

# 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

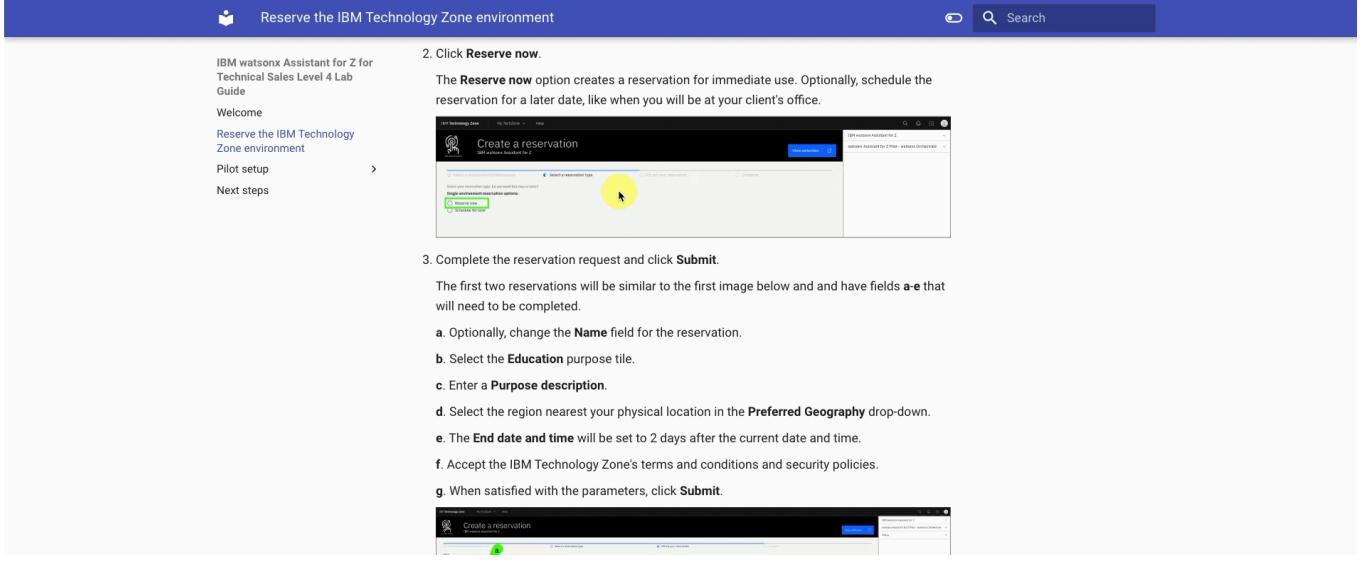
inetegrationID:

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.

- Optionally, 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 X 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 course name, a refresh icon, and a search bar. On the left, a green dashed box highlights the Table of Contents sidebar. The sidebar lists chapters: "Welcome", "Guidance for delivering a demonstration", "Reserve the IBM Technology Zone environment", "Demonstration scenarios", "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 introduces the course goal of providing knowledge and tools for demonstrating IBM watsonx Assistant for Z. It notes that the course focuses on end-user querying and automation skills, while other capabilities like Velocity Pilot are covered in the IBM Technology Zone (ITZ). A section at the bottom encourages obtaining the IBM watsonx Assistant for Z Technical Sales Intermediate badge.

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, causing the sidebar to expand. The sidebar now fully displays the list of chapters from the previous screenshot. The main content area remains the "Welcome" page, identical to the previous view.

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.

The screenshot shows the 'Create a reservation' page in the IBM Technology Zone. At the top, there are tabs for 'IBM Technology Zone', 'My TechZone', and 'Help'. On the right, there are search and filter icons. The main title is 'Create a reservation' with the subtitle 'IBM watsonx Assistant for Z'. Below the title, there are four steps: 'Select a environment/infrastructure', 'Select a reservation type', 'Fill out your reservation', and 'Complete'. Under 'Single environment reservation options:', the 'Reserve now' radio button is selected, while 'Schedule for later' is unselected. A 'View collection' button is located in the top right corner of the main form area.

## 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.

- a.** Name: specify a name for the reservation (optional).
- b.** 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.

- c.** Purpose description: enter a description, for example: Level 4 education.
- d.** Preferred geography: select the region nearest to your physical location for improved performance and reduced network latency.
- e.** 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.
- f.** Accept the IBM Technology Zone's terms and conditions and security policies.
- g.** Click **Submit**.

**Create a reservation**

IBM Watsonx Assistant for Z

Name  a

Select a reservation type  b

Fill out your reservation c

Complete

Purpose

Demo  Deliver a client specific demonstration based on discovery with the client and align to the identified architecture. Automatically captures a Technical Sales Activity in IBM Sales Cloud on the Opportunity code provided.

Pilot  Run a creation build that proves IBM technologies can deliver business value to clients' end users. Serves as a foundation to build a production solution. Automatically captures a Technical Sales Activity in IBM Sales Cloud on the Opportunity code provided.

Please ensure to select the correct purpose as this can **NOT** be updated or changed after this reservation has been created. Review the [Reservation Duration Policy](#) to understand default durations allowed for specific infrastructures based on purpose.

Sales Opportunity number  d

Provide an [IBM Sales Cloud opportunity ID](#), [Watsonx ID](#), or a [Project Work ID](#) will allow you to extend your reservation date.

Purpose description  e

What are you doing? Why do you need this? What are you trying to accomplish?

Preferred Geography  f

End date and time g

Select a date

Select a time

America/Chicago

Reservations policy: Recommended 3 days, but can be reserved up to 2 days on this reservation form. Extend later for 2 days increments up to 4 days total. Max time & days total.

Notes  h

I agree to IBM Technology Zone's [Terms & Conditions](#) and [End User Security Policies](#)  i

**Submit** j

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

Enter date and time | Select a time | Select a time | America/Chicago |

Reservation policy: Recommended 2 days, but can be reserved up to 2 days on this reservation form. Extend later for 2 days increments up to 4 days total. Max time 6 days total.

OCP/Kubernetes Cluster Network: 10.128.0.0/14 (h)

Enable FIPS Security: No (i)

Master Single Node Flavor: 16 vCPU x 64 GB - 300 GB ephemeral storage (j)

OpenShift Version: 4.14 (k)

OCP/Kubernetes Service Network: 172.30.0.0/16 (l)

Notes: Enter any notes you would like to attach to this reservation

I agree to IBM Technology Zone's [Terms & Conditions](#) and [End User Security Policies](#) (m)

Submit (n)

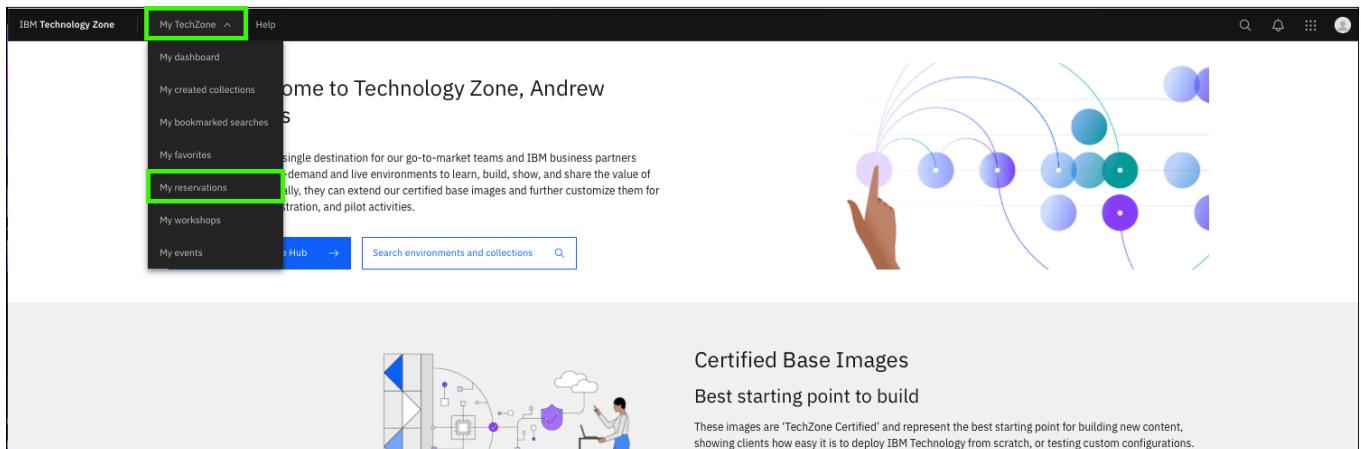
During the provisioning process, multiple emails are sent to you from ITZ as the provisioning process runs. One email states the reservation is provisioning and the other email states that the environment is **Ready**.

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

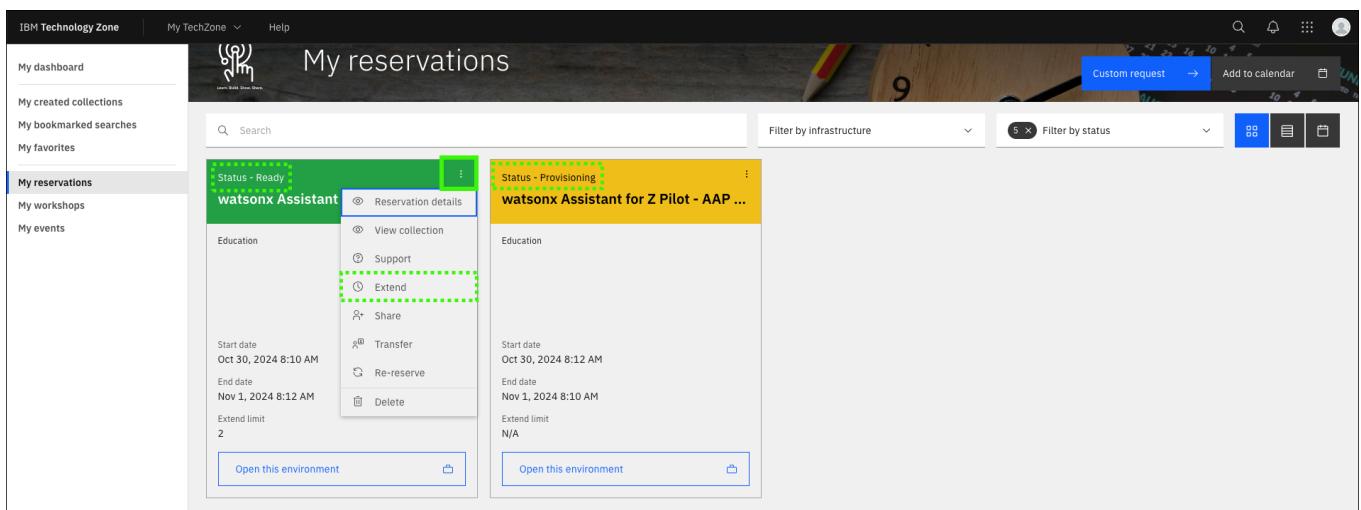
When the reservations are in the **Ready** state, you can extend each reservation beyond its original end date. The duration of the extension will vary by reservation.

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



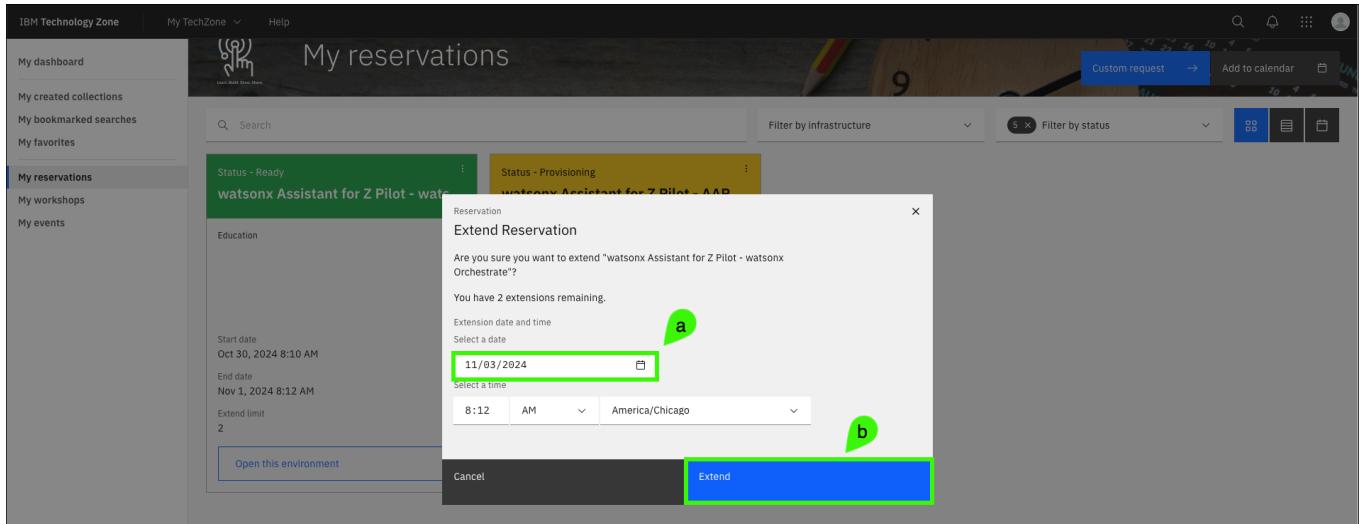
The screenshot shows the IBM Technology Zone portal. The top navigation bar includes 'IBM Technology Zone', 'My TechZone ▾', 'Help', and a search bar. The main content area features a welcome message 'Welcome to Technology Zone, Andrew' and a section titled 'Certified Base Images' with the subtext 'Best starting point to build'. On the left, a sidebar menu lists 'My dashboard', 'My created collections', 'My bookmarked searches', 'My favorites', **My reservations** (which is selected and highlighted with a green box), 'My workshops', and 'My events'. A hand cursor is pointing at the three-dot overflow icon on the 'My reservations' tile.

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



The screenshot shows the 'My reservations' page. The left sidebar is identical to the previous screenshot, with 'My reservations' selected. The main area displays two reservation cards. The first card, for 'watsonx Assistant', has a green dashed box around its 'Extend' option in the context menu. The second card, for 'watsonx Assistant for Z Pilot - AAP ...', also has a context menu open. Both cards show details like start and end dates, and an 'Open this environment' button. The top right of the page includes a search bar, filters for infrastructure and status, and various navigation icons.

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



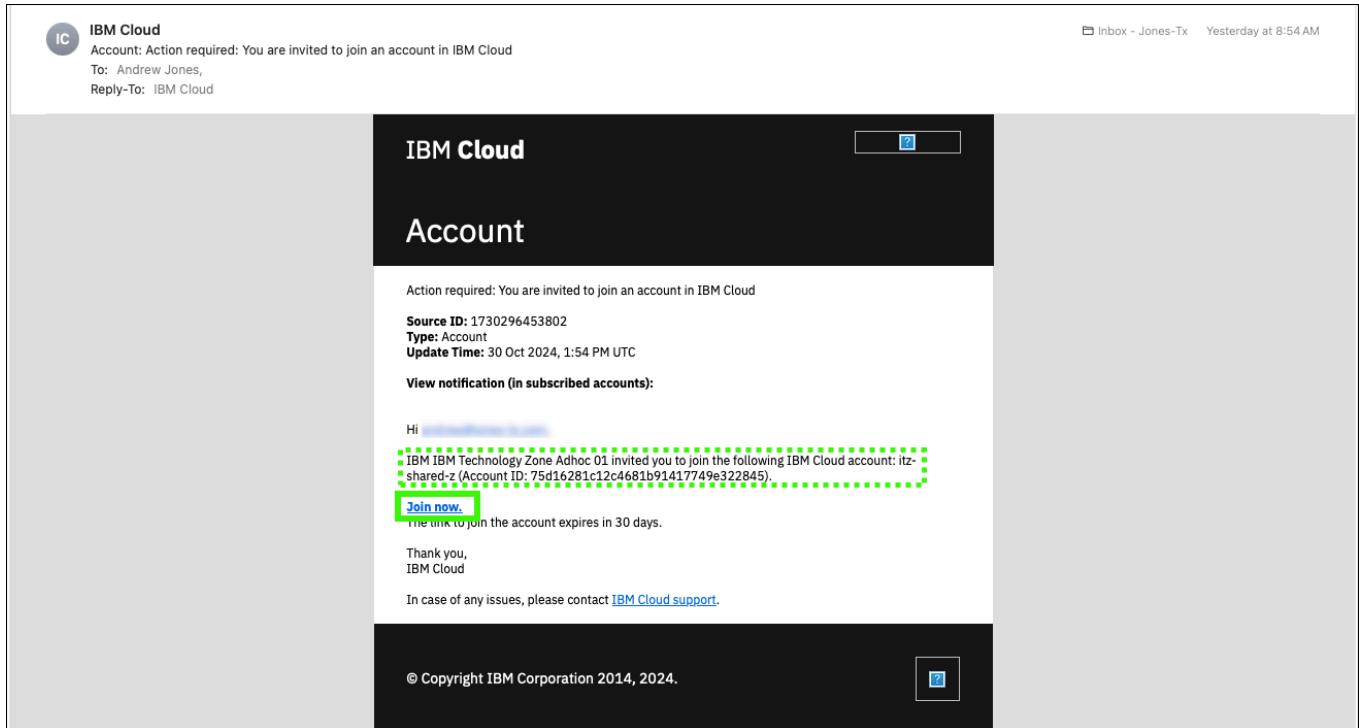
If you anticipate needing more time, repeat Steps 5-6 to extend the reservation to the maximum allowed. Repeat these steps for the other two reservations.

## Join the ITZ IBM Cloud account

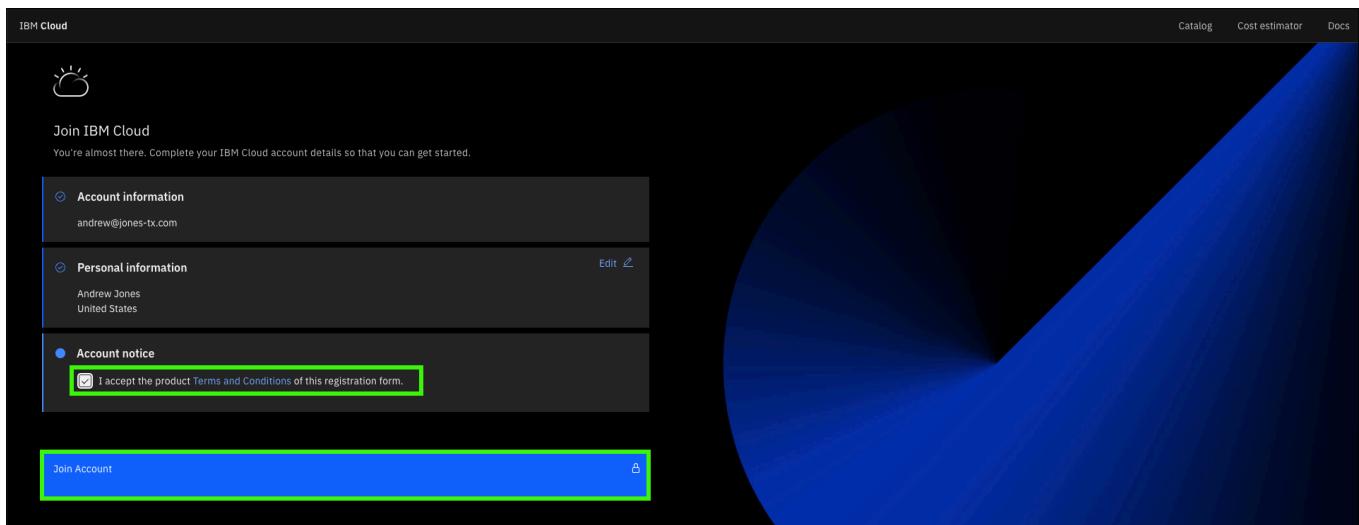
Both the **watsonx Assistant for Z lab – watsonx Orchestrate** and the **Ansible Automation Platform (AAP) & z/OS** environments add you to an IBM Cloud account while your reservation is active. During the provisioning process of these ITZ environments, you receive two emails from IBM Cloud.

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**.

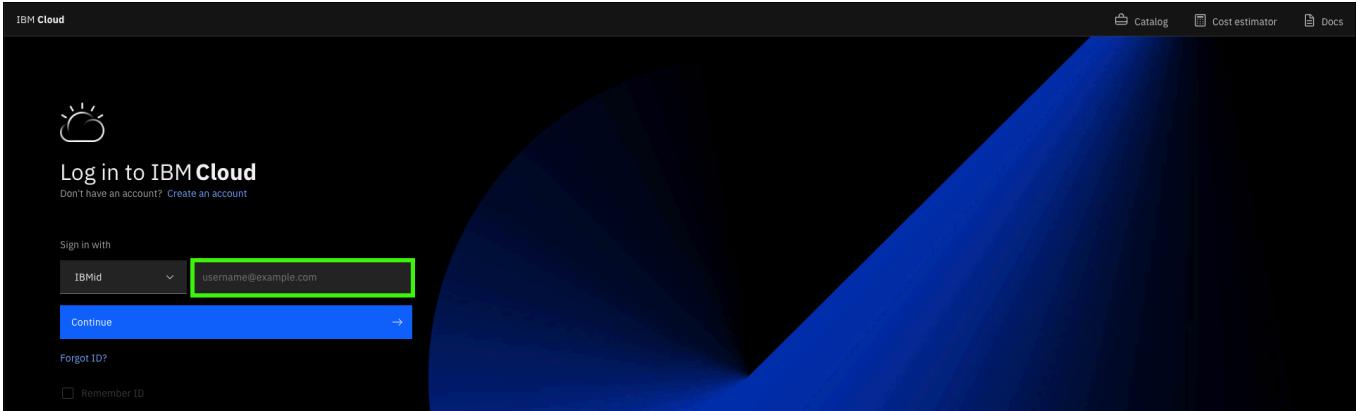


After joining the account, verify that the account appears in your available account list in the IBM Cloud portal.

- 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.

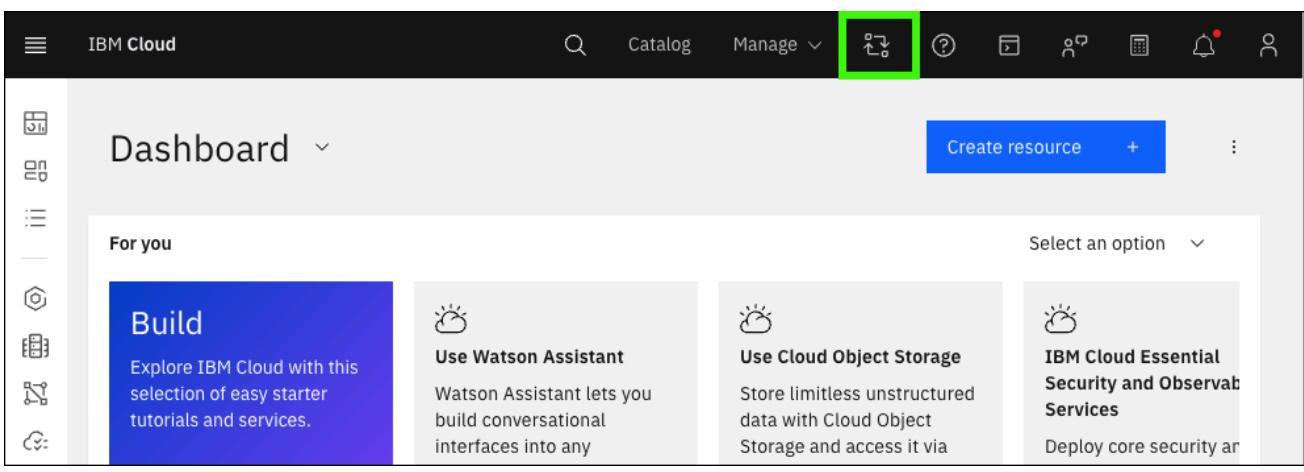


**The account may be different.**

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.



## 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...

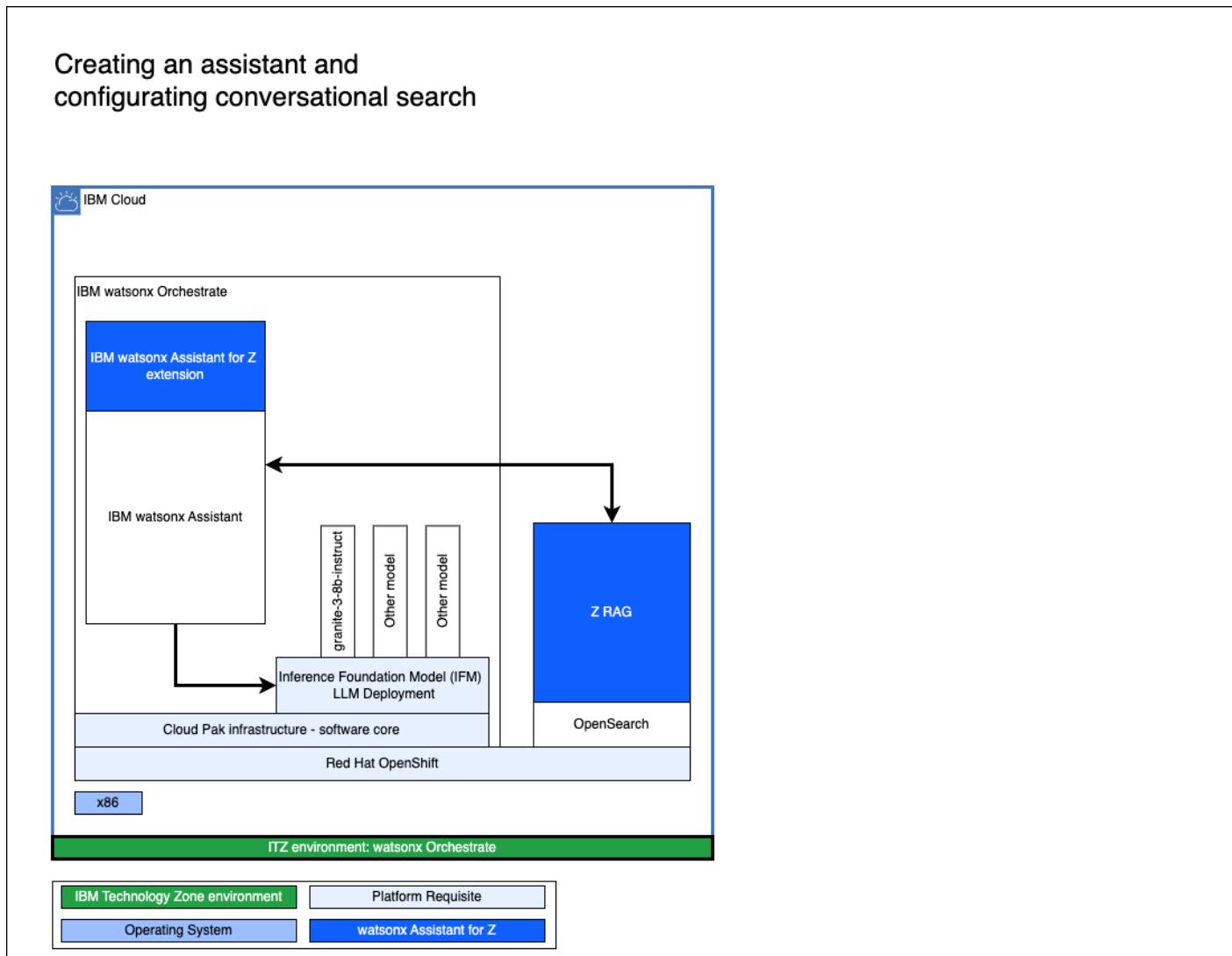
If your reservation for the Single Node OpenShift environment fails, try selecting one of the **eu-gb region** options as the **Preferred Geography**.

# 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](#)

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

## 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](#).

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

**Purpose**

- Purpose
- Education
- Opportunity Product(s)
- Customer(s)

**Environment**

- Reservation ID: 672a091a8f85062f891e081
- Type: IBM Cloud
- Request method: watsonx-orchestrate
- Region: us-south
- Customer data: false
- Idle runtime limit: 10800
- Transaction ID: 115897c9-58a1-4f17-af9c-b16dc2a97590
- Geo: americas
- Datcenter: dal10
- Environment: watsonx-orchestrate-wusdf
- Timeout action:

**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**

Name	Group	Location	Product	Status	Tags
Filter by name or IP address...	Filter by group or org...	Filter...	Filter...	Filter...	Filter...
Compute (0)					
Containers (0)					
Networking (0)					
Storage (0)					
Converged infrastructure (0)					
Enterprise applications (0)					
AI / Machine Learning (1+)					
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. A search bar at the top right contains the text '2953593 - itz-watsonx-036'. Below the search bar, a table lists resources. One row is highlighted with a green box, showing the resource name '2953593 - itz-watsonx-036'.

## Create your Assistant

1. Click the **Resources** icon (☰).

The screenshot shows the IBM Cloud Dashboard. On the left, there is a sidebar with a 'Resources' icon highlighted with a green box. The main area displays a 'Build' section with various service cards. One card for 'Watson Assistant' is visible, along with others for Cloud Object Storage, Maximo Application Suite, and Watson Studio.

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 sidebar has a 'Resources' icon highlighted with a green box. The main area shows a tree view under 'Enterprise applications'. An 'AI / Machine Learning' node is expanded, revealing a single instance row. This row is highlighted with a green box and contains the following details:

Name	Group	Location	Product	Status	Tags
itzwxo-2700039nft-erspw	watsonx-orchestrate-erspw	Dallas	watsonx Orchestrate	Active	

3. Click **Launch watsonx Orchestrate**.

Resource list / itzwxo-2700039nft-erspw Active Add tags [Edit](#)

**Manage**

Service credentials

Start by launching the tool

[Launch Watsonx Orchestrate](#) [Getting started tutorial](#)

Credentials

API key: [Download](#) [Show credentials](#)

URL: <https://api.us-south.assistant-builder.watson.cloud.ibm.com/instances/a7675d8e-e885-4891-92f2-1a2a2a2a2a2a>

**Plan**

Essentials Plan

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

Welcome, Andrew Jones!

Take productivity to the next level.

[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.

**Upgrade**

Standard plan | Skill studio

Automate how your business works

[Learn more](#)

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

Welcome to AI assistant builder

Create  Personalize  Customize  Preview

**Create your first assistant**

Let's get your assistant up and running. Name your assistant, add a description, and choose a language. In following steps we'll gather more information, show you basic customizations, and give you a preview of what your assistant will look like.

Assistant name:

Your assistant name will be kept internally and not visible to your customers.

Description (optional):

Assistant language:

This is the language your assistant will speak.

**Next**

#### 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 deploying your assistant?  
Web

**Tell us about yourself**  
This information will be used to personalize your onboarding experience.  
Which industry do you work in?  
Software

What is your role on the team building the assistant?  
Developer

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

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!

Type something... ➤

## 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**

Assistant's name as known by customers  
Zeeves

Intended purpose  
 Standard: For virtual agents and customer support experiences.  
 Carbon for AI: For use in internal IBM products.

Choose a theme  
Light

Primary color #FFFFFF Secondary color #3D3D3D Chat header User message bubble

Accent color #035AE9

Size  
 The size of the web chat on this page will not change by updating these fields.  
Width 380px Height 640px

IBM Watermark  
Enable IBM Watermark

Streaming  
Enable Streaming  
 Off

Zeeves

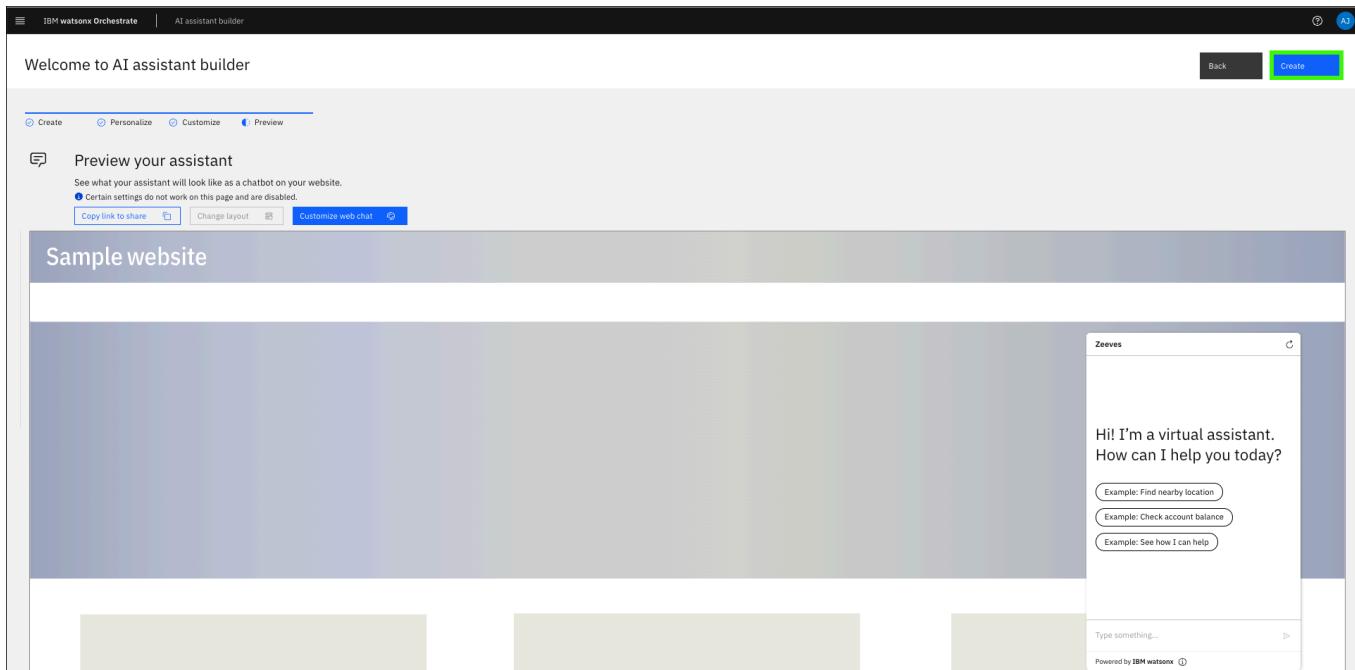
Hi! I'm a virtual assistant.  
How can I help you today?

Example: Find nearby location  
Example: Check account balance  
Example: See how I can help

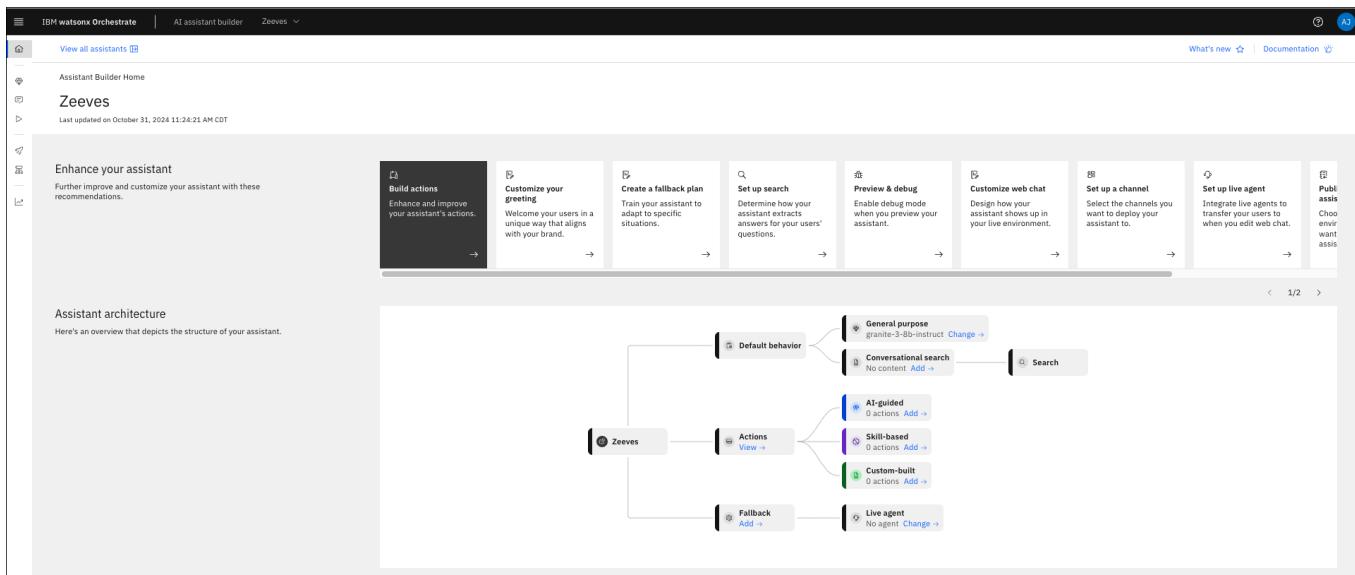
Type something... ➤

Powered by IBM watsonx

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



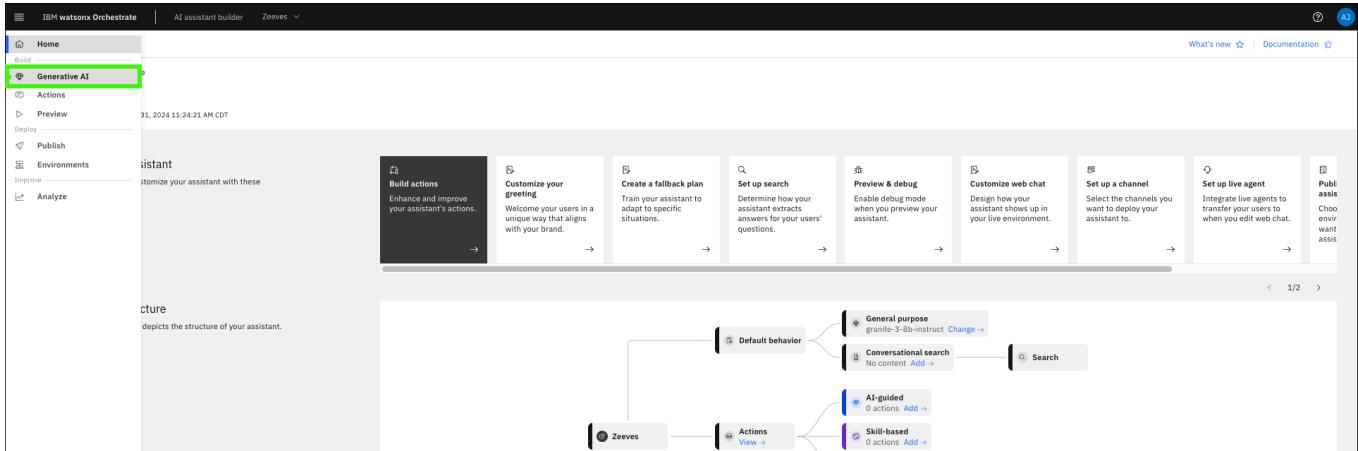
The assistant is now created.



## 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](#).

Generative AI

Base large language model (LLM) Beta

Select a model

granite-3-8b-instruct (recommended)

By using the model you agree to these terms. [Read terms](#)

Add prompt instructions

General-purpose answering

Conversational search

Preview

1:03 PM Greet customer [default] Welcome, how can I assist you?

#### 4. Click Custom service.

Watsonx.ai model deprecation The model granite-13b-chat-v2 will be deprecated on 2024-11-04 and not available after 2025-01-19. Recommended alternative: granite-3-8b-instruct. You will be switched automatically on 2025-01-19.

Base large language model (LLM) Beta

Select a model

granite-13b-chat-v2

Model will not be available after 2025-01-19.

By using the model you agree to these terms. [Read terms](#)

Add prompt instructions

General-purpose answering

Conversational search

Set up a new search integration

**Elasticsearch** RECOMMENDED

Milvus

**Custom service**

Connect to an existing Elasticsearch instance.

Connect to an existing Milvus instance.

Connect to an existing service that will retrieve documents from your content management service. [Learn more](#)

Preview

7:45 AM Greet customer [default] Welcome, how can I assist you?

#### 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.

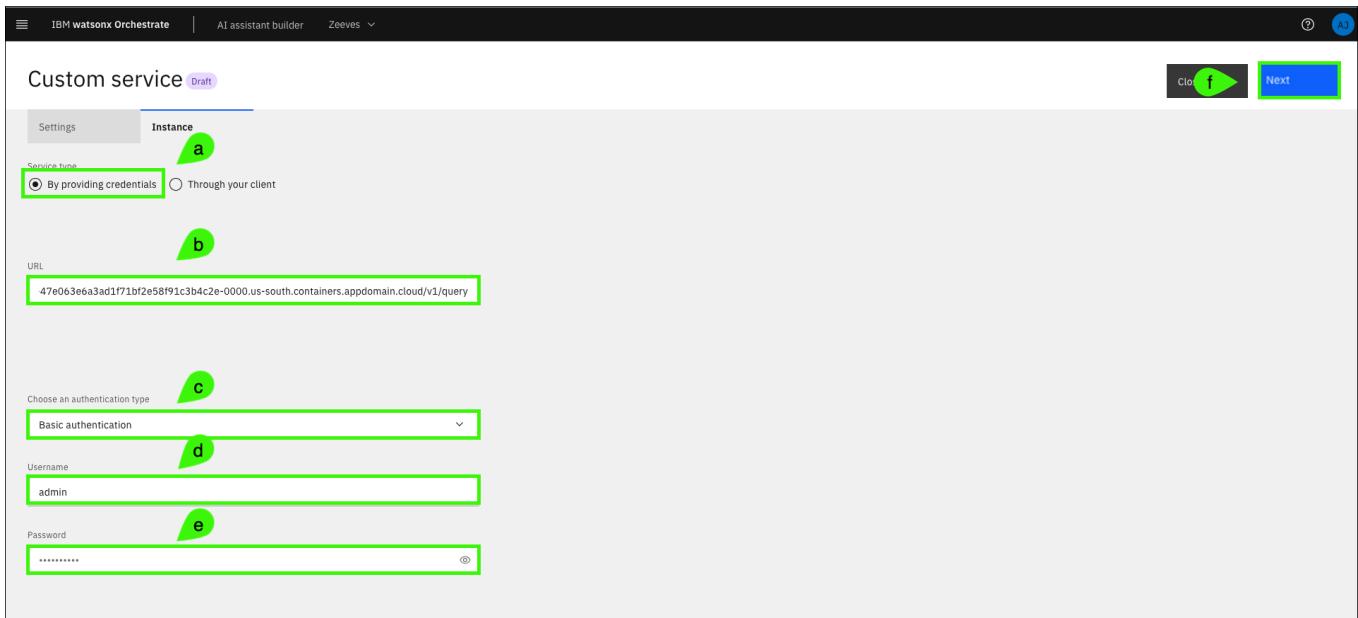
```
https://wxa4z-opensearch-wrapper-wxa4z-demo-v2-1-0.wxo4z-opc-opensearch-clus-47e063e6a3ad1f71bf2e58f91c3b4c2e-0000.us-south.containers.appdomain.cloud/v1/query
```

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

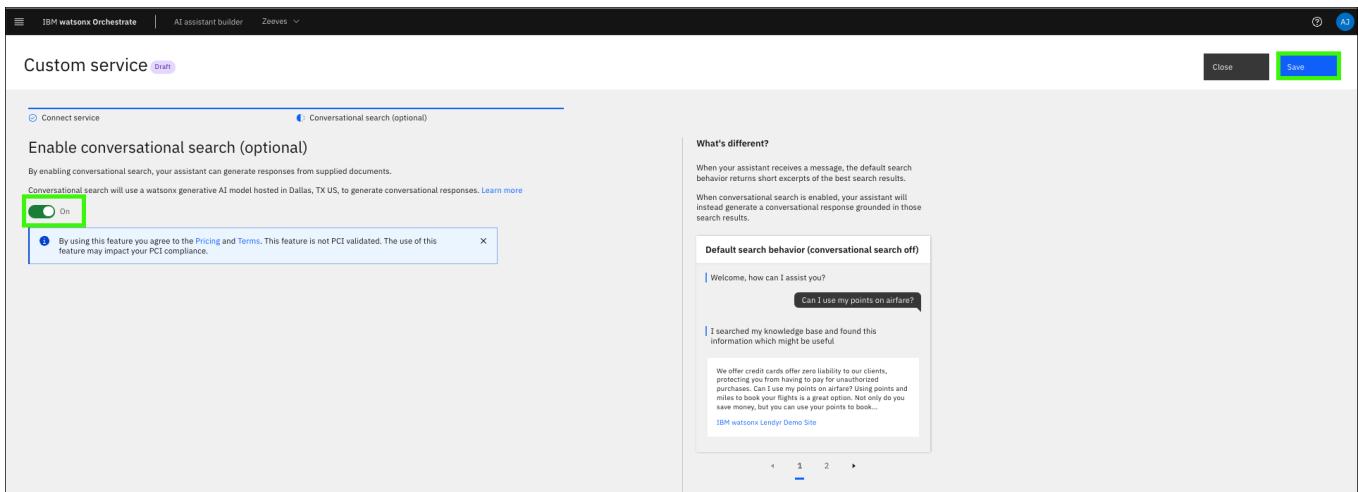
d. Enter **admin** in the **Username** field.

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.

Custom service Draft

**Conversational search**  
Use a watsonx generative AI model hosted in Dallas, TX US, to generate conversational responses.  
[Learn more](#)

**On** **a**

**b** **Single turn**  
The assistant uses only the current user input for retrieving search results and generating answers. This works well for clear, complete inputs but generally won't work with context-dependent queries such as, "Why is that?" after a previous answer.

**c** **Entire conversation** Beta  
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.

Define the text for the citations title. ⓘ

How do we know?

**Search configuration**  
To test this, go to [Citations page](#) ⓘ [Restore default](#)

**Retrieval confidence threshold** ⓘ **d**  **Lowest** Low High Highest

**Generated response length** ⓘ **e**  **Verbose** Concise Moderate

**Response confidence threshold** ⓘ **f**  **Lowest** Low High Highest

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.

Custom service Draft

**Settings** **Instance** **moderate** **verbose**

**Response confidence threshold** ⓘ  **Lowest** Low High Highest

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

**Default filter**  All

**Metadata**  
JSON example:  
{  
 "example\_field": "example\_value",  
 "other\_example\_field": 7  
}

**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 **j**

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

Custom service Draft

**Settings** **Instance**

### Conversational search

Use a watsonx generative AI model hosted in Dallas, TX US, to generate conversational responses. [Learn more](#)

Conversational search  On

Contextual awareness

Single turn  
The assistant uses only the current user input for retrieving search results and generating answers. This works well for clear, complete inputs but generally won't work with context-dependent queries such as, "Why is that?" after a previous answer.

Entire conversation Beta  
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.

Define the text for the citations title. (?)

How do we know?

**Search configuration** Restore default

To test this, go to the [evaluations page](#)

## 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**.

Home

Build

**Generative AI**

**Actions**

Preview Deploy Publish Environments Improve Analyze

Language model (LLM) Beta

model that your assistant uses for all base LLM functions.

recommended)

odel you agree to these terms. [Read terms](#)

ctions

to produce the best general responses. Optionally, you may add instructions to refine the These instructions complement the default prompt, but are not integrated as part of it.

Preview

2:08 PM Greet customer [default]

Welcome, how can I assist you?

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

Actions

All items

Created by you

**Set by assistant**

Variables

Created by you

Set by assistant

Set by integration

Saved responses

Create your first action

With actions, you can help your customers accomplish their goals.

Create action +

### 3. Click No matches.

The screenshot shows the 'Actions' section of the AI assistant builder. A table lists various actions with their names, last edit times, example counts, and statuses. The 'No matches' action is highlighted with a green box around its row.

Name	Last edited	Examples Count	Status
Greet customer	3 hours ago	1	Green
Trigger word detected	3 hours ago	1	Green
No matches	3 hours ago	1	Green
Fallback	3 hours ago	1	Green

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

The screenshot shows the 'Conversation steps' section for the 'No matches' action. Step 1 is highlighted with a green box. It contains a condition 'No matches count <= 3' and a response 'I'm afraid I don't understand. Please rephrase your question.' Below it, there's another condition 'No matches count > 3' and a response 'Action complete'. Step 2 is listed as 'This step has no content'.

### 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 configuration for Step 1. The 'Is taken' dropdown is set to 'with conditions' (a). A confirmation dialog titled 'Clear conditions?' is open, asking if the user wants to take this step without conditions. The 'Clear conditions' button is highlighted with a red box (b).

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

The screenshot shows the configuration for Step 1. The 'Is taken' dropdown is set to 'without conditions'. The 'Assistant says' field is highlighted with a green box. It contains the placeholder text '| or example: Please select from the following options:' and a 'Define customer response' button.

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

The screenshot shows the AI assistant builder interface with the 'Editor' tab selected. In the 'Step 1' section, the 'Is taken' dropdown is set to 'without conditions'. The 'Assistant says' field contains a message template: 'For example: Please select from the following options:' followed by a list of icons. Below this is a 'Define customer response' section. The 'And then' section is expanded, showing a list of actions. The 'End the action' option is highlighted with a green border.

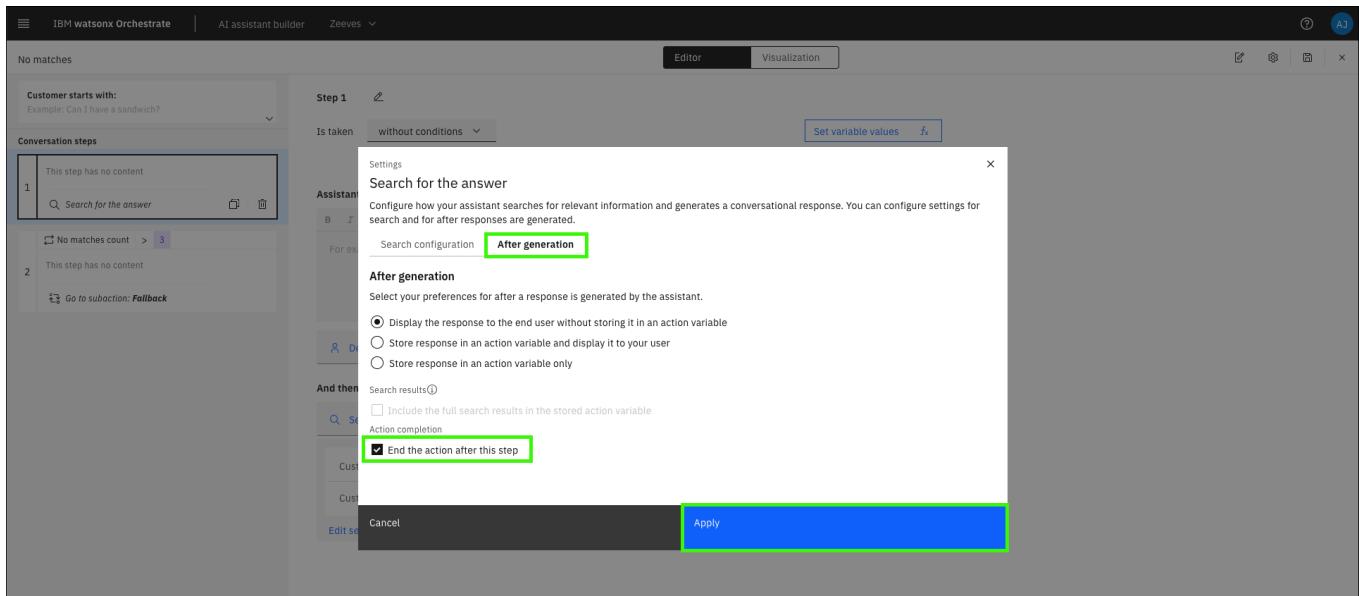
## 8. Click Edit settings.

The screenshot shows the AI assistant builder interface with the 'Editor' tab selected. In the 'Step 1' section, the 'Is taken' dropdown is set to 'without conditions'. The 'Assistant says' field contains a message template: 'For example: Please select from the following options:' followed by a list of icons. Below this is a 'Define customer response' section. The 'And then' section is expanded, showing a list of actions. The 'Search for the answer' option is highlighted with a green border. A modal dialog titled 'Edit settings' is open over the interface, specifically for the 'Search for the answer' action. The dialog shows settings for 'Custom query' (None) and 'Custom filter' (None Optional). The 'Edit settings' button in the dialog is also highlighted with a green border.

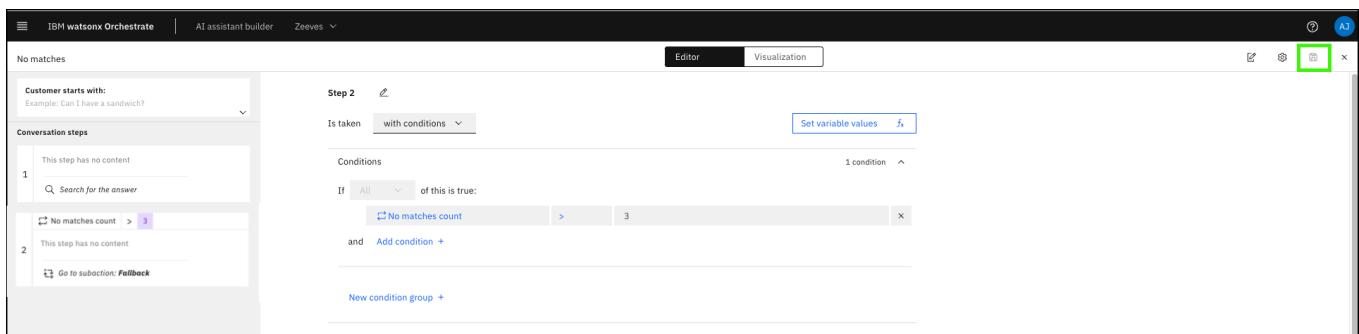
## 9. Click After generation.

The screenshot shows the AI assistant builder interface with the 'Editor' tab selected. A modal dialog titled 'Settings' is open over the interface, specifically for the 'Search for the answer' action. The dialog is titled 'Search configuration' and has a sub-tab 'After generation' selected. It contains fields for 'Custom search query (Optional)' (Example: Which accounts have fees?) and 'Custom results filter' (radio button options: 'Use default' (selected) and 'Set new filter' (Example: enriched\_text.entities.text.:Boston, MA)). At the bottom of the dialog are 'Cancel' and 'Apply' buttons, with 'Apply' being highlighted with a blue background.

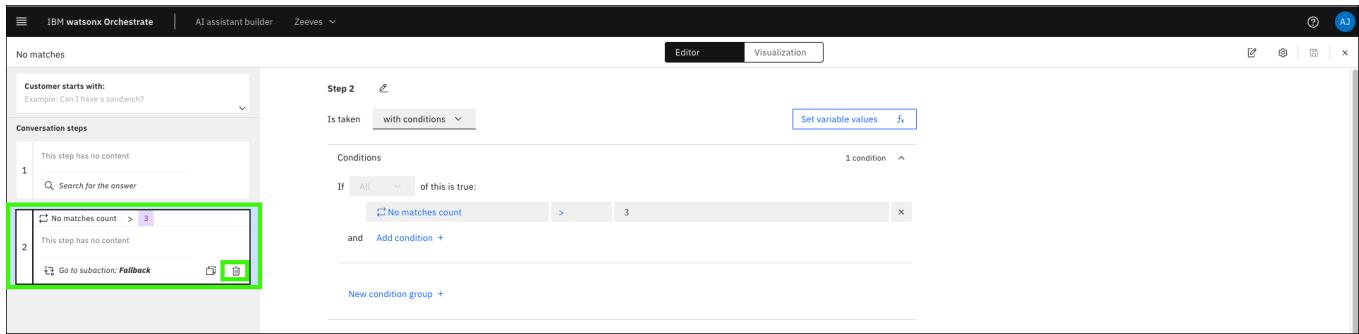
## 10. Select End the action after this step and then click Apply.



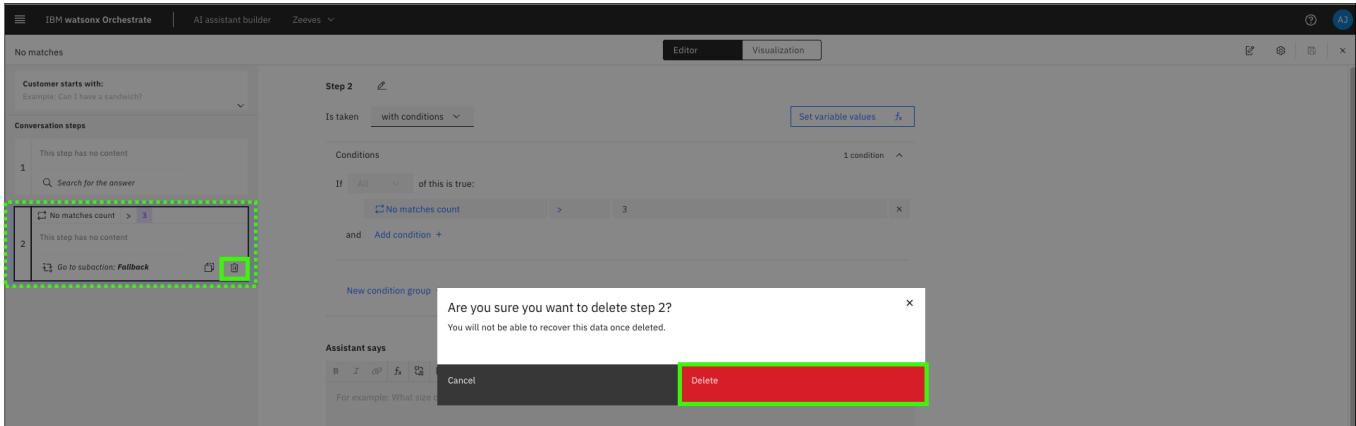
11. Click Save (💾).



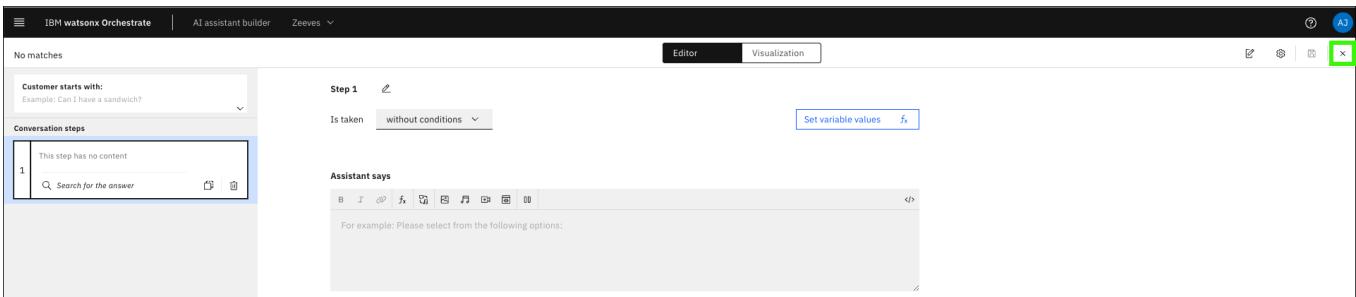
12. Select Step 2 (No matches count) under Conversation steps and click delete (🗑).



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.

Actions				
	Name	Last edited	Examples Count	Status
All items	Greet customer	3 hours ago	0	Green
Created by you	Trigger word detected	3 hours ago	0	Green
Set by assistant	No matches	a minute ago	0	Green
Variables	Fallback	3 hours ago	0	Green
Created by you				
Set by assistant				
Set by integration				
Saved responses				

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 bin icon) and confirm the deletion.

The screenshot shows the AI assistant builder interface for the 'Fallback' scenario. On the left, the 'Conversation steps' list contains five entries, each with a green border and a delete icon (a red square with a white 'X'). The steps are:

- 1 Fallback reason is Failed to confirm topic. (Condition: If All of this is true: Fallback reason is Failed to confirm topic return)
- 2 Fallback reason is Step validation failed. (Condition: If All of this is true: Fallback reason is Step validation failed)
- 3 Fallback reason is Agent requested. (Condition: If All of this is true: Fallback reason is Agent requested)
- 4 Fallback reason is No matches. (Condition: If All of this is true: Fallback reason is No matches)
- 5 Fallback reason is Danger word detected. (Condition: If All of this is true: Fallback reason is Danger word detected)

The right panel shows the 'Step 1' configuration, which is 'Is taken with conditions'. It includes a 'Conditions' section with a single condition: 'If All of this is true: Fallback reason is Failed to confirm topic return'. Below this is the 'Assistant says' section, which contains the response 'Sorry I couldn't confirm if you wanted to return to previous topic, let me connect to an agent.' and a 'Define customer response' button.

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

The screenshot shows the AI assistant builder interface after the conversation steps have been deleted. The 'Conversation steps' list is now empty, indicated by a dashed green border around the list area. The right panel shows the 'Action starts' configuration, which includes a list of triggers: 'Requests to connect to agent', 'Fails step validation within an action', and 'Reaches the limit for No matches'. A note below says 'Use the assistant's default action or customize it.'

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

The screenshot shows the 'Actions' page in the IBM Watsonx Orchestrate interface. The left sidebar has a 'Set by assistant' section selected. The main table lists four actions under the 'Name' column: 'Greet customer', 'Trigger word detected', 'No matches', and 'Fallback'. The 'Status' column for all actions shows a green circle with a checkmark. The top right corner of the table has a green box with a white 'X' icon, indicating the option to delete the row.

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

**Global settings**

Conversation routing Change conversation topic Generative AI **New!** Autocorrection Display formats Algorithm Version Upload/Download

**Ask clarifying questions**

**No matches** (Beta)

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

Customize modes [Beta](#)

Enable disambiguation  On

**Assistant says**

Introduction text before listing the options.

Did you mean:

**No matches**

Option for when the user doesn't see relevant options. Leave blank to omit.

None of the above

**Response modes**

Off

**One action matches**

Option for when user doesn't want to start the matched action

Something else

**Connection to support**

Can be any alternative help, such as a live agent or contact information

Connect 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**

open an account

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 **New!** Autocorrection Display formats Algorithm Version Upload/Download

**Ask clarifying questions**

**No matches** (Beta)

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"

**Save**

## 21. Click **Autocorrection**.

**Global settings**

Conversation routing Change conversation topic Generative AI **New!** **Autocorrection** Display formats Algorithm Version Upload/Download

**Ask clarifying questions**

**No matches** (Beta)

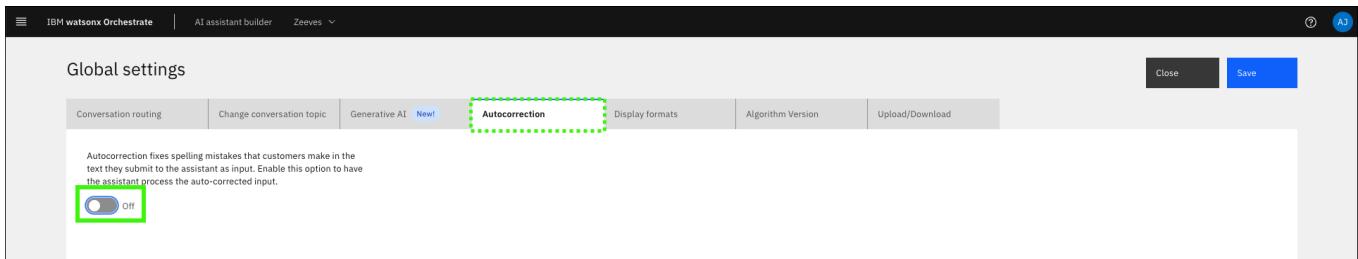
Unrecognized input by customers triggers the **No matches** action that can be configured to fetch answers from a [search integration](#) or trigger the [Fallback](#) action.

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

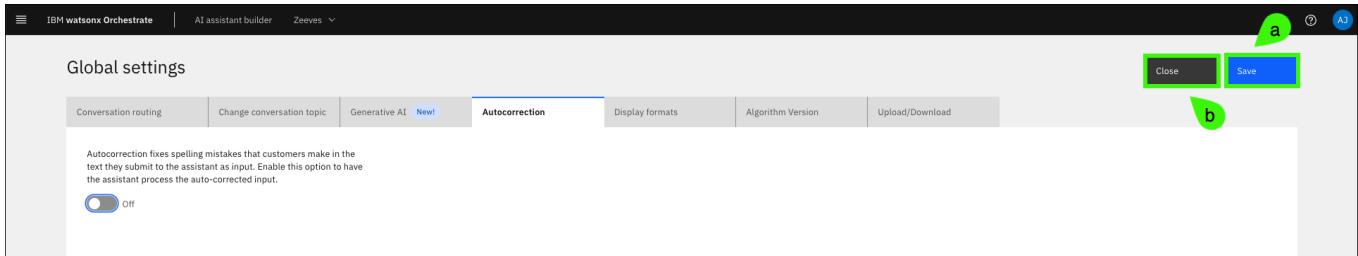
Use "No matches"

**Save**

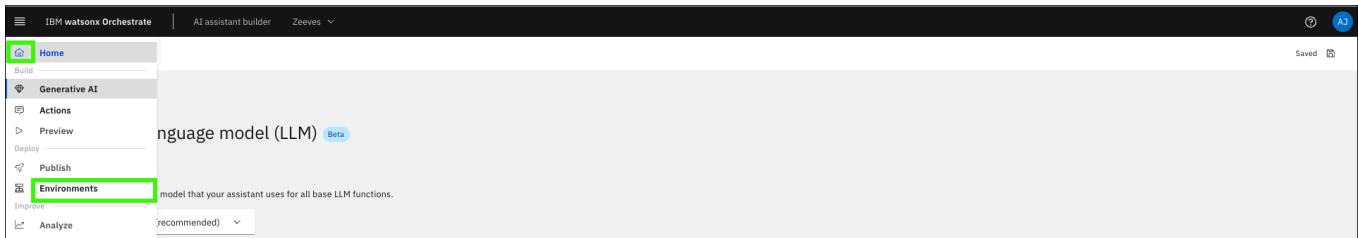
## 22. Click the autocorrection toggle to turn the feature **Off**.



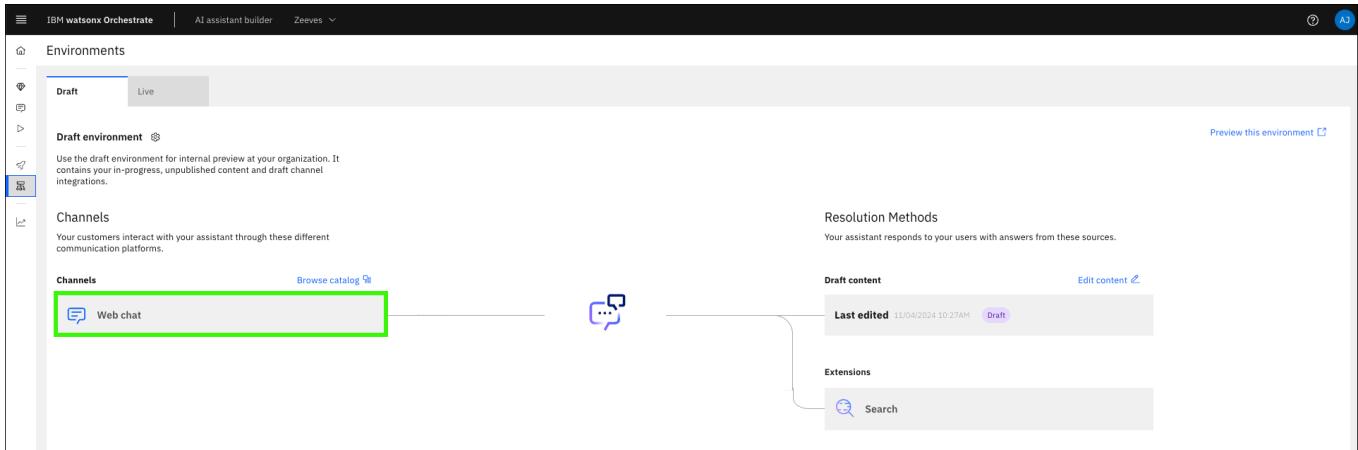
23. Click **Save** (a) and then **Close** (b).



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

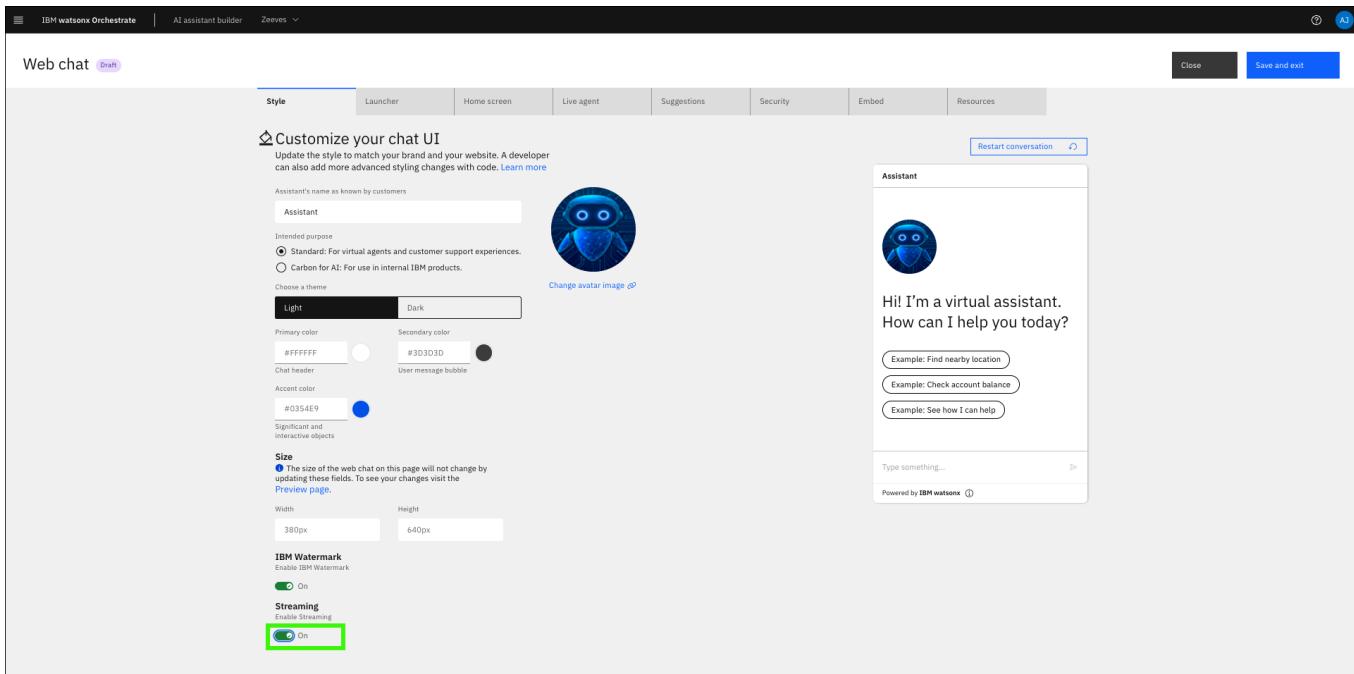


25. Click **Web chat**.

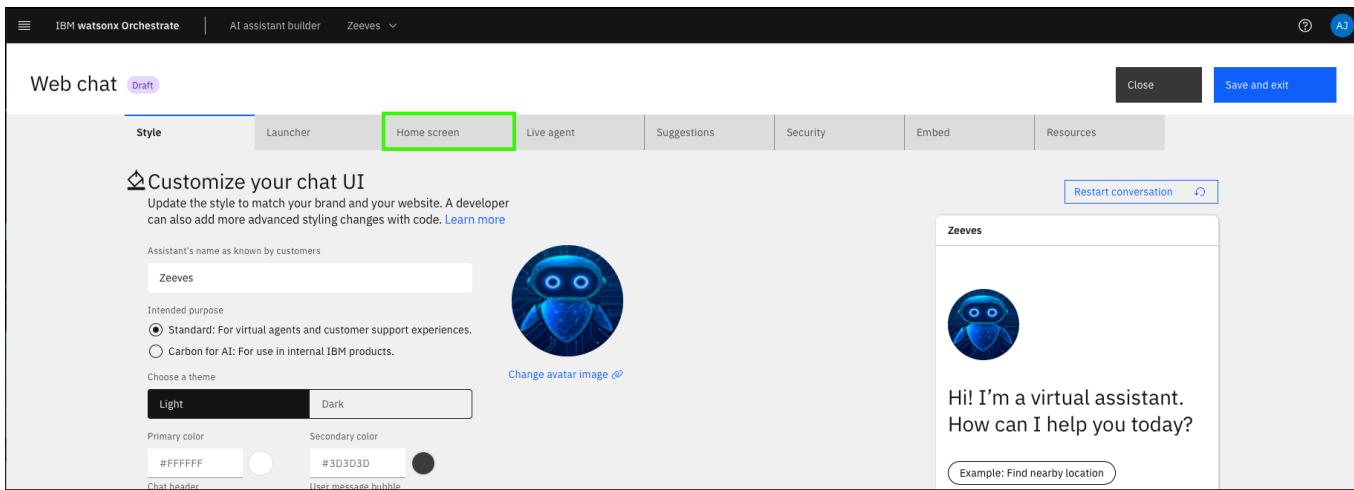


26. On the **Style** tab, click the **Streaming** toggle to enable streaming.

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**.**

**Web chat** Draft

**Style** **Launcher** **Home screen** **Live agent** **Suggestions** **Security** **Embed** **Resources**

**Home screen**

Make a good first impression and get customers the help they need more quickly. [Learn more](#)

On

**Greeting**

Good greetings are welcoming, actionable, and expressive of your assistant's personality.

**Conversation starters**

Help your customers get going. When selected, the text of these starters is sent as a message. Each should be something your assistant is trained and tested to answer.

1 Example: Check account balance

Add a starter (1/5)

**Background style**

Choose a style for your home screen's background.

- None
- Accent color
- Gradient from the top corner
- Gradient from the bottom

**Zeeves**

Hi! I'm Zeeves. How can I help you today?

Type something...

Built with IBM watsonx [?](#)

## 29. Click Suggestions.

**Web chat** Draft

**Style** **Launcher** **Home screen** **Live agent** **Suggestions** **Security** **Embed** **Resources**

**Customize your chat UI**

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

Assistant's name as known by customers: Zeeves

Intended purpose:  Standard: For virtual agents and customer support experiences.  Carbon for AI: For use in internal IBM products.

Choose a theme:  Light  Dark

Primary color: #FFFFFF Secondary color: #303030

Accent color: #0354E9

Significant and interactive objects

Size:  The size of the web chat on this page will not change by updating these fields. To see your changes visit the [Preview page](#).

Width: 380px Height: 640px

**Zeeves**

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

Type something...

Powered by IBM watsonx [?](#)

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

The screenshot shows the 'Web chat' interface in 'Draft' mode. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main tabs are 'Style', 'Launcher', 'Home screen', 'Live agent', 'Suggestions' (which is highlighted in blue), 'Security', 'Embed', and 'Resources'. Below the tabs, there's a section titled 'Suggestions' with a sub-section '② Suggestions' explaining how it helps customers. A green box highlights the 'Off' toggle switch. To the right, the AI character 'Zeeves' is shown with the message 'Hi! I'm Zeeves. How can I help you today?'. At the bottom, there's a text input field 'Type something...' and a note 'Built with IBM watsonx'.

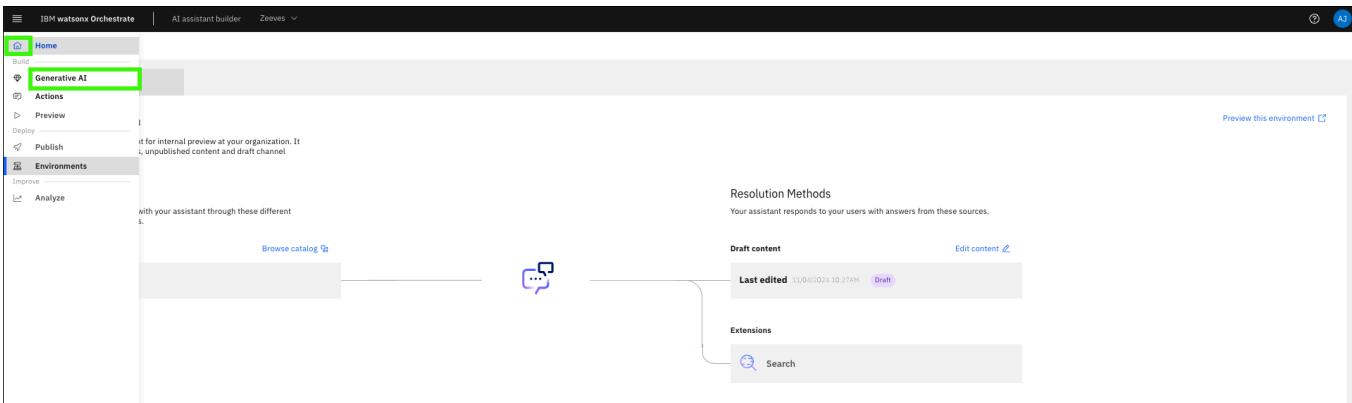
31. Click (a) **Save and exit** and then click (b) **Close**.

This screenshot shows the same interface after performing the steps in the previous step. The 'Save and exit' button is highlighted with a green box (labeled 'a'). The 'Close' button is also highlighted with a green box (labeled 'b'). The AI character 'Zeeves' continues to greet the user with 'Hi! I'm Zeeves. How can I help you today?'.

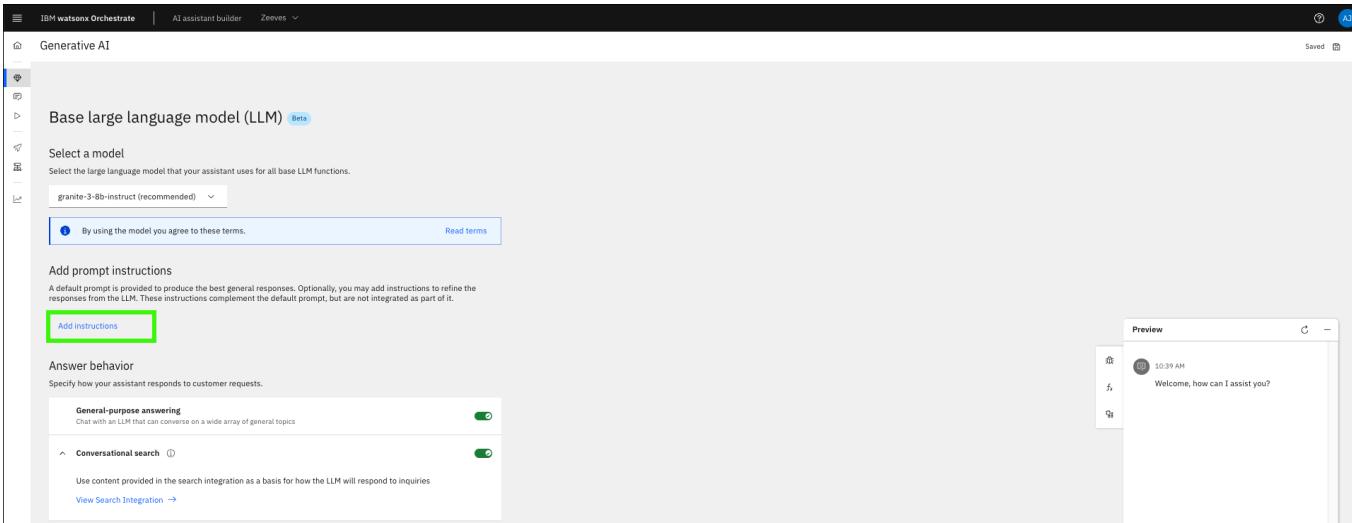
## Configure the base large language model

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 turned on (green). Below it, the 'Conversational search' toggle switch is also turned on (green). A preview window on the right shows a message from 'Greet customer [default]' at 10:47 AM: 'Welcome, how can I assist you?'. The 'Save' button is located in the top right corner of the main configuration area.

#### 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.

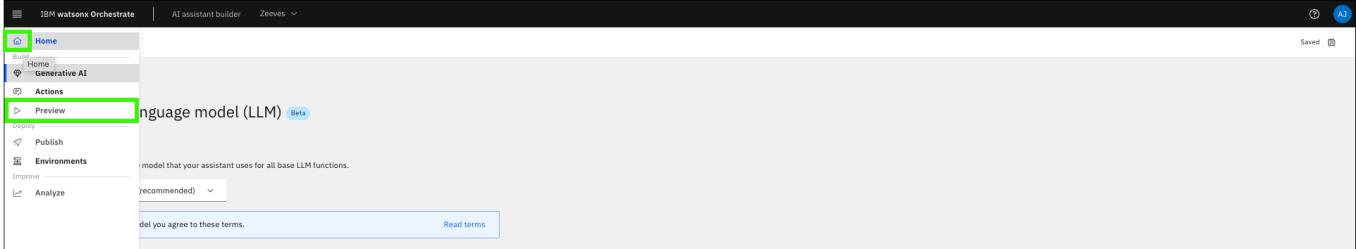
The screenshot shows the 'Generative AI' configuration page. In the 'Answer behavior' section, the 'General-purpose answering' toggle switch is turned off (gray). Below it, the 'Conversational search' toggle switch is turned on (green). A preview window on the right shows a message from 'Greet customer [default]' at 10:47 AM: 'Welcome, how can I assist you?'. The 'Save' button is located in the top right corner of the main configuration area.

## 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:**

**Prompt:**

What is the APF list in z/OS?

**Example output:**

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with icons for preview, AI assistant builder, and Zeeves. The main area is titled "Sample website". In the top right corner, there are buttons for "Copy link to share", "Change layout", "Change background", and "Customize web chat". A small "A3" icon is also present. On the far right, a chat window is open with a message from "Zeeves" at 11:02 AM: "What is the APF list in z/OS? Provide a detailed explanation?" Below it, a response is shown: "The APF (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. These libraries and data sets contain critical programs and utilities that require elevated privileges to function correctly. By authorizing these commands and macros, the APF list ensures system security and integrity by restricting access to sensitive system resources. The APF list is maintained in the system's memory (not on disk) and can be managed using various system commands and macros." At the bottom of the chat window, there's a text input field with "Type something..." and a "Powered by IBM Watsonx" footer.

### Prompt:

Why is Db2 different than other database systems?

### Example output:

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with icons for preview, AI assistant builder, and Zeeves. The main area is titled "Sample website". In the top right corner, there are buttons for "Copy link to share", "Change layout", "Change background", and "Customize web chat". A small "A3" icon is also present. On the far right, a chat window is open with a message from "Zeeves" at 11:03 AM: "Accuracy of generated answers may vary. Why is Db2 different than other database systems?" Below it, a response is shown: "Db2 is a 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 Oracle, MySQL, PostgreSQL, etc. Additionally, Db2 provides a higher level of data management and security features compared to other database systems, as it manages data stored in databases and controls access to the data. Accuracy of generated answers may vary." At the bottom of the chat window, there's a text input field with "Type something..." and a "Powered by IBM Watsonx" footer.

### Prompt:

What happens during an IPL on IBM Z?

### Example output:

The screenshot shows the IBM Watson Assistant for Z interface. At the top, there's a navigation bar with 'IBM Watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. Below the navigation is a toolbar with icons for preview, copy link, change layout, change background, and customize web chat.

The main area is titled 'Sample website' and contains three large, light-gray rectangular boxes. On the far left, there's a vertical sidebar with icons for preview, copy link, change layout, change background, and customize web chat.

To the right of the main area is a chat window titled 'Zeeves'. It shows a message from 'You' at 11:05 AM: 'What happens during an IPL on IBM Z?'. A green box highlights this message. Below it, a response from 'IBM Watson Assistant' at 11:05 AM provides detailed information about the Initial Program Load (IPL) process on IBM Z, mentioning the movement of the system program from the Link Pack Area (LPA) to memory and the beginning of executing it. The response also notes that the process can be initiated from various sources like the Support Element (SE), Hardware Management Console (HMC), or Open Systems Adapter (OSA). It describes how the IPL can be used for loading a new operating system, recovering from a failure, or performing maintenance tasks. It details the detection of the LPA, the reload of the Link Pack Area (LPA), and whether VTO data set pages are preserved. A blue box highlights the 'Powered by IBM Watsonx' footer.



### 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' settings page in the IBM Watsonx Orchestrate interface. The 'Contextual awareness' section is highlighted with a green box around the 'Entire conversation' radio button. The 'Entire conversation' option is selected, while 'Single turn' is unselected. A tooltip for 'Entire conversation' provides a detailed description of how it works.

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

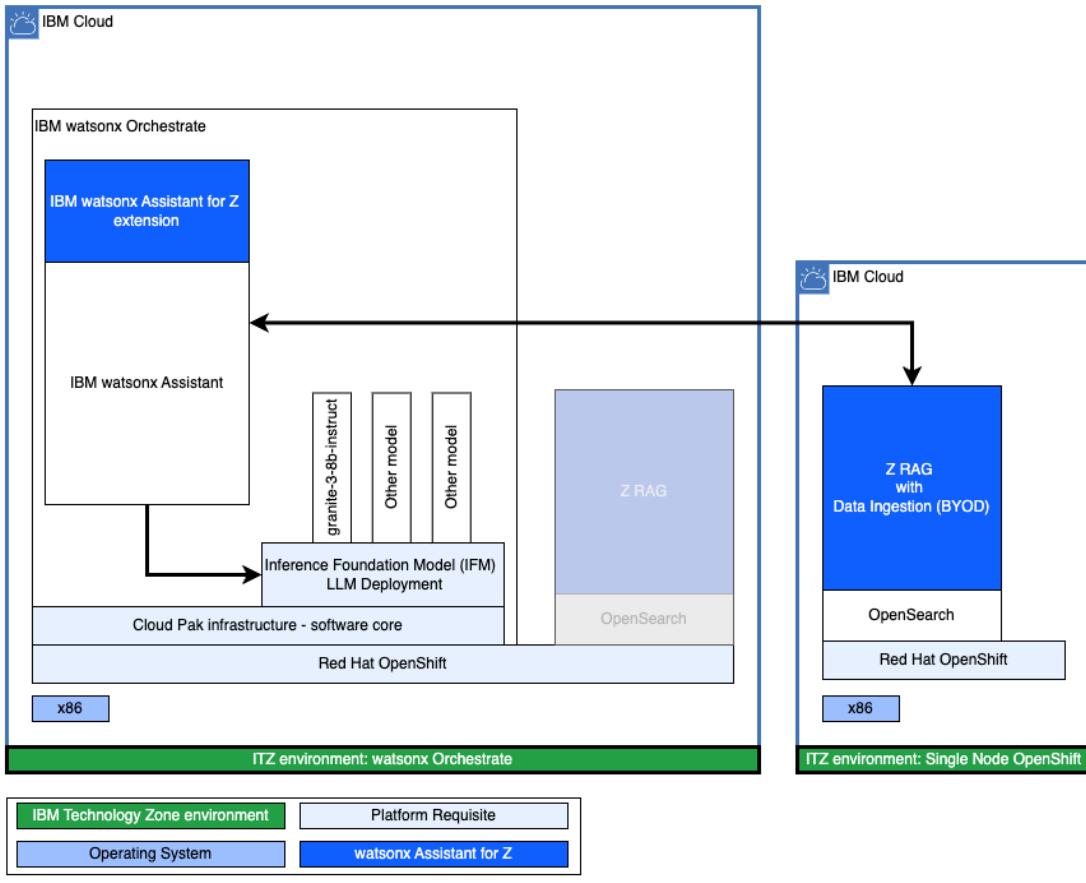
Now that you created and deployed your own assistant with conversational search capabilities, your client can understand how watsonx Assistant for Z provides its content-grounded responses to any Z-related questions. In the previous section, you configured your assistant to use a pre-configured Z RAG 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:6443>

Bastion Password

Bastion RDP address  
api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43389

Bastion SSH connection  
ssh itzuser@api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com -p 40222

Bastion Username  
itzuser

Cluster Admin Username  
kubeadmin

Cluster Admin Password  
[REDACTED]

OCP Console  
<https://console.nginx.ingress.k8s.itzuser-vc.eu.cloud.techzone.ibm.com>

OCP Version  
4.14

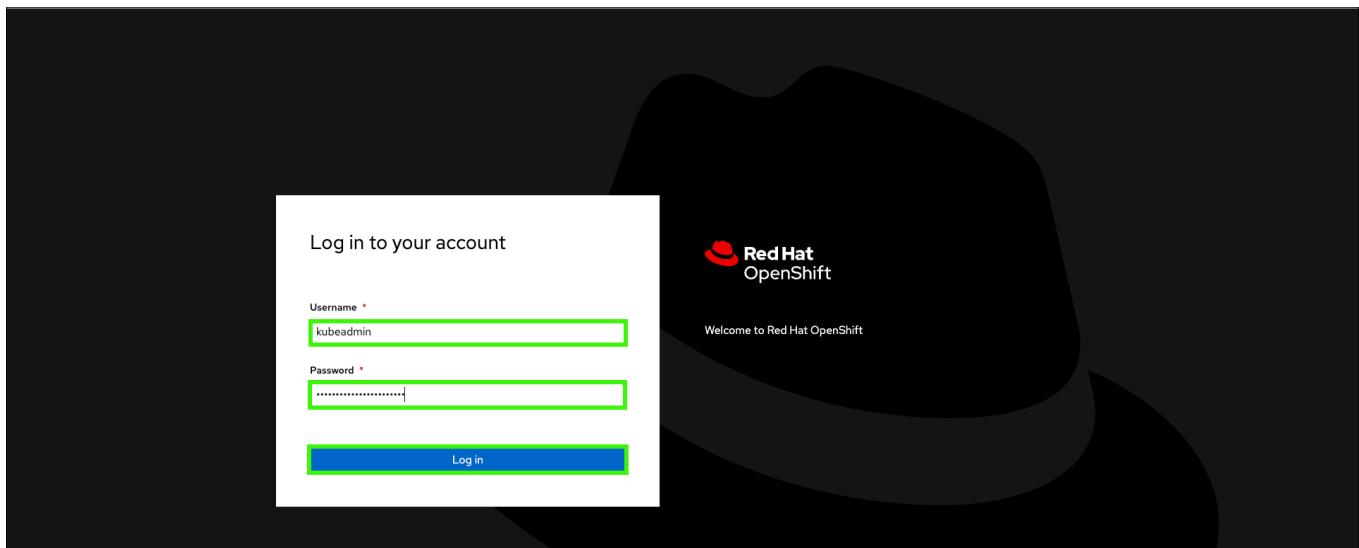
vCenter  
itzuser-vc.eu.cloud.techzone.ibm.com

Download kubeconfig [Download](#)

### 4. Click the OCP Console link.

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

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



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

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

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](#)
- [Download oc for Mac for x86\\_64](#)
- [Download oc for Windows for x86\\_64](#)
- [Download oc for ARM 64](#)
- [Download oc for Mac for ARM 64](#)
- [Download oc for Linux for IBM Power, little endian](#)
- [Download oc for Linux for IBM Z](#)
- [LICENSE](#)

**helm - Helm 3 CLI**

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.

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

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

**Reservation Details**

API URL  
<https://api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:6443>

Bastion Password [REDACTED]

Bastion RDP Address  
api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com:43389

Bastion SSH connection  
ssh itzuser@api.672371d38376796fb96a6c4d.ocp.techzone.ibm.com -p 40222

Bastion Username  
itzuser

Cluster Admin Username  
kubeadmin

Cluster Admin Password [REDACTED]

OCP Console  
<https://console-openshift-console.apps.672371d38376796fb96a6c4d.ocp.techzone.ibm.com>

OCP Version  
4.14

vCenter  
itzeu-vc.eu.cloud.techzone.ibm.com

Download kubeconfig

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

Log in to your account

Welcome to Red Hat OpenShift

Red Hat OpenShift

Username \*

Password \*

Log in

6. 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.

Administrator

Home >

Operators >

Workloads >

Networking >

Storage >

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

7. Click **Display Token**.

Display Token

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

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

Your API token is  
sha256~zuWR0KDnkYniIY0m8g8iKoUXPdFFFmou~o4s5FsrDNA

**Log in with this token**

```
oc login --token=sha256~zuWR0KDnkYniIY0m8g8iKoUXPdFFFmou~o4s5FsrDNA --server=https://api.672b79320c7a71b728e523b4.ocp.techzone.ibm.com:6443
```

**Use this token directly against the API**

```
curl -H "Authorization: Bearer sha256~zuWR0KDnkYniIY0m8g8iKoUXPdFFFmou~o4s5FsrDNA" "https://api.672b79320c7a71b728e523b4.ocp.techzone.ibm.com:6443/v1/users/~"
```

**Request another token**

[Logout](#)

A context menu is open over the token value, with 'Copy' highlighted.

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~mJ4L8K6cUMyNyk2Z69KMm3vbP1sWc8SW0eeOdVqtA94 --server=https://api.672371d38376796fb96a6c4d.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 ~ % cd watsonxAssistant
andrewjones@Andrews-MBP ~ % ls
andrewjones@Andrews-MBP ~ %
```

## 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 left sidebar has a navigation menu with items like 'Administrator', 'Home', 'Operators' (which is currently selected and highlighted in green), 'OperatorHub' (which is also highlighted in green), and 'Installed Operators'. The main content area has a blue header bar with the text 'You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.' Below the header, there's a section titled 'Command Line Tools' with a 'Copy login command' button. Another section titled 'oc - OpenShift Command Line Interface (CLI)' provides information about the 'oc' binary and links to download it for various platforms.

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, there's a navigation sidebar with options like Home, Operators, Workloads, Networking, etc. The 'OperatorHub' section is currently selected. In the main area, there's a dropdown menu labeled 'Project: All Projects'. A green box highlights this dropdown. Below it is a search bar with the placeholder 'Select project...'. There's also a toggle switch for 'Show default projects'. A list of projects is shown, including 'All Projects', 'default', 'kube-node-lease', and 'kube-public'. At the bottom right, it says '624 items'. Navigation tabs at the bottom include 'Community', 'Marketplace', and 'Community' again.

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

This screenshot shows the same interface as the previous one, but the 'openshift-marketplace' operator has been selected from the list of installed operators. A green box highlights this selection. The list includes other operators like 'openshift-kube-controller-manager-operator', 'openshift-kube-scheduler', etc. To the right, there are four tiles representing different operators: CrowdStrike Operator, Hazelcast Platform Operator, Abot Operator, and Accuknox Operator. Each tile has a small icon, a name, a status (e.g., [DEPRECATED]), and a brief description.

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

**Be patient.**

It may take a minute or two for the **IBM Cert Manager** tile to appear.

**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'. The search bar at the top contains 'Q. IBM Cert Manager'. A green box highlights this search term. Below the search bar, there's a list of categories: All Items, Application Runtime, Big Data, Cloud Provider, Database, Developer Tools, etc. Under 'All Items', there's a search result for 'ibm-cert-manager-4.2.7'. This result is highlighted with a green box. It shows the operator name 'IBM Cert Manager', its provider 'provided by IBM', and a brief description: 'Operator for managing deployment of cert-manager service.' At the bottom right, it says '1 items'.

## 7. Click Install.

The screenshot shows the Red Hat OpenShift OperatorHub interface. On the left, there's a sidebar with navigation links like Home, Operators, Workloads, Networking, Storage, Builds, Observe, and Compute. Under Operators, 'OperatorHub' is selected. In the main content area, a search bar at the top right says 'All Items' and has 'IBM Cert Manager' typed into it. Below the search bar, a card for 'ibm-cert-manager-4.2.7' is displayed, which is the version of the IBM Cert Manager operator. To the right of this card, a detailed view of the operator is shown with the title 'IBM Cert Manager 4.2.7 provided by IBM'. It includes sections for 'Channel' (set to v4.2), 'Version' (set to 4.2.7), 'Capability level' (with 'Basic Install' and 'Seamless Upgrades' selected), and 'Source' (ibm-cert-manager-4.2.7). At the bottom right of this detailed view, the word 'Install' is highlighted with a green box.

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

The screenshot shows the 'Install Operator' page for the 'ibm-cert-manager' operator. The left sidebar is identical to the previous screenshot. The main form has several fields: 'Update channel' set to 'v4.2', 'Version' set to '4.2.7', 'Installation mode' set to 'All namespaces on the cluster (default)', 'Installed Namespace' set to 'Operator recommended Namespace: ibm-cert-manager' (with 'Namespace creation' note), and 'Update approval' set to 'Automatic'. On the right side, there are four cards under 'Provided APIs': 'CR CertificateRequest' (Not available), 'CMC Cert Manager Config' (Note about CertManagerConfig schema and license), 'ClusterIssuer' (Not available), and 'Issuer' (Note about Issuer resource). At the bottom of the form, there are two buttons: 'Install' (highlighted with a green box) and 'Cancel'.

**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 extra assistance is needed, refer to this [site](#).

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 IBM Container Software and Cloud Pak Access Management interface. On the left, there's a sidebar with 'My IBM', 'Profile', 'Billing', 'Container Software and Cloud Pak Access Management', and 'Entitlement keys' (which is selected). Below that is 'Container software library'. The main area has a title 'Entitlement keys (1)'. It says 'Access your container software' and explains that an entitlement key allows access to all container software in the IBM Entitled Registry. It shows one active entitlement key issued on October 17, 2022. The key itself is a long string of characters. To the right of the key are 'Copy' and 'Delete' buttons, with 'Copy' highlighted by 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 running commands to create a namespace and a pull secret. The pull secret command is highlighted with a green border. The output shows the secret was created successfully.

```
andrewjones@Andrews-MBP watsonxAssistant % oc create namespace wxa4z-byos  
namespace/wxa4z-byos created  
andrewjones@Andrews-MBP watsonxAssistant % export IBM_CS_ENT_KEY=  
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 %
```

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.eyJpc3MiOiJJQk0gTWFya2V0cGxhY2UiLCJpYXQiOjE2NjYwMTk1ODAsImp0aSI6IjNkOWUyMzjZTAzMDQzMzVhNTJhYTkzMWNmOTcyMDR1In0.4M3XRDXzkHMSOkFNJ4uKVcWZ6SnEA0Z03eL_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/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 search bar at the top contains 'wxa4z-byos'. Below the search bar, there's a list of operators categorized by provider: Community, Marketplace, and Red Hat. One operator, 'ibm watsonx Assistant for Z Operator', is highlighted with a green box.

## 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. The search bar at the top contains 'ibm watsonx'. Below the search bar, there's a list of operators. One operator, 'ibm watsonx Assistant for Z Operator Catalog', is 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.

You are logged in as a temporary administrator.

Project: wxa4z-byos

### OperatorHub

Discover Operators from the Kubernetes community and Red Hat partners, curated by Red Hat. You can purchase services to your developers. After installation, the Operator capabilities will appear in the Developer Catalog provided by IBM.

All Items

Search: ibm watsonx

**IBM watsonx Assistant for Z Operator Catalog**

provided by IBM

IBM watsonx Assistant for Z Operator

**Install**

**Channel**: stable

**Version**: 2.1.0

**Capability level**: Basic Install

- Seamless Upgrades
- Full Lifecycle
- Deep Insights
- Auto Pilot

**Source**: IBM watsonx Operator Catalog

**Provider**: IBM

**Repository**: N/A

**Container image**: N/A

**Created at**: Oct 4, 2024, 1:31 PM

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

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

OperatorHub > Operator Installation

### Install Operator

Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.

**Update channel\***: stable

**Version\***: 2.1.0

**Installation mode\***:

- All namespaces on the cluster (default)  
Operator will be available in all Namespaces.
- A specific namespace on the cluster  
Operator will be available in a single Namespace only. (Selected)

**Installed Namespace\***: wxa4z-byos

**Update approval\***:

- Automatic (Selected)
- Manual

**Install** **Cancel**

**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.**

The screenshot shows the Red Hat OpenShift web console interface. On the left, there's a sidebar with various navigation options like Home, Operators, Workloads, Networking, Storage, Builds, Observe, Compute, User Management, and Administration. The 'Operators' section is expanded, and 'OperatorHub' is selected. The main content area displays a list of operators. One operator, 'ibm-wxa4z-operator:v2.0.1 provided by IBM', is highlighted with a green dashed box. Below the operator name, the message 'Installed operator: ready for use' is visible, along with two buttons: 'View Operator' and 'View installed Operators in Namespace wxa4z-byos'. The top right corner shows the user is logged in as 'kubeadmin'.

- 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.eyJpc3MiOiJJQk0gTWFya2V0cGxhY2UiLCJpYXQiOjE2NjYwMTk1ODAsImp0aS16IjNkOWUyNzZjZTAzMDQzMzNhNTJhYTkzMWNmOTcyMDR1In0.4M3XRD4XzkHMSOkFNJ4uKVcWZ6SnEA0Z03el_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**.

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

ibm watsonx Assistant for Z  
ibm-wxa4z-operatorv2.0.1 provided by IBM

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

Name	Status	Ready	Restarts	Owner	Memory	CPU	Created
1d126367b1ca53dcf2b0c93acd7	Completed	0/1	0	1d126367b1ca53dcf2b0c93acd733	-	-	Nov 4, 2024, 4:44 PM
ibm-wxa4z-operator-catalog-n9m5n	Running	1/1	0	CS ibm-wxa4z-operator-catalog	19.0 MiB	0.003 cores	Nov 4, 2024, 4:32 PM
ibm-wxa4z-operator-controller-manager-7c798d744-87htm	Running	2/2	0	RS ibm-wxa4z-operator-controller-manager-7c798d744	31.0 MiB	0.001 cores	Nov 4, 2024, 4:44 PM

#### 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=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9eyJpc3MiOiJJQk0gTWFya2V0cGxhY2UiLCJpYXQiOjE2NjYwMTk1ODAsImp0aSI6IjNkOWUyMzZjZTAzMDQzMzVhNTJhYTkzMWNm0TcyMDR1In0.4M3XRD4XzkHMSOkFNJ4uKvCwZ6SnEAZO3el_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/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 % 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
da80909aab1563ff0c15b7c6d	Completed	0/1	0	ds80909aab1563ff0c15b7c6da81 a8bbefabaa605065bb6c6b490e87 0e87f0a96f	-	-	Nov 6, 2024, 9:26 AM
ibm-wxa4z-operator-catalog-cfvsv	Running	1/1	0	CS ibm-wxa4z-operator-catalog	25.0 MiB	0.004 cores	Nov 6, 2024, 9:25 AM
ibm-wxa4z-operator-controller-manager-556fcf98bb-4bhmk	Running	2/2	0	RS ibm-wxa4z-operator-controller-manager-556fcf98bb	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-7f98d86c58-9bzth	Running	1/1	0	RS wxa4z-client-ingestion-7f98d86c58	501.8 MiB	0.071 cores	Nov 6, 2024, 12:18 PM
wxa4z-opensearch-wrapper-5cb879f5f8-qw7qt	Running	1/1	0	RS wxa4z-opensearch-wrapper-5cb879f5f8	547.1 MiB	0.031 cores	Nov 6, 2024, 10:22 AM
wxa4z-snapshot-setup-job-nsqlz	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
da80909aab1563ff0c15b7c6d	Completed	0/1	0	ds80909aab1563ff0c15b7c6da81 a8bbefabaa605065bb6c6b490e87 0e87f0a96f	-	-	Nov 6, 2024, 9:26 AM
ibm-wxa4z-operator-catalog-cfvsv	Running	1/1	0	CS ibm-wxa4z-operator-catalog	25.0 MiB	0.004 cores	Nov 6, 2024, 9:25 AM
ibm-wxa4z-operator-controller-manager-556fcf98bb-4bhmk	Running	2/2	0	RS ibm-wxa4z-operator-controller-manager-556fcf98bb	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-7f98d86c58-9bzth	Running	1/1	0	RS wxa4z-client-ingestion-7f98d86c58	534.6 MiB	0.087 cores	Nov 6, 2024, 12:18 PM
wxa4z-opensearch-wrapper-5cb879f5f8-qw7qt	Running	1/1	0	RS wxa4z-opensearch-wrapper-5cb879f5f8	547.1 MiB	0.030 cores	Nov 6, 2024, 10:22 AM
wxa4z-snapshot-setup-job-nsqlz	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.

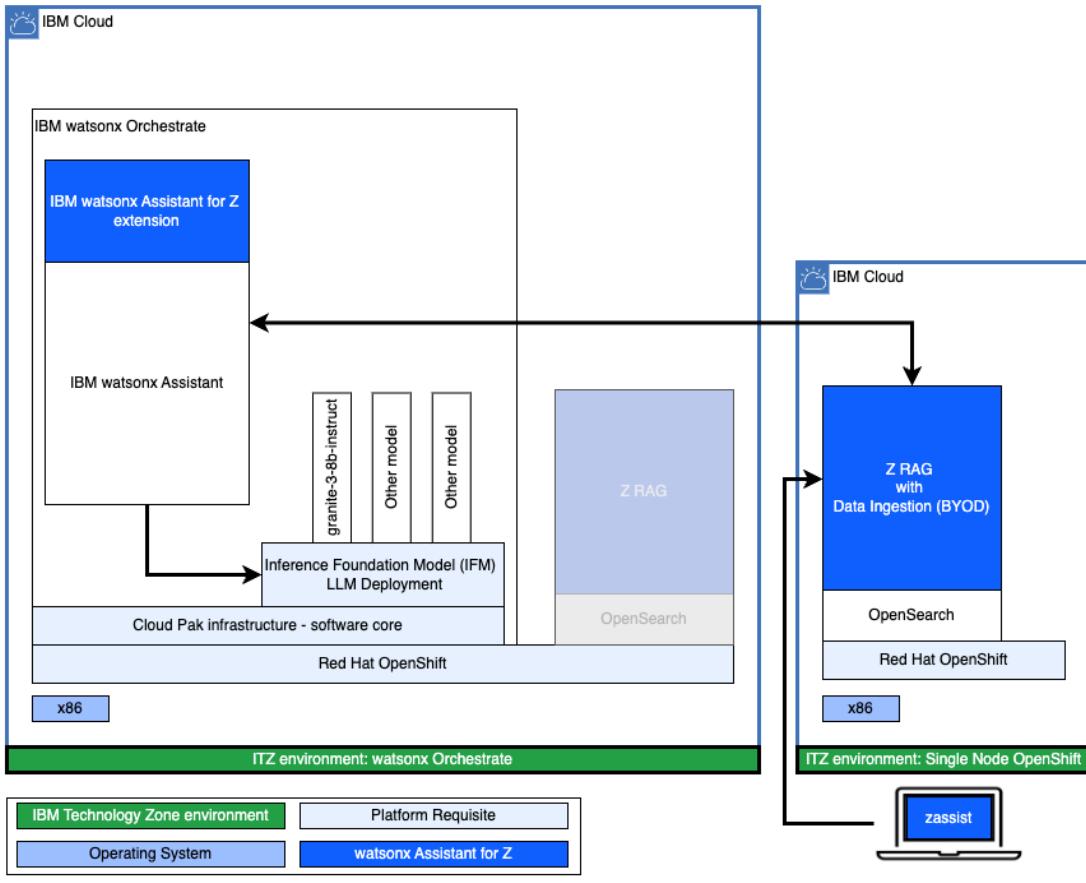
As an example, a client mentioned that their developers often need reference material on company-specific legacy code or company-specific syntax. The users must search through volumes of documentation to find it or look at old code. Also, a need for their operational support group to quickly determine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

### **i** 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>

The screenshot shows a file listing interface. At the top, there's a header with a user icon, the text 'Box@IBM', and a 'zassist.zip' file thumbnail. Below the header is a search bar labeled 'Search files and folders'. The main area displays a single file entry: 'zassist.zip' with a yellow folder icon. To the right of the file name are columns for 'Modified' (Sep 5 at 9:15 AM) and 'Size' (--). At the bottom right of the interface are several small icons: three dots, a green 'Download' button, and a person icon.

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

The screenshot shows a file browser window titled 'watsonxAssistant'. The left sidebar lists several configuration files like 'byos.yaml', 'ca.crt', and 'catalogSource.yaml'. The 'zassist' folder is expanded, showing sub-directories 'linux', 'mac', 'mac-arm', and 'windows'. Inside the 'mac' directory, the 'zassist' file is highlighted with a red box and a green dotted selection border. The right side of the screen is a detailed file list with columns for 'Name', 'Date Modified', 'Size', and 'Kind'. The 'zassist' file under 'mac' is listed with a modified date of 'Sep 5, 2024 at 11:15 AM', a size of '10.2 MB', and a kind of 'Unix Executable File'. Other files in the list include 'byos.yaml' (modified 'Today at 12:17 PM'), 'ca.crt' (modified 'Today at 11:36 AM'), and 'zassist.zip' (modified 'Today at 1:47 PM').

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.

The screenshot shows a terminal window with a light gray background. The prompt is 'andrewjones@Andrews-MacBook-Pro watsonxAssistant %'. The user types 'zassist' and presses Enter. The terminal then displays an error message: 'zassist: error: expected one of "version", "init", "login", "ingest", "load", ...'. The entire command and its output are highlighted with a green dotted selection border.



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
```

## 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.67850b0240c621133ba8749.ocp.techzone.ibm.com	wxa4z-client-ingestion
wxa4z-opensearch-wrapper	Accepted	https://wxa4z-opensearch-wrapper-wxa4z-byos.apps.67850b0240c621133ba8749.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
```

```
andrewjones@Andrews-MBP BYOD % export TLS_VERIFY=false
andrewjones@Andrews-MBP BYOD % zassist init
andrewjones@Andrews-MBP BYOD % echo https://$(oc -n wxa4z-byos get route wxa4z-client-ingestion -o jsonpath=".spec.host")
https://wxa4z-client-ingestion-wxa4z-byos.apps.67850b0240c6211133ba8749.ocp.techzone.ibm.com
andrewjones@Andrews-MBP BYOD % oc -n wxa4z-byos get secret client-ingestion-authkey -o jsonpath=".data.authkey" | base64 -d
[REDACTED]
andrewjones@Andrews-MBP BYOD % zassist login https://wxa4z-client-ingestion-wxa4z-byos.apps.67850b0240c6211133ba8749.ocp.techzone.ibm.com
Enter Password: *
Success.
Gathering local files...
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
andrewjones@Andrews-MBP BYOD %
```

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 IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. Below the navigation is a toolbar with 'Copy link to share', 'Change layout', 'Change background', and 'Customize web chat'. On the left, there's a sidebar with various icons. The main area is titled 'Preview assistant' and shows 'Zeeves'. A blue circular icon of an owl is displayed. Below it, the text 'Hi! I'm Zeeves. How can I help you today?' is shown. At the bottom, a text input field contains the prompt: 'The customer application is failing with ERR-CBL-001, what does this internal error mean?'. To the right of the input field is a blue send button with a white arrow.

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

The customer application is failing with error ERR-CBL-001, which means that a null value was encountered in a mandatory field. This error typically occurs when a required field in the application is left blank or contains a null value. It is important to investigate and resolve this issue as soon as possible to ensure the proper functioning of the application.

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

**CDBL0001E**  
The z/TPF debugger works as a client in debug sessions, where TPF Toolkit is listening for the requests from the z/TPF debugger. pgmname The name of the program that the error originated...

**Mainframe\_COBOL\_Error\_Codes....**  
ERR-CBL-001: Null value encountered in mandatory field.



**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.

**Company-Specific Mainframe COBOL Error Codes**

**List of Error Codes**

- ERR-CBL-001: Null value encountered in mandatory field.
- ERR-CBL-002: Overflow in numeric computation.
- ERR-CBL-003: File not found in JCL input directory.
- ERR-CBL-004: Duplicate record detected during WRITE operation.
- ERR-CBL-005: Index out of bounds in table processing.
- ERR-CBL-006: Invalid REDEFINE clause detected in structure.
- ERR-CBL-007: S0CA abend due to invalid memory reference.
- ERR-CBL-008: Division by zero encountered in arithmetic operation.
- ERR-CBL-009: INSUFFICIENT MEMORY: Unable to allocate storage for SORT operation.
- ERR-CBL-010: File status 35: File not opened correctly before access.
- ERR-CBL-011: Data mismatch error in READ INTO clause.
- ERR-CBL-012: Invalid key on SEARCH ALL for indexed file.
- ERR-CBL-013: S0C7 abend: Data exception encountered.
- ERR-CBL-014: Lock timeout in CICS transaction.
- ERR-CBL-015: File status 90: General file handling error.
- ERR-CBL-016: Job failed due to exceeded time allocation (MAXCC=12).
- ERR-CBL-017: Attempt to WRITE beyond maximum file size.
- ERR-CBL-018: Illegal MOVE operation detected on COMP-3 field.
- ERR-CBL-019: Misaligned DD statement in JCL.

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 AI assistant builder. The 'Metadata' field is highlighted with a green dashed box. The field contains a JSON example:

```
{
  "example_field": "example_value",
  "other_example_field": 7
}
```

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](#).

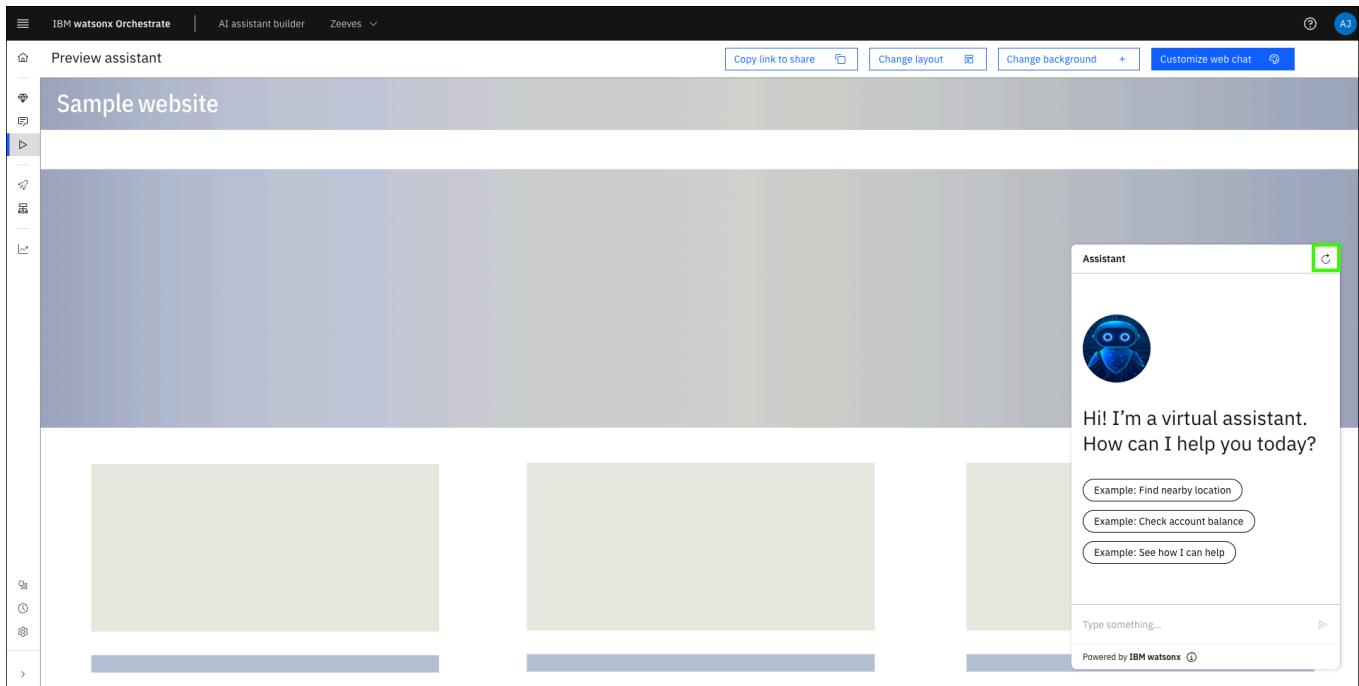
1. Set the (a) **Metadata** field for your BYOS custom search instance to the following value, click (b) **Save**, and then click (c) **Close**. Notice the weight for **customer\_docs** is heavier than the weight for **product\_docs**.

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

The screenshot shows the 'Custom service' configuration page in the IBM Watsonx Orchestrate AI assistant builder. The 'Metadata' field is highlighted with a green box and labeled 'a'. The 'Save' and 'Close' buttons are labeled 'b' and 'c' respectively.

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
```

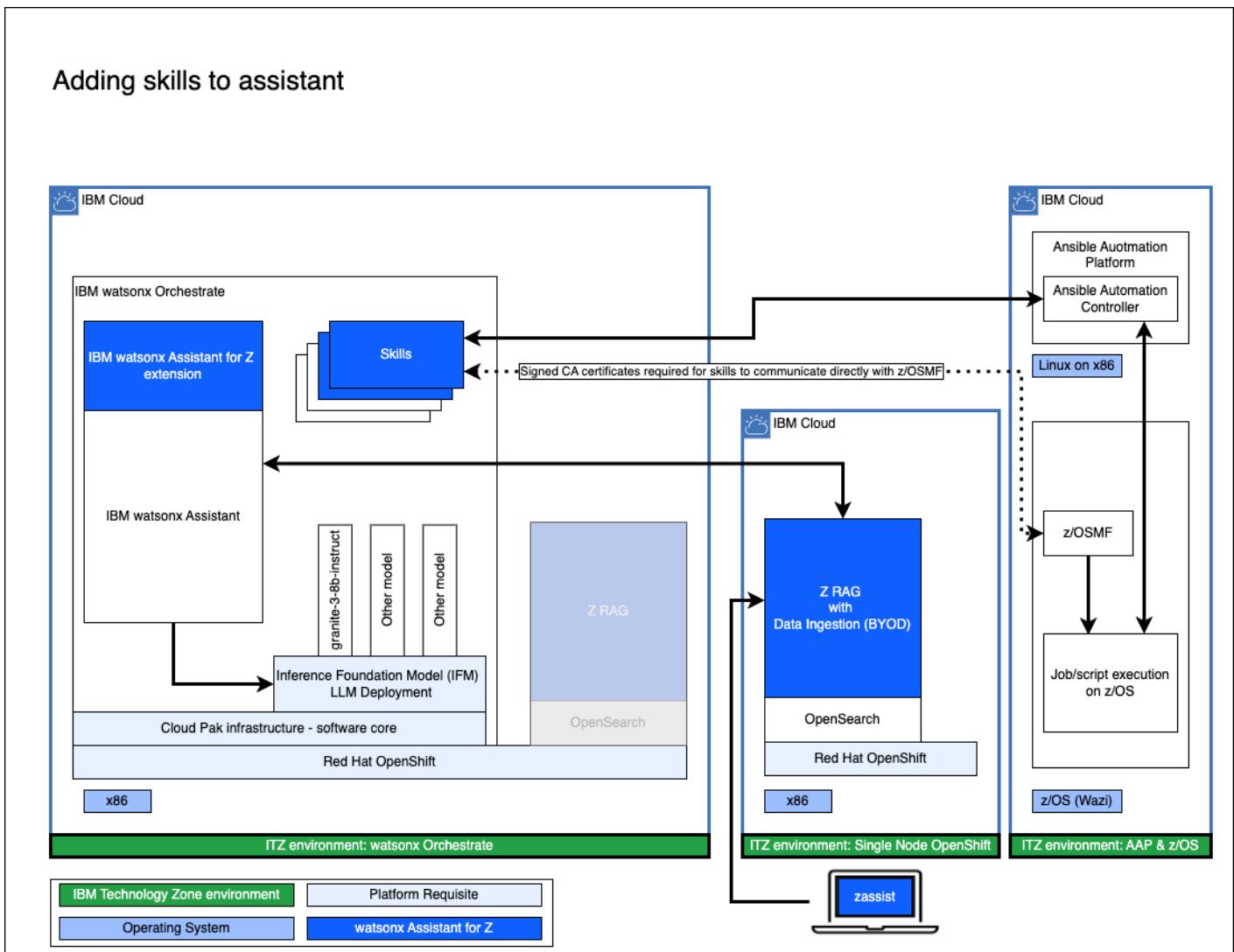
## Skills and skill flows

# 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.

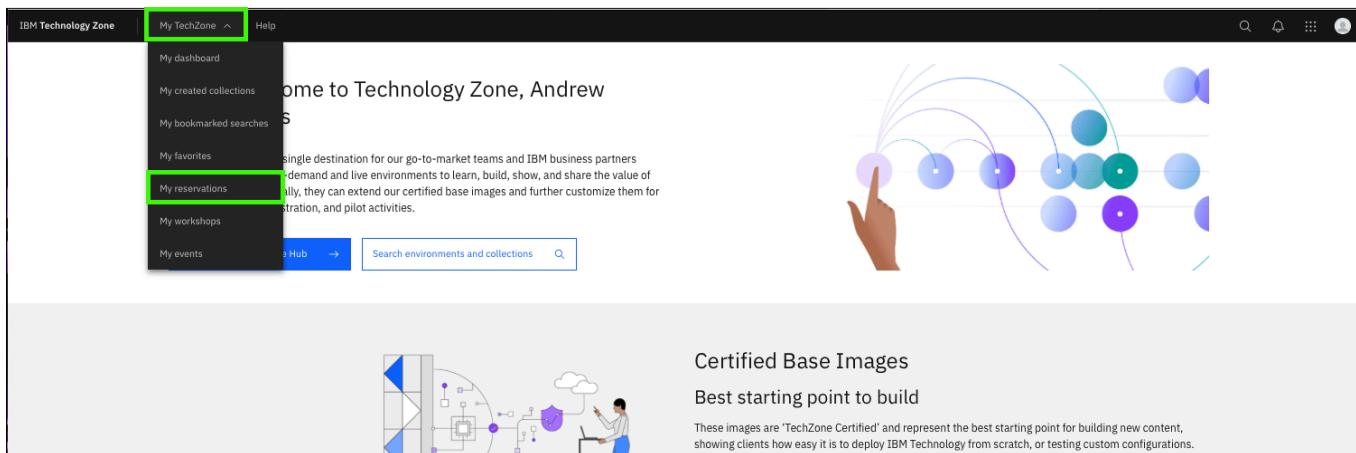
## Access the AAP and Wazi as a Service environment

### 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.

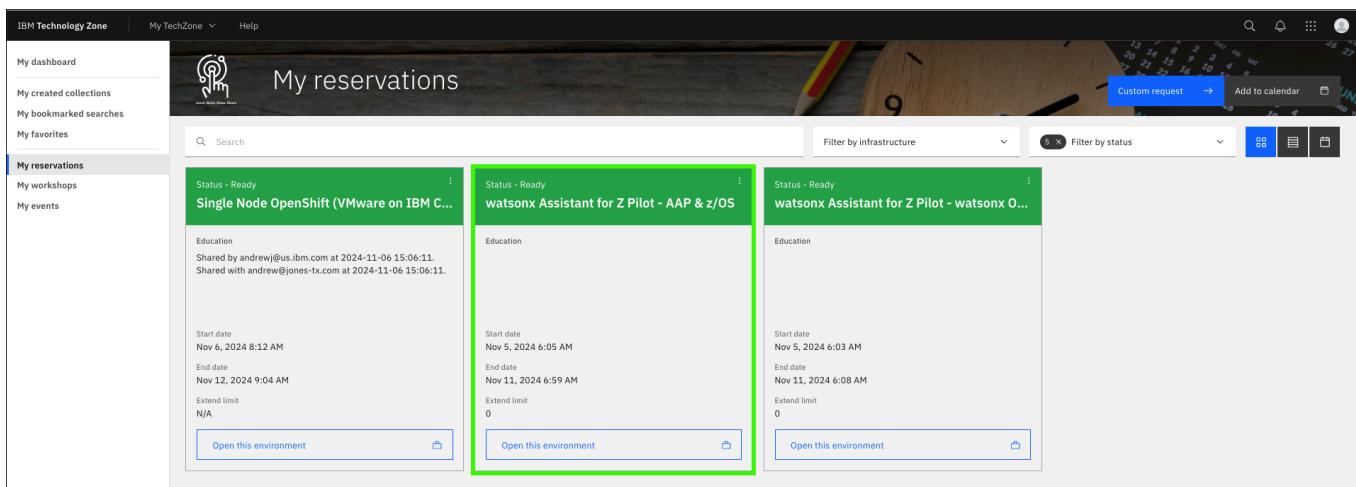
1. In the IBM Technology Zone portal, expand **My TechZone** and select **My Reservations**, or click the following link.

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



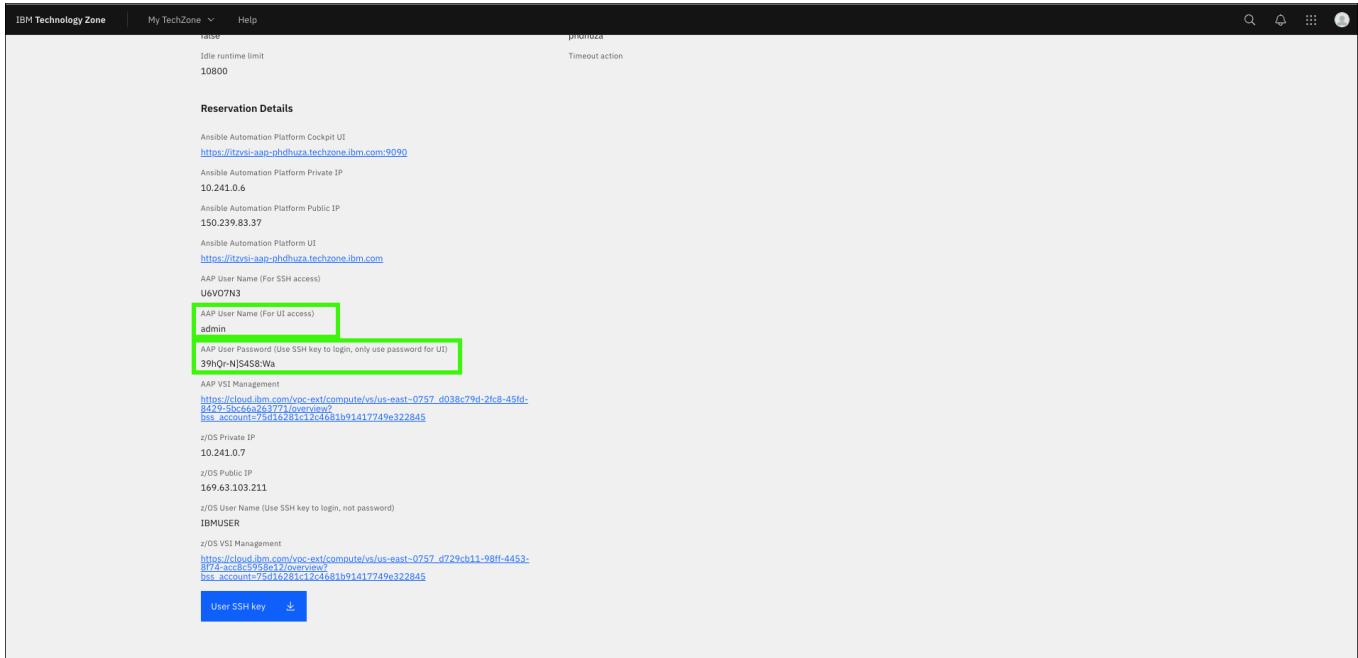
Certified Base Images  
Best starting point to build  
These images are "TechZone Certified" and represent the best starting point for building new content, showing clients how easy it is to deploy IBM Technology from scratch, or testing custom configurations.

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

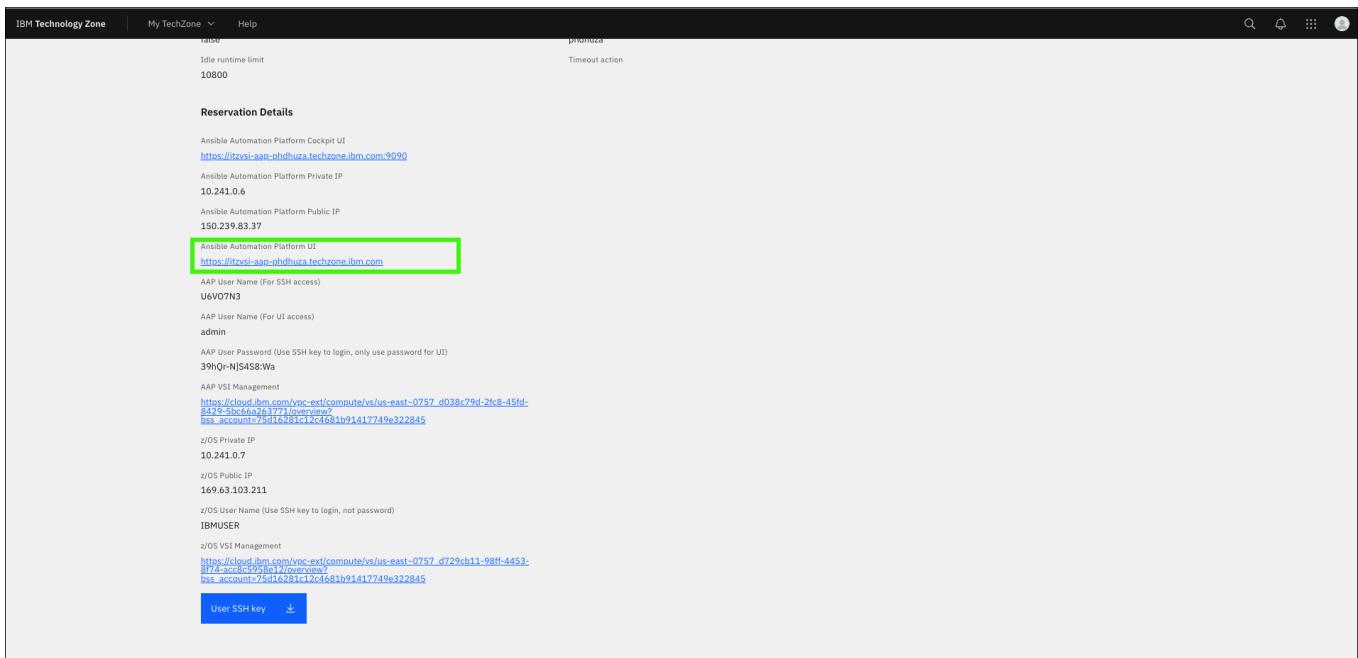


Reservation Type	Status	Description
Single Node OpenShift (VMware on IBM Cloud)	Ready	Shared by andrewj@us.ibm.com at 2024-11-06 15:06:11. Shared with andrewj@us.ibm.com at 2024-11-06 15:06:11.
watsonx Assistant for Z Pilot - AAP & z/OS	Ready	Start date: Nov 5, 2024 6:05 AM. End date: Nov 11, 2024 6:59 AM. Extend limit: 0.

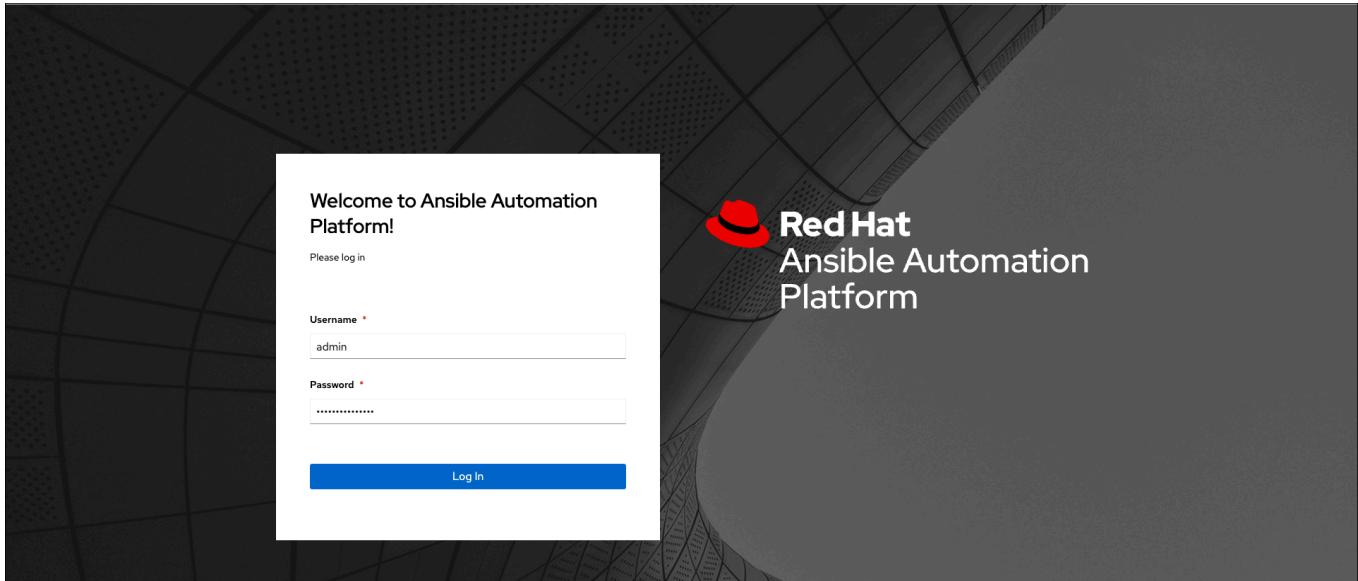
3. Locate and record the **AAP User Name (For UI access)** and **AAP User Password** fields.



#### 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**.



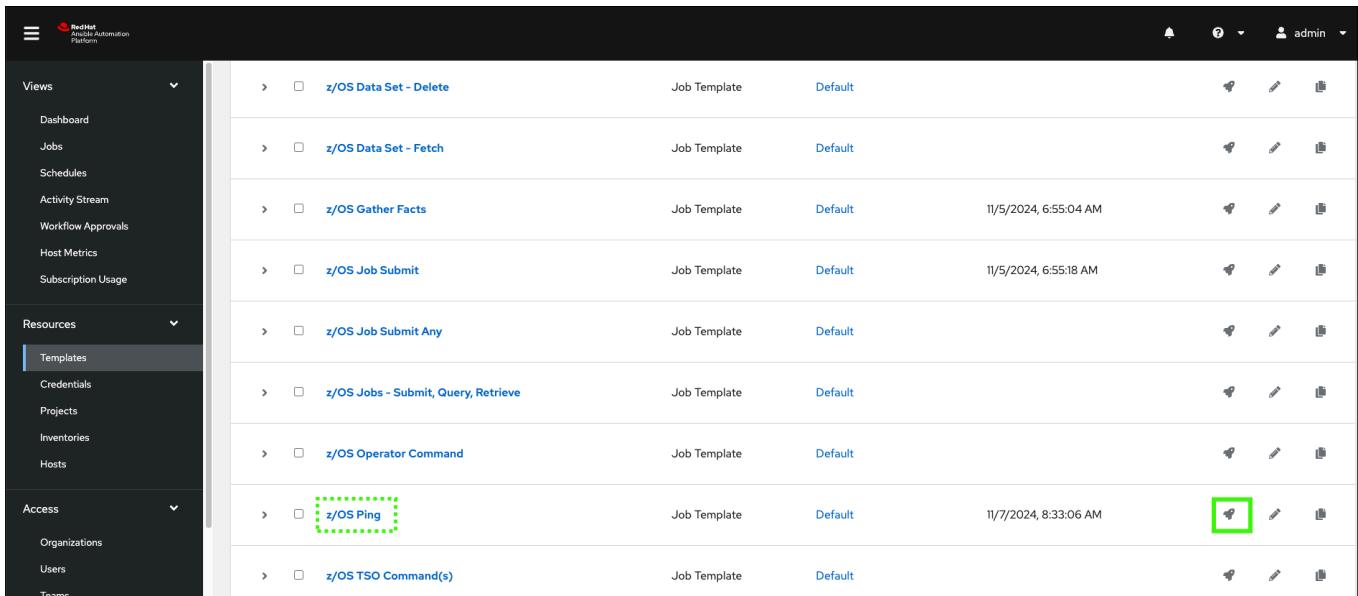
6. Click **Templates** under the **Resources** section.



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

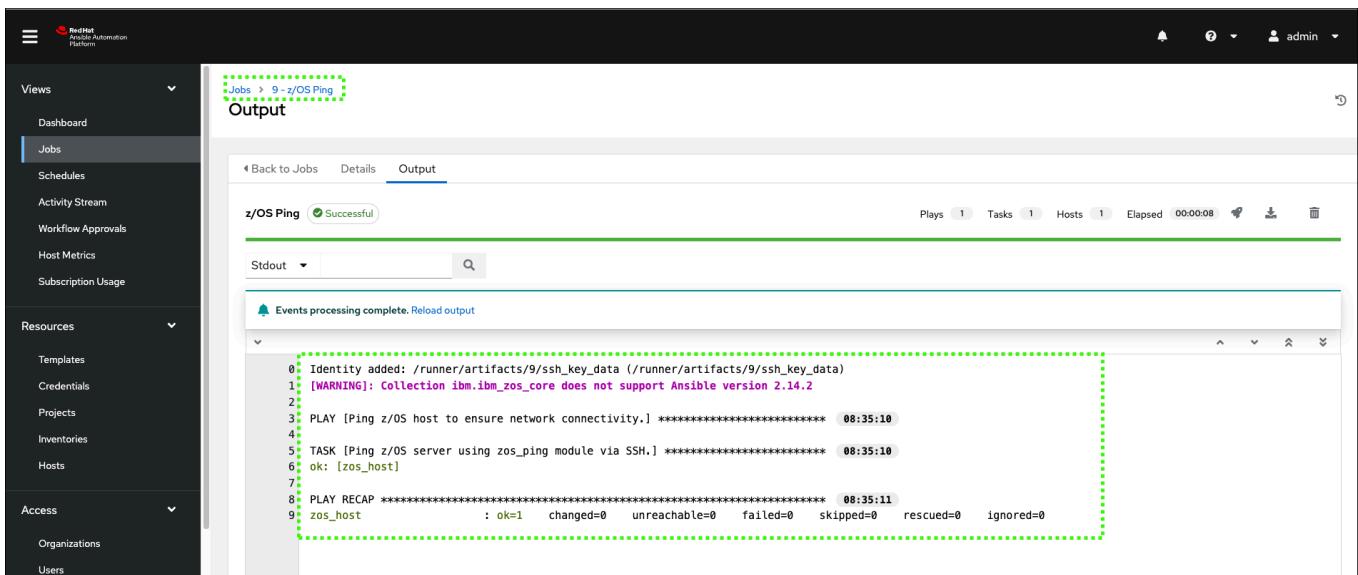
Note that because the AAP instance and the back-end z/OS system are preconfigured, no changes are needed to execute the templates and they will target your provisioned z/OS system by default.

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			
Dashboard	<a href="#">z/OS Data Set - Fetch</a>	Job Template	Default			
Schedules	<a href="#">z/OS Gather Facts</a>	Job Template	Default	1/5/2024, 6:55:04 AM		
Activity Stream	<a href="#">z/OS Job Submit</a>	Job Template	Default	1/5/2024, 6:55:18 AM		
Workflow Approvals	<a href="#">z/OS Job Submit Any</a>	Job Template	Default			
Host Metrics	<a href="#">z/OS Jobs - Submit, Query, Retrieve</a>	Job Template	Default			
Subscription Usage	<a href="#">z/OS Operator Command</a>	Job Template	Default			
Resources	<a href="#">z/OS Ping</a>	Job Template	Default	1/7/2024, 8:33:06 AM		
Templates	<a href="#">z/OS TSO Command(s)</a>	Job Template	Default			
Credentials						
Projects						
Inventories						
Hosts						
Access						
Organizations						
Users						
Teams						

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



Jobs > 9 - z/OS Ping

Output

Events processing complete. Reload output

```

0 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](#).

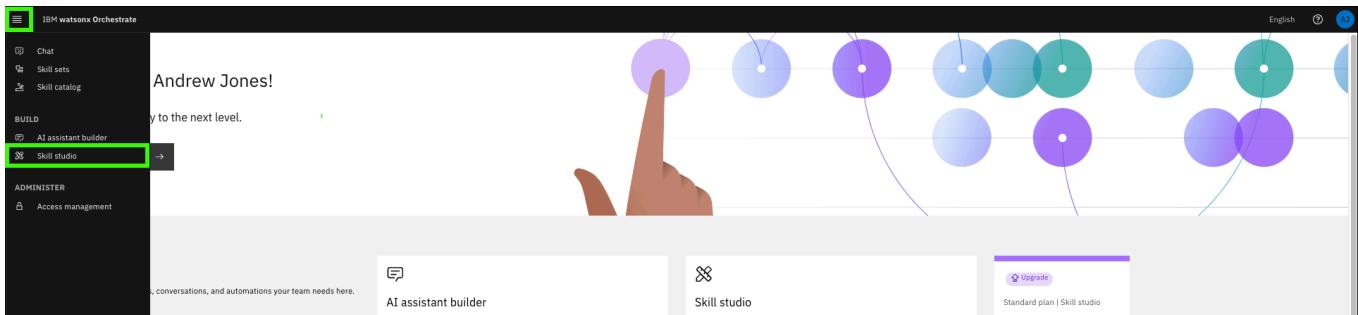
# Importing skills from Ansible Automation Platform

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.

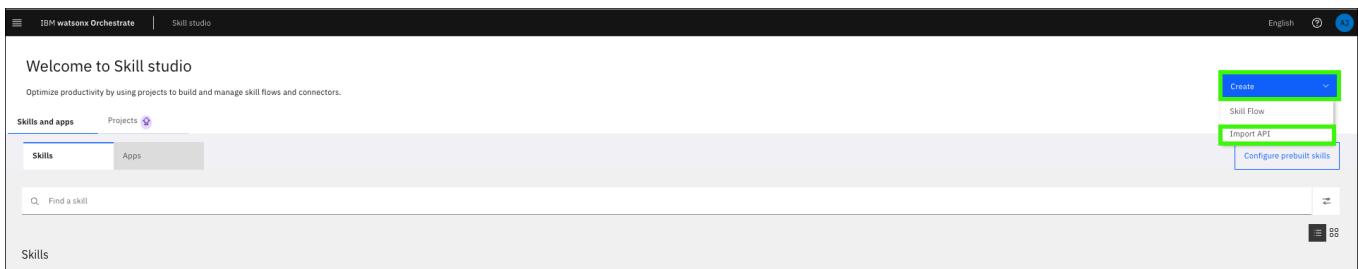
## Import skills into your assistant

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](#).

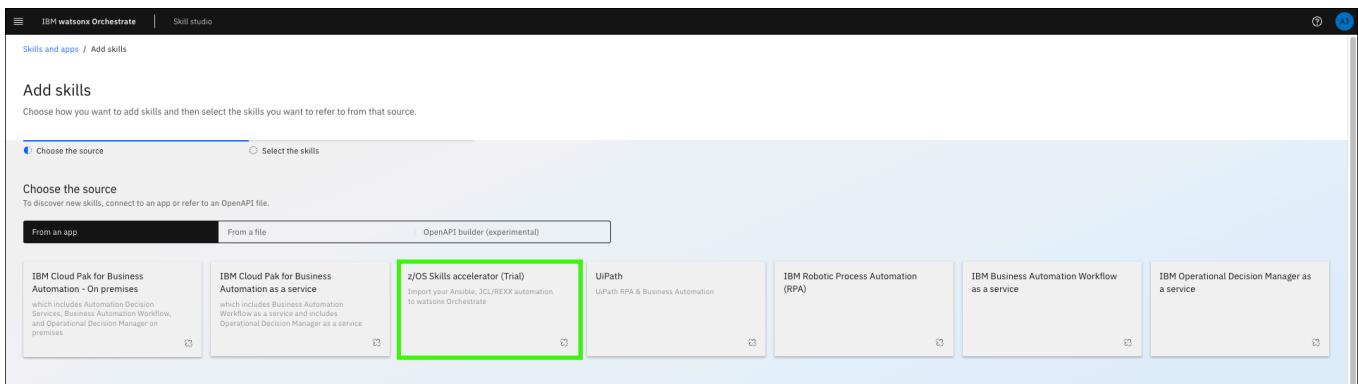
1. Return to your watsonx Orchestrate instance and expand the main menu and click **Skill studio**.



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

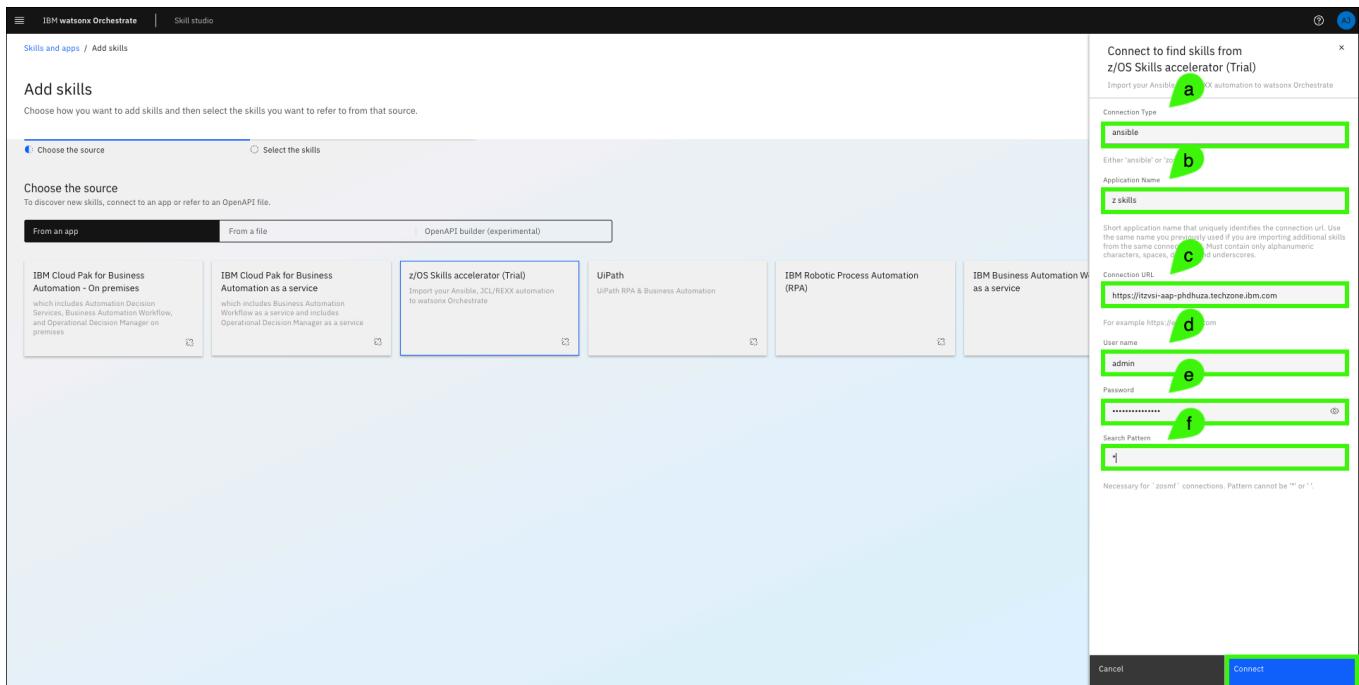
**b:** Application Name: <enter a meaningful name for the skills that you will import> - record this name, you will need in the next section

**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: \*

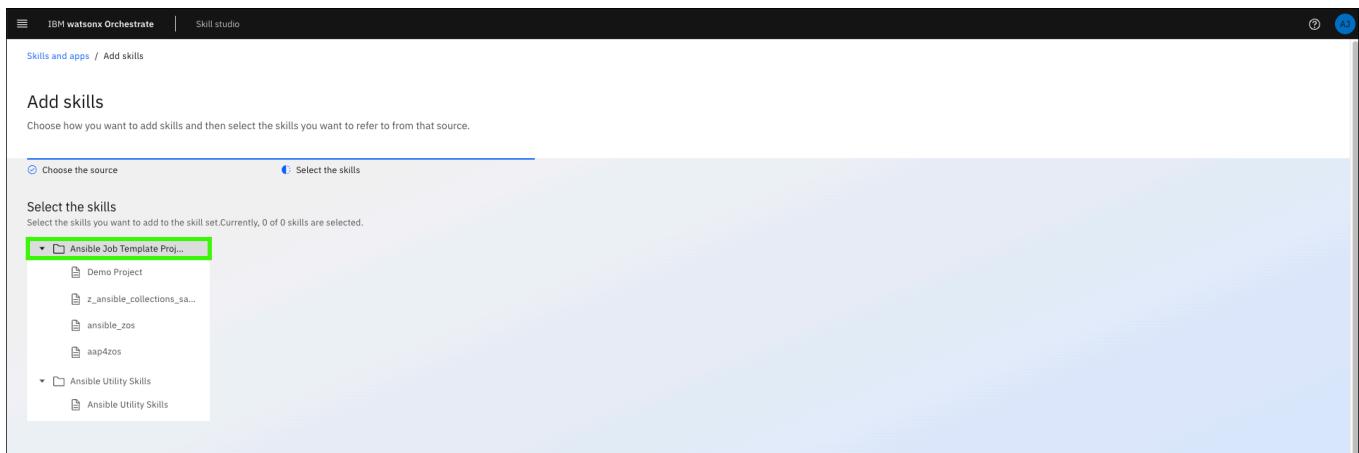


## 5. Expand the Ansible Job Template Proj... folder.



### 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**.

**Select the skills**  
Select the skills you want to add to the skill set. Currently, 0 of 19 skills are selected.

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, ret...	Z skills - this sample playbook...	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.

**Select the skills**  
Select the skills you want to add to the skill set. Currently, 2 of 19 skills are selected.

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 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 se...	Z skills - this playbook enabl...	Ready to add
Z/os certs - create keyring	Z skills - this playbook delet...	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

## 8. Click the ellipses (⋮) for the first 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** Apps

Find a skill

**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 prebuilt 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.

Skills and apps / Enhance this skill

Enhance the "z/OS Gather Facts" 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\*** z/OS Gather Facts

Description 0/100  
z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS-specific information from the z/OS host.

Categories

API version\* 1.0

Preview

The skill will look like this in the catalog.

**z/OS Gather Facts**  
z skills - 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

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

**Skills and apps** Projects

**Skills** Apps

Find a skill

**Skills**

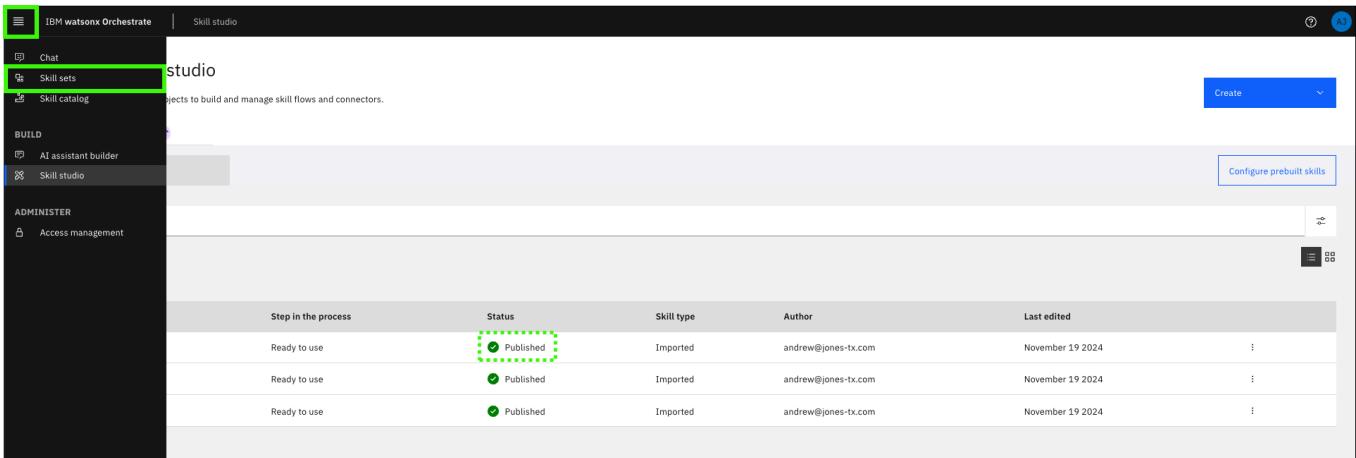
Name	Step in the process	Status	Skill type	Author	Last edited
z/OS Gather Facts	Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
z/OS Ping	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	November 19 2024

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.

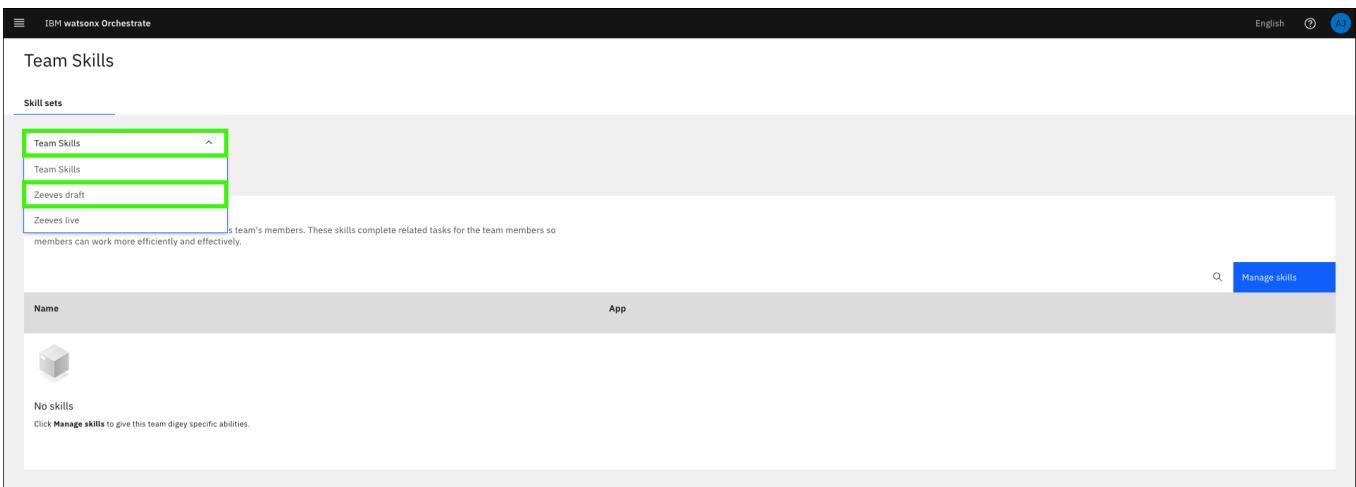
1. Expand the main menu and select **Skill sets**.



The screenshot shows the IBM Watsonx Orchestrate interface. The left sidebar has sections for Chat, Skill sets (which is highlighted with a green box), and Skill catalog. The main area is titled "Skill studio" and contains a "Create" button and a "Configure prebuilt skills" button. Below these are three published skill cards:

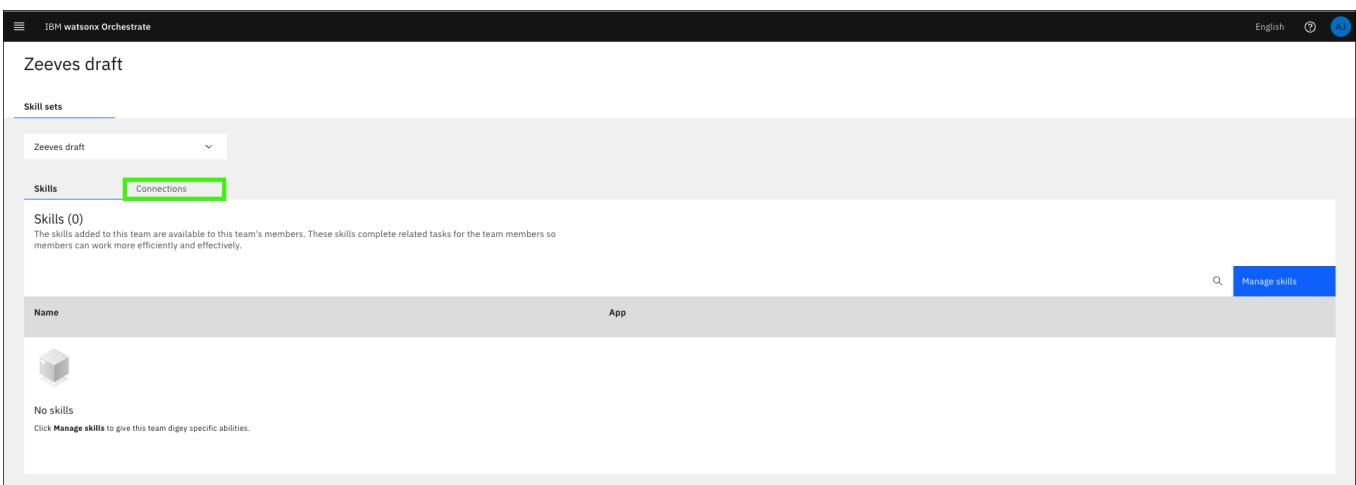
Step in the process	Status	Skill type	Author	Last edited
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	Published	Imported	andrew@jones-tx.com	November 19 2024

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



The screenshot shows the "Team Skills" page. The top navigation bar includes "Skill sets" and "Team Skills". The "Team Skills" dropdown is open, showing "Team Skills" and "Zeeves draft" (which is highlighted with a green box). Below the dropdown is a note about team skills. The main content area shows a table with columns "Name" and "App". There is one row with a placeholder icon and the text "No skills". A "Manage skills" button is located at the bottom right.

3. Click the **Connections** tab.



The screenshot shows the "Zeeves draft" team page. The top navigation bar includes "Skill sets" and "Connections" (which is highlighted with a green box). The main content area shows a note about team skills and a table with columns "Name" and "App". There is one row with a placeholder icon and the text "No skills". A "Manage skills" button is located at the bottom right.

4. Click the **Search (🔍)** icon.

Application	Number of skills	Credential type	Connected by ⓘ	Action
Activate or deactivate attracting candidates using ThisWay Global	4	⚠️ Not specified	-	⋮
Adobe Workfront	37	⚠️ Not specified	-	⋮
Alliance Virtual Office	2	⚠️ Not specified	-	⋮
Amazon S3	8	⚠️ Not specified	-	⋮
Amazon SES	10	⚠️ Not specified	-	⋮

Items per page: 5 | 1-5 of 78 items | 1 ⚏ of 16 pages | ⌂ ⌃

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	-	⋮

Items per page: 5 | 1-1 of 1 items | 1 ⚏ of 1 page | ⌂ ⌃

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	-	⋮

Items per page: 5 | 1-1 of 1 items | Connect app | ⌂ ⌃

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

Zeeves draft

Skill sets

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.

Q z skills

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

Items per page: 5 1-1 of 1 items

Connect to Ansible Controller Skills - z skills

Member credentials  
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

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)**.

Zeeves draft

Skill sets

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.

Q z skills

Ansible Controller Skills - z skills

Items per page: 5 1-1 of 1 items

Connect to Ansible Controller Skills - z skills

a  
admin

b  
\*\*\*\*\*

If the service instance uses legacy credentials for authentication, provide the password for the specified username.

Cancel Connect app c

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

Zeeves draft

Skill sets

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.

Q z skills

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	2	Team	andrew@jones-tx.com	⋮

Items per page: 5 1-1 of 1 items

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](#).

The screenshot shows the IBM Watsonx Orchestrate interface. In the top right corner, there is a user profile icon labeled 'AJ' with a green box around it, labeled 'a'. Below the profile icon is a sidebar with several options: 'Andrew Jones', 'andrew@jones-tx.com', 'Region US-SOUTH', 'Plan Essentials', 'Provide feedback', 'Privacy', 'About', and 'Log out'. The 'Settings' option is highlighted with a green box, labeled 'b'. The main content area shows a 'Skill sets' section with a dropdown menu set to 'Zeeves draft'. Below this is a table with columns: 'Skills' (containing 'Ansible Controller Skills - z skills'), 'Number of skills' (containing '2'), 'Credential type' (containing 'Team'), and 'Connected by' (containing 'andrew@jones-tx.com'). At the bottom of the page, there are pagination controls: 'Items per page: 5', '1-1 of 1 items', and '1 of 1 page'.

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

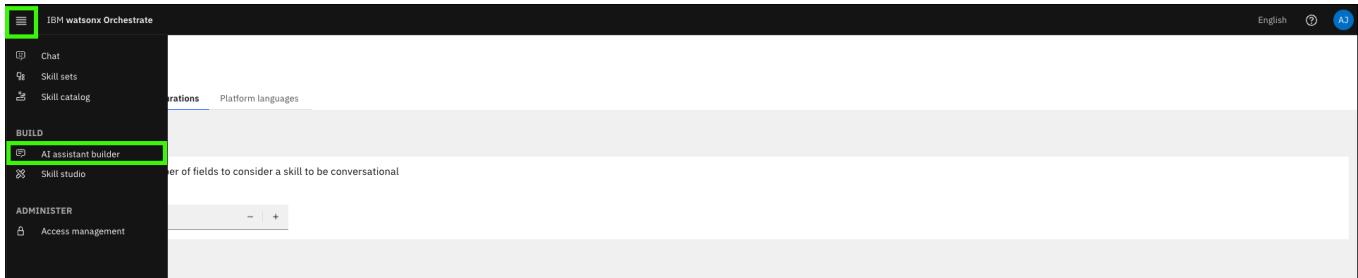
The screenshot shows the 'Settings' page in IBM Watsonx Orchestrate. The 'Skill configurations' tab is selected and highlighted with a green box. The page displays the 'Data Retention Policy' section, which is active. It includes a note: 'The chat history for the users in the tenant are saved for 30 days. After 30 days, the chat history is deleted permanently. Deleting the chat history doesn't affect the connected apps or the added skills.' There are also tabs for 'Data Retention' and 'Platform languages'.

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

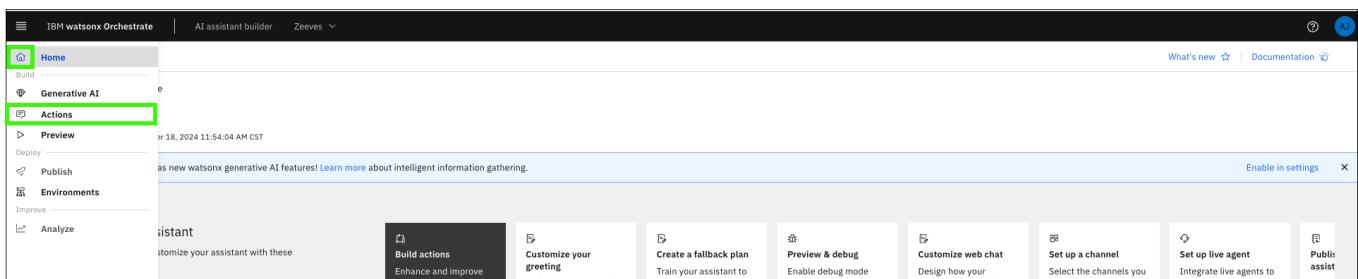
The screenshot shows the 'Skill configurations' page in IBM Watsonx Orchestrate. The 'Number of form fields' input field is highlighted with a green box and contains the value '0'. There is also a minus sign '-' and a plus sign '+' button next to the input field.

## Create actions

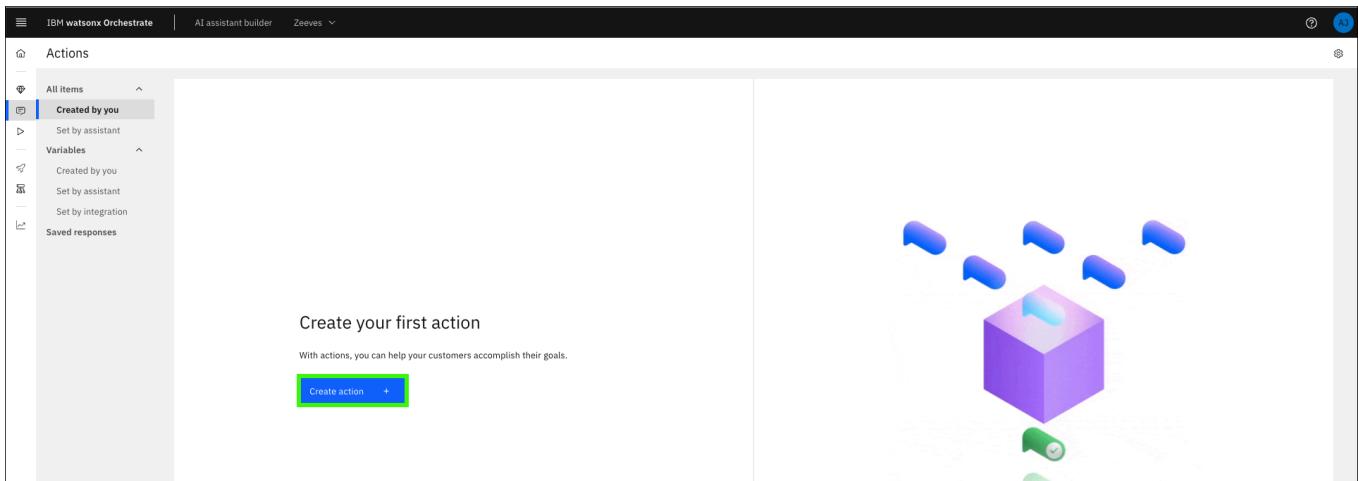
1. Click the main menu and select **AI assistant builder**.



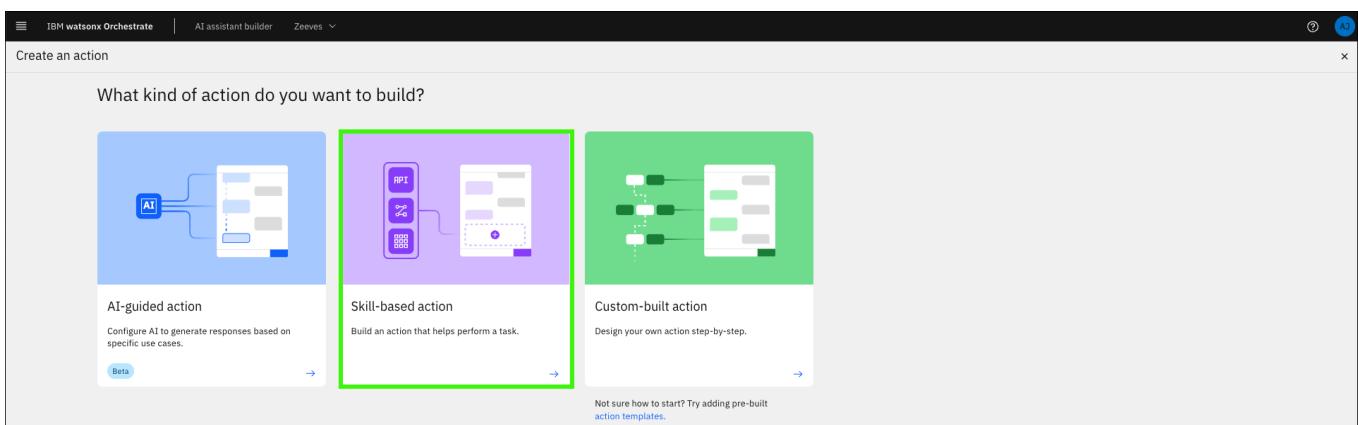
2. Hover over the **Home** icon (🏠) and click **Actions**.



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.

The screenshot shows a list of skills under the heading "Build an action from a skill". The "z/OS Gather Facts" skill is selected and highlighted with a green border. The "Next" button at the top right is also highlighted with a green box.

z/OS Ping	z/OS Gather Facts	Summarize the Webex meeting transcript	Summarize the Box content	Summarize a Zendesk ticket
z skills - This playbook pings the z/OS host to test connectivity. Last updated: 2024-11-19T15:58:20.567Z	z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS-specific information from the z/OS host. Last updated: 2024-11-19T15:56:26.843Z	in watsonx.ai Last updated: 2024-11-04T10:49:16.502Z	in watsonx.ai Last updated: 2024-11-04T10:49:12.077Z	in watsonx.ai Last updated: 2024-11-04T10:49:09.476Z
Summarize a ServiceNow incident in watsonx.ai Last updated: 2024-11-04T10:49:05.828Z	Summarize a Salesforce opportunity in watsonx.ai Last updated: 2024-11-04T10:49:01.769Z	Sharepoint document summary in watsonx.ai Last updated: 2024-11-04T10:48:55.707Z	Salesloft email summary in watsonx.ai Last updated: 2024-11-04T10:48:51.522Z	Salesforce case summarization in watsonx.ai Last updated: 2024-11-04T10:48:48.195Z
Salesforce case sentiment analyze	Outlook email summary	Github issue summarization	Github issue sentiment	Generate an email

- 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**.

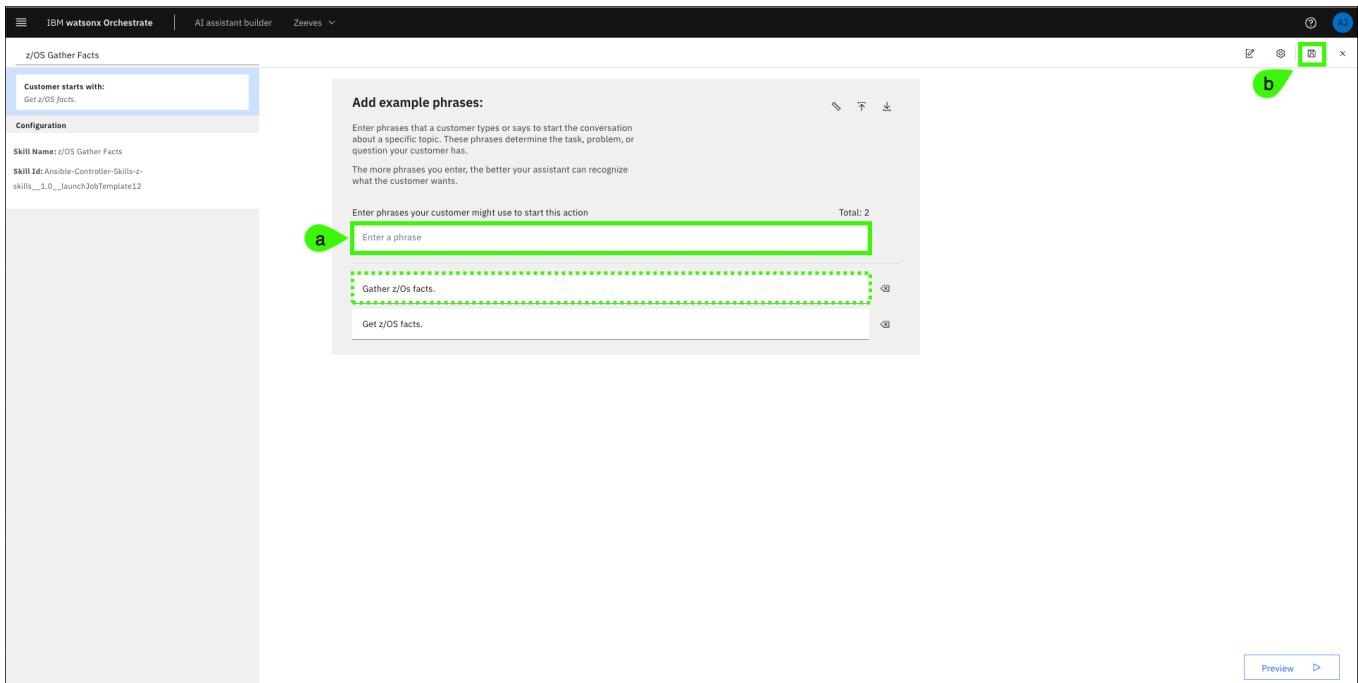
Sample prompts:

Get z/OS facts

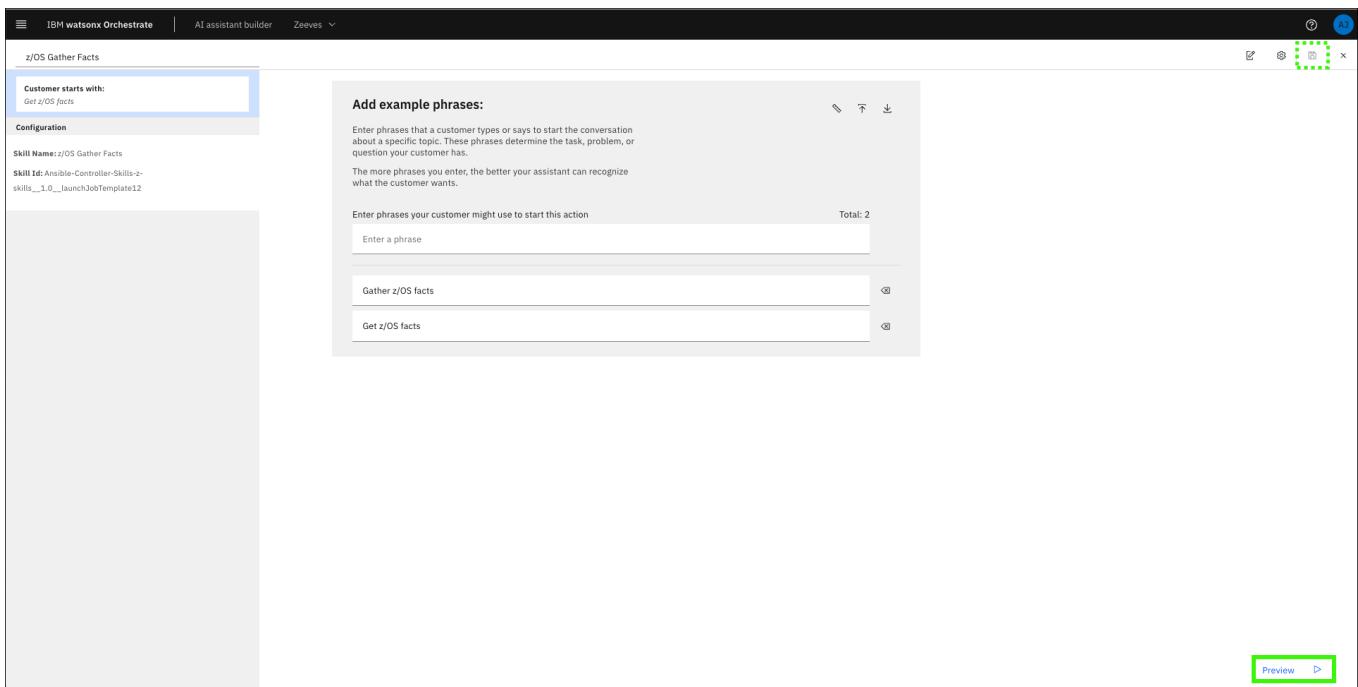
Gather z/OS facts

The screenshot shows the configuration screen for the "z/OS Gather Facts" skill. A modal window titled "New action" is open, prompting the user to enter example phrases. The input field contains "Get z/OS facts." and the "Save" button is highlighted with a blue box and a green box around the entire modal.

- 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

The screenshot shows the configuration interface for the 'z/OS Gather Facts' skill. On the left, there's a sidebar with 'Customer starts with:' set to 'Get z/OS facts'. The main area has a 'Configuration' section with 'Skill Name: z/OS Gather Facts' and 'Skill Id: Ansible-Controller-Skills-z-skills\_1.0.\_launchJobTemplate12'. Below this is an 'Add example phrases:' section where two phrases are listed: 'Gather z/OS facts' and 'Get z/OS facts'. To the right is a 'Preview' window showing a conversation: 'User: Get z/OS facts' and 'Assistant: 12:55 PM Greet customer [default] Welcome, how can I assist you?'. A green box highlights the user message '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.

This screenshot is identical to the one above, showing the 'z/OS Gather Facts' skill configuration. The 'Customer starts with:' field is set to 'Get z/OS facts'. The 'Add example phrases:' section contains 'Gather z/OS facts' and 'Get z/OS facts'. The 'Preview' window shows the same conversation: 'User: Get z/OS facts' and 'Assistant: 12:55 PM Greet customer [default] Welcome, how can I assist you?'. However, the preview now includes a history of messages: 'You 1:04 PM Get z/OS facts' (highlighted with a green dashed box) and 'You 1:05 PM Conversational skill called z/OS Gather Facts recognized' (highlighted with a green dashed box). Below these, a message box contains 'job: 12 status: pending' and a note: 'There are no additional steps for this action. Add a new step or end the action.' A green dashed box also surrounds this message area. At the bottom of the preview window, it says 'Use the up arrow for prior messages'.

# 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.

The screenshot shows the 'Jobs' page in the Red Hat Ansible Automation Platform. The left sidebar has a 'Jobs' menu item highlighted with a green box. In the main list, a job named '12 - z/OS Gather Facts' is also highlighted with a green box. The job details show it is 'Successful', a 'Playbook Run', started at 12/5/2024, 7:17:32 AM, and finished at 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.

The screenshot shows the 'Output' tab for job '12 - z/OS Gather Facts'. The output content is highlighted with a large green dotted box. The output itself shows an Ansible task for printing gathered facts about a z/OS host, with various system parameters listed.

```

25 }
26 TASK [Print out all gathered facts about the z/OS host.] **** 07:17:38
27 ok: [zos_host] => {
28     "ansible_facts": {
29         "arch_level": "2",
30         "cpc_nd_manufacturer": "IBM",
31         "cpc_nd_model": "A00",
32         "cpc_nd_plant": "C1",
33         "cpc_nd_seqno": "237701828347",
34         "cpc_nd_type": "008562",
35         "edt": "00",
36         "hw_name": "...",
37         "ieasym_card": "(00,K2)",
38         "io_config_id": "00",
39         "iodate": "...",
40         "iodesc": "...",
41         "iodf_config": "DEFAULT",
42         "iodf_name": "PROV.IODF00",
43         "iodf_unit_addr": "DE28",
44         "ioproc": "...",
45         "iotime": "...",
46

```

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.

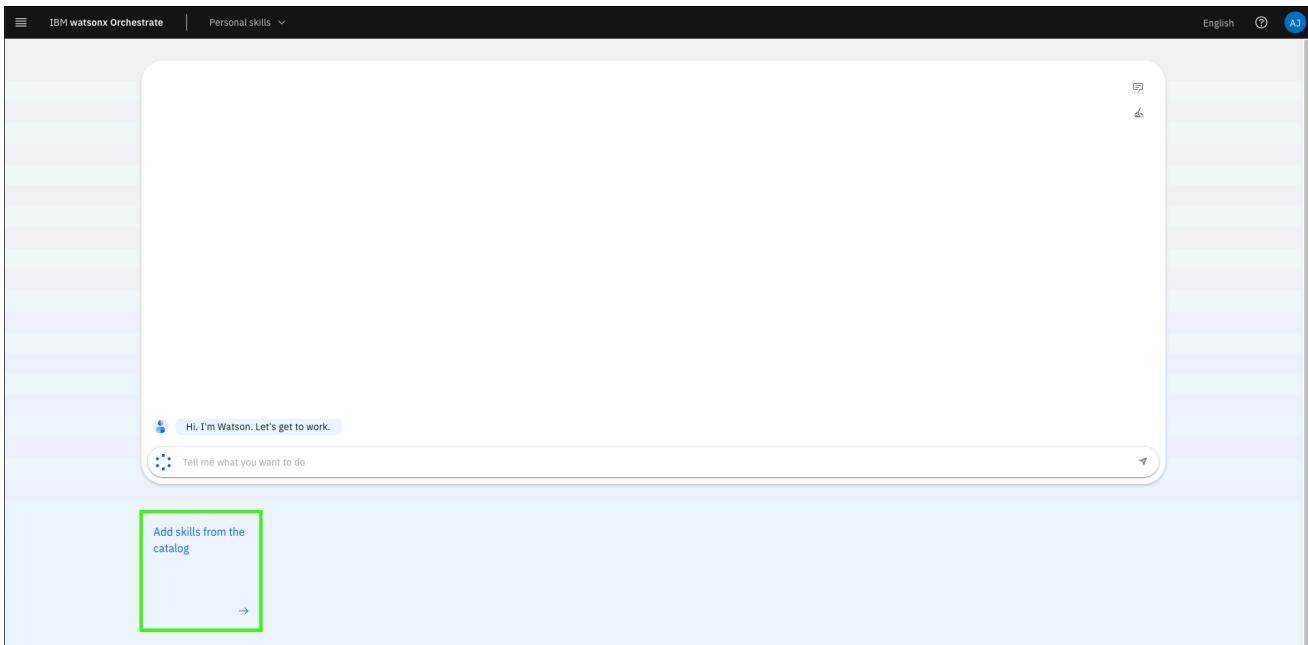
The screenshot shows the AI assistant builder interface. On the left, there's a sidebar with sections like Configuration, Customer starts with:, and Add example phrases:. Under Configuration, the Skill Name is set to "z/OS Gather Facts" and the Skill Id is "Ansible-Controller-Skills-z-skills\_1.0\_launchJobTemplate1". The "Customer starts with:" field contains "Get z/OS facts". The "Add example phrases:" section has two entries: "Gather z/OS facts" and "Get z/OS facts". On the right, there's a "Preview" window showing a simulated chat session. The customer says "Welcome, how can I assist you?", and the bot responds with "Gather z/OS facts". In the preview window, a green dashed box highlights the bot's response. Below it, an error message is shown: "Sorry, we're having issues generating a response." This message is enclosed in a yellow box with an exclamation mark icon. Another yellow box below it contains the text "Error in the response" with a "Details" link. A third yellow box contains the text "Skill error" with an "Inspect" link. At the bottom of the preview window, a note says "There are no additional steps for this action. Add a new step or end the action."

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.

The screenshot shows the main menu of IBM Watsonx Assistant. The "Chat" option is highlighted with a green box. Other menu items include "Skill sets", "Skill catalog", "AI assistant builder", "Skill studio", "Access management", and "Administrator". The "AI assistant builder" section is expanded, showing "Customer starts with:", "Add example phrases:", and a "Preview" window.

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. At the top, there's a search bar with the text 'z skills'. Below it, there's a grid of 'Apps'. One specific app, 'Ansible Controller Skills - z skills', is highlighted with a green box. 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 details for the 'Ansible Controller Skills - z skills' app. It lists two skills: 'z/OS Gather Facts' and 'z/OS Ping'. Each skill card has an 'Add skill +' button at the bottom, which is highlighted with a green box.

5. Click **Connect app**.

Ansible Controller Skills - z skills (2)

**z/OS Gather Facts**  
z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS...  
**z/OS Ping**  
z skills - This playbook pings the z/OS host to test connectivity.

**Connect app**

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**.

Connect to Ansible Controller Skills - z skills

username: admin  
password: \*\*\*\*\*

Cancel Connect app

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

Skills - z skills

**z/OS Ping**  
z skills - This playbook pings the z/OS host to test connectivity.

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 settings for "English" and a blue "Aj" button. 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 green button labeled "Gather z/OS facts" is highlighted. A green dashed box encloses the results of this skill: "z/OS Gather Facts", "status pending", and "job 12". At the bottom, there's a blue button to "Tell me what you want to do". To the left, a box says "Add skills from the catalog" with an arrow pointing to another box labeled "Ansible Controller Skills - z skills" which contains "2 skills".

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

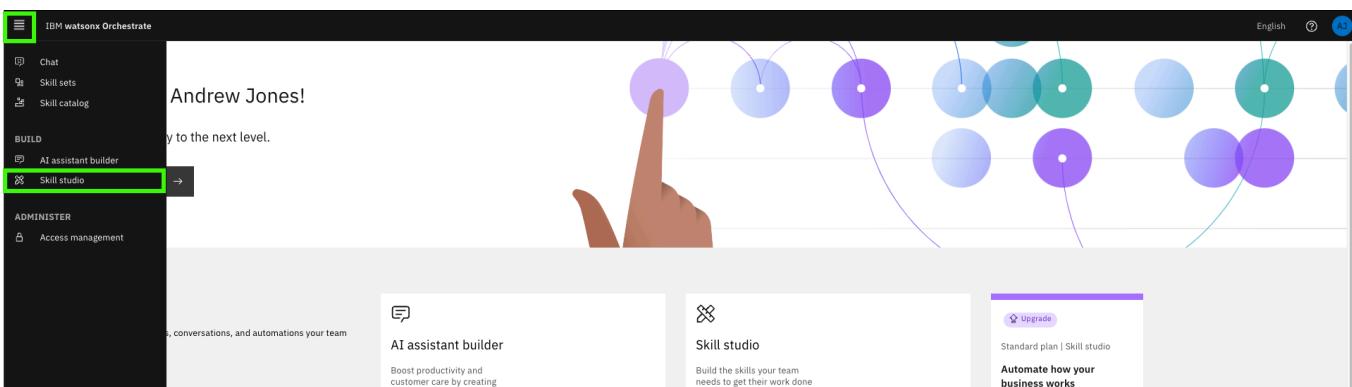
# Creating skill flows

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](#).

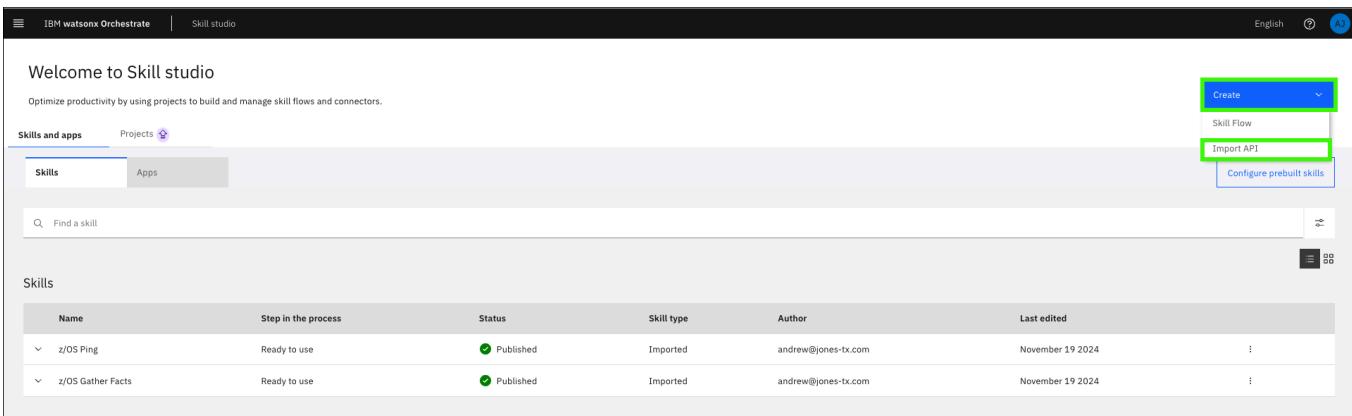
As mentioned in a previous section, default utility skills that are provided with the watsonx Assistant for Z skills collection. The **Retrieve job output** utility skill is used to return the output of a skill.

## Add the utility skill

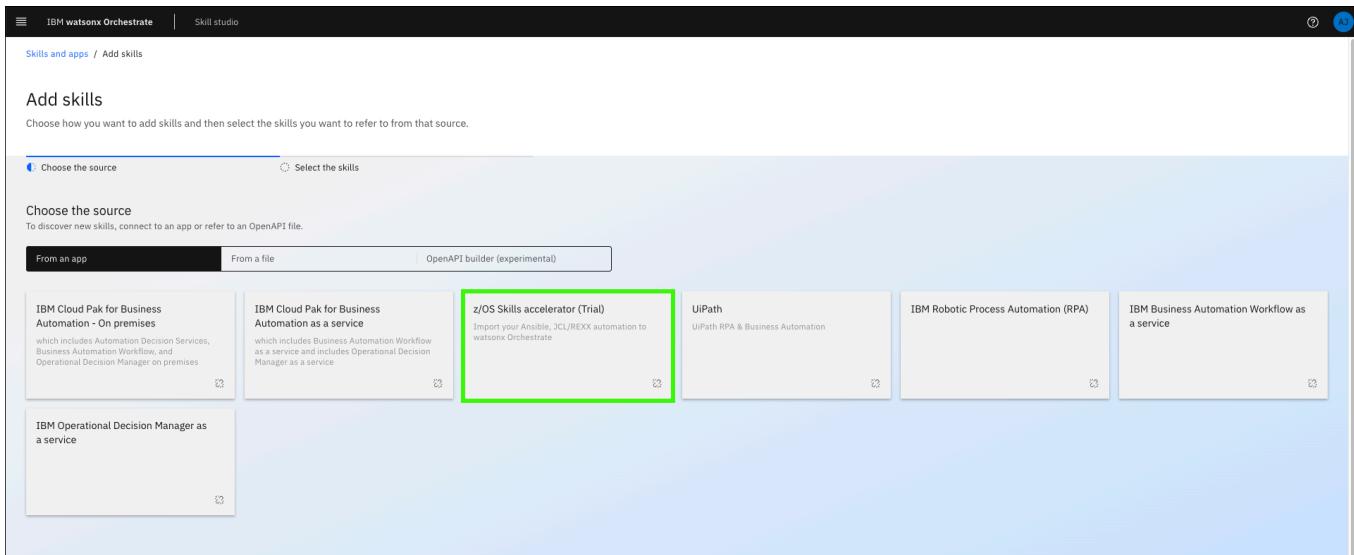
### 1. Open IBM watsonx Orchestrate **Skill studio**.



### 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`

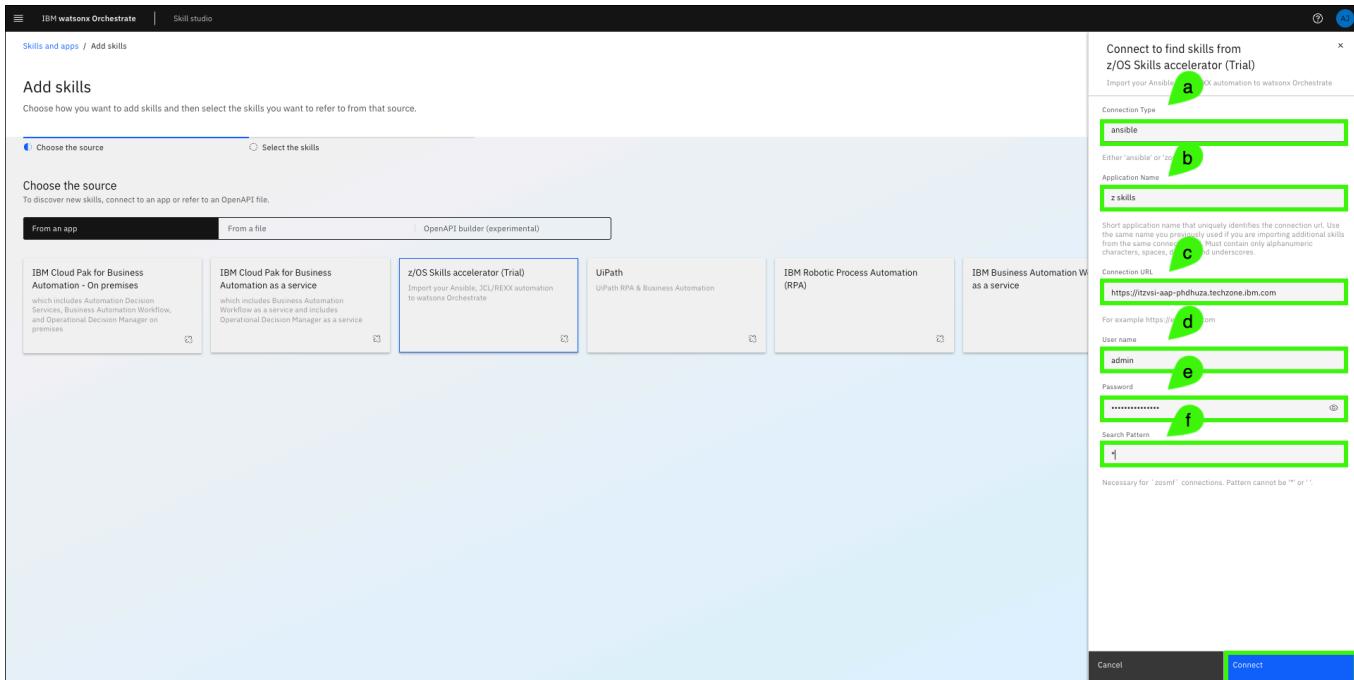
**b:** Application Name: <use the same application name from the previous section>

**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.

Choose the source       Select the skills

**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
Retrieve job output	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.

Choose the source       Select the skills

**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
<input checked="" type="checkbox"/> Retrieve job output	Z skills - retrieve job output...	Ready to add

**Cancel**      **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**      Apps

Find a skill

**Skills**

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

**1 skill successfully imported**  
success  
14:06:04

**Configure prebuilt skills**

**Enhance this skill**

**Export this skill**

**Delete this skill**

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

Name: Retrieve job output  
Input: 0/100  
Output: 0/100  
Security: None  
Phrases: None  
Next best skills: None  
API version: 1.0  
Categories: None  
App: Ansible Controller Skills - z skills

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

Step in the process	Status	Skill type	Author	Last edited
Ready to use	<span style="color: green;">Published</span>	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	<span style="color: green;">Published</span>	Imported	andrew@jones-tx.com	November 19 2024
Ready to use	<span style="color: green;">Published</span>	Imported	andrew@jones-tx.com	November 19 2024

## 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 attracting candidates using ThisWay Global	4	Not specified	-	⋮
Adobe Workfront	37	Not specified	-	⋮
Alliance Virtual Office	2	Not specified	-	⋮
Amazon S3	8	Not specified	-	⋮
Amazon SES	10	Not specified	-	⋮

11. Click the **Search (🔍)** icon.

Application	Number of skills	Credential type	Connected by	Action
Activate or deactivate attracting candidates using ThisWay Global	4	⚠️ Not specified	-	⋮
Adobe Workfront	37	⚠️ Not specified	-	⋮
Alliance Virtual Office	2	⚠️ Not specified	-	⋮
Amazon S3	8	⚠️ Not specified	-	⋮
Amazon SES	10	⚠️ Not specified	-	⋮

Items per page: 5 | 1-5 of 78 items | 1 ⚏ of 16 pages | ⏪ ⏩

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	-	⋮

Items per page: 5 | 1-1 of 1 items | 1 ⚏ of 1 page | ⏪ ⏩

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

Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	3	⚠️ Team	andrew@jones-tx.com	⋮

a  
b  
Edit connection  
Delete connection

Items per page: 5 | 1-1 of 1 items | 1 ⚏ of 1 page | ⏪ ⏩

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.  
\*\*\*\*\*

Zeeves draft

Skill sets

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
Ansible Controller Skills - z skills	3	Team

Items per page: 5 | 1-1 of 1 items

Edit the Ansible Controller Skills - z skills connection **b**

Member credentials  
Each team member uses their own credentials to connect to this app and use its skills.

Team credentials (Active)  
The admin sets the credentials each team member uses to connect to this app and use its skills.

You have an active connection set using **Team credentials**. If you wish to update connection details click on the **Edit** **a** below.

**a** Connected **b**

## Add the skills to your Personal skills

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

IBM watsonx Orchestrate

Chat

Skill sets

**Skill catalog**

BUILD

AI assistant builder

Skill studio

ADMINISTER

Access management

These 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.

Number of skills	Credential type	Connected by <b>a</b>	Action
3	Team	andrew@jones-tx.com	<b>a</b>

1 of 1 items

2. Search for the application name you specified earlier.

Skill catalog

Skills are grouped by app. Select an app to see all the skills that use that app.

**a** Personal skills

**a** z skills

Most popular skills

Send an email from Gmail	Create a lead in Salesforce	Send an email using Outlook
--------------------------	-----------------------------	-----------------------------

All Apps

Coupa	ZoomInfo	HubSpot CRM	Apptio Targetprocess	Salesforce	Zendesk Service
Calendly	Square	Oracle E-Business Suite	GitLab	Toggl Track	Microsoft Teams

3. Click the tile for your application.

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

Skill catalog

Skills are grouped by app. Select an app to see all the skills that use that app.

Personal skills

z skills

Apps

- ZoomInfo (26 skills)
- Workday HCM (36 skills)
- FreshService (25 skills)
- Seismic (22 skills)
- Reveal your existing applic... (7 skills)
- Interview top candidates u... (2 skills)
- Salesforce Chatter (5 skills)
- IBM Process Mining (2 skills)
- Webex (14 skills)
- Adobe Workfront (37 skills)
- Skill flows (167 skills)
- Cognos (8 skills)
- Salessoft (47 skills)
- Ansible Controller Skills - z ... (3 skills)**

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

Skill catalog / Ansible Controller Skills - z skills (3)

Personal skills

Search skills

Ansible Controller Skills - z skills

- Retrieve job output (z skills - Retrieve job output by job Id)
- z/OS Gather Facts (z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls...)
- z/OS Ping (z skills - This playbook pings the z/OS host to test connectivity.)

Connect app

## Create the skill flow

- Click **Skill studio** in the main menu.

IBM Watsonx Orchestrate

English

Chat

Skill sets

Skill catalog

BUILD

AI assistant builder

**Skill studio**

ADMINISTER

Access management

Skills - z skills (3)

z skills

- z/OS Gather Facts (z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls...)
- z/OS Ping (z skills - This playbook pings the z/OS host to test connectivity.)

Added ✓

Added ✓

Added ✓

Connect app

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

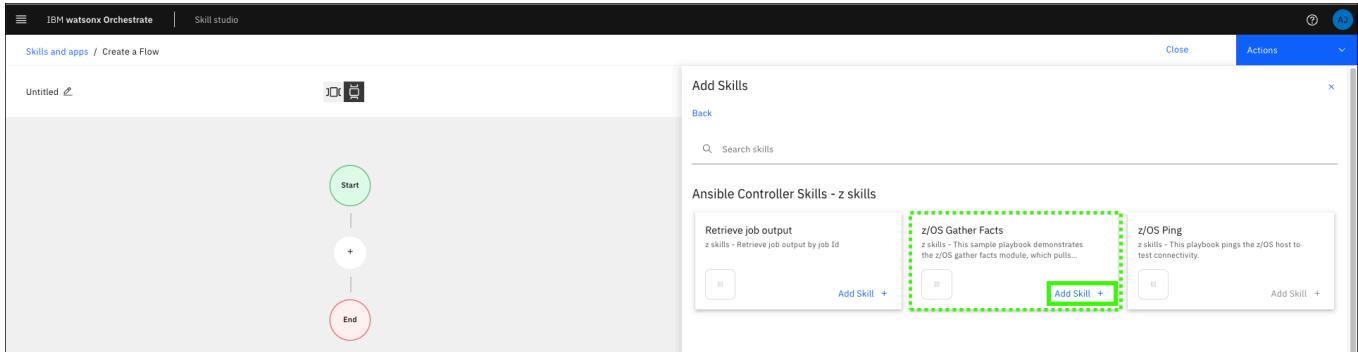
3. Click the + icon.

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.

Skill Name	Number of Skills
ZoomInfo	26 skills
Workday HCM	36 skills
FreshService	25 skills
Seismic	22 skills
Reveal your existing applic...	7 skills
Interview top candidates u...	2 skills
Salesforce Chatter	5 skills
IBM Process Mining	2 skills
Webex	14 skills
Adobe Workfront	37 skills
Skill flows	167 skills
Cognos	8 skills
Salesloft	47 skills
<b>Ansible Controller Skills - ...</b>	3 skills

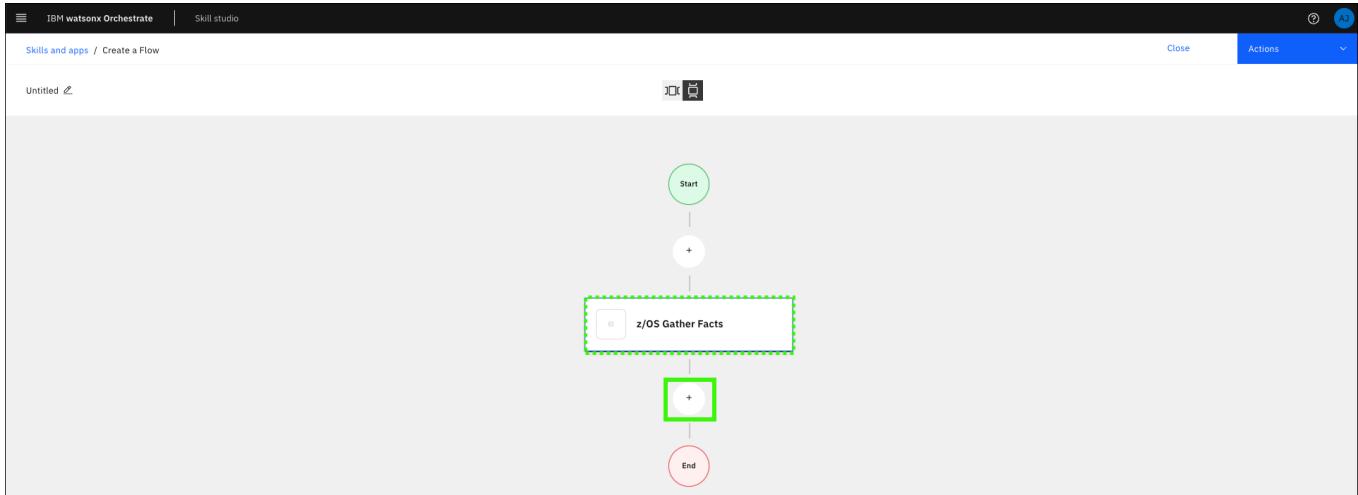
5. Click **Add Skill** in the [z/OS Gather Facts](#) tile.



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

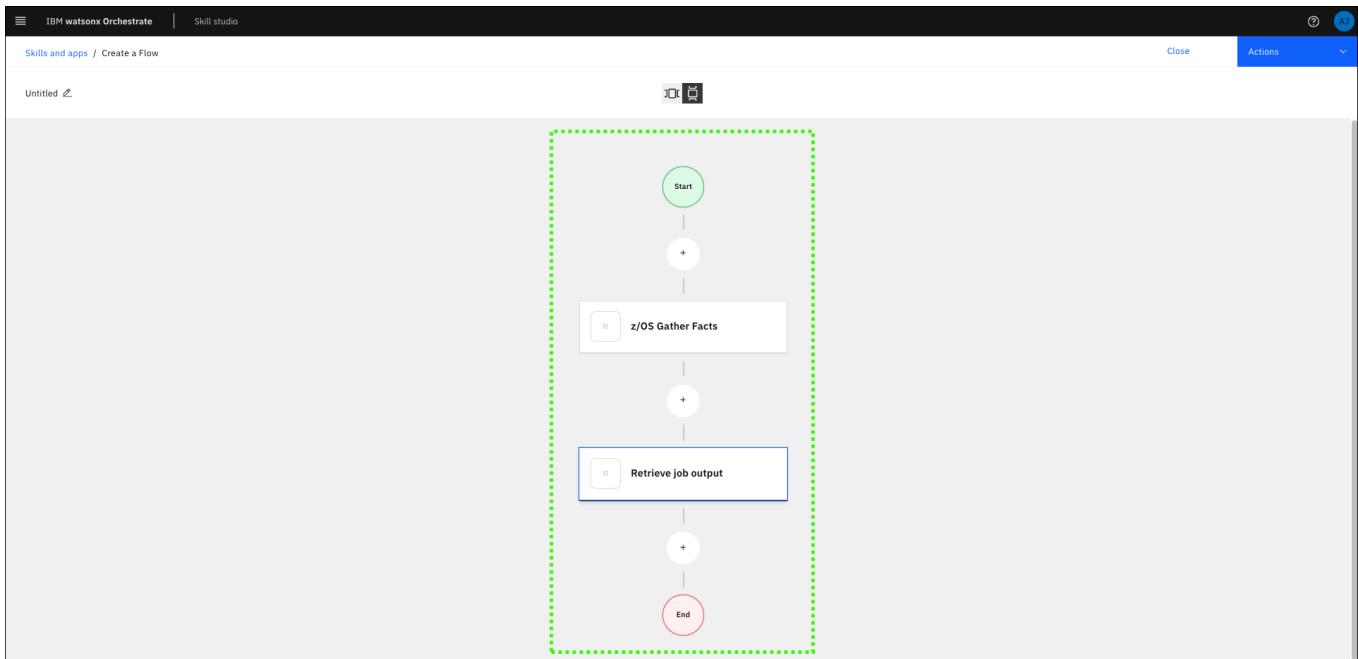


#### 7. Click the + icon after the **z/OS Gather Facts** tile.



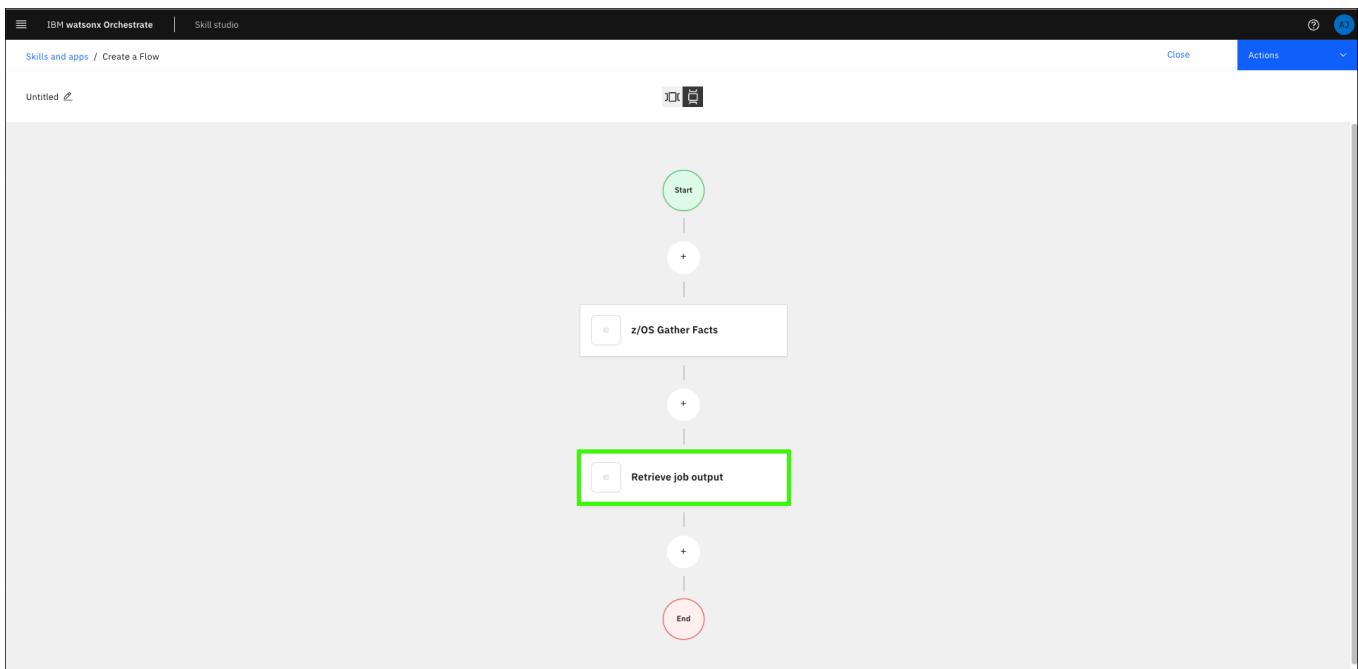
#### 8. Repeat steps 5 and 6 for the **Retrieve job output** skill.

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

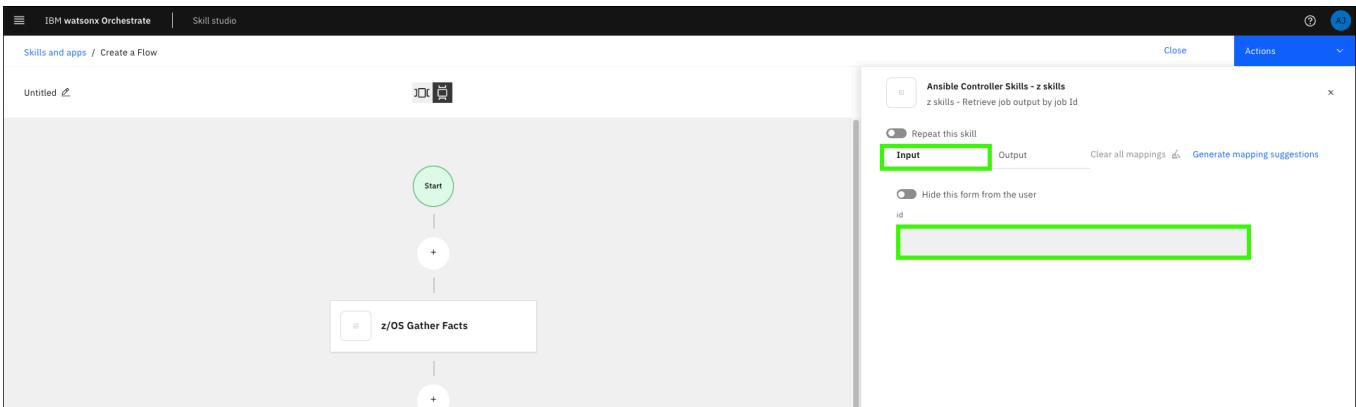


Next you must map the output values of the first skill to the input of the second skill. In this case, pass the job ID output from **z/OS Gather Facts** as an input for **Retrieve job output**.

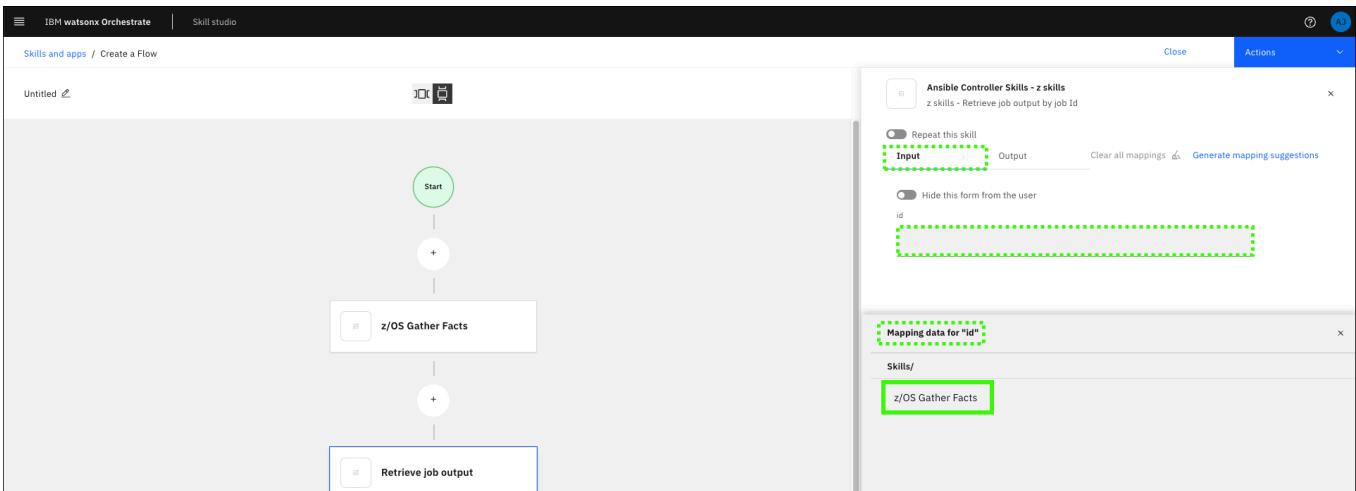
#### 9. Click the **Retrieve job output** tile.



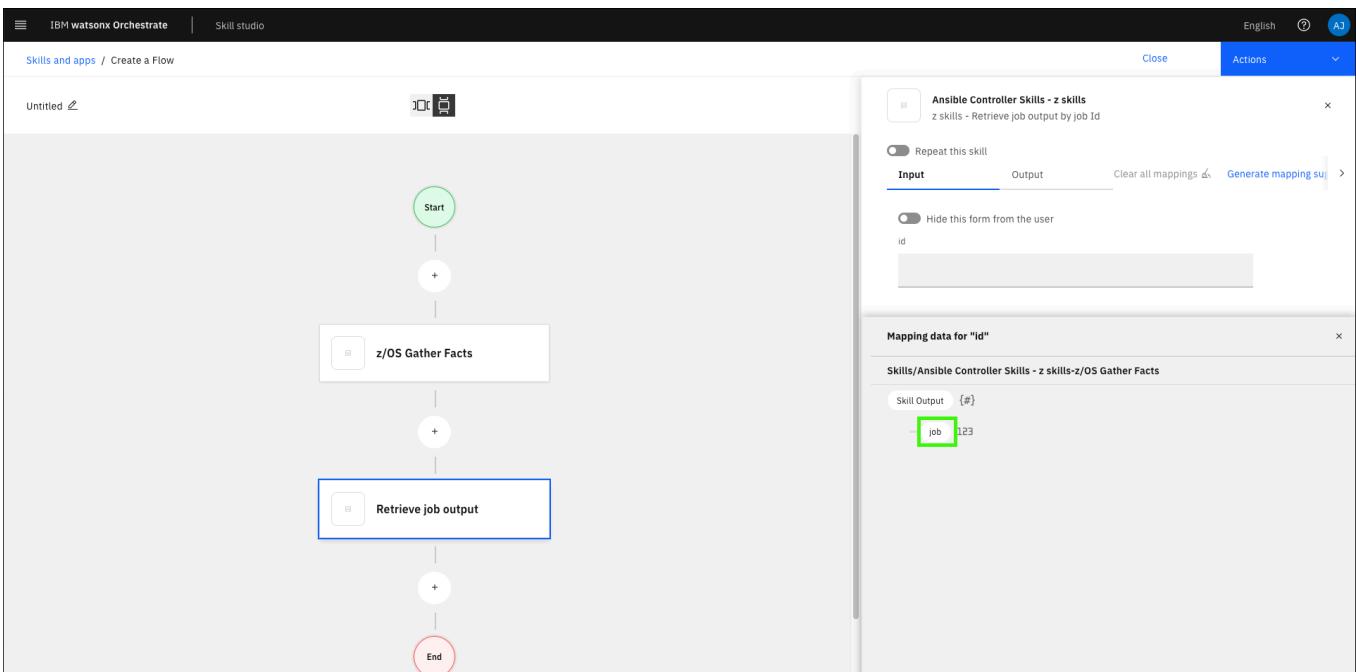
#### 10. Select the **Input** tab and click the **id** field.



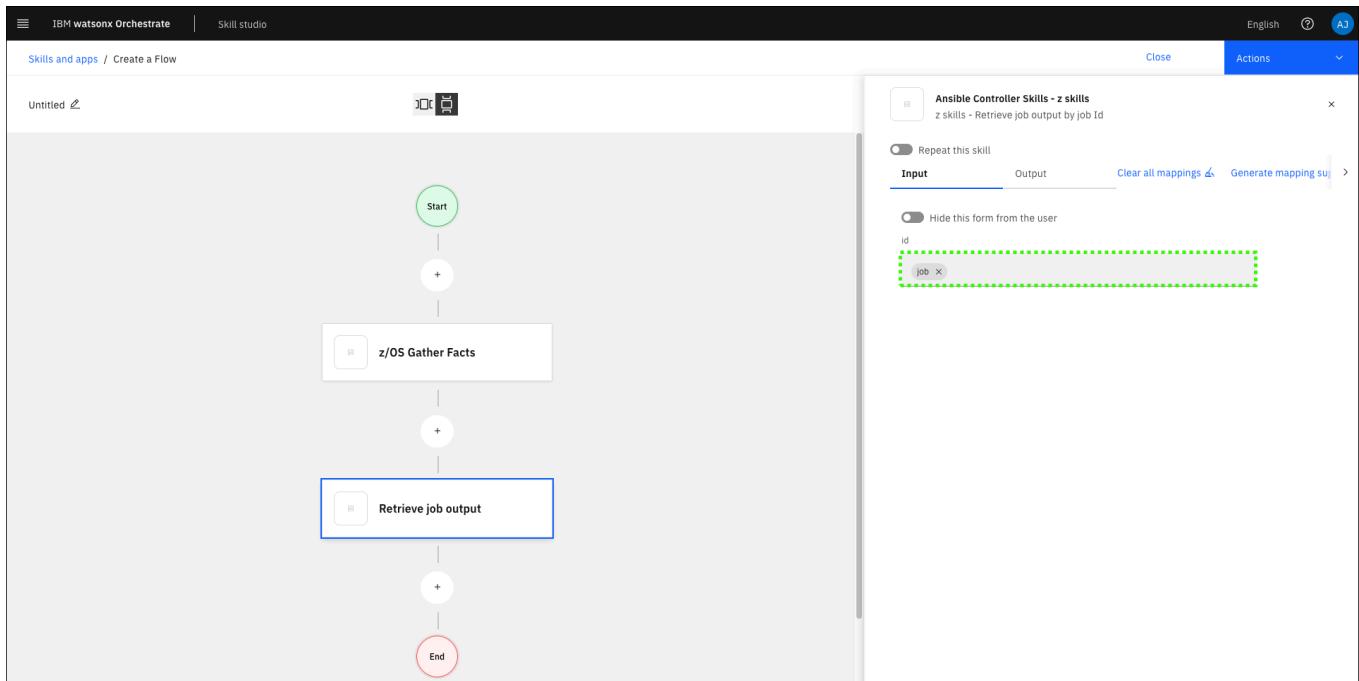
11. Click the **z/OS Gather Facts** skill in the **Mapping data for "id"** section.



12. Click the **job** icon.

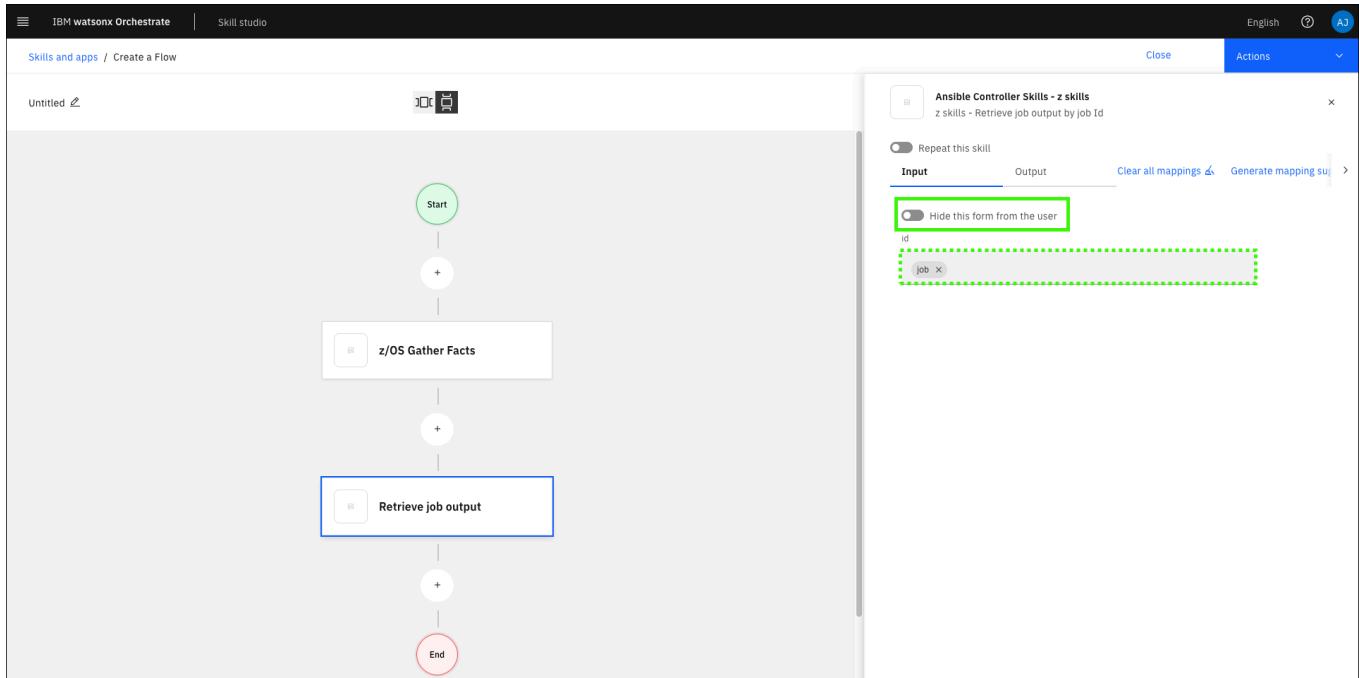


13. Verify that the **job** appears in the **id** field.

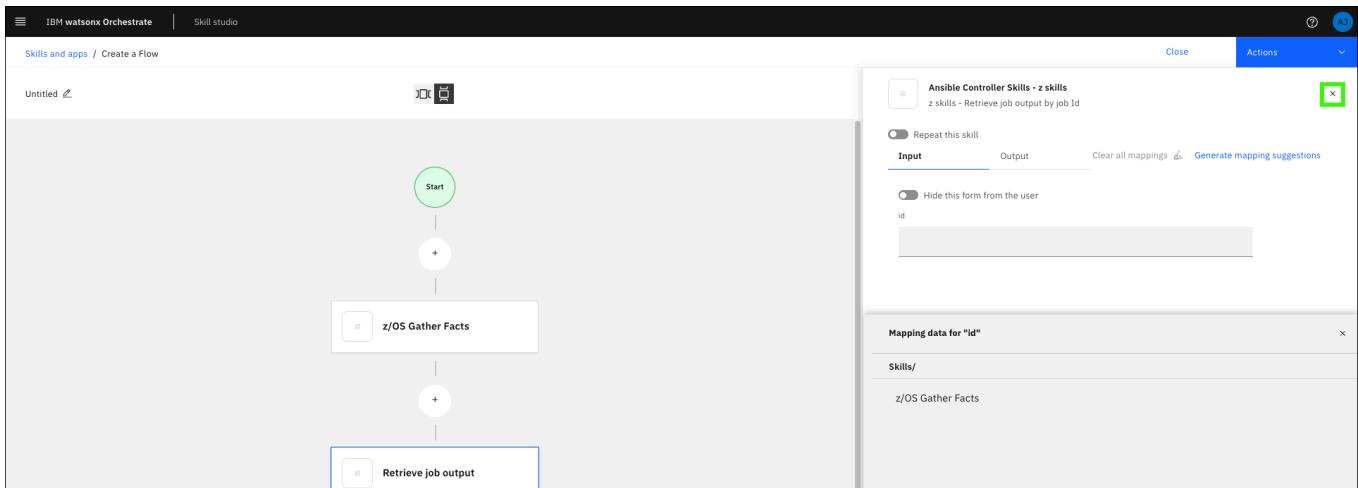


14. Optionally, toggle the **Hide this from the user** setting.

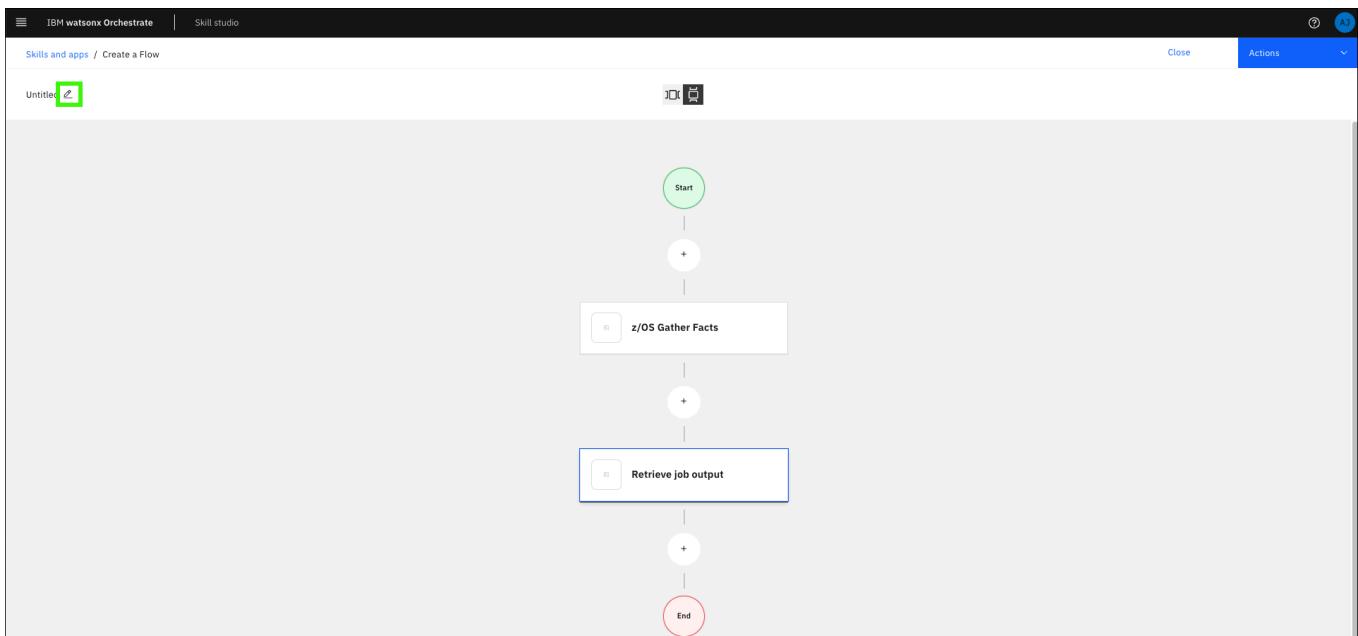
For this lab guide, this option is left disabled. Learn more about this option [here](#).



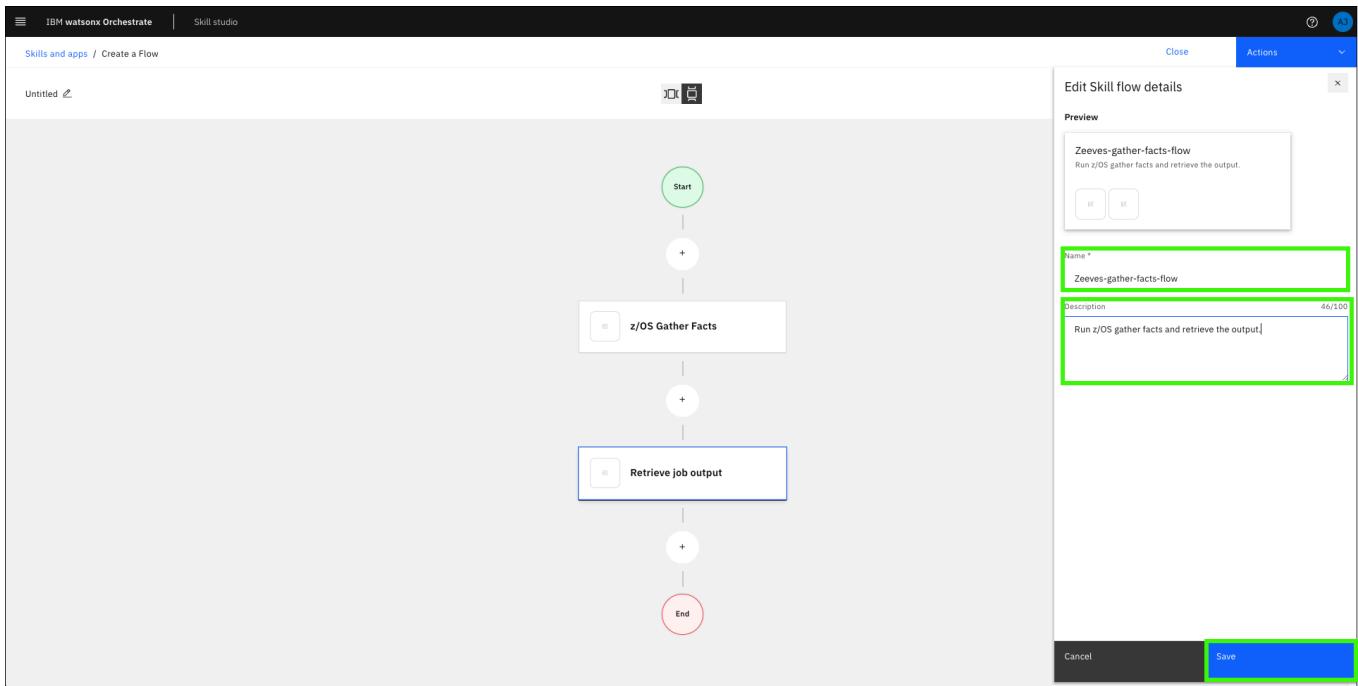
15. Click the **x** to close mapping window.



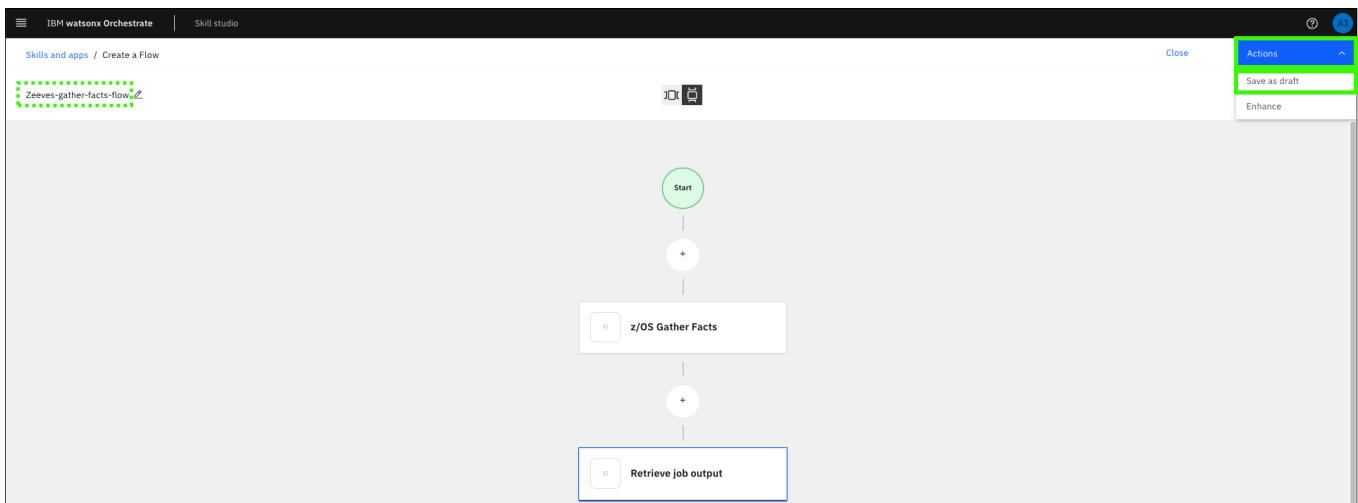
16. Click the pencil ().



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.

Skills and apps / Enhance this skill

Enhance the "Zeeves-gather-facts-flow" skill

Add details that will make people want to use this skill.

Name Phrases Next best skills

Phrases are the text your user types in the chat bar to find and use a skill.

Zeeves-gather-facts-flow

Run z/OS gather facts and retrieve the output.

Enter new train phrase

## 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

Skills and apps / Enhance this skill

Enhance the "Zeeves-gather-facts-flow" skill

Add details that will make people want to use this skill.

Name Phrases Next best skills

Phrases are the text your user types in the chat bar to find and use a skill.

Show me z/OS facts

Gather and display z/OS facts

Enter new train phrase

Cancel Publish Save as draft

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 options: Chat, Skill sets, Skill catalog, BUILD (selected), AI assistant builder (highlighted with a green box), and Skill studio. The main area is titled "Skill studio" and contains a message: "Welcome to the skill studio. This is where you can build and manage skill flows and connectors." Below this is a table with four rows of skill information:

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

A green box highlights the "AI assistant builder" option in the sidebar.

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

The screenshot shows the "Actions" section of the AI assistant builder. The top navigation bar includes Home, Build, Generative AI, Actions (highlighted with a green box), Preview, Deploy, Publish, Environments, Improve, Analyze, and Help. The main content area displays various actions with descriptions and icons:

- 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.

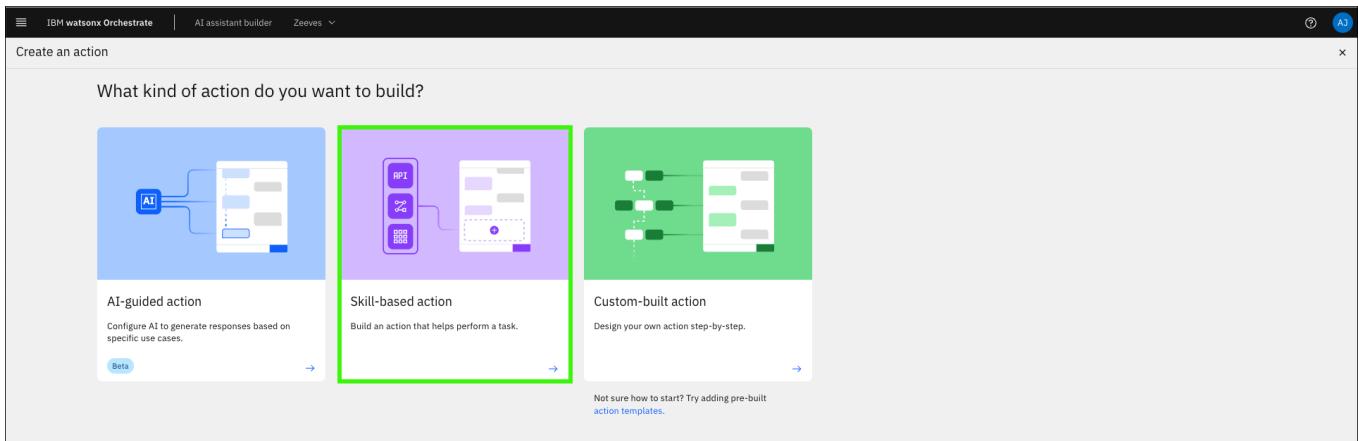
A diagram at the bottom shows a "Default behavior" node branching into "General purpose" and "Conversational search" nodes, which then lead to a "Search" node.

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 others. The main area lists actions with columns: Name, Last edited, Examples count, Steps count, and Status. A green box highlights the "New action" button in the top right corner.

Name	Last edited	Examples count	Steps count	Status
z/OS Gather Facts	2 hours ago	2	0	Green circle icon

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' screen. It lists various skills:

Skill Name	Description	Last updated
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	2024-11-04T10:49:16.502Z
Summarize the Box content	in watsonx.ai	2024-11-04T10:49:12.077Z
Summarize a Zendesk ticket	in watsonx.ai	2024-11-04T10:49:09.476Z
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	2024-11-04T10:48:55.707Z
Salesloft email summary		
Salesforce case summarization		
Salesforce case sentiment analyze		
Outlook email summary		
Github issue summarization		

## 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.

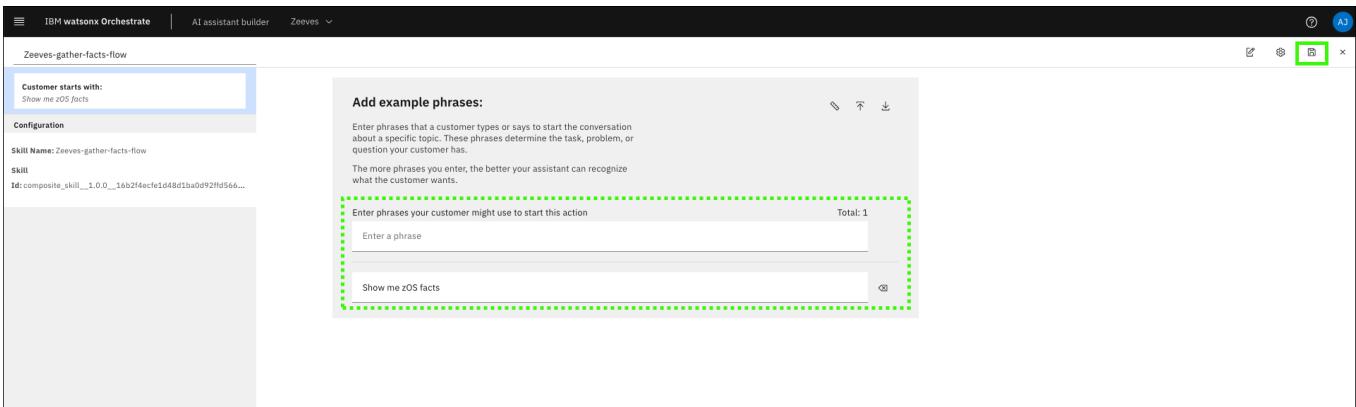
The screenshot shows the configuration screen for the 'Zeeves-gather-facts-flow' skill. It includes sections for 'Customer starts with:' and 'Add example phrases:'.

**Customer starts with:**  
Example: I want to pay my credit card bill.

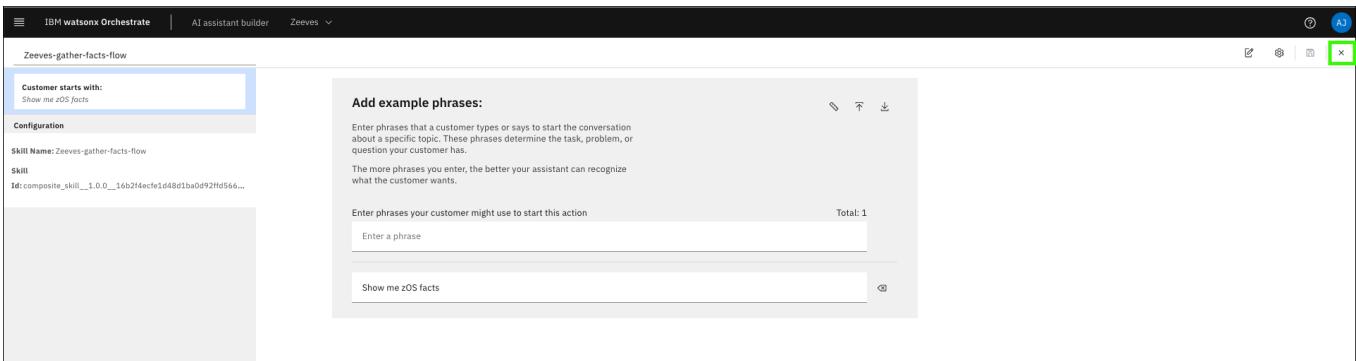
**Add example phrases:**  
Enter phrases that a customer types or says to start the conversation about a specific topic. These phrases determine the task, problem, or question your customer has.  
The more phrases you enter, the better your assistant can recognize what the customer wants.

**New action**  
What does your customer say to start this interaction?  
Show me z/OS facts

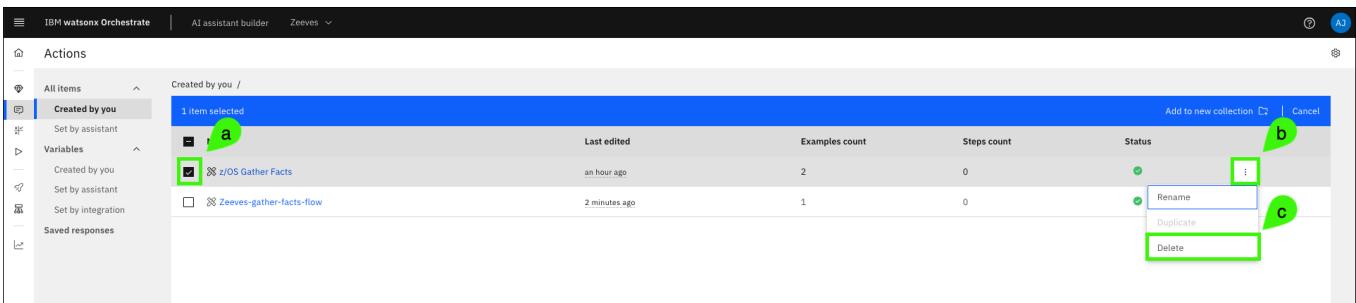
## 7. Enter any additional phrases (prompts) and then click the **save** (💾).



## 8. Click close (x).

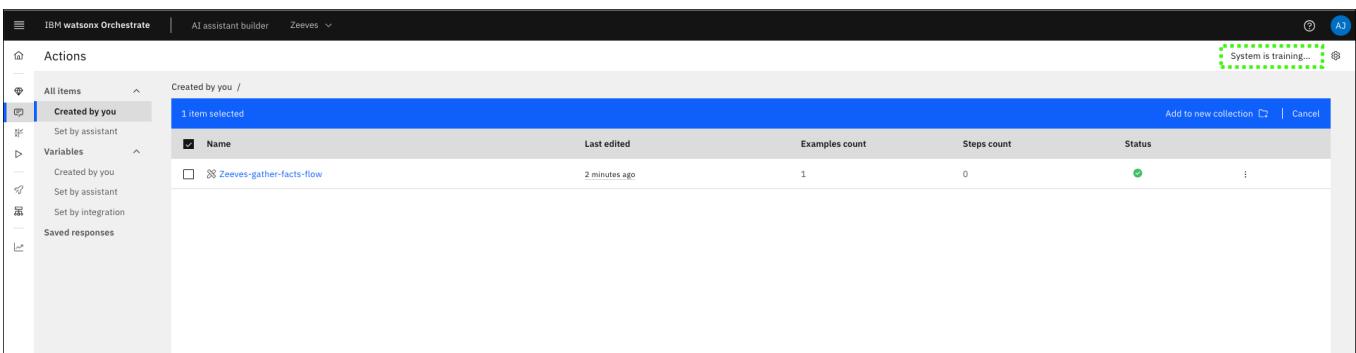


## 9. Select the *original* skill that you created (a) (not the skill flow you just created), click the ellipses (b), and then click **Delete** (c).



## 10. Wait for system training to complete.

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



## 11. Click **Preview**.

Items per page: 50 ▾ Showing 1–1 of 1 items

1 of 1 pages ▾

Preview ▶

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

Items per page: 50 ▾ Showing 1–1 of 1 items

Preview

7:54 AM Greet customer [default]

Welcome, how can I assist you?

You 8:13 AM Show me zOS facts

8:13 AM Conversational skill called Zeeves-gather-facts-flow recognized

Zeeves-gather-facts-flow

id \* 13

Cancel Apply

## 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' selected. Under 'Created by you', there's a list item 'Zeeves-gather-facts-flow'. On the right, a preview window shows a conversation between a user and a bot. The user says 'Show me zOS facts' at 8:13 AM, and the bot responds with 'Zeeves-gather-facts-flow' at 8:13 AM.

#### 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.

The screenshot shows the same interface as before, but the preview window now displays a large amount of JSON-like data. A portion of this data is highlighted with a green dashed box. The data includes parameters like 'ipl\_volume', 'load\_param\_device', and 'operator\_prompt\_flag'.



## 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. The 'Draft' tab is selected, indicated by a green dashed border. The 'Live' tab is also present. On the left, there's a sidebar with icons for environments, channels, and extensions. The main area has sections for 'Draft environment', 'Channels' (listing 'Web chat'), and 'Resolution Methods'. Under 'Resolution Methods', there's a 'Draft content' section with a 'Last edited' timestamp and a 'Edit content' link. A callout box points to the 'Last edited' timestamp and the 'Draft' link. Below this is an 'Extensions' section with a 'Search' button.

This screenshot shows the same 'Environments' page, but the 'Live' tab is selected, indicated by a green dashed border. The 'Draft' tab is now unselected. The main area has sections for 'Live environment', 'Channels' (listing 'Web chat'), and 'Resolution Methods'. Under 'Resolution Methods', there's a 'Content' section with a radio button labeled 'No published version' and a 'Publish version' link. A callout box points to the 'No published version' radio button. Below this is an 'Extensions' section with a 'Search' button and a note about extending what the assistant can answer by searching existing documents. There's also an 'Add +' button at the bottom right of the extensions section.

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**.

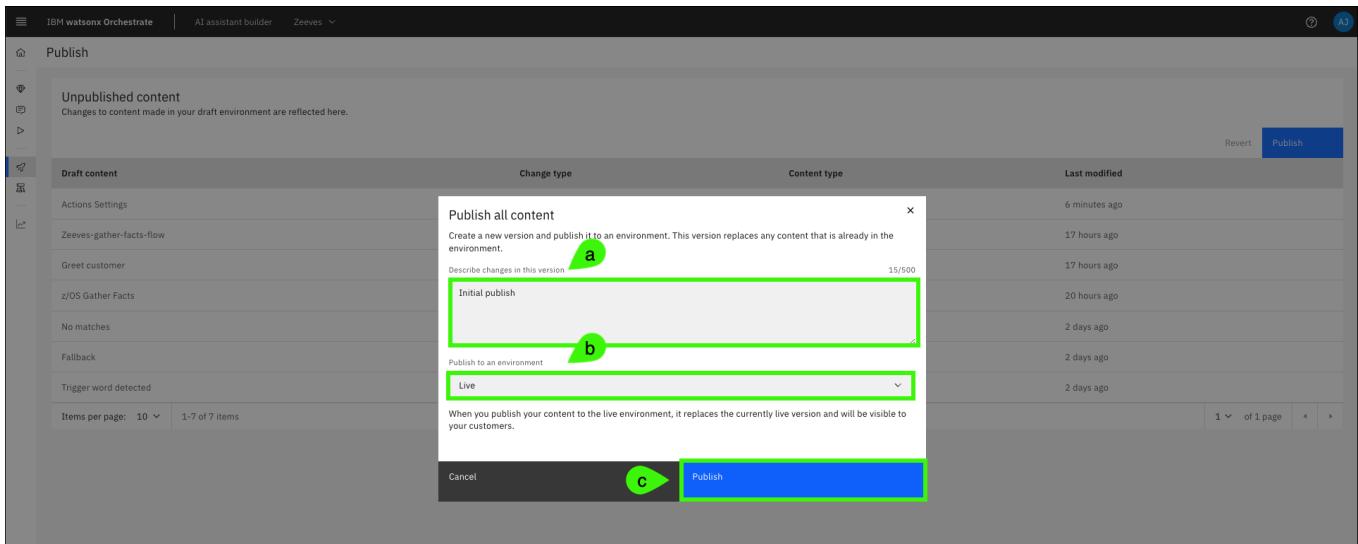
The screenshot shows the IBM Watsonx Orchestrate interface with the "AI assistant builder" tab selected. The left sidebar has a "Publish" button highlighted with a green border. The main area displays various configuration options and a flowchart diagram at the bottom.

2. Click **Publish**.

The screenshot shows the "Publish" page. At the top, there's a message about unpublished content. Below it is a table titled "Draft content" listing various changes made to the assistant. The last column, "Last modified", shows the time for each change. At the bottom right of the table, the "Publish" button is highlighted with a blue border.

Draft content	Change type	Content type	Last modified
Zeeves-gather-facts-flow	Updated	Actions	30 minutes ago
Greet customer	Updated	Actions	31 minutes ago
Actions Settings	Updated	Settings	2 hours ago
Fallback	Updated	Actions	2 hours ago
No matches	Updated	Actions	2 hours ago
Trigger word detected	Updated	Actions	2 hours ago

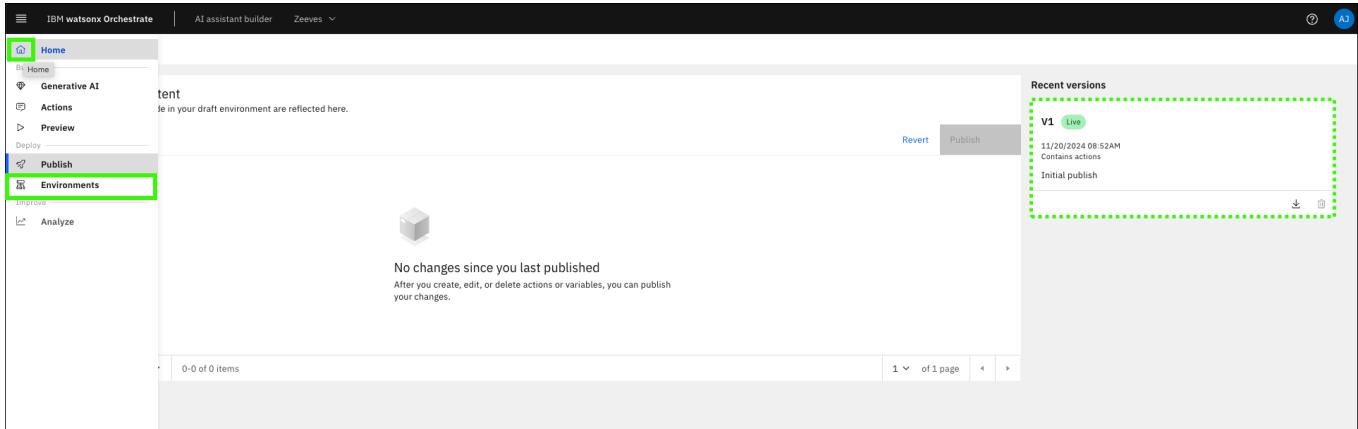
3. Enter a description of the changes (a), set the environment to **Live** (b), and then click **Publish** (c).



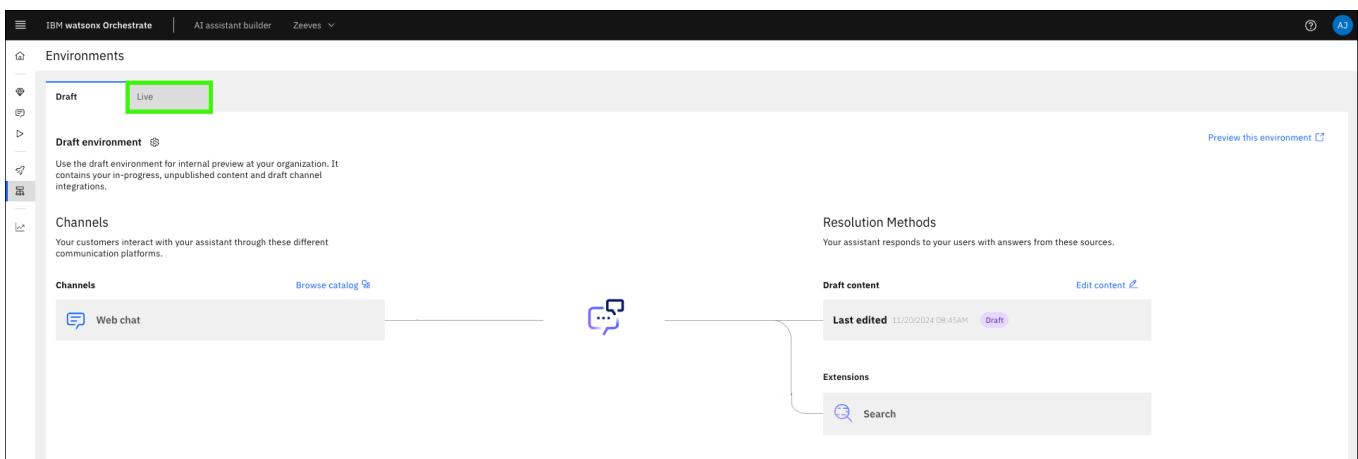
## Configure the live environment

**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.

The screenshot shows the 'Live' environment configuration in the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with icons for environments, channels, and other settings. The main area is titled 'Live environment' and contains the following sections:

- Channels:** A list of communication platforms where customers can interact with the assistant. One item, 'Web chat', is highlighted with a green box.
- Resolution Methods:** Information about how the assistant responds to users.
- Content:** Shows version V1 from 11/20/2024 at 08:52AM, with a 'Live' status indicator.
- Extensions:** An option to extend the assistant's capabilities by searching existing documents.

### 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.

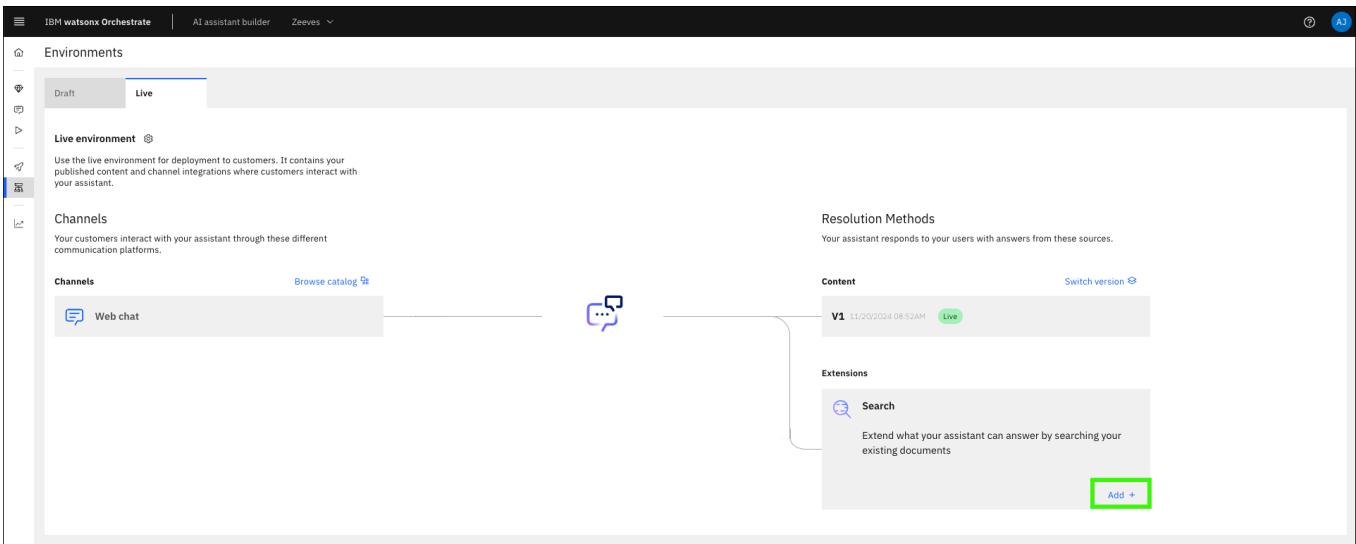
For this lab, toggle **Streaming** on and turn **Suggestions** off on the **Suggestions** tab. You may also want to change the theme to **Dark** to differentiate your draft and live environments.

The screenshot shows the 'Web chat' configuration page. The 'Style' tab is active. In the 'Intended purpose' section, the 'Standard: For virtual agents and customer support experiences.' radio button is selected. Under 'Choose a theme', the 'Dark' color scheme is chosen. The preview window on the right displays a dark-themed chat interface with a blue owl icon and sample messages.

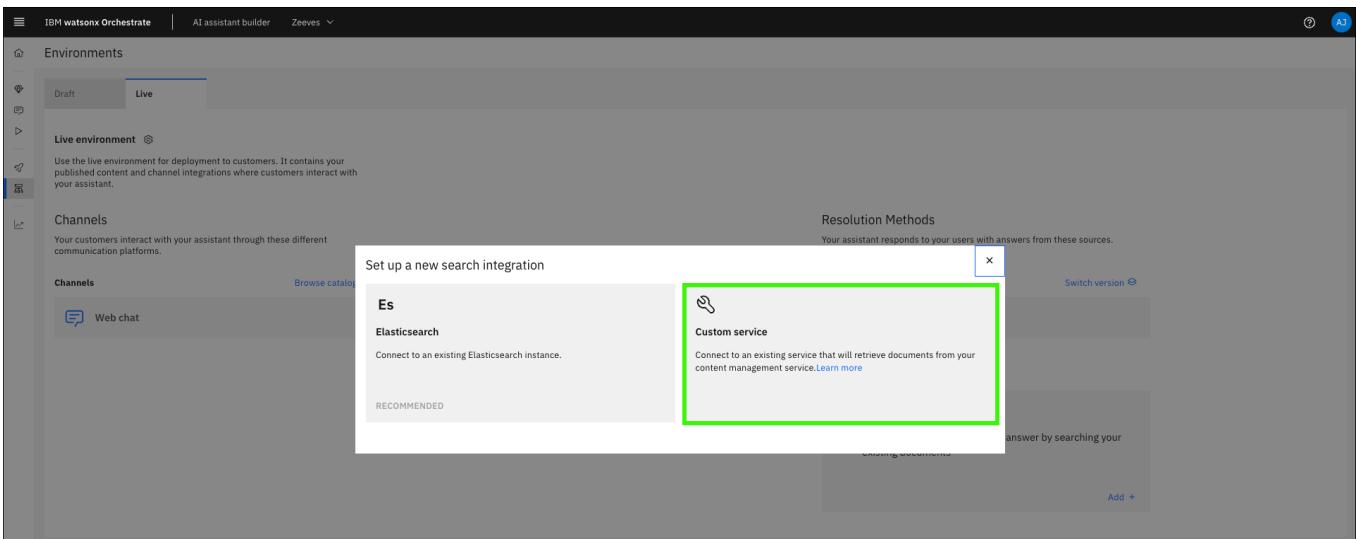
## 5. Click Save and exit.

The screenshot shows the same configuration page after saving changes. The 'Save and exit' button in the top right corner is highlighted with a green box. The interface and preview window remain the same as in the previous screenshot.

## 6. Click Add in the Search tile.



## 7. Click Custom service.



8. 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](#).

Custom service Live

Connect your search provider

By providing credentials  Through your client

a

URL  
ch-wrapper-wxa4z-byos.apps.67585e999266ae8f450d52a8.ocp.techzone.ibm.com/v1/query

b

Choose an authentication type  
Basic authentication

c

Username  
admin

d

Password  
\*\*\*\*\*

e

Close e Next

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

Custom service Live

Enable conversational search (optional)

By enabling conversational search, your assistant can generate responses from supplied documents.

Conversational search will use a watsonx generative AI model hosted in Dallas, TX US, to generate conversational responses. [Learn more](#)

On

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

What's different?

When your assistant receives a message, the default search behavior returns short excerpts of the best search results.

When conversational search is enabled, your assistant will instead generate a conversational response grounded in those search results.

Default search behavior (conversational search off)

Welcome, how can I assist you?  
Can I use my points on airfare?  
I searched my knowledge base and found this information which might be useful  
We offer credit cards offer zero liability to our clients, provides easy financing to pay for travel related purchases. Can I use my points on airfare? Using points and miles to book your flights is a great option. Not only do you save money, but you can use your points to book...

IBM watsonx Lendyr Demo Site

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_*"
  }
}
```

The screenshot shows the 'Custom service' configuration page in the IBM Watsonx Orchestrate AI assistant builder. The interface includes sections for 'Conversational search', 'Search configuration', 'Citations', 'Default filter', and 'Metadata'. Annotations with labels a through h highlight specific features:

- a:** A toggle switch labeled 'On' under 'Conversational search'.
- b:** A radio button labeled 'Single turn' under 'Conversational search'.
- c:** A radio button labeled 'Lowest' under 'Retrieval confidence threshold' in 'Search configuration'.
- d:** A radio button labeled 'Verbose' under 'Generated response length' in 'Search configuration'.
- e:** A radio button labeled 'Lowest' under 'Response confidence threshold' in 'Search configuration'.
- f:** A red box highlighting a JSON configuration block under 'Metadata'.
- g:** A green button labeled 'Save' in the top right corner.
- h:** A green button labeled 'Close' in the top right corner.

## Connect the skills to the live environment

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

The screenshot shows the IBM Watsonx Orchestrate interface. The left sidebar has sections for Chat, Skill sets, Skill catalog, BUILD (AI assistant builder, Skill studio), and ADMINISTER (Access management). The AI assistant builder section is expanded, showing a description of what it is and how to use it. On the right, there's a 'Resolution Methods' panel with 'Content' (version V1, live) and 'Extensions' (Search).

## 2. Select your assistant's live instance in the **Skill sets** list.

The screenshot shows the 'Team Skills' page. The 'Skill sets' list includes 'Team Skills' (selected) and 'Zeeves live'. Below, the 'Skills' section lists 'z/OS Gather Facts', 'Zeeves-gather-facts-flow', and 'Retrieve job output'.

## 3. Click **Connections**.

The screenshot shows the 'Zeeves live' skill set page. The 'Connections' tab is selected. Below, the 'Skills' section lists 'z/OS Gather Facts', 'Zeeves-gather-facts-flow', and 'Retrieve job output'.

## 4. Search for the application name you specified earlier.

Skill sets

Zeeves live

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
Ansible Controller Skills - z skills	3	Not specified	-	

Items per page: 5 1-1 of 1 items

5. Click the ellipses ( ) for your app and then click **Connect app**.

Skill sets

Zeeves live

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
Ansible Controller Skills - z skills	3	Not specified	-	

Items per page: 5 1-1 of 1 items

6. Click **Connect app**.

Skill sets

Zeeves live

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
Ansible Controller Skills - z skills	3	Not specified	-	

Items per page: 5 1-1 of 1 items

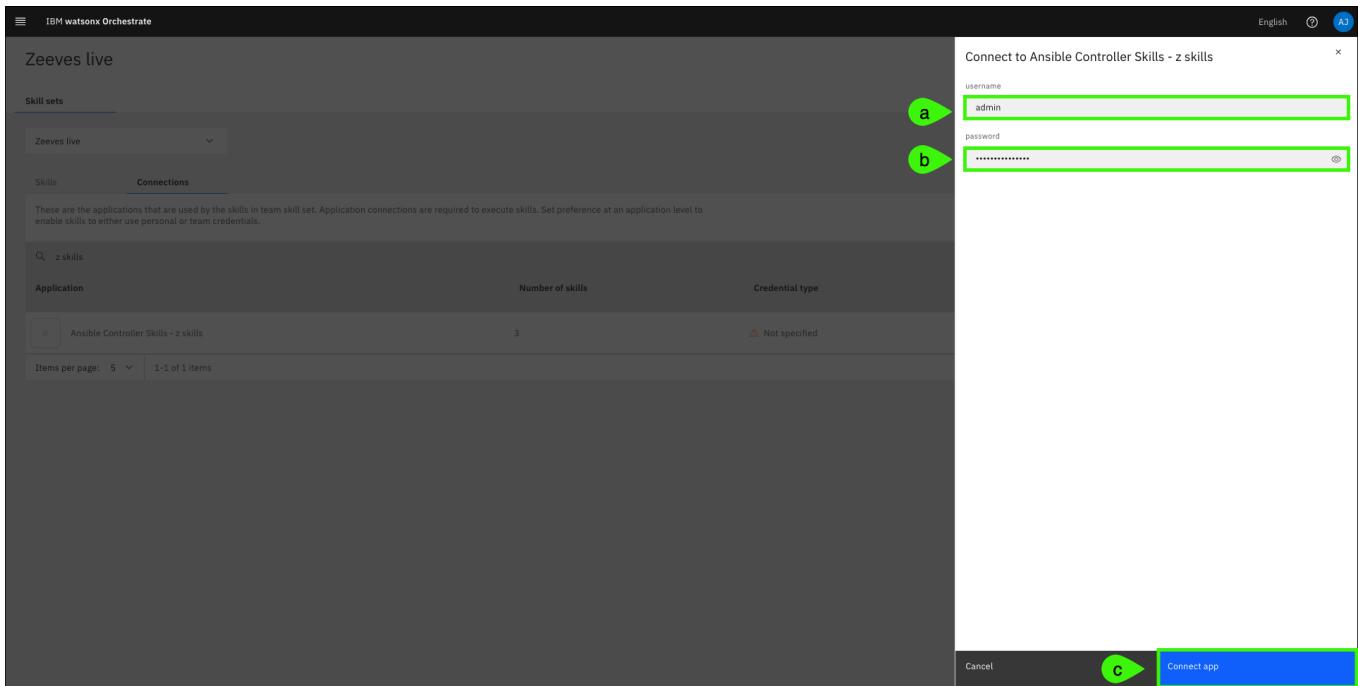
Connect to Ansible Controller Skills - z skills

Member credentials  
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.

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

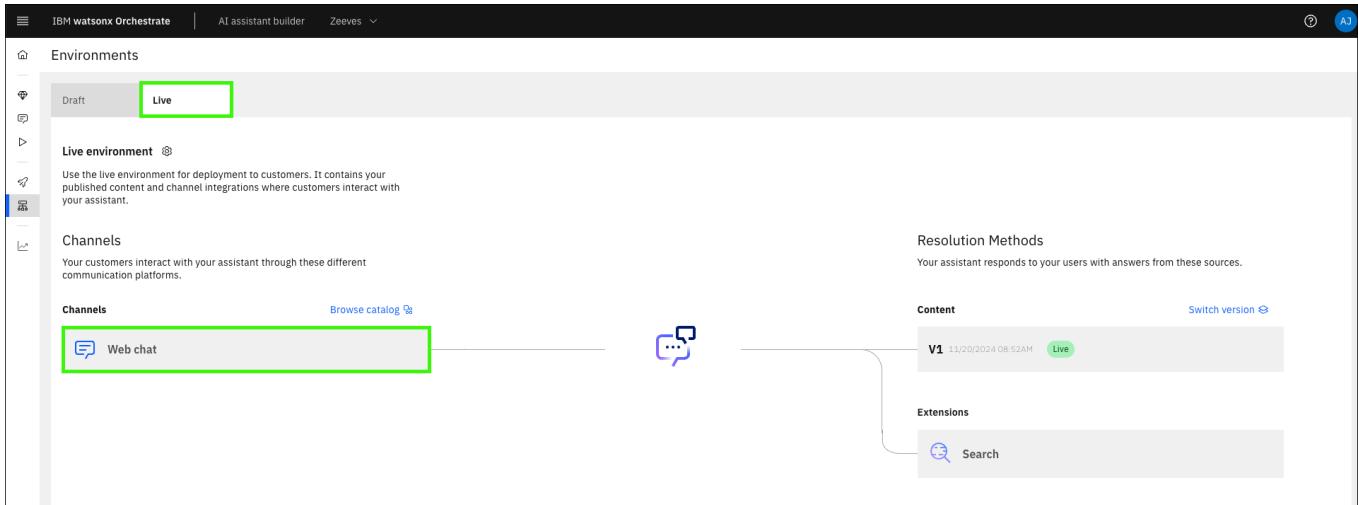
After configuring your assistant's settings and publishing, the final step is to deploy your assistant, which can be done across various channels depending on the use case.

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.

The screenshot shows the 'Web chat' configuration page in the IBM Watsonx Orchestrate interface. The 'Embed' tab is selected and highlighted with a green border. On the right side, there is a preview window titled 'Zeeves 1.0' showing a dark-themed chat interface with a blue owl icon and sample messages. The configuration options include fields for 'Assistant's name as known by customers' (Zeeves 1.0), 'Intended purpose' (Standard: For virtual agents and customer support experiences), 'Choose a theme' (Light selected), 'Primary color' (#FFFFFF), 'Secondary color' (#3D3D3D), 'Chat header' (Accent color #0354E9), 'Significant and interactive objects' (Size: Width 380px, Height 640px), and an 'IBM Watermark' toggle switch set to 'On'. A 'Change avatar image' button is also present.

### 4. Copy and record the `integrationID`, `region`, and `serviceInstanceId` values.

The screenshot shows the 'Web chat' configuration page in the IBM Watsonx Orchestrate interface. The 'Embed' tab is selected and highlighted with a green border. On the right, a code block displays the script for embedding the chat on a website, with the integration ID ('integrationID: "8b4ad299-e6b9-434c-8c8c-2754d9272fe5",'), region ('region: "wco-us-south",'), and service instance ID ('serviceInstanceId: "7c1e3381-8dcf-47eb-bdf2-319f531087be",') highlighted in a red box.

### 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:

```
<script>
  window.watsonAssistantChatOptions = {
    integrationID: "8b4ad299-e6b9-434c-8c8c-2754d9272fe5", // The ID of this integration.
    region: "wco-us-south", // The region your integration is hosted in.
    serviceInstanceId: "7c1e3381-8dcf-47eb-bdf2-319f531087be", // The ID of your service instance.
    onLoad: async (instance) => { await instance.render(); }
  };
  setTimeout(function() {
    const t=document.createElement('script');
    t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.version || "latest") + "/index.js";
    document.head.appendChild(t);
  });
</script>
```

```
<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 ×
Users > andrewjones > Downloads > Watson Assistant Chat.html > html
1  <html lang="en">
2  <body>
3  <script>
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

```

```

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') + '/WatsonAssistant';
  document.head.appendChild(t);
});
</script>
</body>
</html>

```

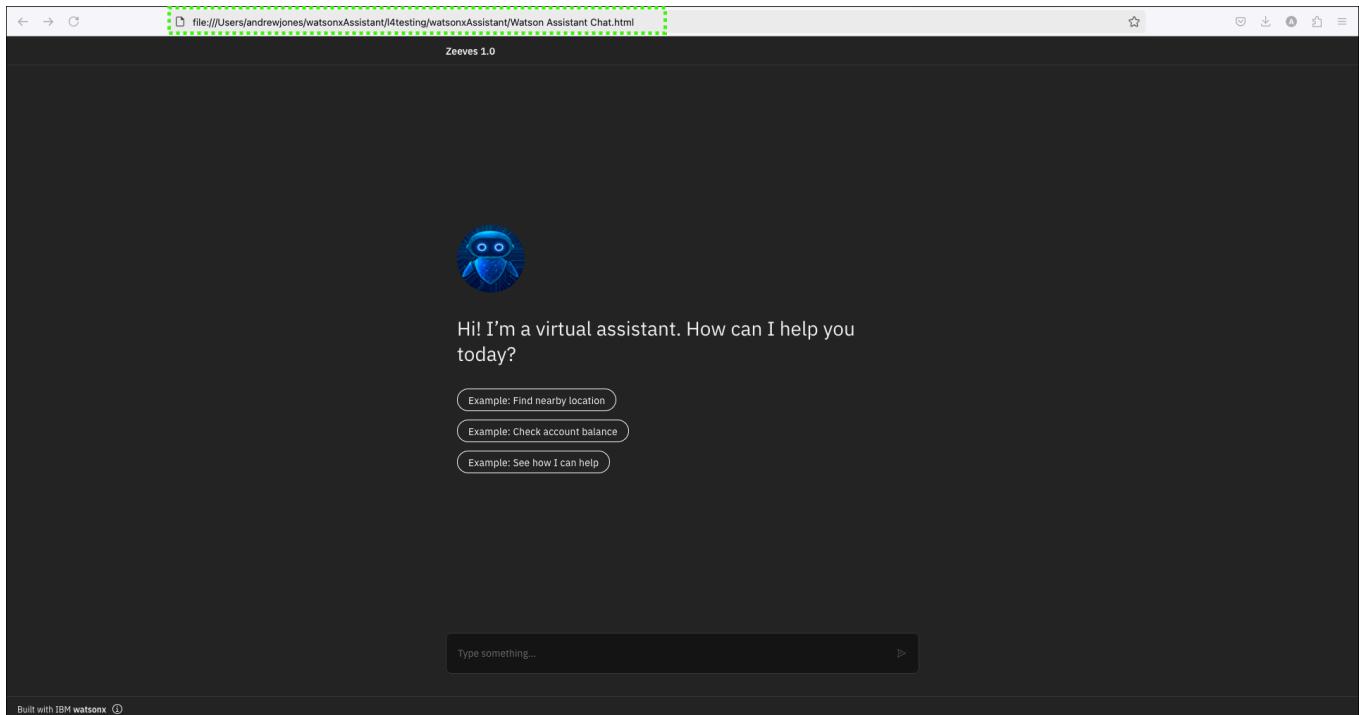
After modification:

```

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 <div class="WebChatContainer">
12
13 <script>
14   const element = document.querySelector('.WebChatContainer');
15
16   window.watsonAssistantChatOptions = {
17     integrationID: "8b4ad299-e6b9-434c-8c8c-2754d9272fe5", // The ID of this integration.
18     region: "wxs-us-south", // The region your integration is hosted in.
19     serviceInstanceId: "7c163381-8dcf-476b-bdf2-319f531087be", // The ID of your service instance.
20     element,
21
22     openChatByDefault: true,
23     hideCloseButton: true,
24
25     layout: {
26       showFrame: false,
27       hasContentMaxWidth: true,
28     },
29
30     onLoad: async (instance) => {
31       window.WACInstance = instance;
32       await instance.render();
33     }
34   };
35
36
37
38
39
40

```

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/OSMV 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-ppxbcn0.techzone.ibm.com:10443>

Before:

{ wxa4z-skillpak-prepackaged-skills.json }

Users > andrewjones > watsonxAssistant > finaltest > watsonxAssistant > {} wxa4z-skillpak-prepackaged-skills.json > ...

```
1 [ "openapi": "3.0.1",
2   "servers": [
3     {
4       "url": "<your z/OSMF URL>"
5     }
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:

```
{
  "openapi": "3.0.1",
  "servers": [
    {
      "url": "https://itzvsi-zos-pwgabob.techzone.ibm.com:10443"
    }
  ],
  "info": [
    "description": "z/OS operations",
    "version": "1.0.0",
    "title": "IBM watsonX Assistant for Z skills pak",
    "x-ibm-annotations": "1.0.0",
    "x-ibm-application-name": "z/OS operations",
    "x-ibm-application-icon": "",
    "x-ibm-application-id": "zos-operations",
    "x-ibm-skill-subtype": "public",
    "x-ibm-skill-type": "imported",
    "x-ibm-disable-default-server-url": "true"
  ],
  "tags": [
    {
      "name": "zOS operations"
    }
  ],
  "paths": {
    "/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**."

**Launch | z/OS TSO Command(s)**

A playbook to execute one or multiple TSO commands on the target z/OS system.

1 Survey    2 Preview

**Command(s) (?)**

```
ALTUSER IBMUSER PHRASE('YOUR PASSWORD PHRASE') NOEXPIRE RESUME
```

**Actions**

Next Back Cancel

## 5. Click Launch.

**Launch | z/OS TSO Command(s)**

A playbook to execute one or multiple TSO commands on the target z/OS system.

1 Survey    2 Preview

Name	z/OS TSO Command(s)	Description	Type	Job Template
Timeout	0 min 0 sec	Job Type	Run	Organization Default
Inventory	AAP z/OS	Project	aap4zos	Execution Environment ap4zos EE
Playbook	zos_tso_command.yml	Forks	0	Verbosity 1 (Verbose)
Show Changes	Off	Job Slicing	1	
Credentials	SSH: z/OS Host SSH ...			
Created	1/17/2025, 5:17:49 AM by admin	Last Modified	1/17/2025, 5:17:49 AM by admin	

**Actions**

Launch Back Cancel

## 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 sections like Views, Jobs, Schedules, Activity Stream, Workflow Approvals, Host Metrics, Subscription Usage, Resources, Access, and Administration. The main area is titled "Jobs > 13 - z/OS TSO Command(s)" and "Output". It shows a "z/OS TSO Command(s)" task with a green "Successful" status. The output pane displays a log of command execution, with line 16 highlighted in green, indicating it failed. The log entries include:

```

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.] *** 07:41:32
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>.

The screenshot shows a web browser window with the URL <https://itzvsi-zos-pwgabob.techzone.ibm.com:10443/zosmf> in the address bar. A warning message box is displayed in the center of the screen, stating: "This Connection Is Not Private. This website may be impersonating "itzvsi-zos-pwgabob.techzone.ibm.com" to steal your personal or financial information. You should close this page." There are "Show Details" and "Close Page" buttons at the bottom of the message box.



**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.

**Welcome to z/OS**

The highly secure, scalable and resilient enterprise operating system for the IBM z Systems mainframe.

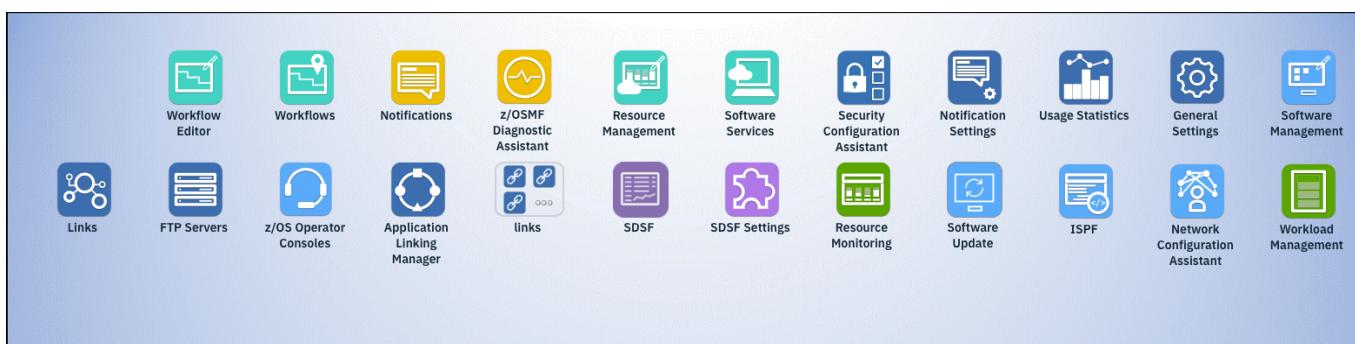
**a** z/OS USER ID  
IBMUSER

**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.

Upgrade 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

Skills

**Create**

Skills Flow

**Import API**

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.

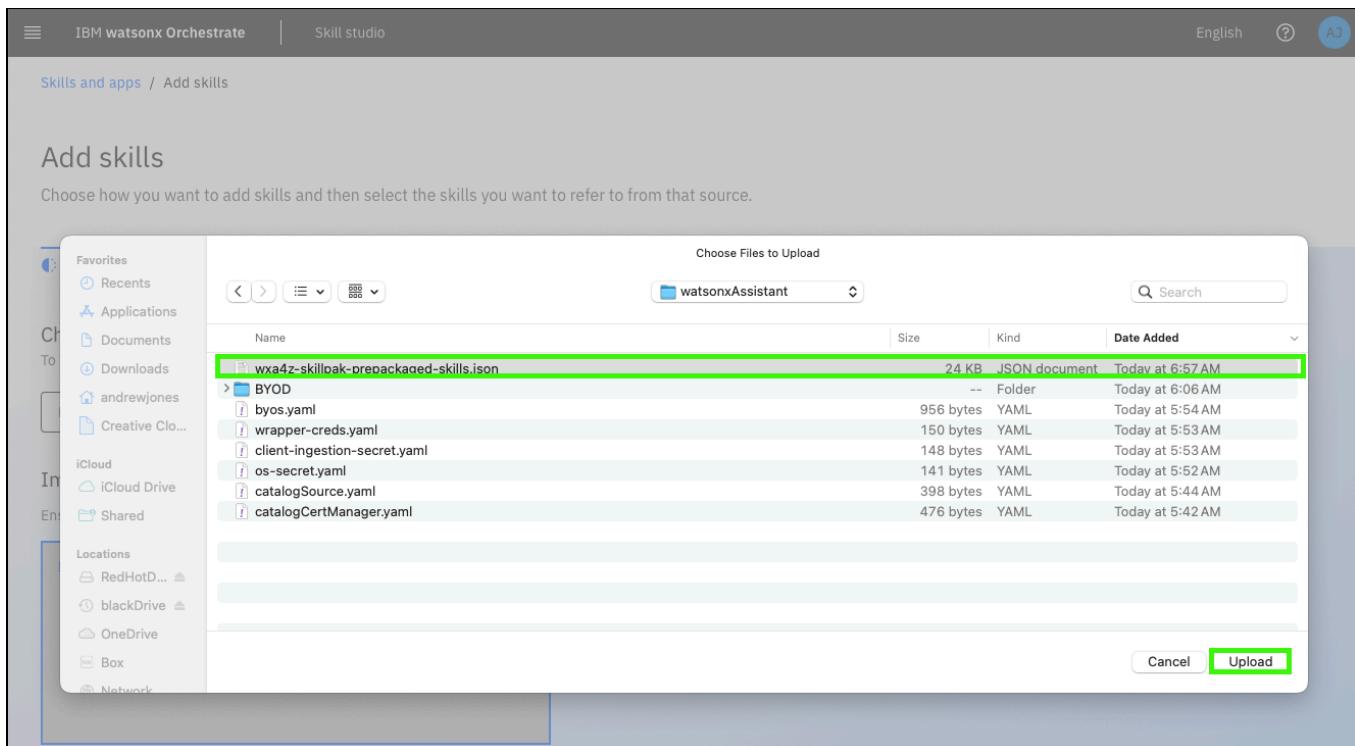
**From an app**    **From a file**    OpenAPI builder (exp...)

**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**.

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.

wxa4z-skillpak-prepackaged-skills.json

**The OpenAPI file or skill package is good to go! ✓**

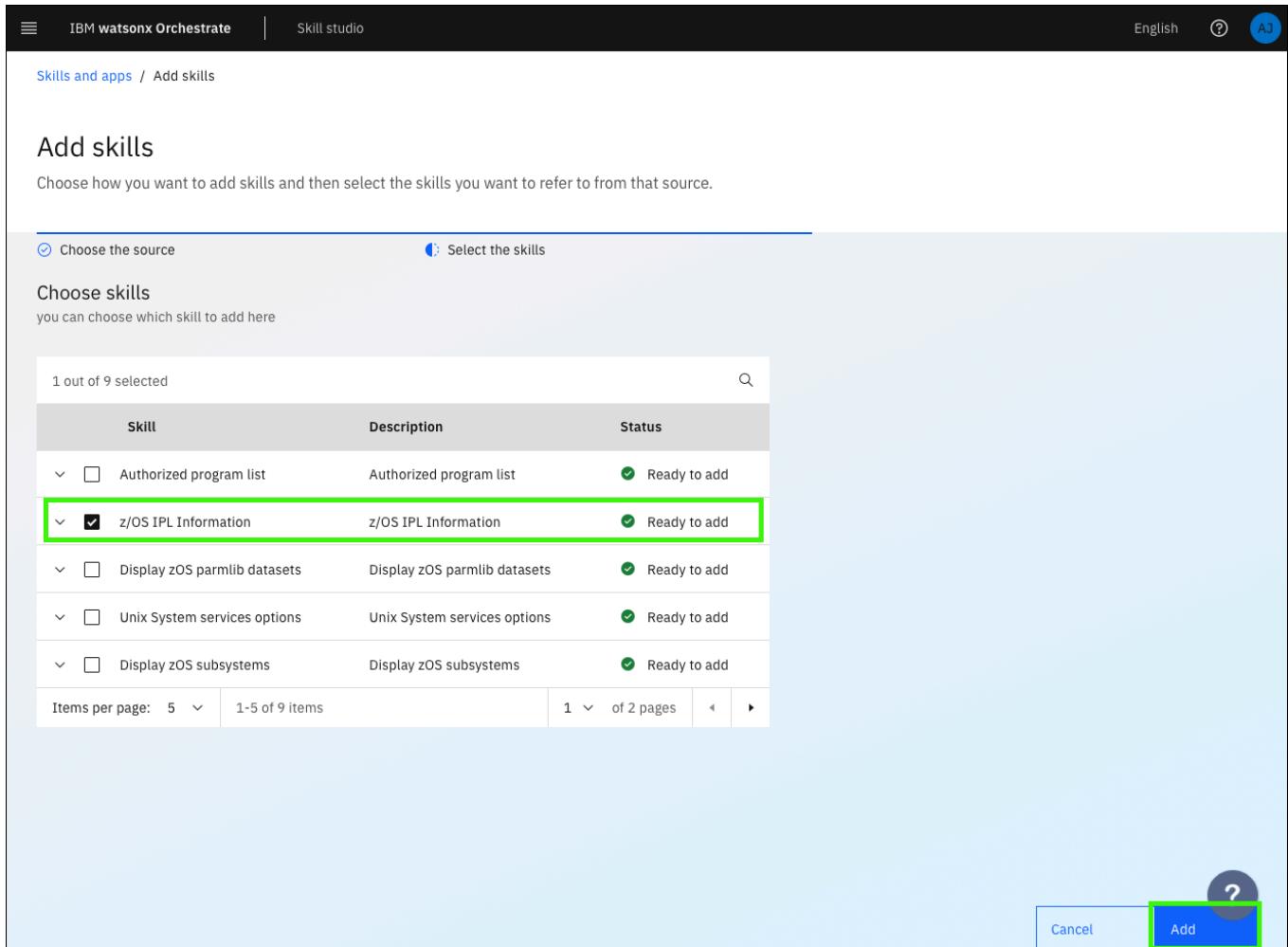
Cancel  ?

### ✖️ 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 adding skills. At the top, there are navigation links for 'Skills and apps' and 'Add skills'. On the right, there are language and help buttons. The main area is titled 'Add skills' with the sub-instruction 'Choose how you want to add skills and then select the skills you want to refer to from that source.' Below this, there are two tabs: 'Choose the source' (selected) and 'Select the skills'. Under 'Choose skills', it says 'you can choose which skill to add here'. A table lists nine skills, with 'z/OS IPL Information' checked and highlighted with a green border. The table columns are 'Skill', 'Description', and 'Status'. The status for all skills is 'Ready to add'. At the bottom, there are pagination controls for 'Items per page' (5), 'Page 1 of 2 pages', and navigation arrows. In the bottom right corner, there are 'Cancel' and 'Add' buttons, with 'Add' being highlighted by a green box.

Skill	Description	Status
<input type="checkbox"/> Authorized program list	Authorized program list	Ready to add
<input checked="" type="checkbox"/> z/OS IPL Information	z/OS IPL Information	Ready to add
<input type="checkbox"/> Display zOS parmlib datasets	Display zOS parmlib datasets	Ready to add
<input type="checkbox"/> Unix System services options	Unix System services options	Ready to add
<input type="checkbox"/> Display zOS subsystems	Display zOS subsystems	Ready to add

#### 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

Configure prebuilt skills

Find a skill

Skills

Name	Step in the process	Status	Skill type	Author	Last edited	More
z/OS IPL Information	Just 1 step away to be ready	<span>Ready to publish</span>	Imported	andrew@jones-tx.com	January 17 2025	<span>⋮</span>
Zeeves-gather-facts-flow	Ready to use	<span>Published</span>	Skill flow	andrew@jones-tx.com	January 17 2025	<span>Enhance this skill</span>
Retrieve job output	Ready to use	<span>Published</span>	Imported	andrew@jones-tx.com	January 17 2025	<span>Export this skill</span>
z/OS Ping	Ready to use	<span>Published</span>	Imported	andrew@jones-tx.com	January 17 2025	<span>Delete this skill</span>
z/OS Gather Facts	Ready to use	<span>Published</span>	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
<b>z/OS IPL Information</b>					

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 IPL Information**

Description 20/100

**z/OS IPL Information**

API version\* 1.0.0

Categories Add categories

App z/OS operations

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.

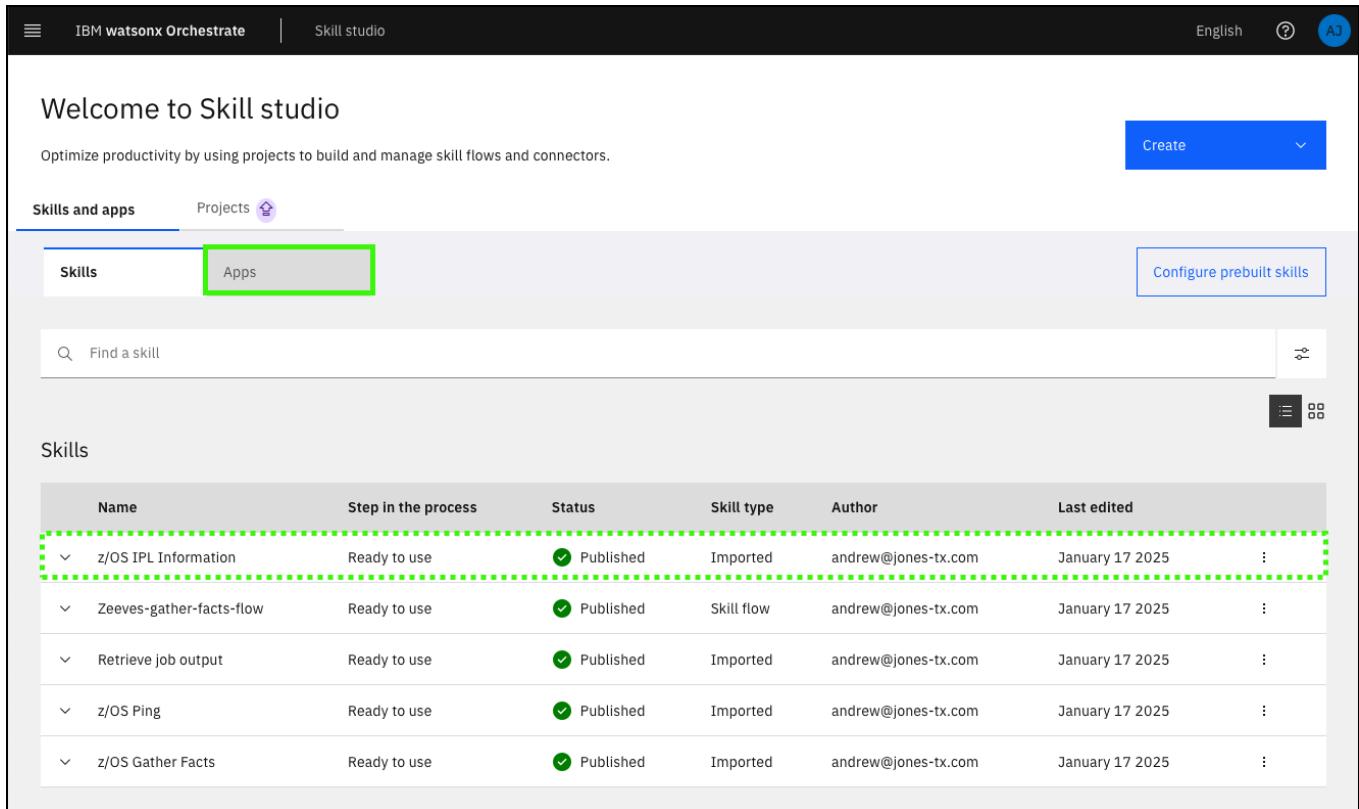
**z/OS IPL Information**

Cancel Publish Save as draft

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 with a dropdown arrow. Below the header, there are two tabs: 'Skills' and 'Apps', with 'Skills' being the active tab. 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 a magnifying glass icon and the placeholder 'Find a skill'. To the right of the search bar are filter and sort icons. The main area is titled 'Skills' and contains a table with the following data:

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.

The screenshot shows the 'Skill studio' interface for 'z/OS operations'. The 'Configuration' tab is selected. In the 'Name\*' field, 'z/OS operations' is entered. The 'Description' field contains 'z/OS IPL Information'. The 'App icon' field shows a small placeholder icon. To the right, there's a preview section showing how the app will look in a skill set, featuring a card with 'z/OS operations' and '1 skill'.

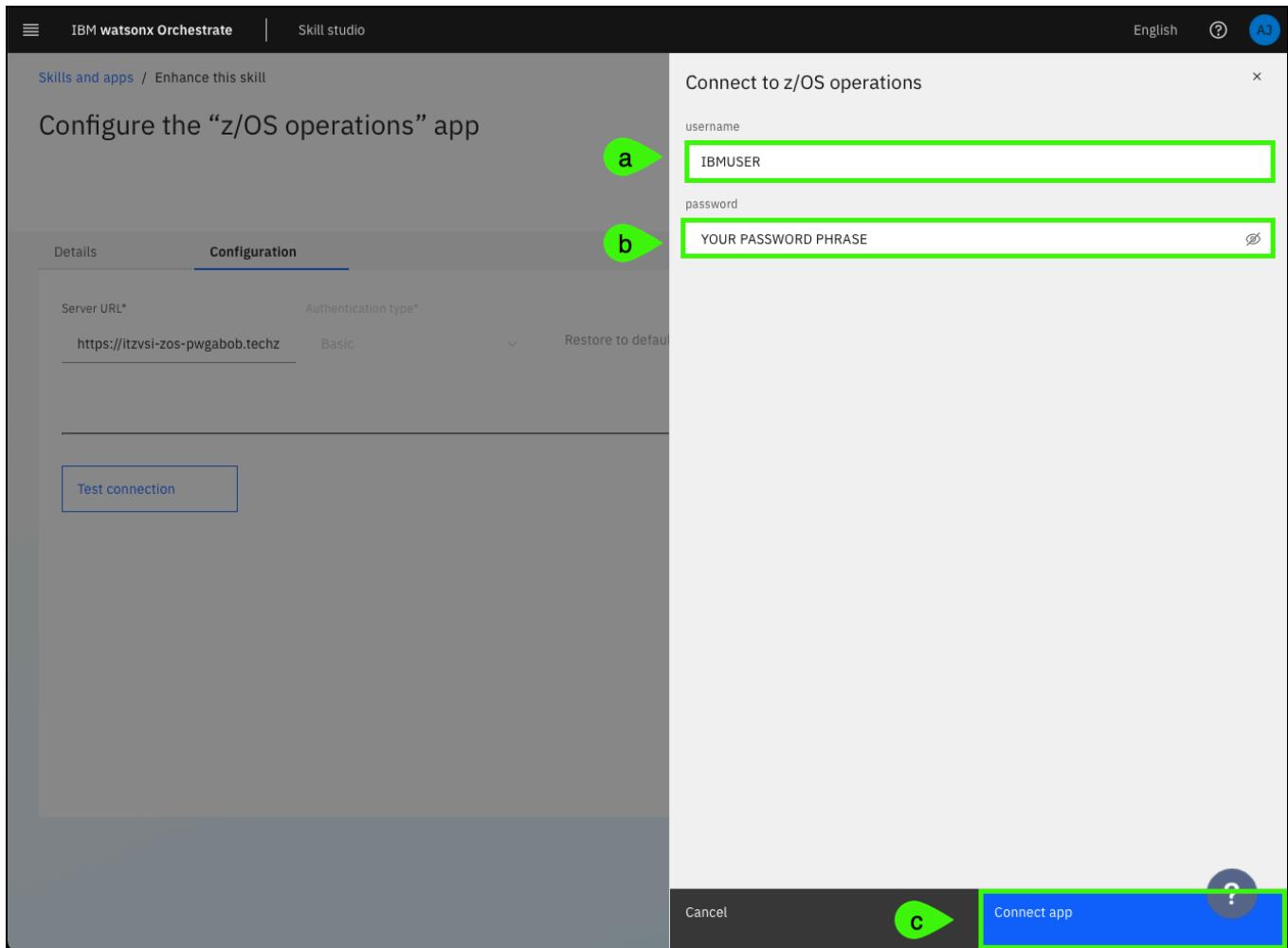
##### 5. Click **Test connection**.

**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> .

The screenshot shows the 'Skill studio' interface for 'z/OS operations' under the 'Configuration' tab. The 'Server URL\*' field contains 'https://itzvsi-zos-pwgabob.techz' and is highlighted with a green dashed border. The 'Authentication type\*' dropdown is set to 'Basic'. Below these fields is a 'Test connection' button, which is also highlighted with a green box.

##### 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\* https://itzvsi-zos-pwgabob.techz Authentication type\* Basic Restore to default ↻

**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 interface. The left sidebar has a dark theme with white text and icons. The 'Chat' option is highlighted with a green border. Other options like 'Skill sets' and 'Skill catalog' are also visible. The main area is titled 'Skill studio' and contains a brief description: 'Build skill flows and connectors to build and manage skill flows and connectors.' A 'Create' button is at the top right. Below it is a 'Configure prebuilt skills' button. A table lists a single skill entry:

Description	Configuration status
z/OS IPL Information	<span>✓ Configured</span>

2. Click **Add skills from the catalog**.

The screenshot shows the AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'Personal skills', 'English', and other settings. A large central area features a chatbot interface with a message from 'Hi. I'm Watson. Check out the skills in the catalog to see how I can help you.' Below is a search bar with the placeholder 'Tell me what you want to do'. At the bottom left, a button labeled 'Add skills from the catalog' is highlighted with a green box. To its right is a card for 'Ansible Controller Skills - z skills' with a small icon and a '2 skills' indicator. A question mark icon is in the bottom right corner.

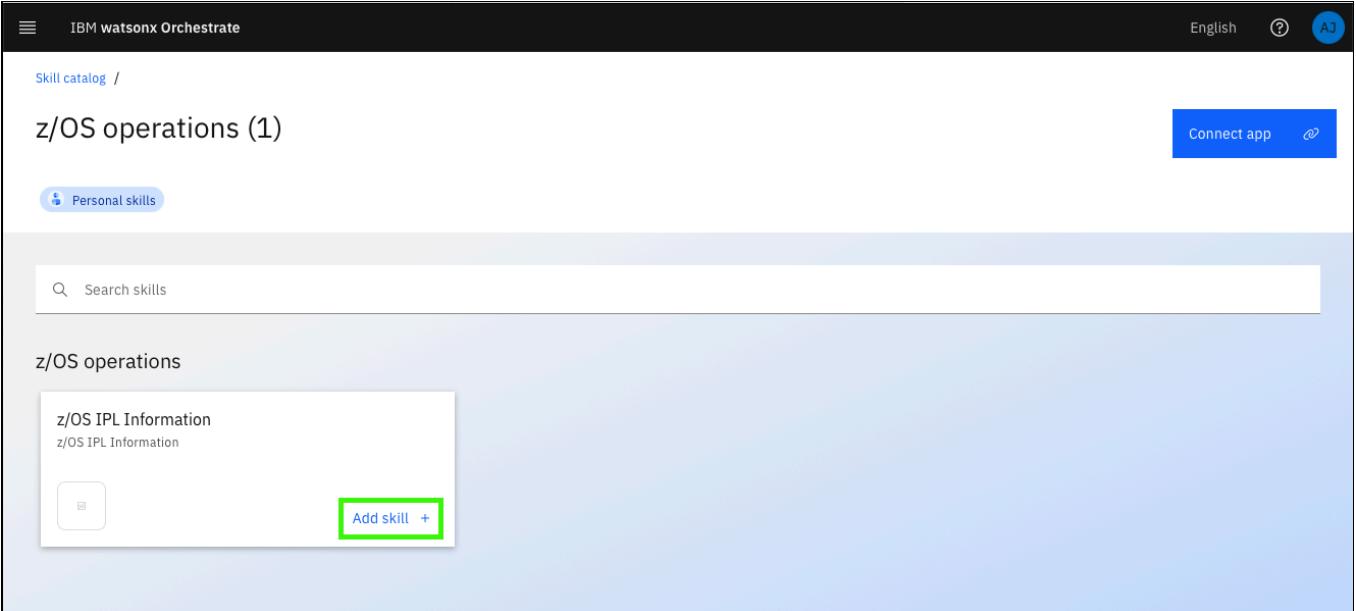
3. In the search apps field, enter the name of the application. Unless you modified the `info` section of the JSON file, the default name is **z/OS operations**.

The screenshot shows the 'Skill catalog' page in IBM Watsonx Orchestrate. At the top, there's a search bar with the placeholder 'z/OS operations'. Below the search bar, a section titled 'Most popular skills' displays three cards: 'Send an email from Gmail', 'Create a lead in Salesforce', and 'Send an email using Outlook'. Each card includes an 'Add skill +' button. Below this, a section titled 'All Apps' shows a grid of app tiles, each with a logo, name, and skill count. The visible apps 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).

- Locate and click the tile for your app (z/OS operations).

This screenshot shows the same 'Skill catalog' page after a search for 'z/OS operations'. The results grid now includes a new row at the bottom right containing the 'z/OS operations' tile, which is highlighted with a green border. This tile also has a green border around its content area.

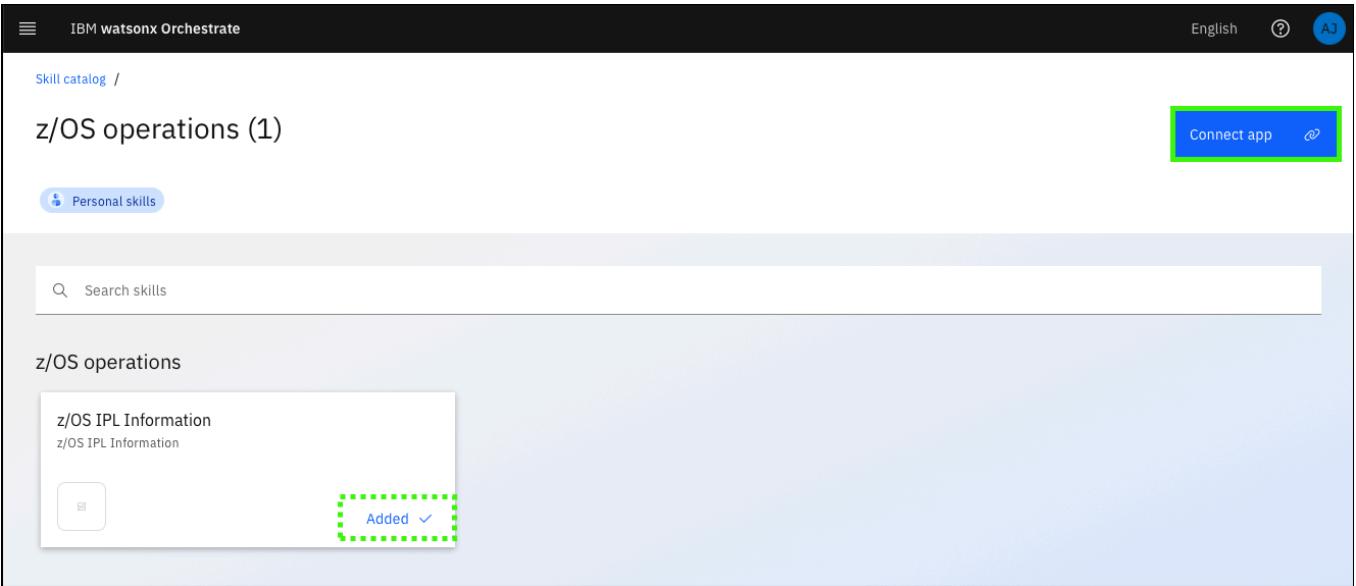
5. Click **Add skill +** for the **z/OS IPL Information** skill.



The screenshot shows the IBM Watsonx Orchestrate interface. At the top, there's a navigation bar with 'Skill catalog /' and 'English'. On the right, there are icons for help and profile. Below the navigation, it says 'z/OS operations (1)'. A blue button labeled 'Connect app' is visible. Under the heading 'z/OS operations', there is a single skill card for 'z/OS IPL Information'. The skill card includes a small icon, the skill name, and an 'Add skill +' button, which is highlighted with a green box.

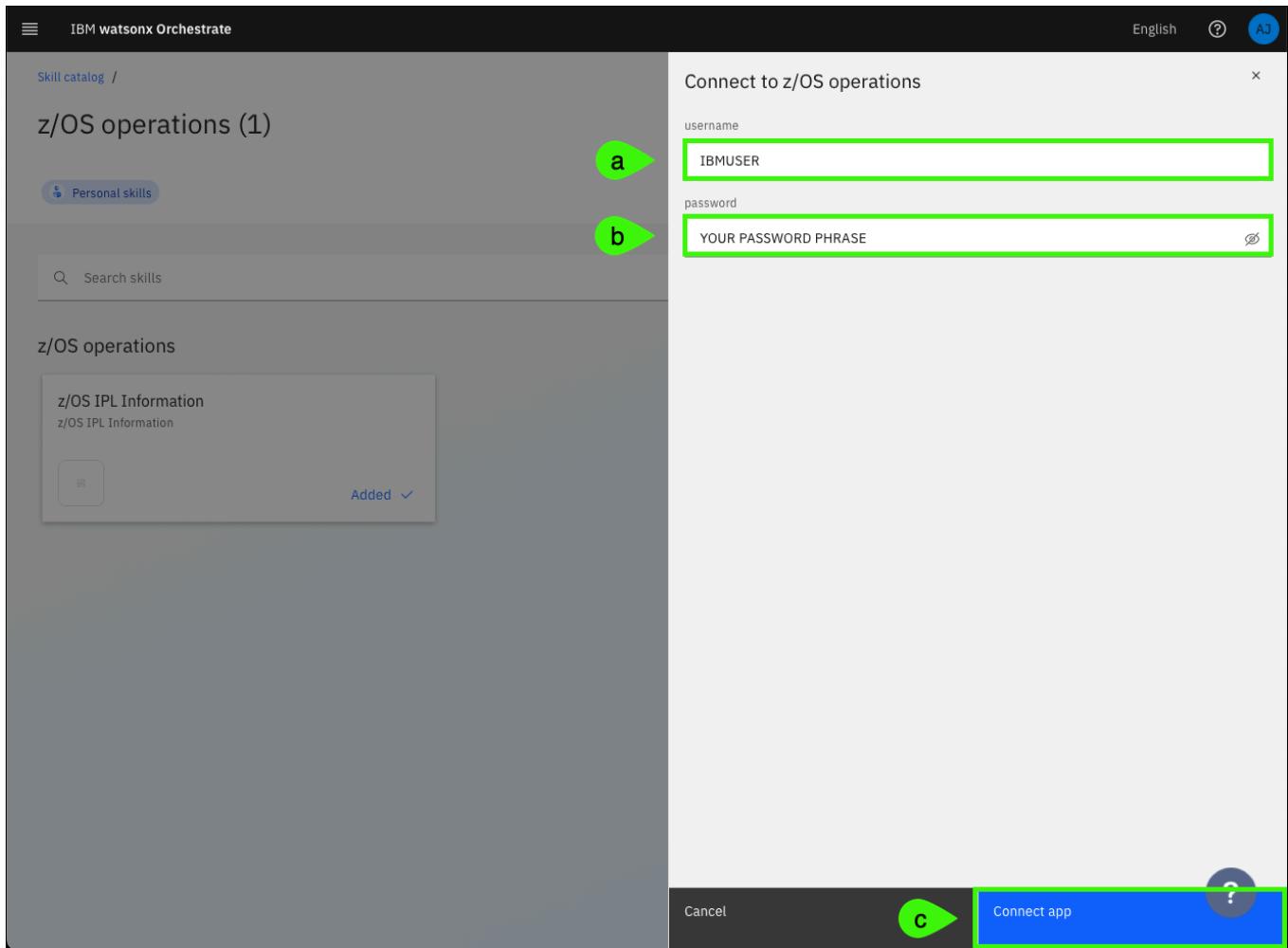
6. Repeat step 5 for any additional skills you added to the app.

7. Click **Connect app**.

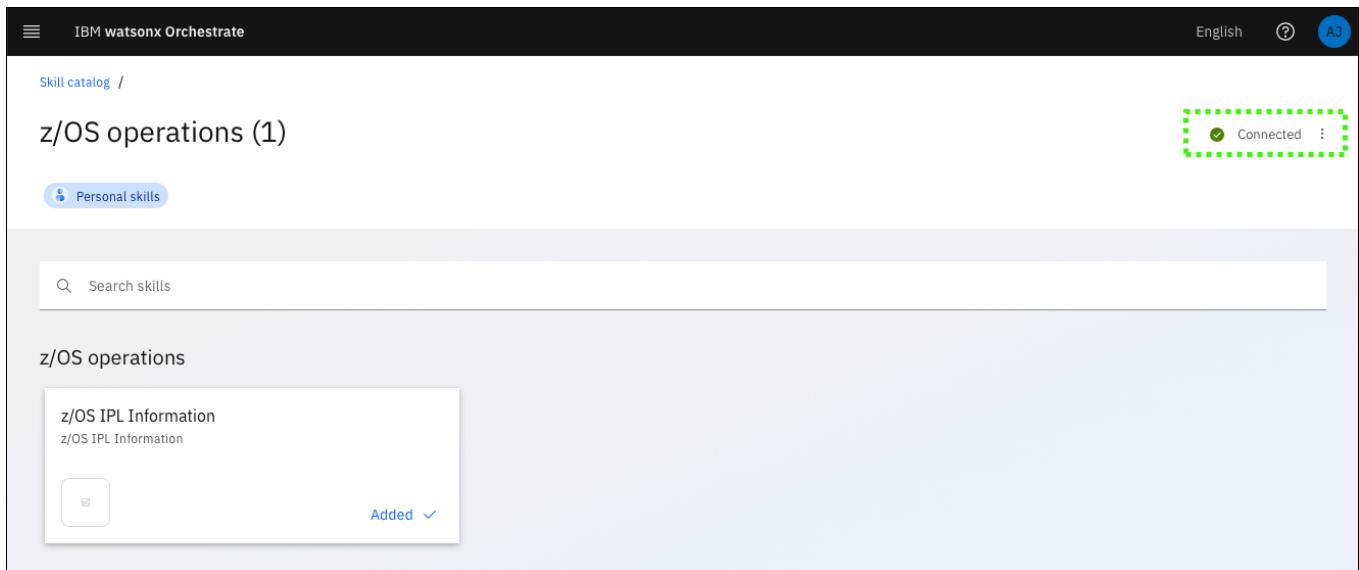


This screenshot shows the same interface as the previous one, but after step 5 has been completed. The 'z/OS IPL Information' skill card now has an 'Added ✓' indicator next to its 'Add skill +' button, and both the 'Add skill +' button and the 'Connect app' button are highlighted with green boxes.

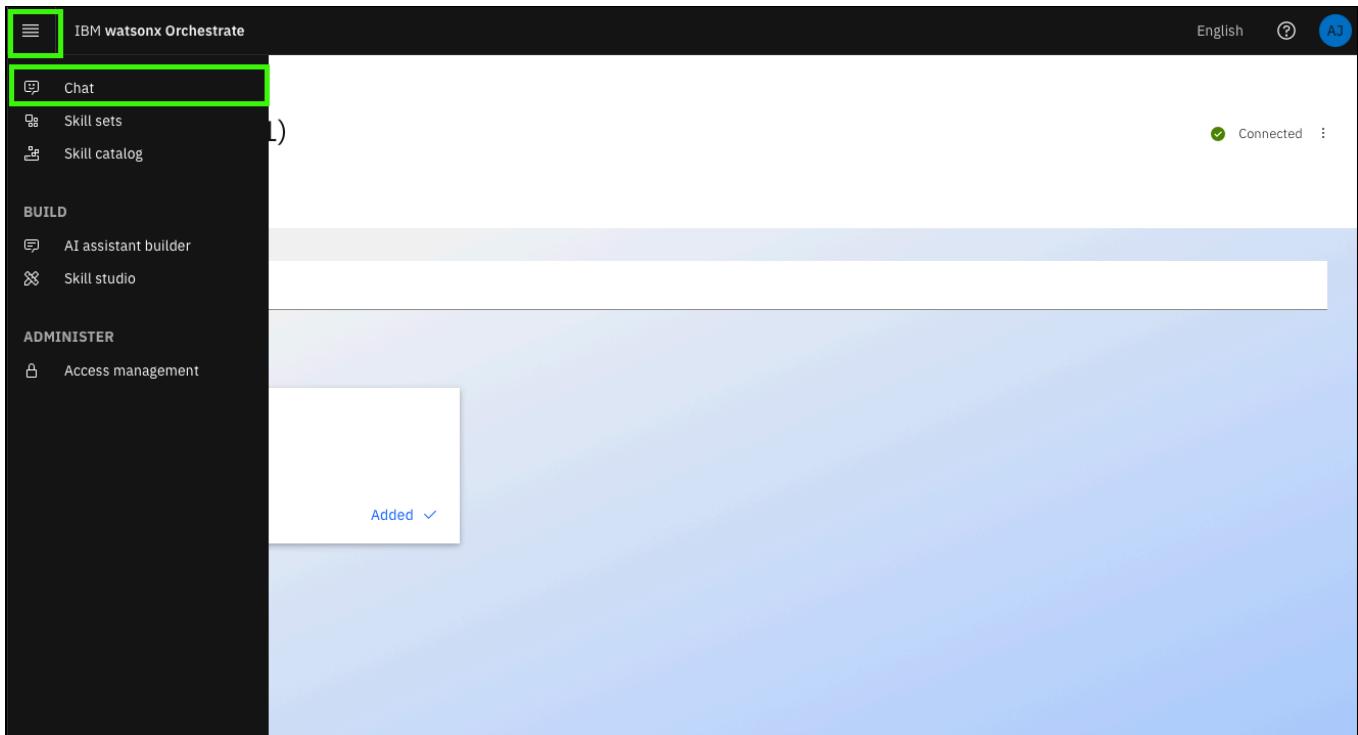
8. Enter (a) **IBMUSER** in the **username** field, (b) your **WaaS password** created earlier in the **password** field, and then (c) click **Connect app**.



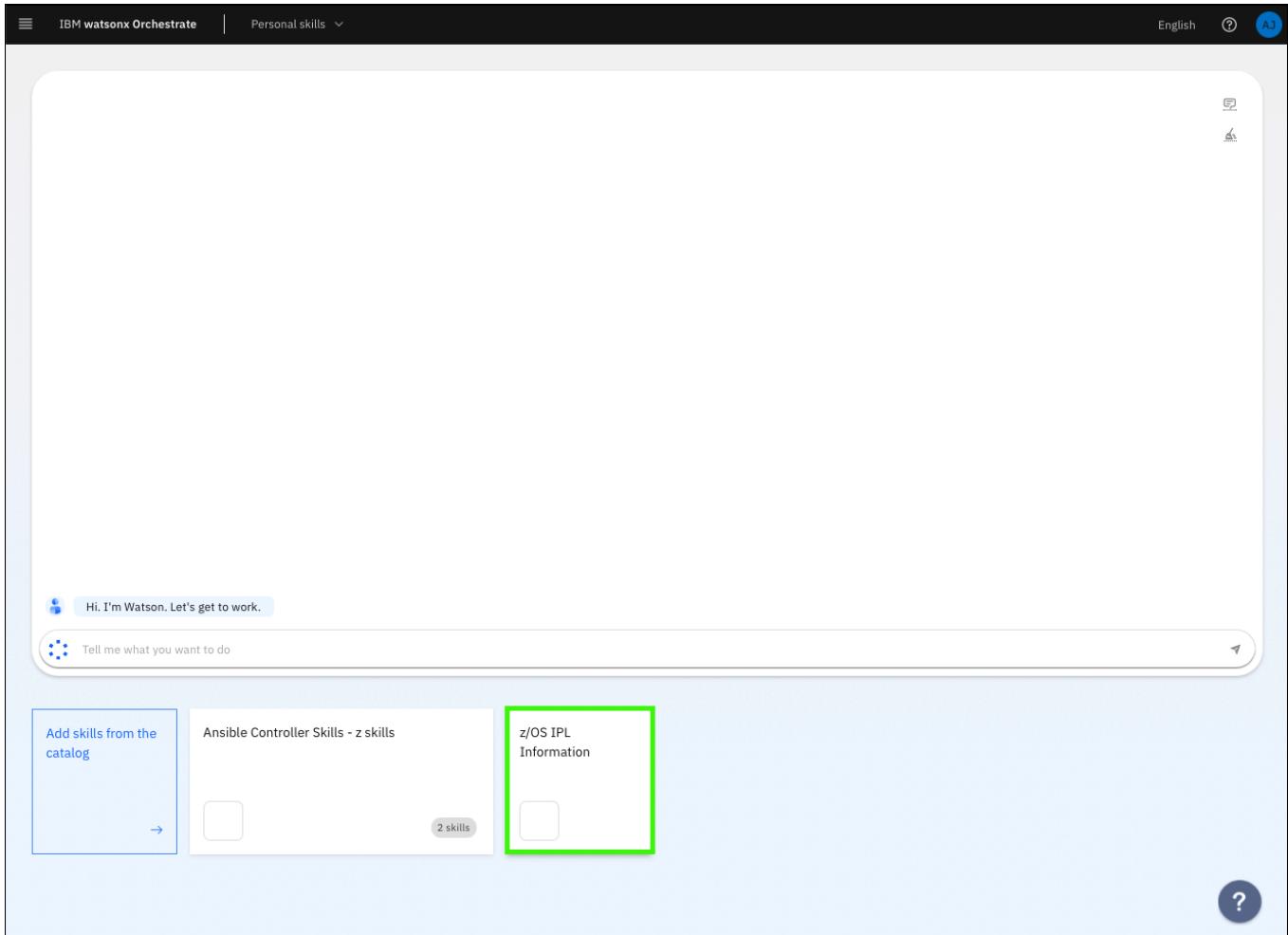
9. Verify that the skill is connected.



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**.

The screenshot shows the IBM Watsonx Orchestrate interface. At the top, it says "IBM Watsonx Orchestrate" and "Personal skills". On the right, there are icons for English, a help symbol, and a user profile. Below the header, a message from Watson says "Hi, I'm Watson. Let's get to work." A section titled "z/OS IPL Information" contains the instruction "You just need to complete this form first." It includes fields for "consoleName \*" (containing "iserVS01"), "cmd \*" (containing "d iplinfo"), and "sol-key \*" (containing "JES"). At the bottom of the form are "Cancel" and "Apply" buttons, with "Apply" being highlighted with a green box. Below the form, a search bar says "Tell me what you want to do" and a question mark icon is visible. At the bottom left, there is a button "Add skills from the catalog" with an arrow pointing right. To the right of the catalog button are two cards: "Ansible Controller Skills - z skills" (with a box icon and "2 skills" label) and "z/OS IPL Information" (with a box icon).

### 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 center, there is a large text area containing the output of a skill named "cmd-response". The output is a series of system logs related to an IPL (Initial Program Load) process. A green dashed rectangular box highlights the first few lines of this output:

```

z/OS IPL Information
sol-key-timeout
sol-key-detected
cmd-response
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 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_75d1
VM EXT NAME = k8s_75d16281c12c4681b91417749e322845_0727_3c9bd7f3-8a47
-4a47-b62c-298da45f40eb

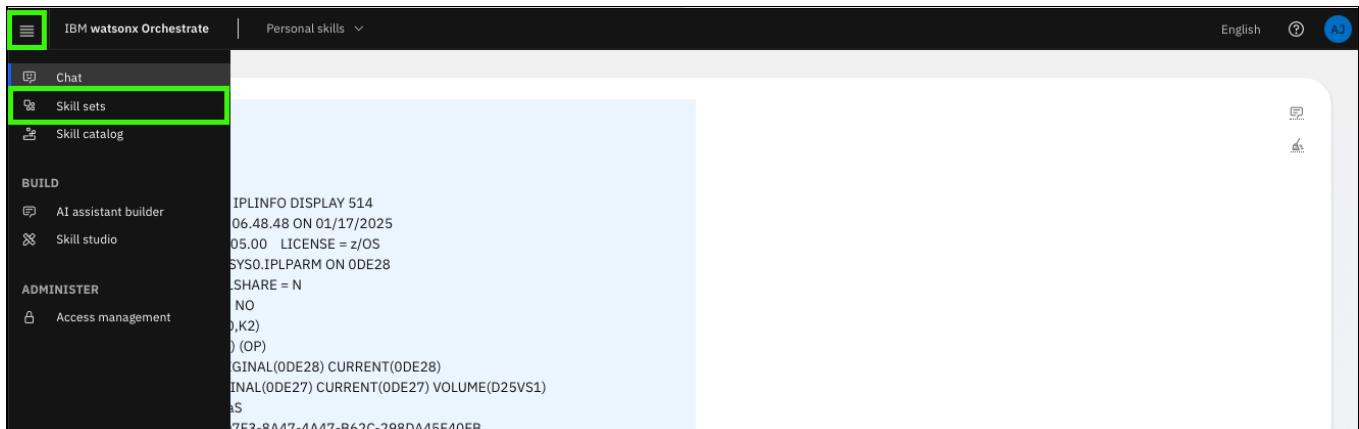
```

Below this text area is a input field labeled "Tell me what you want to do" and a "cmd-response-key" button.

## 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.

**Team Skills**

**Skill sets**

- Team Skills
- Orchestrate Agent Skillset
- Team Skills
- Zeeves draft
- Zeeves live

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.

Name App

No skills

Click **Manage skills** to give this team digest specific abilities.

Manage skills

### 3. Click Connections.

**Zeeves draft**

**Skill sets**

- Zeeves draft

**Skills (2)**  
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.

Name App

Retrieve job output	
Zeeves-gather-facts-flow	

Items per page: 10 ▾ 1-2 of 2 items 1 ▾ of 1 page < >

Manage skills

### 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 1 of 1 page ⏪ ⏪

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

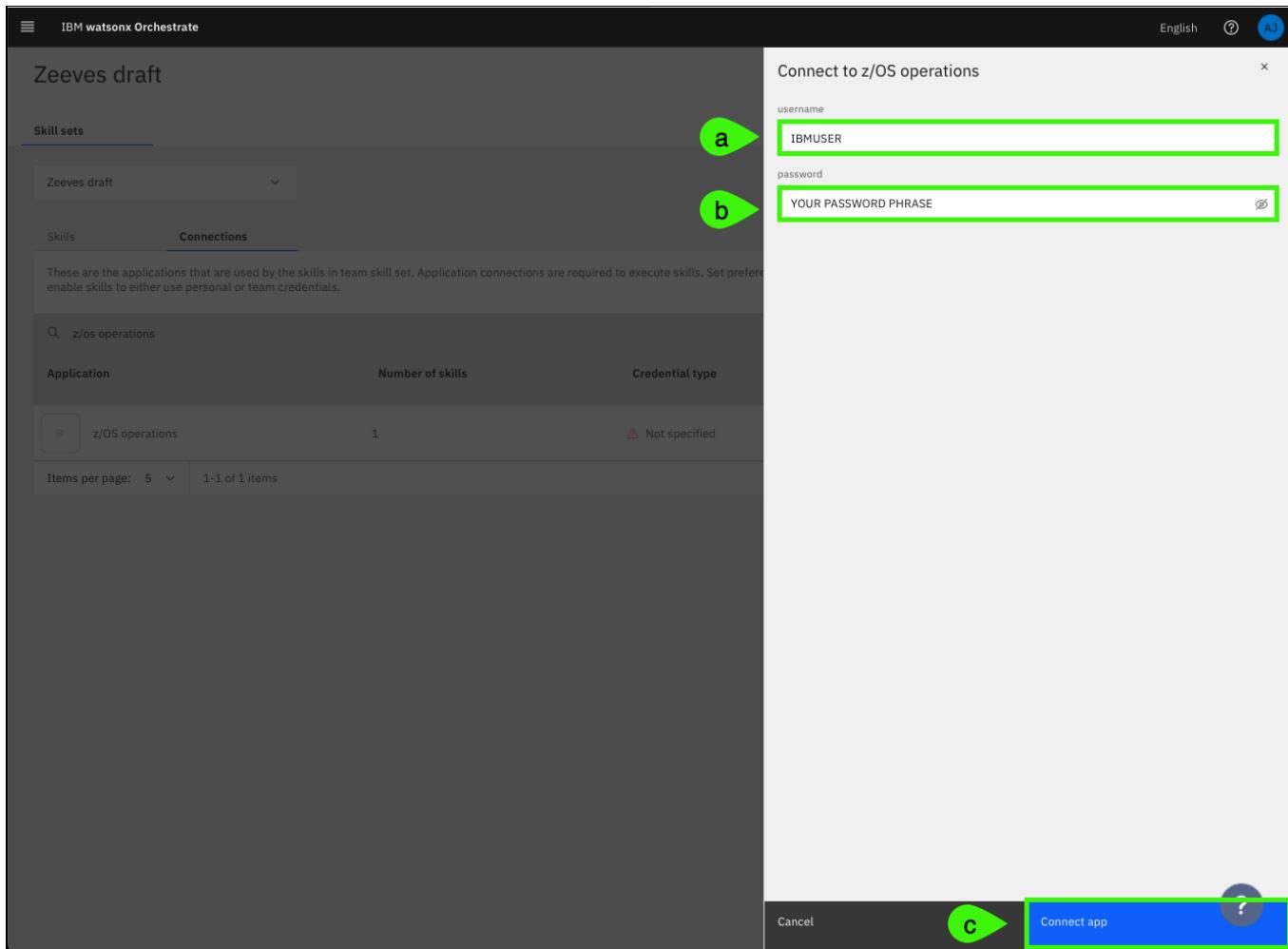
Member credentials  
Each team member uses their own credentials to connect to this app and use its skills.

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.

The screenshot shows the IBM Watsonx Orchestrate interface with the 'AI assistant builder' selected in the sidebar. The main area displays a table of actions:

Action	Number of skills	Credential type	Connected by	Action
Zeeves	1	Team	andrew@jones-tx.com	⋮

Below the table, it says "of 1 items". At the bottom right, there are pagination controls: "1" and "of 1 page" with left and right arrows.

## 2. Select Actions.

The screenshot shows the AI assistant builder interface with the 'Actions' section selected in the sidebar. The main area displays several action cards:

- Build actions**: Enhance and improve your assistant's actions.
- Customize your greeting**: Welcome your users in a unique way that aligns with their behavior.
- 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**: Select what your assistant can do.

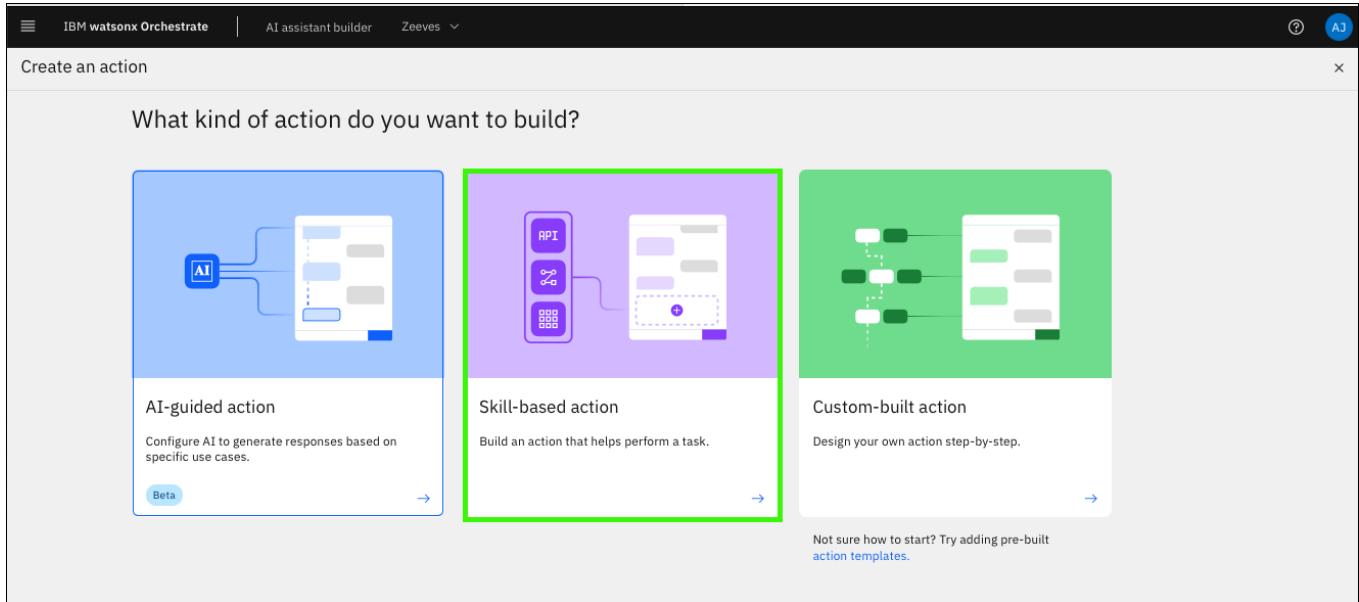
## 3. Click New action +.

The screenshot shows the AI assistant builder interface with the 'Actions' section selected in the sidebar. The main area displays a table of actions:

Name	Last edited	Examples count	Steps count	Status
Zeeves-gather-facts-flow	2 days ago	1	0	✓

At the top right of the table, there is a green button labeled "New action +".

## 4. Click Skill-based action.



## 5. Click **z/OS IPL Information** and then click **Next**.

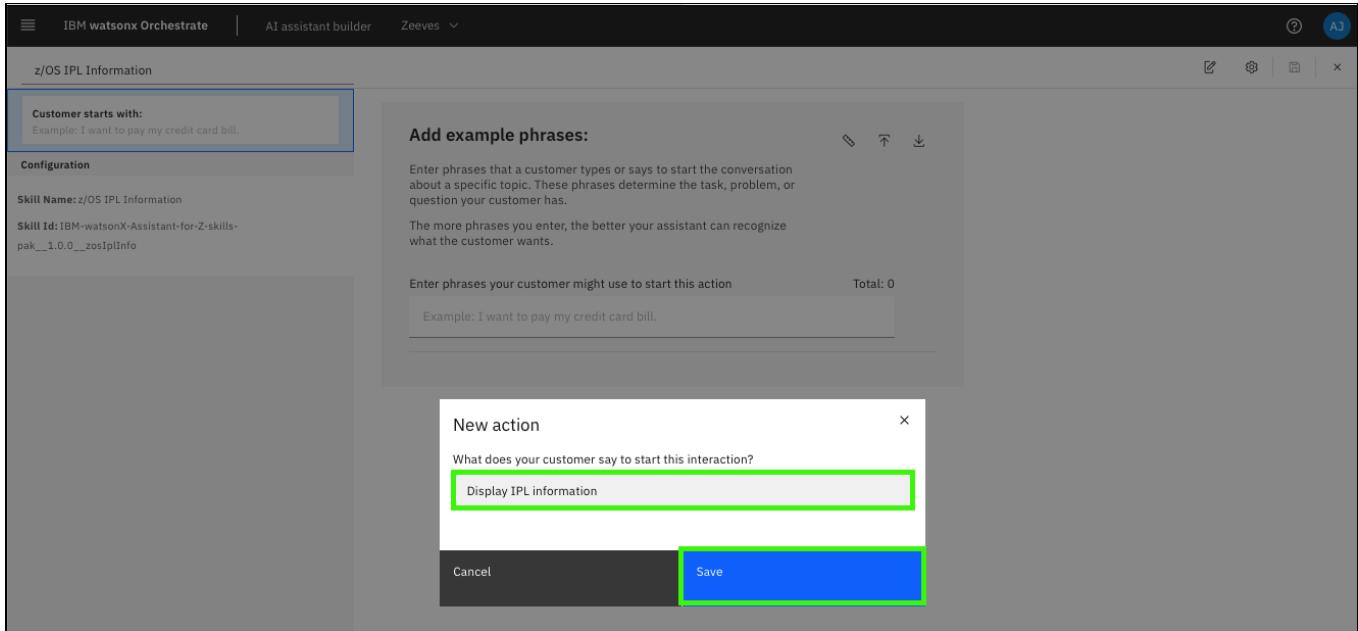
The screenshot shows the 'Build an action from a skill' screen. It lists several skills:

Skill Name	Description	Last Updated
<b>z/OS IPL Information</b>	Run z/OS gather facts and retrieve the output.	2025-01-17T14:39:36.042Z
Zeeves-gather-facts-flow	z skills - Retrieve job output by job Id	2025-01-17T12:24:16.672Z
Retrieve job output	z skills - This playbook pings the z/OS host to test connectivity.	2025-01-17T12:18:14.714Z
<b>z/OS Ping</b>	z skills - This playbook pings the z/OS host to test connectivity.	2025-01-17T12:13:33.194Z
<b>z/OS Gather Facts</b>	z skills - This sample playbook demonstrates the z/OS gather facts module, which pulls z/OS-specific information from the z/OS host.	2025-01-17T14:39:36.042Z
Summarize the Webex meeting transcript	in watsonx.ai	
Summarize the Box content	in watsonx.ai	
Summarize a Zendesk ticket	in watsonx.ai	

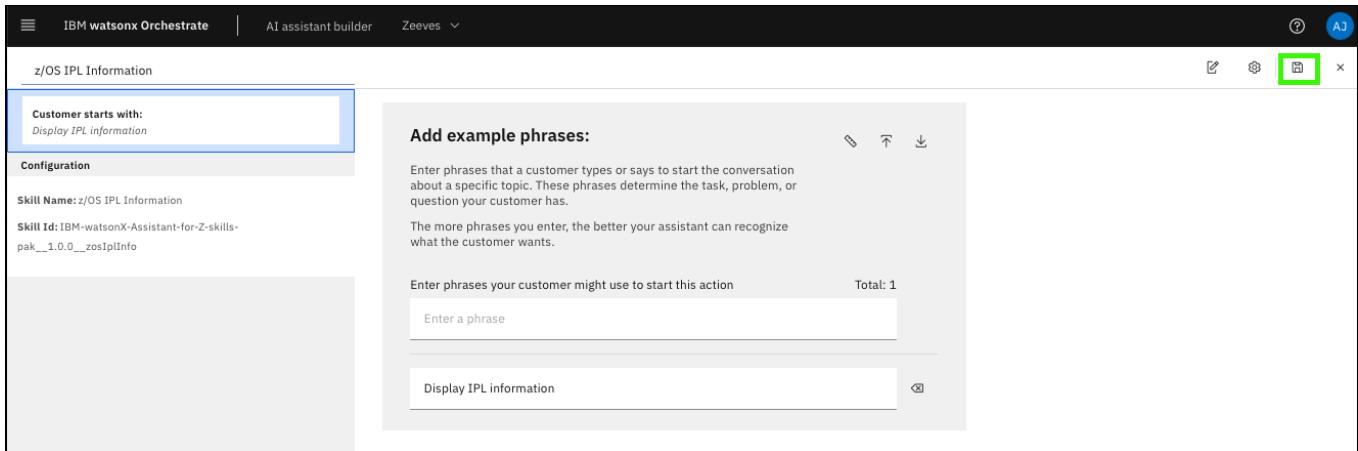
## 6. Enter a prompt like **Display IPL information** that starts the skill and then click **Save**.

Prompt:

Display IPL information



7. Click the **Save** icon (💾).



8. Click **Preview**.

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

Enter a phrase

Display IPL information

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

Enter a phrase

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

Enter a phrase

Display IPL information

**Preview**

9:02 AM  
Conversation skill called  
z/OS IPL Information recognized

z/OS IPL Information

consoleName \*  
iserVS01

cmd \*  
d iplinfo

sol-key \*  
JES

Cancel    Apply

## 11. Verify the results.

**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

Enter a phrase

Display IPL information

**Preview**

z/OS IPL Information

cmd-response  
IEE254I 11.04.51 IPLINFO DISPLAY  
SYSTEM IPLED 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.

**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

Enter a phrase

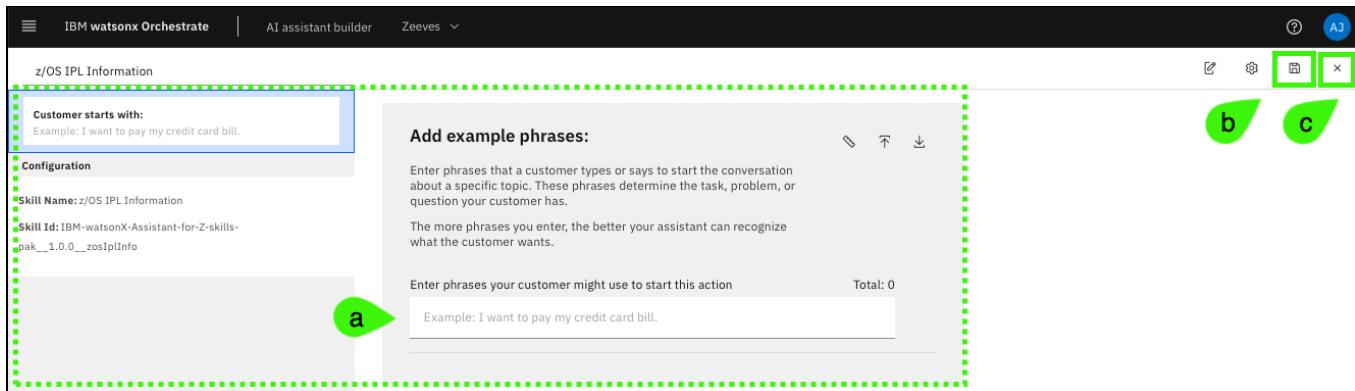
Display IPL information

**Preview**

z/OS IPL Information

cmd-response  
IEE254I 11.04.51 IPLINFO DISPLAY  
SYSTEM IPLED 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

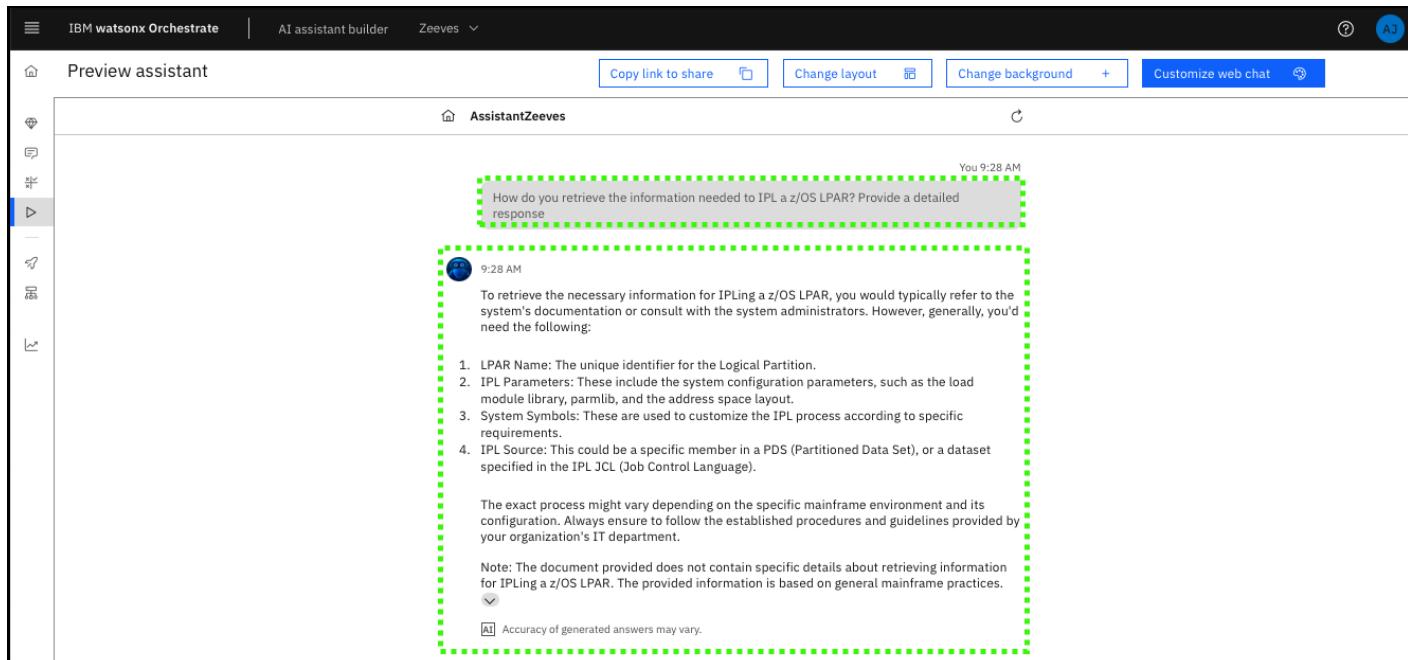
13. Verify (a) all **example phrases** are deleted, (b) click the **Save** icon (💾), and then (c) click **x** to close the action.



## 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 +**.

Actions

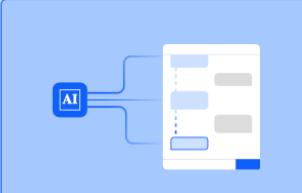
Name	Last edited	Examples count	Steps count	Status
<a href="#">z/OS IPL Information</a>	17 minutes ago	0	3	Active
<a href="#">Zeeves-gather-facts-flow</a>	2 days ago	1	0	Active

New action +

## 2. Click Custom-built action.

Create an action

What kind of action do you want to build?



AI-guided action

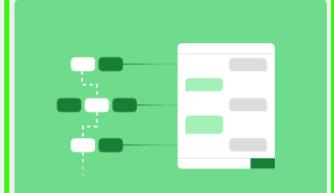
Configure AI to generate responses based on specific use cases.

Beta



Skill-based action

Build an action that helps perform a task.



Custom-built action

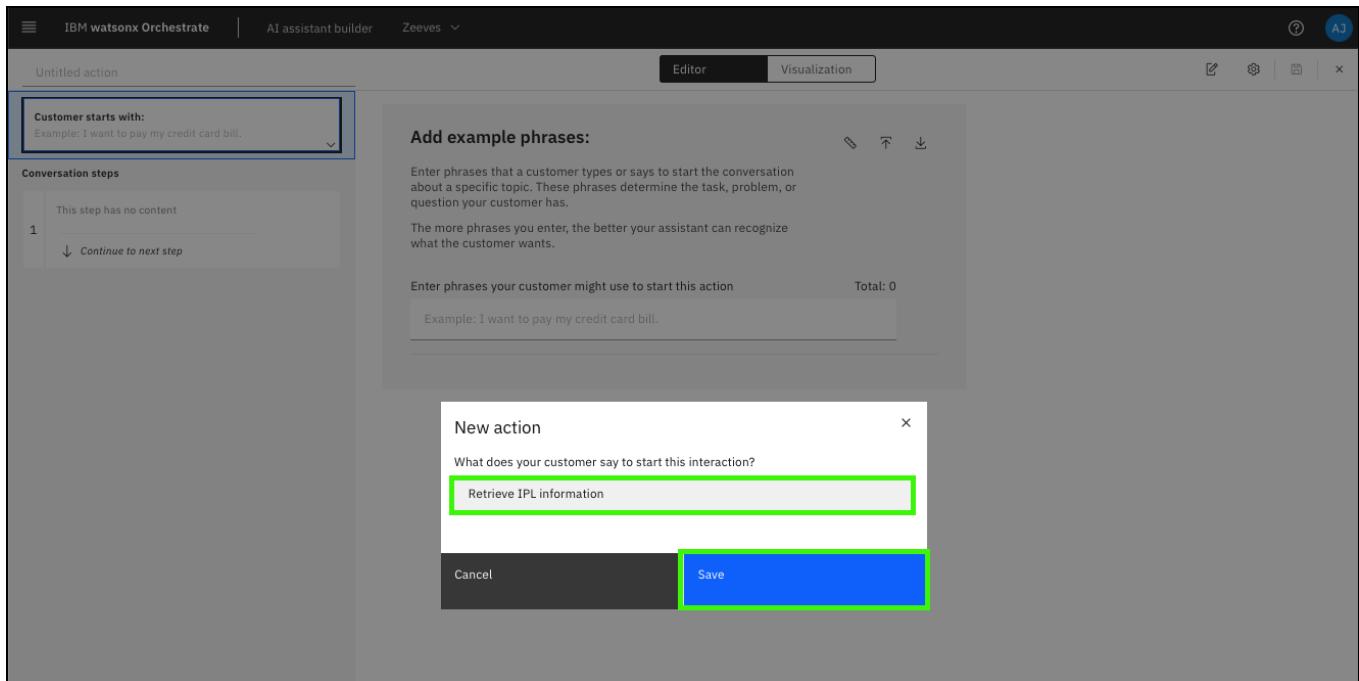
Design your own action step-by-step.

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 Set variable values

Assistant says

For example: Please select from the following options:

Define customer response

And then

Continue to next step  
Re-ask previous step(s)  
Go to a subaction  
Use an extension  
**Search for the answer**  
Connect to agent  
End the action

## 6. Click **Edit settings**.

Customer starts with:  
Retrieve IPL information

Conversation steps

Step 1

Is taken without conditions Set variable values

Assistant says

For example: Please select from the following options:

Define customer response

And then

**Search for the answer**

Custom query	None (Customer's message will be the query)
Custom filter	None (Optional)
<b>Edit settings</b>	

## 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

The screenshot shows the IBM Watsonx Orchestrator interface. In the top navigation bar, it says "IBM Watsonx Orchestrator" and "AI assistant builder". The main area is titled "Retrieve IPL information". A "Conversation steps" section shows one step labeled "1 This step has no content". A search bar contains the placeholder "Search for the answer". Above the search bar, there are tabs for "Step 1" and "without conditions". To the right of the search bar are "Editor" and "Visualization" buttons, along with some icons. A modal window titled "Search for the answer" is open. It has tabs for "Search configuration" (which is selected) and "After generation". The "Search configuration" tab includes a "Custom search query (Optional)" input field containing the text "How do you retrieve the information needed to IPL a z/OS LPAR? Provide a detailed response". Below this is a "Custom results filter" section with a radio button for "Use default" (selected) and another for "Set new filter". A note says "The default filter is set in the search integration settings page." An example of a custom filter is shown: "Example: enriched\_text.entities.text.:\"Boston, MA\"". At the bottom of the modal are "Cancel" and "Apply" buttons, with "Apply" highlighted.

#### 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 Orchestrate AI assistant builder 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 'Retrieve IPL information'. On the left, under 'Conversation steps', there is a step labeled '1' with a placeholder 'This step has no content' and a search bar 'Search for the answer'. To the right, 'Step 1' is set to 'Is taken without conditions'. The 'Assistant says' section contains a rich text editor with a placeholder 'For example: Please select from the following options:' and a dropdown menu 'Define customer response'. Below this, 'And then' is set to 'Search for the answer', with a custom query 'How do you retrieve the information needed to IPL a z/OS LPAR? Provide a detailed response'. A green box highlights the 'New step +' button at the bottom left.

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 continuation of the AI assistant builder interface. It's now Step 2. The 'Conversation steps' section shows two steps: Step 1 (with its previous content) and Step 2, which has a green dashed border around its content area. Step 2 contains a question 'Would you like to display your LPARs IPL information?' and a 'Continue to next step' button below it. The 'Assistant says' section for Step 2 also contains a green box highlighting the text 'Would you like to display your LPARs IPL information?'. The 'And then' section for Step 2 is set to 'Continue to next step'.

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 AI assistant builder interface. In the top navigation bar, it says "IBM Watsonx Orchestrate" and "AI assistant builder". The workspace title is "Zeeves". The main area displays a conversation flow:

- Customer starts with:** "Retrieve IPL information".
- Conversation steps:**
  - Step 1: "This step has no content".
  - Step 2: "Would you like to display your LPARs IPL information?". This step is highlighted with a blue border.
- Assistant says:** "Would you like to display your LPARs IPL information?"

Below the conversation steps, there is a configuration panel titled "Define customer response" with a green border. It includes the following settings:

- System:** "Confirmation" is selected from a dropdown menu.
- Options:** A note states: "Enable customers to select from a set of choices. If more than 4, options show as a list." Below this are two sections:
  - As buttons:** Buttons labeled "Checking", "Savings", "401 (k)", and "Roth IRA".
  - As a list:** A single item "Pay Bill".

11. Click **Next step +**.

The screenshot shows the IBM Watsonx Orchestrate interface. In the top navigation bar, it says "IBM watsonx Orchestrate" and "AI assistant builder". The main area is titled "Retrieve IPL information". On the left, there's a sidebar for "Conversation steps" with two steps defined: Step 1 (This step has no content) and Step 2 (Would you like to display your LPARs IPL information? with a "Confirmation" button). Step 2 is currently selected. The main panel shows "Step 2" with the condition "Is taken without conditions". Below this, the "Assistant says" section contains the message "Would you like to display your LPARs IPL information?" followed by a green dashed box containing "Yes" and "No" buttons. At the bottom, there's an "And then" section with a "Continue to next step" button.

## 12. Click the **Is taken** drop-down and select **with conditions**.

This screenshot shows the same interface after Step 2 has been modified. The "Is taken" dropdown in Step 3 now has "without conditions" highlighted and "with conditions" selected. The "Assistant says" section now contains the message "For example: What size do you want to order?". The "And then" section remains the same.

## 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 AI assistant builder interface. At the top, there are tabs for 'Editor' and 'Visualization'. The main area displays a conversation flow with three steps:

- Step 1:** 'Customer starts with: Retrieve IPL information'.
- Step 2:** 'Would you like to display your LPARs IPL information?' (Confirmation step). It includes a condition: 'If All of this is true: 2. Would you like to d... is Yes'.
- Step 3:** 'This step has no content'.

The condition for Step 2 is highlighted with a green dashed box. Below the conditions, there are buttons for 'Set variable values' and 'New condition group'. The 'Assistant says' section contains the placeholder text 'For example: What size do you want to order?'. The 'Define customer response' section is collapsed. At the bottom, there is a 'And then' section with a 'Continue to next step' button.

14. Enter **Retrieving your system's IPL information...** in the **Assistant says** field.

Assistant says:

Retrieving your system's IPL information...

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. At the top, there are tabs for "Editor" and "Visualization". The main area displays a conversation flow with three steps:

- Step 1:** "Customer starts with: Retrieve IPL information".
- Step 2:** "Would you like to display your LPARs IPL information?". It includes a "Confirmation" button and a "Continue to next step" link.
- Step 3:** "2 is Yes". Below it, the text "Retrieving your system's IPL information..." is shown, followed by a "Continue to next step" link.

On the right side, the "Step 3" editor is open, showing the condition "Is taken with conditions" and a condition block:

```
If All of this is true:  
2. Would you like to d... is Yes  
and Add condition +
```

Below this, there is a "New condition group +" button and an "Assistant says" section containing the text "Retrieving your system's IPL information...". A green box highlights this text. Below it is a "Define customer response" dropdown and a "And then" section with a "Continue to next step" link.

15. Click the **And then** drop-down and select **Go to a subaction**.

Step 3

Is taken with conditions

Conditions

If All of this is true:

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**.

Go to a action

Go to

z/OS IPL Information (highlighted)

End this action after the b action is completed

Cancel Apply

17. Click **Save** (💾) and then click **x**.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with 'Customer starts with: Retrieve IPL information' and a 'Conversation steps' section. Step 1: 'This step has no content'. Step 2: 'Would you like to display your LPARS IPL information?' with a 'Confirmation' button. Step 3: 'Retrieving your system's IPL information...' with a link to 'z/OS IPL Information'. The main area shows 'Step 3' with a condition 'Is taken with conditions' set to 'with conditions'. A condition is defined: 'If All of this is true: 2. Would you like to d... is Yes'. Below this, there's a 'New condition group +' button and an 'Assistant says' section with rich text editor icons.

## 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' menu in the sidebar with 'Preview' highlighted. The main area displays a table of actions created by the user:

Name	Last edited	Examples count	Steps count	Status
<a href="#">z/OS IPL Information</a>	an hour ago	0	3	Green checkmark
<a href="#">Zeeves-gather-facts-flow</a>	2 days ago	1	0	Green checkmark
<a href="#">Retrieve IPL information</a>	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 IBM Watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The left sidebar has various icons for managing assistants. The main preview area is titled 'Preview assistant' for 'AssistantZeeves'. It features a toolbar with 'Copy link to share', 'Change layout', 'Change background', and 'Customize web chat'. Below the toolbar, the assistant's profile picture is displayed, followed by its greeting: 'Hi! I'm Zeeves. How can I help you today?'. At the bottom of the preview area, a message input field contains the question: 'How do you retrieve the information needed to IPL a z/OS LPAR?'. This message is highlighted with a green border.

### 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.

Preview assistant

AssistantZeeves

You 10:33 AM

How do you retrieve the information needed to IPL a z/OS LPAR?

10:33 AM

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.

AI Accuracy of generated answers may vary.

Would you like to display your LPARs IPL information?

Type something... ➤

#### 4. Click Yes.

Preview assistant

AssistantZeeves

You 10:33 AM

How do you retrieve the information needed to IPL a z/OS LPAR?

10:33 AM

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.

AI Accuracy of generated answers may vary.

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. The main area displays an AI-generated response titled "AssistantZeeves". The response discusses retrieving information from the IPL JCL and provides a note about the general nature of the information. It asks if the user would like to display their LPAR's IPL information, with "Yes" and "No" buttons. A message from the AI at 10:37 AM indicates it is retrieving system IPL information. Below this, a modal window titled "z/OS IPL Information" shows form fields for "consoleName" (set to "iservS01"), "cmd" (set to "d iplinfo"), and "sol-key" (set to "JES"). The "Apply" button in the modal is highlighted with a green border.

## 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 sub-section titled "AssistantZeeves". A green dotted box highlights a block of text:

```

z/OS IPL Information
sol-key-timeout
sol-key-detected
cmd-response
IEE254I 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)
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_75d16281c12c4681b91417749e322845_0727_3c9bd
8a47
-4a47-b62c-298da45f40eb

cmd-response-key
C4443879

```

At the bottom of the preview area, it says "10:39 AM". Below the preview area 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.

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(s)	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.

Name	Type	Organization	Last Ran	Actions
z/OS Certs - Add CA Cert	Job Template	Default	1/17/2025, 10:09:29 AM	
z/OS Certs - Create Cert	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 in the Red Hat Ansible Automation Platform. The 'Survey' tab is selected. The form fields are as follows:

- Owner ID: IBMUSER
- Certificate Label: TESTCA
- Type: CERTAUTH
- Sign With: Self
- Sign Label: (empty)
- Organizational Unit: ANSIBLE CORE

At the bottom of the survey step, there are three buttons: '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.

	Name	Description	Type	Job Template
1 Survey	z/OS Certs - Create Cert	This playbook creates a SITE, USER or CERTAUTH certificate. Signed by local CA or self-signed.		
2 Preview				
	Timeout	0 min 0 sec	Job Type	Run
	Inventory	AAP z/OS	Project	aap4zos
	Playbook	zos_concepts/certificate_management/create_cert.yml	Forks	0
	Show Changes	Off	Job Slicing	1
	Credential	SSH: z/OS Host SSH ...		

**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 named "16 - 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 includes tasks like getting FQDN, setting facts, and checking for certificates. A green dashed box highlights the line "ok: [zos\_host]".

```
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 }
```

7. Locate the line **TASK [GENERATE new certificate]**, click the **changed: [zos host]**.

Jobs > 16 - z/OS Certs - Create Cert

**Output**

z/OS Certs - Create Cert Successful

Plays 1 Tasks 15 Hosts 1 Elapsed 00:00:20

Stdout Search

Events processing complete. Reload output

```

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.

Jobs > 16 - z/OS Certs - Create Cert

**Output**

z/OS Certs - Create Cert Successful

Plays 1 Tasks 15 Hosts 1 Elapsed 00:00:20

**Host Details**

Details JSON

**Host** zos\_host **Status** Changed

**Task** Generate new certificate. **Module** ibm.ibm\_zos\_core.zos\_tso\_comm and

Play

Check if SITE cert and keyring already exist. If not, create them and then run health check via TSO and operator commands.

## 9. Review the RACDCERT command that was run to generate the certificate and then click x to close the window.

The screenshot shows the Red Hat Ansible Automation Platform interface. The left sidebar has sections like Views, Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals, Host Metrics, Subscription Usage, Resources, Templates, Credentials, Projects, Inventories, Hosts, Access, Organizations, Users, and Teams. The 'Jobs' section is currently selected. In the center, under 'Jobs', it says 'Jobs > 16 - z/OS Certs - Create Cert'. Below that is the 'Output' tab. A modal window titled 'Host Details' is open, showing the 'Details' tab selected. It contains a JSON dump of the task's execution. The command run was 'RACDCERT GENCERT CERTAUTH KEYUSAGE(CERTSIGN) SUBJECTSDN(CN('itzvsi-zos-pwg'))'. The output shows the command was successful with a rc of 0 and no stderr. The timestamp for this output is 12:58:32. To the right of the modal, there are several other log entries with timestamps from 12:58:38 to 12:58:41.

## 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
z/OS Certs - Add CA Cert	Job Template	Default	1/17/2025, 10:09:29 AM	
z/OS Certs - Create Cert	Job Template	Default	1/19/2025, 12:58:44 PM	
z/OS Certs - Create Keyring	Job	Default		

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!

**Views**

**Launch | z/OS Certs - Create Cert**

This playbook creates a SITE, USER or CERTAUTH certificate. Signed by local CA or self-signed.

**Survey**

**Common Name** company.com

**Country** US

**Keyring** SharedRing1

**Expiration Date** 2025-01-30

**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.

**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		

**Promoted Values**

**Variables** **YAML** **JSON**

```

2 cert_label_survey: TESTSITE
3 cert_type_survey: SITE
4 sign_with_survey: CERTAUTH
5 sign_label_survey: TESTCA

```

**Launch** **Back** **Cancel**

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 '17 - z/OS Certs - Create Cert' with a status of 'Successful'. The 'Output' tab is selected, showing the command-line logs. The logs indicate a task to check if a certificate already exists, debug command output, generate a new certificate, and list the chain for the new certificate. The entire log area is highlighted with a green dashed border.

```

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.

The screenshot shows the Red Hat Ansible Automation Platform web interface. On the left, there's a dark sidebar with several sections: '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 'Templates' link under 'Resources' is highlighted with a green box. In the main content area, a list of job templates is shown. One specific template, 'z/OS Certs - Search and Renew', is also highlighted with a green box. The columns in the list include: Name, Type, Status, Last Run, and three action icons (eye, edit, delete).

Name	Type	Status	Last Run	Action
<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 Security</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!

Views

Dashboard

Jobs

Schedules

Activity

Workflow

Host Metrics

Subscriptions

Resources

Templates

Credentials

Projects

Inventories

Hosts

Access

Organizations

Users

Teams

Administration

z/OS Certs - Add CA Cert

Job Default Template

1/1/2025, 10:09:29 AM

admin

Launch | z/OS Certs - Search and Renew

This playbook searches for certificates with the details specified in the AAP survey and renews them.

1 Survey

2 Preview

Certificate Label \* ⓘ  
TESTSITE

Type \* ⓘ  
SITE

Sign With \* ⓘ  
CERTAUTH

Sign Label ⓘ  
TESTCA

Owner ID \* ⓘ  
IBMUSER

Keyring \* ⓘ  
SharedRing1

Next Back 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.

Setting	Value
Name	z/OS Certs - Search and Renew
Description	This playbook searches for certificates with the details specified in the AAP survey and renews them.
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/search_and_renew.yml
Forks	0
Verbosity	0 (Normal)
Show Changes	Off
Job Slicing	1
Credential	SSH: z/OS Host SSH ...

**Buttons:** Launch (highlighted), 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 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), and Administration (Credential Types). The 'Jobs' section is currently selected.

The main area is titled 'Output' for a job named 'z/OS Certs - Search and Renew'. It shows a summary: Plays 1, Tasks 18, Hosts 1, Elapsed 00:00:32. A green box highlights the 'Successful' status. Below this, there's a 'Stdout' tab with a search bar and a message: 'Events processing complete' with a 'Reload output' button. The terminal output shows several lines of log data, with lines 16 and 17 highlighted in green boxes.

```

16 skipping: [zos_host] => (item={'ddname': 'JESJCL', 'record_count': '8', 'id': '3', 'stepname': 'JES2', 'proctep': None, 'byte_count': '136', 'content': ['      1 //HZSPRINT JOB RACF CERT EXPIRATION CHECK', '          JOB00225', '          // MSGCLASS=H,MSGLEVEL=(1,1),CLASS=A,', '          // NOTIFY=&SYSUID', '          IEFC653I SUBSTITUTION JCL - RACF CERT EXPIRATION CHECK', 'MSGCLASS=H,MSGLEVEL=(1,1),CLASS=A,NOTIFY=IBMUSER', '          2 //HZSPRINT EXEC PGM=HZSPRNT,TIME=100,REGION=0M,PARMDD=SYSIN', '          3 //SYSIN DD *,DLM='@'', '          4 //SYSOUT DD SYSOUT=A,DCB=(LRECL=256)', '          5 //'])}
17 skipping: [zos_host] => (item={'ddname': 'JESYSMSG', 'record_count': '18', 'id': '4', 'stepname': 'JES2', 'proctep': None, 'byte_count': '137', 'content': [' ICH70001I IBMUSER LAST ACCESS AT 15:26:46 ON SUNDAY, JANUARY 19, 2025', ' IEFA111I HZSPRINT IS USING THE FOLLOWING JOB RELATED SETTINGS:', ' SWA=BELLOW,TIOT SIZE=64K,DSENQSHR=DISALLOW,GDGBIAS=JOB', ' IEF236I ALLOC. FOR HZSPRINT HZSPRINT', ' IEF237I JES2 ALLOCATED TO SYSIN', ' IEF237I JES2 ALL LOCATED TO SYSOUT', ' IEF142I HZSPRINT HZSPRINT - STEP WAS EXECUTED - COND CODE 0000', ' IEF285I IBMUSER.HZSPRINT.JOB00225.D0000101.? SYSIN', ' IEF285I IBMUSER.HZSPRINT.JOB00225.D0000102.? SYSOUT', ' IEF373I STEP/HZSPRINT/START 2025019.1526', ' IEF032I STEP/HZSPRINT/STOP 2025019.1526', ' CPU: 0 HR 00 MIN 00.00 SEC SRB: 0 HR 00 MIN 00.00 SEC', ' VIRT: 12K SYS: 304K EXT: 24K SYS: 9748K', ' ATB- REAL: 228K SLOTS: 0K', ' VIRT- ALLOC: C: 13M SHRD: 0M'. ' IEF375I JOB/HZSPRINT/START 2025019.1526'. ' IEF033I JOB/HZSPRINT/STOP 2025019.1526'])

```

## 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 'Templates' page in the Red Hat Ansible Automation Platform. The left sidebar has the 'Templates' tab selected, indicated by a green box. The main area displays a list of templates with columns: Name, Type, Organization, Last Ran, and Actions. Two templates are listed: 'z/OS Certs - Add CA Cert' (Job Template, Default, 1/17/2025, 10:09:29 AM) and 'z/OS Certs - Create Cert' (Job Template, Default, 1/19/2025, 1:16:45 PM). The 'z/OS Certs - Create Cert' template is highlighted with a green box, and its 'Actions' column contains a green box around the launch icon (✿).

Name	Type	Organization	Last Ran	Actions
z/OS Certs - Add CA Cert	Job Template	Default	1/17/2025, 10:09:29 AM	
z/OS Certs - Create Cert	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**

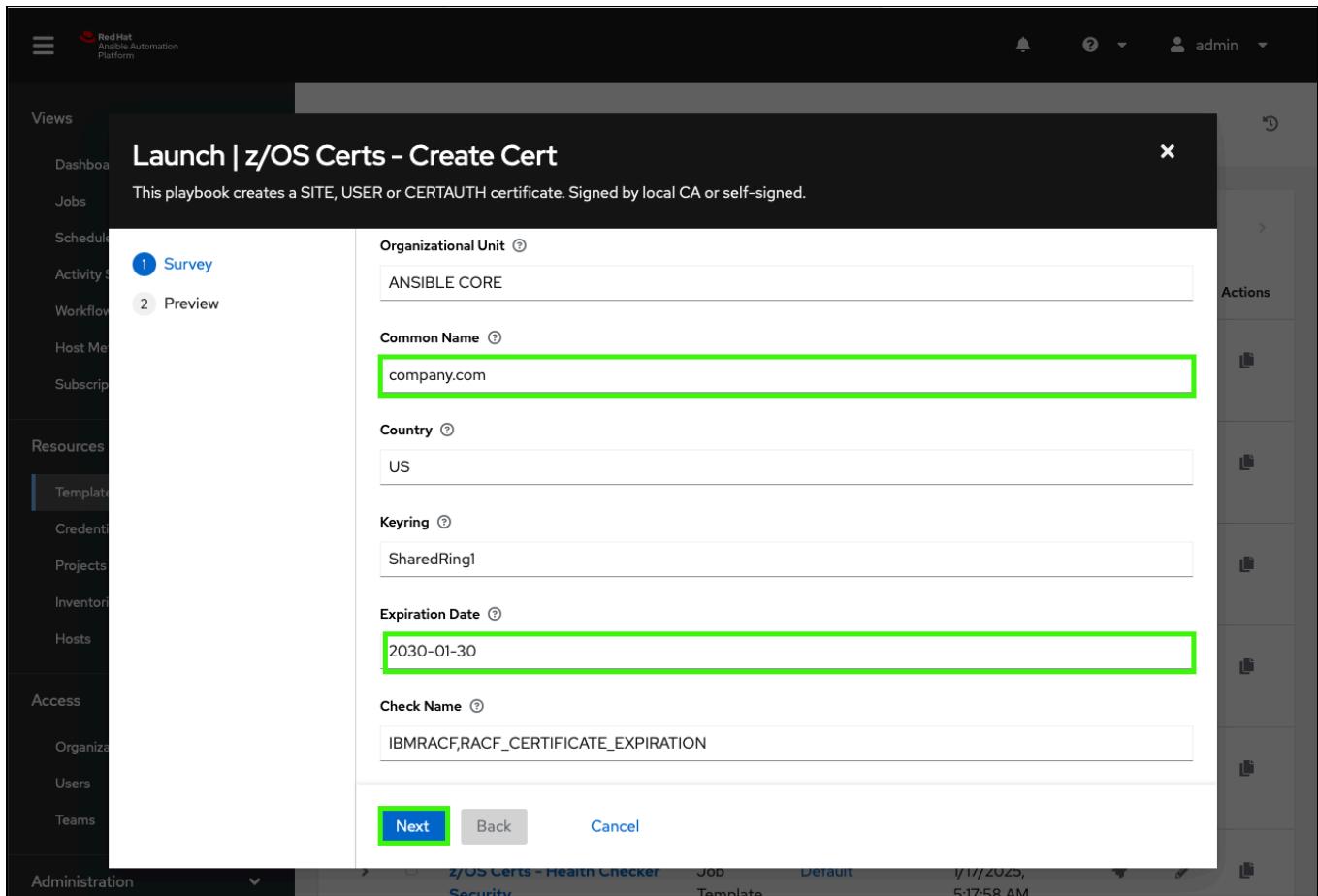
company.com

**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!



3. Click **Launch**.

The screenshot shows the Red Hat Ansible Automation Platform interface. A modal dialog titled "Launch | z/OS Certs - Create Cert" is open. The dialog content includes:

- 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
1	owner_id_survey: IBMUSER	
2	cert_label_survey: DEMOCERT	
3	cert_type_survey: SITE	
4	sign_with_survey: CERTAUTH	

**Buttons:** Launch (highlighted with a green box), Back, Cancel.

4. Verify that the **DEMOCERT** was successfully created.

The screenshot shows the Red Hat Ansible Automation Platform interface. On the left, there's a sidebar with sections like Views, Dashboard, Jobs (which is selected), Schedules, Activity Stream, Workflow Approvals, Host Metrics, and Subscription Usage. Below that are Resources (Templates, Credentials, Projects, Inventories, Hosts), Access (Organizations, Users, Teams), and Administration (Credential Types). The main area shows a job named '19 - z/OS Certs - Create Cert'. The 'Output' tab is active, showing the job's progress. At the top of the output pane, it says 'z/OS Certs - Create Cert' and 'Successful'. Below that, it shows 'Plays 1', 'Tasks 15', 'Hosts 1', and 'Elapsed 00:00:25'. The output itself starts with 'Events processing complete' followed by a series of Ansible task logs numbered 15 to 35, detailing steps like skipping a host, checking for existing certificates, generating new ones, and listing chains.

## 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.

The screenshot shows the IBM Watsonx Orchestrate web interface. On the left, there's a sidebar with sections for Chat, Skill sets, Skill catalog, BUILD (with AI assistant builder and Skill studio), and ADMINISTER (Access management). The main area is titled "AssistantZeeves". At the top right, there are buttons for "Copy link to share", "Change layout", "Change background", and "Customize web chat".

## 2. Click **Create** and then click **Import API**.

The screenshot shows the "Skill studio" interface within Watsonx. The top navigation bar includes "Skill studio" and language settings. A "Create" dropdown menu is open, with "Import API" highlighted by a green box. Below the menu, there are tabs for Skills and Apps, a search bar, and a table listing skills.

Name	Step in the process	Status	Skill type	Author	Last edited
z/OS IPL Information	Ready to use	<span>Published</span>	Imported	andrew@jones-tx.com	January 17 2025
Zeeves-gather-facts-flow	Ready to use	<span>Published</span>	Skill flow	andrew@jones-tx.com	January 17 2025

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

The screenshot shows the "Add skills" interface. It has tabs for "Choose the source" and "Select the skills". Under "Choose the source", there are options like "From an app", "From a file", and "OpenAPI builder (experimental)". A list of tiles is shown, with the "z/OS Skills accelerator (Trial)" tile highlighted by a green box. Other tiles include "IBM Cloud Pak for Business Automation - On premises", "IBM Cloud Pak for Business Automation as a service", "UiPath", "IBM Robotic Process Automation (RPA)", "IBM Business Automation Workflow as a service", and "IBM Operational Decision Manager as a service".

## 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:

\*

**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**

To discover new skills, connect to an app or refer to an OpenAPI file.

**From an app**   **From a file**   **OpenAPI builder (experimental)**

**z/OS Skills accelerator (Trial)**  
Import your Ansible, JCL/REXX automation to watsonx Orchestrate

**a** Connection Type: ansible  
Either 'ansible' or 'zosmf'

**b** Application Name: certs  
Short application name that uniquely identifies the connection url. Use the same name you previously used if you are importing additional skills from the same connection url. Must contain only alphanumeric characters, spaces, dashes, and underscores.

**c** Connection URL: https://itzvsi-aap-pwgabob.techzone.ibm.com  
For example https://example.com

**d** User name: admin

**e** Password: .....

**f** Search Pattern: \*

**g** Connect

## 5. Expand the Ansible Job Template Proj... folder and then click aap4zos.

**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 0 skills are selected.

- Ansible Job Template Proj...
  - Demo Project
  - z\_ansible\_collections\_sa...
  - ansible\_zos
  - aap4zos**
- Ansible Utility Skills

## 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.

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

Select the skills

Select the skills you want to add to the skill set. Currently, 2 of 20 skills are selected.

Skill	Description	Status
<input type="checkbox"/> 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
<input type="checkbox"/> Z/os tso command(s)	Certs - a playbook to execute ...	Ready to add
<input type="checkbox"/> 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.

Create

Skills     Apps

Find a skill

Configure prebuilt skills

**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	<input checked="" type="radio"/> Ready to publish	Imported	andrew@jones-tx.com	January 19 2025	⋮
z/OS Certs - List Cert	Just 1 step away to be ready	<input checked="" type="radio"/> Ready to publish	Imported	andrew@jones-tx.com	January 19 2025	⋮
z/OS IPL Information	Ready to use	<input checked="" type="radio"/> Published	Imported	andrew@jones-tx.com	January 17 2025	⋮
Zeeves-gather-facts-flow	Ready to use	<input checked="" type="radio"/> Published	Skill flow	andrew@jones-tx.com	January 17 2025	⋮
Retrieve job output	Ready to use	<input checked="" type="radio"/> Published	Imported	andrew@jones-tx.com	January 17 2025	⋮
z/OS Ping	Ready to use	<input checked="" type="radio"/> Published	Imported	andrew@jones-tx.com	January 17 2025	⋮
z/OS Gather Facts	Ready to use	<input checked="" type="radio"/> Published	Imported	andrew@jones-tx.com	January 17 2025	⋮

8. Review the skill enhancement options and then click **Publish**.

**Name\***

z/OS Certs - List Cert

**Description**

certs - This playbook displays a certificate's details.

**API version\***

1.0

**Categories**

Add categories

**App**

Ansible Controller Skills - certs

**Preview**

The skill will look like this in the catalog.

**z/OS Certs - List Cert**  
certs - This playbook displays a certificate's details.

The skill will look like this in the skill set.

**z/OS Certs - List Cert**  
Cert

Cancel Publish 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 January 19 2025
	Ready to use	Published	Imported	andrew@jones-tx.com January 19 2025
	Ready to use	Published	Imported	andrew@jones-tx.com January 19 2025

## 2. Enter certs in the search bar.

The screenshot shows the 'Skill catalog' page with a search bar containing 'certs'. Below the search bar, there's a heading 'Most popular skills' followed by three tiles: 'Send an email from Gmail', 'Create a lead in Salesforce', and 'Send an email using Outlook'. Under 'All Apps', there are four app cards: 'z/OS operations', 'Coupa', 'ZoomInfo', and 'HubSpot CRM'.

## 3. Click the certs tile.

The screenshot shows the 'Skill catalog' page again, but now the 'Ansible Controller Skills - certs' tile is highlighted with a green box. This tile indicates there are 3 skills available.

## 4. Click Add skill + for each of the 3 skills in the certs app.

Skill catalog /

## Ansible Controller Skills - certs (3)

[Connect app](#)

[Personal skills](#)

Search skills

Ansible Controller Skills - certs

- Retrieve job output  
certs - Retrieve job output by job Id
- [Add skill +](#)

- z/OS Certs - List Cert  
certs - This playbook displays a certificate's details.
- [Add skill +](#)

- z/OS Certs - Search and Renew  
certs - This playbook searches for certificates with the details specified in the AAP survey and renew...
- [Add skill +](#)

### 5. Click Connect app.

Skill catalog /

## Ansible Controller Skills - certs (3)

[Connect app](#)

[Personal skills](#)

Search skills

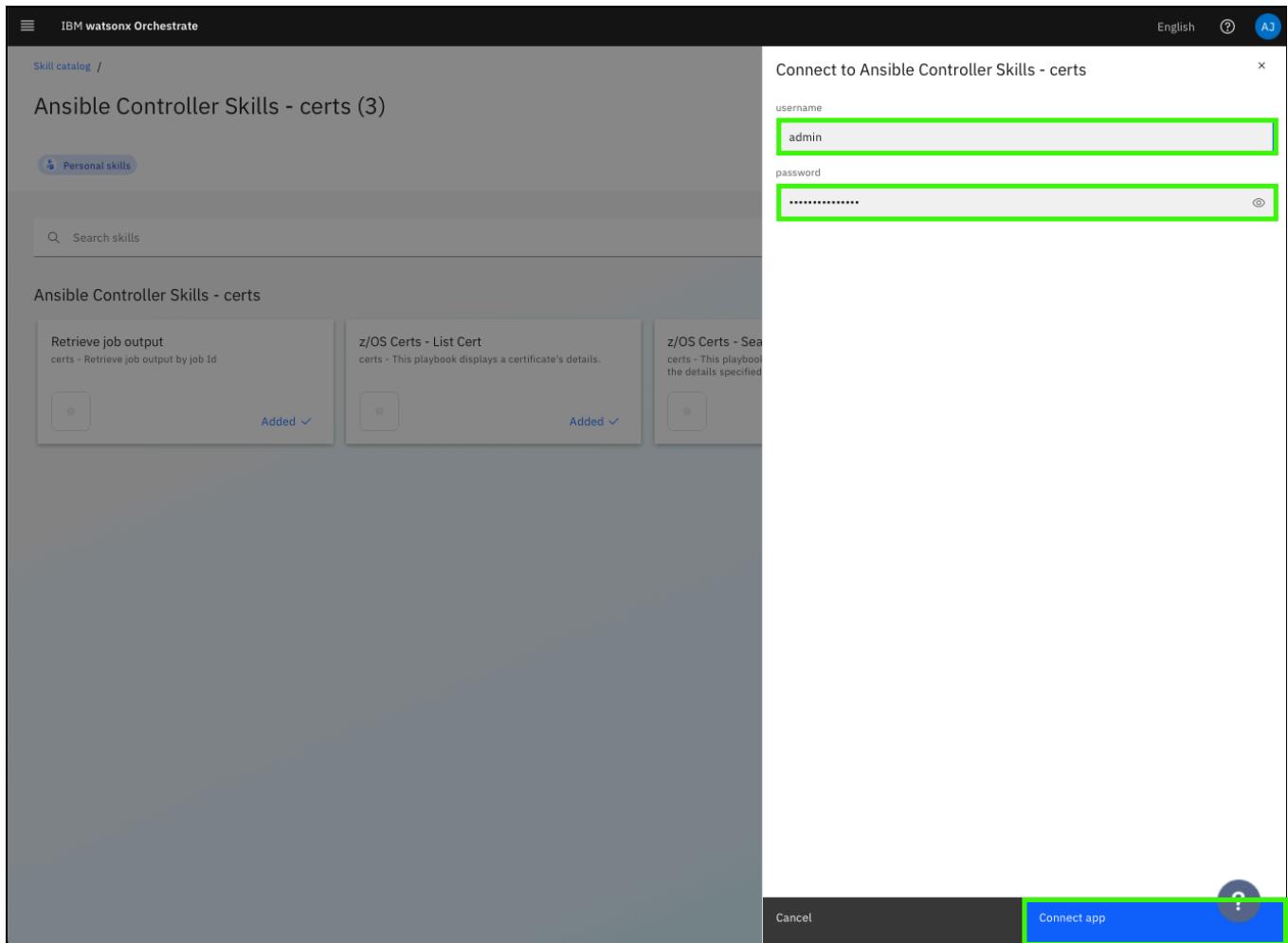
Ansible Controller Skills - certs

- Retrieve job output  
certs - Retrieve job output by job Id
- [Added ✓](#)

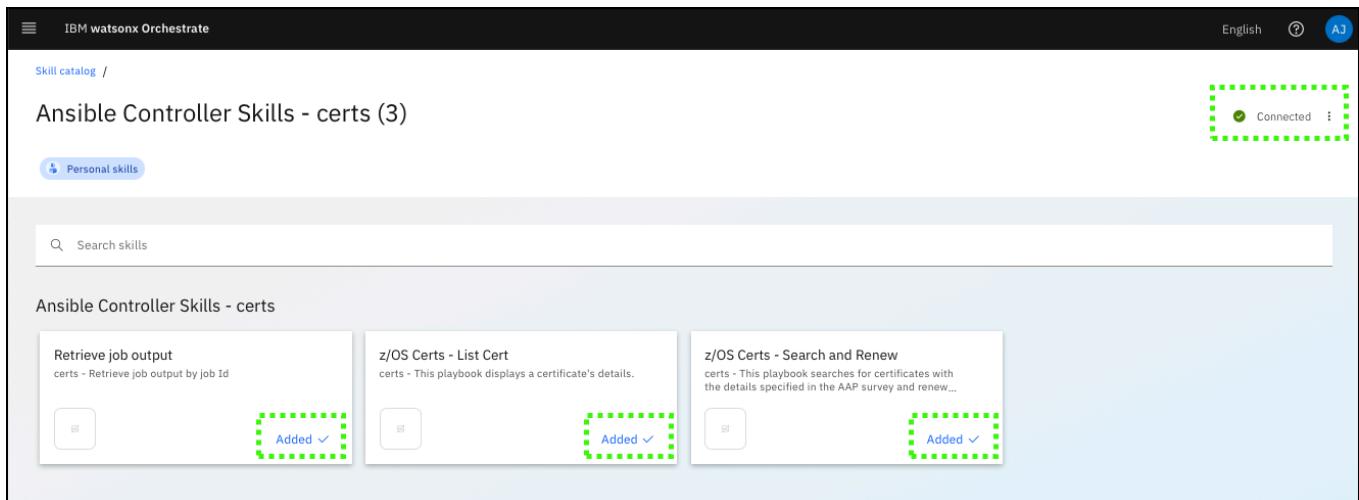
- z/OS Certs - List Cert  
certs - This playbook displays a certificate's details.
- [Added ✓](#)

- z/OS Certs - Search and Renew  
certs - This playbook searches for certificates with the details specified in the AAP survey and renew...
- [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.

Skills - certs (3)

certs

<a href="#">z/OS Certs - List Cert</a> certs - This playbook displays a certificate's details.	<a href="#">Added ✓</a>	<a href="#">z/OS Certs - Search and Renew</a> certs - This playbook searches for certificates with the details specified in the AAP survey and renew...	<a href="#">Added ✓</a>
---	-------------------------	--	-------------------------

## 2. Select the **Draft** version of your assistant and click **Connections**.

Zeeves draft

Skill sets

Zeeves draft	▼
--------------	---

Skills **Connections**

Skills (3)  
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.

Manage skills

Name	App
z/OS IPL Information	
Zeeves-gather-facts-flow	
Retrieve job output	

Items per page: 10 ▼ 1-3 of 3 items 1 ▼ of 1 page

## 3. Enter **certs** in the search bar.

Zeeves draft

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.

certs

Application	Number of skills	Credential type	Connected by ⓘ	Action
Ansible Controller Skills - certs	3	⚠ Not specified	-	⋮

Items per page: 5 ▼ 1-1 of 1 items 1 ▼ of 1 page

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	-	<span>⋮</span> <span>Connect app</span>

5. Click Connect app.

Connect to Ansible Controller Skills - certs

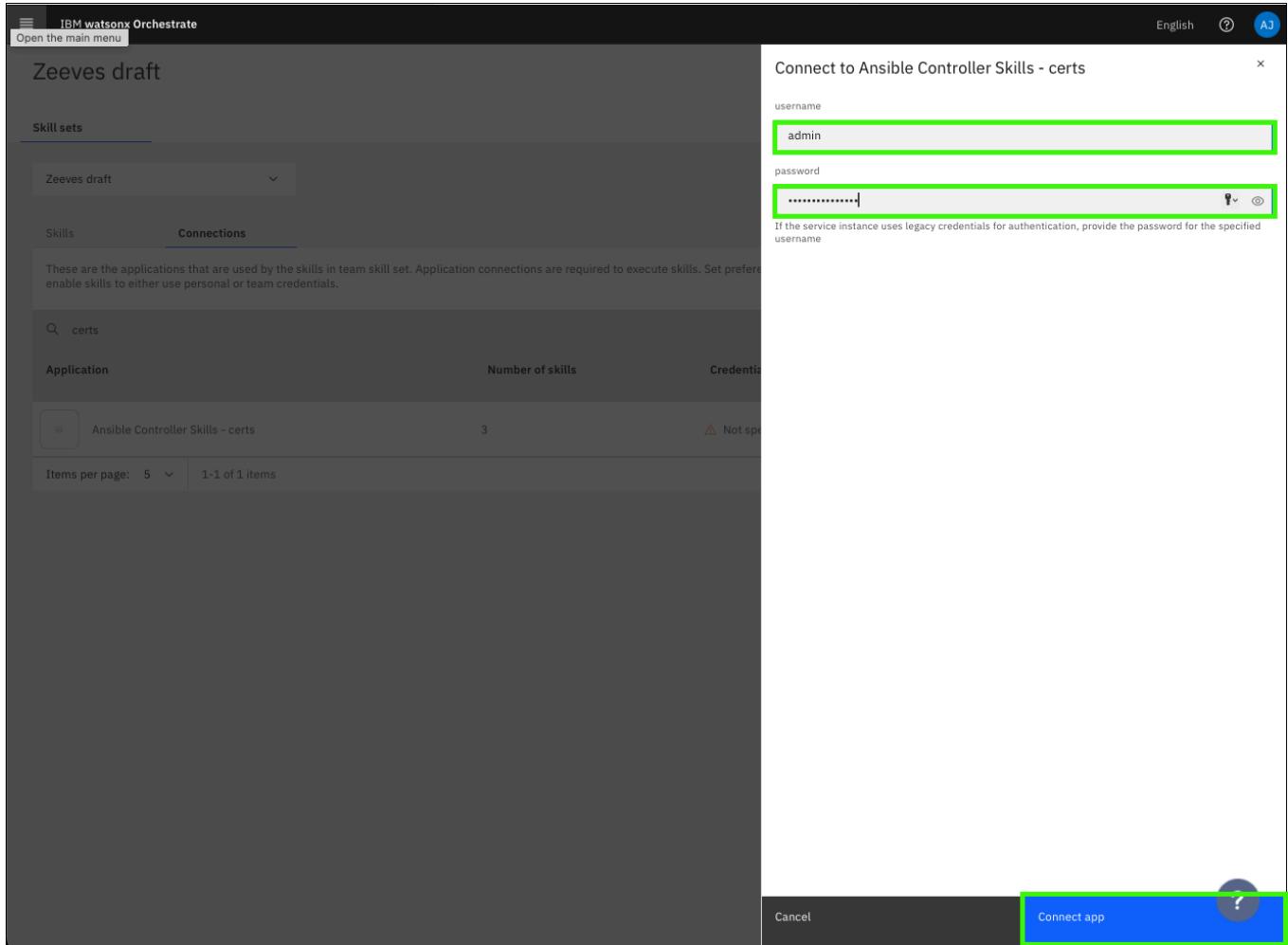
Member credentials  
Each team member uses their own credentials to connect to this app and use its skills.

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

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.

The screenshot shows the IBM Watsonx Orchestrate dashboard. The left sidebar has sections for Chat, Skill sets (which is selected and highlighted with a green border), Skill catalog, BUILD (AI assistant builder, Skill studio), and ADMINISTER (Access management). The main content area displays a table with columns: Number of skills, Credential type, Connected by (with a help icon), and Action. A tooltip for Skill sets explains that it's used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to use global or team credentials.

## 2. Click **Create** and then click **Skill flow**.

The screenshot shows the Skill studio interface. The top navigation bar includes IBM Watsonx Orchestrate, Skill studio, English, and a help icon. The main area is titled "Welcome to Skill studio" with a sub-instruction: "Optimize productivity by using projects to build and manage skill flows and connectors." Below this are tabs for Skills and Apps, with Skills selected. A search bar contains "Find a skill". On the right, a "Create" dropdown menu is open, showing options: Create (selected and highlighted with a green border), Skill Flow (also highlighted with a green border), Import API, and Configure prebuilt skills.

## 3. Click the **+** icon.

The screenshot shows the "Create a Flow" interface. The top navigation bar includes Skills and apps / Create a Flow, English, and a help icon. The main area is titled "Untitled" and shows a basic workflow diagram. It starts with a green circle labeled "Start", followed by a square node with a plus sign (+) which is highlighted with a green border, and ends with a red circle labeled "End". There are also small icons for opening and saving the file.

## 4. Click the **certs** app.



Search on certs if you do not see the tile for your app.

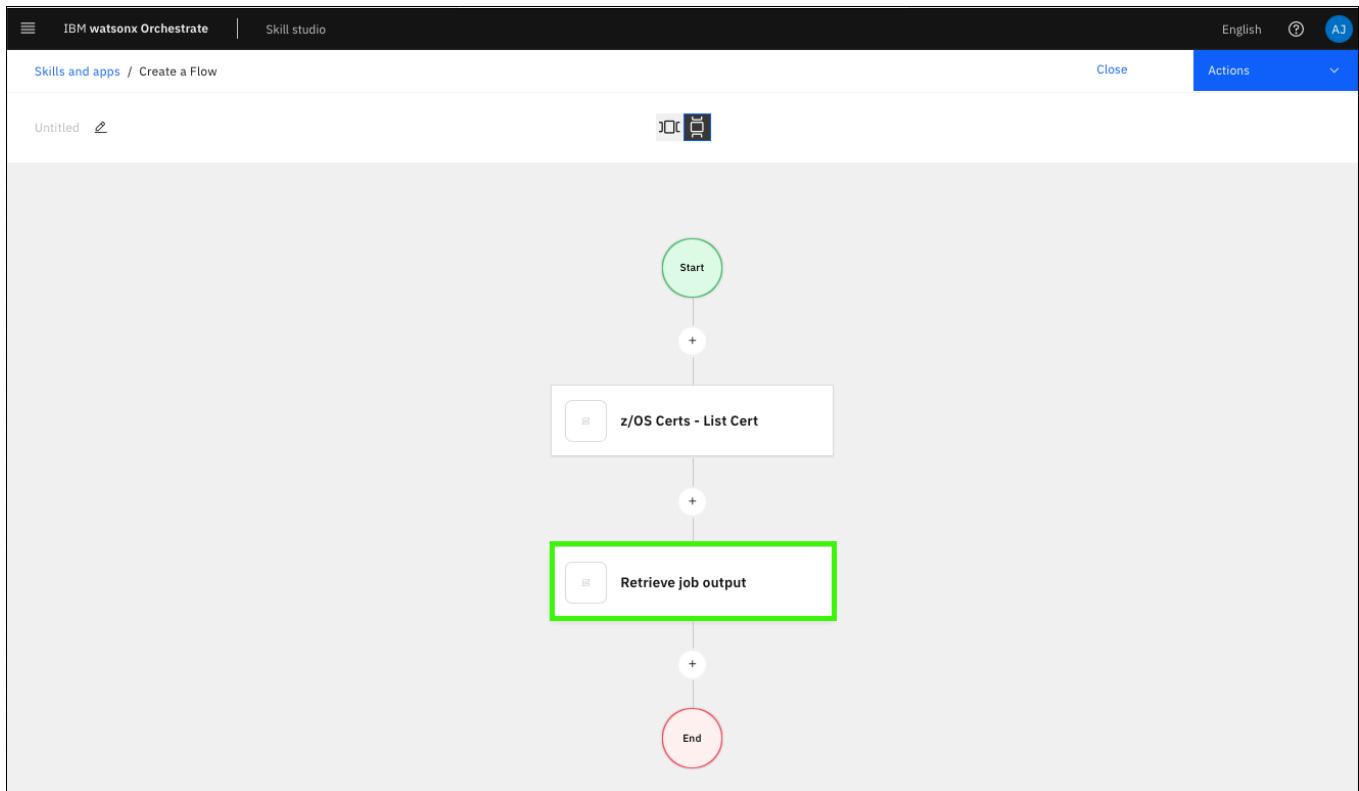
The screenshot shows the 'Skill studio' interface in IBM Watsonx Orchestrator. On the left, there's a flowchart with a green 'Start' node and a red 'End' node. A modal window titled 'Add Skills' is open on the right, displaying a grid of skill categories. One category, 'Ansible Controller Skills - certs', is highlighted with a green border. Below the grid, a message says 'Total apps: 84. Viewing section 1 of 2'.

5. Click Add Skill + in the z/OS Certs - List Cert tile.

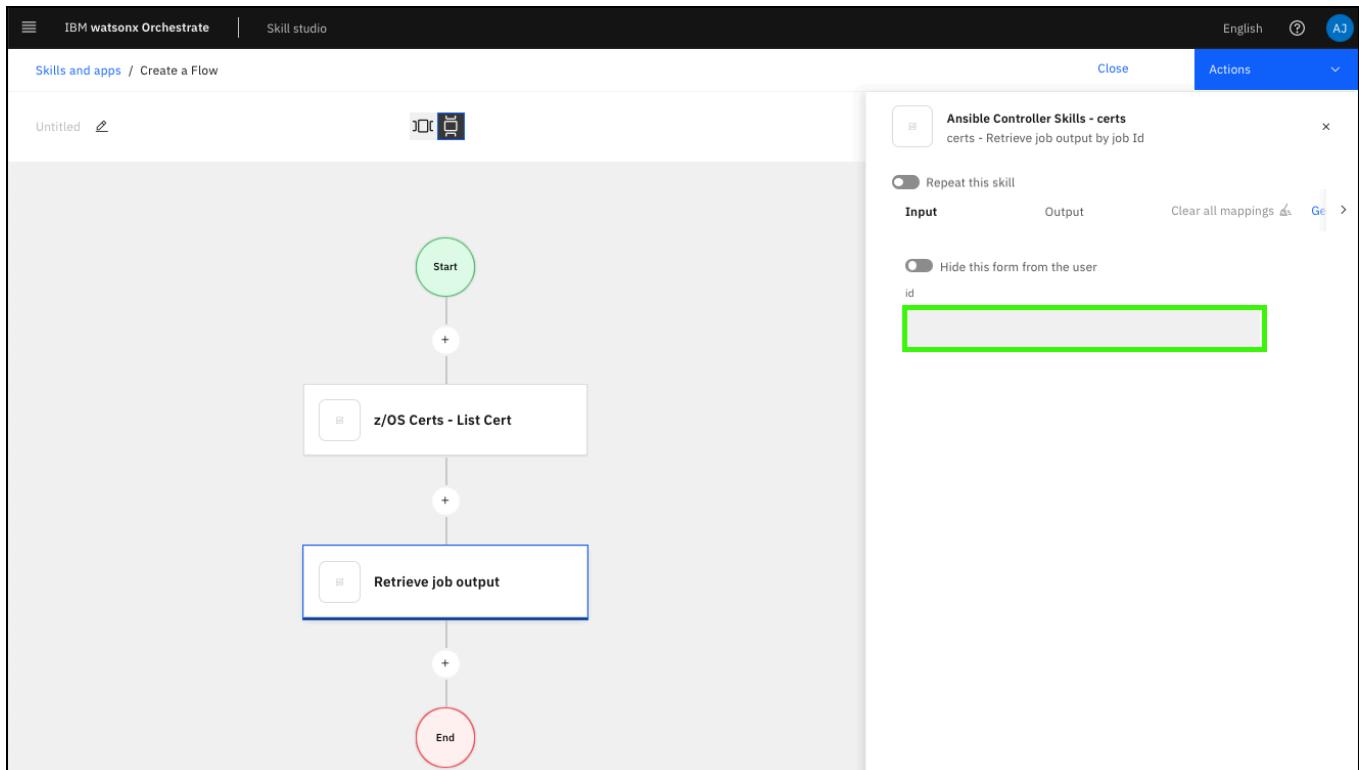
The screenshot shows the 'Ansible Controller Skills - certs' tile expanded. It contains two skills: 'Retrieve job output' and 'z/OS Certs - List Cert'. The 'z/OS Certs - List Cert' skill has an 'Add Skill +' button highlighted with a green box.

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.

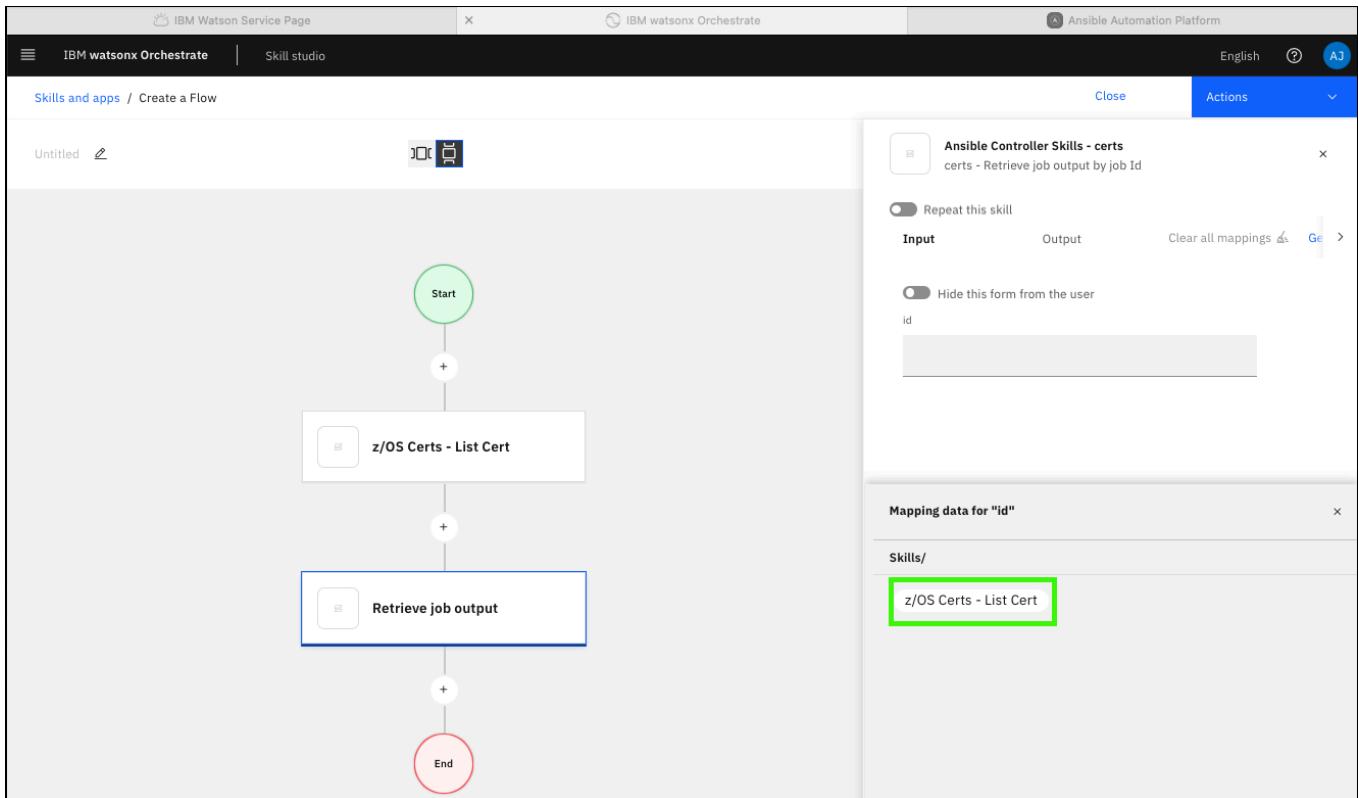
7. Click the **Retrieve job output** skill.



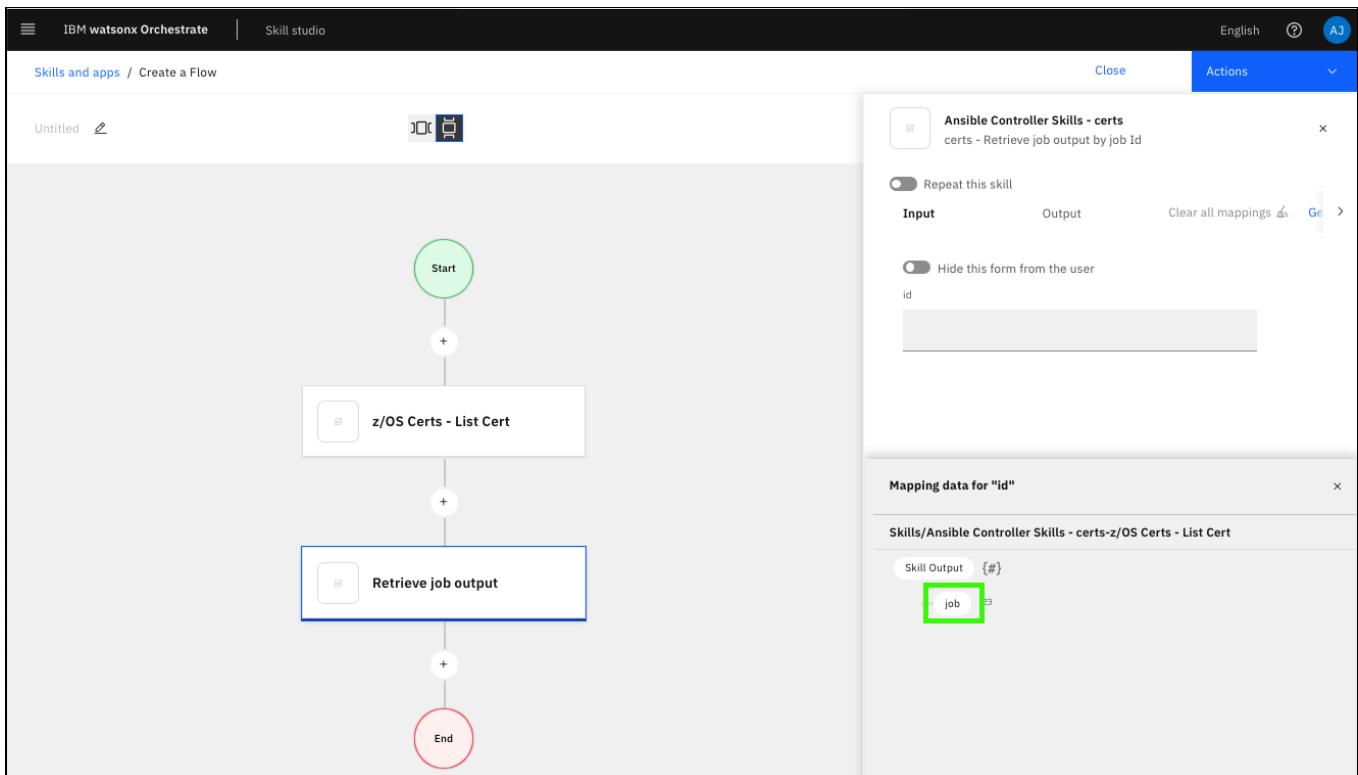
8. Click the **id** field.



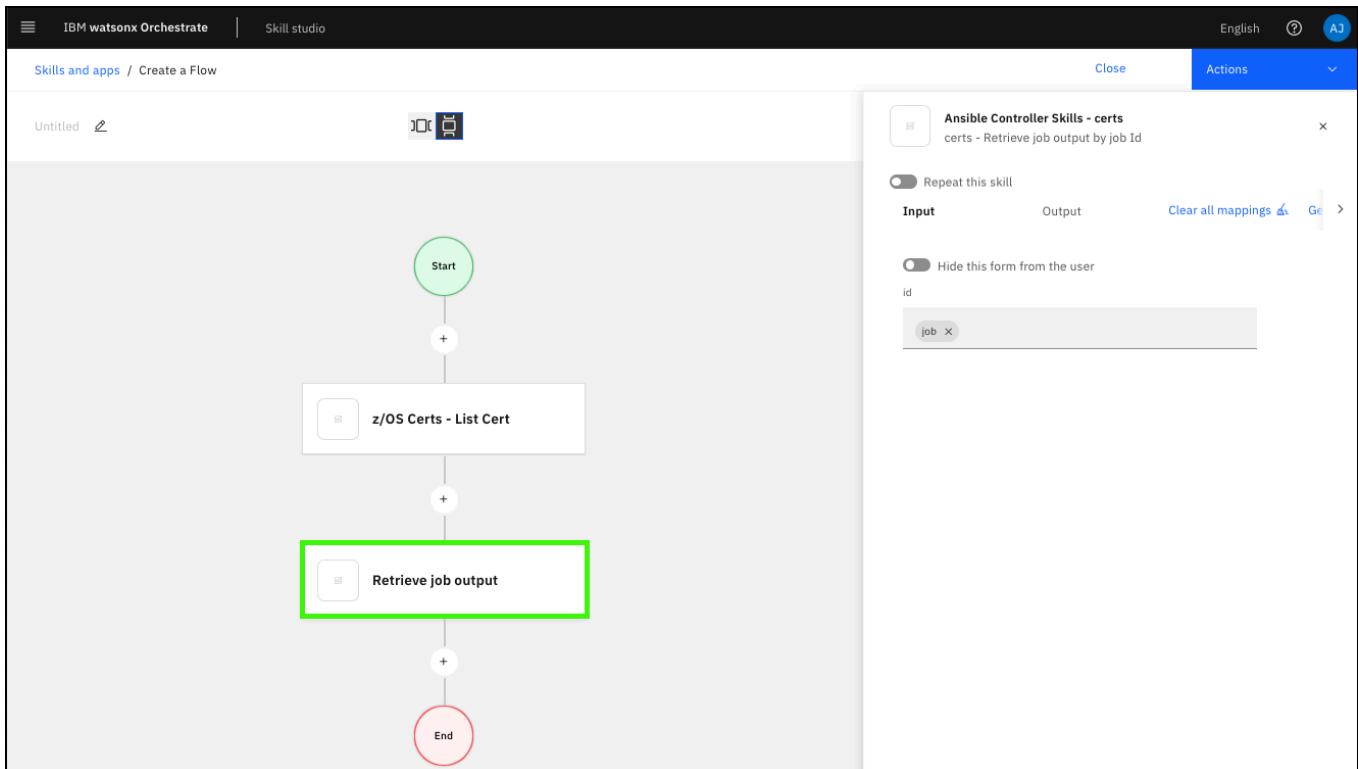
9. Click **z/OS certs - List Cert**.



## 10. Click job.

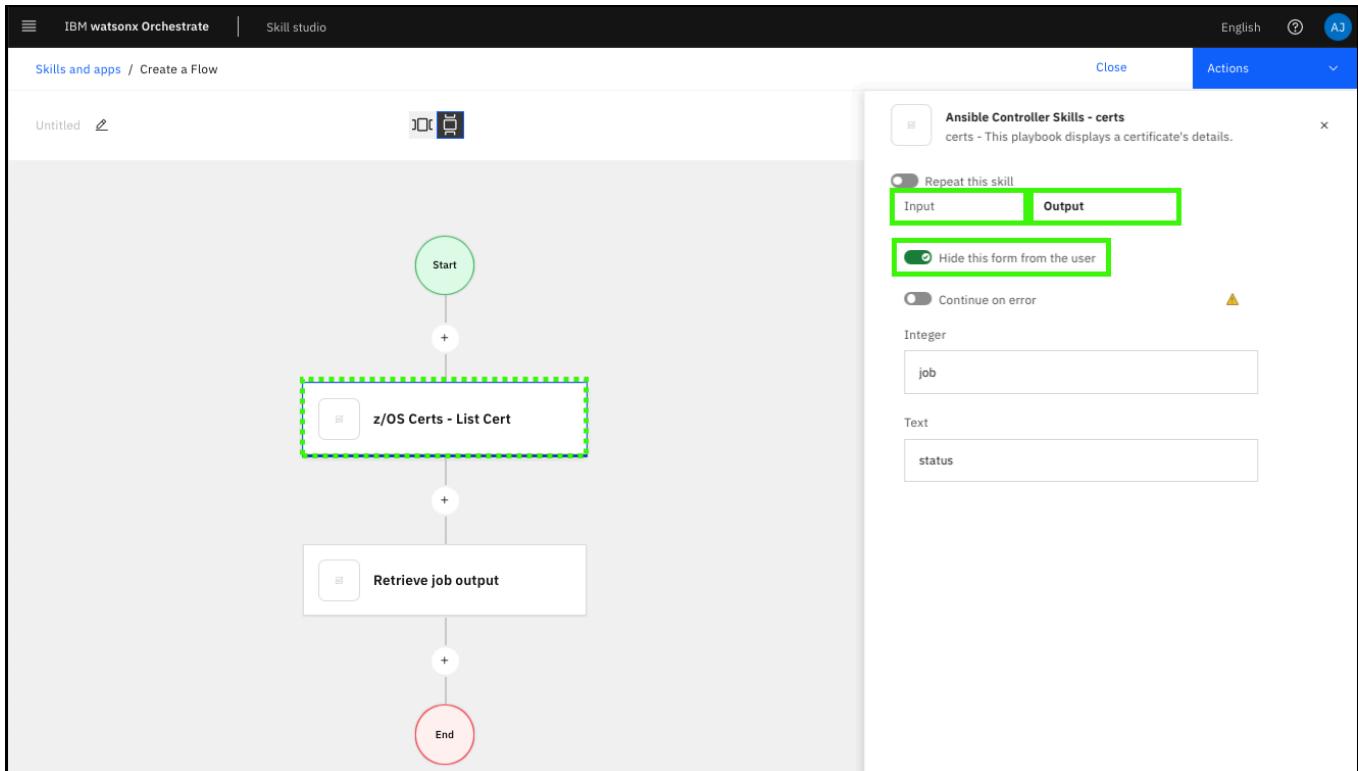


## 11. Click the z/OS Certs - List Cert tile.

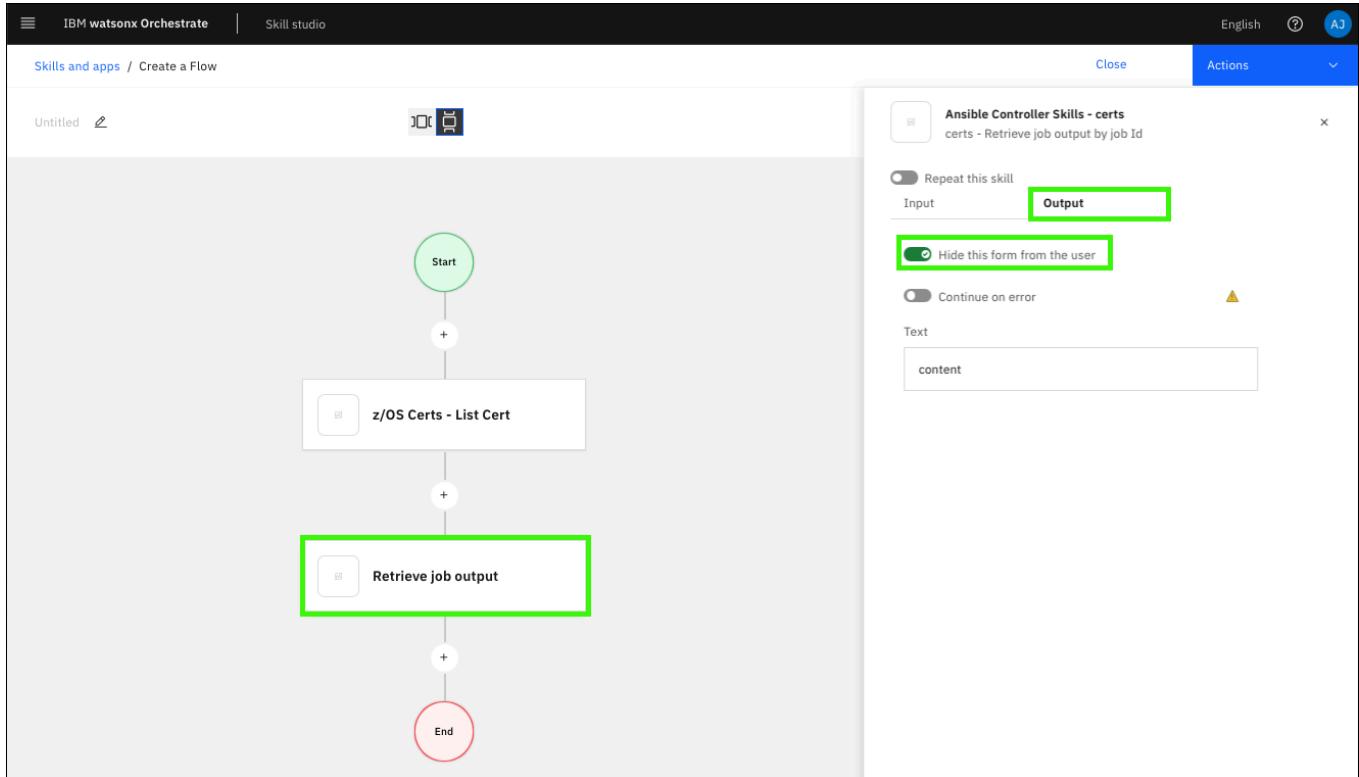


12. On both the **Input** and **Output** tabs for the **z/OS Certs - List Cert** skill, enable the **Hide this form from the user** options.

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.

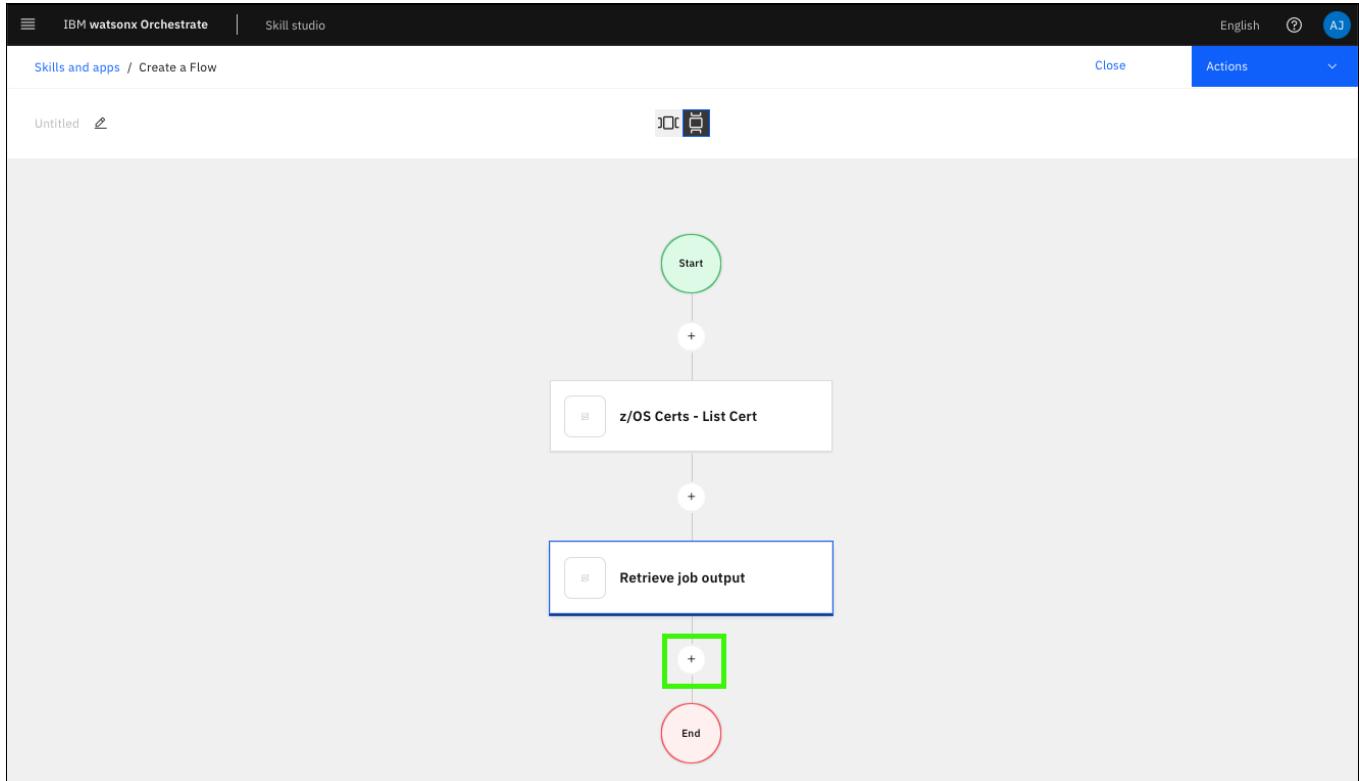


13. Repeat step 12 for the **Output** of the **Retrieve job output** skill.



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.



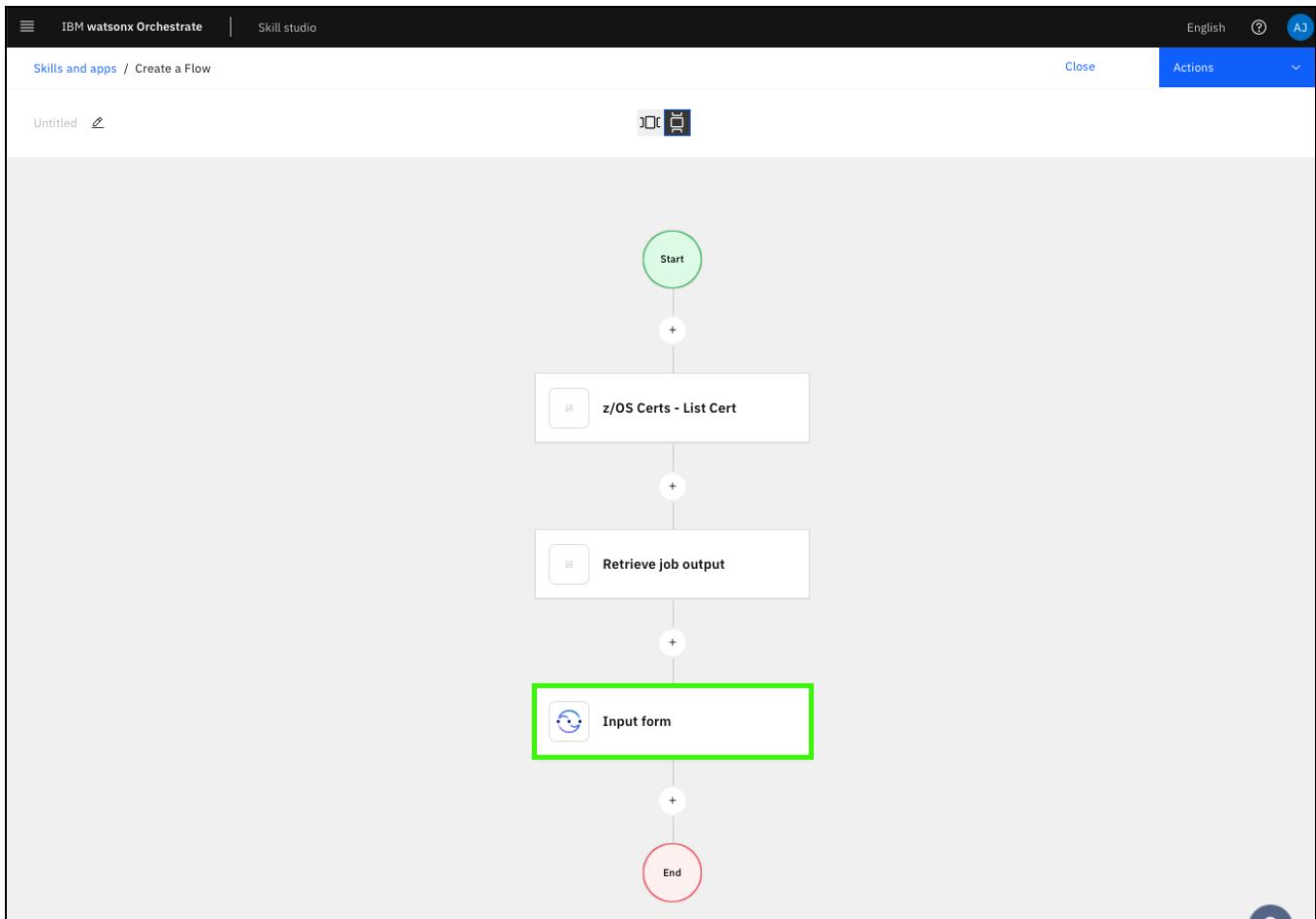
15. Click the **Custom forms**.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, a flow diagram titled 'Untitled' is displayed. It starts with a 'Start' node, followed by a 'z/OS Certs - List Cert' action, then a 'Retrieve job output' step (which is currently selected and highlighted with a blue border), and finally an 'End' node. On the right, a modal window titled 'Add Skills' is open, showing a grid of various skills. The 'Custom forms' skill, which has a smiley face icon and is described as having 2 skills, is highlighted with a green border.

16. Click Add skill + in the Input form.

The screenshot shows the same flow diagram as the previous one. The 'Add Skills' modal is still open, but now the 'Custom Forms' section is visible. Within this section, the 'Input form' skill (described as 'Display a form to input data') is highlighted with a green dashed border. To its right, there is another skill card for 'Output form' (described as 'Display a form to output data'). Below each skill card is a 'Add Skill +' button, which is also highlighted with a green border.

17. Click the **Input form** skill.



18. Click Add input field +.

**Custom forms**

Display a form to input data

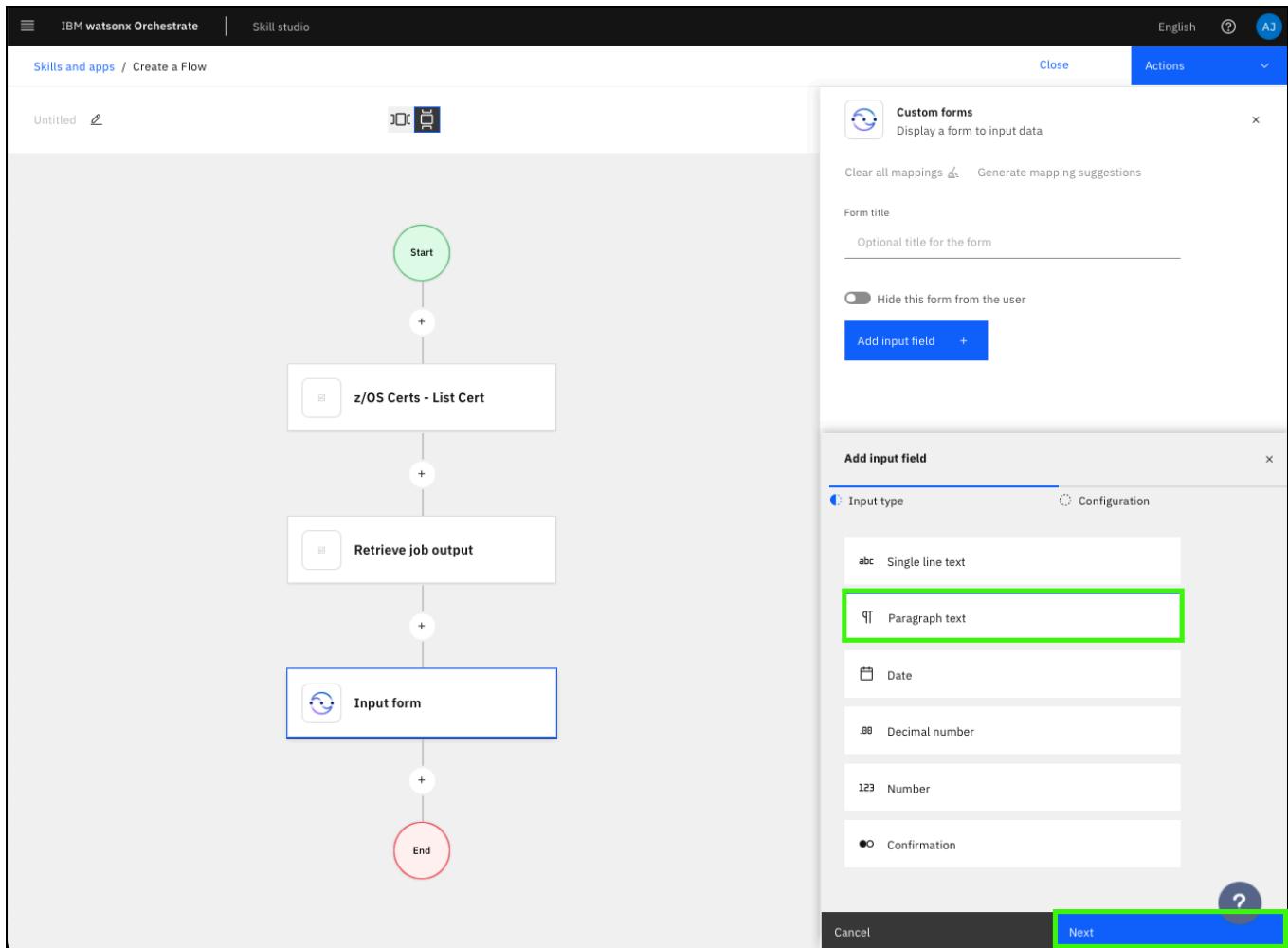
[Clear all mappings](#) [Generate mapping suggestions](#)

Form title

Hide this form from the user

**Add input field +**

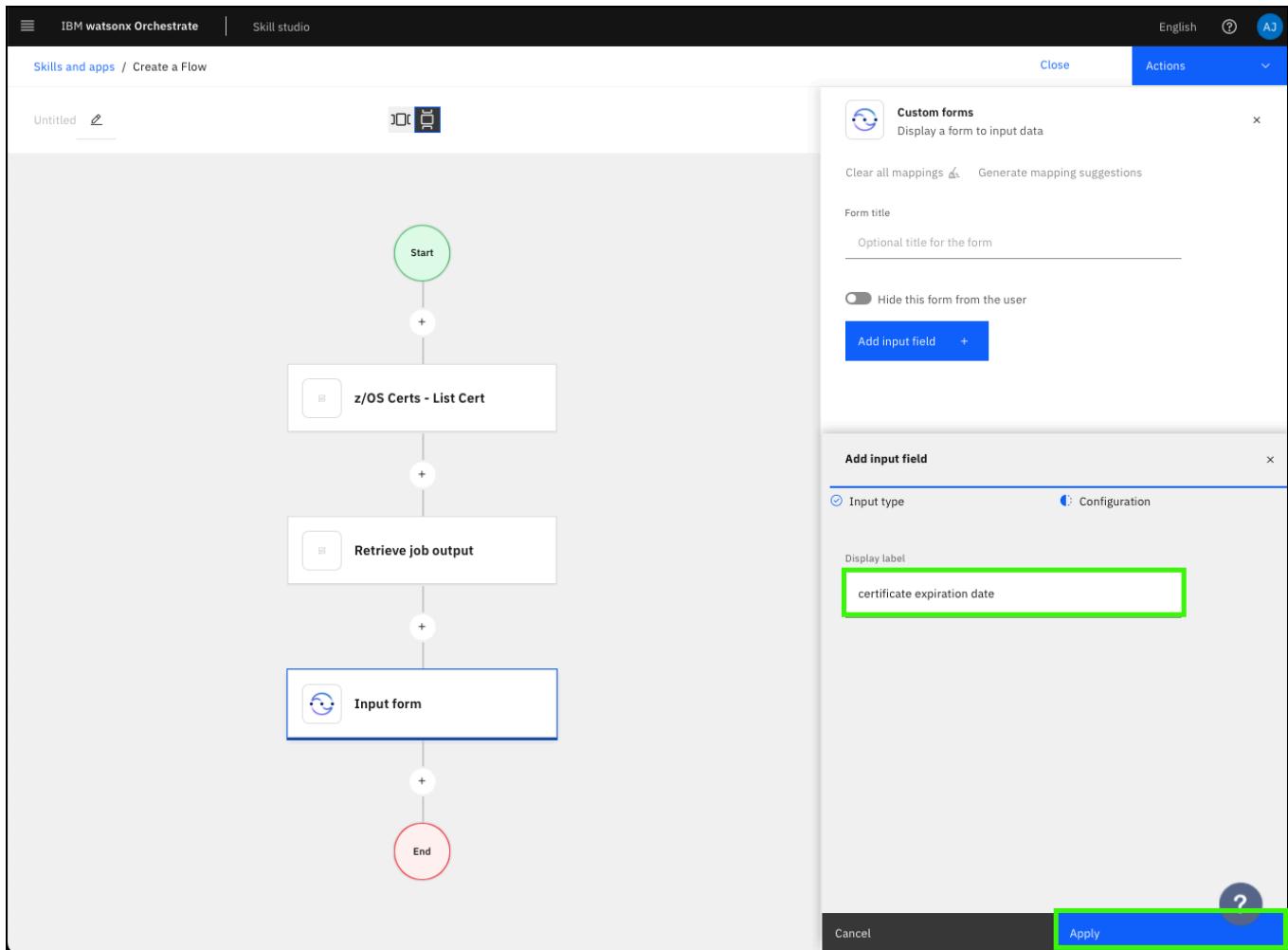
19. Click Paragraph text and then click Next.



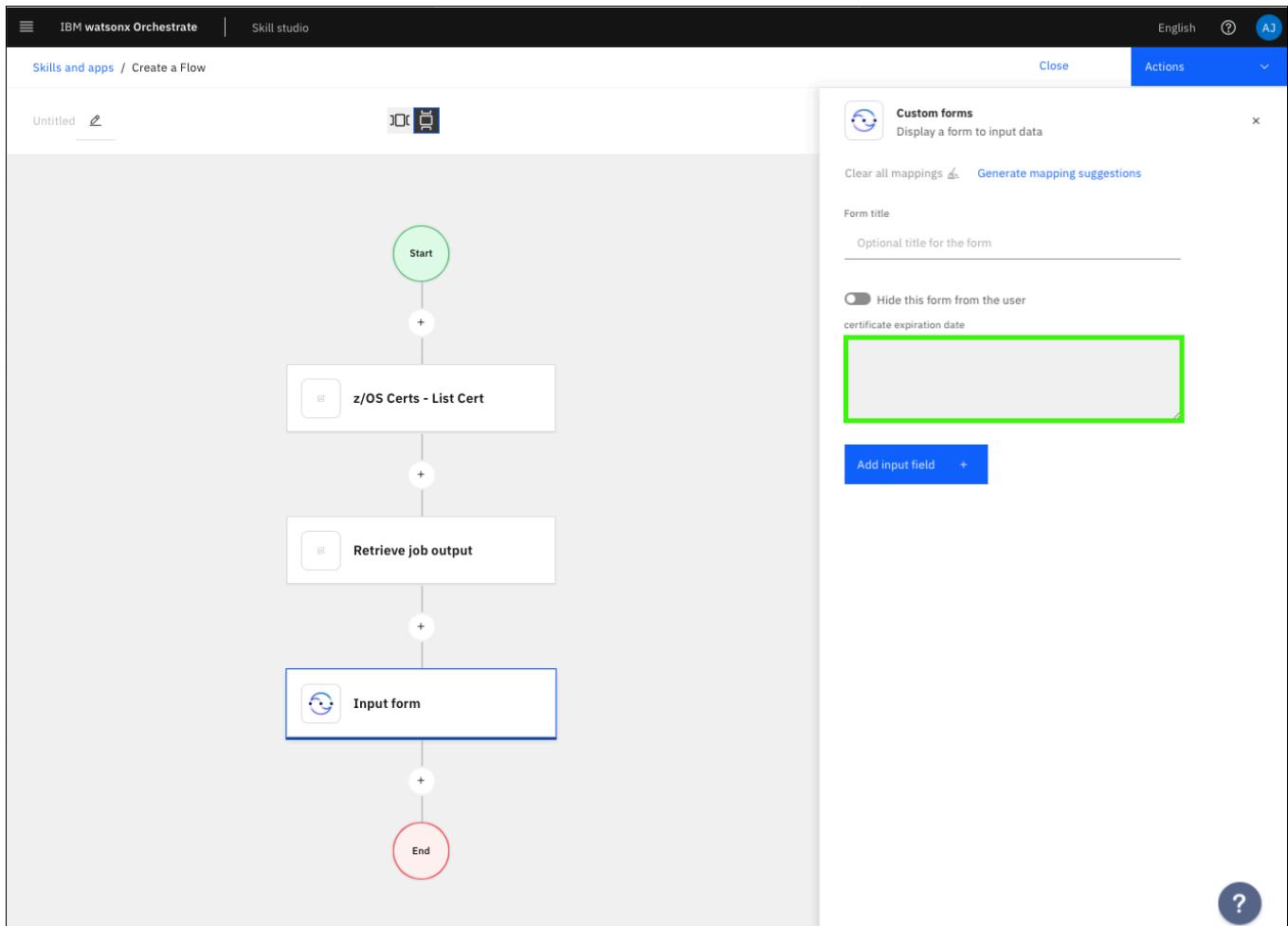
20. Enter certificate expiration date in the **Display label** field and click **Apply**.

**Display label:**

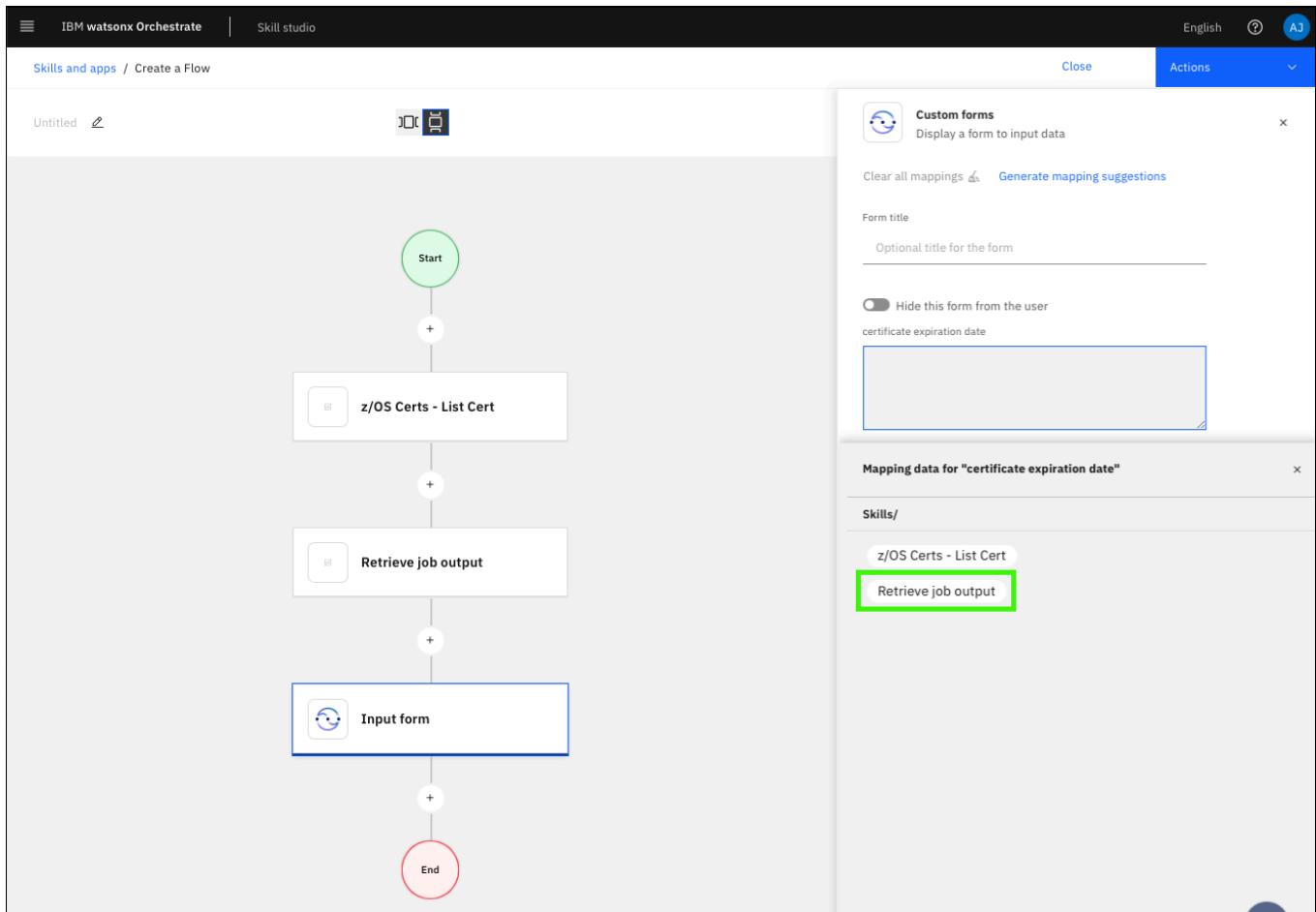
certificate expiration date



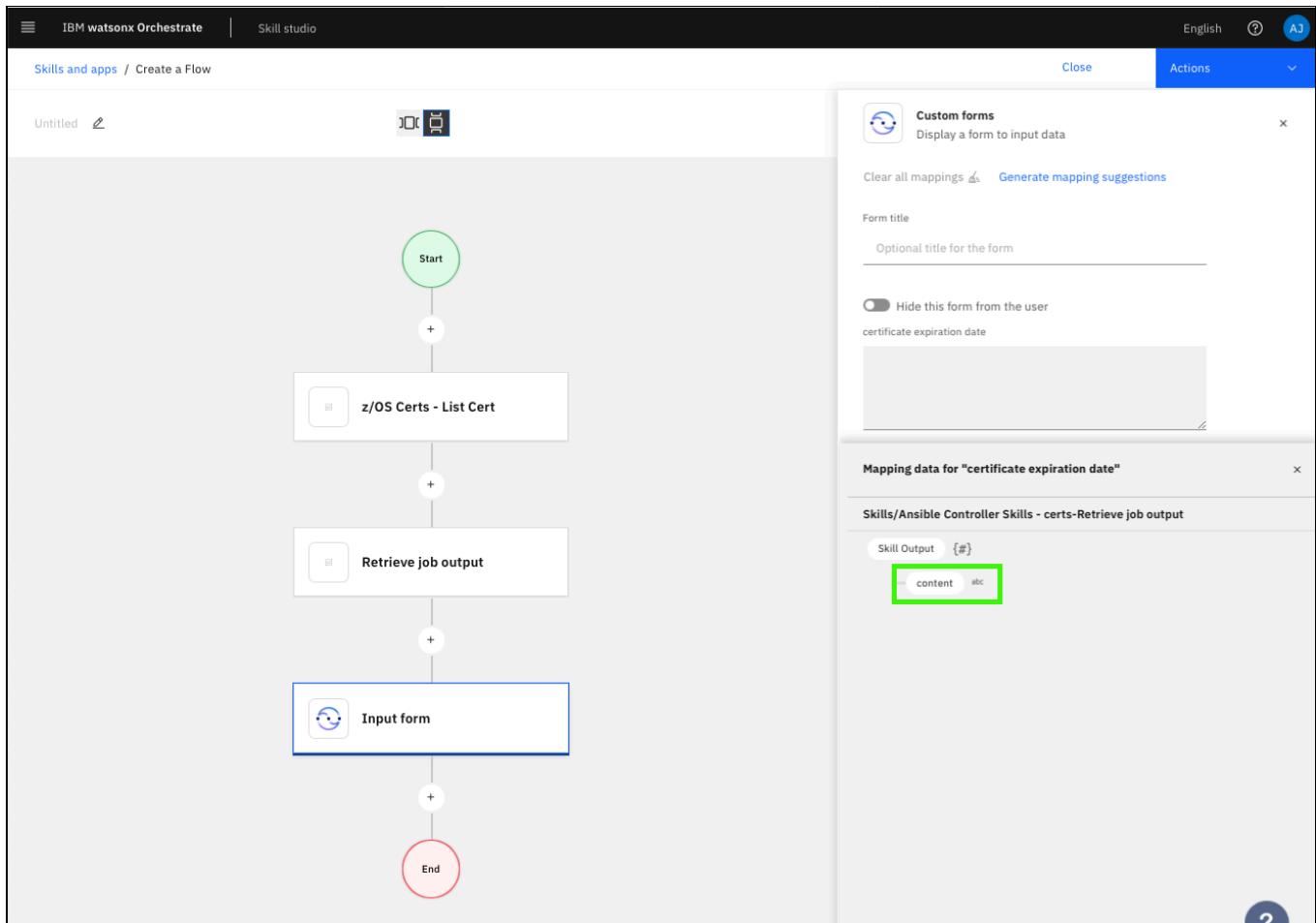
21. Click the **certificate expiration date** entry field.



22. In the **Mapping data for "certificate expiration data"** section, click **Retrieve job output**.

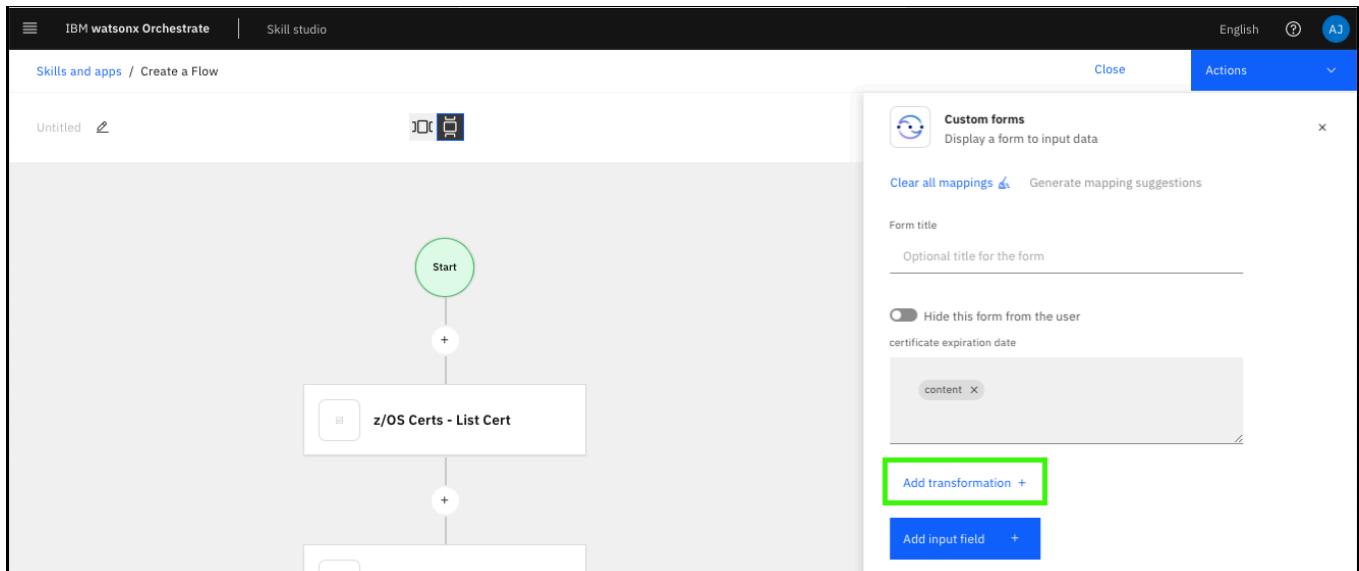


23. Click content.

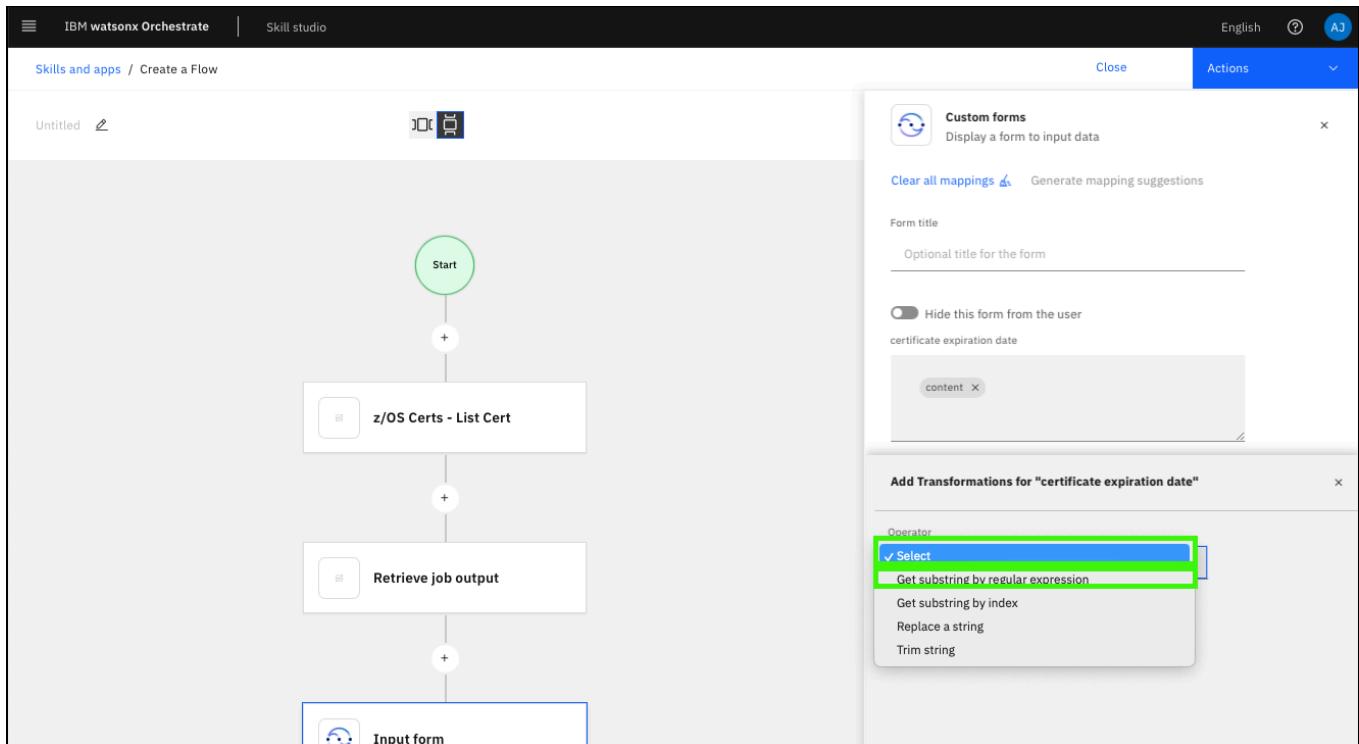


#### 24. Click Add transformation +.

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**.

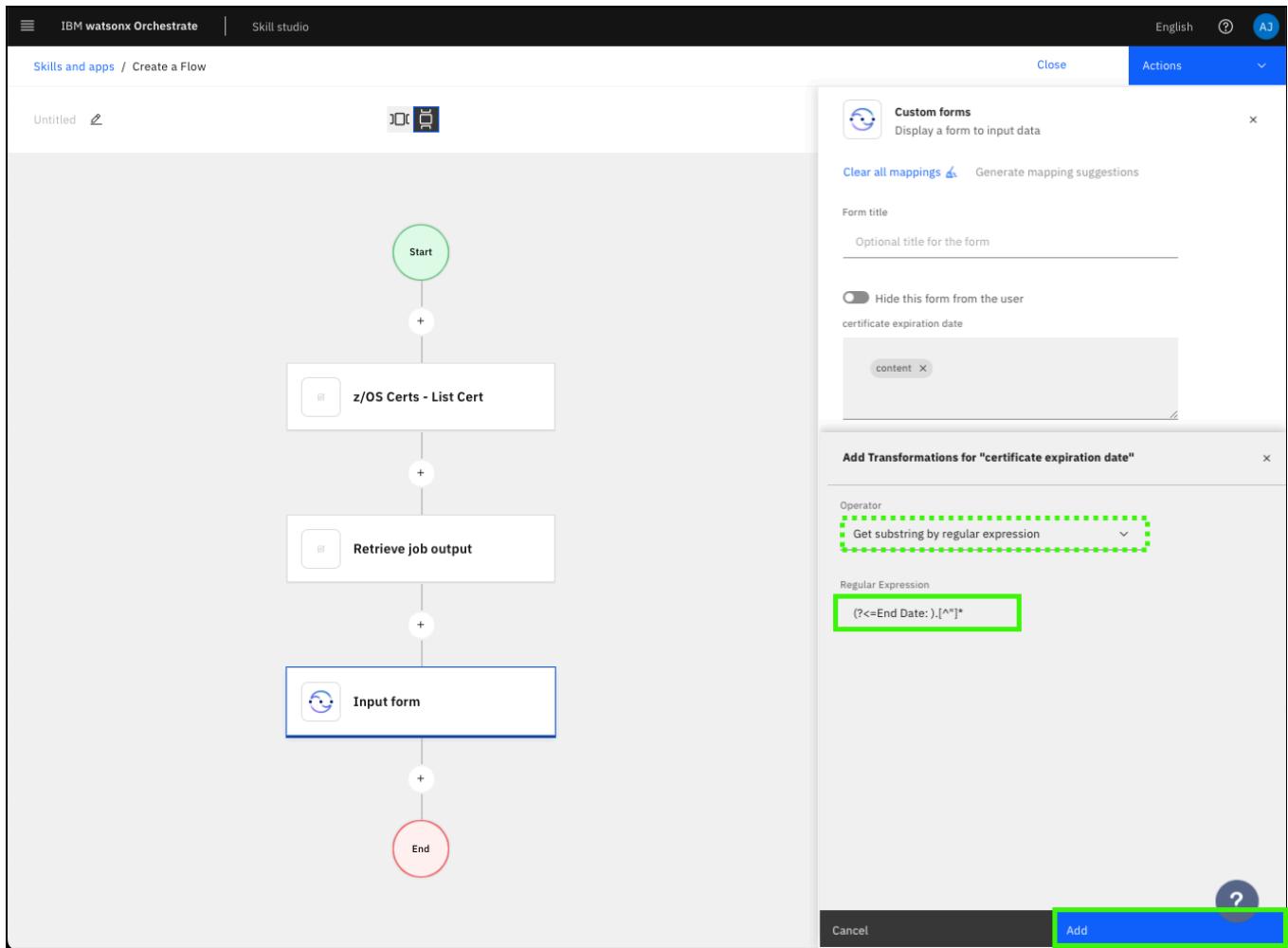
#### Regular Expression:

```
(?=<=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.

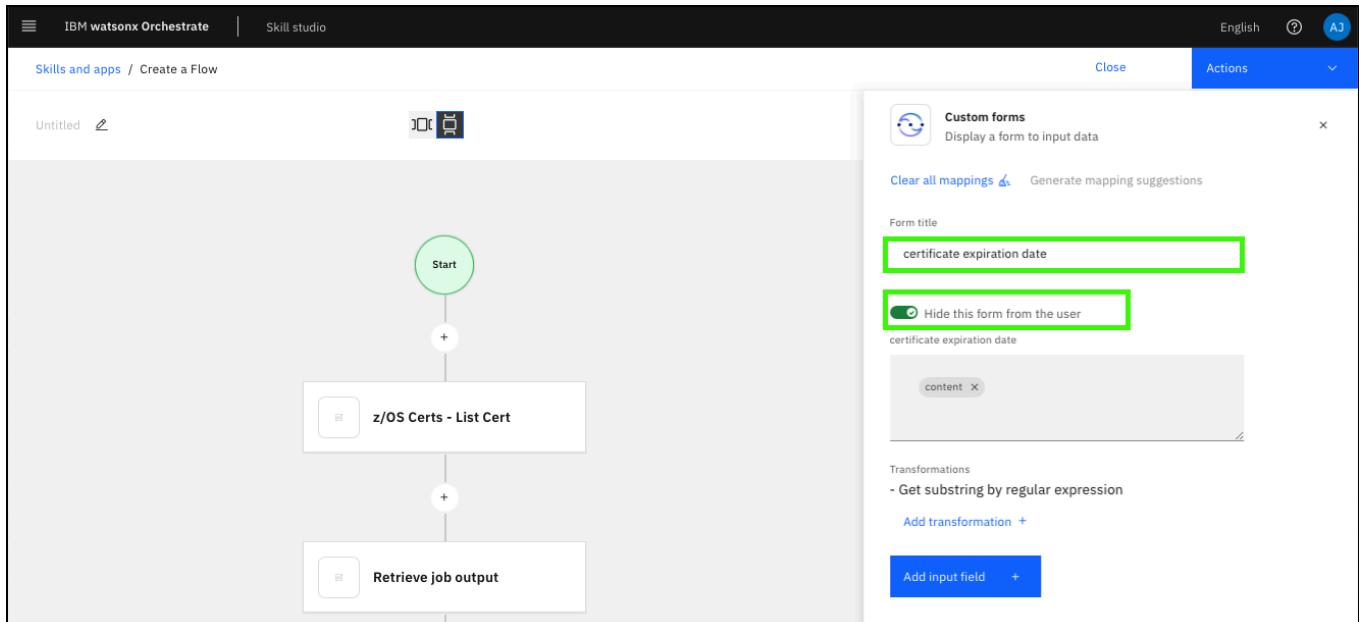
For more information on transforming data within Watsonx Orchestrate, review the documentation found [here](#).



27. Enter `certificate expiration date` in the form title and toggle the **Hide this form from the user** option.

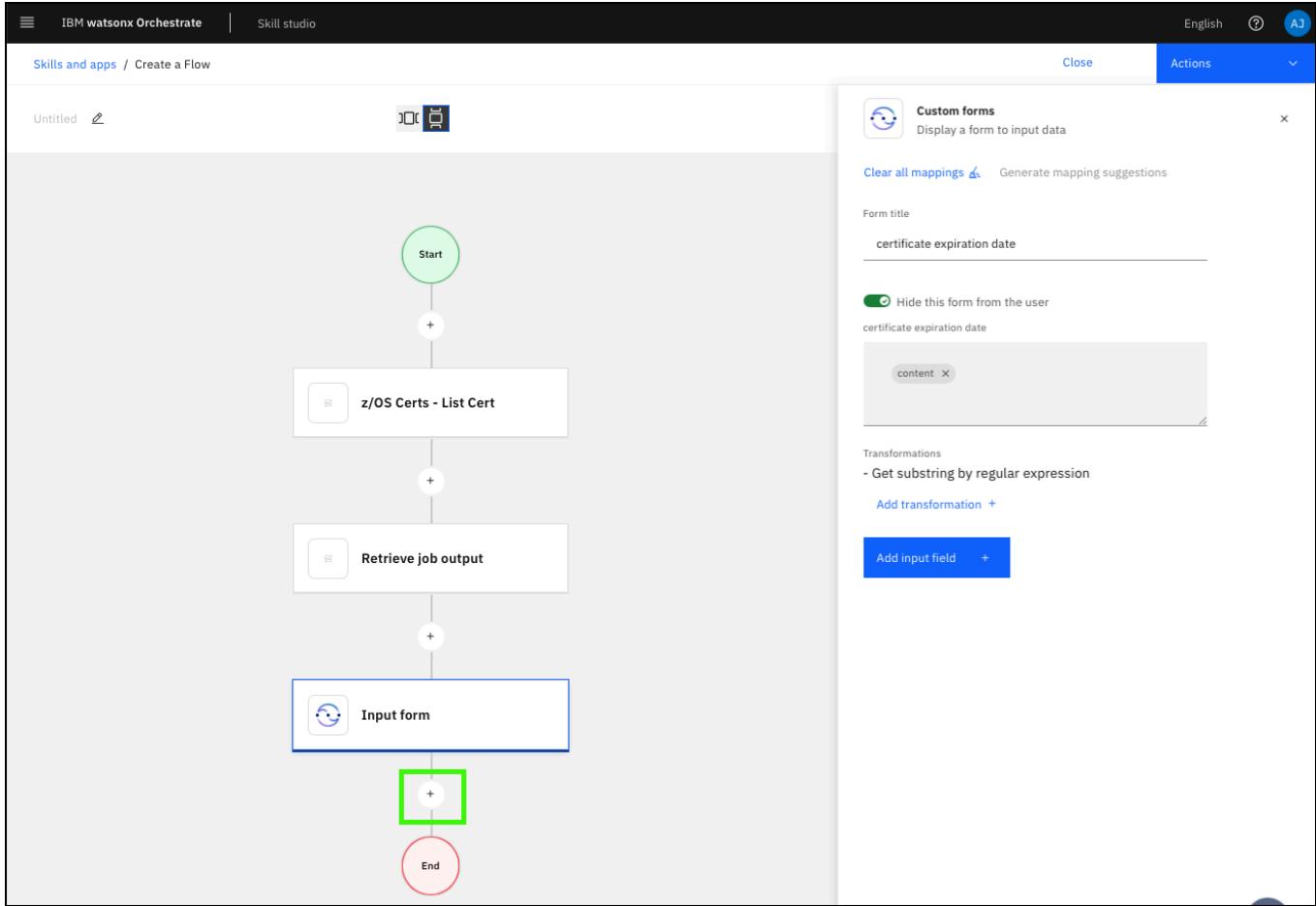
#### Form title:

certificate expiration date

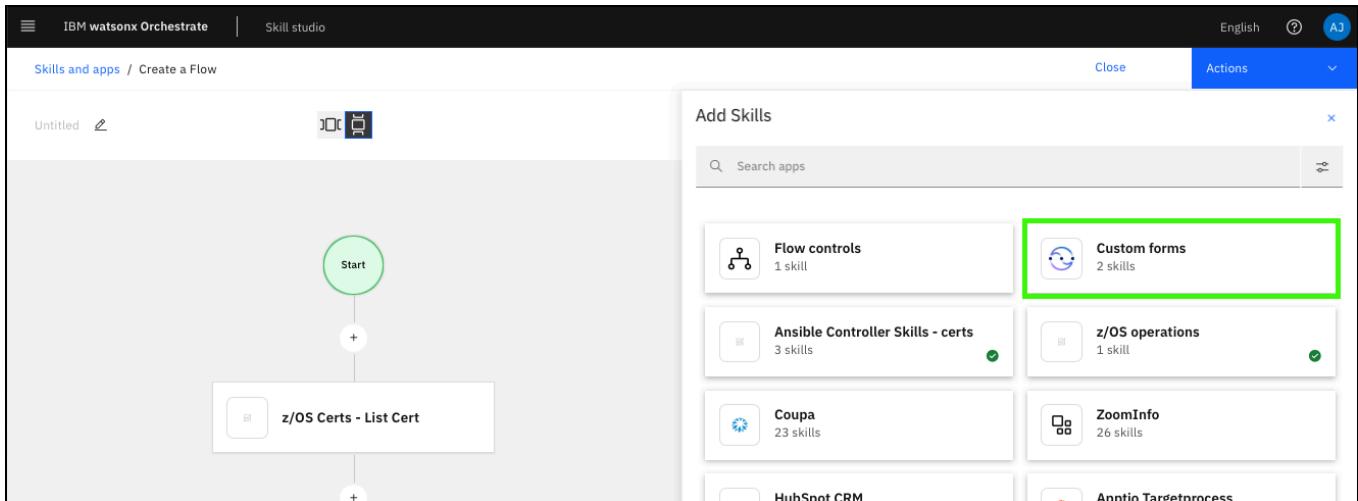


Next, create an output for to return the transformed data from the input skill.

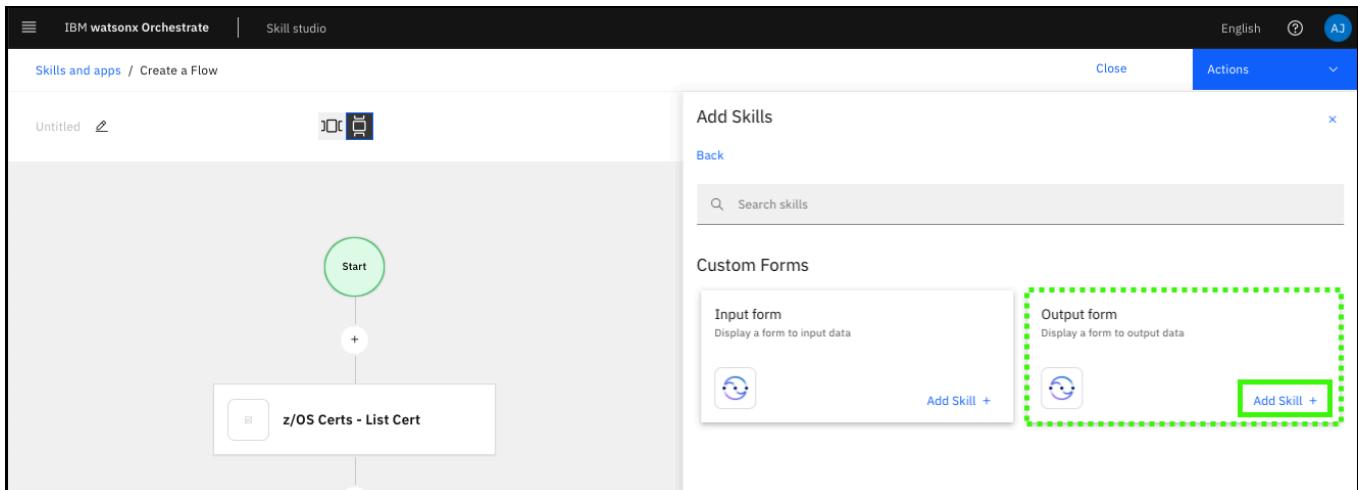
28. Click the **+** icon below the Input form.



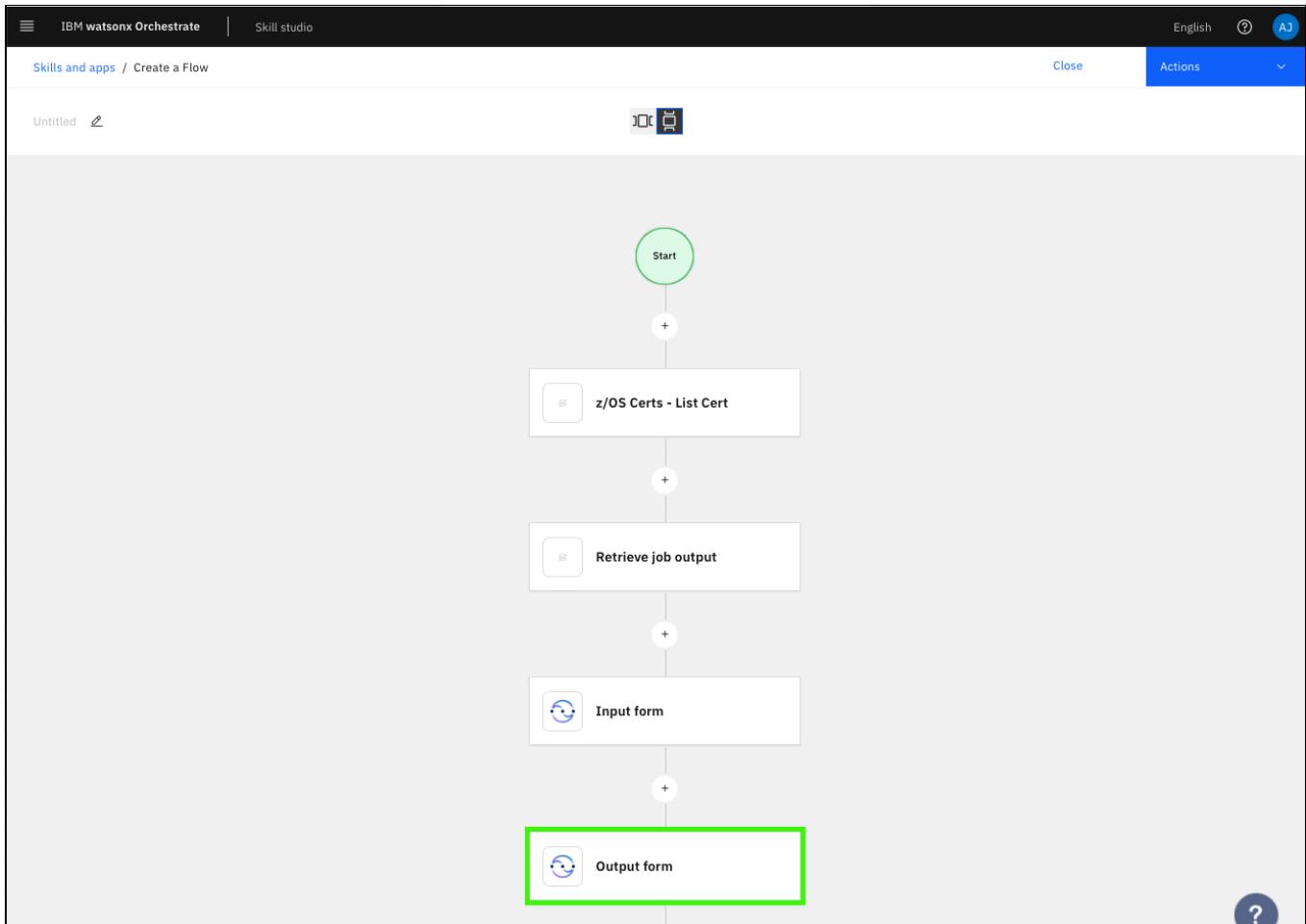
29. Click **Custom forms**.



30. Click **Add Skill +** in the **Output form** tile.

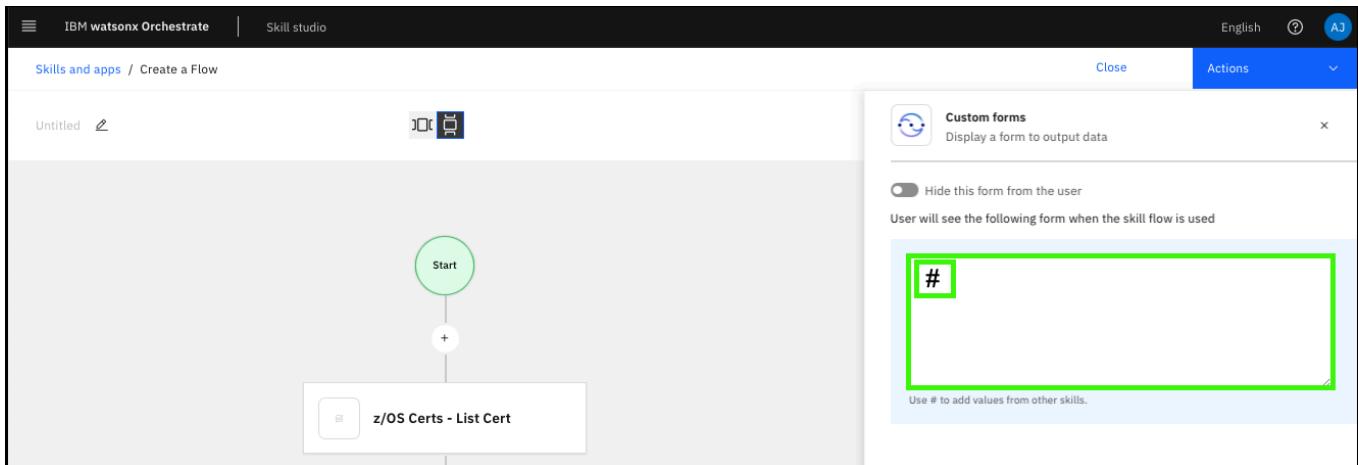


31. Click the **Output form** skill.

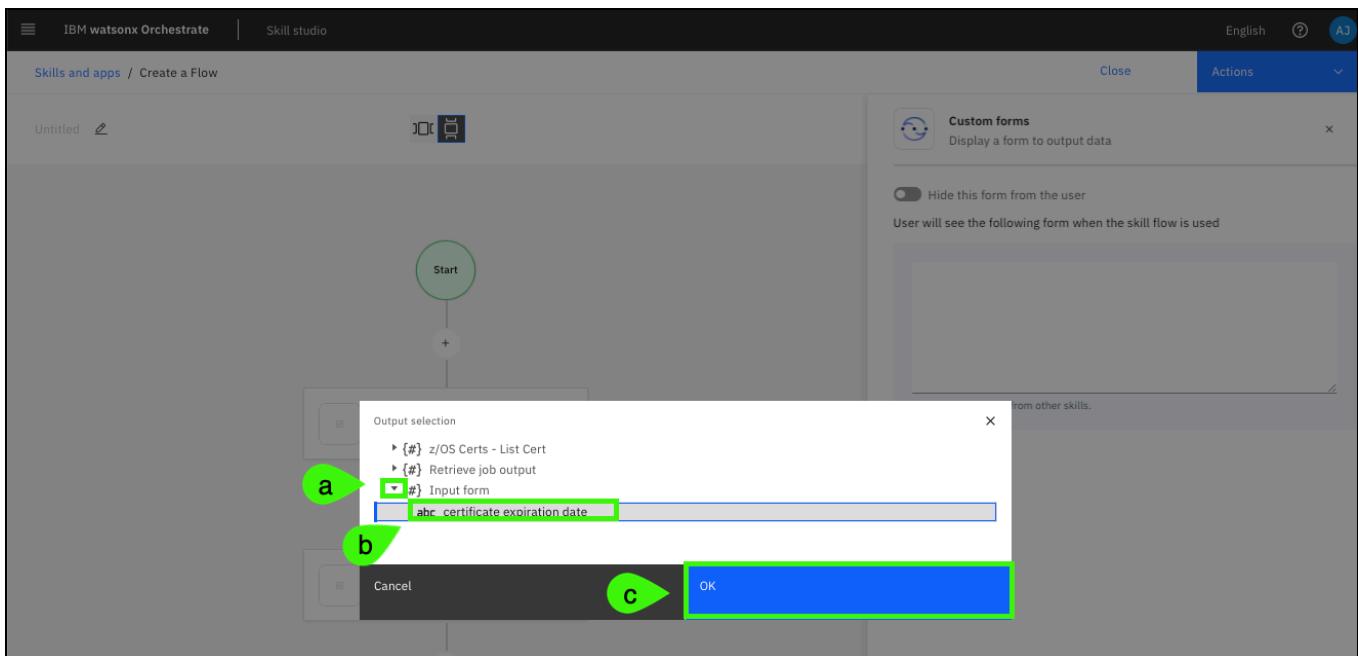


32. Click in the **Custom forms** field and enter `#` (the pound key, also known as the number sign or hash key).

Typing the `#` opens a new dialog window.



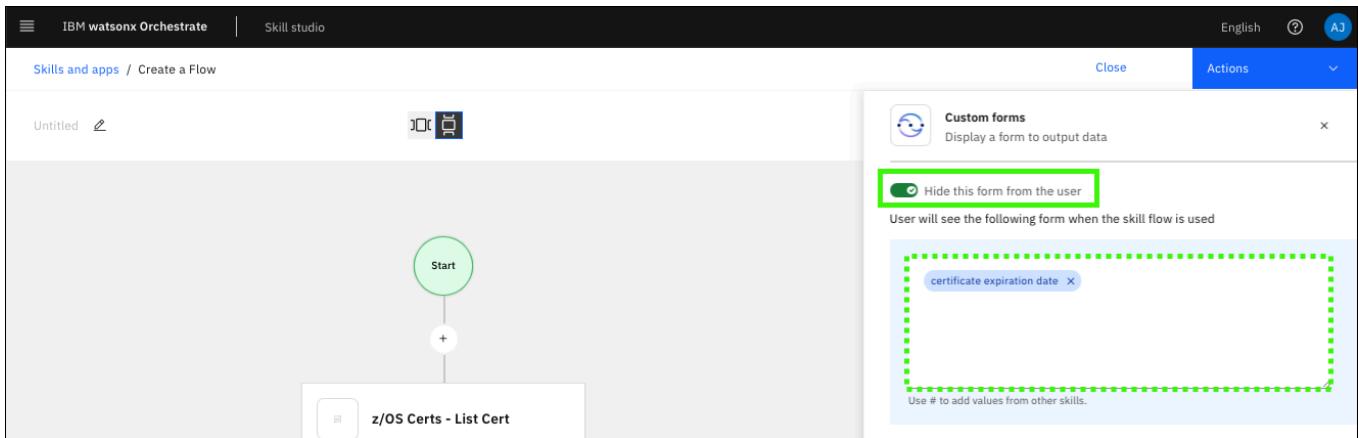
33. Expand (a) **Input form**, select (b) **certificate expiration date**, and then click (c) **OK**.



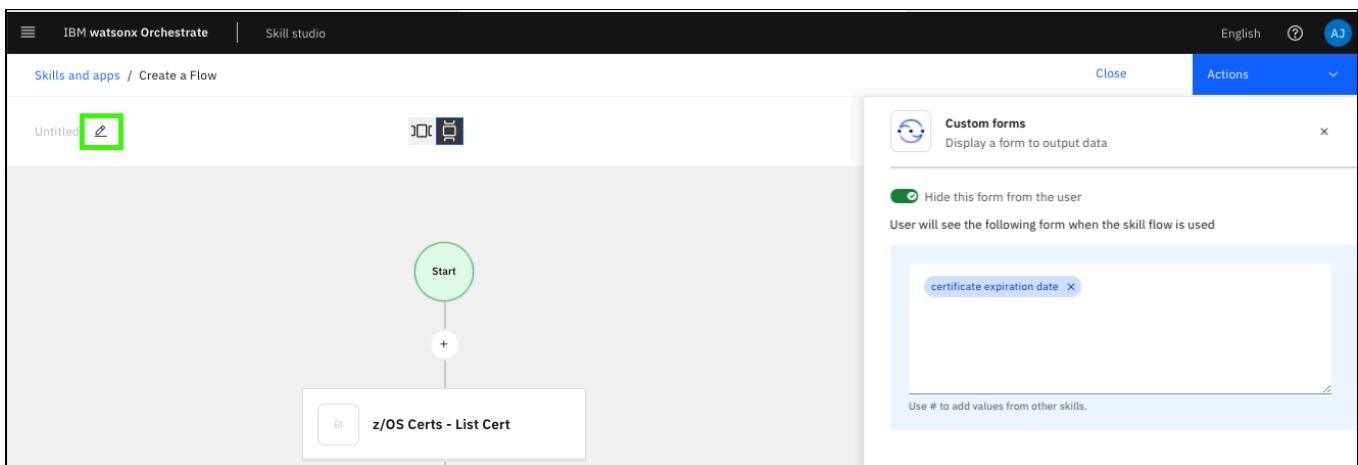
34. Enable the **Hide this form from the user** option.

#### ⓘ 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.



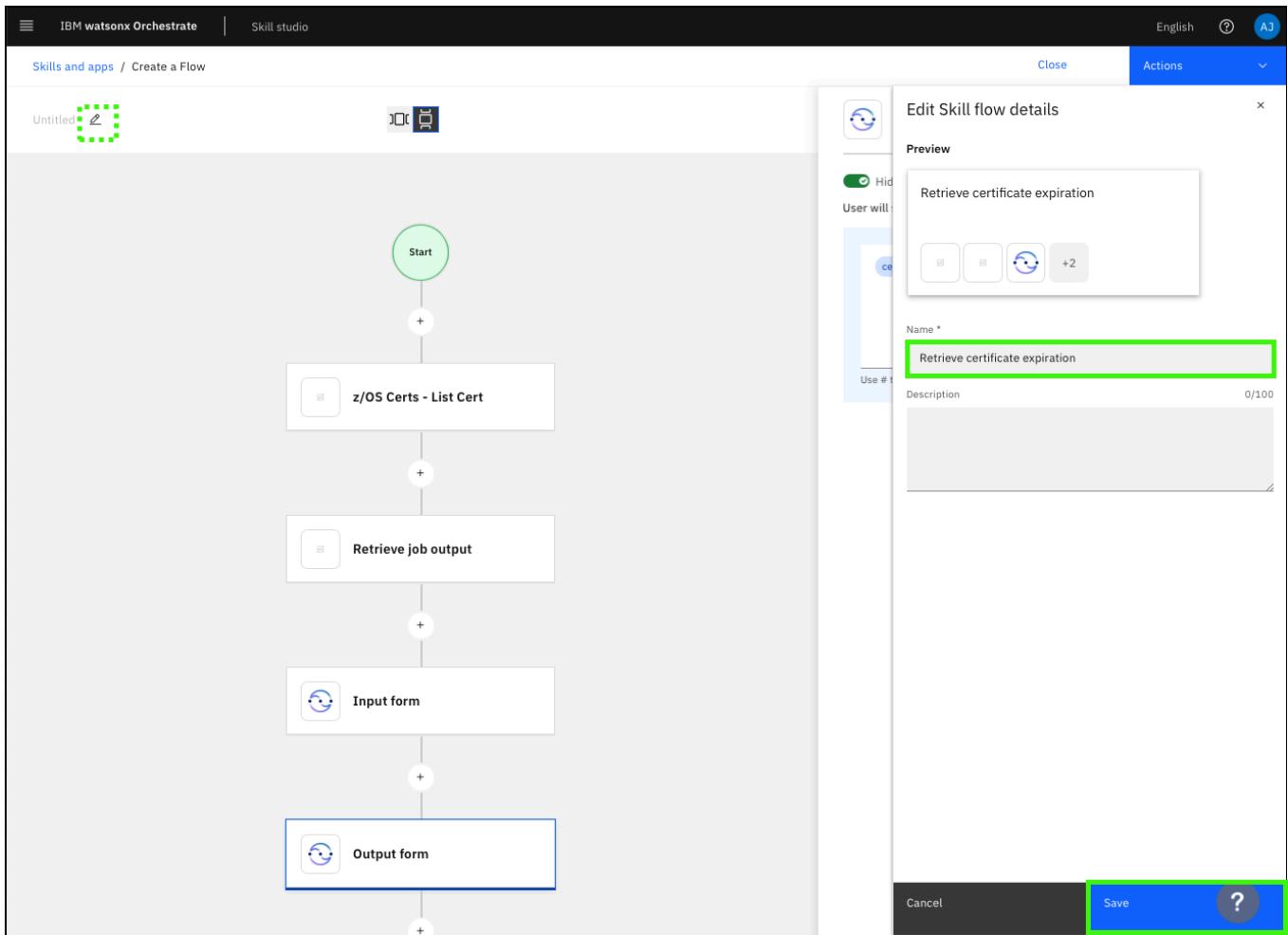
35. Click the **pencil** icon (✍).



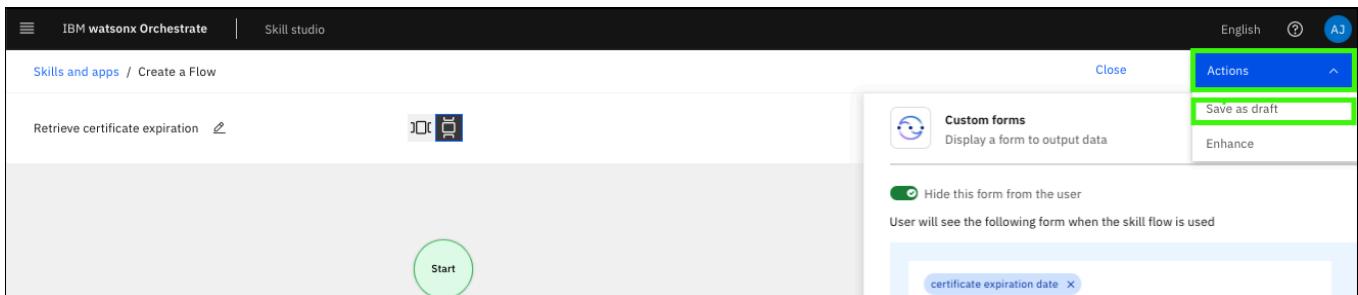
36. Enter **Retrieve certificate expiration** in the **Name** field and click **Save**.

**Name:**

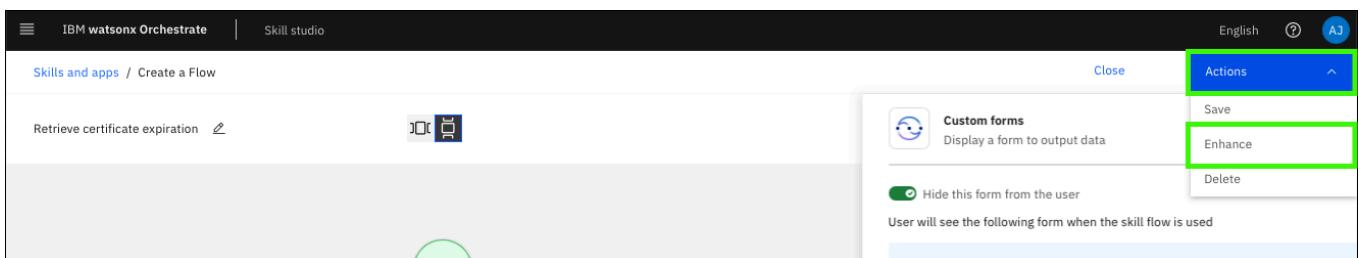
Retrieve certificate expiration



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**.

The screenshot shows the 'Skill studio' interface for enhancing a skill. The skill name is 'Retrieve certificate expiration'. The 'Name' field contains 'Retrieve certificate expiration'. The 'Description' field is empty. The 'API version\*' field is set to '1.0.0'. The 'Categories' and 'App' fields both contain 'Skill flows'. On the right, there are two preview sections: 'Preview' and 'The skill will look like this in the skill set.' Both previews show a card with the skill name and a small circular icon. At the bottom right, there are buttons for 'Cancel', 'Publish' (which is highlighted with a green border), and 'Save as draft'.

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.

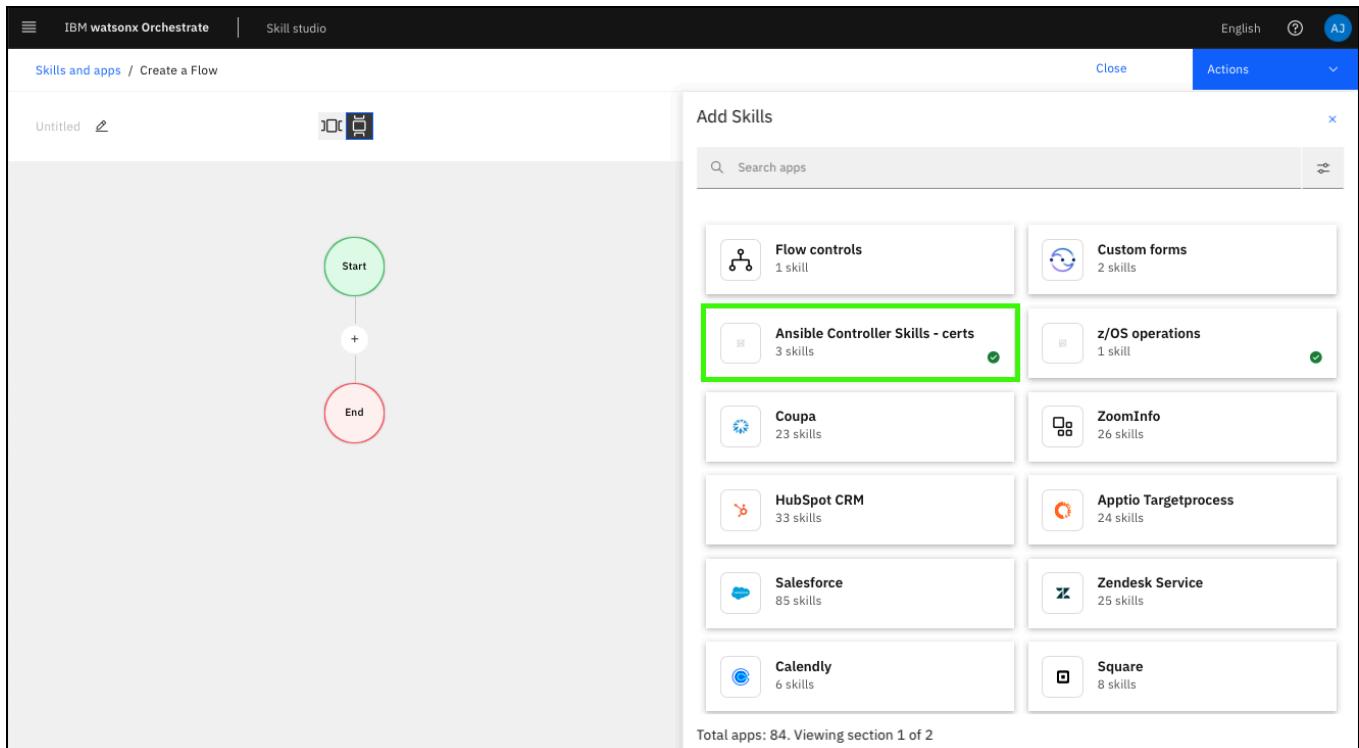
1. In **Skill studio**, click **Create** and then click **Skill flow**.

2. Click the + icon.

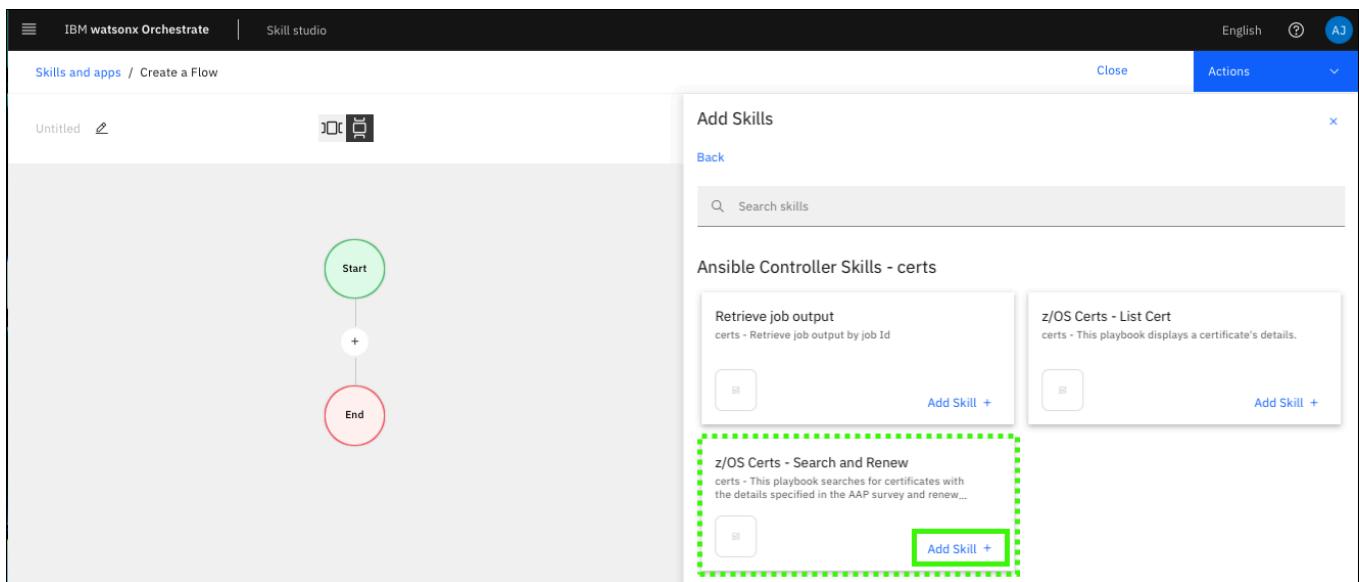
3. Click the certs app.



Search on certs if you do not see the tile for your app.

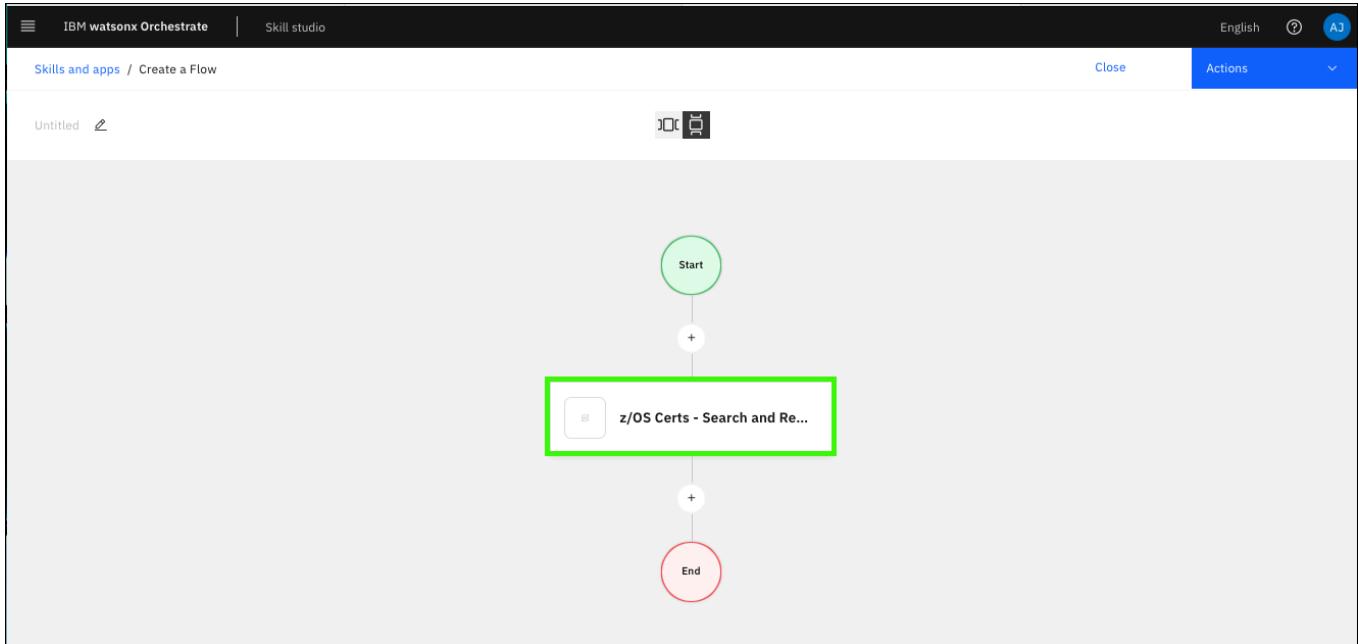


#### 4. Click Add Skill + in the z/OS Certs - Search and Renew tile.



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.

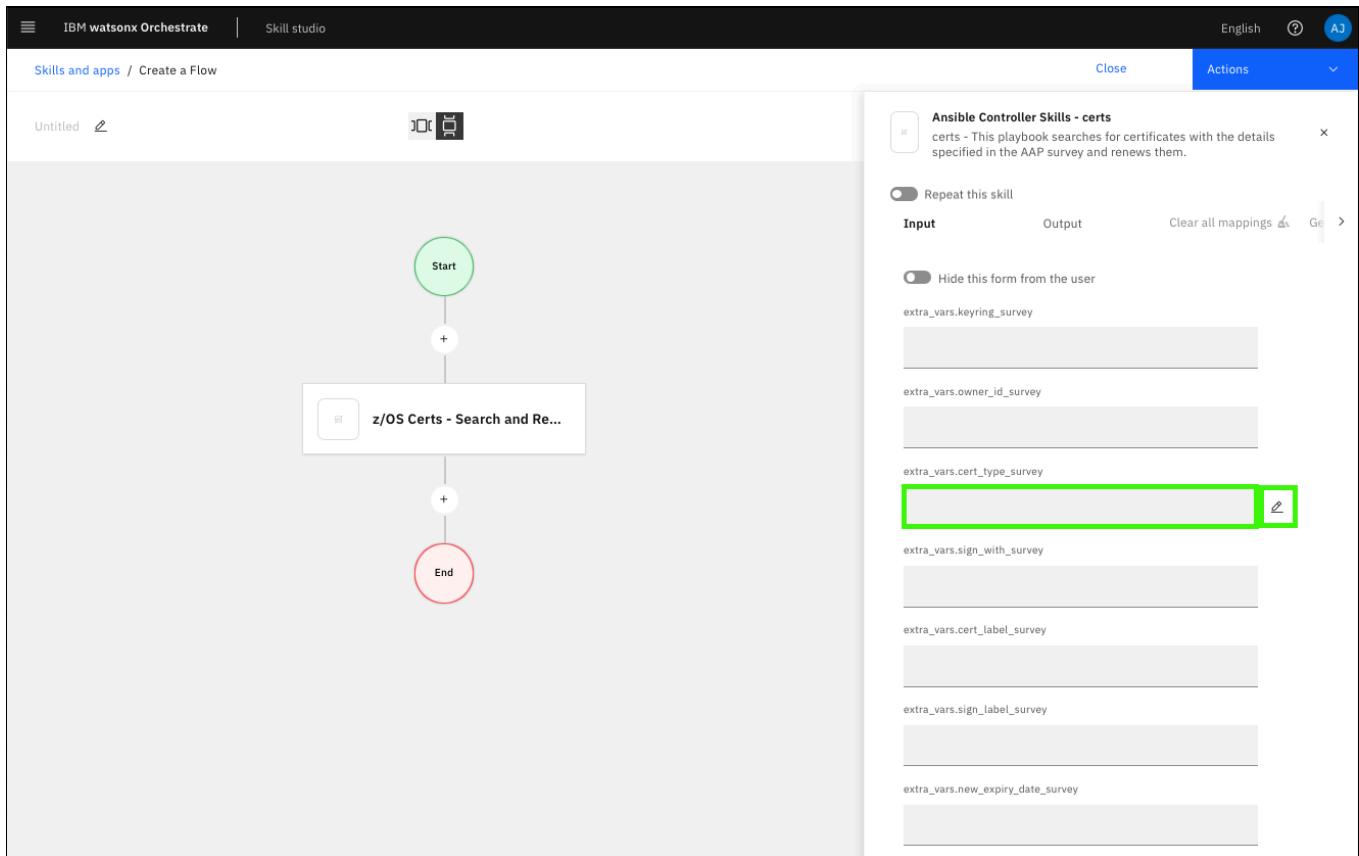
**Ansible Controller Skills - certs**  
certs - This playbook searches for certificates with the details specified in the AAP survey and renews them.

Output Clear all mappings

Repeat this skill  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 ().



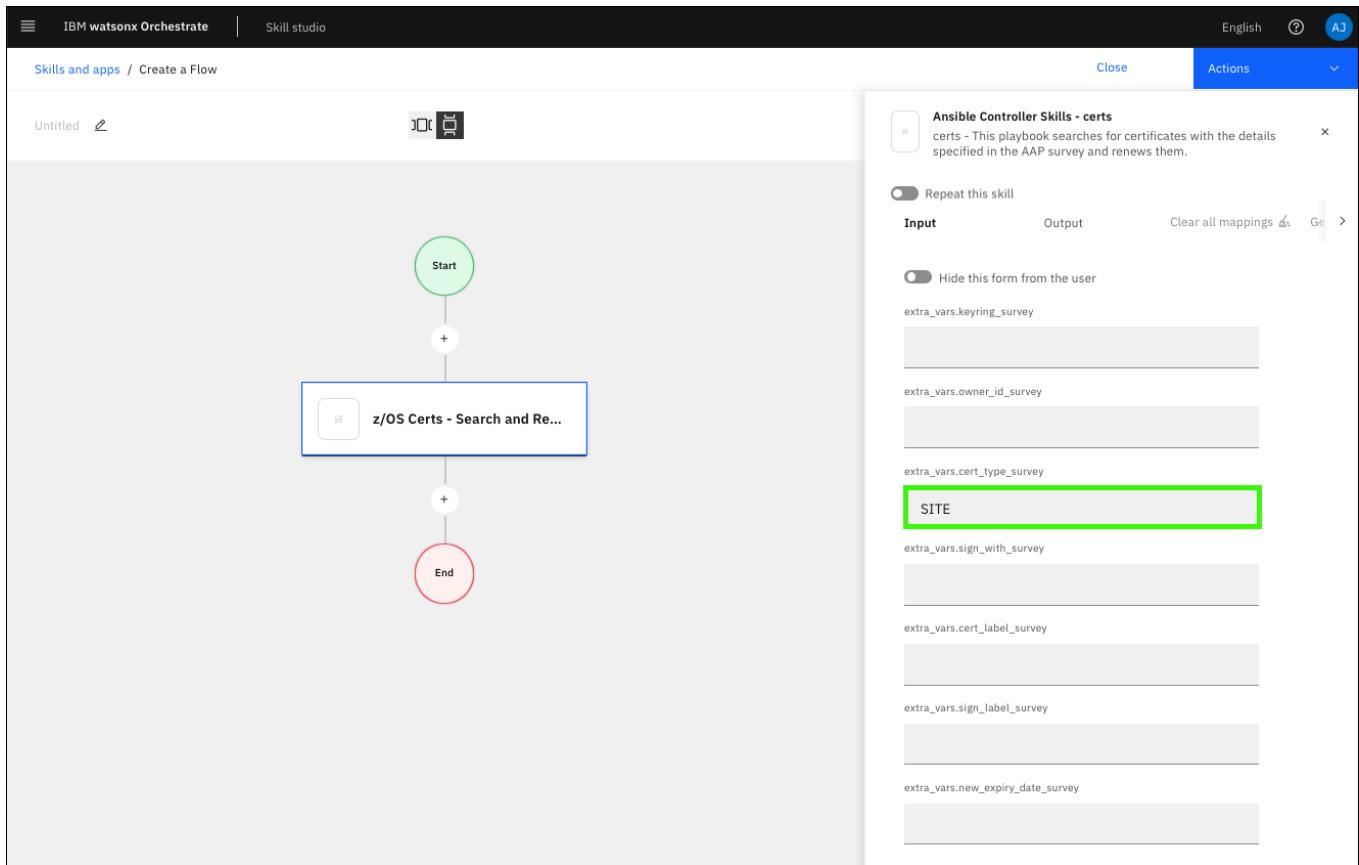
- Click in the **extra\_vars.cert\_type\_survey** input field and enter **SITE**.

**extra\_vars.cert\_type\_survey:**

SITE



**Do not enter spaces before or after the word SITE.**



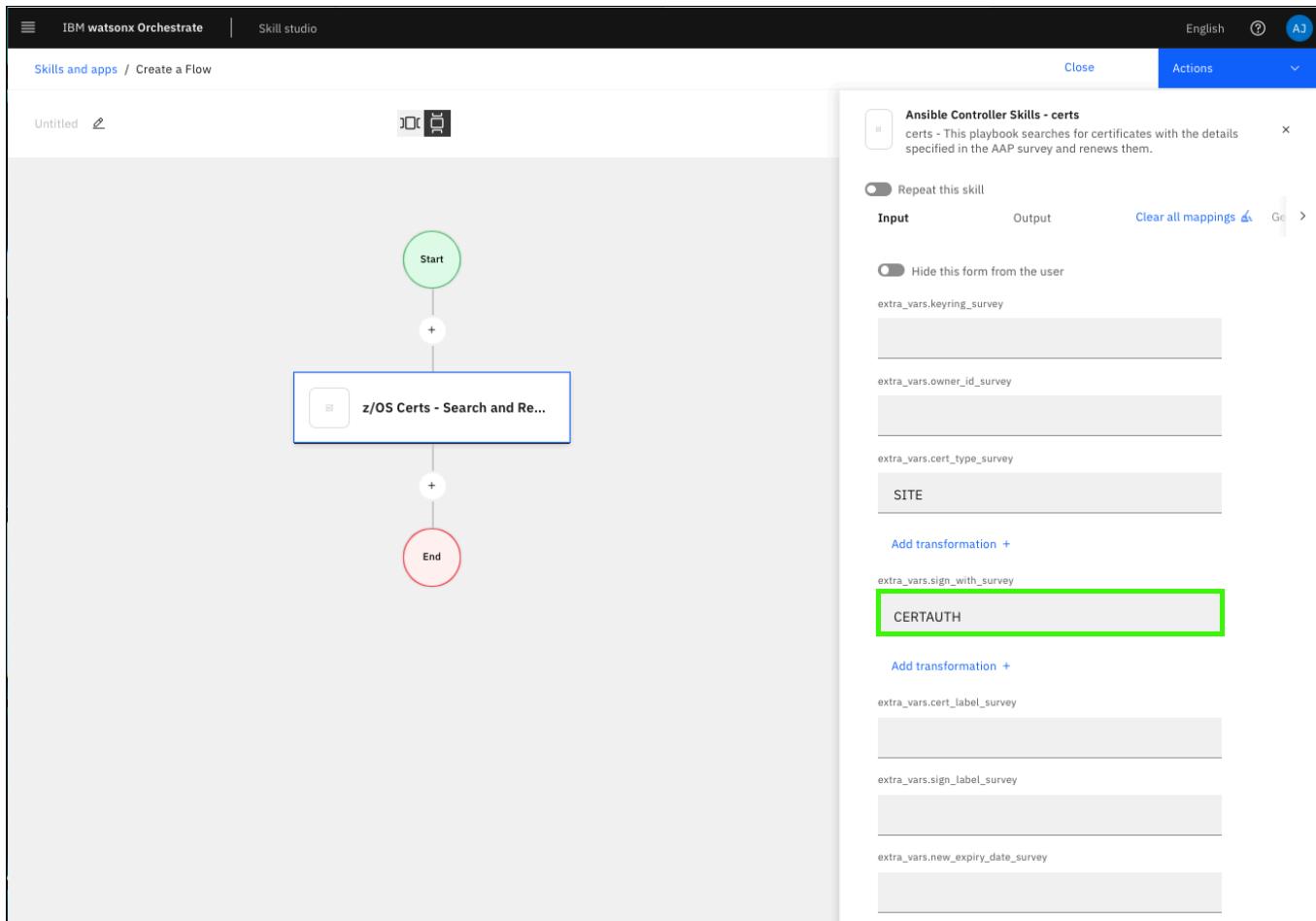
9. Repeat 7 and 8 for the **extra\_vars.sign\_with\_survey** field and enter the word **CERTAUTH**.

**extra\_vars.sign\_with\_survey:**

CERTAUTH



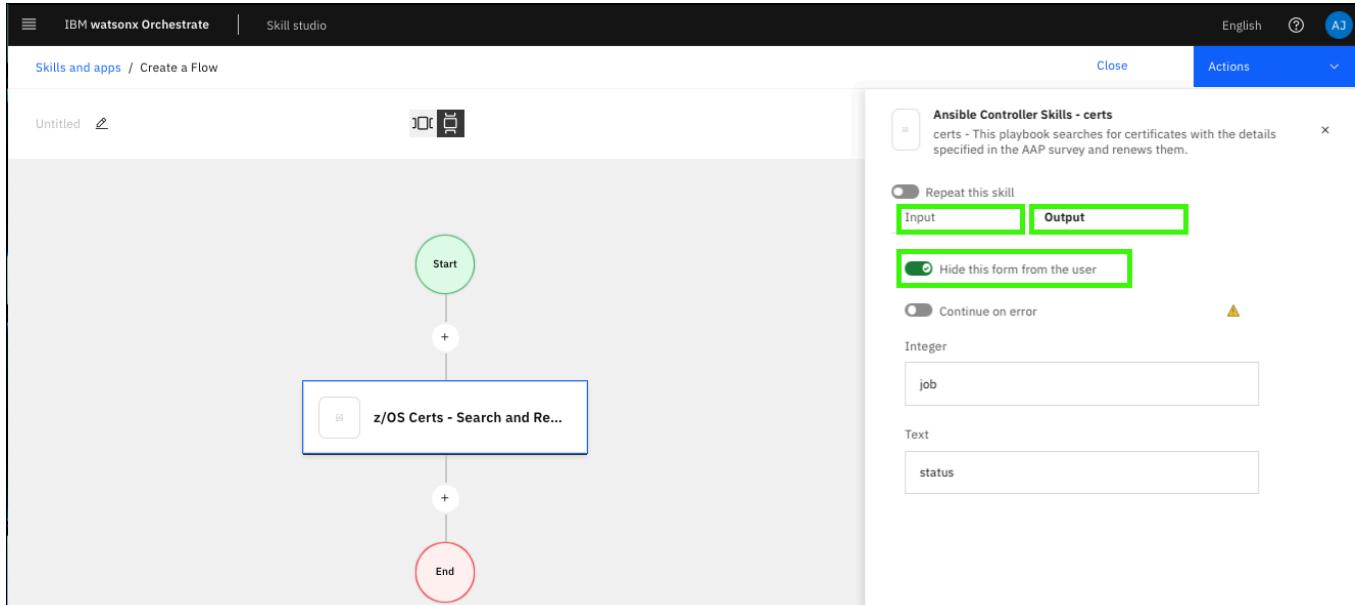
**Do not enter spaces before or after the word CERTAUTH.**



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

IBMWatsonx Orchestrate | Skill studio English ⓘ AJ

Skills and apps / Create a Flow

Untitled a

Edit Skill flow details

Preview

Cert Renewal skill flow

Name \* Cert Renewal skill flow

Description 0/100

Cancel Save

12. Click Actions and then click Save as draft.

IBMWatsonx Orchestrate | Skill studio English ⓘ AJ

Skills and apps / Create a Flow

Cert Renewal skill flow

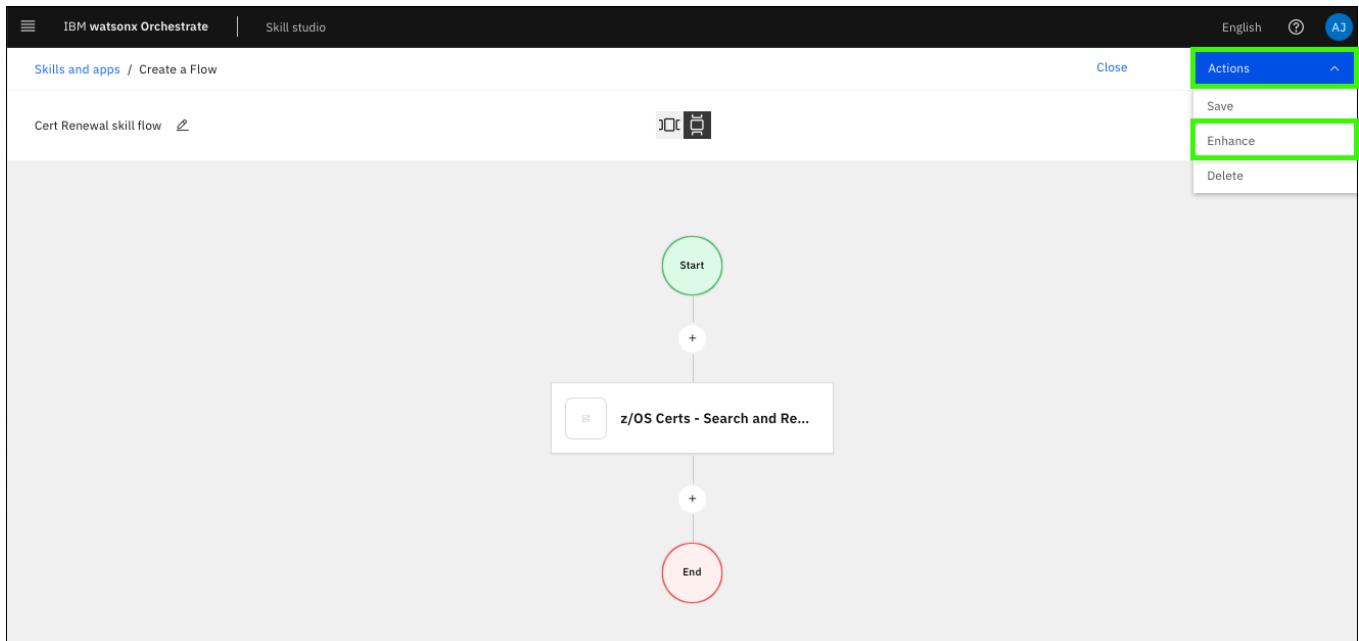
Start

+ z/OS Certs - Search and Re...

+ End

Actions Save as draft Enhance

13. Click Actions and then click Enhance.



#### 14. Review the skill flow settings and click **Publish**.

Name: Cert Renewal skill flow

Description: Enter the description

API version\*: 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**

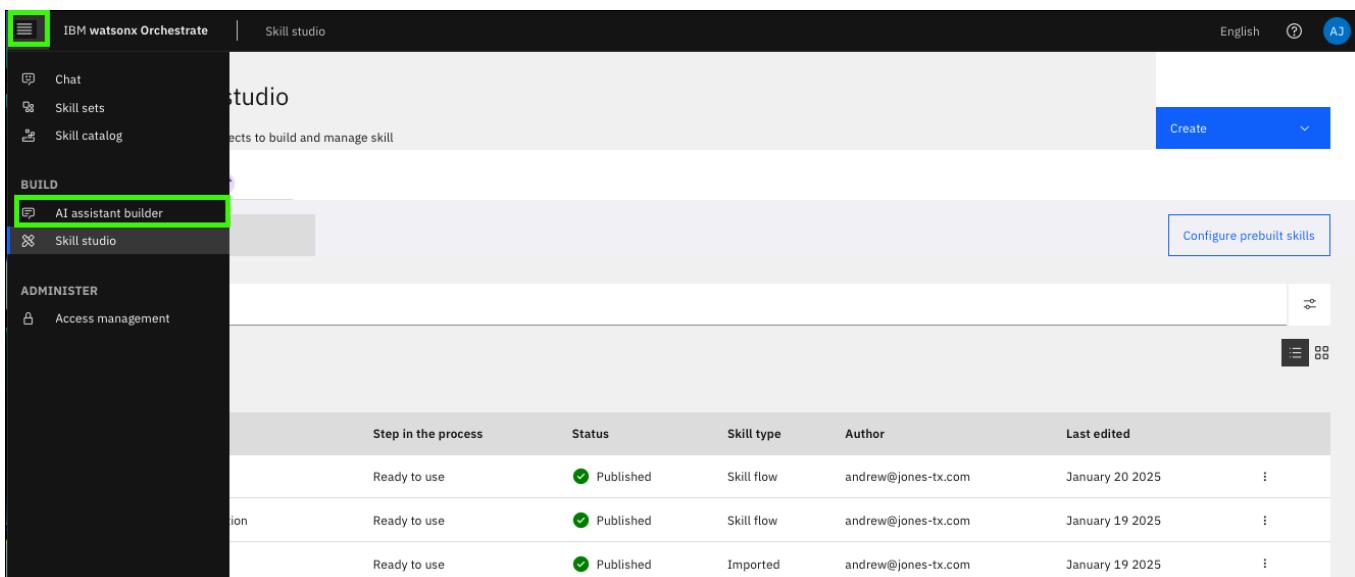
Add the skill flows to the assistant

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.

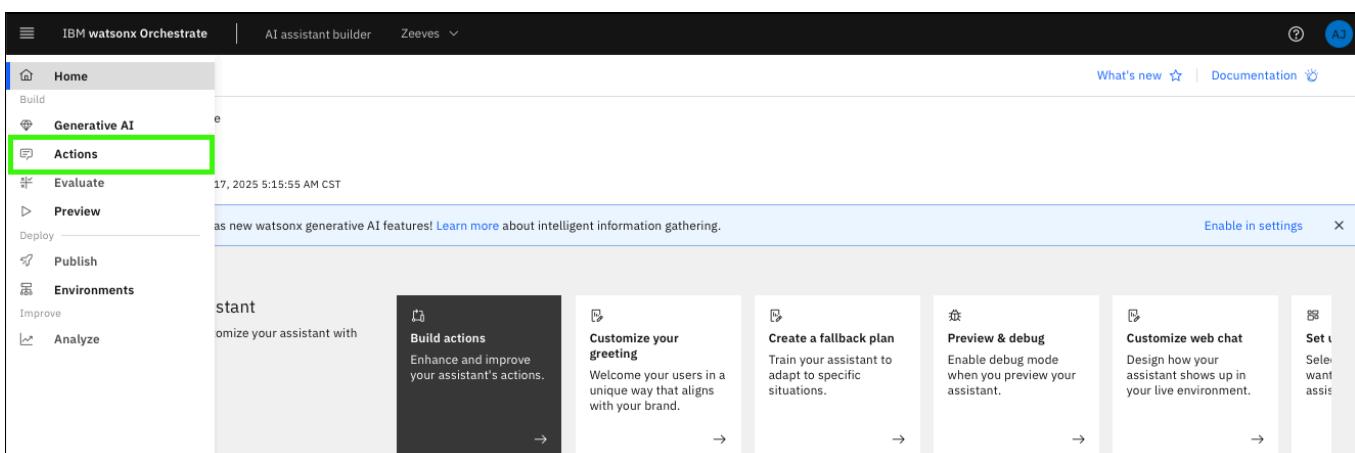
### 1. Open AI assistant builder in watsonx Orchestrate.



The screenshot shows the IBM Watsonx Orchestrate interface. The left sidebar has sections for Chat, Skill sets, Skill catalog, BUILD (with AI assistant builder selected and highlighted with a green box), and ADMINISTER (Access management). The main area is titled "Skill studio" and contains a sub-header "Selects to build and manage skill". A "Create" button is at the top right. Below it is a "Configure prebuilt skills" button. A table lists three skill flows:
 

Step in the process	Status	Skill type	Author	Last edited
Ready to use	<span>✓ Published</span>	Skill flow	andrew@jones-tx.com	January 20 2025
Ready to use	<span>✓ Published</span>	Skill flow	andrew@jones-tx.com	January 19 2025
Ready to use	<span>✓ Published</span>	Imported	andrew@jones-tx.com	January 19 2025

### 2. Click Actions.



The screenshot shows the IBM Watsonx Orchestrate interface with the Actions section selected in the sidebar (highlighted with a green box). The main area displays several actions:
 

- 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: Select what your assistant can do.

### 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: 'z/OS IPL Information' (last edited 'a day ago'), 'Zeeves-gather-facts-flow' (last edited '3 days ago'), and 'Retrieve IPL information' (last edited '21 hours ago'). A 'New action' button is located at the top right of the list.

#### 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 offers three choices: 'AI-guided action' (blue box), 'Skill-based action' (purple box, highlighted with a green border), and 'Custom-built action' (green box). Each choice has a brief description and a 'Beta' button.

#### 5. Click the Retrieve certificate expiration tile and then click Next.

The screenshot shows the 'Build an action from a skill' dialog. It displays a grid of skill tiles. The 'Retrieve certificate expiration' tile (highlighted with a green border) is selected. Other visible tiles include 'Cert Renewal skill flow', 'Retrieve job output', 'z/OS Certs - Search and Renew', 'z/OS Certs - List Cert', 'z/OS IPL Information', 'Zeeves-gather-facts-flow', and 'Retrieve job output'. The 'Next' button is highlighted with a green box in the top right corner.

#### 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.**

Customer starts with:  
Example: I want to pay my credit card bill.

**Configuration**

Skill Name: Retrieve certificate expiration  
Skill  
Id: composite\_skill\_\_1.0.0\_\_42842c359c044d3081624874f04...

**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: 0

Example: I want to pay my credit card bill.

7. Click **x** to close the **Retrieve certificate expiration** skill.

Customer starts with:  
Example: I want to pay my credit card bill.

**Configuration**

Skill Name: Retrieve certificate expiration  
Skill  
Id: composite\_skill\_\_1.0.0\_\_42842c359c044d3081624874f04...

**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: 0

Example: I want to pay my credit card bill.

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.

Name	Last edited	Examples count	Steps count	Status
z/OS IPL Information	a day ago	0	3	✓
Zeeves-gather-facts-flow	3 days ago	1	0	✓
Retrieve IPL information	a day ago	1	3	✓
Retrieve certificate expiration	13 minutes ago	0	3	✓
Cert Renewal skill flow	a few seconds ago	0	7	✓

## 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 +**.

Name	Last edited	Examples count	Steps count	Status
<code>z/OS IPL Information</code>	a day ago	0	3	Green checkmark
<code>Zeeves-gather-facts-flow</code>	3 days ago	1	0	Green checkmark
<code>Retrieve IPL Information</code>	a day ago	1	3	Green checkmark
<code>Retrieve certificate expiration</code>	13 minutes ago	0	3	Green checkmark
<code>Cert Renewal skill flow</code>	a few seconds ago	0	7	Green checkmark

### 2. Click **Custom-built-action**.

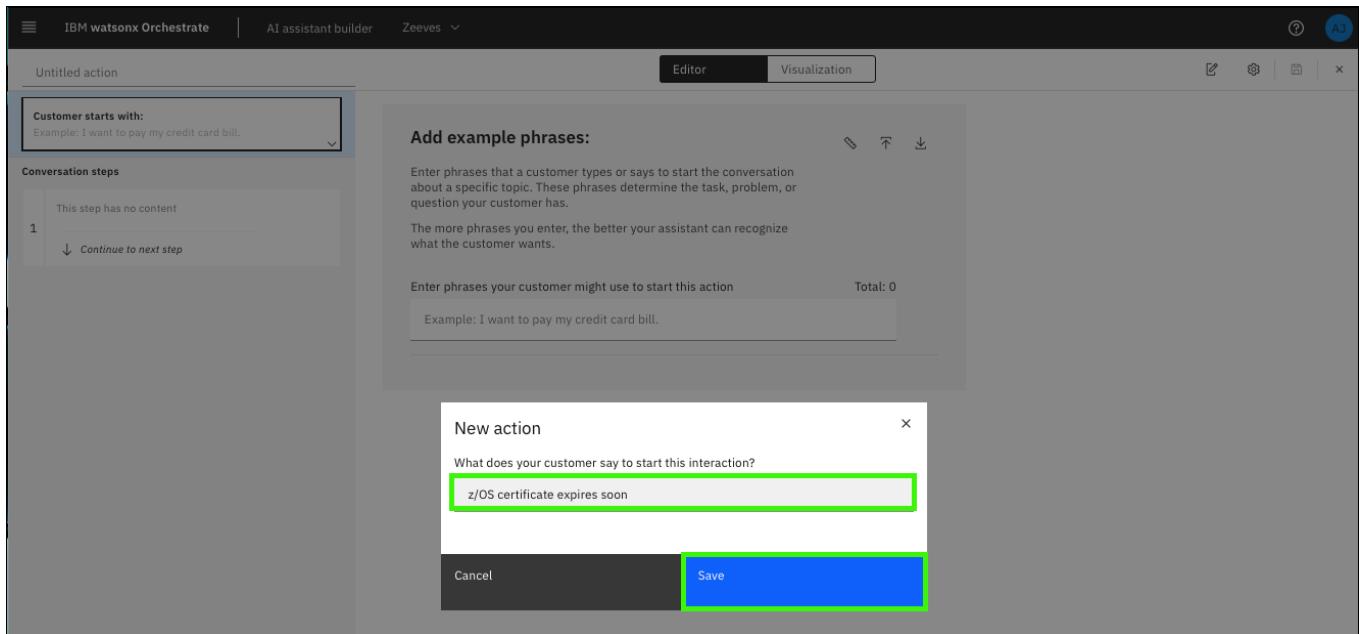
What kind of action do you want to build?

- 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.
- Custom-built action**: Design your own action step-by-step. Not sure how to start? Try adding pre-built action templates.

### 3. Enter `z/OS certificate expires soon` and then click **Save**.

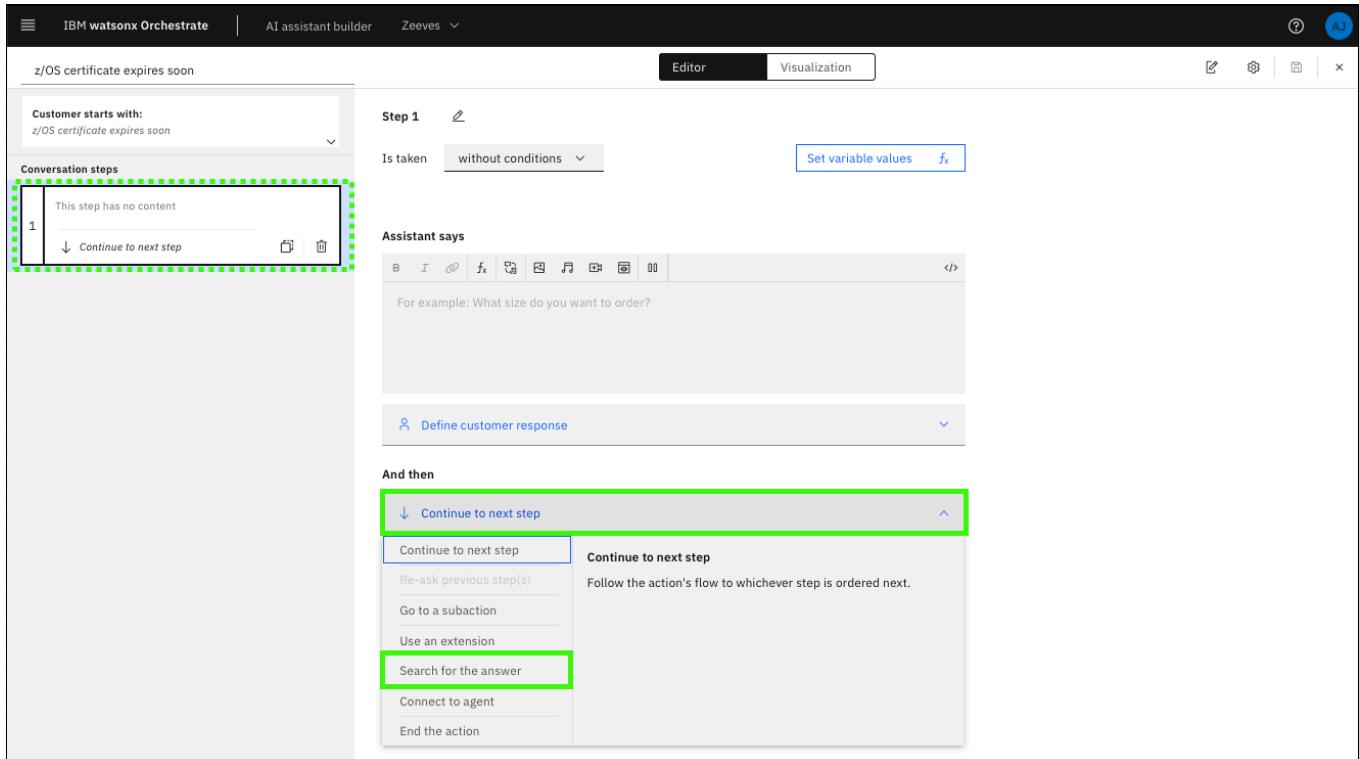
**What does your customer say to start this interaction:**

`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 is titled 'z/OS certificate expires soon'. On the left, there's a sidebar for 'Conversation steps' with one step listed: 'This step has no content' with a note 'Q Search for the answer'. The main panel is divided into sections: 'Step 1' (with a dropdown 'Is taken without conditions'), 'Assistant says' (containing the text 'For example: What size do you want to order?'), and 'And then' (with a dropdown menu containing 'Search for the answer', 'Custom query None (Customer's message will be the query)', 'Custom filter None (Optional)', and a button '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?
```

The screenshot shows the IBM Watsonx Orchestrate interface. A modal dialog titled "Search for the answer" is open over the main editor area. The dialog has tabs for "Search configuration" (which is selected) and "After generation". Under "Search configuration", there is a section for "Custom search query (Optional)" containing the text "My z/OS certificate is going to expire soon. How do I retrieve the expiration date for my certificate?". Below this is a "Custom results filter" section with two options: "Use default" (selected) and "Set new filter". The "Use default" option has a note: "The default filter is set in the search integration settings page." The "Set new filter" option has an example: "Example: enriched\_text.entities.text.:\"Boston, MA\"". At the bottom of the dialog are "Cancel" and "Apply" buttons, with "Apply" highlighted with a green border.

## 7. Click Next step+.

The screenshot shows the IBM Watsonx Orchestrate interface with the configuration for Step 1. The "Assistant says" section contains a rich text editor toolbar and a placeholder text "For example: What size do you want to order?". Below it is a dropdown menu "Define customer response". The "And then" section contains a "Search for the answer" input field with a custom query: "My z/OS certificate is going to expire soon. How do I retrieve the expiration date for my certificate?". There is also a "Custom filter" field set to "None (Optional)". At the bottom of the configuration area are "Edit settings" and "New step +" buttons, with "New step +" highlighted with a green border. In the bottom right corner, there is a "Preview" button.

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 IBM Watsonx Orchestrate interface with the 'AI assistant builder' tab selected. In the 'Conversation steps' section, step 1 has a placeholder message 'This step has no content'. Step 2 contains the message 'Would you like to run the skill to retrieve your certificate's expiration date?'. Below the message, there are buttons for 'Continue to next step', 'Edit', and 'Delete'. The 'Assistant says' field at the top of the step 2 configuration panel also displays the same message, which is highlighted with a green border. The 'Editor' tab is active at the top right.

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 'Define customer response' dropdown for step 2 being expanded. The 'Confirmation' option is selected and highlighted with a green border. The 'Confirmation' section includes a description: 'Enable customers to select from Yes or No response.' and a 'As buttons' section with 'Yes' and 'No' buttons. Other options listed in the dropdown include 'System', 'Free text', 'Regex', 'Number', 'Date', 'Time', and 'Saved'.

10. Click **Next step +**.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there's a sidebar with 'z/OS certificate expires soon' under 'Customer starts with'. Below it is a 'Conversation steps' section with two steps: Step 1 (empty) and Step 2. Step 2 contains the text 'Would you like to run the skill to retrieve your certificate's expiration date?'. Step 2 also has a 'Confirmation' button and a 'Continue to next step' link. The 'Is taken' dropdown in Step 2 is set to 'without conditions'. The 'Assistant says' field contains the question above. Below the 'Assistant says' field is a green dashed box containing 'Yes' and 'No' buttons, and 'View response' and 'Edit validation' links. Another green dashed box below it contains the 'Continue to next step' link. At the bottom left is a 'New step +' button, and at the bottom right 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 'Is taken' dropdown in Step 3 is now set to 'with conditions', which is highlighted with a green box. The 'Assistant says' field contains the placeholder text 'For example: Please select from the following options:'. Below the 'Assistant says' field is a green dashed box containing a 'Define customer response' dropdown and a 'Continue to next step' link. The rest of the interface is similar to the previous screenshot, showing the conversation steps and navigation 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 following details:

- Conversation steps:**
  - Step 1: "Customer starts with: z/OS certificate expires soon". A note says "This step has no content".
  - Step 2: "Would you like to run the skill to retrieve your certificate's expiration date?". A "Confirmation" button is present.
  - Step 3: "What is your certificate label?". A "Continue to next step" button is present.
- Step 3 Configuration:**
  - Is taken:** "with conditions" (highlighted with a green dashed box).
  - Conditions:** "If All of this is true:"  
- Condition 1: "2. Would you like to r... is Yes" (highlighted with a green dashed box).
    - And "Add condition +"
  - Assistant says:** "What is your certificate label?" (highlighted with a green box).
  - Define customer response:** "Free text" (highlighted with a green box).
  - And then:** "Continue to next step" (highlighted with a green box).

13. Click the **Define customer response** drop-down list and select **Free text**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. On the left, a sidebar displays 'z/OS certificate expires soon' and 'Conversation steps'. Step 1: 'This step has no content' with a search bar. Step 2: 'Would you like to run the skill to retrieve your certificate's expiration date?' with a 'Confirmation' button. Step 3: A green-dashed box highlights the condition '2. Would you like to r... is Yes'. Below it, a question 'What is your certificate label?' is shown with a 'Continue to next step' button. At the bottom of the sidebar is a 'New step +' button.

**Step 3**

Is taken **with conditions** **Set variable values**

Conditions 1 condition

If All of this is true:

2. Would you like to r... is Yes

and Add condition +

New condition group +

**Assistant says**

Define customer response

System Options Confirmation Tr Free text (highlighted) Regex Number Date Time Saved

Free text

Enable customers to write out any response using text input

As text My street is Main Street

Preview

14. Click **Next step+**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. On the left, there's a sidebar titled "z/OS certificate expires soon" with sections for "Customer starts with:" and "Conversation steps". The "Conversation steps" section shows three steps: 1. A search step with placeholder "Search for the answer". 2. A confirmation step asking "Would you like to run the skill to retrieve your certificate's expiration date?" with a "Confirmation" button. 3. A free text step asking "What is your certificate label?" with a "Continue to next step" button. On the right, under "Step 3", the condition "Is taken" is set to "with conditions". A condition is defined: "If All of this is true:" followed by "2. Would you like to r... is Yes". Below this, there's a "New condition group" button. Under "Assistant says", there's a response box containing "User enters free text" with edit and validation options. At the bottom, there's a "And then" section with a "Continue to next step" button. A "New step +" button is at the bottom left, and a "Preview" button is at the bottom right.

**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

Customer starts with:  
z/OS certificate expires soon

**Conversation steps**

- This step has no content
- 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?  
Tr Free text  
↓ Continue to next step
- This step has no content  
↓ Continue to next step

**Step 4**

Is taken: without conditions

Set variable values: fx

Assistant says:

For example: What type of transfer would you like to make?

Define customer response

And then:

↓ Continue to next step

## 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

Customer starts with:  
z/OS certificate expires soon

**Conversation steps**

- This step has no content
- 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?  
Tr Free text  
↓ Continue to next step
- 3 is Defined  
This step has no content  
↓ Continue to next step

**Step 4**

Is taken: with conditions

Set variable values: fx

**Conditions**

1 condition

If All of this is true:  
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?

Define customer response

**And then**

↓ Continue to next step

Continue to next step	Go to a subaction
Re-ask previous step(s)	Switch conversation flow to another action to perform a certain task
Go to a subaction	
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.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, there is a conversation history with four steps:

- Step 1:** Customer starts with: z/OS certificate expires soon
- 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

In Step 4, there is a "Go to" configuration section. The "Is taken" dropdown is set to "with conditions". A condition is defined: "3. What is your certifi... is defined".

A modal dialog titled "Go to a subaction" is open over the main interface. It contains the following options:

- Go to: "Select an action..." (highlighted with a green box)
- Cert Renewal skill flow
- Fallback: "Retrieve certificate expiration" (highlighted with a green box)

At the bottom of the modal are "Cancel" and "Apply" buttons, with "Apply" being highlighted with a blue box.

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 Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and a user icon. The main area is titled 'z/OS certificate expires soon'. On the left, there's a 'Conversation steps' panel with four steps numbered 1 to 4. Step 1: 'Customer starts with: z/OS certificate expires soon'. Step 2: 'Would you like to run the skill to retrieve your certificate's expiration date?'. Step 3: 'What is your certificate label?'. Step 4: 'This step has no content' and contains a link 'Go to subaction: Retrieve certificate expiration'. The right side of the screen shows 'Step 4' configuration under 'Is taken with conditions'. A condition is defined: 'If All of this is true: 3. What is your certifi... is defined'. Below this, there's an 'Assistant says' section with a rich text editor and a 'Define customer response' dropdown. Under 'And then', a subaction 'Go to a subaction' is selected, which points to 'Retrievez certificate expiration'. The 'Edit settings' button is highlighted with a green box.

#### 19. Click Set new value + and then select extra\_vars.cert\_label\_survey.

This screenshot shows the 'Edit passed values' dialog box from the previous step. The dialog title is 'Edit passed values' and it includes a note: 'Configure how your skill-based action leverages and stores information in assistant variables. [Learn more](#)'. A warning message states: 'Variables are not private if they are shared with Skill Studio.' The 'Edit variable values' tab is selected. A list of variables is shown, with '1. extra\_vars.cert\_label\_survey' highlighted with a green box. At the bottom of the dialog are 'Cancel' and 'Apply' buttons, and a note 'Goes 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. On the left, there's a conversation steps pane with four steps labeled 1 through 4. Step 1 is a search step, Step 2 is a confirmation step, Step 3 is a free-text step asking 'What is your certificate label?', and Step 4 is a defined step. Step 4 has a 'Goes to action' button labeled 'Retrieve certificate expiration'. In the center, the 'Step 4' configuration panel is open, showing 'Is taken' with 'with conditions' selected. Below it is the 'Conditions' section with a 'Passed values' sub-section. An 'Edit passed values' dialog is open over the main interface. This dialog has tabs for 'Edit variable values' (which is selected) and 'Response behavior'. Under 'Edit variable values', it says 'The values you select from this action will be passed to "Retrieve certificate expiration" skill.' Below this is a 'Set' dropdown set to 'extra\_vars.cert\_label\_survey' and a 'To' dropdown. A yellow circle highlights the 'Action step variables' option in the 'To' dropdown menu. At the bottom of the dialog are 'Cancel' and 'Apply' buttons.

21. Click **Apply**.

This screenshot shows the same interface as the previous one, but the 'To' dropdown in the 'Edit variable values' dialog now displays 'extra\_vars.cert\_label\_survey' and '3. What is your certificate label?' with a green dashed border around them, indicating they have been selected. The 'Apply' button at the bottom of the dialog is also highlighted with a green box.

22. 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 'z/OS certificate expires soon'. On the left, there's a 'Conversation steps' panel showing four steps: 1. 'This step has no content', 2. 'Would you like to run the skill to retrieve your certificate's expiration date?', 3. 'What is your certificate label?', and 4. 'This step has no content'. Step 4 has a blue border and a 'New step +' button at the bottom. The right side shows the configuration for Step 4:

- Step 4:** Set to 'Is taken with conditions'.
- Conditions:** If 'All' of this is true:
  - 3. What is your certifi... is defined
  - and Add condition +
- Assistant says:** Placeholder text: 'For example: What type of transfer would you like to make?' with rich text editor tools.
- And then:** Action selected: 'Go to a subaction' (highlighted in blue).
  - Goes to action: Retrieve certificate expiration
  - Pass values: 1. extra\_vars.cert\_label\_survey
  - Upon return: Continue

### 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 a user icon. The main area is titled 'z/OS certificate expires soon'. The 'Editor' tab is selected. On the left, a 'Conversation steps' pane lists five steps: 1. 'This step has no content', 2. 'Would you like to run the skill to retrieve your certificate's expiration date?', 3. 'What is your certificate label?', 4. 'This step has no content', and 5. 'Below is your certificate's expiration date:'. Step 5 is highlighted with a green dashed border. The 'Step 5' configuration pane shows 'Is taken' set to 'without conditions' (with 'with conditions' also available). Below it is an 'Assistant says' text editor containing the placeholder 'For example: What size do you want to order?'. The 'Conditions' section is collapsed. The 'And then' section contains a 'Continue to next step' button.

24. Enter the following text in the **Assistant says** field.

**Assistant says:**

Below is your certificate's expiration date:

The screenshot shows the same interface after applying conditions to Step 5. The 'Conditions' pane now displays a single condition: 'If All of this is true: 3. What is your certifi... is defined'. The 'Assistant says' text editor now contains the text 'Below is your certificate's expiration date:'.

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 with the 'AI assistant builder' tab selected. The main area displays a conversation flow with five steps:

- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  - This step has no content
  - Would you like to run the skill to retrieve your certificate's expiration date? (Confirmation)
  - What is your certificate label? (Free text)
  - This step has no content (Defined)
  - Below is your certificate's expiration date: <br /> (Defined)

**Step 5:** Is taken with conditions

**Conditions:**

- If All of this is true:
  - 3. What is your certifi... is defined
- and Add condition +

**Assistant says:**

```
Below is your certificate's expiration date:  
$
```

A callout box highlights the '\$' character in the 'Assistant says' field, indicating that it is a special key used to list available functions.

26. Click **Retrieve certificate expiration (step 4)** and then click **Retrieve certificate expiration result variable**.

**Step 5**

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)

Below is your certificate's expiration date:  
1 2 3 4. Retrieve certificate expiration (step 4)

Assistant variables

Integration variables

An Continue to next step

27. Review the **Assistant says** field and then click **Save** (💾).

**Step 5**

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**

Below is your certificate's expiration date:  
1 2 3 4. Retrieve certificate expiration (step 4)

Define customer response

And then

Continue to next step

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**.

The screenshot shows the IBM Watsonx Orchestrate interface with the 'z/OS certificate expires soon' skill selected. The 'Editor' tab is active. On the left, the conversation steps are listed:

- Customer starts with: z/OS certificate expires soon
- This step has no content
- Would you like to run the skill to retrieve your certificate's expiration date?
- This step has no content
- Below is your certificate's expiration date: **1 2 3 Step 4**

Step 5 is currently being configured with the condition "Is taken with conditions". A condition is defined: "If All of this is true: 3. What is your certifi... is defined". The 'Assistant says' section contains the response: "Below is your certificate's expiration date: 1 2 3 Step 4. Retrieve certificate expiration re...". The 'And then' section contains the action: "Continue to next step".

## 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. On the left, a sidebar titled "z/OS certificate expires soon" lists "Conversation steps" numbered 1 through 5. Step 1: "Customer starts with: z/OS certificate expires soon". Step 2: "Would you like to run the skill to retrieve your certificate's expiration date?". Step 3: "What is your certificate label?". Step 4: "This step has no content". Step 5: "Below is your certificate's expiration date: 1 2 3 Step 4". The main panel shows "Step 5" configuration under "Is taken with conditions". A condition is defined: "If All of this is true: 3. What is your certifi... is defined". Below this, there are sections for "Assistant says" (with a rich text editor) and "Define customer response". On the right, a "Preview" window shows a simulated conversation: "Greet customer [default] Welcome, how can I assist you?" followed by a message from the assistant: "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 AI assistant builder interface. On the left, a sidebar titled "z/OS certificate expires soon" lists "Conversation steps". Step 1: "Customer starts with: z/OS certificate expires soon". Step 2: "Would you like to run the skill to retrieve your certificate's expiration date?". Step 3: "What is your certificate label?". Step 4: "This step has no content". Step 5: "Below is your certificate's expiration date: 1 2 3 4. Retrieve certificate expiration re...". A "New step +" button is at the bottom.

**Step 5:** "Is taken with conditions" (selected). Conditions: "If All of this is true: 3. What is your certifi... is defined" and "Add condition +".

**Assistant says:** "Below is your certificate's expiration date: 1 2 3 4. Retrieve certificate expiration re..."

**Preview:** Shows a simulated user interaction. The user asks "z/OS certificate expires soon". The AI responds with "Below is your certificate's expiration date: 1 2 3 4. Retrieve certificate expiration re...". The user then asks "What is your certificate label?". The AI responds with "What is your certificate label?". The user asks "Would you like to run the skill to retrieve your certificate's expiration date?". The AI responds with "Would you like to run the skill to retrieve your certificate's expiration date?". The preview pane includes a "Search info" section with instructions about retrieving certificate expiration dates using commands like 'ODTE' or 'RACDCERT'.

#### 4. Review the response and enter DEMOCRT .

**Prompt:**

DEMOCRT

z/OS certificate expires soon

**Step 5**

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**

Below is your certificate's expiration date:  
1. 4. Retrieve certificate expiration re...

**Define customer response**

Would you like to run the skill to retrieve your certificate's expiration date?  
Yes No

You 10:07 AM Yes

10:07 AM z/OS certificate expires soon recognized

What is your certificate label?

DEMOCERT

5. Click **Apply**.

The screenshot shows the IBM Watsonx Orchestrate interface. On the left, a sidebar lists 'Conversation steps' numbered 1 through 5. Step 1: 'Customer starts with: z/OS certificate expires soon'. Step 2: 'Would you like to run the skill to retrieve your certificate's expiration date?'. Step 3: 'What is your certificate label?'. Step 4: 'This step has no content'. Step 5: 'Below is your certificate's expiration date: 1+3 Step 4'. Below the steps is a 'New step +' button. At the top right, tabs for 'Editor' and 'Visualization' are shown. In the center, 'Step 5' is selected, showing conditions: 'If 3. What is your certifi... is defined'. To the right, a 'Preview' window shows a user message 'z/OS certificate expires soon' and an AI response 'What is your certificate label?'. A configuration window for a skill named 'Retrieve certificate expiration' is open, showing an 'id' field with value '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 interface. On the left, a sidebar titled "z/OS certificate expires soon" displays a "Conversation steps" list:

- Step 1: "Customer starts with: z/OS certificate expires soon". A note says "This step has no content".
- Step 2: "Would you like to run the skill to retrieve your certificate's expiration date?". A "Confirmation" button is shown.
- Step 3: "What is your certificate label?". A "Free text" input field is shown.
- Step 4: "This step has no content". A note says "Go to subaction: Retrieve certificate...".
- Step 5: "Below is your certificate's expiration date: 1 2 3 Step 4". A "Continue to next step" button is shown.

In the center, under "Step 5", the "Is taken" dropdown is set to "with conditions". A condition is defined: "If All of this is true: 3. What is your certifi... is defined".

On the right, the "Preview" pane shows a conversational message:

```

10:18 AM
↳ Conversational skill called
go to Retrieve certificate expiration

Retrieve certificate expiration
id *
28

```

Below the preview, a note says "There are no additional steps for this action. Add a new step or end the action."

2. Click the **Is taken** option list and select **with conditions**.

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 conversation flow titled 'z/OS certificate expires soon'. The flow consists of six steps:

- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  - Step 1:** Is taken (with conditions) - This step has no content. It includes a dropdown menu with options: 'without conditions' (selected), 'with conditions', and 'Set variable values'.
  - Assistant says:** Would you like to run the skill to retrieve your certificate's expiration date? (Confirmation)
  - Step 2:** 2 is Yes
  - Step 3:** What is your certificate label? (Free text)
  - Step 4:** This step has no content
  - Step 5:** Below is your certificate's expiration date: Step 4 (Go to subsection: Retrieve certificate)
  - Step 6:** This step has no content

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 various icons. The main area displays a conversation flow titled 'z/OS certificate expires soon'. The flow consists of six steps:

- Customer starts with:** z/OS certificate expires soon
- Conversation steps:**
  - Step 1:** Is taken with conditions. Condition: 1 (Retrieve certificate) is defined.
  - Step 2:** Confirmation. Question: Would you like to renew your certificate? Response: Continue to next step.
  - Step 3:** Question: What is your certificate label? Response: Continue to next step.
  - Step 4:** Question: This step has no content. Response: Go to subaction: Retrieve certificate.
  - Step 5:** Question: Below is your certificate's expiration date: Step 4. Response: Continue to next step.
  - Step 6:** Is taken with conditions. Condition: 4 (Retrieve certificate) is defined.

**Step 1 Conditions:**

If All of this is true:

- 1 (Retrieve certificate) is defined

**Assistant says:**

Would you like to renew your certificate?

**Define customer response:**

Confirmation

**And then:**

Continue to next step

4. Click the **Define customer response** option list and select **Confirmation**.

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 conversation flow titled 'z/OS certificate expires soon'.

**Conversation steps:**

- Customer starts with: z/OS certificate expires soon
- Would you like to renew your certificate? (Is taken with conditions)
- Would you like to run the skill to retrieve your certificate's expiration date? (Confirmation)
- What is your certificate label? (Free text)
- This step has no content (Go to subaction: Retrieve certifica...)
- Below is your certificate's expiration date: Step 4 (Confirmation)
- Would you like to renew your certificate? (Confirmation)

**Step 1:** Is taken with conditions

**Conditions:** If All of this is true:

- 1 Retrieve certificate is defined

**Assistant says:**

Would you like to renew your certificate?

**Define customer response:**

**System:**

- Options
- Confirmation** (highlighted with a green box)
- Free text
- Regex
- Number
- Date
- Time

**Confirmation:** Enable customers to select from Yes or No response.

As buttons: Yes, No

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 a user icon. 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' (highlighted in green). The 'Is taken' dropdown is set to 'with conditions'. A condition is defined: 'If 1 is Retrieve certificate... is defined'.
- Step 2:** 'Would you like to renew your certificate?' (Confirmation). The response 'Yes' leads to Step 3; 'No' leads to Step 4.
- Step 3:** 'What is your certificate label?' (Free text). Response 'label' leads to Step 4.
- Step 4:** 'This step has no content' (Confirmation). Response 'Step 4' leads to Step 5.
- Step 5:** 'Below is your certificate's expiration date: 123 Step 4' (Confirmation). Response 'Continue to next step' leads to Step 6.
- Step 6:** 'Would you like to renew your certificate?' (Confirmation). Response 'Yes' leads back to Step 2; 'No' leads to a new step.

The right side of the interface shows the 'Assistant says' section, which contains the text 'Would you like to renew your certificate?'. Below this is a conditional block with 'Yes' and 'No' options, and buttons for 'View response' and 'Edit validation'. A green dashed box highlights this conditional logic. The 'And then' section shows a continuation arrow pointing to 'Continue to next 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 confirmation step asking "Would you like to renew your certificate?". It has a condition "with conditions" selected.
- Step 2:** A step with no content, labeled "This step has no content". It has a condition "without conditions" selected.
- Step 3:** A confirmation step asking "Would you like to run the skill to retrieve your certificate's expiration date?". It has a condition "with conditions" selected.
- Step 4:** A step where the user is asked "What is your certificate label?", with a "Free text" input field.
- Step 5:** A step with no content, labeled "This step has no content". It has a condition "Defined" selected.
- Step 6:** A step where the user is asked "Below is your certificate's expiration date:", with a "Step 5" input field.
- Step 7:** A confirmation step asking "Would you like to renew your certificate?", with a "Confirmation" button.

**Assistant says:**

Would you like to renew your certificate?

**View response** **Edit validation**

**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 workflow titled "z/OS certificate expires soon" is displayed. The steps are numbered 1 through 7. Step 7 is highlighted with a green dashed box. The condition for Step 7 is "Is taken with conditions" set to "All" of this is true: "6. Would you like to r... is Yes". Below this, there is a "Define customer response" dropdown set to "Free text". The "Assistant says" section contains the message: "What date would you like to set the renewed certificate's expiration date to? Please enter in the form of YYYY-MM-DD." The "And then" section has a "Continue to next step" button.

- Click the **Define customer response** option list and select **Free text**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. On the left, a vertical list of steps is visible, each with a number, a status indicator (e.g., 'is Yes'), and a description. Step 1: 'Search for the answer'. Step 2: 'Would you like to run the skill to retrieve your certificate's expiration date? Confirmation'. Step 3: 'Continue to next step'. Step 4: 'This step has no content'. Step 5: 'Go to subsection: Retrieve certificate'. Step 6: 'Would you like to renew your certificate? Confirmation'. Step 7: 'Continue to next step'. Step 7 is highlighted with a green dashed box.

**Step 7**

Is taken with conditions with conditions Set variable values fx

Conditions 1 condition x

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

Enable customers to write out any response using text input

As text

My street is Main Street

Free text

Options

Confirmation

RegEx

Number

Date

Time

9. Click New step +.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. On the left, a vertical stack of cards represents a skill flow with steps numbered 1 through 7. Step 7 is currently selected. The right side of the screen displays the configuration for Step 7, specifically the 'Is taken' condition section. It shows a condition: 'If All of this is true:' followed by '6. Would you like to r... is Yes'. Below this, there's a link to 'Add condition +' and a button for '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.

A green dashed box highlights the 'User enters free text' input field, which contains a date in the format YYYY-MM-DD.

**And then:**

**Step 8:** Continue to next step

At the bottom right, there are 'Preview' and 'Next' buttons.

#### 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. On the left, there is a vertical list of steps numbered 2 through 8. Steps 2, 3, 4, 5, 6, and 7 have their own sections with specific questions and responses. Step 8 is currently selected and expanded. In the 'Step 8' section, there is a dropdown menu for 'Is taken' with three options: 'without conditions' (selected), 'with conditions', and another 'without conditions' option. Below this is a 'Assistant says' field with a rich text editor and a placeholder 'For example: What type of transfer would you like to make?'. At the bottom of the expanded Step 8 section, there is a 'And then' section with a 'Continue to next step' button.

## 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 top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and a user icon. The main area displays a workflow titled 'z/OS certificate expires soon'.

**Workflow Steps:**

- Step 1: Confirmation (with condition: 2 is Yes)
- Step 2: Continue to next step
- Step 3: What is your certificate label? (with condition: 3 is Defined)
- Step 4: Continue to next step
- Step 5: This step has no content
- Step 6: Go to subaction: *Retrieve certificate...* (with condition: 3 is Defined)
- Step 7: Continue to next step
- Step 8: Confirmation (with condition: 6 is Yes)
- Step 9: Continue to next step

**Step 8 Configuration:**

- Condition:** 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 then:** Continue to next step

**Assistant says:**

Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.

**Define customer response:**

12. Click the **And then** option list and select **Go to a subaction**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. On the left, a vertical list of steps is displayed:

- z/OS certificate expires soon
- 2 Confirmation (with condition: 2 is Yes)
- What is your certificate label?
- 3 Free text (with condition: 3 is Defined)
- This step has no content
- 4 Go to subaction: Retrieve certificate...
- 5 Step 4 (with condition: 3 is Defined)
- 6 Confirmation (with condition: 6 is Yes)
- 7 Free text (with condition: 7 is Defined)
- 8 Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...

On the right, the "Editor" tab is selected. Step 8 is currently being edited. The "Conditions" section shows a single condition: "If All of this is true: 7. What date would you... is defined". Below this, there is a "New condition group +" button.

The "Assistant says" section contains the message: "Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below." This message is highlighted with a green dashed border.

The "And then" section shows a dropdown menu with the following options:

- 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

At the bottom right of the editor area, there are "Preview" and "Next" buttons.

13. Click the **Go to** option list and select the **Cert Renewal skill flow**.

The screenshot shows the IBM Watsonx Orchestrator interface. On the left, a vertical list of steps is visible, each with a number, a status indicator, and a description. Step 8 is currently selected and expanded. The configuration dialog for Step 8 is open on the right, titled "Step 8". It shows the condition "Is taken with conditions" and a single condition: "If All of this is true: 7. What date would you like to renew your certificate? is defined". Below this, there is a "Conditions" section with an "Add condition +". A modal window titled "Go to a subaction" is displayed over the main interface. This modal has sections for "Settings" (containing "Go to" dropdowns for "Select an action...", "Cert Renewal skill flow", "Fallback", and "Retrieve certificate expiration"), "And then" (containing "Goes to action" with a note "Missing action. Edit settings.", "Pass values", "Upon return" set to "Continue", and buttons for "Edit settings" and "Edit passed values"), and "Privacy" (with a checkbox "Protect data returned after the subaction completes"). At the bottom of the modal are "Cancel" and "Apply" buttons, with "Apply" being highlighted.

#### 14. Click Apply.

This screenshot is similar to the previous one but shows the "Apply" button in the "Go to a subaction" modal dialog being clicked. The dialog is now partially closed, and the main workflow editor interface is visible again. The configuration for Step 8 remains the same, showing the condition "Is taken with conditions" and the "Cert Renewal skill flow" action selected in the "Go to" dropdown.

#### 15. Click Edit passed values.

Edit the passed values to use them in the **Cert Renewal** skill flow subaction.

16. Click **Set new value +** and then select **extra\_vars.cert\_label\_survey**.

17. In the **To** field, select **Action step variables**.

The screenshot shows the IBM Watsonx Orchestrate interface with the 'z/OS certificate expires soon' skill flow. Step 8 is currently being configured. A modal window titled 'Edit passed values' is open, showing a dropdown menu for 'To'. The 'Action step variables' option is selected and highlighted with a green box. The variable '3. What is your certificate label?' is also highlighted with a green box.

## 18. Click What is your certificate label?

The screenshot shows the IBM Watsonx Orchestrate interface with the 'z/OS certificate expires soon' skill flow. Step 8 is currently being configured. A modal window titled 'Edit passed values' is open, showing a dropdown menu for 'To'. The 'Action step variables' option is selected and highlighted with a green box. The variable '3. What is your certificate label?' is also highlighted with a green box.

## 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.

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 8[1].extra\_vars.cert\_label\_survey To 3. What is your certificate label?

Set 8[4].extra\_vars.new\_expiry\_date\_su... To 7. What date would you like to set the ...

Set new value +

Cancel Apply

Goes to action Cert Renewal skill flow

20. Click **Set new value +** and then select **extra\_vars.sign\_label\_survey**.

z/OS certificate expires soon

Step 8

Is taken with conditions

Conditions

Passed values

Edit passed values

The values you select from this action will be passed to "Cert Renewal skill flow" skill.

Set 8[1].extra\_vars.cert\_label\_survey To 3. What is your certificate label?

Set 8[4].extra\_vars.new\_expiry\_date\_su... To 7. What date would you like to set the ...

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

6. extra\_vars.sign\_label\_survey

7. extra\_vars.sign\_with\_survey

Cancel Apply

Goes to action Cert Renewal skill flow

21. In the **To** option list, select **Enter text**.

The screenshot shows the IBM Watsonx Orchestrator interface with a skill flow editor. A modal window titled "Edit passed values" is open, displaying three "Set" actions mapping variables from step 8 to steps 3, 7, and 6. The "To" field for step 6 is highlighted with a green box, and the "Enter text" option is selected. The "Apply" button at the bottom right of the modal is also highlighted with a blue box.

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.

The screenshot shows the 'Edit passed values' dialog in the IBM Watsonx Orchestrate interface. The dialog lists three variables being passed to the 'Cert Renewal skill flow' skill:

- Set 8t3 1. extra\_vars.cert\_label\_survey To 3. What is your certificate label?
- Set 8t3 4. extra\_vars.new\_expiry\_date\_su... To 7. What date would you like to set the ...
- Set 8t3 6. extra\_vars.sign\_label\_survey To TESTCA

The 'TESTCA' value for the third variable is highlighted with a green box. The 'Apply' button at the bottom right of the dialog is also highlighted with a green box.

## 23. Review the **Edit passed values** and then click **Apply**.

Review all 3 variables are set correctly.

The screenshot shows the 'Edit passed values' dialog in the IBM Watsonx Orchestrate interface. A yellow warning message at the top of the dialog states: 'Variables are not private if they are shared with Skill Studio.' The dialog lists three variables being passed to the 'Cert Renewal skill flow' skill:

- Set 8t3 1. extra\_vars.cert\_label\_survey To 3. What is your certificate label?
- Set 8t3 4. extra\_vars.new\_expiry\_date\_su... To 7. What date would you like to set the ...
- Set 8t3 6. extra\_vars.sign\_label\_survey To TESTCA

The 'TESTCA' value for the third variable is highlighted with a green box. The 'Apply' button at the bottom right of the dialog is also highlighted with a green box.

## 24. Click **Next step +**

**z/OS certificate expires soon**

**Step 8**

Is taken with conditions

Conditions

If All of this is true:

7. What date would y... 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.

Define customer response

**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

Preview

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 various icons. The main area displays a workflow titled 'z/OS certificate expires soon'. The steps are numbered 3 through 9. Step 3 asks 'What is your certificate issuer?' with a 'Free text' input field. Step 4 says 'This step has no content'. Step 5 goes to a subaction 'Retrieve certificate...'. Step 6 asks 'Would you like to renew your certificate?' with a 'Confirmation' button. Step 7 asks 'What date would you like to set the renewed certificate's expiration date to? Please enter in the...' with a 'Free text' input field. Step 8 says 'Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...'. Step 9 is a placeholder step with the text 'This step has no content'. The 'Step 9' configuration panel shows 'Is taken' dropdown options: 'without conditions' (selected), 'with conditions', and 'Set variable values'. Below it is an 'Assistant says' section with a rich text editor toolbar and a placeholder text 'For example: Please select from the following options:'. The 'And then' section contains a 'Continue to next step' button.

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 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 workflow titled 'z/OS certificate expires soon'. The workflow consists of several steps:

- Step 1:** 'Continue to next step' (greyed out)
- Step 2:** '3 is Defined' (greyed out)
- Step 3:** 'This step has no content'
- Step 4:** 'Go to subaction: Retrieve certifica...' (greyed out)
- Step 5:** '3 is Defined' (greyed out)
- Step 6:** 'Below is your certificate's expiration date:  
1. Step 4' (greyed out)
- Step 7:** 'Continue to next step' (greyed out)
- Step 8:** 'Would you like to renew your certificate?' (greyed out)
- Step 9:** 'Confirmation' (greyed out)
- Step 10:** 'What date would you like to set the renewed certificate's expiration date to? Please enter in the...  
Tr Free text' (greyed out)
- Step 11:** 'Continue to next step' (greyed out)
- Step 12:** '7 is Defined' (greyed out)
- Step 13:** 'Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...  
Go to subaction: Cert Renewal ski...' (greyed out)
- Step 14:** '7 is Defined' (highlighted with a green dashed border)
- Step 15:** 'Would you like to verify the new expiration date for your certificate?' (highlighted with a green dashed border)
- Step 16:** 'Continue to next step' (greyed out)

**Step 9 Configuration:**

- Condition:** Is taken with conditions
- Set variable values:** Set variable values (button)
- Conditions:** 1 condition
  - If All of this is true:
  - 7. What date would y... is defined
  - and Add condition +
- New condition group +**

**Assistant says:**

Would you like to verify the new expiration date for your certificate?

**Define customer response:**

And then

Continue to next step

27. Click the **Define customer response** option list and select **Confirmation**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. On the left, a vertical list of steps is displayed:

- z/OS certificate expires soon
- ↓ Continue to next step
- 3 is Defined
- This step has no content
- 4 ↗ Go to subaction: *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 ↗ 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 ↗ Continue to next step
- New step +

On the right, the "Step 9" configuration panel is shown. It includes a "Conditions" section with a condition defined as "7. What date would y... is defined". Below this is a "Confirmation" configuration panel, which is highlighted with a green border. The "Confirmation" section is titled "Confirmation" and describes it as "Enable customers to select from Yes or No response." It includes a "Yes" button and a "No" button.

28. Click **Next step +**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. On the left, a vertical stack of cards represents a workflow:

- Step 3:** "is Defined". Below it: "This step has no content".
- Step 4:** "Go to subaction: Retrieve certifica...". Below it: "Below is your certificate's expiration date: Step 4".
- Step 5:** "is Defined". Below it: "Would you like to renew your certificate? Confirmation".
- Step 6:** "is Yes". Below it: "What date would you like to set the renewed certificate's expiration date to? Please enter in the... Free text".
- Step 7:** "is Defined". Below it: "Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an... Go to subaction: Cert Renewal ski...".
- Step 9:** "is Defined". Below it: "Would you like to verify the new expiration date for your certificate? Confirmation".

On the right, the "Editor" tab is active, showing the configuration for **Step 9**:

- Is taken:** "with conditions".
- Conditions:** "If All of this is true:"
- Condition 1:** "7. What date would y... is defined".
- And:** "Add condition +"
- New condition group +**
- Assistant says:** "Would you like to verify the new expiration date for your certificate?"
- Response options:** "Yes" (selected), "No".
- View response** and **Edit validation** buttons.
- And then:** "Continue to next step".
- Buttons at the bottom:** "New step +" (highlighted in green), "Preview", and "Next".

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 area displays a workflow titled 'z/OS certificate expires soon'. The steps are numbered 3 through 10. Step 3: '3 is Defined' with a note 'Below is your certificate's expiration date:' followed by '15 Step 4' and a 'Continue to next step' button. Step 4: '3 is Defined' with a note 'Would you like to renew your certificate?' followed by a 'Confirmation' button and a 'Continue to next step' button. Step 5: '3 is Defined' with a note 'What date would you like to set the renewed certificate's expiration date to? Please enter in the...' followed by a 'Tr Free text' button and a 'Continue to next step' button. Step 6: '6 is Yes' with a note 'Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...' and a 'Confirmation' button. Step 7: '7 is Defined' with a note 'Would you like to verify the new expiration date for your certificate?' and a 'Confirmation' button. Step 8: '7 is Defined' with a note 'This step has no content' and a 'Continue to next step' button. Step 9: '7 is Defined' with a note 'This step has no content' and a 'Confirmation' button. Step 10: '7 is Defined' with a note 'This step has no content' and a 'Continue to next step' button.

**Step 10**

Is taken without conditions  without conditions  with conditions

Set variable values

Assistant says

For example: What type of transfer would you like to make?

Define customer response

And then

↓ Continue to next step

30. Click the **And then** option list and select **Go to a subaction**.

z/OS certificate expires soon

Step 10

Is taken with conditions

Conditions

If All 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

Continue to next step

Go to a subaction

Switch conversation flow to another action to perform a certain task

Preview

31. Click the (a) Go to option list, select (b) Retrieve certificate expiration, and then click (c) Apply.

z/OS certificate expires soon

Step 10

Is taken with conditions

Conditions

If All of this is true:

9. Would you like to v... is Yes

and Add condition +

Settings

Go to a subaction

Go to

Select an action...

Cert Renewal skill flow

Fallback

Retrieve certificate expiration

Cancel

Apply

And then

32. Click Edit passed values.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. The flowchart starts with step 3 (z/OS certificate expires soon) leading to step 4 (Step 4). Step 4 leads to step 5 (Would you like to renew your certificate?). If the answer is Yes (step 6), it leads to step 7 (What date would you like to set the renewed certificate's expiration date to? Please enter in the...). Step 7 leads to step 8 (Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...). Step 8 leads to step 9 (Would you like to verify the new expiration date for your certificate?). Step 9 leads to step 10 (This step has no content). Step 10 leads back to step 4 (Step 4).

**Step 10:**

Is taken with conditions

Conditions: If All 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:

Go to a 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.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface with the 'Edit passed values' dialog open. The dialog title is 'Edit variable values' and the sub-section is 'Response behavior'. It states: 'The values you select from this action will be passed to "Retrieve certificate expiration" skill.' A button labeled 'Set new value +' is highlighted with a green box. Below it, a list of variables is shown, with '1. extra\_vars.cert\_label\_survey' also highlighted with a green box.

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.

Set new value +

< All variables

1. extra\_vars.cert\_label\_survey

2. extra\_vars.cert\_type\_survey

3. extra\_vars.owner\_id\_survey

Cancel Apply

Goes to action: Retrieve certificate expiration

34. In the To option list, click Action step variables.

The screenshot shows the IBM Watsonx Orchestrate interface with a workflow titled 'z/OS certificate expires soon'. Step 10 is currently selected. The 'Edit variable values' dialog is open, showing a warning about shared variables. The 'To' dropdown menu is open, and the option 'Action step variables' is highlighted with a green box.

35. Click What is your certificate label?.

The screenshot shows the same workflow and dialog as the previous step, but the 'To' dropdown menu now highlights 'Action step variables'.

36. Review the Edit variable values and click Apply.

**Step 10**

Is taken with conditions

Conditions

Passed values

Edit passed values

Configurable 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.

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?

Set new value +

Cancel Apply

Goes to action Retrieve certificate expiration

### 37. Click Next step +.

**Step 10**

Is taken with conditions

Conditions

If All 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

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 +

Preview ▶

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 interface with the 'AI assistant builder' tab selected. The main area displays a workflow with several steps numbered 3 through 11. Step 11 is currently being edited. The configuration pane for Step 11 includes the following fields:

- Is taken:** A dropdown menu is open, showing three options: "without conditions" (selected), "without conditions", and "with conditions". The "with conditions" option is highlighted with a green box.
- Assistant says:** A rich text editor field containing placeholder text: "For example: What type of transfer would you like to make?". Below it is a "Define customer response" dropdown.
- And then:** A dropdown menu showing the option "Continue to next step".

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. On the left, a vertical stack of cards represents a workflow:

- Card 1: z/OS certificate expires soon
- Card 2: Continue to next step
- Card 3: 3 is Defined
- Card 4: Would you like to renew your certificate?
- Card 5: Confirmation
- Card 6: 6 is Yes
- Card 7: What date would you like to set the renewed certificate's expiration date to? Please enter in the... (with a 'Tr Free text' button)
- Card 8: Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...
- Card 9: 7 is Defined
- Card 10: 8 Go to subaction: Cert Renewal ski...
- Card 11: 9 is Yes
- Card 12: This step has no content
- Card 13: 10 Go to subaction: Retrieve certific...
- Card 14: 11 Below is the new expiration date of your renewed certificate:

On the right, the "Editor" tab is active, showing the configuration for Step 11:

- Step 11:** Is taken with conditions
- Conditions:** If All of this is true:
  - 9. Would you like to v... is Yes
- Assistant says:** Below is the new expiration date of your renewed certificate:
- Define customer response:** \$
- And then:** Continue to next step

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. On the left, a vertical stack of cards represents a workflow:

- Step 3:** "z/OS certificate expires soon" (Defined)
- Step 6:** "Would you like to renew your certificate?" (Confirmation)
- Step 7:** "What date would you like to set the renewed certificate's expiration date to? Please enter in the..." (Free text)
- Step 8:** "Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an..." (Confirmation)
- Step 9:** "Would you like to verify the new expiration date for your certificate?" (Confirmation)
- Step 10:** "This step has no content" (Go to subaction: Retrieve certificate expiration)
- Step 11:** "Below is the new expiration date of your renewed certificate: <br />" (Confirmation)

**Step 11 Editor View:**

**Condition:** Is taken with conditions

**Conditions:** 1 condition

If All of this is true:

- 9. Would you like to v... is Yes
- and Add condition +

**Assistant says:**

Below is the new expiration date of your renewed certificate:  
\$

Action step variables >

Retrieve certificate expiration (step 4) >

Cert Renewal skill flow (step 8) >

Retrieve certificate expiration (step 10) >

Assistant variables >

Integration variables >

#### 41. Click **Retrieve certificate expiration (step 10)**.



**Be sure to select the output from step 10 and not step 4.**

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 flowchart titled 'z/OS certificate expires soon'.

**Flow Details:**

- Step 1:** 'Would you like to renew your certificate?' (Confirmation)
- Step 2:** 'What date would you like to set the renewed certificate's expiration date to? Please enter in the...' (Free text)
- Step 3:** 'Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an...' (Confirmation)
- Step 4:** 'This step has no content' (Free text)
- Step 5:** 'Below is the new expiration date of your renewed certificate: <br />' (Text)

**Step 11 Configuration:**

- Condition:** 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:

Action step variables >  
 Retrieve certificate expiration (step 4) >  
 Cert Renewal skill flow (step 8) >  
**Retrieve certificate expiration (step 10) >** (highlighted)  
 Assistant variables >  
 Integration variables >

#### 42. Click **Retrieve certificate expiration result variable**.

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 workflow titled 'z/OS certificate expires soon'.

**Step 11:** A condition is defined: "Is taken with conditions". The condition details are as follows:

- If: All of this is true:
  - 9. Would you like to v... is Yes
- and: Add condition +

**Assistant says:** Below is the new expiration date of your renewed certificate:

**And then:**

1. Retrieve certificate expiration result var... (highlighted with a green box)

2. Define customer response

**Step 11 (continued):**

9 is Yes

Below is the new expiration date of your renewed certificate: <br />

↓ Continue to next step

43. Click the **And then** option list and select **End the action**.

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder interface. On the left, a vertical stack of cards represents a workflow:

- Step 1:** z/OS certificate expires soon → Continue to next step
- Step 2:** 3 is Defined → Would you like to renew your certificate? (Confirmation)
- Step 3:** 6 is Yes → What date would you like to set the renewed certificate's expiration date to? Please enter in the... (Free text)
- Step 4:** 7 is Defined → Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an... (Confirmation)
- Step 5:** 8 is Yes → Would you like to verify the new expiration date for your certificate? (Confirmation)
- Step 6:** 9 is Yes → This step has no content (Confirmation)
- Step 7:** 10 is Yes → Below is the new expiration date of your renewed certificate: 11. Retrieve certificate expiration n... (Confirmation)
- Step 8:** 11 is Yes → Below is the new expiration date of your renewed certificate: 11. Step 10 (Confirmation)

On the right, the **Editor** tab is active, showing the configuration for **Step 11**:

- Condition:** Is taken with conditions
- Conditions:** 1 condition
  - If All of this is true:
    - 9. Would you like to v... is Yes
  - and Add condition +
- Assistant says:**
  - Below is the new expiration date of your renewed certificate: 11.10. Retrieve certificate expiration n...
- Define customer response:**
- And then:**
  - Continue to next step (highlighted with a green box)
  - End the action

44. Review the (a) final step, click (b) Save (), and then click (c) x.

**z/OS certificate expires soon**

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  
T Free text

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  
10 This step has no content  
Go to subaction: Retrieve certific...

11 9 is Yes  
Below is the new expiration date of your renewed certificate: 11 Step 10  
Action complete

**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

B I O fx D C S E 10. Retrieve certificate expiration date

Below is the new expiration date of your renewed certificate:  
11 10. Retrieve certificate expiration date

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, Preview, 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. There are six rows listed:

Name	Last edited	Examples count	Steps count	Status
<a href="#">z/OS IPL Information</a>	a day ago	0	3	Green checkmark
<a href="#">Zeeves-gather-facts-flow</a>	3 days ago	1	0	Green checkmark
<a href="#">Retrieve IPL information</a>	a day ago	1	3	Green checkmark
<a href="#">Retrieve certificate expiration</a>	4 hours ago	0	3	Green checkmark
<a href="#">Cert Renewal skill flow</a>	4 hours ago	0	7	Green checkmark
<a href="#">z/OS certificate expires soon</a>	a few seconds ago	1	11	Green checkmark

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 'Preview' view of the AI assistant. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and icons for help and A3. Below the navigation is a toolbar with 'Copy link to share', 'Change layout' (which is highlighted with a blue border), 'Change background', and 'Customize web chat'. The main area is titled 'Preview assistant' and shows 'AssistantZeeves'. In the center, there's a blue owl icon with the text '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?' followed by a green arrow button.

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. On the left is a sidebar with various icons. The main area has a header "Preview assistant" and "AssistantZeeves". A message from the user says "How do I check the expiration date for my certificate that's expiring soon?". The AI's response is: "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." Below this, there are two buttons: "Yes" and "No". At the bottom, there is a text input field with "Type something..." and a send button. The footer says "Built with IBM Watsonx".

4. Enter DEMOCERT after the assistant responds with **What is your certificate label?**

**Prompt:**

DEMOCERT

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 help and settings. The main area is titled 'Preview assistant' and shows a conversation with 'AssistantZeeves'. The conversation history is as follows:

- You 12:37 PM: How do I check the expiration date for my certificate that's expiring soon?
- AssistantZeeves 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?
- AssistantZeeves 12:37 PM: Yes
- You 12:40 PM: What is your certificate label?
- AssistantZeeves 12:40 PM: DEMOCERT ▶

The bottom left corner of the interface indicates 'Built with IBM Watsonx'.

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

The screenshot shows the IBM Watsonx Orchestrate interface. On the left is a sidebar with various icons. The main area has tabs for 'Preview assistant' and 'AI assistant builder'. The 'AI assistant builder' tab is active, showing a conversation with 'AssistantZeeves'. The user asks: 'How do I check the expiration date for my certificate that's expiring soon?'. AssistantZeeves responds: '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.' Below this, a note says: 'Accuracy of generated answers may vary.' The user then asks: 'Would you like to run the skill to retrieve your certificate's expiration date?'. AssistantZeeves asks: 'What is your certificate label?'. The user replies: 'DEMOCERT'. At the bottom of the conversation, there is a screenshot of a skill configuration window titled 'Retrieve certificate expiration'. It has an icon of a gear and a magnifying glass. The 'id \*' field is set to '21'. At the bottom are 'Cancel' and 'Apply' buttons, with 'Apply' being highlighted in green.

#### 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 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 shows a conversation with "AssistantZeeves". The conversation starts with the AI asking if the user wants to run a skill to retrieve their certificate's expiration date. The user replies "Yes". The AI then asks for the certificate label, and the user replies "DEMOCERT". A modal window titled "Retrieve certificate expiration" shows the input field "id \*" with the value "26". The AI then provides the certificate's expiration date: "Below is your certificate's expiration date: {\"certificate\_expiration\_date\": \"2025/01/30 23:59:59\"}.". Finally, the AI asks if the user would like to renew their certificate, with "Yes" highlighted in a green box.

7. Enter a date in the future in the format **YYYY-MM-DD**.

The screenshot shows a conversation in the 'Preview assistant' tab of the IBM Watsonx Assistant builder. The AI assistant, 'AssistantZeeves', asks for the certificate label ('What is your certificate label?') and receives the response 'DEMOCERT'. It then retrieves the certificate expiration ('Retrieve certificate expiration') and finds it to be '2025/01/30 23:59:59'. It asks if the user would like to renew the certificate, with 'Yes' and 'No' buttons. After a 30-second delay, it asks for the new expiration date ('What date would you like to set the renewed certificate's expiration date to? Please enter in the form of YYYY-MM-DD.') and receives the input '2025-06-30'.

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."

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder 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 shows a conversation with "AssistantZeeves".

**AssistantZeeves:**

- 1:05 PM: Below is your certificate's expiration date:  
{"certificate\_expiration\_date": "2025/01/30 23:59:59"}
- 1:05 PM: Would you like to renew your certificate?  
Yes No
- You 1:05 PM: Yes
- 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
- 1:05 PM: Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.
- 1:05 PM: Would you like to verify the new expiration date for your certificate?  
Yes No

At the bottom, there is a text input field with placeholder text "Type something..." and a send button icon.

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

The screenshot shows the IBM Watsonx Orchestrate AI assistant builder 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 shows a conversation with "AssistantZeeves".

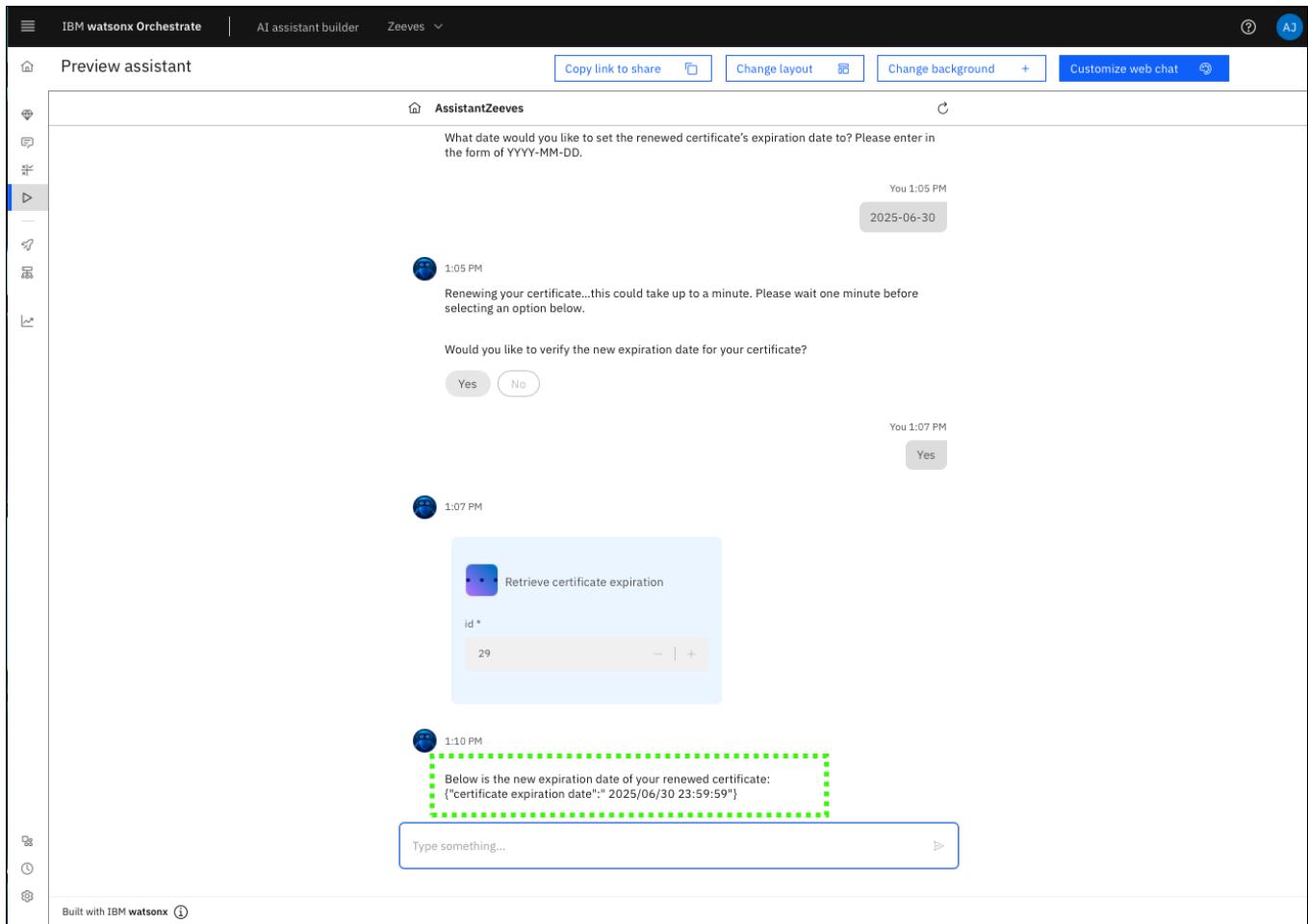
The conversation log:

- 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"
- AssistantZeeves 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?"  
Buttons: Yes (disabled), No (disabled)
- You 1:07 PM: "Yes"
- AssistantZeeves 1:07 PM: A modal dialog titled "Retrieve certificate expiration" is displayed. It contains a text input field with the value "29" and a date range selector with the value "29". The "Apply" button is highlighted with a green border.

At the bottom left, there is a sidebar with various icons. At the bottom right, it says "Built with IBM Watsonx".

## 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

Configure prebuilt skills

**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

## 2. Create a connection for the app.



Use Skills catalog to create the app.

IBMW Watsonx Orchestrate

English ⓘ A3

Skill catalog /

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.
--	--

## 3. Create a skill flow to get the output of the **TSO Command**.

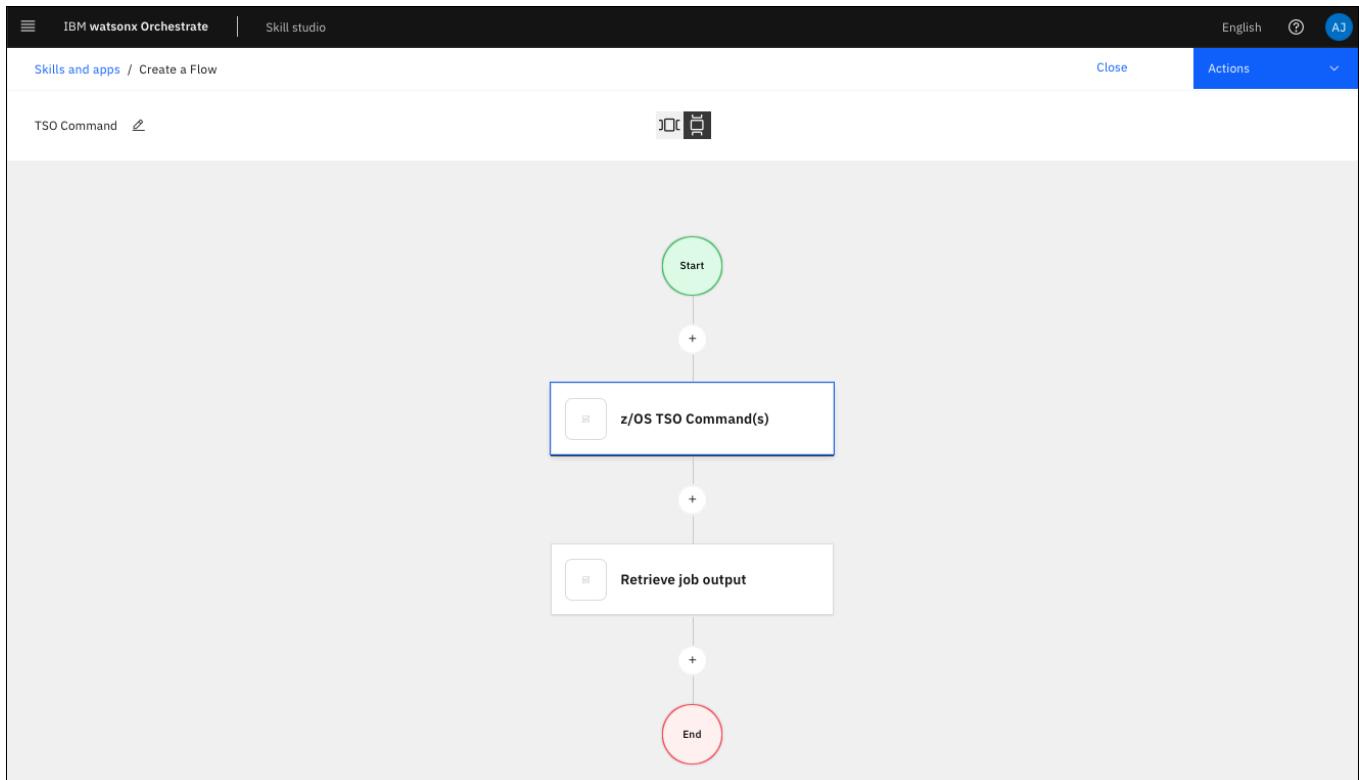


Use Skills studio to create the skill flow.

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
TSO Command	a few seconds ago	0	1	Green checkmark
z/OS certificate expires soon	3 hours ago	1	11	Green checkmark
Cert Renewal skill flow	7 hours ago	0	7	Green checkmark
Retrieve certificate expiration	7 hours ago	0	3	Green checkmark
Retrieve IPL information	a day ago	1	3	Green checkmark
z/OS IPL Information	a day ago	0	3	Green checkmark
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:

The screenshot shows the IBM watsonx Orchestrate interface. On the left, there's a sidebar with 'RACF assistance' and 'Conversation steps'. One step is highlighted with a green dashed border. Inside this step, under 'Assistant says', the text 'What is your RACF User ID?' is displayed. Below it, under 'And then', is the action 'Continue to next step'. The top right of the interface has tabs for 'Editor' and 'Visualization', along with various icons.



### 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 screenshot shows the AI assistant builder interface. The main area displays a conversation flow with steps 1 through 5. Step 1 is a free-text prompt asking for the RACF User ID. Step 2 is currently being configured, with the condition dropdown set to "without conditions". An "Edit passed values" dialog is open over the configuration area, containing the expression editor with the text "'LIST USER ' + 1. What is your RACF User ID?'. The "Apply" button in this dialog is highlighted in blue.

The step should look like the following image:

The screenshot shows the AI assistant builder interface with the completed configuration for Step 2. A green dashed box highlights the "Assistant says" section, which contains the expression "Checking user privileges for 1. What is your RACF User ID?". Another green dashed box highlights the "And then" section, where "Go to a subaction" is selected, leading to the "TSO Command" action. The "Pass values" and "Upon return" fields are also visible in this section.

**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 Orchestrator AI assistant builder interface. On the left, the "Conversation steps" pane displays four steps:

- Customer starts with: RACF assistance
- What is your RACF User ID? (Free text)
- Checking user privileges for Step 1 (Go to subaction: TSO Command)
- Do you need RACF assistance? (Confirmation)

Step 4 is currently being configured. The "Is taken with conditions" section contains a condition:

If All of this is true:

- 3. Do you need RACF ... is Yes

The "Assistant says" section contains the response template:

What type of assistance?

Changing my RACF password Privileges issues

Buttons for "Edit response" and "Edit validation" are visible.

The step should look like the following image:

This screenshot is identical to the one above, but the entire configuration for Step 4 is highlighted with a green dashed border. This visual cue indicates that the user should focus on modifying the condition and response for this specific step.

**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:

The screenshot shows the IBM Watsonx Assistant interface for building AI assistants. The top navigation bar includes 'IBM Watsonx Orchestrate', 'AI assistant builder', 'Zeeves', and various icons. The main workspace displays a flow titled 'RACF assistance' with six steps:

- Step 3:** 'Do you need RACF assistance?' with a 'Confirmation' button.
- Step 4:** 'What type of assistance?' with options 'Changing my ...' and 'Privileges iss...', both with 'Continue to next step' buttons.
- Step 5:** 'This step has no content' with a 'Continue to next step' button.
- Step 6 (highlighted with a green dashed border):** 'Would you like to change your user ID's RACF passphrase?' with a 'Confirmation' button. This step is configured with a condition: 'If All of this is true: 4. What type of assist... is Changing my RACF password'. Below it, the 'Assistant says' section contains the message 'Would you like to change your user ID's RACF passphrase?' with 'Yes' and 'No' buttons.
- Step 7:** 'And then' with a 'Continue to next step' button.

**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 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 screenshot shows the IBM Watsonx Orchestrator interface with the 'RACF assistance' skill selected. A modal window titled 'Edit passed values' is open, containing an 'Expression editor' with the following code:

```
'ALTUSER ' + 1. What is your RACF User ID? + ' PHRASE (' 7. Please enter your new RACF passphrase ') NOEXPIRE RESUME'
```

The step should look like the following image:

The screenshot shows the IBM Watsonx Orchestrator interface with the 'RACF assistance' skill selected. A green dashed box highlights the configuration for Step 8. The configuration includes:

- Condition:** Is taken with conditions
- Action:** Go to subaction: 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:

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' (selected), 'Build', 'Actions', 'Evaluate', 'Preview', 'Publish', 'Environments', 'Improve', and 'Analyze'. The main area shows a 'Build actions' card with 'Enhance and improve your assistant's actions.' Below it is a diagram titled 'Architecture' showing the flow from 'Zeeves' through 'Actions' to 'Default behavior', 'Fallback', and 'Search'. The 'Integrations' section is highlighted with a green border. It contains 'Activity log' and 'Assistant settings'.

## 2. Explore the **Essential channels** and **Channels** sections.

The screenshot shows the 'Integrations' section with a 'Web chat' card. Below it are sections for 'Essential channels' (with a note about utilizing methods like 'Zeeves'), 'Channels' (with a note about deploying to third-party channels), and 'Extensions' (listing 'SMS', 'Facebook messenger', 'Genesys Bot Connector', 'Slack', 'Microsoft teams', and 'WhatsApp with Twilio'). A green dashed box highlights the 'Essential channels' and 'Channels' sections.

## 3. Click **Add** on the **Slack** tile.

**Integrations**  
Add different channels and extensions to easily configure and deploy your assistant.

**Essential channels**  
Add our most utilized methods of deploying assistants. These channels support additional customization and advanced integrations.

**Web chat**  
Built by IBM Lite  
Embed the web chat onto your company website so it can answer questions.  
[Learn more](#)

**Contact center integrations**  
NICE CX ONE ZEN

**Channels**  
Deploy your assistant to third-party channels to expand your reach.

<b>SMS</b>	<b>Facebook messenger</b>	<b>Genesys Bot Connector</b>	<b>Slack</b>	<b>Microsoft teams</b>	<b>WhatsApp with Twilio</b>
<a href="#">Add</a>	<a href="#">Add</a>	<a href="#">Add</a>	<a href="#">Add</a>	<a href="#">Add</a>	<a href="#">Add</a>

**Extensions**

#### 4. Click Add.

**Integrations**  
Add different channels and extensions to easily configure and deploy your assistant.

**Web chat**

**Slack**  
Let your assistant respond directly in Slack. Add your assistant as a bot user to your Slack app, answering questions from direct messages or channels.

**Add**

**Channels**  
Deploy your assistant to third-party channels to expand your reach.

<b>SMS</b>	<b>Facebook messenger</b>	<b>Genesys Bot Connector</b>	<b>Slack</b>	<b>Microsoft teams</b>	<b>WhatsApp with Twilio</b>
<a href="#">Add</a>	<a href="#">Add</a>	<a href="#">Add</a>	<a href="#">Add</a>	<a href="#">Add</a>	<a href="#">Add</a>

#### 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.

**Slack** [Draft](#)

**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](#)

**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

**Close** **Next**

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](#)