

# IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide

Description	IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide
Author(s)	Andrew R. Jones ( <a href="mailto:andrewj@us.ibm.com">andrewj@us.ibm.com</a> ) and Max Weiss ( <a href="mailto:maxwell.g.weiss@ibm.com">maxwell.g.weiss@ibm.com</a> )
Copyright	Copyright © 2024 IBM - V 2.0

# Table of Contents

## Welcome

- Support
- Using the demonstration guide

## IBM Technology Zone environment

- Create a reservation request
- Extend the reservation
- Join the
- Accessing the environments
- Troubleshooting

## Creating an assistant and configuring conversational search

- Access the
- Create your Assistant
- Configure conversational search
- Complete the configuration
- Configure the base large language model
- Testing conversational search
- Troubleshooting

Bring your own search and documents

## Creating a stand-alone OpenSearch instance for document ingestion

- Install the Red Hat OpenShift command line interface utility
- Prepare to ingest documents
  - Log in to the OpenShift cluster from your local terminal
  - Create a working directory
  - Install IBM Certificate Manager on Red Hat OpenShift
  - Install the watsonx Assistant for Z Operator (for OpenSearch)
  - Deploy required secrets and the custom bring-your-own-search (BYOSearch) resources
  - Verify all the required pods are running and get the network route to your
  - Update your assistant with the new
- Troubleshooting

## Installing and using zassist to ingest client documents

- Install the zassist utility
- Ingest client documentation using zassist
- Verify that the assistant is using the ingested documents
- Adjusting the search behavior

## Skills and skill flows

### Getting started with skills and actions

- Environments
  - Watsonx Orchestrate
  - Ansible Automation Platform and Wazi as a Service

### Explore Ansible Automation Platform

- Access the

### Importing skills from Ansible Automation Platform

- Import skills into your assistant

### Connecting skills to your assistant

### Creating actions for your assistant

### Configure the number of input fields

- Create actions

### Verify the job in the Ansible Automation Platform console

- Troubleshooting

### Creating skill flows

- Add the utility skill
- Add the skills to your Personal skills
- Create the skill flow
- Enable the skill flow in your assistant

### Creating custom-built actions

### Importing pre-packaged z/

## Publishing and deploying your assistant

- Publish the assistant
- Configure the live environment
- Connect the skills to the live environment
- Deploy the assistant

## Use cases

### Use case: Retrieve

- Download and customize the watsonx Assistant for Z OpenAP JSON file
- Set the
- Import and publish the pre-packaged skills as an OpenAPI file
- Configure your appURL to connect to the WaaS server
- Test the skill
- Connect the app to your assistant

- Create an action for your assistant
- Create a custom-built action
- Test the custom-built action

#### Use case:

- Create an initial certificate authority (
- Create an
- Renew the
- Create another
- Import the Ansible automations into watsonx Orchestrate
- Verify all the skills are successfully imported and create the app connection.
- Connect the app to the assistant.
- Create a skill flow to retrieve certificate expiration dates.
- Create a skill flow for certificate renewal
- Add the skill flows to the assistant
- Create a custom-built action for
- Test the
- Complete the custom-built skill to renew the certificate
- Run the complete custom-built action

#### Use case:

### Adding other integrations

#### Next steps

- Other resources
- Earn the badge

# Welcome

Welcome to the IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide (lab guide). The lab guide is part of the IBM watsonx Assistant for Z for Technical Sales Level 4 learning plan for IBM and Business Partner Technical Sales and related badge. The learning plan is intended to teach technical sellers and Business Partners how to conduct a proof of experience (PoX) for a client.



## In a fast-paced world, things change...

The products and services may appear differently than what is shown in the lab guide. This can occur if the product or service is updated with a new version.

Responses generated by IBM watsonx Assistant for Z are likely to change over time. The responses you see when you run the queries in this section may differ from the screen images captured in the lab guide.



## Read and follow all the directions.

It is important to read and follow all the documented steps. Skipping steps or sections can cause issues with completing the lab.

Also, invest a few minutes in reading the tips in the [Using the demonstration guide](#) section. The tips will save you time and frustration in completing the lab.

This lab guide covers the setup, configuration, and usage of watsonx Assistant for Z. This lab guide uses the [IBM watsonx Assistant for Z Velocity collection](#) and the three Velocity Pilot lab environments in IBM Technology Zone (ITZ).

The lab guide also enables dedicated lab environments for customized client PoXs and demonstrations. If you are preparing for an actual pilot engagement, refer to the [Pilot Scoping Guide for watsonx Assistant for Z](#) for additional information.

Using the lab guide, you will learn how to:

- Provision the lab environments
- Create an assistant and configuring conversational search
- Configure assistant settings
- Test conversational search
- Deploy a dedicated instance of OpenSearch for client document ingestion (Optional)
- Import skills for z/OS automations
- Connect apps to assistants
- Create assistant actions

- Create skill flows
- Publish and deploy an assistant

In addition, you will import pre-packaged z/OS skills and create custom-built skills to deliver an assistant that handles 3 use cases:

- Retrieving initial program load (IPL) information
- Managing certificates
- Resource Access Control Facility (RACF) administration support



**Not all capabilities of the offering are covered in the lab guide.**

This lab guide covers many features and capabilities of IBM Watson Assistant for Z, but not all. Some uncovered capabilities may be available in ITZ environments, while others may not, such as using skills for OMEGAMON.

## Support

Think something is down? Check the applicable status pages for any known issues such as a site or service not being available:

- [IBM Technology Zone status](#)

For issues with provisioning the ITZ environment for this lab (for example, a failed reservation request due to insufficient quota capacity), open a ticket with ITZ support:

- Web: [IBM Technology Zone ticket system](#)
- Email: [techzone.help@ibm.com](mailto:techzone.help@ibm.com)

For issues related to specific steps found in the demonstration guide after the ITZ environment is provisioned, contact the authors:

- Slack:
  - [#watsonx-assistant-z-technical](#) - IBM only
  - [#wxo-practitioners](#) - IBM only - for questions that are related to the software as a service (SaaS) instance of watsonx Orchestrate
- Email: [andrewj@us.ibm.com](mailto:andrewj@us.ibm.com) and [maxwell.g.weiss@ibm.com](mailto:maxwell.g.weiss@ibm.com)

Business Partners, use the IBM Training live Chat Support service or other support methods that are found on the IBM Training portal [here](#).

## Using the demonstration guide

Use these helpful tips to take full advantage of the IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide.



### Printing the demonstration guide

#### Printed or saved copies can be out of date

The IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide changes regularly to match the IBM watsonx Assistant for Z offering and associated ITZ environment. Printed or saved copies of the demonstration guide can become out-of-date quickly and result in failed steps.

A ready-to-print PDF version of the IBM watsonx Assistant for Z for Technical Sales Level 4 Lab Guide is [here](#).



## Create a reference card for storing user IDs, passwords, and links for your ITZ environments.



You will be creating and using multiple user IDs, passwords, links, and other content throughout the lab. To save time, it is strongly suggested you create a simple text file to store this data so it is readily available and you can easily cut and paste the data when needed. Here is a template to get you started.

watsonx Assistant for Z - Level 4 shortcuts:

watsonx Orchestrate

IBM Cloud account:

IBM Cloud resources: <https://cloud.ibm.com/resources>

watsonx Orchestrate URL:

Assistant name:

Assistant description:

Assistant icon: [https://ibm.github.io/SalesEnablement-L4-watsonx-AssistantForZ/Setup/\\_attachments/Zeeves75x75.png](https://ibm.github.io/SalesEnablement-L4-watsonx-AssistantForZ/Setup/_attachments/Zeeves75x75.png)

OpenShift

Cluster Admin Username: kubeadmin

Cluster Admin Password:

OCP Console:

IBM Cloud container entitlement key:

OS-secret password:

Client ingestion AuthKey:

Wrapper password:

Cluster domain for routes:

Ingestion route (append /v1/query):

Ansible

Ansible Automation Platform URL:

AAP User Name: admin

AAP password:

Wazi User: IBMUSER

Wazi Password:

Wazi URL:

Live Embed

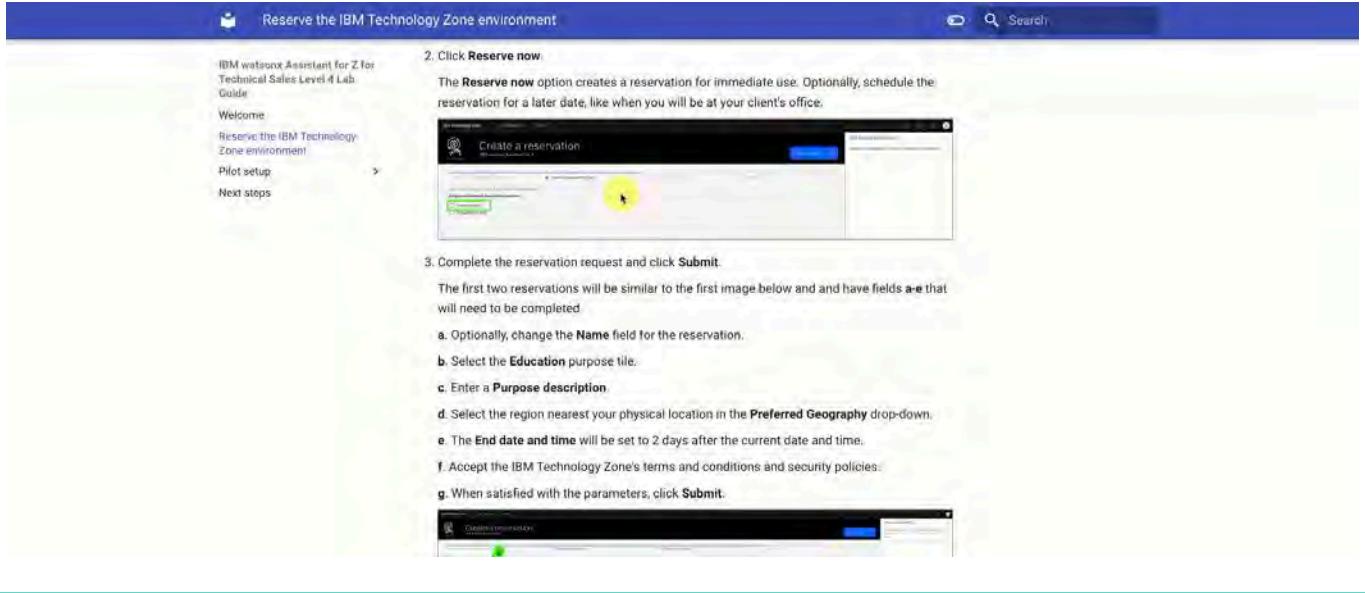
inetegrationID:

region:

serviceInstanceID:

## Viewing images

Images in the demonstration guide can be enlarged by clicking on the image. Press the  key or click the X to dismiss the enlarged image.



2. Click **Reserve now**.

The **Reserve now** option creates a reservation for immediate use. Optionally, schedule the reservation for a later date, like when you will be at your client's office.

3. Complete the reservation request and click **Submit**.

The first two reservations will be similar to the first image below and have fields a-e that will need to be completed.

- Optional, change the **Name** field for the reservation.
- Select the **Education** purpose tile.
- Enter a **Purpose description**.
- Select the region nearest your physical location in the **Preferred Geography** drop-down.
- The **End date and time** will be set to 2 days after the current date and time.
- Accept the IBM Technology Zone's terms and conditions and security policies.
- When satisfied with the parameters, click **Submit**.

## Image highlighting

In some images, the following styles of highlighting are used:

- **Solid highlight box:** This style of box highlights where to click, enter, or select an item.



- **Dash highlight box:** This style of box highlights one of two things: the path to follow to get to a specific location in the user interface, or areas to explore on your own.





## Copying commands and prompts

Copying and pasting commands and prompts from this demonstration guide is easy and can eliminate typographical errors.

Click the highlighted copy icon and then use your operating system's paste function. For example, **[^ Ctrl] + [V]**, or right-click and select **Paste**.

### Prompt 1

What is the APF list in z/OS? Provide a detailed explanation.



## Acronyms and terminology

IBM employees and the tech industry in general, tend to use acronyms. In the demonstration guide, most acronyms will appear with a dashed underline. Hover over the acronym to learn its meaning. A question mark (?) icon will first appear and after a second the tool tip with the acronym's meaning is displayed. Try it here: LPAR.



### Guidance for delivering a demonstration



### Search

IBM watsonx Assistant for Z

Welcome

Reserve the IBM Technology Zone environment

Guidance for delivering a demonstration

Demonstration scenarios >

Next steps

Specific guidance for IBM watsonx Assistant for Z and the ITZ environment

Following the scripts provided in the IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration Guide will help guarantee a successful demonstration. Use these tips to help insure success with IBM watsonx Assistant for Z and the ITZ environment:

- Follow the scripts in the IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration Guide for the automations and skills to execute as expected.

Table of contents

General demonstration guidance

Specific guidance for IBM watsonx Assistant for Z and the ITZ environment



## The Lab Guide table of contents

This **Demonstration Guide** uses a responsive browser-based interface to ensure pages are usable on various devices with different screen sizes. The Demonstration Guide table of contents may be displayed as highlighted in the green dashed box in this image:

The screenshot shows a browser window for the "IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration Guide". The title bar includes the site name and a search function. On the left, a green dashed box highlights the "Table of Contents" sidebar, which lists several sections: Welcome, Guidance for delivering a demonstration, Reserve the IBM Technology Zone environment, Demonstration scenarios (with sub-points like Introduction to scenarios, Scenario 1: Authorized Program Facility, Scenario 2: Certificate renewal, Scenario 3: Db2 versioning, Scenario 4: Initial Program Load on Z, Additional IBM Z related prompts, Summary, Next steps, Printing the guide, and Instructions). The main content area displays the "Welcome" page, which includes an introduction to the course, information about end-user capabilities, and a note about additional features like Velocity Pilot.

However, if the browser window is sized smaller, the table of contents can be accessed by clicking the main menu icon (≡):

The screenshot shows the same browser window after the main menu icon (≡) was clicked. The sidebar is now collapsed, and the main content area displays the "Welcome" page. The main menu icon is highlighted with a green box.

Click the main menu icon (≡) to expand the table of contents.

Continue to the [Reserve the IBM Technology Zone environments](#) section to begin the journey to obtain the IBM watsonx Assistant for Z Technical Sales Advanced badge.

# IBM Technology Zone environment

To enable sellers to learn how to deliver client pilots of IBM Watsonx Assistant for Z, three environments are available in IBM Technology Zone (ITZ). The environments are part of the Watsonx Assistant for Z Velocity lab collection and can be found in the [IBM Watsonx Assistant for Z](#) collection.

- **Watsonx Assistant for Z lab – Watsonx Orchestrate:** provides a dedicated environment on IBM Cloud where you can create and configure the assistant, set up conversational search, import skills, and configure actions.
- **Ansible Automation Platform (AAP) & z/OS:** provides a pre-configured instance of AAP and Wazi z/OS. This environment includes Ansible playbooks, which you can import as skills within Watsonx Orchestrate and connect to your assistant. Preinstalled templates for various use cases are also available (covered in later sections). Learn more about AAP [here](#). Learn more about Wazi, [here](#).
- **Single Node OpenShift with NFS storage:** provisions a single-node Red Hat OpenShift cluster (SNO) on IBM Cloud. This cluster installs a dedicated instance of OpenSearch for Watson Assistant for Z, enabling ingestion of client-supplied documents.-



All activities in this lab guide are required.

To earn the IBM Watsonx Assistant for Z Technical Sales Advanced badge and complete the Level 4 learning plan, you must provision all three ITZ environments and finish every section in the lab guide. Disregard any statements in the ITZ collection that suggest optional environments or tasks.

Follow the instructions to create new reservation requests, extend the reservations, and access the ITZ demonstration environments. Provisioning the SNO environment in ITZ can take several hours, while the other two environments typically provisioning in under 30 minutes.

## Create a reservation request

1. Click each of the links that follow to open a browser to the reservation pages of the **IBM Watsonx Assistant for Z** ITZ environments.



You may be asked to authenticate to IBM Technology Zone.

The steps to authenticate to ITZ are not detailed here as they may vary between users.

[Watsonx Assistant for Z lab – Watsonx Orchestrate - reservation page](#)

[Ansible Automation Platform \(AAP\) & z/OS - reservation page](#)

[Single Node OpenShift with NFS storage - reservation page](#)

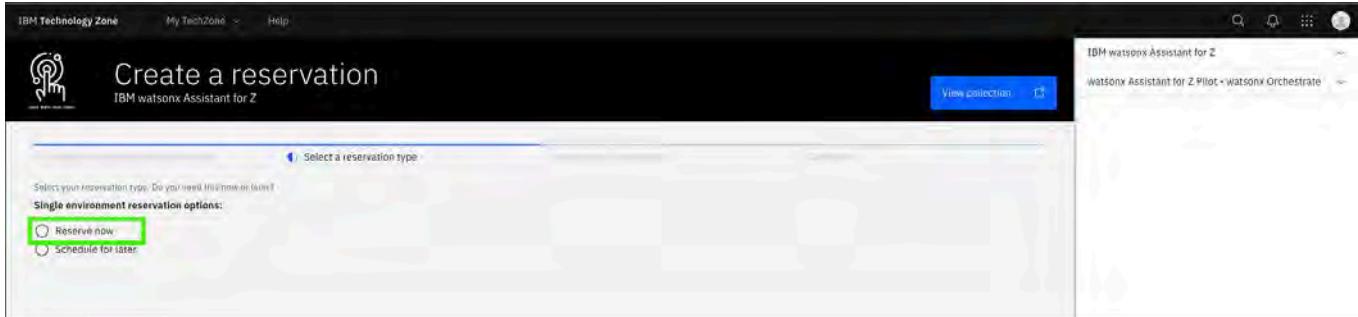


**The next two steps are for one of the three environments. Repeat for all three environments.**

Follow the steps to create a reservation in ITZ for all three environments.

## 2. Select Reserve now.

The **Reserve now** option creates a reservation for immediate use. Optionally, schedule the reservation for a later date, for example, when you are at your client's office to start a pilot.



## 3. Complete the reservation request form and then click Submit.

The first two reservations are similar to the first image and have fields **a-e** that need to be completed.

- Name: specify a name for the reservation (optional).
- Purpose: select the **Education** purpose tile.



**For client pilots...**

For client pilots, set the **Purpose** field in the reservation to **Pilot** and provide an opportunity number to receive a longer reservation.

- Purpose description: enter a description, for example: Level 4 education.
- Preferred geography: select the region nearest to your physical location for improved performance and reduced network latency.
- End date and time: the initial maximum will be set to a specific number of days (typically two, but in some cases longer) after the current date and time. Instructions follow to extend the reservation end date.
- Accept the IBM Technology Zone's terms and conditions and security policies.
- Click **Submit**.

The screenshot shows the 'Create a reservation' interface. At the top, there are tabs for 'My reservations' and 'Help'. Below the title 'Create a reservation' is the subtitle 'IBM Watson Assistant for Z'. There are two main sections: 'Select a reservation type' (with 'FIPS and non-FIPS' options) and 'FIPS and non-FIPS reservation'. The 'Location' dropdown is set to 'EDGAR'. The 'Project' dropdown is set to 'ibm\_z\_techzone'. Under 'Cluster network', the value '10.128.0.0/14' is highlighted with a green circle labeled 'd'. Under 'Service network', the value '172.30.0.0/16' is highlighted with a green circle labeled 'e'. A green circle labeled 'f' points to the checkbox 'I agree to IBM Technology Zone's terms and conditions and security policies'. A green circle labeled 'g' points to the 'Submit' button at the bottom right.

In addition to the preceding fields, the reservation for the **Single Node OpenShift with NFS storage** has these additional fields:

- h.** OCP/Kubernetes cluster network: leave the default setting of **10.128.0.0/14**.
- i.** Enable FIPS security: leave the default setting of **No**. Learn more about the Federal Information Processing Standards (FIPS) [here](#).
- j.** Master single node flavor: select **16 vCPU x 64 GB - 300 GB ephemeral storage**.
- k.** OpenShift version: select **4.14**.
- l.** OCP/Kubernetes service network: leave the default setting of **172.30.0.0/16**.
- m.** Accept the IBM Technology Zone's terms and conditions and security policies.
- n.** Click **Submit**.

IBM Technology Zone   My TechZone   Help

Select a date: 11/01/2024   Set a time: 11:53 AM   Location: America/Chicago

Retention policy: Recommended 2 days. Retention can be increased up to 2 days or decreased by 2 days increments up to 3 days total. Max time 6 days total.

GC01v4 (medium) Cluster Network: 10.128.0.0/16

Enable FIPS Security: No

Master: Single node flavor: 16 vCPU x 64 GB - 300 GB ephemeral storage

OpenShift Version: 4.14

DDN WebStorage Service Network: 172.30.0.0/16

Annotations:

I agree to IBM Technology Zone's Terms & Conditions and End User License Agreement

**Submit**

The screenshot shows the configuration interface for creating a new cluster. The left side displays the configuration details: date (11/01/2024), time (11:53 AM), location (America/Chicago), retention policy (Recommended 2 days), and network settings (GC01v4 (medium) Cluster Network: 10.128.0.0/16, DDN WebStorage Service Network: 172.30.0.0/16). The right side shows the 'Policy' section with a checkbox labeled 'I agree to IBM Technology Zone's Terms & Conditions and End User License Agreement' and a 'Submit' button.

During the provisioning process, multiple emails are sent to you from ITZ as the provisioning process runs. One email states the reservation is provisioning and the other email states that the environment is **Ready**.

In rare cases, the provisioning process can fail. If you receive an email stating the reservation failed, try again by repeating Steps 1-3 for the environment that failed to provision. In addition, review the [Troubleshooting](#) section that follows. If issues continue, open an [ITZ support ticket](#) by using the methods that are mentioned in the [Support](#) section.

## Extend the reservation

When the reservations are in the **Ready** state, you can extend each reservation beyond its original end date. The duration of the extension will vary by reservation.

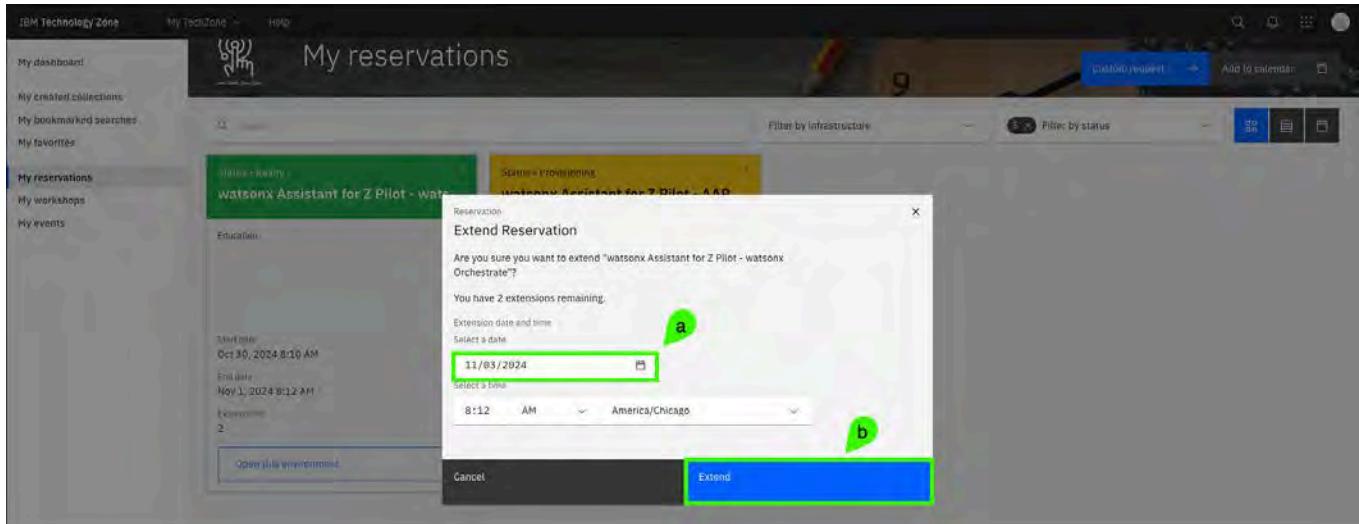
1. In the IBM Technology Zone portal, expand **My TechZone** and select **My Reservations**.

The screenshot shows the IBM Technology Zone portal interface. At the top left, there's a navigation bar with 'IBM Technology Zone', 'My TechZone ▾', 'Help', and a search bar. Below the navigation bar is a sidebar with links: 'My dashboard', 'My created collections', 'My bookmarked searches', 'My favorites', 'My reservations' (which is highlighted with a green box), 'My workshops', and 'My events'. The main content area has a heading 'Welcome to Technology Zone, Andrew' and a sub-section titled 'Certified Base Images' with a sub-sub-section 'Best starting point to build'. There are also sections for 'Watsonx Assistant' and 'Watsonx Assistant for Z Pilot - AAP ...'. On the right side, there's a decorative graphic of a hand interacting with a circular interface.

2. Click the overflow icon (≡) on the reservation tile and select **Extend**.

The screenshot shows the 'My reservations' page. The sidebar on the left lists 'My dashboard', 'My created collections', 'My bookmarked searches', 'My favorites', 'My reservations' (selected), 'My workshops', and 'My events'. The main area displays two reservation cards. The first card, for 'Watsonx Assistant', has a status of 'Status - Ready' and a start date of 'Oct 30, 2024 8:10 AM'. The second card, for 'Watsonx Assistant for Z Pilot - AAP ...', has a status of 'Status - Provisioning' and a start date of 'Oct 30, 2024 8:12 AM'. Both cards have an 'Open this environment' button at the bottom. A context menu is open over the first reservation card, with the 'Extend' option highlighted by a green dashed box.

3. Click the **Select a date** option, (a) specify the date to extend to, and then (b) click **Extend**.



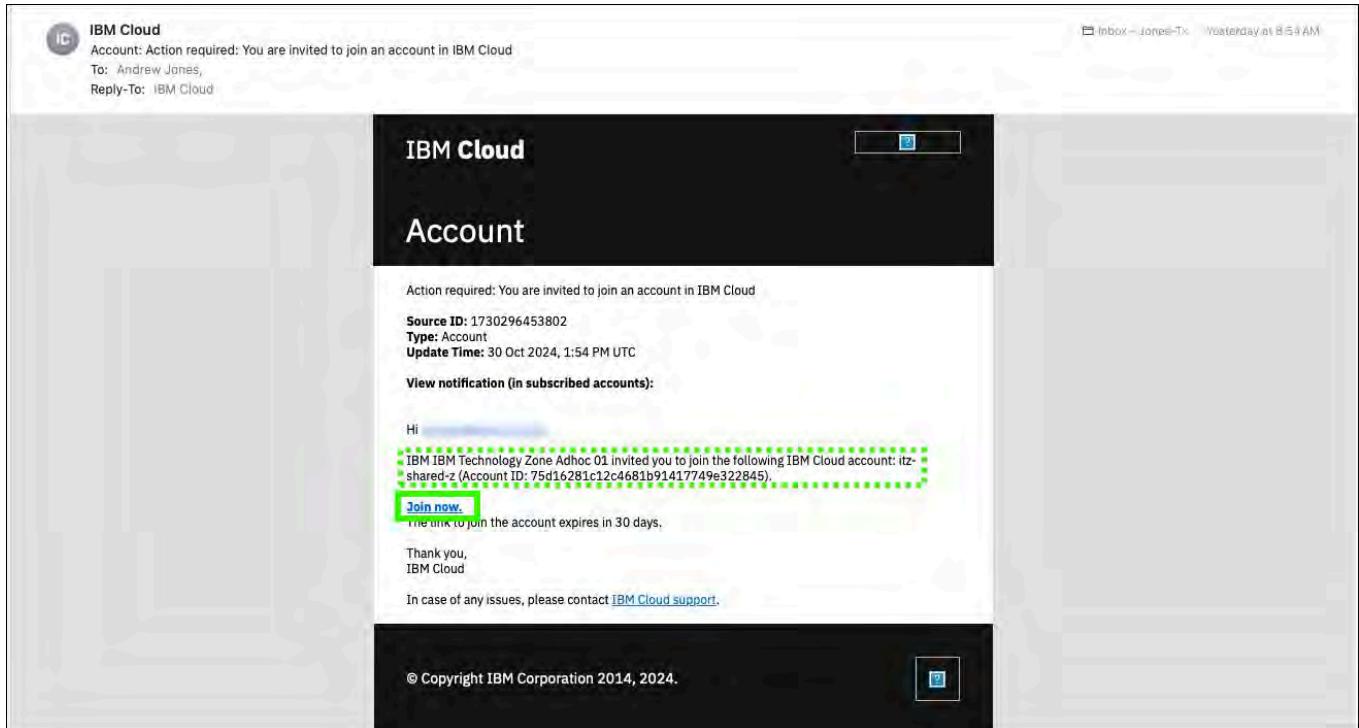
If you anticipate needing more time, repeat Steps 5-6 to extend the reservation to the maximum allowed. Repeat these steps for the other two reservations.

## Join the ITZ IBM Cloud account

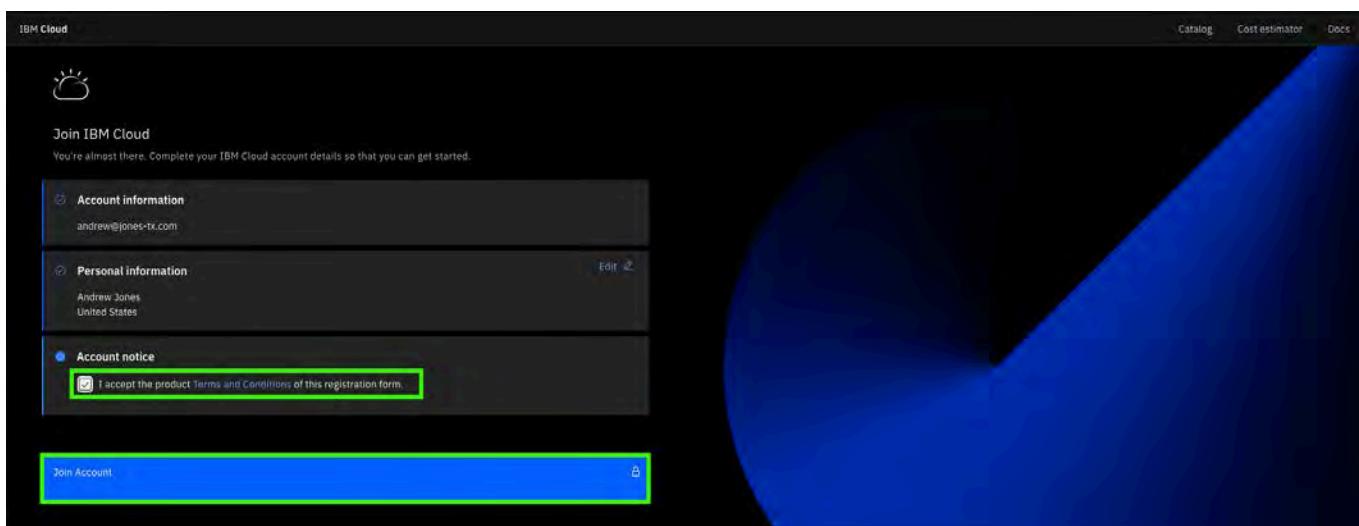
Both the **watsonx Assistant for Z lab – watsonx Orchestrate** and the **Ansible Automation Platform (AAP) & z/OS** environments add you to an IBM Cloud account while your reservation is active. During the provisioning process of these ITZ environments, you receive two emails from IBM Cloud.

You only need to accept the invitation to the **watsonx Assistant for Z lab – watsonx Orchestrate** environment.

1. Open the email from **IBM Cloud** and click the **Join now** links.



2. In the **Join IBM Cloud** browser window that opens, select the **I accept the product Terms and Conditions** of the registration form, and then click **Join Account**.

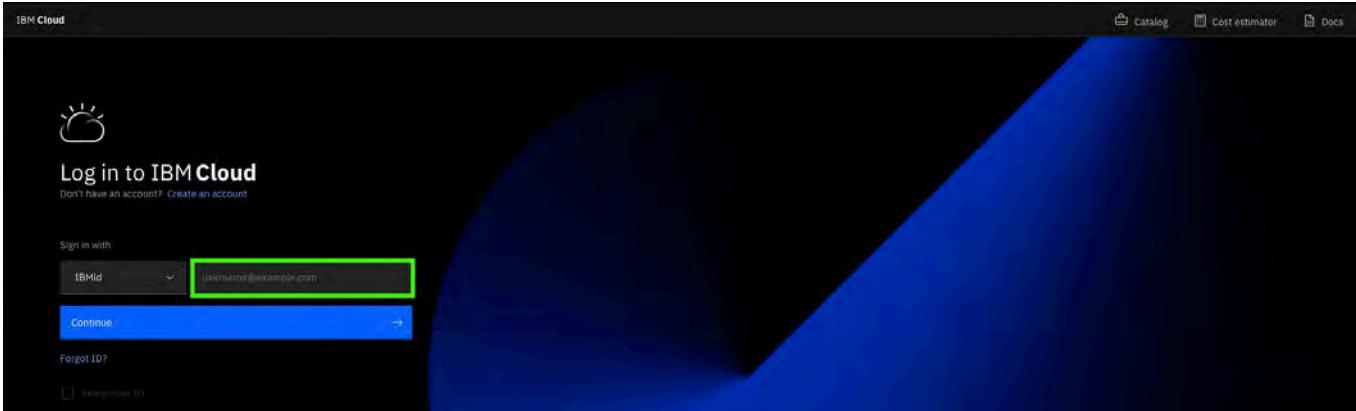


After joining the account, verify that the account appears in your available account list in the IBM Cloud portal.

- Click the following link to open a browser to the IBM Cloud portal.

### IBM Cloud portal

- Follow the directions to complete the authentication to IBM Cloud using the same email address you used to login to ITZ. The login steps vary depending on any two-factor authentication methods enabled.

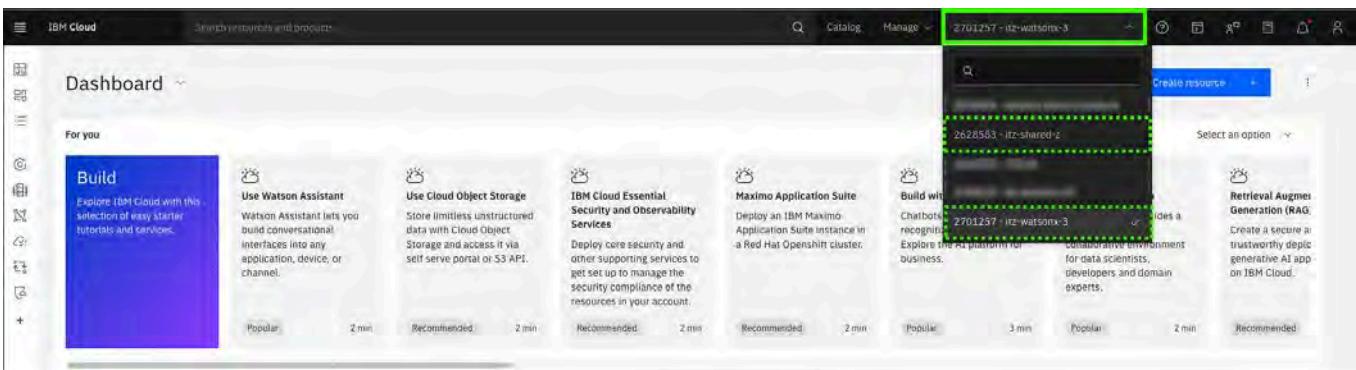


- Click the **account** menu and verify access to the IBM Cloud account listed in your ITZ reservation.



#### The account may be different.

The account name should align with the account named in the invitation email you received.





## Does your IBM Cloud portal view look different?

If your IBM Cloud portal looks different from the images above, it could be because the IBM Cloud portal has gone through a design change, or your browser window is set to smaller size. Instead of the current selected account appearing in the top menu, you may see this **change account** icon: . Click this icon to view the list of accounts you can access.

The screenshot shows the IBM Cloud portal dashboard. At the top, there is a navigation bar with icons for search, catalog, manage, and other account-related functions. The 'Manage' dropdown is currently selected. To its right is a green-outlined box highlighting the 'Change account' icon, which is a person icon with a gear. Below the navigation bar, the main dashboard area is visible, featuring a 'Dashboard' section, a 'For you' section, and several service tiles like 'Build', 'Use Watson Assistant', 'Use Cloud Object Storage', and 'IBM Cloud Essential Security and Observability Services'. A 'Create resource' button is also present.

## Accessing the environments

Each reservation provides access to its respective environment. Details for accessing each environment are provided in the **Pilot setup** sections that follow in the lab guide.

After all three reservations are in the **Ready** state and you accept the invitations to the IBM Cloud accounts, proceed to the next section to complete the pilot setup.

## Troubleshooting



### If your reservation for the Single Node OpenShift environment fails...

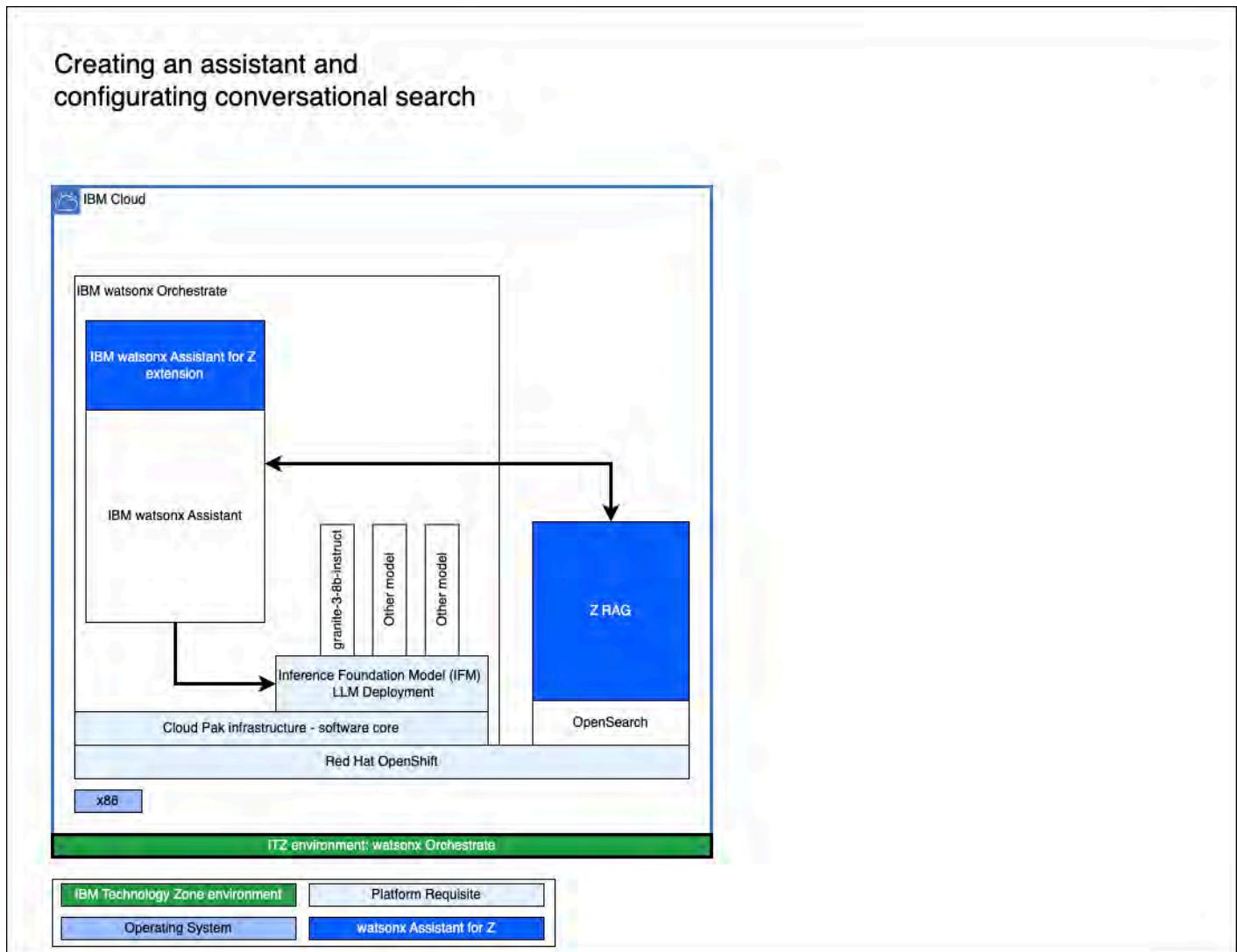
If your reservation for the Single Node OpenShift environment fails, try selecting one of the **eu-gb region** options as the **Preferred Geography**.

# Creating an assistant and configuring conversational search

Watsonx Orchestrate allows you to create and configure an assistant with conversational search capabilities.

Configure your assistant to use conversational search by using a hosted [OpenSearch](#) instance. The pre-configured instance of Watsonx Orchestrate in IBM Technology Zone (ITZ) boasts over 220 knowledge sources and supports Retrieval Augmented Generation (RAG). The large language model (LLM) providing conversational AI augments this knowledge based on IBM Z documentation, generating IBM Z context-aware responses to queries with content-grounded knowledge.

A high-level, logical architecture of the environment is illustrated in the following diagram.



Access the [ITZ IBM Cloud account](#) for the Watsonx Assistant for Z Pilot environment

1. In the IBM Technology Zone portal, expand **My TechZone** and select **My Reservations**, or click the following link.

[ITZ My reservations](#)

Welcome to Technology Zone, Andrew

Certified Base Images

Best starting point to build

These images are 'TechZone Certified' and represent the best starting point for building new content, showing clients how easy it is to deploy IBM Technology from scratch, or testing custom configurations.

## 2. Click the **watsonx Assistant for Z Pilot - watsonx Orchestrate** tile.

Status	Name	Description
Ready	watsonx Assistant for Z Pilot - AAP & z/OS	Education
Ready	watsonx Assistant for Z Pilot - watsonx O...	Education
Ready	Single Node OpenShift (VMware on IBM C...	Education

## 3. Record the ITZ IBM Cloud account name associated with the reservation.

Did you read the tip on the welcome page about creating a reference card? Check it out [here](#).

Watsonx Assistant for Z Pilot - Watsonx Orchestrate

Date: Nov 5, 2024 6:03 AM - Nov 11, 2024 6:03 AM Expires in: 3 days, 22 hours, 45 minutes Extend limit: 0 Status: Ready

**Purpose**

- Purpose: Education
- Opportunity Product(s): Opportunity ID(s):
- Customer(s): L4 training

**Environment**

Reservation ID: 672d09a1a885062b991e081

Resource name: watsonx-orchestrate

Cloud Account: ITZ-WATSONX-036

Region: us-south

Customer data: false

Timeout limit: 10800

**Reservation Details**

IBM Cloud Login: <https://cloud.ibm.com/resource>

## 4. Click the **IBM Cloud Login** link.

**Purpose**

- Promote
- Education
- Opportunity ID(s): Opportunity ID(s):
- Customer(s):

**Environment**

Reservation ID: 072d091a8f85062fb91e081	Type: IBM Cloud
Request method: watsonx-orchestrate	Transaction ID: 115897c9-58a1-417-a9c-b18dc2a97590
Region: us-east1	Grid: americas
Customer data: false	Datacenter: data30
Job running limit: 10800	Environment: watsonx-orchestrate-wusdf
	Timeout: 1440m

**Reservation Details**

IBM Cloud Login  
<https://cloud.ibm.com/resources>

**Steps to authenticate to IBM Cloud are not illustrated here.**

You may need to authenticate to IBM Cloud after clicking the link. These steps are not shown here as they may vary by individual.

- Verify that the current IBM Cloud account is the same as the account name recorded in step 3. If the account is not the same, switch to the proper account.

**Note:** The formatting of the name can appear differently than what is shown in the ITZ reservation.

**Resource list**

IBM Cloud Search resources and products... Catalog Manage 2953593 - itz-watsonx-036

Name	Group	Location	Product	Status	Tags
Filter by name in (0 accounts)	Filter by group or tag...	Filter...	Filter...	Filter...	Filter...
Compute (0)					
Containers (0)					
Networking (0)					
Storage (0)					
Converged infrastructure (0)					
Enterprise applications (0)					
AI / Machine Learning (11)					
Analytics (0)					

If the proper account is not listed, click the account drop down and select the proper account.

**Note:** If your browser window is narrow, the account drop down can be depicted with the Switch Account icon ().

The screenshot shows the IBM Cloud Resource list interface. On the right side, there is a sidebar with a 'Create resource' button. The entire interface is highlighted with a green border.

## Create your Assistant

1. Click the **Resources** icon (☰).

The screenshot shows the IBM Cloud Dashboard. The 'Build' section on the left is highlighted with a green border. The top navigation bar shows the account ID '2701257 - liz-watsonx-3'.

2. Expand the **AI / Machine Learning** section and click the **watsonx Orchestrate** instance listed (the instance name is different than shown in the following image).

The screenshot shows the IBM Cloud Resource list interface. The 'Enterprise applications' section is expanded, and the 'AI / Machine Learning' sub-section is also expanded. A specific instance named 'Watsonx-orchestrate-erspw' is highlighted with a green border. The top navigation bar shows the account ID '2701257 - liz-watsonx-3'.

3. Click **Launch watsonx Orchestrate**.

The screenshot shows the IBM Cloud interface with the service instance 'itzwxo-2700039nft-erspw' selected. In the 'Manage' section, there's a 'Service credentials' table with columns for 'API key' and 'URL'. Below the table, there are two buttons: 'Launch Watson Orchestrate' (highlighted with a green box) and 'Getting started tutorial'. To the right, under the heading 'Plan', it says 'Essentials Plan'. At the top right, there are 'Details' and 'Actions' buttons.

4. Click the **AI assistant builder** tile to start creating a new assistant.

The screenshot shows the 'IBM Watsonx Orchestrate' interface. On the left, there's a 'Welcome, Andrew Jones!' message and a 'Try Skills in Chat' button. In the center, there are several tiles: 'Build' (with a sub-section for 'AI assistant builder'), 'Skill studio', and 'Upgrade'. The 'AI assistant builder' tile is highlighted with a green box. On the right, there's a 'Standard plan | Skill studio' section with a 'Learn more' button.

5. Enter a name and optional description for your assistant and click **Next**.

The screenshot shows the 'Create your first assistant' form. It has tabs for 'Create', 'Personalize', 'Customize', and 'Preview'. The 'Create' tab is active. The 'Assistant name' field contains 'Zeeves' and the 'Description (optional)' field contains 'Zeeves, the Jeeves for everything IBM Z.'. Both fields are highlighted with green boxes. At the bottom right, there's a 'Next' button.

6. Complete the **Personalize your assistant** form and click **Next**.

Explore the personalization options. In creating an assistant for a client pilot, consider specifying attributes that align with the client's business.

- Select **Web**.
- Select the industry of your choice.
- Select the role of your choice.
- Select the need of your choice.

Welcome to AI assistant builder

**Personalize your assistant**

Tell us where your assistant will live  
You may add multiple channels from your dashboard.

Where do you plan on displaying your assistant?  
**a** Web

Tell us about yourself  
This information will be used to personalize your assistant's experience.

Which industry do you work in?  
**b** Software

What statement describes your needs best?  
**c** I want to automate common tasks in a natural way

**d** I'm looking for a size 9 in white

This is what your customers will experience

watsonx Assistant

Do you have the Speed Demons in stock?

The Speed Demons are in stock at our Downtown and Northgate locations, which are both within 5 miles of you.

What size and color do you need?

I'm looking for a size 9 in white

Great news! The Speed Demons are available in white in a size 9.

You can purchase them for curbside pickup or we can ship them to you. Which would you prefer?

I'll pick them up! Ship them to me!

## 7. Complete the **Customize your chat UI** form and click **Next**.

Explore the customization options. When creating an assistant for a client pilot, consider specifying attributes that align with the client (for example, colors and logos).

Welcome to AI assistant builder

**Customize your chat UI**

Update the style to match your brand and your website. You can change these settings later. A developer can also add more advanced styling changes with code. [Learn more](#)

Assistant name or alias by customer  
**a** Zeeves

Primary color  
**b** Dark

Size  
**c** The size of the webchat on this page will not change by updating these fields.

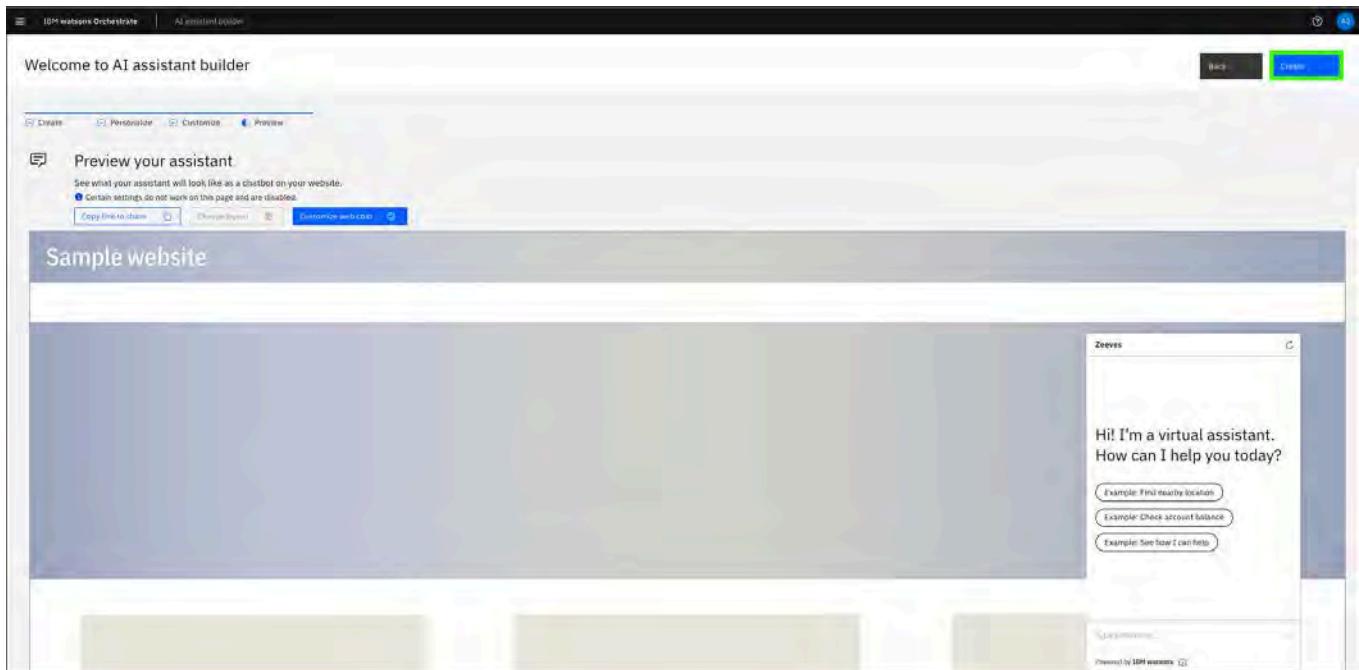
Change avatar image

Zeeves

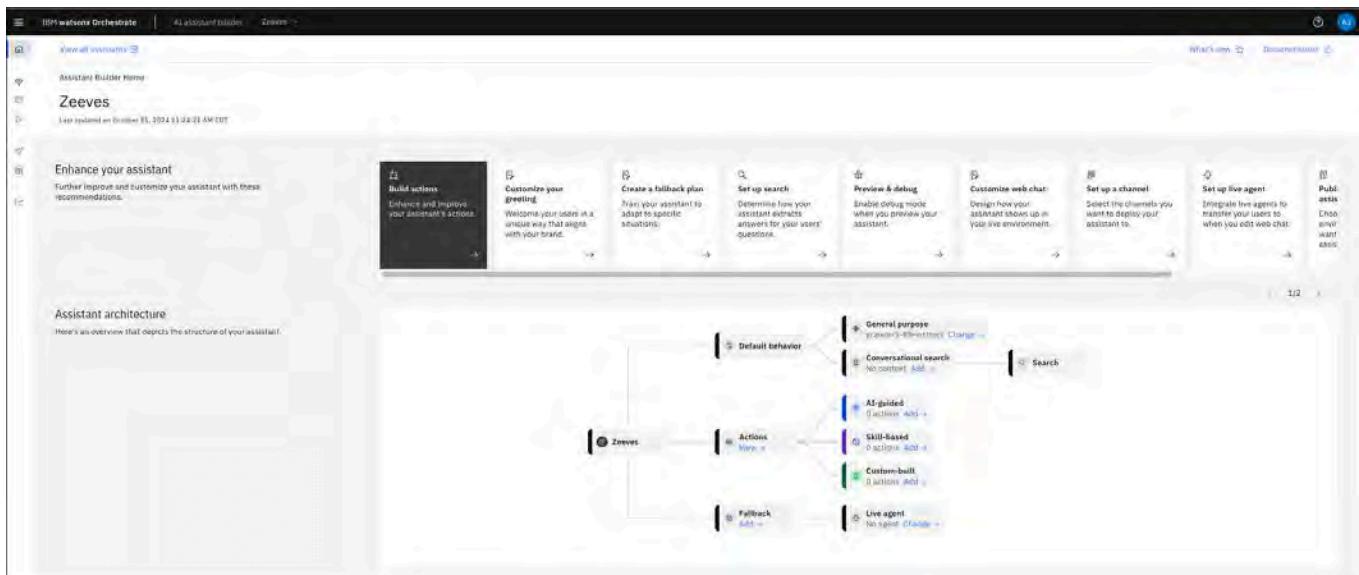
Hi! I'm a virtual assistant.  
How can I help you today?

Powered by IBM Watson

## 8. Preview your assistant and then click **Create**.



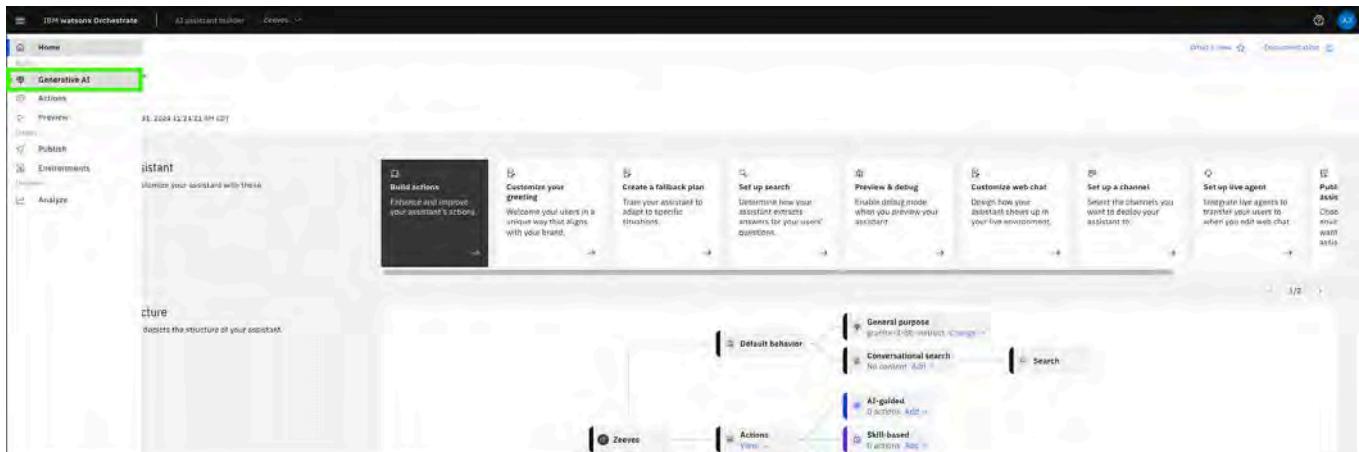
The assistant is now created.



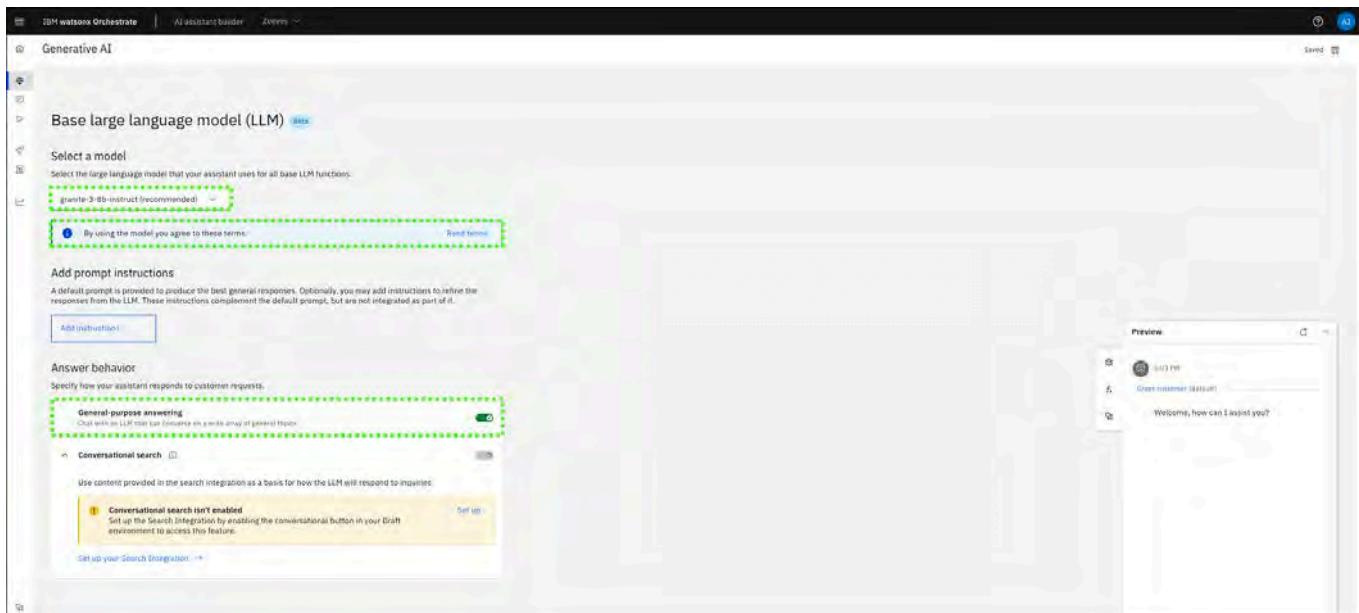
## Configure conversational search

In the next steps you will be to configure **conversational search** for your assistant that uses a hosted instance of OpenSearch.

1. Click **Generative AI** menu item (💡) in the left navigation.

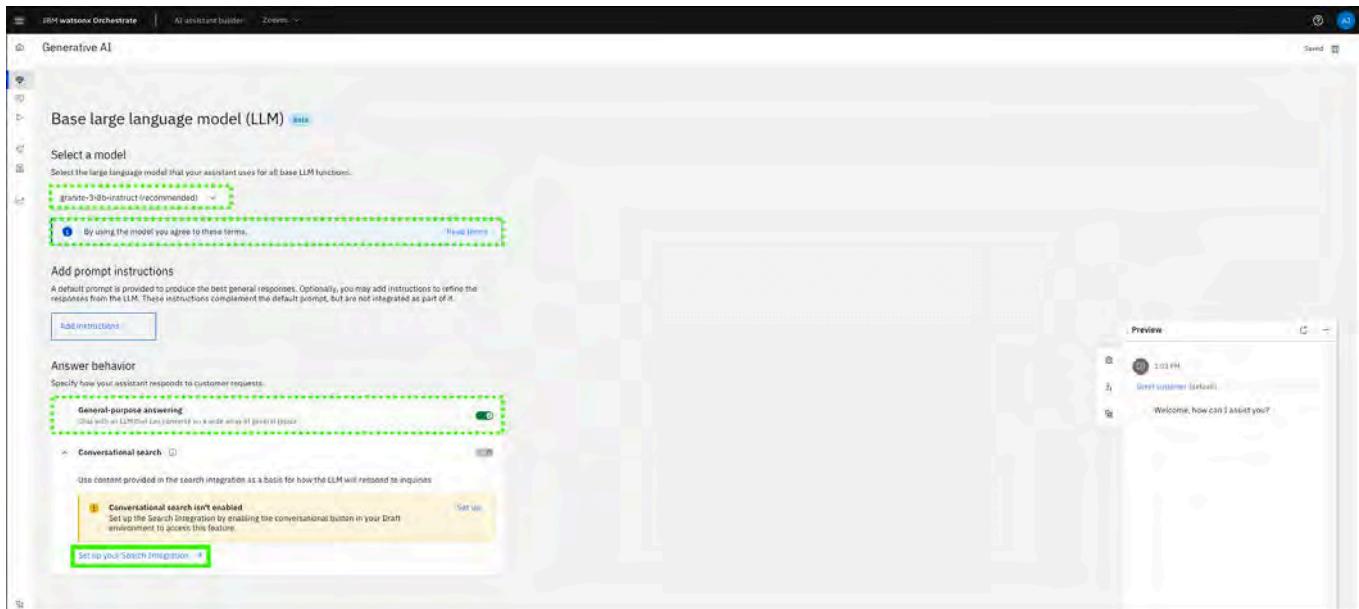


## 2. Select **granite-3-8b-instruct** for the base large language model (LLM) settings.

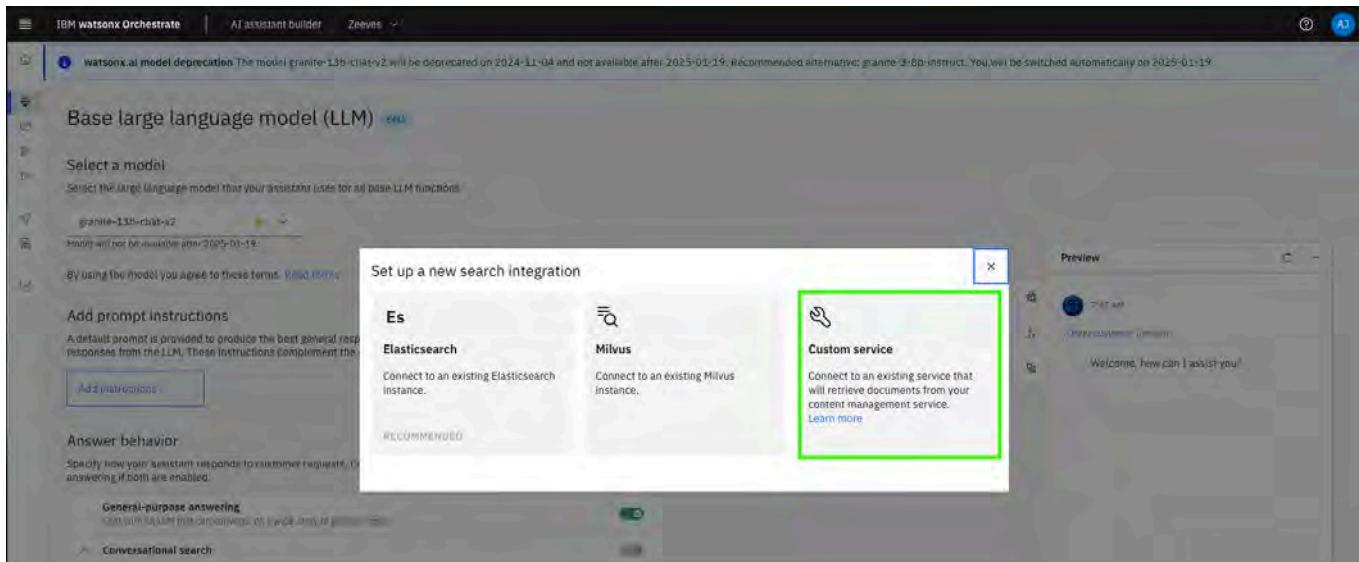


## 3. Click **Set up your Search Integration**.

By default, conversational search is not enabled when an assistant is created. Conversational search takes priority over general-purpose answering if both are enabled. Learn more about conversational search in watsonx [here](#).



#### 4. Click Custom service.



#### 5. Complete the **Custom service (a-e)** form and then click **Next (f)**.

a. Select **By providing credentials**.

b. Enter the following value in the **URL** field (use the copy icon to avoid typographical errors). This is the **URL** for the shared **OpenSearch** instance. In later sections, you create and customize a dedicated instance.

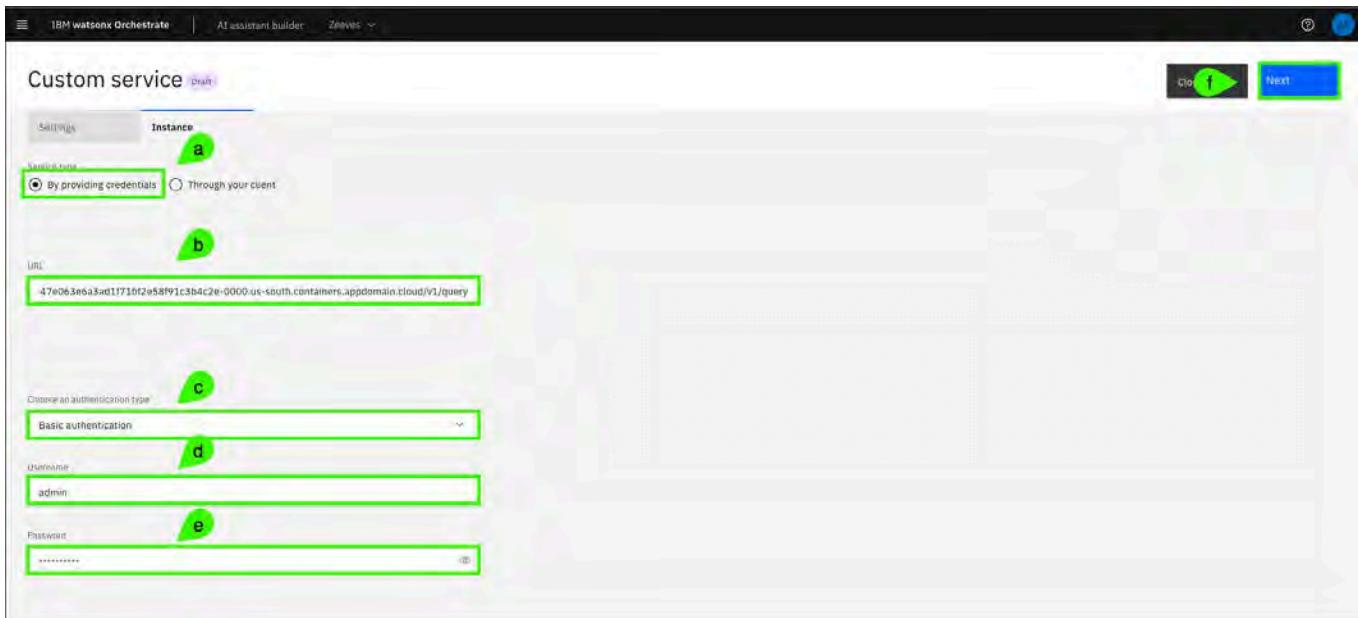
```
https://wxa4z-opensearch-wrapper-wxa4z-demo-v2-1-0.wxo4z-opc-opensearch-clus-47e063e6a3ad1f71bf2e58f91c3b4c2e-0000.us-south.containers.appdomain.cloud/v1/query
```

c. Select **Basic authentication** in the **Choose an authentication type** drop-down list.

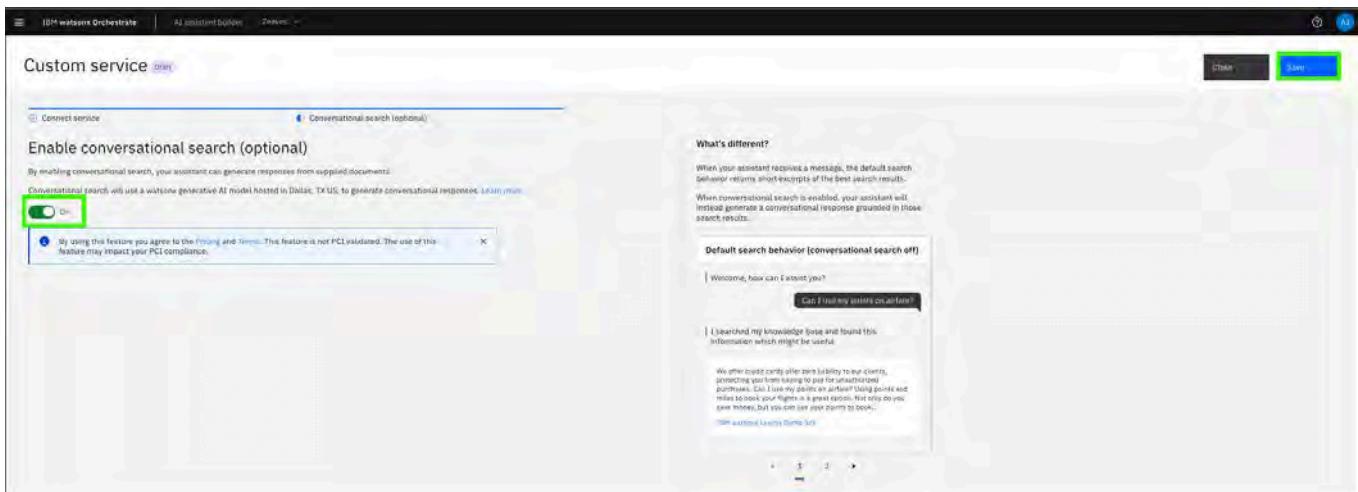
d. Enter **admin** in the **Username** field.

e. Enter **secureP@ssw0rd!** in the **Password** field.

```
secureP@ssw0rd!
```



## 6. Enable conversational search and then click Save.



## 7. Update the conversational search **custom service** settings based on your requirements.

**Note:** The **Settings** page is divided into two sections in the following images to enhance the visibility of the screen captures.

Learn more about these **custom service** settings [here](#).

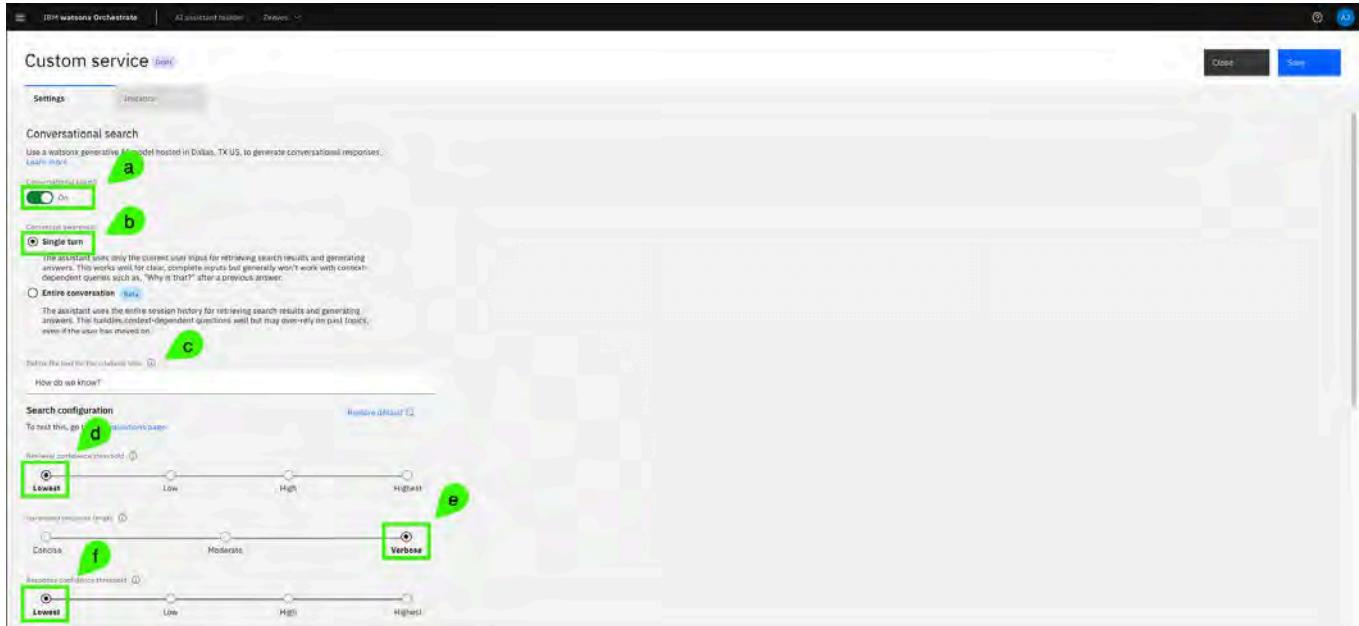
The following settings are proven to work well. You can experiment with these settings to see how they affect queries for your client's pilot.

- a. Enable **Conversational search**.
- b. Select **Single turn. Multi-turn conversation** (by selecting **Entire conversation**) is supported by the offering, but has not been fully included in the lab guide. See the callout in the [Testing conversational search](#) section below.
- c. Specify the text that appears to instruct the user to expand the list of citations in the assistant (except web chat client).
- d. Select **Lowest** for the **retrieval confidence threshold** setting. This setting checks the confidence of the retrieved citations before a response is generated.

e. Select **Verbose** for the **generated response length**. This setting affects the average response length.

Depending on user input, variations from the selected length can occur.

f. Select **Lowest** for the **response confidence threshold**. This setting checks the confidence of the generated citations after the response is generated.



g. Keep the default setting of **All** for the listing of citations.

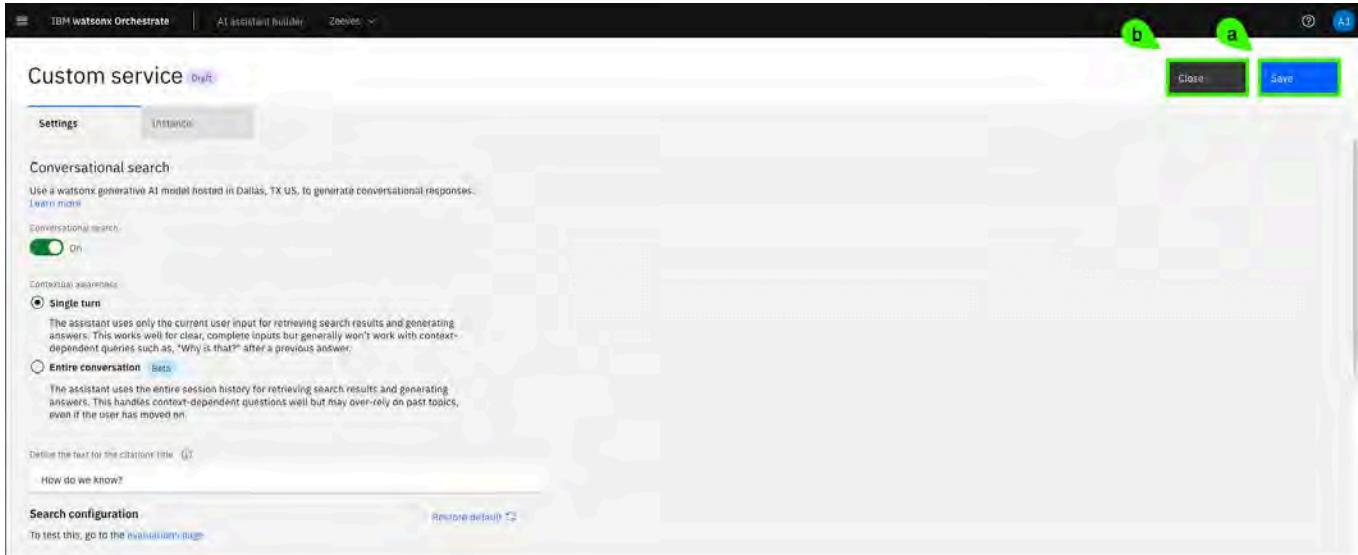
h. Keep the **Default filter** field empty.

i. The **Metadata** field provides a way to adjust your assistant's behavior during conversational search for your OpenSearch instance. This option is explored in detail in the [Installing and using zassist to ingest client documents](#). Leave the field empty for now.

j. The **Search display text** options specify the default text displayed when no results are found or when connectivity issues to the backend search service occur. You can keep the defaults or customize the service.



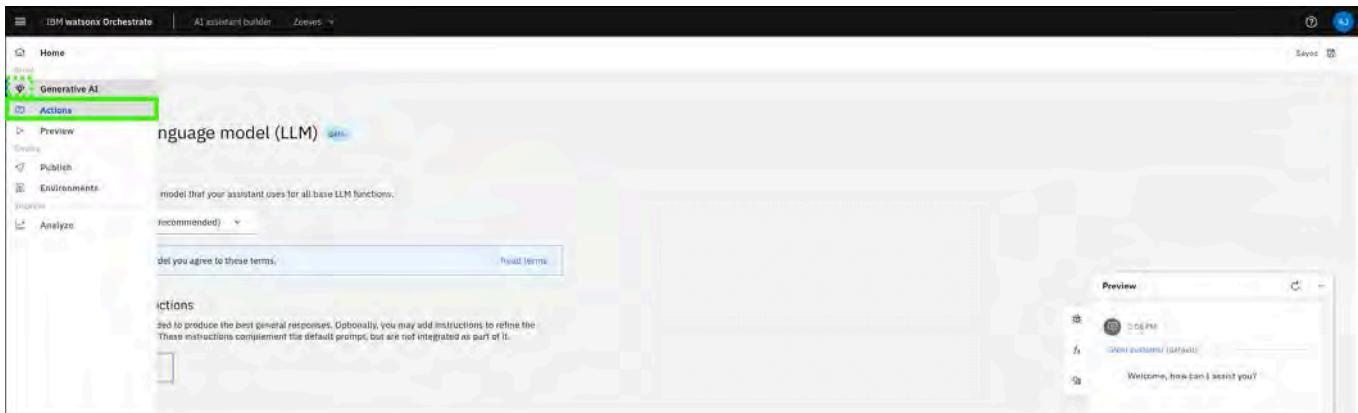
8. Click **Save** (a) and then click **Close** (b).



## Complete the configuration

After you save and close the **Conversational search** configuration page, a few more configurations are needed to get the best experience from your conversational chat. Details on these settings are available [here](#).

1. Hover over the **Generative AI** icon (🌐) in the left navigation and click **Actions**.



2. Click **Set by assistant** under the **All items** menu.



### 3. Click No matches.

The screenshot shows the 'Actions' section of the IBM Watsonx Orchestrator interface. A table lists several actions: 'Get customer' (Last edited 3 hours ago), 'Trigger word detected' (3 hours ago), 'No matches' (highlighted with a green box), 'Fallback' (3 hours ago), and 'Saved responses' (3 hours ago). The 'Status' column indicates each action is active.

### 4. Click Step 1 under Conversation steps.

The screenshot shows the 'Conversation steps' editor for the 'No matches' action. Step 1 is selected and highlighted with a green box. The step details show a condition 'No matches count > 1' and a response 'I'm afraid I don't understand. Please rephrase your question.' Below this, there are sections for 'Action complete' and 'Additional training examples (optional)'. Step 2 is also visible with its own condition and response.

### 5. Select without conditions (a) in the Is taken drop-down menu and then click Clear conditions (b).

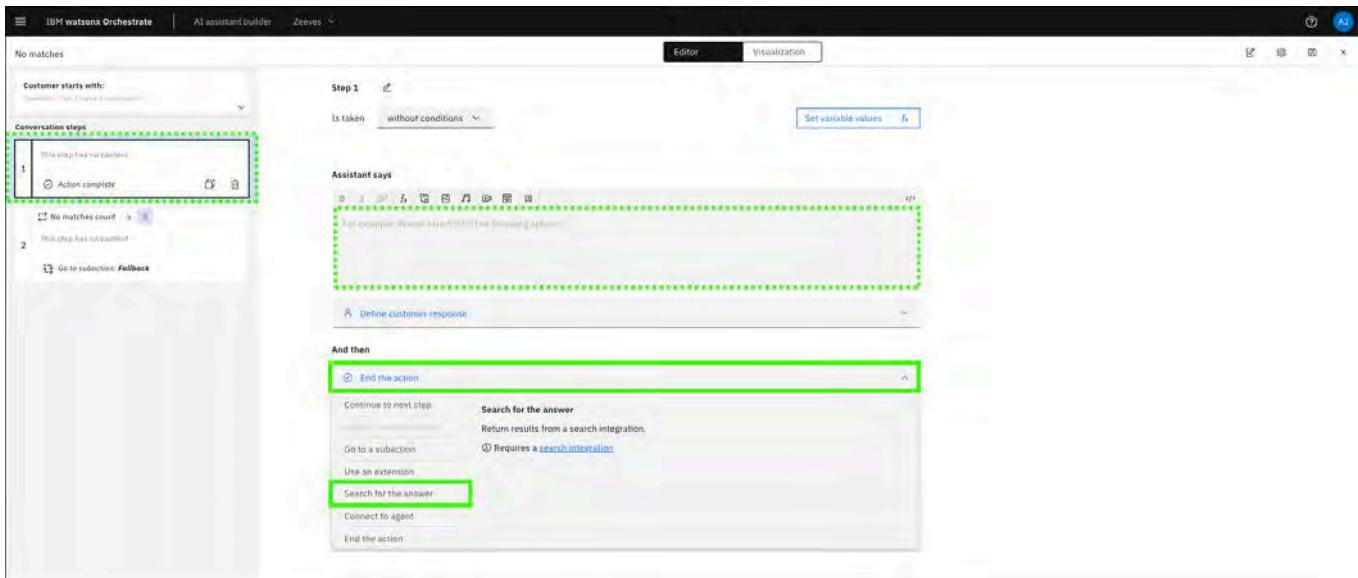
**Note:** the Is taken value does not change from with conditions after selecting without conditions.

The screenshot shows the 'Conversation steps' editor for the 'No matches' action. Step 1 is selected and highlighted with a green box. The 'Is taken' dropdown is set to 'without conditions' (highlighted with a green box). A confirmation dialog box titled 'Clear conditions?' is open in the foreground, asking if the user wants to proceed. The dialog has 'Cancel' and 'Clear conditions' buttons, with 'Clear conditions' highlighted with a red box.

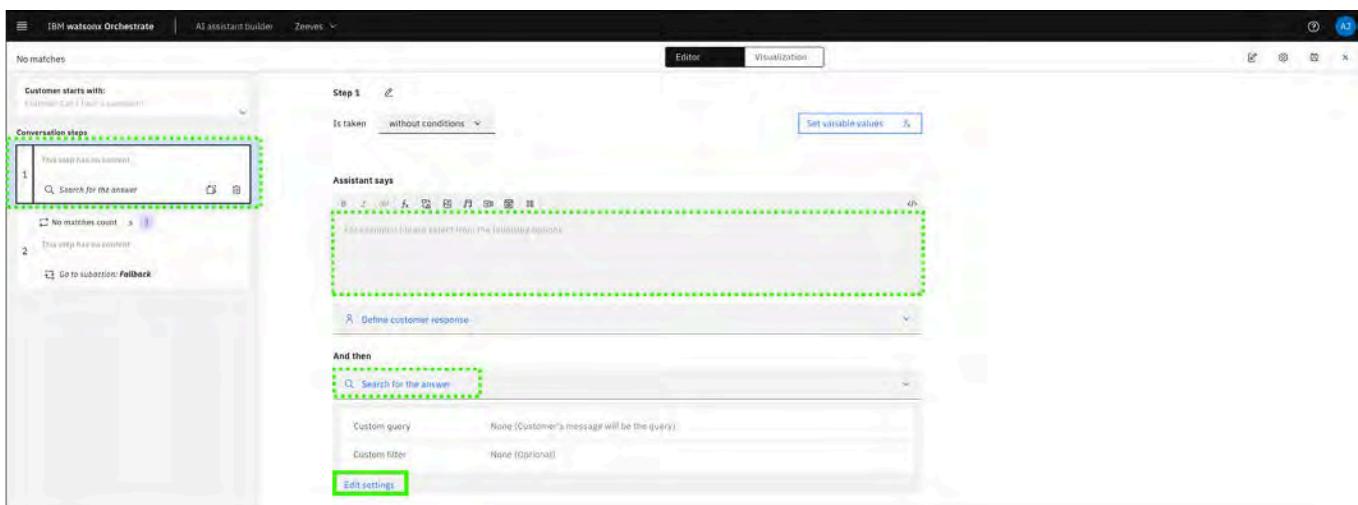
### 6. Delete the default text in the Assistant says entry field.

The screenshot shows the 'Conversation steps' editor for the 'No matches' action. Step 1 is selected and highlighted with a green box. The 'Assistant says' field contains the placeholder text 'Ask a question (1 word project name) or [Retrieving answer]' (highlighted with a green box). Below the 'Assistant says' field, the 'And then' section shows a condition 'Find the action'.

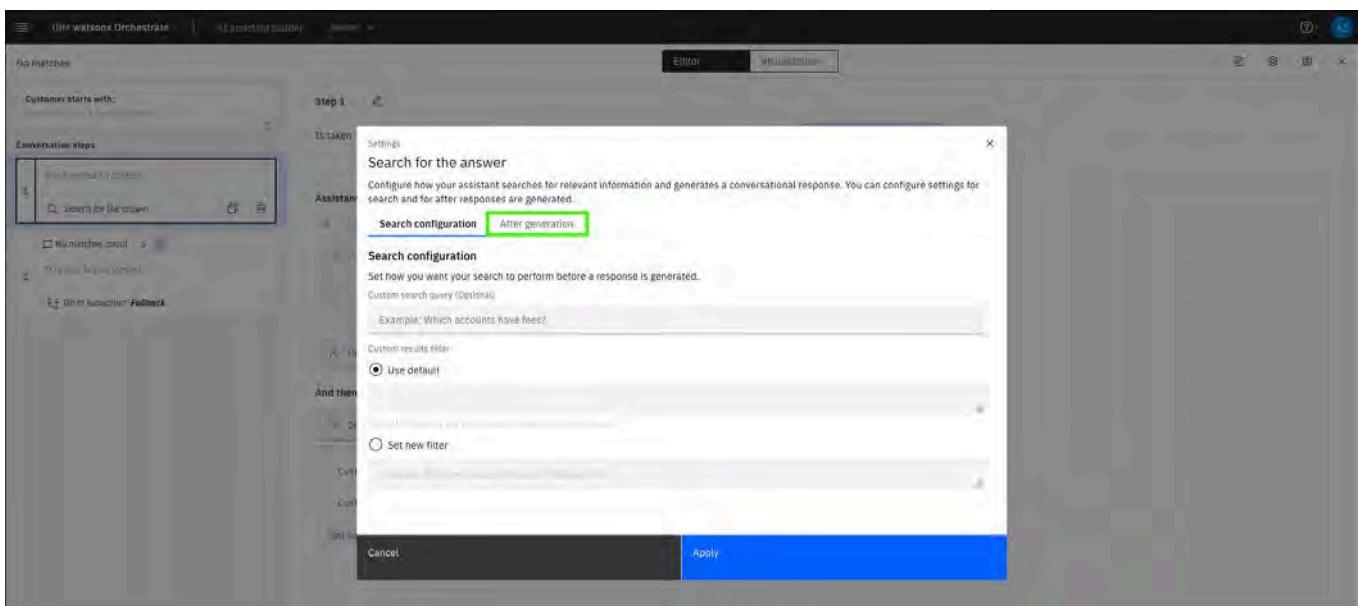
### 7. Expand the And then drop-down menu and select Search for the answer.



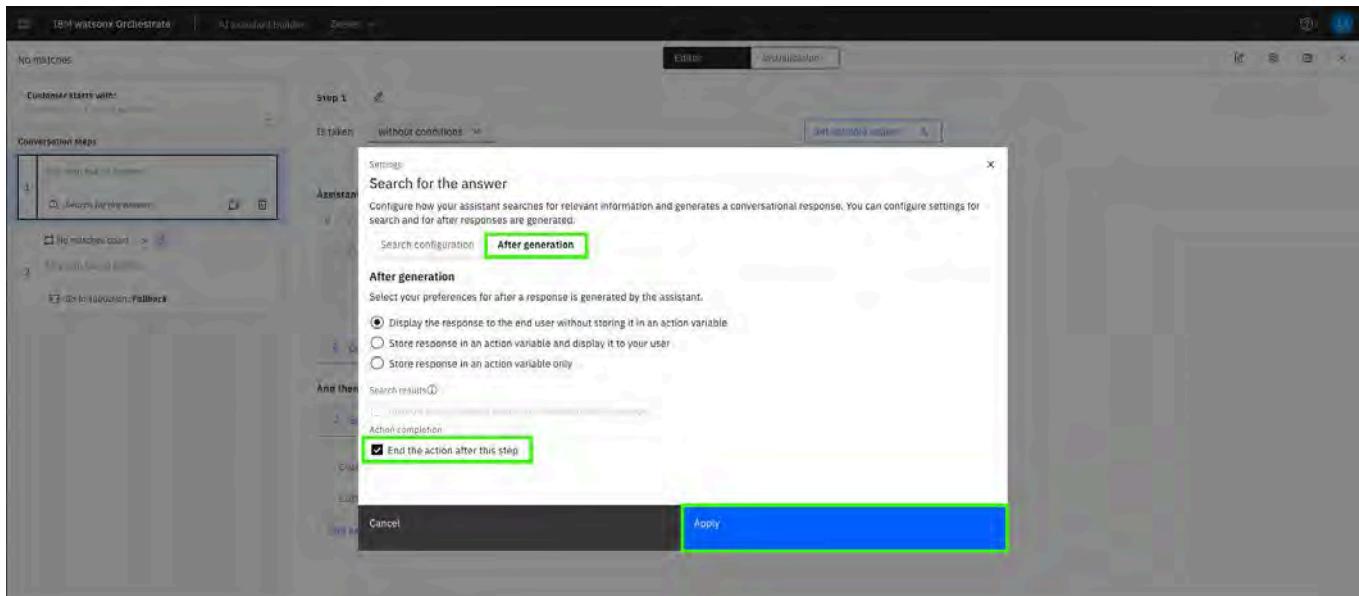
## 8. Click Edit settings.



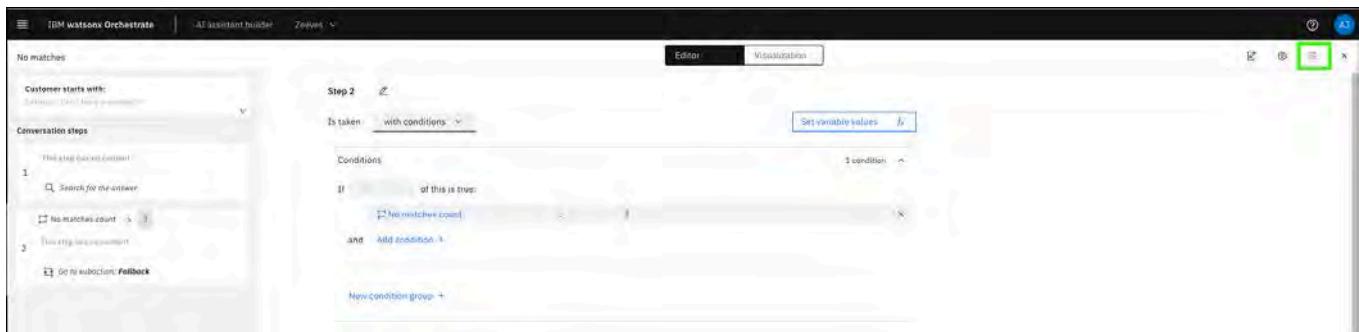
## 9. Click After generation.



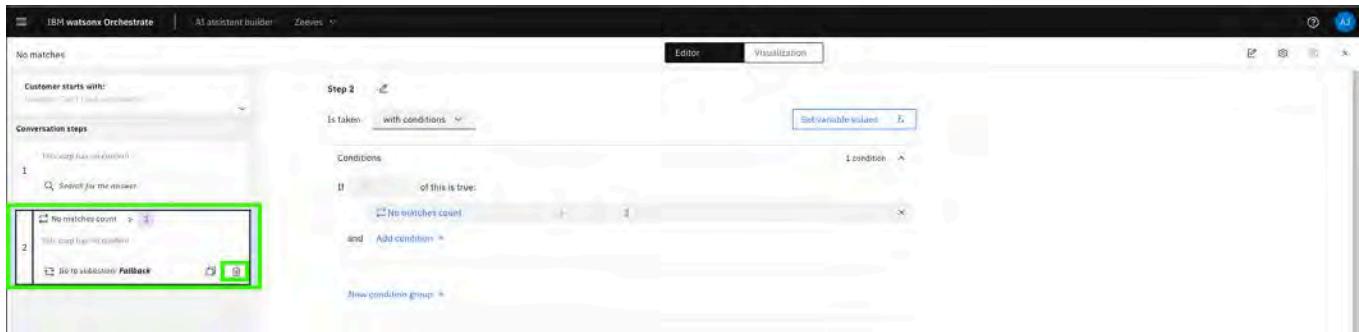
## 10. Select End the action after this step and then click Apply.



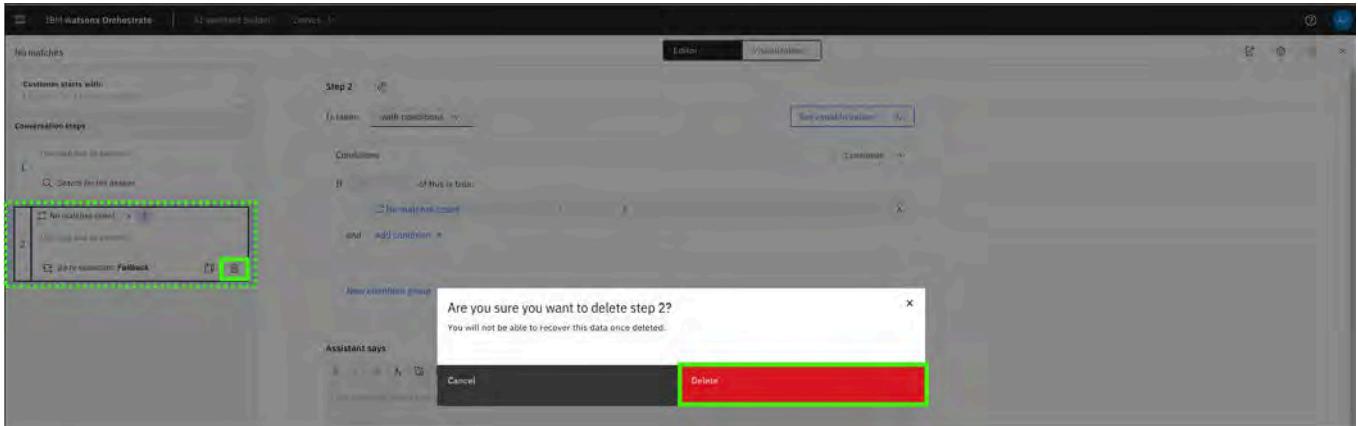
11. Click Save (💾).



12. Select Step 2 (No matches count) under Conversation steps and click delete (🗑).



13. Click Delete in the confirmation dialog to delete Conversation step 2.



14. Click **Close** (the x icon) the **Editor** window.



15. Click **Fallback** in the **Actions** table.



16. Delete all of the **Conversation steps**.

**Note:** The following image is edited. Only five steps are shown, but all six need to be deleted. You need to select each step individually. Click **delete** (trash icon) and confirm the deletion.

**Fallback**

**Conversation steps**

- 1 **Customer starts with:** `call agent`
- 2 **Conversation steps**
  - 1 **Fallback reason is Failed to confirm topic**: Sorry I couldn't confirm if you wanted to return to previous topic, let me connect to an agent.
  - 2 **Fallback reason is Input validation failed**: I'm afraid I don't understand. Can connect you to an agent.
  - 3 **Fallback reason is Agent requested**: Sorry I couldn't connect you. I will connect you to an agent right now.
  - 4 **Fallback reason is No matches**: I am afraid I do not understand what you are asking. Let me connect you to an agent.
  - 5 **Fallback reason is Danger word detected**: It seems this conversation would be best managed by a human agent. Let me connect you to one of our agents.

**Step 1**

Is taken with conditions

Conditions

If **of this is true:**

If **Fallback reason is Failed to confirm topic return**

and **Add condition** **New condition group**

**Assistant says**

Sorry I couldn't confirm if you wanted to return to previous topic, let me connect to an agent.

**Define customer response**

**And then**

**Connect to agent**

17. Verify that all **Conversation steps** are deleted and then click the **x** to close the **Editor** window.

**Fallback**

**Conversation steps**

**Action starts**

When your customer:

- Requests to connect to agent
- Fail step validation within an action
- Reaches the limit for No matches

Use the assistant's default action or customize it.

**Additional training examples for connecting to an agent**

Tip: Add examples here to help your assistant improve its customer responses.

Call agent

Connect to agent

18. Click the **Global settings** (⚙️).

Name	Last edited	Examples Count	Status
Generate customer	4 days ago	5	Green checkmark
Trigger visual detection	4 days ago	5	Green checkmark
No matches	4 days ago	5	Green checkmark
Fallback	4 days ago	5	Green checkmark

19. Click **No matches** under the **Conversation routing** tab.

**Global settings**

Conversation routing | Change conversation topic | Generative AI | Autocorrection | Display format | Algorithm version | Upload/Download | Close

**Ask clarifying questions**

Customize modes: Beta | No matches | Search routing

Your assistant can show options to users when multiple actions seem to match what the customer wants. Learn more about asking a clarifying question.

Enable disambiguation: On

**Assistant says:** *(Intention to) before listing the options.*

Did you mean:

- No matches: Don't for when the user doesn't see relevant options. Leave blank to omit.
- None of the above

**Response modes:** Off | One action matches

**Connection to support**

**Example of asking a clarifying question**

**Important:** Make sure your action names are short and clear to your customers. Action names will appear in the list of choices.

**Bank Bot**

Did you mean:

- Open a new savings account
- Open a new checking account
- I want to apply for a mortgage loan.
- None of the above

## 20. Move the slider to **More often** (or select **More often** in the drop-down).

The setting helps ensure that actions are triggered less often unless the user's query specifically matches the action's input.

**Global settings**

Conversation routing | Change conversation topic | Generative AI | Autocorrection | Display format | Algorithm version | Upload/Download | Close | Save

**Ask clarifying questions**

Customize modes: Beta | No matches | Search routing

Unrecognized input by customers triggers the **No matches** action that can be configured to fetch answers from a search integration or trigger the **Fallback** action.

By setting this threshold, you can affect how often your assistant routes customers to the "No matches" action.

Use "No matches":  More often

## 21. Click **Autocorrection**.

**Global settings**

Conversation routing | Change conversation topic | Generative AI | Autocorrection | Display format | Algorithm version | Upload/Download | Close | Save

**Ask clarifying questions**

Customize modes: Beta | No matches | Search routing

Unrecognized input by customers triggers the **No matches** action that can be configured to fetch answers from a **search integration** or trigger the **Fallback** action.

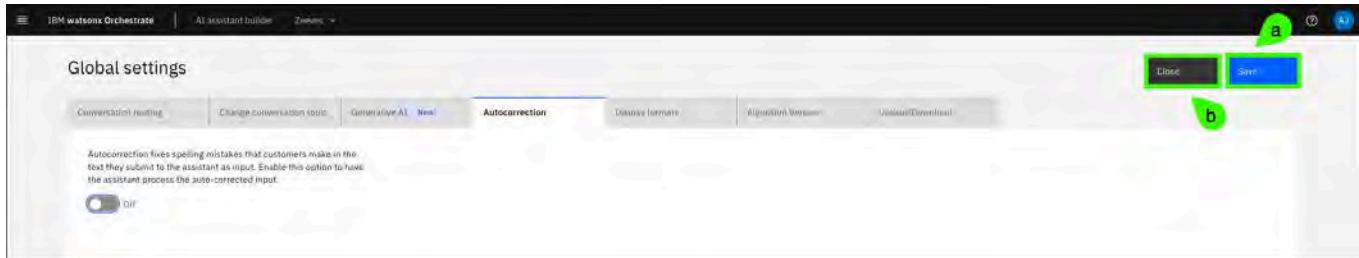
By setting this threshold, you can affect how often your assistant routes customers to the "No matches" action.

Use "No matches":  More often

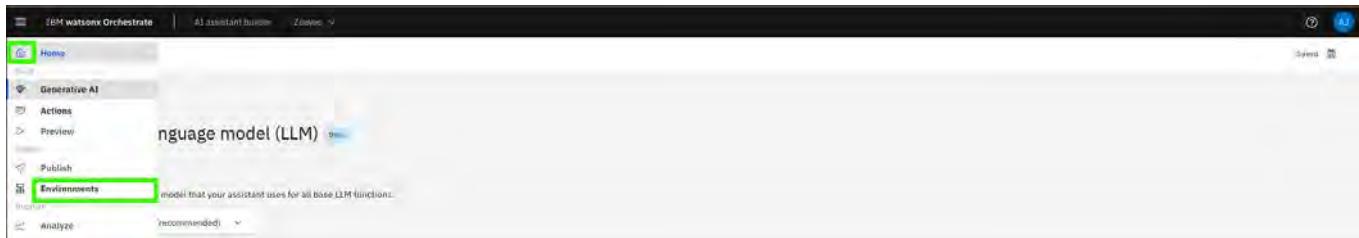
## 22. Click the autocorrection toggle to turn the feature **Off**.



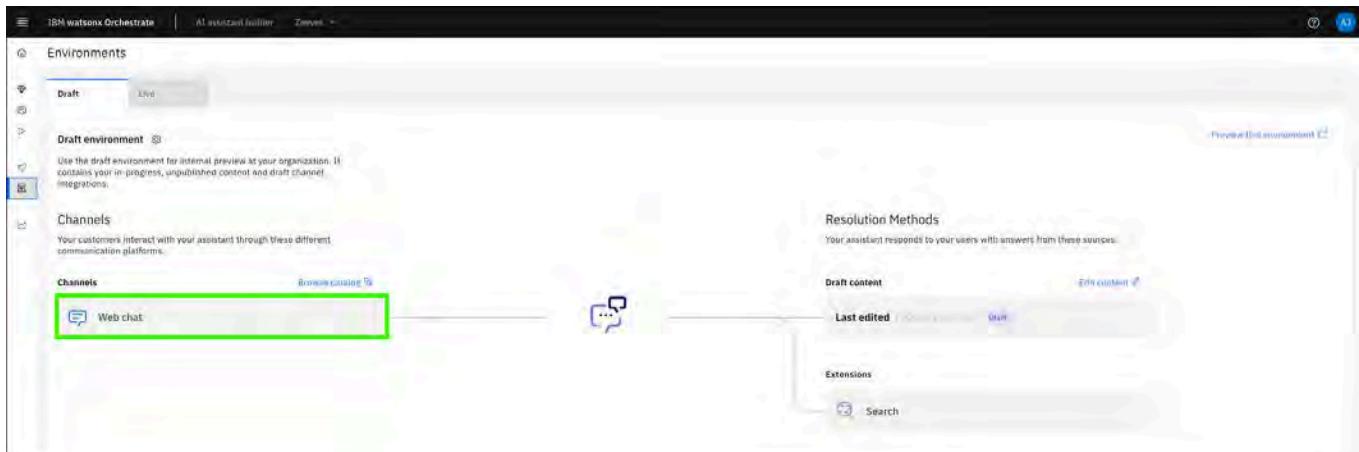
23. Click **Save** (a) and then **Close** (b).



24. Hover over the **Home** (🏠) and click **Environments**.

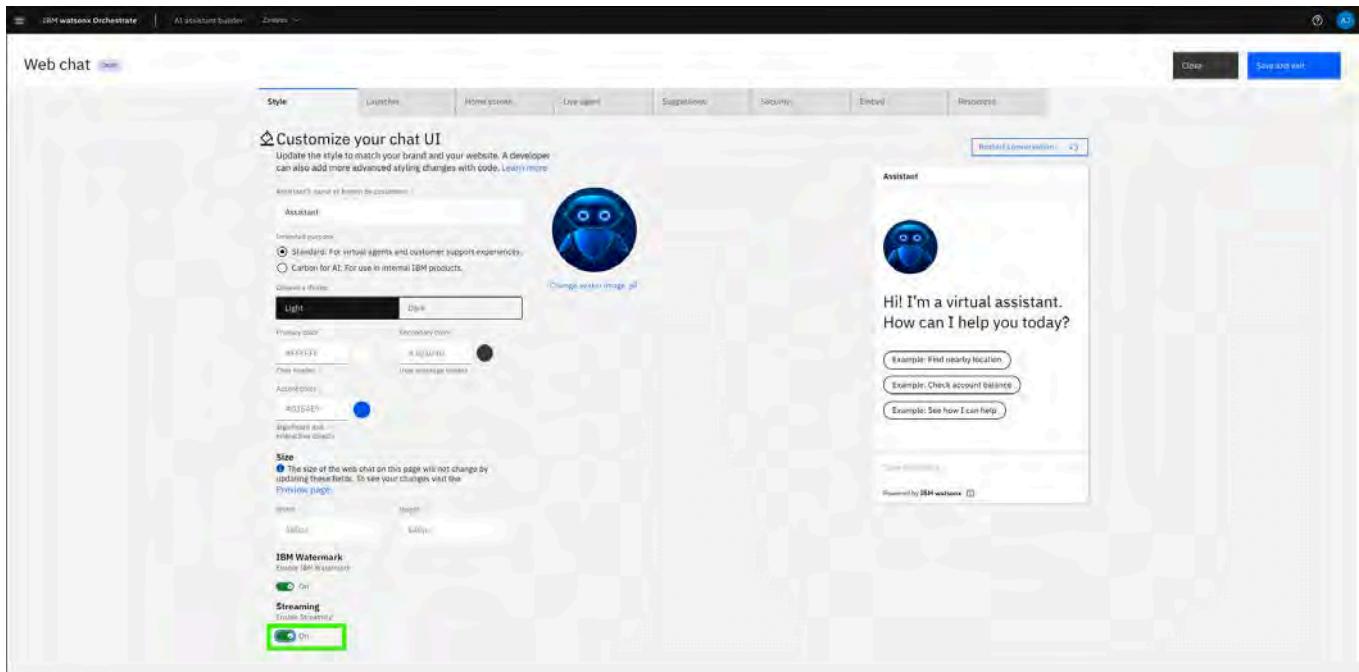


25. Click **Web chat**.

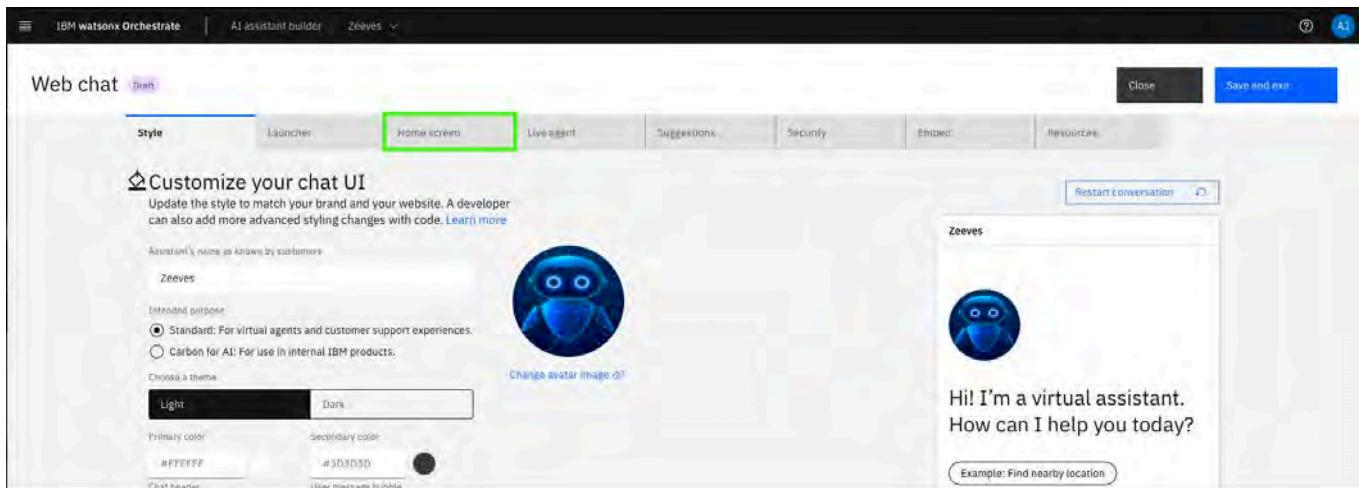


26. On the **Style** tab, click the **Streaming** toggle to enable streaming.

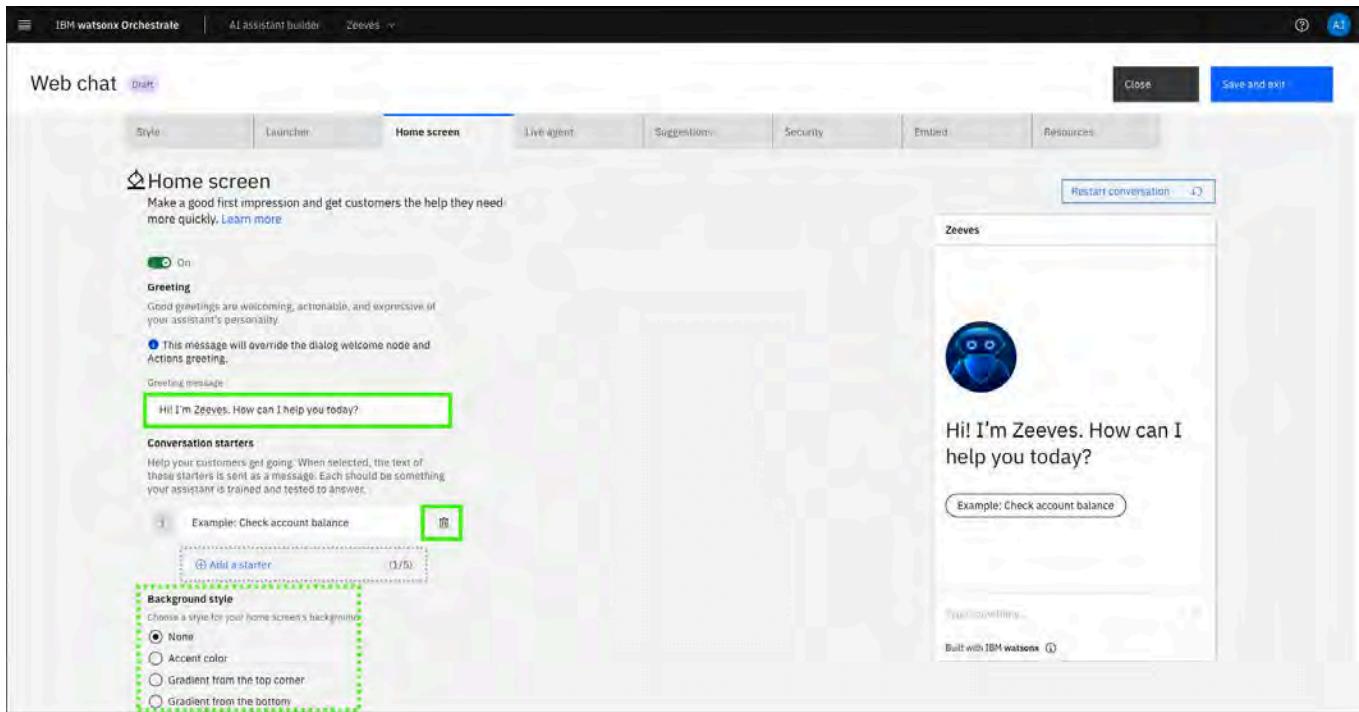
The streaming setting allows responses to be streamed to the assistant and displayed as they are generated versus waiting until the full response is received and then displayed.



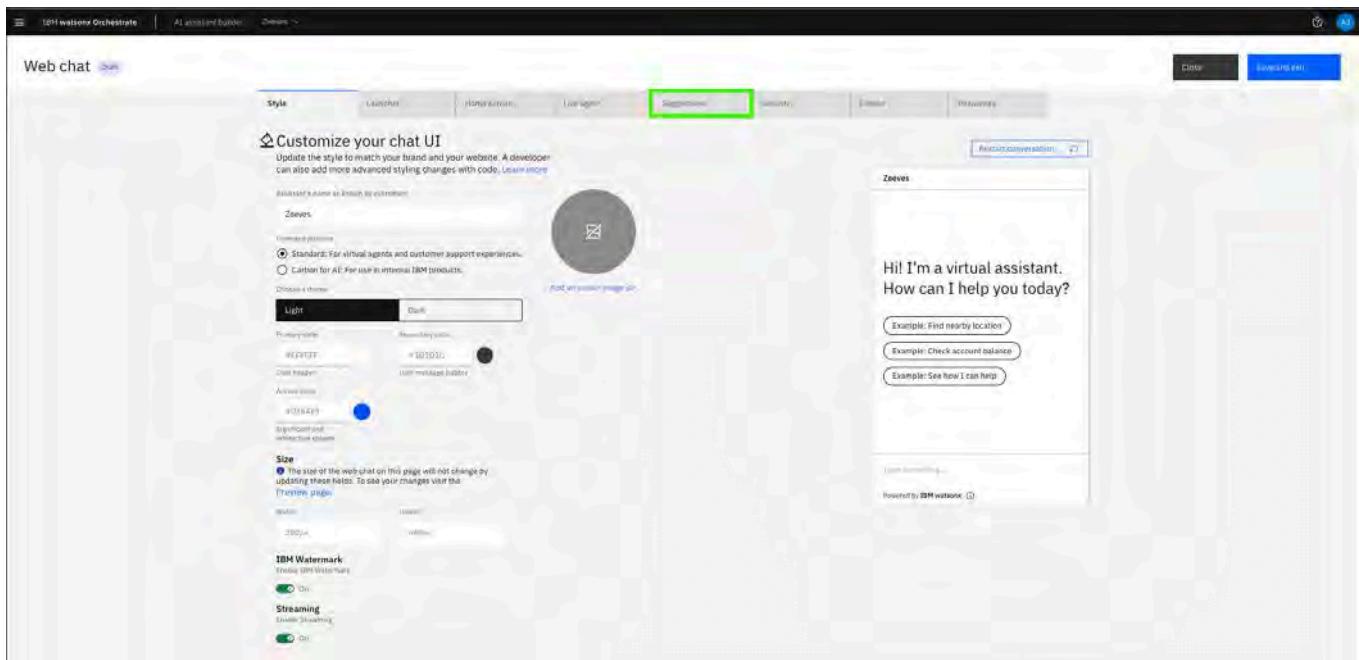
27. Click the **Home screen** tab.



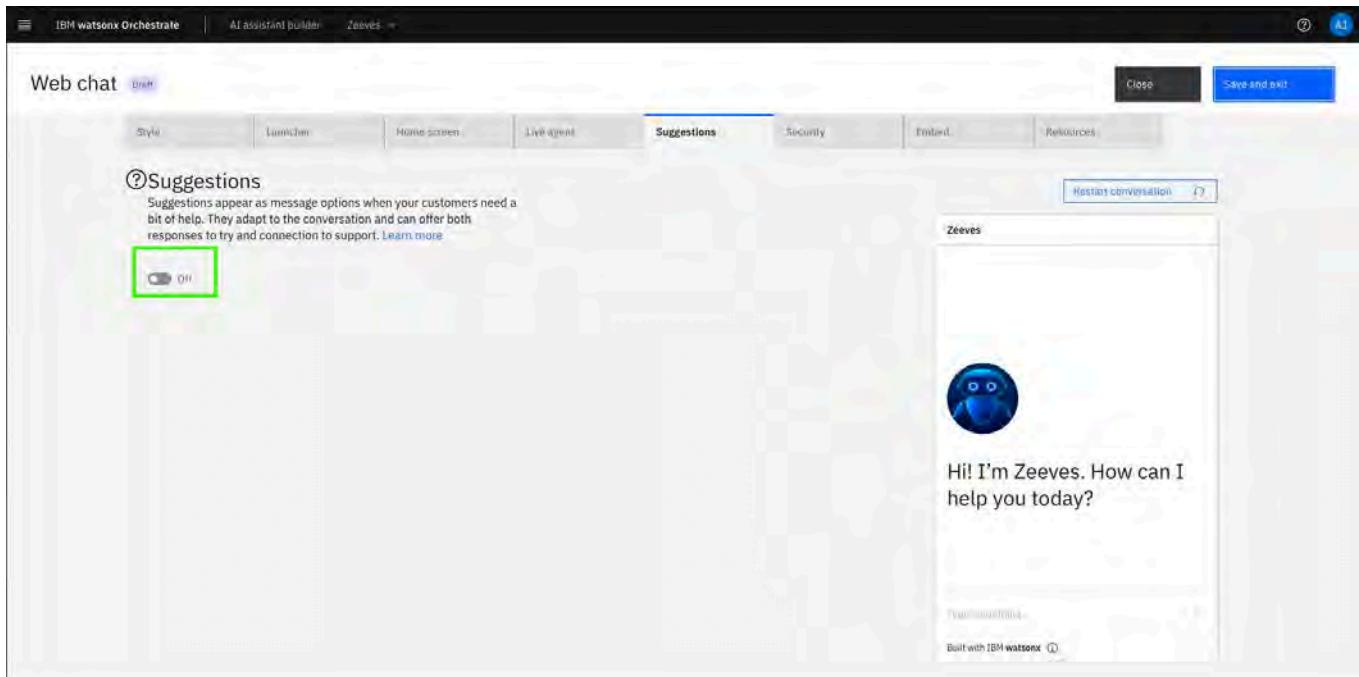
28. Customize the **Home screen** by setting a custom **Greeting message** and deleting the default **Conversation starters**. Optionally, adjust the **Background style**.



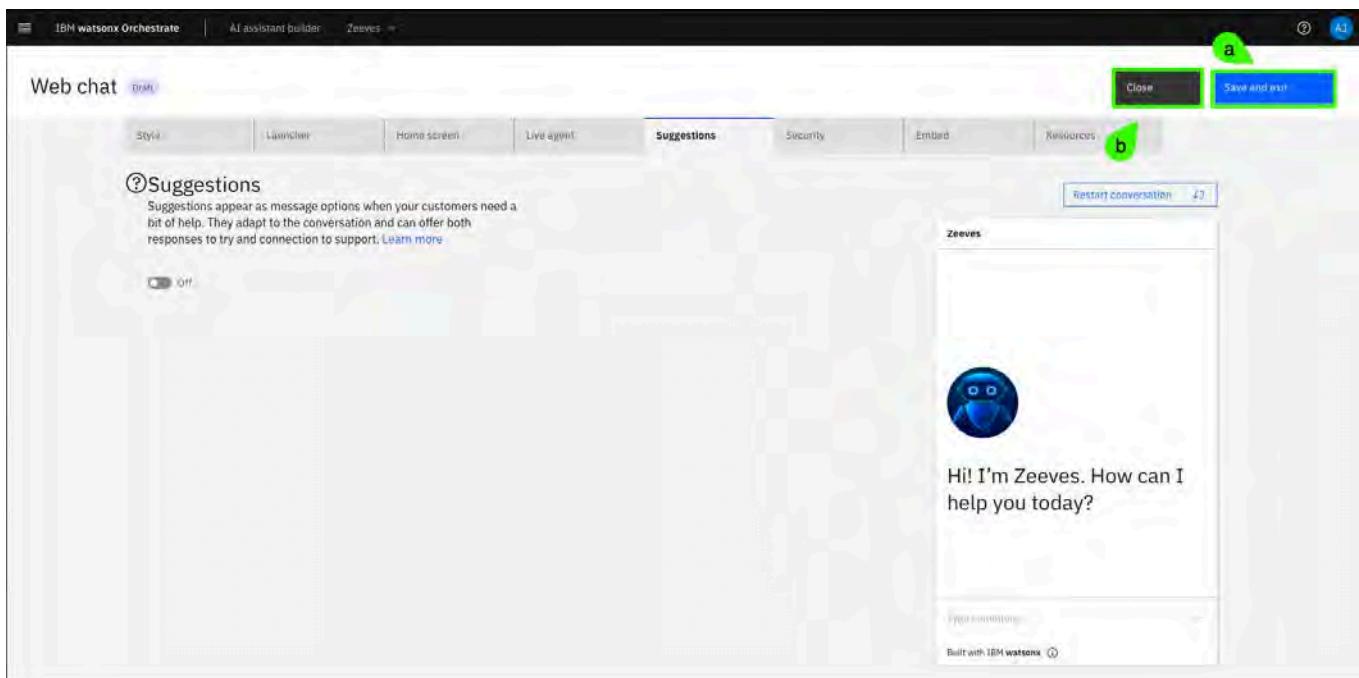
## 29. Click Suggestions.



## 30. Click the Suggestions toggle to turn this feature Off.



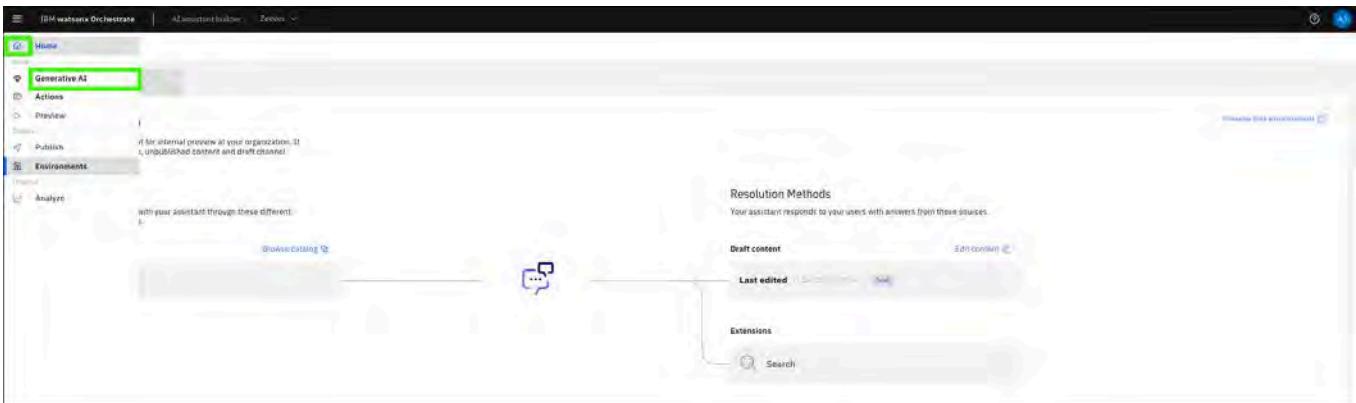
31. Click (a) **Save and exit** and then click (b) **Close**.



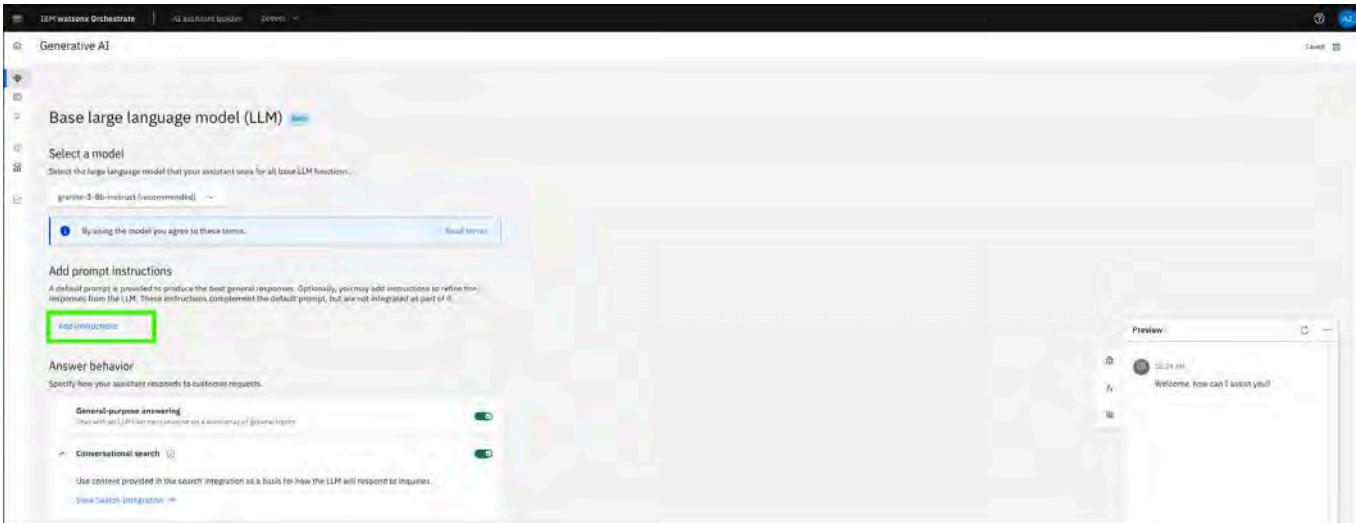
## Configure the base large language model

There are enhancements that you can make to configure how the large language model (LLM) responds to your queries, including adding prompt instructions and configuring the LLM's answer behavior. The options are summarized [here](#).

1. Hover over the **Home** ( ) and click **Generative AI**.



## 2. Click Add instructions.



## 3. Enter a prompt instruction.

Your assistant's LLM gives refined responses by following the prompt's instructions, which clarify how to achieve the end-goal of an action.

Enter prompt instructions in the field. The maximum number of characters you can enter in the prompt instruction field is 1,000.

The following is an example prompt instruction that works well. Experiment with different prompt instructions.

You are a subject matter expert on mainframe systems. Please respond to all prompts with truth and accuracy. Keep all answers short and concise, unless requested to provide details.

**Note:** When the instructions are typed in, they are automatically saved and the LLM is immediately trained on them.

The screenshot shows the 'Generative AI' configuration page. In the 'Answer behavior' section, the 'General-purpose answering' toggle switch is shown with a green box around it, indicating it is currently turned off. The 'Conversational search' toggle switch is also visible below it.

#### 4. Toggle **General-purpose answering** to **Off** and then click **Save** (💾).

The ability exists to configure the answering behavior of your assistant to provide responses that are based on the preinstalled content or general content.

On the **Generative AI** page (under **Prompt Instructions**), you see the **Answer behavior** section. After you configure **Conversational search**, you see that it is enabled (toggled on) with the search integration added.

If you enable both general-purpose answering and conversational search, conversational search answering takes precedence over General-purpose answering.

**Recommendation:** For purposes of retrieving Z-specific answers and responses, it is recommended that you turn off general-purpose answering and leave only conversational search turned on.

This screenshot is identical to the one above, showing the 'Generative AI' configuration page. The 'General-purpose answering' toggle switch is highlighted with a green box, indicating it is turned off. The 'Conversational search' toggle switch is also visible below it.

Testing conversational search

Now, you can begin issuing queries to test the assistant's responses. For more detailed responses, try appending "Please provide a detailed response." to the end of your question.

**Important:** Modify settings iteratively based on your assessment of response quality. Review and change them at any time. For example, add extra prompt instructions, change response verbosity, and modify OpenSearch indexes.

1. Hover over the **Home** ( ) and click **Preview**.

2. Experiment with different prompts and validate that the answers are reasonable and related to IBM Z.

Other prompts and responses follow.

**Note:** The responses that you receive can vary from the ones shown.

**Prompt:**

What is z/OS continuous delivery?

**Example output:**

The screenshot shows the AI Assistant Builder interface with a preview window. The prompt "What is z/OS continuous delivery?" is entered. The AI response is as follows:

6:29 AM  
What is z/OS continuous delivery?  
Z/OS continuous delivery is a model that allows IBM to deliver new functions and capabilities to its z/OS operating system on a regular basis, without waiting for the next major release. This model enables clients to benefit from new features and improvements quickly and with greater flexibility, as it is not centered around a release date. New functions are delivered through APARs and can be selectively rolled out to specific CICS regions or across CICSplexes during the system software migration.

**Prompt:**

What is the APF list in z/OS?

**Example output:**

The screenshot shows the IBM Watson Assistant interface. On the left, there's a sidebar with icons for preview, AI assistant builder, and Zeeves. The main area is titled "Sample website". A small Zeeves window is open in the bottom right corner, displaying a message about the APP list in z/OS.

**Zeeves**

What is the APP list in z/OS? Provider: Watson Assistant

11:02 AM

- The APP (Authorized Program Facility) list in z/OS is a system component that controls which libraries and data sets are authorized to run in privileged mode. This is used to ensure system security and integrity by restricting access to sensitive system resources.
- The APP list contains a list of libraries and data sets that are authorized to run in privileged mode. These libraries and data sets include system programs, applications, and utilities that require elevated privileges to function correctly. By authorizing these commands and macros.

Powered by IBM Watson

**Prompt:**

Why is Db2 different than other database systems?

**Example output:**

The screenshot shows the IBM Watson Assistant interface. On the left, there's a sidebar with icons for preview, AI assistant builder, and Zeeves. The main area is titled "Sample website". A small Zeeves window is open in the bottom right corner, displaying a message about why Db2 is different from other database systems.

**Zeeves**

Why is Db2 different than other database systems? Provider: Watson Assistant

11:03 AM

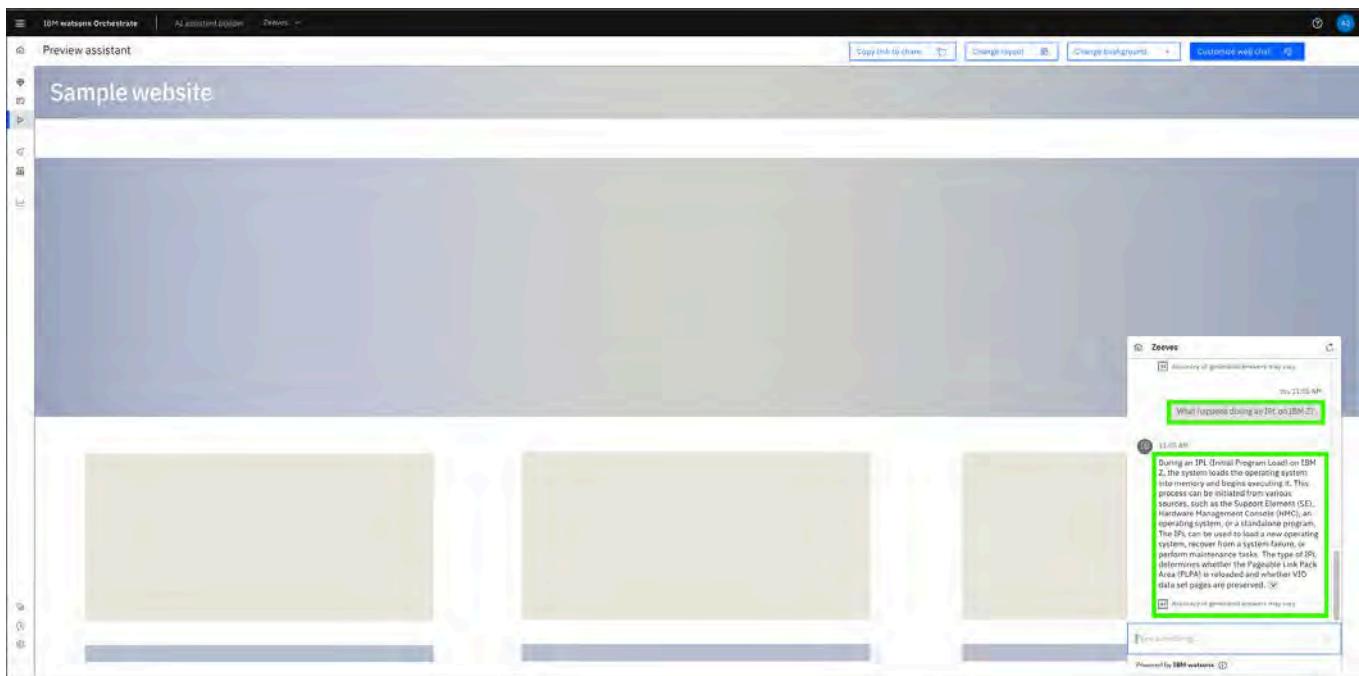
- Db2 is a relational database management system (DBMS) that runs under the z/OS operating system, which is specific to mainframe computers. Other database systems may run on various operating systems like Linux, Windows, or Unix.
- Additionally, Db2 provides a higher level of data management and security features compared to other database systems, as it manages data storage, databases, and controls access to the data.

Powered by IBM Watson

**Prompt:**

What happens during an IPL on IBM Z?

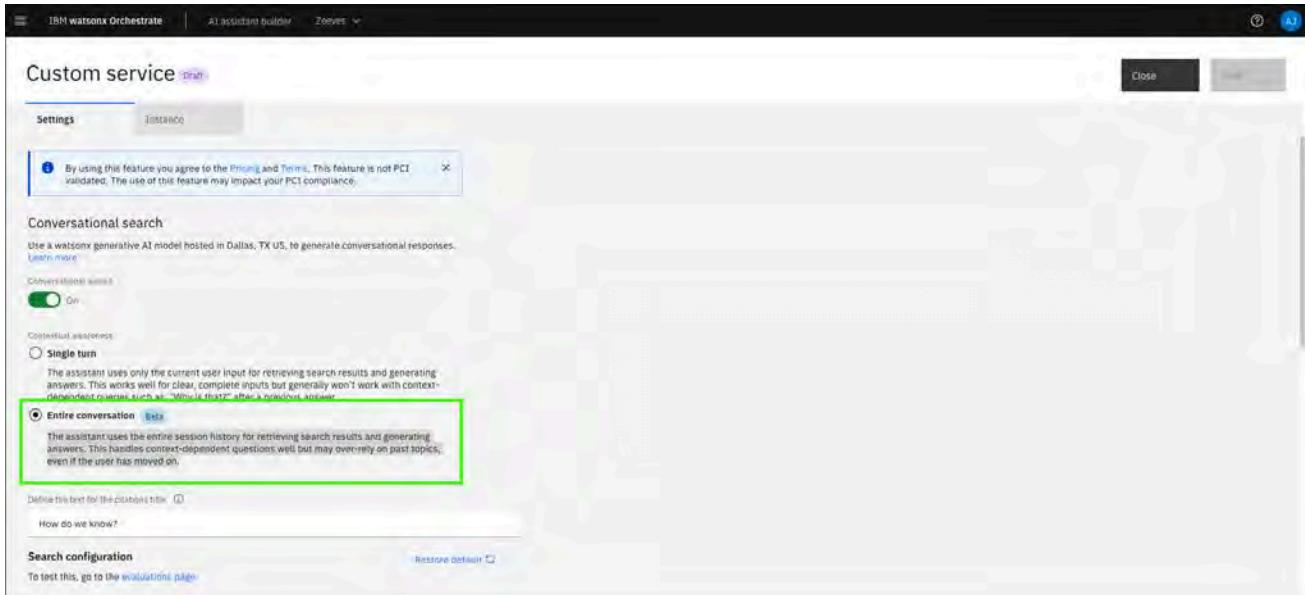
**Example output:**



## Experiment with multi-turn (entire conversation) contextual awareness.

In the December 2024 release of IBM Watsonx Assistant for Z support for multi-turn contextual awareness was added. This capability enables the assistant to use an entire session history for retrieving search results and generating answers. This handles context-dependent questions well but may over-rely on past topics, even if the user has moved on.

Experiment with this setting by changing your custom service contextual awareness setting from **Single turn** to **Entire conversation**.



The screenshot shows the 'Custom service' configuration page in the IBM Watsonx Orchestrate AI Assistant Builder. In the 'Contextual awareness' section, the 'Entire conversation' option is selected, highlighted with a green border. A tooltip for this option states: 'The assistant uses the entire session history for retrieving search results and generating answers. This handles context-dependent questions well but may over-rely on past topics, even if the user has moved on.' Other options shown are 'Single turn' and 'Conversation history'.

Once enabled, try sequential prompts like:

What are some features of z/OS?

Give me an itemized list?

Tell me more about item 3.

You have a working assistant that uses IBM Watson Assistant for Z. Explore different prompt instructions and settings. If you encounter issues, refer to the Troubleshooting section that follows for resolution.

Continue to the [Creating a stand-alone OpenSearch instance for document ingestion](#) to learn how to configure a dedicated OpenSearch instance for ingesting client-specific documentation into the RAG model.

## Troubleshooting

The following are issues that you may encounter. If the provided resolutions do not work, contact support by using the methods that are mentioned in the [Support](#) section.

 **Assistant responds to all prompts with, "I might have information related to your query to share, but am unable to connect to my knowledge base at the moment"** 

This Assistant is unable to connect to the custom service URL specified. This could be a network issue, the service may be down, the service may be restarting, or the service is no longer running at that URL.

Before reaching out to [Support](#), try the following:

- Wait a few minutes and try again. It may be the service was in the process of restarting.
- If you printed this demonstration guide or saved a copy, verify you are using the most current version of the [lab guide](#) and the correct service URL (<https://wxa4z-opensearch-wrapper-wxa4z-demo-v2-1-0.wxo4z-opc-opensearch-clus-47e063e6a3ad1f71bf2e58f91c3b4c2e-0000.us-south.containers.appdomain.cloud/v1/query>). The URL may have changed since you saved or printed the lab guide.

Bring your own search and documents

# Creating a stand-alone OpenSearch instance for document ingestion

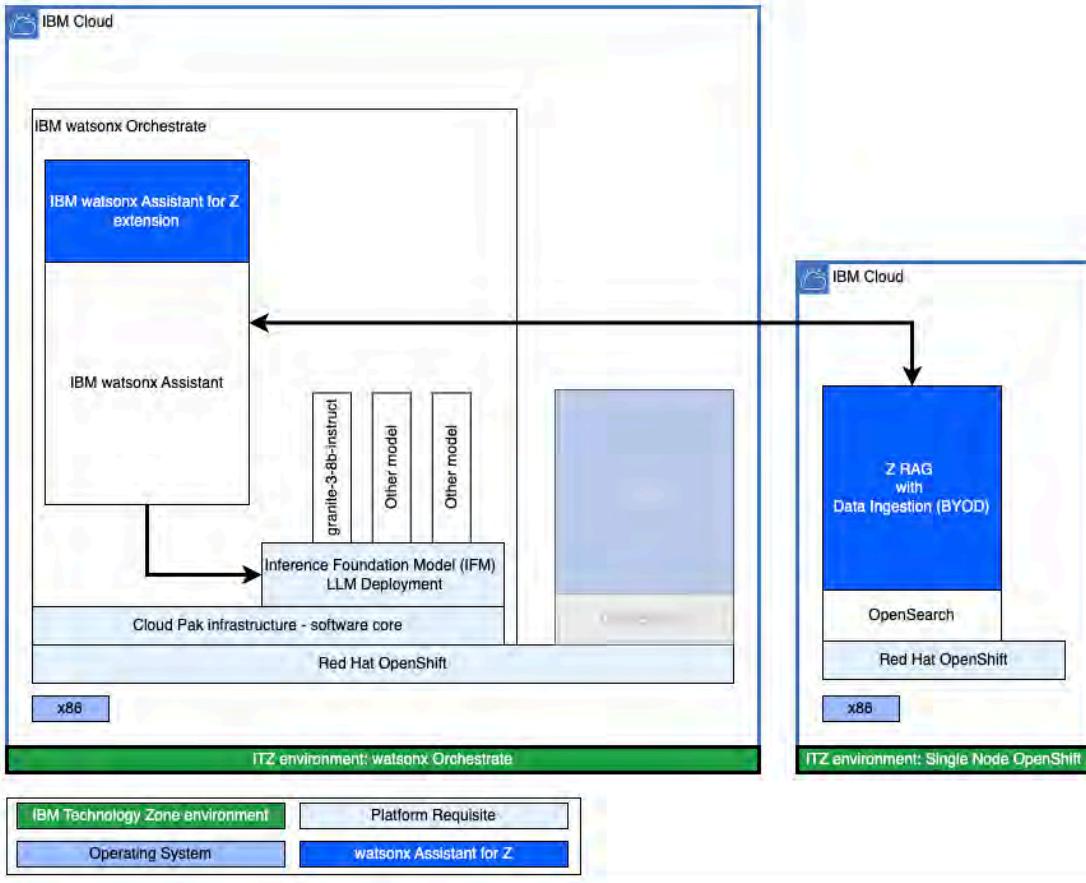
Now that you created and deployed your own assistant with conversational search capabilities, your client can understand how watsonx Assistant for Z provides its content-grounded responses to any Z-related questions. In the previous section, you configured your assistant to use a pre-configured Z RAG that has over 220 knowledge sources, and uses this knowledge to provide AI-generated responses.

Next, learn to enable clients to personalize the assistant with an internal knowledge base that contains documentation they add to the Retrieval Augmented Generation (RAG). This helps provide a level of context-awareness for their own environment when environment-specific questions are asked to the assistant.

Now, install and configure a “Z RAG” on Red Hat OpenShift enabling the bring-your-own-search (BYOS) and bring-your-own-documentation (BYOD) capability to ingest other documentation. In doing so, you deploy a dedicated OpenSearch instance (BYOS). Then, connect your assistant to the new RAG database to provide responses based on the ingested documentation (BYOD).

Below is a high-level, logical architecture of the environment deployed in this section.

## Creating a stand-alone OpenSearch instance for document ingestion



Earlier, you provisioned three IBM Technology Zone (ITZ) environments. One of which was a single-node Red Hat OpenShift (SNO) cluster. If you have not reserved this environment, or it is not in the **Ready** state, return to the [IBM Technology Zone environment](#) section to complete the reservation.

## Install the Red Hat OpenShift command line interface utility

The Red Hat OpenShift command line interface (CLI) utility, which is known as **oc**, must be installed on your local workstation. If you already installed the **oc** utility, you can proceed to [log in to the SNO cluster](#).

1. Click the following link to open a browser window to your ITZ reservations.

[ITZ My reservations](#)

2. Click the **Single Node OpenShift** tile.

### 3. Scroll down and record the Cluster Admin Username and Cluster Admin Password.

**Reservation Details**

API URL: <https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43189>

Cluster Admin Username: `kubecadmin`

Cluster Admin Password: `clusterAdmin@123`

### 4. Click the OCP Console link.

**Note:** OCP stands for OpenShift Container Platform.

Reservation Details

API URL: <http://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43387>

Cluster Password:

Cluster RDP Address: `api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43387`

Cluster SSH Connection: `ssh itzru@api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com -p 40222`

Cluster Username: `itzru`

Cluster Admin Username: `kubeadmin`

Cluster Admin Password:

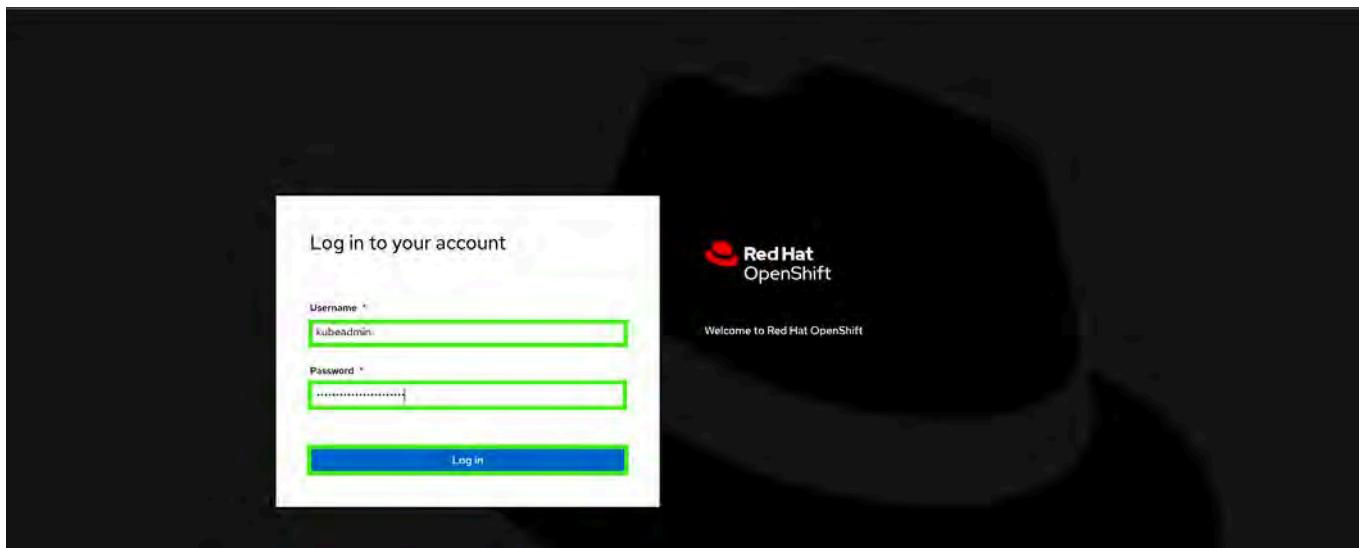
OCP Cluster: `http://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43387`

OCP Version: `4.14`

Region: `itzru-vc.eu.cloud.techzone.ibm.com`

[Download kubeconfig](#)

- Enter the **Cluster Admin Username** and **Cluster Admin Password** values from step 3 and click **Log in**.



- Click Help (?) and then click **Command Line Tools**.

- Click the link under **oc - OpenShift Command Line Interface (CLI)** for the operating system of your local machine.

The screenshot shows the Red Hat OpenShift on IBM Techzone interface. On the left, there's a navigation sidebar with options like Home, Operators, Workloads, Networking, Storage, Builds, Observe, Compute, and User Management. Under Compute, the 'helm - Helm 3 CLI' section is visible. The main content area is titled 'Command Line Tools' and contains a 'Copy login command' button. Below it is a section for the 'oc' command, which is described as the OpenShift command line interface. It offers the same capabilities as the 'kubectl' binary but supports OpenShift Container Platform features. There are download links for 'oc' binaries for Linux (x86\_64), Mac, Windows, ARM, IBM Power, and IBM Z, along with a 'LICENSE' link. A green dashed box highlights this section.

Clicking the preceding link automatically downloads either a **.zip** or **.tar** file specific to your operating system. Extract the file's content. Place the **oc** binary for your operating system (OS) in a directory that is in your default PATH, or set the PATH environment variable to include the location of the **oc** binary.

#### 8. Verify the installation by running the **oc** command on your local workstation.

```
oc --help
```

Sample output:

```
andrewjones@Andrews-MBP ~ % oc --help
OpenShift Client

This client helps you develop, build, deploy, and run your applications on any
OpenShift or Kubernetes cluster. It also includes the administrative
commands for managing a cluster under the 'adm' subcommand.

Basic Commands:
  login           Log in to a server
  new-project     Request a new project
  new-app          Create a new application
  status           Show an overview of the current project
  project          Switch to another project
  projects         Display existing projects
  explain          Get documentation for a resource

Build and Deploy Commands:
  rollout          Manage a Kubernetes deployment or OpenShift deployment
  config           Revert part of an application back to a previous deployment
```



**Mac/OS users may need to adjust security settings.**



The **oc** binary may cause a security exception. Adjust the security settings by opening the **System Settings** utility and clicking **Privacy & Security**. Under **Security** locate the message about the **oc** binary and click **Allow Anyway**. Return to the terminal window and try the **oc --help** command again and click **Allow Anyway** when prompted.

## Prepare to ingest documents

Before ingesting documents, complete the following setup steps.

## Log in to the OpenShift cluster from your local terminal

**Note:** If you just installed the **oc** utility, skip the next 5 steps.

1. Click the following link to open a browser window to your ITZ reservations.

[ITZ My reservations](#)

2. Click the **Single Node OpenShift** tile.

The screenshot shows the 'My reservations' section of the ITZ interface. It displays three reservation cards. The first card, titled 'Single Node OpenShift (VMware on IBM Cloud)', has its status set to 'Ready'. The second card is for 'watsonx Assistant for Z Pilot - AAP & z/OS' and the third for 'watsonx Assistant for Z Pilot - watsonx O...'. Each card includes details like start and end dates, end times, and an 'Open this environment' button. The first card is highlighted with a green border.

3. Scroll to the bottom of the reservation page and record the **Cluster Admin Username** and **Cluster Admin Password**.

The screenshot shows the 'Reservation Details' page for the 'Single Node OpenShift' reservation. It lists two nodes: '672371d38376796fb96ac4d-master-1' and '672371d38376796fb96ac4d-min'. Below this, the 'Reservation Details' section is expanded, showing the 'Cluster Admin Username' field which contains 'kubadmin' and the 'Cluster Admin Password' field which contains 'ituser'. Both fields are highlighted with a green border.

4. Click the **OCP Console** link.

Reservation Details

API URL:  
<http://ocpcluster-openshift-672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43387>

Cluster Password:

Cluster RDP Address:  
api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43387

Cluster SSH Connection:  
ssh itzru@api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com -p 40222

Cluster Username:  
itzru

Cluster Admin Username:  
kubeadmin

Cluster Admin Password:

OCP Cluster:  
<http://ocpcluster-openshift-672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43387>

OCP Version:  
4.14

Region:  
itzru-vc.eu.cloud.techzone.ibm.com

Download kubeconfig

- Enter the **Cluster Admin Username** and **Cluster Admin Password** values from step 3 and click **Log in**.

Welcome to Red Hat OpenShift

**Log in to your account**

Username \*: kubeadmin

Password \*: [REDACTED]

Log in

- Click the **kube:admin** profile drop-down and click **Copy login command**.

You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.

**Command Line Tools**

**Copy login command**

**oc - OpenShift Command Line Interface (CLI)**

With the OpenShift command line interface, you can create applications and manage OpenShift projects from a terminal.

The oc binary offers the same capabilities as the kubectl binary, but it is further extended to natively support OpenShift Container Platform features.

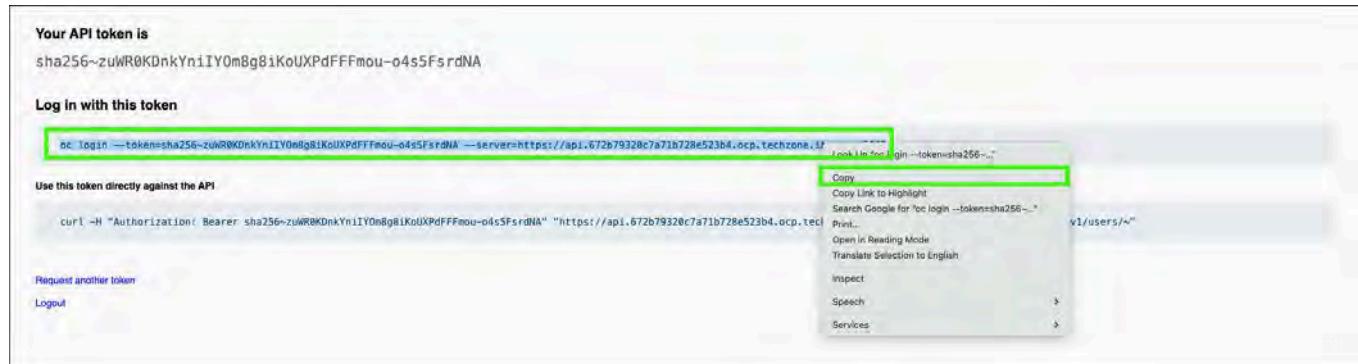
- Download oc for Linux for x86\_64

- Click **Display Token**.

**Display Token**

- Select and copy the **Log in with this token** string.

For most operating systems, double-click the value, then right-click and select **Copy**.



9. Open a command prompt or terminal window on your local workstation.

10. Paste the login command and press **enter**.

```
andrewjones@Andrews-MBP ~ % oc login --token=sha256~mJ4L8K6cUMyNykZ2Z69KMm3vbP1sWc8Sw0eeOdVqtA94 --server=https://api.672371d38376796fb9~6a6c4d.ocp.techzone.ibm.com:6443
Logged into "https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443" as "kube:admin" using the token provided.
You have access to 70 projects, the list has been suppressed. You can list all projects with 'oc projects'
Using project "default".
andrewjones@Andrews-MBP ~ %
```

## Create a working directory

1. Create a directory to store the configuration files that you will create in the next steps.



**Instructions vary by your local workstation's operating system.**

The directions that follow may vary depending on your operating system. The examples provided are based upon MacOS.

```
mkdir watsonxAssistant
```

2. Change to the new directory.

```
cd watsonxAssistant
```

```
andrewjones@Andrews-MBP ~ % oc login --token=sha256~mJ4L8K6cUMyNykZ2Z69KMm3vbP1sWc8Sw0eeOdVqtA94 --server=https://api.672371d38376796fb9~6a6c4d.ocp.techzone.ibm.com:6443
Logged into "https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443" as "kube:admin" using the token provided.
You have access to 70 projects, the list has been suppressed. You can list all projects with 'oc projects'
Using project "default".
andrewjones@Andrews-MBP ~ % mkdir watsonxAssistant
andrewjones@Andrews-MBP ~ % cd watsonxAssistant
andrewjones@Andrews-MBP watsonxAssistant %
```

## Install IBM Certificate Manager on Red Hat OpenShift

1. In a text editor, create a file that is named `catalogCertManager.yaml` and paste the following text in the file.



### Formatting of the yaml file is critical!

The content of the YAML file must be formatted exactly as shown. Use the **Copy** icon to prevent typographical errors.

File name:

```
catalogCertManager.yaml
```

File contents:

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-cert-manager-catalog
  namespace: openshift-marketplace
spec:
  displayName: ibm-cert-manager-4.2.7
  grpcPodConfig:
    securityContextConfig: restricted
    image: icr.io/cpopen/ibm-cert-manager-operator-
catalog@sha256:4dcf4ace4b5f166f83b31063f7e6404dbf78d8e98a9d4fcf52fedf576a55ca6c
  publisher: IBM
  sourceType: grpc
  updateStrategy:
    registryPoll:
      interval: 30m0s
```

2. Install the IBM Certificate Manager operator in the Red Hat OpenShift cluster.

```
oc apply -f catalogCertManager.yaml
```

The preceding command returns a message that states the **ibm-cert-manager-catalog** was created.

3. In the OpenShift web console, click **Operators** and then select **OperatorHub**.

The screenshot shows the Red Hat OpenShift web console interface. The top navigation bar includes the Red Hat logo and the text "OpenShift on IBM Techzone". The left sidebar has a "Administrator" dropdown, a "Home" link, and a "Operators" section with a green highlight. Under "Operators", there is a "OperatorHub" link which is also highlighted with a green box. Other links in the sidebar include "Workloads" and "Networking". The main content area has a title "Command Line Tools" and a sub-section "oc - OpenShift Command Line Interface (CLI)". It provides instructions for using the CLI and links for downloading it for various platforms: "Download oc for Linux for x86\_64", "Download oc for Mac for x86\_64", "Download oc for Windows for x86\_64", and "Download oc for Linux for ARM 64".

4. Click the **Project** to pull-down menu and click the **Show default projects** toggle.

The screenshot shows the Red Hat OpenShift OperatorHub interface. On the left, a sidebar menu includes 'Administrator', 'Home', 'Operators', 'OperatorHub' (which is selected and highlighted in blue), 'Workloads', 'Networking', and 'User Management'. The main area has a header 'Project: All Projects' with a dropdown arrow. Below it is a search bar with placeholder text 'Select project...'. A button labeled 'Show default projects' with a switch icon is visible. The main content area lists 'Projects' under 'All Projects': 'default', 'kube-node-lease', and 'kube-public'. At the bottom right, there are tabs for 'Community', 'Marketplace', and 'Community' again. A note at the top right says 'You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.' A status bar at the bottom right shows 'kubeadmin'.

## 5. Scroll down and select openshift-marketplace.

This screenshot shows the 'Installed Operators' section of the OperatorHub. The sidebar remains the same. The main area lists several operators: 'openshift-kube-controller-manager-operator', 'openshift-kube-scheduler', 'openshift-kube-scheduler-operator', 'openshift-kube-storage-version-migrator', 'openshift-kube-storage-version-migrator-operator', 'openshift-machine-api', 'openshift-machine-config-operator', 'openshift-marketplace' (which is highlighted with a green box), 'openshift-monitoring', 'openshift-multus', 'openshift-network-diagnostics', 'openshift-network-node-identity', 'openshift-network-operator', 'openshift-node', 'openshift-nutanix-infra', and 'openshift-oauth-apiserver'. Below the list is a 'Create Project' button. The status bar at the bottom right shows 'kubeadmin'.

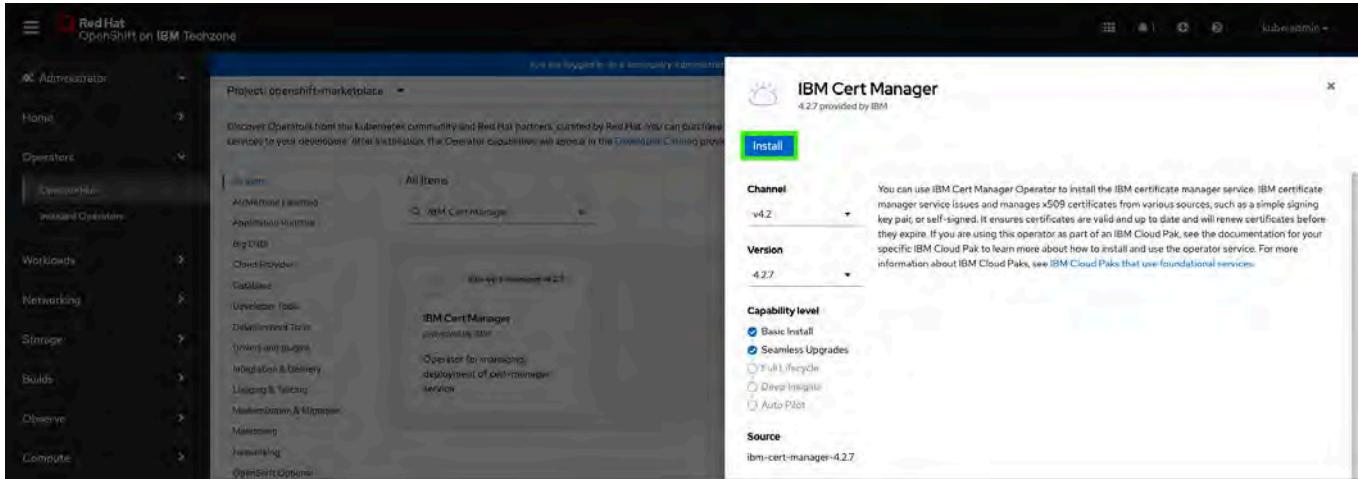
## 6. Enter IBM Cert Manager in the search field and then click the IBM Cert Manager tile.

A message box with a light blue background and a white border. It contains a green icon of a person with a speech bubble and the text 'Be patient.' followed by a period. Below this, in a white box, is the text 'It may take a minute or two for the IBM Cert Manager tile to appear.' The status bar at the bottom right shows 'kubeadmin'.

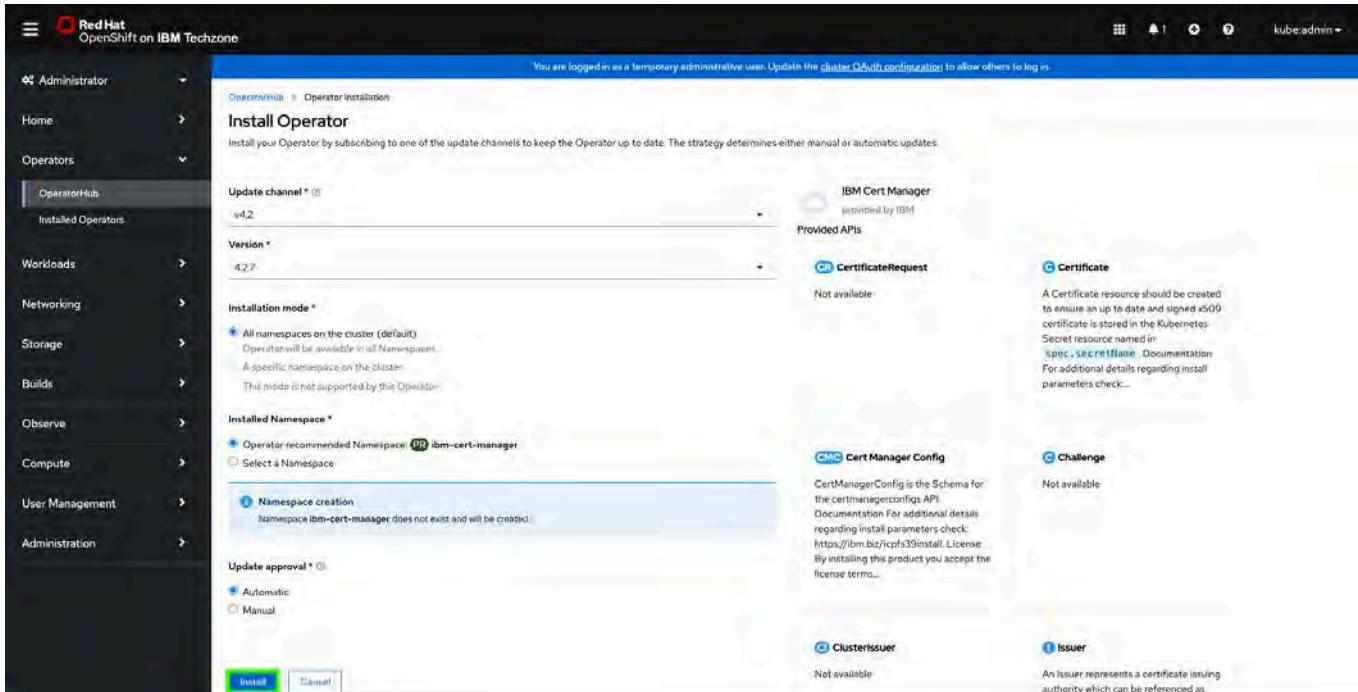
**Note:** The current version of the operator may differ than shown in the image below. Select the most current version.

This screenshot shows the search results for 'IBM Cert Manager' in the OperatorHub. The sidebar and main layout are identical to the previous screenshots. The search bar at the top of the main area has the text 'Q. IBM Cert Manager'. Below the search bar, the results show one item: 'ibm-cert-manager-4.2.7' (highlighted with a green box). The details for this item are: 'IBM Cert Manager' provided by 'IBM'. Description: 'Operator for managing deployment of cert-manager service.' The status bar at the bottom right shows 'kubeadmin'.

## 7. Click Install.



## 8. Keep the default settings and click Install.



**Do not continue until...**

The installation process takes a few minutes. Do not continue until you see the following message: **Installed operator: ready for use.**

## Install the watsonx Assistant for Z Operator (for OpenSearch)

1. In your command prompt or terminal window, create a new namespace called `wxa4z-byos` in the Red Hat OpenShift cluster.

```
oc create namespace wxa4z-byos
```

2. Create or obtain your IBM Container Software production entitlement key.

A production entitlement key is required to pull the container images that get deployed by the operator.

To create or retrieve your existing entitlement key, follow the instructions [here](#).

If you don't have an entitlement key at the above link, click the **Add new key** to create a new one or [visit this link to create a new one](#).

If extra assistance is needed, refer to this [site](#). Note, the process here is not

Locate your existing key or create a new one and continue to the next step.

3. Click **copy** and record your entitlement key for future use in a secure location.

The screenshot shows the 'Entitlement keys (1)' section of the IBM Container Software and Cloud Pak Access Management interface. It includes a brief description of what an entitlement key allows, a note about active entitlement keys, and a table with one row of data. The 'Copy' button for the first row is highlighted with a green border.

4. In your command prompt or terminal window, set an environment variable with your production entitlement key.

Substitute your production entitlement key copied in the last step for <entitlement key> .

Mac OS:

```
export IBM_CS_ENT_KEY=<entitlement key>
```

Microsoft Windows:

```
set IBM_CS_ENT_KEY=<entitlement key>
```

5. Enter the following command to create a pull secret for the **Container Registry**.

Mac OS:

```
oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --  
docker-username=cp --docker-password=$IBM_CS_ENT_KEY
```

Microsoft Windows:

```
oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --  
docker-username=cp --docker-password=%IBM_CS_ENT_KEY%
```

The terminal window shows the user navigating to the namespace 'wxa4z-byos' and running the command to create a secret. The command 'secret/icr-pull-secret created' is highlighted with a green border.

6. In a text editor, create a file that is named `catalogSource.yaml` and paste the following text in the file.

**Formatting of the yaml file is critical!**

The content of YAML files must be formatted exactly as shown. Use the copy icon to prevent typographical errors.

File name:

```
catalogSource.yaml
```

File contents:

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-wxa4z-operator-catalog
  namespace: wxa4z-byos
spec:
  displayName: "IBM watsonx Assistant for Z Operator Catalog"
  image: icr.io/cpopen/ibm-wxa4z-
  catalog:v2.1.0@sha256:a085d360b6aa0e40cf86a632eb5cd190a0407d1c54ec1b2d1d2fb5507f39a524
  publisher: 'IBM'
  sourceType: grpc
  secrets:
  - icr-pull-secret
```

7. Create your document catalog in the Red Hat OpenShift operator.

```
oc apply -f catalogSource.yaml
```

andrewjones@Andrews-MBP watsonxAssistant % oc create namespace wxa4z-byos  
namespace/wxa4z-byos created  
andrewjones@Andrews-MBP watsonxAssistant % export IBM\_CS\_ENT\_KEY=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJQk0gTWFya2V0cGxhY2UiLCJpYXQiOjE2NjYwMTk1ODAsImp0aSI6IjNkOWUyMzzjZTAzMDQzMzVhNTJhYTkzMWNmOTcyMDR1In0.4M3XRDXzkHMS0kFNJ4uKVcWZ6SnEA0Z03eL\_1iA2xy  
andrewjones@Andrews-MBP watsonxAssistant % oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --docker-username=cp --docker-password=\$IBM\_CS\_ENT\_KEY  
secret/icr-pull-secret created  
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml  
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml  
andrewjones@Andrews-MBP watsonxAssistant % oc apply -f catalogSource.yaml  
catalogsource.operator.coreos.com/ibm-wxa4z-operator-catalog created  
andrewjones@Andrews-MBP watsonxAssistant %

8. In the Red Hat OpenShift web console, click **OperatorHub** and select the **wxa4z-byos** project.

The screenshot shows the Red Hat OpenShift OperatorHub interface. The left sidebar has 'OperatorHub' selected. The main area shows a search bar with 'Project: openshift-marketplace'. Below it is a list of operators categorized by source: Community, Marketplace, and Red Hat. One operator, 'ibm-watsonx-for-z-operator', is highlighted with a green box. The total count of items is 624.

9. Enter **ibm watsonx** in the search field and the click the **IBM watsonx Assistant for Z Operator Catalog** tile.

**Be patient.**

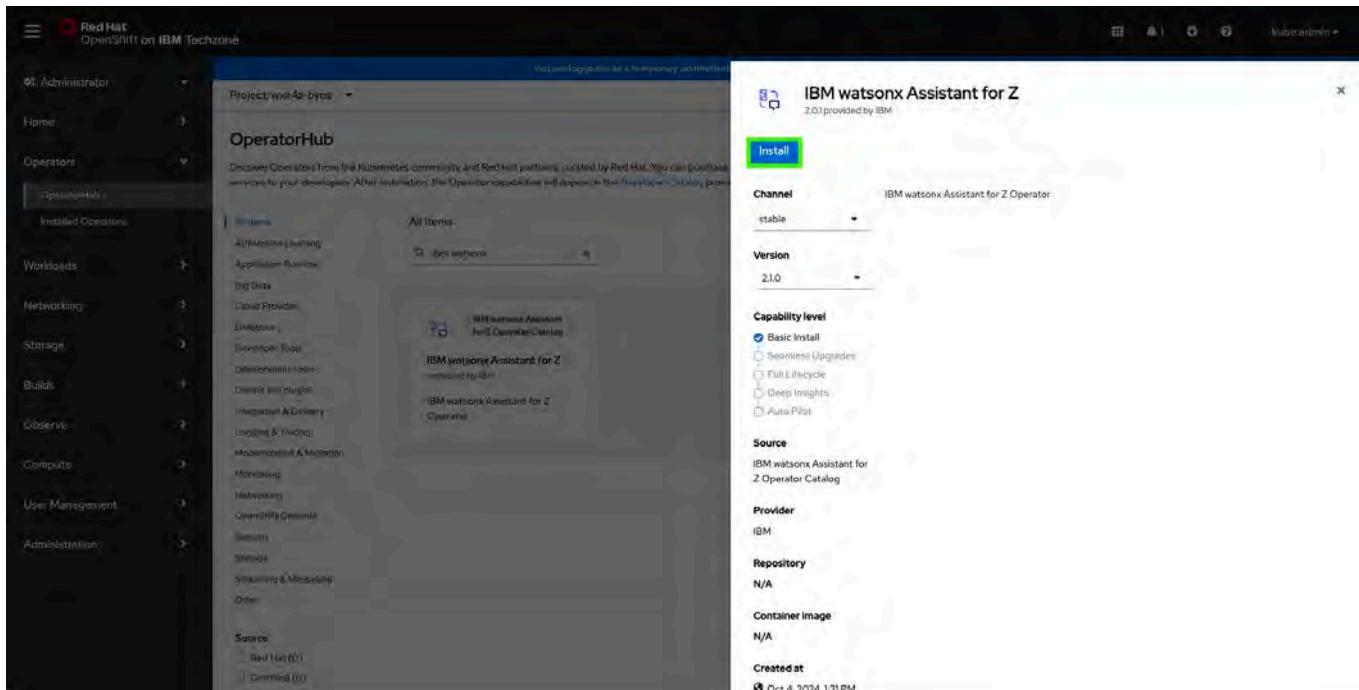
It may take a minute or two for the **IBM watsonx Assistant for Z Operator Catalog** tile to appear. Reload the browser page if the operator is not listed.

**Note:** The current version of the operator may differ than that shown in the image below.

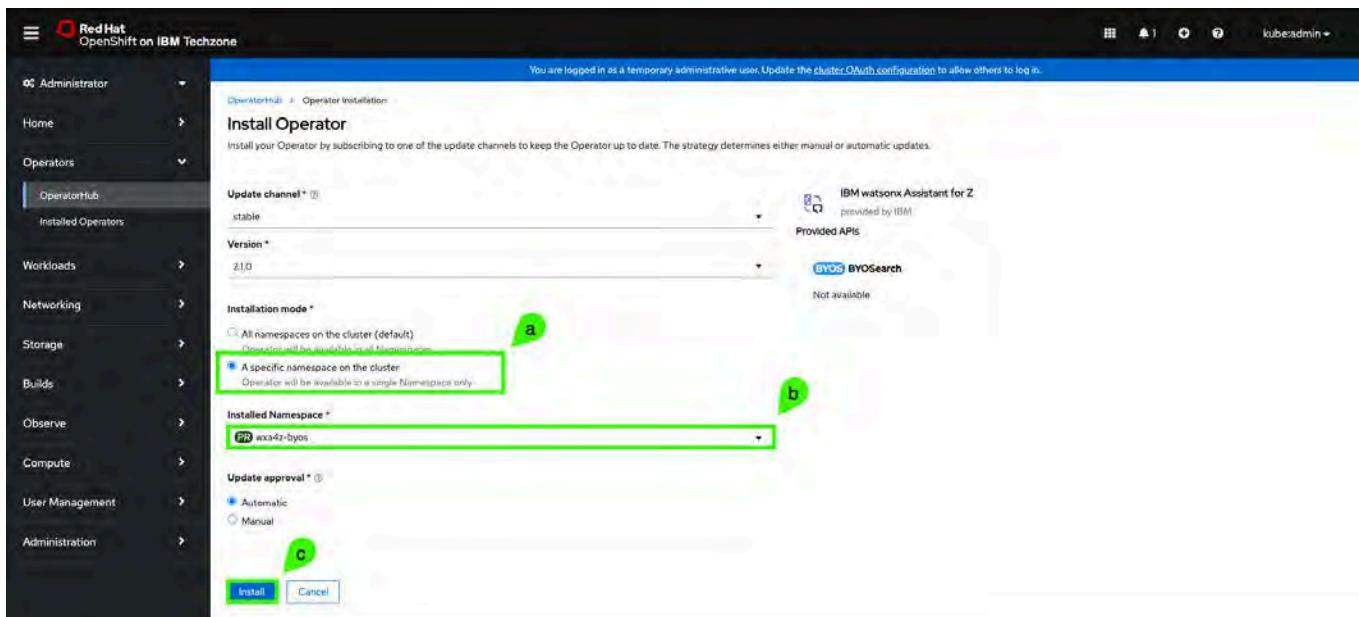
The screenshot shows the Red Hat OpenShift OperatorHub interface with a search bar containing 'ibm watsonx'. The results list shows one item: 'IBM Watson Assistant for Z Operator Catalog' provided by IBM. This item is also highlighted with a green box.

10. Click **Install**.

**Note:** The current version of the operator may differ than the one shown in the image after this. Select the most current version.



11. Select **A specific namespace on the cluster (a)** under **Installation mode** and **wxa4z-byos (b)** for the **Installed Namespace**, then click **Install (c)**.



**Do not continue until...**

The installation process takes a few minutes. Do not continue until you see the following message: **Installed operator: ready for use.**

- In your command prompt or terminal window, run the following commands to add the Container Registry credential to the operator's service account.

Mac OS and Microsoft Windows:

```
oc project wxa4z-byos
```

Mac OS:

```
oc patch serviceaccount ibm-wxa4z-operator-controller-manager --type merge -p
'{"imagePullSecrets": [{"name": "icr-pull-secret"}]}'
```

Microsoft Windows:

```
oc patch serviceaccount ibm-wxa4z-operator-controller-manager --type merge -p
'{"imagePullSecrets": [{"name": "icr-pull-secret"}]}'
```

```
andrewjones@Andrews-MBP watsonxAssistant % oc create namespace wxa4z-byos
namespace/wxa4z-byos created
andrewjones@Andrews-MBP watsonxAssistant % export IBM_CS_ENT_KEY=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJJQk0gTWFya2V0cGxhY2UiLCJpYXQiOjE2NjYwMTk1ODAsImp0aS16IjNkOWUyMzJzTAzMDQzMzNhNTJhYTkzMWNmOTcyMDR1In0.4M3XRD4XzkHMS0kFNJ4uKVcWZ6SnEA0Z03el_11A2xY
andrewjones@Andrews-MBP watsonxAssistant % oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --docker-username=cp --docker-password=$IBM_CS_ENT_KEY
secret/icr-pull-secret created
andrewjones@Andrews-MBP WatsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % oc apply -f catalogSource.yaml
catalogsource.coreos.com/ibm-wxa4z-operator-catalog created
andrewjones@Andrews-MBP watsonxAssistant % oc project wxa4z-byos
Now using project "wxa4z-byos" on server "https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443".
andrewjones@Andrews-MBP watsonxAssistant % oc patch serviceaccount ibm-wxa4z-operator-controller-manager --type merge -p '{"imagePullSecrets": [{"name": "icr-pull-secret"}]}'
serviceaccount/ibm-wxa4z-operator-controller-manager patched
andrewjones@Andrews-MBP watsonxAssistant %
```

- In the Red Hat OpenShift web console, under **Workloads**, click **Pods**.

The screenshot shows the Red Hat OpenShift web interface. On the left, the navigation sidebar includes 'Administrator', 'Home', 'Operators', 'OperatorHub', 'Workloads' (which is selected), 'Pods' (which is also selected), 'Deployments', 'DeploymentConfigs', 'StatefulSets', and 'Secrets'. The main content area displays a pod named 'ibm-wxa4z-operator' with a status of 'Running' and a green checkmark icon.

14. Verify the two pods that start with **ibm-wxa4z-operator** have a status of **Running** and that all pods are **Ready**.

The screenshot shows the Red Hat OpenShift web interface with the 'wxa4z-byos' project selected. The 'Pods' section lists several pods. Two specific pods are highlighted with green boxes: 'ibm-wxa4z-operator-catalog' and 'ibm-wxa4z-operator-controller-manager'. Both of these pods are listed as 'Running' with green dashed boxes around their status indicators.

15. Run the following command to set the administrative policy for the workspace.

```
oc -n wxa4z-byos adm policy add-scc-to-user privileged -z byos
```

```
andrewjones@Andrews-MBP watsonxAssistant % oc create namespace wxa4z-byos
namespace/wxa4z-byos created
andrewjones@Andrews-MBP watsonxAssistant % export IBM_CS_ENT_KEY=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJJQk0gTWFya2V0cGxhY2UiLCJpYXQiOjE2NjYwMTk1ODAsImp0aSI6IjNkOWUyMzjZTAzMDozMsVhNTJhYTkzMWNmOTcyMDRlIn0.4M3XRD4XzkHMSOkFNJ4uKvcWZ6SnEAZO3eL_11A2xY
andrewjones@Andrews-MBP watsonxAssistant % oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --docker-username=cp --docker-password=$IBM_CS_ENT_KEY
secret/icr-pull-secret created
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % vi catalogSource.yaml
andrewjones@Andrews-MBP watsonxAssistant % oc apply -f catalogSource.yaml
catalogsource.operator.coreos.com/wxa4z-operator-catalog created
andrewjones@Andrews-MBP watsonxAssistant % oc project wxa4z-byos
Now using project "wxa4z-byos" on server "https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443".
andrewjones@Andrews-MBP watsonxAssistant % oc patch serviceaccount ibm-wxa4z-operator-controller-manager --type merge -p '{"imagePullSecrets": [{"name": "icr-pull-secret"}]}'
serviceaccount/ibm-wxa4z-operator-controller-manager patched
andrewjones@Andrews-MBP watsonxAssistant % oc -n wxa4z-byos adm policy add-scc-to-user privileged -z byos
clusterrole.rbac.authorization.k8s.io/system:openshift:scc:privileged added: "byos"
andrewjones@Andrews-MBP watsonxAssistant %
```

## Deploy required secrets and the custom bring-your-own-search (BYOSearch) resources

- In a text editor, create a file that is named `os-secret.yaml`, paste the following text in the file, and then modify the default password.

File name:

```
os-secret.yaml
```

**Substitute a secure password of your choosing for the string <OPENSEARCH\_PASSWORD>. Record this value for later use.**

File contents:

```
apiVersion: v1
stringData:
  password: <OPENSEARCH_PASSWORD>
kind: Secret
metadata:
  name: opensearch-creds
  namespace: wxa4z-byos
type: Opaque
```

2. Create the secret by running the following command.

```
oc apply -f os-secret.yaml
```

3. In a text editor, create a file that is named `client-ingestion-secret.yaml`, paste the following text in the file, and then modify the default password.

File name:

```
client-ingestion-secret.yaml
```

**Substitute a secure authentication key of your choosing for the string <CLIENT\_INGESTION\_AUTHKEY>. The authentication key can be a random password. Record this value for later use.**

File contents:

```
apiVersion: v1
stringData:
  authkey: <CLIENT_INGESTION_AUTHKEY>
kind: Secret
metadata:
  name: client-ingestion-authkey
  namespace: wxa4z-byos
type: Opaque
```

4. Create the secret by running the following command.

```
oc apply -f client-ingestion-secret.yaml
```

5. In a text editor, create a file that is named `wrapper-creds.yaml`, paste the following text in the file, and then modify the default password.

File name:

```
wrapper-creds.yaml
```

**Substitute a secure password credential of your choosing for the string <WRAPPER\_PASSWORD> . The password can be a random password. Record this value for later use. Use this password in the following steps when you configure your BYOS connection in your assistant to connect to the network route.**

File contents:

```
apiVersion: v1
stringData:
  username: admin
  password: <WRAPPER_PASSWORD>
kind: Secret
metadata:
  name: wrapper-creds
  namespace: wxa4z-byos
type: Opaque
```

6. Create the secret by running the following command.

```
oc apply -f wrapper-creds.yaml
```

7. Obtain and record your cluster domain that is used for routes by running the following command.

```
oc -n openshift-ingress-operator get ingresscontroller default -o jsonpath="{.status.domain}"
```



**The output from the command does not include a newline.**

The value returned for the cluster domain does not include a newline. When copying the value do not include the character or characters used for your command line prompt. Do not include the your prompt in the next step!

**Note:** The output of the command will be a string similar to:

**apps.672b79320c7a71b728e523b4.ocp.techzone.ibm.com**

8. In a text editor, create a file that is named `byos.yaml` and paste the following text in the file.

File name:

```
byos.yaml
```

Substitute the domain name recorded in the previous step for the string <YOUR\_CLUSTER\_DOMAIN> .

File contents:

```

apiVersion: wxa4z.watsonx.ibm.com/v1
kind: BYOSearch
metadata:
  name: byosearch
  namespace: wxa4z-byos
spec:
  imagePullSecrets:
    - name : icr-pull-secret
  namespace: wxa4z-byos
  clusterName: wxa4z-byos-cluster
  clusterDomain: <YOUR_CLUSTER_DOMAIN>

  opensearch:
    secretName: opensearch-creds

  persistence:
    enabled: true
    storageClass: "managed-nfs-storage"
    accessModes:
      - ReadWriteOnce
    size: 24Gi

  wrapper:
    createRoute: true
    resources:
      requests:
        cpu: 2
        memory: "500Mi"
      limits:
        cpu: 2
        memory: "1Gi"

  clientIngestion:
    secretName: client-ingestion-authkey

    resources:
      limits:
        cpu: "500m"
        memory: 2Gi
        nvidia.com/gpu: "0"
      requests:
        cpu: "500m"
        memory: 1Gi
        nvidia.com/gpu: "0"
    pvc:
      storageClass: "managed-nfs-storage"
      enabled: true
      size: 24Gi

```

9. Run the following command to deploy BYOS on your cluster.

```
oc apply -f byos.yaml
```

Verify all the required pods are running and get the network route to your BYOS instance

1. In the OCP console, verify that all pods have the status of **Running** or **Completed**.

**⚠️ Do not continue until...**

The BYOS deployment can take 20 minutes or more to complete. Do not continue until all the pods have a status of "Running" or "Completed". The next step is to retrieve your BYOS endpoint URL.

Name	Status	Ready	Restarts	Owner	Memory	CPU	Created
ibm-wxa4z-operator-catalog-ctvsv	Running	1/1	0	RS ibm-wxa4z-operator-catalog-556fcf9bb	25.0 MiB	0.004 cores	Nov 6, 2024, 9:25 AM
ibm-wxa4z-operator-controller-manager-556fcf9bb	Running	2/2	0	RS ibm-wxa4z-operator-controller-manager-556fcf9bb	142.8 MiB	0.023 cores	Nov 6, 2024, 9:26 AM
wxa4z-byos-cluster-0	Running	1/1	0	SS wxa4z-byos-cluster	1,095.8 MiB	0.017 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-1	Running	1/1	0	SS wxa4z-byos-cluster	1,169.1 MiB	0.016 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-2	Running	1/1	0	SS wxa4z-byos-cluster	1,113.3 MiB	0.015 cores	Nov 6, 2024, 10:22 AM
wxa4z-client-ingestion-7fb8d6c58-98th	Running	1/1	0	RS wxa4z-client-ingestion-7fb8d6c58	501.8 MiB	0.071 cores	Nov 6, 2024, 12:18 PM
wxa4z-opensearch-wrapper-5cb879f58-qw7qt	Running	1/1	0	RS wxa4z-opensearch-wrapper-5cb879f58	547.1 MiB	0.031 cores	Nov 6, 2024, 10:22 AM
wxa4z-snapshot-setup-job-nsqtz	Completed	0/1	0	U wxa4z-snapshot-setup-job	-	-	Nov 6, 2024, 10:22 AM

2. Under **Networking**, click **Routes**.

Name	Status	Ready	Restarts	Owner	Memory	CPU	Created
ibm-wxa4z-operator-catalog-ctvsv	Running	1/1	0	RS ibm-wxa4z-operator-catalog-556fcf9bb	25.0 MiB	0.004 cores	Nov 6, 2024, 9:25 AM
ibm-wxa4z-operator-controller-manager-556fcf9bb	Running	2/2	0	RS ibm-wxa4z-operator-controller-manager-556fcf9bb	141.2 MiB	0.022 cores	Nov 6, 2024, 9:26 AM
wxa4z-byos-cluster-0	Running	1/1	0	SS wxa4z-byos-cluster	1,095.8 MiB	0.016 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-1	Running	1/1	0	SS wxa4z-byos-cluster	1,169.1 MiB	0.016 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-2	Running	1/1	0	SS wxa4z-byos-cluster	1,113.3 MiB	0.015 cores	Nov 6, 2024, 10:22 AM
wxa4z-client-ingestion-7fb8d6c58-98th	Running	1/1	0	RS wxa4z-client-ingestion-7fb8d6c58	544.6 MiB	0.097 cores	Nov 6, 2024, 12:18 PM
wxa4z-opensearch-wrapper-5cb879f58-qw7qt	Running	1/1	0	RS wxa4z-opensearch-wrapper-5cb879f58	547.1 MiB	0.030 cores	Nov 6, 2024, 10:22 AM
wxa4z-snapshot-setup-job-nsqtz	Completed	0/1	0	U wxa4z-snapshot-setup-job	-	-	Nov 6, 2024, 10:22 AM

3. Copy and record the location for the **wxa4z-opensearch-wrapper** route.

Name	Status	Location	Service
wxa4z-client-ingestion	Accepted	https://wxa4z-client-ingestion-wxa4z-byos.apps.672b79320c7a71b728e523b4.ocp.techzone.ibm.com	wxa4z-client-ingestion
wxa4z-opensearch-wrapper	Accepted	https://wxa4z-opensearch-wrapper-wxa4z-byos.apps.672b79320c7a71b728e523b4.ocp.techzone.ibm.com	wxa4z-opensearch-wrapper

## Update your assistant with the new BYOS instance route

You are now ready to configure your assistant with the route to your BYOS instance.

1. Using the network route for your BYOS instance, append the string **/v1/query** to complete the URL endpoint.

The URL should look similar to:

```
https://wxa4z-opensearch-wrapper-wxa4z-
byos.apps.672b79320c7a71b728e523b4.ocp.techzone.ibm.com/v1/query
```

**Important:** The above URL will not work for you. Use the value of your specific OpenSearch instance that is recorded in the previous step.

2. Update your assistant's custom search integration URL.

Next, you need to return to your assistant in the watsonx Orchestrate AI assistant builder and update the custom search integration URL. Use the URL from the network route (with **/v1/query**) appended. Use **admin** for the **Username** and the **Password** will be the password that you specified in the `wrapper-creds.yaml` file.



### Don't recall how to set the customer search URL?

Refer back to [Creating an assistant and configuring conversational search](#) if you don't remember how to specify the customer search URL.

3. Test your assistant and verify that it is still answering questions that are related to IBM Z.

Experiment with different prompts and validate that the answers provided are reasonable, and that you can view the documentation that was sourced. If responses are not received as expected, verify that the URL is formatted correctly and you specified the `wrapper-creds.yaml` password as the **admin** password.

## Troubleshooting

The following are issues that you may encounter. If the provided resolutions do not work, contact support by using the methods that are mentioned in the [Support](#) section.

#### Pods have a status of ErrImagePull or ImagePullBackoff

If the pods starting with **ibm-wxa4z-operator** have a status of “ErrImagePull” or “ImagePullBackoff”, you can delete the pod and it will automatically restart and pull the image successfully. Wait until the pod is re-created successfully.

#### The wxa4z-client-ingestion pod does not start

Did you include the % character in the **clusterDomain** name when creating the **byos.yaml**? To resolve, edit the **byos.yaml** file and run the following command again. The current pod will be terminated and a new one started. This will take about 20 minutes to start.

```
oc apply -f byos.yaml
```

# Installing and using zassist to ingest client documents

With bring-your-own-search (BYOS) installed and configured in your assistant, you can now prepare for document ingestion (bring-your-own-documents (BYOD)). BYOD demonstrates how clients can augment their assistant's conversational search by creating an internal knowledge base with their documentation. Using the client's documentation allows the assistant to provide valuable responses to a range of questions not possible with the default documentation available.

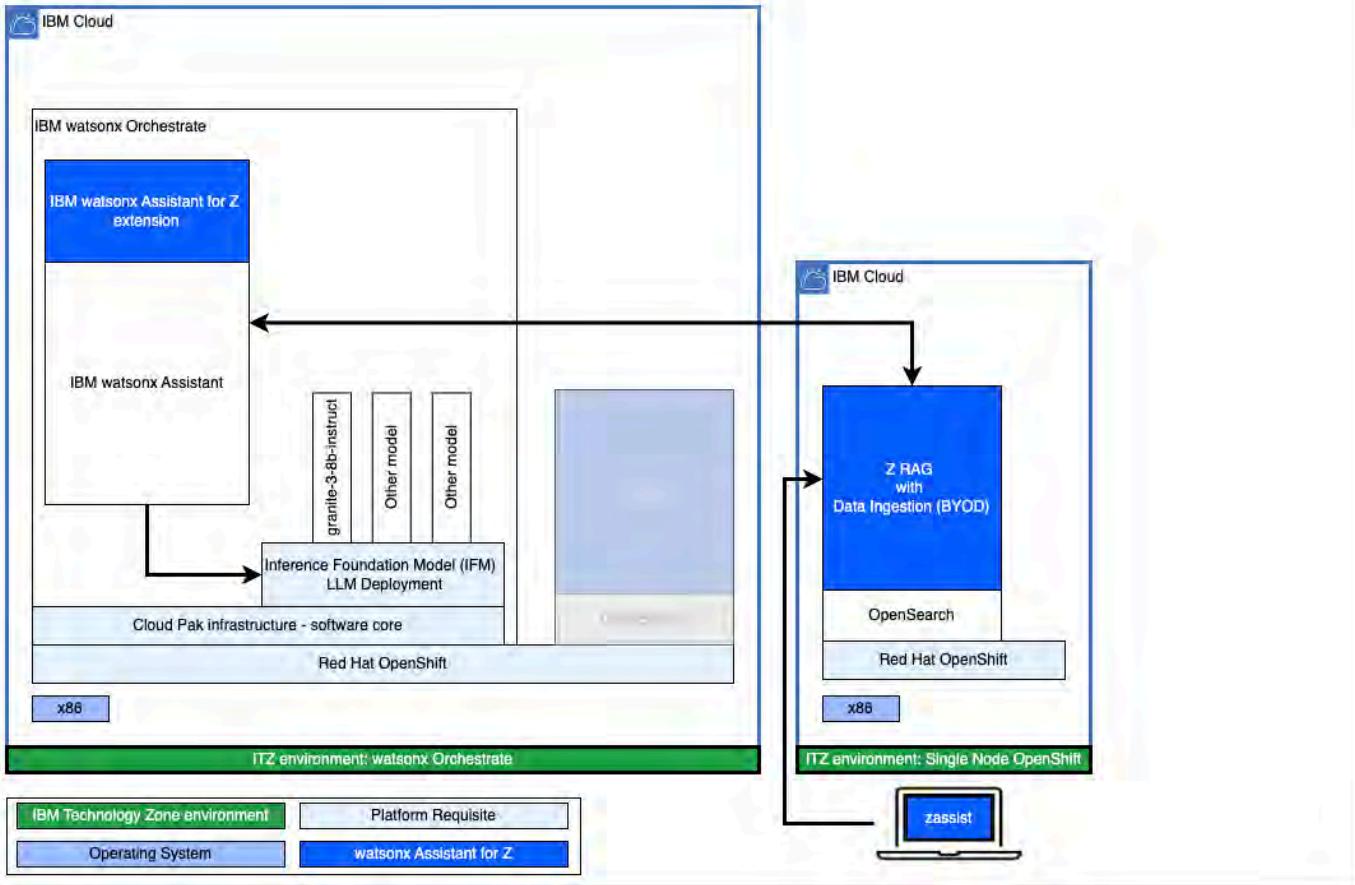
As an example, a client mentioned that their developers often need reference material on company-specific legacy code or company-specific syntax. The users must search through volumes of documentation to find it or look at old code. Also, a need for their operational support group to quickly determine how to resolve technical issues using runbooks exists.

You can show your client how watsonx Assistant for Z can assist developers and operational support personnel in finding answers about internal processes for code development and deployment.

Currently, only PDF, HTML, and DOCX file formats are supported for ingestion.

A high-level, logical architecture of the environment is illustrated in the following diagram.

## Installing and using zassist to ingest client documents



To prepare for document ingestion, you can also reference the setup instructions that are located [here](#).

## Install the zassist utility

The **zassist** utility is an executable program that automates the ingestion of client documentation into the RAG for Watsonx Assistant for Z. A version of zassist is available for download for IBMers and Business Partners for conducting pilots. Follow the steps to download and install **zassist**.

### How do clients get the zassist utility?

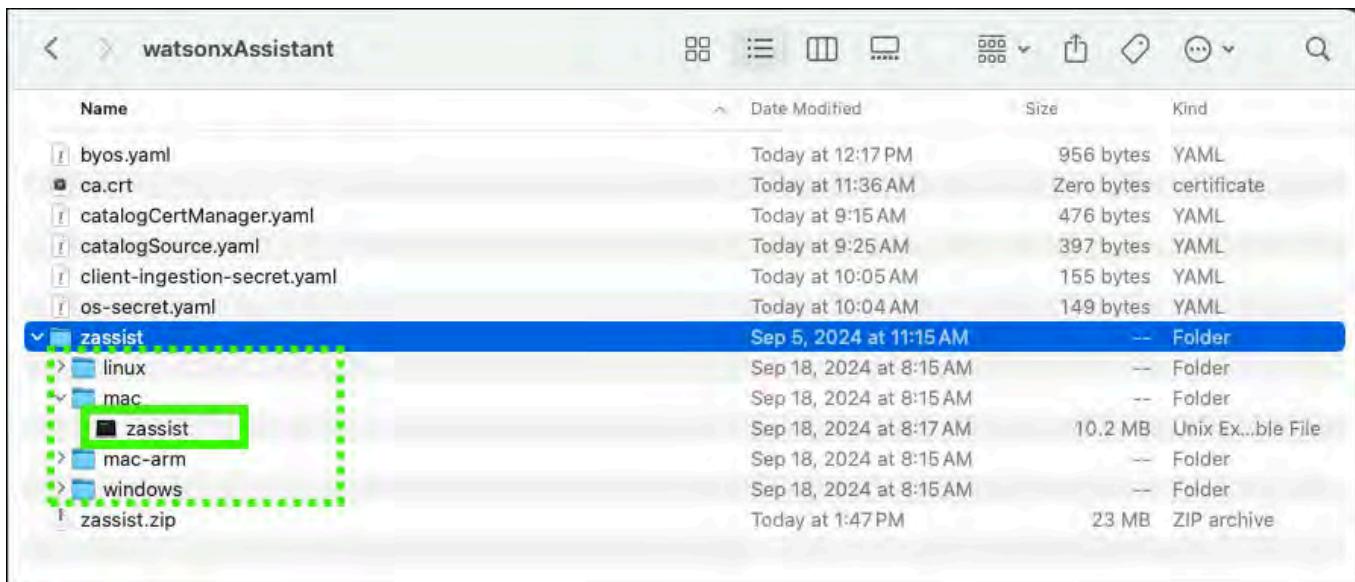
The utility is available to clients through [IBM Passport Advantage](#).

1. Click the following link and download the **zassist.zip** file.

<https://ibm.box.com/s/j3nt5iw4fqd5w2jgcqwxnjlsu8bpvl77>



2. Extract the **zassist.zip** file.
3. Locate the appropriate file for your local workstation's operating system.



4. Either copy the appropriate **zassist** file to a directory in your PATH, or add the appropriate directory to your PATH environment variable.

Additional information for running the preceding tasks can be found [here](#).



#### Windows users may need to rename the file **zassist** file!

If the **zassist** file does not execute properly, rename the file to **zassist.exe**.

5. Run the **zassist** command to verify that it is working.

```
zassist
andrewjones@Andrews-MacBook-Pro watsonxAssistant % PATH=$PATH:.
andrewjones@Andrews-MacBook-Pro watsonxAssistant % zassist
zassist: error: expected one of "version", "init", "login", "ingest", "load", ...
andrewjones@Andrews-MacBook-Pro watsonxAssistant %
```



**Mac/OS users may need to adjust security settings.**



The **zassist** binary may cause a security exception. Adjust the security settings by opening the **System Settings** utility and clicking **Privacy & Security**. Under **Security** locate the message about the **zassist** binary and click **Allow Anyway**. Return to the terminal window and try running the command again.

## Ingest client documentation using zassist

With the **zassist** command installed, you are now able to begin ingesting data.

Step-by-step guidance for ingesting documents using **zassist** is provided in the IBM watsonx Assistant for Z documentation.

1. Download the `BYOD.zip` file.

[BYOD.zip](#)



### What is in the sample client documentation?

Three sample documents are included:

- `Mainframe_COBOL_Error_Codes.pdf`

This is a document containing company-specific mainframe COBOL error codes for their application.

Developers within the organization typically review this document to quickly diagnose issues based on the application error codes returned.

- `Mainframe_Operational_Incidents_Log.pdf`

This document is leveraged by the organization's operational support team and contains historical records of production-level incidents that occurred. For each incident, there's a record of what the incident was, the date, how it was resolved and who was involved in resolving the incident.

- `COBOL-CICS-to-Java-Internal-Framework.pdf`

This document is leveraged by the development team and contains details about the organization's internal framework for developing applications consisting of legacy COBOL CICS interoperating with new Java code. Within the document contains company-specific coding practices and code syntax that the developers frequently reference.

2. Extract the `BYOD.zip` file.
3. Change to the `BYOD` directory.
4. Set the `TLS_VERIFY` environment variable to `false`.

Mac OS:

```
export TLS_VERIFY=false
```

Windows OS:

```
set TLS_VERIFY=false
```

Windows OS PowerShell:

```
Set-Item Env:\TLS_VERIFY "false"
```

5. Initialize the `zassist` environment.

```
zassist init
```

6. Retrieve the server URL for the client ingestion server.

Mac OS:

```
echo https://$(oc -n wxa4z-byos get route wxa4z-client-ingestion -o jsonpath=".spec.host")
```

The output of this command is your unique URL for your client ingestion server.

Windows OS (this method can also be used by Mac OS users):

You can retrieve the URL in your OCP Web console by navigating to Networking Routes and then copy the URL for the **wxa4z-client-ingestion** route.

Name	Status	Location	Service
wxa4z-client-ingestion	Accepted	https://wxa4z-client-ingestion-wxa4z-byos.apps.67850b0240c621f33ba8749.ocp.techzone.ibm.com	wxa4z-client-ingestion
wxa4z-opensearch-wrapper	Accepted	https://wxa4z-opensearch-wrapper-wxa4z-byos.apps.67850b0240c621f33ba8749.ocp.techzone.ibm.com	wxa4z-opensearch-wrapper

7. Retrieve the `client-ingestion-authkey`.

```
oc -n wxa4z-byos get secret client-ingestion-authkey -o jsonpath=".data.authkey" | base64 -d
```

The output of this command is your unique auth-key that you had previously set. You will need the output of both previous commands in the next step.

✖ If the command doesn't work for you...

You can find your `authkey` value by viewing the `client-ingestion-secret.yaml` file you created and copying the value set for the `authkey` parameter.

8. Login to your server. Replace <server url> with the value from step 6.

```
zassist login <server url>
```

9. When prompted, enter the password from step 7. Verify that a **Success** message is received.

10. Verify zassist is ready to ingest documents by checking the status.

```
zassist status
```

Local	Ingested	Loaded	Path
yes	no	no	dev/COBOL-CICS-to-Java-Internal-Framework.pdf
yes	no	no	dev/Mainframe_COBOL_Error_Codes.pdf
yes	no	no	ops/Mainframe_Operational_Incidents_Log.pdf

11. Ingest the documentation using the commands.

For the next steps, you must be in the root directory called BYOD.

```
zassist ingest dev
```

```
zassist ingest ops
```

12. Upload the ingested documents.

```
zassist load dev
```

```
zassist load ops
```

13. Verify that all documents were successfully ingested and loaded.

```
zassist status
```

```
andrewjones@Andrews-MBP BYOD % zassist ingest dev
Gathering local files...
Comparing uploaded files with local files...
Uploaded 2 documents
Started 2 embedding jobs
Completed 2 embedding jobs
andrewjones@Andrews-MBP BYOD % zassist ingest ops
Gathering local files...
Comparing uploaded files with local files...
Uploaded 1 documents
Started 1 embedding jobs
Completed 1 embedding jobs
andrewjones@Andrews-MBP BYOD % zassist load dev
Gathering local files...
Loaded 2 documents into Opensearch database
andrewjones@Andrews-MBP BYOD % zassist load ops
Gathering local files...
Loaded 1 documents into Opensearch database
andrewjones@Andrews-MBP BYOD % zassist status
Gathering local files...
Local Ingested Loaded Path
yes yes yes dev/COBOL-CICS-to-Java-Internal-Framework.pdf
yes yes yes dev/Mainframe_COBOL_Error_Codes.pdf
yes yes yes ops/Mainframe_Operational_Incidents_Log.pdf
andrewjones@Andrews-MBP BYOD %
andrewjones@Andrews-MBP BYOD %
```

## Verify that the assistant is using the ingested documents

Use the Watsonx Orchestrate AI assistant builder to verify your document ingestion.

1. Enter the following prompt in your assistant and **record the response** (cut and paste into a text file on your local machine).

The customer application is failing with ERR-CBL-001, what does this internal error mean?

The screenshot shows the Watsonx Orchestrate interface with the AI assistant named "Zeeves". The assistant's profile picture is a blue robot head. Below it, a message box displays the prompt: "The customer application is failing with ERR-CBL-001, what does this internal error mean?". To the right of the message box is a blue "Send" button.

2. Click the **Down arrow** to view the citations for the response.

### 3. Click View source for the Mainframe\_COBOL\_Error\_Codes-... citation.



**Take note of the order of the response citations!**

### 4. Accept the security risk to view the source document for any ingested document cited.

The steps to accept the security risk for the document are not shown as it varies by the browser you are using. The risk occurs because the certificate for the connection to the SNO instance is not secure. Notice that the URL contains the path to your SNO instance route.



5. Repeat the preceding steps for the following prompts in your assistant and **record the responses** (cut and paste into a text file on your local machine).

Are there any production incidents that were resolved in relation to Data corruption in the production database. If yes who can I collaborate with to resolve a similar issue today and what are their names?

What specific syntax changes do I need to make in COBOL to call Java using the internal framework? Please provide a detailed explanation.

What is the internal git lab link to execute the Java on z/OS pipeline?

## Adjusting the search behavior

Do you recall the **Metadata** field when you configured your assistant?

The screenshot shows the 'Custom service' configuration page in the IBM Watson Assistant interface. The 'Metadata' field is highlighted with a green dashed border. The field contains a JSON example: 
 

```
{
      "ibm_indices": "*_ibm_docs_slate,*_ibm_redbooks_slate",
      "customer_indices": "customer_"
    }
```

The Metadata field provides a way to adjust your assistant's behavior during conversational search for your OpenSearch instance. Now that you have your own docs that are ingested for conversational search, you can set the metadata field for your assistant to use those documents in its content-grounded search. If you leave the metadata field empty, then it defaults to settings found to perform well but may not use the ingested documents as part of the search results.

If you leave the Metadata field empty, OpenSearch relies on the default settings, which means OpenSearch searches all the default IBM-provided documentation and all of the ingested customer documentation using the following value:

```
{"ibm_indices": "*_ibm_docs_slate,*_ibm_redbooks_slate",
"customer_indices": "customer_*"}
```

Replacing the wildcard string with an explicit list of indices allows for personalization. The metadata setting is where you can input specific indices (pointing to the underlying documentation) that you want your assistant to use for the content-grounded search. There are over 220 products and topics that the OpenSearch instance has IBM Documentation for. You can find those indices and products [here](#).

You can input a subset of indices into the “Metadata” field in cases where you want your assistant to only gather context for specific IBM products or topics. The specific indices can be listed out in this format:

```
{"ibm_indices": "<comma separated index values>", "customer_indices": "customer_*"}
```

For example, if you want your assistant to reference only documentation for “Db2 Analytics Accelerator for z/OS” and no ingested client documentation, you can enter the following into the metadata field:

```
{"ibm_indices": "ss4lq8_ibm_docs_slate"}
```

If you have a mix of IBM Documentation and client documentation ingested, then there's an optional search string that you can use to set the "weights" used for each.

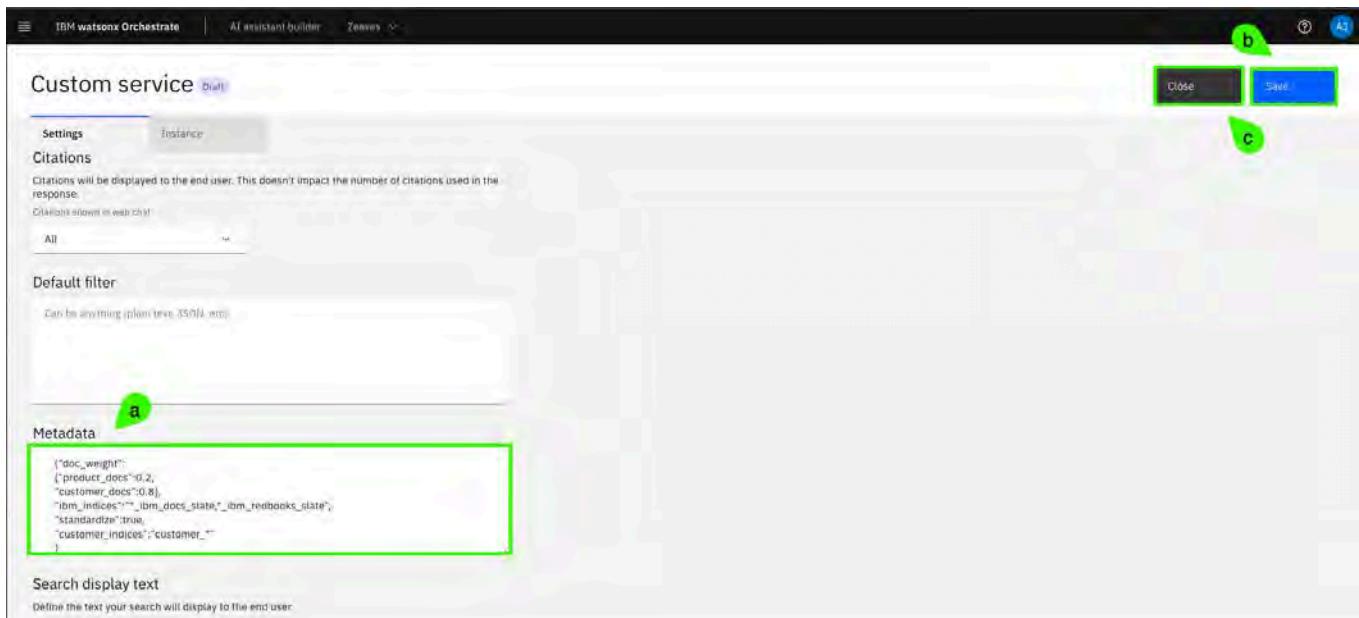
For example:

```
{"doc_weight":  
{"product_docs":0.5,  
"customer_docs":0.5},  
"ibm_indices": "*_ibm_docs_slate,*_ibm_redbooks_slate",  
"standardize":true,  
"customer_indices":"customer_*"  
}
```

In this case, "product\_docs" is the weight that is assigned to "ibm\_indices" and "customer\_docs" is the weight that is assigned to "customer\_indices". For more information on customizing the metadata field for conversational search, refer to this supplemental video found [here](#).

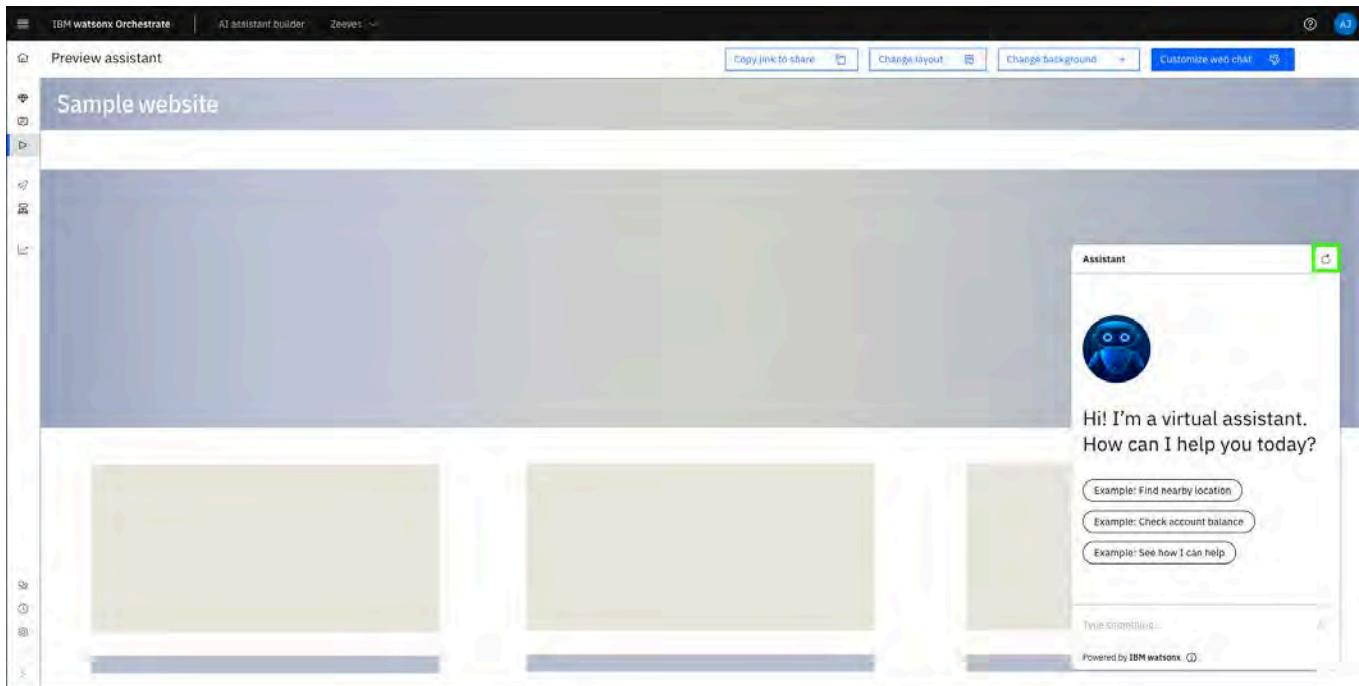
1. Set the (a) **Metadata** field for your BYOS custom search instance to the following value, click (b) **Save**, and then click (c) **Close**. Notice the weight for **customer\_docs** is heavier than the weight for **product\_docs**.

```
{"doc_weight":  
{"product_docs":0.2,  
"customer_docs":0.8},  
"ibm_indices": "*_ibm_docs_slate,*_ibm_redbooks_slate",  
"standardize":true,  
"customer_indices":"customer_*"
```



2. Hover over the **Home** (🏠) icon and click **Preview**.

3. Click the **Restart conversation** (⟳) icon.



#### 4. Repeat the queries four queries run earlier and record the results and the order of the response citations.

Compare the two sets of results. Notice how the answers changed based on the weighting of the ingested documents versus the IBM product documentation. Were the ingested documents always the first document cited? If not, why do you think that is?

Before proceeding, experiment with different metadata and other configuration settings for your custom service instance.

#### For client pilots

If you or your client have other documents to ingest, you can do so by repeating the steps using **zassist**. The Velocity Pilot ITZ environment is limited in compute and storage capacity. The following limits should be adhered to:

- Loading documents can take a long time, especially with > 100 MB of text.
- It is recommended to run large loads late at night.
- When loading, ensure your workstations does not sleep during the process.
- If you receive a **batch time error**, set the batch size to a lower number for that command. For example:

```
zassist ingest . -s 50
```

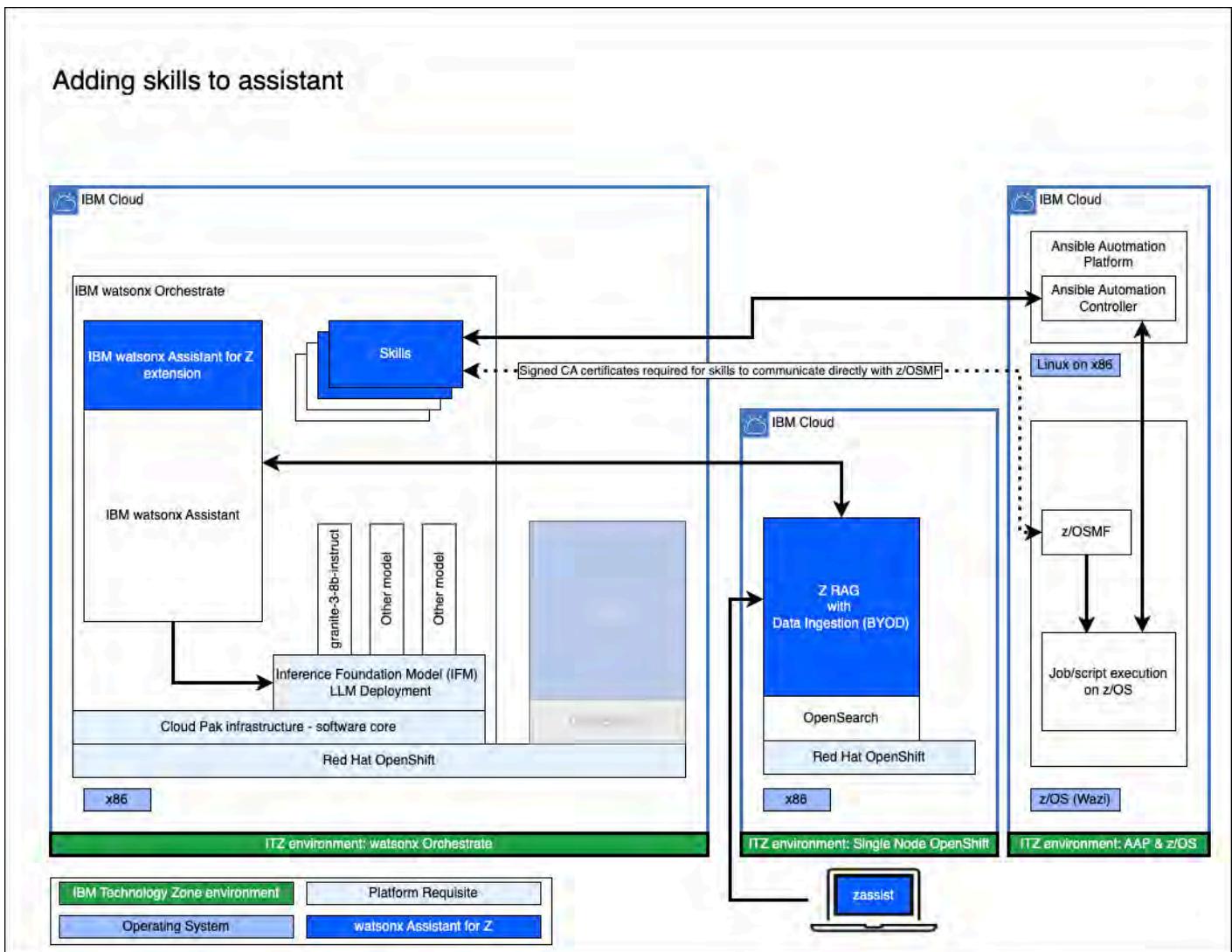
## Skills and skill flows

# Getting started with skills and actions

Watsonx Assistant for Z automates a range of IBM Z related tasks through assistant interactions by importing skills. Each skill is a pre-defined automation that accomplishes some unit or units of work by running tasks. For example, skills can view z/OS IPL information or work with z/OS datasets.

Watsonx Assistant for Z extends Watsonx Orchestrate, allowing users to build new skills from Ansible Automation platform or z/OS Management Facility (z/OSMF) through the Z Skills Accelerator extension. The Z Skills Accelerator extension connects Ansible and z/OS application programming interfaces (APIs) and imports automation as Ansible Playbooks, JCL, or REXX as skills. Learn more importing and building skills [here](#).

A high-level, logical architecture of the environment is illustrated in the figure that follows.



## Environments

### Watsonx Orchestrate

The Z Skills Accelerator extension is already configured in your watsonx Orchestrate IBM Technology Zone (ITZ) environment. You can use this component to import new skills.

## Ansible Automation Platform and Wazi as a Service

To import skills for automations, you use Ansible Automation Platform (AAP) and Wazi as a Service (Wazi aaS) to serve as the z/OS back-end. Learn more about AAP [here](#). Learn more about Wazi, [here](#).

The two resources are provisioned together in the ITZ environment that you reserved earlier. This environment enables the ability to manage and automate z/OS tasks and subsystems with various preinstalled Ansible playbooks. It includes a z/OS back-end (Wazi as a Service) with all needed prerequisites.

The playbooks provided cover various use cases for automating z/OS management. Ansible's capabilities for automating various Z-specific tasks are not limited to the use cases that are preinstalled in the AAP instance. The preinstalled playbooks are tasks from the 'IBM z/OS core collection'. Using this environment accelerates the ability to showcase the value of watsonx assistant for Z, and to get started with simple automations that can be expanded.

The ITZ environment gives you access to AAP, which is preconfigured to target the accompanying z/OS Wazi system, along with web-based access to AAP to experiment with different playbook templates. These templates are imported into watsonx Orchestrate as skills and connected to your assistant.

For more information on the AAP and Wazi z/OS environments, refer to this [document](#).

The playbook templates that are preinstalled in AAP cover various use cases, which you can explore, including:

- z/OS Certificate Management (create, delete, list, and renew certificates)
- dataset management (create, delete, fetch datasets)
- Submit JCL
- Run Operator commands
- Run TSO commands
- And more

Each of the sections that follow build upon each other. Complete each to successfully enhance your assistant by starting with [Explore Ansible Automation Platform](#).

# Explore Ansible Automation Platform

After you reserved the Ansible Automation Platform (AAP) and Wazi z/OS environment in IBM Technology Zone (ITZ) and the reservation is in the **Ready** state, follow these steps to explore AAP.

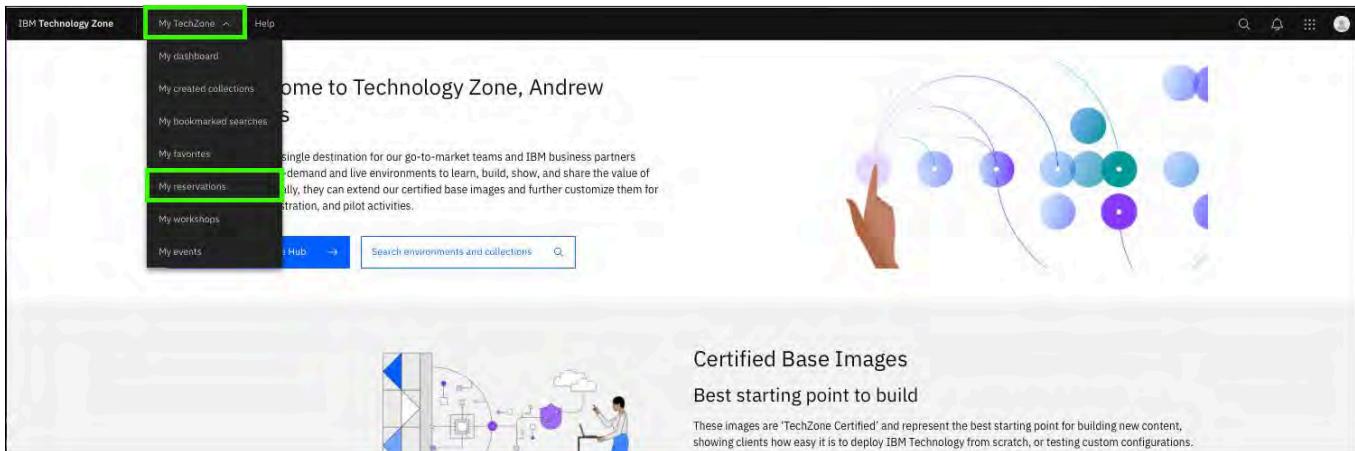
## Access the AAP and Wazi as a Service environment

### Be sure to record the information as instructed

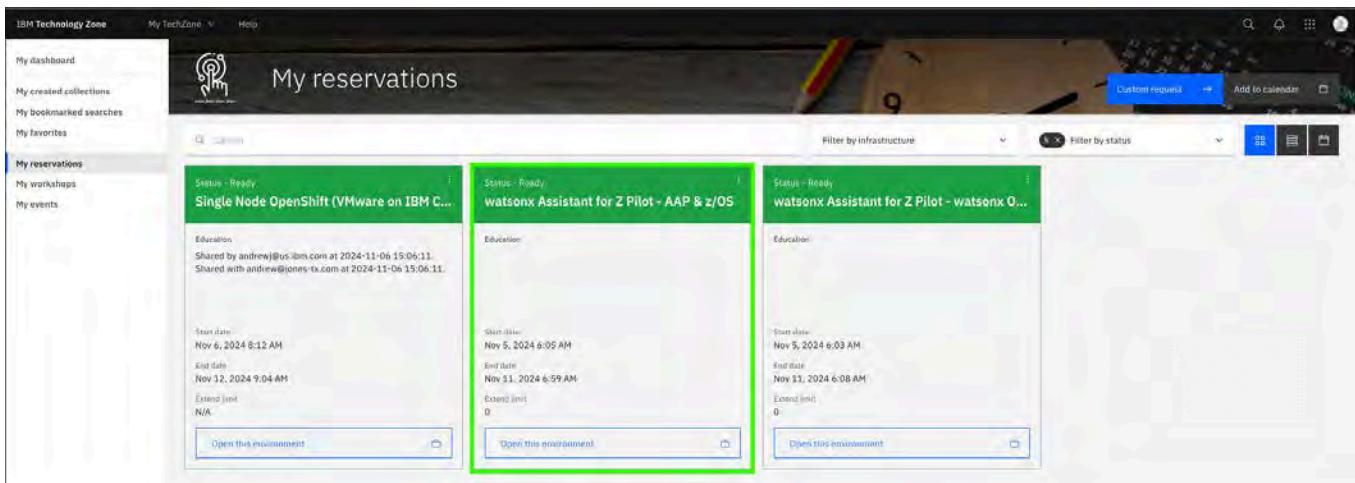
Several of the steps below instruct you to record values from your ITZ reservation. Be sure to do this as they will not only be used in this section, but also in later sections of the lab guide.

1. In the IBM Technology Zone portal, expand **My TechZone** and select **My Reservations**, or click the following link.

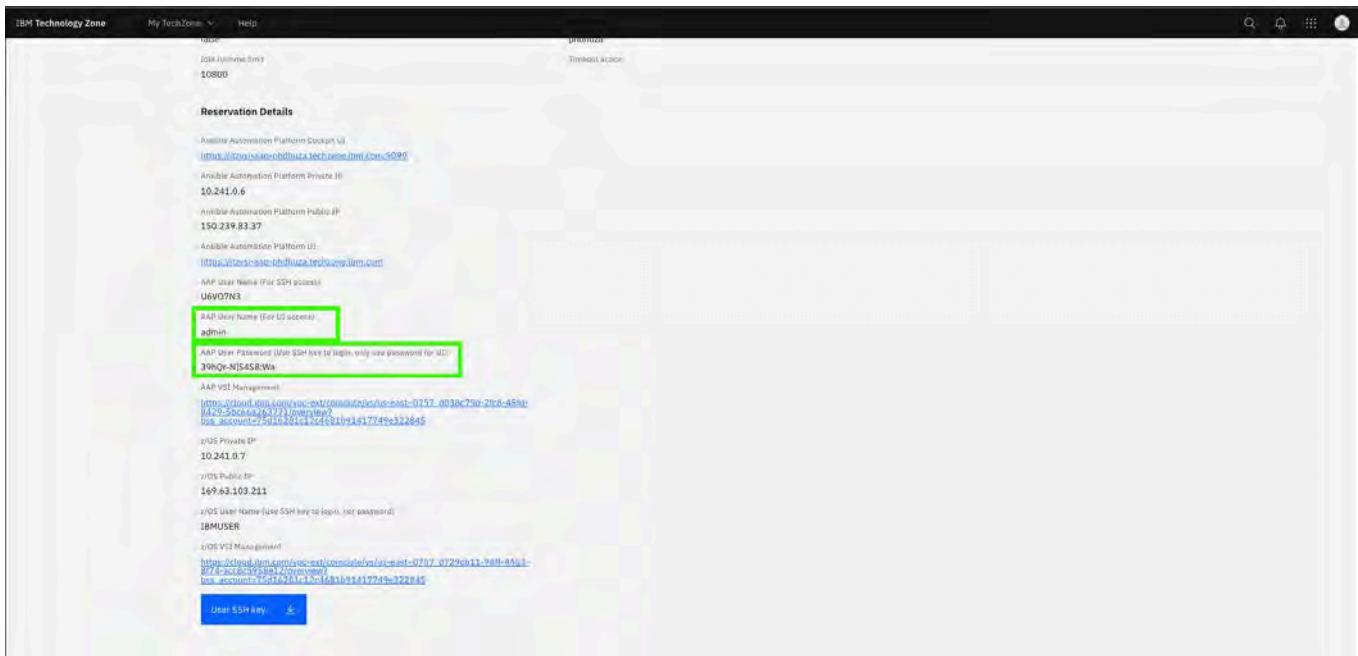
#### [ITZ My reservations](#)



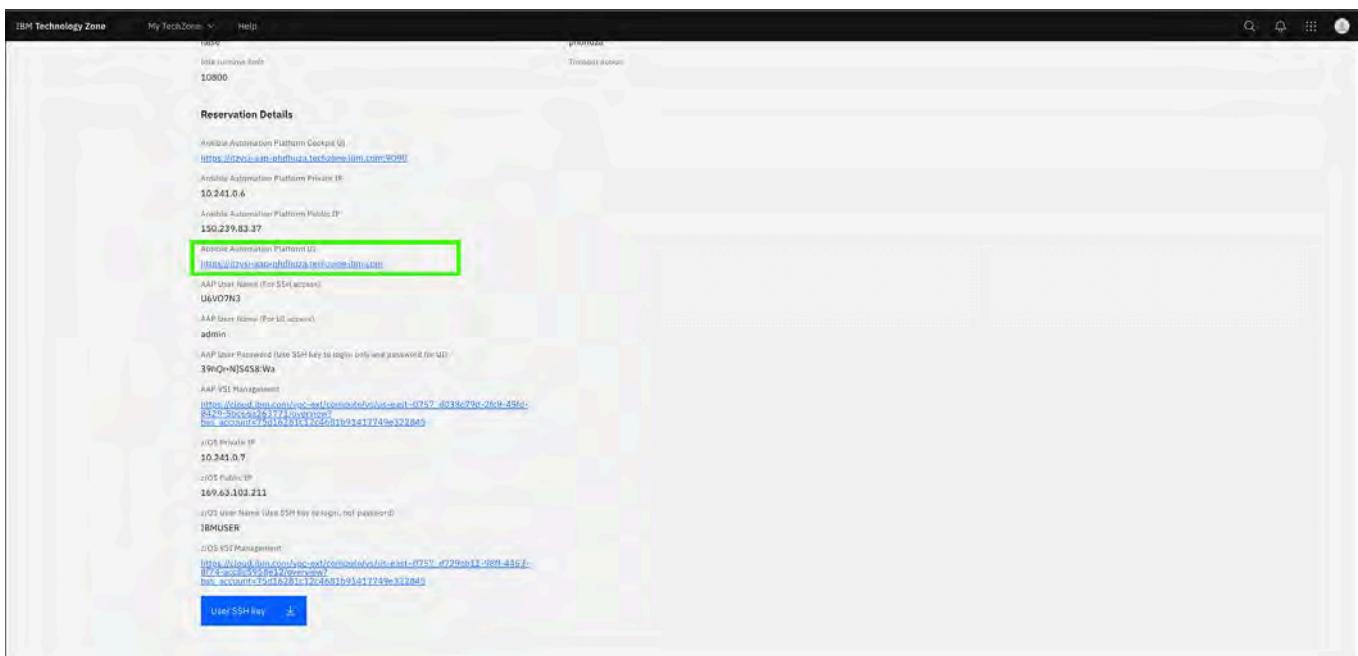
2. Click the **watsonx Assistant for Z Pilot - AAP & z/OS** tile.



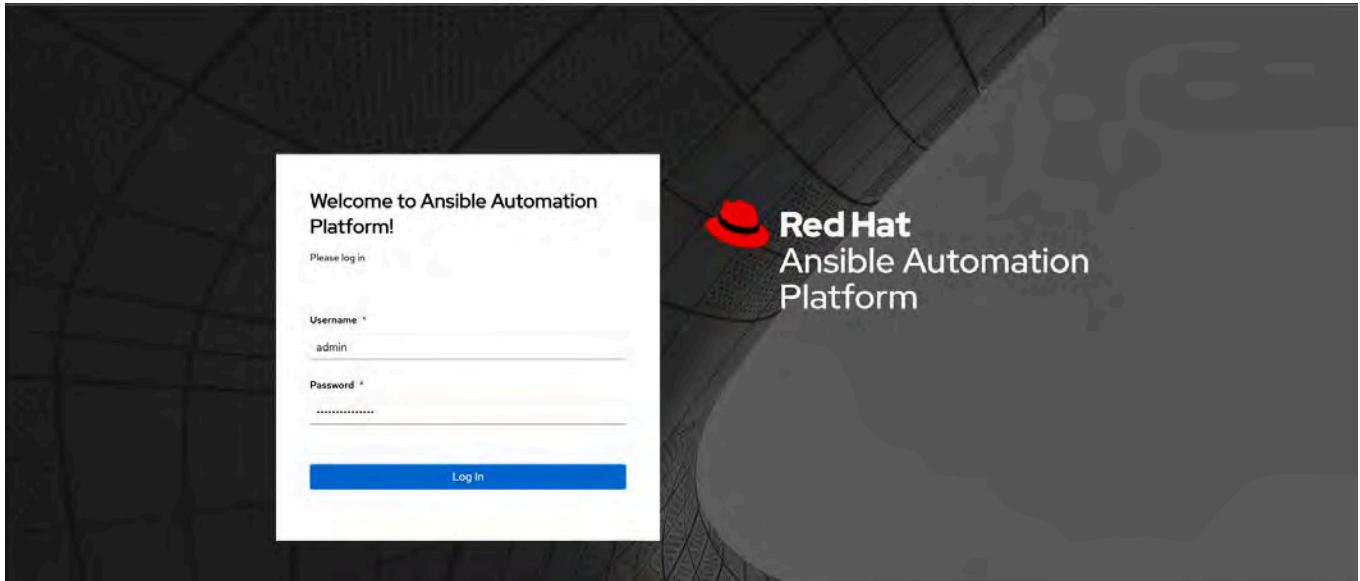
3. Locate and record the **AAP User Name (For UI access)** and **AAP User Password** fields.



#### 4. Record and then click the Ansible Automation Platform UI link.



#### 5. Enter the **Username** and **Password** that is recorded in step 3 and click **Log In**.

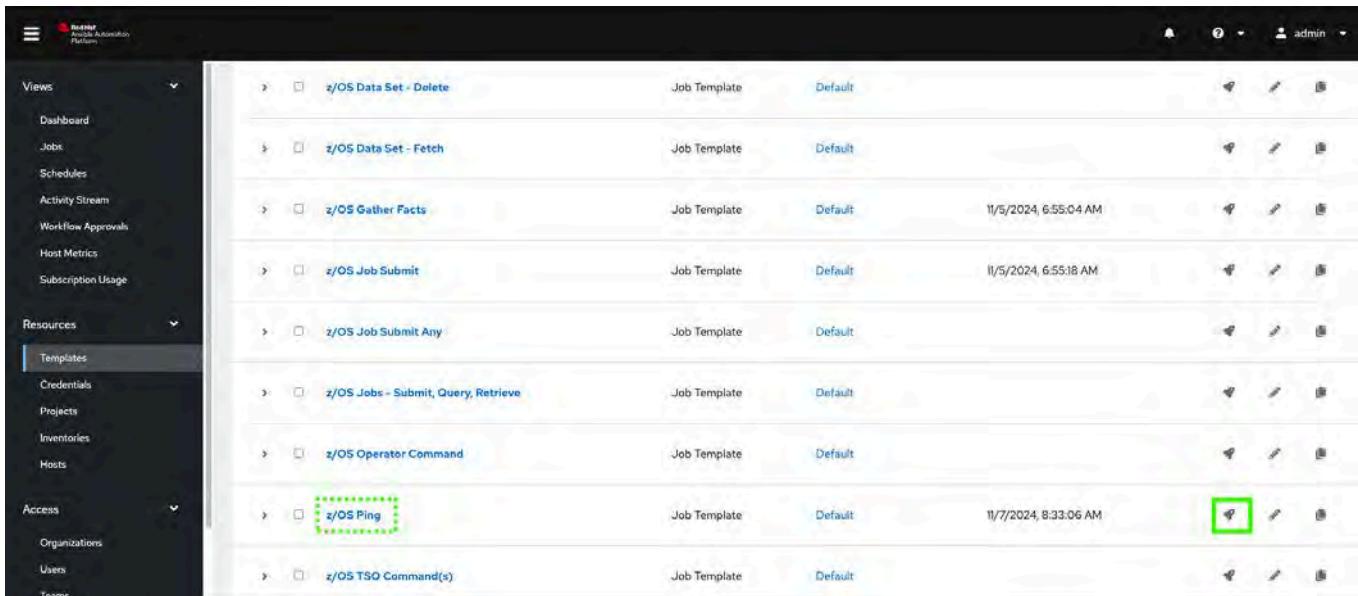


6. Click **Templates** under the **Resources** section.

**i** **The AAP instance is preconfigured to the Wazi aaS instance**

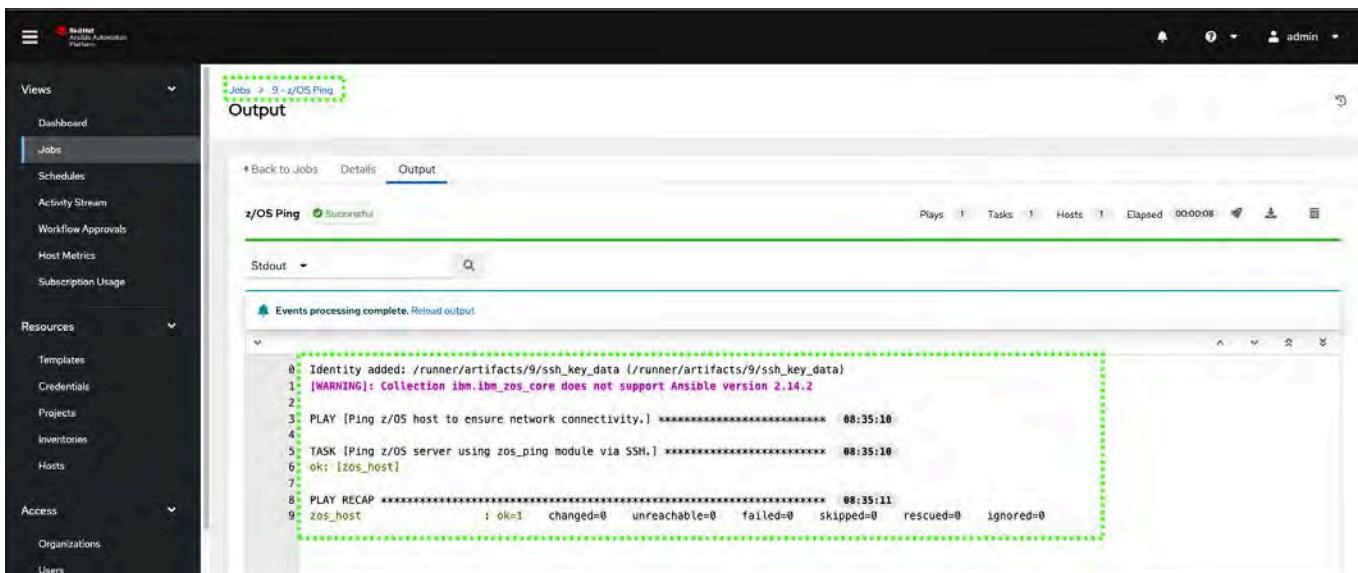
Note that because the AAP instance and the back-end z/OS system are preconfigured, no changes are needed to execute the templates and they will target your provisioned z/OS system by default.

7. Locate the **z/OS Ping** template and click the rocket (🚀) icon to start the template.



Views	<a href="#">z/OS Data Set - Delete</a>	Job Template	Default	<a href="#"></a> <a href="#"></a> <a href="#"></a>
Dashboard	<a href="#">z/OS Data Set - Fetch</a>	Job Template	Default	<a href="#"></a> <a href="#"></a> <a href="#"></a>
Jobs	<a href="#">z/OS Gather Facts</a>	Job Template	Default	11/5/2024, 6:55:04 AM <a href="#"></a> <a href="#"></a> <a href="#"></a>
Schedules	<a href="#">z/OS Job Submit</a>	Job Template	Default	11/5/2024, 6:55:18 AM <a href="#"></a> <a href="#"></a> <a href="#"></a>
Activity Stream	<a href="#">z/OS Job Submit Any</a>	Job Template	Default	<a href="#"></a> <a href="#"></a> <a href="#"></a>
Workflow Approvals	<a href="#">z/OS Jobs - Submit, Query, Retrieve</a>	Job Template	Default	<a href="#"></a> <a href="#"></a> <a href="#"></a>
Host Metrics	<a href="#">z/OS Operator Command</a>	Job Template	Default	<a href="#"></a> <a href="#"></a> <a href="#"></a>
Subscription Usage	<a href="#">z/OS Ping</a>	Job Template	Default	11/7/2024, 8:33:06 AM <a href="#"></a> <a href="#"></a> <a href="#"></a>
Resources	<a href="#">z/OS TSO Command(s)</a>	Job Template	Default	<a href="#"></a> <a href="#"></a> <a href="#"></a>
Templates				
Credentials				
Projects				
Inventories				
Hosts				
Access				
Organizations				
Users				
Teams				

## 8. Observe the z/OS Ping job run.



The screenshot shows the 'Jobs' section of the Red Hat Ansible Platform. A specific job named 'z/OS Ping' is selected, indicated by a green border around its row in the list. The 'Output' tab is active, showing the command-line output of the job's execution. The output text is as follows:

```

Events processing complete. Reload output.
=====
8 Identity added: /runner/artifacts/9/ssh_key_data /runner/artifacts/9/ssh_key_data
1 [WARNING]: Collection ibm.ibm_zos_core does not support Ansible version 2.14.2
2
3 PLAY [Ping z/OS host to ensure network connectivity.] **** 08:35:10
4
5 TASK [Ping z/OS server using zos_ping module via SSH.] **** 08:35:10
6 ok: [zos_host]
7
8 PLAY RECAP **** 08:35:11
9 zos_host : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
=====
```

Take time to explore the other templates that are ready to use. Learn more about the automation capabilities [here](#).

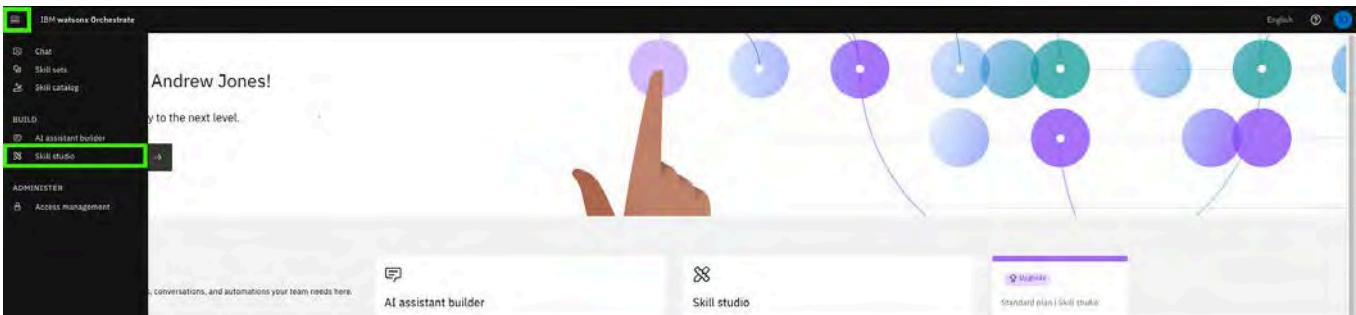
# Importing skills from Ansible Automation Platform

Now that you understand Ansible Automation Platform (AAP) and the preinstalled automations available, you can import them as skills into your watsonx Orchestrate instance, which is used for assistant guided actions.

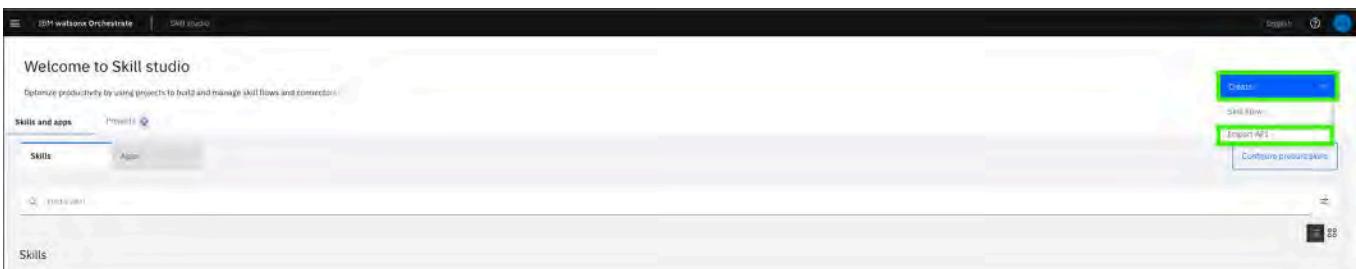
## Import skills into your assistant

The next steps assume that you have an active browser window to the watsonx Orchestrate ITZ cloud account. If you do not, refer to the initial steps in [Creating an assistant and configuring conversational search](#).

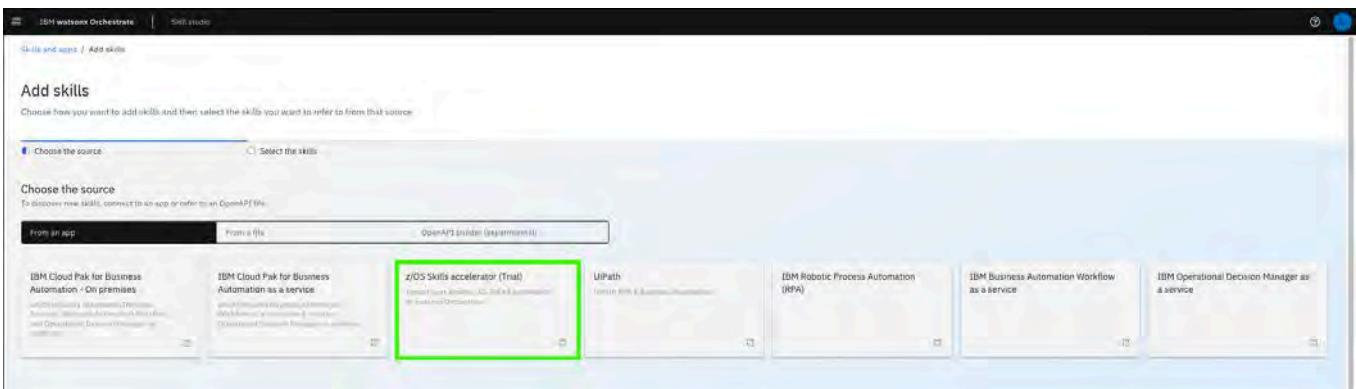
1. Return to your watsonx Orchestrate instance and expand the main menu and click **Skill studio**.



2. Expand **Create** and click **Import API**.



3. Click the **z/OS Skills accelerator (Trial)** tile.



4. Enter the following values in the **z/OS Skills accelerator** form and then click **Connect**.

Use the **URL**, **User Name**, and **Password** values recorded in the [Explore Ansible Automation Platform](#) section earlier.

**a:** Connection Type: ansible

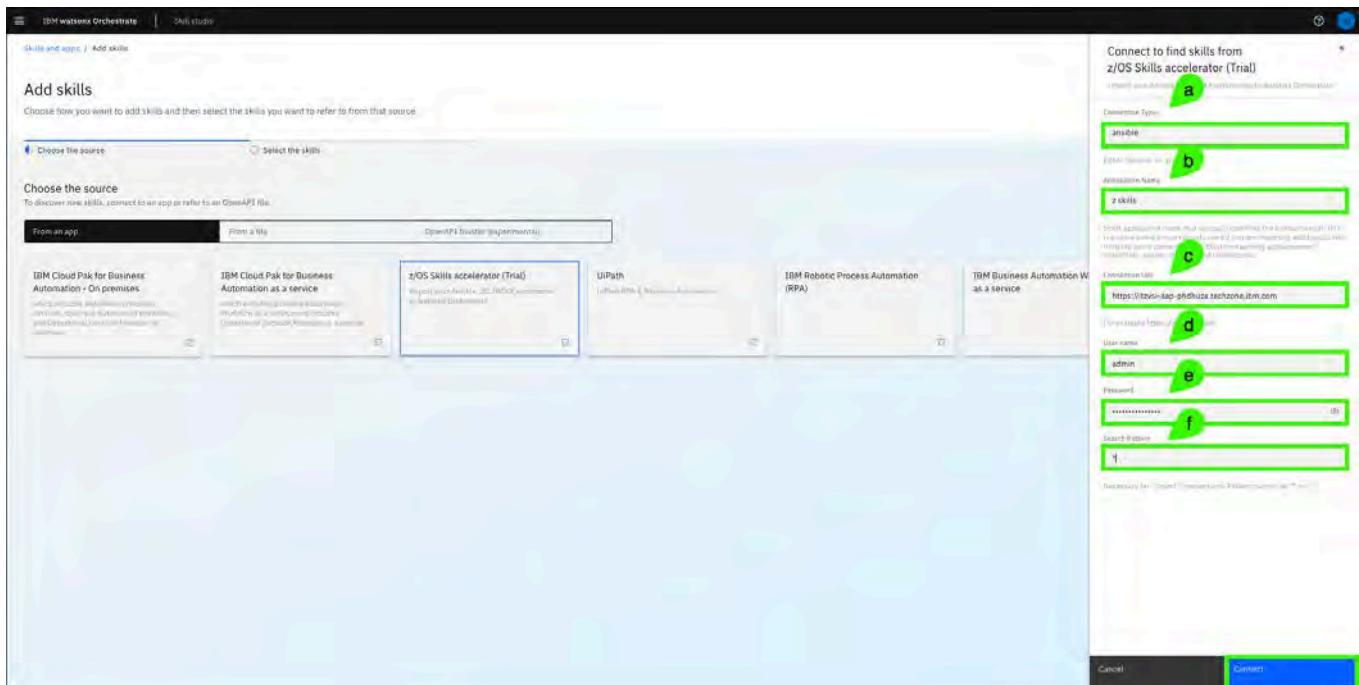
**b:** Application Name: <enter a meaningful name for the skills that you will import> - record this name, you will need in the next section

**c:** Connection URL: <enter the URL for your AAP UI>

**d:** User Name: <enter the AAP User Name (for UI access)>

**e:** Password: <enter the AAP User Password>

**f:** Search Pattern: \*

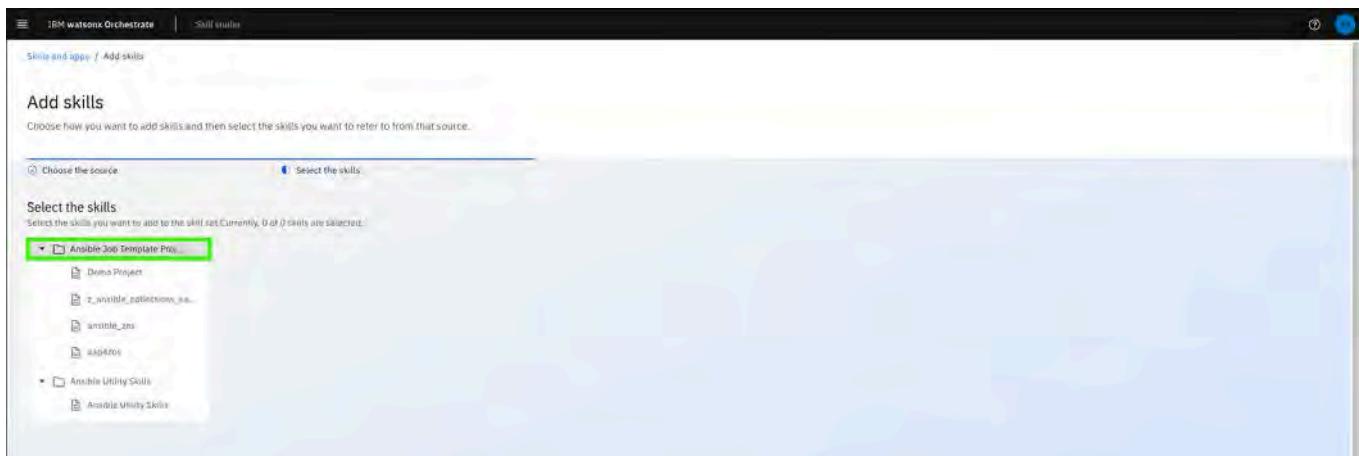


## 5. Expand the Ansible Job Template Proj... folder.



### Explore the other available skills

Take time to explore the available skills. There are many utility skills provided out of the box with the "Z Skills Accelerator" which are needed for actions such as retrieving the output of an Ansible skill. Consider importing these utility skills to enable more complete automation execution flows.



## 6. Click **aap4zos**.

Skill	Description	Status
Z/os ping	Z skills - this playbook pings...	Ready to add
Z/os gather facts	Z skills - this sample playbook...	Ready to add
Z/os job submit	Z skills - this playbook shows...	Ready to add
Z/os job submit any	Z skills - a playbook for subm...	Ready to add
Z/os jobs - submit, query, rel...	Z skills - this sample playbook...	Ready to add
Z/os users - add	Z skills - this playbook shows...	Ready to add
Z/os data set - basics	Z skills - this sample playbook...	Ready to add
Z/os data set - create	Z skills - this playbook creat...	Ready to add
Z/os data set - delete	Z skills - this playbook delet...	Ready to add
Z/os data set - fetch	Z skills - this playbook fetch...	Ready to add
Z/os certs - health checker an...	Z skills - this playbook enabl...	Ready to add
Z/os certs - create keyring	Z skills - this playbook creat...	Ready to add
Z/os certs - delete keyring	Z skills - this playbook delet...	Ready to add
Z/os certs - create cert	Z skills - this playbook creat...	Ready to add
Z/os certs - delete cert	Z skills - this playbook delet...	Ready to add

## 7. Select the skills that you want to import into your application and then click **Save as draft**.

For this lab, select the **Z/os ping** and **Z/os gather facts** skills.

## 8. Click the ellipses (⋮) for the first skill and select **Enhance this skill**.

Welcome to Skill studio

Skills and apps Projects

Skills

Name	Step in the process	Status	Skill type	Author	Last edited
z/OS Gather Facts	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	November 19, 2024
z/OS Ping	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	November 19, 2024

Configure prebuild skills

## 9. Review the skill enhancement options and then click **Publish**.

On the **Enhance this skill** page, you can specify enhancements to the default skill. Refer to this documentation for more information on enhancing skills.

Enhance the "z/OS Gather Facts" skill

Name and describe this skill in a way that tells users how it's used and why they would want to use it...

Name: z/OS Gather Facts

Description: z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS-specific information from the z/OS host.

API version: 1.0

Category: z/OS Categories

AAC: Assume Controller Access Available

Preview

The skill will look like this in the catalog.

**z/OS Gather Facts**  
A skill - This sample playbook demonstrates the z/OS gather facts.

The skill will look like this in the skill set.

**z/OS Gather Facts**

(Cancel) Publish Save as draft

## 10. Repeat steps 8 and 9 for each skill you imported.

Welcome to Skill studio

Skills and apps Projects

Skills

Name	Step in the process	Status	Skill type	Author	Last edited
z/OS Gather Facts	Ready to use	Published	Imported	andrew@jones-tx.com	November 19, 2024
z/OS Ping	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	November 19, 2024

Configure prebuild skills

The selected skills are now ready for use and available to your assistant. In the next section, learn how to connect them to your assistant.

# Connecting skills to your assistant

Once you have a subset of skills that are published, the application you created can be connected to your assistant.

1. Expand the main menu and select **Skill sets**.

The screenshot shows the 'Skill studio' section of the IBM Watsonx Orchestrate interface. On the left, a sidebar lists 'Chat', 'Skill sets' (which is selected and highlighted in green), and 'Skill catalog'. Under 'BUILD', there's an 'AI assistant builder' and 'Skill studio'. Under 'ADMINISTER', there's 'Access management'. The main area is titled 'studio' and contains a sub-header 'Skills to build and manage skill flows and connectors.' A 'Create' button is at the top right. Below is a table listing three skills:

Step in the process	Status	Skill type	Author	Last edited
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024

2. Click the **Team Skills** drop-down and select the **Draft** of your assistant.

The screenshot shows the 'Team Skills' page. At the top, a dropdown menu is open under 'Skill sets', showing 'Team Skills' (selected and highlighted in green), 'Team Skills', 'Power draft' (highlighted in green), and 'Zeeves team'. Below the dropdown, it says 'CREATE A TEAM: These skills complete related tasks for the team members so members can work more efficiently and effectively.' There's a 'Manage skills' button at the bottom right. The main area shows a team named 'Zeeves draft' with a placeholder message 'No skills' and a note to 'Click Manage skills to give this team some specific abilities.'

3. Click the **Connections** tab.

The screenshot shows the 'Connections' tab for the 'Zeeves draft' team. The top navigation bar has 'Skill sets' and 'Connections' (highlighted in green). The main area shows a table with one row: 'Skills (0)'. It includes a note: 'The skills added to this team are available to its team's members. These skills complete related tasks for the team members so members can work more efficiently and effectively.' Below the table is a 'Manage skills' button.

4. Click the **Search (🔍)** icon.

Application	Number of skills	Credential type	Connected by	Action
Activite or deactivite attracting candidates using ThisWay (Cloud)	4	Not specified		
Adobe Workfront	17	Not specified		
Alliance Virtual Office	2	Not specified		
Amazon S3	8	Not specified		
Amazon SES	10	Not specified		

5. Search for the application name you specified in the previous section.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z-skills	2	Not specified		

6. Click the ellipses (⋮) and click **Connect app**.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z-skills	2	Not specified		<b>⋮ Connect app</b>

7. On the **Connect to Ansible Controller Skills** form, keep the defaults and click **Connect app**.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with 'Skill sets' and 'Applications'. Under 'Skill sets', 'Zeeves draft' is selected. In the main area, there's a table with columns 'Application', 'Number of skills', and 'Credential type'. One row in the table is highlighted with a green border. On the right, a modal window titled 'Connect to Ansible Controller Skills - z skills' is open. It has sections for 'Lifetime' (set to 'Team'), 'Usernames' (containing 'admin'), 'Passwords' (containing 'password'), and a note about using legacy credentials. At the bottom of the modal is a blue 'Connect app' button, which is also highlighted with a green arrow.

8. Enter the **username (a)** and **password (b)** using the username ( admin ) and password for your IBM Technology Zone (ITZ) watsonx Assistant for Z Pilot - AAP & z/OS reservation, and then click **Connect app (c)**.

This screenshot shows the same connection dialog as the previous one, but with different fields highlighted with green arrows. The 'Usernames' field contains 'admin' and the 'Passwords' field contains 'password'. The blue 'Connect app' button at the bottom is also highlighted with a green arrow.

The application is now connected to the draft version of your assistant.

This screenshot shows the 'Zeeves draft' skill set page again. The 'Applications' section lists 'Ansible Controller Skills - z skills' with a green dotted box around it. Below the table, there's a note about enabling skills for personal or team credentials. At the bottom, there's a table with columns 'Action', 'Connected by', 'Credential type', 'Number of skills', and 'Application'. The first row in this table is also highlighted with a green dotted box. The 'Connected by' column shows 'andrew@jones-tx.com' and the 'Credential type' column shows 'XG\_Team'.

Continue to the next section to create actions for your assistant.

# Creating actions for your assistant

Now that the skills in your application are connected to your assistant, you are ready to create actions that are tied to those skills. Learn more about building actions [here](#)

## Configure the number of input fields

Before configuring actions, it is important to modify a setting within watsonx Orchestrate that allows triggered skills to display as forms (versus conversational skills).

1. Click your (a) profile icon and then click (b) **Settings**

Learn more about configuring input fields [here](#).

2. Click the **Skill configurations** tab.

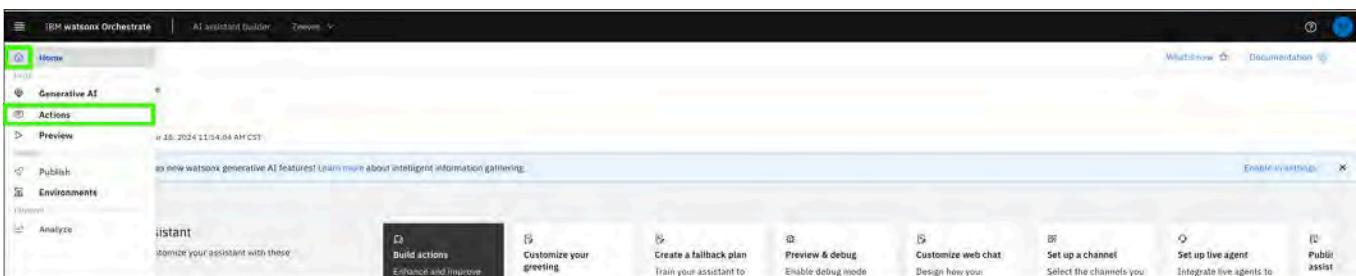
3. Enter 0 for the **Number of form fields**.

## Create actions

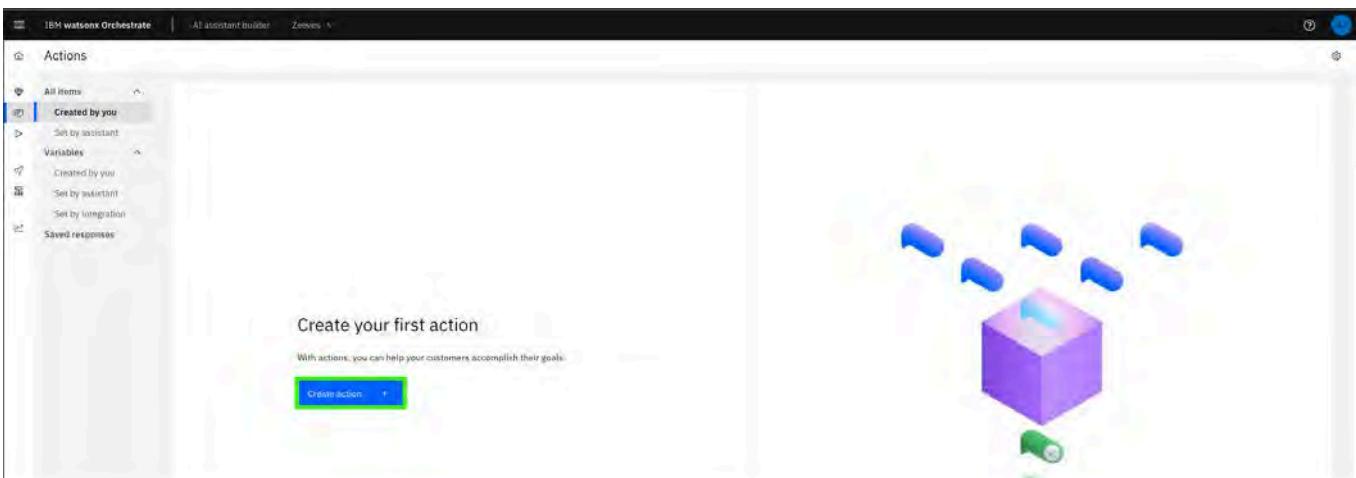
1. Click the main menu and select **AI assistant builder**.



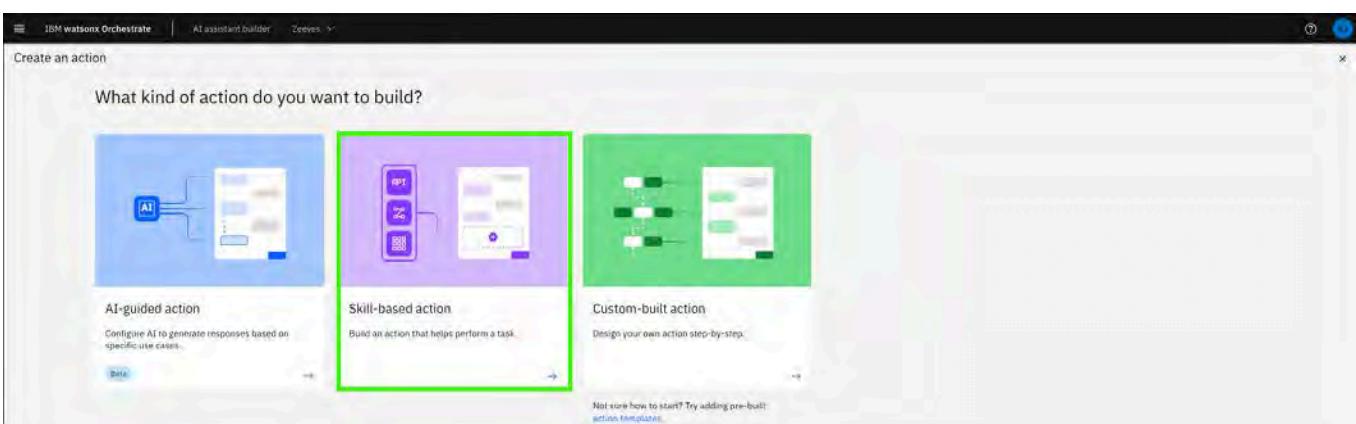
2. Hover over the **Home** icon ( ) and click **Actions**.



3. Click **Create action**.



4. Click the **Skill-based action** tile.



5. Select the **z/OS Gather Facts** tile and click **Next**.

Note, it may take a minute for the page to display the action tiles. The date that is shown in the **z/OS Gather Facts** tile reflects when you added the skill to your application.

Build an action from a skill

Select a skill

Choose a conversational skill published as a foundation of your action.

<b>z/OS Ping</b> This playbook ping the z/OS host to test connectivity. Last updated: 2024-11-19T10:49:29.967Z	<b>z/OS Gather Facts</b> This sample playbook demonstrates the z/OS gather Facts module, which pulls z/OS-specific information from the z/OS host. Last updated: 2024-11-19T10:49:26.643Z	<b>Summarize the Webex meeting transcript</b> io.watson.ai Last updated: 2024-11-04T10:49:16.602Z	<b>Summarize the Box content</b> io.WatsonAI Last updated: 2024-11-04T10:49:12.077Z	<b>Summarize a Zendesk ticket</b> io.WATSONX Last updated: 2024-11-04T10:49:09.496Z
<b>Summarize a ServiceNow incident</b> in.watson.ai Last updated: 2024-11-04T10:49:05.838Z	<b>Summarize a Salesforce opportunity</b> in.watson.ai Last updated: 2024-11-04T10:49:01.769Z	<b>Sharepoint document summary</b> in.watson.ai Last updated: 2024-11-04T10:48:55.707Z	<b>Salesforce email summary</b> in.watson.ai Last updated: 2024-11-04T10:48:55.522Z	<b>Salesforce case summarization</b> in.WATSONX Last updated: 2024-11-04T10:48:48.155Z
<b>Salesforce case sentiment analyze</b>	<b>Outlook email summary</b>	<b>Github issue summarization</b>	<b>Github issue sentiment</b>	<b>Generate an email</b>

- On the **New action** dialog, (a) enter a prompt a user of the assistant might use to initiate the action and then (b) click **Save**.

Sample prompts:

Get z/OS facts

Gather z/OS facts

**z/OS Gather Facts**

**Add example phrases:**

Enter phrases that a customer might use to start this action. These phrases determine the task, decisions, or workflow your customer has.

For each phrase you enter, the system will automatically recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 0

Example: Trace of trouble / problem / help etc.

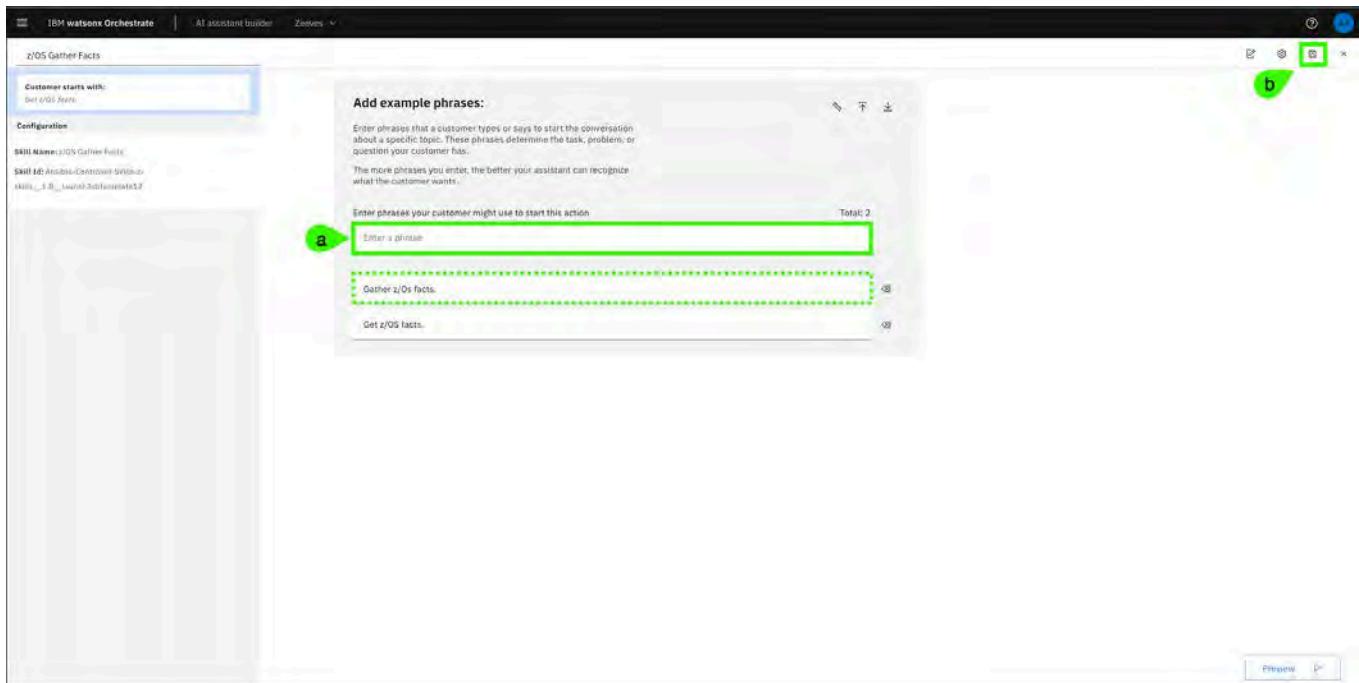
**New action**

What does your customer say to start this interaction?

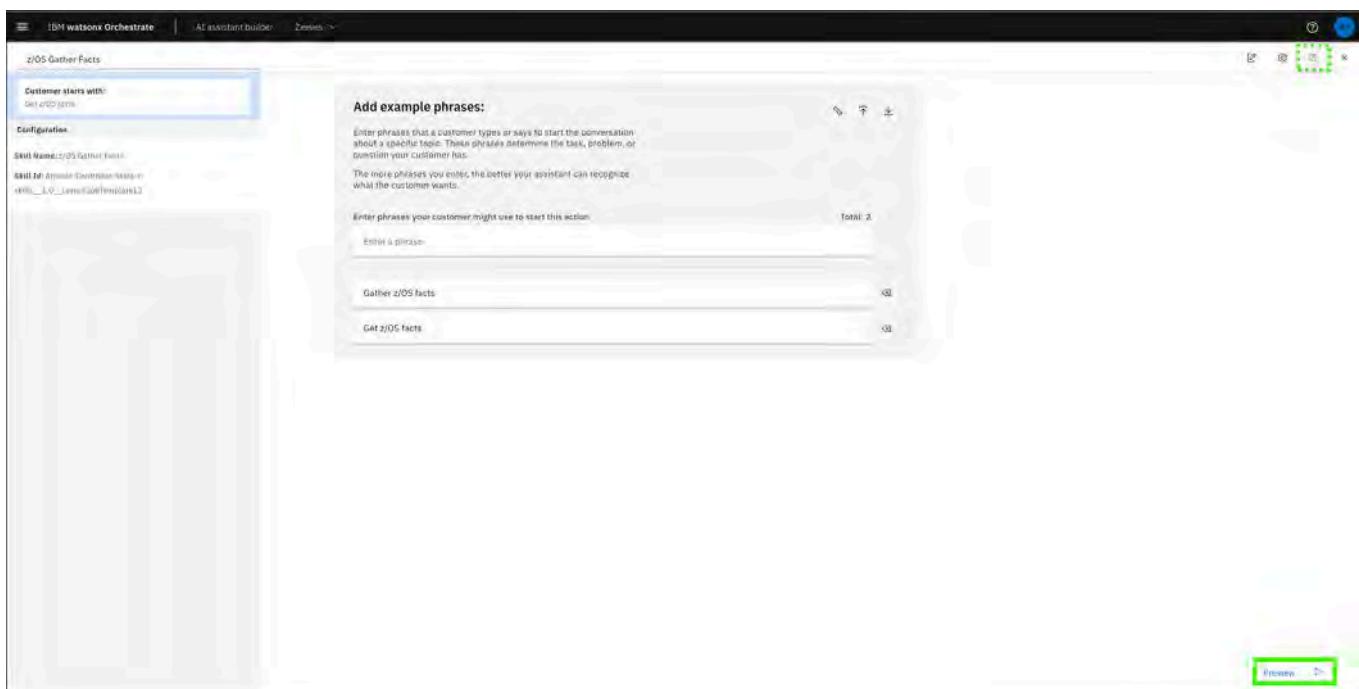
Get z/OS facts.

Cancel Save

- Add any extra prompts (a) and then click the save (b) (b).



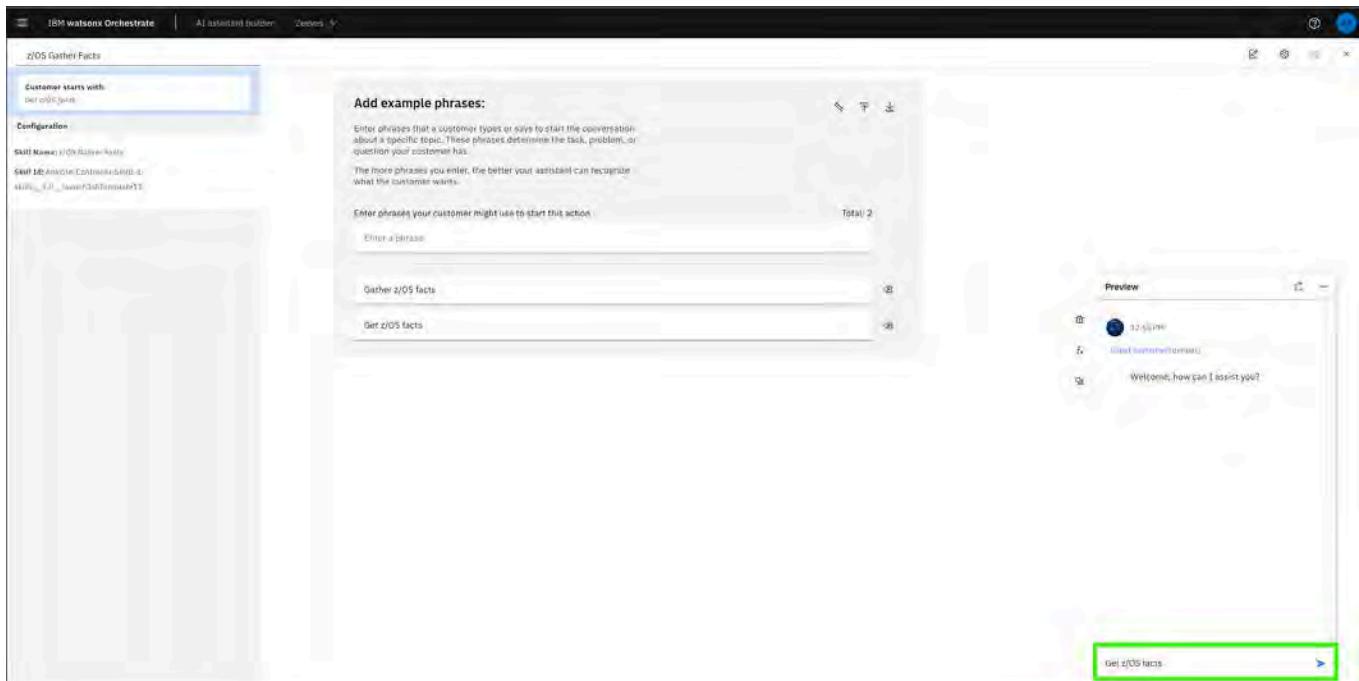
#### 8. Click Preview.



#### 9. Enter one of the prompts you specified in step 9 or 10.

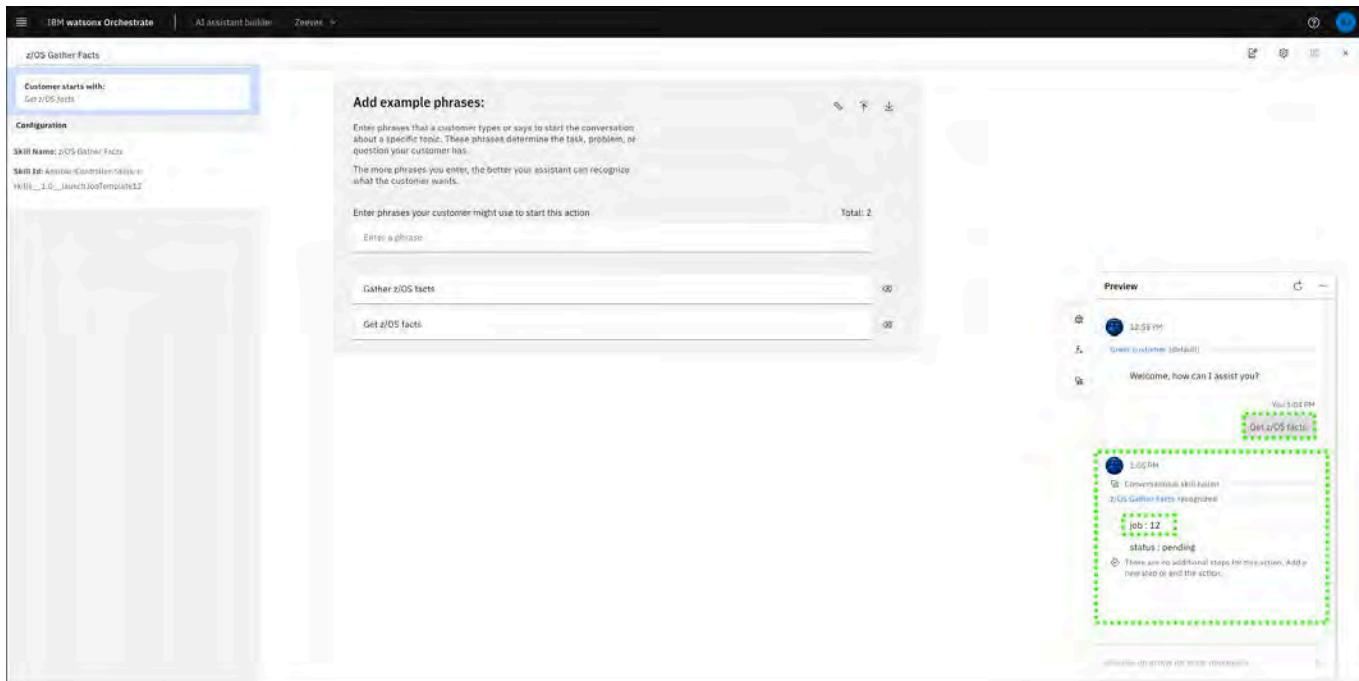
Prompt:

Get z/OS facts



## 10. Review the returned results and record the job number.

**⚠ If an error is generated or the action is not performed and only search results are returned, review the following Troubleshooting section.**



# Verify the job in the Ansible Automation Platform console

Return to the Ansible Automation Platform (AAP) console and review the job information.

1. Click **Jobs** and then click the **job** number that you recorded in the previous step for the **z/OS Gather Facts** skill.

Name	Status	Type	StartTime	Finish Time	Actions
12 - z/OS Gather Facts	Successful	Playbook Run	12/5/2024, 7:17:32 AM	12/5/2024, 7:17:41 AM	

2. Review both the **Details** and **Output** for the **z/OS Gather Facts** job.

Recall, that in the assistant, the contents shown in the **Output** of the Ansible job were not displayed.

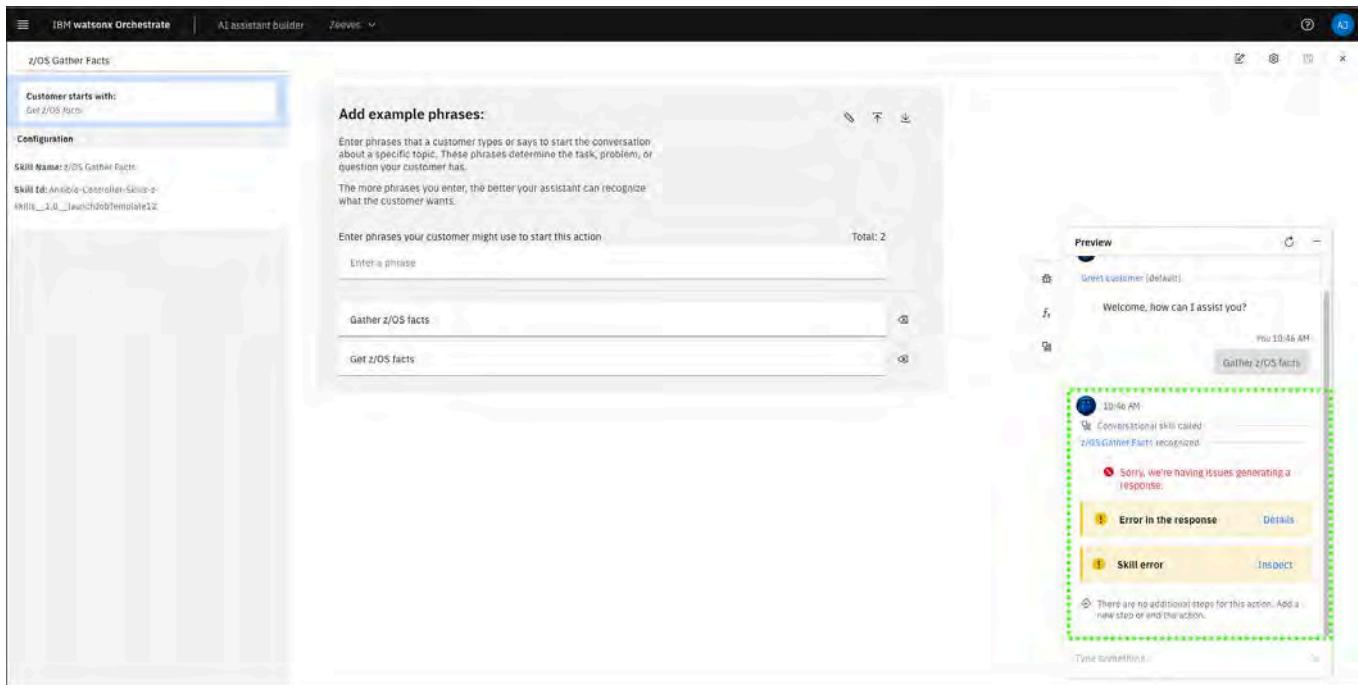
```

25 }
26
27 TASK [Print out all gathered facts about the z/OS host.] **** 07:17:38
28 ok: [zos_host] => {
29     "ansible_facts": {
30         "arch_level": "2",
31         "cpc_nd_manufacturer": "IBM",
32         "cpc_nd_model": "A08",
33         "cpc_nd_plant": "C1",
34         "cpc_nd_seqno": "237701828347",
35         "cpc_nd_type": "008562",
36         "edit": "00",
37         "hw_name": "",
38         "leasymp_card": "(00,K2)",
39         "io_config_id": "00",
40         "iodate": "",
41         "iodesc": "",
42         "iodf_config": "DEFAULT",
43         "iodf_name": "PROV.IODF00",
44         "iodf_unit_addr": "DE28",
45         "ioproc": "",
46         "iotime": ""
47     }
48 }
```

IBM watsonx Assistant for Z provides utility skills to retrieve the job output. It is also possible to create a skill flow that executes the **z/OS Gather Facts** skill followed by the **Retrieve job output** utility skill in sequence; passing the job ID from the first skill to the second to view the output within the assistant. Creating a skill flow is covered in the next section.

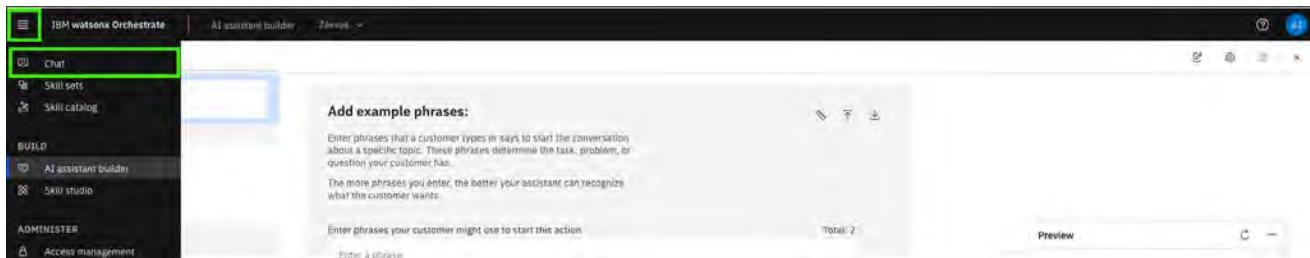
## Troubleshooting

X Skill returns "Sorry, we're having issues generating a response" or the action is not performed and only search results are returned. ✖

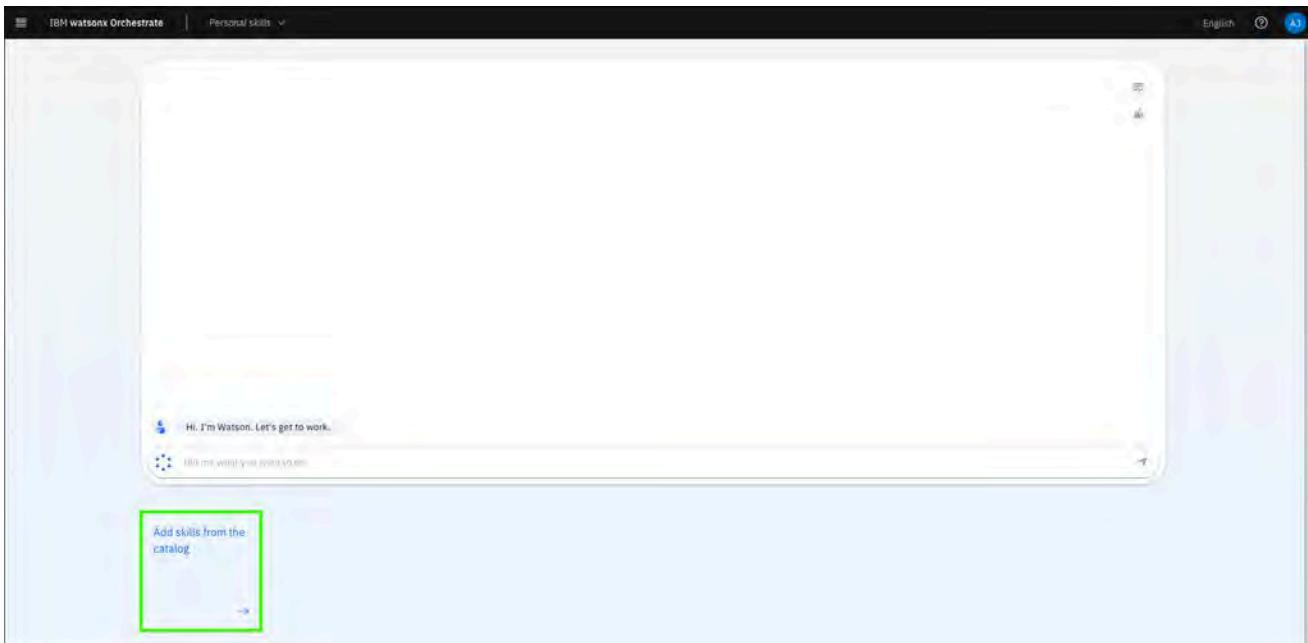


This error appears to be an intermittent issue when a skill is first added. To resolve, add the skill to your personal skills catalog using the steps that follow. If you encounter the issue, try the steps that follow:

1. Expand the main menu and select Chat.



2. Click Add skills from the catalog.



3. Search for the skill app you created earlier and click the tile for your app.

The screenshot shows the 'Skill catalog' page in IBM Watsonx Orchestrator. A search bar at the top contains the text 'z skills'. A green box highlights the 'Ansible Controller Skills - z skills' app tile, which has 2 skills. Other visible apps include Salesloft, Cognos, Skill flows, Adobe Workfront, Webex, IBM Process Mining, Salesforce Chatter, Interview top candidates using ..., Reveal your existing applicants..., Seismic, FreshService, Workday HCM, and ZoomInfo.

4. Click Add skill for all the skills you want to add.

The screenshot shows the 'Ansible Controller Skills - z skills' page in IBM Watsonx Orchestrator. It displays two skill cards: 'z/OS Gather Facts' and 'z/OS Ping'. Each card has an 'Add skill' button, which is highlighted with a green box. There are also 'Connect app' and 'Personal skills' buttons at the top right.

5. Click Connect app.

The screenshot shows the 'Ansible Controller Skills - z skills' section of the IBM Watsonx Orchestrate interface. It displays two skill cards: 'z/OS Gather Facts' and 'z/OS Ping'. Both cards have a green 'Added' button at the bottom right. In the top right corner of the main content area, there is a blue 'Connect app' button, which is also highlighted with a green box.

6. Enter the (a) **username** and (b) **password** using the username (admin) and password for your IBM Technology Zone (ITZ) watsonx Assistant for Z Pilot - AAP & z/OS reservation (AAP User Password (Use SSH key to login, only use password for UI)), and then click **Connect app**.

A modal dialog box titled 'Connect to Ansible Controller Skills - z skills' is displayed. It contains two input fields: 'username' with the value 'admin' and 'password' with a masked value. At the bottom right of the dialog is a blue 'Connect app' button, which is highlighted with a green box.

7. Expand the main menu and select **Chat**.

The screenshot shows the main navigation menu of IBM Watsonx Orchestrate. The 'Chat' option under the 'Skills' section is highlighted with a green box. The menu also includes 'Skill sets', 'Skill catalog', 'BUILD' (with 'AI assistant builder' and 'Skill studio'), and 'ADMINISTER' (with 'Access management'). The main content area shows the 'Skills - z skills' section with the 'z/OS Ping' skill card visible.

8. Try one of the prompts you created for your skill.

Prompt:

Gather z/OS facts

The screenshot shows the IBM Watsonx Assistant interface. At the top, it says "IBM Watsonx Orchestrate" and "Personal skills". On the right, there are language and user icons. The main area has a message from Watson: "Hi, I'm Watson. Check out the skills in the catalog to see how I can help you." Below this, a button labeled "Gather z/OS facts" is highlighted with a green border. A message follows: "Here are the results of the z/OS Gather Facts skill." It lists "z/OS Gather Facts" under "status" and "pending" under "io0". There is also a small number "12". At the bottom, a blue bar says "Tell me what you want to do". In the bottom left, a box says "Add skills from the catalog" with an arrow pointing to another box labeled "Ansible Controller Skills - 2 skills".

You should now be able to run the skill through the assistant preview.

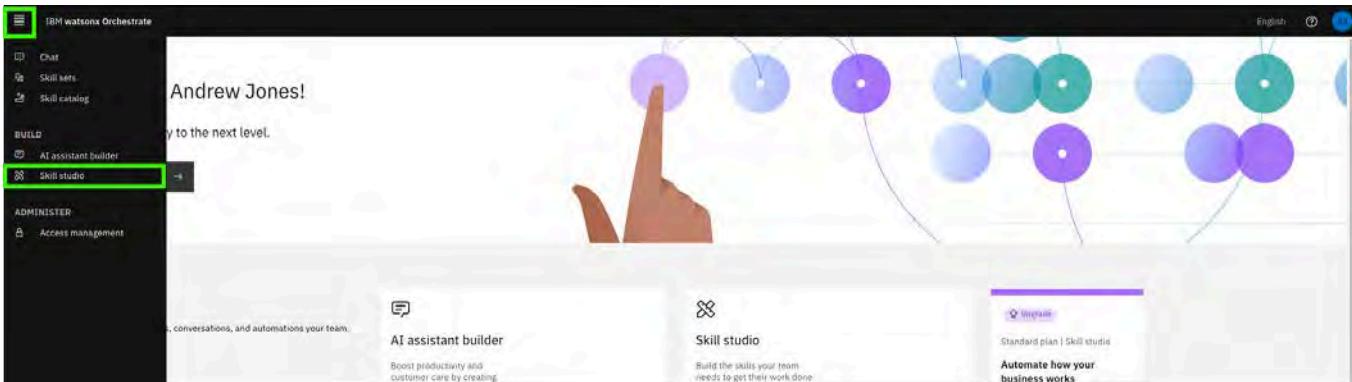
# Creating skill flows

In the previous section, you ran the **Gather z/OS facts** skill, but the output was not displayed in the assistant. To both run the action and display the results, a skill flow is needed. Skills are often more valuable when combined with other skills. You can create a skill flow to use two or more skills together to finish a task (like returning the output of a previous skill). When you create a skill flow, you map the output of one skill as the input for subsequent skills. Learn more about creating skill flows [here](#).

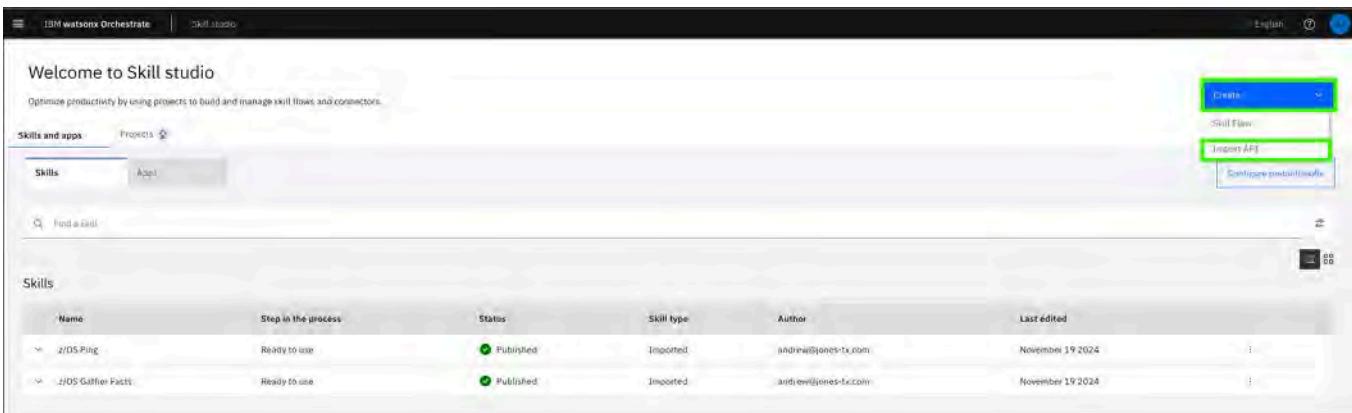
As mentioned in a previous section, default utility skills that are provided with the watsonx Assistant for Z skills collection. The **Retrieve job output** utility skill is used to return the output of a skill.

## Add the utility skill

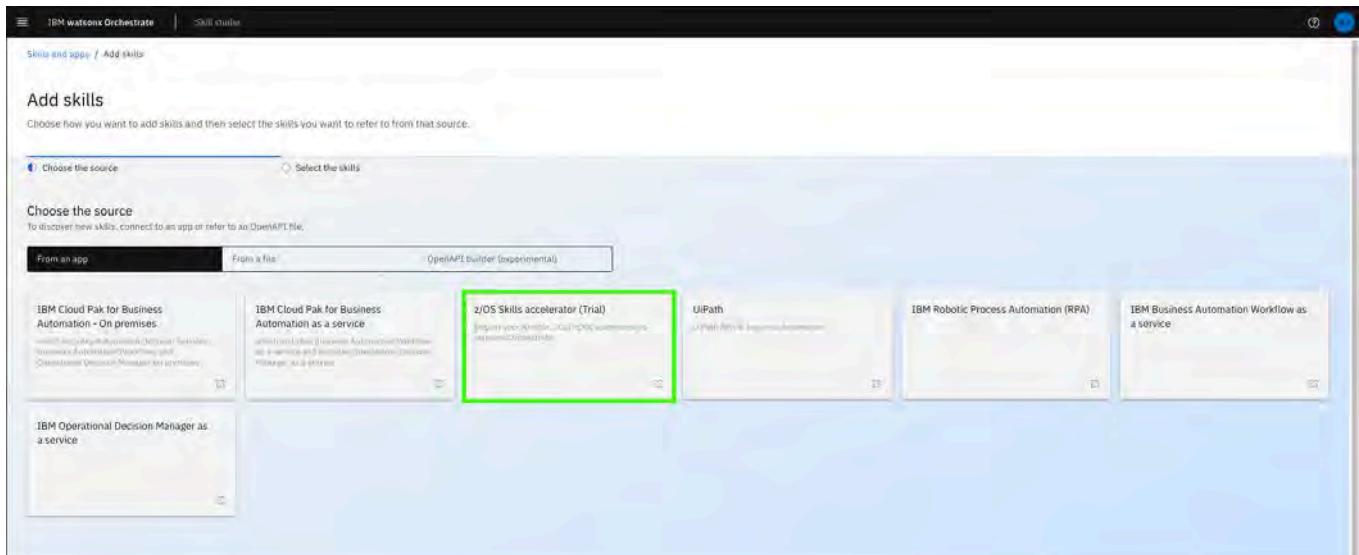
### 1. Open IBM watsonx Orchestrate **Skill studio**.



### 2. Expand **Create** and click **Import API**.



### 3. Click the **z/OS Skills accelerator (Trial)** tile.



#### 4. Enter the following values in the z/OS Skills accelerator form and then click **Connect**.

Use the **URL**, **User Name**, and **Password** values recorded in the [Explore Ansible Automation Platform](#) section earlier.

**a:** Connection Type: `ansible`

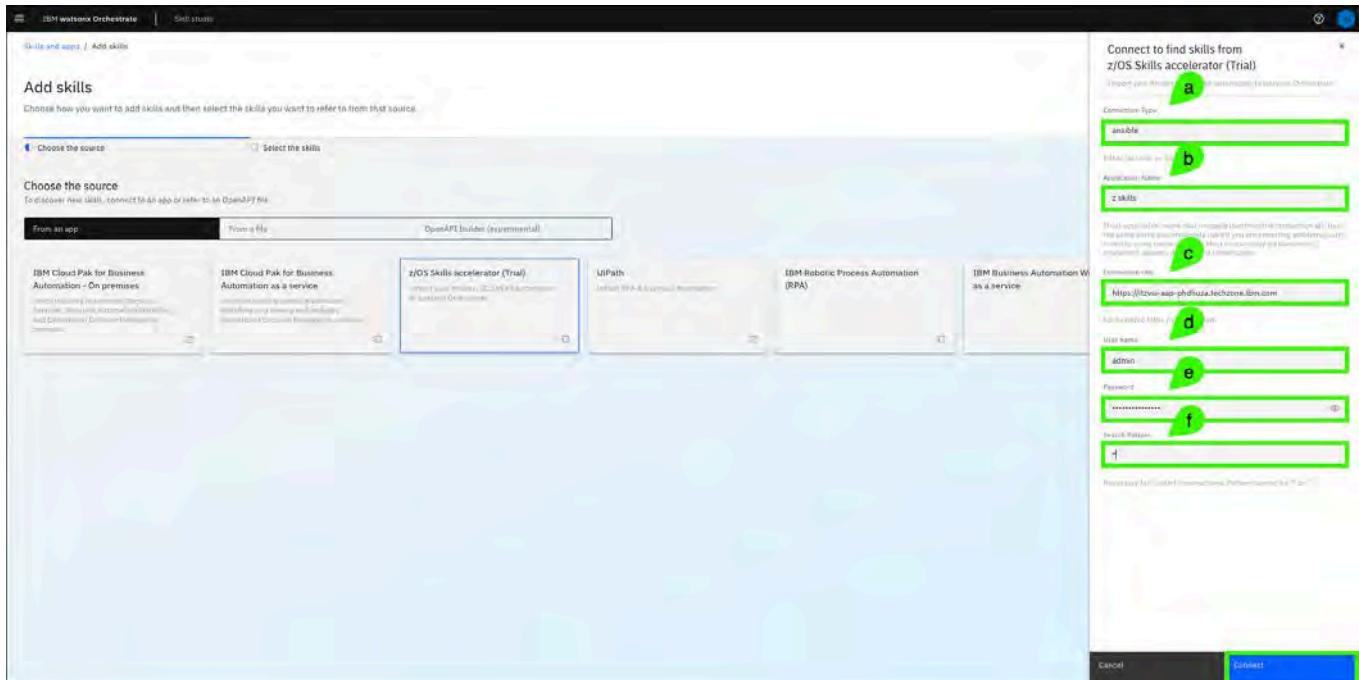
**b:** Application Name: <use the same application name from the previous section>

**c:** Connection URL: <enter the URL for your AAP UI>

**d:** User Name: <enter the AAP User Name (for UI access)>

**e:** Password: <enter the AAP User Password>

**f:** Search Pattern: `*`



#### 5. Expand **Ansible Utility Skills** and click **Ansible Utility Skills**.

**Add skills**

Choose how you want to add skills and then select the skills you want to refer to from that source.

**Select the skills**

Select the skills you want to add to the skill set. Currently, 0 of 4 skills are selected.

Skill	Description	Status
List hosts	Z skills - list hosts on this ...	Ready to add
List inventories	Z skills - list inventories on... ...	Ready to add
Retrieve job status	Z skills - retrieve job status...	Ready to add
<b>Retrieve job output</b>	Z skills - retrieve job output...	Ready to add

## 6. Select **Retrieve job output** and click **Save as draft**.

**Add skills**

Choose how you want to add skills and then select the skills you want to refer to from that source.

**Select the skills**

Select the skills you want to add to the skill set. Currently, 1 of 4 skills are selected.

Skill	Description	Status
List hosts	Z skills - list hosts on this ...	Ready to add
List inventories	Z skills - list inventories on... ...	Ready to add
Retrieve job status	Z skills - retrieve job status...	Ready to add
<b>Retrieve job output</b>	Z skills - retrieve job output...	Ready to add

**Save as draft**

## 7. Click the ellipses (...) for the **Retrieve job output** skill and select **Enhance this skill**.

**Welcome to Skill studio**

Optimize productivity by using projects to build and manage skill flows and connectors.

**Skills and apps** **Projects**

**Skills** **App**

**Skills**

Name	Step in the process	Status	Skill type	Author	Last edited
Retrieve job output	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	November 19 2024
z/OS Ping	Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
z/OS Gather Facts	Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024

**Enhance this skill**

**Export this skill**

**Delete this skill**

## 8. Review the skill settings and then click **Publish**.

## 9. Select Skill sets from the main menu.

## 10. Select (a) your draft assistant in the Team Skills drop-down list and (b) click the Connections tab.

Application	Number of skills	Credential type	Connected by	Action
Activate or deactivate interviewing candidates using TheWay Global	0	Not specified		
Adobe Workfront	27	Not specified		
Alliance Virtual Office	2	Not specified		
Amazon S3	0	Not specified		
Amazon SNS	10	Not specified		

11. Click the **Search (🔍)** icon.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	2	Not specified	zg_team	<span>⋮</span>
Adobe Workfront	17	Not specified		<span>⋮</span>
Alliance Virtual Office	2	Not specified		<span>⋮</span>
Amazon S3	8	Not specified		<span>⋮</span>
Amazon SES	10	Not specified		<span>⋮</span>

12. Search for the application name you specified earlier.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	2	Not specified	zg_team	<span>⋮</span>

13. Click the (a) ellipses (⋮) for your application and (b) click **Edit connection**.

14. Verify that the application is **Connected (a)** and then click **Close (b)**.



Connect the application if it is not connected.

Use the AAP user name (admin) and the AAP password for your ITZ reservation.

The screenshot shows the 'Connections' tab in the 'Zeeves draft' workspace. A table lists applications and their connection status to 'Ansible Controller Skills - z skills connection'. One entry, 'Ansible Controller Skills - z skills connection', is highlighted with a green border and has a callout box pointing to it. The callout box contains the text: 'You have an active connection to using Team credentials. If you wish to update connection details click on the Edit icon now.' Below this, there's a radio button labeled 'Connected' with a checked status.

Application	Number of skills	Credential type
Ansible Controller Skills - z skills connection	84 skills	Team

## Add the skills to your Personal skills

1. Click **Skill catalog** in the main menu.

The screenshot shows the main menu with 'Skill catalog' selected. The 'Skill catalog' page displays a table of skills connected to various applications. A callout box highlights the 'Connected by' column, which shows 'andrew@yahoo.com' for the 'Ansible Controller Skills - z skills connection' row.

	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills connection	84 skills	Team	andrew@yahoo.com	

2. Search for the application name you specified earlier.

The screenshot shows the 'Skill catalog' page with a search bar containing 'z skills'. The results section is titled 'Most popular skills' and shows tiles for 'Send an email from Gmail', 'Create a lead in Salesforce', and 'Send an email using G Suite'. Below this is a grid of 'All Apps' with tiles for Coupa, ZoomInfo, HubSpot CRM, Apprio Targetprocess, Salesforce, Zendesk Service, Calendly, Square, Oracle E-Business Suite, GitLab, Toggl Track, and Microsoft Teams.

3. Click the tile for your application.

Note, the tile name is proceeded by **Ansible Controller Skills**.

- Click Add skill for each of the skills you want to add to the flow.

## Create the skill flow

- Click Skill studio in the main menu.

- Expand the Create drop-down menu and click Skill flow.

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

**Skills and apps** Projects

**Skills**

Name	Step in the process	Status	Skill type	Author	Last edited
Retrieve job output	Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
z/OS Ping	Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
z/OS Gather Facts	Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024

3. Click the + icon.

Skills and apps / Create a Flow

Untitled

```

graph TD
    Start((Start)) --> Plus[+]
    Plus --> End((End))
  
```

Next, you need to add the **z/OS Gather Facts** skill and the **Retrieve job output** skill to the skill flow. Use the **Search apps** function to locate the skills.

4. Search for the application name you specified earlier and click the tile.

Skills and apps / Create a Flow

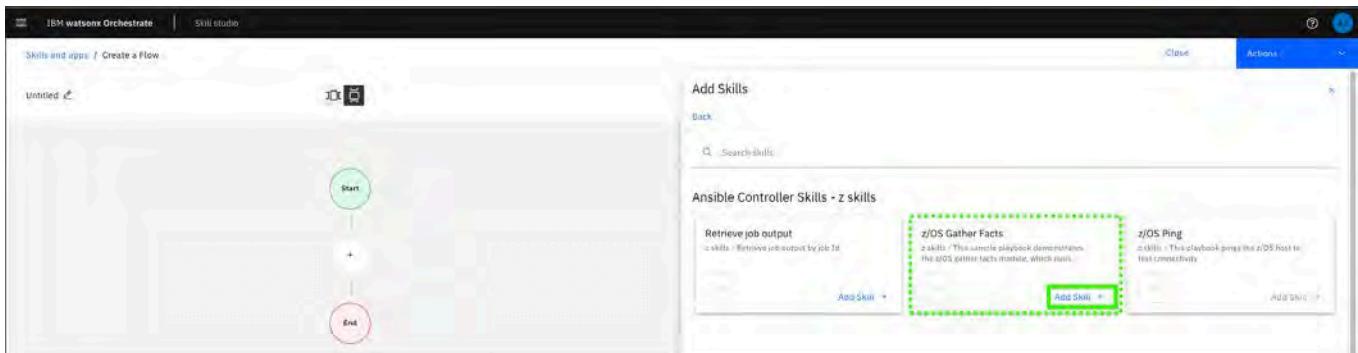
Untitled

Add Skills

z skills

ZoomInfo	Workday HCM	FreshService
Seismic	Reveal your existing applic...	Interview top candidates u...
Salesforce Chatter	IBM Process Mining	Webex
Adobe Workfront	Skill flows	An Ansible Controller Skills
Cognos	Salessoft	... 3 skills

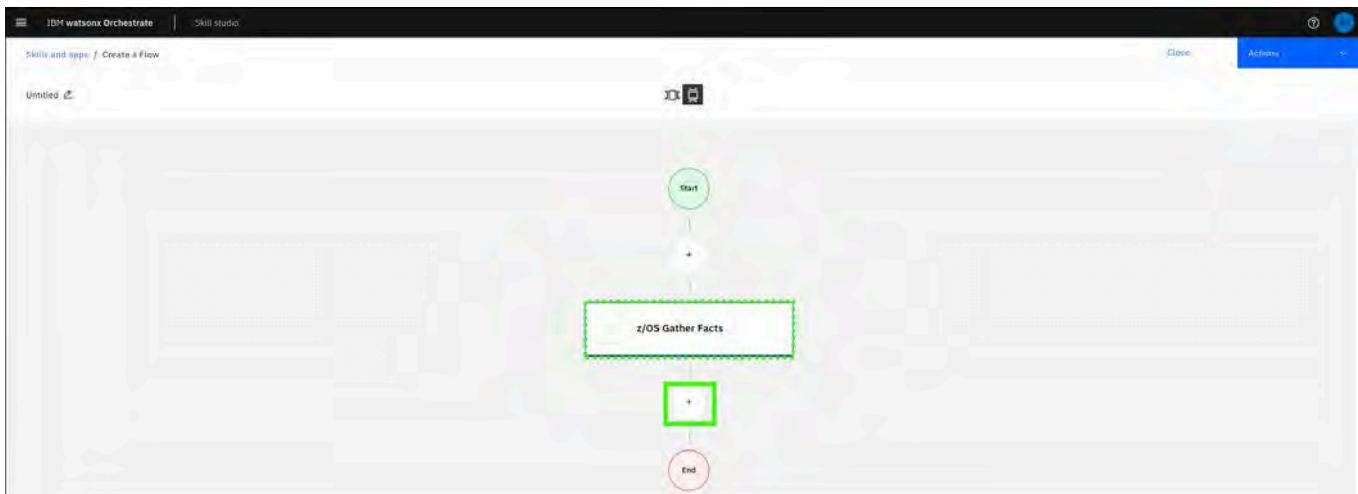
5. Click **Add Skill** in the **z/OS Gather Facts** tile.



6. Verify the **z/OS Gather Facts** skill is added to the skill flow.

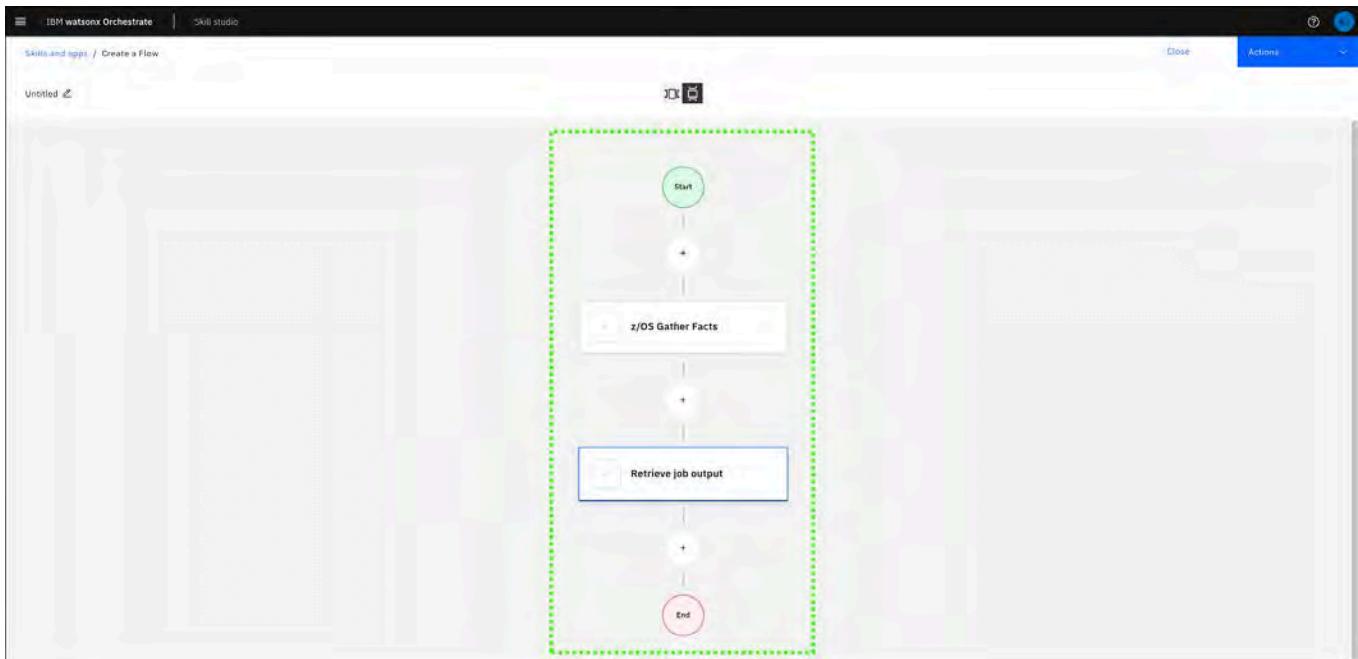


7. Click the + icon after the **z/OS Gather Facts** tile.



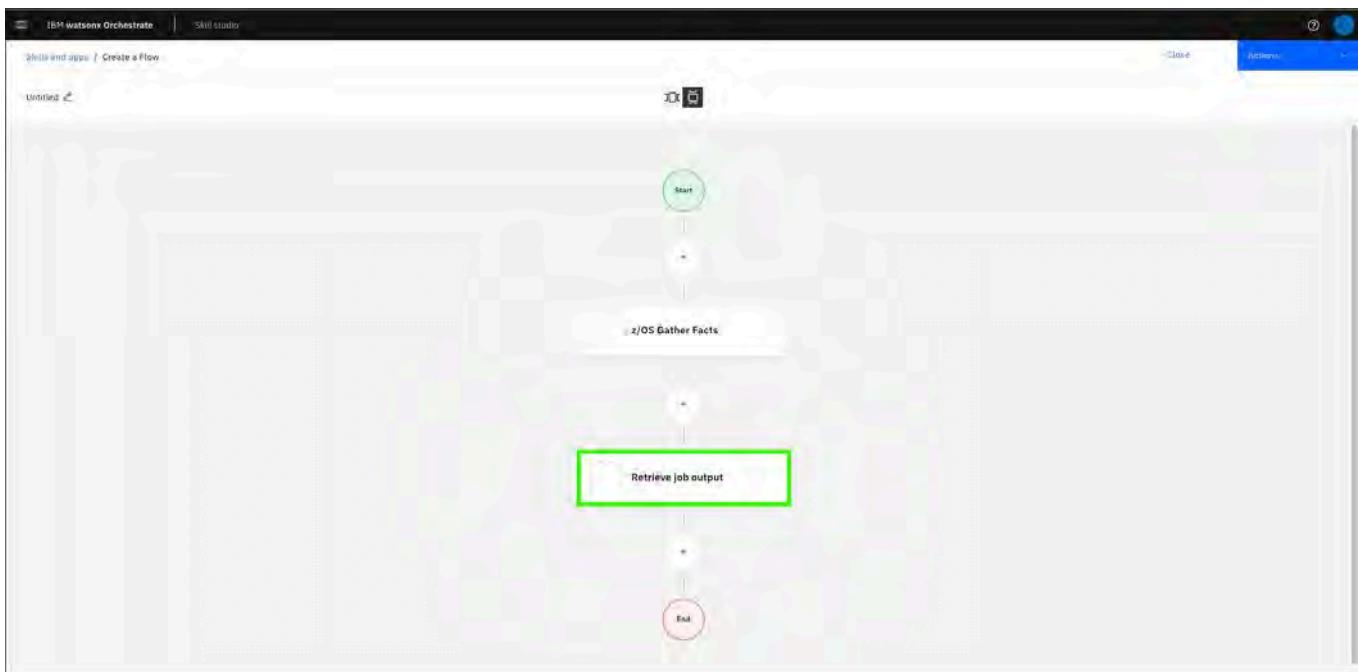
8. Repeat steps 5 and 6 for the **Retrieve job output** skill.

After adding the **Retrieve job output** skill, your skill flow should look like:

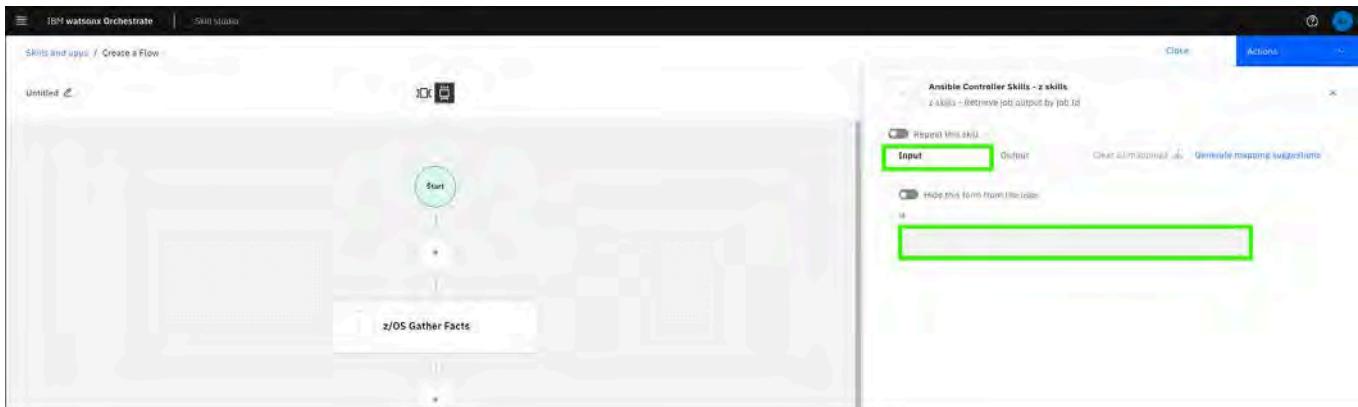


Next you must map the output values of the first skill to the input of the second skill. In this case, pass the job ID output from **z/OS Gather Facts** as an input for **Retrieve job output**.

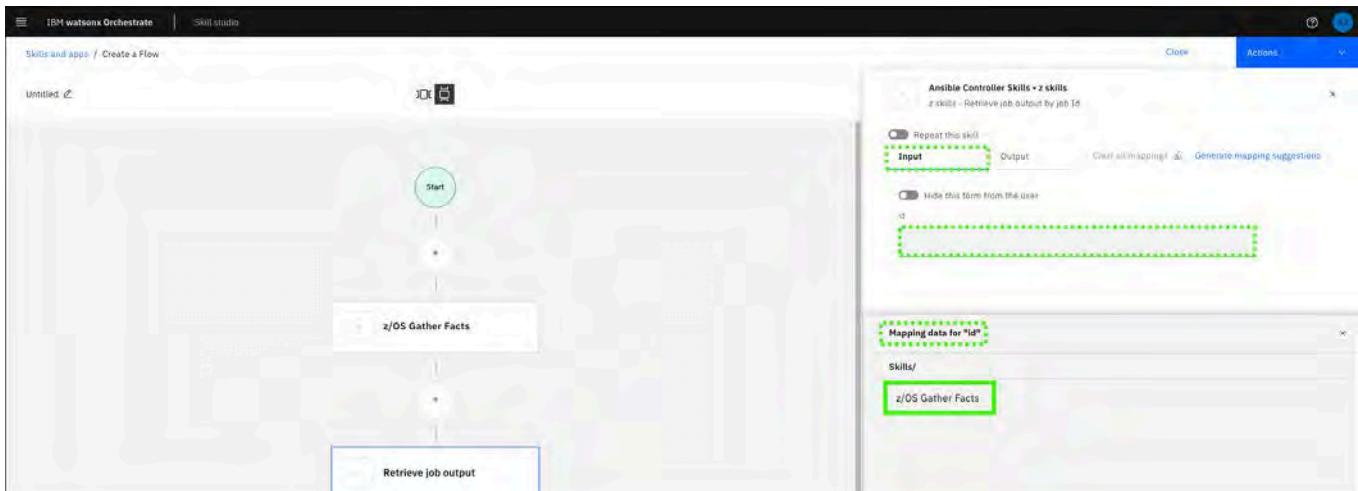
#### 9. Click the **Retrieve job output** tile.



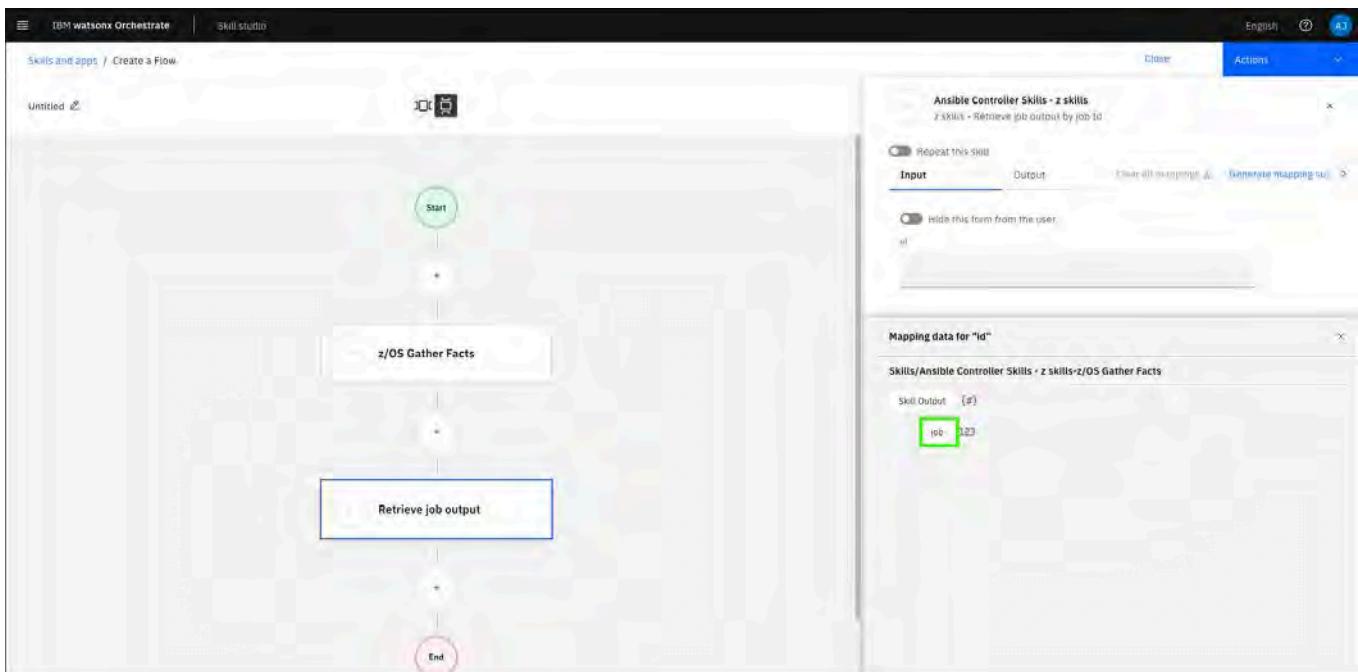
#### 10. Select the **Input** tab and click the **id** field.



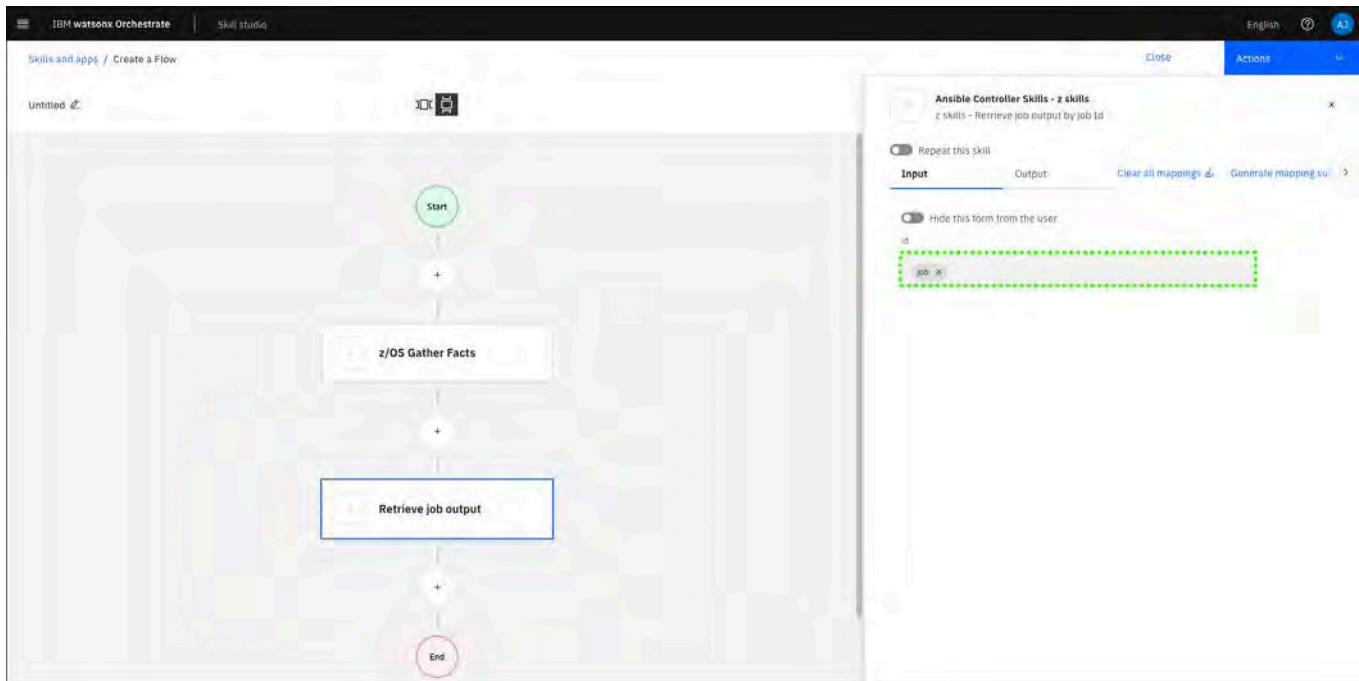
11. Click the **z/OS Gather Facts** skill in the **Mapping data for "id"** section.



12. Click the **job** icon.

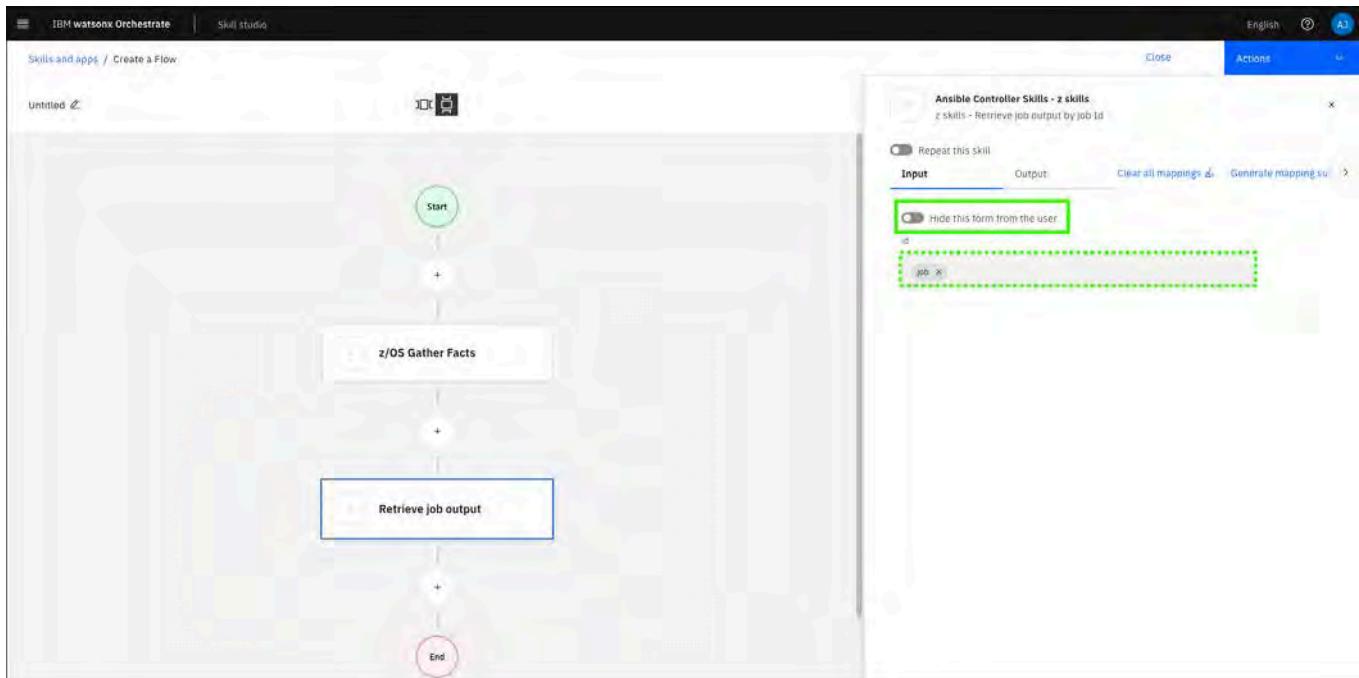


13. Verify that the **job** appears in the **id** field.

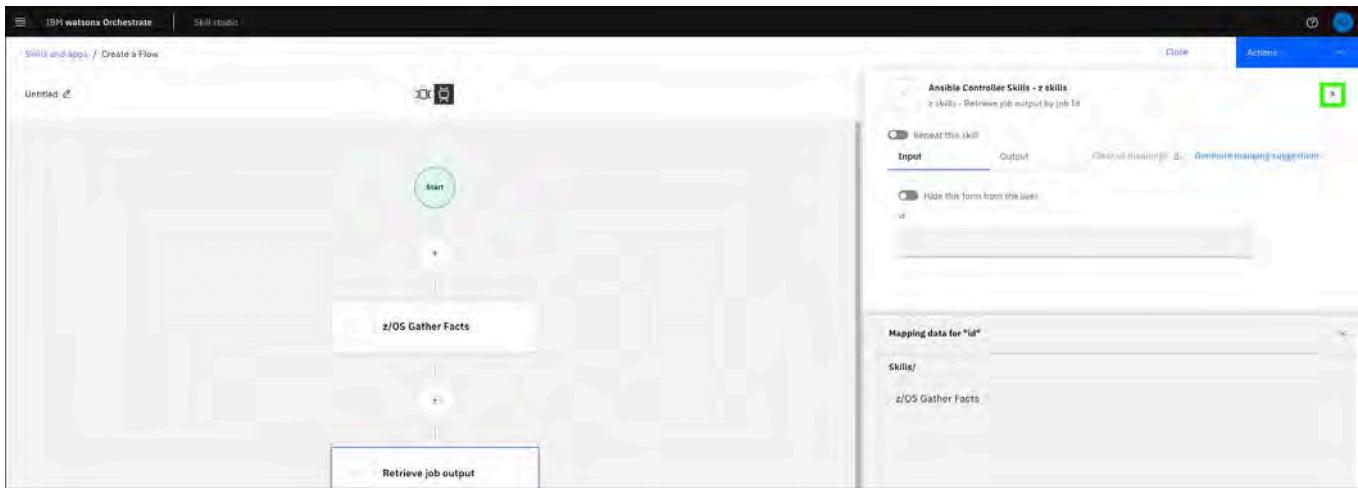


14. Optionally, toggle the **Hide this from the user** setting.

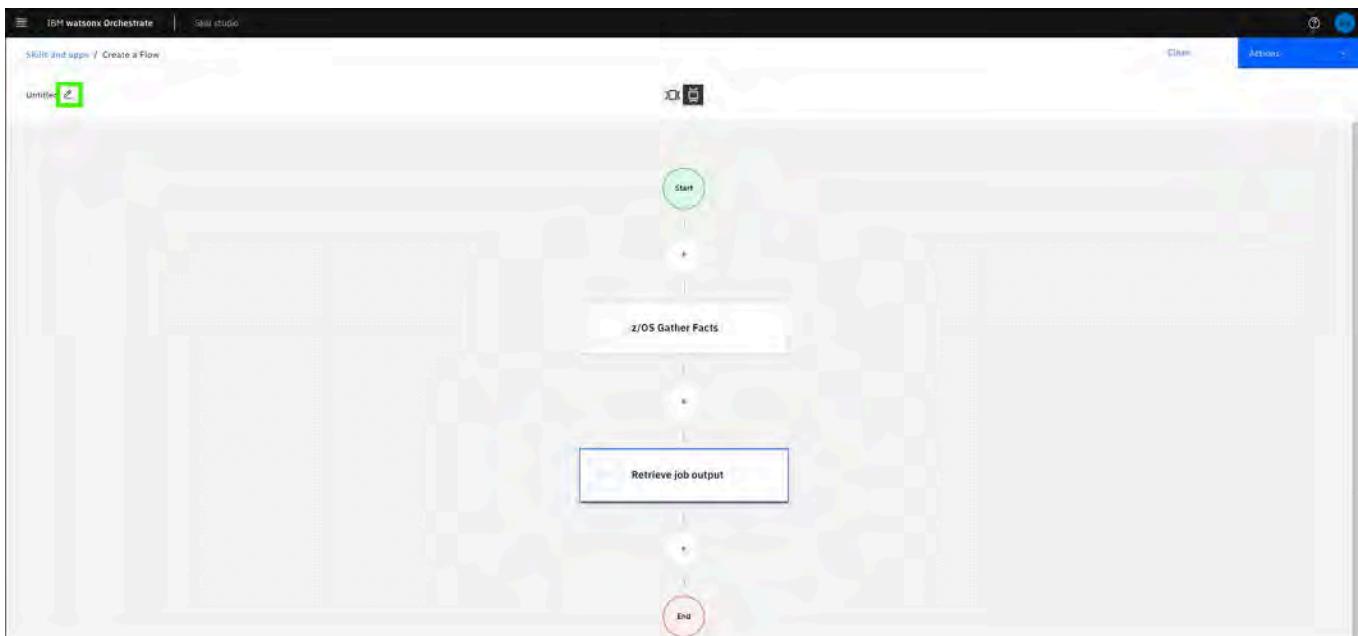
For this lab guide, this option is left disabled. Learn more about this option [here](#).



15. Click the x to close mapping window.



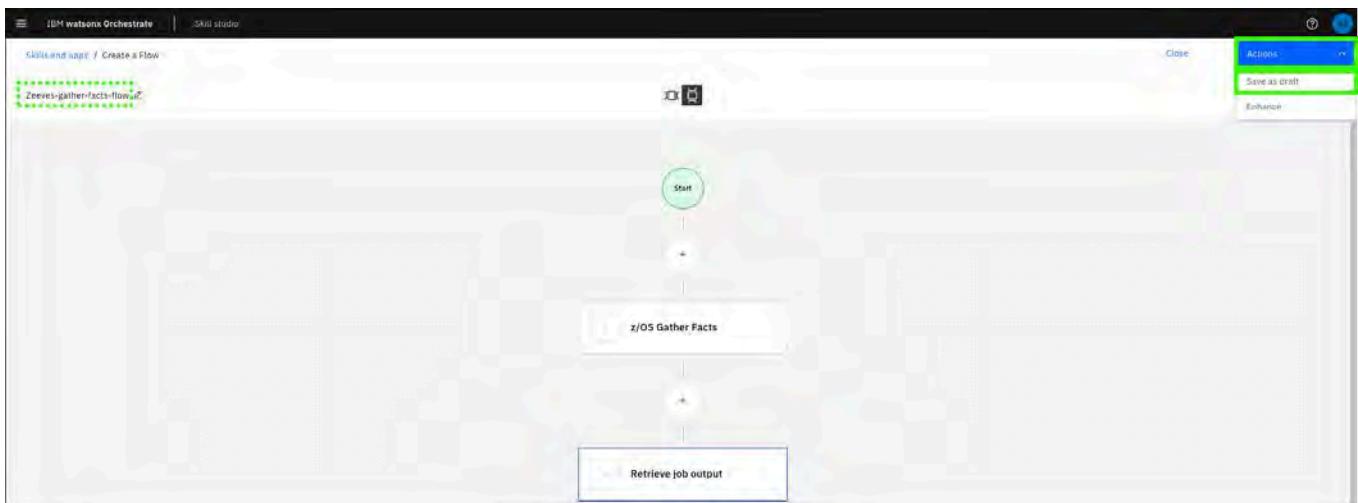
16. Click the pencil ().



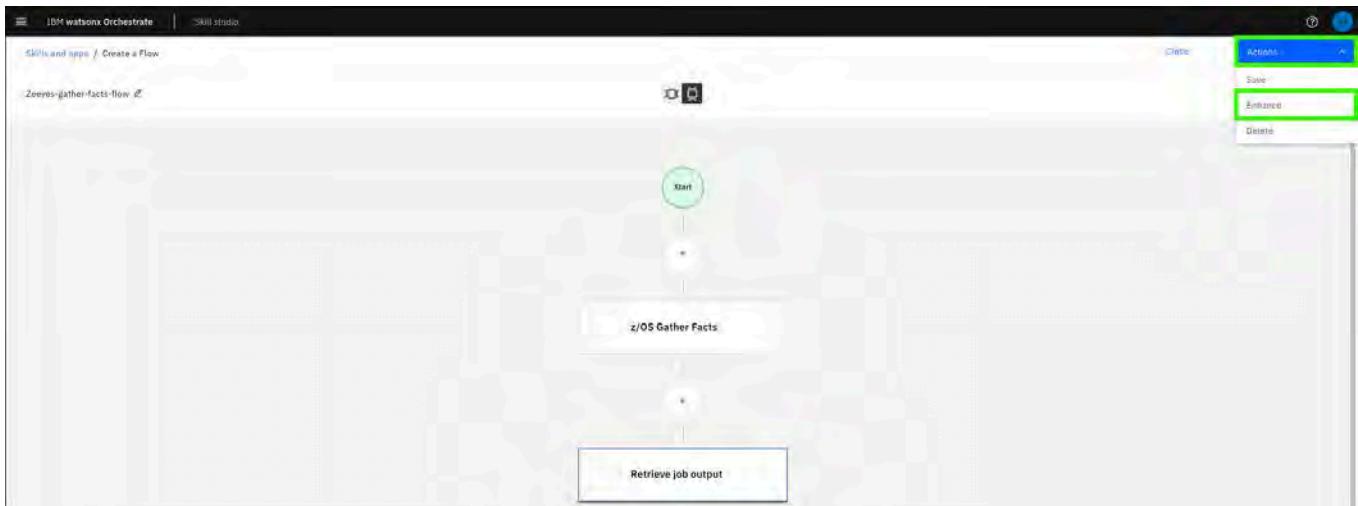
17. Enter a (a) **Name** and (b) **Description** for your skill flow and then (c) click **Save**.



18. Expand the **Actions** pull-down and click **Save as draft**.



19. Expand the **Actions** pull-down and click **Enhance**.



On the **Enhancing the skill** pages, you can:

- modify the skill name, description, and version
- add phrases (prompts) that will be recognized by the assistant to call the skill flow

## 20. Click the **Phrases** tab.

## 21. Replace the existing **phrases** (prompts) and then click **Publish**.

Notice that the default prompts are either not intuitive (the skill flow name) or a bit verbose. Replace the existing phrases with phrases that you anticipate users will use.

Example prompts:

Show me z/OS facts

Gather and display z/OS facts

Enable the skill flow in your assistant

1. Click **AI assistant builder** in the main menu.

The screenshot shows the IBM Watsonx Orchestrate interface. The left sidebar has a dark theme with several sections: Chat, Skill sets, Skill catalog, BUILD (with AI assistant builder highlighted), Skill studio, ADMINISTER (with Access management), and a user profile section. The main area is titled "Skill studio" and contains a sub-section "studio". A message at the top right says "Published successful Published skill Zenver-gather-facts-flow 15.18.32". Below this is a button labeled "Configure previous skills". The central part of the screen displays a table with four rows of skill flow entries. The columns are: Step in the process, Status, Skill type, Author, and Last edited. All entries are "Published" Skill flow type by "andrew@jones-tx.com" last edited on November 19 2024.

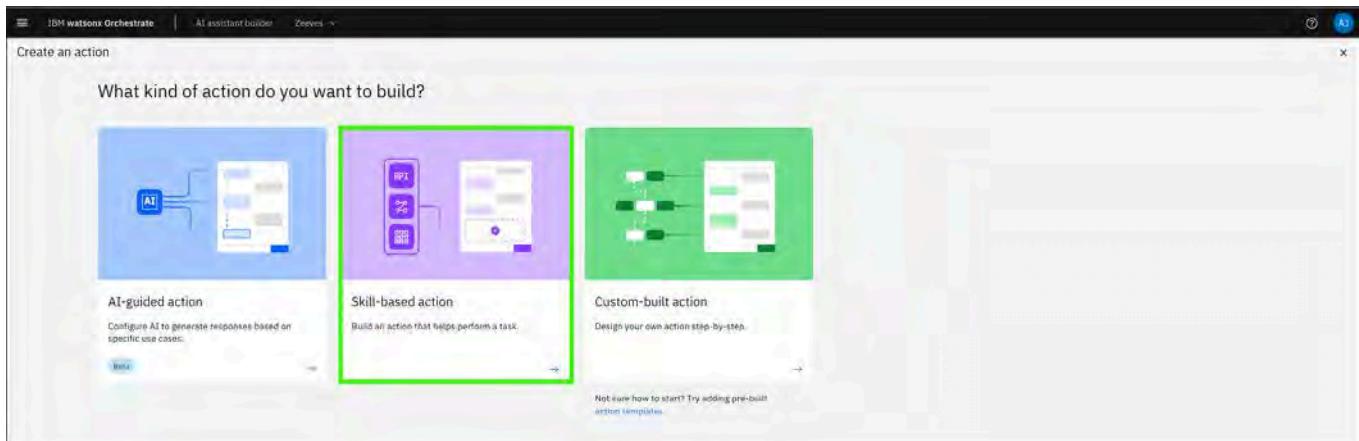
2. Hover over the **Home (🏡)** and click **Actions**.

The screenshot shows the "AI assistant builder" interface. The left sidebar has sections: Home (highlighted with a green box), Generative AI, and Actions (also highlighted with a green box). The main area is titled "Actions" and includes a "Preview" section with a timestamp of "Nov 16, 2024 11:42 (4 AM CEST)". Below this is a "as new Watson generative AI features! Learn more about intelligent information gathering." message. The main content area shows a series of numbered tiles: 1. Build actions, 2. Customize your greeting, 3. Create a fallback plan, 4. Preview & debug, 5. Customize web chat, 6. Set up a channel, 7. Set up live agent, and 8. Publish first. At the bottom, there's a "STRUCTURE" section with a diagram showing "Default behavior" branching into "General purpose" (granite-3-#b-instruct) and "Conversational search" (Custom-service). There's also a "Search" bar.

3. Click **New action**.

The screenshot shows the "Actions" list interface. The left sidebar has sections: All items, Created by you (highlighted with a green box), Variables, Saved responses, and a search bar. The main area lists actions with columns: Name, Last edited, Examples count, Steps count, and Status. One action is listed: "2025 Daffodil Field" (Last edited 2 hours ago, 0 examples, 0 steps, Status: Active). At the bottom right, there is a green button labeled "New action".

4. Click the **Skill-based action tile**.



## 5. Click the skill flow that you created earlier and then click **Next**.

**Note:** it may take a minute for the tiles to appear on the screen.

The screenshot shows the 'Build an action from a skill' interface. It lists several skills:

Skill Name	Description	Last Updated	Actions
Zeeves-gather-facts-flow	Run z/OS gather facts and retrieve the output.	2024-11-19T21:18:31.793Z	
Retrieve job output	z skills - Retrieve job output by job Id	2024-11-19T20:08:59.538Z	
z/OS Ping	z skills - This playbook pings the z/OS host to test connectivity.	2024-11-19T15:58:20.567Z	
z/OS Gather Facts	z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS-specific information from the z/OS host.	2024-11-19T15:56:26.843Z	
Summarize the Webex meeting transcript	in-watsonx.ai	2024-11-04T10:49:36.502Z	
Summarize the Box content	in-watsonx.ai	2024-11-04T10:49:32.077Z	
Summarize a Zendesk ticket	in-watsonx.ai	2024-11-04T10:49:39.476Z	
Summarize a ServiceNow incident	in-watsonx.ai	2024-11-04T10:49:05.828Z	
Summarize a Salesforce opportunity	in-watsonx.ai	2024-11-04T10:49:01.769Z	
Sharepoint document summary	in-watsonx.ai	2024-11-04T10:49:55.707Z	
Salesloft email summary	in-watsonx.ai	2024-11-04T10:49:52.072Z	
Salesforce case summarization	in-watsonx.ai	2024-11-04T10:49:09.476Z	
Salesforce case sentiment analyze	in-watsonx.ai	2024-11-04T10:49:05.828Z	
Outlook email summary	in-watsonx.ai	2024-11-04T10:49:01.769Z	
Github issue summarization	in-watsonx.ai	2024-11-04T10:49:55.707Z	

## 6. Enter an example prompt for the skill and click **Save**.

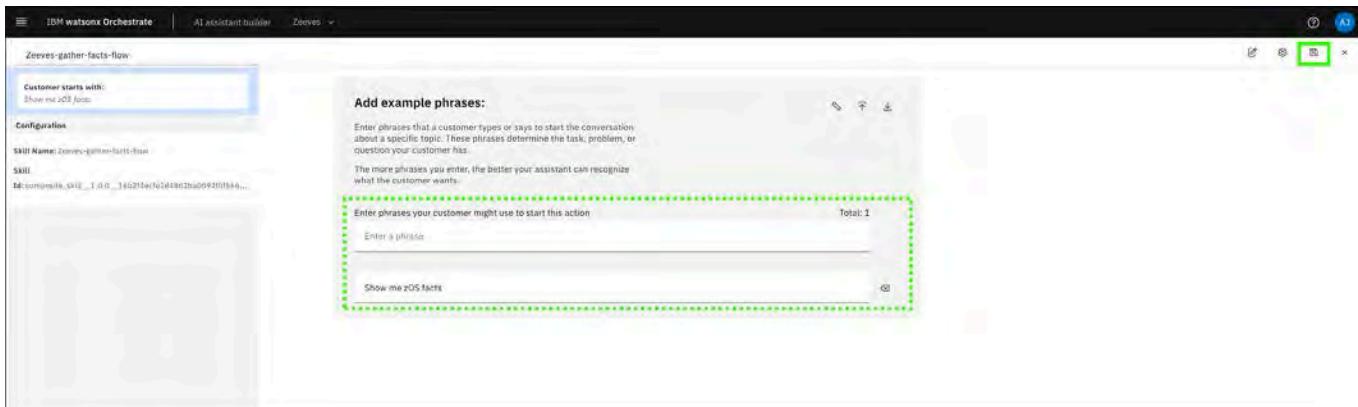
You can use one of the prompts you used earlier for the skill flow.

The screenshot shows the 'Add example phrases' interface for the 'Zeeves-gather-facts-flow' skill. It includes fields for 'Customer starts with:' and 'Add example phrases:'.

In the 'Add example phrases:' section, there is a text input field containing 'Show me z/OS facts'. A note below says 'The more phrases you enter, the better your assistant understands what the customer wants.'

A modal window titled 'New action' is open, asking 'What does your customer say to start this interaction?' with the input 'Show me z/OS facts' highlighted with a green border.

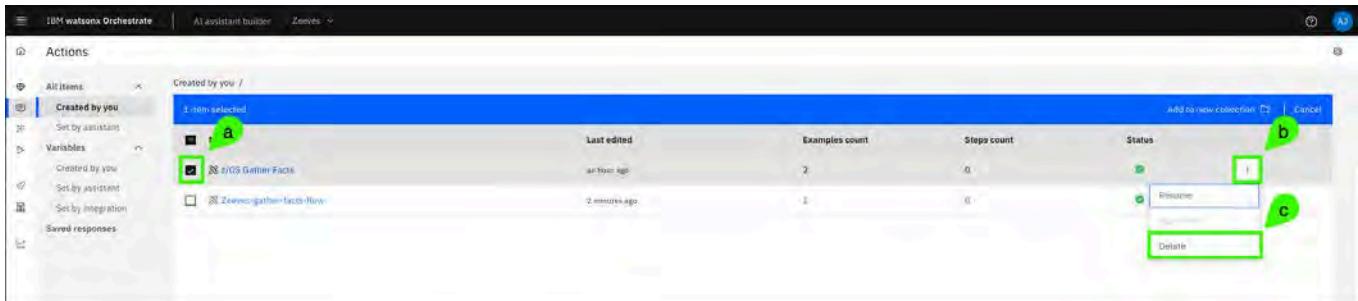
## 7. Enter any additional phrases (prompts) and then click the **save** (💾).



8. Click close (x).

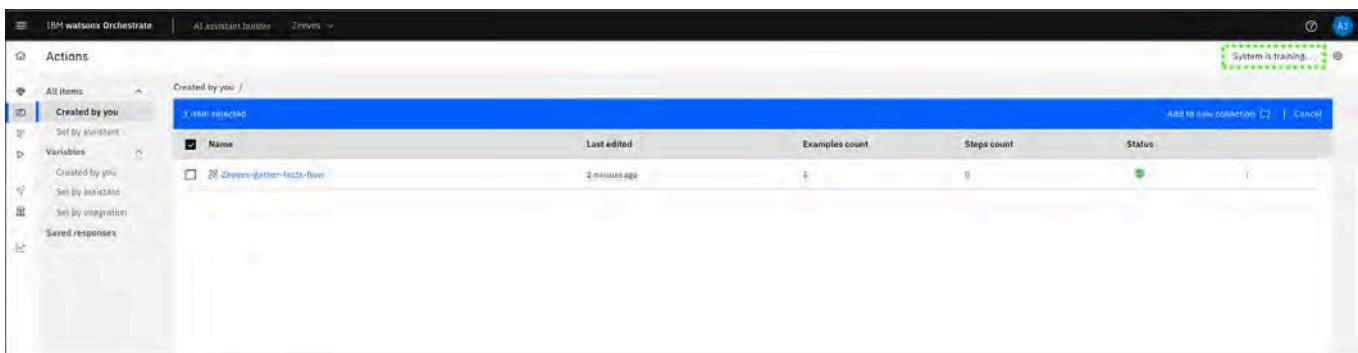


9. Select the *original* skill that you created (a) (not the skill flow you just created), click the ellipses (b), and then click **Delete** (c).



10. Wait for system training to complete.

**Note:** The message changes to "System is trained" and then disappears.



11. Click **Preview**.

Name	Last edited	Examples count	Steps count	Status
Zoszos-gather-facts-flow	3 minutes ago	1	0	<span style="color: green;">Green</span>

12. Enter one of the prompts you specified into the assistant preview.

Show me z/OS facts

Name	Last edited	Examples count	Steps count	Status
Zoszos-gather-facts-flow	11 minutes ago	1	0	<span style="color: green;">Green</span>

13. Wait 10 seconds and then click **Apply**.

**Note:** It is important to wait for the first job to complete before submitting the second job in the flow.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with 'Actions' (All items, Created by you, Set by assistant, Variables, Saved responses), 'Variables' (Created by you, Set by assistant, Set by integration), and 'Saved responses'. The main area is titled 'Created by you /' and shows a table with one item selected: 'Zeeves-gather-facts-flow'. The table columns are Name, Last edited, Examples count, Steps count, and Status. The status is green with a 'Running' icon. A preview window on the right shows a conversation history and a message input field.

#### 14. Review the results from the skill flow.

Use both scroll bars in the assistant preview to review all the returned information. The output is similar to what was seen in the AAP web console. The character strings like [0;32m are special characters that are not properly displayed in the assistant preview interface.

This screenshot is similar to the previous one but with the 'Preview' window expanded to show more of the conversation history. The history includes several messages, each with a timestamp, a user icon, and a message content. The content includes various text snippets and some special characters like [0;32m, which are highlighted with green dashed boxes. The message input field at the bottom is also visible.



## Sample output from the Z/OS gather facts flow.



### Content

```

Identity added: /runner/artifacts/16/ssh_key_data (/runner/artifacts/16/ssh_key_data)
[1;35m[WARNING]: Collection ibm.ibm_zos_core does not support Ansible version 2.14.2[0m

PLAY [Gather z/OS-specific facts.] *****
TASK [Gather all facts about z/OS host.] *****
TASK [Print gathered facts about the master catalog.] *****
[0;32mok: [zos_host][0m
[0;32m      "master catalog dsn: CATALOG.VS01.MASTER",0m [0;32m      "master catalog volser: OPEVS1"[0m
[0;32m ][0m [0;32m}{0m

TASK [Print only CPC and IODF info from gathered z/OS facts.] *****
[0;32mok: [zos_host] => {[0m
[0;32m      "msg": {[0m [0;32m      "manufacturer: IBM",0m [0;32m      "model: A00",0m [0;32m      "plant: C1",0m
[0;32m      "iodf name: PROV.IODF00",0m [0;32m      "iodf config: DEFAULT"[0m [0;32m ][0m [0;32m}{0m

TASK [Print out all gathered facts about the z/OS host.] *****
[0;32mok: [zos_host] => {[0m
[0;32m      "ansible_facts": {[0m [0;32m      "arch_level": "2",0m [0;32m      "cpc_nd_manufacturer": "IBM",0m
[0;32m      "cpc_nd_model": "A00",0m [0;32m      "cpc_nd_plant": "C1",0m
[0;32m      "cpc_nd_seqno": "20D90792EB76",0m [0;32m      "cpc_nd_type": "008562",0m [0;32m      "edt": "00",
[0m [0;32m      "hw_name": "",0m [0;32m      "ieasym_card": "(00,K2)",0m [0;32m      "io_config_id": "00",0m
[0;32m      "iodate": "",0m [0;32m      "iodesc": "",0m [0;32m      "iodf_config": "DEFAULT",0m
[0;32m      "iodf_name": "PROV.IODF00",0m [0;32m      "iodf_unit_addr": "DE28",0m [0;32m      "ioproc": "",0m
[0;32m      "iotime": "",0m [0;32m      "ipaloadxx": "K2",0m [0;32m      "ipl_volume": "D25VS1",0m
[0;32m      "load_param_device_num": "DE28",0m [0;32m      "load_param_dsn": "SYS0.IPLPARM",0m
[0;32m      "lpar_name": "",0m [0;32m      "master_catalog_dsn": "CATALOG.VS01.MASTER",0m
[0;32m      "master_catalog_volser": "OPEVS1",0m [0;32m      "nucleus_id": "1",0m
[0;32m      "operator_prompt_flag": "M",0m [0;32m      "parmlib_dsn": "K2.PARMLIB",0m
[0;32m      "parmlib_volser": "USRVS1",0m [0;32m      "primary_jes": "JES2",0m
[0;32m      "product_mod_level": "00",0m [0;32m      "product_name": "z/OS",0m
[0;32m      "product_owner": "IBM CORP",0m [0;32m      "product_release": "05",0m
[0;32m      "product_version": "02",0m [0;32m      "smf_name": "VS01",0m [0;32m      "sys_name": "VS01",0m
[0;32m      "sysplex_name": "LOCAL",0m [0;32m      "tsoe_rel": "05",0m [0;32m      "tsoe_ver": "4",0m
[0;32m      "vm_name": ""}[0m [0;32m }{0m [0;32m}{0m

PLAY RECAP *****
[0;32mzos_host[0m          : [0; 32mok=4
[0m changed=0  unreachable=0  failed=0  skipped=0  rescued=0  ig nored=0

```

The previous scenario might or might not be relevant for your client's use case. The scenario illustrates how to sequence skills together in a skill flow to create an action that your assistant triggers based on prompts that use the pre-configured Ansible automation templates. You are encouraged to create your own skill flows and prompts that use other skills available within the AAP instance. As an example, create a skill flow for the **z/OS Ping** skill. Be sure to add the **Retrieve job output** skill to view the results.

Next, learn about custom-built actions.

# Creating custom-built actions

To this point, you learned how to:

- import skills into Watsonx Orchestrate
- add applications with those skills to your assistant
- create skill-based actions for your assistant
- combine skills in a skill flow

You can also create **custom-built** actions. Custom-built actions have actions with different steps to take in conversations and form sequences of prompts that define the conversation experience. The steps can be defined with or without conditions, which help control the custom responses. Steps within the custom action can end with routing to conversational search, triggering another existing subaction, and other actions. Custom-built actions are a powerful way of customizing the user's experience.

Learn more about creating custom-built actions [here](#).

# Importing pre-packaged z/OS skills

Provided with Version 2 of watsonx Assistant for Z is a set of pre-packaged skills. These skills are used to automate various tasks on z/OS, such as running different console commands and retrieving logs from batch jobs.

The list of pre-packaged skills available include:

- Authorized program list
- z/OS IPL Information
- Display zOS parmlib datasets
- UNIX System Services options
- Display zOS subsystems
- List spool files
- Retrieve dataset content
- Retrieve spool file content
- Retrieve z/OS Management Facility (OSMF) job status

IBM watsonx Orchestrate requires that any OSMF environment you connect to for skill execution has certificate authority (CA) signed certificates.

As an example, the following are console commands that are used in some of the pre-packaged skills:

- Authorized Program list - `operator command -> d prog,lnklist`
- z/OS IPL Information - `operator command -> d iplinfo`
- Display zOS parmlib datasets - `operator command -> d parmlib`

You can import the pre-packaged skills to your sandbox environment by downloading the compressed file [here](#) and following [these instructions](#).

Extract the embedded JSON file and modify the file for your environment by following [these instructions](#).

# Publishing and deploying your assistant

To this point, acting as an Assistant Builder, you built the assistant, configured conversational search, and added skills and automations. You tested your assistant by using the **preview** capability of AI Assistant Builder. The **preview** capability is a closed environment for experimenting with prompts.

After your assistant is finalized, you can publish it to make it available to users. Each assistant that you create comes with two **environments**: *draft* and *live*. You configured your assistant in the draft environment. Each environment has its own set of IDs, URLs, and service credentials referenced by external services.

The **Environments** page in the AI assistant builder has tabs for managing both the **Draft environment** and the **Live environment**:

This screenshot shows the 'Environments' page in the AI Assistant Builder. At the top, there are tabs for 'Draft' and 'Live'. The 'Draft' tab is highlighted with a green dashed border. Below the tabs, there's a section titled 'Draft environment' with a description: 'Use the draft environment for internal preview at your organization. It contains your in-progress, unpublished content and draft channel integrations.' To the right, there are sections for 'Channels' (Web chat), 'Resolution Methods' (Draft content, Last edited), and 'Extensions' (Search). A 'Preview this environment' button is located in the top right corner of this section.

This screenshot shows the same 'Environments' page, but the 'Live' tab is now highlighted with a green dashed border. The 'Live environment' section has a description: 'Use the live environment for deployment to customers. It contains your published content and channel integrations where customers interact with your assistant.' The 'Channels' section (Web chat) and 'Extensions' section (Search) are visible on the right. The 'Resolution Methods' section is partially visible at the bottom.

The **Draft environment** contains all your in-progress work in the Actions, Preview, and Publish pages. Use the **Draft environment** tab to build out your assistant and use for internal testing before deployment. Any integrations (channels) that you use for the **Draft environment** are unique to that environment, and changes to draft integrations don't affect the **Live environment**.

## Publish the assistant

Each time that you publish, you're creating a new version of the assistant, for example V1. When you publish your content, you're creating a snapshot of the draft content, resulting in a version.

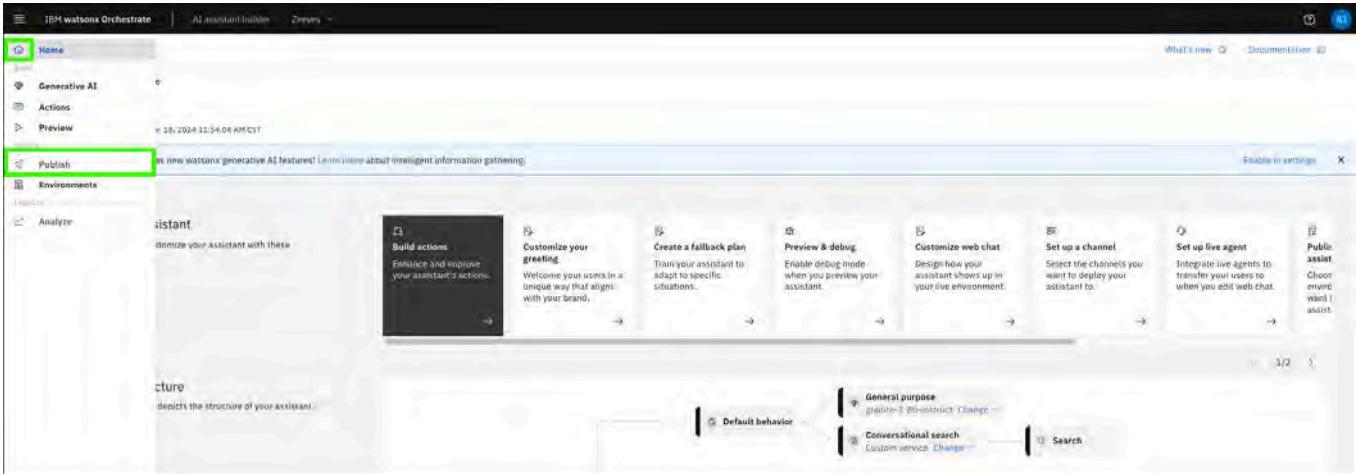
### ⚠ Versions do not contain integration configurations or environment settings

Published versions contain all of the content from actions, including settings and variables. **However, versions do not contain integration configurations or environment settings.** Integration configurations and environment settings must be configured manually in each environment.

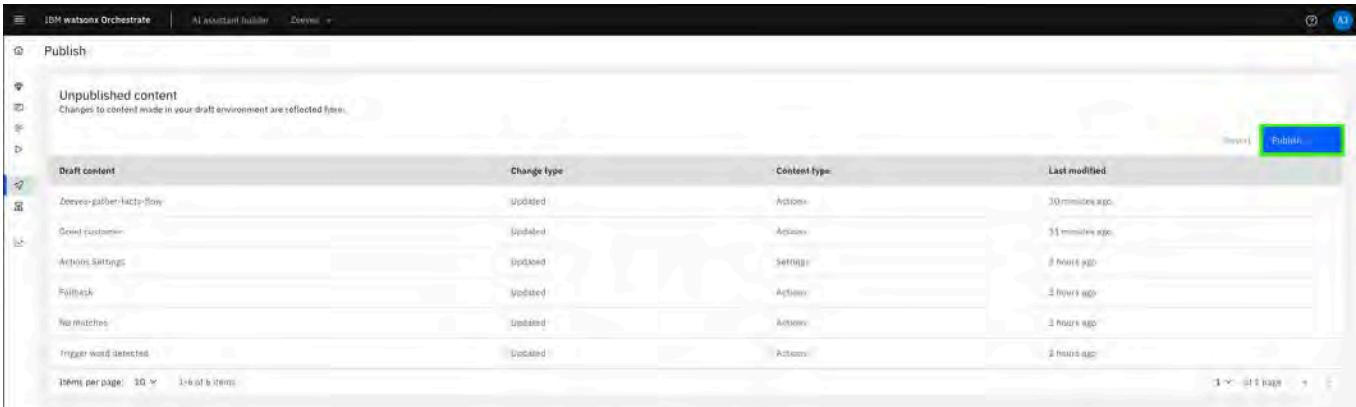
For managing quality-control and versioning, the Live environment is the version of the assistant to give to users.

Follow these steps to publish the first version of your assistant by using Assistant Builder:

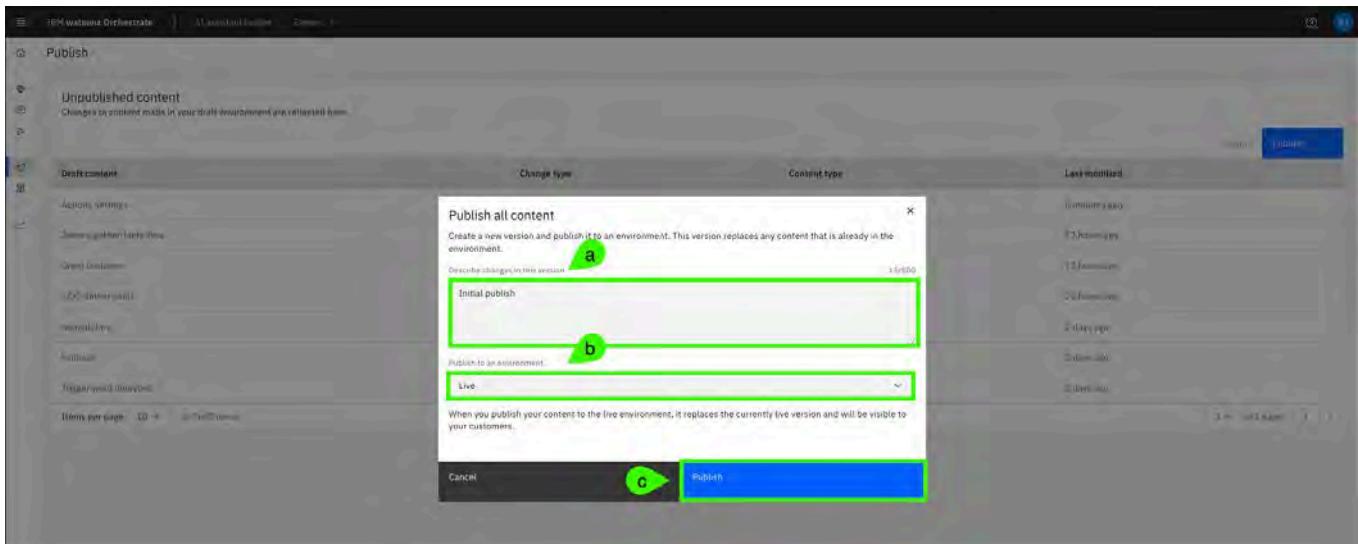
1. Hover over the **Home** icon () and click **Publish**.



2. Click **Publish**.



3. Enter a description of the changes (a), set the environment to **Live** (b), and then click **Publish** (c).



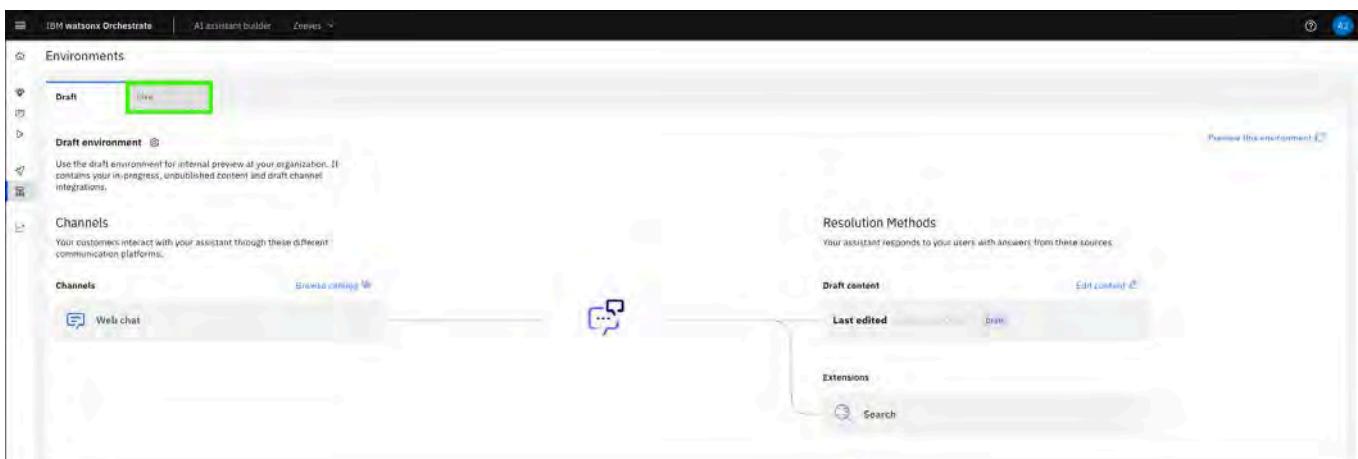
## Configure the live environment

**Important:** When the live environment is created, the settings from the draft environment are not carried over (for example, the configuration of the OpenSearch instance used for conversational search).

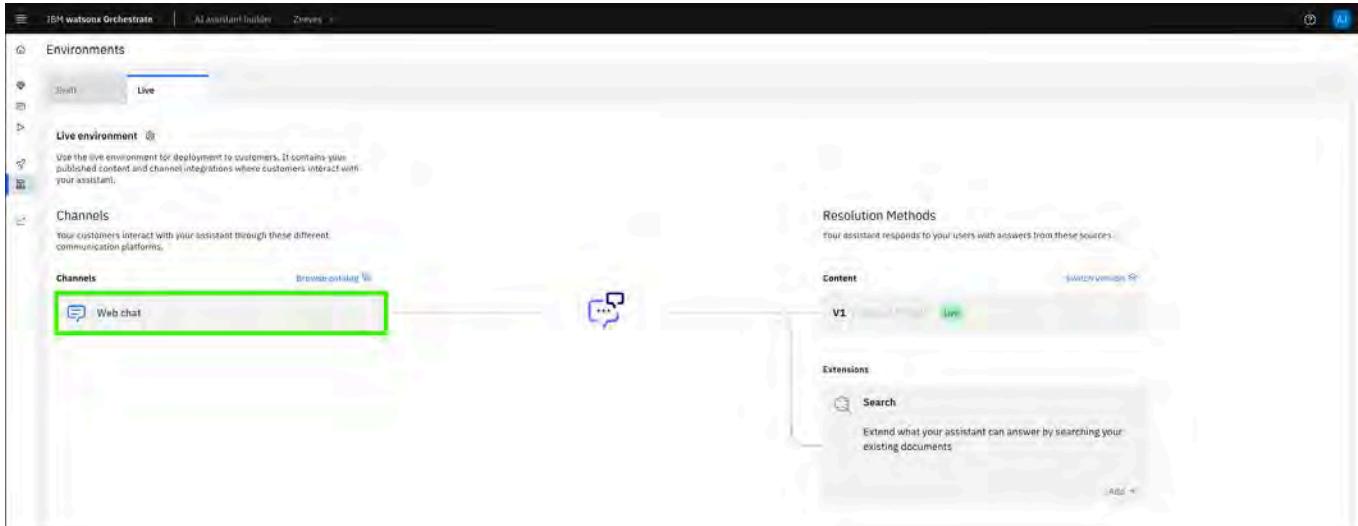
1. Hover over the **Home** icon (🏡) and click **Environments**.



2. Click **Live**.



### 3. Click Web chat.



### 4. Customize the live assistant as you see fit.

On the **Style** tab, you're able to set the Assistant name that is displayed on the chat window when users are interacting with the assistant. For pilots or demos, consider personalizing this name for the client. Also in the **Style** tab, you can set the themes and display settings of the chat windows, including the ability to enable the IBM Watermark and enable streaming.

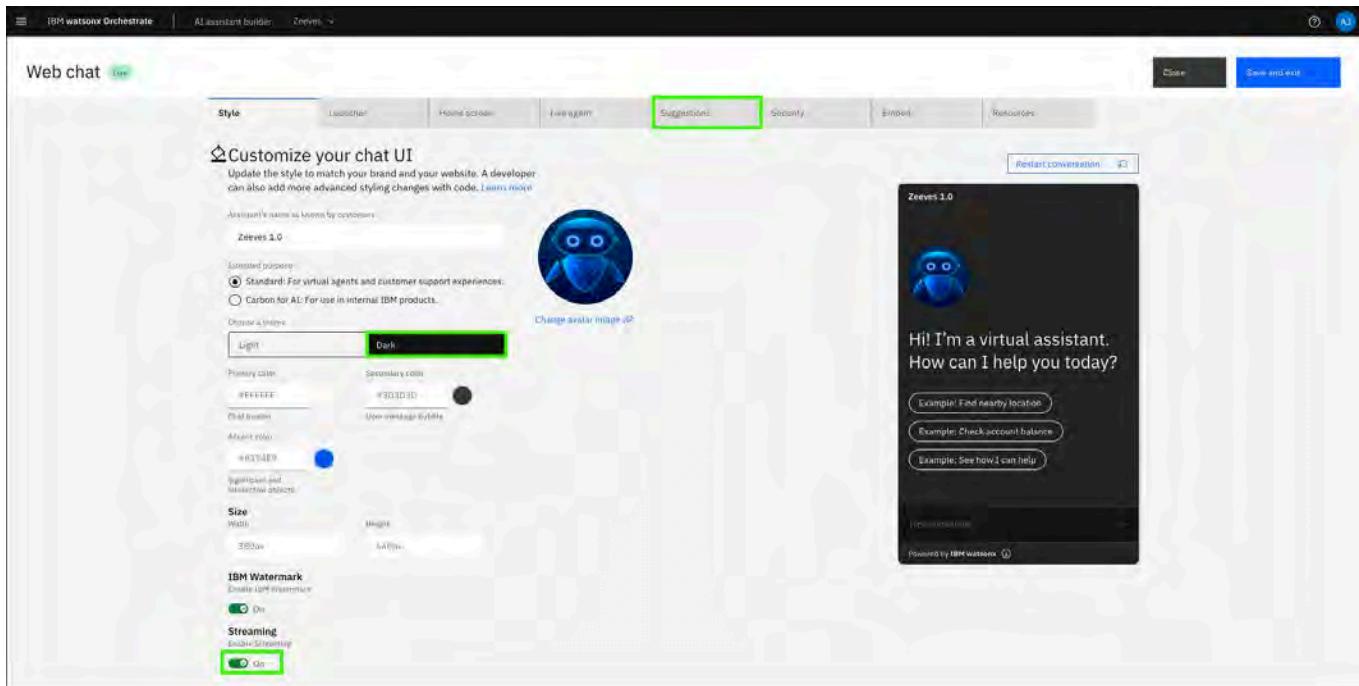
On the **Home** tab, you enable and customize the greeting message from the assistant when the user accesses the assistant chat. You are also able to set **Conversation starters** that are displayed in the chat window. When selected by the user, the text of the conversation starter is sent as a prompt, so it is important that your assistant is trained and tested to answer correctly. It is highly recommended to remove the default conversation starters and create your own. The ability to add a Background style for the assistant chat window is on the home screen tab.

Explore all the other tabs.

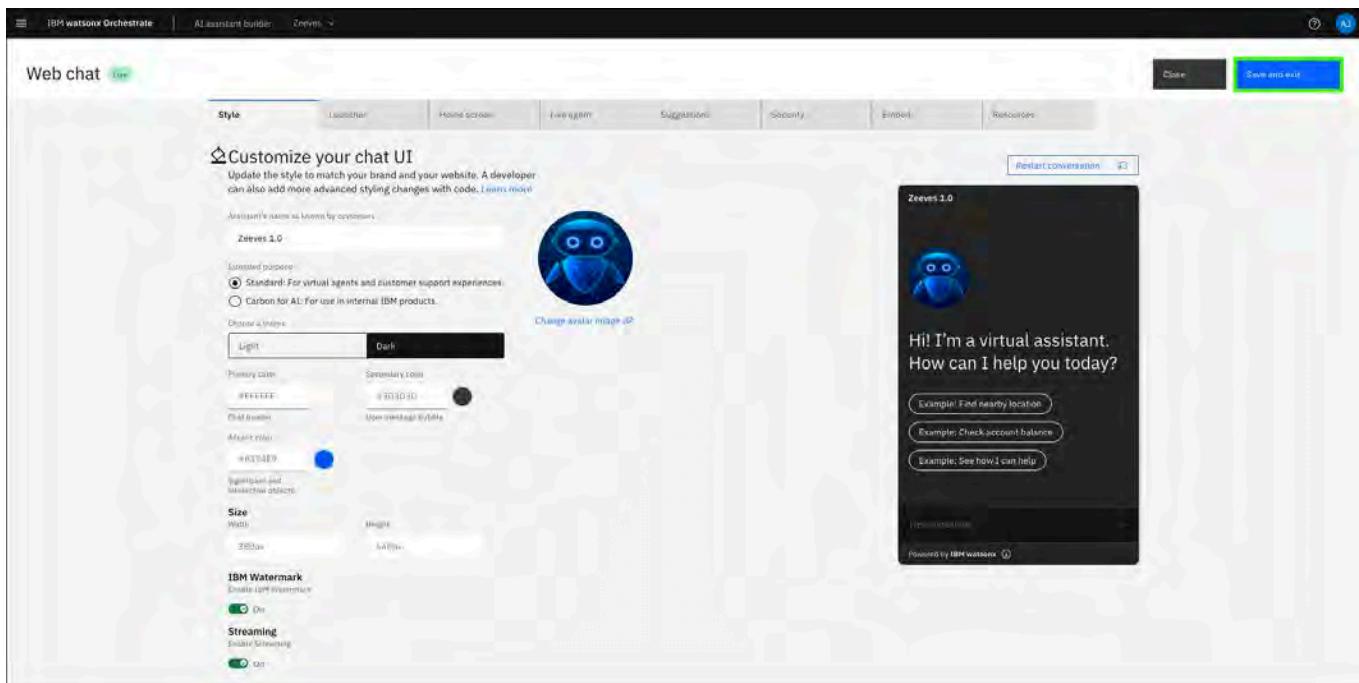


#### Customize your live environment.

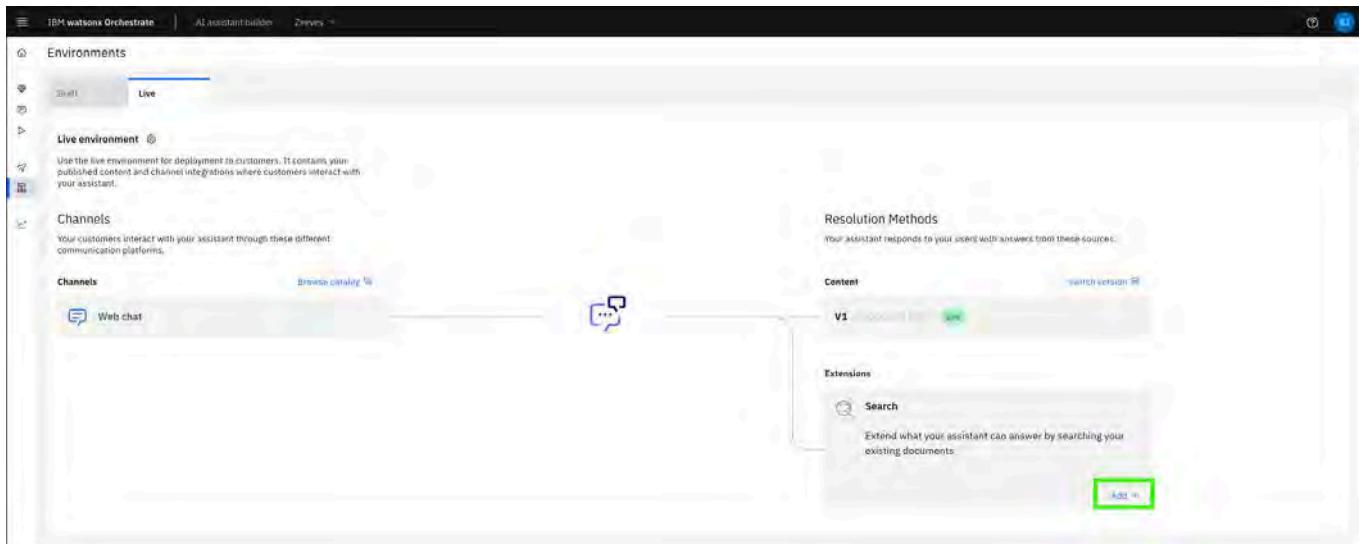
For this lab, toggle **Streaming** on and turn **Suggestions** off on the **Suggestions** tab. You may also want to change the theme to **Dark** to differentiate your draft and live environments.



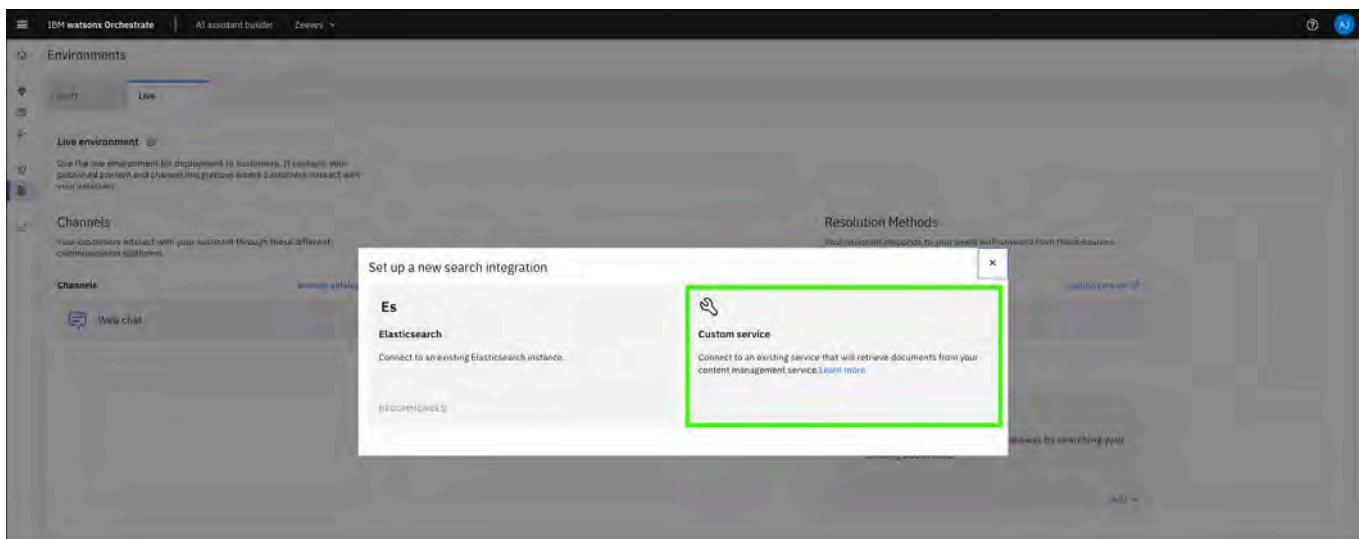
## 5. Click Save and exit.



## 6. Click Add in the Search tile.



## 7. Click Custom service.

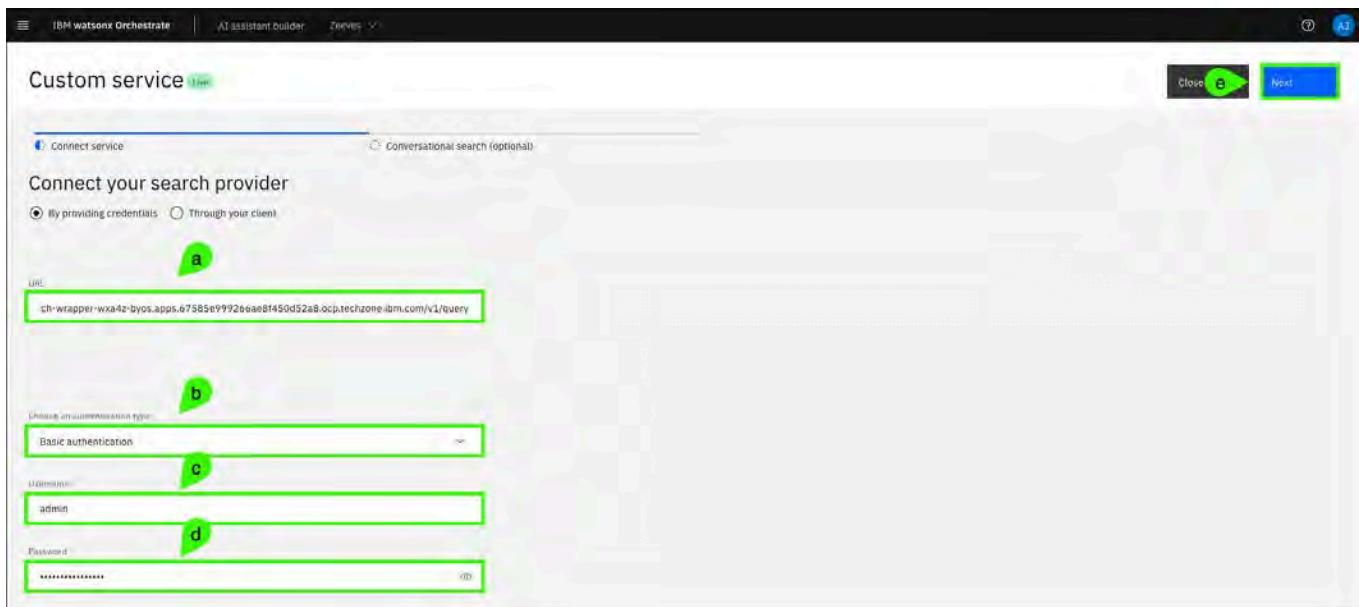


- Enter the URL for your bring-your-own-search (BYOS) engine (a), select Basic authentication for the authentication type (b), enter admin for the Username (c), enter the password that you specified in the wrapper-creds.yaml file for the Password (d), and then click Next (e).

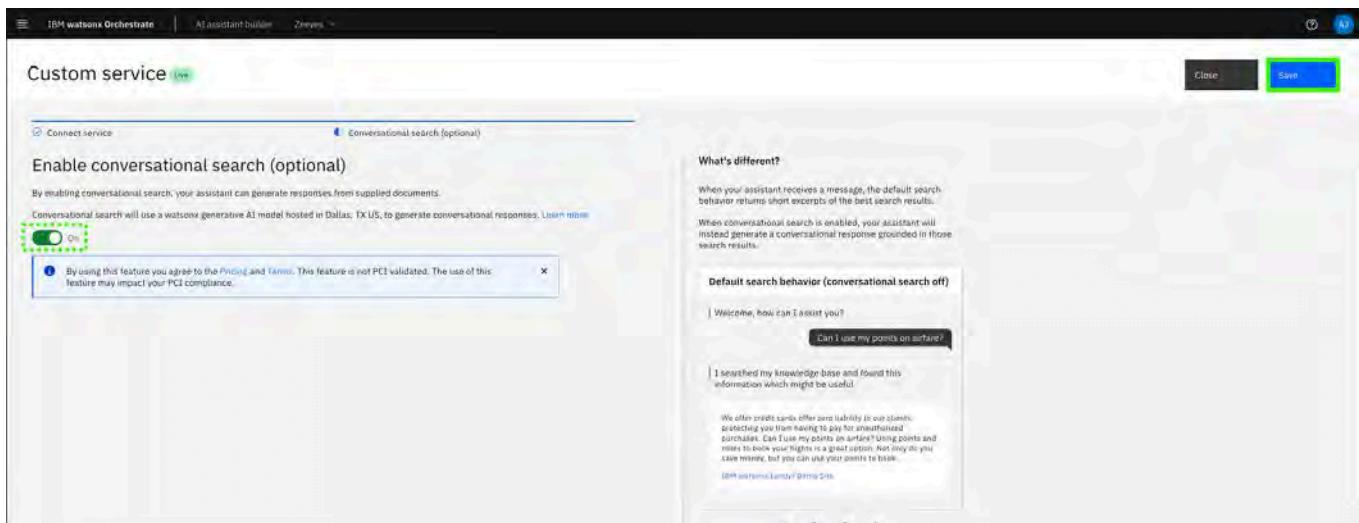


**Use the correct URL and authentication type!**

Use the URL and credentials for your BYOS OpenSearch engine created earlier [here](#).



## 9. Verify conversational search is enabled and click Save.



## 10. Update the **Custom service** settings (a-f), click **Save** (g), and then click **Close** (h).

### Customize the settings.

This is your assistant. Feel free to customize the settings. The settings shown below reflect the changes made earlier in the lab guide to the draft version of the assistant. This includes the **Metadata** field to weigh ingested client documents higher using:

```
{
  "doc_weight": {
    "product_docs": 0.2,
    "customer_docs": 0.8,
    "ibm_indices": "*_ibm_docs_slate, *_ibm_redbooks_slate",
    "standardize": true,
    "customer_indices": "customer_"
  }
}
```

**Custom service**

**Settings** **Instance**

By using this feature you agree to the [Privacy and Terms](#). This feature is not PCI validated. The use of this feature may impact your PCI compliance.

**Conversation search**

Use a Watson generative AI model hosted in Dallas, TX US, to generate conversational responses.

**Conversation type**

**a** On

**b** Off

**Search configuration**

To test this, go to the evaluations page

**c** Retrieval confidence threshold: Lowest

**d** Generated response length: Verbose

**e** Retain references threshold: Lowest

**Citations**

Citations will be displayed to the end user. This doesn't impact the number of citations used in the response.

**Default filter**

Can be anything (e.g., test, 350), etc.

**Metadata**

```
{"doc_weight": "product_docs:0.2, customer_docs:0.8", "item_id_index": "...", "item_doco_state": "customer_impressions", "customer_impressions": "customer..."}
```

**Search display text**

Define the text your search will display to the end user:

No results found Connectivity issue

I searched my knowledge base, but did not find anything related to your query.

**Close** **Save**

## Connect the skills to the live environment

1. Click **Skill sets** in the main menu.

2. Select your assistant's live instance in the **Skill sets** list.

3. Click **Connections**.

4. Search for the application name you specified earlier.

Zeeves live

Skill sets

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to evaluate skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	1	Not specified		<span style="border: 1px solid green; padding: 2px;">⋮</span>

Items per page: 8 1-1 of 1 items

5. Click the ellipses (⋮) for your app and then click **Connect app**.

Zeeves live

Skill sets

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to evaluate skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	1	Not specified		<span style="border: 1px solid green; padding: 2px;">⋮</span> <span style="border: 1px solid green; padding: 2px;">Connect app</span>

Items per page: 8 1-1 of 1 items

6. Click **Connect app**.

Zeeves live

Skill sets

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to evaluate skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	1	Not specified		<span style="border: 1px solid green; padding: 2px;">⋮</span>

Items per page: 8 1-1 of 1 items

Connect to Ansible Controller Skills - z skills

Each team member uses their own credentials to connect to this app and use its skills.

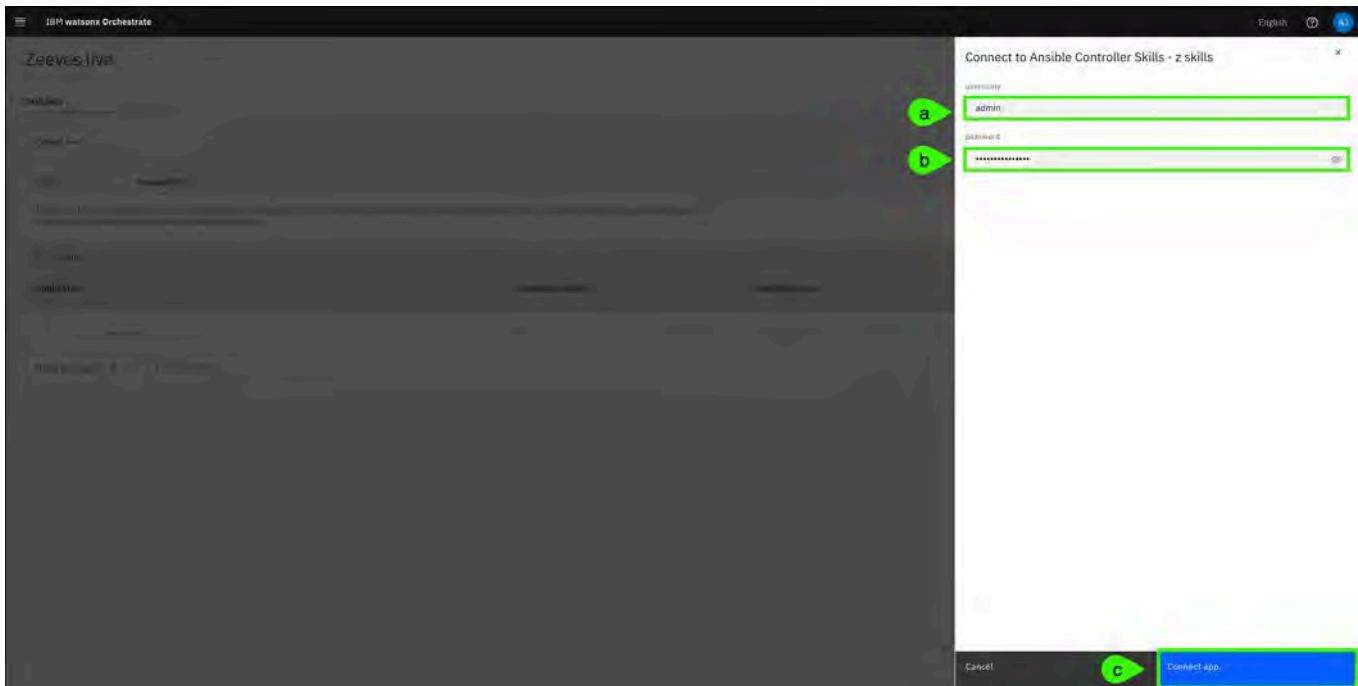
Team credentials

The admin sets the credentials each team member uses to connect to this app and use its skills.

You selected **Team credentials** for the credential type. Click **Connect app** to provide the credentials your team will use to connect to the app.

Connect app

7. Enter the **username (a)** and **password (b)** using the username ( admin ) and password for your IBM Technology Zone (ITZ) watsonx Assistant for Z Pilot - AAP & z/OS reservation, and then click **Connect app (c)**.



Learn more about publishing your assistant and creating live environments [here](#).

## Deploy the assistant

After configuring your assistant's settings and publishing, the final step is to deploy your assistant, which can be done across various channels depending on the use case.

Several options exist for deploying your assistant through channels and integrations to satisfy the use cases that you might encounter. Learn more about all the deployment options [here](#).

For this lab, deploy the assistant by using the web chat integration. The web chat integration provides an assistant interface that can integrate with a website. Learn more about the web chat integration [here](#).

1. Open the **Environments** page in the **AI assistant builder**.

2. Click **Web chat** for the **Live** environment.



3. Click the **Embed** tab.

4. Copy and record the `integrationID`, `region`, and `serviceInstanceId` values.

```
<script>
  const integrationID = "80da0299-e0b9-413c-8c8c-2754d9272fe3"; // The ID of this integration
  const region = "wco-us-south"; // The region your integration is hosted in...
  const serviceInstanceId = "7c1e33b1-8df1-47ab-bef2-319f531687ba"; // The ID of your service instance.
  const url = `https://web-chat.global.assistant.watson.appdomain.cloud/v2/tts?${integrationID}&${region}&${serviceInstanceId}&${urlParams}`;

  const timeout = 5000;
  const script = document.createElement('script');
  script.src = url;
  document.head.appendChild(script);
</script>
```

5. In a text editor, create a file that is named `Watson Assistant Chat.html` and paste the following text in the file.

File name:

```
Watson Assistant Chat.html
```

File contents:

```

<html lang="en">
<head>
<title>Watson Assistant Chat</title>
<meta name="viewport" content="width=device-width, initial-scale=1">

<style>
.WebChatContainer {
  position: absolute;
  left: 0;
  right: 0;
  top: 0;
  bottom: 0;
}
</style>
</head>
<body>

<div class="WebChatContainer"/>

<script>
const element = document.querySelector('.WebChatContainer');

window.watsonAssistantChatOptions = {
  integrationID: "<YOUR INTEGRATION ID>", // The ID of this integration.
  region: "<YOUR REGION>", // The region your integration is hosted in.
  serviceInstanceId: "<YOUR SERVICE INSTANCE ID>", // The ID of your service instance.
  element,

  openChatByDefault: true,
  hideCloseButton: true,

  layout: {
    showFrame: false,
    hasContentMaxWidth: true,
  },
  onLoad: async (instance) => {
    window.WACInstance = instance;
    await instance.render();
  }
};

setTimeout(function() {
  const t = document.createElement('script');
  t.src = 'https://web-chat.global.assistant.test.watson.appdomain.cloud/versions/' +
  (window.watsonAssistantChatOptions.clientVersion || 'latest') +
  '/WatsonAssistantChatEntry.js';
  document.head.appendChild(t);
});
</script>

</body>
</html>

```

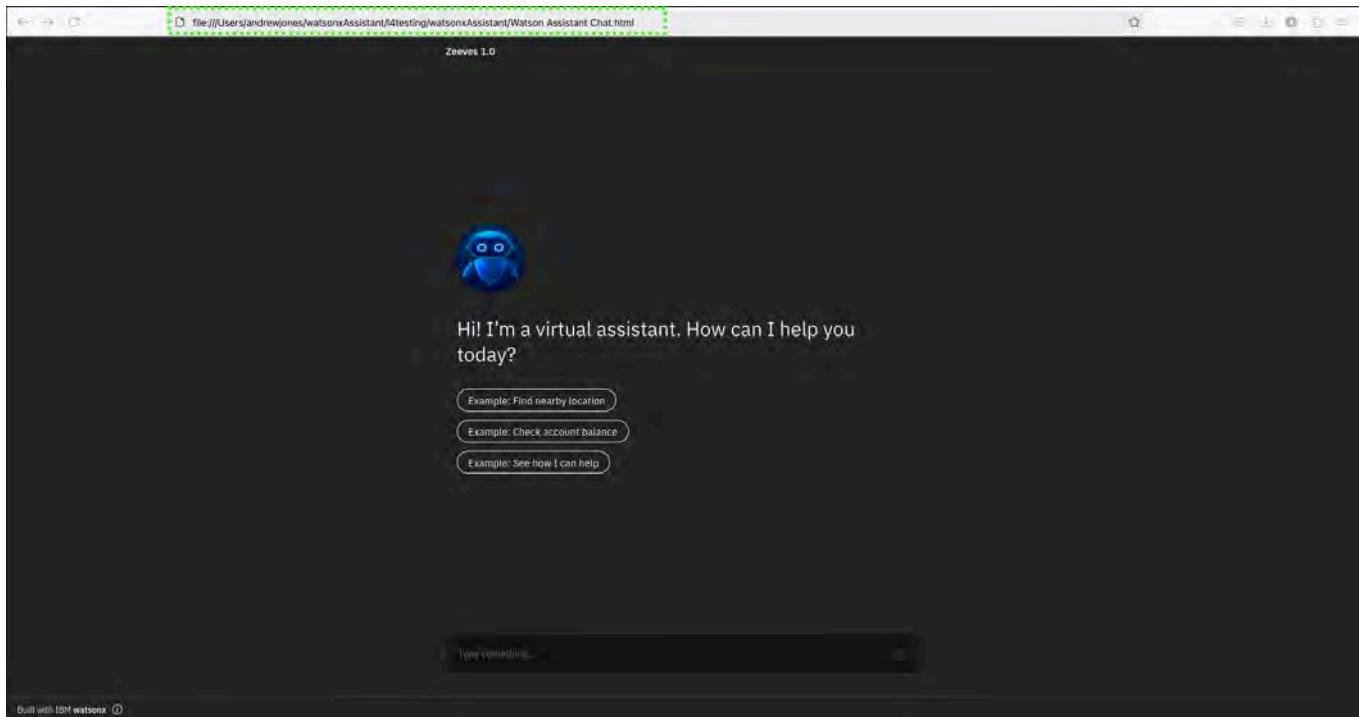
Before modification:

```
Watson Assistant Chat.html X
Users > andrewjones > Downloads > Watson Assistant Chat.html > html
  1  <html lang="en">
  2    <body>
  3      <script>
  4
  5        window.watsonAssistantChatOptions = {
  6          integrationID: "<YOUR INTEGRATION ID>", // The ID of this integration.
  7          region: "<YOUR REGION>", // The region your integration is hosted in.
  8          serviceInstanceId: "<YOUR SERVICE INSTANCE ID>", // The ID of your service instance.
  9          element,
 10
 11        openChatByDefault: true,
 12        hideCloseButton: true,
 13
 14        layout: {
 15          showFrame: false,
 16          hasContentMaxWidth: true,
 17        },
 18
 19        onLoad: async (instance) => {
 20          window.WACInstance = instance;
 21          await instance.render();
 22        }
 23      };
 24
 25      setTimeout(function() {
 26        const t = document.createElement('script');
 27        t.src = 'https://web-chat.global.assistant.test.watson.appdomain.cloud/versions/' + (window.watsonAssistantChatOptions.clientVersion || 'latest') + '/WatsonAssistantChat';
 28        document.head.appendChild(t);
 29      });
 30    </script>
 31  </body>
 32 </html>
```

After modification:

```
Users > andrewjones > Downloads > Watson Assistant Chat.html > html > body > script > layout
  1  <html lang="en">
  2    <head>
  3      <style>
  4        .WebChatContainer {
  5          top: 0;
  6          bottom: 0;
  7        }
  8      </style>
  9    </head>
 10   <body>
 11
 12     <div class="WebChatContainer">
 13
 14       <script>
 15         const element = document.querySelector('.WebChatContainer');
 16
 17         window.watsonAssistantChatOptions = {
 18           integrationID: "8b4ad299-e6b9-434c-8c8c-2754d9272fe5", // The ID of this integration.
 19           region: "wco-us-south", // The region your integration is hosted in.
 20           serviceInstanceId: "7c163381-8dcf-476b-bdf2-319f531087be", // The ID of your service instance.
 21           element,
 22
 23           openChatByDefault: true,
 24           hideCloseButton: true,
 25
 26           layout: {
 27             showFrame: false,
 28             hasContentMaxWidth: true,
 29           },
 30
 31           onLoad: async (instance) => {
 32             window.WACInstance = instance;
 33             await instance.render();
 34           }
 35         };
 36
 37       </script>
 38
 39     </div>
 40   </body>
 41 </html>
```

6. Open the `Watson Assistant Chat.html` file in a web browser.



Your assistant is now live. Explore some of the earlier prompts to verify that the assistant is accessing the ingested documents and your skills and skill flows are active.

**⚠ Wait 5-10 seconds before clicking apply on skill actions.**

Prompts to try:

What is z/OS continuous delivery?

Get z/OS facts

Show me z/OS facts

Gather and display z/OS facts

## Use cases

# Use case: Retrieve IPL information

Next, explore a use case to improve productivity for early-tenure system programmers (SysProg) who are preparing for an upcoming initial program load (IPL) for a logical partition (LPAR).

To prepare for the IPL, the SysProgs need to familiarize themselves with the process. Rather than spending time reading through the wide array of documentation available online, they would like to use watsonx Assistant for Z. The content-grounded capabilities that are provided by watsonx Assistant for Z return accurate responses to their questions quickly and uses automations to perform actions.

As part of the pilot, they already explored prompting the assistant with questions. In one example prompt, they asked the assistant is “*What information is needed to perform an IPL on a z/OS LPAR?*” In reading the response, the SysProg learns they need information about their system in preparation for the IPL. For example, the IPL Volume and the IPL LOAD PARM.

For this use case, show how a simple automation for retrieving this type of information can be infused in a natural conversation with watsonx Assistant for Z. You use pre-packaged skills to automate various tasks on z/OS. The pre-packaged skills are provided as an OpenAPI JSON file. Learn more about OpenAPI [here](#). The file includes skill definitions that can be uploaded to the Skill Studio within watsonx Orchestrate to import the pre-packaged skills. First, the file must first be customized for your z/OS server.

The list of pre-packaged skills available include:

- Authorized program list
- z/OS IPL Information
- Display zOS parmlib datasets
- Unix System services options
- Display zOS subsystems
- List spool files
- Retrieve dataset content
- Retrieve spool file content
- Retrieve z/OS Management facility (z/OSMF) job status

A great value of watsonx Orchestrate is the ability to build skills that anyone can use. You can build your own custom skills by importing an OpenAPI file into watsonx Orchestrate as a JSON or YAML file. For more information on building skills by importing OpenAPI files, refer to the documentation [here](#).

Watsonx Orchestrate also makes it possible to build, edit, and generate OpenAPI specifications by using the OpenAPI builder. With the OpenAPI Builder, you can use the AI function to simplify the process of generating these specifications. For more information on using the OpenAPI Builder, refer to the documentation [here](#).

For this use case, you are importing the skill for retrieving a z/OS server’s IPL information. The next steps walk you through the process of downloading that OpenAPI JSON file and customizing it for your environment.

## Download and customize the watsonx Assistant for Z OpenAP JSON file

1. Download the watsonx Assistant for Z OpenAP JSON file.

[wxa4z-skillpak-prepackaged-skills.json.zip](#)

2. Extract the file.

3. In a text editor, open the `wxa4z-skillpak-prepackaged-skills.json`, modify the server's url field as described, and save the file.

**File name:**

```
wxa4z-skillpak-prepackaged-skills.json
```

Substitute your Wazi as a Service (WaaS) instance URL for the string `<your z/OSMF URL>`. Your WaaS URL is based on your ITZ AAP URL that can be obtained from your watsonx Assistant for Z Pilot - AAP & z/OS ITZ reservation.

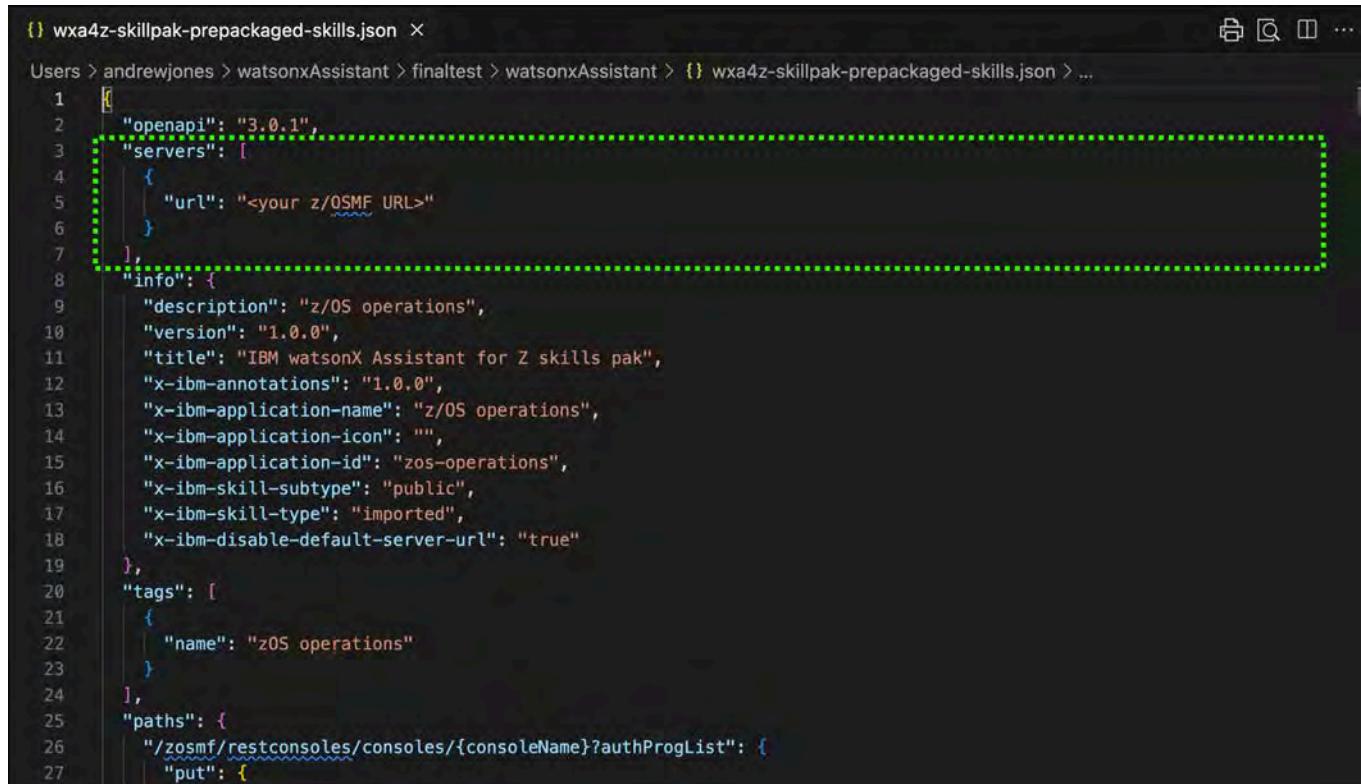
The AAP URL is similar to:

<https://itzvsi-aap-ppxbcno.techzone.ibm.com>

Change the `aap` string to `zos` and append `:10443` to the URL value. **Record this value for later use!** Your new URL is similar to:

<https://itzvsi-zis-ppxbcno.techzone.ibm.com:10443>

Before:



```
(1) wxa4z-skillpak-prepackaged-skills.json ×
Users > andrewjones > watsonxAssistant > finaltest > watsonxAssistant > {} wxa4z-skillpak-prepackaged-skills.json > ...
1  [
2    "openapi": "3.0.1",
3    "servers": [
4      {
5        "url": "<your z/OSMF URL>"  This line is highlighted by a green dotted rectangle.
6      }
7    ],
8    "info": {
9      "description": "z/OS operations",
10     "version": "1.0.0",
11     "title": "IBM watsonX Assistant for Z skills pak",
12     "x-ibm-annotations": "1.0.0",
13     "x-ibm-application-name": "z/OS operations",
14     "x-ibm-application-icon": "",
15     "x-ibm-application-id": "zos-operations",
16     "x-ibm-skill-subtype": "public",
17     "x-ibm-skill-type": "imported",
18     "x-ibm-disable-default-server-url": "true"
19   },
20   "tags": [
21     {
22       "name": "zos operations"
23     }
24   ],
25   "paths": {
26     "/zosmf/restconsoles/consoles/{consoleName}?authProgList": {
27       "put": {
```

After:

```

{} wxa4z-skillpak-prepackaged-skills.json •
Users > andrewjones > watsonxAssistant > finaltest > watsonxAssistant > {} wxa4z-skillpak-prepackaged-skills.json > {} info
1  {
2    "openapi": "3.0.1",
3    "servers": [
4      {
5        "url": "https://itzvsi-zos-pwgabob.techzone.ibm.com:10443"
6      }
7    ],
8    "info": {
9      "description": "z/OS operations",
10     "version": "1.0.0",
11     "title": "IBM watsonX Assistant for Z skills pak",
12     "x-ibm-annotations": "1.0.0",
13     "x-ibm-application-name": "z/OS operations",
14     "x-ibm-application-icon": "",
15     "x-ibm-application-id": "zos-operations",
16     "x-ibm-skill-subtype": "public",
17     "x-ibm-skill-type": "imported",
18     "x-ibm-disable-default-server-url": "true"
19   },
20   "tags": [
21     {
22       "name": "zOS operations"
23     }
24   ],
25   "paths": {
26     "/zosmf/restconsoles/consoles/{consoleName}?authProgList": {

```



In watsonx Orchestrate, each app is associated with a single URL.

If you have not imported previous skills into an app called `z/OS operations`, the default values in the `info` section are fine. If you already have an app named "z/OS operations", modify the `info` section to meet your needs.

For more information on modifying the OpenAPI JSON file, see the instructions [here](#).

## Set the RACF passphrase for z/OSMF authentication

For your skills to run successfully on your WaaS instance, you need to ensure that you can authenticate to it from watsonx Orchestrate. To achieve successful authentication, setup a new RACF Passphrase for the `IBMUSER` ID that is a pre-defined user ID on the WaaS server. The following steps take you through the steps of setting a new passphrase for your user and verifying access.

1. Open and log in to the Ansible Automation Platform (AAP) web console.



**Don't remember how?**

Refer to the first 5 steps in [Explore Ansible Automation Platform](#).

2. Click **Templates** under the **Resources** section.

Name	Type	Organization	Last Ran	Actions
z/OS Certs - Add CA Cert	Job Template	Default	1/17/2025, 5:17:58 AM	
z/OS Certs - Create Cert	Job Template	Default		
z/OS Certs - Create Keyring	Job Template	Default		
z/OS Certs - Delete Cert	Job Template	Default		
z/OS Certs - Delete Keyring	Job Template	Default		

3. Click the launch icon () for the **z/OS TSO Command(s)** template.

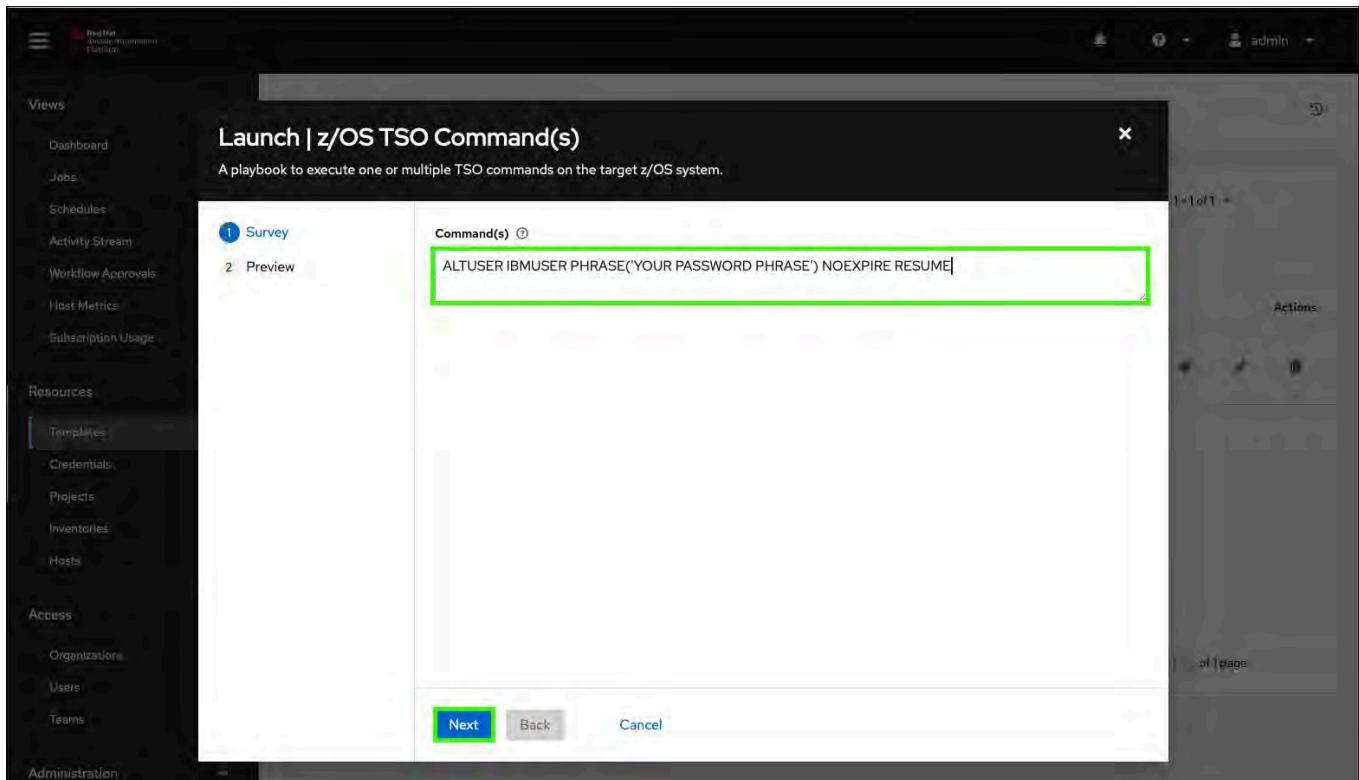
Name	Type	Organization	Last Ran	Actions
z/OS TSO Command(s)	Job Template	Default		

4. Replace the default command with the text that follows and substituting a password of your choosing for the string `YOUR PASSWORD PHRASE` and then click **Next**.

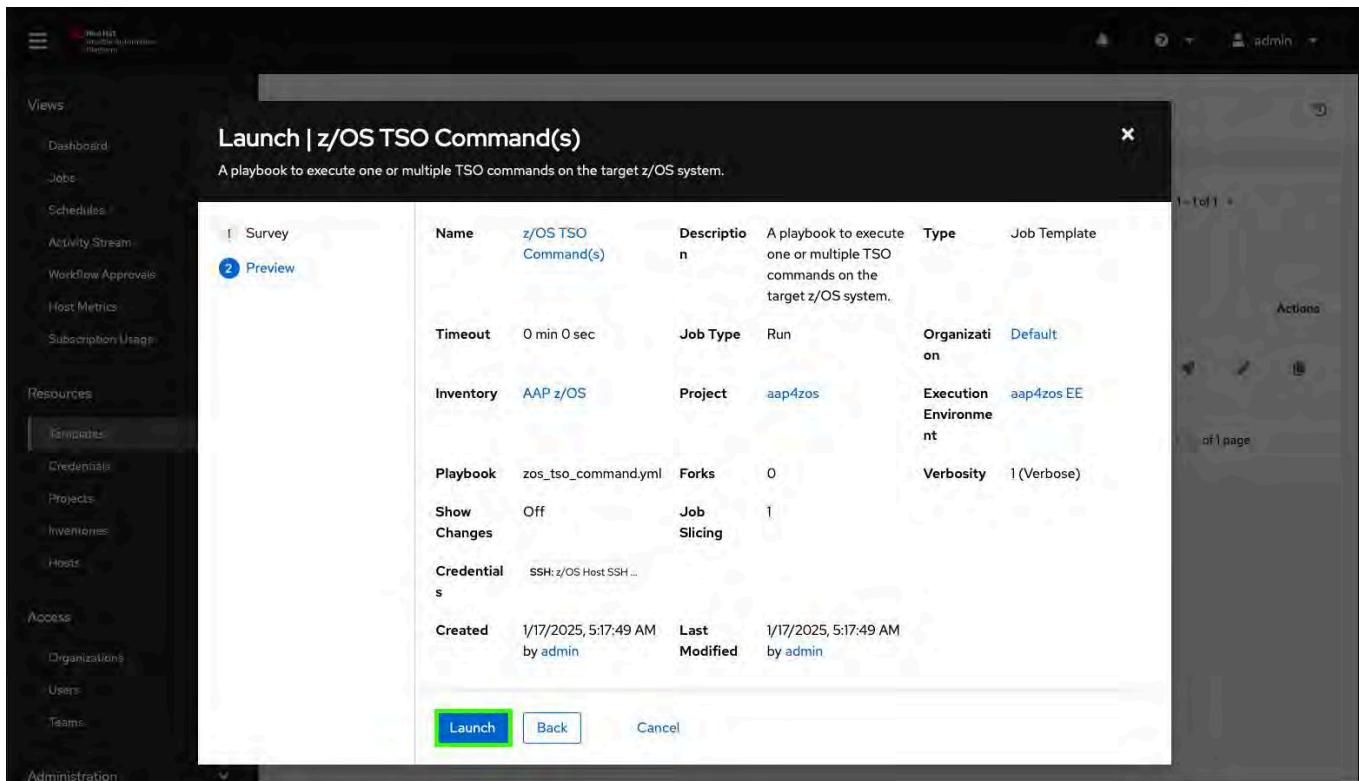
```
ALTUSER IBMUSER PHRASE('YOUR PASSWORD PHRASE') NOEXPIRE RESUME
```

**Avoid typographical errors later... keep the password simple.**

If you type the command yourself, be sure to include the single quotes before and after the password. **Record the password as it will be needed later.** We will refer to this as your **WaaS password**."



## 5. Click Launch.



## 6. Verify that the job is **Successful** by locating the message "failed": false in the job output.

The screenshot shows the Red Hat Ansible Automation Platform interface. On the left, there's a sidebar with 'Views' (Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals, Host Metrics, Subscription Usage), 'Resources' (Templates, Credentials, Projects, Inventories, Hosts), 'Access' (Organizations, Users, Teams), and 'Administration' (Credential Types, Notifications). The main area is titled 'Jobs > 13 - z/OS TSO Command(s)' and has tabs for 'Output' (selected), 'Details', and 'Back to Jobs'. It shows a 'z/OS TSO Command(s)' job with a status of 'Successful'. The output tab displays 'Stdout' with the following content:

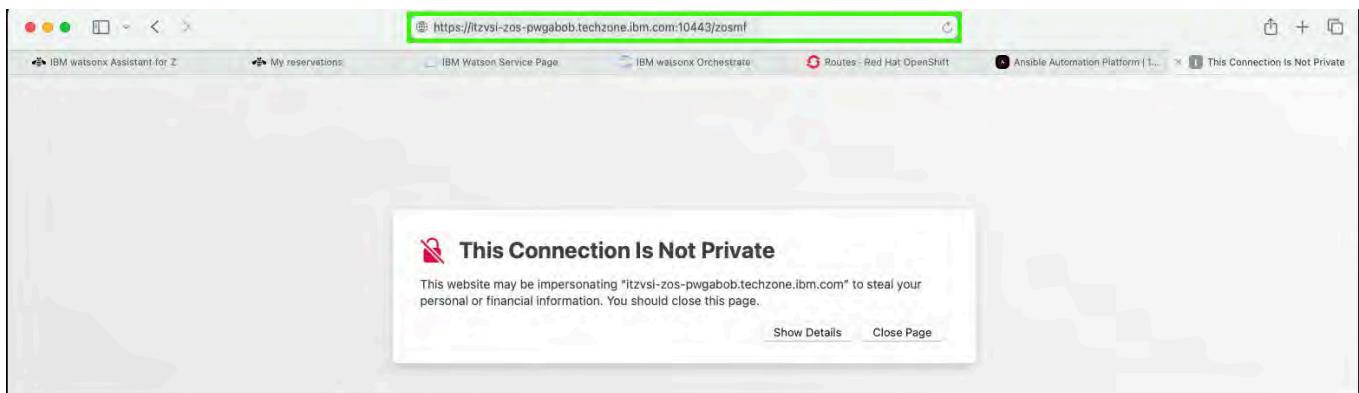
```

Events processing complete. Reload output
5 TASK [Run z/OS TSO command(s).] *****
6 changed: [zos_host] => {"changed": true, "max_rc": 0, "output": [{"command": "ALTUSER IBMUSER PHRASE('YOUR PASSWORD PH RASE') NOEXPIRE RESUME", "content": "", "failed": false, "lines": 1, "rc": 0, "stderr": ""}]}
7
8 TASK [Print results from command(s). Click on the output and go to the 'JSON' tab to show more output.] ***
9 ok: [zos_host] => {
10     "command.output": [
11         {
12             "command": "ALTUSER IBMUSER PHRASE('YOUR PASSWORD PHRASE') NOEXPIRE RESUME",
13             "content": "",
14             "failed": false,
15             "lines": 1,
16             "rc": 0,
17             "stderr": ""
18         }
19     ]
20 }
21
22 }
23
24 ]

```

## 7. Verify that you can log in to z/OSMF in a new browser tab.

Use the WaaZ URL created earlier and append `/zosmf` to the string. The URL is similar to: <https://itzvsi-zos-pwgabob.techzone.ibm.com:10443/zosmf>.



**Accept any connection not private messages to open the page.**

## 8. Enter (a) IBMUSER for the z/OS USER ID, (b) the password you specified in step 4 for the z/OS PASSWORD, and then (c) click LOG IN.

IBM z/OS Management Facility

LEARN MORE NEED HELP?

## Welcome to z/OS

The highly secure, scalable and resilient enterprise operating system for the IBM z Systems mainframe.

**a** z/OS USER ID  
IBMUSER

**b** z/OS PASSWORD  
.....

**c** LOG IN

Shopz. z Systems Redbooks WSC Flashes and Techdocs IBM z/OS documentation

IBM Support z/OSMF Home Page z/OS Home Page

9. Close the new browser tab after verifying a screen similar to the image that follows is displayed.



Import and publish the pre-packaged skills as an OpenAPI file

1. Open **Skills studio** in watsonx Orchestrate.

Welcome, Andrew Jones!

Take productivity to the next level.

Intuitive interaction | Natural conversations | Contextual clarity

Try Skills in Chat →

**Build**

Start building the skills, conversations, and automations your team needs here.

**AI assistant builder**

Boost productivity and customer care by creating conversational experiences.

**Skill studio**

Build the skills your team needs to get their work done more quickly.

Standard plan | Skill studio

Automate how your business works

Learn more →

**Explore**

2. Click **Create** and then click **Import API**.

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

**Skills and apps** Projects

Skills Apps

Find a skill

**Create**

**Import API**

Skill Flow

Configure prebuilt skills

3. Click **From a file**.

**Add skills**

Choose how you want to add skills and then select the skills you want to refer to from that source.

Choose the source       Select the skills

**Choose the source**

To discover new skills, connect to an app or refer to an OpenAPI file.

**From an app**    **From a file**    **OpenAPI builder (exp...)**

**IBM Cloud Pak for Business Automation - On premises**  
which includes Automation Decision Services, Business Automation Workflow, and Operational Decision Manager on-premises

**IBM Cloud Pak for Business Automation as a service**  
which includes Business Automation Workflow as a service and includes Operational Decision Manager as a service

**z/OS Skills accelerator (Trial)**  
Import your Ansible JCL/REXX automation to Watsonx Orchestrate

#### 4. Click Drag and drop files here or click to upload in the Import a skill file window.

**Skills and apps / Add skills**

**Add skills**

Choose how you want to add skills and then select the skills you want to refer to from that source.

Choose the source       Select the skills

**Choose the source**

To discover new skills, connect to an app or refer to an OpenAPI file.

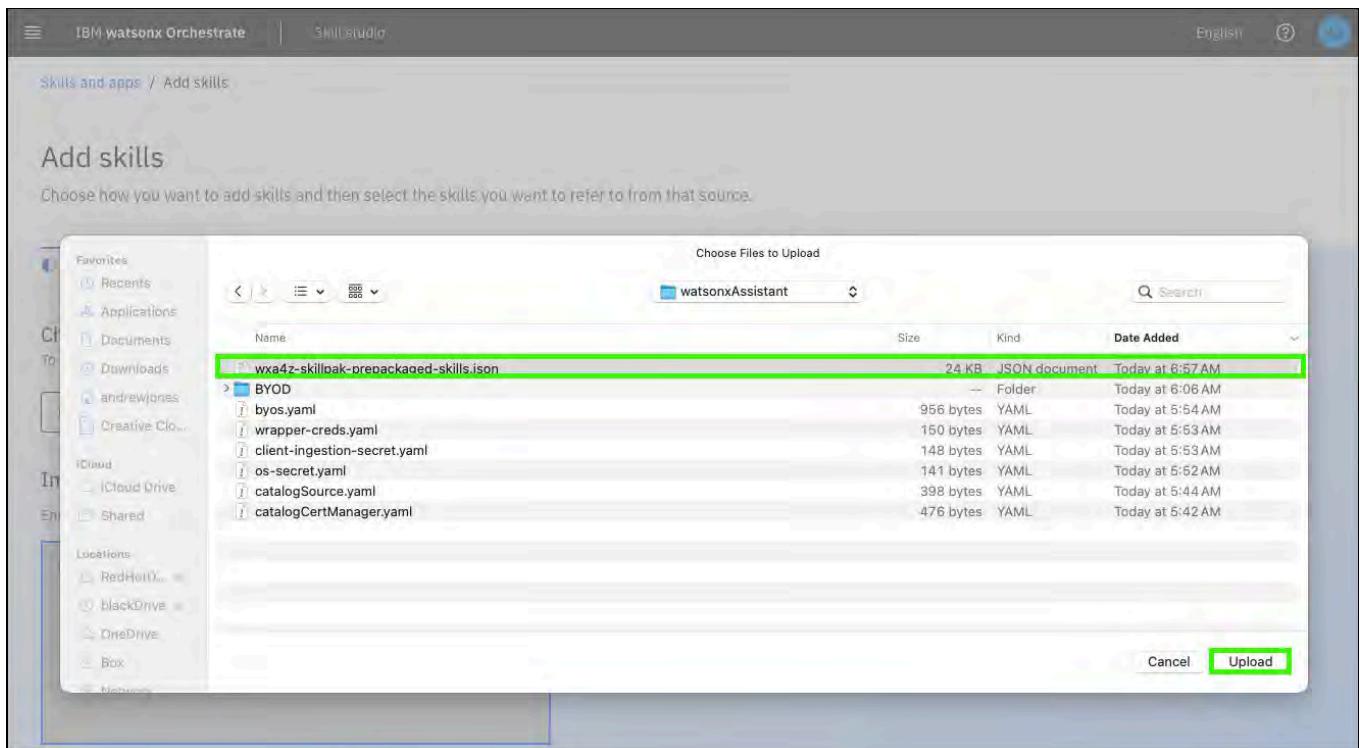
**From an app**    **From a file**    **OpenAPI.builder (exp...)**

**Import a skill file**

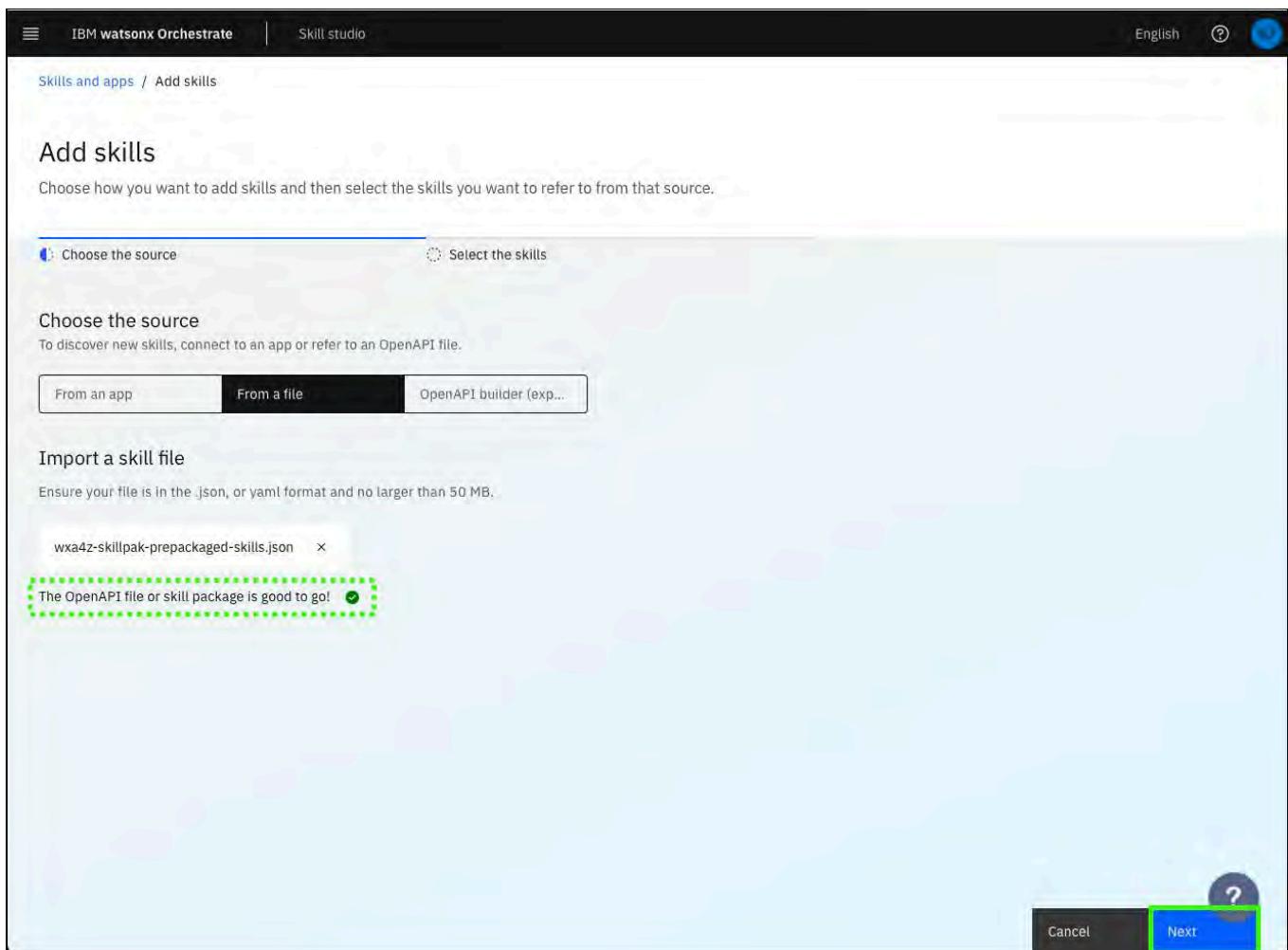
Ensure your file is in the .json, or yaml format and no larger than 50 MB.

Drag and drop files here or click to upload

#### 5. Locate and select the JSON file that you modified earlier and then click **Upload**.



6. Verify that you receive the message **The Open file or skill package is good to go!** and then click **Next**.

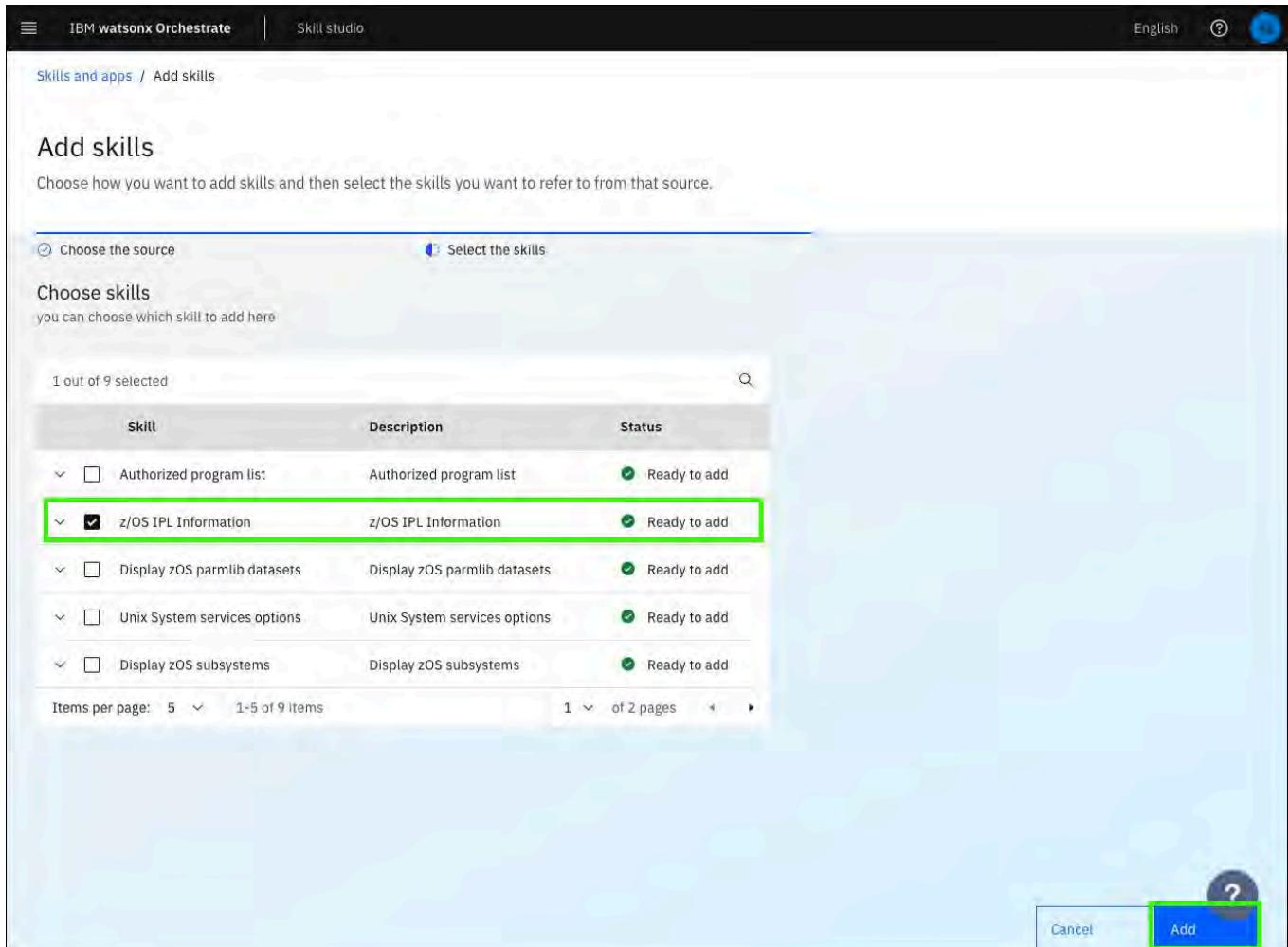


### ✖️ Not good to go?

If the file does not load properly you will need to verify not formatting or errors were made in your json file. Return to the previous section to verify the file contents and then reload the JSON file.

#### 7. Select the **z/OS IPL Information** skill and then click **Add**.

**Note:** Only the **z/OS IPL Information** skill is required for this use case, but you can add as many skills as you like for testing purposes.



The screenshot shows the 'Skill studio' interface for 'Skills and apps / Add skills'. At the top, there are tabs for 'Choose the source' (radio button) and 'Select the skills' (radio button, which is selected). Below this, a section titled 'Choose skills' with the sub-instruction 'you can choose which skill to add here' is shown. A table lists five skills:

Skill	Description	Status
<input type="checkbox"/> Authorized program list	Authorized program list	<span style="color: green;">✓ Ready to add</span>
<input checked="" type="checkbox"/> z/OS IPL Information	z/OS IPL Information	<span style="color: green;">✓ Ready to add</span>
<input type="checkbox"/> Display zOS parmlib datasets	Display zOS parmlib datasets	<span style="color: green;">✓ Ready to add</span>
<input type="checkbox"/> Unix System services options	Unix System services options	<span style="color: green;">✓ Ready to add</span>
<input type="checkbox"/> Display zOS subsystems	Display zOS subsystems	<span style="color: green;">✓ Ready to add</span>

At the bottom of the table, there are pagination controls: 'Items per page: 5' and '1-5 of 9 items'. On the far right, there are buttons for 'Cancel' and 'Add', with the 'Add' button being highlighted by a green box.

#### 8. Click the ellipses icon (~⋮~) for the **z/OS IPL Information** skill and then click **Enhance this skill**.

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

**Create**

**Skills and apps** Projects

**Skills** Apps

Find a skill

**Skills**

Name	Step in the process	Status	Skill type	Author	Last edited	Actions
z/OS IPL Information	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	January 17 2025	<span>⋮</span>
Zeeves-gather-facts-flow	Ready to use	Published	Skill flow	andrew@jones-tx.com	January	<span>Enhance this skill</span>
Retrieve job output	Ready to use	Published	Imported	andrew@jones-tx.com	January	<span>Export this skill</span>
z/OS Ping	Ready to use	Published	Imported	andrew@jones-tx.com	January	<span>Delete this skill</span>
z/OS Gather Facts	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025	<span>⋮</span>

## 9. Review the skill enhancements options and then click Publish.

Skills and apps / Enhance this skill

Enhance the “z/OS IPL Information” skill

Add details that will make people want to use this skill.

**Name**  Input Output Security Phrases Next best skills

Name and describe this skill in a way that tells users how it's used and why they would want to use it.

Name\*

Description  20/100

API version\*  ×

Categories

App

**Preview**

The skill will look like this in the catalog.

**z/OS IPL Information**  
z/OS IPL Information

The skill will look like this in the skill set.

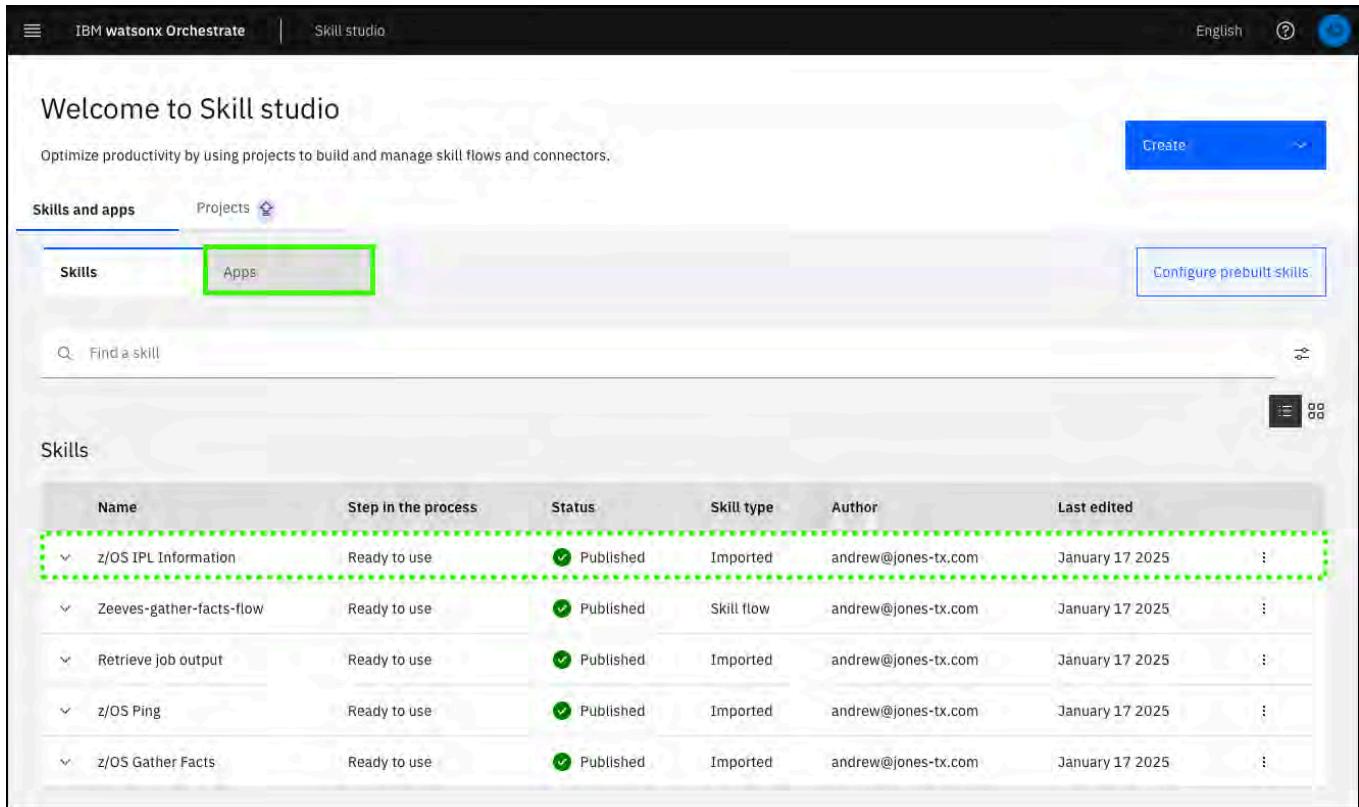
**z/OS IPL Information**

**Publish**

10. If you added other skills in step 7, repeat the previous 2 steps for each skill added.

Configure your appURL to connect to the WaaS server

1. In **Skill studio**, click the **Apps** tab.



The screenshot shows the 'Skill studio' section of the IBM Watsonx Orchestrate interface. At the top, there's a navigation bar with 'IBM Watsonx Orchestrate', 'Skill studio', 'English', and a help icon. Below the navigation is a header 'Welcome to Skill studio' with a sub-instruction 'Optimize productivity by using projects to build and manage skill flows and connectors.' To the right of the header is a 'Create' button and a dropdown menu. Below the header, there are two tabs: 'Skills' and 'Apps', with 'Skills' currently selected. A green box highlights the 'Apps' tab. To the right of the tabs is a 'Configure prebuilt skills' button. Below the tabs is a search bar with the placeholder 'Find a skill'. Underneath the search bar is a section titled 'Skills' with a table. The table has columns: Name, Step in the process, Status, Skill type, Author, and Last edited. A green dashed box highlights the first row of the table, which contains the skill 'z/OS IPL Information'. The table lists five skills in total:

Name	Step in the process	Status	Skill type	Author	Last edited
z/OS IPL Information	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025
Zeeves-gather-facts-flow	Ready to use	Published	Skill flow	andrew@jones-tx.com	January 17 2025
Retrieve job output	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025
z/OS Ping	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025
z/OS Gather Facts	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025

2. In the search field, enter the name of the application in the search field. Unless you modified the `info` section of the JSON file, the default name is **z/OS operations**.

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

Create

Skills and apps Projects

Skills Apps

Configure prebuilt skills

z/OS operations

Select an app to configure the settings for all the app's skills.

Name	Description	Configuration status
Workday HCM		Not Configured
Salesloft		Not Configured
Adobe Workfront		Not Configured
Webex		Not Configured
ZoomInfo		Not Configured

- Click the ellipses icon (~) for the z/OS Operations and then click Edit.

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

Create

Skills and apps Projects

Skills Apps

Configure prebuilt skills

z/OS operations

Select an app to configure the settings for all the app's skills.

Name	Description	Configuration status
z/OS operations	z/OS IPL Information	Not Configured

- Click the Configuration tab.

Skills and apps / Enhance this skill

## Configure the “z/OS operations” app

**Details** Configuration

Change the app's name and icon to whatever makes more sense for your organization. Be sure to provide the connection information.

Name\*  
z/OS operations

Description\*  
z/OS IPL Information

0/100

App icon

**Preview**

The app will look like this in the skill set.

**z/OS operations**  
1 skill

The app will look like this in the skill set.

**z/OS operations**  
1 skill

### 5. Click **Test connection**.

**Note:** the **Server URL** field must match the URL you created for your WaaS server. It is similar to:

<https://itzvsi-zos-pwgabob.techzone.ibm.com:10443> .

Skills and apps / Enhance this skill

## Configure the “z/OS operations” app

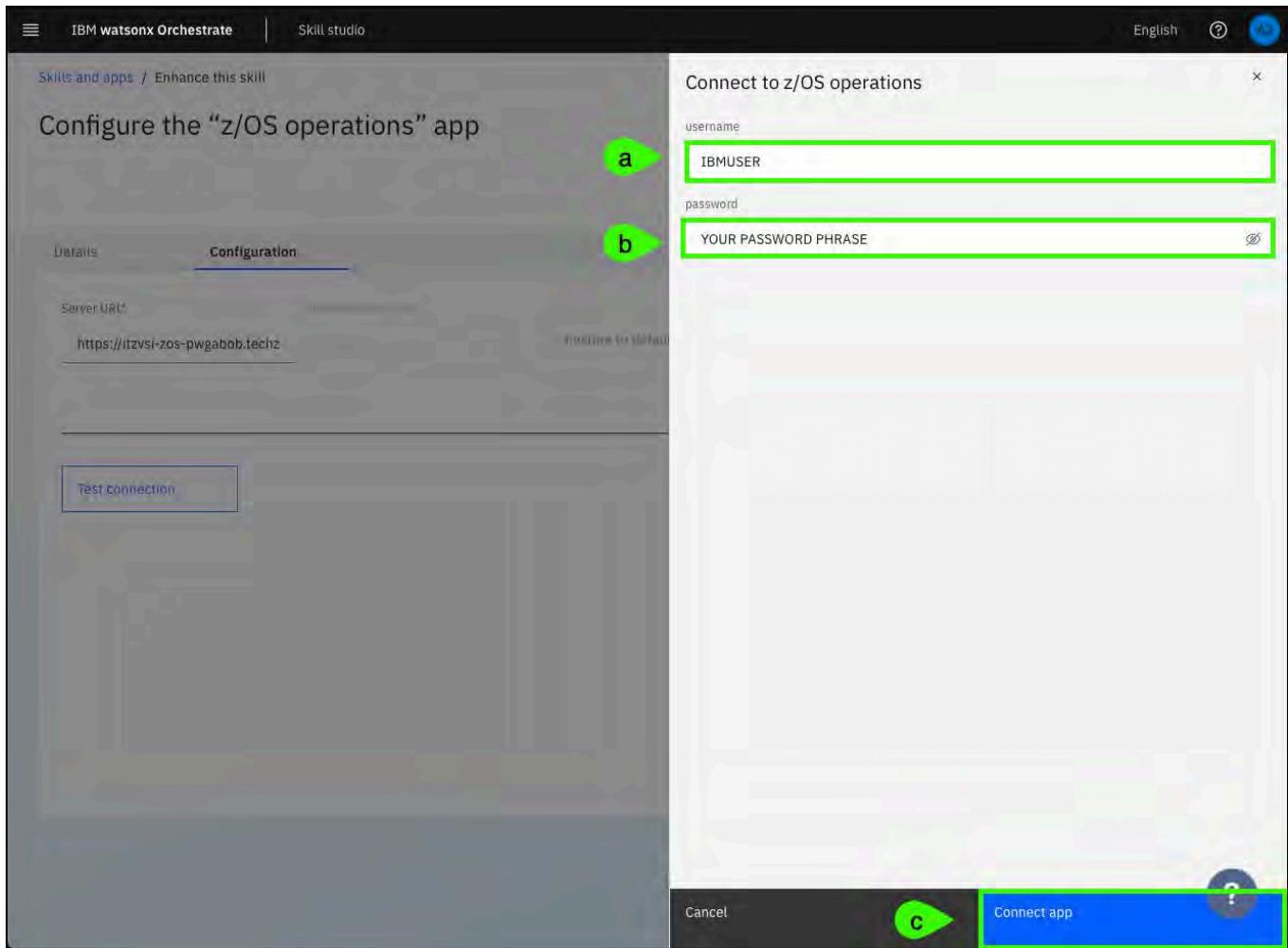
**Details** Configuration

Server URL\*  
https://itzvsi-zos-pwgabob.techz

Restore to default ↻

**Test connection**

### 6. Enter (a) **IBMUSER** in the **username** field, (b) your **WaaS password** created earlier in the **password** field, and then (c) click **Connect app**.



7. Verify that the connection is successful and then click **Save**.

Skills and apps / Enhance this skill

## Configure the “z/OS operations” app

**Configuration**

Server URL\*  Authentication

Test connection

Connection successful. Valid URL  
08:58:14

Cancel Save

8. Repeat step 2 above to verify that the **Configuration status** is **Configured**.

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

Skills and apps Projects

Skills Apps

Configure prebuilt skills

z/OS operations

Select an app to configure the settings for all the app's skills.

Name	Description	Configuration status
z/OS operations	z/OS IPL Information	Configured

Test the skill

1. Open **Chat** in watsonx Orchestrate.

The screenshot shows the IBM Watsonx Orchestrate application. The left sidebar has a dark theme with white text and icons. The 'Chat' option is highlighted with a green border. Other options in the sidebar include 'Skill sets', 'Skill catalog', 'AI assistant builder', and 'Skill studio'. The main area is titled 'Skill studio' and contains a sub-section titled 'Configure prebuilt skills'. A table is present with two columns: 'Description' and 'Configuration status'. One row in the table is for 'z/OS IPL Information' with a status of 'Configured' indicated by a green checkmark.

2. Click **Add skills from the catalog**.

The screenshot shows the 'Personal skills' catalog interface. At the top, there's a message from Watson: 'Hi. I'm Watson. Check out the skills in the catalog to see how I can help you.' Below this is a search bar with the placeholder 'Tell me what you want to do...'. On the left, a button labeled 'Add skills from the catalog' is highlighted with a green box. To its right, a card displays 'Ansible Controller Skills - z skills' with a small icon and a '2 skills' badge. A large question mark icon is in the bottom right corner.

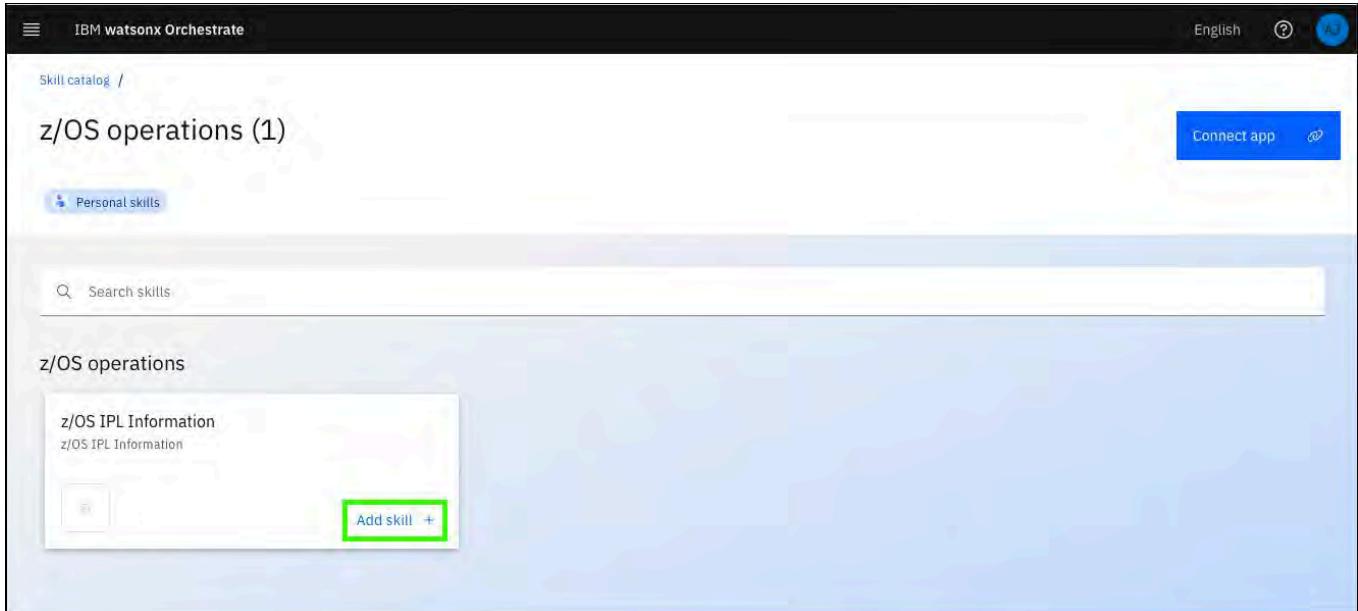
3. In the search apps field, enter the name of the application. Unless you modified the `info` section of the JSON file, the default name is **z/OS operations**.

The screenshot shows the 'Skill catalog' page in IBM Watsonx Orchestrate. At the top, there's a search bar with the text 'z/OS operations' highlighted by a green box. Below the search bar, the heading 'Most popular skills' is displayed. Under this heading, there are three cards: 'Send an email from Gmail' (using the Gmail icon), 'Create a lead in Salesforce' (using the Salesforce icon), and 'Send an email using Outlook' (using the Microsoft Outlook icon). Each card has an 'Add skill +' button. Below this section, the heading 'All Apps' is shown, followed by a grid of app tiles. The tiles include: Coupa (23 skills), ZoomInfo (26 skills), HubSpot CRM (33 skills), Apptio Targetprocess (24 skills), Salesforce (85 skills), Zendesk Service (25 skills), Calendly (6 skills), and Square (8 skills).

4. Locate and click the tile for your app (z/OS operations).

This screenshot shows the same 'Skill catalog' page as the previous one, but with a different focus. The 'z/OS operations' tile is now highlighted with a green box. This tile is located in the bottom row of the grid under the 'All Apps' heading. It shows the z/OS icon, the text 'z/OS operations', and '1 skill'. The rest of the grid contains other app tiles like Microsoft Dynamics 365, Slack, Box, Google Contacts, Jira, Microsoft OneDrive, Marketo, monday.com, Amazon SES, Salesforce Commerce, DocuSign, Google Calendar, Seismic, Microsoft Azure Blob storage, Google Cloud Storage, UKG, GitHub, Google Drive, Dropbox, Amazon SNS, Salesforce Chatter, bambooHR, AWS Lambda, Webex, Adobe Workfront, Cognos, Ansible Controller Skills, Salesloft, and Gmail.

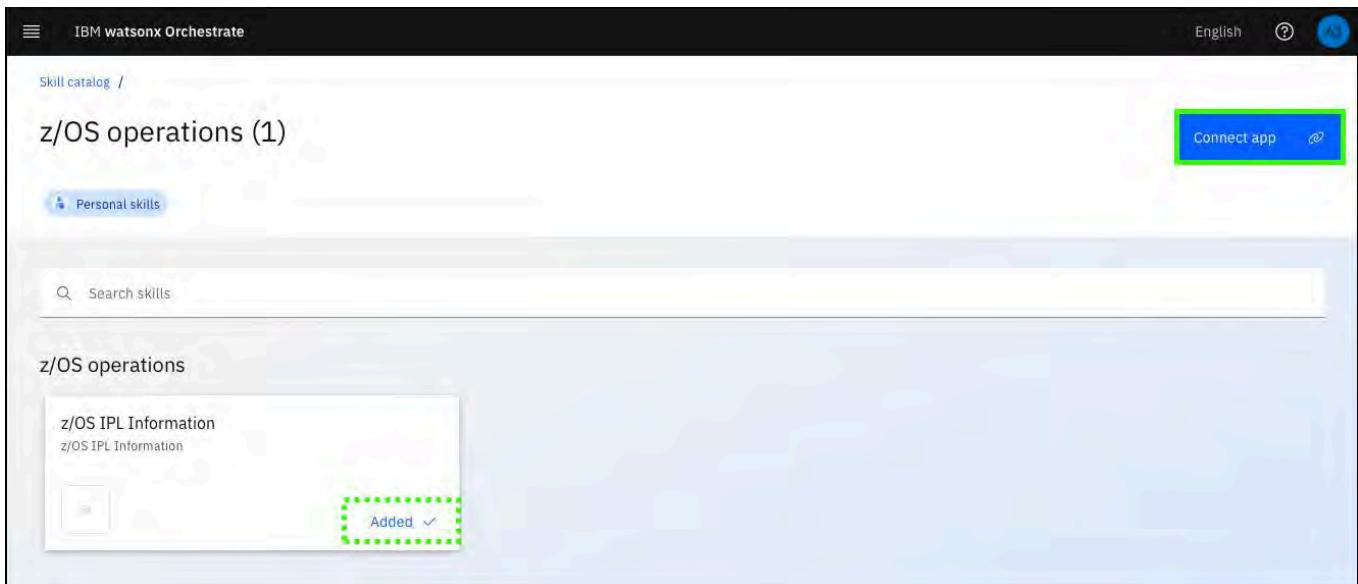
5. Click **Add skill +** for the **z/OS IPL Information** skill.



The screenshot shows the IBM Watsonx Orchestrate interface. At the top, there's a navigation bar with 'Skill catalog /' and language options 'English', '?', and 'AI'. Below the navigation is a search bar labeled 'Search skills'. Under the heading 'z/OS operations (1)', there's a single skill card for 'z/OS IPL Information'. To the right of the skill name is a small icon and a green-bordered 'Add skill +' button. The entire screenshot is framed by a light blue border.

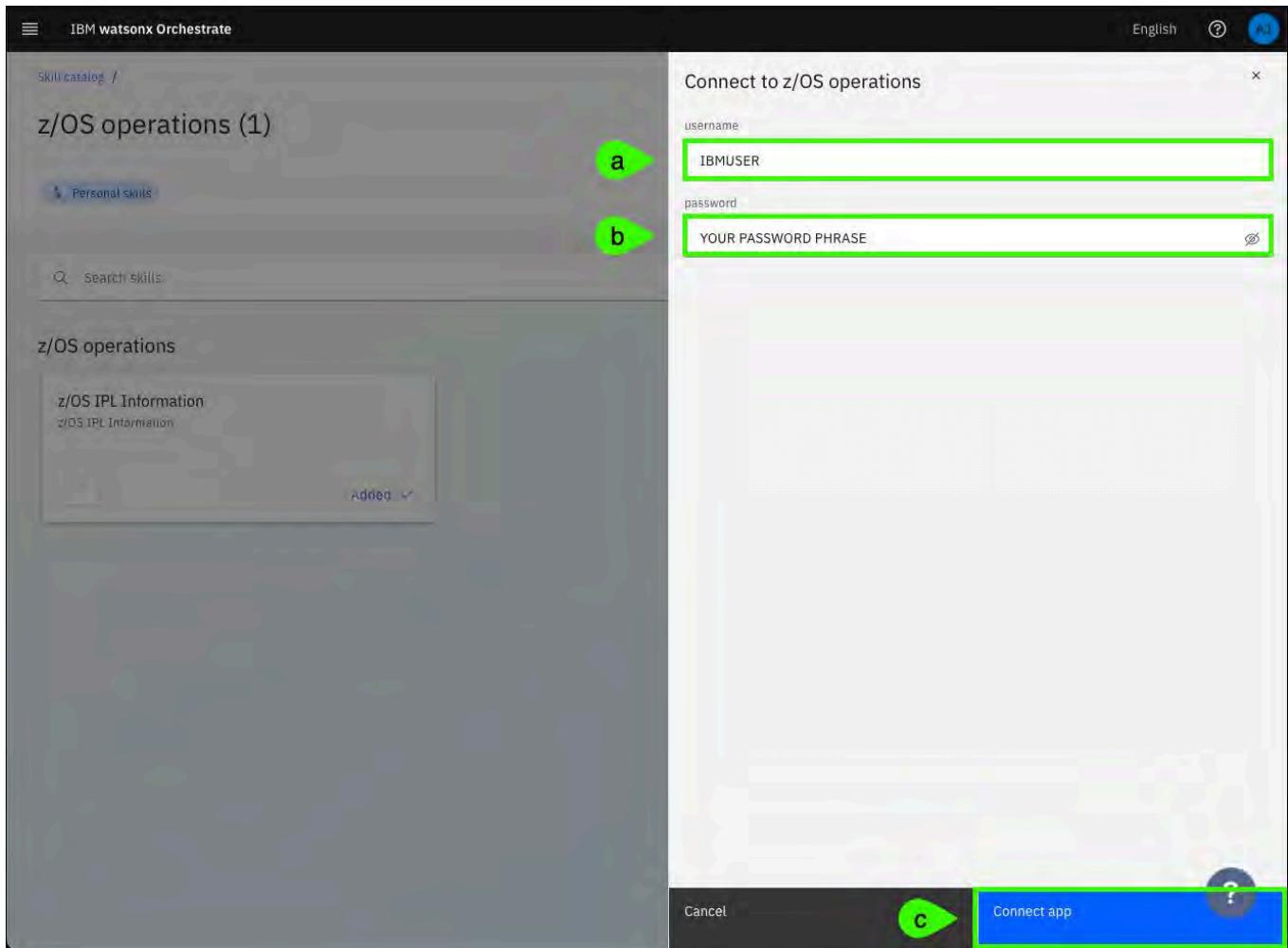
6. Repeat step 5 for any additional skills you added to the app.

7. Click **Connect app**.

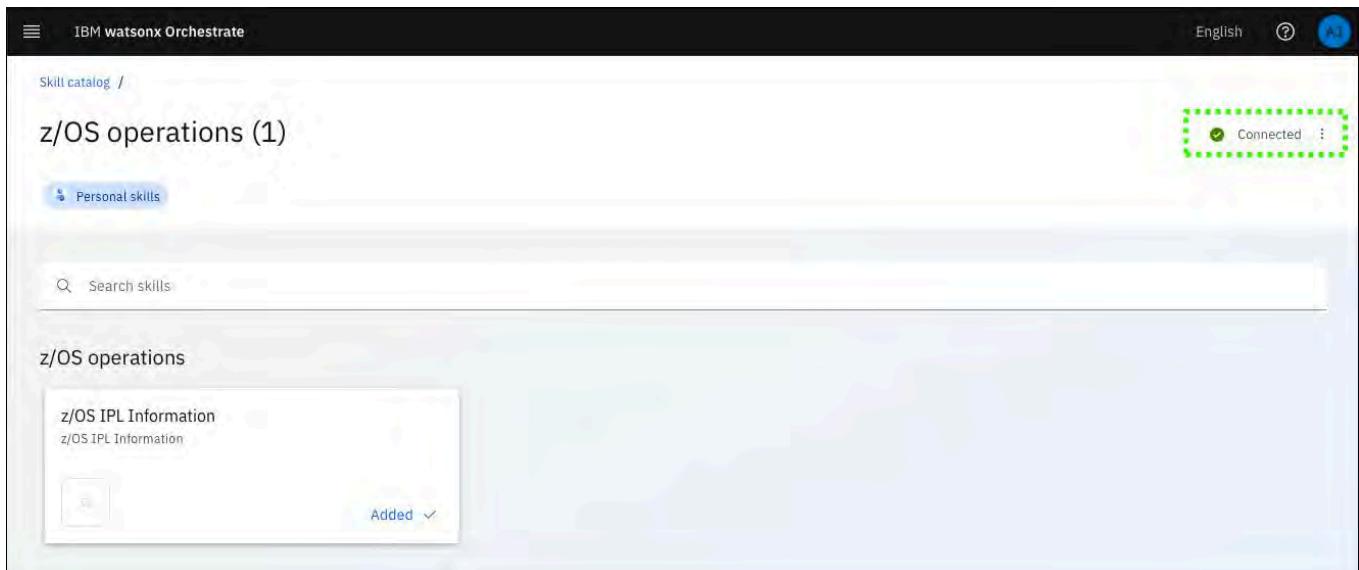


This screenshot is similar to the previous one, showing the 'z/OS operations (1)' skill list. The 'z/OS IPL Information' skill card now has a green dashed border around it, and the word 'Added' with a checkmark is displayed below the 'Add skill +' button. The 'Connect app' button at the top right is also highlighted with a green border. The overall layout is consistent with the first screenshot, including the navigation bar and search bar.

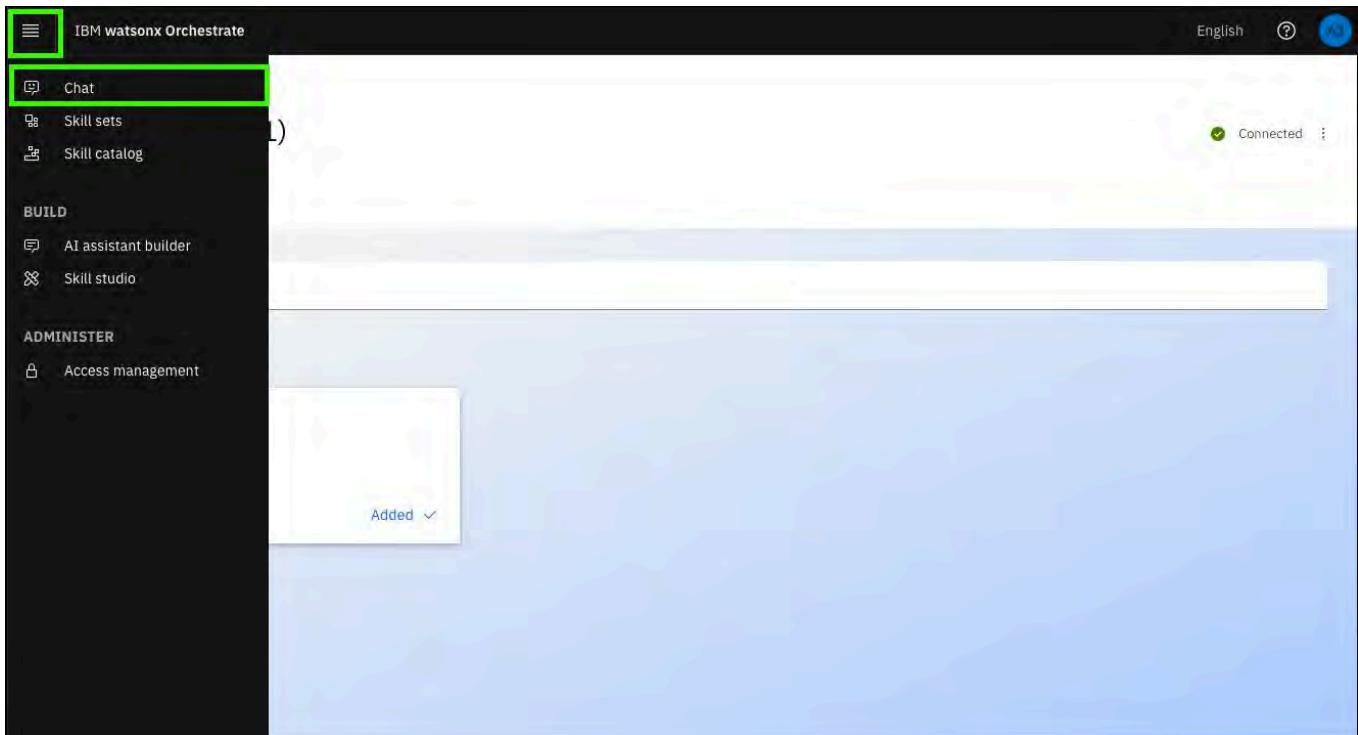
8. Enter (a) **IBMUSER** in the **username** field, (b) your **WaaS password** created earlier in the **password** field, and then (c) click **Connect app**.



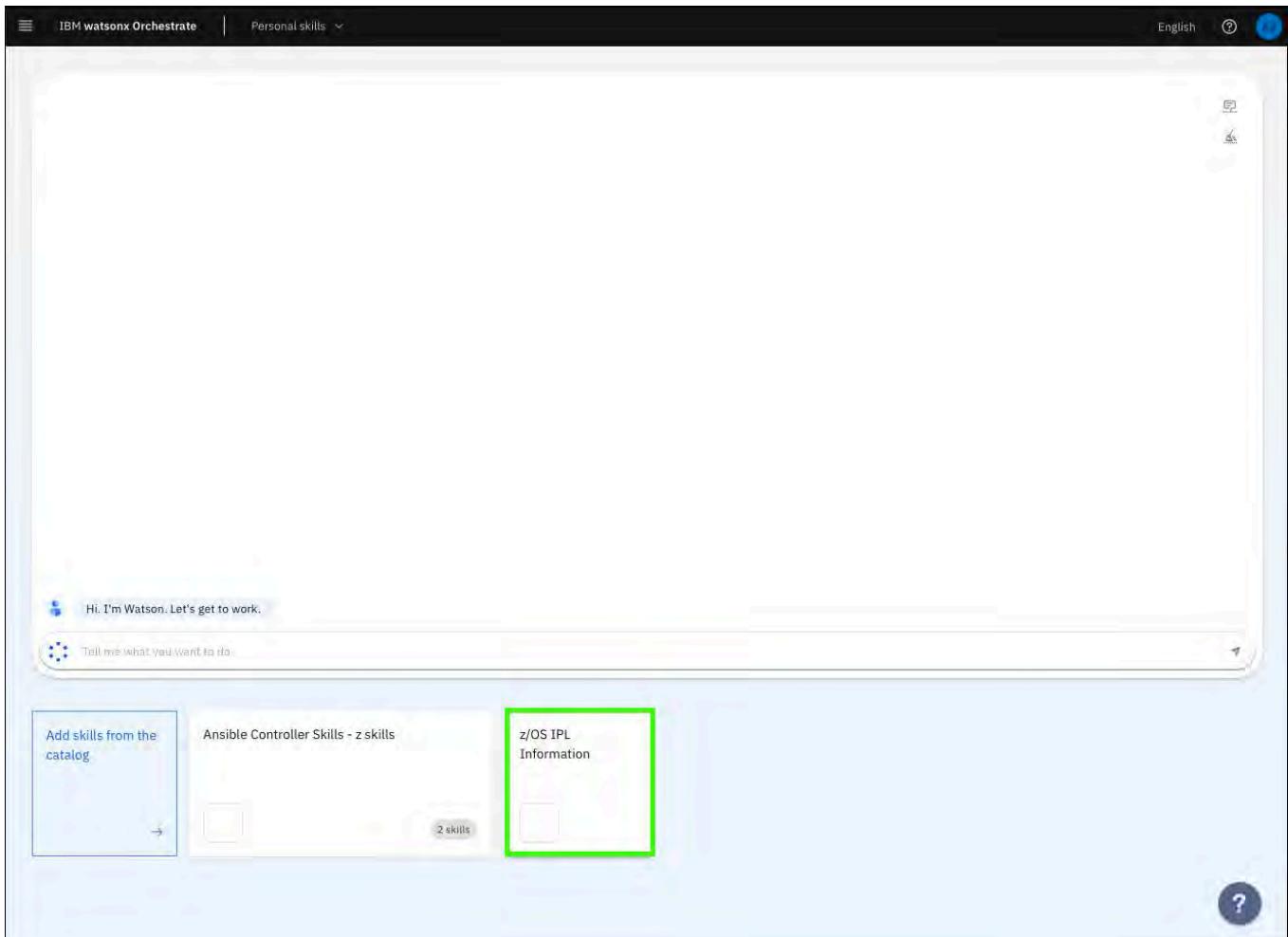
9. Verify that the skill is connected.



10. Open **Chat** in watsonx Orchestrate.



11. Click the **z/OS IPL Information** tile.



**The name of the tile may be different.**

If you added multiple skills to your app, the tile name may be **z/OS operations** and the number of skills included will be shown.

12. Using the defaults in the form, click **Apply**.

Hi, I'm Watson. Let's get to work.

**z/OS IPL Information**

You just need to complete this form first.

z/OS IPL Information

consoleName \*

iserVS01

cmd \*

d iplinfo

sol-key \*

JES

Cancel **Apply**

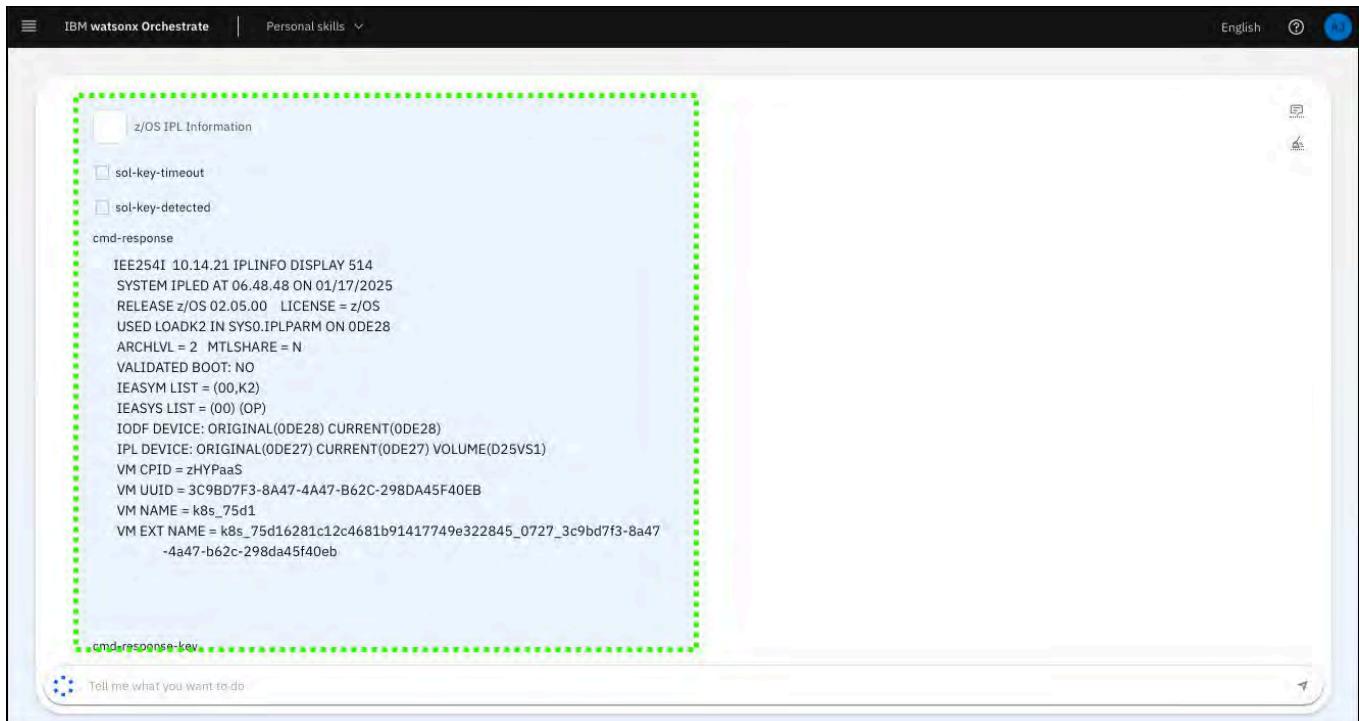
Tell me what you want to do

Add skills from the catalog → Ansible Controller Skills - z skills 2 skills z/OS IPL Information

13. Review the returned information.

The provided output shows information the early-tenure SysProg needs to prepare for an IPL on their z/OS LPAR. For example, the date and time the system was last IPL'ed, the z/OS release level, the IPL volume, the IPL LOAD PARM used during the IPL, and other details.

If you loaded other pre-packaged skills, test them now.



The screenshot shows the IBM Watsonx Orchestrate interface. In the top navigation bar, 'IBM Watsonx Orchestrate' and 'Personal skills' are visible, along with language settings for 'English'. The main content area displays the 'z/OS IPL Information' skill. A green dashed box highlights the 'cmd-response' section, which contains the following text:

```

IEE2541 10.14.21 IPLINFO DISPLAY 514
SYSTEM IPLED AT 06.48.48 ON 01/17/2025
RELEASE z/OS 02.05.00 LICENSE = z/OS
USED LOAD2 IN SYS0.IPLPARM ON ODE28
ARCHLVL = 2 MTLSHARE = N
VALIDATED BOOT: NO
IEASYM LIST = (00,K2)
IEASYS LIST = (00) (OP)
IODF DEVICE: ORIGINAL(ODE28) CURRENT(ODE28)
IPL DEVICE: ORIGINAL(ODE27) CURRENT(ODE27) VOLUME(D25VS1)
VM CPID = zHYPaaS
VM UUID = 3C9BD7F3-8A47-4A47-B62C-298DA45F40EB
VM NAME = k8s_75d1
VM EXT NAME = k8s_75d16281c12c4681b91417749e322845_0727_3c9bd7f3-8a47
-4a47-b62c-298da45f40eb

```

At the bottom of the skill details, there is a 'cmd-response-key' link.

## Connect the app to your assistant

Before configuring the **z/OS IPL Information** skill as an assistant action, the app containing the skill must first be connected to the assistant.

### 1. Open Skill sets in watsonx orchestrate.



### 2. Click the **Skill sets** drop-down list and select the **Draft** of your assistant.

The screenshot shows the 'Team Skills' page in the IBM Watsonx Orchestrate interface. At the top, there's a navigation bar with the title 'IBM Watsonx Orchestrate' and language settings ('English'). Below the title, the heading 'Team Skills' is displayed. A sidebar on the left lists 'Skill sets' with 'Team Skills' selected. Under 'Team Skills', there are two items: 'Orchestrate Agent Skillset' and 'Zeeves draft'. The 'Zeeves draft' item is highlighted with a green border. A tooltip for 'Zeeves draft' states: 'The skills added to this team are available to this team's members. These skills complete related tasks for the team members so members can work more efficiently and effectively.' To the right of the sidebar, there's a search bar and a blue button labeled 'Manage skills'. Below the sidebar, there are sections for 'Name' and 'App', each containing a placeholder image. A note below says 'No skills' and 'Click Manage skills to give this team digest specific abilities.' At the bottom, there are pagination controls.

### 3. Click Connections.

The screenshot shows the 'Connections' tab for the 'Zeeves draft' skill set. The top navigation bar and sidebar are identical to the previous screenshot. The main content area now displays the 'Connections' tab, which has a green border. Below the tab, there's a section titled 'Skills (2)' with a note: 'The skills added to this team are available to this team's members. These skills complete related tasks for the team members so members can work more efficiently and effectively.' Two skills are listed: 'Retrieve job output' and 'Zeeves-gather-facts-flow'. At the bottom, there are pagination controls.

### 4. Enter your skill app name (z/OS operations) in the search bar.

Skill sets

Zeeves draft

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type	Connected by	Action
z/os operations	1	Not specified		⋮

Items per page: 5 1-1 of 1 items

5. Click the ellipses icon (~⋮) for the z/OS operations app and then click **Connect app**.

Skill sets

Zeeves draft

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type	Connected by	Action
z/os operations	1	Not specified		⋮

Items per page: 5 1-1 of 1 items

6. In the dialog, click **Connect app**.

Skill sets

Zeeves draft

Skills Connections

These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

Application	Number of skills	Credential type
z/os operations	1	Not specified

Items per page: 5 1-1 of 1 items

Connect to z/OS operations

Each team member uses their own credentials to connect to this app and use its skills.

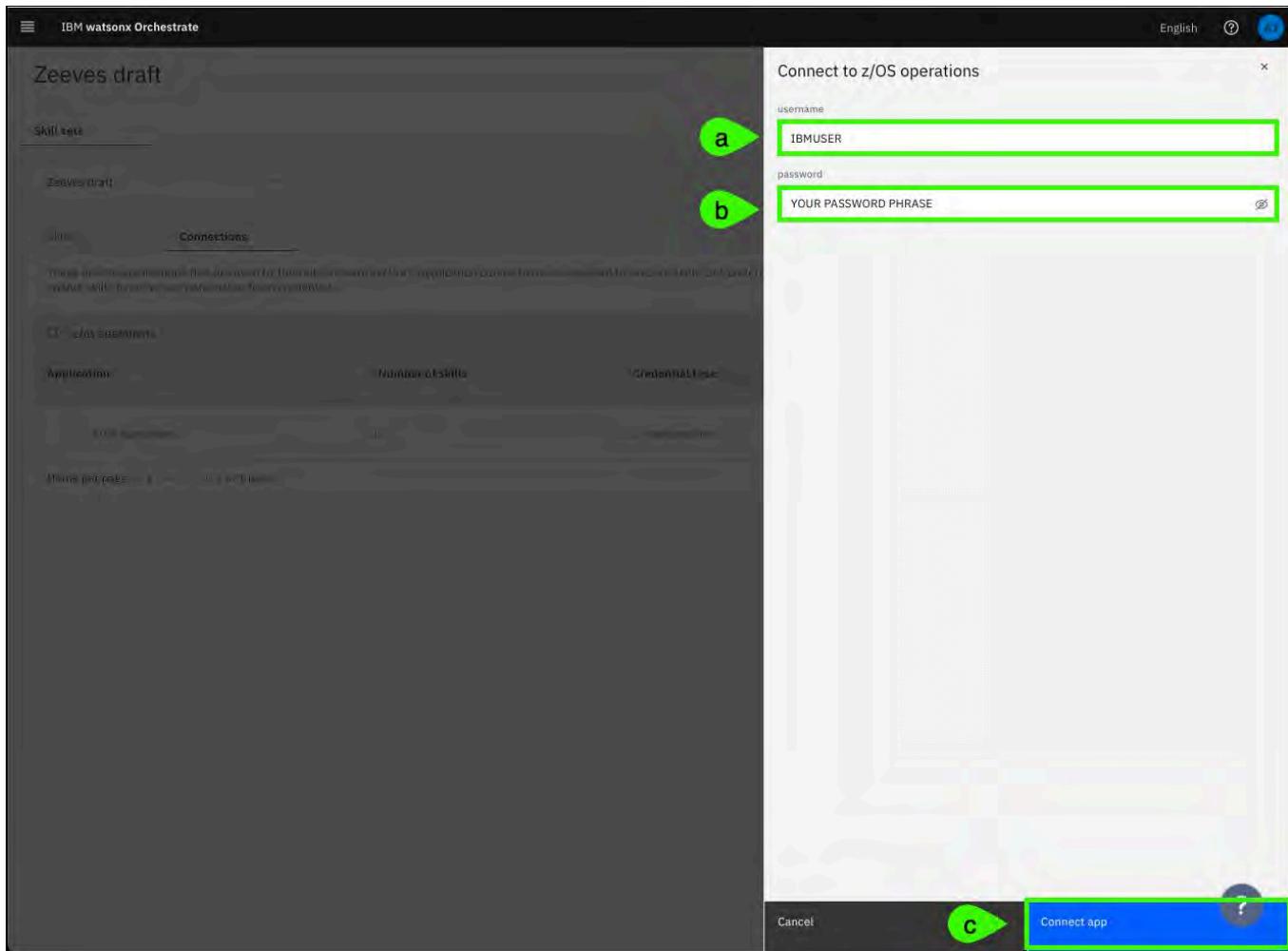
Team credentials

The admin sets the credentials each team member uses to connect to this app and use its skills.

You selected **Team credentials** for the credential type. Click **Connect app** to provide the credentials your team will use and to connect to the app.

Connect app ⌂

7. Enter (a) **IBMUSER** in the username field, (b) your **WaaS password** in the password field, and then (c) click **Connect app**.



## Create an action for your assistant

Next, create a skill-based action that uses the **z/OS IPL Information**. Recall, with the **z/OS Gather Facts** skill flow that was created earlier, adding the skill as a skill-based action allows the skill to run based on user prompts to the assistant.

Skill-based actions also serve another purpose. After creating the skill-based action, you can then call that action from a custom-built action. This is accomplished through subactions. For this use case, create an action that triggers the **z/OS IPL Information** skill on z/OS to display the LPARs IPL information. That action is integrated into another custom-built action (as a subaction) to provide a customized user experience.

1. Open **AI assistant builder** in watsonx Orchestrate.

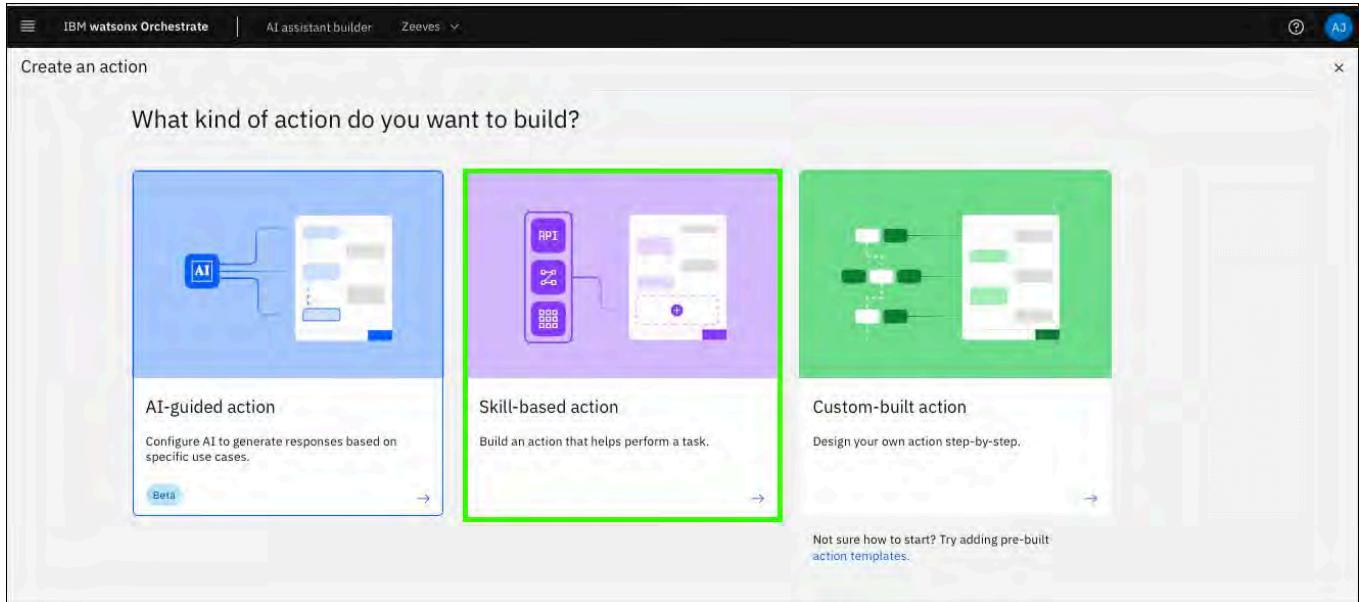
Number of skills	Credential type	Connected by	Action
1	Team	andrew@jones-tx.com	[Edit]

## 2. Select Actions.

## 3. Click New action +.

Name	Last edited	Examples count	Steps count	Status
Zeeves-gather-facts-flow	2 days ago	1	0	<span>Green icon</span>

## 4. Click Skill-based action.



## 5. Click **z/OS IPL Information** and then click **Next**.

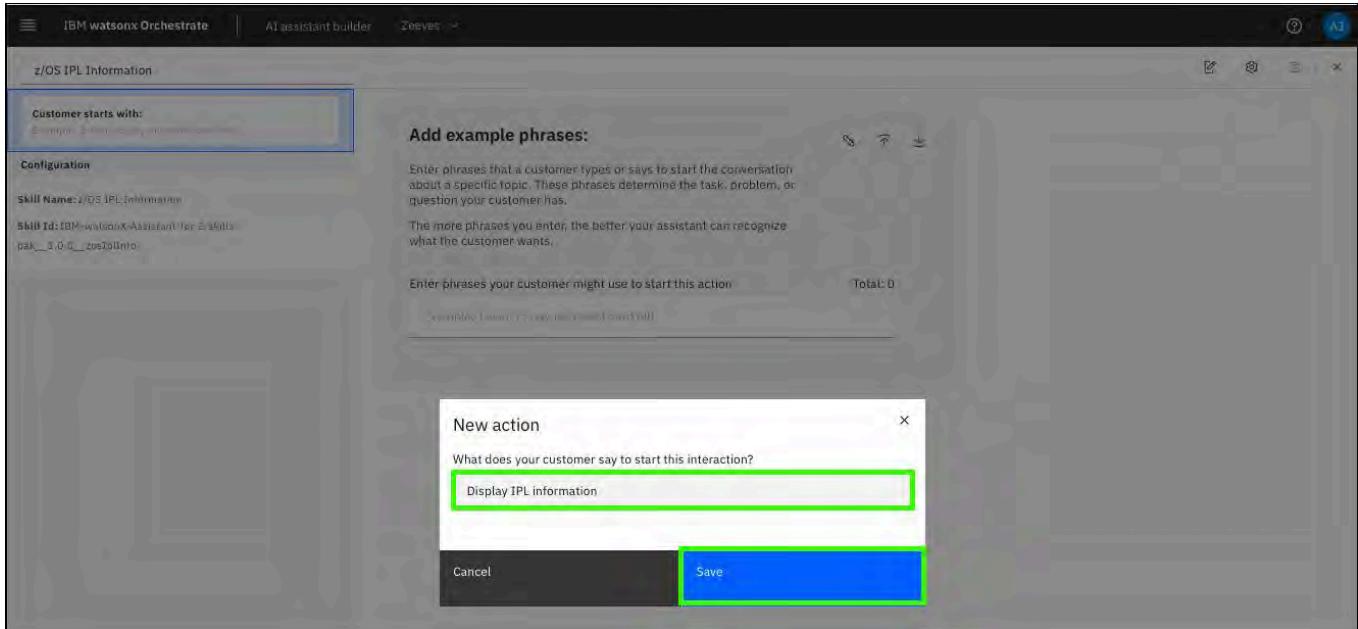
The screenshot shows the 'Build an action from a skill' screen. A specific skill, 'z/OS IPL Information', is selected and highlighted with a green box. The 'Next' button at the top right is also highlighted with a green box.

z/OS IPL Information	Zeeves-gather-facts-flow	Retrieve job output	z/OS Ping
z/OS IPL Information	Run z/OS gather facts and retrieve the output.	z skills - Retrieve job output by job Id	z skills - This playbook pings the z/OS host to test connectivity.
Last updated: 2025-01-17T14:39:36.042Z	Last updated: 2025-01-17T12:24:16.672Z	Last updated: 2025-01-17T12:18:14.714Z	Last updated: 2025-01-17T12:13:33.194Z
z/OS Gather Facts	Summarize the Webex meeting transcript	Summarize the Box content	Summarize a Zendesk ticket
z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS-specific information from the z/OS host.	in watsonx.ai	in watsonx.ai	in watsonx.ai

## 6. Enter a prompt like **Display IPL information** that starts the skill and then click **Save**.

Prompt:

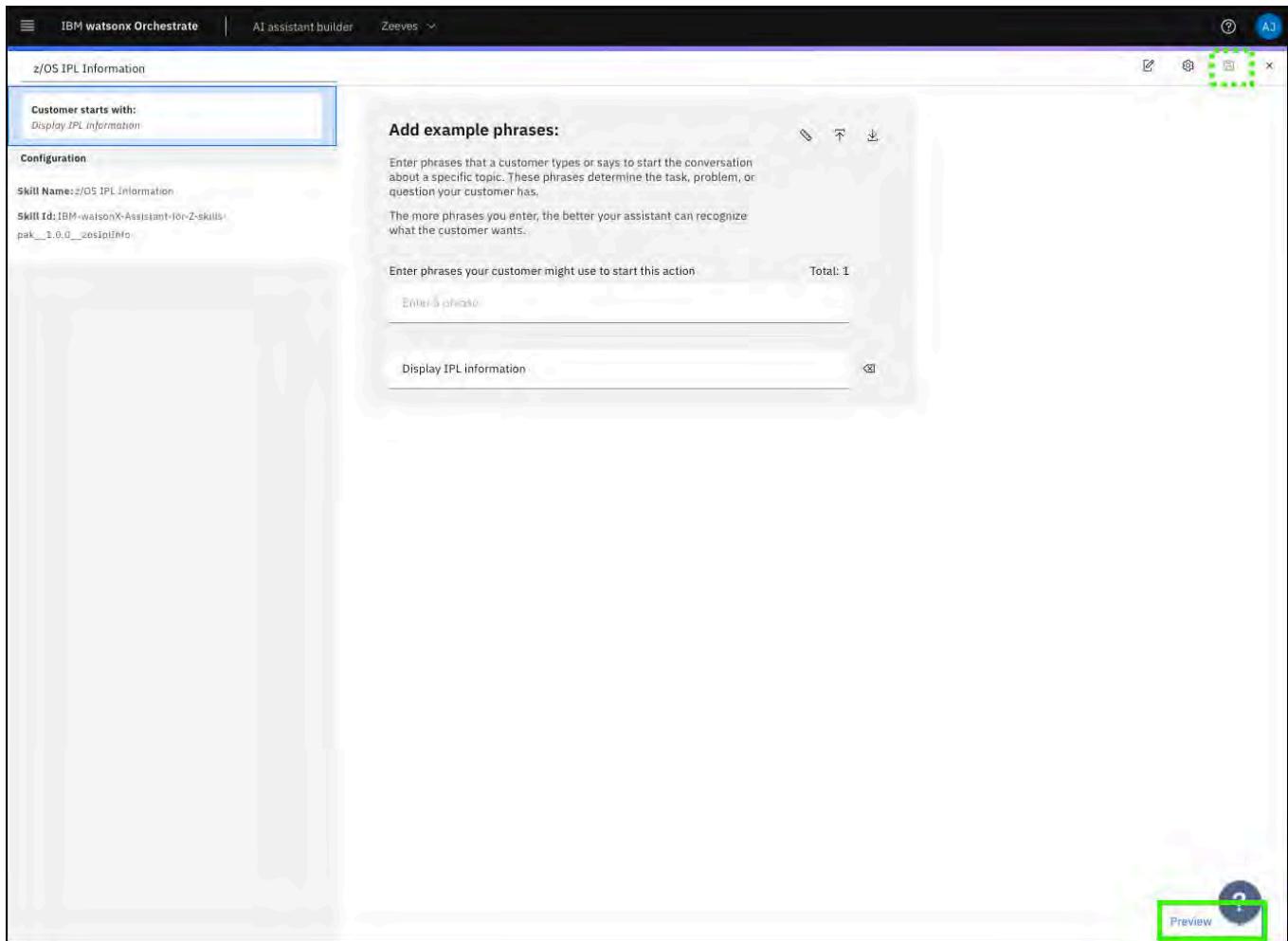
Display IPL information



7. Click the **Save** icon (💾).



8. Click **Preview**.



Wait for the Your changes are being added. message to disappear before proceeding.

9. Enter the prompt (**Display IPL information**) to test the skill.

z/OS IPL Information

**Customer starts with:**  
Display IPL information

**Configuration**

**Skill Name:** z/OS IPL Information

**Skill Id:** IBM-watsonX-Assistant-for-Z-skills-pak\_1.0.0\_zosiplinfo

**Add example phrases:**

Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.

The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 1

Display IPL information

**Preview**

8:46 AM  
Greet customer (default)

Welcome, how can I assist you?

Display IPL information

10. Using the defaults in the form, click **Apply**.

**Customer starts with:**  
Display IPL information

**Configuration**

**Skill Name:** z/OS IPL Information

**Skill Id:** IBM-watsonX-Assistant-for-Z-skills-pak\_1.0.0\_zosiplInfo

**Add example phrases:**

Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.

The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 1

Entering phrase

Display IPL information

**Preview**

9:02 AM  
Conversational skill called  
z/OS IPL Information recognized

z/OS IPL 1nformation  
consoleName \*  
iserVS01  
cmd \*  
diplinfo  
sol-key \*  
JES

Cancel Apply

## 11. Verify the results.

**Add example phrases:**

Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.

The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 1

Enter phrase:

Display IPL information

**Preview**

z/OS IPL Information

cmd-response  
IEE254I 11.04.51 IPLINFO DISPLAY  
SYSTEM IPLLED AT 06.48.48 ON 01/1  
RELEASE z/OS 02.05.00 LICENSE =  
USED LOADK2 IN SYS0.IPLPARM ON  
ARCHLVL = 2 MTLSHARE = N  
VALIDATED BOOT: NO  
IEASYM LIST = (00,K2)  
IEASYS LIST = (00) (OP)  
IODF DEVICE: ORIGINAL(ODE28) CUI  
IPL DEVICE: ORIGINAL(ODE27) CURF  
VM CPID = zHYPaa5  
VM UUID = 3C9BD7F3-8A47-  
4A47-B62C-298DA45F40EB  
VM NAME = k8s\_75d1  
VM EXT NAME = k8s\_75d16281c12c  
8a47

12. Click the **delete** icon (✖) to remove the example phrase (**Display IPL information**) from the skill.

Next, you will add this skill-based action as a subaction to a custom-build action. To have the custom-built action started rather than the skill-based action, the existing *example phrases* need to be removed.

**Add example phrases:**

Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.

The more phrases you enter, the better your assistant can recognize what the customer wants.

Enter phrases your customer might use to start this action Total: 1

Enter phrase:

Display IPL information

**Preview**

z/OS IPL Information

13. Verify (a) all **example phrases** are deleted, (b) click the **Save** icon (💾), and then (c) click **x** to close the action.

## Create a custom-built action

Custom-built actions allow you to define each step of a conversation with your assistant. You can define steps with or without conditions to control the user responses. You can include your skill-based actions as sub-actions. You can pass generated values between the subactions.

Recall the use case of an early-tenure SysProg who is preparing for an upcoming IPL on one of their LPARs. After going through documentation, the SysProg found that there is information that is needed about their LPAR before they can begin the IPL process. To gather that information, the SysProg might ask the assistant "How do you retrieve the information needed to IPL a z/OS LPAR? Provide a detailed response". A response the assistant might return is shown in the following image.

A custom-built action can help the SysProg to take the next steps that are required by adding next-step action suggestions and start skills to gather the needed information.

1. Click **New action +**.

The screenshot shows the 'Actions' section of the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with filters like 'All items', 'Created by you', 'Variables', and 'Saved responses'. The main area lists two actions: 'z/OS IPL Information' (created 17 minutes ago) and 'Zeeves-gather-facts-flow' (created 2 days ago). A green box highlights the 'New action' button in the top right corner of the list area.

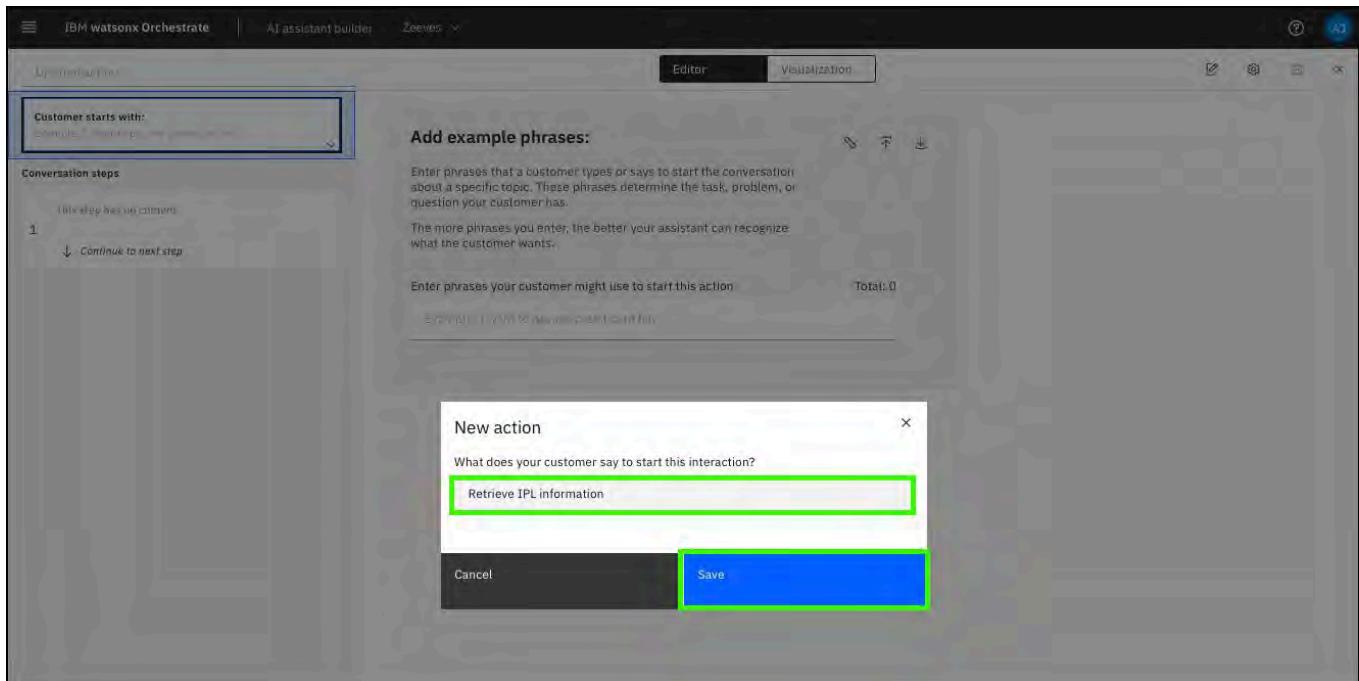
## 2. Click Custom-built action.

The screenshot shows the 'Create an action' dialog. It asks 'What kind of action do you want to build?' and lists three options: 'AI-guided action', 'Skill-based action', and 'Custom-built action'. The 'Custom-built action' option is highlighted with a green box. Below it, a note says 'Not sure how to start? Try adding pre-built action templates.'.

## 3. Enter an example phrase (**Retrieve IPL information**) to prompt the assistant to start the custom-built action and click **Save**.

Prompt:

Retrieve IPL information



#### 4. Review the form to create a custom-built action.

A custom-built action can consist of multiple steps with each step that is taken with or without conditions. Each step specifies the assistant's response and the next step to start.



#### For client demonstrations and pilots...

For client demonstrations and pilots, it is possible to add a custom search query such that when conversational search is executed in the 1st step, the query being used is hard-coded and isn't necessarily what the end-user inputted. This can be done by clicking on the Edit settings option under Search for the answer and specifying a custom search query that specifies the exact query to be used.

#### 5. Click the **And then** drop-down list and select **Search for the answer**.

Customer starts with:  
Retrieve IPL information

Conversation steps

Step 1: Is taken without conditions

Assistant says

And then

- Continue to next step
- Search for the answer**
- Continue to next step
- Go to a subaction
- Use an extension
- Connect to agent
- End the action

## 6. Click **Edit settings**.

Customer starts with:  
Retrieve IPL information

Conversation steps

Step 1: Is taken without conditions

Assistant says

And then

Custom query None (Customer's message will be the query)

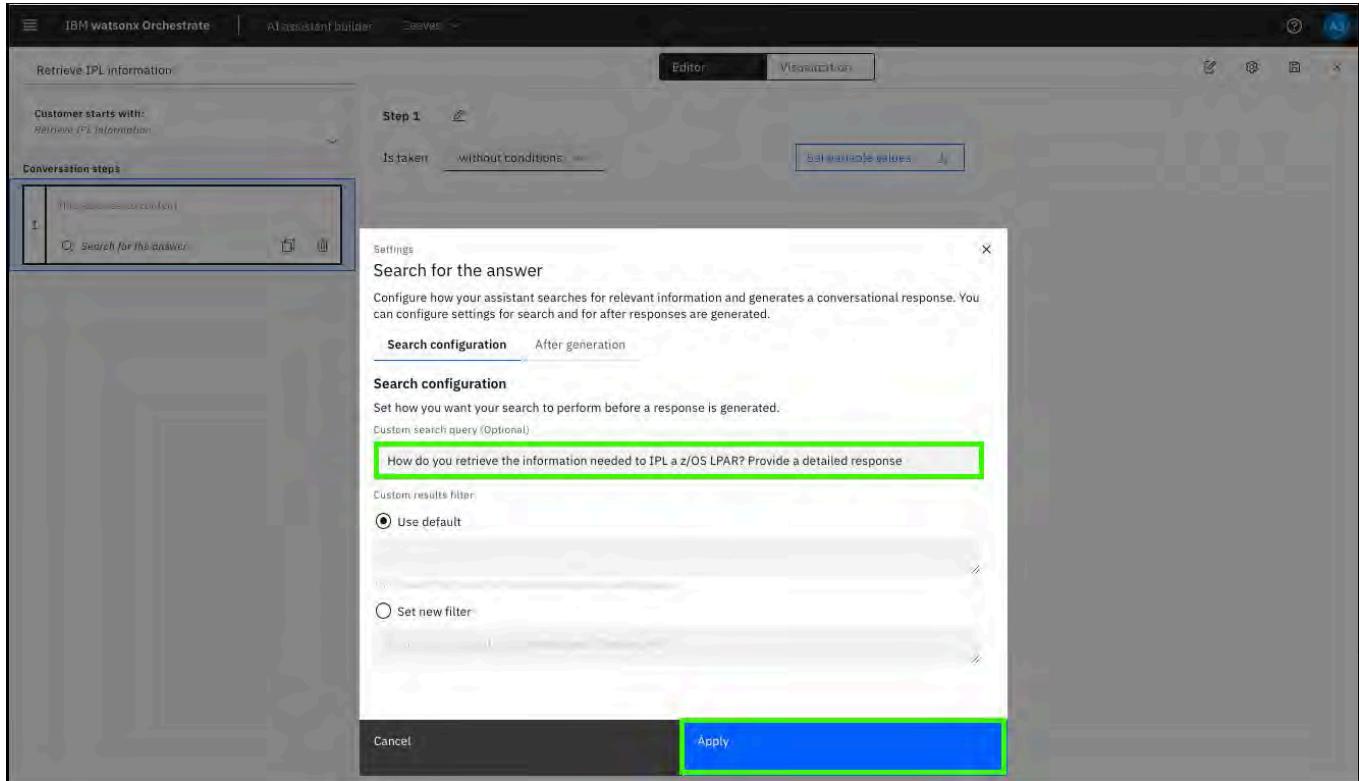
Custom filter None (Optional)

**Edit settings**

## 7. Enter a **Custom search query** and click **Apply**.

Custom search query:

How do you retrieve the information needed to IPL a z/OS LPAR? Provide a detailed response



#### 8. Click Next step +.

The new custom-built action only responds with a description of how to retrieve the IPL information. Now, infuse automation into the conversation and ask the user if they want to perform the action.

The screenshot shows the IBM Watsonx Assistant builder interface. At the top, it says "IBM Watsonx Orchestrate" and "AI assistant builder". Below that, "Retrieve IPL information" is selected under "Customer starts with: Retrieve IPL information". The main area is titled "Step 1" with the status "Is taken without conditions". A "Set variable values" button is visible. The "Conversation steps" section shows step 1, which has no content and a placeholder "Search for the answer". The "Assistant says" section contains a text input field with the placeholder "or example, Please select from the following options". Below this is a "Define customer response" dropdown. The "And then" section shows step 2 with a "Search for the answer" input field. It includes a "Custom query" field with the question "How do you retrieve the information needed to IPL a z/OS LPAR? Provide a detailed response.", a "Custom filter" field set to "None (Optional)", and an "Edit settings" button. At the bottom left is a "New step +" button, and at the bottom right is a "Preview" button.

9. In the **Assistant says** field, enter a response asking if the user wants to display the IPL information.

Assistant says:

Would you like to display your LPARs IPL information?

The screenshot shows the IBM Watsonx Assistant builder interface. At the top, it says "IBM Watsonx Orchestrate" and "AI assistant builder". Below that, "Retrieve IPL information" is selected under "Customer starts with: Retrieve IPL information". The main area is titled "Step 2" with the status "Is taken without conditions". A "Set variable values" button is visible. The "Conversation steps" section shows step 1 with a "Search for the answer" input field and step 2, which has a green dashed border around its entire content area. Step 2 contains a "Search for the answer" input field with the text "Would you like to display your LPARs IPL information?". The "Assistant says" section contains a text input field with the same text "Would you like to display your LPARs IPL information?". Below this is a "Define customer response" dropdown. The "And then" section shows step 3 with a "Search for the answer" input field. It includes a "Custom query" field with the question "Would you like to display your LPARs IPL information?", a "Custom filter" field set to "None (Optional)", and an "Edit settings" button. At the bottom left is a "Continue to next step" button, and at the bottom right is a "Preview" button.

10. Click the **Define customer response** drop-down and select **Confirmation**.

Now, after providing a conversational search-based response to the original question, the assistant asks the user if they want to display the IPL information and prompts the user to select **Yes** or **No**.

The screenshot shows the IBM Watsonx Orchestrate interface. At the top, it says "Retrieve IPL information". Below that, "Customer starts with:" is set to "Retrieve IPL information". Under "Conversation steps", step 1 has a placeholder "Search for the answer". Step 2 is currently selected, with the condition "Is taken without conditions". The "Assistant says" section contains the message "Would you like to display your LPARs IPL information?". A blue box highlights the second step and its message. A green box highlights the "Define customer response" dialog, which is open over the step details. The dialog shows the "System" tab selected, with "Confirmation" highlighted. It also shows options for "As buttons" (with "Checking", "Savings", "401 (k)", and "Roth IRA" listed) and "As a list" (with "Pay Bill" listed).

11. Click **Next step +**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and various icons. The main area is titled 'Retrieve IPL information' and shows 'Step 2' configuration. The 'Is taken' dropdown is set to 'without conditions'. The 'Assistant says' section contains a message: 'Would you like to display your LPARs IPL information?' followed by a 'Yes' or 'No' button. Below this, there's a 'View response' and 'Edit validation' link. The 'And then' section has a 'Continue to next step' button. On the left, 'Conversation steps' show steps 1 and 2. Step 1: 'Search for the answer' with a confirmation message. Step 2: 'Would you like to display your LPARs IPL information?' with a 'Confirmation' button. Step 3 is listed as 'This step has no content'. At the bottom, there's a 'New step +' button and a 'Preview' button.

## 12. Click the **Is taken** drop-down and select **with conditions**.

This screenshot shows the same interface after step 12. The 'Is taken' dropdown in 'Step 3' is now set to 'with conditions', which is highlighted with a green box. The rest of the interface remains largely the same, showing the conversation steps and the configuration for step 3.

## 13. Review the **Conditions**.

Notice that the default condition is based on **Step 2** and the user clicking **Yes** when prompted.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, the 'Conversation steps' pane displays three steps: 1. Search for the answer, 2. Confirmation, and 3. Retrieving IPL information. Step 3 has a note 'This step has no content'. In the center, 'Step 3' is selected, showing its configuration. It is set to 'Is taken with conditions'. A green dashed box highlights the 'Conditions' section, which contains a single condition: 'If 2. Would you like to d... is Yes'. Below this, there is a link 'and Add condition +' and a button 'New condition group +'.

#### 14. Enter **Retrieving your system's IPL information...** in the **Assistant says** field.

Assistant says:

Retrieving your system's IPL information...

This screenshot is identical to the one above, showing the same conversation steps and Step 3 configuration. However, the 'Assistant says' field in the 'Step 3' configuration pane is now highlighted with a thick green border. It contains the text 'Retrieving your system's IPL information...'.

#### 15. Click the **And then** drop-down and select **Go to a subaction**.

Customer starts with: Retrieve IPL information

Conversation steps

This step has no content.

1 Q. Search for the answer

Would you like to display your LPAR's IPL information?

2 Confirmation

↓ Continue to next step

3 2 is Yes  
Retrieving your system's IPL information...  
↓ Continue to next step

**Step 3**

Is taken with conditions

Set variable values

Conditions

If 2. Would you like to d... is Yes

and Add condition +

New condition group +

**Assistant says**

Retrieving your system's IPL information...

Define customer response

And then

- Continue to next step (highlighted)
- Re-ask previous step(s)
- Go to a subaction (highlighted)
- Use an extension
- Search for the answer
- Connect to agent
- End the action

Preview ?

16. Click (a) the **Go to** drop-down, (b) select your skill-based action (**z/OS IPL Information**), and then (c) click **Apply**.

Customer starts with: Retrieve IPL information

Conversation steps

This step has no content.

1 Q. Search for the answer

Would you like to display your LPAR's IPL information?

2 Confirmation

↓ Continue to next step

3 2 is Yes  
Retrieving your system's IPL information...  
↓ Continue to next step

**Step 3**

Is taken with conditions

Set variable values

Conditions

If 2. Would you like to d... is Yes

and Add condition +

**Go to a action**

Go to z/OS IPL Information (highlighted)

End this action after the b action is completed

Privacy (?)

Protect data returned after the subaction completes

Cancel Apply

17. Click **Save** (H) and then click **x**.

The screenshot shows the IBM Watsonx Orchestrate interface. At the top, it says "IBM Watsonx Orchestrate" and "AI assistant builder". Below that, "Zeeves" is selected. The main area is titled "Retrieve IPL information". It shows a conversation flow with three steps:

- Customer starts with:** Retrieve IPL information
- Conversation steps:**
  - 1. This step has no content.
  - 2. Would you like to display your LPARs IPL information? (Confirmation)
  - 3. Retrieving your system's IPL information... (Details: Go to subsection: z/OS IPL Inform...)
- Step 3:** Is taken with conditions. A condition is defined: If 2. Would you like to d... is Yes, then 3. Retrieving your system's IPL information... is taken.

## Test the custom-built action

You can now practice demonstrating the flow of this use case. Recall the steps that were taken and the scenario of the early-tenure SysProg being tasked with preparing for an upcoming IPL.

### 1. Click Preview.

The screenshot shows the "Actions" section of the AI assistant builder. The "Preview" option is highlighted with a green box. On the right, there is a list of actions created by the user:

Name	Last edited	Examples count	Steps count	Status
z/OS IPL Information	an hour ago	0	3	Green checkmark
Zeeves-gather-facts-flow	2 days ago	1	0	Green checkmark
Retrieve IPL information	15 minutes ago	1	3	Green checkmark

### 2. Enter How do you retrieve the information needed to IPL a z/OS LPAR? in the assistant prompt.

Prompt:

How do you retrieve the information needed to IPL a z/OS LPAR?



Use the Change layout option to open a full page view of the assistant.

The screenshot shows the 'Preview assistant' view in the IBM Watsonx Orchestrate interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main area is titled 'AssistantZeeves'. A toolbar at the top right contains 'Copy link to share', 'Change layout' (with a green dashed box around it), 'Change background', and 'Customize web chat'. On the left, there's a sidebar with various icons. The main content area features a blue owl icon and the text 'Hi! I'm Zeeves. How can I help you today?'. Below this is a search bar with the question 'How do you retrieve the information needed to IPL a z/OS LPAR?' followed by a green arrow button.

### 3. Review the response.

By using the conversational search capability of Watsonx Assistant for Z, grounded on Z-domain knowledge in the Z RAG, the user is able to quickly find an answer to their question. The assistant returns a response that shows the exact operator command they might use, and the types of details it provides which is relevant for preparing for an upcoming IPL. For example, the system's release level, the LOAD LIB information used for the IPL, the IODF device, the IPL device, and other system information.

You are also able to improve their productivity by infusing automations into the natural conversation that will provide them exactly the information they need to accomplish their task.

How do you retrieve the information needed to IPL a z/OS LPAR?

To retrieve the necessary information for IPLing a z/OS LPAR, you would typically refer to the system's documentation or consult with the system administrators. However, generally, you'd need the following:

1. LPAR Name: The unique identifier for the Logical Partition.
2. IPL Parameters: These include the system configuration parameters, such as the load module library, parmlib, and the address space layout.
3. System Symbols: These are used to customize the IPL process according to specific requirements.
4. IPL Source: This could be a specific member in a PDS (Partitioned Data Set), or a dataset specified in the IPL JCL (Job Control Language).

The exact process might vary depending on the specific mainframe environment and its configuration. Always ensure to follow the established procedures and guidelines provided by your organization's IT department.

Note: The document provided does not contain specific details about retrieving information for IPLing a z/OS LPAR. The provided information is based on general mainframe practices.

Would you like to display your LPARs IPL information?

#### 4. Click Yes.

How do you retrieve the information needed to IPL a z/OS LPAR?

To retrieve the necessary information for IPLing a z/OS LPAR, you would typically refer to the system's documentation or consult with the system administrators. However, generally, you'd need the following:

1. LPAR Name: The unique identifier for the Logical Partition.
2. IPL Parameters: These include the system configuration parameters, such as the load module library, parmlib, and the address space layout.
3. System Symbols: These are used to customize the IPL process according to specific requirements.
4. IPL Source: This could be a specific member in a PDS (Partitioned Data Set), or a dataset specified in the IPL JCL (Job Control Language).

The exact process might vary depending on the specific mainframe environment and its configuration. Always ensure to follow the established procedures and guidelines provided by your organization's IT department.

Note: The document provided does not contain specific details about retrieving information for IPLing a z/OS LPAR. The provided information is based on general mainframe practices.

Would you like to display your LPARs IPL information?

#### 5. Using the defaults in the form, click Apply.

The screenshot shows the IBM Watsonx Orchestrate interface with the "AI assistant builder" tab selected. On the left, a sidebar lists "Generative AI", "Actions", "Evaluate", "Preview" (which is selected), "Deploy", "Publish", "Environments", "Improve", "Analyze", "Integrations", and "Activity log". The main area displays an AI-generated document titled "AssistantZeeves" with the following content:

```

AssistantZeeves
specified in the IPL JCL (Job Control Language).

The exact process might vary depending on the specific mainframe environment and its configuration. Always ensure to follow the established procedures and guidelines provided by your organization's IT department.

Note: The document provided does not contain specific details about retrieving information for IPLing a z/OS LPAR. The provided information is based on general mainframe practices.

[Note: Accuracy of generated answers may vary.]

Would you like to display your LPARs IPL information?
  Yes
  No
  You 10:37 AM
  Yes
  10:37 AM
  Retrieving your system's IPL information...
  z/OS IPL Information
  consoleName *
  iserV501
  cmd *
  d iplinfo
  sol-key *
  JES
  Cancel
  Apply

```

A green dashed box highlights the "Apply" button at the bottom of the "z/OS IPL Information" dialog.

## 6. Review the response to the custom-built action.

The output provides relevant pieces of information the early-tenure SysProg needs to prepare for an IPL on their z/OS LPAR. For example, when the system was last IPLed, the z/OS release they have running (2.5), the IPL volume (D25VS1), the IPL LOAD PARM (LOADK2) used during the last IPL, and other relevant details.

The screenshot shows the IBM Watsonx Assistant interface. At the top, there are tabs for "IBM Watsonx Orchestrate", "AI assistant builder", and "Zeeves". On the right, there are buttons for "Copy link to share", "Change layout", "Change background", and "Customize web chat". The main area is titled "Preview assistant" and contains a card titled "AssistantZeeves". The card displays "Z/OS IPL Information" with the following content:

```
IEE2541 12.40.03 IPLINFO DISPLAY 770
SYSTEM IPLED AT 06.48.48 ON 01/17/2025
RELEASE z/OS 02.05.00 LICENSE = z/OS
USED LOADK2 IN SYS0.IPLPARM ON ODE28
ARCHLVL = 2 MTLSHARE = N
VALIDATED BOOT: NO
IEASYM LIST = (00,K2)
IEASYS LIST = (00) (OP)
IODEVICE: ORIGINAL(ODE28) CURRENT(ODE28)
IPL DEVICE: ORIGINAL(ODE27) CURRENT(ODE27) VOLUME(D25VS1)
VM CPID = zHYPaaS
VM UUID = 3C9BD7F3-8A47-4A47-B62C-298DA45F40EB
VM NAME = k8s_75d16281c12c4681b91417749e322845_0727_3c9bd7f3-8a47-b62c-298da45f40eb
cmd-response-key
C4443879
```

At the bottom of the card, it says "10:39 AM". Below the card is a text input field with the placeholder "Type something..." and a send button.

The ability to infuse skills and automations into the conversation allows the user to issue the operator command on their system directly within the assistant itself. The assistant provides a single interface for retrieving details that are required for the IPL process to go smoothly and increases the user's efficiency.

# Use case: SSL Certificate renewal on z/OS

Now, shift roles to that of a mainframe Security Administrator (SA). The client want to understand how watsonx Assistant for Z can help them to verify that their critical security certificates are up to date and reduce the risk of expired certificates disrupting their organization's services.

Secure Sockets Layer (SSL) certificates, often referred to as digital certificates, are used to establish an encrypted connection between communicating parties over a network. Certificate management is crucial for maintaining the security of a company's z/OS environment. The SA has not performed the tasks to manage and renew a certificate in some time. The SA recalls that there are many steps that are required on z/OS and various RACF commands that need to be run to renew a certificate. Rather than going to their senior SA for assistance, demonstrate how using watsonx Assistant for Z can help the SA automate the certificate renewal process.

In this scenario, use the Ansible automation templates that are provided with AAP and the WAZI z/OS environment to create assistant actions. The actions guide the client through the process of identifying their SSL certificate's expiration dates, and automating the certificate renewal process for them. The assistant saves them time and improve their productivity.

## Create an initial certificate authority (CA) certificate to sign future SITE certificates

For this use case, a certificate authority (CA) certificate is needed to sign new SITE certificates.

1. Open and log in to the Ansible Automation Platform (AAP) web console.



**Don't remember how?**

Refer to the first 5 steps in [Explore Ansible Automation Platform](#).

2. Click **Templates** under the **Resources** section.

The screenshot shows the 'Jobs' list page. The left sidebar has a 'Resources' section with 'Templates' selected, which is highlighted with a green box. The main area displays a table of jobs with columns: Name, Status, Type, Start Time, Finish Time, and Actions. Three jobs are listed:

Name	Status	Type	Start Time	Finish Time	Actions
15 – Cleanup Job Details	Successful	Management Job	1/19/2025, 5:02:46 AM	1/19/2025, 5:02:48 AM	
14 – z/OS Certs - Add CA Cert	Successful	Playbook Run	1/17/2025, 10:08:07 AM	1/17/2025, 10:09:29 AM	
13 – z/OS TSO Command	Successful	Playbook Run	1/17/2025, 7:41:26 AM	1/17/2025, 7:41:35 AM	

3. Click the **launch** icon ( ) for the **z/OS Certs - Create Cert** template.

The screenshot shows the 'Templates' list page. The left sidebar has a 'Resources' section with 'Templates' selected, which is highlighted with a green box. The main area displays a table of templates with columns: Name, Type, Organization, Last Ran, and Actions. Three templates are listed:

Name	Type	Organization	Last Ran	Actions
z/OS Certs - Add CA Cert	Job Template	Default	1/17/2025, 10:09:29 AM	
<b>z/OS Certs - Create Cert</b>	Job Template	Default		
z/OS Certs - Create Keyring	Job Template	Default		

4. On the **Survey** screen, modify the **Certificate Label** and **Type** fields with the values that follow and then click **Next**.

**a: Certificate Label**

TESTCA

**b: Type**

CERTAUTH



Leave the default values for all other fields.

The screenshot shows the 'Launch | z/OS Certs - Create Cert' survey step. The left sidebar lists various navigation options like Dashboard, Jobs, Activity, etc. The main window title is 'Launch | z/OS Certs - Create Cert'. It says 'This playbook creates a SITE, USER or CERTAUTH certificate. Signed by local CA or self-signed.' The survey step is labeled 'Survey' (step 1) and 'Preview' (step 2). The form fields are:

- Owner ID**: IBMUSER
- Certificate Label**: TESTCA
- Type**: CERTAUTH
- Sign With**: Self
- Sign Label**: (empty)
- Organizational Unit**: ANSIBLE CORE

At the bottom are buttons for **Next** (highlighted in green), **Back**, and **Cancel**.

5. Click **Launch**.

**Launch | z/OS Certs - Create Cert**

This playbook creates a SITE, USER or CERTAUTH certificate. Signed by local CA or self-signed.

Setting	Value
Name	z/OS Certs - Create Cert
Description	This playbook creates a SITE, USER or CERTAUTH certificate. Signed by local CA or self-signed.
Timeout	0 min 0 sec
Job Type	Run
Organization	Default
Inventory	AAP z/OS
Project	aap4zos
Execution Environment	aap4zos EE
Playbook	zos_concepts/certificate_management/create_cert.yml
Forks	0
Verbosity	0 (Normal)
Show Changes	Off
Job Slicing	1
Credential	SSH: z/OS Host SSH ... s

**Actions**

Launch Back Cancel

## 6. Review the output of the job.

In the output of the playbook, notice that a new keyring is created, a certificate is created, and the certificate is connected to the key ring.

The screenshot shows the Red Hat Ansible Automation Platform interface. The left sidebar has sections for Views (Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals, Host Metrics, Subscription Usage), Resources (Templates, Credentials, Projects, Inventories, Hosts), Access (Organizations, Users, Teams), and Administration (Credential Types). The main area shows a job titled "z/OS Certs - Create Cert" with a status of "Successful". The "Output" tab is selected, showing the command-line output of the job. The output text is as follows:

```
Events processing complete. Reload output
10
11 TASK [Get FQDN of this z/OS server from USS.] ****
12 ok: [zos_host]
13
14 TASK [Set cn fact from previous task.] ****
15 ok: [zos_host]
16
17 TASK [Check if certificate to be created already exists.] ****
18 ok: [zos_host]
19
20 TASK [Debug command output.] ****
21 ok: [zos_host] => {
22     "msg": "4"
23 }
24
25 TASK [Debug.] ****
26 ok: [zos_host] => {
27     "cert_not_found": true
28 }
29
```

7. Locate the line **TASK [GENERATE new certificate]**, click the **changed: [zos host]**.

The screenshot shows the Red Hat Ansible Automation Platform interface. The left sidebar is dark-themed and includes sections for Views, Dashboard, Jobs (which is selected), Schedules, Activity Stream, Workflow Approvals, Host Metrics, Subscription Usage, Resources, Templates, Credentials, Projects, Inventories, Hosts, Access, Organizations, Users, Teams, Administration, Credential Types, and Notifications. The main area is titled 'Jobs > 16 - z/OS Certs - Create Cert' and has tabs for 'Output' (selected), 'Details', and 'Back to Jobs'. Below the tabs, it says 'z/OS Certs - Create Cert' and 'Successful'. It shows metrics: Plays 1, Tasks 15, Hosts 1, Elapsed 00:00:20. Under 'Stdout', there is a log window with a message 'Events processing complete. Reload output'. The log itself is a series of numbered lines representing task logs. Line 37, which contains the text 'changed: [zos\_host]', is highlighted with a green rectangular box.

```

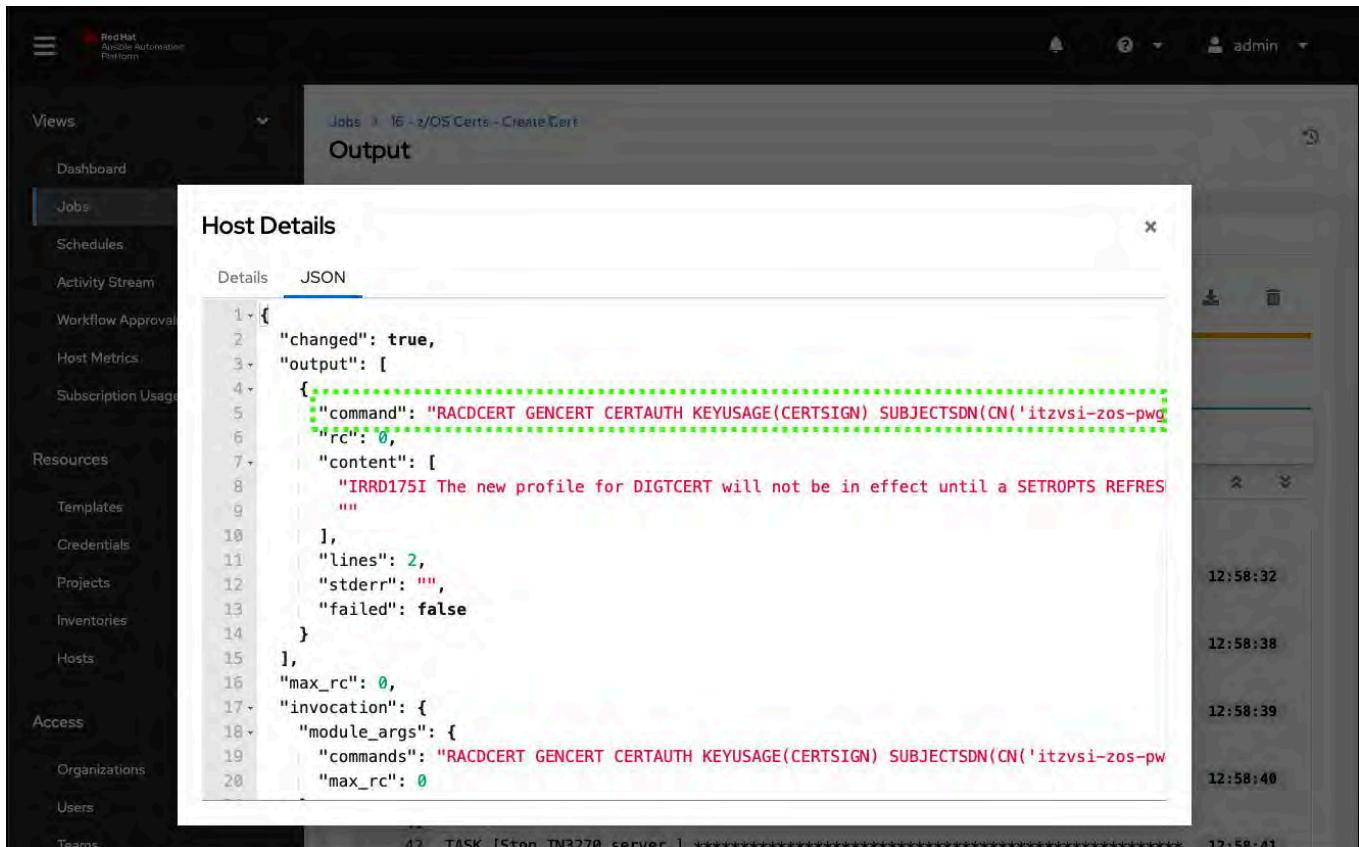
27      "cert_not_found": true
28  }
29
30 TASK [Generate new certificate.] *****
31 changed: [zos_host]
32
33 TASK [List chain for new certificate.] *****
34 changed: [zos_host]
35
36 TASK [Connect new certificate to keyring.] *****
37 changed: [zos_host]
38
39 TASK [List keyring.] *****
40 changed: [zos_host]
41
42 TASK [Stop TN3270 server.] *****
43 skipping: [zos_host]
44
45 TASK [Start TN3270 server] *****
46 skipping: [zos_host]

```

## 8. Click JSON.

This screenshot shows the same interface as the previous one, but with a modal window open over the 'Output' tab. The modal is titled 'Host Details' and has two tabs: 'Details' (which is the active tab) and 'JSON'. The 'Details' tab shows information for the host 'zos\_host', including its status as 'Changed'. The 'JSON' tab displays a JSON object representing the host details. A tooltip is visible on the right side of the modal, providing instructions: 'Check if SITE cert and keyring already exist. If not, create them and then run health check via TSO and operator commands.' The background shows the log output from the previous screenshot.

## 9. Review the RACDCERT command that was run to generate the certificate and then click x to close the window.



```

1: {
2:   "changed": true,
3:   "output": [
4:     {
5:       "command": "RACDCERT GENCERT CERTAUTH KEYUSAGE(CERTSIGN) SUBJECTSDN(CN('itzvsi-zos-pw0'))",
6:       "rc": 0,
7:       "content": [
8:           "IRRD175I The new profile for DIGTCERT will not be in effect until a SETROPTS REFRESH is issued."
9:       ],
10:      "lines": 2,
11:      "stderr": "",
12:      "failed": false
13:    }
14:  },
15:  "max_rc": 0,
16:  "invocation": {
17:    "module_args": {
18:      "commands": "RACDCERT GENCERT CERTAUTH KEYUSAGE(CERTSIGN) SUBJECTSDN(CN('itzvsi-zos-pw0'))",
19:      "max_rc": 0
20:    }
21:  }
22: }

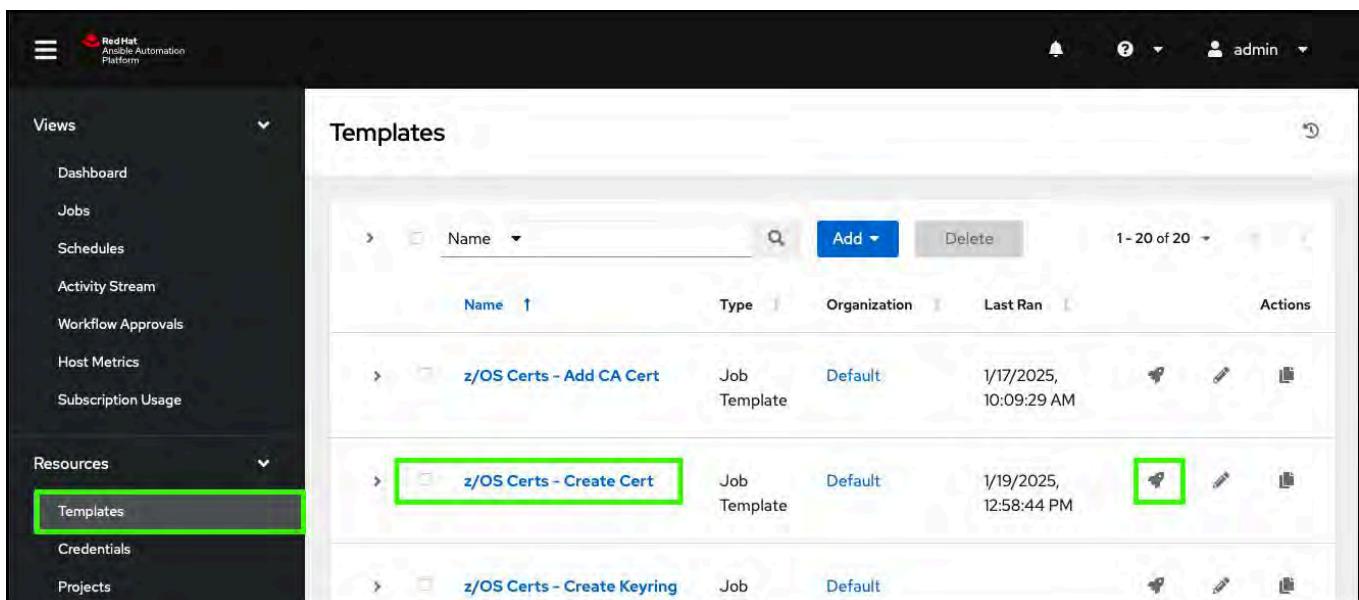
```

The screenshot shows the 'Output' tab for a specific job run. The 'Host Details' panel is open, displaying the JSON output of the job. The output shows a command execution log with a redacted section. The log includes a warning message from IRRD175I about the new profile not being effective until a SETROPTS REFRESH is issued. The job was run at 12:58:32 and completed successfully at 12:58:40.

## Create an expiring certificate

Now, create an expiring certificate that uses the CA certificate that you just created.

1. Return to the **Templates** tab and click the **launch** icon (🔗) for the **z/OS Certs - Create Cert** template.



Name	Type	Organization	Last Ran	Actions
<a href="#">z/OS Certs - Add CA Cert</a>	Job Template	Default	1/17/2025, 10:09:29 AM	
<a href="#">z/OS Certs - Create Cert</a>	Job Template	Default	1/19/2025, 12:58:44 PM	
<a href="#">z/OS Certs - Create Keyring</a>	Job	Default		

The screenshot shows the 'Templates' list. The 'z/OS Certs - Create Cert' template is highlighted with a green box. The 'Actions' column for this template includes a launch icon (highlighted with a green box).

2. On the **Survey** screen, modify the fields that follow with the values specified and then click **Next**.

a: Type

SITE

**b: Sign with**

CERTAUTH

**c: Sign Label**

TESTCA

**d: Common Name**

company . com

**e: Expiration Date**

Enter a date that occurs within the next 30 days. The date must be in the format YYYY-MM-DD.

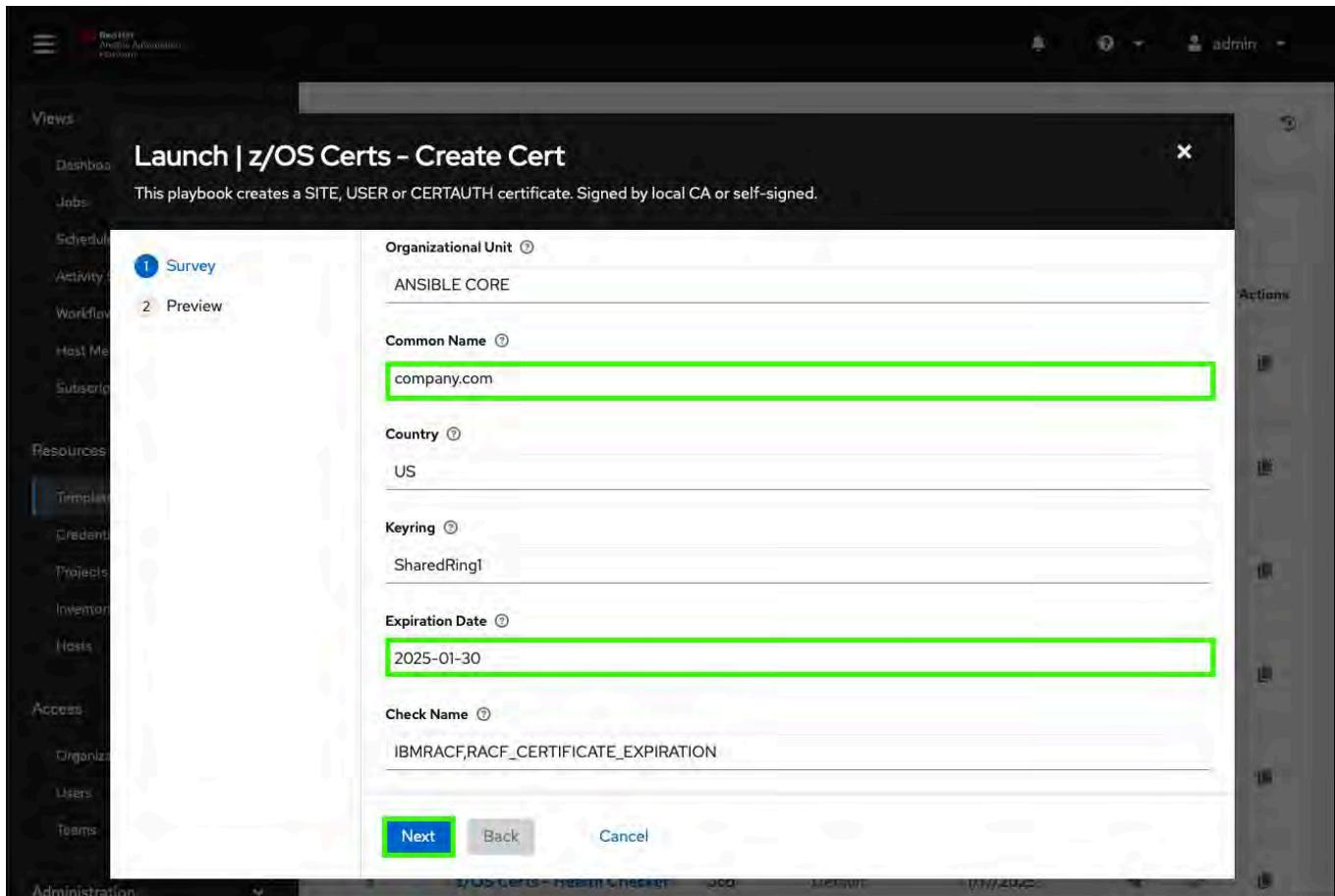


Leave the default values for all other fields.

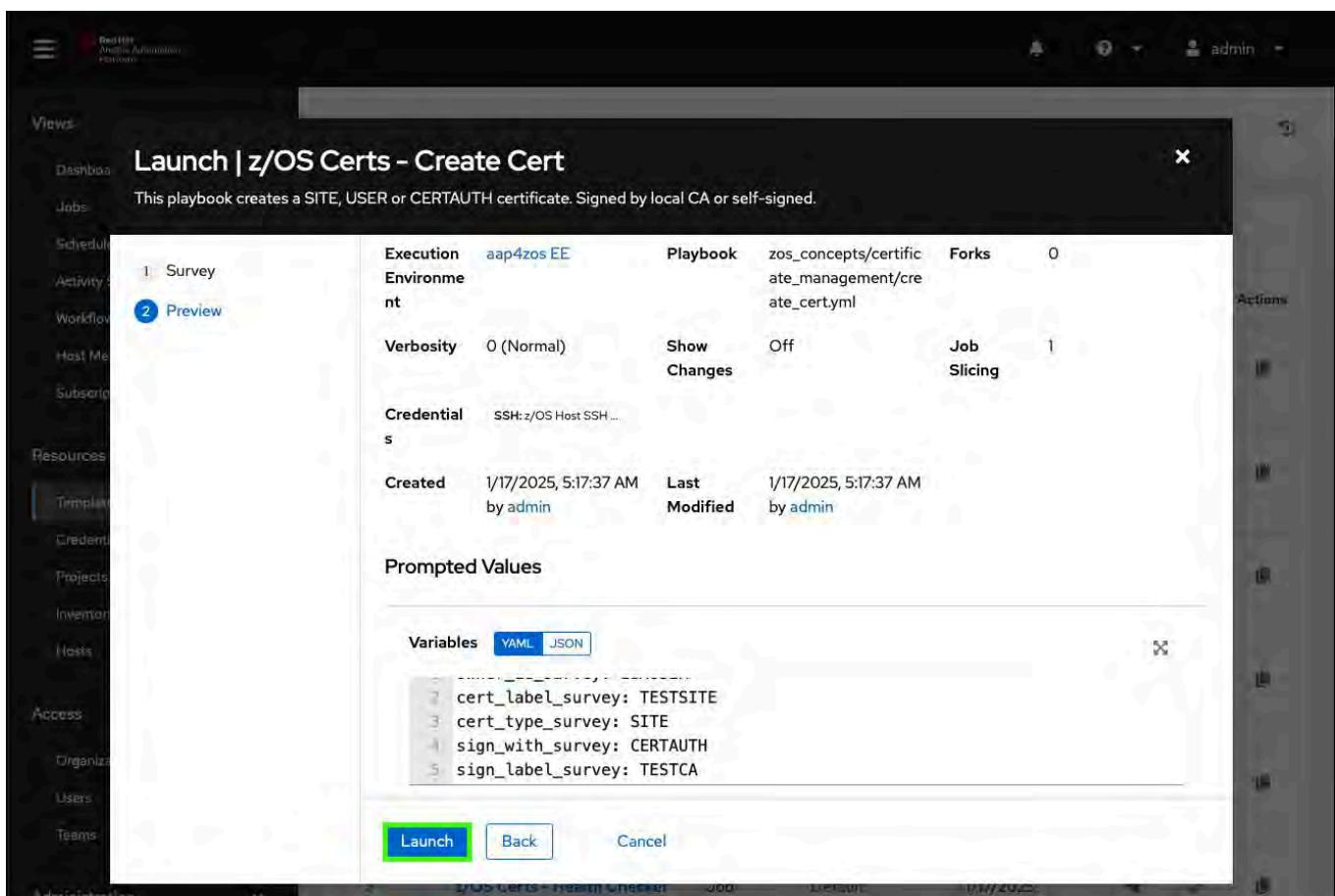
Unlike the first certificate you created which was *self-signed*, this certificate will be signed by the local certificate authority that uses the CA you created.



The following image does not highlight all the fields that need to be modified!



### 3. Click Launch.



4. Verify that the job was successful and inspect the output of the job.

The screenshot shows the Red Hat Ansible Automation Platform interface. The left sidebar has sections for Views (Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals, Host Metrics, Subscription Usage), Resources (Templates, Credentials, Projects, Inventories, Hosts), Access (Organizations, Users, Teams), and Administration (Credential Types, Notifications). The main area shows a job titled "z/OS Certs - Create Cert" with a status of "Successful". Below it, the "Output" tab is selected, showing command-line logs. The logs include:

```

15: *SKIPPING: [zos_host]*****
16:
17: TASK [Check if certificate to be created already exists.] **** 13:16:28
18: ok: [zos_host]
19:
20: TASK [Debug command output.] **** 13:16:29
21: ok: [zos_host] => {
22:   "msg": "4"
23: }
24:
25: TASK [Debug.] **** 13:16:29
26: ok: [zos_host] => {
27:   "cert_not_found": true
28: }
29:
30: TASK [Generate new certificate.] **** 13:16:29
31: changed: [zos_host]
32:
33: TASK [List chain for new certificate.] **** 13:16:34
34: changed: [zos_host]
35: ****

```

## Renew the expiring certificate

Now that you have a certificate and it is expiring within 30 days, it is time to renew the certificate.

1. Return to the **Templates** tab and click the **launch** icon (💡) for the **z/OS Certs - Search and Renew** template.

Template Name	Type	Status	Last Run	Action Icons
<a href="#">z/OS Certs - Add CA Cert</a>	Job Template	Default	1/17/2025, 10:09:29 AM	
<a href="#">z/OS Certs - Create Cert</a>	Job Template	Default	1/19/2025, 1:16:45 PM	
<a href="#">z/OS Certs - Create Keyring</a>	Job Template	Default		
<a href="#">z/OS Certs - Delete Cert</a>	Job Template	Default		
<a href="#">z/OS Certs - Delete Keyring</a>	Job Template	Default		
<a href="#">z/OS Certs - Health Checker</a>	Job Template	Default	1/17/2025, 5:17:58 AM	
<a href="#">z/OS Certs - List Cert</a>	Job Template	Default		
<a href="#">z/OS Certs - Search and Renew</a>	Job Template	Default		
<a href="#">z/OS Data Set - Basics</a>	Job Template	Default		

2. On the **Survey** screen, modify the fields that follow with the values specified and then click **Next**.

**a: Certificate Label**

TESTSITE

**b: Type**

SITE

**c: Sign with**

CERTAUTH

**d: Sign Label**

TESTCA

**e: Expiration Date**

Specify a new expiration date in the format YYYY-MM-DD.



The following image does not highlight all the fields that need to be modified!

This screenshot shows a configuration dialog titled "Launch | z/OS Certs - Search and Renew". The dialog is part of a larger interface with a sidebar containing various navigation items like Dashboard, Jobs, Schedules, Activity, Workflow, Hosts, Subscriptions, Resources, Templates, Credentials, Projects, Inventories, and Access. The main title bar indicates the current page is "z/OS Certs - Add CA Lets" and shows the date and time as "1/17/2025, 10:09:29 AM". The user is currently on the "Survey" tab, which is highlighted with a blue circle.

The survey configuration fields are as follows:

- Certificate Label \***: TESTSITE
- Type \***: SITE
- Sign With \***: CERTAUTH
- Sign Label**: TESTCA
- Owner ID \***: IBMUSER
- Keyring \***: SharedRing1

At the bottom of the dialog, there are three buttons: "Next" (highlighted with a green box), "Back", and "Cancel".

3. Click **Launch**.

**Launch | z/OS Certs - Search and Renew**

This playbook searches for certificates with the details specified in the AAP survey and renews them.

Survey	Name	Description	Type	Job Template
2 Preview	z/OS Certs - Search and Renew	This playbook searches for certificates with the details specified in the AAP survey and renews them.		
	Timeout	0 min 0 sec	Job Type	Run
	Inventory	AAP z/OS	Project	aap4zos
	Playbook	zos_concepts/certificate_management/search_and_renew.yml	Forks	0
	Show Changes	Off	Job Slicing	1
	Credential	SSH: z/OS Host SSH ...		

**Launch**   **Back**   **Cancel**

4. Verify that the job was **Successful** and review the output.

**Note:** Click the **Reload Output** button to view the full output after the job completes.

Review the tasks that were run within the automation to renew the certificate. Some of the steps completed include:

- Run the RACF\_CERTIFICATE\_EXPIRATION z/OS Health Check
- Submit JCL to pull a report on the z/OS Health Check
- Search the output of the report for the given certificate label
- Print the expiring certificate, if it is found. You see: 'TESTSITE expiring – True'
- If the certificate is expiring, start a series of RACDCERT commands to do the following:
  - Backup the expiring certificate
  - Rekey the certificate and give it a new temporary label
  - Generate a CSR for the new certificate
  - Sign the new certificate with the local CA
  - Delete the old certificate
  - Relabel the new certificate to use the same label as before
  - Refresh the digital certificate list

The screenshot shows the Red Hat Ansible Automation Platform web interface. On the left, there's a sidebar with sections like Views, Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals, Host Metrics, Subscription Usage, Resources, Templates, Credentials, Projects, Inventories, Hosts, Access, Organizations, Users, Teams, Administration, and Credential Types. The 'Jobs' section is currently selected. In the main area, a job titled 'z/OS Certs - Search and Renew' is displayed. The 'Output' tab is active. The job status is shown as 'Successful' with a green checkmark icon. Below the status, there's a yellow progress bar. The output pane contains a log entry: 'Events processing complete. Reload output'. The log itself is a large block of text detailing a series of RACF commands and their execution details.

## Create another expiring certificate

Create one more expiring certificate to use with the assistant and the new skills you will create.

1. Return to the **Templates** tab and click the **launch** icon (💡) for the **z/OS Certs - Create Cert** template.

The screenshot shows the Red Hat Ansible Automation Platform web interface. The 'Templates' tab is selected in the sidebar. A table lists existing templates. One template, 'z/OS Certs - Create Cert', is highlighted with a green box around its row. Its corresponding launch icon (a lightbulb icon) is also highlighted with a green box. Other columns in the table include Name, Type, Organization, Last Ran, and Actions.

Name	Type	Organization	Last Ran	Actions
z/OS Certs - Add CA Cert	Job Template	Default	1/17/2025, 10:09:29 AM	
<b>z/OS Certs - Create Cert</b>	Job Template	Default	1/19/2025, 1:16:45 PM	

2. On the **Survey** screen, modify the fields that follow with the values specified and then click **Next**.

**a: Certificate Label**

DEMOCERT

**b: Type**

SITE

**c: Sign with**

CERTAUTH

**d: Sign Label**

TESTCA

**e: Common Name**

company.com

**f: Expiration Date**

Enter a date that falls within the next 30 days in the format YYYY-MM-DD.



The following image does not highlight all the fields that need to be modified!

Views

Dashboard

Jobs

Schedule

Activity

Workflow

Host Mgmt

Subscriptions

Resources

Templates

Credentials

Projects

Inventories

Hosts

Access

Organizations

Users

Teams

Administration

Launch | z/OS Certs - Create Cert

This playbook creates a SITE, USER or CERTAUTH certificate. Signed by local CA or self-signed.

1 Survey

2 Preview

Organizational Unit

ANSIBLE CORE

Common Name

company.com

Country

US

Keyring

SharedRing1

Expiration Date

2030-01-30

Check Name

IBMRACF,RACF\_CERTIFICATE\_EXPIRATION

Next Back Cancel

3. Click **Launch**.

Views

Launch | z/OS Certs - Create Cert

This playbook creates a SITE, USER or CERTAUTH certificate. Signed by local CA or self-signed.

1 Survey      2 Preview

Execution Environment: aap4zos EE  
Playbook: zos\_concepts/certificate\_management/create\_cert.yml  
Forks: 0  
Verbosity: 0 (Normal)  
Show Changes: Off  
Job Slicing: 1  
Credential: SSH:z/OS Host SSH ...  
Created: 1/17/2025, 5:17:37 AM by admin  
Last Modified: 1/17/2025, 5:17:37 AM by admin

Prompted Values

Variables: [YAML](#) [JSON](#)

```
[{"owner_id_survey": "IBMUSER"}, {"cert_label_survey": "DEMOCERT"}, {"cert_type_survey": "SITE"}, {"sign_with_survey": "CERTAUTH"}]
```

Launch Back Cancel

4. Verify that the **DEMOCERT** was successfully created.

The screenshot shows the Red Hat Ansible Automation Platform interface. On the left, a sidebar navigation includes 'Views', 'Dashboard', 'Jobs' (selected), 'Schedules', 'Activity Stream', 'Workflow Approvals', 'Host Metrics', and 'Subscription Usage'. Under 'Resources', it lists 'Templates', 'Credentials', 'Projects', 'Inventories', and 'Hosts'. Under 'Access', it lists 'Organizations', 'Users', and 'Teams'. Under 'Administration', it lists 'Credential Types'. The main content area shows a job named '19 - z/OS Certs - Create Cert'. The 'Output' tab is selected, showing a summary bar with 'Successful' status, 1 play, 15 tasks, 1 host, and an elapsed time of 00:00:25. Below this is a search bar set to 'Stdout'. A message 'Events processing complete' with a 'Reload output' button is displayed. The main log area shows the following Ansible command output:

```

15 Skipping: [zos_host]
16
17 TASK [Check if certificate to be created already exists.] *****
18 ok: [zos_host]
19
20 TASK [Debug command output.] *****
21 ok: [zos_host] => {
22     "msg": "4"
23 }
24
25 TASK [Debug.] *****
26 ok: [zos_host] => {
27     "cert_not_found": true
28 }
29
30 TASK [Generate new certificate.] *****
31 changed: [zos_host]
32
33 TASK [List chain for new certificate.] *****
34 changed: [zos_host]
35

```

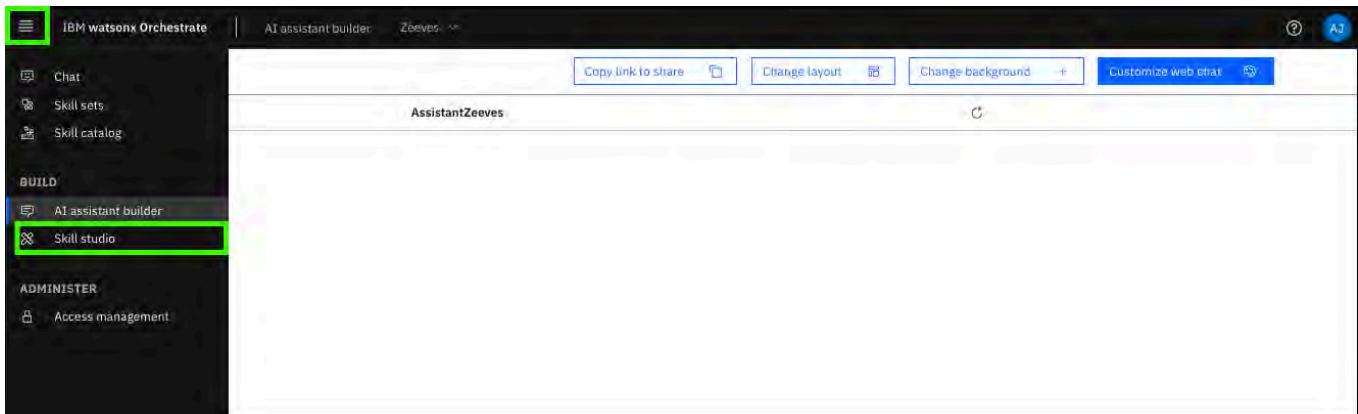
## Import the Ansible automations into watsonx Orchestrate

For this use case, configure the assistant to guide the user through the process of identifying their SSL certificate's expiration date and automate the certificate renewal process. To do so, import the needed AAP templates into watsonx Orchestrate as skills.

For this use case, the ansible templates you import are:

- z/OS Certs – List Cert
- z/OS Certs – Search and Renew
- Retrieve job output (utility skill)

1. Open **Skill studio** in watsonx Orchestrate.



2. Click **Create** and then click **Import API**.

3. Click the **z/OS Skills accelerator (Trial)** tile.

4. Enter the following values in the **z/OS Skills accelerator** form and then click **Connect**.

Use the **URL**, **User Name**, and **Password** values recorded in the [Explore Ansible Automation Platform](#) section earlier.

**a:** Connection Type:

```
ansible
```

**b:** Application Name:

```
certs
```

**c:** Connection URL:

<Enter the URL for your AAP UI>

**d:** User Name:

<Enter the AAP User Name (for UI access)>

**e:** Password:

<Enter the AAP User Password>

**f:** Search Pattern:

```
*
```

The screenshot shows the 'Add skills' dialog in the Skill studio. On the left, there are two tabs: 'Choose the source' (selected) and 'Select the skills'. Below these are three buttons: 'From an app', 'From a file', and 'OpenAPI builder (experimental...)'. To the right, a large panel titled 'z/OS Skills accelerator (Trial)' contains fields for 'Connection Type' (set to 'ansible'), 'Application Name' (set to 'certs'), 'Connection URL' (set to 'https://itzvsi-aap-pwgabob.techzone.ibm.com'), 'User name' (set to 'admin'), 'Password' (redacted), and 'Search Pattern' (set to '+'). A note at the bottom states: 'Necessary for "ansimf" connections. Pattern cannot be "" or "+"'. At the bottom right are 'Cancel' and 'Connect' buttons, with 'Connect' highlighted.

## 5. Expand the Ansible Job Template Proj... folder and then click aap4zos.

The screenshot shows the 'Select the skills' dialog. It displays a tree view of skill sets. The 'Ansible Job Template Proj...' folder is expanded, showing its contents: 'Demo Project', 'z\_ansible\_collections\_sa...', 'ansible\_zos', and 'aap4zos'. The 'aap4zos' folder is also expanded, showing its contents: 'Ansible Utility Skills'. At the top of the dialog, there is a note: 'Select the skills you want to add to the skill set. Currently, 0 of 0 skills are selected.'

## 6. Select Z/os certs - list cert and Z/os certs - search and renew and then click Save as draft.



Scroll through the table of skills to find the required skills.

**Add skills**

Choose how you want to add skills and then select the skills you want to refer to from that source.

**Select the skills**

Select the skills you want to add to the skill set. Currently, 2 of 20 skills are selected.

Skill	Description	Status
Z/os certs - delete cert	Certs - this playbook deletes ...	Ready to add
<input checked="" type="checkbox"/> Z/os certs - list cert	Certs - this playbook displays...	Ready to add
<input checked="" type="checkbox"/> Z/os certs - search and renew	Certs - this playbook searches...	Ready to add
Z/os tso command(s)	Certs - a playbook to execute ...	Ready to add
Z/os operator command	Certs - a playbook to execute ...	Ready to add

Items per page: 5 | 16-20 of 20 items

Cancel Save as draft

7. Click the ellipses (⋮) for the **z/OS Certs - List Cert** skill and select **Enhance this skill**.

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

**Skills**

Name	Step in the process	Status	Skill type	Author	Last edited
z/OS Certs - Search and Renew	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	January 19 2025
z/OS Certs - List Cert	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	January 19 2025
z/OS IPL Information	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025
Zeeves-gather-facts-flow	Ready to use	Published	Skill flow	andrew@jones-tx.com	January 17 2025
Retrieve job output	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025
z/OS Ping	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025
z/OS Gather Facts	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025

Find a skill

Create

Configure prebuilt skills

Enhance this skill

Export this skill

Delete this skill

8. Review the skill enhancement options and then click **Publish**.

The screenshot shows the 'Skill studio' interface in IBM Watsonx Orchestrate. The current skill being enhanced is 'z/OS Certs - List Cert'. The 'Name' field is populated with 'z/OS Certs - List Cert'. The 'Description' field contains the text: 'certs - This playbook displays a certificate's details.' The 'API version' is set to '1.0'. Under 'Categories', there is a placeholder 'Add categories...'. In the 'App' section, 'Ansible Controller Skills > certs' is listed. On the right side, there are two preview sections: one for the catalog view showing the skill card with its name and a brief description, and another for the skill set view showing the skill card. At the bottom right, there are three buttons: 'Cancel', 'Publish' (which is highlighted with a green border), and 'Save as draft'.

9. Repeat steps 7 and 8 for the **z/OS Certs - Search and Renew** skill.

10. **Challenge:** You also need to add the **Retrieve job output** utility to your **certs** app just like you did when creating the **Gather Facts** skill flow. Repeat steps 2 - 8 to add the **Retrieve job output** utility skill to your **certs** app.

Verify all the skills are successfully imported and create the app connection.

1. Open **Skill catalog** in watsonx Orchestrate.

The screenshot shows the IBM Watsonx Orchestrate interface with the 'Skill studio' tab selected. On the left, a sidebar lists 'Chat', 'Skill sets', and 'Skill catalog' under 'Skill studio'. Below this are sections for 'BUILD' (AI assistant builder) and 'ADMINISTER' (Access management). The main area is titled 'Skill catalog' and contains a table of skills. The columns are 'Step in the process', 'Status', 'Skill type', 'Author', and 'Last edited'. There are three rows of skills, all marked as 'Published' and 'Imported' by 'andrew@jones-tx.com' on January 19, 2025.

Step in the process	Status	Skill type	Author	Last edited
	Ready to use	Published	Imported	andrew@jones-tx.com
	Ready to use	Published	Imported	andrew@jones-tx.com
	Ready to use	Published	Imported	andrew@jones-tx.com

## 2. Enter certs in the search bar.

The screenshot shows the 'Skill catalog' page with a search bar at the top containing the text 'certs'. Below the search bar, there's a heading 'Most popular skills' followed by three cards: 'Send an email from Gmail', 'Create a lead in Salesforce', and 'Send an email using Outlook'. Underneath these, there's a section for 'All Apps' with several app tiles visible, including 'z/OS operations', 'Coupa', 'ZoomInfo', and 'HubSpot CRM'.

## 3. Click the certs tile.

The screenshot shows the 'Skill catalog' page with the search bar still containing 'certs'. Below the search bar, there's a heading 'Apps' followed by a single card titled 'Ansible Controller Skills - certs' which contains '3 skills'. This card is highlighted with a green box.

## 4. Click Add skill + for each of the 3 skills in the certs app.

Skill catalog /

## Ansible Controller Skills - certs (3)

Personal skills

Search skills:

Ansible Controller Skills - certs

- Retrieve job output  
certs - Retrieve job output by job Id.
- z/OS Certs - List Cert  
certs - This playbook displays a certificate's details.
- z/OS Certs - Search and Renew  
certs - This playbook searches for certificates with the details specified in the AAP survey and renew...

Add skill + Add skill + Add skill +

5. Click **Connect app**.

Skill catalog /

## Ansible Controller Skills - certs (3)

Personal skills

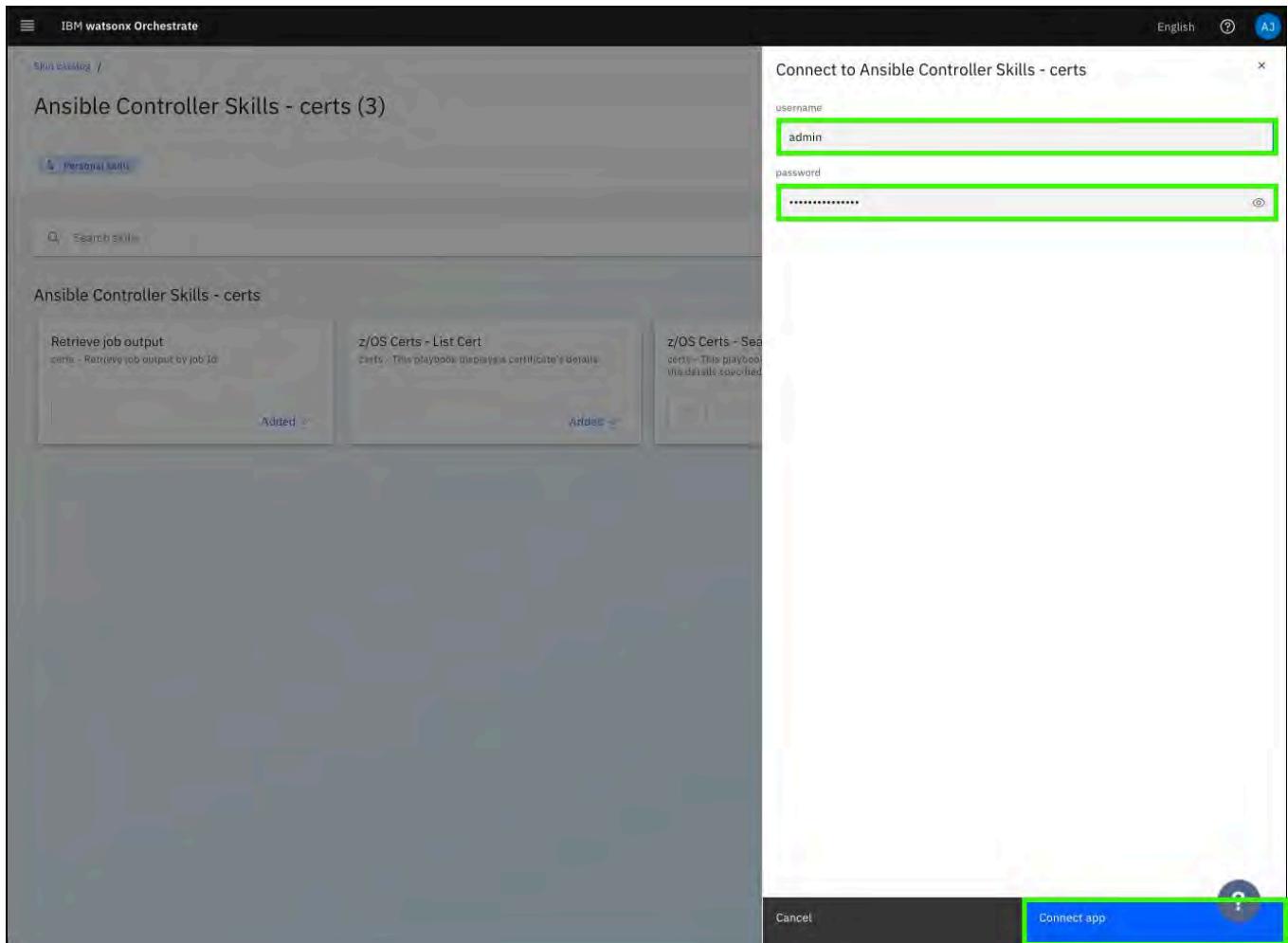
Search skills:

Ansible Controller Skills - certs

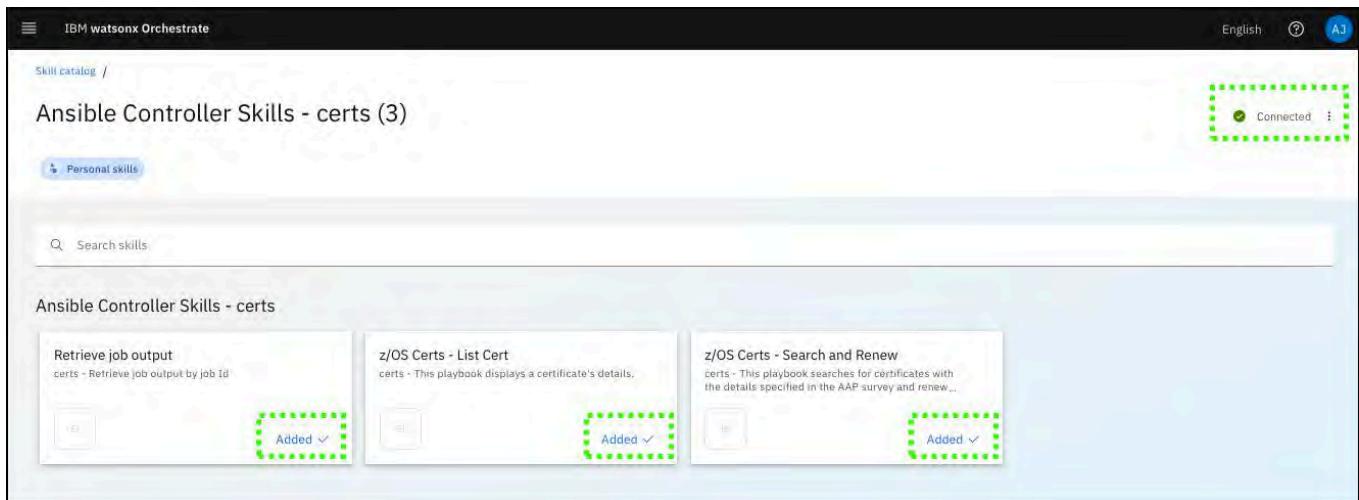
- Retrieve job output  
certs - Retrieve job output by job Id.
- z/OS Certs - List Cert  
certs - This playbook displays a certificate's details.
- z/OS Certs - Search and Renew  
certs - This playbook searches for certificates with the details specified in the AAP survey and renew...

Added ✓ Added ✓ Added ✓

6. Enter your **AAP Username** and **AAP Password** and then click **Connect app**.



7. Verify that the app is connected.



Connect the app to the assistant.

1. Open **Skill catalog** in watsonx Orchestrate.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with sections for Chat, Skill sets, Skill catalog, BUILD (AI assistant builder, Skill studio), and ADMINISTER (Access management). The 'Skill sets' section is highlighted with a green box. The main area is titled 'Skills - certs (3)' and contains three cards: 'z/OS Certs - List Cert' (Added ✓), 'z/OS Certs - Search and Renew' (Added ✓), and another partially visible card.

## 2. Select the Draft version of your assistant and click Connections.

The screenshot shows the 'Zeeves draft' skill set page. The 'Skill sets' dropdown is set to 'Zeeves draft'. Below it, there are tabs for 'Skills' (highlighted with a green box) and 'Connections' (also highlighted with a green box). The 'Skills (3)' section lists three skills: 'z/OS IPL Information', 'Zeeves-gather-facts-flow', and 'Retrieve job output'. At the bottom, there's a search bar with 'certs' typed in, and a 'Manage skills' button.

## 3. Enter certs in the search bar.

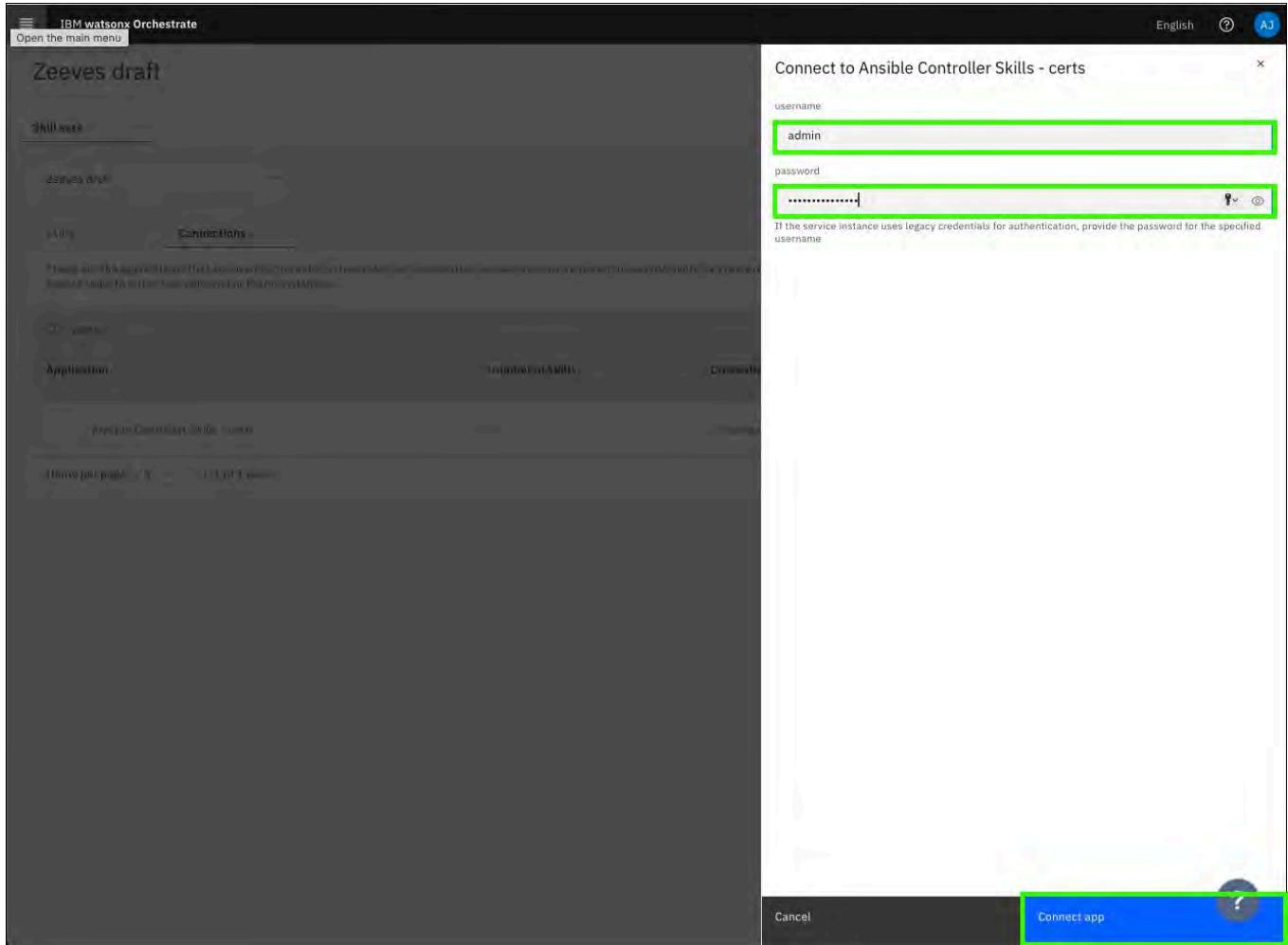
The screenshot shows the 'Zeeves draft' skill set page again. The 'Skill sets' dropdown is still set to 'Zeeves draft'. The 'Connections' tab is highlighted with a green box. A search bar at the top has 'certs' typed in. Below the search bar, there's a table with columns: Application, Number of skills, Credential type, Connected by, and Action. One row in the table is for 'Ansible Controller Skills - certs'. At the bottom, there's a search bar with 'certs' typed in, and a 'Manage skills' button.

4. Click the ellipses (⋮) for the certs app and select Connect app\*.

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - certs	3	Not specified	-	<button>Connect app</button>

5. Click Connect app.

6. Enter your AAP Username and AAP Password and then click Connect app.

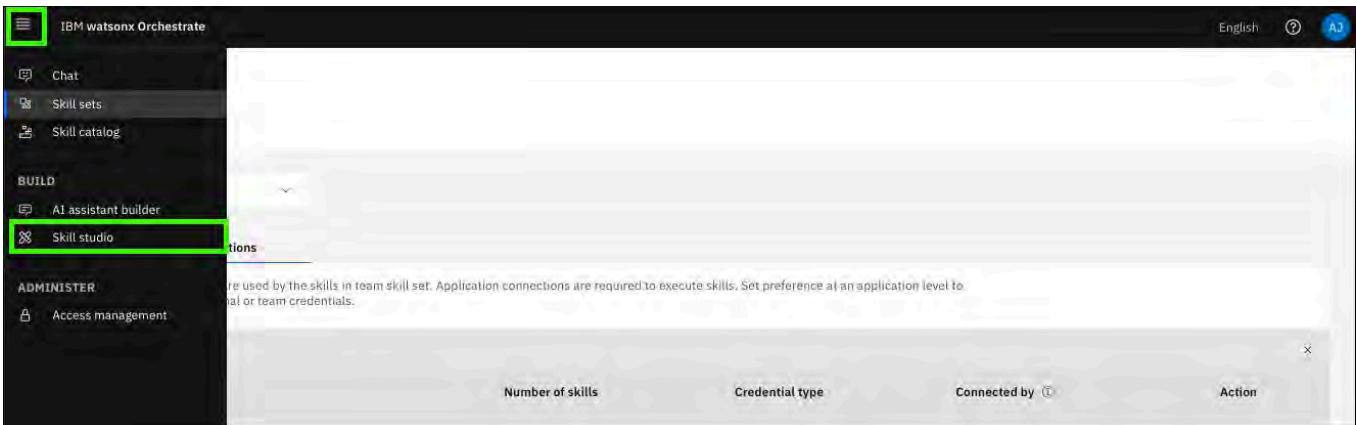


## Create a skill flow to retrieve certificate expiration dates.

The goal of this scenario is to configure the assistant to automate the certificate renewal process for the client. The first step in that process is to help the SA identify the expiration date of their z/OS certificate. You have imported the **z/OS Certs – List Cert** skill from Ansible Automation Platform. Next, create a new skill flow that uses the **z/OS Certs – List Cert** skill that can later be used in a natural conversation through assistant actions.

First, create a skill flow to retrieve and display the expiration date of a z/OS certificate based on the certificate label the user provides.

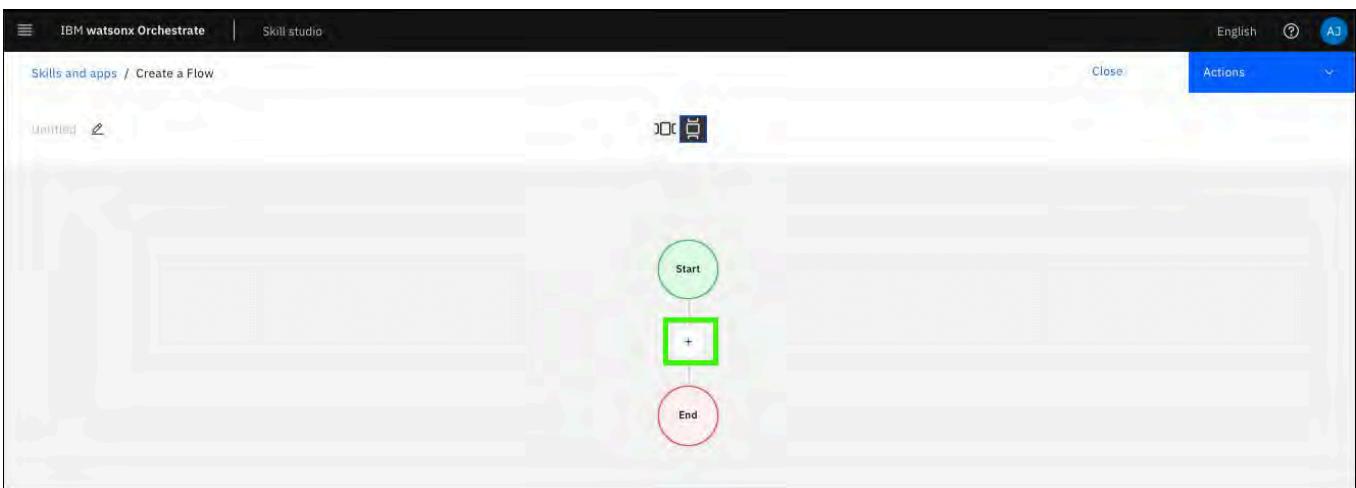
1. Open **Skill studio** in watsonx Orchestrate.



2. Click **Create** and then click **Skill flow**.



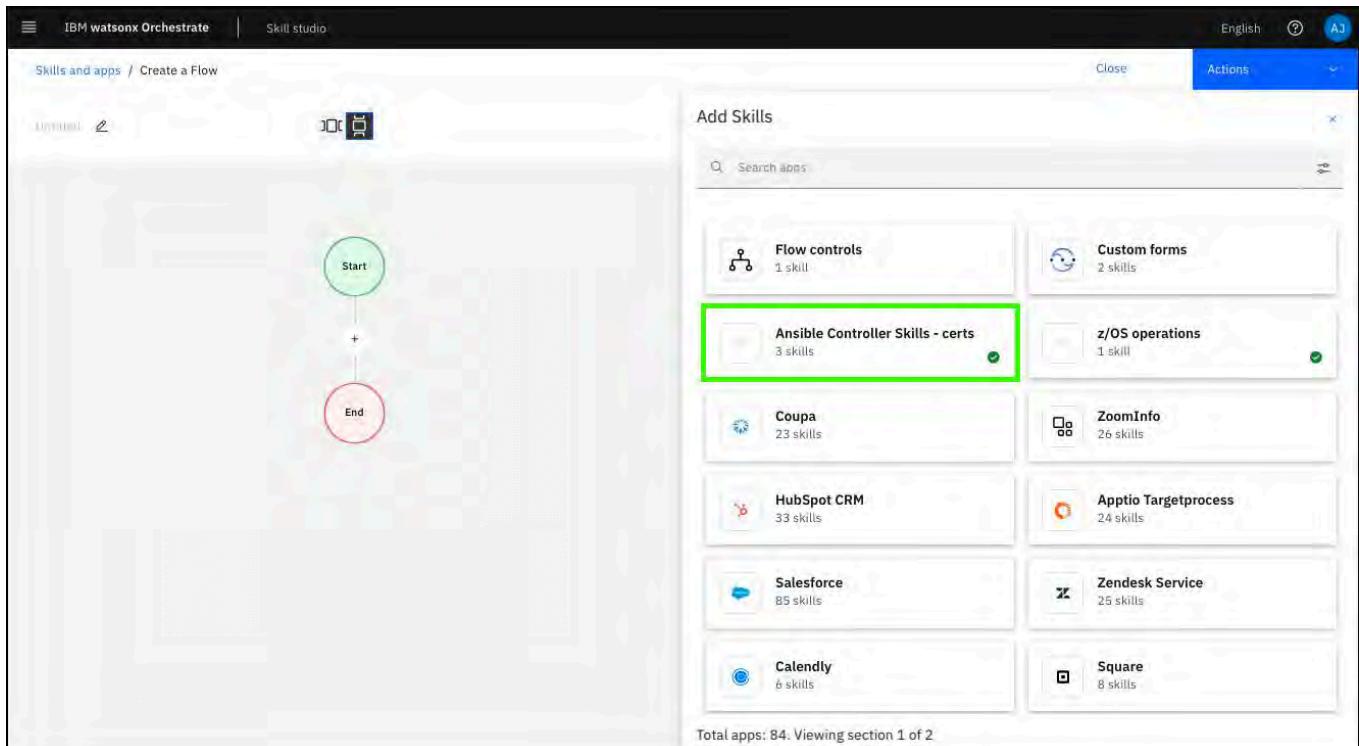
3. Click the **+** icon.



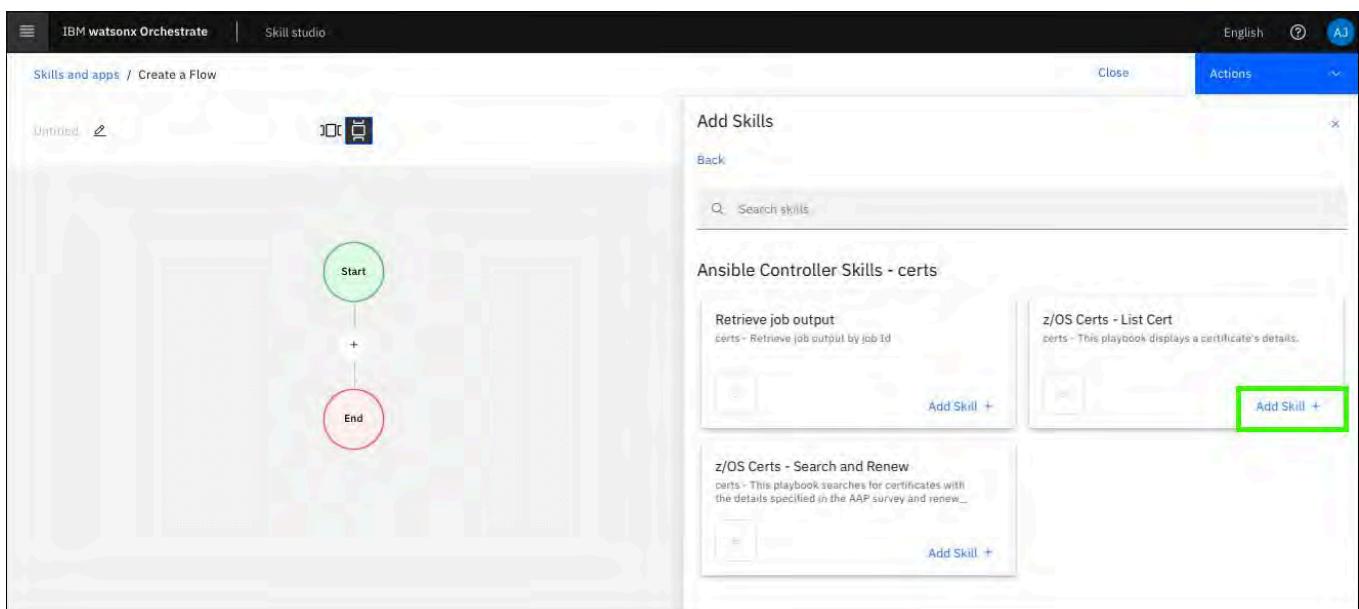
4. Click the **certs** app.



Search on certs if you do not see the tile for your app.

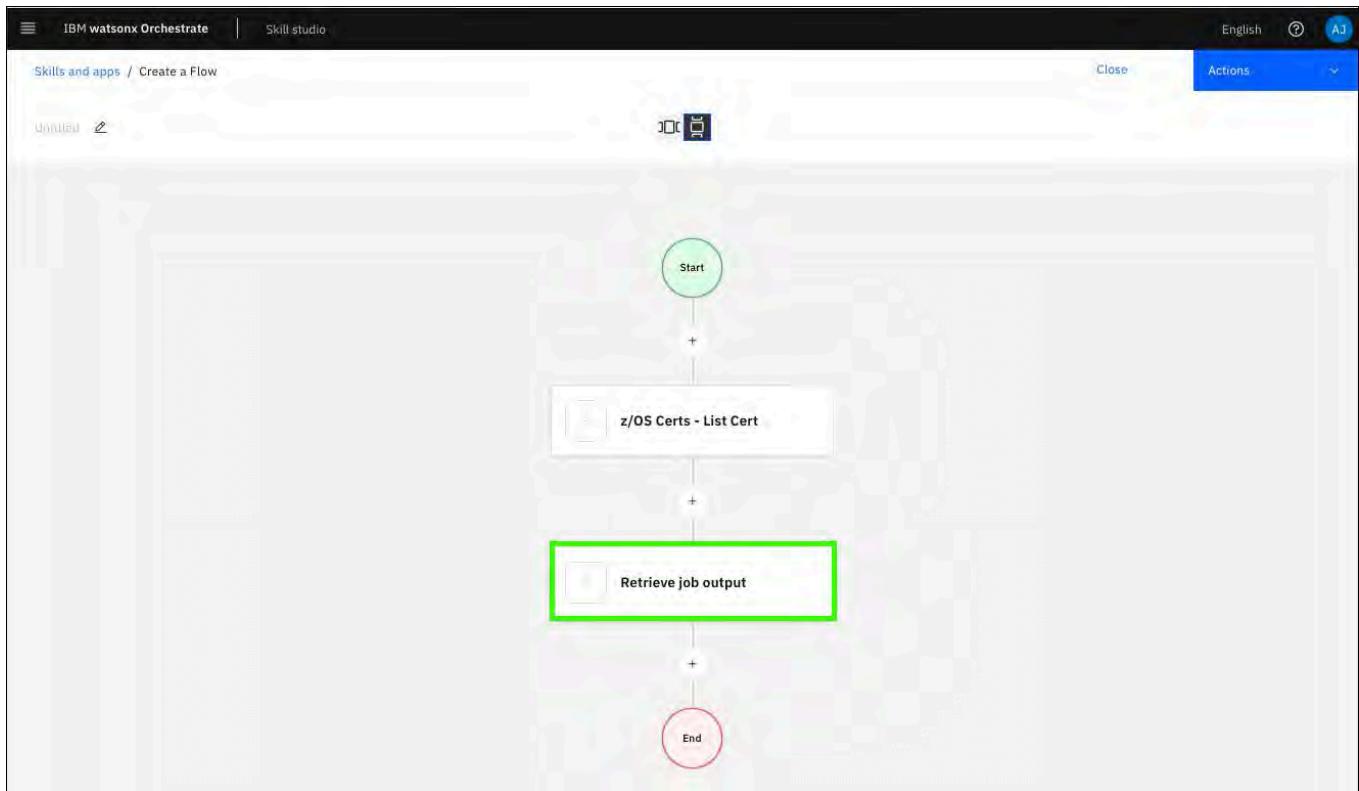


5. Click Add Skill + in the z/OS Certs - List Cert tile.



6. Click the + icon **below** the z/OS Certs - List Cert skill and repeat steps 4 and 5 to add the **Retrieve job output** skill.

7. Click the **Retrieve job output** skill.



8. Click the **id** field.

**Ansible Controller Skills - certs**

certs - Retrieve job output by job Id

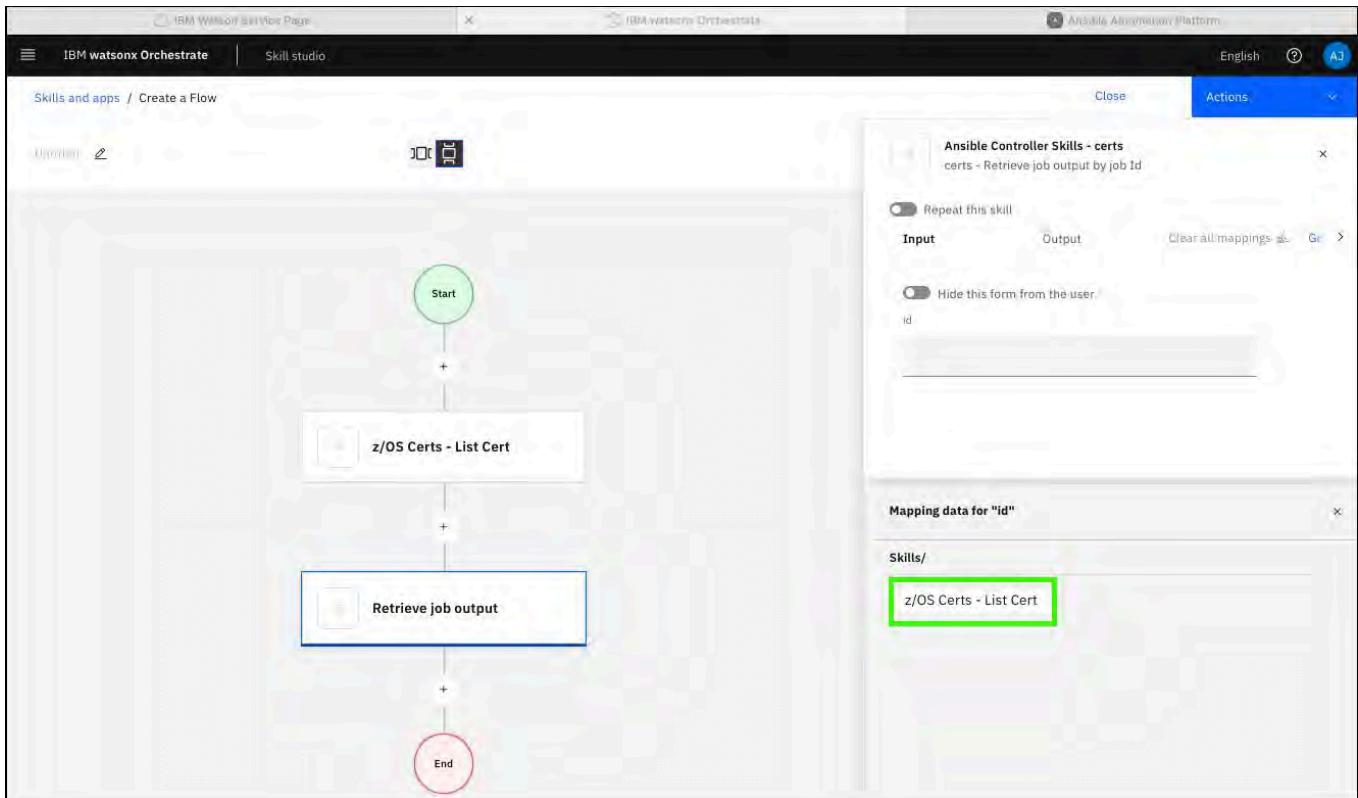
Repeat this skill

**Input**      **Output**      [Clear all mappings](#) [Get >](#)

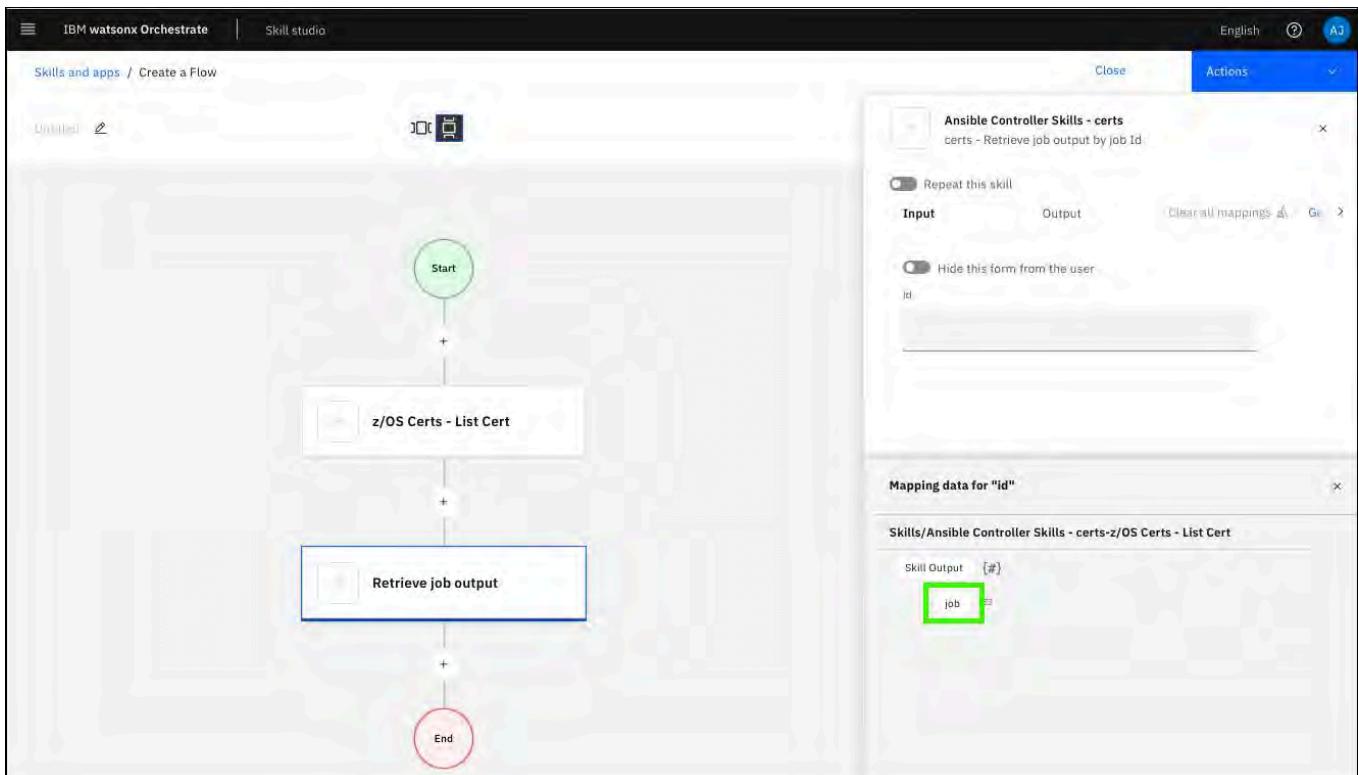
Hide this form from the user

**id**

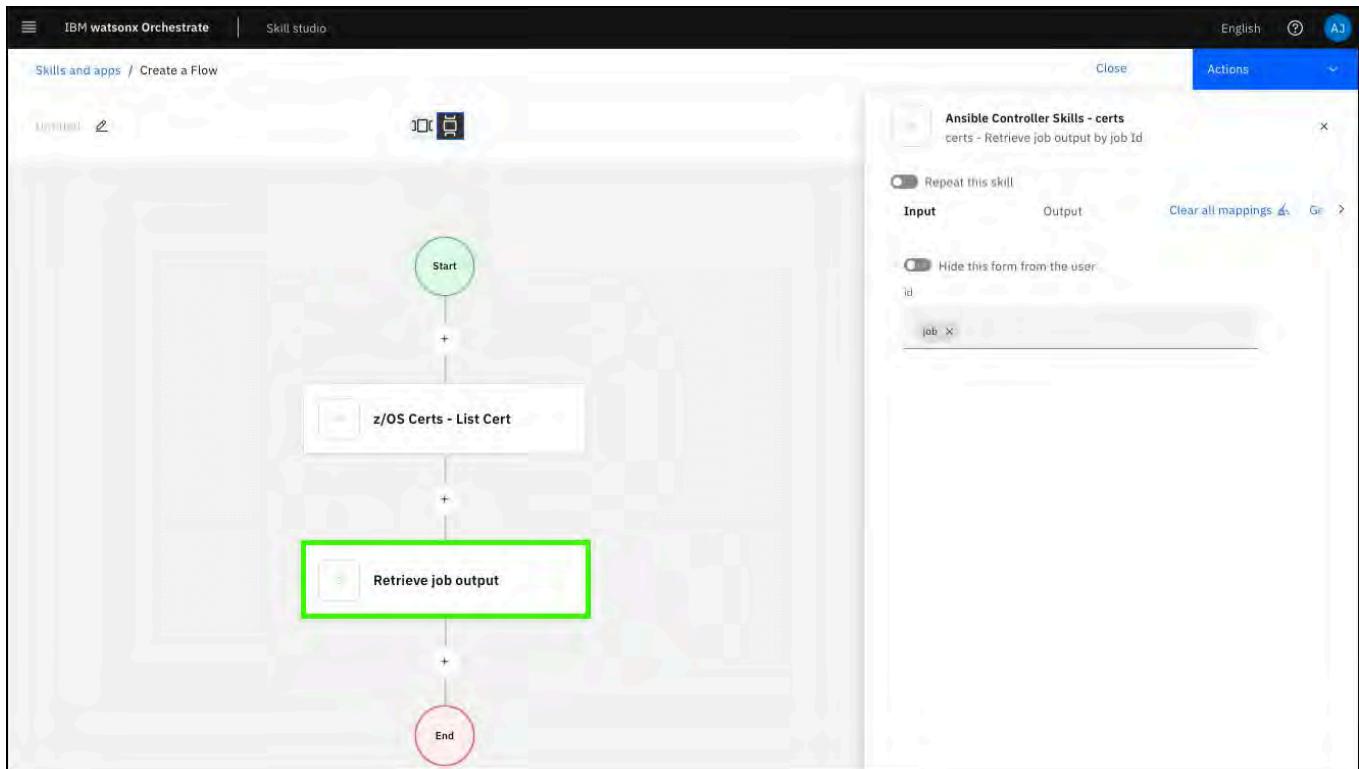
9. Click **z/OS certs - List Cert**.



10. Click job.

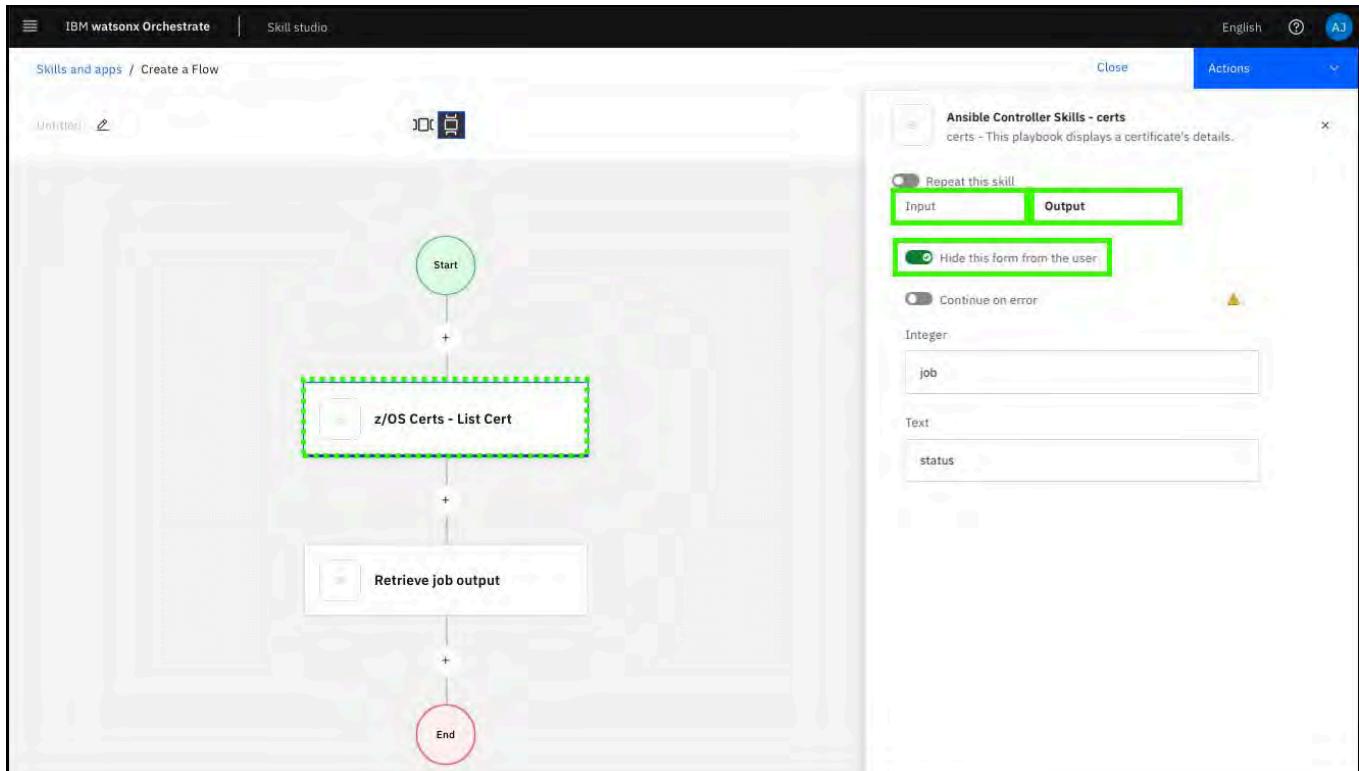


11. Click the z/OS Certs - List Cert tile.

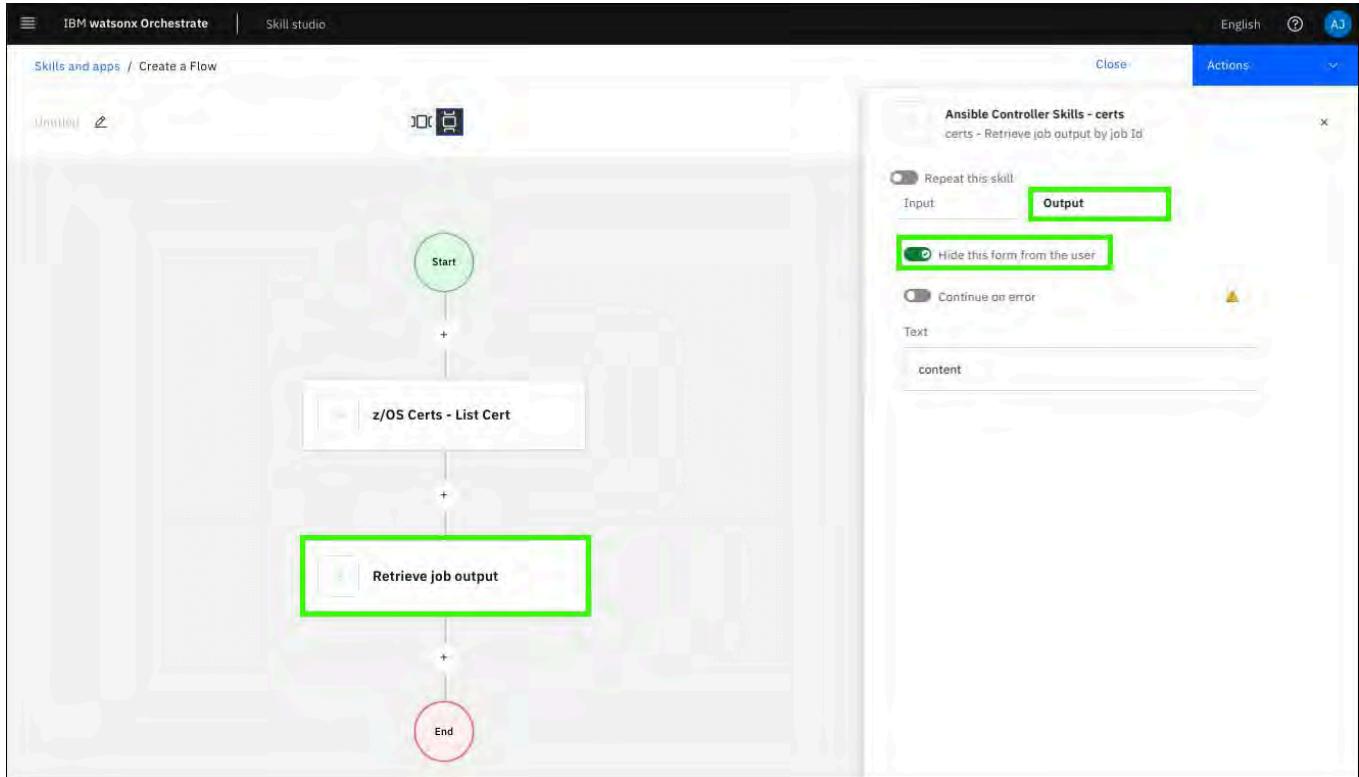


12. On both the **Input** and **Output** tabs for the **z/OS Certs - List Cert** skill, enable the **Hide this form from the user** options.

To enhance the user experience, hide the input and output forms from the user. This disables the List Cert skill form from being displayed. Rather than the user entering in their certificate details as input to the skill form, those details can be gathered into the skill through user prompts when creating an assistant action. This enables a more natural conversation flow when interacting with the assistant.

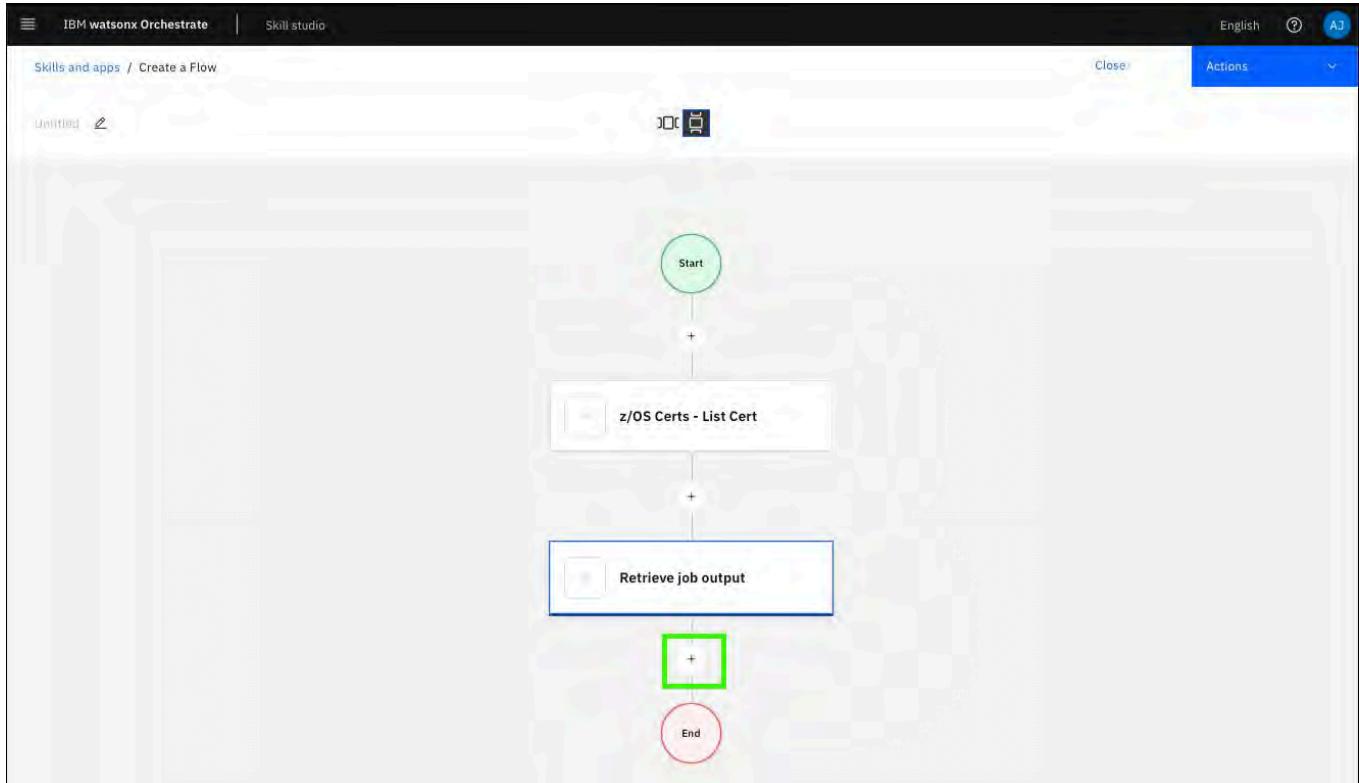


13. Repeat step 12 for the **Output** of the **Retrieve job output** skill.

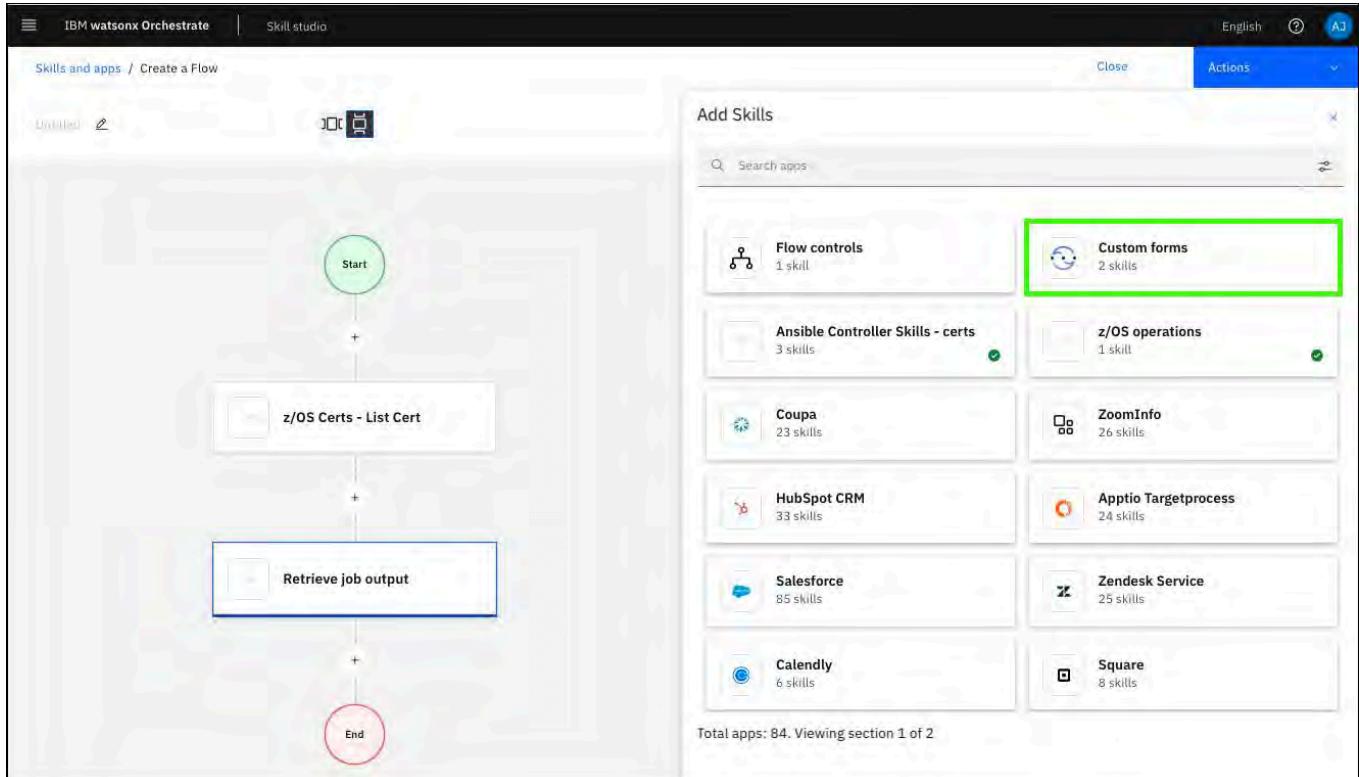


The output of the **List Cert** skill includes a large amount of data. In the assistant, only the **certificate expiration date** is needed. In the next steps, transform the output to return only the **certificate expiration date**.

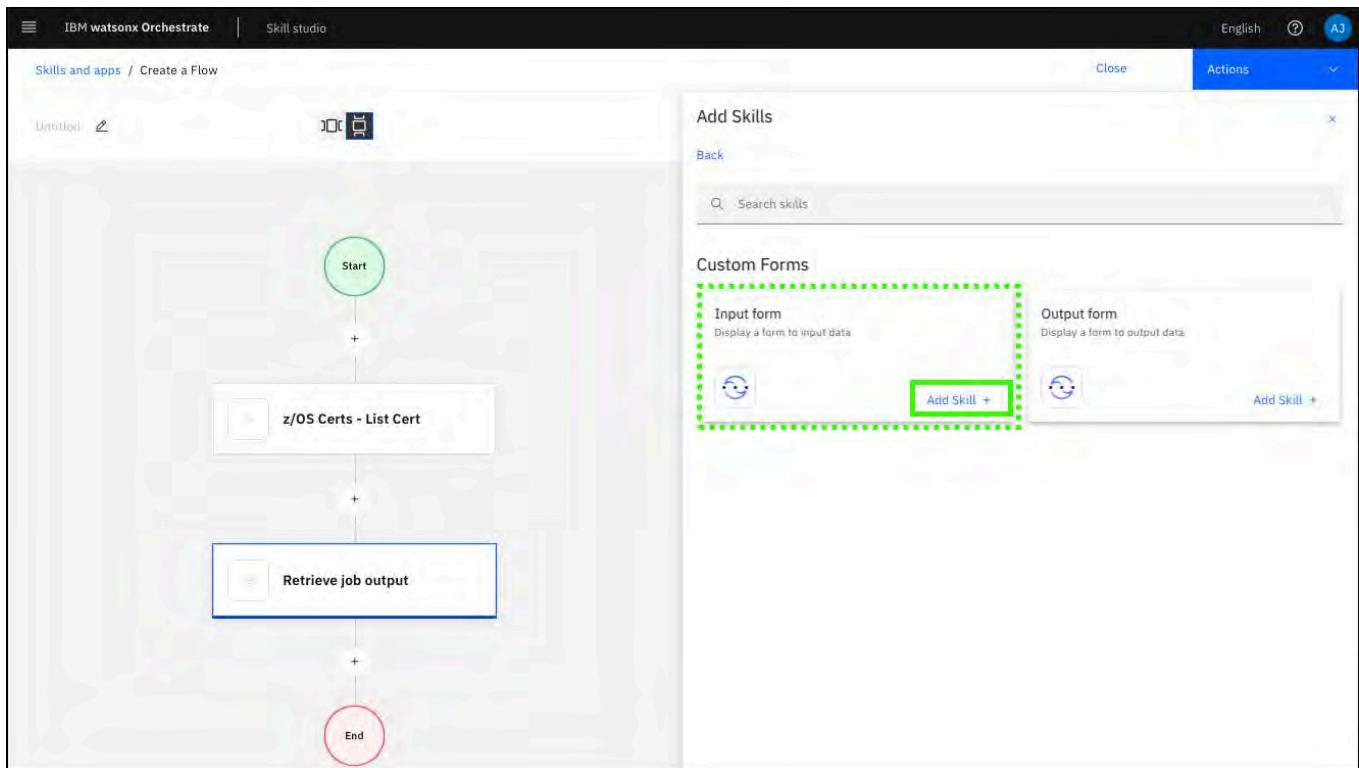
14. Click the **+** icon **below** the **Retrieve job output** skill.



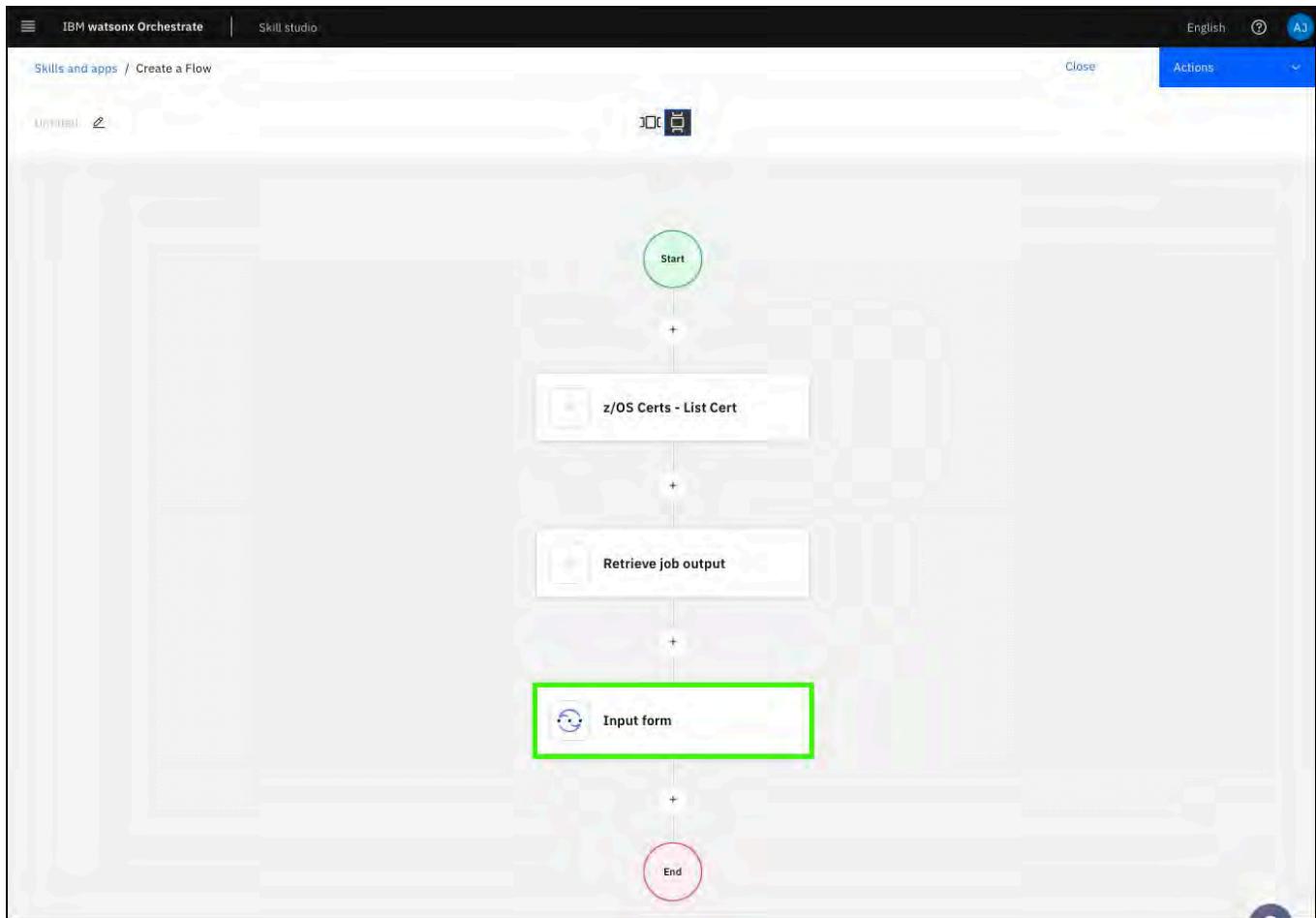
15. Click the **Custom forms**.



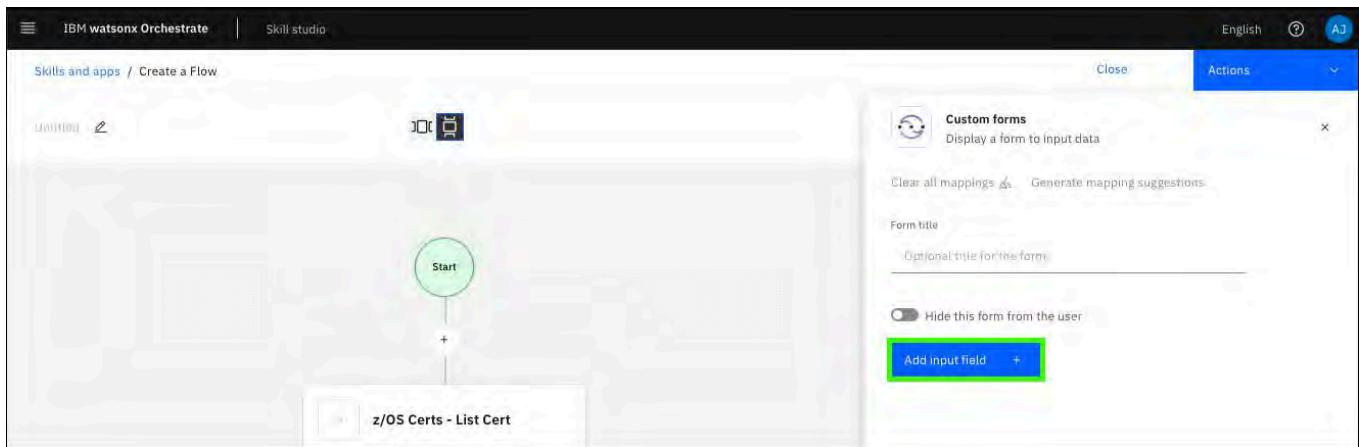
16. Click Add skill + in the Input form.



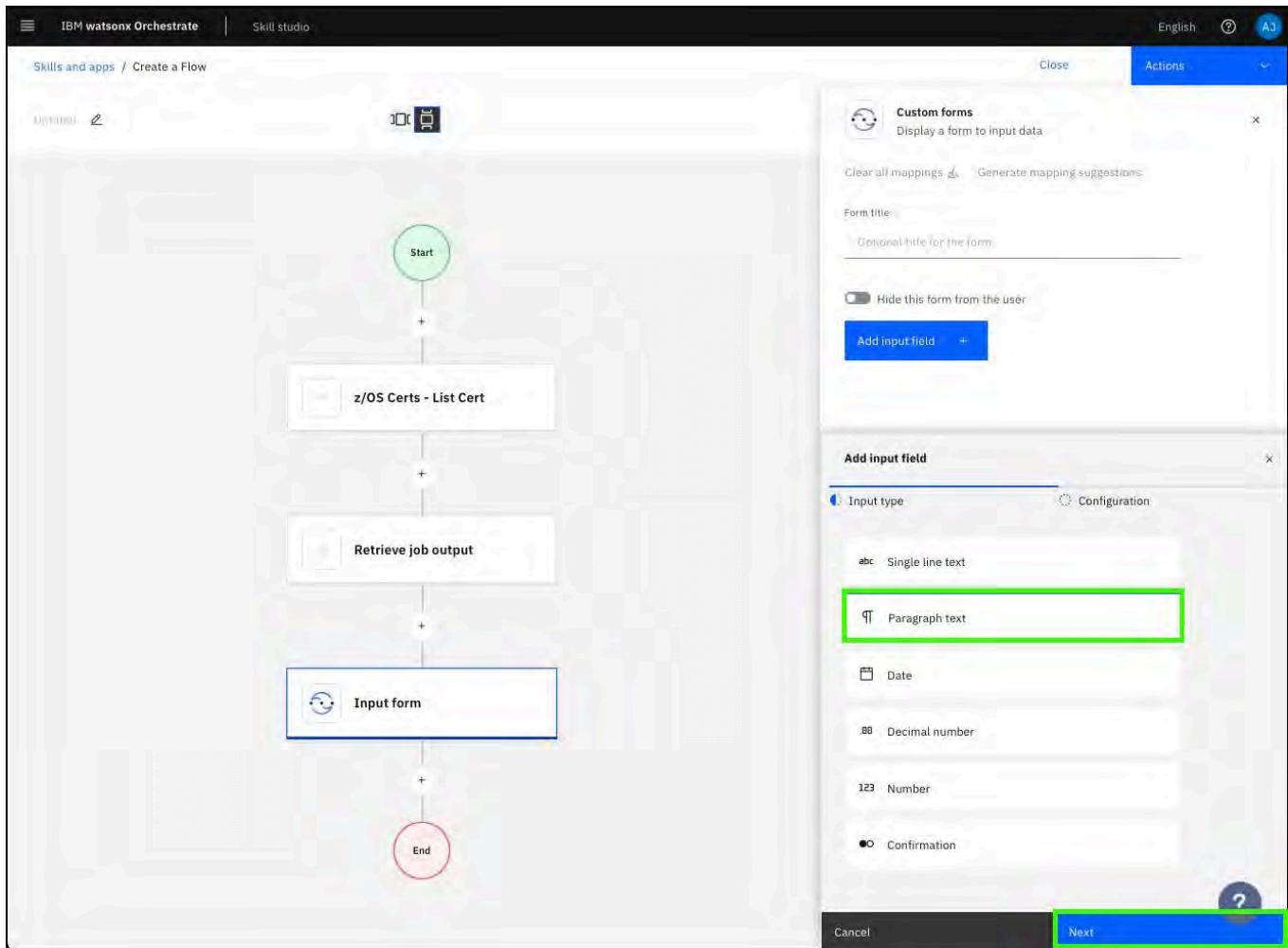
17. Click the **Input form** skill.



18. Click Add input field +.



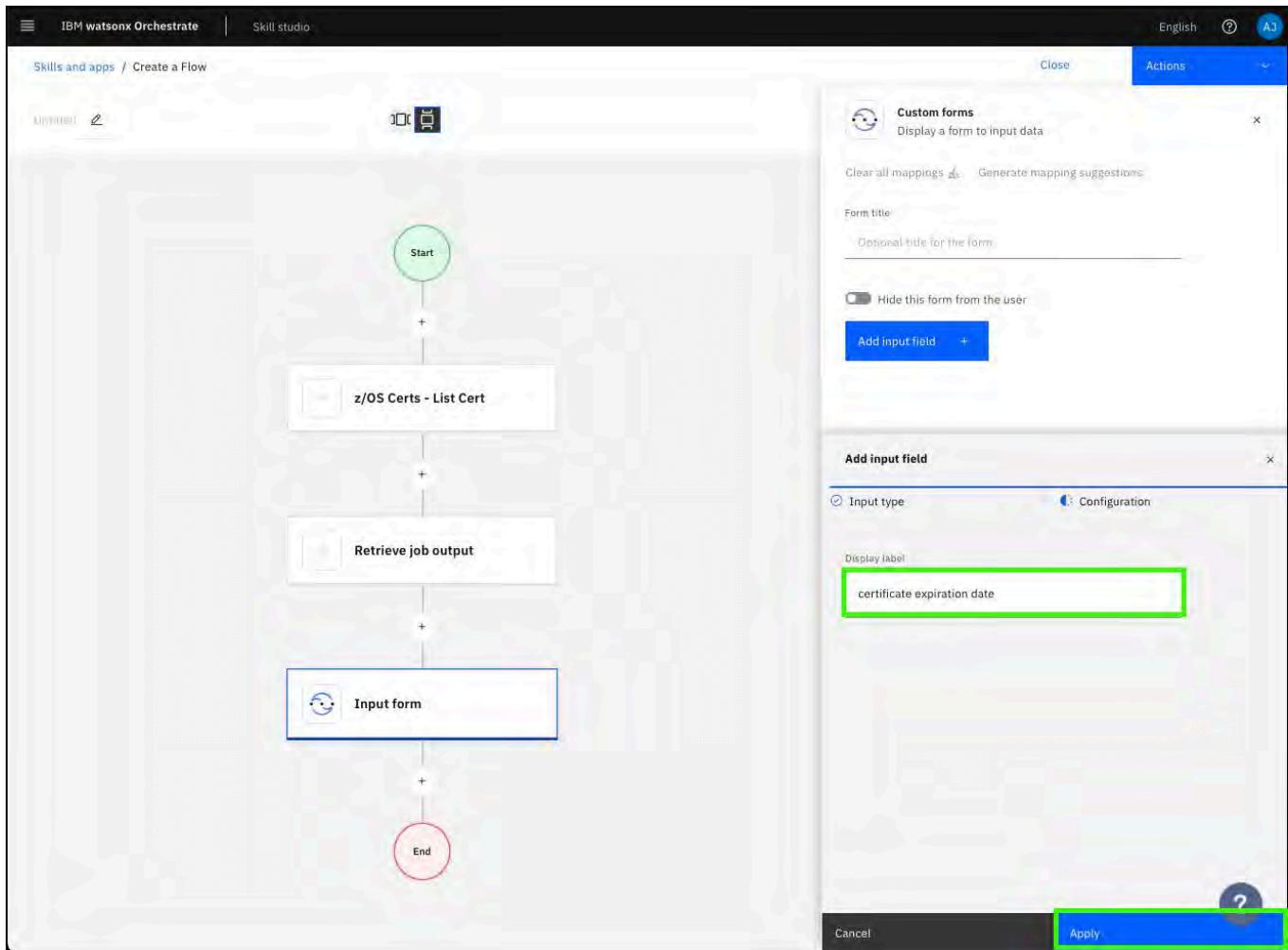
19. Click Paragraph text and then click Next.



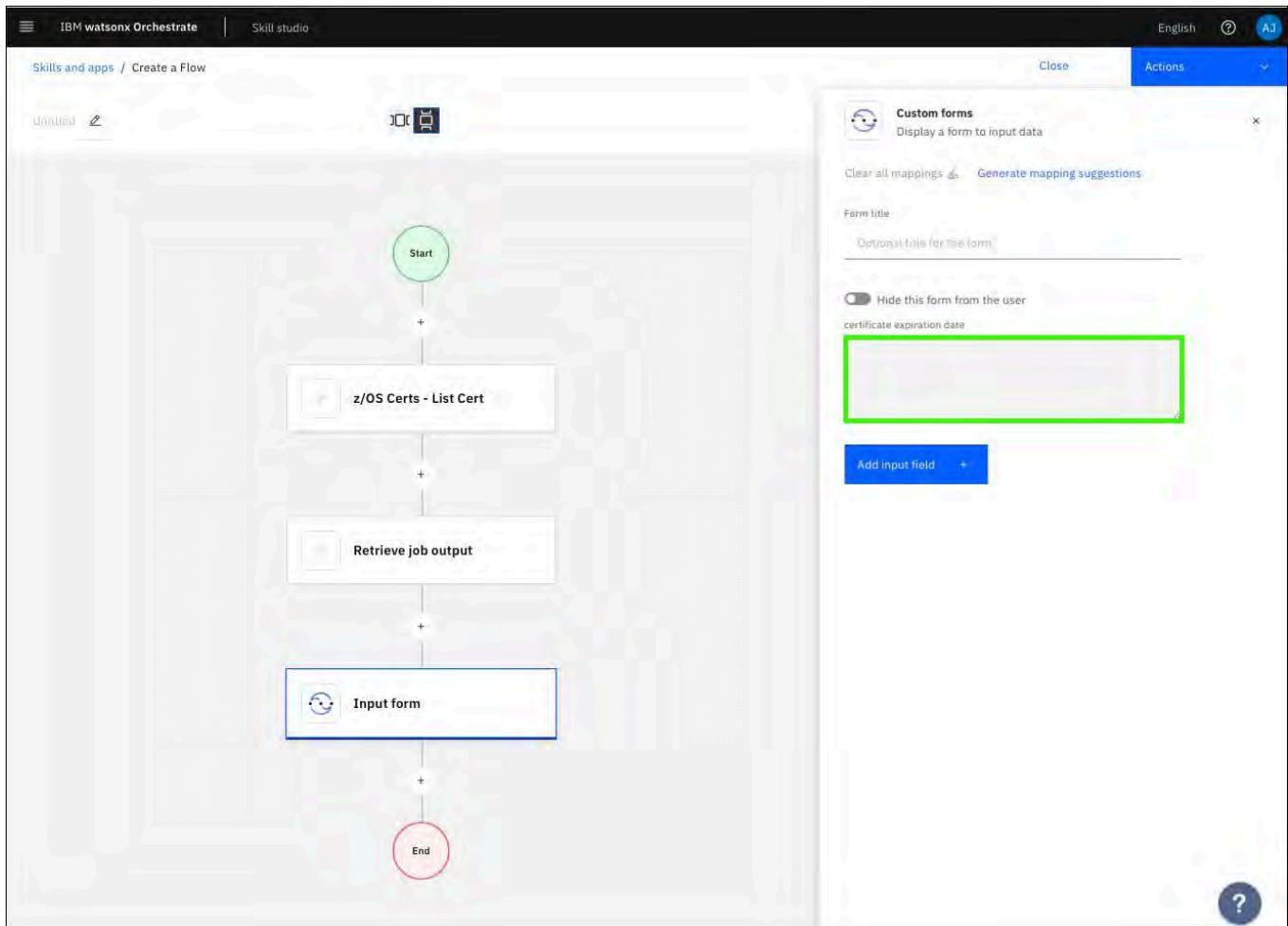
20. Enter certificate expiration date in the **Display label** field and click **Apply**.

**Display label:**

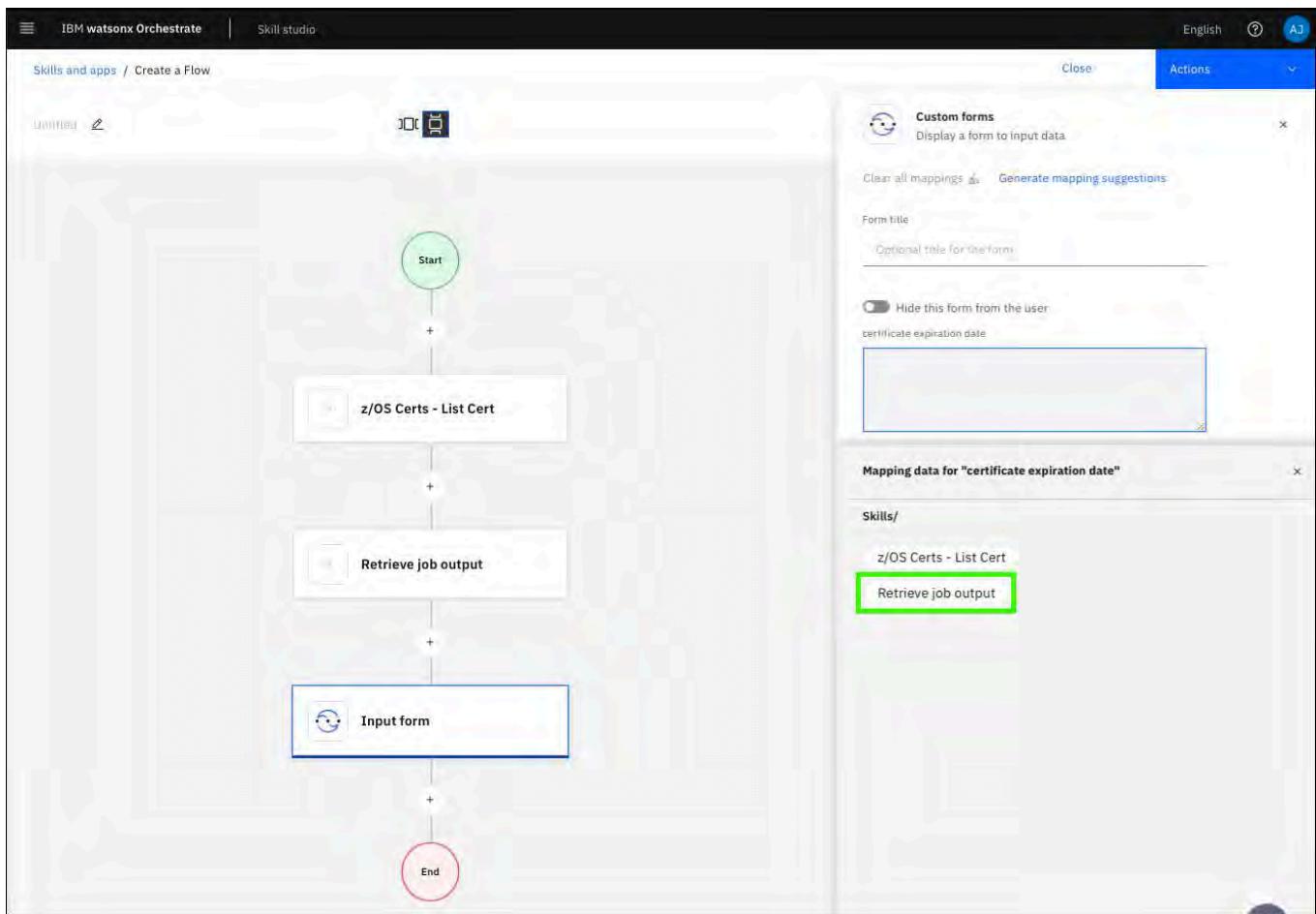
certificate expiration date



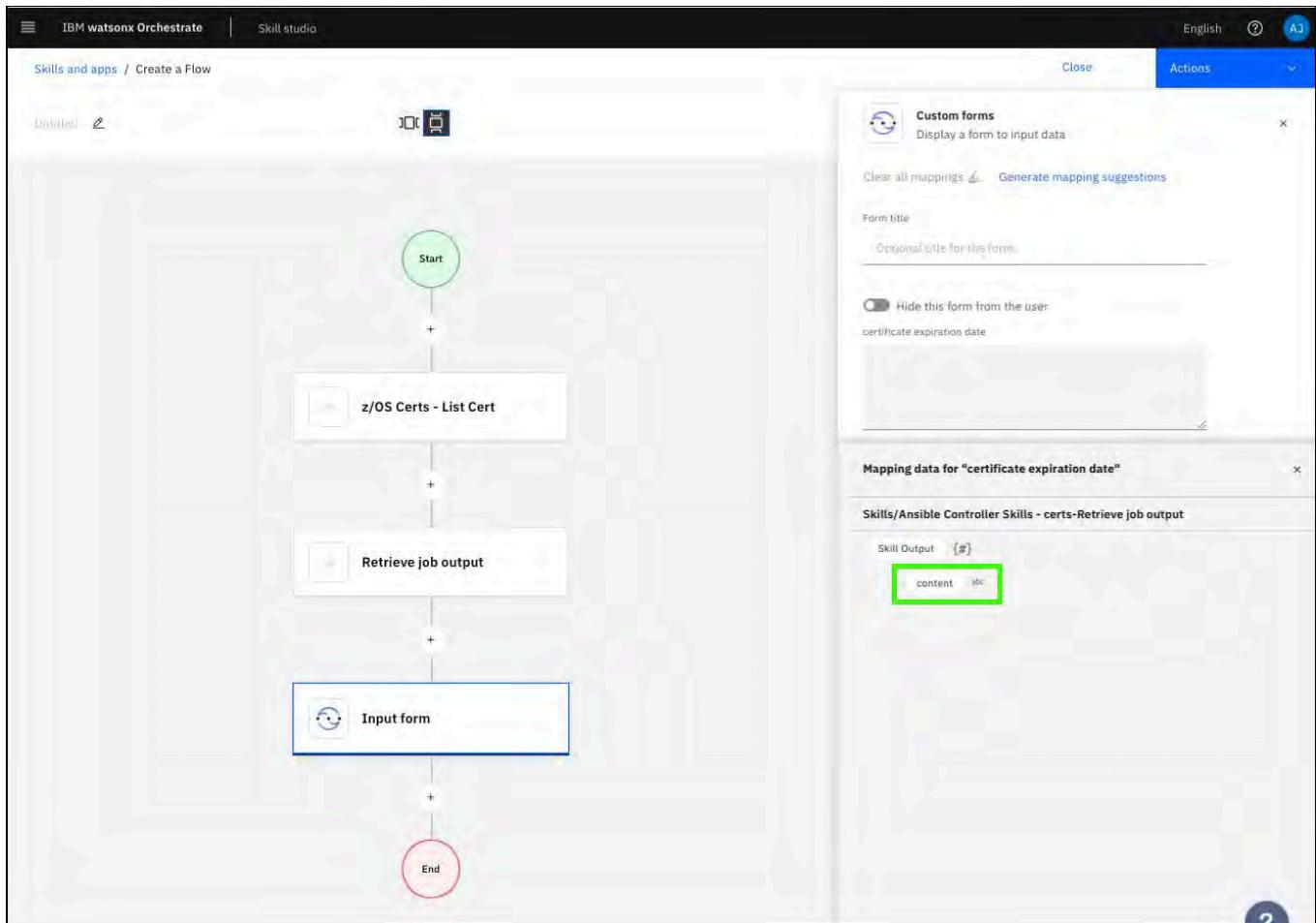
21. Click the **certificate expiration date** entry field.



22. In the **Mapping data for "certificate expiration data"** section, click **Retrieve job output**.

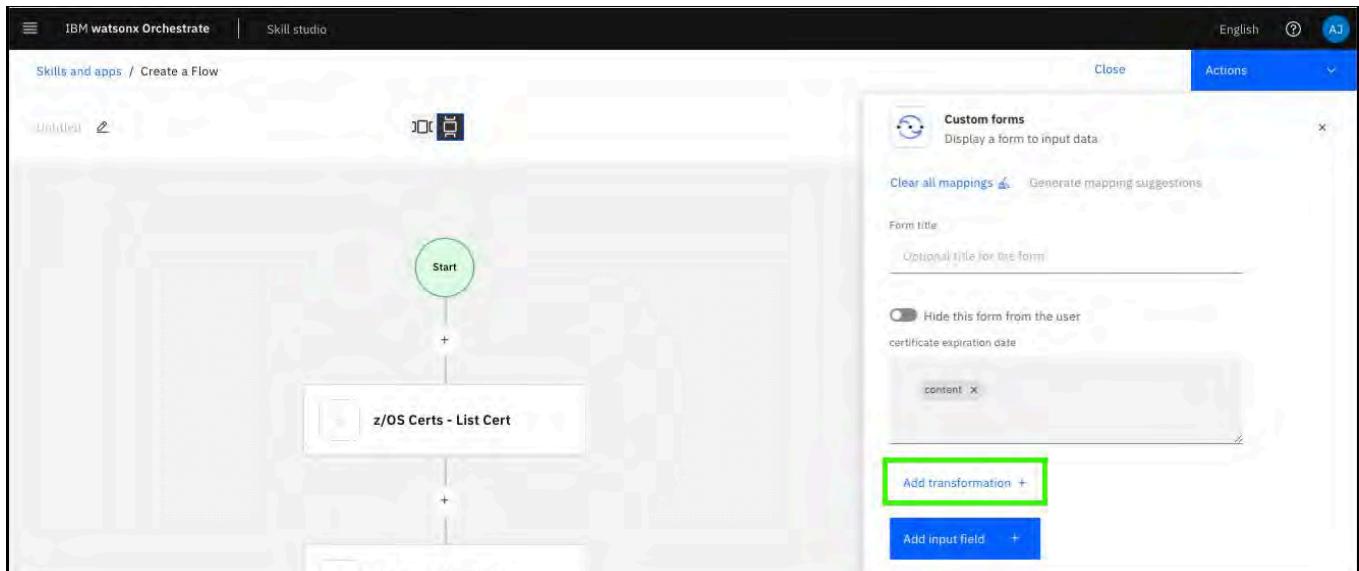


23. Click content.

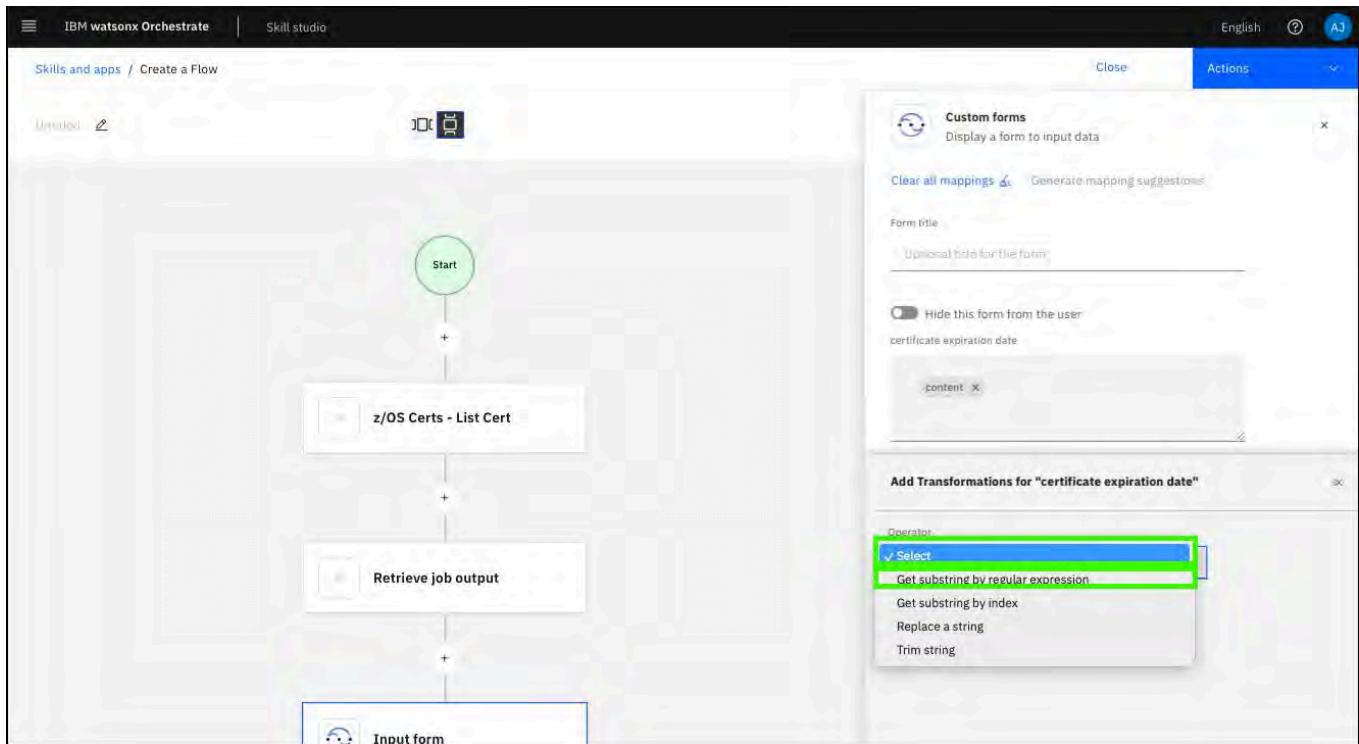


24. Click **Add transformation +**.

A transformation is used to extract the **certificate expiration date** from all the job output data.



25. Click the **Select** drop-down and select **Get substring by regular expression**.



26. Cut and paste the *regular express* that follows to extract the certificate end date and then click **Add**.

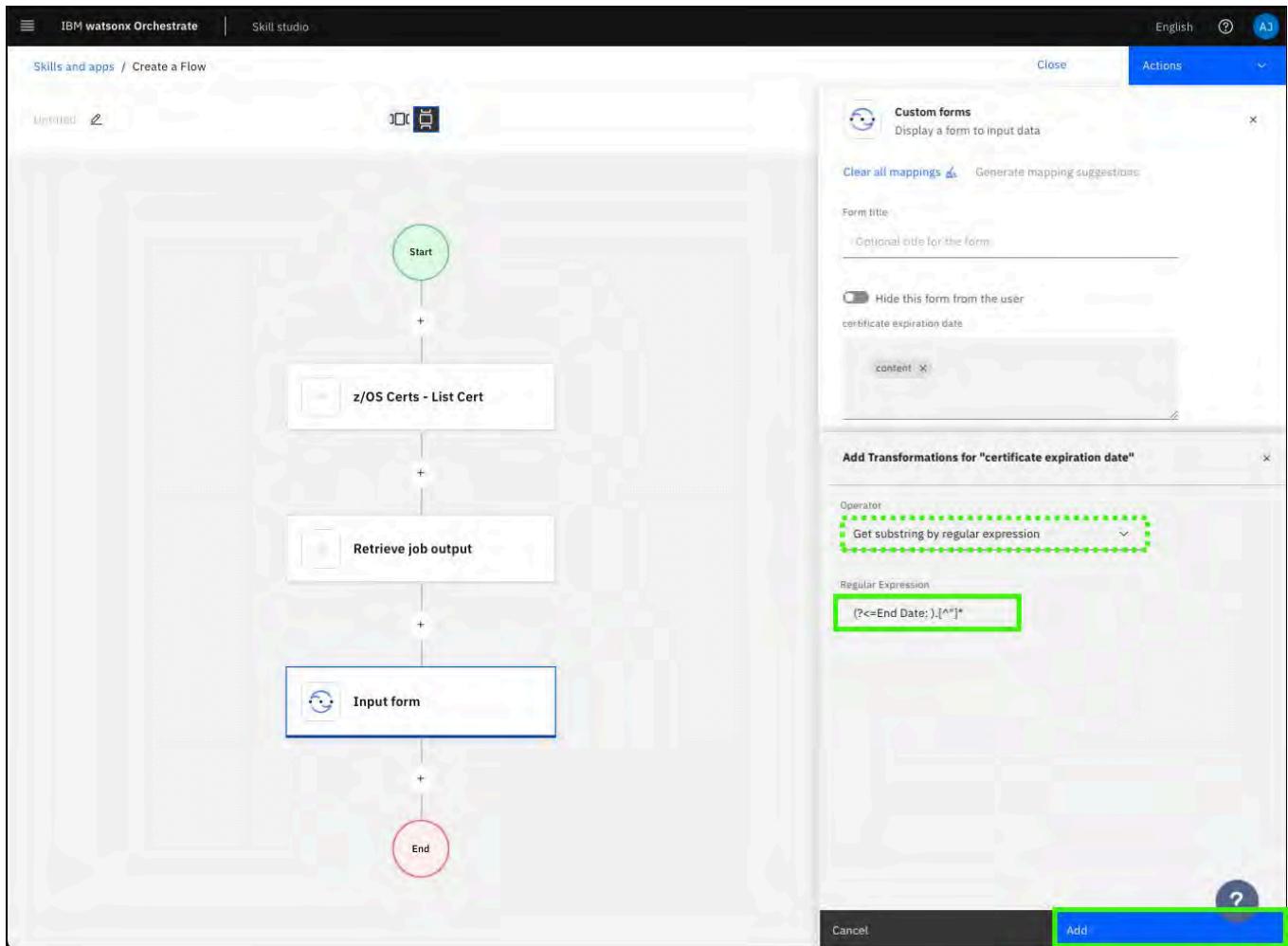
#### Regular Expression:

```
(?=<End Date: ).[^"]*
```

#### Note

There are several ways to transform data to match the type of output you need. In the above example, regular expression is used to get the needed output (the certificate expiration date). This regular expression was tested against the output of the z/OS Certs – List Cert Ansible job to extract the value assigned to the 'End Date' field in the job's output. After completing this use case, you can experiment with other regular expressions to extract additional information from the job's output.

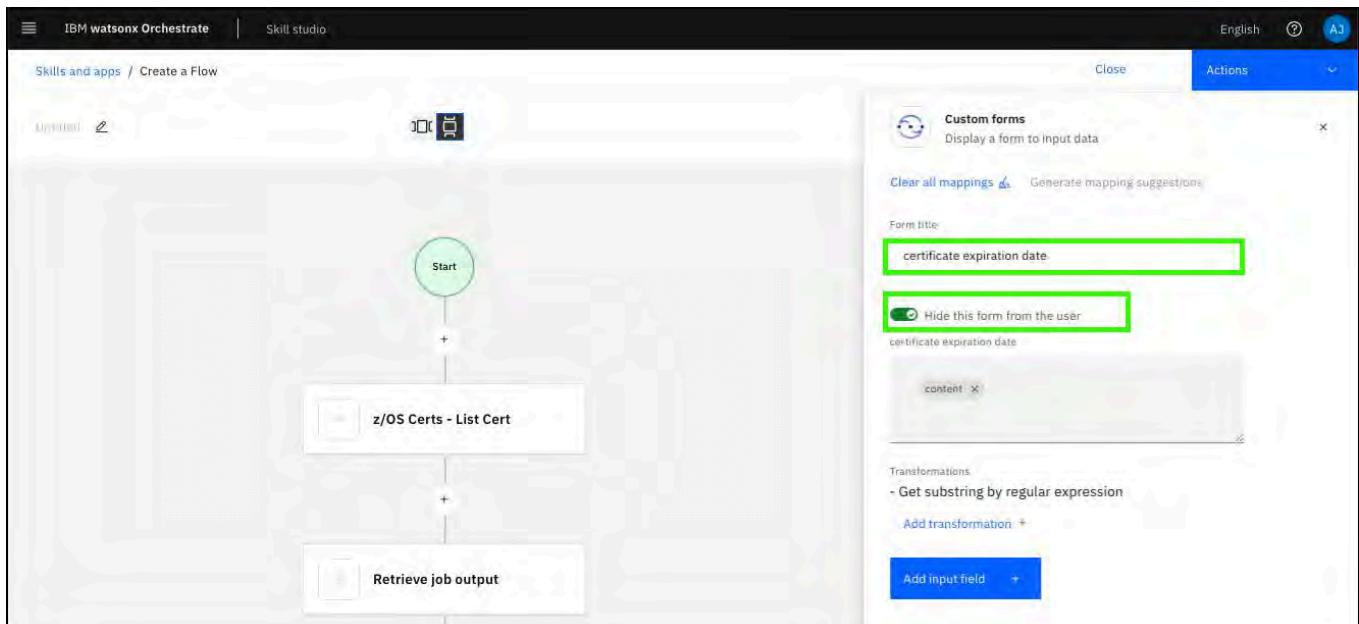
For more information on transforming data within Watsonx Orchestrate, review the documentation found [here](#).



27. Enter **certificate expiration date** in the form title and toggle the **Hide this form from the user** option.

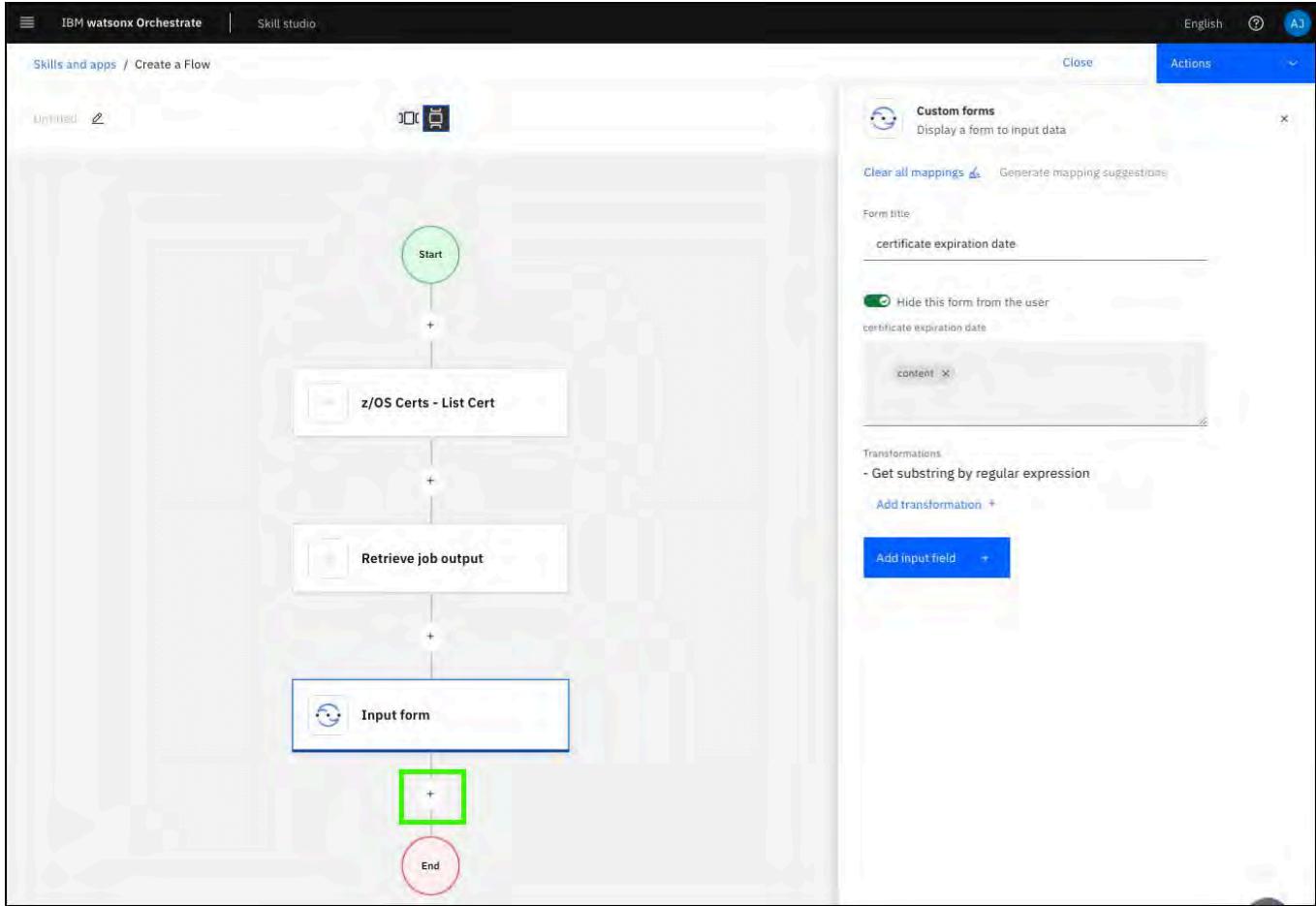
#### Form title:

certificate expiration date

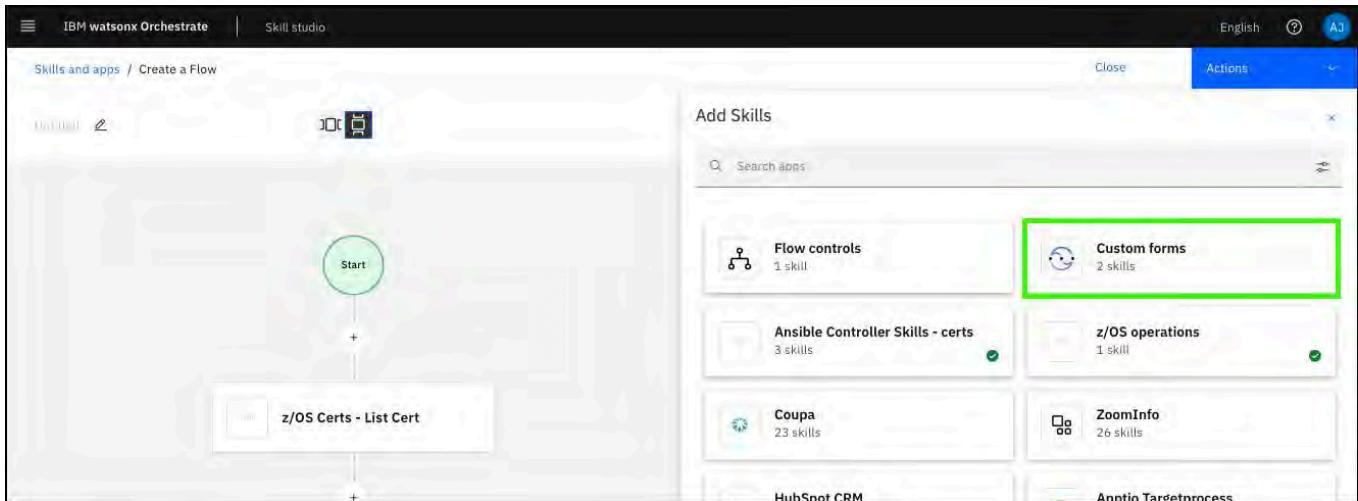


Next, create an output for to return the transformed data from the input skill.

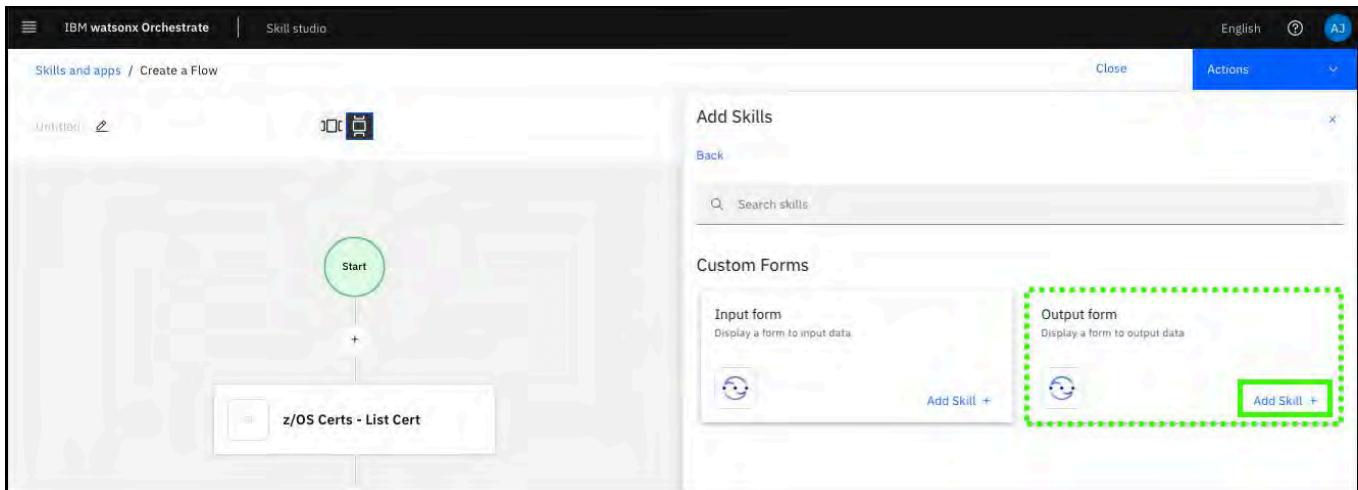
28. Click the **+** icon below the Input form.



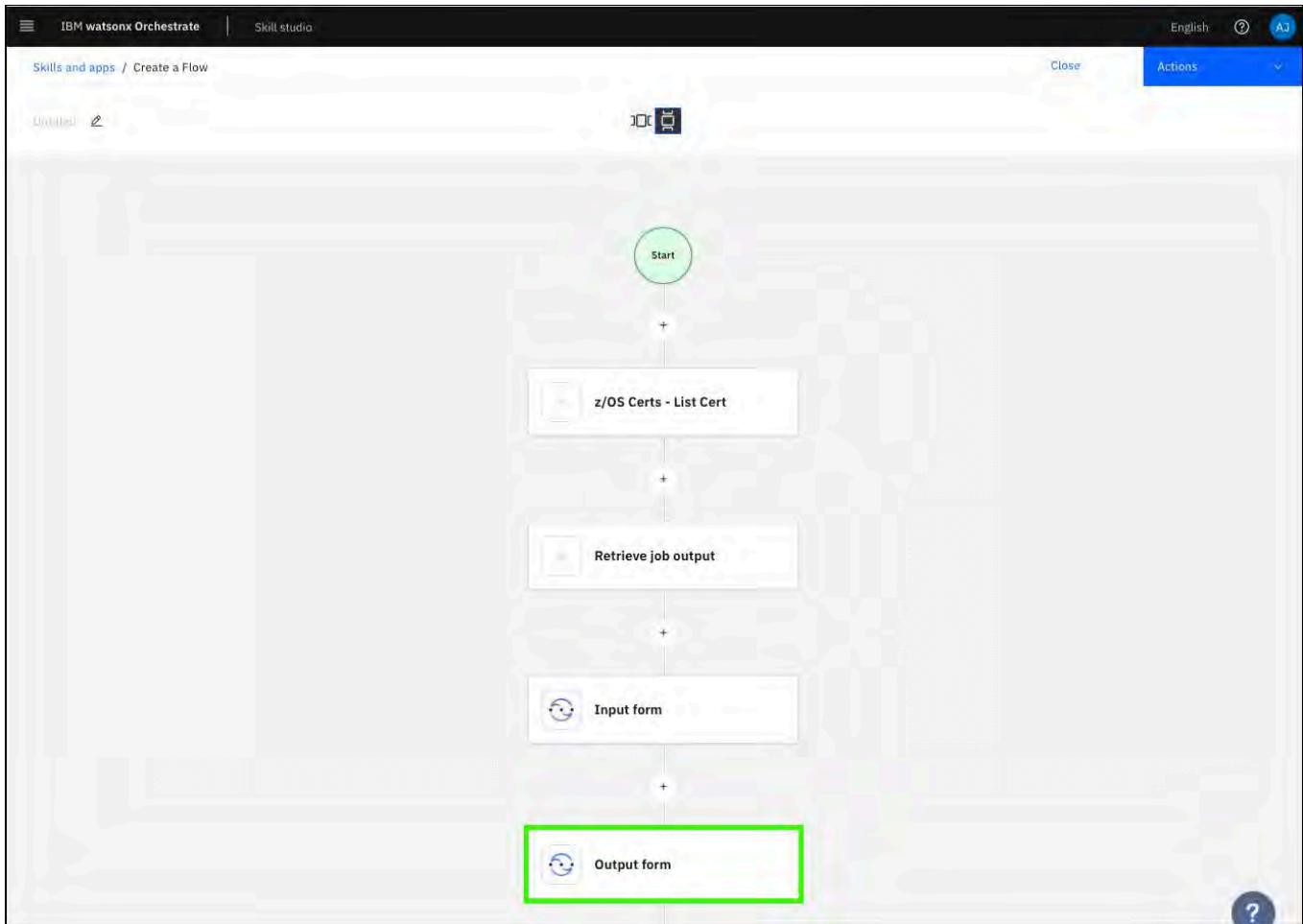
29. Click **Custom forms**.



30. Click **Add Skill +** in the **Output form** tile.

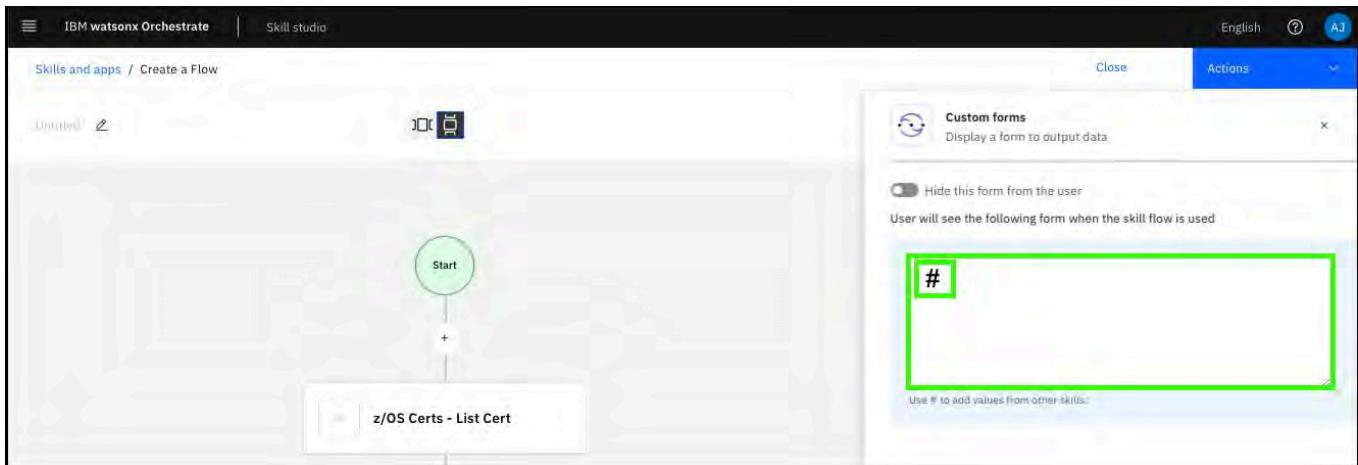


31. Click the **Output form** skill.

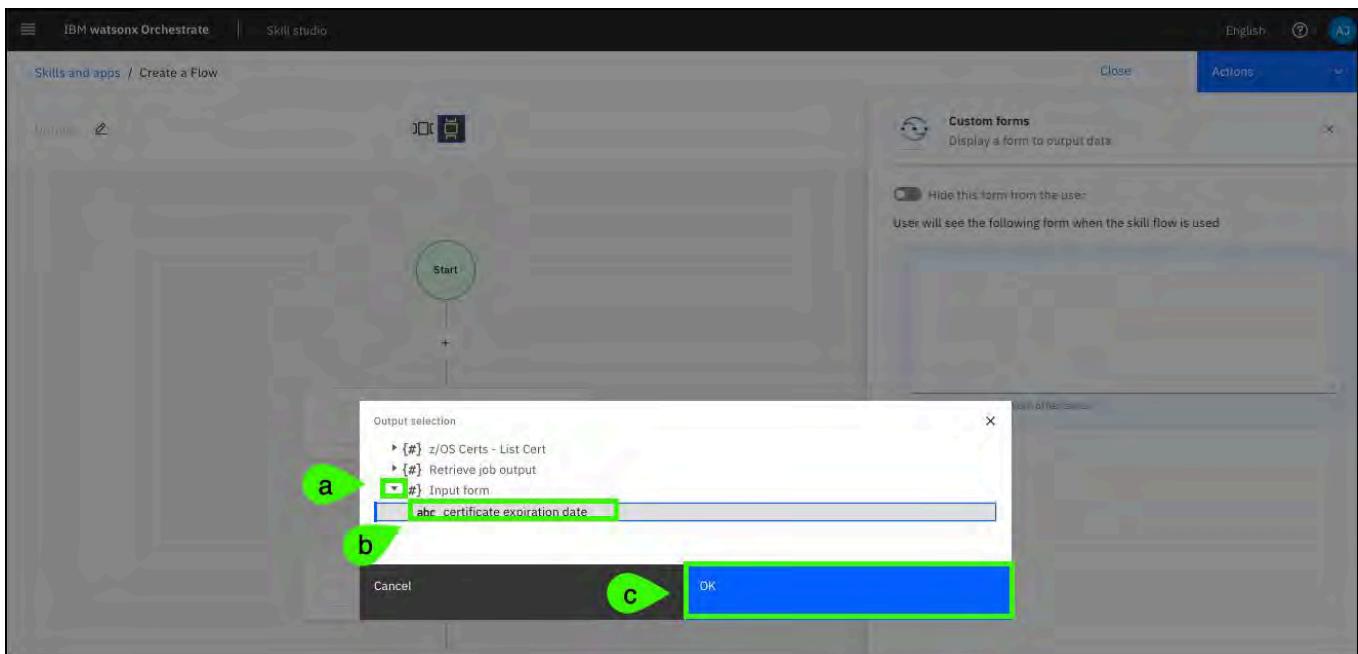


32. Click in the **Custom forms** field and enter `#` (the pound key, also known as the number sign or hash key).

Typing the `#` opens a new dialog window.



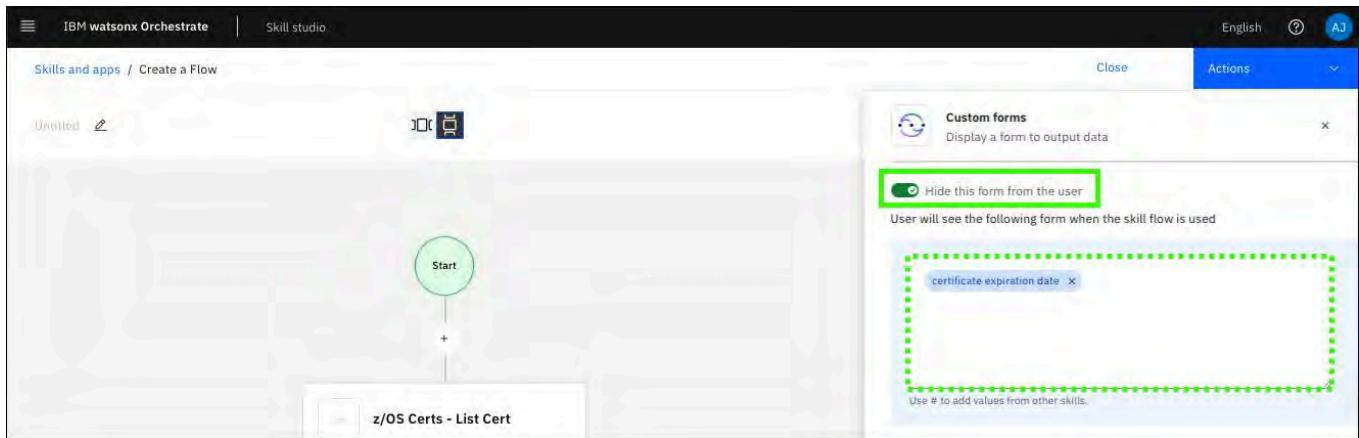
33. Expand (a) **Input form**, select (b) **certificate expiration date**, and then click (c) **OK**.



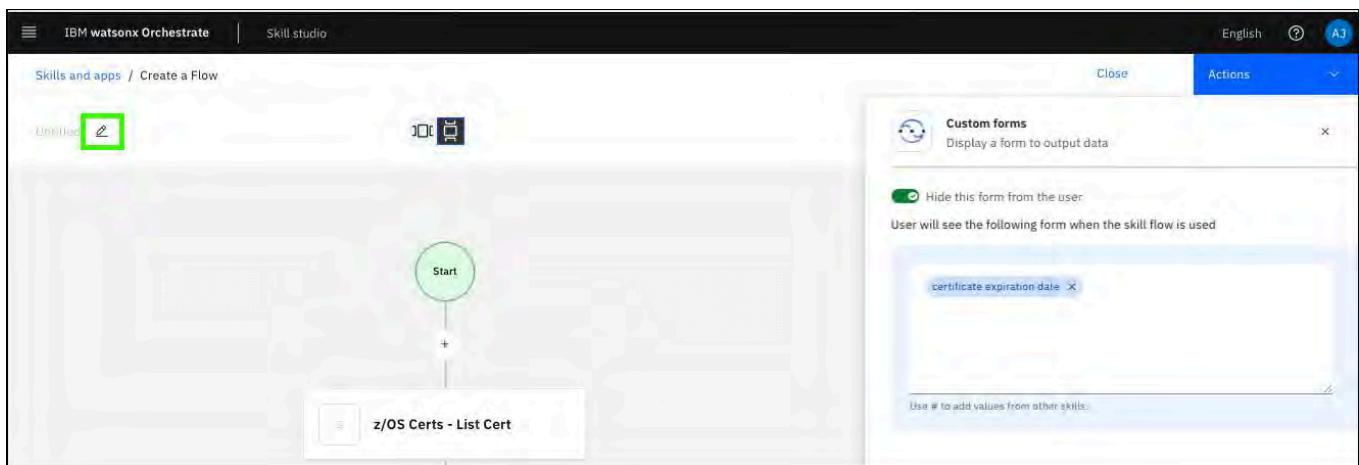
34. Enable the **Hide this form from the user** option.

#### **?** Why hide some of the forms?

You may be wondering why hide the input and output forms for the skills in the skill flow. This is done to execute the automation based on user prompts for the inputs of the skills. This is done through natural conversation with the assistant when the skill flow is configured as an assistant ‘action’ (you will do this soon). Although the final output is hidden, it is accessible as a variable in a custom-built action. The value can be displayed exactly at the point it is expected in the conversation.



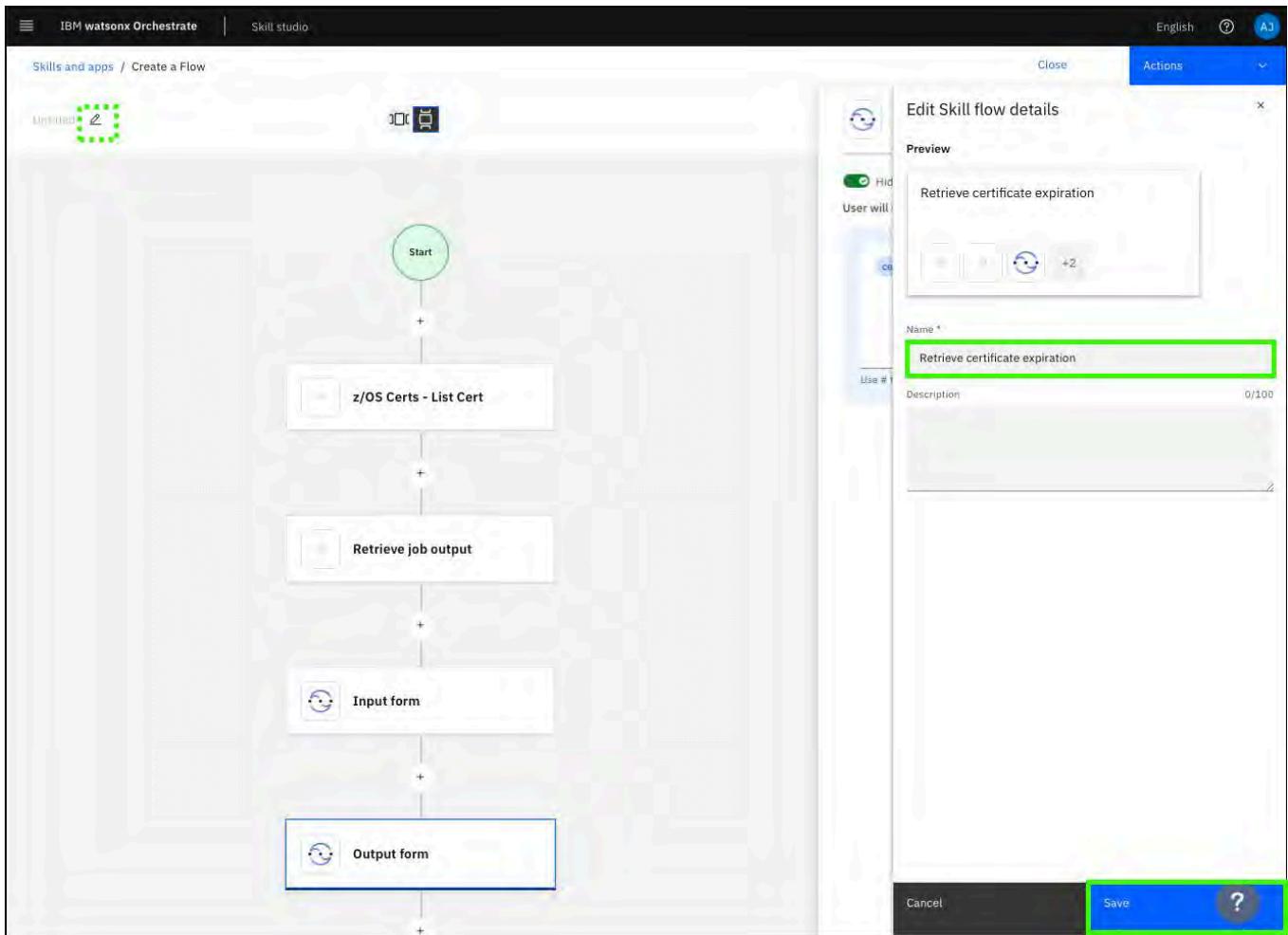
35. Click the pencil icon (✍).



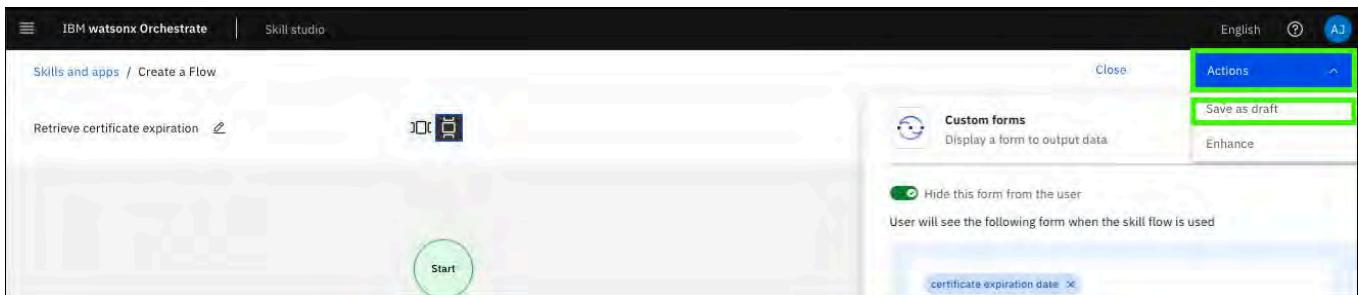
36. Enter Retrieve certificate expiration in the Name field and click Save.

**Name:**

Retrieve certificate expiration



37. Click **Actions** and then click **Save as draft**.



38. Click **Actions** and then click **Enhance**.



39. Review the skill flow settings and click **Publish**.

You created a new skill flow that accomplishes part of the use case – retrieving and displaying the expiration date of a z/OS certificate based on the certificate label the user provides.

In the next section, you will create a simpler skill flow for the z/OS Certs – Search and Renew skill that you previously imported. After this additional skill flow is created, add both skill flows as skill-based actions to be called in a custom-built action that maps inputs to the skill flows through natural conversation.

## Create a skill flow for certificate renewal

The final step before configuring the assistant with actions is to create a skill flow for renewing certificates. Recall the z/OS Certs – Search and Renew automation imported from Ansible Automation Platform earlier. The skill flow that you create next is composed of that single skill. There is no need to return the output. After the automation is triggered, the user can verify the new expiration date by running the retrieve certificate expiration date flow.

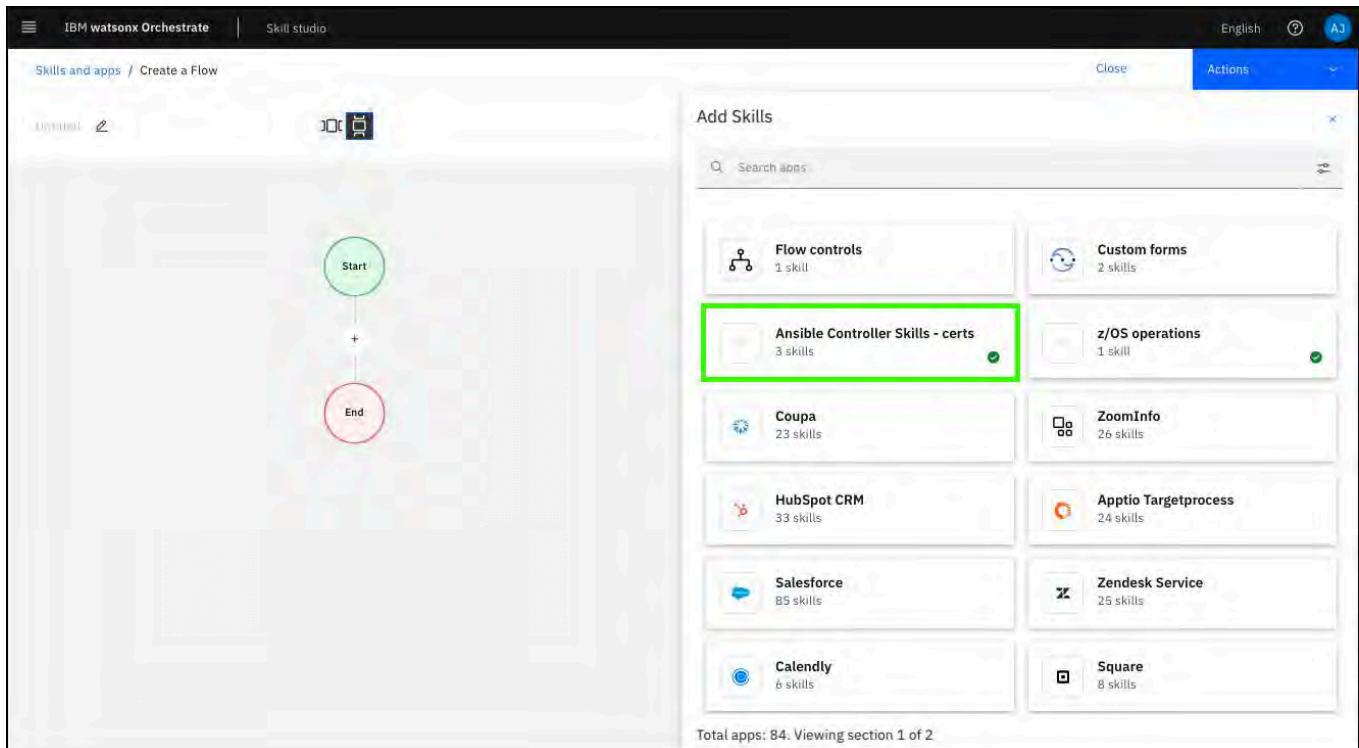
1. In **Skill studio**, click **Create** and then click **Skill flow**.

2. Click the + icon.

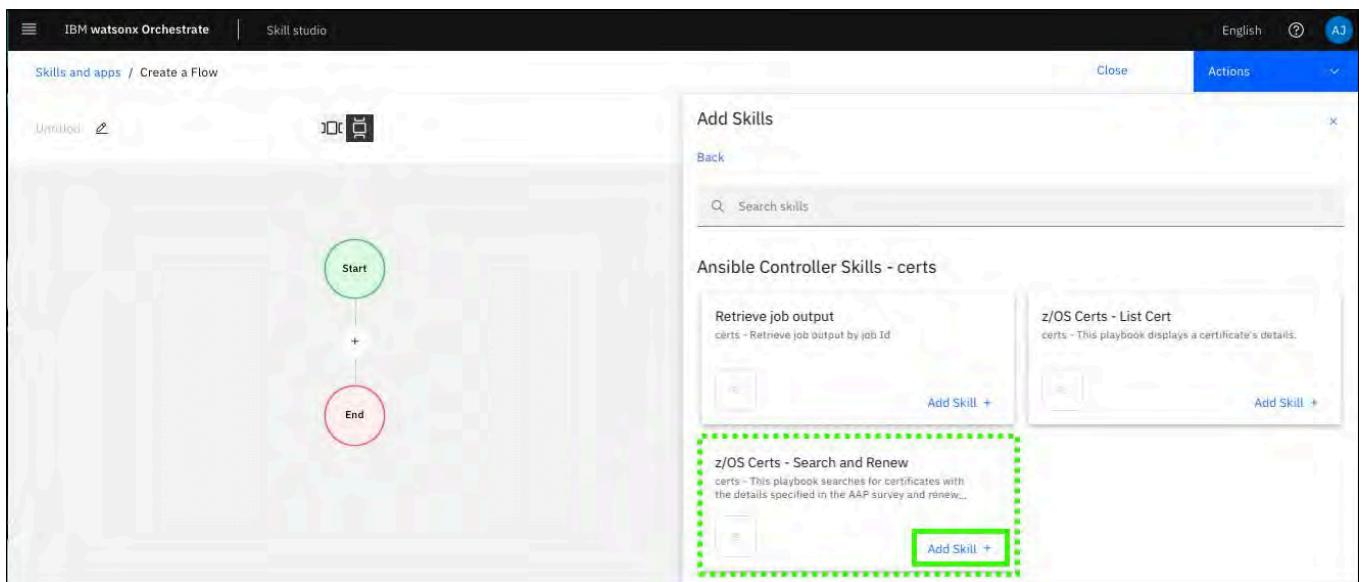
3. Click the certs app.



Search on certs if you do not see the tile for your app.



4. Click Add Skill + in the z/OS Certs - Search and Renew tile.



As mentioned, there is no need to return the Ansible job output of this skill when it is run. The **z/OS Certs - Search and Renew** is used to set default values for some of the inputs. In this use case, assume that the SA will be renewing their SITE certificates that are signed with a previously generated certificate authority.

5. Click the **z/OS Certs - Search and Renew** skill.



## 6. Click Input.

**Ansible Controller Skills - certs**  
certs - This playbook searches for certificates with the details specified in the AAP survey and renews them.

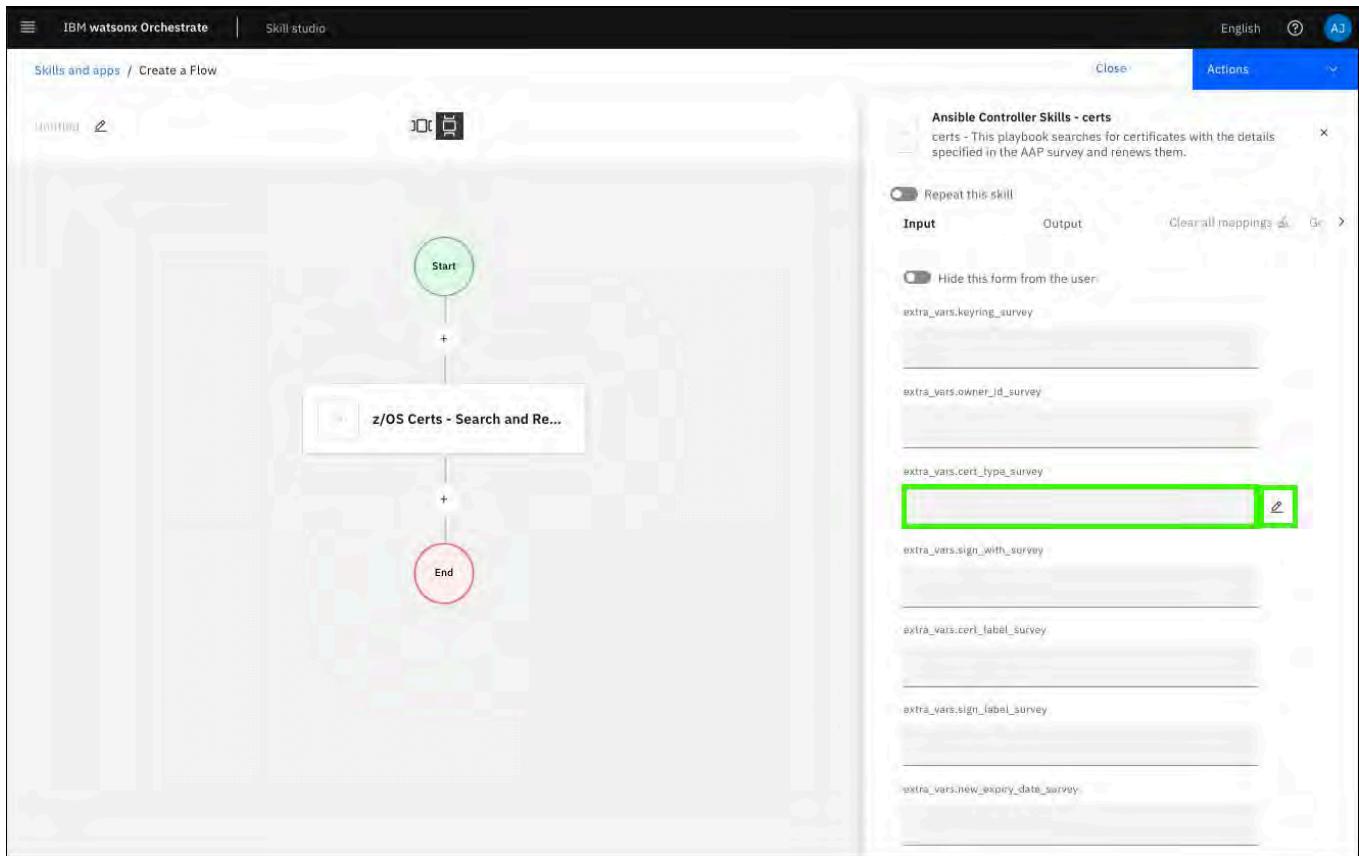
Repeat this skill

**Input**

Hide this form from the user

extra\_vars.keyring\_survey  
extra\_vars.owner\_id\_survey  
extra\_vars.cert\_type\_survey  
extra\_vars.sign\_with\_survey  
extra\_vars.cert\_label\_survey  
extra\_vars.sign\_label\_survey  
extra\_vars.new\_expiry\_date\_survey

## 7. Hover over the **extra\_vars.cert\_type\_survey** input field and click the pencil icon (✉).



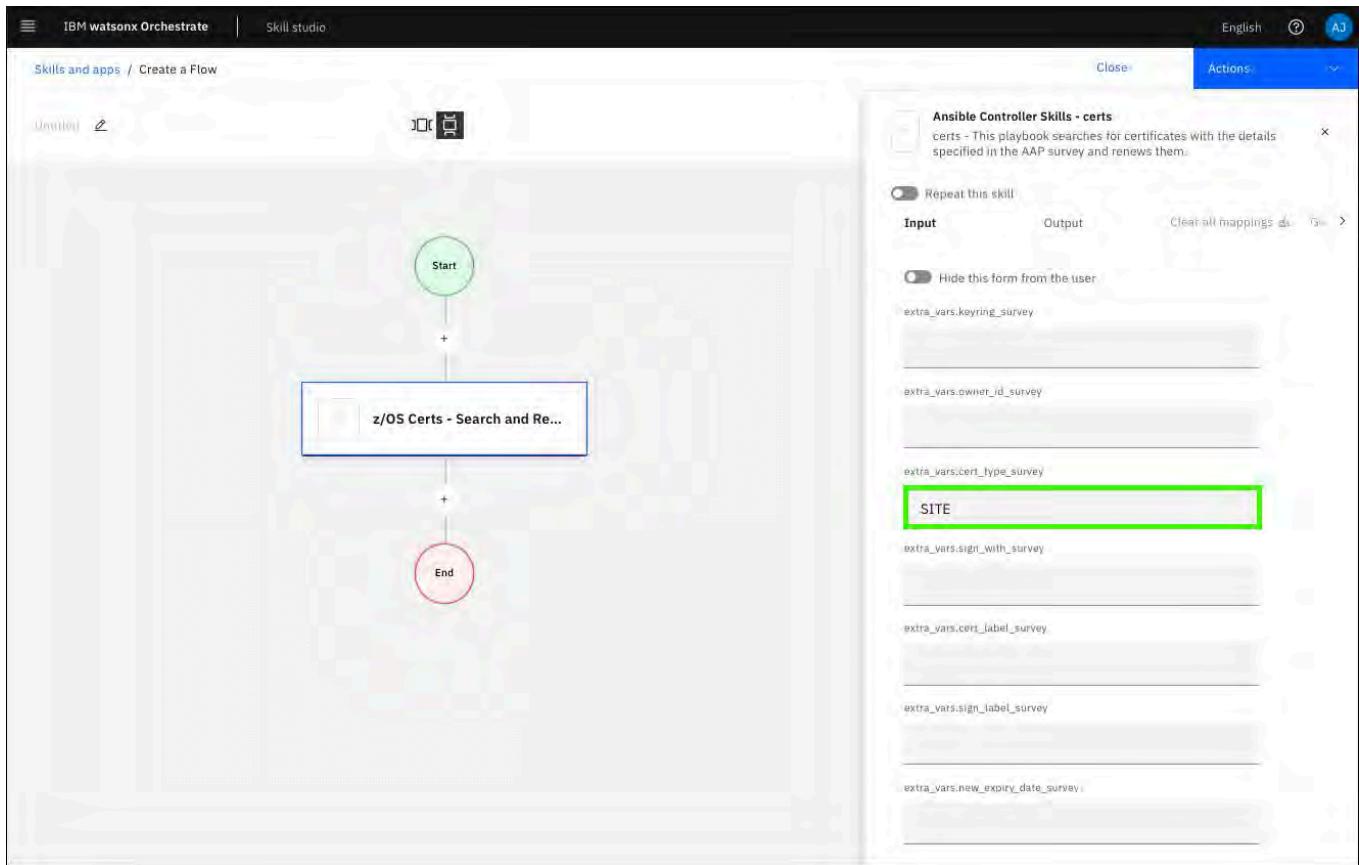
- Click in the **extra\_vars.cert\_type\_survey** input field and enter **SITE**.

**extra\_vars.cert\_type\_survey:**

SITE



**Do not enter spaces before or after the word SITE .**



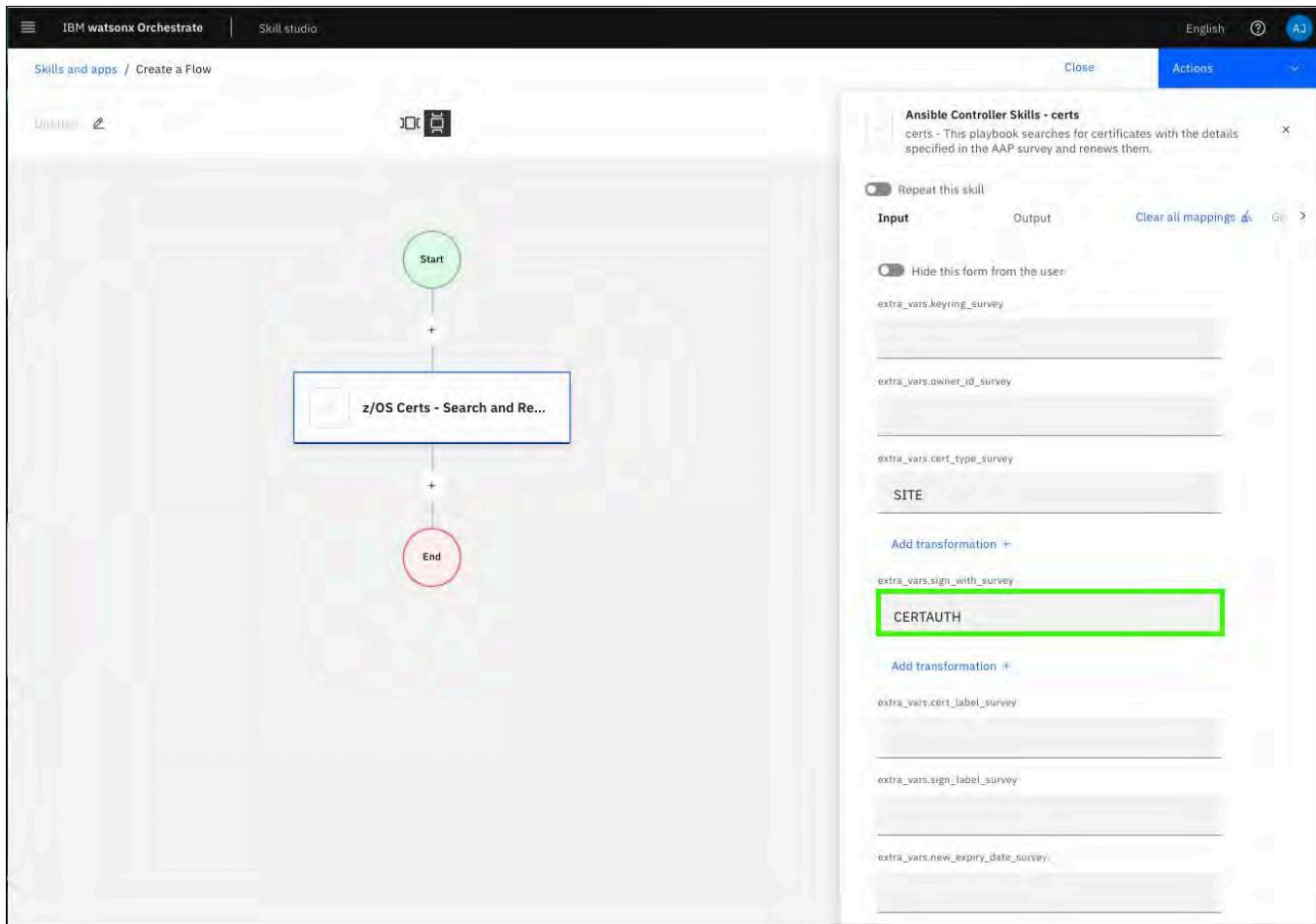
9. Repeat 7 and 8 for the **extra\_vars.sign\_with\_survey** field and enter the word **CERTAUTH**.

**extra\_vars.sign\_with\_survey:**

CERTAUTH



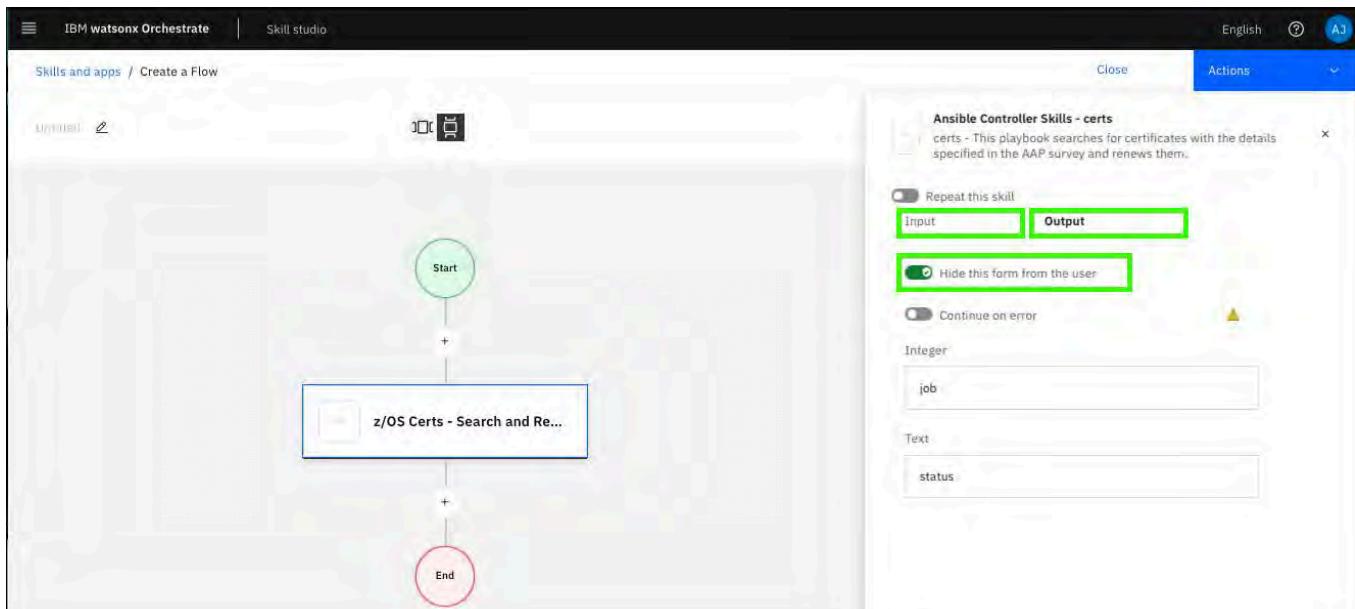
**Do not enter spaces before or after the word CERTAUTH.**



10. Enable the **Hide this form from the user** option for both the **Input** and **Output** forms.



**The image that follows only shows the Output form, but enable the option for both forms.**



11. Click **(a)** the pencil icon ( for the skill flow, enter **(b)** Cert Renewal skill flow in the **Name** field, and click **(c) Save**.

Name:

Cert Renewal skill flow

IBM Watsonx Orchestrate | Skill studio | English | ? | AJ

Skills and apps / Create a Flow

Actions

Edit Skill flow details

Preview

Cert Renewal skill flow

Name \* Cert Renewal skill flow

Description 0/100

Start

+

z/OS Certs - Search and Re...

+

End

Cancel

**C**

Save

12. Click **Actions** and then click **Save as draft**.

Cert Renewal skill flow

Actions

Save as draft

Enhance

Start

+

z/OS Certs - Search and Re...

+

End

13. Click **Actions** and then click **Enhance**.



**14. Review the skill flow settings and click **Publish**.**

Skills and apps / Enhance this skill

### Enhance the “Cert Renewal skill flow” skill

Add details that will make people want to use this skill.

Name	Phrases	Next best skills
Cert Renewal skill flow		

Name and describe this skill in a way that tells users how it's used and why they would want to use it.

Name<sup>\*</sup>  
Cert Renewal skill flow

Description  
Enter the description

API version<sup>\*</sup>  
1.0.0

Categories  
Add categories:

App  
Skill flows

Preview  
The skill will look like this in the catalog.

Cert Renewal skill flow

The skill will look like this in the skill set.

Cert Renewal skill flow

**Publish**

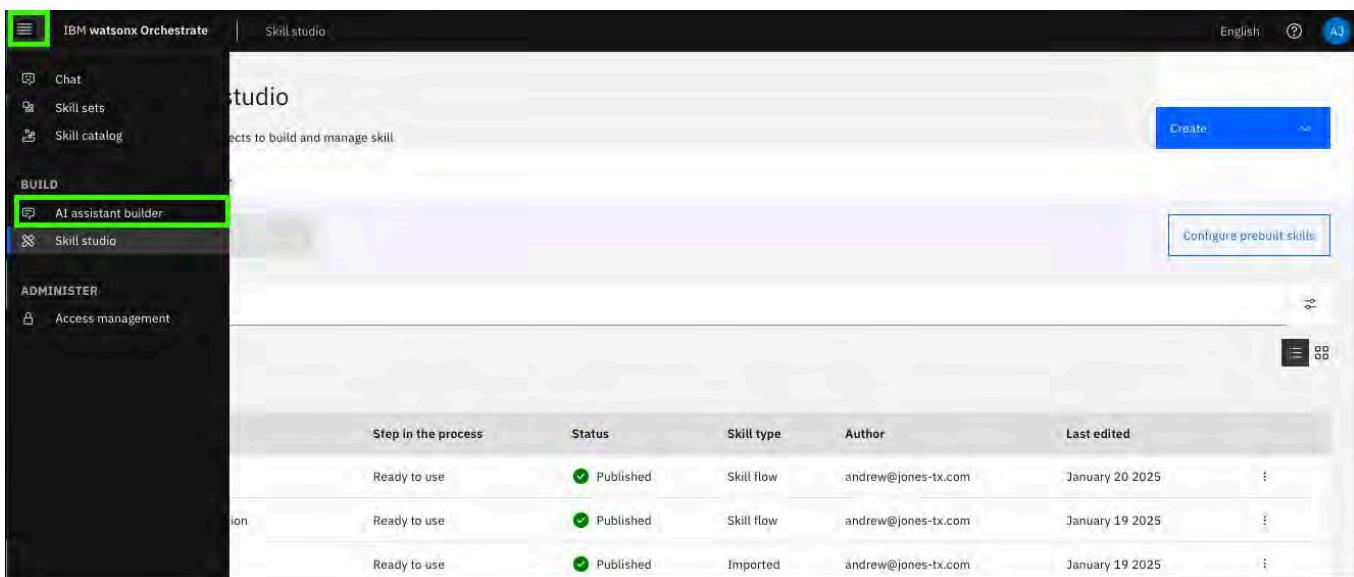
Add the skill flows to the assistant

Next, create 2 skill-based actions that use the skill flows. The skill-based actions enable the ability to call the skill flow as a subaction within a new custom-built action. For this use case, create two skill-based actions that use the previously created skill flows:

- Retrieve certificate expiration – maps the user prompted certificate label as input and extracts the certificate expiration date from the Ansible job's output.
- Cert Renewal skill flow – maps the user prompted certificate label and new expiration date as input and runs the Search and Renew Ansible job to extend the expiration date of the certificate.

After the 2 skill flows are added as skill-based actions, integrate the actions into a custom-built action that defines the entire conversation flow. The flow assists the SA with the certificate renewal process.

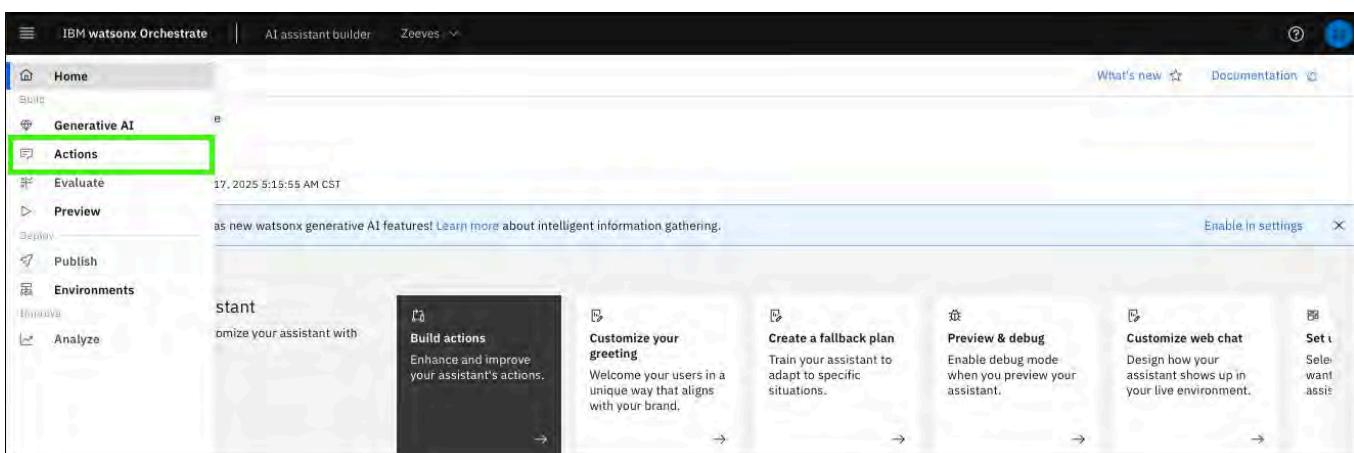
### 1. Open AI assistant builder in watsonx Orchestrate.



The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there is a dark sidebar with several sections: Chat, Skill sets, Skill catalog, BUILD (with AI assistant builder highlighted and outlined in green), Skill studio, ADMINISTER, and Access management. The main area is titled "Skill studio" and contains the sub-instruction "Selects to build and manage skill". A blue "Create" button is at the top right. Below it is a "Configure prebuilt skills" button. The bottom half of the screen displays a table of skill flows:

Step in the process	Status	Skill type	Author	Last edited
Ready to use	<span style="color: green;">✓ Published</span>	Skill flow	andrew@jones-tx.com	January 20 2025
Ready to use	<span style="color: green;">✓ Published</span>	Skill flow	andrew@jones-tx.com	January 19 2025
Ready to use	<span style="color: green;">✓ Published</span>	Imported	andrew@jones-tx.com	January 19 2025

### 2. Click Actions.



The screenshot shows the AI assistant builder interface. On the left, there is a sidebar with sections: Home, Build (with Generative AI selected and outlined in green), Actions (selected and outlined in green), Evaluate, Preview, Publish, Environments, Analyze, and Help. The main area has a message: "as new watsonx generative AI features! Learn more about intelligent information gathering." Below this are several cards:

- Build actions**: Enhance and improve your assistant's actions.
- Customize your greeting**: Welcome your users in a unique way that aligns with your brand.
- Create a fallback plan**: Train your assistant to adapt to specific situations.
- Preview & debug**: Enable debug mode when you preview your assistant.
- Customize web chat**: Design how your assistant shows up in your live environment.
- Set**: Select what want assist

### 3. Click New action+.

The screenshot shows the 'Actions' section of the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with filters for 'All items', 'Created by you', 'Variables', and 'Saved responses'. The main area lists three actions:

Name	Last edited	Examples count	Steps count	Status
z/OS IPL Information	a day ago	0	3	Green
Zeeves-gather-facts-flow	3 days ago	1	0	Green
Retrieve IPL information	21 hours ago	1	3	Green

#### 4. Click Skill-based action.

The screenshot shows the 'Create an action' dialog. It asks 'What kind of action do you want to build?' and provides three options:

- AI-guided action**: Configure AI to generate responses based on specific use cases. (Beta)
- Skill-based action**: Build an action that helps perform a task. This option is highlighted with a green box.
- Custom-built action**: Design your own action step-by-step. (Not sure how to start? Try adding pre-built action templates.)

#### 5. Click the Retrieve certificate expiration tile and then click Next.

The screenshot shows the 'Build an action from a skill' dialog. It asks 'Select a skill' and lists several options:

Cert Renewal skill flow	<b>Retrieve certificate expiration</b>	Retrieve job output	z/OS Certs - Search and Renew
Last updated: 2025-01-20T13:33:55.307Z	Last updated: 2025-01-19T22:56:11.560Z	certs - Retrieve job output by job Id	certs - This playbook searches for certificates with the details specified in the AAP survey and renews them.
<a href="#">z/OS Certs - List Cert</a>	<a href="#">z/OS IPL Information</a>	<a href="#">Zeeves-gather-facts-flow</a>	<a href="#">Retrieve job output</a>
certs - This playbook displays a certificate's details	z/OS IPL Information	Run z/OS gather facts and retrieve the output.	z skills - Retrieve job output by job Id

#### 6. Click Cancel on the New action dialog.



For this use case, the action is triggered from a custom-built action. To prevent the skill flow from being run as the skill-based action, do not enter any example phrases.

The screenshot shows the 'AI assistant builder' tab selected in the top navigation bar. Below it, the 'Retrieve certificate expiration' skill is displayed. A modal window titled 'New action' is overlaid on the main interface, prompting the user to enter a phrase that starts an interaction. The input field contains the example phrase 'I want to pay my credit card bill.'

7. Click **x** to close the **Retrieve certificate expiration** skill.

The screenshot shows the same skill configuration as the previous one, but the 'x' button in the top right corner of the skill's configuration panel is highlighted with a green box, indicating where to click to close the skill.

8. Repeat steps 3 - 7 to create a skill-based action for the **Cert Renewal** skill flow.



**This action is also triggered from a custom-built action. Do not enter any example phrases.**

9. Verify that both skill-based actions are available.

The screenshot shows the 'Actions' list in the 'AI assistant builder' tab. It lists several actions, including 'z/OS IPL Information', 'Zeeves-gather-facts-flow', 'Retrieve IPL information', 'Retrieve certificate expiration', and 'Cert Renewal skill flow'. The last two items are highlighted with green boxes, indicating they are the ones created in the previous steps.

Name	Last edited	Examples count	Steps count	Status
z/OS IPL Information	a day ago	0	3	Green circle
Zeeves-gather-facts-flow	3 days ago	1	0	Green circle
Retrieve IPL information	a day ago	1	3	Green circle
Retrieve certificate expiration	13 minutes ago	0	3	Green circle
Cert Renewal skill flow	a few seconds ago	0	7	Green circle

## Create a custom-built action for SSL Certificate Renewal

Next, create a custom-built action that runs the new skill-based actions as subactions. Configure the custom-built action to enable a natural conversation with the assistant, gather relevant details from the user, and map those details to the action inputs.

### 1. Click **New action +**.

The screenshot shows the 'Actions' section in the IBM Watsonx Orchestrate interface. The 'Created by you' tab is selected. A green dashed box highlights the 'New action +' button in the top right corner of the table header. The table lists several actions, including 'Retrieve certificate expiration' and 'Cert Renewal skill flow', which are also highlighted with a green dashed box.

Name	Last edited	Examples count	Steps count	Status
z/OS IPL Information	a day ago	0	3	Green
Zeeves-gather-facts-flow	3 days ago	1	0	Green
Retrieve IPL Information	a day ago	1	3	Green
<b>Retrieve certificate expiration</b>	33 minutes ago	0	3	Green
Cert Renewal skill flow	a few seconds ago	0	7	Green

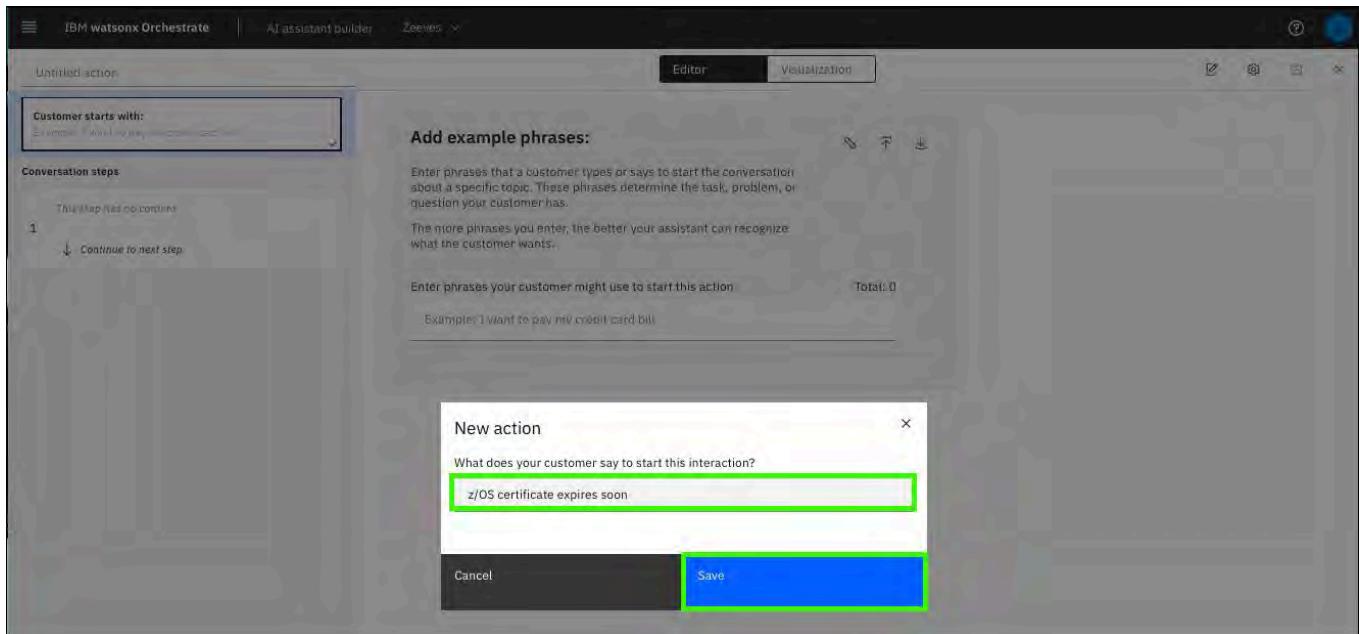
### 2. Click **Custom-built-action**.

The screenshot shows the 'Create an action' dialog. It asks 'What kind of action do you want to build?'. Three options are shown: 'AI-guided action' (blue box), 'Skill-based action' (purple box), and 'Custom-built action' (green box). The 'Custom-built action' box is highlighted with a green dashed box. Below it, a note says 'Not sure how to start? Try adding pre-built action templates.'

### 3. Enter `z/OS certificate expires soon` and then click **Save**.

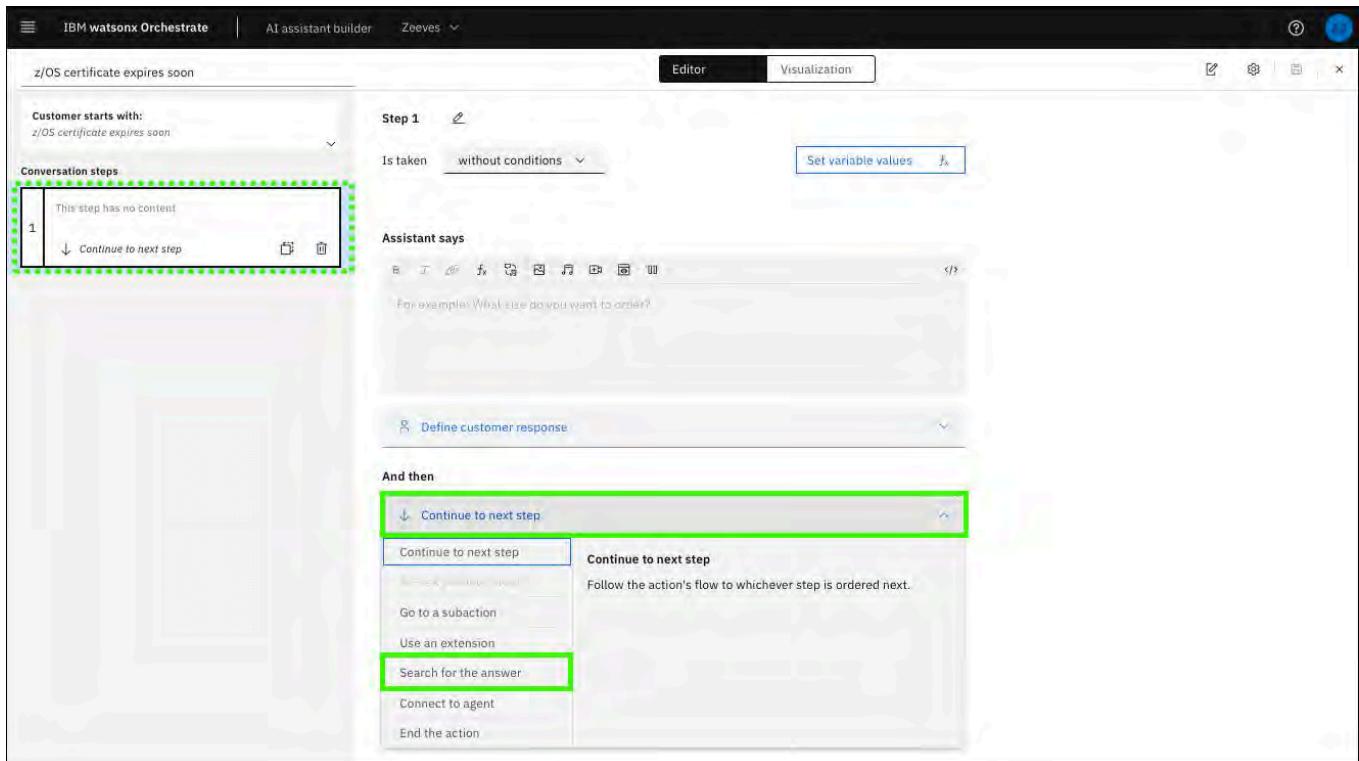
**What does your customer say to start this interaction:**

```
z/OS certificate expires soon
```



The conversational search capability that is provided by Watsonx Assistant for Z can provide step-by-step guidance for determining certificate expiration and renewing certificates, and is grounded on Z domain-specific knowledge. In the first step to be taken when the user prompts the assistant with `z/OS certificate expires soon`, configure the assistant to use conversational search to provide a response on the process and the ability to automate the process.

#### 4. Click the **And then** drop down and select **Search for the answer**.



The result is that anytime the user input matches the example phrase `z/OS certificate expires soon`, the first step that is taken is for the assistant to use conversational search and provide a response to their original question.

Like in the IPL Information scenario, add a custom search query so when conversational search is run in the first conversation step, the query used is hardcoded and not what the user input.

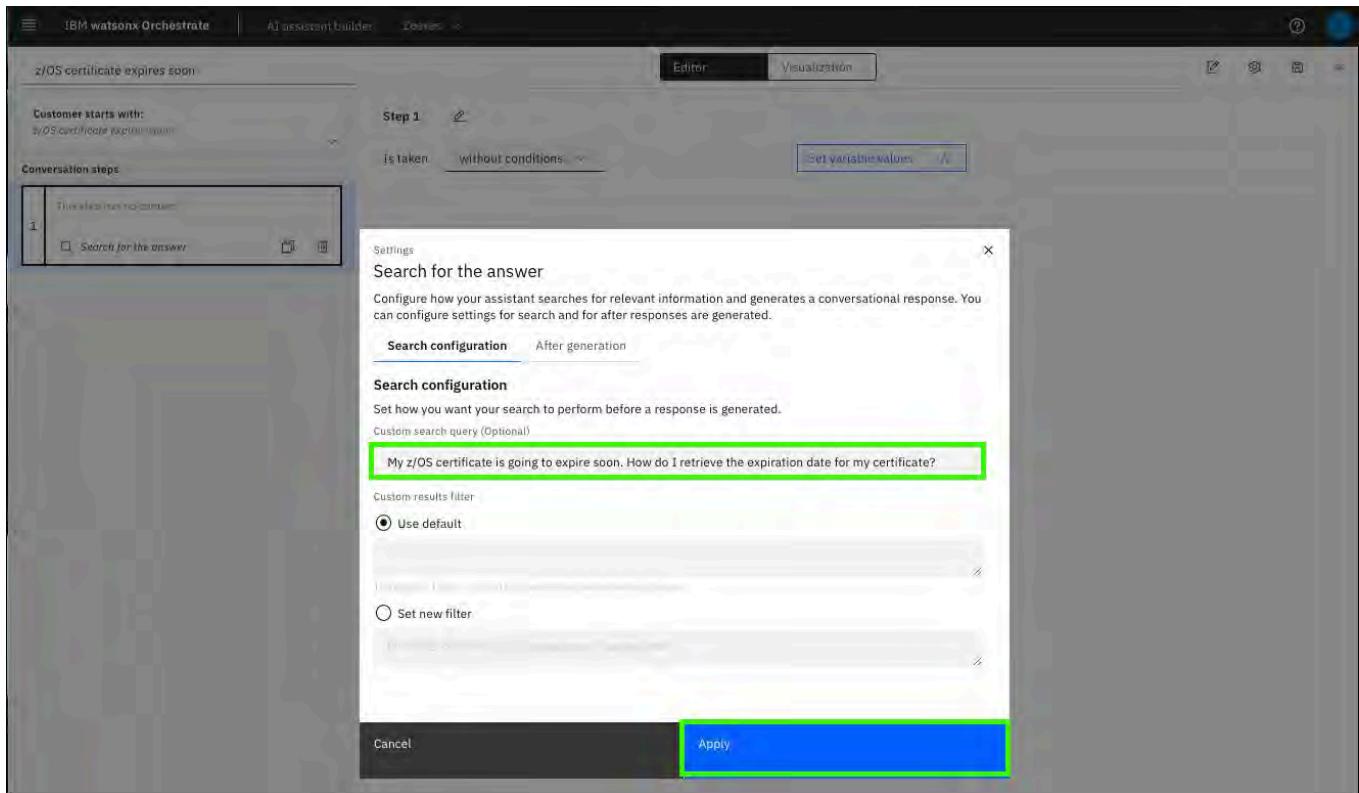
##### 5. Click **Edit settings**.

The screenshot shows the IBM Watsonx Orchestrate interface. In the top navigation bar, 'IBM Watsonx Orchestrate' and 'AI assistant builder' are visible. The main area displays a conversation step titled 'Step 1'. The 'Assistant says' section contains a search bar with the placeholder 'Search for the answer'. Below this, under 'And then', there is another search bar with the same placeholder. At the bottom of the 'And then' section, there is a button labeled 'Edit settings' which is highlighted with a green box.

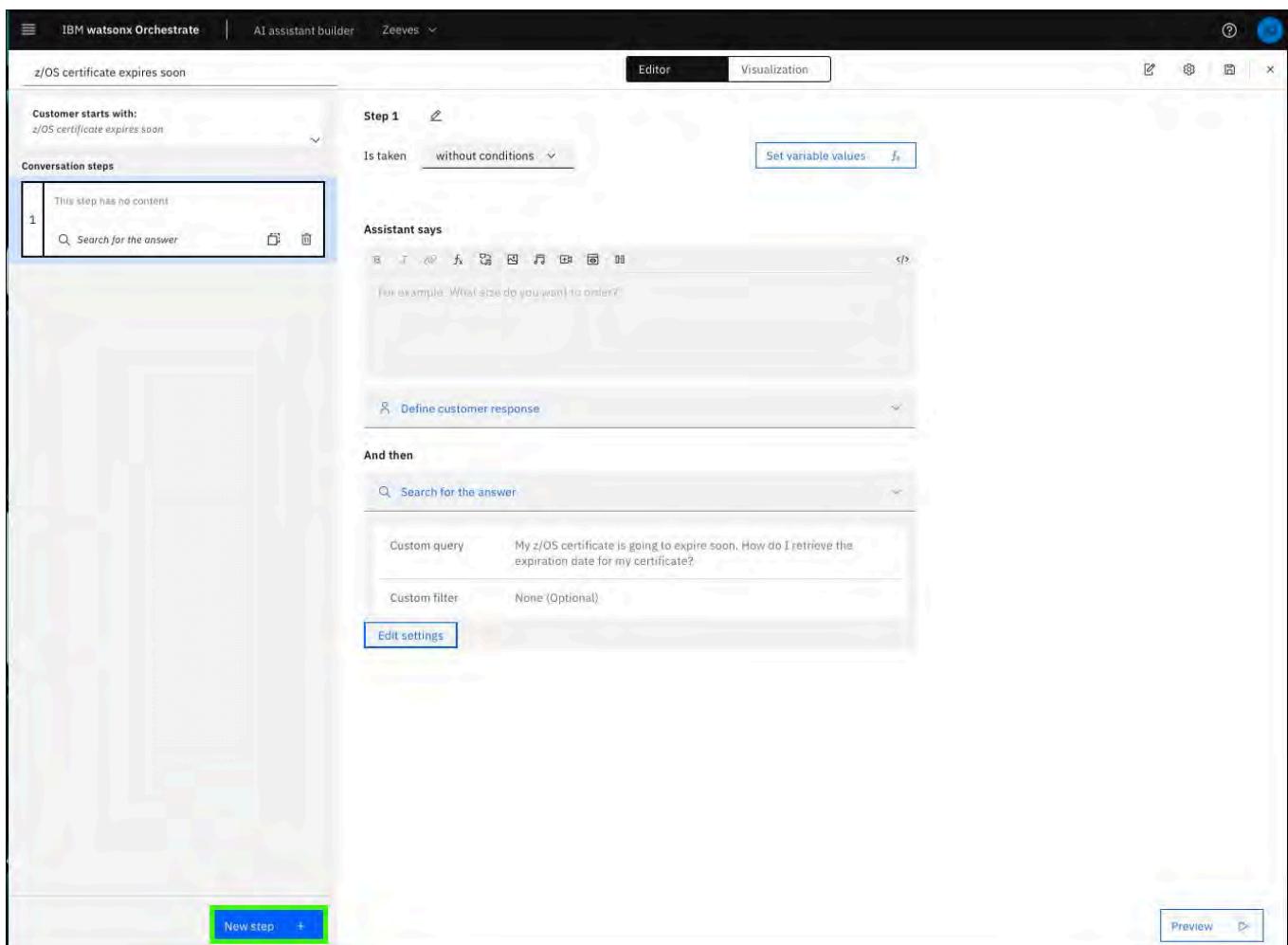
##### 6. Enter the following prompt to be used in the **Custom search query** field and then click **Apply**.

###### Custom search query:

```
My z/OS certificate is going to expire soon. How do I retrieve the expiration date for my certificate?
```



## 7. Click Next step+.



8. Enter the following response in the **Assistant says** field.

**Assistant says:**

Would you like to run the skill to retrieve your certificate's expiration date?

The screenshot shows the AI assistant builder interface with the following details:

- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  - Step 1: Q: Search for the answer
  - Step 2: This step has no content
  - Step 3: Q: Would you like to run the skill to retrieve your certificate's expiration date?  
A: Continue to next step
  - Step 4: Q: Define customer response
  - Step 5: A: Continue to next step
- Step 2 Assistant says:** Would you like to run the skill to retrieve your certificate's expiration date?
- Buttons:** Editor, Visualization, Set variable values, and a gear icon.

9. Click the **Define customer response** option list and select **Confirmation**.

The **Confirmation** option prompts the user to select **Yes** or **No**.

The screenshot shows the AI assistant builder interface with the following details:

- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  - Step 1: Q: Search for the answer
  - Step 2: This step has no content
  - Step 3: Q: Would you like to run the skill to retrieve your certificate's expiration date?  
A: Continue to next step
  - Step 4: Q: Define customer response
  - Step 5: A: Continue to next step
- Define customer response dropdown:**
  - System** tab selected
  - Confirmation** option selected (highlighted with a green box)
  - Description: Enable customers to select from Yes or No response.
  - Buttons: As buttons, Yes, No
  - Other options: Options, Free text, Regex, Number, Date, Time, Saved

10. Click **Next step +**.

The screenshot shows the IBM Watsonx Orchestrate interface. In the top navigation bar, 'IBM watsonx Orchestrate' is selected. Below it, 'AI assistant builder' and 'Zeeves' are shown. The main area is titled 'z/OS certificate expires soon'. A 'Step 2' section is highlighted with a green dashed box. It contains a 'Customer starts with:' field with the value 'z/OS certificate expires soon'. An 'Is taken' dropdown menu is open, showing 'without conditions' (selected) and 'with conditions'. To the right, there's a 'Set variable values' button. Below this, the 'Assistant says' section shows the message 'Would you like to run the skill to retrieve your certificate's expiration date?'. There are 'Yes' and 'No' buttons below the message. A 'View response' and 'Edit validation' button are also present. The 'And then' section shows a 'Continue to next step' option. At the bottom left, a 'New step +' button is highlighted with a green box. At the bottom right, there are 'Preview' and 'Next' buttons.

#### 11. Click the **Is taken** option list and select **with conditions**.

This step handles the flow when the user selects **Yes** in the previous step, indicating that they want to run the skill to retrieve the certificate's expiration date. To run the **Retrieve certificate expiration action** created earlier, the assistant needs the certificate label. This label is mapped as input to the skill.

The screenshot shows the IBM Watsonx Orchestrate interface. The 'Step 3' section is highlighted with a green dashed box. The 'Is taken' dropdown menu is open, showing 'without conditions' (selected), 'without conditions' (disabled), and 'with conditions' (highlighted with a green box). To the right, there's a 'Set variable values' button. Below this, the 'Assistant says' section shows the message 'For example, Please select from the following options'. The 'And then' section shows a 'Continue to next step' option. At the bottom left, a 'New step +' button is highlighted with a green box. At the bottom right, there are 'Preview' and 'Next' buttons.

#### 12. Enter the following text in the **Assistant says** field.

**Assistant says:**

What is your certificate label?

The screenshot shows the IBM Watsonx Orchestrate interface with the title "z/OS certificate expires soon". The "Editor" tab is selected. The conversation steps are as follows:

- Customer starts with: z/OS certificate expires soon
- This step has no content.
- Search for the answer: Would you like to run the skill to retrieve your certificate's expiration date?
- Confirmation: Continue to next step
- Condition: 2. Would you like to r... is Yes
- What is your certificate label?
- Continue to next step

The "Assistant says" section contains the question "What is your certificate label?".

Annotations highlight the condition "2. Would you like to r... is Yes" and the "Define customer response" dropdown in the "Assistant says" section.

13. Click the **Define customer response** drop-down list and select **Free text**.

The screenshot shows the IBM Watsonx Orchestrate interface. At the top, there are tabs for "Editor" and "Visualization". The main area displays a conversation flow:

- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  1. Q Search for the answer: Would you like to run the skill to retrieve your certificate's expiration date?
  2. Confirmation: Continue to next step
  3. Confirmation: What is your certificate label? (This step is highlighted with a green dashed box)
- Step 3:** Is taken with conditions. A condition is defined: If 2. Would you like to r... is Yes.
- Assistant says:** What is your certificate label?

A modal dialog is open, titled "Define customer response". It has a "System" dropdown menu with the following options:

- Options (selected)
- Confirmation
- Free text (highlighted with a green box)
- Regex
- Number
- Date
- Time
- Saved

The "Free text" section contains the sub-section "As text" with the value "My street is Main Street".

14. Click **Next step+**.

The screenshot shows the IBM Watsonx Orchestrate AI Assistant builder interface. The main area displays a conversation flow with three steps:

- Step 1:** Customer starts with "z/OS certificate expires soon". Response: "Search for the answer". Condition: "This step has no content".
- Step 2:** Conversation steps: "Would you like to run the skill to retrieve your certificate's expiration date?". Response: "Confirmation". Condition: "2. Would you like to r... is Yes".
- Step 3:** Conversation steps: "What is your certificate label?". Response: "Free text". Condition: "User enters free text". Next step: "Continue to next step".

The "Step 3" section is highlighted with a green dashed box. The condition "2. Would you like to r... is Yes" is also highlighted with a green dashed box. The "User enters free text" response is highlighted with a green dashed box.

15. Click the **Is taken** option list and select **with conditions**.

After the user enters the certificate label as free text, the next step is to run the **Retrieve certificate expiration** skill-based action created earlier. To do so, map the user input to the skill flow and retrieve the expiration date for that certificate.

z/OS certificate expires soon

**Step 4**

Is taken: without conditions

Assistant says:

For example: What type of transfer would you like to make?

And then:

- Continue to next step
- Go to a subaction
- Re-ask previous step(s)
- Switch conversation flow to another action to perform a certain task
- Use an extension
- Search for the answer
- Connect to agent
- End the action

## 16. Click the **And then** option list and click **Go to a subaction**.

Notice that the default condition validates the free text is defined from the previous step.

z/OS certificate expires soon

**Step 4**

Is taken: with conditions

Conditions:

If 3. What is your certifi... is defined

and Add condition +

New condition group +

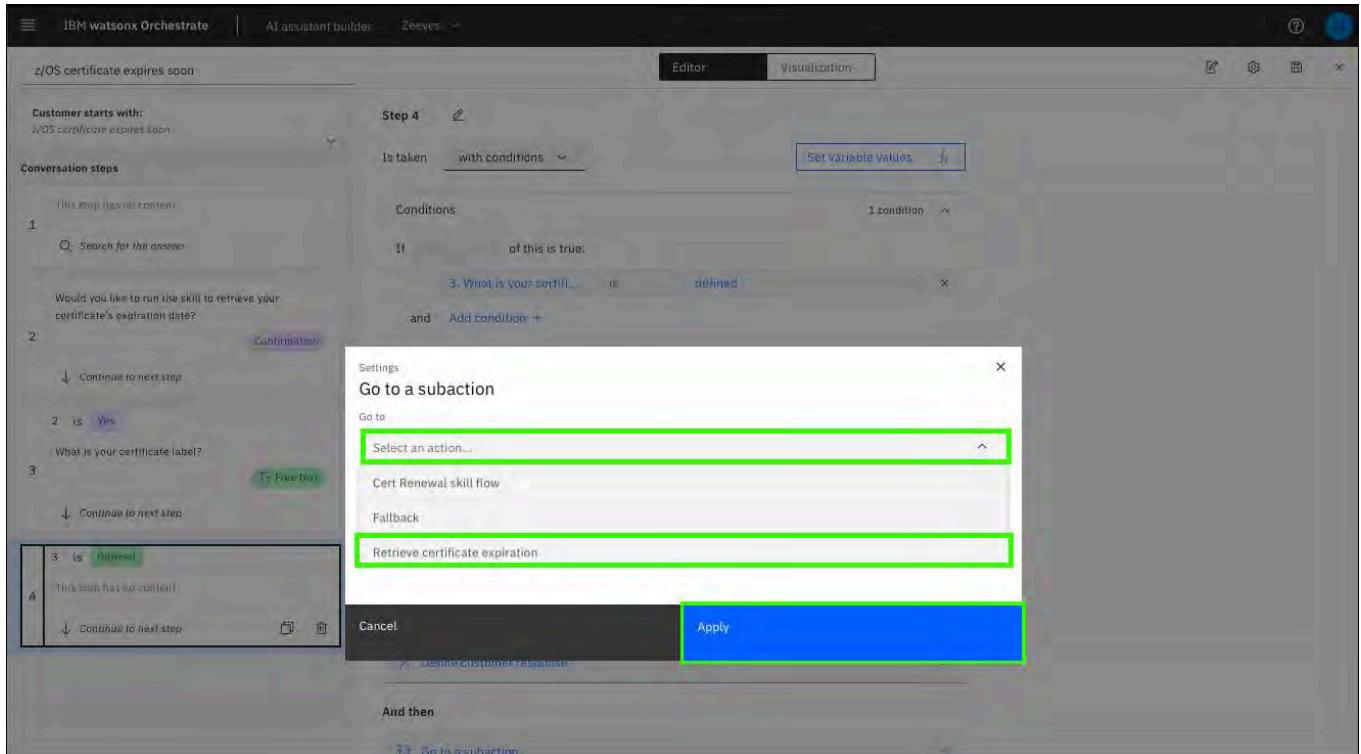
Assistant says:

For example: What type of transfer would you like to make?

And then:

- Continue to next step
- Go to a subaction
- Re-ask previous step(s)
- Switch conversation flow to another action to perform a certain task
- Use an extension
- Search for the answer
- Connect to agent
- End the action

17. Click the (a) Go to option list, select the (b) Retrieve certificate expiration skill-based action, and then click (c) Apply.



18. Click Edit passed values.

To run the **Retrieve certificate expiration** subaction that uses the users certificate label, the passed value needs to be modified.

The screenshot shows the IBM Watsonx Assistant builder interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and tabs for 'Editor' (selected) and 'Visualization'. The main area displays a conversation flow and step configuration.

**Conversation steps:**

1. Q: Search for the answer  
Would you like to run the skill to retrieve your certificate's expiration date?
2. Confirmation  
↓ Continue to next step
3. What is your certificate label?  
3 is Defined  
↓ Continue to next step
4. This step has no content  
Go to subaction: Retrieve certificate expiration

**Step 4 Configuration:**

- Is taken:** with conditions
- Conditions:**
  - If All of this is true:
    - 3. What is your certifi... is defined
    - and Add condition +
  - New condition group +
- Assistant says:** (Text input field: For example: What type of transfer would you like to make?)
- Define customer response:** (Dropdown menu)
- And then:**
  - Go to a subaction (highlighted with a green dashed box)
  - Goes to action: Retrieve certificate expiration (highlighted with a green dashed box)
  - Pass values
  - Upon return: Continue
  - Edit settings
  - Edit passed values (highlighted with a green box)

**Buttons at the bottom:**

- New step +
- Preview
- Done

#### 19. Click Set new value + and then select extra\_vars.cert\_label\_survey.

The screenshot shows the 'Edit passed values' dialog box overlaid on the main builder interface. The dialog has a yellow header bar with the message: 'Variables are not private if they are shared with Skill Studio.'

**Passed values:**

**Edit variable values:**

The list of variables is as follows, with 'extra\_vars.cert\_label\_survey' highlighted with a green box:

- Set new value + (highlighted with a green box)
- All variables
1. extra\_vars.cert\_label\_survey (highlighted with a green box)
2. extra\_vars.cert\_type\_survey
3. extra\_vars.owner\_id\_survey

**Buttons at the bottom:**

- Cancel
- Done
- Go to action
- Retrieve certificate expiration

20. In the To field, select **Action step variables**, and then select **What is your certificate label?**.

The screenshot shows the IBM Watsonx Assistant Editor interface. A modal dialog is open over the main workspace, specifically for Step 4. The dialog title is "Edit variable values". It shows a dropdown menu under "Set" with the value "extra\_vars.cert\_label\_survey". The "To" dropdown menu is open, displaying several options: "Enter text", "Expression", "Action step variables", and "Assistant variables". The "Action step variables" option is highlighted with a yellow circle and a cursor icon. Below the dropdowns are "Set new value" and "+" buttons. At the bottom of the dialog are "Cancel" and "Apply" buttons.

21. Click **Apply**.

This screenshot is similar to the previous one, but the "Action step variables" option has been selected in the "To" dropdown. The dropdown now shows "extra\_vars.cert\_label\_survey" followed by "To 3. What is your certificate label?". The "Apply" button at the bottom of the dialog is highlighted with a blue rectangle.

22. Click **Next step +**.

The screenshot shows the IBM Watsonx Assistant Editor interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and tabs for 'Editor' and 'Visualization'. The main area is titled 'Step 4' and shows a conversation flow with four steps:

1. Search for the answer: 'Would you like to run the skill to retrieve your certificate's expiration date?'
2. Confirmation: '2 is Yes'
3. Free text: 'What is your certificate label?' (with a 'Free text' button)
4. Subaction: '3 is Defined' (with a 'Go to subaction: Retrieve certifica...' button)

The 'Is taken' dropdown is set to 'with conditions'. A condition is defined: '3. What is your certifi...' is defined. The 'And then' section shows a 'Go to a subaction' action with 'Retrieve certificate expiration' selected. The 'New step +' button is highlighted with a green box.

### 23. Click the **Is taken** option list and select **with conditions**.

In the previous step, you configured the assistant to run the Retrieve certificate expiration subaction you created, passing the certificate label the user inputted to the skills inputs. Recall when the **Retrieve certificate expiration** skill flow was created, the output form at the end of the skill flow was hidden. That form contained the expiration date. As a result, nothing is returned when running the subaction in the previous step. Now, configure the custom-action to provide that output as a response.

The screenshot shows the IBM Watsonx Orchestrator AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrator', 'AI assistant builder', 'Zeeves', and various icons. The main area is titled 'z/OS certificate expires soon'. On the left, there's a 'Conversation steps' pane with steps 1 through 5. Step 1: 'Search for the answer'. Step 2: 'Confirmation' with a link to 'Continue to next step'. Step 3: 'What is your certificate label?' with a 'Try Free text' button. Step 4: 'This step has no content'. Step 5: 'This step has no content' with a link to 'Continue to next step'. The right side shows 'Step 5' configuration under 'Is taken': 'without conditions' is selected, with 'with conditions' also visible. Below this is the 'Assistant says' field, which contains the text 'For example. What size do you want to order?'. The 'Conditions' section is collapsed. The 'And then' section also contains a 'Continue to next step' link.

24. Enter the following text in the **Assistant says** field.

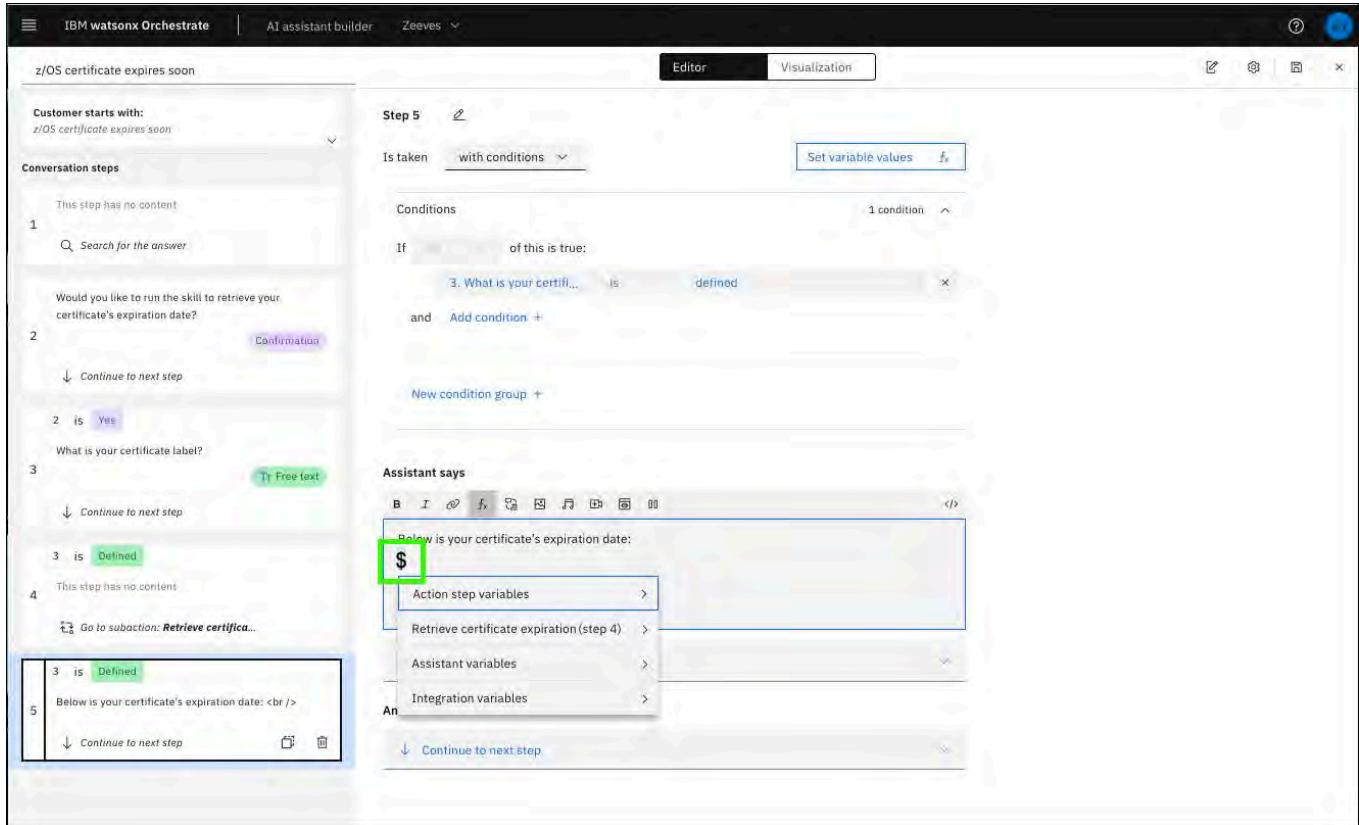
**Assistant says:**

Below is your certificate's expiration date:

This screenshot shows the same interface as above, but with changes made to Step 5. The 'Is taken' dropdown now shows 'with conditions' selected. In the 'Conditions' section, there is one condition defined: 'If 3. What is your certifi... is defined' (with the condition text highlighted by a green dashed box). The 'Assistant says' field now contains the text 'Below is your certificate's expiration date:' (also highlighted by a green box). The rest of the interface remains largely the same, with the conversation steps and other configurations visible.

25. While still in the **Assistant says** field, press **return** and then type `$`.

 The `$` is a special key that lists available functions. The following image is edited to show that you must type the `$`, but it is not displayed on your screen.



The screenshot shows the IBM Watsonx Orchestrator interface. On the left, there's a sidebar with 'Conversation steps' numbered 1 through 5. Step 1: 'Search for the answer'. Step 2: 'Confirmation' with a link to 'Continue to next step'. Step 3: 'Free text' with a link to 'Continue to next step'. Step 4: 'Action step variables' with a link to 'Retrieve certificate expiration (step 4)'. Step 5: 'Integration variables' with a link to 'Continue to next step'. The main area is titled 'Step 5' and shows 'Is taken with conditions'. A condition is defined: 'If 3. What is your certifi... is defined'. The 'Assistant says' field contains the text 'Below is your certificate's expiration date:' followed by a red box containing the character '\$'. This '\$' is highlighted with a green box. Below the '\$' are several dropdown menus: 'Action step variables', 'Retrieve certificate expiration (step 4)', 'Assistant variables', and 'Integration variables'.

26. Click **Retrieve certificate expiration (step 4)** and then click **Retrieve certificate expiration result variable**.

The screenshot shows the IBM Watsonx Assistant builder interface. On the left, there's a sidebar titled 'z/OS certificate expires soon' under 'Customer starts with:'. Below it is a 'Conversation steps' section with five numbered steps:

- This step has no content.
- Q. Search for the answer. Confirmation: Continue to next step.
- Would you like to run the skill to retrieve your certificate's expiration date? (Yes)
- This step has no content.
- Below is your certificate's expiration date: <br /> (Defined)

In the main area, Step 5 is selected. The 'Step 5' configuration panel shows:

- Is taken:** with conditions
- Conditions:** If All of this is true:
  - 3. What is your certifi... is defined
  - and Add condition +
- New condition group +**
- Assistant says:**
  - Action step variables (highlighted with a yellow circle)
  - Retrieve certificate expiration (step 4) (highlighted with a yellow circle)
  - Assistant variables
  - Integration variables
- And then:** Continue to next step

27. Review the **Assistant says** field and then click **Save** (💾).

The screenshot shows the same interface after saving. The 'Assistant says' field now contains the text:

Below is your certificate's expiration date:  
1 to 4. Retrieve certificate expiration res

A green dashed box surrounds the 'Assistant says' field, and a green square surrounds the 'Save' button in the top right corner.

Test the **z/OS certificate expires soon** custom-built skill

Before completing the use case, test the **z/OS certificate expires soon** custom-built skill that uses the **DEMOCERT** certificate created earlier.

## 1. Click **Preview**.

## 2. Enter the following prompt in the preview.

### **Prompt:**

My z/OS certificate is going to expire soon. How do I retrieve the expiration date for my certificate?

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and tabs for 'Editor' and 'Visualization'. The main area displays a conversation flow:

- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  1. Q: Search for the answer  
Would you like to run the skill to retrieve your certificate's expiration date?
  2. Confirmation  
↓ Continue to next step
  3. What is your certificate label?  
3 is Yes  
↓ Continue to next step
  4. This step has no content  
Go to subaction: Retrieve certifica...
  5. 3 is Defined  
Below is your certificate's expiration date:  
1 2 3 Step 4  
↓ Continue to next step
- Step 5:** Is taken with conditions
  - Conditions: If 3 is defined of this is true:
    - 3. What is your certifi... is defined
    - and Add condition +
  - New condition group +
- Assistant says:**
  - Define customer response
  - And then
    - ↓ Continue to next step
- Preview:**
  - 7:55 AM Greet customer [default]: Welcome, how can I assist you?
  - My z/OS certificate is going to expire soon. How do I retrieve the expiration date for my certificate? ➤

### 3. Review the response and click Yes.

The assistant responds by calling Conversational search and returns a response by using the Z RAG, displaying the RACDCERT command that can be used. The assistant then prompts Would you like to run the skill to retrieve?.

The screenshot shows the IBM Watsonx Orchestrate interface with the following details:

- Conversation steps:**
  1. Q: Search for the answer  
Would you like to run the skill to retrieve your certificate's expiration date?
  2. Confirmation  
↓ Continue to next step
  3. What is your certificate label?  
3 is Yes  
↓ Continue to next step
  4. This step has no content  
Go to subsection: Retrieve certifica...
  5. 3 is Defined  
Below is your certificate's expiration date:  
1. Step 4  
↓ Continue to next step
- Step 5:** Is taken with conditions
  - Conditions: If 3. What is your certif... is defined
  - and Add condition +
- Assistant says:**
  - Below is your certificate's expiration date:  
1. Step 4. Retrieve certificate expiration re...
  - Define customer response
  - And then  
↓ Continue to next step
- Preview:**
  - 10:06 AM
  - Conversational search called
  - Search info
  - To retrieve the expiration date for your z/OS certificate, you would typically check the certificate itself or consult the system logs. However, based on the provided context, there isn't a direct command or utility mentioned for this purpose. You may need to contact your system administrator or refer to your specific system's documentation for precise instructions. Generally, you can use commands like 'ODTE' or 'RACDCERT' to list and inspect certificates.
  - Accuracy of generated answers may vary.
  - Would you like to run the skill to retrieve your certificate's expiration date?  
Yes No
  - Type something...

#### 4. Review the response and enter DEMOCRT .

**Prompt:**

DEMOCRT

The screenshot shows the IBM Watsonx Orchestrate AI Assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and tabs for 'Editor' (selected) and 'Visualization'. The main workspace displays a conversation flow:

- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  - This step has no content.
  - Q Search for the answer
    - Would you like to run the skill to retrieve your certificate's expiration date?
  - Confirmation
    - ↓ Continue to next step
  - is Yes
    - What is your certificate label?
  - is Defined
    - This step has no content
  - Go to subsection: Retrieve certificate...
- Step 5** (highlighted in blue)
  - Is taken with conditions
  - Set variable values
  - Conditions (1 condition):
    - If All of this is true:
      - 3. What is your certifi... is defined
      - and Add condition +
  - New condition group +
- Assistant says**
  - Define customer response
  - And then
    - ↓ Continue to next step
- Preview**
  - Accuracy of generated answers may vary.
  - Would you like to run the skill to retrieve your certificate's expiration date?
    - Yes (highlighted)
  - You 10:07 AM
  - 10:07 AM z/OS certificate expires soon recognized
  - What is your certificate label? (highlighted)
  - DEMOCERT

5. Click **Apply**.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, a sidebar titled 'z/OS certificate expires soon' lists 'Conversation steps'. Step 1: 'Search for the answer' (Q: Would you like to run the skill to retrieve your certificate's expiration date?). Step 2: 'Confirmation' (A: Continue to next step). Step 3: 'What is your certificate label?' (T: Try Free text) (A: Continue to next step). Step 4: 'This step has no content' (A: Go to subaction: Retrieve certificate expiration). Step 5: 'Is taken with conditions' (Condition: 3. What is your certifi... is defined). Step 6: 'Define customer response' (And then: Continue to next step). A preview window on the right shows a user message 'z/OS certificate expires soon' and an AI response 'What is your certificate label?'. A modal for the 'Retrieve certificate expiration' skill is open, showing an ID field with '20' and an 'Apply' button.

## 6. Review the response.

If you see the following response (the date may differ), the custom-built skill ran successfully. The output of the skill flow was not the entire output of the z/OS Certs – List Cert Ansible job, but rather the certificate expiration date that was extracted from the full job output by using the Regular Expression transformation.

## Complete the custom-built skill to renew the certificate

Now that the custom-built action is working, add steps to include the certificate renewal process. After retrieving and displaying the user's certificate expiration date, ask the user if they want to renew the certificate, and if so, prompt for the new date and renew the certificate.

### 1. Click **Next step +**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and tabs for 'Editor' and 'Visualization'. The main area displays a conversation flow titled 'z/OS certificate expires soon'. The flow consists of five steps:

- Step 1:** 'Customer starts with: z/OS certificate expires soon' (Conversation steps: This step has no content)
- Step 2:** 'Search for the answer' (Confirmation: Would you like to run the skill to retrieve your certificate's expiration date?)
- Step 3:** 'What is your certificate label?' (Free text: Try Free text)
- Step 4:** 'This step has no content'
- Step 5:** 'Is taken with conditions' (Conditions: If 3. What is your certifi... is defined, and Add condition +)

The 'Assistant says' section contains the response: "Below is your certificate's expiration date: 1 2 3 4. Retrieve certificate expiration res". The 'Define customer response' section shows the condition: "3 is Defined". The 'And then' section contains the action: "1 2 3 Step 4".

**Preview:**

- 10:18 AM: Conversational skill called go to Retrieve certificate expiration
- 10:18 AM: id \* 2B
- 10:18 AM: 10:18 AM Conversational skill called
- 10:18 AM: returned to z/OS certificate expires soon
- 10:18 AM: Below is your certificate's expiration date: ("certificate expiration date": "2030/01/30 23:59:59")
- 10:18 AM: There are no additional steps for this action. Add a new step or end the action.

A text input field at the bottom right says 'Type something...'. A blue button labeled 'New step +' is located at the bottom left of the main editor area.

2. Click the **Is taken** option list and select **with conditions**.

The screenshot shows the IBM Watsonx Assistant AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main area displays a conversation flow titled 'z/OS certificate expires soon'. The flow consists of six steps:

- Step 1:** 'Customer starts with: z/OS certificate expires soon'. The 'Is taken' dropdown is set to 'without conditions' (highlighted with a green box). Below it is a 'Set variable values' button.
- Step 2:** 'Conversation steps' section. Step 1 is listed with 'This step has no content' and a 'Continue to next step' button.
- Step 3:** 'Assistant says' field: 'Would you like to run the skill to retrieve your certificate's expiration date?'. Step 2 is listed with 'Confirmation' and a 'Continue to next step' button.
- Step 4:** '2 is Yes'. Step 3 is listed with 'What is your certificate label?' and a 'Tir Free text' button.
- Step 5:** Step 4 is listed with 'Below is your certificate's expiration date:' and a '111 Step 4' button.
- Step 6:** Step 5 is listed with 'This step has no content' and a 'Continue to next step' button. This step is highlighted with a green dashed border.

3. Enter the following text in the **Assistant says** field.

**Assistant says:**

Would you like to renew your certificate?

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and tabs for 'Editor' and 'Visualization'. The main area displays a conversation flow:

- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  - Step 1:** Is taken with conditions. Condition: If 1 is defined of this is true: 1. Retrieve certificate is defined.
  - Step 2:** Confirmation. Assistant says: Would you like to renew your certificate?
  - Step 3:** is Yes. Assistant says: Below is your certificate's expiration date: Step 4.
  - Step 4:** is Defined. Assistant says: Would you like to renew your certificate?
  - Step 5:** Go to subaction: Retrieve certificate. Assistant says: Would you like to renew your certificate?
  - Step 6:** is Defined. Assistant says: Would you like to renew your certificate?

The 'Assistant says' sections for steps 2, 3, and 6 are highlighted with a green box. A dropdown menu for step 3 is open, showing 'Define customer response' selected. Other options include 'Text' and 'Image'.

- Click the **Define customer response** option list and select **Confirmation**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. On the left, the conversation steps are listed:

- z/OS certificate expires soon**
- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  - 1** 4 is Defined  
Would you like to renew your certificate?  
↓ Continue to next step
  - 2** Confirmation  
Would you like to run the skill to retrieve your certificate's expiration date?  
↓ Continue to next step
  - 3** 2 is Yes  
What is your certificate label?  
↓ Continue to next step
  - 4** 3 is Defined  
This step has no content.  
Go to subaction: Retrieve certifica...
  - 5** 5 is Step 4  
Below is your certificate's expiration date:  
↓ Continue to next step
  - 6** 4 is Defined  
Would you like to renew your certificate?  
↓ Continue to next step

**Step 1** configuration:

- Is taken with conditions
- Conditions:
  - If 1 is defined of this is true:
    - 1 1.1 Retrieve certific... is defined
    - and Add condition +
- Assistant says:
  - Would you like to renew your certificate?

A modal window titled "Define customer response" is open, showing the "Confirmation" configuration:

- System**: Confirmation
- Options**: Enable customers to select from Yes or No response.
- Confirmation** (highlighted with a green box): As buttons
- Buttons**: Yes, No
- Text**: TT Free text, Regex, Number, Date, Time

5. Click **Next step +**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and various icons for search, refresh, and export.

The main area displays a conversation flow titled 'z/OS certificate expires soon'. The flow consists of six steps:

- Step 1:** 'Is taken' condition. If '1 Retrieve certificate...' is defined, then... (highlighted with a green dashed box)
- Step 2:** Confirmation message: 'Would you like to renew your certificate?' with a 'Continue to next step' button.
- Step 3:** Confirmation message: 'Would you like to run the skill to retrieve your certificate's expiration date?' with a 'Continue to next step' button.
- Step 4:** Confirmation message: 'What is your certificate label?' with a 'Continue to next step' button.
- Step 5:** Confirmation message: 'Below is your certificate's expiration date:' followed by a placeholder '1 Step 4' and a 'Continue to next step' button.
- Step 6:** Confirmation message: 'Would you like to renew your certificate?' with a 'Continue to next step' button and a 'New step +' button highlighted with a green box.

The right side of the interface shows the 'Editor' tab selected, with sections for 'Step 1' conditions, 'Assistant says' (with a rich text editor), and 'And then' (with a 'Continue to next step' button). There are also 'View response' and 'Edit validation' options for the current step.

#### 6. Click the **Is taken** option list and select **with conditions**.

This step handles the flow in which the user selects **Yes** in the previous step indicating they want to renew their expiring certificate. Before initiating the Cert Renewal skill flow action to automate this, the assistant first needs the new expiration date for the certificate.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The workflow consists of the following steps:

- Step 1:** A conditional step where the condition "Is taken with conditions" is selected. The condition "1 Retrieve certific..." is defined as "is defined".
- Step 2:** A step asking "Would you like to run the skill to retrieve your certificate's expiration date?".
- Step 3:** A confirmation step where the user is asked if they want to continue.
- Step 4:** A free text step asking "What is your certificate label?".
- Step 5:** A step asking "Below is your certificate's expiration date:".
- Step 6:** A step labeled "Step 5" with a "Continue to next step" button.
- Step 7:** A confirmation step where the user is asked if they want to renew their certificate.

**Assistant says:**

Would you like to renew your certificate?

**And then:**

Continue to next step

## 7. Enter the following text in the **Assistant says** field.

**Assistant says:**

What date would you like to set the renewed certificate's expiration date to? Please enter in the form of YYYY-MM-DD.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. A skill flow is displayed on the left, and the 'Editor' tab is selected on the right. Step 7 is highlighted with a green dashed box. The condition 'Is taken with conditions' is set to 'If All of this is true: 6. Would you like to... is Yes'. The customer response field is highlighted with a green box and contains the placeholder 'What date would you like to set the renewed certificate's expiration date to? Please enter in the form of YYYY-MM-DD.' The 'Define customer response' dropdown is also highlighted with a green box.

- Click the **Define customer response** option list and select **Free text**.

**z/OS certificate expires soon**

**Step 7**

Is taken with conditions

Conditions 1 condition ^

If all of this is true:

6. Would you like to r... is Yes

and Add condition +

New condition group +

**Assistant says**

What date would you like to set the renewed certificate's expiration date to? Please enter in the form of YYYY-MM-DD.

**Define customer response**

System Free text

Options

Confirmation

Tr Free text  As text

Regex

Number

Date

Time

**Step 7**

Search for the answer

Would you like to run the skill to retrieve your certificate's expiration date?

Confirmation

Continue to next step

2 is Yes

What is your certificate label?

3 Try Free text

Continue to next step

3 is Defined

This step has no content

4 Go to subsection: Retrieve certifica...

3 is Defined

Below is your certificate's expiration date:

5 Step 4

Continue to next step

3 is Defined

Would you like to renew your certificate?

6 Confirmation

Continue to next step

6 is Yes

What date would you like to set the renewed certificate's expiration date to? Please enter in the...

7 Continue to next step

## 9. Click New step +.

The screenshot shows the IBM Watson Assistant AI assistant builder interface. The top navigation bar includes 'IBM Watson Assistant', 'AI assistant builder', 'Zeeves', and a search bar. The main workspace displays a skill flow with several steps:

- Step 1:** 'z/OS certificate expires soon' (Confirmation)
- Step 2:** 'Would you like to run the skill to retrieve your certificate's expiration date?' (Confirmation)
- Step 3:** 'What is your certificate label?' (Free text)
- Step 4:** 'This step has no content' (Confirmation)
- Step 5:** 'Go to subaction: Retrieve certificate...' (Confirmation)
- Step 6:** 'Would you like to renew your certificate?' (Confirmation)
- Step 7:** 'Is taken with conditions' (Confirmation)
  - Condition: If 6. Would you like to r... is Yes
  - And: Add condition +

The 'Assistant says' section contains a text input field with the placeholder 'What date would you like to set the renewed certificate's expiration date to? Please enter in the form of YYYY-MM-DD.' A green dashed box highlights the 'User enters free text' input field.

#### 10. Click the **Is taken** option list and select **with conditions**.

With the new expiration date entered by the user, the next step is to run the Cert Renewal skill flow action as a subaction. Next, trigger the renewal skill flow and pass the user provided details as input to the action to renew the certificate and extend the certificates expiration date.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and a user icon. The main area displays a flowchart titled 'z/OS certificate expires soon' with steps numbered 1 through 8.

- Step 1:** Confirmation (purple box)
- Step 2:** Confirmation (purple box)
- Step 3:** Free text (green box)
- Step 4:** This step has no content
- Step 5:** Go to subaction: *Retrieve certificate...*
- Step 6:** Confirmation (purple box)
- Step 7:** Free text (green box)
- Step 8:** Configuration panel with the following settings:
  - Is taken:** A dropdown menu with three options: 'without conditions' (selected), 'with conditions', and 'with conditions' (disabled).
  - Assistant says:** A text input field containing the placeholder 'For example: What type of transfer would you like to make?' Below it is a 'Define customer response' button.
  - And then:** A dropdown menu with the option 'Continue to next step'.

## 11. Enter the following text in the **Assistant says** field.

This assistant first responds with the message that follows before triggering the certificate renewal skill-flow. When performing a demo of this use case, mention the [z/OS Certs – Search and Renew](#) Ansible playbook typically takes a minute or so to complete.

### Assistant says:

Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The workflow is titled "z/OS certificate expires soon".

- Step 1:** Confirmation (z/OS certificate expires soon)
- Step 2:** Confirmation (Continue to next step)
- Step 3:** Confirmation (2 is Yes)
  - What is your certificate label?
  - Try Free text
- Step 4:** Confirmation (3 is Defined)
  - This step has no content
  - Go to subaction: Retrieve certificate...
- Step 5:** Confirmation (1 is Step 4)
  - 1 is Step 4
  - Continue to next step
- Step 6:** Confirmation (6 is Yes)
  - Would you like to renew your certificate?
  - Continue to next step
- Step 7:** Confirmation (7 is Defined)
  - What date would you like to set the renewed certificate's expiration date to? Please enter in the...  
Try Free text
  - Continue to next step
- Step 8:** Confirmation (8 is Defined)
  - Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...  
Try Free text
  - Continue to next step

**Step 8 Conditions:**

If [Condition] of this is true:

- 7. What date would y... is defined
- and Add condition +

**Assistant says:**

Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.

**And then:**

Continue to next step

12. Click the **And then** option list and select **Go to a subaction**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and various icons. The main area displays a step configuration for handling certificate expiration.

**Step 8:** Confirmation

**Condition:** Is taken with conditions

**Conditions:** If 7. What date would you like to set the renewed certificate's expiration date to? is defined

**Assistant says:**

Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.

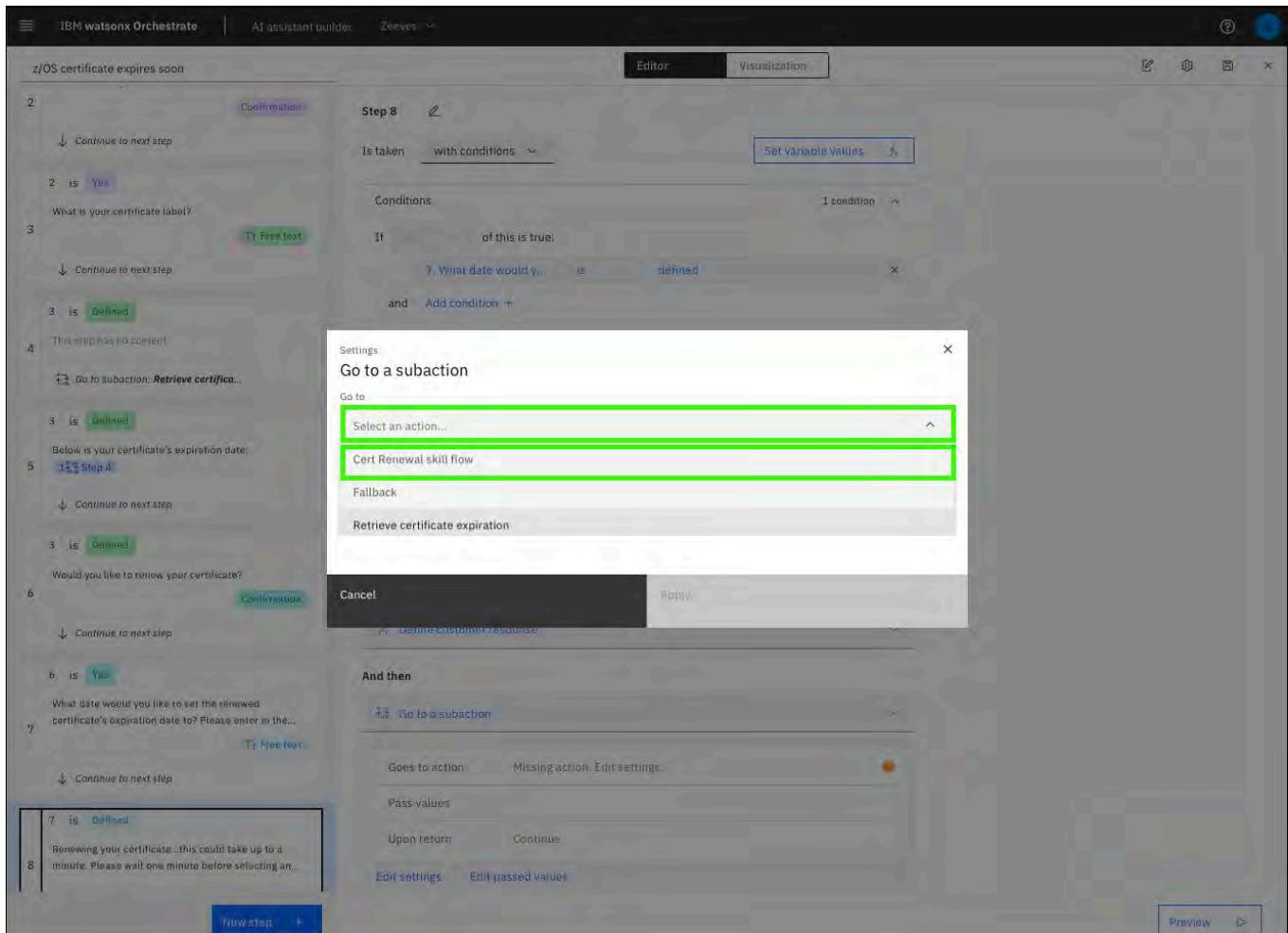
**And then:**

- Continue to next step (highlighted with a green box)
- Re-ask previous step(s)
- Go to a subaction (highlighted with a green box)
- Use an extension
- Search for the answer
- Connect to agent
- End the action

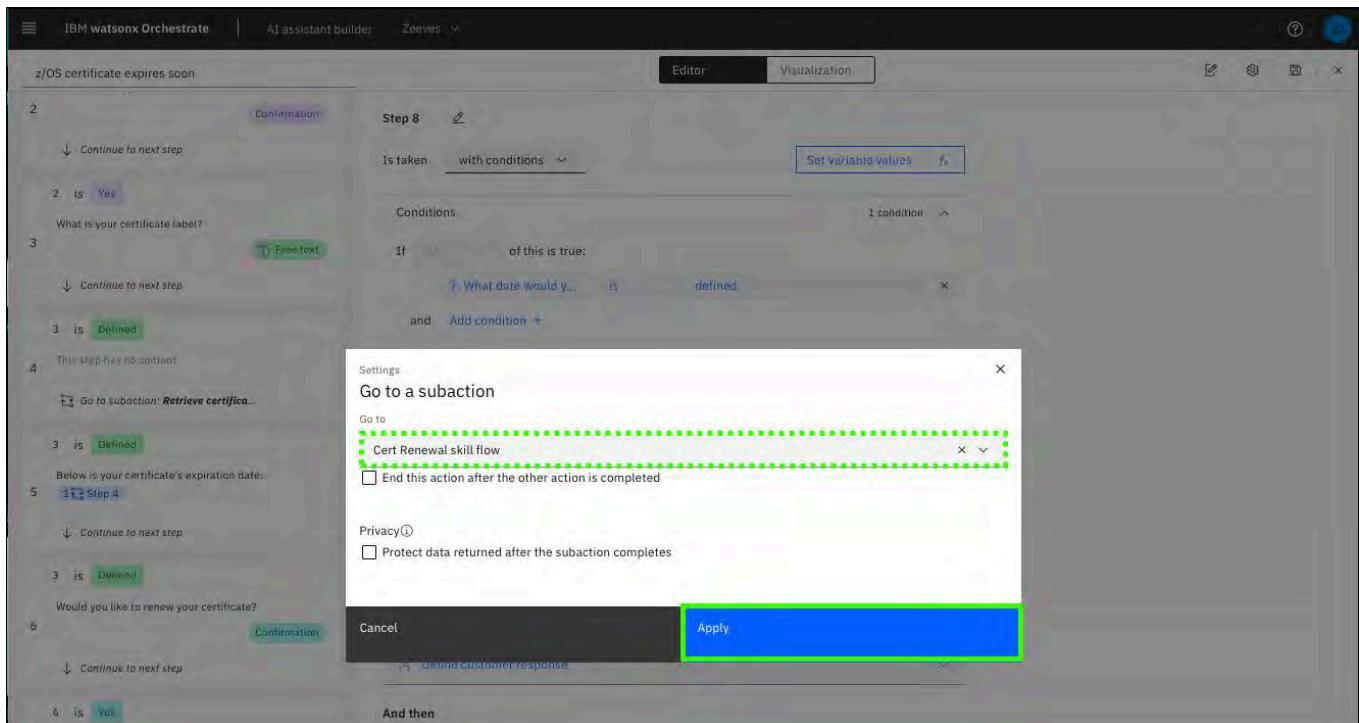
**Buttons at the bottom:**

- New step +
- Preview
- Done

13. Click the **Go to** option list and select the **Cert Renewal skill flow**.



#### 14. Click Apply.



#### 15. Click Edit passed values.

Edit the passed values to use them in the **Cert Renewal** skill flow subaction.

z/OS certificate expires soon

**Step 8**

Is taken with conditions

Conditions

If All of this is true:

7. What date would you like to set the renewed certificate's expiration date to? Please enter in the... is defined

and Add condition +

New condition group +

**Assistant says**

Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.

**And then**

Go to a subaction

Goes to action Cert Renewal skill flow

Pass values

Upon return Continue

Edit settings Edit passed values

16. Click Set new value + and then select extra\_vars.cert\_label\_survey.

z/OS certificate expires soon

**Step 8**

Is taken with conditions

Conditions

Passed values

Edit passed values

Configure how your skill-based action leverages and stores information in assistant variables. [Learn more](#)

Variables are not private if they are shared with Skill Studio.

**Edit variable values** Response behavior

The values you select from this action will be passed to "Cert Renewal skill flow" skill.

Set new value +

< All variables

1. extra\_vars.cert\_label\_survey

2. extra\_vars.cert\_type\_survey

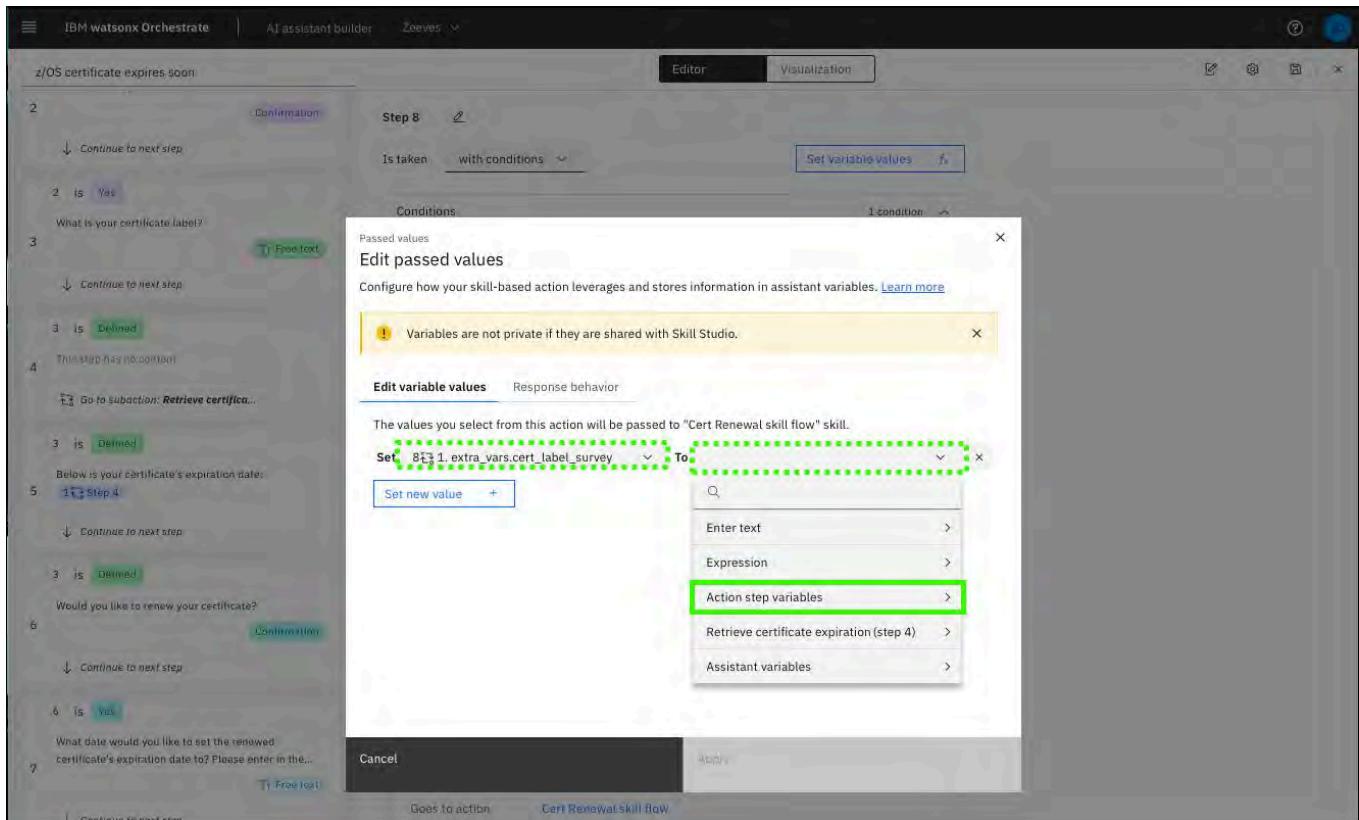
3. extra\_vars.keyring\_survey

4. extra\_vars.new\_expiry\_date\_survey

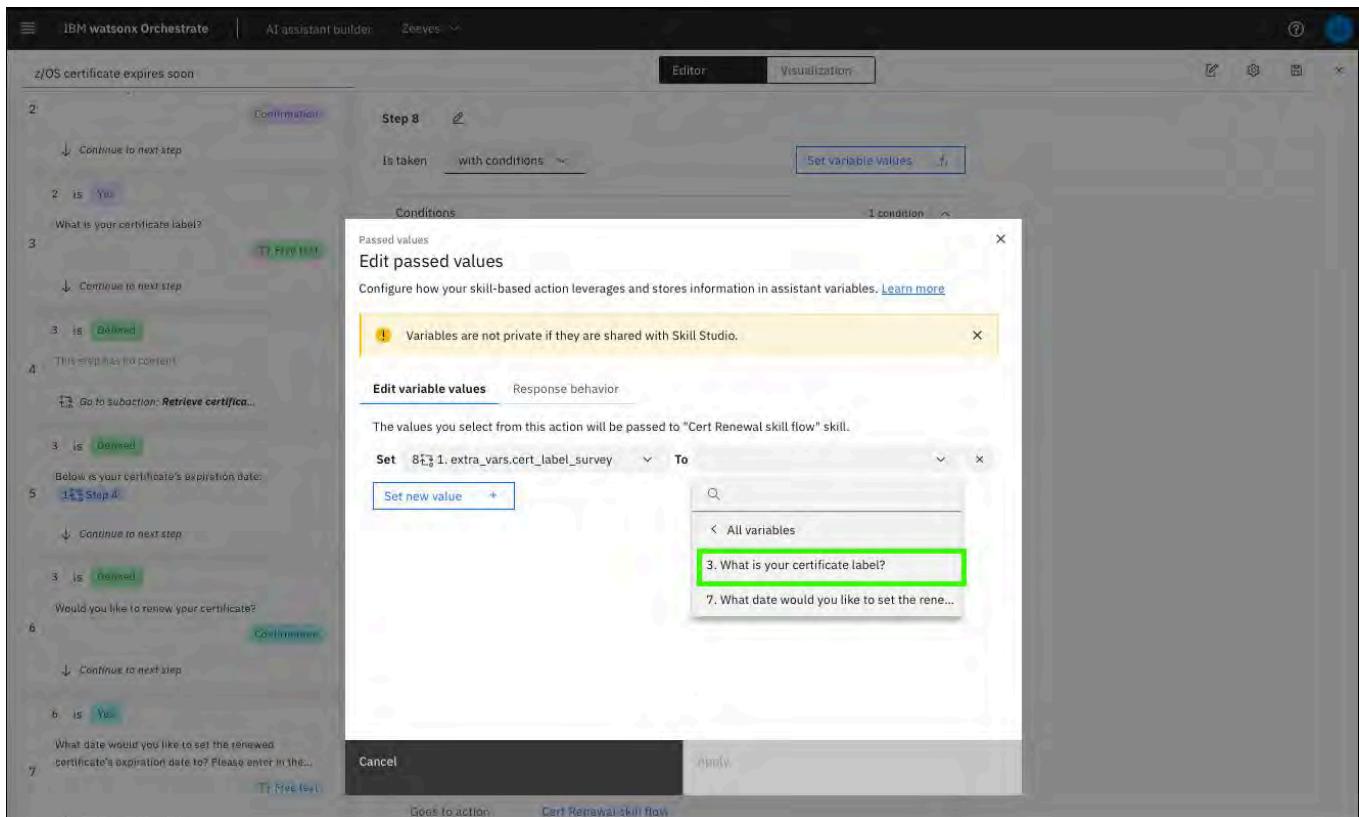
5. extra\_vars.owner\_id\_survey

Cancel Apply

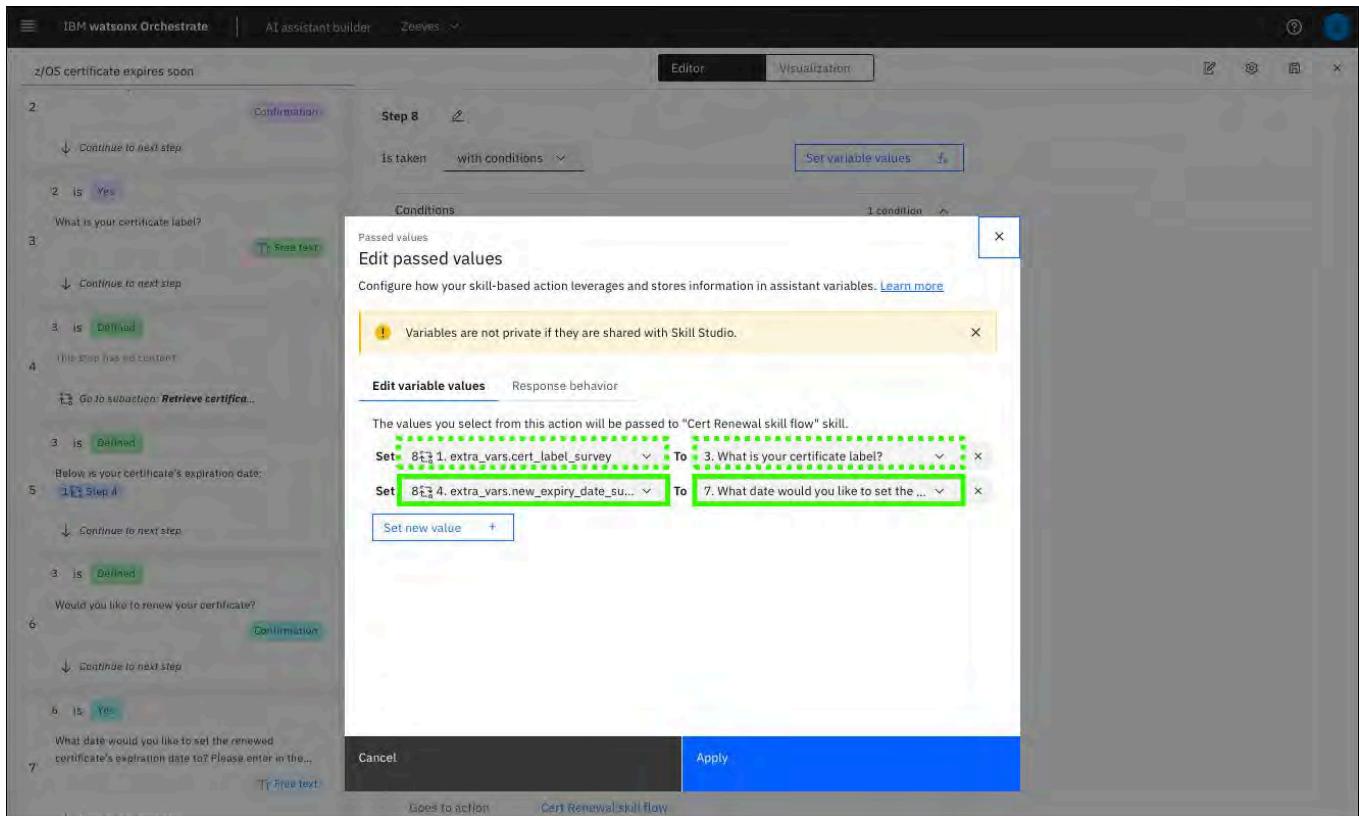
17. In the To field, select Action step variables.



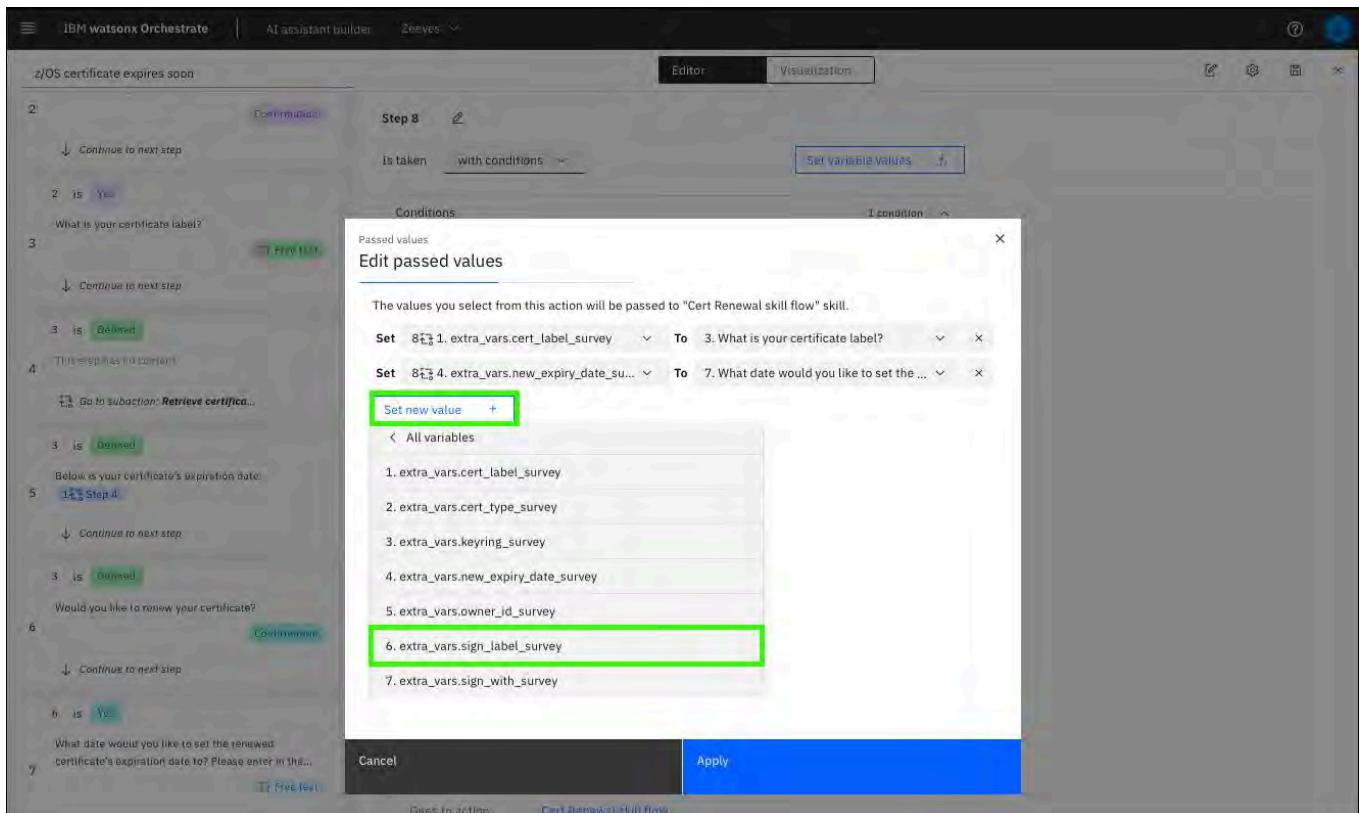
### 18. Click What is your certificate label?



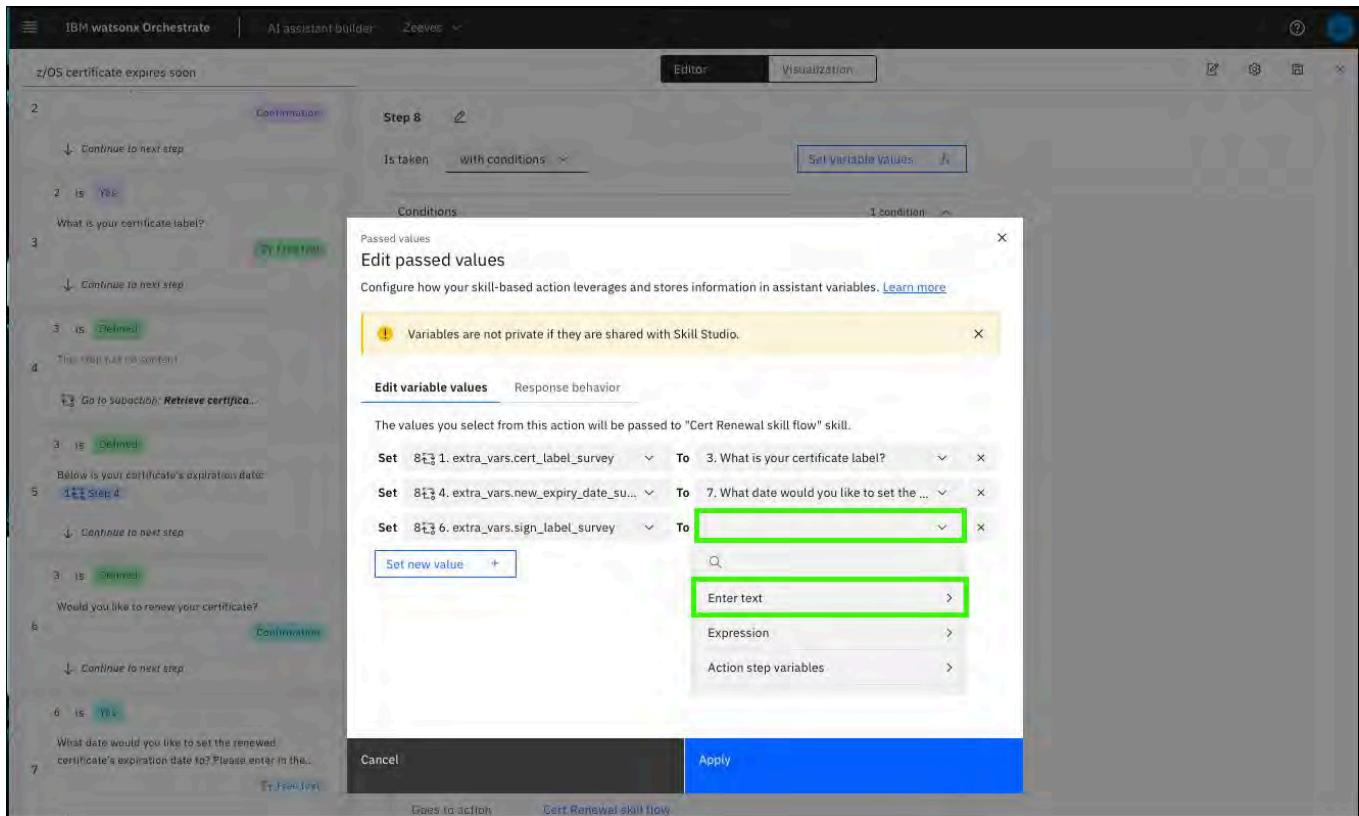
### 19. Repeat steps 16 - 18 adding the extra\_vars.new\_expiry\_date\_survey input variable and What date would you like to set the... in the To field.



20. Click **Set new value +** and then select **extra\_vars.sign\_label\_survey**.



21. In the **To** option list, select **Enter text**.

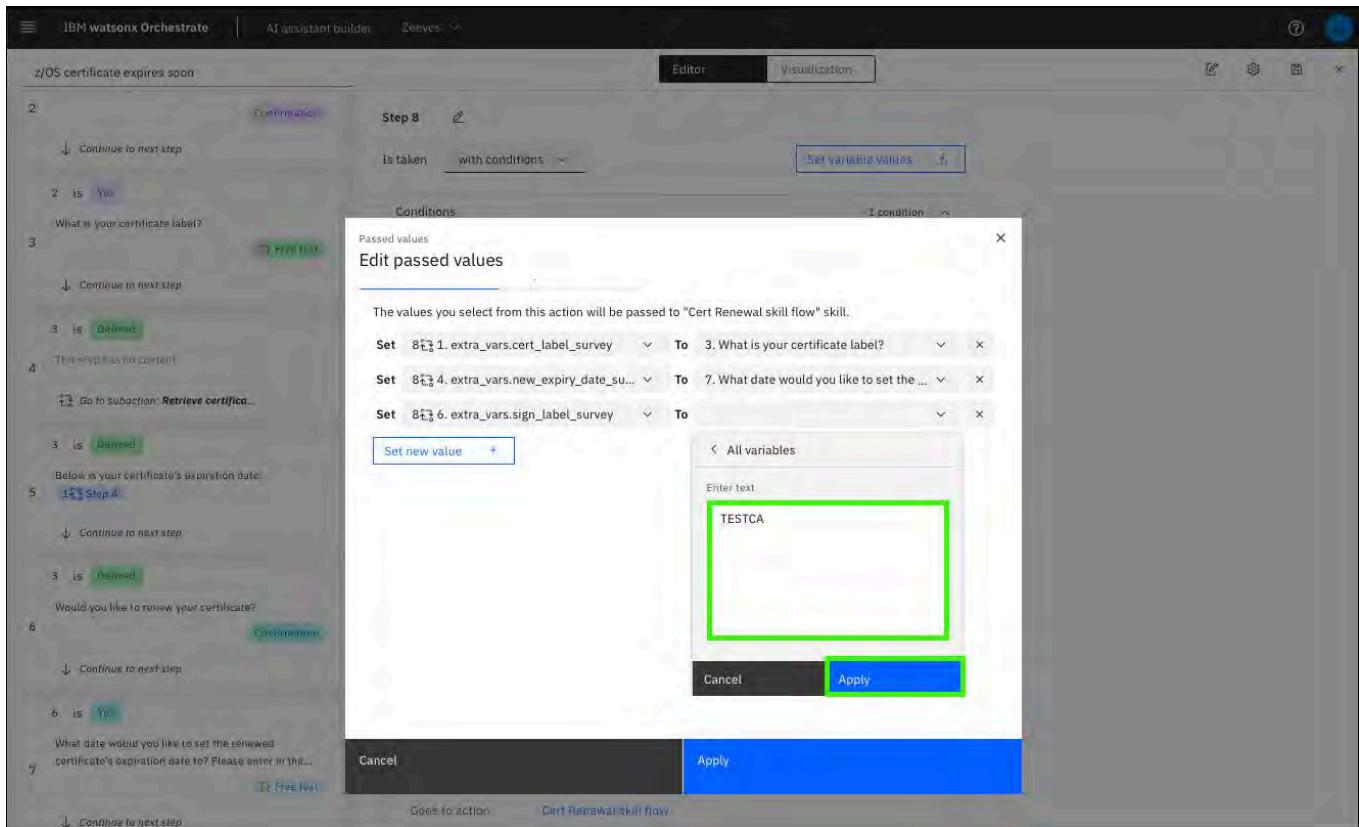


22. Enter **TESTCA** in the **Enter text** field and click **Apply** for the **To** option list.

**Enter text:**

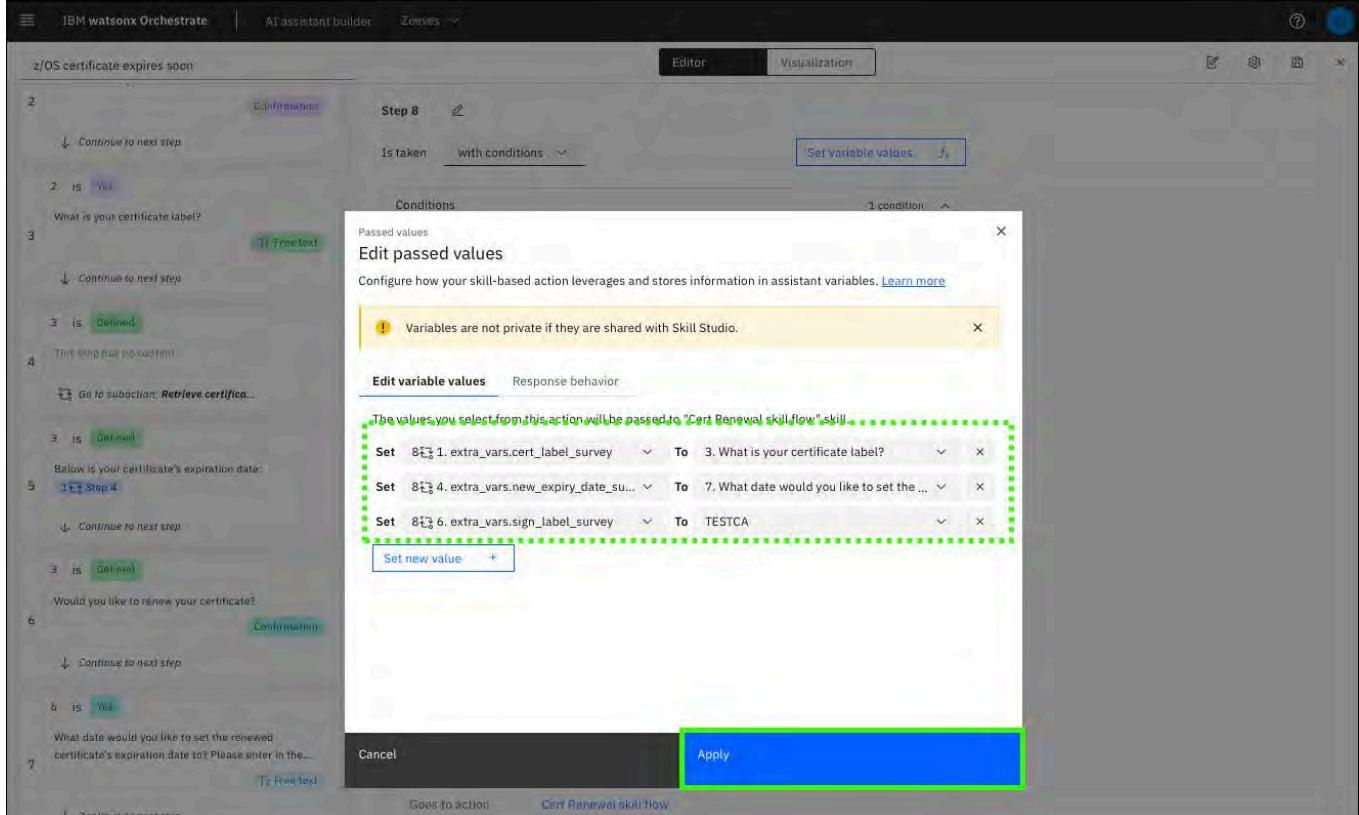
TESTCA

For this passed value, hardcode **TESTCA** in the skill flow's input for the **sign\_label** variable. This is the **CA** certificate created earlier for demo purposes in the **AAP web console**.



### 23. Review the **Edit passed values** and then click **Apply**.

Review all 3 variables are set correctly.



### 24. Click **Next step +**

**z/OS certificate expires soon**

**Step 8**

Is taken with conditions

Conditions

If 7. What date would y... is defined

New condition group +

Assistant says

Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.

And then

Go to a subaction

Goes to action: Cert Renewal skill flow

Pass values:

- 1. extra\_vars.cert\_label\_survey
- 4. extra\_vars.new\_expiry\_date\_survey
- 6. extra\_vars.sign\_label\_survey

Upon return: Continue

Edit settings Edit passed values

25. Click the **Is taken** option list and select **with conditions**.

To complete the flow, ask the user if they want to verify the new expiration date.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and a search bar. The main workspace displays a workflow with the following steps:

- Step 1:** 'z/OS certificate expires soon' (Free text)
- Step 2:** 'What is your certificate issuer?' (Free text)
- Step 3:** 'Is taken without conditions' (Free text)
- Step 4:** 'This step has no content'
- Step 5:** 'Go to subaction: Retrieve certificate...' (Free text)
- Step 6:** '3 is Defined' (Free text)
- Step 7:** 'Below is your certificate's expiration date:' (Text input field containing '11/Step 4')
- Step 8:** 'Continue to next step' (Free text)
- Step 9:** 'Assistant says' (Free text)
  - 'Is taken without conditions' (highlighted with a green box)
  - 'without conditions' (highlighted with a blue box)
  - 'with conditions' (highlighted with a green box)
- Step 10:** 'Set variable values' (button)
- Step 11:** 'Define customer response' (Free text)
- Step 12:** 'And then' (Free text)
- Step 13:** 'Continue to next step' (Free text)
- Step 14:** 'Confirmation' (Free text)
- Step 15:** 'Continue to next step' (Free text)
- Step 16:** '6 is Yes' (Free text)
- Step 17:** 'What date would you like to set the renewed certificate's expiration date to? Please enter in the...' (Text input field)
- Step 18:** '7 is Free text' (Free text)
- Step 19:** 'Continue to next step' (Free text)
- Step 20:** '7 is Defined' (Free text)
- Step 21:** 'Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...' (Text input field)
- Step 22:** 'Continue to next step' (Free text)
- Step 23:** 'Go to subaction: Cert Renewal ski...' (Free text)
- Step 24:** 'This step has no content' (Free text)
- Step 25:** 'Continue to next step' (Free text)

26. Enter the text that follows in the **Assistant says** field.

**Assistant says:**

Would you like to verify the new expiration date for your certificate?

The screenshot shows the IBM Watsonx Assistant AI assistant builder interface. The current step is Step 9, which is titled "Is taken with conditions". A condition is defined: "If 7. What date would you like to set the renewed certificate's expiration date to? is defined". Below this, there is a "New condition group" button. The "Assistant says" section contains a message: "Would you like to verify the new expiration date for your certificate?". This message is highlighted with a green border. Below it, there is a "Define customer response" dropdown menu with "Confirmation" selected. The "And then" section contains a "Continue to next step" button.

27. Click the **Define customer response** option list and select **Confirmation**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The main workspace displays a workflow titled "z/OS certificate expires soon". The workflow consists of several steps:

- Step 1:** "Continue to next step"
- Step 2:** "3 is Defined" (with a note: "This step has no content")
- Step 3:** "Go to subaction: Retrieve certificate..."
- Step 4:** "Below is your certificate's expiration date:  
12-31-2024 Step 4"
- Step 5:** "Continue to next step"
- Step 6:** "3 is Defined" (with a note: "Would you like to renew your certificate?")
- Step 7:** "Confirmation" (with a note: "What date would you like to set the renewed certificate's expiration date to? Please enter in the...")
- Step 8:** "6 is Yes" (with a note: "Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...")
- Step 9:** "7 is Defined" (with a note: "Would you like to verify the new expiration date for your certificate?")

On the right side of the interface, there is a "Step 9" configuration panel:

- Condition:** "Is taken with conditions" (with a "Set variable values" button)
- Conditions:** "If All of this is true:"  
7. What date would y... is defined
- Buttons:** "New condition group +", "Add condition +", "Assistant says" (with a rich text editor), and "Preview" button.

28. Click **Next step +**.

The screenshot shows the IBM Watsonx Assistant Editor interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and tabs for 'Editor' (selected) and 'Visualization'. The main workspace displays a skill-flow titled 'z/OS certificate expires soon'. Step 9 is currently selected, with the condition 'Is taken with conditions' chosen. A condition is defined: 'If 7. What date would you... is defined'. Step 7 is highlighted with a green box. Other steps shown include Step 3 (is Defined), Step 4 (This step has no content), Step 5 (Go to subaction: Retrieve certifica...), Step 6 (is Defined), Step 7 (What date would you like to set the renewed certificate's expiration date to? Please enter in the...), Step 8 (Free text), Step 9 (Would you like to verify the new expiration date for your certificate?), and Step 10 (Confirmation). A 'New step +' button is visible at the bottom right.

29. Click the **Is taken** option list and select **with conditions**.

On the condition that the user selected **Yes** in the previous step, configure a step to run the **Retrieve certificate expiration** skill-flow again to retrieve and display the new expiration date of the renewed certificate.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and various icons. The main workspace is titled 'z/OS certificate expires soon'. On the left, a vertical list of steps is shown:

- Step 1: Go to subaction: [Retrieve certifica...](#)
- Step 2: 3 is [Defined](#)
- Step 3: Below is your certificate's expiration date:  
5 [Step 4](#)
- Step 4: Continue to next step
- Step 5: 3 is [Defined](#)
- Step 6: Would you like to renew your certificate?  
6 [Confirmation](#)
- Step 7: Continue to next step
- Step 8: 6 is [Yes](#)
- Step 9: What date would you like to set the renewed certificate's expiration date to? Please enter in the...  
7 [Free text](#)
- Step 10: Continue to next step
- Step 11: 7 is [Defined](#)
- Step 12: Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...  
8 [Cert Renewal ski...](#)
- Step 13: Go to subaction: [Cert Renewal ski...](#)
- Step 14: 7 is [Defined](#)
- Step 15: Would you like to verify the new expiration date for your certificate?  
9 [Confirmation](#)
- Step 16: Continue to next step
- Step 17: 10 This step has no content.

On the right side, the 'Editor' tab is selected, showing the configuration for Step 10. The 'Is taken' dropdown is set to 'without conditions' (highlighted with a green box). Below it is a list of options: 'without conditions' (selected, highlighted with a blue box) and 'with conditions' (highlighted with a green box). A 'Set variable values' button is also present. The 'Assistant says' section contains placeholder text: 'For example: "What type of transfer would you like to make?"'. The 'And then' section is collapsed. The bottom right corner of the workspace has a blue button labeled 'Run'.

30. Click the **And then** option list and select **Go to a subaction**.

z/OS certificate expires soon

z/OS certificate expires soon

Step 10

Is taken with conditions

Set variable values

Conditions

If 9. Would you like to... is Yes

and Add condition +

New condition group +

Assistant says

For example: What type of transfer would you like to make?

Define customer response

And then

Continue to next step

Go to a subaction

Re-ask previous step(s)

Use an extension

Search for the answer

Connect to agent

End the action

Preview

31. Click the (a) Go to option list, select (b) Retrieve certificate expiration, and then click (c) Apply.

z/OS certificate expires soon

z/OS certificate expires soon

Step 10

Is taken with conditions

Set variable values

Conditions

If 9. Would you like to... is Yes

and Add condition +

Go to a subaction

Select an action...

Cert Renewal skill flow

Fallback

Retrieve certificate expiration

Cancel

Apply

And then

32. Click Edit passed values.

**Step 10**

Is taken with conditions

Conditions

If 3 is Defined of this is true:

9. Would you like to v... is Yes

and Add condition +

New condition group +

**Assistant says**

For example: What type of transfer would you like to make?

**Define customer response**

**And then**

10. Go to subaction: Retrieve certificate expiration

Goes to action: Retrieve certificate expiration

Pass values

Upon return Continue

Edit settings Edit passed values

33. Click Set new value + and then select extra\_vars.cert\_label\_survey.

**Step 10**

Is taken with conditions

Conditions

Passed values

Edit passed values

Configure how your skill-based action leverages and stores information in assistant variables. [Learn more](#)

Variables are not private if they are shared with Skill Studio.

**Edit variable values Response behavior**

The values you select from this action will be passed to "Retrieve certificate expiration" skill.

Set new value +

< All variables

1. extra\_vars.cert\_label\_survey

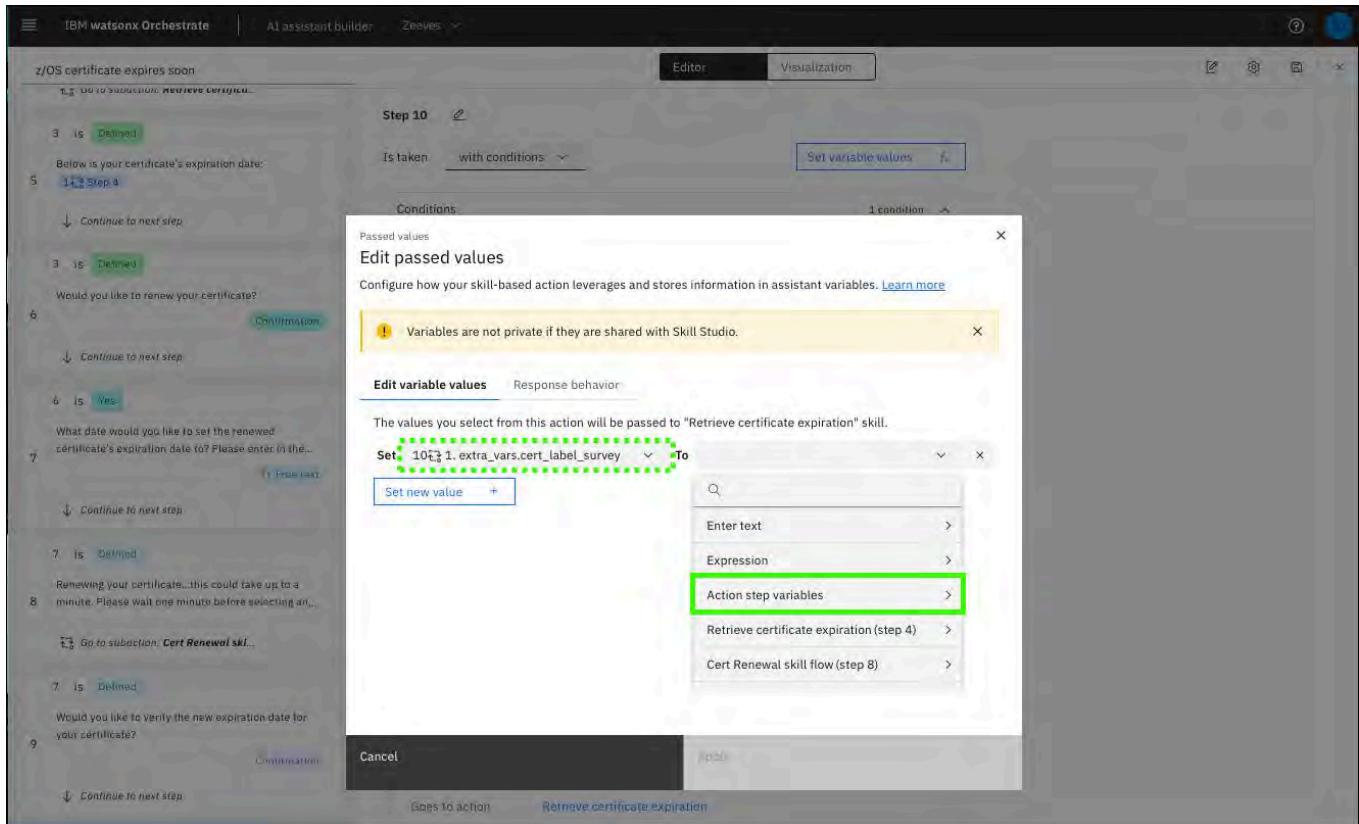
2. extra\_vars.cert\_type\_survey

3. extra\_vars.owner\_id\_survey

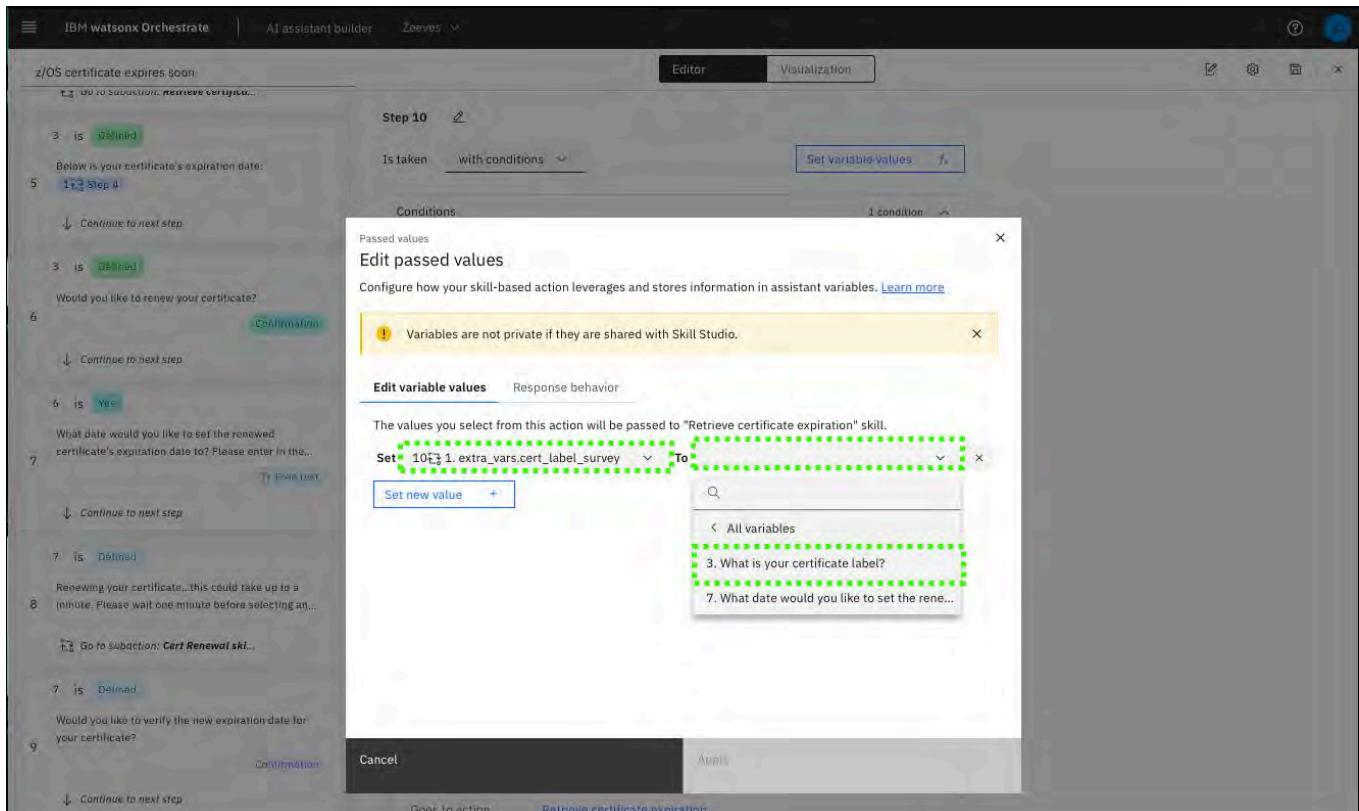
Cancel

Goes to action: Retrieve certificate expiration

34. In the To option list, click Action step variables.



35. Click What is your certificate label?.



36. Review the Edit variable values and click Apply.

z/OS certificate expires soon

Step 10

Is taken with conditions

Conditions

Passed values

Edit passed values

The values you select from this action will be passed to "Retrieve certificate expiration" skill.

Set 10: 3.1.extra\_vars.cert\_label\_survey To 3. What is your certificate label?

Cancel Apply

### 37. Click Next step +.

z/OS certificate expires soon

Step 10

Is taken with conditions

Conditions

If 9. Would you like to v... is Yes

New condition group +

Assistant says

For example: What type of transfer would you like to make?

Define customer response

And then

Go to a subaction

Goes to action Retrieve certificate expiration

Pass values 1. extra\_vars.cert\_label\_survey

Upon return Continue

Edit settings Edit passed values

New step +

38. Click the **Is taken** option list and select **with conditions**.

The final step is to display new expiration date of the certificate. Nothing is returned in the previous step when running the Retrieve certificate expiration skill flow - this was because the output form was hidden when the skill was created. In this step, provide the output as an assistant response to the user, with only the expiration date extracted from the full job output.

The screenshot shows the IBM Watsonx Orchestrator AI assistant builder interface. The current step is Step 11, which is part of a workflow titled "z/OS certificate expires soon". The "Step 11" tab is selected. In the "Is taken" dropdown menu, the option "with conditions" is highlighted with a green box. Below the dropdown, there is a "Set variable values" button. The "Assistant says" field is empty. The "And then" section below contains a "Continue to next step" link. The overall interface includes various buttons and icons for managing the workflow.

39. Enter the following text in the **Assistant says** field.

**Assistant says:**

Below is the new expiration date of your renewed certificate:

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and a search bar. The main area displays a workflow for certificate renewal:

- Step 11:** A conditional step labeled 'Is taken with conditions'. It checks if step 9 ('Would you like to renew your certificate?') is 'Yes'. A 'Set variable values' button is available.
- Conditions:** A detailed condition is shown: 'If All of this is true: 9. Would you like to v... is Yes'.
- Assistant says:** The text 'Below is the new expiration date of your renewed certificate:' is displayed in a green-bordered box.
- And then:** A continuation step labeled 'Continue to next step'.
- Step 11 (continued):** Shows the result of the condition: '9 is Yes' and the message 'Below is the new expiration date of your renewed certificate:'.

40. While still in the **Assistant says** field, press **return** and then type **\$**.



The **\$** is a special key that lists available functions. The following image is edited to show that you must type the **\$**, but it is not displayed on your screen.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and a search bar. The main area is titled 'z/OS certificate expires soon'.

**Step 11:** This step is taken "with conditions". A condition is defined: "If 9. Would you like to v... is Yes".

**Assistant says:** The response is "\$".

**Action step variables:** Retrieve certificate expiration (step 4)

**Conditions:** 1 condition

**Output:**

```

11 9 is Yes
Below is the new expiration date of your renewed
certificate: <br />

```

#### 41. Click **Retrieve certificate expiration (step 10)**.



Be sure to select the output from step 10 and not step 4.

**z/OS certificate expires soon**

**Step 11**

Is taken with conditions

Conditions

If 9. Would you like to v... is Yes

and Add condition +

New condition group +

**Assistant says**

Action step variables >

- Retrieve certificate expiration (step 4) >
- Cert Renewal skill flow (step 8) >
- Retrieve certificate expiration (step 10) >**
- Assistant variables >
- Integration variables >

Below is the new expiration date of your renewed certificate:

9 is Yes

Below is the new expiration date of your renewed certificate: <br />

↓ Continue to next step

#### 42. Click Retrieve certificate expiration result variable.

The screenshot shows the IBM Watsonx Assistant Editor interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', 'Editor' (selected), and 'Visualization'. The main area displays a step titled 'Step 11' with the condition 'Is taken with conditions'. A green box highlights the condition '9. Would you like to v... is Yes'. Below this, there's a 'Conditions' section with a 'New condition group' button. The 'Assistant says' section contains a list of variables, with '1. Retrieve certificate expiration result var...' highlighted by a green box. The 'And then' section has a 'Continue to next step' button. Step 11 itself contains a message about the new expiration date and a 'Continue to next step' button.

43. Click the **And then** option list and select **End the action**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The workflow is titled "z/OS certificate expires soon".

- Step 11:** A condition is set: "Is taken with conditions". A condition is defined: "If 9. Would you like to v... is Yes".
- Content:** "Below is the new expiration date of your renewed certificate: 11. Step 10. Retrieve certificate expiration n".
- Actions:** A dropdown menu under "And then" shows options: "Continue to next step" (highlighted with a green box), "End the action" (also highlighted with a green box), and other options like "Re-ask previous step(s)", "Go to a subaction", etc.
- Step 10:** "This step has no content".
- Step 11:** "9. is Yes". Content: "Below is the new expiration date of your renewed certificate: 11. Step 10. Retrieve certificate expiration n".
- Buttons at the bottom:** "New step" and "Preview".

44. Review the (a) final step, click (b) Save (), and then click (c) x.

**z/OS certificate expires soon**

↓ Continue to next step

3 is Defined

Would you like to renew your certificate?

6 Confirmation

↓ Continue to next step

6 is Yes

What date would you like to set the renewed certificate's expiration date to? Please enter in the...  
7 Free text

↓ Continue to next step

7 is Defined

Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...  
8

Go to subaction: Cert Renewal ski...

7 is Defined

Would you like to verify the new expiration date for your certificate?  
9 Confirmation

↓ Continue to next step

9 is Yes

This step has no content  
10 Go to subaction: Retrieve certific...

11 9 is Yes  
Below is the new expiration date of your renewed certificate: 11 Step 10  
Action complete

New step +

Editor Visualization

Step 11

Is taken with conditions Set variable values fx

Conditions 1 condition

If all of this is true:

9. Would you like to v... is Yes

and Add condition +

New condition group +

Assistant says

Below is the new expiration date of your renewed certificate:  
11 Step 10. Retrieve certificate expiration

Define customer response

And then

End the action

Preview

## Run the complete custom-built action

The custom-built action is now complete and can be demonstrated to the SA for this use case. In demonstrating the ability to infuse Ansible automations into a natural conversation, the SA is able to see the value that Watsonx Assistant for Z can provide in helping them improve productivity and remove the need to go to their senior colleagues for assistance.

1. Open **Preview** in the **Ai assistant builder**.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with options like Home, Build, Generative AI, Actions (which is selected and highlighted with a green border), Evaluate, Deploy, Publish, Environments, Improve, and Analyze. The main area is titled 'Created by you /' and contains a table of actions. The columns are Name, Last edited, Examples count, Steps count, and Status. The table lists several actions, each with a small icon and some descriptive text.

Name	Last edited	Examples count	Steps count	Status
z/OS IPL Information	a day ago	0	3	<span>Green</span>
Zeeves-gather-facts-flow	3 days ago	1	0	<span>Green</span>
Retrieve IPL information	a day ago	1	3	<span>Green</span>
Retrieve certificate expiration	4 hours ago	0	3	<span>Green</span>
Cert Renewal skill flow	4 hours ago	0	7	<span>Green</span>
z/OS certificate expires soon	a few seconds ago	1	11	<span>Green</span>

2. Enter the following text in the assistant.

**Prompt:**

How do I check the expiration date for my certificate that's expiring soon?



Use the Change layout option to open a full page view of the assistant.

The screenshot shows the AI assistant builder in full-page view. At the top, there are tabs for 'Preview assistant' (selected), 'AI assistant builder', and 'Zeeves'. Below the tabs are buttons for 'Copy link to share', 'Change layout' (which is highlighted with a green border), 'Change background', and 'Customize web chat'. The main area is titled 'AssistantZeeves' and features a large blue owl icon. A message from the AI assistant says, 'Hi! I'm Zeeves. How can I help you today?'. At the bottom, there's a text input field containing the prompt 'How do I check the expiration date for my certificate that's expiring soon?' with a green border around it.

3. Review the response and click Yes.

The assistant responds with conversational search, providing a content-grounded answer based on IBM Z documentation. The response includes a RACF command that the SA might use to determine their certificate's expiration date.

Following the response, the assistant prompts the user if they want to run the skill to retrieve a certificate's expiration date.

The screenshot shows the IBM Watsonx Orchestrate interface with the 'AI assistant builder' tab selected. A sidebar on the left contains icons for preview, AI assistant, Zeeves, and other tools. The main area is titled 'Preview assistant' and shows a conversation with 'AssistantZeeves'. The user asks, 'How do I check the expiration date for my certificate that's expiring soon?'. The AI assistant responds with a detailed answer about checking certificates and provides a RACF command: 'ODTE' or 'RACDCERT'. Below this, the AI asks if the user wants to run a skill to retrieve the certificate's expiration date, with 'Yes' and 'No' buttons. A note at the bottom says, 'Accuracy of generated answers may vary.' The bottom right of the interface has a text input field with placeholder text 'Type something...'. The bottom left corner shows the text 'Built with IBM Watsonx'.

4. Enter DEMOCERT after the assistant responds with **What is your certificate label?**

**Prompt:**

DEMOCERT

The screenshot shows the IBM Watson Assistant interface. At the top, there are tabs for 'IBM Watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. Below the tabs, a header bar includes 'Preview assistant', 'Copy link to share', 'Change layout', 'Change background', and 'Customize web chat'. The main area displays a conversation between the user and the AI assistant:

You 12:37 PM  
How do I check the expiration date for my certificate that's expiring soon?

Assistant 12:37 PM  
To retrieve the expiration date of your z/OS certificate, you would typically check the certificate itself or consult the system logs. However, based on the provided documents, there isn't a direct command or utility mentioned for this purpose. You may need to contact your system administrator or refer to your specific system's documentation for precise instructions. Generally, you can use commands like 'ODTE' or 'RACDCERT' to list and inspect certificates.

A Accuracy of generated answers may vary.

You 12:37 PM  
Would you like to run the skill to retrieve your certificate's expiration date?

Assistant 12:37 PM  
Yes

You 12:40 PM  
Yes

Assistant 12:40 PM  
What is your certificate label?

The input field at the bottom contains the text 'DEMOCERT'.

5. Wait 10 seconds and then click **Apply**.

The screenshot shows the IBM Watsonx Orchestrate interface. At the top, there are tabs for 'IBM Watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. On the right side of the header are buttons for 'Copy link to share', 'Change layout', 'Change background', and 'Customize web chat'. Below the header, the main area is titled 'Preview assistant' and shows a conversation with 'AssistantZeeves'. The conversation starts with a user message: 'How do I check the expiration date for my certificate that's expiring soon?'. The AI responds with: 'To retrieve the expiration date of your z/OS certificate, you would typically check the certificate itself or consult the system logs. However, based on the provided documents, there isn't a direct command or utility mentioned for this purpose. You may need to contact your system administrator or refer to your specific system's documentation for precise instructions. Generally, you can use commands like 'ODTE' or 'RACDCERT' to list and inspect certificates.' A note below says: 'AI Accuracy of generated answers may vary.' The user then asks: 'Would you like to run the skill to retrieve your certificate's expiration date?'. The AI replies with two options: 'Yes' and 'No'. The user selects 'Yes'. The AI then asks: 'What is your certificate label?'. The user responds with 'DEMOCERT'. Finally, the AI displays a modal window titled 'Retrieve certificate expiration' with an 'id \*' field containing '21'. There are 'Cancel' and 'Apply' buttons at the bottom, with 'Apply' being highlighted.

#### 6. Review the response and then click Yes.

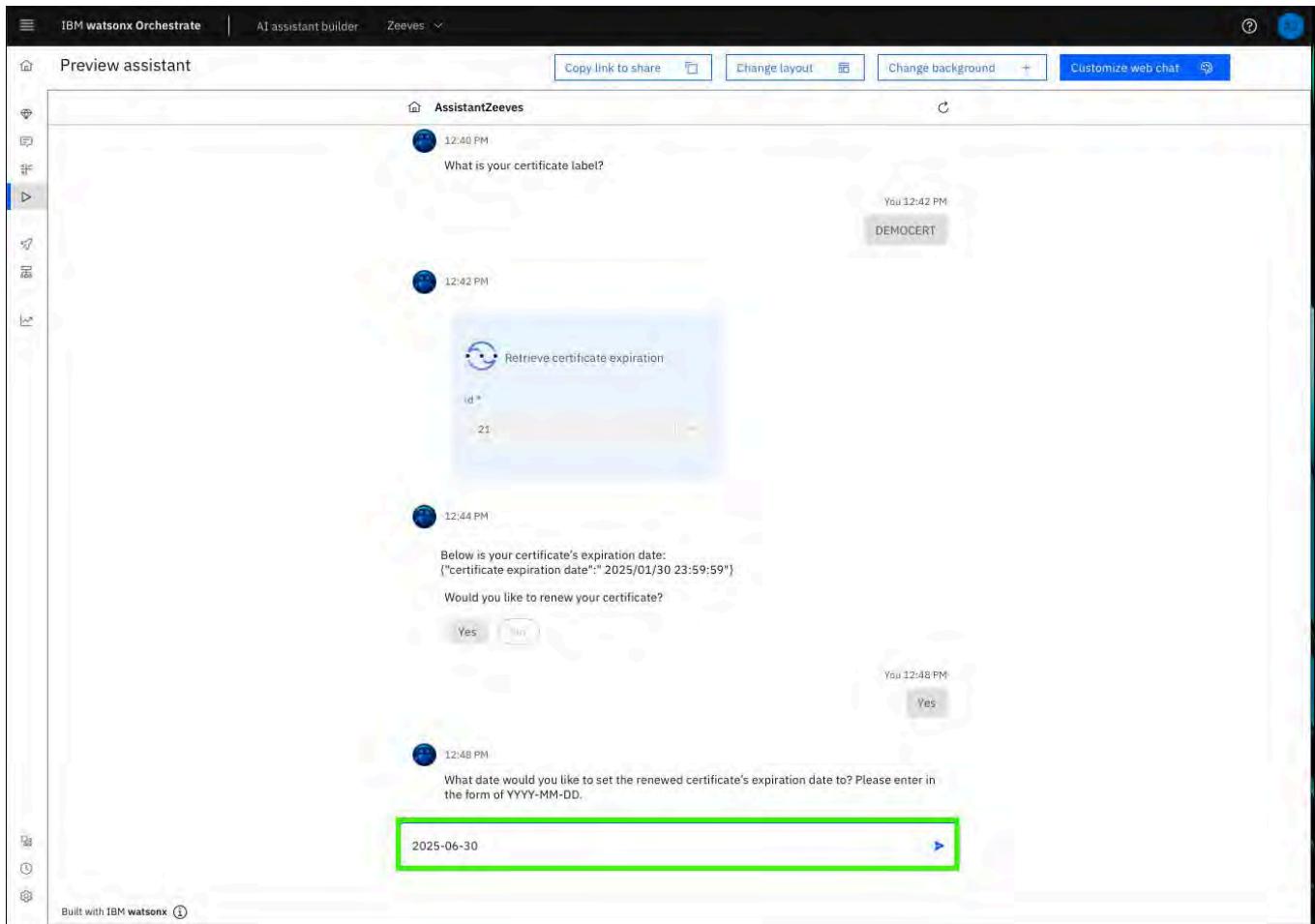
By providing the automation within the assistant conversation, it makes it very quick for the SA to identify the certificate's expiration date. In addition to providing this valuable information, the assistant is configured with another automation to renew the certificate if they choose to do so.



**The expiration date you see may differ from the image that follows.**

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and various configuration buttons like 'Copy link to share', 'Change layout', 'Change background', and 'Customize web chat'. The main area is titled 'Preview assistant' and shows a conversation with 'AssistantZeeves'. The conversation starts with a message from the user asking if they want to run a skill to retrieve the certificate's expiration date. The AI responds with 'Yes' and asks for the certificate label. The user replies with 'DEMOCERT'. The AI then asks for an ID, which is entered as '26'. Finally, the AI provides the certificate's expiration date: '2025/01/30 23:59:59'. A green dashed box highlights this response. The user is then asked if they would like to renew the certificate, with 'Yes' being the highlighted option.

7. Enter a date in the future in the format **YYYY-MM-DD**.

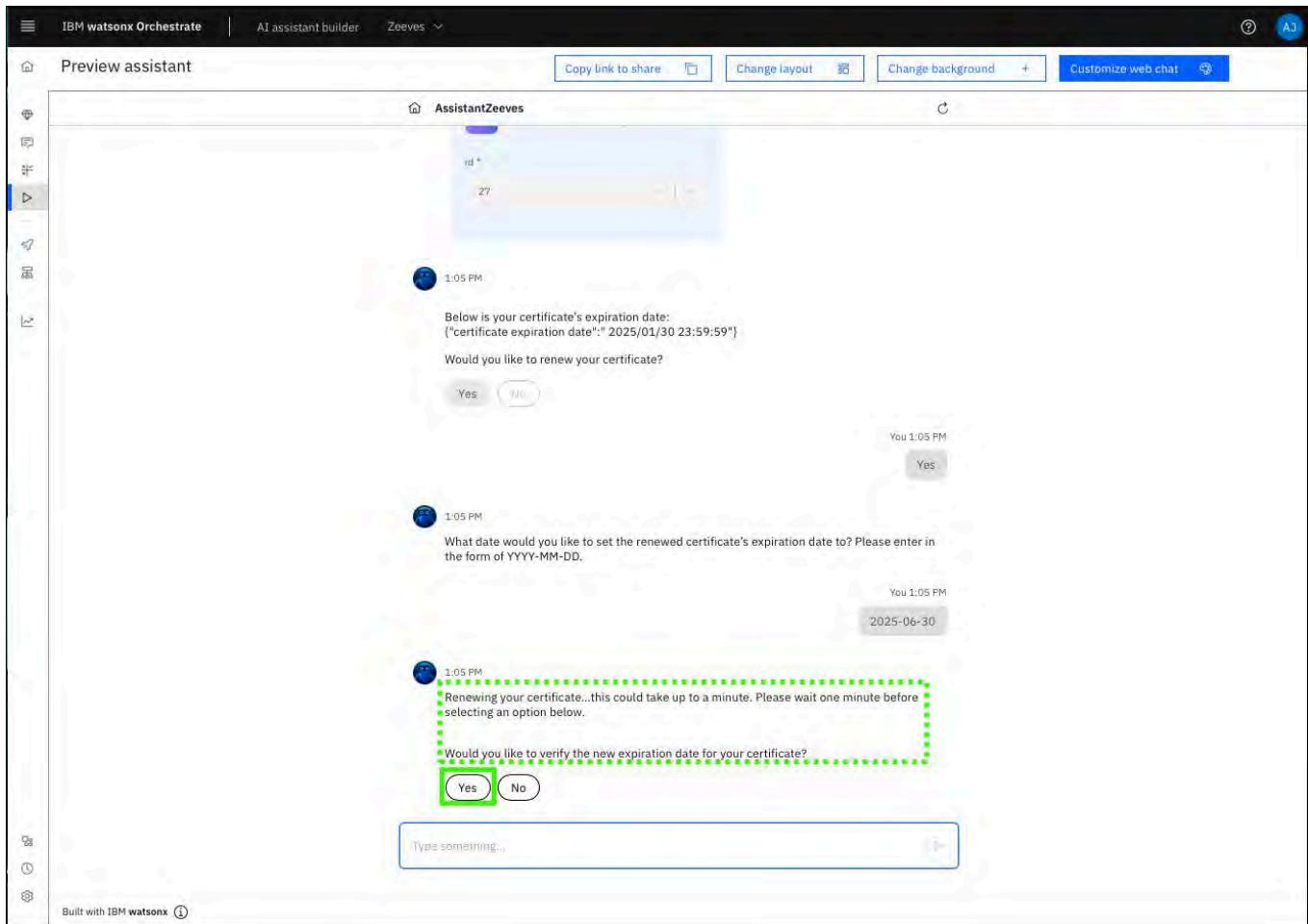


8. Review the response, wait 30 seconds to a minute, and then click **Yes**.



**It is crucial that you wait 30 seconds to a minute before selecting Yes.**

This is because in the background, your z/OS Certs – Search and Renew automation is running within AAP (which you can verify within the [AAP Web console](#)). This is mapping the user-inputted expiration date as well as the original certificate label provided by the end-user to the inputs of this AAP automation."



9. Wait 10 seconds and then click **Apply**.

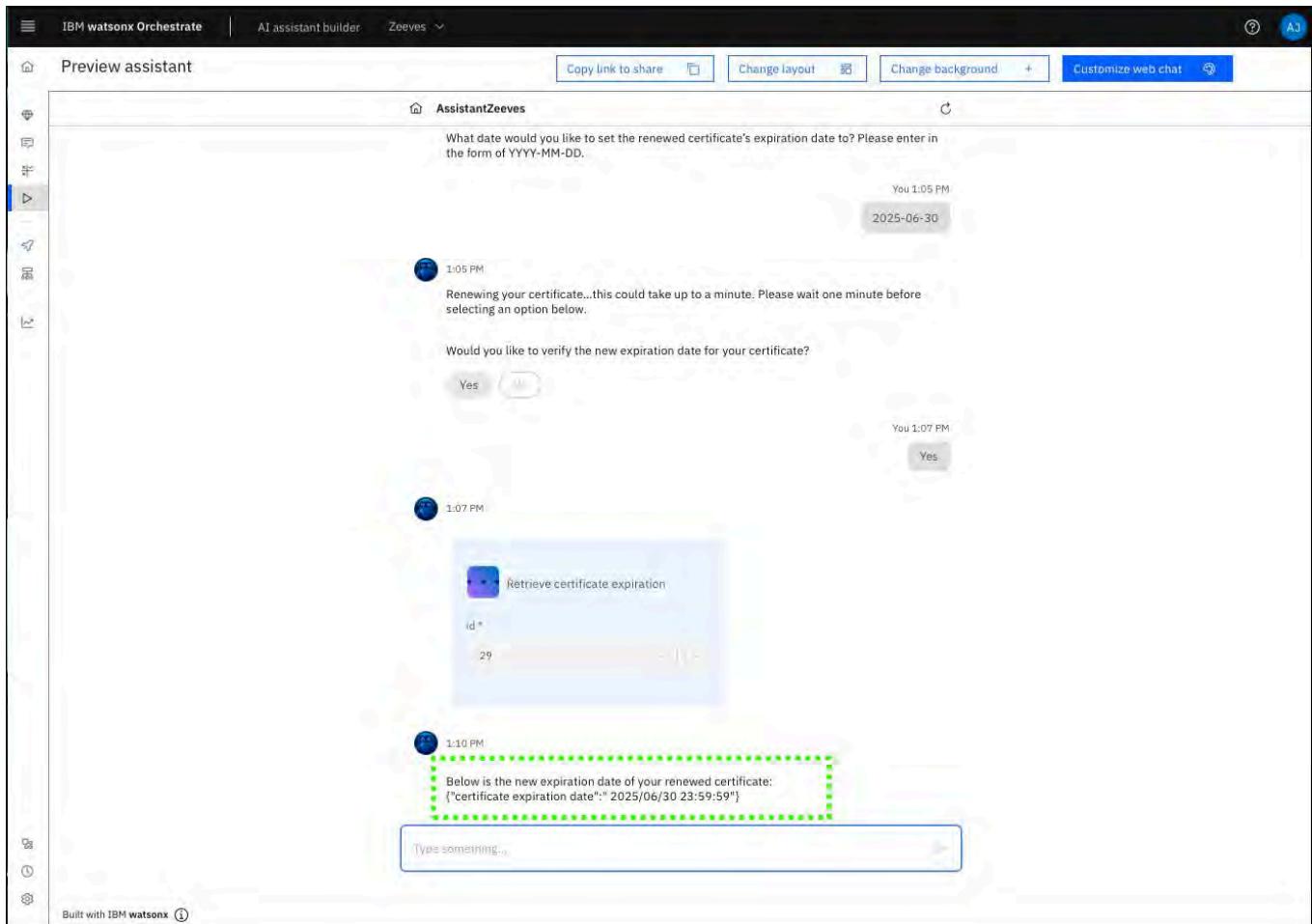
The screenshot shows the 'Preview assistant' window in the IBM Watsonx Orchestrate interface. The window title is 'AssistantZeeves'. The conversation log is as follows:

- You 1:05 PM: What date would you like to set the renewed certificate's expiration date to? Please enter in the form of YYYY-MM-DD.
- You 1:05 PM: 2025-06-30
- You 1:05 PM: Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.
- You 1:05 PM: Would you like to verify the new expiration date for your certificate?
- You 1:07 PM: Yes

At the bottom, there is a modal dialog titled 'Retrieve certificate expiration' with a green dashed border. It contains a single input field labeled 'id \*' with the value '29'. At the bottom of the dialog are two buttons: 'Cancel' and 'Apply', with 'Apply' highlighted in blue.

#### 10. Review the response.

The response should match the date that is entered in step 7.



In the demo, the SA receives immediate guidance on identifying the certificate expiration date via RACF commands. The SA runs automation that is proposed by the assistant to retrieve the certificate information. Also, because the assistant is configured with step-by-step conversation flows, it is possible to add other prompts within the conversation. For example, proposing the automation of renewing the certificate on their behalf. By doing so, the SA is able to reduce the time it takes to complete this routine task.

Recall how many steps were involved in the Ansible template for **z/OS Certs – Search and Renew**. By automating these tasks with Ansible, the System Administrator streamlines the entire process and ensures that their critical certificates are up to date and reduce the risk of expired certificates disrupting their business services.

# Use case: RACF administration support



This is a **challenge use case**.

In this use case, step-by-step directions are not provided. Use the knowledge gained from previous sections of the lab guide to complete updates to the assistant to meet the use case requirements. Several help tips are provided and a sample demonstrate flow is included.

This use case explores the ability of watsonx Assistant for Z to provide self-service support for activities related to Resource Access Control Facility RACF (RACF) Administration support. The client mentioned that their RACF Administrators are often inundated with requests to help reset user's Time Sharing Option (TSO) passwords. Also, requests to grant users certain authorizations to RACF profiles required for the users to complete their jobs are numerous.

Creating a custom-built action for the assistant that provides self-service options to users requesting RACF assistance. Depending on the assistance being requested, the assistant will trigger automations that complete tasks on their behalf without intervention from the RACF Administrator.



**The actions and configuration that follow are meant purely for demonstration purposes.**

The actions and configuration that follow are meant purely for demonstration purposes to show the art of the possible with watsonx Assistant for Z. There are security considerations to keep in mind when configuring assistants for this purpose, including user authentication and the ability to authorize users to execute certain automations. However, by following the steps in this section, you will show the level of customization that is possible with watsonx Assistant for Z and different an assistant can improve productivity.

1. Create an app with the required TSO skills.



**Use Skills studio to create the app.**

**App name:**

TSO Command

Skills to include: z/OS TSO CCommand(s) Retrieve job output (utility skill)

Welcome to Skill studio

Optimize productivity by using projects to build and manage skill flows and connectors.

**Create**

**Skills and apps** Projects

**Skills** Apps

Find a skill

**Skills**

Name	Step in the process	Status	Skill type	Author	Last edited
Retrieve job output	Ready to use	Published	Imported	andrew@jones-tx.com	January 20 2025
z/OS TSO Command(s)	Ready to use	Published	Imported	andrew@jones-tx.com	January 20 2025

Configure prebuilt skills

## 2. Create a connection for the app.

Use Skills catalog to create the app.

Ansible Controller Skills - TSO Command (2)

Connected

Personal skills

Search skills

Ansible Controller Skills - TSO Command

Retrieve job output TSO Command - Retrieve job output by job Id	z/OS TSO Command(s) TSO Command - A playbook to execute one or multiple TSO commands on the target z/OS system.
--	--

Added ✓

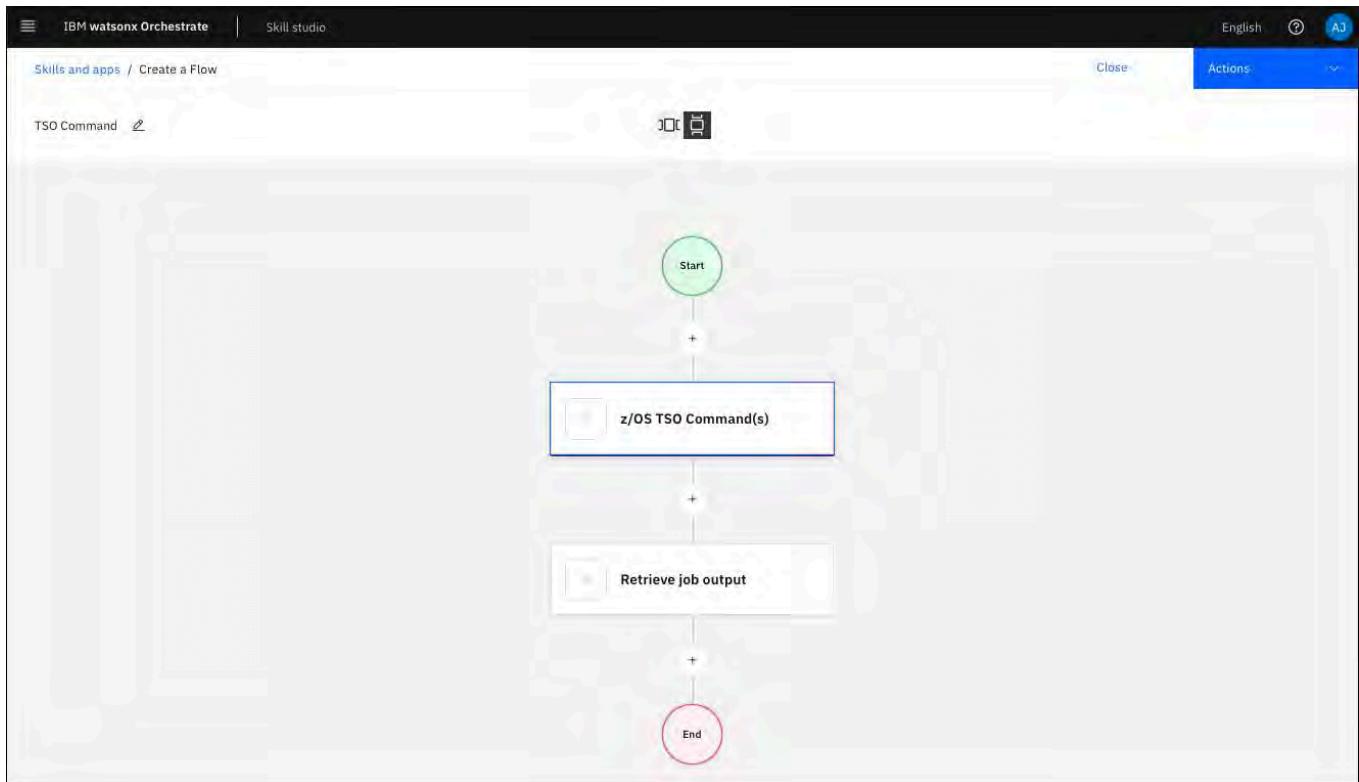
## 3. Create a skill flow to get the output of the **TSO Command**.

Use Skills studio to create the skill flow.

The skill flow should run the **TSO Command** skill and then the **Retrieve job output** utility skill. The output from the **TSO Command** should be mapped to the input of the **Retrieve job output** utility skill. Hide both the input and output forms for the **TSO Command** skill.

**Skill flow name:**

TSO COMMAND



#### 4. Add the skills to the **draft** version of the assistant.



**Use Skill sets to add the skills to the assistant.**

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - TSO Command	2	Team	andrew@jones-tx.com	⋮

#### 5. Create a skill-based action that uses the skill flow.

Create an action that triggers the skill flow to execute TSO commands on z/OS and then display the output of those commands.



**Use AI Assistant builder to create the action.**



**Do not add example phrases to the skill-based action.**

Name	Last edited	Examples count	Steps count	Status
X TSO Command	a few seconds ago	0	1	Green checkmark
X z/OS certificate expires soon	3 hours ago	1	11	Green checkmark
X Cert Renewal skill flow	7 hours ago	0	7	Green checkmark
X Retrieve certificate expiration	7 hours ago	0	3	Green checkmark
X Retrieve IPL information	a day ago	1	3	Green checkmark
X z/OS IPL Information	a day ago	0	3	Green checkmark
X Zeeves-gather-facts-flow	3 days ago	1	0	Green checkmark

## 6. Create a custom-built action for RACF administration support.



**Use AI Assistant builder to create the action.**

**Phrase to start the interaction:**

RACF assistance

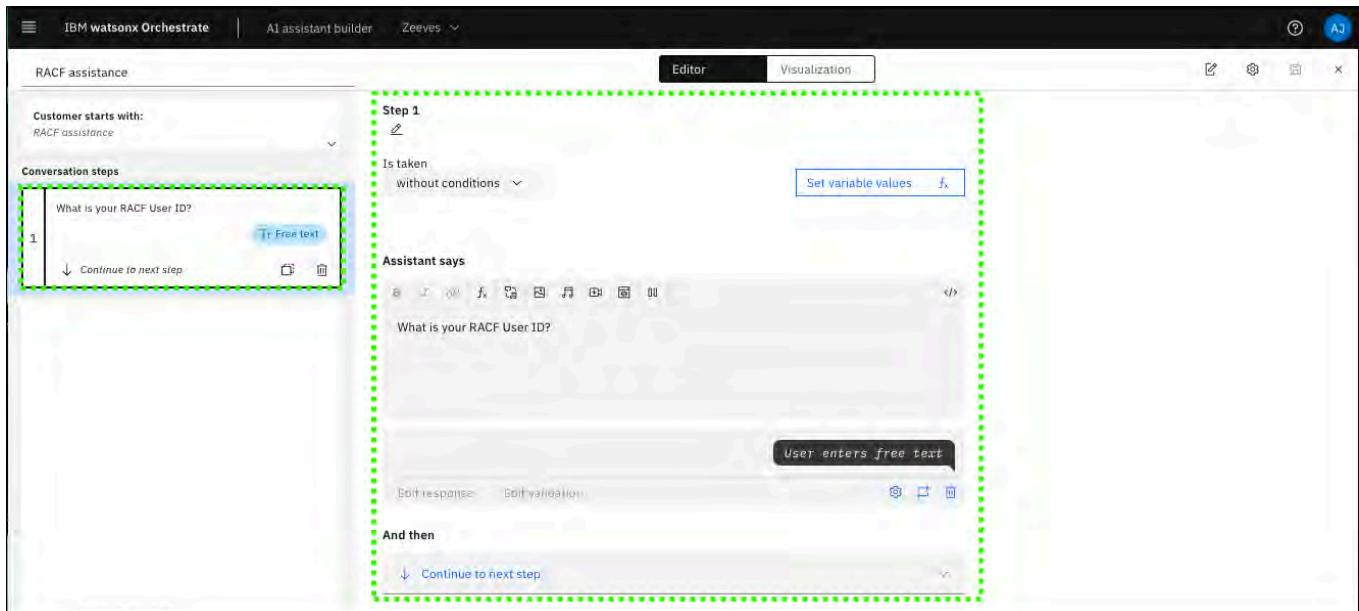
The steps for the custom-built action follow:

**Step 1.** Without conditions, prompt the user for their RACF User ID as free text.

**Assistant says:**

What is your RACF User ID?

The step should look like the following image:



**Use the following guidance when entering prompts and expressions in the following custom-built action steps.**

Do NOT use the copy and paste icon if the string contains a `$`. Type each string manually.

When `$` appears in the string to enter, type the `$` character and then select the variable specified in the `<>`. For example: User passphrase changed to `$<7`. Please enter your new RACF passphrase>

Use the single quote character ' for all single quotes shown.

**Step 2.** *Without conditions*, use provided user ID and display a message that the assistant is checking the current privileges.

**Assistant says:**

```
Checking user privileges for $<1. What is your RACF User ID>
```

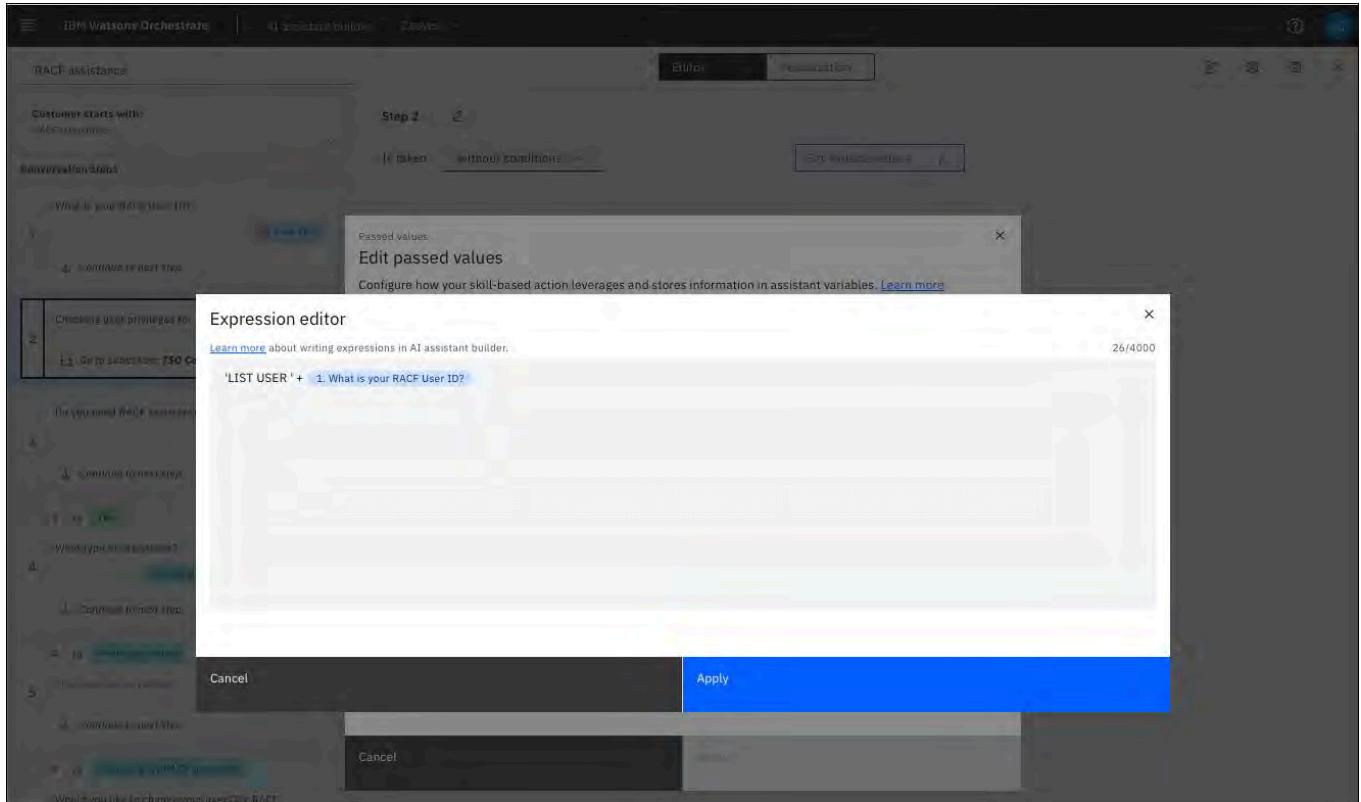
Include in the **Assistant says** the provided user ID by including the function `1. What is your RACF User ID`.

Add an **And then** option to run the **TSO Command** as a subaction.

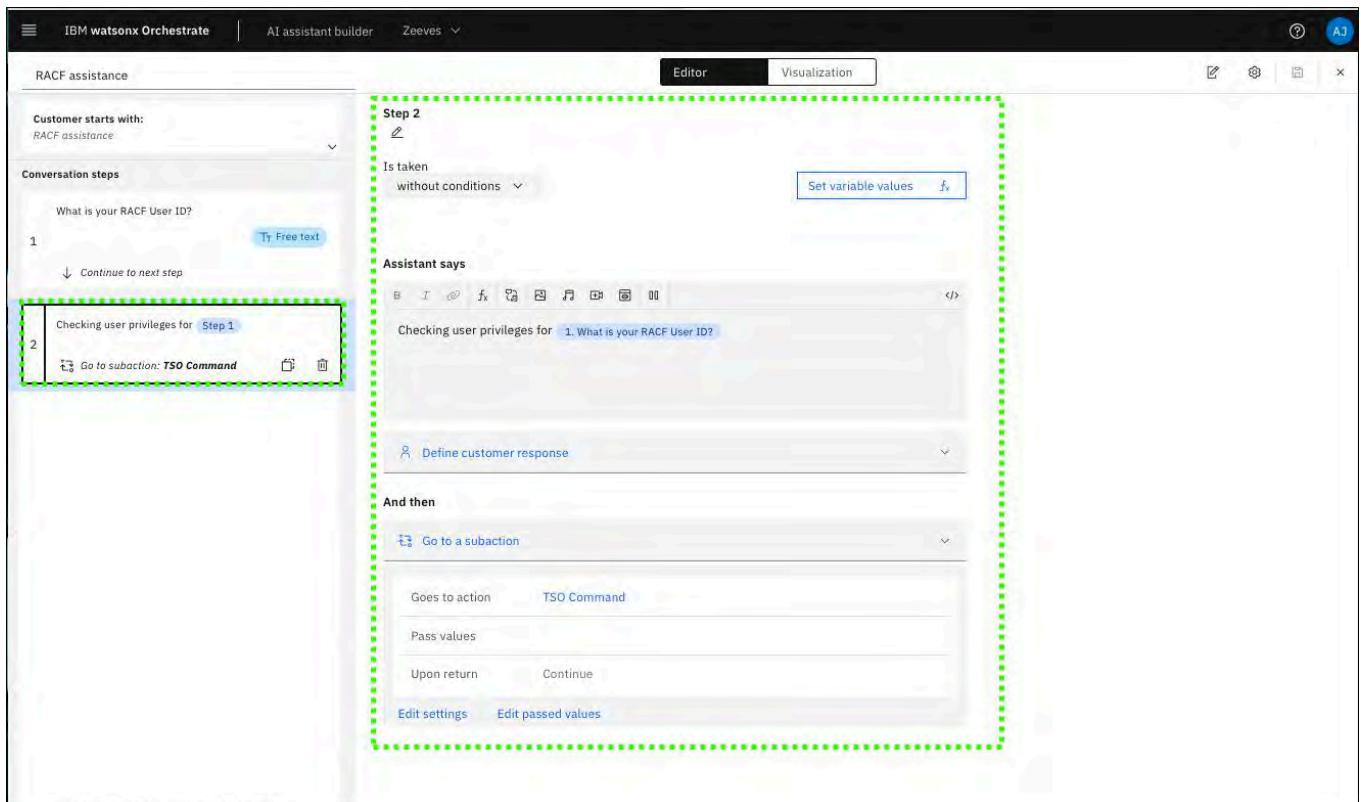
Edit the **passed values** for the subaction and add a new passed value for `1. extra_vars.zos_tso_command`. The **To** of the variable will be an **Expression** and include the string:

```
'LIST USER ' + $<1. What is your RACF User ID>
```

After the `+`, enter a `$` and select **Action step variables** and then select `1. What is your RACF User ID`.



The step should look like the following image:

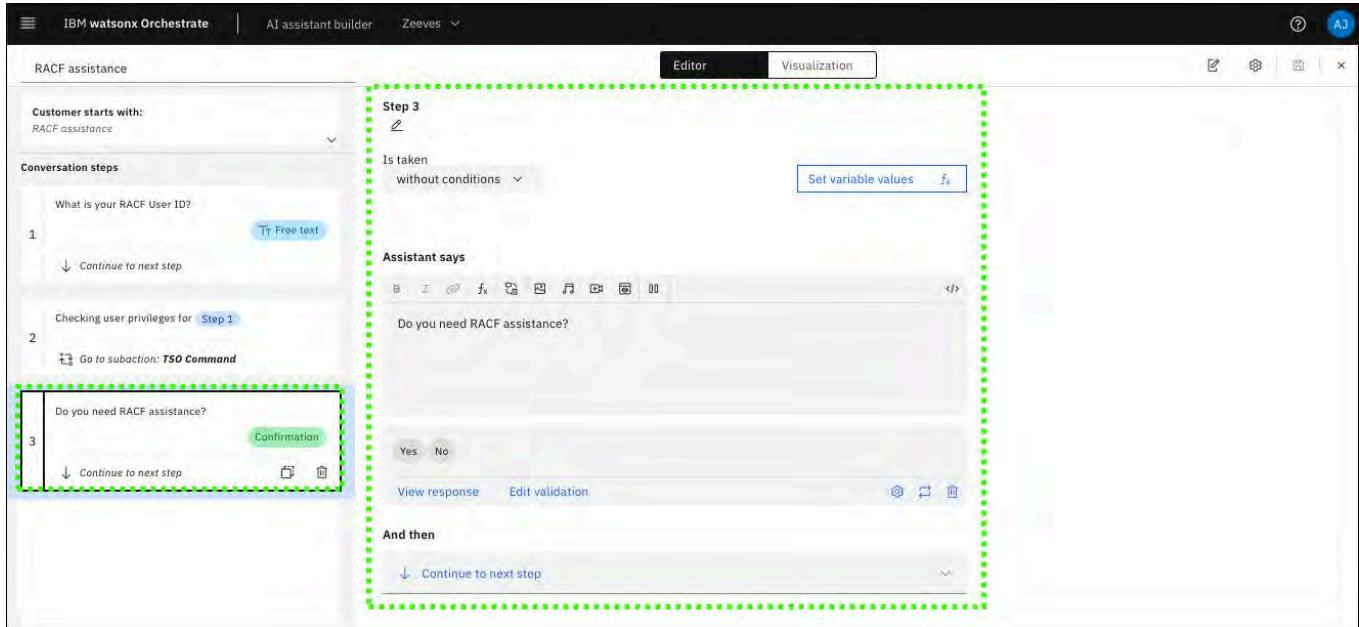


**Step 3.** Without conditions, prompt the user if they want RACF assistance and get a Yes or No confirmation.

**Assistant says:**

Do you need RACF assistance?

The step should look like the following image:



**Step 4.** With conditions, prompt the user what type of assistance they need.

**Assistant says:**

What type of assistance?

Create a **Options** type **Customer response** with two options:

Changing my RACF password

and

Privileges issue

Change the **Customer response** settings to **Always ask for this information, regardless of previous messages**.

The screenshot shows the IBM Watsonx Assistant Editor interface. On the left, there's a sidebar titled "Conversation steps" with four numbered steps:

- What is your RACF User ID? (Free text)
- Checking user privileges for Step 1 (TSO Command)
- Do you need RACF assistance? (Confirmation)
- What type of assistance? (Buttons: Changing my ..., Privileges issue...)

Step 4 is highlighted with a green dashed border. The "Editor" tab is selected at the top. In the main area, under "Step 4", it says "Is taken with conditions". A condition is defined: "If 3. Do you need RACF ... is Yes". Below this, there's a "Conditions" section with a "New condition group" button. The "Assistant says" section contains a message: "What type of assistance?". At the bottom, there are "Edit response" and "Edit validation" buttons.

The step should look like the following image:

This screenshot is identical to the one above, but the entire content of Step 4 is now highlighted with a thick green dashed border. This visual cue indicates that the user should focus on modifying this specific part of the conversation logic.

**Step 5. With conditions, change the condition Step 4. What type of assistance? and the value of Privilege issue.**

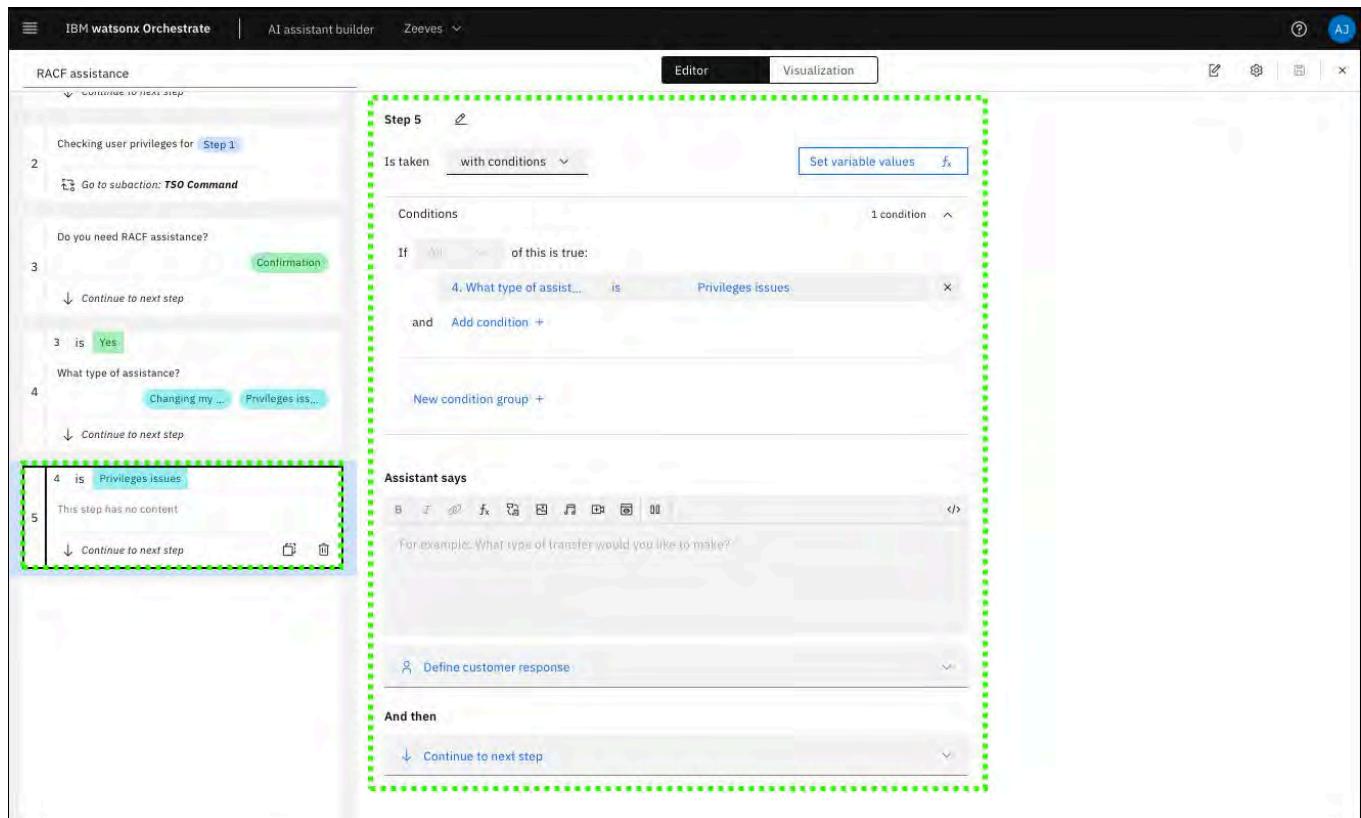
**Assistant says:**

Notifying the RACF Administrator.

 **Other options exist.**

There are alternative actions that can be taken if the user selects 'Privileges issue'. For example, the assistant can trigger an automated email to the RACF administrator. At the time of writing this documentation, Orchestrate does not allow Outlook integrations to IBM's Outlook organization, so this action cannot be demonstrated. But for the purpose of the flow, simply have the assistant respond with 'Notifying the RACF Administrator' for demonstration purposes.

The step should look like the following image:



**Step 6. With conditions, change the condition Step 4. What type of assistance? and the value of Changing my RACF password, and add a Yes or No confirmation.**

**Assistant says:**

Would you like to change your user ID's RACF passphrase?

The step should look like the following image:

**Step 6**

Is taken with conditions

Conditions

If 4. What type of assistance? is Changing my RACF password

New condition group +

Assistant says

Would you like to change your user ID's RACF passphrase?

Yes No

View response Edit validation

And then

↓ Continue to next step

**Step 7. With conditions, prompt the user to enter their new password as free text.**

**Assistant says:**

Please enter your new RACF passphrase.

The step should look like the following image:

**Step 8. With conditions**, change the condition to **Step 6. Would you like to change your user ID's RACF passphrase?** equals `Yes`, inform the user the password is being changed, and create a **TSO Command** subaction to change the passphrase.

**Assistant says:**

Issuing RACF command ...



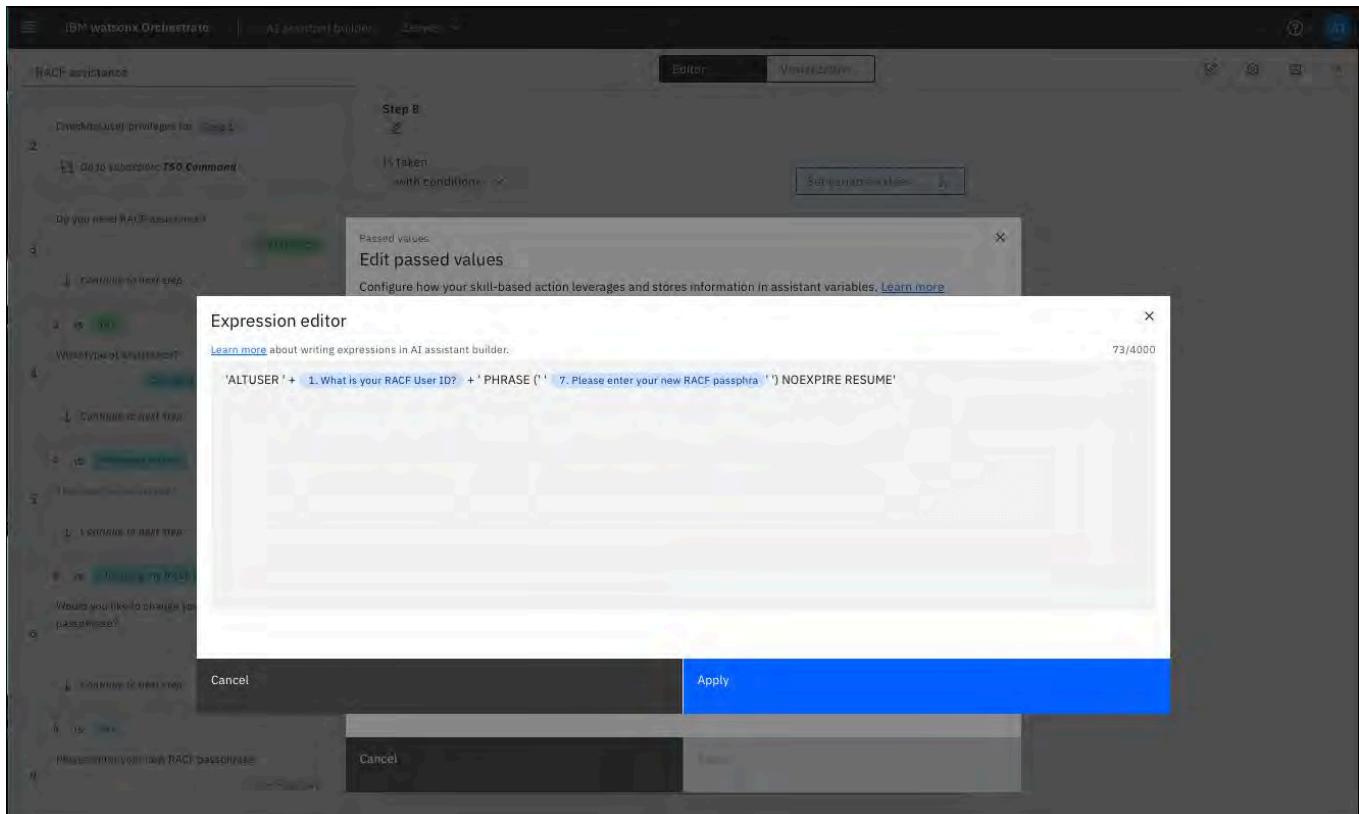
The formatting of the values that follow is important.

When creating the **To** expression for the passed values, be very careful with typing the expression. Do NOT cut & paste this value. Type each character. All quotes are single-quotes. The `$<>` denotes typing `$` and then selecting the appropriate **action step variable**.

Edit the passed values for the **TSO Command** subaction to include the **1. extra\_vars.zos\_tso\_command** variable with a **To** expression with the value of:

```
'ALTUSER ' + $<1. What is your RACF User ID?> + ' PHRASE('''$<7. Please enter your new RACF
passphrase>'') NOEXPIRE RESUME'
```

The expression should look like the following image:



The step should look like the following image:

**Step 8**

Is taken  with conditions  with conditions

Conditions

If All of this is true:

- 6. Would you like to c... is Yes
- and Add condition +

New condition group +

**Assistant says**

Issuing RACF command ...

**Define customer response**

**And then**

Go to subaction: TSO Command

Goes to action: TSO Command

Pass values: 1. extra\_vars.zos\_tso\_command

Upon return: Continue

**Step 9.: With conditions, change the condition to Step 6. Would you like to change your user IDS passphrase equals Yes , and inform the user the passphrase has been changed. Change the And then option to End the action.**

**Assistant says:**

User passphrase changed to \$<7. Please enter your new RACF passphrase.>

When entering the above string, after typing \$ select the **7. Please enter your new RACF passphrase**.

The step should look like the following image:

The screenshot shows the AI Assistant builder interface for creating a custom-built action. The main pane displays a sequence of steps (3 through 9) and their associated responses. Step 9 is highlighted with a green dashed box. In the 'Conditions' section of Step 9, a condition is defined: "If 6. Would you like to c... is Yes". The 'Assistant says' section contains the message "User passphrase changed to 7. Please enter your new RACF passphrase". The 'And then' section is set to "End the action".

7. Be sure to save your custom-built action.

8. Demonstrate the custom-built action.

Using the **AI Assistant builder** preview, run the custom-built action. Use the APP web console to verify the passphrase was changed.

The following video shows how the demonstration should work. The video does not have audio.

## 404 - Not found

This use case demonstrates the value watsonx Assistant for Z can provide to offload common, manual tasks from subject matter experts like RACF Administrators. The use case shows the level of customization offered with infusing automations into natural conversations. Watsonx Assistant for Z improves employee productivity and reduces effort needed by individuals to completing manual tasks.

# Adding other integrations

IBM watsonx Assistant for Z can integrate with other delivery channels beyond a web page. Other channels include Slack, Microsoft teams, WhatsApp, and many others. Integrating with these and other channels are not covered in the lab guide. However, follow the steps to find the current channels that are supported and where to get more information.

1. Hover over the **Home** ( ) and click **Integrations**.

The screenshot shows the IBM Watsonx Orchestrate interface. The left sidebar has a tree view with 'Home' (highlighted with a green box) expanded, showing 'Generative AI', 'Actions', 'Evaluate', 'Preview', 'Publish', 'Environments', 'Analyze', and 'Integrations' (also highlighted with a green box). The main content area displays several cards: 'Build actions' (enhance and improve your assistant's actions), 'Customize your greeting' (welcome your users in a unique way that aligns with your brand), 'Create a fallback plan' (train your assistant to adapt to specific situations), 'Preview & debug' (enable debug mode when you preview your assistant), 'Customize web chat' (design how your assistant shows up in your live environment), 'Set up a channel' (select the channels you want to deploy your assistant to), 'Set up live agent' (integrate live agents to transfer your users to when you edit web chat), and 'Check' (analyze performance reviews specific to your assistant). Below these cards is a diagram titled 'Architecture' showing the flow from 'Default behavior' through 'Zeeves' to 'Actions', 'Fallback', and finally 'Live agent'. A note says 'Structure depicts the structure of your assistant'. At the bottom left, there are links for 'Activity log' and 'Assistant settings'.

2. Explore the **Essential channels** and **Channels** sections.

The screenshot shows the 'Integrations' page. The left sidebar has a tree view with 'Integrations' (highlighted with a green box) expanded, showing 'Essential channels' and 'Channels'. The main content area has a large 'Web chat' card with a phone icon. Below it is a section titled 'Web chat' with a sub-section 'Get started'. Further down are 'Connect center integrations' with icons for H2C, Twilio, and others, and a 'Open' button. At the bottom, there are sections for 'Channels' (with icons for SMS, Facebook messenger, Genesys Bot Connector, Slack, Microsoft teams, and WhatsApp with Twilio) and 'Extensions' (with icons for Google Assistant, Amazon Alexa, and others).

3. Click **Add** on the **Slack** tile.

The screenshot shows the 'Integrations' section of the IBM Watsonx Orchestrate interface. It includes sections for 'Essential channels' (Web chat, Slack, Microsoft Teams, WhatsApp with Twilio), 'Channels' (SMS, Facebook messenger, Genesys Bot Connector, Slack, Microsoft teams, WhatsApp with Twilio), and 'Extensions'. A green box highlights the 'Add' button next to the Slack icon in the Channels list.

#### 4. Click Add.

The screenshot shows the Slack integration setup dialog. It includes fields for 'Identity' (Set your assistant respond identity in Slack. All your assistant is listed in your Slack app, answering questions from direct messages or channels.) and 'Add'. A green box highlights the 'Add' button.

#### 5. Review the step-by-step instructions and additional information available for adding a Slack integration.

**Note:** Most regular users do not have permissions to integrate with your enterprise slack deployment as doing so requires administrative rights.

The screenshot shows the 'Get started' steps for setting up Slack. It includes 'Get started' (Through Slack, your assistant is ready to join the collaboration hub that brings the right people, information, and tools together to get work done. Learn more), 'Slack bot' (A bot you can add to your Slack workspace to interact with your AI assistant.), 'Connect Slack' (Connect your AI assistant to your Slack workspace to receive messages from your Slack channel.), 'Steps to setting up Slack' (1. Set up your Slack bot, 2. Connect AI assistant builder to Slack, 3. Configure your Slack bot, 4. Connect your assistant), and 'Close' and 'Next' buttons. A green box highlights the 'Next' button.

Take time to further explore the next steps for adding a Slack integration channel and the other supported integration channels.

Learn more about adding integrations [here](#).

# Next steps

This lab guide covered many of the IBM Watsonx Assistant for Z capabilities and provides a good base for conducting basic client pilots. However, there is still more to learn about IBM Watsonx Assistant for Z. Updates and new releases of the offering and the underlying offerings like Watsonx Orchestrate rollout regularly. Be sure to stay informed by bookmarking and regularly reviewing the [offering landing page](#) and the [product documentation](#).

## Other resources

The following resources are also available; however, not all are available to Business Partners:

Slack channel: [#watsonx-assistant-z-technical](#)

Wiki: [Development team's wiki](#)

Box: [wxa4z Q&A - questions with high-quality responses.](#)

## Earn the badge

Finally, remember to earn the IBM Watsonx Assistant for Z Technical Sales Advanced you must complete the IBM Watsonx Assistant for Z for Technical Sales Level 4 learning plan.

**IBM technical sellers:** [Your Learning learning plan](#)

**Business Partners** [IBM Training learning plan](#)