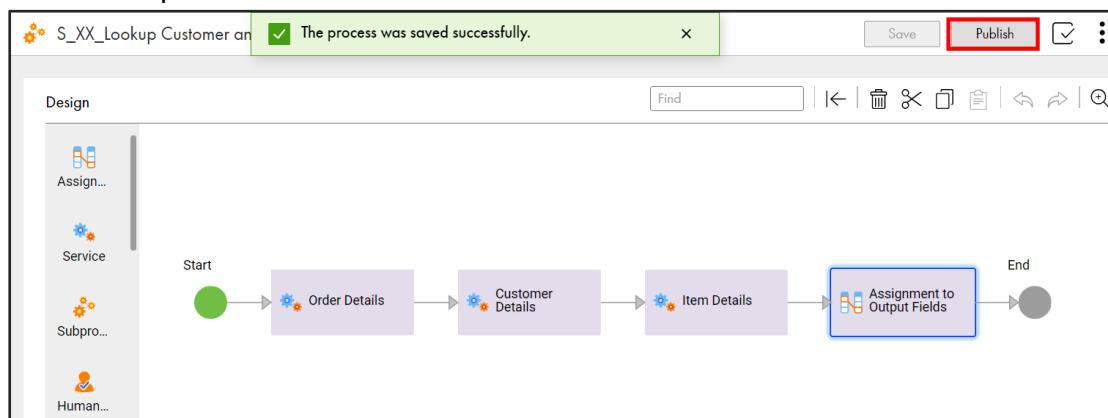
**Assignments**

| Field       | Assigned Using | From       |
|-------------|----------------|------------|
| C_FirstName | Field          | firstname  |
| C_LastName  | Field          | lastname   |
| C_Total     | Field          | total      |
| Item_URL    | Field          | items_link |

60. Save the process.

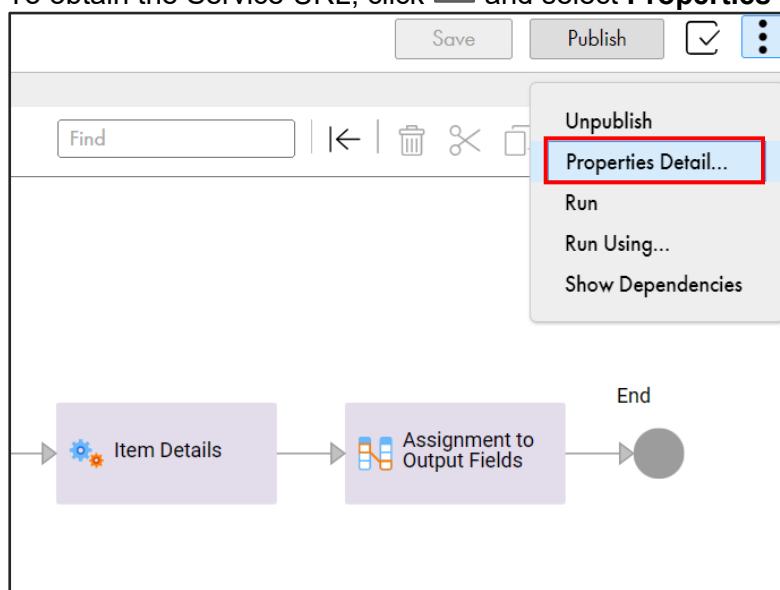


61. Publish the process.

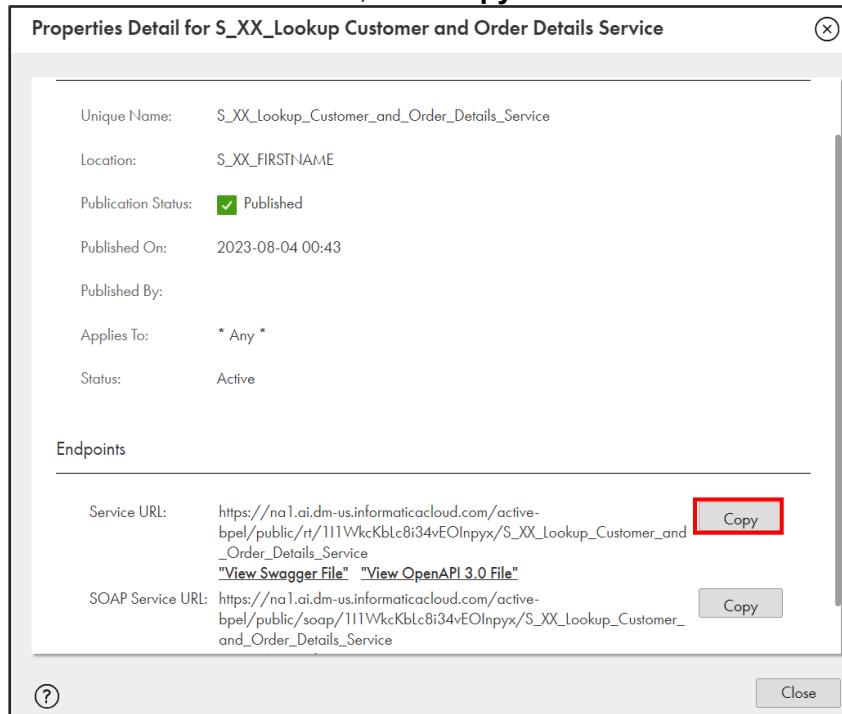


**Execute the Process:**

62. To obtain the Service URL, click  and select **Properties Detail**.



63. From the Service URL field, click **Copy**.



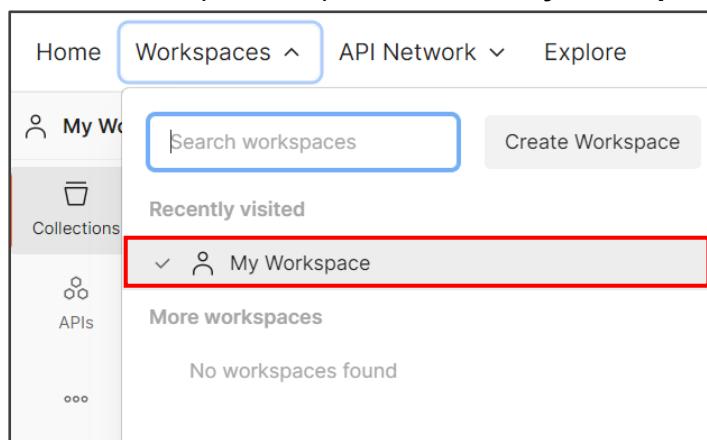
Now, run the process in Postman.

64. Open a web browser and enter the following link to open Postman:

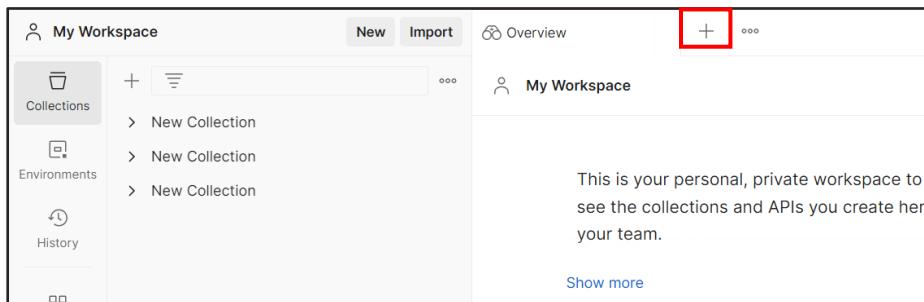
<https://identity.getpostman.com/>

65. Log in to Postman using your credentials.

66. From the Workplace drop-down, select **My Workspace**.

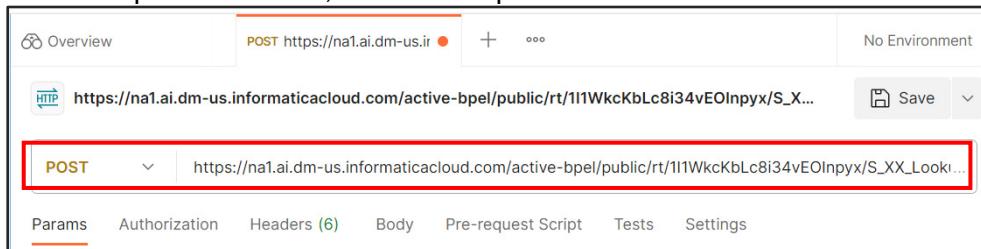


67. Click the '+' icon.



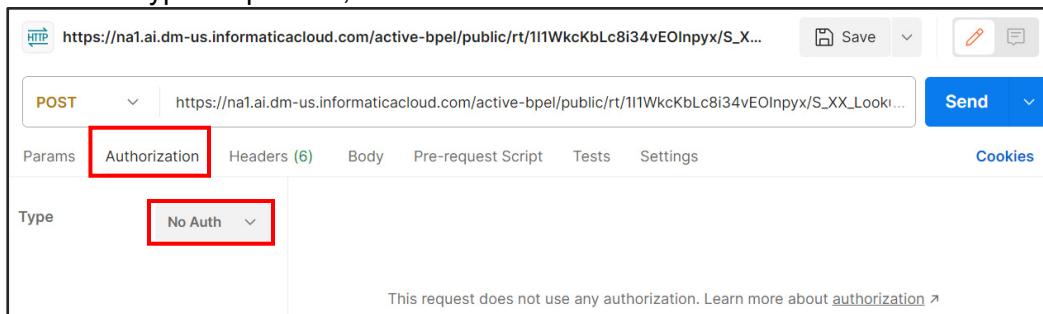
68. From the drop-down, select **POST**.

69. In the Request URL field, enter the copied Service URL.



70. Select the **Authorization** tab.

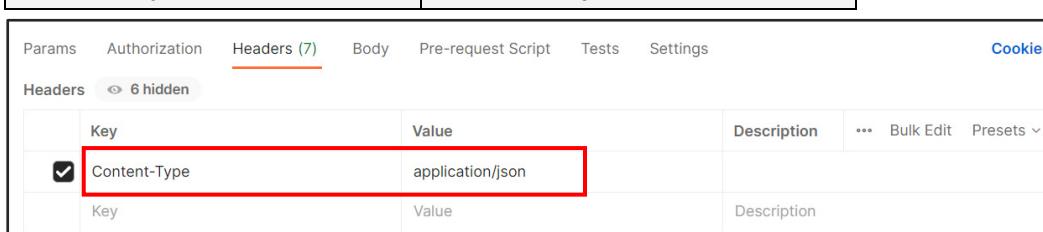
71. From the Type drop-down, select **No Auth**.



72. Select the **Headers** tab.

73. Enter **Key** and **Value** as shown in the table below:

| Key          | Value            |
|--------------|------------------|
| Content-Type | application/json |

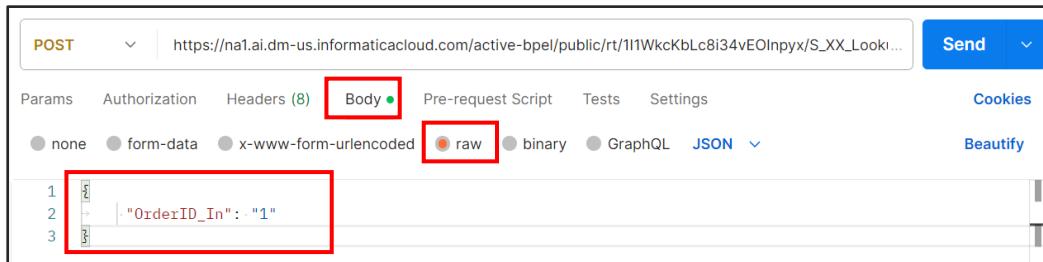
74. Select the **Body** tab and select **raw**.

75. Enter the following text body:

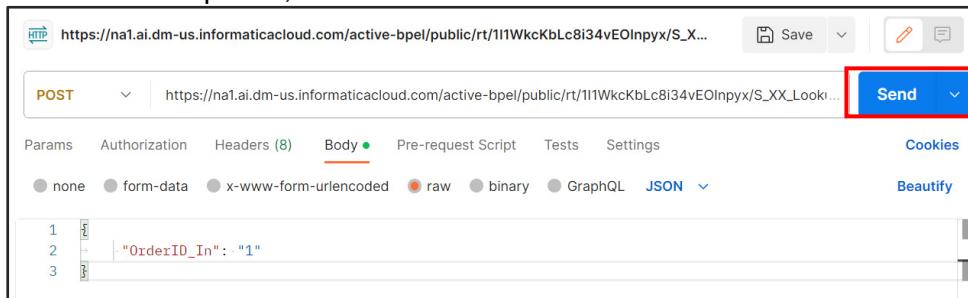
```
{
  "OrderID_In": "1"
}
```

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **11\_LabGuide\_CAI\_CustomerAndOrdersDetails\_7-3**. Copy the command mentioned under **Step A** and paste it in the body field.



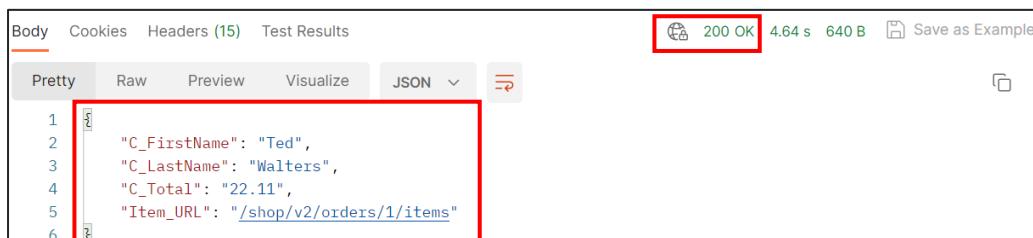
76. To view the response, click **Send**.



77. Verify that the status of the response is **200 OK**.

78. Examine the details in the response.

**Note:** The C\_Total value may differ based on the live data available in the API.



**Note:** Observe that you get the customer and order details for order id '1'.

*This concludes the lab.*

# Module 7: App Connections

## Lab 7-4: Create a File Connection

### Overview:

The File Connector provides connectivity between Application Integration and file systems to monitor file systems for new files, move files, read, and write content from files, and select options to handle processed files. You can use the File Connector to write log files that are needed as part of a larger integration, or to monitor a file system for new .csv files, read the delimited content from the files, or use the generated XML in a set of process objects.

In this lab, you will create a File connection using the built-in File Connector.

### Objectives:

- Create a File connection
- Publish the connection

### Duration:

15 Minutes

---

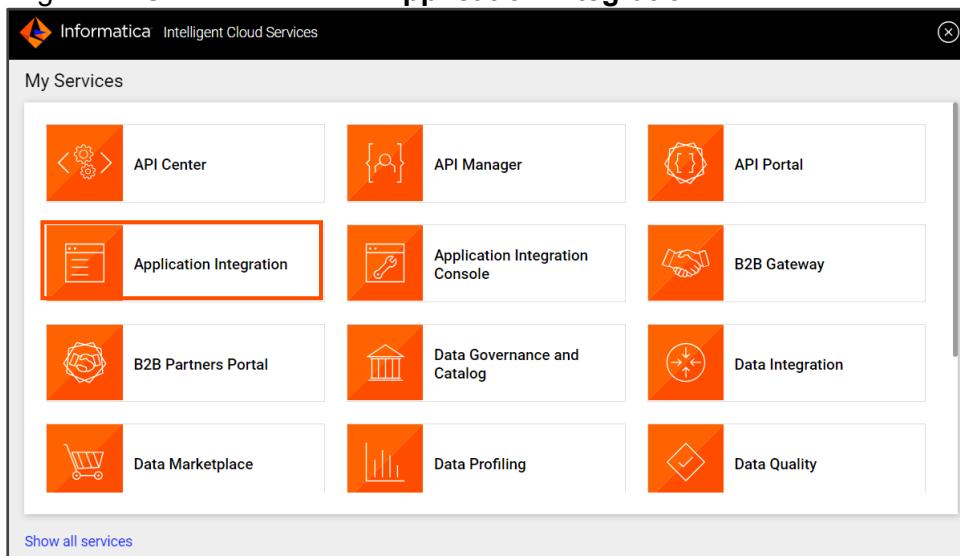
### Tasks

**Note:** The screen in your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

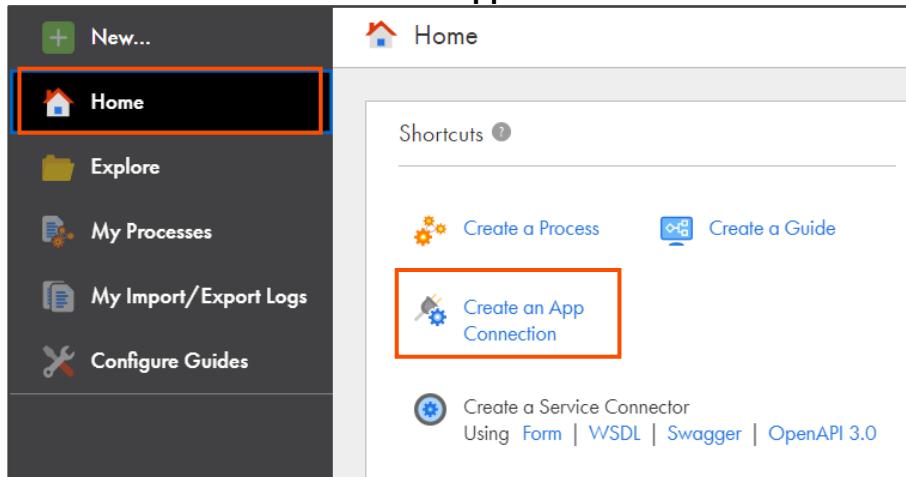
**IMPORTANT:** Make sure you have created a folder named **CAI** in the C: drive of your machine. Also, inside the CAI folder, you have created a folder **DataFiles**. The DataFiles folder path is **C:\CAI\DataFiles**.

### Create a Connection

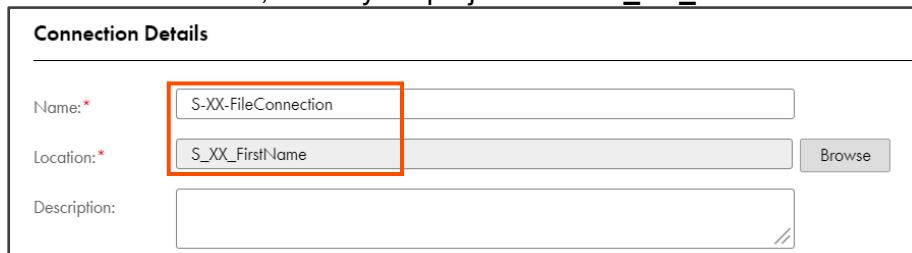
1. Log in to IICS and select the **Application Integration** service.



2. Click **Home** and select **Create an App Connection**.

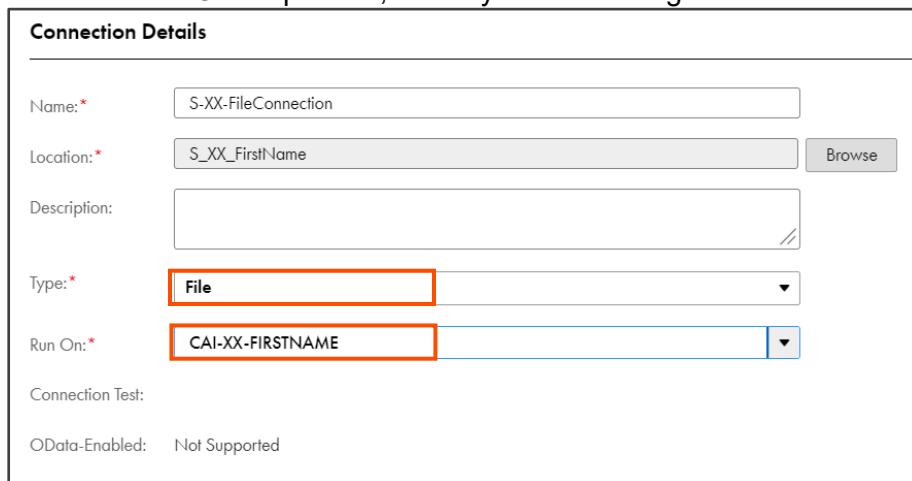


3. In the Connection Details page, enter the connection name as **S-XX-FileConnection**.  
 4. In the Location field, select your project folder **S\_XX\_FirstName**.



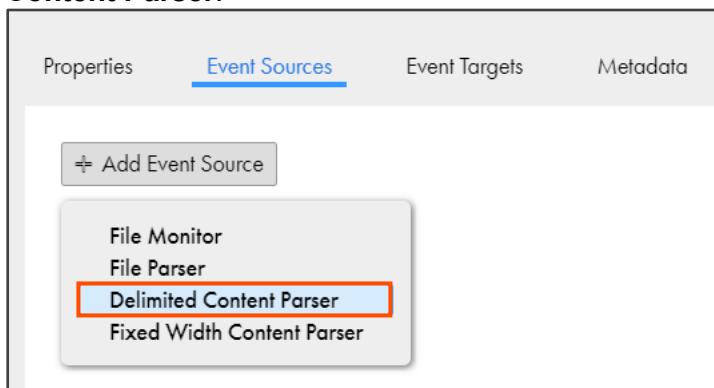
This screenshot shows the 'Connection Details' page. It has fields for 'Name:' (containing 'S-XX-FileConnection') and 'Location:' (containing 'S\_XX\_FirstName'). There's also a 'Browse' button next to the location field and a 'Description:' field which is empty.

5. From the Type drop-down, select **File**.  
 6. From the Run On drop-down, select your Secure Agent.



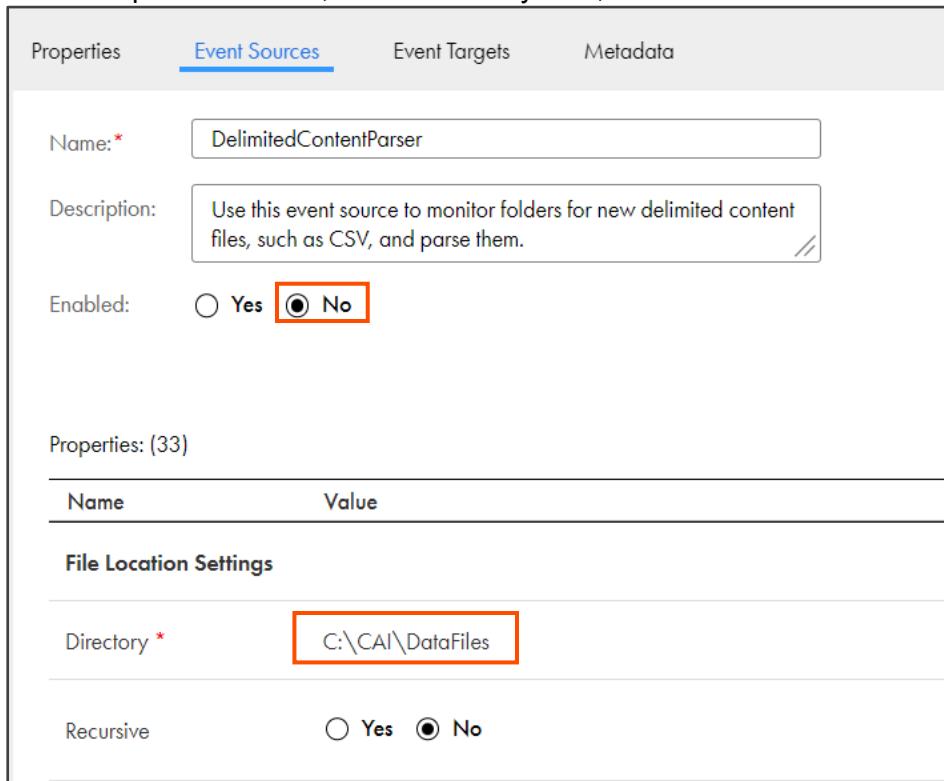
This screenshot shows the 'Connection Details' page with more fields. The 'Name:' field contains 'S-XX-FileConnection', the 'Location:' field contains 'S\_XX\_FirstName', and the 'Type:' dropdown is set to 'File' (highlighted with a red box). The 'Run On:' dropdown is set to 'CAI-XX-FIRSTNAME' (also highlighted with a red box). Other visible fields include 'Description:', 'Connection Test:', and 'OData-Enabled: Not Supported'.

7. To configure the Event Sources, click the **Event Sources** tab.
8. To add different Event Sources, click “**Add Event Source**” and select “**Delimited Content Parser**.”



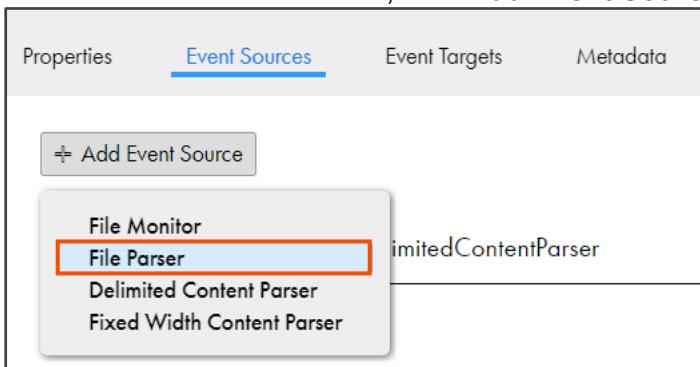
9. Retain the values for the **Name** and **Description** fields.
10. For the “Enabled” option, select **No**.  
**Note:** When you select No, the event source is disabled until you are ready to use it. If you select Yes, the event source is available immediately after it is published.

11. In the Properties section, in the Directory field, enter **C:\CAI\DataFiles**.



| Name        | Value   |
|-------------|---|
| Directory * | C:\CAI\DataFiles  |
| Recursive   | <input type="radio"/> Yes <input checked="" type="radio"/> No |

12. To add another Event Source, click **Add Event Source** and select **File Parser**.

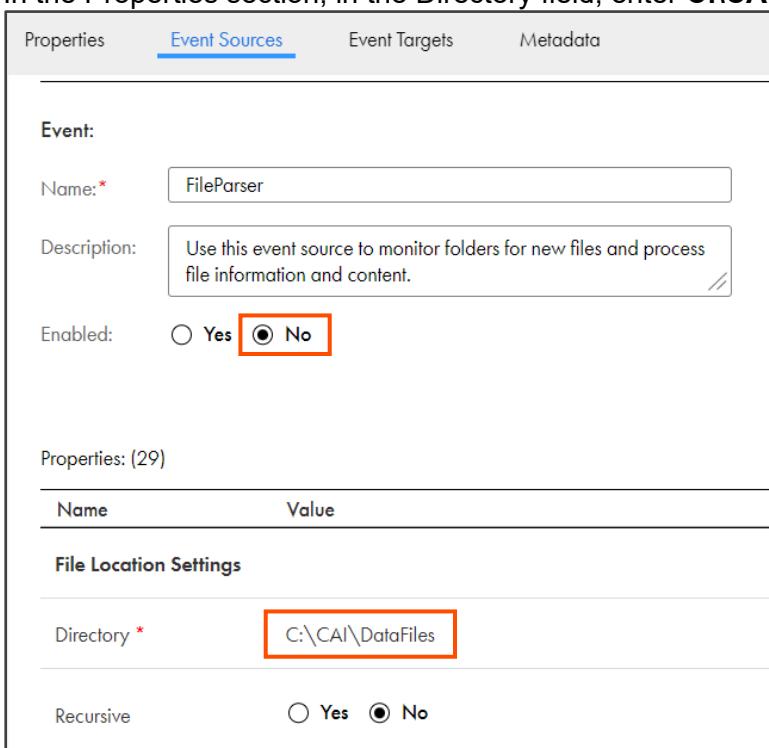


The screenshot shows the 'Event Sources' tab selected in the top navigation bar. Below it is a button labeled '+ Add Event Source'. A dropdown menu is open, listing four options: 'File Monitor', 'File Parser' (which is highlighted with a red box), 'Delimited Content Parser', and 'Fixed Width Content Parser'. The 'File Parser' option is currently selected.

13. Retain the values for the **Name** and **Description** fields.

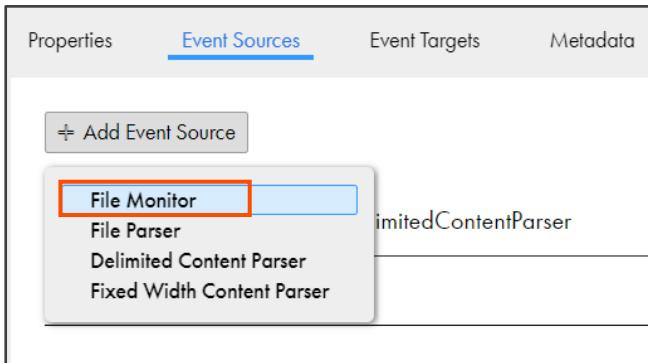
14. For the Enabled option, select **No**.

15. In the Properties section, in the Directory field, enter **C:\CAI\DataFiles**.



The screenshot shows the properties for a 'File Parser' event source. The 'Event Sources' tab is selected in the top navigation. The 'Event:' section contains fields for 'Name:' (set to 'FileParser') and 'Description:' (containing a placeholder text about monitoring folders for new files). The 'Enabled:' section has radio buttons for 'Yes' and 'No', with 'No' selected. Below this is a 'Properties: (29)' section, which includes a table with two rows: 'File Location Settings' and 'Recursive'. The 'Directory' field in the 'File Location Settings' row is set to 'C:\CAI\DataFiles' and is highlighted with a red box. The 'Recursive' field has radio buttons for 'Yes' and 'No', with 'No' selected.

16. Add another Event Source and select **File Monitor**.



The screenshot shows the 'Event Sources' tab selected in the top navigation bar. Below it is a button labeled '+ Add Event Source'. A dropdown menu is open, listing four options: 'File Monitor' (which is highlighted with a red box), 'File Parser', 'Delimited Content Parser', and 'Fixed Width Content Parser'. The 'File Monitor' option is currently selected.

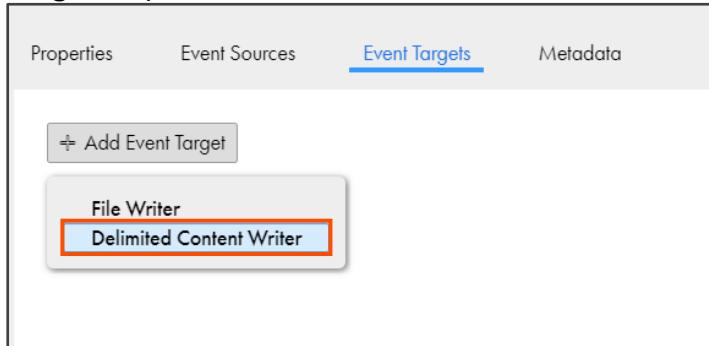
17. Retain the values for the **Name** and **Description** fields.
18. For the “Enabled” option, select **No**.
19. In the Properties section, in the Directory field, enter **C:\CAI\DataFiles**.

| Properties  | <u>Event Sources</u>  | Event Targets | Metadata |      |       |                               |  |             |   |           |   |
|---|---|---------------|----------|------|-------|-------------------------------|--|-------------|---|-----------|---|
| <p><b>Event:</b></p> <p>Name: * <input type="text" value="FileMonitor"/></p> <p>Description: <input type="text" value="Use this event source to monitor folders for new files."/></p> <p>Enabled: <input type="radio"/> Yes <input checked="" type="radio"/> No</p>   |   |               |          |      |       |                               |  |             |   |           |   |
| <p>Properties: (23)</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td colspan="2"><b>File Location Settings</b></td> </tr> <tr> <td>Directory *</td> <td><input type="text" value="C:\CAI\DataFiles"/></td> </tr> <tr> <td>Recursive</td> <td><input type="radio"/> Yes <input checked="" type="radio"/> No</td> </tr> </tbody> </table> |   |               |          | Name | Value | <b>File Location Settings</b> |  | Directory * | <input type="text" value="C:\CAI\DataFiles"/> | Recursive | <input type="radio"/> Yes <input checked="" type="radio"/> No |
| Name  | Value   |               |          |      |       |                               |  |             |   |           |   |
| <b>File Location Settings</b>   |   |               |          |      |       |                               |  |             |   |           |   |
| Directory *   | <input type="text" value="C:\CAI\DataFiles"/>                 |               |          |      |       |                               |  |             |   |           |   |
| Recursive   | <input type="radio"/> Yes <input checked="" type="radio"/> No |               |          |      |       |                               |  |             |   |           |   |

20. In the Include Files field, add an extension **\*.txt**.

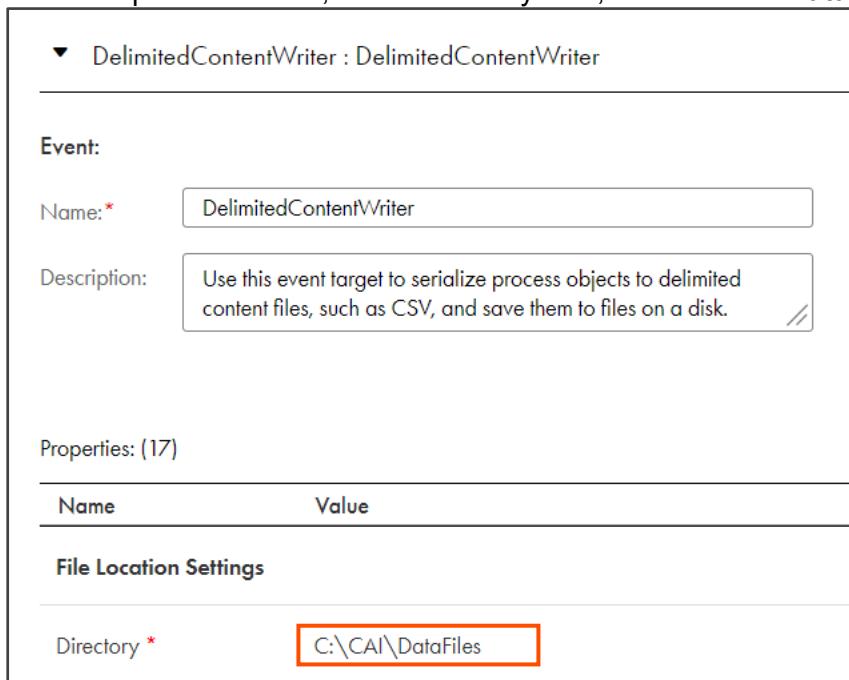
|                               |   |
|-------------------------------|---|
| <b>File Location Settings</b> |   |
| Directory *                   | <input type="text" value="C:\CAI\DataFiles"/>                 |
| Recursive                     | <input type="radio"/> Yes <input checked="" type="radio"/> No |
| Include Files                 | <input type="text" value="*.txt"/>                            |
| Exclude Files                 |   |

21. To configure the Event Targets, click the **Event Targets** tab, and from the **Add Event Target** drop-down, select **Delimited Content Writer**.

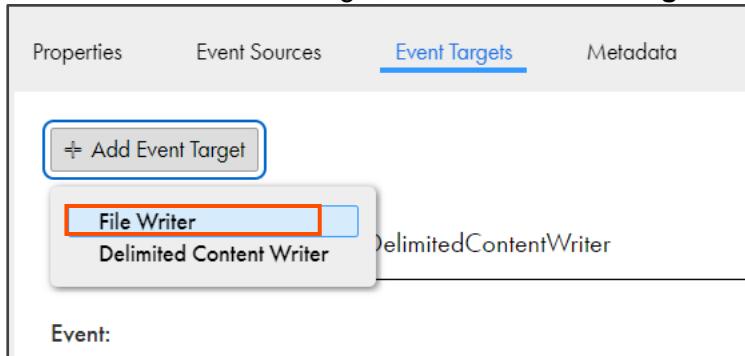


22. Retain the values for the **Name** and **Description** fields.

23. In the Properties section, in the Directory field, enter **C:\CAI\DataFiles**.

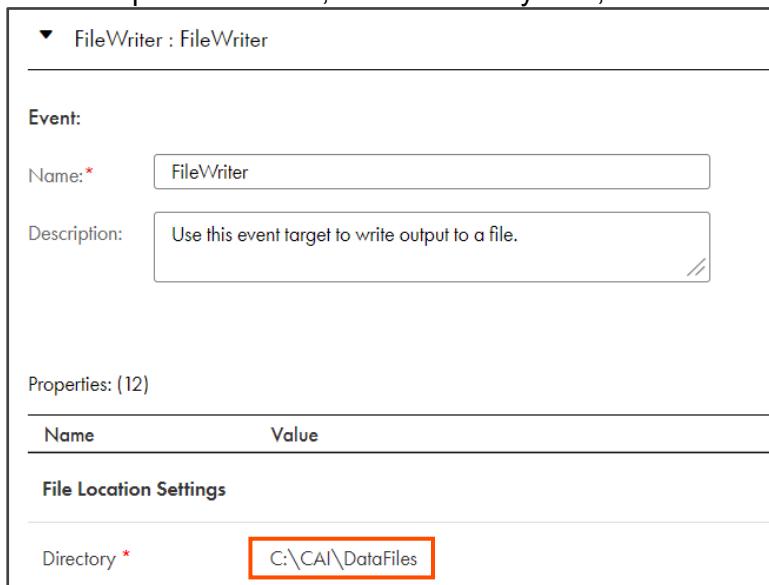

 A screenshot of a properties configuration screen for a 'DelimitedContentWriter'. At the top, it shows the type as 'DelimitedContentWriter : DelimitedContentWriter'. The 'Event:' section contains a 'Name:' field set to 'DelimitedContentWriter' and a 'Description:' field containing a tooltip about serializing objects to CSV files. Below this is a 'Properties: (17)' section with a table header 'Name' and 'Value'. Under 'File Location Settings', there is a 'Directory' field containing 'C:\CAI\DataFiles'.

24. To add another Event Target, click **Add Event Target** and select **File Writer**.



25. Retain the values for the **Name** and **Description** fields.

26. In the Properties section, in the Directory field, enter **C:\CAI\DataFiles**.



▼ FileWriter : FileWriter

**Event:**

Name: \*

Description:

Properties: (12)

| Name                          | Value   |
|-------------------------------|---|
| <b>File Location Settings</b> |   |
| Directory *                   | <input type="text" value="C:\CAI\DataFiles"/> |

27. Save and Test the connection.



S-XX-FileConnection |  Valid

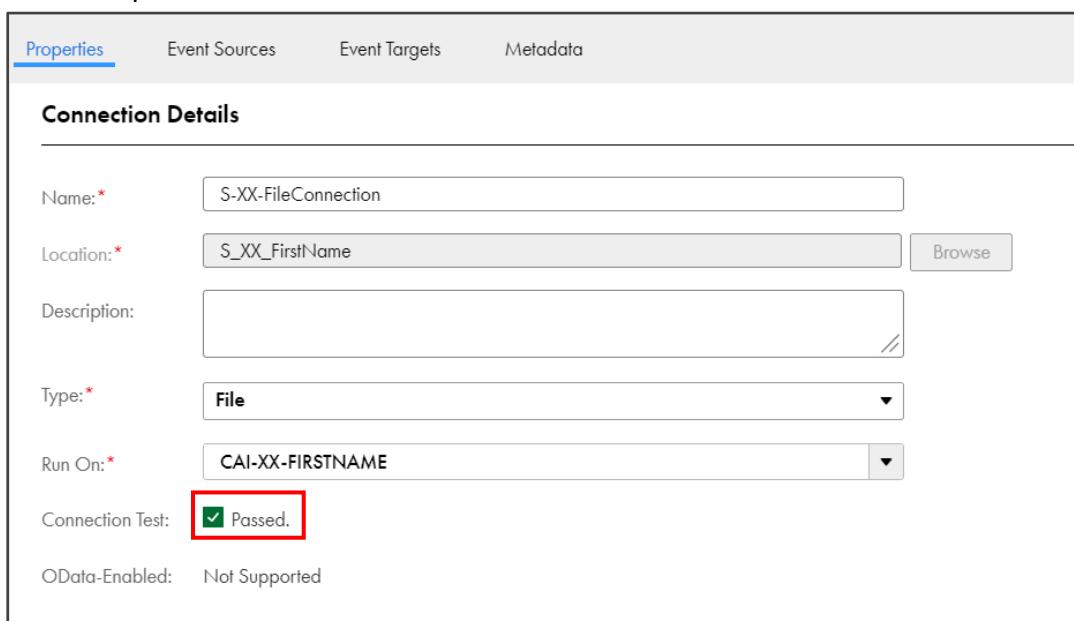
Save Publish Test

Properties Event Sources Event Targets Metadata

+ Add Event Target

► DelimitedContentWriter : DelimitedContentWriter

28. In the Properties tab, the Connection Test field mentions “Passed”.



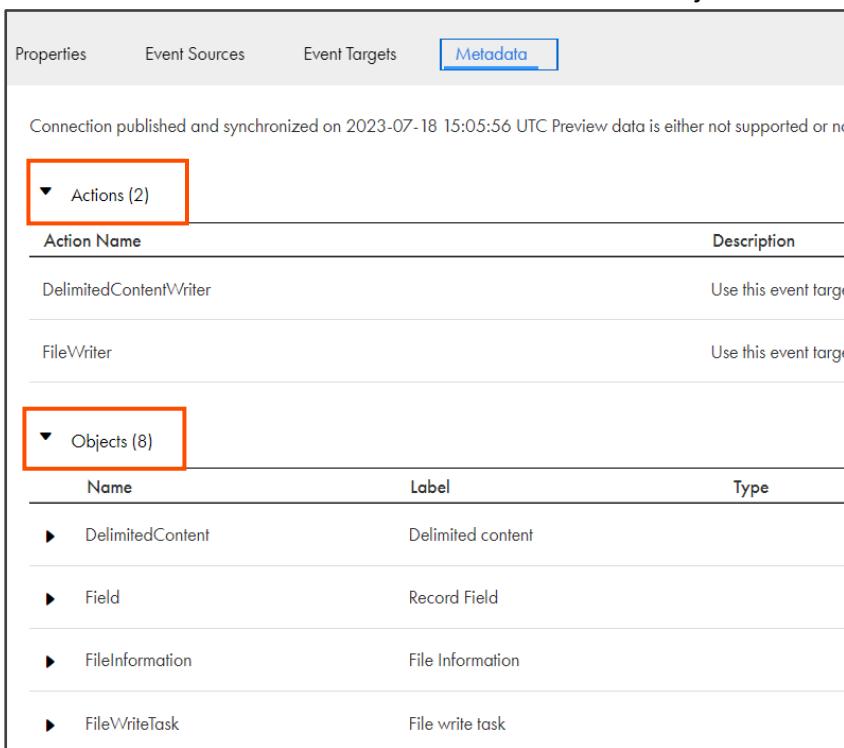
| Properties                |  | Event Sources | Event Targets | Metadata                              |
|---------------------------|--|---------------|---------------|---------------------------------------|
| <b>Connection Details</b> |  |               |               |                                       |
| Name: *                   | <input type="text" value="S-XX-FileConnection"/> |               |               |                                       |
| Location: *               | <input type="text" value="S_XX_FirstName"/>      |               |               | <input type="button" value="Browse"/> |
| Description:              | <input type="text"/>                             |               |               |                                       |
| Type: *                   | <input type="text" value="File"/>                |               |               |                                       |
| Run On: *                 | <input type="text" value="CAI-XX-FIRSTNAME"/>    |               |               |                                       |
| Connection Test:          | <input checked="" type="checkbox"/> Passed.      |               |               |                                       |
| OData-Enabled:            | Not Supported                                    |               |               |                                       |

29. Publish the connection.



You will get a notification that '**The asset [S-XX-FileConnection] was published successfully.**'.

30. In the **Metadata** tab, observe that the Actions and Objects are added.



The screenshot shows the "Metadata" tab selected in the Informatica interface. It displays a list of actions and objects. The "Actions" section, which contains 2 items, is highlighted with a red box. The "Objects" section, which contains 8 items, is also highlighted with a red box. The table below lists the actions:

| Action Name            | Description   |
|------------------------|---|
| DelimitedContentWriter | Use this event target to write delimited content to a file. |
| FileWriter             | Use this event target to write content to a file.           |

| Name               | Label             | Type |
|--------------------|-------------------|------|
| ► DelimitedContent | Delimited content |      |
| ► Field            | Record Field      |      |
| ► FileInformation  | File Information  |      |
| ► FileWriteTask    | File write task   |      |

---

*This concludes the lab.*

## Module 7: App Connections

### Lab 7-5: Create a Salesforce Connection

**Overview:**

Salesforce is a cloud-based Customer Relationship Management (CRM) solution for sales teams to manage contacts and sales activities. You can use Salesforce to store and manage contacts and data of the sales activities of your organization.

In this lab, you will install the Managed Package and create a Salesforce connection.

**Objectives:**

- Install Managed Package
- Create a Salesforce connection

**Duration:**

10 Minutes

---

**Tasks****Important:**

Before you start this lab, ensure that you have a functional Salesforce Developer account.

If you do not have a Salesforce Developer account, create one by using the following URL:  
<https://developer.salesforce.com/signup>

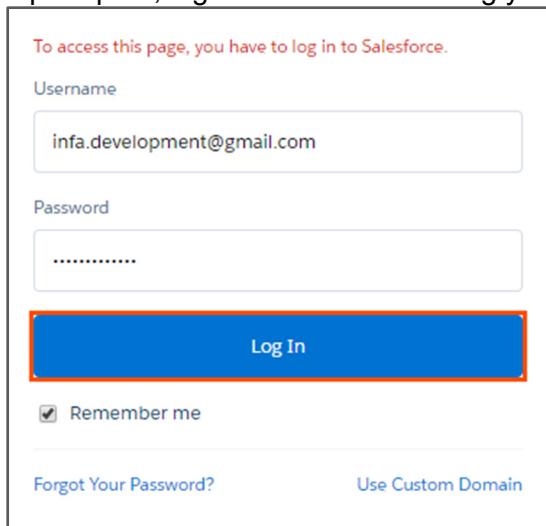
On the sign-up page, provide your details and enter the validation code that is sent to your email.

**Troubleshooting Tip:** If at any time the browser hangs and the desired output is not obtained (assets do not get published, windows do not open, and so on), try closing the browser window and opening a fresh one.

**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

## Install Salesforce Managed Package

1. To install the Managed Package, open a web browser, and enter the link mentioned below:  
<https://login.salesforce.com/packaging/installPackage.apexp?p0=04t1J000000g0tF>  
OR,  
Open the **CAI Lab Prep Files** folder and open the notepad file named **13\_LabGuide\_CAI\_SalesforceConnection\_7-5**. Copy the command mentioned under **Step A** and paste it in the body field.
2. If prompted, log in to Salesforce using your Salesforce credentials.



To access this page, you have to log in to Salesforce.

Username

Password

Remember me

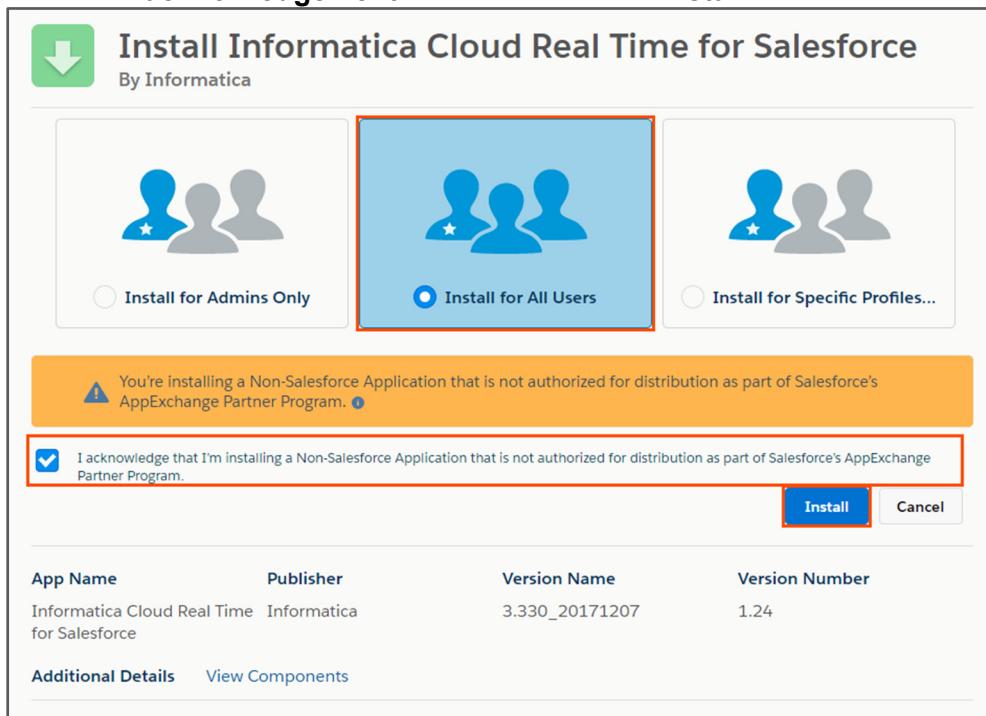
[Forgot Your Password?](#) [Use Custom Domain](#)

**Log In**

A screenshot of the Salesforce login page. It shows a red rectangular box highlighting the blue "Log In" button. The page includes fields for Username (infa.development@gmail.com) and Password, and a "Remember me" checkbox. Below the login form are links for password recovery and custom domains.

3. On the installation page, select the **Install for All Users** option.

4. Select the **acknowledgement** checkbox and click **Install**.

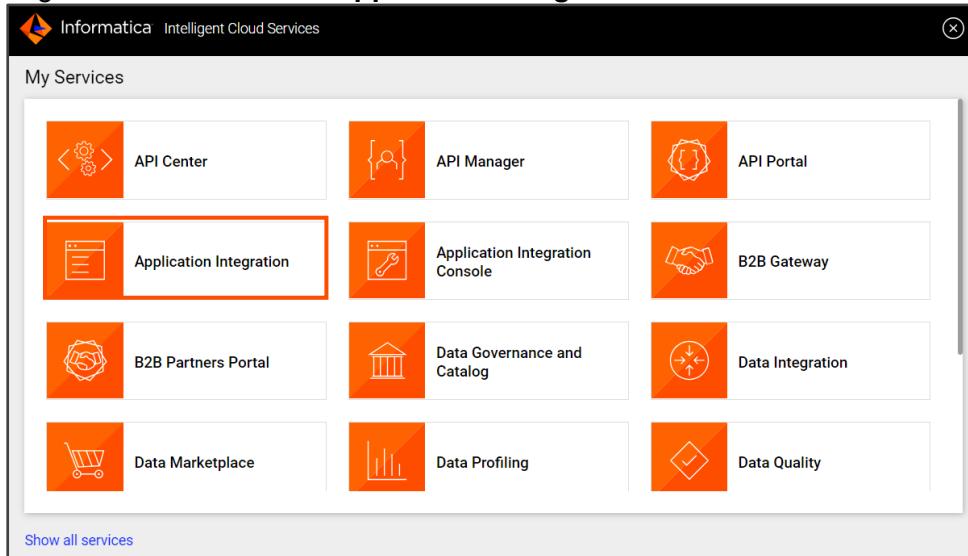


**Note:** It takes 3-5 minutes to install the package. When the installation is complete, you will receive a confirmation email to your registered Salesforce account.

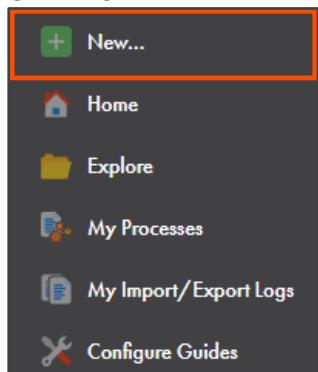
5. Click **Done**.

## Create Salesforce connection

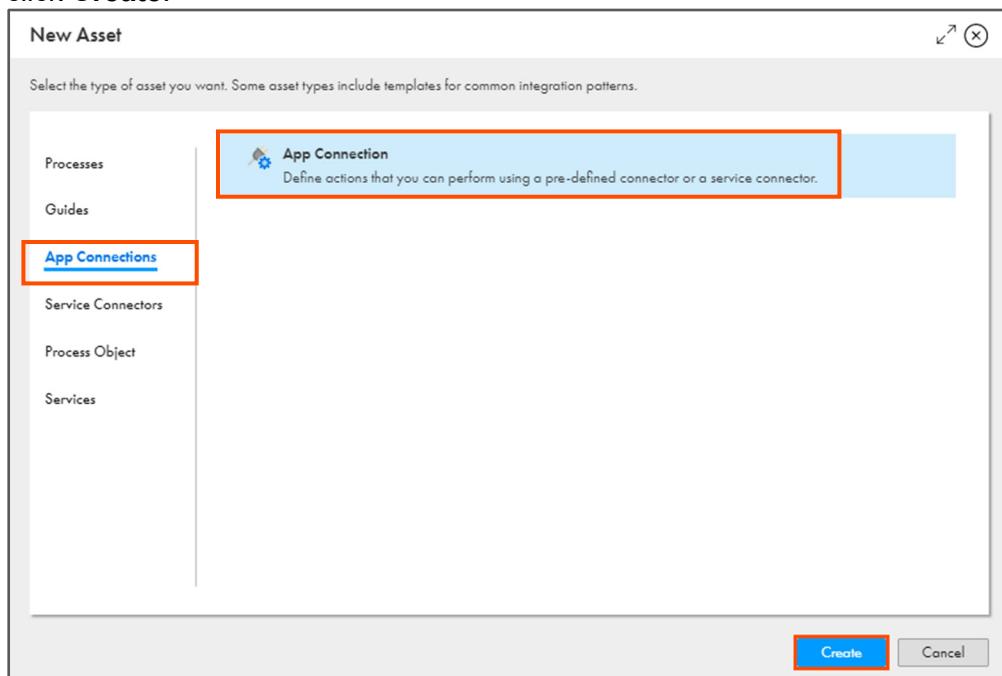
6. Log in to IICS and select **Application Integration**.



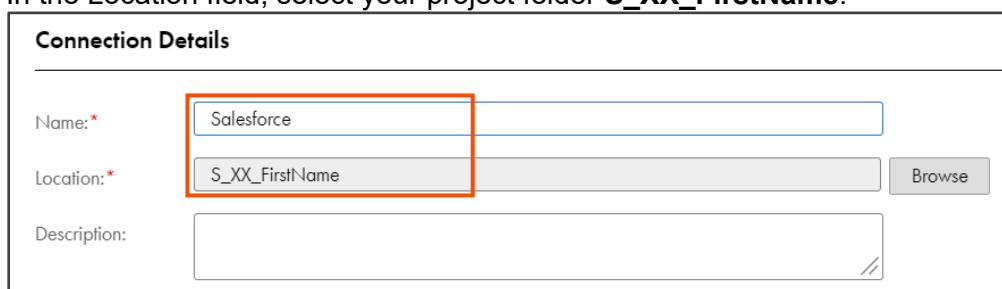
7. Click **New**.



8. To create a new App Connection, select **App Connections > App Connection** and click **Create**.



9. In the Connection Details page, enter the connection name as **Salesforce**.  
**Note:** To use this connection to create Guides, it is mandatory to name the connection as **Salesforce**.
10. In the Location field, select your project folder **S\_XX\_FirstName**.



| Connection Details |   |
|--------------------|---|
| Name:*             | <input type="text" value="Salesforce"/>   |
| Location:*         | <input type="text" value="S_XX_FirstName"/> <input type="button" value="Browse"/> |
| Description:       | <input type="text"/>  |

11. From the Type drop-down, select **Salesforce**.
12. From the Run On drop-down, select your **Cloud Server or any Secure Agent**.

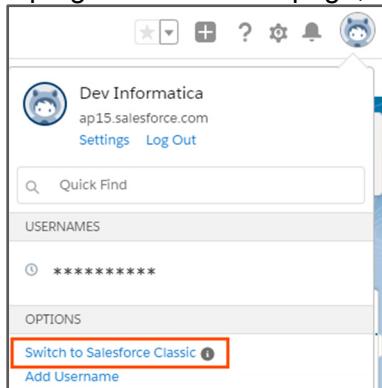
|                  |   |
|------------------|---|
| Description:     | <input type="text"/>  |
| Type:*           | <input type="text" value="Salesforce"/>                       |
| Run On:*         | <input type="text" value="Cloud Server or any Secure Agent"/> |
| Connection Test: |   |
| OData-Enabled:   | <input checked="" type="radio"/> No <input type="radio"/> Yes |

**Note:** Publishing could be faster when we publish on the cloud server instead of any secure agent. That is why, we selected the run-time environment as **Cloud Server or any Secure Agent** and not Secure Agent.

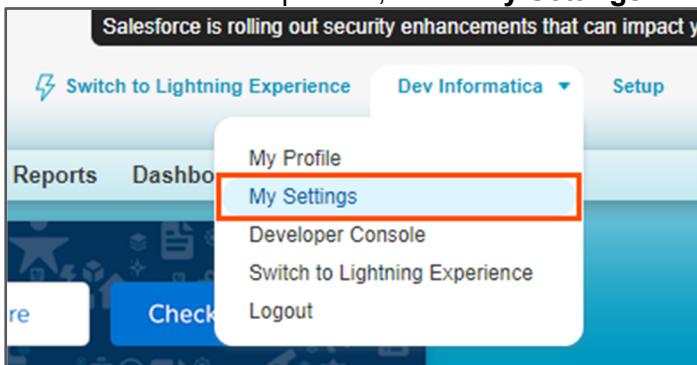
13. Retain the Authentication Type as **Password**.
14. Enter the **User Name** and **Password** of your Salesforce account.

| <b>Authentication:</b>   |                                       |      |       |             |                          |            |       |                  |  |
|--|---------------------------------------|------|-------|-------------|--------------------------|------------|-------|------------------|--|
| Authentication Type:   | <input type="text" value="Password"/> |      |       |             |                          |            |       |                  |  |
| <table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>User Name *</td> <td>rtrivedi@informatica.com</td> </tr> <tr> <td>Password *</td> <td>*****</td> </tr> <tr> <td>Security Token *</td> <td></td> </tr> </tbody> </table> |                                       | Name | Value | User Name * | rtrivedi@informatica.com | Password * | ***** | Security Token * |  |
| Name   | Value                                 |      |       |             |                          |            |       |                  |  |
| User Name *  | rtrivedi@informatica.com              |      |       |             |                          |            |       |                  |  |
| Password *   | *****                                 |      |       |             |                          |            |       |                  |  |
| Security Token *   |                                       |      |       |             |                          |            |       |                  |  |

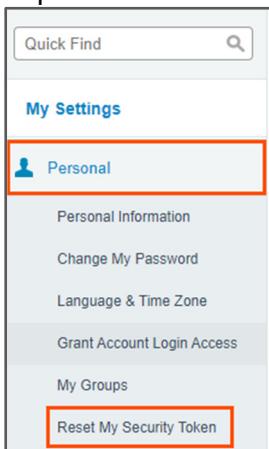
15. To obtain the Security Token, navigate back to your Salesforce account.
- Note:** If you do not have the Salesforce Classic view, then you need to switch to the Classic view. To switch from Lightning to Classic view, click the **My Profile** icon on the top right corner of the page, and select **Switch to Salesforce Classic**.



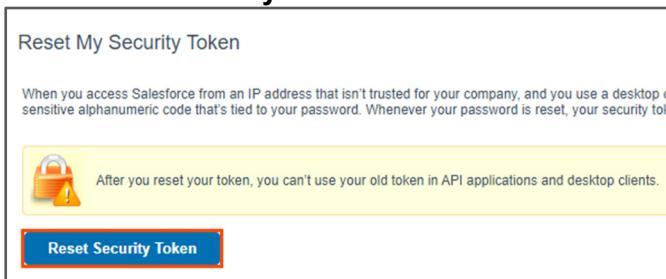
16. From the account drop-down, select **My Settings**.



17. Expand the **Personal** option and select **Reset My Security Token**.



18. Click **Reset Security Token**.



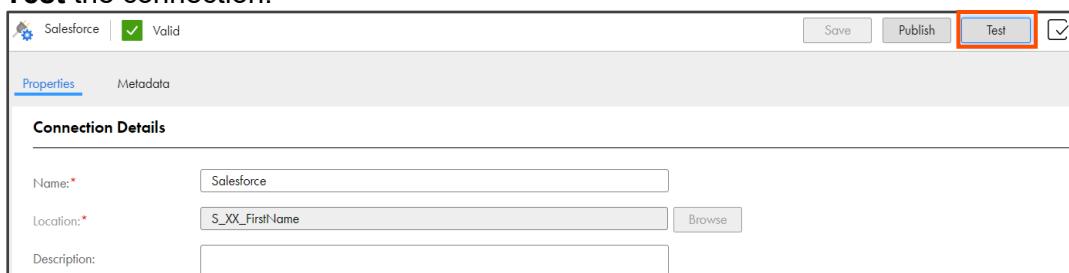
**Note:** When you reset the security token, it is mailed to your registered e-mail account. This security token validates the Salesforce connection properties and configures the connection.

19. Copy the Security Token from the e-mail and paste it in the Security Token field in IICS.

| Name             | Value   |
|------------------|---|
| User Name *      | rtrivedi@informatica.com                          |
| Password *       | • • • • • • • •                                   |
| Security Token * | • • • • • • • • • • • • • •                       |
| Service URL *    | https://login.salesforce.com/services/Soap/c/56.0 |

20. **Save** the connection.

21. **Test** the connection.



Salesforce | Valid

Properties      Metadata

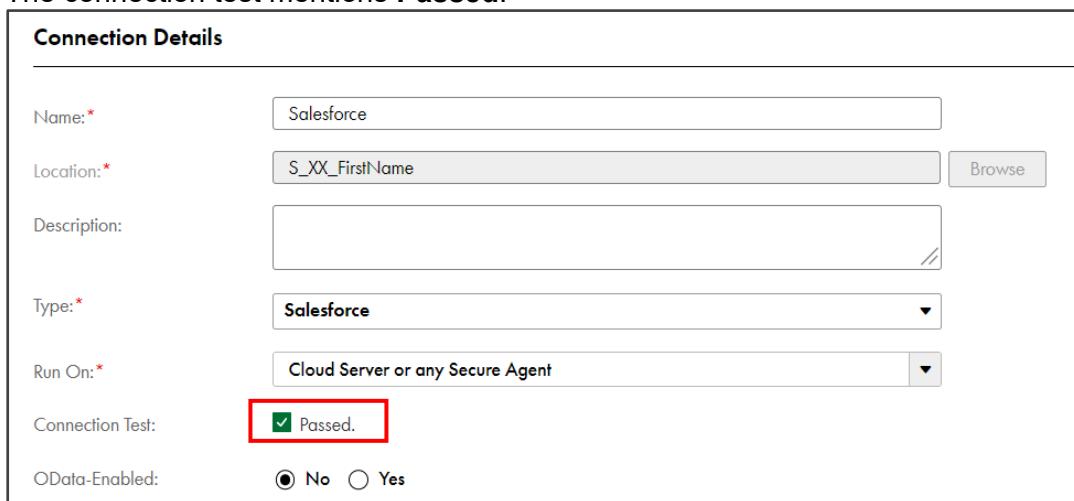
**Connection Details**

Name: \* Salesforce

Location: \* S\_XX\_FirstName

Description:

The connection test mentions **Passed**.



**Connection Details**

Name: \* Salesforce

Location: \* S\_XX\_FirstName

Description:

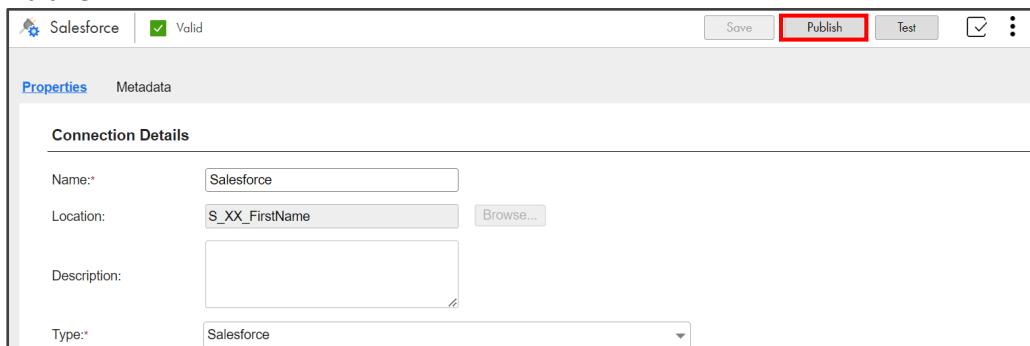
Type: \* Salesforce

Run On: \* Cloud Server or any Secure Agent

Connection Test:  Passed.

OData-Enabled:  No  Yes

22. **Publish** the connection.



Salesforce | Valid

Properties      Metadata

**Connection Details**

Name: \* Salesforce

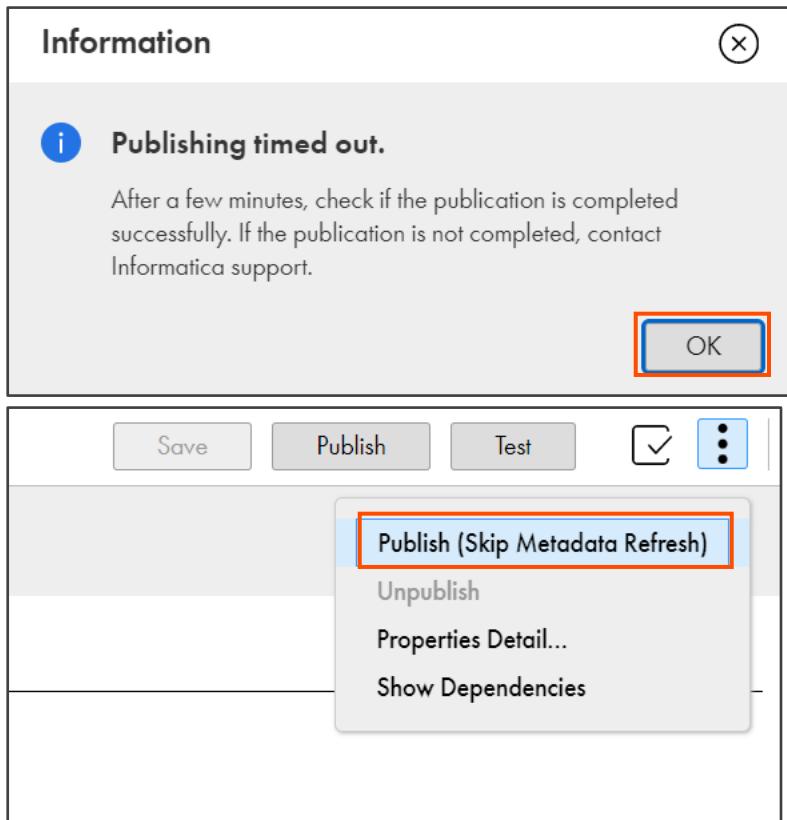
Location: \* S\_XX\_FirstName

Description:

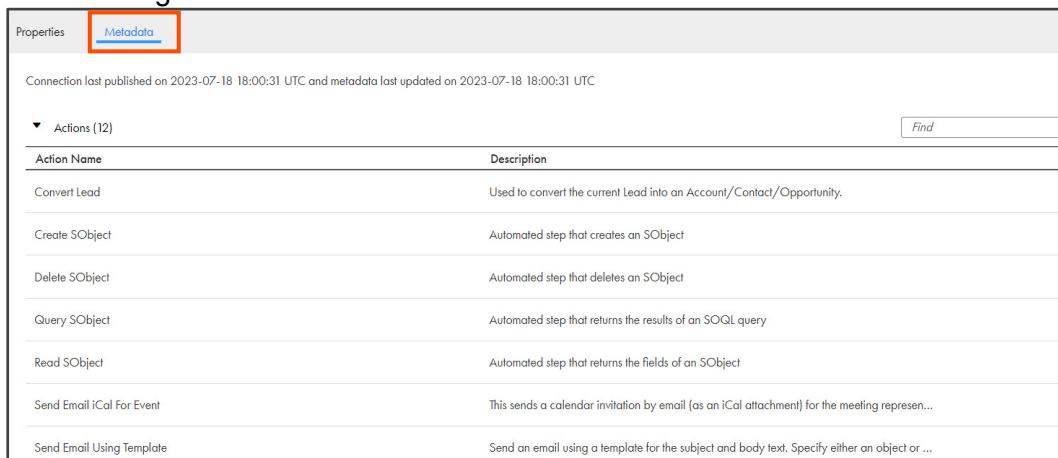
Type: \* Salesforce

**Note:** Sometimes, publishing takes a very long time. Wait for 5-6 minutes for the connection to be published.

If you get the "Publishing timed out" error, click **OK**, and then use the "Publish" (**Skip Metadata Refresh**) option to publish.



- After you publish the connection, the **Metadata** tab lists all the process objects that the connection generates.



The screenshot shows the "Metadata" tab of a connection's properties. At the top, it says "Connection last published on 2023-07-18 18:00:31 UTC and metadata last updated on 2023-07-18 18:00:31 UTC". Below this is a table titled "Actions (12)". The table has two columns: "Action Name" and "Description". The actions listed are:

| Action Name               | Description   |
|---------------------------|---|
| Convert Lead              | Used to convert the current Lead into an Account/Contact/Opportunity.                         |
| Create SObject            | Automated step that creates an SObject  |
| Delete SObject            | Automated step that deletes an SObject  |
| Query SObject             | Automated step that returns the results of an SOQL query                                      |
| Read SObject              | Automated step that returns the fields of an SObject  |
| Send Email iCal For Event | This sends a calendar invitation by email (as an iCal attachment) for the meeting represen... |
| Send Email Using Template | Send an email using a template for the subject and body text. Specify either an object or ... |

- Close** the Salesforce tab.

*This concludes the lab.*

## Module 7: App Connections

### Lab 7-6: Create a Kafka Connection

#### **Overview:**

Kafka is a distributed publish-subscribe messaging system that is fast, scalable, and durable. It stores messages in the form of topics. Producers write data to a topic and consumers read data from the topic. Kafka topics are partitioned and replicated across multiple nodes thereby allowing distributed processing. Kafka uses ZooKeeper to manage the Kafka cluster.

In this lab, you will create a Kafka connection using the built-in Kafka connector and use it in a process to read and write data.

#### **Objectives:**

- Start Apache ZooKeeper and Kafka server
- Create a topic
- Create a Kafka connection
- Publish the connection
- Create and publish a process to write data to a topic

#### **Duration:**

35 Minutes

---

#### **Tasks**

**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

## IMPORTANT – QUICK CHECKS:

- For this lab, you have already installed Kafka Server and Apache ZooKeeper as part of the Getting Started labs on your machine.
- In the **C:\Users\Administrator\Documents\Kafka\kafka\config** folder (or, in the Kafka config folder where you have installed Kafka Server), open the **server.properties** file. Scroll down and ensure that the listeners field value is as mentioned below:

**listeners= PLAINTEXT://127.0.0.1:9092**

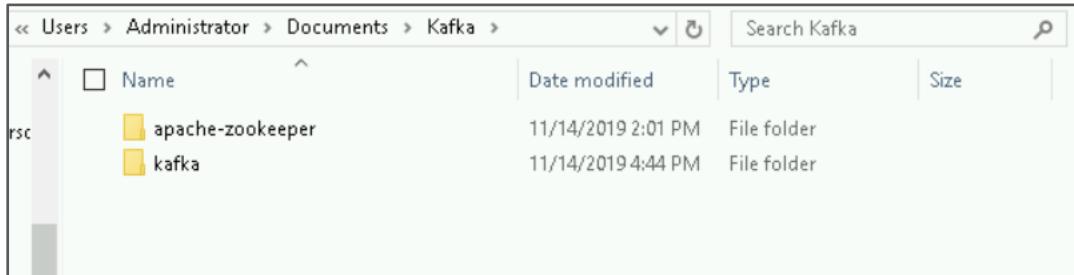
```

22
23 ##### Socket Server Settings #####
24
25 # The address the socket server listens on. It will get the value returned from
26 # java.net.InetAddress.getCanonicalHostName() if not configured.
27 # FORMAT:
28 #   listeners = listener_name://host_name:port
29 # EXAMPLE:
30 #   listeners = PLAINTEXT://your.host.name:9092
31 listeners= PLAINTEXT://127.0.0.1:9092
32
33 # Hostname and port the broker will advertise to producers and consumers. If not set,
34 # it uses the value for "listeners" if configured. Otherwise, it will use the value
35 # returned from java.net.InetAddress.getCanonicalHostName().
36 #advertised.listeners=PLAINTEXT://your.host.name:9092
37
38 # Maps listener names to security protocols, the default is for them to be the same. See the confi
39 #listener.security.protocol.map=PLAINTEXT:PLAINTEXT,SSL:SSL,SASL_PLAINTEXT:SASL_PLAINTEXT,SASL_SS
40

```

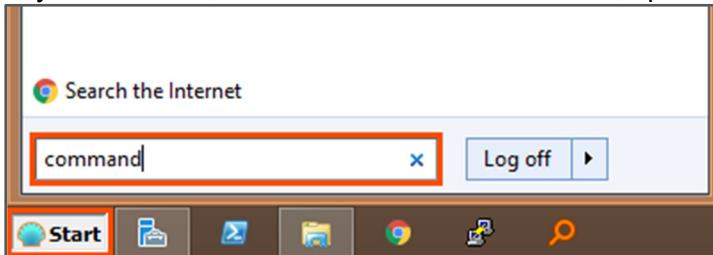
(The above-mentioned port and IP address are the default values where the Kafka Server will be installed. They will change in the scenarios where you do not have the default port as free. In such cases, it will take the next available free port.)

- Navigate to the Kafka installation directory **C:\Users\Administrator\Documents\Kafka\**, and make sure you DO NOT see a folder named **kafkaData**. In case you see the kafkaData folder, delete it.

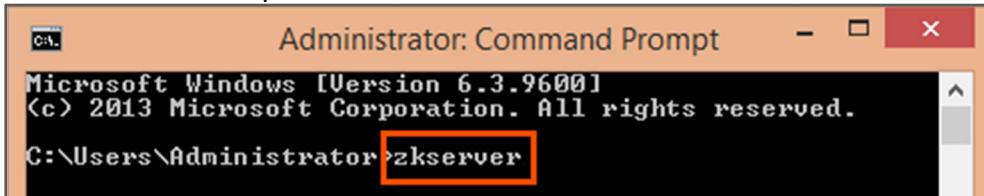


## Start Apache ZooKeeper and Kafka Server

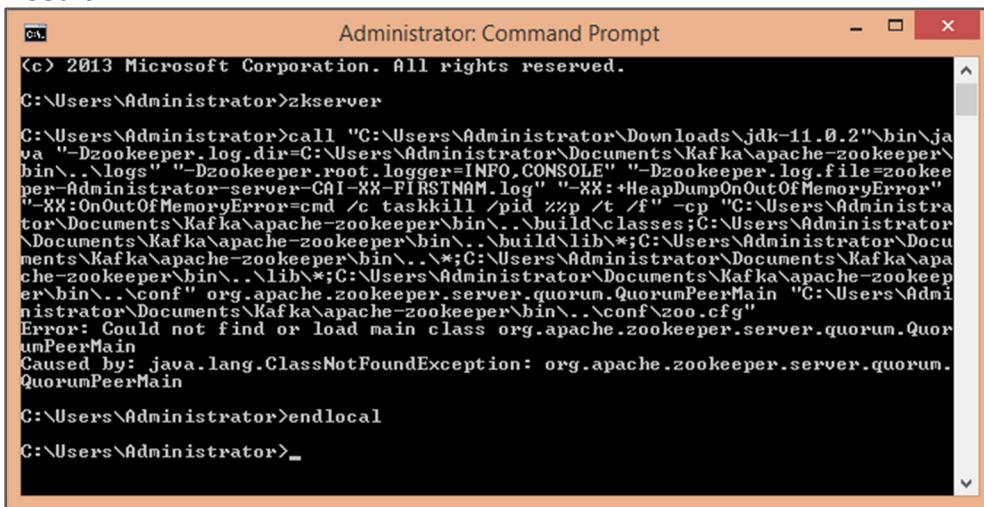
1. In your machine, from the Windows Start menu, open the **Command Prompt**.



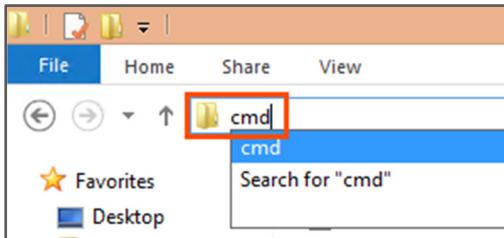
2. To start the ZooKeeper server, enter the command **zkserver**, and hit **Enter**.



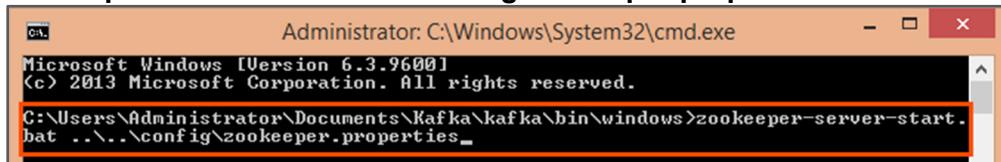
**Result:**



3. From the **C:\Users\Administrator\Documents\Kafka\kafka\bin\windows** folder, open a command prompt.



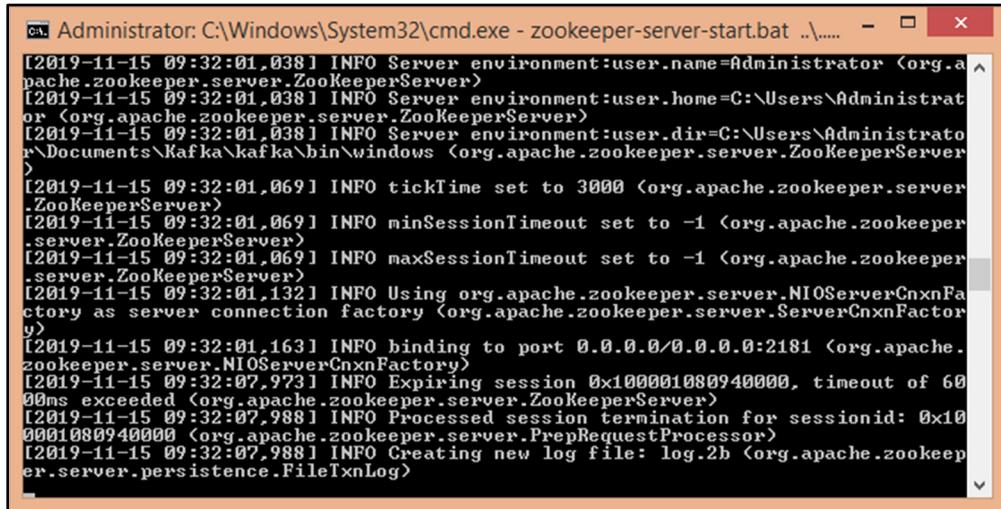
4. To execute the ZooKeeper server internally, enter the following command:  
**zookeeper-server-start.bat ..\..\config\zookeeper.properties**



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>zookeeper-server-start.bat ..\..\config\zookeeper.properties
```

**Result:**

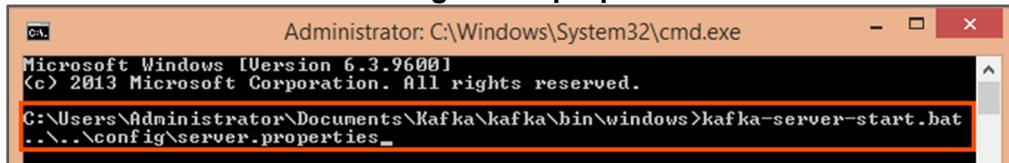


```
[2019-11-15 09:32:01,038] INFO Server environment:user.name=Administrator <org.apache.zookeeper.server.ZooKeeperServer>
[2019-11-15 09:32:01,038] INFO Server environment:user.home=C:\Users\Administrator <org.apache.zookeeper.server.ZooKeeperServer>
[2019-11-15 09:32:01,038] INFO Server environment:user.dir=C:\Users\Administrator\Documents\Kafka\kafka\bin\windows <org.apache.zookeeper.server.ZooKeeperServer>
[2019-11-15 09:32:01,069] INFO tickTime set to 3000 <org.apache.zookeeper.server.ZooKeeperServer>
[2019-11-15 09:32:01,069] INFO minSessionTimeout set to -1 <org.apache.zookeeper.server.ZooKeeperServer>
[2019-11-15 09:32:01,069] INFO maxSessionTimeout set to -1 <org.apache.zookeeper.server.ZooKeeperServer>
[2019-11-15 09:32:01,132] INFO Using org.apache.zookeeper.server.NIOServerCnxnFactory as server connection factory <org.apache.zookeeper.server.ServerCnxnFactory>
[2019-11-15 09:32:01,163] INFO binding to port 0.0.0.0/0.0.0.0:2181 <org.apache.zookeeper.server.NIOServerCnxnFactory>
[2019-11-15 09:32:07,973] INFO Expiring session 0x100001080940000, timeout of 600ms exceeded <org.apache.zookeeper.server.ZooKeeperServer>
[2019-11-15 09:32:07,988] INFO Processed session termination for sessionid: 0x100001080940000 <org.apache.zookeeper.server.PrepRequestProcessor>
[2019-11-15 09:32:07,988] INFO Creating new log file: log.2b <org.apache.zookeeper.persistence.FileTxnLog>
```

**Important:** Do not close any of the command prompts till the end of this lab.

5. In the same folder, open another command prompt, and enter the following command to start the kafka server:

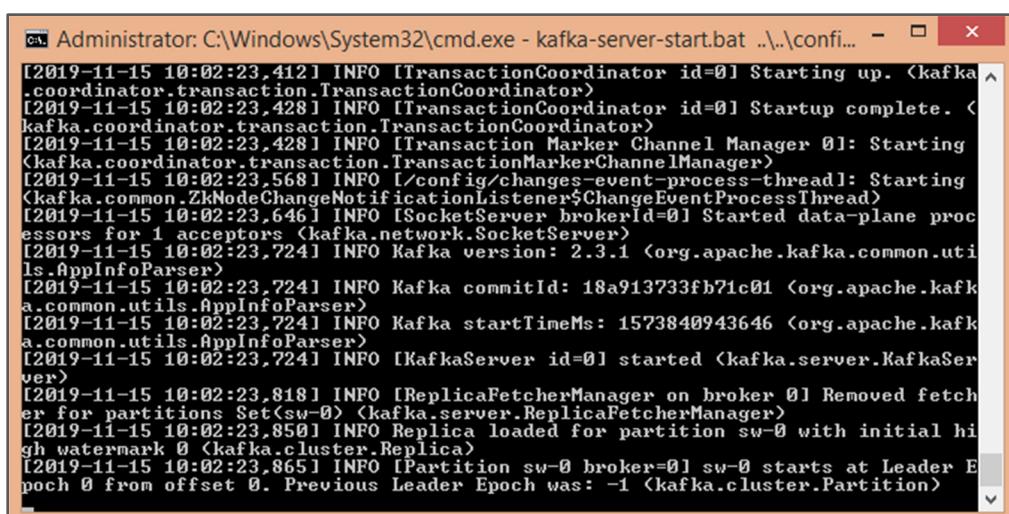
**kafka-server-start.bat ..\..\config\server.properties**



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>kafka-server-start.bat ..\..\config\server.properties
```

**Result:**

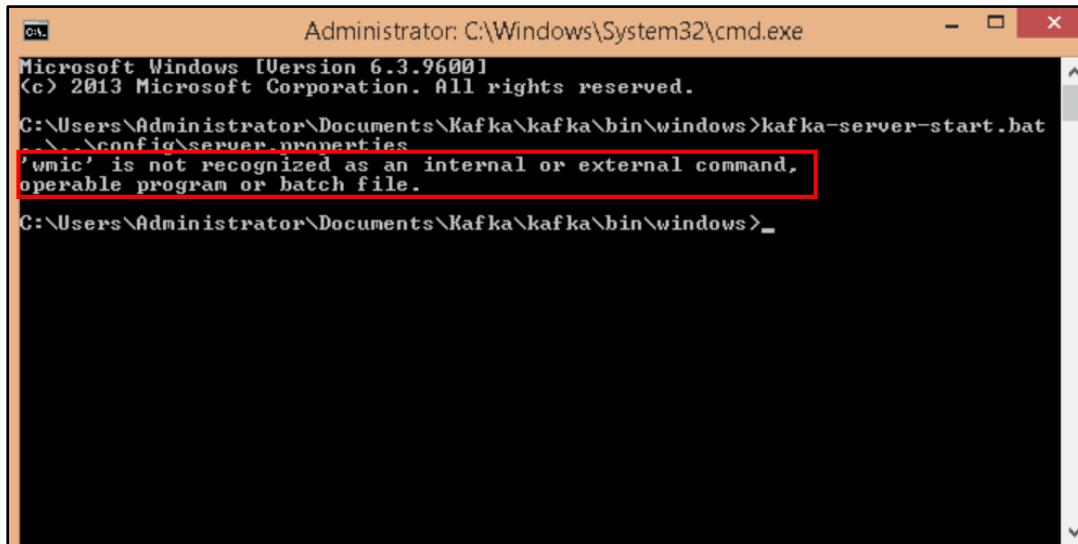


```
[2019-11-15 10:02:23,412] INFO [TransactionCoordinator id=0] Starting up. <kafka.coordinator.transaction.TransactionCoordinator>
[2019-11-15 10:02:23,428] INFO [TransactionCoordinator id=0] Startup complete. <kafka.coordinator.transaction.TransactionCoordinator>
[2019-11-15 10:02:23,428] INFO [Transaction Marker Channel Manager @1]: Starting <kafka.coordinator.transaction.TransactionMarkerChannelManager>
[2019-11-15 10:02:23,568] INFO [/config/changes-event-process-thread]: Starting <kafka.common.ZkNodeChangeNotificationListener$ChangeEventProcessThread>
[2019-11-15 10:02:23,646] INFO [SocketServer brokerId=0] Started data-plane processors for 1 acceptors <kafka.network.SocketServer>
[2019-11-15 10:02:23,724] INFO Kafka version: 2.3.1 <org.apache.kafka.common.utils.AppInfoParser>
[2019-11-15 10:02:23,724] INFO Kafka commitId: 18a913733fb71c01 <org.apache.kafka.common.utils.AppInfoParser>
[2019-11-15 10:02:23,724] INFO Kafka startTimeMs: 1573840943646 <org.apache.kafka.common.utils.AppInfoParser>
[2019-11-15 10:02:23,724] INFO [KafkaServer id=0] started <kafka.server.KafkaServer>
[2019-11-15 10:02:23,818] INFO [ReplicaFetcherManager on broker @1] Removed fetcher for partitions Set<sw-0> <kafka.server.ReplicaFetcherManager>
[2019-11-15 10:02:23,850] INFO Replica loaded for partition sw-0 with initial high watermark 0 <kafka.cluster.Replica>
[2019-11-15 10:02:23,865] INFO [Partition sw-0 broker=@1] sw-0 starts at Leader Epoch 0 from offset 0. Previous Leader Epoch was: -1 <kafka.cluster.Partition>
```

**Important:** Do not close any of the command prompts till the end of this lab.

**Notes:**

Sometimes, you can notice the following error while executing the above command:  
**ERROR: "wmic is not recognized as an internal or external command, operable program or batch file."**



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>kafka-server-start.bat
..\..\config\server.properties
'wmic' is not recognized as an internal or external command,
operable program or batch file.

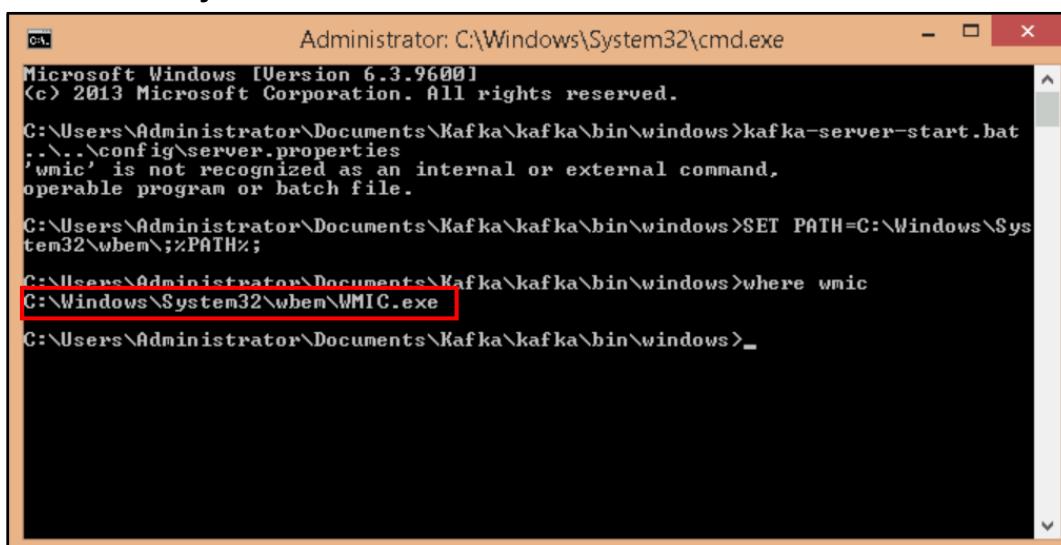
C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>_
```

If you get the above error, then:

- Run the following command in the same command window for adding the folder containing wmic to PATH environment variable:  
**SET PATH=C:\Windows\System32\wbem;%PATH%;**
- After setting the path variable using the command above, run the following command to confirm whether *wmic* is picked properly:  
**where wmic**

The above command should return the following as output:

**C:\Windows\System32\wbem\WMIC.exe**



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

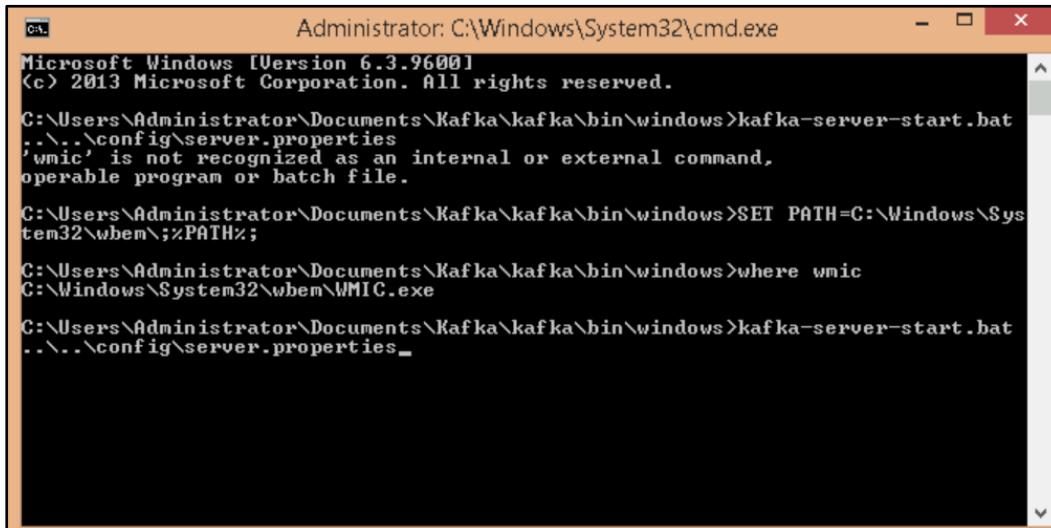
C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>kafka-server-start.bat
..\..\config\server.properties
'wmic' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>SET PATH=C:\Windows\System32\wbem;%PATH%;

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>where wmic
C:\Windows\System32\wbem\WMIC.exe

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>_
```

- c. Do Not close the command prompt. In the same command prompt, enter the kafka-server-start.bat ..\..\config\server.properties command and press Enter.



```

Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>kafka-server-start.bat
..\..\config\server.properties
'wmic' is not recognized as an internal or external command,
operable program or batch file.

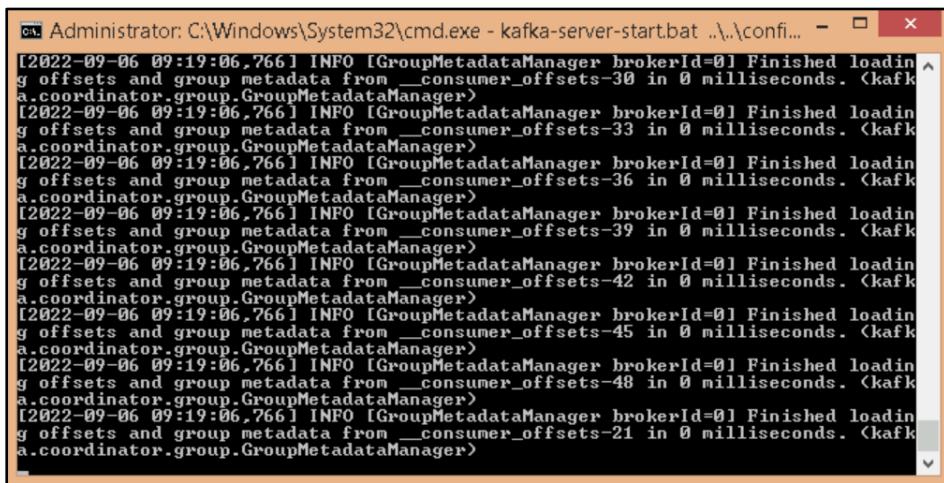
C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>SET PATH=C:\Windows\System32\whem\;%PATH%;

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>where wmic
C:\Windows\System32\whem\WMIC.exe

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>kafka-server-start.bat
..\..\config\server.properties_

```

**Result:**

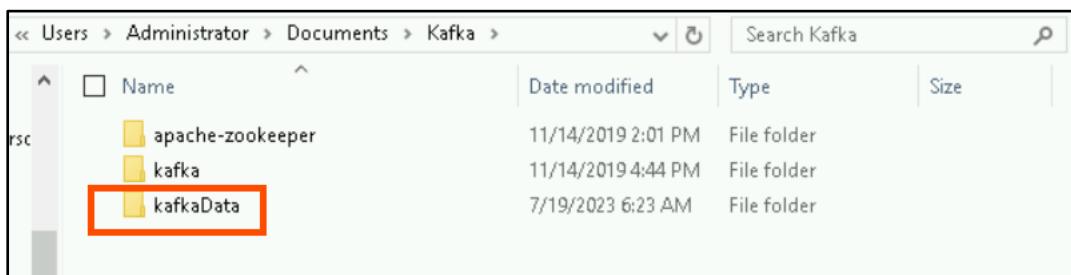


```

Administrator: C:\Windows\System32\cmd.exe - kafka-server-start.bat ..\..\config\server.properties
[2022-09-06 09:19:06,766] INFO [GroupMetadataManager brokerId=0] Finished loading offsets and group metadata from __consumer_offsets-30 in 0 milliseconds. <kafka.coordinator.group.GroupMetadataManager>
[2022-09-06 09:19:06,766] INFO [GroupMetadataManager brokerId=0] Finished loading offsets and group metadata from __consumer_offsets-33 in 0 milliseconds. <kafka.coordinator.group.GroupMetadataManager>
[2022-09-06 09:19:06,766] INFO [GroupMetadataManager brokerId=0] Finished loading offsets and group metadata from __consumer_offsets-36 in 0 milliseconds. <kafka.coordinator.group.GroupMetadataManager>
[2022-09-06 09:19:06,766] INFO [GroupMetadataManager brokerId=0] Finished loading offsets and group metadata from __consumer_offsets-39 in 0 milliseconds. <kafka.coordinator.group.GroupMetadataManager>
[2022-09-06 09:19:06,766] INFO [GroupMetadataManager brokerId=0] Finished loading offsets and group metadata from __consumer_offsets-42 in 0 milliseconds. <kafka.coordinator.group.GroupMetadataManager>
[2022-09-06 09:19:06,766] INFO [GroupMetadataManager brokerId=0] Finished loading offsets and group metadata from __consumer_offsets-45 in 0 milliseconds. <kafka.coordinator.group.GroupMetadataManager>
[2022-09-06 09:19:06,766] INFO [GroupMetadataManager brokerId=0] Finished loading offsets and group metadata from __consumer_offsets-48 in 0 milliseconds. <kafka.coordinator.group.GroupMetadataManager>
[2022-09-06 09:19:06,766] INFO [GroupMetadataManager brokerId=0] Finished loading offsets and group metadata from __consumer_offsets-21 in 0 milliseconds. <kafka.coordinator.group.GroupMetadataManager>

```

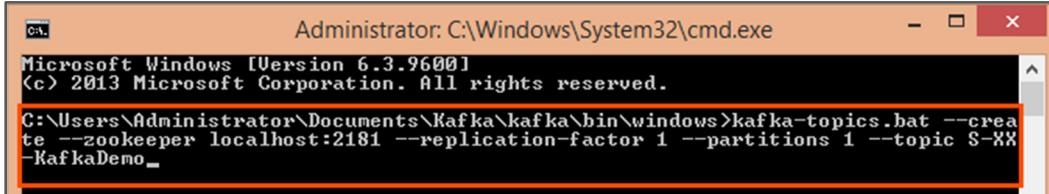
6. Once the Kafka server has started, navigate to the following Kafka directory **C:\Users\Administrator\Documents\Kafka**, and verify that the **kafkaData** folder is created.



**Note:** The above directory may change if you do not install Kafka in the default folder.

## Create a Topic

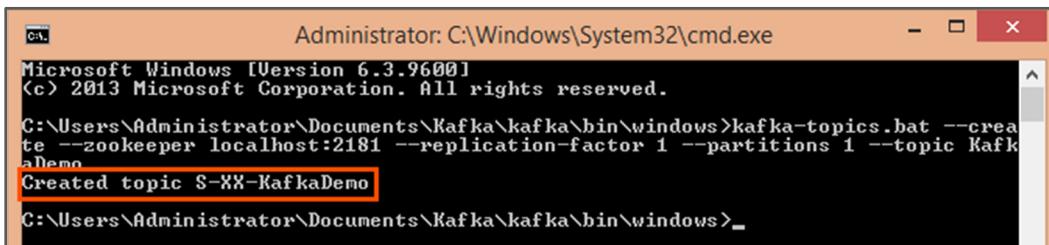
7. In the same folder (**C:\Users\Administrator\Documents\Kafka\kafka\bin\windows**), open another command prompt, and enter the following command:  
**kafka-topics.bat --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic S-XX-KafkaDemo**



```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>kafka-topics.bat --crea
te --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic S-XX
-KafkaDemo
```

### Result:



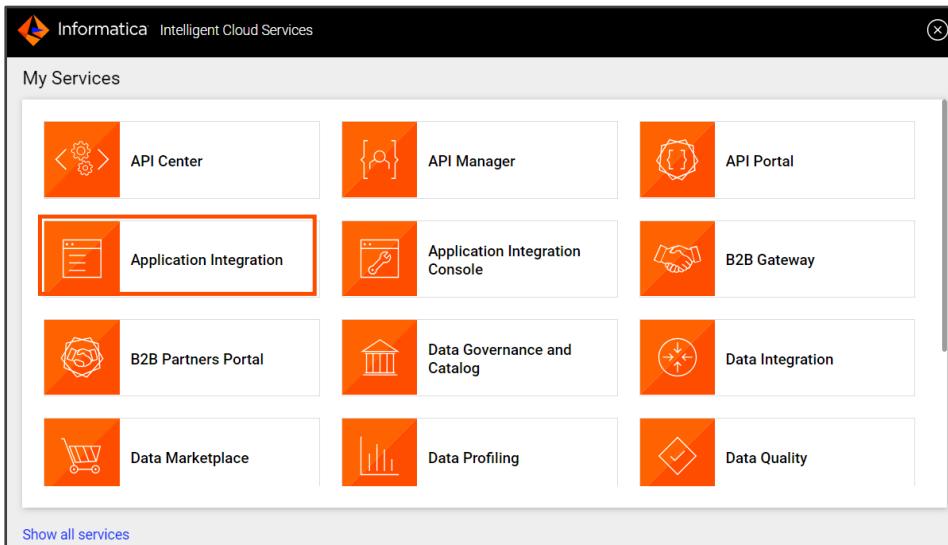
```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>kafka-topics.bat --crea
te --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic Kafka
Demo
Created topic S-XX-KafkaDemo

C:\Users\Administrator\Documents\Kafka\kafka\bin\windows>_
```

## Create a Kafka Connection

8. Log in to IICS and access the **Application Integration** service.

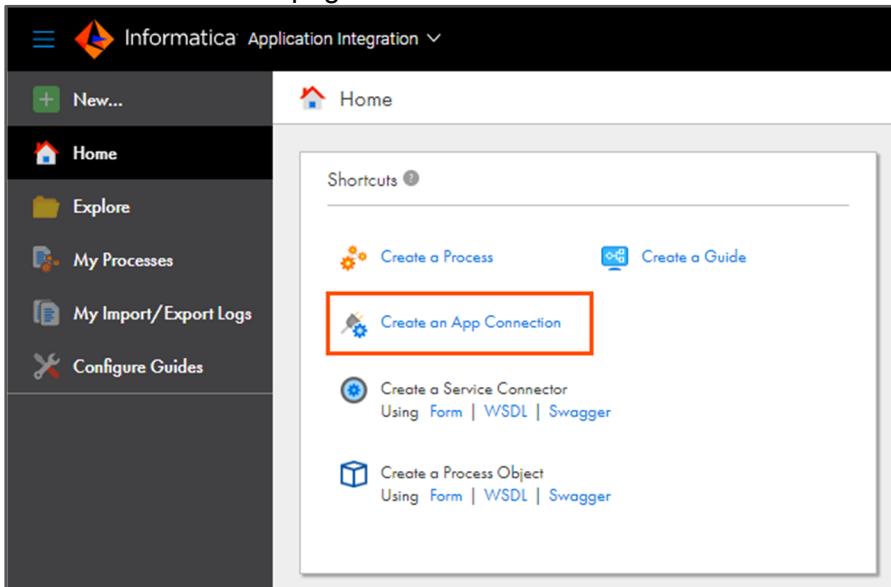


The screenshot shows the IICS My Services interface. It displays a grid of nine service icons:

- API Center
- API Manager
- API Portal
- Application Integration** (highlighted)
- Application Integration Console
- B2B Gateway
- B2B Partners Portal
- Data Governance and Catalog
- Data Integration
- Data Marketplace
- Data Profiling
- Data Quality

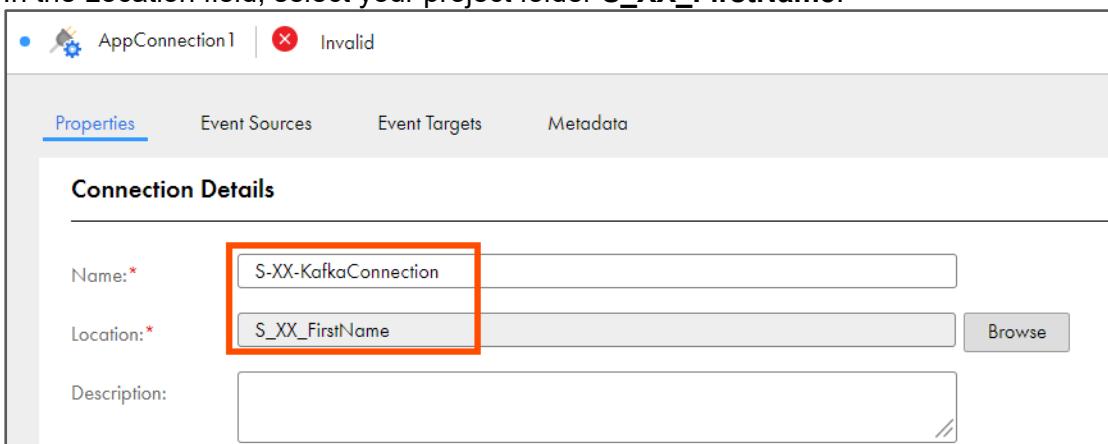
At the bottom left, there is a link labeled "Show all services".

9. Create a new **App Connection**. You can also create a new App Connection using the shortcut on the Home page.



10. In the Name field, enter **S-XX-KafkaConnection**.

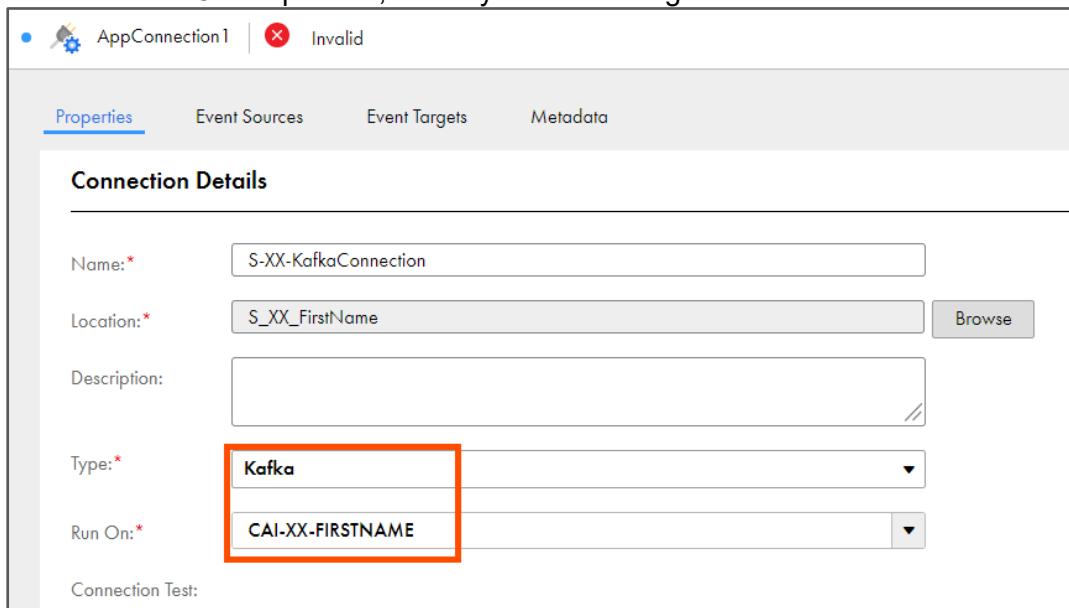
11. In the Location field, select your project folder **S\_XX\_FirstName**.



The screenshot shows the 'Properties' tab for an app connection named 'AppConnection1'. The 'Invalid' status is shown. Below, the 'Connection Details' section has tabs for 'Properties', 'Event Sources', 'Event Targets', and 'Metadata'. The 'Properties' tab is active. The 'Name:' field contains 'S-XX-KafkaConnection' and the 'Location:' field contains 'S\_XX\_FirstName'. Both fields are highlighted with red boxes.

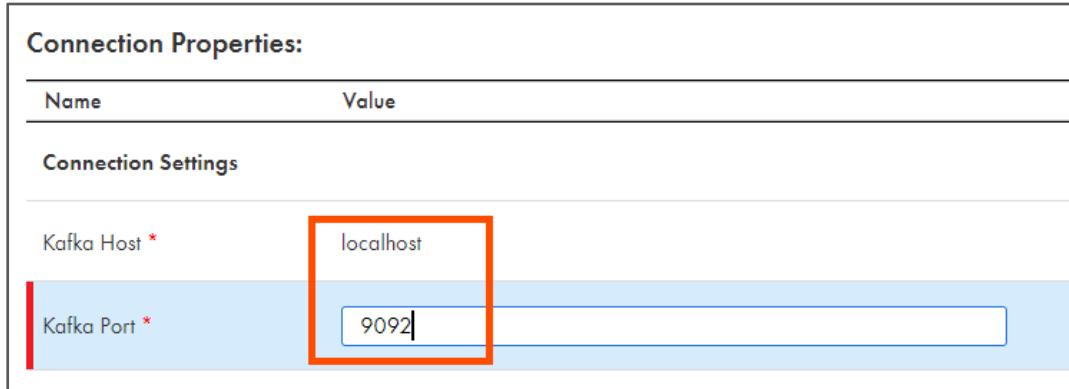
12. From the Type drop-down, select **Kafka**.

13. From the Run On drop-down, select your Secure Agent.



The screenshot shows the 'Properties' tab of the AppConnection1 configuration interface. The 'Run On:' dropdown is highlighted with a red box, showing the value 'CAI-XX-FIRSTNAME'. Other fields visible include 'Name:' (S-XX-KafkaConnection), 'Location:' (S\_XX\_FirstName), and 'Type:' (Kafka).

14. In the Connection Properties section, click inside the value field and enter Kafka Host as **localhost**, and Kafka Port as **9092**.

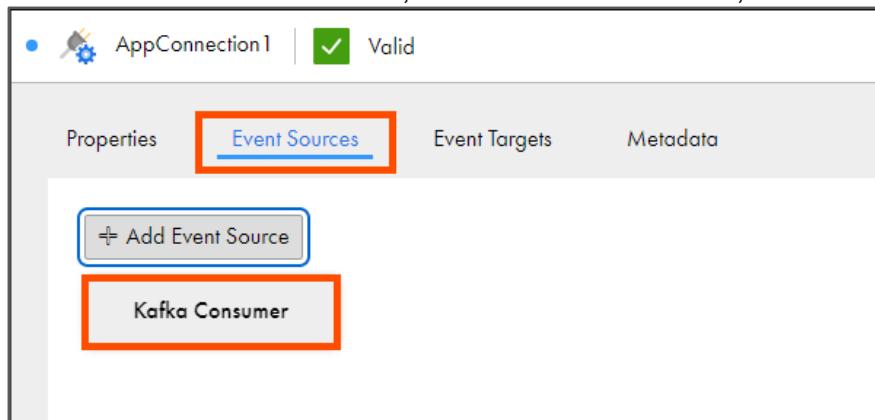


The screenshot shows the 'Connection Properties' section. Under 'Connection Settings', the 'Kafka Host' field contains 'localhost' and the 'Kafka Port' field contains '9092', both of which are highlighted with red boxes.

**Note:** This is the same port number that you checked in the notes section at the beginning of this lab.

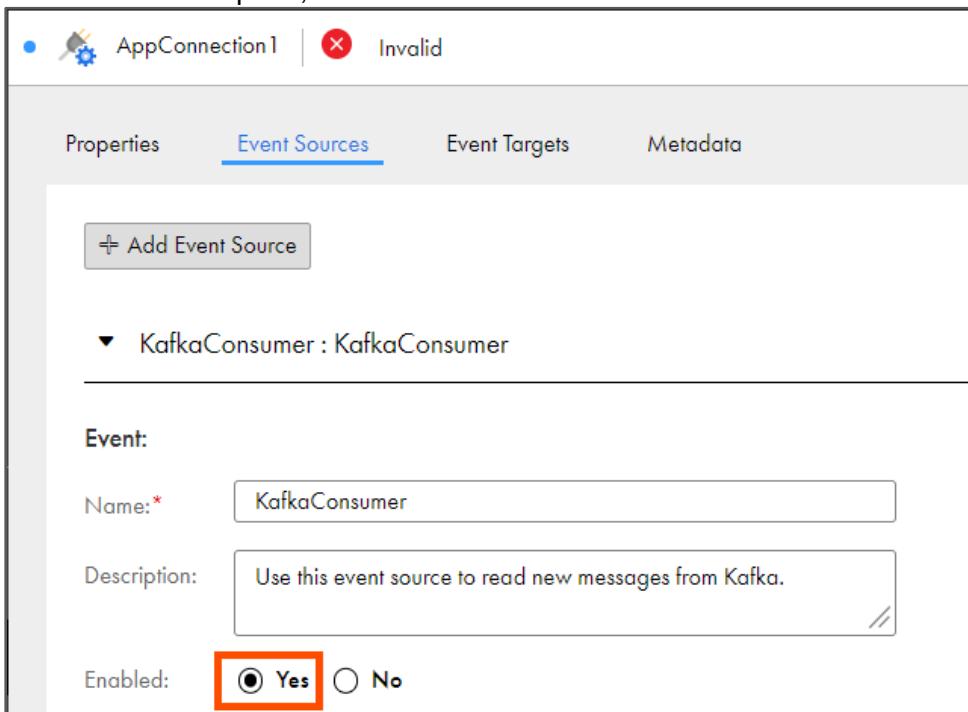
15. To configure the Event Sources, click the **Event Sources** tab.

16. To add different Event Sources, click **Add Event Source**, and select **Kafka Consumer**.



The screenshot shows the 'Event Sources' tab selected. A blue box highlights the 'Add Event Source' button. Another blue box highlights the 'Kafka Consumer' option in the list below.

17. For the **Name** and **Description** fields, retain the default values.
18. For the Enabled option, retain **Yes**.



The screenshot shows the configuration interface for an AppConnection named "AppConnection1". The "Event Sources" tab is selected. A red box highlights the "Enabled" field, which is set to "Yes".

Properties    Event Sources    Event Targets    Metadata

Add Event Source

KafkaConsumer : KafkaConsumer

Event:

Name: \* KafkaConsumer

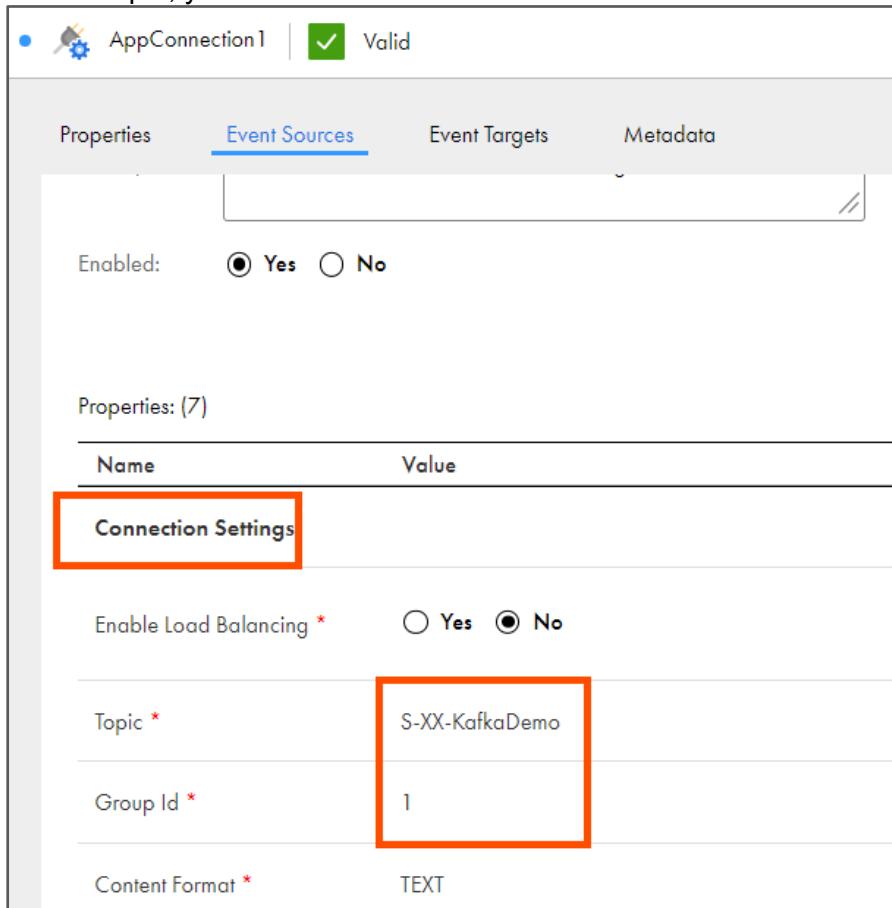
Description: Use this event source to read new messages from Kafka.

Enabled:  Yes  No

**Note:** When you select No, the event source is disabled until you are ready to use it. If you select Yes, the event source is available immediately after it is published.

19. In the Connection Settings section, in the Topic field, enter **S-XX-KafkaDemo**.
20. In the Group Id field, enter a Group Id. For this lab, you can enter your date of birth.

For example, you can enter the value 1.



AppConnection1 | Valid

Properties Event Sources Event Targets Metadata

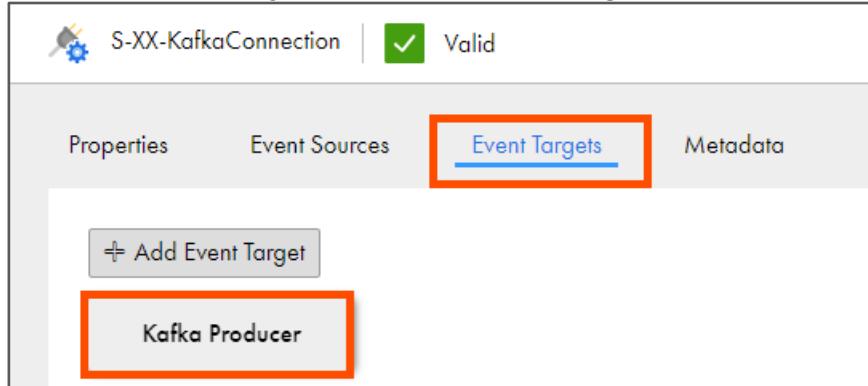
Enabled:  Yes  No

Properties: (7)

| Name                    | Value   |
|-------------------------|---|
| Connection Settings     |   |
| Enable Load Balancing * | <input type="radio"/> Yes <input checked="" type="radio"/> No |
| Topic *                 | S-XX-KafkaDemo  |
| Group Id *              | 1   |
| Content Format *        | TEXT  |

21. Save the connection.

22. To add an Event Targets, click Add Event Target and select Kafka Producer.



S-XX-KafkaConnection | Valid

Properties Event Sources Event Targets Metadata

+ Add Event Target

Kafka Producer

23. In the Connection Settings section, in the Topic field, enter **S-XX-KafkaDemo**.

24. In the Group Id field, enter a Group Id. For this lab, you can enter your date of birth.

For example, you can enter the value 1.

S-XX-KafkaConnection |  Valid

Properties Event Sources Event Targets Metadata

▼ KafkaProducer : KafkaProducer

**Event:**

|              |   |
|--------------|---|
| Name:*       | KafkaProducer   |
| Description: | Use this event target to write new messages to Kafka. |

Properties: (5)

| Name                       | Value          |
|----------------------------|----------------|
| <b>Connection Settings</b> |                |
| Topic *                    | S-XX-KafkaDemo |
| Group Id *                 | 1              |
| Content Format *           | TEXT           |

**25. Save and test the connection.**

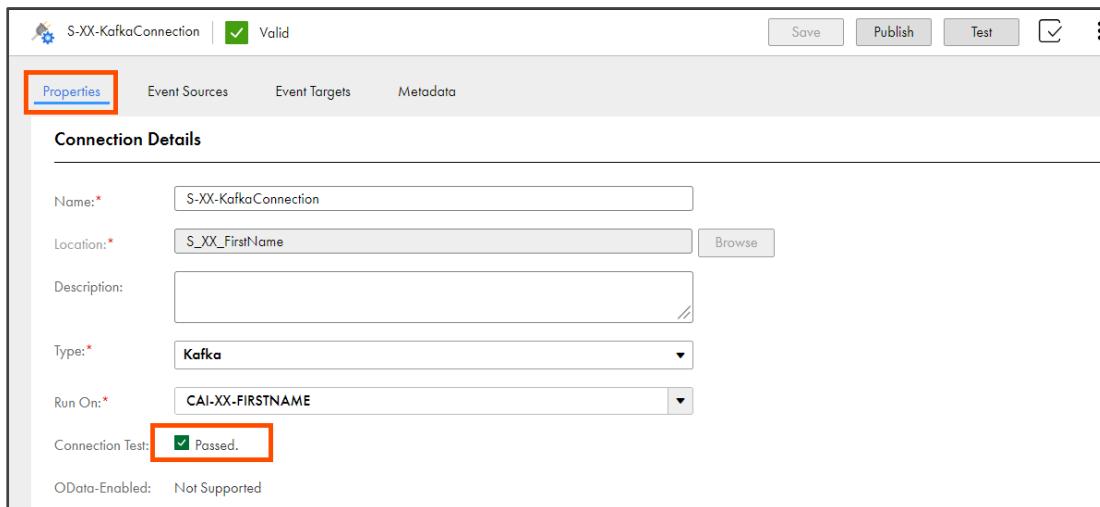
S-XX-KafkaConnection |  Valid

Save Publish **Test** :

Properties Event Sources Event Targets Metadata

Add Event Target

**Note:** In the Properties tab, observe that the Connection Test field mentions **Passed**.



S-XX-KafkaConnection |  Valid

Save Publish Test :

**Properties** Event Sources Event Targets Metadata

**Connection Details**

Name: \* S-XX-KafkaConnection

Location: \* S\_XX\_FirstName

Description:

Type: \* Kafka

Run On: \* CAI-XX-FIRSTNAME

Connection Test:  Passed.

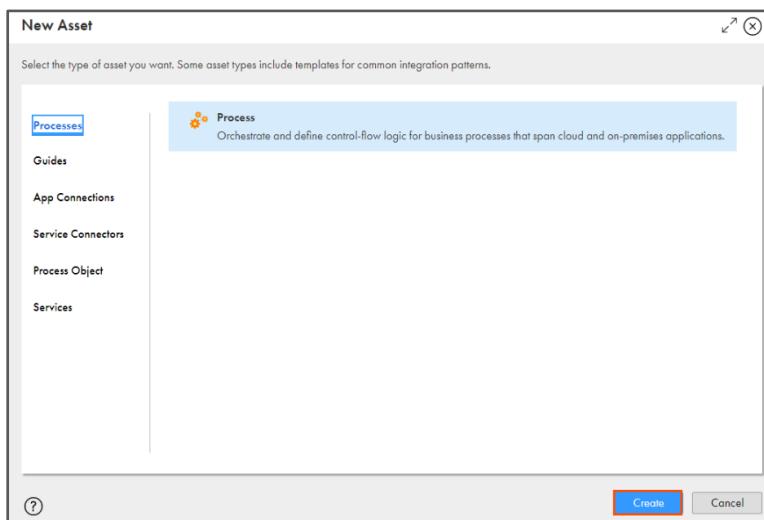
OData-Enabled: Not Supported

**Publish** the connection. You get a notification – ‘**The asset [S-XX-KafkaConnection] was published successfully**’.

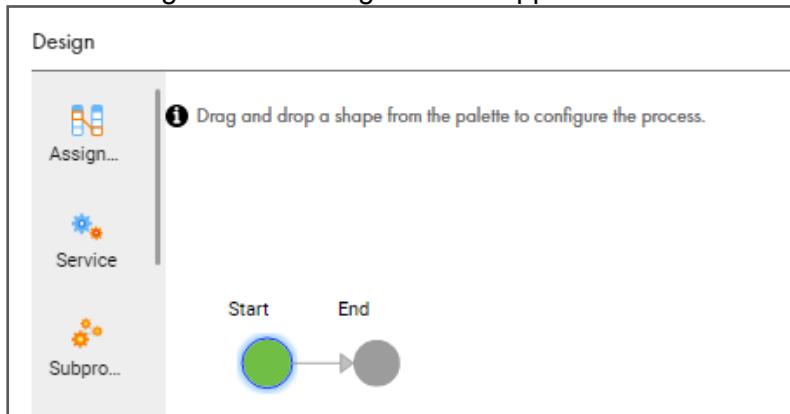
## Create a Write Process

You will now create a process to write a message to the topic.

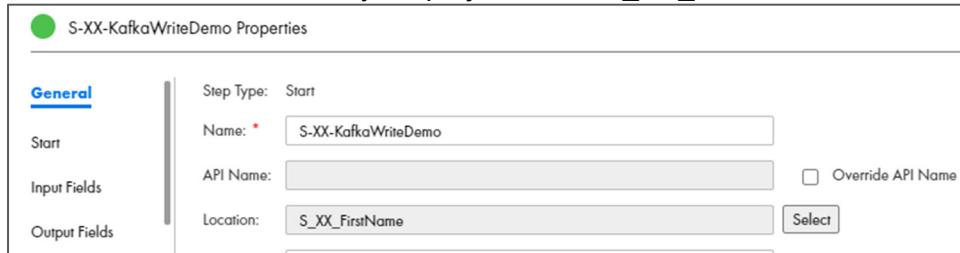
### 26. Create a new Process.



The following Process Design canvas appears with the Start and End steps.



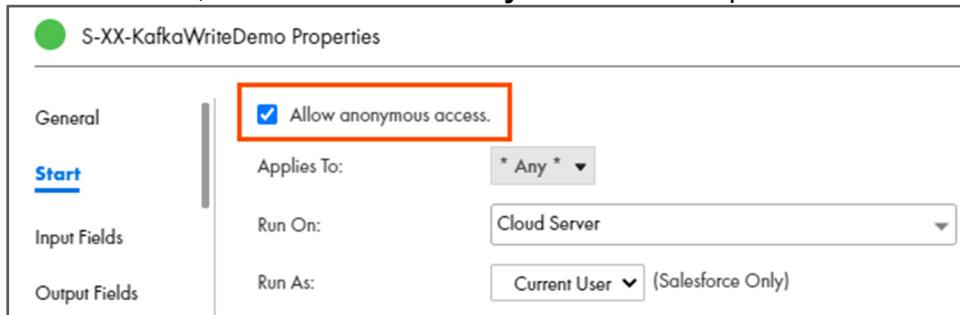
27. From the process canvas, select the **Start** step.
28. In the General tab, enter the name as **S-XX-KafkaWriteDemo**.
29. Skip the API Name option.
30. In the Location field, select your project folder **S\_XX\_FirstName**.



**S-XX-KafkaWriteDemo Properties**

|                |   |
|----------------|---|
| <b>General</b> | Step Type: Start<br>Name: * S-XX-KafkaWriteDemo<br>API Name: <input type="text"/><br>Location: S_XX_FirstName <input type="button" value="Select"/> |
| Start          |   |
| Input Fields   |   |
| Output Fields  |   |

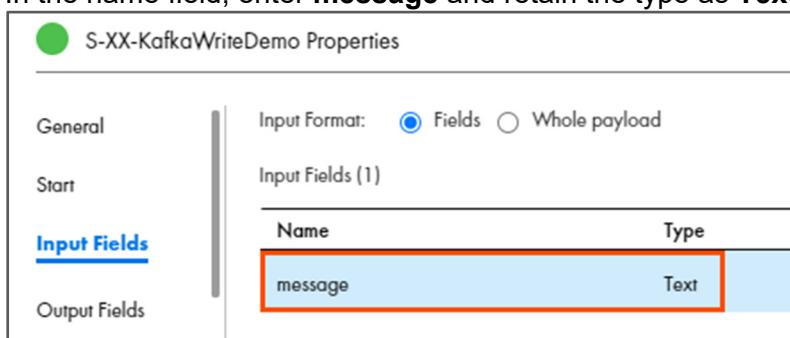
31. In the Start tab, select the **Allow anonymous access** option.



**S-XX-KafkaWriteDemo Properties**

|                |   |
|----------------|---|
| <b>General</b> | <input checked="" type="checkbox"/> Allow anonymous access.                           |
| <b>Start</b>   | Applies To: * Any *<br>Run On: Cloud Server<br>Run As: Current User (Salesforce Only) |
| Input Fields   |   |
| Output Fields  |   |

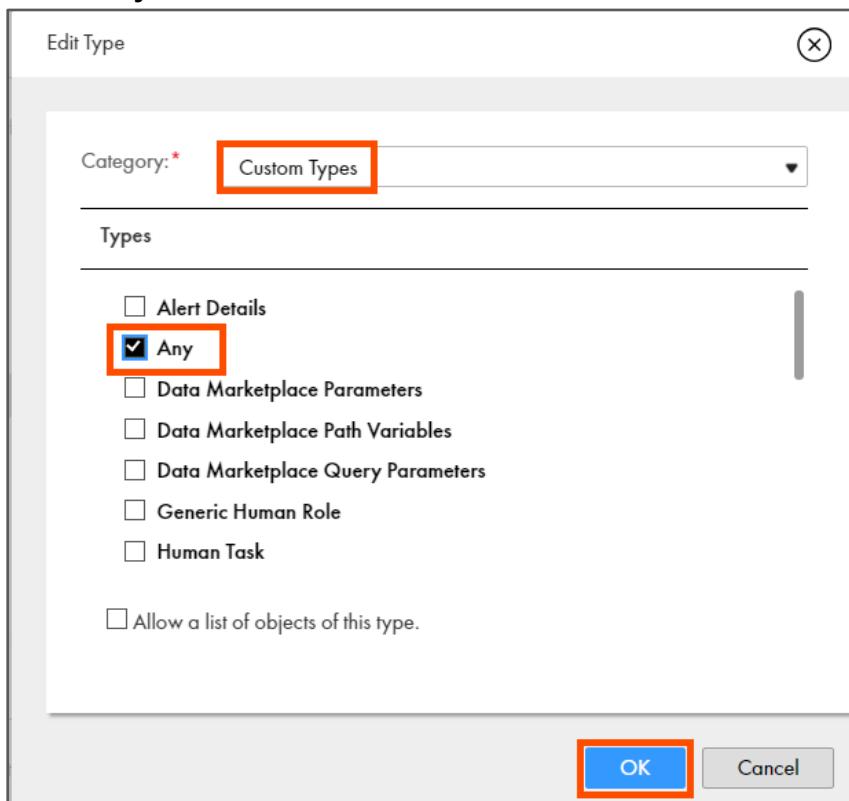
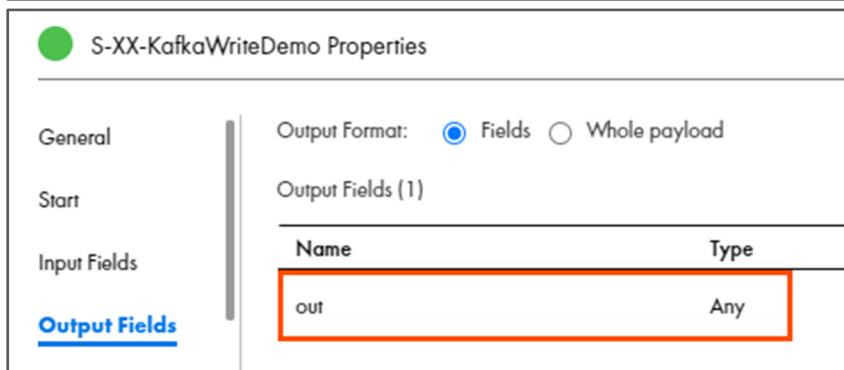
32. In the Input Fields tab, to add a new input field, click .
33. In the name field, enter **message** and retain the type as **Text**.



**S-XX-KafkaWriteDemo Properties**

| <b>General</b>      | Input Format: <input checked="" type="radio"/> Fields <input type="radio"/> Whole payload  |      |      |         |      |
|---------------------|--|------|------|---------|------|
| <b>Start</b>        |  |      |      |         |      |
| <b>Input Fields</b> | Input Fields (1)<br><table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>message</td> <td>Text</td> </tr> </tbody> </table> | Name | Type | message | Text |
| Name                | Type   |      |      |         |      |
| message             | Text   |      |      |         |      |
| Output Fields       |  |      |      |         |      |

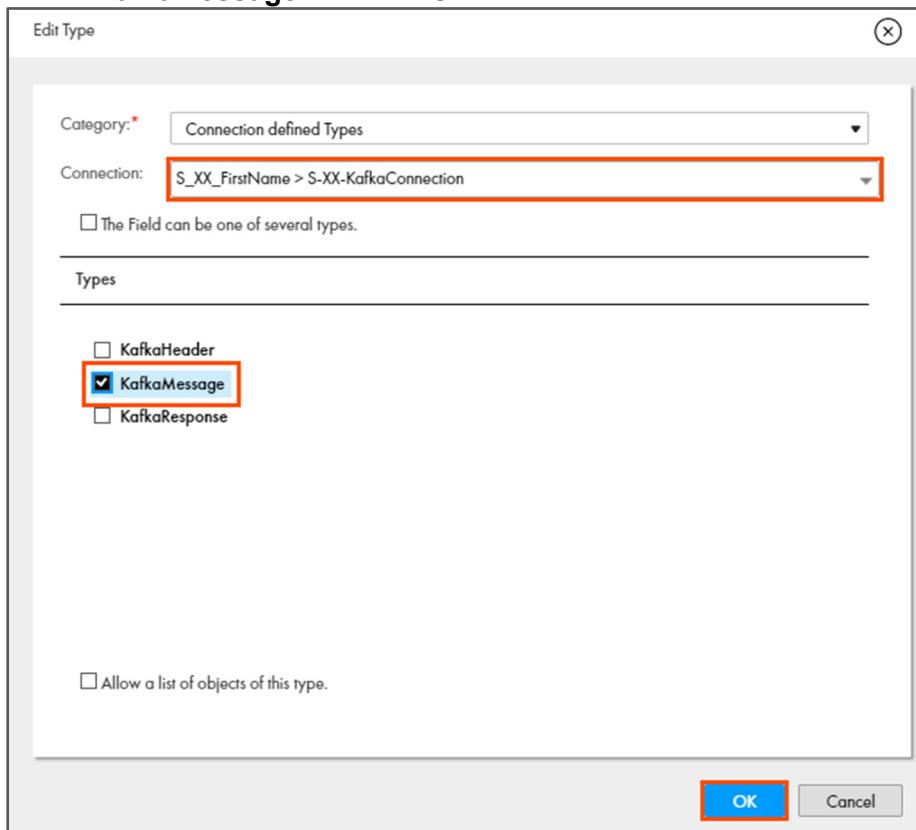
34. In the Output Fields tab, to add a new output field, click .
35. In the name field, enter **out**.
36. From the Type drop-down, select **More types** and from the Category drop-down, select **Custom Types**.
37. Select **Any** and click **OK**.

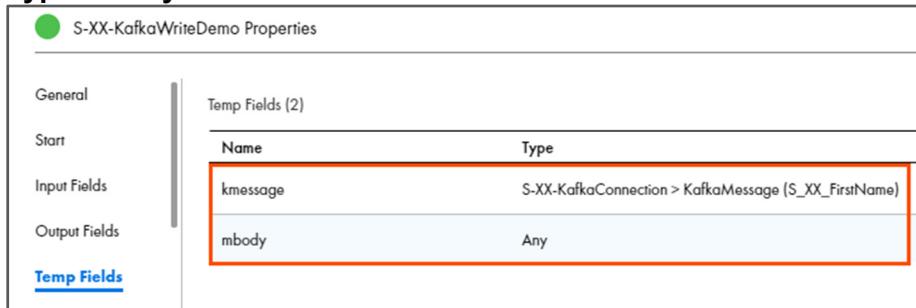
| Name | Type |
|------|------|
| out  | Any  |

38. In the Temp Fields tab, to add a new temp field, click .
39. In the Name field, enter **kmessage**.
40. From the Type field, select **More types** and from the Category field, select **Connection defined Types**.
41. From the Connection drop-down, from your project folder, and select **S-XX-KafkaConnection**.

42. Select **KafkaMessage** and click **OK**.

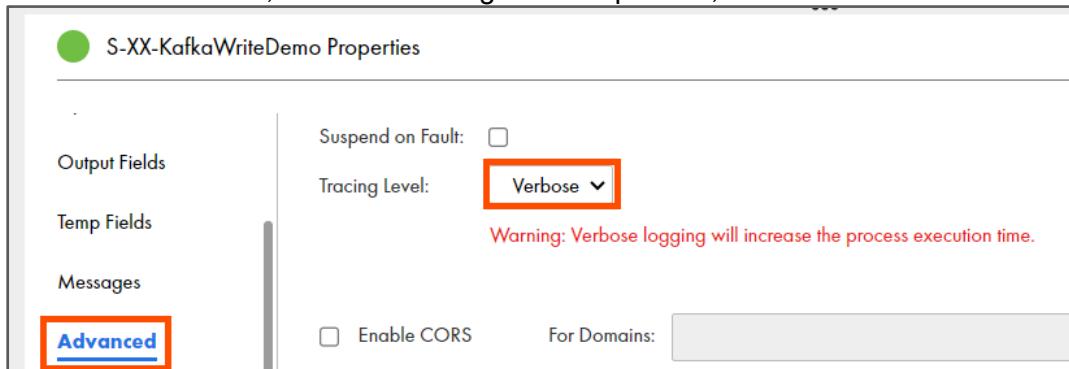


43. Similarly, add another temp field named **mbody**, and Type as **More Types > Custom Types > Any**.

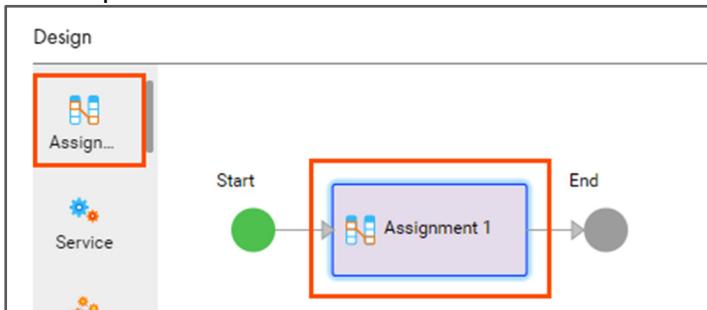


| Name     | Type   |
|----------|--|
| kmessage | S-XX-KafkaConnection > KafkaMessage (S_XX_FirstName) |
| mbody    | Any  |

44. In the Advanced tab, from the Tracing level drop-down, select **Verbose**.



45. On the process canvas, drag and drop an **Assignment** step between the Start and the End steps.

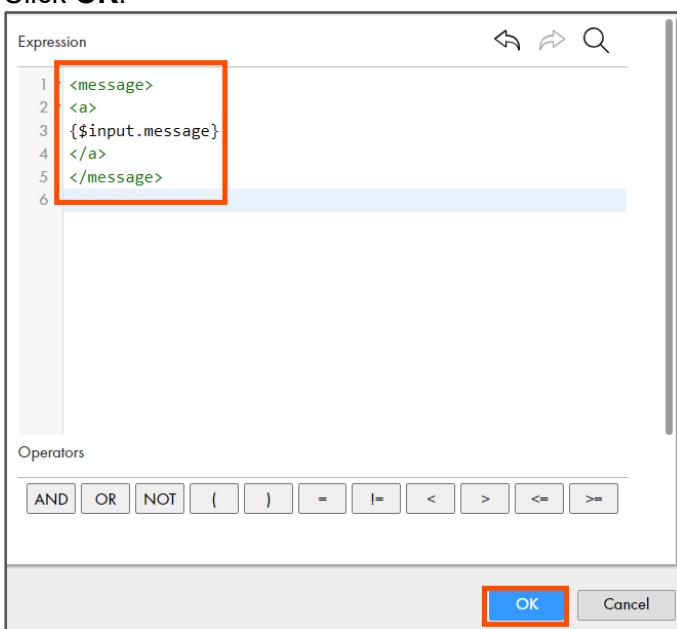


46. Retain the default name in the General tab.  
 47. In the Assignments tab, click plus (+) icon.  
 48. From the list of fields, select **mbody** and from the Assigned Using drop-down, select **Formula**.  
 49. To open the Formula Editor, click 

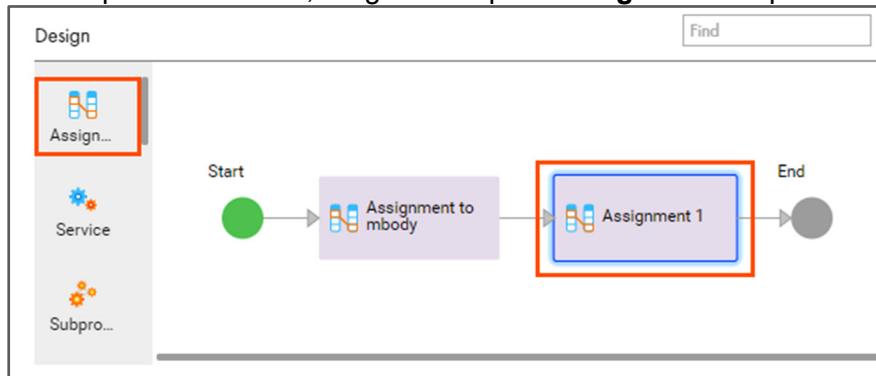


50. In the Expression Editor window, enter the formula mentioned below:  
**<message>**  
**<a>**  
**{\$input.message}**  
**</a>**  
**</message>**

51. Click **OK**.



52. On the process canvas, drag and drop an **Assignment** step before the End step.

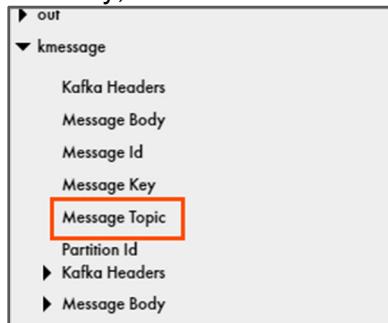


53. In the Assignments tab, click the **plus**  icon.

54. From the kmessage drop-down, select **Message Body**.

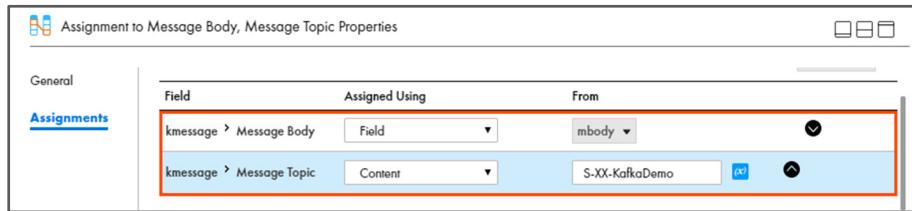


55. Similarly, add another field **kmessage > Message Topic**.

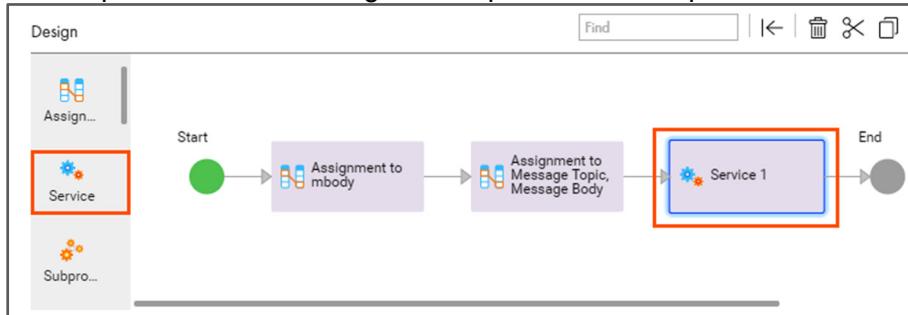


56. Assign values to the fields as mentioned in the table:

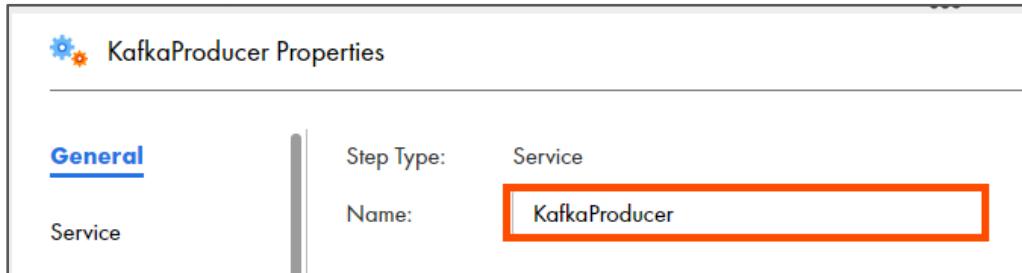
| Field         | Assigned Using | From           |
|---------------|----------------|----------------|
| Message Body  | Field          | mbody          |
| Message Topic | Content        | S-XX-KafkaDemo |



57. On the process canvas, drag and drop a **Service** step before the End step.



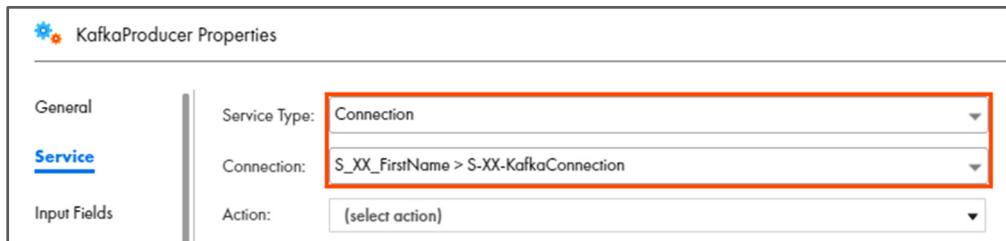
58. In the Name field, enter **KafkaProducer**.



| General |  | Step Type: | Service       |
|---------|--|------------|---------------|
| Service |  | Name:      | KafkaProducer |

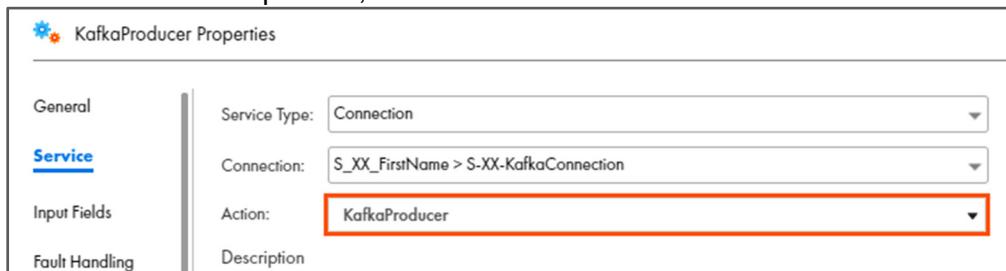
59. In the Service tab, from the Service Type drop-down, select **Connection**.

60. From the Connection drop-down, from your project folder, select **S-XX-KafkaConnection**.



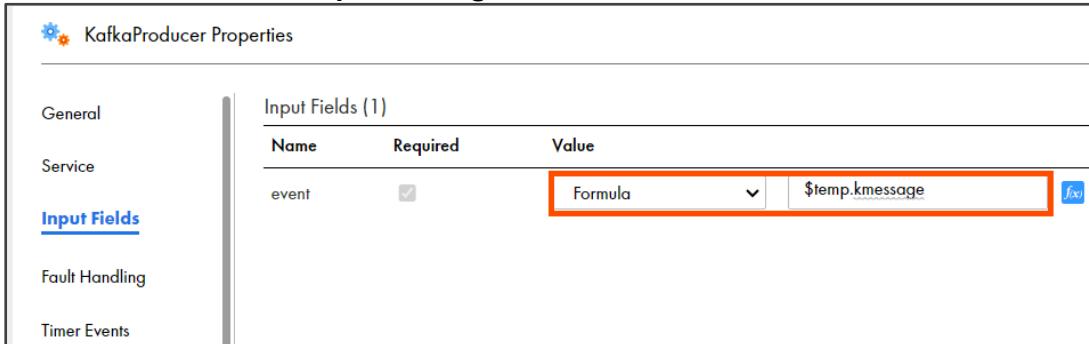
| General      |  | Service Type: | Connection                            |
|--------------|--|---------------|---------------------------------------|
| Service      |  | Connection:   | S_XX_FirstName > S-XX-KafkaConnection |
| Input Fields |  | Action:       | (select action)                       |

61. From the Action drop-down, select **KafkaProducer**.



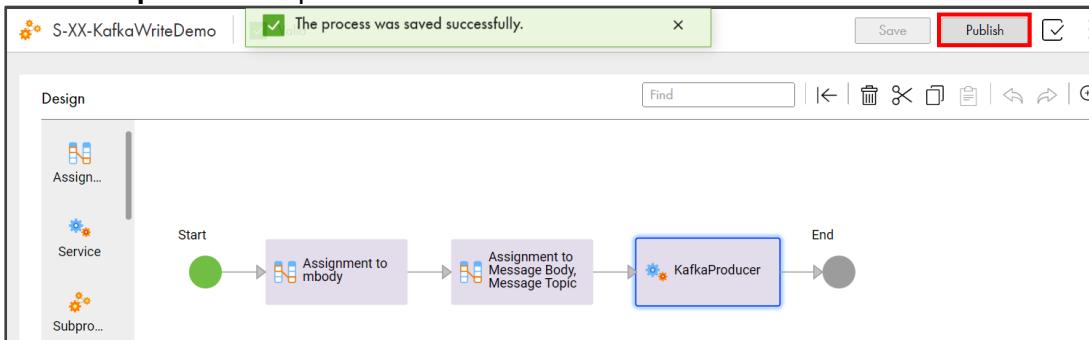
| General        |  | Service Type: | Connection                            |
|----------------|--|---------------|---------------------------------------|
| Service        |  | Connection:   | S_XX_FirstName > S-XX-KafkaConnection |
| Input Fields   |  | Action:       | KafkaProducer                         |
| Fault Handling |  | Description:  | (select action)                       |

62. In the Input Fields tab, for the event input field, from the Value drop-down, select **Formula**, and enter **\$temp.kmessage**.



| Name  | Required                            | Value  |
|-------|-------------------------------------|--|
| event | <input checked="" type="checkbox"/> | <input type="button" value="Formula"/> \$temp.kmessage <input type="button" value="fx"/> |

63. **Save and publish** the process.



The process was saved successfully.

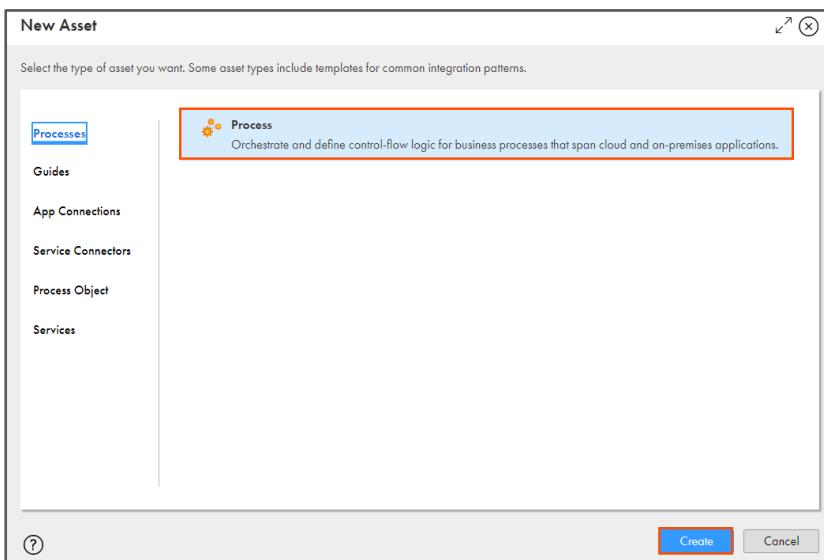
```

graph LR
    Start((Start)) --> Assignment1[Assignment to mbody]
    Assignment1 --> Assignment2[Assignment to Message Body, Message Topic]
    Assignment2 --> Kafka[KafkaProducer]
    Kafka --> End((End))
  
```

## Create a Read Process

Now, you will read the data from the topic. This is needed in order to read a message from the Kafka topic.

64. Create a new Process.

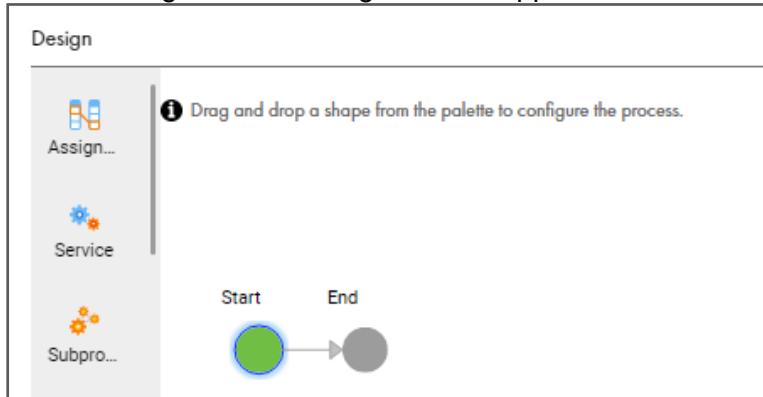


Select the type of asset you want. Some asset types include templates for common integration patterns.

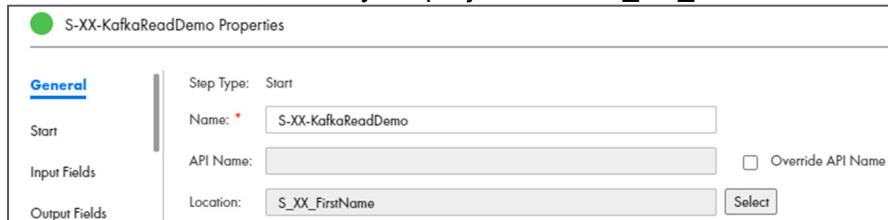
- Processes**
- Guides
- App Connections
- Service Connectors
- Process Object
- Services

**Process**  
Orchestrate and define control-flow logic for business processes that span cloud and on-premises applications.

The following Process Design canvas appears with the Start and End steps.

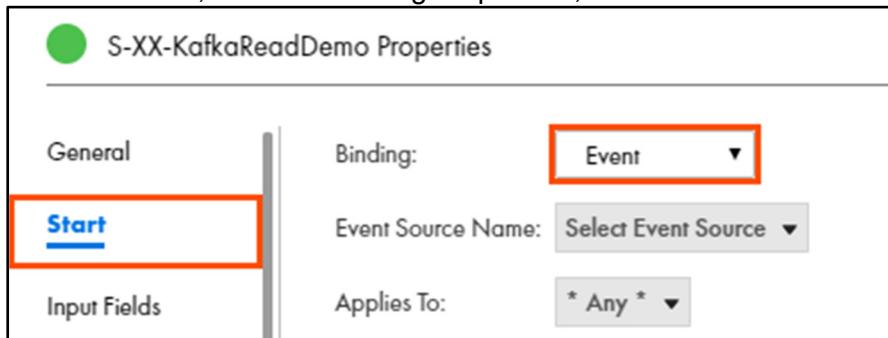


65. From the process canvas, select the **Start** step.
66. In the General tab, enter the name as **S-XX-KafkaReadDemo**.
67. Skip the API Name option.
68. In the Location field, select your project folder **S\_XX\_FirstName**.



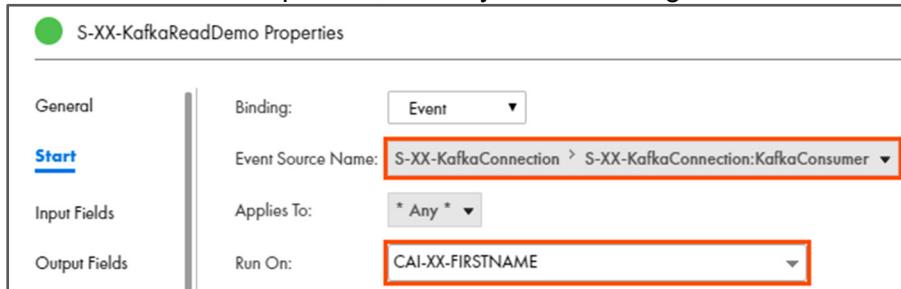
| S-XX-KafkaReadDemo Properties |  |
|-------------------------------|--|
| <b>General</b>                | Step Type: Start<br>Name: * S-XX-KafkaReadDemo<br>API Name: <input type="text"/><br><input type="checkbox"/> Override API Name<br>Location: S_XX_FirstName <input type="button" value="Select"/> |
| Start                         |  |
| Input Fields                  |  |
| Output Fields                 |  |

69. In the Start tab, from the Binding drop-down, select **Event**.



| S-XX-KafkaReadDemo Properties |   |
|-------------------------------|---|
| <b>General</b>                | Binding: <b>Event</b> <input type="button" value="▼"/><br><b>Start</b><br>Event Source Name: Select Event Source <input type="button" value="▼"/><br>Input Fields<br>Applies To: * Any * <input type="button" value="▼"/> |
| Input Fields                  |   |

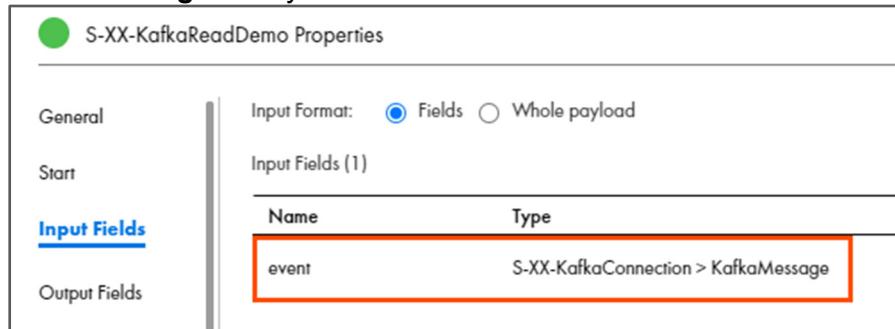
70. From the Event Source Name field, select **S-XX-KafkaConnection > S-XX-KafkaConnection:KafkaConsumer**.
71. From the Run On drop-down, select your Secure Agent.



| S-XX-KafkaReadDemo Properties |  |
|-------------------------------|--|
| <b>General</b>                | Binding: Event <input type="button" value="▼"/><br><b>Start</b><br>Event Source Name: <b>S-XX-KafkaConnection &gt; S-XX-KafkaConnection:KafkaConsumer</b> <input type="button" value="▼"/><br>Input Fields<br>Applies To: * Any * <input type="button" value="▼"/><br>Output Fields<br>Run On: CAI-XX-FIRSTNAME <input type="button" value="▼"/> |
| Input Fields                  |  |

72. Ensure that the **Input Fields** tab contains the **event** field.  
**Note:** If the event field does not exist, perform the following steps:

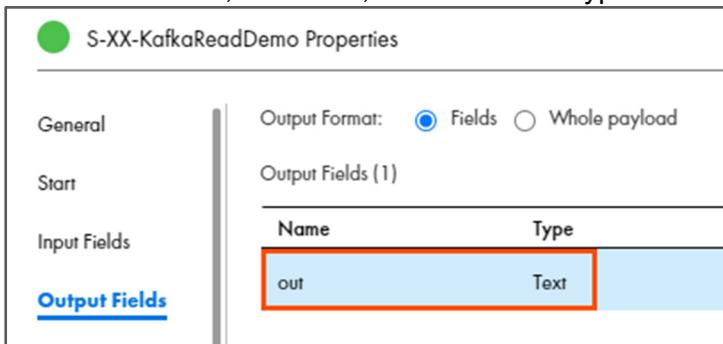
- i. In the Input Fields tab, click **Add Field**.
- ii. In the Name field, enter **event**, and select **S-XX-KafkaConnection > KafkaMessage** from your folder.



| Name  | Type                                |
|-------|-------------------------------------|
| event | S-XX-KafkaConnection > KafkaMessage |

73. In the **Output fields** tab, click the plus (+) icon to add a field.

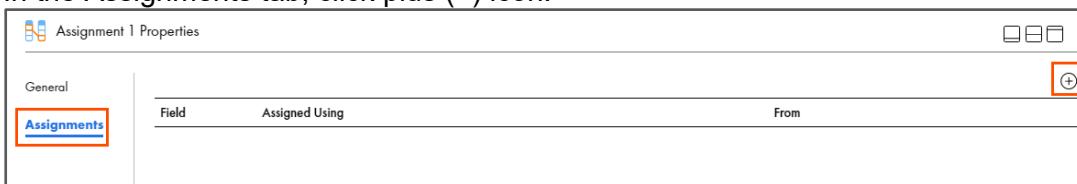
74. In the Name field, enter **out**, and retain the Type as **Text**.



| Name | Type |
|------|------|
| out  | Text |

75. On the process canvas, drag and drop an **Assignment** step between the Start and the End steps.

76. In the Assignments tab, click plus (+) icon.



| Field | Assigned Using | From |
|-------|----------------|------|
|       |                | (+)  |

77. Select **out**.

78. From the Assigned Using drop-down, select **Field**.

79. From the From drop-down, select **event > Message Body**.



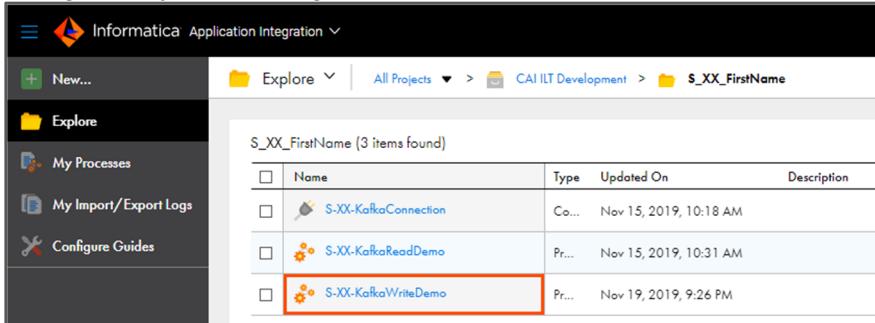
| Field | Assigned Using | From                 |
|-------|----------------|----------------------|
| out   | Field          | event > Message Body |

80. **Save and publish** the process.

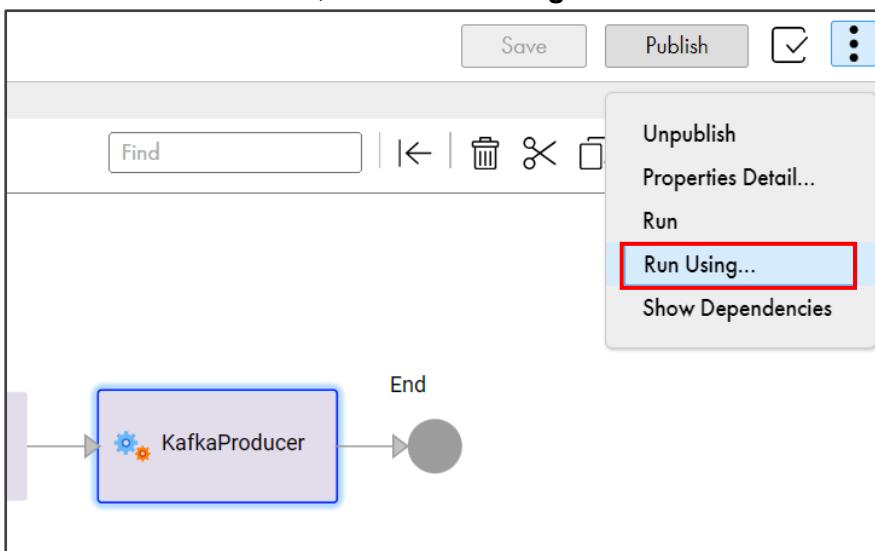
**Note:** As this process is an event-driven process, it keeps on pulling the Kafka topic, and as soon as there is a message in the topic it automatically triggers this process.

**Execute the process to read and write data to the topic:**

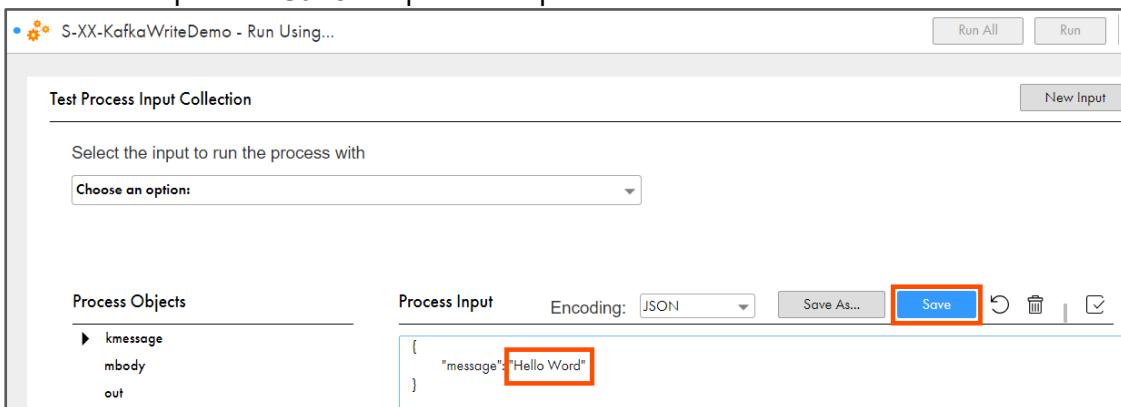
81. Navigate to your working folder and open the **S\_XX\_KafkaWriteDemo** process.



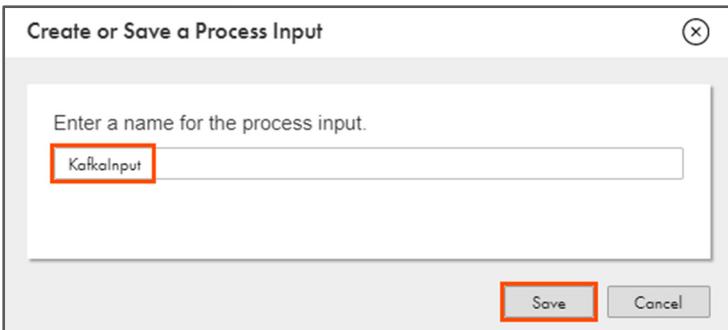
82. From the **Actions** menu, select **Run Using...**



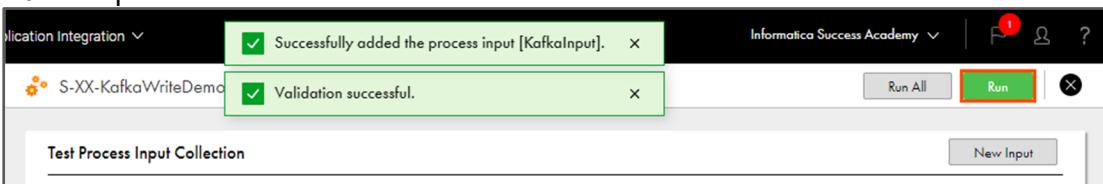
83. Provide an input and **Save** the process input.



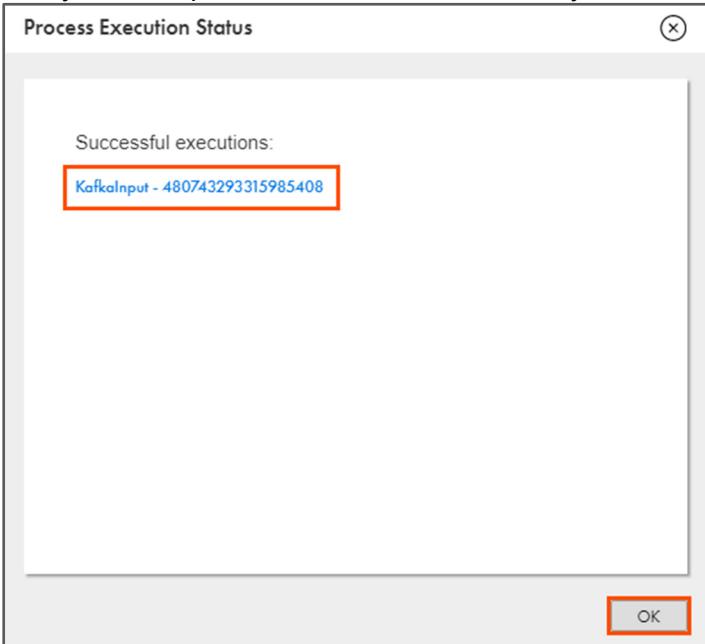
| Process Objects          | Process Input                            | Encoding: | Save As... | Save        |
|--------------------------|--|-----------|------------|-------------|
| kmessage<br>mbody<br>out | <pre>{   "message": "Hello Word" }</pre> | JSON      |            | <b>Save</b> |

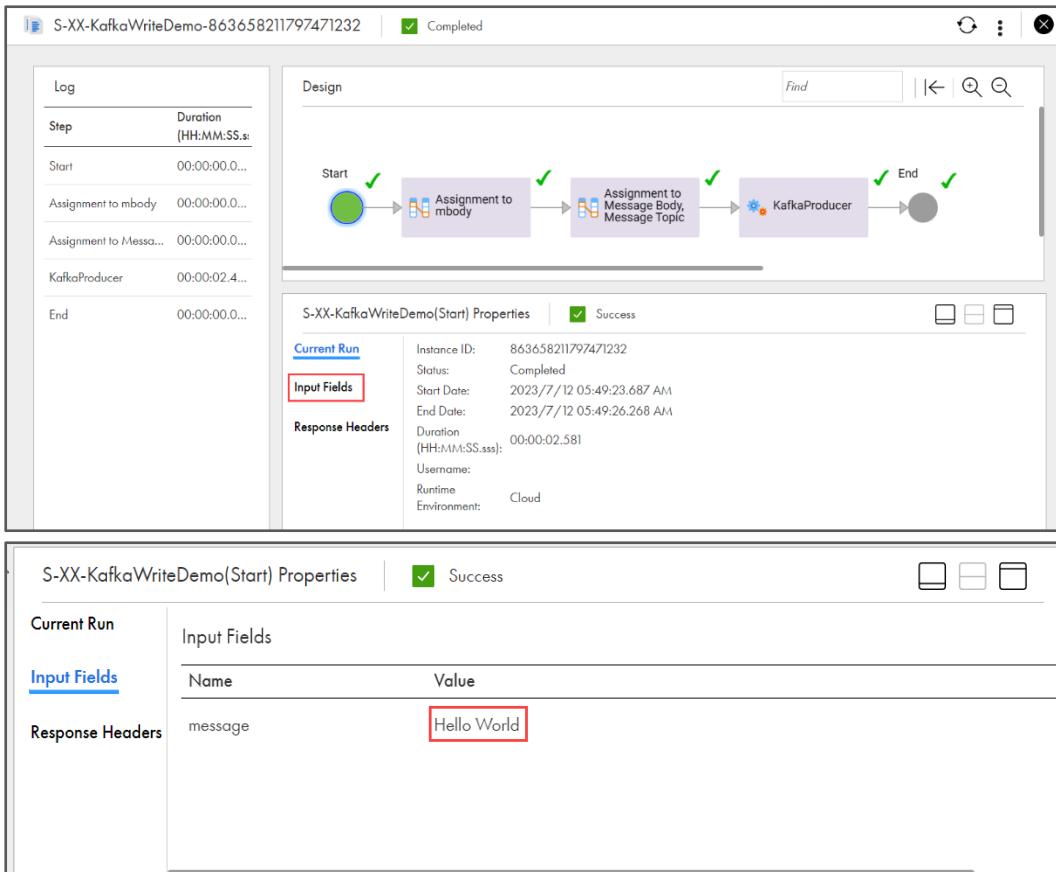


84. Run the process.



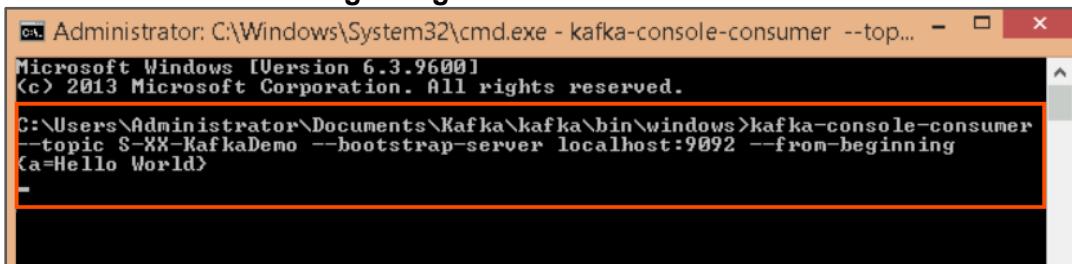
85. Verify that the process executes successfully. Click OK.





**Note:** As soon as you write the KafkaWriteDemo process, it will insert a message to the topic. KafkaReadDemo process that you have created will automatically trigger as well, because now, you have a message within the topic.

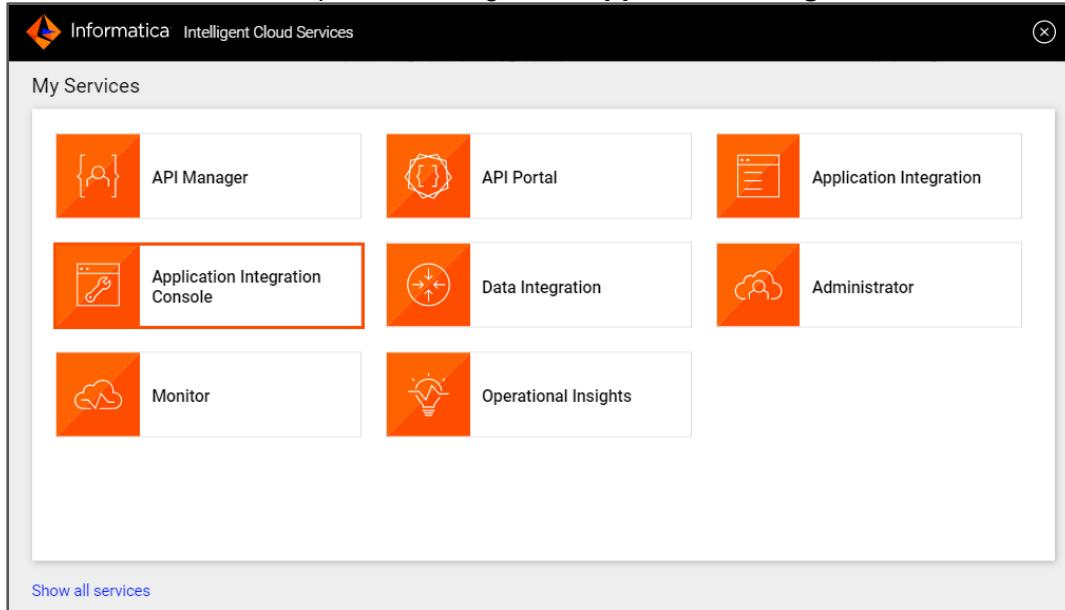
86. To validate the write process, from the **C:\Users\Administrator\Documents\Kafka\kafka\bin\windows** folder, open a command prompt.
87. Execute the following command:  
**kafka-console-consumer --topic S-XX-KafkaDemo --bootstrap-server localhost:9092 --from-beginning**



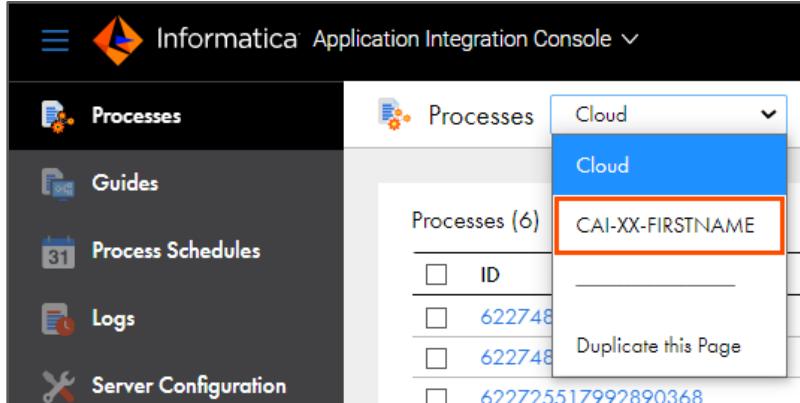
**Note:** Observe that the message that you entered in the input field is obtained in the command prompt. You can change your input message, run the process again, and check the values in this command prompt. You can close all the command prompts later. If you keep sending the messages (running the process) in the CAI console with different or same inputs, all the messages keep appending in the same above command prompt window in a new line.

As the read process is automatically invoked by read process, you can verify that the read process ran in the Application Integration Console service.

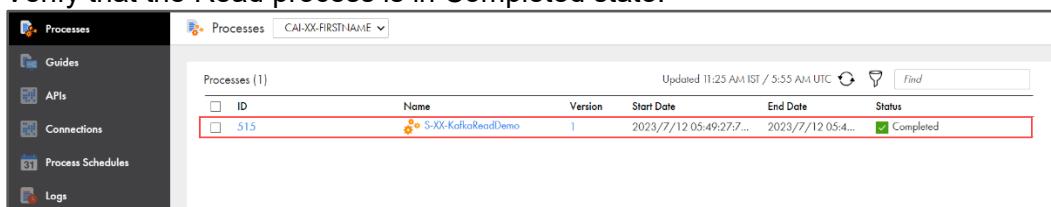
88. From the Processes drop-down, navigate to **Application Integration Console**.



89. From Process drop-down, select your Secure Agent.



90. Verify that the Read process is in Completed state.



| ID  | Name               | Version | Start Date              | End Date          | Status    |
|-----|--------------------|---------|-------------------------|-------------------|-----------|
| 515 | S-XX-KafkaReadDemo | 1       | 2023/7/12 05:49:27.7... | 2023/7/12 05:4... | Completed |

---

*This concludes the lab.*

# Module 8: Managing and Migrating Assets

## Lab 8-1: Export an Asset

### Overview:

When you export assets, Informatica Intelligent Cloud Services creates an export ZIP file that contains the assets that you selected for export.

You can select a single asset, multiple assets, or a project to export.

To include multiple assets, you can either select each asset within a folder or select a project or folder to export all of its assets. If you export a project, during import you can import the entire project or import only the assets that you select.

In this lab, you will export a single asset and view the log details.

### Objectives:

- Export an asset
- View the log details

### Duration:

10 Minutes

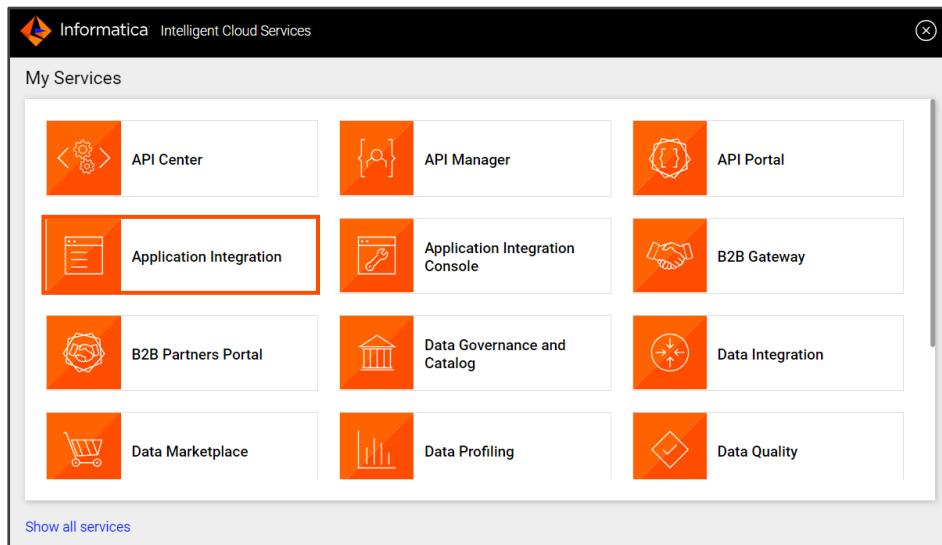
---

## Tasks

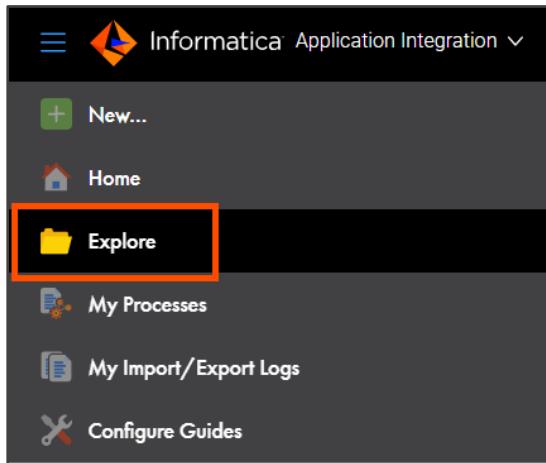
**Note:** The screen in your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

### Export an Asset

1. Log into IICS and access the **Application Integration** service.



2. From the navigation pane, click **Explore**.

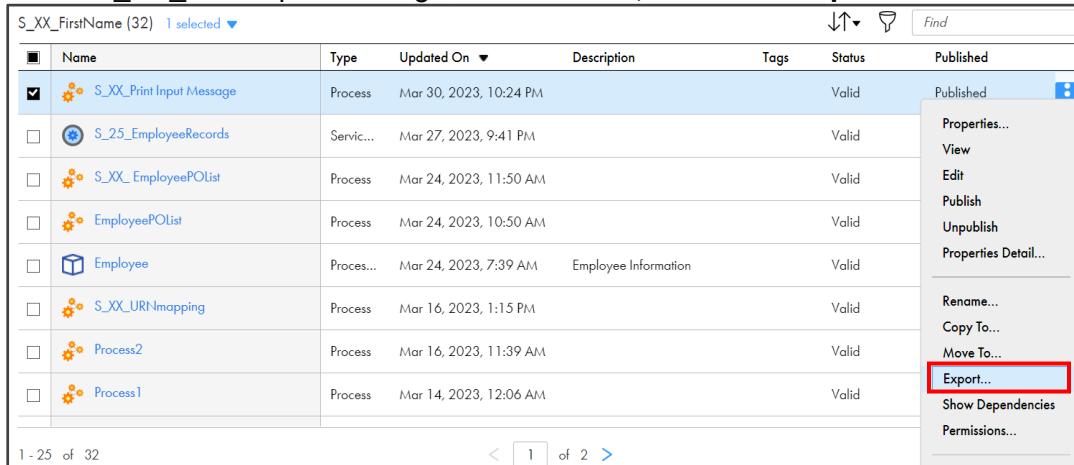


3. Open your project folder S\_XX\_FirstName.

4. Locate **S\_XX\_Print Input Message**.

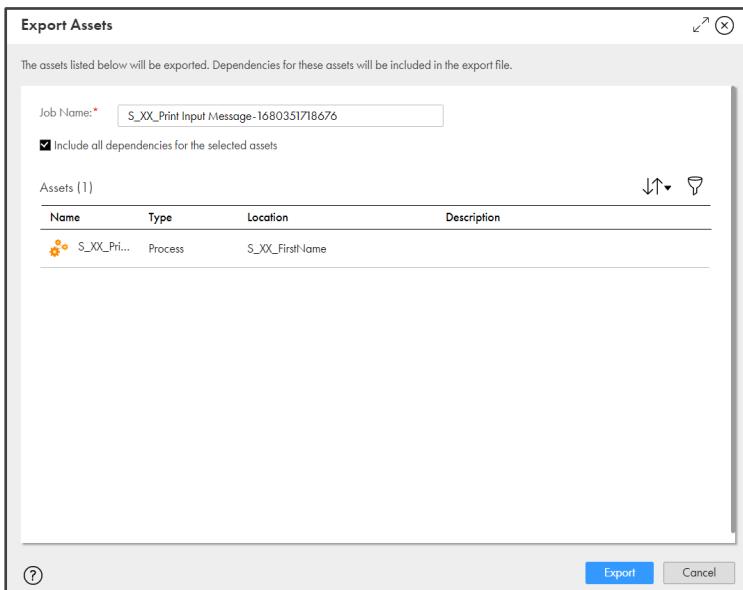
**Note:** In this lab, we are exporting S\_XX\_Print Input Message. You can select any other asset to export.

5. Select S\_XX\_Print Input Message, click **Actions**, and select **Export**.

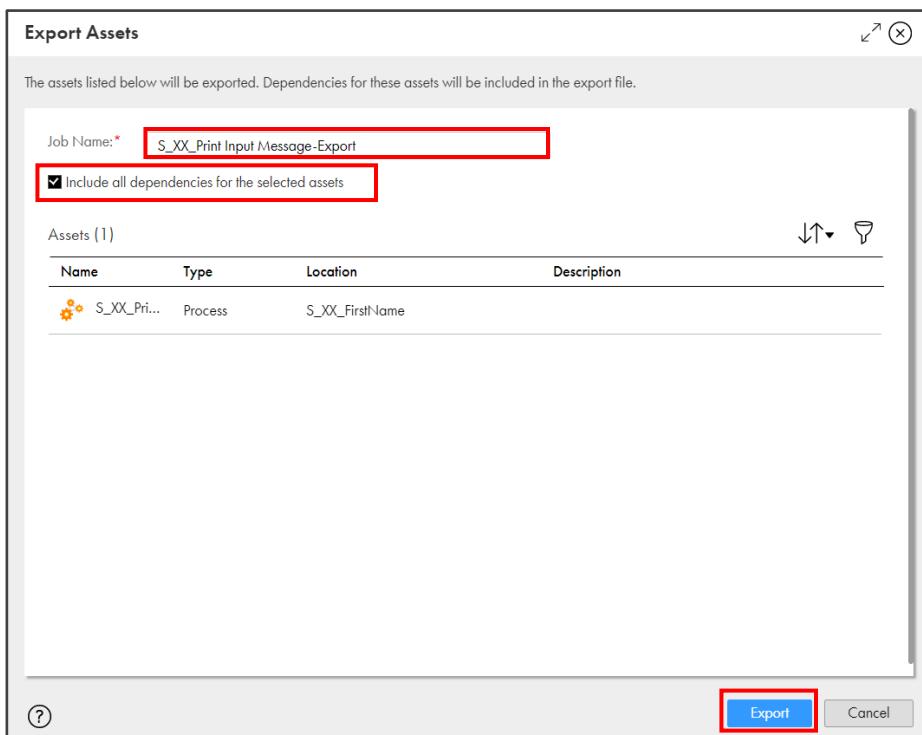


| Name   | Type    | Updated On             | Description          | Tags | Status | Published |
|--|---------|------------------------|----------------------|------|--------|-----------|
| <input checked="" type="checkbox"/> S_XX_Print Input Message | Process | Mar 30, 2023, 10:24 PM |                      |      | Valid  | Published |
| <input type="checkbox"/> S_25_EmployeeRecords                | Service | Mar 27, 2023, 9:41 PM  |                      |      | Valid  |           |
| <input type="checkbox"/> S_XX_EmployeePList                  | Process | Mar 24, 2023, 11:50 AM |                      |      | Valid  |           |
| <input type="checkbox"/> Employee                            | Process | Mar 24, 2023, 10:50 AM | Employee Information |      | Valid  |           |
| <input type="checkbox"/> S_XX_URNMapping                     | Process | Mar 16, 2023, 1:15 PM  |                      |      | Valid  |           |
| <input type="checkbox"/> Process2                            | Process | Mar 16, 2023, 11:39 AM |                      |      | Valid  |           |
| <input type="checkbox"/> Process1                            | Process | Mar 14, 2023, 12:06 AM |                      |      | Valid  |           |

The **Export Assets** window appears.



6. Change Job Name to **S\_XX\_Print Input Message-Export**.
7. Select the **Include all dependencies for the select assets** checkbox.
8. Click **Export**.



A message **Export started. Check My Import/Export Logs to download the export file** appears.

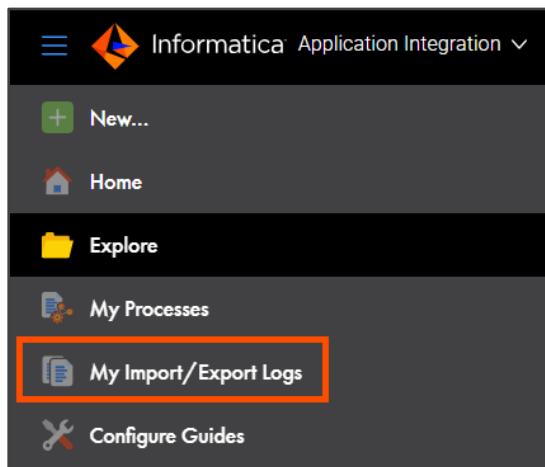
Explore ▼ All Projects ▾ >  Export started. Check [My Import/Export Logs](#) to download the export file. X

S\_XX\_FirstName (32) 1 selected ▾

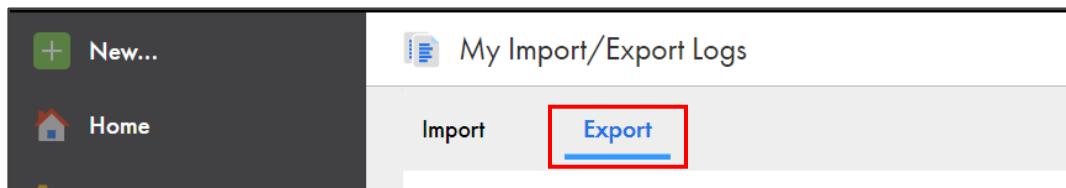
| <input type="checkbox"/>            | Name   | Type    | Updated On ▾           | Description |
|-------------------------------------|--|---------|------------------------|-------------|
| <input checked="" type="checkbox"/> |  S_XX_Print Input Message | Process | Mar 30, 2023, 10:24 PM |             |

## View My Import/Export Logs

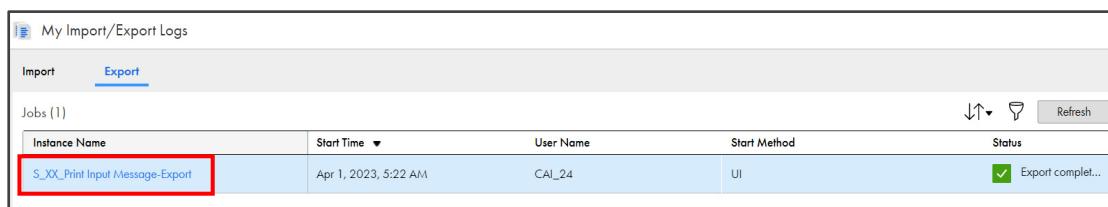
- To see the progress of the job, from the navigation pane, select **My Import/Export Logs**.



- From the menu at the top of the page, select **Export**.



- Click **S\_XX\_Print Input Message-Export** to open the log details page.



| Instance Name                   | Start Time ▾         | User Name | Start Method | Status  |
|---------------------------------|----------------------|-----------|--------------|---|
| S_XX_Print Input Message-Export | Apr 1, 2023, 5:22 AM | CAI_24    | UI           |  Export complet... |

12. Observe the log details.

**Note:** Check the Export File field. File is exported as a ZIP file.

| Properties    |   |
|---------------|---|
| Name:         | S_XX_Print Input Message-Export   |
| Type:         | Export  |
| Start Time:   | Apr 1, 2023, 5:22 AM  |
| End Time:     | Apr 1, 2023, 5:27 AM  |
| Started By:   | CAI_24  |
| Start Method: | UI  |
| Status:       | <input checked="" type="checkbox"/> Export completed successfully   |
| Export File:  | <a href="#">S_XX_Print Input Message-Export.zip</a>   |
| Export Log:   |  <a href="#">Download Export Log</a> |

| Exported Assets (1)   |         |                 |             |                                     |
|---|---------|-----------------|-------------|-------------------------------------|
| Name  | Type    | Source Location | Description | Status                              |
|  S_XX... | Process | S_XX_FirstName  |             | <input checked="" type="checkbox"/> |

13. To download the export file, click **S\_XX\_Print Input Message-Export.zip**.

| Properties    |   |
|---------------|---|
| Name:         | S_XX_Print Input Message-Export   |
| Type:         | Export  |
| Start Time:   | Apr 1, 2023, 5:22 AM  |
| End Time:     | Apr 1, 2023, 5:27 AM  |
| Started By:   | CAI_24  |
| Start Method: | UI  |
| Status:       | <input checked="" type="checkbox"/> Export completed successfully   |
| Export File:  | <a href="#">S_XX_Print Input Message-Export.zip</a>   |
| Export Log:   |  <a href="#">Download Export Log</a> |

| Exported Assets (1)   |         |                 |             |                                     |
|---|---------|-----------------|-------------|-------------------------------------|
| Name  | Type    | Source Location | Description | Status                              |
|  S_XX... | Process | S_XX_FirstName  |             | <input checked="" type="checkbox"/> |

14. Close the asset tab.

*This concludes the lab.*

## Module 9: Using XQuery in CAI

### Lab 9-1: Create a Process Using Lists Function of XQuery

#### Overview:

XQuery is a functional language which is responsible for finding and extracting elements and attributes from XML documents. It can be used to extract information to use in a web service, generate summary reports, transform XML data to XHTML and search web documents for relevant information.

#### Objectives:

- Create a Process Object – Employee
- Create a Process using Lists functions

#### Duration:

50 Minutes

---

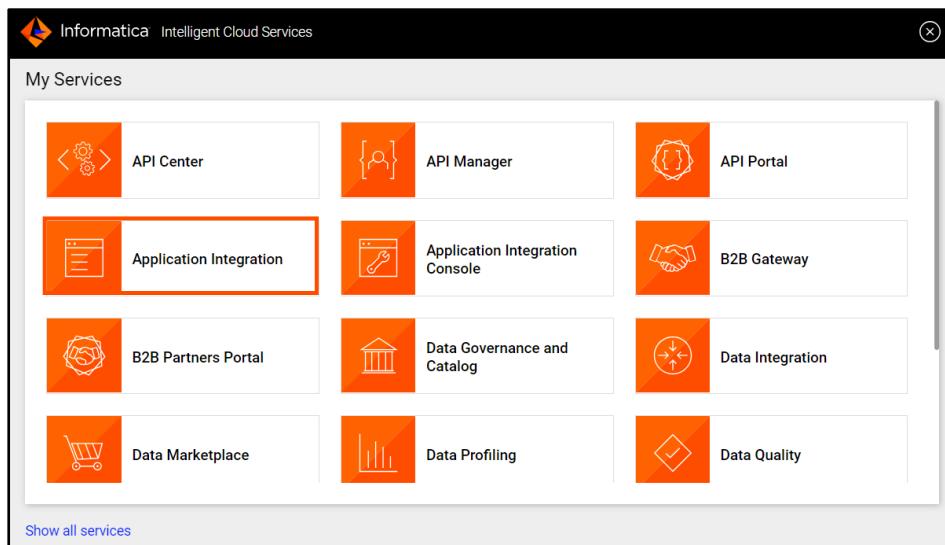
#### Tasks:

**Note:** The screen in your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

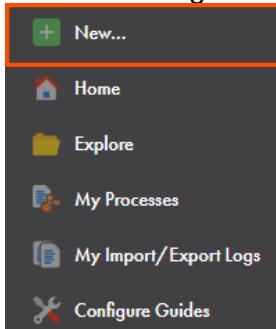
### Create a Process Object – Employee

This process object includes the Employee ID, first name, and last name of the employee.

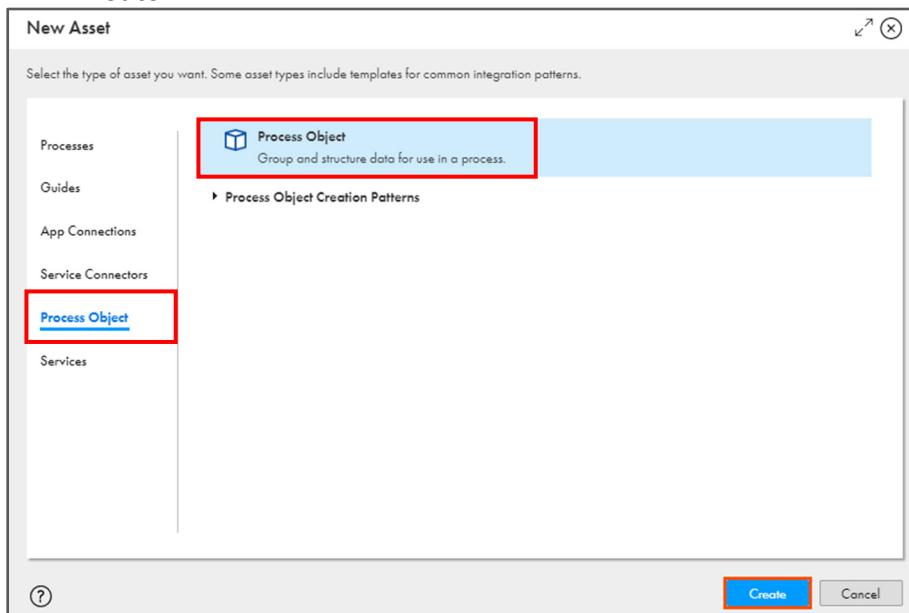
1. Log into IICS and access the Cloud **Application Integration** service.



2. From the Navigation pane, click **New**.



3. In the New Asset window, from the Process Object tab, select **Process Object**, and click **Create**.



4. In the Process Object Properties page, in the Name field, enter **Employee**.  
 5. In the Location field, select your project folder **S\_XX\_FirstName**.  
 6. In the Description field, enter **Employee Information**.

| Properties   |                      |
|--------------|----------------------|
| Name:*       | Employee             |
| Location:*   | S_XX_FirstName       |
| Description: | Employee Information |

For the Employee Process Object, the required fields are – **EMP\_ID**, **FIRST\_NAME**, **LAST\_NAME**, **CREATED\_DATE**, **CLOBFIELD**, and **TEST2**.

7. For the first field, enter Name and Label as **EMP\_ID**.
8. From the Type drop-down, select **Text**.
9. Click **Add Field**.

Fields

|                    |        |        |        |       |      |  |
|--------------------|--------|--------|--------|-------|------|--|
| Name: <sup>*</sup> | EMP_ID | Label: | EMP_ID | Type: | Text | <input type="button" value="Add Field"/> |
|--------------------|--------|--------|--------|-------|------|--|

10. Similarly, add the other fields as shown in the table below:

| Name         | Label        | Type |
|--------------|--------------|------|
| FIRST_NAME   | FIRST_NAME   | Text |
| LAST_NAME    | LAST_NAME    | Text |
| CREATED_DATE | CREATED_DATE | Text |
| CLOBFIELD    | CLOBFIELD    | Text |
| TEST2        | TEST2        | Text |

| Name*          | Label        | Type | Required                 | Nullable                            |
|----------------|--------------|------|--------------------------|-------------------------------------|
| ► EMP_ID       | EMP_ID       | Text | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| ► FIRST_NAME   | FIRST_NAME   | Text | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| ► LAST_NAME    | LAST_NAME    | Text | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| ► CREATED_DATE | CREATED_DATE | Text | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| ► CLOBFIELD    | CLOBFIELD    | Text | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| ► TEST2        | TEST2        | Text | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

11. Save the Process Object.

Employee     Valid     The process object was saved successfully.

---

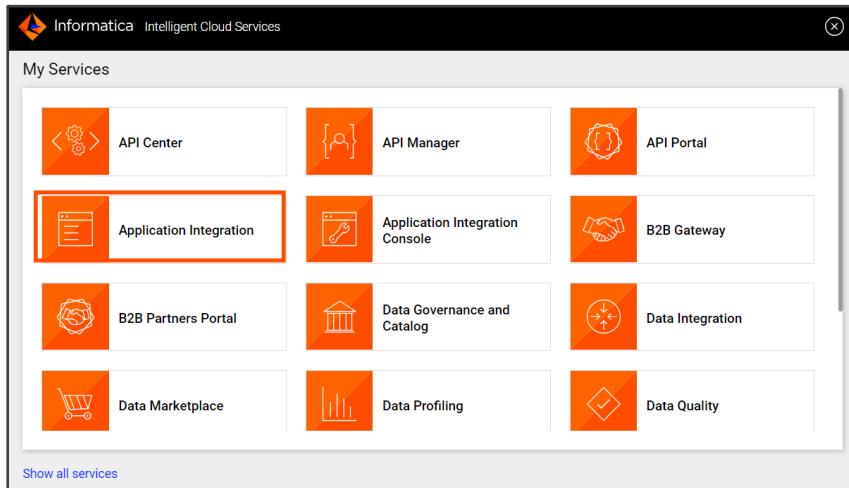
**Properties:**

|              |                      |        |
|--------------|----------------------|--------|
| Name:        | Employee             |        |
| Location:    | S_XX_FirstName       | Select |
| Description: | Employee Information |        |

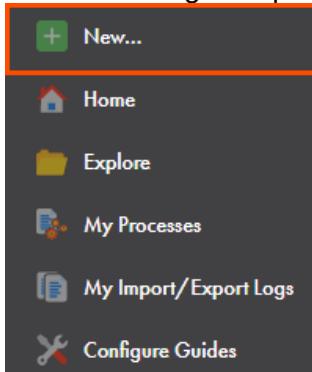
You have successfully designed a Process Object.

## Create a Process S\_XX\_EmployeePOList

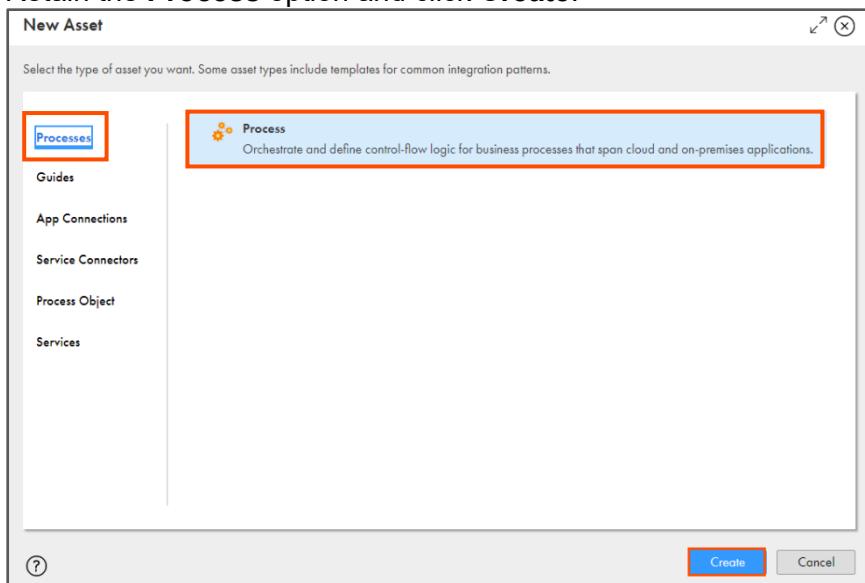
### 12. Navigate to Application Integration.



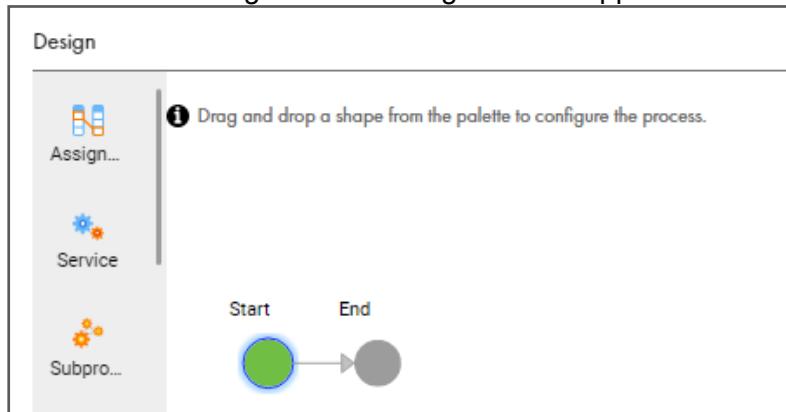
### 13. From the navigation pane, select New.



### 14. Retain the Process option and click Create.



**Note:** The following Process Design canvas appears with the Start and End steps.



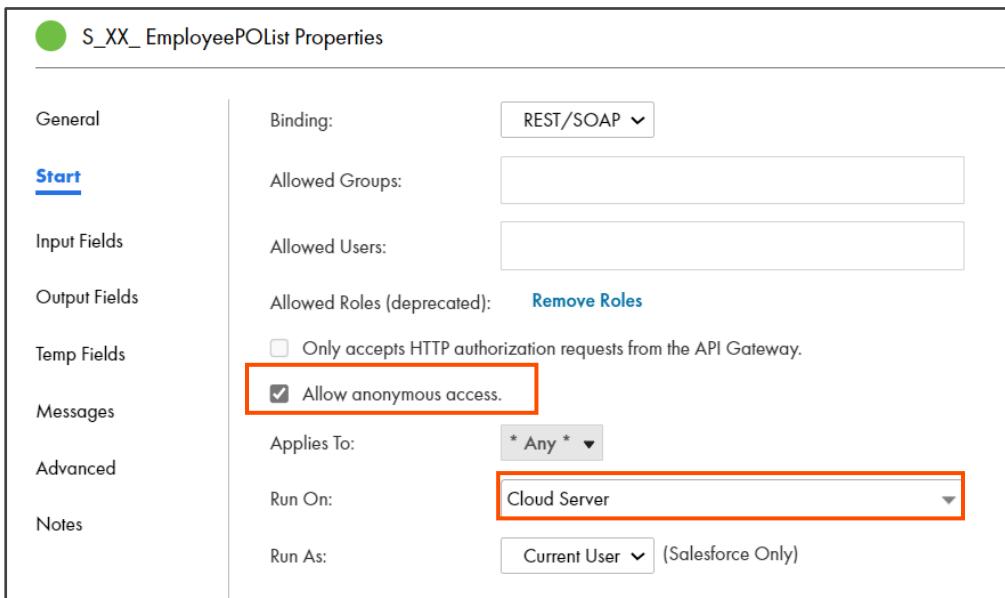
15. From the process canvas, select the **Start** step.
16. In the General properties tab, in the Name field, enter **S\_XX\_EmployeePOList**.
17. Skip the API Name option.
18. In the Location field, select your project folder **S\_XX\_FirstName**.



The screenshot shows the 'S\_XX\_EmployeePOList Properties' dialog. The left sidebar has tabs for General, Start, Input Fields, Output Fields, Temp Fields, Messages, and Advanced. The 'General' tab is selected. The 'Step Type:' dropdown is set to 'Start'. The 'Name:' field contains 'S\_XX\_EmployeePOList'. The 'API Name:' field is empty. The 'Location:' field contains 'S\_XX\_FirstName' with a 'Select' button next to it. The 'Override API Name' checkbox is unchecked. The 'Description:' field is empty.

19. From the properties pane, select **Start**.
20. Select the **Allow anonymous access** option.

21. From the Run On drop-down, retain **Cloud Server**.

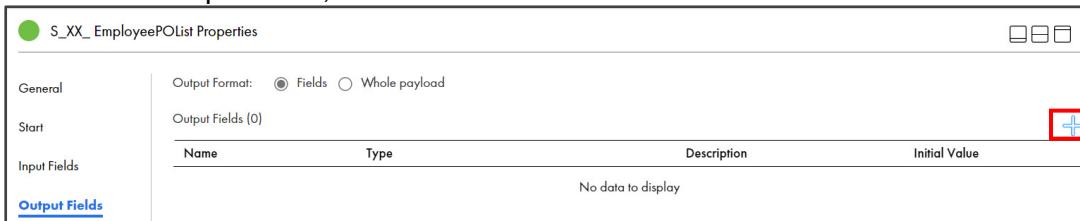


The screenshot shows the 'Start' tab of the properties dialog for 'S\_XX\_EmployeePOList Properties'. The 'Run On:' dropdown is set to 'Cloud Server', which is highlighted with a red box. The 'Run As:' dropdown is set to 'Current User (Salesforce Only)'. The 'Allow anonymous access' checkbox is checked and highlighted with a red box.

Next, you will create the output fields to display the results.

22. From the properties pane, select **Output Fields**.

23. To add new output fields, click .



The screenshot shows the 'Output Fields' tab of the properties dialog. The 'Output Format' section has 'Fields' selected. The 'Output Fields (0)' table has a 'Name' column header and a 'Type' column header. A red box highlights the '+' button in the top right corner of the table area.

24. Create the output fields, as shown in the table below:

| Name         | Type |
|--------------|------|
| Pretail      | Text |
| Posttail     | Text |
| DataPretail  | Text |
| DataPosttail | Text |

S\_XX\_EmployeePOList Properties

|                      |  |
|----------------------|--|
| General              | Output Format: <input checked="" type="radio"/> Fields <input type="radio"/> Whole payload |
| Start                | Output Fields (4)  |
| Input Fields         |  |
| <b>Output Fields</b> |  |
| Temp Fields          |  |
| Messages             |  |
| Advanced             |  |
| Notes                |  |

**Name**      **Type**

|              |      |
|--------------|------|
| Pretail      | Text |
| Posttail     | Text |
| DataPretail  | Text |
| DataPosttail | Text |

Next, you will create the Temp fields.

25. To add a new Temp field, click .
26. In the Name field, enter **Empobj**.
27. From the Type field drop-down, select **More Types**.
28. From the Category drop-down, select **Custom Types**.
29. Select your project folder.
30. Select **Employee**.
31. Click **OK**.

Edit Type

Category: \*

Types

Template

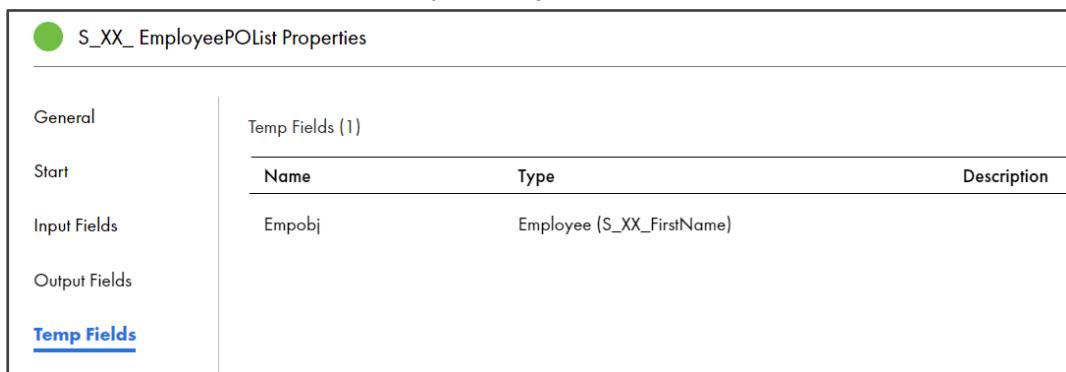
S\_XX\_FirstName

Employee

Allow a list of objects of this type.

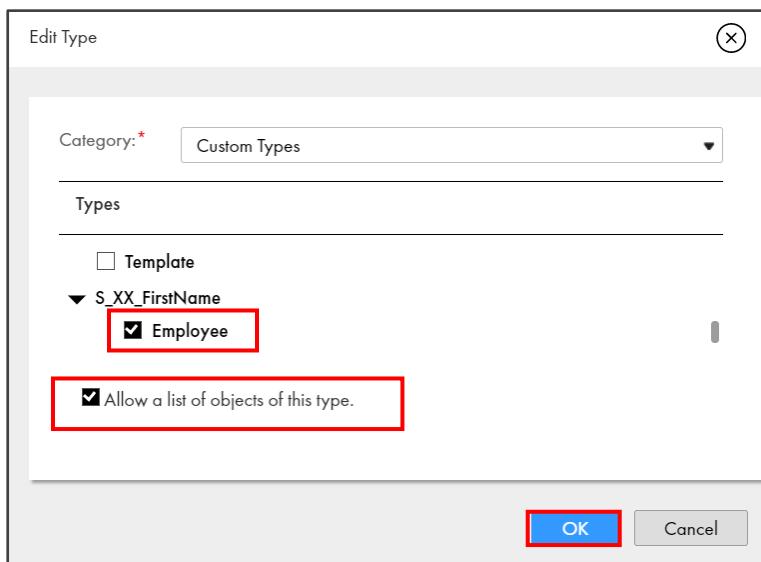
**OK** **Cancel**

**Note:** Here, S\_XX\_FirstName is your project folder.



| Name   | Type                      | Description |
|--------|---------------------------|-------------|
| Empobj | Employee (S_XX_FirstName) |             |

32. Create one more Temp field named as **EmpList**.
33. From the Type field drop-down, select **More Types**.
34. From the Category drop-down, select **Custom Types**.
35. Select your project folder.
36. Select **Employee**.
37. Select **Allow a list of objects of this type**.
38. Click **OK**.



39. Similarly, create three more Temp fields, as shown in the table below:

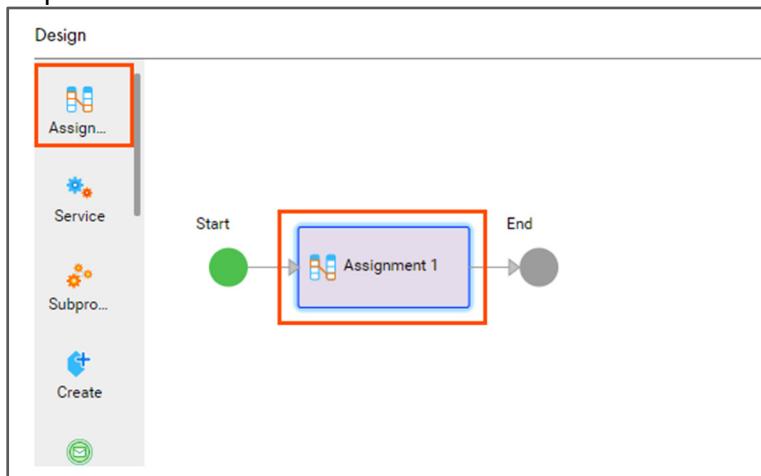
| Name     | Type                       |
|----------|----------------------------|
| taillist | Employees (S_XX_FirstName) |
| Headlist | Employee (S_XX_FirstName)  |

|                    |                 |                                   |             |
|--------------------|-----------------|-----------------------------------|-------------|
| General            | Temp Fields (4) |                                   |             |
| Start              | Name            | Type                              | Description |
| Input Fields       | Empobj          | Employee {S_XX_FirstName}         |             |
| Output Fields      | Emplist         | List of Employee {S_XX_FirstName} |             |
| <b>Temp Fields</b> | taillist        | List of Employee {S_XX_FirstName} |             |
| Messages           | Headlist        | List of Employee {S_XX_FirstName} |             |
| Advanced           |                 |                                   |             |
| Notes              |                 |                                   |             |

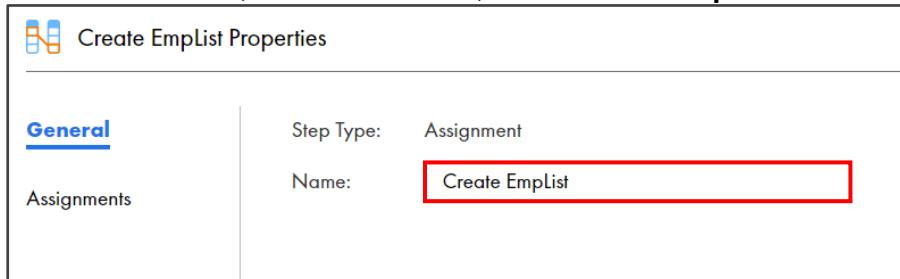
### Add an Assignment step Create EmpList:

Now, you will add an Assignment step ‘Create EmpList’ to display the list of all elements.

40. From the palette, drag and drop an **Assignment** step between the Start and the End step on the canvas.

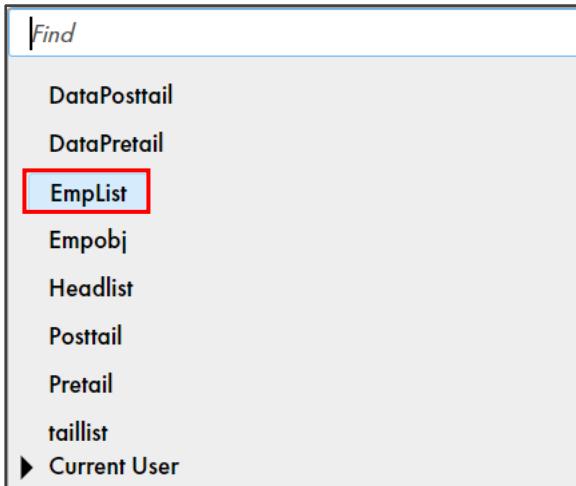


41. Select the **Assignment** step.  
 42. In the General tab, in the Name field, enter **Create EmpList**.



|   |   |
|---|---|
|  Create EmpList Properties |   |
| <b>General</b>  | Step Type: Assignment                             |
| Assignments   | Name: <input type="text" value="Create EmpList"/> |

43. In the Assignments tab, click the plus (+) icon, and select **EmpList**.



44. From the **Assigned Using** drop-down, select **Formula**.

45. To open the Formula Editor, click .

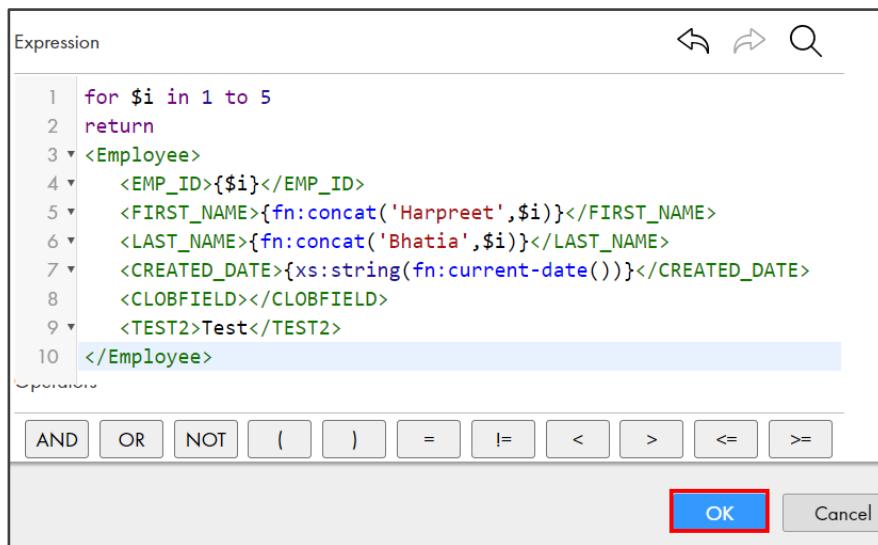
In the Expression field, enter the following expression:

```
for $i in 1 to 5
return
<Employee>
  <EMP_ID>{$i}</EMP_ID>
  <FIRST_NAME>{fn:concat('Harpreet',$i)}</FIRST_NAME>
  <LAST_NAME>{fn:concat('Bhatia',$i)}</LAST_NAME>
  <CREATED_DATE>{xs:string(fn:current-date())}</CREATED_DATE>
  <CLOBFIELD></CLOBFIELD>
  <TEST2>Test</TEST2>
</Employee>
```

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **16\_LabGuide\_CAI\_Create a Process Using Lists Function of XQuery\_9-1**. Copy the command mentioned under **Step A** and paste it in the Expression field.

46. Click **OK**.



The screenshot shows the Expression dialog box with the following XQuery code:

```

1 for $i in 1 to 5
2 return
3 <Employee>
4   <EMP_ID>{$i}</EMP_ID>
5   <FIRST_NAME>{fn:concat('Harpreet',$i)}</FIRST_NAME>
6   <LAST_NAME>{fn:concat('Bhatia',$i)}</LAST_NAME>
7   <CREATED_DATE>{xs:string(fn:current-date())}</CREATED_DATE>
8   <CLOBFIELD></CLOBFIELD>
9   <TEST2>Test</TEST2>
10 </Employee>

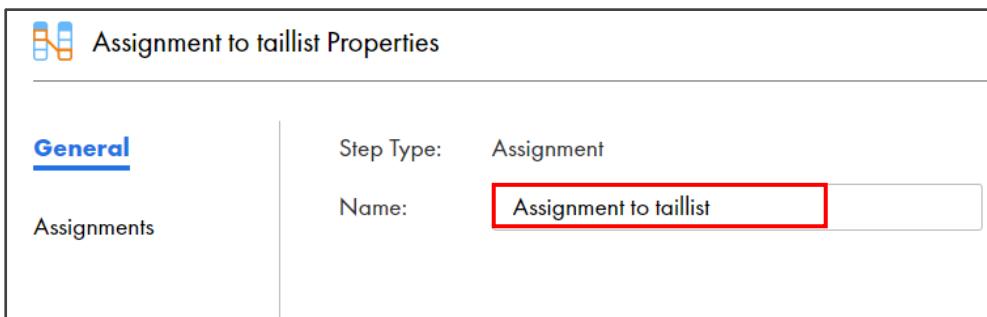
```

Below the code editor are several comparison operators: AND, OR, NOT, (, ), =, !=, <, >, <=, >=. The **OK** button is highlighted with a red border.

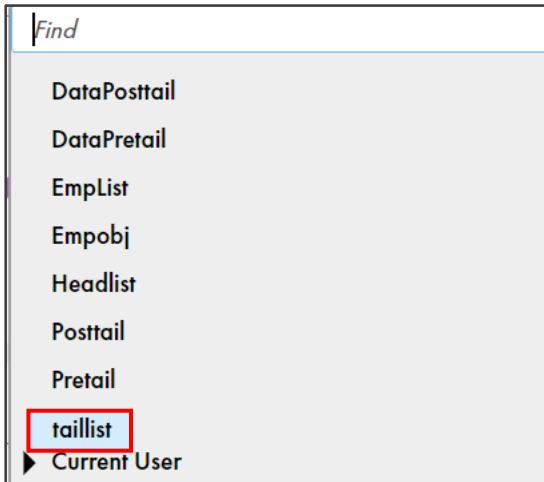
### Add an Assignment step Assignment to taillist

Now, you will add an Assignment step to display the list of elements except the first one.

47. From the Design palette, drag and drop an **Assignment** step on the link between the Create EmpList and End step.
48. In the General section of Assignment properties, enter the Name as **Assignment to taillist**.



49. In the Assignments tab, click the plus (+) icon, and select **taillist**.



50. From the **Assigned Using** drop-down, select **Formula**.

51. To open the Formula Editor, click .

In the Expression field, enter the following expression:

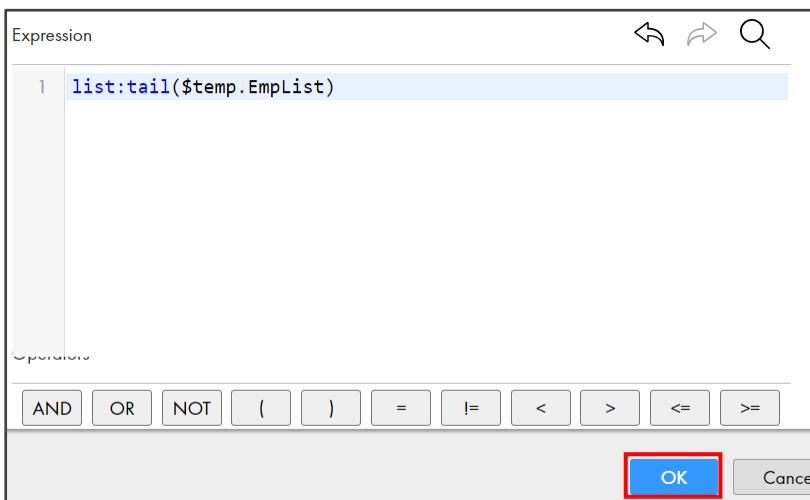
`list:tail($temp.EmpList)`

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **16\_LabGuide\_CAI\_Create a Process Using Lists Function of XQuery\_9-1**. Copy the command mentioned under **Step B** and paste it in the Expression field.

**Note:** The tail function returns an array containing all members except the first from a supplied array.

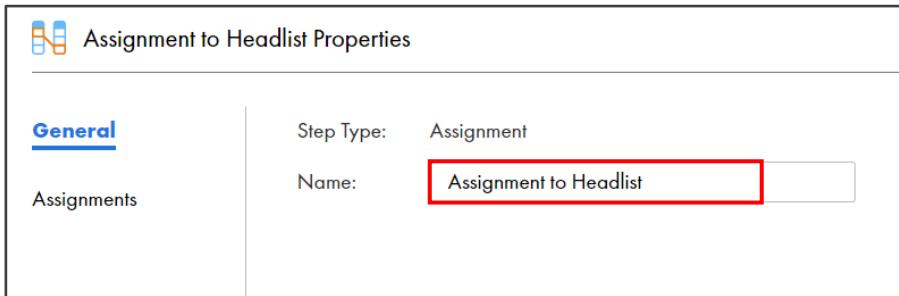
52. Click **OK**.



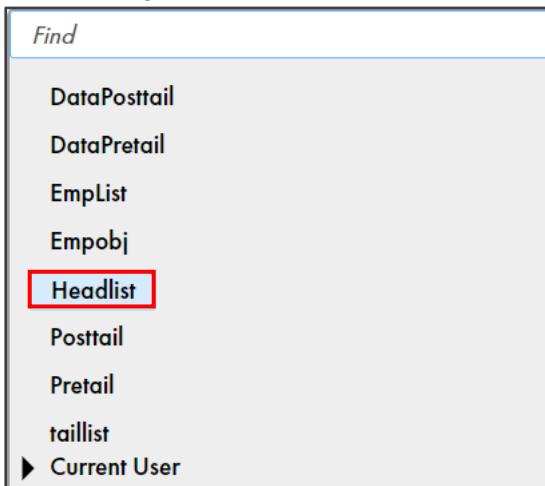
## Add an Assignment step Assignment to Headlist

Now, you will add an Assignment step to display the first element of the list.

53. From the Design palette, drag and drop an **Assignment** step on the link between the Assignment to taillist and End step.
54. In the General section of Assignment properties, enter the Name as **Assignment to Headlist**.



55. In the Assignments tab, click the plus (+) icon, and select **Headlist**.



56. From the **Assigned Using** drop-down, select **Formula**.

57. To open the Formula Editor, click .

In the Expression field, enter the following expression:

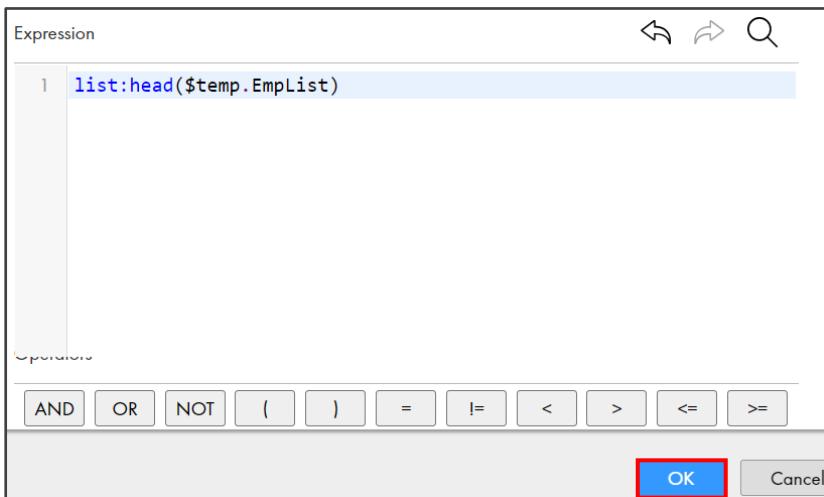
**list:head(\$temp.EmpList)**

OR

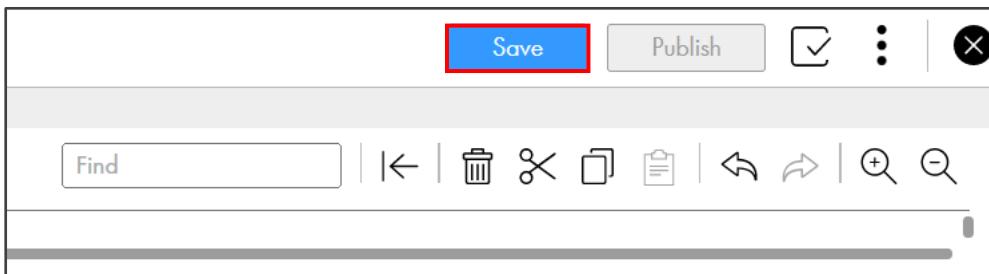
Open the **CAI Lab Prep Files** folder and open the notepad file named **16\_LabGuide\_CAI\_Create a Process Using Lists Function of XQuery\_9-1**. Copy the command mentioned under **Step C** and paste it in the Expression field.

**Note:** The head function returns the first element of an array.

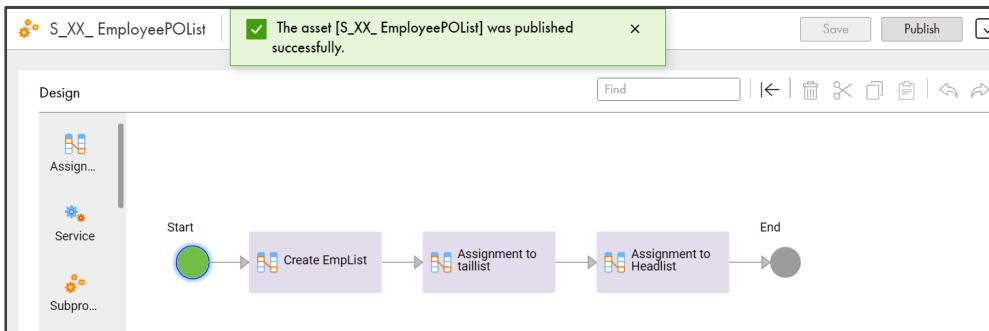
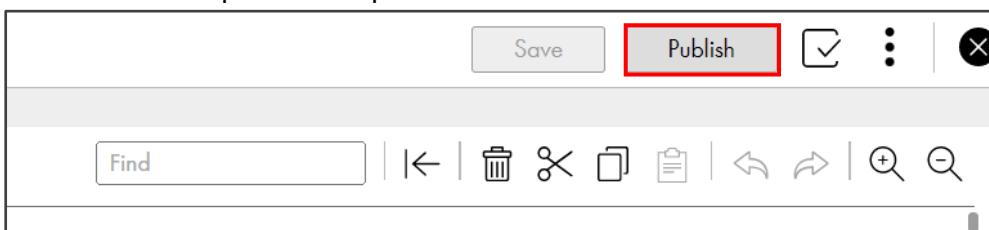
58. Click **OK**.



59. Click **Save**.



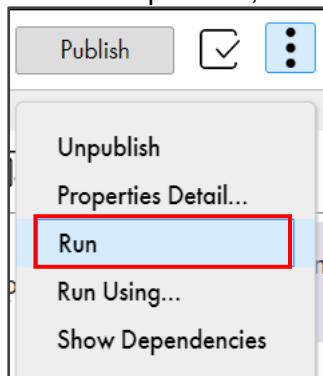
60. Click **Publish** to publish the process.



The process is published successfully.

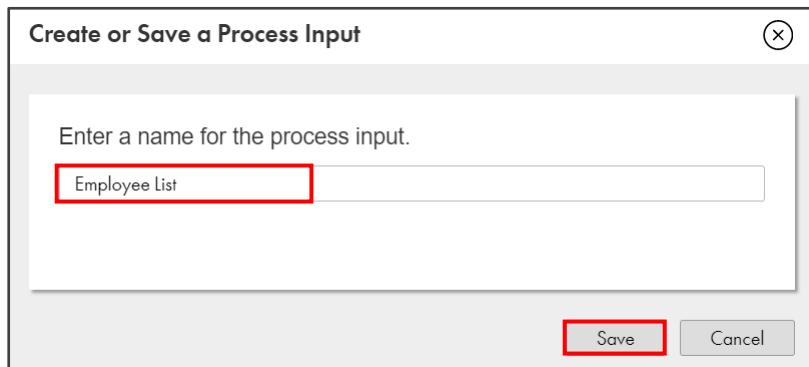
## Invoke the Process from the Interface

61. To run the process, click , and select **Run**.



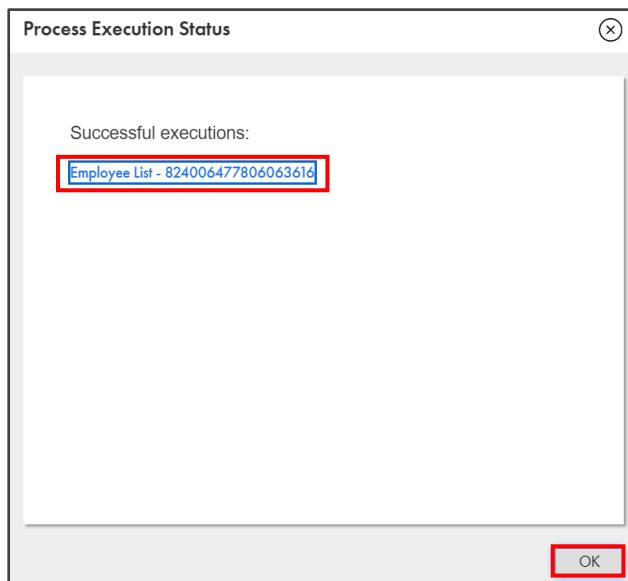
62. Click **Save As...** to Save the input.

63. Save the input as **Employee List**.

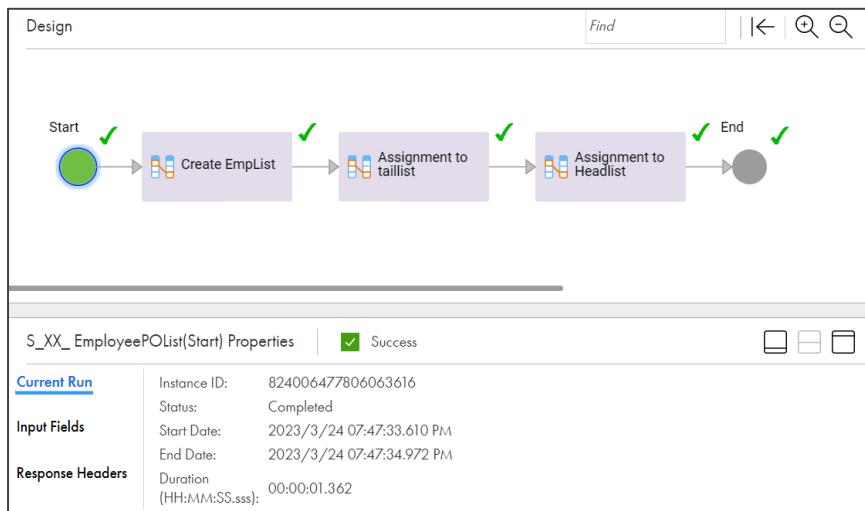


64. After the input is added and validated successfully, click **Run**.

65. Select the Process Execution link and click **OK**.

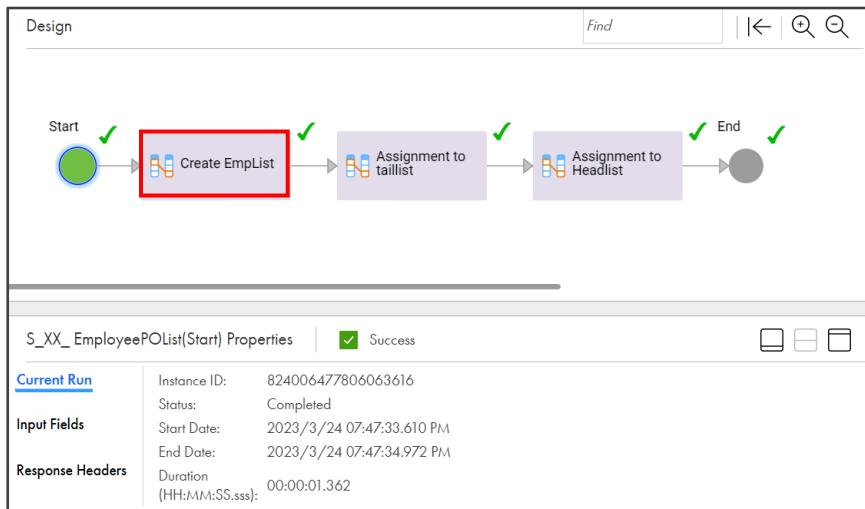


Process is executed successfully. The execution details are as shown below:



**Note:** If you are not able to see this screen, click **Refresh**.

66. From the Design selection, click **Create EmpList** to check the list of the employee.



67. In the Create EmpList Properties section, observe the output. A list of five employees is created.

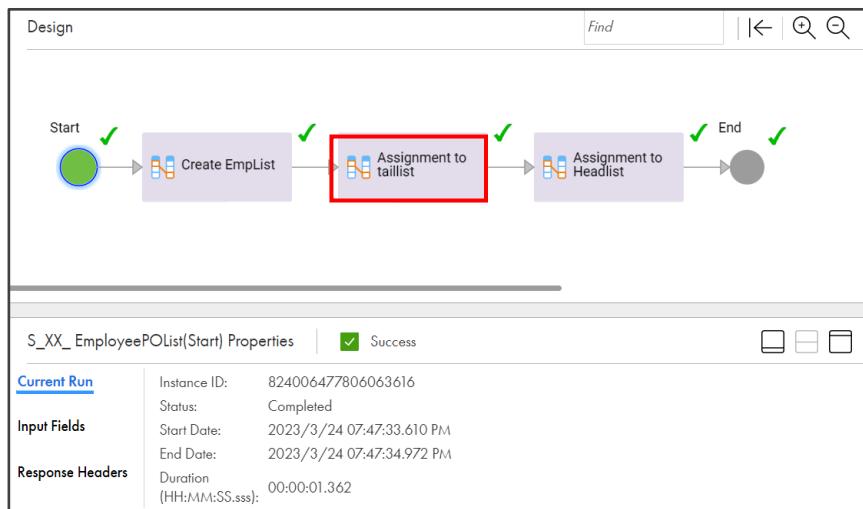


The screenshot displays two separate instances of the "Create EmpList Properties" window, both showing successful execution. Each window lists five employees with their details:

- Employee 1:** ID 1, Name Harpreet1, Surname Bharia1, Date 2023-03-27Z, Test
- Employee 2:** ID 2, Name Harpreet2, Surname Bharia2, Date 2023-03-27Z, Test
- Employee 3:** ID 3, Name Harpreet3, Surname Bharia3, Date 2023-03-27Z, Test
- Employee 4:** ID 4, Name Harpreet4, Surname Bharia4, Date 2023-03-27Z, Test
- Employee 5:** ID 5, Name Harpreet5, Surname Bharia5, Date 2023-03-27Z, Test

The left window has fields "temp" and "Emplist" visible, while the right window has a field "002" visible.

68. From the Design selection, click **Assignment to tailist**.



The screenshot shows the Informatica Design workspace with a workflow diagram and its properties.

**Workflow Diagram:**

```

graph LR
    Start((Start)) --> Create[Create EmpList]
    Create --> Assignment1[Assignment to tailist]
    Assignment1 --> Assignment2[Assignment to Headlist]
    Assignment2 --> End((End))
  
```

A red box highlights the "Assignment to tailist" step. Green checkmarks are placed along the arrows indicating successful flow between steps.

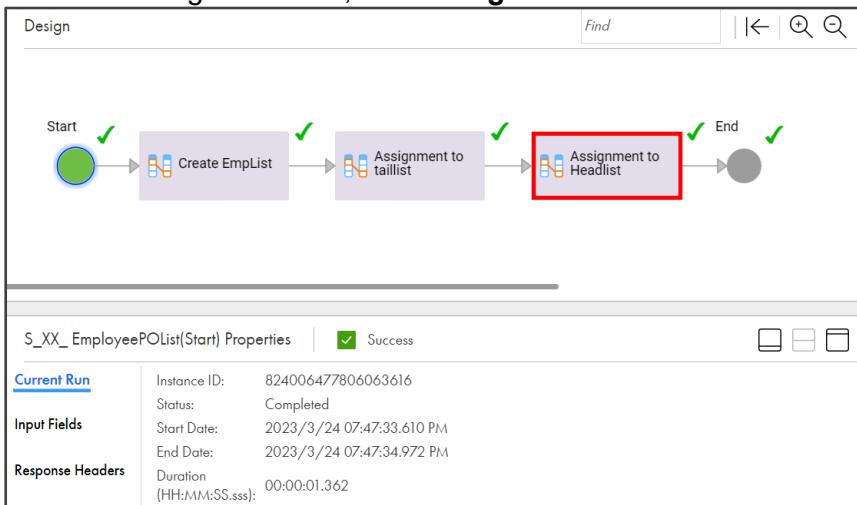
**S\_XX\_EmployeePOList(Start) Properties:**

| S_XX_EmployeePOList(Start) Properties |              | Success                     |
|---------------------------------------|--------------|-----------------------------|
| <input type="checkbox"/>              |              |                             |
| <b>Current Run</b>                    | Instance ID: | 824006477806063616          |
|                                       | Status:      | Completed                   |
|                                       | Start Date:  | 2023/3/24 07:47:33.610 PM   |
|                                       | End Date:    | 2023/3/24 07:47:34.972 PM   |
| <b>Input Fields</b>                   | Duration     | 00:00:01.362 (HH:MM:SS.sss) |
| <b>Response Headers</b>               |              |                             |

69. In the Create Assignment to taillist Properties section, observe the output. A list of all employees is displayed except the first one.

| Assignment to taillist Properties |          | Success                                  |
|-----------------------------------|----------|--|
| Fields                            |          |  |
|                                   |          | 2<br>Harpreet2<br>Bhatia2<br>2023-03-27Z |
|                                   |          | Test                                     |
|                                   |          | 3<br>Harpreet3<br>Bhatia3<br>2023-03-27Z |
|                                   |          | Test                                     |
| temp                              | taillist | 4<br>Harpreet4<br>Bhatia4                |

70. From the Design selection, click **Assignment to Headlist**.

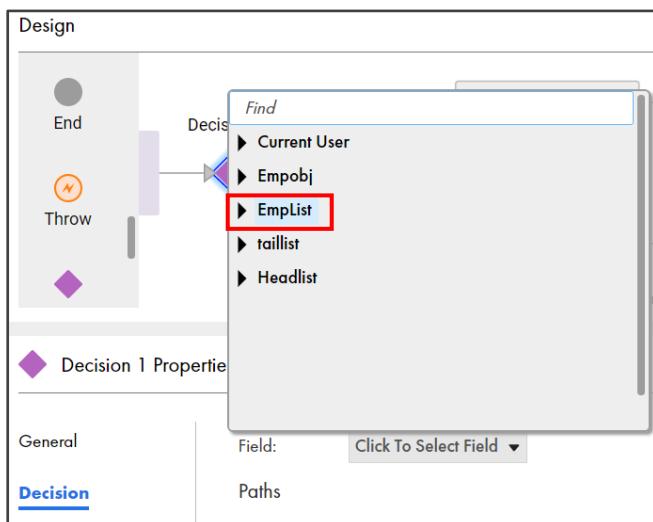


71. In the Create Assignment to Headlist Properties section, observe the output. Only the first employee is displayed.

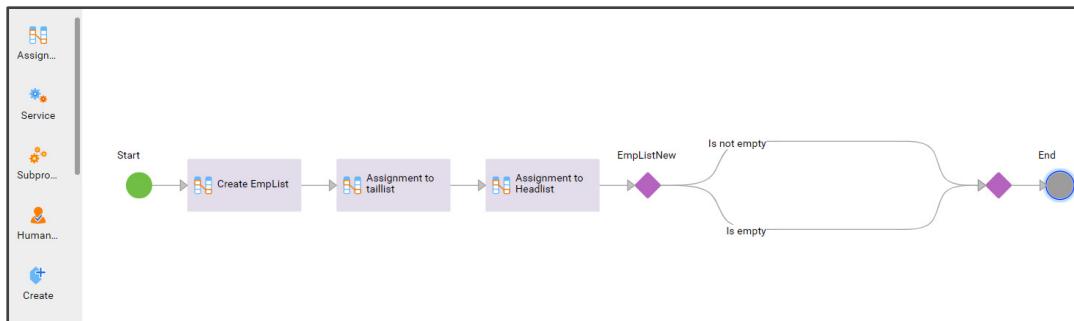
| Assignment to Headlist Properties |          | Success                                  |
|-----------------------------------|----------|--|
| Fields                            |          |  |
|                                   | Type     | Name                                     |
|                                   |          | Value                                    |
| temp                              | Headlist | 1<br>Harpreet1<br>Bhatia1<br>2023-03-27Z |
|                                   |          | Test                                     |

## Add a Decision step EmpListNew

72. From the Design palette, drag and drop a **Decision** step on the link between the Assignment to Headlist and End step.
73. To configure the step, from the process canvas, select the **Decision** step.
74. In the General section of Assignment properties, enter the Name as **EmpListNew**.
75. From the properties pane, select **Decision**.
76. From the Field drop-down, select **EmpList**.

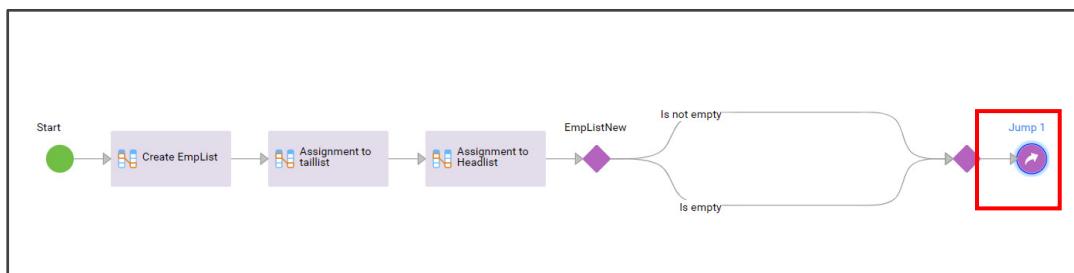


Your process should look like this:



Next, you will add a Jump step.

77. From the Design palette, drag a **Jump** step and drop it at the **End** step.

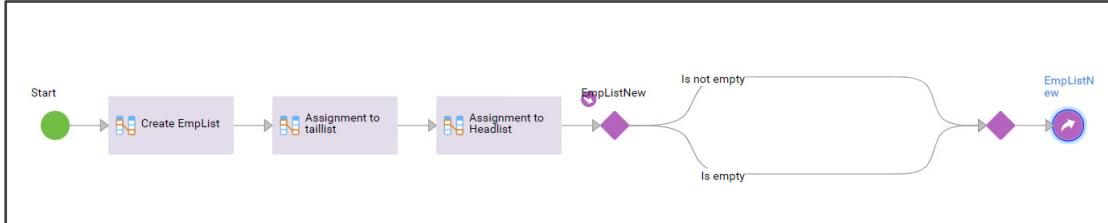


78. In the Jump properties, from the To drop-down, select **EmpListNew**.

**Jump 1 Properties**

To:

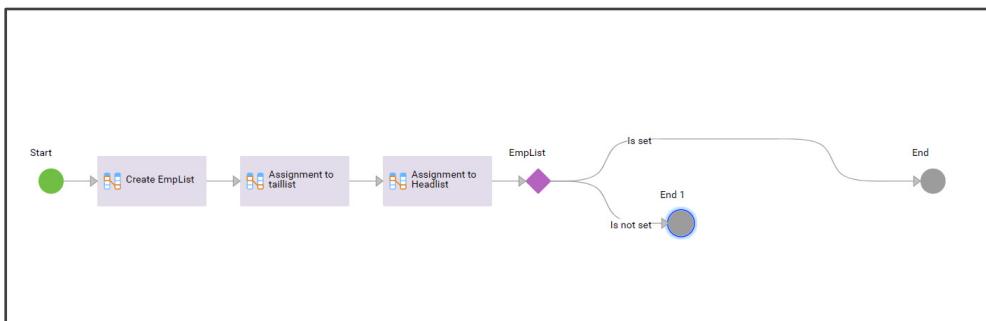
Assignment to Headlist  
 Assignment to tailist  
 Create EmpList  
**EmpListNew**



```

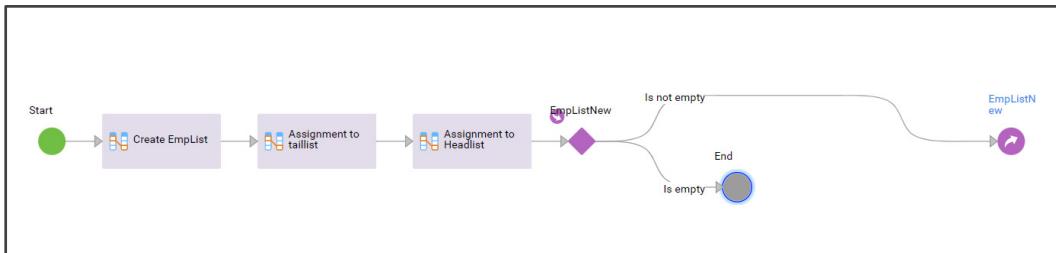
graph LR
    Start((Start)) --> Create[Create EmpList]
    Create --> Tailist[Assignment to tailist]
    Tailist --> Headlist[Assignment to Headlist]
    Headlist --> Decision{EmpListNew}
    Decision -- "Is not empty" --> Loop(( ))
    Decision -- "Is empty" --> End((End))
    
```

79. From the Design palette, drag and drop an **End** step to the **is empty** path of the Decision step.



80. In the General tab of the End properties, change the Name field to **End**.

Your process should look like this:

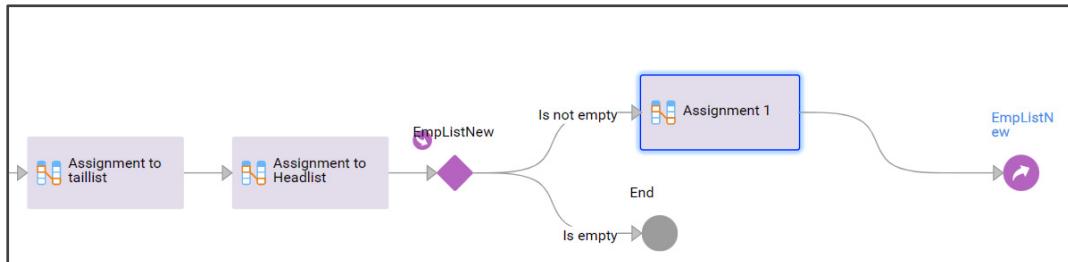


Next, you will add an Assignment step to display the last employee of the list.

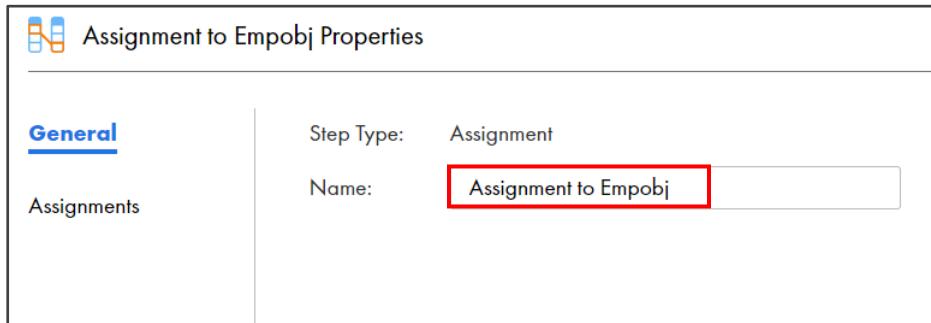
## Add an Assignment step Assignment to Empobj

Now, you will display the last element of the list after the list is read.

81. From the Design palette, drag and drop an **Assignment** step to the **is not empty** path of the Decision step.



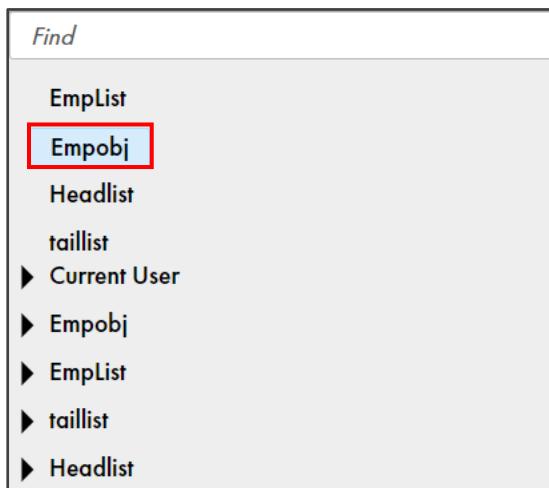
82. In the General tab of the Assignment properties, enter the Name as **Assignment to Empobj**.



83. From the properties pane, select **Assignments**.

84. Click **(+)** icon.

85. From the list, select **Empobj**.



86. From the **Assigned Using** drop-down, select **Formula**.

87. To open the Formula Editor, click **f(x)**.

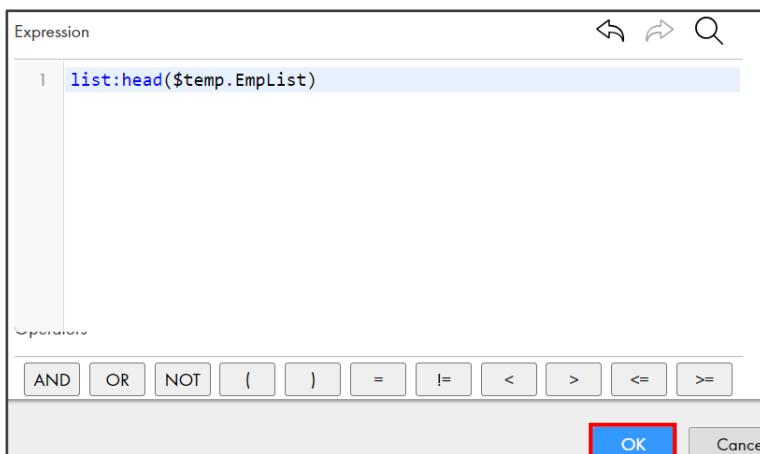
88. In the Expression field, enter the following expression:

**list:head(\$temp.EmpList)**

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **16\_LabGuide\_CAI\_Create a Process Using Lists Function of XQuery\_9-1**. Copy the command mentioned under **Step D** and paste it in the Expression field.

89. Click **OK**.

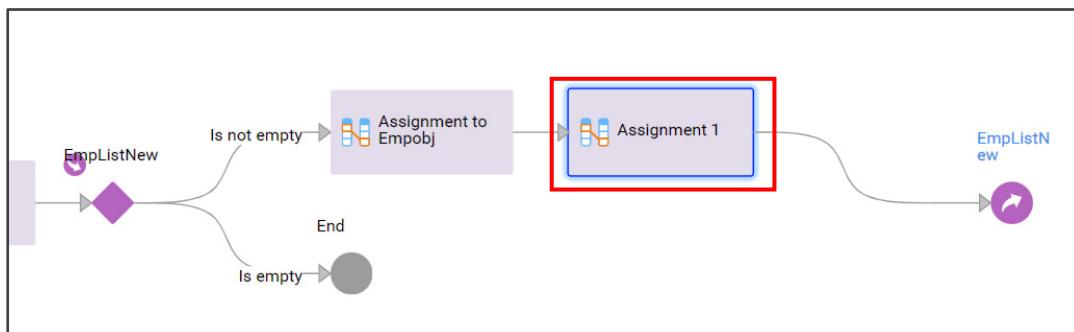


**Note:** After the list is read, the head function will return the last element of the list.

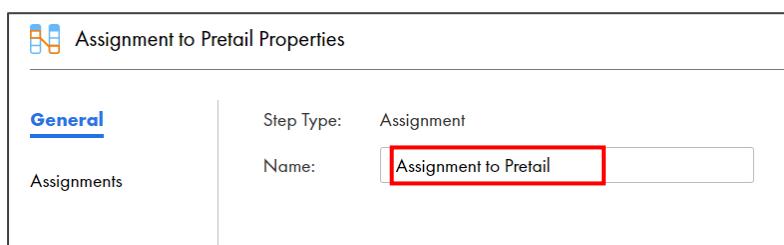
### Add an Assignment step Assignment to Pretail

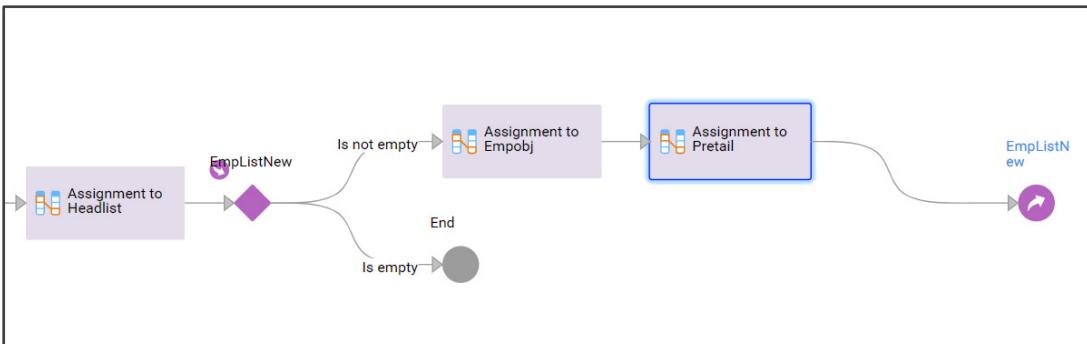
Now, you will add an Assignment step to count and return the total elements present in the list.

90. From the Design palette, drag and drop an **Assignment** step between Assignment to Empobj and Jump steps.



91. In the General tab of the Assignment properties, enter the Name as **Assignment to Pretail**.

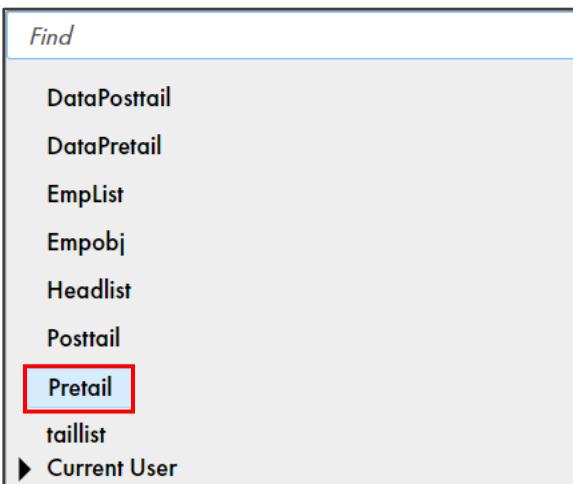




92. From the properties pane, select **Assignments**.

93. Click **(+)** icon.

94. From the list, select **Pretail**.



95. From the **Assigned Using** drop-down, select **Formula**.

96. To open the Formula Editor, click .

97. In the Expression field, enter the following expression:

```

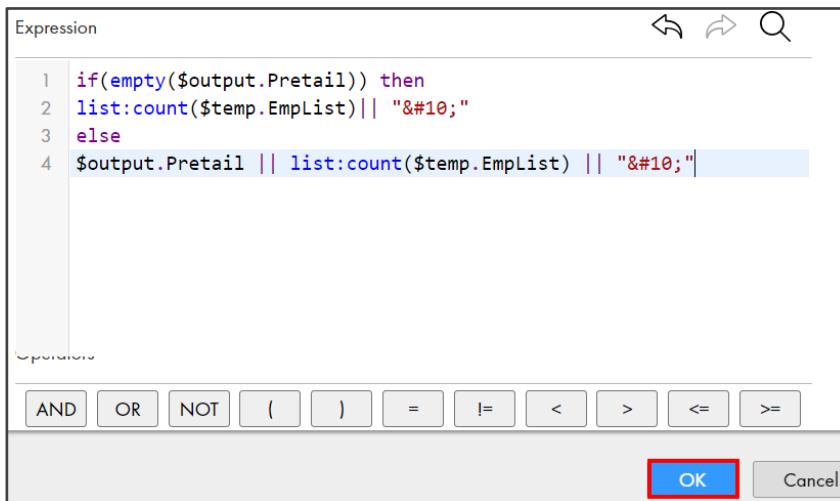
if(empty($output.Pretail)) then
  list:count($temp.EmpList)|| "
""
else
  $output.Pretail || list:count($temp.EmpList) || "
""
  
```

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **16\_LabGuide\_CAI\_Create a Process Using Lists Function of XQuery\_9-1**. Copy the command mentioned under **Step E** and paste it in the Expression field.

**Note:** The count function counts and returns the total elements present in an array. Here, **&#xA;** is the carriage return which will insert a line break to the beginning of each element.

98. Click **OK**.



The screenshot shows the Expression dialog box with the following XQuery code:

```

1 if(empty($output.Pretail)) then
2   list:count($temp.EmpList)|| "\#10;" 
3 else
4   $output.Pretail || list:count($temp.EmpList) || "\#10;" 

```

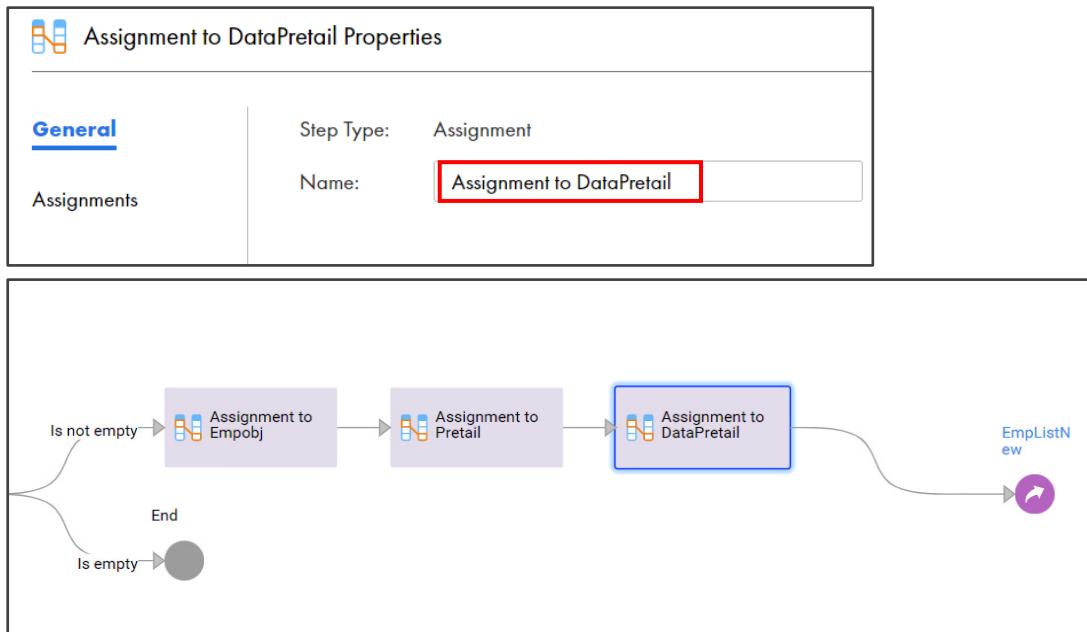
Below the code editor are several operators: AND, OR, NOT, (, ), =, !=, <, >, <=, >=. The OK button is highlighted with a red border.

### Add an Assignment step Assignment to DataPretail

Now, you will display all elements available in the list.

99. From the Design palette, drag and drop an **Assignment** step between Assignment to Pretail and Jump steps.

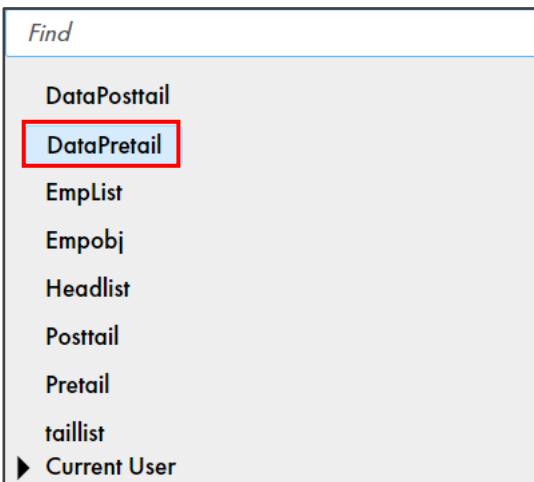
100. In the General tab of the Assignment properties, enter the Name as **Assignment to DataPretail**.



101. From the properties pane, select **Assignments**.

102. Click **(+)** icon.

103. From the list, select **DataPretail**.



104. From the **Assigned Using** drop-down, select **Formula**.

105. To open the Formula Editor, click .

106. In the Expression field, enter the following expression:

```
if(empty($output.DataPretail)) then
    $temp.Empobj[1]/FIRST_NAME || "
""
else
    $output.DataPretail || $temp.Empobj[1]/FIRST_NAME || "
"
```

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **16\_LabGuide\_CAI\_Create a Process Using Lists Function of XQuery\_9-1**. Copy the command mentioned under **Step F** and paste it in the Expression field.

107. Click **OK**.

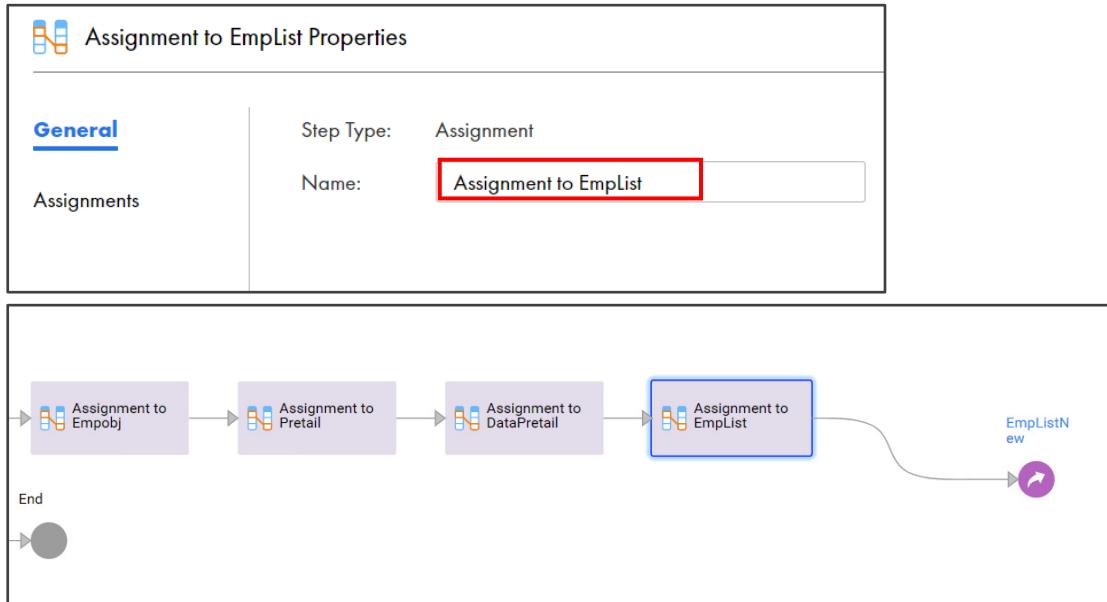


## Add an Assignment step Assignment to EmpList

In this step, entire list will be read, and empty list will be displayed.

108. From the Design palette, drag and drop an **Assignment** step between Assignment to DataPretail and Jump steps.

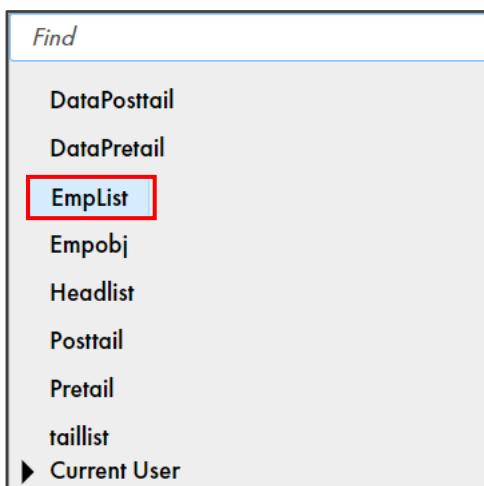
109. In the General tab of the Assignment properties, enter the Name as **Assignment to EmpList**.



110. From the properties pane, select **Assignments**.

111. Click (+) icon.

112. From the list, select **EmpList**.



113. From the **Assigned Using** drop-down, select **Formula**.

114. To open the Formula Editor, click .

115. In the Expression field, enter the following expression:

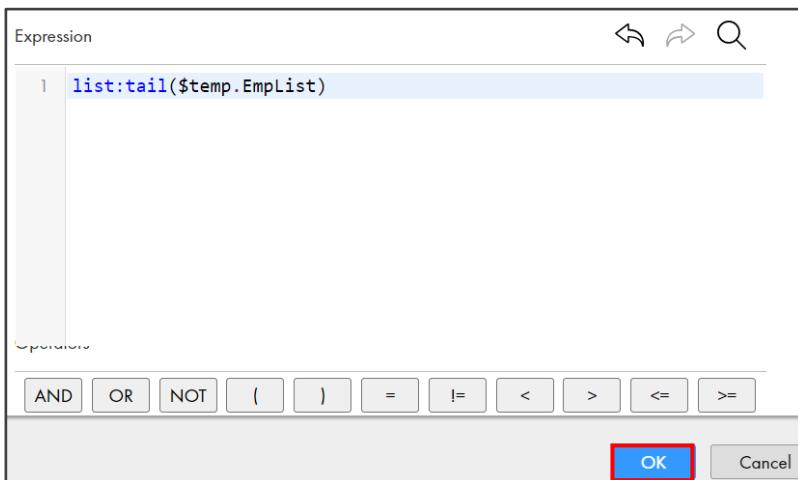
**list:tail(\$temp.EmpList)**

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named

**16\_LabGuide\_CAI\_Create a Process Using Lists Function of XQuery\_9-1**. Copy the command mentioned under **Step G** and paste it in the Expression field.

116. Click **OK**.

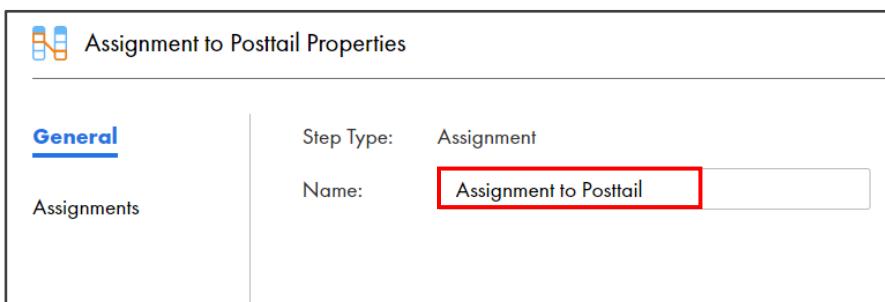


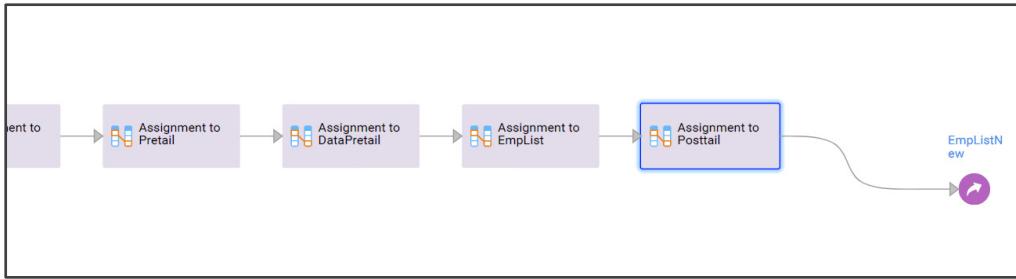
### Add an Assignment step Assignment to Posttail

In this step, the order of elements in the tail will be read and displayed.

117. From the Design palette, drag and drop an **Assignment** step between Assignment to EmpList and Jump steps.

118. In the General tab of the Assignment properties, enter the Name as **Assignment to Posttail**.

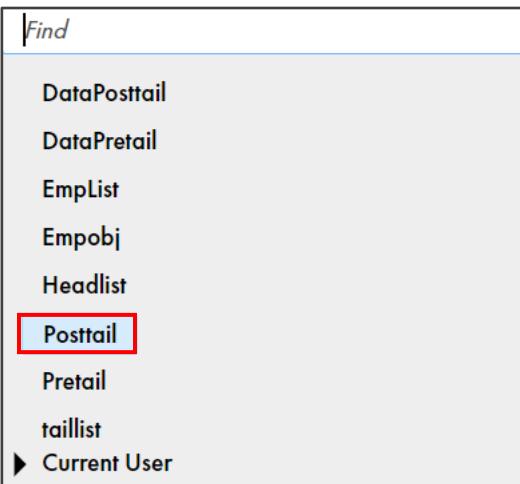




119. From the properties pane, select **Assignments**.

120. Click (+) icon.

121. From the list, select **Posttail**.



122. From the **Assigned Using** drop-down, select **Formula**.

123. To open the Formula Editor, click .

124. In the Expression field, enter the following expression:

```
if(empty($output.Posttail)) then
list:count($temp.EmpList)|| "
"  

else  

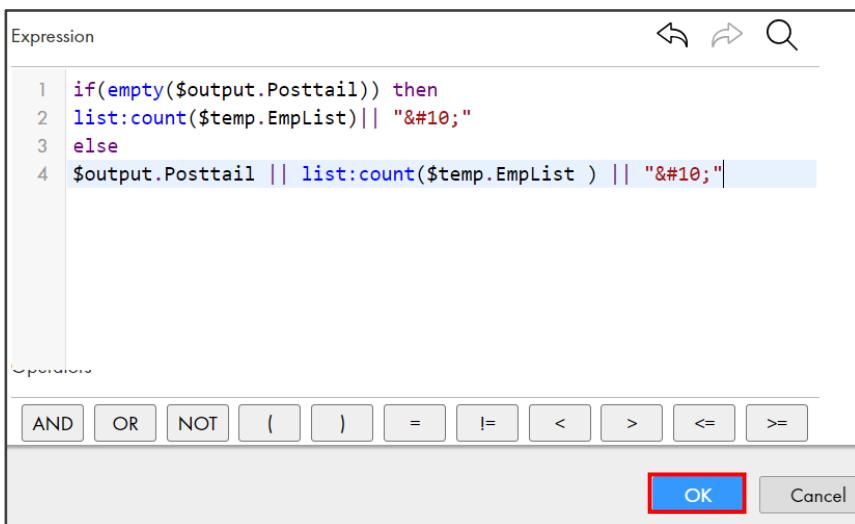
$output.Posttail || list:count($temp.EmpList ) || "
"
```

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named

**16\_LabGuide\_CAI\_Create a Process Using Lists Function of XQuery\_9-1**. Copy the command mentioned under **Step H** and paste it in the Expression field.

125. Click **OK**.



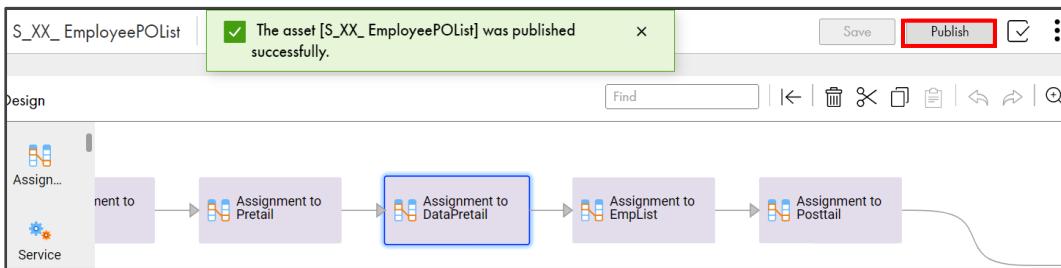
```

1 if(empty($output.Posttail)) then
2 list:count($temp.EmpList)|| "
" 
3 else
4 $output.Posttail || list:count($temp.EmpList ) || "
" 

```

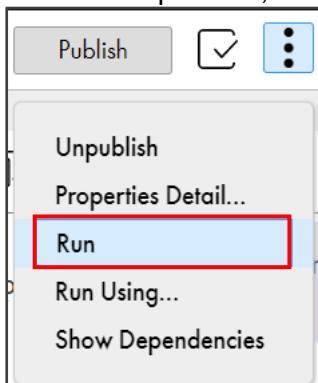
The screenshot shows an 'Expression' editor window. The code area contains the provided XQuery snippet. Below the code are various operators (AND, OR, NOT, etc.) and comparison operators (=, !=, <, >, <=, >=). At the bottom right are 'OK' and 'Cancel' buttons, with 'OK' being highlighted.

126. Save and Publish the process.

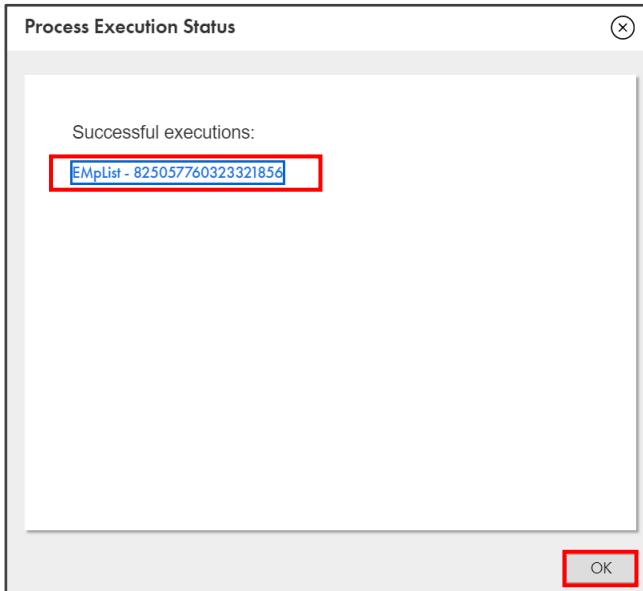


### Invoke the Process from the Interface

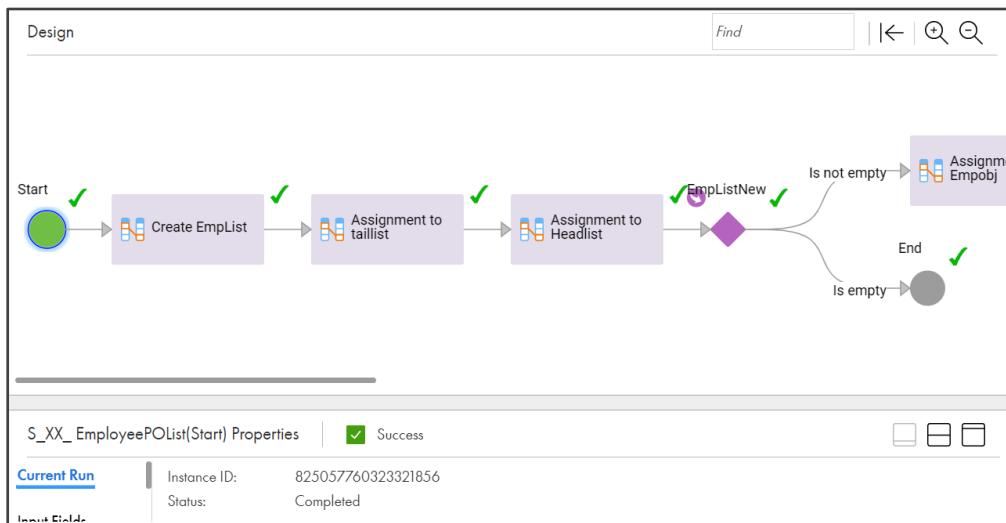
127. To run the process, click , and select **Run**.



128. Select the **Process Execution** link and click **OK**.

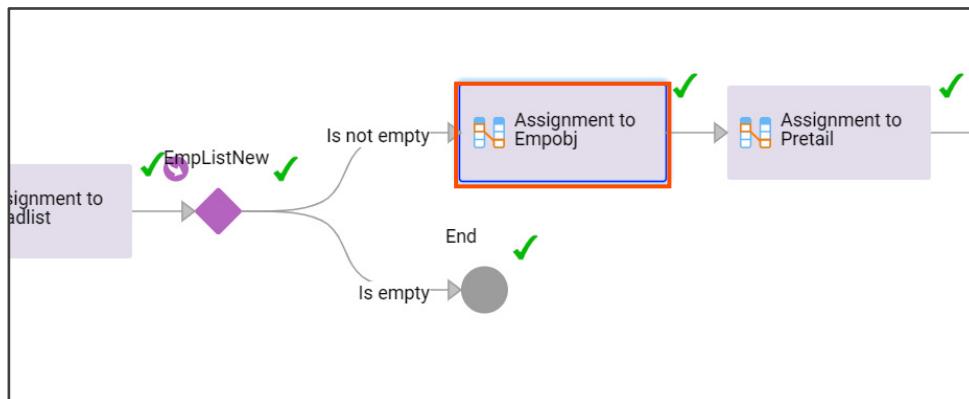


Process is executed successfully. The execution details are as shown below:



**Note:** If you are not able to see this screen, click **Refresh**.

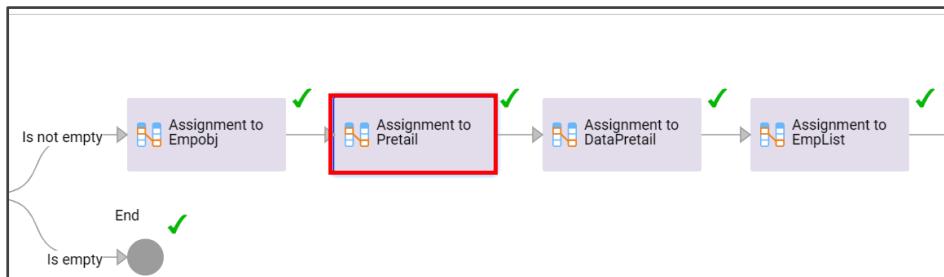
129. From the Design selection, click **Assignment to EmpObj**.



130. In the Assignment to EmpObj Properties section, observe the output. Only last employee in the list is displayed.

| Assignment to Empobj Properties |        |  |
|---------------------------------|--------|--|
| Fields                          |        |  |
| Type                            | Name   | Value                                    |
| temp                            | Empobj | 5<br>Harpreet5<br>Bhatia5<br>2023-03-27Z |
|                                 |        | Test                                     |

131. From the Design selection, click **Assignment to Pretail**.

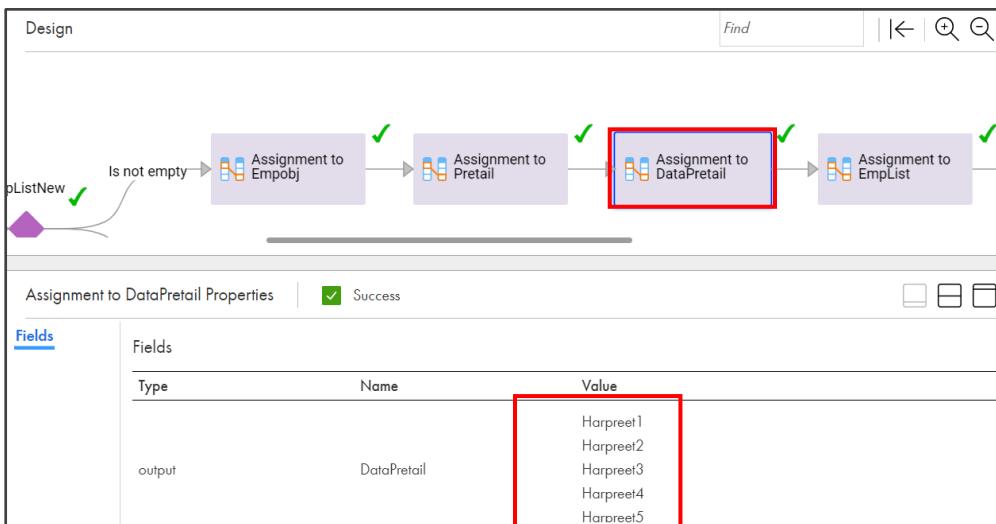


132. In the Assignment to Pretail Properties section, observe the output. You can see the total counts of the employees present in the array.

| Assignment to Pretail Properties |         | <input checked="" type="checkbox"/> Success |
|----------------------------------|---------|---|
| Fields                           |         |   |
| Type                             | Name    | Value                                       |
| output                           | Pretail | 5<br>4<br>3<br>2<br>1                       |

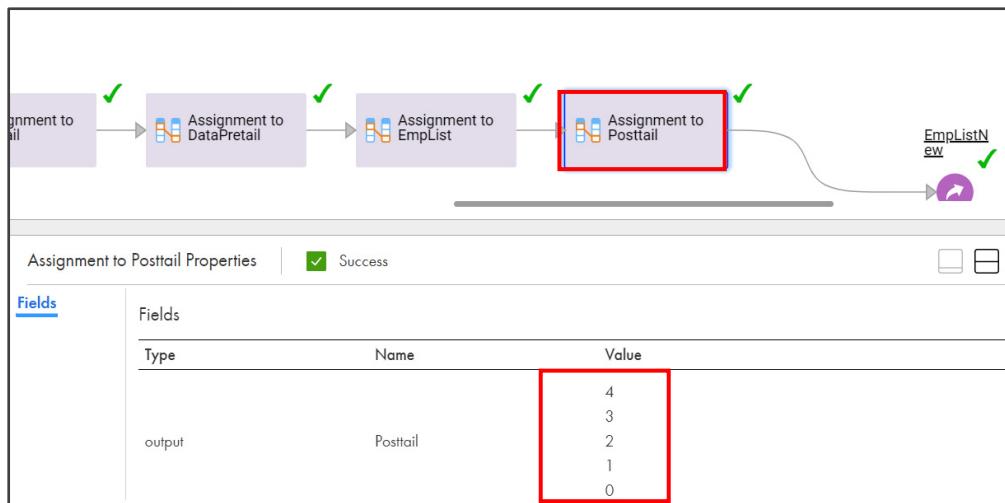
133. From the Design selection, click **Assignment to DataPretail**.

134. In the Assignment to DataPretail Properties section, observe the output. You can see the list of total employees present in the array.



135. From the Design selection, click **Assignment to Posttail**.

136. In the Assignment to Posttail Properties section, observe the output. The order of elements in the tail is displayed.




---

*This concludes the lab.*

# Module 10: Adding Web Service to a Process

## Lab 10-1: Invoke a Synchronous Web Service Call

### Overview:

In this lab, you will create two processes. One process runs on the selected Secure Agent and the other process is executed on the cloud server. Using the Service URL generated by the process that executes on the cloud server, you will run the process using synchronous web service call. The process objects and connections utilized in this process are already created in the previous modules.

**Note:** This lab is the last part of a set of three labs. Earlier, you created process objects (Lab 4-1: Create Process Objects) and then created a JDBC connection (Lab 7-2: Create a JDBC Connection). Using the assets created in the previous labs, you will create a process to obtain the exchange rates for the specified currencies.

### Objectives:

- Create ExchangeRate Process
- Create ExchangeRateService Process
- Execute the ExchangeRateService process using a Rest Client

### Duration:

30 Minutes

---

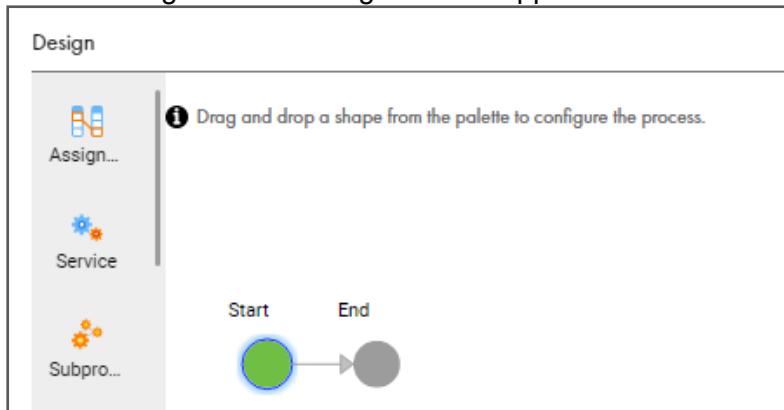
### Tasks:

**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

#### Create a new process

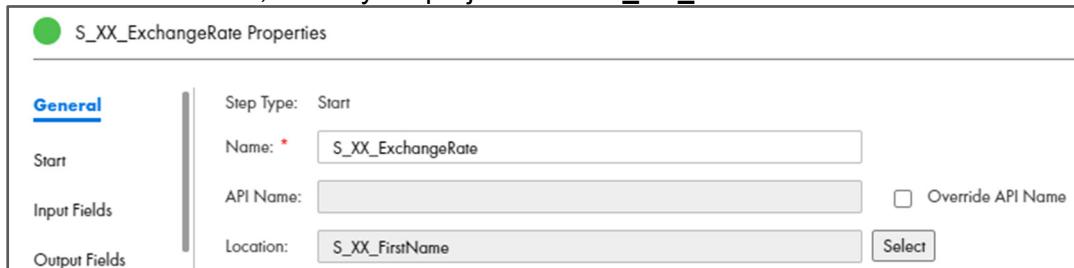
1. Create a new Process.

The following Process Design canvas appears with the Start and End steps.



2. From the process canvas, select the **Start** step.
3. In the General tab, enter the name as **S\_XX\_ExchangeRate**.
4. Skip the API Name option.

5. In the Location field, select your project folder **S\_XX\_FirstName**.

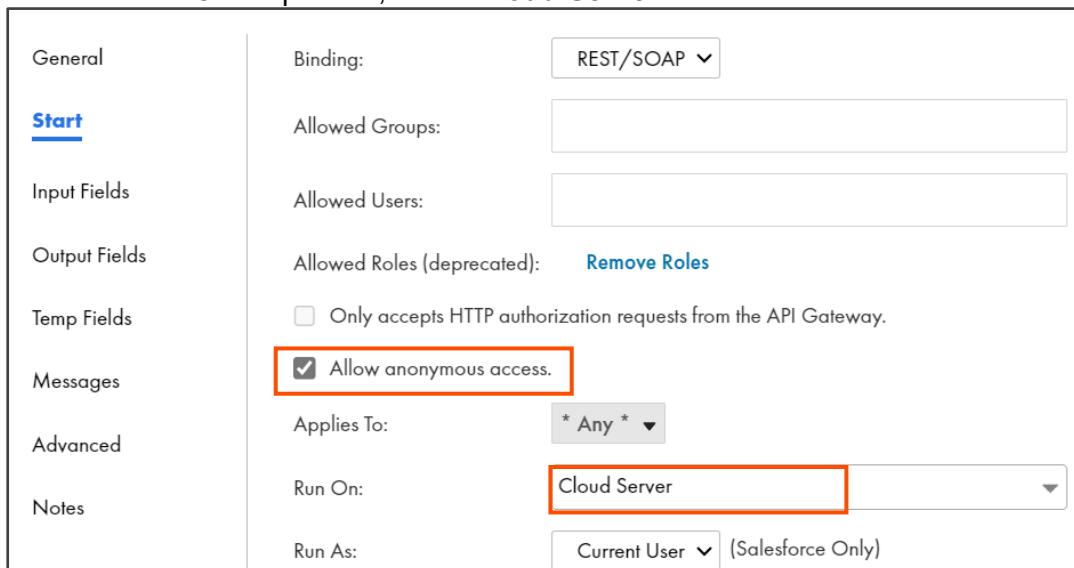


S\_XX\_ExchangeRate Properties

**General**

Step Type: Start  
 Name: \* S\_XX\_ExchangeRate  
 API Name:   
 Override API Name  
 Location: S\_XX\_FirstName

6. From the properties pane, select **Start**.  
 7. Select the **Allow anonymous access** option.  
 8. From the Run On drop-down, retain **Cloud Server**.



|               |   |
|---------------|---|
| General       | Binding: REST/SOAP  |
| <b>Start</b>  | Allowed Groups:   |
| Input Fields  | Allowed Users:  |
| Output Fields | Allowed Roles (deprecated): <a href="#">Remove Roles</a>                                |
| Temp Fields   | <input type="checkbox"/> Only accepts HTTP authorization requests from the API Gateway. |
| Messages      | <input checked="" type="checkbox"/> Allow anonymous access.                             |
| Advanced      | Applies To: * Any *   |
| Notes         | Run On: <b>Cloud Server</b>   |
|               | Run As: Current User (Salesforce Only)  |

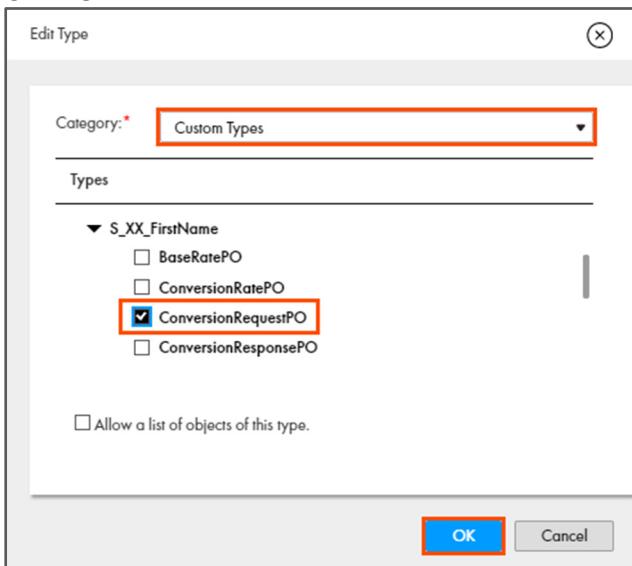
**Note:** Allowed Users and Allowed Groups textboxes remain blank as you have selected the Allow anonymous access checkbox.

9. From the properties pane, select **Input Fields**.  
 10. Add a new input field as shown in the table below:

| Name            | Type                      |
|-----------------|---------------------------|
| ExchangeRequest | More Types > Custom Types |

11. Expand your project folder and select **ConversionRequestPO**.

12. Click **OK**.



13. Select the **Required** checkbox.

| Input Format: <input checked="" type="radio"/> Fields <input type="radio"/> Whole payload |                                      |             |                                     |
|---|--------------------------------------|-------------|-------------------------------------|
| Input Fields (1)  |                                      |             |                                     |
| Name  | Type                                 | Description | Required                            |
| ExchangeRequest   | ConversionRequestPO (S_XX_FirstName) |             | <input checked="" type="checkbox"/> |

14. To add new output fields, in the **Output Fields** tab, click .

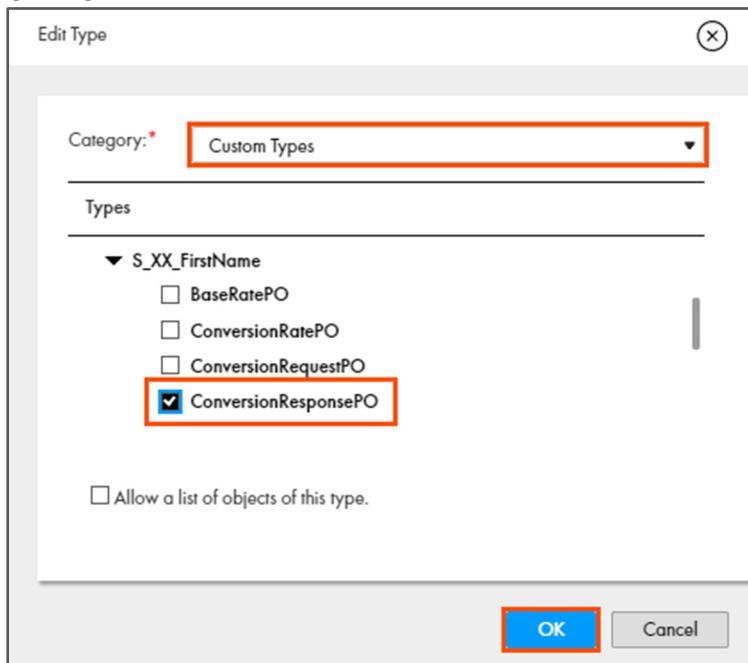
| General<br><br>Start<br><br>Input Fields<br><br><b>Output Fields</b><br><br>Temp Fields | Output Format: <input checked="" type="radio"/> Fields <input type="radio"/> Whole payload |      |             |               |
|---|--|------|-------------|---------------|
|   | Output Fields (0)  |      |             |               |
|   | Name   | Type | Description | Initial Value |
|   | No data to display   |      |             |               |
|   |  |      |             |               |
|   |  |      |             |               |

15. Create the output fields, as shown in the table below:

| Name             | Type                      |
|------------------|---------------------------|
| ExchangeResponse | More Types > Custom Types |

16. Expand your project folder and select **ConversionResponsePO**.

17. Click **OK**.



18. From the Output Format field, select **Whole payload**.

| General              | Output Format: <input type="radio"/> Fields <input checked="" type="radio"/> Whole payload   |             |               |             |               |                  |                                       |  |  |
|----------------------|--|-------------|---------------|-------------|---------------|------------------|---------------------------------------|--|--|
| Start                | Output Fields [1]  |             |               |             |               |                  |                                       |  |  |
| Input Fields         |  |             |               |             |               |                  |                                       |  |  |
| <b>Output Fields</b> | <table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Description</th> <th>Initial Value</th> </tr> </thead> <tbody> <tr> <td>ExchangeResponse</td> <td>ConversionResponsePO [S_XX_FirstName]</td> <td></td> <td></td> </tr> </tbody> </table> | Name        | Type          | Description | Initial Value | ExchangeResponse | ConversionResponsePO [S_XX_FirstName] |  |  |
| Name                 | Type   | Description | Initial Value |             |               |                  |                                       |  |  |
| ExchangeResponse     | ConversionResponsePO [S_XX_FirstName]  |             |               |             |               |                  |                                       |  |  |
| Temp Fields          |  |             |               |             |               |                  |                                       |  |  |

19. From the properties pane, select **Temp Fields**.

20. Add a new **Temp** field as shown in the table below:

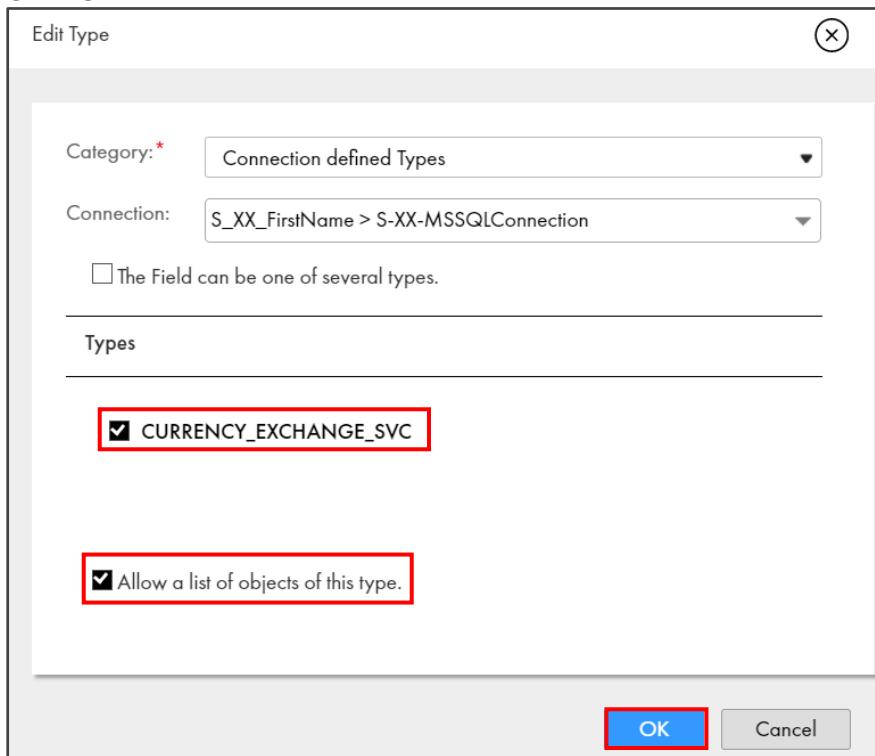
| Name                        | Type                                  |
|-----------------------------|---------------------------------------|
| ListofCurrencyExchangeRates | More Types > Connection defined Types |

**Note:** This temp field is used to fetch the list of all currency exchange rates from the MS SQL table CURRENCY\_EXCHANGE\_SVC.

21. From the Connection drop-down, expand your project folder, and select **S-XX-MSSQLConnection**.

22. Select **CURRENCY\_EXCHANGE\_SVC** and **Allow a list of objects of this type** checkboxes.

23. Click **OK**.



24. Create another temp field, as shown in the table below:

| Name         | Type |
|--------------|------|
| SQLCondition | Text |

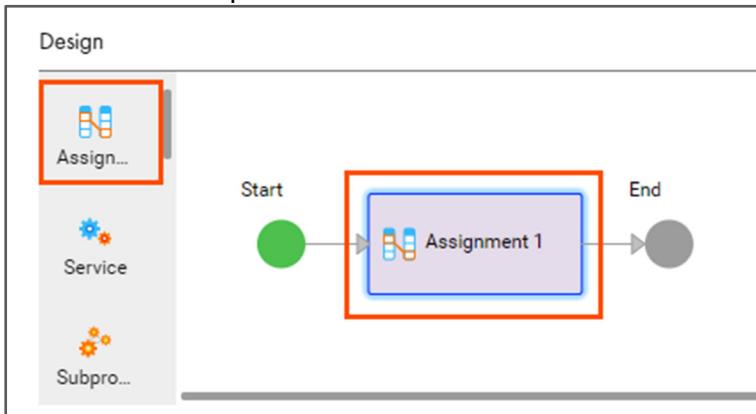


| Name         | Type |
|--------------|------|
| SQLCondition | Text |

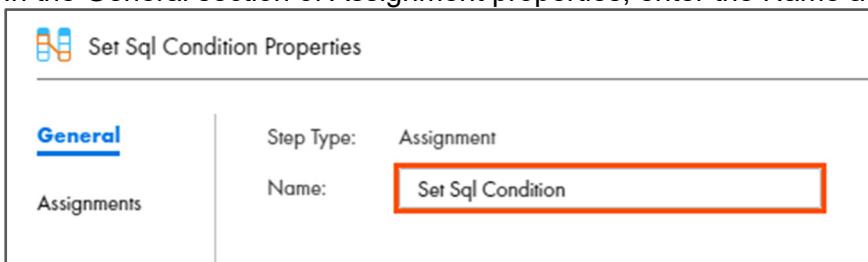
**Note:** This temp field is created to store the SQL condition that converts all the currency rates.

25. **Save** the process.

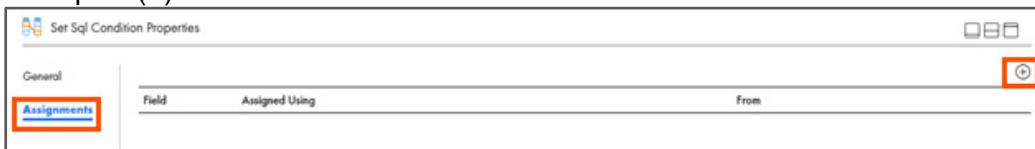
26. From the Design palette, drag and drop an **Assignment** step on the link between the Start and End steps.



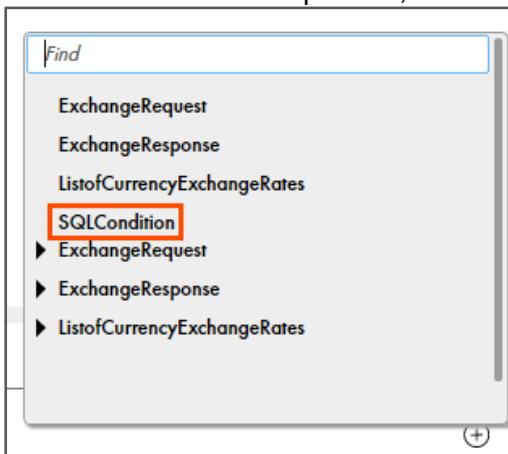
27. To configure the step, from the process canvas, select the **Assignment** step.  
 28. In the General section of Assignment properties, enter the Name as **Set Sql Condition**.



29. From the properties pane, select **Assignments**.  
 30. Click plus (+) icon.



31. From the Add Field drop-down, select **SQLCondition**.



32. From the Assigned Using drop-down, select **Formula**.

33. To open the Formula Editor, click .



**Note:** A new Expression Editor for SQLCondition window appears.

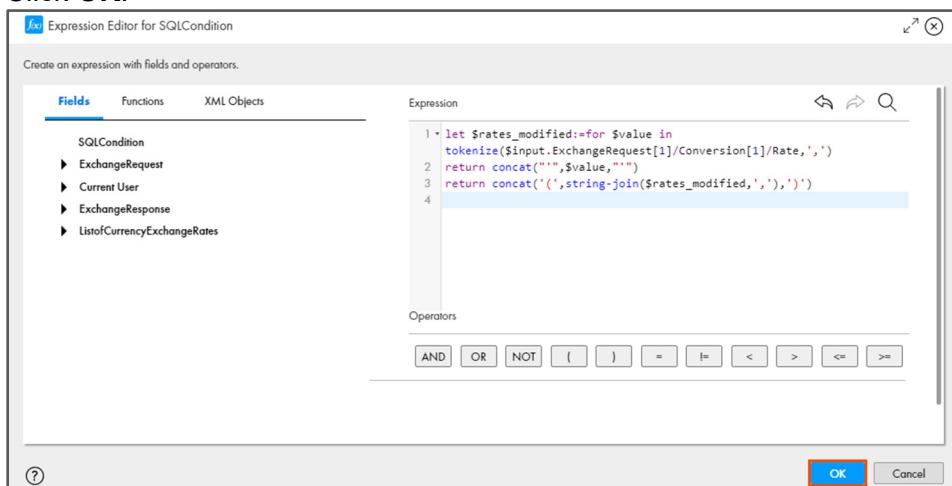
34. In the Expression field, enter the following expression:

```
let $rates_modified:=for $value in
 tokenize($input.ExchangeRequest[1]/Conversion[1]/Rate,',')
return concat("",$value,"")
return concat('(',string-join($rates_modified,''),')')
```

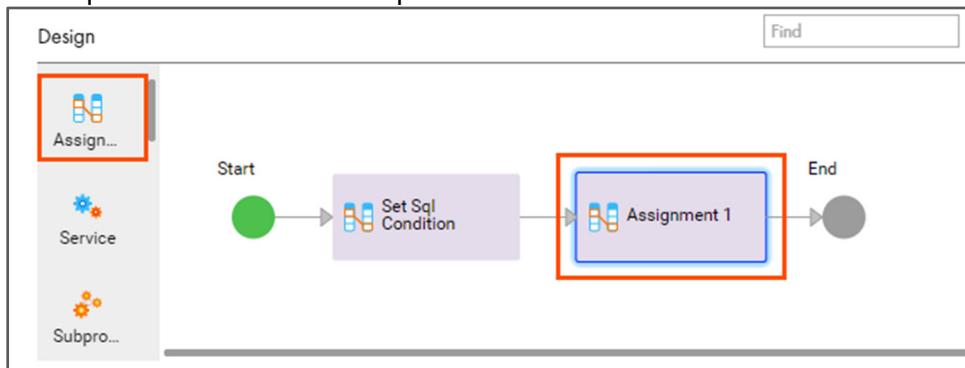
OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **17\_LabGuide\_CAI\_SynchronousWebServiceCall\_10-1**. Copy the command mentioned under **Step A** and paste it in the Expression field.

35. Click **OK**.



36. From the Design palette, drag and drop an **Assignment** step on the link between the Set Sql Condition and End step.



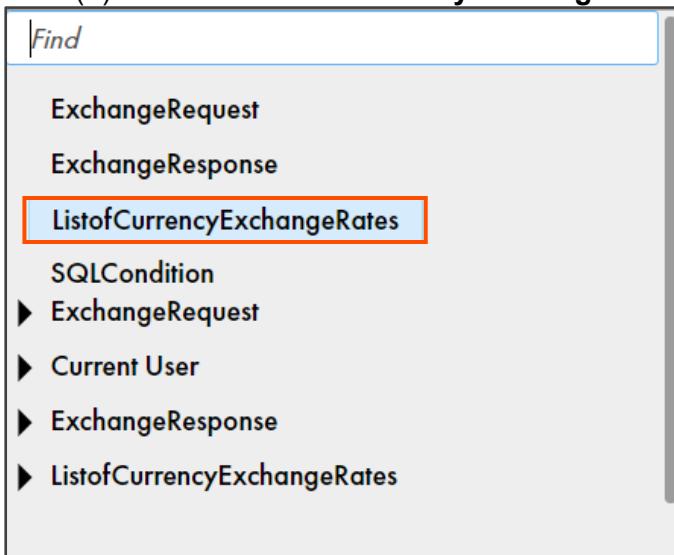
37. To configure the step, from the process canvas, select the **Assignment** step.

38. In the General section of Assignment properties, enter the Name as **Set Sql Condition\_query**.



39. From the properties pane, select **Assignments**.

40. Click (+) and select **ListofCurrencyExchangeRates**.



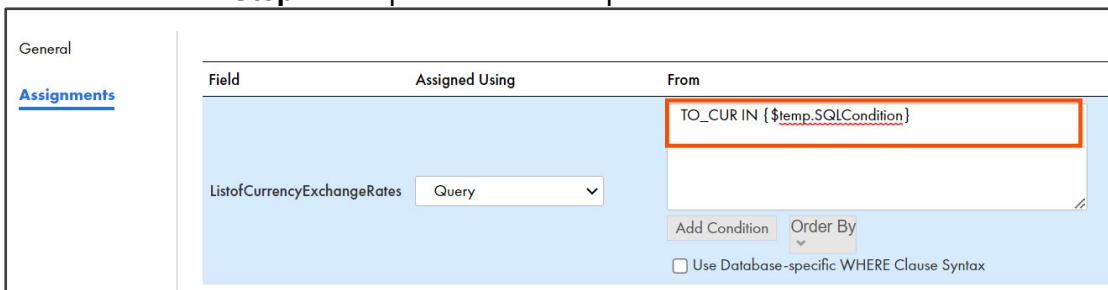
41. From the Assigned Using drop-down, select **Query**.

42. In the Query field, enter the following formula:

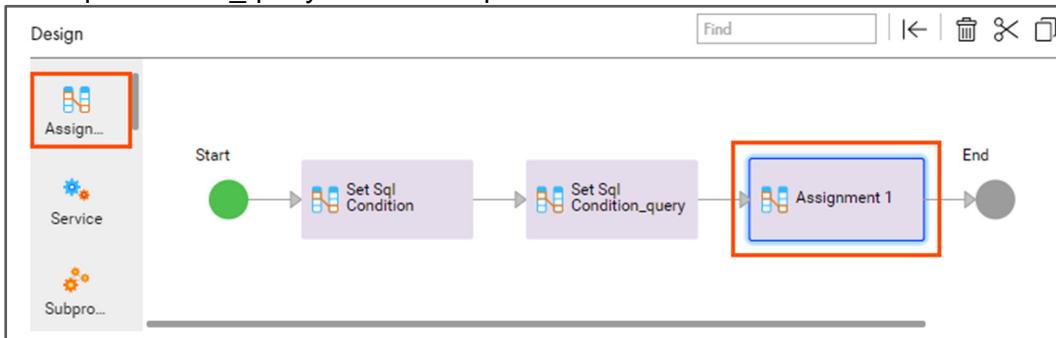
**TO\_CUR IN {\$temp.SQLCondition}**

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **17\_LabGuide\_CAI\_SynchronousWebServiceCall\_10-1**. Copy the command mentioned under **Step B** and paste it in the Expression field.

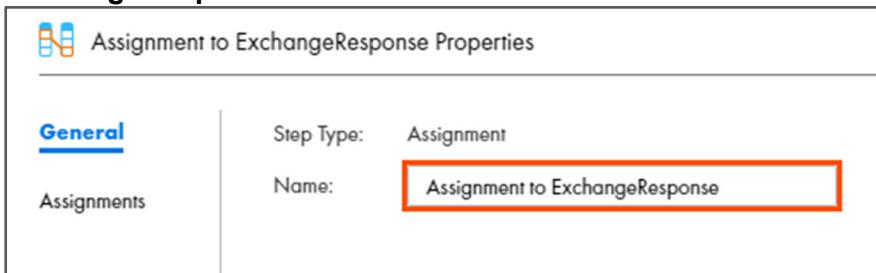


43. From the Design palette, drag and drop an **Assignment** step on the link between the Set Sql Condition\_query and End step.



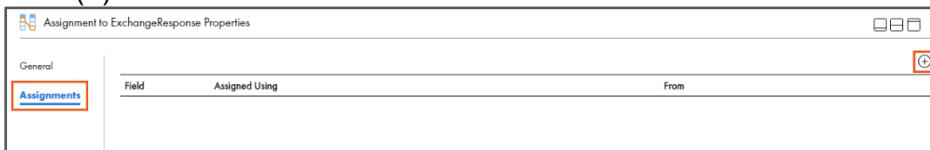
44. To configure the step, from the process canvas, select the **Assignment** step.

45. In the General section of Assignment properties, enter the Name as **Assignment to ExchangeResponse**.

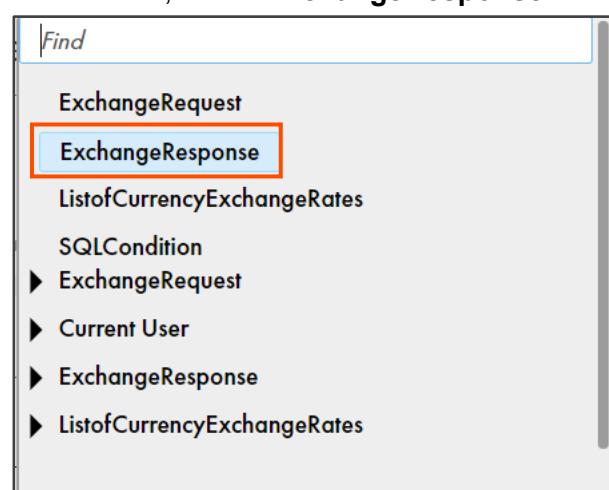


46. From the properties pane, select **Assignments**.

47. Click (+) icon.



48. From the list, select **ExchangeResponse**.



49. From the Assigned Using drop-down, select **Formula**.



50. To open the Formula Editor, click .



| Field            | Assigned Using | From  |
|------------------|----------------|---|
| ExchangeResponse | Formula        |  |

**Note:** A new Expression Editor for ExchangeResponse window appears.

51. In the Expression field, enter the following expression:

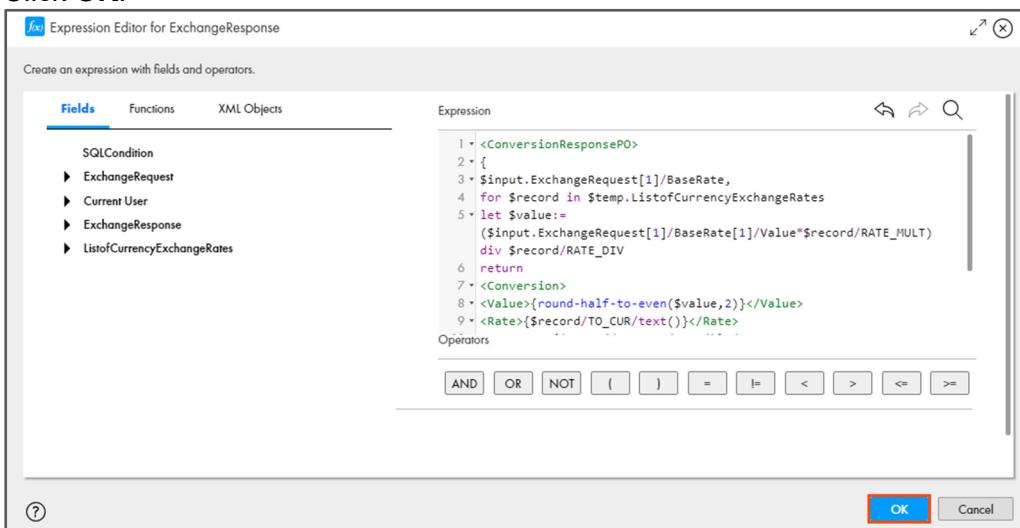
<ConversionResponsePO>

```
{
$input.ExchangeRequest[1]/BaseRate,
for $record in $temp.ListofCurrencyExchangeRates
let $value:=($input.ExchangeRequest[1]/BaseRate[1]/Value*$record/RATE_MULT)
    div $record/RATE_DIV
return
<Conversion>
<Value>{round-half-to-even($value,2)}</Value>
<Rate>{$record/TO_CUR/text()}</Rate>
<RateType>{$record/RT_TYPE/text()}</RateType>
</Conversion>
}
</ConversionResponsePO>
```

OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **17\_LabGuide\_CAI\_SynchronousWebServiceCall\_10-1**. Copy the command mentioned under **Step C** and paste it in the Expression field.

52. Click **OK**.

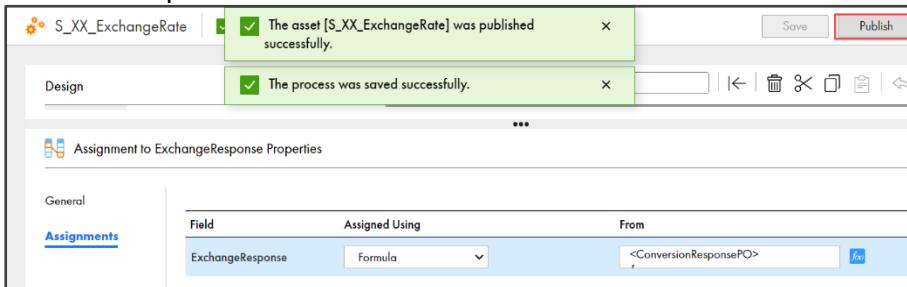


The dialog box shows the following interface:

- Fields:** SQLCondition, ExchangeRequest, Current User, ExchangeResponse, ListofCurrencyExchangeRates.
- Functions:**
- XML Objects:**
- Expression:** The code block provided in step 51.
- Operators:** AND, OR, NOT, (, ), =, !=, <, >, <=, >=.
- Buttons:** OK (highlighted), Cancel.

53. **Save** the process.

54. **Publish** the process.



The screenshot shows the Informatica interface with two success notifications at the top:

- The asset [S\_XX\_ExchangeRate] was published successfully.
- The process was saved successfully.

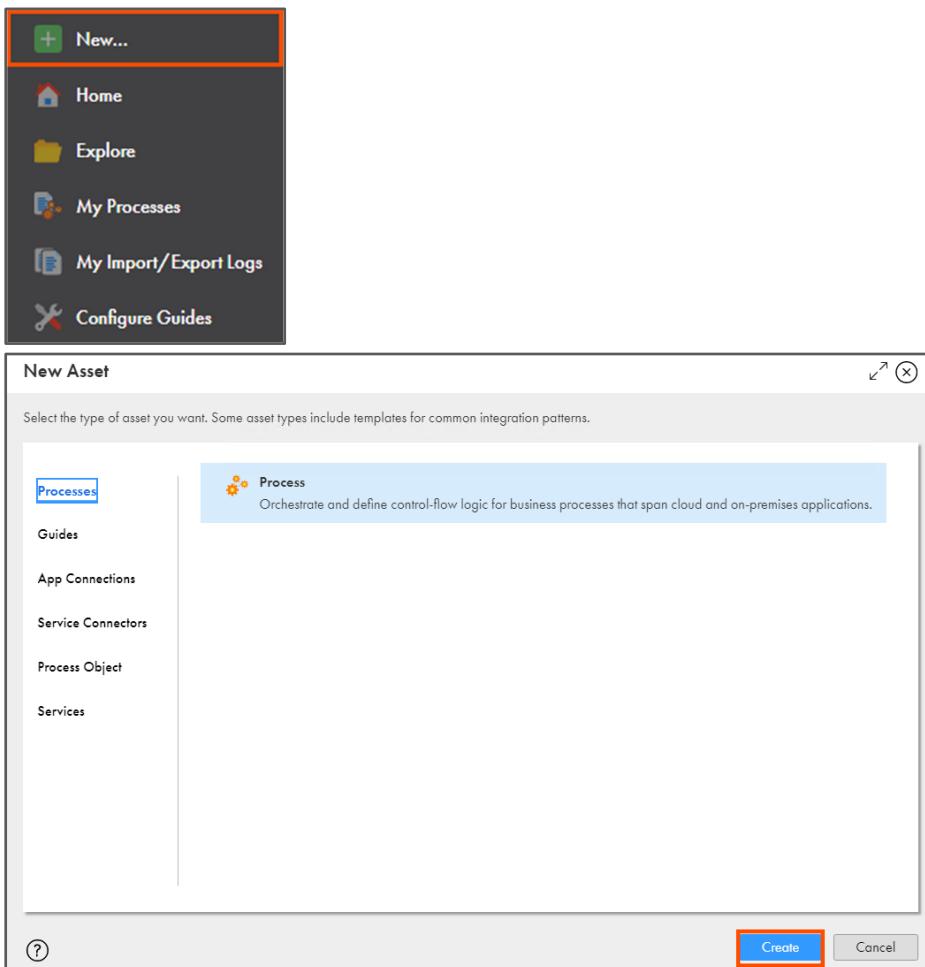
The "Assignments" tab is selected in the sidebar, and a table shows an assignment for the "ExchangeResponse" field, assigned using a formula from the "ConversionResponsePO" object.

You will get a notification that '**The asset [S-XX-ExchangeRate] was published successfully.**'.

#### Configure ExchangeRateService Process Properties:

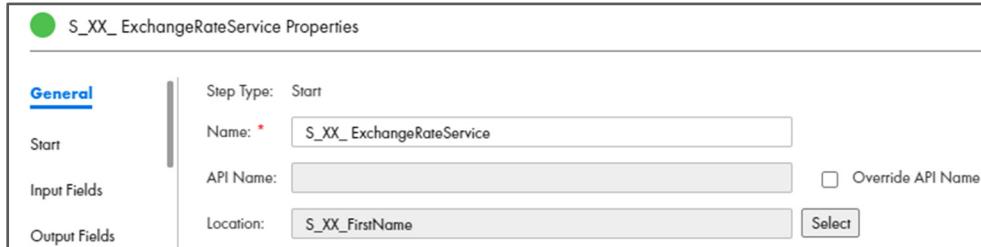
This is the parent process that runs on the Cloud Server and has the ExchangeRate process embedded in it. This process will be exposed as both REST and SOAP webservices for third-party access.

55. Create a new **Process**.



The "New Asset" dialog is open, showing the "Processes" category selected on the left. A "Process" item is highlighted with a blue box, showing its description: "Orchestrate and define control-flow logic for business processes that span cloud and on-premises applications." At the bottom right of the dialog are "Create" and "Cancel" buttons, with the "Create" button also highlighted with a blue box.

56. From the process canvas, select the **Start** step.
57. In the General tab, enter the name as **S\_XX\_ ExchangeRateService**.
58. Skip the API Name option.
59. In the Location field, select your project folder **S\_XX\_FirstName**.

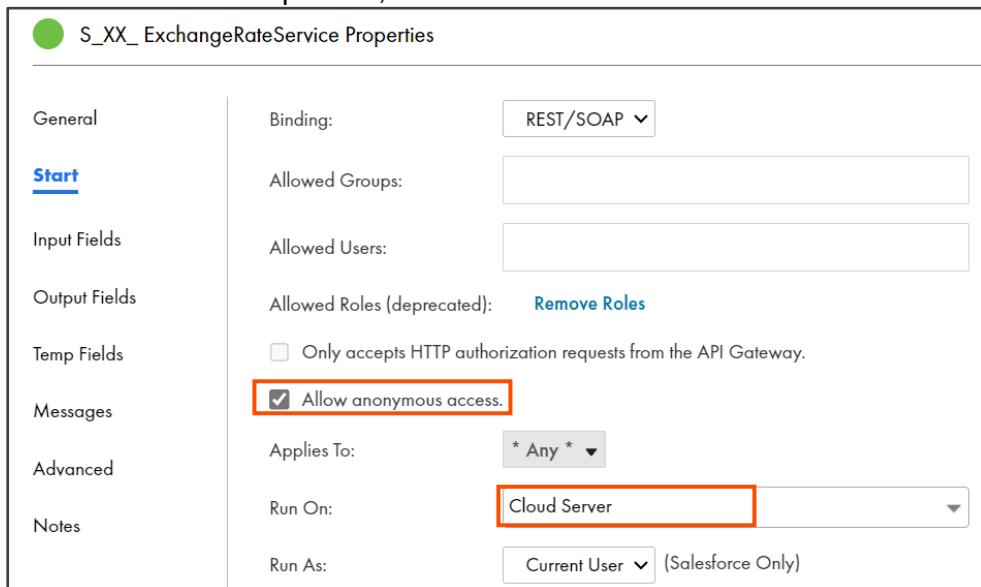


S\_XX\_ ExchangeRateService Properties

**General**

Step Type: Start  
 Name: \* S\_XX\_ ExchangeRateService  
 API Name:   
 Override API Name  
 Location: S\_XX\_FirstName

60. From the properties pane, select **Start**.
61. Select the **Allow anonymous access** option.
62. From the Run On drop-down, retain **Cloud Server**.



S\_XX\_ ExchangeRateService Properties

**Start**

Binding: REST/SOAP  
 Allowed Groups:  
 Allowed Users:  
 Allowed Roles (deprecated): [Remove Roles](#)  
 Only accepts HTTP authorization requests from the API Gateway.  
 Allow anonymous access.  
 Applies To: \* Any \*  
 Run On: Cloud Server  
 Run As: Current User (Salesforce Only)

**Note:** Allowed Users and Allowed Groups textboxes remain blank as you have selected the Allow anonymous access checkbox.

63. From the properties pane, select **Input Fields**.

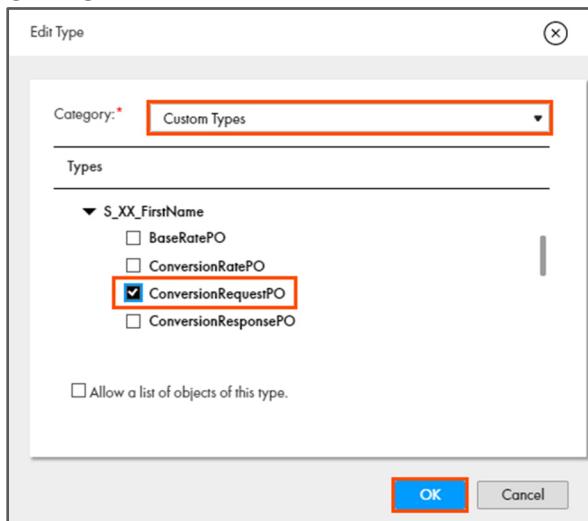
64. To add a new input field, click .

65. Create an input field, as shown in the table below:

| Name            | Type                      |
|-----------------|---------------------------|
| ExchangeRequest | More Types > Custom Types |

66. Expand your project folder and select **ConversionRequestPO**.

67. Click **OK**.



68. From the properties pane, select **Output Fields**.

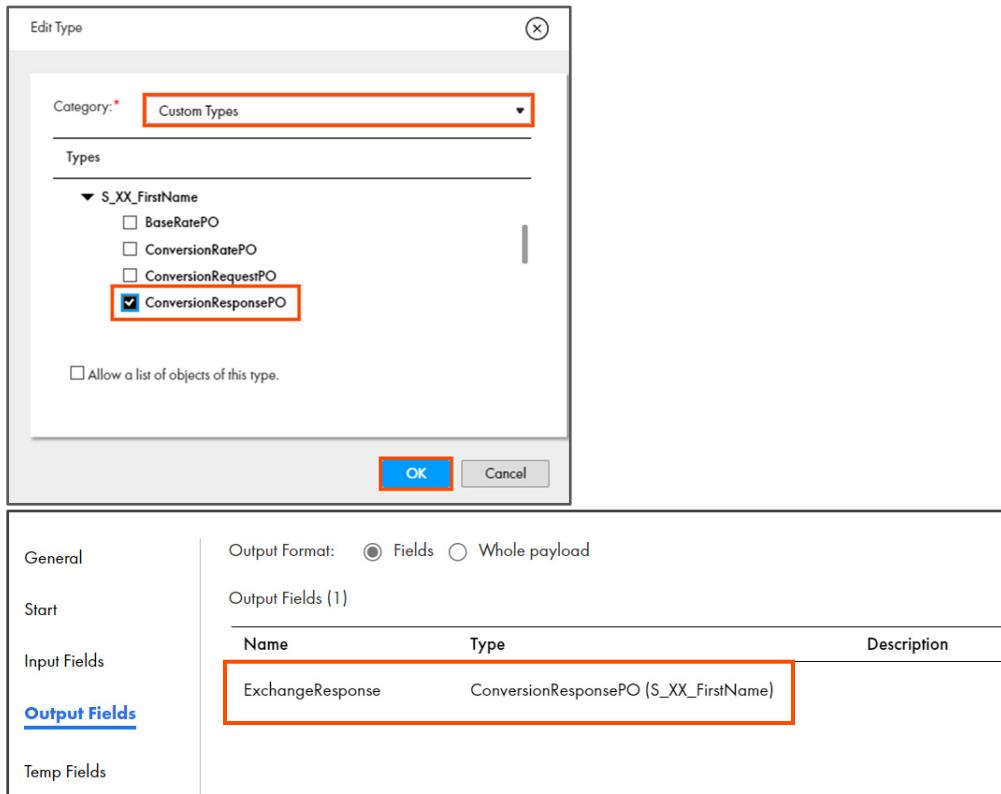
69. To add a new output field, click .

70. Create an output field, as shown in the table below:

| Name             | Type                      |
|------------------|---------------------------|
| ExchangeResponse | More Types > Custom Types |

71. Expand your project folder and select **ConversionResponsePO**.

72. Click **OK**.



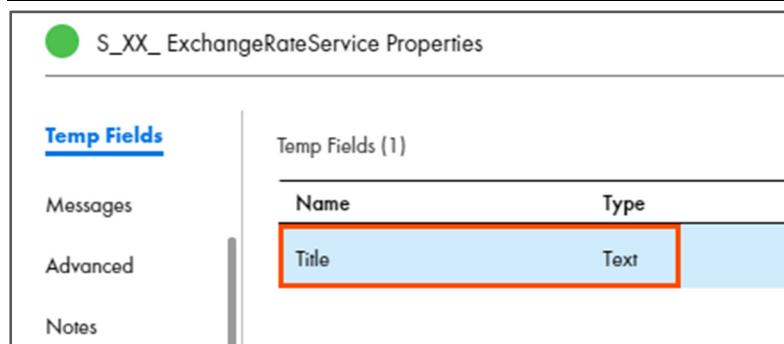
| General              | Output Format:    | <input checked="" type="radio"/> Fields | <input type="radio"/> Whole payload |
|----------------------|-------------------|---|-------------------------------------|
| Start                | Output Fields (1) |   |                                     |
| Input Fields         | Name              | Type                                    | Description                         |
| <b>Output Fields</b> | ExchangeResponse  | ConversionResponsePO (S_XX_FirstName)   |                                     |
| Temp Fields          |                   |   |                                     |

73. From the properties pane, select **Temp Fields**.

74. To add a new temporary field, click .

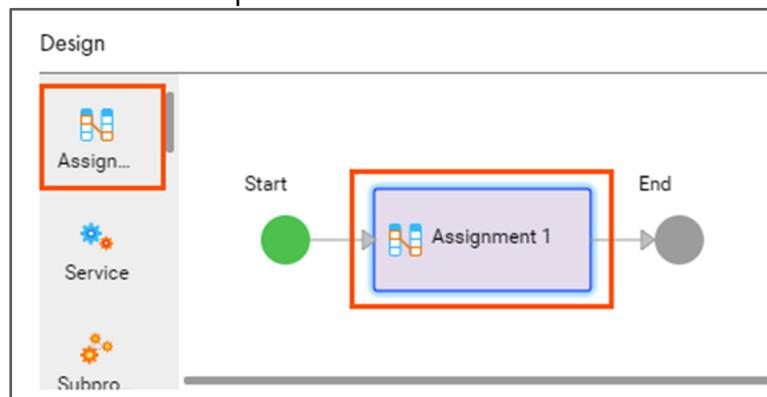
75. Create a temp field, as shown in the table below:

| Name  | Type |
|-------|------|
| Title | Text |



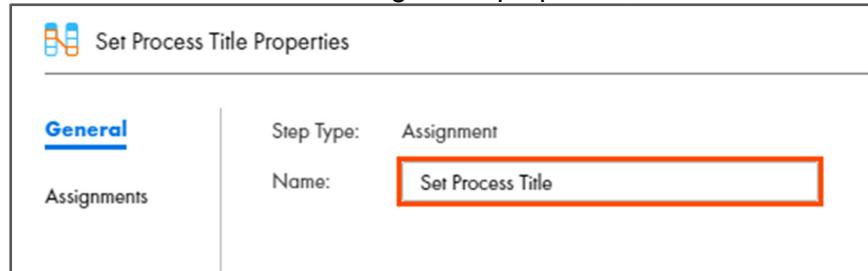
| Temp Fields (1) |      |
|-----------------|------|
| Name            | Type |
| Title           | Text |

76. From the Design palette, drag and drop an **Assignment** step on the link between the Start and End steps.



77. To configure the step, from the process canvas, select the **Assignment** step.

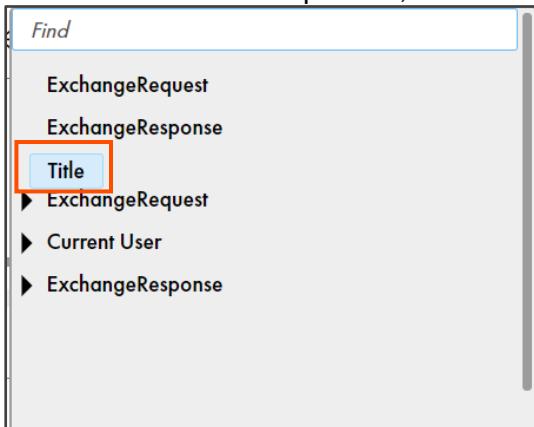
78. In the General section of Assignment properties, enter the Name as **Set Process Title**.



| General     |                         |
|-------------|-------------------------|
| Assignments | Step Type: Assignment   |
|             | Name: Set Process Title |

79. From the properties pane, select **Assignments**.

80. From the Add Field drop-down, select **Title**.



81. From the Assigned Using drop-down, select **Formula**.



82. To open the Formula Editor, click .



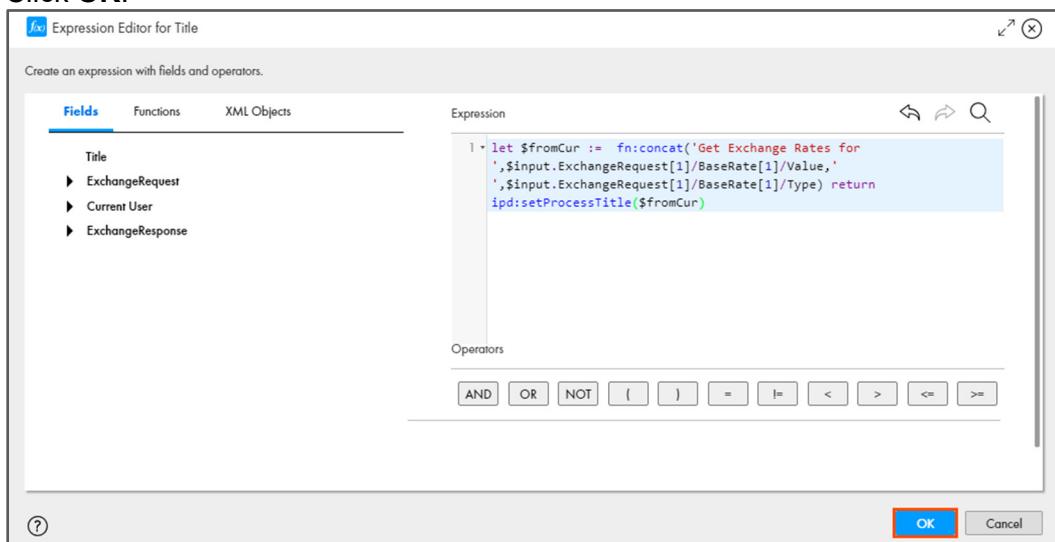
83. In the Expression field, enter the following formula:

```
let $fromCur := fn:concat('Get Exchange Rates for
',$input.ExchangeRequest[1]/BaseRate[1]/Value,
,$input.ExchangeRequest[1]/BaseRate[1]/Type) return
ipd:setProcessTitle($fromCur)
```

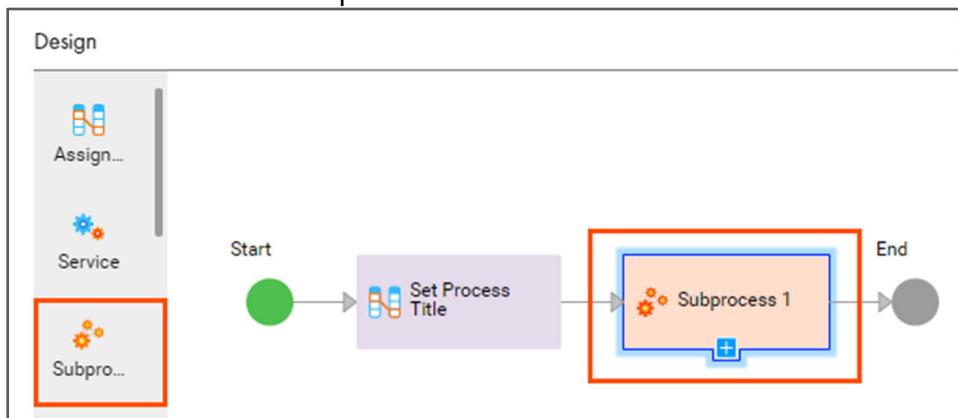
OR

Open the **CAI Lab Prep Files** folder and open the notepad file named **17\_LabGuide\_CAI\_SynchronousWebServiceCall\_10-1**. Copy the command mentioned under **Step D** and paste it in the Expression field.

84. Click **OK**.

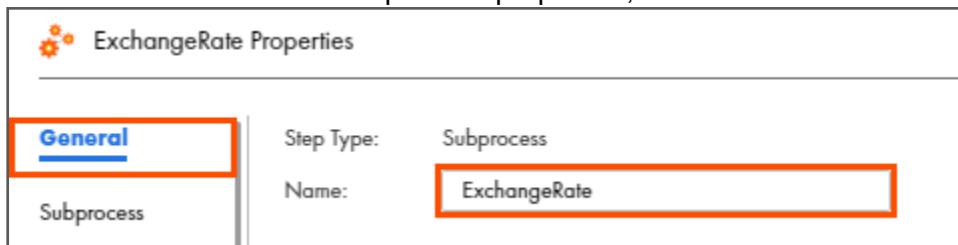


85. From the Design palette, drag and drop a **Subprocess** step on the link between the Set Process Title and End steps.



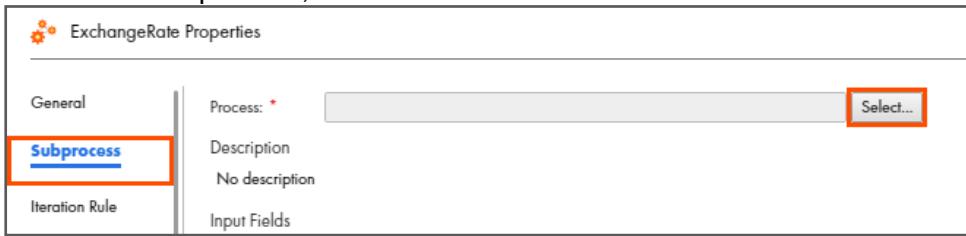
86. To configure the step, from the process canvas, select the **Subprocess** step.

87. In the General section of Subprocess properties, enter the Name as **ExchangeRate**.



88. From the properties pane, select **Subprocess**.

89. To add the subprocess, click **Select**.

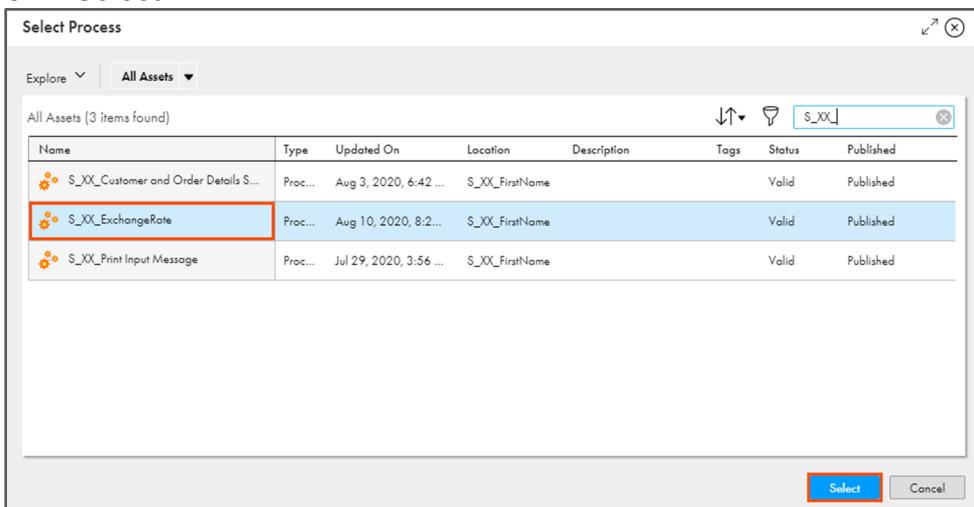


**Note:** A new Select Process window appears.

90. From the list, select the **S\_XX\_ExchangeRate** process.

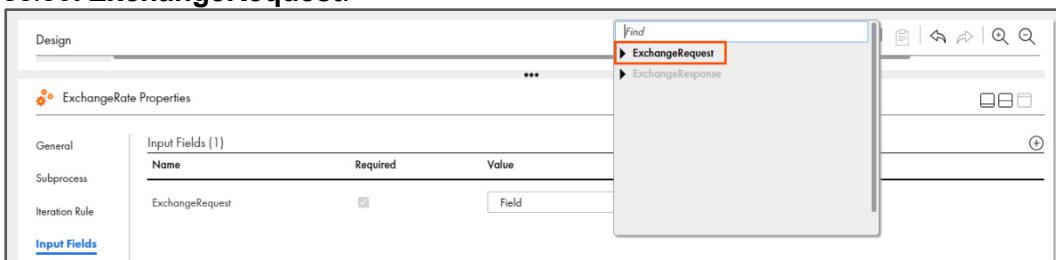
**Recommended:** To quickly locate the process, use the **Search** field.

91. Click **Select**.

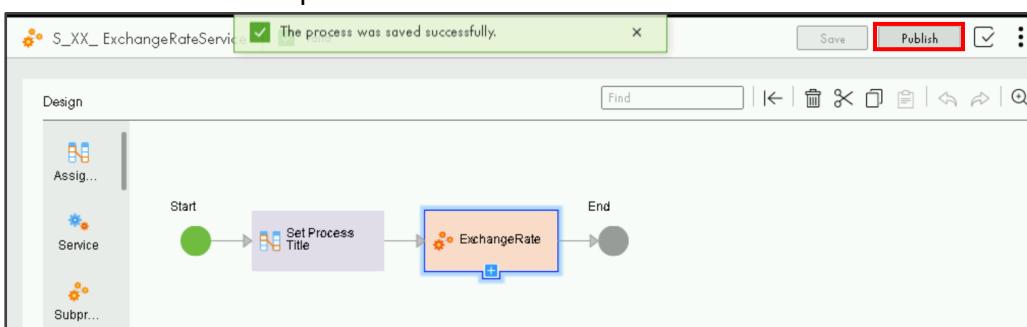


92. From the properties pane, select **Input Fields**.

93. From the ExchangeRequest field, select the **Click To Select Field** drop-down, and select **ExchangeRequest**.



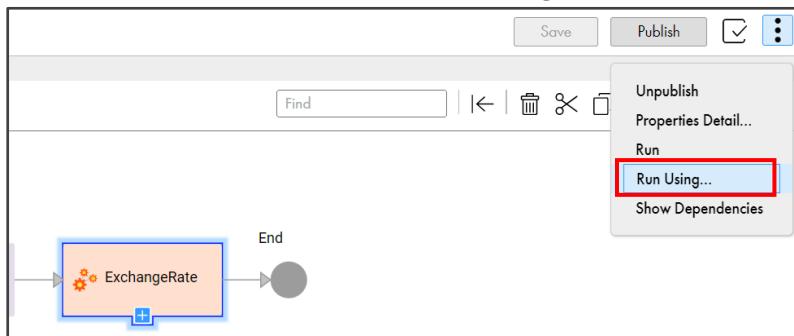
94. Save and Publish the process.



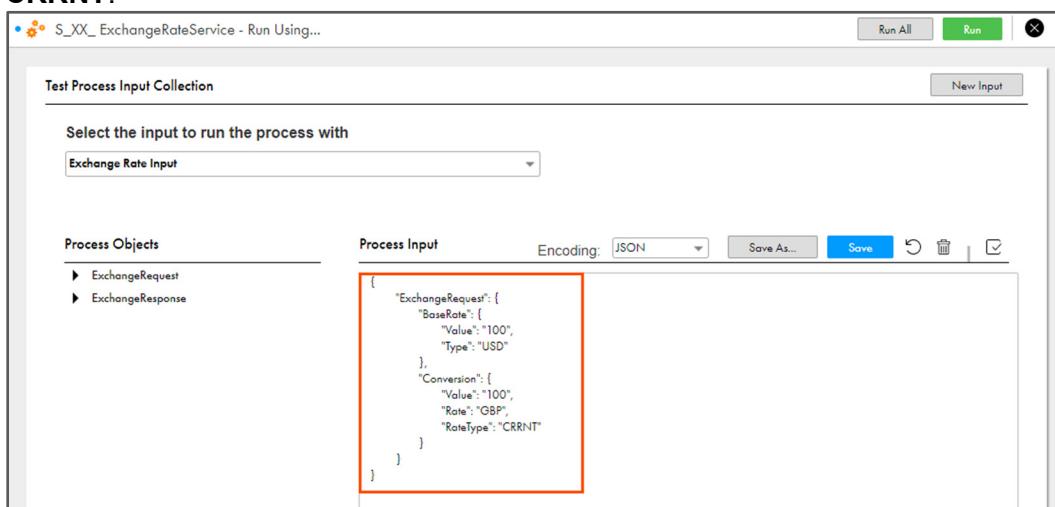
## Execute the Process

You will now see how to run the process in two ways. Firstly, you will run it from the interface.

95. From the Actions menu, select **Run Using...**

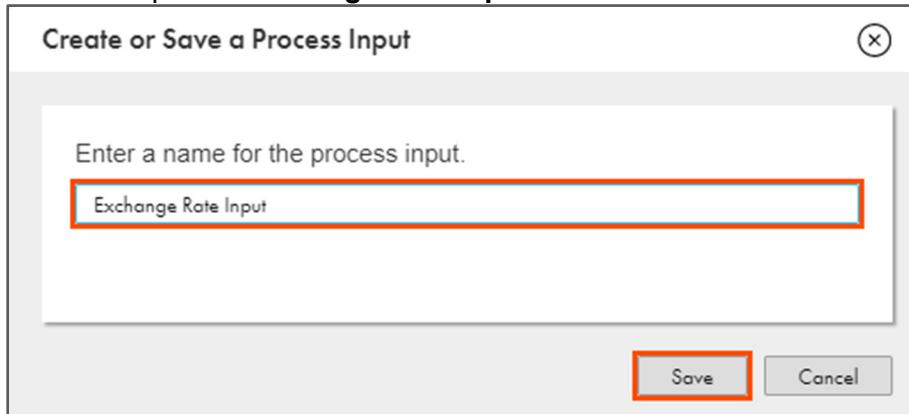


96. In the Process Input section, for the **BaseRate** object, add the Value as **100** and Type as **USD**.
97. For the **Conversion** object, add the Value as **100**, Rate as **GBP**, and the RateType as **CRRNT**.

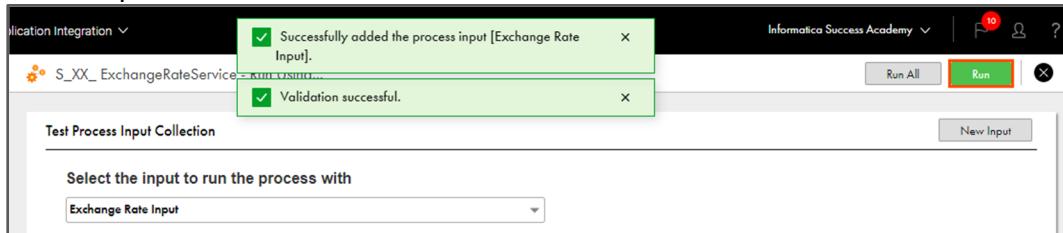


**Note:** This enables you to find the conversion rate of the Great Britain Pound against the US Dollars. You can replace GBP with AUD, CAD, EUR, and so on to obtain the conversion rates for other currencies.

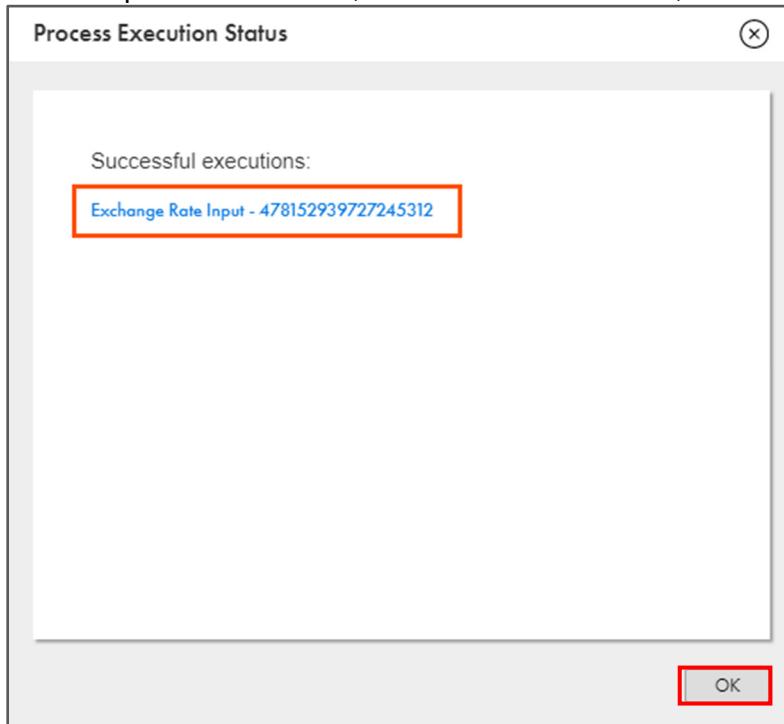
98. Save the input as **Exchange Rate Input**.



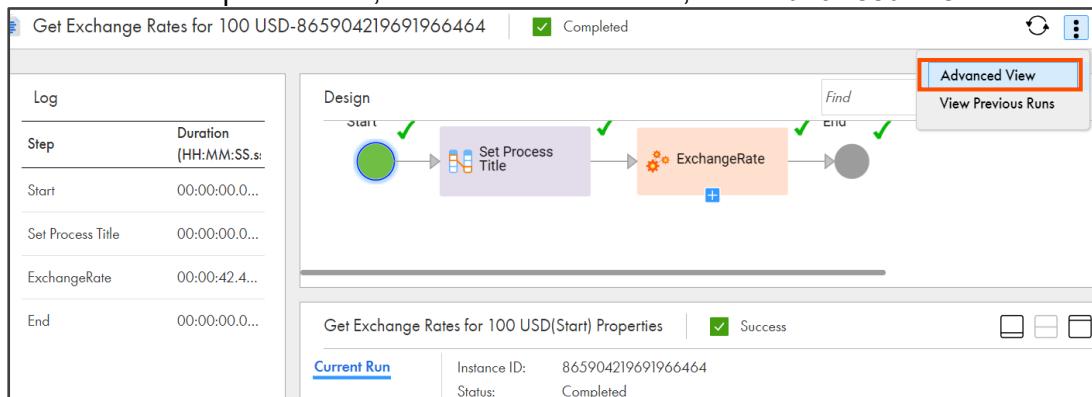
99. Run the process.



100. After the process executes, click the **Execution** link, and click **OK**.



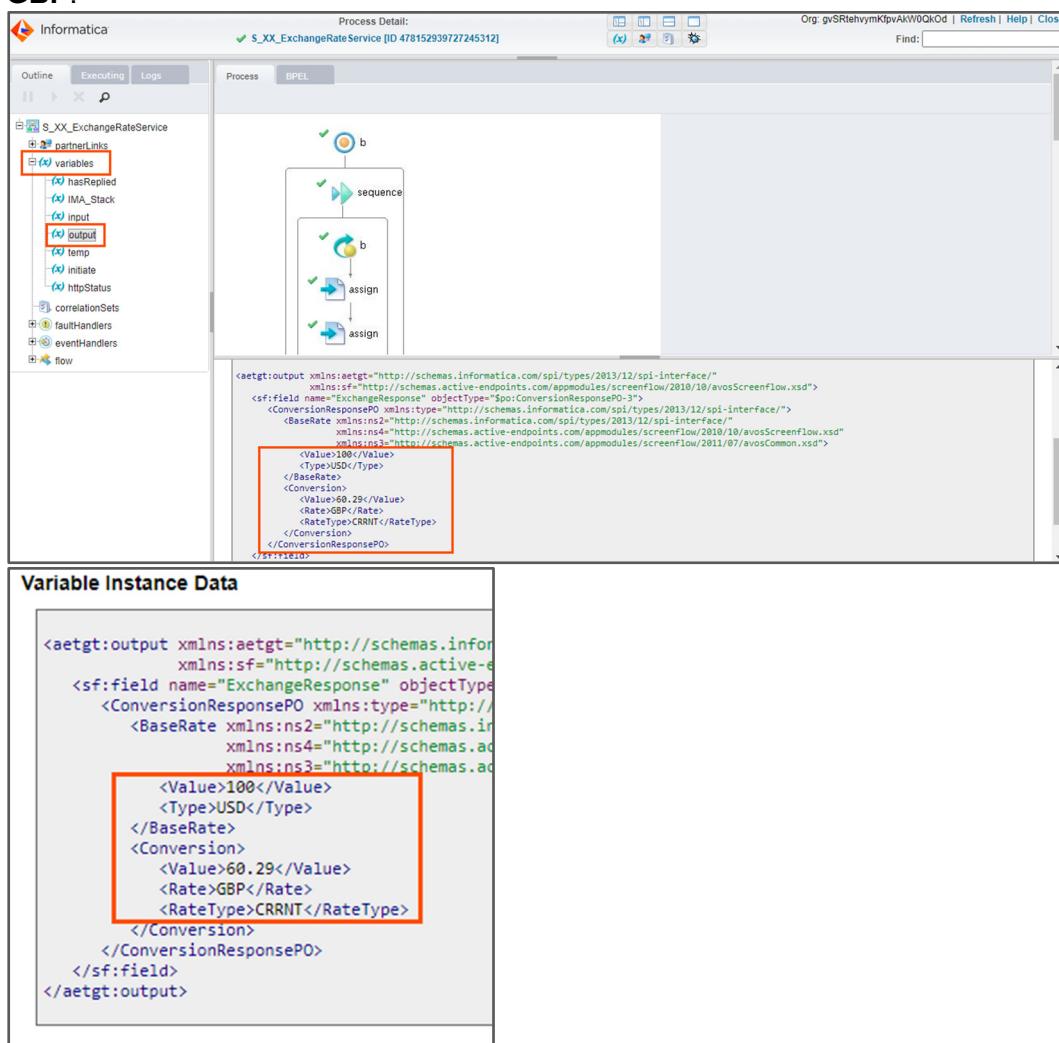
101. To check the output in detail, from the Actions menu, click **Advanced View**.



This directs you to Application Integration Console.

102. From the variables section, select **output**.

103. In the Variable Instance Data section, observe that you get the conversion value for **GBP**.



The screenshot shows the BPEL editor for the process "S\_XX\_ExchangeRateService". The left sidebar shows variables like hasReplied, IMA\_Stack, input, output, temp, initiate, httpStatus, and correlationSets. The "output" variable is selected and highlighted with a red box. The main pane shows the BPEL code for the process, including the conversion logic. The "Variable Instance Data" pane at the bottom shows the XML response with the GBP conversion value highlighted.

```

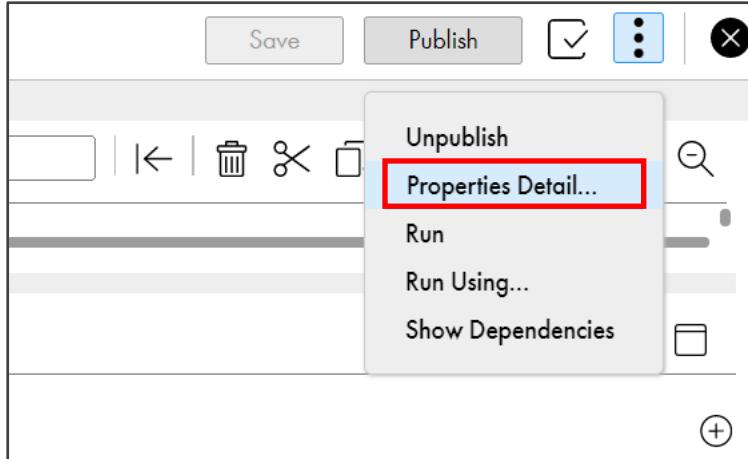
<aetgt:output xmlns:aetgt="http://schemas.informatica.com/spi/types/2013/12/spi-interface/">
  <ns1:sf><http://schemas.active-endpoints.com/appmodules/screenflow/2010/10/avosScreenflow.xsd>
  <sf:field name="ExchangeResponse" objectType="http://schemas.informatica.com/spi/types/2013/12/spi-interface/>
    <ConversionResponsePO objectURI="http://schemas.informatica.com/spi/types/2013/12/spi-interface/>
      <BaseRate xmlns:ns2="http://schemas.informatica.com/spi/types/2013/12/spi-interface/">
        <Value>60.29</Value>
        <Type>GBP</Type>
        <Rate>GBP</Rate>
        <RateType>CRNNT</RateType>
      </BaseRate>
      <Conversion>
        <Value>100</Value>
        <Type>USD</Type>
        <BaseRate>
          <Conversion>
            <Value>60.29</Value>
            <Rate>GBP</Rate>
            <RateType>CRNNT</RateType>
          </Conversion>
        </BaseRate>
      </Conversion>
    </ConversionResponsePO>
  </sf:field>
</aetgt:output>

```

You can also execute the process from a third-party application, such as Postman. You will use the Service URL to execute it.



104. In CAI, navigate back to the S\_XX\_ExchangeRateService process, and click .
105. Select **Properties Detail**.



106. From the Properties Detail page, to copy the Service URL, click **Copy**.

**Properties Detail for S\_XX\_ExchangeRateService**

**Basic**

Unique Name: S\_XX\_ExchangeRateService  
 Location: S\_XX\_FirstName  
 Publication Status:  Published  
 Published On: 2020-08-12 16:41  
 Published By: \*\*\*\*\*  
 Applies To: \* Any \*

**Endpoints**

|                   |   |                                     |
|-------------------|---|-------------------------------------|
| Service URL:      | <a href="https://na1.ai.dmc.informaticacloud.com/active-bpel/public/rt/gvSRtehvymKfpvAkW0QkOd/S_XX_ExchangeRateService">https://na1.ai.dmc.informaticacloud.com/active-bpel/public/rt/gvSRtehvymKfpvAkW0QkOd/S_XX_ExchangeRateService</a>     | <input type="button" value="Copy"/> |
| SOAP Service URL: | <a href="https://na1.ai.dmc.informaticacloud.com/active-bpel/public/soap/gvSRtehvymKfpvAkW0QkOd/S_XX_ExchangeRateService">https://na1.ai.dmc.informaticacloud.com/active-bpel/public/soap/gvSRtehvymKfpvAkW0QkOd/S_XX_ExchangeRateService</a> | <input type="button" value="Copy"/> |

[View Swagger File](#)    [View WSDL File](#)

(?)    Close

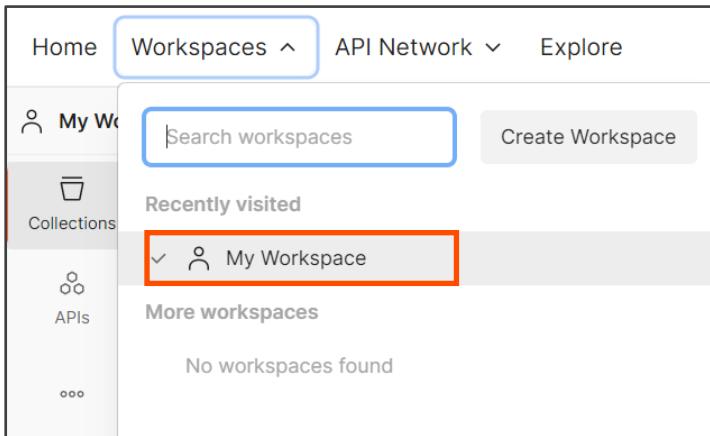
107. Copy the Service URL on a notepad file.

108. Open the web browser and enter the following link to open Postman:

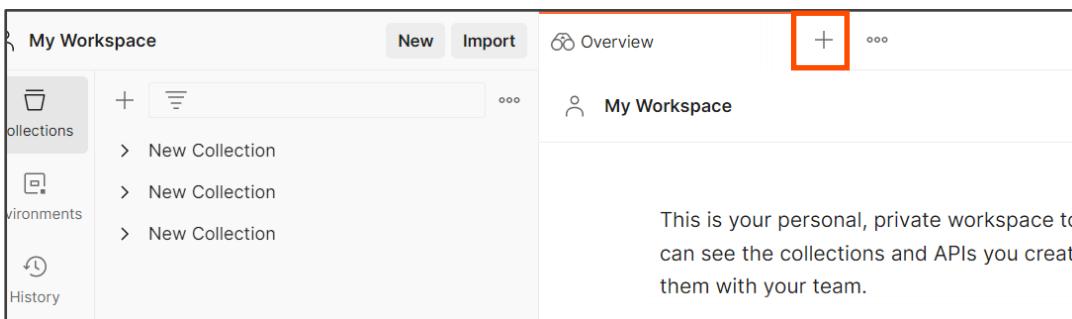
<https://identity.getpostman.com/>

109. Login to Postman using your credentials.

110. From the Workplace drop-down, select **My Workspace**.

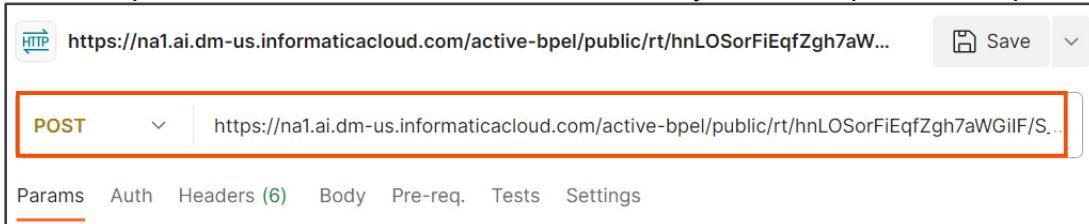


111. Click the plus (+) icon.



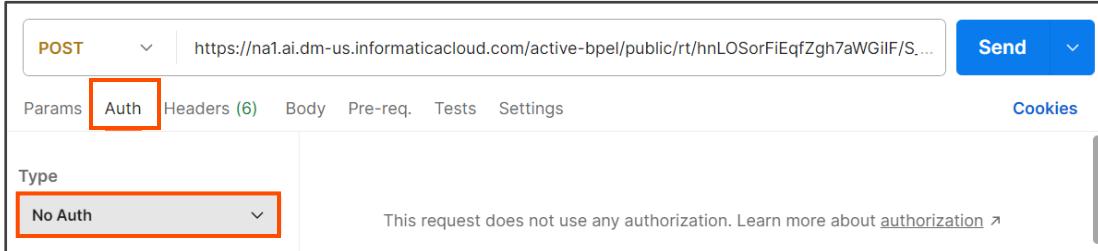
112. In the Workspace section, from the drop-down, select **POST**.

113. In the Request URL field, enter the Service URL that you have copied on notepad file.



114. Select the **Auth** tab.

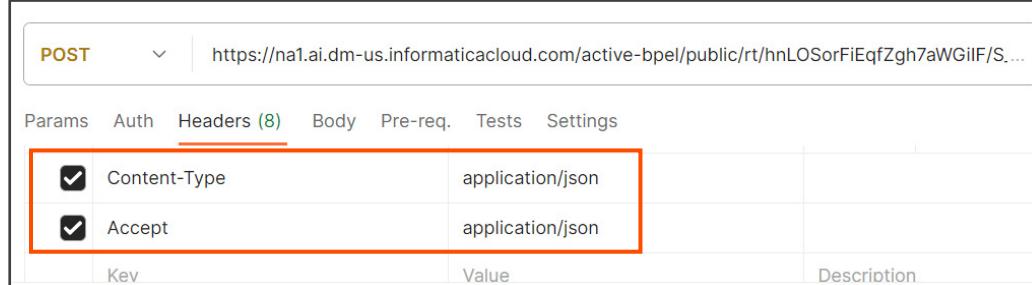
115. From the Type drop-down, select **No Auth**.



116. Select the **Headers** tab.

117. Enter Key and Value as shown in the table below:

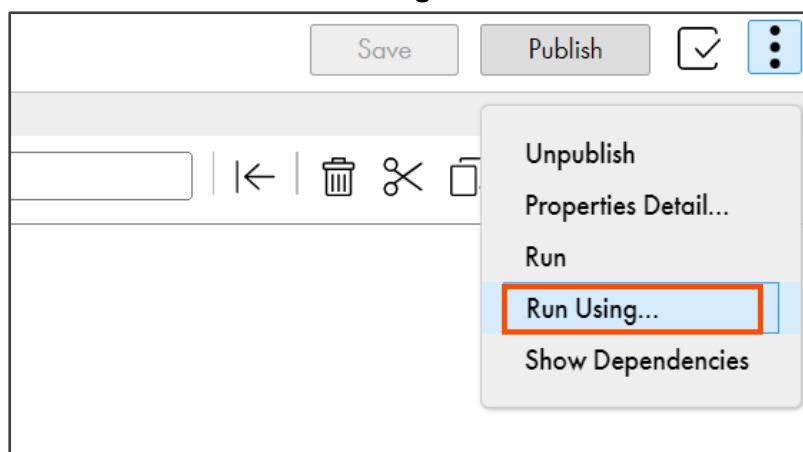
| Key          | Value            |
|--------------|------------------|
| Content-Type | application/json |
| Accept       | application/json |

The screenshot shows the Postman interface with a POST request to a specific URL. The 'Headers' tab is selected, showing two entries: 'Content-Type' set to 'application/json' and 'Accept' set to 'application/json'. Both entries are highlighted with a red box.

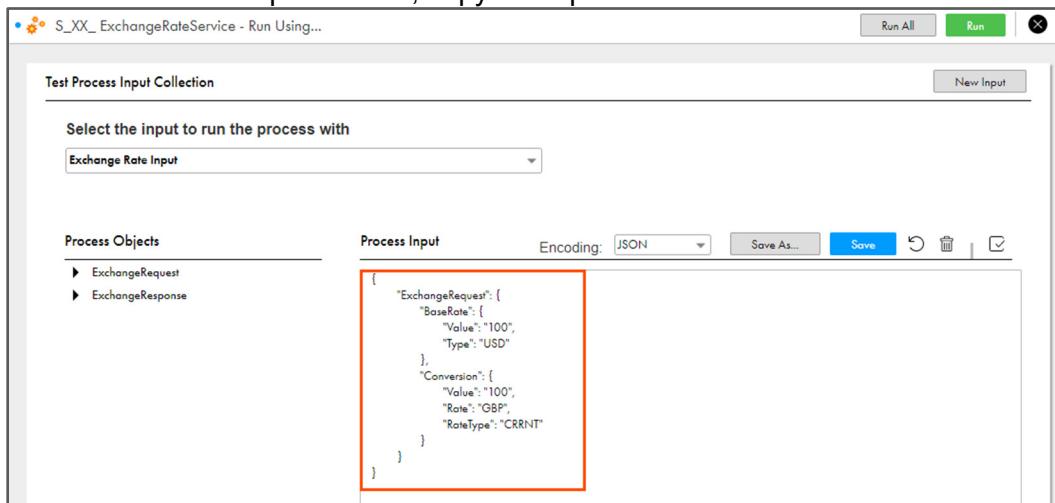
118. Select the **Body** tab and select **raw**.

119. In Cloud Application Integration, in S\_XX\_ ExchangeRateService process, from the **Actions** Menu, select **Run Using....**



The screenshot shows the Cloud Application Integration interface for a process named 'S\_XX\_ ExchangeRateService'. The 'Actions' menu is open, and the 'Run Using...' option is highlighted with a red box.

120. From the Process Input section, copy the input.



The screenshot shows the 'Test Process Input Collection' screen for the 'S\_XX\_ ExchangeRateService' process. The 'Process Input' section displays a JSON object with two main fields: 'ExchangeRequest' and 'Conversion'. The entire JSON structure is highlighted with a red box.

```

{
  "ExchangeRequest": {
    "BaseRate": [
      {"Value": "100", "Type": "USD"}
    ],
    "Conversion": [
      {"Value": "100", "Rate": "GBP", "RateType": "CRRNT"}
    ]
}
  
```

**Note:** While copying, make sure that input values are available.

121. Navigate back to Postman.

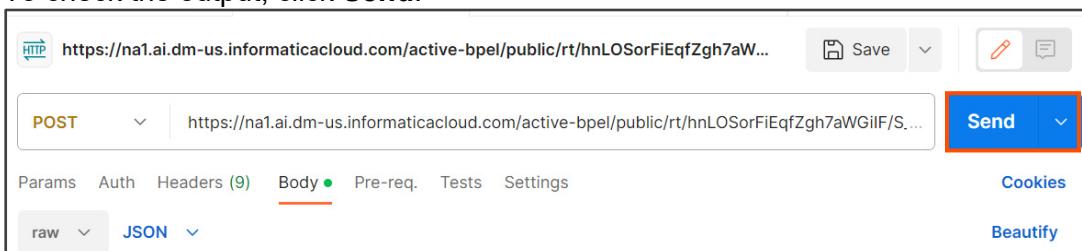
122. In the Body tab, select **raw** and paste the input.



```

1  {
2   "ExchangeRequest": {
3     "BaseRate": {
4       "Value": "100",
5       "Type": "USD"
6     },
7     "Conversion": {
8       "Value": "100",
9       "Rate": "GBP",
10      "RateType": "CRRNT"
11    }
12  }
13 }
```

123. To check the output, click **Send**.



HTTP [https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfIEqfZgh7aWGiF/S...](https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfIEqfZgh7aW...)

POST <https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfIEqfZgh7aWGiF/S...>

Save |  

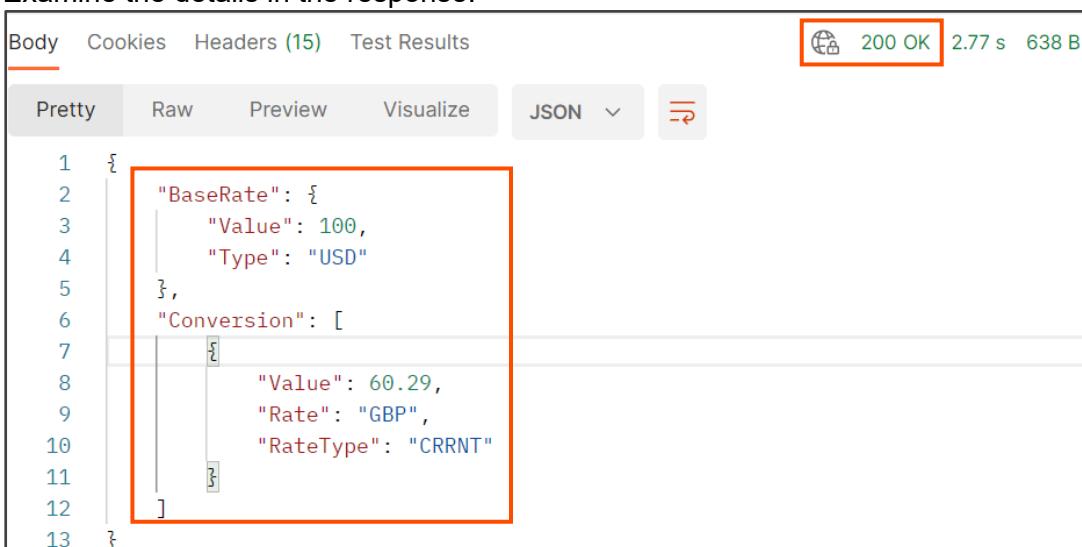
Params Auth Headers (9) Body Pre-req. Tests Settings Cookies Beautify

raw JSON

**Send**

124. Scroll down and verify that the status of the response is **200 OK**.

125. Examine the details in the response.



Body Cookies Headers (15) Test Results **200 OK** 2.77 s 638 B

Pretty Raw Preview Visualize JSON 

```

1  {
2   "BaseRate": {
3     "Value": 100,
4     "Type": "USD"
5   },
6   "Conversion": [
7     {
8       "Value": 60.29,
9       "Rate": "GBP",
10      "RateType": "CRRNT"
11    }
12  }
13 }
```

126. **Close** the Postman application.

In this way, you can add a web service to a process and execute it from the interface or a third-party application.

---

*This concludes the lab.*

# Module 10: Adding Web Service to a Process

## Lab 10-2: Invoke an Asynchronous Web Service Call

### Overview:

In this lab, you will create a process which incorporates the asynchronous web service call and invoke the process.

### Objectives:

- Install and setup a JMS Provider
- Configure a JMS Service in CAI
- Create a process with asynchronous web service call
- Run the process from a third-party application
- View the contents of the queue after message has been written

### Duration:

40 Minutes

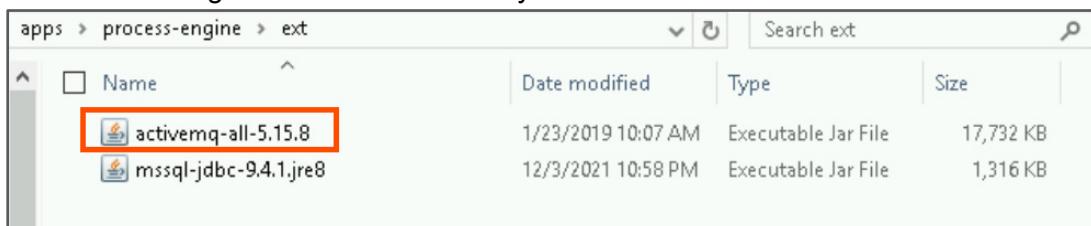
---

## Tasks

**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

### Install and setup a JMS Provider

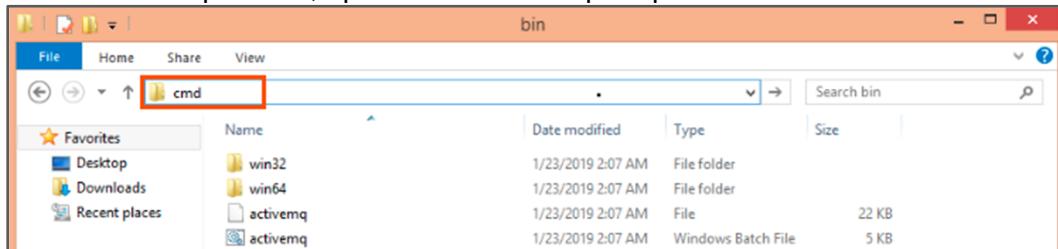
1. On the desktop, from the **CAI Lab Prep Files** folder, open the **apache-activemq-5.15.8** folder.
2. Copy the jar file **activemq-all-5.15.8** and paste it in the **C:\Program Files\Informatica Cloud Secure Agent\apps\process-engine\ext** folder of the Secure Agent installation directory.



3. After you paste the jar file in the Secure Agent Installation directory, you must **Restart** the Secure Agent and wait for it to come up and running.

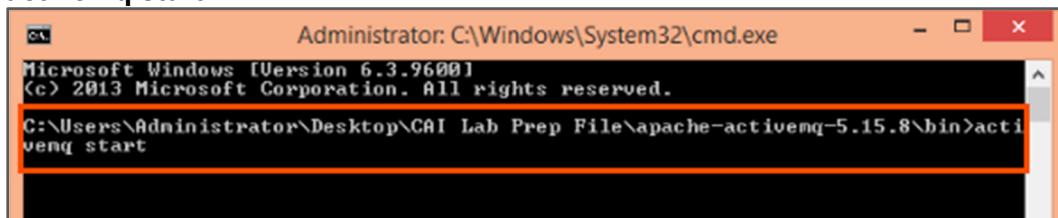
**Note:** To restart Secure Agent, from the Windows taskbar, launch and open the agent window, and click **Restart**. Wait for the agent to start all the services before proceeding with the next step.

4. In the CAI Lab Prep Files folder, navigate to the **apache-activemq-5.15.8\bin** folder and from the folder path bar, open the command prompt as shown in the screenshot below:

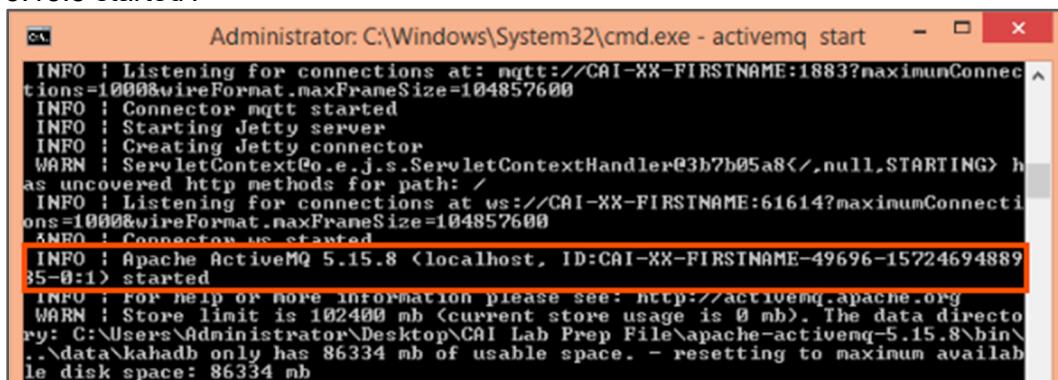


5. To start the activemq service, enter the following command:

**activemq start**



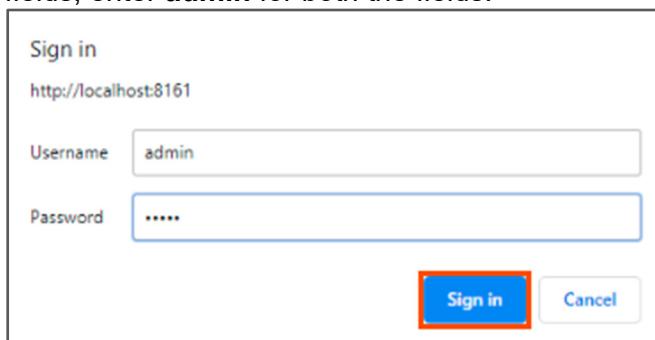
**Note:** Wait for few seconds and scroll up till you find the message 'Apache ActiveMQ 5.15.8 started'.



6. Open a web browser and enter the following URL:

**http://localhost:8161/admin**

**Note:** If the localhost prompts for login credentials, in the Username and Password fields, enter **admin** for both the fields.



|  |       |
|--|-------|
| Sign in  |       |
| <a href="http://localhost:8161">http://localhost:8161</a>                    |       |
| Username   | admin |
| Password   | ..... |
| <input type="button" value="Sign in"/> <input type="button" value="Cancel"/> |       |

7. On the ActiveMQ homepage, click the **Queues** tab.

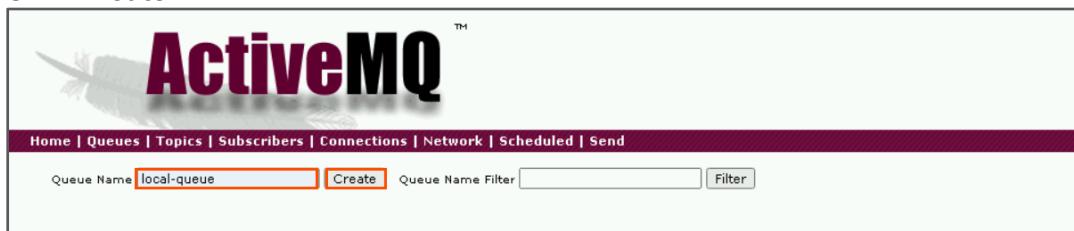


Welcome!

Welcome to the Apache ActiveMQ Console of **localhost** (ID:CAI-XX-FIRSTNAME-51411-1572479565201-0:1)

You can find more information about Apache ActiveMQ on the [Apache ActiveMQ Site](#)

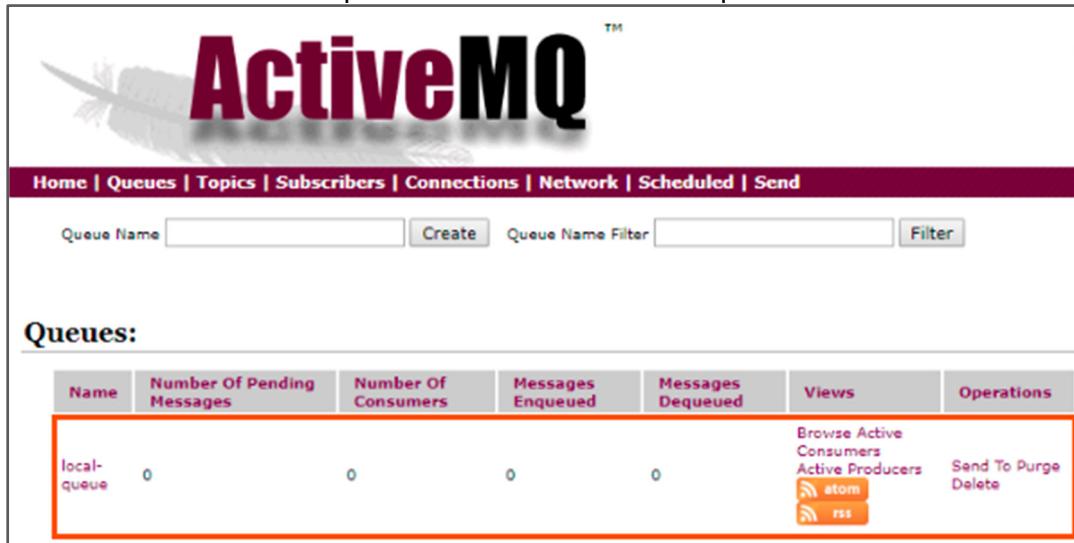
8. In the Queue Name field, enter **local-queue**.  
 9. Click **Create**.



Queue Name: local-queue

Create

**Note:** You can see 'local-queue' in the list of available queues.

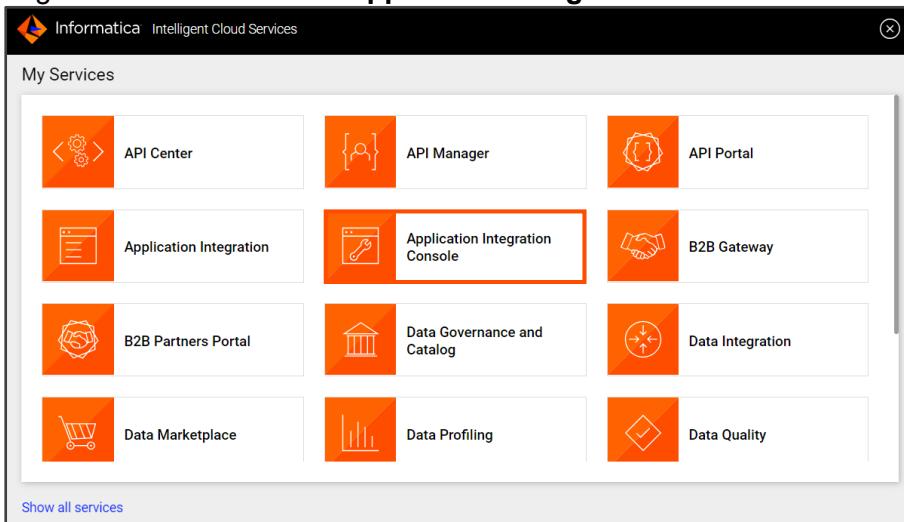


| Name        | Number Of Pending Messages | Number Of Consumers | Messages Enqueued | Messages Dequeued | Views   | Operations  |
|-------------|----------------------------|---------------------|-------------------|-------------------|---|---|
| local-queue | 0                          | 0                   | 0                 | 0                 | <a href="#">Browse Active Consumers</a><br><a href="#">Active Producers</a><br><br> | <a href="#">Send To Purge</a><br><a href="#">Delete</a> |

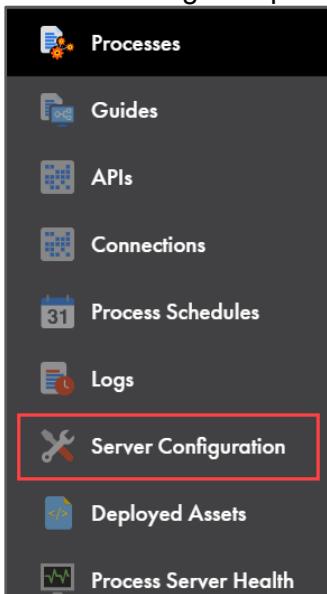
You have successfully created the JMS queue to read messages from or write messages to it.

## Configure JMS Service

10. Log in to IICS and access **Application Integration Console**.



11. From the navigation pane, select **Server Configuration**.



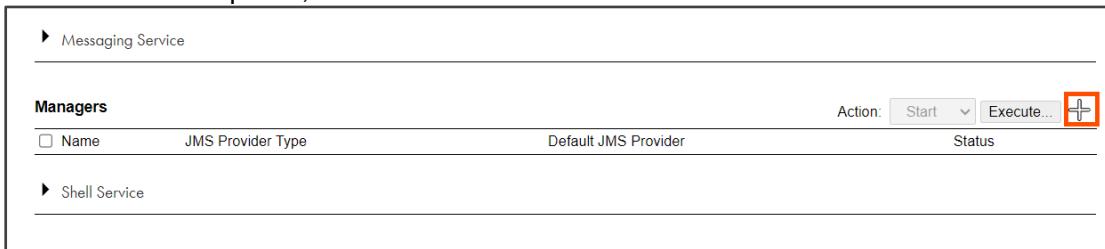
12. To configure the JMS service, from the Server Configuration drop-down, select your secure agent group.

13. Click **System Services**.



14. From the System Services menu, expand **Messaging Service**.

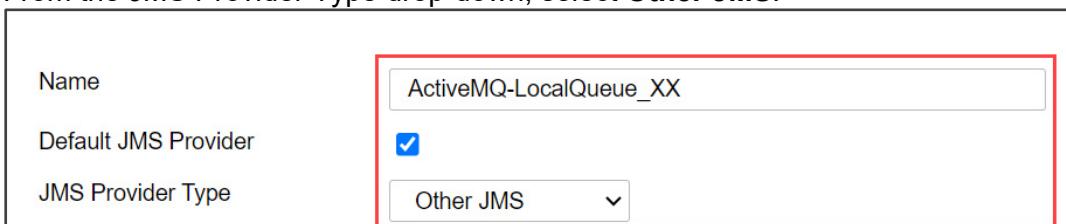
15. To add the local queue, click .



The screenshot shows the Informatica Management console. In the left sidebar, 'Messaging Service' is selected. Under 'Managers', there is a table with columns 'Name', 'JMS Provider Type', and 'Status'. A red box highlights the 'Add' button (a plus sign icon) located at the top right of the table.

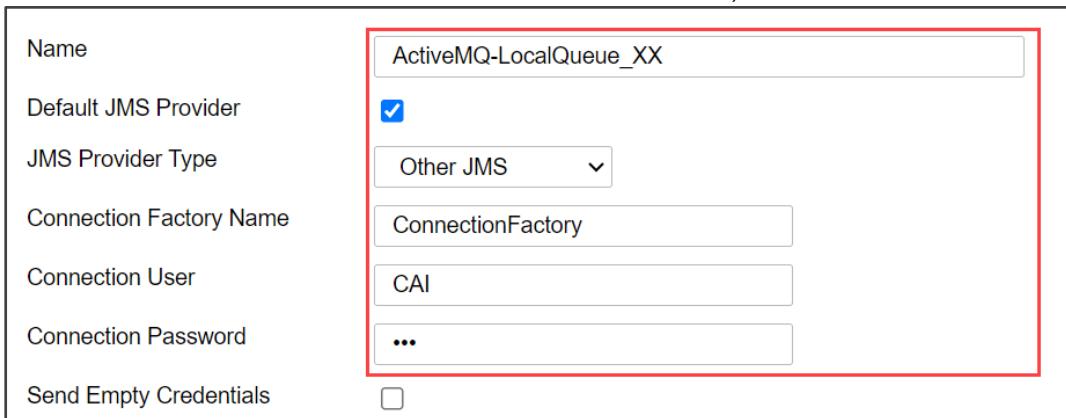
A new Messaging Service window appears.

16. In the Name field, enter **ActiveMQ-LocalQueue\_XX**.
17. Select the **Default JMS Provider** option.
18. From the JMS Provider Type drop-down, select **Other JMS**.



The screenshot shows the 'Add Manager' dialog. It has three fields: 'Name' (ActiveMQ-LocalQueue\_XX), 'Default JMS Provider' (checkbox checked), and 'JMS Provider Type' (dropdown set to 'Other JMS'). A red box highlights the entire form area.

19. In the Connection Factory Name field, enter **ConnectionFactory**.
20. In the Connection User and Connection Password fields, enter **CAI**.



The screenshot shows the 'Add Manager' dialog with more fields. It includes: 'Name' (ActiveMQ-LocalQueue\_XX), 'Default JMS Provider' (checkbox checked), 'JMS Provider Type' (dropdown set to 'Other JMS'), 'Connection Factory Name' (ConnectionFactory), 'Connection User' (CAI), 'Connection Password' (ellipsis), and 'Send Empty Credentials' (checkbox). A red box highlights the entire form area.

## Configure ExchangeRateService Process Properties

To enable the secure agent to receive JMS messages, you must set the **Initial Context Properties**.

In the Initial Context Properties, the following four properties are prepopulated.

**java.naming.provider.url:** Specifies the URL that the application client code uses to look up objects on the server.

**java.naming.factory.initial:** Specifies which initial context factory is used to create a new initial context object.

**java.naming.security.principal:** Specifies the user name that is required to authenticate the user that connects to the JMS provider.

**java.naming.security.credentials:** Specifies the password that is required to authenticate the user that connects to the JMS provider.

21. Enter the values for the fields as shown in the table below:

| Name                             | Value  |
|----------------------------------|--|
| java.naming.provider.url         | tcp://localhost:61616                                  |
| java.naming.factory.initial      | org.apache.activemq.jndi.ActiveMQInitialContextFactory |
| java.naming.security.principal   | Admin  |
| java.naming.security.credentials | Admin  |

22. To add another field, in Initial Context Properties, click .

23. In the Name field, enter **queue.local-queue**.

24. In the Value field, enter **local-queue**.

| Initial Context Properties       |  |
|----------------------------------|--|
| Name                             | Value  |
| java.naming.provider.url         | tcp://localhost:61616                                  |
| java.naming.factory.initial      | org.apache.activemq.jndi.ActiveMQInitialContextFactory |
| java.naming.security.principal   | Admin  |
| java.naming.security.credentials | Admin  |
| queue.local-queue                | local-queue  |

**Note:** The property name must be in the format: **queue.<name of queue>**, where name of the queue is the queue name that you create in ActiveMQ client, that is local-queue for this lab.

## Configure the Queues & Listeners of the JMS Service

25. To add a queue in the Queues & Listeners section, click .

26. In the Name and JNDI Location fields, enter **local-queue**.

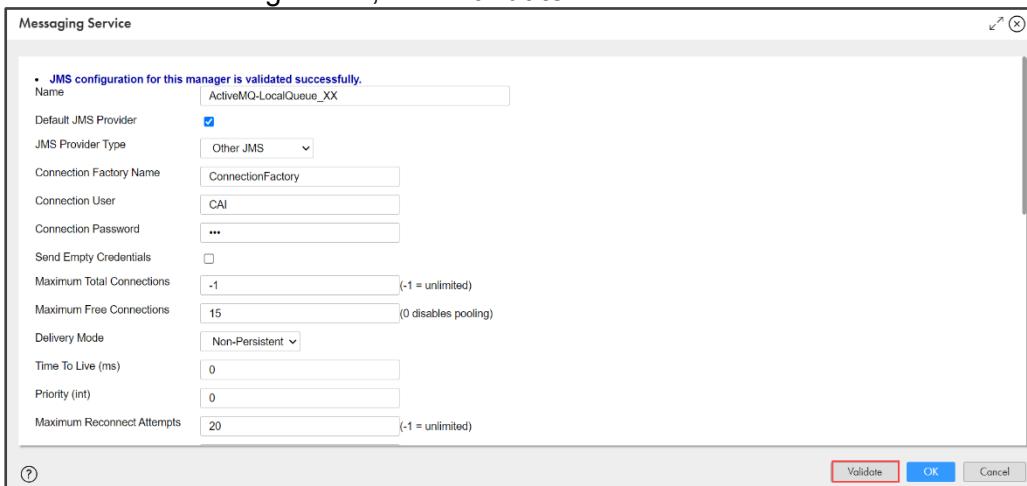
27. If the Listener Count is 1, keep it as is. Else, in the Listener Count field, enter **1**.

| Queues & Listeners |  |
|--------------------|--|
| Name               |  |
| local-queue        | local-queue  |
| JNDI Location      | com.activevos.rt.socrates.web.mom.jms.ARealTimeJmsListener |
| Listener Class     |  |
| Listener Count     | 1  |
| Selector           |  |
| Default Service    |  |
| Run-As Identity    |  |
| XA Transaction     | <input type="checkbox"/>                                   |
| Rollback On Error  | <input checked="" type="checkbox"/>                        |

**Note:** Notice here the **listener count** is 1. Number of connections to keep open. When the server starts, the JMS manager creates instances of this class that serve as asynchronous consumers on the destination. Each asynchronous consumer has its own connection to the JMS server. The number of consumers and connections created is controlled by the listener count.

28. Retain default values in the other fields.

29. To validate the configuration, click **Validate**.



JMS configuration for this manager is validated successfully.

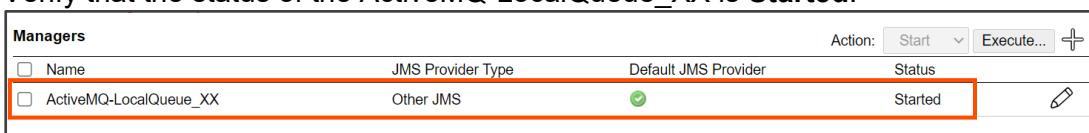
|                            |                          |                      |
|----------------------------|--------------------------|----------------------|
| Name                       | ActiveMQ-LocalQueue_XX   |                      |
| Default JMS Provider       | Other JMS                |                      |
| JMS Provider Type          | Other JMS                |                      |
| Connection Factory Name    | ConnectionFactory        |                      |
| Connection User            | CAI                      |                      |
| Connection Password        | ...                      |                      |
| Send Empty Credentials     | <input type="checkbox"/> |                      |
| Maximum Total Connections  | -1                       | (-1 = unlimited)     |
| Maximum Free Connections   | 15                       | (0 disables pooling) |
| Delivery Mode              | Non-Persistent           |                      |
| Time To Live (ms)          | 0                        |                      |
| Priority (int)             | 0                        |                      |
| Maximum Reconnect Attempts | 20                       | (-1 = unlimited)     |

Validate OK Cancel

A message, '**JMS Configuration for this manager is validated successfully.**' appears on top of the page.

30. Click **OK**.

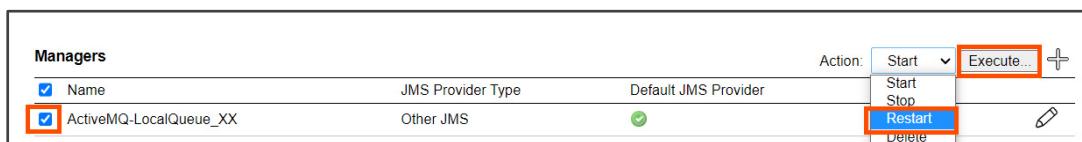
31. Verify that the status of the ActiveMQ-LocalQueue\_XX is **Started**.



| Managers   |                   |   |  |
|--|-------------------|---|--|
|  | JMS Provider Type | Action:   |  |
| <input type="checkbox"/> Name                              |                   |   | <input type="button" value="Start"/> <input type="button" value="Execute..."/> <input type="button" value="Delete"/> |
| <input checked="" type="checkbox"/> ActiveMQ-LocalQueue_XX | Other JMS         | <input checked="" type="button" value="Start"/> <input checked="" type="button" value="Stop"/> <input checked="" type="button" value="Execute..."/> <input type="button" value="Delete"/> | <input checked="" type="button" value="Restart"/>  |

**Note:**

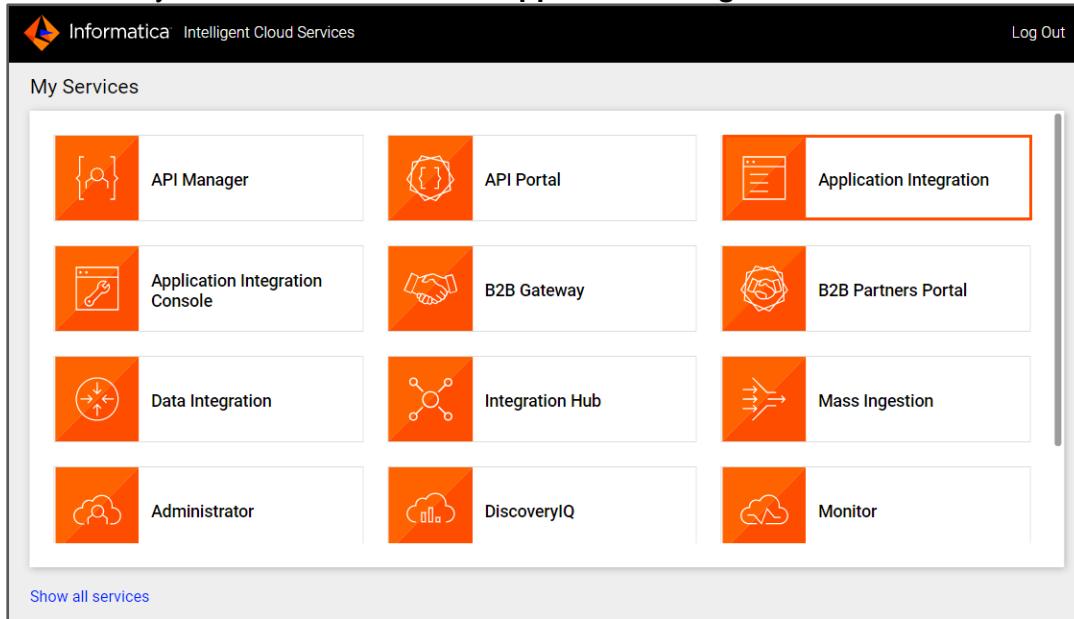
- If the status of the ActiveMQ-LocalQueue\_XX is **Stopped**, check if the Apache ActiveMQ 5.15.8 is started. If not, then navigate to the **apache-activemq-5.15.8/bin** folder. From the folder path bar, open the command prompt and re-execute the **activemq start** command.
- Then, select the ActiveMQ manager and from the Action drop down, choose **Restart** and click **Execute**.



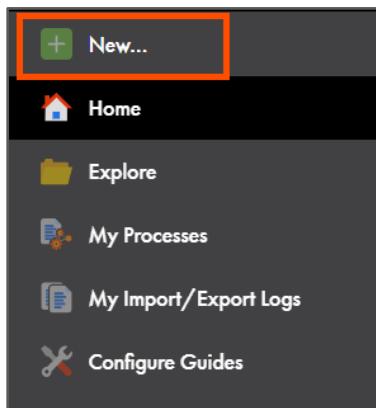
| Managers   |                   |   |   |
|--|-------------------|---|---|
|  | JMS Provider Type | Action:   |   |
| <input checked="" type="checkbox"/> Name                   |                   | <input type="button" value="Start"/> <input type="button" value="Stop"/> <input type="button" value="Execute..."/> <input type="button" value="Delete"/>                                  |   |
| <input checked="" type="checkbox"/> ActiveMQ-LocalQueue_XX | Other JMS         | <input checked="" type="button" value="Start"/> <input checked="" type="button" value="Stop"/> <input checked="" type="button" value="Execute..."/> <input type="button" value="Delete"/> | <input checked="" type="button" value="Restart"/> |

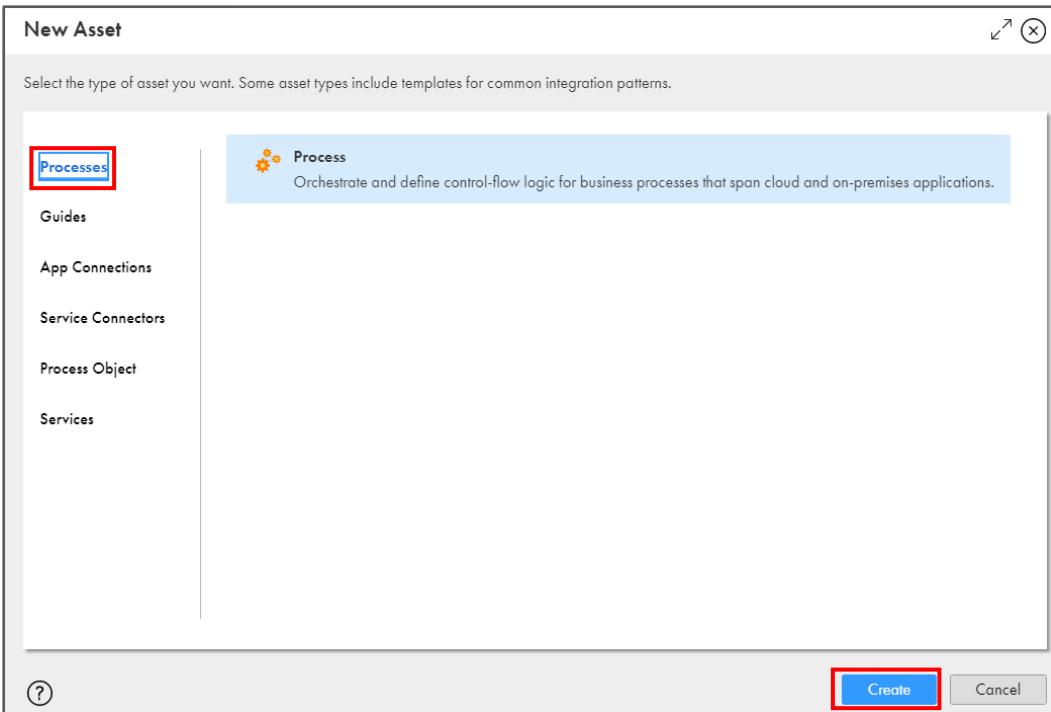
## Create a process to publish a message to the queue

32. To switch between the available services, from the toolbar, select the drop-down next to Application Integration Console.
33. From the My Services window, select **Application Integration**.



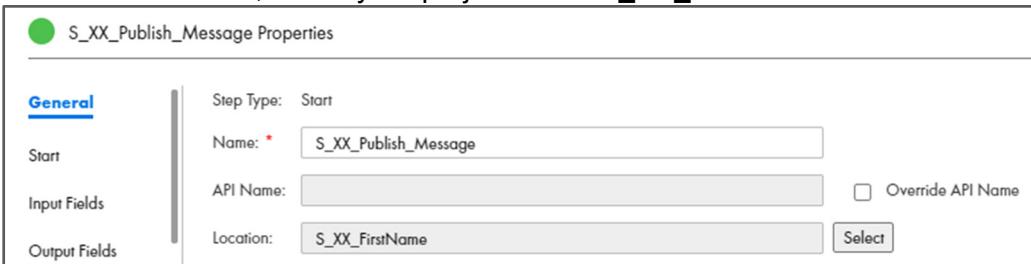
34. Create a new process.





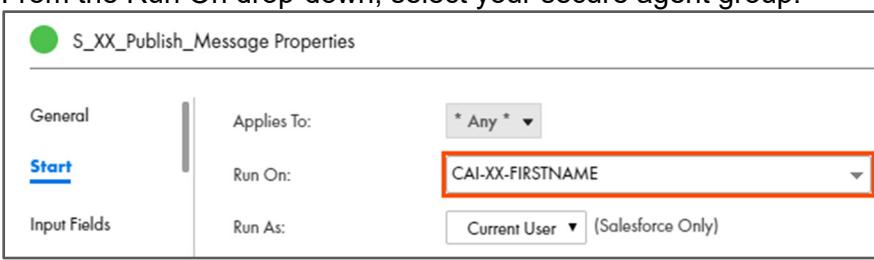
A new Process window appears.

35. From the process canvas, select the **Start** step.
36. In the General tab, enter the name as **S\_XX\_Publish\_Message**.
37. Skip the API Name option.
38. In the Location field, select your project folder **S\_XX\_FirstName**.



| S_XX_Publish_Message Properties |  |
|---------------------------------|--|
| <b>General</b>                  | Step Type: Start<br>Name: * <input type="text" value="S_XX_Publish_Message"/><br>API Name: <input type="text"/><br><input type="checkbox"/> Override API Name<br>Location: <input type="text" value="S_XX_FirstName"/> <input type="button" value="Select"/> |
| Start                           |  |
| Input Fields                    |  |
| Output Fields                   |  |

39. From the properties pane, select **Start**.
40. From the Applies To drop-down, retain **Any**.
41. From the Run On drop-down, select your secure agent group.



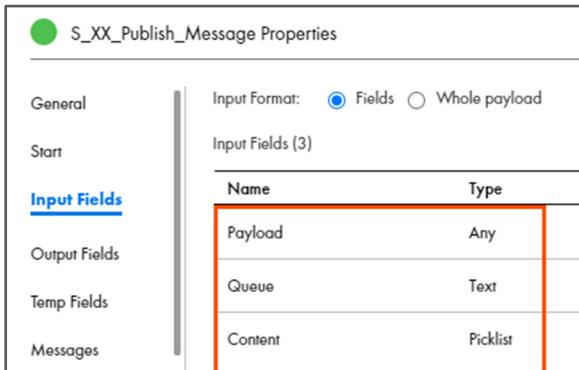
| S_XX_Publish_Message Properties |   |
|---------------------------------|---|
| General                         | Applies To: * Any *   |
| <b>Start</b>                    | Run On: <input type="text" value="CAI-XX-FIRSTNAME"/>               |
| Input Fields                    | Run As: <input type="text" value="Current User (Salesforce Only)"/> |

42. From the properties pane, select **Input Fields**.
43. From the Input Format field, retain **Fields**.

44. To add the Input Fields, click .

45. Add the Input Fields listed in the following table:

| Name    | Type                                 |
|---------|--------------------------------------|
| Payload | More Types > Custom Types > Any      |
| Queue   | Text                                 |
| Content | More Types > Simple Types > Picklist |



S\_XX\_Publish\_Message Properties

General      Input Format:  Fields  Whole payload

Start

**Input Fields**

Output Fields

Temp Fields

Messages

| Name    | Type     |
|---------|----------|
| Payload | Any      |
| Queue   | Text     |
| Content | Picklist |

46. To edit the Content field, click .



S\_XX\_Publish\_Message Properties

General      Input Format:  Fields  Whole payload

Start

**Input Fields**

Output Fields

Temp Fields

Messages

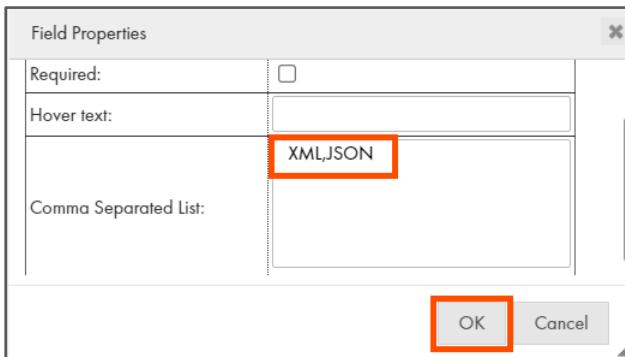
| Name    | Type     | Description | Required   |
|---------|----------|-------------|--|
| Payload | Any      |             | <input type="checkbox"/>   |
| Queue   | Text     |             | <input type="checkbox"/>   |
| Content | Picklist |             | <input type="checkbox"/>   |

Field Properties

A new Field Properties window appears.

47. In the Comma Separated List field, enter **XML, JSON**.

48. Click **OK**.



Field Properties

Required:

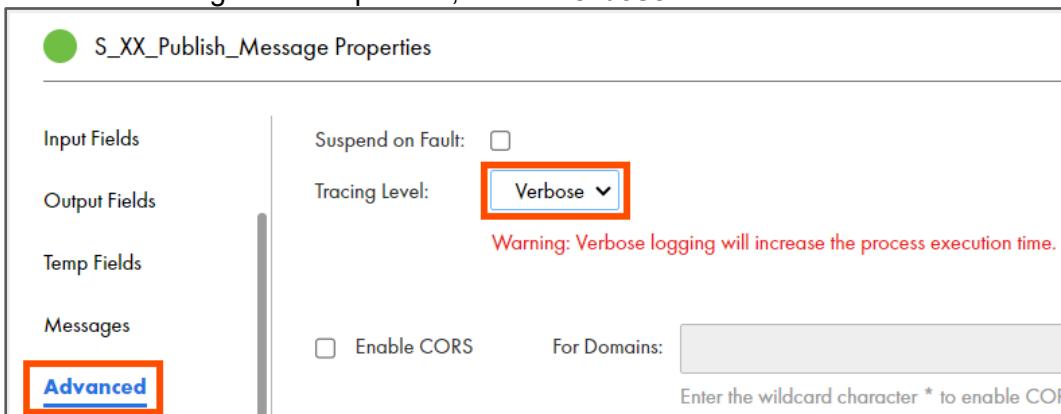
Hover text:

Comma Separated List: **XML,JSON**

OK Cancel

49. From the properties pane, select **Advanced**.

50. From the Tracing Level drop-down, select **Verbose**.



S\_XX\_Publish\_Message Properties

Input Fields      Suspend on Fault:

Output Fields      Tracing Level: **Verbose**

Temp Fields

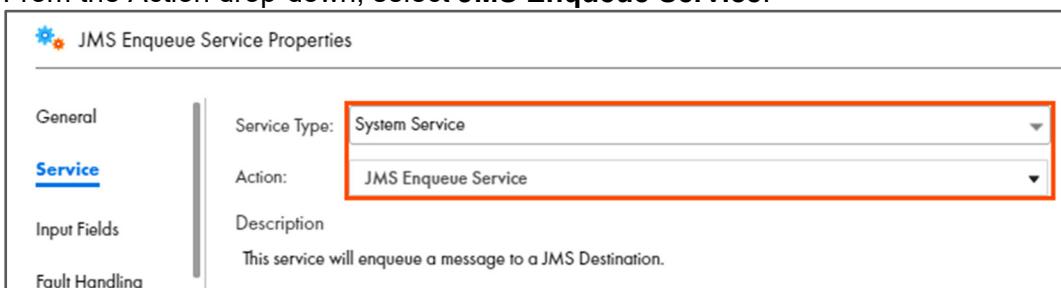
Messages

**Advanced**

Warning: Verbose logging will increase the process execution time.

Enable CORS      For Domains:   
Enter the wildcard character \* to enable CORS

51. From the Design palette, drag and drop a **Service** step on the link between Start and End steps.
52. Retain the name in the General tab and from the properties pane, select **Service**.
53. From Service Type drop-down, select **System Service**.
54. From the Action drop-down, select **JMS Enqueue Service**.



JMS Enqueue Service Properties

General      Service Type: **System Service**

**Service**      Action: **JMS Enqueue Service**

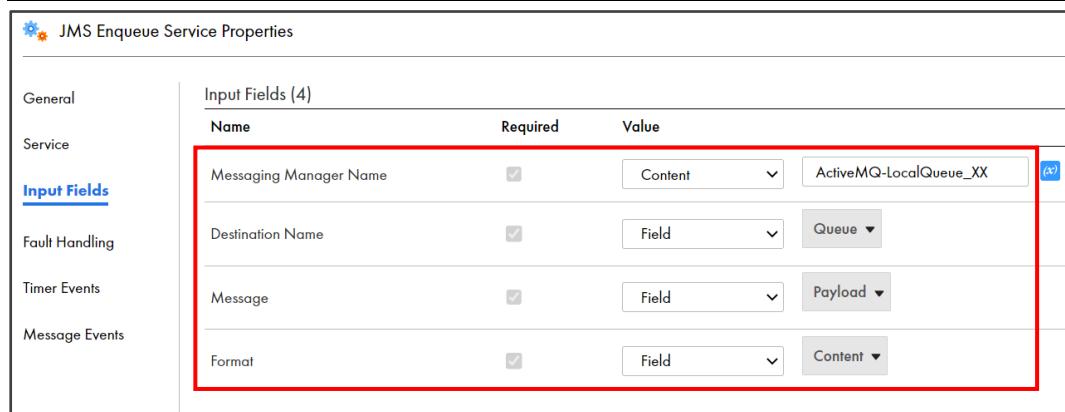
Input Fields

Description  
This service will enqueue a message to a JMS Destination.

Fault Handling

55. From the properties pane, select **Input Fields**.
56. Configure the input fields as shown in the table below:

| Name                   | Value   |                        |
|------------------------|---------|------------------------|
| Messaging Manager Name | Content | ActiveMQ-LocalQueue_XX |
| Destination Name       | Field   | Queue                  |
| Message                | Field   | Payload                |
| Format                 | Field   | Content                |



JMS Enqueue Service Properties

General      Service

**Input Fields**      Input Fields (4)

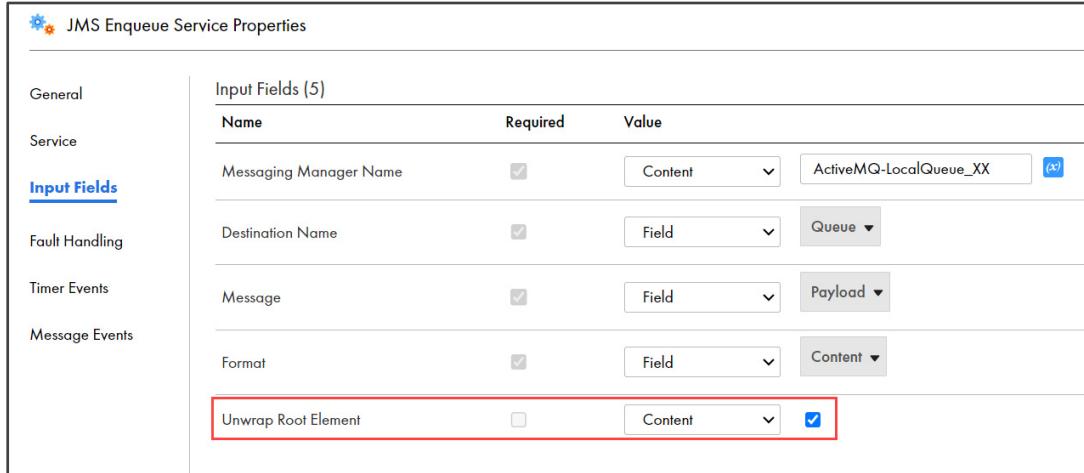
| Name                   | Required                            | Value   |
|------------------------|-------------------------------------|---|
| Messaging Manager Name | <input checked="" type="checkbox"/> | <input type="button" value="Content"/> ActiveMQ-LocalQueue_XX |
| Destination Name       | <input checked="" type="checkbox"/> | <input type="button" value="Field"/> Queue                    |
| Message                | <input checked="" type="checkbox"/> | <input type="button" value="Field"/> Payload                  |
| Format                 | <input checked="" type="checkbox"/> | <input type="button" value="Field"/> Content                  |

Fault Handling

Timer Events

Message Events

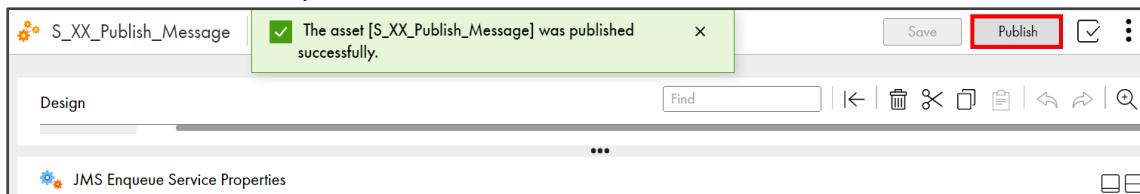
57. To add another input field, click plus (+) icon.  
 58. From the Add Field drop-down, select **Unwrap Root Element**.  
 59. From Value drop-down, retain **Content**, and select the checkbox next to it.



The screenshot shows the 'Input Fields (5)' section of the 'JMS Enqueue Service Properties' dialog. It lists five fields with their names, required status, and values:

| Name                   | Required                            | Value  |
|------------------------|-------------------------------------|--|
| Messaging Manager Name | <input checked="" type="checkbox"/> | Content<br>ActiveMQ-LocalQueue_XX              |
| Destination Name       | <input checked="" type="checkbox"/> | Field<br>Queue                                 |
| Message                | <input checked="" type="checkbox"/> | Field<br>Payload                               |
| Format                 | <input checked="" type="checkbox"/> | Field<br>Content                               |
| Unwrap Root Element    | <input type="checkbox"/>            | Content<br><input checked="" type="checkbox"/> |

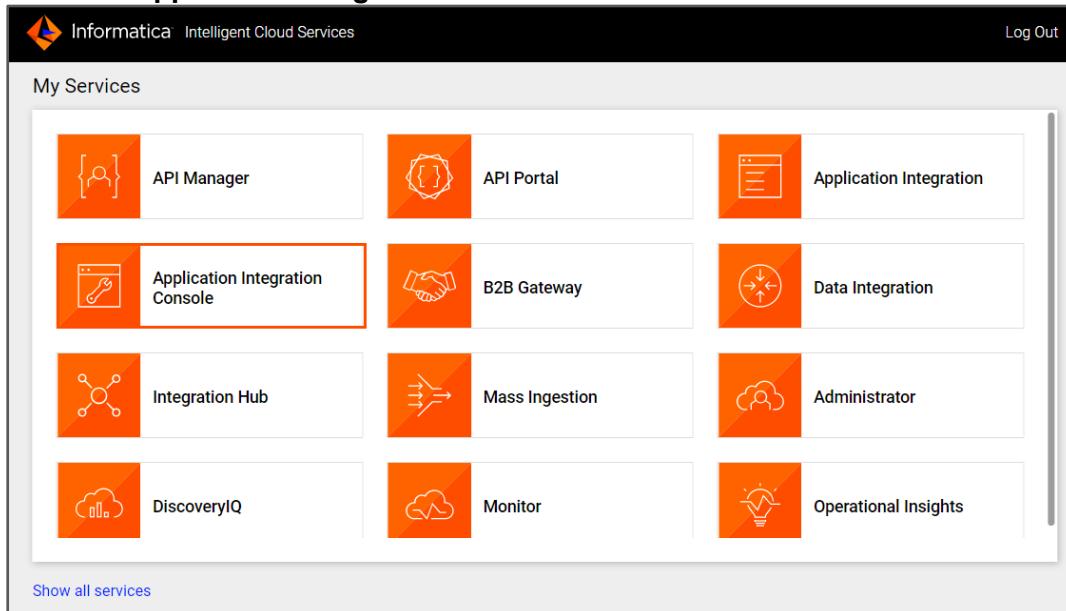
60. **Save and Publish** the process.



The screenshot shows the process editor interface. A green success message box displays: "The asset [S\_XX\_Publish\_Message] was published successfully." The 'Publish' button is highlighted with a red box.

## Create a process to publish a message to the queue

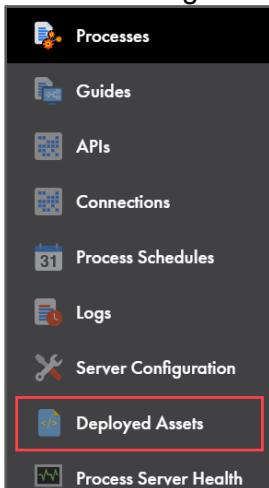
61. Switch to **Application Integration Console**.



The screenshot shows the 'My Services' dashboard of the Application Integration Console. It displays nine service tiles:

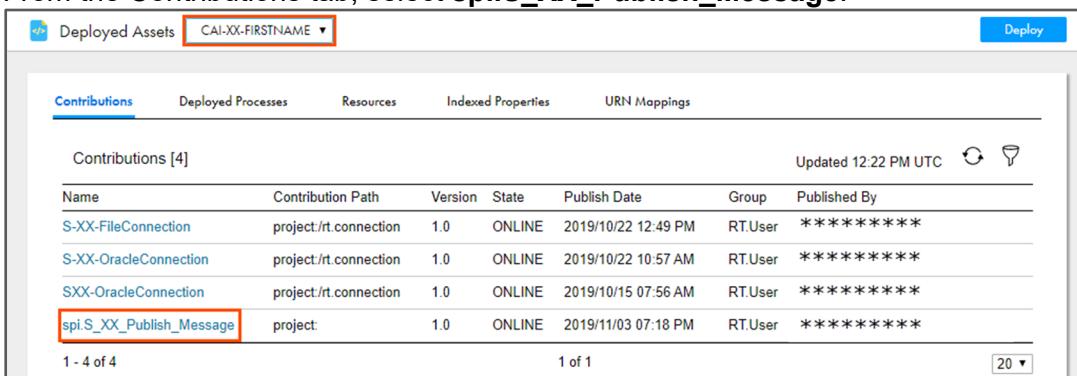
- API Manager
- Application Integration
- Application Integration Console (highlighted with a red box)
- B2B Gateway
- Data Integration
- Integration Hub
- Mass Ingestion
- Administrator
- DiscoveryIQ
- Monitor
- Operational Insights

62. From the navigation pane, select **Deployed Assets**.



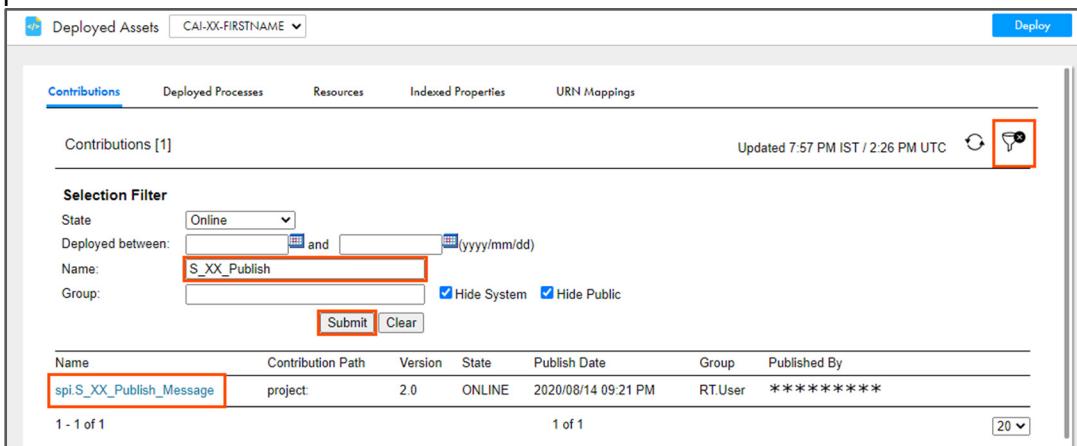
63. From the Deployed Assets drop-down, select your Secure Agent Group.

64. From the Contributions tab, select **spi.S\_XX\_Publish\_Message**.



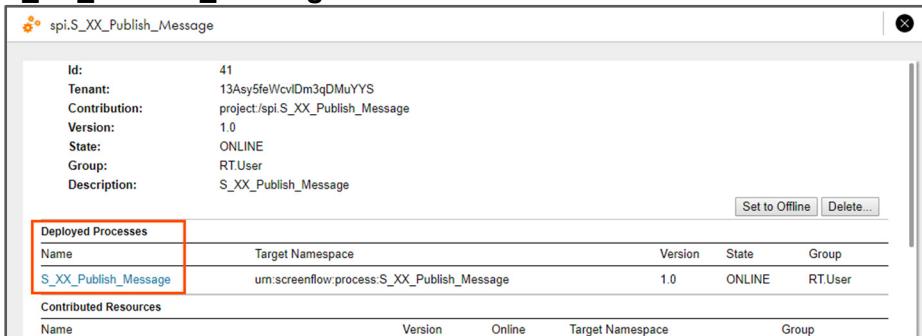
| Name                            | Contribution Path      | Version | State  | Publish Date        | Group   | Published By |
|---------------------------------|------------------------|---------|--------|---------------------|---------|--------------|
| S-XX-FileConnection             | project:/rt.connection | 1.0     | ONLINE | 2019/10/22 12:49 PM | RT.User | *****        |
| S-XX-OracleConnection           | project:/rt.connection | 1.0     | ONLINE | 2019/10/22 10:57 AM | RT.User | *****        |
| SXX-OracleConnection            | project:/rt.connection | 1.0     | ONLINE | 2019/10/15 07:56 AM | RT.User | *****        |
| <b>spi.S_XX_Publish_Message</b> | project:               | 1.0     | ONLINE | 2019/11/03 07:18 PM | RT.User | *****        |

**Recommendation:** In case of multiple assets, you can filter the search based on the process name.



| Name                            | Contribution Path | Version | State  | Publish Date        | Group   | Published By |
|---------------------------------|-------------------|---------|--------|---------------------|---------|--------------|
| <b>spi.S_XX_Publish_Message</b> | project:          | 2.0     | ONLINE | 2020/08/14 09:21 PM | RT.User | *****        |

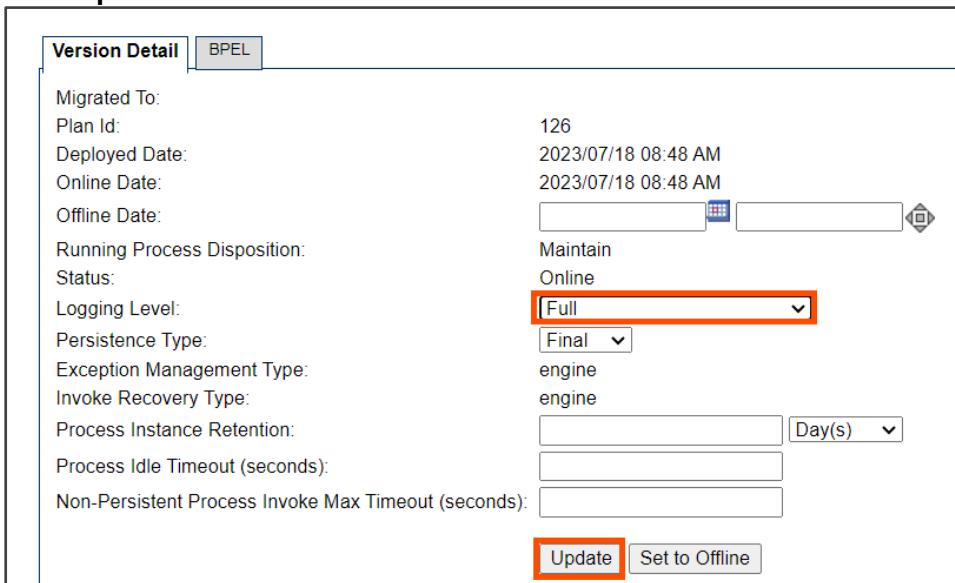
65. On the spi.Publish\_Message page, from the Deployed Processes section, select **S\_XX\_Publish\_Message**.



The screenshot shows the 'spi.S\_XX\_Publish\_Message' page. In the 'Deployed Processes' section, the row for 'S\_XX\_Publish\_Message' is highlighted with a red box. The table columns are 'Name', 'Target Namespace', 'Version', 'State', and 'Group'. The row data is: Name - S\_XX\_Publish\_Message, Target Namespace - um:screenflow:process:S\_XX\_Publish\_Message, Version - 1.0, State - ONLINE, Group - RT.User.

66. From the Logging Level drop-down, select **Full**.

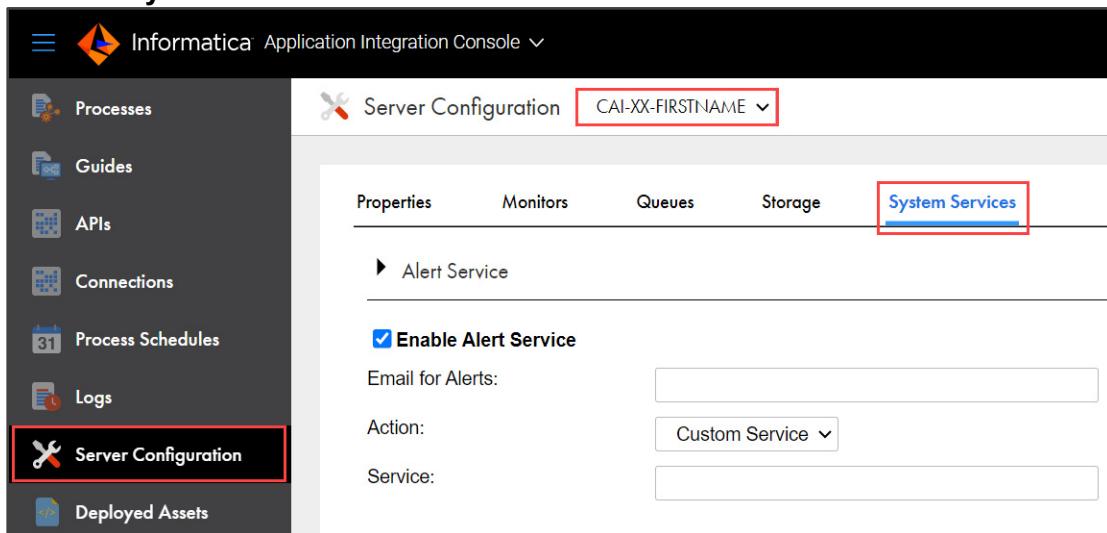
67. Click **Update**.



The screenshot shows the 'Version Detail' tab of a process configuration page. The 'Logging Level' dropdown is highlighted with a red box and contains the value 'Full'. Other visible settings include 'Status' (Online), 'Persistence Type' (Final), and 'Process Instance Retention' (Day(s)). At the bottom are 'Update' and 'Set to Offline' buttons.

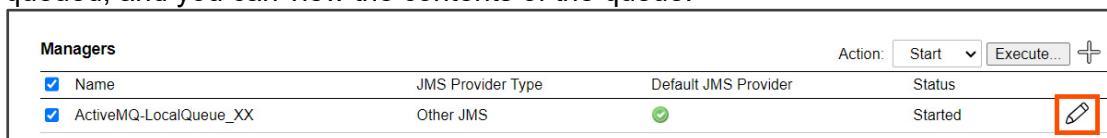
## Set the Listener Count for the messaging service to 0

68. From the navigation pane, select **Server Configuration**.
69. From the Server Configuration drop-down, ensure that your Secure Agent Group is selected.
70. Click the **System Services** tab.



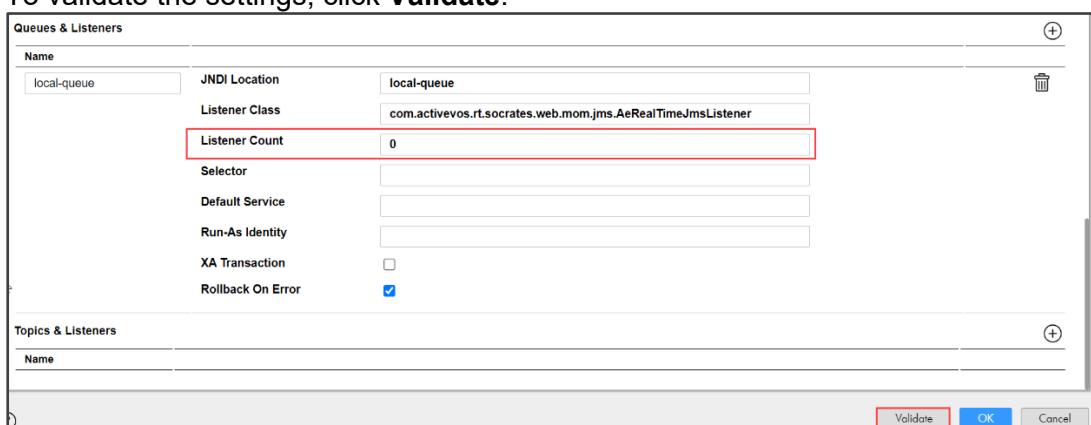
The screenshot shows the Informatica Application Integration Console interface. On the left is a dark sidebar with various icons and labels: Processes, Guides, APIs, Connections, Process Schedules, Logs, **Server Configuration** (which is highlighted with a red box), and Deployed Assets. The main area has a title bar 'Informatica Application Integration Console' with a dropdown set to 'CAI-XX-FIRSTNAME'. Below the title bar are tabs: Properties, Monitors, Queues, Storage, and **System Services** (which is also highlighted with a red box). Under the System Services tab, there's a section for 'Alert Service' with a checked checkbox 'Enable Alert Service'. Below it are fields for 'Email for Alerts' (empty), 'Action' (set to 'Custom Service'), and 'Service' (empty).

71. Expand **Messaging Service** and under Managers, select **ActiveMQ-LocalQueue\_XX**.
72. Click  to edit the number of consumers so that the message is not automatically dequeued, and you can view the contents of the queue.



| Managers   |                   |                      |   |
|--|-------------------|----------------------|---|
|  | Action:           | Start                | Execute...  |
| <input checked="" type="checkbox"/> Name                   | JMS Provider Type | Default JMS Provider | Status  |
| <input checked="" type="checkbox"/> ActiveMQ-LocalQueue_XX | Other JMS         | Started              |  |

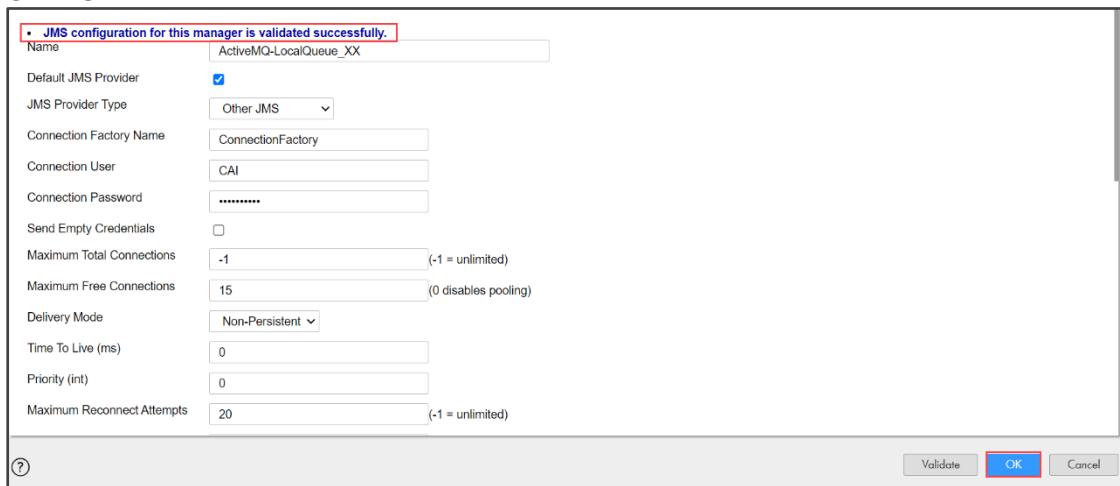
73. Under Queues & Listeners section, set the Listener Count to **0**.
74. To validate the settings, click **Validate**.



The screenshot shows the 'Queues & Listeners' configuration dialog. It has two sections: 'Queues & Listeners' and 'Topics & Listeners'. In the 'Queues & Listeners' section, the 'Name' field is 'local-queue'. Under 'Listener Location', 'JNDI Location' is 'local-queue' and 'Listener Class' is 'com.activevos.rt.socrates.web.mom.jms.AeRealTimeJmsListener'. The 'Listener Count' field is highlighted with a red box and contains the value '0'. Other fields include 'Selector', 'Default Service', 'Run-As Identity', 'XA Transaction' (unchecked), and 'Rollback On Error' (checked). At the bottom right are 'Validate', 'OK', and 'Cancel' buttons.

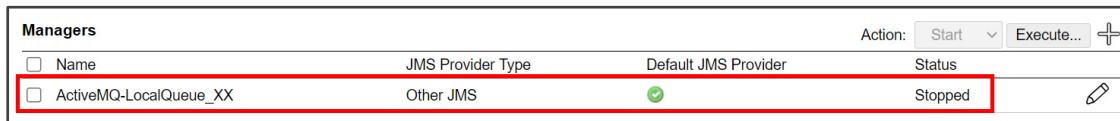
A message '**JMS configuration for this manager is validated successfully**' appears at the top of the Messaging Service page.

75. Click **OK**.



The dialog box displays the configuration for a JMS manager named "ActiveMQ-LocalQueue\_XX". The "Name" field is set to "ActiveMQ-LocalQueue\_XX". The "Default JMS Provider" checkbox is checked. The "JMS Provider Type" dropdown is set to "Other JMS". The "Connection Factory Name" is "ConnectionFactory", "Connection User" is "CAI", and "Connection Password" is masked. The "Send Empty Credentials" checkbox is unchecked. "Maximum Total Connections" is set to "-1" (unlimited). "Maximum Free Connections" is set to "15" (disables pooling). "Delivery Mode" is "Non-Persistent". "Time To Live (ms)" is "0". "Priority (int)" is "0". "Maximum Reconnect Attempts" is set to "20" (unlimited). At the bottom, there are "Validate", "OK", and "Cancel" buttons.

**Note:** You may notice that ActiveMQ-LocalQueue\_XX status is Stopped.

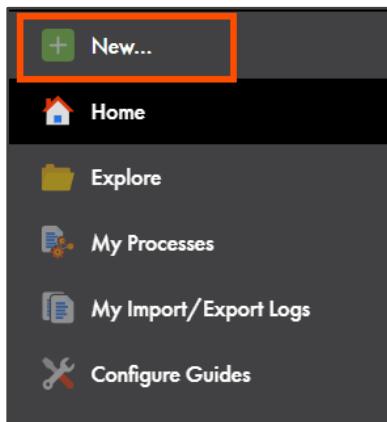


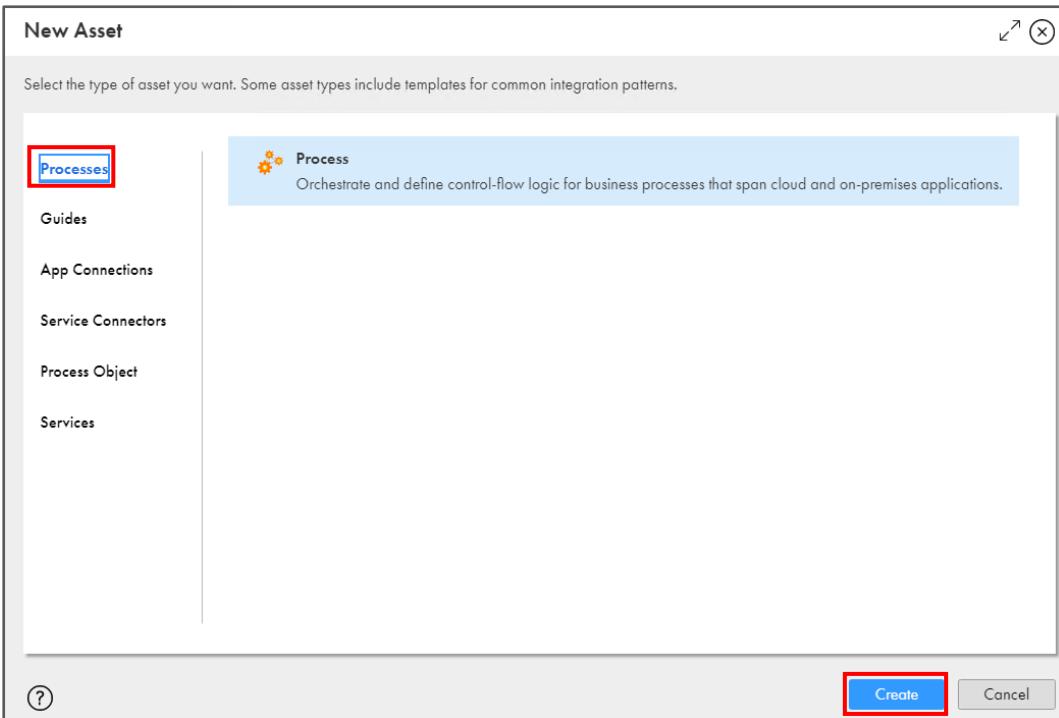
A table titled "Managers" showing a single row for "ActiveMQ-LocalQueue\_XX". The columns are "Name", "JMS Provider Type", "Default JMS Provider", and "Status". The "Name" is "ActiveMQ-LocalQueue\_XX", "JMS Provider Type" is "Other JMS", "Default JMS Provider" is checked (indicated by a green circle), and "Status" is "Stopped". A red box highlights the entire row.

### Create a Parent Process to Publish the Exchange Rate to a Queue:

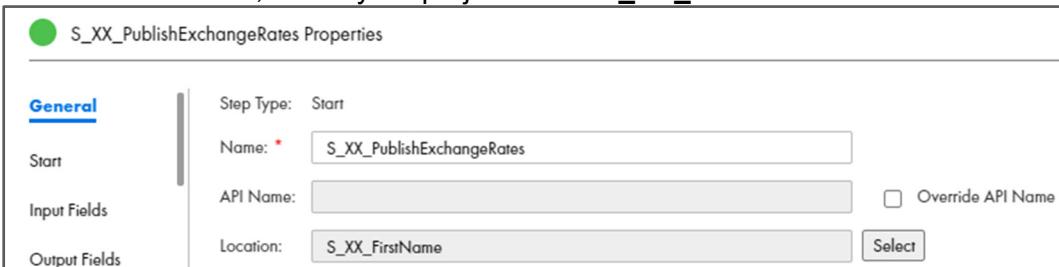
76. Navigate to the **Application Integration** service.

77. Create a new **Process**.

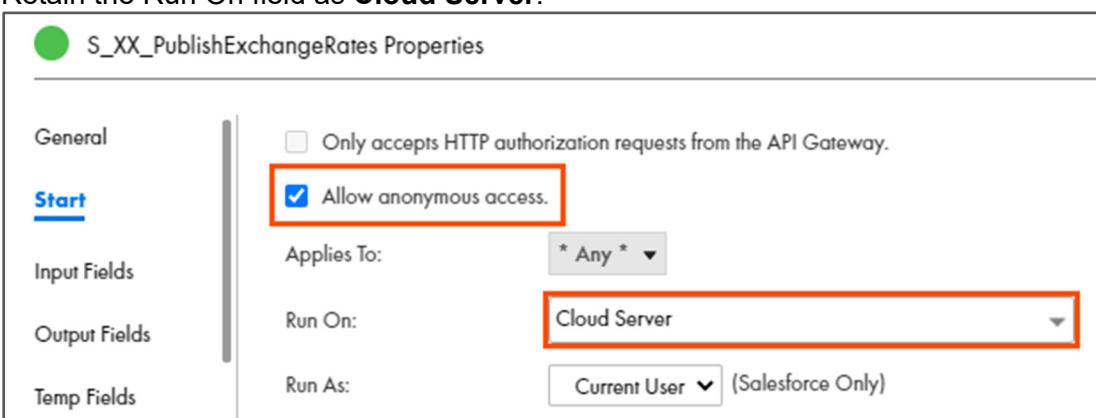




78. From the process canvas, select the **Start** step.
79. In the General tab, enter the name as **S\_XX\_PublishExchangeRates**.
80. Skip the API Name option.
81. In the Location field, select your project folder **S\_XX\_FirstName**.



82. From the properties pane, select **Start**.
83. Select the **Allow anonymous access** checkbox.
84. Retain the Run On field as **Cloud Server**.

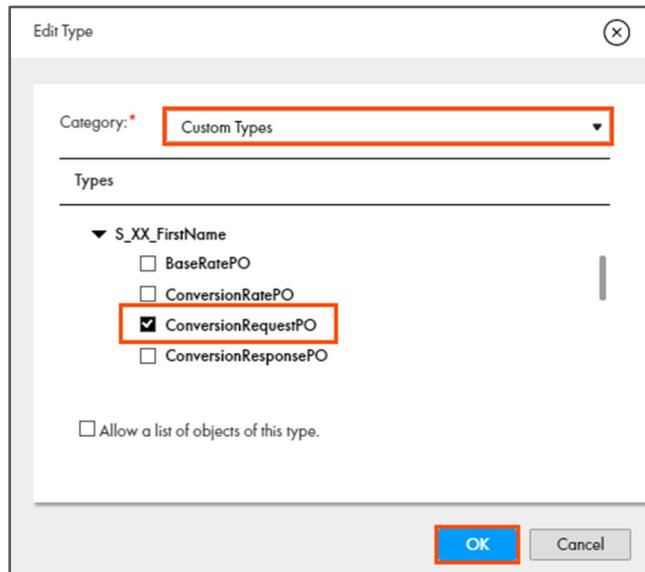


85. From the properties pane, select **Input Fields**.

86. To add input fields, click .

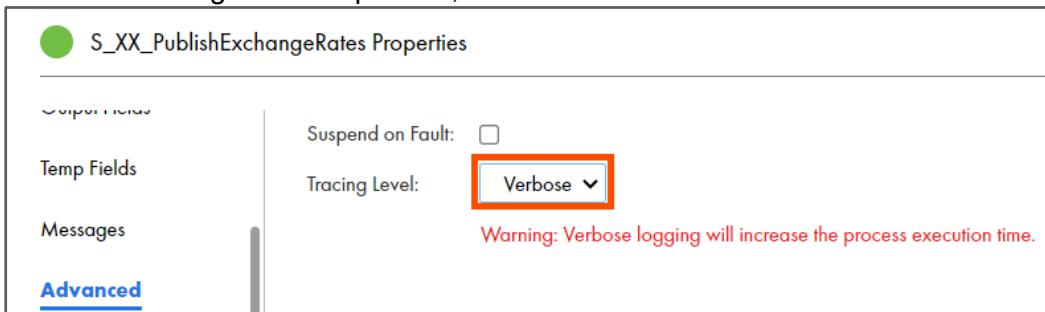
87. Create the input field, as shown in the table below:

| Name            | Type   |
|-----------------|--|
| ExchangeRequest | More types > Custom Types > S_XX_FirstName > ConversionRequestPO |

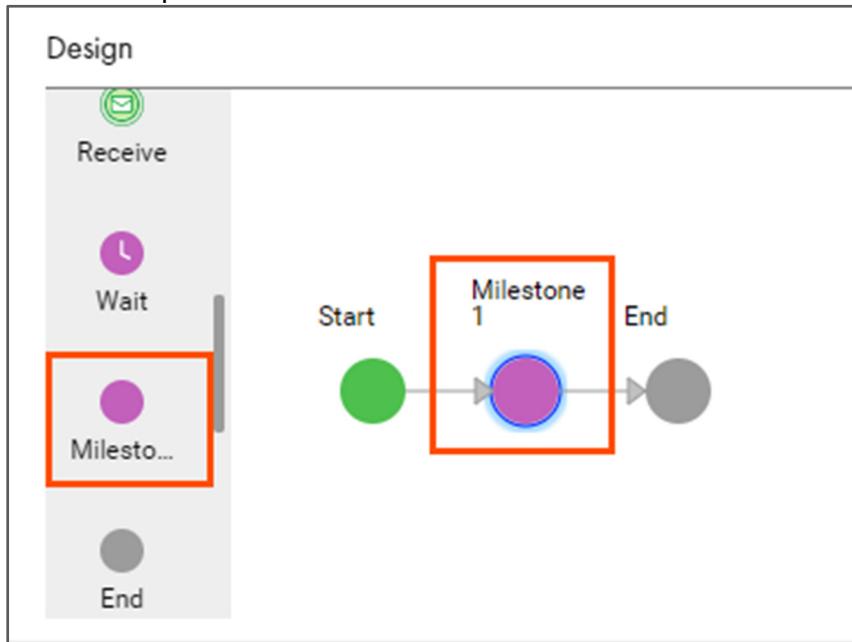


88. From the properties pane, select **Advanced**.

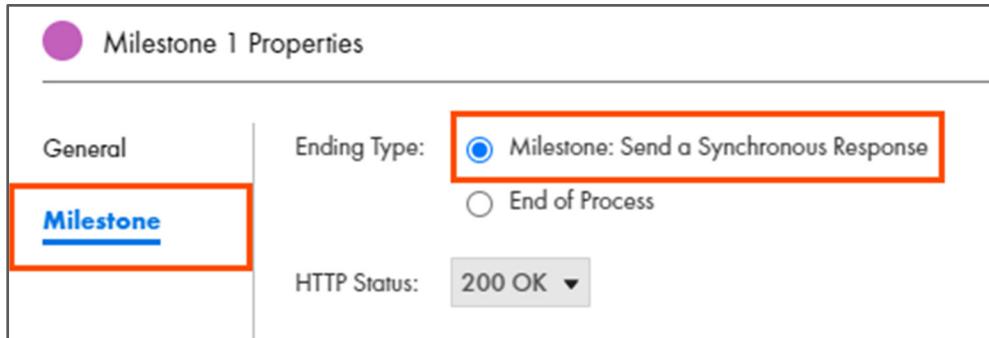
89. From the Tracing Level drop-down, select **Verbose**.



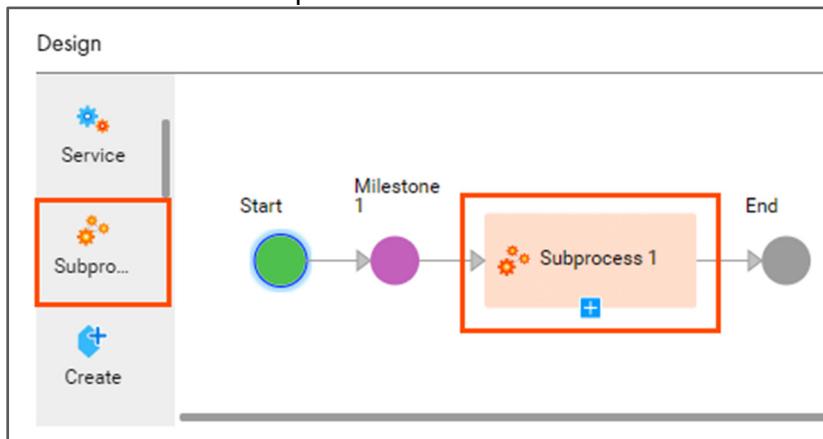
90. From the Design palette, drag and drop a **Milestone** step on the link between the Start and End steps.



91. From the properties pane, select **Milestone**.  
 92. Verify that the Ending Type is set as **Milestone: Send a Synchronous Response** and HTTP Status is **200 OK**.



93. From the Design palette, drag and drop a **Subprocess** step on the link between the Milestone and End steps.



94. From the properties pane, select **Subprocess**.

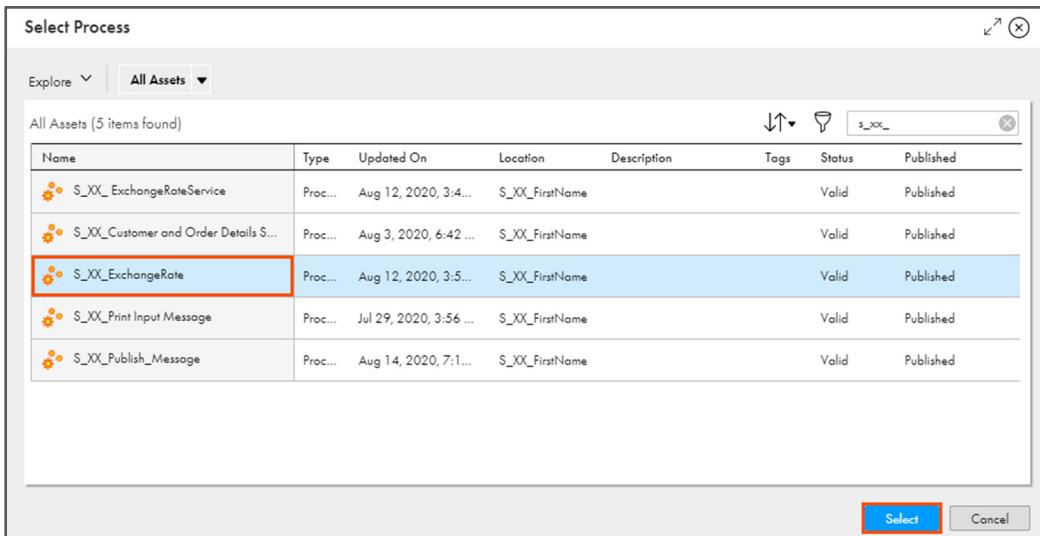
95. To select the subprocess, click **Select**.



A Select Process window appears.

96. From the list, select the **S\_XX\_ExchangeRate** process created the previous lab.

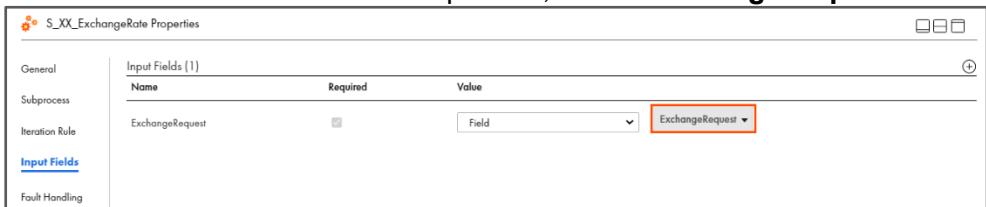
97. Click **Select**.



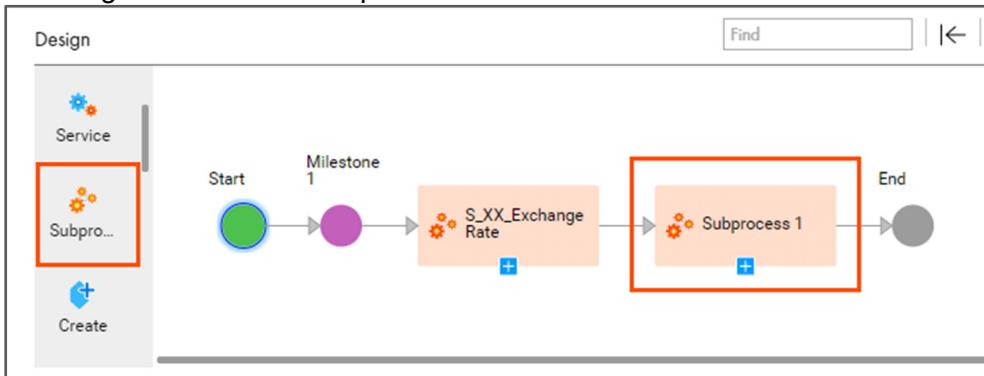
98. From the properties pane, select **Input Fields**.

99. You will see an Input Field named ExchangeRequest. If you do not see any field, add the input field using the plus (+) icon and select **ExchangeRequest**.

100. From the Click To Select Field drop-down, select **ExchangeRequest**.

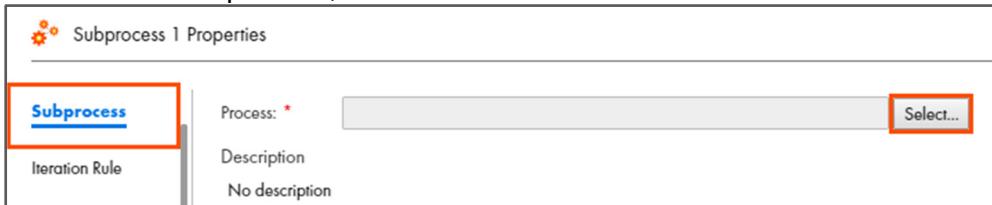


101. From the Design palette, drag and drop a **Subprocess** step on the link between the ExchangeRate and End steps.



102. From the properties pane, select **Subprocess**.

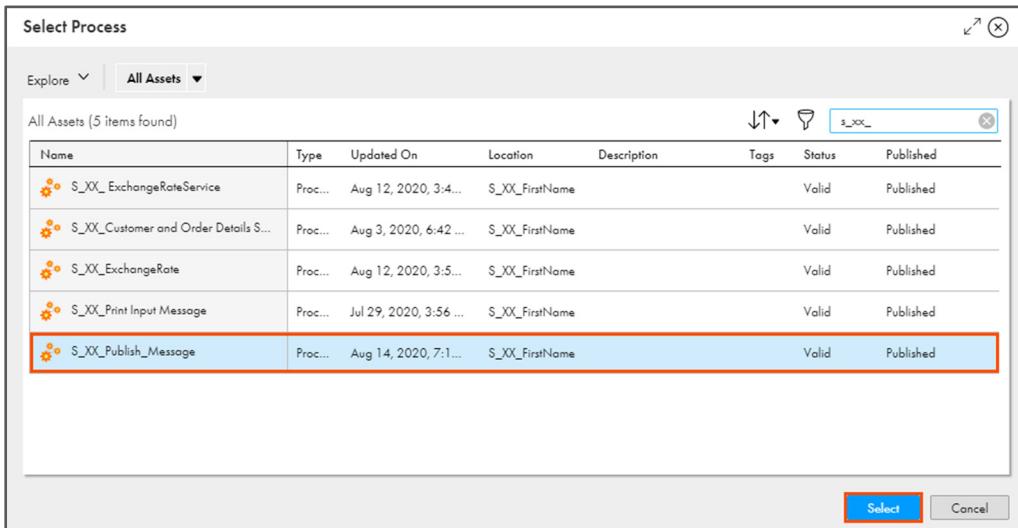
103. To select the subprocess, click **Select**.



A Select Process window appears.

104. From the list, select **S\_XX\_Publish\_Message** process.

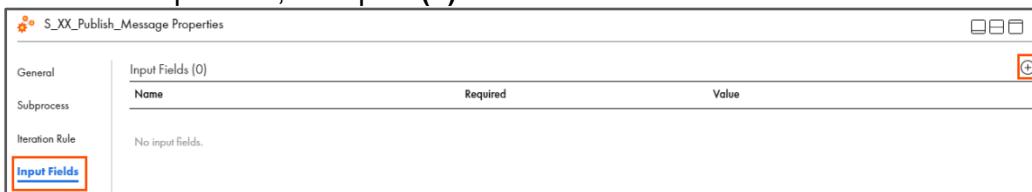
105. Click **Select**.



**Note:** This step adds the process that will publish the response to the JMS queue.

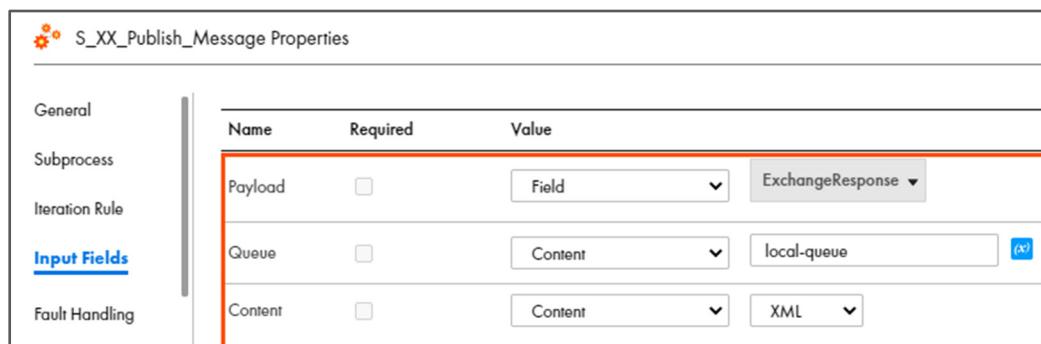
106. From the properties pane, select **Input Fields**.

107. To add the input field, click plus (+) icon.



108. Add the fields as shown in the table below:

| Name    | Value   | Type             |
|---------|---------|------------------|
| Payload | Field   | ExchangeResponse |
| Queue   | Content | local-queue      |
| Content | Content | XML              |



S\_XX\_Publish\_Message Properties

General Subprocess Iteration Rule **Input Fields** Fault Handling

| Name    | Required                 | Value                     |
|---------|--------------------------|---------------------------|
| Payload | <input type="checkbox"/> | Field<br>ExchangeResponse |
| Queue   | <input type="checkbox"/> | Content<br>local-queue    |
| Content | <input type="checkbox"/> | Content<br>XML            |

109. Save and Publish the process.



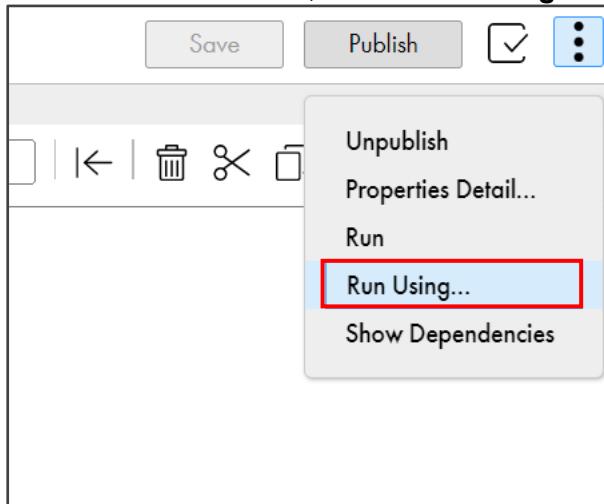
The asset [S\_XX\_PublishExchangeRates] was published successfully.

Save Publish

Milestone 1 → S\_XX\_Exchange Rate → S\_XX\_Publish\_Message → End

## Execute the Process

110. From the Actions menu, select Run Using....



Save Publish

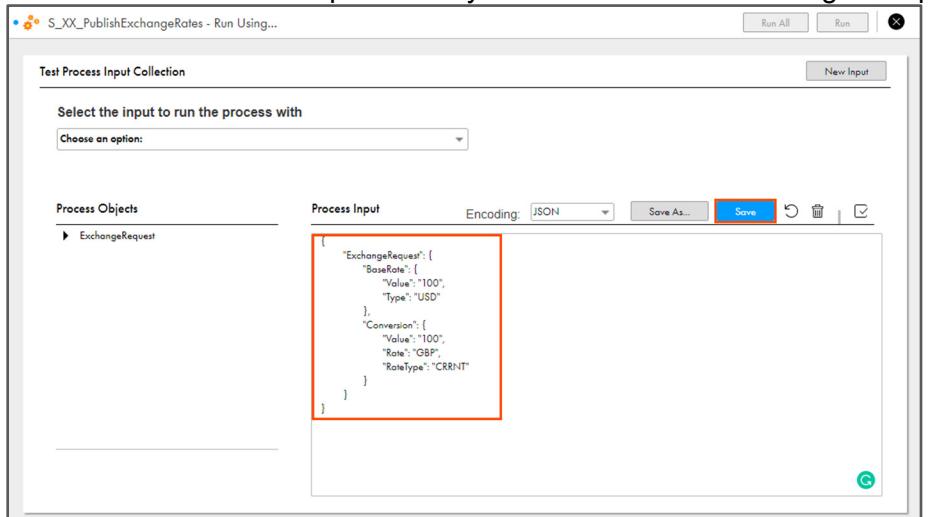
Unpublish Properties Detail... Run

**Run Using...**

Show Dependencies

111. In the Process Input section, for the BaseRate object, add the Value as **100** and Type as **USD**.

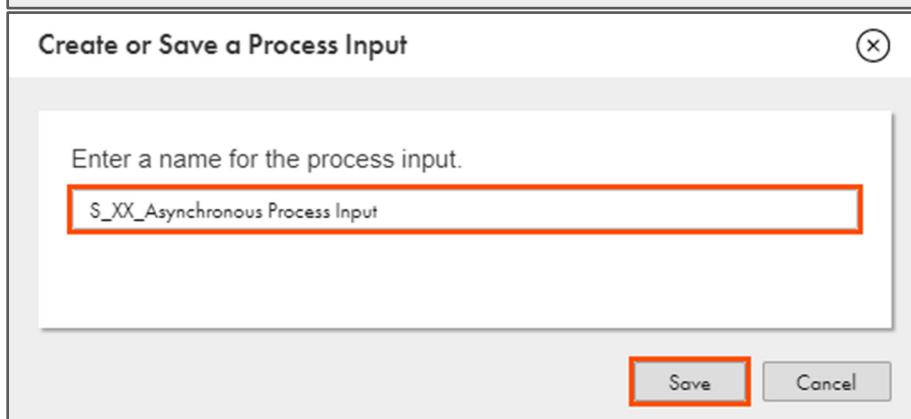
112. For the Conversion object, add the Value as **100**, Rate as **GBP**, and the RateType as **CRRNT**. Click **Save** and provide any random name while saving the input.



The screenshot shows the 'Test Process Input Collection' window. On the left, under 'Process Objects', there is a tree view with 'ExchangeRequest' expanded, showing 'BaseRate' and 'Conversion' objects. On the right, under 'Process Input', the JSON input definition is displayed:

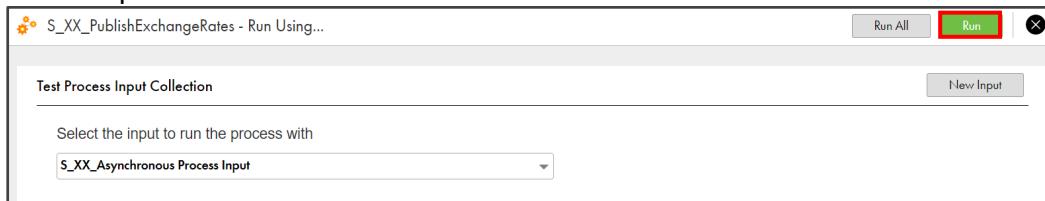
```
[{"ExchangeRequest": { "BaseRate": [ {"Value": "100", "Type": "USD"} ], "Conversion": [ {"Value": "100", "Rate": "GBP", "RateType": "CRRNT"} ] }}
```

The 'Save' button is highlighted with a red box.



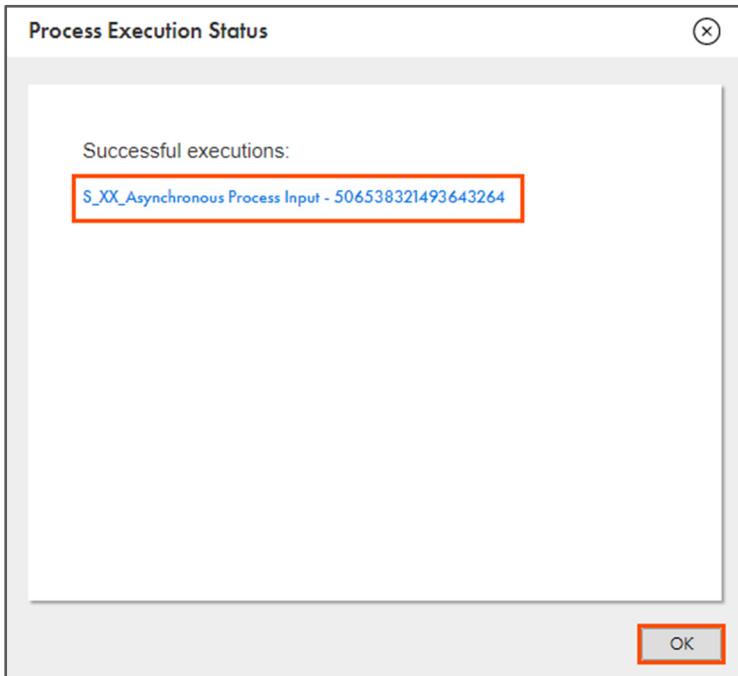
The screenshot shows the 'Create or Save a Process Input' dialog box. It has a text input field labeled 'Enter a name for the process input.' containing the value 'S\_XX\_Asyncronous Process Input'. The 'Save' button is highlighted with a red box.

113. Run the process.

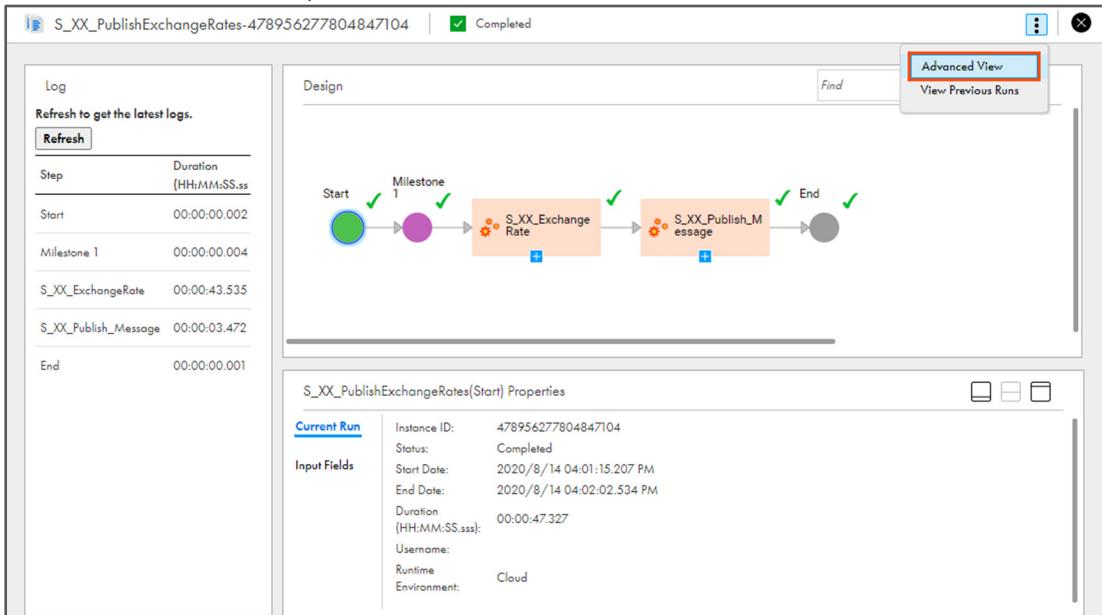


The screenshot shows the 'Test Process Input Collection' window again. The 'Run' button is highlighted with a red box. The status bar at the bottom indicates 'Running'.

114. When the process executes successfully, to check the output, click the execution link, and click **OK**.

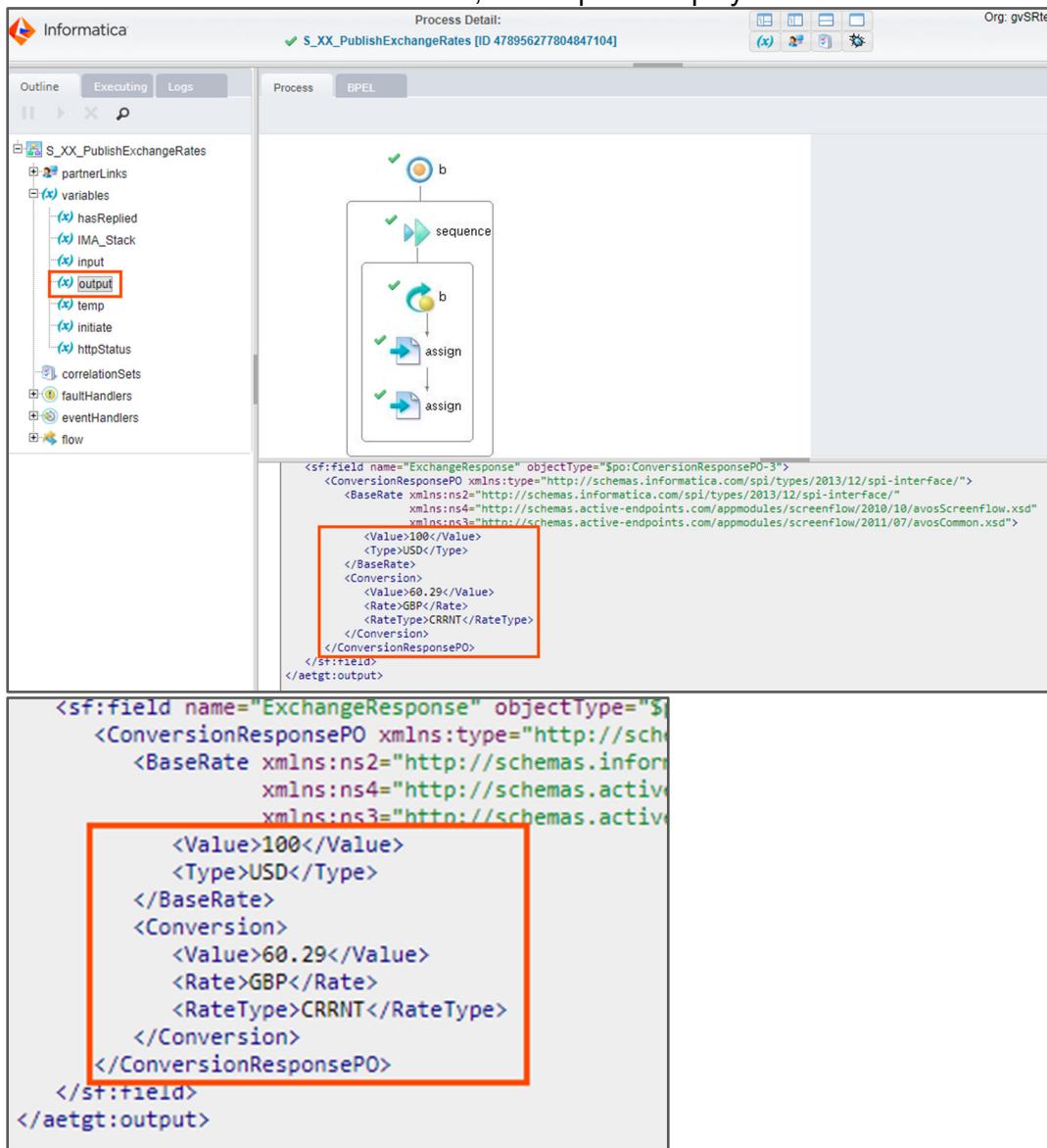


115. From the Actions menu, click **Advanced View**.



116. In the Advanced View, expand the **variables** section, and click **output**.

117. In the Variable Instance Data section, the output is displayed.



```

<sf:field name="ExchangeResponse" objectType="$po:ConversionResponsePO-3">
    <ConversionResponsePO xmlns:type="http://schemas.informatica.com/spi/types/2013/12/spi-interface/">
        <BaseRate xmlns:ns2="http://schemas.informatica.com/spi/types/2013/12/spi-interface/">
            <Value>100</Value>
            <Type>USD</Type>
        </BaseRate>
        <Conversion>
            <Value>60.29</Value>
            <Rate>GBP</Rate>
            <RateType>CRRNT</RateType>
        </Conversion>
    </ConversionResponsePO>
</sf:field>
</aetgt:output>

```

In this way, you can invoke a process using an asynchronous web service call and view the response stored in the JMS queue.

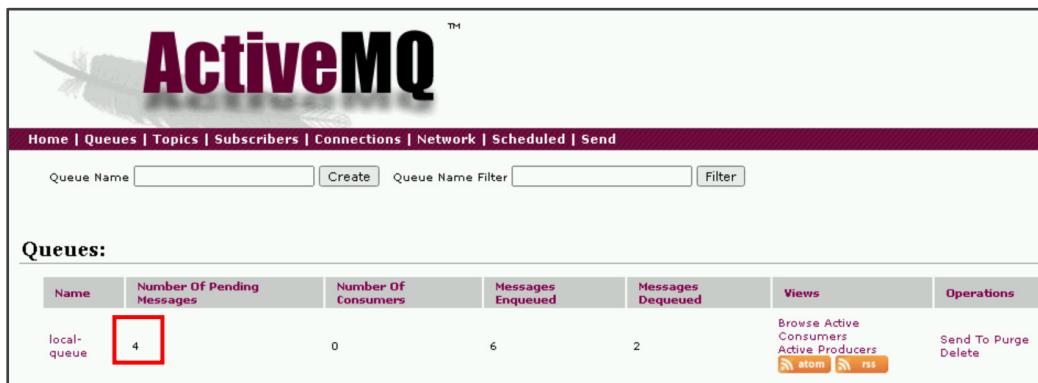
## Observation

118. Go back to the ActiveMQ page in the browser (<http://localhost:8161/admin/>), click the **Queues** tab.

**Note:** Refresh the ActiveMQ page if the Queues tab is already open.

119. On the ActiveMQ homepage, click the **Queues** tab.

120. Observe the **Number of Pending Messages** column.



The screenshot shows the ActiveMQ Admin Console interface. At the top, there's a navigation bar with links: Home, Queues, Topics, Subscribers, Connections, Network, Scheduled, and Send. Below the navigation bar, there are search fields for Queue Name and Queue Name Filter, along with a Filter button. The main area is titled "Queues:" and contains a table. The table has columns: Name, Number Of Pending Messages, Number Of Consumers, Messages Enqueued, Messages Dequeued, Views, and Operations. One row in the table is highlighted with a red box around the "Number Of Pending Messages" cell, which contains the value "4". To the right of the table, there are links for Browse Active Consumers, Active Producers, atom, and rss, along with buttons for Send To Purge and Delete.

| Name        | Number Of Pending Messages | Number Of Consumers | Messages Enqueued | Messages Dequeued | Views   | Operations  |
|-------------|----------------------------|---------------------|-------------------|-------------------|---|---|
| local-queue | 4                          | 0                   | 6                 | 2                 | <a href="#">Browse Active Consumers</a><br><a href="#">Active Producers</a><br><a href="#">atom</a> <a href="#">rss</a> | <a href="#">Send To Purge</a><br><a href="#">Delete</a> |

**Note:** The value of Number of Pending Messages depends on the number of times the process S\_XX\_PublishExchangeRates runs in the local-queue.

*This concludes the lab.*

# Module 11: Fault Handling

## Lab 11-1: Handle Sub-process Failures

### Overview:

In this lab, you will configure processes that handle process execution faults. You will then run the process and view failed processes and error messages. For the process, you will customize the error messages so that business users receive simple messages with failure reasons, instead of long technical messages that are difficult to decode.

### Objectives:

- Configure error handling for sub-process failure
- Run the process
- Check failed process instance

### Duration:

15 Minutes

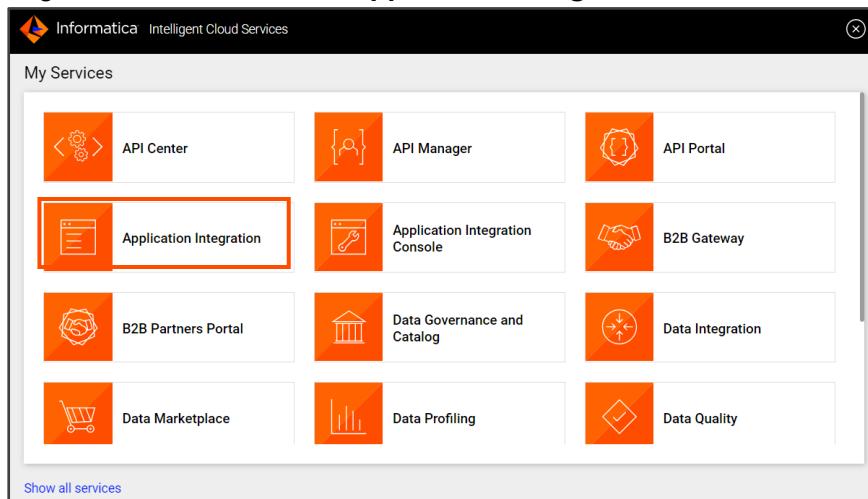
---

### Tasks:

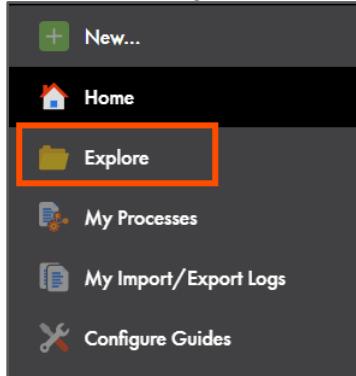
**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

### Configure error handling for sub-process failure

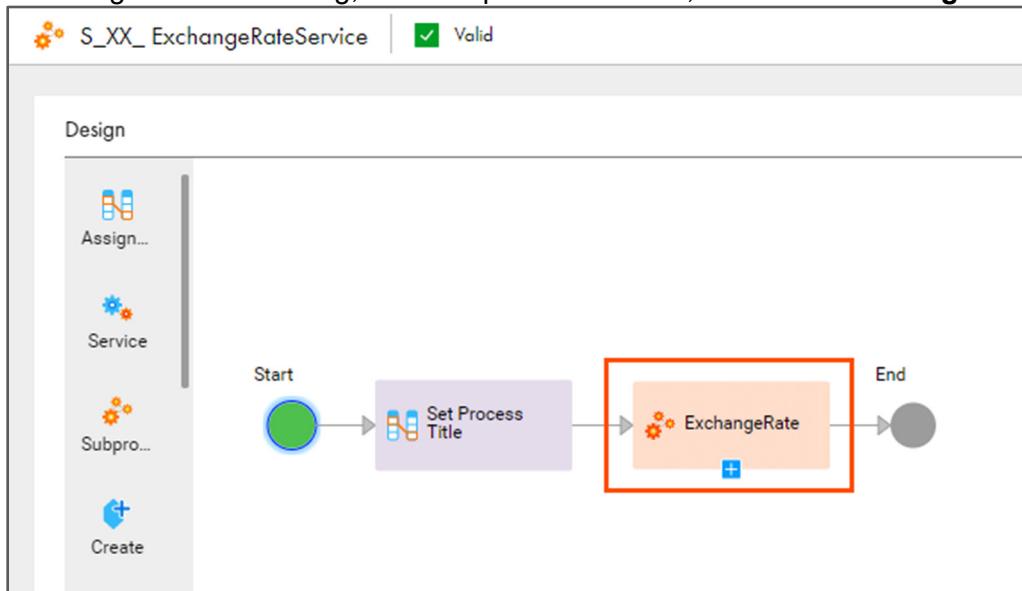
1. Log in to IICS and access **Application Integration**.



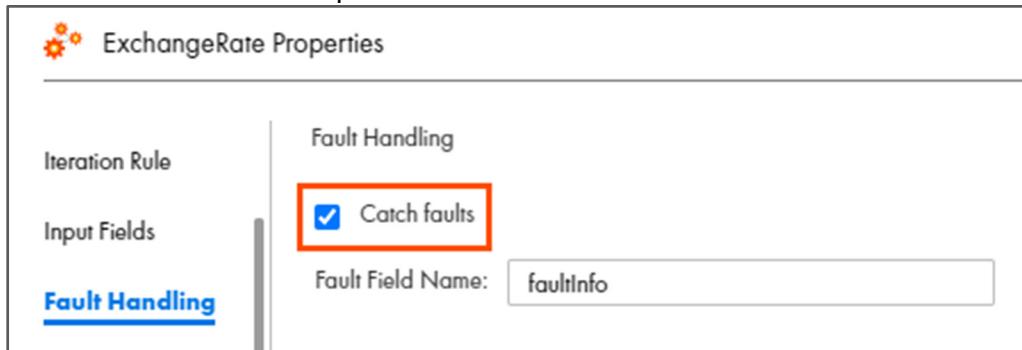
- From the Navigation pane, click **Explore**.



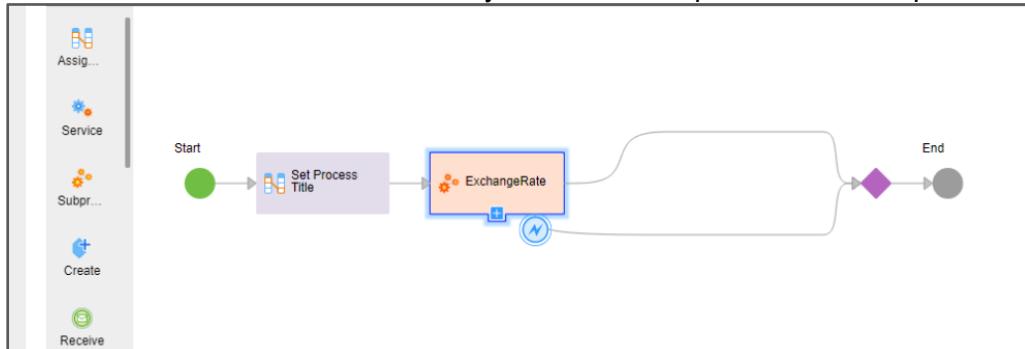
- Navigate to your project folder and open the **S\_XX\_ExchangeRateService** process.
- To configure fault handling, from the process canvas, select the **ExchangeRate** step.



- From the properties pane, select **Fault Handling**.
- Select the **Catch faults** option.



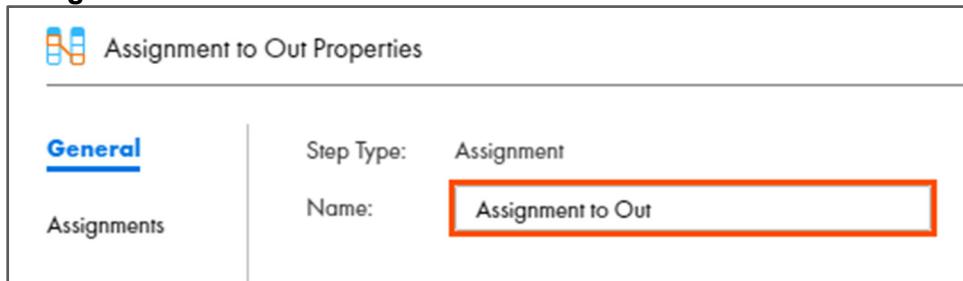
**Note:** Observe that IICS automatically adds two new paths for the subprocess step.



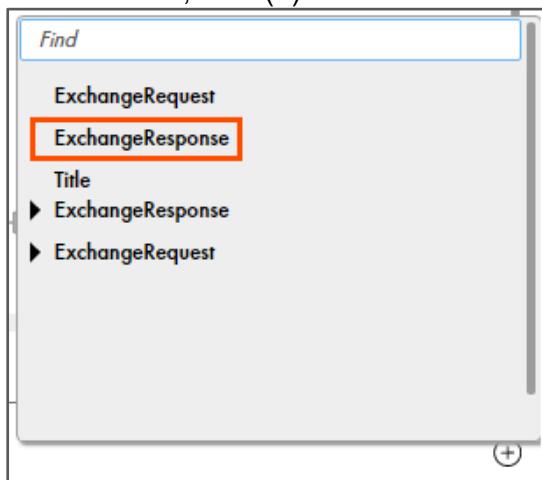
- From the Design palette, drag and drop an **Assignment** step on the upper path.



- From the Assignment properties pane, in the General tab, enter the Name as **Assignment to Out**.



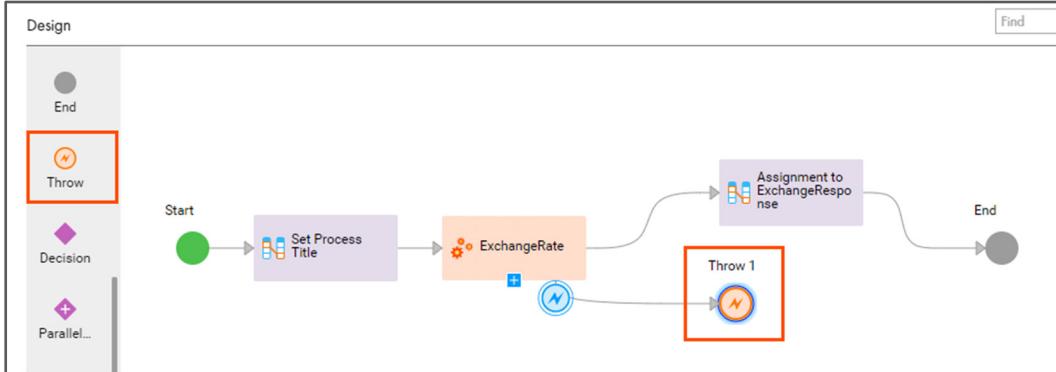
- From the properties pane, select **Assignments**.
- To add a field, click (+) icon and select **ExchangeResponse**.



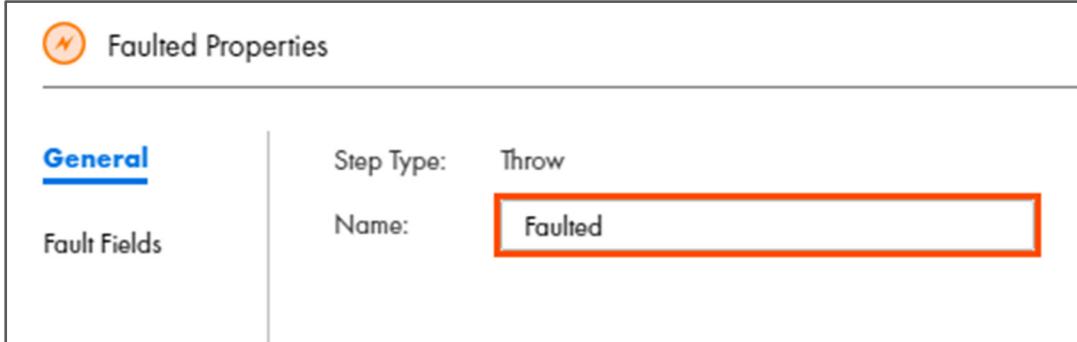
11. From the Assigned Using drop-down, retain **Field**.
12. From the From drop-down, select **ExchangeResponse**.



13. From the Design palette, drag and drop a **Throw** step on the lower path.



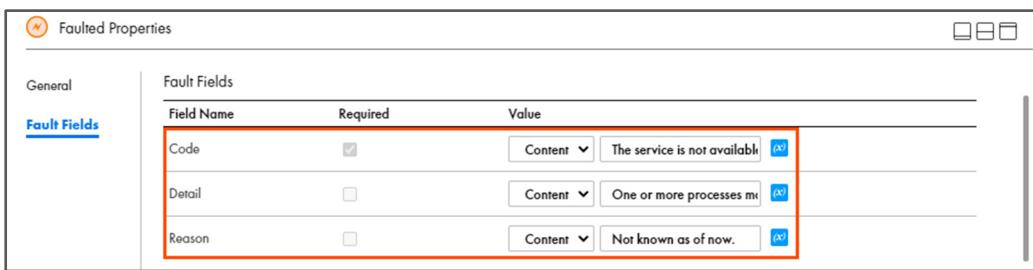
14. From the Throw properties pane, in the General tab, enter the Name as **Faulted**.



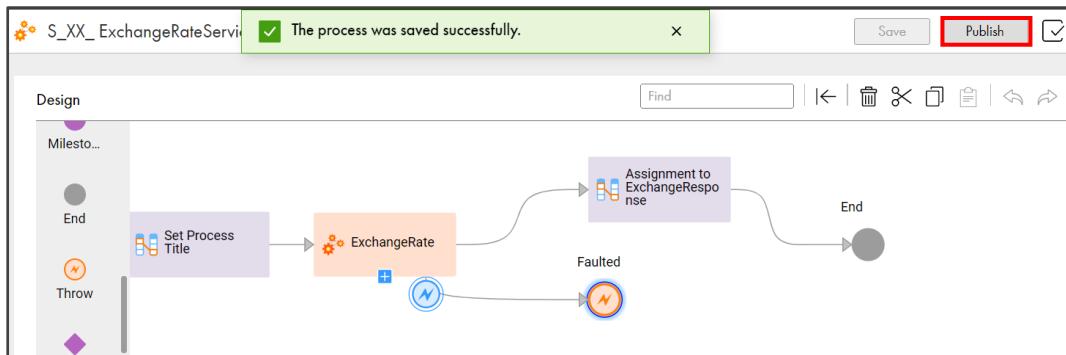
15. From the properties pane, select **Fault Fields**.

16. For the Fault Fields, configure the values as shown in the table below:

| Field Name | Value   | Type   |
|------------|---------|--|
| Code       | Content | The service is not available. Please try after some time.    |
| Detail     | Content | One or more processes may not be executing, hence the fault. |
| Reason     | Content | Not known as of now.   |



17. Save and Publish the process.

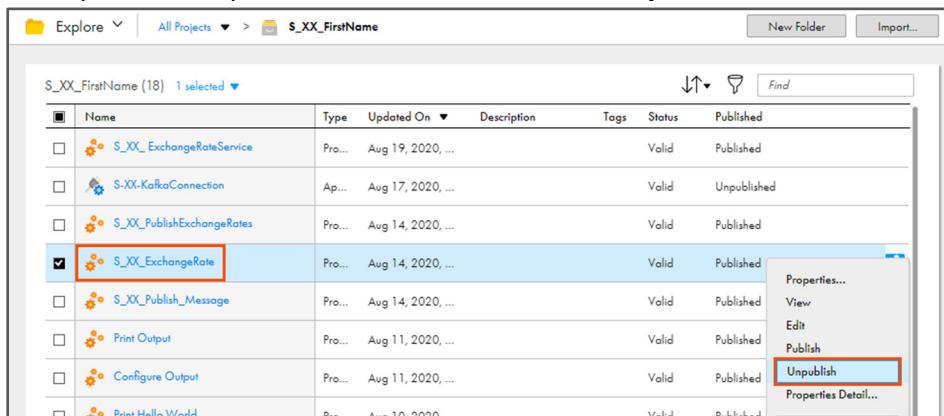


**Note:** If the Publish option is disabled, unpublish the process, save, and then re-publish the process.

Now, to understand fault handling, you will introduce an error to the process.

18. Navigate to your project folder and locate **S\_XX\_ExchangeRate** process.

19. To unpublish the process, click  and select **Unpublish**.

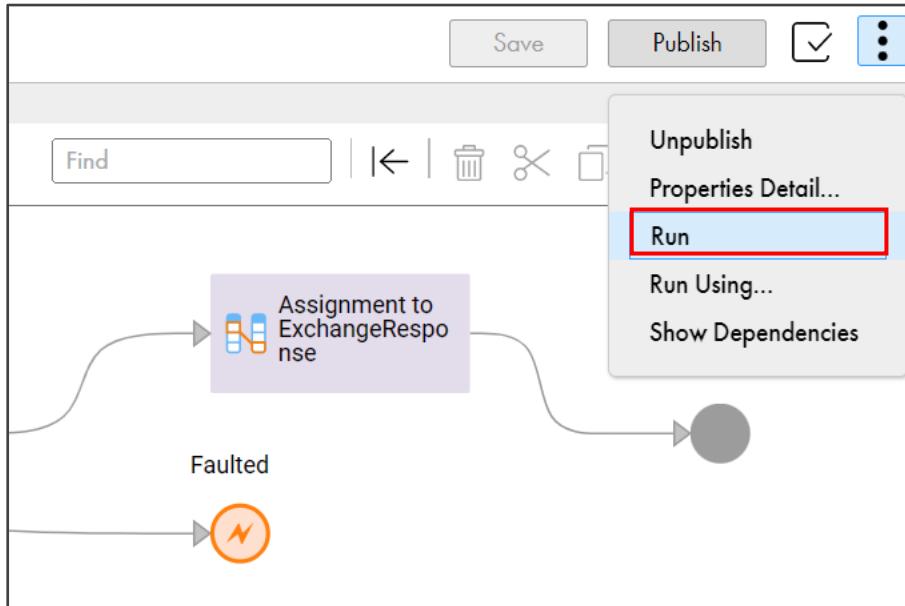


| Name                      | Type       | Updated On        | Description | Tags | Status | Published   |
|---------------------------|------------|-------------------|-------------|------|--------|-------------|
| S_XX_ExchangeRateService  | Process    | Aug 19, 2020, ... |             |      | Valid  | Published   |
| S_XX_KafkaConnection      | Connection | Aug 17, 2020, ... |             |      | Valid  | Unpublished |
| S_XX_PublishExchangeRates | Process    | Aug 14, 2020, ... |             |      | Valid  | Published   |
| <b>S_XX_ExchangeRate</b>  | Process    | Aug 14, 2020, ... |             |      | Valid  | Published   |
| S_XX_Publish_Message      | Process    | Aug 14, 2020, ... |             |      | Valid  | Published   |
| Print Output              | Process    | Aug 11, 2020, ... |             |      | Valid  | Published   |
| Configure Output          | Process    | Aug 11, 2020, ... |             |      | Valid  | Published   |
| Print Hello World         | Process    | Aug 10, 2020, ... |             |      | Valid  | Published   |

**Note:** Unpublishing the ExchangeRate process introduces an error when you run the ExchangeRateService Process. You can use this error to understand the fault handling techniques used in CAI.

## Execute the Process

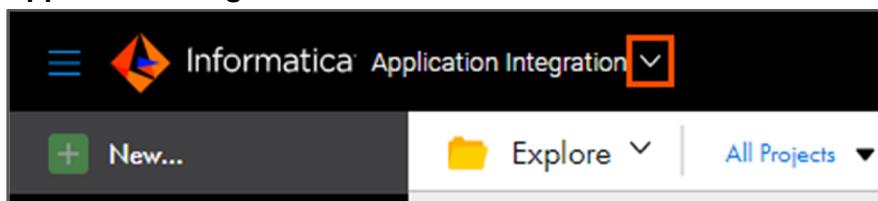
20. Navigate back to the **S\_XX\_ExchangeRateService** process and from the **Actions** menu, click **Run**.



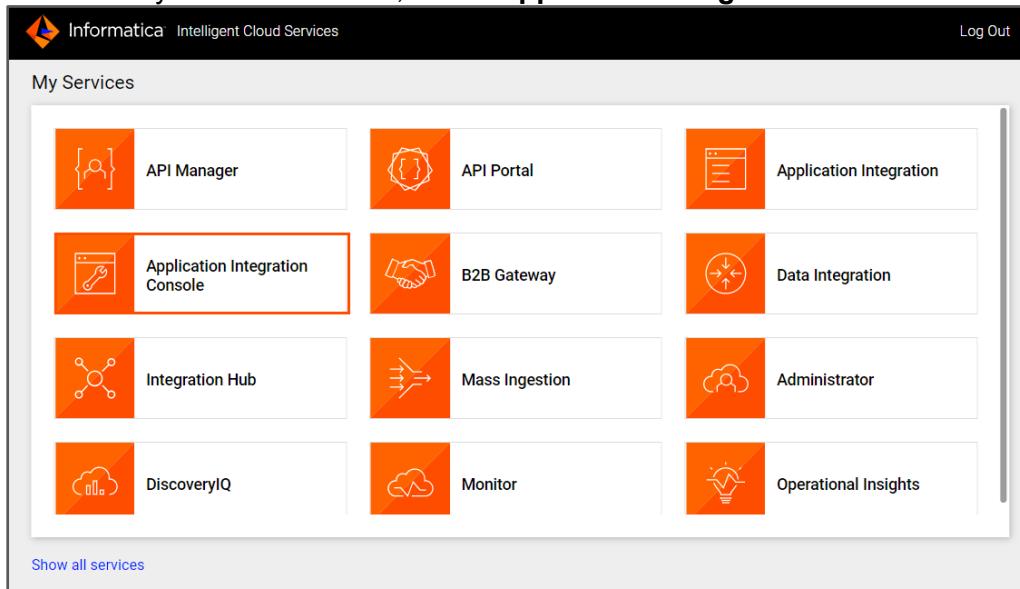
**Note:** The Run option executes the process based on the previous process input provided to it.

## Observe the Fault Process

21. To switch between the available services, from the toolbar, select the drop-down next to **Application Integration**.



22. From the My Services window, select **Application Integration Console**.

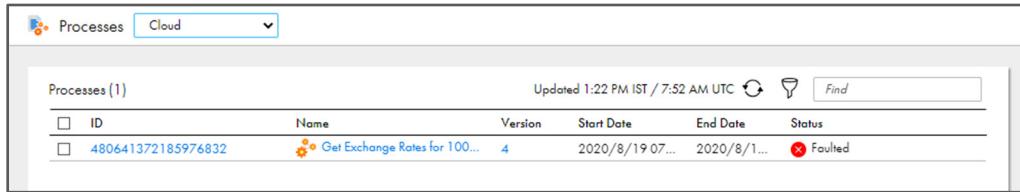


The screenshot shows the 'My Services' dashboard of the Intelligent Cloud Services platform. It displays nine service tiles arranged in a 3x3 grid:

- API Manager
- API Portal
- Application Integration
- Application Integration Console (highlighted with a red border)
- B2B Gateway
- Data Integration
- Integration Hub
- Mass Ingestion
- Administrator
- DiscoveryIQ
- Monitor
- Operational Insights

A 'Show all services' link is located at the bottom left of the grid.

23. In the Processes page, refresh the page periodically using the **Refresh** icon and observe the status of the service as **Faulted**.



The screenshot shows the 'Processes' page. The header includes tabs for 'Processes' (selected) and 'Cloud', and a dropdown menu. The main area displays a table titled 'Processes (1)' with the following data:

| ID                 | Name                          | Version | Start Date      | End Date    | Status  |
|--------------------|-------------------------------|---------|-----------------|-------------|---------|
| 480641372185976832 | Get Exchange Rates for 100... | 4       | 2020/8/19 07... | 2020/8/1... | Faulted |

At the top right of the table, there is a message: 'Updated 1:22 PM IST / 7:52 AM UTC' with a refresh icon, and a 'Find' search bar.

It may take 2 to 3 minutes for the process to finish running.

---

*This concludes the lab.*

# Module 11: Fault Handling

## Lab 11-2: Handle Credential Failures

### Overview:

In this lab, you will introduce an error in the Salesforce connection by changing your account password in Salesforce. You will then create a process that uses the Salesforce connection and using the Catch Faults feature, you will customize the error message in the process.

### Objectives:

- Change the Salesforce password to introduce an error
- Create a process
- Enable Fault Handling to simplify the error message

### Duration:

15 Minutes

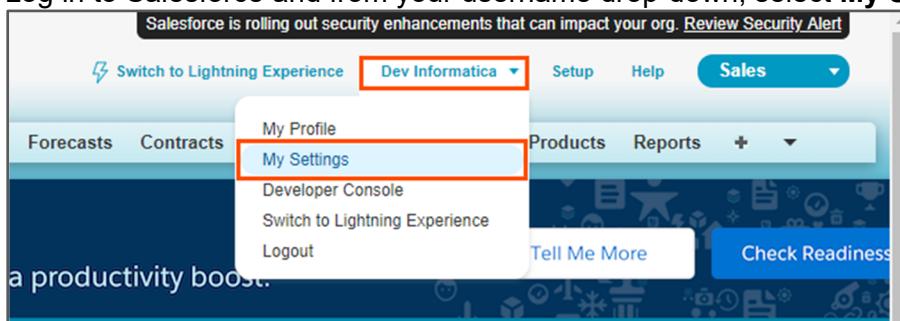
---

### Tasks

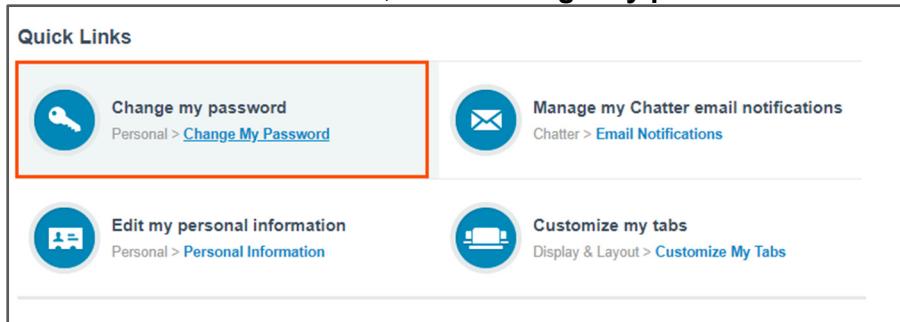
**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

#### Change Salesforce Password to Introduce Error

1. In a web browser, open Salesforce using the following URL:  
<https://login.salesforce.com/>
2. Log in to Salesforce and from your username drop-down, select **My Settings**.



3. From the Quick Links section, select **Change my password**.



4. Enter the details and click **Save**.

Change My Password

\* Current Password  
.....  
↑ Caps Lock is on.

New Password  
..... Good

Confirm New Password  
..... Match

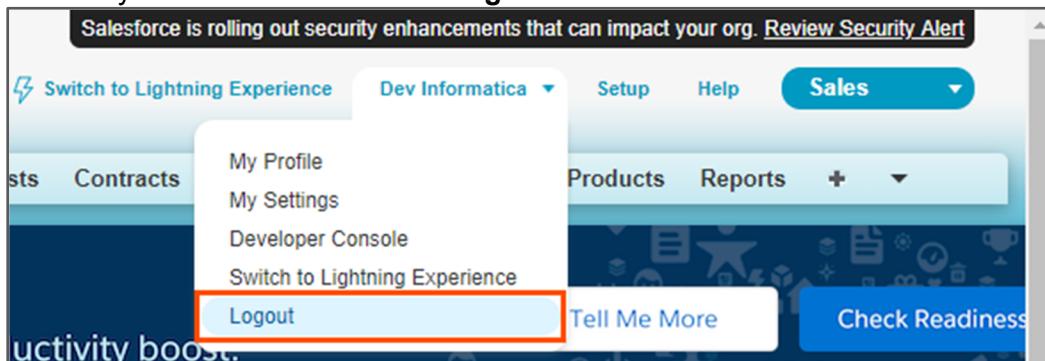
Security Question  
.....

Answer  
.....

Your security token is tied to your password and subject to any password policies your administrators have configured. Whenever your password is reset, your security token is also reset.

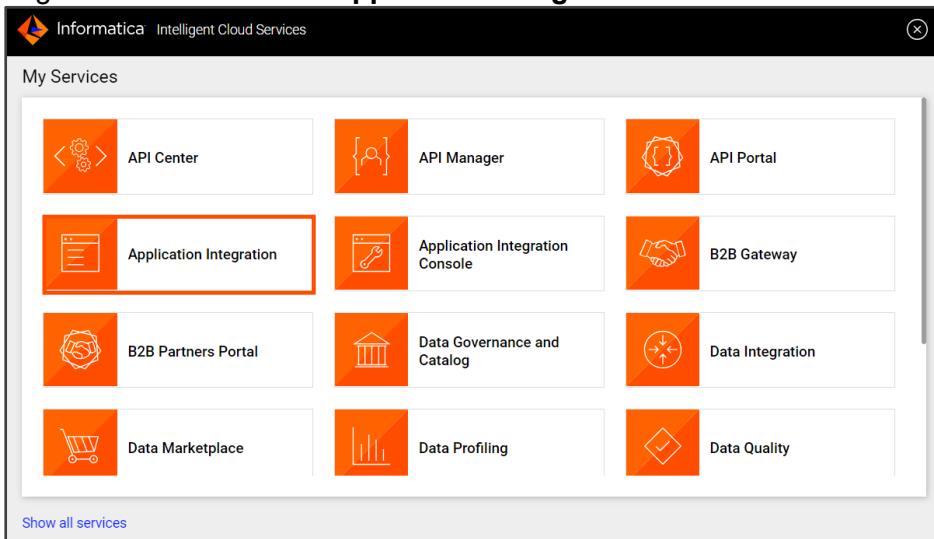
**Save**

5. Click on your Username and click **Logout**.

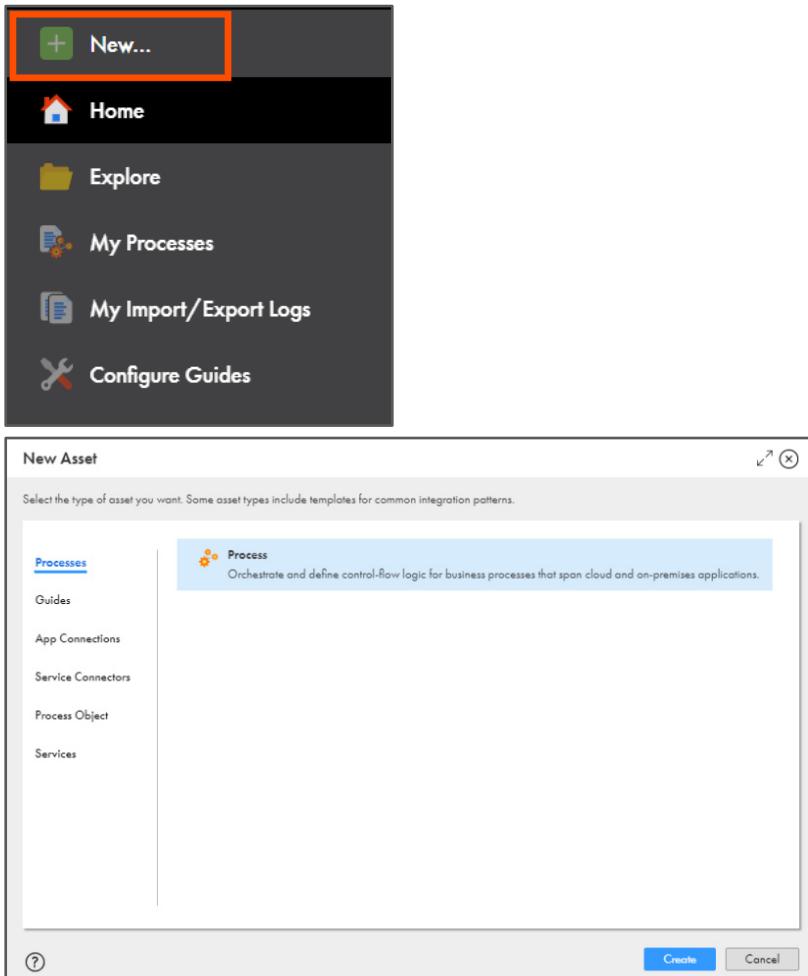


## Create a process

6. Log in to IICS and select **Application Integration**.



7. Create a new Process.

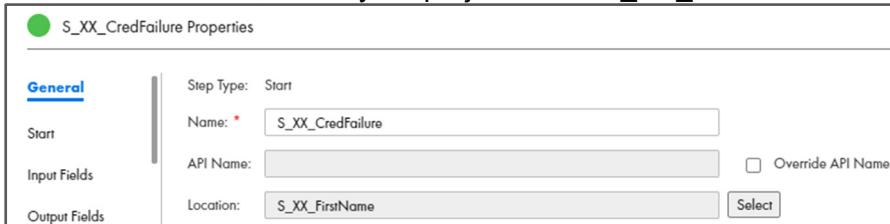


The image contains two screenshots of the application interface:

- Top Screenshot:** Shows the main menu with a 'New...' button highlighted by a red box. The menu items are: Home, Explore, My Processes, My Import/Export Logs, and Configure Guides.
- Bottom Screenshot:** Shows the 'New Asset' dialog box. On the left is a sidebar with categories: Processes, Guides, App Connections, Service Connectors, Process Object, and Services. The 'Processes' category is selected. On the right, there is a detailed view of the 'Process' asset type, which is described as "Orchestrate and define control-flow logic for business processes that span cloud and on-premises applications." At the bottom right of the dialog are 'Create' and 'Cancel' buttons.

A new Process window appears.

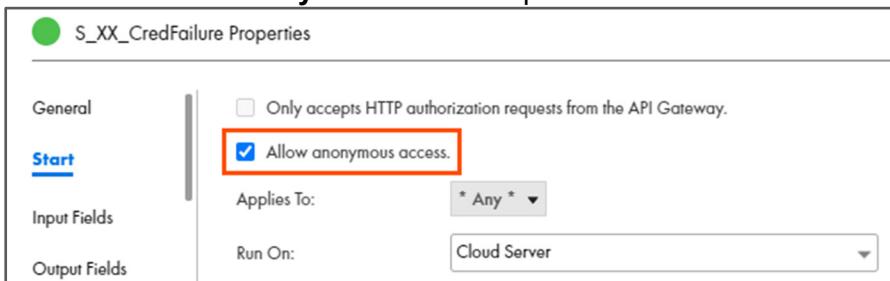
8. From the Process Canvas, select the **Start** step.
9. In the General tab, enter the name as **S\_XX\_CredFailure**.
10. Skip the API Name option.
11. In the Location field, select your project folder **S\_XX\_FirstName**.



The screenshot shows the 'General' tab of the properties dialog for the 'S\_XX\_CredFailure' step. The 'Step Type' is set to 'Start'. The 'Name' field contains 'S\_XX\_CredFailure'. The 'Location' field is set to 'S\_XX\_FirstName'. There is an 'Override API Name' checkbox, which is unchecked.

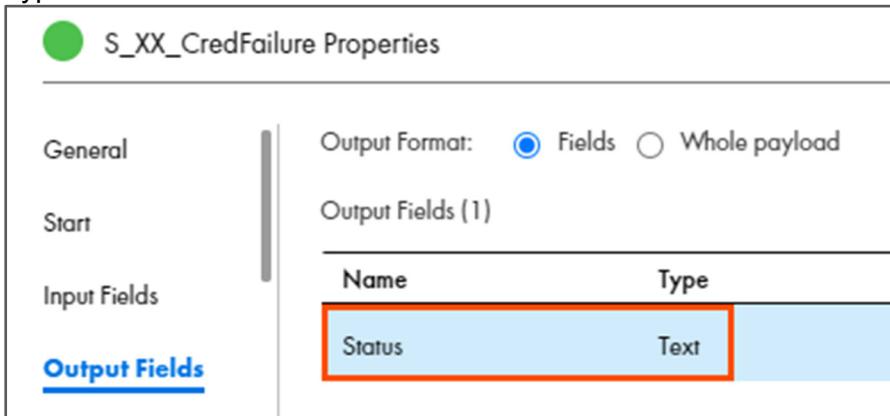
12. From the Start properties pane, select **Start**.

13. Select the **Allow anonymous access** option.



The screenshot shows the 'Start' tab of the properties dialog. The 'Allow anonymous access' checkbox is checked. Other settings include 'Applies To: \* Any \*' and 'Run On: Cloud Server'.

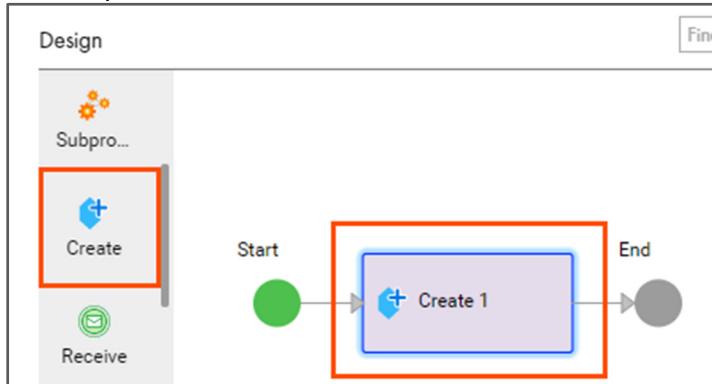
14. In the Output Fields tab, click **+**, and in the Name field, enter Name as **Status** and Type as **Text**.



The screenshot shows the 'Output Fields' tab of the properties dialog. It displays a table with one row. The row has two columns: 'Name' (containing 'Status') and 'Type' (containing 'Text'). The entire row is highlighted with a red box.

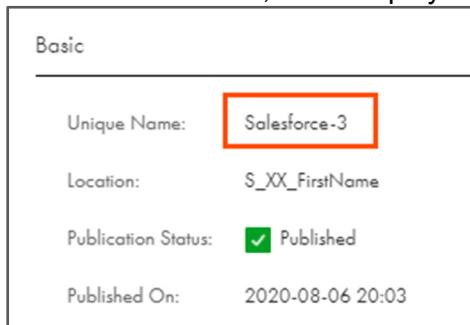
| Name   | Type |
|--------|------|
| Status | Text |

15. From the Design palette, drag and drop a **Create** step on the link between the Start and End steps.



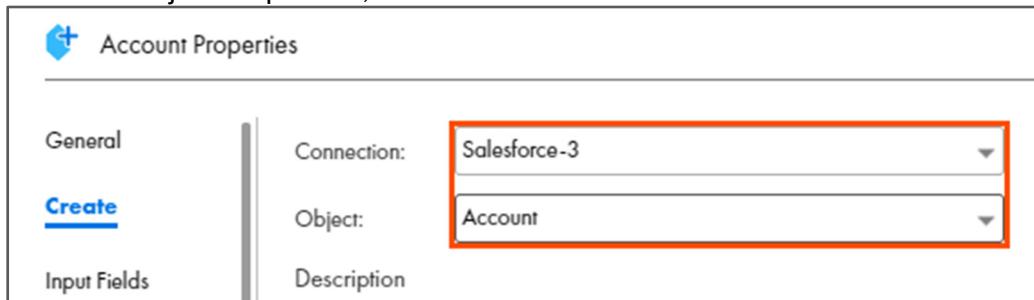
16. In the Create tab, from the Connection drop-down, select **Salesforce**.

**Important:** If you see multiple assets with the same name, check the Unique Name of your Salesforce connection. Navigate to **Explore > your\_project\_folder (for example, S\_XX\_FirstName) > Salesforce** connection. Click the Ellipsis icon and from the Properties Detail page of the connection, check the unique name as shown in the screenshot. The unique name and the asset name can be different. So, when you search for the asset, it will display the unique name.



| Basic               |   |
|---------------------|---|
| Unique Name:        | Salesforce-3                                  |
| Location:           | S_XX_FirstName                                |
| Publication Status: | <input checked="" type="checkbox"/> Published |
| Published On:       | 2020-08-06 20:03                              |

17. From the Object drop-down, select **Account**.



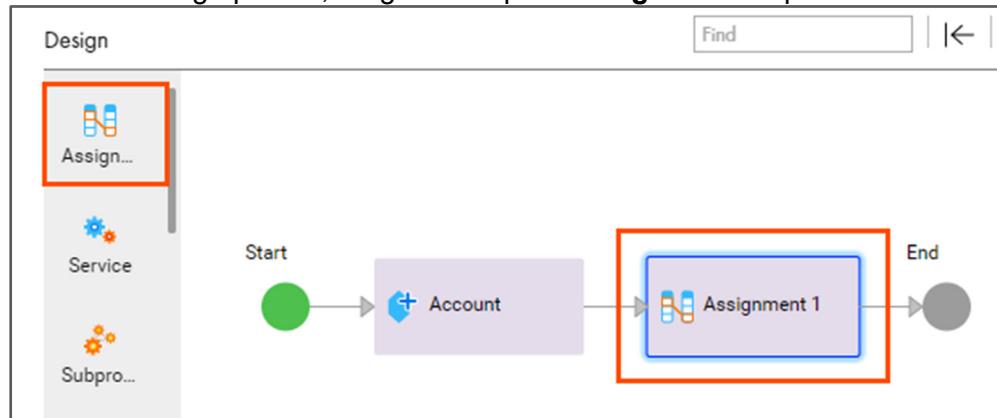
| Account Properties |                          |
|--------------------|--------------------------|
| General            | Connection: Salesforce-3 |
| <b>Create</b>      | Object: Account          |
| Input Fields       | Description              |

18. In the Input Fields tab, for the Account Name field, enter **S-XX-Test**.

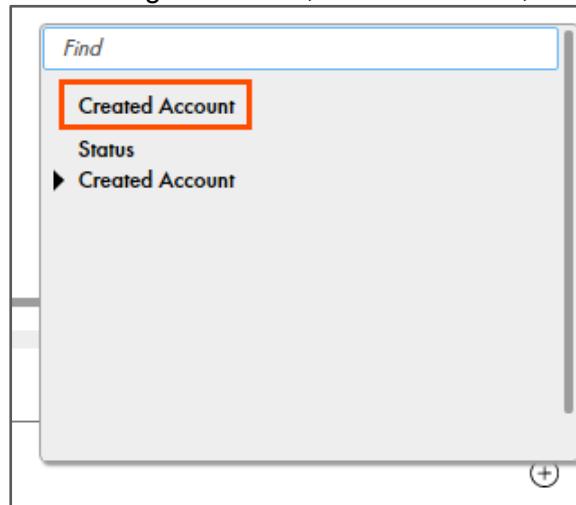


| Input Fields (1) |                                     |  |
|------------------|-------------------------------------|--|
| Name             | Required                            | Value                                    |
| Account Name     | <input checked="" type="checkbox"/> | Content <input type="button" value="x"/> |

19. From the Design palette, drag and drop an **Assignment** step before the End step.



20. In the Assignments tab, click **Add Field**, and select **Created Account**.

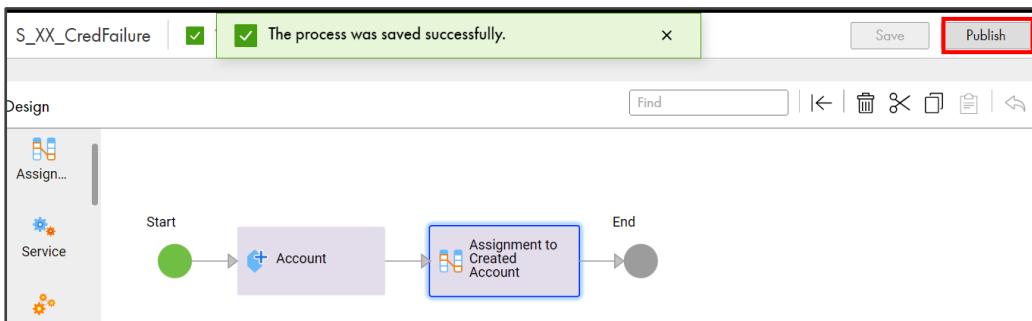


21. From the Assigned Using drop-down, select **Field** and from the From drop-down, select **Created Account > Account ID**.

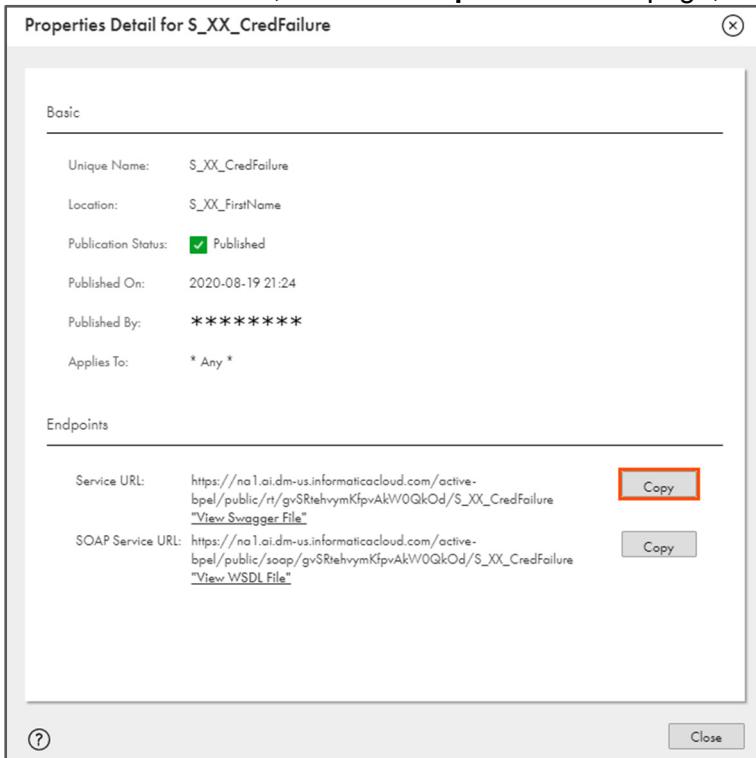


| Field           | Assigned Using | From                         |
|-----------------|----------------|------------------------------|
| Created Account | Field          | Created Account > Account ID |

**22. Save and Publish the process.**



**23. In the Actions menu, from the Properties Detail page, copy the Service URL.**



**24. In a web browser, enter the copied Service URL.**



**25. Observe that you get the output error as shown below:**



**Note:** To catch the faults, you must enable Fault Handling.

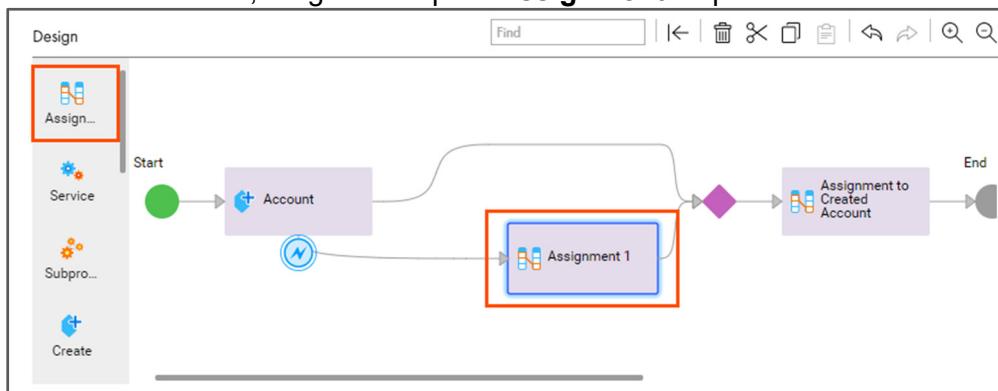
**26. Navigate back to the CAI console and close the Properties Detail window.**

**27. In the **S\_XX\_CredFailures** process, select the **Create (Account)** step.**

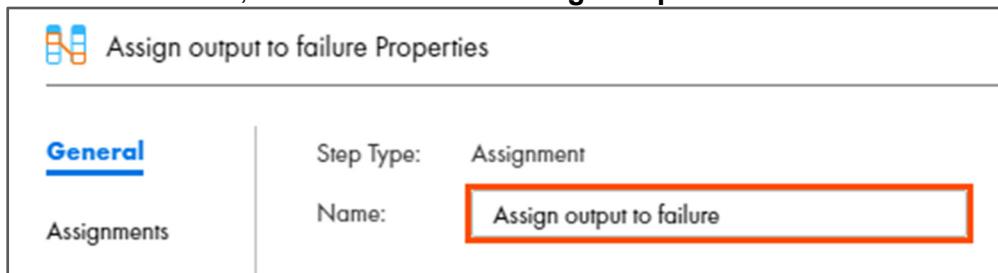
28. From the Properties Pane, select **Fault Handling** and select the **Catch faults** checkbox.



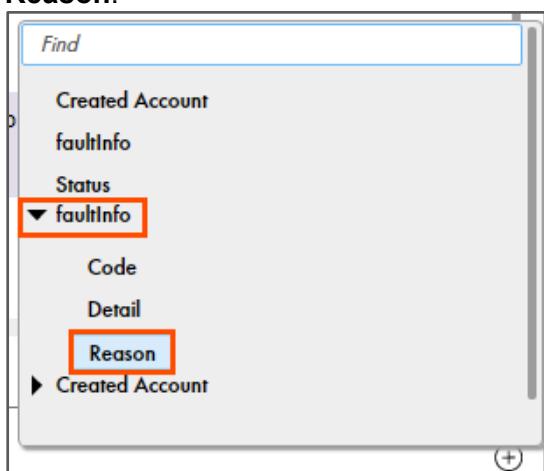
29. In the Fault branch, drag and drop an **Assignment** step.



30. In the General tab, enter the Name as **Assign output to failure**.



31. In the Assignments tab, click plus (+) icon and from the **faultInfo** drop-down, select **Reason**.



32. In the From field, enter **Authentication Failure: Check the Salesforce credentials.**

| Field              | Assigned Using | From                        |
|--------------------|----------------|-----------------------------|
| faultInfo > Reason | Content        | Authentication Failure: Che |

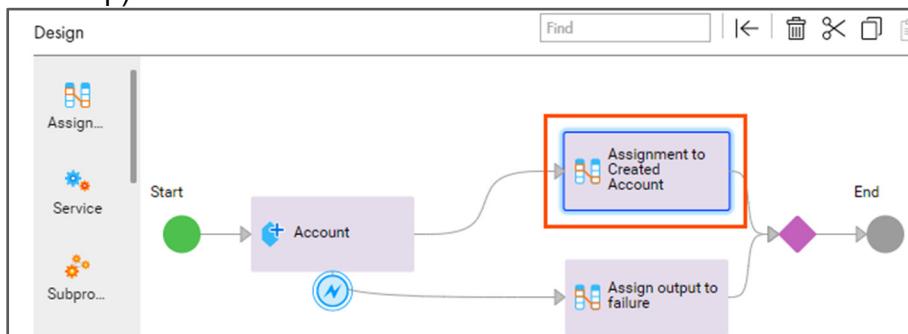
33. To add the Status field, click **Add Field**, and select **Status**.

34. From the Assigned Using drop-down, select **Field**

35. From the From drop-down, select **faultInfo > Reason**.

| Field              | Assigned Using | From                        |
|--------------------|----------------|-----------------------------|
| faultInfo > Reason | Content        | Authentication Failure: Che |
| Status             | Field          | faultInfo > Reason          |

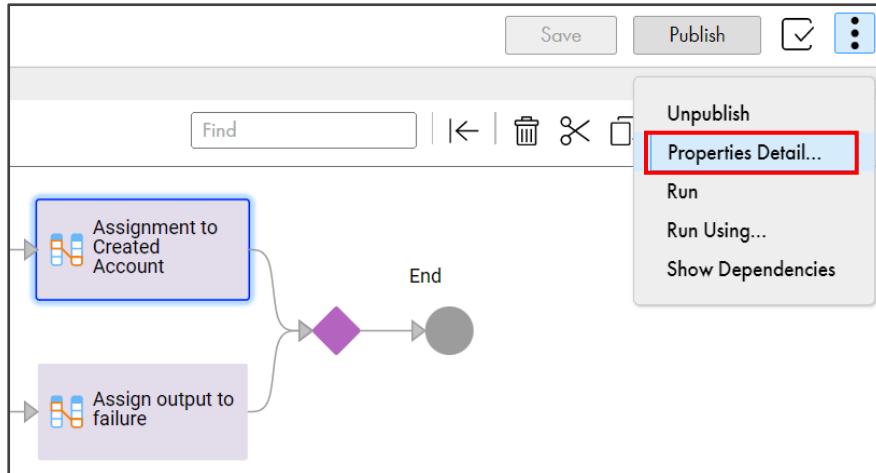
36. Drag and drop the **Assignment to Created Account** step (that is present before the End step) into the no-fault branch.



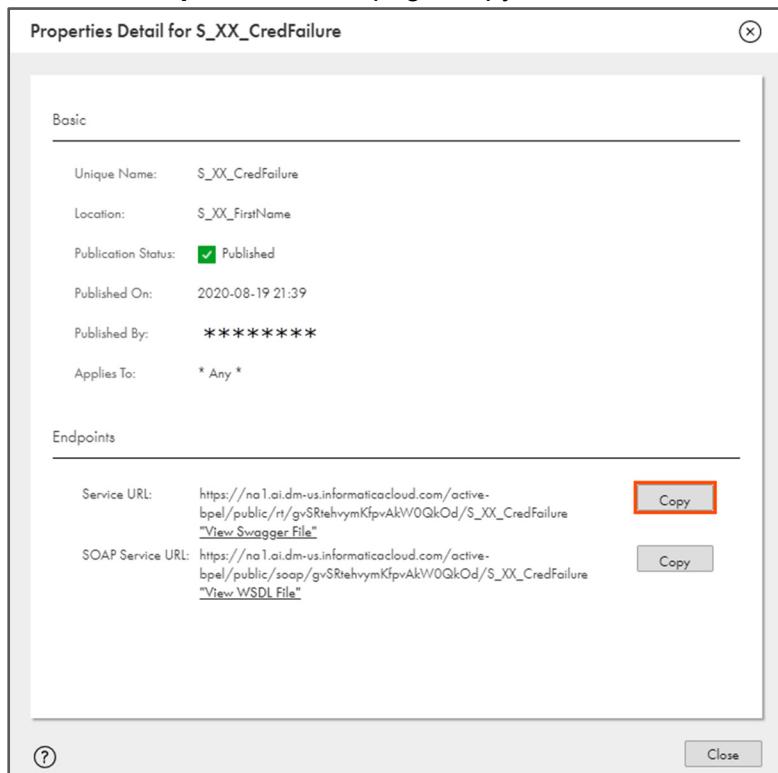
37. **Save and Publish** the process.

The process was saved successfully.

38. From the Action menu, select **Properties Detail....**



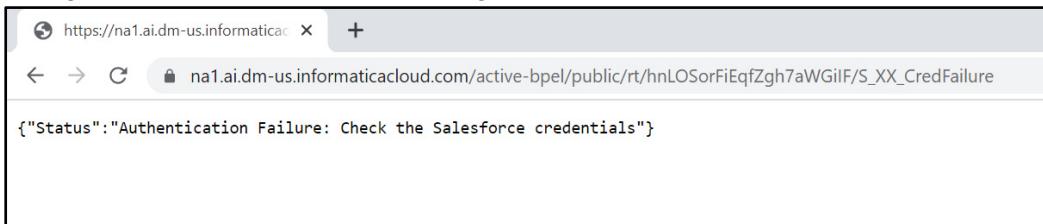
39. From the **Properties Detail** page, copy **Service URL**.



40. In a web browser, enter the copied Service URL.



You get the customized error message as shown below:



A screenshot of a web browser window. The address bar shows the URL: https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRiEqfZgh7aWGif/S\_XX\_CredFailure. The main content area of the browser displays the following JSON object:  

```
{"Status": "Authentication Failure: Check the Salesforce credentials"}
```

41. **Close** the browser tab.
- 

*This concludes the lab.*

# Module 12: Introduction to Guides Designer

## Lab 12-1: Create a Simple Guide

### Overview:

A Guide is a set of screens that helps a user to implement a business logic. It prompts the users to enter, review, or confirm data. Guides run within mobile apps or on traditional platforms such as a PC or a Mac.

In this lab, you will create a simple calculator guide and run it through the interface. You will specify the operations to be performed on the specified inputs and obtain an output based on the selected operation.

### Objectives:

- Create a Guide
- Run the Guide through Interface

### Duration:

20 Minutes

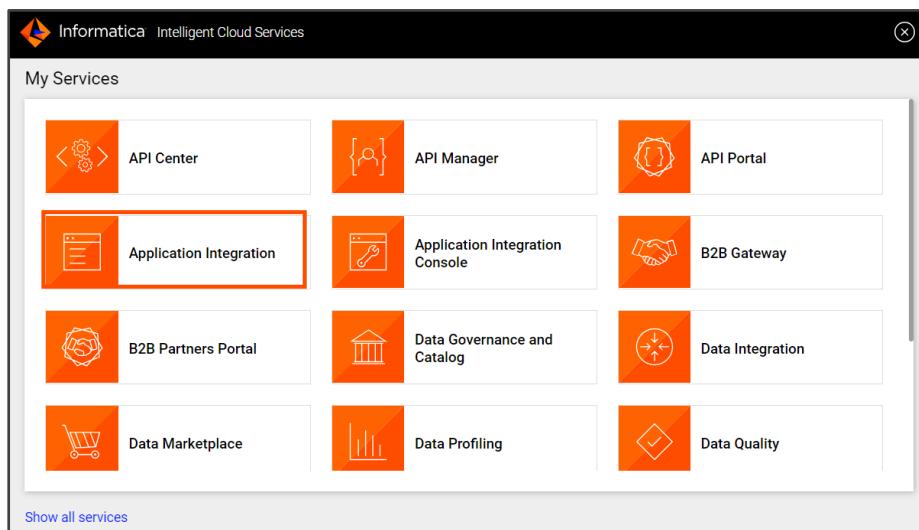
---

### Tasks

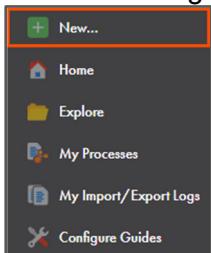
**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

### Create a New Guide

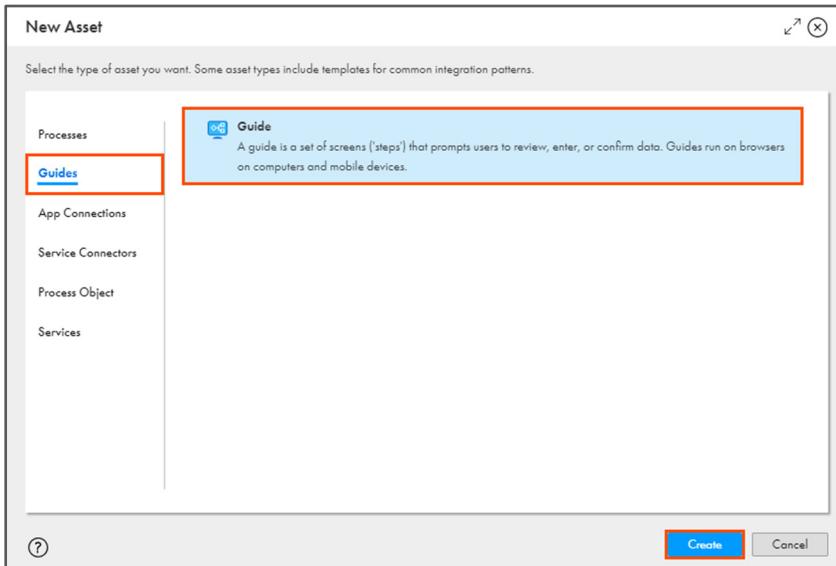
1. Log in to IICS and access the **Application Integration** service.



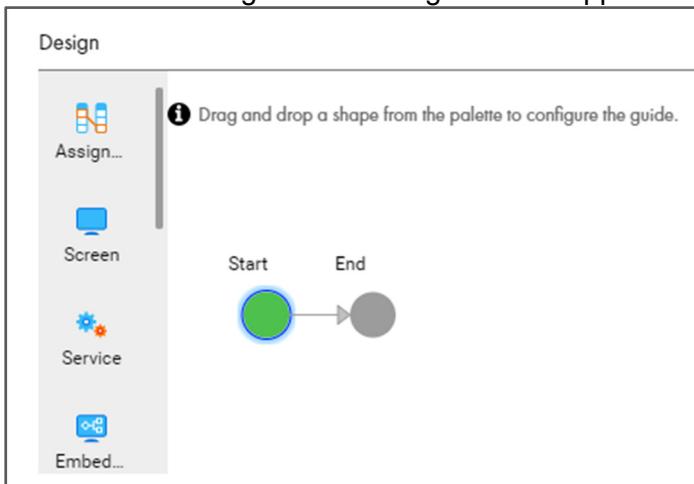
2. From the Navigation pane, click **New**.



3. In the New Asset window, from the **Guides** tab, retain the **Guide** option, and click **Create**.

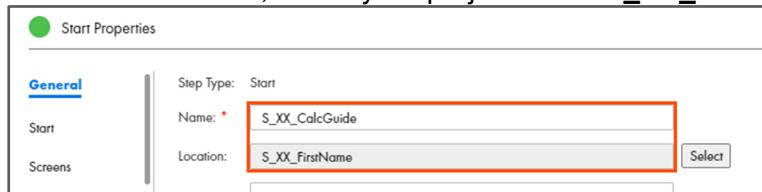


**Note:** The following Guides Design canvas appears with the Start and End steps.

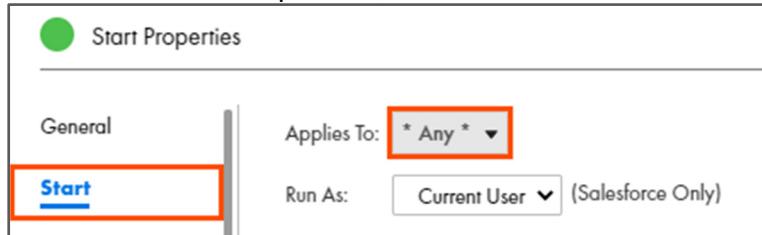


4. From the canvas, select the **Start** step.  
 5. In the General tab, enter the name as **S\_XX\_CalcGuide**.

6. In the Location field, select your project folder **S\_XX\_FirstName**.

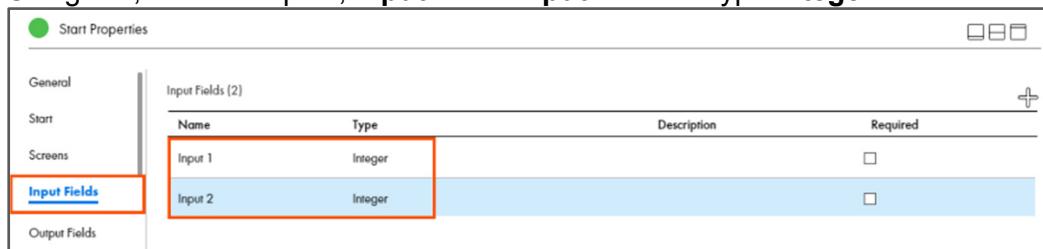


7. From the Start properties pane, select **Start**.  
 8. From the Applies To drop-down, select **Any**.  
 9. From the Run As drop-down, retain **Current User**.



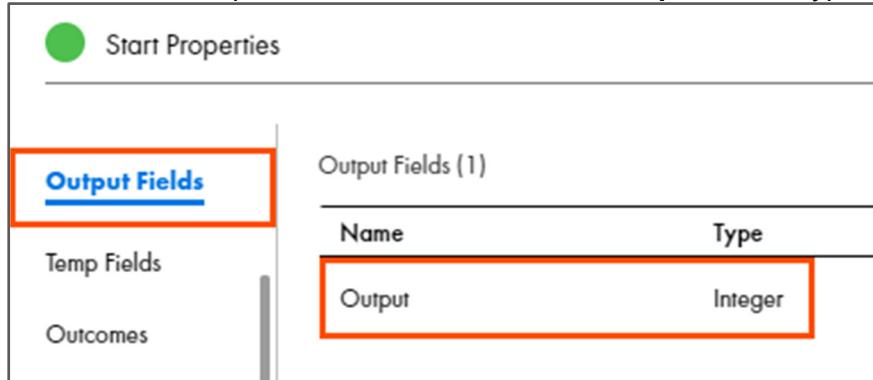
10. To configure the input fields, select the **Input Fields** tab.

11. Using , add two inputs, **Input 1** and **Input 2** of the Type **Integer**.

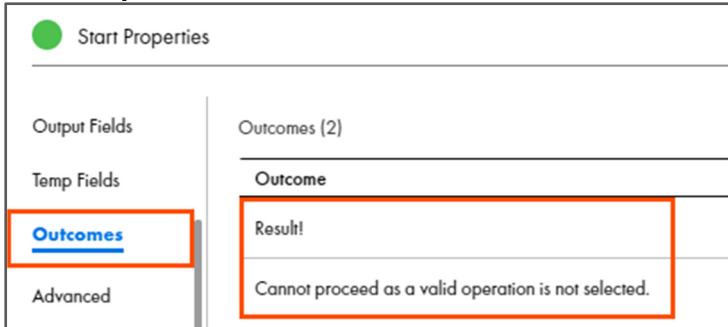


12. From the properties pane, select **Output Fields**.

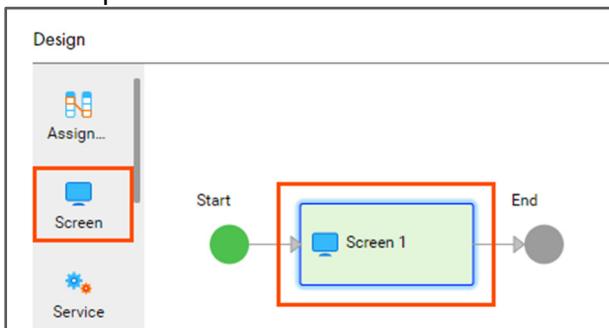
13. To add a new output, click  and add a field **Output** of the Type **Integer**.



14. In the Outcomes tab, click  and add two outcomes: **Result!** and **Cannot proceed as a valid operation is not selected.**



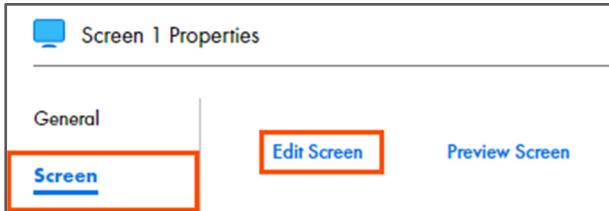
15. From the Design palette, drag and drop a **Screen** step on the link between Start and End steps.



16. To configure the step, select the **Screen** step.

17. From the properties pane, select **Screen**.

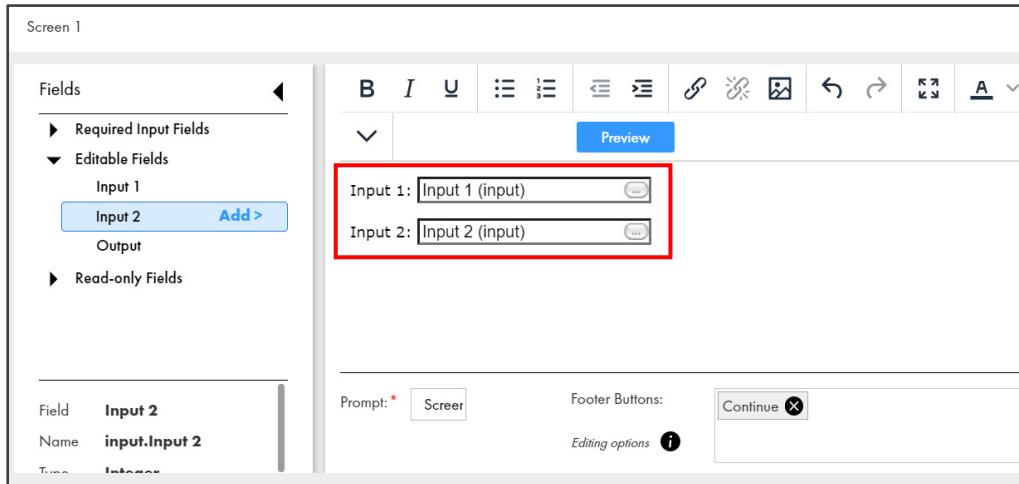
18. Select **Edit Screen**.



19. In the screen editor, from the Fields section, expand the **Editable Fields** option.

20. Select **Input 1** and click **Add**.

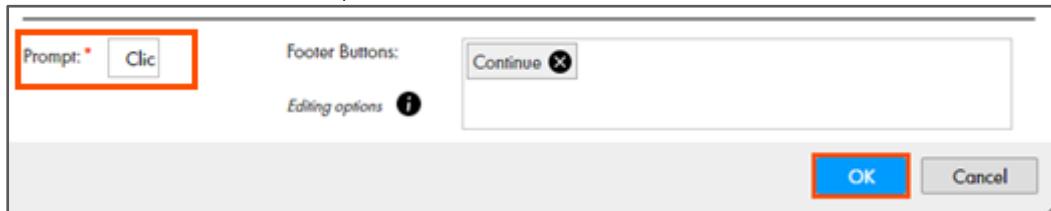
21. Similarly, add **Input 2** and configure the screen as shown in the screenshot:



**Note:** To bring Input 2 in the next line, place the cursor after Input 1 (input) box and press **Enter**. Add a : between the field name and the field. The screen editor looks as shown in the screenshot above.

22. In the Prompt field, enter **Click Continue to proceed**.

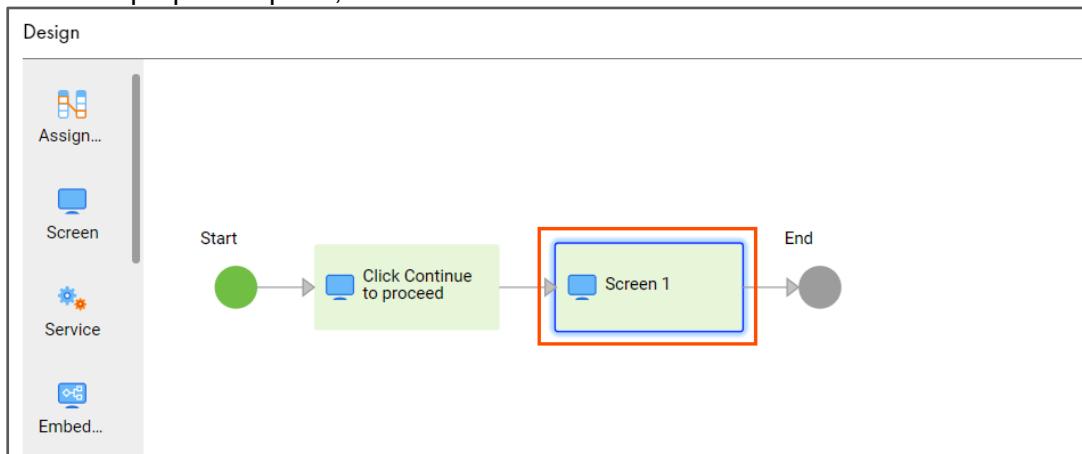
23. In the Footer Buttons field, retain the **Continue** button and click **OK**.



**Note:** If you notice multiple Continue buttons, remove them by clicking the  icon.

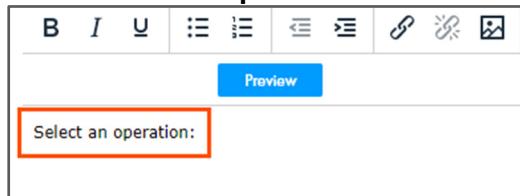
24. Add another Screen step.

25. From the properties pane, select **Screen**.



26. Select **Edit Screen**.

27. Enter **Select an operation** as shown in the screenshot:



28. In the Prompt field, enter **Select any one of the following**:

29. In the Footer Buttons field, remove the **Continue** button.

30. To add a button, in the Footer Buttons field, enter **Addition**, and press **enter**.

31. Similarly, add **Subtraction**, **Multiplication**, **Division**, and **None of these** buttons and click **OK**.

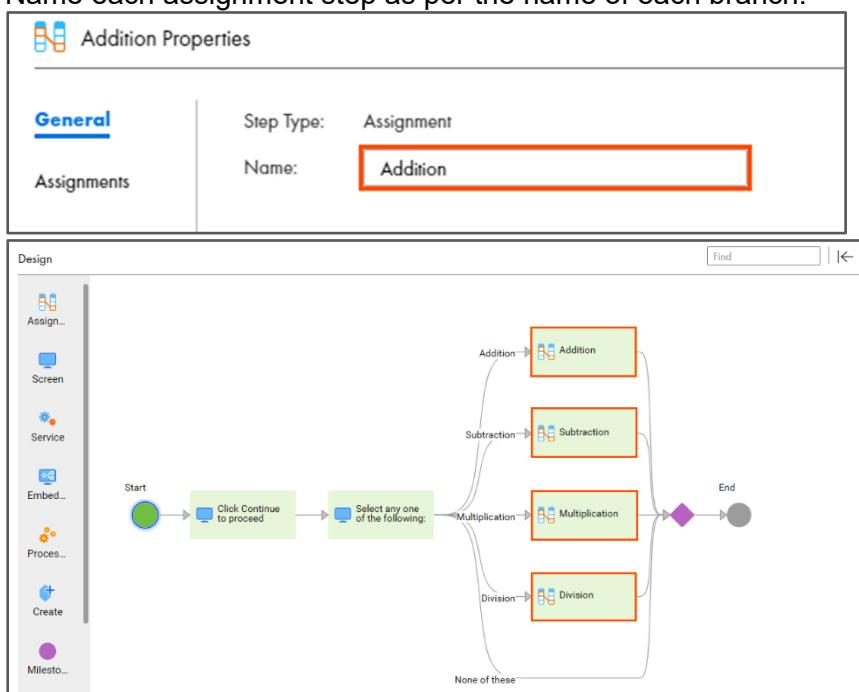


**Notes:** If the Continue button still appears, go back to the Edit Screen step, remove the Continue button, and click **OK**. Also, observe that the Guide creates a branch for each button.

32. For each branch, except the None of these branches, drag and drop an **Assignment** step.

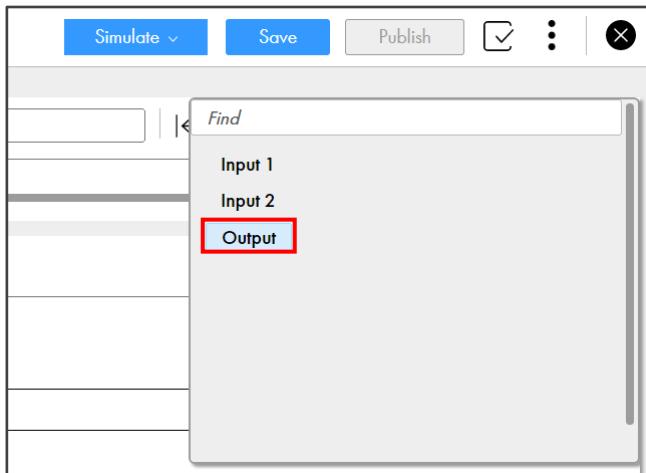


33. Name each assignment step as per the name of each branch.



34. Select **Addition** assignment step.

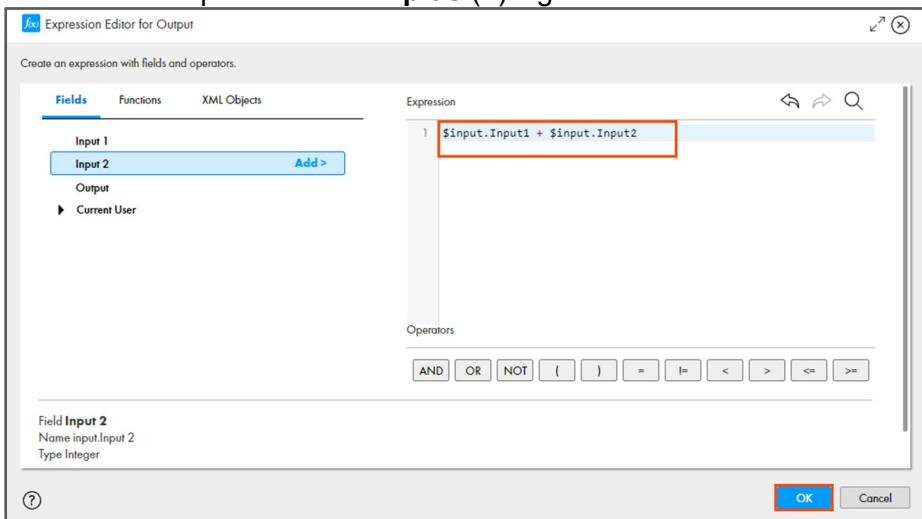
35. In the Assignments tab, click the plus '+' icon and select **Output**.



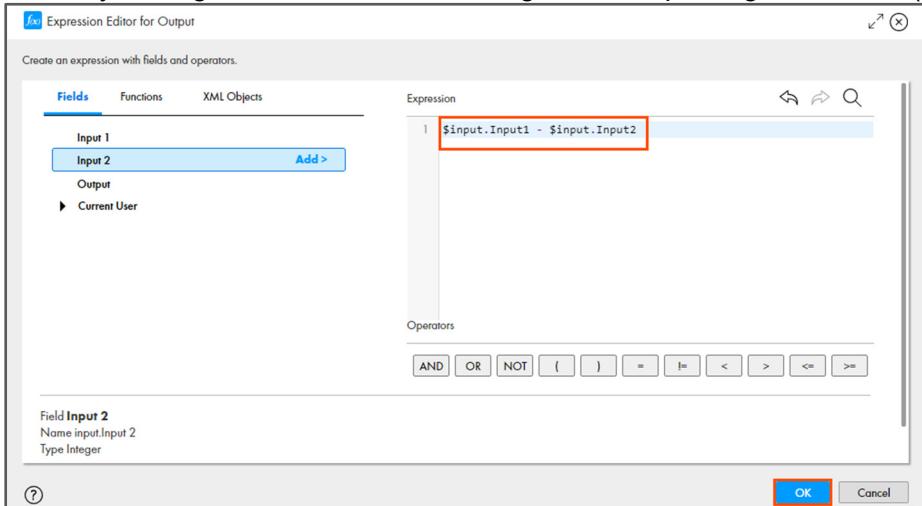
36. From the Assigned Using field, select **Formula** and from the From field, select **f(x)**.

| Field  | Assigned Using | From |
|--------|----------------|------|
| Output | Formula        | f(x) |

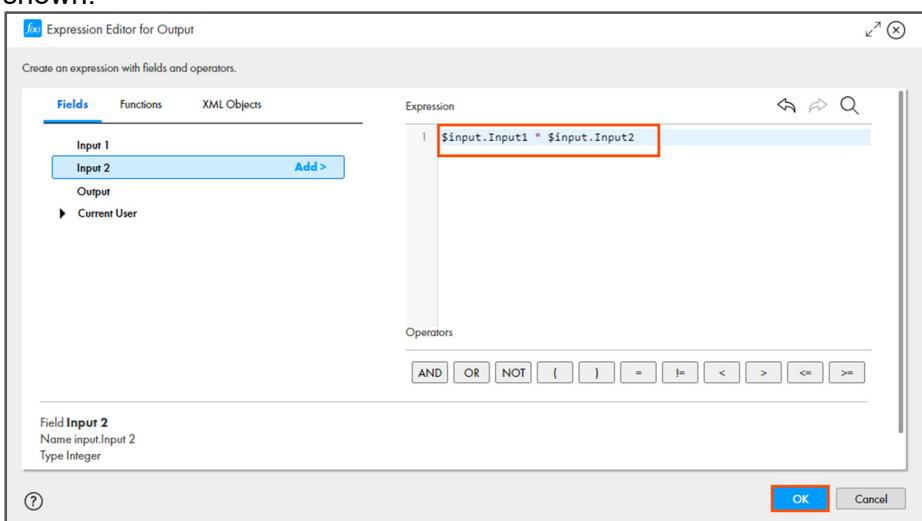
37. Add both the inputs and add a **plus (+)** sign between them as shown and click **OK**.



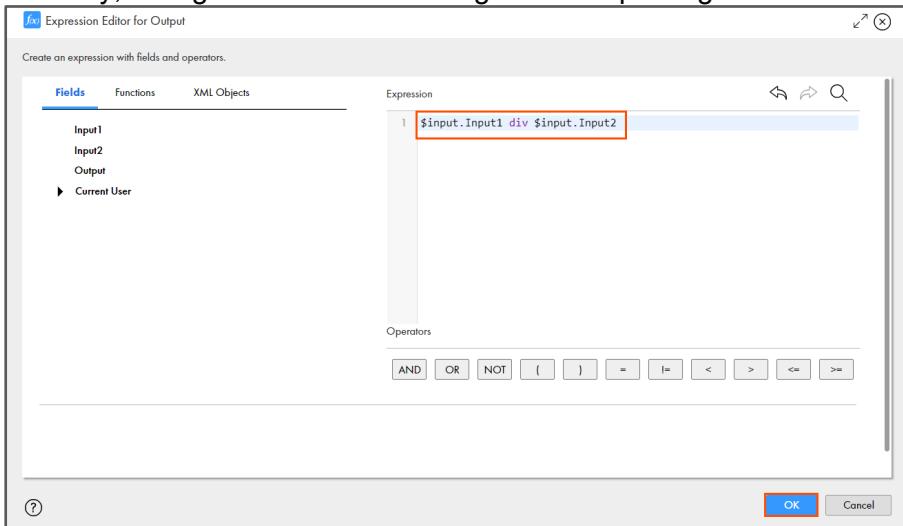
38. Similarly, configure the Subtraction Assignment step using the **minus (-)** sign as shown:



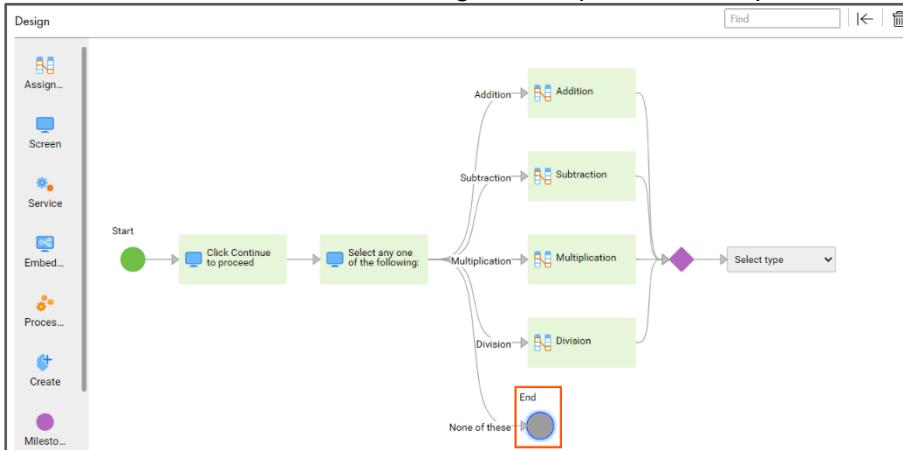
39. Similarly, configure the Multiplication Assignment step using the **asterisk (\*)** sign as shown:



40. Similarly, configure the Division Assignment step using **div** as shown:

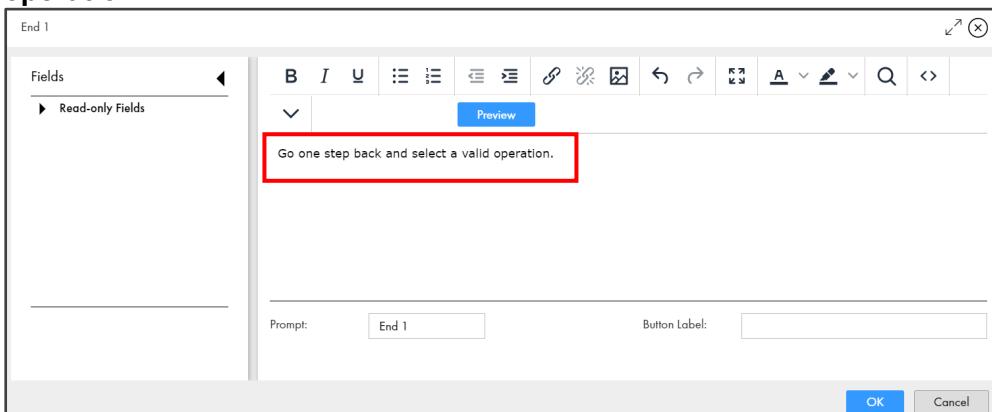


41. In the None of these branches, drag and drop the **End** step.



42. In the End tab, from the Outcome drop-down, select **Cannot proceed as a valid operation is not selected**.

43. Select **Edit Screen** and configure the screen as **Go one step back and select a valid operation**.

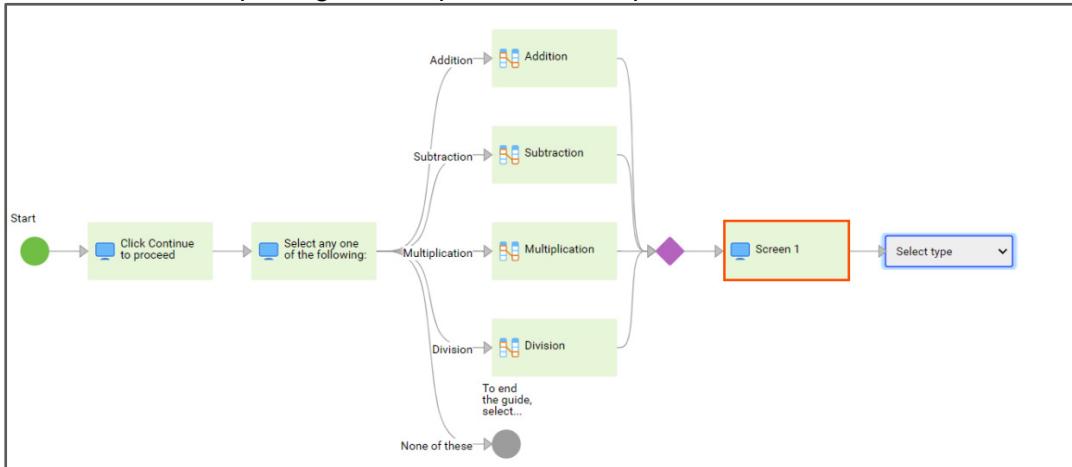


44. In the Prompt field, enter **To end the guide, select Done**.

45. In the Button Label field, enter **Done** and click **OK**.

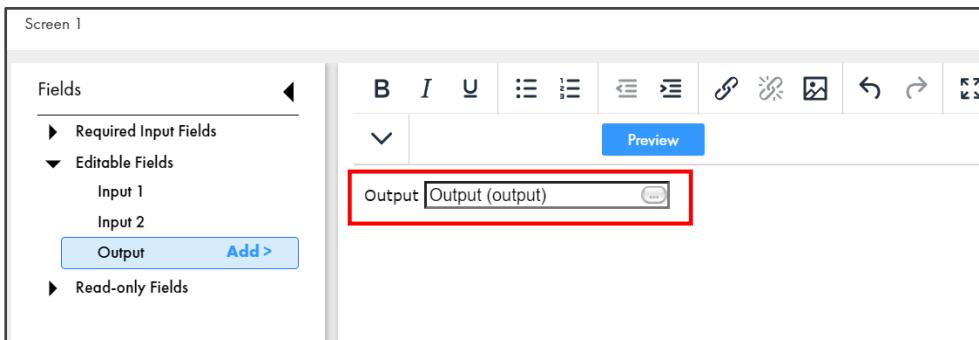


46. Before the last step, drag and drop a **Screen** step.



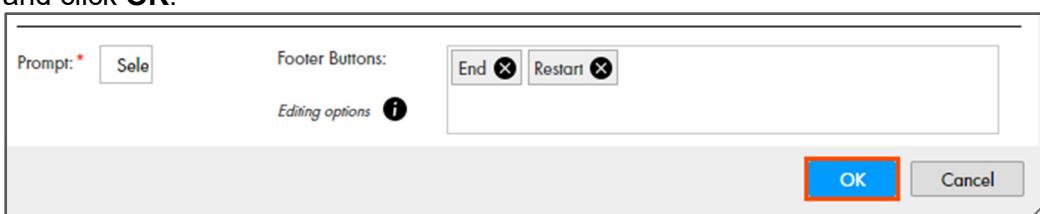
47. In the Screen tab, select **Edit Screen**.

48. From the Editable fields drop-down, select **Output**, click **Add**, and configure the screen as shown:

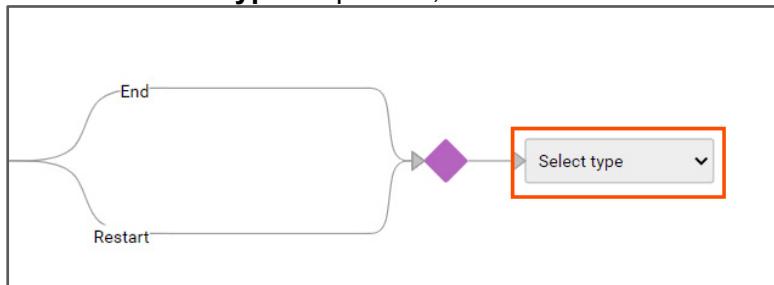


49. In the Prompt field, enter **Select an option to proceed**.

50. In the Footer Buttons field, remove **Continue** and add two buttons, **End** and **Restart** and click **OK**.



51. From the **Select type** drop-down, select **End**.



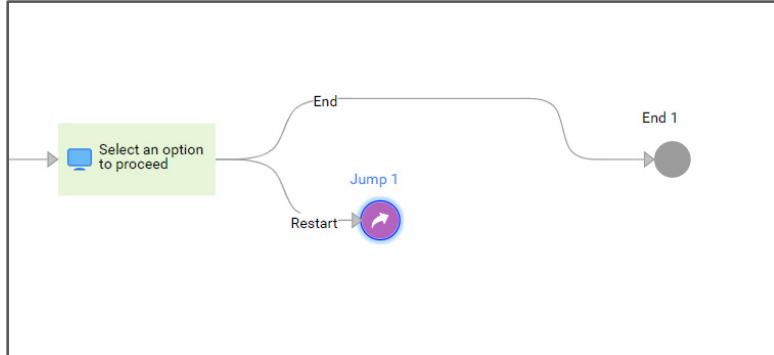
52. For the End branch, select the **End** tab.

53. From the Outcome drop-down, select **Result!**.

End 1 Properties

|            |               |  |
|------------|---------------|--|
| General    | Ending Type:  | <input type="radio"/> Milestone<br><input checked="" type="radio"/> End of Guide |
| <b>End</b> | Show Screen:  | <input checked="" type="checkbox"/> Edit Screen      Preview Screen              |
|            | Outcome:      | <b>Result!</b>   |
|            | Preprocessing | Refresh Current Object ▾   |

54. In the Restart branch, drag and drop a **Jump** step.

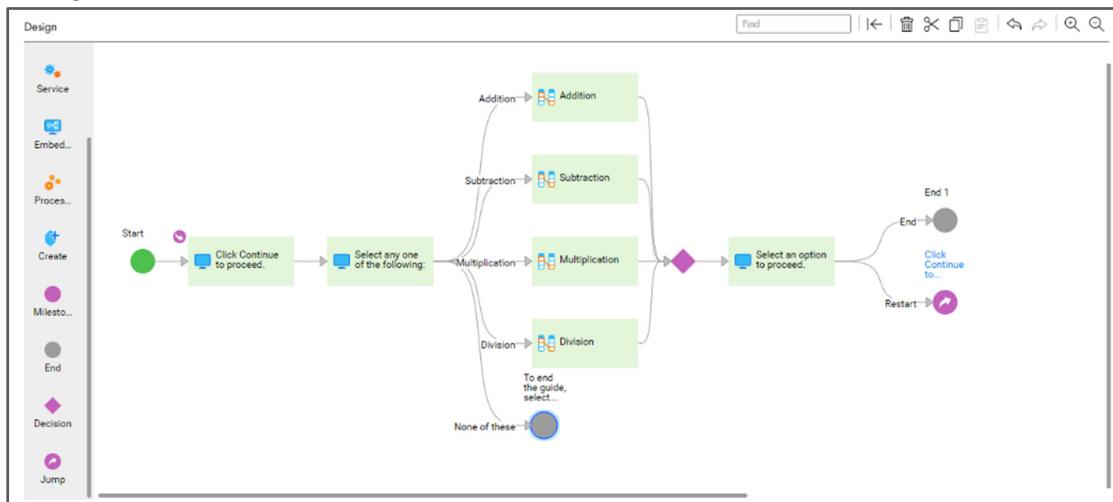


55. For the Jump step, from the To drop-down, select **Click Continue to proceed**.

Click Continue to proceed Properties

|     |                                  |
|-----|----------------------------------|
| To: | <b>Click Continue to proceed</b> |
|-----|----------------------------------|

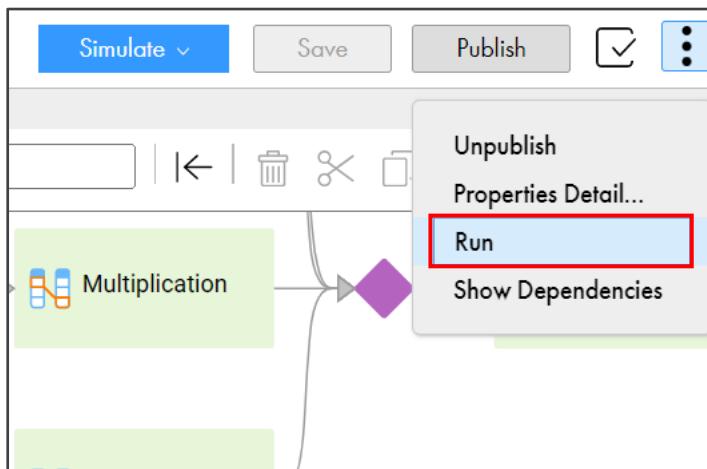
56. The Guide should look as shown in the screenshot:



57. Save and Publish the Guide.



58. To execute the Guide, from the Actions menu, click Run.



This opens the Guide in a new browser tab.

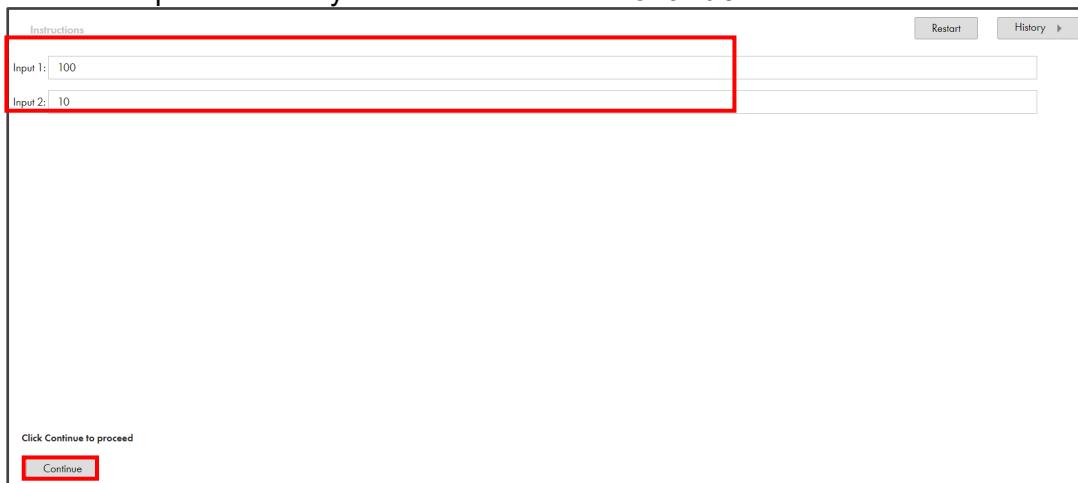
59. Enter the input values of your choice and click **Continue**.

Instructions

Input 1:   
Input 2:

Click Continue to proceed

**Continue**



60. Select an operation of your choice.

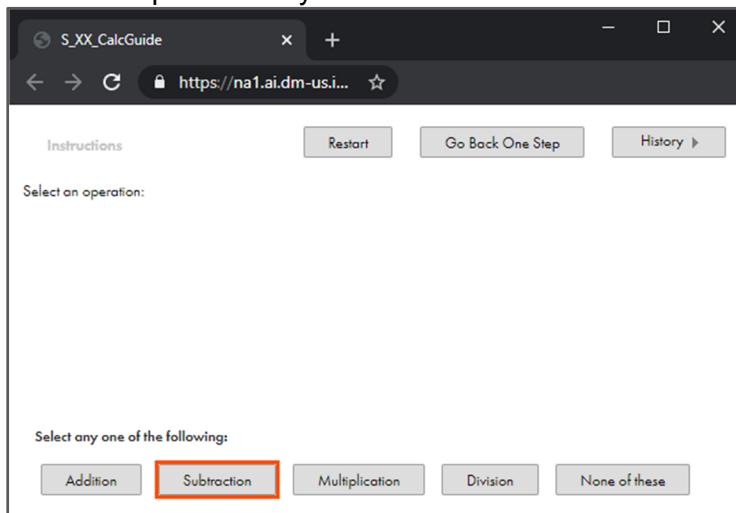
S\_XX\_CalcGuide

Instructions      Restart      Go Back One Step      History

Select an operation:

Select any one of the following:

Addition      **Subtraction**      Multiplication      Division      None of these



61. In the next screen, you will get the output.

Instructions      Restart      Go Back One Step      History

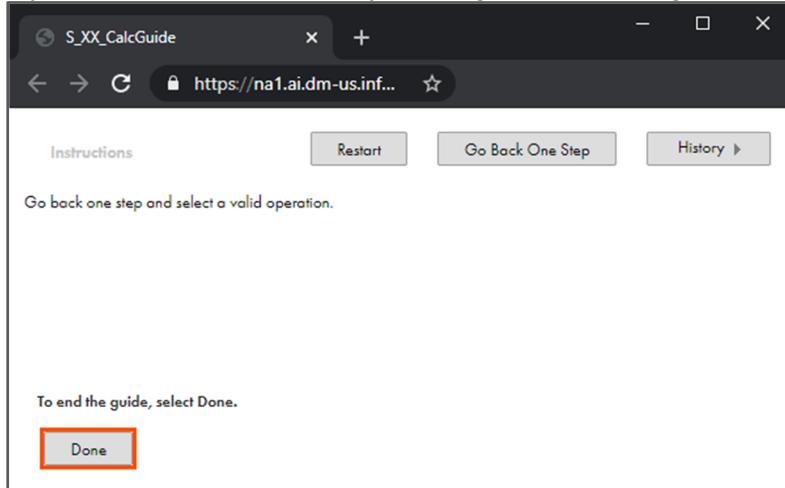
Output:

Select an option to proceed

End      Restart



62. If you select None of these, you will get the following window:



---

*This concludes the lab.*

## Module 13: Monitoring

### Lab 13-1: Create and Use URN Mapping

#### Overview:

You can assign a physical address to a universal resource name (URN). The URN is a logical address of a partner link, specified in a deployment resource. URN mappings provide a flexible and dynamic way to define target endpoint references.

By mapping a URN to a URL, you do not have to rely on invoking a statically defined endpoint address. URN mappings give you flexibility, for example, to deploy the same BPR files for testing and production environments. Also, if you specify a URL, you can replace the URL by mapping it to a different URL.

In this lab, you will create a URN to a URL mapping and then use this mapping in a process.

#### Objectives:

- Create a URN to URL Mapping
- Use the URN Mapping in a process

#### Duration:

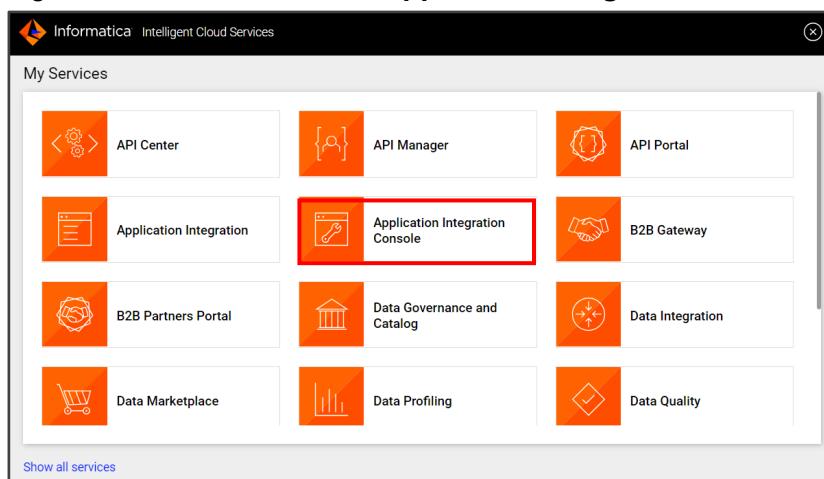
15 Minutes

---

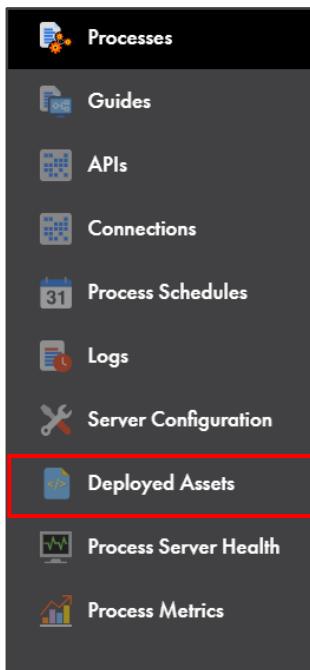
## Tasks

### Create a URN Mapping

1. Log into IICS and access the **Application Integration Console** service.



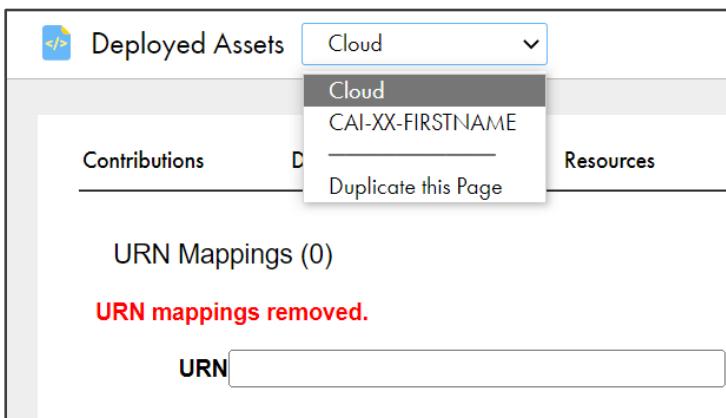
2. From the navigation pane, click **Deployed Assets**.



3. Click **URN Mappings**.

| Contributions           | Deployed Processes                            | Resources | Indexed Properties | URN Mappings        |                     |
|-------------------------|---|-----------|--------------------|---------------------|---------------------|
| Contributions [45]      |   |           |                    |                     | Updated 8:36 PM IST |
| Name                    | Contribution Path                             |           | Version            | State               | Publish Date        |
| AppConnection1          | project:/hnLOSorFiEqfZgh7aWGifF/rt.connection | 1.0       | ONLINE             | 2023/03/06 02:31 PM |                     |
| Email-Service           | project:/hnLOSorFiEqfZgh7aWGifF/rt.connection | 1.0       | ONLINE             | 2023/02/01 10:49 AM |                     |
| Get-Inventory-Detail-XX | project:/hnLOSorFiEqfZgh7aWGifF/rt.connection | 1.0       | ONLINE             | 2023/02/01 09:58 AM |                     |

4. From the Deployed Assets drop-down, select **Cloud**.



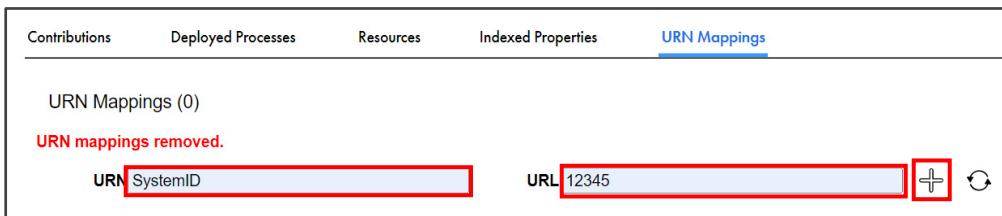
URN Mappings (0)

**URN mappings removed.**

URN

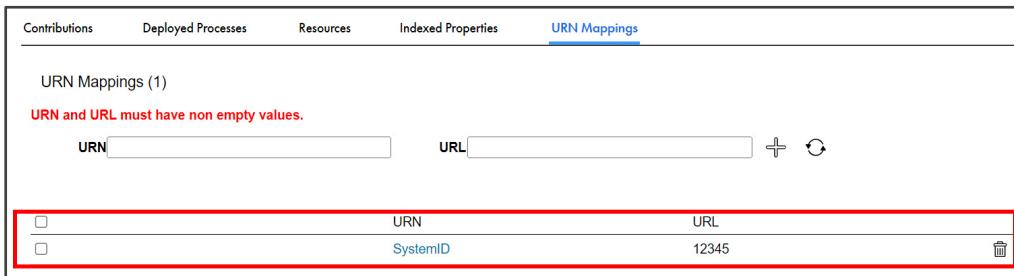
5. In the URN field, enter **SystemID**.  
 6. In the URL field, enter **12345**.

7. Click the '+' icon.



The screenshot shows the 'URN Mappings' tab selected in the top navigation bar. Below it, a message says 'URN Mappings (0)' and 'URN mappings removed.' A red box highlights the 'URN' input field containing 'SystemID' and the 'URL' input field containing '12345'. To the right of these fields are a '+' button and a refresh icon.

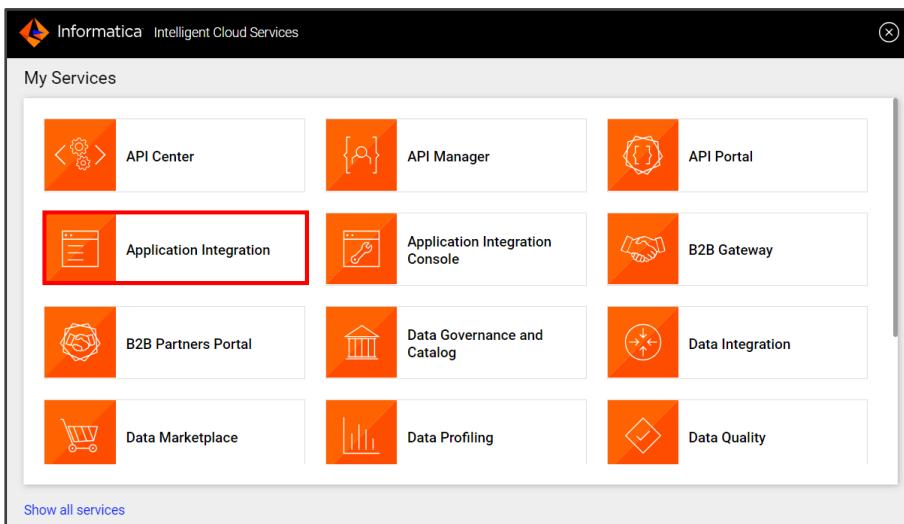
URN mapping is created.



The screenshot shows the 'URN Mappings' tab selected again. It displays 'URN Mappings (1)' and a note 'URN and URL must have non empty values.' Below this, there's a table with two rows. The first row has empty 'URN' and 'URL' fields. The second row contains 'SystemID' in the 'URN' field and '12345' in the 'URL' field. A red box highlights this second row. To the right of the table is a delete icon.

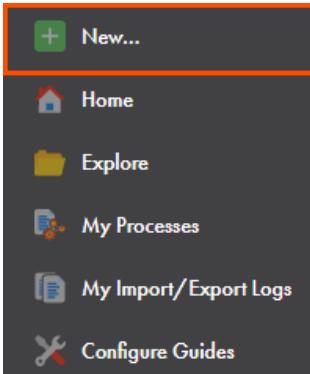
## Create a Process

8. Navigate to the Application Integration service.



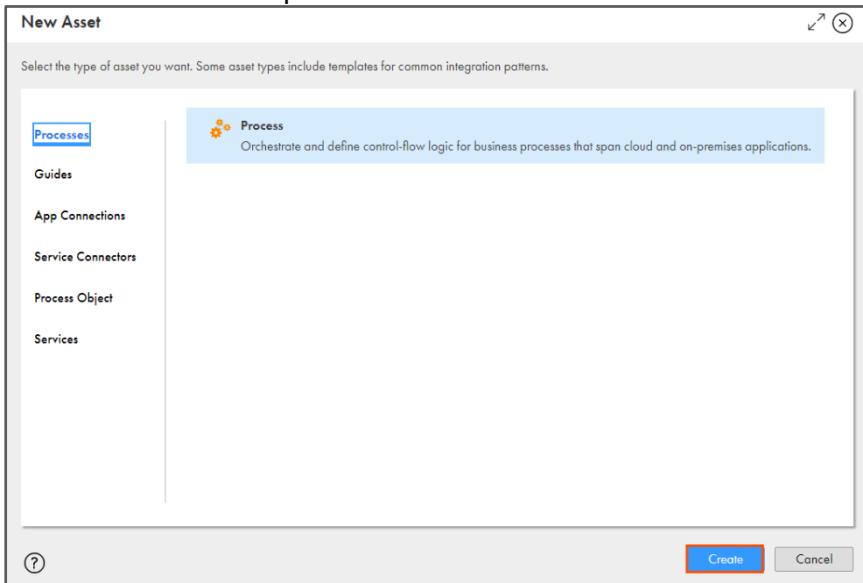
The screenshot shows the 'My Services' dashboard. It features a grid of service icons. The 'Application Integration' service is highlighted with a red box. Other services include API Center, API Manager, API Portal, Application Integration Console, B2B Gateway, B2B Partners Portal, Data Governance and Catalog, Data Integration, Data Marketplace, Data Profiling, and Data Quality. At the bottom left is a 'Show all services' link.

9. From the navigation pane, select **New**.



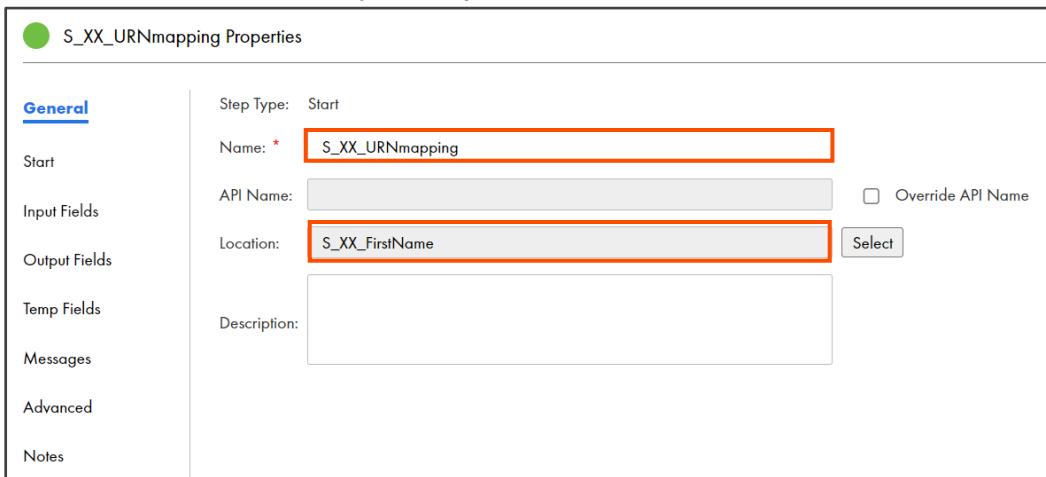
The screenshot shows the navigation pane on the left. The 'New...' option is highlighted with a red box. Other options include Home, Explore, My Processes, My Import/Export Logs, and Configure Guides.

10. Retain the **Process** option and click **Create**.



11. In the General properties tab, in the Name field, enter **S\_XX\_URNmapping**.

12. In the Location field, select your project folder **S\_XX\_FirstName**.



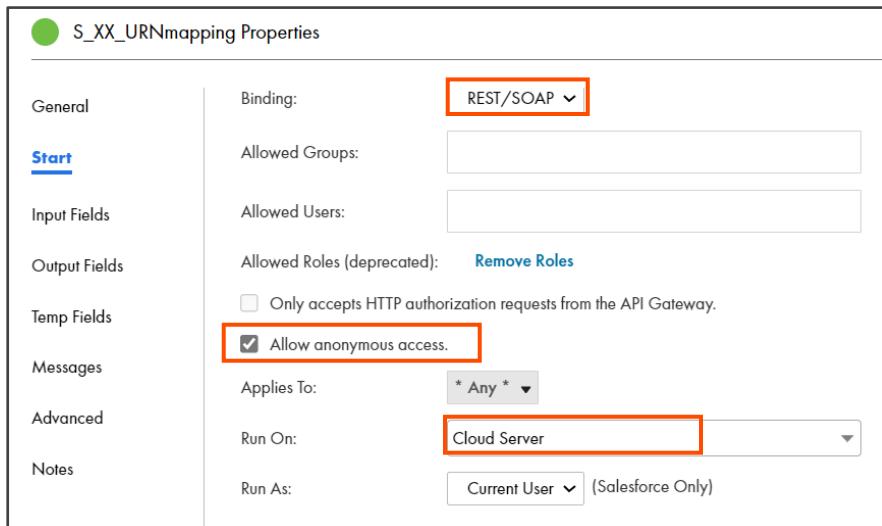
The screenshot shows the 'S\_XX\_URNmapping Properties' dialog box in the 'General' tab. On the left, there's a sidebar with tabs: General (selected and highlighted in blue), Start, Input Fields, Output Fields, Temp Fields, Messages, Advanced, and Notes. The main area contains the following fields:

- Step Type:** Start
- Name:** \* **S\_XX\_URNmapping** (highlighted with a red box)
- API Name:**   Override API Name
- Location:** **S\_XX\_FirstName** (highlighted with a red box)
- Description:**

13. In the Start tab, retain the Binding option as **REST/SOAP**.

14. Select the **Allow anonymous access** option.

15. From the Run On drop-down, retain **Cloud Server**.



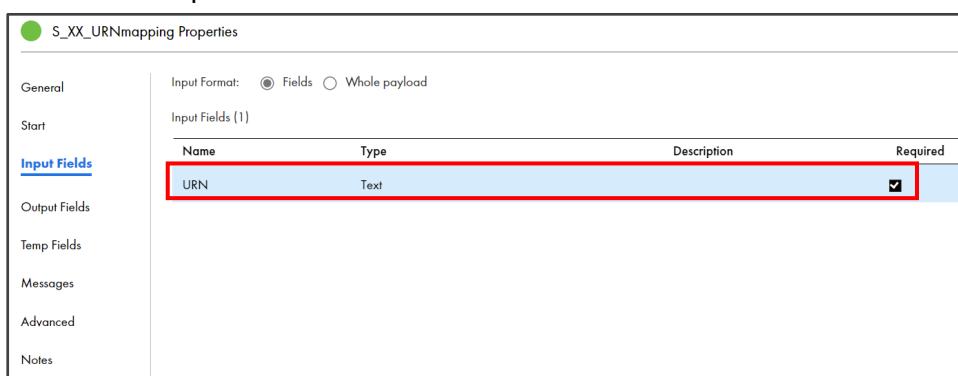
S\_XX\_URNmapping Properties

|               |   |
|---------------|---|
| General       | Binding: REST/SOAP  |
| Start         | Allowed Groups:   |
| Input Fields  | Allowed Users:  |
| Output Fields | Allowed Roles (deprecated): Remove Roles  |
| Temp Fields   | <input type="checkbox"/> Only accepts HTTP authorization requests from the API Gateway. |
| Messages      | <input checked="" type="checkbox"/> Allow anonymous access.                             |
| Advanced      | Applies To: * Any *   |
| Notes         | Run On: Cloud Server  |
|               | Run As: Current User (Salesforce Only)  |

16. In the Input Fields tab, to add a new field, click .

17. In the Name field, enter **URN** and retain the Type as **Text**.

18. Select the Required checkbox.

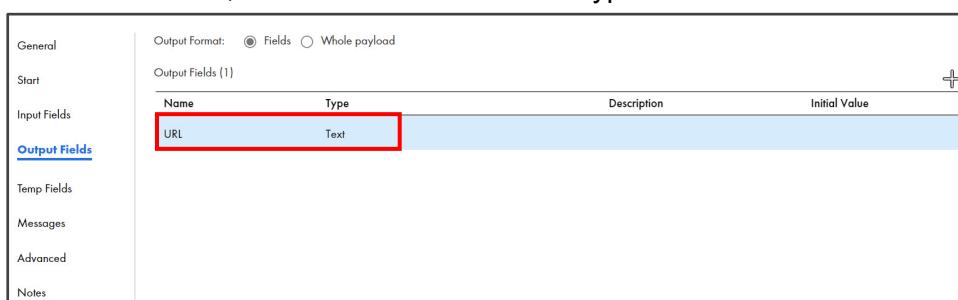


S\_XX\_URNmapping Properties

| General       | Input Format: <input checked="" type="radio"/> Fields <input type="radio"/> Whole payload  |             |                                     |             |          |     |      |  |                                     |
|---------------|--|-------------|-------------------------------------|-------------|----------|-----|------|--|-------------------------------------|
| Start         | Input Fields (1)   |             |                                     |             |          |     |      |  |                                     |
| Input Fields  | <table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Description</th> <th>Required</th> </tr> </thead> <tbody> <tr> <td>URN</td> <td>Text</td> <td></td> <td><input checked="" type="checkbox"/></td> </tr> </tbody> </table> | Name        | Type                                | Description | Required | URN | Text |  | <input checked="" type="checkbox"/> |
| Name          | Type   | Description | Required                            |             |          |     |      |  |                                     |
| URN           | Text   |             | <input checked="" type="checkbox"/> |             |          |     |      |  |                                     |
| Output Fields |  |             |                                     |             |          |     |      |  |                                     |
| Temp Fields   |  |             |                                     |             |          |     |      |  |                                     |
| Messages      |  |             |                                     |             |          |     |      |  |                                     |
| Advanced      |  |             |                                     |             |          |     |      |  |                                     |
| Notes         |  |             |                                     |             |          |     |      |  |                                     |

19. In the Output Fields tab, to add a new output field, click .

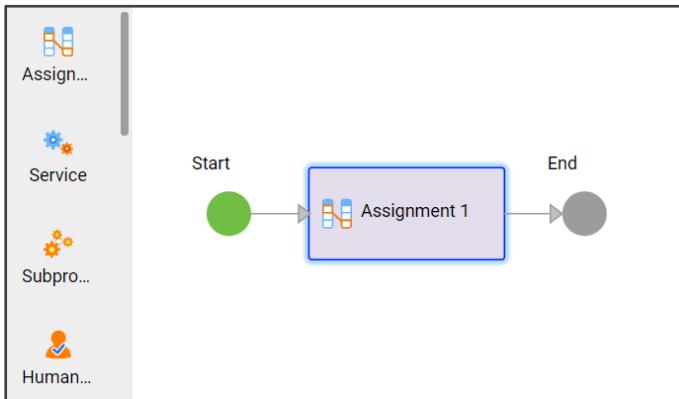
20. In the Name field, enter **URL** and retain the Type as **Text**.



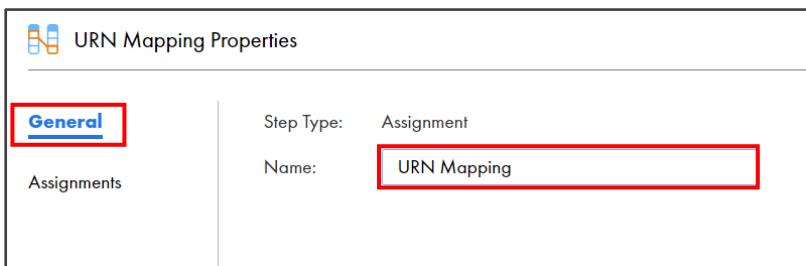
S\_XX\_URNmapping Properties

| General       | Output Format: <input checked="" type="radio"/> Fields <input type="radio"/> Whole payload   |             |               |             |               |     |      |  |  |
|---------------|--|-------------|---------------|-------------|---------------|-----|------|--|--|
| Start         | Output Fields (1)  |             |               |             |               |     |      |  |  |
| Input Fields  |  |             |               |             |               |     |      |  |  |
| Output Fields | <table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Description</th> <th>Initial Value</th> </tr> </thead> <tbody> <tr> <td>URL</td> <td>Text</td> <td></td> <td></td> </tr> </tbody> </table> | Name        | Type          | Description | Initial Value | URL | Text |  |  |
| Name          | Type   | Description | Initial Value |             |               |     |      |  |  |
| URL           | Text   |             |               |             |               |     |      |  |  |
| Temp Fields   |  |             |               |             |               |     |      |  |  |
| Messages      |  |             |               |             |               |     |      |  |  |
| Advanced      |  |             |               |             |               |     |      |  |  |
| Notes         |  |             |               |             |               |     |      |  |  |

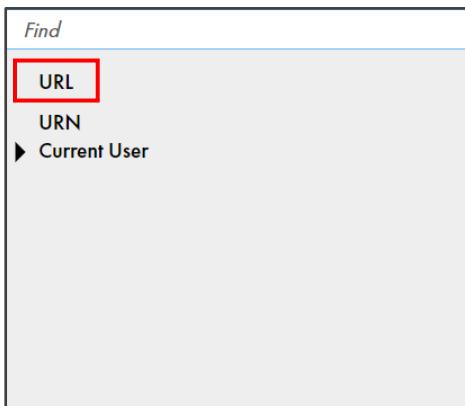
21. From the palette, drag and drop an **Assignment** step between the Start and the End step on the canvas.



22. Select the **Assignment** step.  
 23. In the General tab, in the Name field, enter **URN Mapping**.



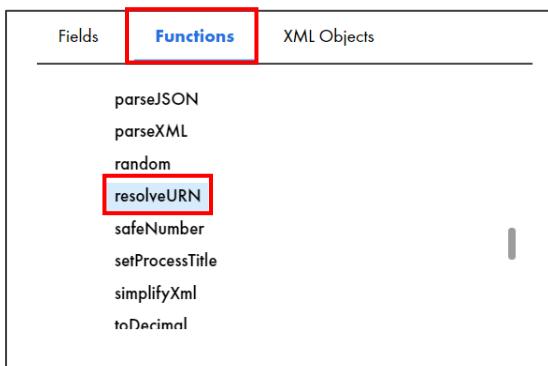
24. In the Assignments tab, click the plus (+) icon, and select **URL**.



25. From the **Assigned Using** drop-down, select **Formula**.

26. Click  to open the Expression Editor.

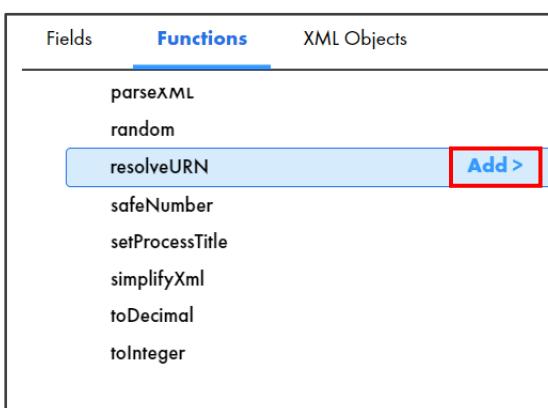
27. Click **Functions** and select **Miscellaneous > resolveURN**.



The screenshot shows a list of functions under the 'Functions' tab. The 'resolveURN' function is highlighted with a red box.

| parseJSON         |
|-------------------|
| parseXML          |
| random            |
| <b>resolveURN</b> |
| safeNumber        |
| setProcessTitle   |
| simplifyXml       |
| toDecimal         |

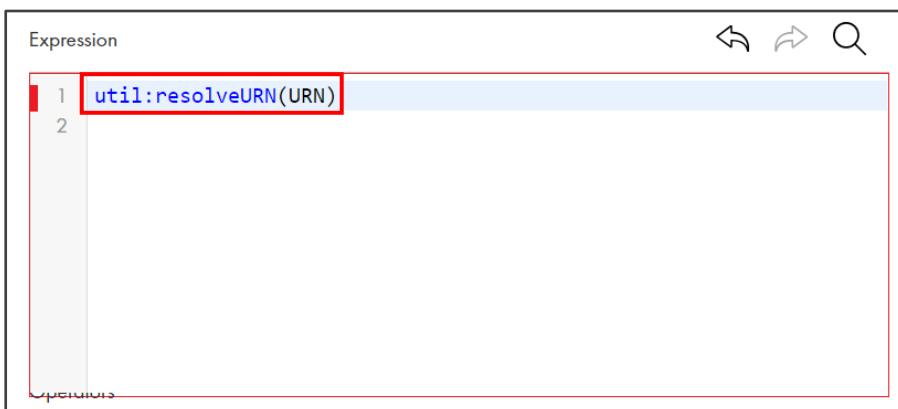
28. Click **Add >**.



The screenshot shows the 'resolveURN' function selected in the list. The 'Add >' button is highlighted with a red box.

| parseXML          |
|-------------------|
| random            |
| <b>resolveURN</b> |
| safeNumber        |
| setProcessTitle   |
| simplifyXml       |
| toDecimal         |
| toInteger         |

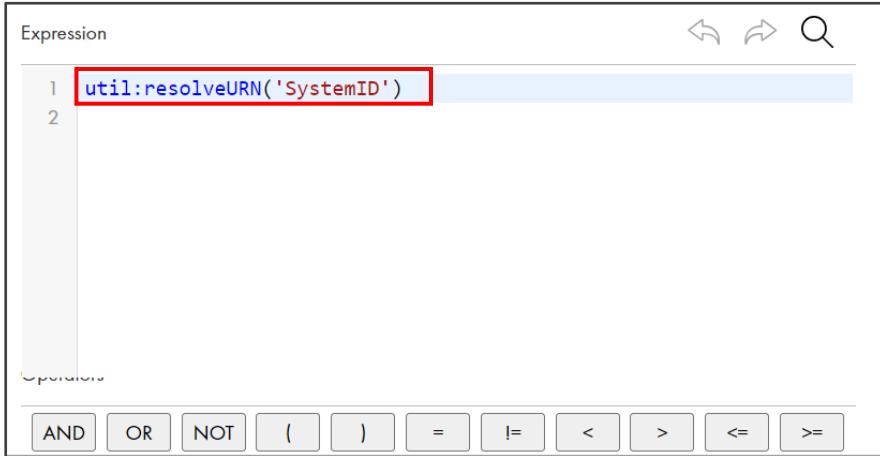
29. In the Expression section, observe the **util:resolveURN(urn)** function.



The screenshot shows the 'Expression' section with the 'util:resolveURN(urn)' function highlighted with a red box.

| util:resolveURN(urn) |
|----------------------|
| 2                    |

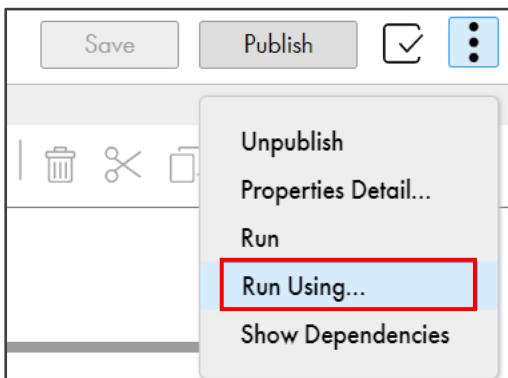
30. Replace URN with '**SystemID**' as displayed here:



```
Expression
1 util:resolveURN('SystemID')
2
Operators
AND OR NOT ( ) = != < > <= >=
```

**Note:** SystemID should be in parenthesis.

31. Click **OK**.
32. **Save and Publish** the process.
33. From the Action menu, select **Run Using....**

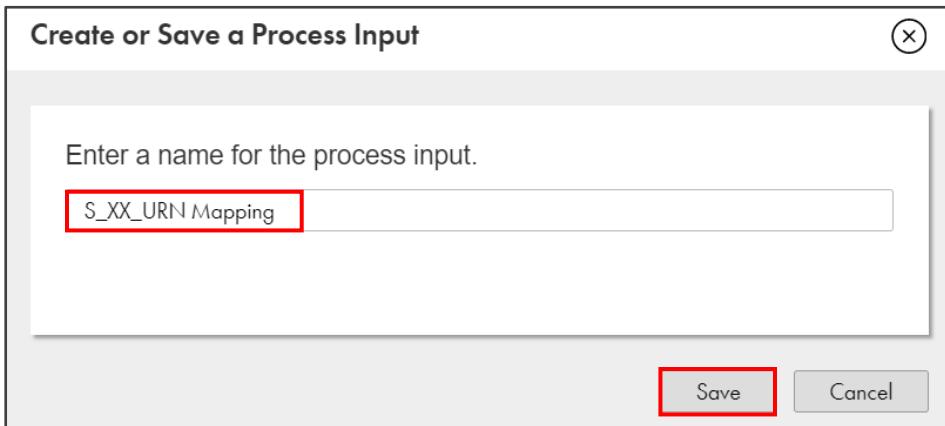


34. In the Process Input section, enter the input as **SystemID**, and click **Save**.



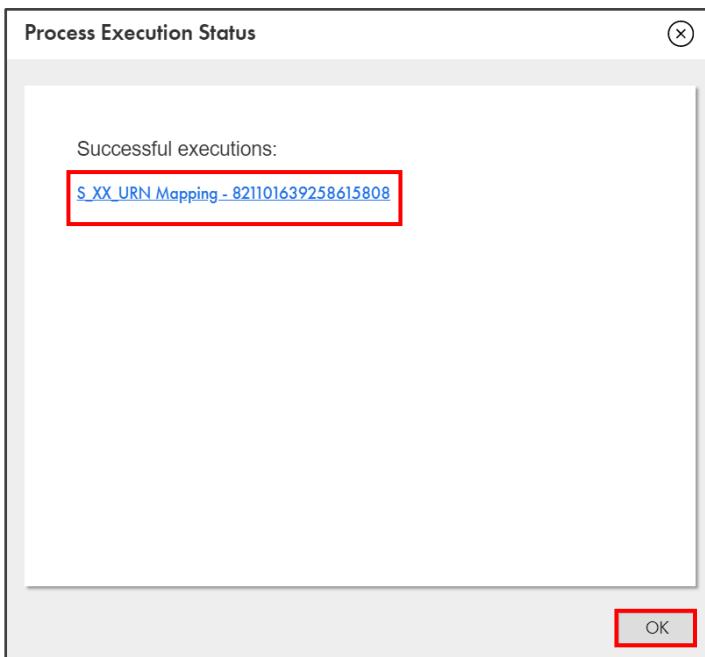
The image shows the 'Process Input' section of a process editor. It has fields for 'Process Input', 'Encoding: JSON', 'Save As...', and a 'Save' button. Below these is a text area containing a JSON object: { "URN": "SystemID" }. The string 'SystemID' is highlighted with a red box. The 'Save' button is also highlighted with a red box.

35. To save the input, enter the input name as **S\_XX\_URN Mapping** and **Save** it.

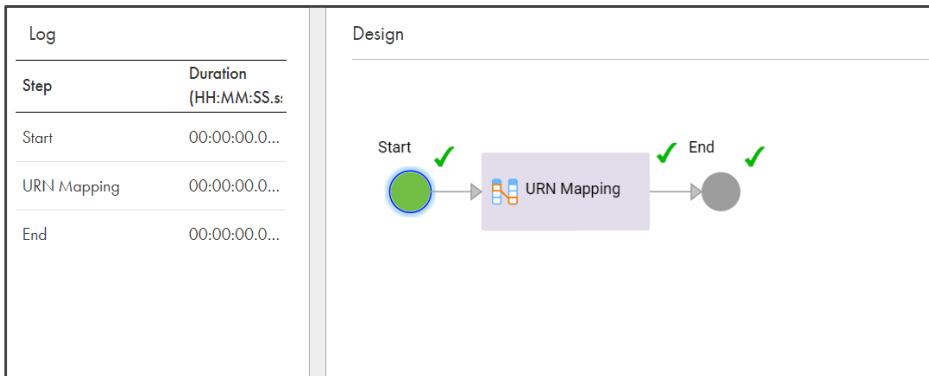


36. After the input is added and validated successfully, click **Run**.

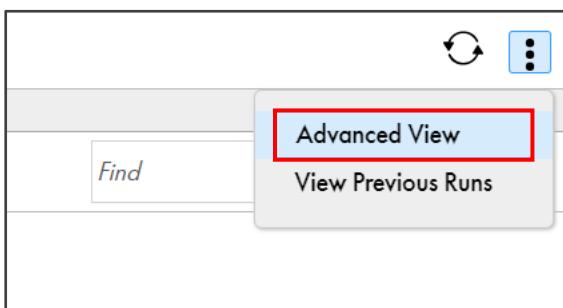
37. To check the Process Execution Status, click the **Process Execution** link, and then click **OK**.



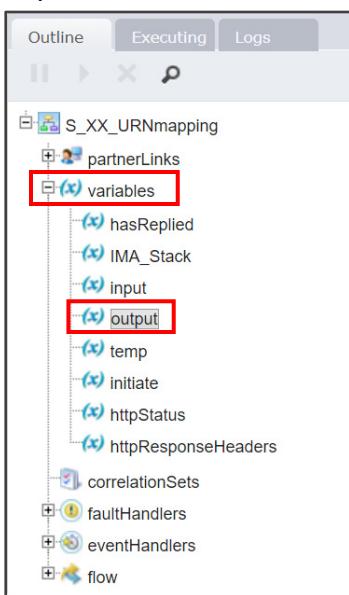
The execution details are displayed as shown below:



38. To check the output, from the **Actions** menu, select **Advanced View**.



39. Expand the **variables** section and select **output**.



40. In the Variable Instance Data section, observe the value of URL.

**(x) Variable**

|                |   |
|----------------|---|
| Name           | output  |
| Path           | /process/variables/variable[@name='output']                     |
| Element Type   | type:output   |
| Type Namespace | http://schemas.informatica.com/spi/types/2013/12/spi-interface/ |

**Variable Instance Data**

```
<aetgt:output xmlns:aetgt="http://schemas.informatica.com/spi/types/2013/12/spi-interface/" xmlns:sf="http://schemas.active-endpoints.com/appmodu
  <sf:field name="URL ">12345</sf:field>
</aetgt:output>
```

**Note:** Notice the value of URL. It is the same value which you have given while creating the URN mapping.

---

*This concludes the lab.*

## Module 13: Monitoring

### Lab 13-2: Create a New Process Schedule

#### Overview:

A Process Schedule can be created to perform routine maintenance or another recurring task. If desired, you can deploy the process and then set a schedule to run it, such as weekly or on a specific day of the month. You can set a fine-grained schedule (intraday, seconds), if desired.

In this lab, you will create a Process Schedule for an existing process.

#### Objectives:

- Create a Process Schedule for an existing process

#### Duration:

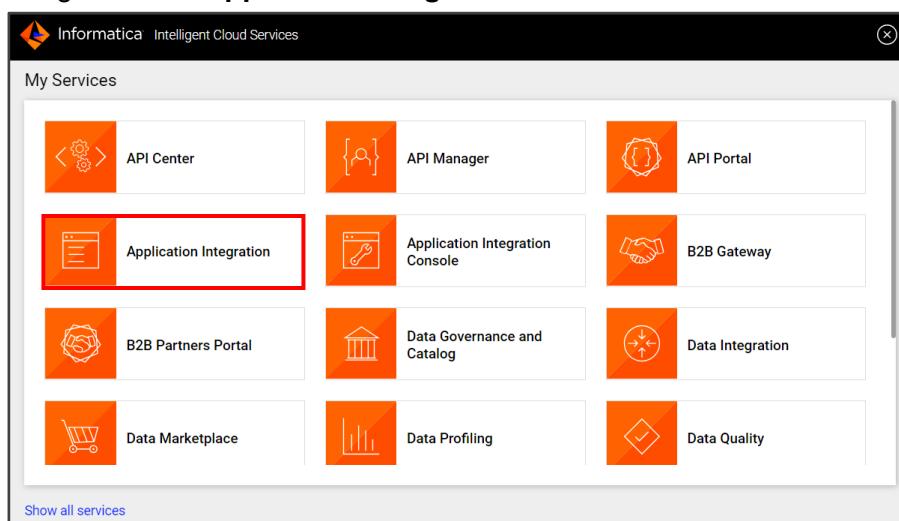
15 Minutes

---

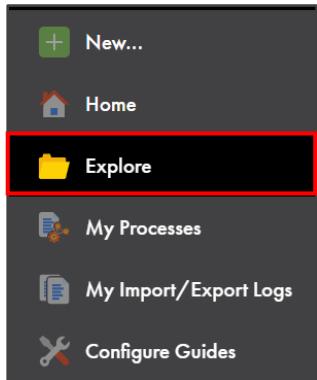
### Tasks

#### Copy the Process Input

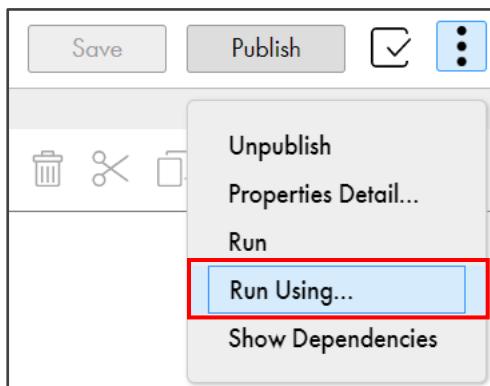
1. Navigate to the **Application Integration** service.



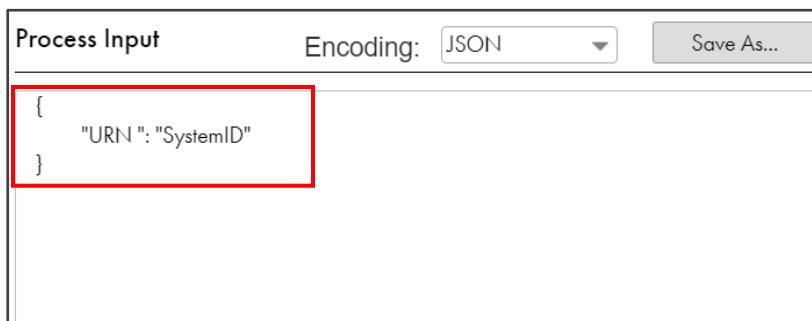
2. From the navigation pane, select **Explore**.



3. Navigate to your folder **S\_XX\_Firstname**.
4. Locate and open the **S\_XX\_URNmapping** process.  
**Note:** Use the Search box to find the process.
5. From the Actions menu, select **Run Using...**



6. From the Process Input field, copy the code and paste it in a notepad file.

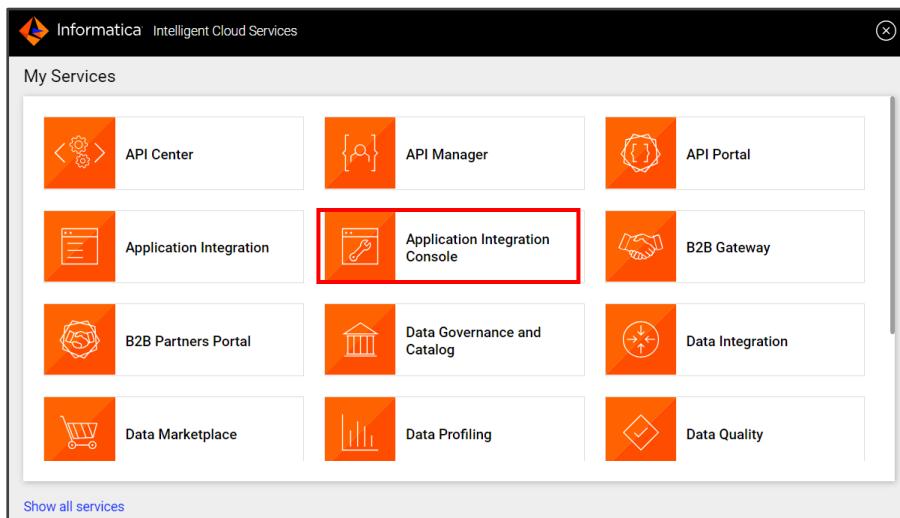


```
{  
    "URN ": "SystemID"  
}
```

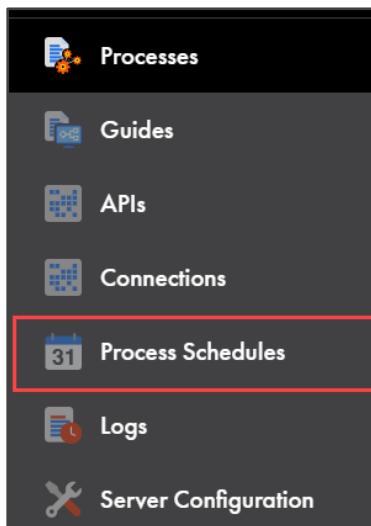
## Create a Process Schedule

In this step, we will schedule an execution for the S\_XX\_URNmapping process.

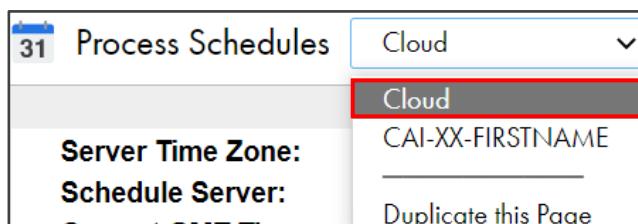
- Access the Application Integration Console service.



- From the navigation pane, click **Process Schedules**.



- From the Process Schedules drop-down, select **Cloud**.



10. Check the **Current Server Time** and make a note of it.

|                             |  |
|-----------------------------|--|
| <b>Server Time Zone:</b>    | GMT (Z)  |
| <b>Schedule Server:</b>     | application-integration-obm-3.application-integration-obm.cai-prod-usw1.svc.cluster.local:8443 |
| <b>Current GMT Time:</b>    | 2023-07-18 19:40:00  |
| <b>Current Server Time:</b> | 2023-07-18 19:40:00  |

Process Schedules [0]

No Scheduled Processes found matching the selection

**Note:** You will use this time while creating the Process Schedule.

11. Click the '+' icon to add a new Process Schedule.

|                             |  |
|-----------------------------|--|
| <b>Server Time Zone:</b>    | GMT (Z)  |
| <b>Schedule Server:</b>     | application-integration-obm-3.application-integration-obm.cai-prod-usw1.svc.cluster.local:8443 |
| <b>Current GMT Time:</b>    | 2023-07-18 19:40:00  |
| <b>Current Server Time:</b> | 2023-07-18 19:40:00  |

Process Schedules [0] Updated 1:10 AM IST / 7:40 PM GMT   

No Scheduled Processes found matching the selection

12. In the Execution Schedule window, in the Name field, enter **S\_XX\_SchedulePrint**.

**Execution Schedule**

|                          |  |
|--------------------------|--|
| <b>Name:</b>             | <input style="border: 2px solid orange; width: 400px; height: 20px;" type="text" value="S_XX_SchedulePrint"/>                              |
| <b>Frequency:</b>        | Once    |
| <b>Start Date (GMT):</b> | 2023-07-18  (yyyy-mm-dd)                                |
| <b>Start Time (GMT):</b> | 00:00:00  (hh:mm:ss)                                    |
| <b>Run Options:</b>      | <input checked="" type="radio"/> Skip if running<br><input type="radio"/> Terminate if running<br><input type="radio"/> Create new process |

13. From the Frequency drop-down, select **Once**.

14. In the Start Date field, select today's date.

15. In the Start Time field, enter 15 minutes delayed time from the Current Server Time. **For example**, if the Current Server Time is 19:40:00, then Start Time should be 19:55:00.

**Note:** This is the time when you want to schedule the process execution.

16. From Run Options, select **Terminate if running**.

**Execution Schedule**

|                   |  |
|-------------------|--|
| Name:             | S_XX_SchedulePrint   |
| Frequency:        | Once   |
| Start Date (GMT): | 2023-07-18 (yyyy-mm-dd)  |
| Start Time (GMT): | 19:55:00 (hh:mm:ss)  |
| Run Options:      | <input type="radio"/> Skip if running<br><input checked="" type="radio"/> Terminate if running<br><input type="radio"/> Create new process |

17. From Service Type, select **IPD Process**.

18. From the Select Service drop-down, select **S\_XX\_URNmapping**.

**Execution Schedule**

|                   |  |
|-------------------|--|
| Name:             | S_XX_SchedulePrint   |
| Frequency:        | Once   |
| Start Date (GMT): | 2023-07-18 (yyyy-mm-dd)  |
| Start Time (GMT): | 19:55:00 (hh:mm:ss)  |
| Run Options:      | <input type="radio"/> Skip if running<br><input checked="" type="radio"/> Terminate if running<br><input type="radio"/> Create new process |
| Service Type:     | <input type="radio"/> BPEL Service<br><input checked="" type="radio"/> IPD Process   |
| Select Service:   | S_XX_URNmapping  |

19. In the Post Body field, enter the following code:

```
{
  "URN ": "SystemID"
}
```

|                 |                                     |
|-----------------|-------------------------------------|
| Select Service: | S_XX_URNmapping                     |
| POST Body:      | <pre>{   "URN ": "SystemID" }</pre> |

20. Click **OK**.

The Process Schedule is created.

| Process Schedules [1]    |         |                    |                 |             |                      | Updated 1:17 AM IST / 7:46 PM GMT |  |  |  |
|--------------------------|---------|--------------------|-----------------|-------------|----------------------|-----------------------------------|--|--|--|
| <input type="checkbox"/> | Enabled | Name               | Service Name    | Last Status | Last Execution (GMT) | Next Execution (GMT)              |  |  |  |
| <input type="checkbox"/> |         | S_XX_SchedulePrint | S_XX_URNmapping | -           | -                    | 2023-07-18 19:55:00               |  |  |  |

1 - 1 of 1      1 of 1      20

**Note:** Observe the **Last Status** field. It is blank.

21. Wait until the **Next Execution** time matches with the **Current Server Time**.

| Process Schedules [1]    |         |                    |                 |             |                      | Updated 1:25 AM IST / 7:55 PM GMT |  |  |  |
|--------------------------|---------|--------------------|-----------------|-------------|----------------------|-----------------------------------|--|--|--|
| <input type="checkbox"/> | Enabled | Name               | Service Name    | Last Status | Last Execution (GMT) | Next Execution (GMT)              |  |  |  |
| <input type="checkbox"/> |         | S_XX_SchedulePrint | S_XX_URNmapping | Completed   | 2023-07-18 19:55:00  | -                                 |  |  |  |

1 - 1 of 1      1 of 1      20

**Note:** Keep clicking the refresh icon to check the Current Server Time.

22. Observe the Last Status field. It should be **Completed**.

| Process Schedules [1]    |         |                    |                 |             |                      | Updated 1:25 AM IST / 7:55 PM GMT |  |  |  |
|--------------------------|---------|--------------------|-----------------|-------------|----------------------|-----------------------------------|--|--|--|
| <input type="checkbox"/> | Enabled | Name               | Service Name    | Last Status | Last Execution (GMT) | Next Execution (GMT)              |  |  |  |
| <input type="checkbox"/> |         | S_XX_SchedulePrint | S_XX_URNmapping | Completed   | 2023-07-18 19:55:00  | -                                 |  |  |  |

1 - 1 of 1      1 of 1      20

The process is executed successfully.

*This concludes the lab.*

## Module 13: Monitoring

### Lab 13-3: Start and Stop an Event Source in a Listener-based Connection

#### **Overview:**

You can start and stop event sources in listener-based connections that run on a Secure Agent or a Secure Agent group. With this option, you do not have to unpublish and republish the connection to stop and start the event-based processes.

In Lab 7-6: Create a Kafka Connection, you have created two processes:

**S-XX-KafkaWriteDemo** and **S-XX-KafkaReadDemo**. The S-XX-KafkaWriteDemo process writes a message to the topic. Whereas the S-XX-KafkaReadDemo is an Even-driven process that reads the data from the topic.

In this lab, you will stop and start the Event Source in the Kafka connection and observe that S-XX-KafkaReadDemo is executing without republishing the process.

#### **Objectives:**

- Stop the Event Source in the Kafka connection
- Start the Event Source in the Kafka connection

#### **Duration:**

20 Minutes

---

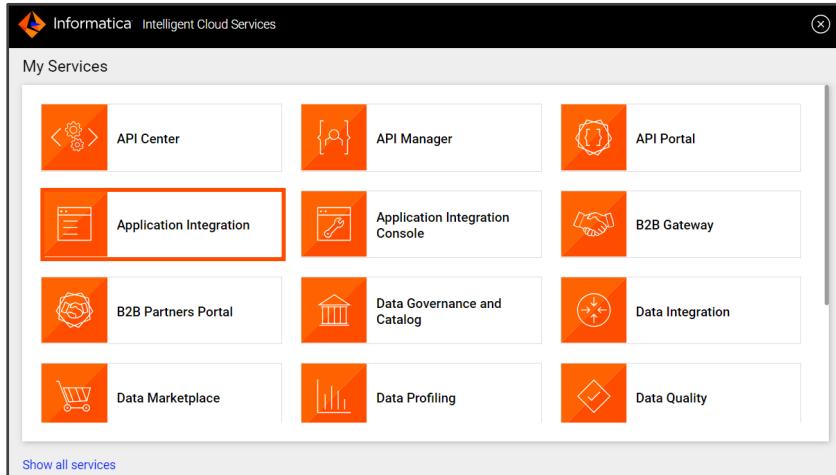
#### **Tasks**

**Note:** The screen in your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

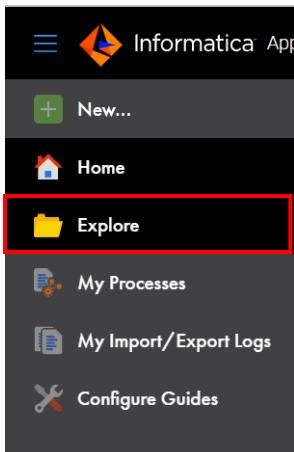
**IMPORTANT:** Before executing this lab, make sure that Kafka server is running on the Secure Agent installed system or machine.

## Execute an Event-driven Process

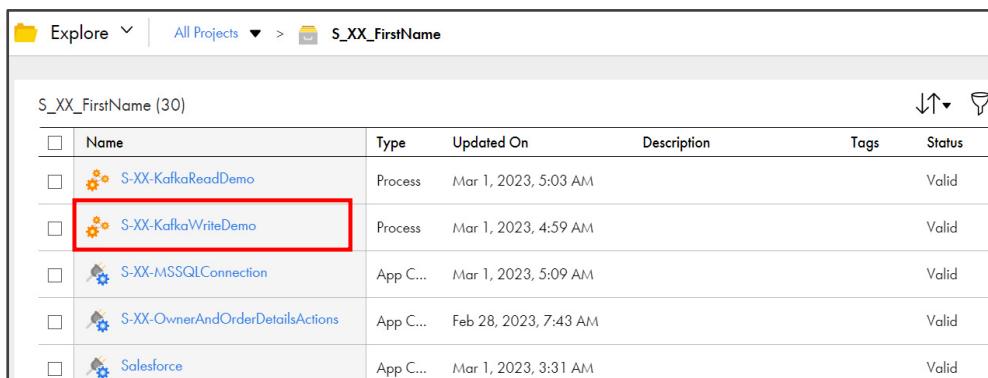
1. Log into IICS and access the **Application Integration** service.



2. From the navigation pane, click **Explore**.



3. Open your project folder S\_XX\_FirstName.
4. Search and open **S-XX-KafkaWriteDemo**.

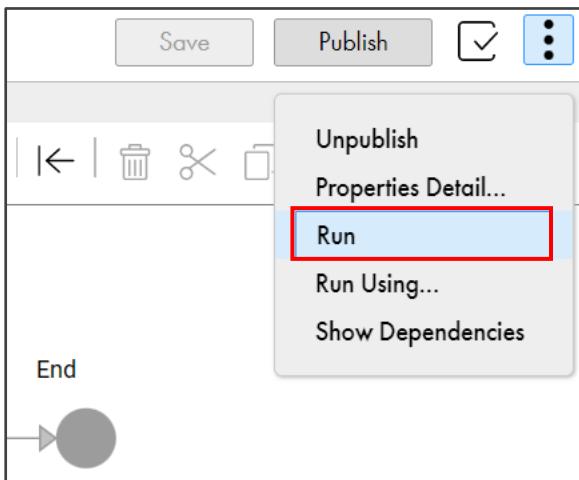


The screenshot shows the 'Explore' interface with the 'S\_XX\_FirstName' project selected. The table lists several processes and app components. The 'S-XX-KafkaWriteDemo' process is highlighted with a red box.

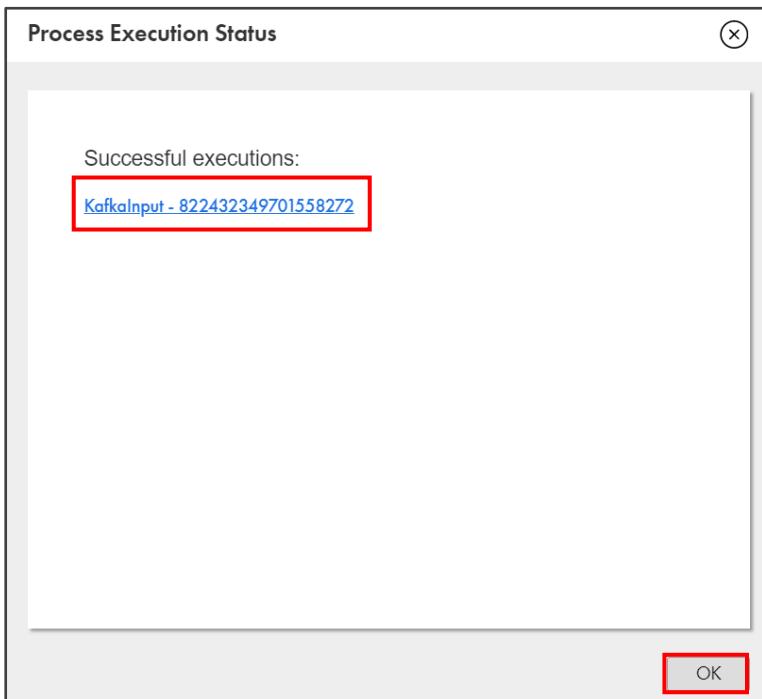
|                          | Name                             | Type     | Updated On            | Description | Tags | Status |
|--------------------------|----------------------------------|----------|-----------------------|-------------|------|--------|
| <input type="checkbox"/> | S-XX-KafkaReadDemo               | Process  | Mar 1, 2023, 5:03 AM  |             |      | Valid  |
| <input type="checkbox"/> | <b>S-XX-KafkaWriteDemo</b>       | Process  | Mar 1, 2023, 4:59 AM  |             |      | Valid  |
| <input type="checkbox"/> | S-XX-MSSQLConnection             | App C... | Mar 1, 2023, 5:09 AM  |             |      | Valid  |
| <input type="checkbox"/> | S-XX-OwnerAndOrderDetailsActions | App C... | Feb 28, 2023, 7:43 AM |             |      | Valid  |
| <input type="checkbox"/> | Salesforce                       | App C... | Mar 1, 2023, 3:31 AM  |             |      | Valid  |

**Tip:** You can use the Search box to find S-XX-KafkaWriteDemo.

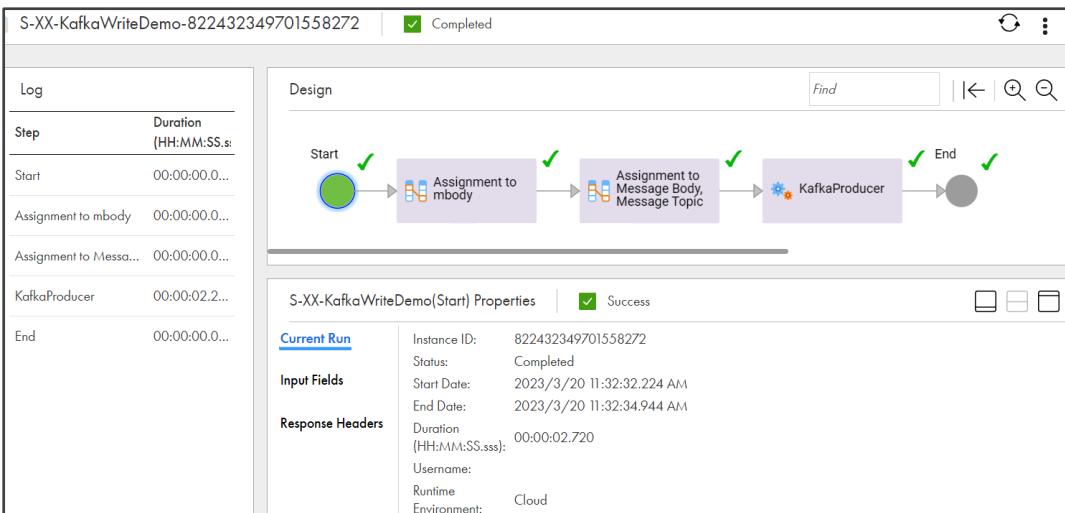
5. Click the Actions menu and select **Run**.



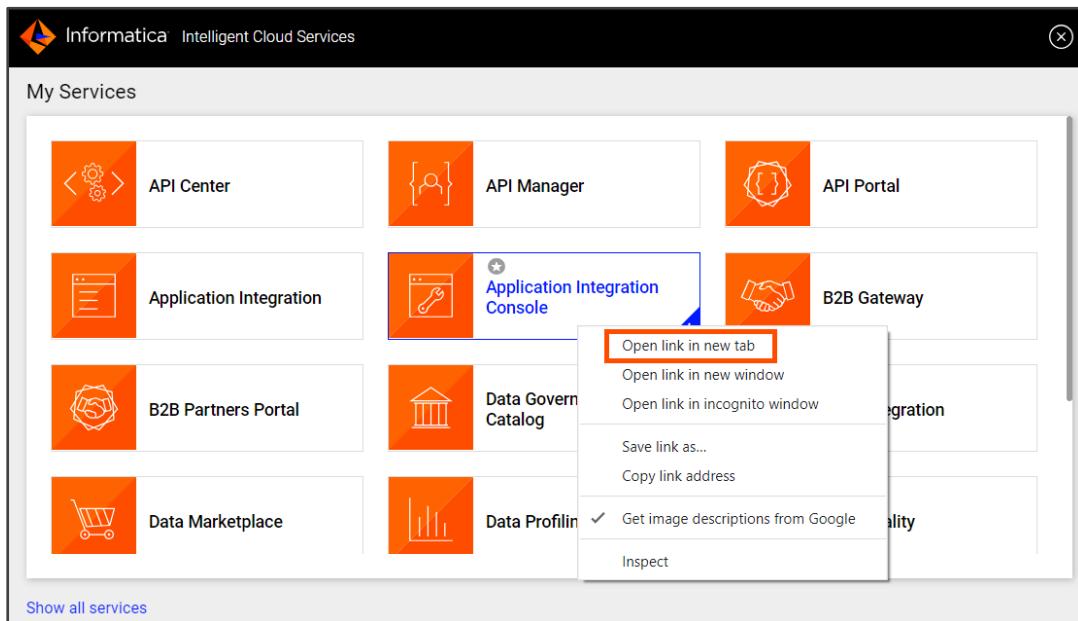
6. Select the execution link and click **OK**.



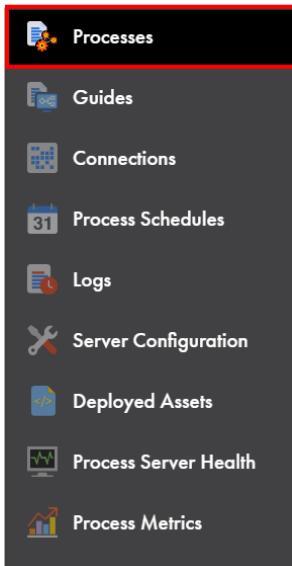
The S-XX-KafkaWriteDemo process executed successfully.



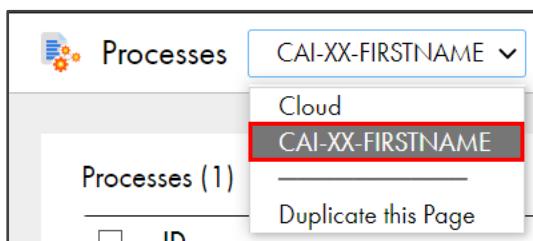
7. Open the **Application Integration Console** service in a new tab as you will toggle between Application Integration and Application Integration Console services frequently.



8. From the navigation pane, click **Processes** if it is not selected.



9. From the Process drop-down on the top, select your Secure Agent.

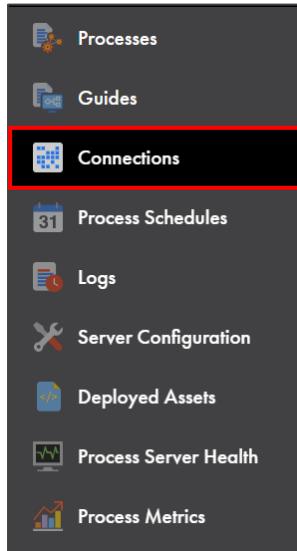


10. Observe that S-XX-KafkaReadDemo is executed.

| Processes (1) |                                    |         |                         |                   |   | Updated 5:29 PM IST / 11:59 AM UTC |  |  | Find |
|---------------|------------------------------------|---------|-------------------------|-------------------|---|------------------------------------|--|--|------|
| ID            | Name                               | Version | Start Date              | End Date          | Status  |                                    |  |  |      |
| 2554          | <a href="#">S-XX-KafkaReadDemo</a> | 1       | 2023/3/20 11:57:59.2... | 2023/3/20 11:5... | <input checked="" type="checkbox"/> Completed |                                    |  |  |      |

## Stop the Event Source in the Kafka Connection

11. From the navigation pane, click **Connections**.



12. From the Connections drop-down on the top, select your Secure Agent to see the connections.

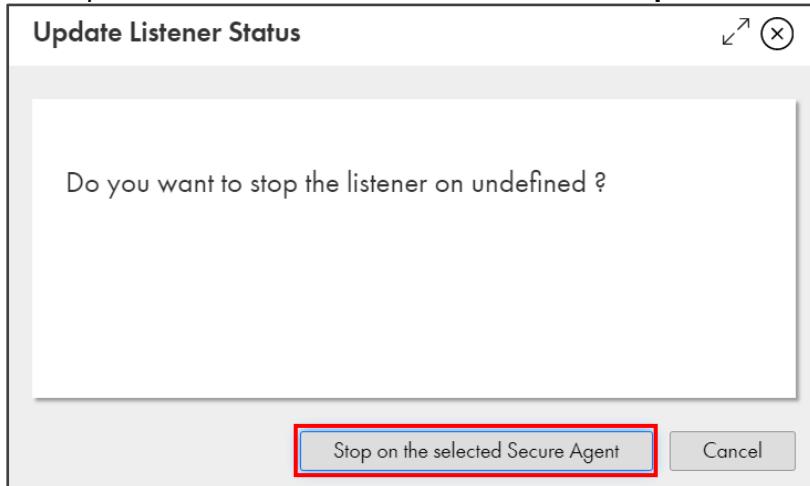
13. Expand S-XX-KafkaConnection to view the Event Source.

| Connections          |      |        |                |            | Updated 5:42 PM IST / 12:12 PM UTC | Start | Stop |
|----------------------|------|--------|----------------|------------|------------------------------------|-------|------|
| Name                 | Type | Status | Location       | Updated On |                                    |       |      |
| S-XX-KafkaConnection |      |        | S_XX_FirstName |            |                                    |       |      |

14. Select the **KafkaConsumer** event source and click **Stop**.

| Connections          |       |         |                |                           | Updated 5:42 PM IST / 12:12 PM UTC | Start | Stop | Find |
|----------------------|-------|---------|----------------|---------------------------|------------------------------------|-------|------|------|
| Name                 | Type  | Status  | Location       | Updated On                |                                    |       |      |      |
| S-XX-KafkaConnection |       |         | S_XX_FirstName |                           |                                    |       |      |      |
| KafkaCons...         | Kafka | Started |                | 2023/3/20 11:56:57:032 AM |                                    |       |      |      |

15. To stop an event source in a connection, click **Stop on the selected Secure Agent**.

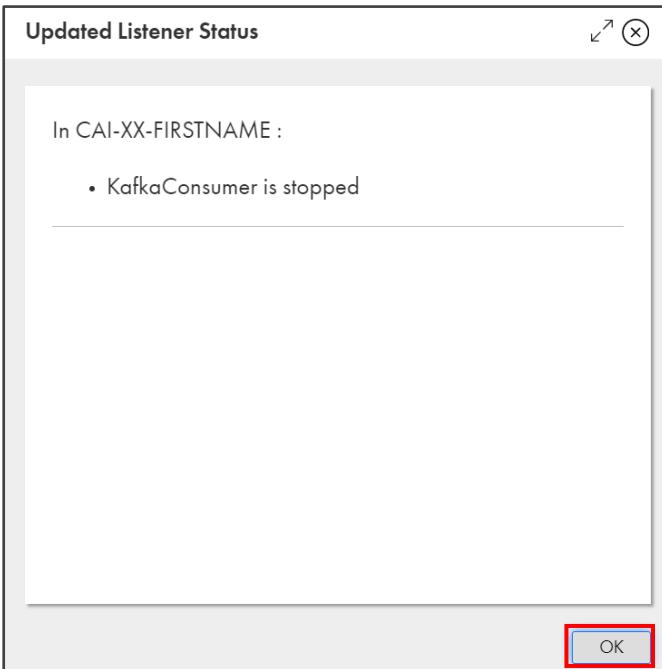


Update Listener Status

Do you want to stop the listener on undefined?

Stop on the selected Secure Agent      Cancel

16. Click **OK**.



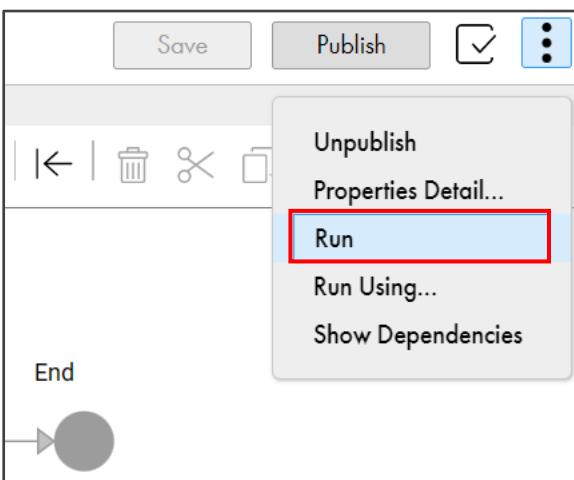
17. On the Connections page, expand the connection and observe that the status of the KafkaConsumer event source is updated to **Stopped**. Click the **Refresh** icon in case you are not viewing the updated status.

| Connections                |       |         |                |            | Updated 5:55 PM IST / 12:25 PM UTC | Start | Stop |
|----------------------------|-------|---------|----------------|------------|------------------------------------|-------|------|
| Name                       | Type  | Status  | Location       | Updated On |                                    |       |      |
| ▼ S-XX-<br>KafkaConnection |       |         | S_XX_FirstName |            |                                    |       |      |
| KafkaCons...               | Kafka | Stopped |                |            | 2023/3/20 12:23:38:40              |       |      |

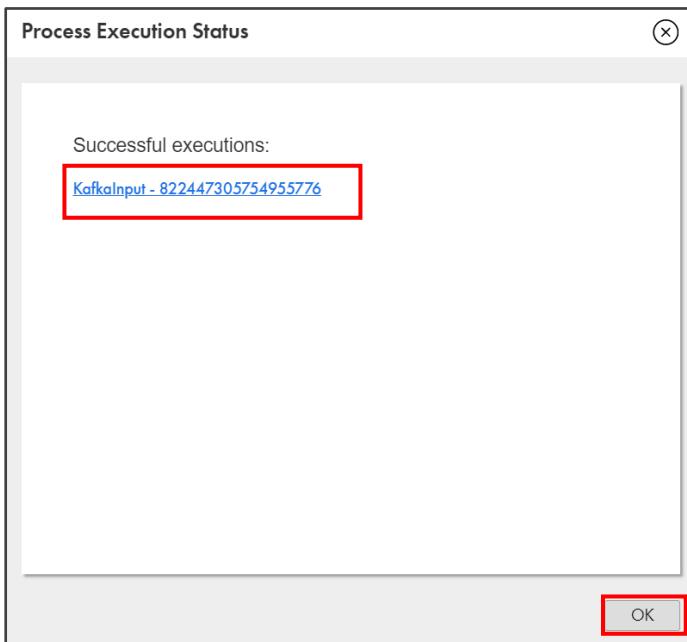
18. Go back to the **Application Integration** service.

19. Open the **S-XX-KafkaWriteDemo** process if it is closed.

20. Click the Actions menu and select **Run**.

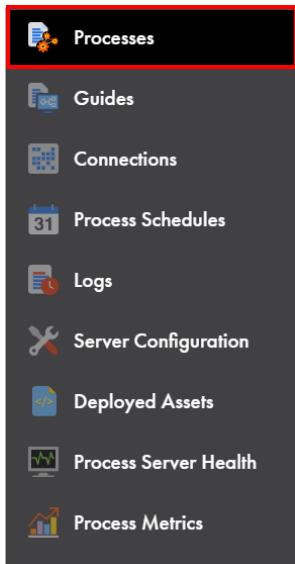


21. Select the execution link and click **OK**.

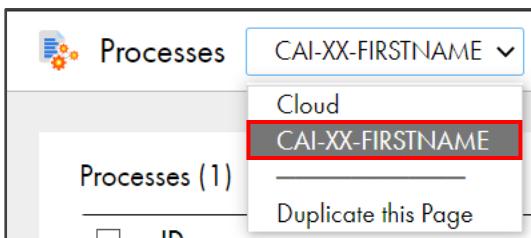


The S-XX-KafkaWriteDemo process executed successfully.

22. Access the **Application Integration Console** service and from the navigation pane, click **Processes**.



23. From the Process drop-down on the top, select your Secure Agent.



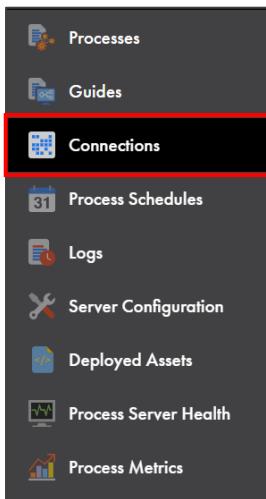
24. Observe that S-XX-KafkaReadDemo is not executed this time. This is because you have stopped the KafkaConsumer event source.

| Processes (1) |                    |         |                         |                   |           |   |
|---------------|--------------------|---------|-------------------------|-------------------|-----------|---|
| ID            | Name               | Version | Start Date              | End Date          | Status    |   |
| 2554          | S-XX-KafkaReadDemo | 1       | 2023/3/20 11:57:59:2... | 2023/3/20 11:5... | Completed |   |

The job that you currently viewing the previous run's process.

## Start the Event Source in the Kafka Connection

25. From the navigation pane, click **Connections**.



26. Make sure that Secure Agent is selected in the Connections drop-down on the top.

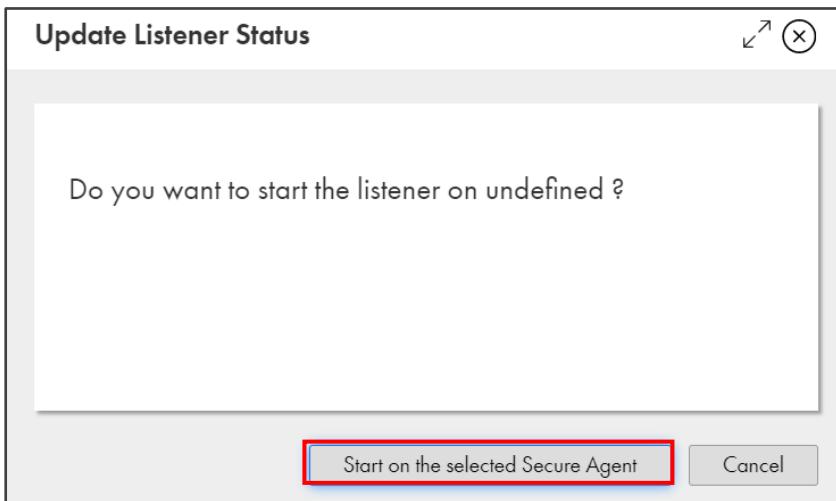
27. Expand S-XX-KafkaConnection to view the Event Source.

| Connections            |      |        |                |            |
|------------------------|------|--------|----------------|------------|
| Name                   | Type | Status | Location       | Updated On |
| ► S-XX-KafkaConnection |      |        | S_XX_FirstName |            |

28. Select the **KafkaConsumer** event source and click **Start**.

| Connections  |       |         |                |                       |
|--|-------|---------|----------------|-----------------------|
| Name   | Type  | Status  | Location       | Updated On            |
| ▼ S-XX-KafkaConnection                                 |       |         | S_XX_FirstName |                       |
| <input checked="" type="checkbox"/> KafkaCons... Kafka | Kafka | Stopped |                | 2023/3/20 12:23:38:40 |

29. To start the Event Source, click **Start on the selected Secure Agent**.



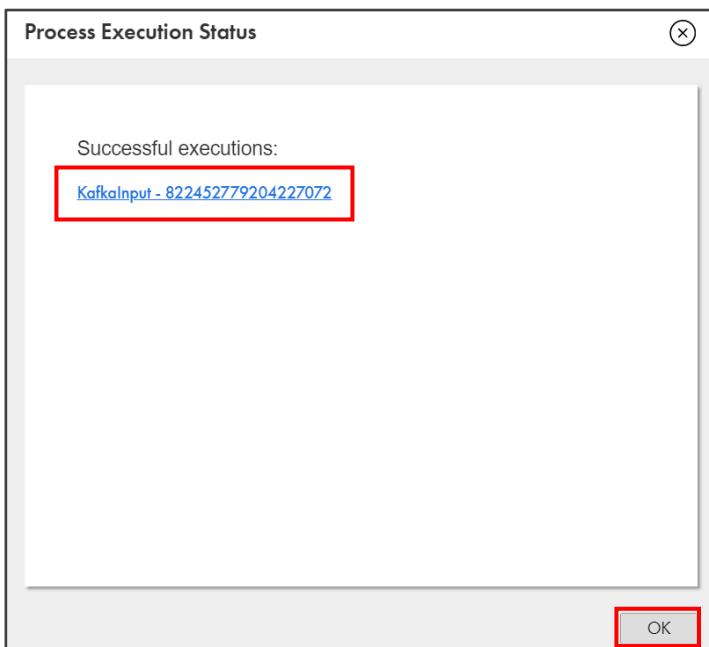
30. Click **OK**.

The status of the KafkaConsumer event source is updated to Started on the Connections page.

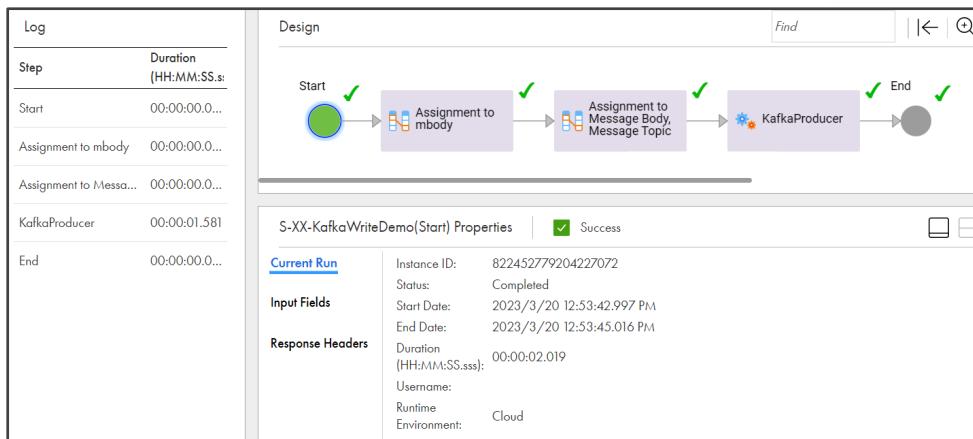
| Connections            |       |         |                |            | Updated 6:18 PM IST / 12:48 PM UTC | Start | Stop |
|------------------------|-------|---------|----------------|------------|------------------------------------|-------|------|
| Name                   | Type  | Status  | Location       | Updated On |                                    |       |      |
| ▼ S-XX-KafkaConnection |       |         | S_XX_FirstName |            |                                    |       |      |
| KafkaCons...           | Kafka | Started |                |            | 2023/3/20 12:48:13:78              |       |      |

31. Go back to the **Application Integration** service and run the **S-XX-KafkaWriteDemo** process once again.

32. Select the execution link and click **OK**.



The S-XX-KafkaWriteDemo process executed successfully.



| Step                                      | Duration (HH:MM:SS.s) |
|---|-----------------------|
| Start                                     | 00:00:00.0...         |
| Assignment to imbody                      | 00:00:00.0...         |
| Assignment to Message Body, Message Topic | 00:00:00.0...         |
| KafkaProducer                             | 00:00:01.581          |
| End                                       | 00:00:00.0...         |

33. Switch back to the **Application Integration Console** service and from the navigation pane, click **Processes**.
34. Click the  icon to refresh the Processes page.
35. Observe that S-XX-KafkaReadDemo is executed this time. This is because you have started the KafkaConsumer event source.

| Processes (2)            |      |  |         |                         |                   |   |
|--------------------------|------|--|---------|-------------------------|-------------------|---|
|                          | ID   | Name   | Version | Start Date              | End Date          | Status  |
| <input type="checkbox"/> | 2556 |  S-XX-KafkaReadDemo  | 1       | 2023/3/20 12:48:14:2... | 2023/3/20 12:4... |  Completed  |
| <input type="checkbox"/> | 2554 |  S-XX-KafkaReadDemo | 1       | 2023/3/20 11:57:59:2... | 2023/3/20 11:5... |  Completed |

This concludes the lab.

# Module 14: API Management

## Lab 14-1: API Manager

### Overview:

You can create a managed API for any Informatica Cloud Application Integration service. By default, the managed API is active and API Portal view for the API is enabled. The name that you assign to the API is part of the API URL.

In this lab, you will create a managed API for the ExchangeRate process.

### Objectives:

- Create a Managed API for the ExchangeRate process

### Duration:

15 Minutes

---

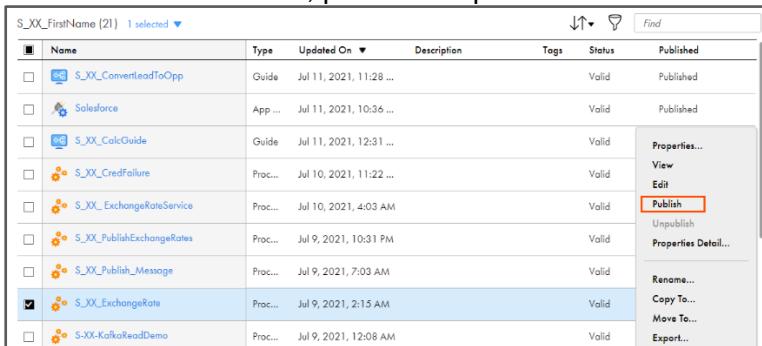
### Tasks

**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

### Publish the ExchangeRateService Process

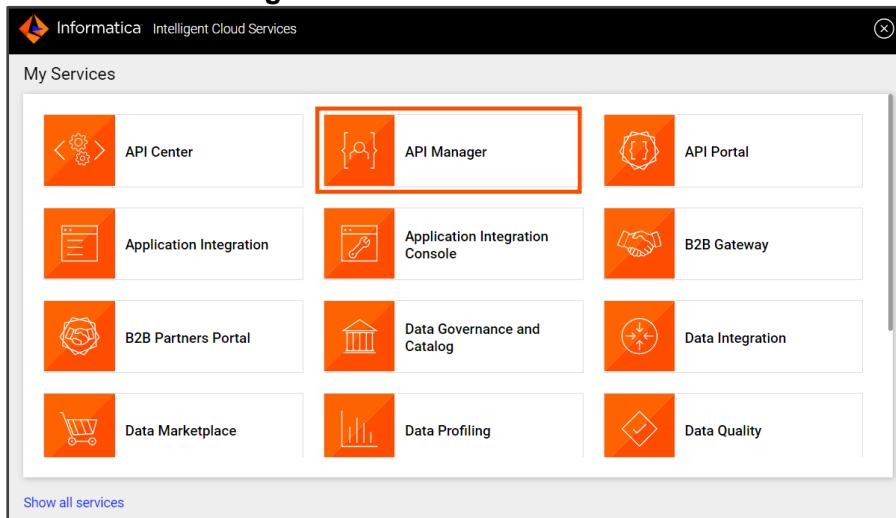
While executing the sub-process failure lab (11-1), you have unpublished the ExchangeRate process. For this lab, you must publish the process.

1. Navigate to the **Application Integration** service.
2. Open your project folder and locate the **S\_XX\_ExchangeRate** process.
3. From the Actions menu, publish the process.



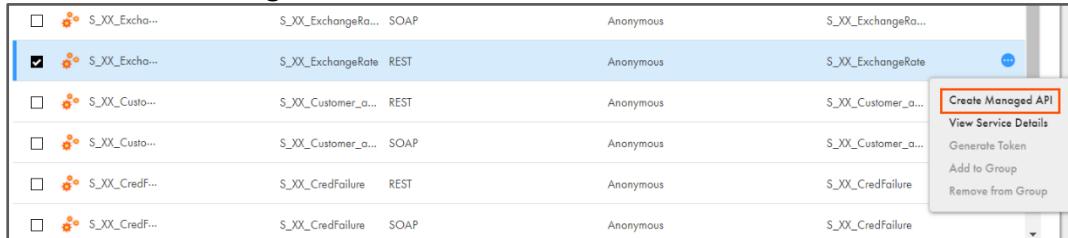
## Create a Managed API for the ExchangeRate process

4. Switch to **API Manager**.



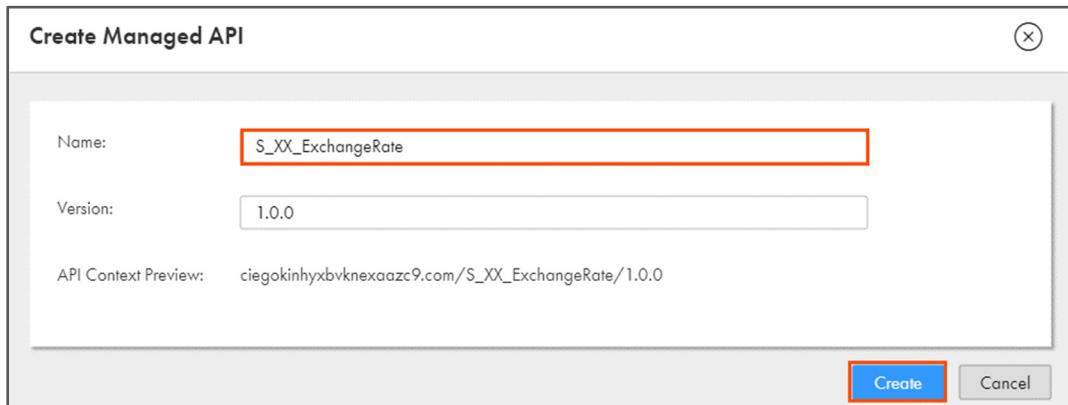
**Note:** If you get the API Domain Name window, click **Save**.

5. In the **API Registry** tab, for the ExchangeRate (REST) process, from the **Actions** menu, select **Create Managed API**.



|   |                    |      |           |                    |
|---|--------------------|------|-----------|--------------------|
| <input type="checkbox"/> S_XX_Excha...            | S_XX_ExchangeRa... | SOAP | Anonymous | S_XX_ExchangeRa... |
| <input checked="" type="checkbox"/> S_XX_Excha... | S_XX_ExchangeRate  | REST | Anonymous | S_XX_ExchangeRate  |
| <input type="checkbox"/> S_XX_Custo...            | S_XX_Customer_a... | REST | Anonymous | S_XX_Customer_a... |
| <input type="checkbox"/> S_XX_Custo...            | S_XX_Customer_a... | SOAP | Anonymous | S_XX_Customer_a... |
| <input type="checkbox"/> S_XX_CredF...            | S_XX_CredFailure   | REST | Anonymous | S_XX_CredFailure   |
| <input type="checkbox"/> S_XX_CredF...            | S_XX_CredFailure   | SOAP | Anonymous | S_XX_CredFailure   |

6. Click **Create**.



**Create Managed API**

Name:

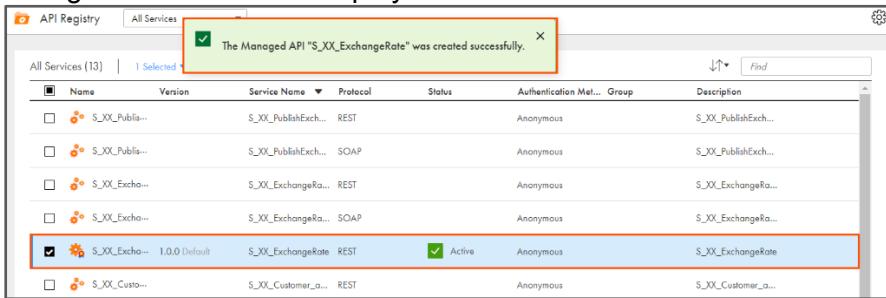
Version:

API Context Preview: ciegokinhxvknexazc9.com/S\_XX\_ExchangeRate/1.0.0

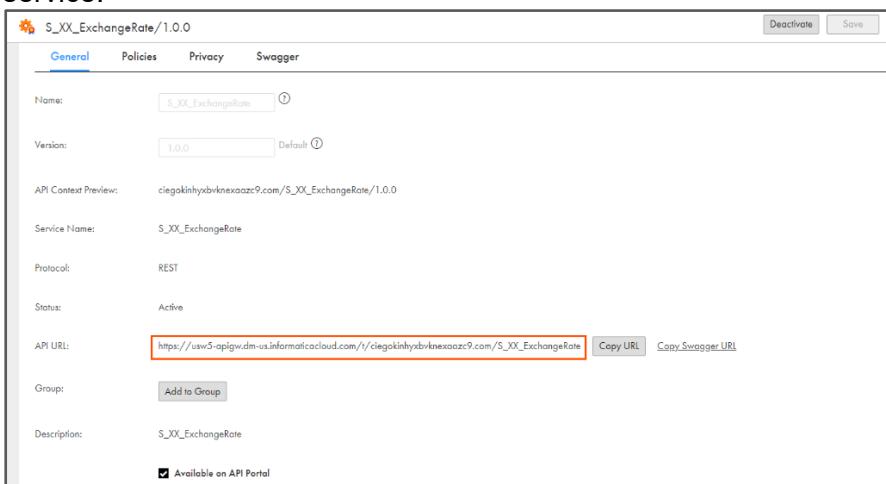
**Create** **Cancel**

A message '**Managed API is created successfully**' appears.

7. Locate the **S\_XX\_ExchangeRate** REST service and observe that the icon of this service changes and the status displays as **Active**.



8. Open the service and check the **API URL** field. This is a proxy URL provided for the service.



| General   |   | Policies | Privacy | Swagger |
|---|---|----------|---------|---------|
| Name:   | S_XX_ExchangeRate   |          |         |         |
| Version:  | 1.0.0 Default   |          |         |         |
| API Context Preview:  | ciegokinhydbvknexaazz9.com/S_XX_ExchangeRate/1.0.0  |          |         |         |
| Service Name:   | S_XX_ExchangeRate   |          |         |         |
| Protocol:   | REST  |          |         |         |
| Status:   | Active  |          |         |         |
| API URL:  | <input type="text" value="https://usw5-apigw.dm-us.informaticacloud.com/t/ciegokinhydbvknexaazz9.com/S_XX_ExchangeRate"/> <span>Copy URL</span> <span>Copy Swagger URL</span> |          |         |         |
| Group:  | <input type="button" value="Add to Group"/>   |          |         |         |
| Description:  | S_XX_ExchangeRate   |          |         |         |
| <input checked="" type="checkbox"/> Available on API Portal |   |          |         |         |

---

*This concludes the lab.*

# Module 15: CAI and CDI Integration

## Lab 15-1: Create a Flat File Connection

### Overview:

Flat file connections store the information to create, access, and store flat files.

In this lab, you will set up a connection to access flat files on your local machine.

### Objective:

- Create a flat file connection

### Duration:

10 minutes

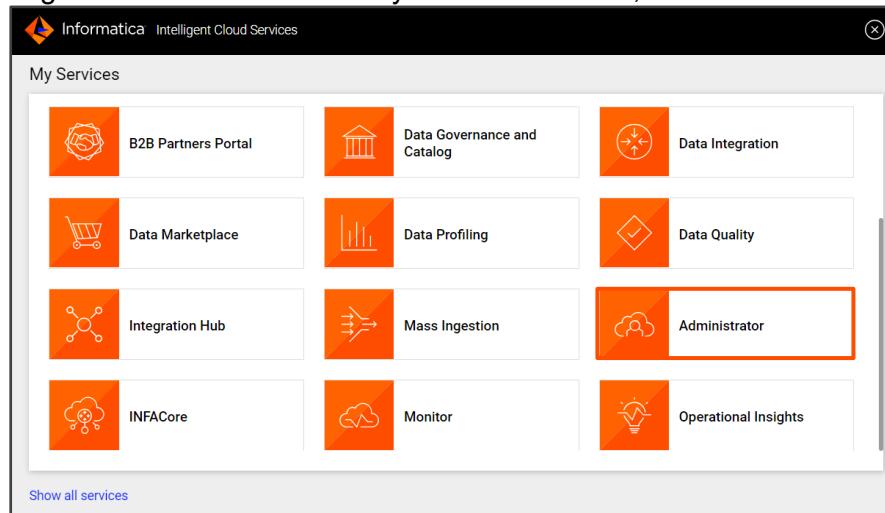
---

### Tasks

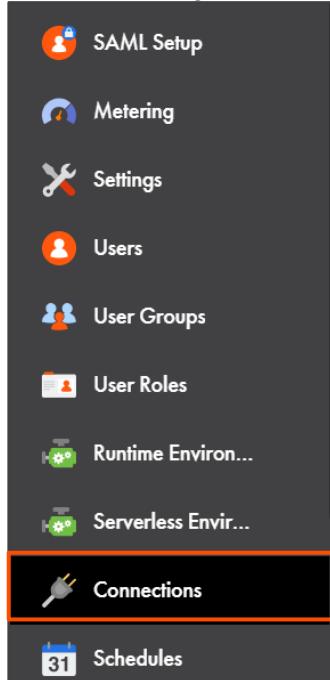
**Note:** The screen in your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

### Create a Flat File Connection in IICS

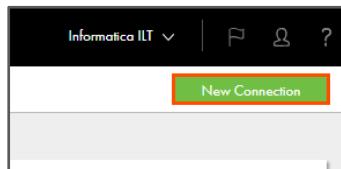
1. Log in to IICS and from the My Services window, select **Administrator**.



2. From the Navigation pane, select **Connections**.



3. To create a new connection, select **New Connection**.



4. Enter the Connection Name as **S\_XX\_FlatFile**.

5. From the Type drop-down, select **Flat File**.


 A screenshot of the "Connection Details" dialog box. It has a header with a gear icon. Below the header, there are three input fields:
 

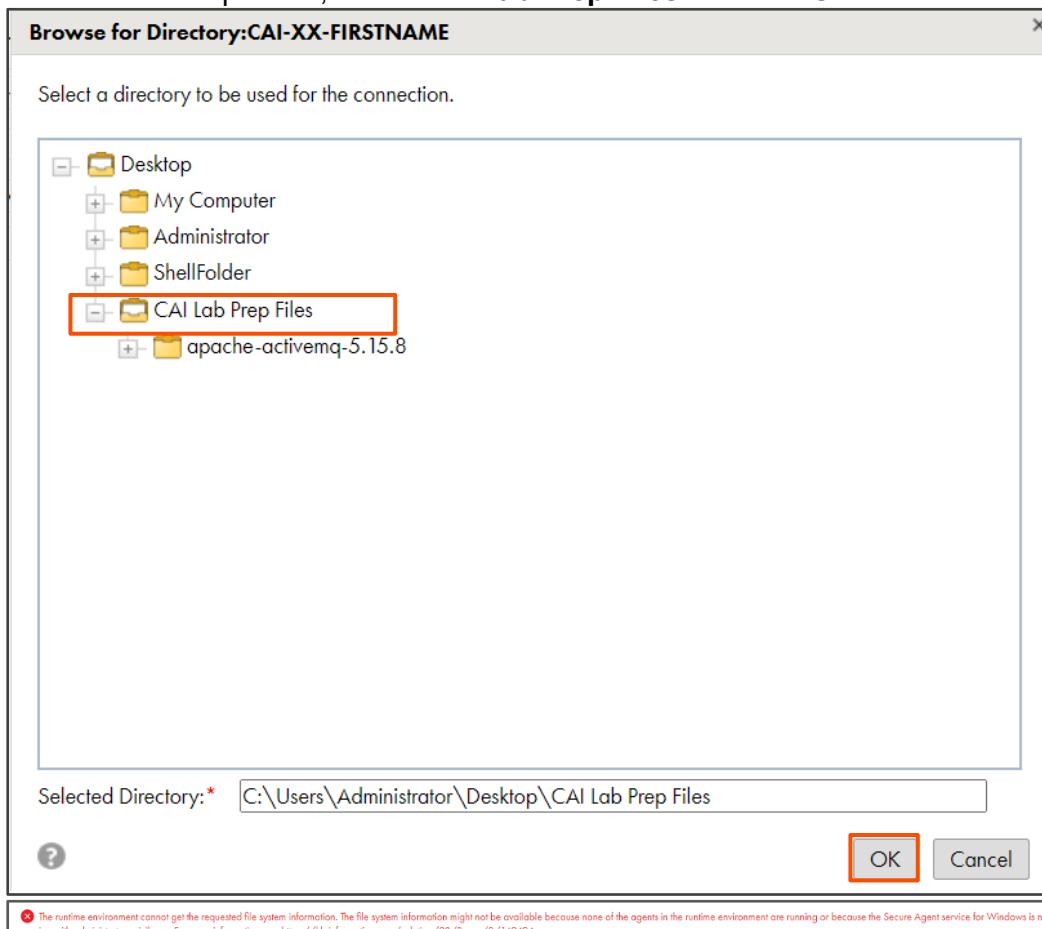
- Connection Name:**  (highlighted with a red box)
- Description:**
- Type:**  (highlighted with a red box)

6. From the **Runtime Environment** drop-down, select your secure agent group.

7. From the Directory field, click **Browse**.



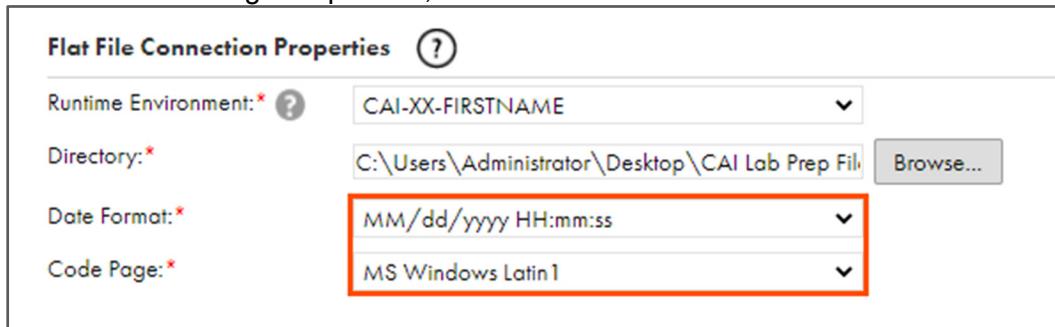
8. From the Desktop folder, select **CAI Lab Prep Files** and click **OK**.



**Note:** If you get the above error, copy the **CAI Lab Prep Files** location, and paste it in the Selected Directory field.

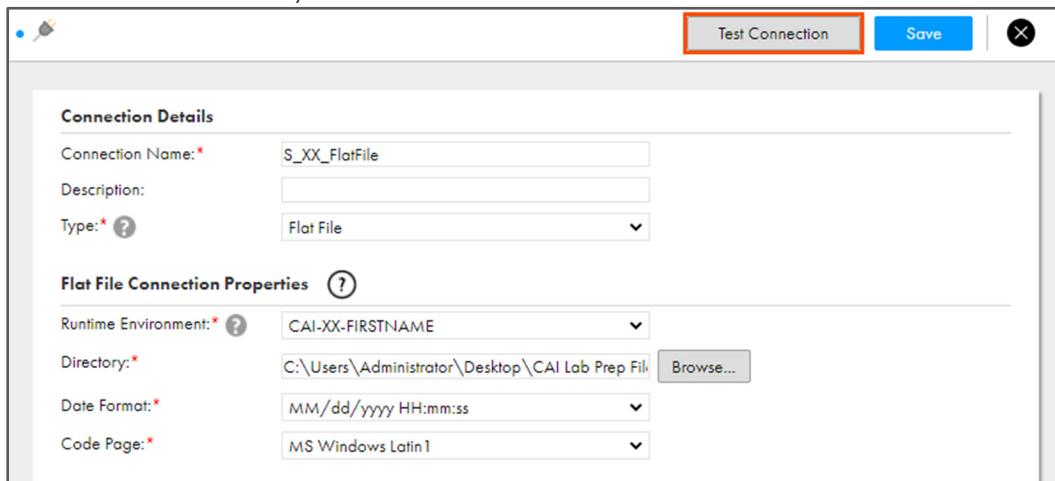
9. Retain the value in the **Date Format** field.

10. From the Code Page drop-down, select **MS Windows Latin1**.



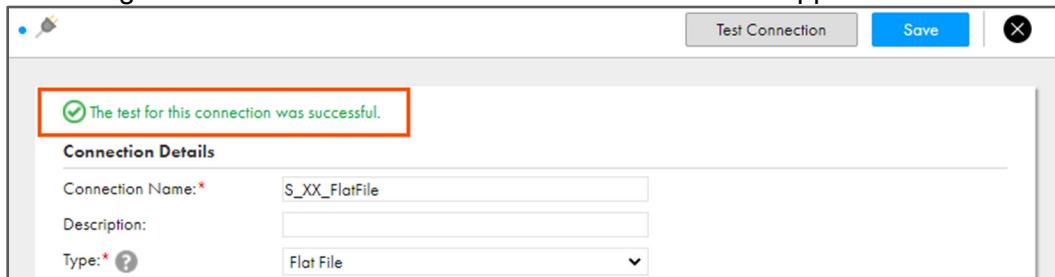
The screenshot shows the 'Flat File Connection Properties' dialog. The 'Code Page:' dropdown is highlighted with a red box, showing the value 'MS Windows Latin1'. Other fields include 'Runtime Environment:' set to 'CAI-XX-FIRSTNAME', 'Directory:' set to 'C:\Users\Administrator\Desktop\CAI Lab Prep Fil...', 'Date Format:' set to 'MM/dd/yyyy HH:mm:ss', and 'Type:' set to 'Flat File'.

11. To test the connection, click **Test Connection**.



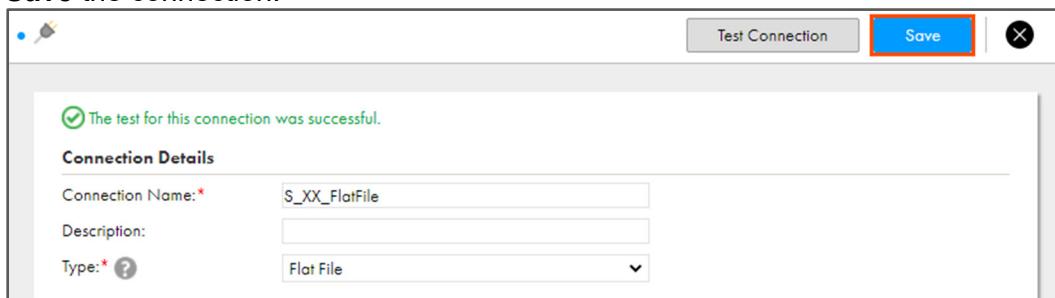
The screenshot shows the 'Flat File Connection Properties' dialog with the 'Test Connection' button highlighted with a red box. The connection details and properties are identical to the previous screenshot.

A message '**The test for the connection was successful**' appears.



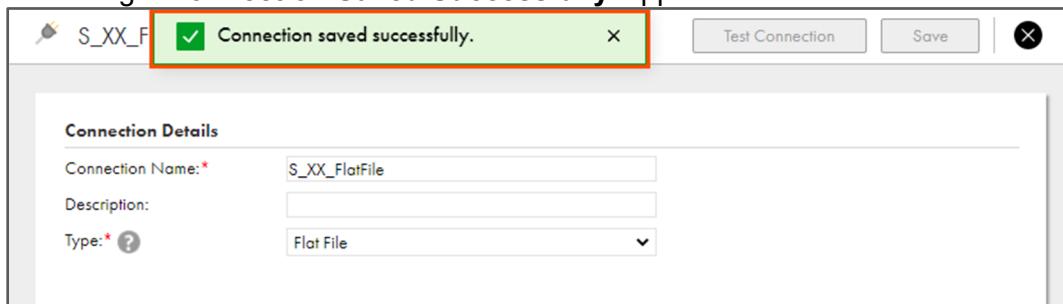
The screenshot shows the 'Flat File Connection Properties' dialog with a message box containing the text 'The test for this connection was successful.' highlighted with a red box. The connection details and properties are identical to the previous screenshots.

12. **Save** the connection.



The screenshot shows the 'Flat File Connection Properties' dialog with the 'Save' button highlighted with a red box. A message box at the top left contains the text 'The test for this connection was successful.' The connection details and properties are identical to the previous screenshots.

13. A message 'Connection Saved Successfully' appears.



---

*This concludes the lab.*

## Module 15: CAI and CDI Integration

### Lab 15-2: Create a Basic Mapping and a Mapping Task

#### Overview:

In IICS, the Mapping designer creates a mapping that you can use in several mapping tasks.

In this lab, you will create a mapping to filter the Order data and create a mapping task.

#### Objective:

- Use the Mapping Designer to create a mapping
- Create a Mapping task

#### Duration:

20 minutes

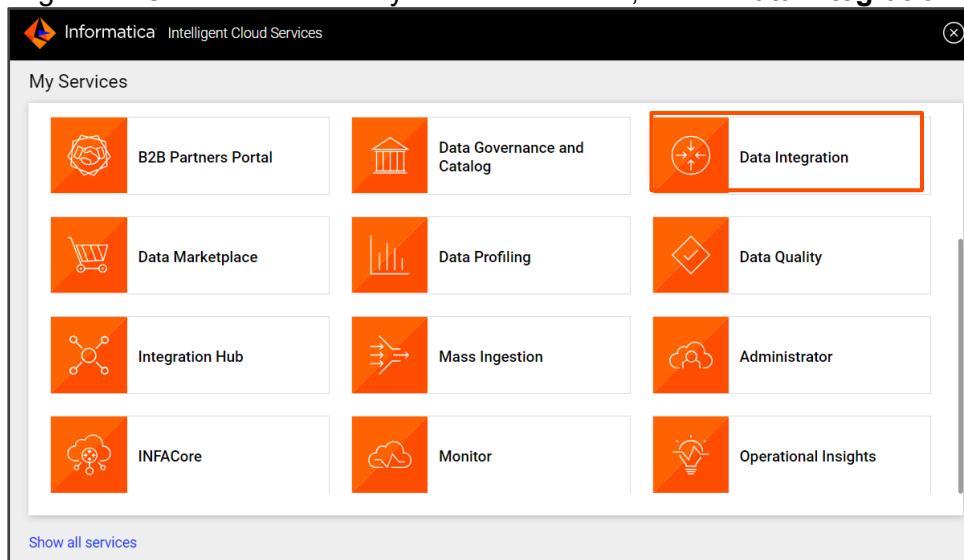
---

### Tasks

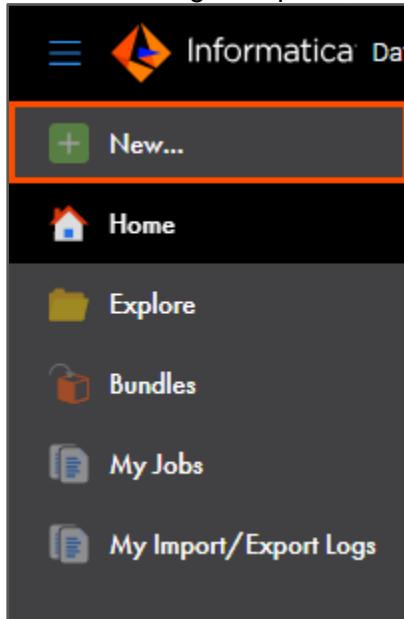
**Note:** The screen in your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

#### Create a Mapping

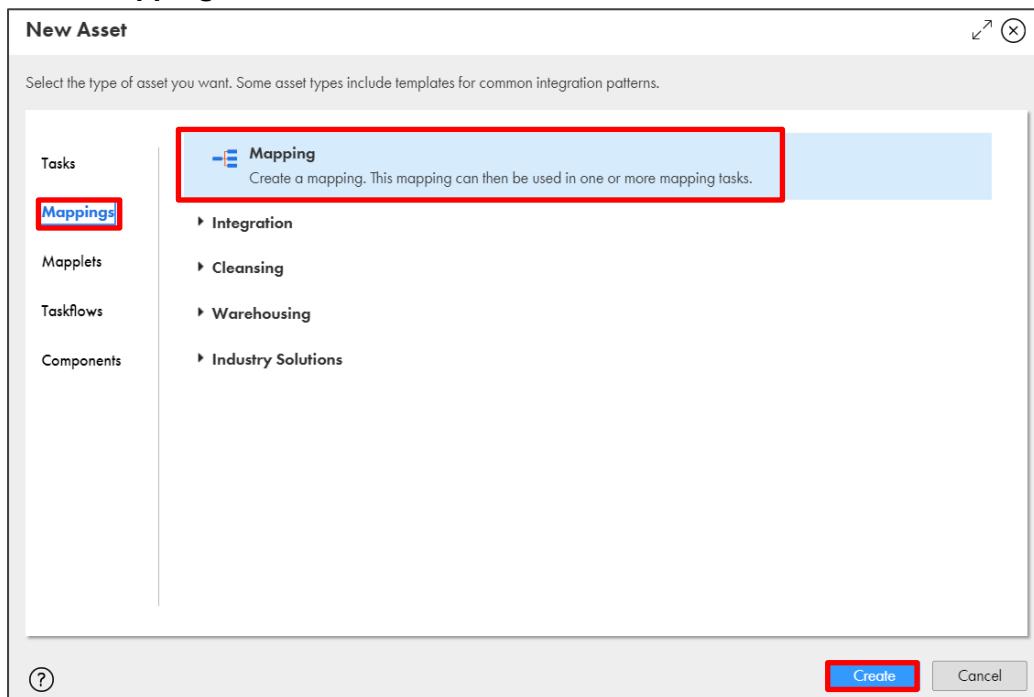
1. Log in to IICS and from the My Services window, select **Data Integration**.



2. From the Navigation pane, select **New**.

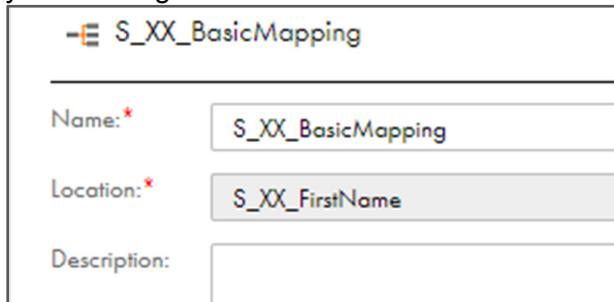


3. From the New Asset window, select the **Mappings** tab.  
 4. Select **Mapping** and click **Create**.



The Mapping Designer opens.

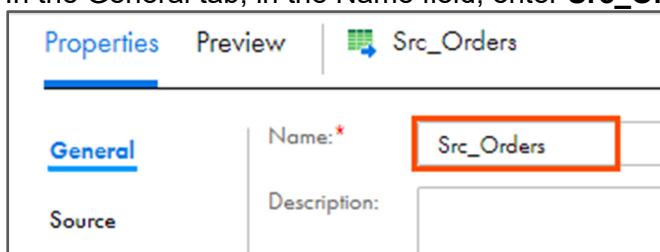
5. In the Name field, enter **S\_XX\_BasicMapping** and verify that the asset location is set to your working folder.



The screenshot shows a mapping configuration window titled "S\_XX\_BasicMapping". It contains three fields: "Name:" with the value "S\_XX\_BasicMapping", "Location:" with the value "S\_XX\_FirstName", and a "Description:" field which is empty.

6. To configure the source, from the mapping canvas, click the **Source** transformation.

7. In the General tab, in the Name field, enter **Src\_Orders**.

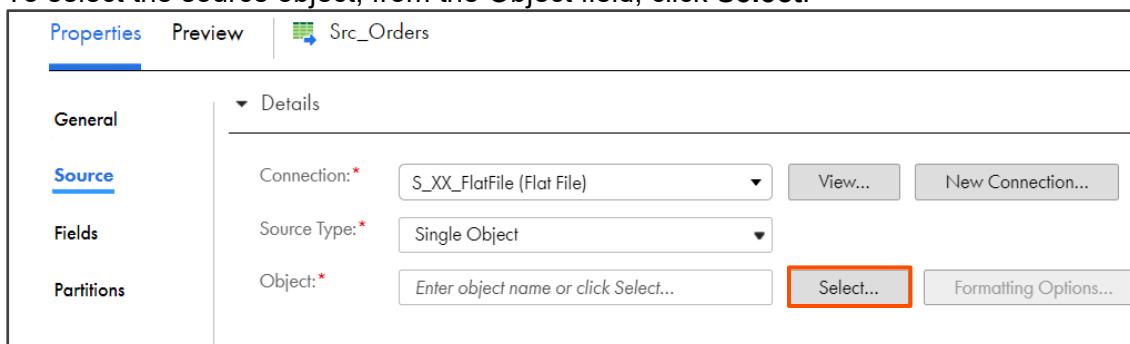


The screenshot shows the properties pane for the "Src\_Orders" transformation. The "General" tab is selected. The "Name:" field is highlighted with a red box and contains the value "Src\_Orders". The "Description:" field is empty.

8. From the properties pane, click **Source**.

9. From Connection drop-down, select **S\_XX\_FlatFile** and retain Source Type as **Single Object**.

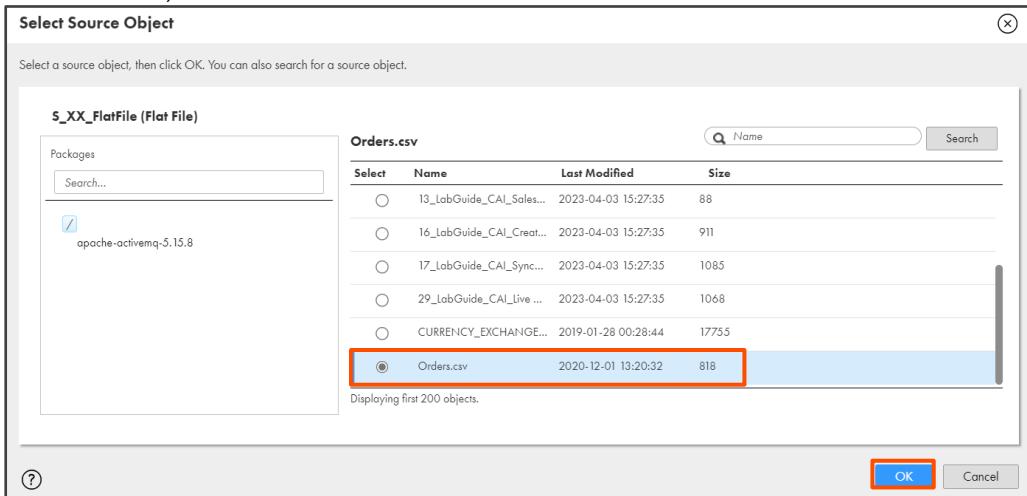
10. To select the source object, from the Object field, click **Select**.



The screenshot shows the "Select Source Object" window. The "Source" tab is selected. The "Connection:" dropdown is set to "S\_XX\_FlatFile (Flat File)". The "Source Type:" dropdown is set to "Single Object". The "Object:" field contains the placeholder text "Enter object name or click Select...". The "Select..." button is highlighted with a red box.

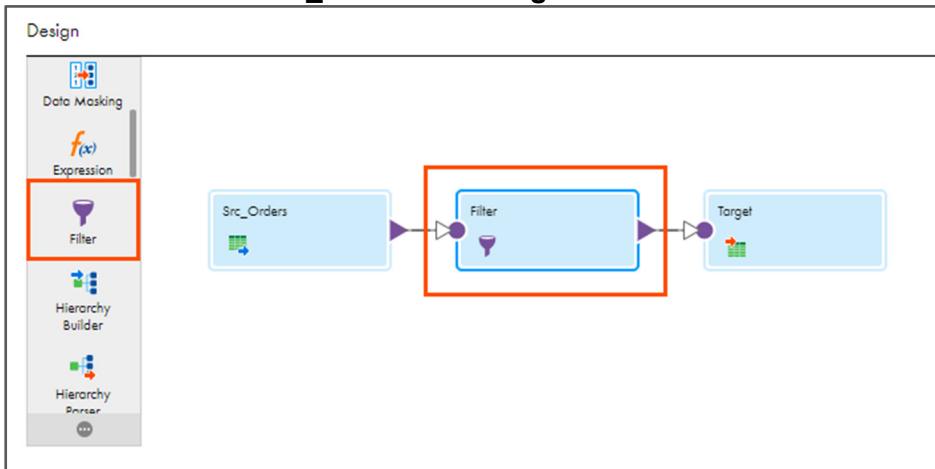
The Select Source Object window appears.

11. From the list, select **Orders.csv** and click **OK**.

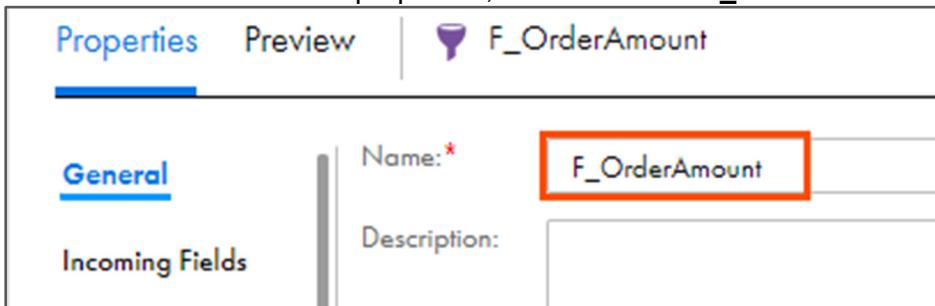


### Add Filter Transformation

12. From the list of available transformations, drag and drop **Filter** transformation on to the connector between **Src\_Orders** and **Target**.



13. In General section of Filter properties, enter Name as **F\_OrderAmount**.



14. From the properties pane, click **Filter**.

15. Retain Filter Condition as **Simple** and to add a new filter condition, click .

16. Enter filter condition as shown in the table below:

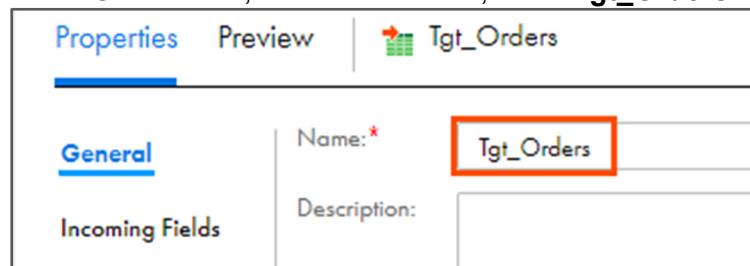
| Field Name  | Operator                    | Value |
|-------------|-----------------------------|-------|
| OrderAmount | >= (Greater than or equals) | 2000  |



The screenshot shows the Properties pane for a transformation. Under the General tab, the Filter Condition is set to "Simple". In the Filter Conditions section, there is one entry: "OrderAmount" with operator ">=" and value "2000". The entire row for this filter condition is highlighted with a red box.

17. To configure the target, from the mapping canvas, click the **target** transformation.

18. In the General tab, in the Name field, enter **Tgt\_Orders**.

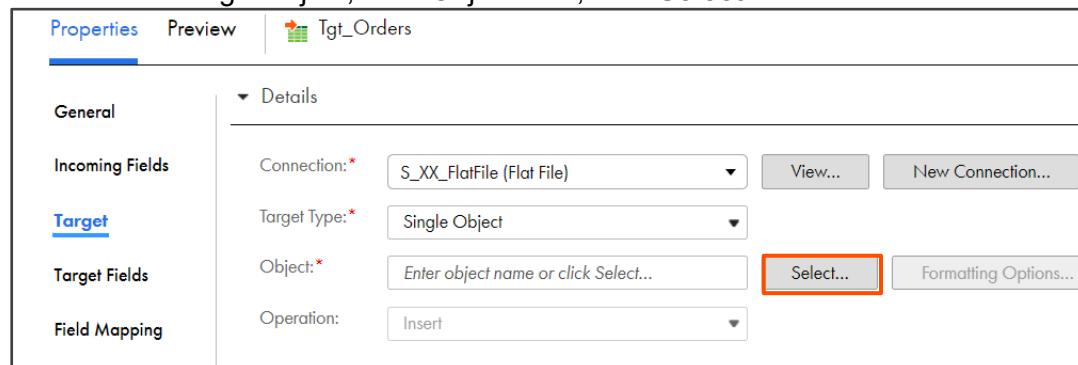


The screenshot shows the Properties pane for a target transformation named "Tgt\_Orders". Under the General tab, the "Name" field is set to "Tgt\_Orders" and is highlighted with a red box. The "Description" field is empty.

19. From the properties pane, click **Target**.

20. From Connection drop-down, select **S\_XX\_FlatFile** and retain the Target Type as **Single Object**.

21. To select the target object, from Object field, click **Select...**.

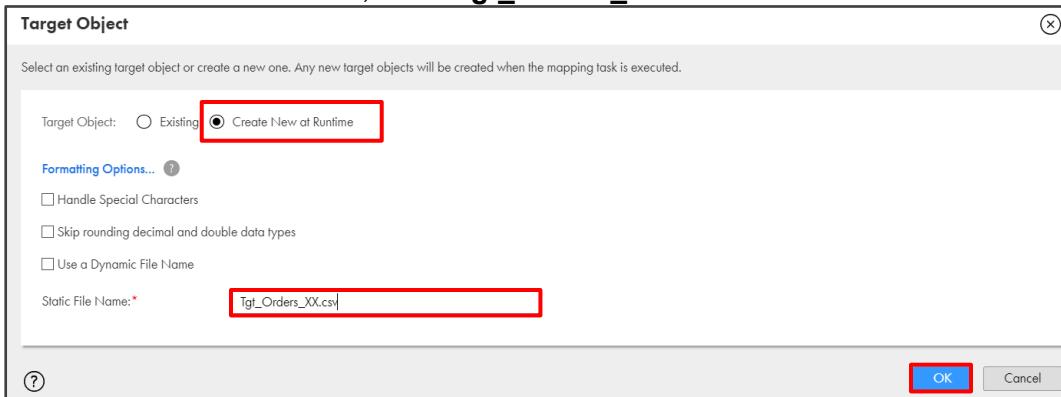


The screenshot shows the Properties pane for a target transformation. The "Target" tab is selected. Under "Connection:", "S\_XX\_FlatFile (Flat File)" is selected. Under "Target Type:", "Single Object" is selected. Under "Object:", there is a text input field "Enter object name or click Select..." and a button "Select..." which is highlighted with a red box. Other tabs like "General", "Incoming Fields", "Target Fields", and "Field Mapping" are also visible.

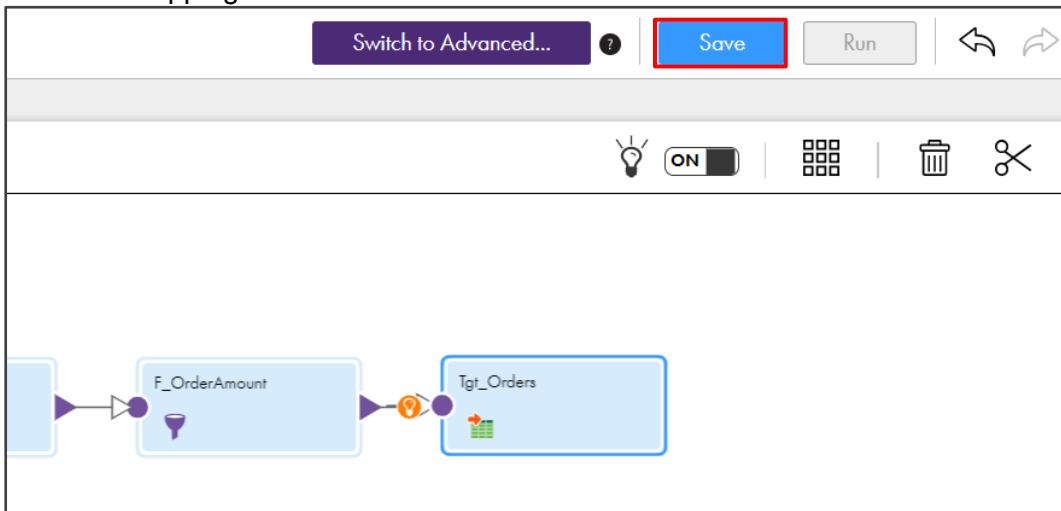
The Target Object window appears.

22. To create the target at runtime, select **Create New at Runtime**.

23. In the Static File Name field, enter **Tgt\_Orders\_XX.csv** and click **OK**.



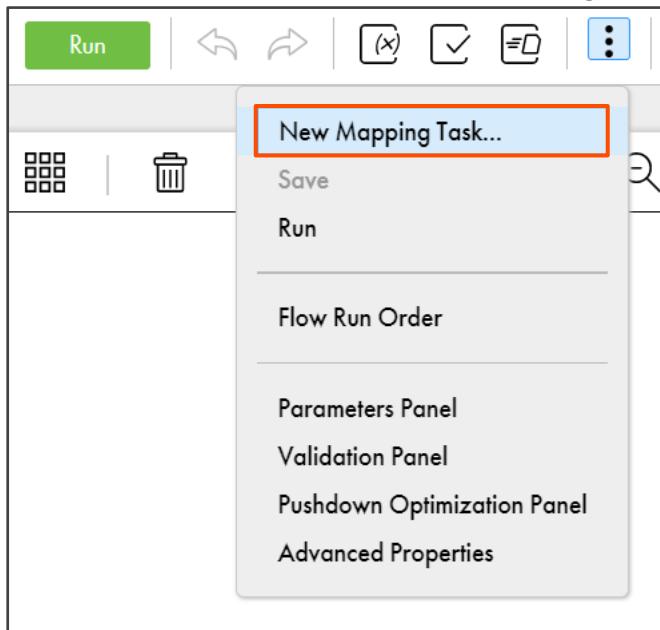
24. Save the mapping.



**Note:** After you save the mapping, the mapping status (**Valid or Invalid**) is automatically displayed next to the mapping name. To check for the validity of a mapping while working with the mapping, click .

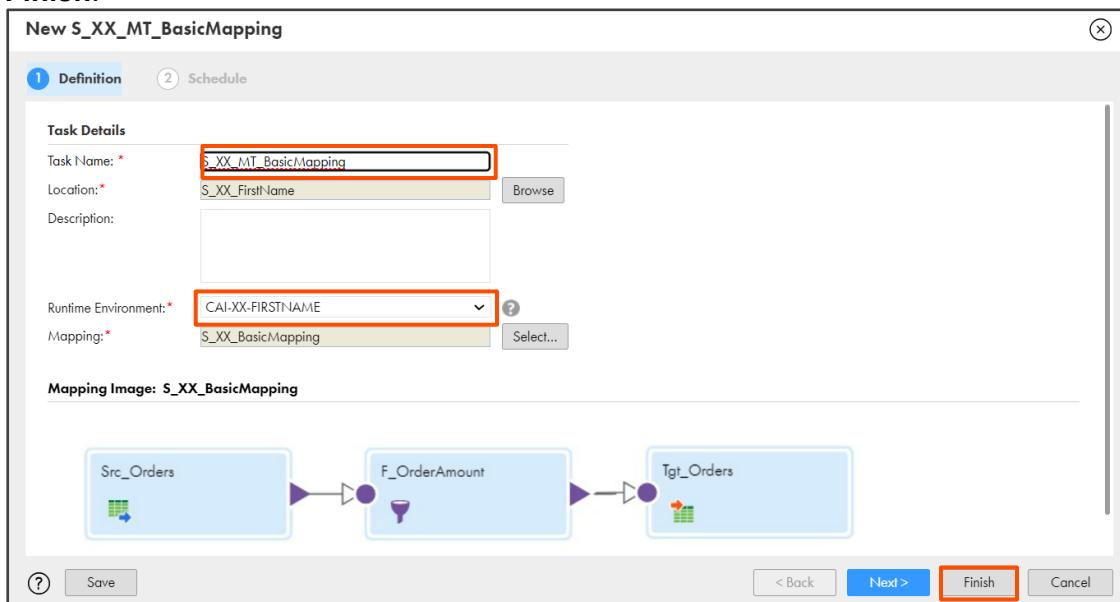
You can create a mapping in advanced mode when you want to process multilevel hierarchical data, embedded code snippets, and workloads at any scale. The Mapping Designer updates the mapping canvas to include the transformations and functions that enable advanced functionality. Mappings in advanced mode can run on advanced clusters in addition to the Secure Agent.

25. From the Actions menu, select **New Mapping Task**.



26. In the Task Name field, enter **S\_XX\_MT\_BasicMapping**.

27. From the Runtime Environment drop-down, select your secure agent group and click **Finish**.



**New S\_XX\_MT\_BasicMapping**

**1 Definition**    **2 Schedule**

**Task Details**

Task Name: \* **S\_XX\_MT\_BasicMapping**

Location: \* S\_XX\_FirstName

Description:

Runtime Environment: \* **CAI-XX-FIRSTNAME**

Mapping: \* **S\_XX\_BasicMapping**

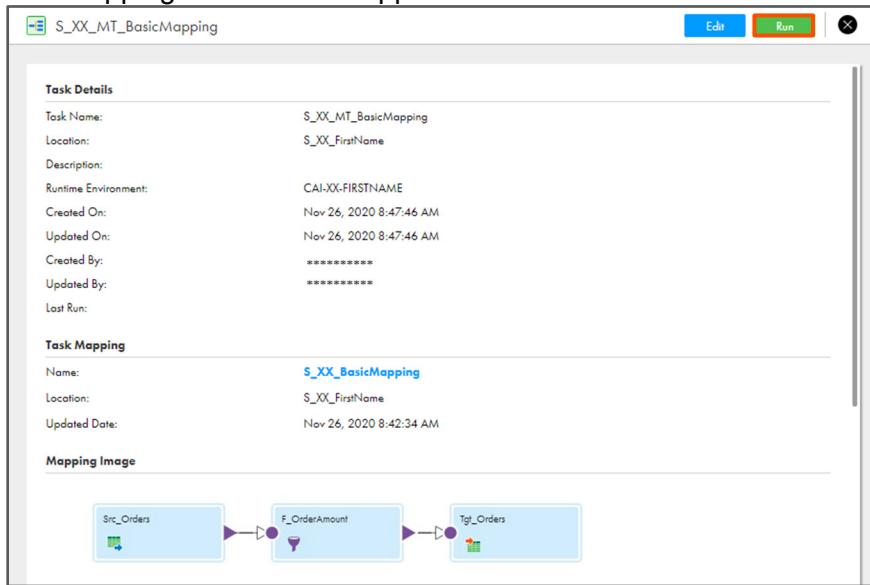
**Mapping Image: S\_XX\_BasicMapping**

```

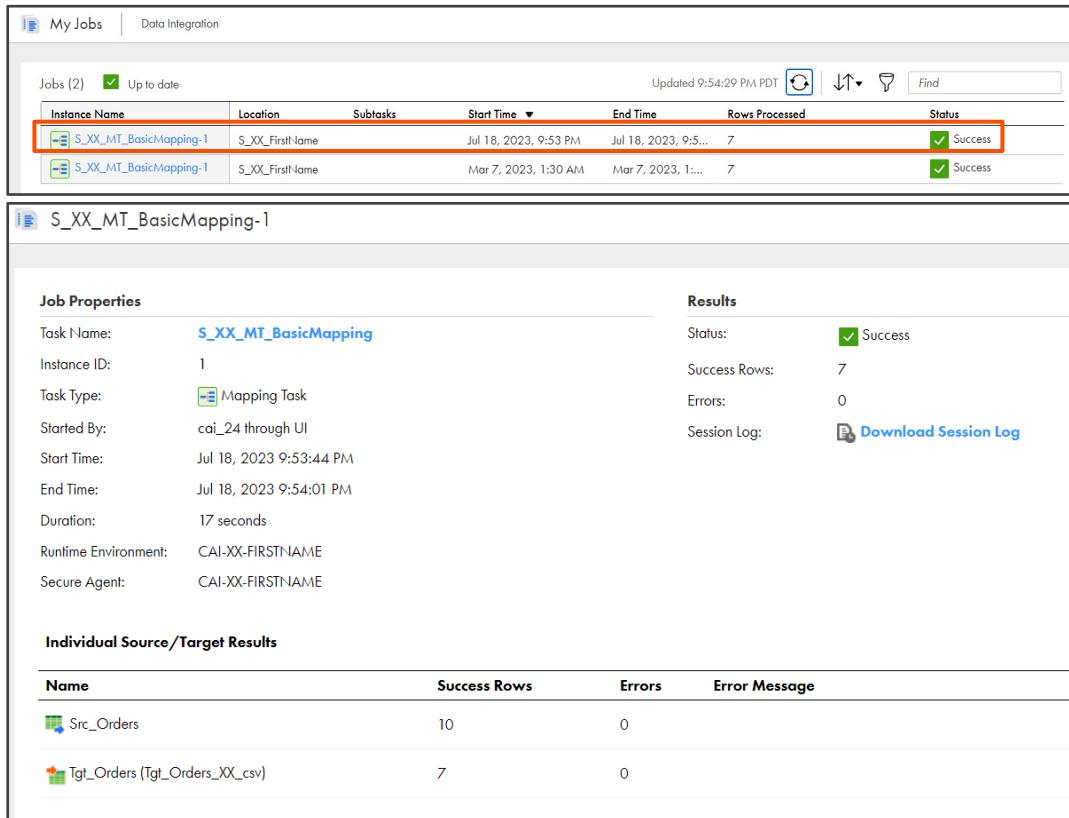
graph LR
    Src[Src_Orders] --> F[F_OrderAmount]
    F --> Tgt[Tgt_Orders]
  
```

**Finish**

28. The Mapping Task window appears. Click **Run**.



29. Navigate to **My Jobs** and click the task to check the results.



**My Jobs** | Data Integration

Jobs (2)  Up to date Updated 9:54:29 PM PDT    Find

| Instance Name          | Location       | Subtasks | Start Time            | End Time             | Rows Processed | Status                                      |
|------------------------|----------------|----------|-----------------------|----------------------|----------------|---|
| S_XX_MT_BasicMapping-1 | S_XX_FirstName |          | Jul 18, 2023, 9:53 PM | Jul 18, 2023, 9:5... | 7              | <input checked="" type="checkbox"/> Success |
| S_XX_MT_BasicMapping-1 | S_XX_FirstName |          | Mar 7, 2023, 1:30 AM  | Mar 7, 2023, 1:...   | 7              | <input checked="" type="checkbox"/> Success |

**S\_XX\_MT\_BasicMapping-1**

**Job Properties**

|                      |                         |
|----------------------|-------------------------|
| Task Name:           | S_XX_MT_BasicMapping    |
| Instance ID:         | 1                       |
| Task Type:           | Mapping Task            |
| Started By:          | cai_24 through UI       |
| Start Time:          | Jul 18, 2023 9:53:44 PM |
| End Time:            | Jul 18, 2023 9:54:01 PM |
| Duration:            | 17 seconds              |
| Runtime Environment: | CAI-XX-FIRSTNAME        |
| Secure Agent:        | CAI-XX-FIRSTNAME        |

**Results**

|               |   |
|---------------|---|
| Status:       | <input checked="" type="checkbox"/> Success         |
| Success Rows: | 7   |
| Errors:       | 0   |
| Session Log:  | <input type="button" value="Download Session Log"/> |

**Individual Source/Target Results**

| Name                           | Success Rows | Errors | Error Message |
|--------------------------------|--------------|--------|---------------|
| Src_Orders                     | 10           | 0      |               |
| Tgt_Orders (Tgt_Orders_XX_csv) | 7            | 0      |               |

---

This concludes the lab.

## Module 15: CAI and CDI Integration

### Lab 15-3: Invoke a Mapping in CAI

#### Overview:

In this lab, you will create a process to invoke a mapping in CAI. In the previous labs, you created a flat file connection, a basic mapping, and a mapping task. You will now utilize this mapping task in a Service step to invoke the mapping.

#### Objectives:

- Create a Process
- Use a service step and add a Cloud task
- Execute the Process

#### Duration:

20 Minutes

---

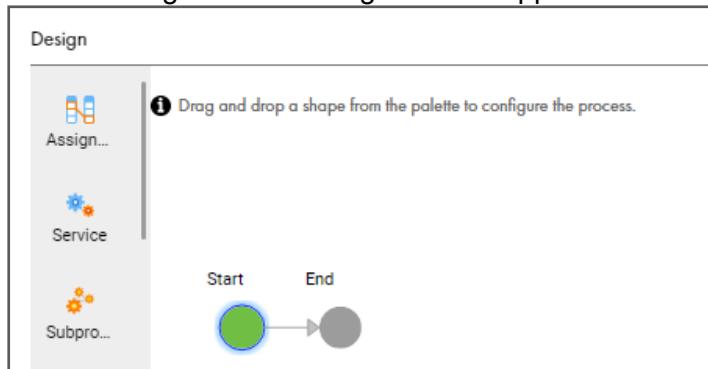
## Tasks

**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

#### Create a New Process

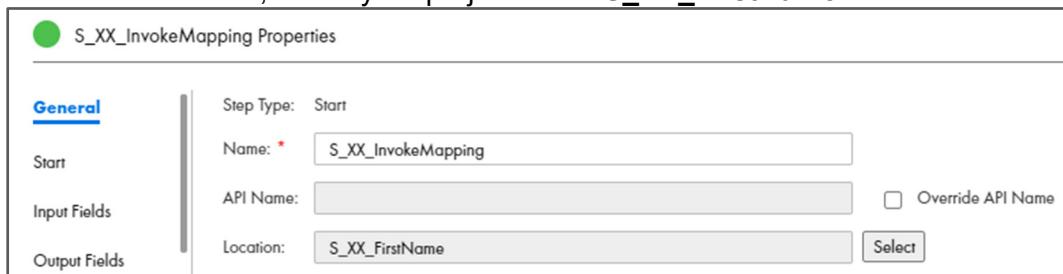
1. Switch to the **Application Integration** service and create a new **Process**.

The following Process Design canvas appears with the Start and End steps.



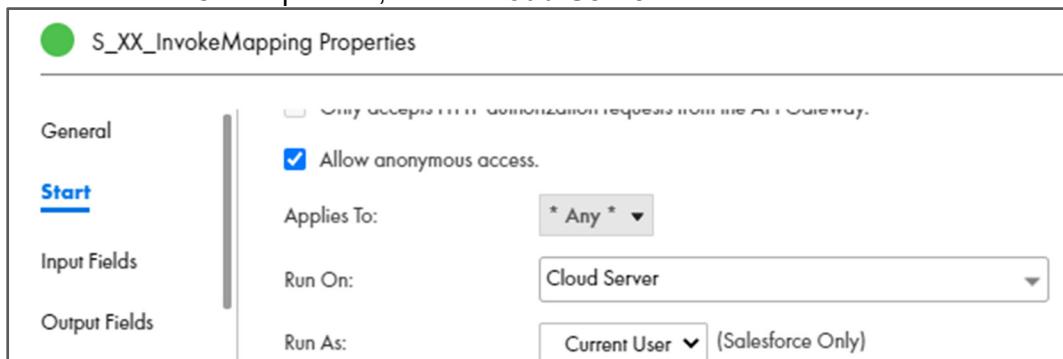
2. From the process canvas, select the **Start** step.
3. In the General tab, enter the name as **S\_XX\_InvokeMapping**.
4. Skip the API Name option.

5. In the Location field, select your project folder **S\_XX\_FirstName**.



| S_XX_InvokeMapping Properties |  |
|-------------------------------|--|
| <b>General</b>                | Step Type: Start<br>Name: * <input type="text" value="S_XX_InvokeMapping"/><br>API Name: <input type="text"/><br><input type="checkbox"/> Override API Name<br>Location: <input type="text" value="S_XX_FirstName"/> <input type="button" value="Select"/> |
| Start                         |  |
| Input Fields                  |  |
| Output Fields                 |  |

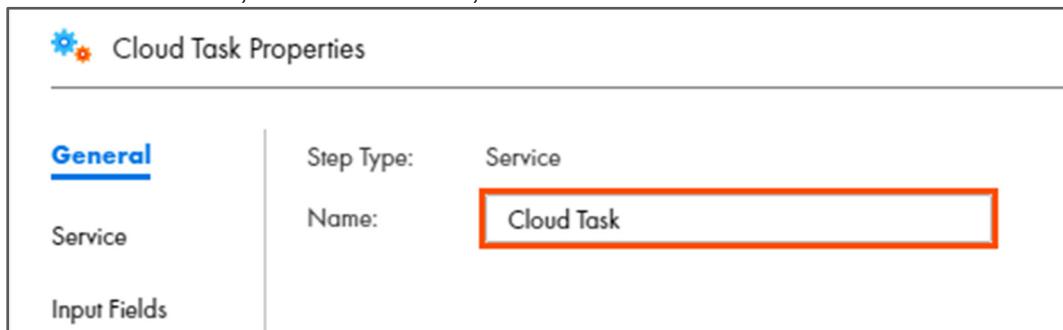
6. From the properties pane, select **Start**.  
 7. Select the **Allow anonymous access** option.  
 8. From the Run On drop-down, retain **Cloud Server**.



| S_XX_InvokeMapping Properties |  |
|-------------------------------|--|
| <b>General</b>                | <input type="checkbox"/> Only accept OAuth authorization requests from the API Gateway.  |
| <b>Start</b>                  | <input checked="" type="checkbox"/> Allow anonymous access.<br>Applies To: * Any * ▾<br>Run On: <input type="text" value="Cloud Server"/><br>Run As: <input type="text" value="Current User"/> (Salesforce Only) |
| Input Fields                  |  |
| Output Fields                 |  |

**Note:** The Allowed Users and Allowed Groups textboxes remain blank as you have selected the Allow anonymous access checkbox.

9. From the process canvas, drag and drop the **Service** step between the Start and the End step.  
 10. In the General tab, in the name field, enter **Cloud Task**.



| Cloud Task Properties |   |
|-----------------------|---|
| <b>General</b>        | Step Type: Service<br>Name: <input type="text" value="Cloud Task"/> |
| Service               |   |
| Input Fields          |   |

11. In the Service tab, from the Service Type drop-down, select **System Service** and from the Action drop-down, select **Run Cloud Task**.



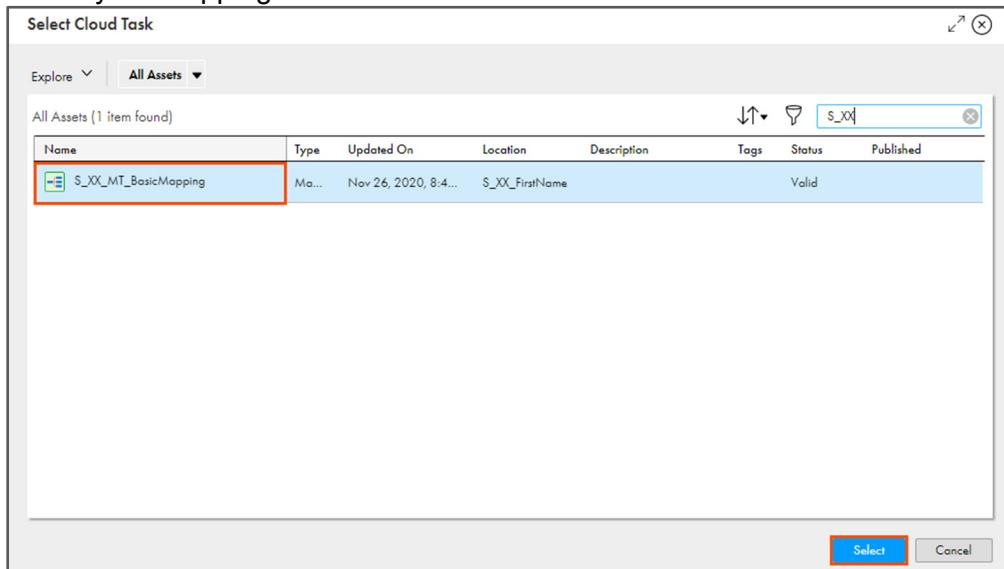
| Cloud Task Properties |   |
|-----------------------|---|
| <b>General</b>        | Service Type: <input type="text" value="System Service"/> |
| <b>Service</b>        | Action: <input type="text" value="Run Cloud Task"/>       |
| Input Fields          | Description   |

12. In the Input Fields tab, for the Task Name field, click **Select**.



| Name                      | Required                            | Value  |
|---------------------------|-------------------------------------|--|
| Task Name                 | <input checked="" type="checkbox"/> | Content <input type="button" value="Select..."/> |
| Wait for Task to Complete | <input type="checkbox"/>            | Content <input type="checkbox"/>                 |

13. Select your mapping task and click **Select**.



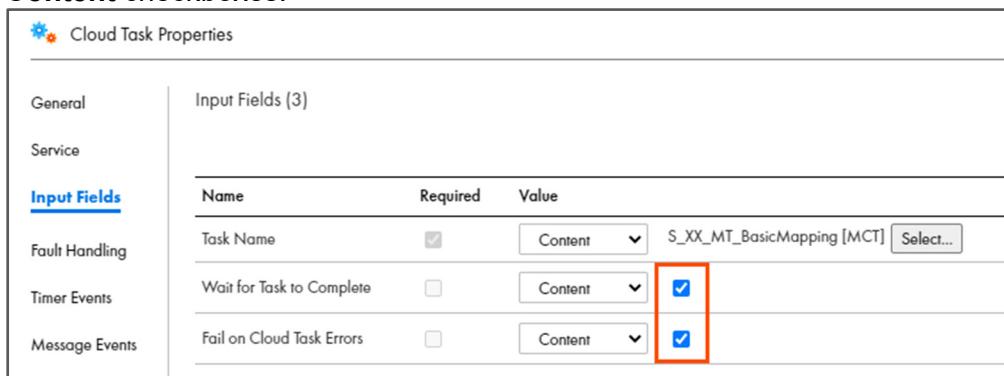
Select Cloud Task

All Assets (1 item found)

| Name                 | Type  | Updated On           | Location       | Description | Tags | Status | Published |
|----------------------|-------|----------------------|----------------|-------------|------|--------|-----------|
| S_XX_MT_BasicMapping | Ma... | Nov 26, 2020, 8:4... | S_XX_FirstName |             |      | Valid  |           |

Select Cancel

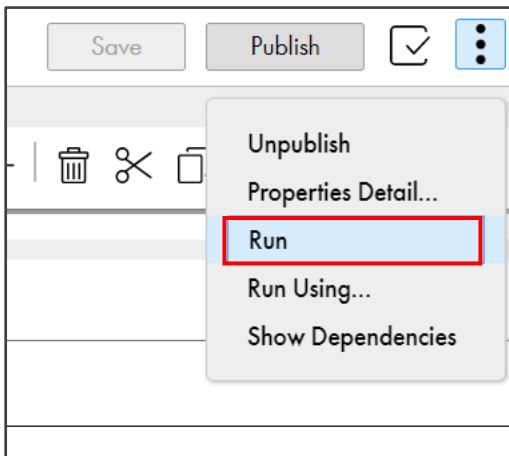
14. For the **Wait for Task to Complete** and **Fail on Cloud Task Errors** fields, select the **Content** checkboxes.



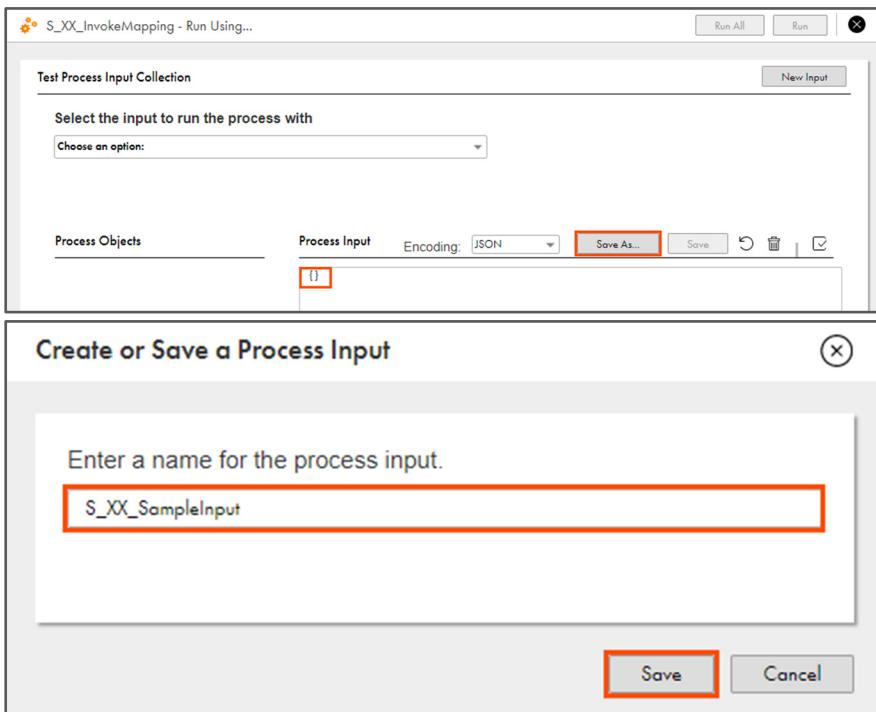
| Name                      | Required                            | Value   |
|---------------------------|-------------------------------------|---|
| Task Name                 | <input checked="" type="checkbox"/> | Content <input type="button" value="S_XX_MT_BasicMapping [MCT] Select..."/> |
| Wait for Task to Complete | <input type="checkbox"/>            | Content <input checked="" type="checkbox"/>                                 |
| Fail on Cloud Task Errors | <input type="checkbox"/>            | Content <input checked="" type="checkbox"/>                                 |

15. **Save** and **Publish** the process.

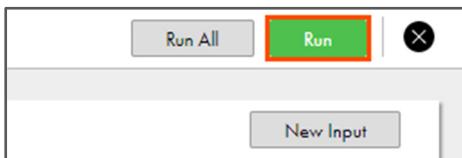
16. To execute the process, click **Run**.



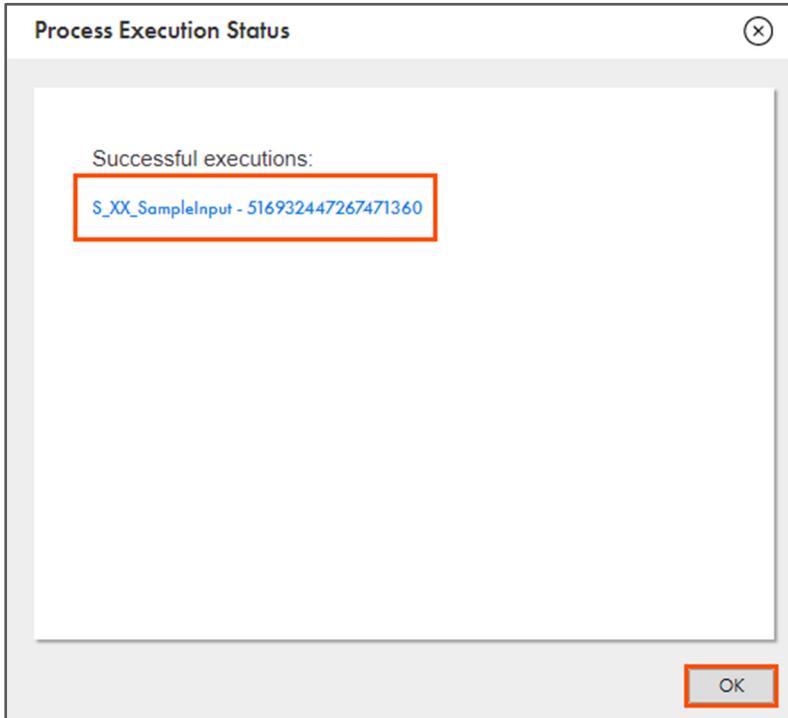
17. To save the sample input, click **Save As**.



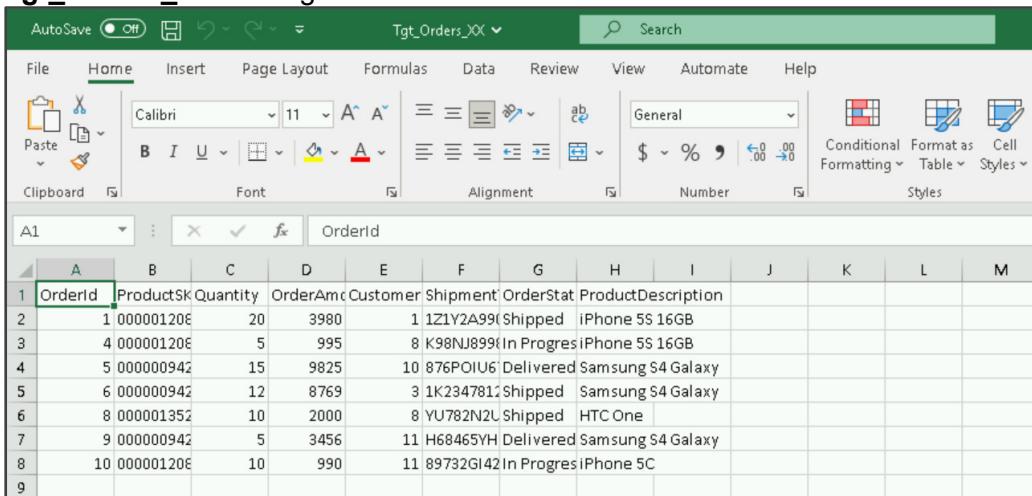
18. Run the process.



19. Wait for a couple of seconds for the process to finish execution. After the process executes, from the Process Execution Status window, click **OK**.



20. On your system's desktop, in the CAI Lab Prep Files folder, observe that the **Tgt\_Orders\_XX** file is generated.



The screenshot shows a Microsoft Excel spreadsheet titled "Tgt\_Orders\_XX". The table has the following data:

|   | OrderId      | ProductSk | Quantity | OrderAmt     | Customer    | Shipment          | OrderStat | ProductDescription |
|---|--------------|-----------|----------|--------------|-------------|-------------------|-----------|--------------------|
| 1 | 1 00000120E  | 20        | 3980     | 1 1ZLY2A99   | Shipped     | iPhone 5S 16GB    |           |                    |
| 2 | 4 00000120E  | 5         | 995      | 8 K98NJJ899  | In Progress | iPhone 5S 16GB    |           |                    |
| 3 | 5 000000942  | 15        | 9825     | 10 876POIU6  | Delivered   | Samsung S4 Galaxy |           |                    |
| 4 | 6 000000942  | 12        | 8769     | 3 1K2347811  | Shipped     | Samsung S4 Galaxy |           |                    |
| 5 | 8 000001352  | 10        | 2000     | 8 YU782N2U   | Shipped     | HTC One           |           |                    |
| 6 | 9 000000942  | 5         | 3456     | 11 H68465YH  | Delivered   | Samsung S4 Galaxy |           |                    |
| 7 | 10 00000120E | 10        | 990      | 11 89732GI42 | In Progress | iPhone 5C         |           |                    |
| 8 |              |           |          |              |             |                   |           |                    |
| 9 |              |           |          |              |             |                   |           |                    |

---

*This concludes the lab.*

# Module 17: CAI Live Project Walk-Through

## Lab 17-1: CAI Live Project- Order Management

### Overview:

In this lab, you will use the IICS Application Integration service to create a process that fetches inventory details and enables you to obtain margin and commission details. You will also create a managed API using the API Manager Service.

### Objectives:

- Create and invoke the Order Management Process
- Connect to a Margin Service
- Modify the Process to add the Margin Details Service
- Create a Service Connector to the Email Service
- Create an Email Connection
- Create a process that uses the email connection to send emails
- Invoke the Order Management Process
- Enable Fault Handling
- Create a Managed API

### Duration:

90 Minutes

---

### Tasks

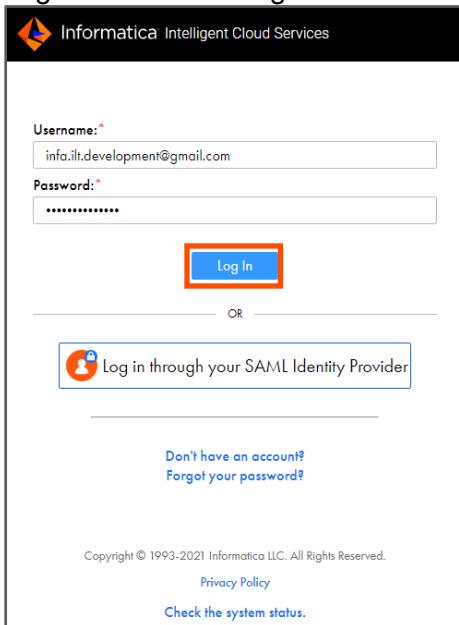
**Note:** The screen on your lab environment and the images in the lab guide may differ in folder names, asset names, and asset numbers. However, the steps remain the same.

#### Import a Service Connector and Connection

1. In a web browser, enter the following URL:  
<https://network.informatica.com/docs/DOC-17733>
2. Download the **Get Inventory Detail Connector.zip** file and save it on your computer.

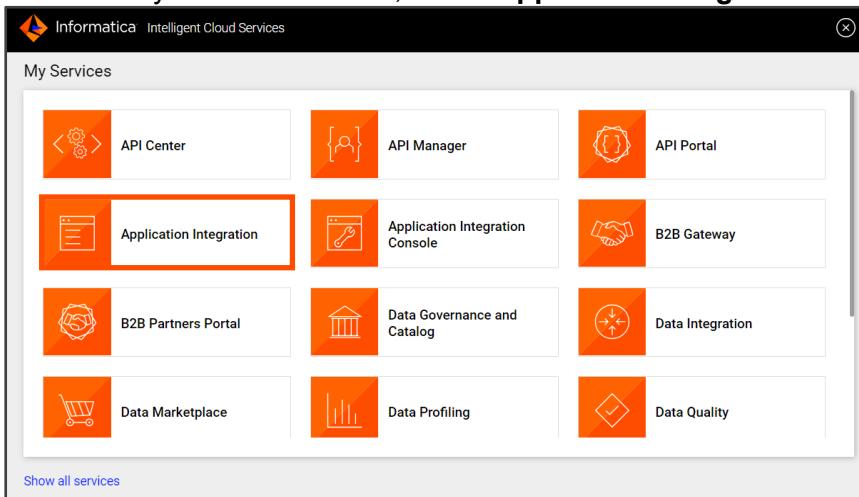
**Note:** You can also find the Get Inventory Detail Connector.zip file in the **CAI Lab Prep Files** folder.

3. Log in to the IICS Org.

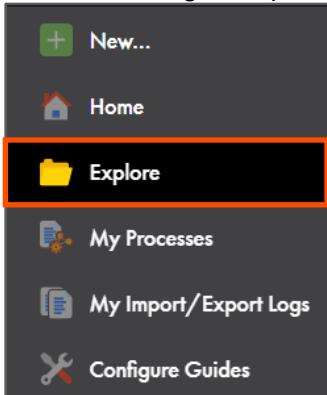


The screenshot shows the Informatica Intelligent Cloud Services login interface. It features a top navigation bar with the Informatica logo and the text "Informatica Intelligent Cloud Services". Below this is a form with fields for "Username:" (containing "info.ilt.development@gmail.com") and "Password:". A large blue "Log In" button is centered below the password field, with a red box drawn around it. To the right of the log in button is the text "OR". Below "OR" is a blue rectangular button labeled "Log in through your SAML Identity Provider" with a user icon. At the bottom of the page, there are links for "Don't have an account?", "Forgot your password?", and copyright information: "Copyright © 1993-2021 Informatica LLC. All Rights Reserved.", "Privacy Policy", and "Check the system status."

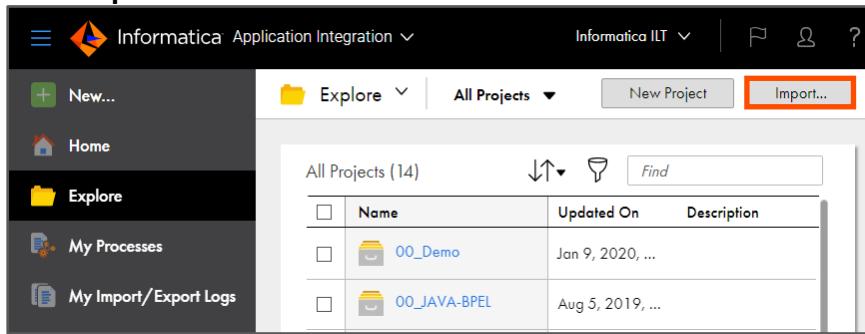
4. From the My Services window, select **Application Integration**.



5. From the navigation pane, select **Explore**.



6. Click **Import**.



7. Click **Choose File**.

8. Navigate to the location where you have saved the Get Inventory Detail Connector.zip file.

9. Select **Get Inventory Detail Connector.zip** and click **Open**.

The following assets are available for import:

- Get Inventory Detail (a service connector)
- Get-Inventory-Detail (a connection)

| ▼ Select Assets                     |           |              |                |                         |   |   |
|-------------------------------------|-----------|--------------|----------------|-------------------------|---|---|
| Assets (2) All selected ▾           |           |              |                |                         |   |   |
| <input checked="" type="checkbox"/> | Name      | Dependencies | Type           | Location                | Description                                 | Status  |
| <input checked="" type="checkbox"/> | Get In... | 0            | Service Con... | Order Management\Shared | Connector to interact with InventoryService | <input checked="" type="checkbox"/> Overwrite existi... |
| <input checked="" type="checkbox"/> | Get-In... | 1            | App Conne...   | Order Management\Shared |   | <input checked="" type="checkbox"/> Overwrite existi... |

10. Scroll down and select **Publish Application Integration assets after import**.

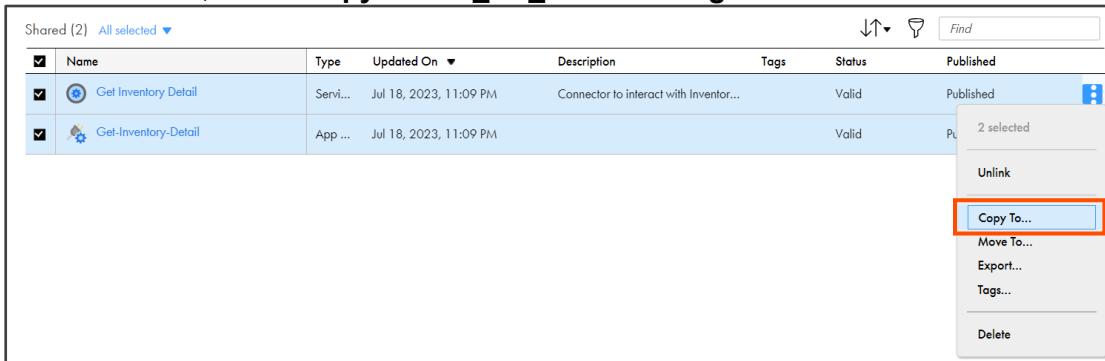
|   |
|---|
| ▼ Review Locations  |
| The selected assets will be imported into the following projects. You can accept the default values or select different target locations. |
| Source Project 1: Order Management      Target Project 1: Order Management -- Default   |
| ▼ Publish Options   |
| <input checked="" type="checkbox"/> Publish Application Integration assets after import   |

11. Click **Import**.

|   |  |   |
|---|--|---|
| • Import Assets   | Test                                   | Import  |
| Start an import job, review the assets from the import file, and resolve any error related to location, connections, or runtime environments. |  |   |
| ▼ Select Import File  |  |   |
| Filename:*  | Get Inventory Detail Connector (4).zip | X Choose File...  |
| ▼ Specify Import Job Details  |  |   |
| Define the import behavior when an asset already exists, and provide a name that will be used to identify the import job on the My Jobs page. |  |   |
| Job Name:*  | Get Inventory Detail Connector         | <input checked="" type="checkbox"/> Overwrite existing assets, excluding connections and runtime environments |

12. In the Explore tab, create Project **S\_XX\_Order Management**.

13. Go to the folder **Order Management > Shared** and select **Get Inventory Detail** and **Get-Inventory-Detail**.
14. From the menu, select **Copy To > S\_XX\_Order Management**.

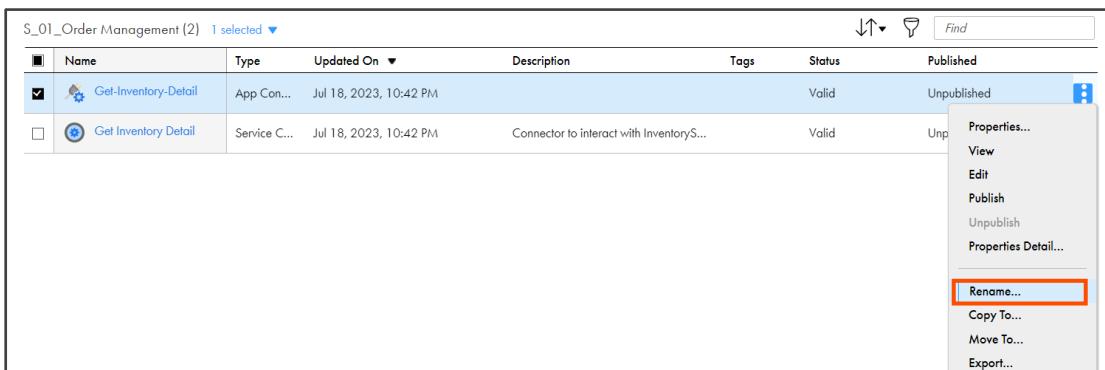


| Shared (2) All selected ▾           |                      |           |                        |   |       |           |   |
|-------------------------------------|----------------------|-----------|------------------------|---|-------|-----------|---|
|                                     | Name                 | Type      | Updated On ▾           | Description                             | Tags  | Status    | Published   |
| <input checked="" type="checkbox"/> | Get Inventory Detail | Service   | Jul 18, 2023, 11:09 PM | Connector to interact with Inventory... | Valid | Published |  |
| <input checked="" type="checkbox"/> | Get-Inventory-Detail | App Conn. | Jul 18, 2023, 11:09 PM |   | Valid | Published |  |

2 selected

- Unlink
- Copy To...** Copy To...
- Move To...
- Export...
- Tags...
- 
- Delete

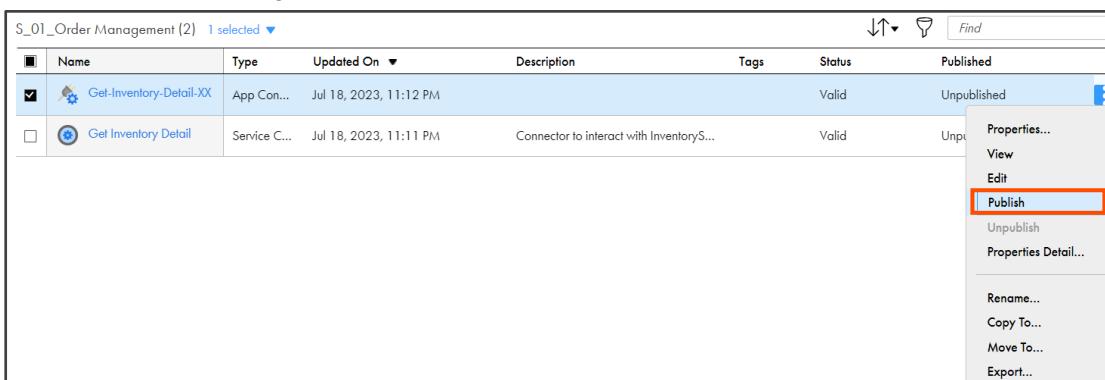
15. Rename the **Get-Inventory-Detail** connection to **Get-Inventory-Detail-XX**.



| S_01_Order Management (2) 1 selected ▾ |                      |               |                        |  |       |             |   |
|--|----------------------|---------------|------------------------|--|-------|-------------|---|
|  | Name                 | Type          | Updated On ▾           | Description                              | Tags  | Status      | Published   |
| <input checked="" type="checkbox"/>    | Get-Inventory-Detail | App Conn.     | Jul 18, 2023, 10:42 PM |  | Valid | Unpublished |  |
| <input type="checkbox"/>               | Get Inventory Detail | Service Conn. | Jul 18, 2023, 10:42 PM | Connector to interact with InventoryS... | Valid | Unpublished |  |

- Properties...
- View
- Edit
- Publish
- Unpublish
- Properties Detail...
- Rename...** Rename...
- Copy To...
- Move To...
- Export...

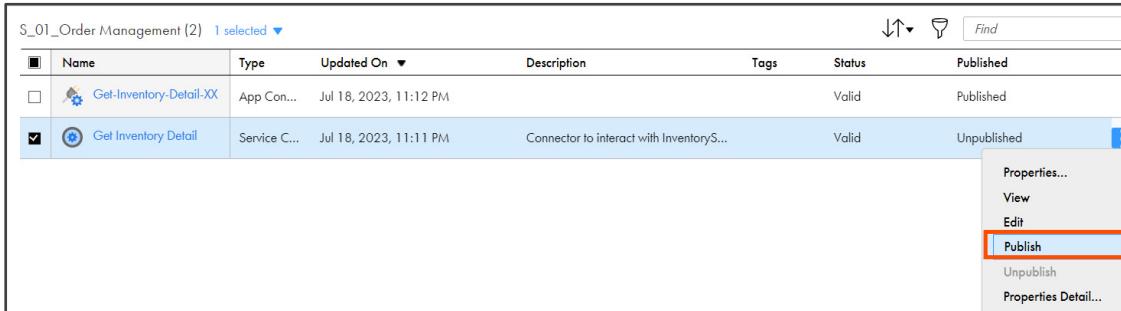
16. Select **Get-Inventory-Detail-XX**, and from the Actions menu, select **Publish**.



| S_01_Order Management (2) 1 selected ▾ |                         |               |                        |  |       |             |   |
|--|-------------------------|---------------|------------------------|--|-------|-------------|---|
|  | Name                    | Type          | Updated On ▾           | Description                              | Tags  | Status      | Published   |
| <input checked="" type="checkbox"/>    | Get-Inventory-Detail-XX | App Conn.     | Jul 18, 2023, 11:12 PM |  | Valid | Unpublished |  |
| <input type="checkbox"/>               | Get Inventory Detail    | Service Conn. | Jul 18, 2023, 11:11 PM | Connector to interact with InventoryS... | Valid | Unpublished |  |

- Properties...
- View
- Edit
- Publish** Publish
- Unpublish
- Properties Detail...
- Rename...
- Copy To...
- Move To...
- Export...

17. Select the **Get Inventory Detail** service connector, and from the Actions menu, select **Publish**.



| Name                    | Type         | Updated On             | Description                              | Tags | Status | Published   |
|-------------------------|--------------|------------------------|--|------|--------|-------------|
| Get-Inventory-Detail-XX | App Con...   | Jul 18, 2023, 11:12 PM |  |      | Valid  | Published   |
| Get Inventory Detail    | Service C... | Jul 18, 2023, 11:11 PM | Connector to interact with InventoryS... |      | Valid  | Unpublished |

A message '**The asset [Get Inventory Detail] was published successfully.**' appears.

18. Open **Get-Inventory-Detail-XX** connection. Verify the connection details and **Save** the connection if it is not saved already.

19. **Publish** the **Get-Inventory-Detail-XX** again.



Type: \* S\_xx\_Order Management > Get Inventory Detail

Run On: \* Cloud Server or any Secure Agent

Connection Test: Not Supported

OData-Enabled: Not Supported

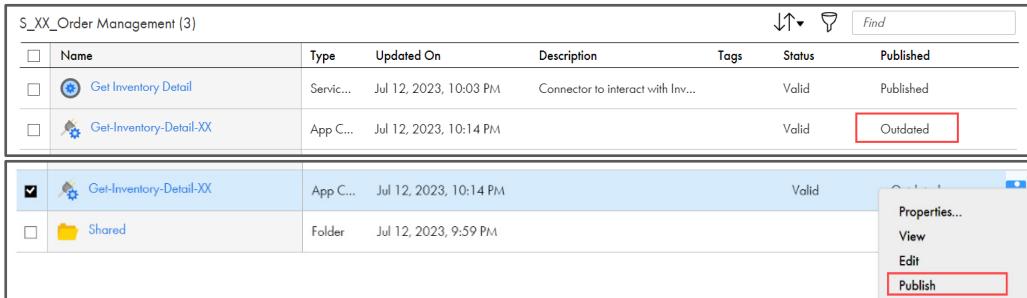
**Note:** Get Inventory Detail must be populated in the Type field.

20. Close the asset from the navigation pane.



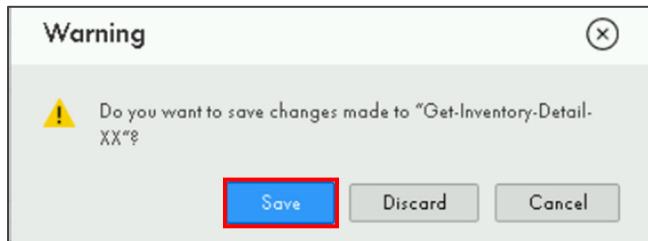
| Name                    | Type       | Updated On           | Description                       | Tags | Status | Published |
|-------------------------|------------|----------------------|-----------------------------------|------|--------|-----------|
| Get-Inventory-Detail-XX | App C...   | Aug 3, 2021, 8:39 AM |                                   |      | Valid  | Published |
| Get Inventory Detail    | Service... | Aug 2, 2021, 9:57 PM | Connector to interact with Inv... |      | Valid  | Published |

**Note:** If the status of **Get-Inventory-Detail-XX** is outdated, click ellipses, and select **Publish**.



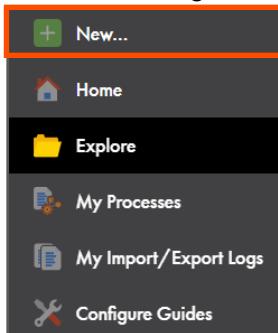
| Name                    | Type       | Updated On             | Description                       | Tags | Status | Published |
|-------------------------|------------|------------------------|-----------------------------------|------|--------|-----------|
| Get Inventory Detail    | Service... | Jul 12, 2023, 10:03 PM | Connector to interact with Inv... |      | Valid  | Published |
| Get-Inventory-Detail-XX | App C...   | Jul 12, 2023, 10:14 PM |                                   |      | Valid  | Outdated  |

If the Warning window pops-up, click **Save**.

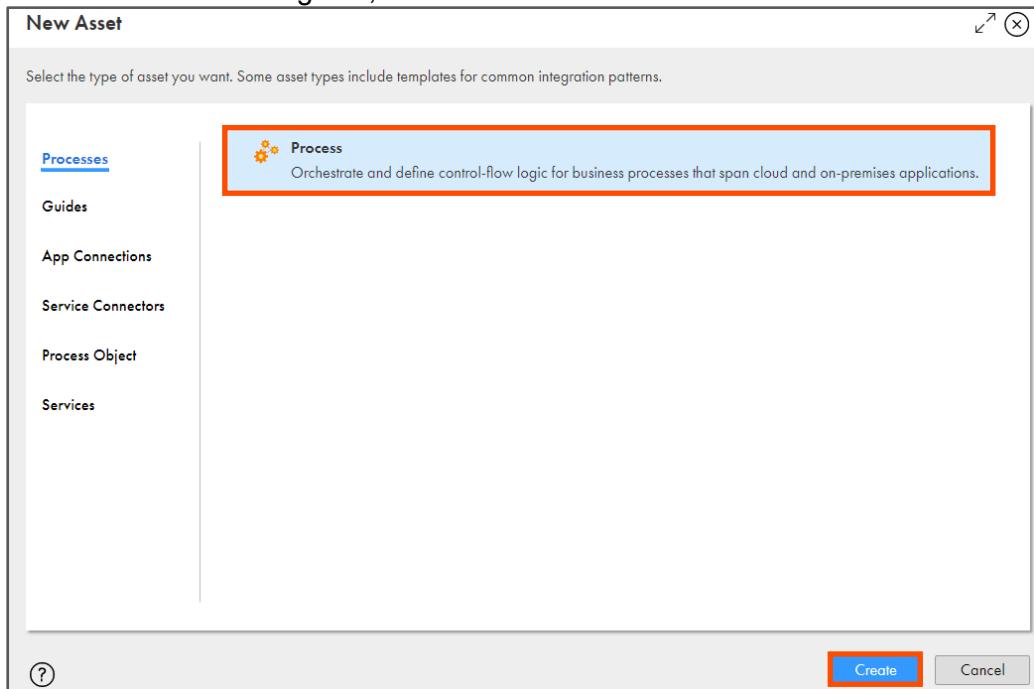


## Create a New Process

21. From the Navigation pane, click **New**.

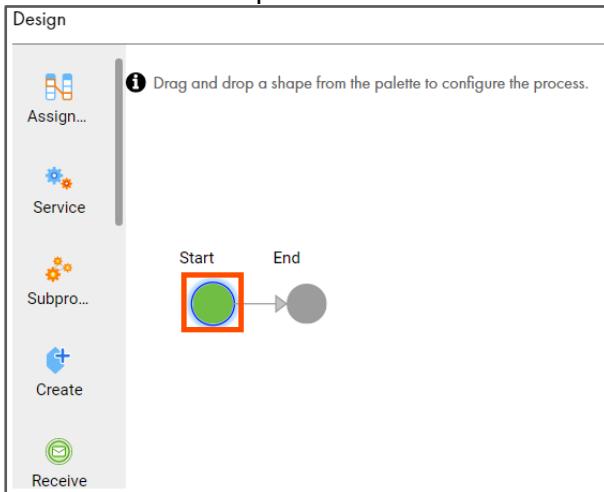


22. In the New Asset dialog box, select **Processes > Process** and click **Create**.



Process Designer opens with a process template.

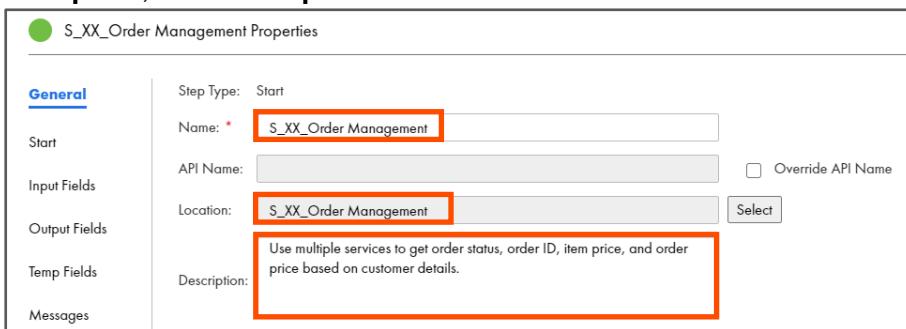
23. Select the **Start** step.



24. In the General tab, in the Name field, enter **S\_XX\_Order Management**.

25. In the Location field, select your project folder **S\_XX\_Order Management**.

26. In the Description field, enter **Use multiple services to get order status, order ID, item price, and order price based on customer details.**

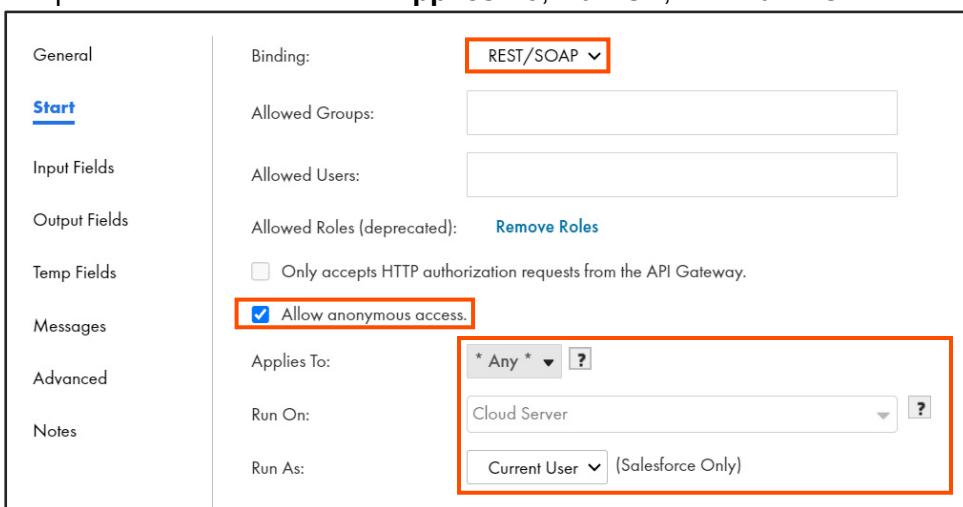


| S_XX_Order Management Properties |   |
|----------------------------------|---|
| <b>General</b>                   | Step Type: Start<br>Name: * <b>S_XX_Order Management</b><br>API Name: <input type="text"/><br>Location: <b>S_XX_Order Management</b> <input type="button" value="Select"/><br>Description: <b>Use multiple services to get order status, order ID, item price, and order price based on customer details.</b> |
| Start                            |   |
| Input Fields                     |   |
| Output Fields                    |   |
| Temp Fields                      |   |
| Messages                         |   |

27. In the Start tab, for the Binding field, select **REST/SOAP**.

28. For Allowed Roles, select the **Allow anonymous access** option.

29. Keep the default selection for **Applies To**, **Run On**, and **Run As**.



|               |   |
|---------------|---|
| General       | Binding: <b>REST/SOAP</b>   |
| <b>Start</b>  | Allowed Groups: <input type="text"/>  |
| Input Fields  | Allowed Users: <input type="text"/>   |
| Output Fields | Allowed Roles (deprecated): <a href="#">Remove Roles</a>                                |
| Temp Fields   | <input type="checkbox"/> Only accepts HTTP authorization requests from the API Gateway. |
| Messages      | <input checked="" type="checkbox"/> Allow anonymous access.                             |
| Advanced      | Applies To: <b>* Any *</b>  |
| Notes         | Run On: <b>Cloud Server</b>   |
|               | Run As: <b>Current User</b> (Salesforce Only)   |

30. In the Input Fields tab, to add a new field, click .

31. Add the following fields:

| Name          | Type    | Required |
|---------------|---------|----------|
| CustomerName  | Text    | Yes      |
| CustomerEmail | Text    | Yes      |
| ItemName      | Text    | Yes      |
| ItemCount     | Integer | Yes      |

| General<br><br>Start<br><br><u><b>Input Fields</b></u><br><br>Output Fields<br><br>Temp Fields<br><br>Messages<br><br>Advanced | Input Format: <input checked="" type="radio"/> Fields <input type="radio"/> Whole payload<br><br>Input Fields (4)<br><table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Description</th> <th>Required</th> </tr> </thead> <tbody> <tr> <td>CustomerName</td> <td>Text</td> <td></td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>CustomerEmail</td> <td>Text</td> <td></td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>ItemName</td> <td>Text</td> <td></td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>ItemCount</td> <td>Integer</td> <td></td> <td><input checked="" type="checkbox"/></td> </tr> </tbody> </table> | Name        | Type                                | Description | Required | CustomerName | Text |  | <input checked="" type="checkbox"/> | CustomerEmail | Text |  | <input checked="" type="checkbox"/> | ItemName | Text |  | <input checked="" type="checkbox"/> | ItemCount | Integer |  | <input checked="" type="checkbox"/> |
|--|---|-------------|-------------------------------------|-------------|----------|--------------|------|--|-------------------------------------|---------------|------|--|-------------------------------------|----------|------|--|-------------------------------------|-----------|---------|--|-------------------------------------|
| Name   | Type  | Description | Required                            |             |          |              |      |  |                                     |               |      |  |                                     |          |      |  |                                     |           |         |  |                                     |
| CustomerName   | Text  |             | <input checked="" type="checkbox"/> |             |          |              |      |  |                                     |               |      |  |                                     |          |      |  |                                     |           |         |  |                                     |
| CustomerEmail  | Text  |             | <input checked="" type="checkbox"/> |             |          |              |      |  |                                     |               |      |  |                                     |          |      |  |                                     |           |         |  |                                     |
| ItemName   | Text  |             | <input checked="" type="checkbox"/> |             |          |              |      |  |                                     |               |      |  |                                     |          |      |  |                                     |           |         |  |                                     |
| ItemCount  | Integer   |             | <input checked="" type="checkbox"/> |             |          |              |      |  |                                     |               |      |  |                                     |          |      |  |                                     |           |         |  |                                     |

**Note:** You enter values for the above Input Fields at runtime when you invoke the process.

32. In the Output Fields tab, to add a new field, click .

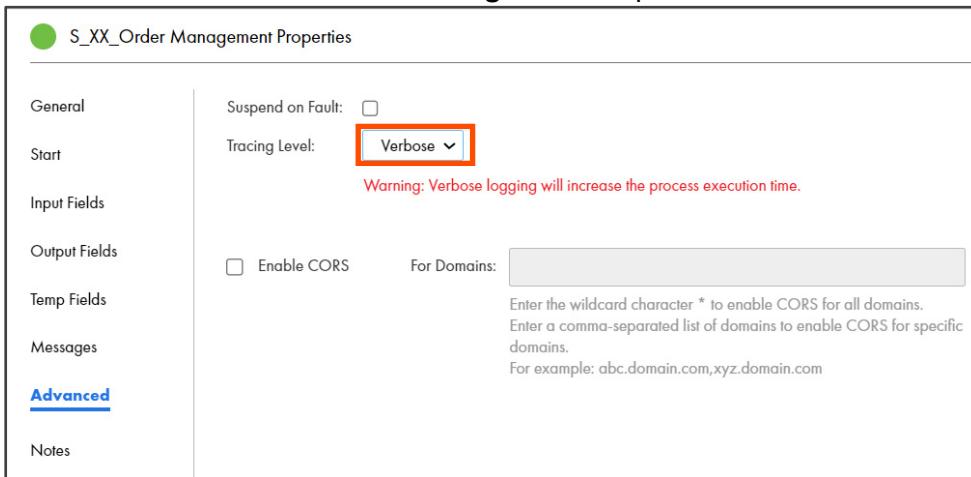
33. Add the following fields:

| Name       | Type              |
|------------|-------------------|
| Status     | Text              |
| OrderID    | Text              |
| ItemPrice  | More types>Number |
| OrderPrice | Number            |

| General<br><br>Start<br><br>Input Fields<br><br><u><b>Output Fields</b></u><br><br>Temp Fields<br><br>Messages<br><br>Advanced | Output Format: <input checked="" type="radio"/> Fields <input type="radio"/> Whole payload<br><br>Output Fields (4)<br><table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Name</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>Status</td> <td>Text</td> </tr> <tr> <td>OrderID</td> <td>Text</td> </tr> <tr> <td>ItemPrice</td> <td>Number</td> </tr> <tr> <td>OrderPrice</td> <td>Number</td> </tr> </tbody> </table> | Name | Type | Status | Text | OrderID | Text | ItemPrice | Number | OrderPrice | Number |
|--|---|------|------|--------|------|---------|------|-----------|--------|------------|--------|
| Name   | Type  |      |      |        |      |         |      |           |        |            |        |
| Status   | Text  |      |      |        |      |         |      |           |        |            |        |
| OrderID  | Text  |      |      |        |      |         |      |           |        |            |        |
| ItemPrice  | Number  |      |      |        |      |         |      |           |        |            |        |
| OrderPrice   | Number  |      |      |        |      |         |      |           |        |            |        |

**Note:** This is the output you see after you invoke the process.

34. In the Advanced tab, from the Tracing Level drop-down, select **Verbose**.



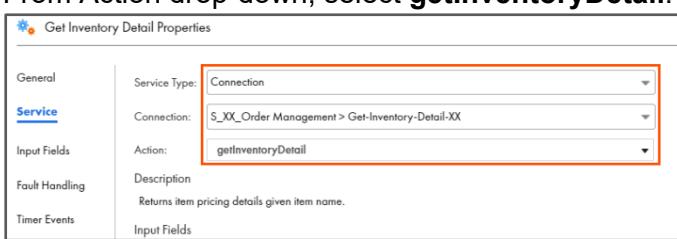
**Note:** you will notice a warning Verbose logging will increase the process execution time. Ignore this warning. When the Tracing Level is Verbose, the Application Integration Console logs all the steps. This is useful if you need to debug the process.

## Get Inventory Details

35. Between the Start and End steps, drag and drop a **Service** step.



36. Select the **Service** step.  
 37. In the General tab, in the Name field, enter **Get Inventory Detail**.  
 38. In the Service tab, from the Service Type drop-down, select **Connection**.  
 39. From the Connection drop-down, select **S\_XX\_Order Management > Get-Inventory-Detail-XX**.  
 40. From Action drop-down, select **getInventoryDetail**.

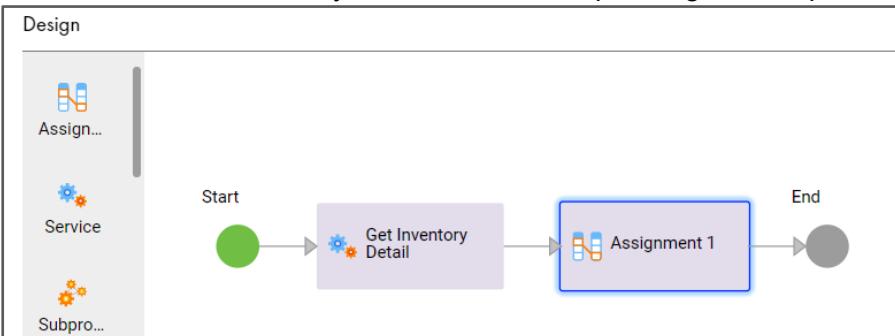


41. In the Input Fields tab, click the plus (+) icon. Select **Item Name**.

42. From the Value drop-down, select **Field > ItemName**.

| Input Fields   | Name      | Required                 | Value   |
|----------------|-----------|--------------------------|---|
| Fault Handling | Item Name | <input type="checkbox"/> | Field<br><input style="border: none; background-color: #f0f0f0; padding: 0 5px;" type="button" value="..."/> ItemName <input style="border: none; background-color: #f0f0f0; padding: 0 5px;" type="button" value="..."/> |

43. Between the Get Inventory Detail and End steps, drag and drop an **Assignment** step.



44. Select the **Assignment** step.

38. In the General tab, in the Name field, enter **Assign Order Details**.

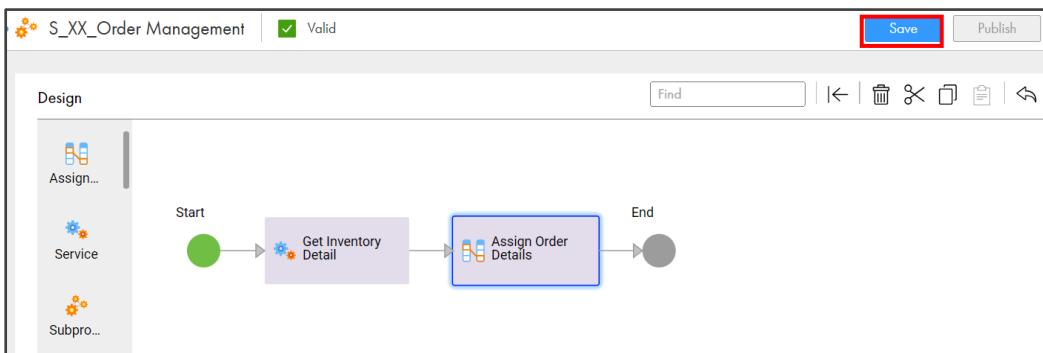
|                |                                   |
|----------------|-----------------------------------|
| <b>General</b> | Step Type: Assignment             |
| Assignments    | Name: <b>Assign Order Details</b> |

39. In the **Assignments** tab, assign values to the following output fields:

| Target     | Value Type | Value                                |
|------------|------------|--------------------------------------|
| Status     | Content    | Your order is accepted.              |
| OrderID    | Formula    | infa:getProcessId()                  |
| ItemPrice  | Field      | item>sellprice                       |
| OrderPrice | Formula    | \$output.ItemPrice*\$input.ItemCount |

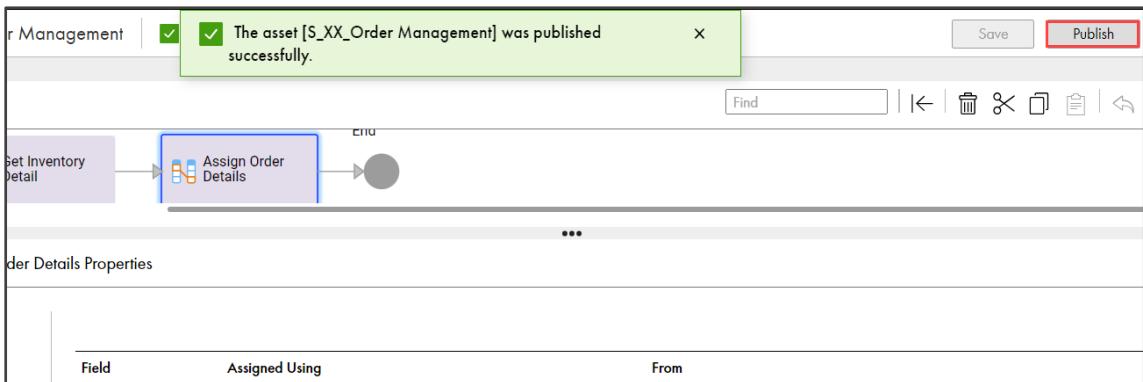
| Field      | Assigned Using | From   |
|------------|----------------|--|
| Status     | Content        | Your order is accepted <input style="border: none; background-color: #f0f0f0; padding: 0 5px;" type="button" value="..."/>               |
| OrderID    | Formula        | infa:getProcessId() <input style="border: none; background-color: #f0f0f0; padding: 0 5px;" type="button" value="..."/>                  |
| ItemPrice  | Field          | Item > sellprice <input style="border: none; background-color: #f0f0f0; padding: 0 5px;" type="button" value="..."/>                     |
| OrderPrice | Formula        | \$output.ItemPrice*\$input.ItemCount <input style="border: none; background-color: #f0f0f0; padding: 0 5px;" type="button" value="..."/> |

40. Click **Save**.

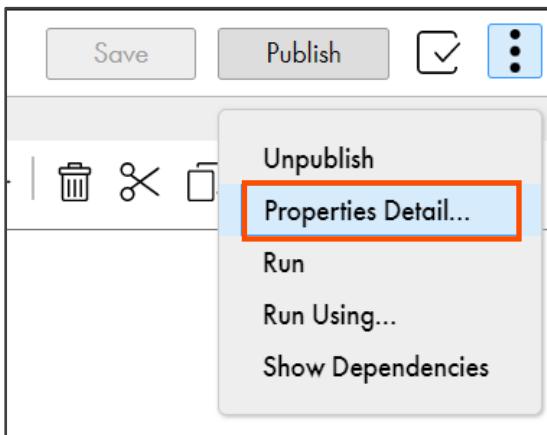


A message '**The process was saved successfully**' appears.

41. Click **Publish**.

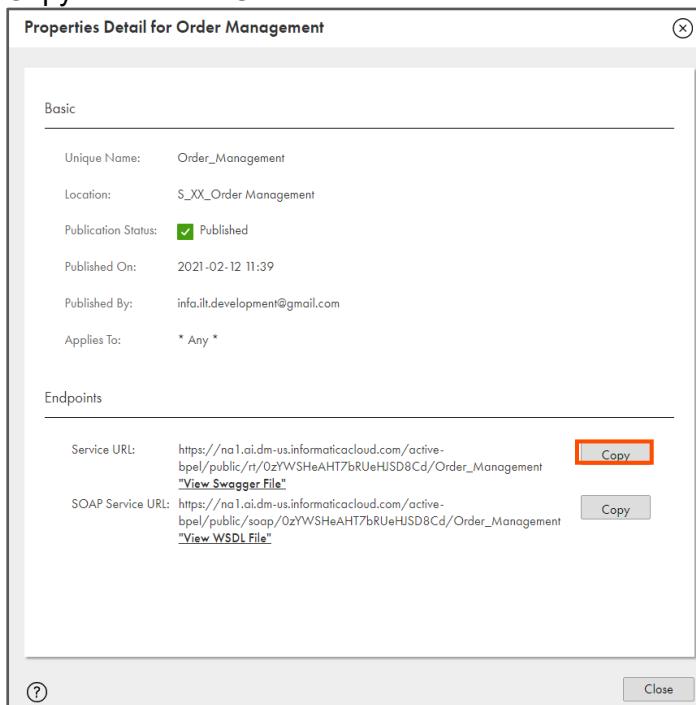


42. Click **Actions > Properties Detail**.



The Properties Detail for Order Management window appears.

43. Copy the service URL.



44. Open a notepad file, paste the Service URL, and add the following input parameters to the end of service URL:

**<Service  
URL>?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&ItemName=item1&ItemCount=2**

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **29\_LabGuide\_CAI\_Live Project\_17-1**. Copy the command mentioned under **Step A**.

The following URL is a sample service URL with input parameters:

**https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/0zYWSHeAHT7bRUeHJSD8Cd/Order\_Management?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&ItemName=item1&ItemCount=2**

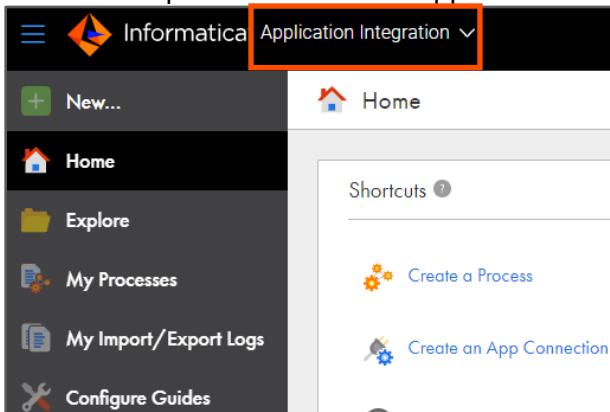
45. Open a browser and paste the service URL with input parameters.

46. Enter your Informatica Intelligent Cloud Services credentials, if prompted.

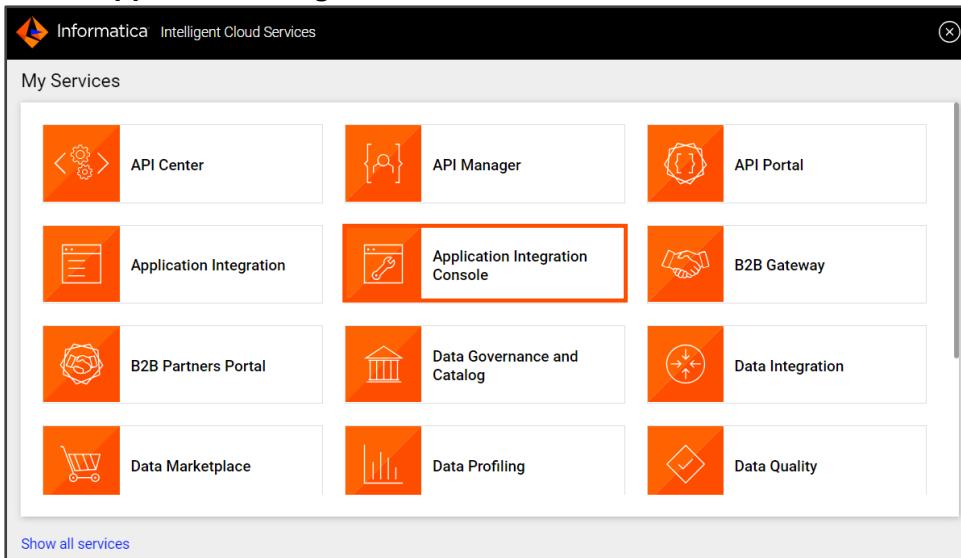
You see the following output:

```
{"Status":"Your order is accepted","OrderID":"544763854068928512","ItemPrice":50.0,"OrderPrice":100.0}
```

47. Click the drop-down beside the Application Integration service.



48. Select Application Integration Console.

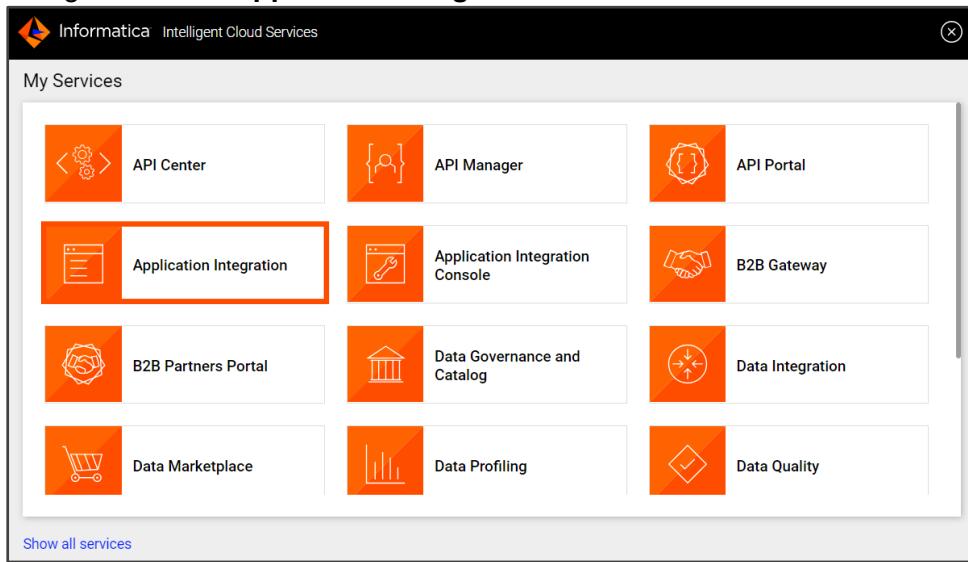


49. Click Processes.

50. You can see that the process status is Completed.

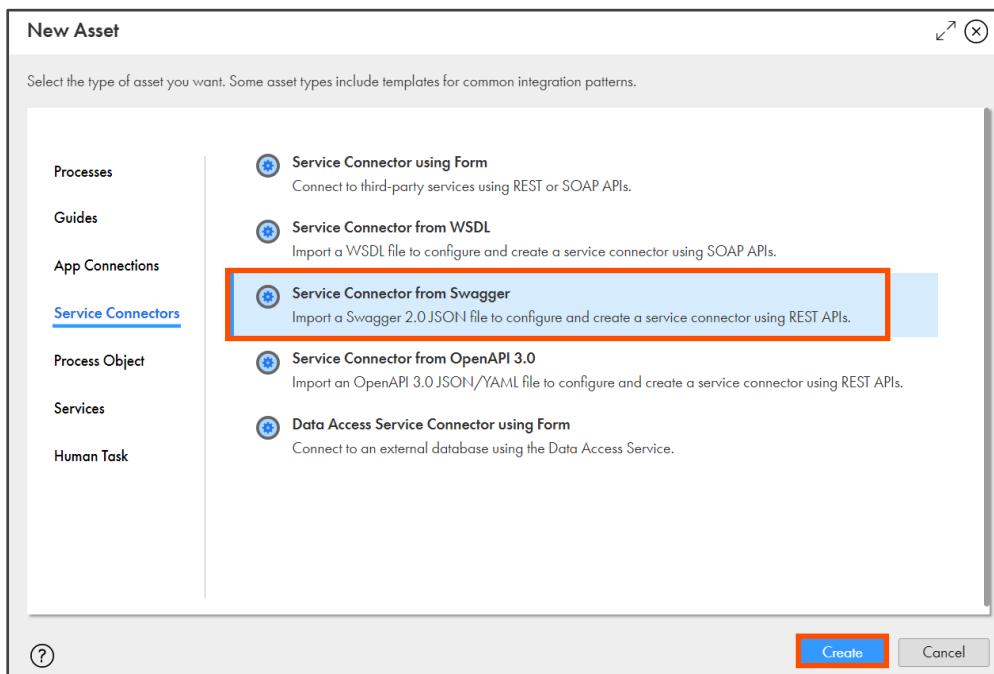
| Processes (22)           |                    |   |         |                         |                    |   |
|--------------------------|--------------------|---|---------|-------------------------|--------------------|---|
|                          | ID                 | Name  | Version | Start Date              | End Date           | Status  |
| <input type="checkbox"/> | 607148752113344512 |  S_XX_Order Management | 1       | 2021/8/3 09:52:22:79... | 2021/8/3 09:52:... |  Completed |

51. Navigate back to **Application Integration**.



### Create a Service Connector to the Margin Service

52. From the Navigation pane, click **New**.
53. In the New Asset dialog box, select **Service Connectors > Service Connector from Swagger**, and click **Create**.



54. In the New Connector from Swagger dialog box, enter the following properties in the Swagger Source tab:

| Property Name  | Property Content   |
|----------------|--|
| Name           | S_XX_Get Margin Detail   |
| Location       | S_XX_Order Management  |
| Description    | Access the margin service at <a href="https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt/Calculate_Margin_Service?swagger">https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt/Calculate_Margin_Service?swagger</a> |
| Swagger Source | URL  |
| Swagger URL    | <a href="https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt/Calculate_Margin_Service?swagger">https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt/Calculate_Margin_Service?swagger</a>                              |
| User Name      | testuser2  |
| Password       | password2#   |

55. Click **Next**.

New Service Connector from Swagger

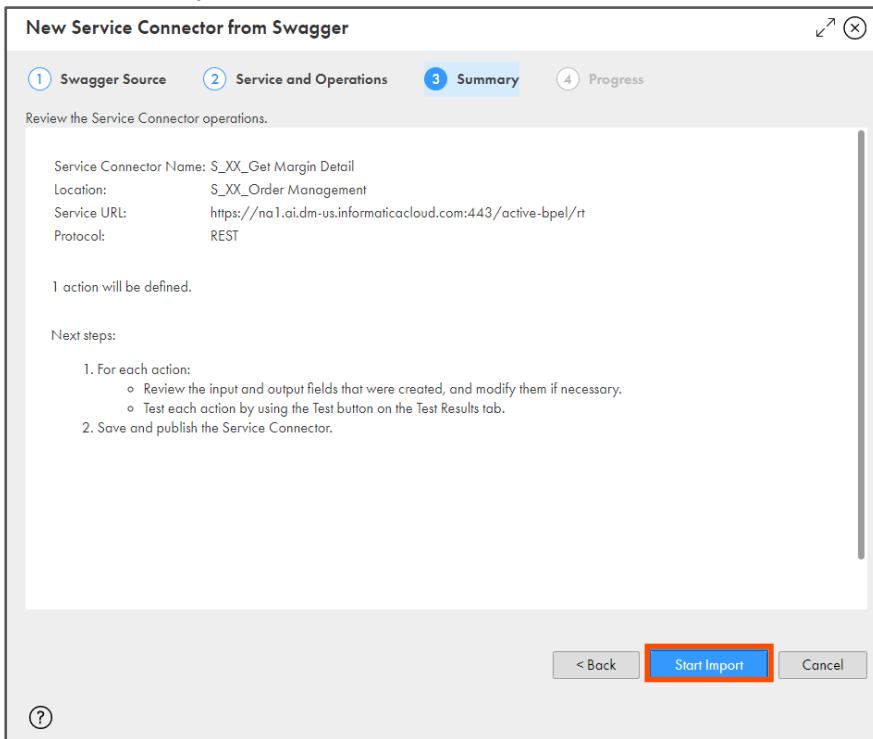
Specify information to identify the Service Connector and Swagger JSON file.

|                              |   |
|------------------------------|---|
| Name: <sup>*</sup>           | <input type="text" value="S_XX_Get Margin Detail"/>   |
| Location: <sup>*</sup>       | <input type="text" value="S_01_Order Management"/> <input type="button" value="Browse"/>  |
| Description:                 | <input type="text" value="Access the margin service at https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt/Calculate_Margin_Service?swagger"/> |
| <hr/>                        |   |
| Swagger File                 |   |
| Swagger Source: <sup>*</sup> | <input checked="" type="radio"/> File <input type="radio"/> URL   |
| Swagger URL: <sup>*</sup>    | <input type="text" value="https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt"/> <input type="button" value="X"/>                              |
| Use Authentication:          | <input checked="" type="checkbox"/>   |
| User Name: <sup>*</sup>      | <input type="text" value="testuser2"/>  |
| Password: <sup>*</sup>       | <input type="text" value="*****"/>  |

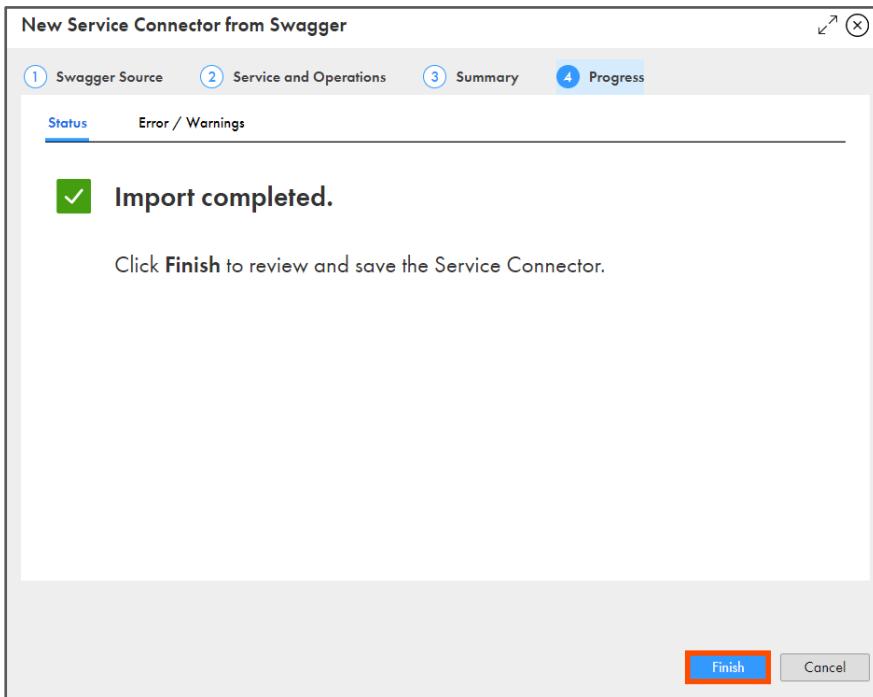
(?)

56. Retain the default selection for Service and Operations and click **Next**.

57. In the Summary tab, click **Start Import**.



58. A message ‘Import Completed.’ appears, click **Finish**.



59. Click **Save**.

60. Publish the connector.



### Create a Connection

61. From the Navigation Pane, click **New**.
62. In the New Asset dialog box, select **App Connections > App Connection**, and click **Create**.

| Property Name | Property Content                             |
|---------------|--|
| Name          | S-XX-Get-Margin-Detail                       |
| Location      | S_XX_Order Management                        |
| Description   | Connection for the Get margin Detail SVC.    |
| Type          | S_XX_Order Management>S_XX_Get Margin Detail |
| Run On        | Cloud Server or any Secure Agent             |
| Hostname      | na1.ai.dm-us.informaticacloud.com:443        |
| Username      | testuser2                                    |
| Password      | password2#                                   |

**Connection Details**

|                  |  |        |
|------------------|--|--------|
| Name: *          | S-XX-Get-Margin-Detail                         |        |
| Location: *      | S_01_Order Management                          | Browse |
| Description:     | Connection for the Get margin Detail SVC.      |        |
| Type: *          | S_XX_Order Management > S_XX_Get Margin Detail |        |
| Run On: *        | Cloud Server or any Secure Agent               |        |
| Connection Test: | Not Supported                                  |        |
| OData-Enabled:   | Not Supported                                  |        |

**Connection Properties:**

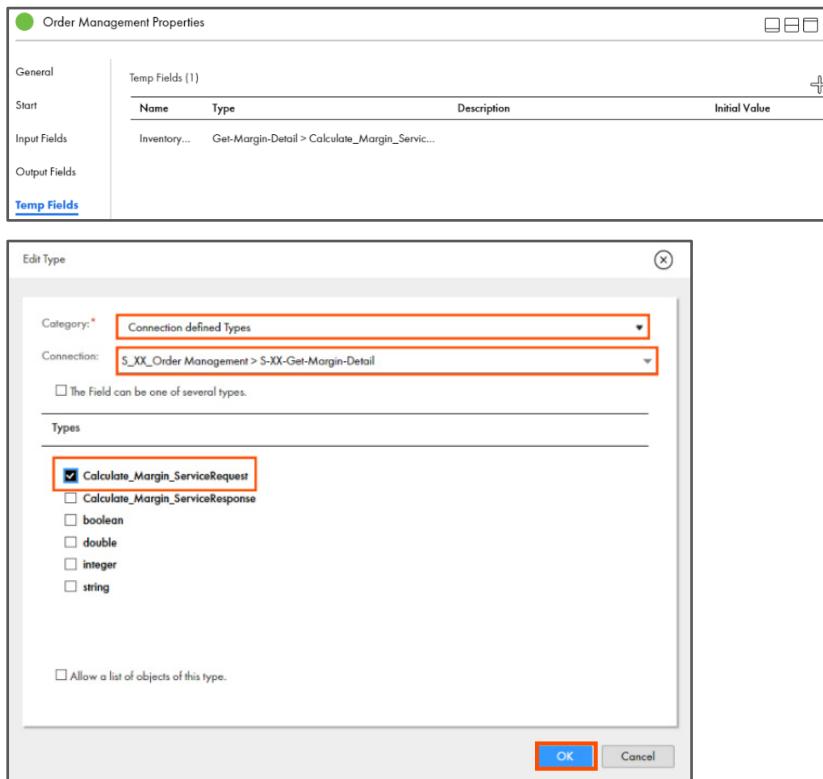
| Name       | Value                                 | Description |
|------------|---------------------------------------|-------------|
| hostname * | na1.ai.dm-us.informaticacloud.com:443 | Host Name   |
| username * | testuser2                             | User Name   |
| password * | *****                                 | Password    |

**63. Save and Publish** the asset.

A message 'The asset [Get-Inventory-Detail] was published successfully.' appears.

### Modify the Process to Add the Margin Details Service

64. In Application Integration, navigate to **S\_XX\_Order Management** folder.
65. Reopen the **S\_XX\_Order Management** process and select the **Start** step.
66. In the Temp Fields tab, to add a new field, click .
67. In the field name, enter **InventoryDetails**.
68. From Type drop-down, select **More Types**.
69. In the Edit Type dialog box, from the Category drop-down, select **Connection defined Types**.
70. From the Connection drop-down, select **S\_XX\_Order Management > S-XX-Get-Margin-Detail**.
71. Select the **Calculate\_Margin\_ServiceRequest** option and click **OK**.



The screenshot displays two windows. The top window is 'Order Management Properties' with the 'Temp Fields' tab selected, showing a single entry: 'Inventory...' with 'Type' set to 'Get-Margin-Detail > Calculate\_Margin\_ServiceRequest'. The bottom window is 'Edit Type', where 'Category' is set to 'Connection defined Types' and 'Connection' is set to 'S\_XX\_Order Management > S-XX-Get-Margin-Detail'. Under the 'Types' section, the 'Calculate\_Margin\_ServiceRequest' checkbox is checked, while other options like 'boolean', 'double', 'integer', and 'string' are unchecked. The 'OK' button at the bottom of the 'Edit Type' dialog is highlighted with a red box.

- 72. Click **Save**.**

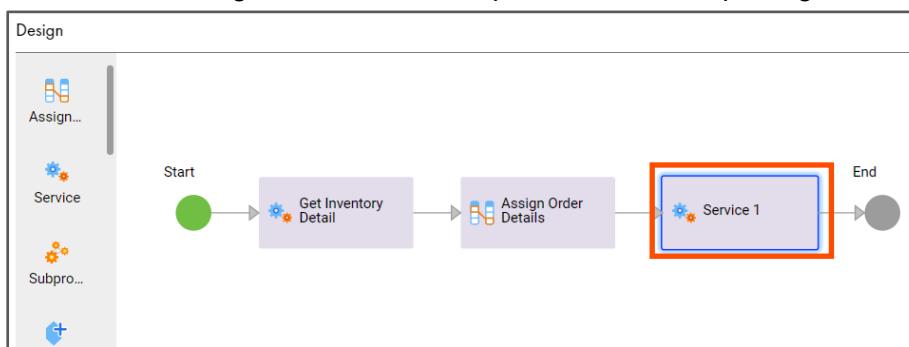
73. Select the **Assign Order Details** step and in the **Assignments** tab, add the following fields:

| Target   | Value Type | Value           |
|--|------------|-----------------|
| InventoryDetails > ItemCostPrice               | Field      | Item>costprice  |
| InventoryDetails > ItemCount                   | Field      | ItemCount       |
| InventoryDetails > ItemSellingPrice            | Field      | Item>sellprice  |
| InventoryDetails > SalesCommissionInPercentage | Field      | Item>commission |

|  |         |  |   |
|--|---------|--|---|
| Status   | Content | Your order is accepted                 | ✓ |
| OrderID  | Formula | info.getProcessId()                    | ✓ |
| ItemPrice                                      | Field   | Item > sellprice                       | ✓ |
| OrderPrice                                     | Formula | \$output.ItemPrice * \$input.ItemCount | ✓ |
| InventoryDetails > ItemCostPrice               | Field   | Item > costprice                       | ✓ |
| InventoryDetails > ItemCount                   | Field   | ItemCount                              | ✓ |
| InventoryDetails > ItemSellingPrice            | Field   | Item > sellprice                       | ✓ |
| InventoryDetails > SalesCommissionInPercentage | Field   | Item > commission                      | ✓ |

74. Click **Save**.

75. Between the Assign Order Details step and the End step, drag and drop a **Service** step.



76. In the General tab, enter **Get Margin Detail**.

|                |                                |
|----------------|--------------------------------|
| <b>General</b> | Step Type: Service             |
| Service        | Name: <b>Get Margin Detail</b> |

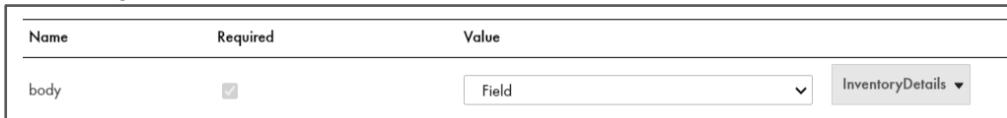
77. In the Service tab, from the Service Type drop-down, select **Connection**.

78. From the Connection drop-down, select **S\_XX\_Order Management > S-XX-Get-Margin-Detail**.

79. From the Action drop-down, select **Calculate\_Margin\_ServiceOperation**.



80. In the Input Fields tab, retain the default name, assign the Value Field > **InventoryDetails**.

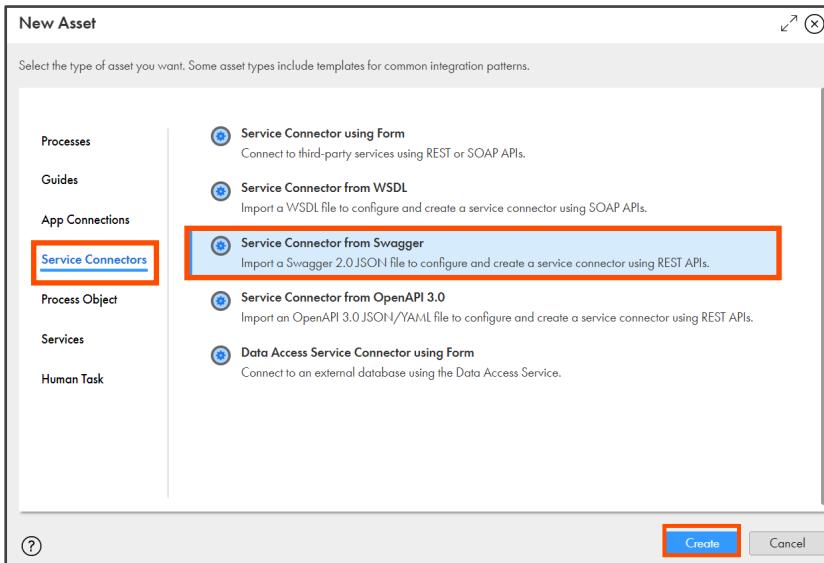


81. **Save and Publish** the process.

Now, you will create an email service connector and a connection. You later use these connections in the process to send emails to consumers and service providers.

## Create a Service Connector to the Email Service

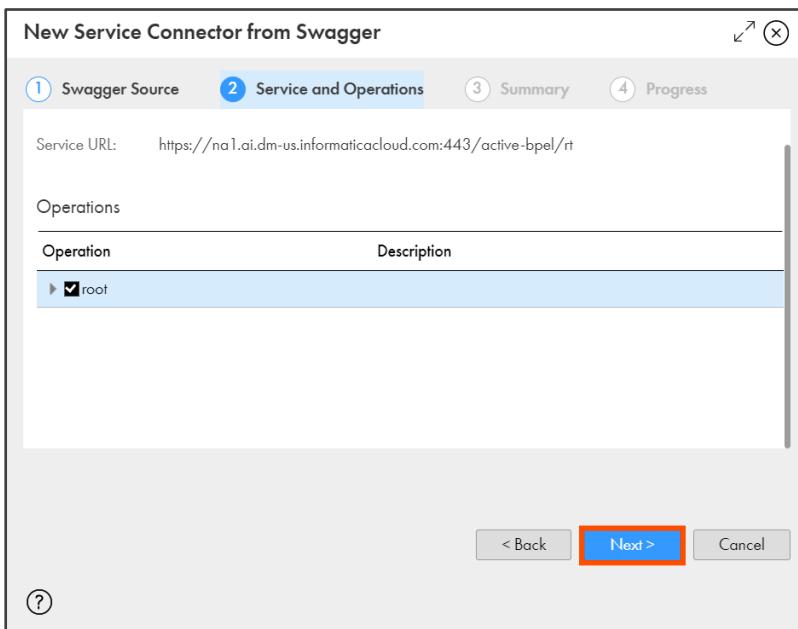
82. In Application Integration, click **New**.
83. In the New Asset dialog box, select **Service Connectors > Service Connector from Swagger > Create**.



84. In the New Connector from Swagger dialog box, enter the following properties in the Swagger Source tab:

| Property Name  | Property Content  |
|----------------|---|
| Name           | Email Service   |
| Location       | S_XX_Order Management   |
| Description    | Sends emails using the https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt/Email_Service service.  |
| Swagger Source | URL   |
| Swagger URL    | <a href="https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt/Email_Service?swagger">https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt/Email_Service?swagger</a> |
| User Name      | testuser2   |
| Password       | password2#  |

85. Click **Next**.  
 86. Review the **Service and Operations** tab and then click **Next**.



New Service Connector from Swagger

① Swagger Source    ② Service and Operations    ③ Summary    ④ Progress

Service URL: https://na1.ai.dm-us.informaticacloud.com:443/active-bpel/rt

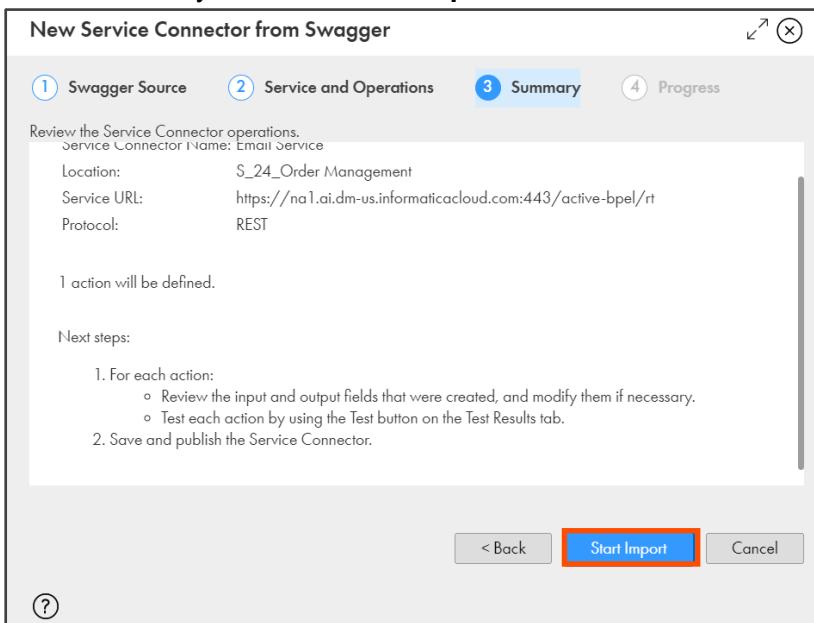
Operations

| Operation | Description                         |
|-----------|-------------------------------------|
| root      | <input checked="" type="checkbox"/> |

< Back    **Next >**    Cancel

(?)

87. In the Summary tab, click **Start Import**.

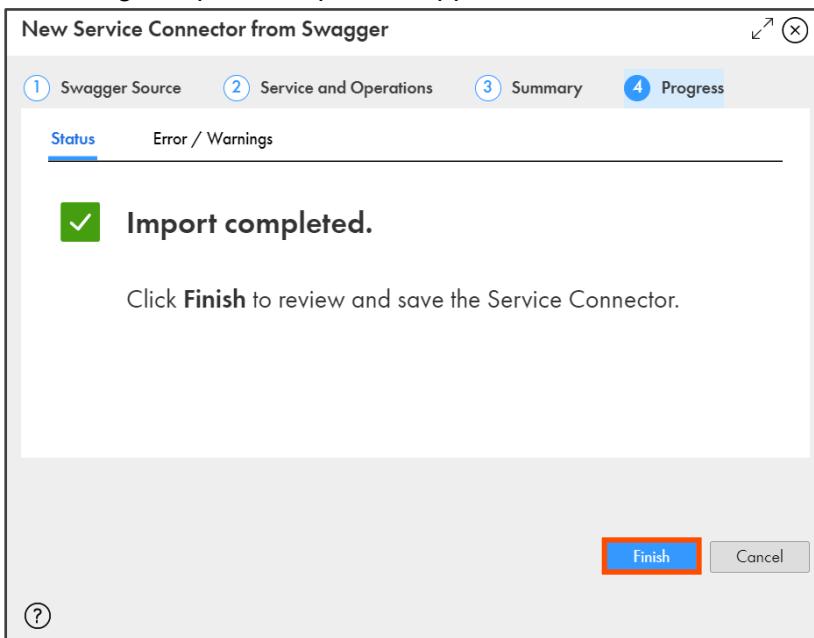


The screenshot shows the 'New Service Connector from Swagger' dialog box with the 'Summary' tab selected. The service connector name is 'Email Service'. The location is 'S\_24\_Order Management', the service URL is 'https://na1.ai.dm-us.informaticacloud.com:443/active-bpel/rt', and the protocol is 'REST'. It indicates that 1 action will be defined. Below this, there are instructions for next steps:

- For each action:
  - Review the input and output fields that were created, and modify them if necessary.
  - Test each action by using the Test button on the Test Results tab.
- Save and publish the Service Connector.

At the bottom, there are buttons for '< Back', 'Start Import' (which is highlighted with a red border), and 'Cancel'.

88. A message 'Import Completed.' appears, click **Finish**.

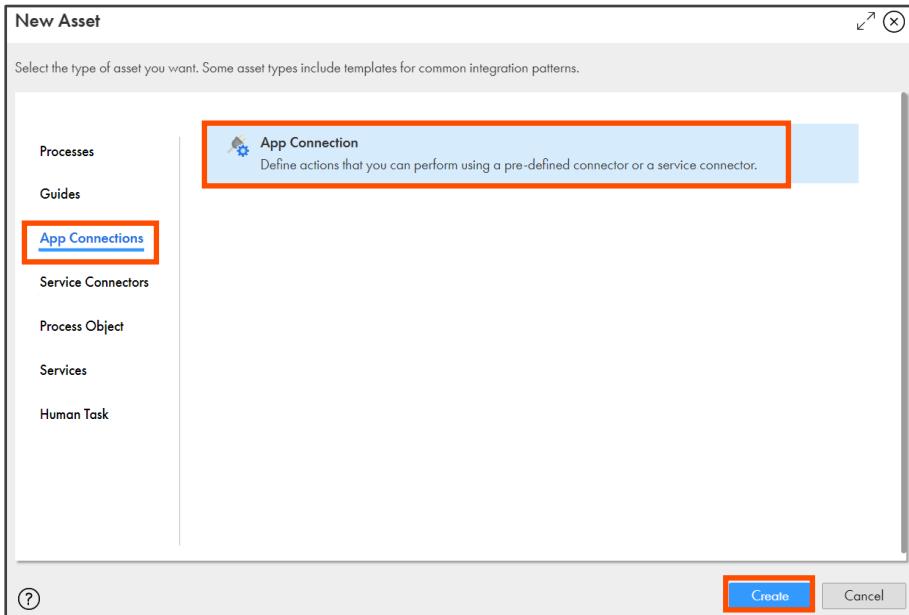


The screenshot shows the 'New Service Connector from Swagger' dialog box with the 'Status' tab selected. It displays a green checkmark icon followed by the text 'Import completed.' Below this, it says 'Click Finish to review and save the Service Connector.' At the bottom, there are buttons for 'Finish' (highlighted with a red border) and 'Cancel'.

89. **Save and Publish** the process.

## Create an Email Connection

90. In Application Integration, click **New**.
91. In the New Asset dialog box, select **App Connections > App Connection > Create**.



92. In the Name field, enter **Email-Service**.
93. In the Location field, select your project folder **S\_XX\_Order Management**.
94. In the Description field, enter **Connection for the Email Service SVC**.
95. In the Type field, enter **S\_XX\_Order Management > Email Service**.
96. From the Run On drop-down, select **Cloud Server or any Secure Agent**.

| Connection Details     |  |
|------------------------|--|
| Name: <sup>*</sup>     | <input type="text" value="Email-Service"/>   |
| Location: <sup>*</sup> | <input type="text" value="S_01_Order Management"/> <span style="float: right;">Browse</span> |
| Description:           | <input type="text" value="Connection for the Email Service SVC."/>                           |
| Type: <sup>*</sup>     | <input type="text" value="S_01_Order Management &gt; Email Service"/>                        |
| Run On: <sup>*</sup>   | <input type="text" value="Cloud Server or any Secure Agent"/>                                |
| Connection Test:       | Not Supported  |
| OData-Enabled:         | Not Supported  |

97. In the hostName field, enter **na1.ai.dm-us.informaticacloud.com:443**
98. In the username field, enter **testuser2**.

99. In the password field, enter **password2#**.

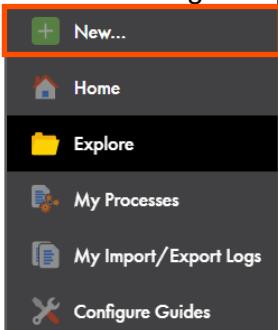
| Connection Properties |                                       |
|-----------------------|---------------------------------------|
| Name                  | Value                                 |
| hostName:*            | na1.ai.dm-us.informaticacloud.com:443 |
| username:*            | testuser2                             |
| password:*            | *****                                 |

100. Click **Save** and then click **Publish**.

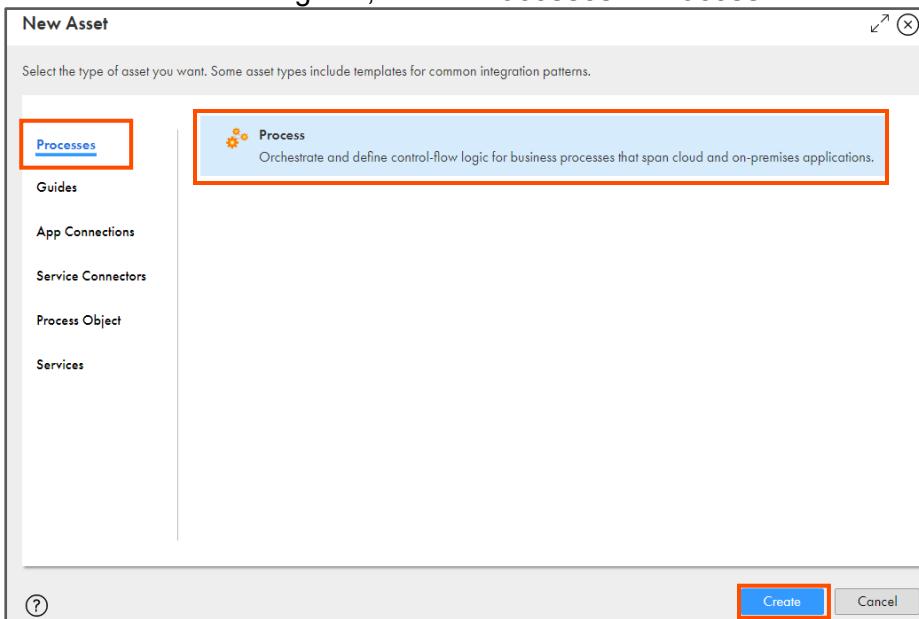
## Create a Subprocess

Now, you will create a process that uses the email connection to send emails to consumers and vendors. When you add this process to the Order Management process, it becomes a subprocess.

101. From the Navigation pane, click **New**.

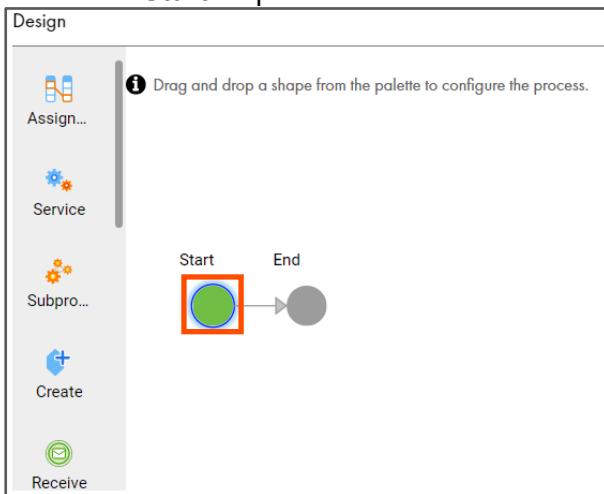


102. In the New Asset dialog box, select **Processes > Process** and click **Create**.



**Note:** Process Designer opens with a process template.

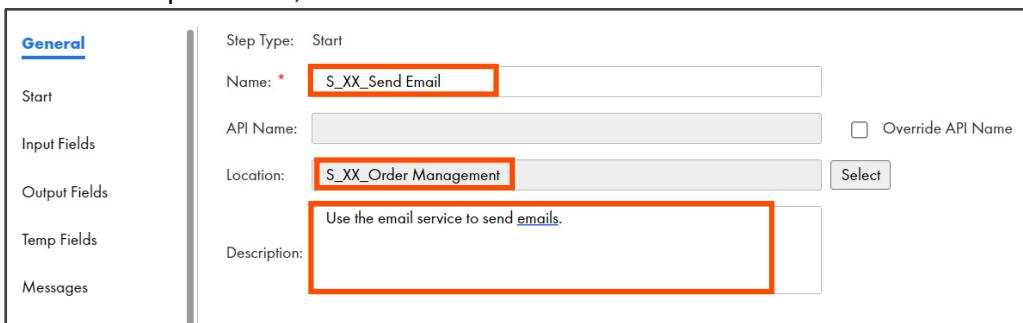
103. Select the **Start** step.



104. In the General tab, in the Name field, enter **S\_XX\_Send Email**.

105. In the Location field, select your project folder **S\_XX\_Order Management**.

106. In the Description field, enter **Use the email service to send emails**.

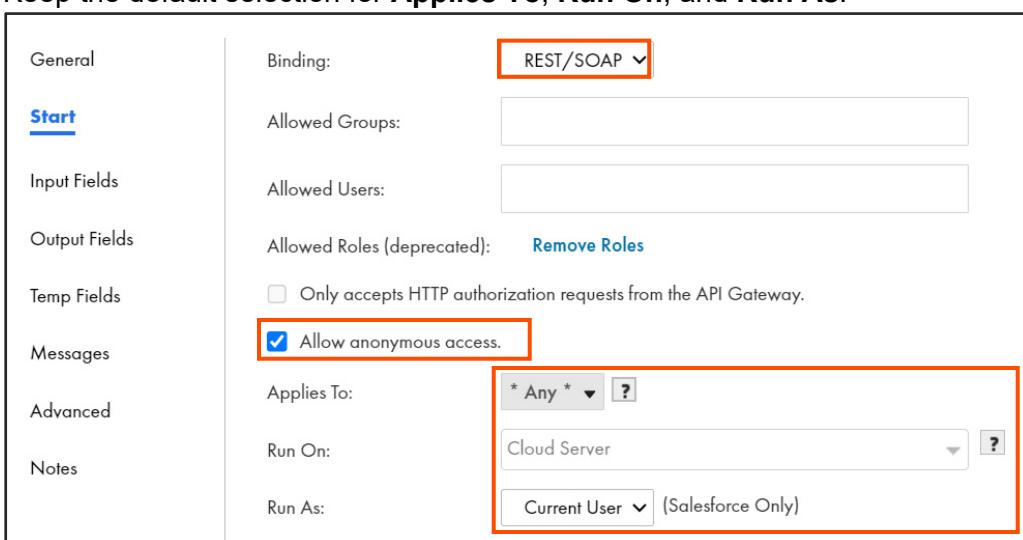


|                |  |
|----------------|--|
| <b>General</b> | Step Type: Start<br>Name: * <b>S_XX_Send Email</b><br>API Name:<br>Location: <b>S_XX_Order Management</b><br>Description: <b>Use the email service to send emails.</b> |
| Start          |  |
| Input Fields   |  |
| Output Fields  |  |
| Temp Fields    |  |
| Messages       |  |

107. In the Start tab, for the Binding field, select **REST/SOAP**.

108. For Allowed Roles, select the **Allow anonymous access** option.

109. Keep the default selection for **Applies To**, **Run On**, and **Run As**.

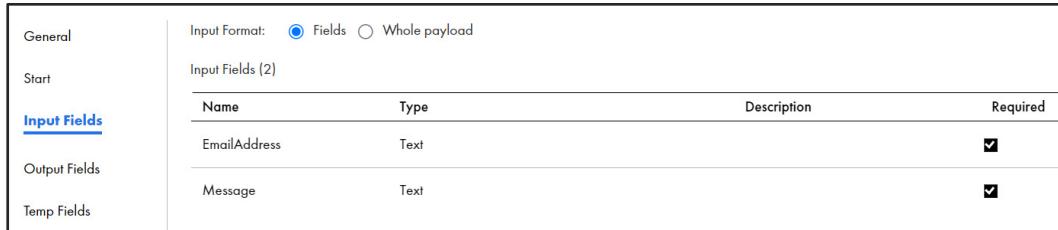


|               |  |
|---------------|--|
| General       | Binding: <b>REST/SOAP</b>  |
| <b>Start</b>  | Allowed Groups:<br>Allowed Users:<br>Allowed Roles (deprecated): <b>Remove Roles</b><br><input type="checkbox"/> Only accepts HTTP authorization requests from the API Gateway.<br><input checked="" type="checkbox"/> Allow anonymous access. |
| Input Fields  |  |
| Output Fields |  |
| Temp Fields   |  |
| Messages      |  |
| Advanced      | Applies To: <b>* Any *</b><br>Run On: <b>Cloud Server</b><br>Run As: <b>Current User</b> <small>(Salesforce Only)</small>  |
| Notes         |  |

110. In the Input Fields tab, to add a new field, click .

111. Add the following fields:

| Name         | Type | Required |
|--------------|------|----------|
| EmailAddress | Text | Yes      |
| Message      | Text | Yes      |

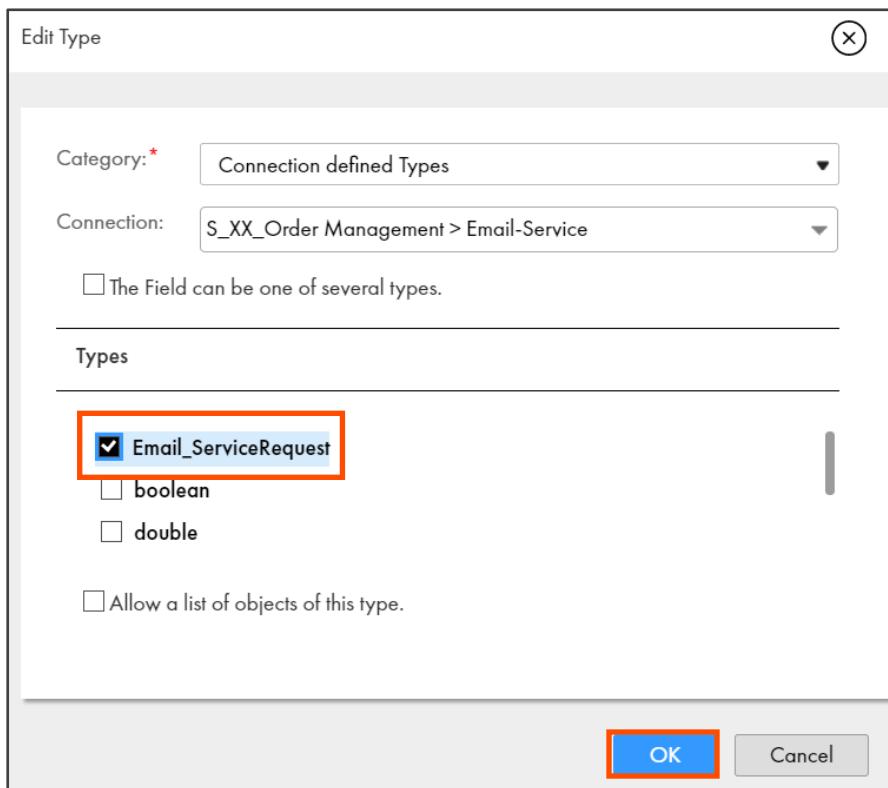


The screenshot shows the 'Input Fields' tab selected in the sidebar. Under 'Input Fields (2)', there are two entries: 'EmailAddress' (Type: Text, Required: Yes) and 'Message' (Type: Text, Required: Yes). The 'Fields' radio button is selected under 'Input Format'.

112. In the Temp Fields tab, to add a new field, click .

113. Add the following field:

| Name             | Type   |
|------------------|--|
| InventoryDetails | More Types > Connection Defined Types > S_XX_Order Management > Email-Service > Email_ServiceRequest |



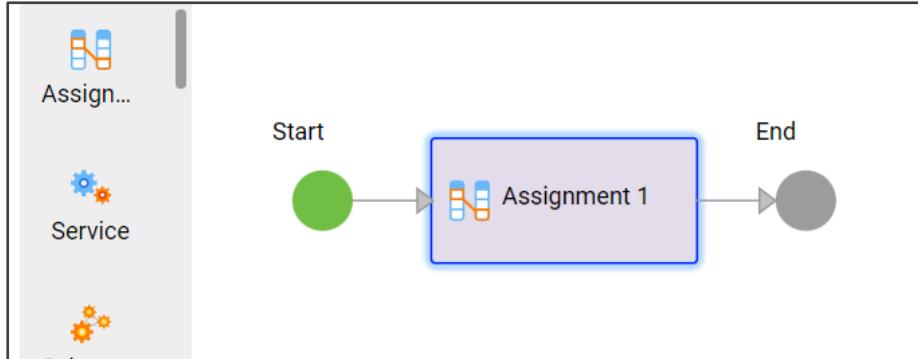
The screenshot shows the 'Edit Type' dialog box. Under 'Category': **Connection defined Types**. Under 'Connection': **S\_XX\_Order Management > Email-Service**. There is a checkbox for 'The Field can be one of several types.' Below it, under 'Types', the 'Email\_ServiceRequest' checkbox is selected and highlighted with a red box. Other options like 'boolean' and 'double' are also listed. At the bottom, there is a checkbox for 'Allow a list of objects of this type.' The 'OK' button is highlighted with a red box.

114. In the Advanced tab, from the **Tracing Level** drop-down, select **Verbose**.

S\_XX\_Send Email Properties

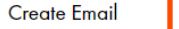
|              |   |
|--------------|---|
| General      | Suspend on Fault: <input type="checkbox"/>  |
| Start        | Tracing Level: <b>Verbose</b>  |
| Input Fields | Warning: Verbose logging will increase the process execution time.  |

115. Drag and drop an **Assignment** step between the Start and End steps,



116. Select the General tab and enter **Create Email** in the Name field.

Create Email Properties

|                |   |
|----------------|---|
| <b>General</b> | Step Type: Assignment   |
| Assignments    | Name: <b>Create Email</b>  |

117. Select the **Assignment** step.

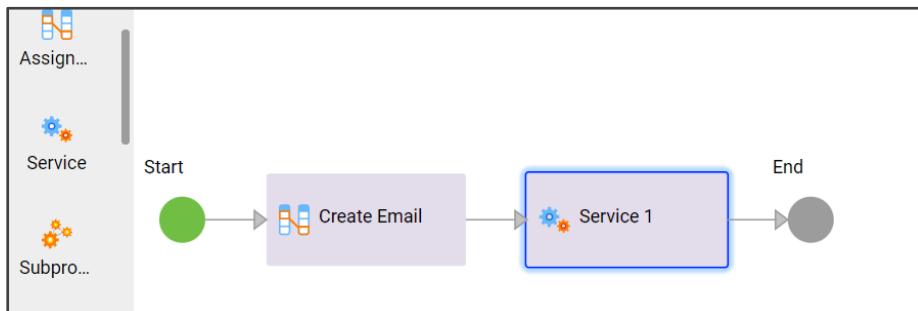
118. In the **Assignments** tab, assign values to the following output fields:

| Field                      | Assigned Using | From         |
|----------------------------|----------------|--------------|
| InventoryDetails > Message | Field          | Message      |
| InventoryDetails > SendTo  | Field          | EmailAddress |

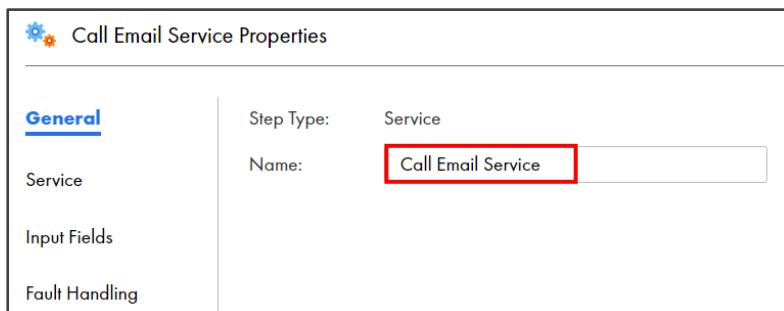
Create Email Properties

|                    |                            |   |  |
|--------------------|----------------------------|---|--|
| General            | Field                      | Assigned Using  | From   |
| <b>Assignments</b> | InventoryDetails > Message | Field  | Message       |
|                    | InventoryDetails > SendTo  | Field  | EmailAddress  |

119. Drag and drop a service step on to the canvas in between the Create Email step and the End step.



120. Select the General tab and enter **Call Email Service** in the Name field.



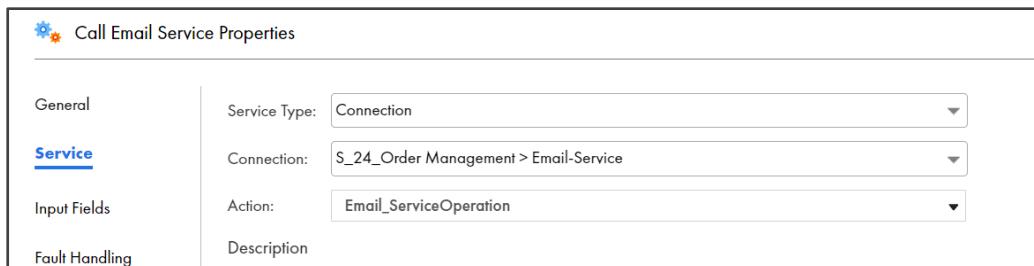
| Call Email Service Properties |   |
|-------------------------------|---|
| <b>General</b>                | Step Type: Service<br>Name: <b>Call Email Service</b> |
| Service                       |   |
| Input Fields                  |   |
| Fault Handling                |   |

121. Select the **Service** tab.

122. Set the Service Type to **Connection**.

123. Set the Connection to **S\_XX\_Order Management > Email-Service**.

124. Set the Action field to **Email\_ServiceOperation**.



| Call Email Service Properties |   |
|-------------------------------|---|
| <b>General</b>                | Service Type: Connection                          |
| <b>Service</b>                | Connection: S_24_Order Management > Email-Service |
| Input Fields                  | Action: Email_ServiceOperation                    |
| Fault Handling                | Description                                       |

125. Select the **Input** tab and assign the value **Field > Inventory Details** to the body field.

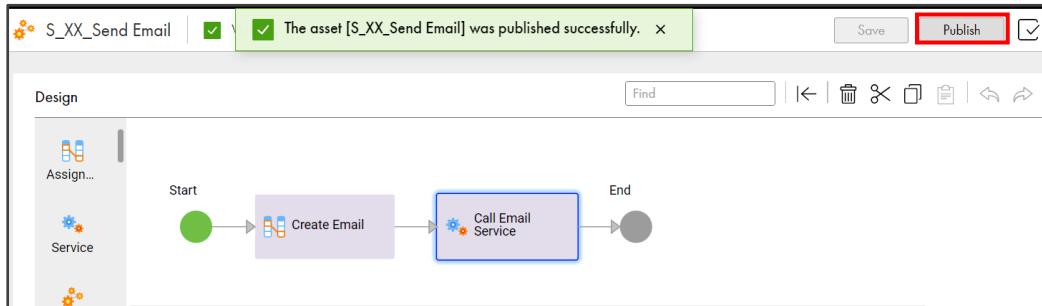


| Input Fields (1) |                                     |       |
|------------------|-------------------------------------|-------|
| Name             | Required                            | Value |
| body             | <input checked="" type="checkbox"/> | Field |

Value dropdown: InventoryDetails ▾

126. Click **Save**.

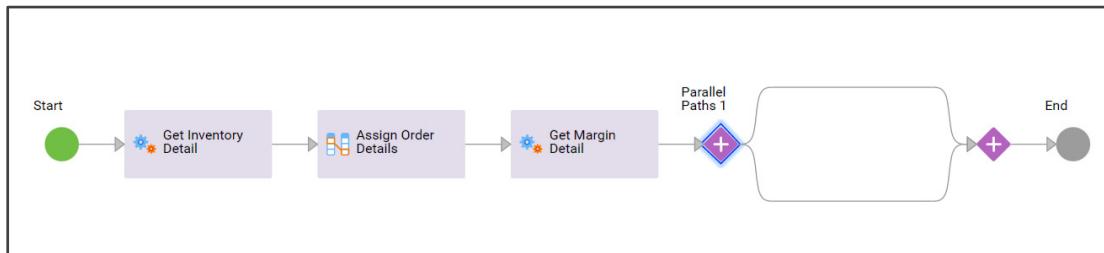
127. Click **Publish**.



### Add the Subprocess to the Main Process

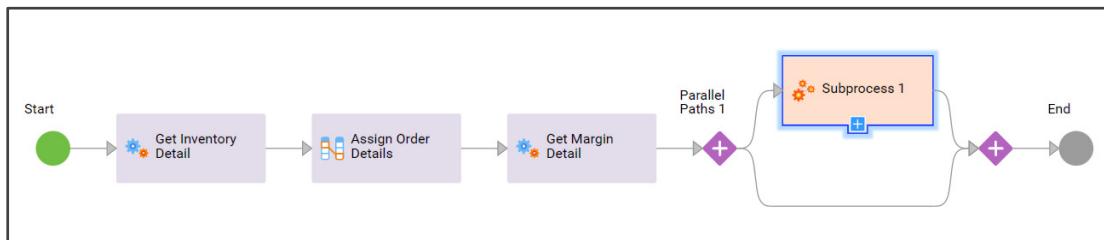
Now, you will add the S\_XX\_Send Email process to the S\_XX\_Order Management process. You need to send emails to the consumer and to the vendor. To do this, add a Parallel Paths step with two branches and add a service step on each branch. The first branch sends an email to the consumer and the second branch sends an email to the vendor.

128. In Application Integration, click **Explore**.
  129. Open the **S\_XX\_Order Management** process.
- Note:** You can use Search box to find the S\_XX\_Order Management process.
130. Drag and drop a **Parallel Path** step onto the canvas in between the Get Margin Detail Service step and the End step.

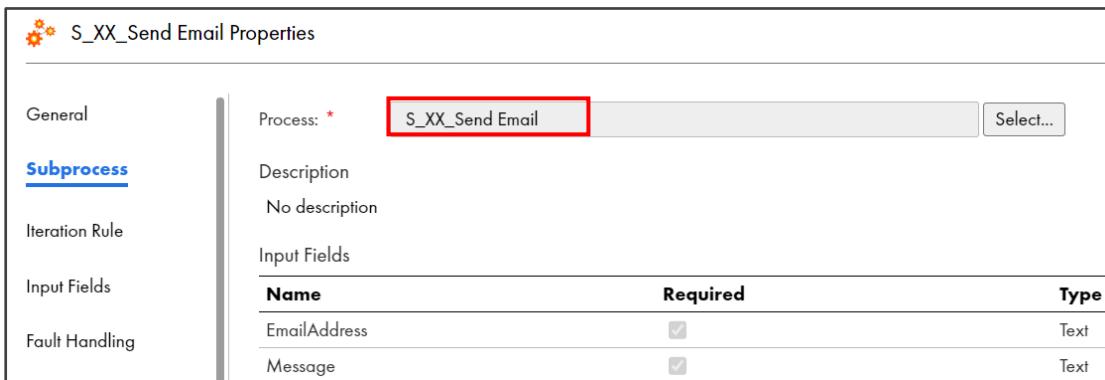


Next, you will add the S\_XX\_Send Email process to both branches. The first branch sends an email to the consumer. The second branch sends an email to the vendor.

131. Drag and drop a **Subprocess** step onto the first branch.



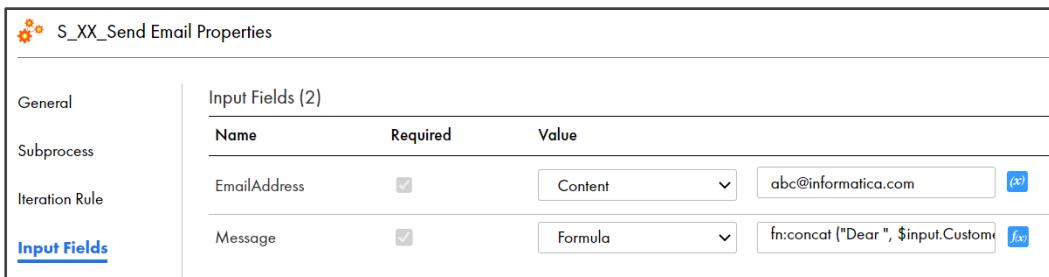
132. From the Subprocess tab, click **Select** and select the **S\_XX\_Send Email** process.



| General           | Process: *   | S_XX_Send Email                     | Select...   |
|-------------------|--------------|-------------------------------------|-------------|
| <b>Subprocess</b> | Description  | No description                      |             |
| Iteration Rule    |              |                                     |             |
| Input Fields      | <b>Name</b>  | <b>Required</b>                     | <b>Type</b> |
|                   | EmailAddress | <input checked="" type="checkbox"/> | Text        |
|                   | Message      | <input checked="" type="checkbox"/> | Text        |
| Fault Handling    |              |                                     |             |

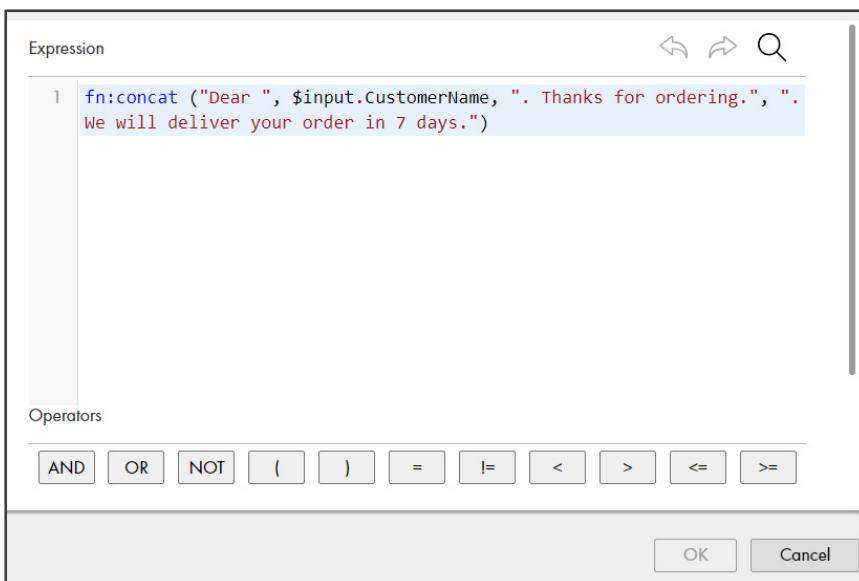
133. Select the **Input Fields** tab and add the following fields:

| Name         | Type    | Value  |
|--------------|---------|--|
| EmailAddress | Content | Enter a valid email address.   |
| Message      | Formula | fn:concat ("Dear ", \$input.CustomerName, ". Thanks for ordering.", ". We will deliver your order in 7 days.") |



| General             | Input Fields (2) |                                     |  |
|---------------------|------------------|-------------------------------------|--|
| Subprocess          | Name             | Required                            | Value  |
| Iteration Rule      | EmailAddress     | <input checked="" type="checkbox"/> | Content abc@informatica.com  |
| <b>Input Fields</b> | Message          | <input checked="" type="checkbox"/> | Formula fn:concat ("Dear ", \$input.CustomerName, ". Thanks for ordering.", ". We will deliver your order in 7 days.") |

**Note:** To enter the value in the Expression field, from the **CAI Lab Prep Files** folder, open the notepad file named **29\_LabGuide\_CAI\_Live Project\_17-1**. Copy the command mentioned under **Step B** and paste it in the Expression field.



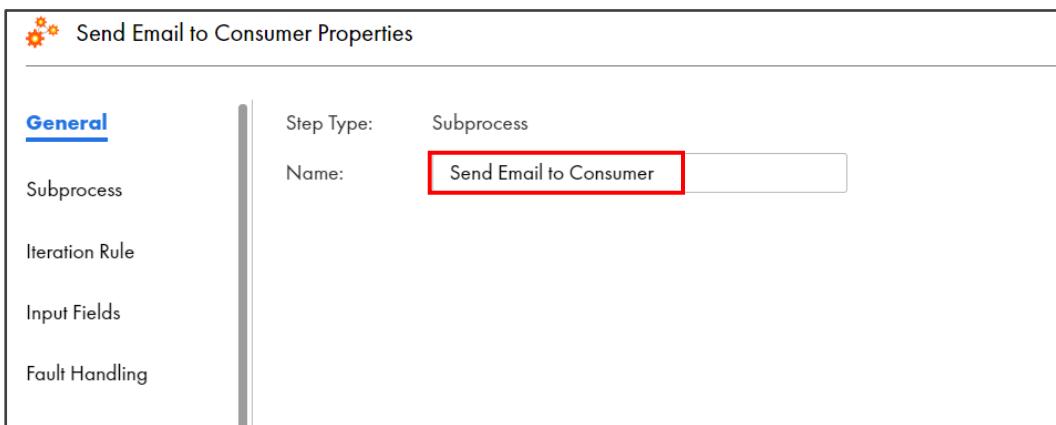
Expression

```
fn:concat ("Dear ", $input.CustomerName, ". Thanks for ordering.", ". We will deliver your order in 7 days.")
```

Operators

AND OR NOT ( ) = != < > <= >=

134. Select the **General** tab and in the Name field, enter **Send Email to Consumer**.



**Send Email to Consumer Properties**

**General**

Step Type: Subprocess

Name: **Send Email to Consumer**

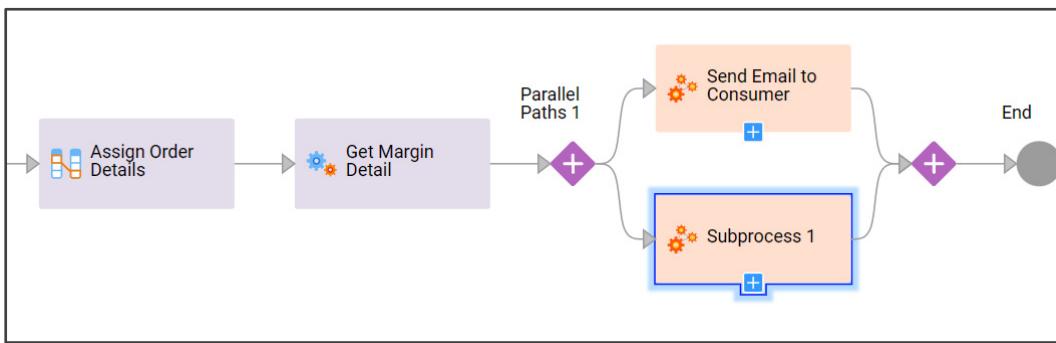
Subprocess

Iteration Rule

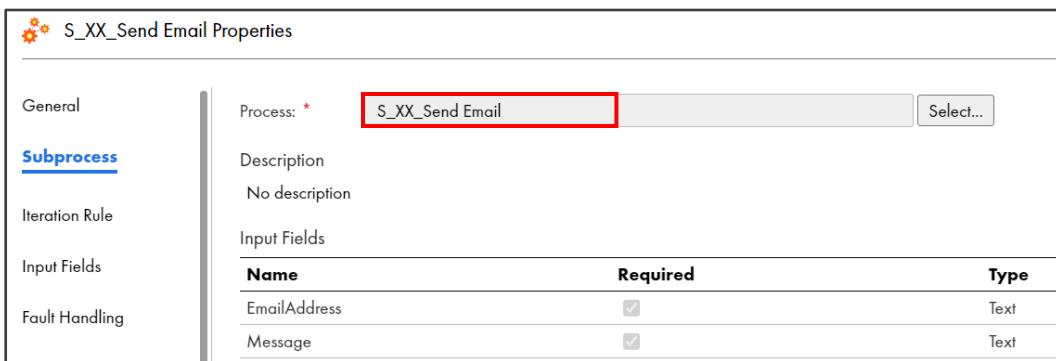
Input Fields

Fault Handling

135. Drag and drop a **Subprocess** step onto the second branch.



136. From the Subprocess tab, click **Select** and select the **S\_XX\_Send Email** process.



**S\_XX\_Send Email Properties**

**General**

Process: \* **S\_XX\_Send Email**

**Subprocess**

Description  
No description

Input Fields

| Name         | Required                            | Type |
|--------------|-------------------------------------|------|
| EmailAddress | <input checked="" type="checkbox"/> | Text |
| Message      | <input checked="" type="checkbox"/> | Text |

Iteration Rule

Input Fields

Fault Handling

137. Select the Input Fields tab and add the following fields:

| Name         | Type    | Value  |
|--------------|---------|--|
| EmailAddress | Content | Enter a valid email address.   |
| Message      | Formula | <pre>fn:concat ("Order has been accepted. Please note below details for your records.", " Overall Profit: ", \$output.Calculate_Margin_ServiceResponse[1]/MarginBeforeCommission ,". SalesCommission: ",\$output.Calculate_Margin_ServiceResponse[1]/SalesCommission,". Profit after Commission: ", \$output.Calculate_Margin_ServiceResponse[1]/MarginAfterCommission )</pre> |

 S\_XX\_Send Email Properties

|                     |                  |                                     |   |
|---------------------|------------------|-------------------------------------|---|
| General             | Input Fields (2) |                                     |   |
| Subprocess          | Name             | Required                            | Value                                       |
| Iteration Rule      | EmailAddress     | <input checked="" type="checkbox"/> | Content<br>abc@informatica.com              |
| <u>Input Fields</u> | Message          | <input checked="" type="checkbox"/> | Formula<br>fn:concat ("Order has been accep |
| Fault Handling      |                  |                                     |   |

**Note:** To enter the value in the Expression field, from the **CAI Lab Prep Files** folder, open the notepad file named **29\_LabGuide\_CAI\_Live Project\_17-1**. Copy the command mentioned under **Step C** and paste it in the Expression field.

Expression

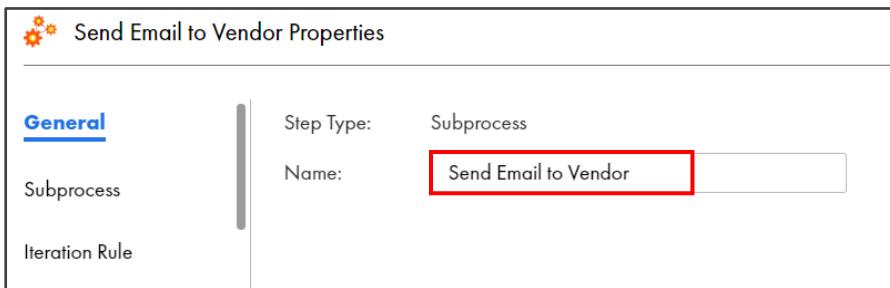
```
1 fn:concat ("Order has been accepted. Please note below details for
your records.", " Overall Profit: ",
$output.Calculate_Margin_ServiceResponse[1]/MarginBeforeCommission
,". SalesCommission:
",$output.Calculate_Margin_ServiceResponse[1]/SalesCommission,".
Profit after Commission: ",
$output.Calculate_Margin_ServiceResponse[1]/MarginAfterCommission )
```

Operators

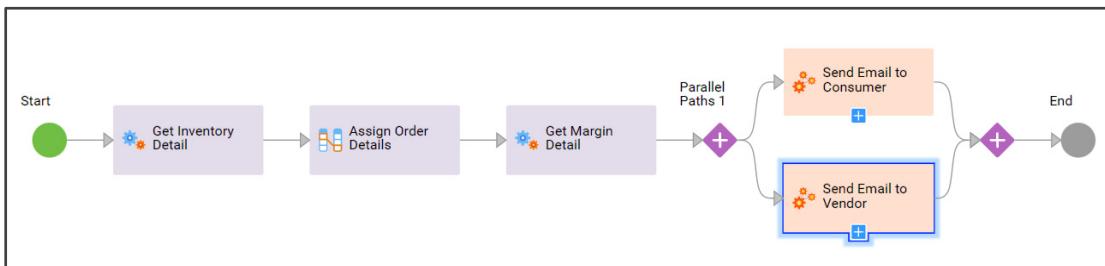
AND OR NOT ( ) = != < > <= >=

OK Cancel

138. Select the **General** tab and in the Name field, enter **Send Email to Vendor**.



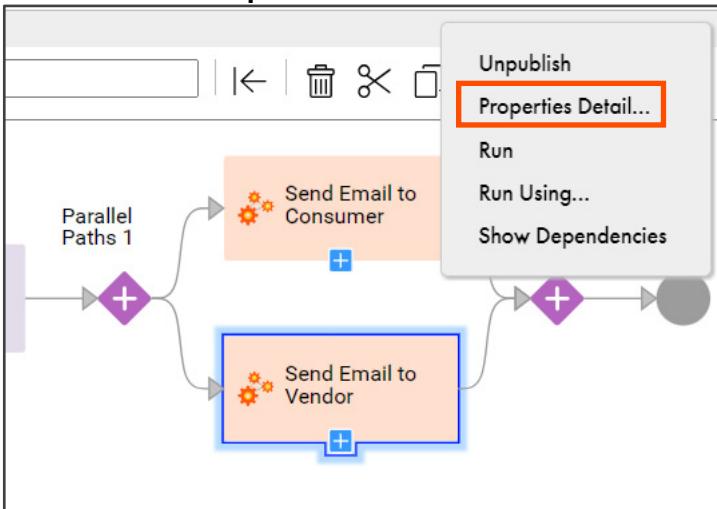
The following image shows the process canvas:



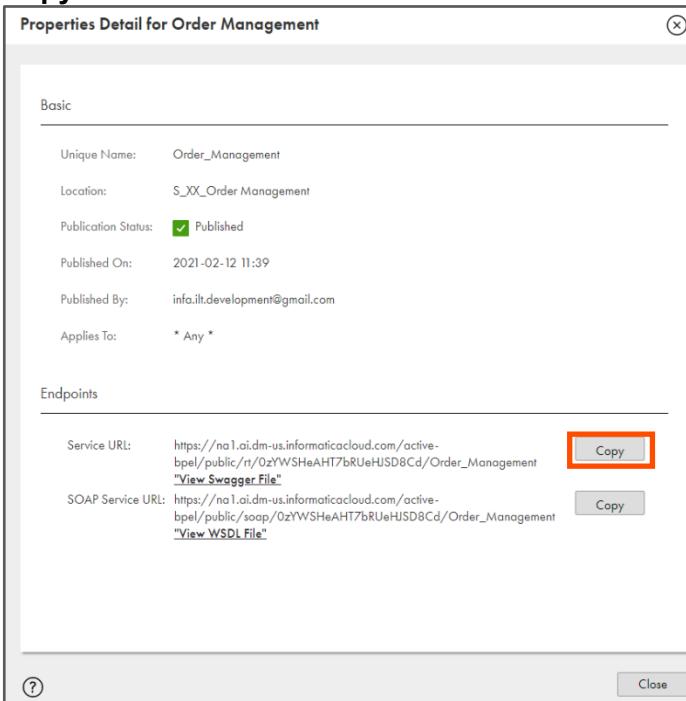
139. **Save and Publish** the process.

## Invoke the Modified Order Management Process

140. Click Action > **Properties Detail**.



**141. Copy the service URL.**



**142. Open a notepad file, paste the Service URL, and add the following input parameters to the end of service URL:**

```
<serviceURL>?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&ItemName=item1&ItemCount=2
```

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **29\_LabGuide\_CAI\_Live Project\_17-1**. Copy the command mentioned under **Step D**.

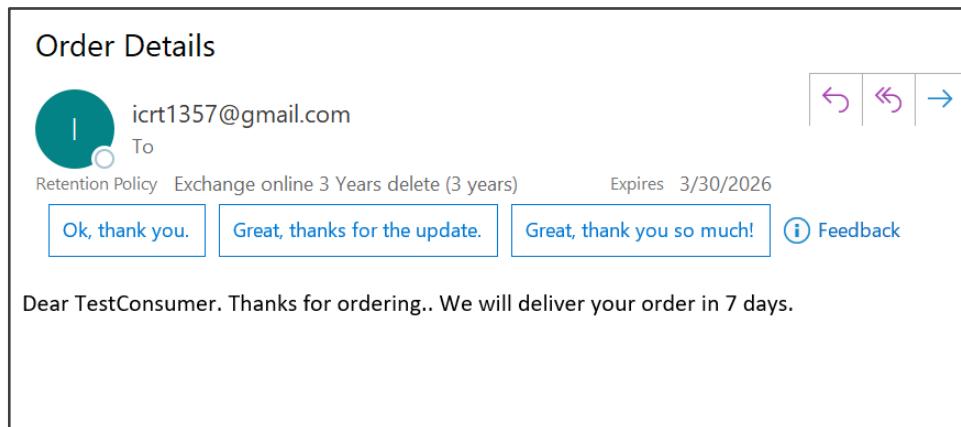
The following URL is a sample service URL with input parameters:

```
https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfIEqfZgh7aWGifF/S_XX_Order_Management-1?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&ItemName=item1&ItemCount=2
```

**143. Open a browser and paste the service URL with input parameters.**



**Note:** You see a successful response and an email to the given email address.



The screenshot shows an "Order Details" interface. At the top, there's an email icon and the recipient's email address: icrt1357@gmail.com. Below it, the word "To" is followed by a retention policy: "Retention Policy Exchange online 3 Years delete (3 years)". To the right, it says "Expires 3/30/2026". There are four buttons at the bottom: "Ok, thank you.", "Great, thanks for the update.", "Great, thank you so much!", and "Feedback". A message in the body of the email reads: "Dear TestConsumer. Thanks for ordering.. We will deliver your order in 7 days."

### Invoke Process to Show a Fault

144. In a browser, paste the following parameterized service URL. Here, we have changed the item name to Newitem1, which is an invalid value.

```
<serviceURL>?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&itemName>Newitem1&ItemCount=2
```

OR

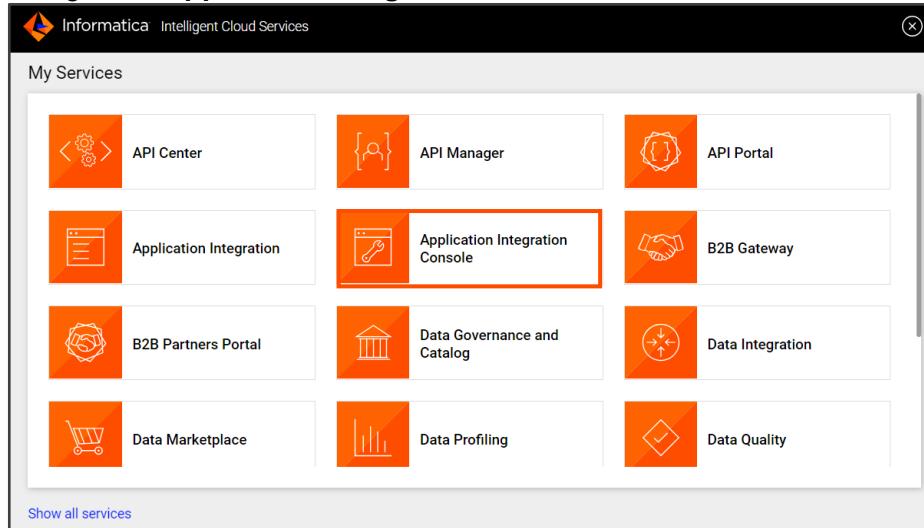
From the **CAI Lab Prep Files** folder, open the notepad file named **29\_LabGuide\_CAI\_Live Project\_17-1**. Copy the command mentioned under **Step E** and paste it in the browser.



The screenshot shows a browser window with a secure connection. The URL is https://na1.ai.dm-us.informaticacloud.com/active-bpel/rt/Order\_Man. The page content is a JSON object: {"error": {"code": 500, "detail": {"reason": "{}", "code": "HTTP\_400"}, "message": "{}"}}

You see an HTTPS 400 error response.

145. Navigate to **Application Integration Console**.



The screenshot shows the "My Services" dashboard of the Informatica Intelligent Cloud Services. It features a grid of nine service icons: API Center, API Manager, API Portal, Application Integration (which is highlighted with a red border), Application Integration Catalog, B2B Gateway, B2B Partners Portal, Data Governance and Catalog, Data Integration, Data Marketplace, Data Profiling, and Data Quality. At the bottom left, there is a link to "Show all services".

146. Click Processes.

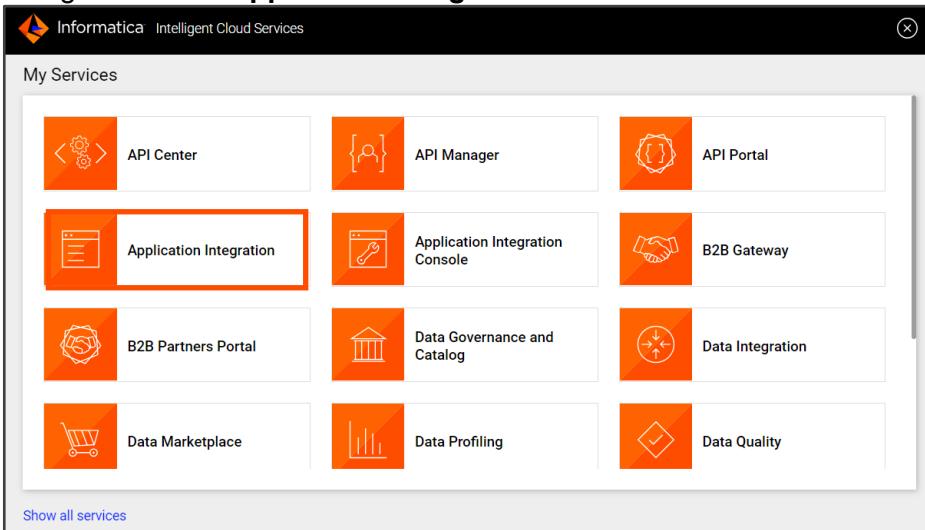
147. You can see that the Process status is Faulted.

| ID                 | Name             | Version | Start Date            | End Date         | Status  |
|--------------------|------------------|---------|-----------------------|------------------|---------|
| 548458362774237184 | Order Management | 1       | 2021/2/22 10:57:43... | 2021/2/22 10:... | Faulted |

Note: Process has faulted because "newitem1" does not exist.

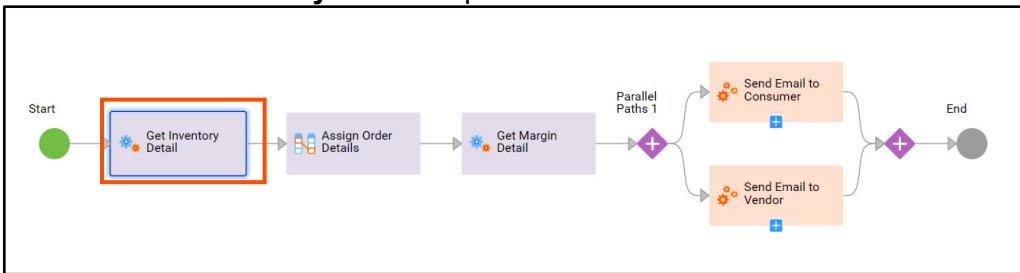
## Enable Fault Handling

148. Navigate back to Application Integration.

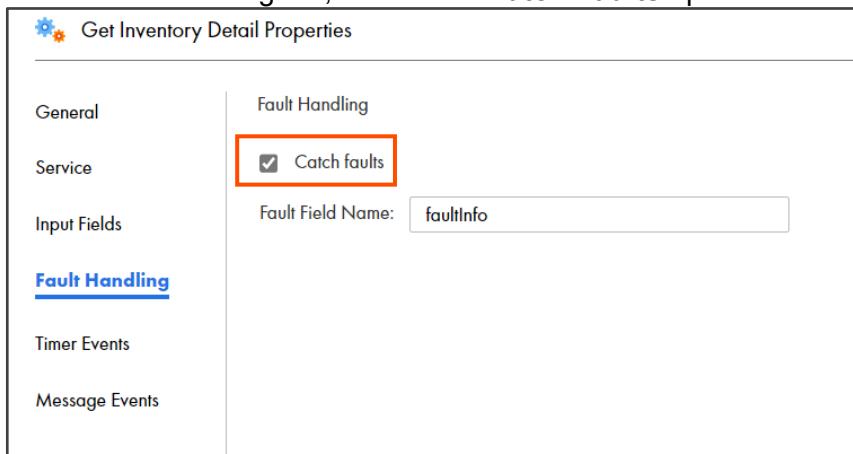


149. From the Explore window, select and open the S\_XX\_Order Management process.

150. Select the **Get Inventory Detail** step.

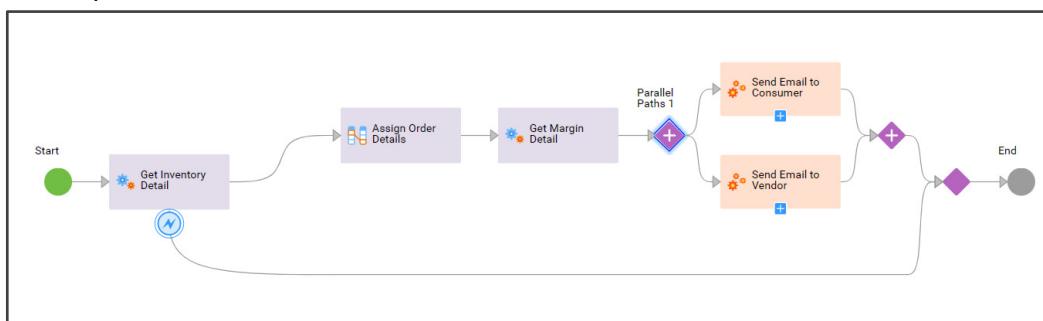


151. In the Fault Handling tab, enable the **Catch Faults** option.

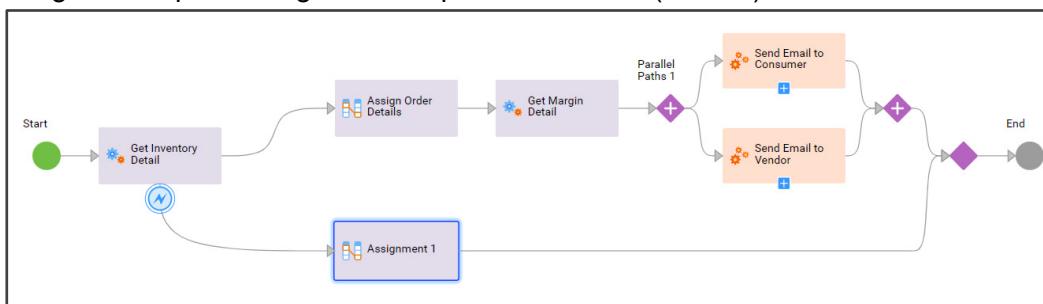


152. Drag and drop the **Assign Order Details** step to the “no fault” branch.

153. Repeat the previous step for all steps except the End step. Do not change the order of the steps.



154. Drag and drop an Assignment step onto the Fault (bottom) branch.



155. In the General tab, in the Name field, enter **Assignment to Status**.

156. In the Assignments tab, add the Status field with the Value as **Content > Item not in stock. Apologies for the inconvenience.**



| Field  | Assigned Using | From  |
|--------|----------------|---|
| Status | Content        | Item not in stock. Apologies for the inconvenience.  |

157. Save and publish the process.

158. Click **Actions > Properties Detail**.

159. **Copy the Service URL.**

160. Open a notepad file, paste the Service URL, and add the following parameters to the end of service URL:

<ServiceURL>?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&ItemName=newitem1&ItemCount=2

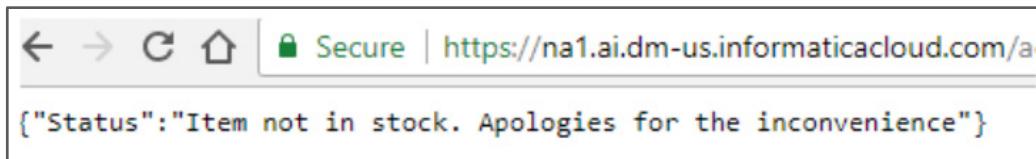
OR

From the **CAI Lab Prep Files** folder, open the notepad file named **29\_LabGuide\_CAI\_Live Project\_17-1**. Copy the command mentioned under **Step F**.

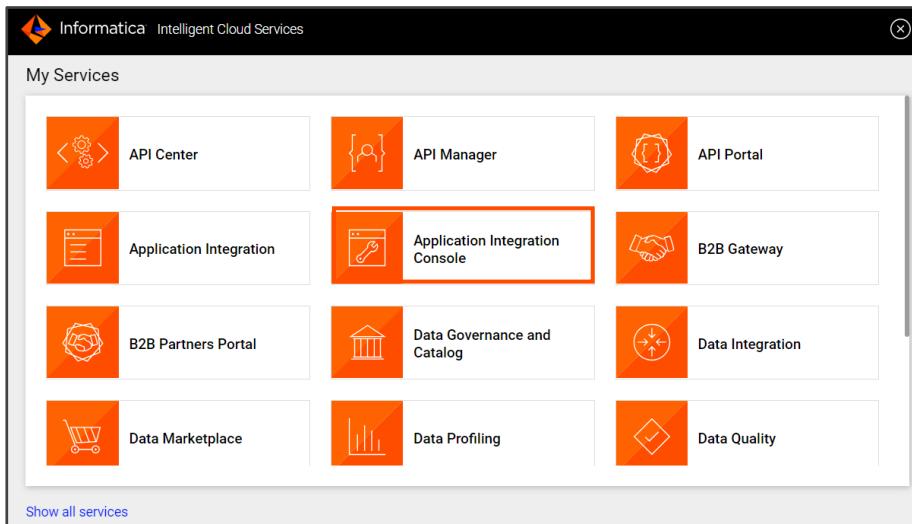
The following URL is a sample service URL with input parameters:

[https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfIEqfZgh7aWGILF/S\\_XX\\_Order\\_Management-1?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&ItemName=newitem1&ItemCount=2](https://na1.ai.dm-us.informaticacloud.com/active-bpel/public/rt/hnLOSoRfIEqfZgh7aWGILF/S_XX_Order_Management-1?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&ItemName=newitem1&ItemCount=2)

161. Open a browser and paste the service URL with input parameters.



162. Navigate to **Application Integration Console**.



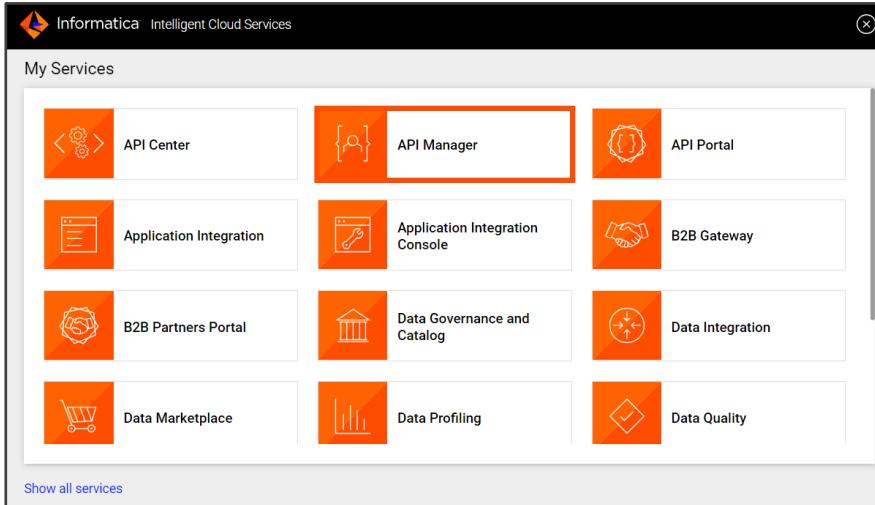
163. Click **Processes**.

164. You can see that the Process status is not Faulted.

| ID                 | Name             | Version | Start Date            | End Date              | Status    |
|--------------------|------------------|---------|-----------------------|-----------------------|-----------|
| 548459046227697664 | Order Management | 1       | 2021/2/22 11:00:26... | 2021/2/22 11:00:26... | Completed |
| 548458362774237184 | Order Management | 1       | 2021/2/22 10:57:43... | 2021/2/22 10:57:43... | Faulted   |

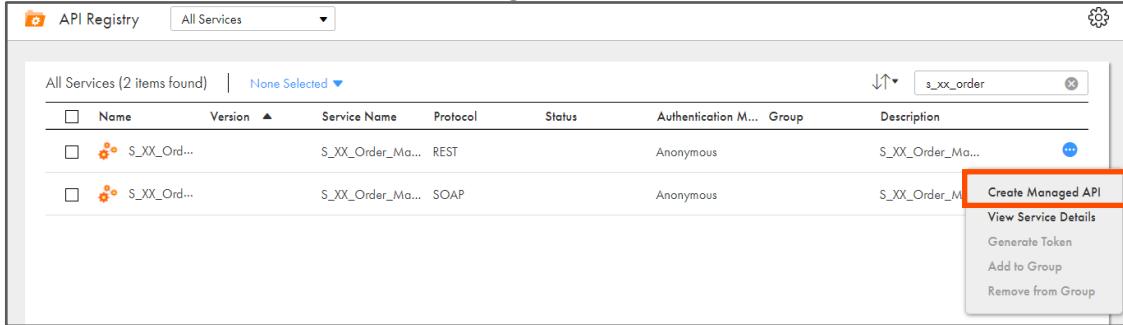
## Create a Managed API

165. Navigate to API Manager.



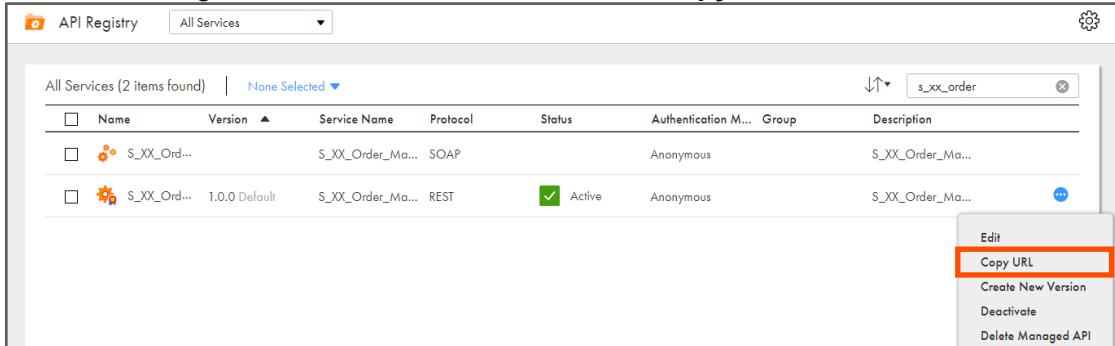
**Note:** If you are prompted to choose an API domain name. Retain the default name and click **Save**.

166. On the API Registry page, for the S\_XX\_Order\_Management process with the REST protocol, click **Actions > Create Managed API**.



167. In the Create Managed API page, keep the default details and click **Create**. The managed API is successfully created.

168. After the Managed API is created, click **Actions > Copy URL**.



169. Open a notepad file, paste the URL, and add the following parameters to the end of URL:

<URL>?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&itemName= item1&ItemCount=2

OR

From the **CAI Lab Prep Files** folder, open the notepad file named **29\_LabGuide\_CAI\_Live Project\_17-1**. Copy the command mentioned under **Step G**.

The URL should look like the following URL:

[https://apigw-pod1.dms.us.informaticacloud.com/t/gvsrtehvymkfpvakw0qkod.com/Order\\_Management?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&itemName=%20item1&ItemCount=2](https://apigw-pod1.dms.us.informaticacloud.com/t/gvsrtehvymkfpvakw0qkod.com/Order_Management?CustomerName=TestConsumer&CustomerEmail=testconsumer@mailinator.com&itemName=%20item1&ItemCount=2)

170. Paste the URL in a browser.

You see the same result as when you invoked the process with the Application Integration service URL.



You have successfully created a process and invoked it.

---

*This concludes the lab.*