

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 (andrewj@us.ibm.com) and Max Weiss (maxwell.g.weiss@ibm.com)
Copyright	Copyright © 2024 IBM - V 2.0

Table of Contents

Welcome

- Prerequisite knowledge
- Lab overview
- 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.

Prerequisite knowledge

The prerequisite requirements to complete this course are as follows:

- Basic Windows or Linux (MacOS) Operating System admin skills, including:
 - Open a command prompt or power shell on Windows, or a terminal window on Linux (MacOS)
 - Change directories, create directories, and move directories
 - Set environment variables and add directories to the default PATH
 - Create and modify plain text files with an editor and save as plain text files
 - Familiarity with a visual code editor like VS Studio Code is a plus
- The following links may help with these basic skills. This is not an endorsement of the sources. Try searching the internet for related resources.
 - [Windows tutorials](#) (and other videos in the series).
 - [Windows PowerShell](#).
 - [MacOS terminal](#).
 - [The 50 most popular Linux & terminal commands - full course for beginners](#) (based on the following book).
 - [The Linux Commands Handbook](#) (Linux/MacOS).

**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.

Lab overview

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 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 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

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 and 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
```

```
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  **Esc** key or click the **X** to dismiss the enlarged image.

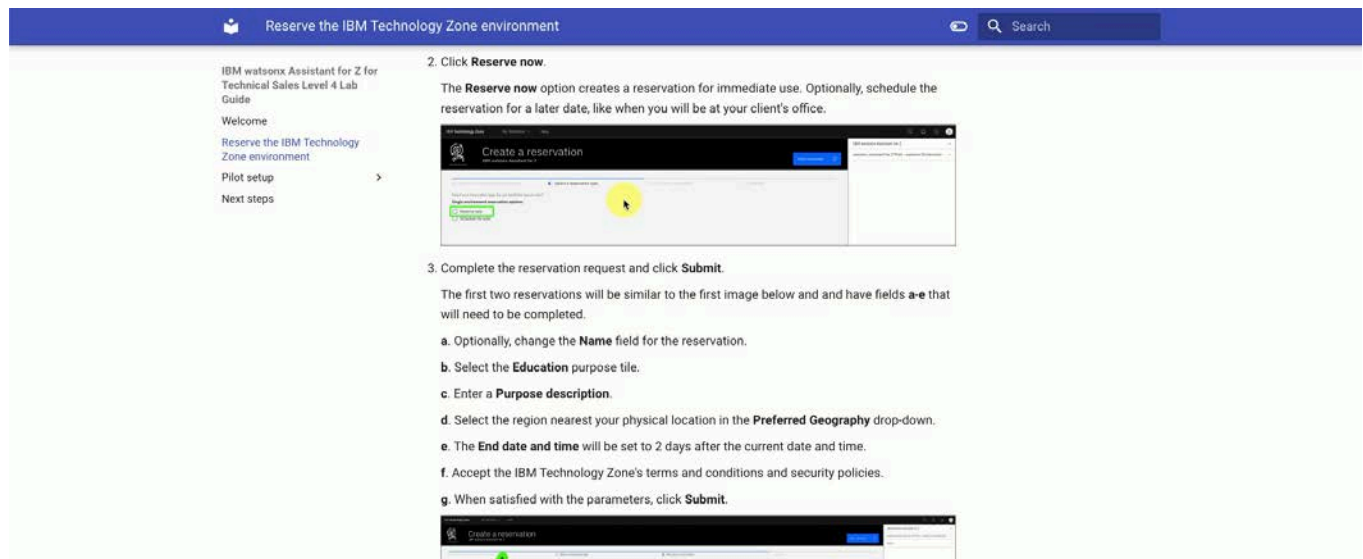
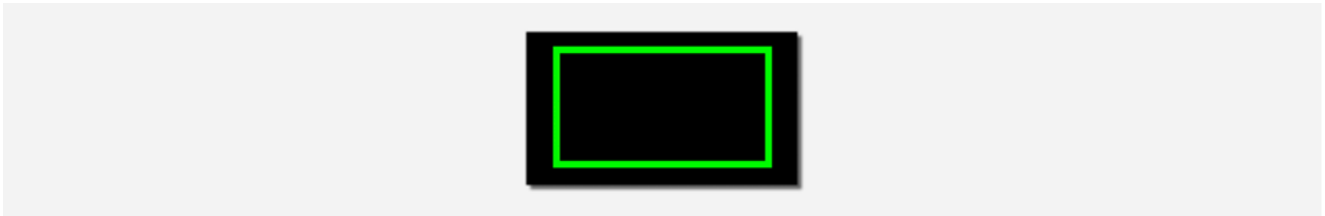


Image highlighting

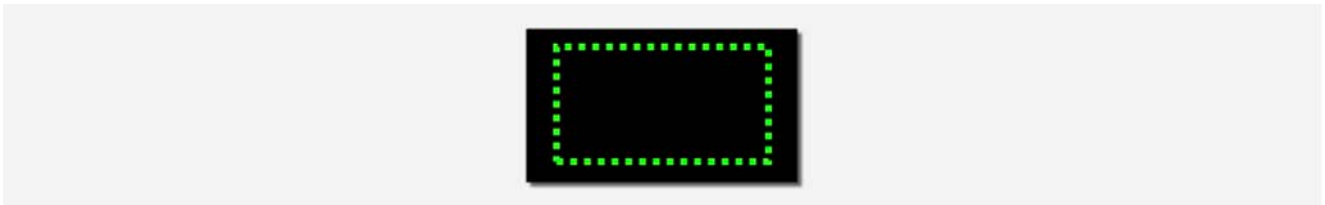


In some images, the following styles of highlighting are used:

- **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, `^ Ctrl1` + `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.

The screenshot shows the 'Guidance for delivering a demonstration' page. The main content area has a heading 'Specific guidance for IBM watsonx Assistant for Z and the ITZ environment'. Below this, it states: '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:'. A bulleted list follows: '• Follow the scripts in the IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration Guide for the automations and skills to execute as expected.' A tooltip is visible over the acronym 'LPAR' in the text, showing the text 'IBM Technology Zone'.



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:

IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration Guide

Search

- IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration Guide
- 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
 - Instructions

Welcome

Welcome to the **IBM watsonx Assistant for Z for Technical Sales Level 3** course and demonstration guide! The goal of this course is to provide IBM and Business Partner technical sellers the knowledge and tools to demonstrate **IBM watsonx Assistant for Z**.

This course is not intended to serve as an in-depth study of IBM watsonx Assistant for Z, watsonx, or IBM Z; rather the focus is on learning how to deliver a client demonstration of IBM watsonx Assistant for Z. More specifically, the focus is on the *end-user* creating queries and by using trusted automations through skills that are suggested by IBM watsonx Assistant for Z.

Other aspects of IBM watsonx Assistant for Z including creating and configuring assistants, importing and creating new skills, and ingestion of client documentation are **not** covered in this demonstration guide. To learn more and get hands-on with these additional capabilities, check out the Velocity Pilot information in IBM Technology Zone (ITZ) [here](#).

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 (☰):

IBM watsonx Assistant for Z for Technical Sales Level 3 Demonstration...

Search

Welcome

Welcome to the **IBM watsonx Assistant for Z for Technical Sales Level 3** course and demonstration guide! The goal of this course is to provide IBM and Business Partner technical sellers the knowledge and tools to demonstrate **IBM watsonx Assistant for Z**.

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

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

IBM Technology Zone environment

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

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

 **All activities in this lab guide are required.**

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

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

Create a reservation request

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

 **You may be asked to authenticate to IBM Technology Zone.**

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

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

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

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



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

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

2. Select **Reserve now**.

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

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

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

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
watsonx Assistant for Z Pilot - watsonx Orchestrator

Purpose
☐ Demo: Explore a short 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: Rapid on-remote build that proves IBM technologies can deliver business value to client's 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.
☐ Education: Gaining experience with specific technology, product, or solution.
☐ Test: Research test a specific function, configuration, or implementation.

Project Opportunity number
Enter an opportunity number.

Provisioning on
[IBM Sales Cloud prospecting tool](#) | [Sales Navigator](#) | [LinkedIn Sales Navigator](#) | [Other](#)
 Provisioning on [IBM Sales Cloud prospecting tool](#) will allow you to enter your reservation here.

Level 4 learning
Level 4 learning

Preferred Organization
IBM watsonx-2 - AMERICAS - us-south region - data0 datacenter

End date and time
 Select a date: 12/02/2024
 Select a time: 8:07 AM
 America/ENGLAND

Submit

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

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

IBM Technology Zone

My TechZone

Help

Select a date

11/01/2024

Select a time

11:51 AM

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.

OCPI/Kubernetes Cluster Network

10.128.0.0/14

Enable FIPS Security

No

Master Single Node Flavor

16 vCPU x 64 GB - 300 GB ephemeral storage

OpenShift Version

4.14

OCPI/Kubernetes Service Network

172.30.0.0/16

Notes

Enter any notes you would like to attach to this reservation.

IBM watsonx Assistant for Z

Single Node OpenShift (VMware on IBM Cloud) - NFS

Policy

m

n

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

Submit

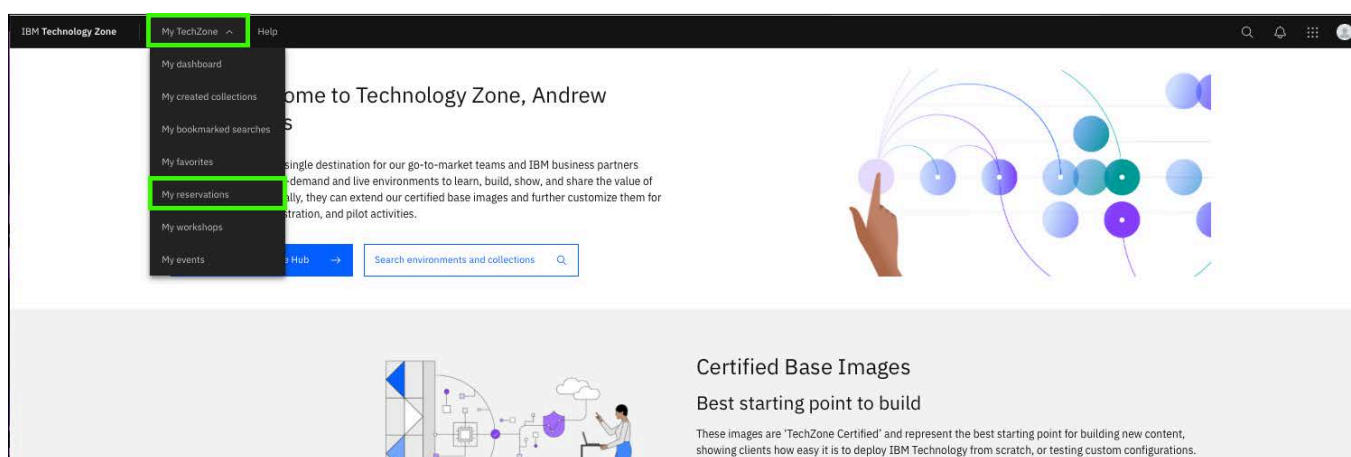
During the provisioning process, multiple emails are sent to you from ITZ as the provisioning process runs. One email states the reservation is provisioning and the other email states that the environment is **Ready**.

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.

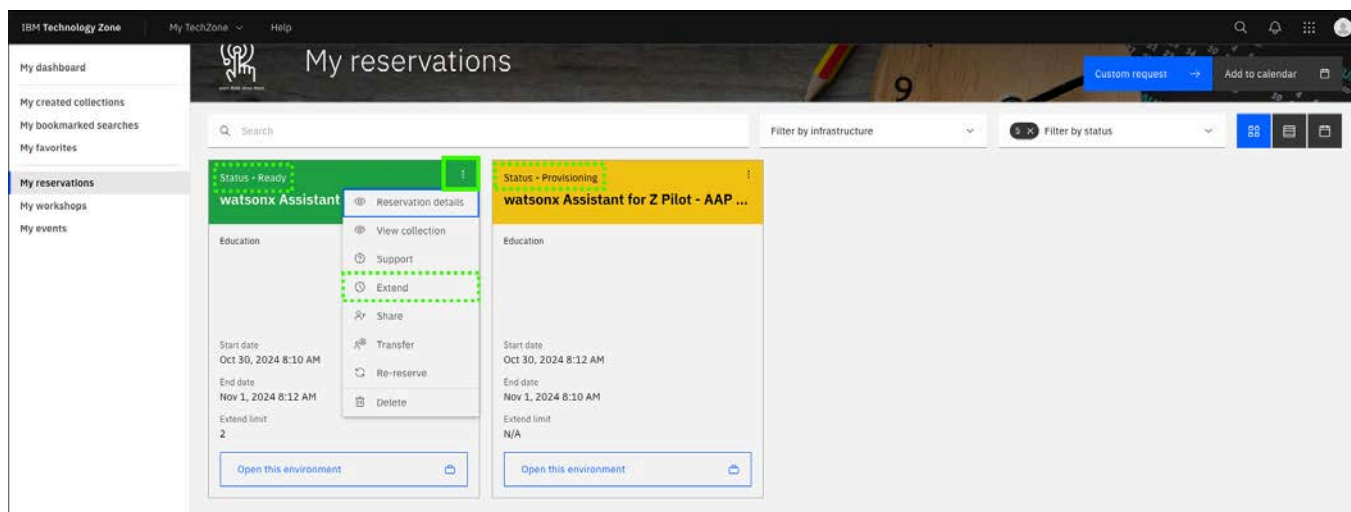
Extend the reservation

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.

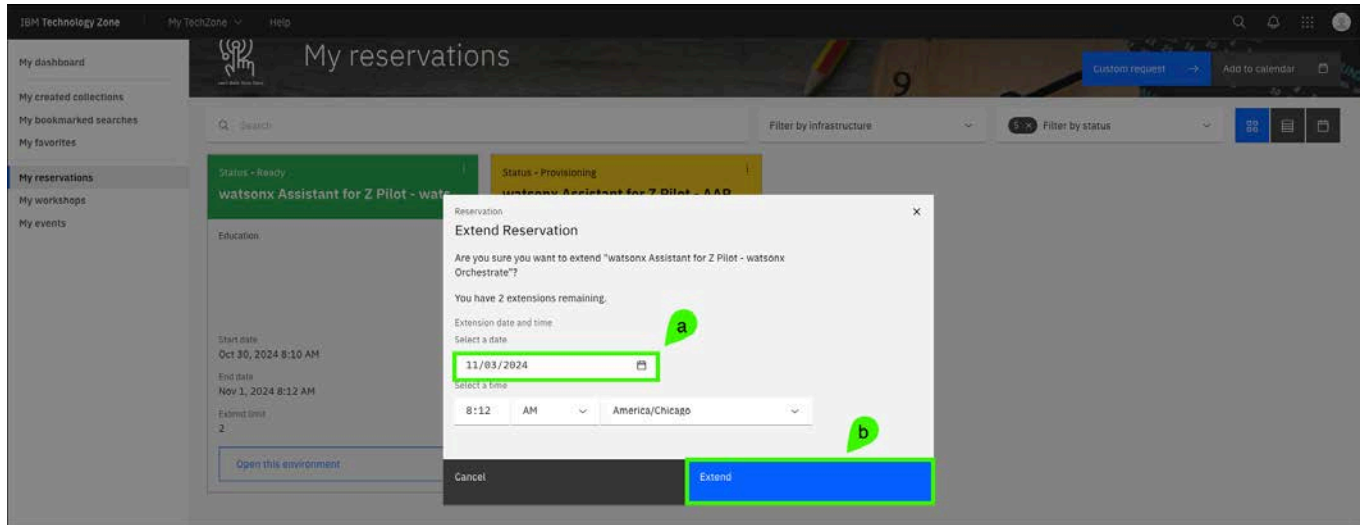
1. In the IBM Technology Zone portal, expand **My TechZone** and select **My Reservations**.



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



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



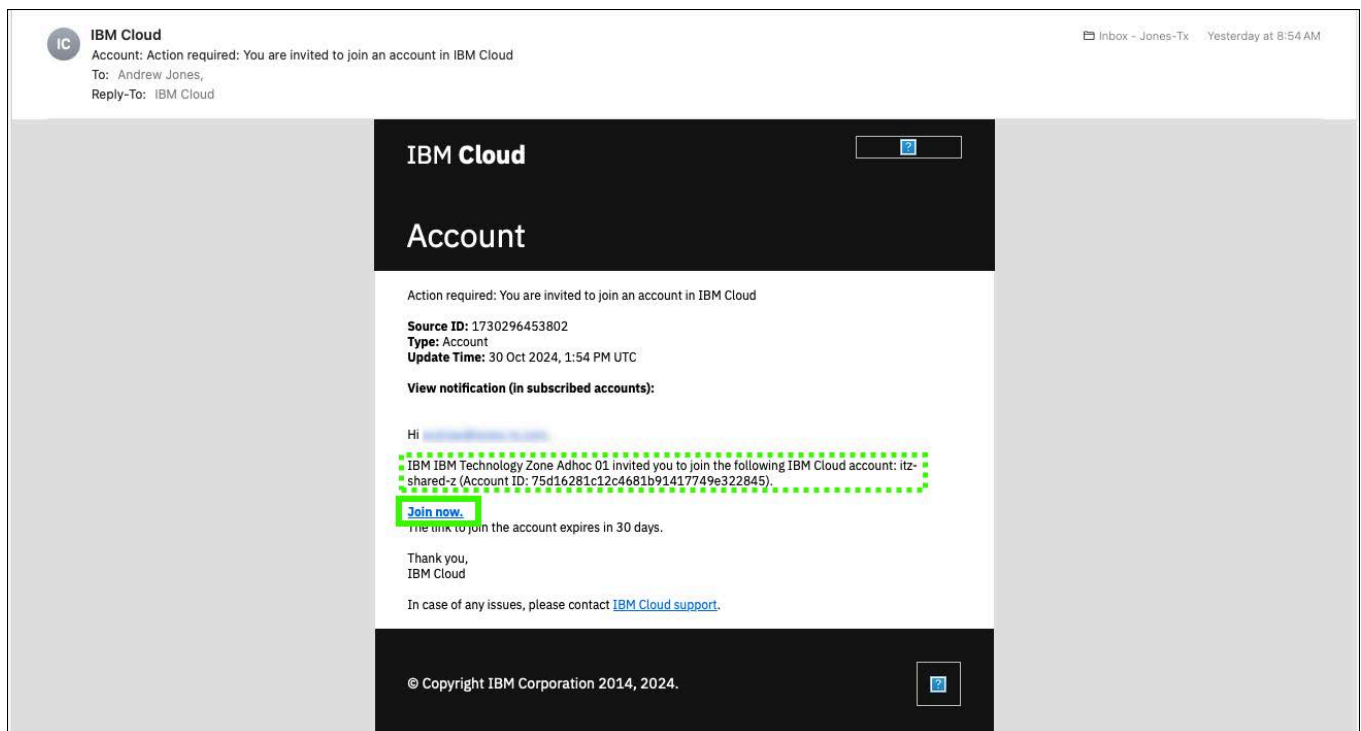
If you anticipate needing more time, repeat Steps 5-6 to extend the reservation to the maximum allowed. Repeat these steps for the other two reservations.

Join the ITZ IBM Cloud account

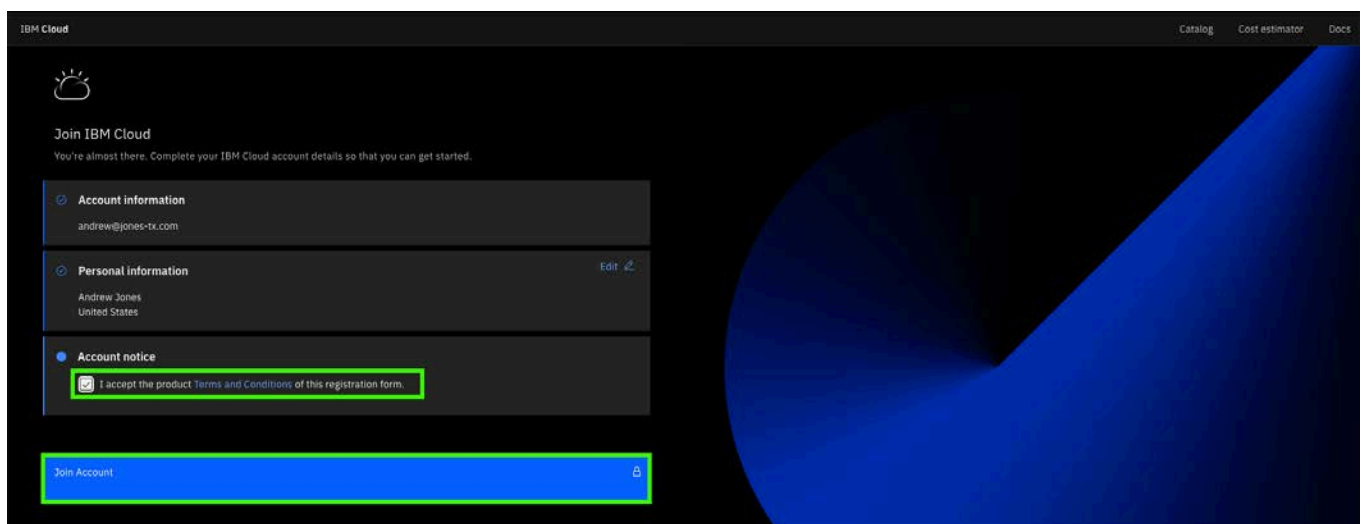
Both the **watsonx Assistant for Z lab – watsonx Orchestrate** and the **Ansible Automation Platform (AAP) & z/OS** environments add you to an IBM Cloud account while your reservation is active. During the provisioning process of these ITZ environments, you receive two emails from IBM Cloud.

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**.

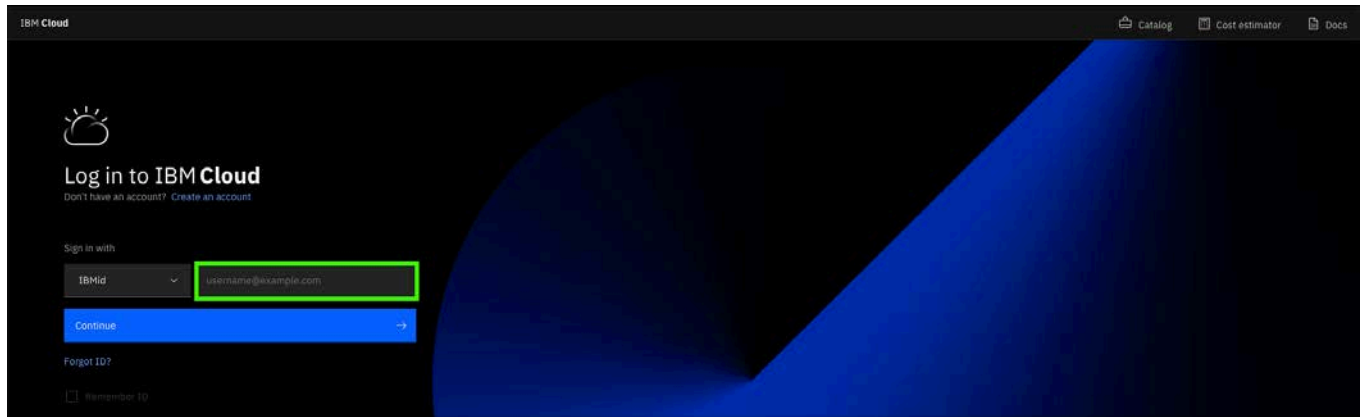


After joining the account, verify that the account appears in your available account list in the IBM Cloud portal.

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

IBM Cloud portal

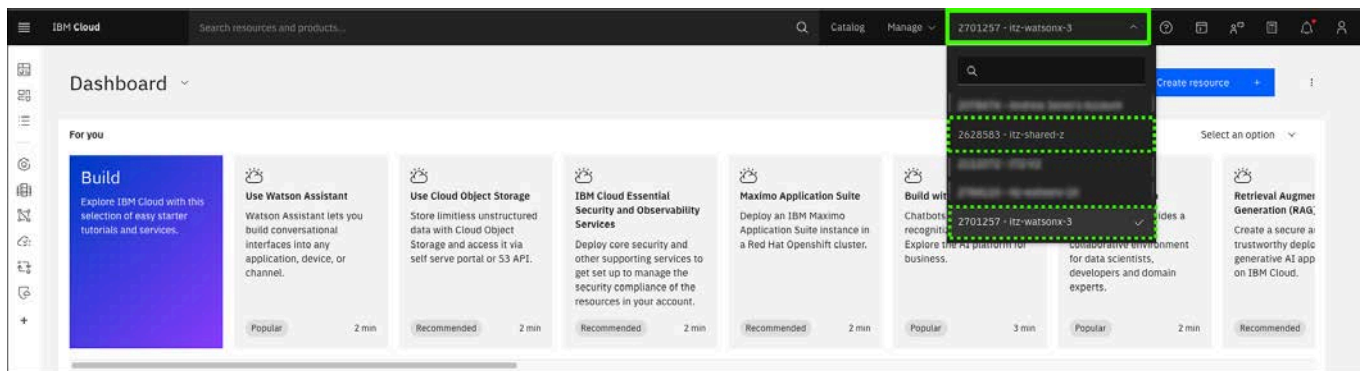
- Follow the directions to complete the authentication to IBM Cloud using the same email address you used to login to ITZ. The login steps vary depending on any two-factor authentication methods enabled.



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

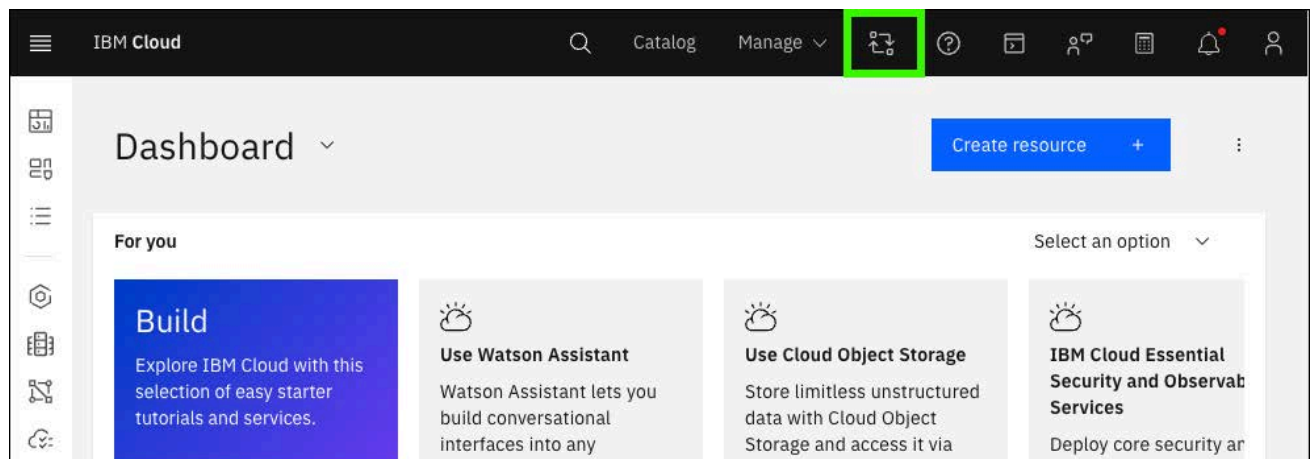
 **The account may be different.**

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

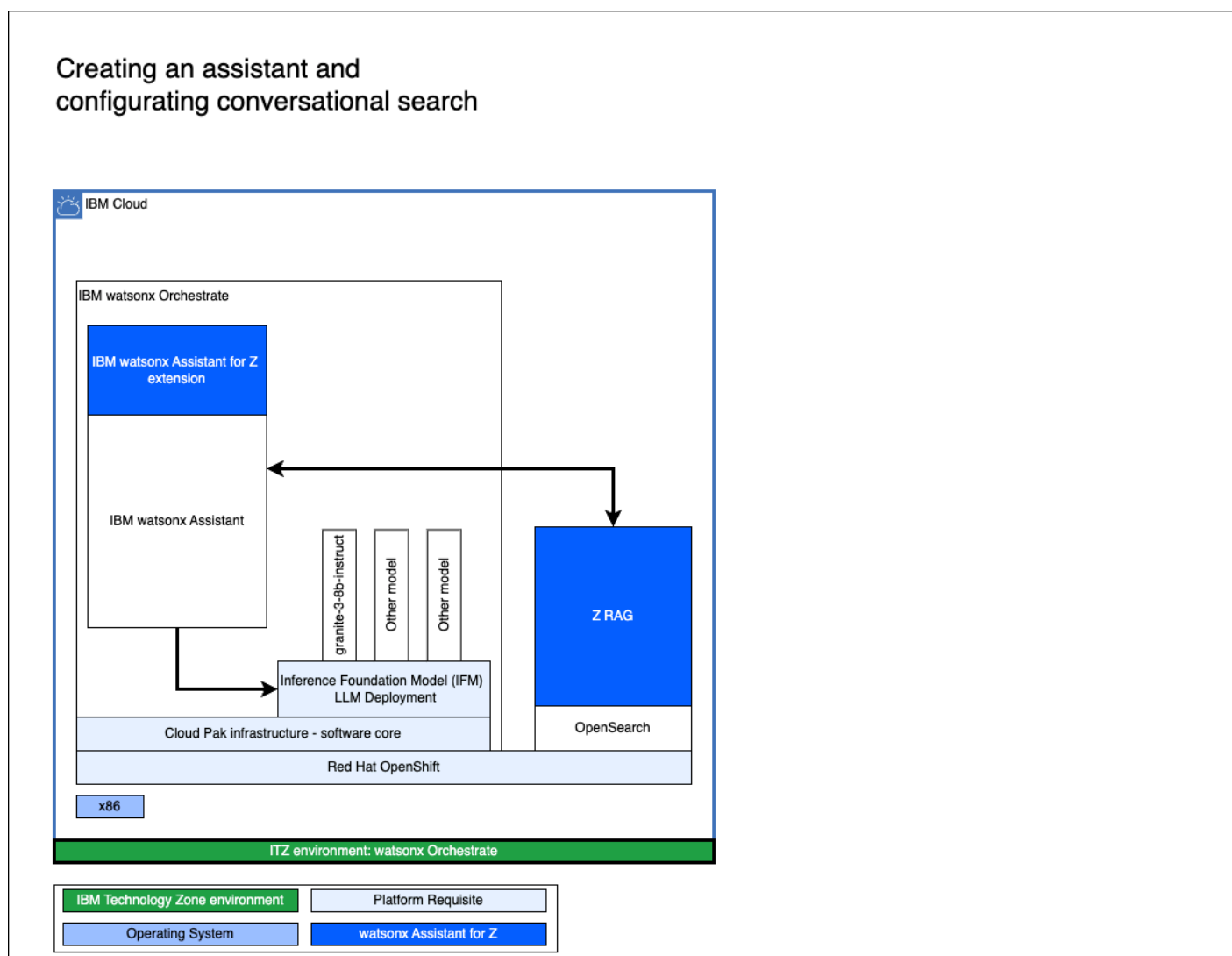
If your reservation for the Single Node OpenShift environment fails...

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.

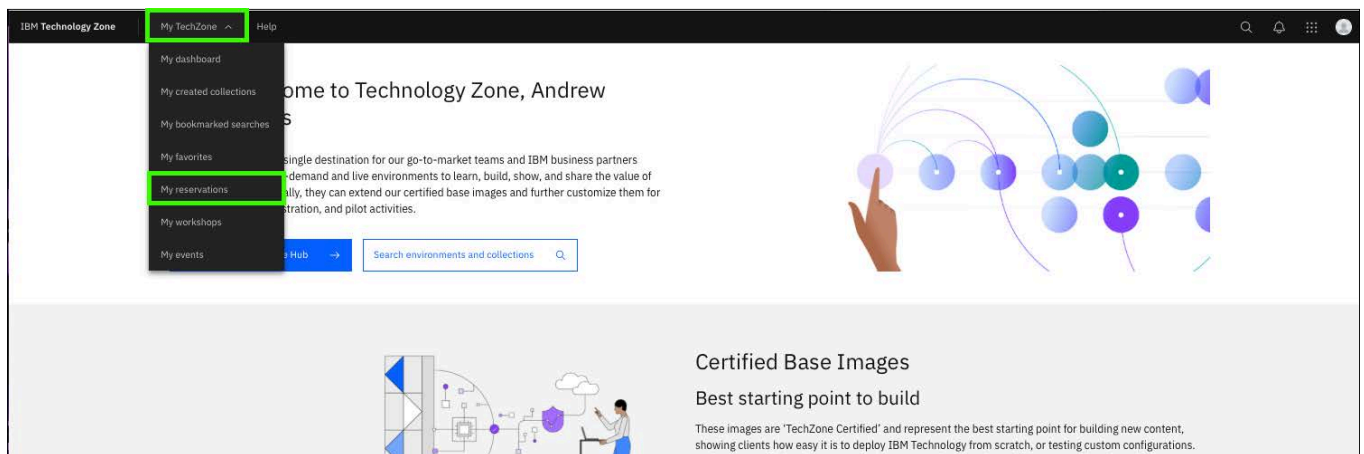
A high-level, logical architecture of the environment is illustrated in the following diagram.



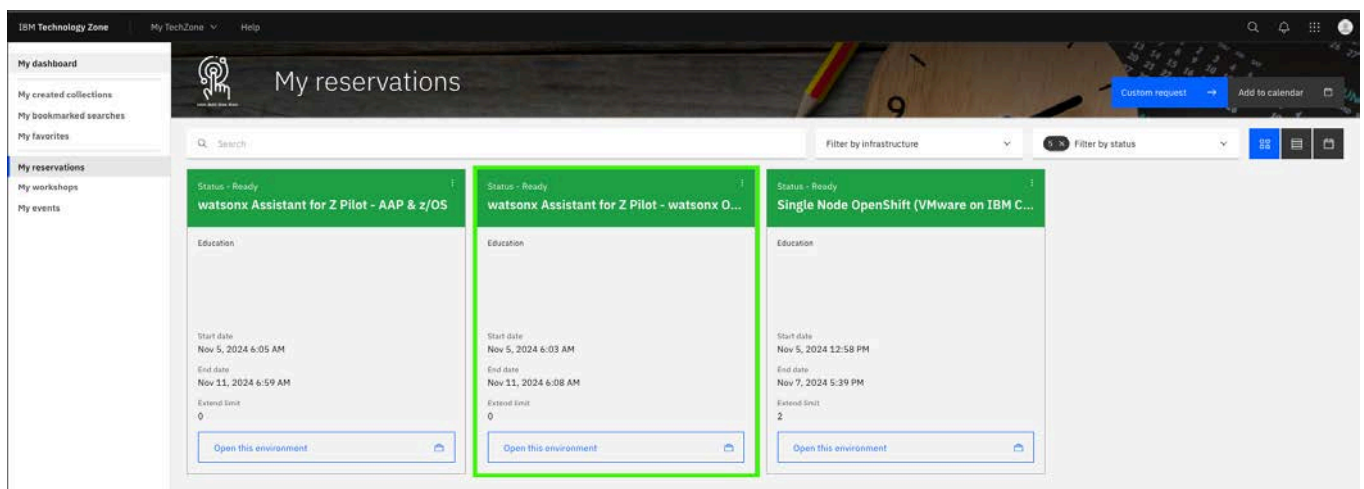
Access the ITZ IBM Cloud account for the watsonx Assistant for Z Pilot environment

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

[ITZ My reservations](#)

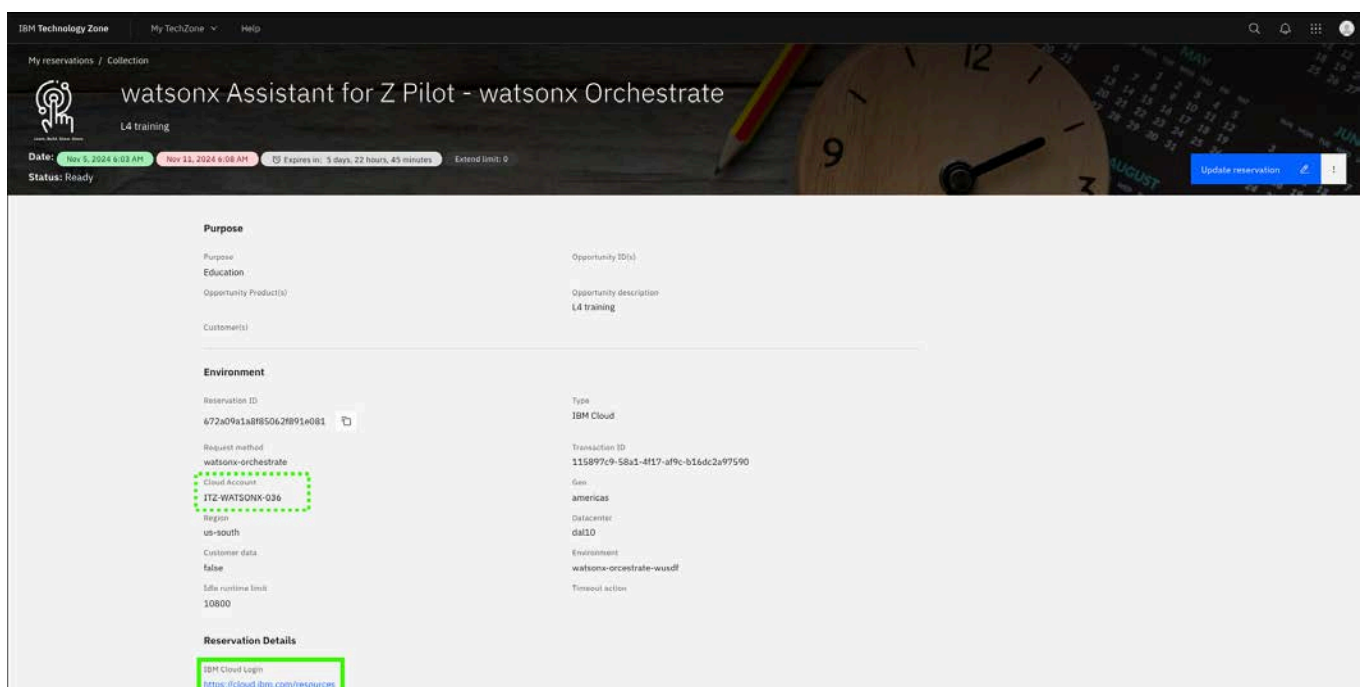


2. Click the **watsonx Assistant for Z Pilot - watsonx Orchestrate** tile.



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: 672a07a1a8f85062f891e081

Type: IBM Cloud

Request method: watsonx-orchestrate

Transaction ID: 115897c9-98a1-4f17-a19c-b16dc2a97590

Cloud Account: ITZ-WATSONX-036

Geo: americas

Region: uk-south

Datacenter: dal10

Customer data: false

Environment: watsonx-orchestrate-wusdf

Idle runtime limit: 10800

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.

IBM Cloud

Search resources and products...

Account: 2953593 - itz-watsonx-036

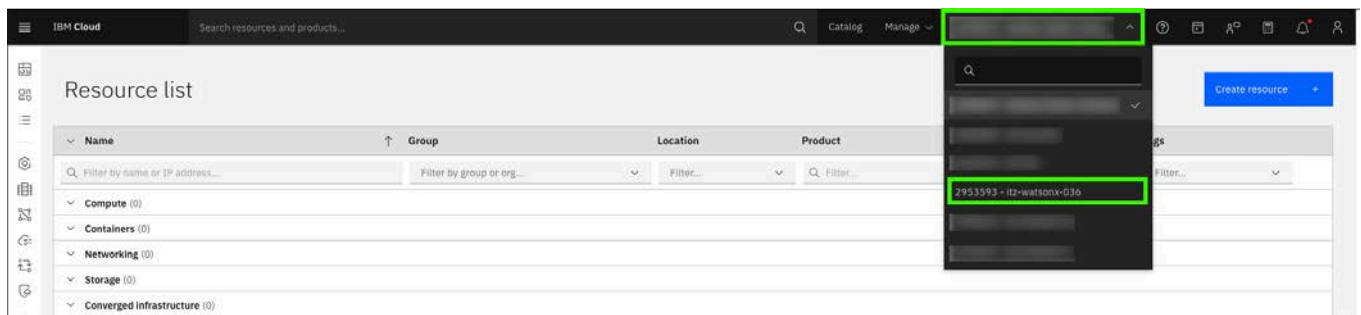
Resource list

Name	Group	Location	Product	Status	Tags
Filter by name or IP address...					
Filter by group or org...					
Filter...					
Filter...					
Filter...					

Categories: Compute (0), Containers (0), Networking (0), Storage (0), Converged infrastructure (0), Enterprise applications (0), AI / Machine Learning (1+), Analytics (0).

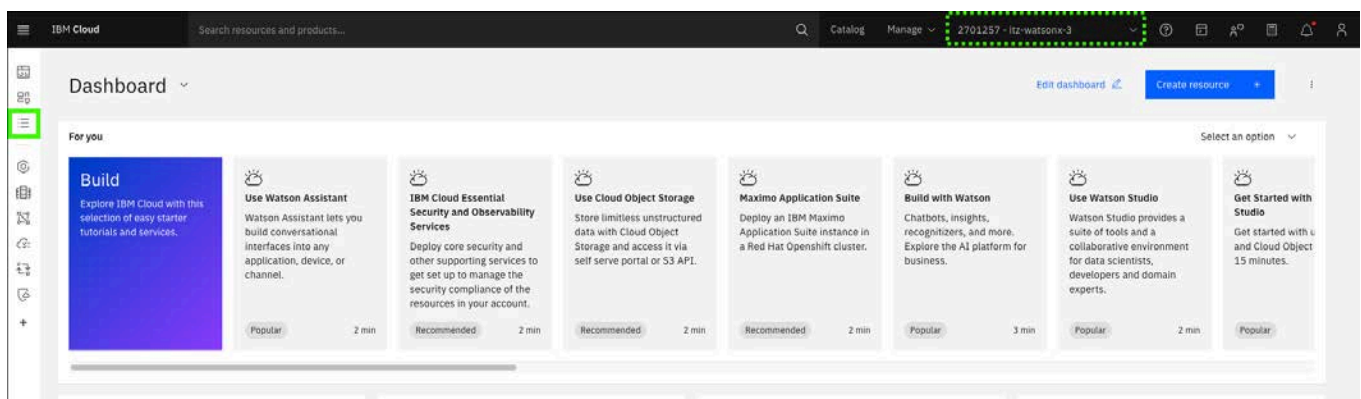
If the proper account is not listed, click the account drop down and select the proper account.

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

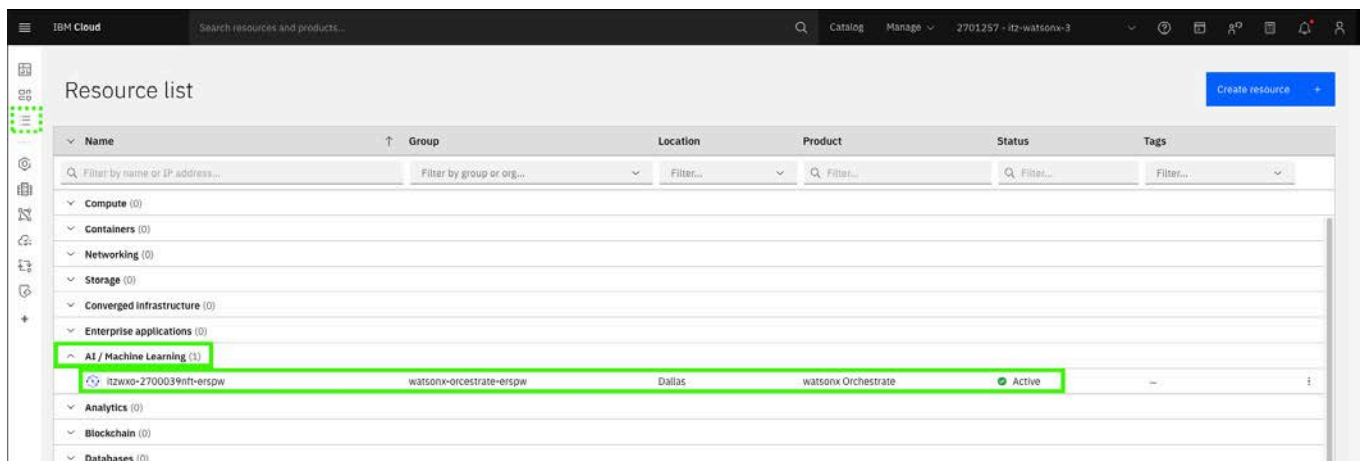


Create your Assistant

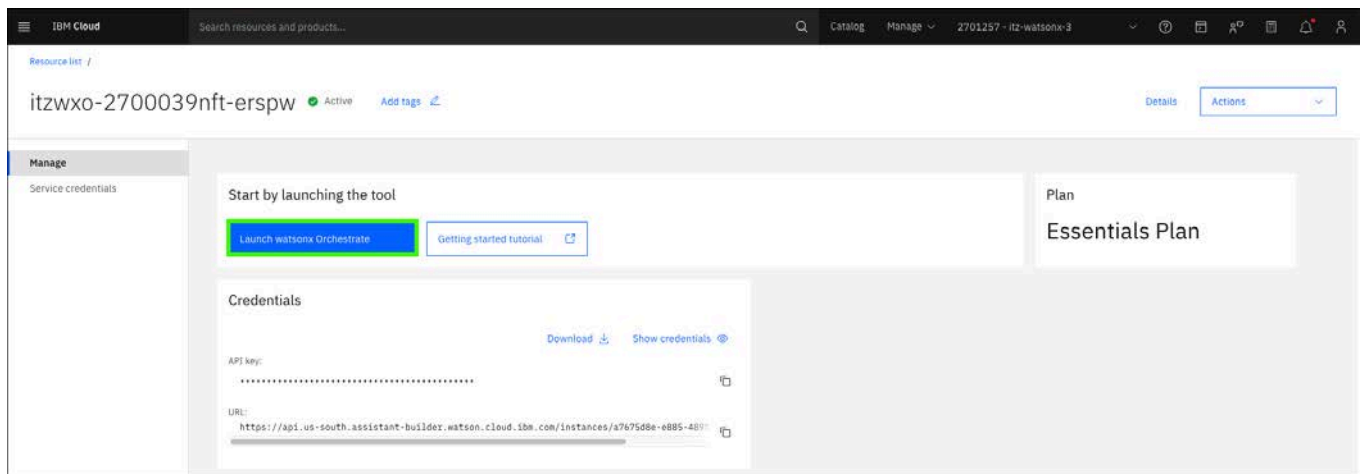
1. Click the **Resources** icon (☰).



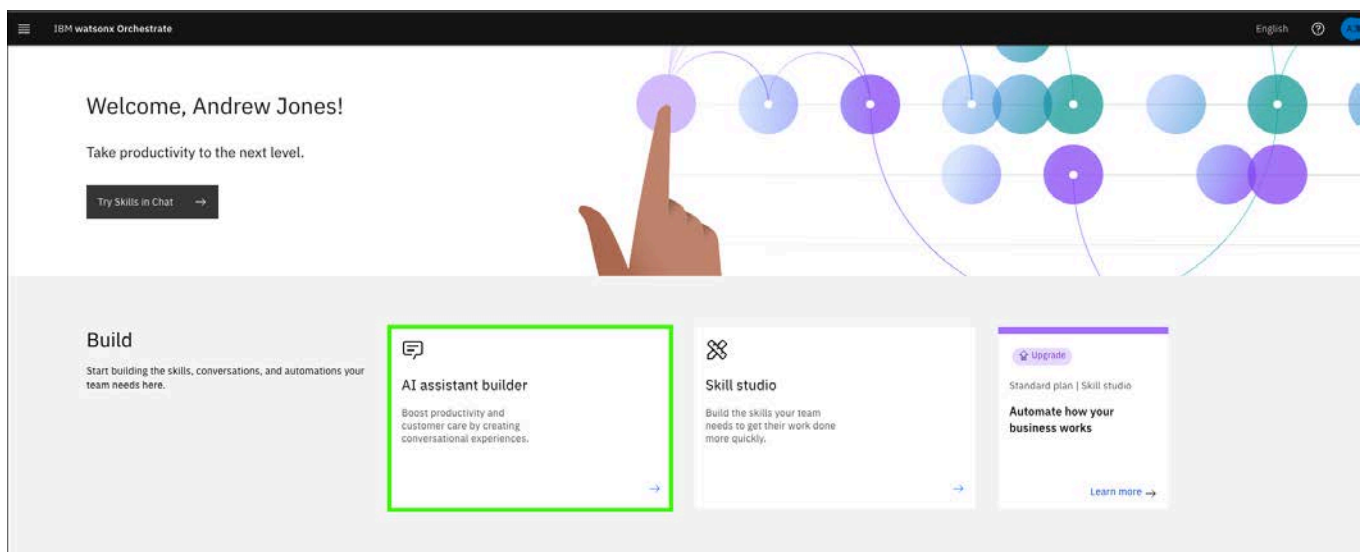
2. Expand the **AI / Machine Learning** section and click the **watsonx Orchestrate** instance listed (the instance name is different than shown in the following image).



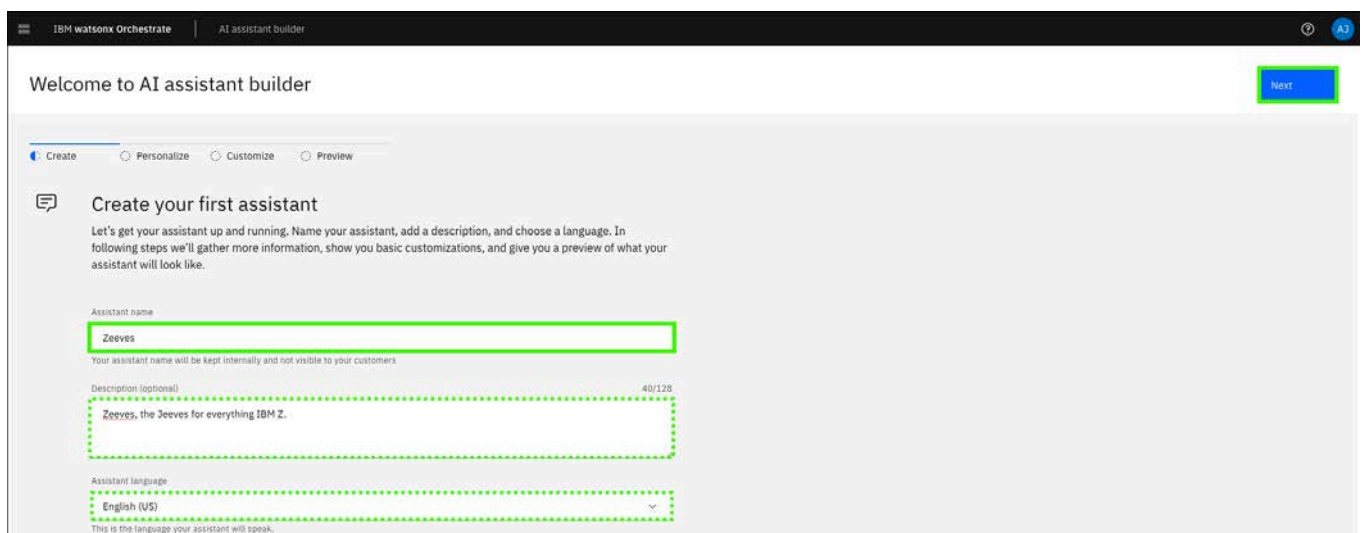
3. Click **Launch watsonx Orchestrate**.



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



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

The screenshot shows the 'Personalize your assistant' step in the IBM watsonx Assistant builder. The interface includes a navigation bar with 'Create', 'Personalize', 'Customize', and 'Preview' tabs. The 'Personalize' tab is active. The main content area is titled 'Personalize your assistant' and contains four sections:

- Tell us where your assistant will live:** A dropdown menu labeled 'Web' is selected. A green circle 'a' is next to it.
- Tell us about yourself:** A dropdown menu labeled 'Software' is selected. A green circle 'b' is next to it.
- What is your role on the team building the assistant?:** A dropdown menu labeled 'Developer' is selected. A green circle 'c' is next to it.
- Which statement describes your needs best?:** A dropdown menu labeled 'I want to automate common tasks in a natural way' is selected. A green circle 'd' is next to it.

On the right side, there is a preview window titled 'This is what your customers will experience' showing a chat interface with a 'watsonx Assistant' header and a conversation about 'Speed Demons'.

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

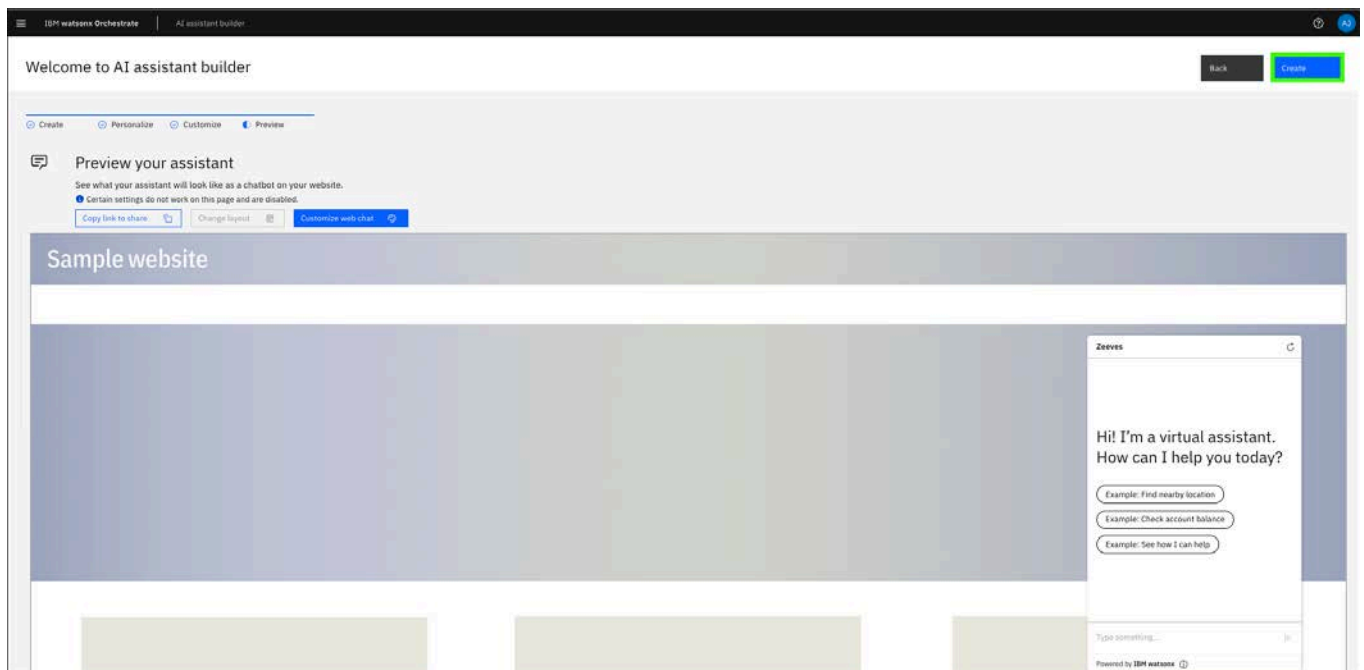
Explore the customization options. When creating an assistant for a client pilot, consider specifying attributes that align with the client (for example, colors and logos).

The screenshot shows the 'Customize your chat UI' step in the IBM watsonx Assistant builder. The interface includes a navigation bar with 'Create', 'Personalize', 'Customize', and 'Preview' tabs. The 'Customize' tab is active. The main content area is titled 'Customize your chat UI' and contains several sections:

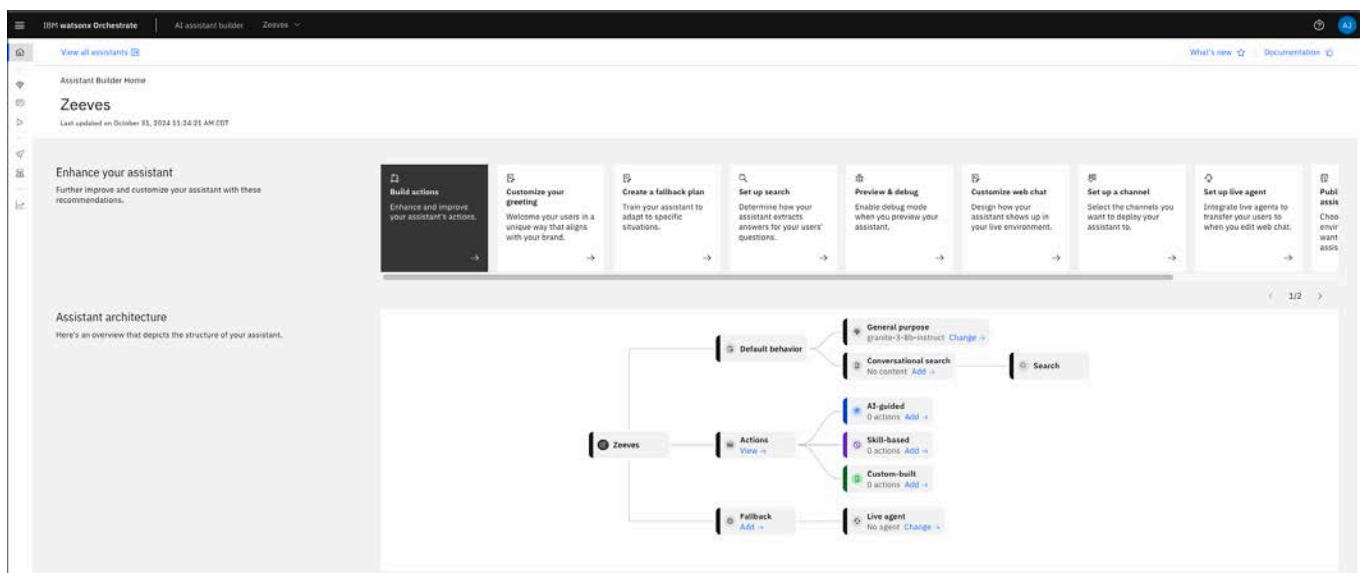
- Assistant's name as known by customers:** A text input field labeled 'Zeeves' is shown. A green dashed box highlights it.
- Intended purpose:** Two radio buttons are shown: 'Standard: For virtual agents and customer support experiences.' (selected) and 'Carbon for AI: For use in internal IBM products.'
- Choose a theme:** Two buttons are shown: 'Light' (selected) and 'Dark'.
- Primary color:** A color picker is shown with the value '#FFFFFF'.
- Secondary color:** A color picker is shown with the value '#303030'.
- Chat header:** A text input field labeled 'User message bubble' is shown.
- Assistant color:** A color picker is shown with the value '#00429F'.
- Size:** A text input field labeled 'The size of the web chat on this page will not change by updating these fields.' is shown.
- IBM Watermark:** A checkbox labeled 'IBM Watermark' is shown, with a green dashed box highlighting it.
- Streaming:** A checkbox labeled 'Streaming' is shown, with a green dashed box highlighting it.

On the right side, there is a preview window titled 'Zeeves' showing a chat interface with a 'Hi! I'm a virtual assistant. How can I help you today?' message and three example buttons: 'Example: Find nearby location', 'Example: Check account balance', and 'Example: See how I can help'.

- Preview your assistant and then click **Create**.



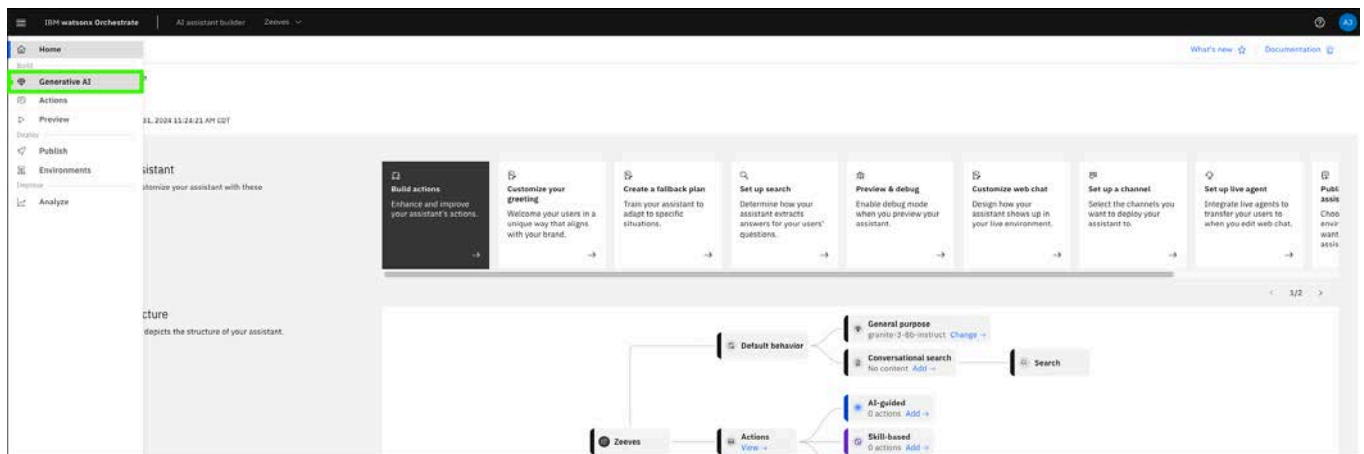
The assistant is now created.



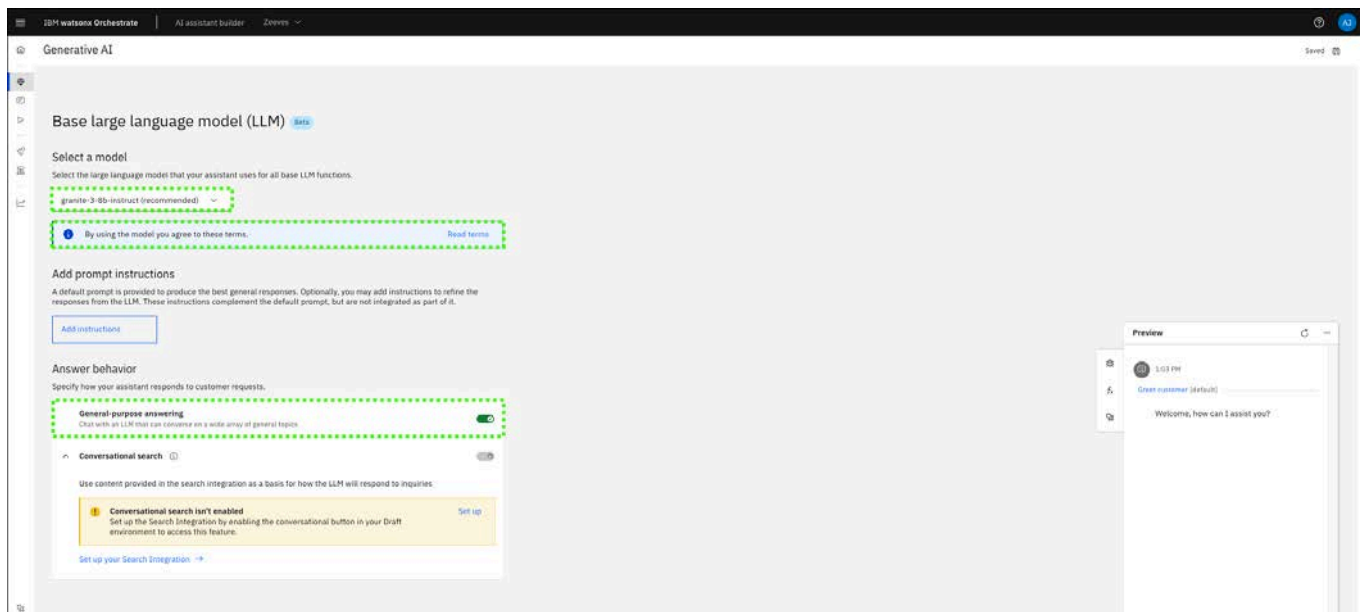
Configure conversational search

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

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

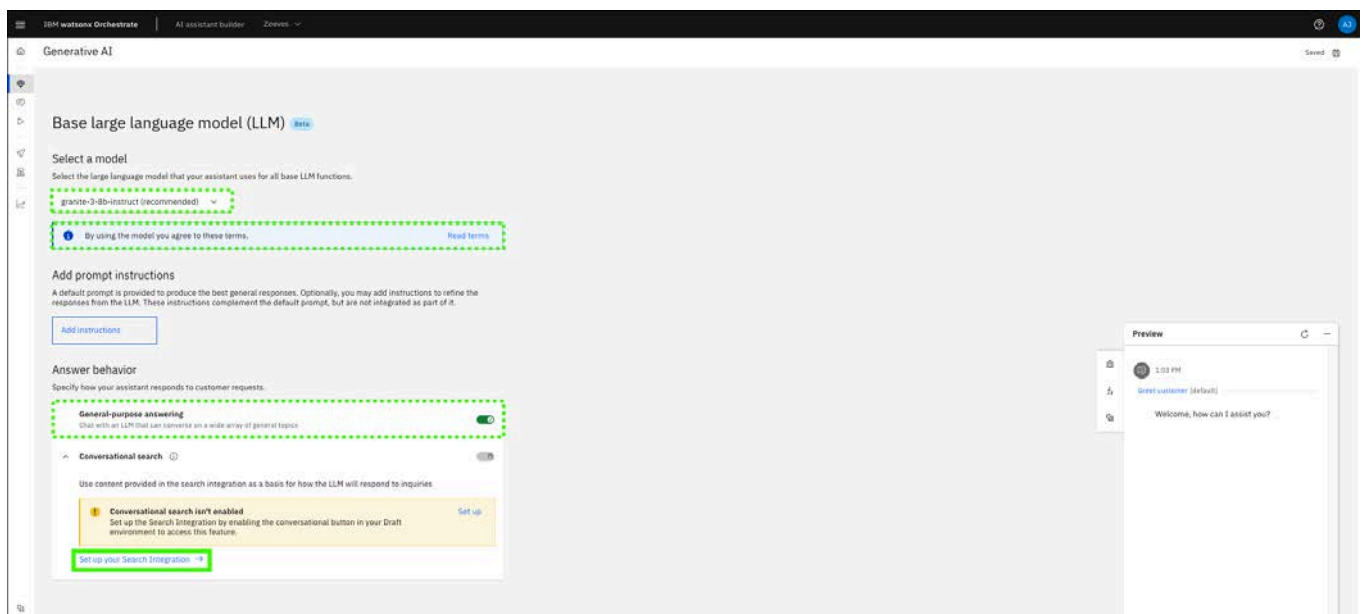


2. Select **granite-3-8b-instruct** for the base large language model (LLM) settings.

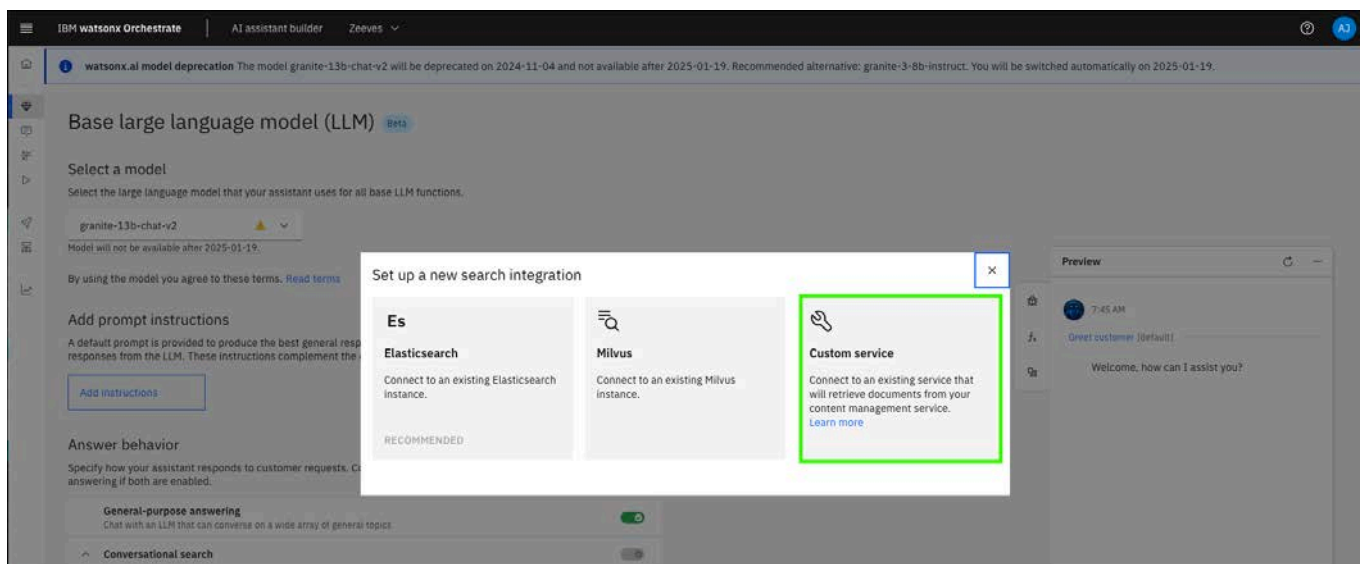


3. Click **Set up your Search Integration**.

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



4. Click **Custom service**.



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

a. Select **By providing credentials**.

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

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

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

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

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

```
secureP@ssw0rd!
```

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

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

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

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

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

a. Enable Conversational search.

b. Select Single turn. Multi-turn conversation (by selecting Entire conversation) is supported by the offering, but has not been fully included in the lab guide. See the callout in the [Testing conversational search](#) section below.

c. Specify the text that appears to instruct the user to expand the list of citations in the assistant (except web chat client).

d. Select Lowest for the retrieval confidence threshold setting. This setting checks the confidence of the retrieved citations before a response is generated.

e. Select **Verbose** for the **generated response length**. This setting affects the average response length. Depending on user input, variations from the selected length can occur.

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

Custom service [Draft](#) Close Save

Settings **Instance**

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

Conversational search
☒ On

Conversational response
☒ 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
To test this, go to the [solutions page](#). [Restore default](#)

Retrieved confidence threshold
☒ Lowest ☐ Low ☐ High ☐ Highest

Generated response length
☐ Concise ☐ Moderate ☒ Verbose

Response confidence threshold
☒ 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 zassit 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](#) Close Save

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 each chat
All

Default filter
Can be anything (plain text, JSON, etc.)

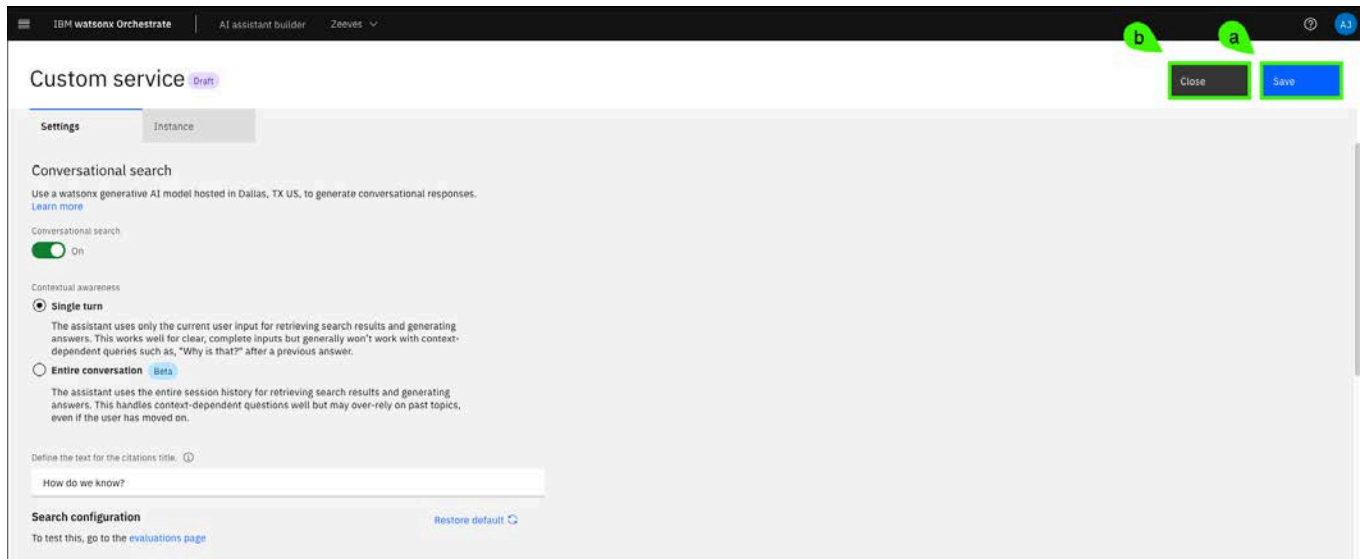
Metadata
JSON example:
{"example_field": "example_value",
"other_example_field": 7}

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.

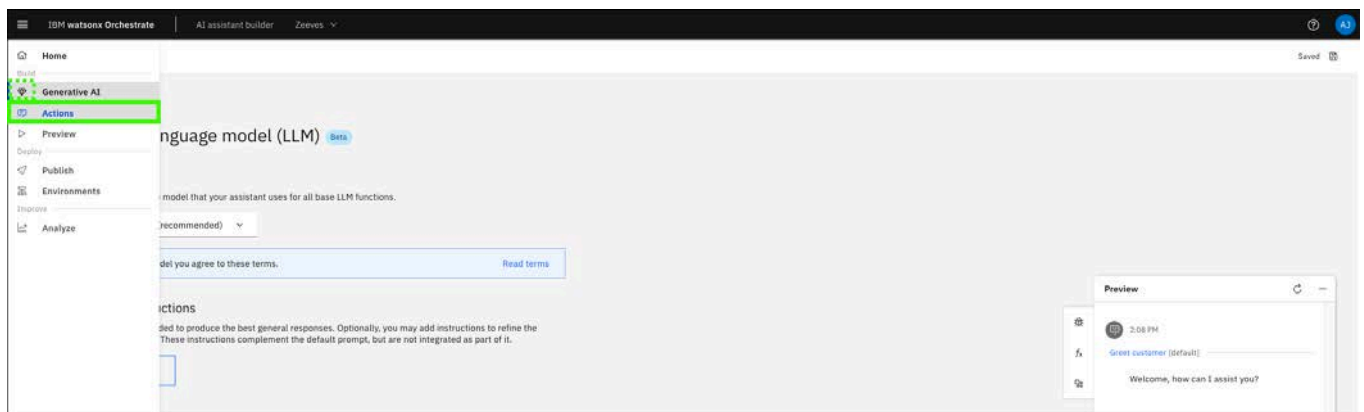
8. Click **Save** (a) and then click **Close** (b).



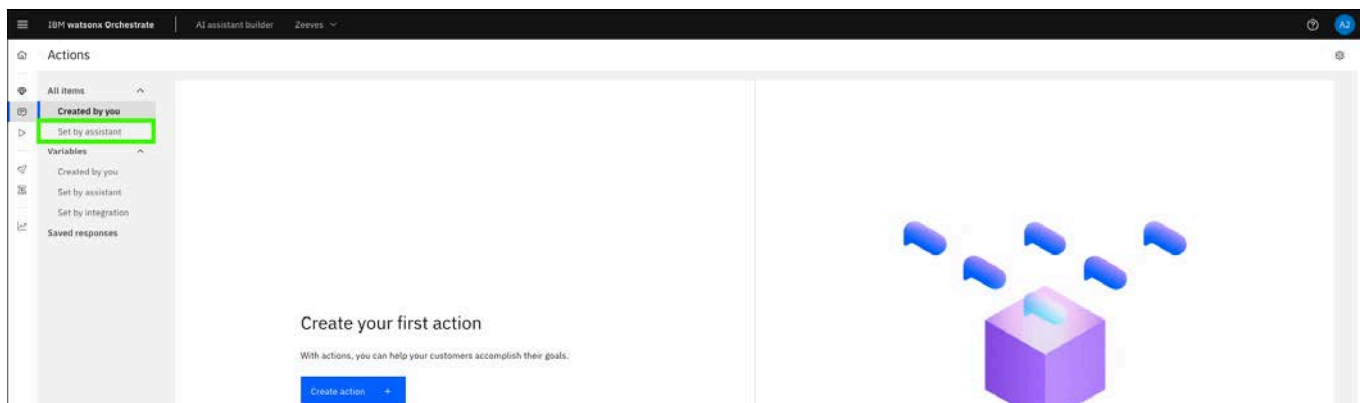
Complete the configuration

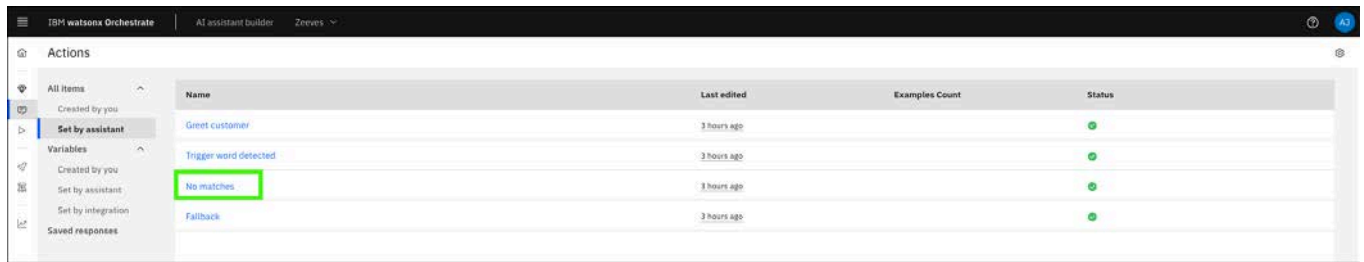
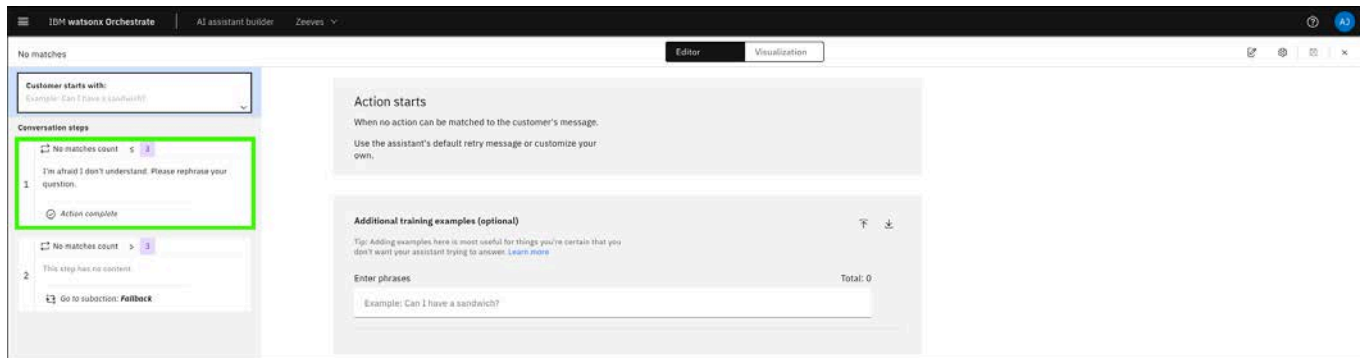
After you save and close the **Conversational search** configuration page, a few more configurations are needed to get the best experience from your conversational chat. Details on these settings are available [here](#).

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

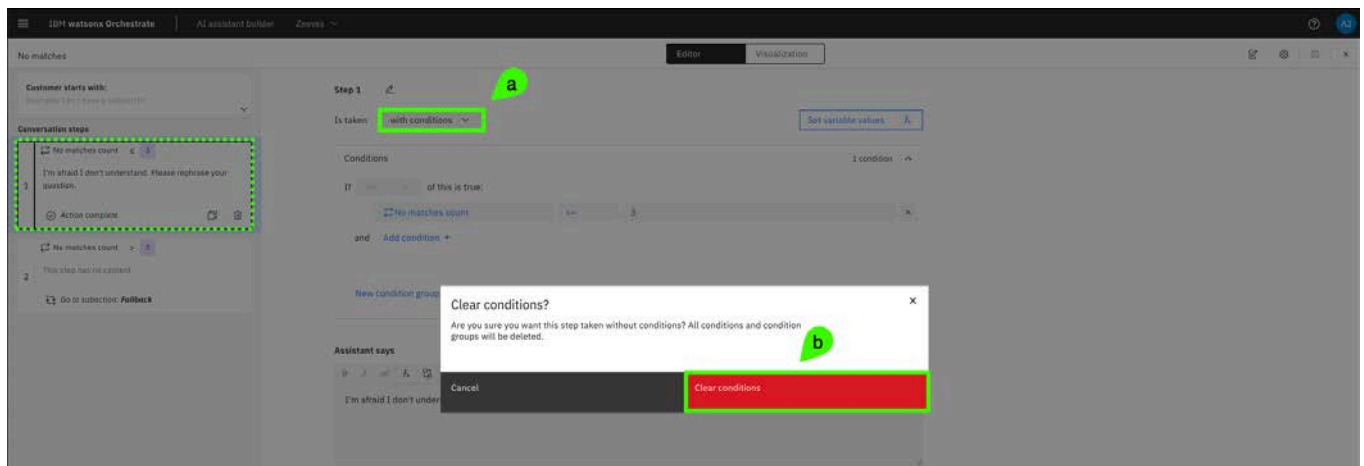
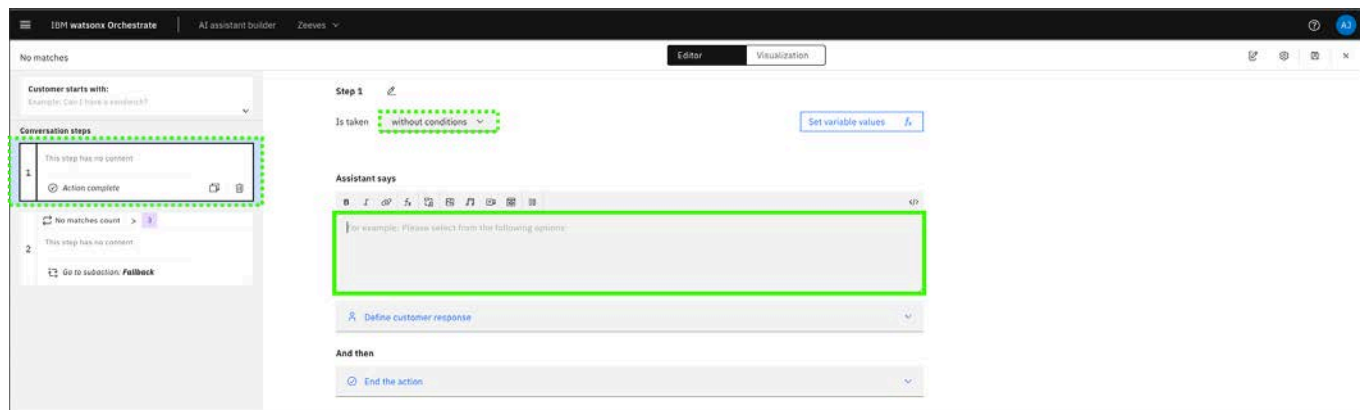


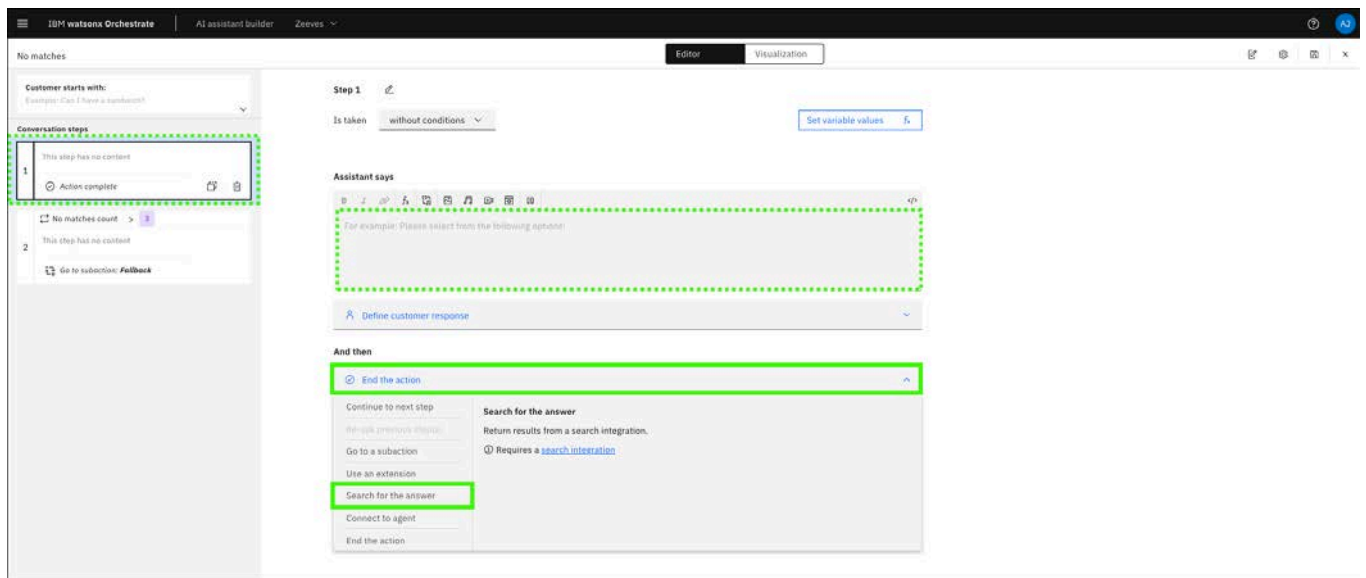
2. Click **Set by assistant** under the **All items** menu.



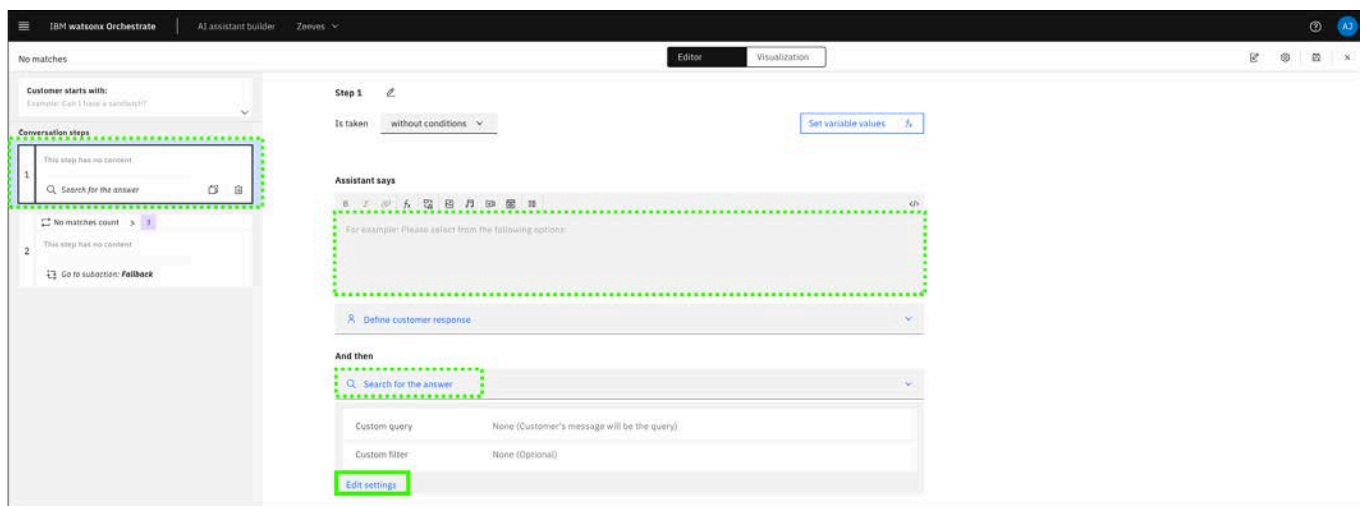
3. Click **No matches**.4. Click **Step 1** under **Conversation steps**.5. Select **without conditions** (a) in the **Is taken** drop-down menu and then click **Clear conditions** (b).

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

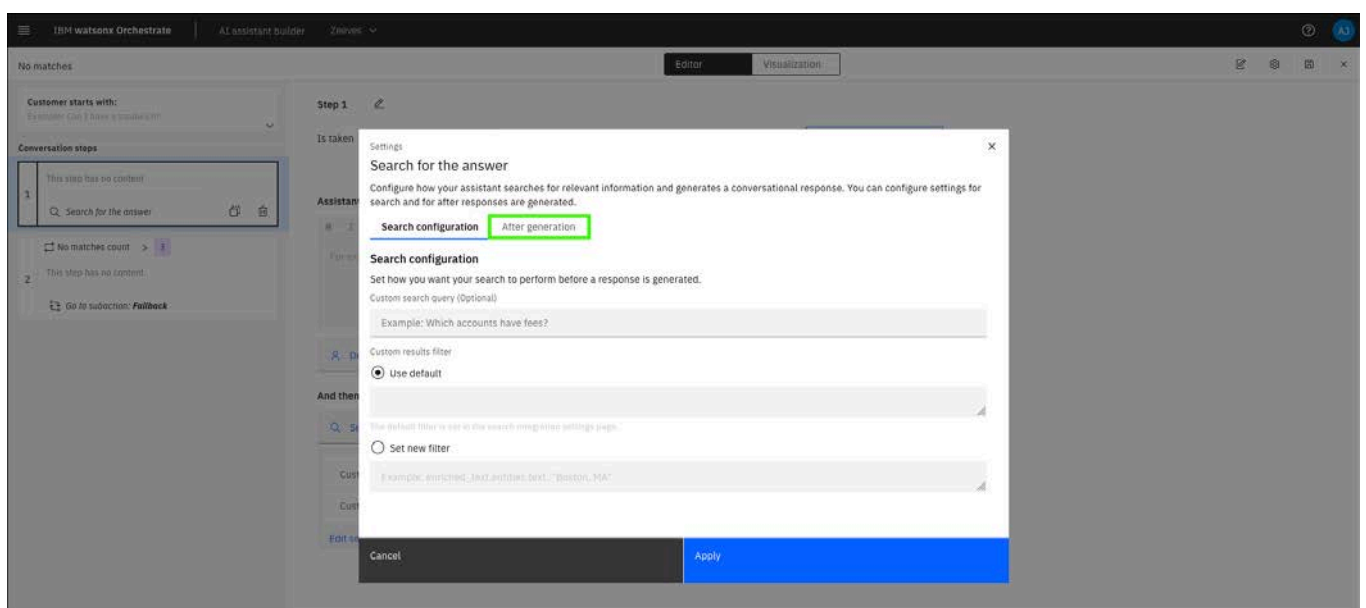
6. Delete the default text in the **Assistant says** entry field.7. Expand the **And then** drop-down menu and select **Search for the answer**.



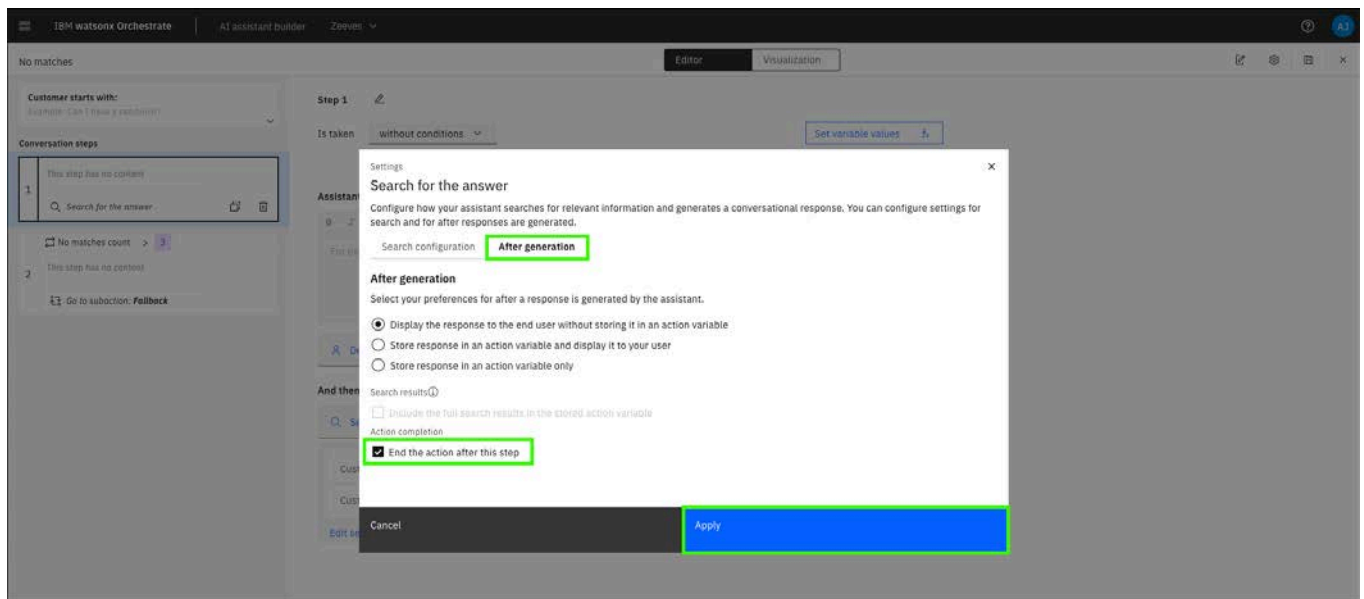
8. Click **Edit settings**.



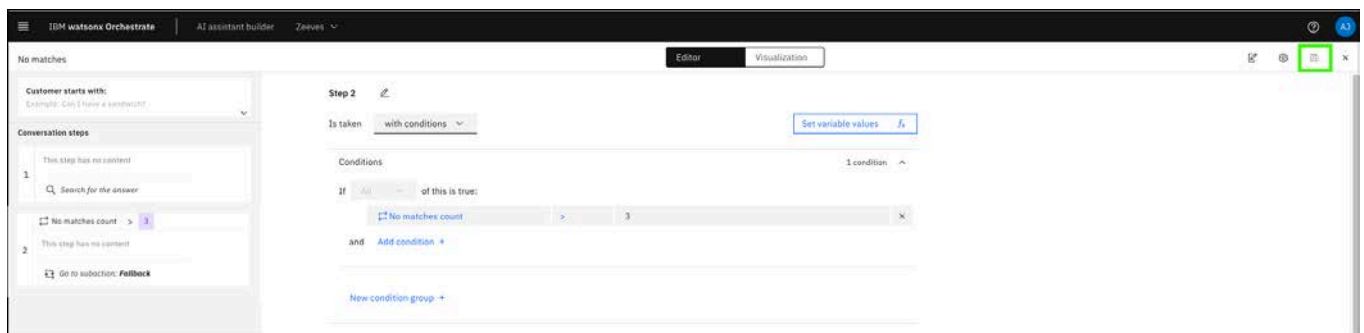
9. Click **After generation**.



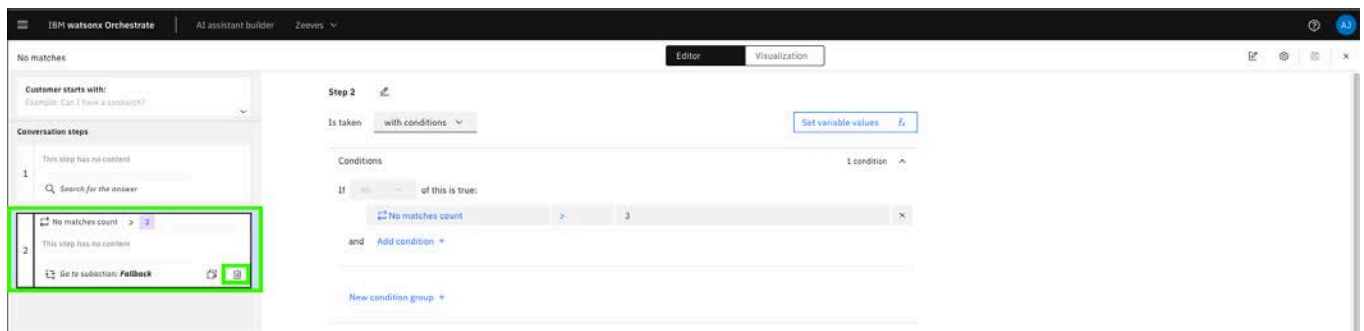
10. Select **End the action after this step** and then click **Apply**.



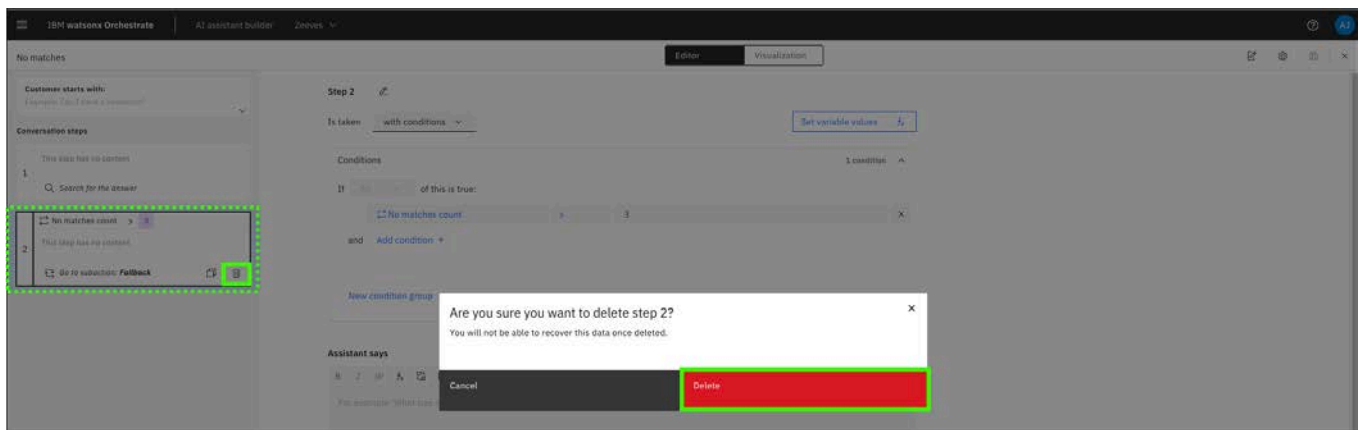
11. Click **Save** (💾).



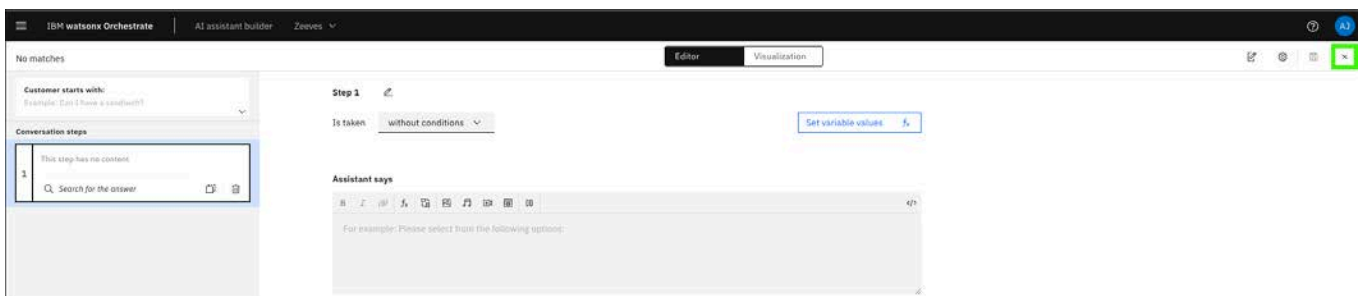
12. Select **Step 2 (No matches count)** under **Conversation steps** and click **delete** (🗑️).



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



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

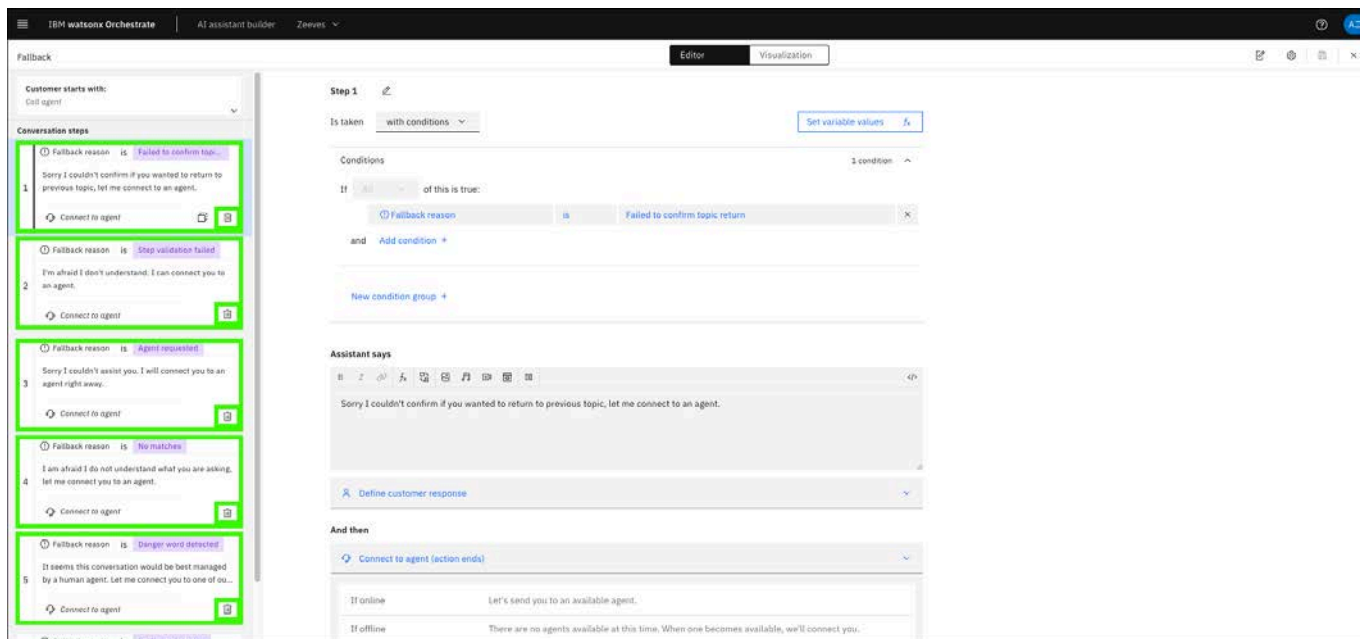


15. Click **Fallback** in the **Actions** table.

Actions				
	Name	Last edited	Examples Count	Status
Created by you	Greet customer	3 hours ago		●
Created by you	Trigger word detected	3 hours ago		●
Created by you	No matches	a minute ago		●
Created by you	Fallback	3 hours ago		●

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

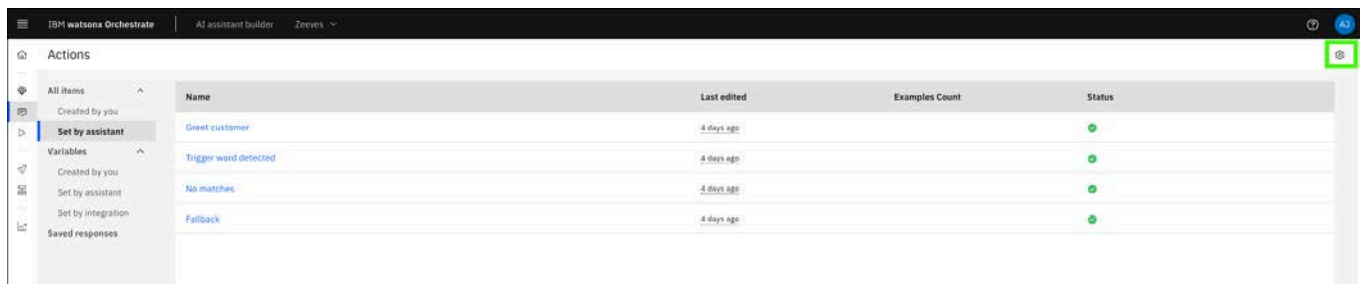
Note: The following image is edited. Only five steps are shown, but all six need to be deleted. You need to select each step individually. Click **delete** (🗑️) and confirm the deletion.



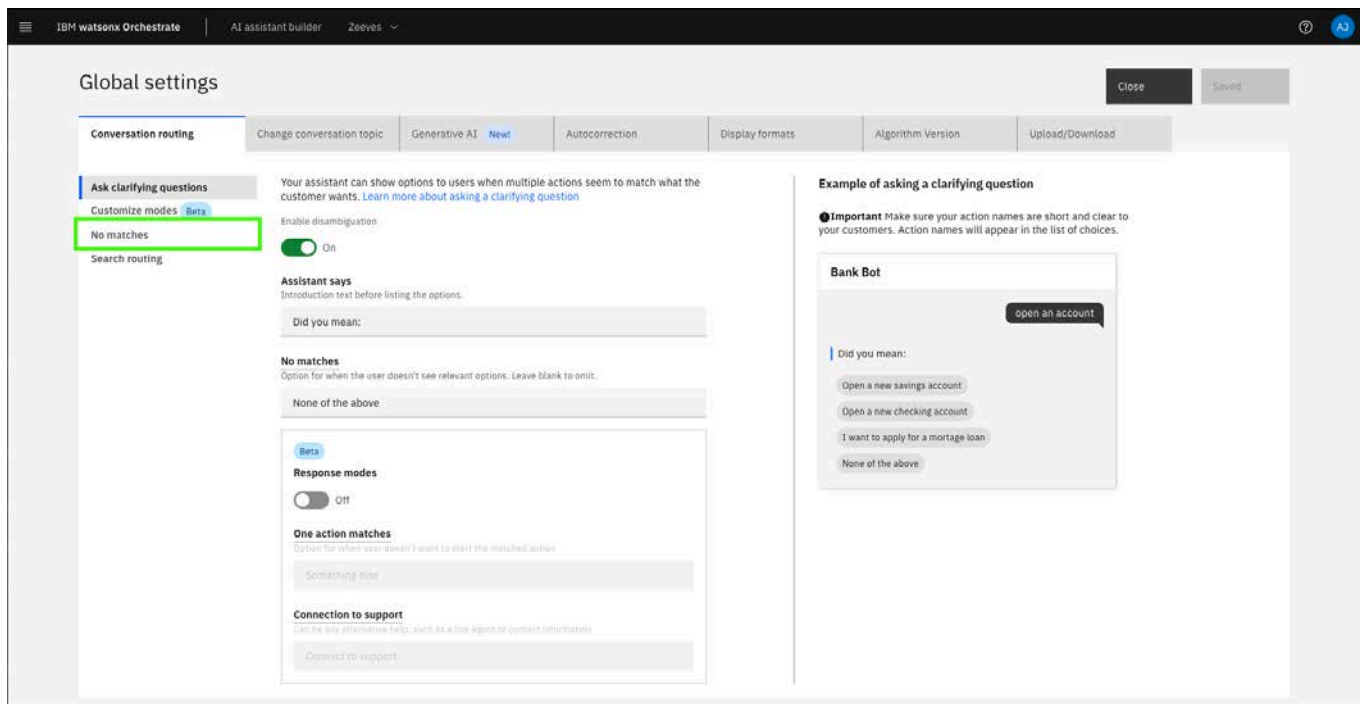
17. Verify that all **Conversation steps** are deleted and then click the **x** to close the **Editor** window.



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

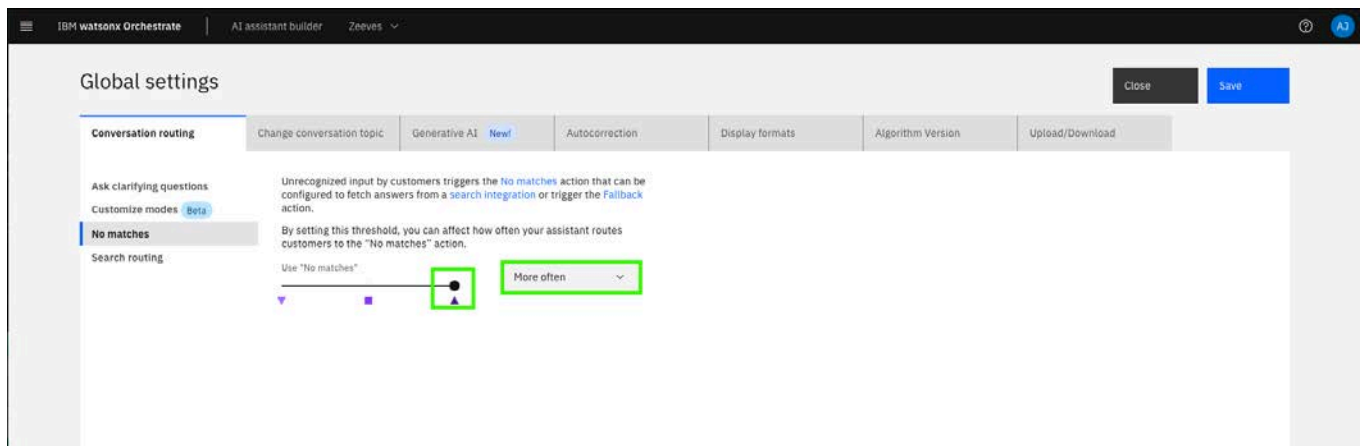


19. Click **No matches** under the **Conversation routing** tab.

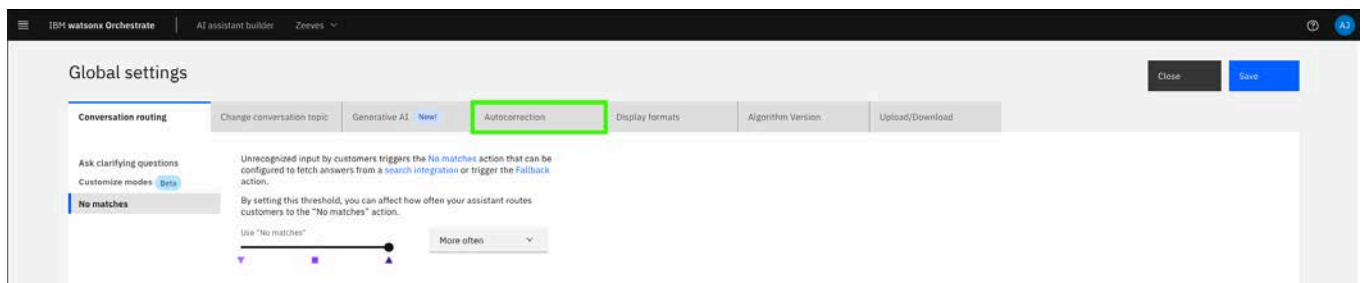


20. Move the slider to **More often** (or select **More often** in the drop-down).

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



21. Click **Autocorrection**.



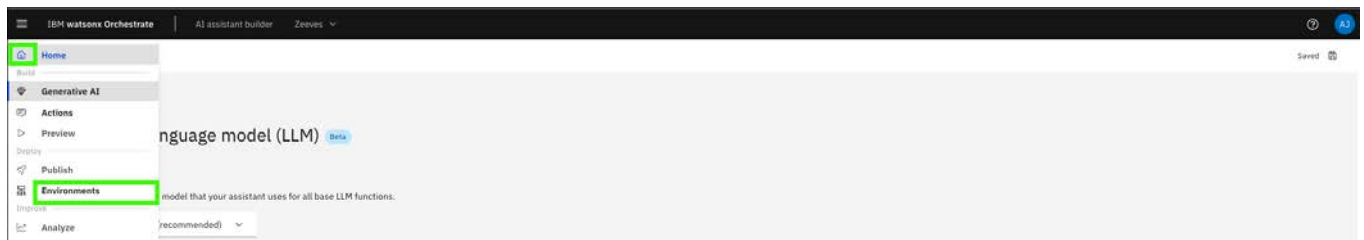
22. Click the autocorrection toggle to turn the feature **Off**.



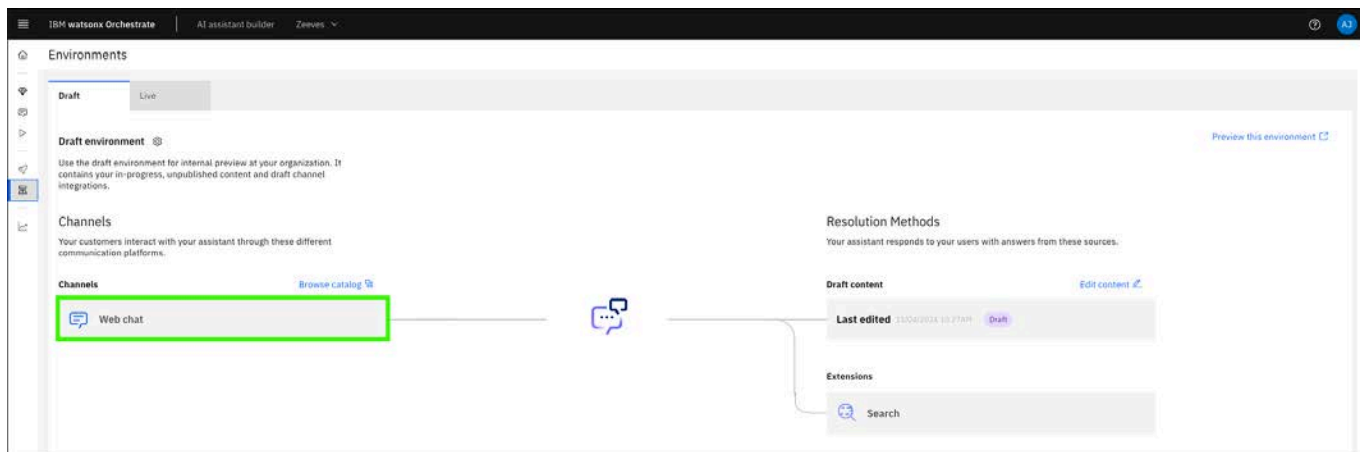
23. Click **Save** (a) and then **Close** (b).



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

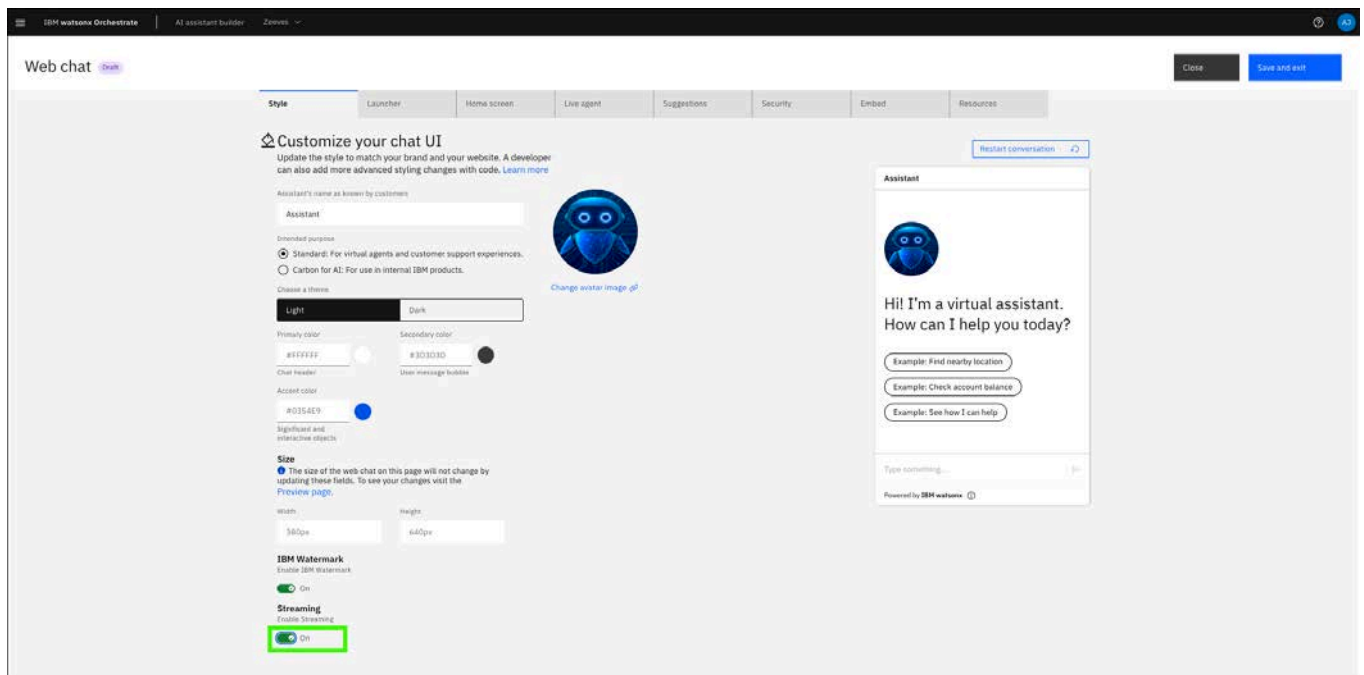


25. Click **Web chat**.

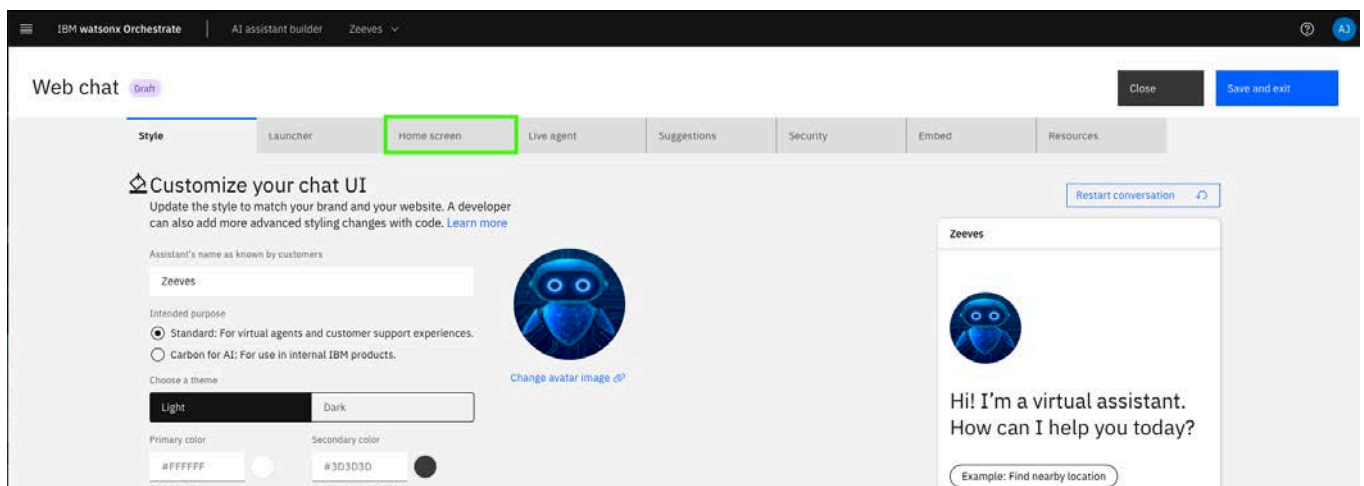


26. On the **Style** tab, click the **Streaming** toggle to enable streaming.

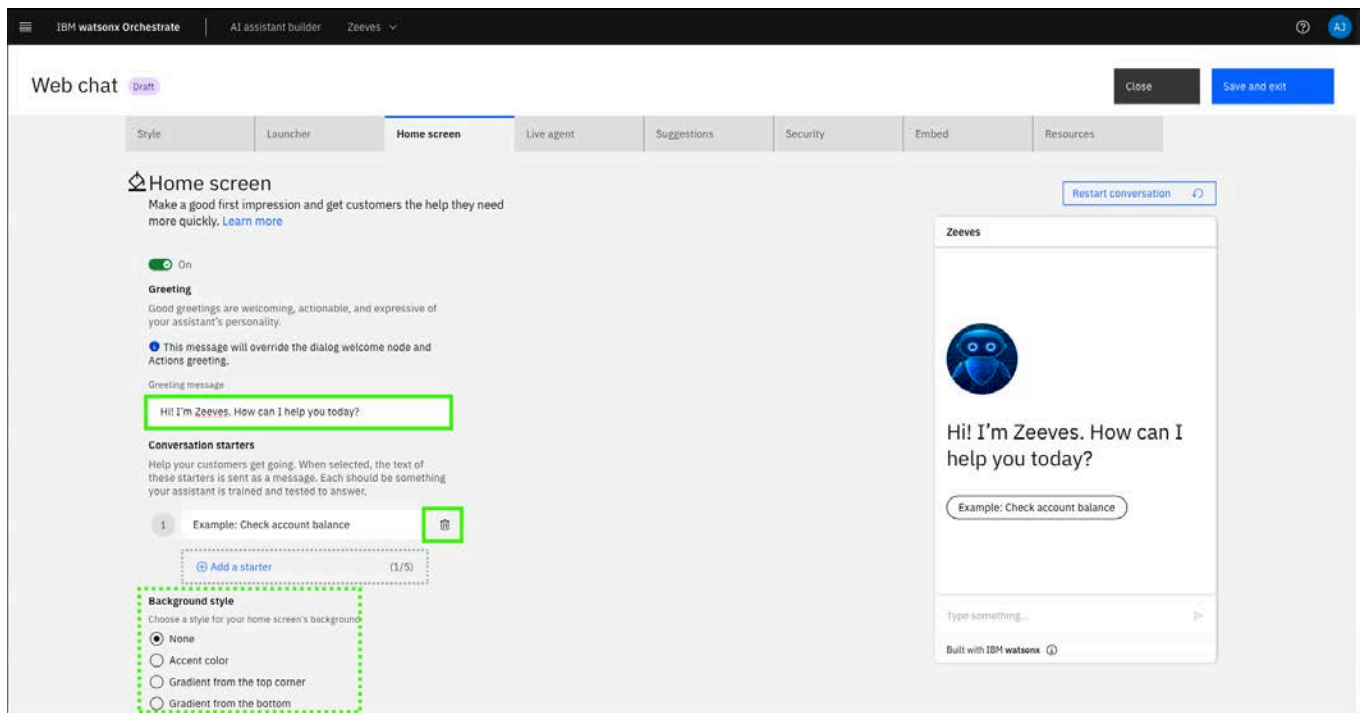
The streaming setting allows responses to be streamed to the assistant and displayed as they are generated versus waiting until the full response is received and then displayed.



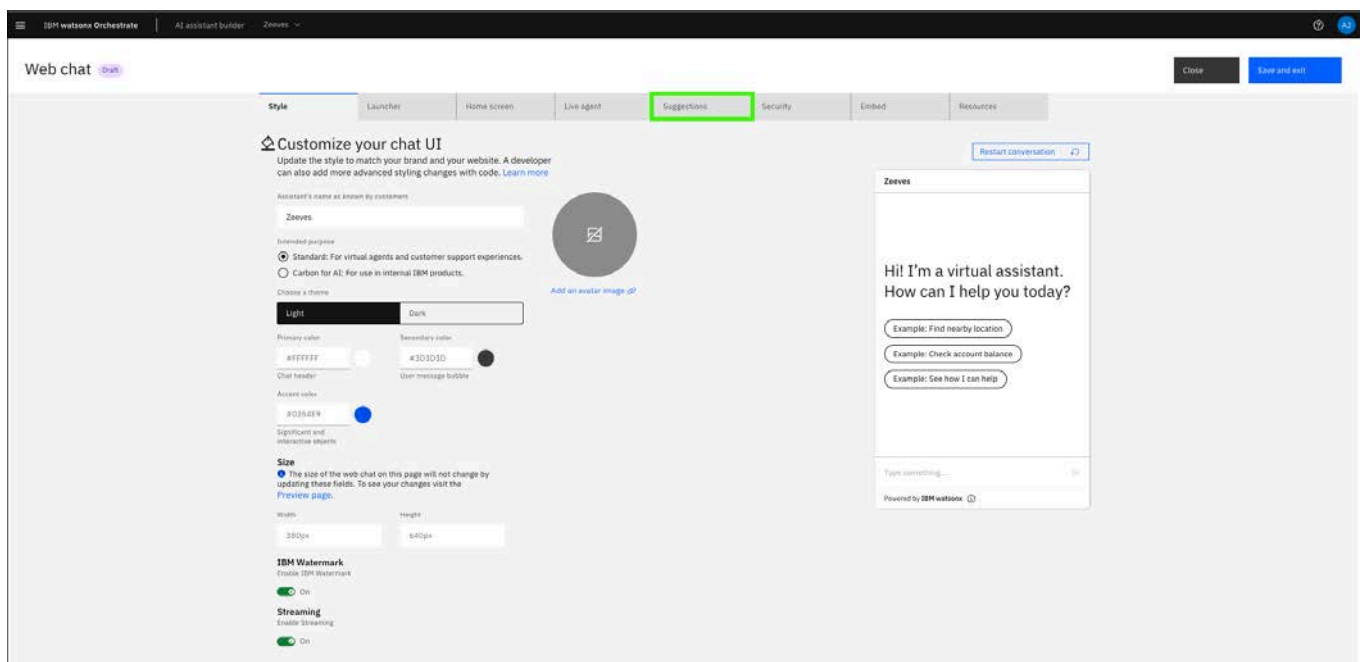
27. Click the **Home screen** tab.



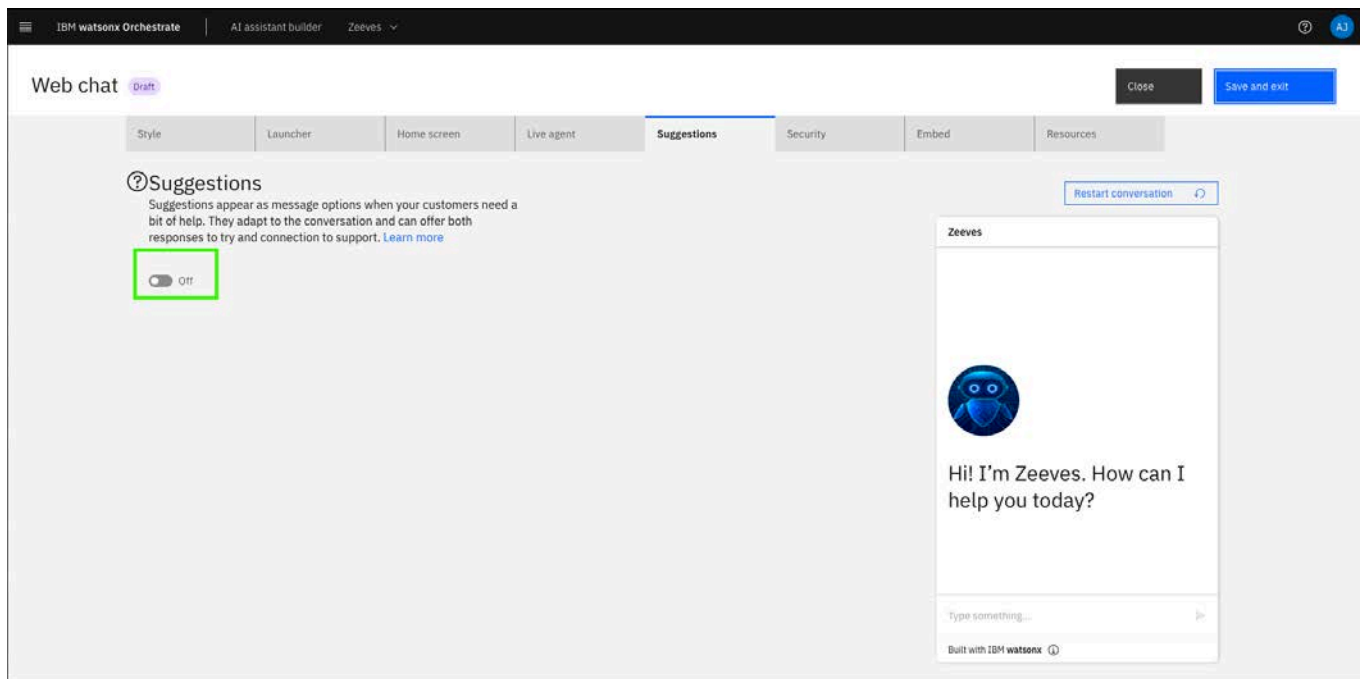
28. Customize the **Home screen** by setting a custom **Greeting message** and deleting the default **Conversation starters**. Optionally, adjust the **Background style**.



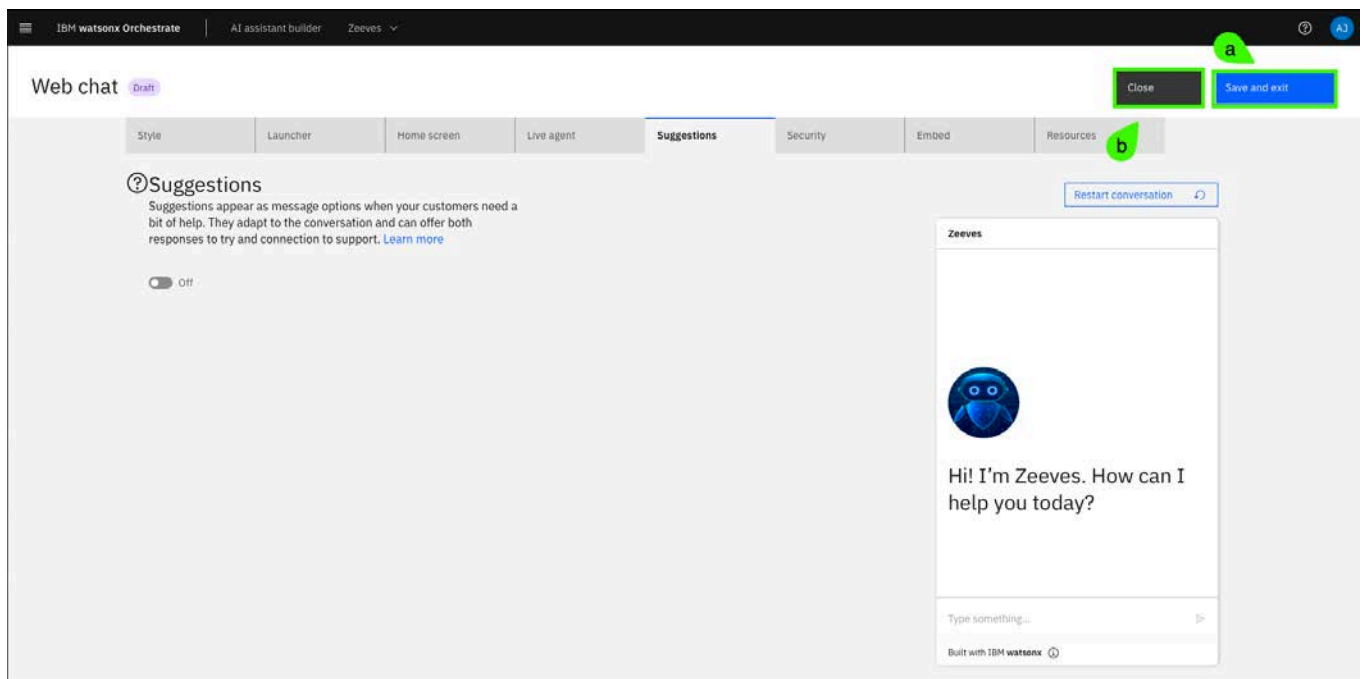
29. Click **Suggestions**.



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



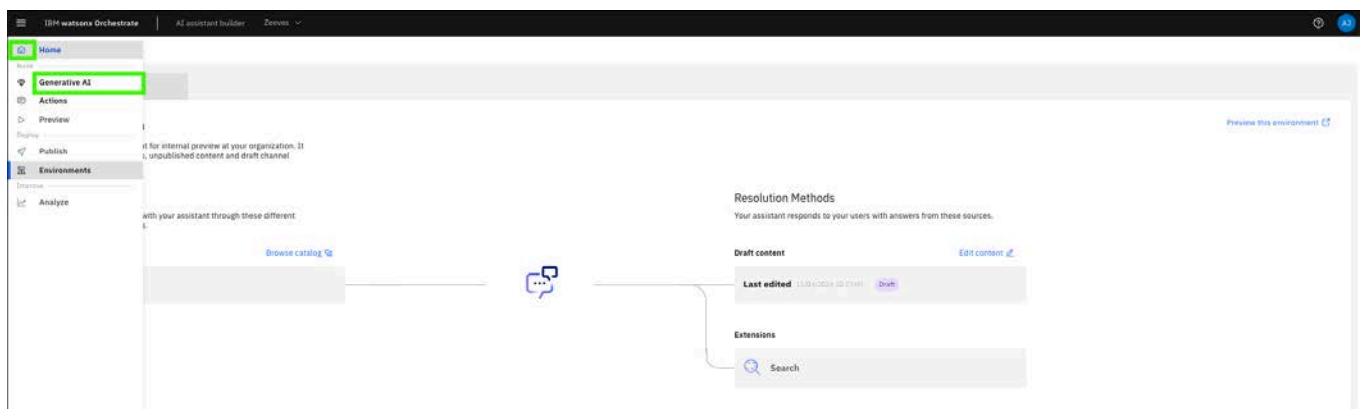
31. Click (a) **Save and exit** and then click (b) **Close**.



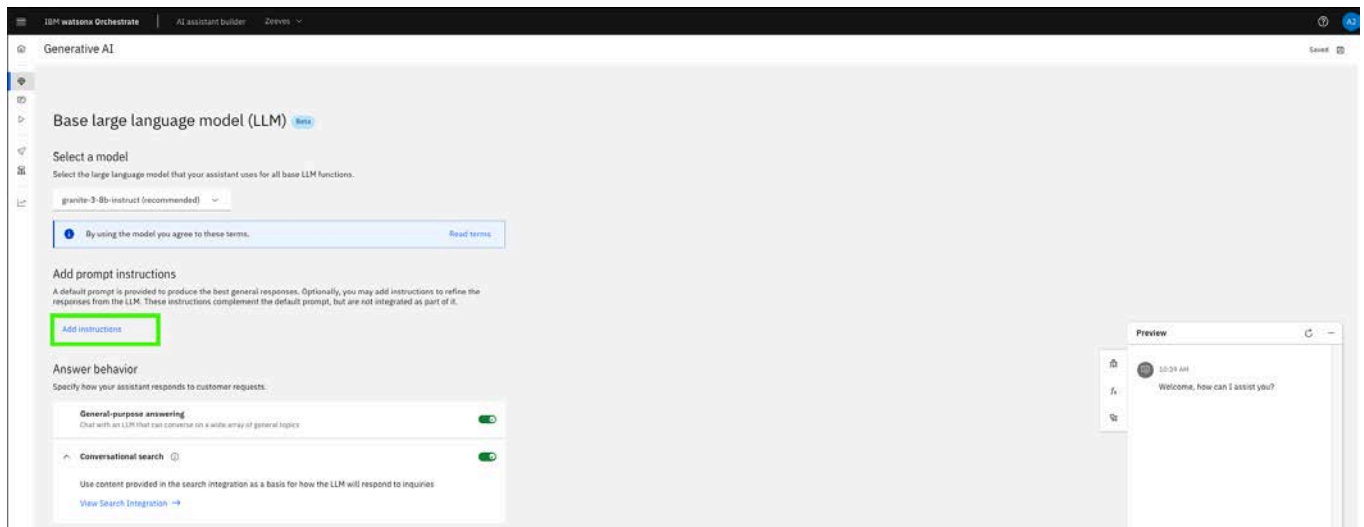
Configure the base large language model

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

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



2. Click **Add instructions**.



3. Enter a **prompt instruction**.

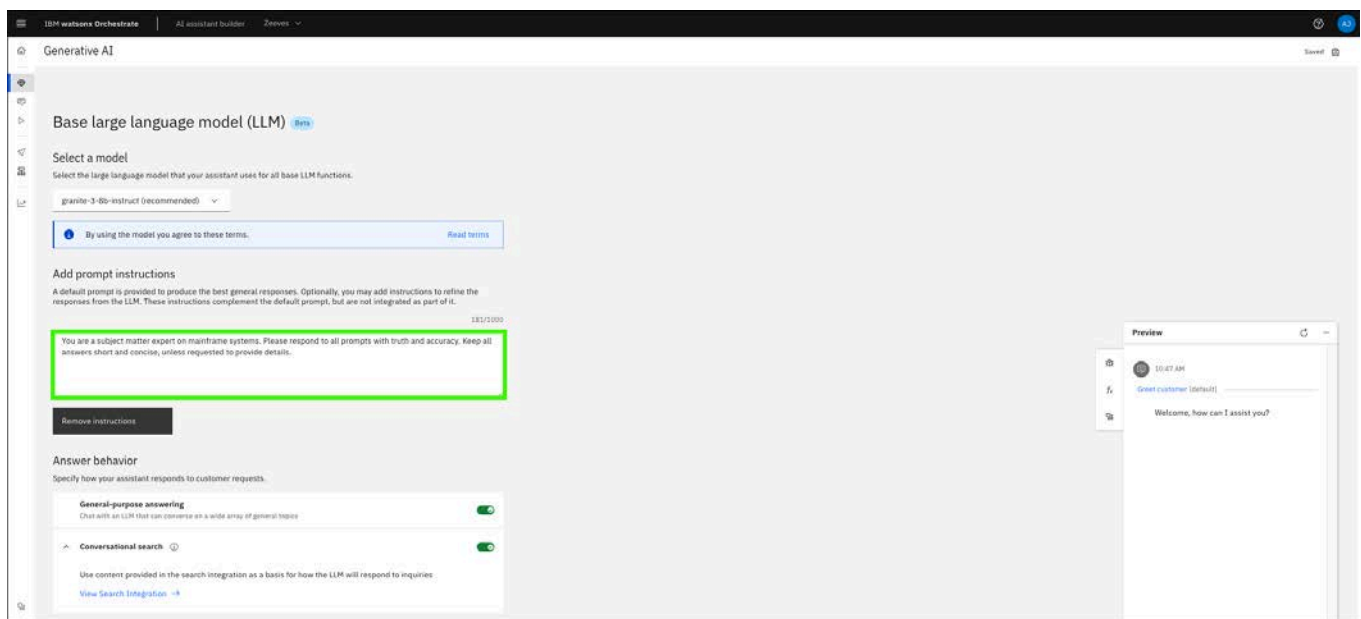
Your assistant's LLM gives refined responses by following the prompt's instructions, which clarify how to achieve the end-goal of an action.

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

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

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

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



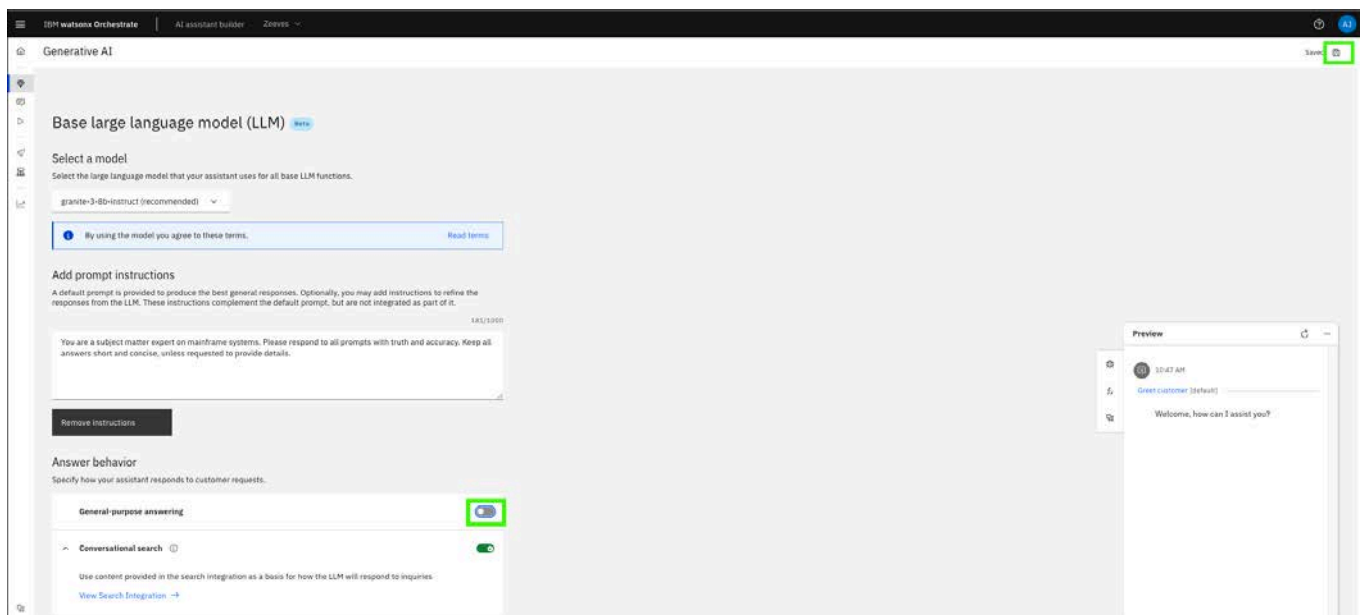
4. Toggle **General-purpose answering** to **Off** and then click **Save** (💾).

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.

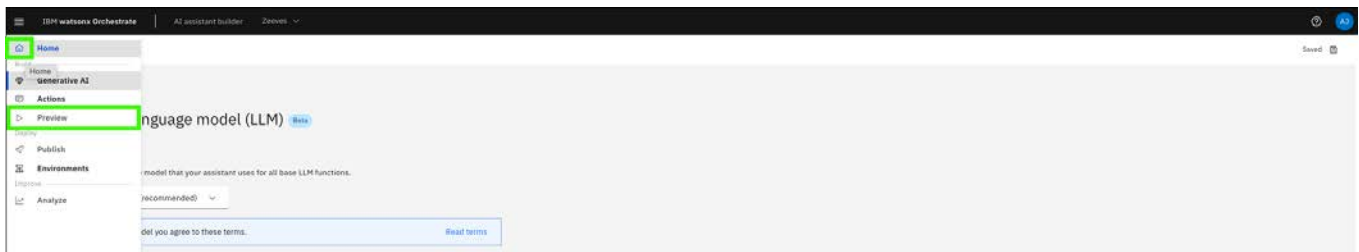


Testing conversational search

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.

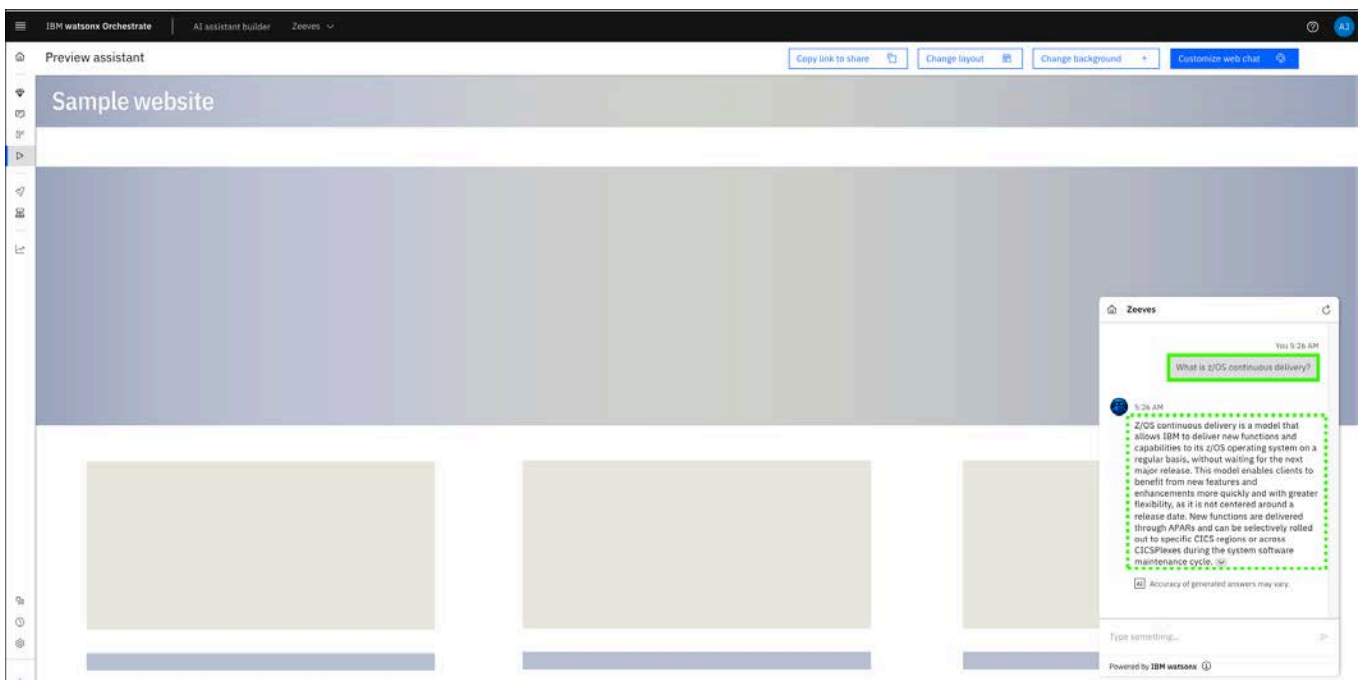
Other prompts and responses follow.

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

Prompt:

What is z/OS continuous delivery?

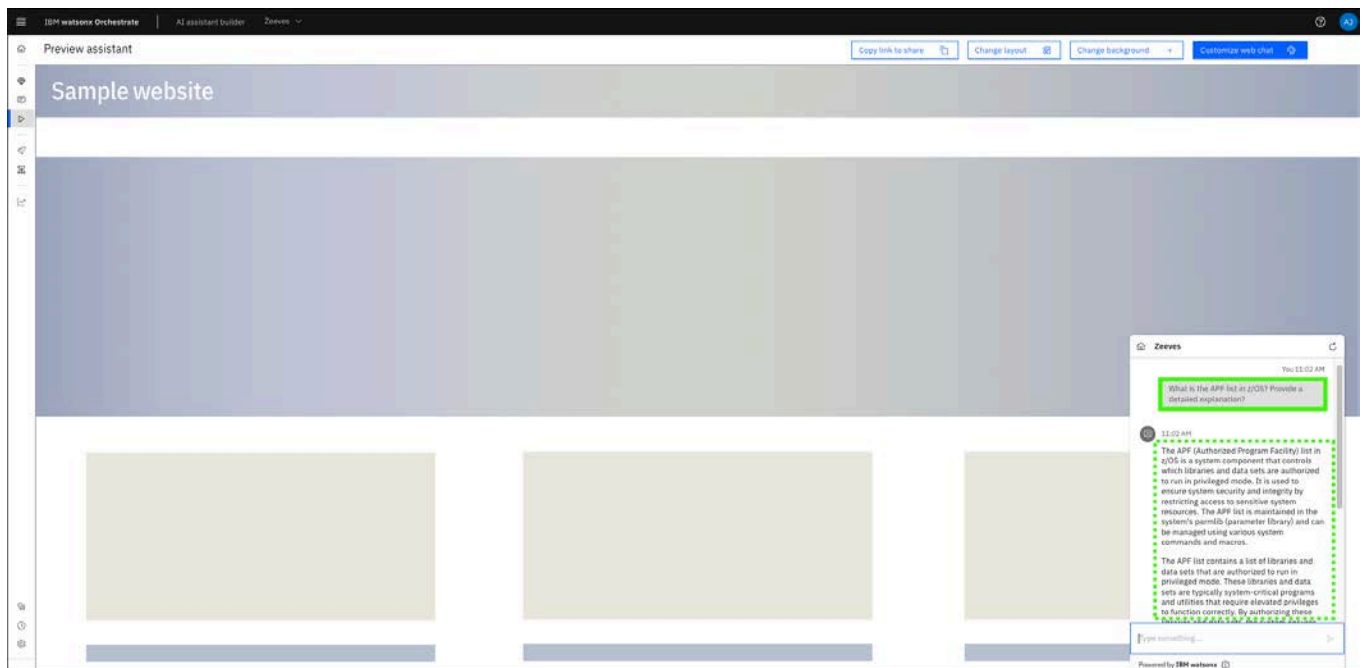
Example output:



Prompt:

What is the APF list in z/OS?

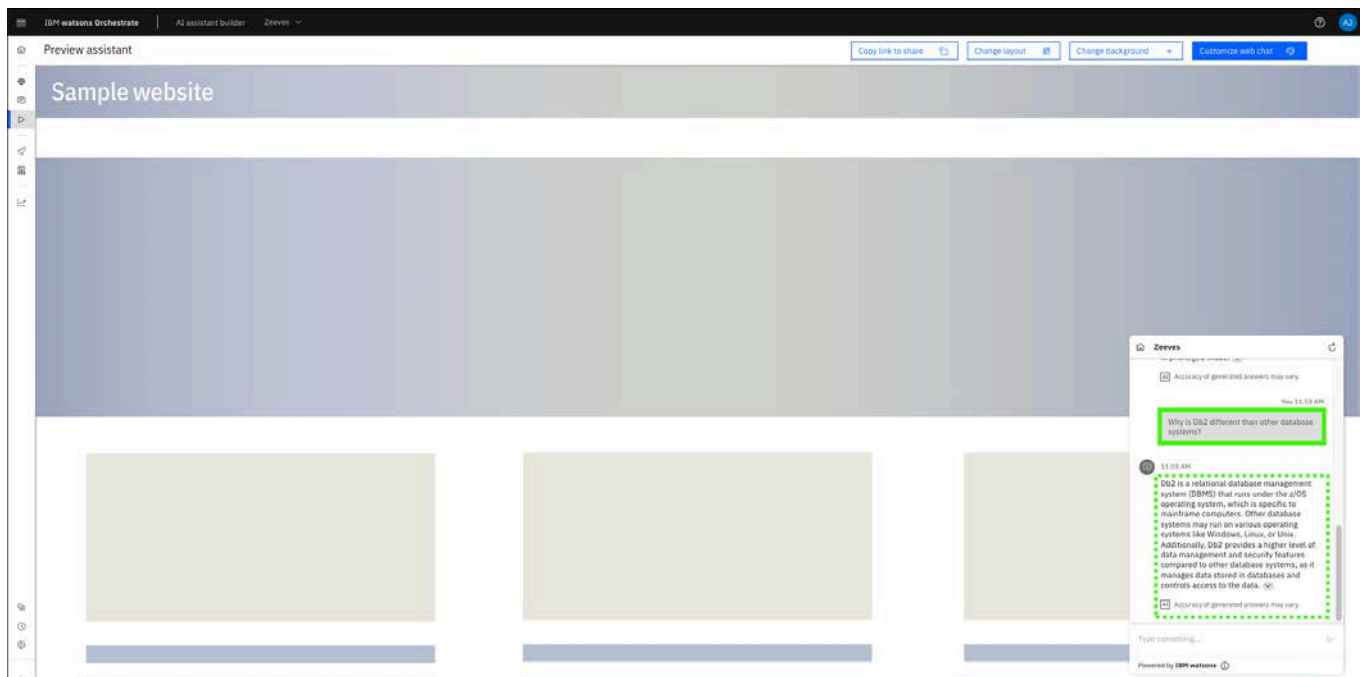
Example output:



Prompt:

Why is Db2 different than other database systems?

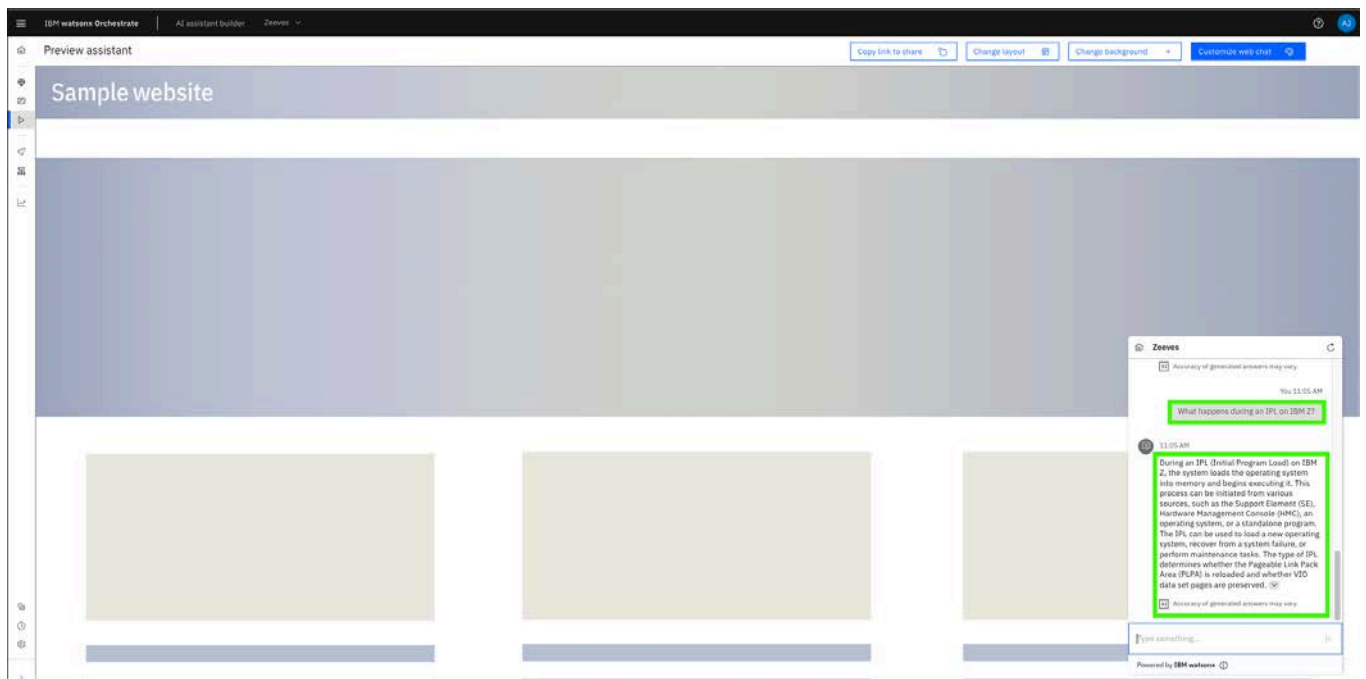
Example output:



Prompt:

What happens during an IPL on IBM Z?

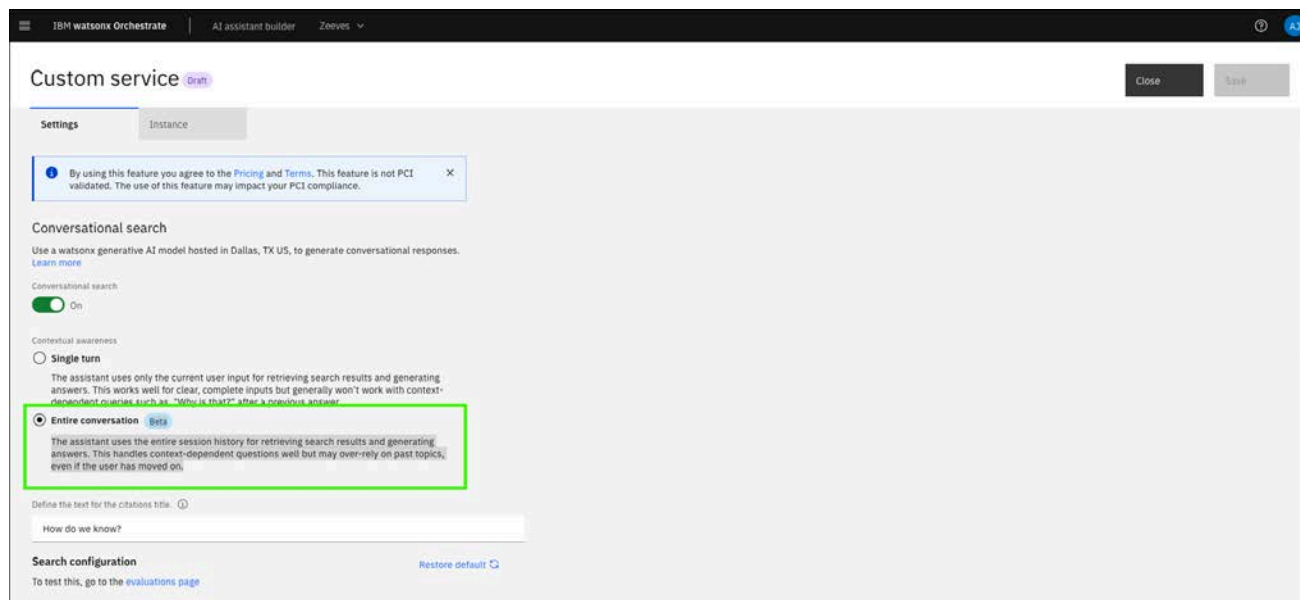
Example output:



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

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

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



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

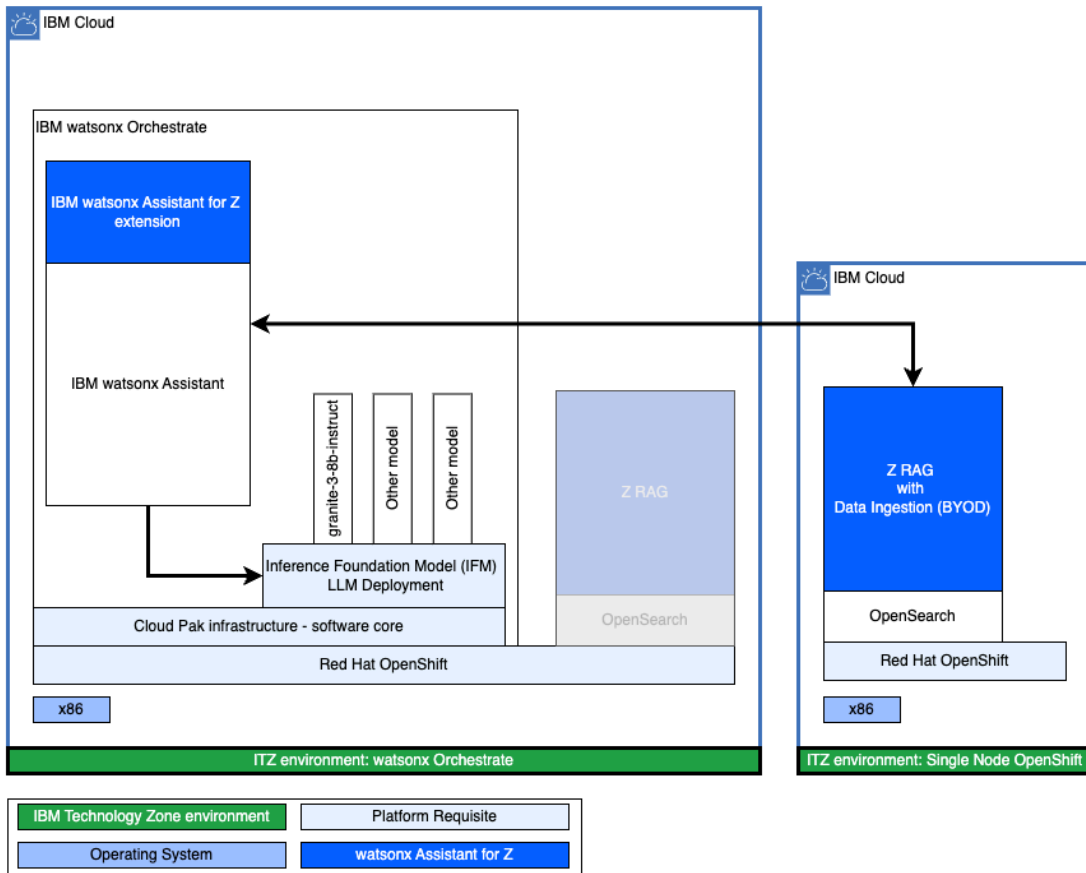
Now that you created and deployed your own assistant with conversational search capabilities, your client can understand how watsonx Assistant for Z provides its content-grounded responses to any Z-related questions. In the previous section, you configured your assistant to use a pre-configured [Z RAG](#)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

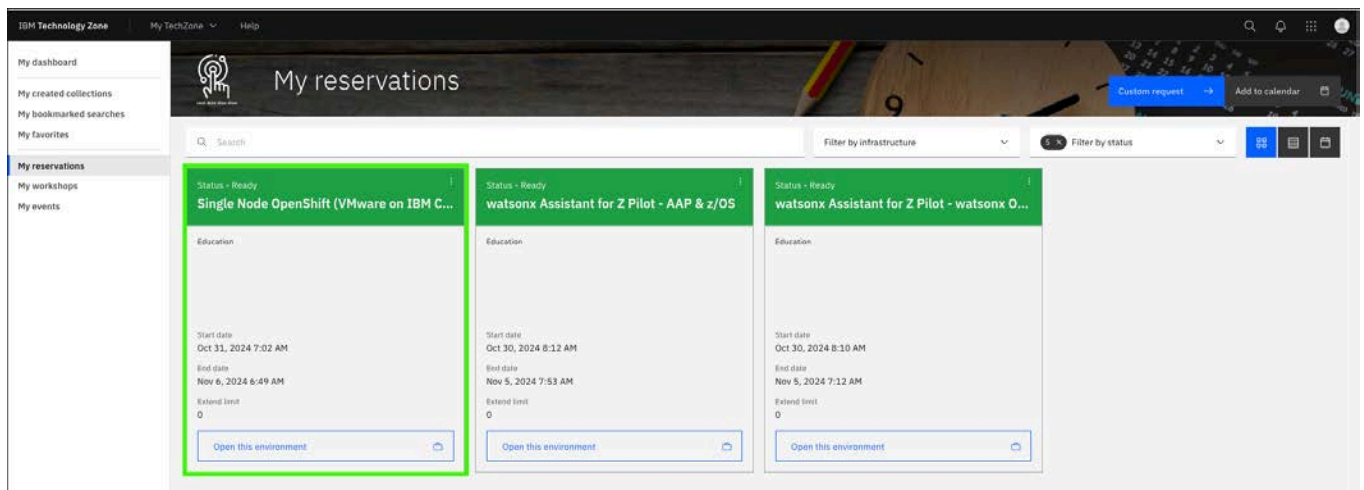
Install the Red Hat OpenShift command line interface utility

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](#).

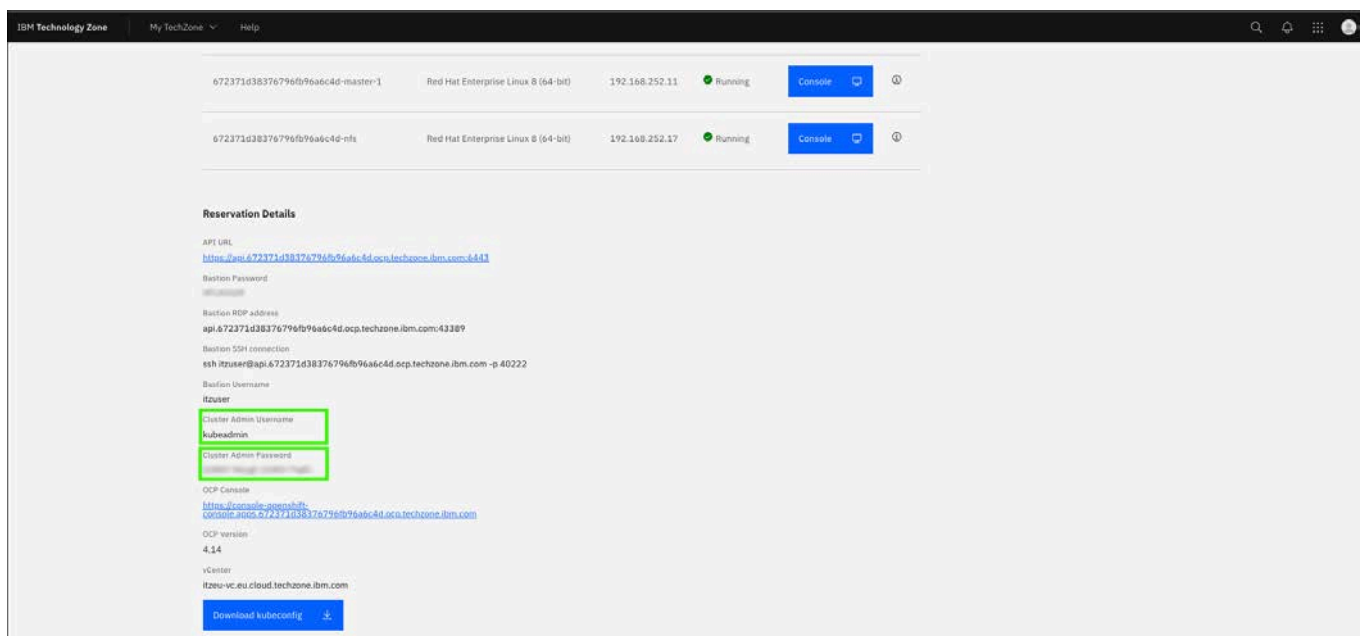
1. Click the following link to open a browser window to your ITZ reservations.

[ITZ My reservations](#)

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

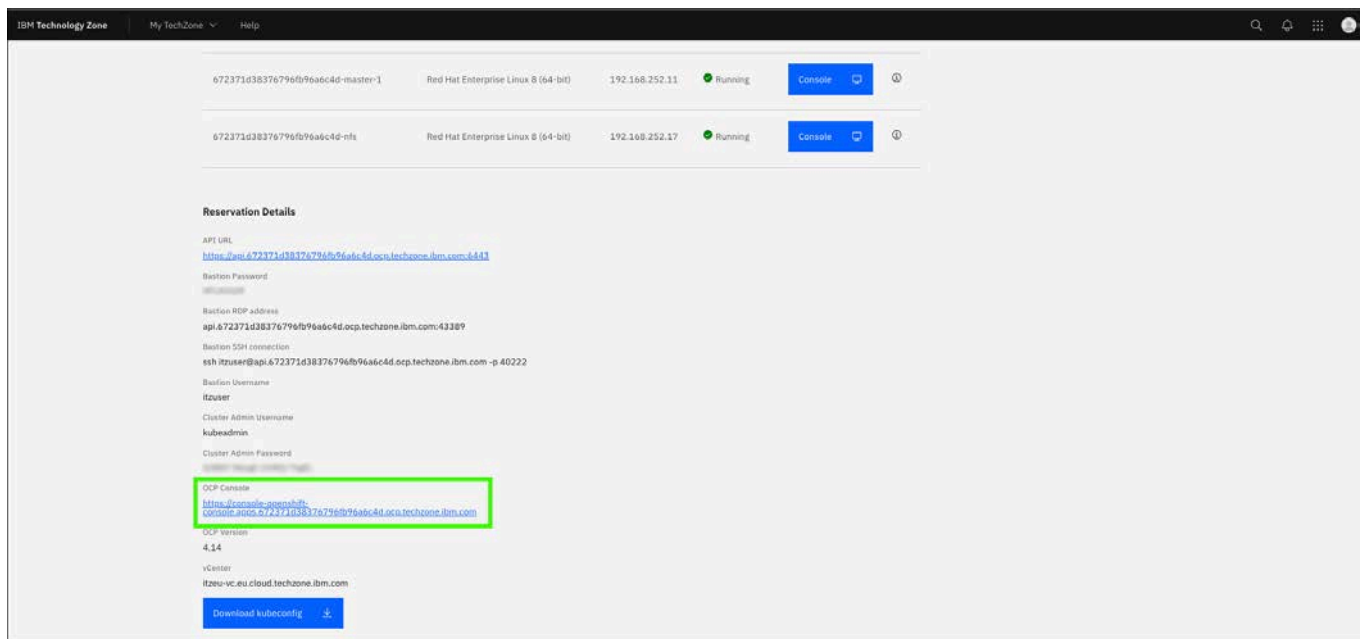


3. Scroll down and record the **Cluster Admin Username** and **Cluster Admin Password**.

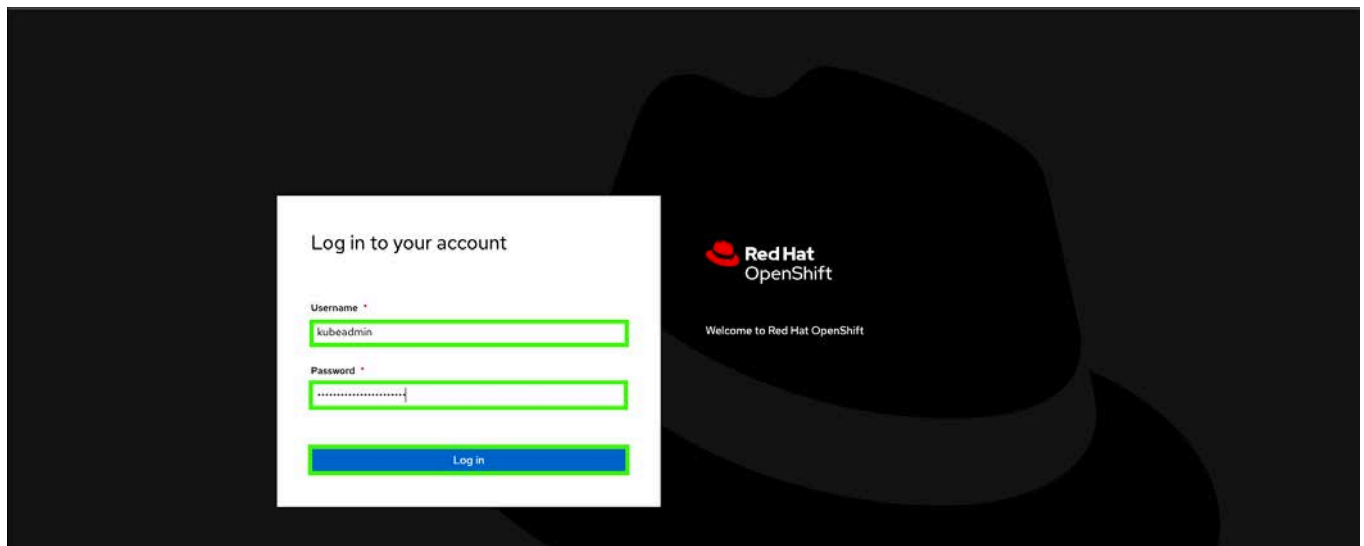


4. Click the **OCP Console** link.

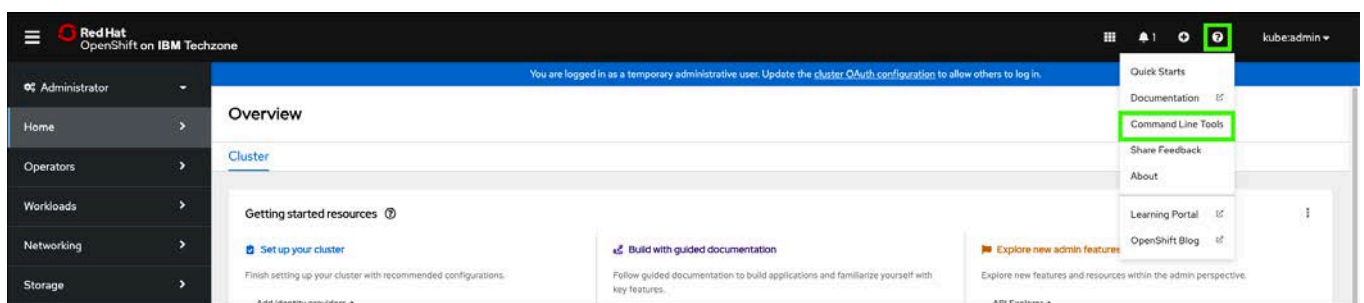
Note: OCP stands for OpenShift Container Platform.



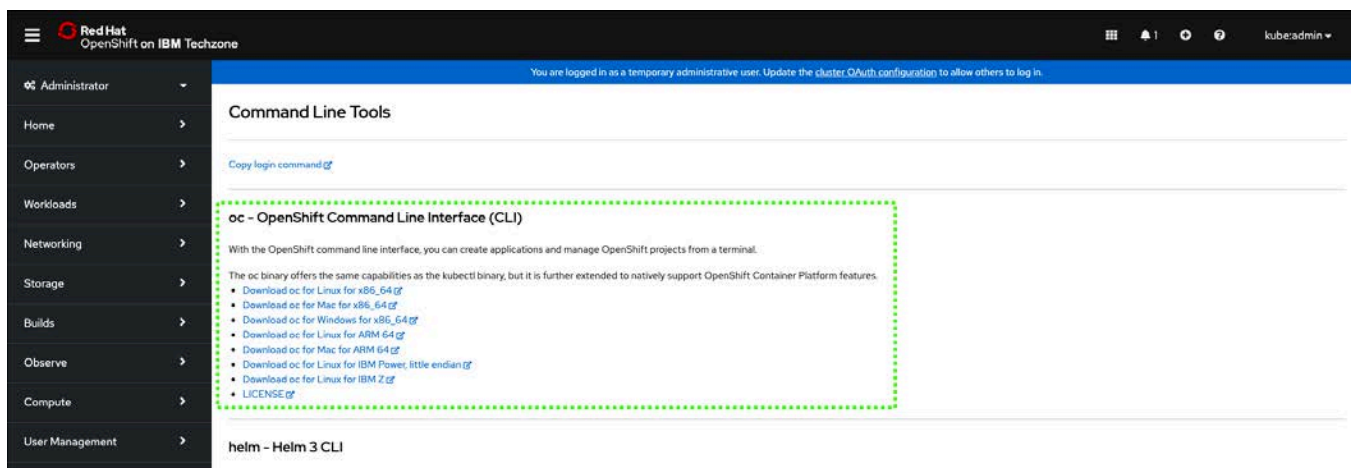
5. Enter the **Cluster Admin Username** and **Cluster Admin Password** values from step 3 and click **Log in**.



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.



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
  rollback        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.

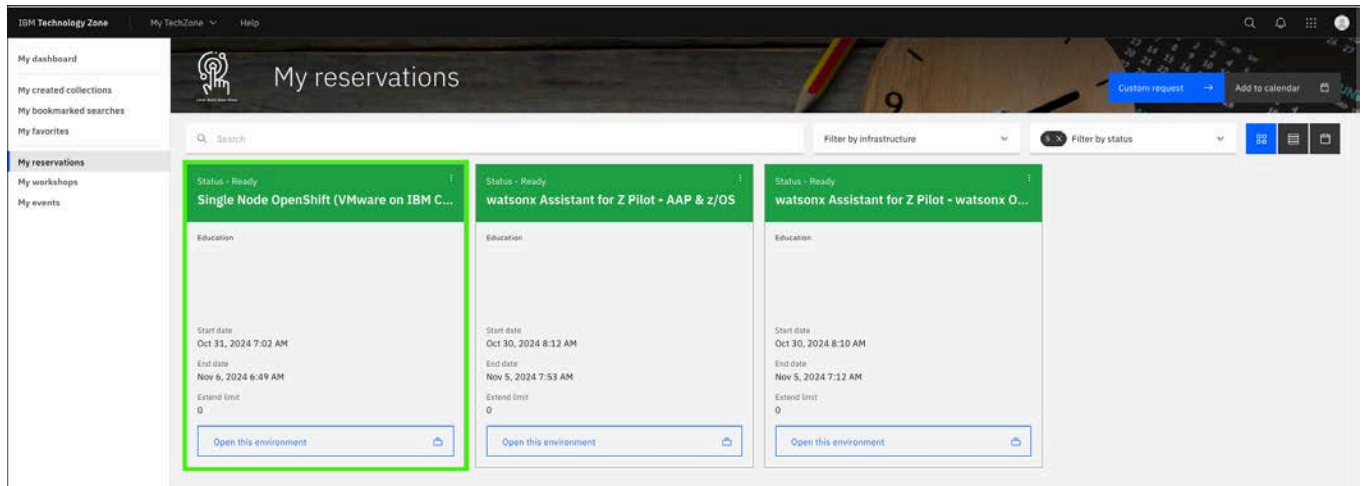
Log in to the OpenShift cluster from your local terminal

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

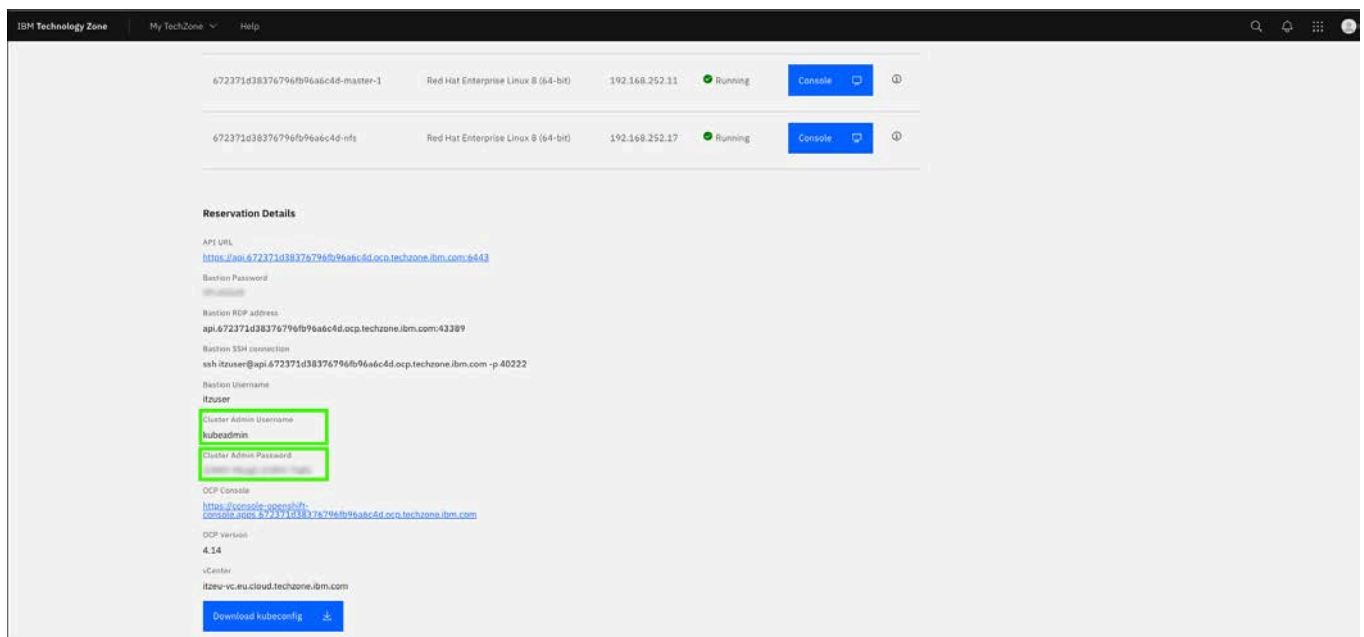
1. Click the following link to open a browser window to your ITZ reservations.

[ITZ My reservations](#)

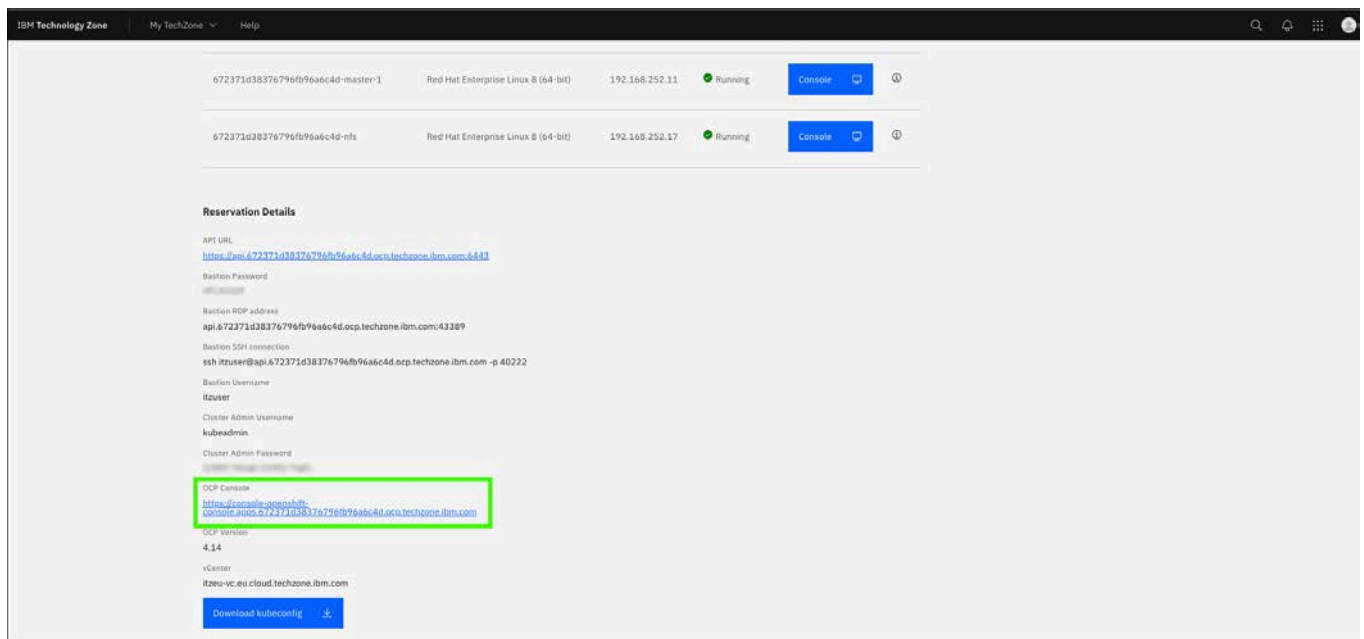
2. Click the **Single Node OpenShift** tile.



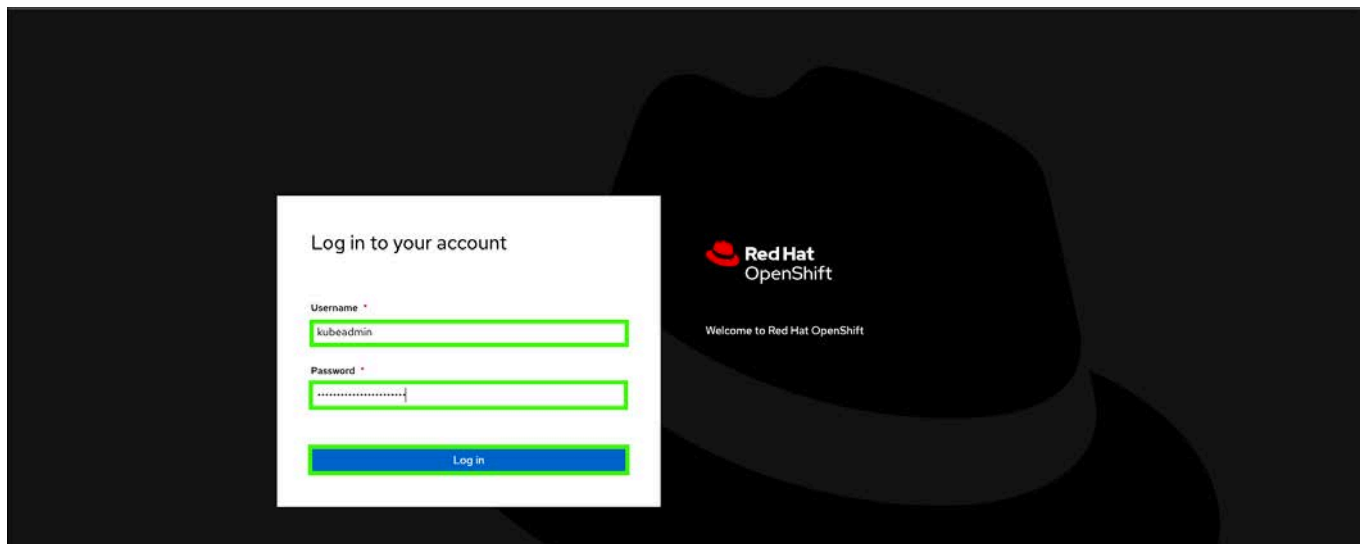
3. Scroll to the bottom of the reservation page and record the **Cluster Admin Username** and **Cluster Admin Password**.



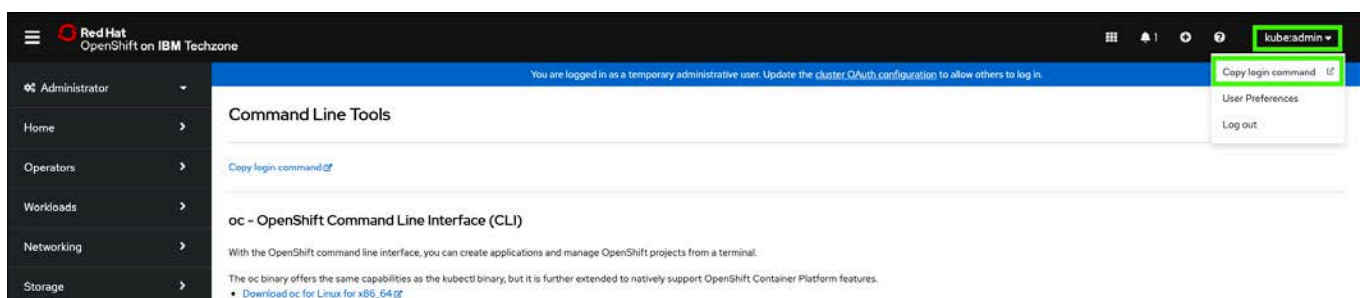
4. Click the OCP Console link.



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



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

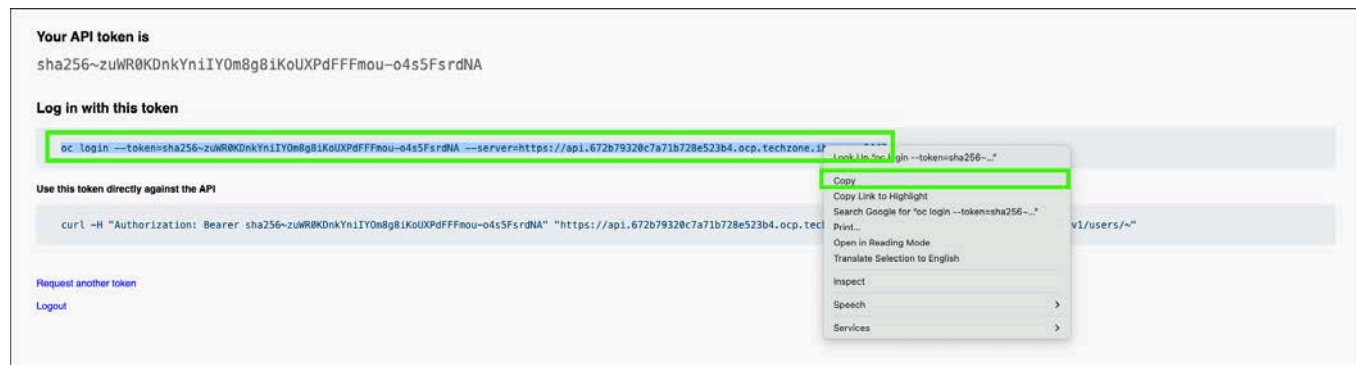


7. Click **Display Token**.



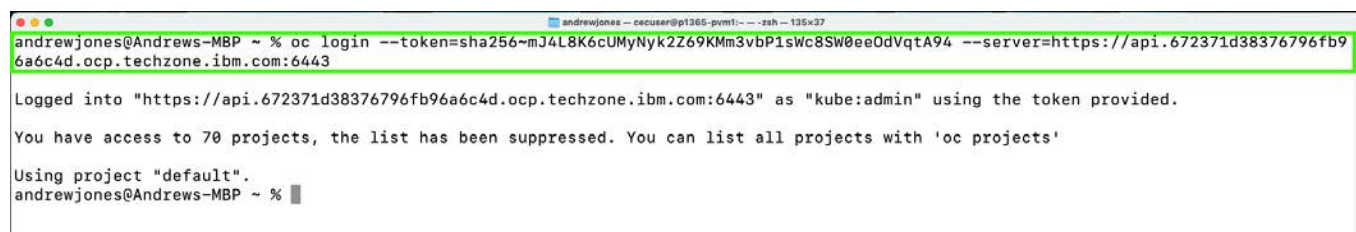
8. Select and copy the **Log in with this token** string.

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



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

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



Create a working directory

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

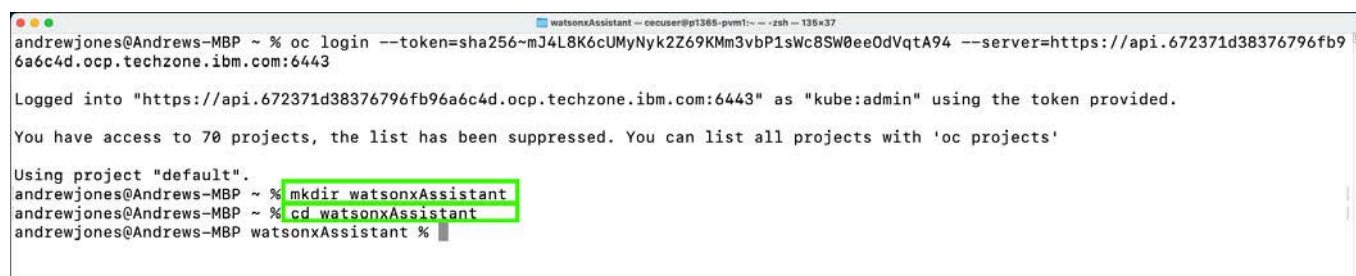
 **Instructions vary by your local workstation's operating system.**

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
```



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:

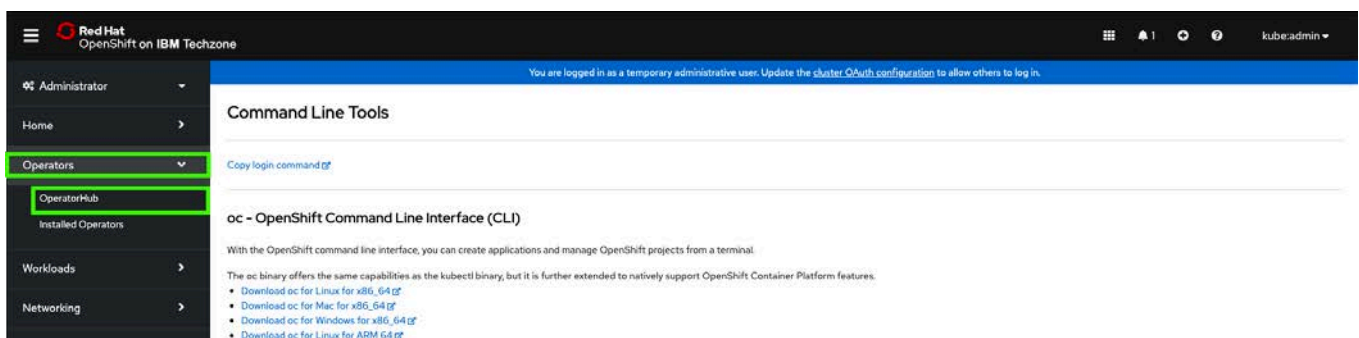
```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-cert-manager-catalog
  namespace: openshift-marketplace
spec:
  displayName: ibm-cert-manager-4.2.7
  grpcPodConfig:
    securityContextConfig: restricted
  image: icr.io/cpopen/ibm-cert-manager-operator-
catalog@sha256:4dcf4ace4b5f166f83b31063f7e6404dbf78d8e98a9d4fcf52fedf57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

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

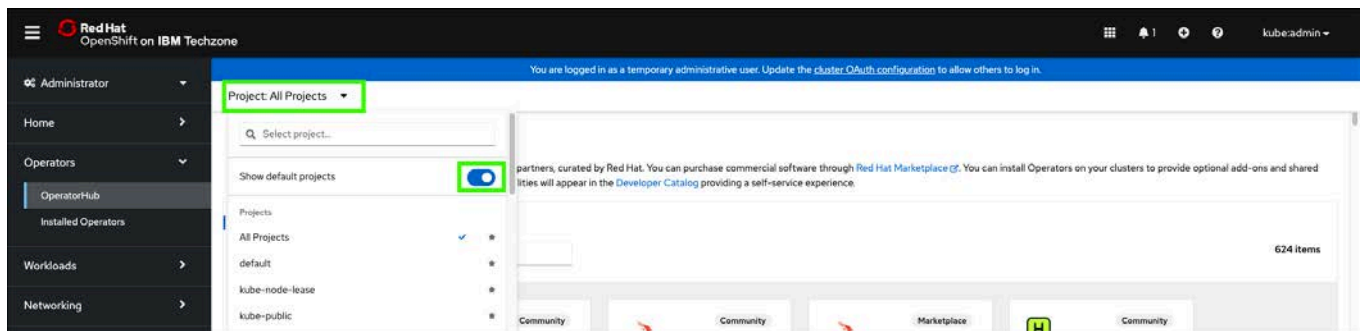
```
oc apply -f catalogCertManager.yaml
```

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

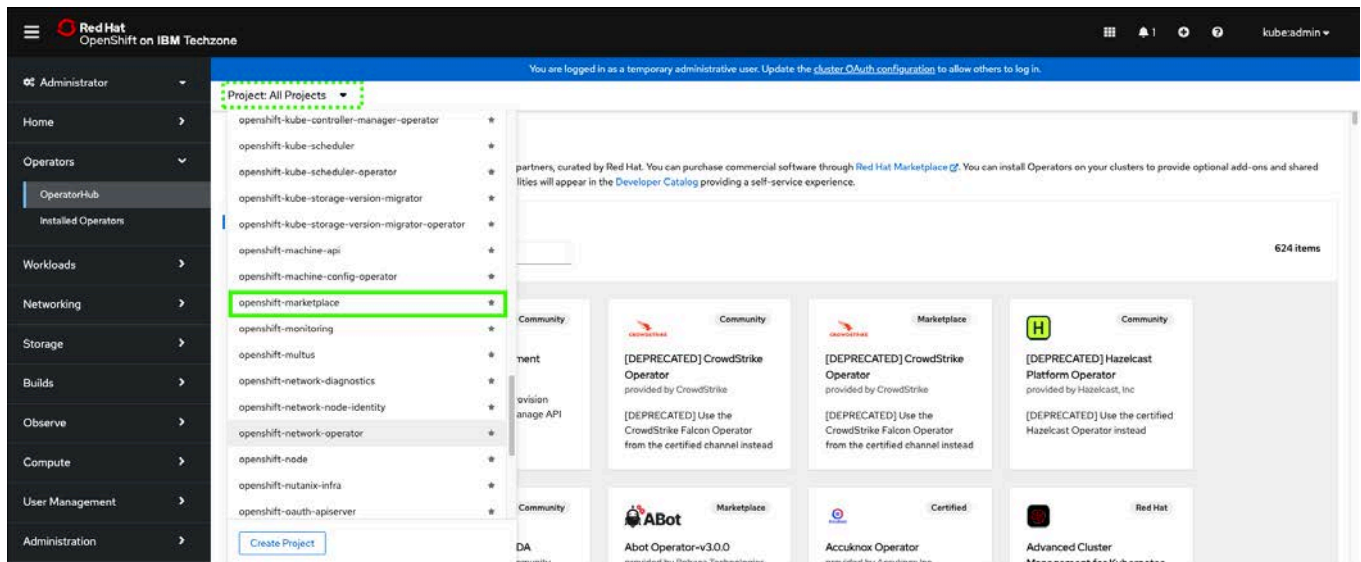
3. In the OpenShift web console, click **Operators** and then select **OperatorHub**.



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



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



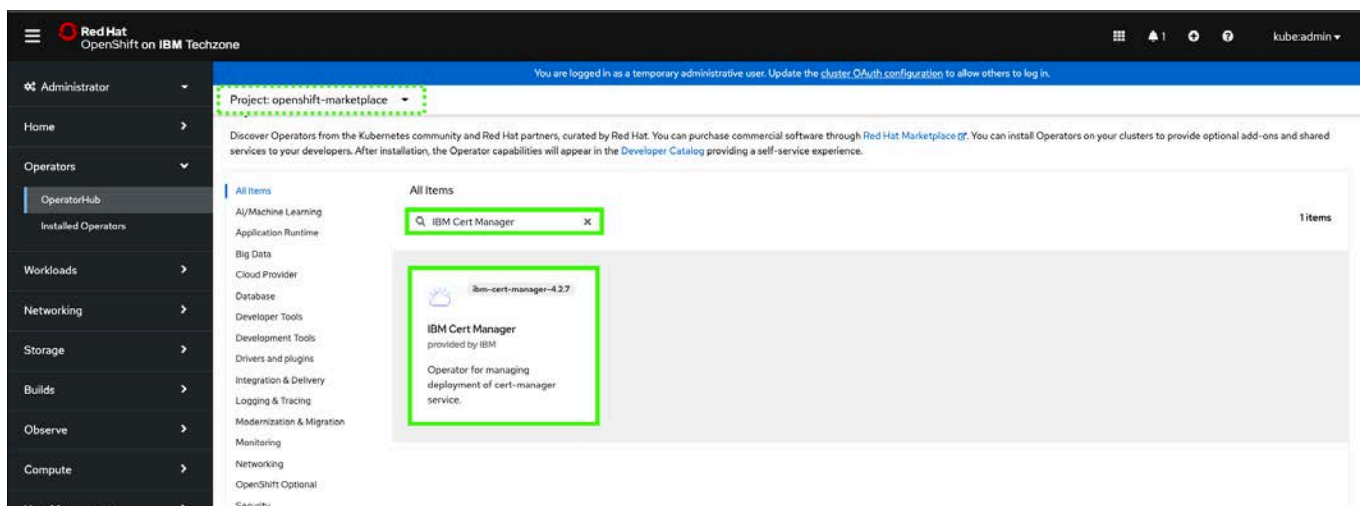
6. Enter **IBM Cert Manager** in the search field and then click the **IBM Cert Manager** tile.



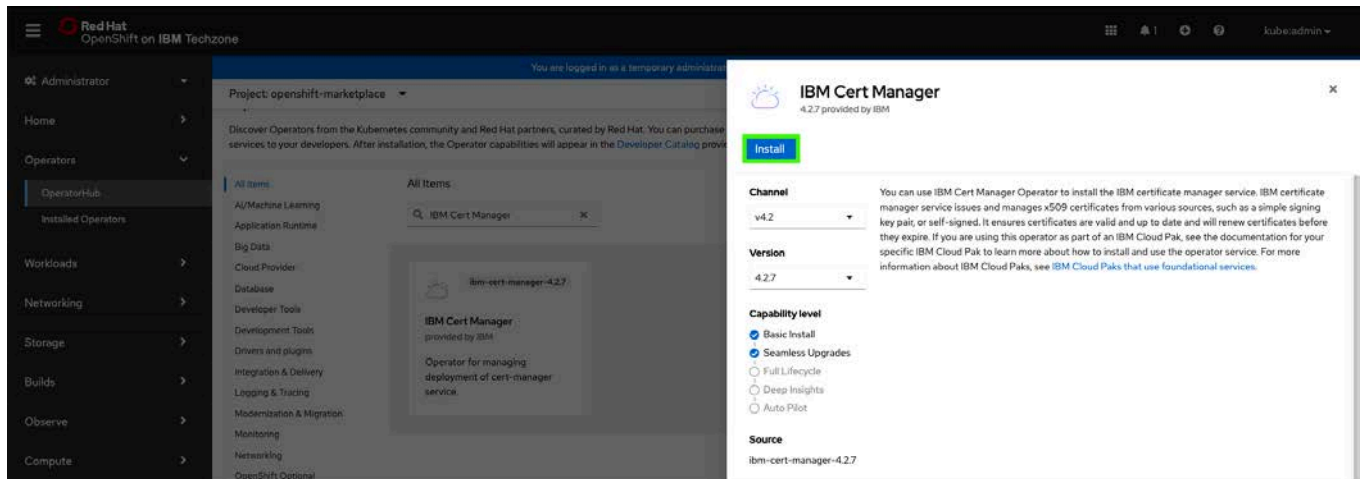
Be patient.

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

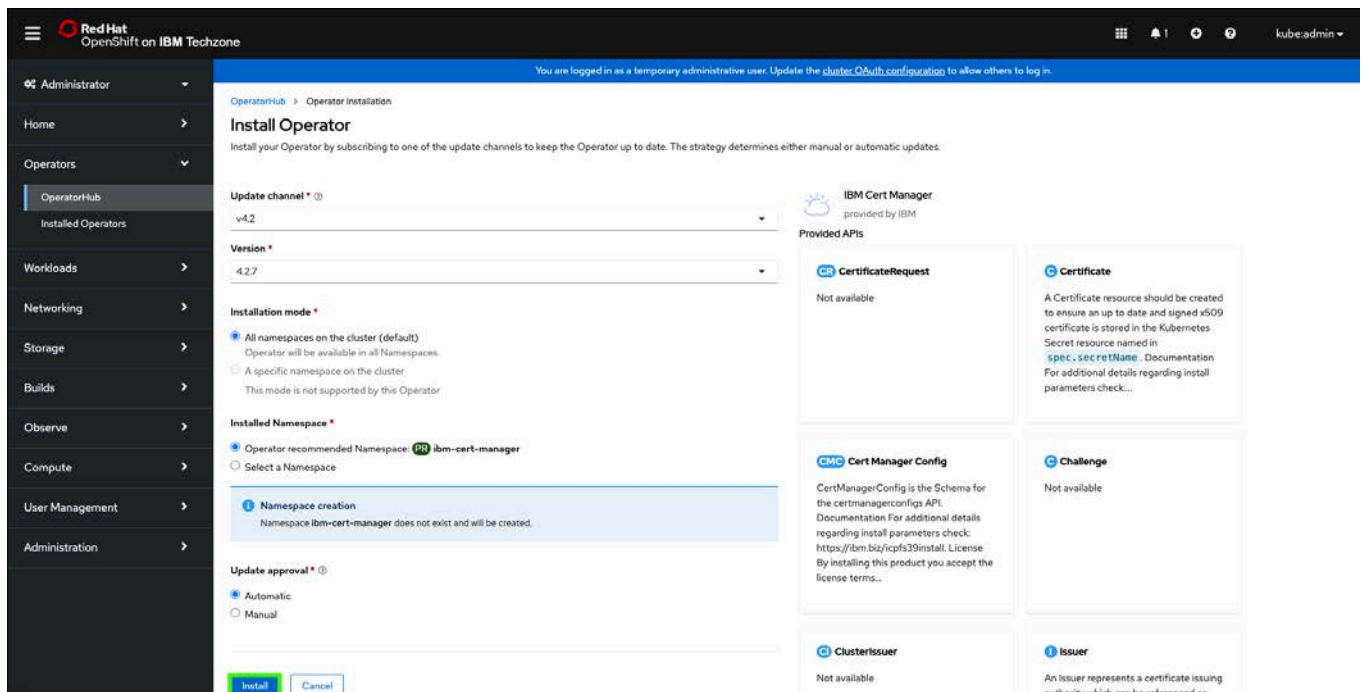
Note: The current version of the operator may differ than shown in the image below. Select the most current version.



7. Click **Install**.



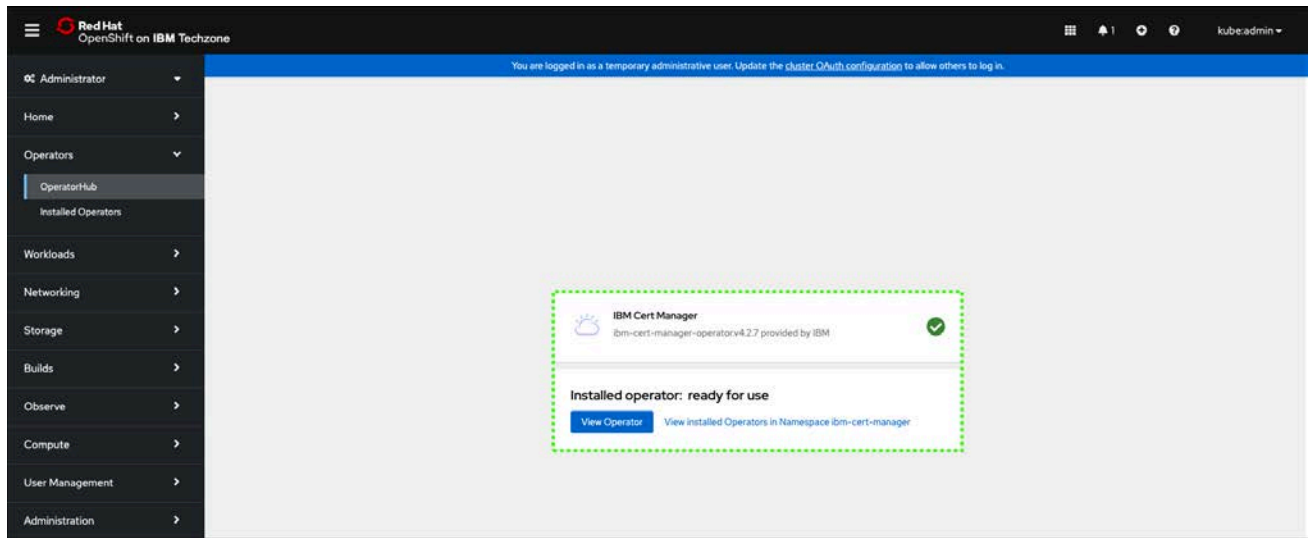
8. Keep the default settings and click **Install**.





Do not continue until...

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



Install the watsonx Assistant for Z Operator (for OpenSearch)

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

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

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

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

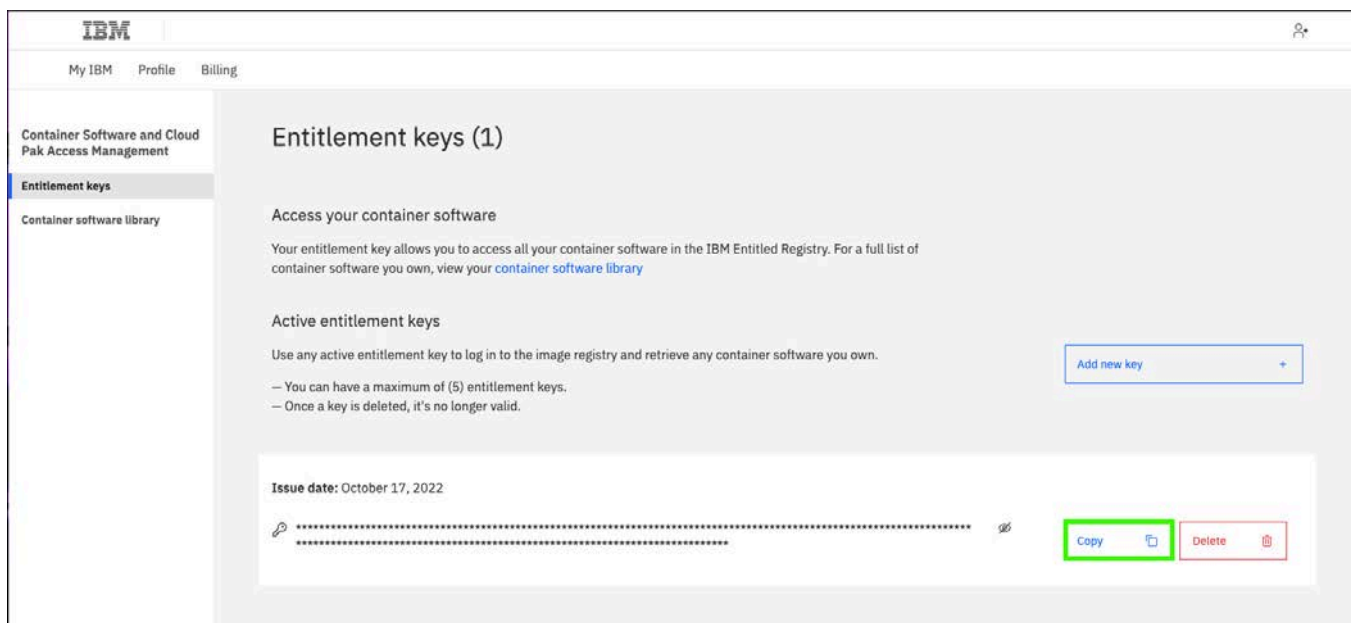
To create or retrieve your existing entitlement key, follow the instructions [here](#).

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

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

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

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



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%
```

```
watsonxAssistant - cecuser@p1365-pvm1 ~ - ssh - 135x37
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 --d
ocker-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.

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

catalogSource.yaml

```
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
```

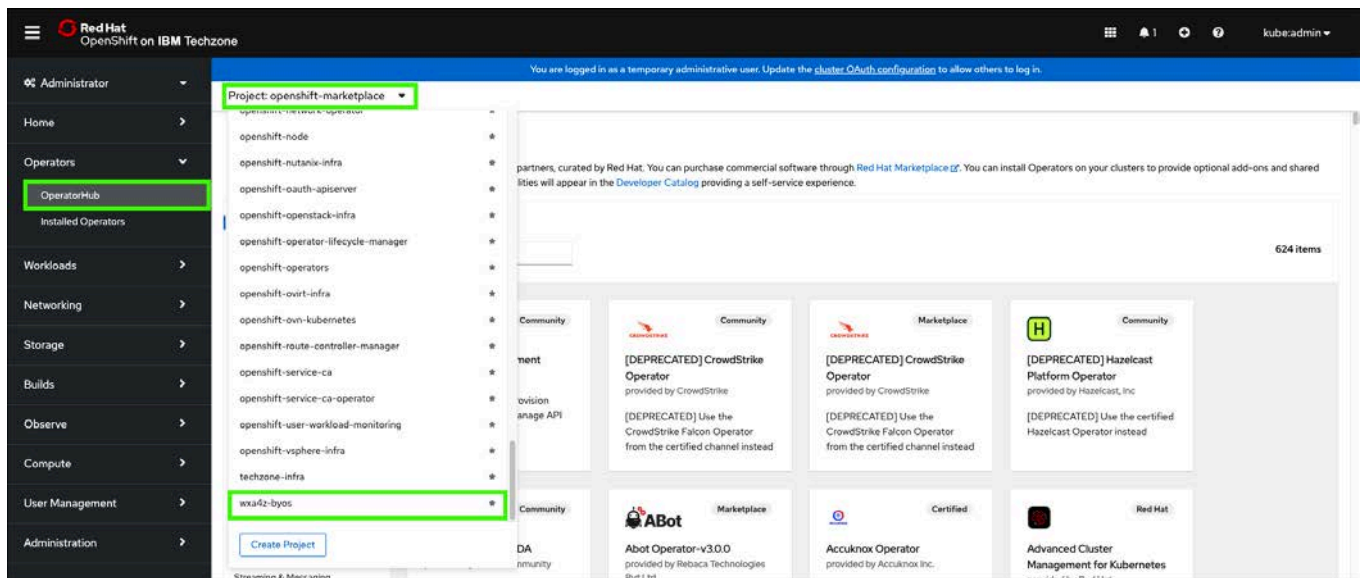
```
oc apply -f catalogSource.yaml
```

```

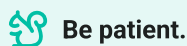
andrewjones@Andrews-MBP watsonxAssistant % oc create namespace wxa4z-byos
namespace/wxa4z-byos created
andrewjones@Andrews-MBP watsonxAssistant % export IBM_CS_ENT_KEY=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJJQk0gTWYya2V0cGxhY2U1LCJpYXQiOiJlZnJyYmTKODAsImpaSI6IjNkOWUyMzZjZTAzMzQzMzZhNTJhYTkzMWVhZDQyMDRlIn0.4M3XRd4XzKhMSOKFN34uKvCwZ6SnEAD030el_11A2xY
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.operators.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.



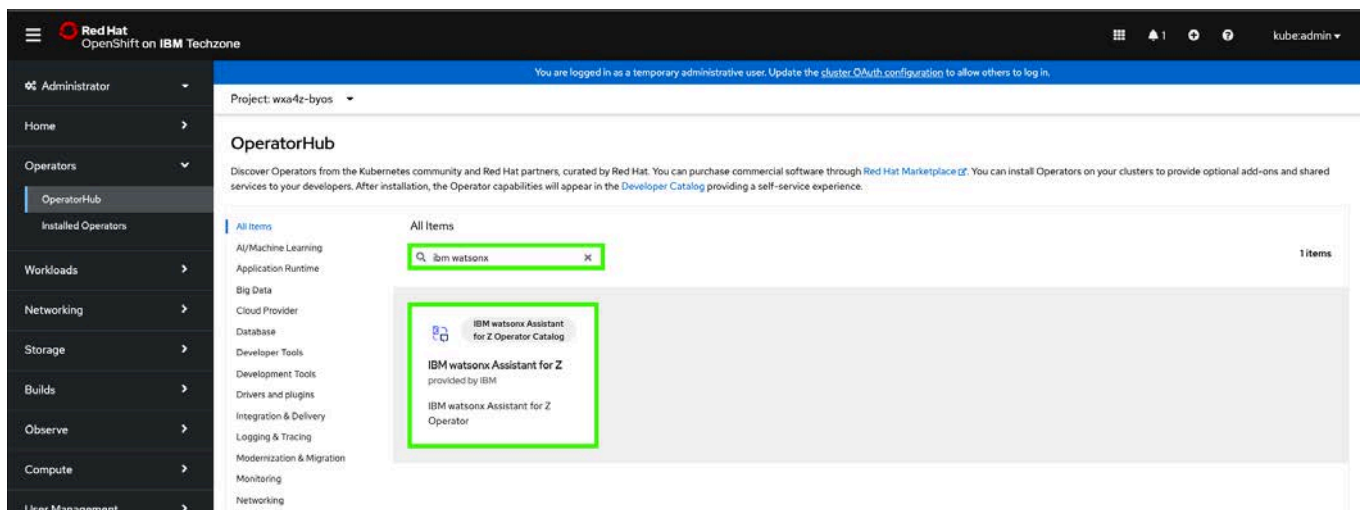
9. Enter **ibm watsonx** in the search field and click the **IBM watsonx Assistant for Z Operator Catalog** tile.



Be patient.

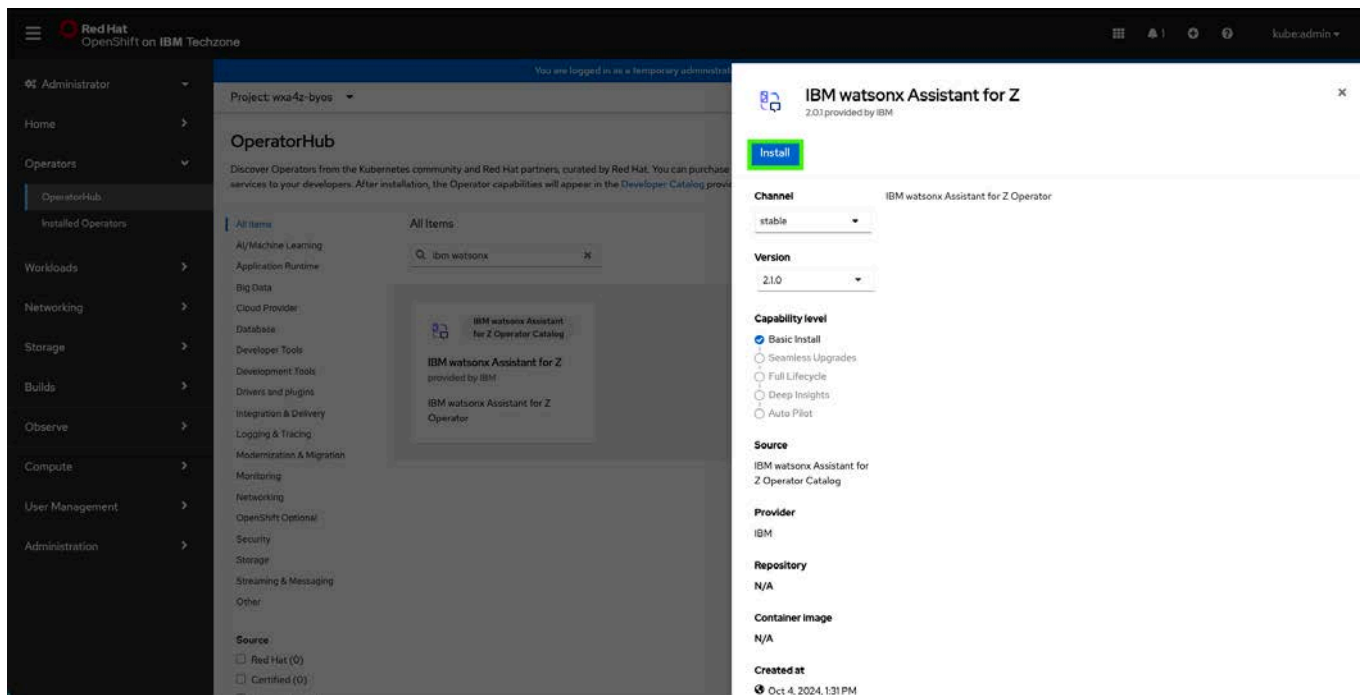
It may take a minute or two for the **IBM watsonx Assistant for Z Operator Catalog** tile to appear. Reload the browser page if the operator is not listed.

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

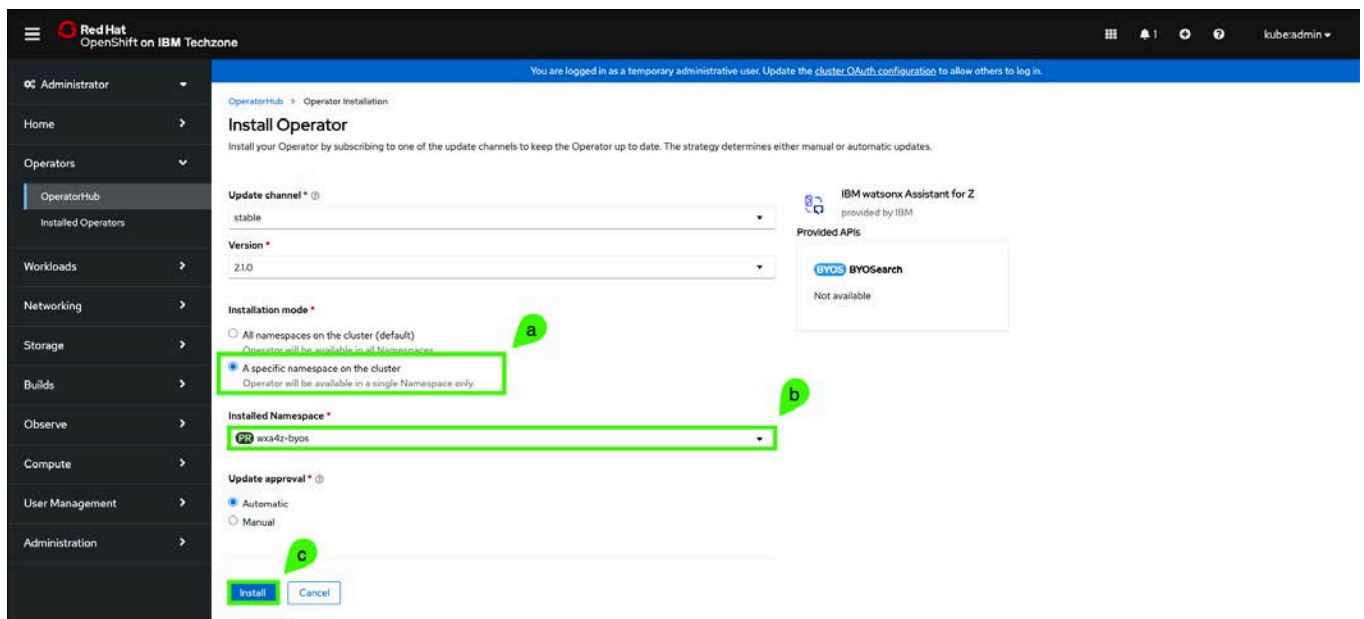


10. Click **Install**.

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



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



Red Hat
OpenShift on IBM Techzone

☰

Administrator

Home

Operators

OperatorHub

Installed Operators

Workloads

Networking

Storage

Builds

Observe

Compute

User Management

Administration

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-operator/v2.0.1 provided by IBM

Installed operator: ready for use

View Operator View installed Operators in Namespace wxa4z-byos

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

The screenshot shows the OpenShift console interface. On the left is a dark sidebar with navigation links: Administrator, Home, Operators, Workloads, Pods (selected), Deployments, DeploymentConfigs, StatefulSets, and Secrets. The main header area is blue and contains the Red Hat OpenShift on IBM Techzone logo, a user profile icon, and 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, the 'Pods' page for the 'wxax4z-byos' project is displayed. It includes a filter dropdown, a search bar, and a table of pods. The table has columns: Name, Status, Ready, Restarts, Owner, Memory, CPU, and Created. Three pods are listed. The first pod is in a 'Completed' state. The second and third pods are in a 'Running' state and are highlighted with green dashed boxes. The second pod is 'ibm-wxa4z-operator-catalog-q9n5m' and the third is 'ibm-wxa4z-operator-controller-manager-7c7898d7d4-87htm'.

Name	Status	Ready	Restarts	Owner	Memory	CPU	Created
1st26367b1ca53def2b0c93acd733e33875cdb4b6c382d5eb4f12032a2a2u6	Completed	0/1	0	1st26367b1ca53def2b0c93acd733e33875cdb4b6c382d5eb4f12032a2a2u6	-	-	Nov 4, 2024, 4:44 PM
ibm-wxa4z-operator-catalog-q9n5m	Running	1/1	0	ibm-wxa4z-operator-catalog	19.0 MiB	0.003 cores	Nov 4, 2024, 4:32 PM
ibm-wxa4z-operator-controller-manager-7c7898d7d4-87htm	Running	2/2	0	ibm-wxa4z-operator-controller-manager-7c7898d7d4	31.0 MiB	0.001 cores	Nov 4, 2024, 4:44 PM

```

andrewjones@Andrews-MBP watsonxAssistant % oc create namespace wxa4z-byos
namespace/wxa4z-byos created
andrewjones@Andrews-MBP watsonxAssistant % export IBM_CS_ENT_KEY=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJJKQk0dTWFya2V0cGxhY2UiLCJpYXQiOiJlZ2NjYmMTk0ODAsImpaSaS6IjnkOWUyMzZjZTAzMDQzMzVhNTJhYTkyMWNmOTcyMDRlIn0.4M3XRd4XzkHMSOKFNJ4ukVcwZ6SNEAOZ03eL_11A2xY
andrewjones@Andrews-MBP watsonxAssistant % oc -n wxa4z-byos create secret docker-registry icr-pull-secret --docker-server=cp.icr.io --d
ocker-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.operators.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 -ttype merge -p '{"imagePullSe
crets": [{"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 %

```

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 your prompt in the next step! For example, the command may return `apps.678e206af501a868a5828e4a.ocp.techzone.ibm.com%`. You do NOT want to include the `%` character which was part of the terminal command prompt.

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.

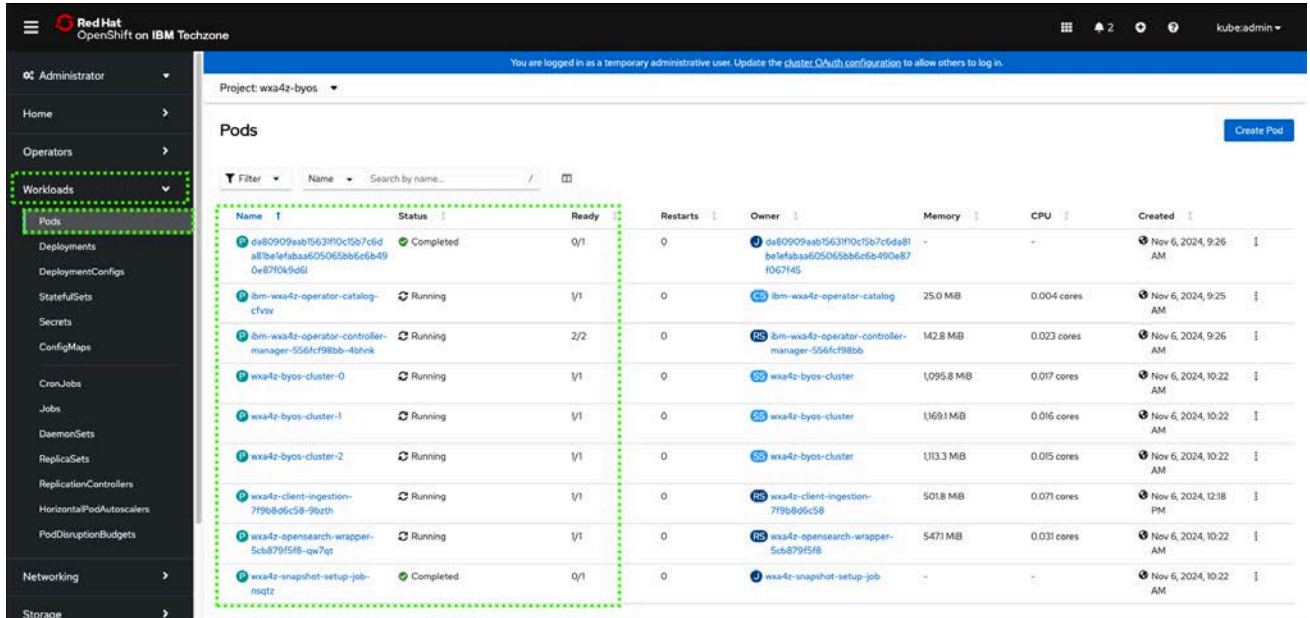
```
oc apply -f byos.yaml
```

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

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

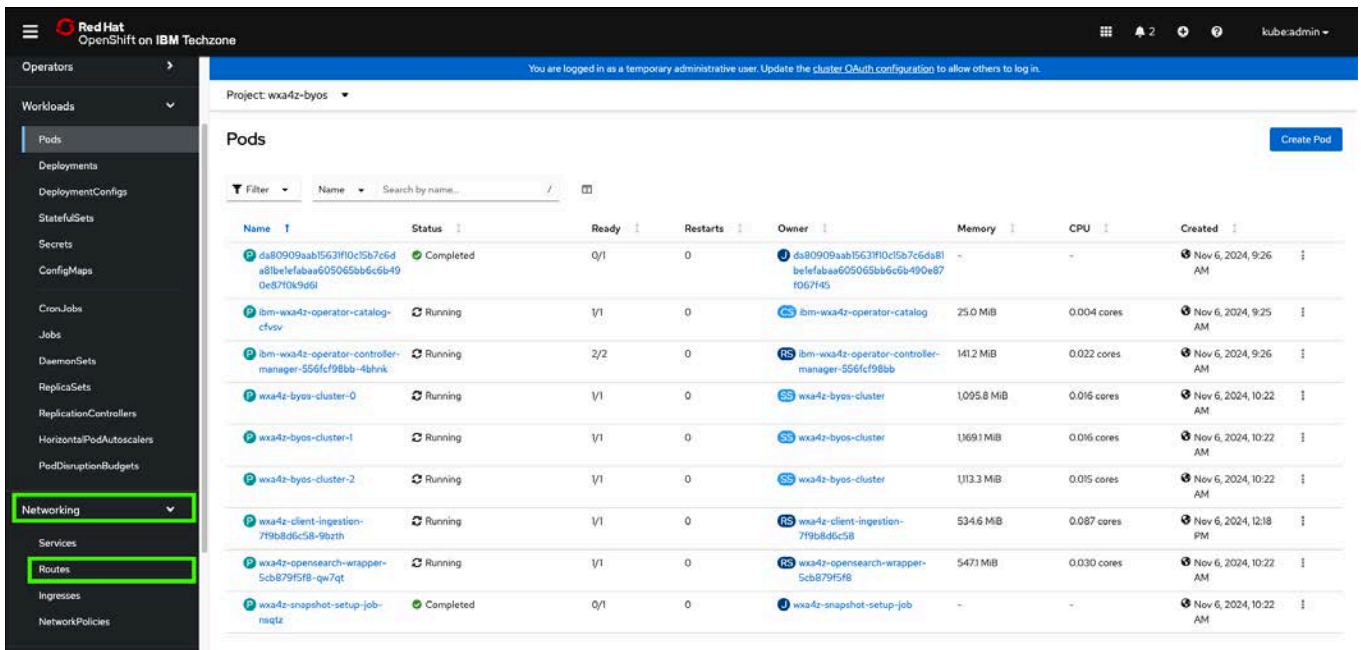
Do not continue until...

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.



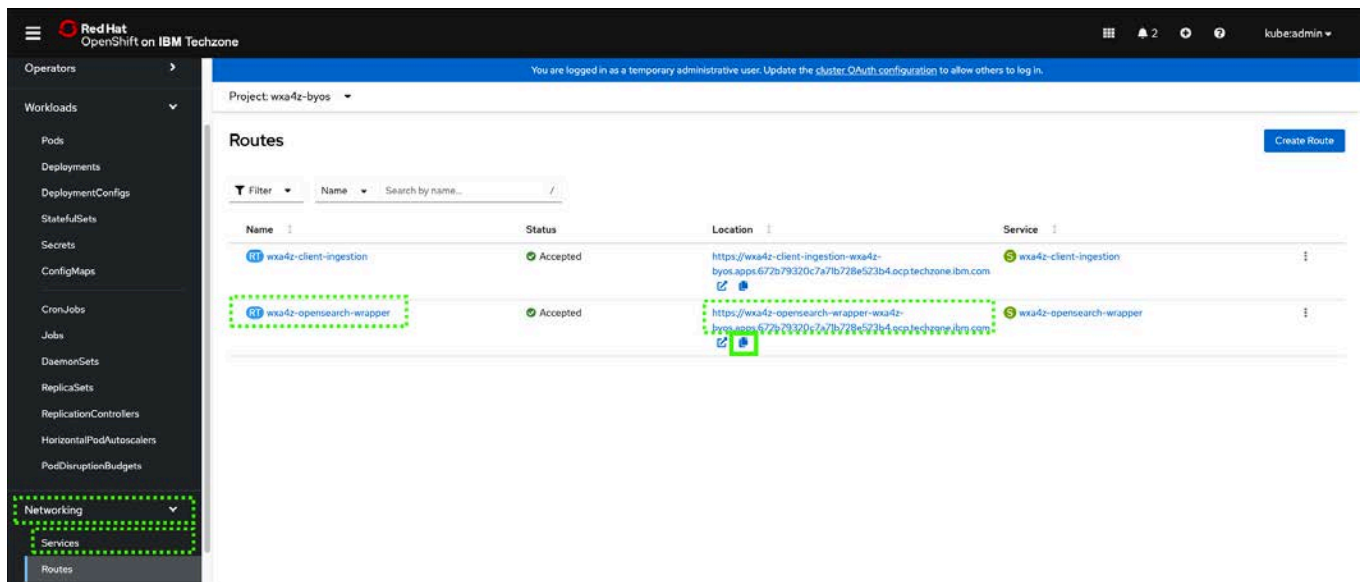
Name	Status	Ready	Restarts	Owner	Memory	CPU	Created
da80909aab1563f10c15b7c6da81a81b1efabaa605065bb6c6b490e87f0K9dgl	Completed	0/1	0	da80909aab1563f10c15b7c6da81a81b1efabaa605065bb6c6b490e87f067f45	-	-	Nov 6, 2024, 9:26 AM
ibm-wxa4z-operator-catalog-cfzv	Running	1/1	0	ibm-wxa4z-operator-catalog	25.0 MiB	0.004 cores	Nov 6, 2024, 9:25 AM
ibm-wxa4z-operator-controller-manager-556fc98bb-4bhnk	Running	2/2	0	ibm-wxa4z-operator-controller-manager-556fc98bb	142.8 MiB	0.023 cores	Nov 6, 2024, 9:26 AM
wxa4z-byos-cluster-0	Running	1/1	0	wxa4z-byos-cluster	1,095.8 MiB	0.017 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-1	Running	1/1	0	wxa4z-byos-cluster	1,691 MiB	0.016 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-2	Running	1/1	0	wxa4z-byos-cluster	1,113.3 MiB	0.015 cores	Nov 6, 2024, 10:22 AM
wxa4z-client-ingestion-7f9b8d6c58-9bztb	Running	1/1	0	wxa4z-client-ingestion-7f9b8d6c58	501.8 MiB	0.071 cores	Nov 6, 2024, 12:18 PM
wxa4z-opensearch-wrapper-5cb879f5f8-qw7qt	Running	1/1	0	wxa4z-opensearch-wrapper-5cb879f5f8	547.1 MiB	0.031 cores	Nov 6, 2024, 10:22 AM
wxa4z-snapshot-setup-job-nsqtz	Completed	0/1	0	wxa4z-snapshot-setup-job	-	-	Nov 6, 2024, 10:22 AM

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



Name	Status	Ready	Restarts	Owner	Memory	CPU	Created
da80909aab1563f10c15b7c6da81a81b1efabaa605065bb6c6b490e87f0K9dgl	Completed	0/1	0	da80909aab1563f10c15b7c6da81a81b1efabaa605065bb6c6b490e87f067f45	-	-	Nov 6, 2024, 9:26 AM
ibm-wxa4z-operator-catalog-cfzv	Running	1/1	0	ibm-wxa4z-operator-catalog	25.0 MiB	0.004 cores	Nov 6, 2024, 9:25 AM
ibm-wxa4z-operator-controller-manager-556fc98bb-4bhnk	Running	2/2	0	ibm-wxa4z-operator-controller-manager-556fc98bb	141.2 MiB	0.022 cores	Nov 6, 2024, 9:26 AM
wxa4z-byos-cluster-0	Running	1/1	0	wxa4z-byos-cluster	1,095.8 MiB	0.016 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-1	Running	1/1	0	wxa4z-byos-cluster	1,691 MiB	0.016 cores	Nov 6, 2024, 10:22 AM
wxa4z-byos-cluster-2	Running	1/1	0	wxa4z-byos-cluster	1,113.3 MiB	0.015 cores	Nov 6, 2024, 10:22 AM
wxa4z-client-ingestion-7f9b8d6c58-9bztb	Running	1/1	0	wxa4z-client-ingestion-7f9b8d6c58	534.6 MiB	0.087 cores	Nov 6, 2024, 12:18 PM
wxa4z-opensearch-wrapper-5cb879f5f8-qw7qt	Running	1/1	0	wxa4z-opensearch-wrapper-5cb879f5f8	547.1 MiB	0.030 cores	Nov 6, 2024, 10:22 AM
wxa4z-snapshot-setup-job-nsqtz	Completed	0/1	0	wxa4z-snapshot-setup-job	-	-	Nov 6, 2024, 10:22 AM

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



Update your assistant with the new BYOS instance route

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

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

The URL should look similar to:

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

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

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

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



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

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

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

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

Troubleshooting

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

Pods have a status of ErrImagePull or ImagePullBackoff

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

The wxa4z-client-ingestion pod does not start

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

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


Installing and using zassist to ingest client documents

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

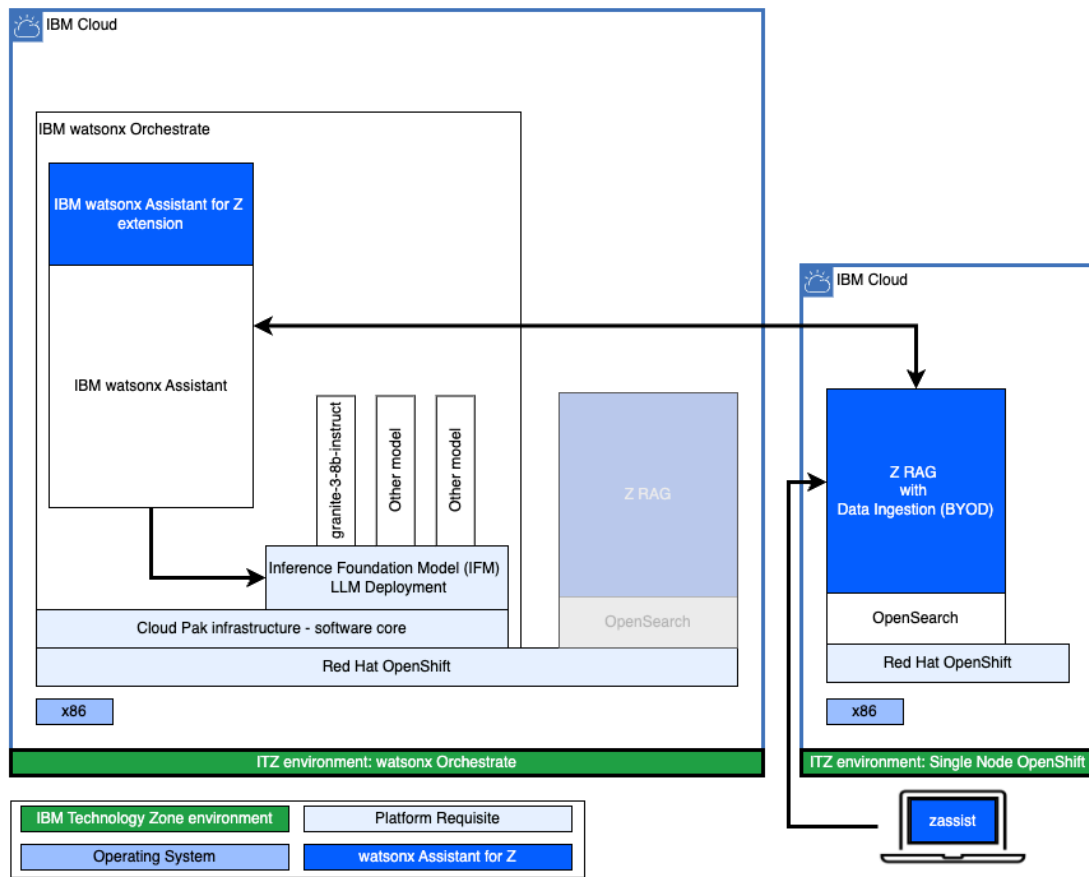
As an example, a client mentioned that their developers often need reference material on company-specific legacy code or company-specific syntax. The users must search through volumes of documentation to find it or look at old code. Also, a need for their operational support group to quickly determine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

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

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

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

Installing and using zassist to ingest client documents



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

Install the zassist utility

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

How do clients get the zassist utility?

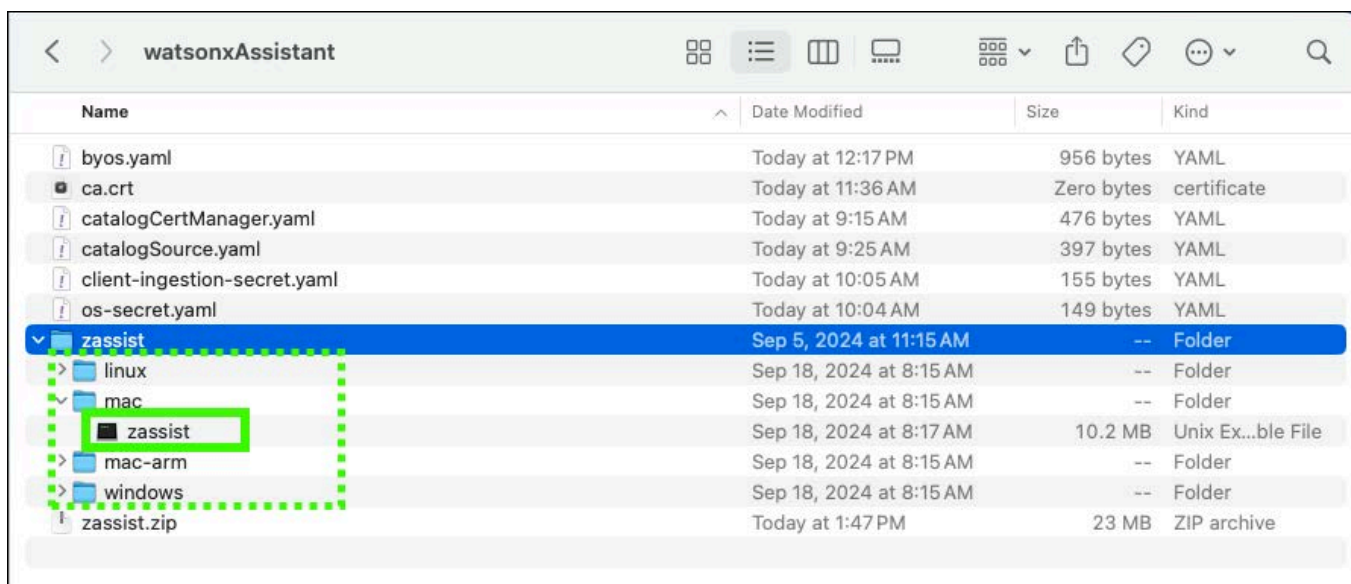
The utility is available to clients through [IBM Passport Advantage](#).

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

<https://ibm.box.com/s/j3nt5iw4fqd5w2jgcqwxnjl8bpvl77>



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



4. Either copy the appropriate **zasslist** 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 zasslist file!

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

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

```
zasslist
```



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

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

Ingest client documentation using zassist

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

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

1. Download the `BYOD.zip` file.

[BYOD.zip](#)

**What is in the sample client documentation?**

Three sample documents are included:

- `Mainframe_COBOL_Error_Codes.pdf`

This is a document containing company-specific mainframe COBOL error codes for their application. Developers within the organization typically review this document to quickly diagnose issues based on the application error codes returned.

- `Mainframe_Operational_Incidents_Log.pdf`

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

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

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

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

Mac OS:

```
export TLS_VERIFY=false
```

Windows OS:

```
set TLS_VERIFY=false
```

Windows OS PowerShell:

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

5. Initialize the `zassist` environment.

```
zassist init
```

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

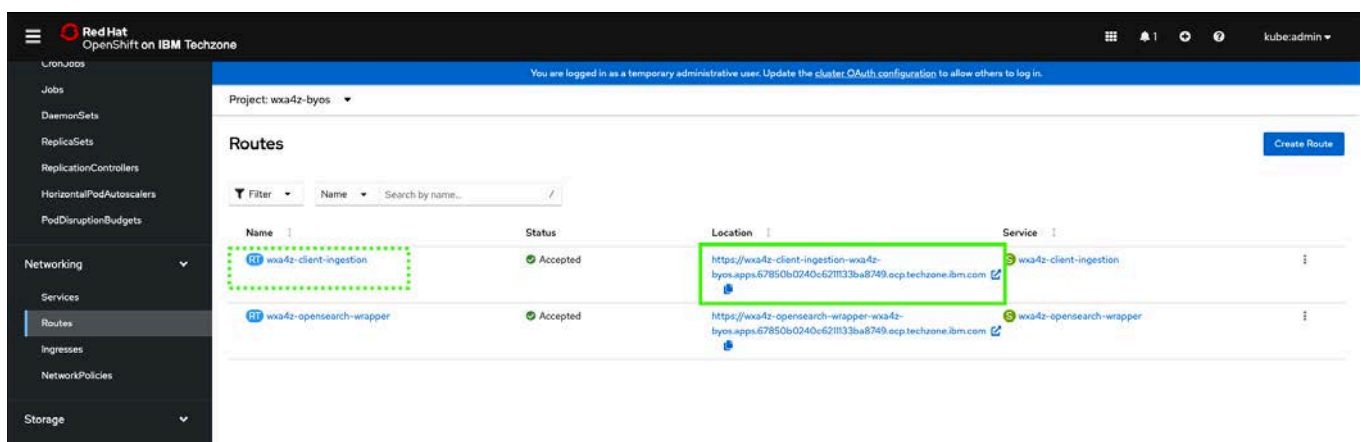
Mac OS:

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

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

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

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



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.

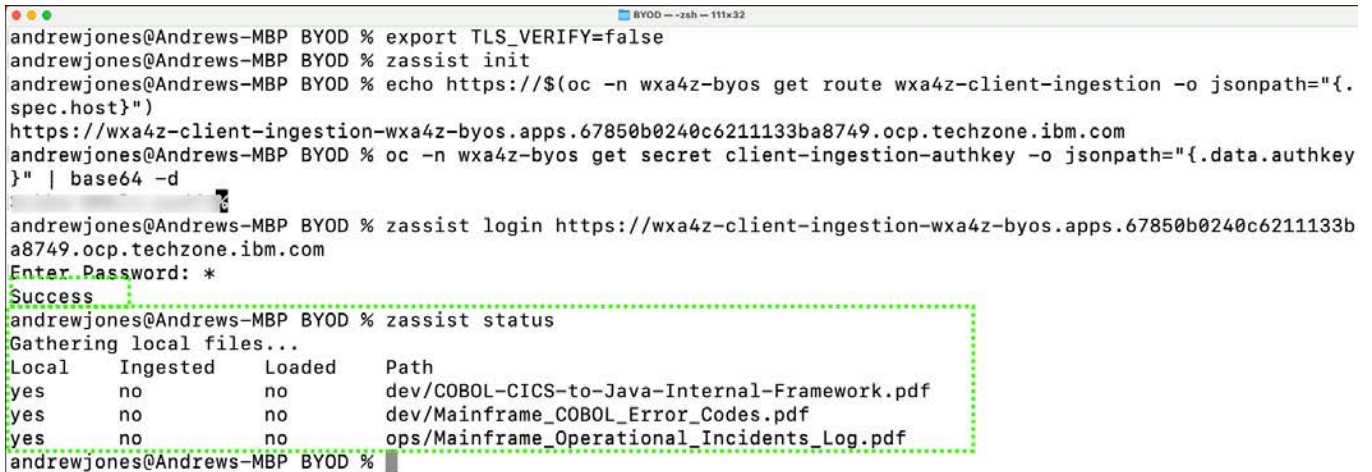
8. Login to your server. Replace `<server url>` with the value from step 6.

```
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.67850b0240c621133ba8749.ocp.techzone.ibm.com
andrewjones@Andrews-MBP BYOD % oc -n wxa4z-byos get secret client-ingestion-authkey -o jsonpath="{.data.authkey}" | base64 -d
a8749.ocp.techzone.ibm.com
andrewjones@Andrews-MBP BYOD % zassist login https://wxa4z-client-ingestion-wxa4z-byos.apps.67850b0240c621133ba8749.ocp.techzone.ibm.com
Enter Password: *
Success
andrewjones@Andrews-MBP BYOD % zassist status
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 % zassistent 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 % zassistent 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 % zassistent load dev
Gathering local files...
Loaded 2 documents into Opensearch database
andrewjones@Andrews-MBP BYOD % zassistent load ops
Gathering local files...
Loaded 1 documents into Opensearch database
andrewjones@Andrews-MBP BYOD % zassistent 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 %

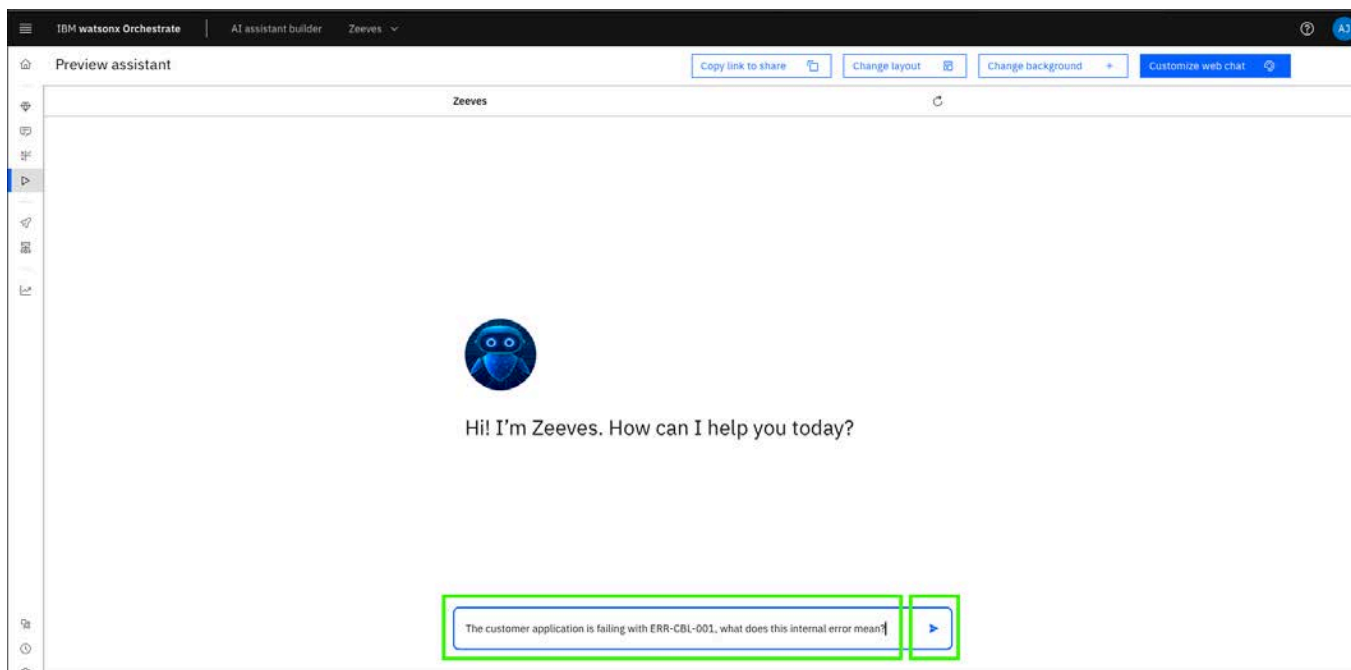
```

Verify that the assistant is using the ingested documents

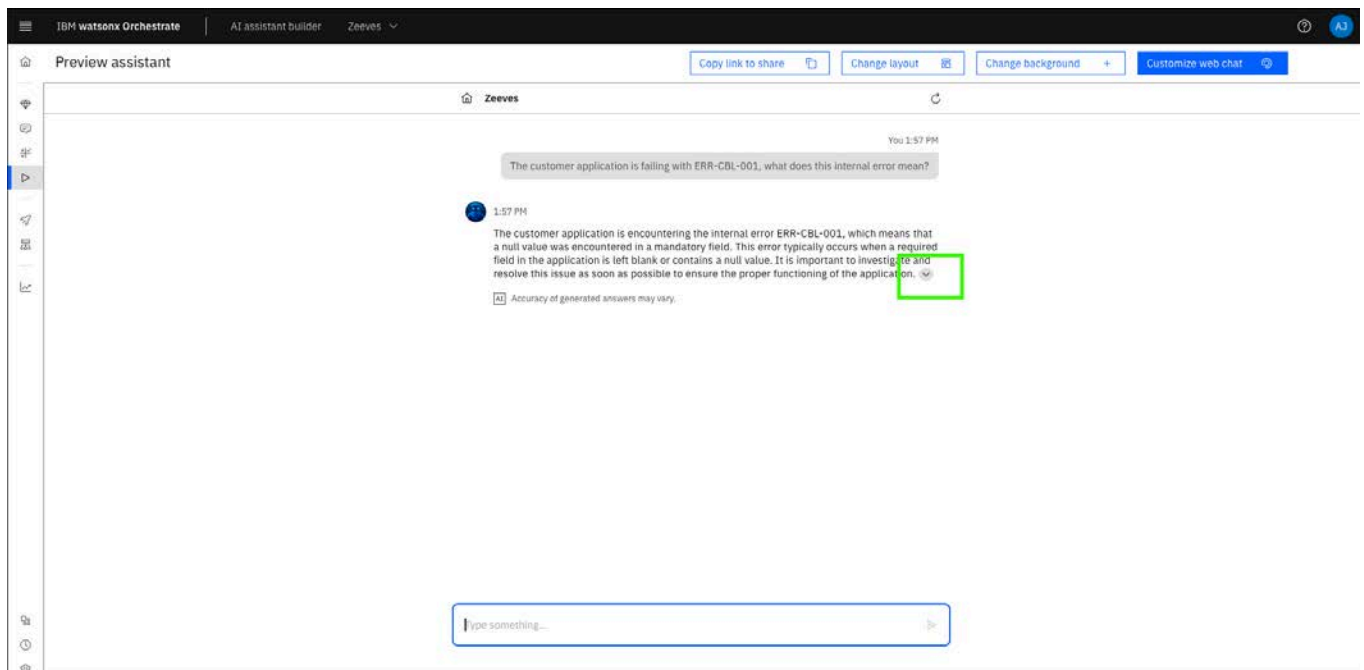
Use the watsonx Orchestrate AI assistant builder to verify your document ingestion.

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

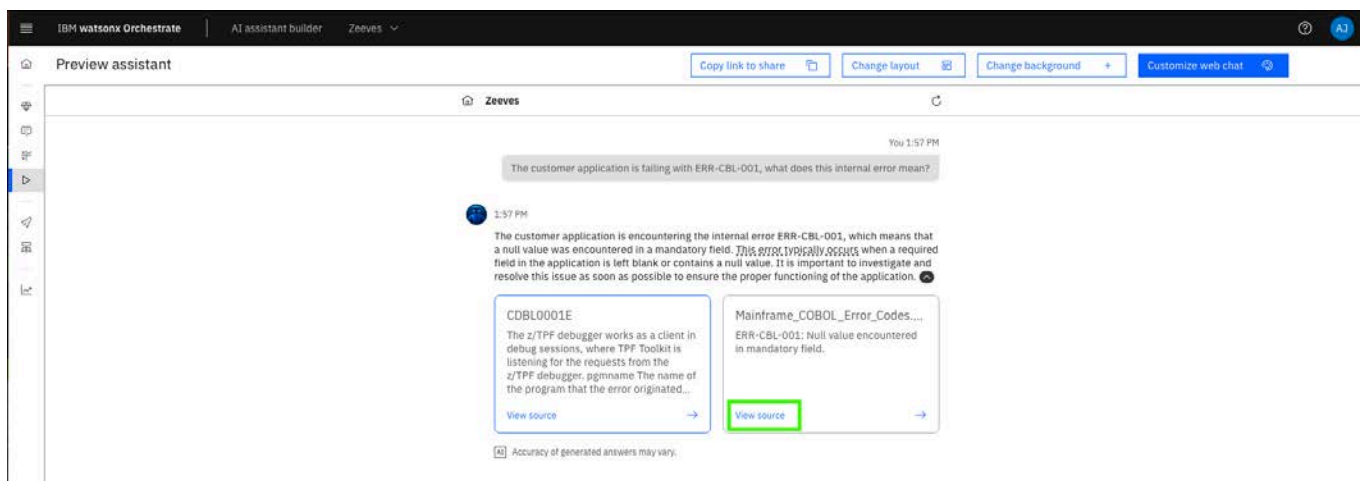
The customer application is failing with ERR-CBL-001, what does this internal error mean?



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



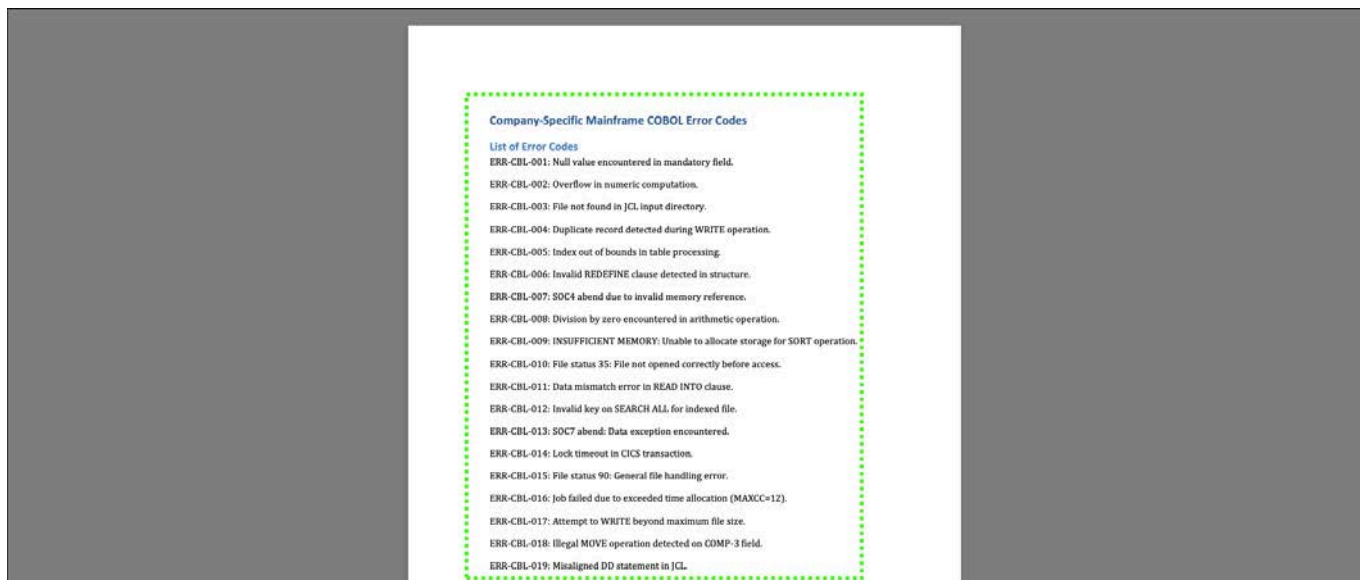
3. Click **View source** for the **Mainframe_COBOL_Error_Codes-...** citation.



Take note of the order of the response citations!

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

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



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

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

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

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

Adjusting the search behavior

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

The screenshot shows the 'Custom service' settings page in the IBM watsonx Assistant interface. The 'Settings' tab is selected. The 'Metadata' field is highlighted with a green dashed border. The 'Search display text' field shows a message: 'I searched my knowledge base, but did not find anything related to your query'.

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": "ss41q8_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_*"
}
```

IBM watsonx Orchestrate | AI assistant builder | Zeeves

Custom service Draft

Settings Instance

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

All

Default filter

Can be anything (plain text, JSON, etc)

Metadata a

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

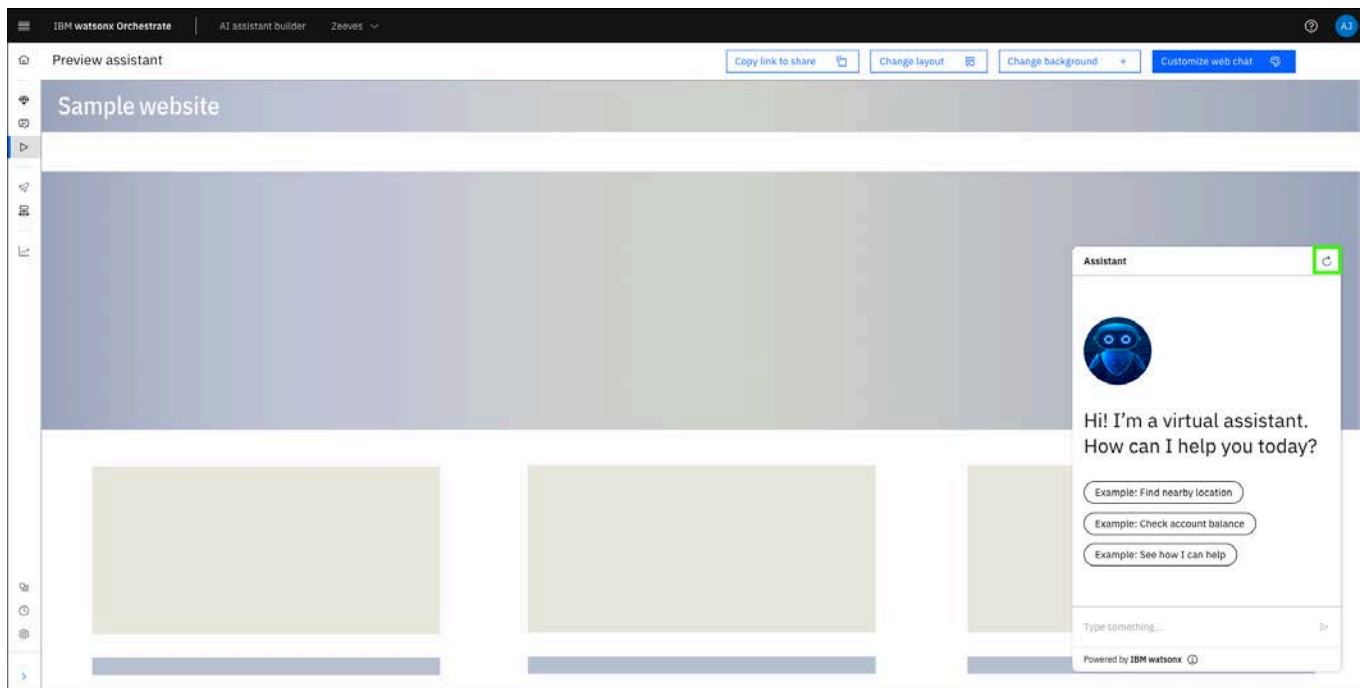
Search display text

Define the text your search will display to the end user

Close Save b

c

2. Hover over the **Home** (🏠) icon and click **Preview**.
3. Click the **Restart conversation** (🔄) icon.



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

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

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



For client pilots

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

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

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

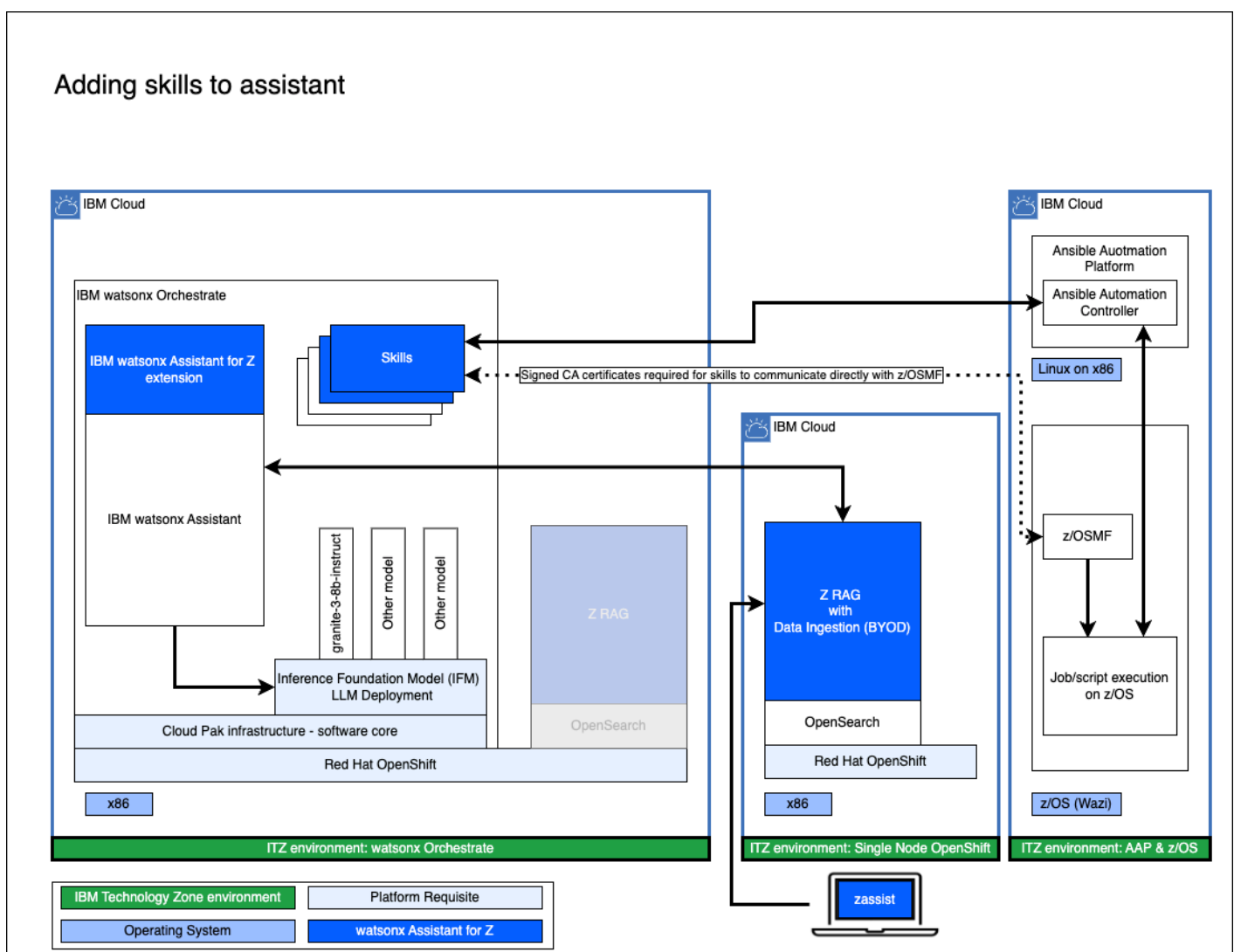
Skills and skill flows

Getting started with skills and actions

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

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

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



Environments

Watsonx Orchestrate

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

Ansible Automation Platform and Wazi as a Service

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

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

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

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

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

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

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

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

Explore Ansible Automation Platform

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

Access the AAP and Wazi as a Service environment

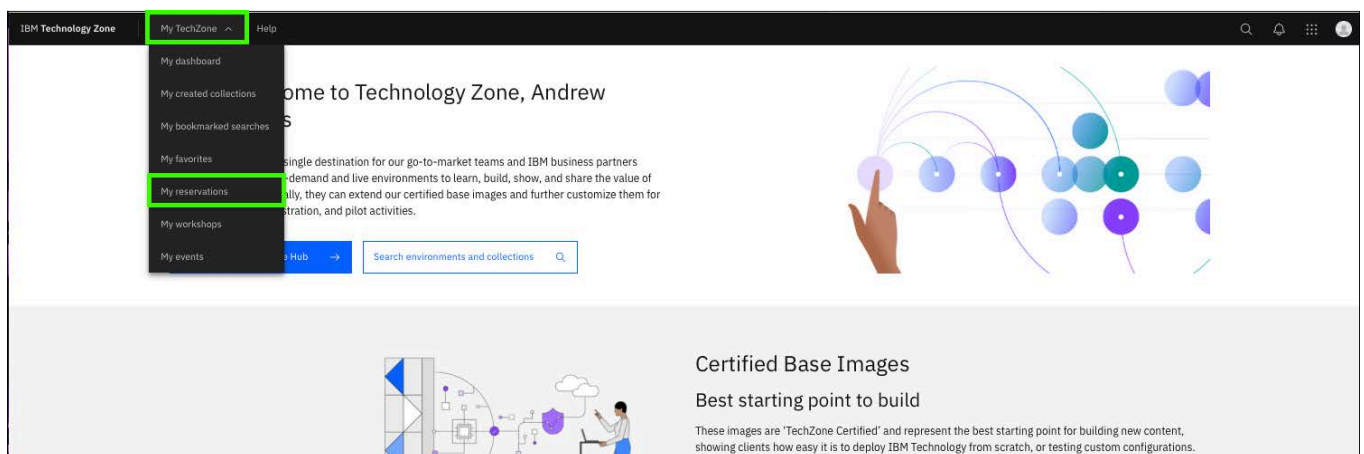


Be sure to record the information as instructed

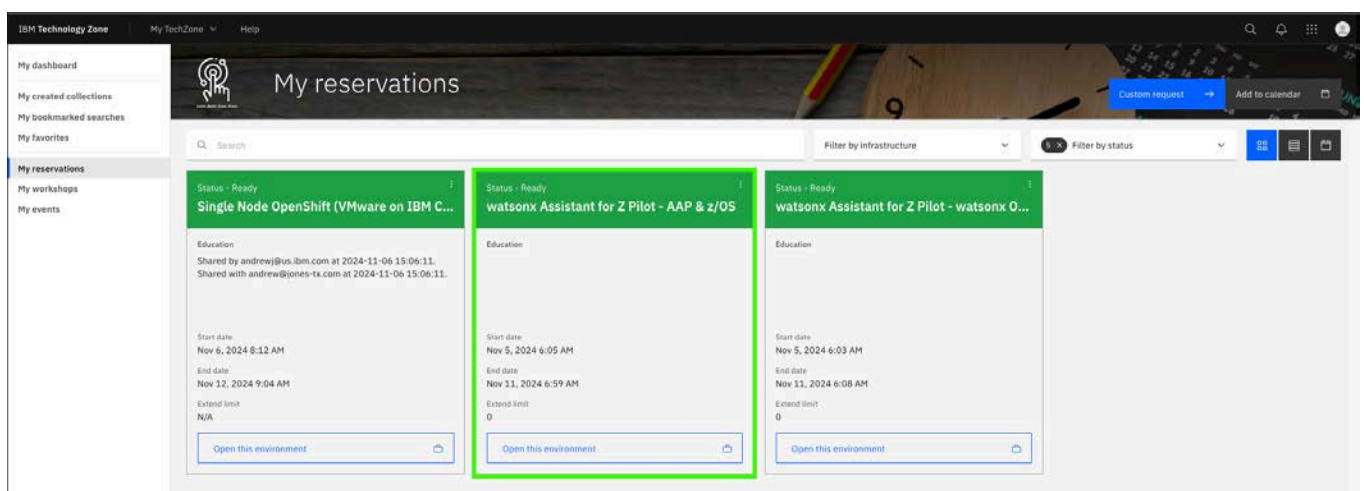
Several of the steps below instruct you to record values from your ITZ reservation. Be sure to do this as they will not only be used in this section, but also in later sections of the lab guide.

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

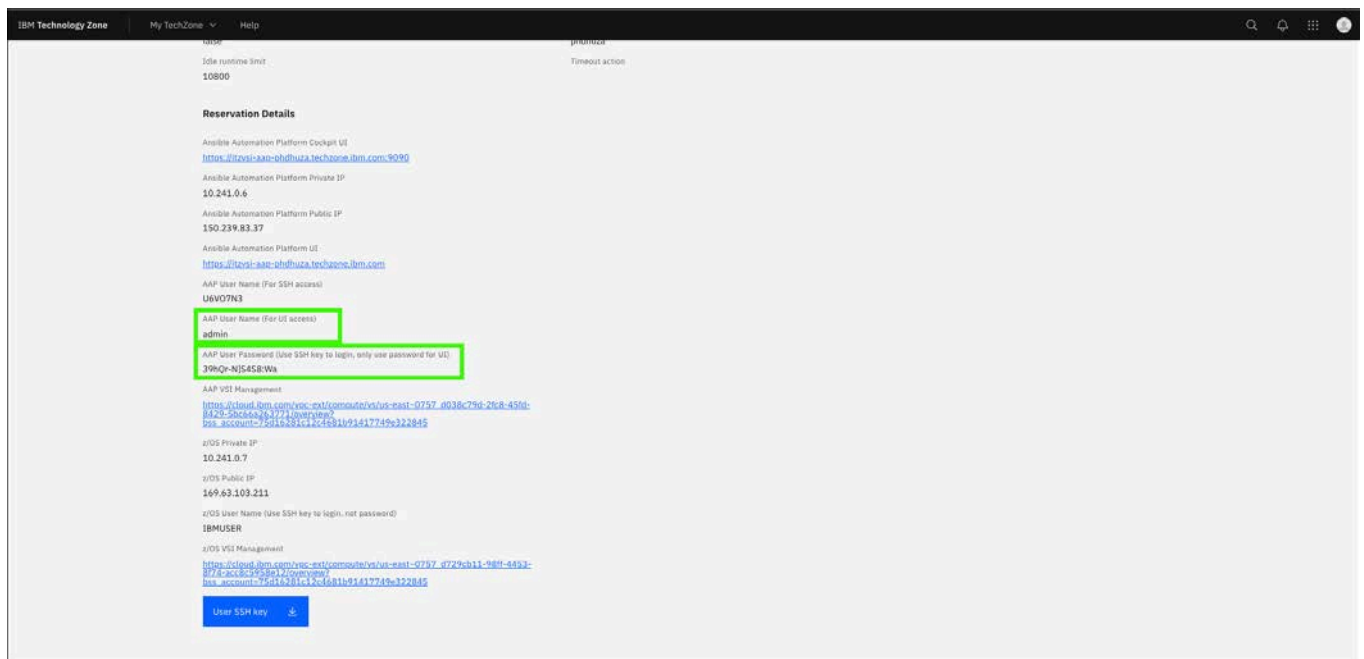
ITZ My reservations



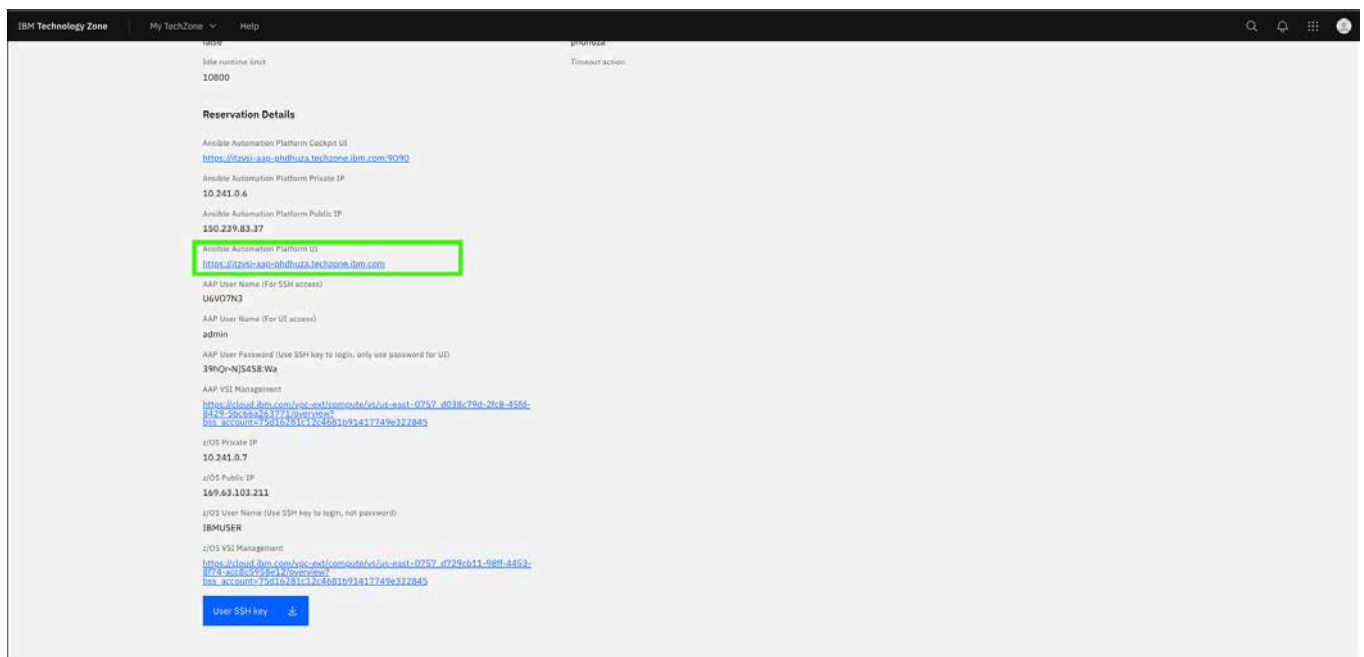
2. Click the **watsonx Assistant for Z Pilot - AAP & z/OS** tile.



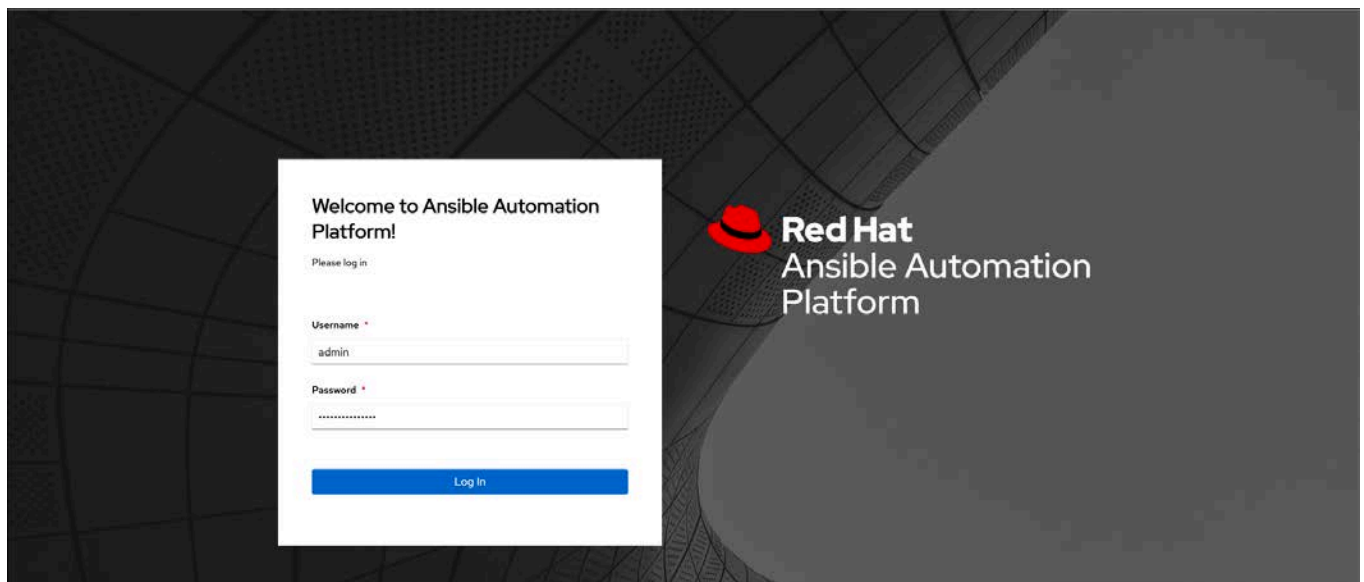
3. Locate and record the **AAP User Name (For UI access)** and **AAP User Password** fields.



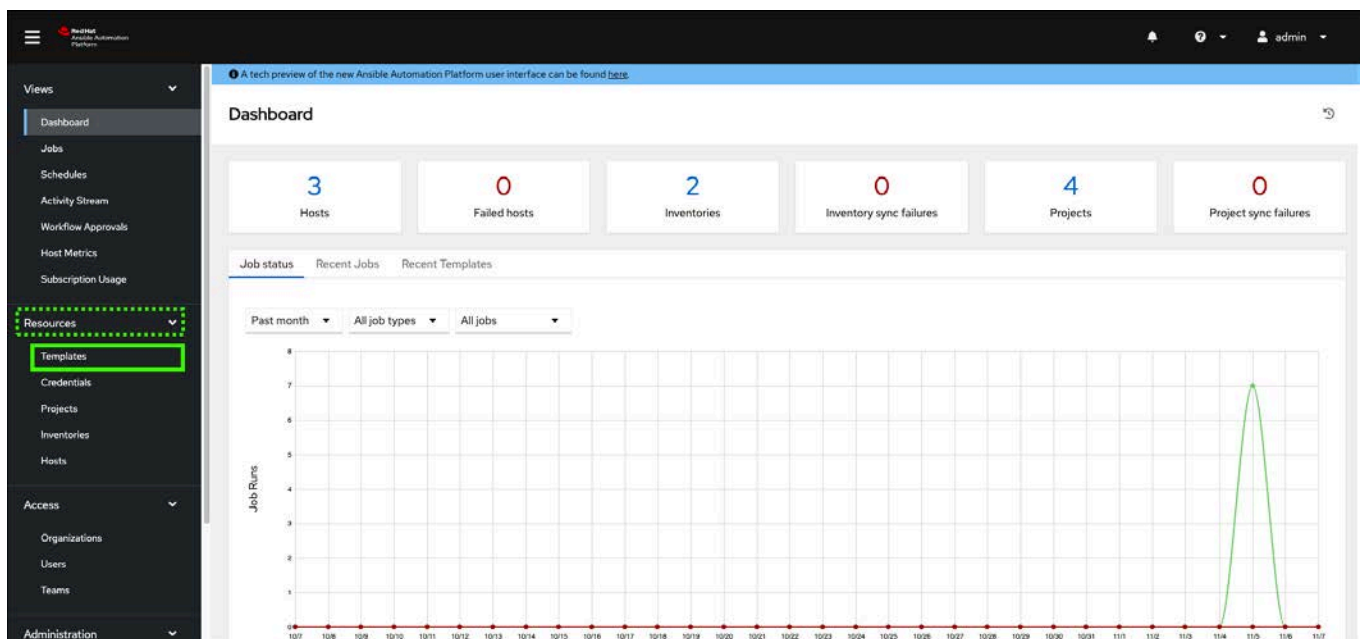
4. Record and then click the **Ansible Automation Platform UI** link.



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



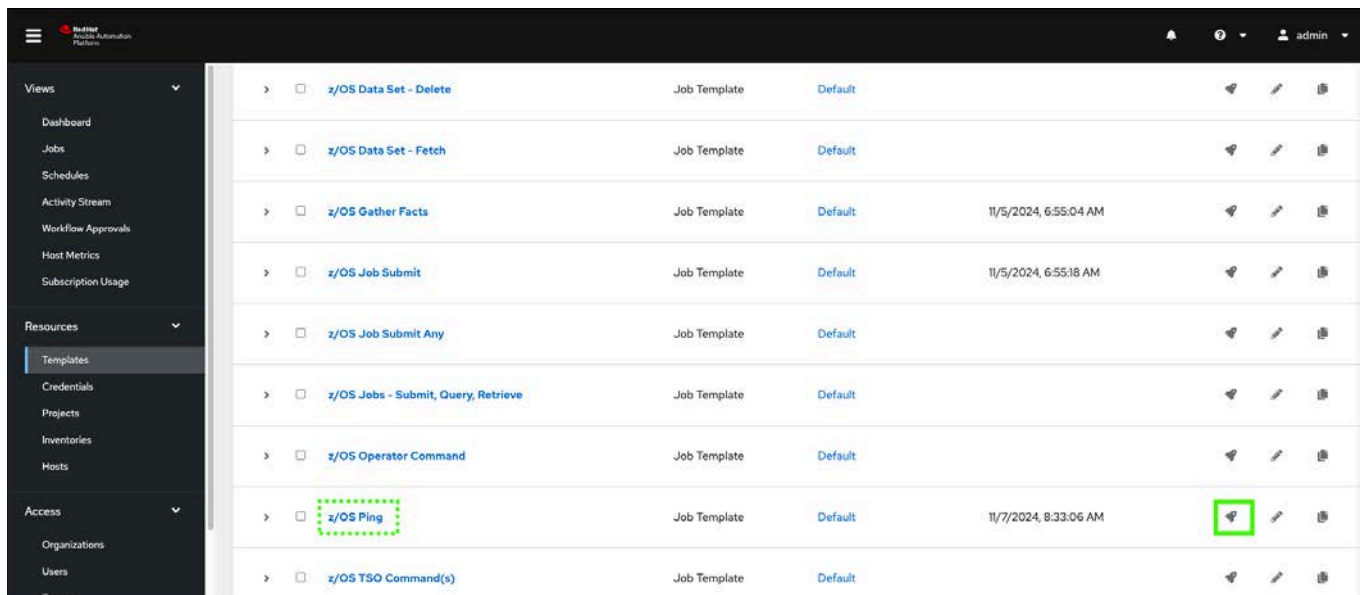
6. Click **Templates** under the **Resources** section.



The AAP instance is preconfigured to the Wazi aaS instance

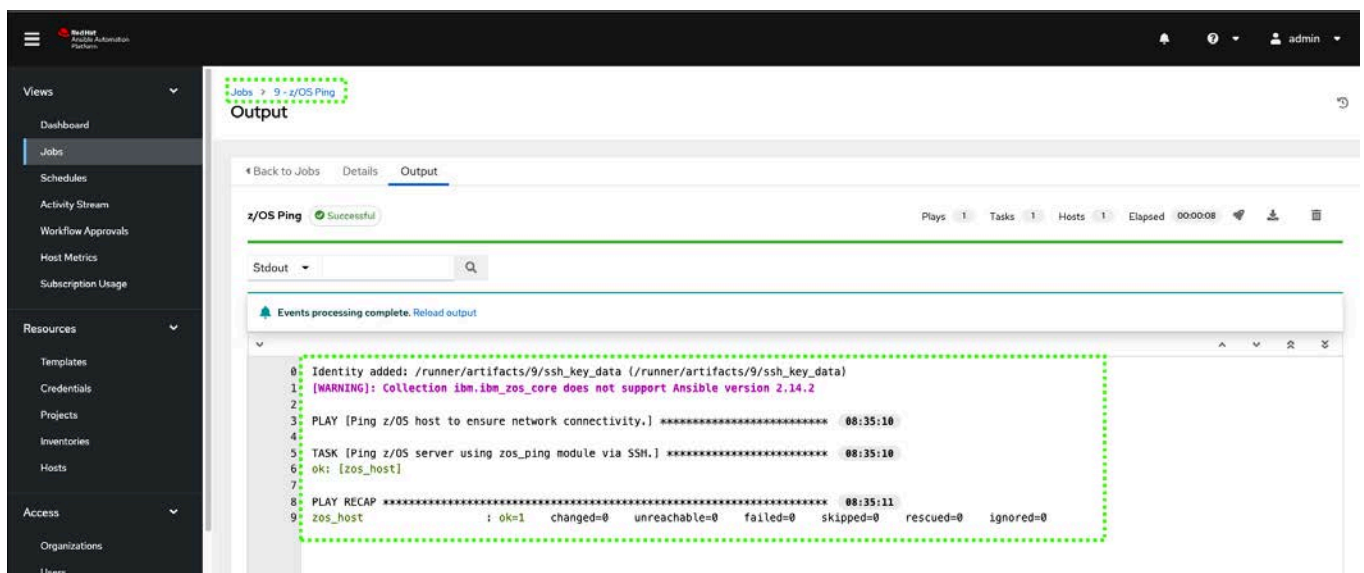
Note that because the AAP instance and the back-end z/OS system are preconfigured, no changes are needed to execute the templates and they will target your provisioned z/OS system by default.

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



Job Template	Default	Last Run	Status
z/OS Data Set - Delete	Default		
z/OS Data Set - Fetch	Default		
z/OS Gather Facts	Default	11/5/2024, 6:55:04 AM	
z/OS Job Submit	Default	11/5/2024, 6:55:18 AM	
z/OS Job Submit Any	Default		
z/OS Jobs - Submit, Query, Retrieve	Default		
z/OS Operator Command	Default		
z/OS Ping	Default	11/7/2024, 8:33:06 AM	Success
z/OS TSO Command(s)	Default		

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



Jobs > z/OS Ping

Output

z/OS Ping Successful Plays: 1 Tasks: 1 Hosts: 1 Elapsed: 00:00:08

Stdout

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](#).

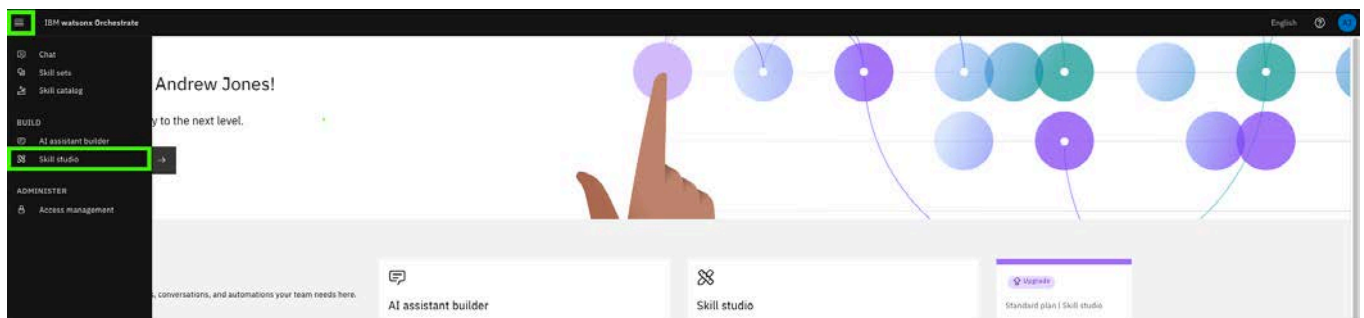
Importing skills from Ansible Automation Platform

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](#).

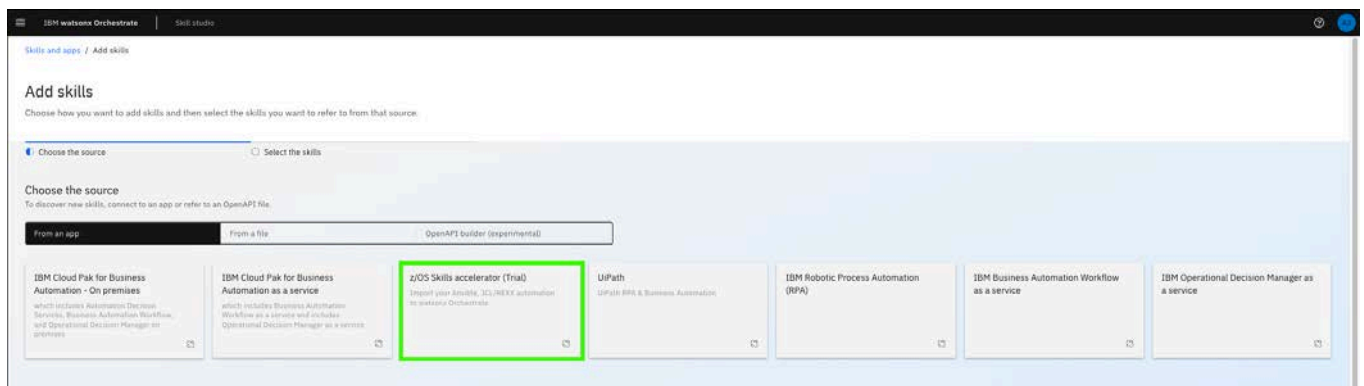
1. Return to your watsonx Orchestrate instance and expand the main menu and click **Skill studio**.



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>

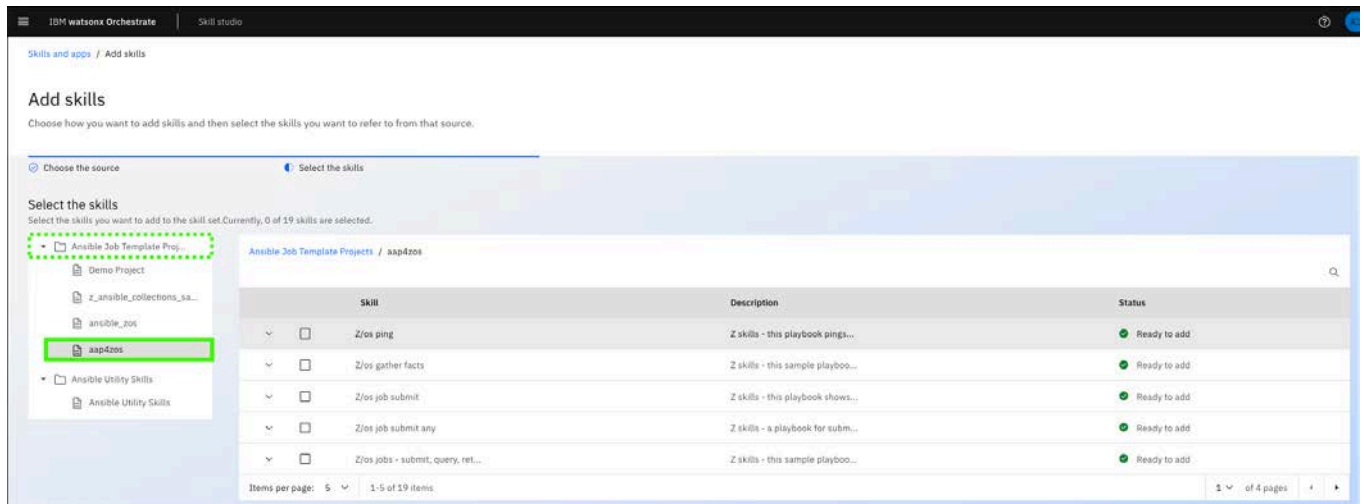
f: Search Pattern: `*` (displays all skills available)

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

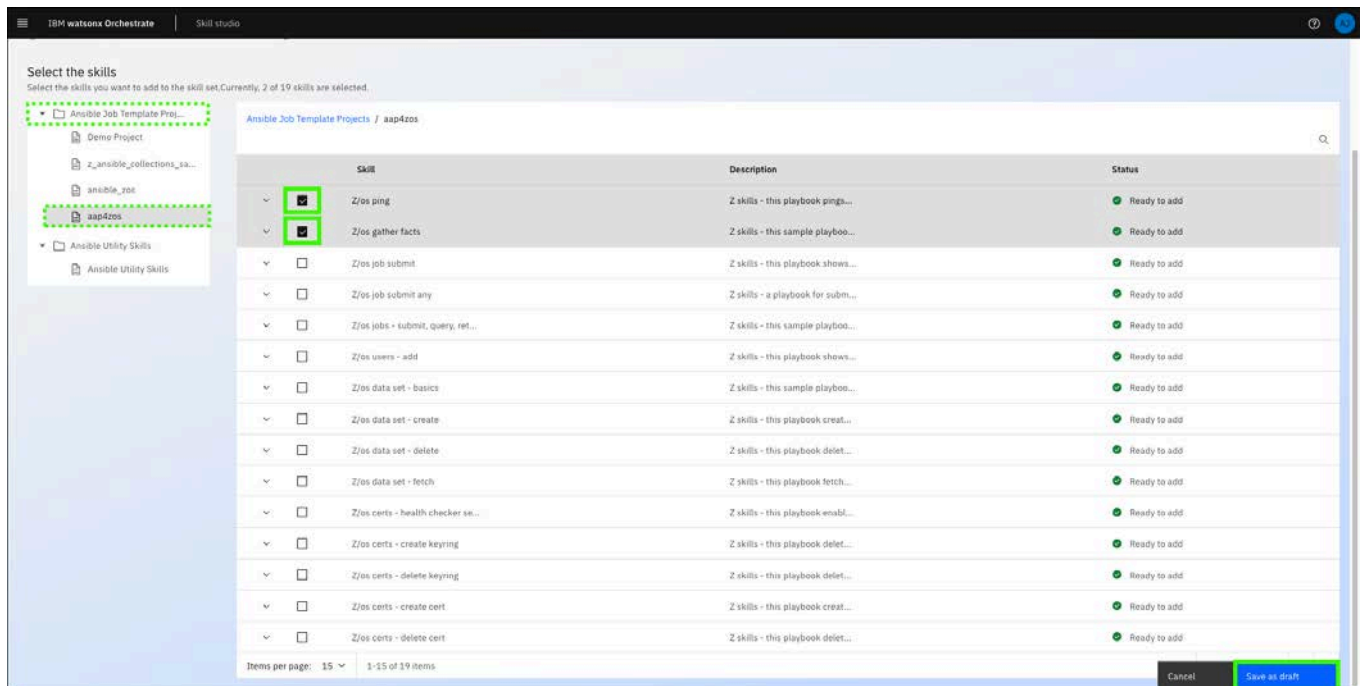


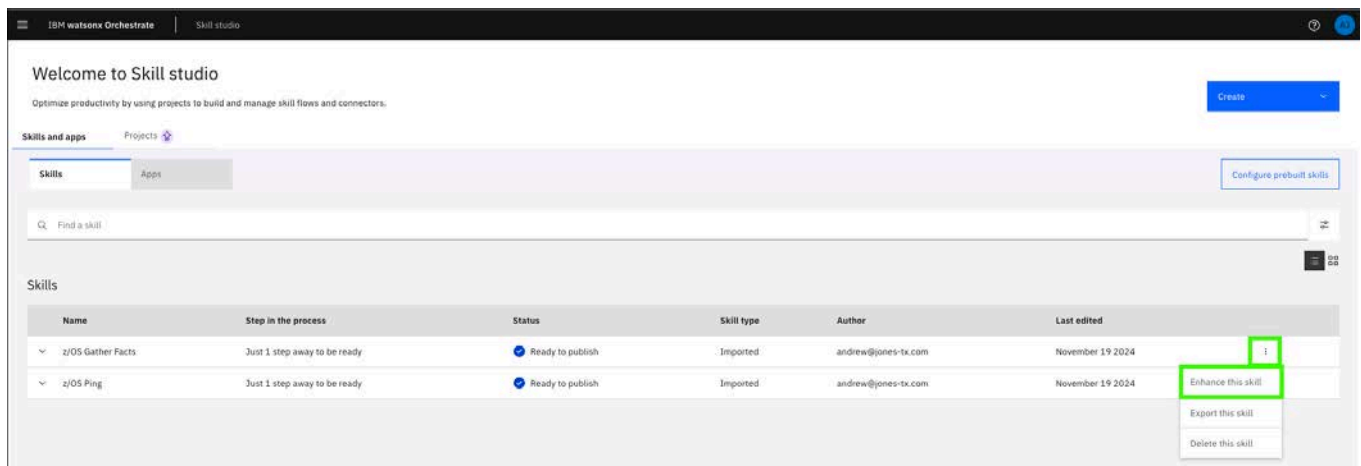
Explore the other available skills

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

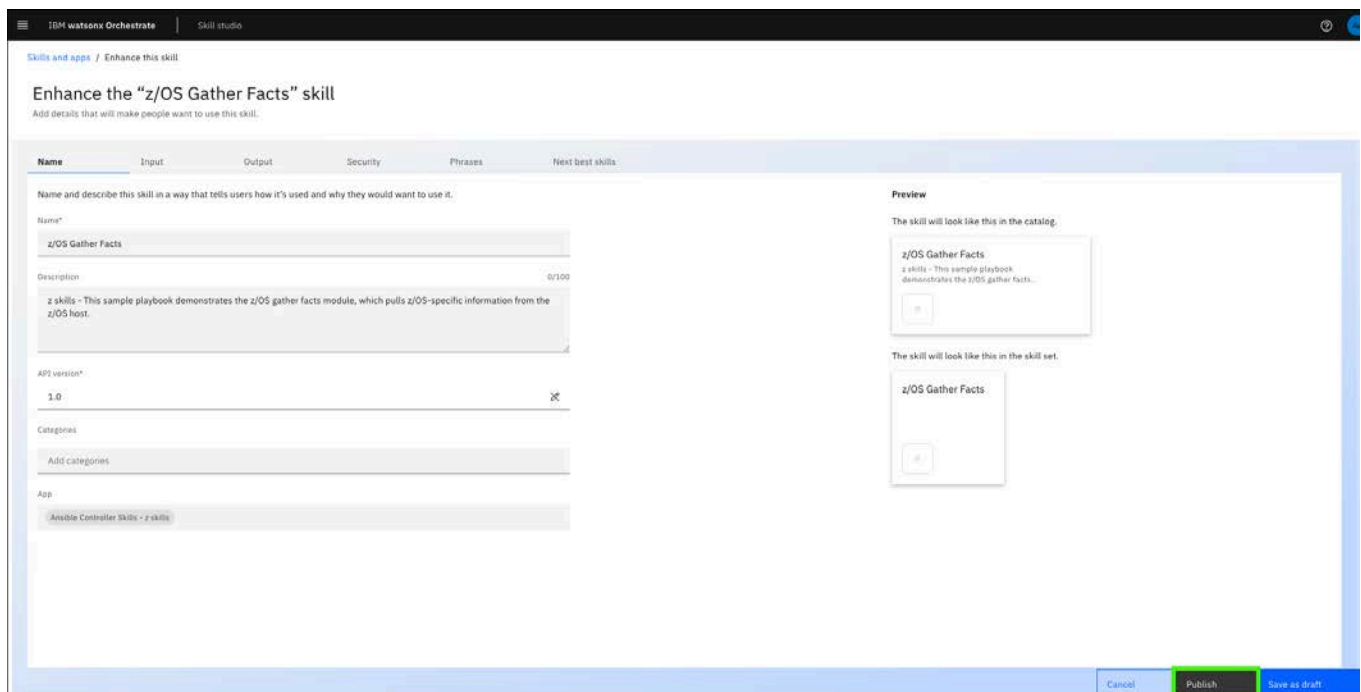
For this lab, select the **Z/os ping** and **Z/os gather facts** skills.

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

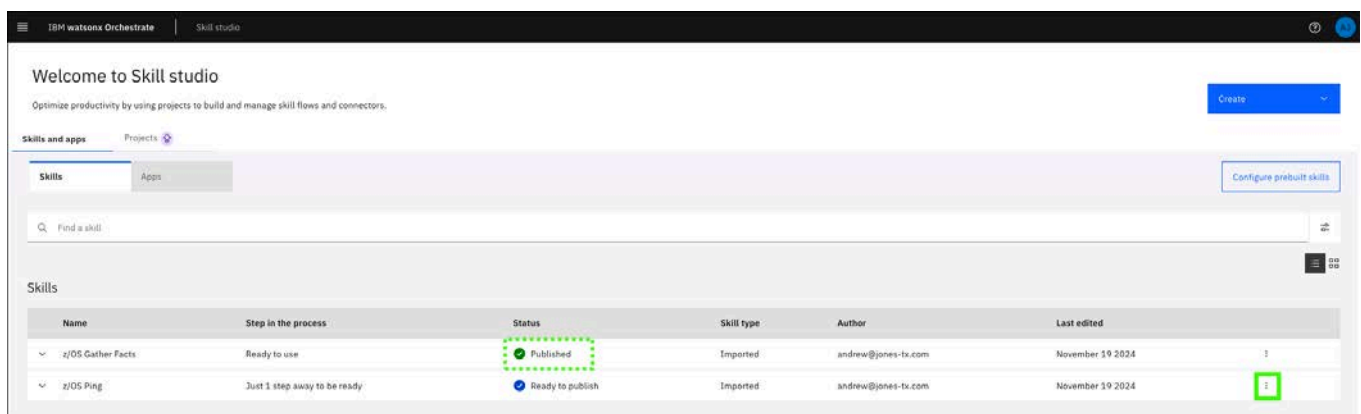


9. Review the skill enhancement options and then click **Publish**.

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



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

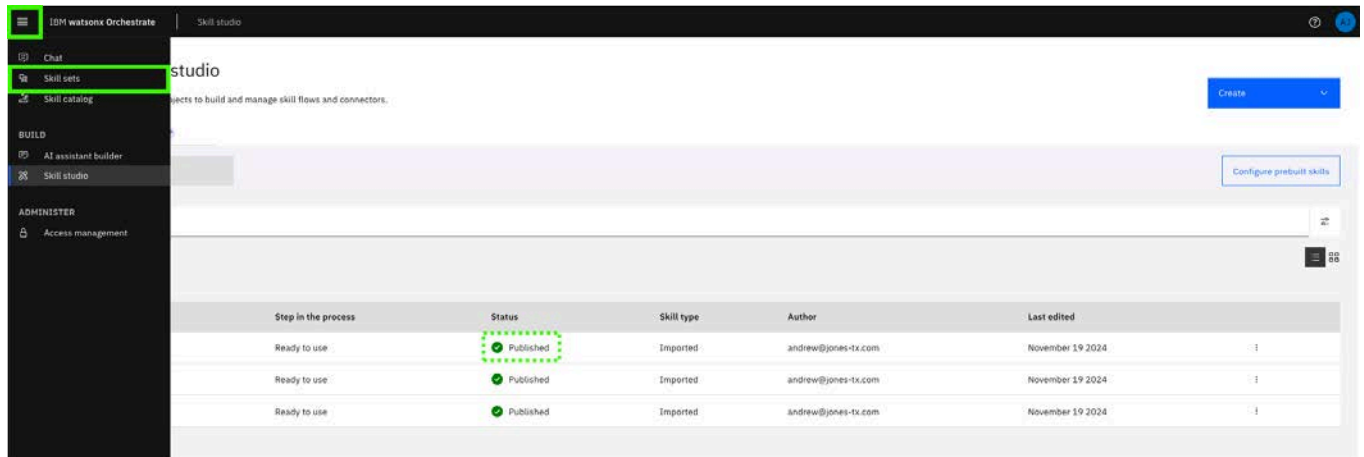


The selected skills are now ready for use and available to your assistant. In the next section, learn how to connect them to your assistant.

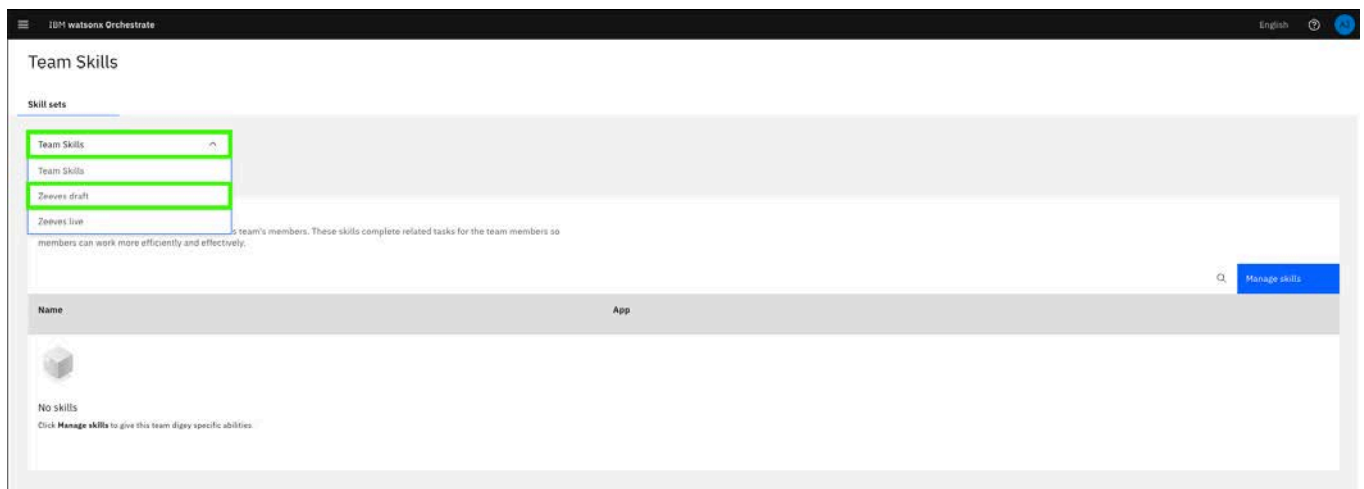
Connecting skills to your assistant

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

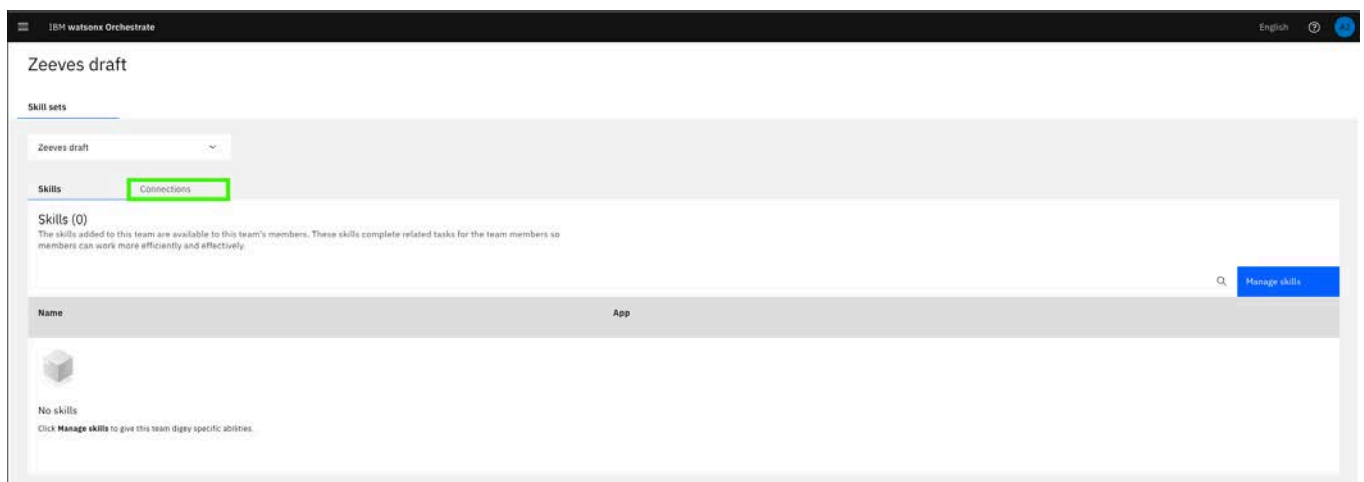
1. Expand the main menu and select **Skill sets**.



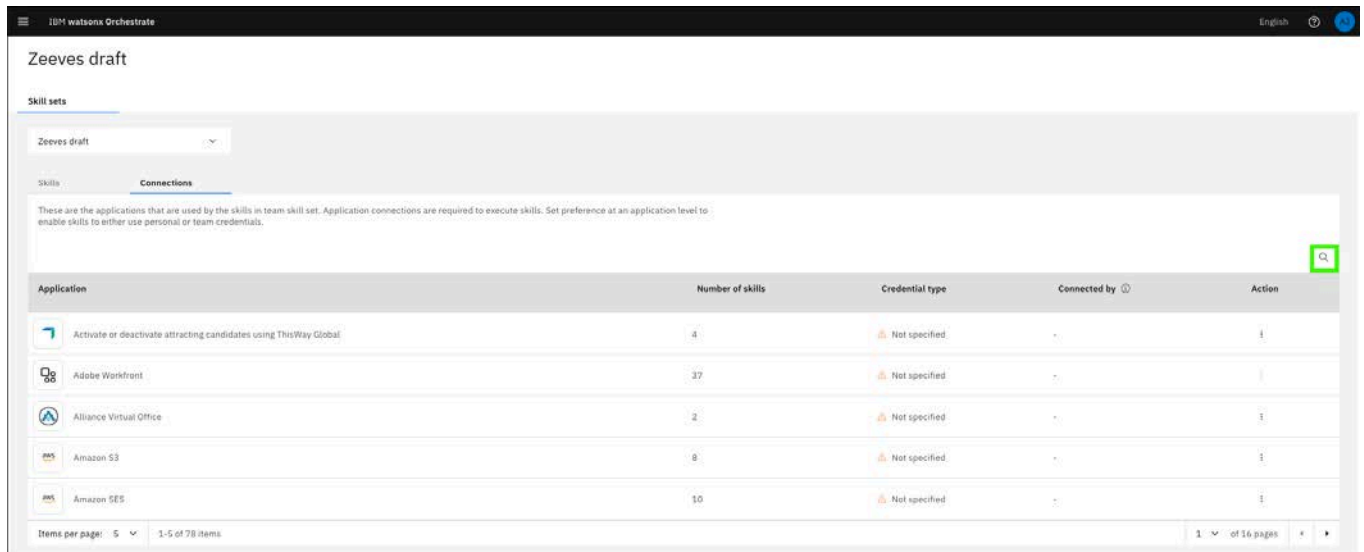
2. Click the **Team Skills** drop-down and select the **Draft** of your assistant.



3. Click the **Connections** tab.



4. Click the **Search** (🔍) icon.



IBM watsonx Orchestrate

Zeeves draft

Skill sets

Zeeves draft

Skills

Connections

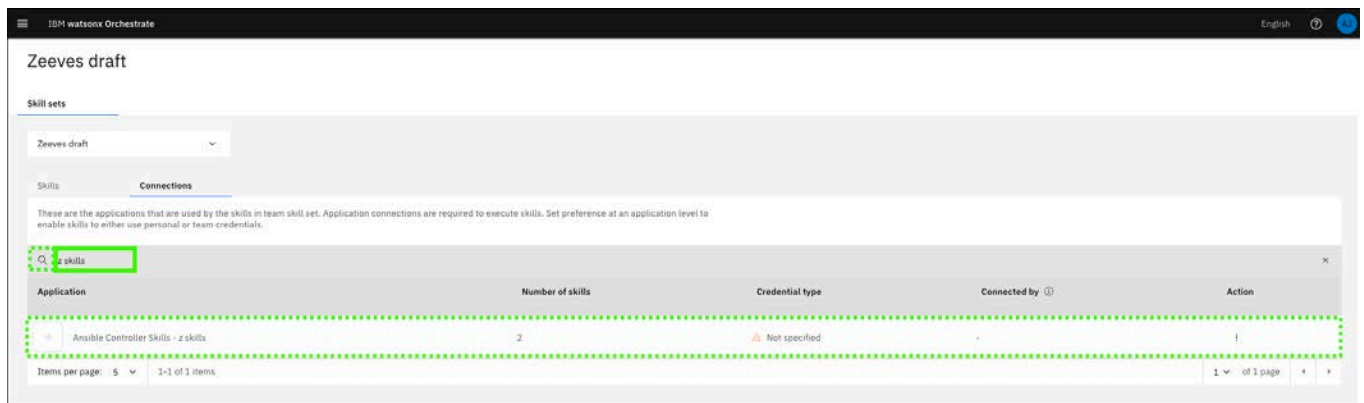
These are the applications that are used by the skills in team skill set. Application connections are required to execute skills. Set preference at an application level to enable skills to either use personal or team credentials.

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](#).



IBM watsonx Orchestrate

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.

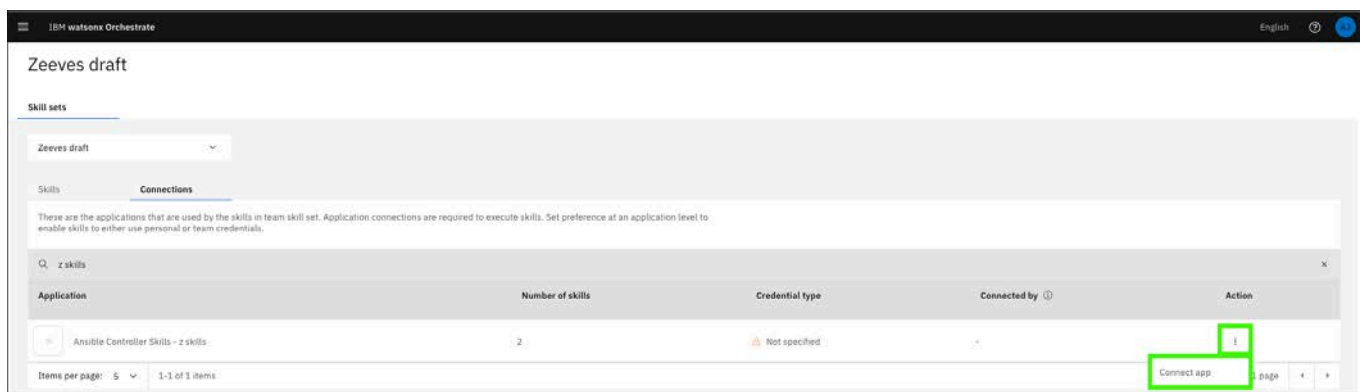
z skills

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**.



IBM watsonx Orchestrate

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.

z skills

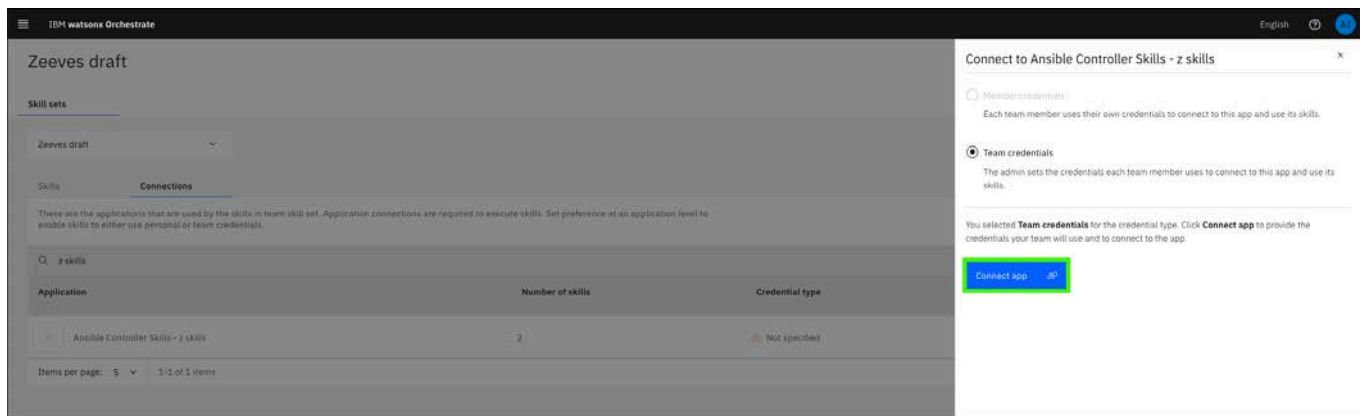
Application	Number of skills	Credential type	Connected by	Action
Ansible Controller Skills - z skills	2	Not specified	-	!

Items per page: 5 1-1 of 1 items

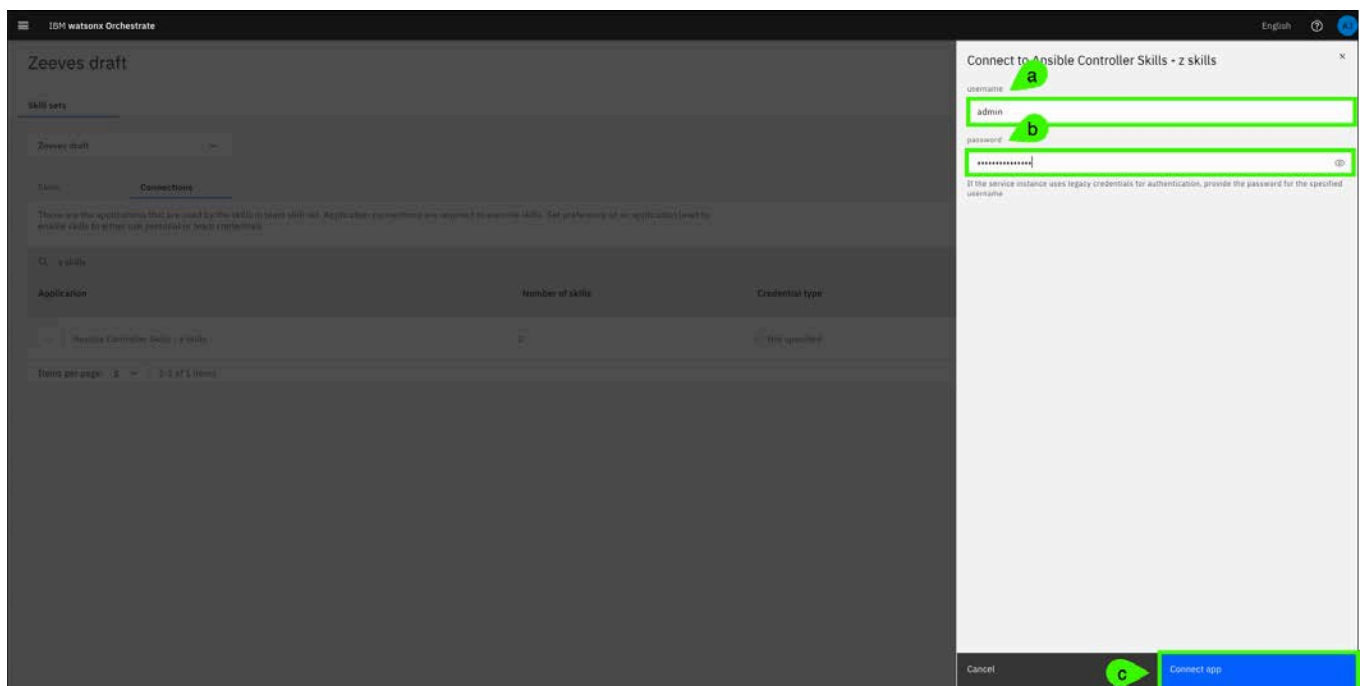
1 page

Connect app

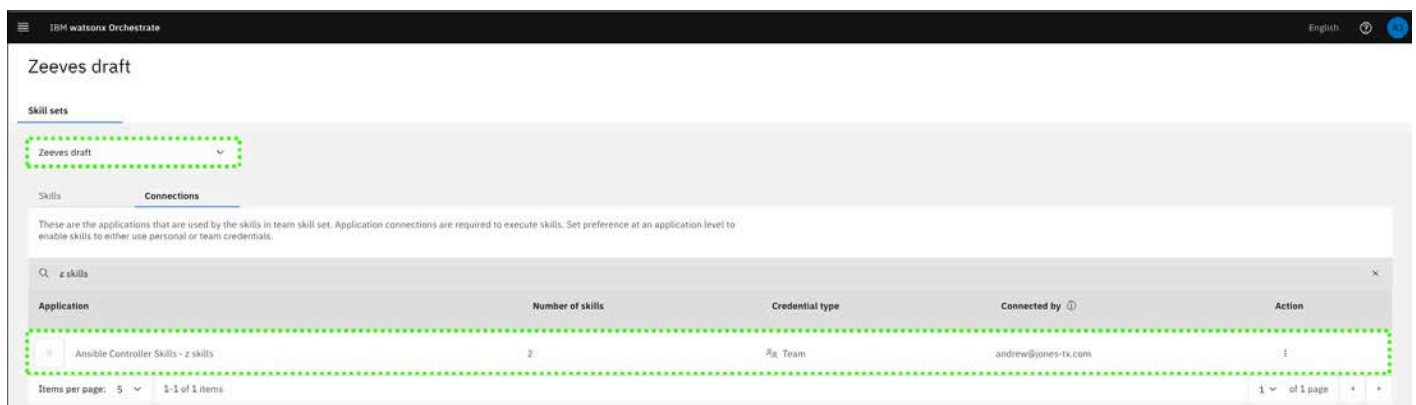
7. On the **Connect to Ansible Controller Skills** form, keep the defaults and click **Connect app**.



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)**.



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



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

Creating actions for your assistant

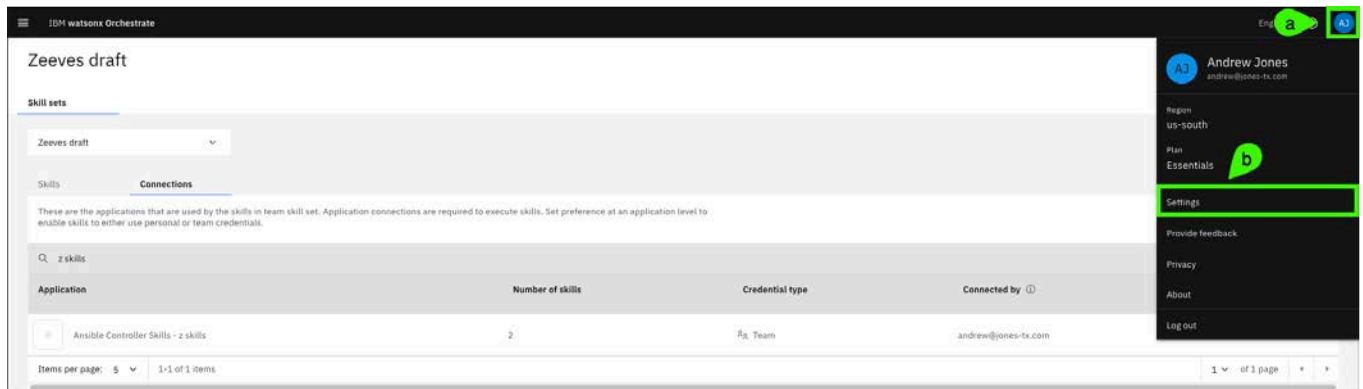
Now that the skills in your application are connected to your assistant, you are ready to create actions that are tied to those skills. Learn more about building actions [here](#)

Configure the number of input fields

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

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

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



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



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

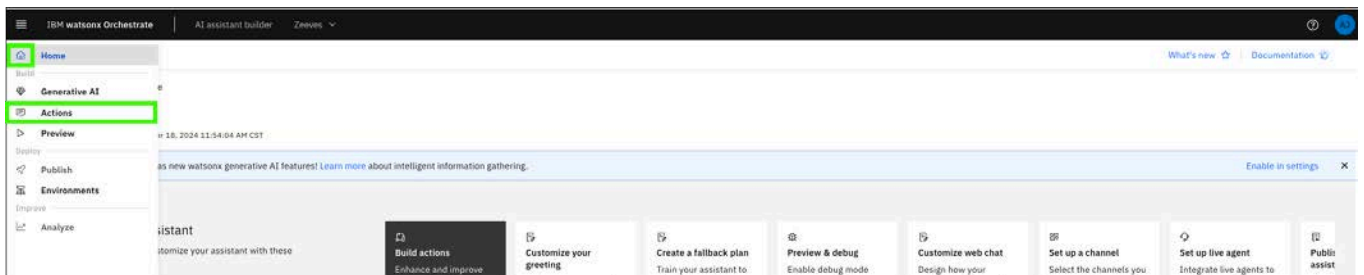


Create actions

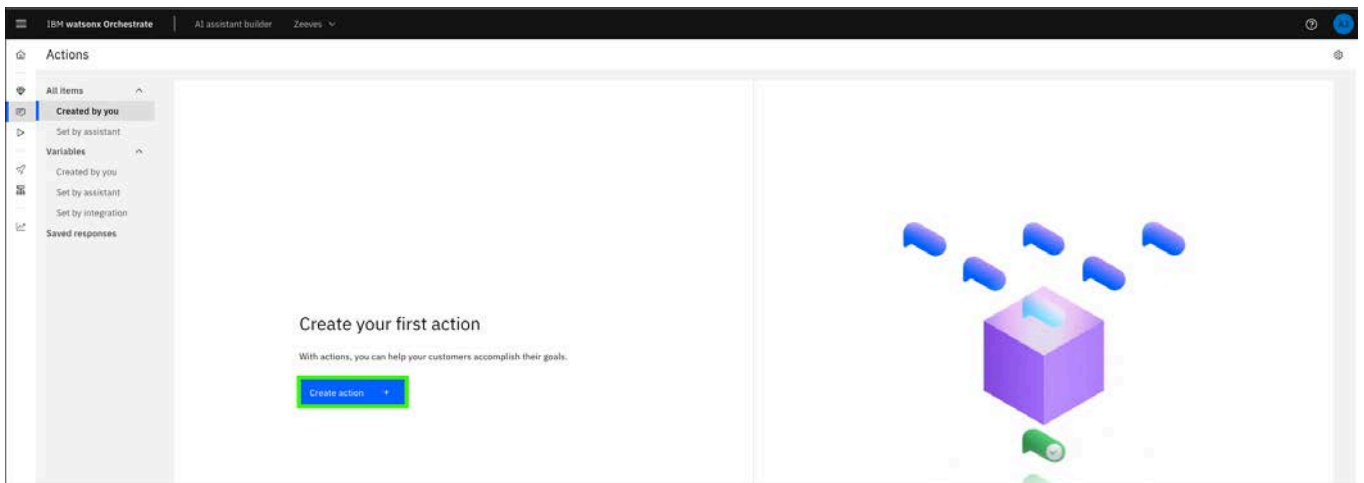
1. Click the main menu and select **AI assistant builder**.



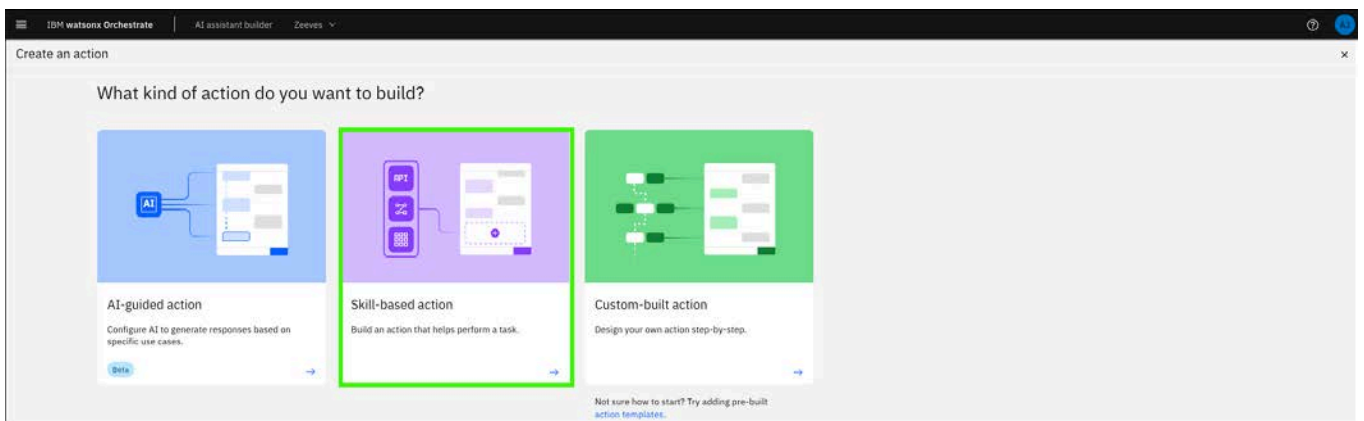
2. Hover over the **Home** icon (🏠) and click **Actions**.



3. Click **Create action**.

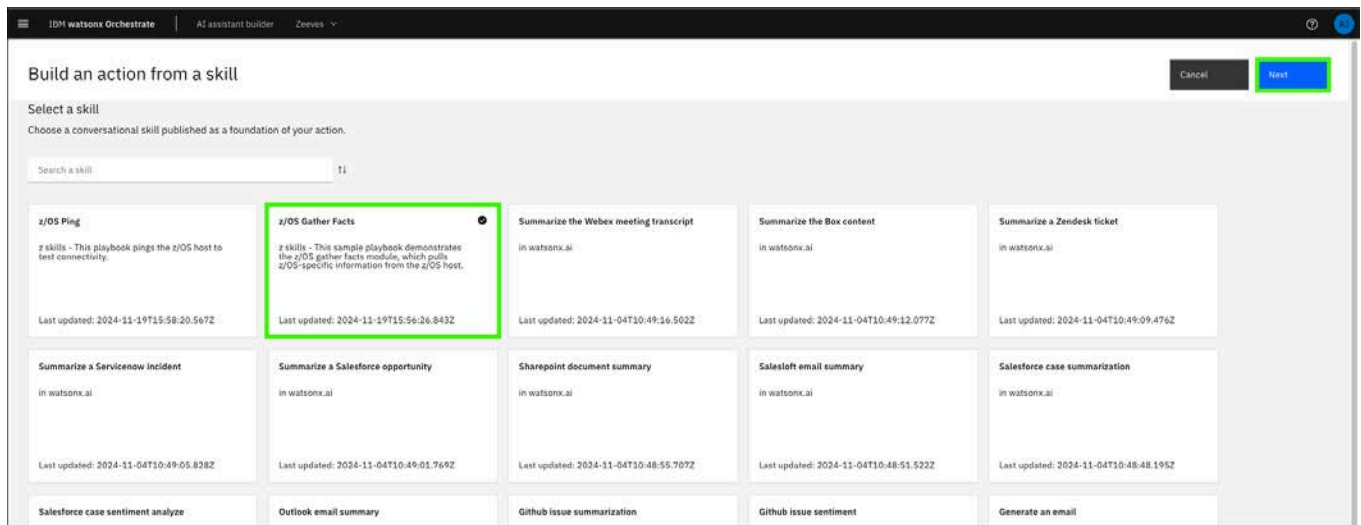


4. Click the **Skill-based action** tile.



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

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

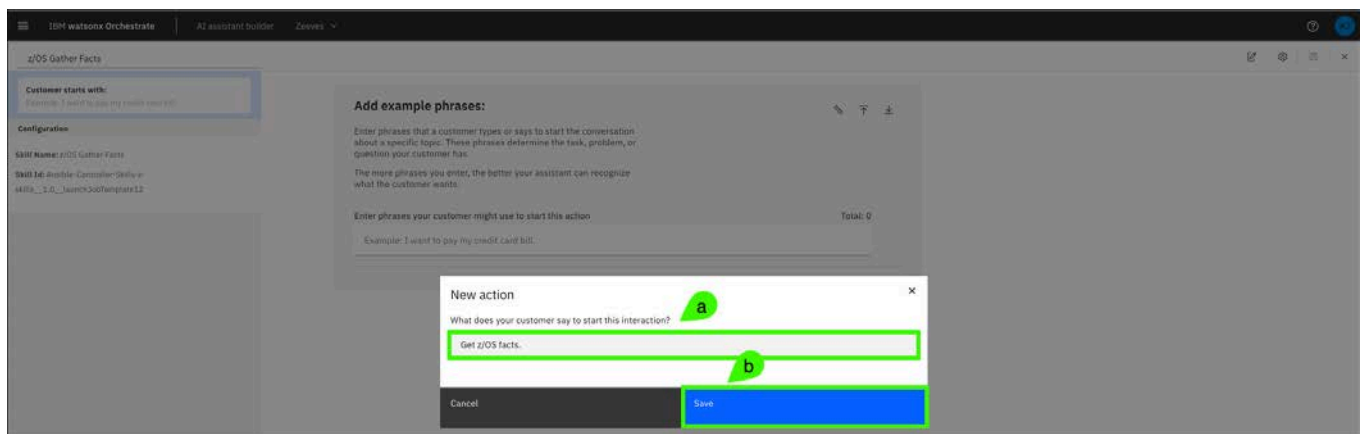


6. On the **New action** dialog, (a) enter a prompt a user of the assistant might use to initiate the action and then (b) click **Save**.

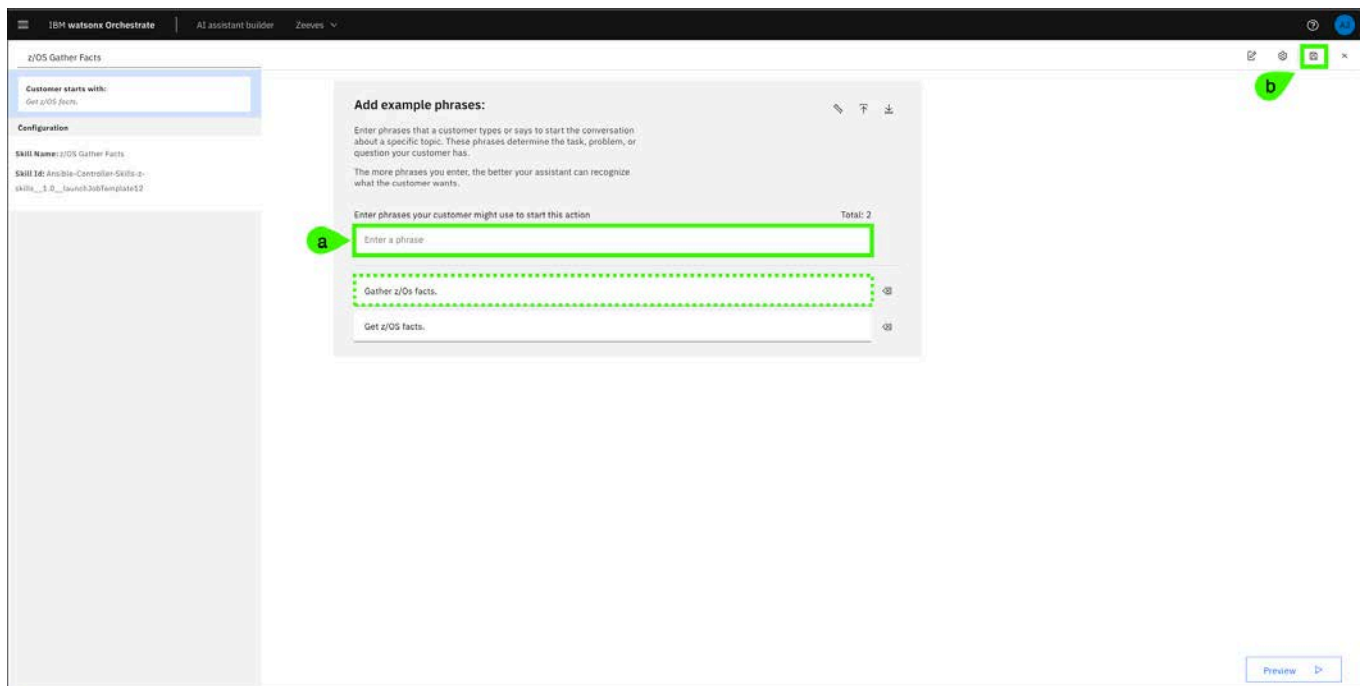
Sample prompts:

Get z/OS facts

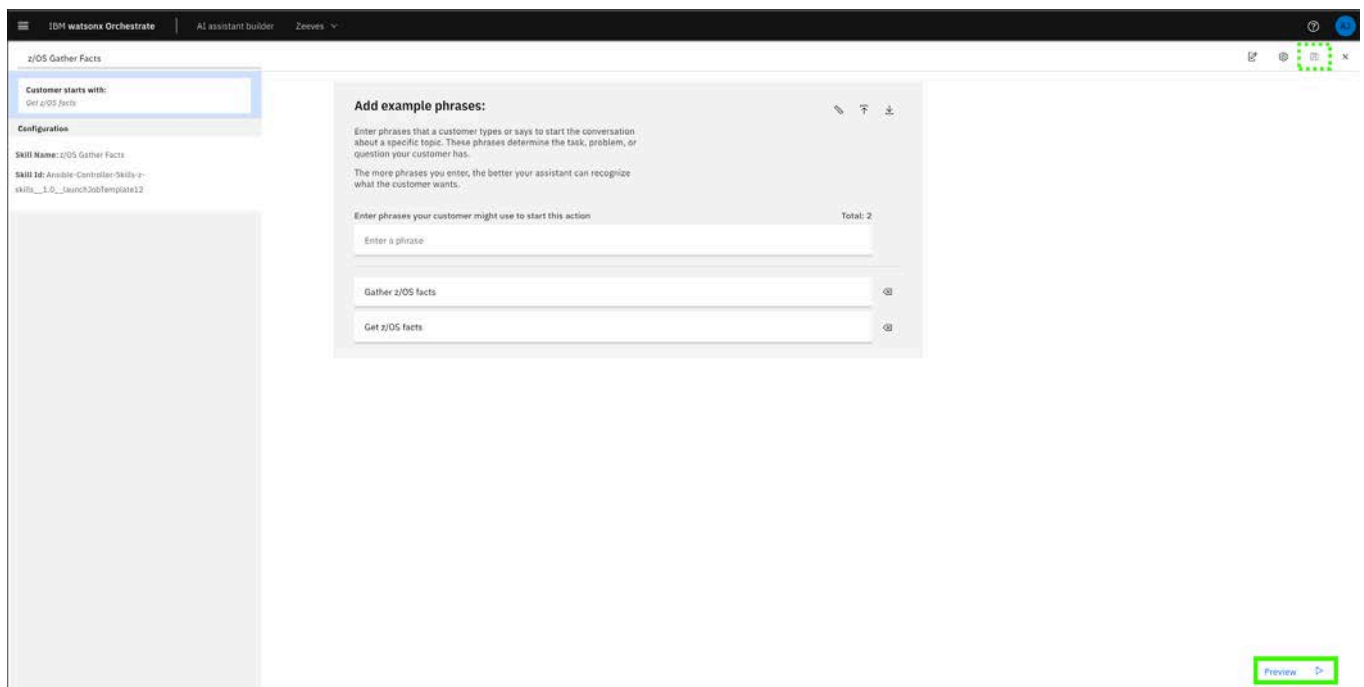
Gather z/OS facts



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



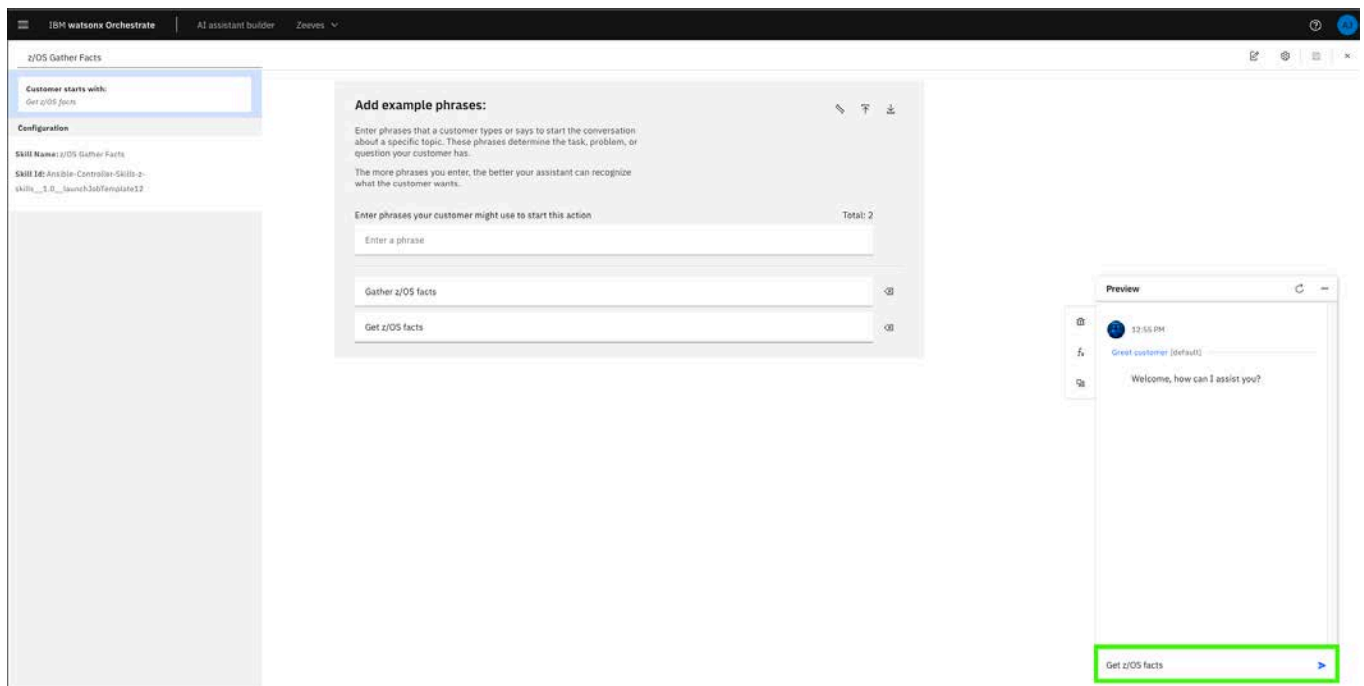
8. Click **Preview**.



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

Prompt:

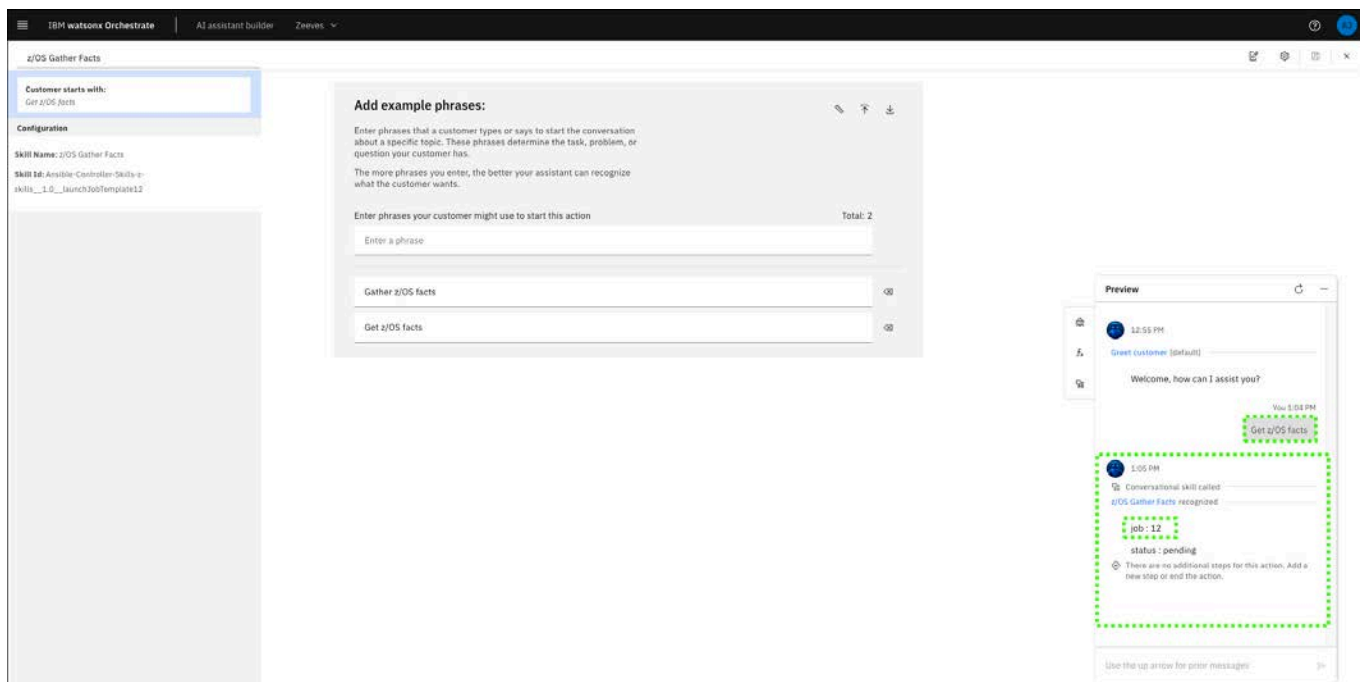
Get z/OS facts



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



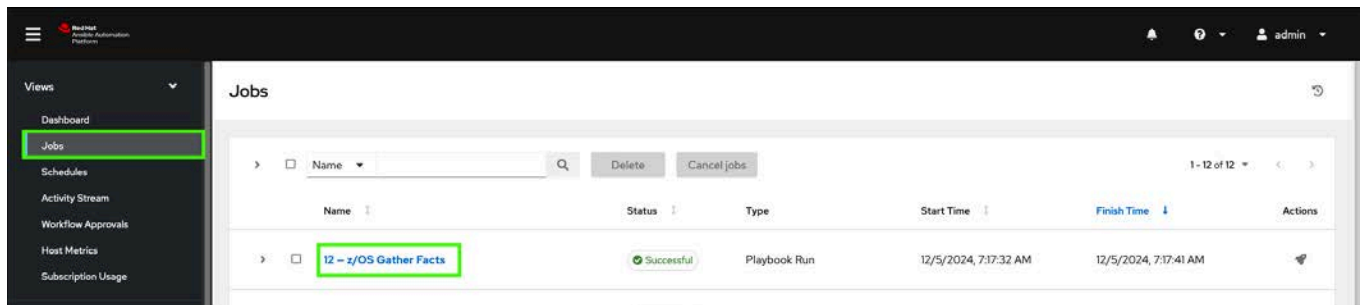
If an error is generated or the action is not performed and only search results are returned, review the following Troubleshooting section.



Verify the job in the Ansible Automation Platform console

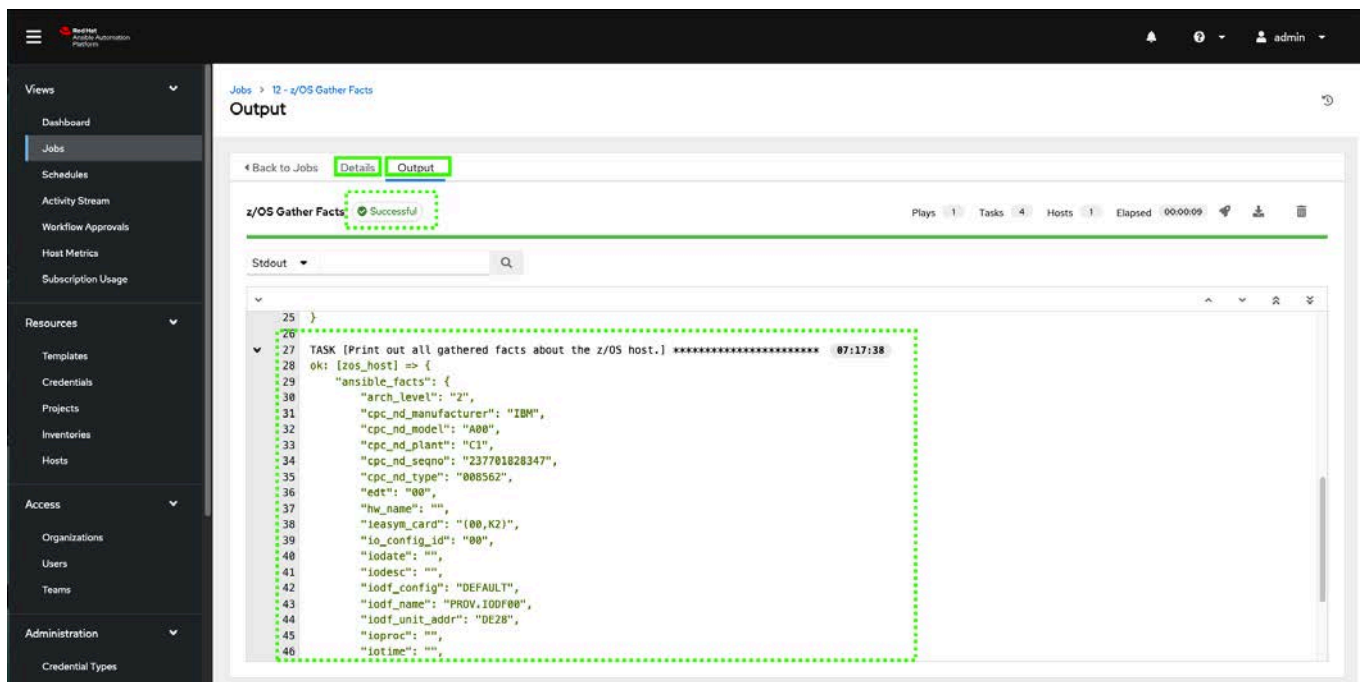
Return to the Ansible Automation Platform (AAP) console and review the job information.

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



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.



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

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

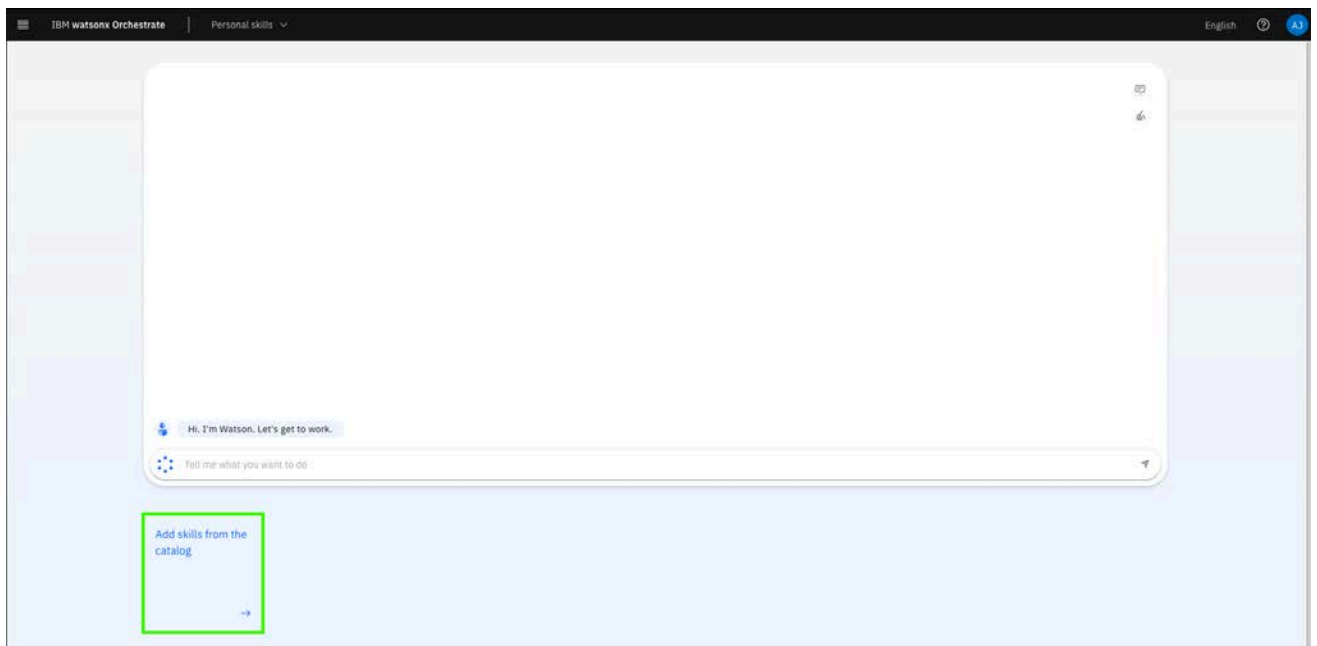
The screenshot shows the IBM watsonx Orchestrate AI assistant builder interface. On the left, the 'Configuration' panel shows the skill name 'z/OS Gather Facts' and its ID. The main area is titled 'Add example phrases:' and contains a text input field 'Enter a phrase' and a list of phrases: 'Gather z/OS facts' and 'Get z/OS facts'. On the right, a 'Preview' window shows a chat conversation. The chat starts with a user message 'Greet customer [default]' and a system response 'Welcome, how can I assist you?'. The user then says 'You 10:46 AM' and the system responds 'Gather z/OS facts'. The chat then shows a system message '10:46 AM' and a system response 'Conversational skill called z/OS Gather Facts recognized'. Below this, a red error message is displayed: 'Sorry, we're having issues generating a response.' This error message is highlighted with a green dashed border. Below the error message, there are two buttons: 'Error in the response' and 'Skill error'. At the bottom of the preview window, there is a text input field 'Type something...' and a 'Preview' button.

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:

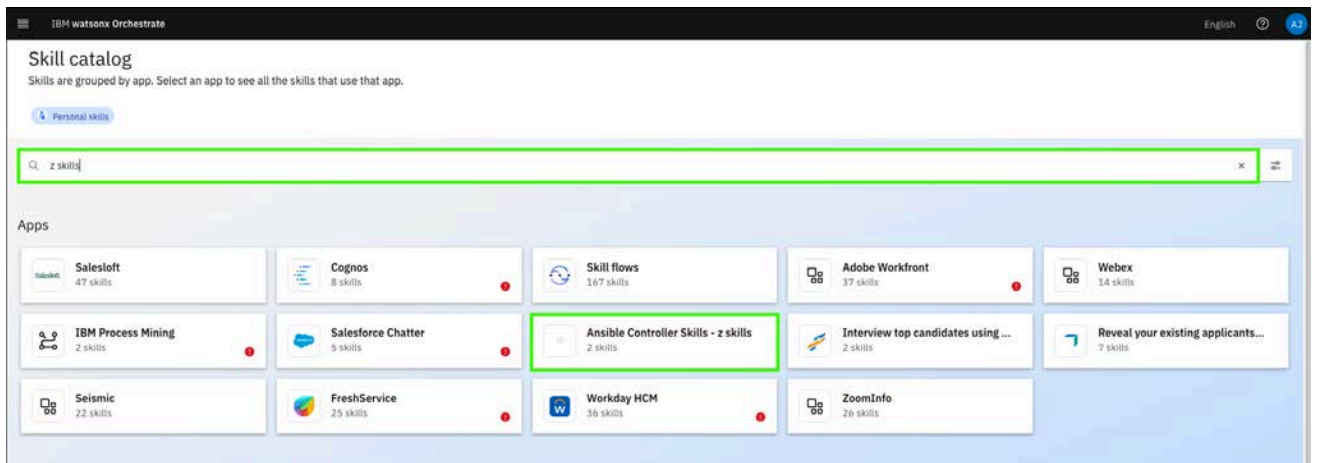
1. Expand the main menu and select **Chat**.

The screenshot shows the IBM watsonx Orchestrate AI assistant builder interface. The main menu is expanded on the left, showing options like 'Chat', 'Skill sets', 'Skill catalog', 'BUILD', 'AI assistant builder', 'Skill studio', 'ADMINISTER', and 'Access management'. The 'Chat' option is highlighted with a green box. The main area is titled 'Add example phrases:' and contains a text input field 'Enter a phrase' and a list of phrases: 'Gather z/OS facts' and 'Get z/OS facts'. On the right, a 'Preview' window is visible, showing a chat conversation.

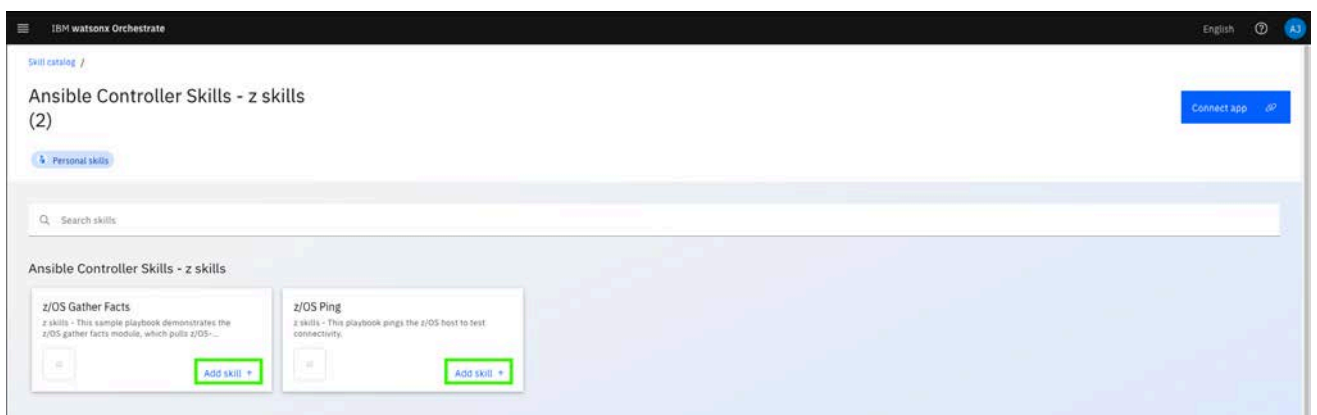
2. Click **Add skills from the catalog**.



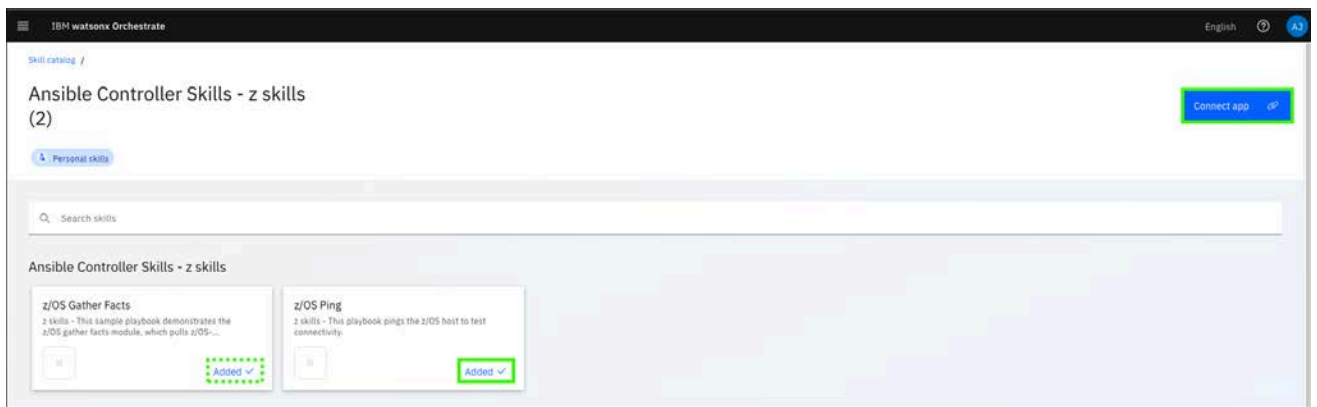
3. Search for the skill app you created earlier and click the tile for your app.



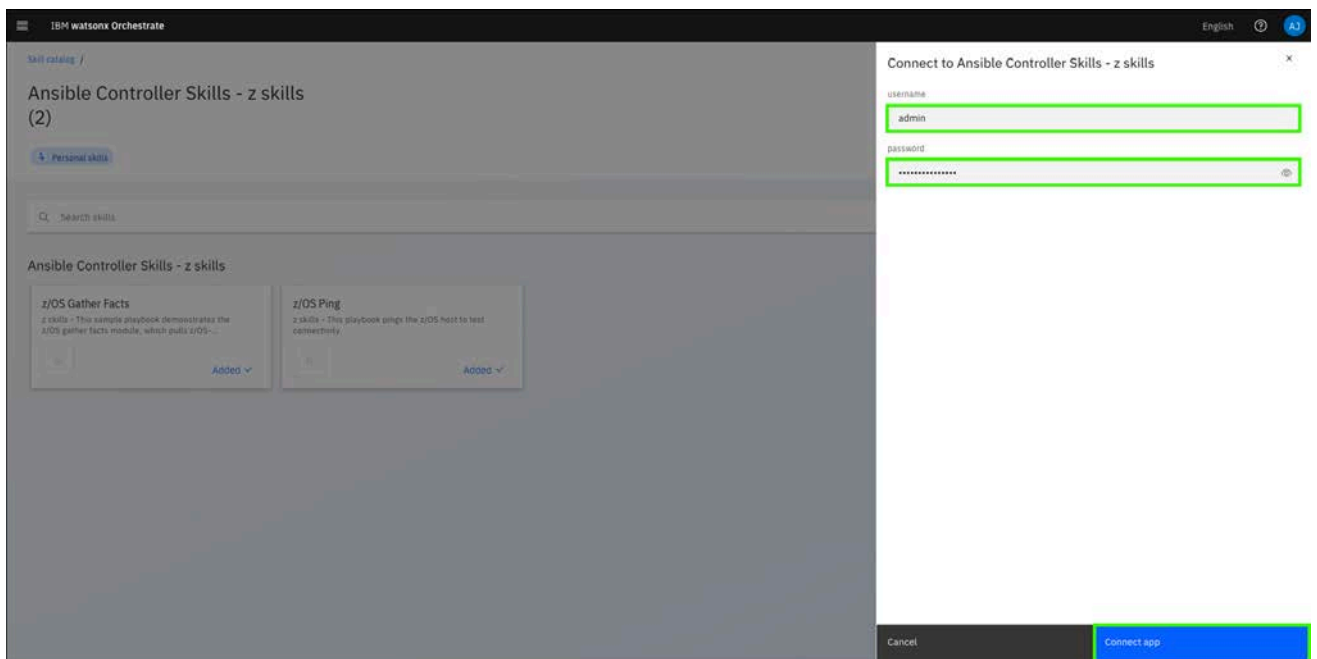
4. Click **Add skill** for all the skills you want to add.



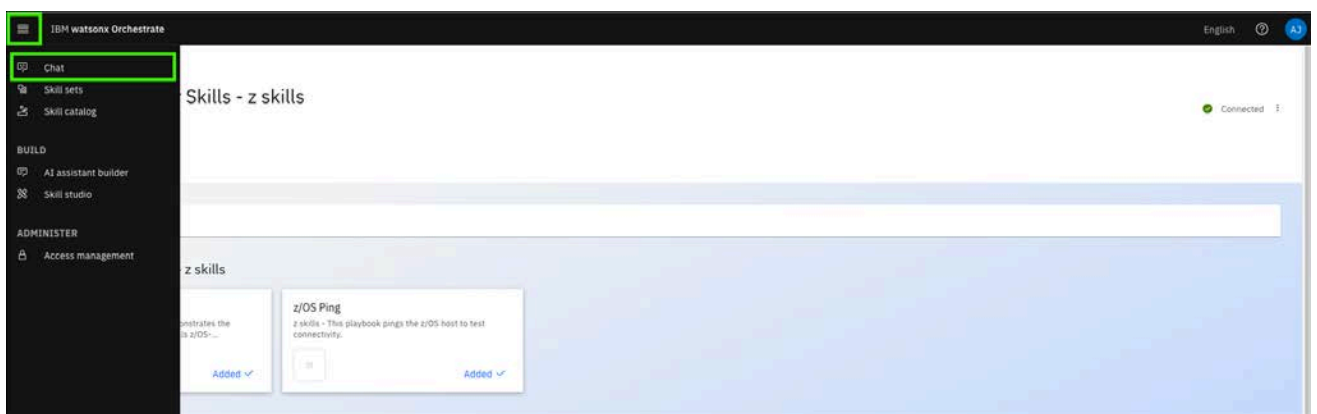
5. Click **Connect app**.



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**.



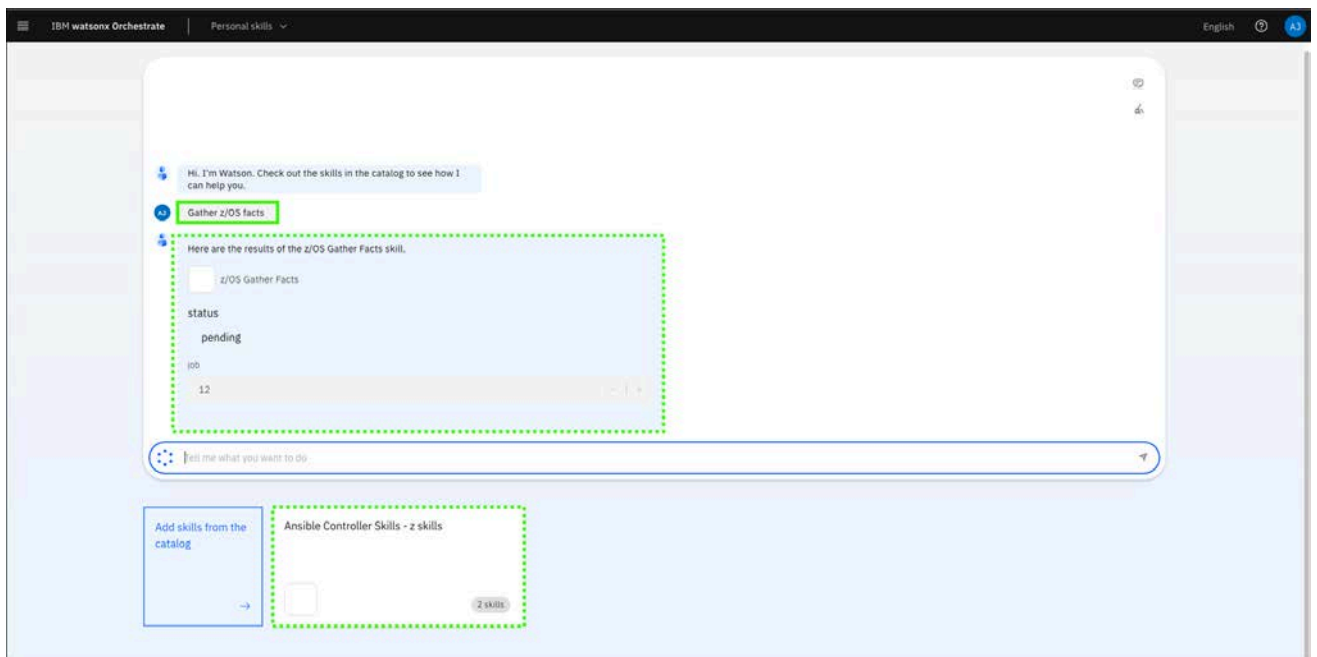
7. Expand the main menu and select **Chat**.



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

Prompt:

Gather z/OS facts



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

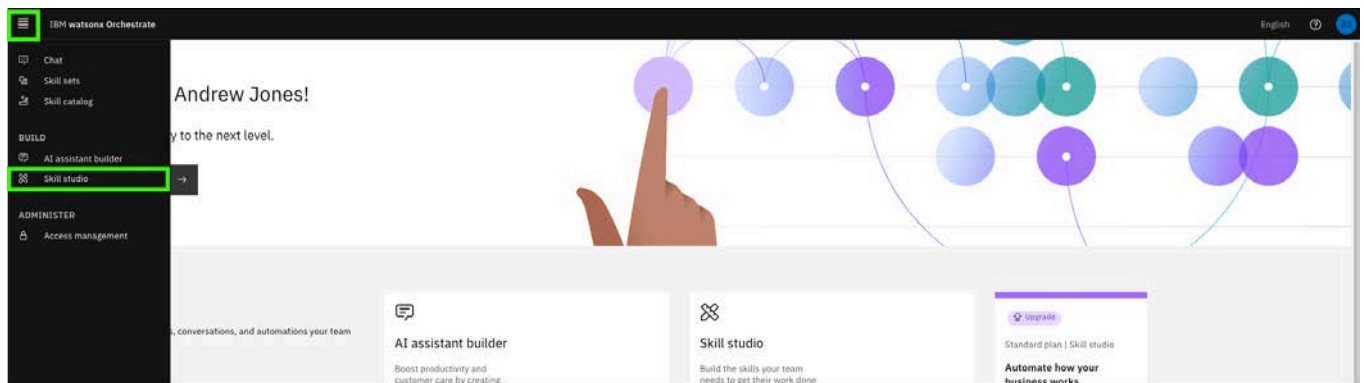
Creating skill flows

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](#).

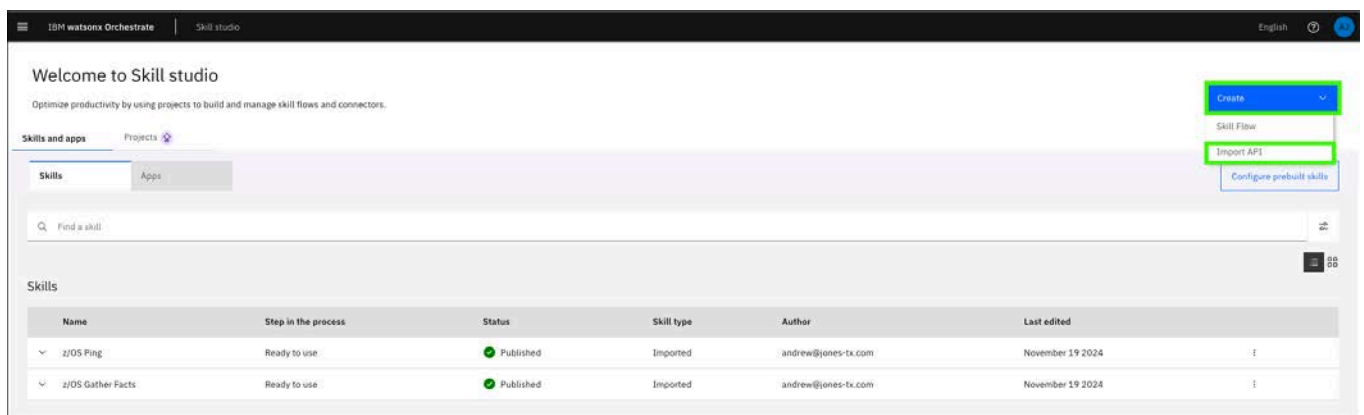
As mentioned in a previous section, default utility skills that are provided with the watsonx Assistant for Z skills collection. The **Retrieve job output** utility skill is used to return the output of a skill.

Add the utility skill

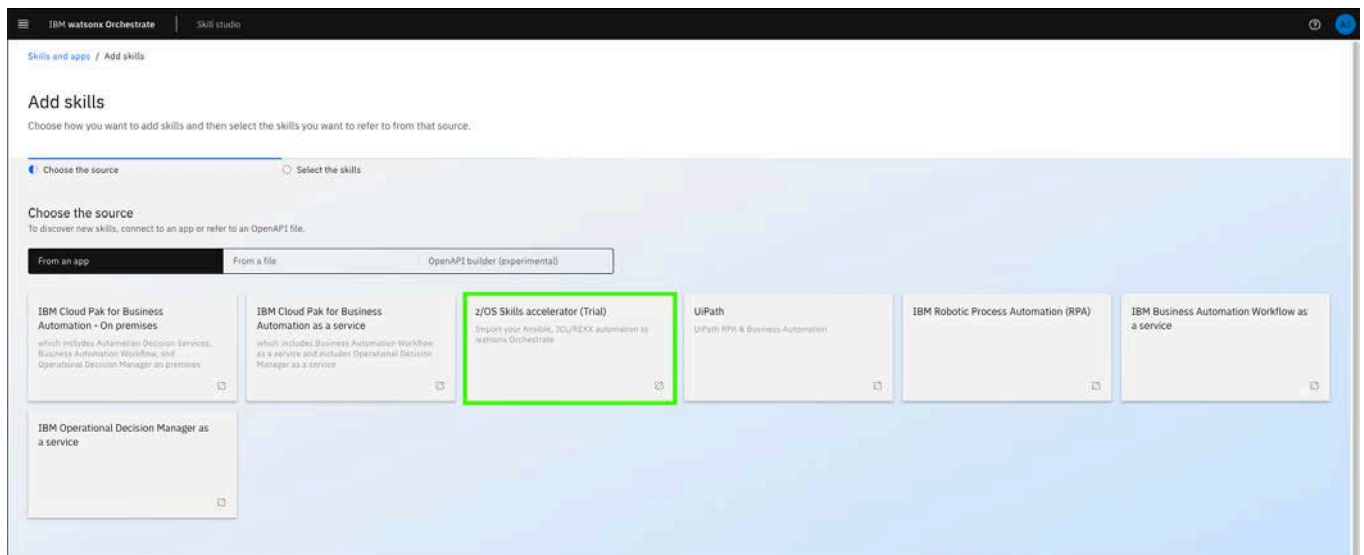
1. Open IBM watsonx Orchestrate **Skill studio**.



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**

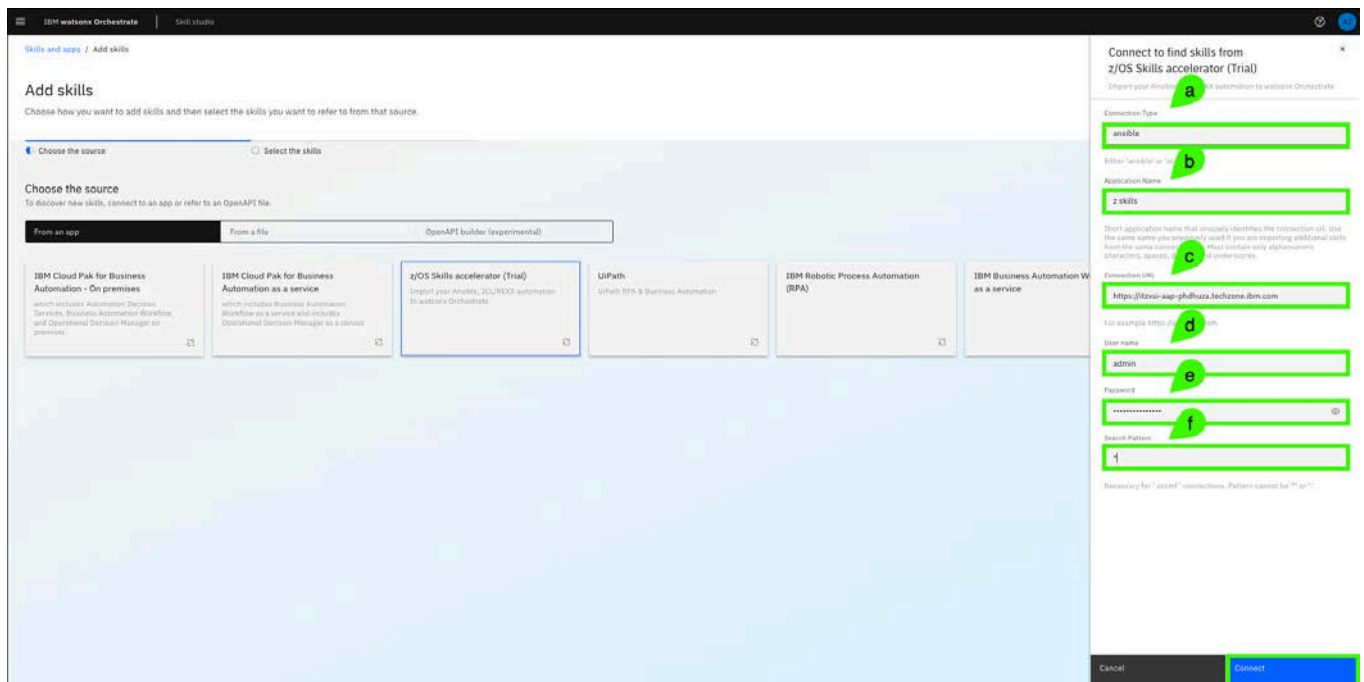
b: Application Name: <use the same application name from the previous section>

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

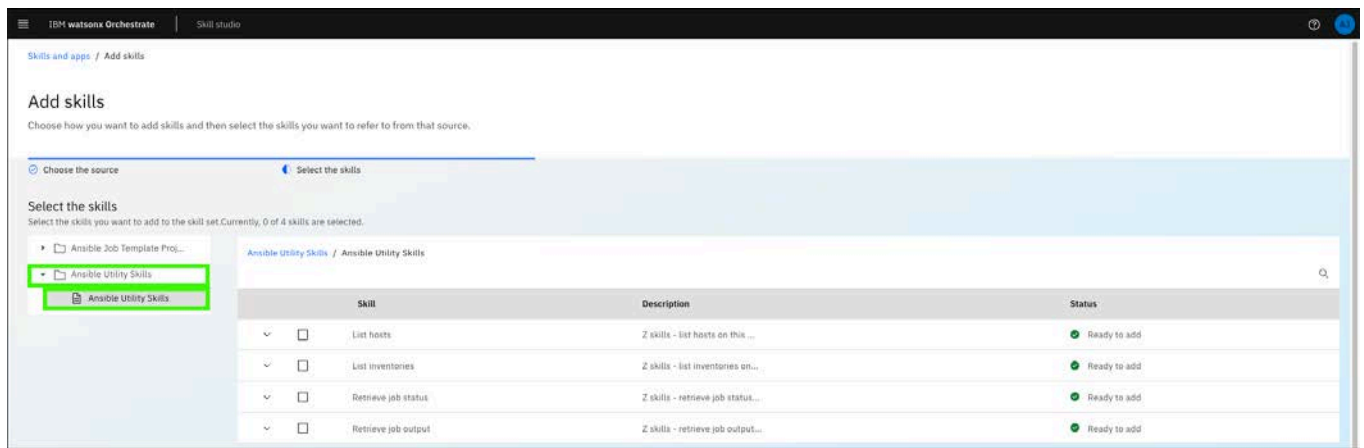
d: User Name: <enter the AAP User Name (for UI access)>

e: Password: <enter the AAP User Password>

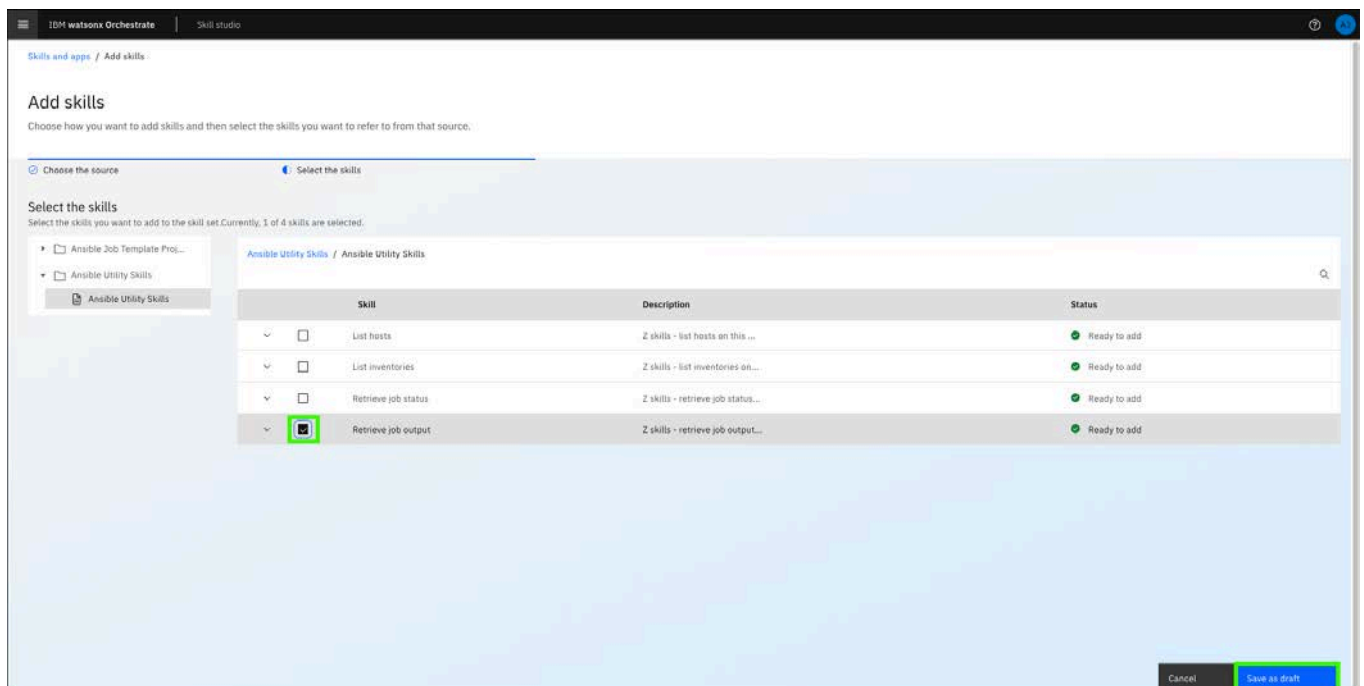
f: Search Pattern: *



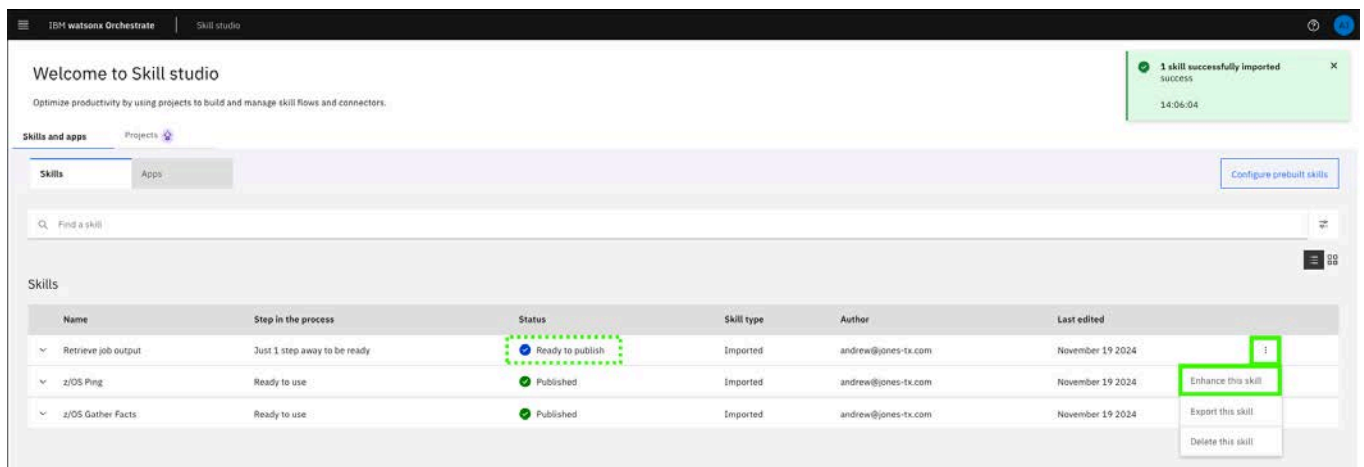
5. Expand **Ansible Utility Skills** and click **Ansible Utility Skills**.



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



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



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

IBM watsonx Orchestrate | Skill studio

Skills and apps / Enhance this skill

Enhance the "Retrieve job output" 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*	Retrieve job output				
Description	z skills - Retrieve job output by job Id				
API version*	1.0				
Categories	Add categories				
App	Amplitude Controller Skills - z skills				

Preview

The skill will look like this in the catalog.

Retrieve job output
z skills - Retrieve job output by job Id

The skill will look like this in the skill set.

Retrieve job output

Cancel Publish Save as draft

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

IBM watsonx Orchestrate | Skill studio

Chat Skill sets Skill catalog

BUILD AI assistant builder Skill studio

ADMINISTER Access management

Create

Configure prebuilt skills

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

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

IBM watsonx Orchestrate

Zeeves draft

Skill sets

Zeeves draft

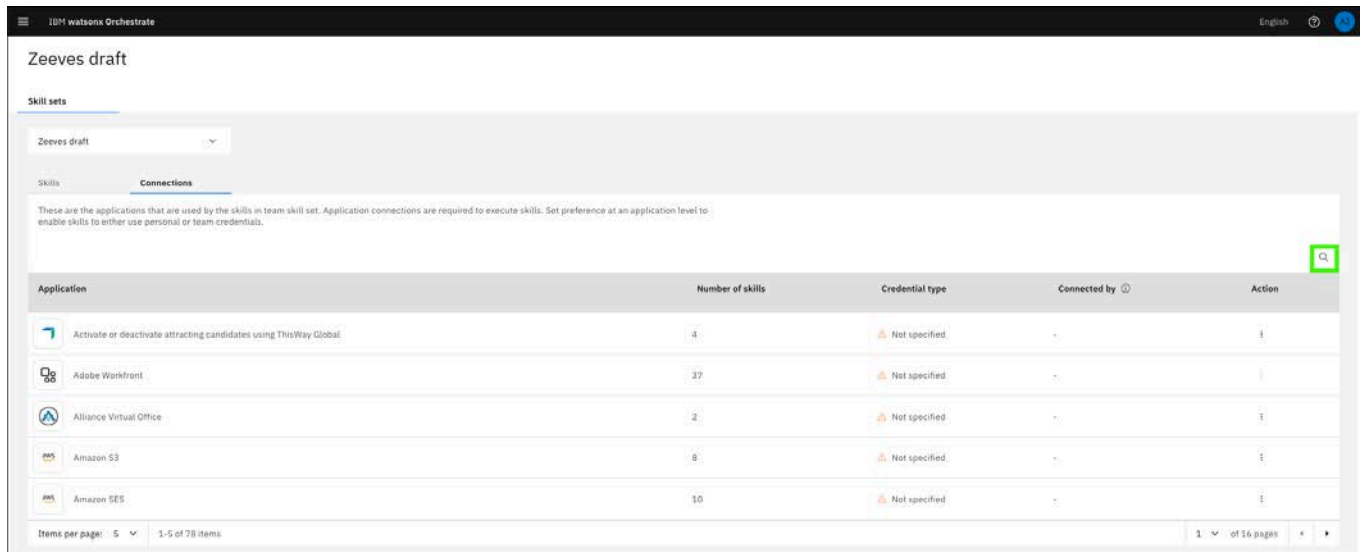
Skills Connections

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.

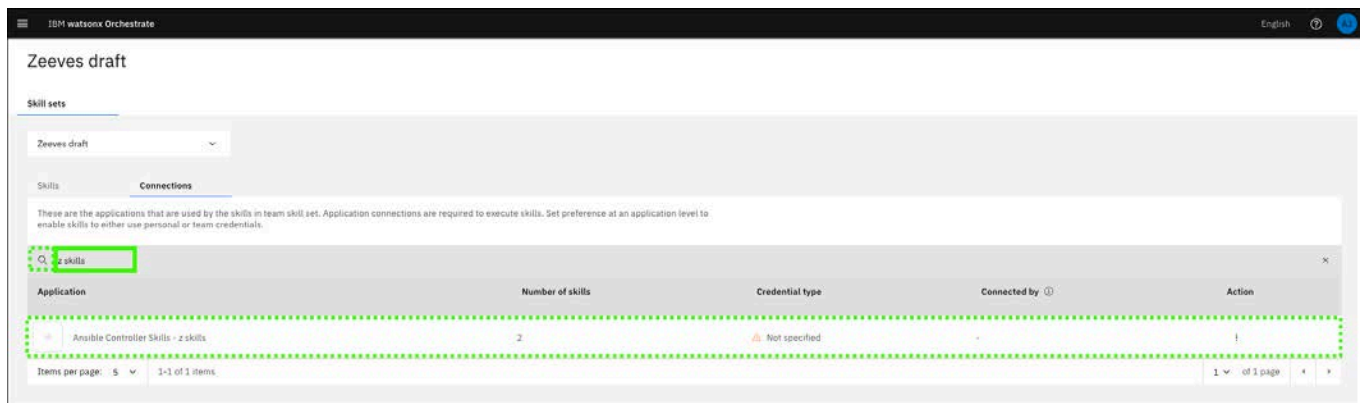
Application	Number of skills	Credential type	Connected by	Action
Activate or deactivate attracting candidates using ThisWay Global	4	Not specified	-	
Adobe Workfront	37	Not specified	-	
Alliance Virtual Office	2	Not specified	-	
Amazon S3	8	Not specified	-	
Amazon SES	10	Not specified	-	

Items per page: 5 1-5 of 78 items 1 of 16 pages

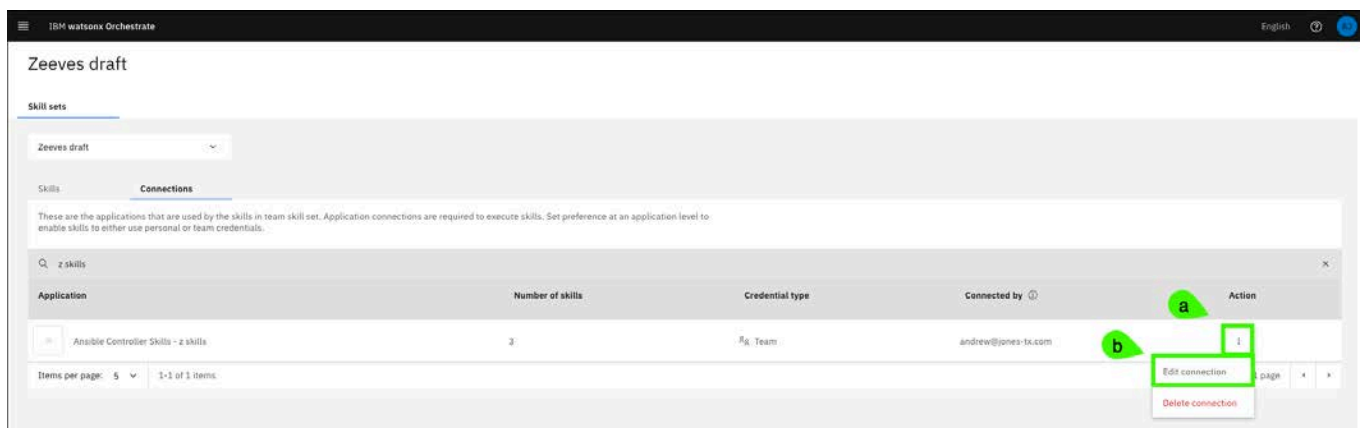
11. Click the **Search** (🔍) icon.



12. Search for the application name you specified earlier.



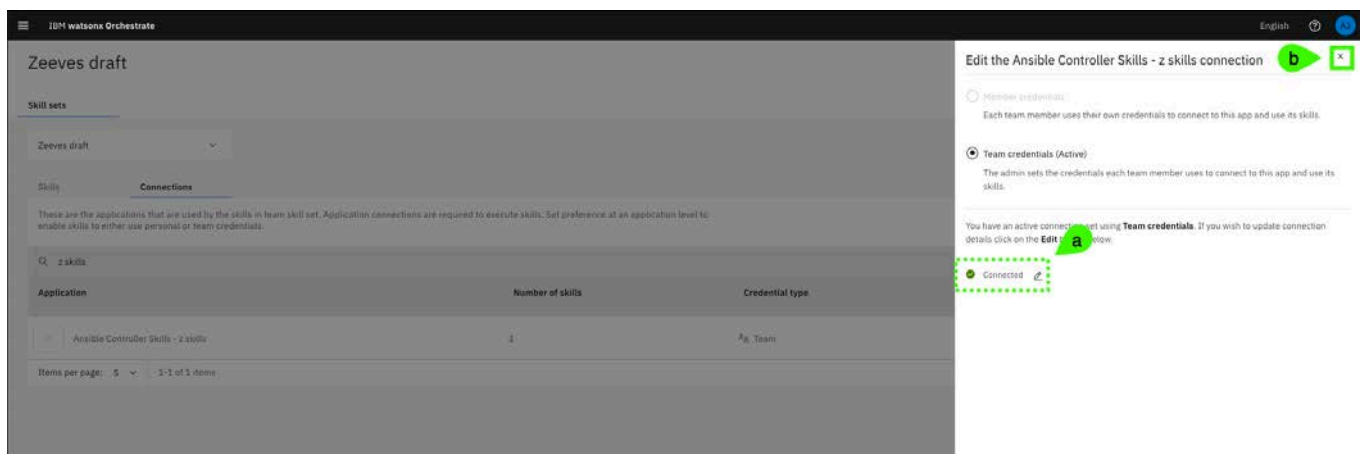
13. Click the (a) ellipses (⋮) for your application and (b) click **Edit connection**.



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

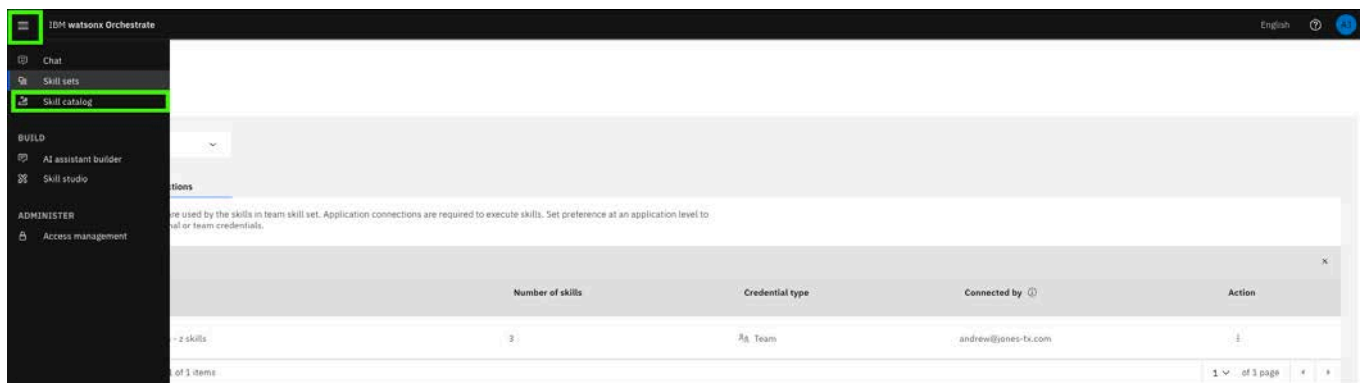
 **Connect the application if it is not connected.**

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

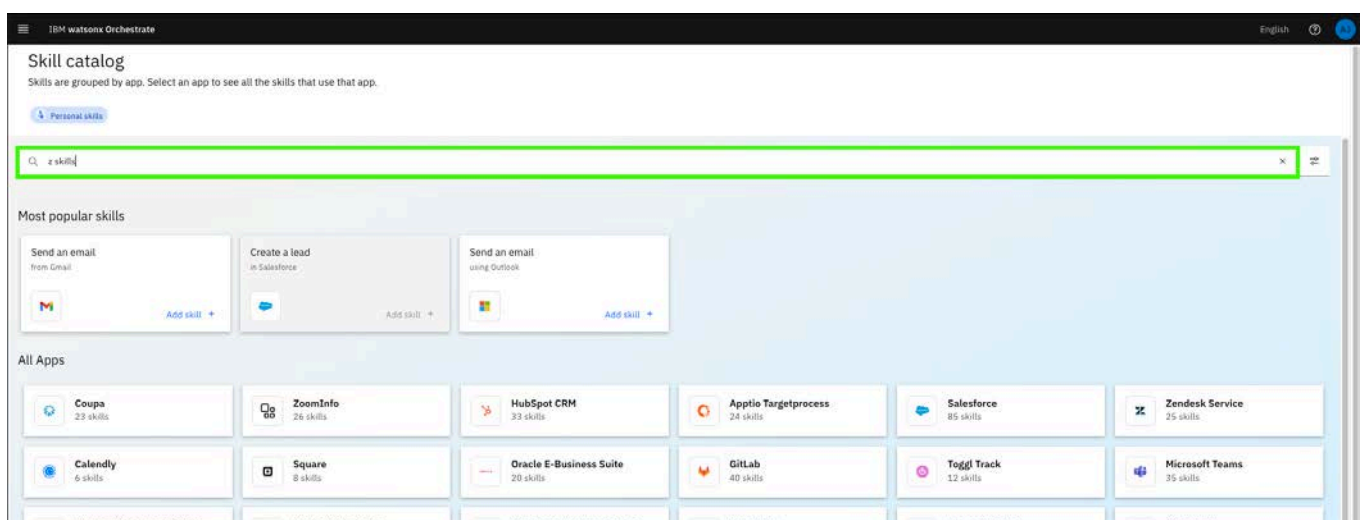


Add the skills to your Personal skills

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

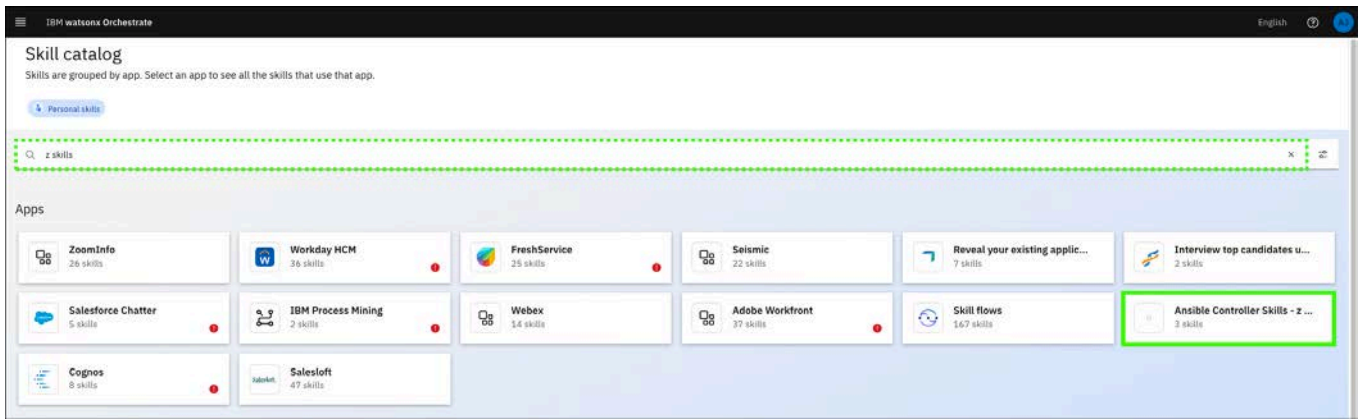


2. Search for the application name you specified earlier.

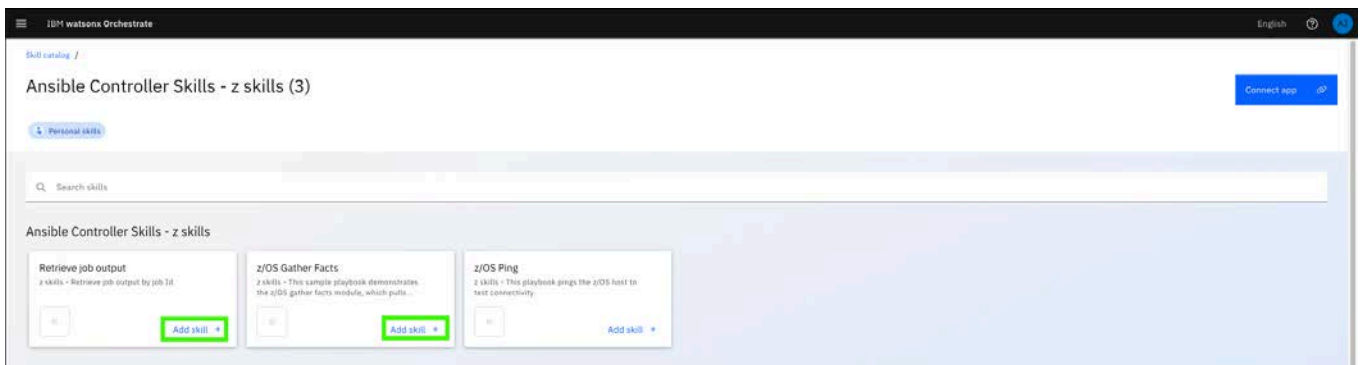


3. Click the tile for your application.

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

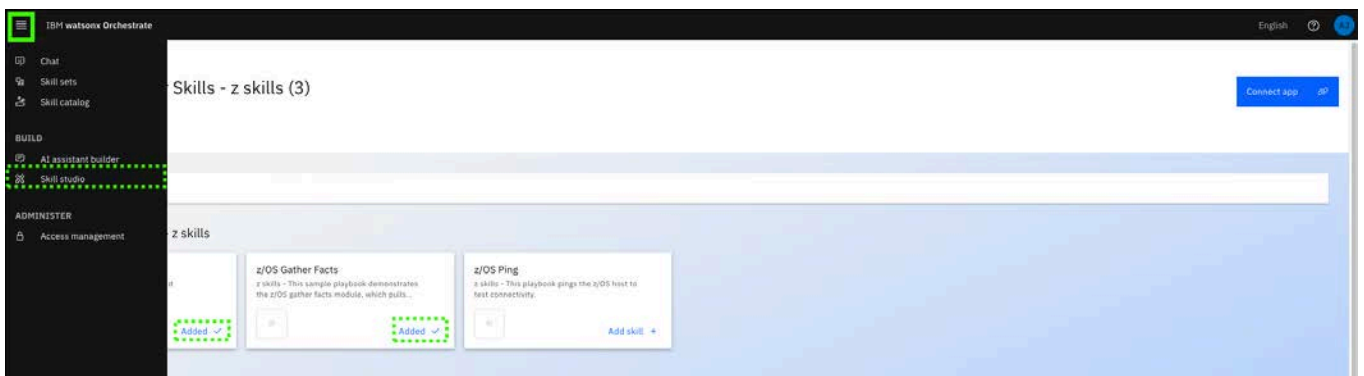


4. Click **Add skill** for each of the skills you want to add to the flow.

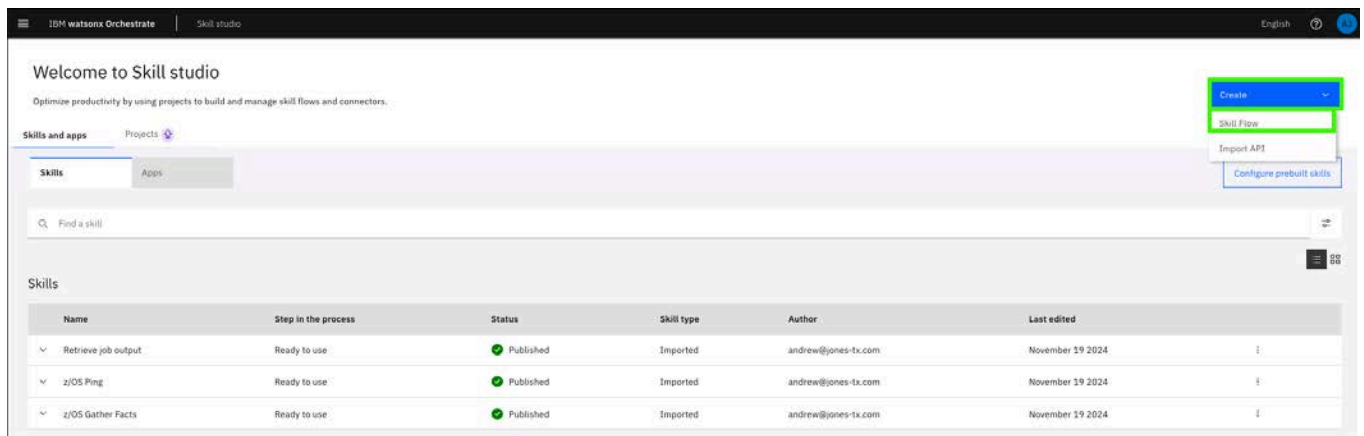


Create the skill flow

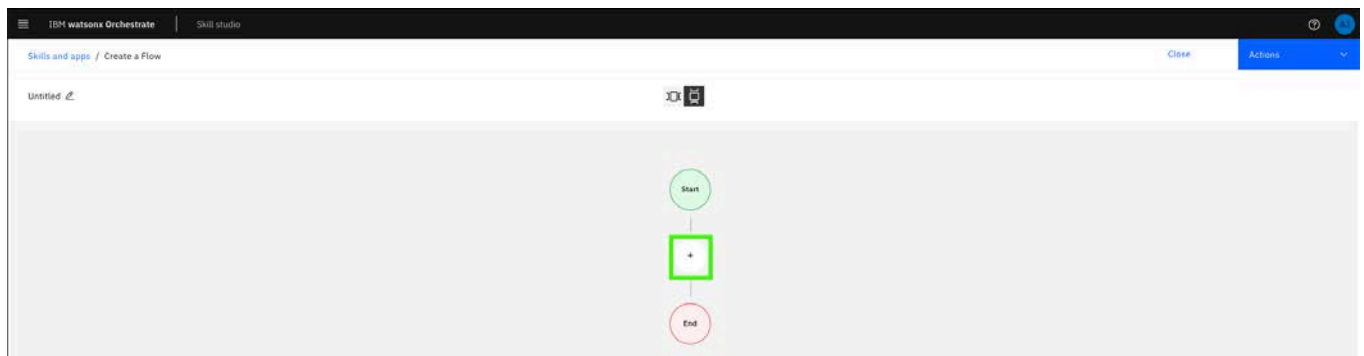
1. Click **Skill studio** in the main menu.



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

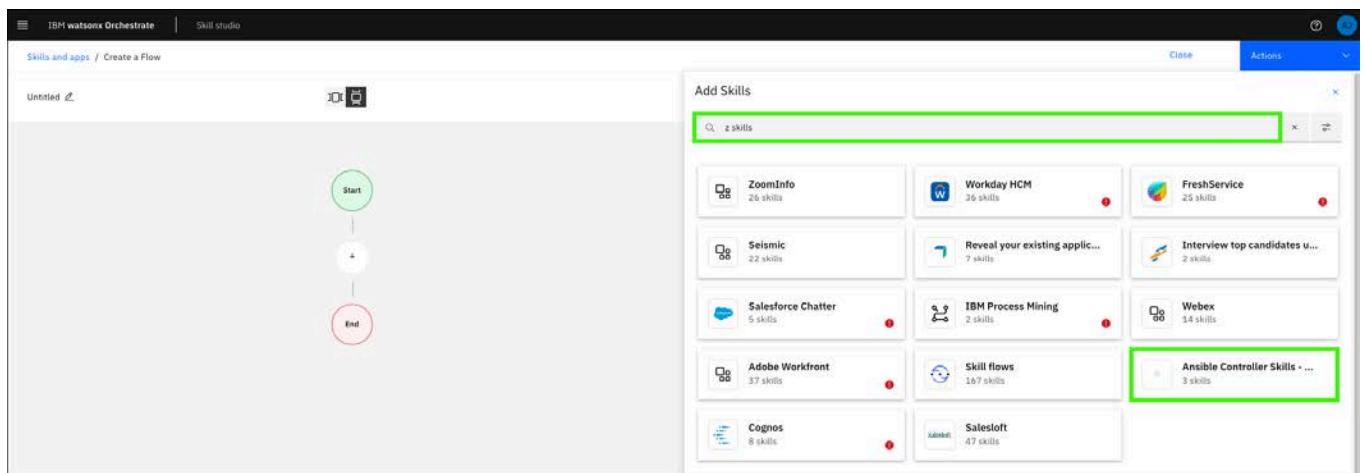


3. Click the + icon.

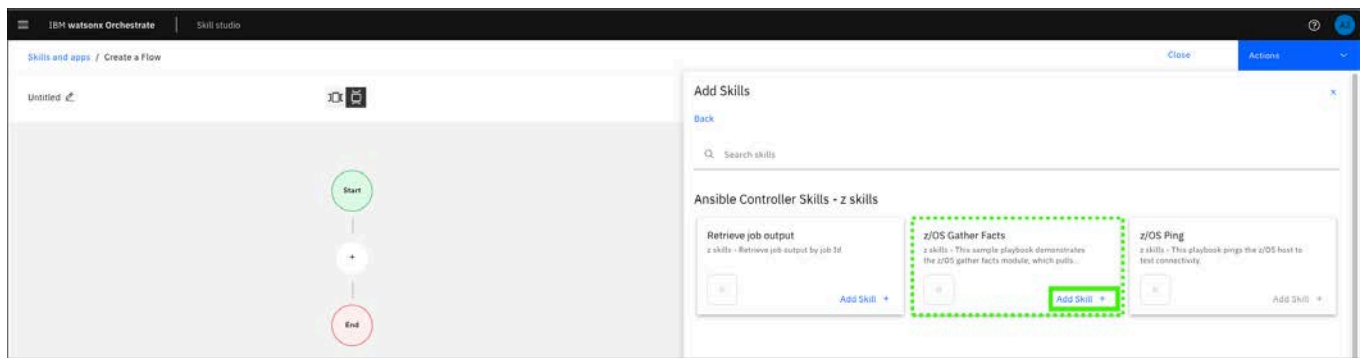


Next, you need to add the **z/OS Gather Facts** skill and the **Retrieve job output** skill to the skill flow. Use the **Search apps** function to locate the skills.

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



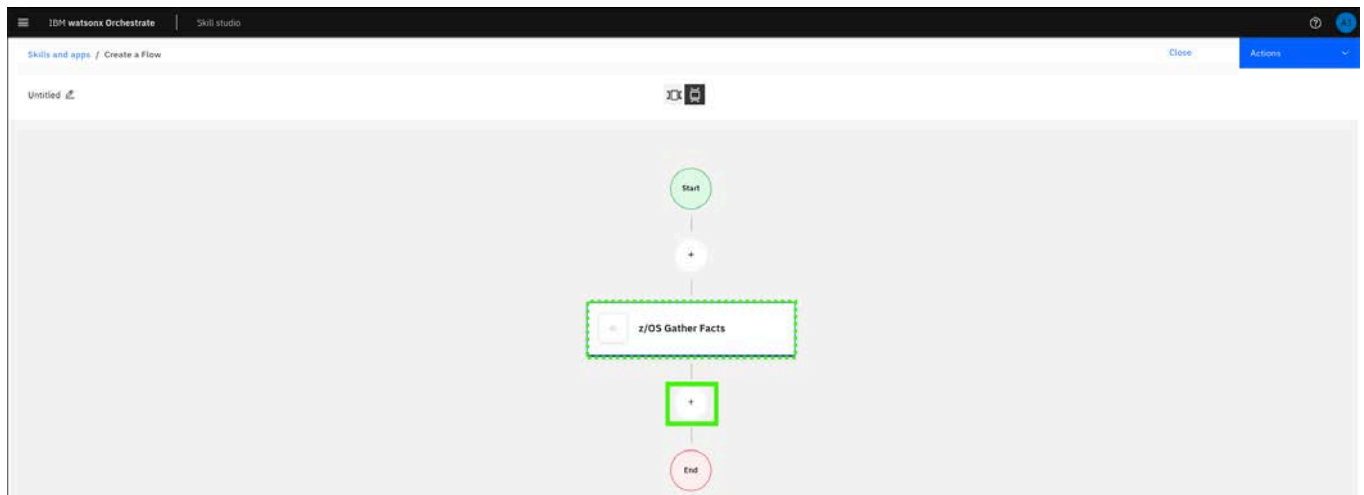
5. Click **Add Skill** in the **z/OS Gather Facts** tile.



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

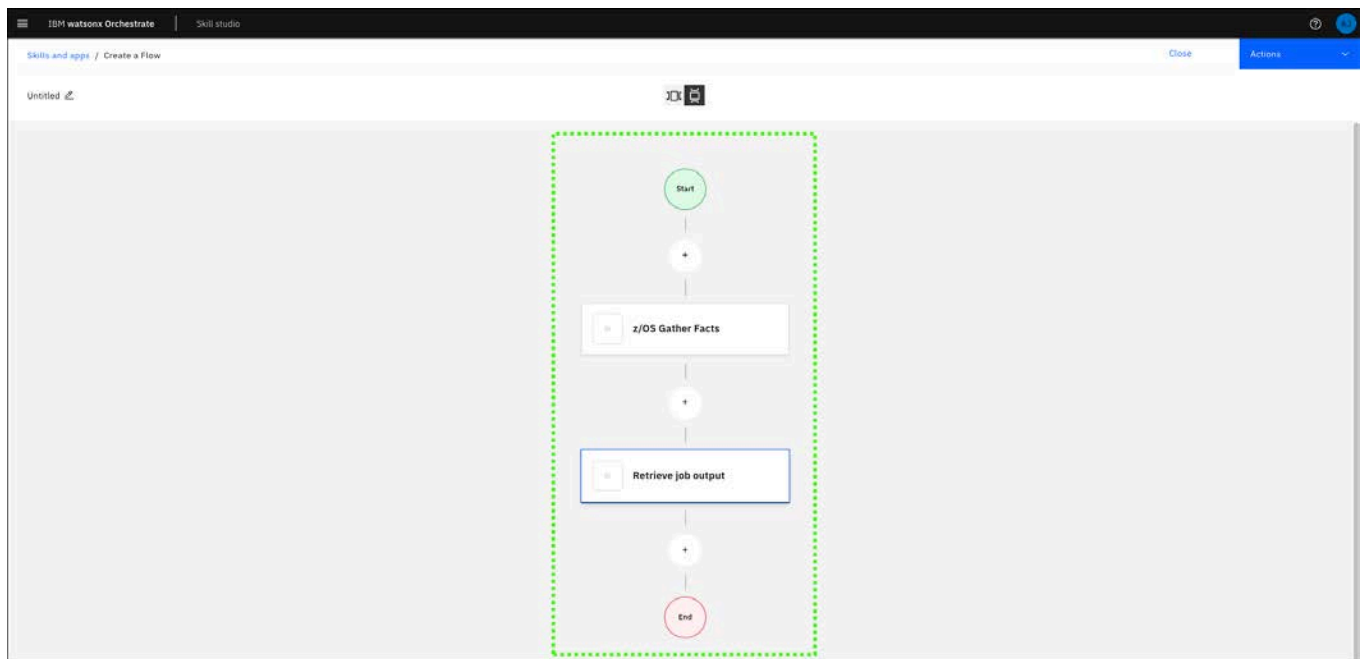


7. Click the **+** icon **after** the **z/OS Gather Facts** tile.



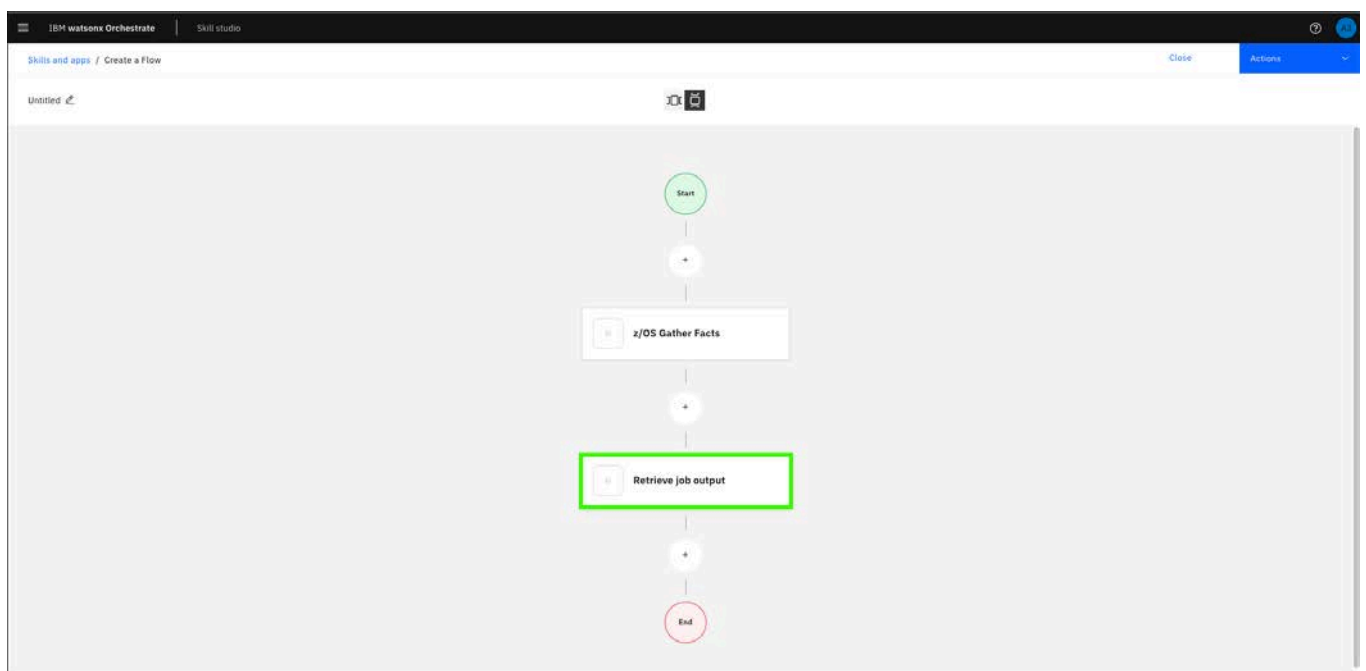
8. Repeat steps 5 and 6 for the **Retrieve job output** skill.

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

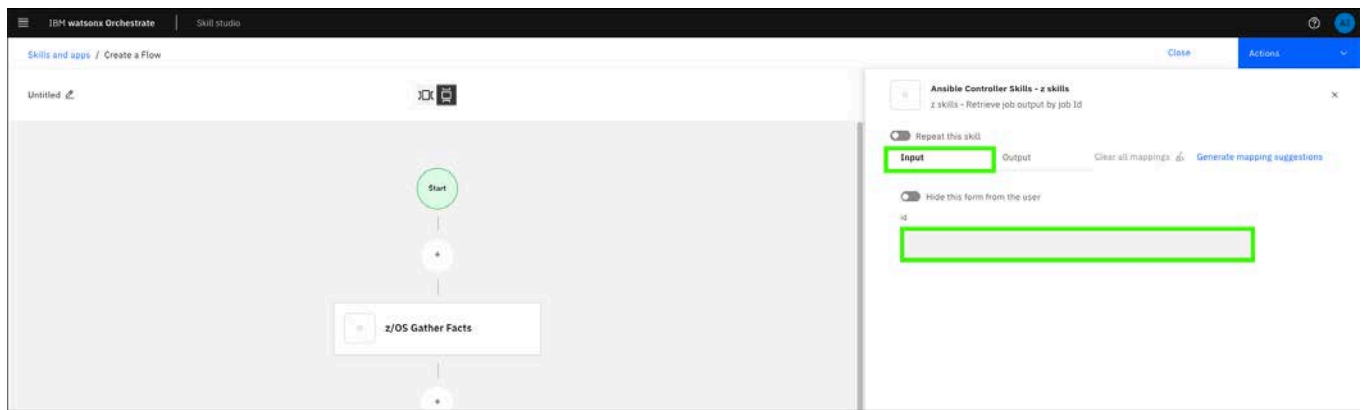


Next you must map the output values of the first skill to the input of the second skill. In this case, pass the job ID output from **z/OS Gather Facts** as an input for **Retrieve job output**.

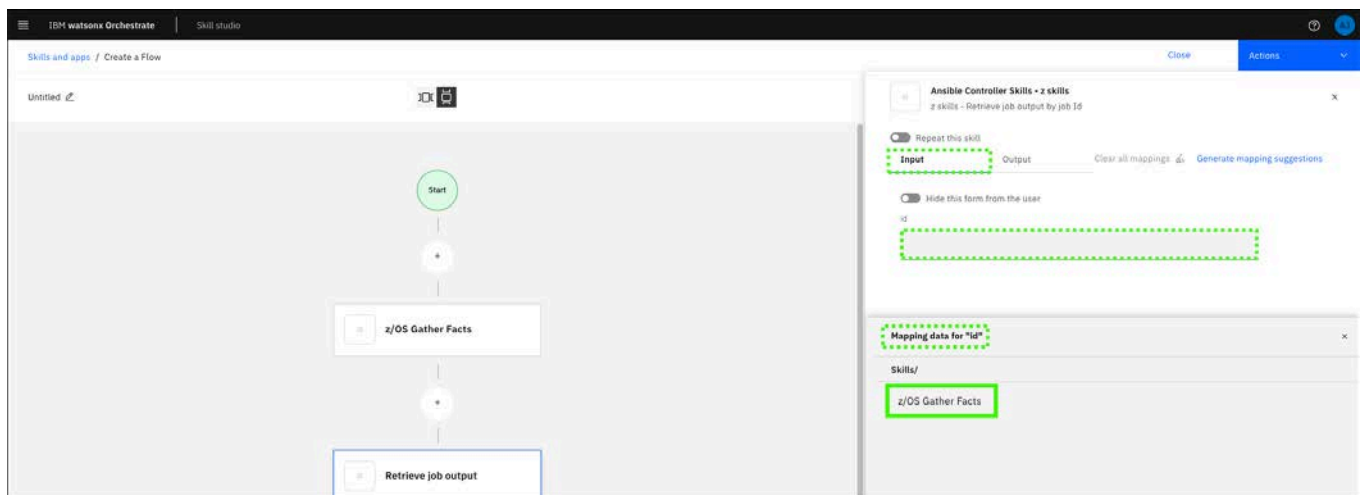
9. Click the **Retrieve job output** tile.



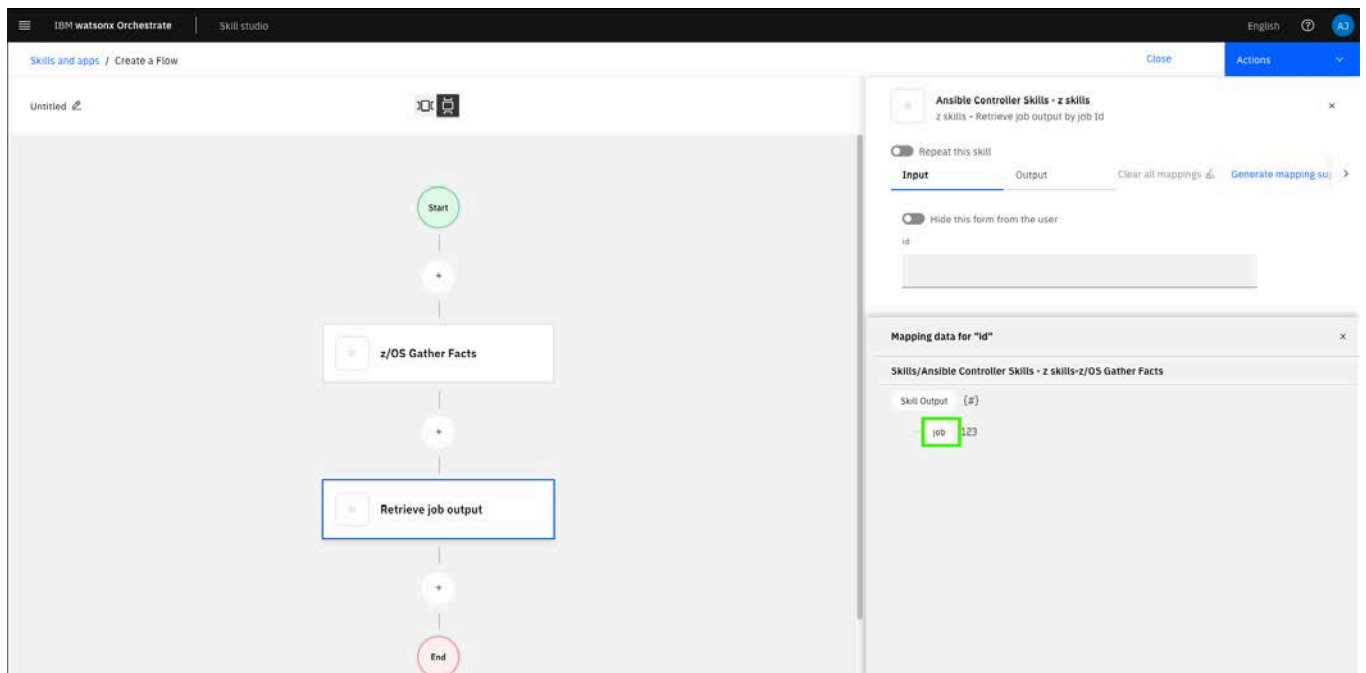
10. Select the **Input** tab and click the **id** field.



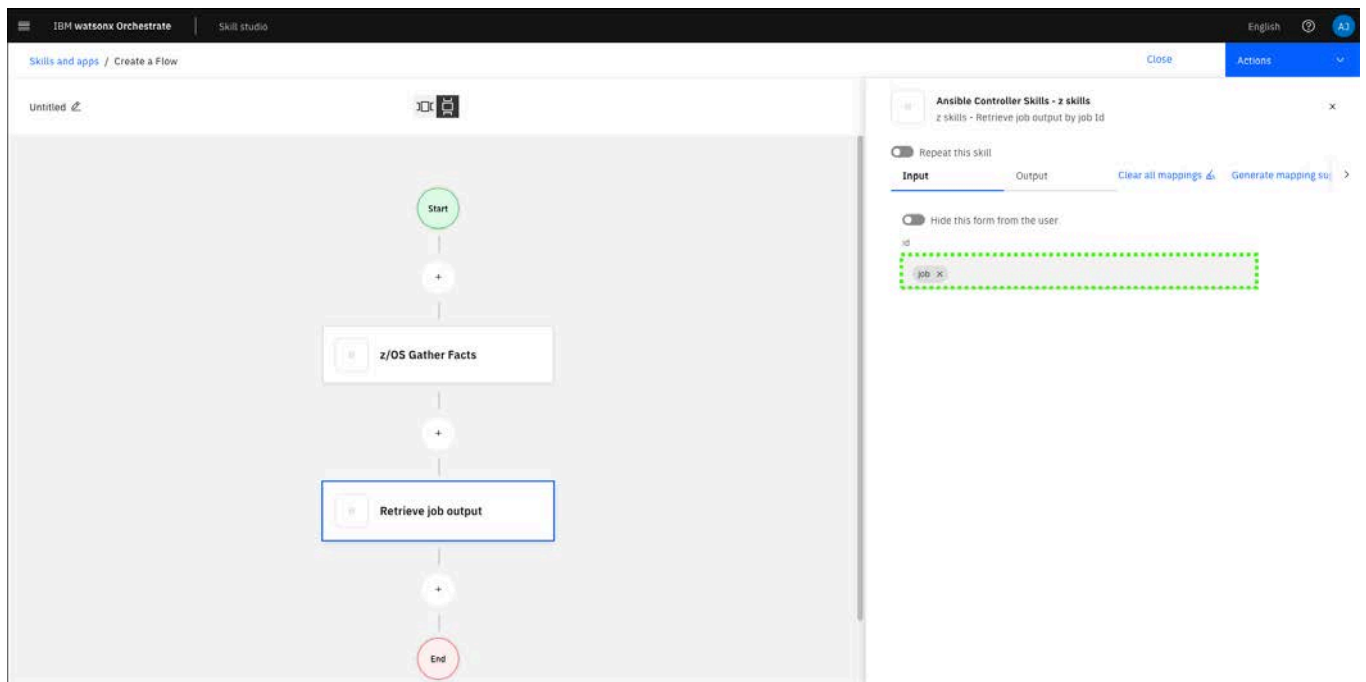
11. Click the **z/OS Gather Facts** skill in the **Mapping data for "id"** section.



12. Click the **job** icon.

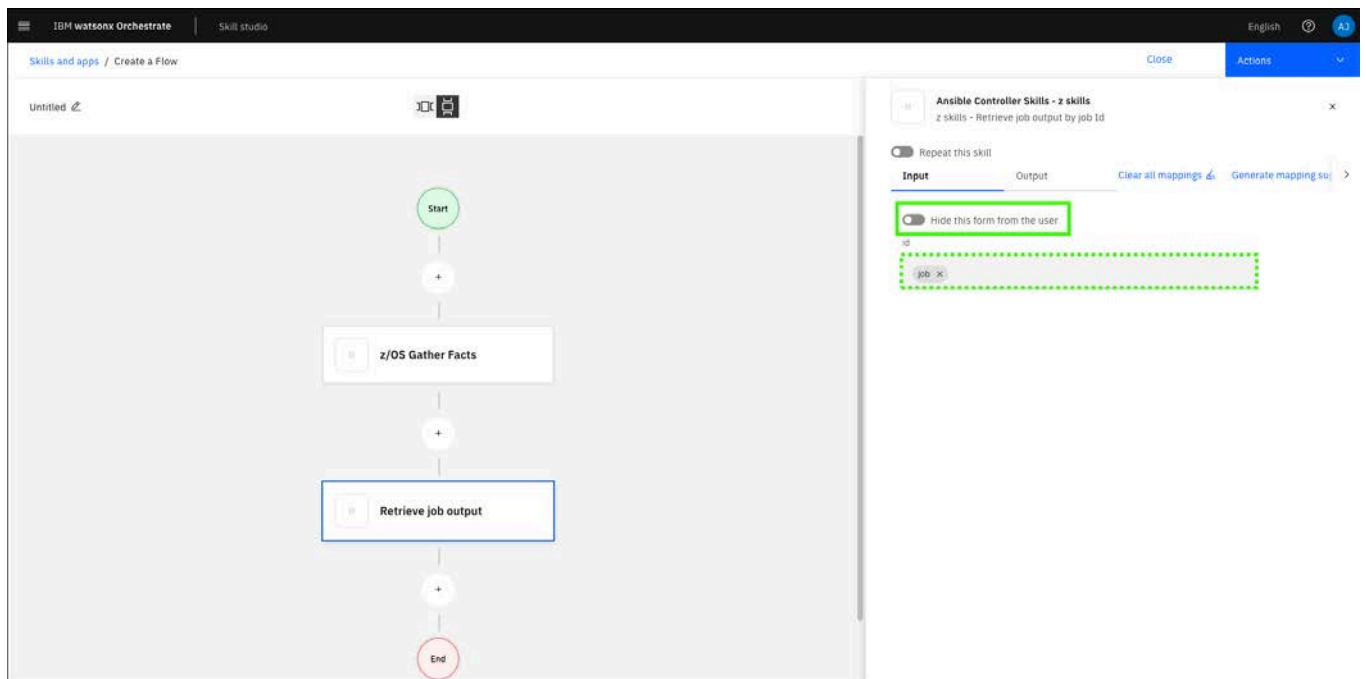


13. Verify that the **job** appears in the **id** field.

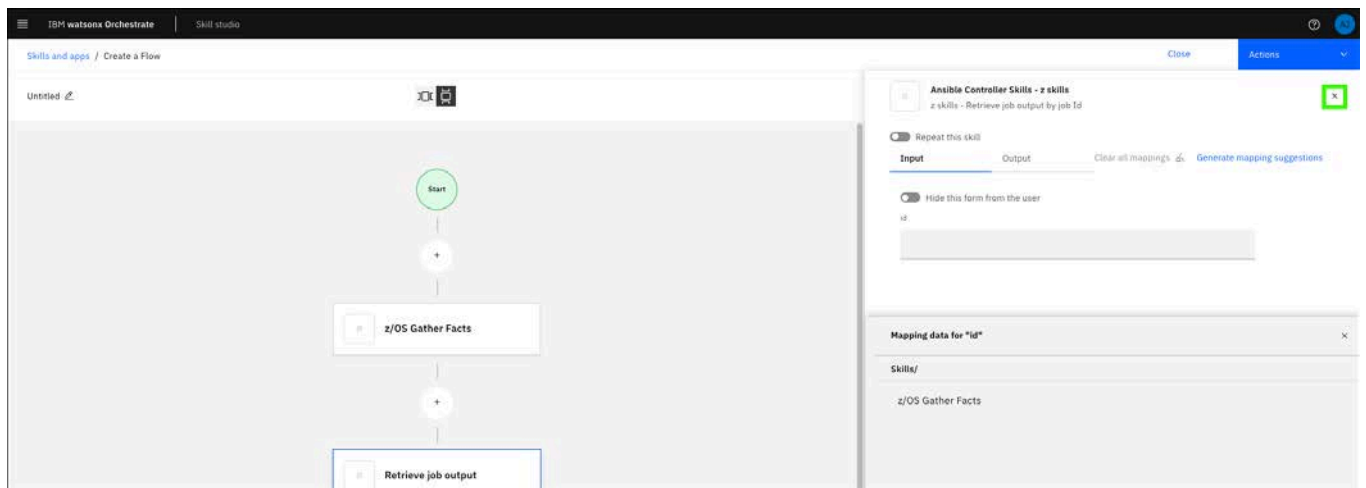


14. Do not toggle/enable the **Hide this from the user** setting as it is needed in order to display the full output of the previous skill.

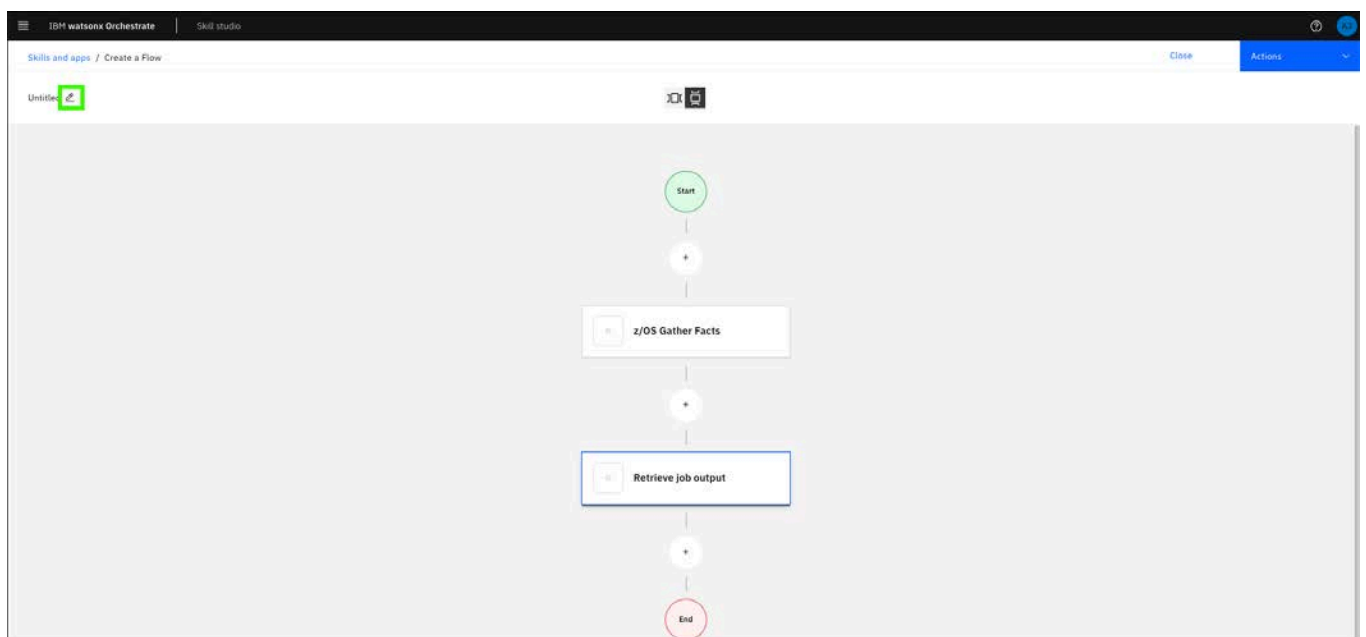
For this lab guide, this option is left disabled. Learn more about this option [here](#).



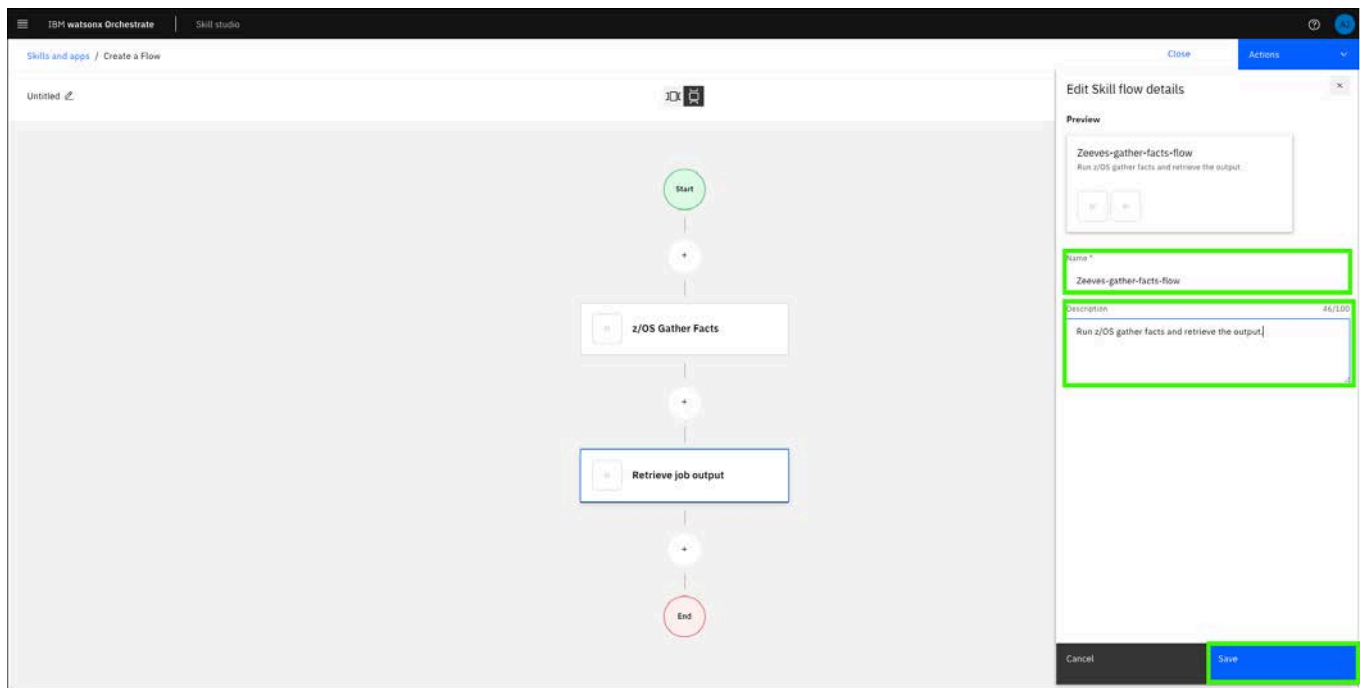
15. Click the **x** to close mapping window.



16. Click the pencil (✎).



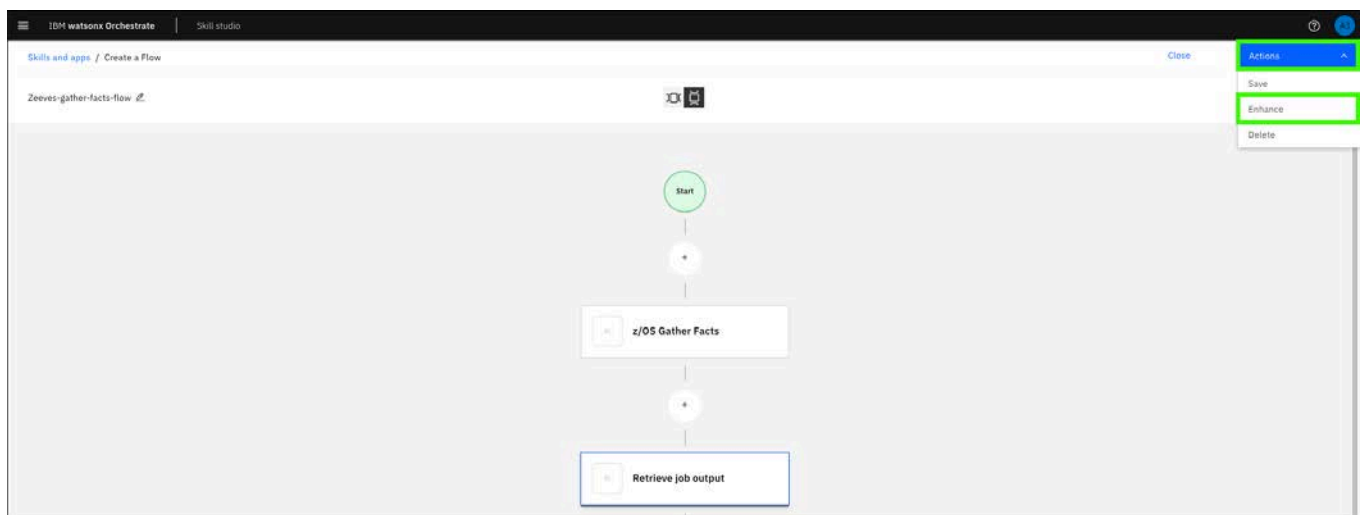
17. Enter a (a) **Name** and (b) **Description** for your skill flow and then (c) click **Save**.



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



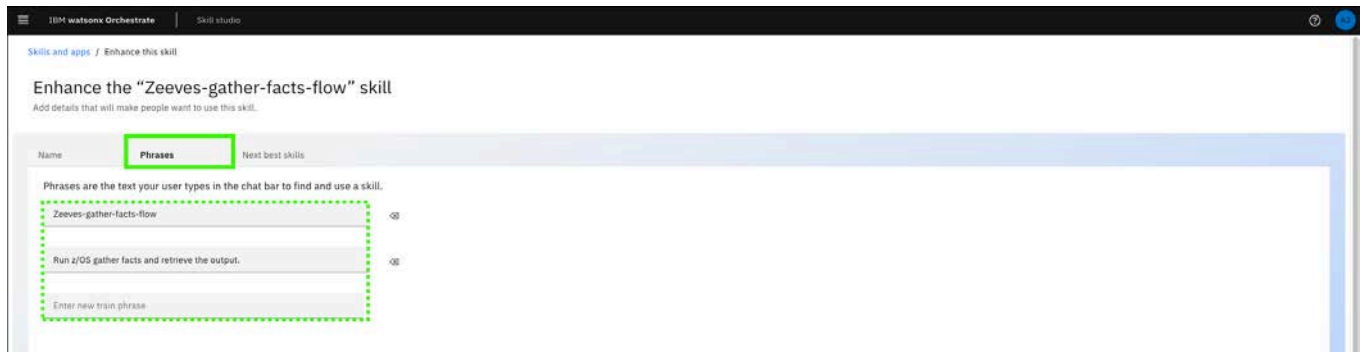
19. Expand the **Actions** pull-down and click **Enhance**.



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

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

20. Click the **Phrases** tab.



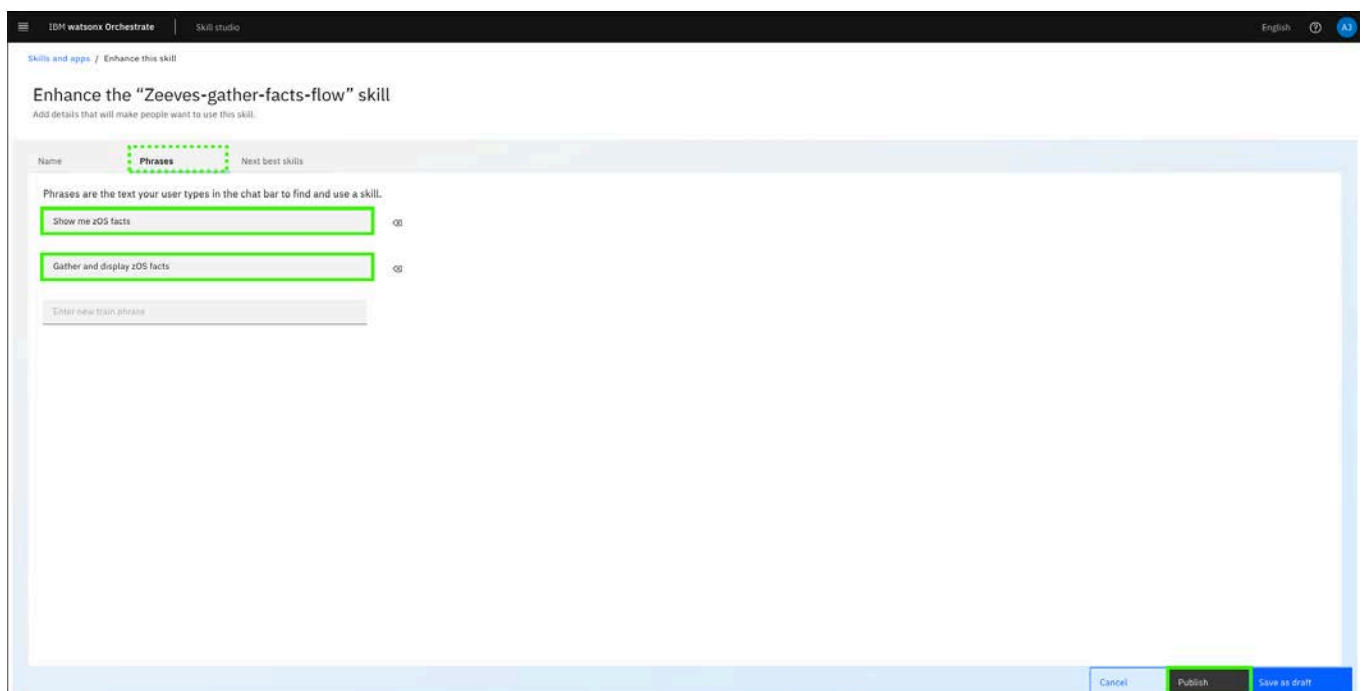
21. Replace the existing **phrases** (prompts) and then click **Publish**.

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

Example prompts:

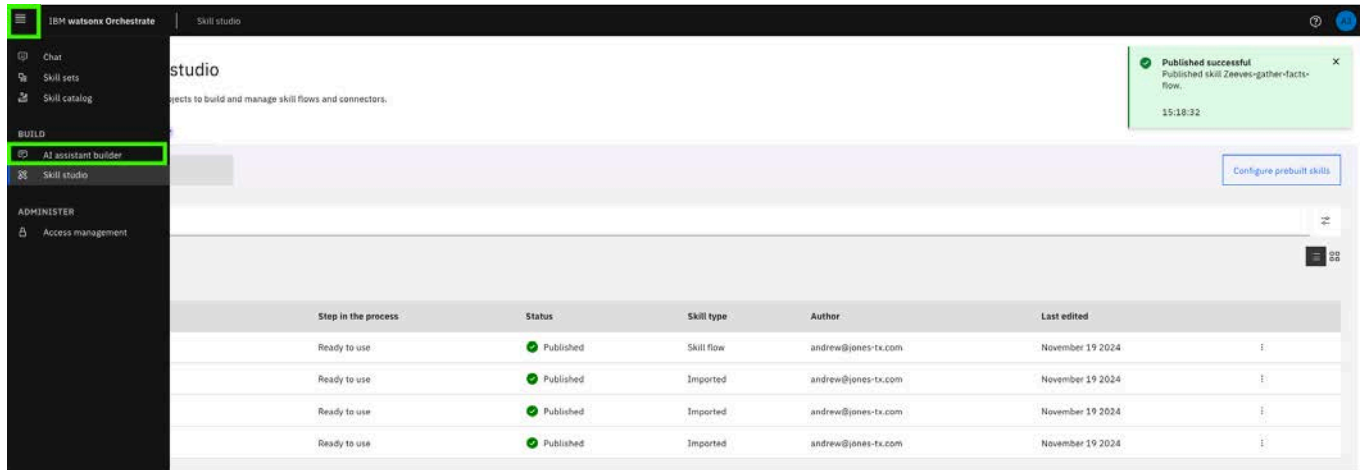
Show me z/OS facts

Gather and display z/OS facts

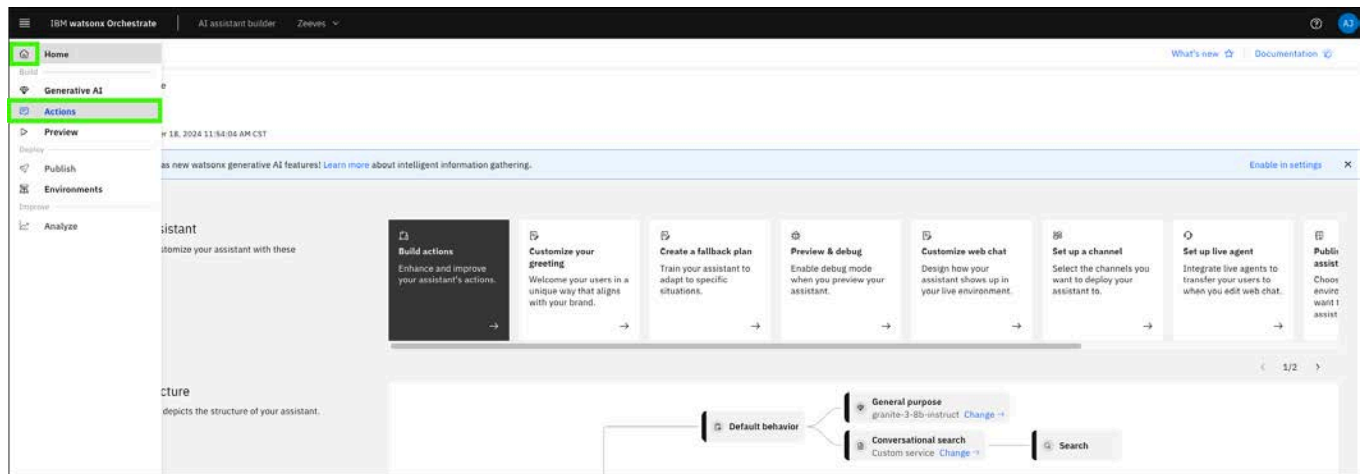


Enable the skill flow in your assistant

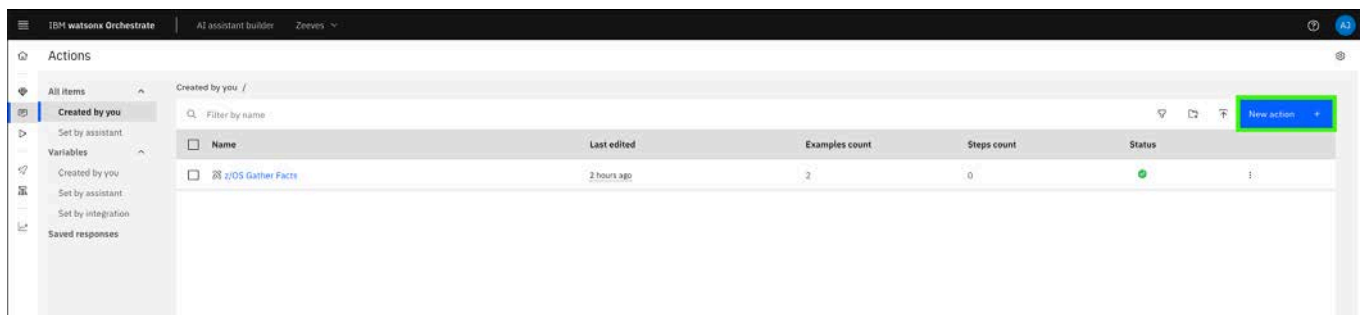
1. Click **AI assistant builder** in the main menu.



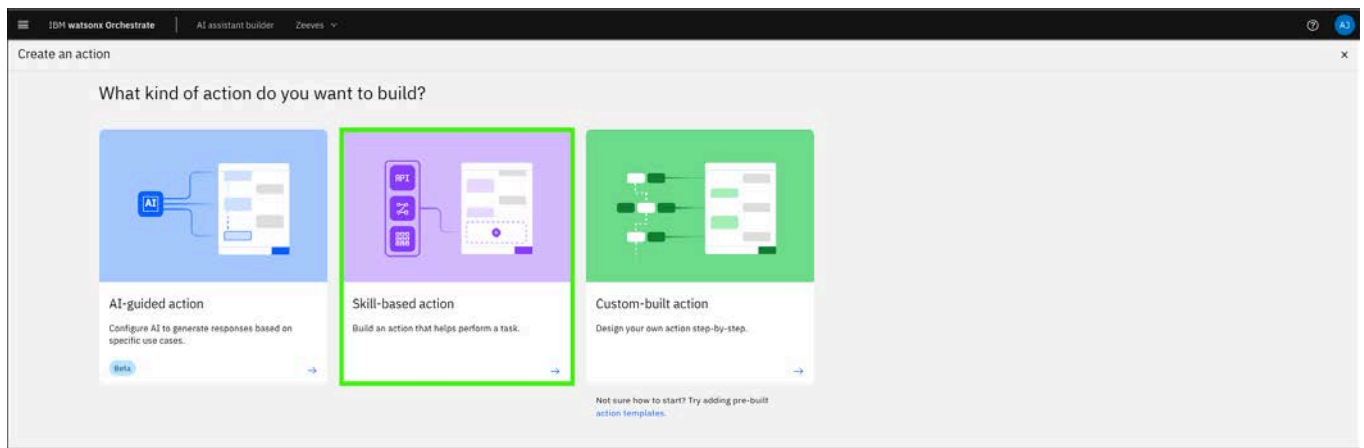
2. Hover over the **Home** (🏠) and click **Actions**.



3. Click **New action**.

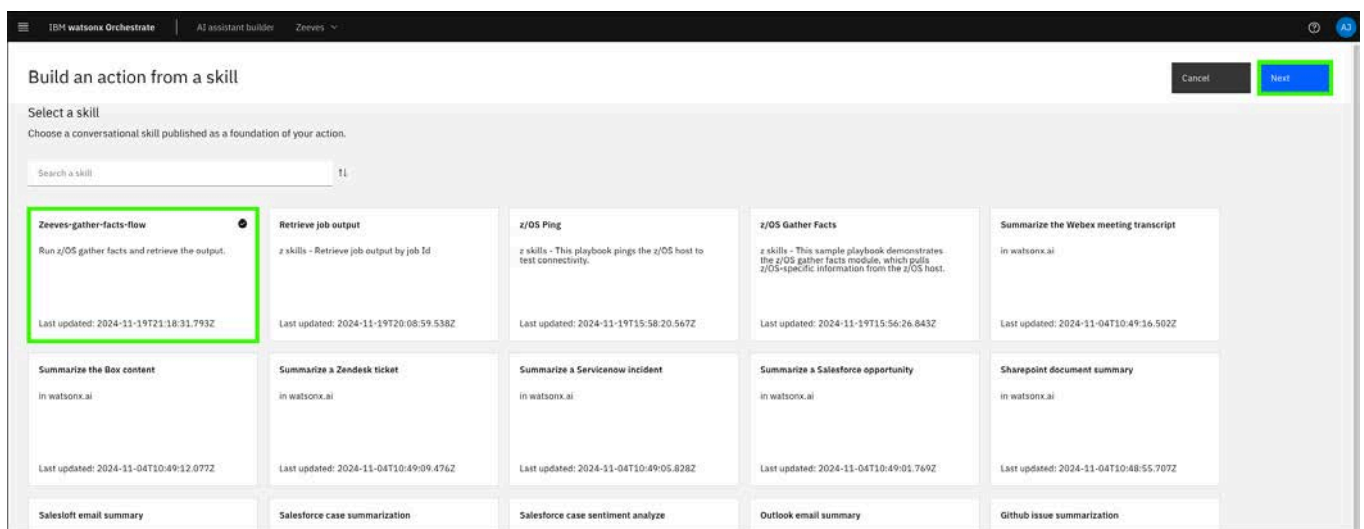


4. Click the **Skill-based action** tile.



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

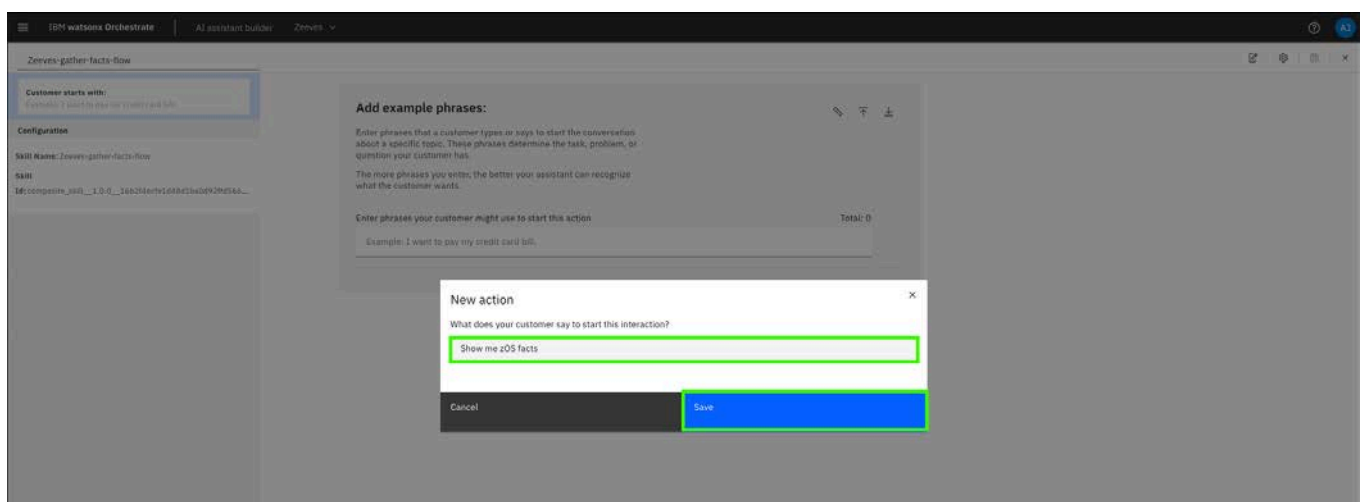
Note: it may take a minute for the tiles to appear on the screen.



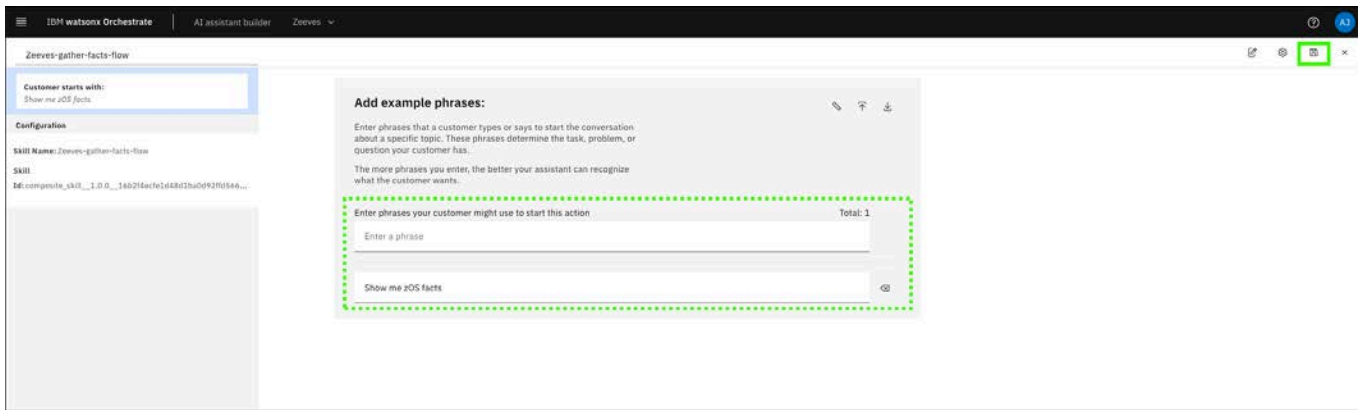
6. Enter an example prompt for the skill and click **Save**.

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

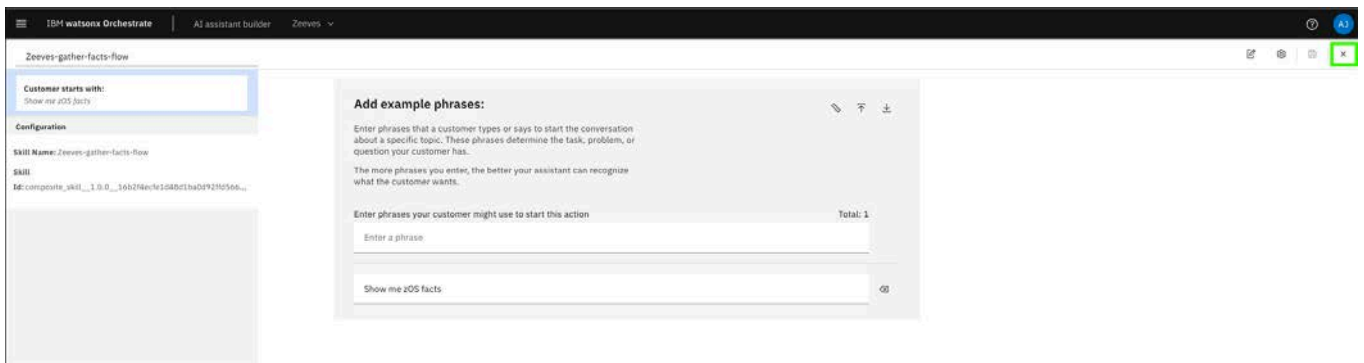
Show me z/OS facts



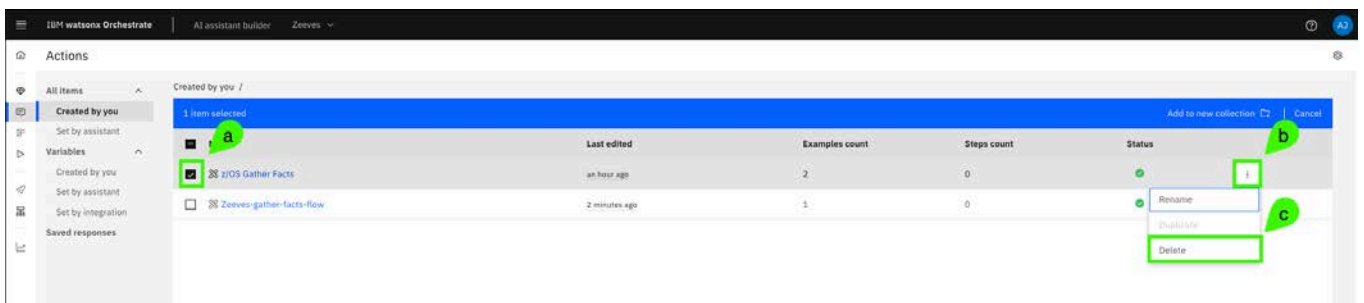
7. Enter any additional phrases (prompts) and then click the **save** (📁).



8. Click close (x).

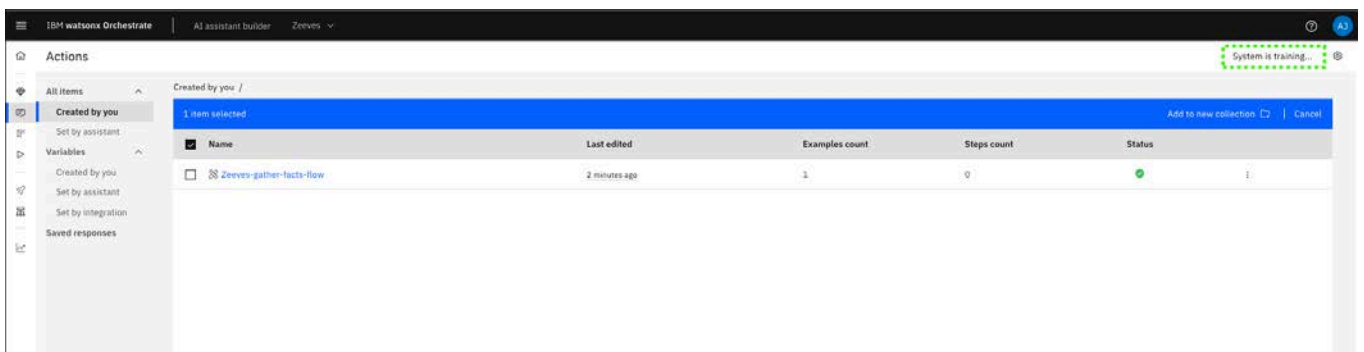


9. Select the *original* skill that you created (a), click the ellipses (b), and then click **Delete** (c).

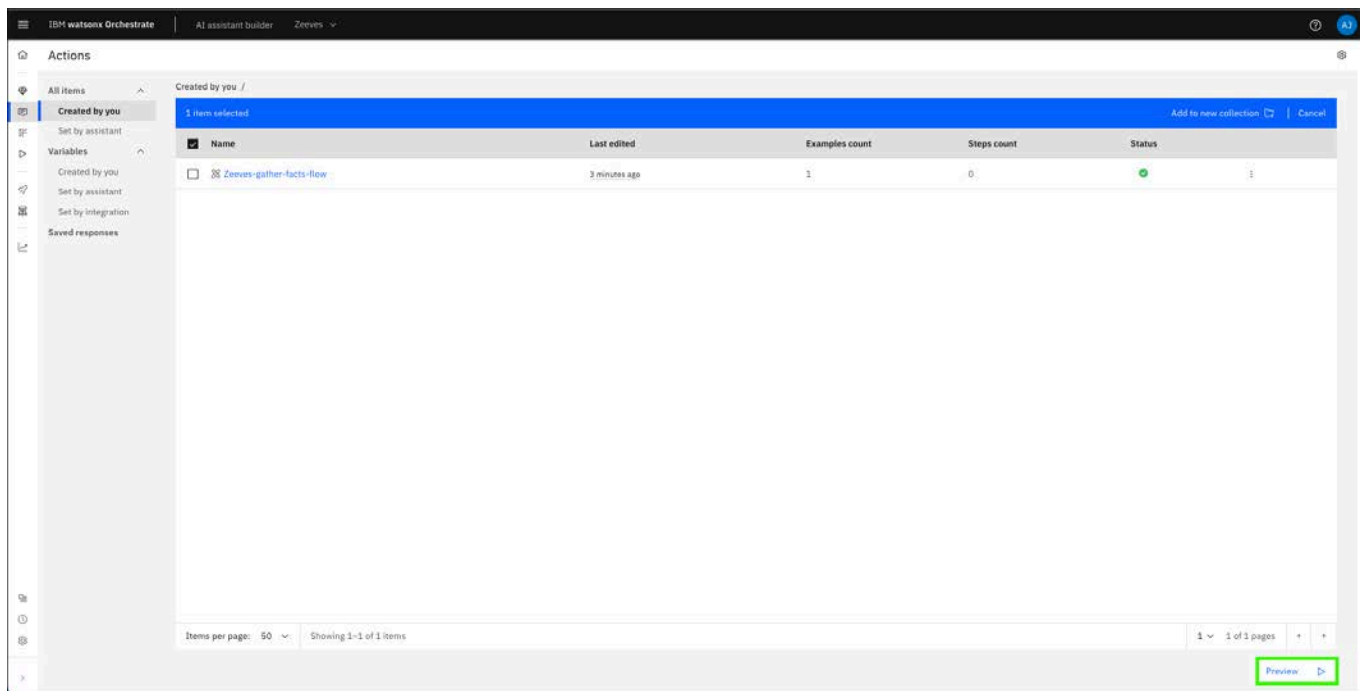


10. Wait for system training to complete.

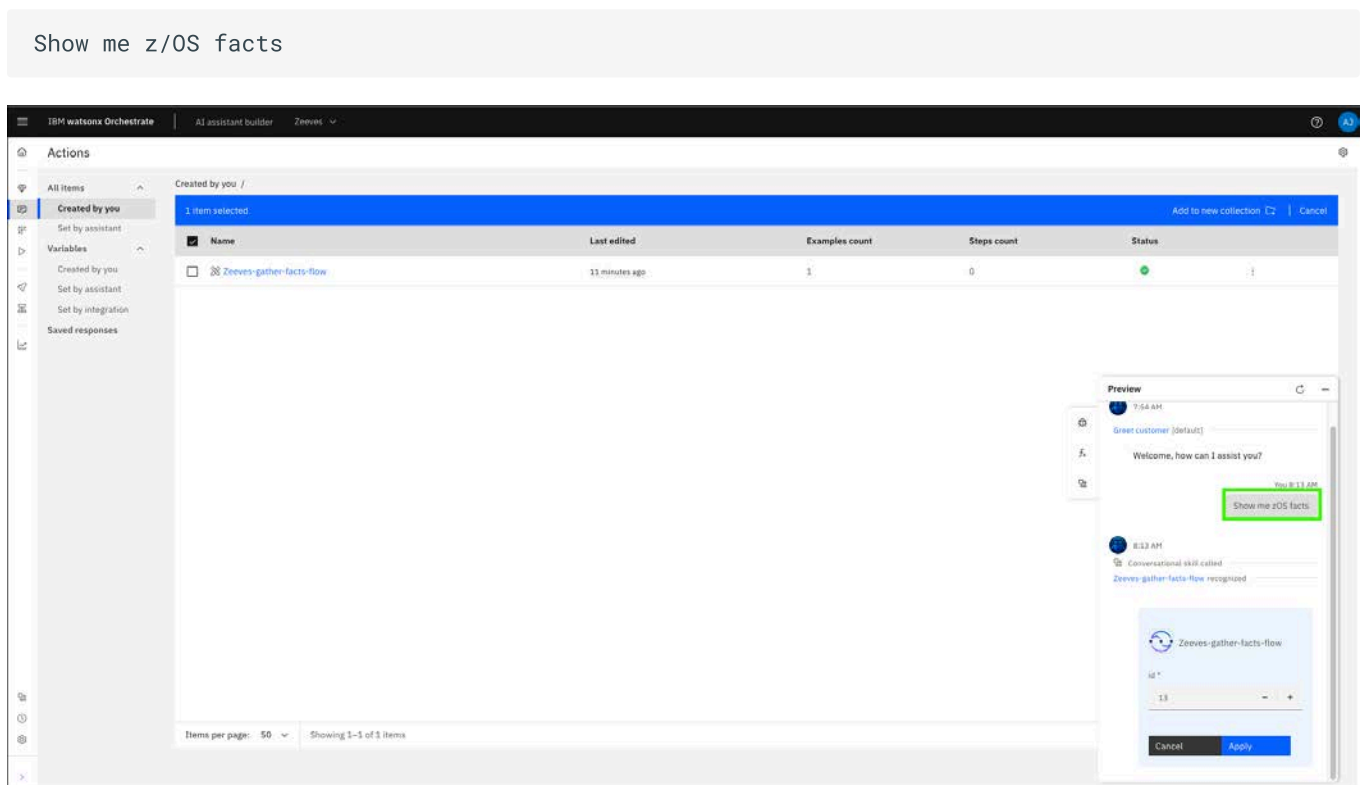
Note: The message changes to "System is trained" and then disappears.



11. Click **Preview**.

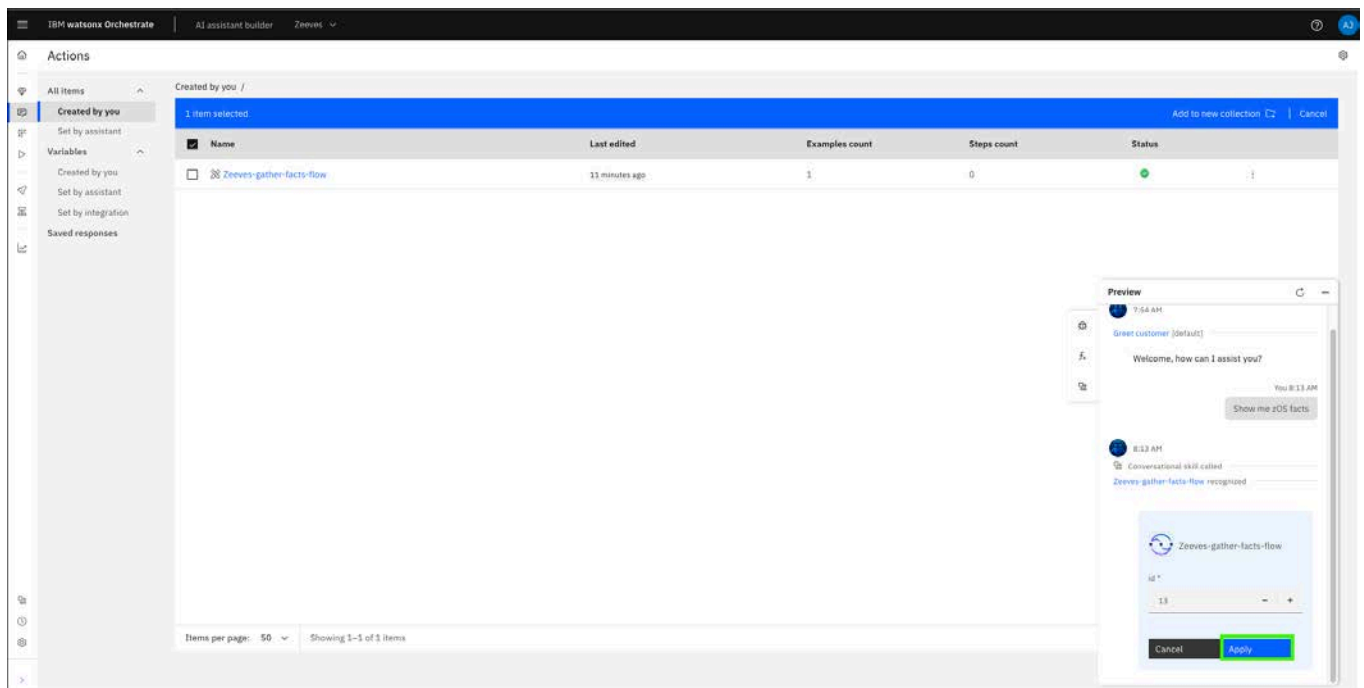


12. Enter one of the prompts you specified into the assistant preview.



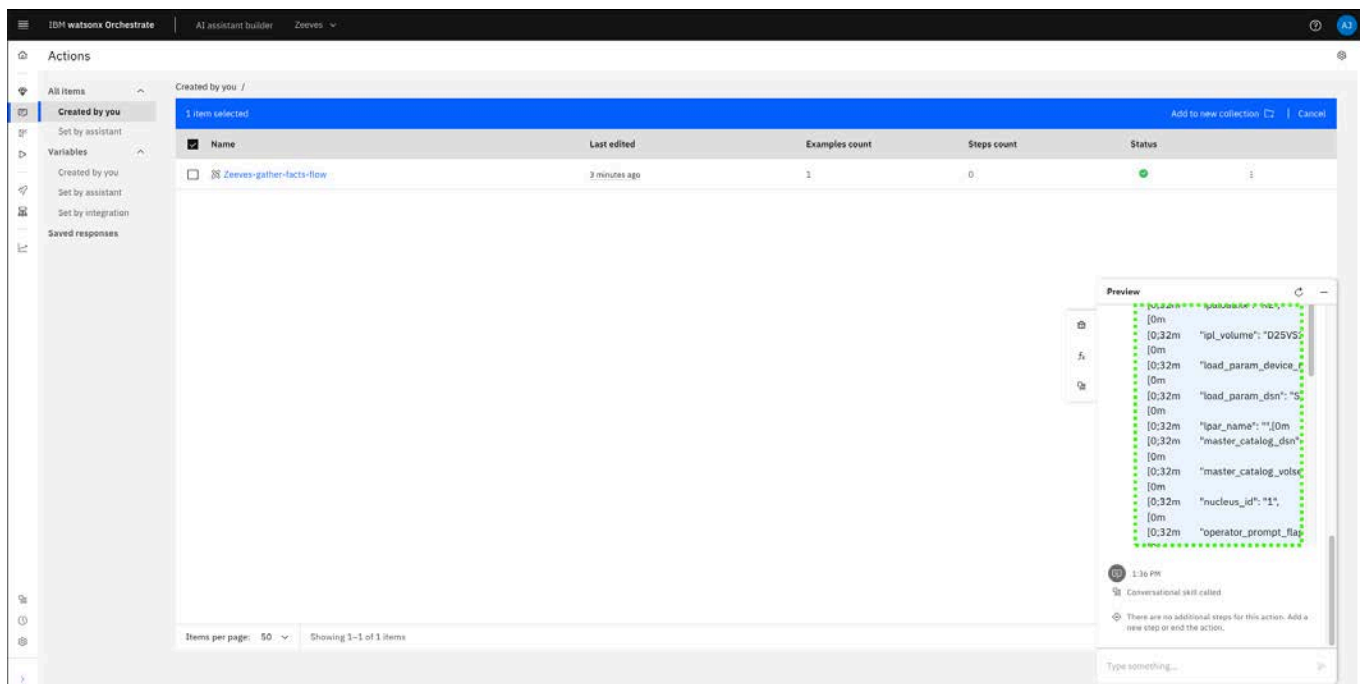
13. Wait 10 seconds and then click **Apply**.

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



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.





Sample output form 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.] ***** [0;32mok: [zos_host][0m

TASK [Print gathered facts about the master catalog.] **** [0;32mok: [zos_host] => {[0m [0;32m  "msg": [[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  ignored=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](#).

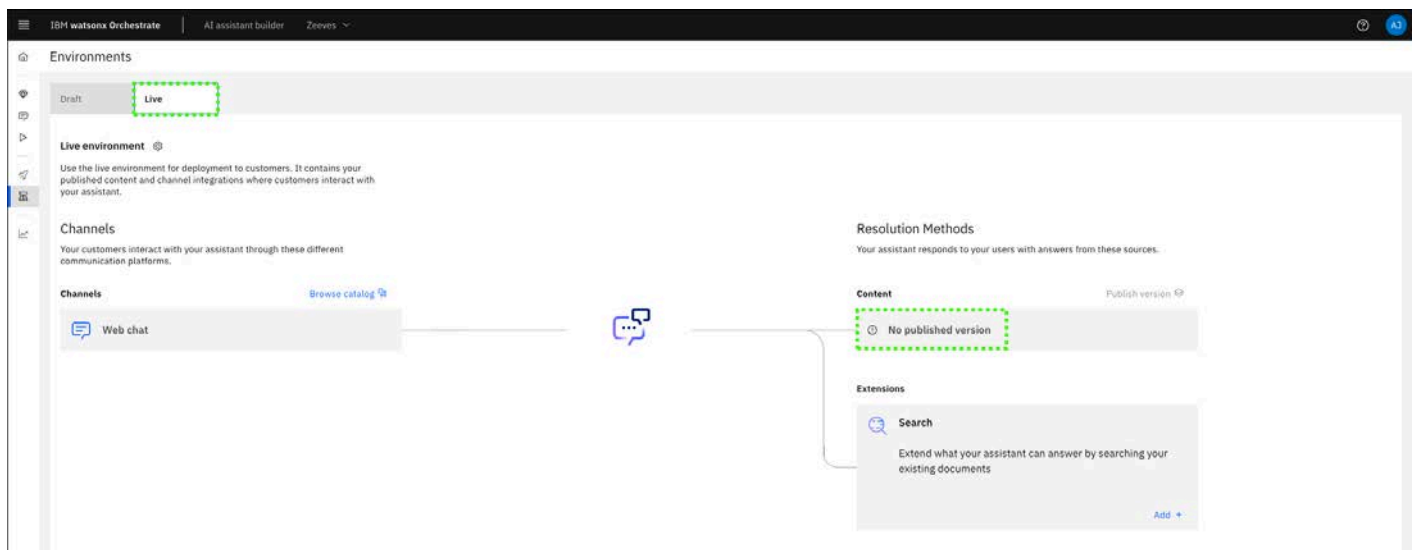
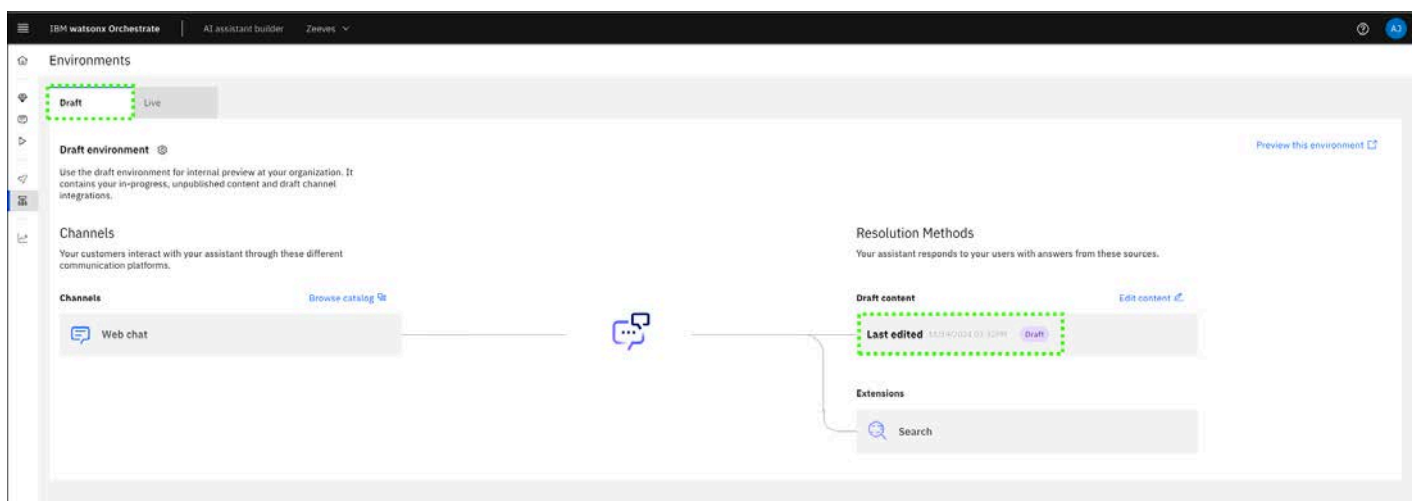
Extract the embedded JSON file and modify the file for your environment by following [these instructions](#).

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**:



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

Publish the assistant

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

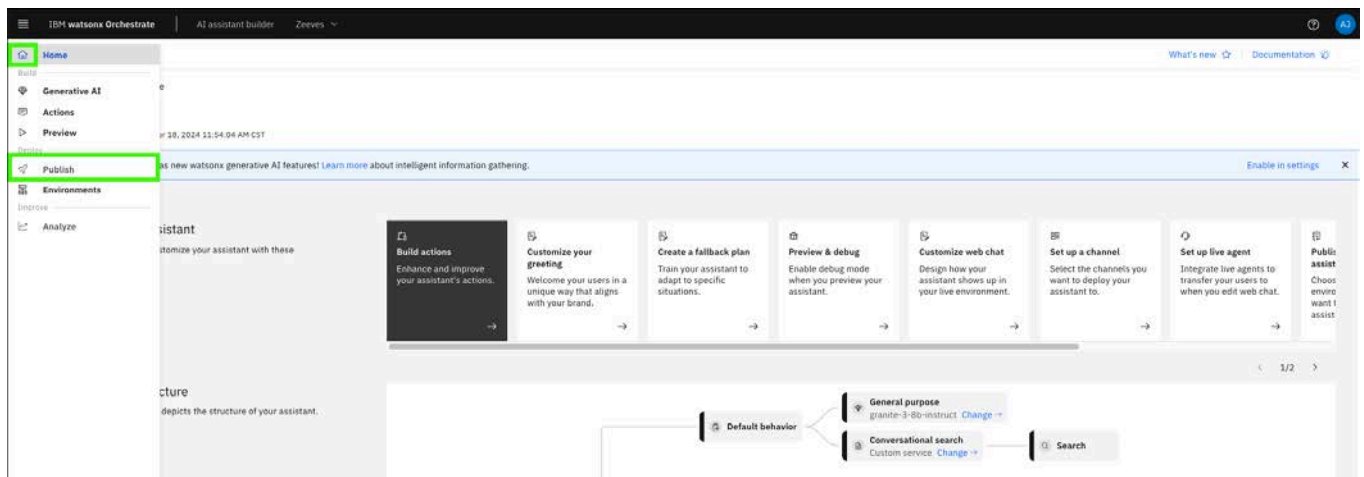
Versions do not contain integration configurations or environment settings

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

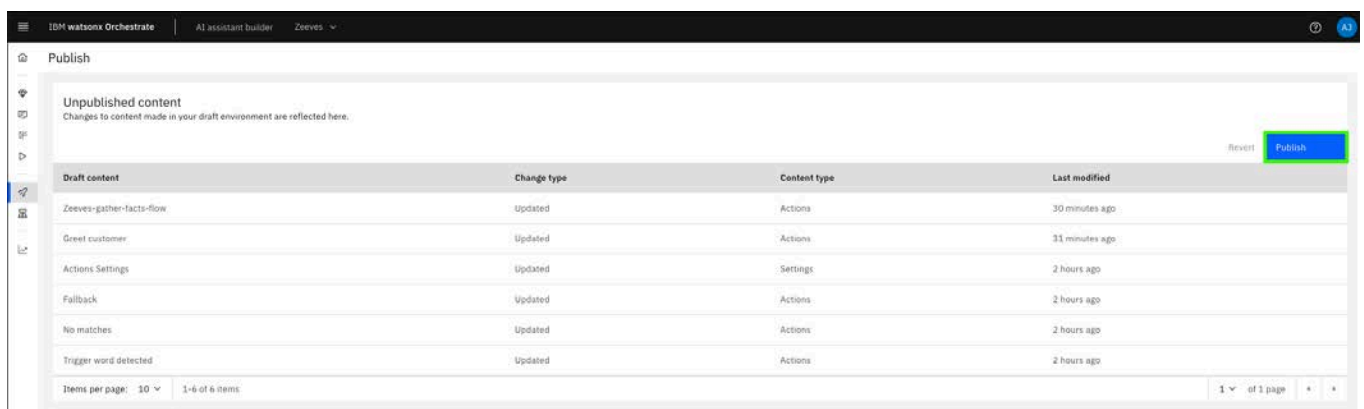
For managing quality-control and versioning, the Live environment is the version of the assistant to give to users.

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

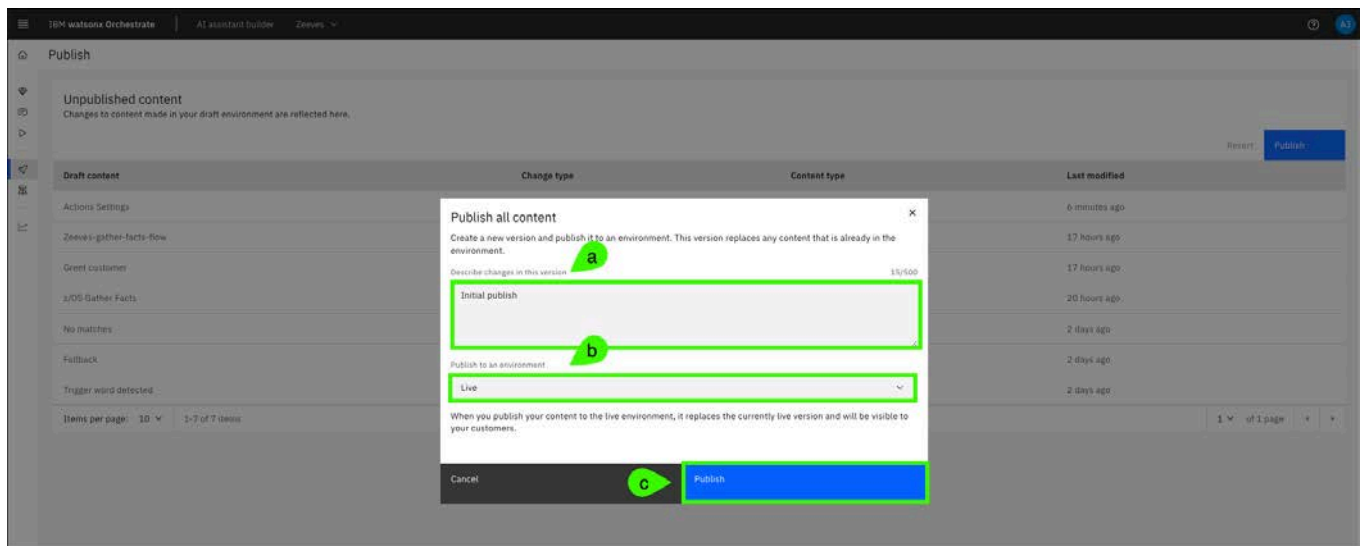
1. Hover over the **Home** icon () and click **Publish**.



2. Click **Publish**.



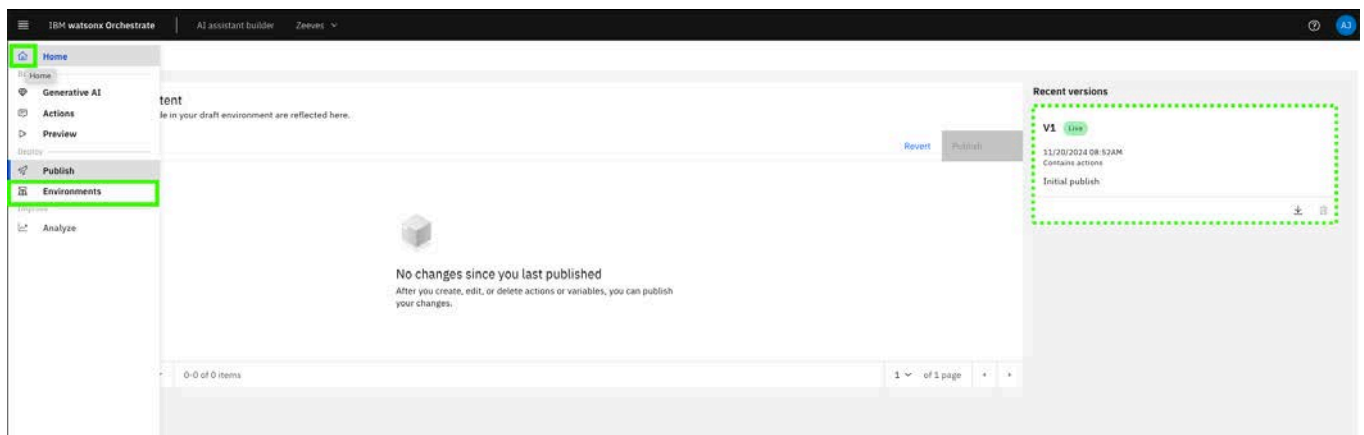
3. Enter a description of the changes (a), set the environment to **Live** (b), and then click **Publish** (c).



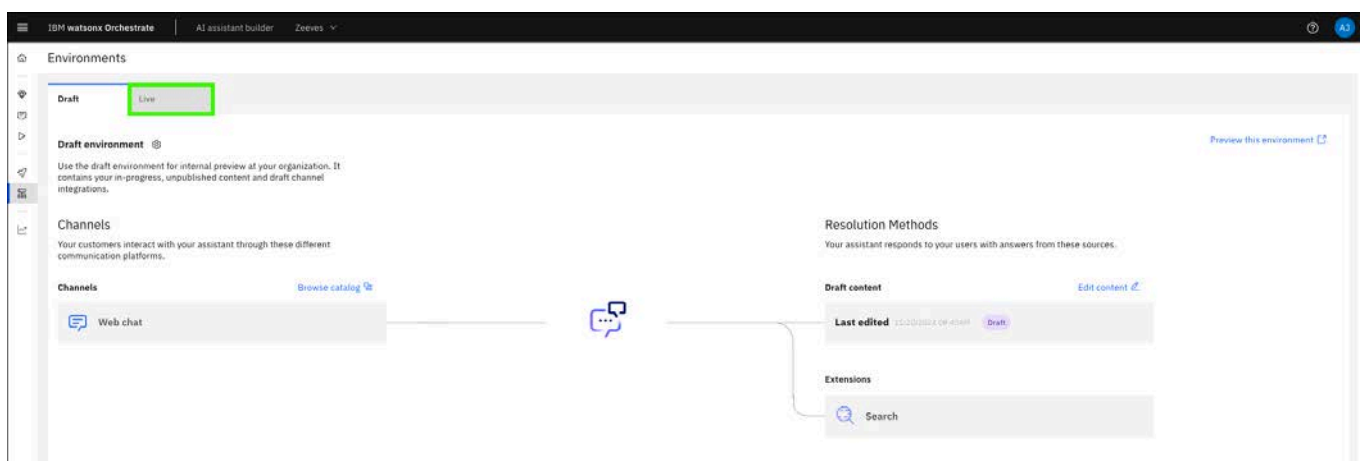
Configure the live environment

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

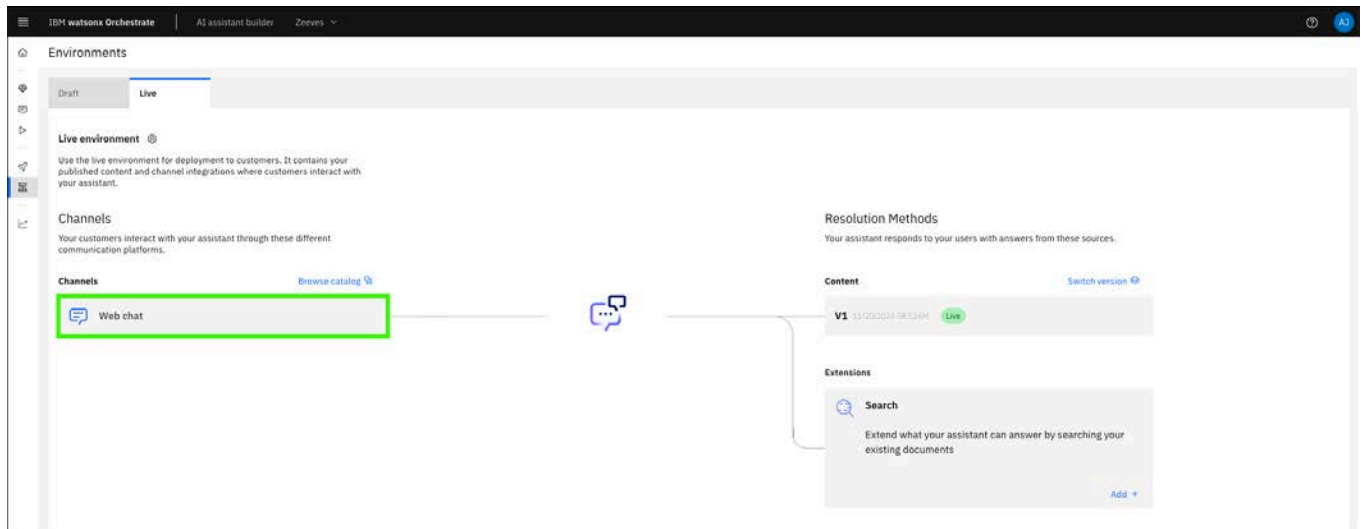
1. Hover over the **Home** icon (🏠) and click **Environments**.



2. Click **Live**.



3. Click **Web chat**.



4. Customize the live assistant as you see fit.

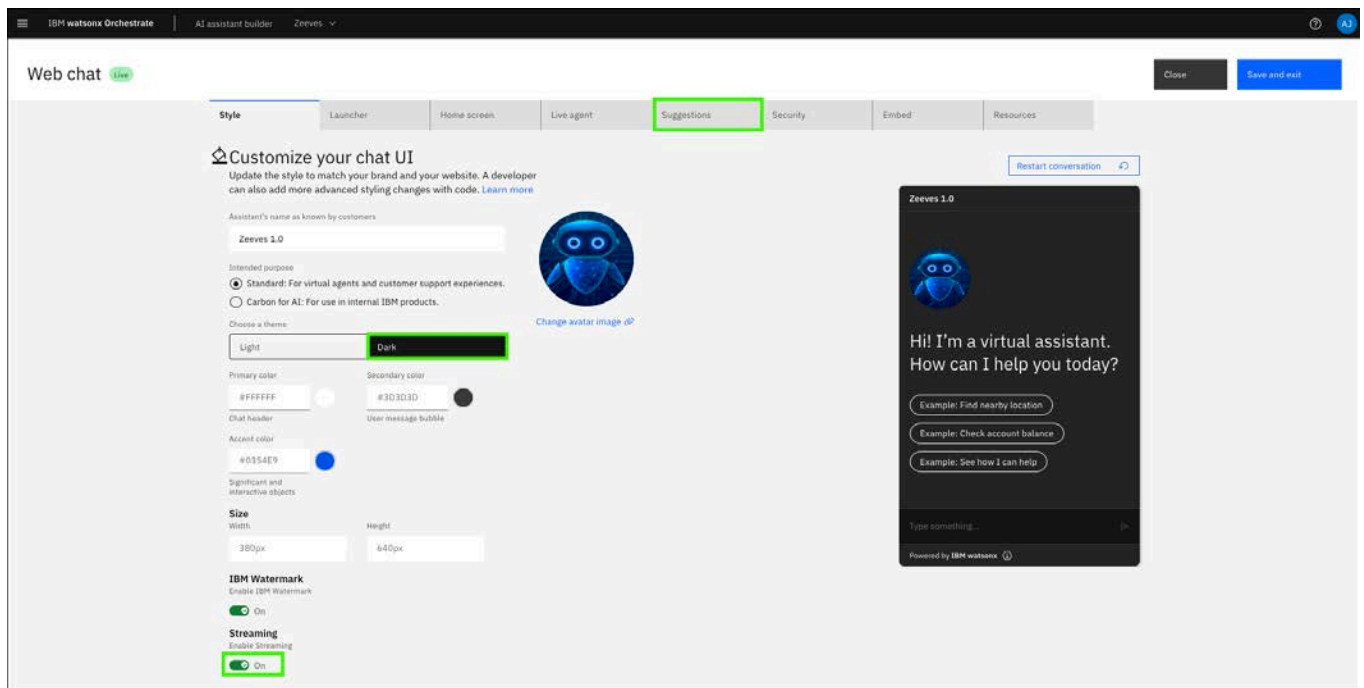
On the **Style** tab, you're able to set the Assistant name that is displayed on the chat window when users are interacting with the assistant. For pilots or demos, consider personalizing this name for the client. Also in the **Style** tab, you can set the themes and display settings of the chat windows,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

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

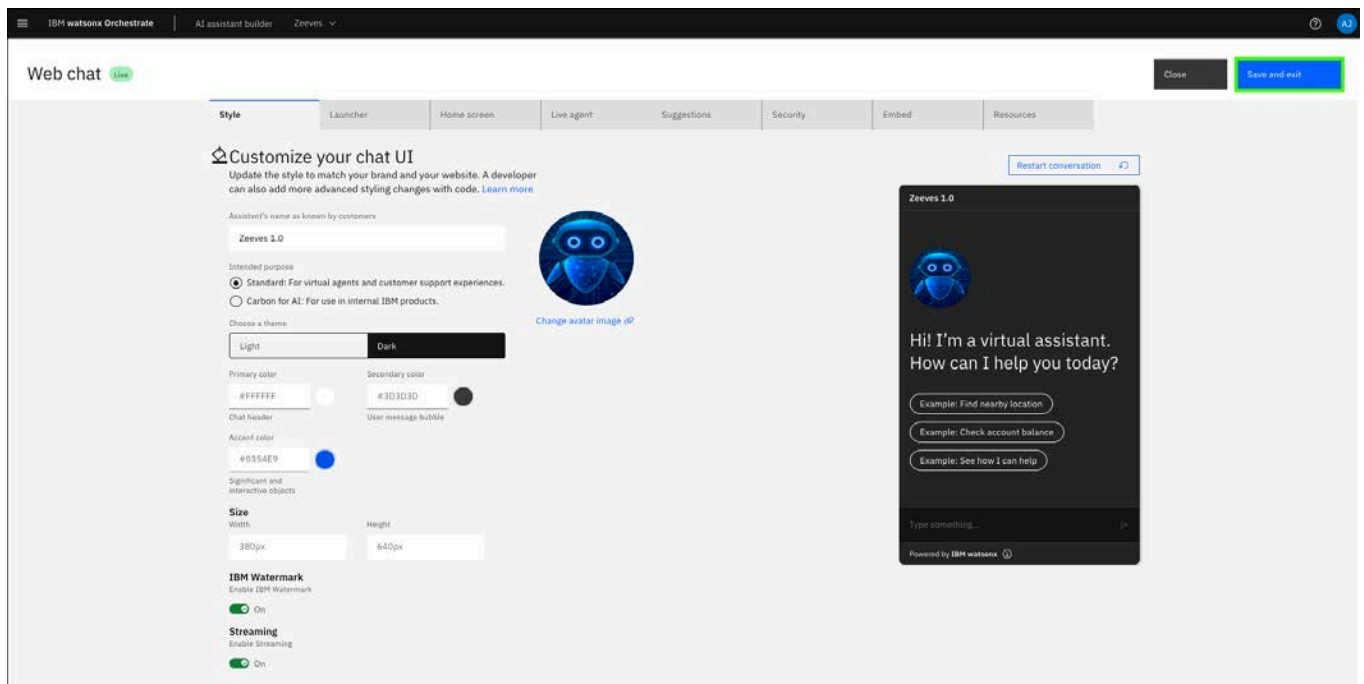
Explore all the other tabs.

Customize your live environment.

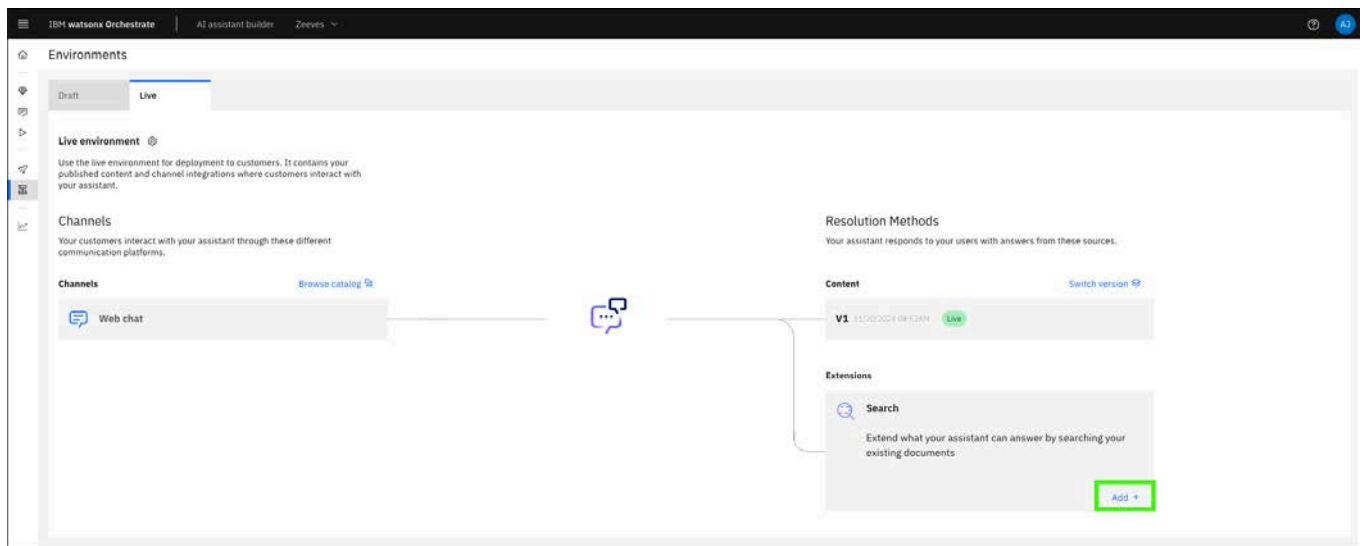
For this lab, toggle **Streaming** on and turn **Suggestions** off on the **Suggestions** tab. You may also want to change the theme to **Dark** to differentiate your draft and live environments.



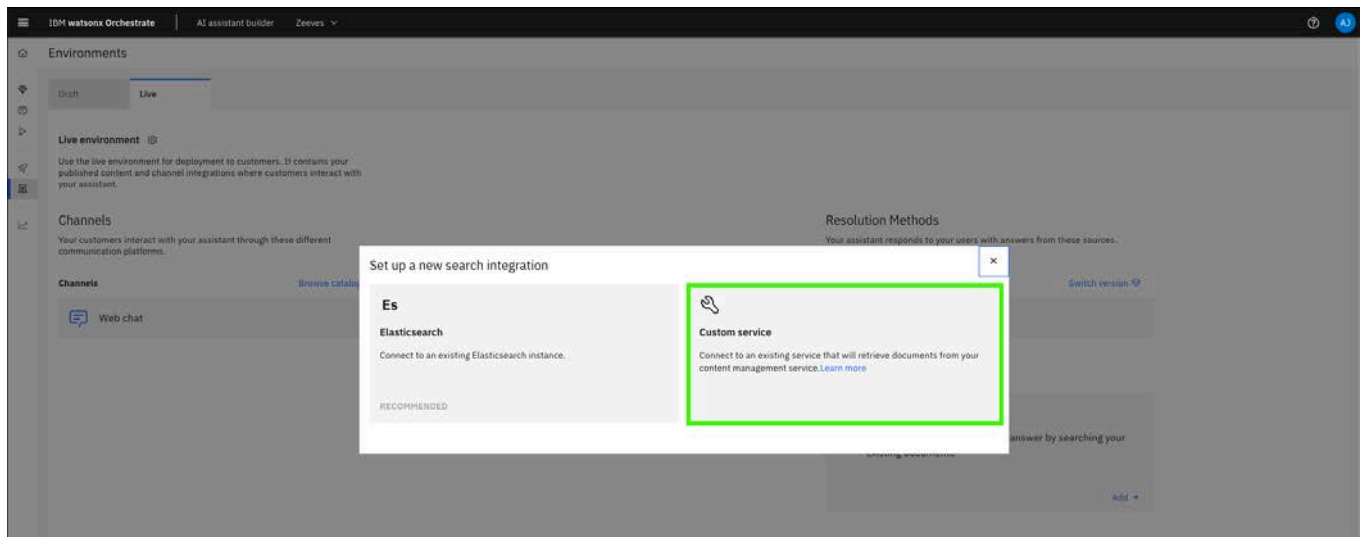
5. Click **Save and exit**.



6. Click **Add** in the **Search** tile.



7. Click **Custom service**.



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](#).

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

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



Customize the settings.

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

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

Custom service Live

Settings Instance

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.

Conversational search

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

Conversational search

☒ On

Search mode

☒ 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

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

Retrieval confidence threshold

☒ Lowest ☐ Low ☐ High ☐ Highest

Generated response length

☐ Concise ☐ 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

Default filter

Metadata

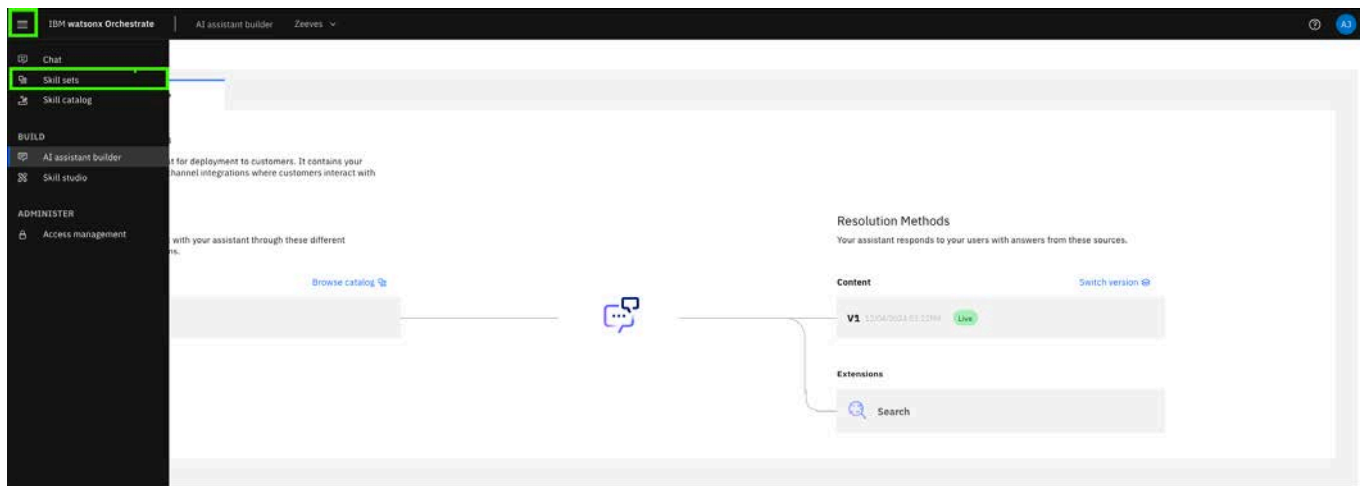
Search display text

Define the text your search will display to the end user

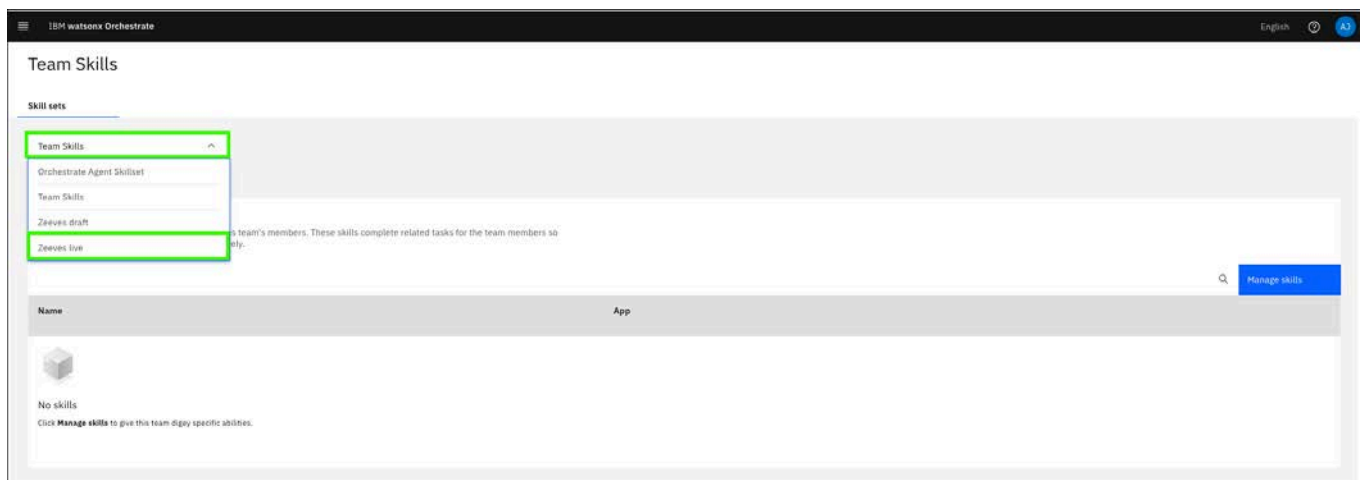
No results found [Connectivity issue](#)

Connect the skills to the live environment

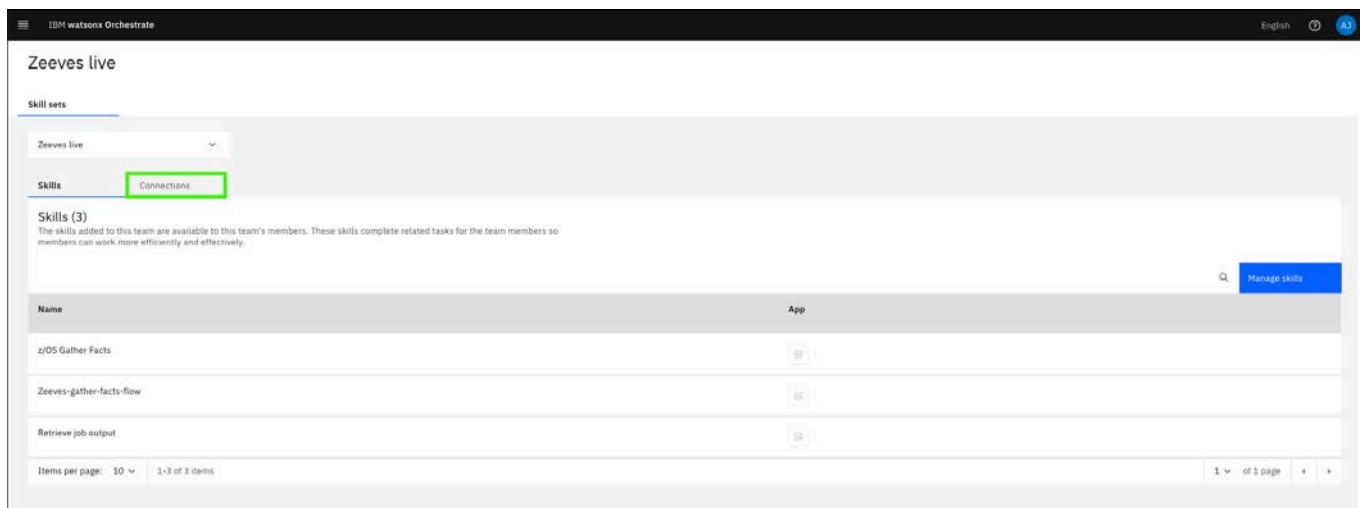
1. Click **Skill sets** in the main menu.



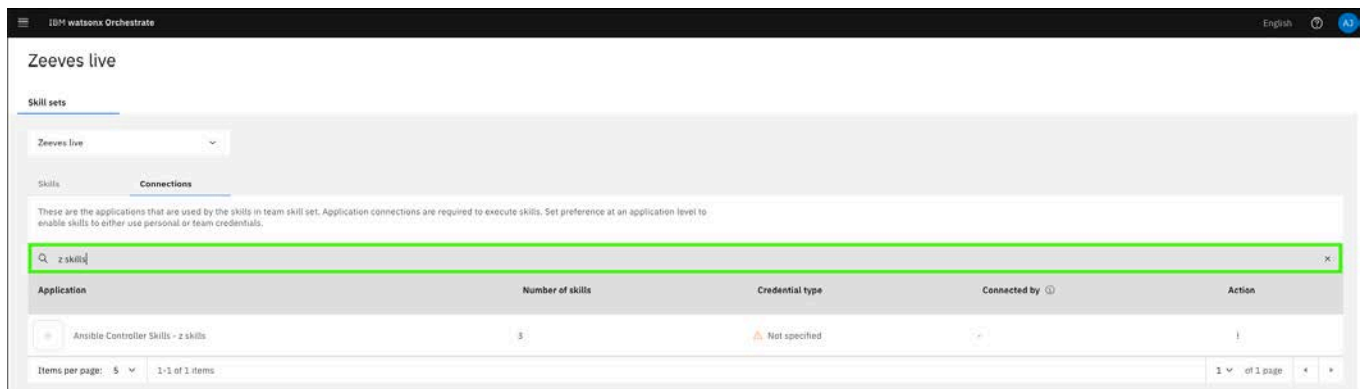
2. Select your assistant's live instance in the **Skill sets** list.



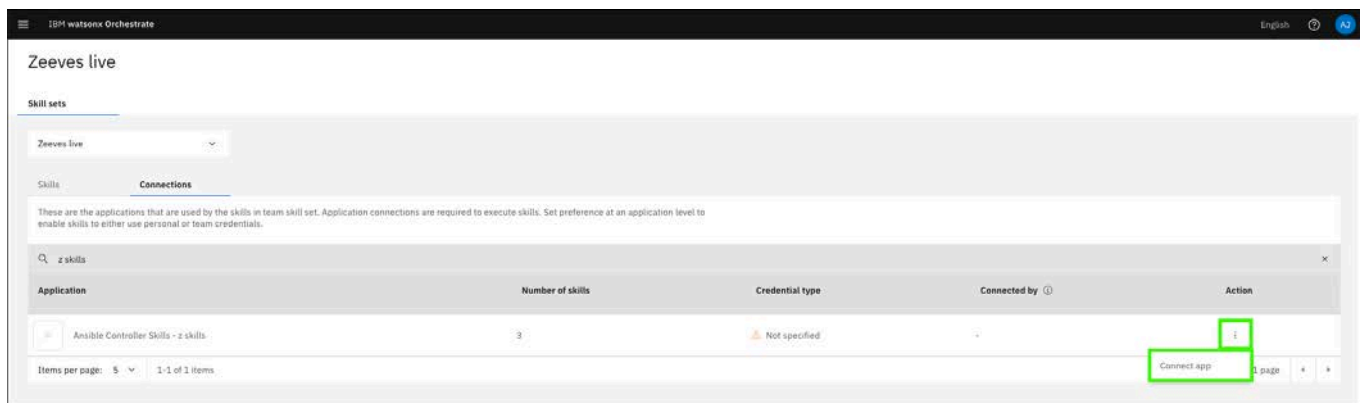
3. Click **Connections**.



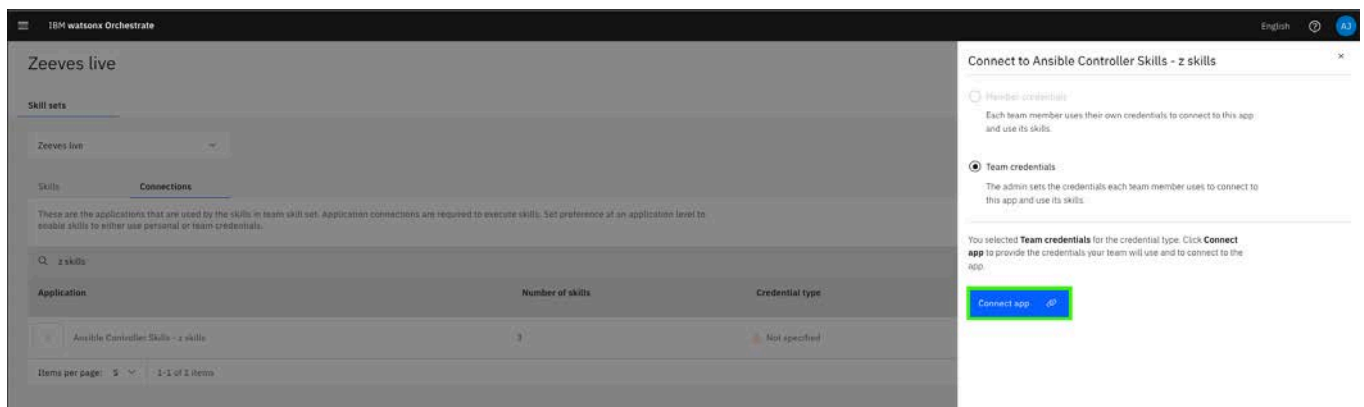
4. Search for the application name you specified earlier.



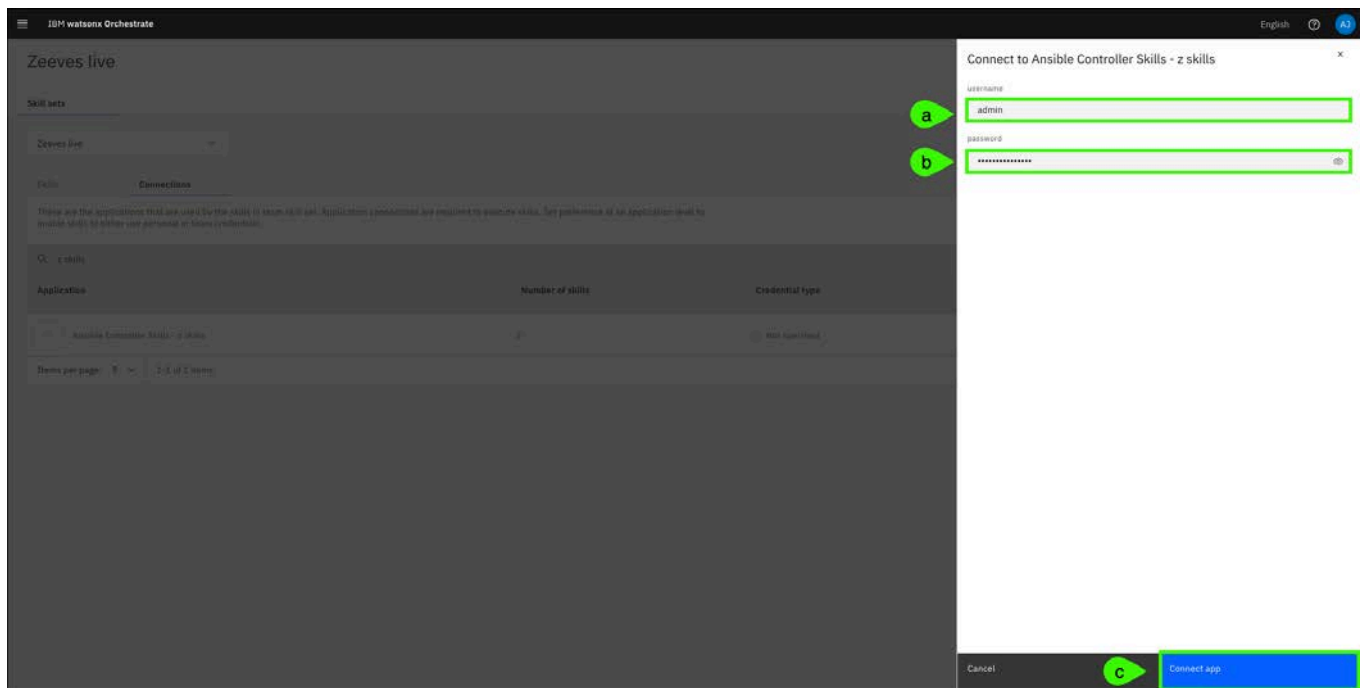
5. Click the ellipses () for your app and then click **Connect app**.



6. Click **Connect app**.



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



Learn more about publishing your assistant and creating live environments [here](#).

Deploy the assistant

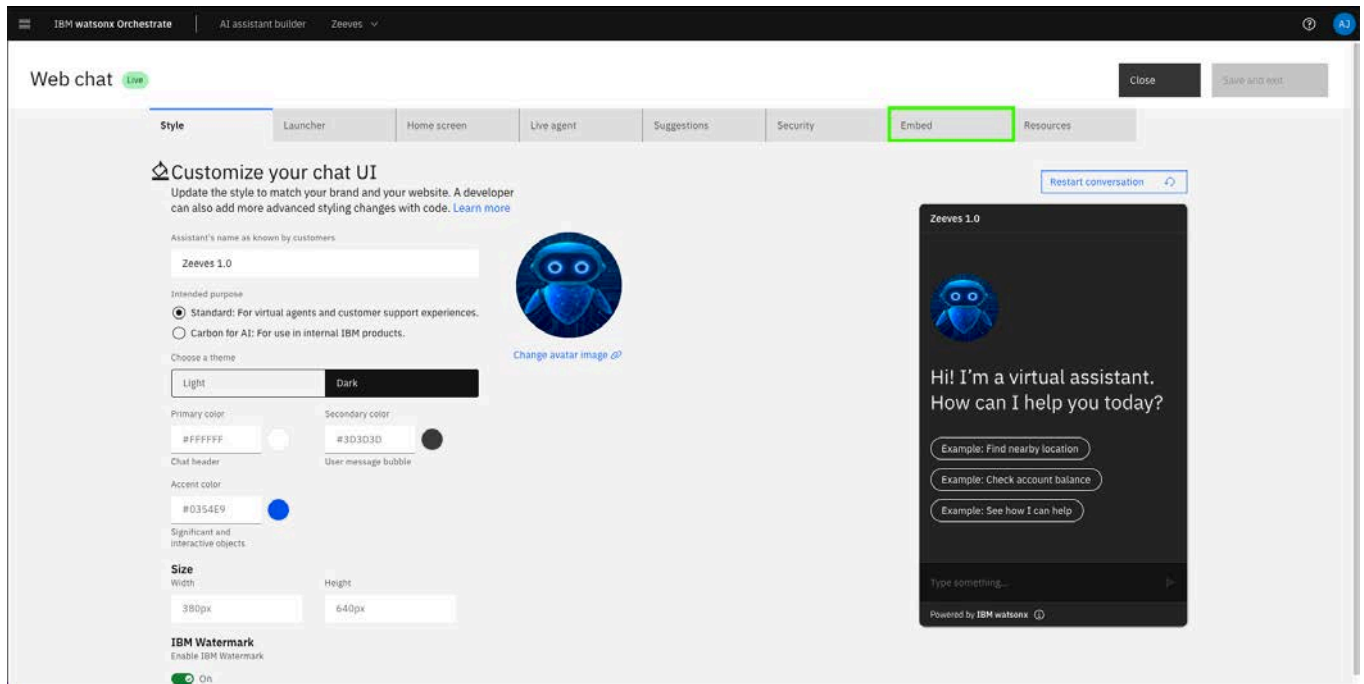
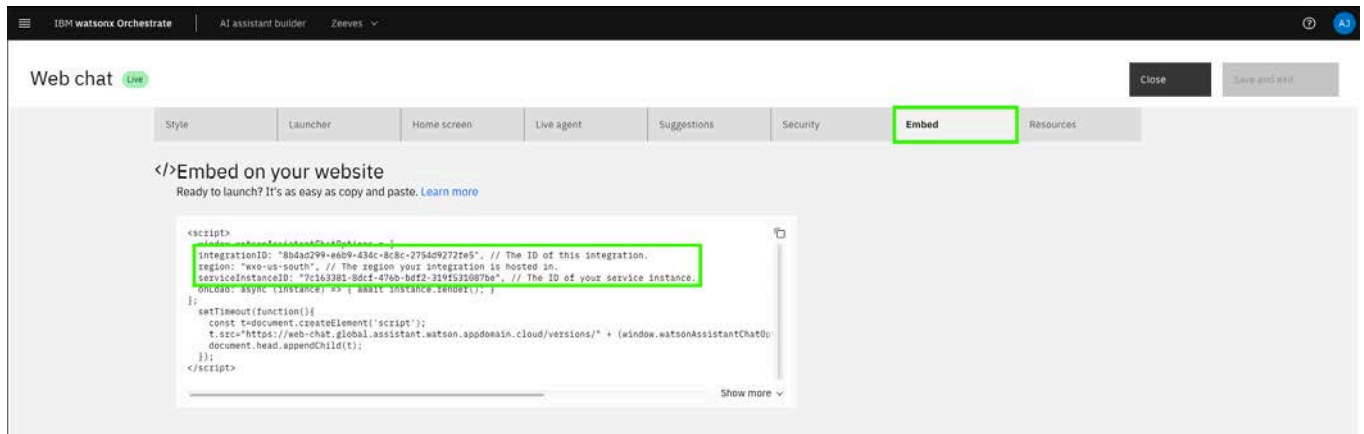
After configuring your assistant's settings and publishing, the final step is to deploy your assistant, which can be done across various channels depending on the use case.

Several options exist for deploying your assistant through channels and integrations to satisfy the use cases that you might encounter. Learn more about all the deployment options [here](#).

For this lab, deploy the assistant by using the web chat integration. The web chat integration provides an assistant interface that can integrate with a website. Learn more about the web chat integration [here](#).

1. Open the **Environments** page in the **AI assistant builder**.
2. Click **Web chat** for the **Live** environment.



3. Click the **Embed** tab.4. Copy and record the `integrationID`, `region`, and `serviceInstanceID` values.5. In a text editor, create a file that is named `Watson Assistant Chat.html` and paste the following text in the file.

File name:

Watson Assistant Chat.html

File contents:


```

<html lang="en">
<head>
<title>Watson Assistant Chat</title>
<meta name="viewport" content="width=device-width, initial-scale=1">

<style>
  .WebChatContainer {
    position: absolute;
    left: 0;
    right: 0;
    top: 0;
    bottom: 0;
  }
</style>
</head>
<body>

<div class="WebChatContainer"/>

<script>
const element = document.querySelector('.WebChatContainer');

window.watsonAssistantChatOptions = {
  integrationID: "<YOUR INTEGRATION ID>", // The ID of this integration.
  region: "<YOUR REGION>", // The region your integration is hosted in.
  serviceInstanceID: "<YOUR SERVICE INSTANCE ID>", // The ID of your service instance.
  element,

  openChatByDefault: true,
  hideCloseButton: true,

  layout: {
    showFrame: false,
    hasContentMaxWidth: true,
  },

  onLoad: async (instance) => {
    window.WACInstance = instance;
    await instance.render();
  }
};

setTimeout(function() {
  const t = document.createElement('script');
  t.src = 'https://web-chat.global.assistant.test.watson.appdomain.cloud/versions/' +
(window.watsonAssistantChatOptions.clientVersion || 'latest') +
'/WatsonAssistantChatEntry.js';
  document.head.appendChild(t);
});
</script>

</body>
</html>

```

Before modification:

```

Watson Assistant Chat.html
Users > andrewjones > Downloads > Watson Assistant Chat.html > html
1  <html lang="en">
16 <body>
20 <script>
21
22
23 window.watsonAssistantChatOptions = {
24   integrationID: "<YOUR INTEGRATION ID>", // The ID of this integration.
25   region: "<YOUR REGION>", // The region your integration is hosted in.
26   serviceInstanceID: "<YOUR SERVICE INSTANCE ID>", // The ID of your service instance.
27   element,
28
29   openChatByDefault: true,
30   hideCloseButton: true,
31
32   layout: {
33     showFrame: false,
34     hasContentMaxWidth: true,
35   },
36
37   onLoad: async (instance) => {
38     window.WACInstance = instance;
39     await instance.render();
40   }
41 };
42
43 setTimeout(function() {
44   const t = document.createElement('script');
45   t.src = 'https://web-chat.global.assistant.test.watson.appdomain.cloud/versions/' + (window.watsonAssistantChatOptions.clientVersion || 'latest') + '/WatsonAssistant
46   document.head.appendChild(t);
47 });
48 </script>
49
50 </body>
51 </html>

```

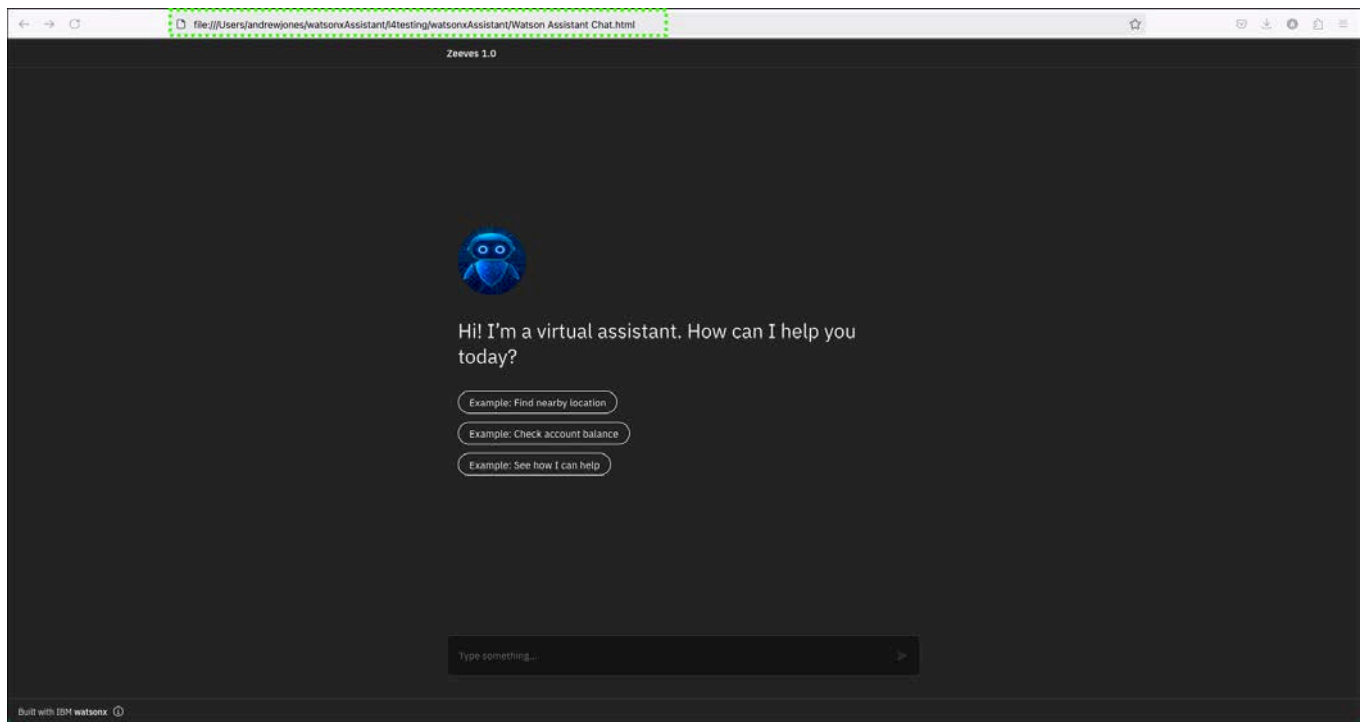
After modification:

```

Users > andrewjones > Downloads > Watson Assistant Chat.html > html > body > script > layout
1  <html lang="en">
2  <head>
6    <style>
7      .WebChatContainer {
11        top: 0;
12        bottom: 0;
13      }
14    </style>
15  </head>
16  <body>
17
18  <div class="WebChatContainer"/>
19
20  <script>
21    const element = document.querySelector('.WebChatContainer');
22
23    window.watsonAssistantChatOptions = {
24      integrationID: "8b4ad299-e6b9-434c-8c8c-2754d9272fe5", // The ID of this integration.
25      region: "wso-us-south", // The region your integration is hosted in.
26      serviceInstanceID: "7c163381-8dcf-476b-bdf2-319f531087be", // The ID of your service instance.
27      element,
28
29      openChatByDefault: true,
30      hideCloseButton: true,
31
32      layout: {
33        showFrame: false,
34        hasContentMaxWidth: true,
35      },
36
37      onLoad: async (instance) => {
38        window.WACInstance = instance;
39        await instance.render();
40      }
41    };

```

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 parmliib datasets
- Unix System services options
- Display zOS subsystems
- List spool files
- Retrieve dataset content
- Retrieve spool file content
- Retrieve z/OS Management facility (z/OSMF) job status

A great value of watsonx Orchestrate is the ability to build skills that anyone can use. You can build your own custom skills by importing an OpenAPI file into watsonx Orchestrate as a JSON or YAML file. For more information on building skills by importing OpenAPI files, refer to the documentation [here](#).

Watsonx Orchestrate also makes it possible to build, edit, and generate OpenAPI specifications by using the OpenAPI builder. With the OpenAPI Builder, you can use the AI function to simplify the process of generating these specifications. For more information on using the OpenAPI Builder, refer to the documentation [here](#).

For this use case, you are importing the skill for retrieving a z/OS server's IPL information. The next steps walk you through the process of downloading that OpenAPI JSON file and customizing it for your environment.

Download and customize the watsonx Assistant for Z OpenAP JSON file

1. Download the watsonx Assistant for Z OpenAP JSON file.

[wxa4z-skillpak-prepackaged-skills.json.zip](#)

2. Extract the file.

3. In a text editor, open the `wxa4z-skillpak-prepackaged-skills.json`, modify the server's url field as described, and save the file.

File name:

```
wxa4z-skillpak-prepackaged-skills.json
```

Substitute your Wazi as a Service (WaaS) instance URL for the string `<your z/OSMF URL>`. Your WaaS URL is based on your ITZ AAP URL that can be obtained from your watsonx Assistant for Z Pilot - AAP & z/OS ITZ reservation.

The AAP URL is similar to:

```
https://itzvsi-aap-ppxbcno.techzone.ibm.com
```

Change the `aap` string to `zos` and append `:10443` to the URL value. **Record this value for later use!** Your new URL is similar to:

```
https://itzvsi-zos-ppxbcno.techzone.ibm.com:10443
```

Before:

```
{
  "openapi": "3.0.1",
  "servers": [
    {
      "url": "<your z/OSMF URL>"
    }
  ],
  "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": {
      "put": {
```

After:

```

{} wxa4z-skillpak-prepackaged-skills.json •
Users > andrewjones > watsonxAssistant > finaltest > watsonxAssistant > {} wxa4z-skillpak-prepackaged-skills.json > {} info
1  {
2    "openapi": "3.0.1",
3    "servers": [
4      {
5        "url": "https://itzvsi-zos-pwgabob.techzone.ibm.com:10443"
6      }
7    ],
8    "info": {
9      "description": "z/OS operations",
10     "version": "1.0.0",
11     "title": "IBM watsonX Assistant for Z skills pak",
12     "x-ibm-annotations": "1.0.0",
13     "x-ibm-application-name": "z/OS operations",
14     "x-ibm-application-icon": "",
15     "x-ibm-application-id": "zos-operations",
16     "x-ibm-skill-subtype": "public",
17     "x-ibm-skill-type": "imported",
18     "x-ibm-disable-default-server-url": "true"
19   },
20   "tags": [
21     {
22       "name": "zOS operations"
23     }
24   ],
25   "paths": {
26     "/zosmf/restconsoles/consoles/{consoleName}?authProgList": {

```



In watsonx Orchestrate, each app is associated with a single URL.

If you have not imported previous skills into an app called `z/OS operations`, the default values in the **info** section are fine. If you already have a 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 `IBMUUSER` 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.

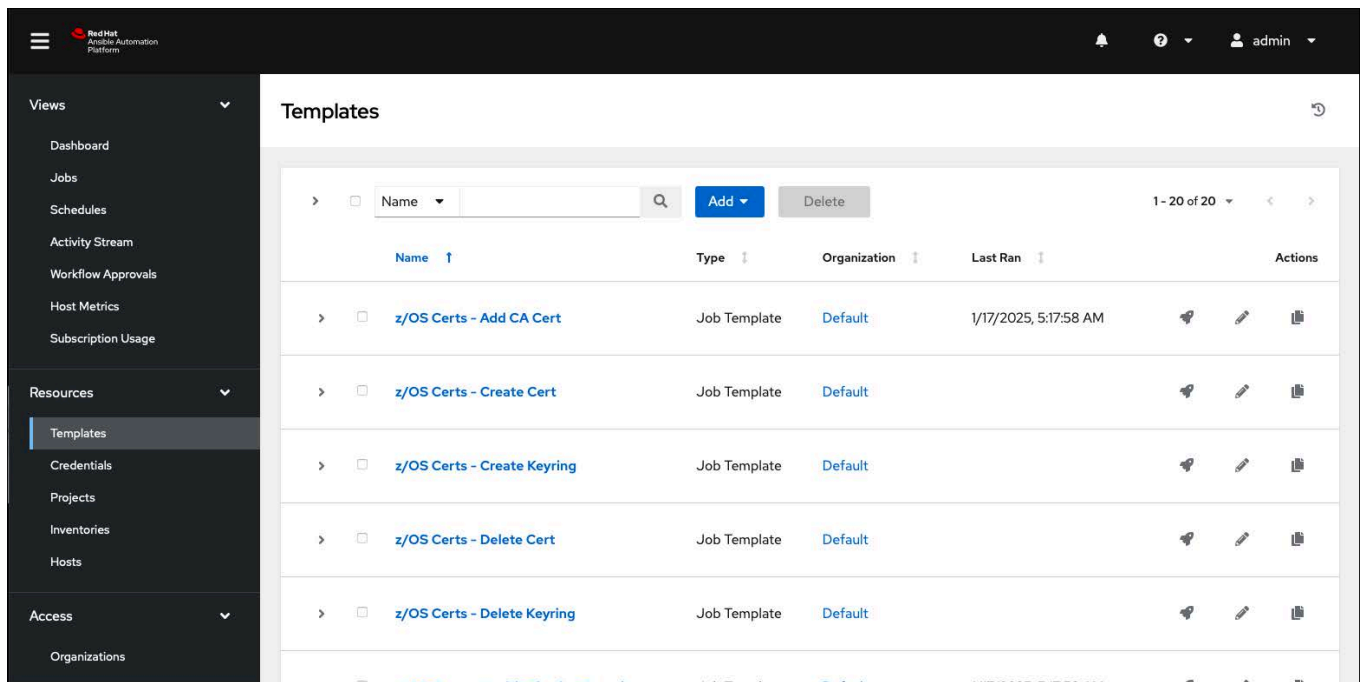
1. Open and log in to the Ansible Automation Platform (AAP) web console.



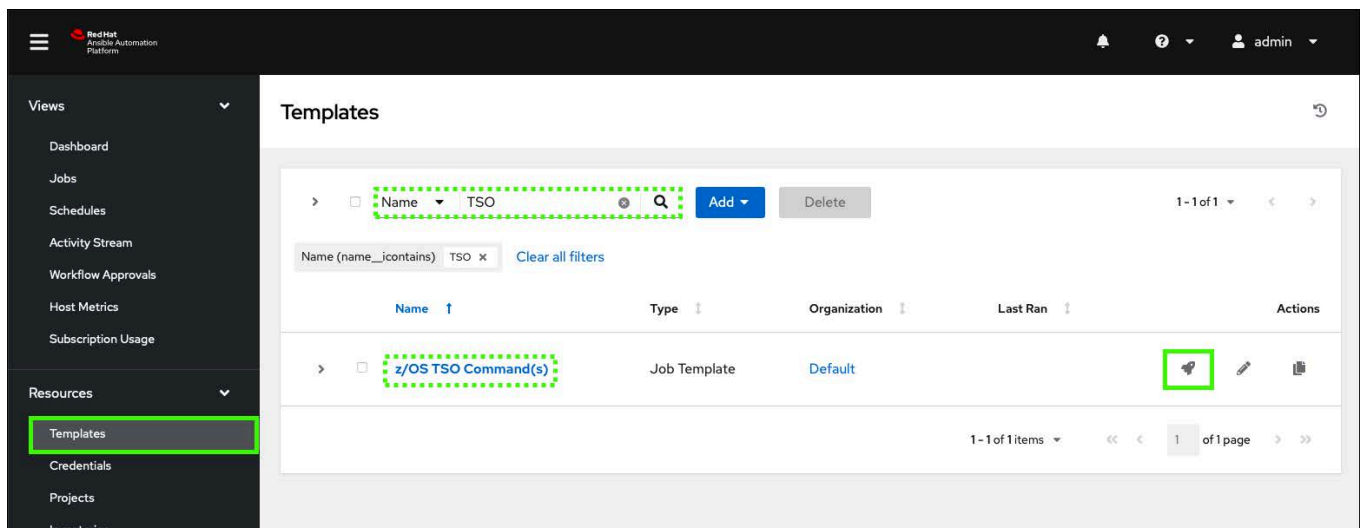
Don't remember how?

Refer to the first 5 steps in [Explore Ansible Automation Platform](#).

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



3. Click the launch icon (🔗) for the **z/OS TSO Command(s)** template.



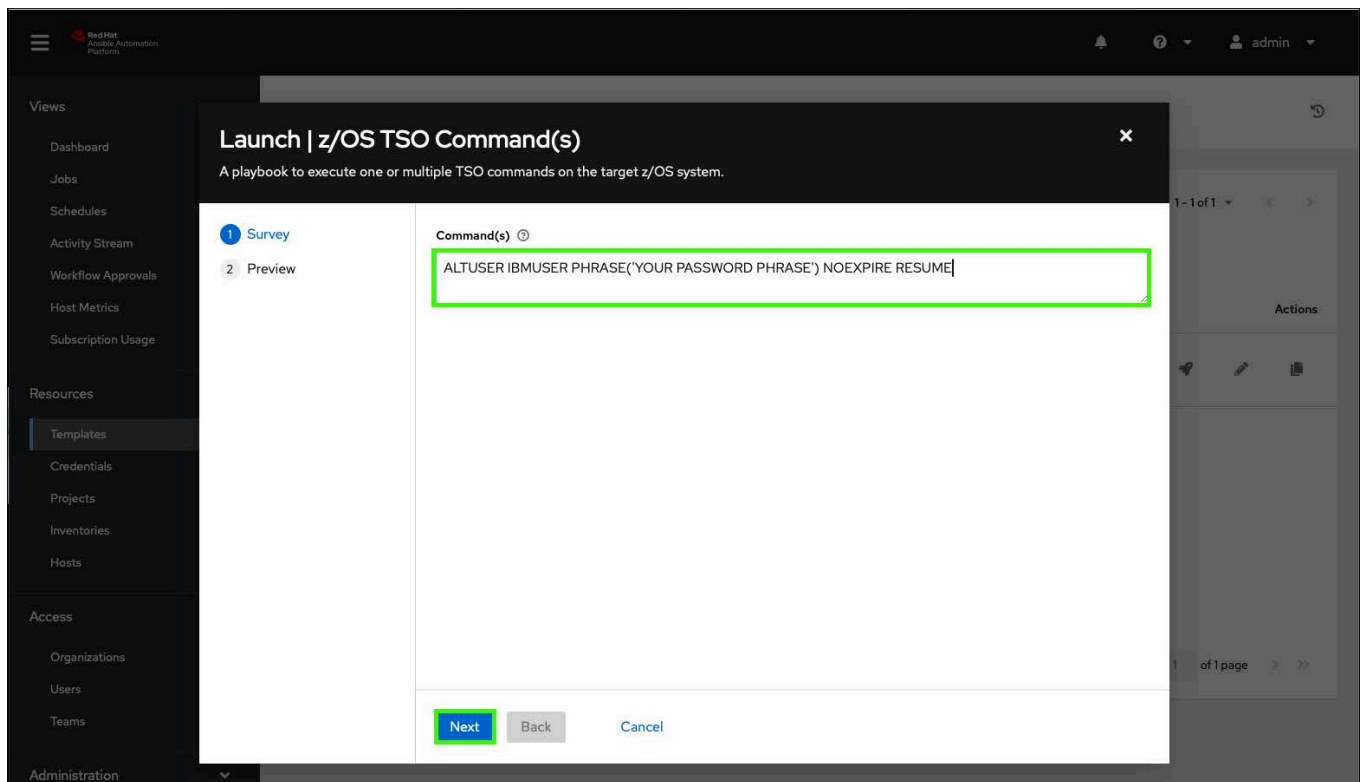
4. Replace the default command with the text that follows and substituting a password of your choosing for the string `YOUR PASSWORD PHRASE` and then click **Next**.

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

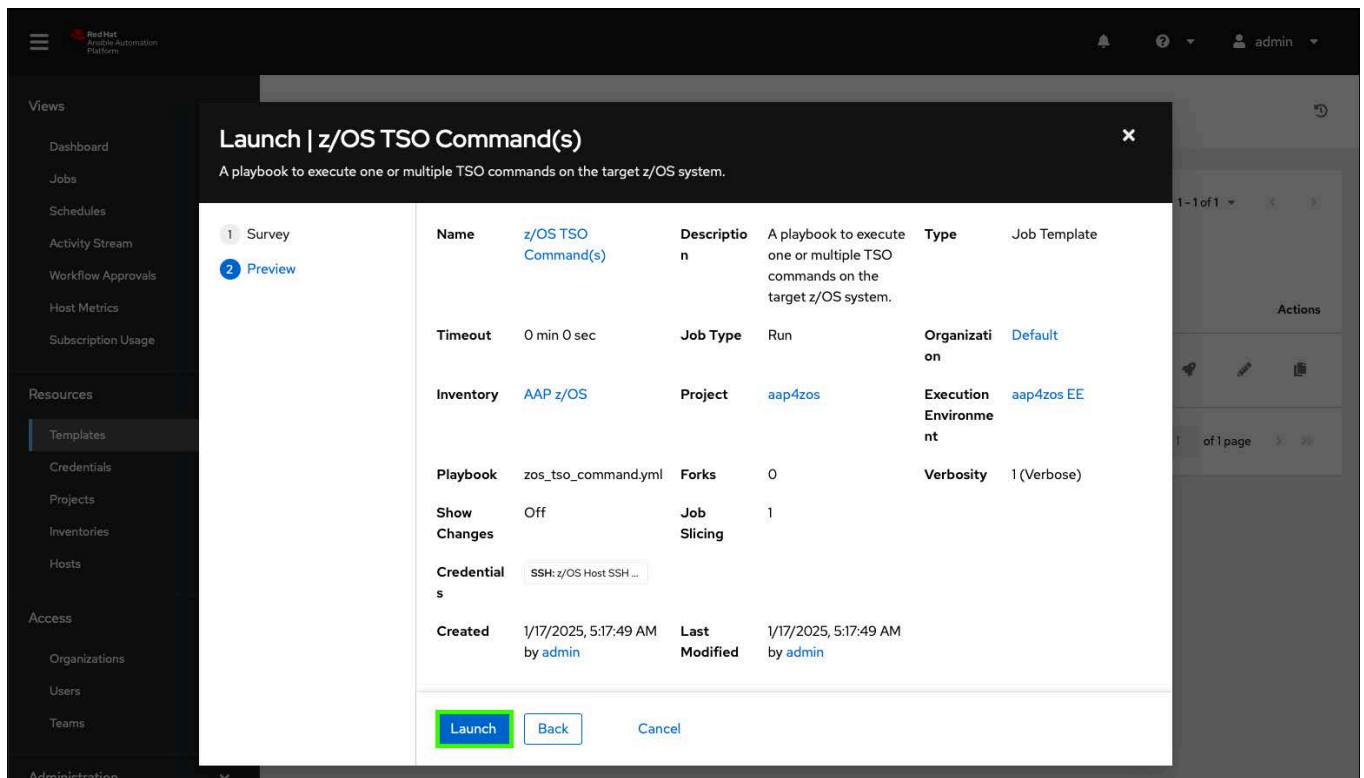


Avoid typographical errors later... keep the password simple.

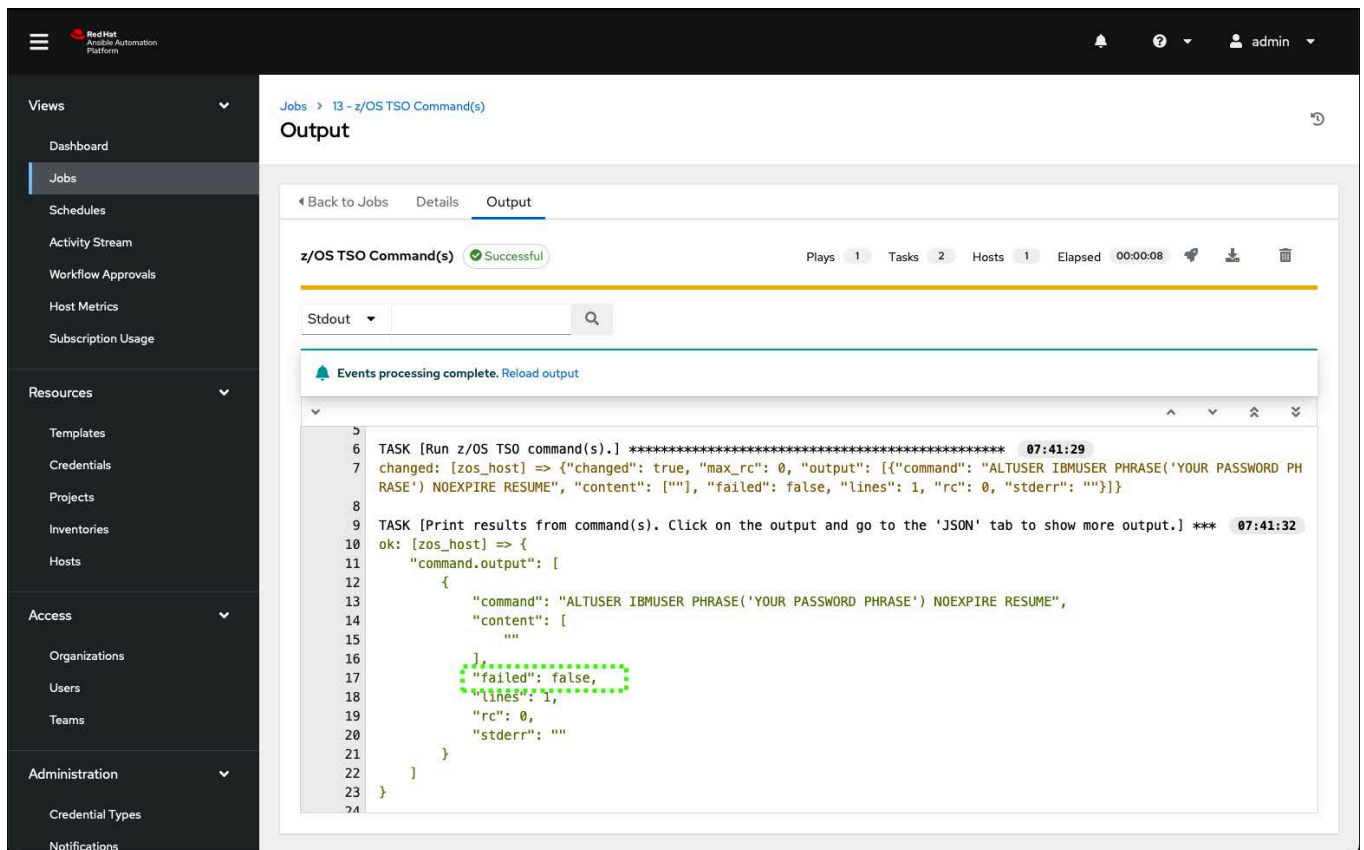
If you type the command yourself, be sure to include the single quotes before and after the password. **Record the password as it will be needed later.** We will refer to this as your **WaaS password**."



5. Click **Launch**.



6. Verify that the job is **Successful** by locating the message `"failed": false` in the job output.



The screenshot shows the Red Hat Ansible Automation Platform interface. On the left is a sidebar with navigation options: 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 panel displays the 'Output' of a job named 'z/OS TSO Command(s)'. The job status is 'Successful'. The output shows a task that changed the [zos_host] variable and then printed the results. The output is in JSON format, showing the command executed and the results.

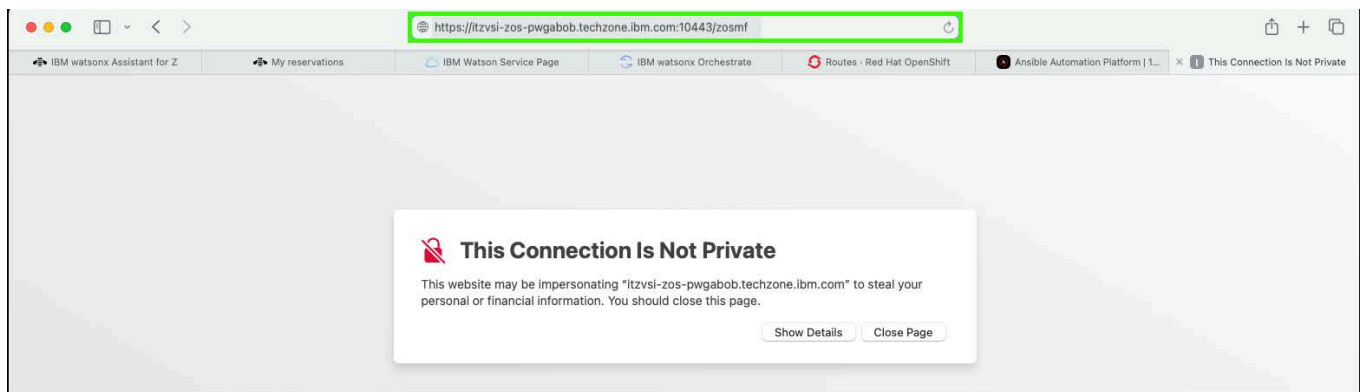
```

5 TASK [Run z/OS TSO command(s).] ***** 07:41:29
6 changed: [zos_host] => {"changed": true, "max_rc": 0, "output": [{"command": "ALTUSER IBMUSER PHRASE('YOUR PASSWORD PH
7 RASE') NOEXPIRE RESUME", "content": [""], "failed": false, "lines": 1, "rc": 0, "stderr": ""}]}
8
9 TASK [Print results from command(s). Click on the output and go to the 'JSON' tab to show more output.] *** 07:41:32
10 ok: [zos_host] => {
11   "command.output": [
12     {
13       "command": "ALTUSER IBMUSER PHRASE('YOUR PASSWORD PHRASE') NOEXPIRE RESUME",
14       "content": [
15         ""
16       ],
17       "failed": false,
18       "lines": 1,
19       "rc": 0,
20       "stderr": ""
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`.

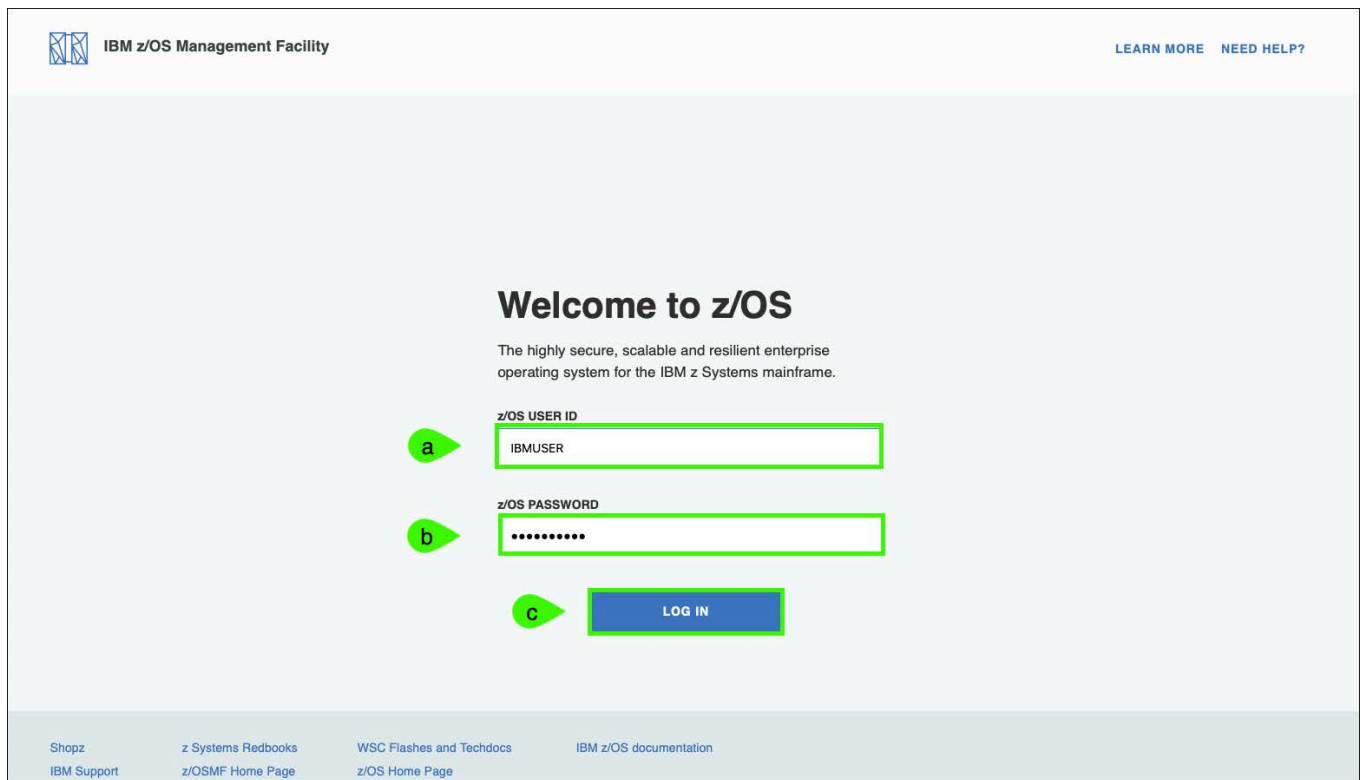


The screenshot shows a web browser with the URL `https://itzvsi-zos-pwgabob.techzone.ibm.com:10443/zosmf` in the address bar. A warning message is displayed in the center of the browser window, stating: "This Connection Is Not Private". Below the warning, it says: "This website may be impersonating 'itzvsi-zos-pwgabob.techzone.ibm.com'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



Accept any *connection not private* messages to open the page.

8. Enter (a) `IBMUSER` for the z/OS USER ID, (b) the password you specified in step 4 for the z/OS PASSWORD, and then (c) click **LOG IN**.



IBM z/OS Management Facility

[LEARN MORE](#) [NEED HELP?](#)

Welcome to z/OS

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

a z/OS USER ID
IBMUUSER

b z/OS PASSWORD

c LOG IN

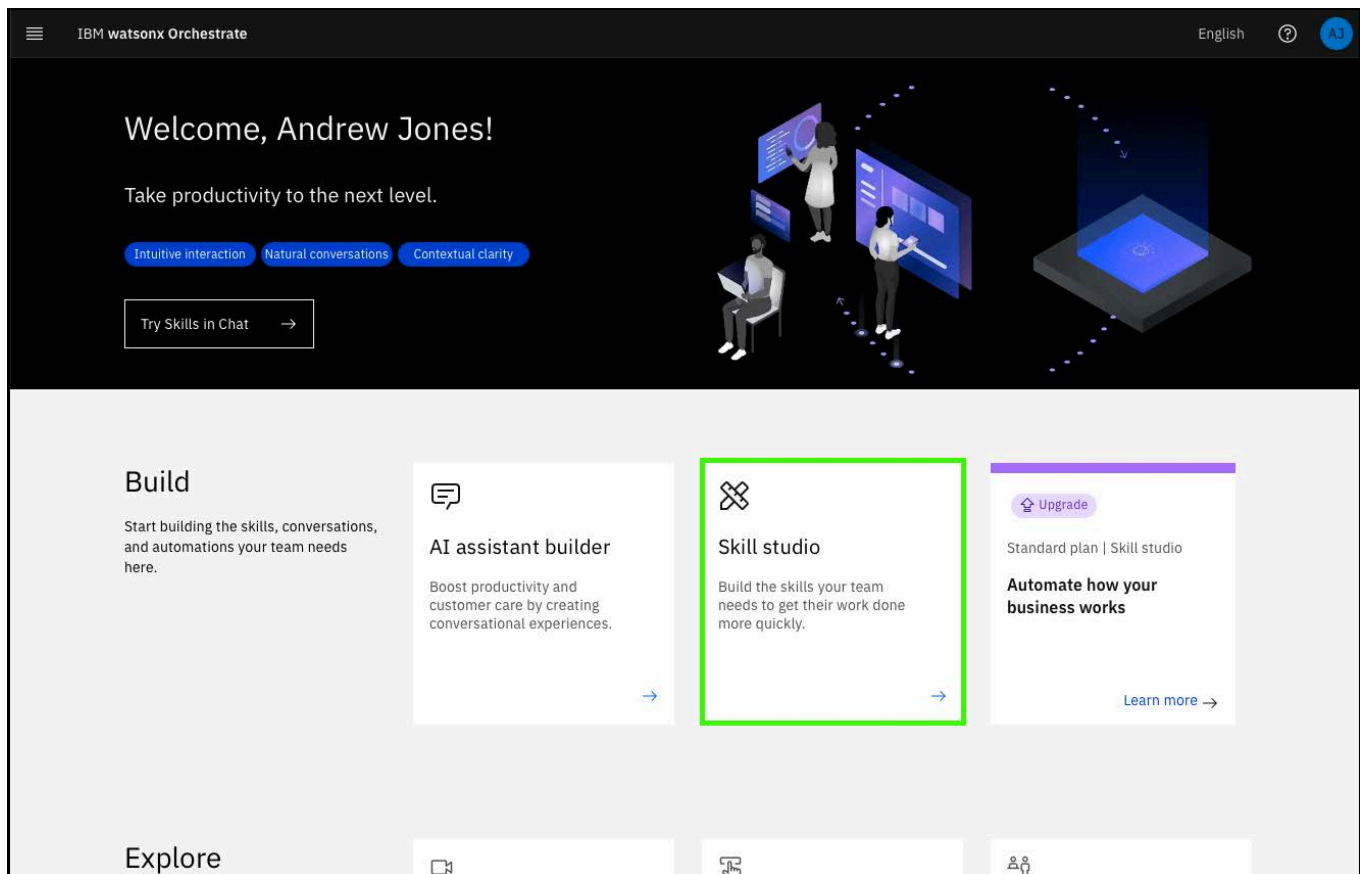
[Shopz](#) [z Systems Redbooks](#) [WSC Flashes and Techdocs](#) [IBM z/OS documentation](#)
[IBM Support](#) [z/OSMF Home Page](#) [z/OS Home Page](#)

9. Close the new browser tab after verifying a screen similar to the image that follows is displayed.

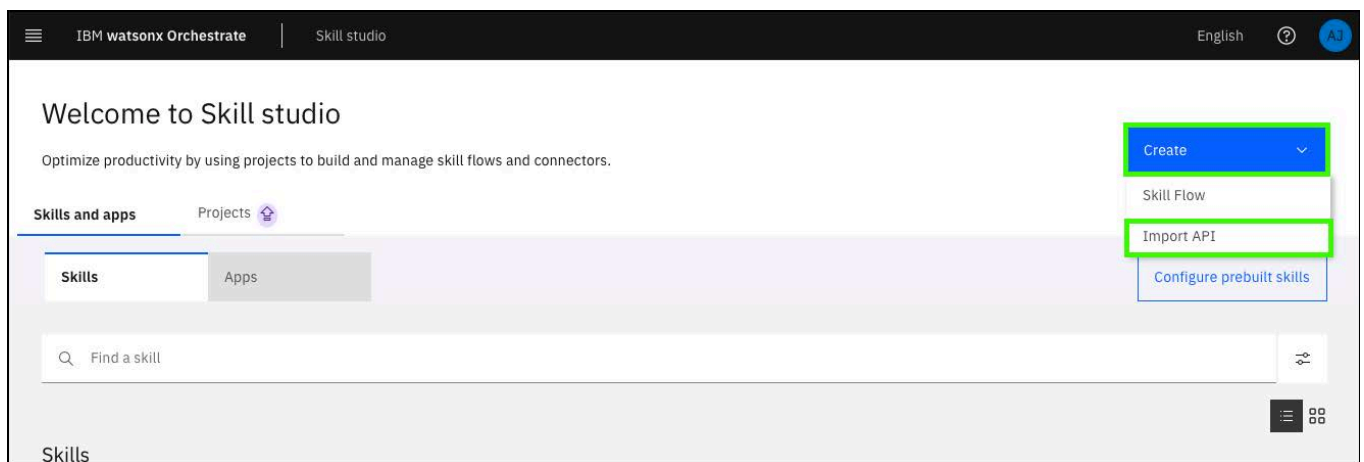


Import and publish the pre-packaged skills as an OpenAPI file

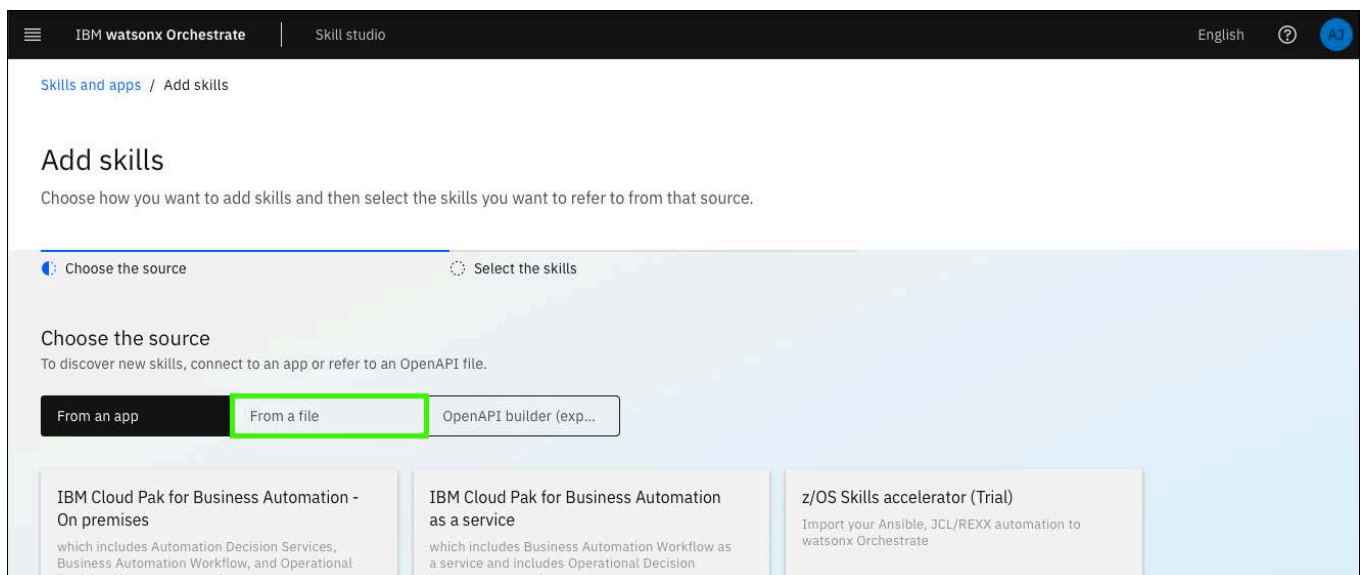
1. Open **Skills studio** in watsonx Orchestrate.



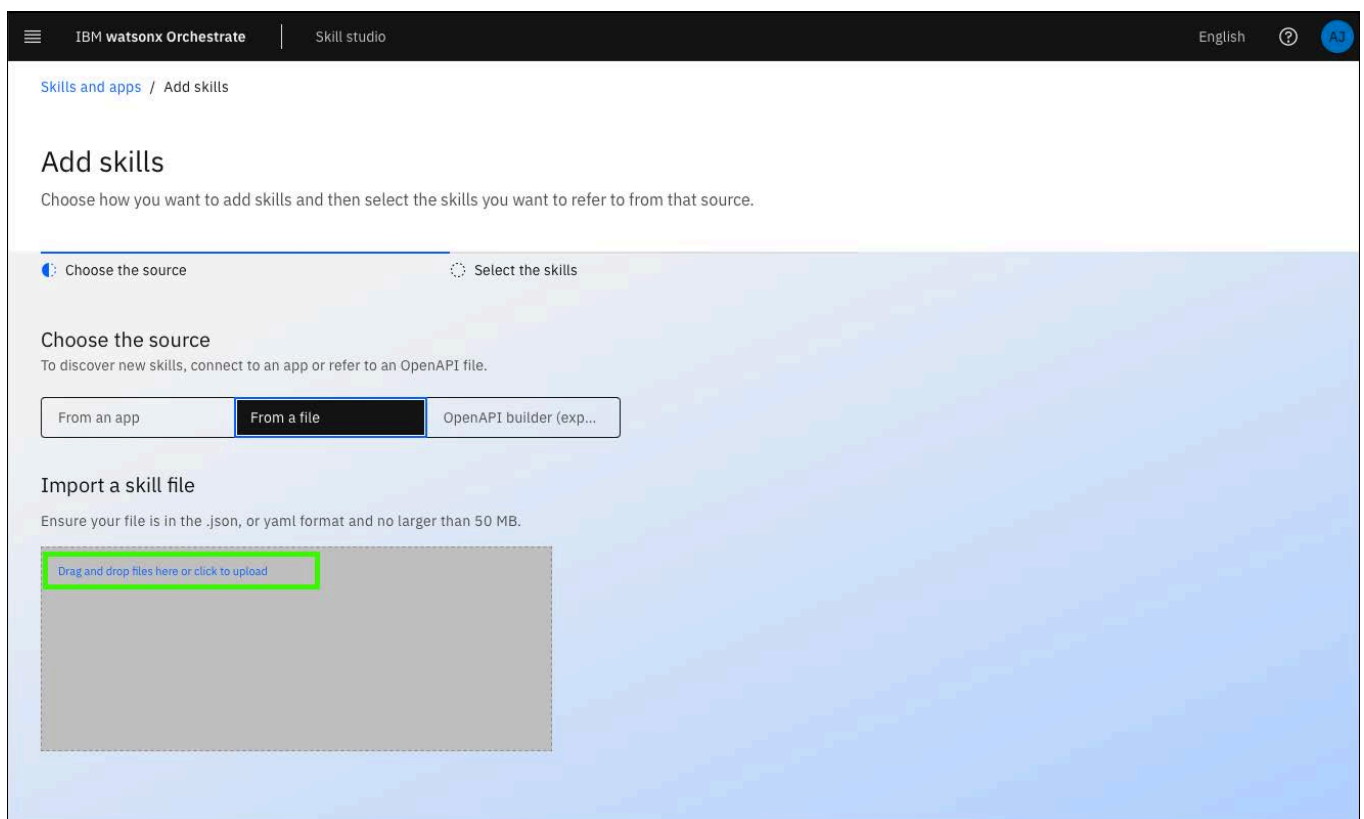
2. Click **Create** and then click **Import API**.



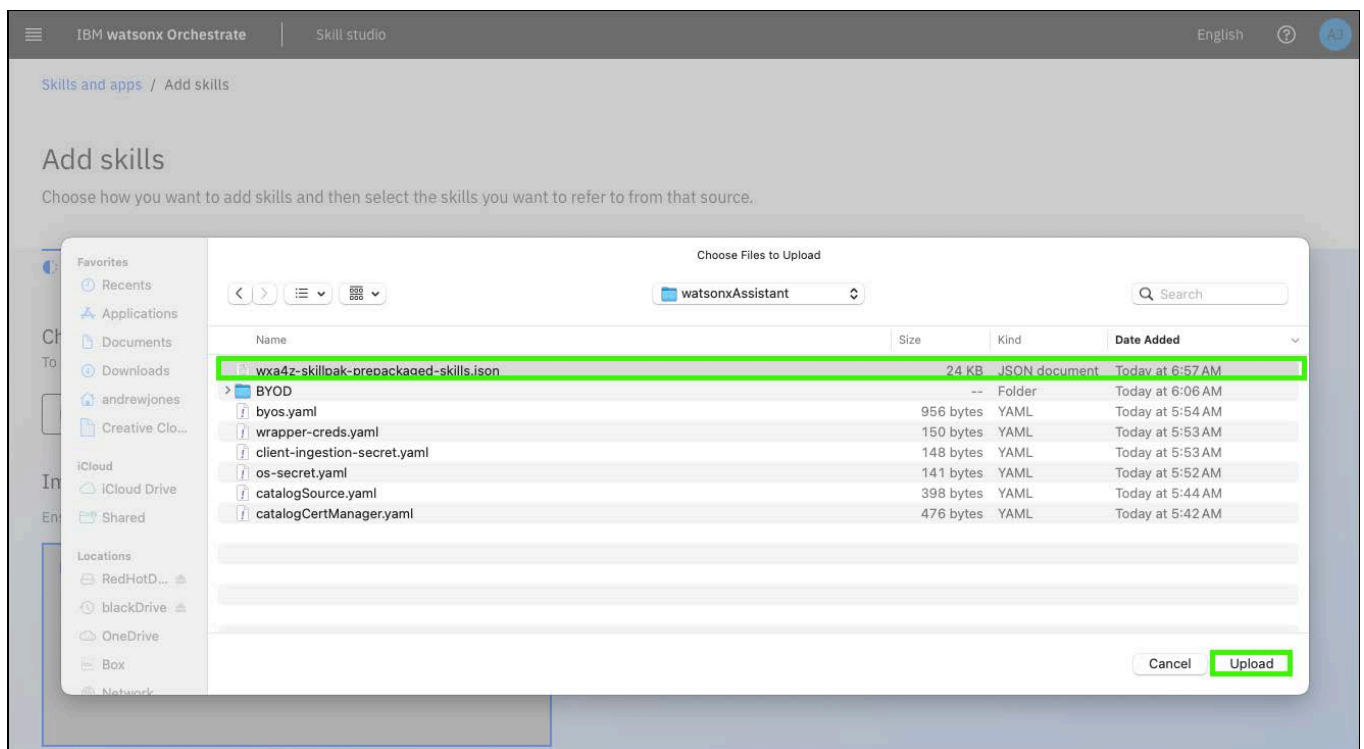
3. Click **From a file**.



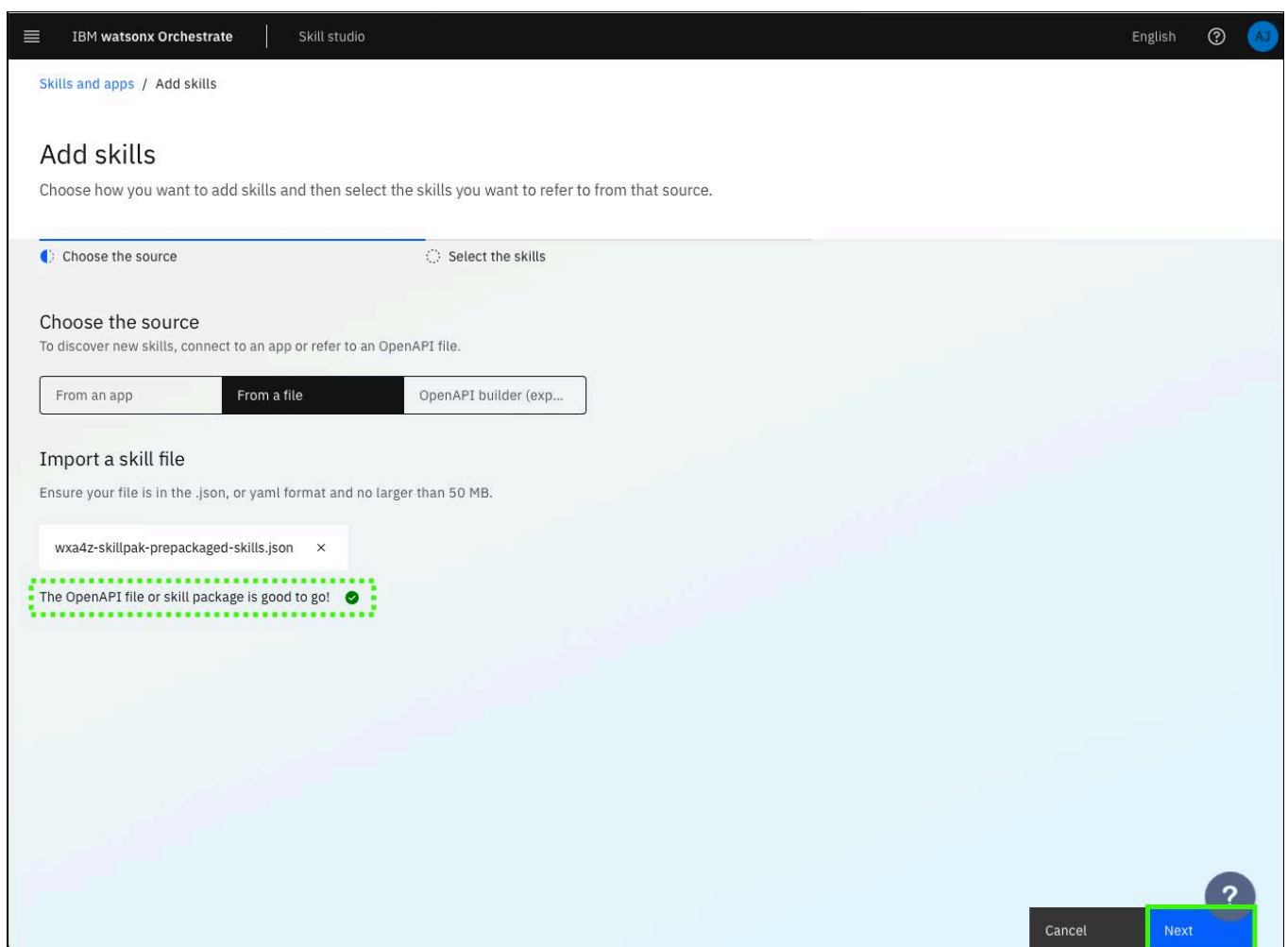
4. Click **Drag and drop files here or click to upload** in the **Import a skill file** window.



5. Locate and select the JSON file that you modified earlier and then click **Upload**.



6. Verify that you receive the message **The Open file or skill package is good to go!** and then click **Next**.

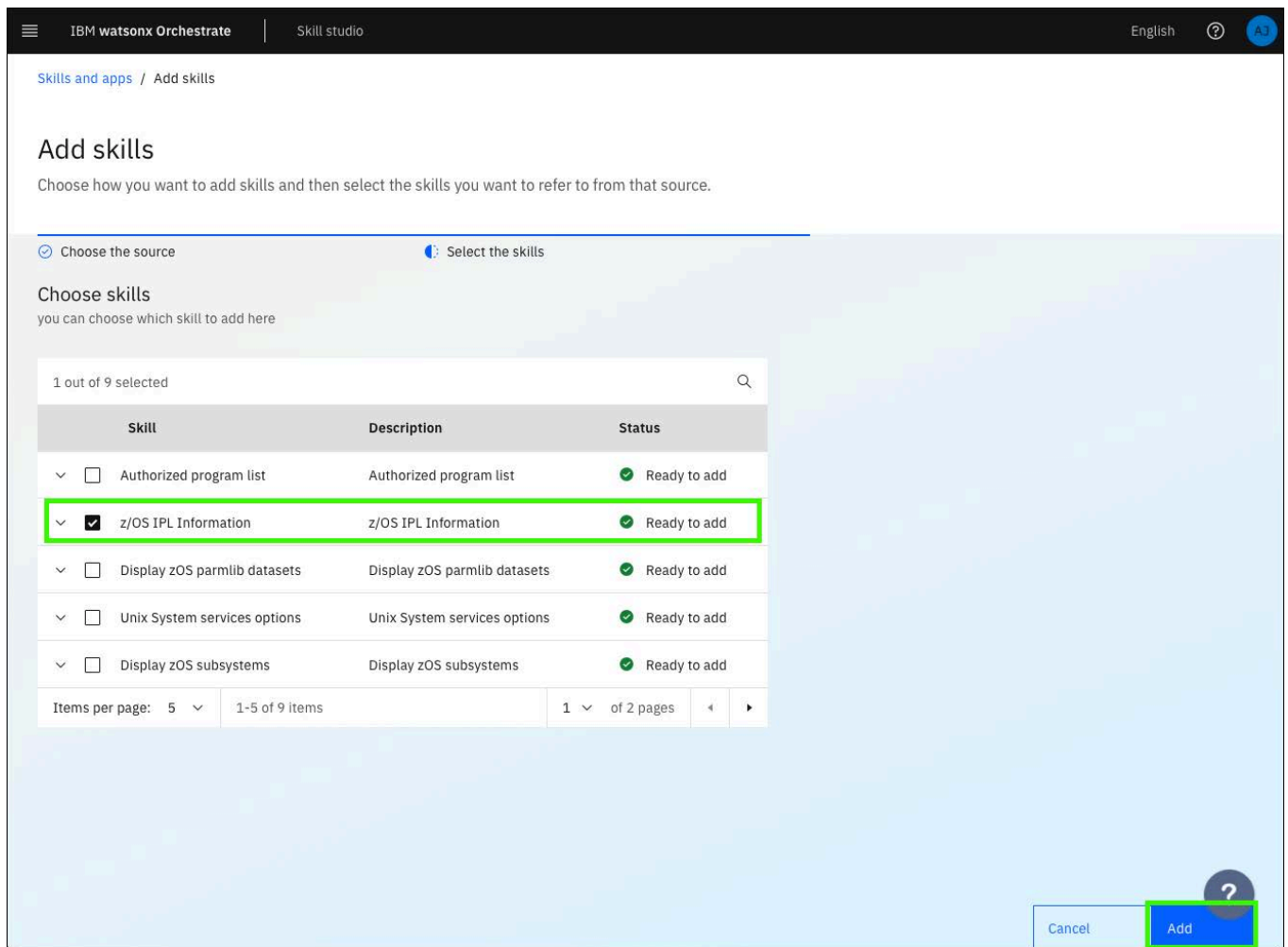


⊗ Not good to go?

If the file does not load properly you will need to verify not formatting or errors were made in your json file. Return to the previous section to verify the file contents and then reload the JSON file.

7. Select the **z/OS IPL Information** skill and then click **Add**.

Note: Only the **z/OS IPL Information** skill is required for this use case, but you can add as many skills as you like for testing purposes.



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 skills

you can choose which skill to add here

1 out of 9 selected

Skill	Description	Status
☐ Authorized program list	Authorized program list	✓ Ready to add
☒ z/OS IPL Information	z/OS IPL Information	✓ Ready to add
☐ Display zOS parmlib datasets	Display zOS parmlib datasets	✓ Ready to add
☐ Unix System services options	Unix System services options	✓ Ready to add
☐ Display zOS subsystems	Display zOS subsystems	✓ Ready to add

Items per page: 5 1-5 of 9 items 1 of 2 pages

Cancel Add

8. Click the ellipses icon (⋮) for the **z/OS IPL Information** skill and then click **Enhance this skill**.

IBM watsonx Orchestrate | Skill studio English ? A3

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

Configure prebuilt skills

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

9. Review the skill enhancements options and then click **Publish**.

IBM watsonx Orchestrate | Skill studio English ? A3

Skills and apps / Enhance this skill

Enhance the “z/OS IPL Information” skill

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

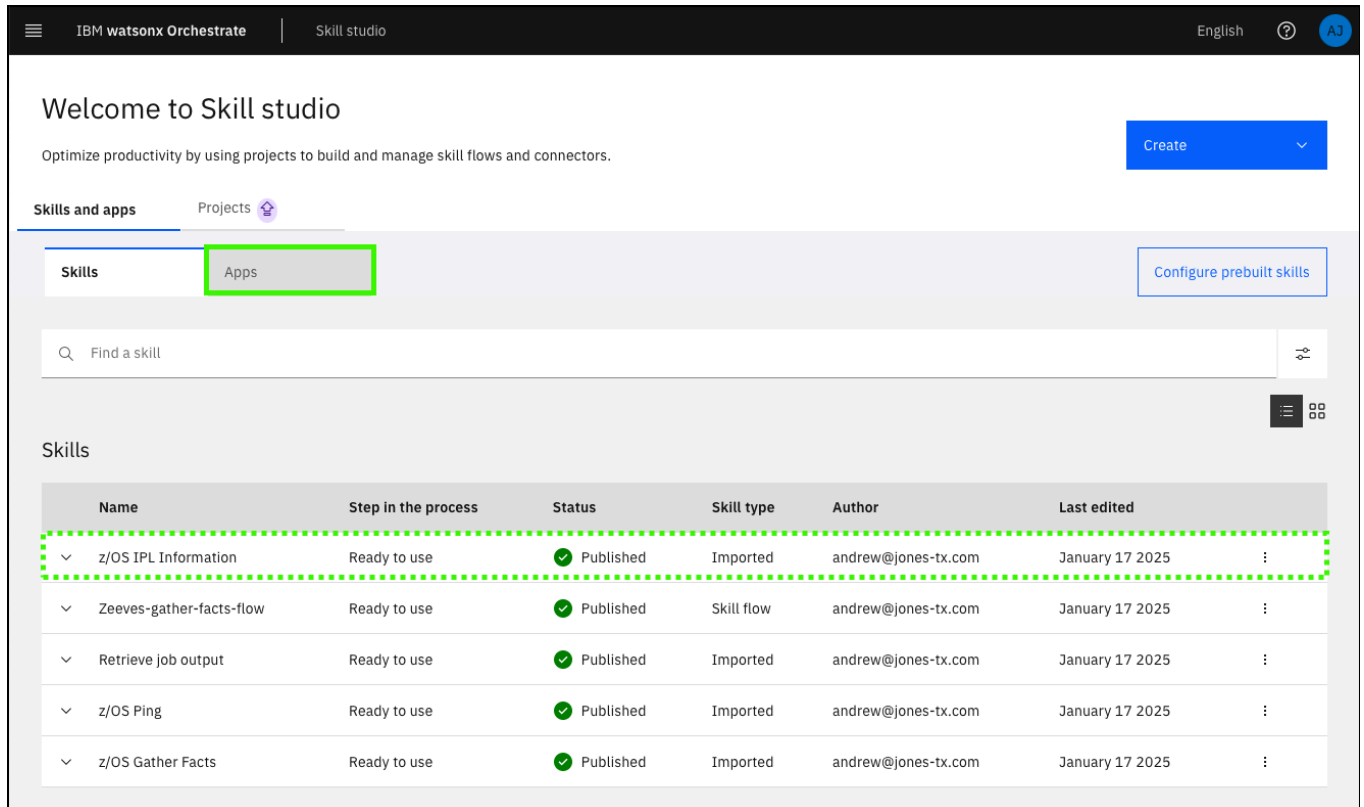
Name	Input	Output	Security	Phrases	Next best skills
Name and describe this skill in a way that tells users how it's used and why they would want to use it. Name* z/OS IPL Information Description 20/100 z/OS IPL Information API version* 1.0.0 Categories Add categories App z/OS operations					
Preview 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 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.



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

Find a skill

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.

Skills and apps | Projects

Skills | **Apps** | Configure prebuilt skills

Search: 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

3. 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.

Skills and apps | Projects

Skills | **Apps** | Configure prebuilt skills

Search: 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

Edit

4. Click the **Configuration** tab.

IBM watsonx Orchestrate | Skill studio

Skills and apps / Enhance this skill

Configure the “z/OS operations” app

Details | **Configuration**

Change the app's name and icon to whatever makes more sense for your organization. Be sure to provide the connection information.

Name*
z/OS operations

Description 0/100
z/OS IPL Information

App icon

Preview

The app will look like this in the skill set.

The app will look like this in the skill set.

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`.

IBM watsonx Orchestrate | Skill studio

Skills and apps / Enhance this skill

Configure the “z/OS operations” app

Details | **Configuration**

Server URL*
`https://itzvsi-zos-pwgabob.techz`

Authentication type*
Basic

Restore to default

Test connection

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

IBM watsonx Orchestrate | Skill studio

English ? AJ

Skills and apps / Enhance this skill

Configure the “z/OS operations” app

a

b

Details **Configuration**

Server URL* Authentication type*

https://itzvsi-zos-pwgabob.techz Basic Restore to default

Test connection

Connect to z/OS operations

username

IBMUUSER

password

YOUR PASSWORD PHRASE

Cancel **c** Connect app

7. Verify that the connection is successful and then click **Save**.

IBM watsonx Orchestrate | Skill studio English ? A3

Skills and apps / Enhance this skill

Configure the “z/OS operations” app

Details

Configuration

Server URL* Authentication type*

Basic Restore to default ↺

Connection successful. Valid URL. 08:58:14

Cancel

8. Repeat step 2 above to verify that the **Configuration status** is **Configured**.

IBM watsonx Orchestrate | Skill studio English ? A3

Welcome to Skill studio

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

Skills and apps Projects

Skills **Apps**

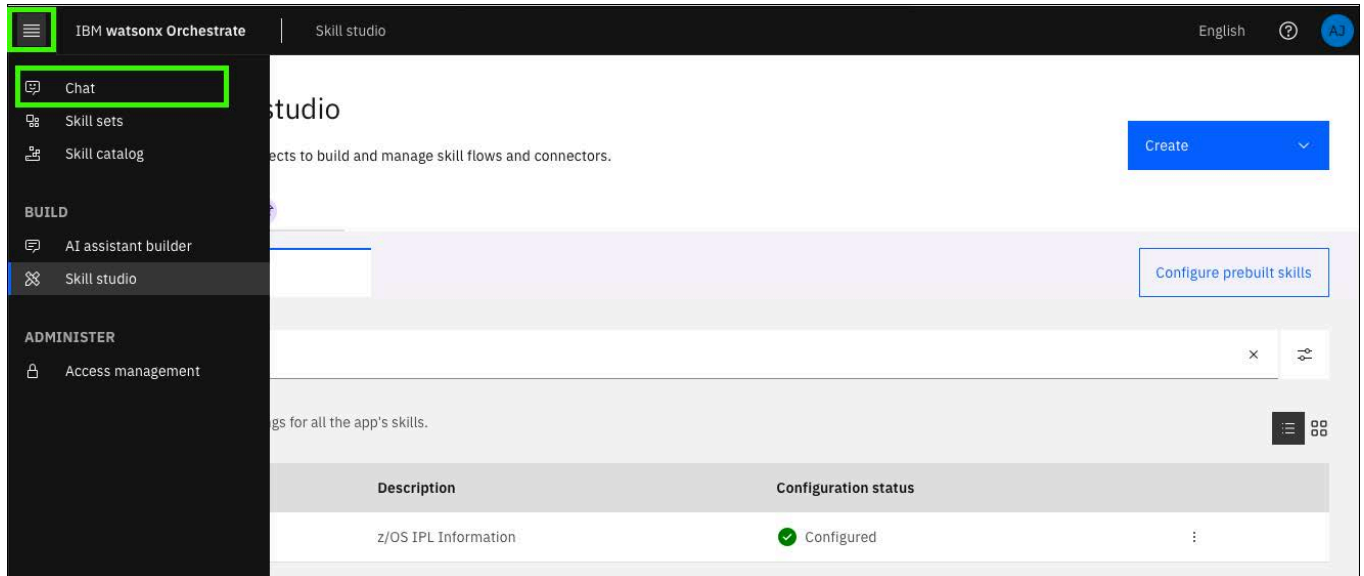
Q z/OS operations

Select an app to configure the settings for all the app's skills.

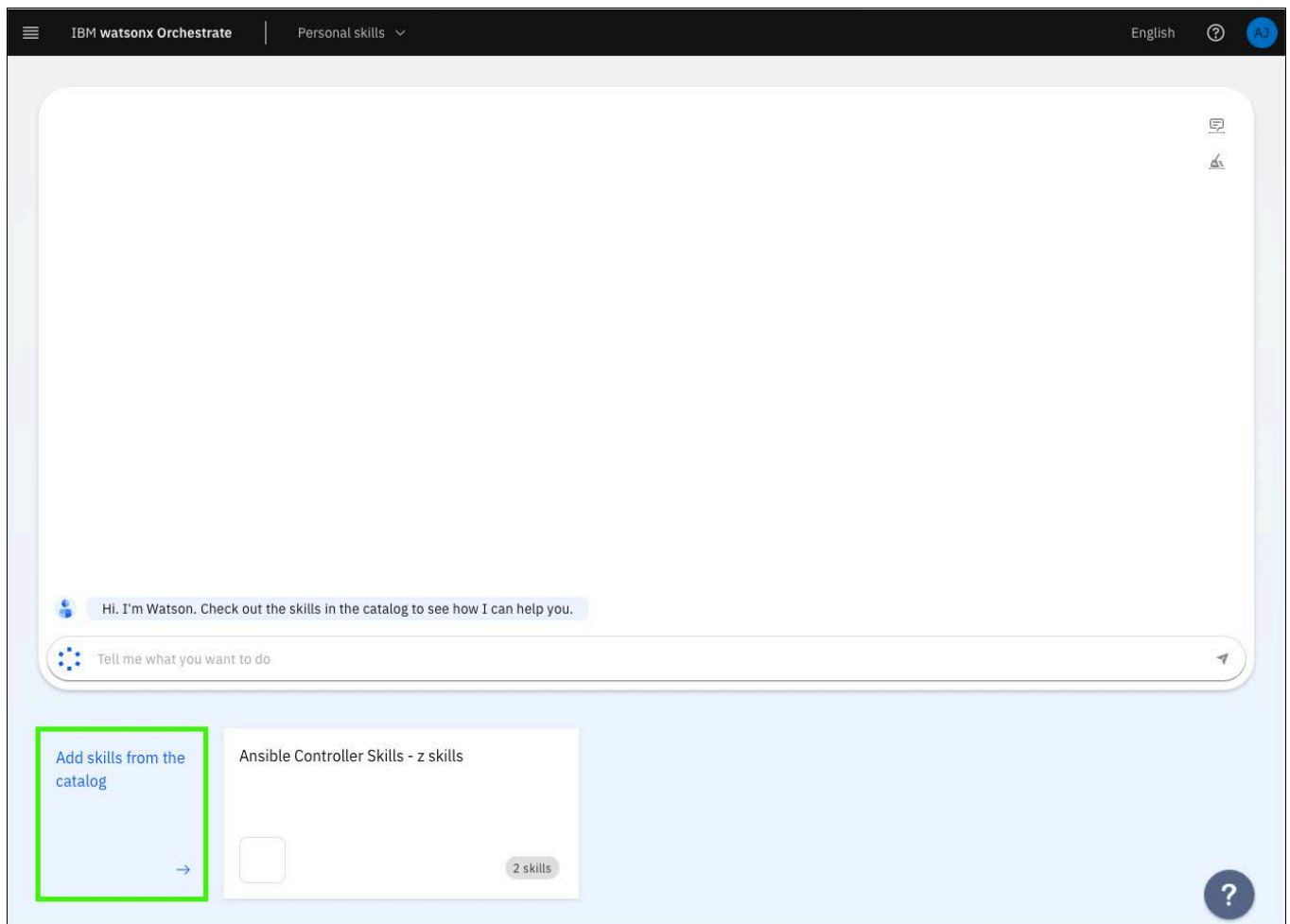
Name	Description	Configuration status
z/OS operations	z/OS IPL Information	☒ Configured

Test the skill

1. Open **Chat** in watsonx Orchestrate.



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



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**.

IBM watsonx Orchestrate

English

Skill catalog

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

Personal skills

z/OS operations

Most popular skills

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

All Apps

Coupa 23 skills	ZoomInfo 26 skills	HubSpot CRM 33 skills	Apptio Targetprocess 24 skills
Salesforce 85 skills	Zendesk Service 25 skills	Calendly 6 skills	Square 8 skills

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

IBM watsonx Orchestrate

English

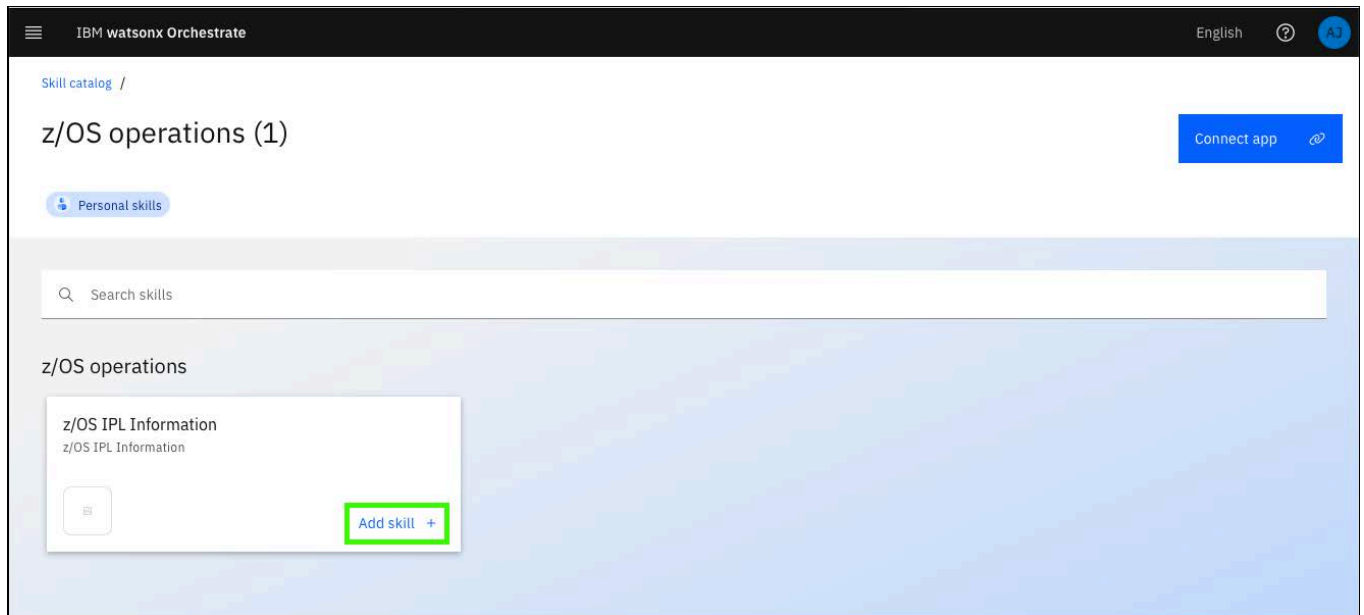
Skill catalog

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

Personal skills

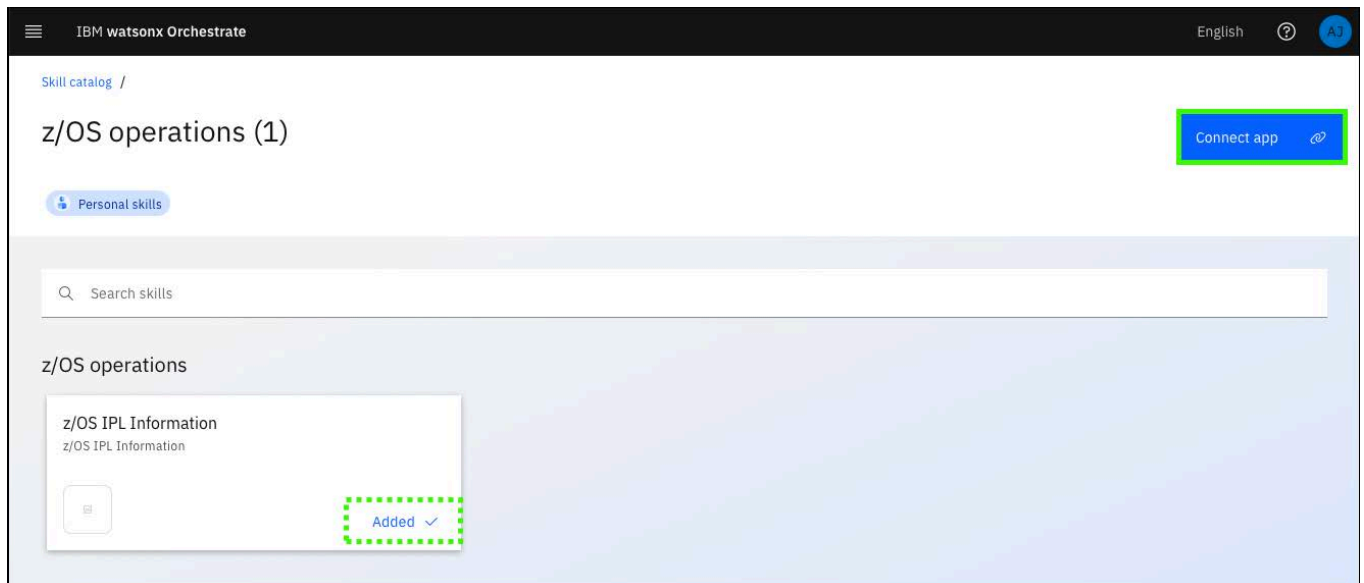
Microsoft Outlook 28 skills	Microsoft Dynamics 365 ... 27 skills	Slack 6 skills	Box 20 skills
Google Contacts 17 skills	Jira 4 skills	Microsoft OneDrive for B... 15 skills	Marketo 10 skills
monday.com 24 skills	Amazon SES 10 skills	Salesforce Commerce CL... 38 skills	DocuSign 3 skills
Google Calendar 16 skills	Seismic 22 skills	Microsoft Azure Blob sto... 12 skills	Google Cloud Storage 6 skills
UKG 8 skills	GitHub 11 skills	Google Drive 15 skills	Dropbox 9 skills
Amazon SNS 9 skills	Salesforce Chatter 5 skills	bambooHR 8 skills	AWS Lambda 2 skills
Webex 14 skills	Adobe Workfront 37 skills	z/OS operations 1 skill	Cognos 8 skills
Ansible Controller Skills ... 3 skills	Salesloft 47 skills	Gmail 1 skill	

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



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

7. Click **Connect app**.



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

IBM watsonx Orchestrator

English ? A3

Skill catalog /

z/OS operations (1)

Personal skills

Search skills

z/OS operations

z/OS IPL Information
z/OS IPL Information

Added ✓

Connect to z/OS operations

username

IBMUSER

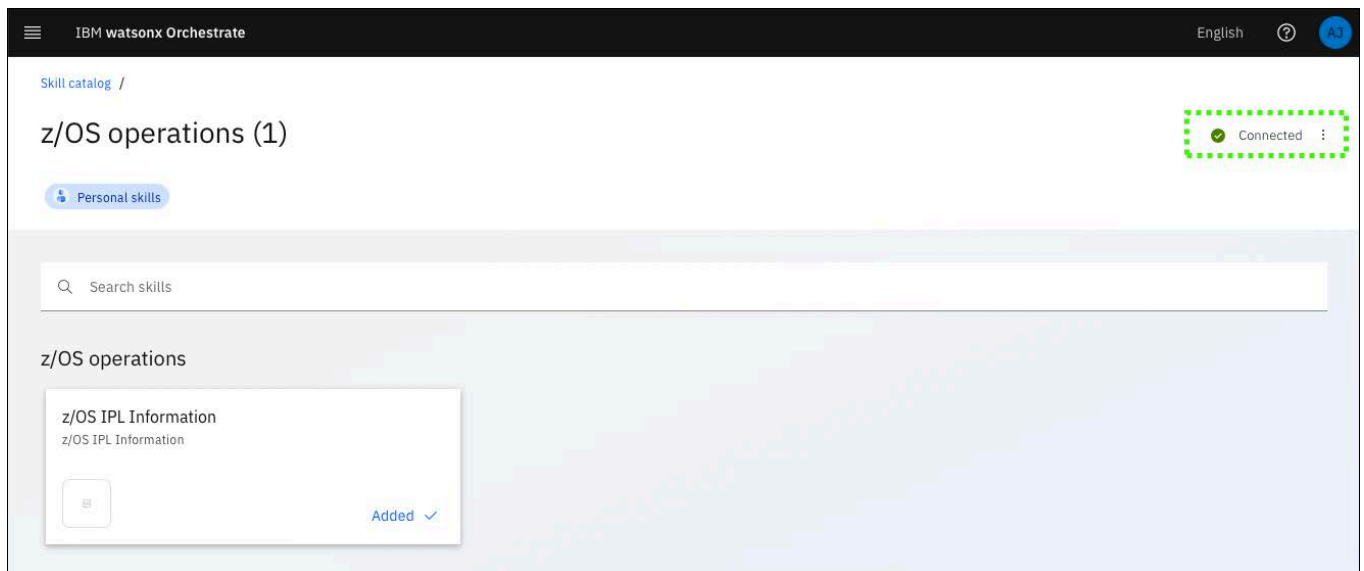
password

YOUR PASSWORD PHRASE

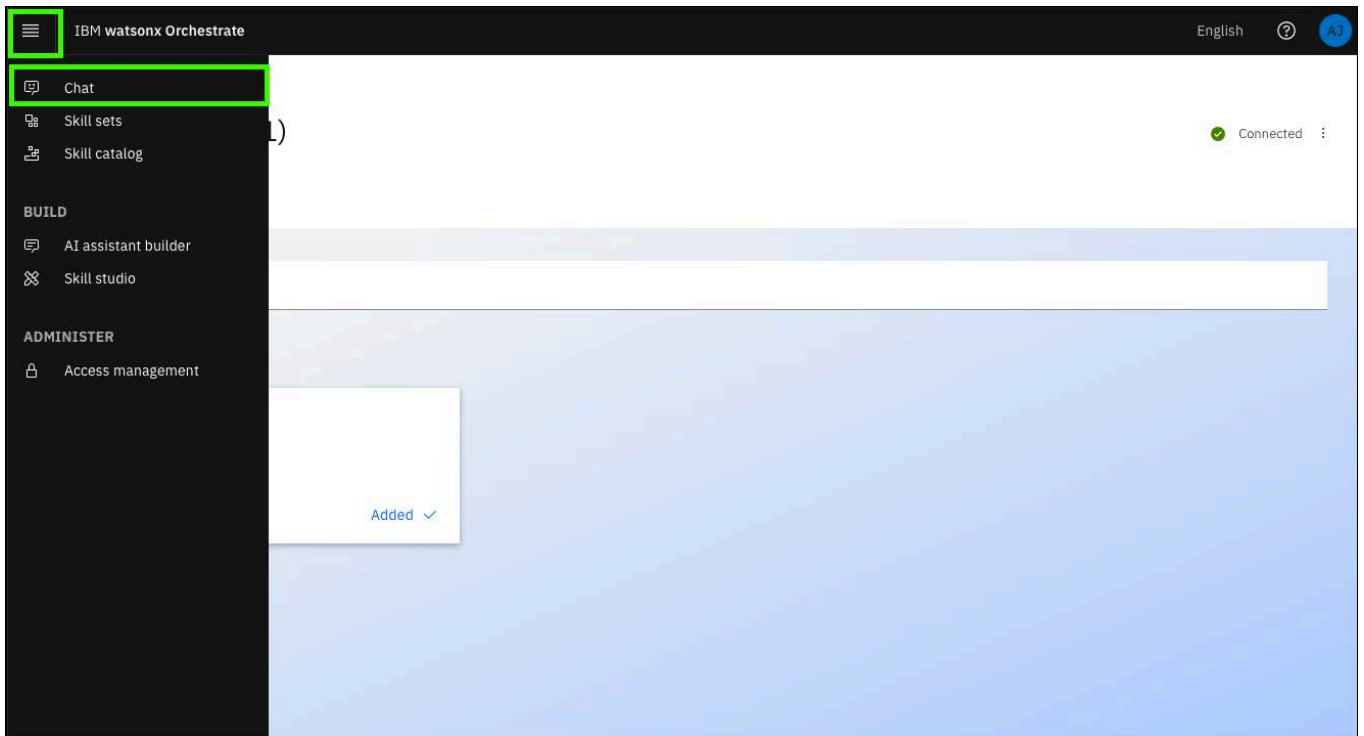
Cancel

Connect app

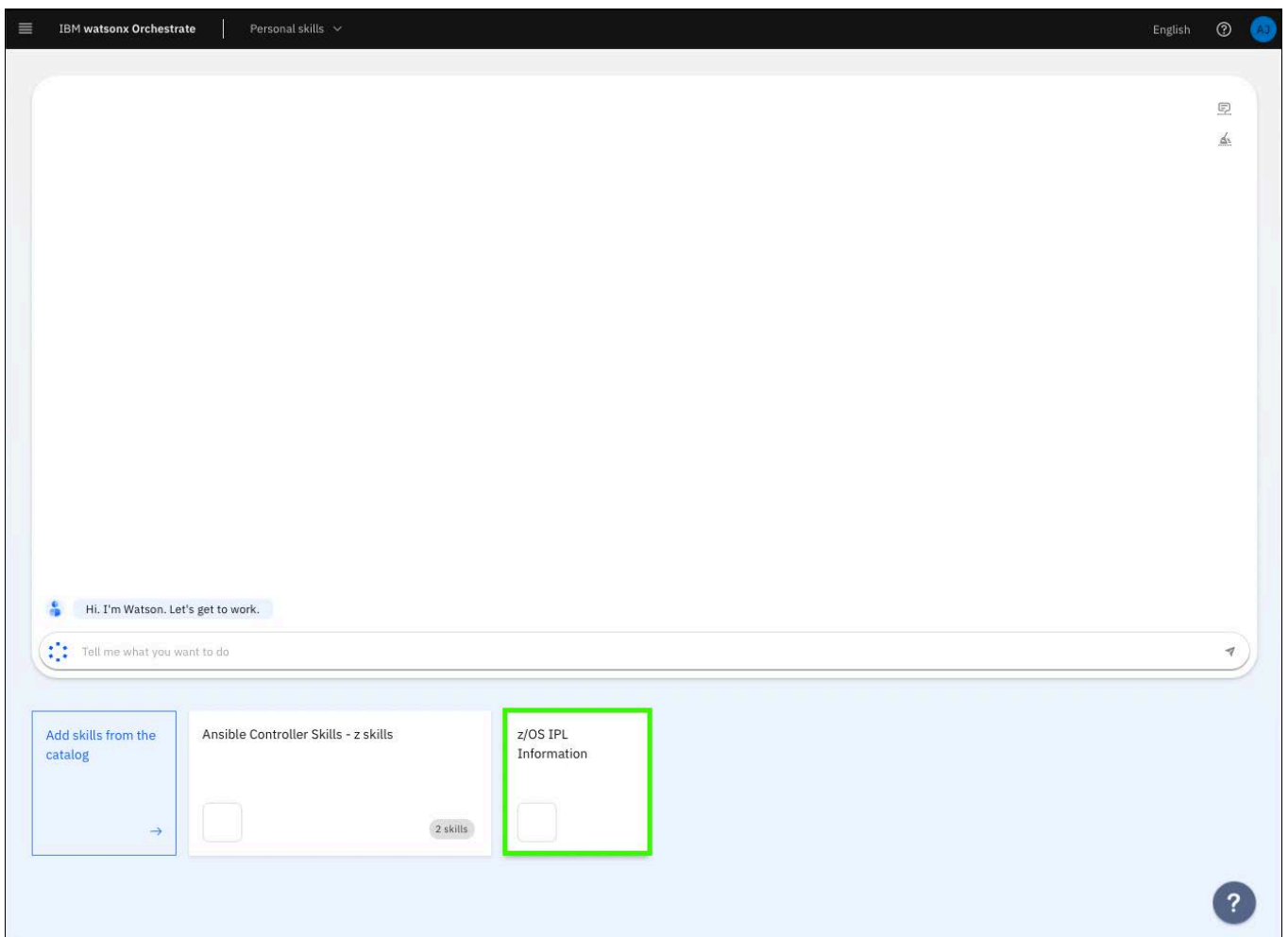
9. Verify that the skill is connected.



10. Open **Chat** in watsonx Orchestrator.



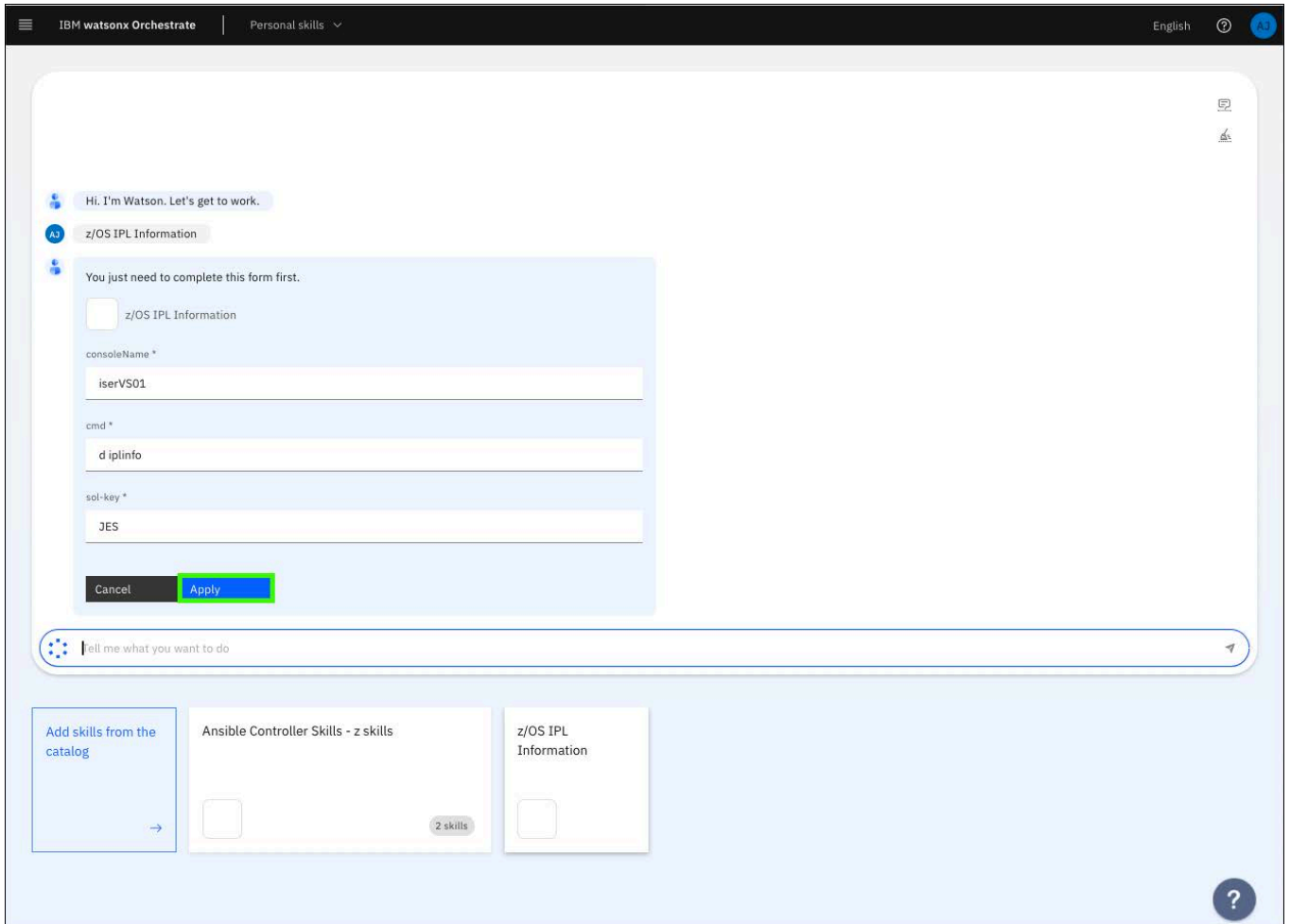
11. Click the **z/OS IPL Information** tile.



 **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, there's a header with "IBM watsonx Orchestrate" and "Personal skills". Below the header, a chat window displays a conversation: "Hi. I'm Watson. Let's get to work." followed by "z/OS IPL Information". A message from Watson says, "You just need to complete this form first." Below this is a form titled "z/OS IPL Information" with the following fields:

- consoleName ***: iserVS01
- cmd ***: d iplinfo
- sol-key ***: JES

At the bottom of the form are "Cancel" and "Apply" buttons. Below the form is a search bar with the placeholder text "Tell me what you want to do". At the bottom of the interface, there are three skill tiles: "Add skills from the catalog", "Ansible Controller Skills - z skills" (with a "2 skills" badge), and "z/OS IPL Information". A help icon (?) is located in the bottom right corner.

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.



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.

IBM watsonx Orchestrate

English ? A3

Team Skills

Skill sets

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

team's members. These skills complete related tasks for the team members so

ely.

Manage skills

Name	App
No skills	

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

3. Click **Connections**.

IBM watsonx Orchestrate

English ? A3

Zeeves draft

Skill sets

- Zeeves draft**

Skills **Connections**

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.

Manage skills

Name	App
Retrieve job output	
Zeeves-gather-facts-flow	

Items per page: 10 1-2 of 2 items 1 of 1 page

4. Enter your skill app name (z/OS operations) in the search bar.

IBM watsonx Orchestrate

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.

z/os operations

Application	Number of skills	Credential type	Connected by ⓘ	Action
z/os operations	1	⚠ Not specified	-	⋮

Items per page: 5 1-1 of 1 items

5. Click the ellipses icon (⋮) for the **z/OS operations** app and then click **Connect app**.

IBM watsonx Orchestrate

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.

z/os operations

Application	Number of skills	Credential type	Connected by ⓘ	Action
z/OS operations	1	⚠ Not specified	-	⋮

Items per page: 5 1-1 of 1 items

Connect app

6. In the dialog, click **Connect app**.

IBM watsonx Orchestrate

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.

z/os operations

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.

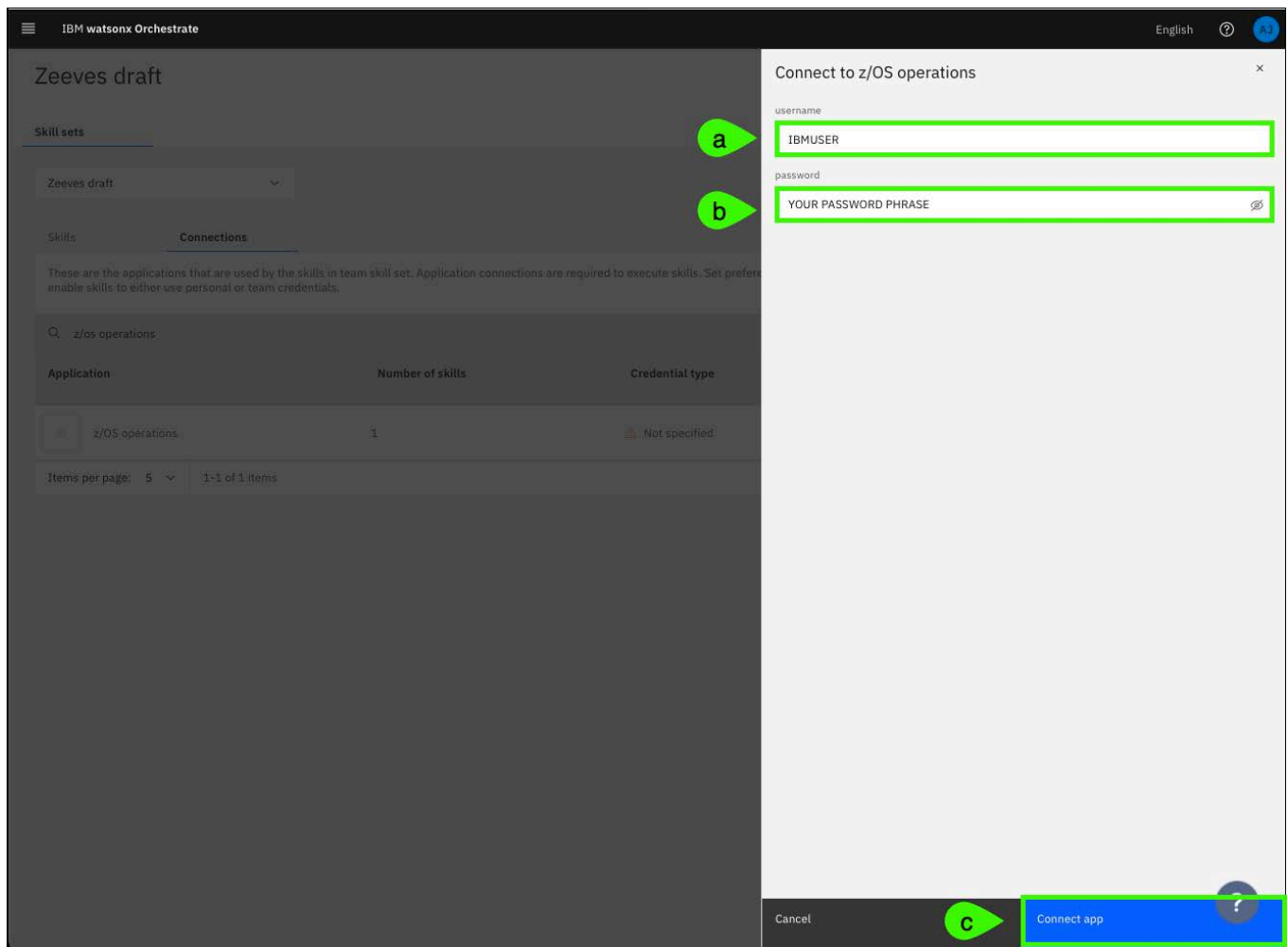
☒ Team credentials

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) **IBMUUSER** in the username field, (b) your **WaaS password** in the password field, and then (c) click **Connect app**.

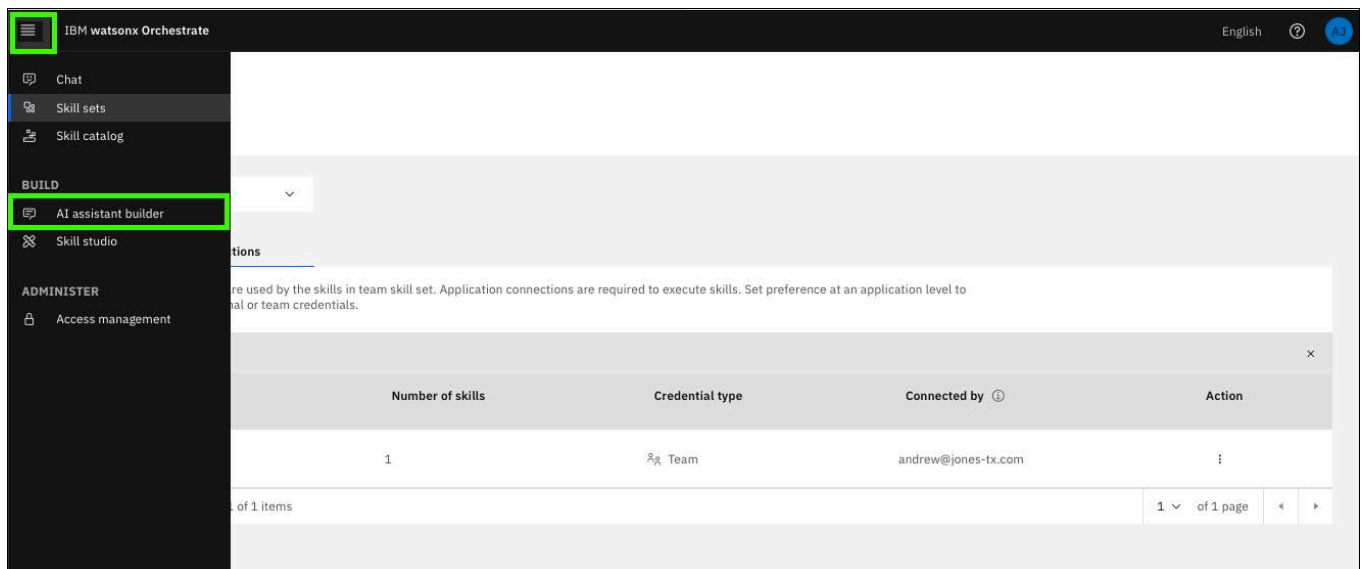


Create an action for your assistant

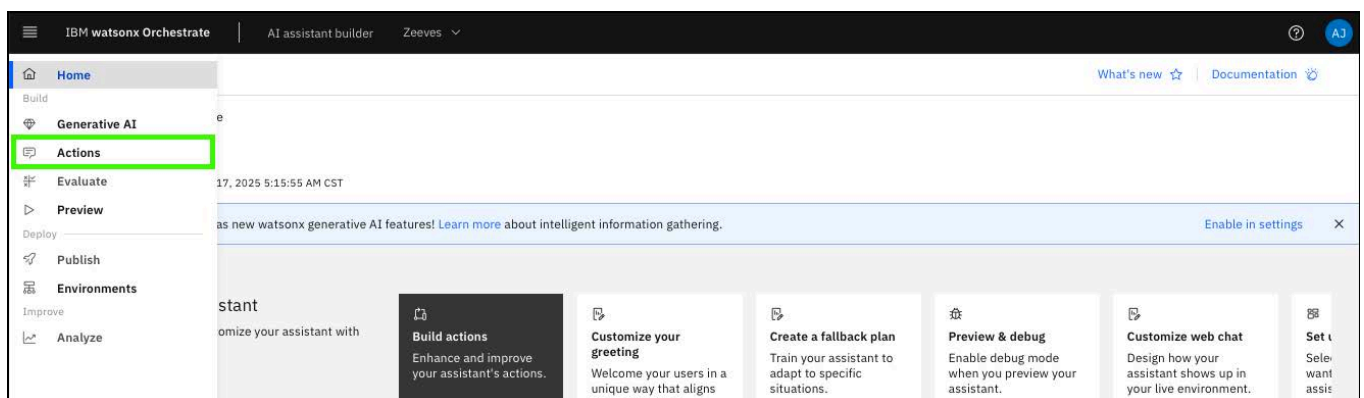
Next, create a skill-based action that uses the **z/OS IPL Information**. Recall, with the **z/OS Gather Facts** skill flow that was created earlier, adding the skill as a skill-based action allows the skill to run based on user prompts to the assistant.

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.

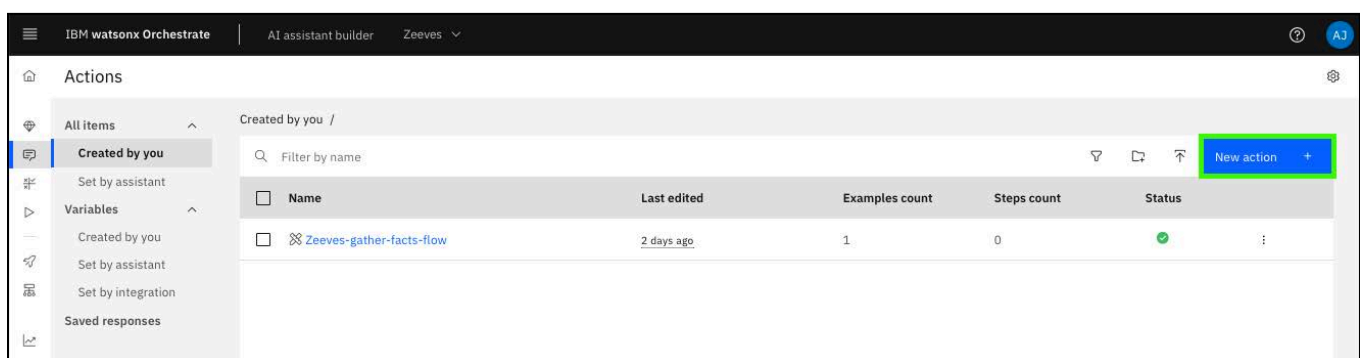
1. Open **AI assistant builder** in watsonx Orchestrate.



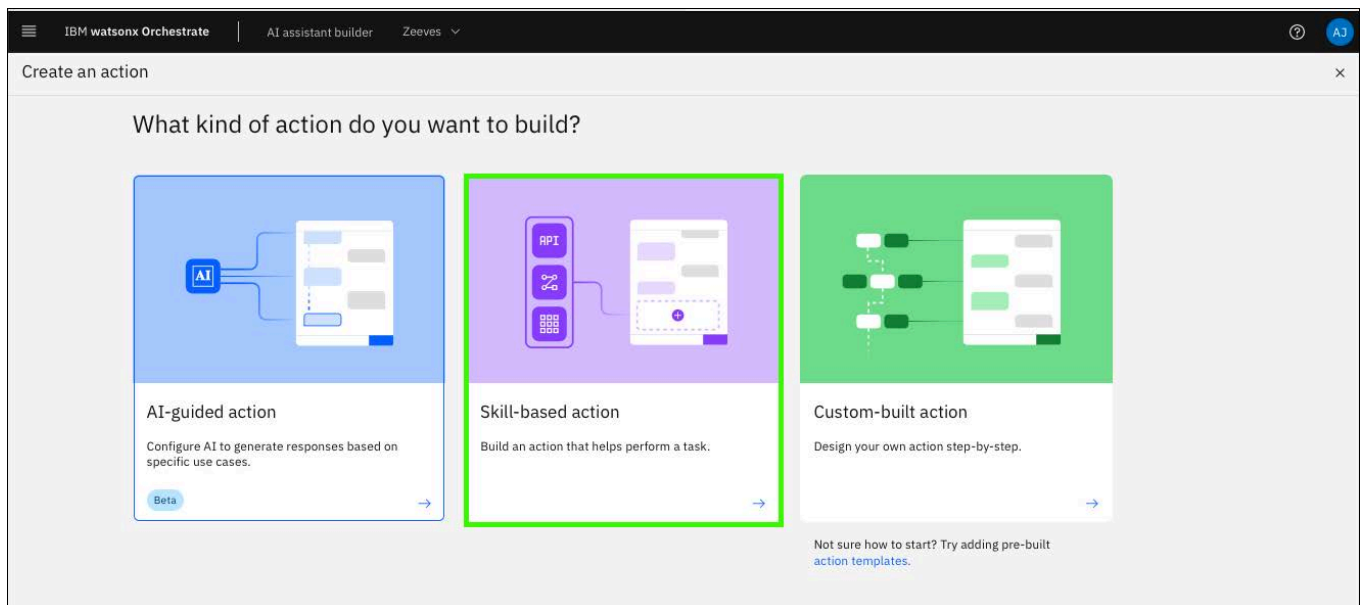
2. Select **Actions**.



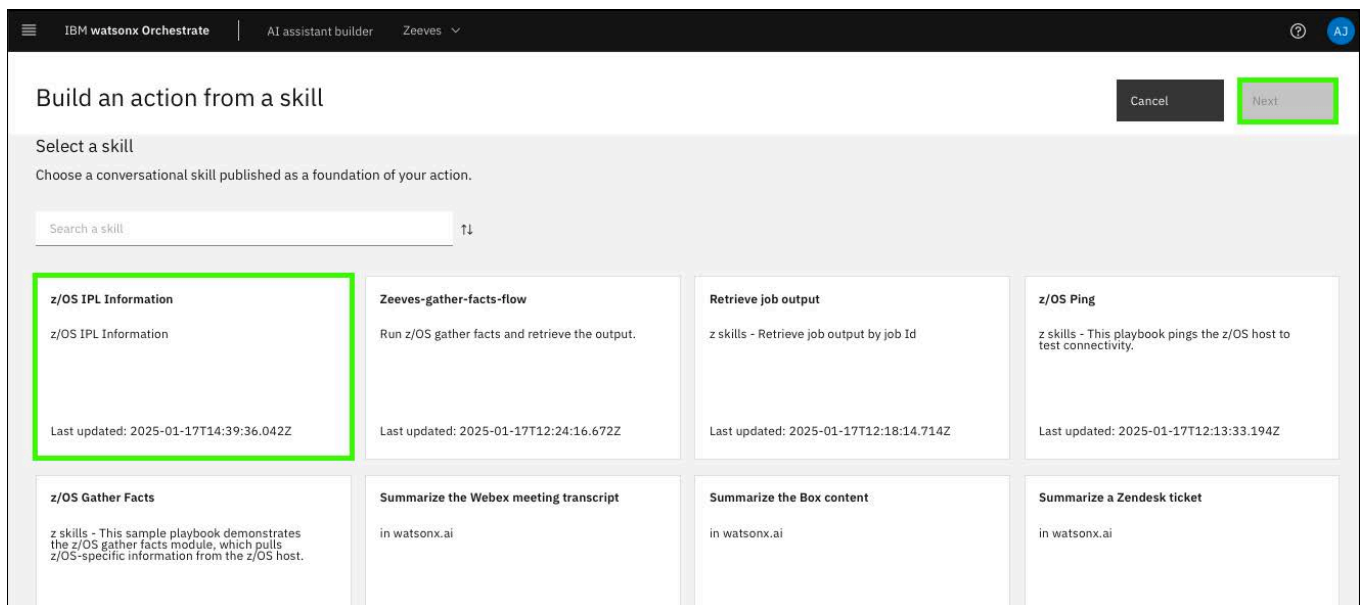
3. Click **New action +**.



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



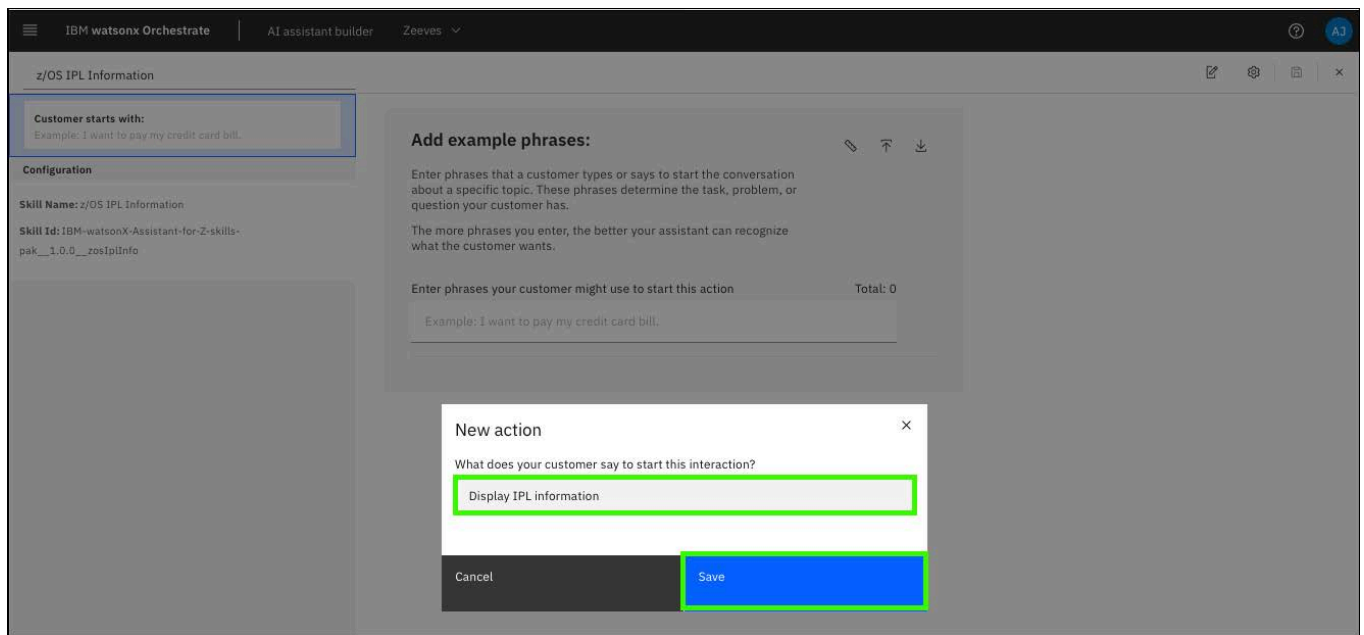
5. Click **z/OS IPL Information** and then click **Next**.



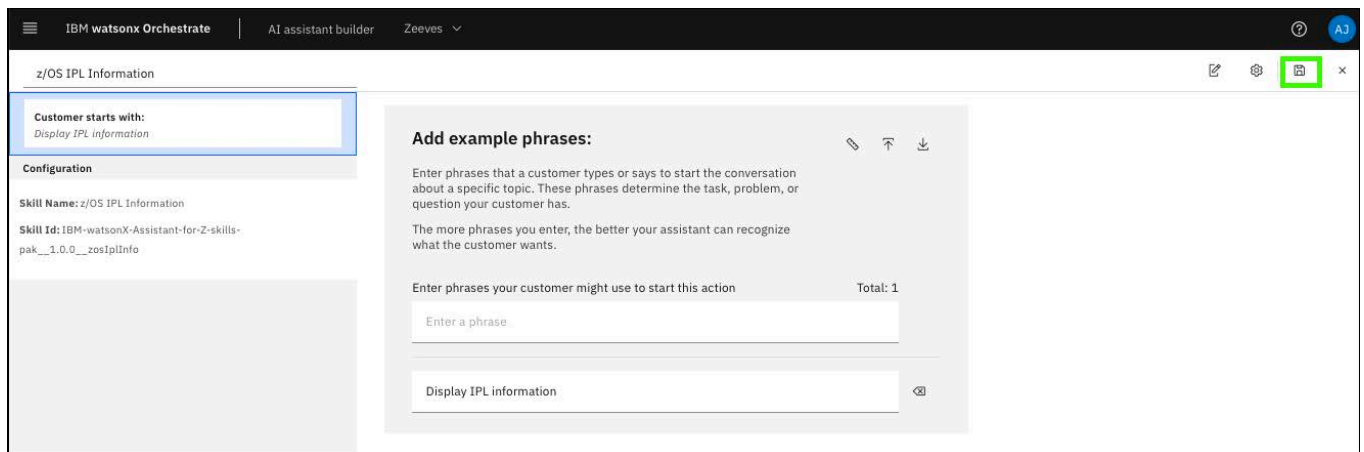
6. Enter a prompt like **Display IPL information** that starts the skill and then click **Save**.

Prompt:

Display IPL information



7. Click the **Save** icon (💾).



8. Click **Preview**.

The screenshot shows the IBM watsonx Assistant for Z interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main content area is titled 'z/OS IPL Information'. On the left, under 'Customer starts with:', there is a text input field containing 'Display IPL information'. Below this, the 'Configuration' section shows the 'Skill Name' as 'z/OS IPL Information' and the 'Skill Id' as 'IBM-watsonX-Assistant-for-Z-skills-pak__1.0.0__zosIplInfo'. The right side of the interface is titled 'Add example phrases:' and contains instructions: '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.' Below the instructions, there is a section 'Enter phrases your customer might use to start this action' with a 'Total: 1' indicator. A text input field contains 'Enter a phrase', and a button labeled 'Display IPL information' is visible. In the bottom right corner, there is a 'Preview' button with a question mark icon.



Wait for the Your changes are being added. message to disappear before proceeding.

9. Enter the prompt (**Display IPL information**) to test the skill.

The screenshot displays the IBM watsonx Assistant for Z interface. The top navigation bar includes the logo, 'AI assistant builder', and a user profile 'Zeeves'. The main workspace is titled 'z/OS IPL Information'. On the left, a 'Configuration' panel shows the skill name and ID. The central area, 'Add example phrases:', provides instructions on how to enter phrases and a list of phrases: 'Enter a phrase' and 'Display IPL information'. A 'Preview' window on the right shows a chat interaction where the assistant responds with 'Welcome, how can I assist you?'. A green box highlights the 'Display IPL information' phrase in the list, and a blue circle with a question mark is next to it in the preview window.

IBM watsonx Orchestrate | AI assistant builder | Zeeves

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**.

The screenshot displays the IBM watsonx Orchestrate AI assistant builder interface. The top navigation bar includes the IBM watsonx Orchestrate logo, the AI assistant builder tab, and the Zeeves user profile. The main workspace is titled 'z/OS IPL Information' and is divided into two main sections: Configuration and Add example phrases.

Configuration:

- Customer starts with:** Display IPL information
- 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

Conversational 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.

The screenshot shows the IBM watsonx Orchestrate AI assistant builder interface. The main section is titled 'Add example phrases:'. It contains instructions: '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.' Below this, there is a section 'Enter phrases your customer might use to start this action' with a 'Total: 1' count. A text input field contains 'Display IPL information'. To the right, a 'Preview' window shows a sample command response for 'z/OS 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

☐ sol-key-timeout

☐ sol-key-detected

cmd-response

```
IEE254I 11.04.51 IPLINFO DISPLAY
SYSTEM IPLD AT 06.48.48 ON 01/11
RELEASE z/OS 02.05.00 LICENSE =
USED LOADK2 IN SYS0.IPLPARM ON
ARCHLVL = 2 MTLSHARE = N
VALIDATED BOOT: NO
IEASYS LIST = (00,K2)
IEASYS LIST = (00) (OP)
IODF DEVICE: ORIGINAL(ODE28) CURF
IPL DEVICE: ORIGINAL(ODE27) CURF
VM CPID = zHYPaas
VM UUID = 3C9BD7F3-8A47-
4A47-B62C-298DA45F40EB
VM NAME = k8s_75d1
VM EXT NAME = k8s_75d16281c12c-
8a47-
4A47-B62C-298DA45F40EB
```

Type something...

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.

The screenshot shows the IBM watsonx Orchestrate AI assistant builder interface. The main section is titled 'Add example phrases:'. It contains instructions: '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.' Below this, there is a section 'Enter phrases your customer might use to start this action' with a 'Total: 1' count. A text input field contains 'Display IPL information'. A green box highlights the delete icon (🗑️) next to the phrase. To the right, a 'Preview' window shows a sample command response for 'z/OS 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

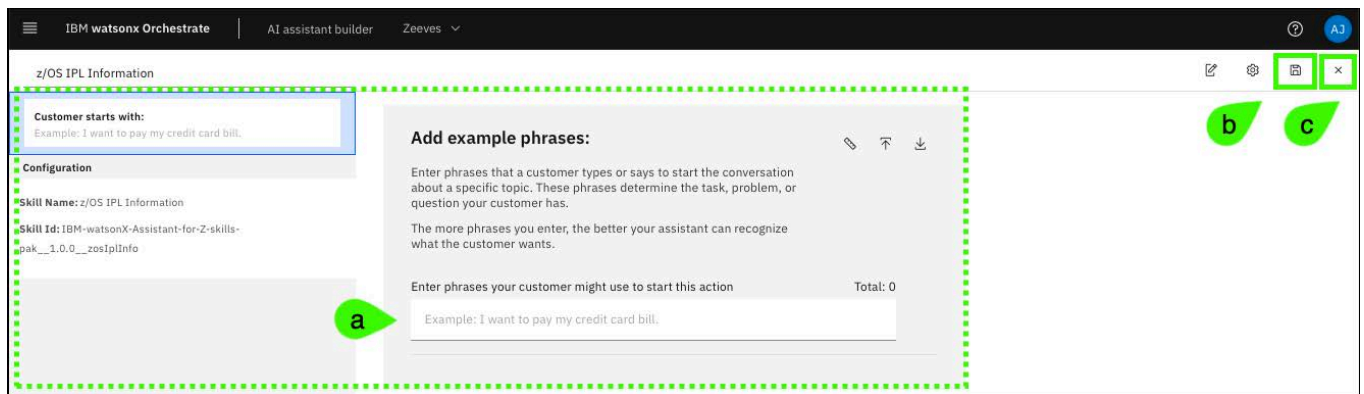
Enter a phrase

Display IPL information

Preview

z/OS IPL Information

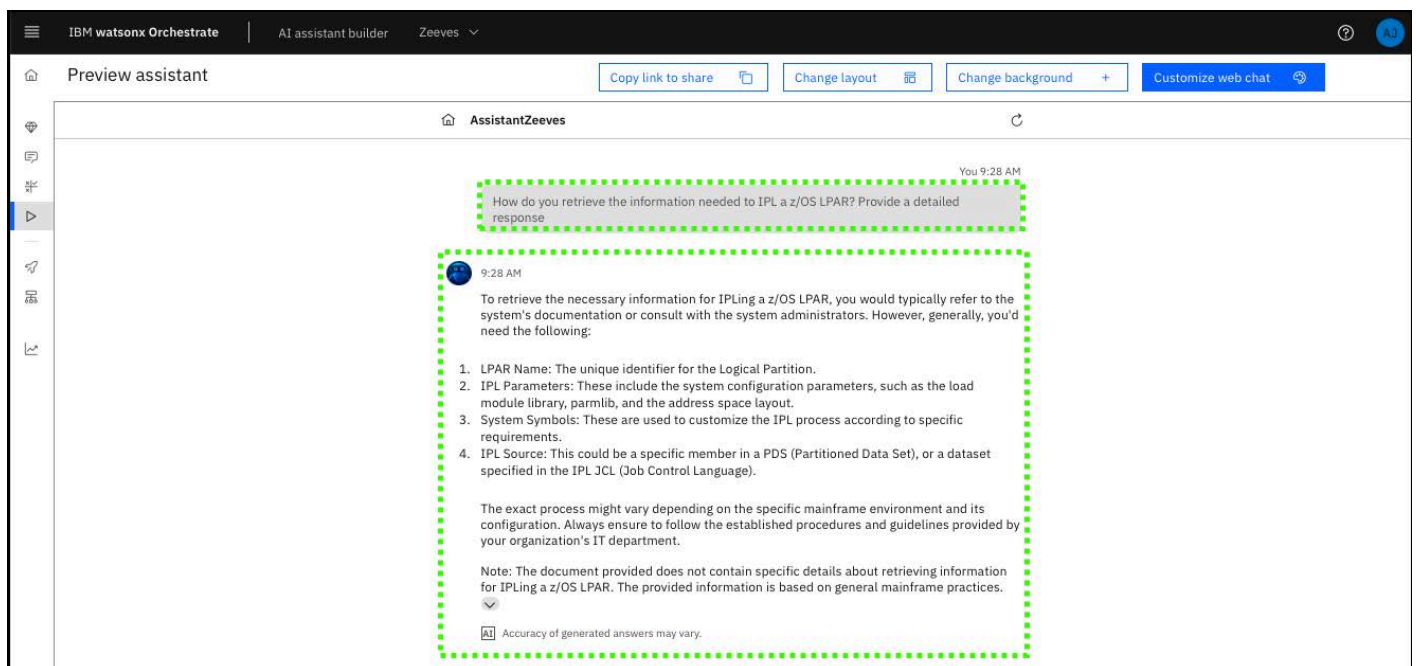
13. Verify (a) all **example phrases** are deleted, (b) click the **Save** icon (💾), and then (c) click **x** to close the action.



Create a custom-built action

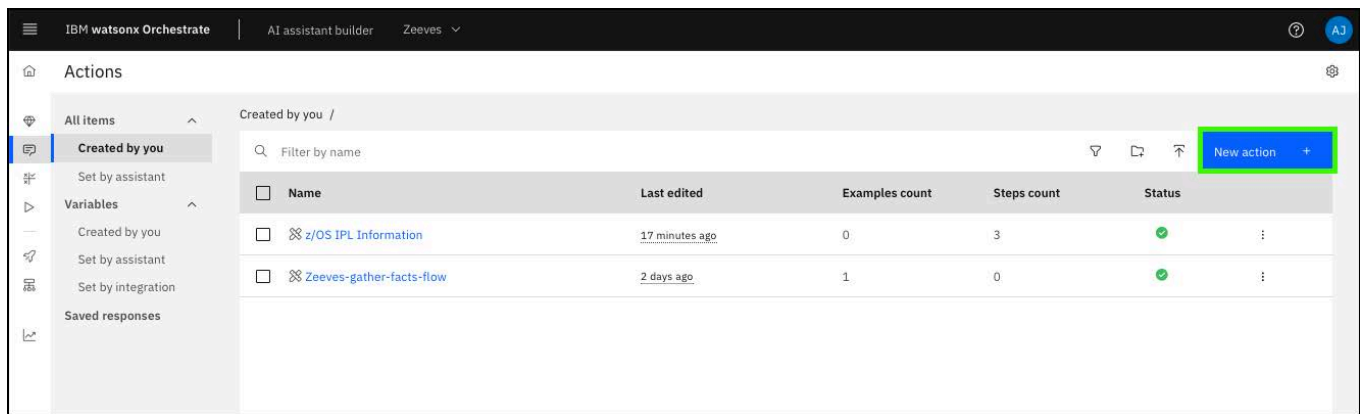
Custom-built actions allow you to define each step of a conversation with your assistant. You can define steps with or without conditions to control the user responses. You can include your skill-based actions as sub-actions. You can pass generated values between the subactions.

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.

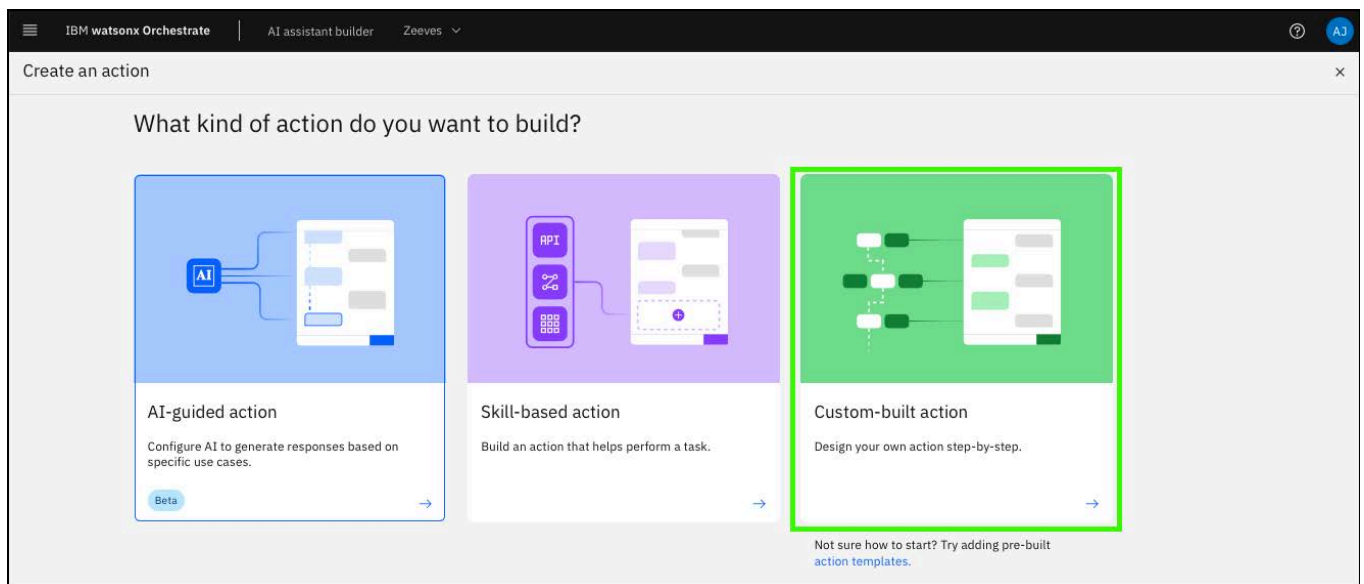


A custom-built action can help the SysProg to take the next steps that are required by adding next-step action suggestions and start skills to gather the needed information.

1. Click **New action +**.



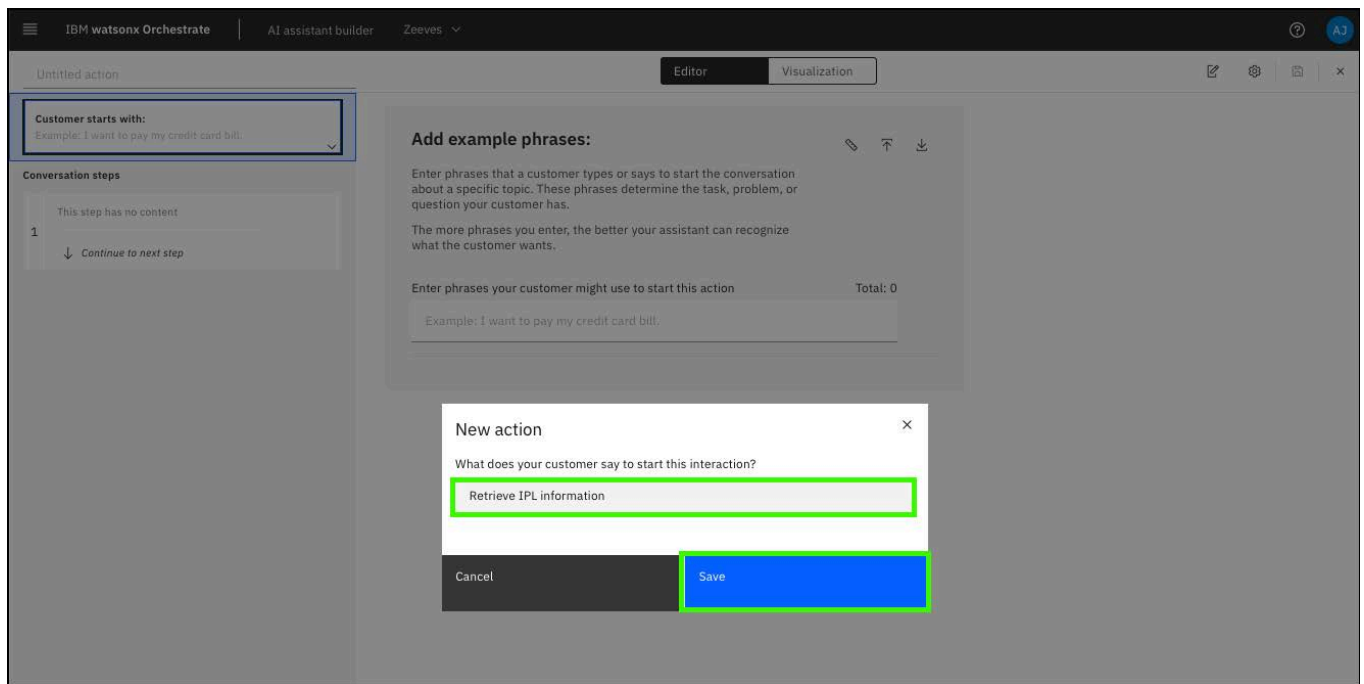
2. Click **Custom-built action**.



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



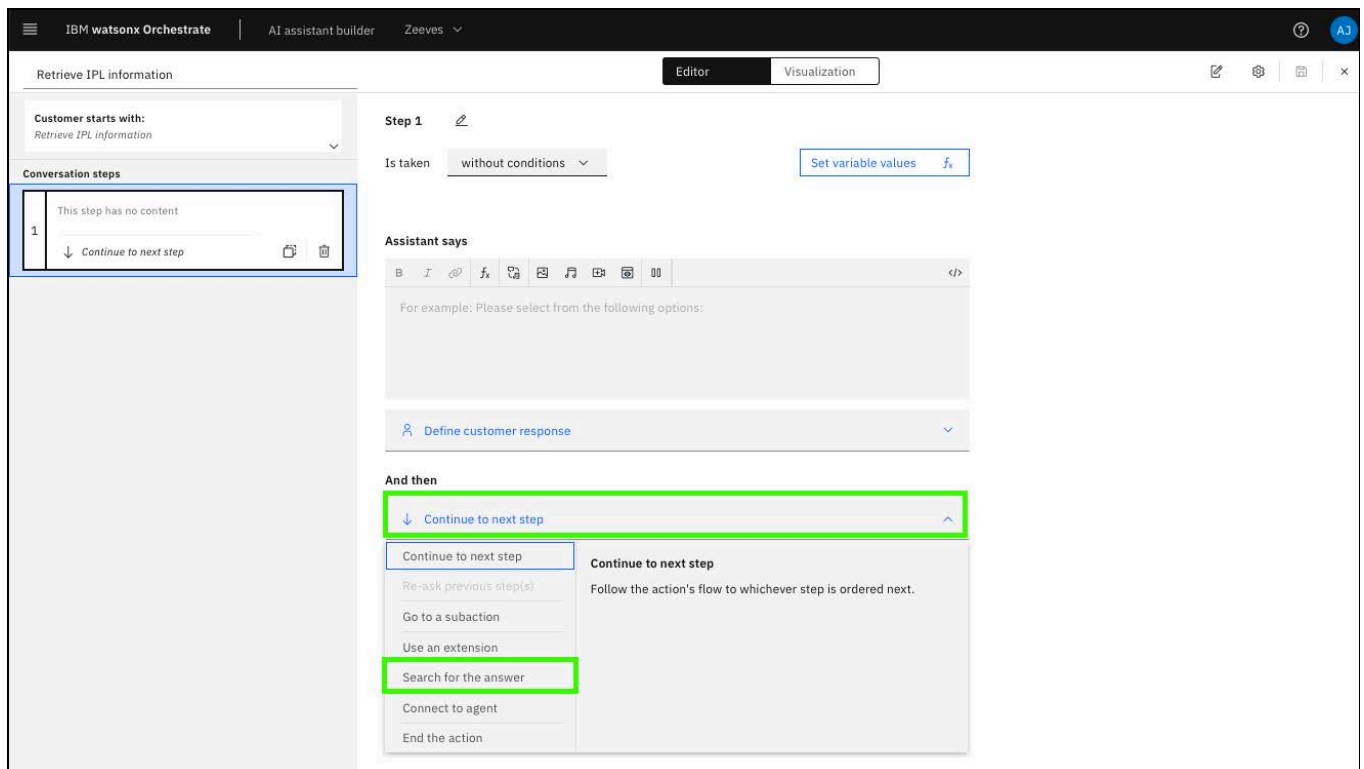
4. Review the form to create a custom-built action.

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.

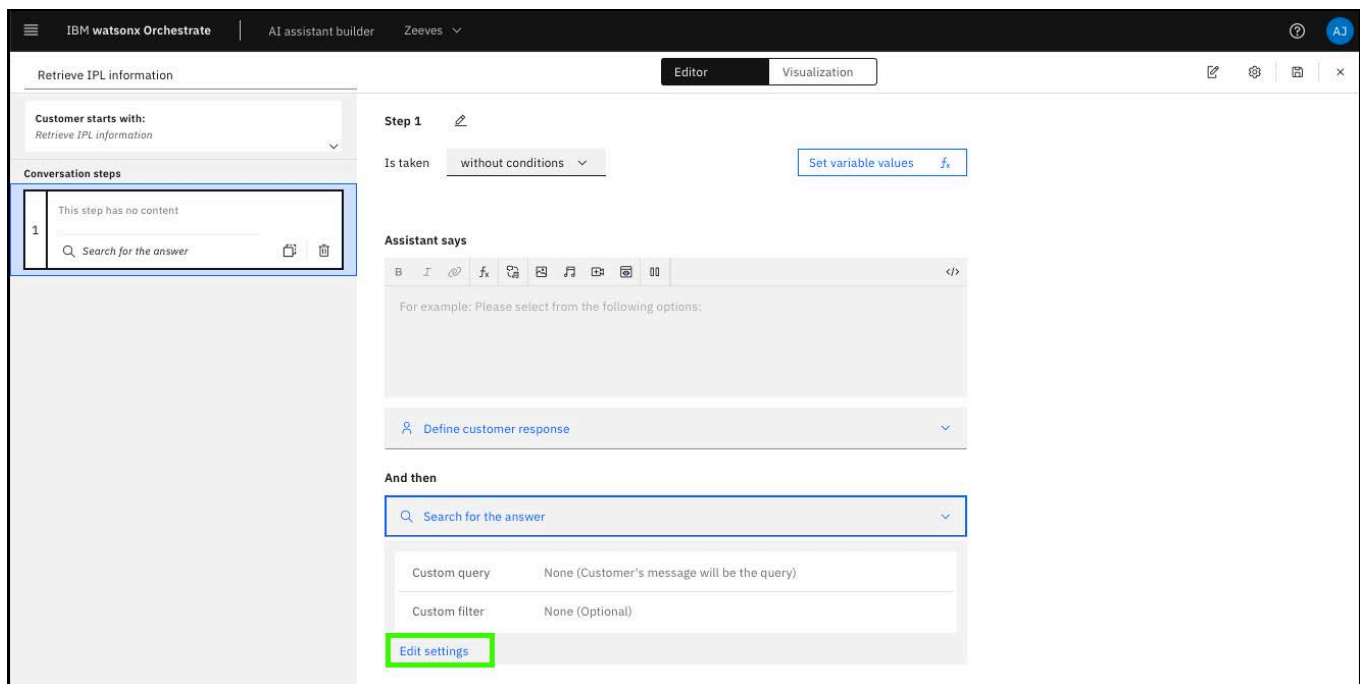
For client demonstrations and pilots...

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**.



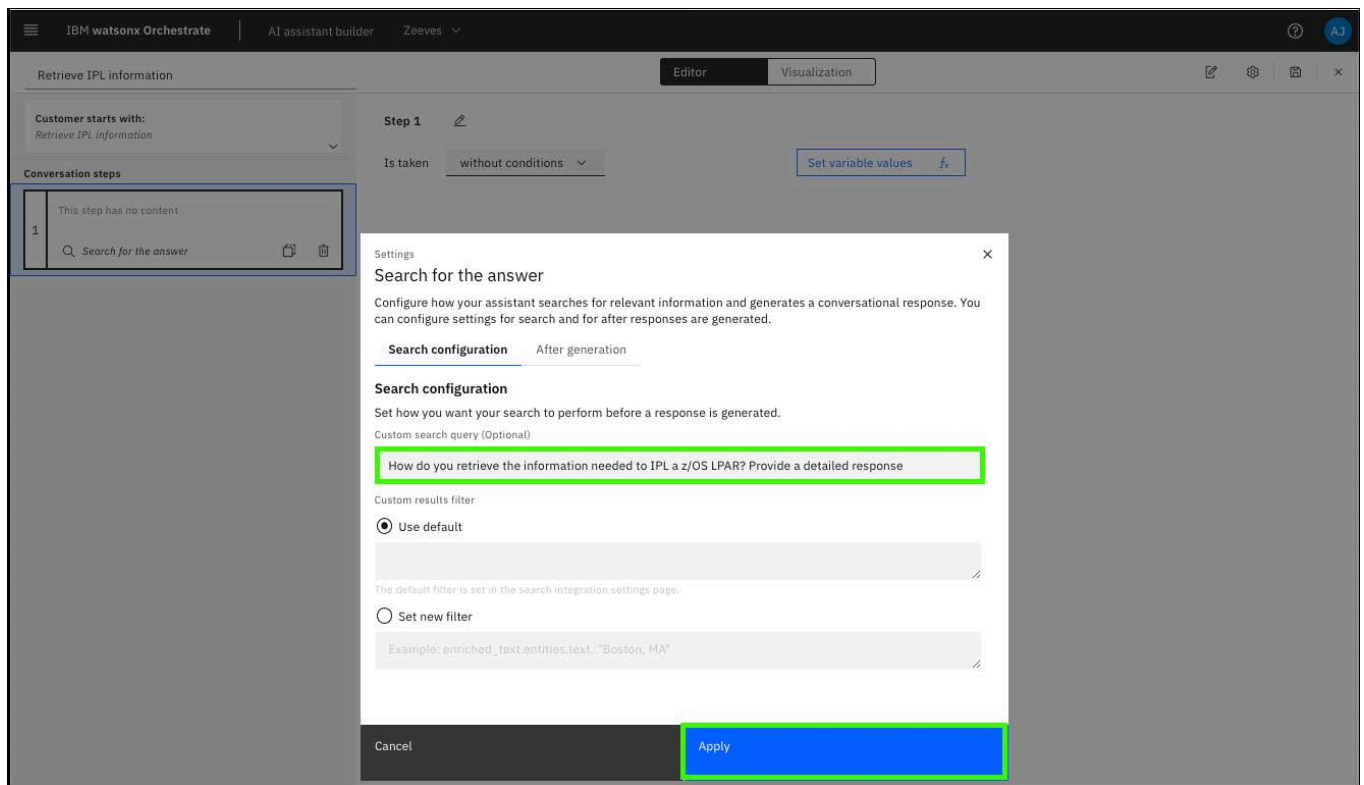
6. Click **Edit settings**.



7. Enter a **Custom search query** and click **Apply**.

Custom search query:

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



8. Click **New 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.

Retrieve IPL information

Customer starts with:
Retrieve IPL information

Conversation steps

1 This step has no content
Search for the answer

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

Custom filter None (Optional)

Edit settings

New step

Preview

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?

Open the main menu

Retrieve IPL information

Customer starts with:
Retrieve IPL information

Conversation steps

1 This step has no content
Search for the answer

2 Would you like to display your LPARs IPL information?
Continue to next step

Step 2

Is taken without conditions

Set variable values

Assistant says

Would you like to display your LPARs IPL information?

Define customer response

And then

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 displays the IBM watsonx Orchestrate AI assistant builder interface. The top navigation bar includes the IBM watsonx Orchestrate logo, the AI assistant builder title, and the user name Zeeves. The main workspace is titled "Retrieve IPL information" and has tabs for "Editor" and "Visualization".

On the left, the "Conversation steps" panel shows two steps. Step 1 is labeled "This step has no content" with a search icon and the text "Search for the answer". Step 2 is highlighted with a blue border and contains the text "Would you like to display your LPARs IPL information?" followed by a "Continue to next step" button.

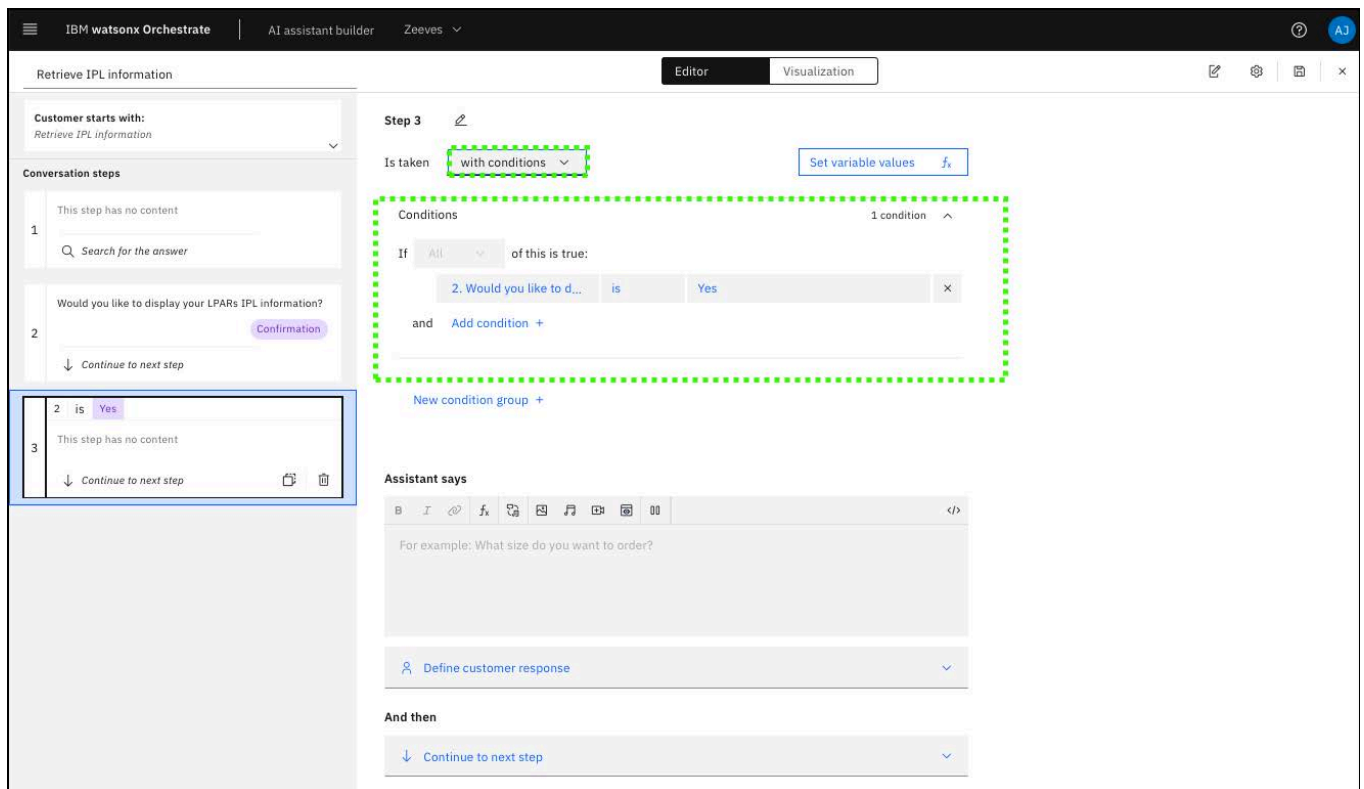
The main editor area shows "Step 2" configuration. The "Is taken" condition is set to "without conditions". The "Assistant says" section contains the text "Would you like to display your LPARs IPL information?". Below this, the "Define customer response" section is highlighted with a green border. It includes a "System" dropdown menu with "Options" and "Confirmation" (highlighted with a green box), and a "Saved" dropdown. The "Options" section is expanded, showing "As buttons" with options "Checking", "Savings", "401 (k)", and "Roth IRA", and "As a list" with the option "Pay Bill".

11. Click **New step +**.

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

13. Review the **Conditions**.

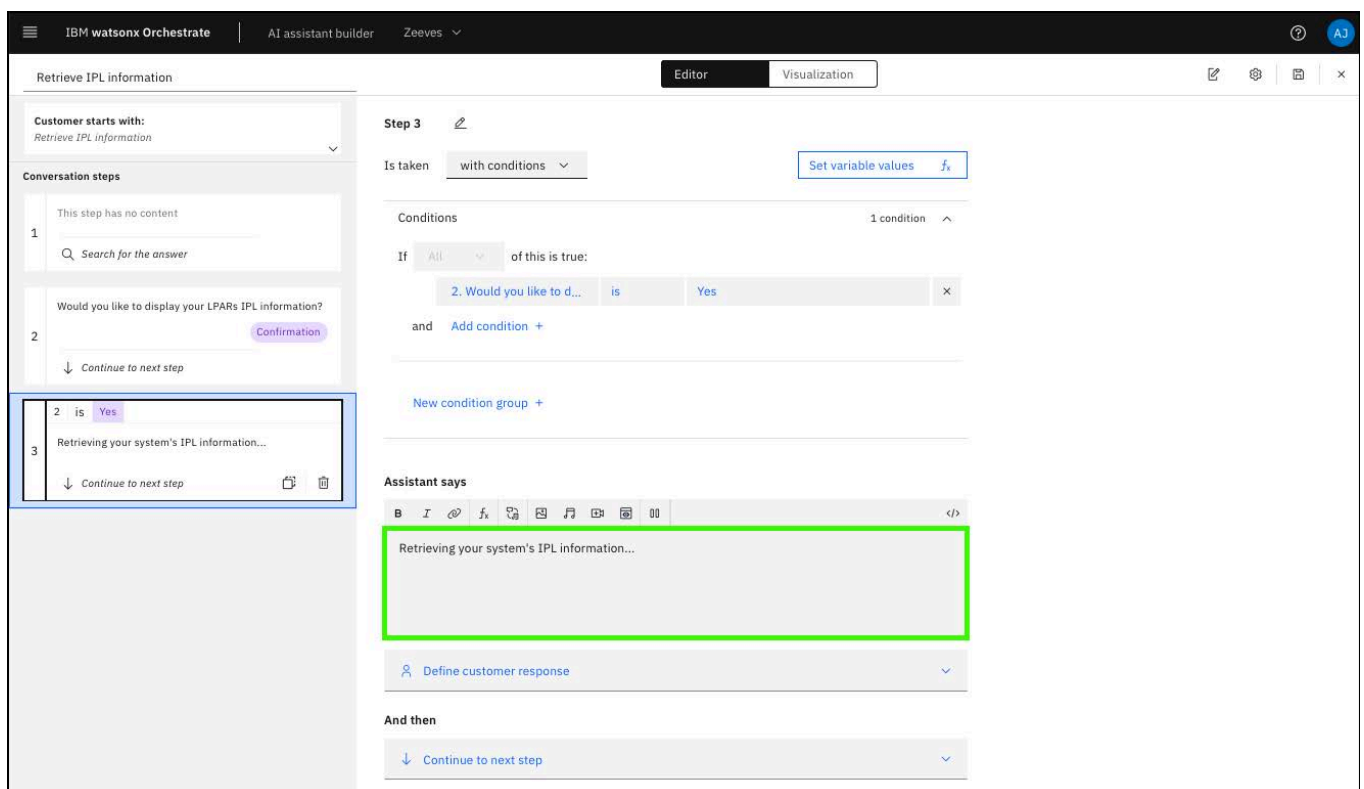
Notice that the default condition is based on **Step 2** and the user clicking **Yes** when prompted.



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

Assistant says:

Retrieving your system's IPL information...



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

Retrieve IPL information

Customer starts with: Retrieve IPL information

Conversation steps

- 1 This step has no content
- 2 Would you like to display your LPARs IPL information? Confirmation
- 3 Retrieving your system's IPL information...

Step 3

Is taken with conditions

Conditions 1 condition

If All of this is true:

2. Would you like to display your LPARs IPL information? 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

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

Continue to next step

Follow the action's flow to whichever step is ordered next.

New step +

Preview ?

16. Click (a) the **Go to** drop-down, (b) select your skill-based action (**z/OS IPL Information**), and then (c) click **Apply**.

Retrieve IPL information

Customer starts with: Retrieve IPL information

Conversation steps

- 1 This step has no content
- 2 Would you like to display your LPARs IPL information? Confirmation
- 3 Retrieving your system's IPL information...

Step 3

Is taken with conditions

Conditions 1 condition

If All of this is true:

2. Would you like to display your LPARs IPL information? 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

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

Continue to next step

Follow the action's flow to whichever step is ordered next.

New step +

Preview ?

Settings

Go to action

Go to

z/OS IPL Information

End this action after the subaction is completed

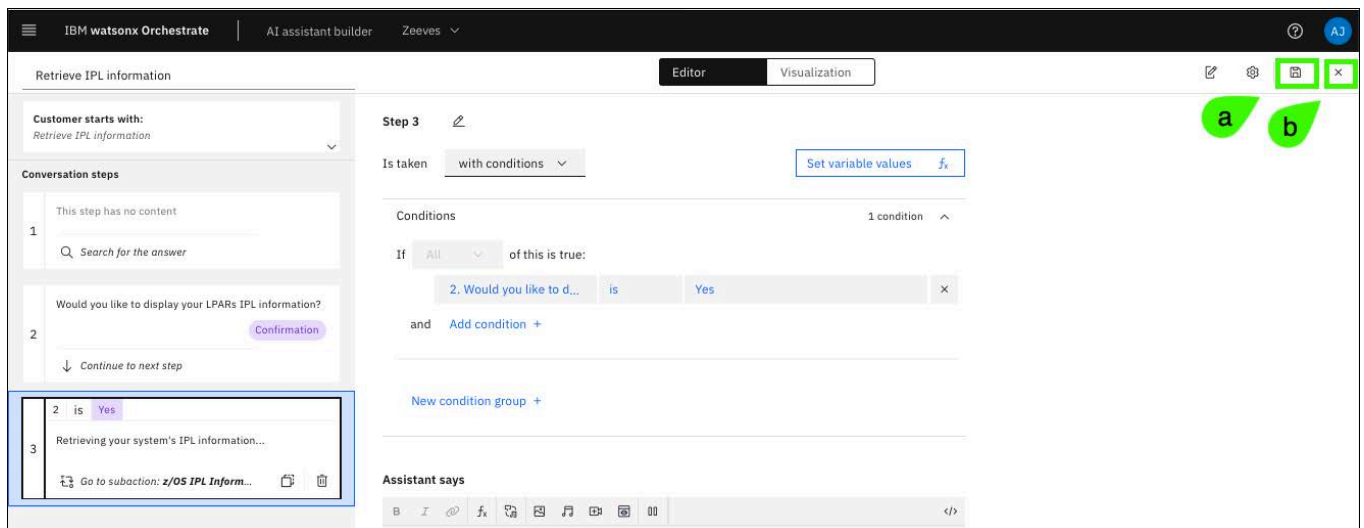
Privacy

Protect data returned after the subaction completes

Cancel

Apply

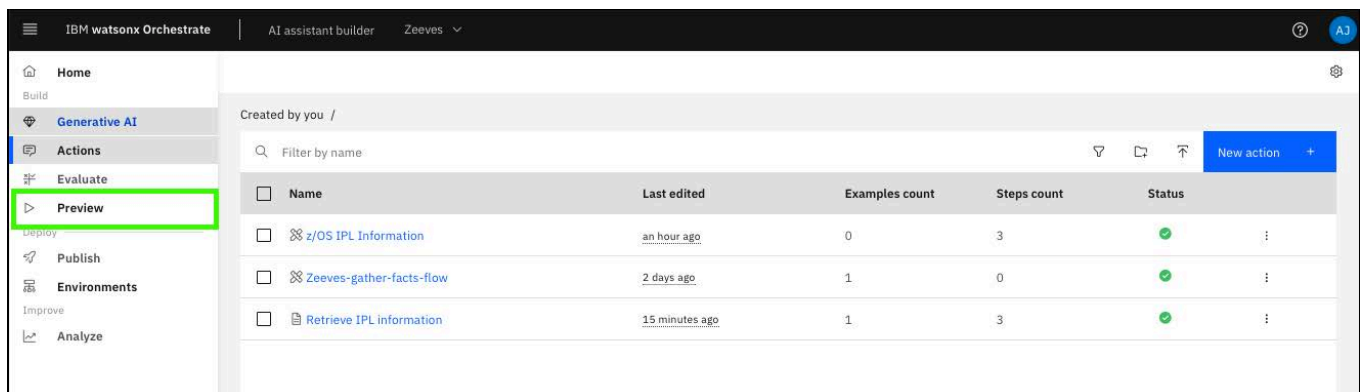
17. Click **Save** (📁) and then click **x**.



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**.



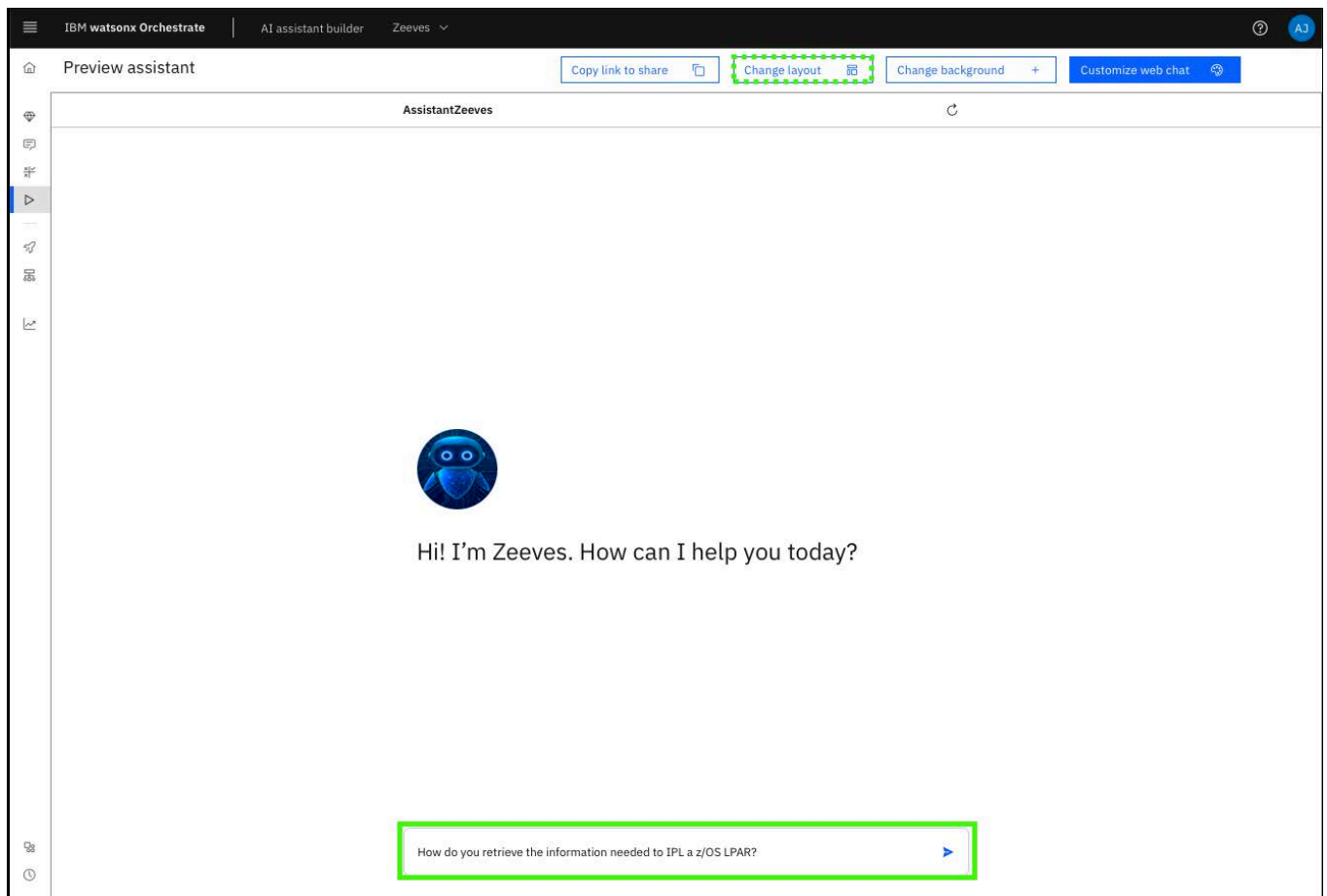
2. Enter **How do you retrieve the information needed to IPL a z/OS LPAR?** in the assistant prompt.

Prompt:

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



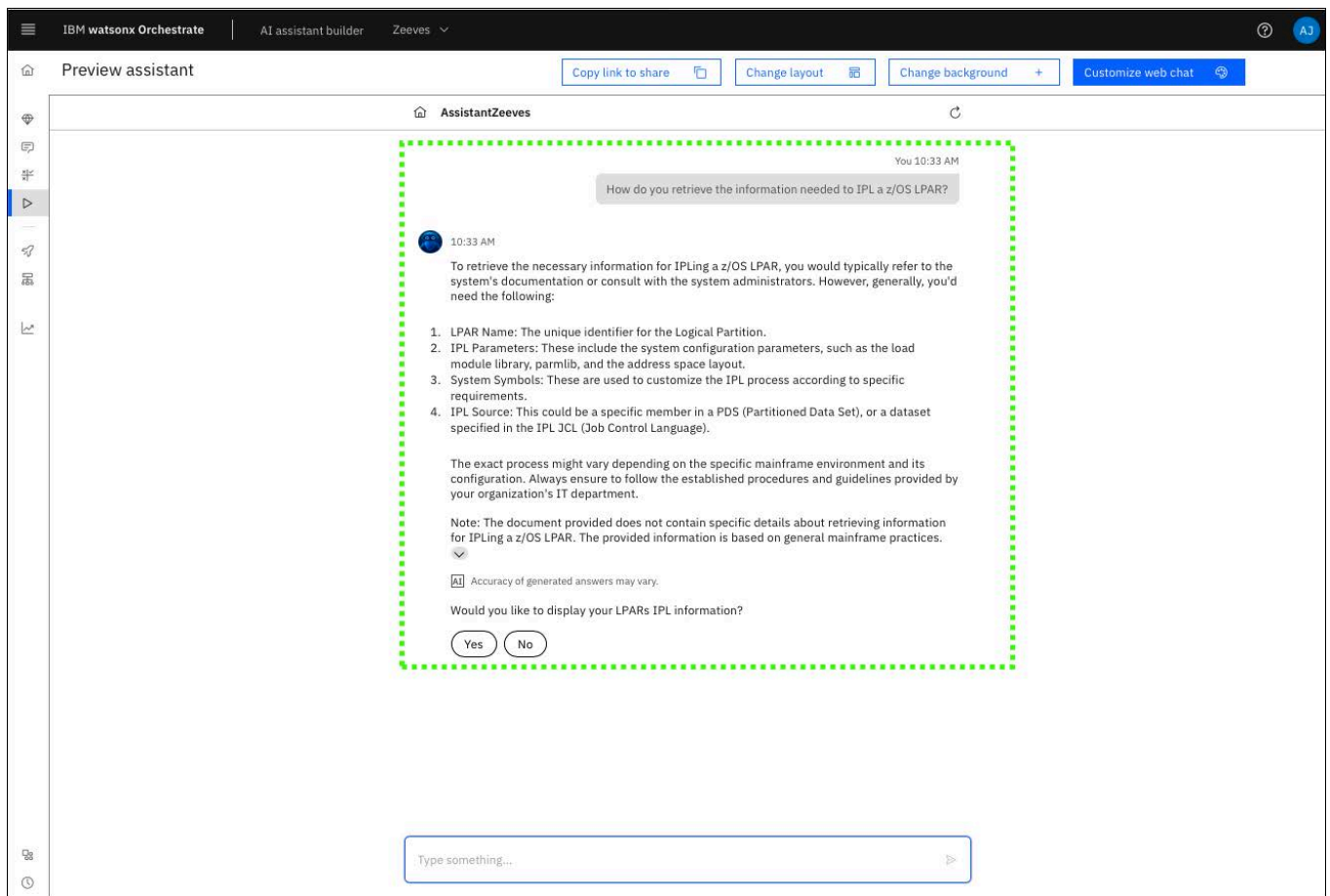
Use the Change layout option to open a full page view of the assistant.



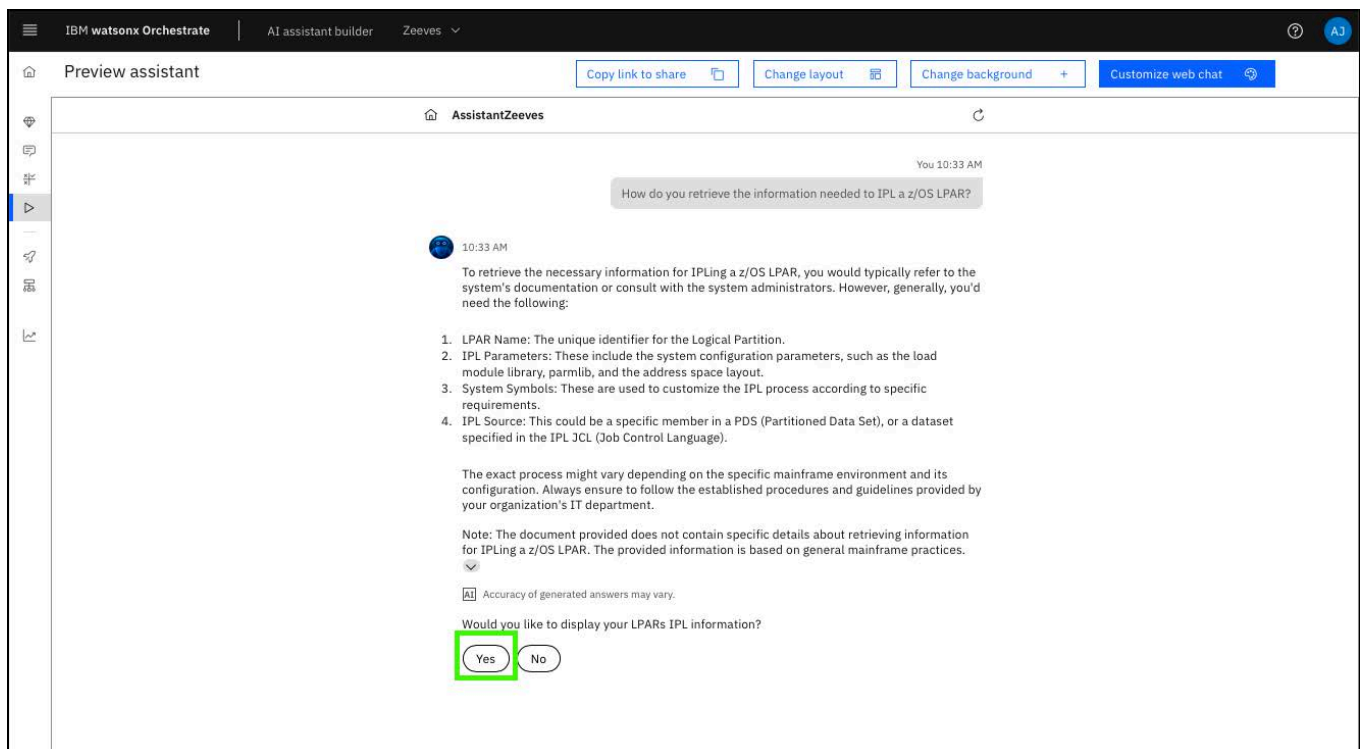
3. Review the response.

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.

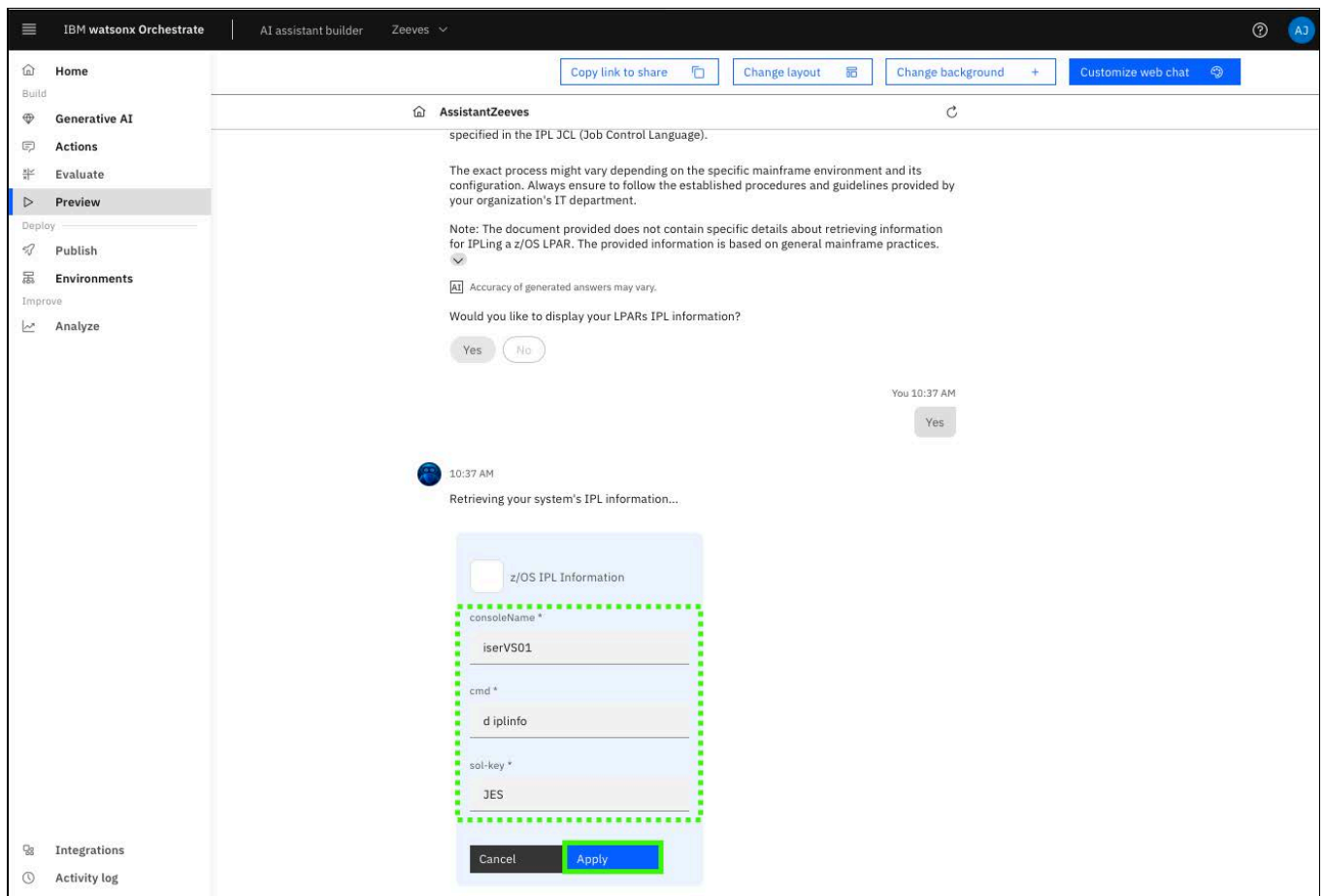
You are also able to improve their productivity by infusing automations into the natural conversation that will provide them exactly the information they need to accomplish their task.



4. Click **Yes**.

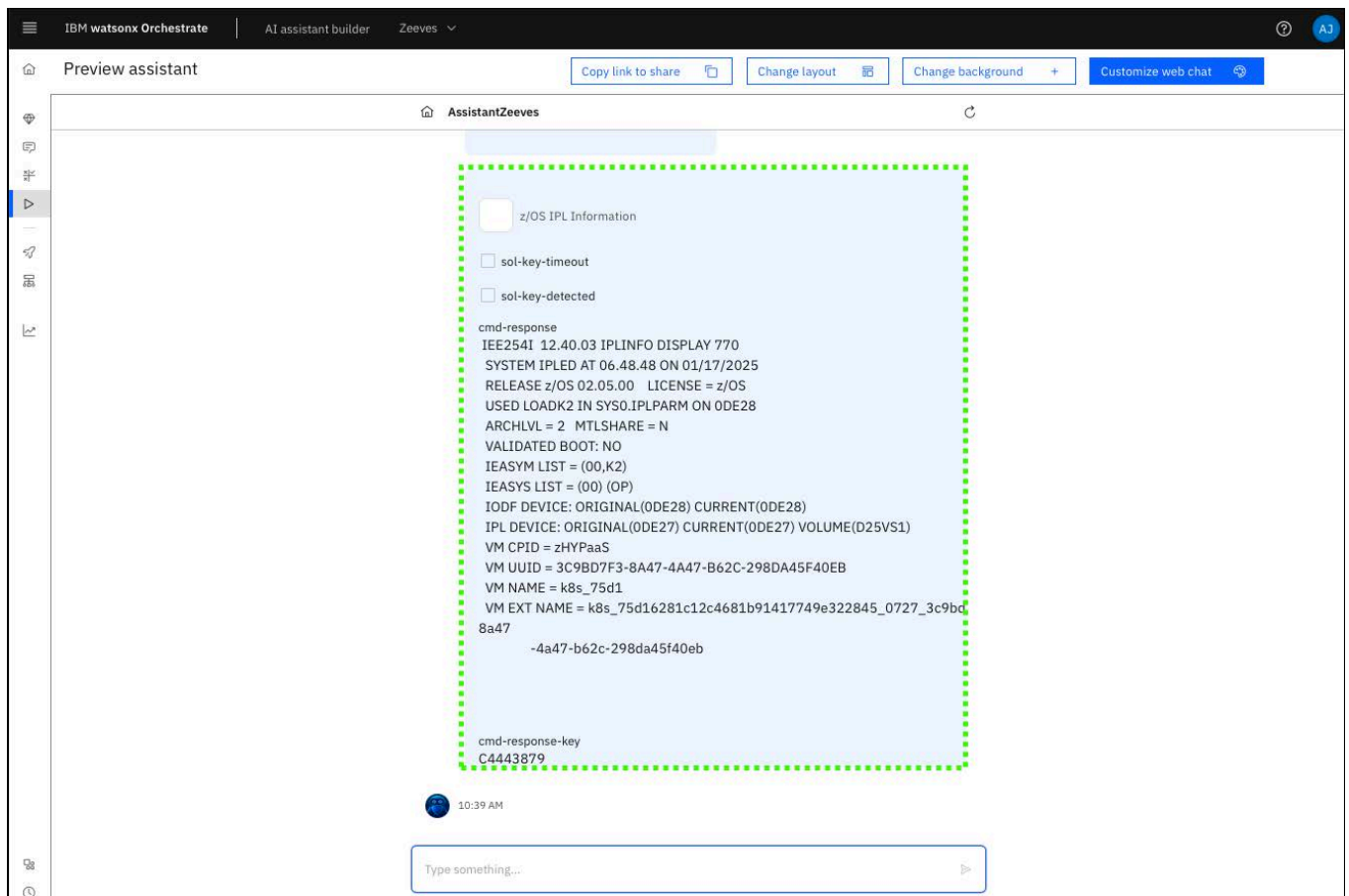


5. Using the defaults in the form, click **Apply**.



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 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.

Jobs

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.

Templates

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.

Red Hat Ansible Automation Platform

Views

Launch | z/OS Certs - Create Cert

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

1 Survey

2 Preview

Owner ID ⓘ

IBMUSER

Certificate Label ⓘ

TESTCA

Type * ⓘ

CERTAUTH

Sign With ⓘ

Self

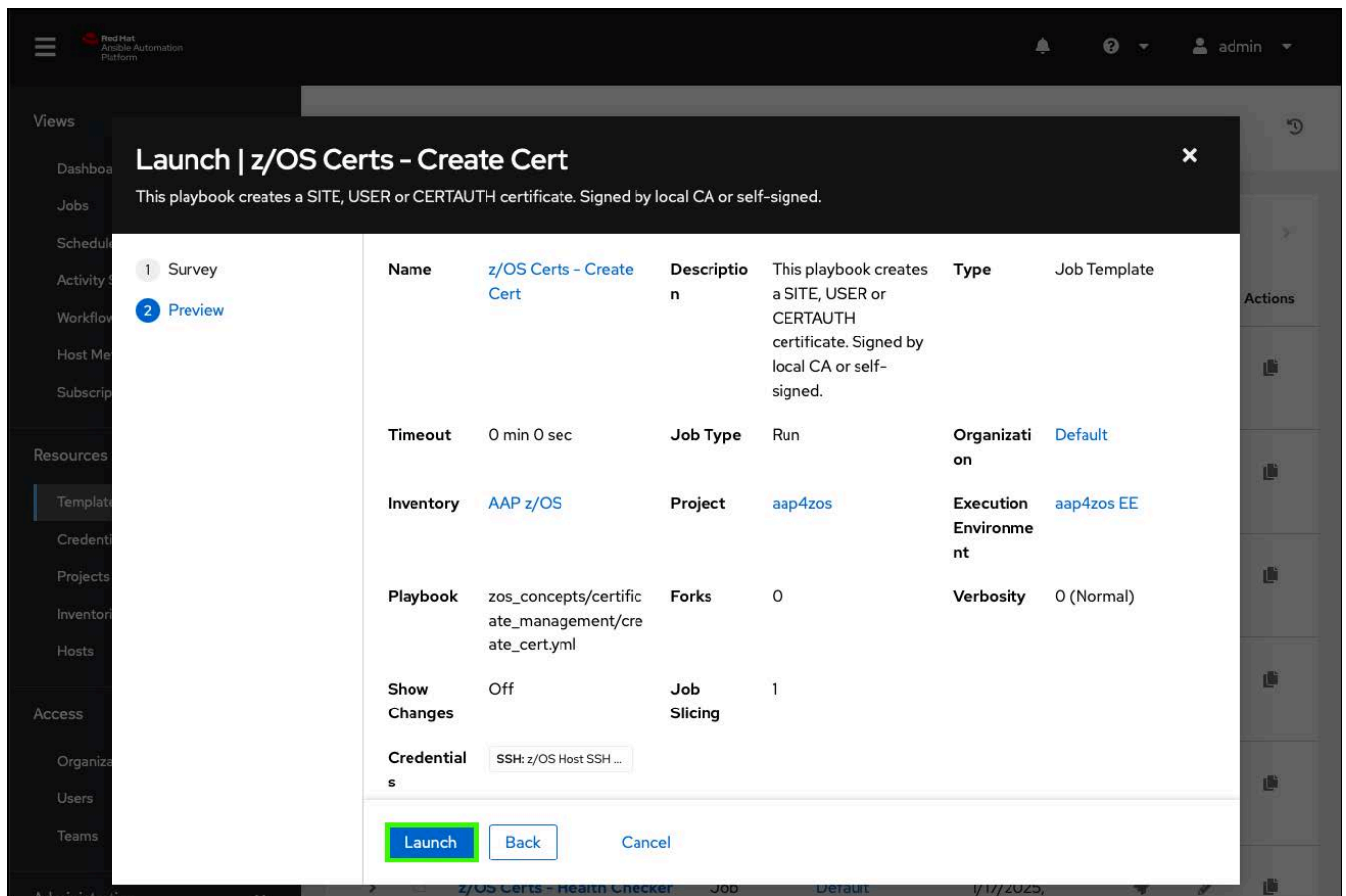
Sign Label ⓘ

Organizational Unit ⓘ

ANSIBLE CORE

Next Back Cancel

5. Click **Launch**.



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 displays the Red Hat Ansible Automation Platform interface. On the left is a sidebar with navigation menus: 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 the 'Output' of a job titled 'z/OS Certs - Create Cert', which is marked as 'Successful'. The job summary indicates 1 Play, 15 Tasks, 1 Host, and an elapsed time of 00:00:20. Below this, a search bar is set to 'Stdout'. A notification states 'Events processing complete. Reload output'. The task output is shown in a scrollable list, with a green dashed box highlighting the following tasks:

```

10
11 TASK [Get FQDN of this z/OS server from USS.] ***** 12:58:30
12 ok: [zos_host]
13
14 TASK [Set cn fact from previous task.] ***** 12:58:31
15 ok: [zos_host]
16
17 TASK [Check if certificate to be created already exists.] ***** 12:58:31
18 ok: [zos_host]
19
20 TASK [Debug command output.] ***** 12:58:32
21 ok: [zos_host] => {
22   "msg": "4"
23 }
24
25 TASK [Debug.] ***** 12:58:32
26 ok: [zos_host] => {
27   "cert_not_found": true
28 }
29

```

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

Views: Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals, Host Metrics, Subscription Usage

Resources: Templates, Credentials, Projects, Inventories, Hosts

Access: Organizations, Users, Teams

Administration: Credential Types, Notifications

Jobs > 16 - z/OS Certs - Create Cert

Output

Back to Jobs Details Output

z/OS Certs - Create Cert Successful Plays 1 Tasks 15 Hosts 1 Elapsed 00:00:20

Stdout

Events processing complete. Reload output

```

27     "cert_not_found": true
28   }
29
30   TASK [Generate new certificate.] ***** 12:58:32
31   changed: [zos_host]
32
33   TASK [List chain for new certificate.] ***** 12:58:38
34   changed: [zos_host]
35
36   TASK [Connect new certificate to keyring.] ***** 12:58:39
37   changed: [zos_host]
38
39   TASK [List keyring.] ***** 12:58:40
40   changed: [zos_host]
41
42   TASK [Stop TN3270 server.] ***** 12:58:41
43   skipping: [zos_host]
44
45   TASK [Start TN3270 server] ***** 12:58:41
46   skipping: [zos_host]

```

8. Click **JSON**.

Views: Dashboard, Jobs, Schedules, Activity Stream, Workflow Approvals, Host Metrics, Subscription Usage

Resources: Templates, Credentials, Projects, Inventories, Hosts

Access: Organizations, Users, Teams

Administration: Credential Types, Notifications

Jobs > 16 - z/OS Certs - Create Cert

Output

Back to Jobs Details Output

z/OS Certs - Create Cert Successful Plays 1 Tasks 15 Hosts 1 Elapsed 00:00:20

Stdout

Events processing complete. Reload output

Host Details

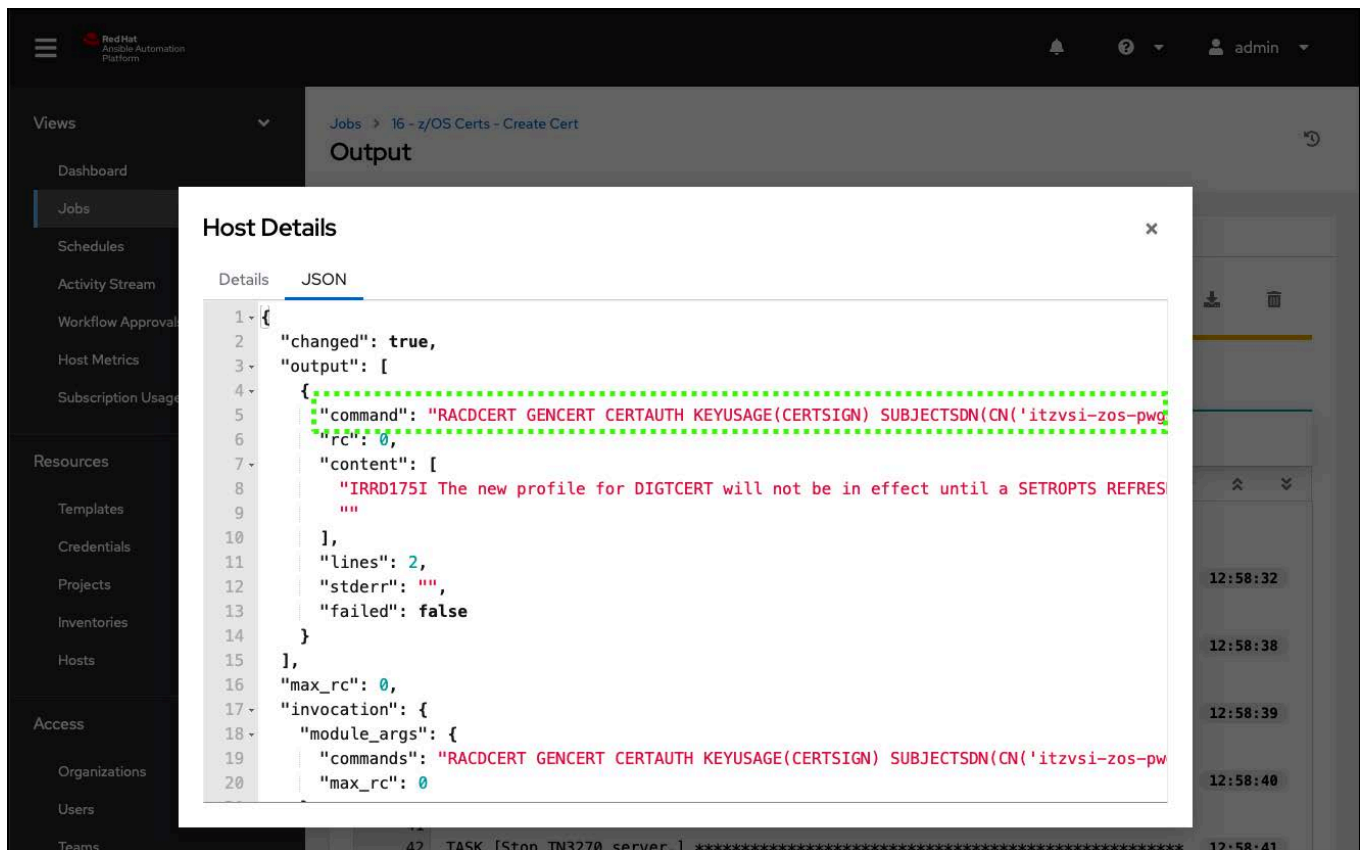
Details **JSON**

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

Task: Generate new certificate. Module: ibm.ibm_zos_core.zos_tso_comm and

Play button

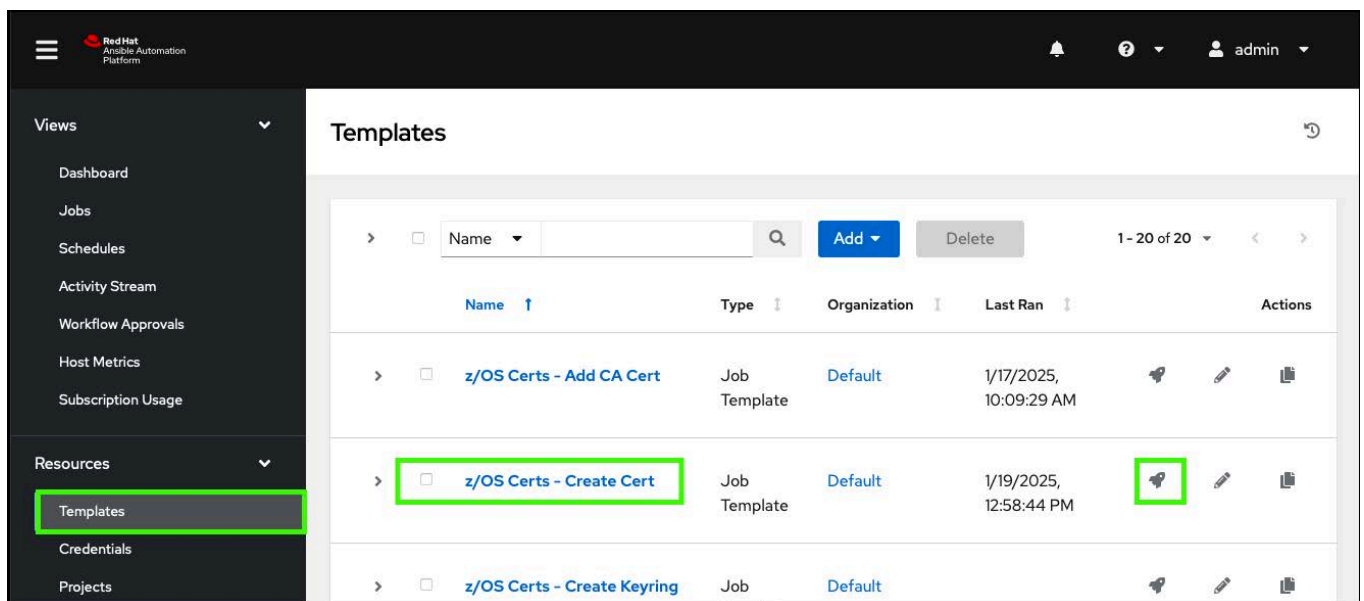
9. Review the **RACDCERT** command that was run to generate the certificate and then click **x** to close the window.



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.



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!

Launch | z/OS Certs - Create Cert

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

1 Survey
2 Preview

Organizational Unit
ANSIBLE CORE

Common Name
company.com

Country
US

Keyring
SharedRing1

Expiration Date
2025-01-30

Check Name
IBMRACF,RACF_CERTIFICATE_EXPIRATION

Next Back Cancel

3. Click **Launch**.

Launch | z/OS Certs - Create Cert

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

1 Survey
2 Preview

Execution Environment aap4zos EE

Playbook zos_concepts/certificate_management/create_cert.yml

Forks 0

Verbosity 0 (Normal)

Show Changes Off

Job Slicing 1

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

```

Launch Back Cancel

4. Verify that the job was successful and inspect the output of the job.

The screenshot displays the Red Hat Ansible Automation Platform interface. On the left is a sidebar with navigation menus: 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 panel shows the 'Jobs' section with a breadcrumb 'Jobs > 17 - z/OS Certs - Create Cert'. The 'Output' tab is selected, showing a job status of 'Successful' (indicated by a green checkmark in a dashed green box). Below the status bar, there's a search bar and a message 'Events processing complete. Reload output'. The output log shows the following tasks:

```

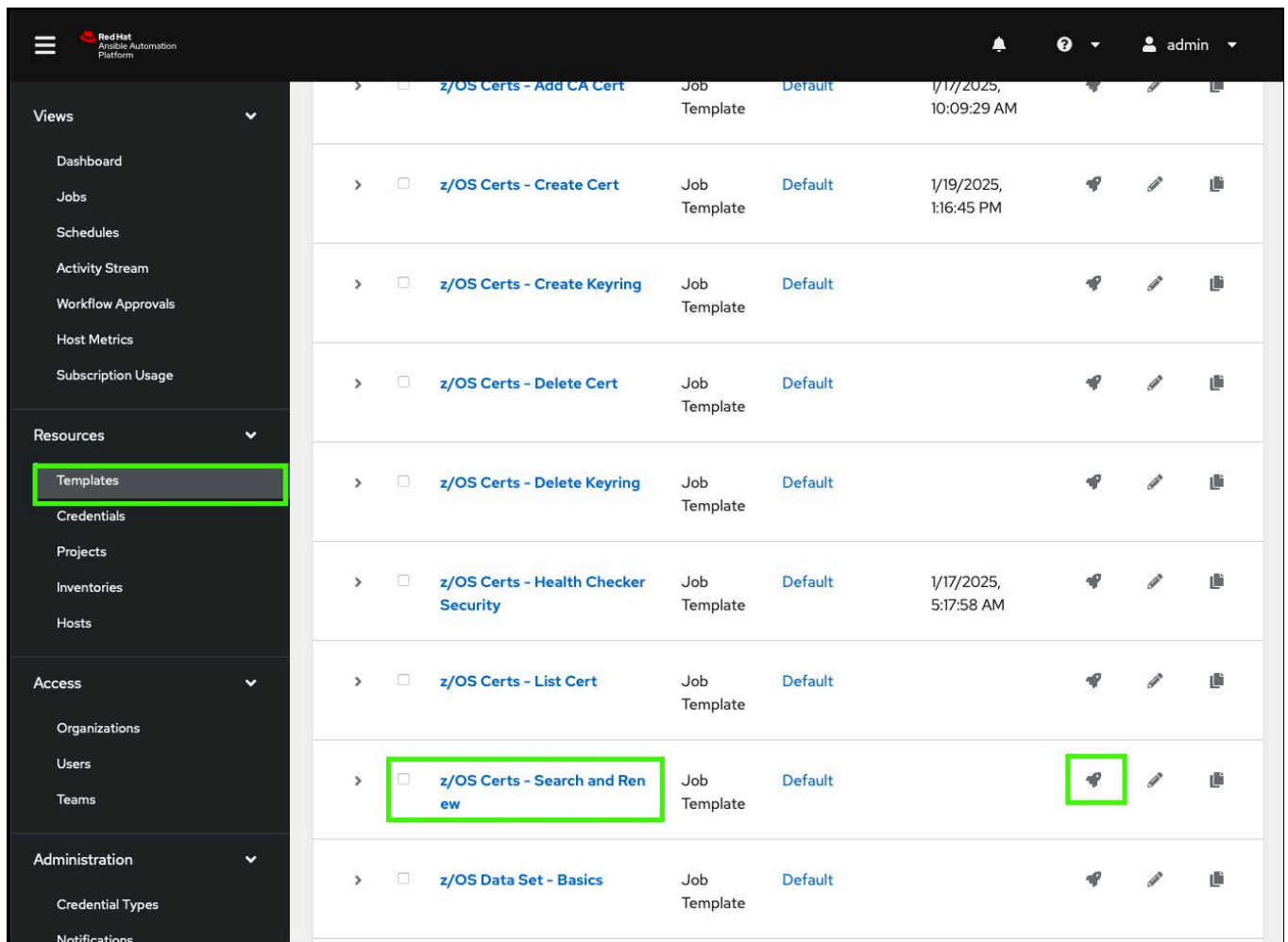
15 skipping: [zos_host]
16
17 TASK [Check if certificate to be created already exists.] ***** 13:16:28
18 ok: [zos_host]
19
20 TASK [Debug command output.] ***** 13:16:29
21 ok: [zos_host] => {
22   "msg": "4"
23 }
24
25 TASK [Debug.] ***** 13:16:29
26 ok: [zos_host] => {
27   "cert_not_found": true
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.



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!

Red Hat Ansible Automation Platform

Views

Dashboard

Jobs

Schedules

Activity Stream

Workflows

Hosts

Subscriptions

Resources

Templates

Credentials

Projects

Inventories

Hosts

Access

Organizations

Users

Teams

Administration

z/OS Certs - Add CA Cert

Job Template

Default

1/17/2025, 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 *

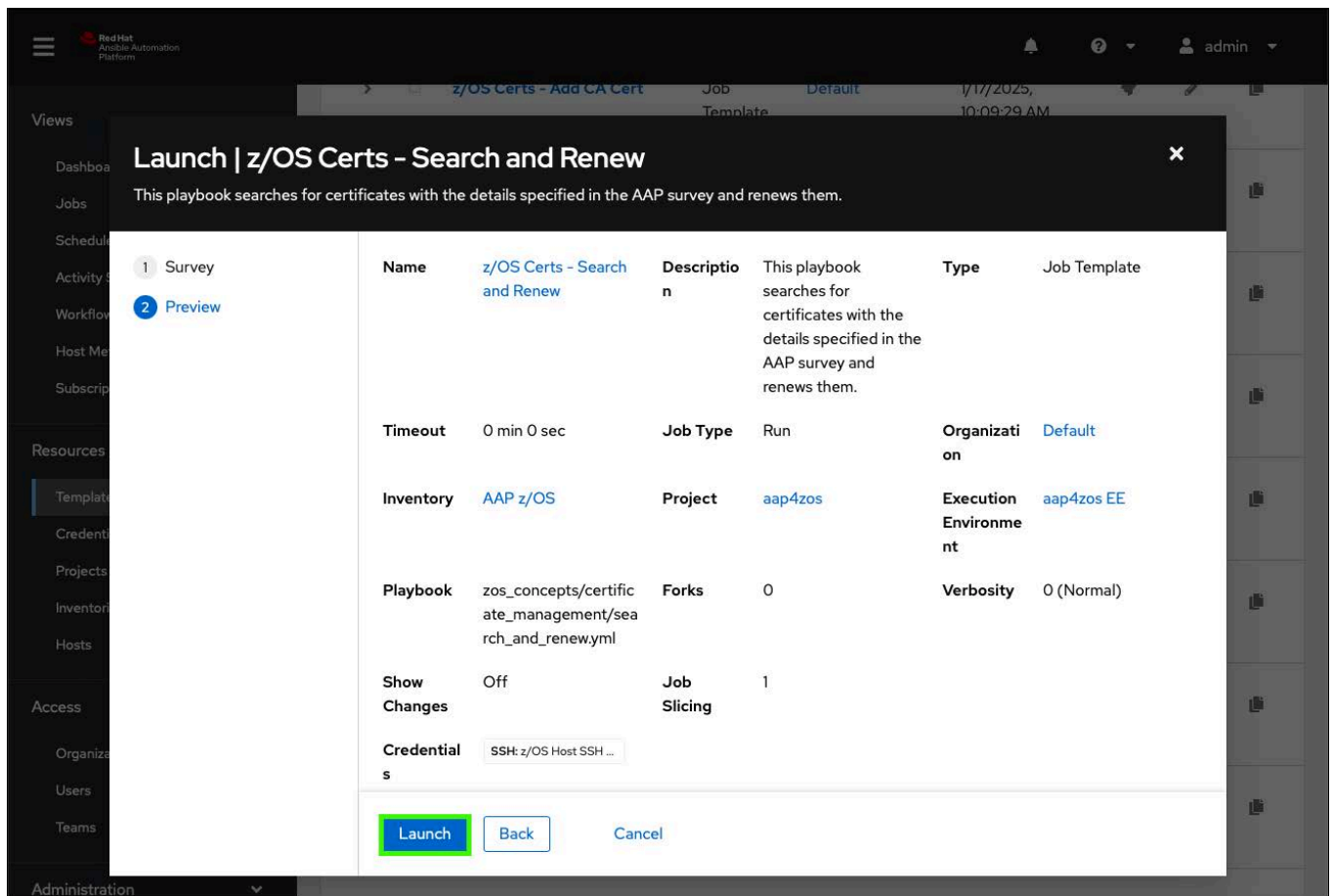
IBMUSER

Keyring *

SharedRing1

Next Back Cancel

3. Click **Launch**.



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

Jobs > 18 - z/OS Certs - Search and Renew

Output

z/OS Certs - Search and Renew

Successful Plays: 1 Tasks: 18 Hosts: 1 Elapsed: 00:00:32

Stdout

Events processing complete. Reload output

```

16 skipping: [zos_host] => (item={'ddname': 'JESJCL', 'record_count': '8', 'id': '3', 'stepname': 'JES2', 'proctep': None, 'byte_count': '136', 'content': ['      1 //HZSPRINT JOB 'RAC
F CERT EXPIRATION CHECK',                                JOB00225', '      //      MSGCLA
SS=H,MSGLEVEL=(1,1),CLASS=A,', '      //      NOTIFY=&SYSUID', '      IEF653I SU
BSTITUTION JCL - 'RACF CERT EXPIRATION CHECK',MSGCLASS=H,MSGLEVEL=(1,1),CLASS=A,NOTIFY=IBMUS
ER', '      2 //HZSPRINT EXEC PGM=HZSPRINT,TIME=100,REGION=0M,PARMDD=SYSIN', '      3 //S
YSIN 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 AC
CESS AT 15:26:46 ON SUNDAY, JANUARY 19, 2025', ' IEF4111I HZSPRINT IS USING THE FOLLOWING JO
B RELATED SETTINGS:', '      SWA=BELOW,TIOT SIZE=64K,DSENQSHR=DISALLOW,GDGBIAS=JOB', ' I
EF236I ALLOC. FOR HZSPRINT HZSPRINT', ' IEF237I JES2 ALLOCATED TO SYSIN', ' IEF237I JES2 ALL
OCATED TO SYSOUT', ' IEF142I HZSPRINT HZSPRINT - STEP WAS EXECUTED - COND CODE 0000', ' IEF2
85I  IBMUSER.HZSPRINT.JOB00225.D0000101.?      SYSIN', ' IEF285I  IBMUSER.HZSPRINT.JOB0
0225.D0000102.?      SYSOUT', ' IEF373I STEP/HZSPRINT/START 2025019.1526', ' IEF032I STE
P/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- ALLO
C:    13M SHRD:    0M'. ' IEF375I  JOB/HZSPRINT/START 2025019.1526'. ' IEF033I  JOB/HZ

```

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.

Templates

Name Type Organization Last Ran Actions

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!

Red Hat Ansible Automation Platform

Views

Launch | z/OS Certs - Create Cert

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

1 Survey

2 Preview

Organizational Unit ⓘ

ANSIBLE CORE

Common Name ⓘ

company.com

Country ⓘ

US

Keyring ⓘ

SharedRing1

Expiration Date ⓘ

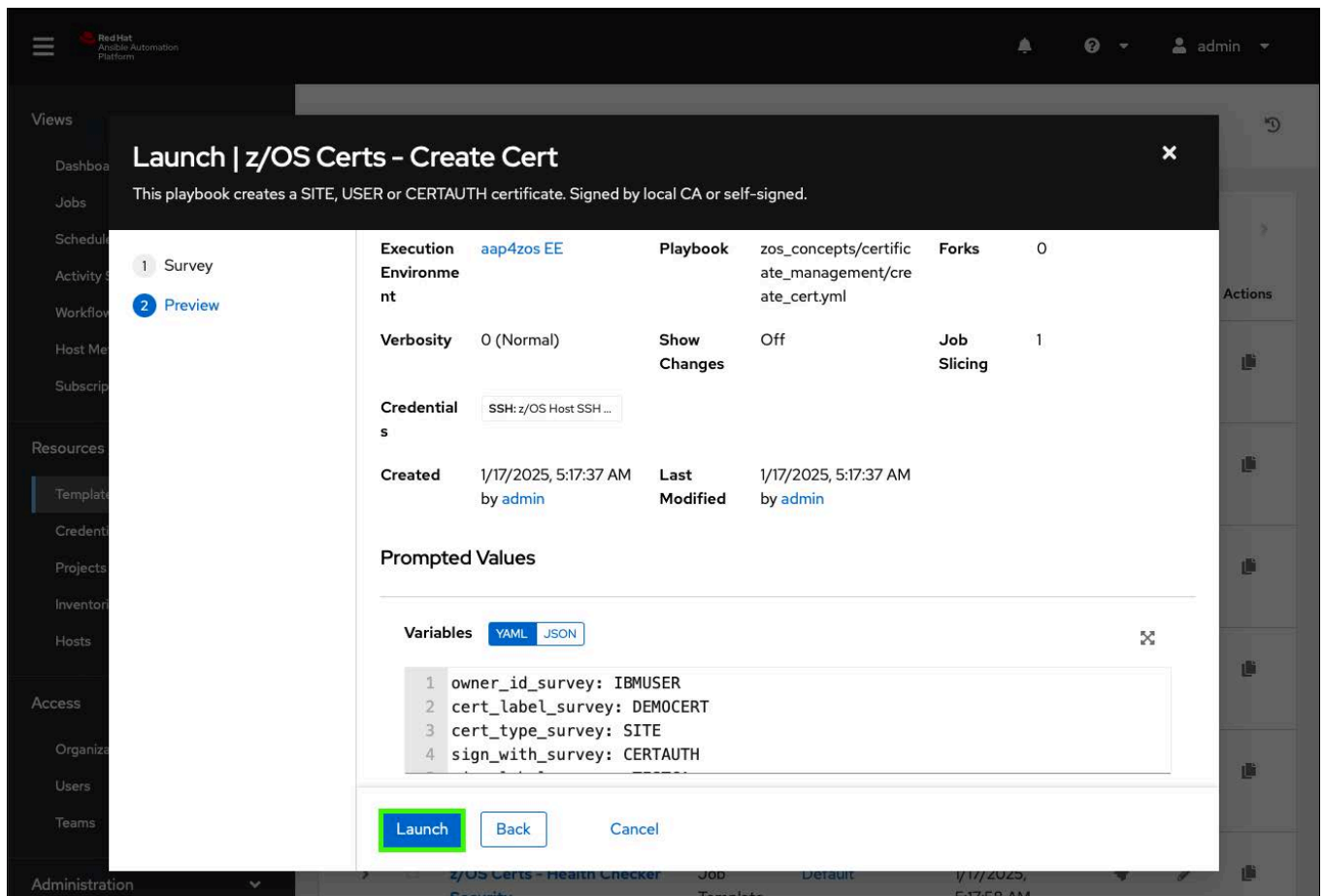
2030-01-30

Check Name ⓘ

IBMRACF,RACF_CERTIFICATE_EXPIRATION

Next Back Cancel

3. Click **Launch**.



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

The screenshot displays the Red Hat Ansible Automation Platform (AAP) interface. On the left is a sidebar with navigation menus: 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 panel shows the 'Output' tab for a job titled 'z/OS Certs - Create Cert'. The job status is 'Successful'. Below the status bar, there's a search bar and a 'Stdout' dropdown. A notification bar indicates 'Events processing complete' with a 'Reload output' link. The output content shows a series of tasks: skipping [zos_host], TASK [Check if certificate to be created already exists.], TASK [Debug command output.], TASK [Debug.], TASK [Generate new certificate.], and TASK [List chain for new certificate.]. Each task is followed by its status and a timestamp.

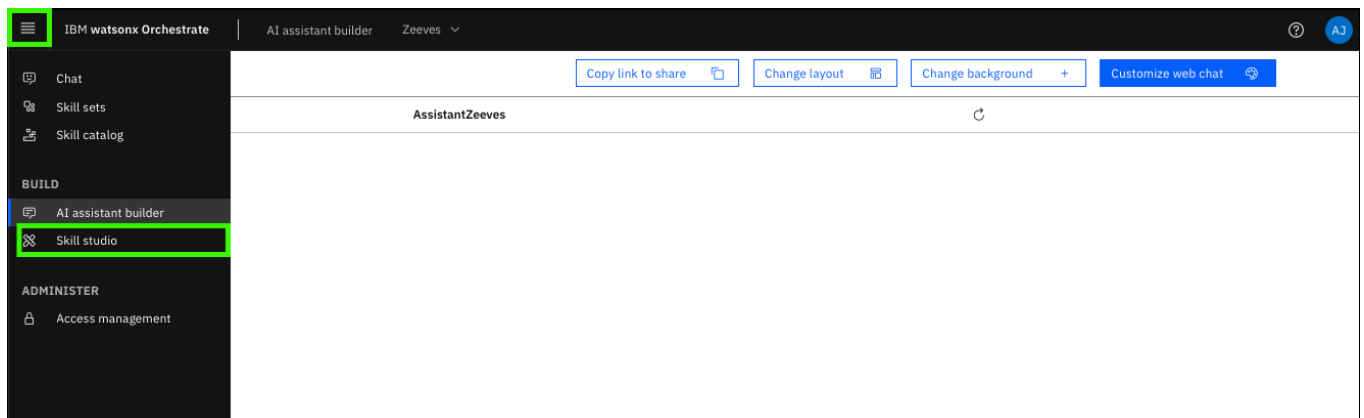
Import the Ansible automations into watsonx Orchestrate

For this use case, configure the assistant to guide the user through the process of identifying their SSL certificate's expiration date and automate the certificate renewal process. To do so, import the needed AAP templates into watsonx Orchestrate as skills.

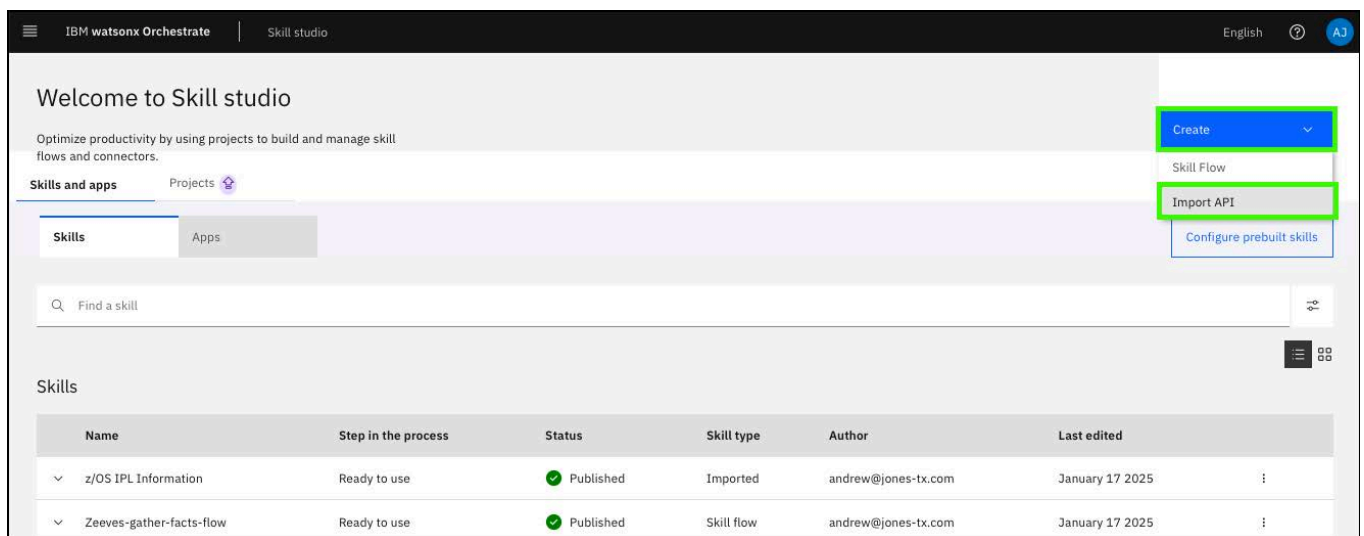
For this use case, the ansible templates you import are:

- z/OS Certs – List Cert
- z/OS Certs – Search and Renew
- Retrieve job output (utility skill)

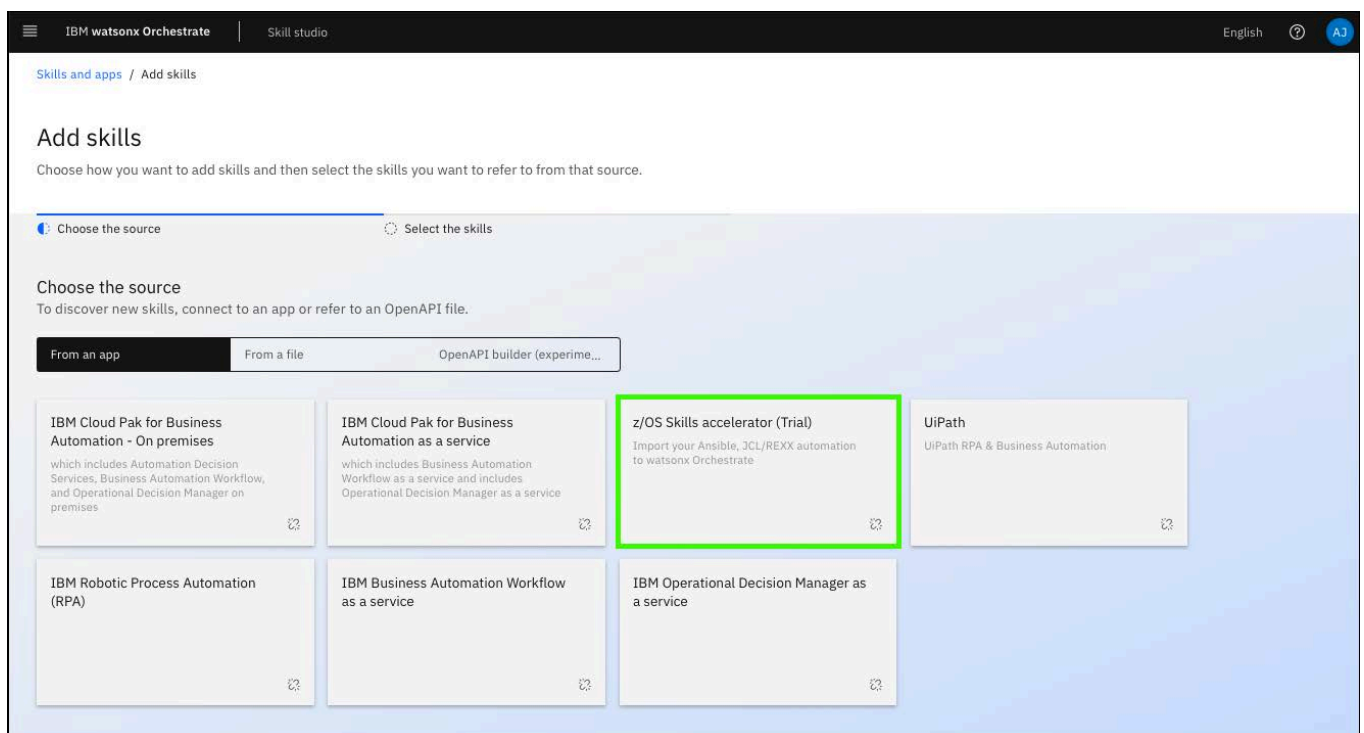
1. Open **Skill studio** in watsonx Orchestrate.



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



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



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

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

a: Connection Type:

ansible

b: Application Name:

certs

c: Connection URL:

<Enter the URL for your AAP UI>

d: User Name:

<Enter the AAP User Name (for UI access)>

e: Password:

<Enter the AAP User Password>

f: Search Pattern:

*

IBM watsonx Orchestrate | Skill studio

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

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

From an app | From a file | OpenAPI builder (experime...)

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

IBM Robotic Process Automation (RPA)

IBM Business Automation Workflow as a service

IBM Operational Decision Manager as a service

Connect to find skills from z/OS Skills accelerator (Trial)

Import your Ansible, JCL/REXX automation to watsonx Orchestrate

Connection Type
ansible

Either 'ansible' or 'zosmf'

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.

Connection URL
https://itzvsi-aap-pwgabob.techzone.ibm.com

For example https://example.com

User name
admin

Password

Search Pattern
*

Necessary for 'zosmf' connections. Pattern cannot be '*' or '!'

Cancel | **Connect**

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

IBM watsonx Orchestrate | Skill studio

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**

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.

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, 2 of 20 skills are selected.

Ansible Job Template Projects / aap4zos

Skill	Description	Status
☐ Z/os certs - delete cert	Certs - this playbook deletes ...	Ready to add
☒ Z/os certs - list cert	Certs - this playbook displays...	Ready to add
☒ Z/os certs - search and renew	Certs - this playbook searches...	Ready to add
☐ Z/os tso command(s)	Certs - a playbook to execute ...	Ready to add
☐ Z/os operator command	Certs - a playbook to execute ...	Ready to add

Items per page: 5 16-20 of 20 items 4 of 4 pages

Cancel **Save as draft**

7. Click the ellipses (⋮) for the **z/OS Certs - List Cert** skill and select **Enhance this skill**.

Welcome to Skill studio

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

Skills and apps Projects

Skills Apps

Find a skill

Skills

Name	Step in the process	Status	Skill type	Author	Last edited	
z/OS Certs - Search and Renew	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	January 19 2025	⋮
z/OS Certs - List Cert	Just 1 step away to be ready	Ready to publish	Imported	andrew@jones-tx.com	January 19 2025	⋮
z/OS IPL Information	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025	Enhance this skill
Zeeves-gather-facts-flow	Ready to use	Published	Skill flow	andrew@jones-tx.com	January 17 2025	Export this skill
Retrieve job output	Ready to use	Published	Imported	andrew@jones-tx.com	January 17 2025	Delete this skill
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	⋮

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

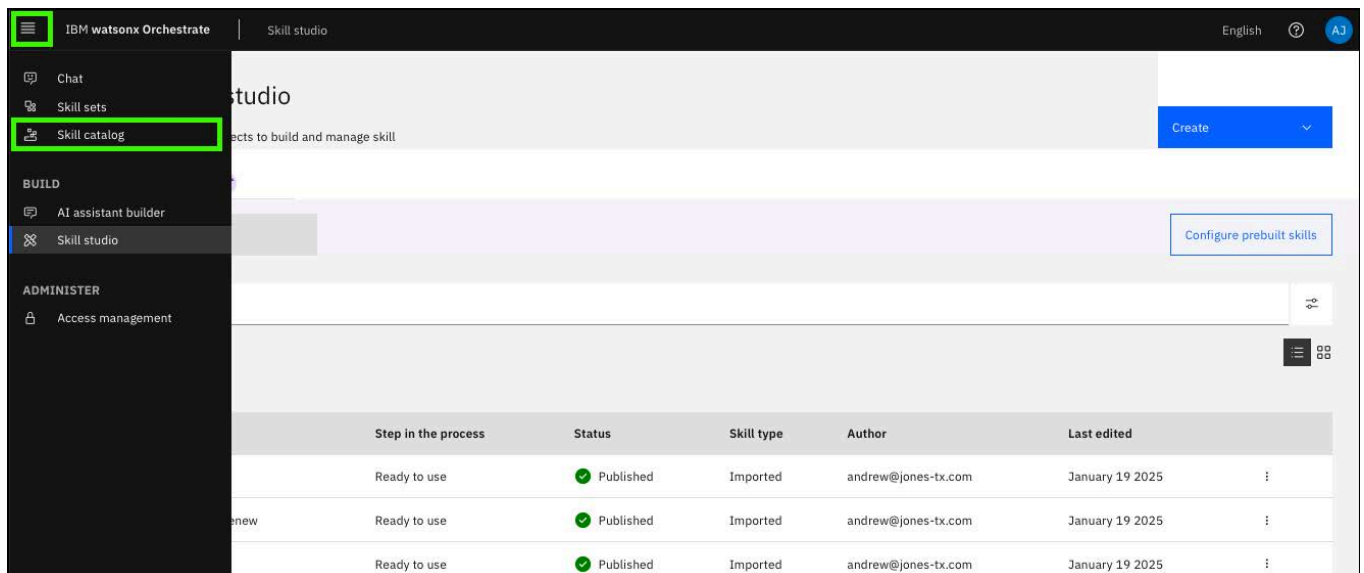
The screenshot shows the 'Skill studio' interface in IBM watsonx Orchestrate. The main heading is 'Enhance the “z/OS Certs - List Cert” skill' with a subtext 'Add details that will make people want to use this skill.' Below this is a tabbed interface with 'Name' selected. The 'Name' tab contains fields for 'Name*' (filled with 'z/OS Certs - List Cert'), 'Description' (filled with 'certs - This playbook displays a certificate's details.'), 'API version*' (filled with '1.0'), 'Categories' (with an 'Add categories' button), and 'App' (filled with 'Ansible Controller Skills - certs'). To the right, a 'Preview' section shows two visual representations of the skill: one for the catalog and one for the skill set. At the bottom right, there are three buttons: 'Cancel', 'Publish' (highlighted with a green border), and 'Save as draft'.

9. Repeat steps 7 and 8 for the **z/OS Certs - Search and Renew** skill.

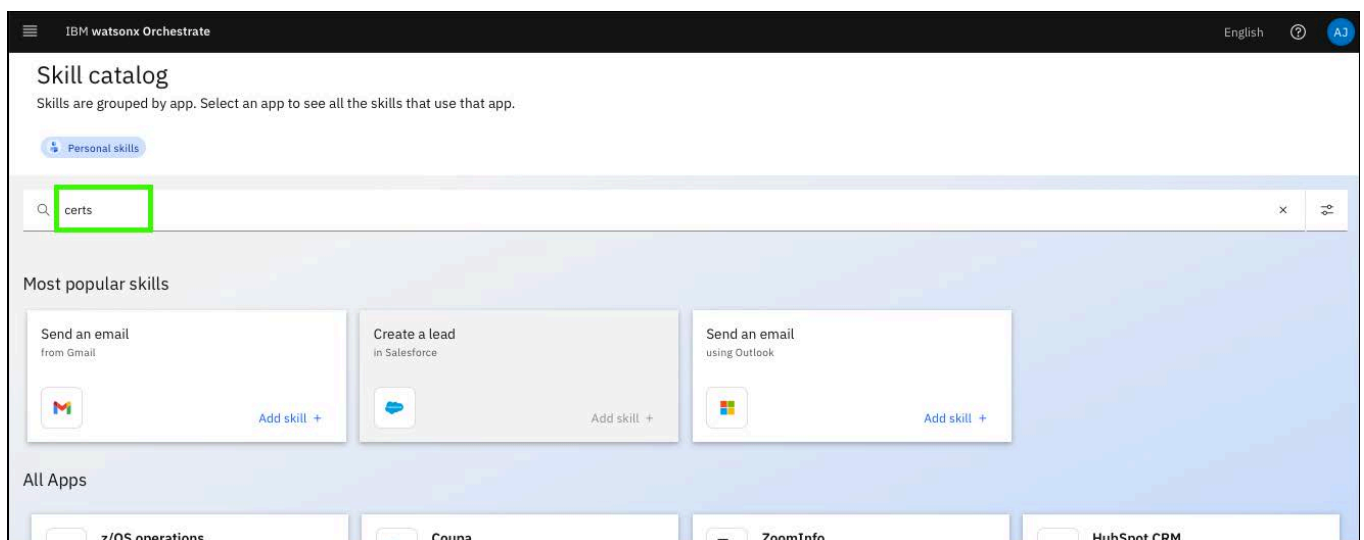
10. **Challenge:** You also need to add the **Retrieve job output** utility to your **certs** app just like you did when creating the **Gather Facts** skill flow. Repeat steps 2 - 8 to add the **Retrieve job output** utility skill to your **certs** app.

Verify all the skills are successfully imported and create the app connection.

1. Open **Skill catalog** in watsonx Orchestrate.



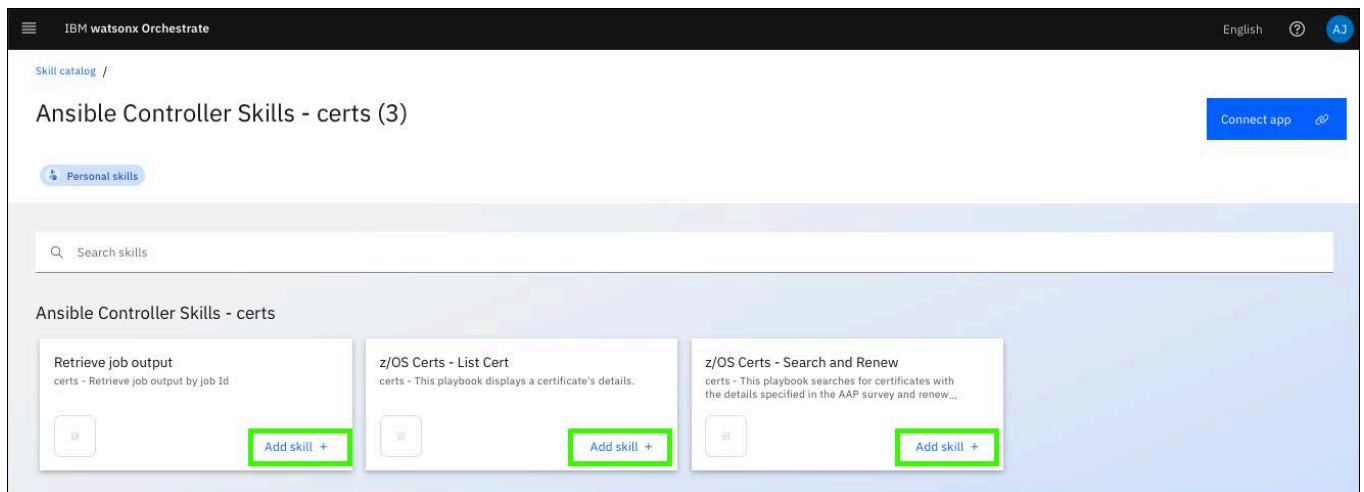
2. Enter **certs** in the search bar.



3. Click the **certs** tile.



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



IBM watsonx Orchestrate

English ? A3

Skill catalog /

Ansible Controller Skills - certs (3)

Personal skills

Search skills

Ansible Controller Skills - certs

Retrieve job output
certs - Retrieve job output by job Id

Add skill +

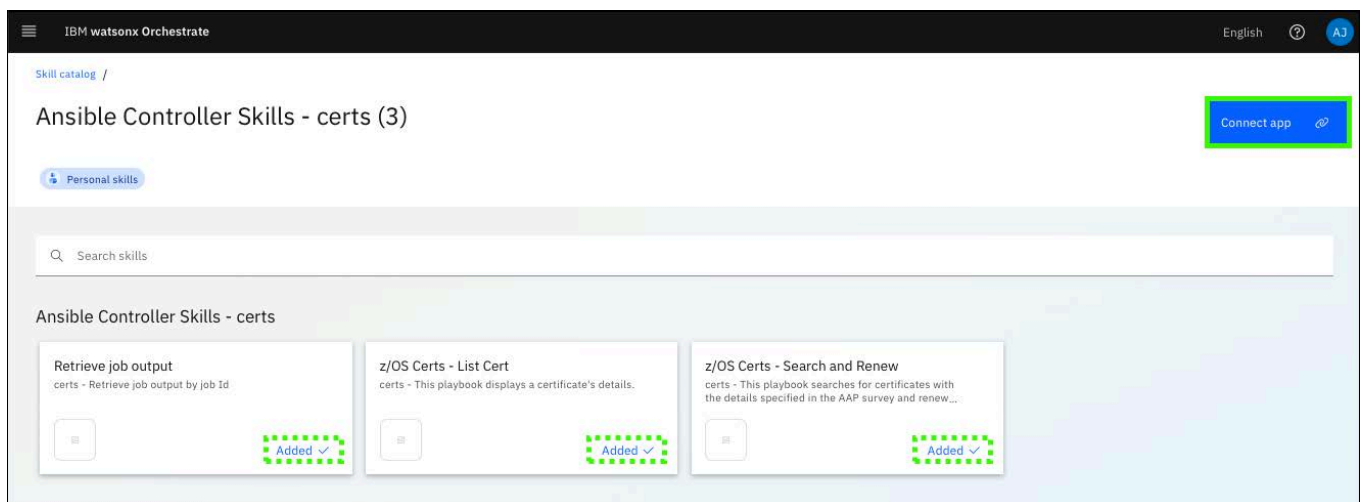
z/OS Certs - List Cert
certs - This playbook displays a certificate's details.

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**.



IBM watsonx Orchestrate

English ? A3

Skill catalog /

Ansible Controller Skills - certs (3)

Personal skills

Search skills

Ansible Controller Skills - certs

Retrieve job output
certs - Retrieve job output by job Id

Added ✓

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

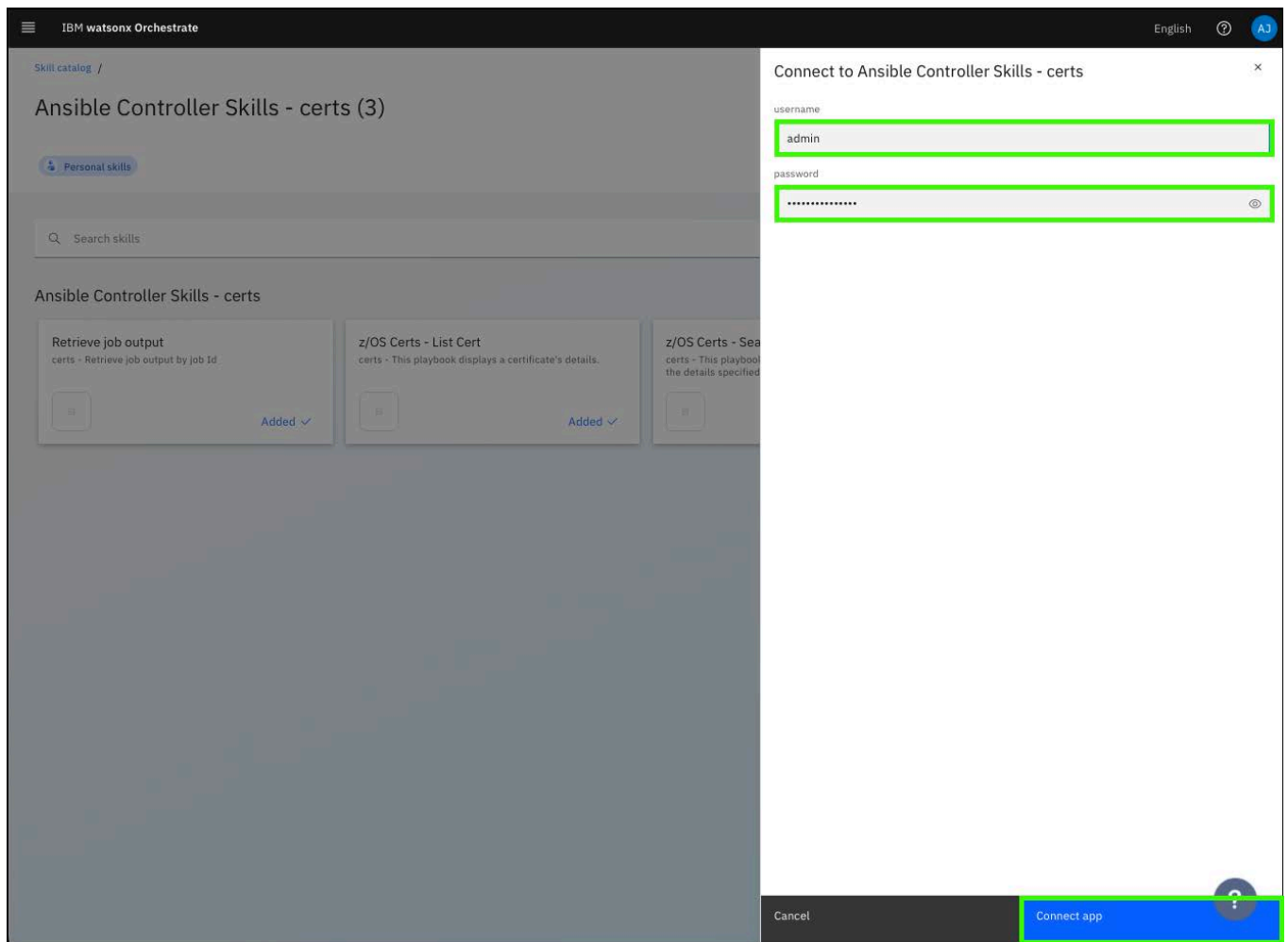
Added ✓

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

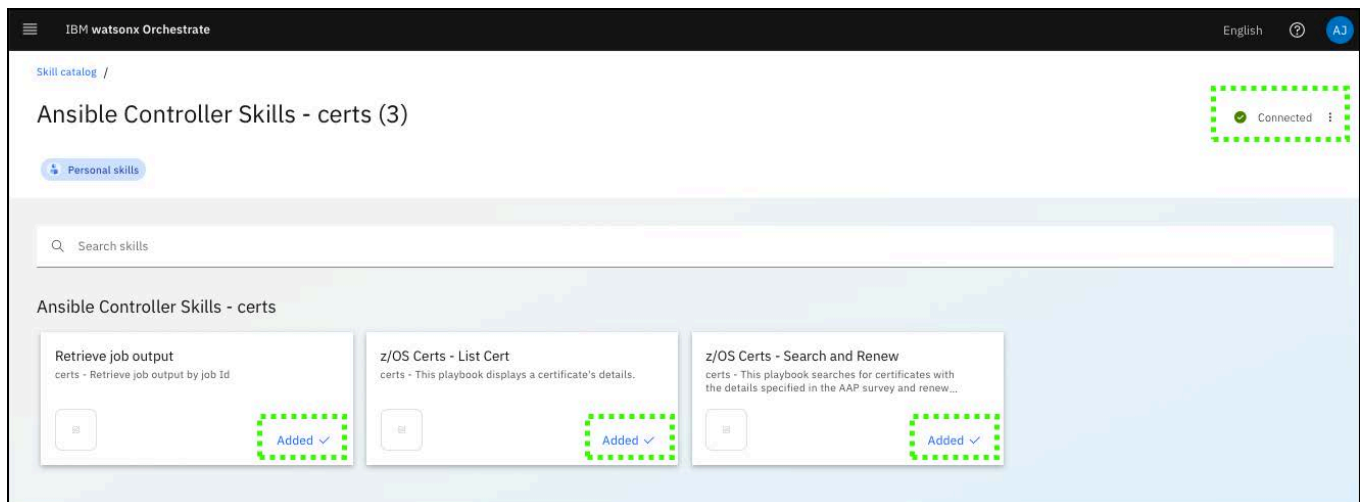
Added ✓

Connect app

6. Enter your **AAP Username** and **AAP Password** and then click **Connect app**.

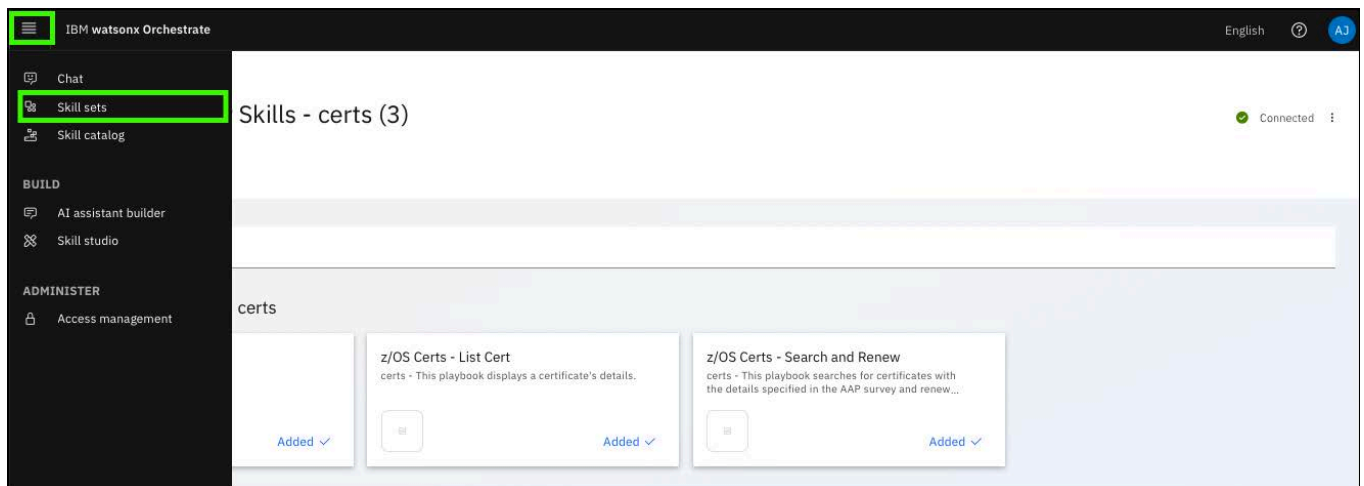


7. Verify that the app is connected.

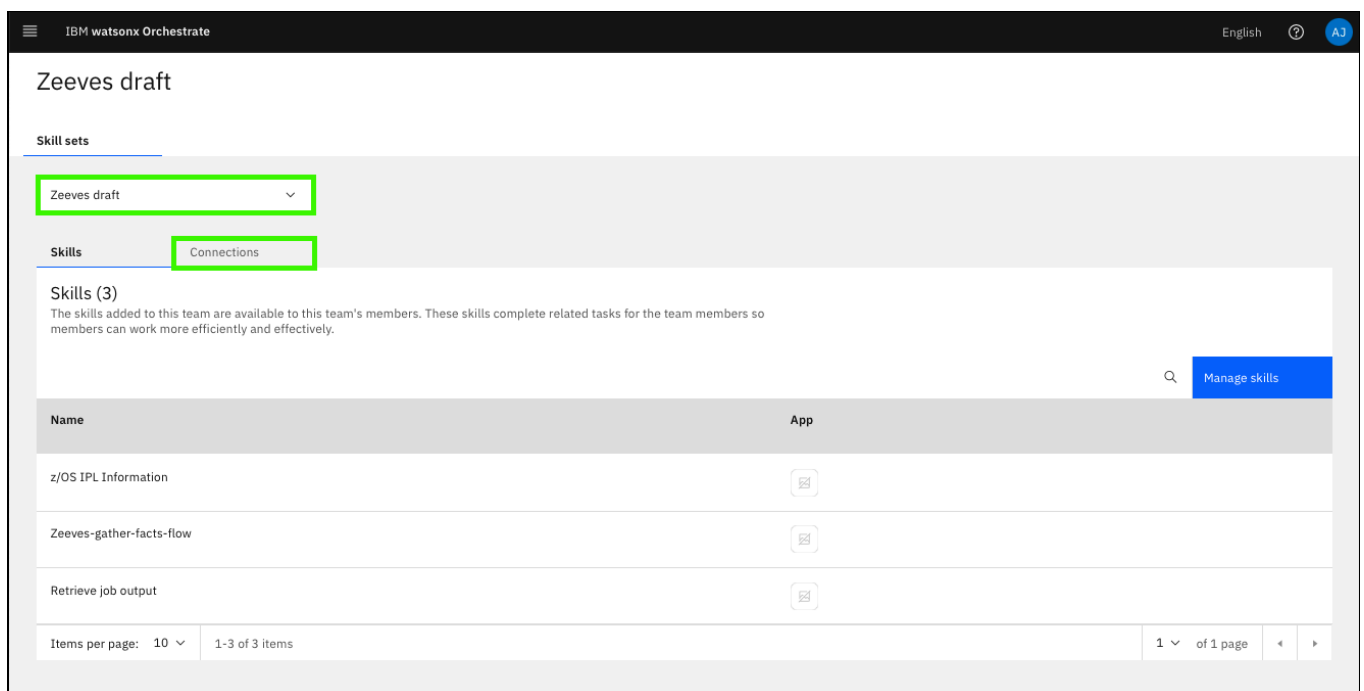


Connect the app to the assistant.

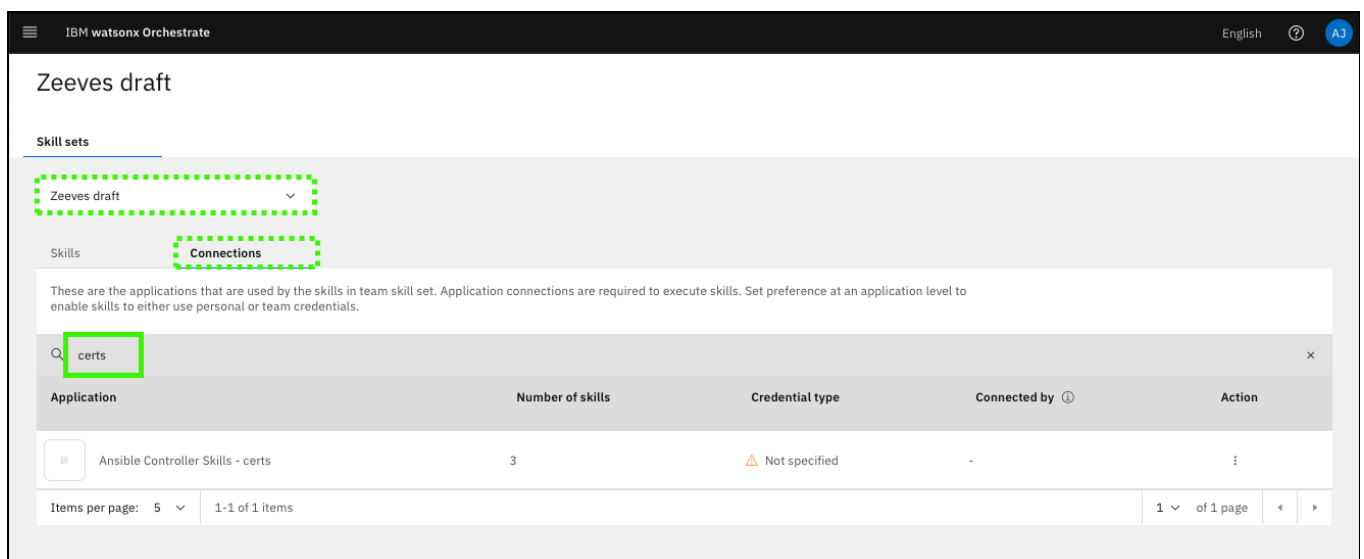
1. Open **Skill catalog** in watsonx Orchestrate.



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



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



4. Click the ellipses (⋮) for the *certs app* and select *Connect app**.

IBM watsonx Orchestrate

English ? A3

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.

Q 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

Connect app

5. Click **Connect app**.

IBM watsonx Orchestrate

English ? A3

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.

Q certs

Application	Number of skills	Credential type
Ansible Controller Skills - certs	3	Not specified

Items per page: 5 1-1 of 1 items

Connect to Ansible Controller Skills - certs

☐ Member credentials

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

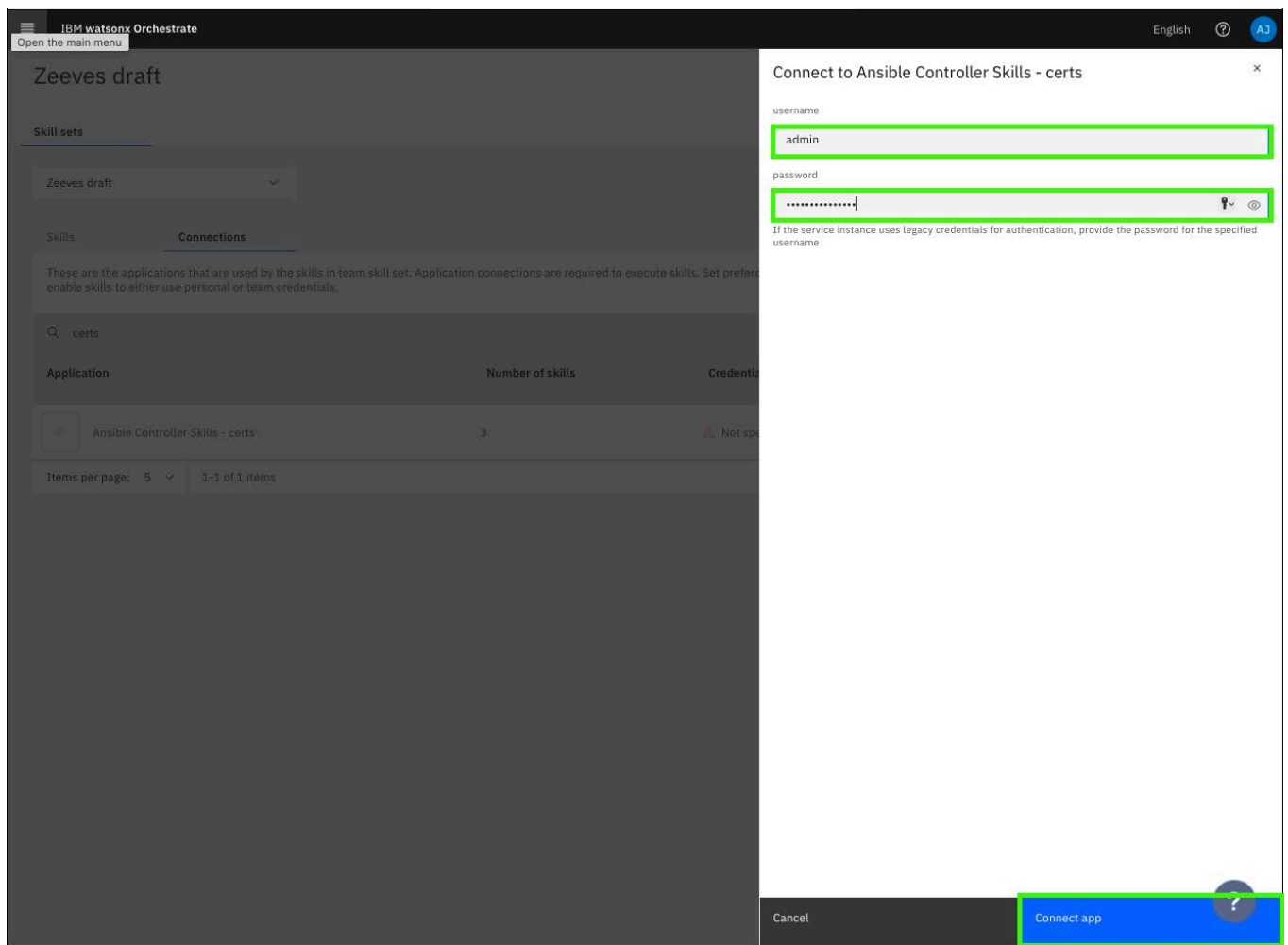
☒ Team credentials

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**.

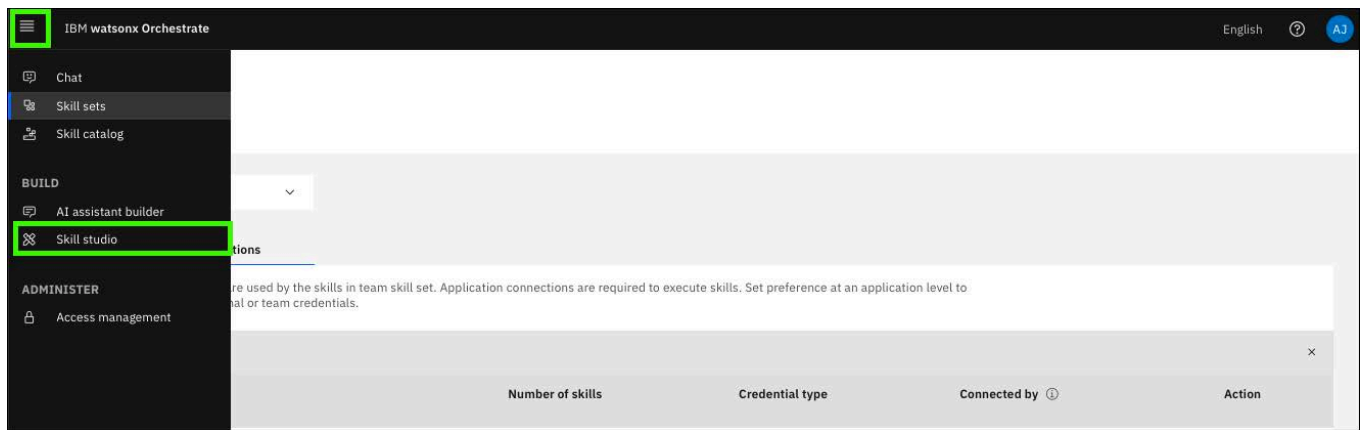


Create a skill flow to retrieve certificate expiration dates.

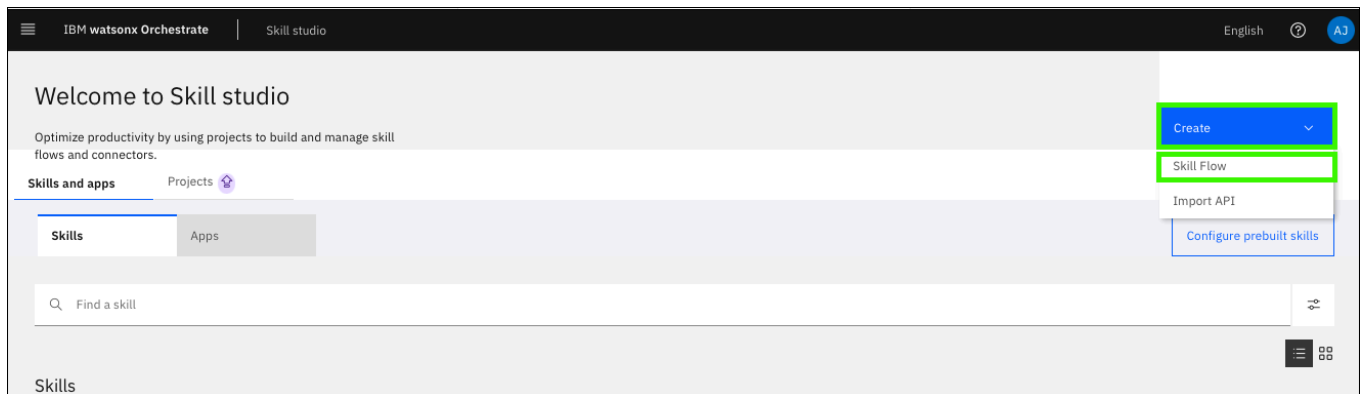
The goal of this scenario is to configure the assistant to automate the certificate renewal process for the client. The first step in that process is to help the SA identify the expiration date of their z/OS certificate. You have imported the **z/OS Certs – List Cert** skill from Ansible Automation Platform. 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.



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



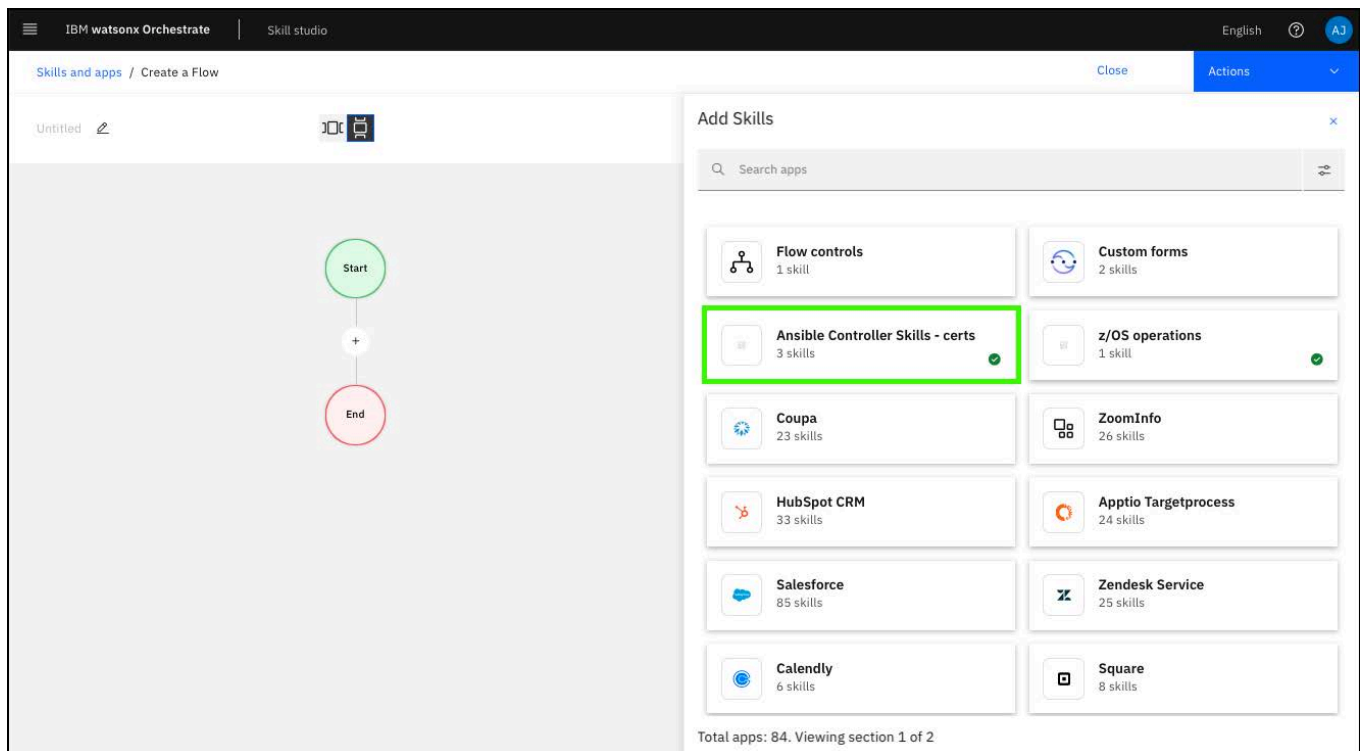
3. Click the **+** icon.



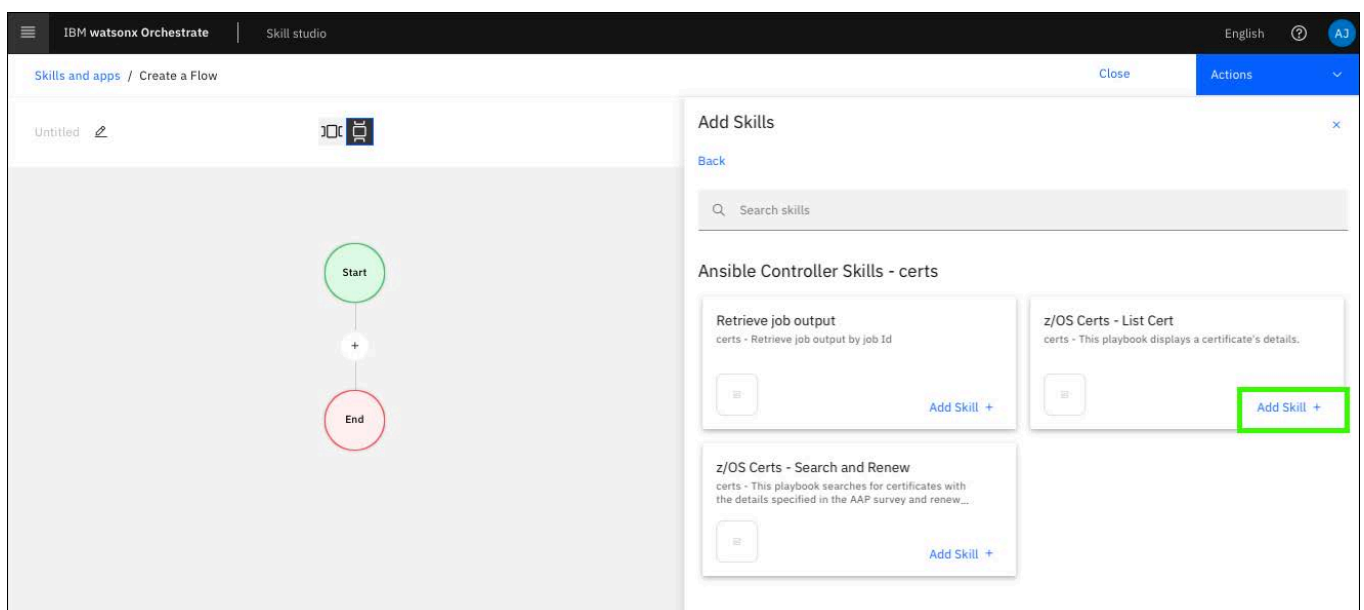
4. Click the **certs** app.



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

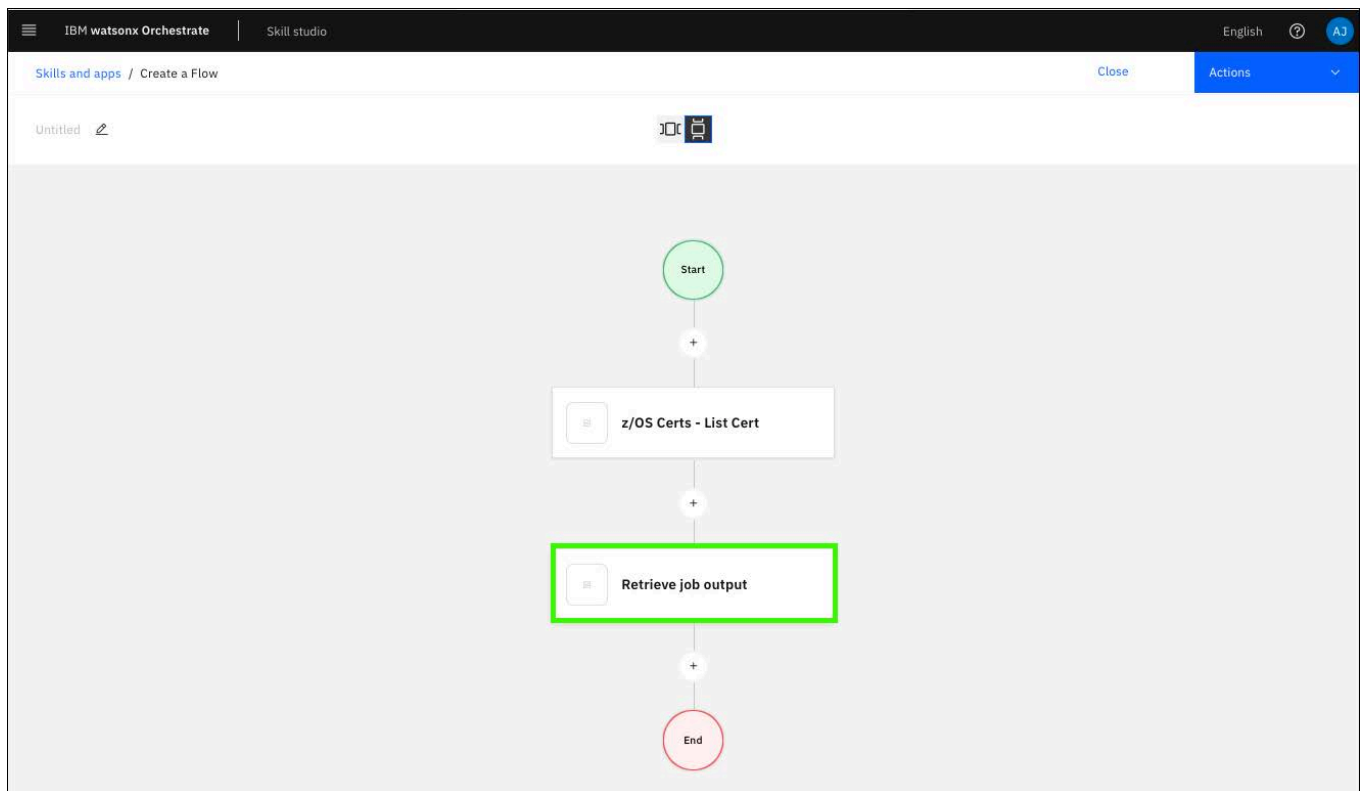


5. Click **Add Skill +** in the **z/OS Certs - List Cert** tile.

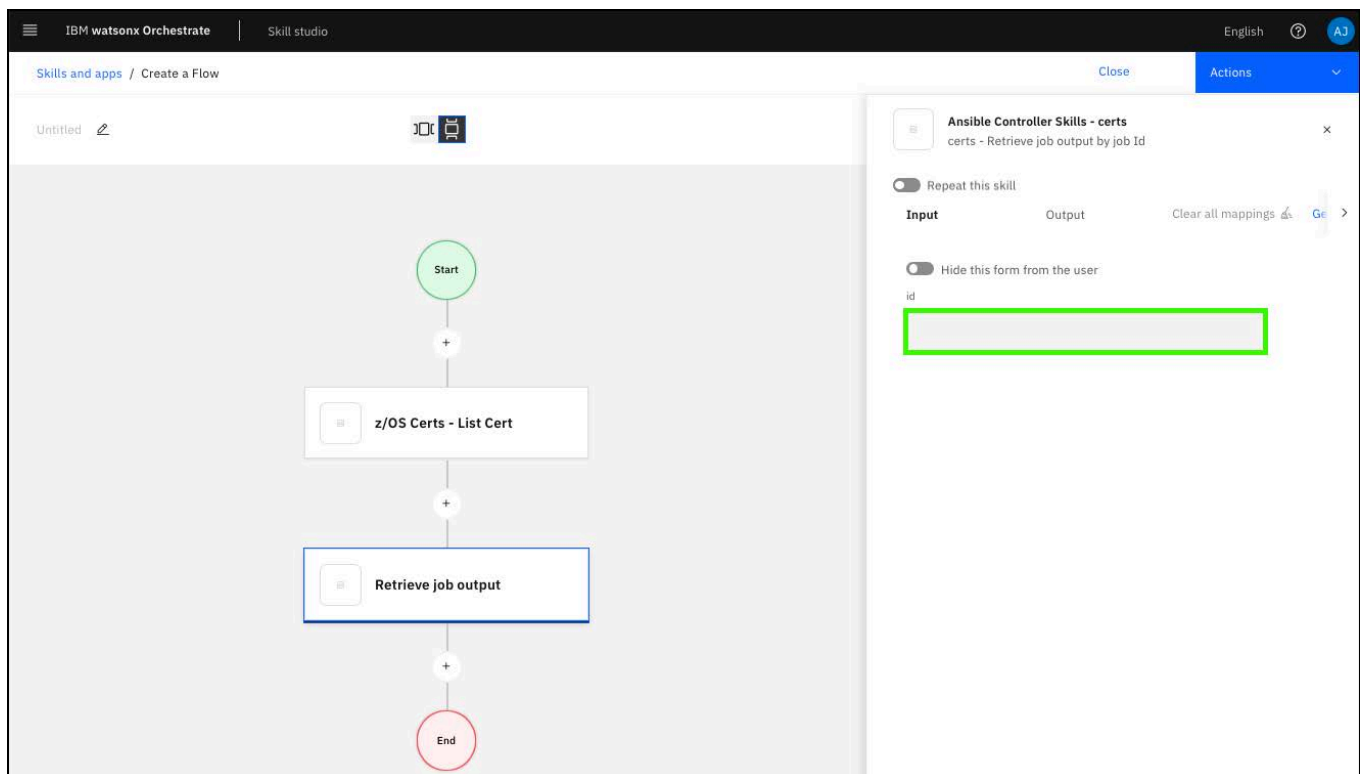


6. Click the **+** icon **below** the **z/OS Certs - List Cert** skill and repeat steps 4 and 5 to add the **Retrieve job output** skill.

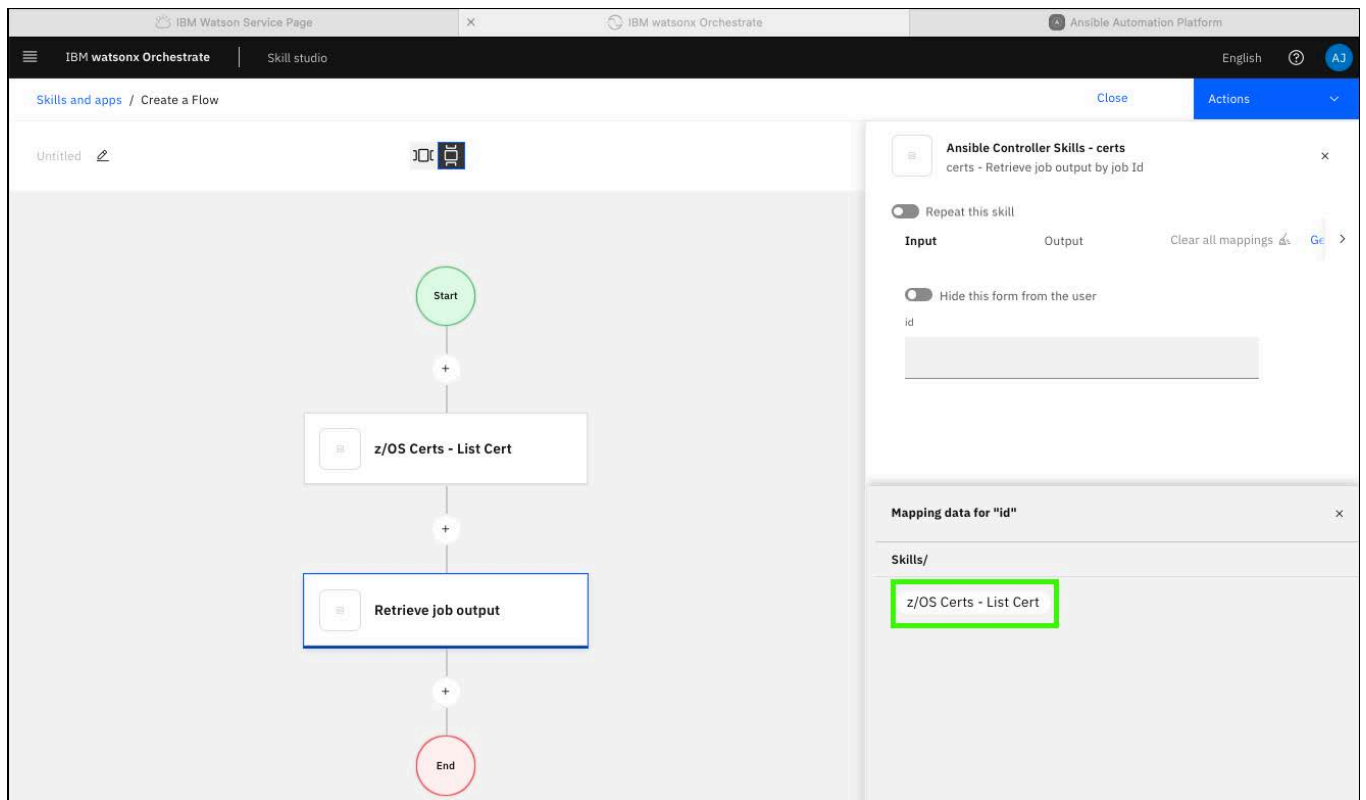
7. Click the **Retrieve job output** skill.



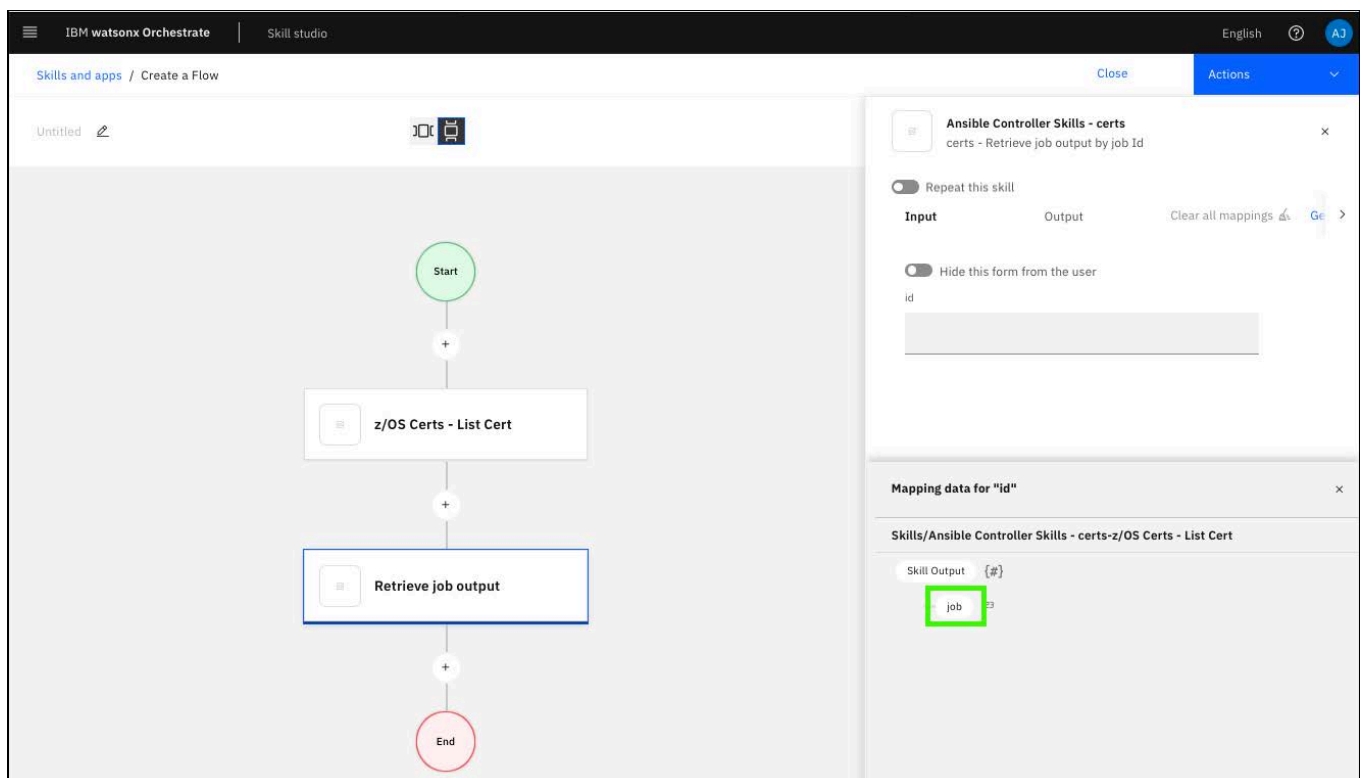
8. Click the **id** field.



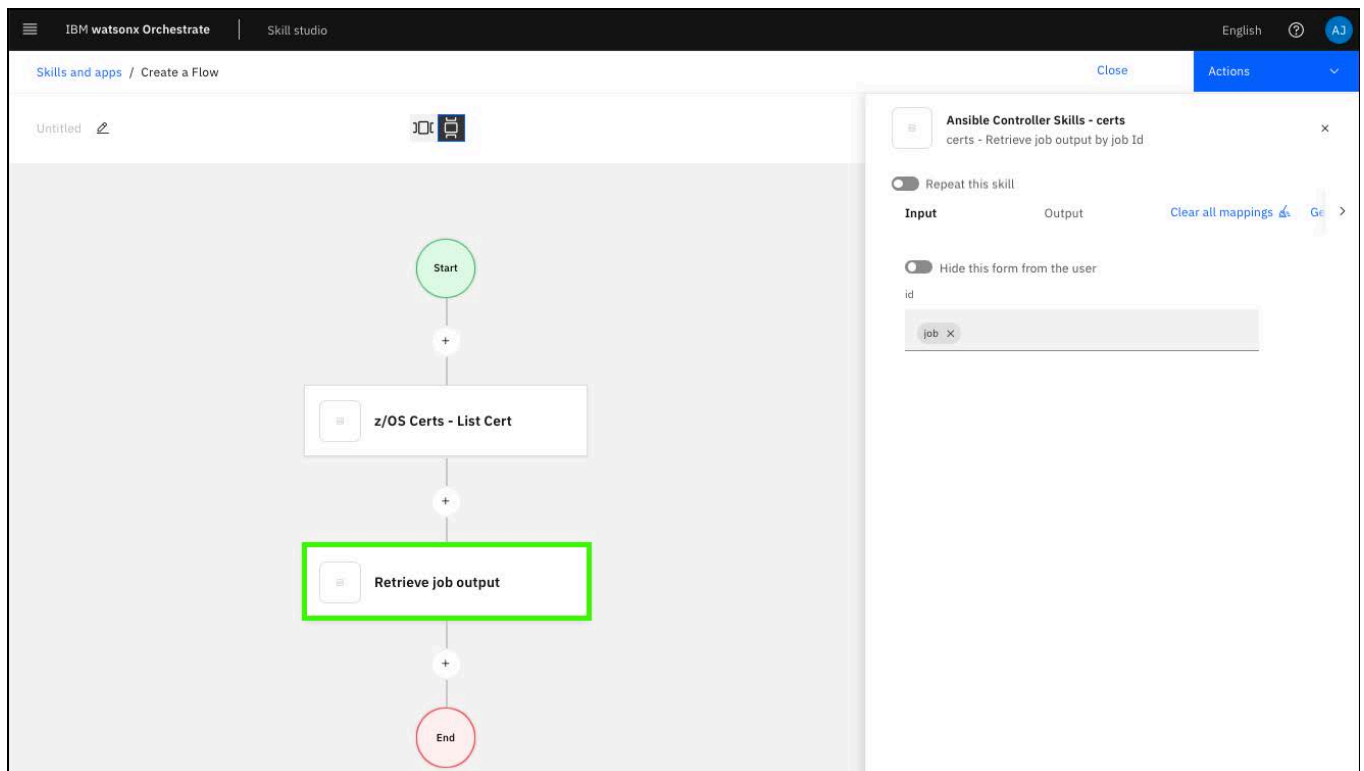
9. Click **z/OS certs - List Cert**.



10. Click **job**.

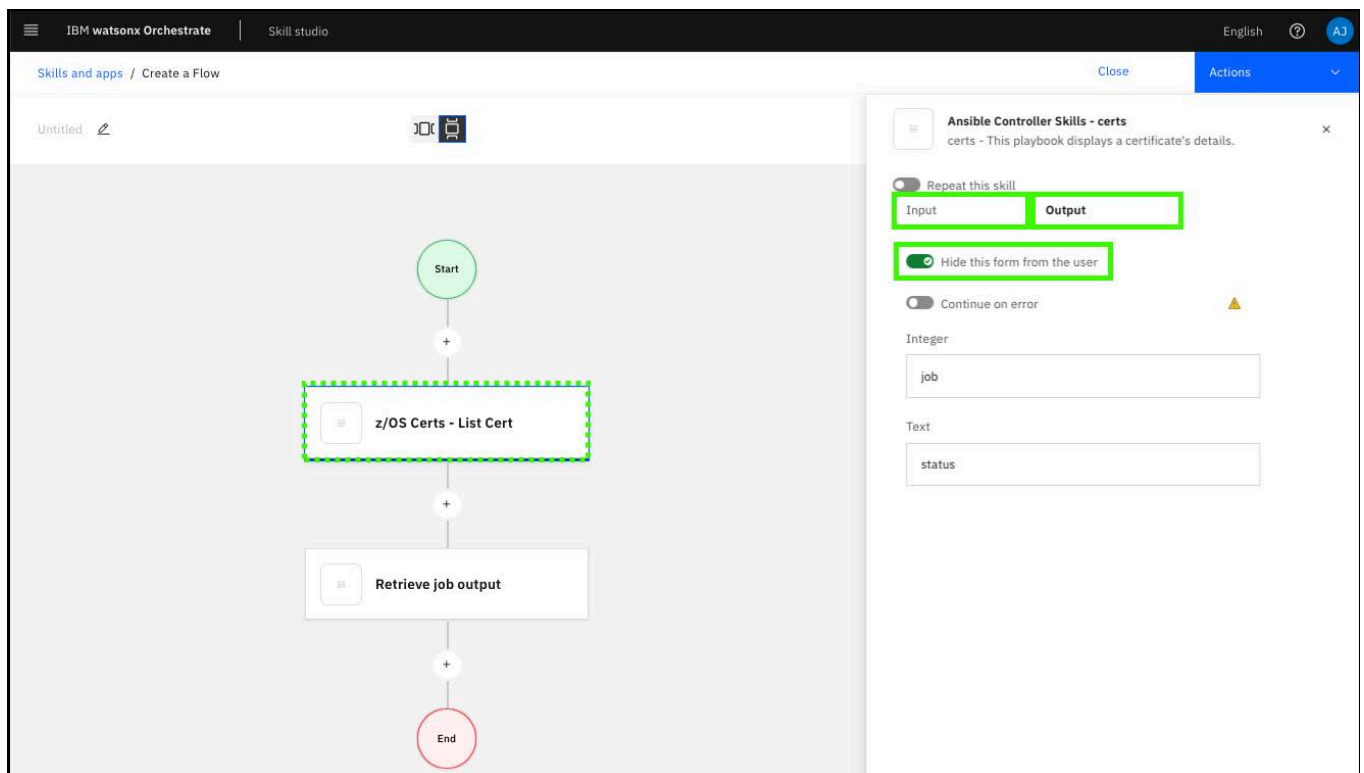


11. Click the **z/OS Certs - List Cert** tile.



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 screenshot shows the IBM watsonx Orchestrate Skill studio interface. On the left, a flow diagram is displayed with a vertical sequence of steps: a green 'Start' circle, a '+' connector, a white box labeled 'z/OS Certs - List Cert', another '+' connector, a white box labeled 'Retrieve job output' (highlighted with a green border), a third '+' connector, and a red 'End' circle. On the right, the 'Actions' panel for the 'Retrieve job output' skill is open. It shows the skill name 'Ansible Controller Skills - certs' and its description 'certs - Retrieve job output by job Id'. The 'Repeat this skill' toggle is off. The 'Input' field is empty, and the 'Output' field is highlighted with a green border. The 'Hide this form from the user' toggle is on. The 'Continue on error' toggle is off. The 'Text' field contains the word 'content'.

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.

The screenshot shows the same IBM watsonx Orchestrate Skill studio interface as before. The flow diagram is identical, but the '+' connector below the 'Retrieve job output' skill is now highlighted with a green border, indicating it is the target for the next action.

15. Click the **Custom forms**.

The screenshot shows the IBM watsonx Orchestrate Skill studio interface. On the left, a flow diagram is visible with a 'Start' node, a 'z/OS Certs - List Cert' skill, a 'Retrieve job output' skill, and an 'End' node. On the right, the 'Add Skills' panel is open, displaying a grid of available skills. The 'Custom forms' skill is highlighted with a green border. Below the grid, it indicates 'Total apps: 84. Viewing section 1 of 2'.

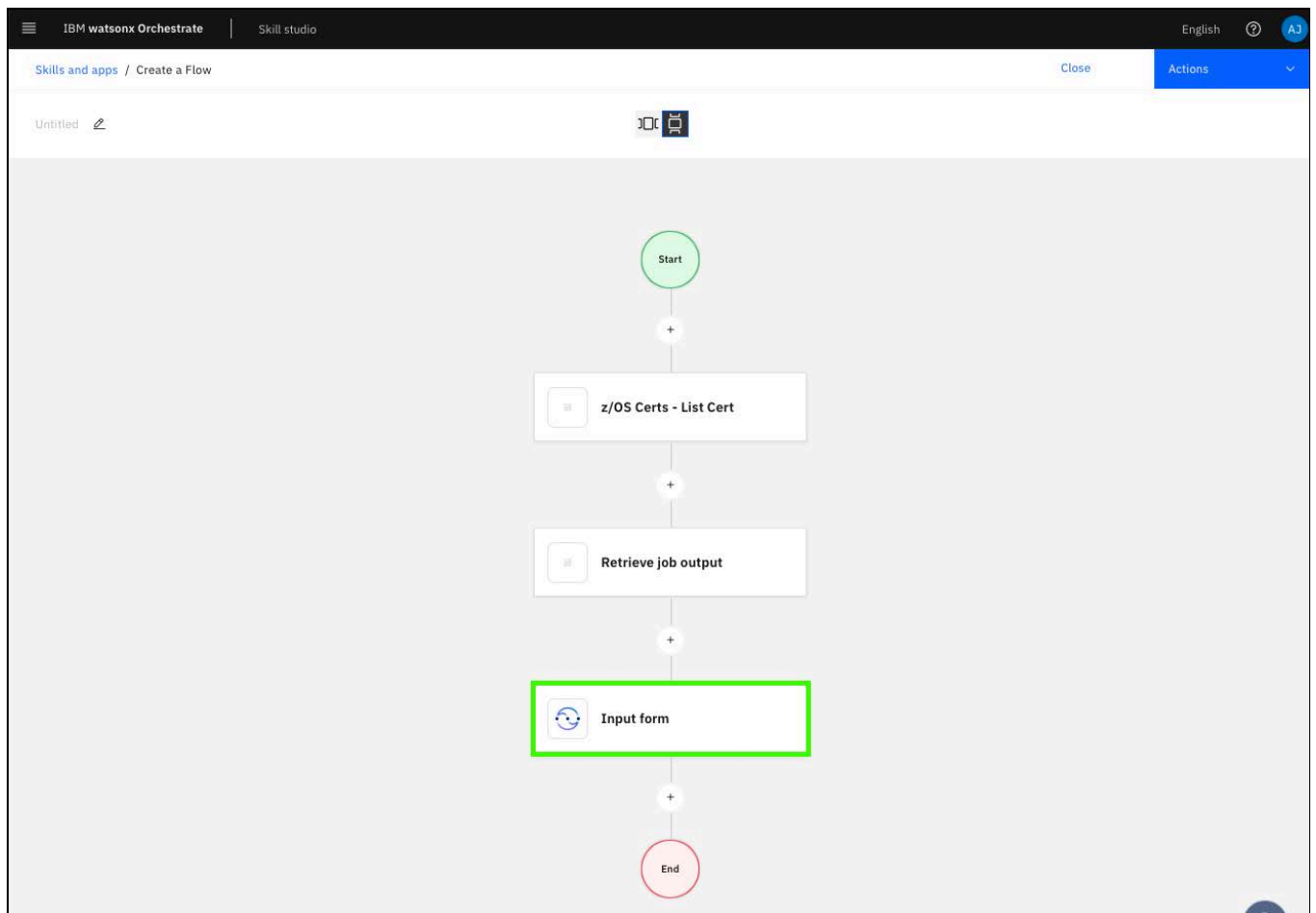
Skill Name	Number of Skills
Flow controls	1 skill
Custom forms	2 skills
Ansible Controller Skills - certs	3 skills
z/OS operations	1 skill
Coupa	23 skills
ZoomInfo	26 skills
HubSpot CRM	33 skills
Apptio Targetprocess	24 skills
Salesforce	85 skills
Zendesk Service	25 skills
Calendly	6 skills
Square	8 skills

16. Click **Add skill +** in the **Input form**.

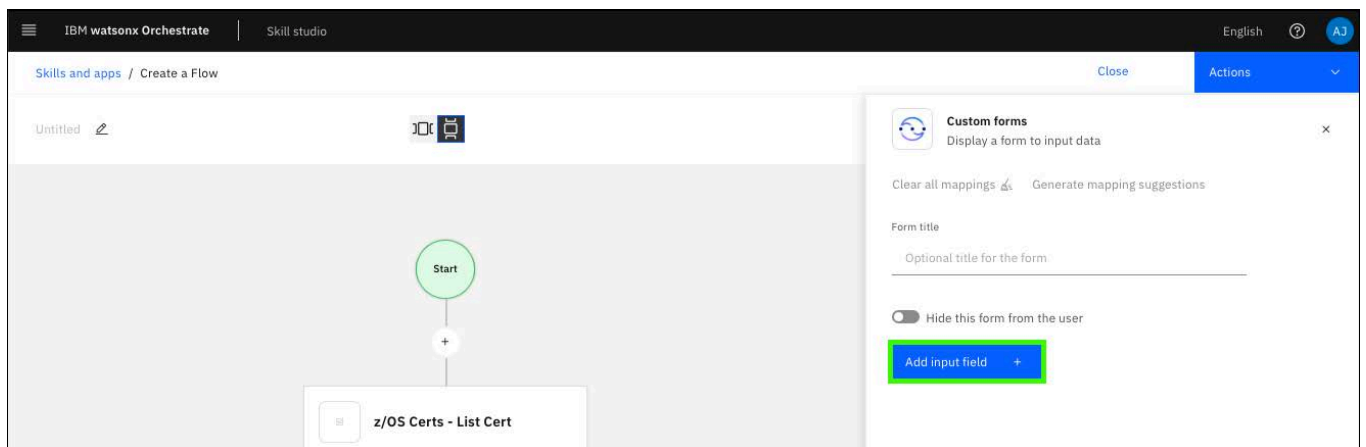
The screenshot shows the IBM watsonx Orchestrate Skill studio interface. On the left, the same flow diagram is visible. On the right, the 'Add Skills' panel is open, and the 'Custom Forms' section is selected. Two skills are listed: 'Input form' (Display a form to input data) and 'Output form' (Display a form to output data). The 'Input form' skill is highlighted with a green border, and the 'Add Skill +' button next to it is also highlighted with a green border.

Skill Name	Description	Action
Input form	Display a form to input data	Add Skill +
Output form	Display a form to output data	Add Skill +

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



18. Click **Add input field +**.



19. Click **Paragraph text** and then click **Next**.

The screenshot shows the IBM watsonx Orchestrate Skill studio interface. The main workspace displays a flow diagram with the following steps: Start (green circle), z/OS Certs - List Cert (white rectangle), Retrieve job output (white rectangle), Input form (white rectangle with a form icon), and End (red circle). The flow is connected by plus signs and arrows.

On the right side, there are two panels:

- Custom forms**: A panel titled "Display a form to input data". It includes options to "Clear all mappings" and "Generate mapping suggestions". There is a "Form title" field with the placeholder "Optional title for the form". A toggle switch for "Hide this form from the user" is currently off. A blue button "Add input field" with a plus icon is at the bottom.
- Add input field**: A panel with two tabs: "Input type" (selected) and "Configuration". Under "Input type", there is a list of input types: "Single line text", "Paragraph text" (highlighted with a green border), "Date", "Decimal number", "Number", and "Confirmation". At the bottom of this panel, there is a "Next" button (highlighted with a green border) and a "Cancel" button.

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

Display label:

`certificate expiration date`

The screenshot displays the IBM watsonx Orchestrate Skill studio interface. The main workspace shows a flow diagram with the following steps: a green 'Start' node, followed by a 'z/OS Certs - List Cert' action, then a 'Retrieve job output' action, an 'Input form' action, and finally an 'End' node. On the right side, the 'Custom forms' panel is open, showing options to 'Clear all mappings' and 'Generate mapping suggestions'. Below this, the 'Form title' field is present, followed by a toggle for 'Hide this form from the user'. An 'Add input field' button is also visible. The 'Add input field' sub-panel is active, showing the 'Input type' selected and the 'Configuration' tab. Under 'Display label', the text 'certificate expiration date' is entered and highlighted with a green border. At the bottom right of the panel, the 'Apply' button is highlighted with a green border, next to a 'Cancel' button.

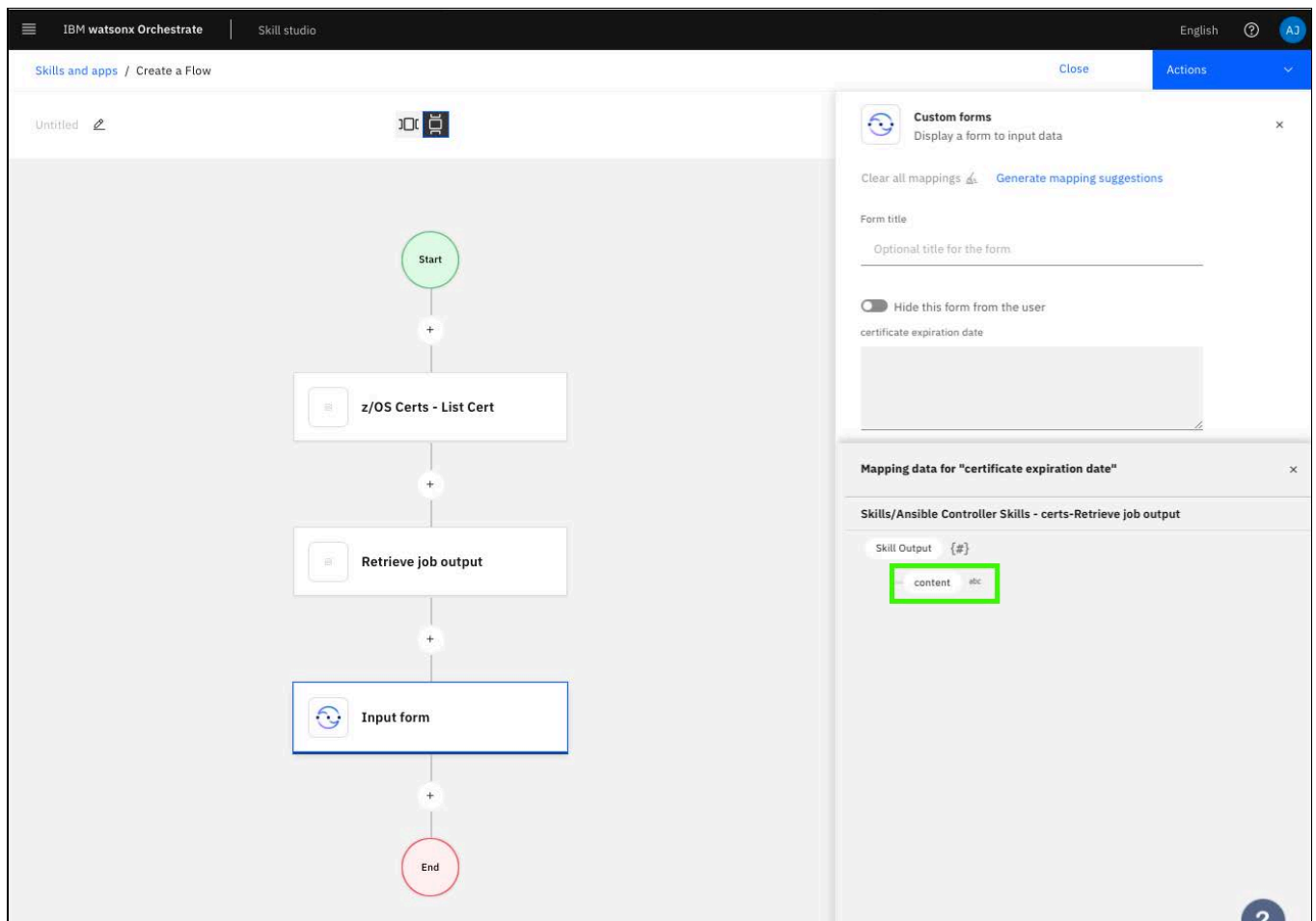
21. Click the **certificate expiration date** entry field.

The screenshot displays the IBM watsonx Orchestrate Skill studio interface. The main workspace shows a flow diagram with the following steps: a green 'Start' node, followed by a connector '+', then a 'z/OS Certs - List Cert' action, another connector '+', a 'Retrieve job output' action, a third connector '+', an 'Input form' action (highlighted with a blue border), a fourth connector '+', and finally a red 'End' node. On the right side, the 'Custom forms' panel is open, showing the configuration for the 'Input form' action. It includes a 'Form title' field with the placeholder 'Optional title for the form', a toggle for 'Hide this form from the user', and a section for 'certificate expiration date' which contains a large empty text area highlighted with a green border. Below this area is a blue button labeled 'Add input field +'. The top of the interface shows the 'Skills and apps / Create a Flow' breadcrumb and a 'Close' button. The bottom right corner has a help icon (question mark in a circle).

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

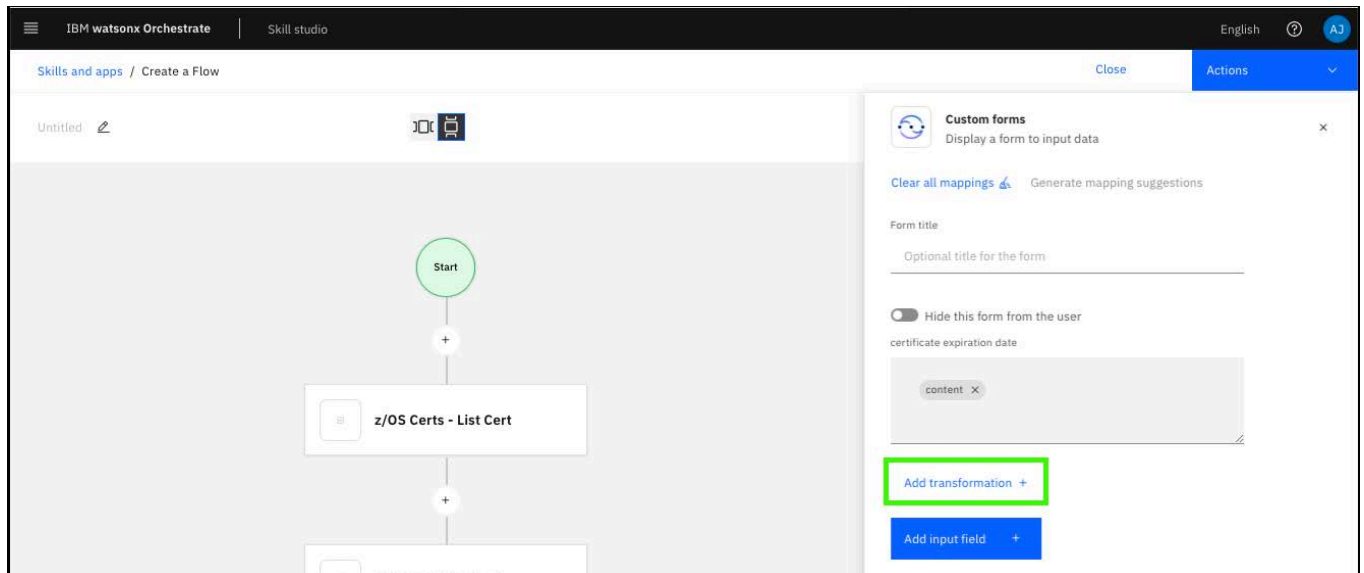
The screenshot displays the IBM watsonx Orchestrate Skill studio interface. The main workspace shows a flow diagram with the following steps: Start (green circle), a connector (+), z/OS Certs - List Cert (white rectangle), another connector (+), Retrieve job output (white rectangle), a third connector (+), Input form (white rectangle with a form icon), a fourth connector (+), and End (red circle). On the right, the 'Custom forms' panel is open, showing the configuration for the 'Input form' step. It includes a 'Form title' field, a 'Hide this form from the user' toggle, and a 'certificate expiration date' input field. Below this, the 'Mapping data for "certificate expiration date"' panel is visible, showing a list of skills: 'z/OS Certs - List Cert' and 'Retrieve job output'. The 'Retrieve job output' skill is highlighted with a green border, indicating it is the selected mapping for the 'certificate expiration date' field.

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](#).

The screenshot shows the IBM watsonx Orchestrator Skill Studio interface. On the left, a flow diagram is visible with the following steps: Start (green circle), z/OS Certs - List Cert (white rectangle), Retrieve job output (white rectangle), Input form (white rectangle with a form icon), and End (red circle). On the right, the 'Custom forms' configuration panel is open. It shows the 'Form title' field with the text 'certificate expiration date'. Below it, the 'Hide this form from the user' toggle is turned off. The 'Add Transformations for "certificate expiration date"' section is expanded, showing the 'Operator' as 'Get substring by regular expression' and the 'Regular Expression' as '(?<=End Date:).[^\"]*'. The 'Add' button at the bottom right of the panel is highlighted with a green border.

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

Form title:

certificate expiration date

The screenshot shows the IBM watsonx Orchestrator Skill Studio interface after the configuration changes. The flow diagram on the left is the same as in the previous screenshot. The 'Custom forms' configuration panel on the right has been updated: the 'Form title' field now contains 'certificate expiration date', and the 'Hide this form from the user' toggle is now turned on (indicated by a green circle). The 'Add Transformations' section is no longer expanded. The 'Add input field' button at the bottom of the panel is highlighted with a blue border.

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

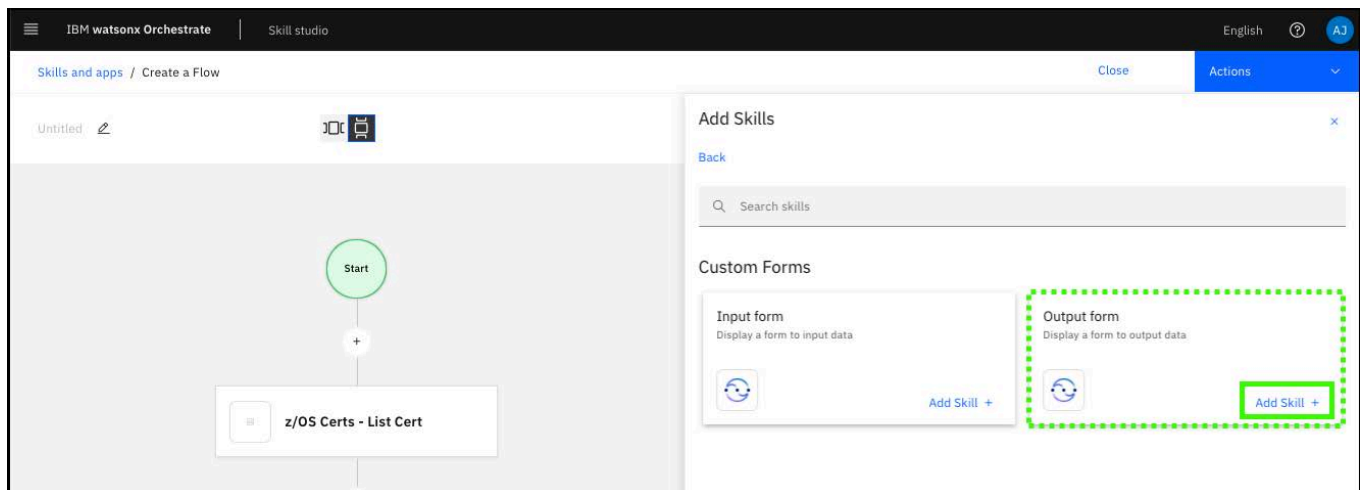
28. Click the **+** icon **below** the **Input form**.

The screenshot shows the IBM watsonx Orchestrate Skill studio interface. On the left, a flow diagram is visible with the following steps: a green 'Start' circle, a '+' connector, a 'z/OS Certs - List Cert' skill tile, another '+' connector, a 'Retrieve job output' skill tile, a third '+' connector, an 'Input form' skill tile, a fourth '+' connector (highlighted with a green box), and finally a red 'End' circle. On the right, the 'Custom forms' panel is open, showing options to 'Clear all mappings', 'Generate mapping suggestions', and a form title 'certificate expiration date'. There is a checkbox for 'Hide this form from the user' which is checked. Below this, there is a 'content' field with a close icon. At the bottom of the panel, there is a 'Transformations' section with the text '- Get substr by regular expression' and a button to 'Add transformation'. At the very bottom of the panel is a blue button labeled 'Add input field' with a '+' icon.

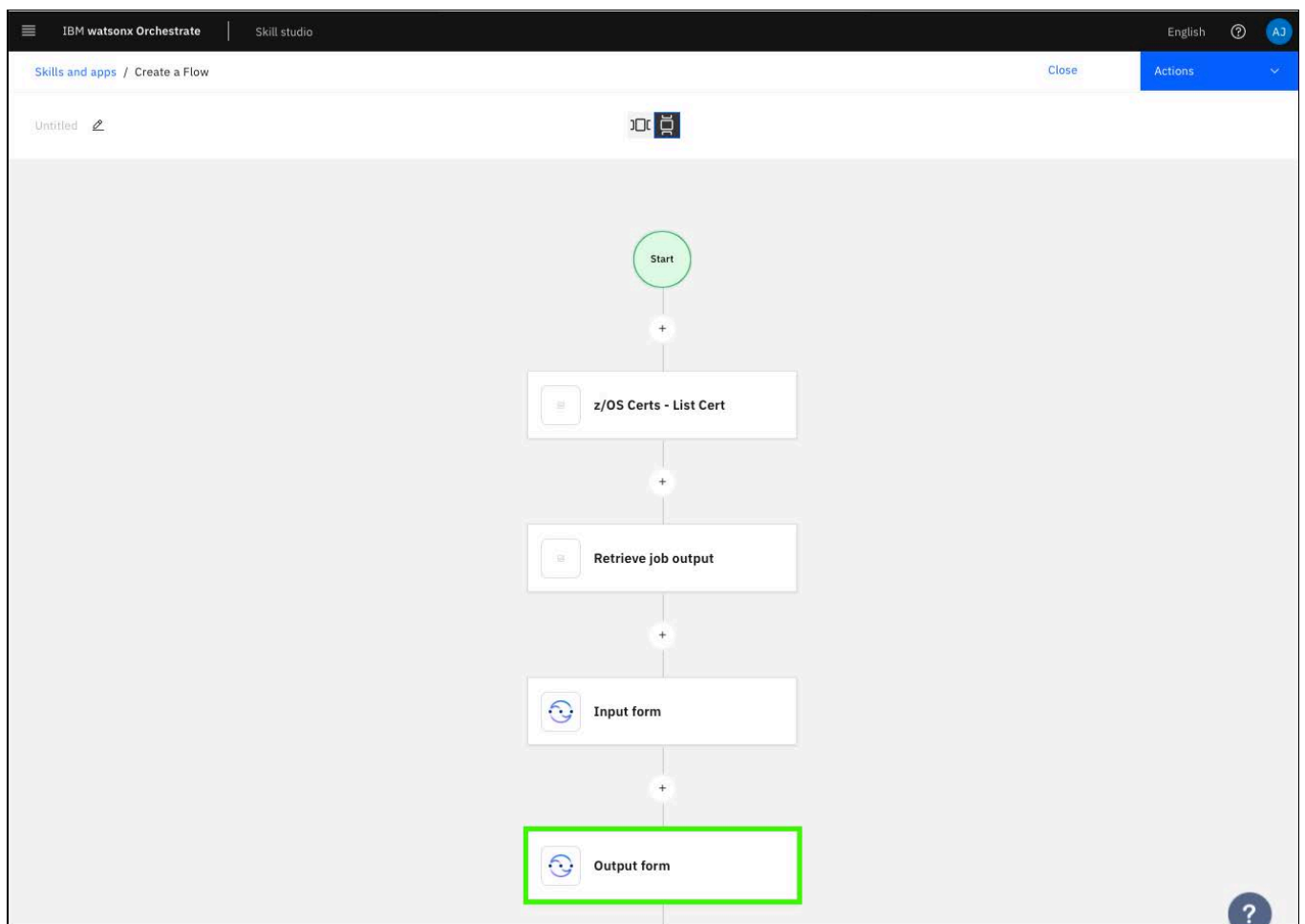
29. Click **Custom forms**.

The screenshot shows the IBM watsonx Orchestrate Skill studio interface. On the left, the flow diagram is partially visible, showing a green 'Start' circle, a '+' connector, and a 'z/OS Certs - List Cert' skill tile. On the right, the 'Add Skills' panel is open, displaying a search bar and a grid of skill categories. The 'Custom forms' category, which has 2 skills, is highlighted with a green box. Other visible categories include 'Flow controls' (1 skill), 'Ansible Controller Skills - certs' (3 skills), 'Coupa' (23 skills), 'HubSpot CRM', 'z/OS operations' (1 skill), 'ZoomInfo' (26 skills), and 'Apptio Targetprocess'.

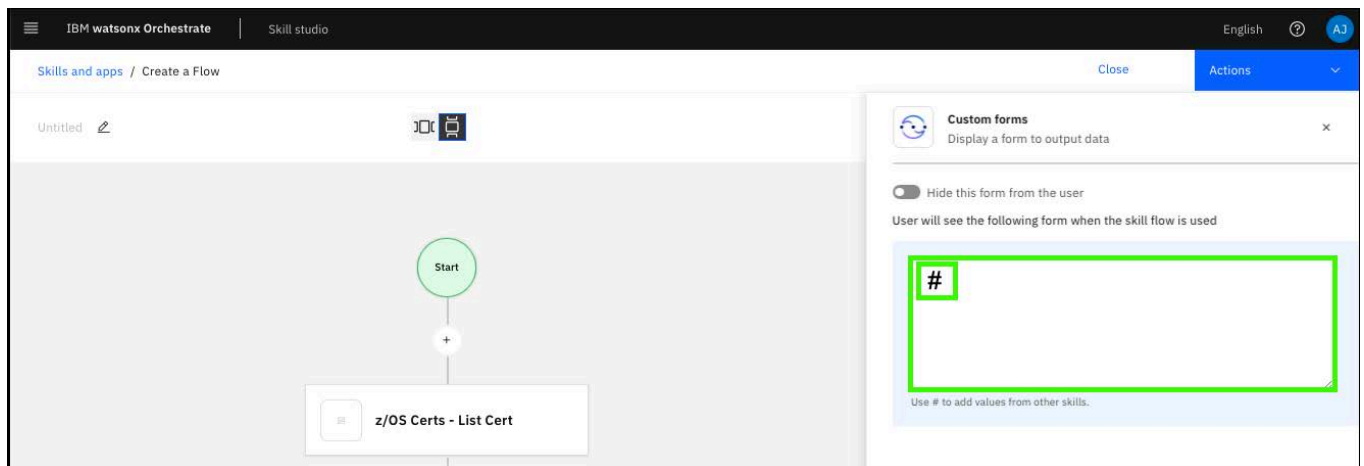
30. Click **Add Skill +** in the **Output form** tile.



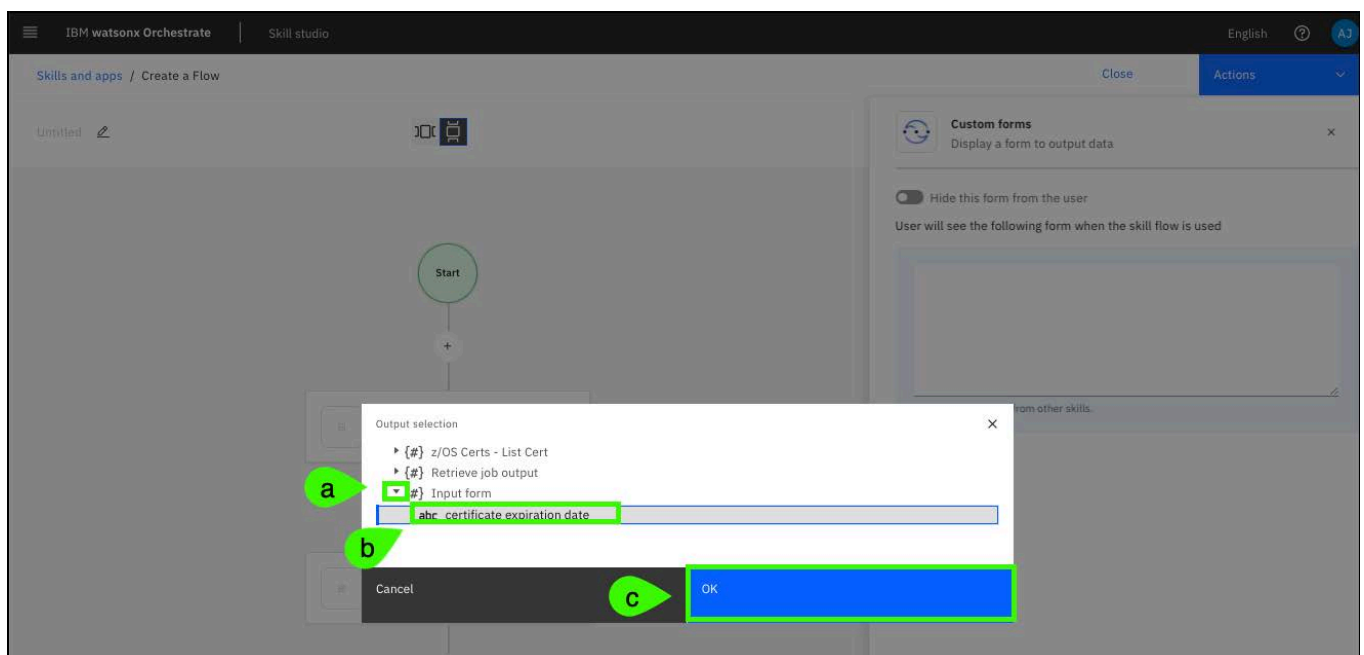
31. Click the **Output form** skill.



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.



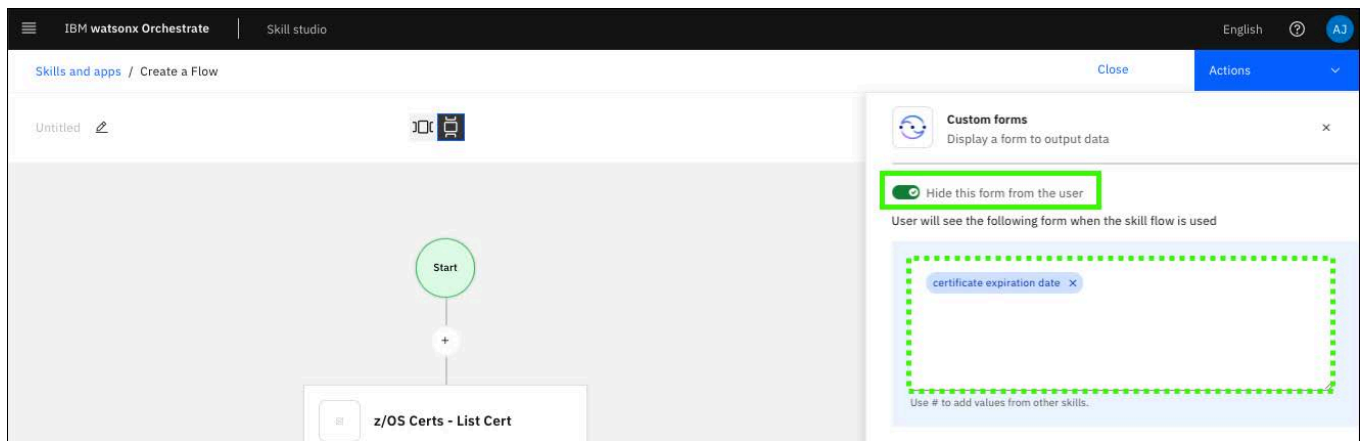
33. Expand (a) **Input form**, select (b) **certificate expiration date**, and then click (c) **OK**.



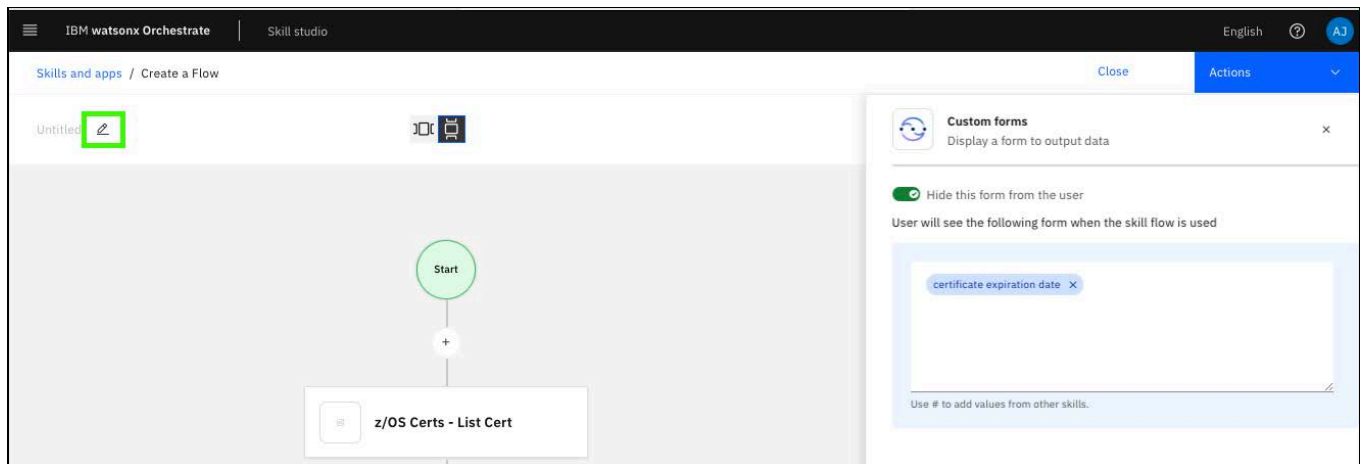
34. Enable the **Hide this form from the user** option.

? 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.



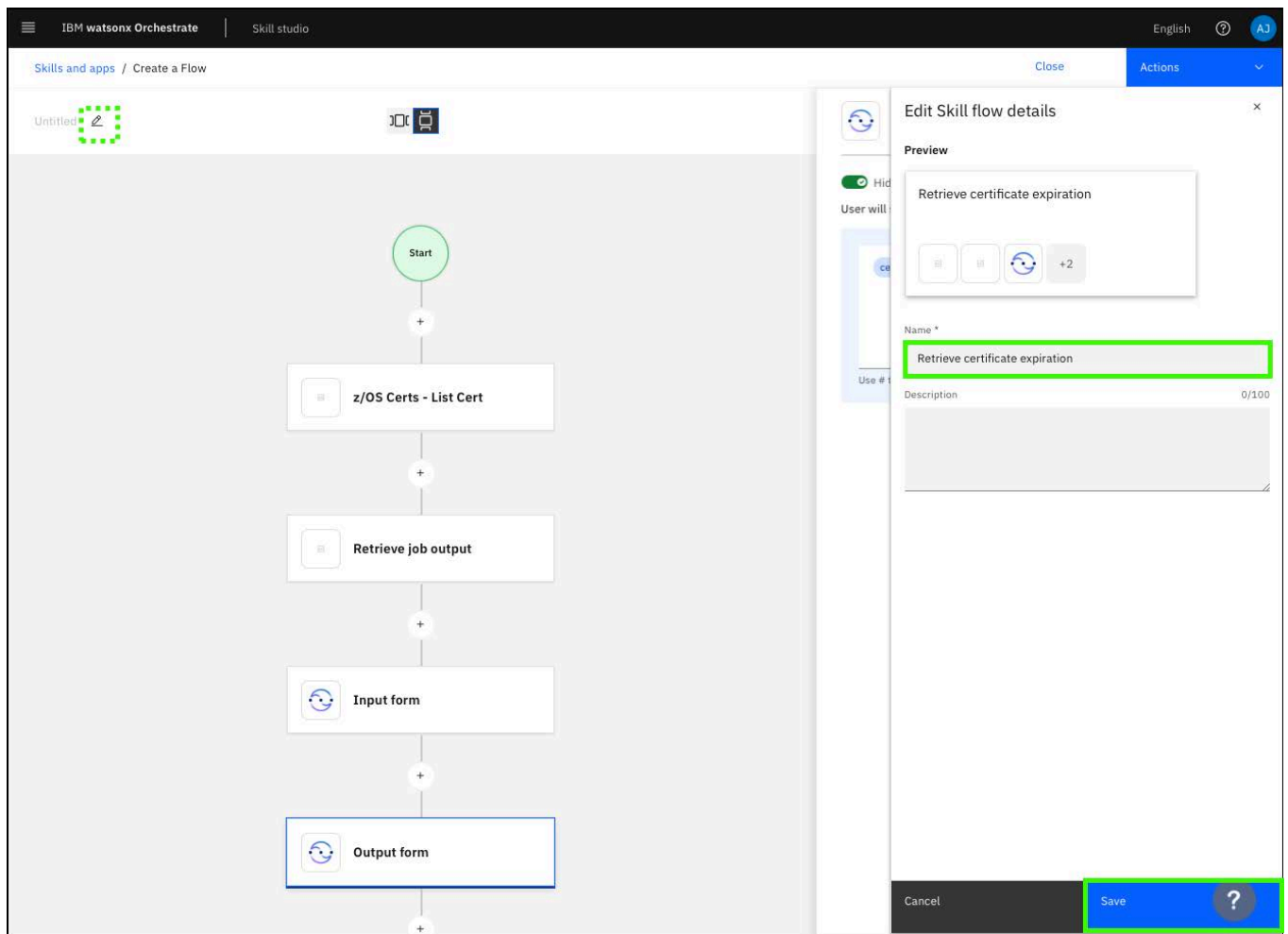
35. Click the **pencil icon** (✎).



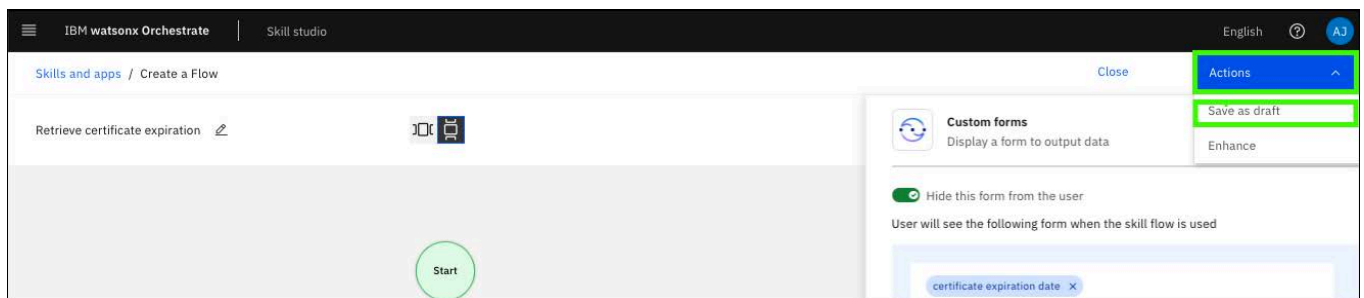
36. Enter `Retrieve certificate expiration` in the **Name** field and click **Save**.

Name:

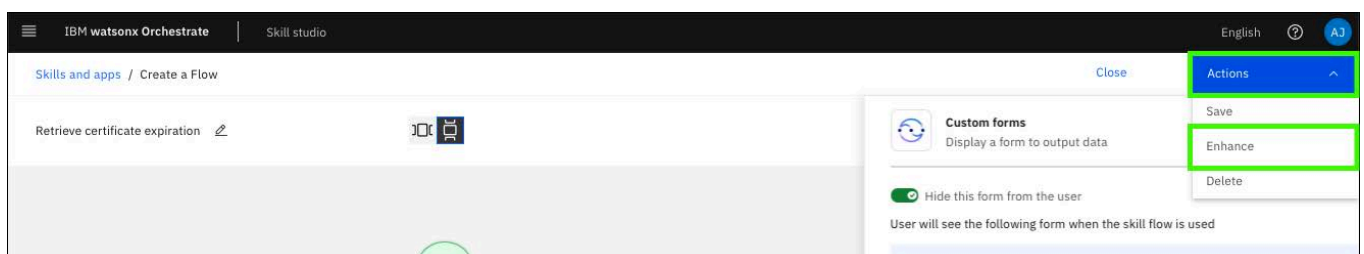
Retrieve certificate expiration



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 in IBM watsonx Orchestrator. The main heading is 'Enhance the "Retrieve certificate expiration" skill' with a subtext 'Add details that will make people want to use this skill.' Below this are three tabs: 'Name', 'Phrases', and 'Next best skills'. The 'Name' tab is active, showing a form with the following fields:

- Name***: A text box containing 'Retrieve certificate expiration'.
- Description**: A text box with a character count '0/100' and the placeholder 'Enter the description'.
- API version***: A dropdown menu showing '1.0.0'.
- Categories**: A text box with the placeholder 'Add categories'.
- App**: A dropdown menu showing 'Skill flows'.

On the right side, there is a 'Preview' section with two visualizations of how the skill will appear in the catalog and skill set. Both visualizations show a card with the title 'Retrieve certificate expiration' and a circular refresh icon. At the bottom right, there are three buttons: 'Cancel', 'Publish' (highlighted with a green border), and 'Save as draft'. A help icon (?) is also visible in the bottom right corner.

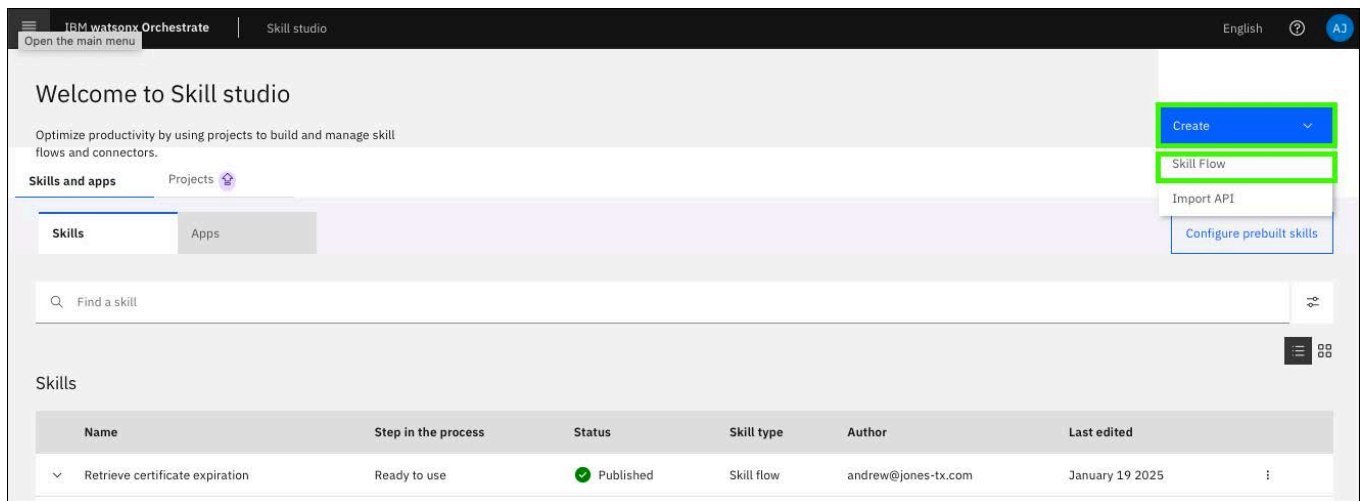
You created a new skill flow that accomplishes part of the use case – retrieving and displaying the expiration date of a z/OS certificate based on the certificate label the user provides.

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**.



Welcome to Skill studio

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

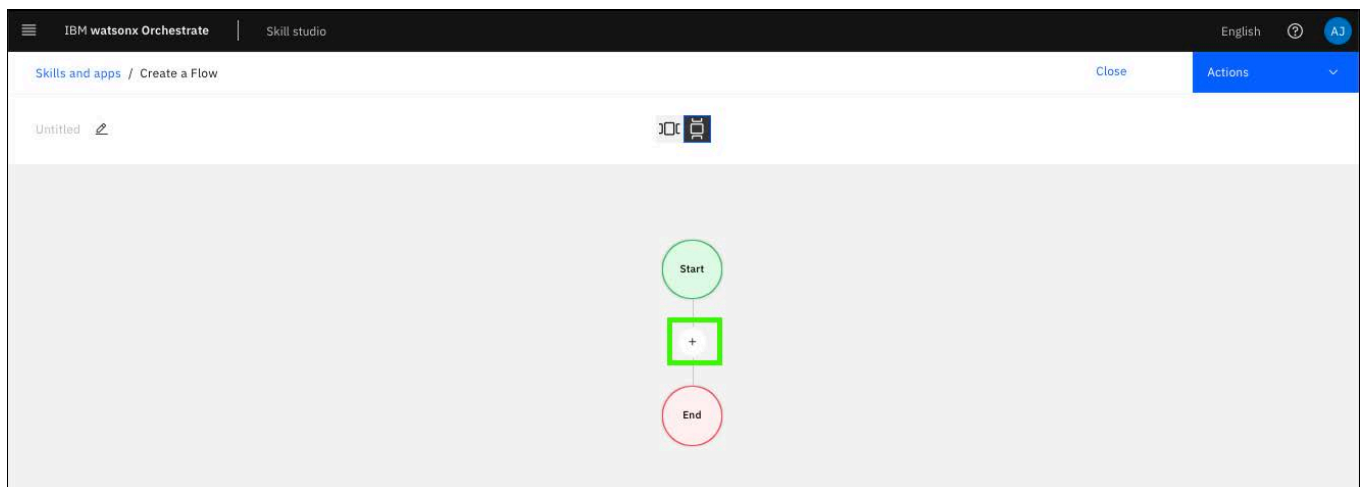
Skills and apps Projects

Skills Apps

Find a skill

Name	Step in the process	Status	Skill type	Author	Last edited
Retrieve certificate expiration	Ready to use	Published	Skill flow	andrew@jones-tx.com	January 19 2025

2. Click the + icon.



Skills and apps / Create a Flow

Close Actions

Untitled

Start

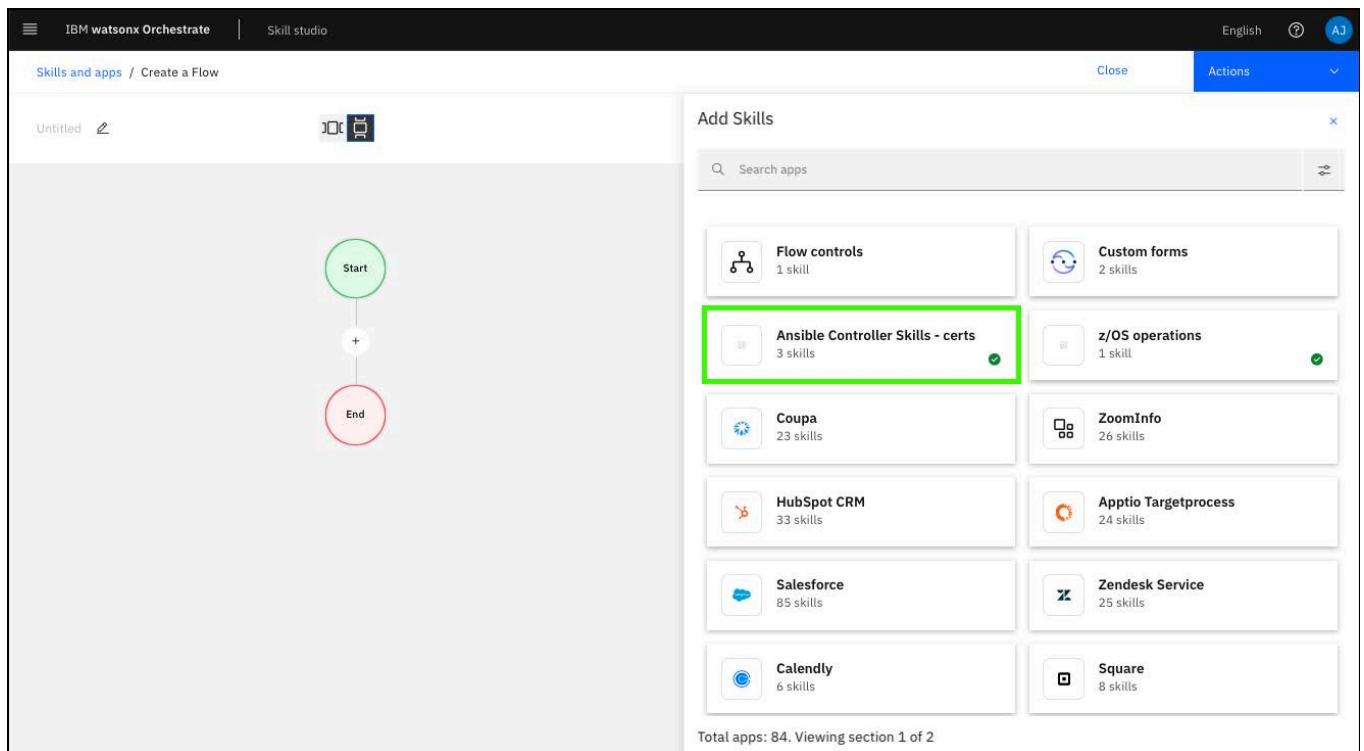
+

End

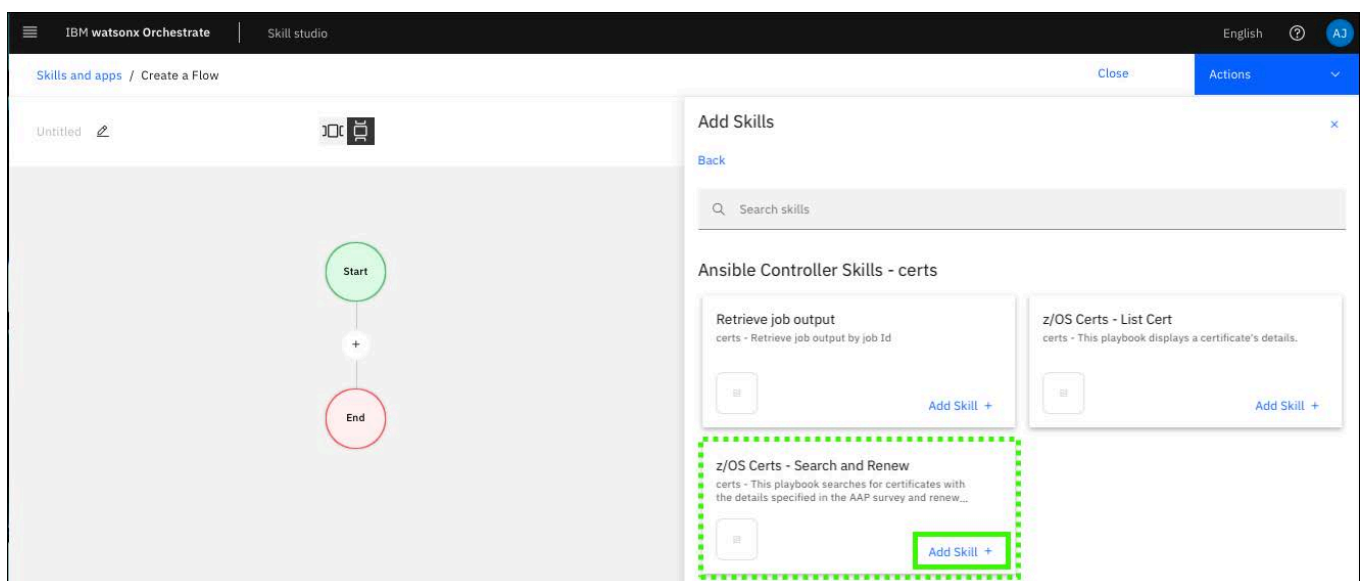
3. Click the **certs** app.



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

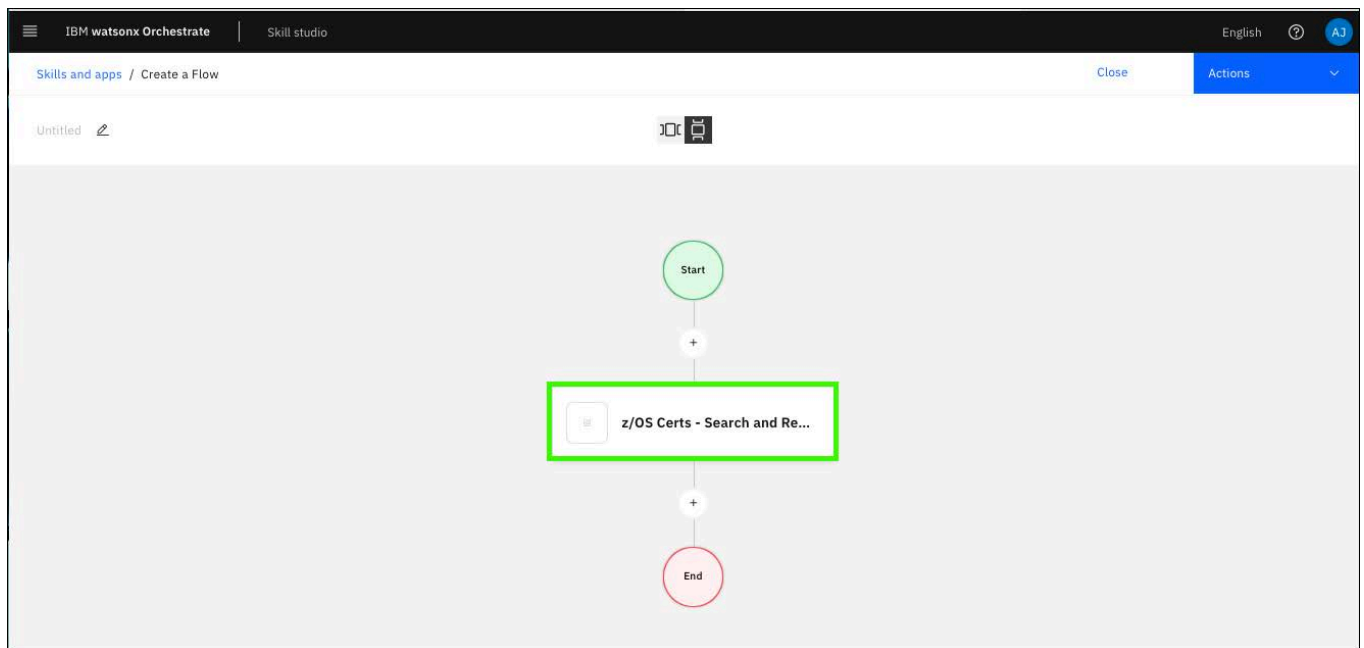


4. Click **Add Skill +** in the **z/OS Certs - Search and Renew** tile.



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**.

The screenshot shows the IBM watsonx Orchestrate Skill studio interface with the 'Input' configuration panel open for the 'z/OS Certs - Search and Re...' skill. The flow diagram is visible in the background. The 'Input' panel on the right has a green border and contains the following elements:

- Ansible Controller Skills - certs**: A title bar with a close button (X).
- Repeat this skill**: A toggle switch.
- Input**: A tab selected with a green border.
- Output**: A tab.
- Clear all mappings**: A button with a trash icon.
- Hide this form from the user**: A toggle switch.
- Input fields**: A list of input fields for extra variables, each with a text input area and a pencil icon for editing:
 - 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 (✎).

IBM watsonx Orchestrate | Skill studio

Skills and apps / Create a Flow

Untitled

Start

+

z/OS Certs - Search and Re...

+

End

Ansible Controller Skills - certs

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

Repeat this skill

Input

Output

Clear all mappings

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

8. 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`.

The screenshot shows the IBM watsonx Orchestrate Skill Studio interface. On the left, a flow diagram is visible with a 'Start' node, a connector, a skill node labeled 'z/OS Certs - Search and Re...', another connector, and an 'End' node. On the right, the configuration panel for the 'Ansible Controller Skills - certs' skill is open. The 'Input' tab is selected, and the 'Hide this form from the user' option is enabled. The 'extra_vars.sign_with_survey' field is highlighted with a green box and contains the value '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.

The screenshot shows the same IBM watsonx Orchestrate Skill Studio interface, but the 'Output' tab is now selected in the configuration panel. The 'Hide this form from the user' option is still enabled. The 'Integer' field is labeled 'job' and the 'Text' field is labeled 'status'.

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:

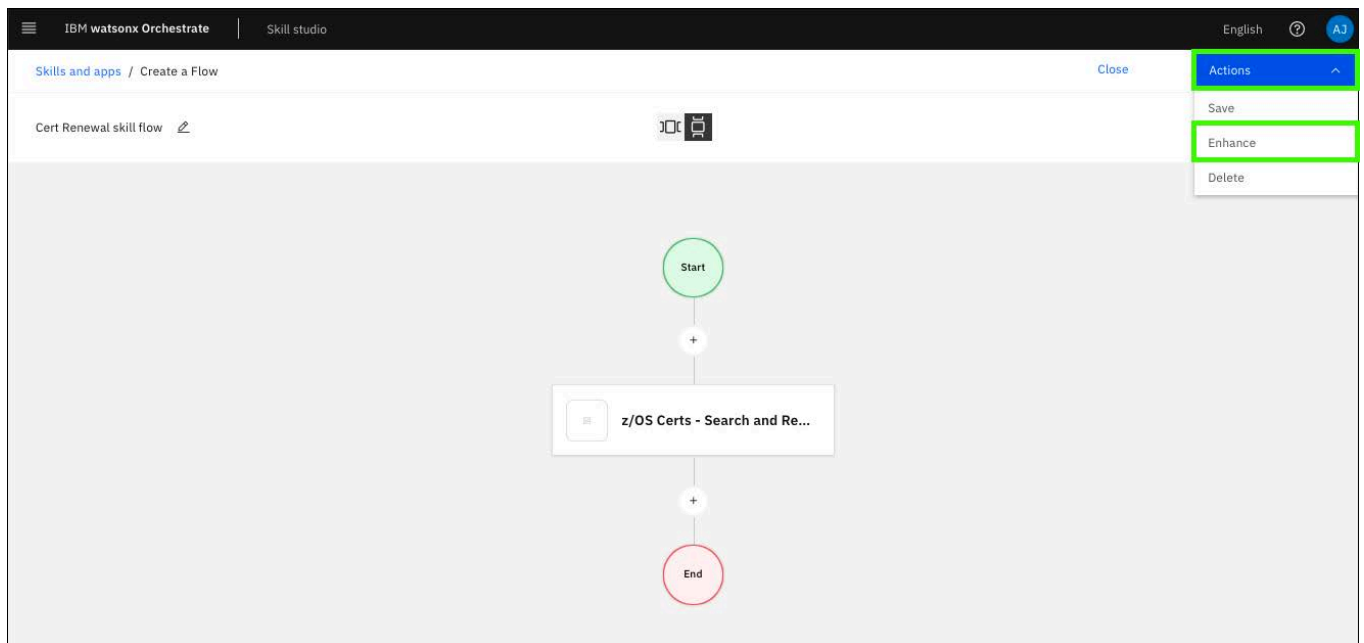
Cert Renewal skill flow

The screenshot shows the IBM watsonx Orchestrate Skill studio interface. The main workspace displays a workflow diagram with a 'Start' node, a connector, a task node labeled 'z/OS Certs - Search and Re...', another connector, and an 'End' node. The 'Edit Skill flow details' panel on the right shows the 'Name' field set to 'Cert Renewal skill flow'. The 'Actions' menu at the top right is open, and the 'Save' button is highlighted. A green box highlights the 'Save' button, and a green arrow labeled 'c' points to it.

12. Click **Actions** and then click **Save as draft**.

The screenshot shows the IBM watsonx Orchestrate Skill studio interface. The main workspace displays the same workflow diagram. The 'Actions' menu at the top right is open, and the 'Save as draft' option is highlighted. A green box highlights the 'Save as draft' option, and a green arrow labeled 'c' points to it.

13. Click **Actions** and then click **Enhance**.



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

The screenshot shows the 'Enhance the "Cert Renewal skill flow" skill' settings page in the IBM watsonx Orchestrate Skill studio. The page has tabs for 'Name', 'Phrases', and 'Next best skills', with 'Name' selected. The main content area includes fields for 'Name*' (filled with 'Cert Renewal skill flow'), 'Description' (with a placeholder 'Enter the description' and a 0/100 character count), 'API version*' (filled with '1.0.0'), 'Categories' (with a placeholder 'Add categories'), and 'App' (with a placeholder 'Skill flows'). On the right, a 'Preview' section shows how the skill will look in the catalog and in the skill set, both displaying a card for 'Cert Renewal skill flow'. At the bottom right, there are three buttons: 'Cancel', 'Publish' (highlighted with a green border), and 'Save as draft'.

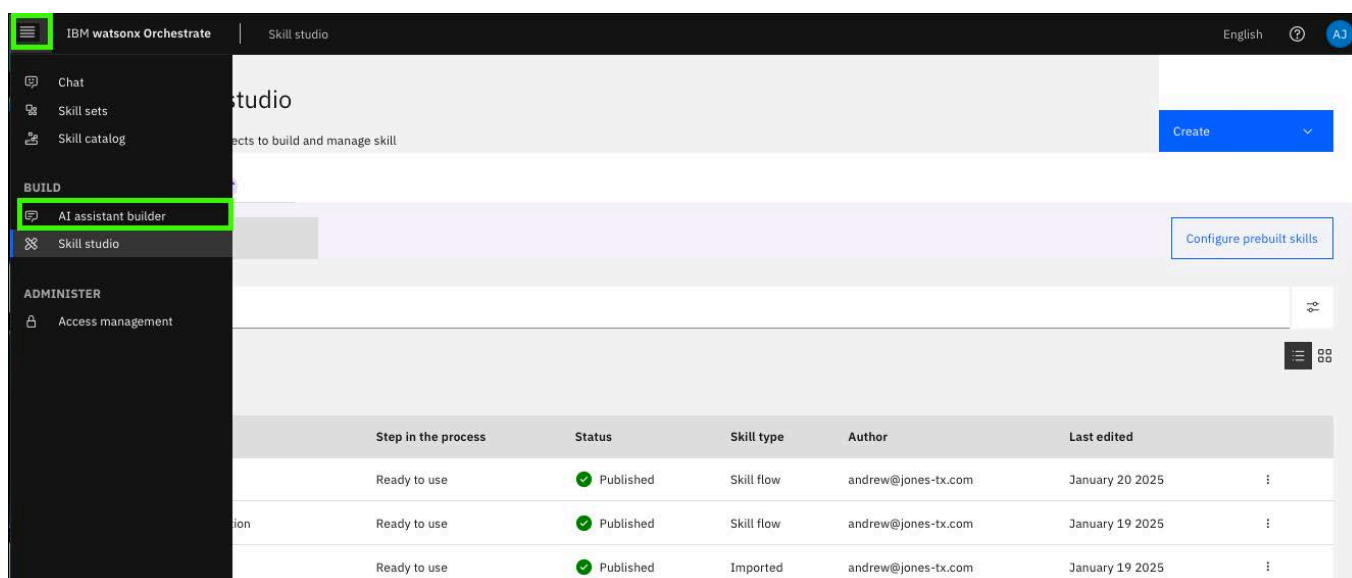
Add the skill flows to the assistant

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:

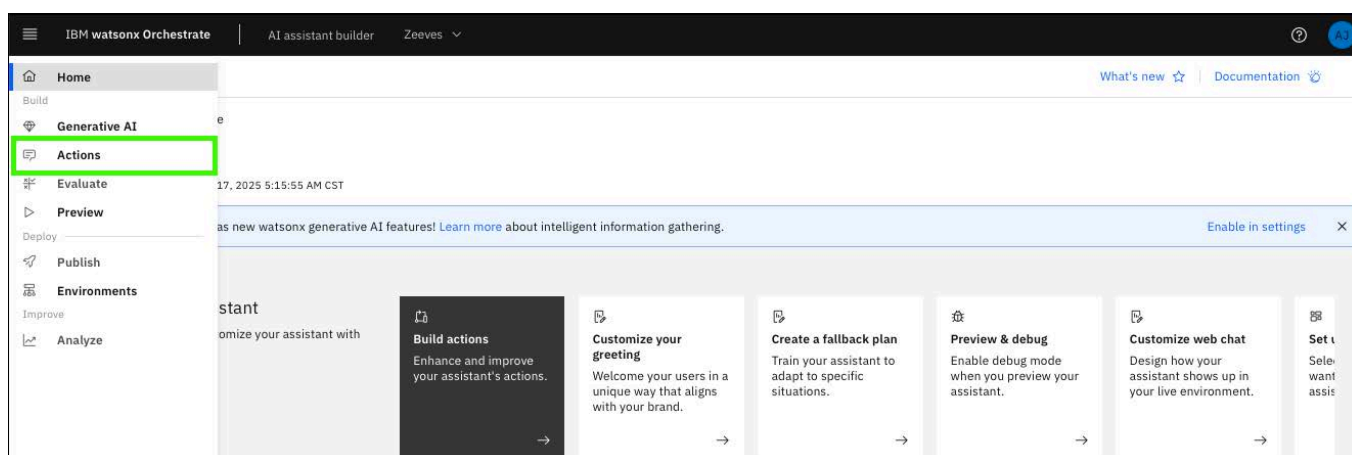
- Retrieve certificate expiration – maps the user prompted certificate label as input and extracts the certificate expiration date from the Ansible job's output.
- Cert Renewal skill flow – maps the user prompted certificate label and new expiration date as input and runs the Search and Renew Ansible 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.

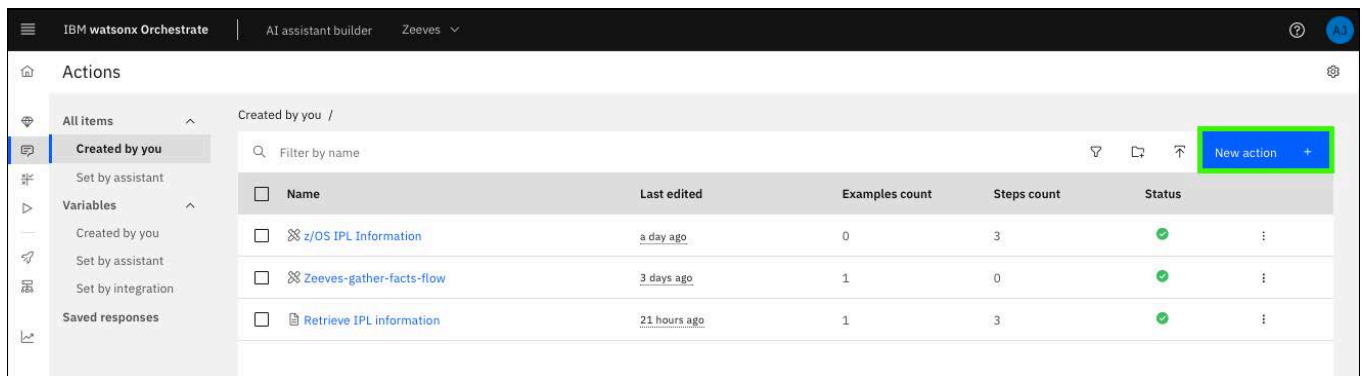
1. Open **AI assistant builder** in watsonx Orchestrate.



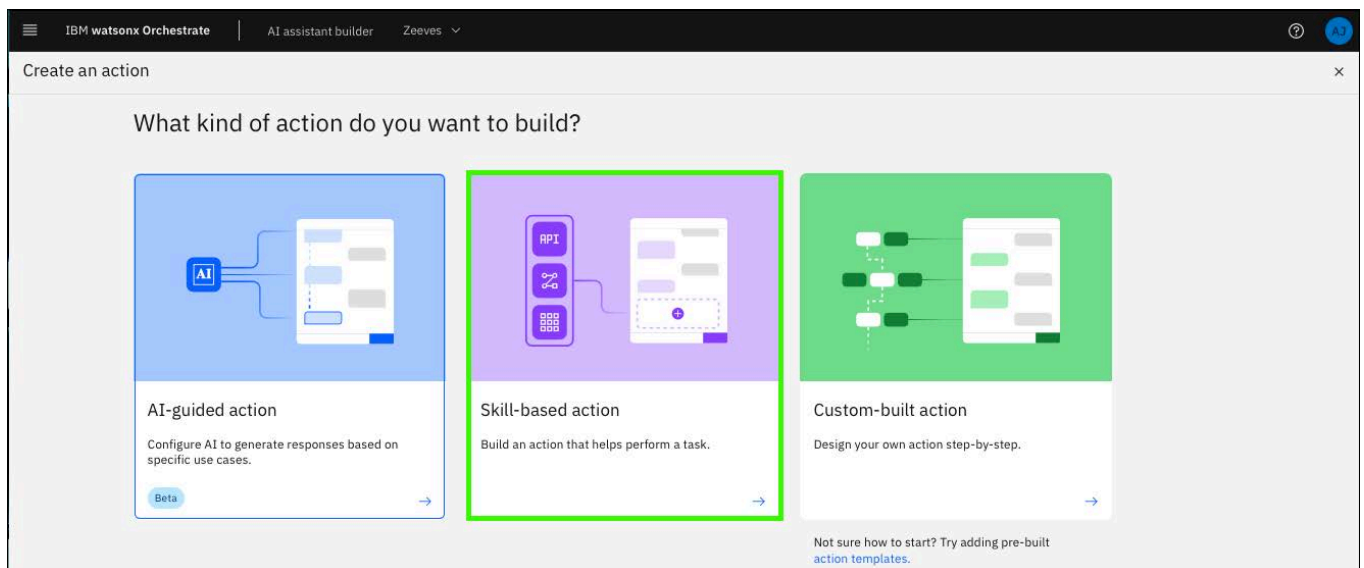
2. Click **Actions**.



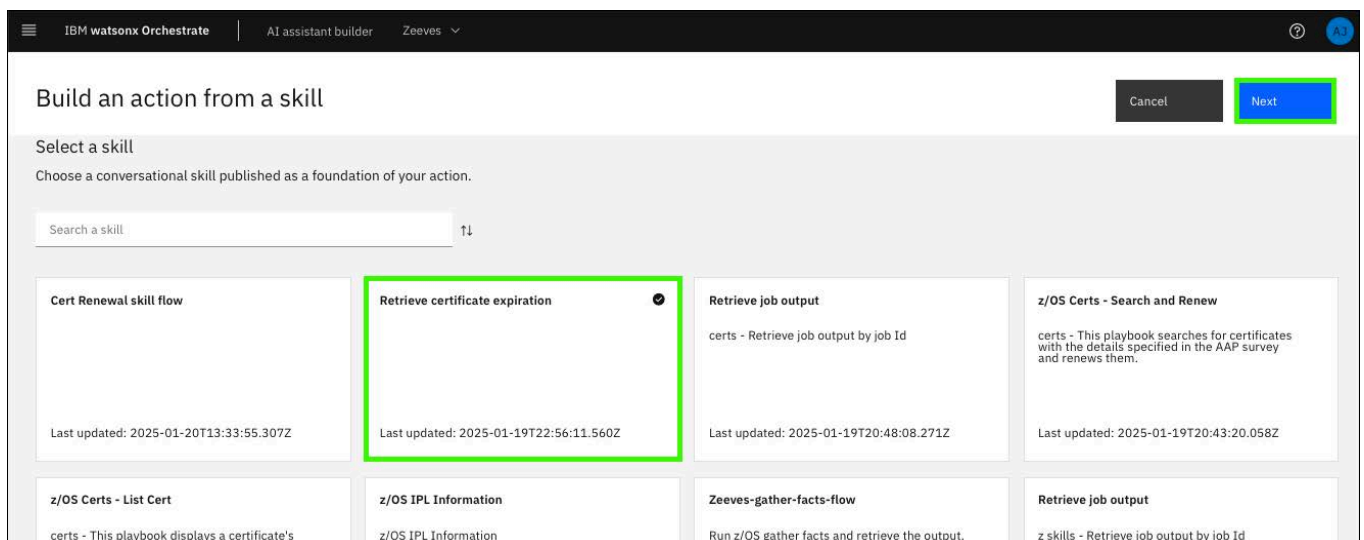
3. Click **New action+**.



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



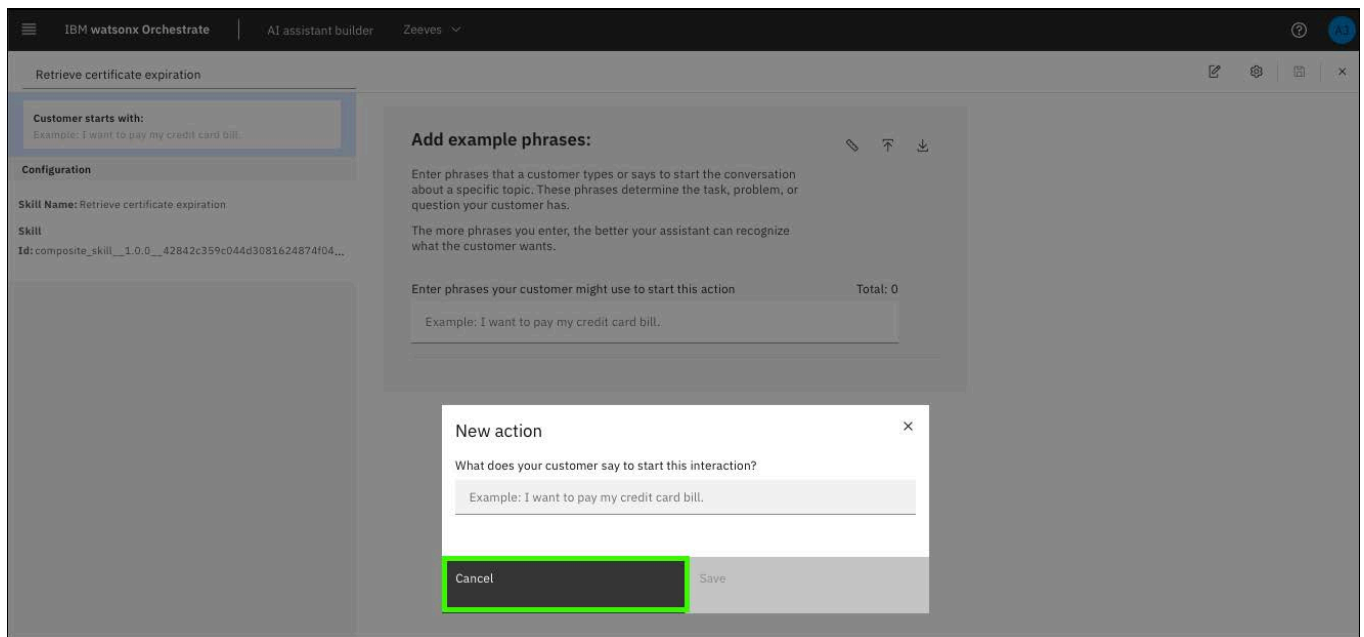
5. Click the **Retrieve certificate expiration** tile and then click **Next**.



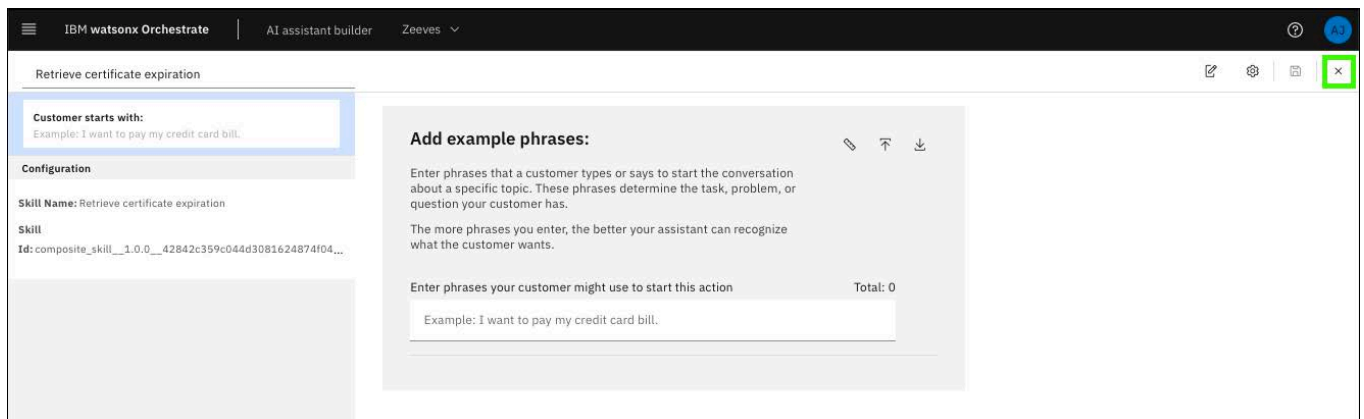
6. Click **Cancel** on the **New action** dialog.



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.



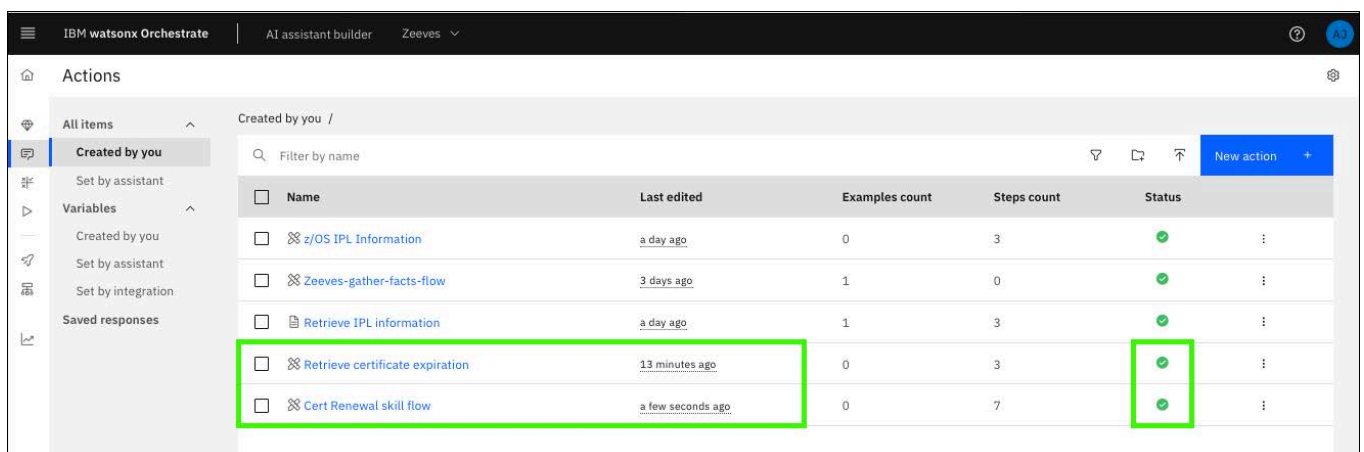
7. Click **x** to close the **Retrieve certificate expiration** skill.



8. Repeat steps 3 - 7 to create a skill-based action for the **Cert Renewal skill flow**.

 This action is also triggered from a custom-built action. Do not enter any example phrases.

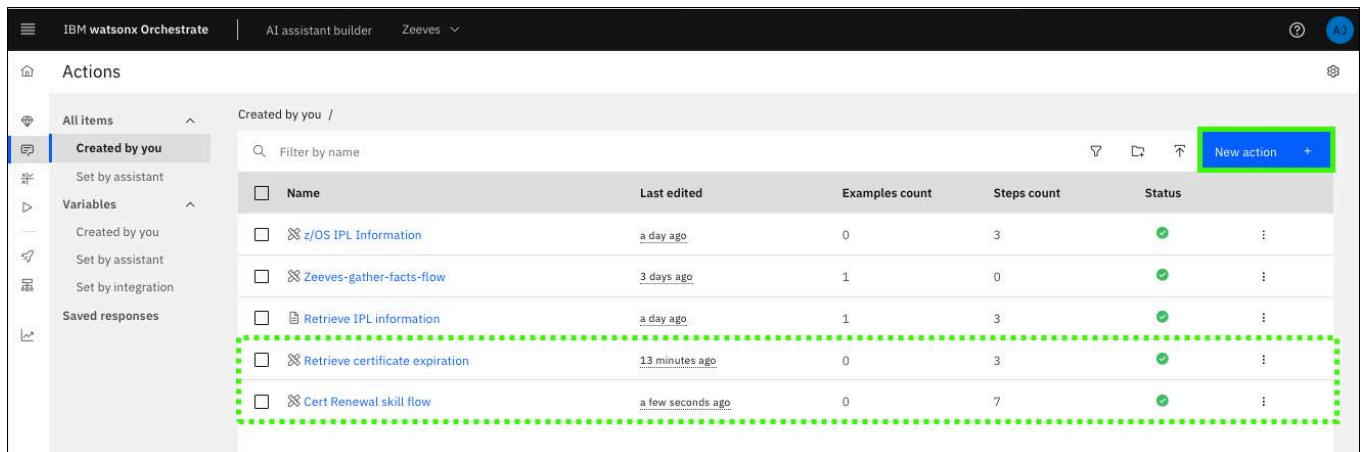
9. Verify that both skill-based actions are available.



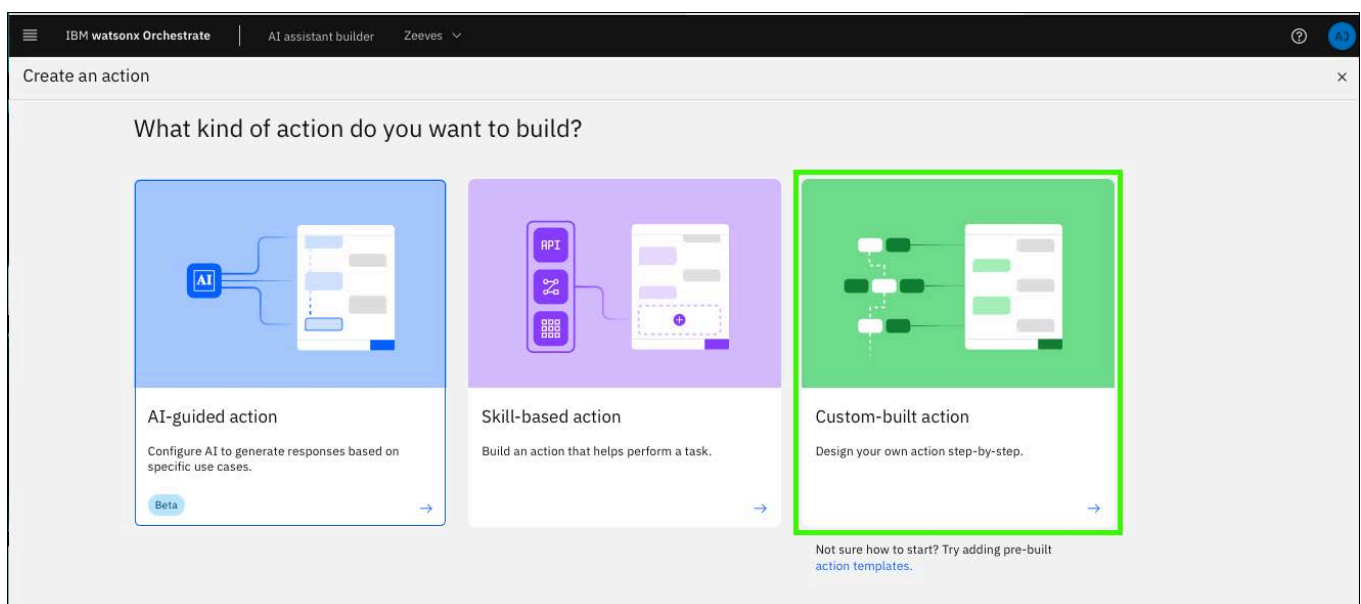
Create a custom-built action for SSL Certificate Renewal

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 +**.



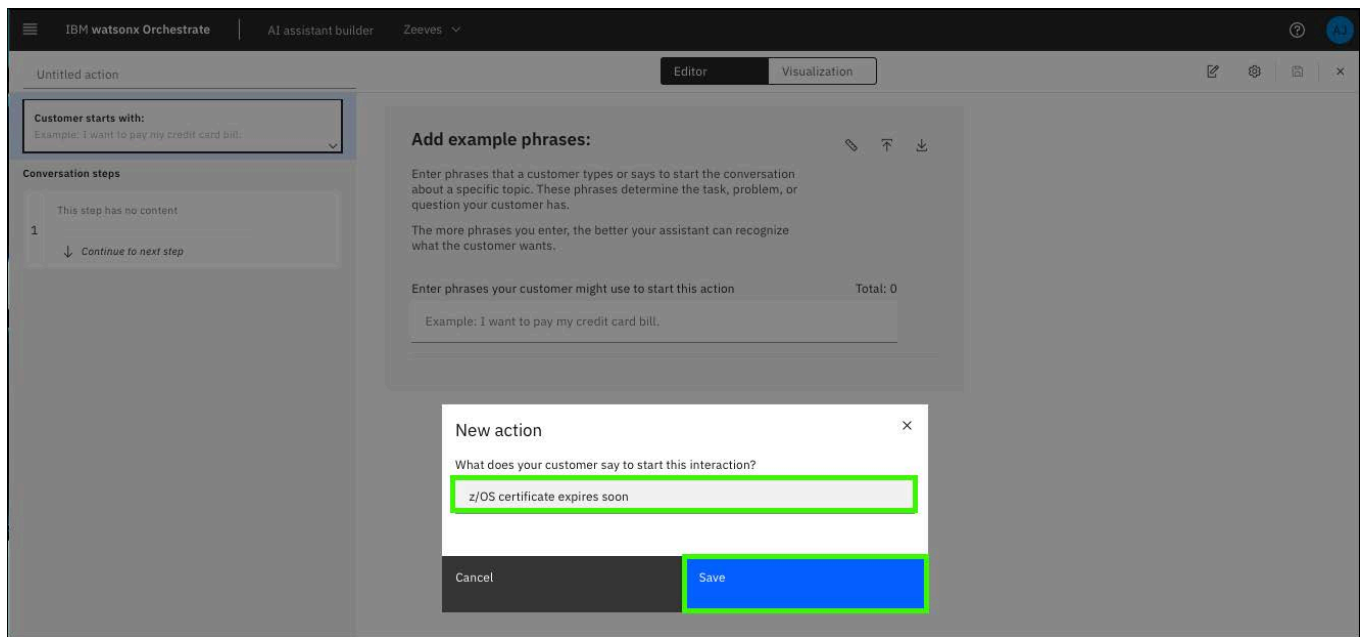
2. Click **Custom-built-action**.



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

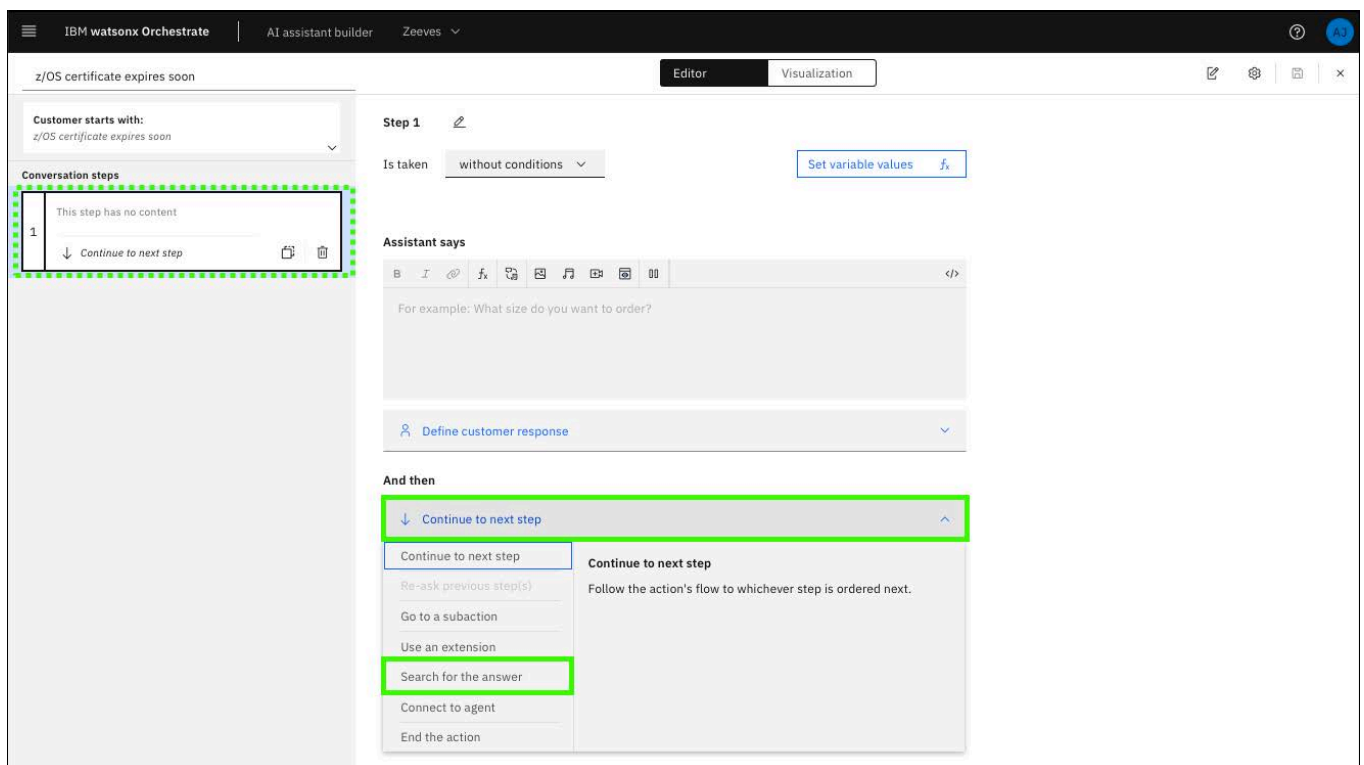
What does your customer say to start this interaction:

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.

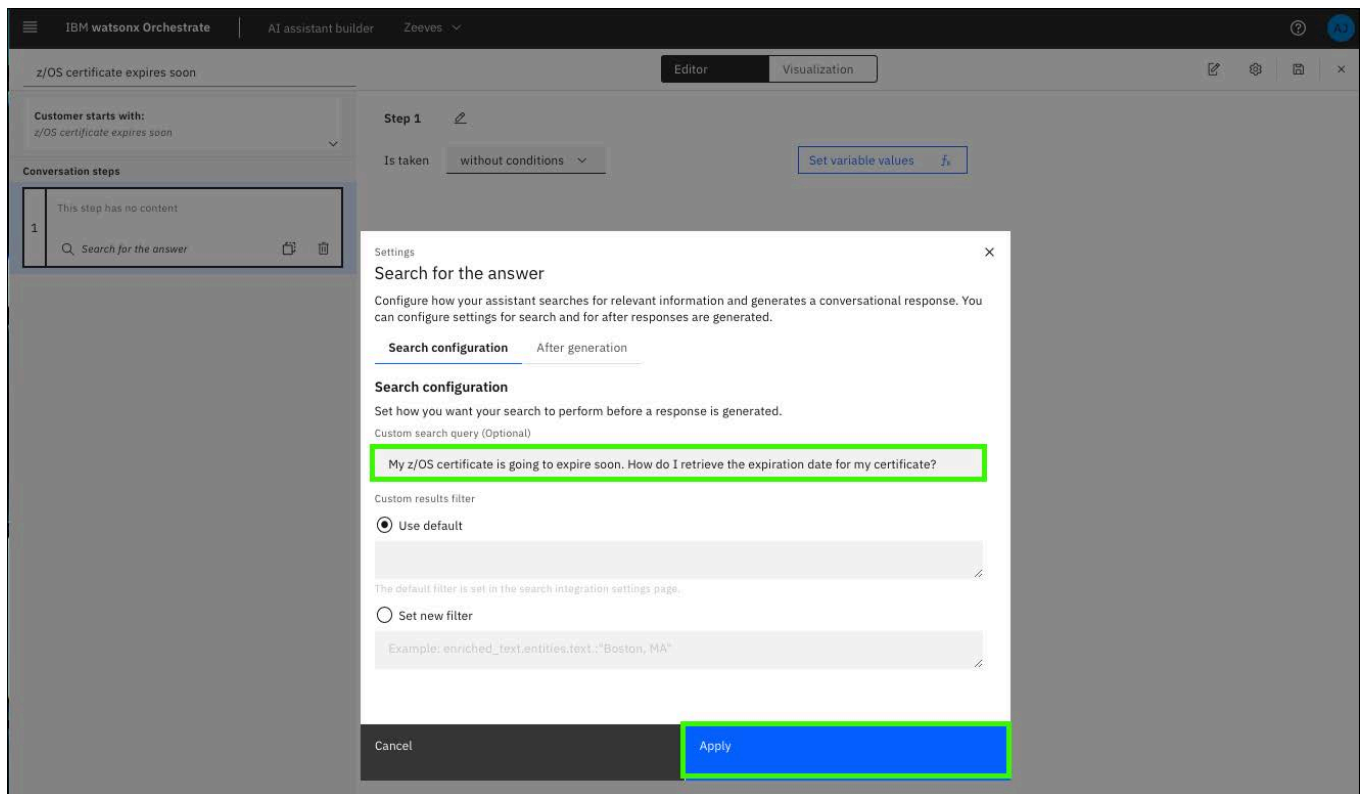
Like in the IPL Information scenario, add a custom search query so when conversational search is run in the first conversation step, the query used is hardcoded and not what the user input.

5. Click **Edit settings**.

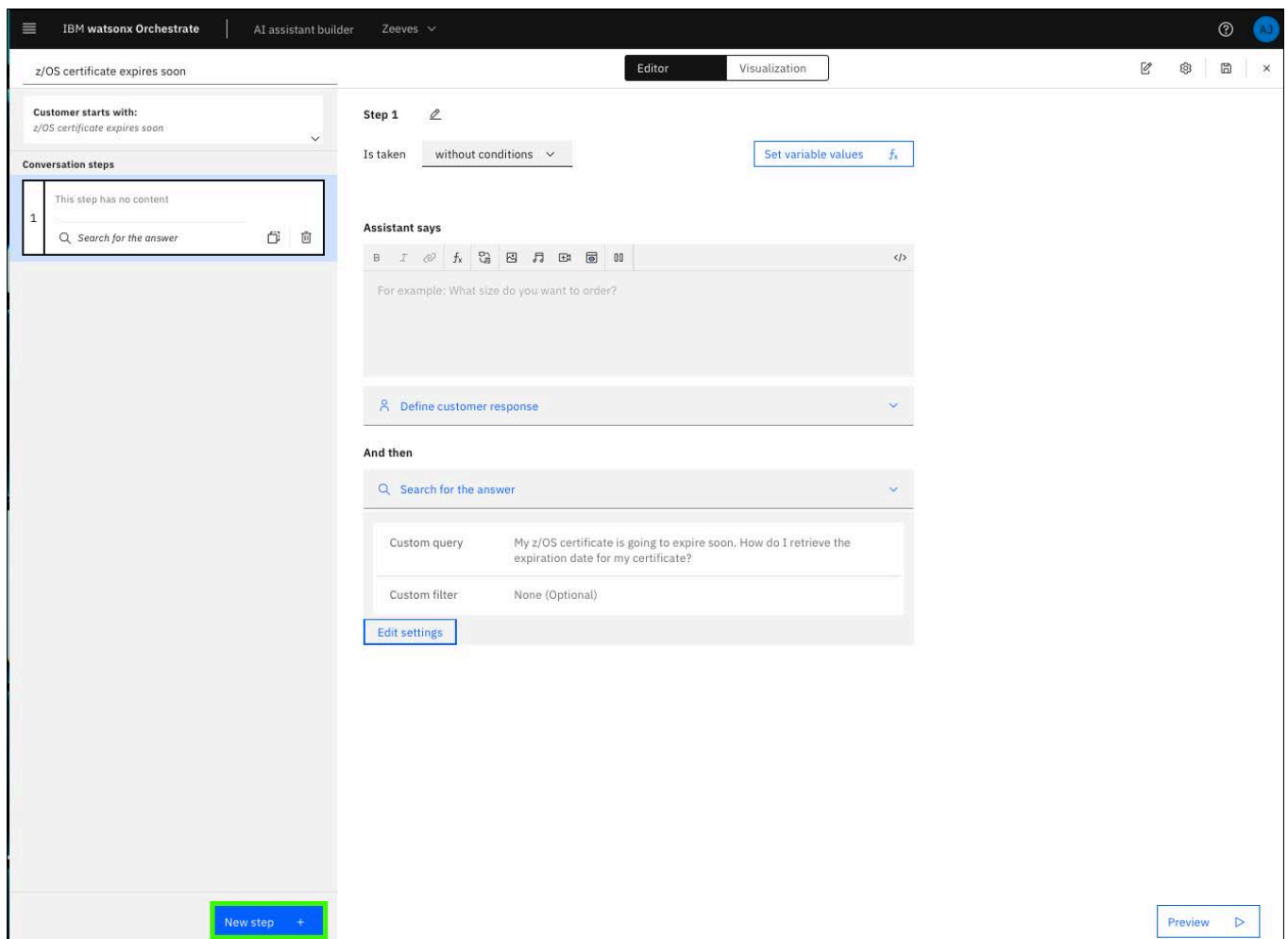
6. Enter the following prompt to be used in the **Custom search query** field and then click **Apply**.

Custom search query:

My z/OS certificate is going to expire soon. How do I retrieve the expiration date for my certificate?



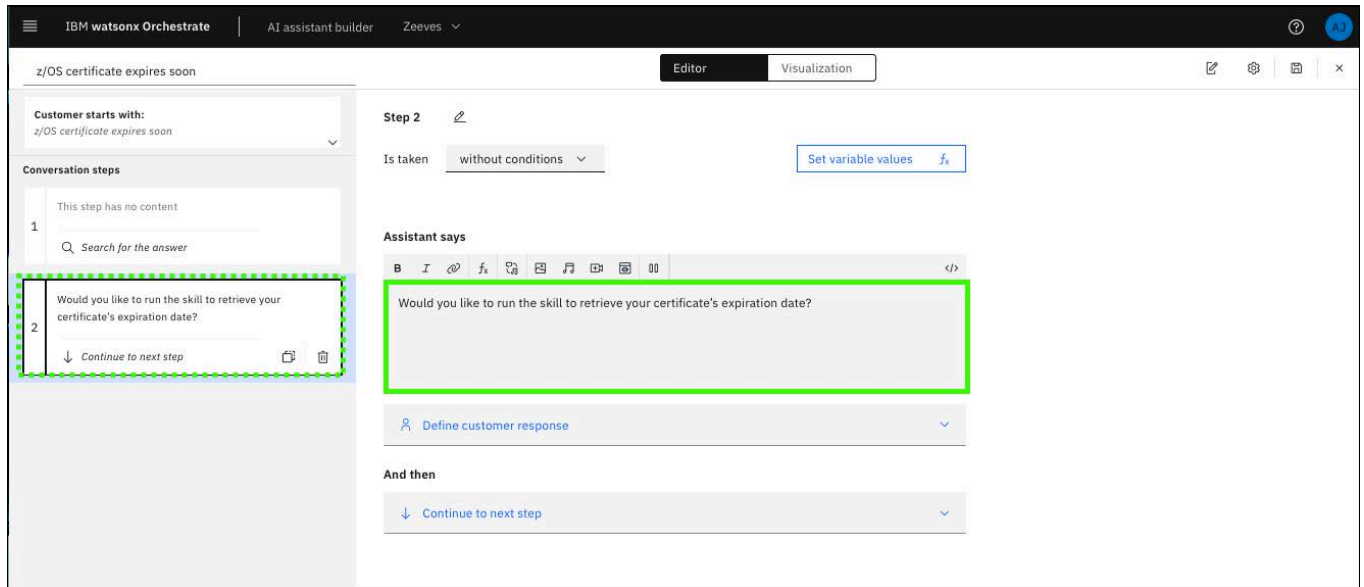
7. Click **Next step+**.



8. Enter the following response in the **Assistant says** field.

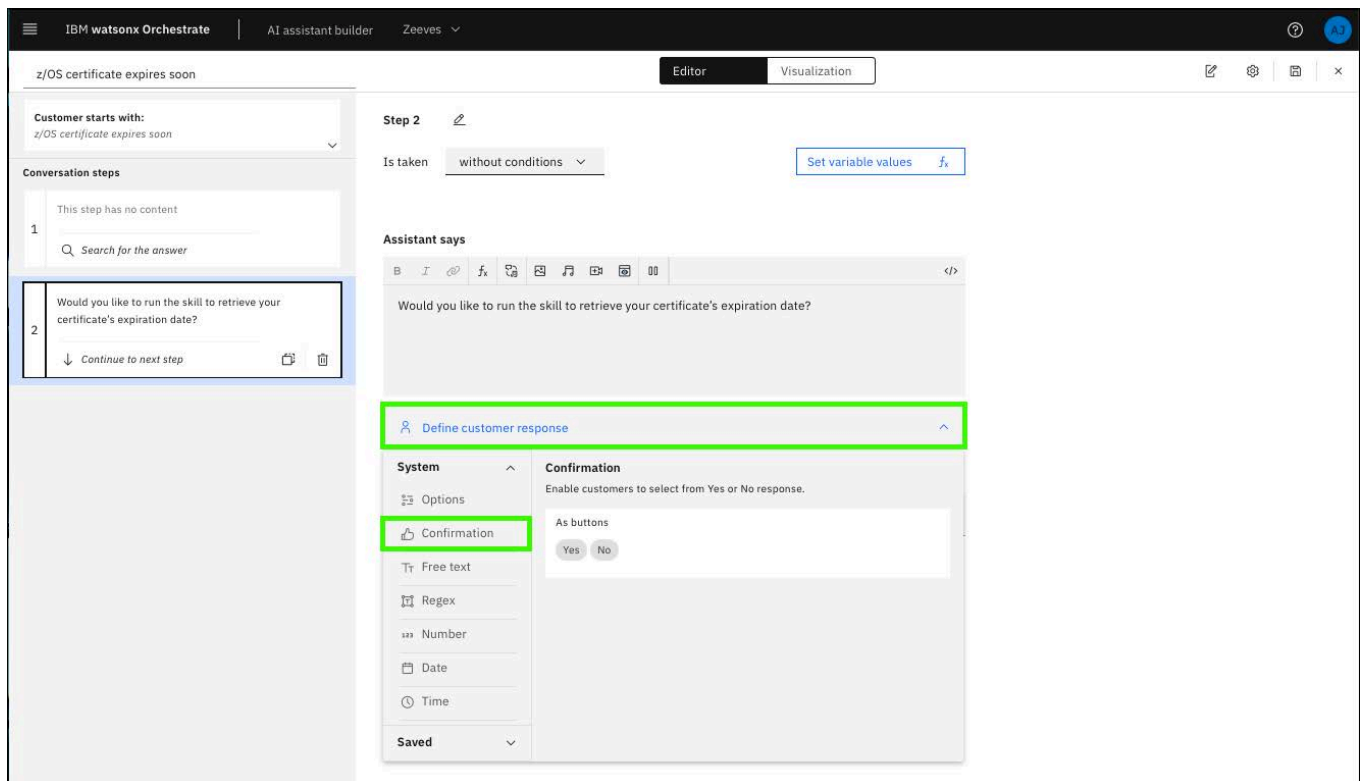
Assistant says:

Would you like to run the skill to retrieve your certificate's expiration date?

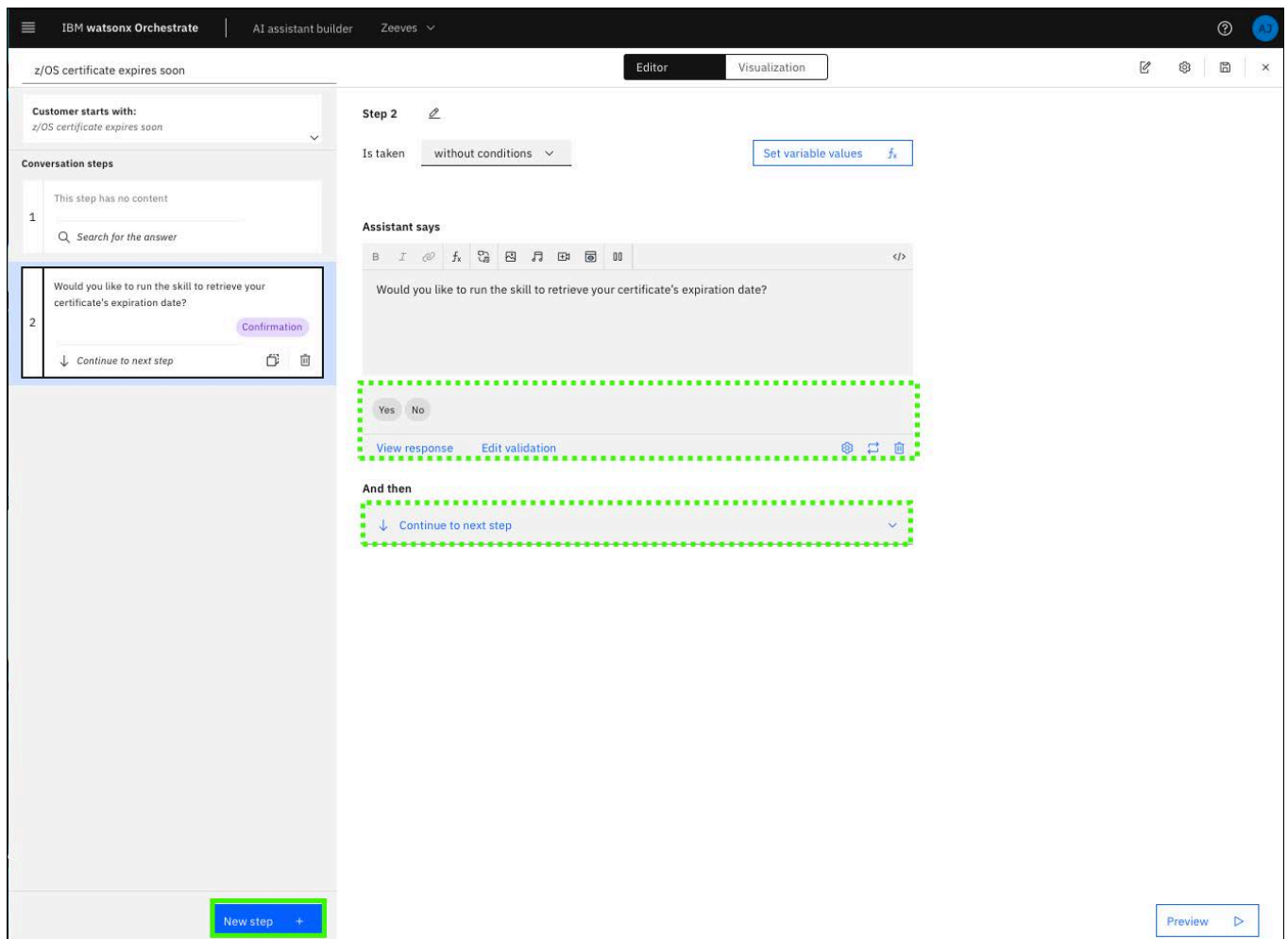


9. Click the **Define customer response** option list and select **Confirmation**.

The **Confirmation** option prompts the user to select **Yes** or **No**.

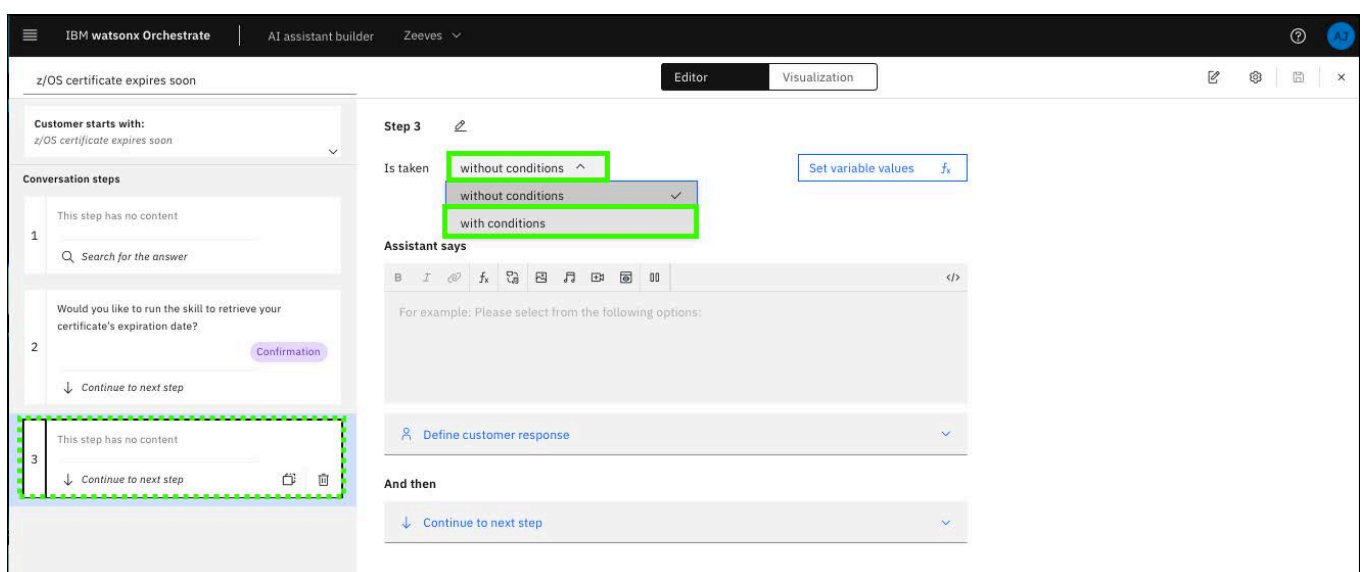


10. Click **Next step +**.



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.



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

Assistant says:

What is your certificate label?

The screenshot displays the IBM watsonx Orchestrate interface for creating a workflow. The top navigation bar includes the IBM logo, the product name 'watsonx Orchestrate', and tabs for 'AI assistant builder' and 'Zeeves'. The main workspace is divided into three sections:

- Left Sidebar (Conversation steps):** Lists the steps of the workflow. Step 1 is 'Search for the answer'. Step 2 is 'Confirmation' with a 'Continue to next step' button. Step 3 is 'What is your certificate label?' with a 'Continue to next step' button and a trash icon.
- Main Editor:**
 - Step 3:** The step is titled 'Is taken' with a dropdown menu set to 'with conditions'. A 'Set variable values' button is visible.
 - Conditions:** A single condition is defined: 'If All of this is true:'. The condition is '2. Would you like to r... is Yes'. A 'New condition group' button is at the bottom.
 - Assistant says:** A text box contains the prompt 'What is your certificate label?'.
 - Define customer response:** A dropdown menu is set to 'Continue to next step'.
- Right Sidebar (Set variable values):** A panel for defining variable values, currently empty.

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

The screenshot shows the IBM watsonx Assistant for Z interface. The top bar indicates the project name 'z/OS certificate expires soon' and the current view is 'Editor'. The left sidebar shows the 'Conversation steps' list with three steps. Step 3 is highlighted with a green dashed border. The main editor area shows the configuration for Step 3, including conditions and assistant responses. The 'Define customer response' section is highlighted with a green border. The right sidebar shows the 'System' settings, including 'Free text' and 'Options'.

Customer starts with:
z/OS certificate expires soon

Conversation steps

- 1 This step has no content
Search for the answer
- 2 Would you like to run the skill to retrieve your certificate's expiration date?
Confirmation
Continue to next step
- 3 What is your certificate label?
Continue to next step

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

What is your certificate label?

[Define customer response](#)

System

- Options
- Confirmation
- Free text**
- Regex
- Number
- Date
- Time

Free text

Enable customers to write out any response using text input

As text

My street is Main Street

[New step +](#)

[Preview](#)

14. Click **Next step+**.

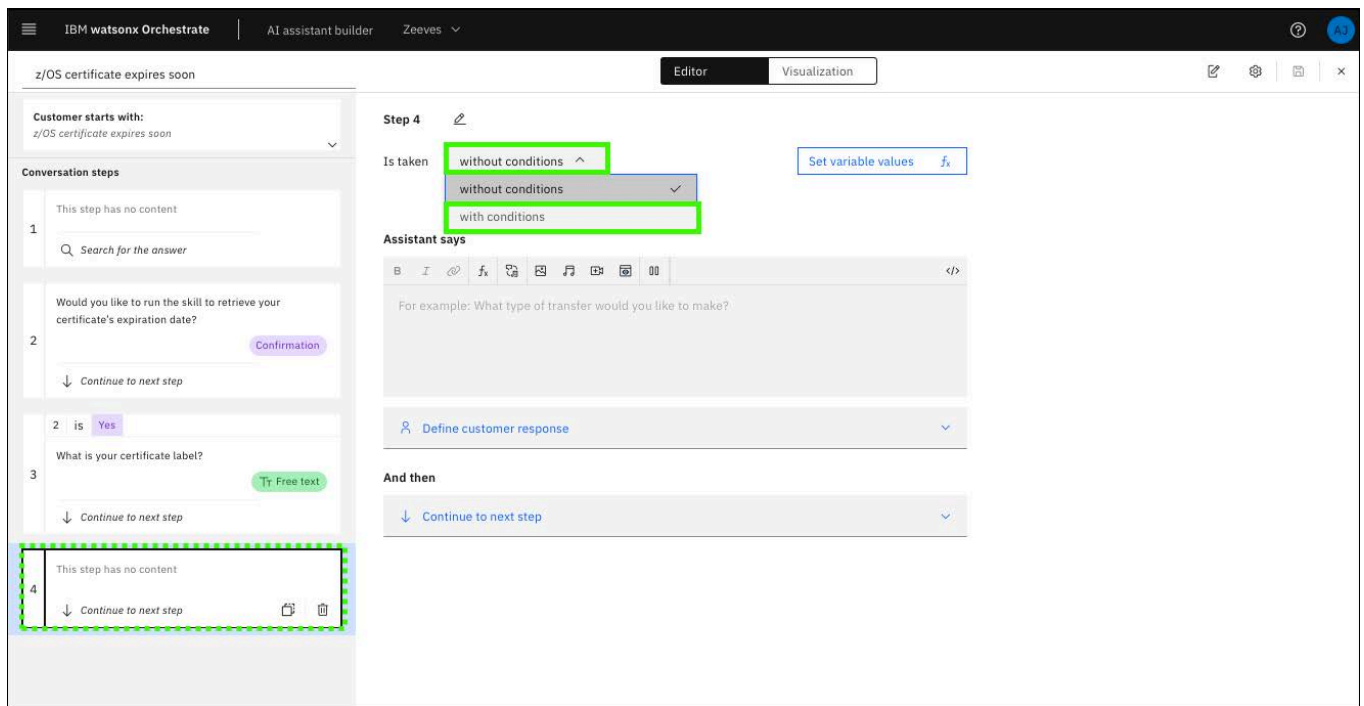
The screenshot displays the IBM watsonx Assistant Builder interface. The top navigation bar shows 'IBM watsonx Orchestrate' and 'AI assistant builder'. The main workspace is divided into three panels:

- Left Panel (Conversation steps):** Shows a list of steps. Step 3 is highlighted, showing its content: 'What is your certificate label?' with a 'Free text' input type and a 'Continue to next step' button.
- Center Panel (Step 3 configuration):**
 - Is taken:** Set to 'with conditions'.
 - Conditions:** A single condition is defined: 'If All of this is true: 2. Would you like to r... is Yes'.
 - Assistant says:** A text prompt 'What is your certificate label?' is shown, followed by a 'User enters free text' input field.
 - And then:** A 'Continue to next step' button is configured.
- Right Panel (Tools):** Includes a 'Set variable values' button and a 'New condition group' button.

At the bottom of the interface, there is a 'New step +' button on the left and a 'Preview' button on the right.

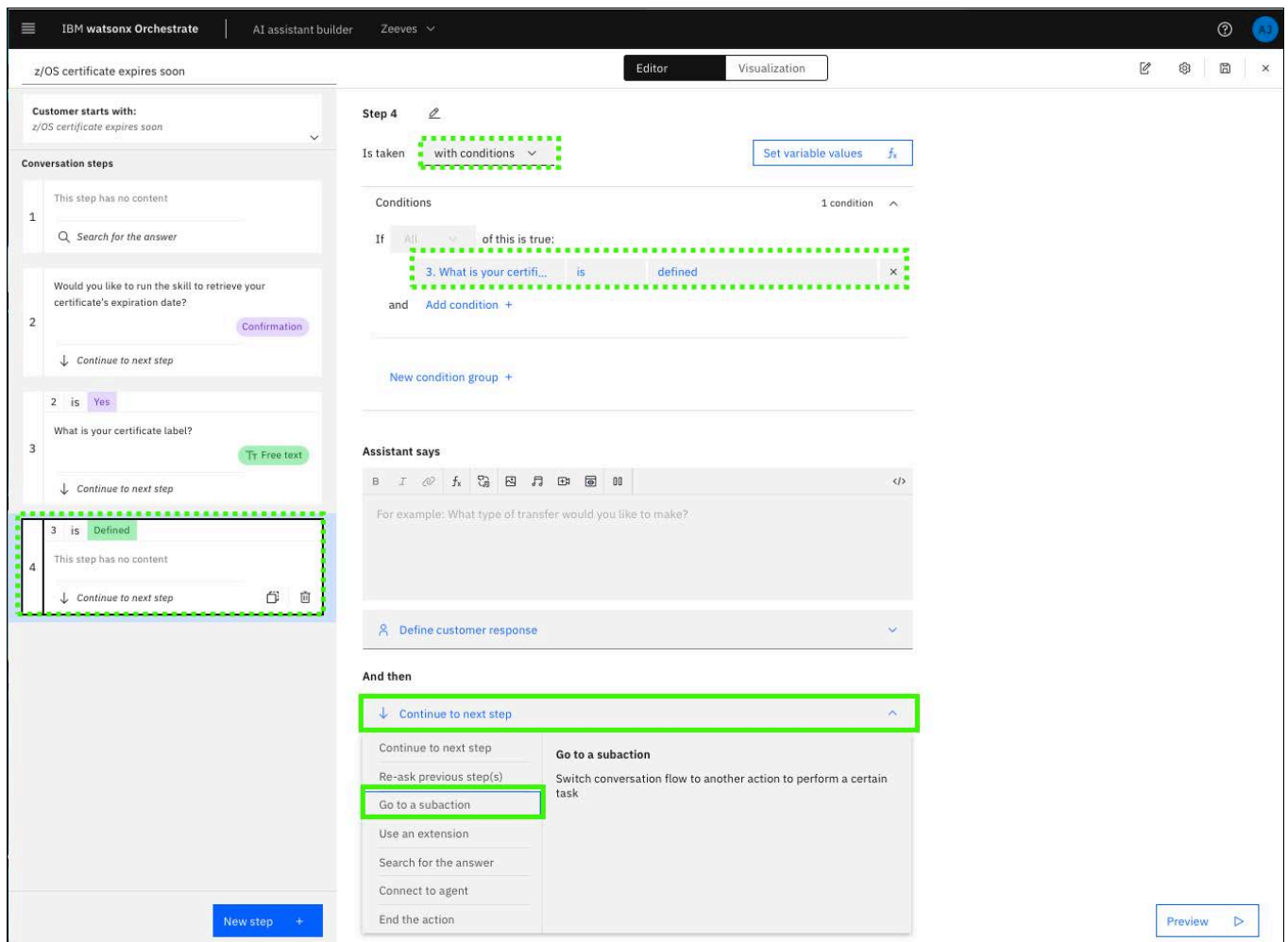
15. Click the **Is taken** option list and select **with conditions**.

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.

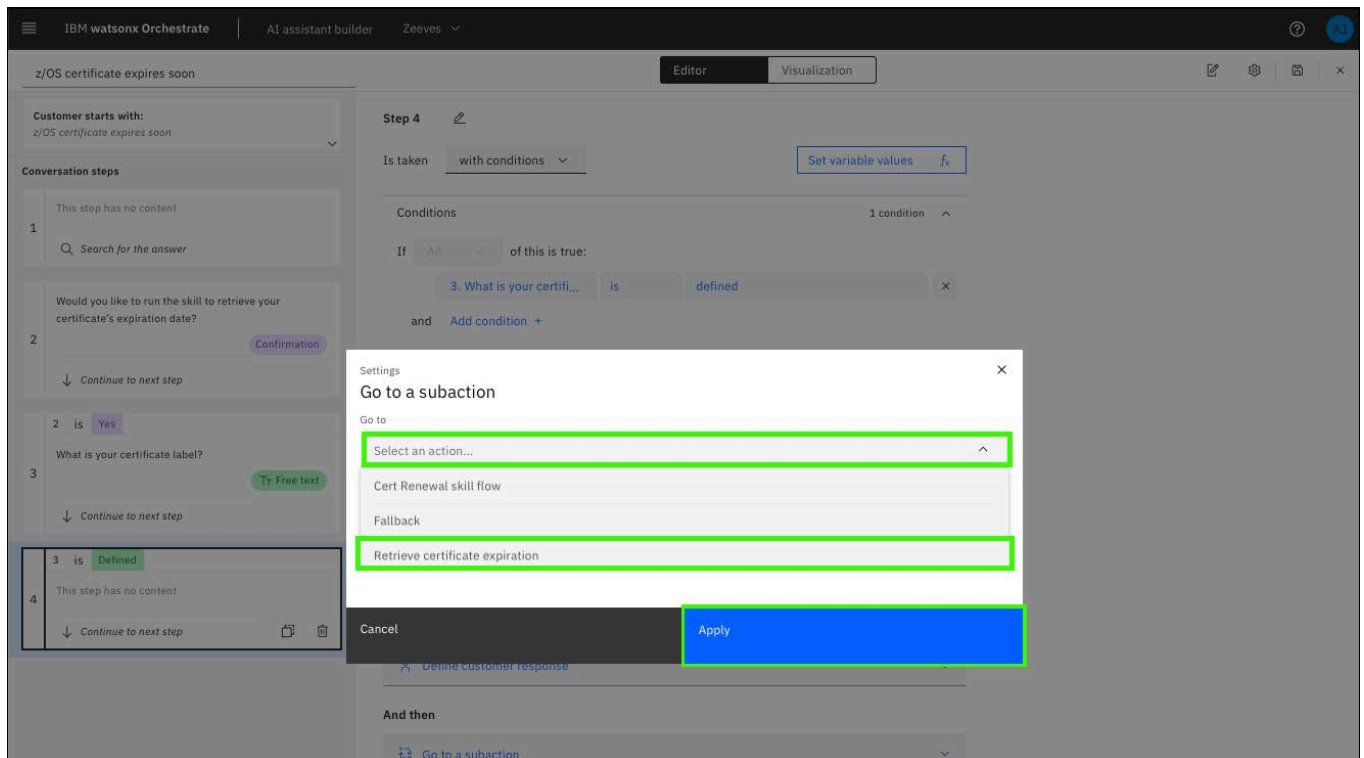


16. Click the **And then** option list and click **Go to a subaction**.

Notice that the default condition validates the *free text* is defined from the previous step.



17. Click the (a) **Go to** option list, select the (b) **Retrieve certificate expiration** skill-based action, and then click (c) **Apply**.



18. Click **Edit passed values**.

To run the **Retrieve certificate expiration** subaction that uses the users certificate label, the passed value needs to be modified.

IBM watsonx Orchestrator | AI assistant builder | Zeeves

z/OS certificate expires soon

Editor Visualization

Customer starts with: z/OS certificate expires soon

Conversation steps

1. This step has no content. Search for the answer.
2. 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. Go to subaction: Retrieve certifica...

Step 4

Is taken with conditions Set variable values f_x

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

- Go to a subaction
- Goes to action Retrieve certificate expiration
- Pass values
- Upon return Continue
- Edit settings Edit passed values

New step +

Preview

19. Click **Set new value +** and then select **extra_vars.cert_label_survey**.

IBM watsonx Orchestrator | AI assistant builder | Zeeves

z/OS certificate expires soon

Editor Visualization

Customer starts with: z/OS certificate expires soon

Conversation steps

1. This step has no content. Search for the answer.
2. 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. Go to subaction: Retrieve certifica...

Step 4

Is taken with conditions Set variable values f_x

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

- Go to a subaction
- Goes to action Retrieve certificate expiration
- Pass values
- Upon return Continue
- Edit settings Edit passed values

New step +

Preview

Passed values

Edit passed values

Configure how your skill-based action leverages and stores information in assistant variables. [Learn more](#)

Variables are not private if they are shared with Skill Studio.

Edit variable values Response behavior

The values you select from this action will be passed to "Retrieve certificate expiration" skill.

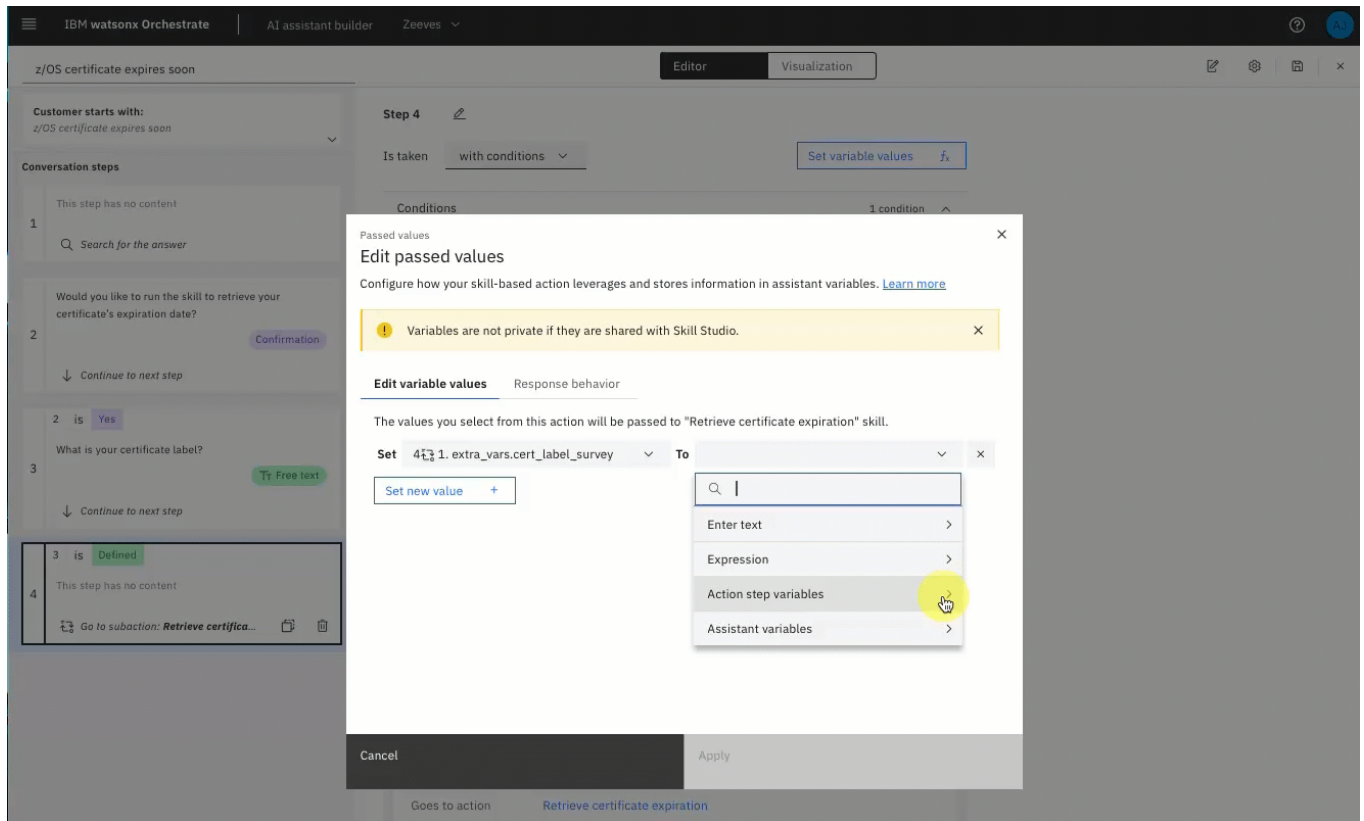
Set new value +

- 1. extra_vars.cert_label_survey
- 2. extra_vars.cert_type_survey
- 3. extra_vars.owner_id_survey

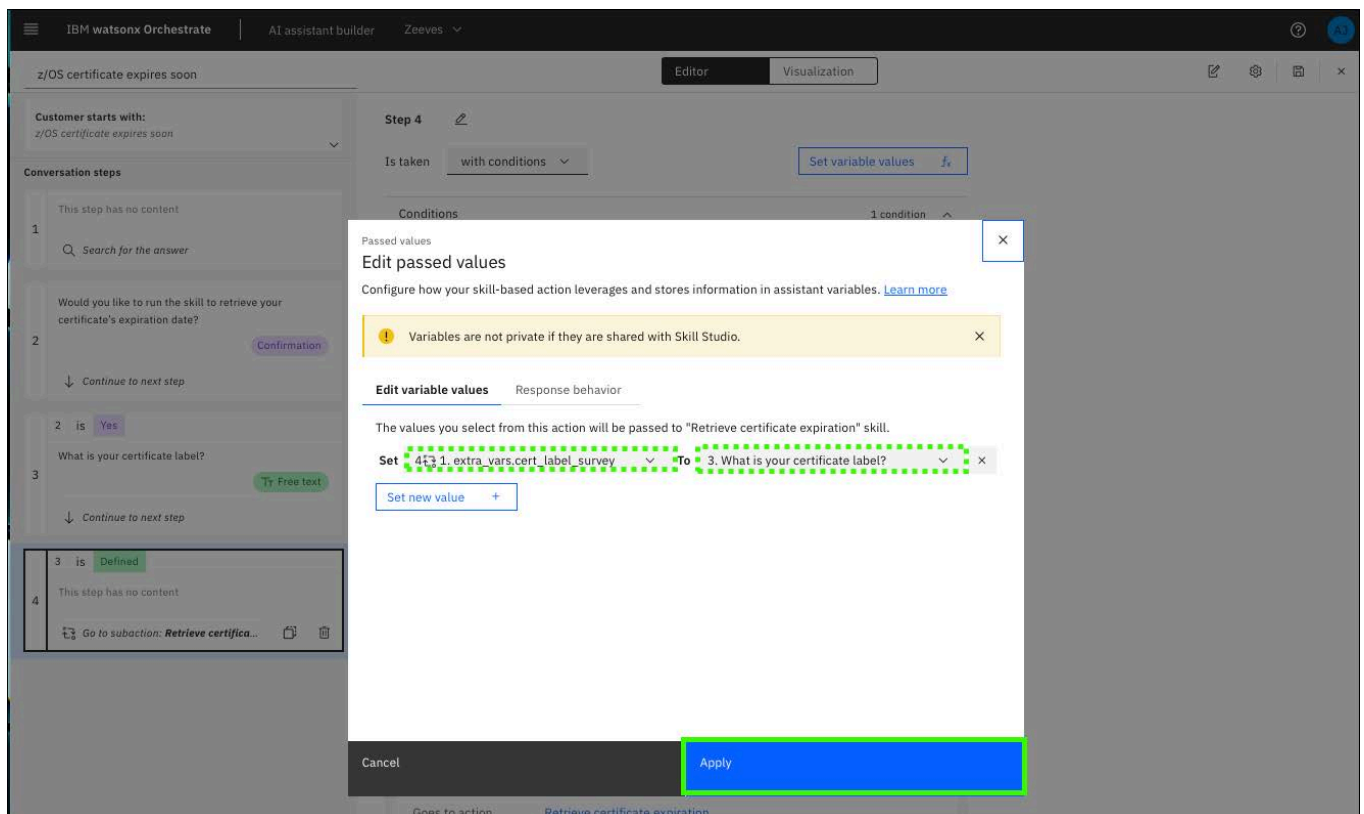
Cancel Apply

Goes to action Retrieve certificate expiration

20. In the **To** field, select **Action step variables**, and then select **What is your certificate label?**.



21. Click **Apply**.



22. Click **Next step +**.

The screenshot shows the IBM watsonx Assistant Editor interface. The top bar indicates the project is 'z/OS certificate expires soon'. The left sidebar shows the conversation steps, with Step 4 highlighted. The main editor area for Step 4 shows the 'Is taken' dropdown set to 'with conditions'. The conditions section shows a single condition: '3. What is your certifi...' is defined. The 'Assistant says' section shows a text input field. The 'And then' section shows the action 'Go to a subaction' leading to 'Retrieve certificate expiration', with the pass values set to '1. extra_vars.cert_label_survey' and the return action set to '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.

IBM watsonx Orchestrate | AI assistant builder | Zeeves

z/OS certificate expires soon

Editor Visualization

Customer starts with:
z/OS certificate expires soon

Conversation steps

1 This step has no content
Search for the answer

2 Would you like to run the skill to retrieve your certificate's expiration date?
Confirmation
Continue to next step

2 is Yes

3 What is your certificate label?
Free text
Continue to next step

3 is Defined

4 This step has no content
Go to subaction: Retrieve certifica...

5 This step has no content
Continue to next step

Step 5

Is taken
without conditions
without conditions
with conditions

Set variable values

Assistant says

For example: What size do you want to order?

Define customer response

And then

Continue to next step

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

Assistant says:

Below is your certificate's expiration date:

IBM watsonx Orchestrate | AI assistant builder | Zeeves

z/OS certificate expires soon

Editor Visualization

Customer starts with:
z/OS certificate expires soon

Conversation steps

1 This step has no content
Search for the answer

2 Would you like to run the skill to retrieve your certificate's expiration date?
Confirmation
Continue to next step

2 is Yes

3 What is your certificate label?
Free text
Continue to next step

3 is Defined

4 This step has no content
Go to subaction: Retrieve certifica...

5 Below is your certificate's expiration date:
Continue to next step

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:

Define customer response

And then

Continue to next step

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 displays the IBM watsonx Orchestrate AI assistant builder interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main workspace is divided into two panels: 'Editor' and 'Visualization'. The 'Editor' panel shows a conversation flow on the left and a step editor on the right. The conversation flow on the left lists five steps, with Step 5 highlighted. Step 5 is titled 'Below is your certificate's expiration date:
'. The step editor on the right shows the 'Assistant says' field, which contains the text 'Below is your certificate's expiration date:'. A green box highlights the '\$' symbol in the 'Assistant says' field, which has triggered a dropdown menu showing available functions: 'Action step variables', 'Retrieve certificate expiration (step 4)', 'Assistant variables', and 'Integration variables'.

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

IBM watsonx Orchestrate | AI assistant builder | Zeeves

z/OS certificate expires soon

Editor Visualization

Customer starts with: z/OS certificate expires soon

Conversation steps

1 This step has no content
Search for the answer

2 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?
Free text
Continue to next step

3 is Defined
This step has no content
Go to subaction: Retrieve certifica...

3 is Defined
Below is your certificate's expiration date:

Continue to next step

Step 5

Is taken with conditions Set variable values

Conditions 1 condition

If All of this is true:

3. What is your certi... is defined

and Add condition +

New condition group +

Assistant says

Below is your certificate's expiration date:

Action step variables >

Retrieve certificate expiration (step 4)

Assistant variables >

Integration variables >

Continue to next step

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

IBM watsonx Orchestrate | AI assistant builder | Zeeves

z/OS certificate expires soon

Editor Visualization

Customer starts with: z/OS certificate expires soon

Conversation steps

1 This step has no content
Search for the answer

2 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?
Free text
Continue to next step

3 is Defined
This step has no content
Go to subaction: Retrieve certifica...

3 is Defined
Below is your certificate's expiration date:

Continue to next step

Step 5

Is taken with conditions Set variable values

Conditions 1 condition

If All of this is true:

3. What is your certi... is defined

and Add condition +

New condition group +

Assistant says

Below is your certificate's expiration date:

1 4. Retrieve certificate expiration res

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 Orchestrator interface for configuring a skill named "z/OS certificate expires soon". The interface has a top navigation bar with "IBM watsonx Orchestrator", "AI assistant builder", and "Zeeves". Below the navigation bar, there are tabs for "Editor" and "Visualization".

On the left, a sidebar titled "Conversation steps" lists five steps. Step 5 is highlighted with a blue border. The steps are:

1. This step has no content. Search for the answer.
2. Would you like to run the skill to retrieve your certificate's expiration date? Confirmation. Continue to next step.
3. 2 is Yes. What is your certificate label? Tr Free text. Continue to next step.
4. 3 is Defined. This step has no content. Go to subaction: Retrieve certifica...
5. 3 is Defined. Below is your certificate's expiration date: 1 4. Retrieve certificate expiration res. Continue to next step.

The main "Editor" area shows the configuration for Step 5. It includes a "Conditions" section with a single condition: "If All of this is true: 3. What is your certifi... is defined". Below the conditions, there is a section for "Assistant says" which contains the text: "Below is your certificate's expiration date: 1 4. Retrieve certificate expiration res". At the bottom right of the editor, there is a "Preview" button with a play icon.

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 displays the IBM watsonx Assistant Editor interface for a workflow titled "z/OS certificate expires soon".

Left Sidebar (Conversation steps):

- Step 1: "This step has no content".
- Step 2: "Would you like to run the skill to retrieve your certificate's expiration date?". It includes a "Confirmation" button and a "Continue to next step" link.
- Step 3: "What is your certificate label?". It includes a "Try Free text" button and a "Continue to next step" link.
- Step 4: "This step has no content". It includes a "Go to subaction: Retrieve certifica..." link.
- Step 5: "Below is your certificate's expiration date:". It includes a "Continue to next step" link and a "New step +" button.

Central Editor (Step 5):

- Is taken:** "with conditions".
- Conditions:** "1 condition".
- If:** "All" of this is true:
 - Condition 1: "3. What is your certifi..." is defined.
- and:** "Add condition +".
- Assistant says:** "Below is your certificate's expiration date: 1. 4. Retrieve certificate expiration res:".
- And then:** "Continue to next step".

Right Preview Pane:

- Preview:** Shows a chat interface with a "Greet customer [default]" button and a "Welcome, how can I assist you?" message.
- Input:** "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?".

z/OS certificate expires soon

Customer starts with:
z/OS certificate expires soon

Conversation steps

- This step has no content
Search for the answer
- Would you like to run the skill to retrieve your certificate's expiration date?
Confirmation
Continue to next step
- 2 is Yes
What is your certificate label?
Free text
Continue to next step
- 3 is Defined
This step has no content
Go to subaction: Retrieve certifica...
- 3 is Defined
Below is your certificate's expiration date:
1 4. Retrieve certificate expiration re:
Continue to next step

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

And then

Continue to next step

Preview

10:06 AM
Conversational search called

Search info Inspect

To retrieve the expiration date for your z/OS certificate, you would typically check the certificate itself or consult the system logs. However, based on the provided context, there isn't a direct command or utility mentioned for this purpose. You may need to contact your system administrator or refer to your specific system's documentation for precise instructions. Generally, you can use commands like 'ODTE' or 'RACDCERT' to list and inspect certificates.

Accuracy of generated answers may vary.

Would you like to run the skill to retrieve your certificate's expiration date?

Yes No

Type something...

4. Review the response and enter DEMOCERT .

Prompt:

DEMOCERT

The screenshot displays the IBM watsonx Assistant for Z Level 4 Lab Guide interface. The main window is titled "z/OS certificate expires soon" and shows a conversation flow with five steps. Step 5 is currently selected and being edited.

Conversation steps:

- Step 1: This step has no content.
- 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.
- Step 5: Below is your certificate's expiration date: 1. 4. Retrieve certificate expiration res: (Step 4)

Step 5 configuration:

- Is taken:** with conditions
- 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 res:
 - Define customer response
- And then:** Continue to next step

Preview:

The preview shows a chat interface with the following content:

- AI: certificates; you would typically check the certificate itself or consult the system logs. However, based on the provided context, there isn't a direct command or utility mentioned for this purpose. You may need to contact your system administrator or refer to your specific system's documentation for precise instructions. Generally, you can use commands like 'ODTE' or 'RACDCERT' to list and inspect certificates.
- AI: Accuracy of generated answers may vary.
- User: Would you like to run the skill to retrieve your certificate's expiration date?
- User: Yes
- AI: 10:07 AM
- AI: z/OS certificate expires soon recognized
- User: What is your certificate label?
- User: DEMOCERT

5. Click **Apply**.

The screenshot displays the IBM watsonx Assistant Editor interface. The top navigation bar shows 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main workspace is titled 'z/OS certificate expires soon' and has tabs for 'Editor' and 'Visualization'. The left sidebar shows a list of conversation steps, with Step 5 selected. The central editor for Step 5 shows a condition '3. What is your certifi... is defined' and an assistant response 'Below is your certificate's expiration date: 11:4. Retrieve certificate expiration res:'. The right preview pane shows a chat interaction where the user asks 'What is your certificate label?' and the assistant responds with the expiration date and a link to 'Retrieve certificate expiration'.

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 displays the IBM watsonx Assistant for Z Level 4 Lab Guide interface. The main workspace is titled "z/OS certificate expires soon" and is in the "Editor" mode. The interface is divided into several sections:

- Customer starts with:** A dropdown menu showing "z/OS certificate expires soon".
- Conversation steps:** A list of steps on the left. Step 5 is highlighted, showing a condition "3 is Defined".
- Step 5 configuration:**
 - Is taken:** A dropdown menu set to "with conditions".
 - Conditions:** A section showing a single condition: "3. What is your certifi... is defined".
 - Assistant says:** A text area containing "Below is your certificate's expiration date: 1. 4. Retrieve certificate expiration res".
 - And then:** A dropdown menu set to "Continue to next step".
- Preview:** A panel on the right showing a simulated conversation. It includes a message from the user: "Below is your certificate's expiration date: {'certificate expiration date': '2030/01/30 23:59:59'}".

A "New step" button is visible at the bottom left of the interface.

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

The screenshot shows the IBM watsonx Assistant Editor interface. The top bar includes the logo, 'AI assistant builder', and 'Zeeves'. The main workspace is divided into two tabs: 'Editor' and 'Visualization'. The left sidebar shows a list of conversation steps for the skill 'z/OS certificate expires soon'. The main area displays the details for 'Step 1'. The 'Is taken' section has three options: 'without conditions' (selected), 'without conditions' (checked), and 'with conditions'. The 'Assistant says' field is currently empty, with a placeholder text 'For example: Please select from the following options:'. Below this, there is a 'Define customer response' button. The 'And then' section has a 'Continue to next step' button. The bottom of the screen shows a preview of the conversation flow, with step 6 highlighted.

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

Assistant says:

Would you like to renew your certificate?

The screenshot displays the IBM watsonx Assistant for Z Level 4 Lab Guide interface. The top navigation bar shows 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main title is 'z/OS certificate expires soon'. The interface is split into three panels:

- Conversation steps:** A list of steps for the skill. Step 1 is 'Would you like to renew your certificate?'. Step 2 is 'Would you like to run the skill to retrieve your certificate's expiration date?'. Step 3 is 'What is your certificate label?'. Step 4 is 'This step has no content'. Step 5 is 'Below is your certificate's expiration date:'. Step 6 is 'Would you like to renew your certificate?'. Step 6 is highlighted with a green dashed border.
- Step 1 configuration:** The configuration for the first step. It shows 'Is taken' as 'with conditions'. The conditions are '1 Retrieve certifica...' is 'defined'. The 'Assistant says' section shows the text 'Would you like to renew your certificate?'. The 'Define customer response' dropdown is set to 'Confirmation'.
- Assistant says:** A text area for the assistant's response, currently containing 'Would you like to renew your certificate?'.

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

The screenshot displays the IBM watsonx Orchestrate AI assistant builder interface. The top navigation bar includes the IBM watsonx Orchestrate logo, 'AI assistant builder', and a user profile 'Zeeves'. The main workspace is titled 'z/OS certificate expires soon' and has tabs for 'Editor' and 'Visualization'.

Conversation steps:

- Step 1:** 'Would you like to renew your certificate?' (Status: Defined). It includes a 'Continue to next step' button.
- Step 2:** 'Would you like to run the skill to retrieve your certificate's expiration date?' (Status: Confirmation). It includes a 'Continue to next step' button.
- Step 3:** 'What is your certificate label?' (Status: Yes). It includes a 'Continue to next step' button.
- Step 4:** 'This step has no content' (Status: Defined). It includes a 'Go to subaction: Retrieve certifica...' button.
- Step 5:** 'Below is your certificate's expiration date:' (Status: Defined). It includes a 'Continue to next step' button.
- Step 6:** 'Would you like to renew your certificate?' (Status: Defined). It includes a 'Continue to next step' button.

Step 1 configuration:

- Is taken:** with conditions
- Conditions:** 1 condition
 - If All of this is true:
 - 1 Retrieve certifica... is defined
 - and Add condition +
 - New condition group +

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 displays the IBM watsonx Assistant Editor interface. The top navigation bar shows 'IBM watsonx Orchestrate' and 'AI assistant builder'. The main workspace is titled 'z/OS certificate expires soon'. On the left, the 'Conversation steps' list shows a sequence of steps, with step 1 highlighted. The right panel, 'Step 1', is in 'Editor' mode. It shows a condition '1 Retrieve certifica... is defined' and an 'Assistant says' section with a 'Would you like to renew your certificate?' prompt and a 'Yes' button highlighted in green. The 'Assistant says' section also includes a 'View response' button and an 'Edit validation' button. The 'And then' section shows a 'Continue to next step' button. A 'New step' button is visible at the bottom left, and a 'Preview' button is at the bottom right.

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 displays the IBM watsonx Orchestrator AI assistant builder interface. The left pane shows a workflow for "z/OS certificate expires soon" with steps 1 through 7. Step 7 is highlighted with a green dashed border. The right pane shows the configuration for Step 1, including "Is taken" (with conditions), "Condition" (with conditions), and "Assistant says" field.

Step 1 Configuration:

- Is taken:** with conditions (selected)
- Condition:** with conditions (selected)
- If:** All of this is true:
 - 1 Retrieve certific... is defined
- Assistant says:** Would you like to renew your certificate?
- Buttons:** Yes, No
- 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 displays the IBM watsonx Orchestrate AI assistant builder interface. The top navigation bar includes the IBM watsonx Orchestrate logo, the AI assistant builder tab, and the Zeeves dropdown menu. The main workspace is divided into two panels: a left sidebar showing a sequence of steps (1-7) and a right panel for configuring the current step (Step 7).

Step 7 Configuration:

- Is taken:** Set to "with conditions".
- Conditions:** A single condition is defined: "If All of this is true: 6. Would you like to r... is Yes".
- Assistant says:** A text box containing the prompt: "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:** A dropdown menu is visible.
- And then:** A dropdown menu is set to "Continue to next step".

Step 7 Preview:

The preview shows a step card for Step 7 with the text: "What date would you like to set the renewed certificate's expiration date to? Please enter in the...". The card includes a "Continue to next step" button and a "Free text" input type indicator.

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

The screenshot displays the IBM watsonx Assistant Editor interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main workspace is titled 'z/OS certificate expires soon' and is in 'Editor' mode. The workflow is shown on the left, with steps 1 through 7. Step 7 is highlighted with a green dashed border. The right panel shows the configuration for Step 7, including a condition '6. Would you like to r...' and a 'Free text' response type. The 'Assistant says' section shows a prompt for the user to enter a date.

Step 7 Configuration:

- Is taken:** with conditions
- Conditions:** 1 condition
 - If All of this is true:
 - 6. Would you like to r... is Yes
- 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:** Free text
- Free text:** Enable customers to write out any response using text input. As text: My street is Main Street

9. Click **New step +**.

The screenshot shows the IBM watsonx Orchestrate AI assistant builder interface. The top bar indicates the skill name 'z/OS certificate expires soon' and the current view is 'Editor'. The left pane shows a list of steps, with Step 7 highlighted. The right pane shows the configuration for Step 7, including conditions and the assistant's response. The conditions are set to 'Is taken with conditions' and 'If All of this is true: 6. Would you like to r... is Yes'. The assistant's response is 'What date would you like to set the renewed certificate's expiration date to? Please enter in the form of YYYY-MM-DD.' and the 'And then' action is 'Continue to next step'.

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 displays the IBM watsonx Orchestrate AI assistant builder interface. The left pane shows a flowchart for the skill 'z/OS certificate expires soon'. The flowchart includes steps 2 through 8. Step 8 is highlighted with a green dashed border. The right pane shows the configuration for Step 8. The 'Is taken' section has three conditions: 'without conditions', 'without conditions', and 'with conditions'. The 'Assistant says' field contains the text: 'For example: What type of transfer would you like to make?'. The 'And then' section has a single action: 'Continue to next step'.

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

This assistant first responds with the message that follows before triggering the certificate renewal skill-flow. When performing a demo of this use case, mention the **z/OS Certs – Search and Renew** Ansible playbook typically takes a minute or so to complete.

Assistant says:

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

The screenshot displays the IBM watsonx Orchestrate AI assistant builder interface. The top navigation bar includes the IBM watsonx Orchestrate logo, the AI assistant builder section, and the user 'Zeeves'. The main workspace is divided into two panels: 'Editor' and 'Visualization'.

Left Panel (Workflow Steps):

- Step 2:** Confirmation step with the prompt 'Continue to next step'.
- Step 3:** Free text input step with the prompt 'What is your certificate label?'.
- Step 4:** Confirmation step with the prompt 'Continue to next step'.
- Step 5:** Free text input step with the prompt 'What date would you like to set the renewed certificate's expiration date to? Please enter in the...'.
- Step 6:** Confirmation step with the prompt 'Continue to next step'.
- Step 7:** Free text input step with the prompt 'What date would you like to set the renewed certificate's expiration date to? Please enter in the...'.
- Step 8:** Confirmation step with the prompt 'Continue to next step'.

Right Panel (Step 8 Configuration):

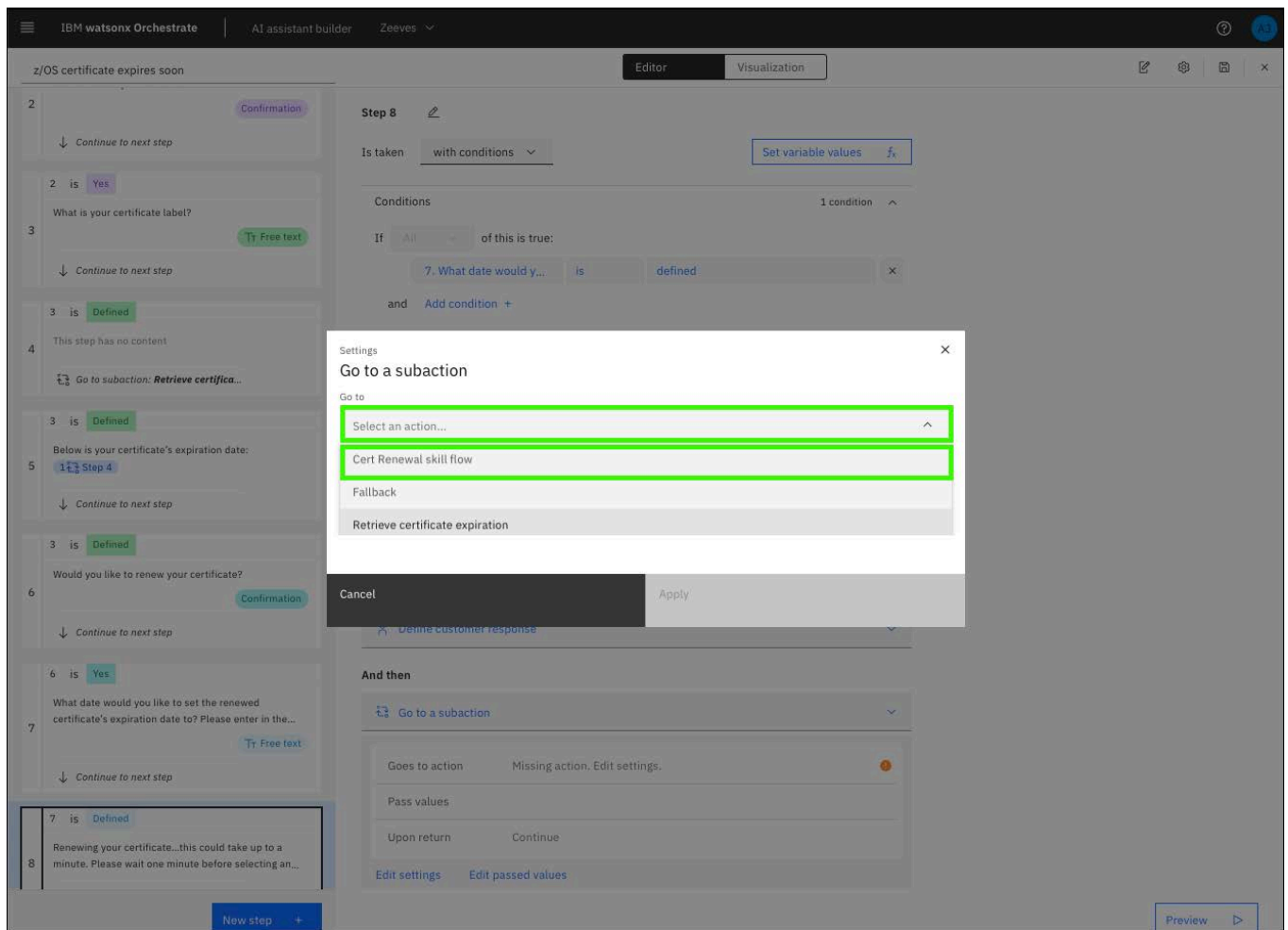
- Is taken:** 'with conditions' (highlighted with a green dashed box).
- Conditions:** 1 condition.
- If:** 'All' of this is true:
 - Condition 1: '7. What date would y...' is defined (highlighted with a green dashed box).
- and:** 'Add condition +'
- Assistant says:** A text box containing the message: 'Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.' (highlighted with a green dashed box).
- Define customer response:** A dropdown menu.
- And then:** A dropdown menu with the option 'Continue to next step'.

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

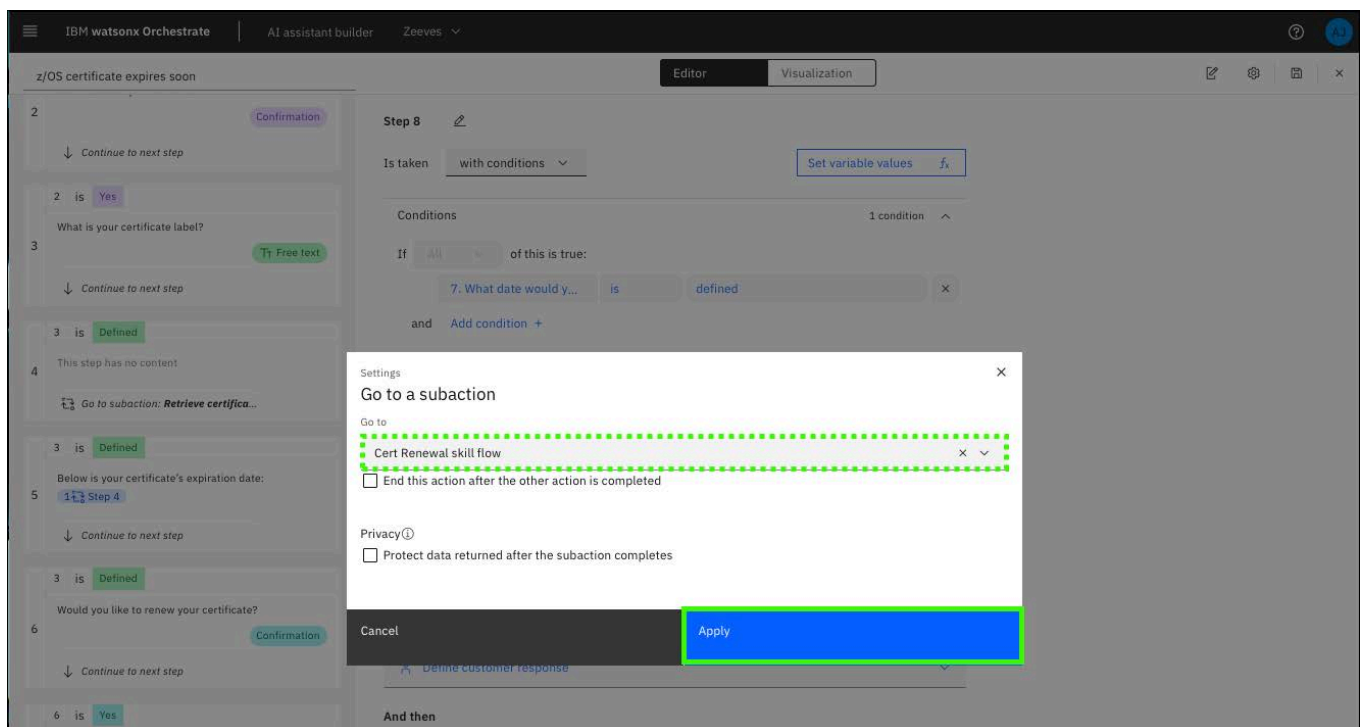
The screenshot displays the IBM watsonx Assistant Editor interface for a workflow titled "z/OS certificate expires soon". The interface is divided into three main sections:

- Left Sidebar (Step List):** Shows a sequence of steps. Step 8 is highlighted, indicating the current step being edited.
- Main Editor (Step 8 Configuration):**
 - Is taken:** Set to "with conditions".
 - Conditions:** A single condition is defined: "7. What date would y... is defined".
 - Assistant says:** A text area containing the message: "Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below."
 - And then:** A list of actions. The "Go to a subaction" option is highlighted with a green box.
- Right Sidebar (Preview):** A "Preview" button with a play icon.

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



14. Click **Apply**.



15. Click **Edit passed values**.

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

IBM watsonx Orchestrate | AI assistant builder | Zeeves

z/OS certificate expires soon

Editor Visualization

Step 8

Is taken with conditions Set variable values

Conditions 1 condition

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

Upon return Continue

Edit settings Edit passed values

16. Click **Set new value +** and then select **extra_vars.cert_label_survey**.

IBM watsonx Orchestrate | AI assistant builder | Zeeves

z/OS certificate expires soon

Editor Visualization

Step 8

Is taken with conditions Set variable values

Conditions 1 condition

Passed values

Edit passed values

Configure how your skill-based action leverages and stores information in assistant variables. [Learn more](#)

Variables are not private if they are shared with Skill Studio.

Edit variable values Response behavior

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

Set new value +

1. extra_vars.cert_label_survey

2. extra_vars.cert_type_survey

3. extra_vars.keyring_survey

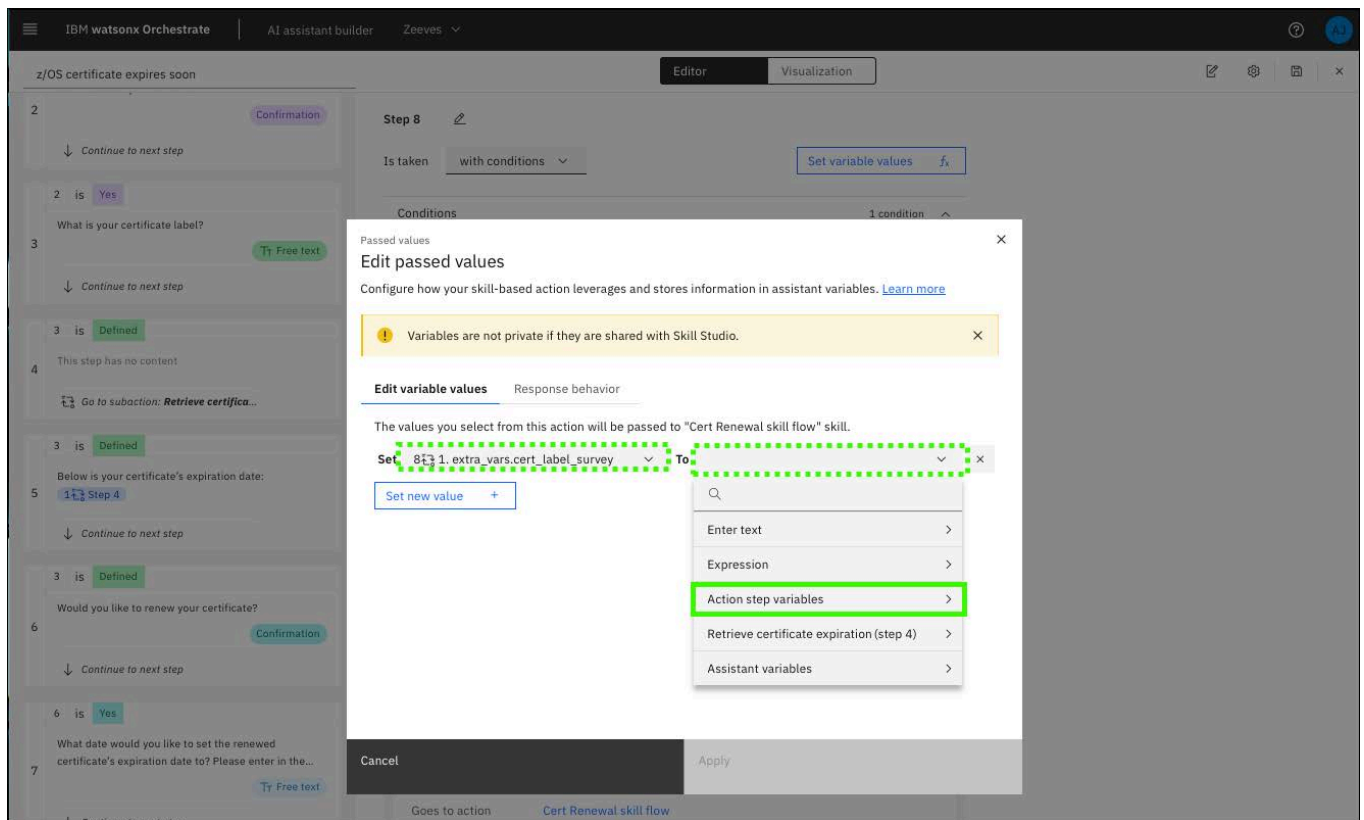
4. extra_vars.new_expiry_date_survey

5. extra_vars.owner_id_survey

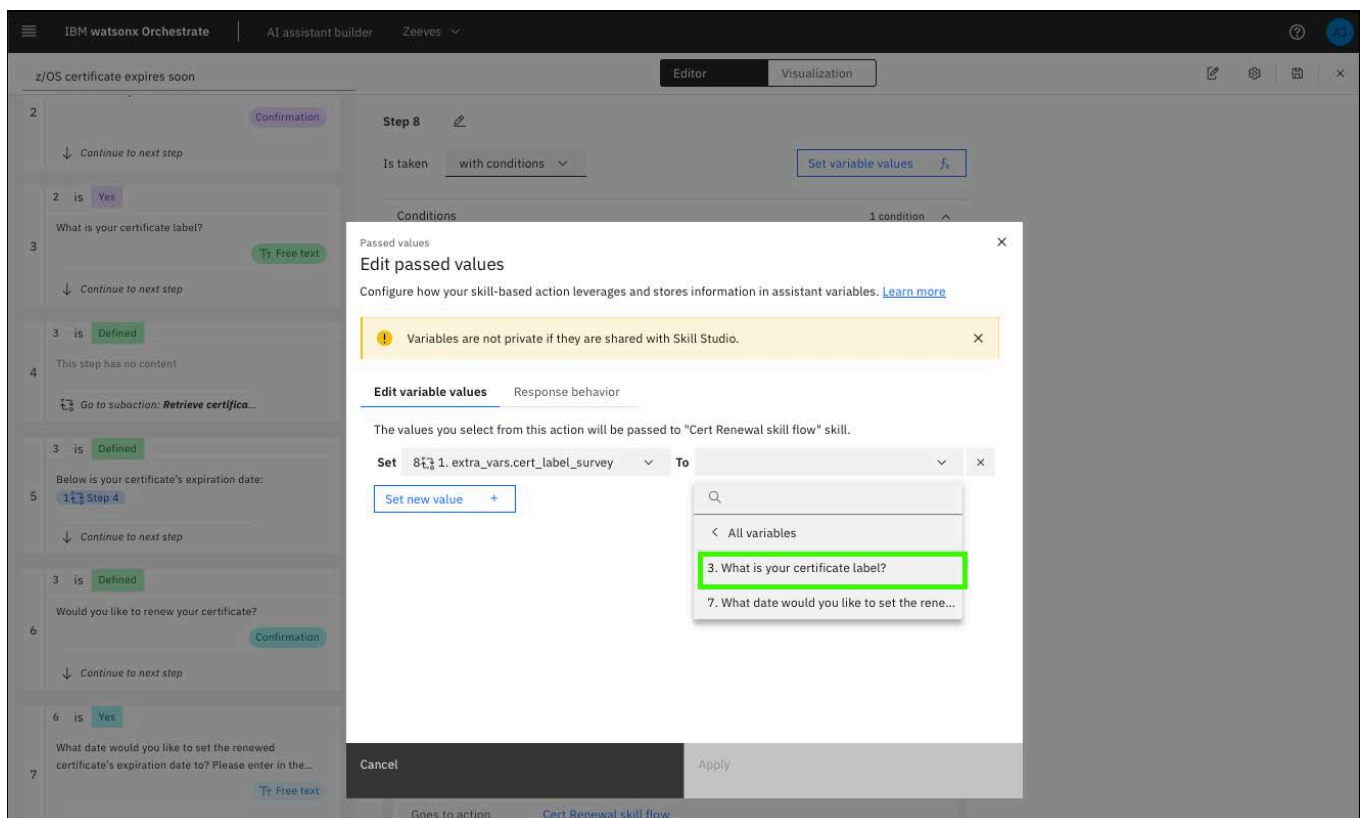
Cancel Apply

Goes to action Cert Renewal skill flow

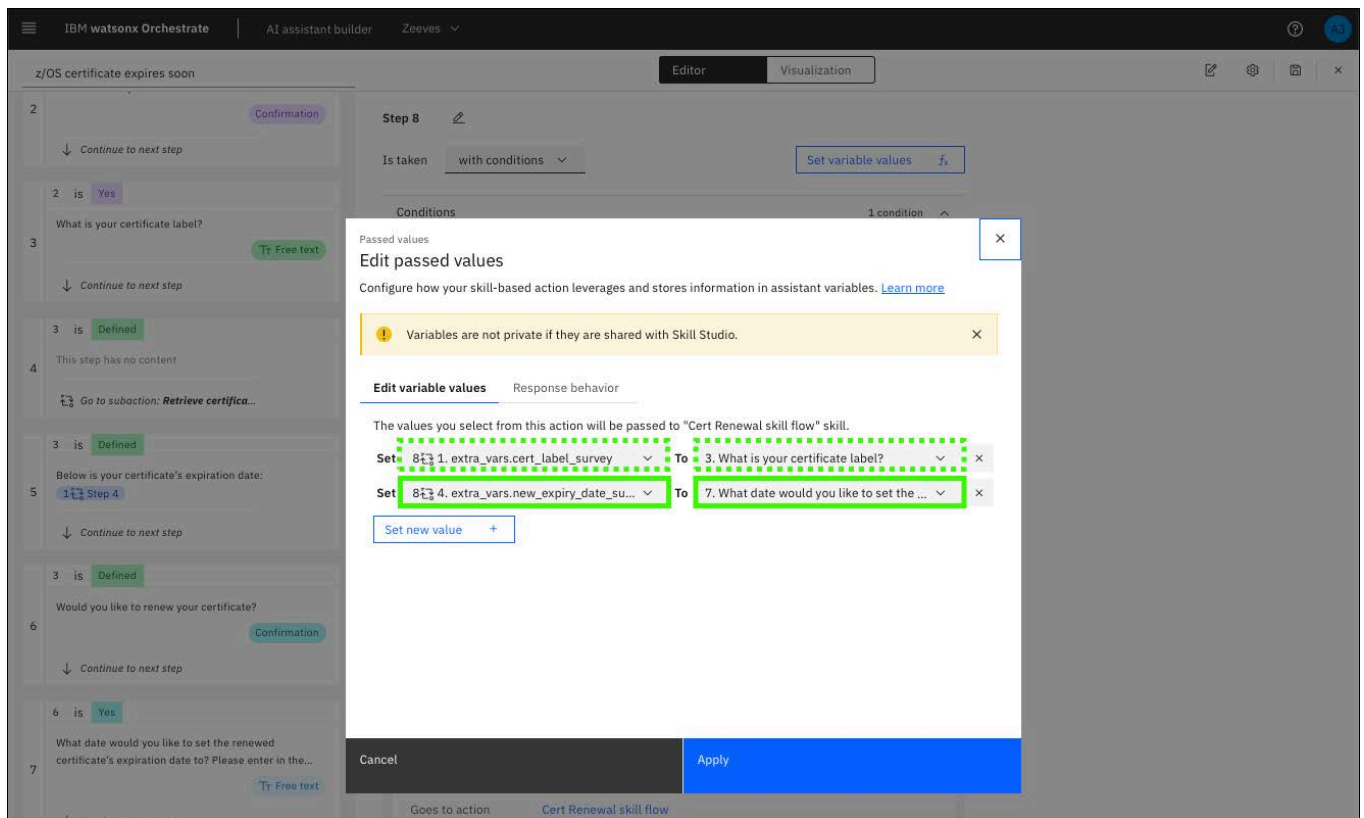
17. In the **To** field, select **Action step variables**.



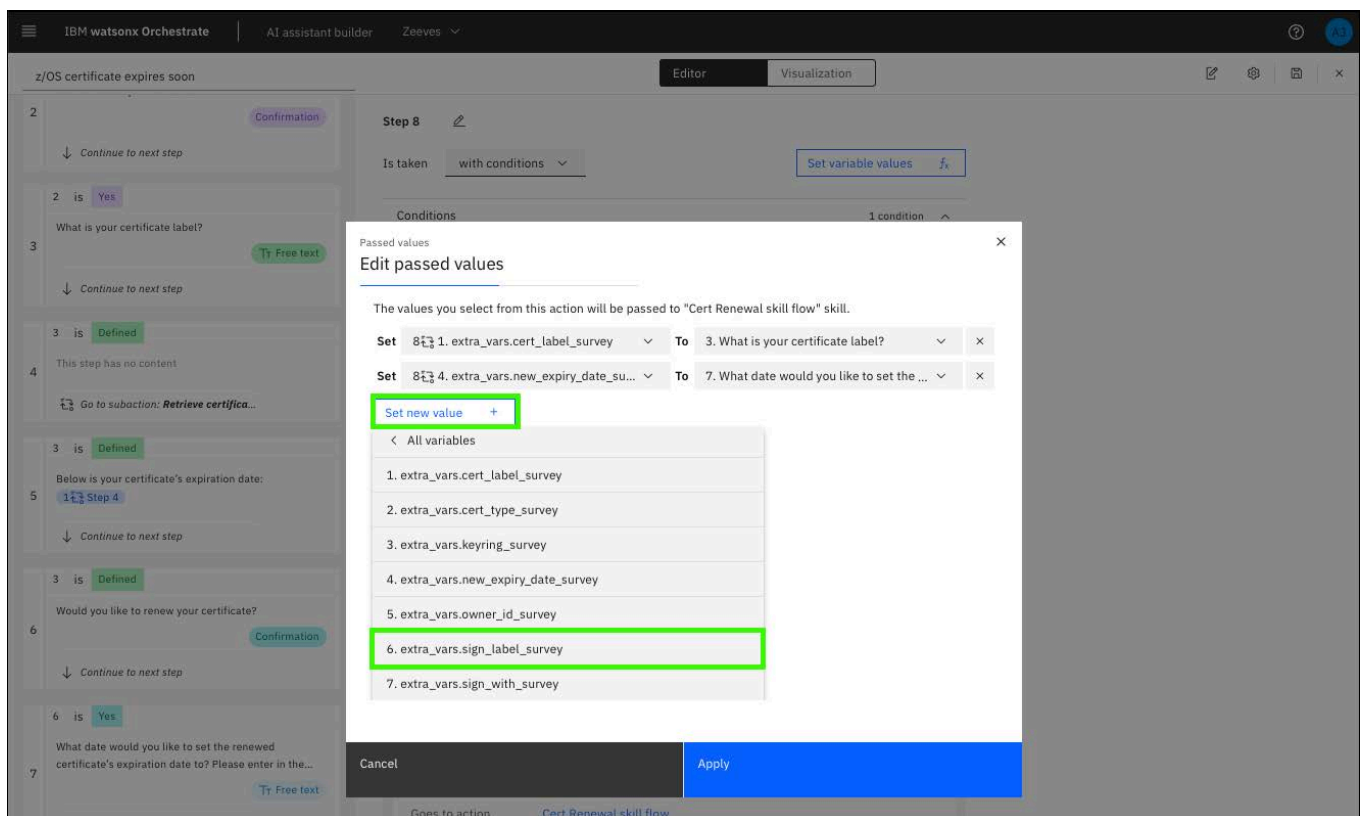
18. Click **What is your certificate label?**



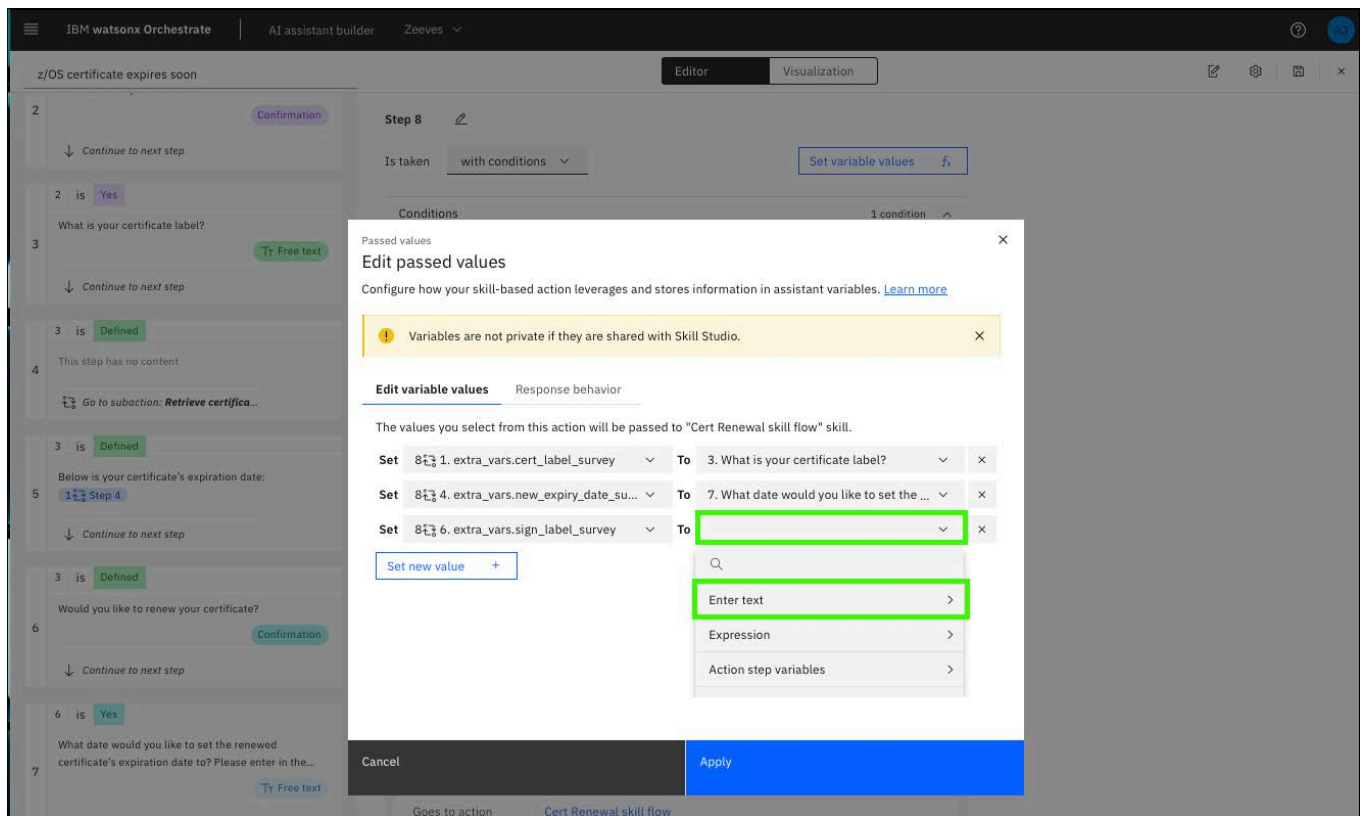
19. Repeat steps 16 - 18 adding the **extra_vars.new_expiry-date_survey** input variable and **What date would you like to set the...** in the To field.



20. Click **Set new value +** and then select **extra_vars.sign_label_survey**.



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

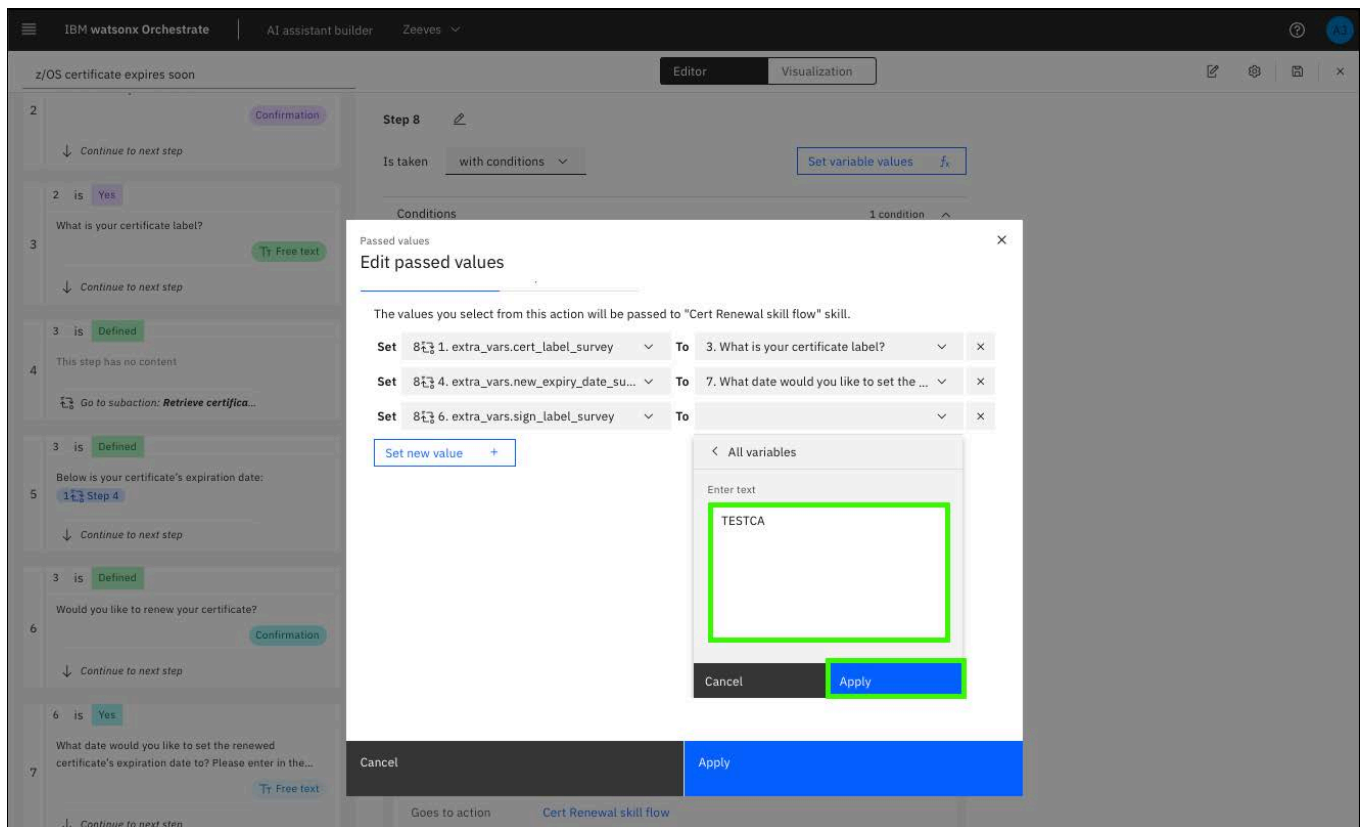


22. Enter `TESTCA` in the **Enter text** field and click **Apply** for the **To** option list.

Enter text:

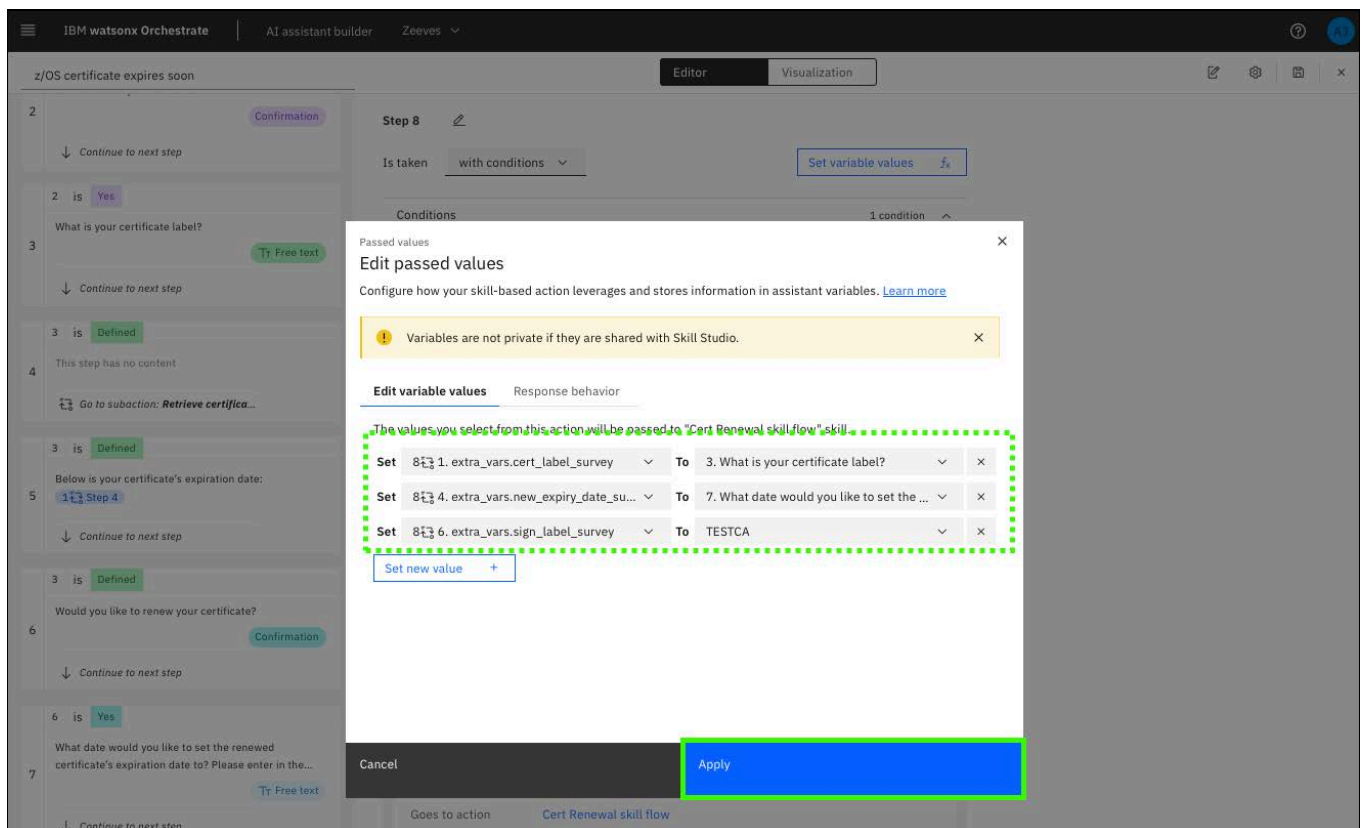
TESTCA

For this passed value, hardcode `TESTCA` in the skill flow's input for the `sign_label` variable. This is the CA certificate created earlier for demo purposes in the AAP web console.



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

Review all 3 variables are set correctly.



24. Click **Next step +**

The screenshot displays the IBM watsonx Assistant Builder interface. The left pane shows a flow diagram with steps 1-8. Step 8 is highlighted with a red box. The right pane shows the configuration for Step 8, including conditions and actions.

Step 8 Configuration:

- Is taken:** with conditions
- Conditions:** 1 condition
 - If All of this is true:
 - 7. What date would y... is defined
- Assistant says:**

```
Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below.
```
- And then:**
 - Go to a subaction: 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

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.

z/OS certificate expires soon

Editor Visualization

Step 9

Is taken

- without conditions ^
- without conditions ✓
- with conditions

Set variable values *f_x*

Assistant says

B I @ *f_x* [Icons] 00 </>

For example: Please select from the following options:

Define customer response

And then

Continue to next step

Step 9

This step has no content

Continue to next step

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 displays the IBM watsonx Assistant Editor interface. The top navigation bar shows 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main workspace is titled 'z/OS certificate expires soon' and has tabs for 'Editor' and 'Visualization'. The left sidebar shows a list of steps, with Step 9 highlighted. The main editor area for Step 9 shows the following configuration:

- Is taken:** with conditions (highlighted with a green dashed border)
- 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? (highlighted with a green dashed border)
 - Define customer response (dropdown menu)
- And then:** Continue to next step

The bottom of the left sidebar shows a preview of Step 9, which is also highlighted with a green dashed border. It displays the question 'Would you like to verify the new expiration date for your certificate?' and a 'Continue to next step' button.

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

The screenshot displays the IBM watsonx Assistant Builder interface. The top navigation bar shows 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main workspace is titled 'z/OS certificate expires soon' and has tabs for 'Editor' and 'Visualization'. The left sidebar shows a list of steps, with Step 9 highlighted. The main editor area for Step 9 shows a configuration for 'Is taken' with a condition '7. What date would y... is defined'. The right sidebar shows the 'Assistant says' section with a confirmation dialog and the 'Define customer response' section.

Step 9

Is taken **with conditions** [Set variable values](#)

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](#)

System

Options

Confirmation

Enable customers to select from Yes or No response.

As buttons

Yes No

Free text

Regex

Number

Date

Time

Saved

[New step](#)

[Preview](#)

28. Click **Next step +**.

The screenshot displays the IBM watsonx Assistant Editor interface. The left sidebar shows a list of steps for the workflow 'z/OS certificate expires soon'. Step 9 is highlighted, showing its configuration. The main editor area shows the configuration for Step 9, which is titled 'Step 9'. The 'Is taken' condition is set to 'with conditions'. A single condition is defined: '7. What date would you like to set the renewed certificate's expiration date to? Please enter in the...' with the value 'defined'. The 'Assistant says' section contains the text 'Would you like to verify the new expiration date for your certificate?'. The 'And then' section shows 'Continue to next step'. The 'New step' button is visible at the bottom left, and the 'Preview' button is at the bottom right.

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

On the condition that the user selected **Yes** in the previous step, configure a step to run the **Retrieve certificate expiration** skill-flow again to retrieve and display the new expiration date of the renewed certificate.

The screenshot displays the IBM watsonx Assistant Editor interface. The top navigation bar shows 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main workspace is titled 'z/OS certificate expires soon' and has tabs for 'Editor' and 'Visualization'. The left sidebar shows a list of steps (3, 5, 6, 7, 8, 9, 10) with their respective triggers and actions. Step 10 is highlighted with a green dashed border. The main editor area for Step 10 shows the 'Is taken' condition set to 'without conditions' (selected with a checkmark) and the 'Assistant says' section containing a text prompt: 'For example: What type of transfer would you like to make?'. The 'And then' section has a 'Continue to next step' option. The right sidebar shows a 'Set variable values' button and a 'Define customer response' dropdown.

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

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

32. Click **Edit passed values**.

IBM watsonx Orchestrate | AI assistant builder | Zeeves

z/OS certificate expires soon

Step 10

Is taken with conditions

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

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

Upon return Continue

Edit settings Edit passed values

33. Click **Set new value +** and then select **extra_vars.cert_label_survey**.

IBM watsonx Orchestrate | AI assistant builder | Zeeves

z/OS certificate expires soon

Step 10

Is taken with conditions

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

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

Upon return Continue

Edit settings Edit passed values

Passed values

Edit passed values

Configure how your skill-based action leverages and stores information in assistant variables. [Learn more](#)

Variables are not private if they are shared with Skill Studio.

Edit variable values Response behavior

The values you select from this action will be passed to "Retrieve certificate expiration" skill.

Set new value +

< All variables

1. extra_vars.cert_label_survey

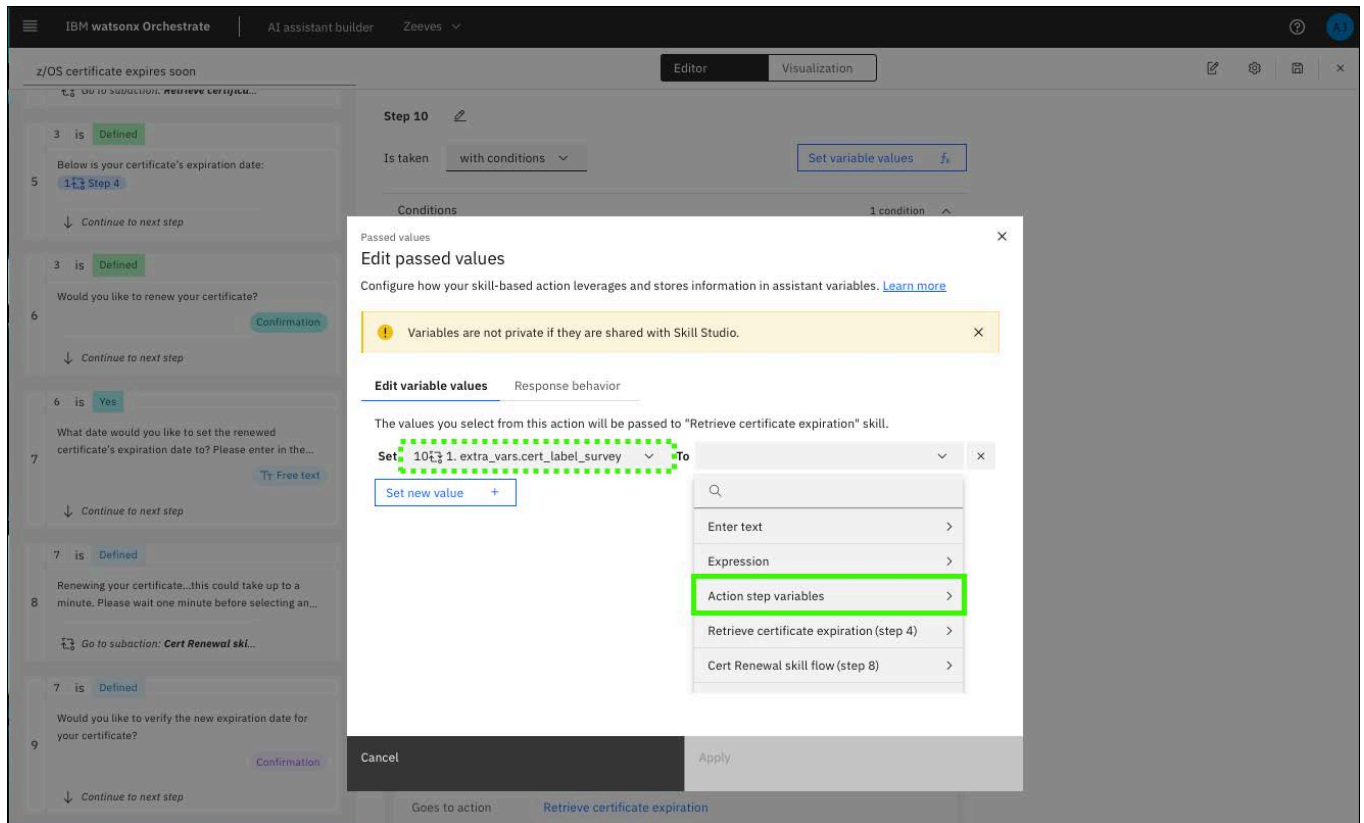
2. extra_vars.cert_type_survey

3. extra_vars.owner_id_survey

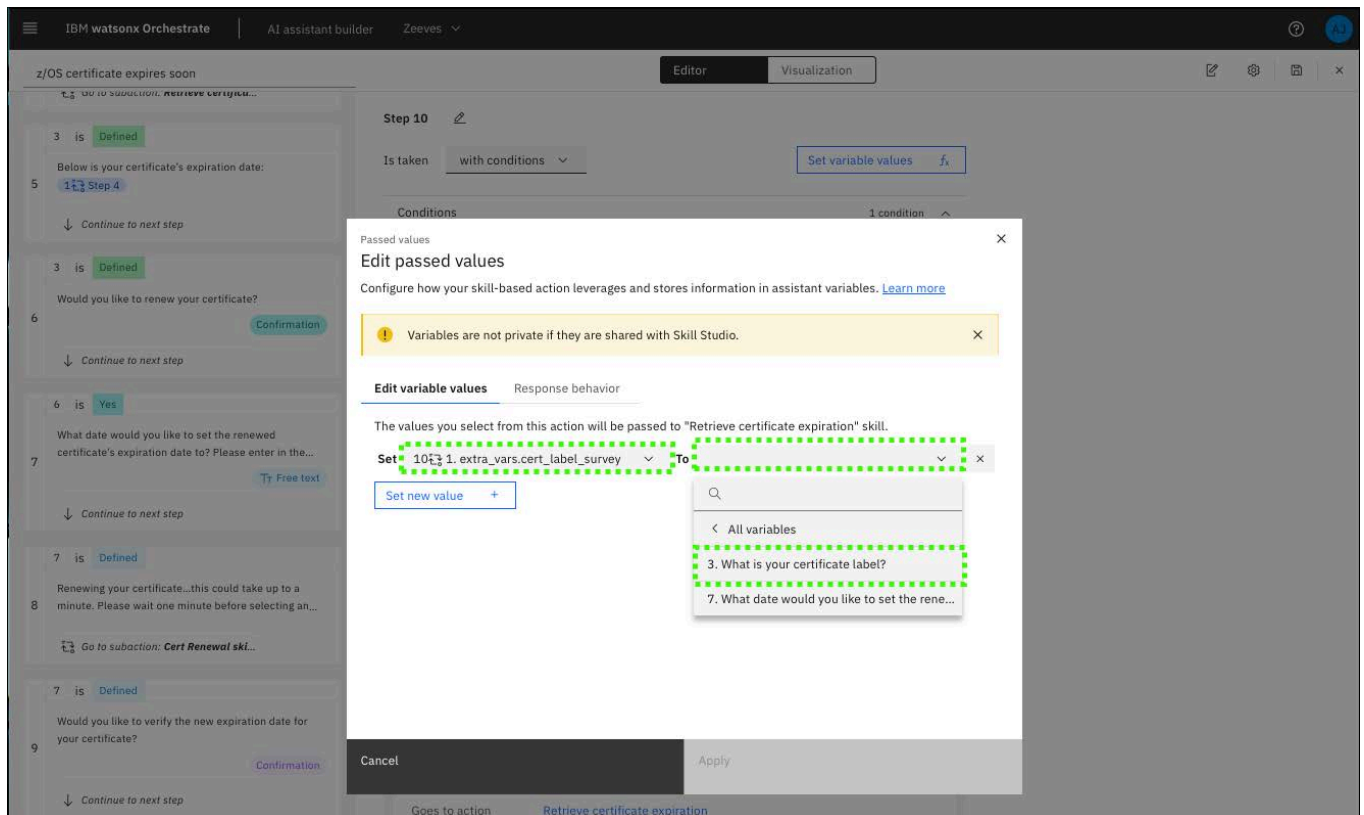
Cancel Apply

Goes to action Retrieve certificate expiration

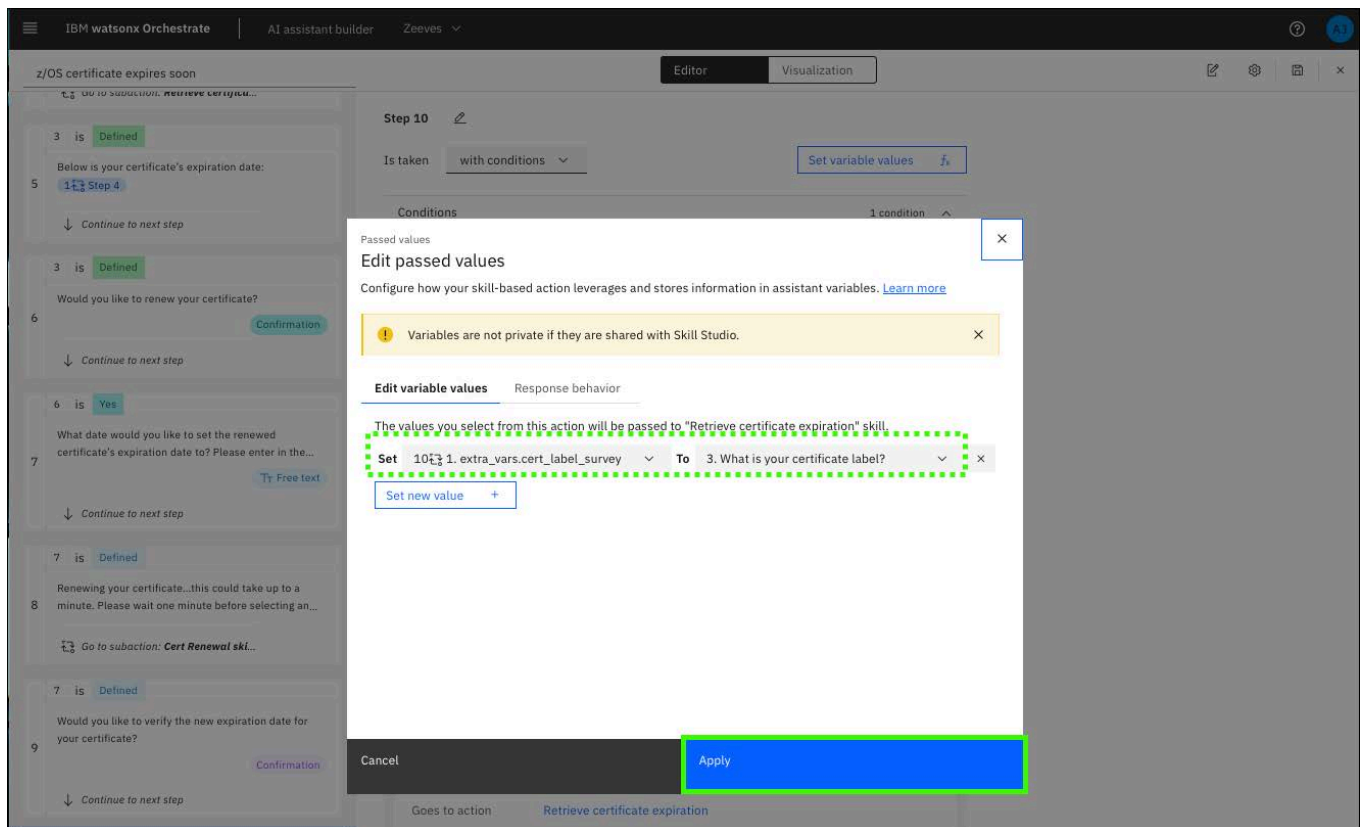
34. In the **To** option list, click **Action step variables**.



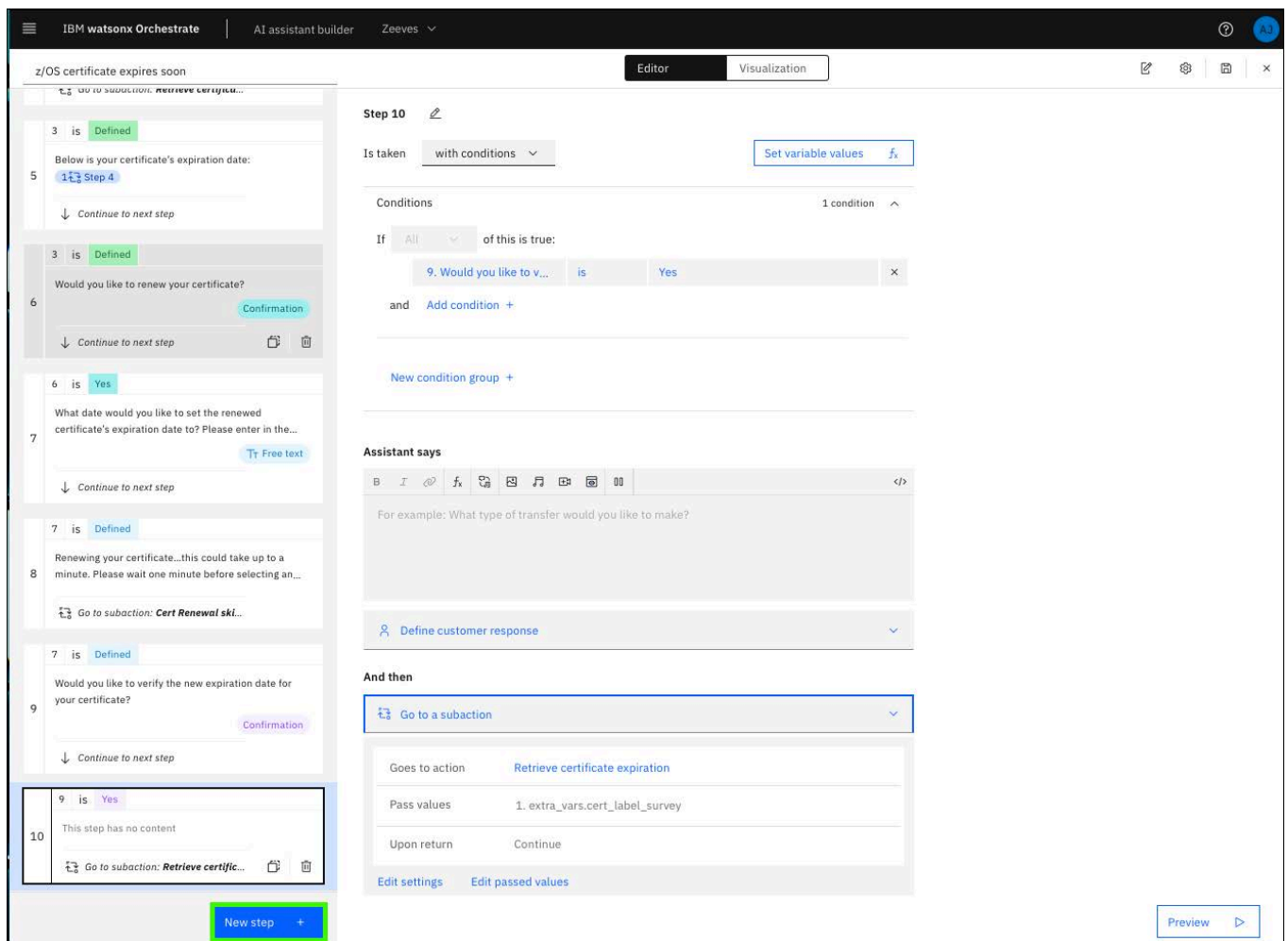
35. Click **What is your certificate label?**.



36. Review the **Edit variable values** and click **Apply**.



37. Click **Next step +**.



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

The final step is to display new expiration date of the certificate. Nothing is returned in the previous step when running the Retrieve certificate expiration skill flow - this was because the output form was hidden when the skill was created. In this step, provide the output as an assistant response to the user, with only the expiration date extracted from the full job output.

The screenshot shows the IBM watsonx Orchestrate AI assistant builder interface. The workflow is titled "z/OS certificate expires soon". The left pane shows a list of steps, with step 11 highlighted. The right pane shows the configuration for Step 11. In the "Is taken" section, the "with conditions" option is selected. The "Assistant says" field is empty, and the "And then" section shows a "Continue to next step" action.

Step 11 Configuration:

- Is taken:** without conditions (selected)
- Assistant says:** (Empty text area)
- And then:** 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 displays the IBM watsonx Assistant Editor interface. The left sidebar shows a workflow titled "z/OS certificate expires soon" with steps 1 through 11. Step 11 is highlighted with a green dashed border. The main editor area shows the configuration for Step 11. The "Is taken" condition is set to "with conditions". The "Conditions" section shows a single condition: "If All of this is true: 9. Would you like to v... is Yes". The "Assistant says" field is highlighted with a green border and contains the text: "Below is the new expiration date of your renewed certificate:". Below this field is a "Define customer response" button. The "And then" section shows a "Continue to next step" button.

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 displays the IBM watsonx Assistant Editor interface. The left sidebar shows a list of steps in a workflow titled "z/OS certificate expires soon". Step 11 is highlighted, showing its content: "Below is the new expiration date of your renewed certificate:
". The right pane shows the configuration for Step 11. It includes a "Conditions" section with a single condition: "If All of this is true: 9. Would you like to v... is Yes". Below this is a "New condition group +" button. The "Assistant says" section shows a code editor with a dropdown menu open, displaying options to select variables from previous steps: "Action step variables", "Retrieve certificate expiration (step 4)", "Cert Renewal skill flow (step 8)", "Retrieve certificate expiration (step 10)", "Assistant variables", and "Integration variables".

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



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

The screenshot displays the IBM watsonx Assistant Editor interface. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main workspace is titled 'z/OS certificate expires soon' and has tabs for 'Editor' and 'Visualization'.

On the left, a list of steps is shown. Step 11 is highlighted, showing its content: 'Below is the new expiration date of your renewed certificate:
'. Below this, there is a 'Continue to next step' button.

On the right, the configuration for Step 11 is shown. It includes a 'Set variable values' button and a 'Conditions' section. The conditions are set to 'If All of this is true:' with one condition: '9. Would you like to v... is Yes'. Below the conditions, there is a 'New condition group +' button.

At the bottom of the right pane, there is a list of variables. The 'Retrieve certificate expiration (step 10)' variable is highlighted with a green box. Other variables include 'Action step variables', 'Retrieve certificate expiration (step 4)', 'Cert Renewal skill flow (step 8)', 'Assistant variables', and 'Integration variables'.

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

The screenshot displays the IBM Watsonx Orchestrate interface for configuring a workflow. The left sidebar shows a sequence of steps: Step 3 (Defined), Step 6 (Confirmation), Step 7 (Free text), Step 8 (Defined), Step 9 (Confirmation), and Step 10 (Empty). Step 11 is currently selected and highlighted in blue. The main workspace shows the configuration for Step 11, which is a confirmation step. It includes a 'Conditions' section with a single condition: 'If All of this is true: 9. Would you like to v... is Yes'. Below the conditions, there is a section for 'Assistant says' which contains a text block: 'Below is the new expiration date of your renewed certificate: 1. Retrieve certificate expiration result var...'. The 'And then' section at the bottom shows a 'Continue to next step' action.

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

The screenshot displays the IBM watsonx Assistant Editor interface. The top navigation bar shows 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main workspace is titled 'z/OS certificate expires soon' and has tabs for 'Editor' and 'Visualization'. The left sidebar shows a list of steps, with Step 11 highlighted. The main editor area for Step 11 shows a configuration for 'Is taken' with conditions. The right sidebar shows a 'Continue to next step' button and an 'End the action' button. The 'Assistant says' section shows a message about the new expiration date.

Step 11

Is taken: with conditions

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:

10. Retrieve certificate expiration r...

Define customer response

And then

Continue to next step

End the action

Make this a final step that completes the action.

Continue to next step

Re-ask previous step(s)

Go to a subaction

Use an extension

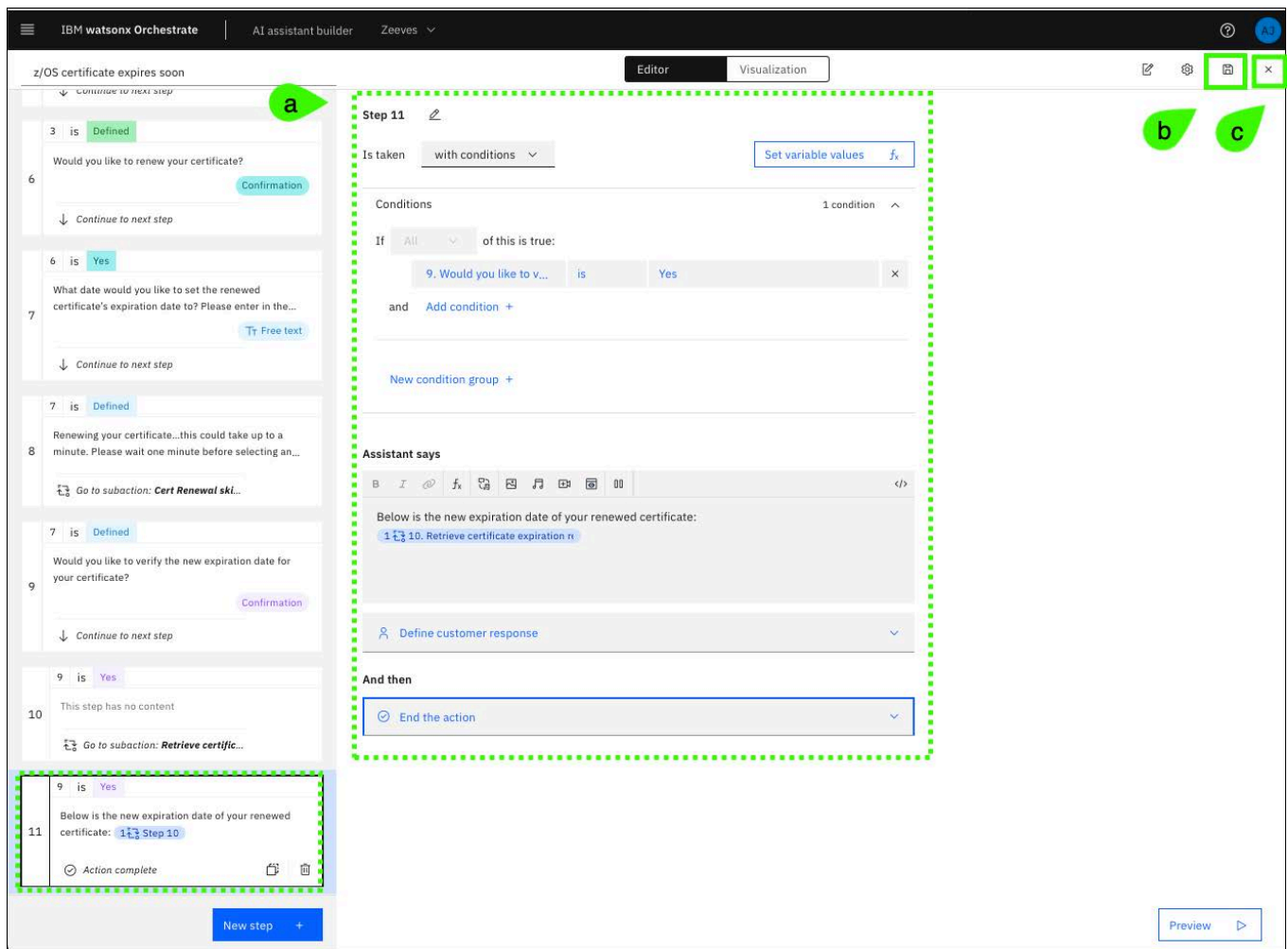
Search for the answer

Connect to agent

End the action

Preview

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



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**.

Created by you /

Filter by name

☐	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	4 hours ago	0	3	✓
☐	Cert Renewal skill flow	4 hours ago	0	7	✓
☐	z/OS certificate expires soon	a few seconds ago	1	11	✓

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.

Preview assistant

Copy link to share

Change layout

Change background

Customize web chat

AssistantZeeves

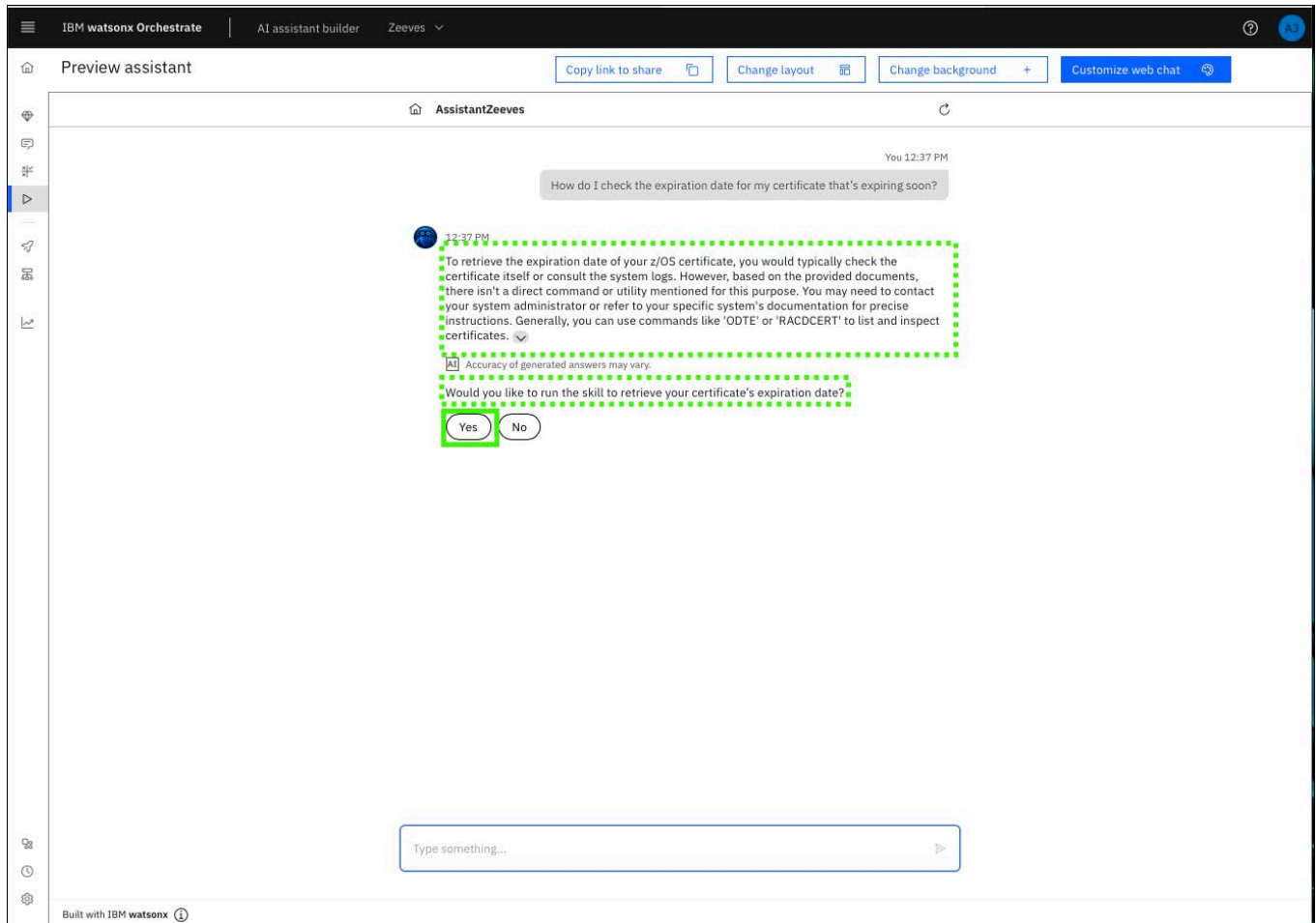
Hi! I'm Zeeves. How can I help you today?

How do I check the expiration date for my certificate that's expiring soon?

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.

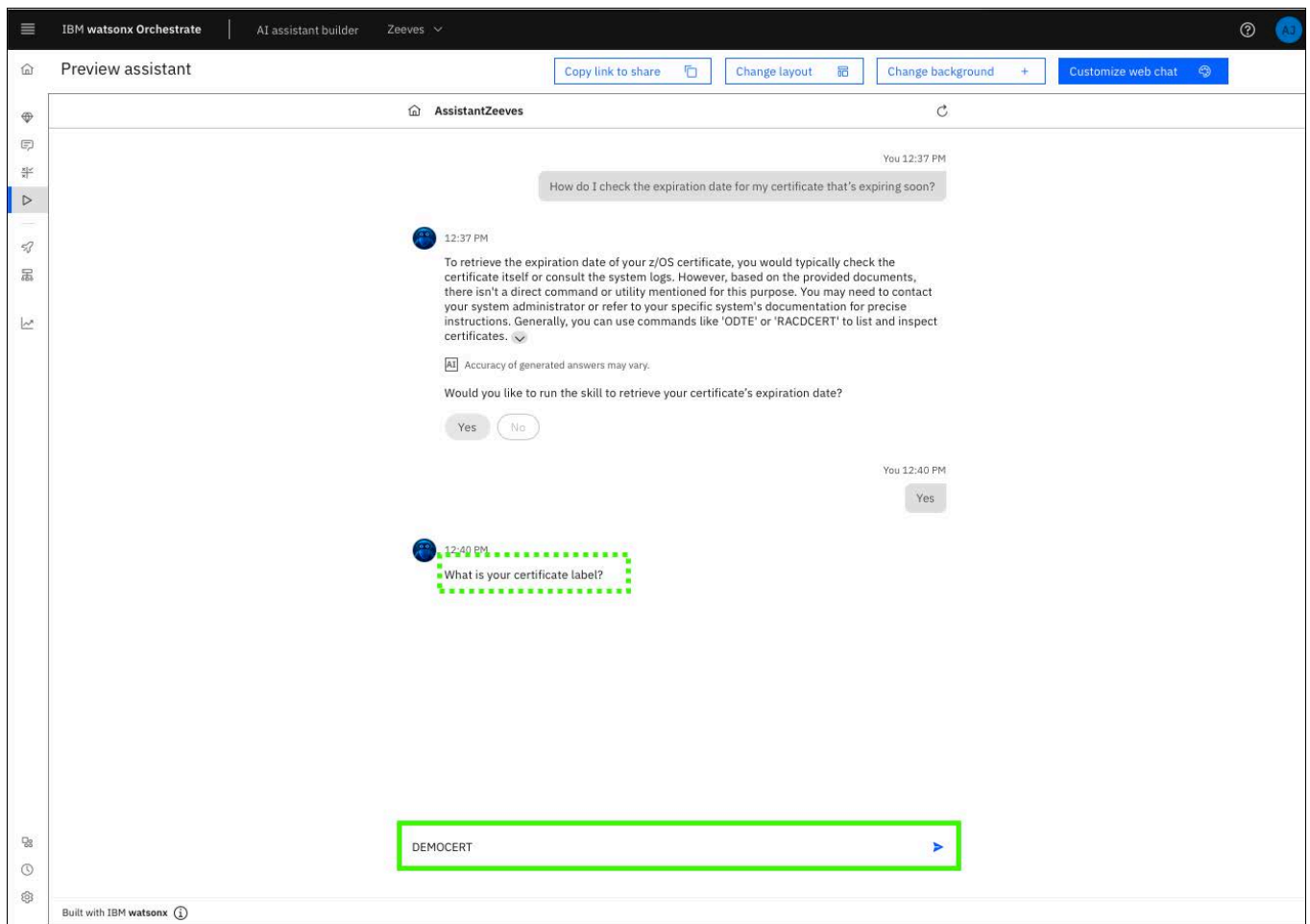
Following the response, the assistant prompts the user if they want to run the skill to retrieve a certificate's expiration date.



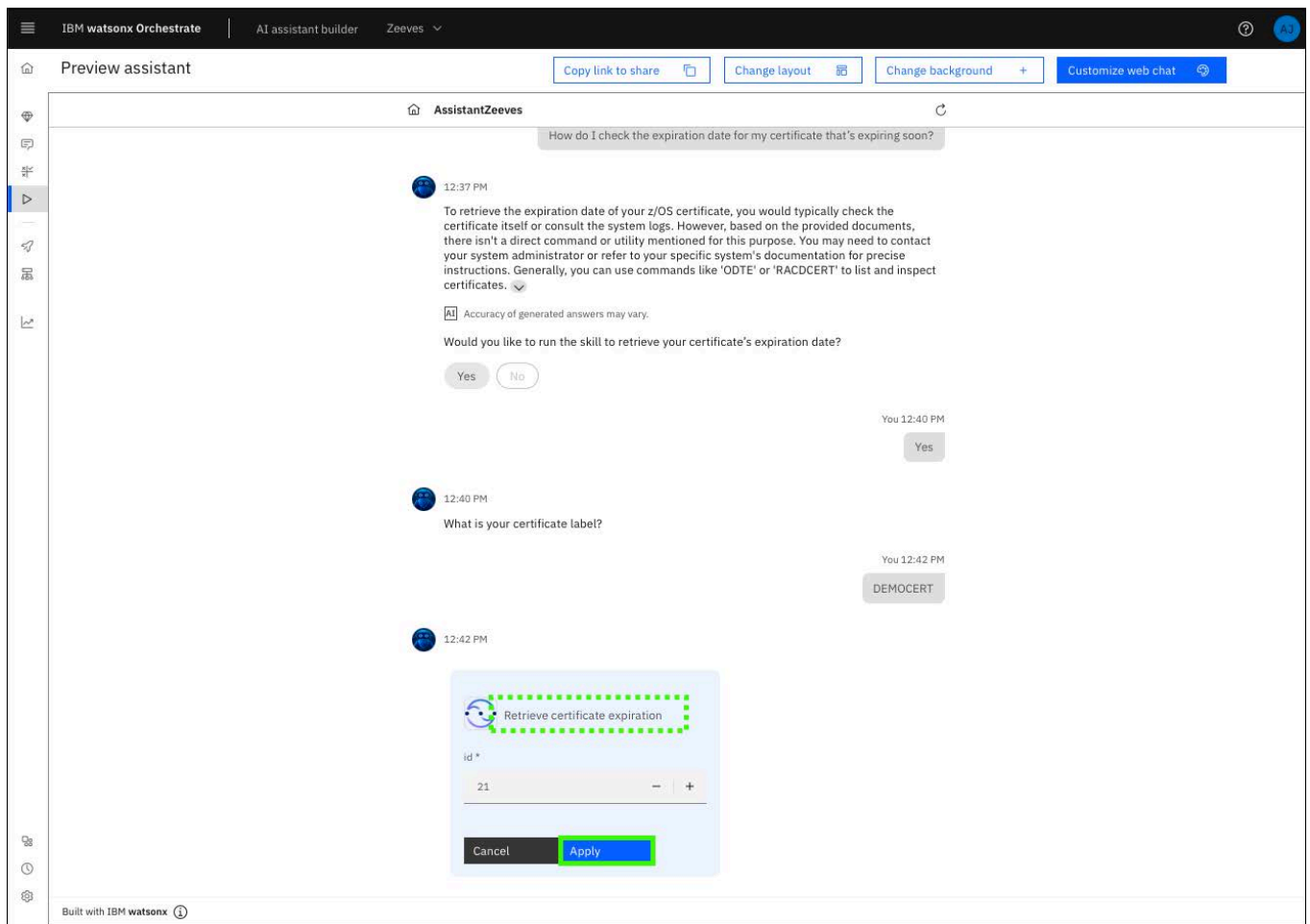
4. Enter `DEMOCERT` after the assistant responds with **What is your certificate label?**

Prompt:

DEMOCERT



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

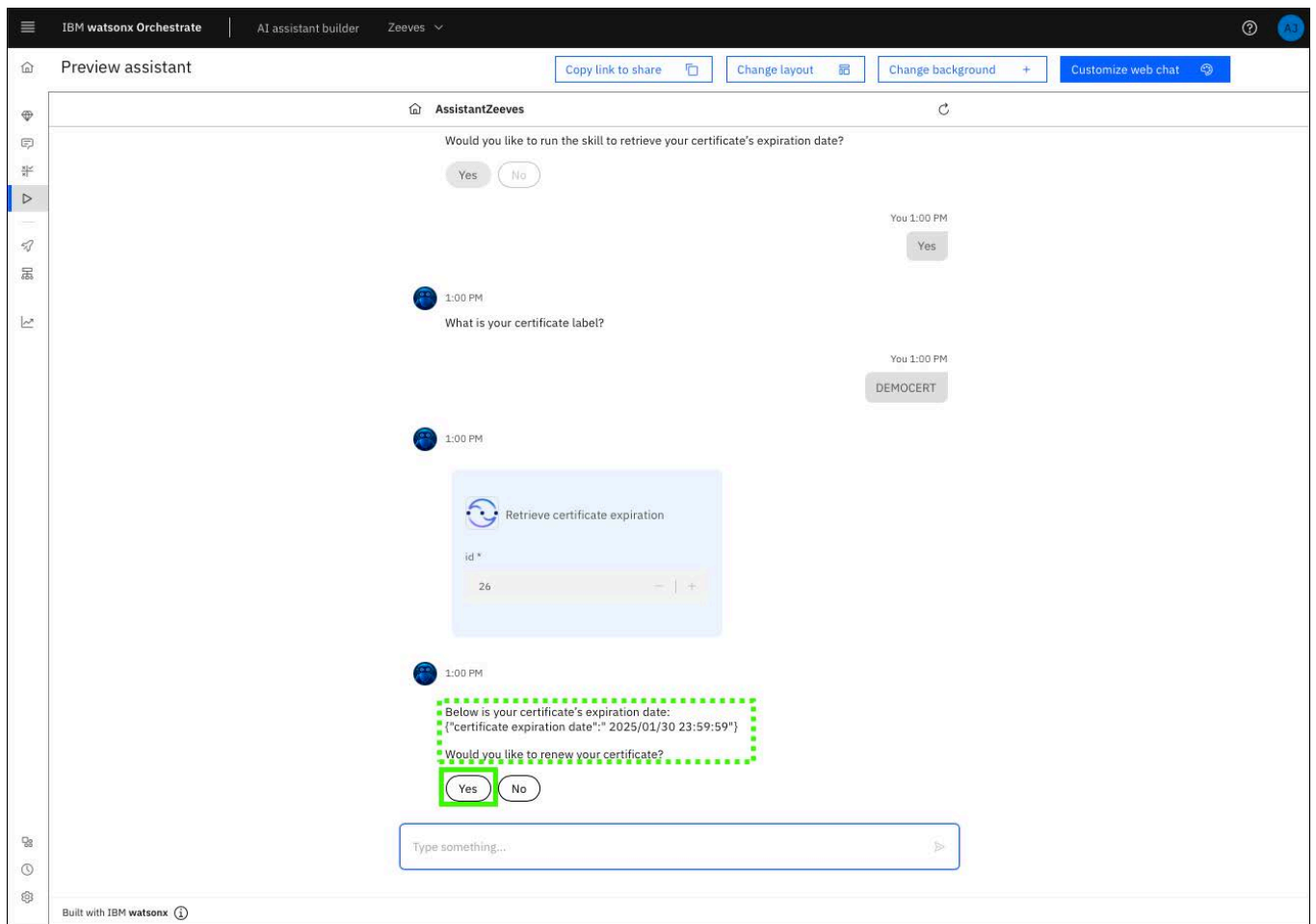


6. Review the response and then click **Yes**.

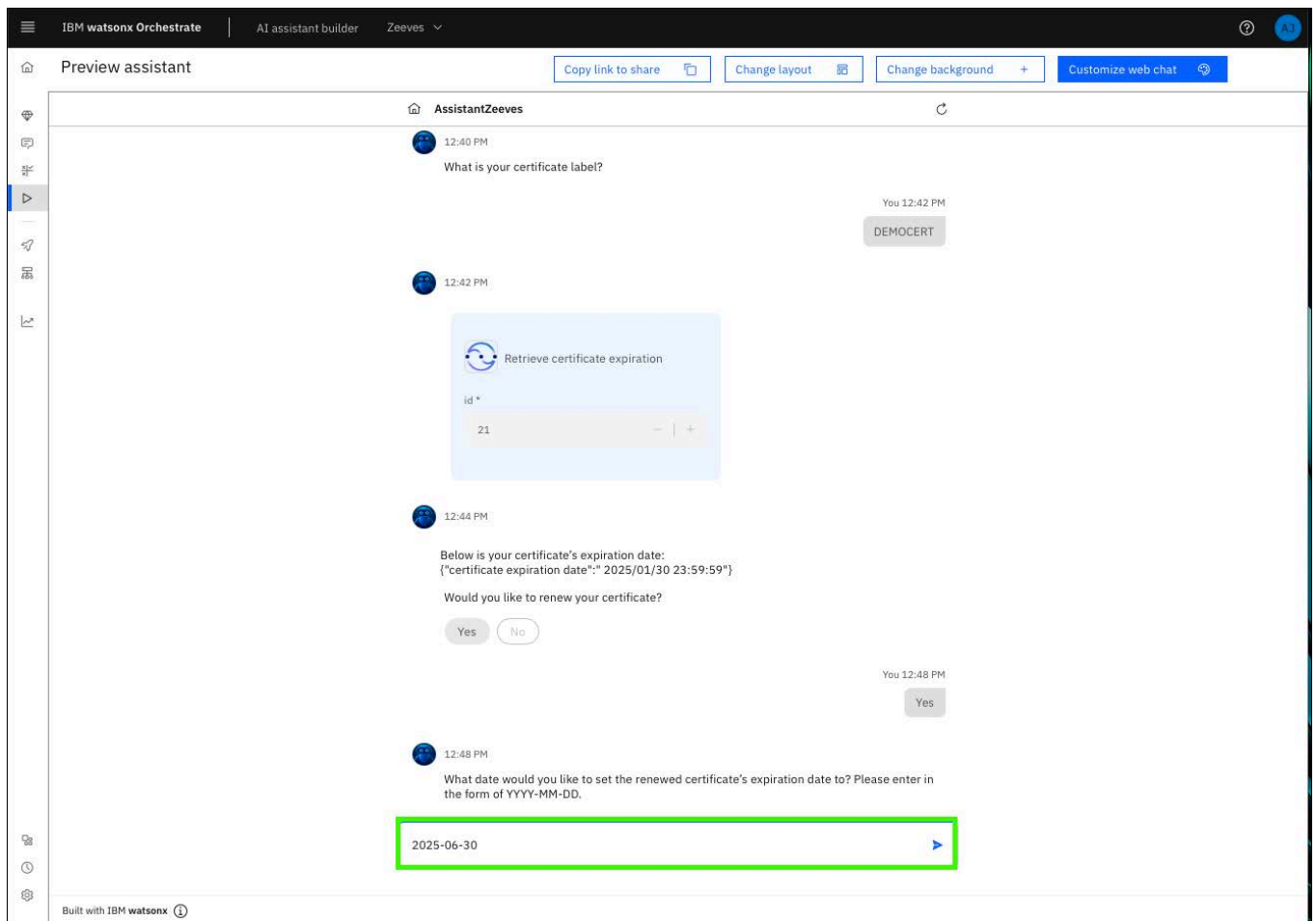
By providing the automation within the assistant conversation, it makes it very quick for the SA to identify the certificate's expiration date. In addition to providing this valuable information, the assistant is configured with another automation to renew the certificate if they choose to do so.



The expiration date you see may differ from the image that follows.



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



8. Review the response, wait 30 seconds to a minute, and then click **Yes**.



It is crucial that you wait 30 seconds to a minute before selecting Yes.

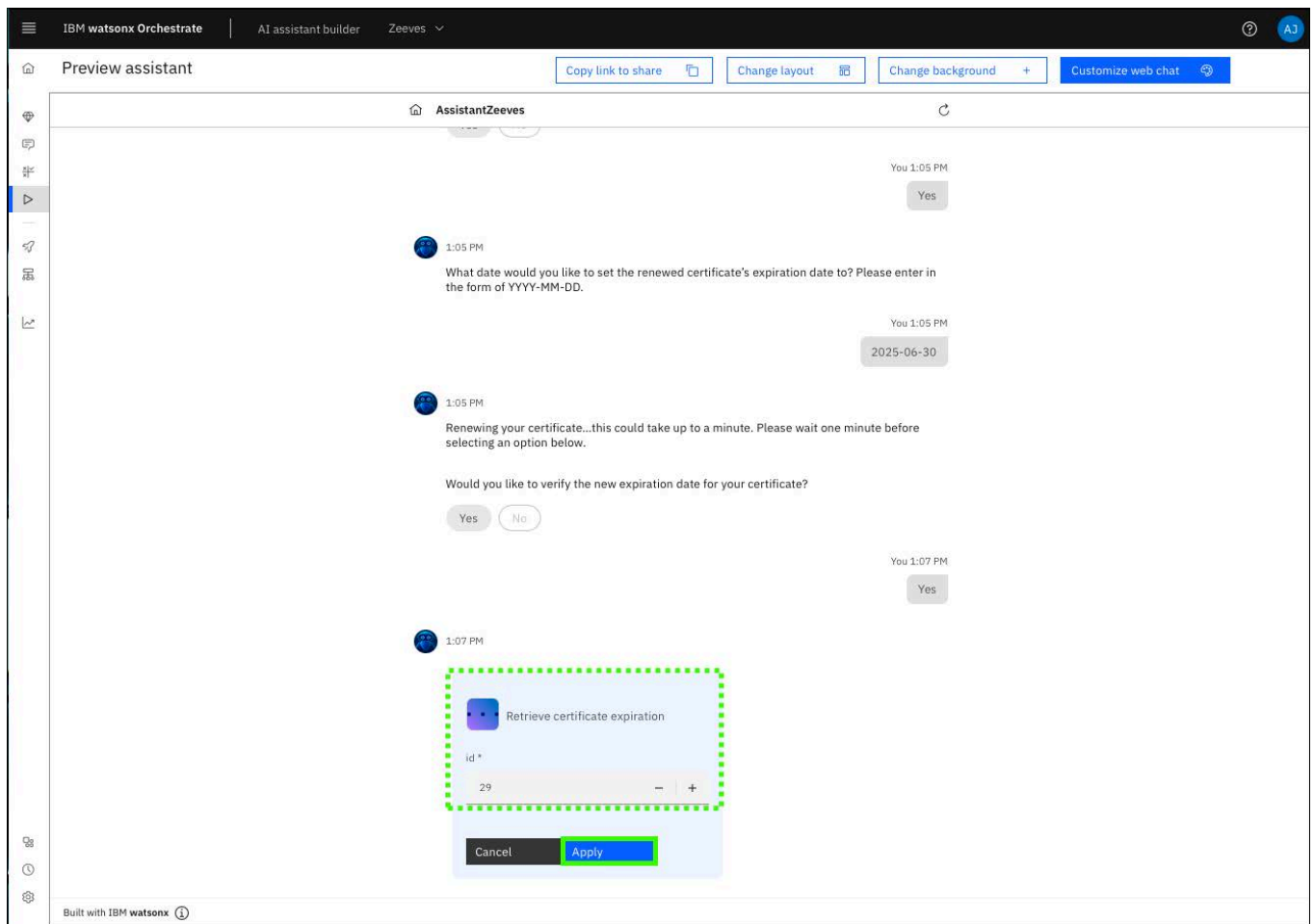
This is because in the background, your z/OS Certs – Search and Renew automation is running within AAP (which you can verify within the AAP Web console). This is mapping the user-inputted expiration date as well as the original certificate label provided by the end-user to the inputs of this AAP automation."

The screenshot displays the IBM watsonx Assistant interface in 'Preview assistant' mode. The top navigation bar includes 'IBM watsonx Orchestrate', 'AI assistant builder', and 'Zeeves'. The main chat area, titled 'AssistantZeeves', shows a sequence of messages:

- A system message at the top displays a variable `id *` with the value `27`.
- A bot message at 1:05 PM: "Below is your certificate's expiration date: {'certificate expiration date': '2025/01/30 23:59:59'}. Would you like to renew your certificate?" with 'Yes' and 'No' buttons.
- A user message at 1:05 PM: "Yes".
- A bot message at 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." with a text input field.
- A user message at 1:05 PM: "2025-06-30".
- A bot message at 1:05 PM: "Renewing your certificate...this could take up to a minute. Please wait one minute before selecting an option below. Would you like to verify the new expiration date for your certificate?" with 'Yes' and 'No' buttons. The 'Yes' button is highlighted with a green dashed border.
- A text input field at the bottom with the placeholder "Type something...".

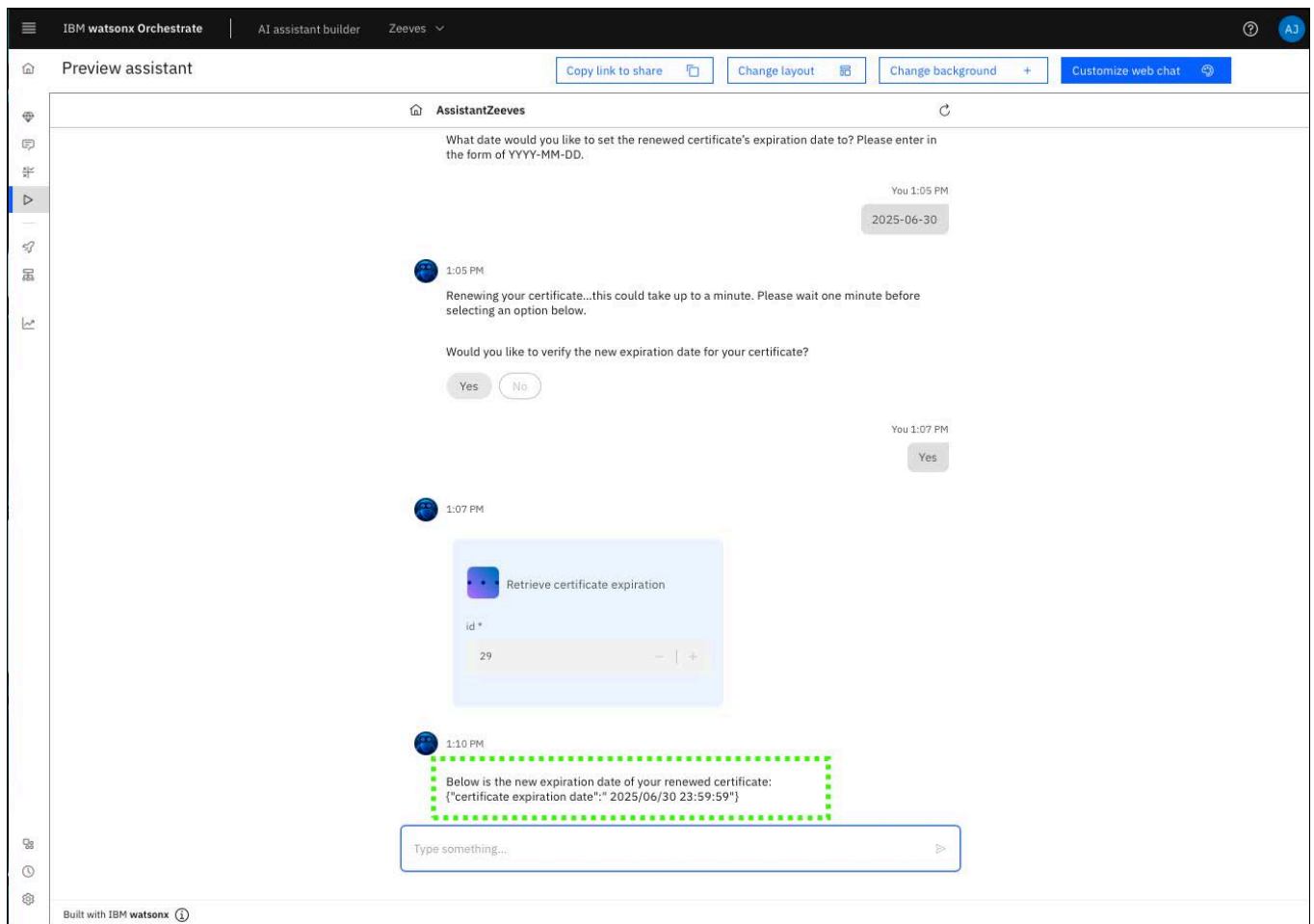
The bottom status bar indicates "Built with IBM watsonx".

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



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 **COmmand(s) Retrieve job output** (utility skill)

IBM watsonx Orchestrate | Skill studio

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

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.

IBM watsonx Orchestrate

Skill catalog /

Ansible Controller Skills - TSO Command (2)

Personal skills

Search skills

Ansible Controller Skills - TSO Command

Retrieve job output
TSO Command - Retrieve job output by job Id

Added ✓

z/OS TSO Command(s)
TSO Command - A playbook to execute one or multiple TSO commands on the target z/OS system.

Added ✓

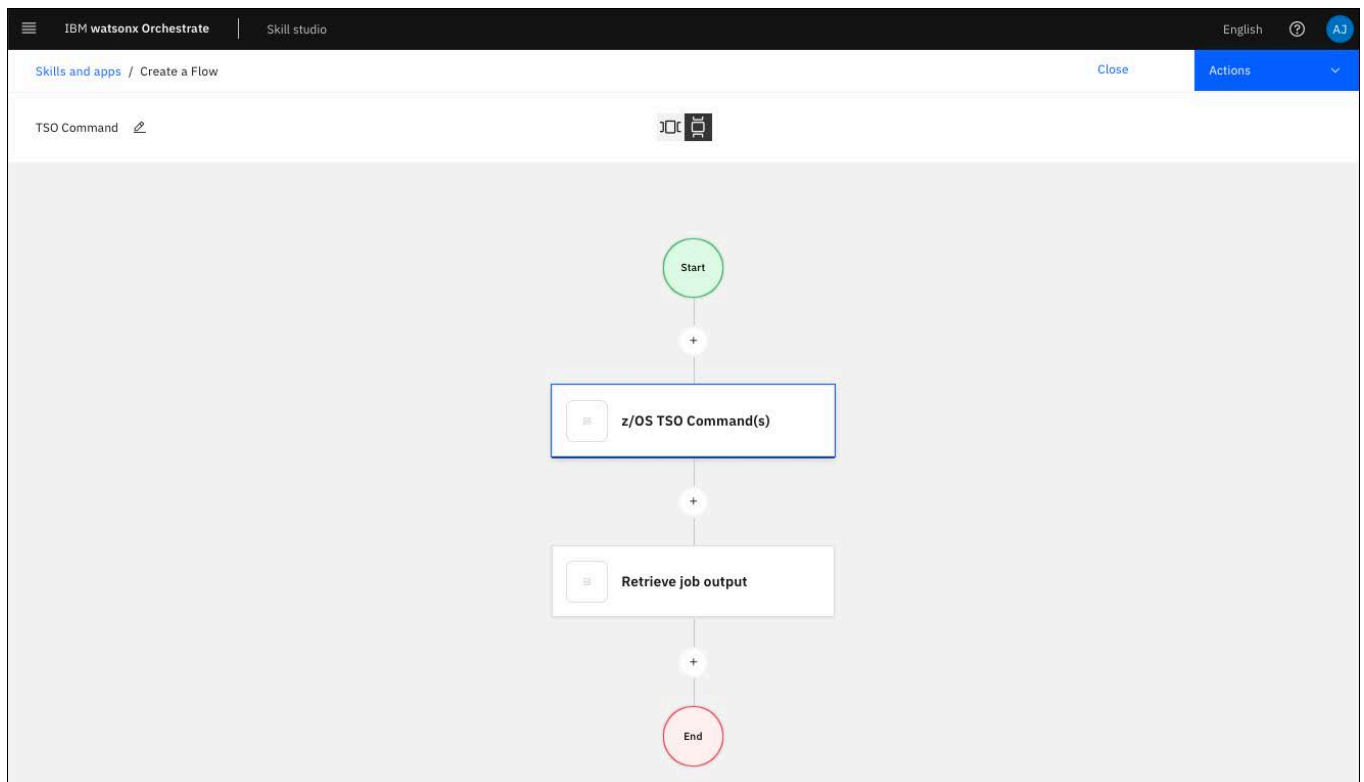
3. Create a skill flow to get the output of the TSO Command.

Use Skills studio to create the skill flow.

The skill flow should run the TSO Command skill and then the **Retrieve job output** utility skill. The output from the TSO Command should be mapped to the input of the **Retrieve job output** utility skill. Hide both the input and output forms for the TSO Command skill.

Skill flow name:

TSO COMMAND



4. Add the skills to the **draft** version of the assistant.



Use Skill sets to add the skills to the assistant.

Zeeves draft

Skill sets

Zeeves draft

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 - TSO Command	2	Team	andrew@jones-tx.com	

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

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.

The screenshot shows the 'Actions' section of the IBM watsonx Orchestrate AI assistant builder. The interface includes a sidebar with navigation options like 'All items', 'Created by you', 'Set by assistant', 'Variables', 'Created by you', 'Set by assistant', 'Set by integration', and 'Saved responses'. The main area displays a table of actions created by the user.

Name	Last edited	Examples count	Steps count	Status
☐ TSO Command	a few seconds ago	0	1	✓
☐ z/OS certificate expires soon	3 hours ago	1	11	✓
☐ Cert Renewal skill flow	7 hours ago	0	7	✓
☐ Retrieve certificate expiration	7 hours ago	0	3	✓
☐ Retrieve IPL information	a day ago	1	3	✓
☐ z/OS IPL Information	a day ago	0	3	✓
☐ Zeeves-gather-facts-flow	3 days ago	1	0	✓

6. Create a custom-built action for RACF administration support.



Use AI Assistant builder to create the action.

Phrase to start the interaction:

RACF assistance

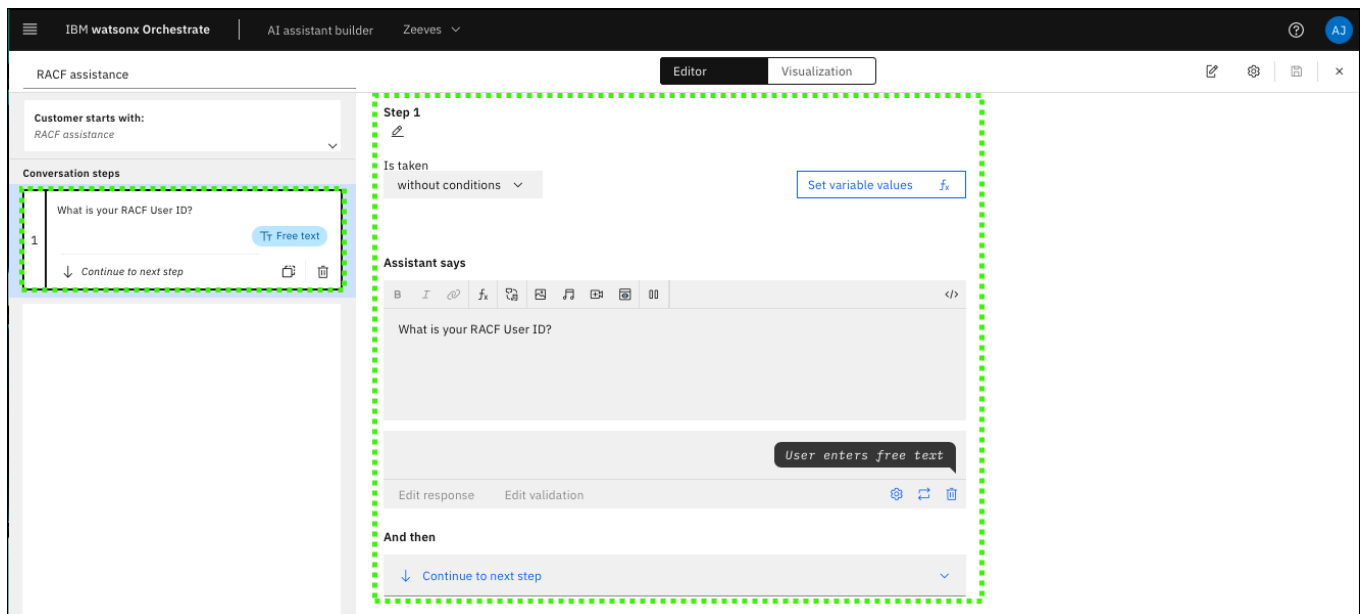
The steps for the custom-built action follow:

Step 1. Without conditions, prompt the user for their RACF User ID as *free text*.

Assistant says:

What is your RACF User ID?

The step should look like the following image:



Use the following guidance when entering prompts and expressions in the following custom-built action steps.

Do NOT use the copy and paste icon if the string contains a \$. Type each string manually.

When \$ appears in the string to enter, type the \$ character and then select the variable specified in the <>. For example: User passphrase changed to \$<7. Please enter your new RACF passphrase>

Use the single quote character ' for all single quotes shown.

Step 2. *Without conditions*, use provided user ID and display a message that the assistant is checking the current privileges.

Assistant says:

Checking user privileges for \$<1. What is your RACF User ID>

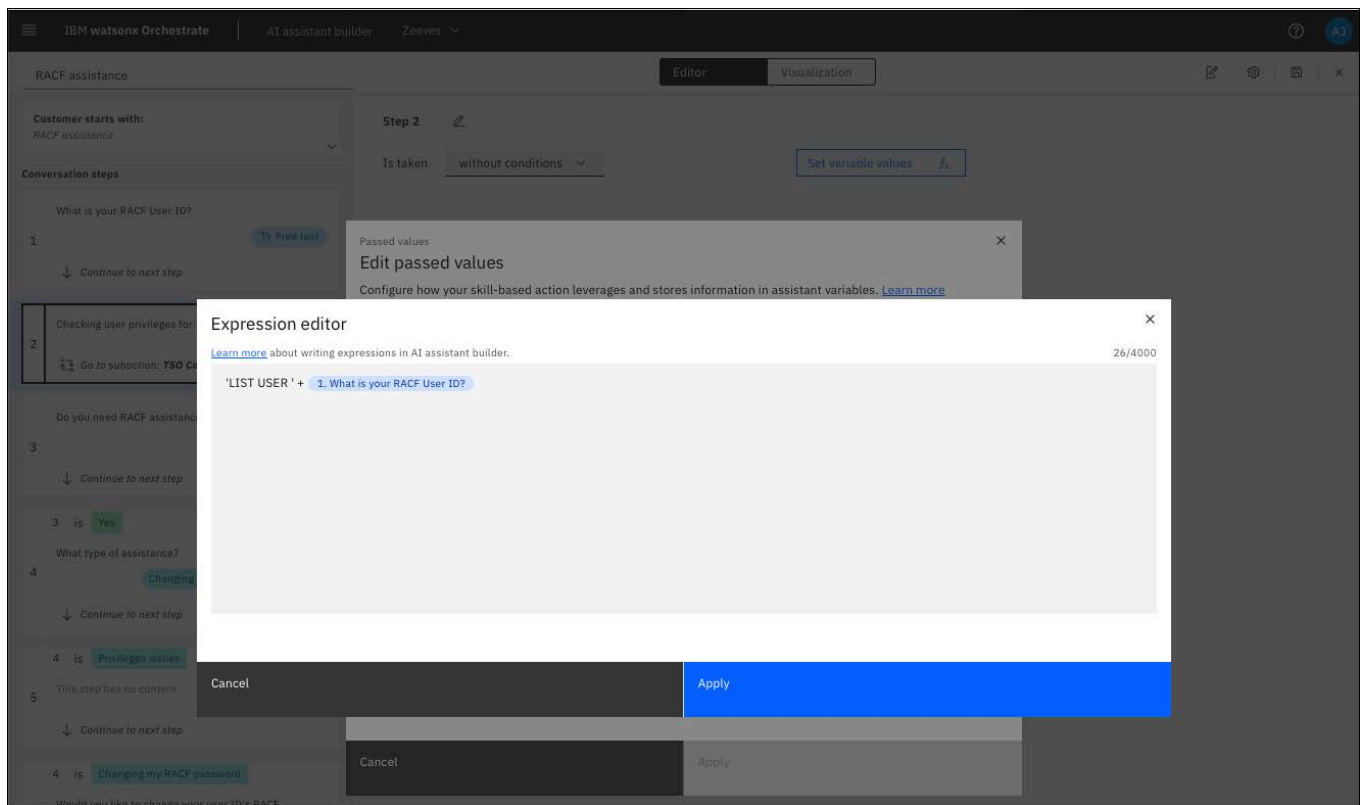
Include in the **Assistant says** the provided user ID by including the function 1. *What is your RACF User ID.*

Add an **And then** option to run the **TSO Command** as a subaction.

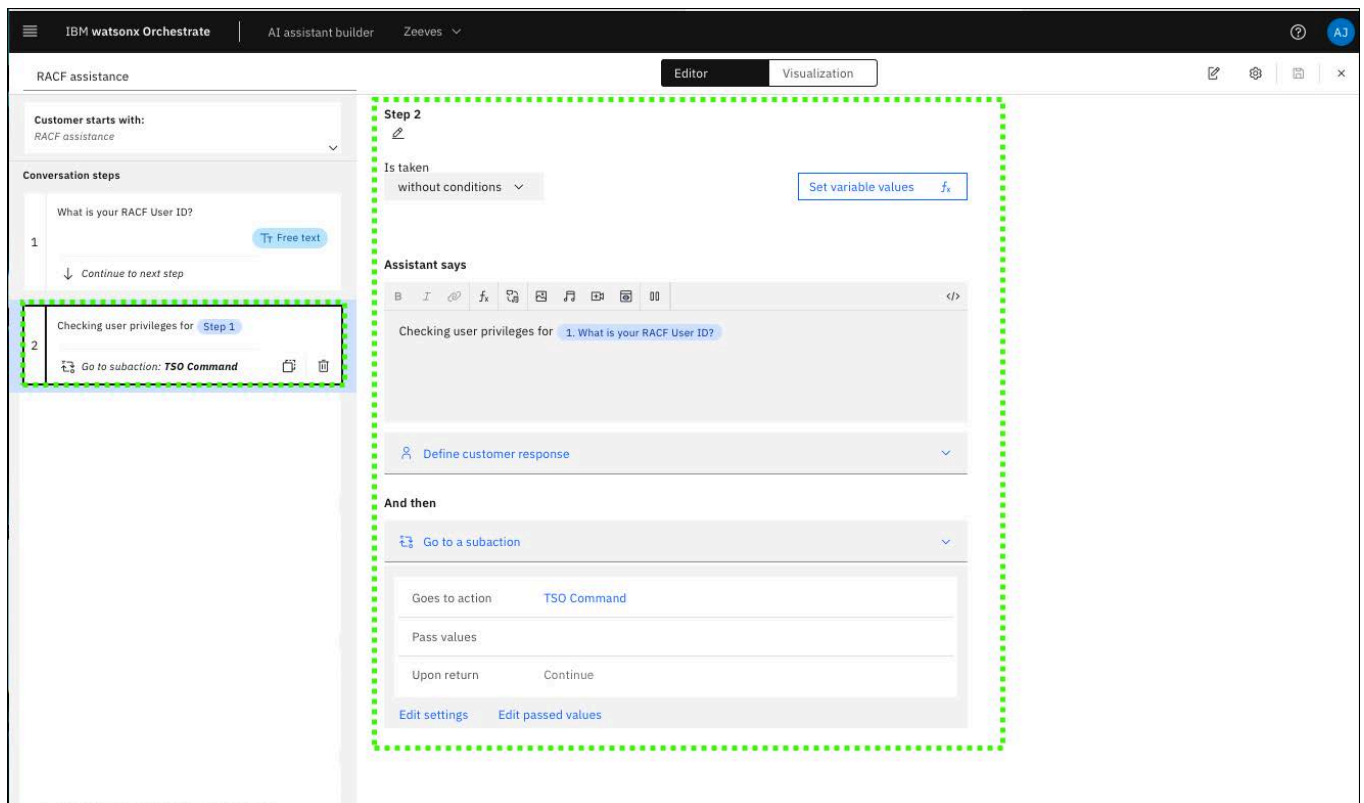
Edit the **passed values** for the subaction and add a new passed value for 1. *extra_vars.zos_tso_command*. The **To** of the variable will be an **Expression** and include the string:

'LIST USER ' + \$<1. What is your RACF User ID>

After the +, enter a \$ and select **Action step variables** and then select 1. *What is your RACF User ID.*



The step should look like the following image:

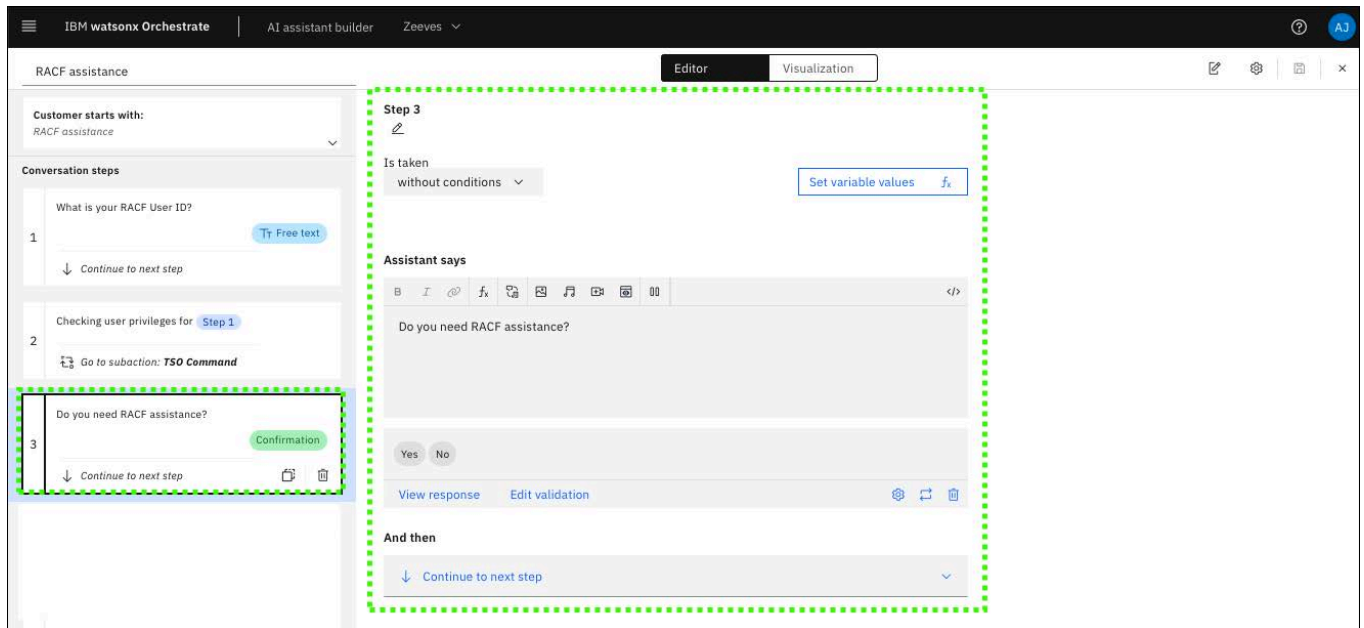


Step 3. Without conditions, prompt the user if they want RACF assistance and get a **Yes** or **No** confirmation.

Assistant says:

Do you need RACF assistance?

The step should look like the following image:



Step 4. *With conditions*, prompt the user what type of assistance they need.

Assistant says:

What type of assistance?

Create a **Options** type **Customer response** with two options:

Changing my RACF password

and

Privileges issue

Change the **Customer response** settings to **Always ask for this information, regardless of previous messages**.

The screenshot shows the IBM watsonx Orchestrator AI assistant builder interface. On the left, the 'Conversation steps' list shows four steps. Step 4 is highlighted with a blue border. The main editor area shows the configuration for Step 4. It is titled 'Step 4' and has a 'Set variable values' button. The 'Is taken' section is set to 'with conditions'. The 'Conditions' section shows a single condition: 'If All of this is true: 3. Do you need RACF ... is Yes'. The 'Assistant says' section shows the prompt 'What type of assistance?' and two possible responses: 'Changing my RACF password' and 'Privileges issues'. The 'And then' section shows the action 'Continue to next step'.

The step should look like the following image:

This screenshot is identical to the one above, showing the configuration for Step 4. However, the entire configuration area for Step 4, including the 'Is taken' section, 'Conditions', 'Assistant says' section, and 'And then' section, is enclosed in a green dashed border. This highlights the specific configuration that needs to be modified in the next step.

Step 5. With conditions, change the condition **Step 4. What type of assistance?** and the value of **Privilege issue**.

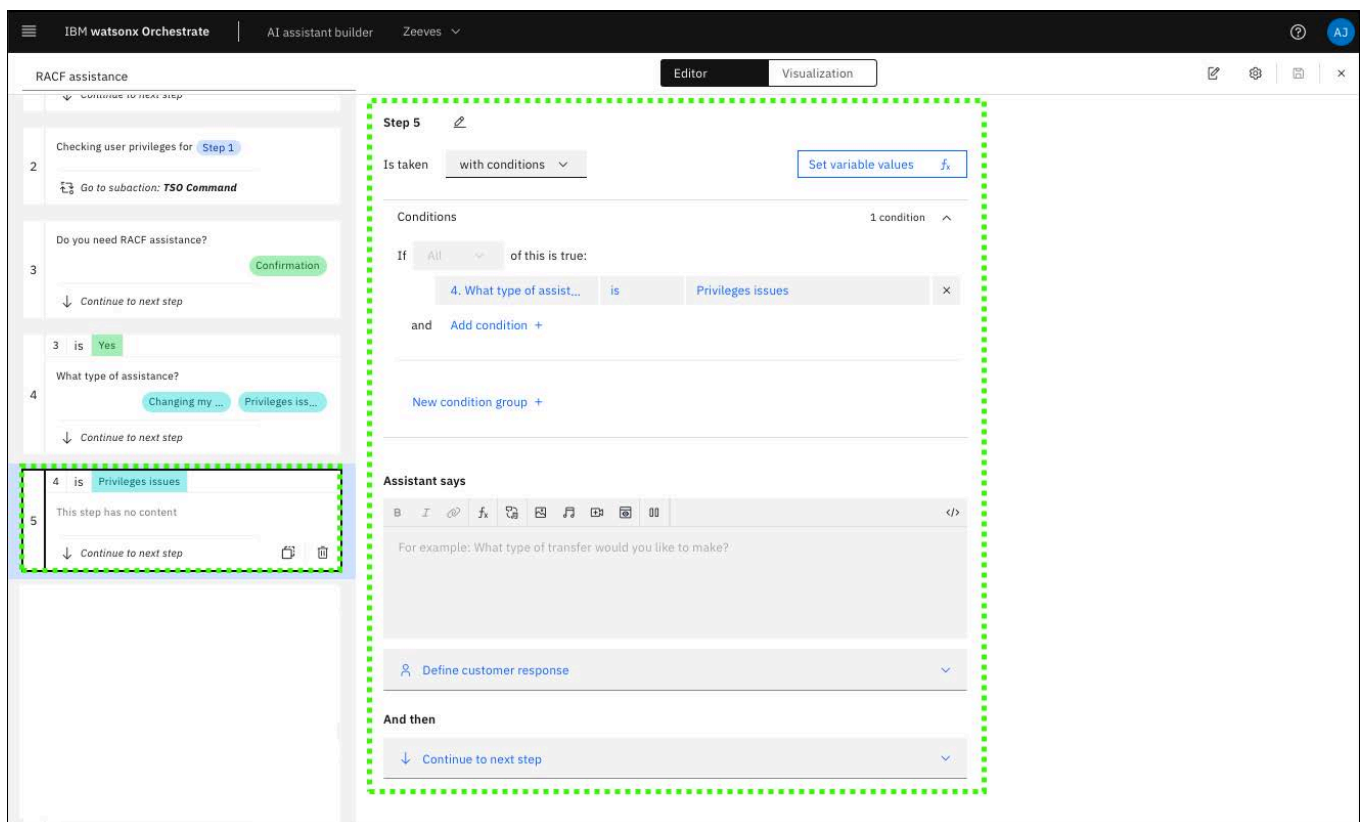
Assistant says:

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:

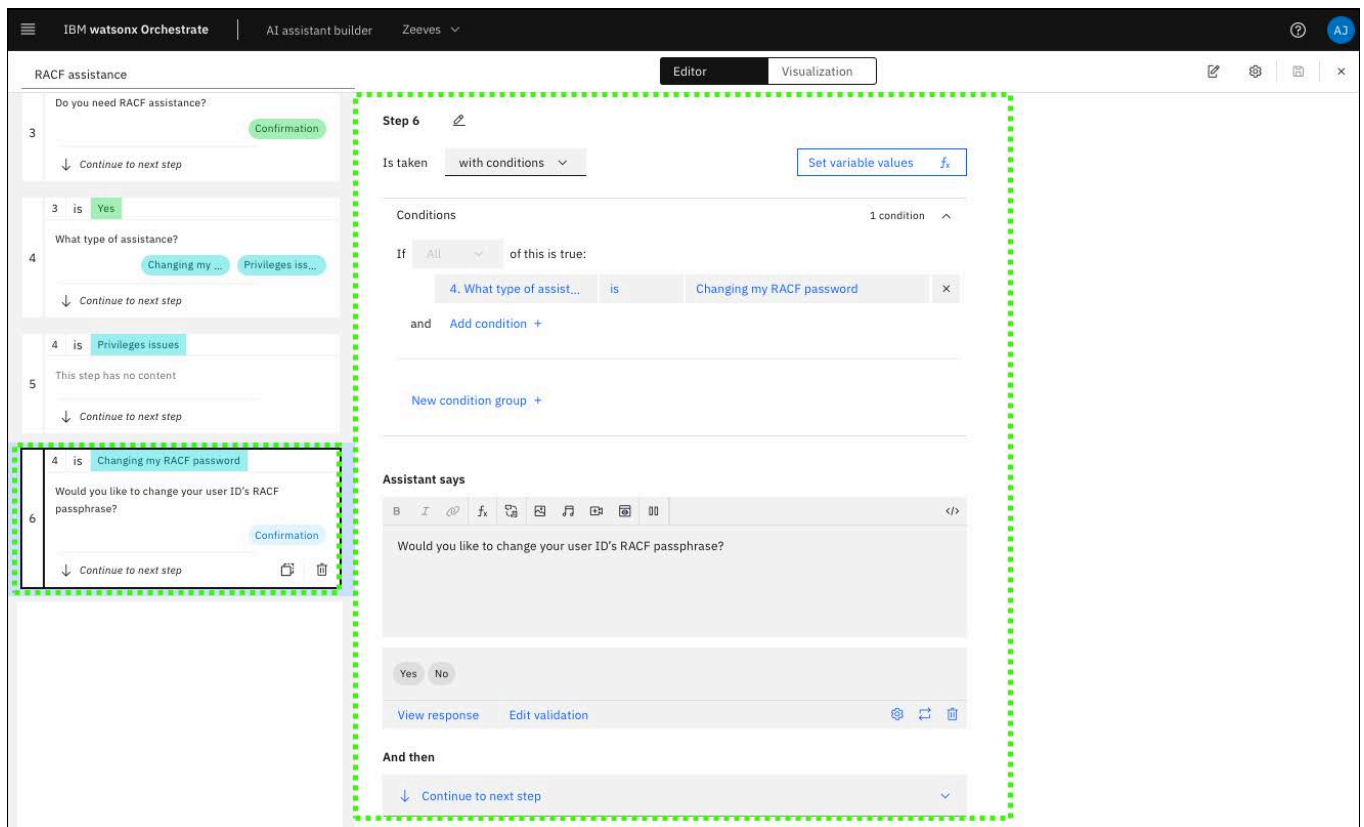


Step 6. *With conditions*, change the condition **Step 4. What type of assistance?** and the value of **Changing my RACF password**, and add a **Yes** or **No** confirmation.

Assistant says:

Would you like to change your user ID's RACF passphrase?

The step should look like the following image:

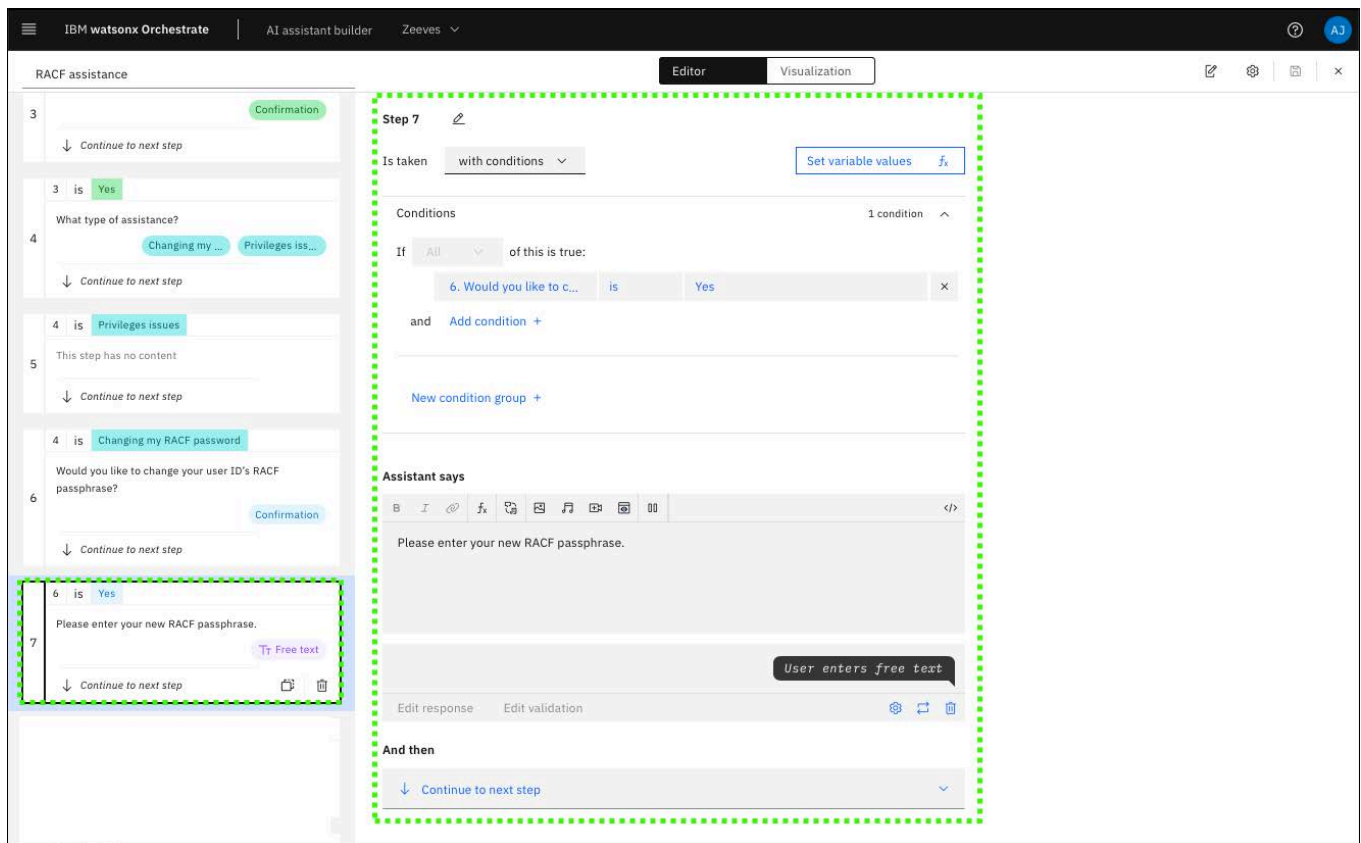


Step 7. *With conditions*, prompt the user to enter their new password as *free text*.

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 IDS passphrase equals Yes**, inform the user the password is being changed, and create a **TSO Command** subaction to change the passphrase.

Assistant says:

Issuing RACF command ...



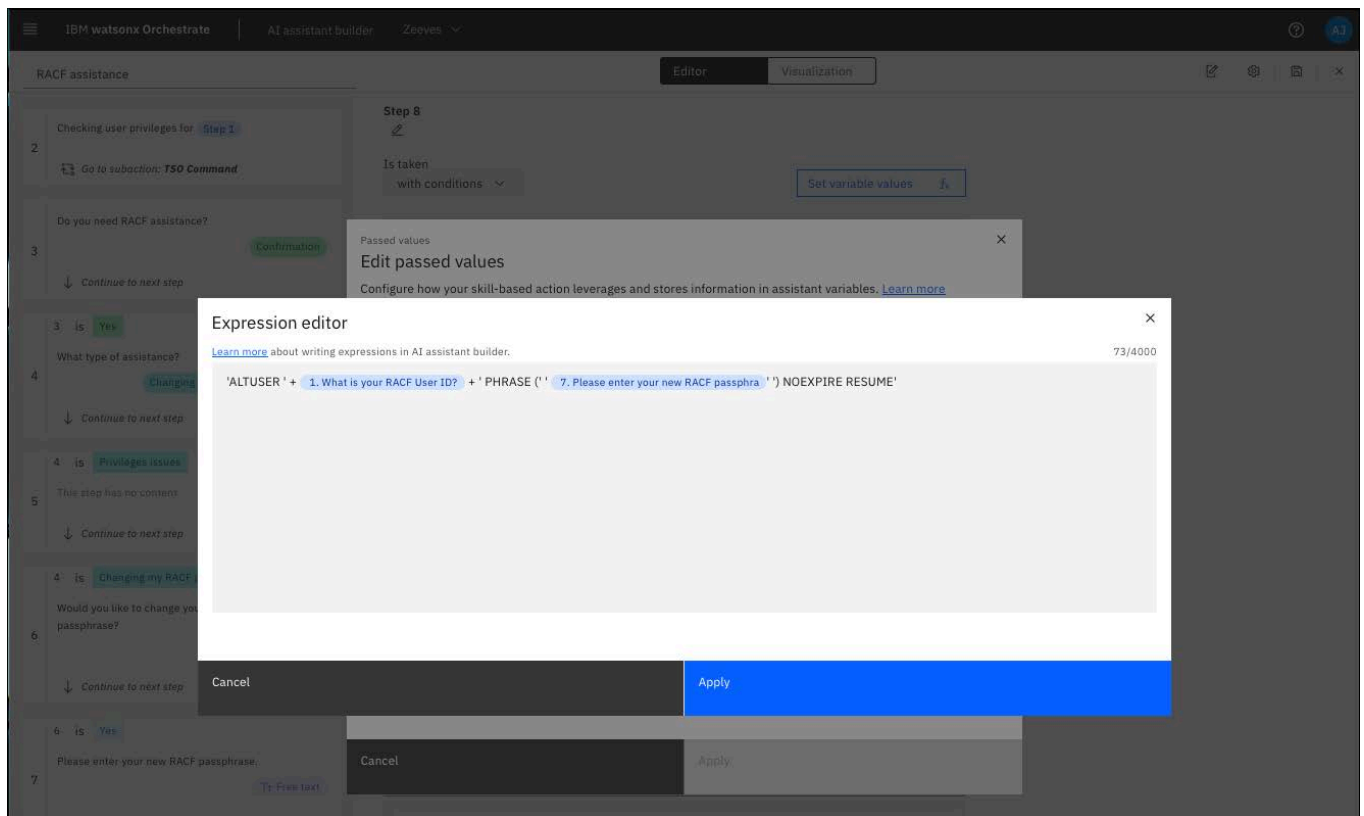
The formatting of the values that follow is important.

When creating the **To** expression for the passed values, be very careful with typing the expression. Do NOT cut & paste this value. Type each character. All quotes are single-quotes. The `$<>` denotes typing `$` and then selecting the appropriate **action step variable**.

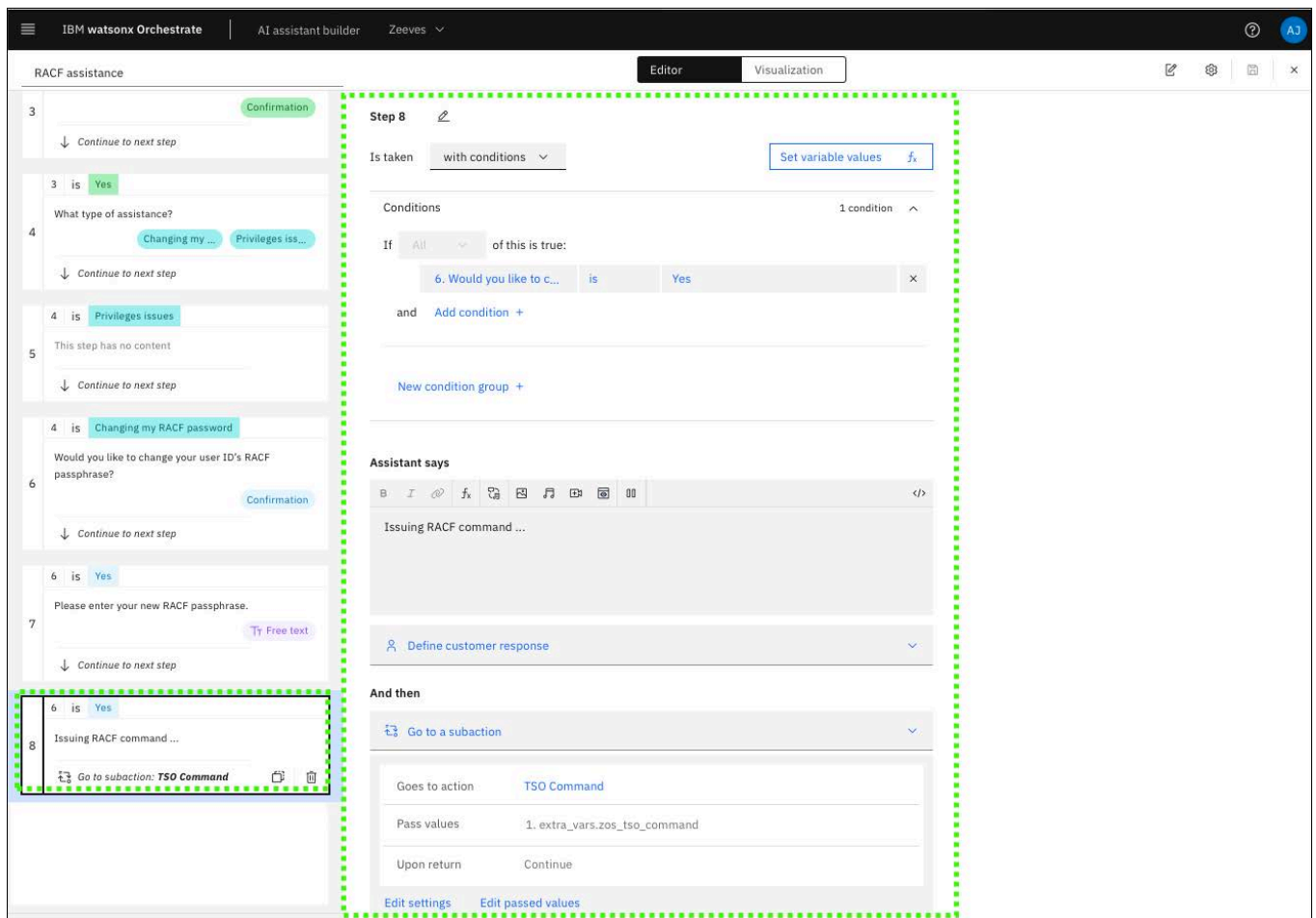
Edit the passed values for the **TSO Command** subaction to include the **1. extra_vars.zos_tso_command** variable with a **To** expression with the value of:

```
'ALTUSER ' + $<1. What is your RACF User ID?> + ' PHRASE(''$<7. Please enter your new RACF passphrase>'') NOEXPIRE RESUME'
```

The expression should look like the following image:



The step should look like the following image:



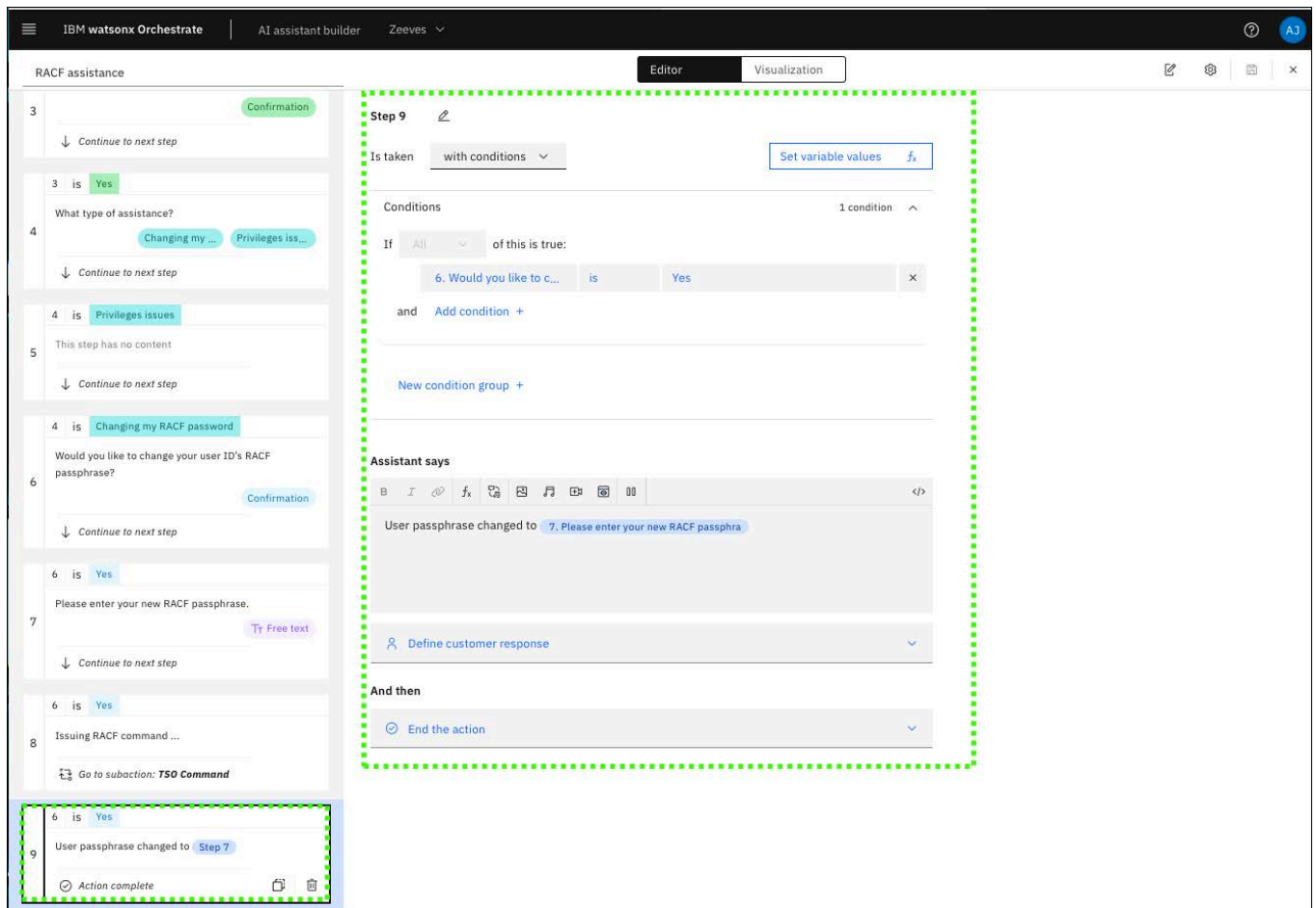
Step 9.: *With conditions*, change the condition to **Step 6. Would you like to change your user IDS passphrase equals Yes**, and inform the user the passphrase has been changed. Change the **And then** option to **End the action**.

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 ws changed.

The following video shows how the demonstration should work. The video does not have audio.

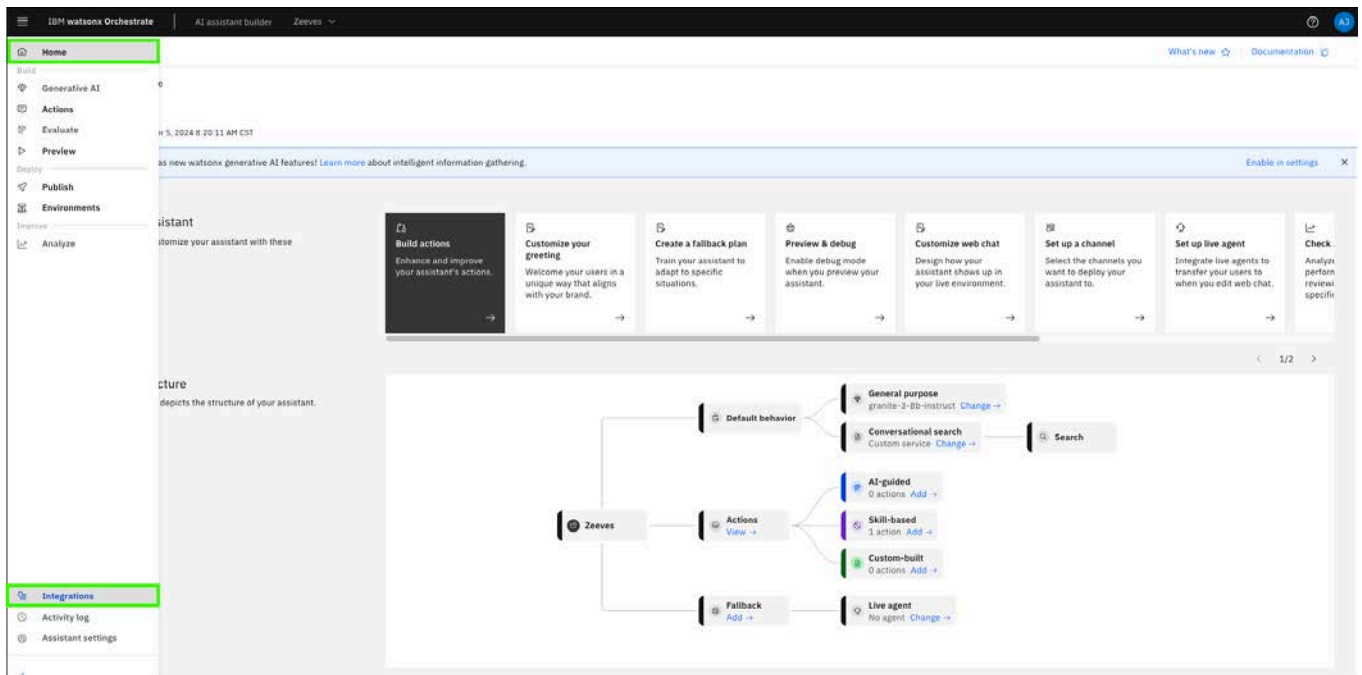
404 - Not found

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.

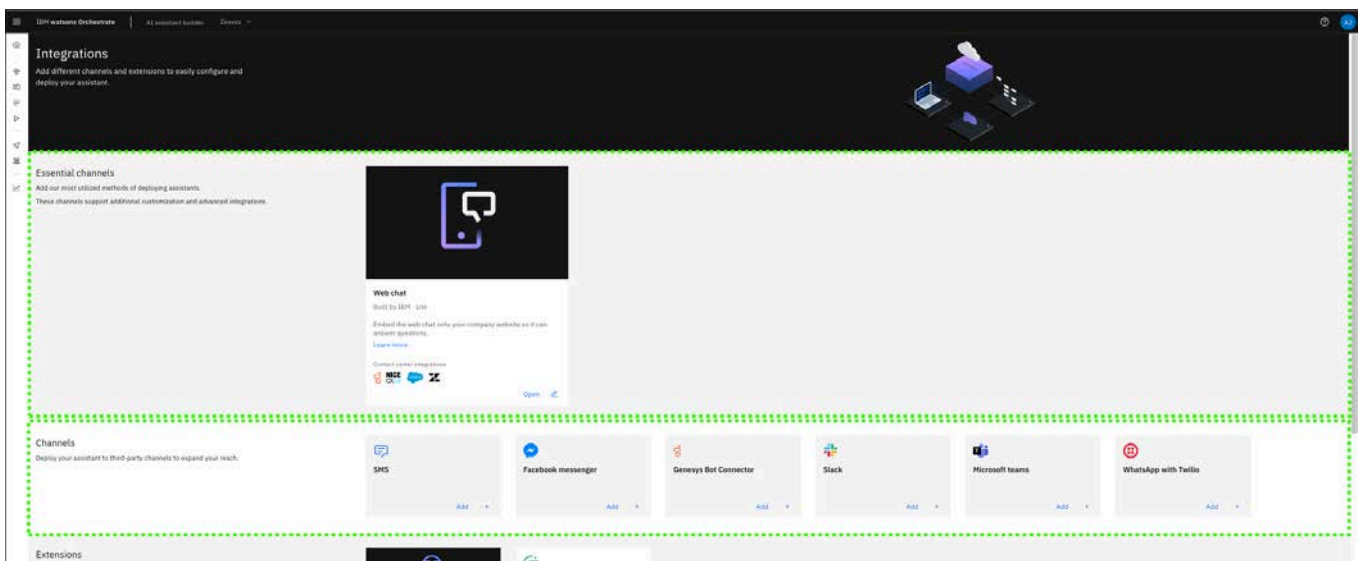
Adding other integrations

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.

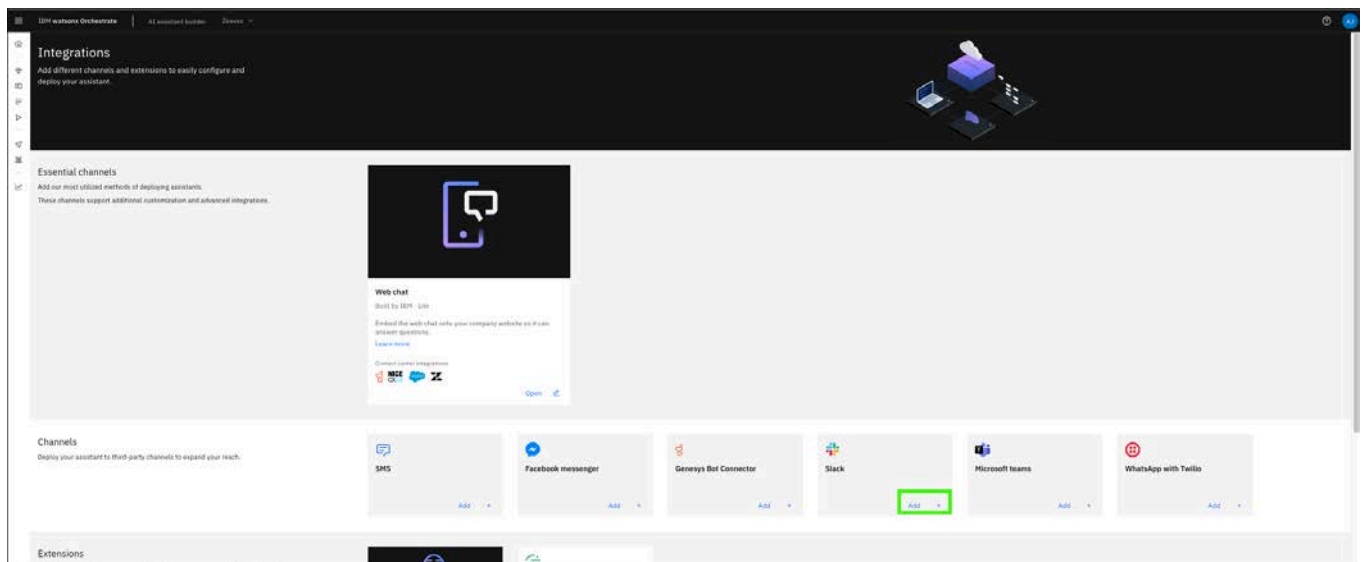
1. Hover over the **Home** (🏠) and click **Integrations**.



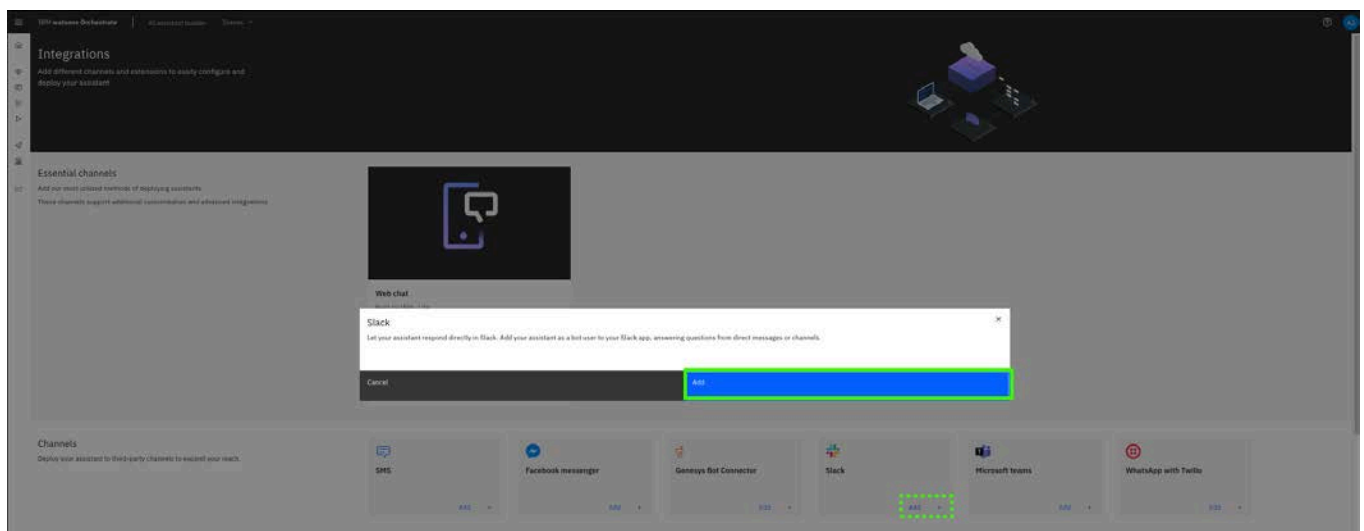
2. Explore the **Essential channels** and **Channels** sections.



3. Click **Add** on the **Slack** tile.

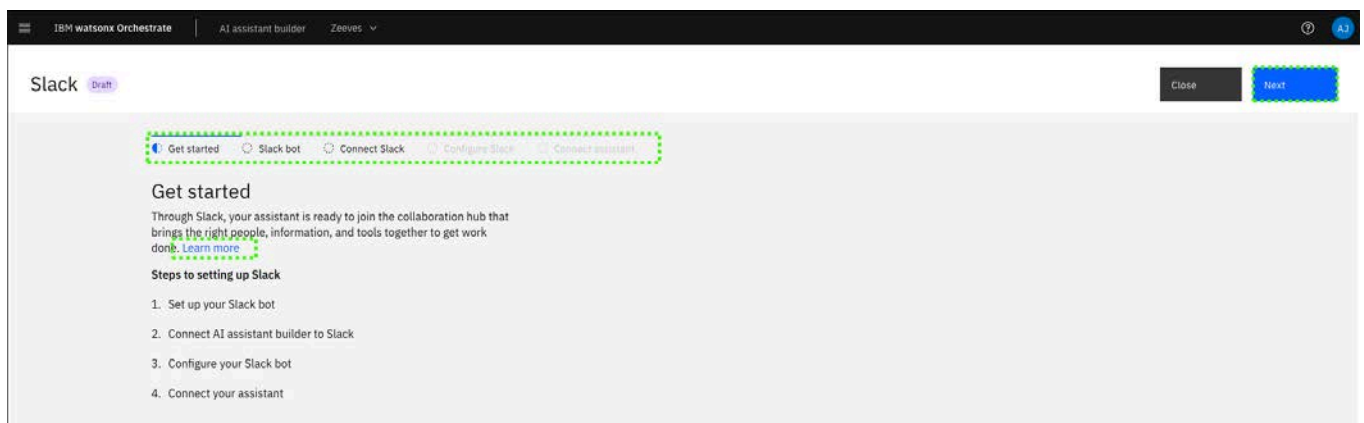


4. Click **Add**.



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.



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](#) badge 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](#)