IBM **TechXchange**

Orchestrate What's Next with AI Agents Hackathon

IBM

Orchestrate What's Next with AI Agents Hackathon

AI agents are intelligent systems that can perceive their environment, reason about it, and take actions to achieve specific goals. These agents are capable of operating autonomously or collaboratively, making them ideal for solving complex real-world problems.

In this hackathon, you will design and build a proof-of-concept agentic AI solution using IBM watsonx Orchestrate – a no-code and low-code platform designed to orchestrate AI agents across business workflows. With watsonx Orchestrate, you can create, deploy, and manage intelligent assistants and agents that automate tasks, streamline processes, and enhance productivity.

This hackathon guide will walk you through accessing watsonx technologies and building your solution.

Contents

Hackathon expectations

In this hackathon, participants are expected to design and build a **proof-of-concept agentic AI solution** using **IBM watsonx Orchestrate**. Your solution should align with one of the following themes:

- **Agent mode activated challenge**
  Use IBM watsonx Orchestrate to create a proof-of-concept that helps people research, collaborate, ideate, or innovate faster. Show the world how orchestrated AI agents can reduce barriers, spark new ideas, and turn concepts into impactful solutions.

- **Industry, innovation, and infrastructure challenge, brought to you by Call for Code**
  Reliable infrastructure, sustainable industrial practices, and technological advancement are needed to support economic development, create jobs, and improve human well-being for all. We're calling on you to build a proof-of-concept, agentic AI solution using IBM watsonx Orchestrate to address an issue that

falls under the United Nations Sustainable Development Goal (SDG) 9: Industry, innovation, and infrastructure.

You are **required** to use IBM watsonx Orchestrate to build your solution. The watsonx Orchestrate platform provide the foundation for building and orchestrating intelligent AI agents and assistants.

Participants may also **optionally** use the below listed watsonx products:

- IBM watsonx.ai is a powerful AI studio that supports the development of agentic AI solutions using a wide range of foundation models. Through the Prompt Lab, participants can experiment with IBM's Granite models and other leading models to create agents that understand and respond to natural language. IBM watsonx.ai can also serve as an inference provider for your agents, allowing them to generate responses, make decisions, and interact with users or systems intelligently.
- IBM watsonx Code Assistant to accelerate software application development. Generate code from natural language, integrate APIs, and streamline backend logic to support your agentic AI solution.

You may refer to example use cases to help you get started with your solution idea.

Note on data sets before you begin

Participants are required to bring their own datasets to build the solution aligning to your use case. As you collect data for your project, you'll want to use the best practices. Here are some helpful tips:

- Teams are responsible for ensuring data is compliant.
- Data from public websites may be used, if the terms allow for commercial use, but please keep a list of the websites you use.
- Do not use data or assets containing company confidential data, or any other data without permission from the data owner. Teams are responsible for getting approval.
- Do not use any client data.
- Do not use any data containing personal information (PI).
- Do not use data obtained from social media.

Get started with IBM watsonx

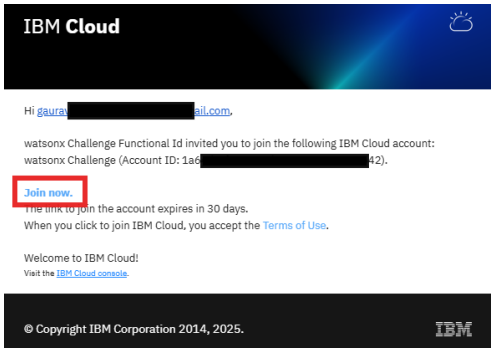
To access and use the watsonx products for this hackathon, participants must be registered for the hackathon and have access to the hackathon site. Once you have access to the hackathon site, follow the instructions on the "Complete the hackathon" page to request a pre-configured IBM Cloud account for your team. This account will provide the necessary environment to work with watsonx Orchestrate and other supporting watsonx products for this hackathon.

Accessing your hackathon IBM Cloud account

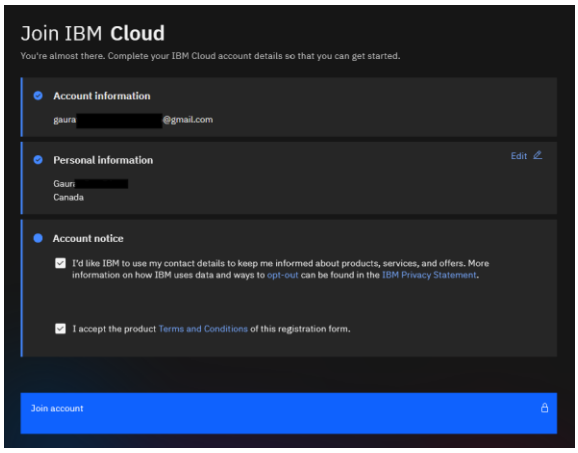
Once your team has been provisioned an IBM Cloud account, all team members will receive an email invite to join the cloud account. Follow the steps below to access your team's cloud account:

1. Check the email inbox you used to register for the hackathon and open the email you received from the IBM Cloud team about joining your cloud account. Please check your junk/spam folders if you are not able to find the email in your inbox. You can also quickly search for "IBM Cloud" to locate the email.

2. Click the **Join Now** button seen in that email. A new browser tab will open with the cloud account sign up page.

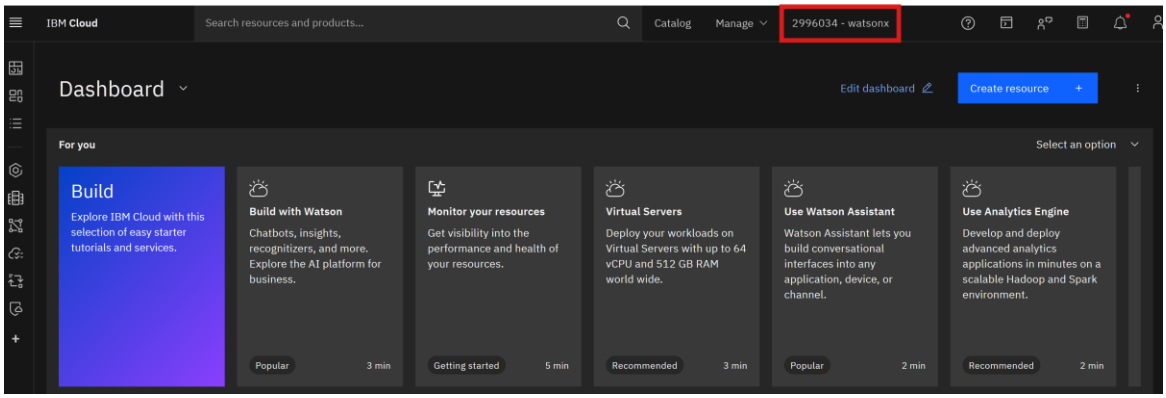


3. Review your account and personal information. Read and accept the Account notice and click the **Join Account** button.

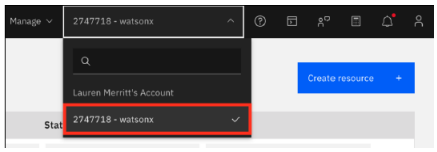


4. Complete the authentication process by clicking the Continue button.

5. After you authenticate successfully, you will be taken to the IBM Cloud dashboard.



6. If you have an existing personal IBM Cloud account for the same email/IBMid, sometimes you will be directed to your personal account. In this case, please switch your account to the **xxxxxxx - watsonx** account. Select your account drop-down at top-right of the dashboard and select watsonx account. Refer to the below image on switching accounts in your cloud dashboard.



Accessing and utilizing IBM watsonx products

To begin building your agentic AI solution, explore the capabilities and resources for each IBM watsonx product enabled for this hackathon. Note, **watsonx Orchestrate** is a **required** product in this hackathon and the other two products (watsonx.ai and watsonx Code Assistant) are optional.

1. IBM watsonx Orchestrate

   The core platform for designing, deploying, and managing AI agents that automate workflows and tasks using a no-code/low-code interface.

2. IBM watsonx.ai (Optional)

A powerful AI studio that enables you to experiment with foundation models like IBM Granite through the Prompt Lab. It can be used to enhance your agents with natural language understanding, intelligent decision-making, and dynamic response generation—serving as an inference engine to make your AI agents smarter and more interactive.

3. IBM watsonx Code Assistant (Optional)

   Enhance your solution with AI-powered code generation and automation, especially useful for developers looking to streamline coding tasks.

1. IBM watsonx Orchestrate

After successfully joining the IBM Cloud account, you can now access the watsonx Orchestrate to work on the platform and build your solution.

IBM watsonx Orchestrate is an intuitive, AI-powered platform that you can use to create, configure, and deploy intelligent agents that can automate business tasks. Whether you're automating repetitive workflows or building complex multi-agent systems, the platform is designed to support users of all skill levels.

With watsonx Orchestrate, you can:

- Build and deploy agents without writing code
- Automate tasks such as scheduling, data entry, and approvals
- Start small and scale up with advanced tools and integrations
- Explore the clean, guided UI that simplifies agent creation and management.

Get a quick overview of how to use watsonx Orchestrate.

Note on available watsonx Orchestrate services

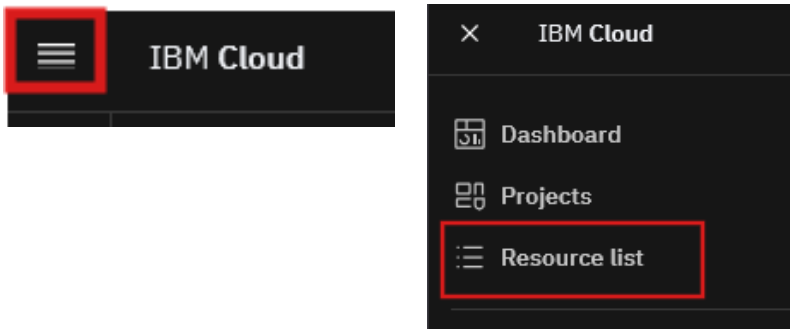
A **Trial** watsonx Orchestrate service is **pre-configured with only the services required** to complete the hackathon. If you notice a permission/access issue for any service or the cloud catalog, then they are not required/available for this hackathon.

These features/capabilities are out of scope for this hackathon:

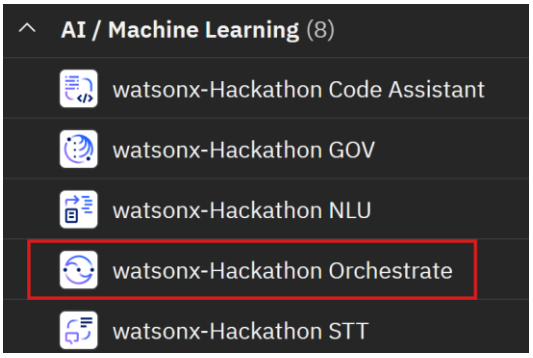
- Configuring a voice

Accessing watsonx Orchestrate

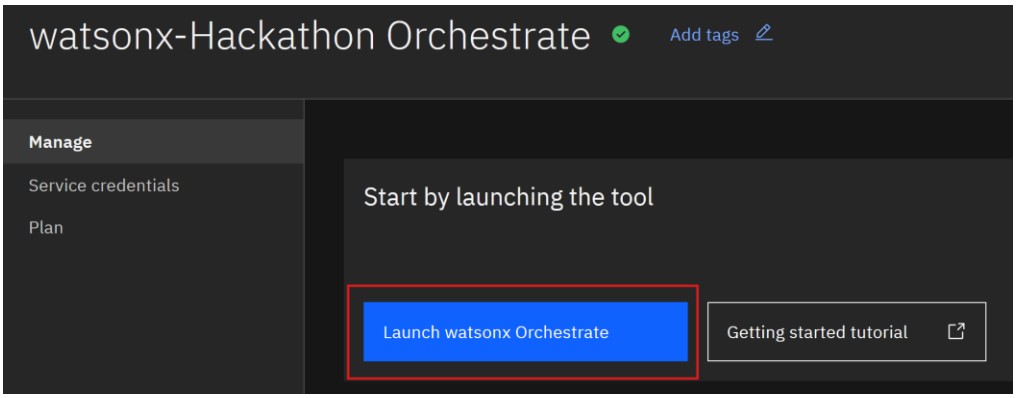
In your IBM Cloud account dashboard, select the **Navigation menu** on the top left of the dashboard and select the **Resource list** option.



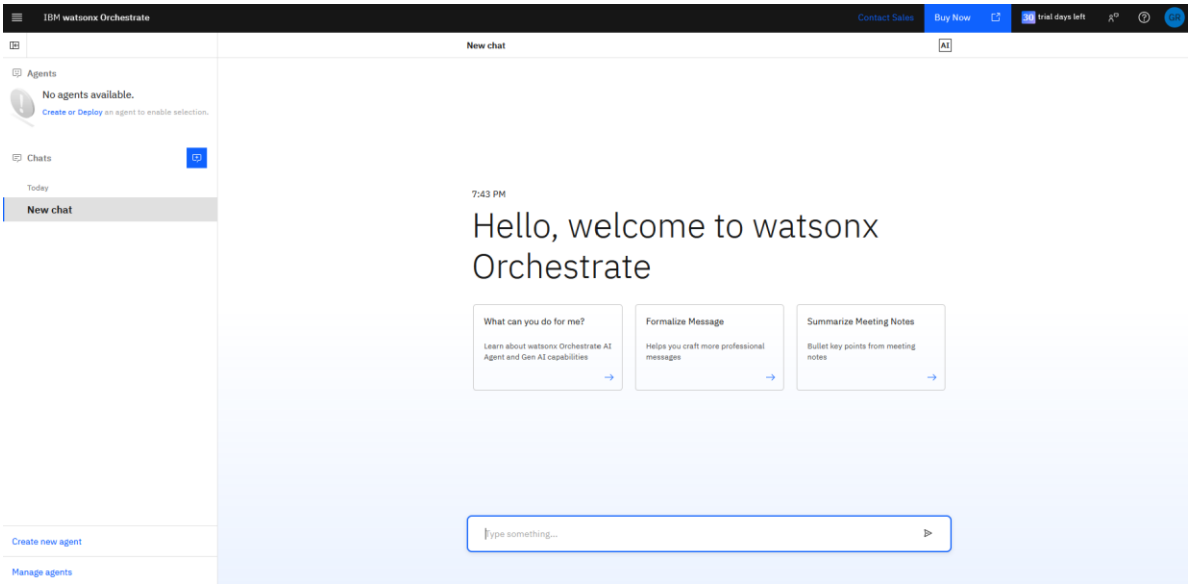
Expand the **AI / Machine Learning** section and select **watsonx-Challenge Orchestrate** service.



You will be navigated to **watsonx-Challenge Orchestrate** service instance dashboard. Select the **Launch watsonx Orchestrate** button.

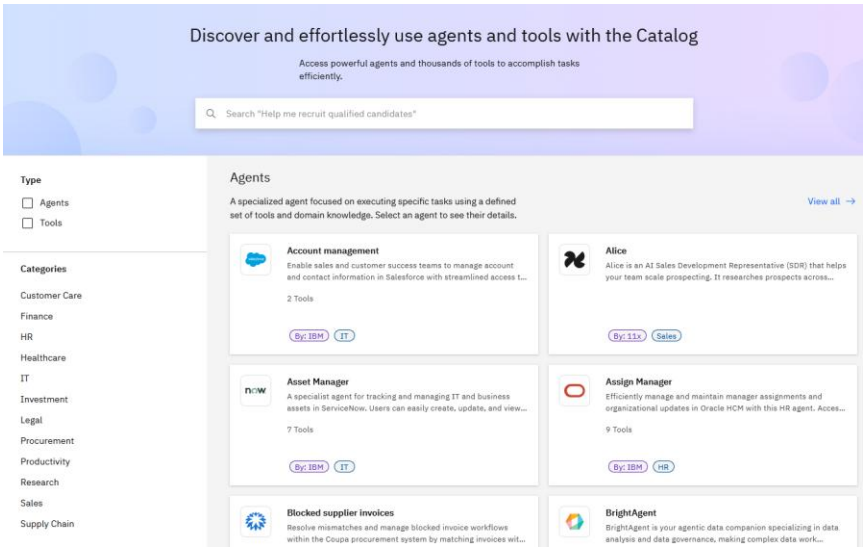


You will be navigated to the watsonx Orchestrate platform with a welcome message and a new chat window.



Discover the catalog

The IBM watsonx Orchestrate catalog is your gateway to a rich collection of prebuilt AI agents and tools, designed to support a wide range of business functions and use cases. Whether you're looking to automate tasks, enhance productivity, or integrate with backend systems, the catalog helps you find the right solutions quickly and efficiently. Learn more about discovering the catalog.



Building agents

In IBM watsonx Orchestrate, agents are a key component of the agentic AI framework, enabling you to create complex, dynamic systems that can adapt and respond to changing conditions.

By building agents, you can:

- Automate repetitive tasks

- Improve decision-making processes
- Enhance customer and employee experiences
- Increase operational efficiency

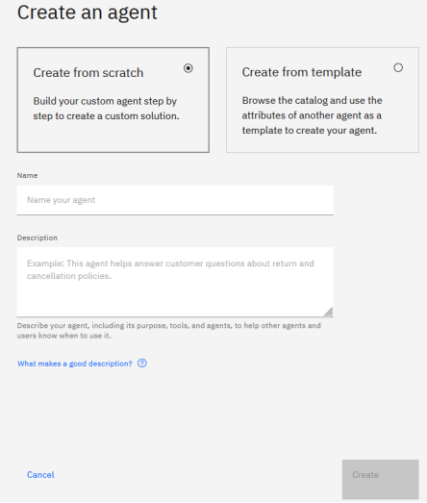
1. Prepare to build AI agents

Developing intelligent, scalable, and reliable agents in watsonx Orchestrate requires a strategic, multi-phase approach. It goes beyond writing prompts or connecting APIs. Successful agents need thoughtful planning, structured development, testing, and ongoing governance. The following overview breaks down each phase of agent development, highlighting key considerations. Learn more about preparing to build AI agents.

2. Creating and customizing agents

Creating and customizing an AI agent involves defining its purpose and personality through thoughtful descriptions, selecting the most suitable foundation model, and choosing a style that aligns with your brand or use case. You can also configure key elements like the welcome message and starter prompts to ensure the agent engages users effectively from the first interaction. Together, the configured components shape how the agent communicates, responds, and delivers value across conversations.

Learn how to create an agent from scratch or from a template.

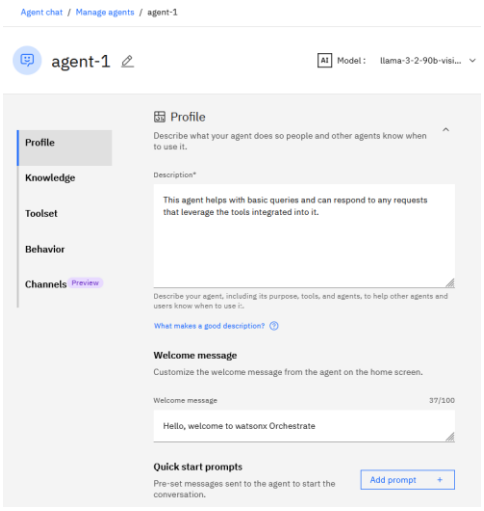


Expanding your agent's capabilities:

To enhance your agent's abilities, explore the following configuration options:

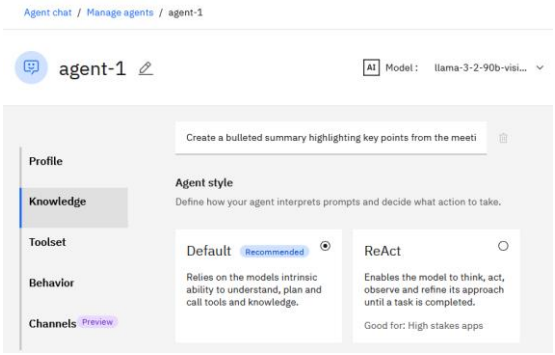
- **Define a profile**

  Provide a clear and specific description of the agent's purpose. This helps in multi-agent orchestration by enabling accurate selection based on capabilities. See Defining the description of your agent.



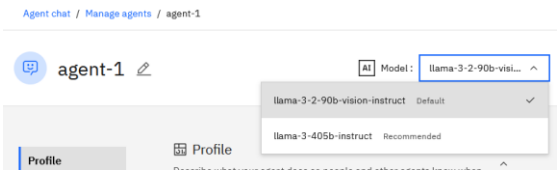
- **Define a style**

  Choose how the agent interprets and responds to user requests, selects actions, and engages with tools. See Choosing a style for your agent.



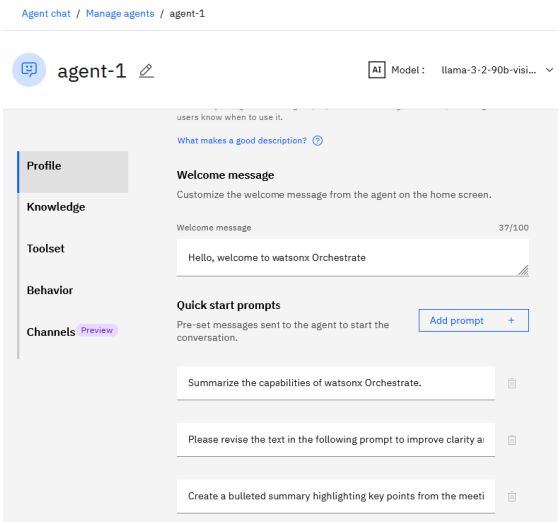
- **Choosing a foundational model**

  Select a foundation model from the Agent Builder UI to define your AI agent's core capabilities. See Choosing an LLM model.



- **Customize the welcome message and starter prompts**

  Customize the welcome message and starter prompts to guide users when they begin interacting with the agent. See Customizing the welcome message and starter prompts.

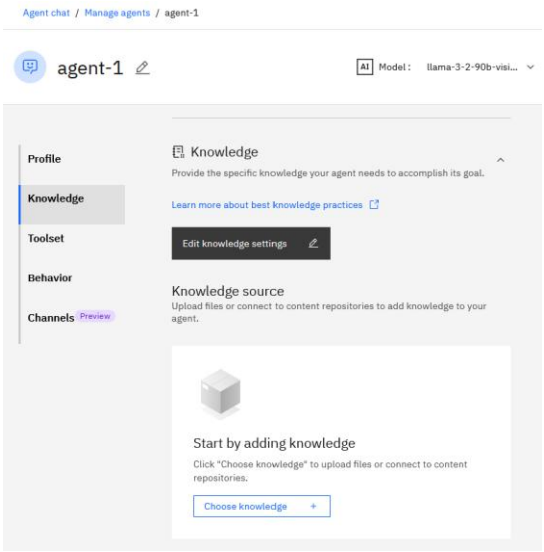


- **Customize the text below the welcome message**

  You can now modify the default light gray text below the welcome message only through ADK. For more information, see Customize the light gray text.

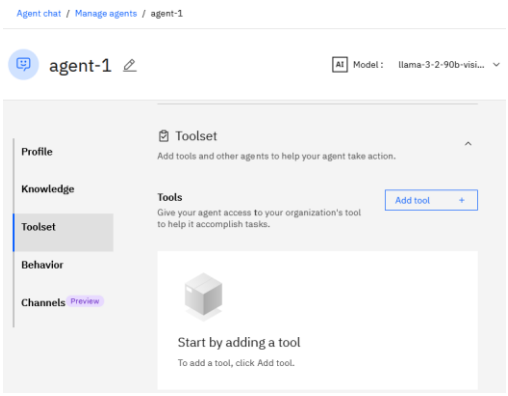
- **Add knowledge**

  Enhance the agent's domain expertise by adding contextual knowledge from files or content repositories. See Adding knowledge to agents.



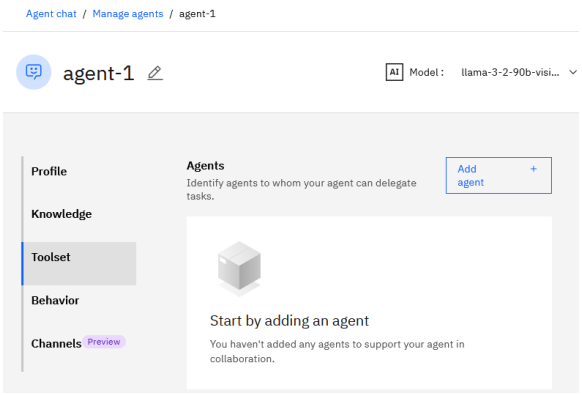
- **Add tools**

  Integrate tools to enable the agent to perform automated tasks such as retrieving data or sending emails. See Adding and managing tools for agents.



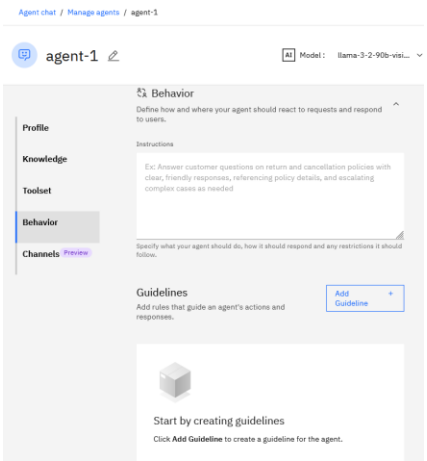
- **Add collaborator agents**

  Enable multi-agent orchestration by adding collaborator agents that work together to achieve shared goals. See Adding agents for orchestration.



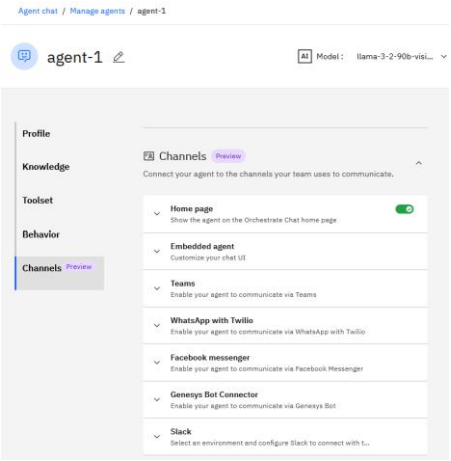
- **Add instructions**

  Define behavioral rules and interaction logic to guide the agent's responses and actions. See Adding instructions to agents.



- **Connect to channels**

  Make the agent accessible to your team by connecting it to various interaction channels. See Connecting to channels.

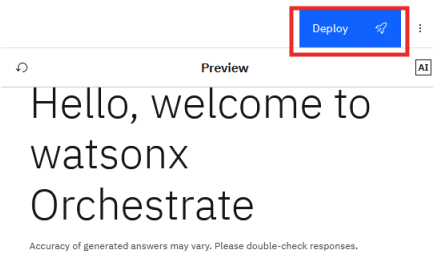


- **Configure rich responses**

  Enable rich responses to incorporate multimedia and structured elements, making AI interactions clearer, more efficient, and engaging for users. See Configuring rich responses from the AI assistant builder.

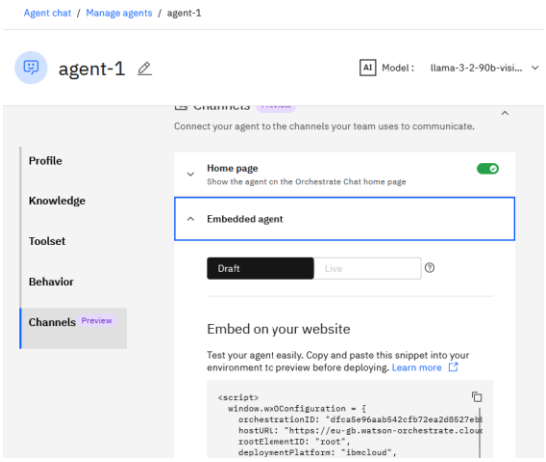
- **Deploy the agent**

  Finalize the setup by deploying the agent to make it available in live environments such as chat interfaces. See Deploying the agent.



- **Embedding your agents**

  You can embed your agents into various applications such as websites, Slack workspaces, or other messaging apps, allowing users to interact with them across different environments. See Using an agent in embedded chat.



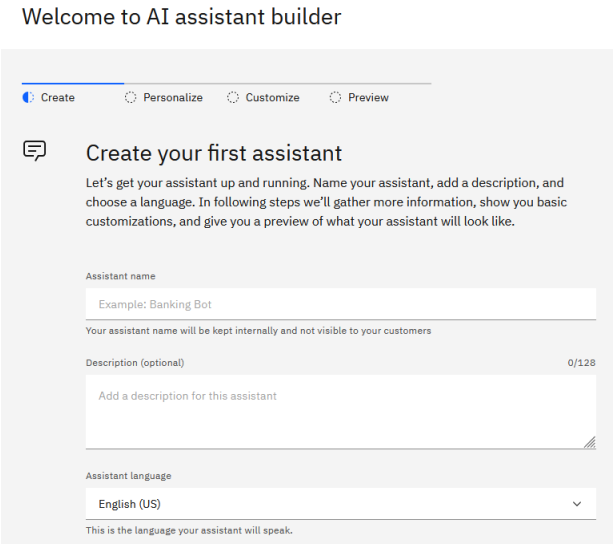
- **Building agents using the ADK**

  You can build powerful, customizable agents using the IBM watsonx Orchestrate Agent Development Kit (ADK). Learn more about using the ADK.

Building AI assistants

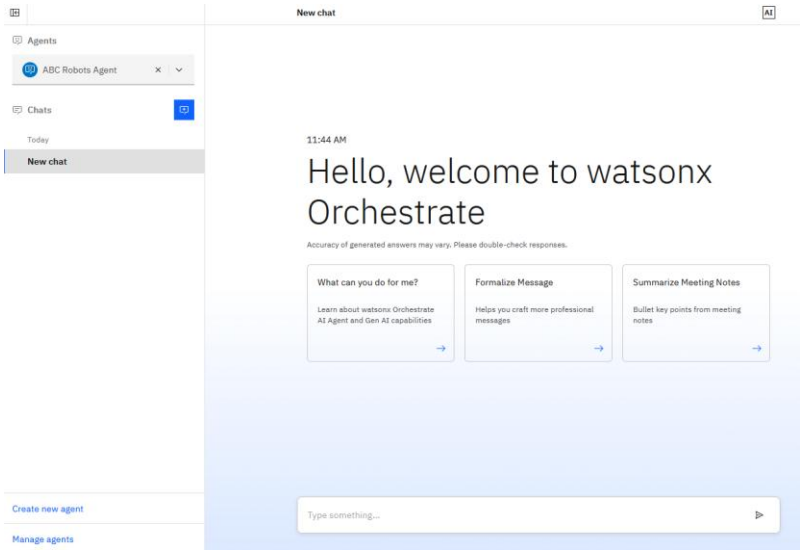
In IBM watsonx Orchestrate, you build the AI assistant by using AI assistant builder. AI assistant builder is a chat interface builder that helps to deploy an engaging and embedded chatbot experience. AI assistant builder integrates the power of large language models (LLMs) and conversational capabilities of watsonx Assistant to enable responsive and interactive conversation between the users and watsonx Orchestrate.

To learn more, see Building AI assistants in AI assistant builder.



Using agents in the chat

In IBM watsonx Orchestrate, agents collaborate to automate tasks and manage workflows. Learn more using agents in Orchestrate Chat.



Managing app connections

To use the external applications within IBM watsonx Orchestrate, you must establish a connection between them which acts as bridge enabling communication between watsonx Orchestrate and the external applications. Learn more about managing app connections and credentials.

Admin capabilities

IBM watsonx Orchestrate provides a robust set of administrative capabilities to help organizations manage their environments securely, efficiently, and at scale. As an administrator, you can control user access, configure integrations, and ensure compliance with enterprise policies.

- Managing users

  Add, remove, and manage users. Assign roles such as Admin, Builder, or User to control access and permissions.

- Managing settings

  Customize system configurations.

- Securing your instance

  Learn how to secure your watsonx Orchestrate instance.

Quick start hands-on exercises

Try the quick start hands-on exercises for sample use cases to get started with using watsonx Orchestrate:

- Ask-benefits health care exercise
- Develop agents with no code using watsonx Orchestrate
- AgentOps in watsonx Orchestrate: Observability for Agents with Langfuse and IBM Telemetry

2. IBM watsonx.ai (OPTIONAL)

After successfully joining the IBM Cloud account, you can now access IBM watsonx.ai to work on the platform and build your solution.

Note on IBM watsonx.ai service usage

For this hackathon, **$100 credits** will be automatically applied on the provisioned **IBM watsonx.ai platform**. This should be sufficient for designing and creating compelling submissions.

You will receive periodic email notifications about your **credit consumption** at the following usage levels: **25%, 50%,** and **80%**. Once you reach **100% usage**, your account will be **suspended**. You can appeal the suspension by completing the form shared in the account suspension notification email.

Please note that these email notifications are sent **once per hour**, so there is a possibility that you may **exhaust all your credits before receiving an alert**.

Please plan to use the watsonx.ai efficiently and back up your work accordingly. Refer **tips to work efficiently on watsonx.ai platform (Tokens and CUH explained)** and **saving your work**.

Important:

- **Foundation model inferencing** consumes tokens, which are measured as Resource Units (RUs). **1,000 tokens = 1 RU,** and each RU costs **$0.0001 USD.**
  Learn more about tokens and tokenization.
- If you are using **Jupyter Notebook editor on watsonx.ai,** consider selecting **a lower runtime environment** to avoid high resource consumption and quickly depleting your credits. Notebook runtimes are billed based on **Capacity Unit Hours (CUH)** at a rate of **$1.02 USD per CUH.**
  Learn more about capacity unit hours and watsonx.ai Studio pricing plans.

Note on available watsonx.ai services

The watsonx.ai platform is **pre-configured with only the services required** to complete the hackathon. If you notice a permission/access issue for any service or the cloud catalog, then they are not required/available for this hackathon.

These features/capabilities are out of scope for this hackathon:

- Agent Lab (Beta)
- Deploy on IBM Cloud/watsonx.ai (including Deployment space)
- Bring your own model
- Fine tuning models
- AutoAI pipeline
- SPSS Modeler

- Cloud Object Storage service

DO NOT USE the below listed models as they are out of scope for the hackathon and can negatively impact the judgment of your project submission.

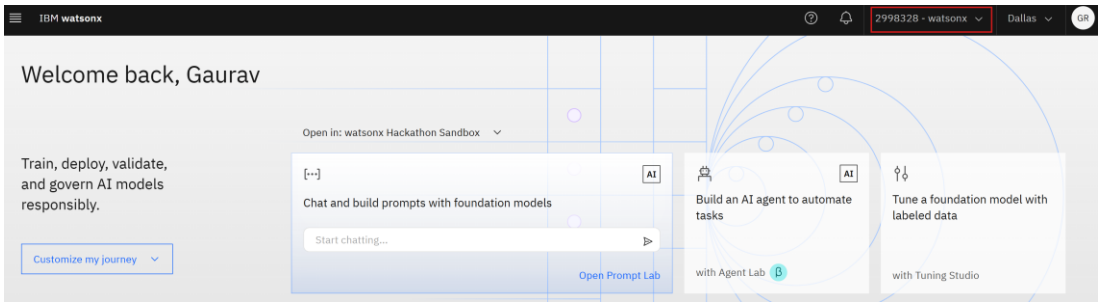
- llama-3-405b-instruct
- mistral-medium-2505
- mistral-small-3-1-24b-instruct-2503

The hackathon provisioned IBM Cloud account will be deactivated after the completion of the hackathon. Please plan to save your work at the end of the hackathon.

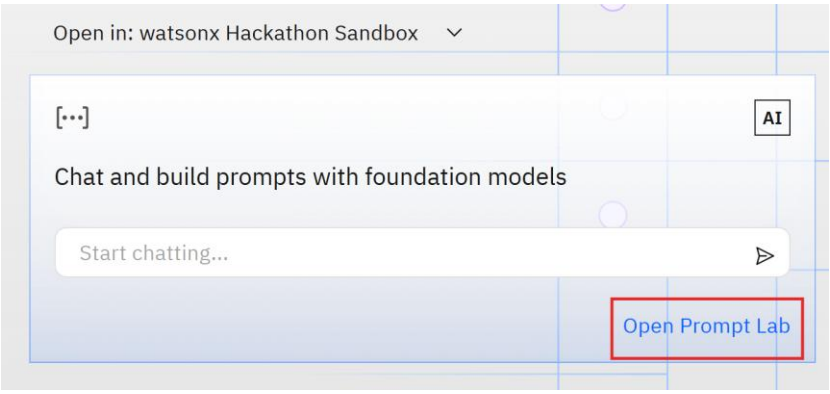
Access Prompt Lab on watsonx.ai

After successfully joining the IBM Cloud account, you can now access the Prompt Lab on watsonx.ai platform to work with the AI models supported on the platform and build your solution.

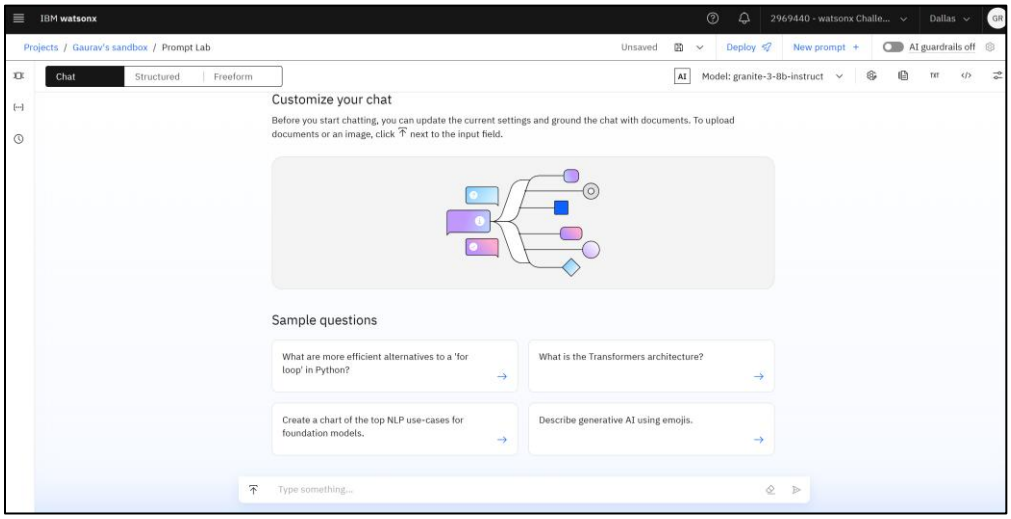
1. Log in to the watsonx.ai platform with the email you used to access your IBM Cloud account.
2. After successful authentication, you will see "Welcome to watsonx" widget. You have to read and accept the terms, data use policy and cookie use to proceed. You can either take the tour or skip it.
3. Next, you will see the watsonx.ai dashboard. Ensure the name of the account is "**xxxxxxx – watsonx**" and the region is "**Dallas**".



4. Select the **Open Prompt Lab** button on the "Chat and build prompts with foundational models" widget.



5. The “Welcome to Prompt Lab” tour will be displayed. You can take the tour to get a quick introduction or skip it.
6. The Prompt Lab Editor opens with a chat window to get you started with the prompt session.



Work with the watsonx.ai Prompt Lab

The watsonx.ai Prompt Lab is an easy-to-use prompt engineering interface where you can experiment prompting different AI foundation models, explore sample prompts, tune model parameters, integrate applications with an API endpoint, and save and share your best prompts.

Take a tour of the Prompt Lab and try the interactive demo.

You can access and use the AI models to build your innovative solution using Prompt Lab.

Prompt Lab editor

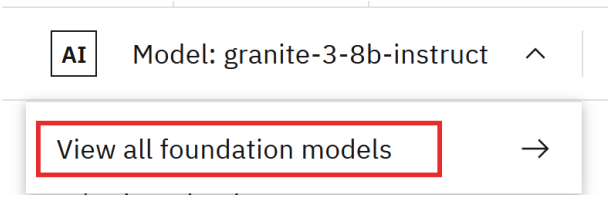
In the Prompt Lab, you can experiment with prompting different foundation models, explore sample prompts, as well as save and share your best prompts. The Prompt Lab editor is a great place to experiment and iterate with your prompts. Try the quick start lab.

However, you can also prompt foundation models in watsonx.ai programmatically. Refer to “Programmatic access (API/SDK)” section.

Selecting an AI model

A default AI model will be pre-selected in the Prompt Lab editor. You can either use the same model or change to a different model. To select a different model:

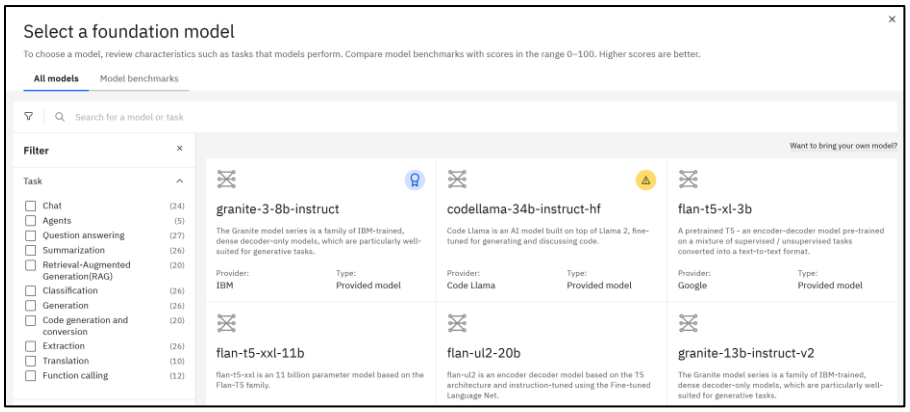
1. Select the AI Model drop-down menu at the top-right of the editor and select **View all foundation models**.



2. The **Select a foundation model** widget will appear. Clear the filters to see all the available models. You can use the filters to choose the right model for your solution building. You can select a model tile to learn about the model and use it. If you are limited to only "**Chat**" supported models, change the Prompt Lab editor to **Structured** or **Freeform** view and try selecting the models to see all the available model options.

Important: **DO NOT USE the below listed models as they are out of scope for the hackathon and can negatively impact the judgment of your project submission.**

- llama-3-405b-instruct
- mistral-medium-2502
- mistral-small-3-1-24b-instruct-2503



To understand how models can address your use case, including information on model modalities, supported languages, tuning, and indemnification, see our product documentation on choosing a model.

Note: Bigger models are not always better. Learn why smaller models can be better and more cost effective.

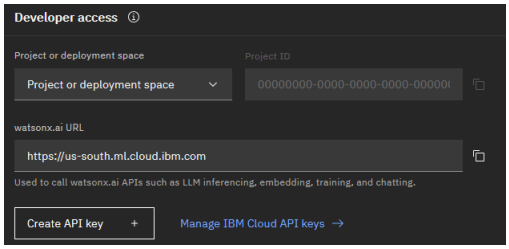
Programmatic access (API/SDK)

You can inference the watsonx.ai models with API or SDK requests.

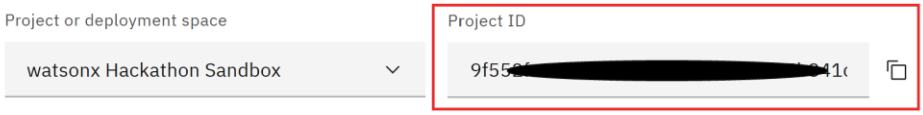
Developer access information

To use the supported watsonx.ai APIs/SDKs, you will need three values: a **project ID**, an **endpoint URL** and an **API key**.

- Go to watsonx.ai home page.
- Scroll down to the "**Developer access**" section.

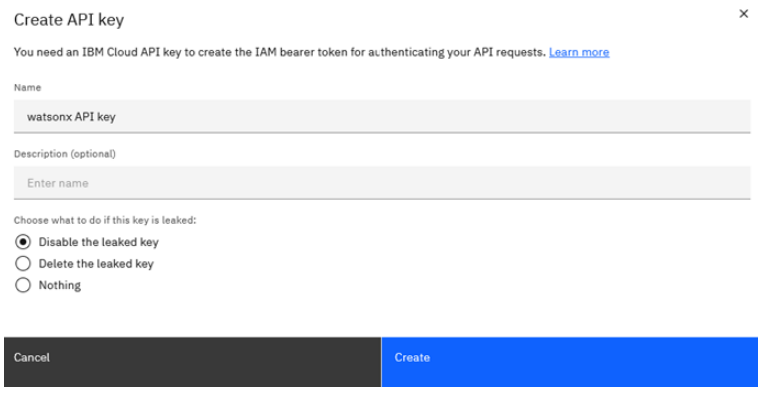


- Select the **Project or deployment space** drop-down and select the **watsonx Challenge Sandbox** option. A **project ID** will be displayed.
  Note: A space ID is **not required** as it is out of scope for the hackathon.

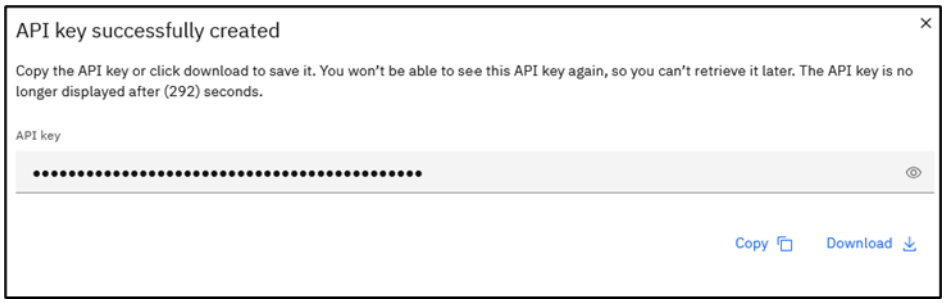


- A default **watsonx.ai endpoint URL** will be displayed for the Dallas region. Ensure the region is always set to **Dallas** at the top right of the watsonx.ai home page.

- Select the **Create API key** button. A **Create API key** widget will be displayed. Enter a name, provide optional description and choose the "Disable the leaked key" option. Click the **Create** button.



- An API key will be created successfully. Copy the API key and save it safely to use for calling the API/SDK. You can also download and save the file in a secure path in your system.



watsonx.ai programmatic options

There are multiple options to help you get started using watsonx.ai APIs/SDKs.

Option 1: Prompt Code on Prompt Lab

Refer to the access prompt code instructions to learn how to quickly get access to the text generation API within the watsonx.ai Prompt Lab.

Option 2: Different watsonx.ai API capabilities

Explore and leverage different watsonx.ai API capabilities in your solution.

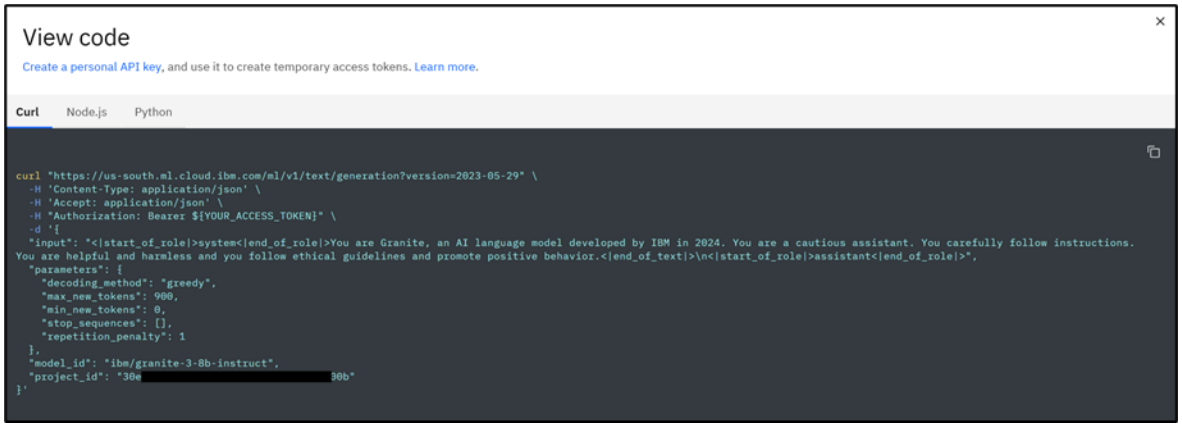
- Chat
- Agent-driven chat
- Tool calling
- Text generation
- Time series
- Text rerank
- Embeddings
- Text extraction

Refer to supported API functionality by model here.

Access the prompt code (API) from Prompt Lab editor

To prompt an AI model programmatically, you can view and copy the prompt code by selecting the **View code** icon </> at the top-right of the prompt lab editor.

The prompt code is available as a Curl, Node.js and Python.



You will require an IAM access token to authorize the prompt code and need to replace **${YOUR_ACCESS_TOKEN}** placeholder in the prompt code. You can create an IAM access token using an API key.

- **API key:**

  Refer to Developer access information to get an API key.

- **Generate IAM Access Token:**

  Programmatically generate an IAM access token with the API key using the following cURL command:

```
curl -X POST 'https://iam.cloud.ibm.com/identity/token' -H 'Content-Type: application/x-www-form-urlencoded' -d 'grant_type=urn:ibm:params:oauth:grant-type:apikey&apikey=MY_APIKEY'
```

- **curl -X POST** → Specifies an HTTP **POST** request.
- **URL ("https://iam.cloud.ibm.com/identity/token")** → The endpoint to request an authentication token from IBM Cloud.
- **-H "Content-Type: application/x-www-form-urlencoded"** → Sets the request header to indicate that the data is sent in form-encoded format.
- **-d (Data Payload)** → Sends the required data:
- **grant_type=urn:ibm:params:oauth:grant-type:apikey** → Specifies the OAuth grant type as API Key.
- **apikey=MY_IBM_CLOUD_API_KEY** → Replace MY_IBM_CLOUD_API_KEY with your actual IBM Cloud API key.

```
Expected Response:

{
    "access_token": "eyJhbGciOiJIUz......sgrKIi8hdFs",
    "refresh_token": "not_supported",
    "token_type": "Bearer",
    "expires_in": 3600,
    "expiration": 1473188353,
    "scope": “ibm openid”
}
```

Note: An IAM token is valid for up to 60 minutes, and it is subject to change. When a token expires, you must generate a new one. Use the property "*expires_in*" for the expiration of the IAM token that you have just created.

watsonx.ai AI agent libraries and tutorials

Explore the watsonx.ai supported AI agent framework libraries and tutorials to help you get started building your AI agent solution.

- LangChain
- LangGraph
- LlamaIndex
- CrewAI
- BeeAI
- AutoGen
- Python SDK
- Node.js SDK

Quick start hands-on exercises

Try the quick start exercises and notebooks for sample use cases to get started with using watsonx.ai.

Important notes:

- Refer to developer access information section to use watsonx.ai credentials as you try the exercises.
- Some of the exercises could include the usage of old model version. You can replace them with newer versions for better performance and output. To check the latest supported AI models on watsonx.ai, either follow selecting an AI model on Prompt Lab or refer to supported foundation models on watsonx.ai.
- The hackathon provisioned cloud accounts **do not support solution deployment.** You can run your solution deployment locally on your machine and showcase them in your submissions.
- **Foundation model inferencing** consumes tokens, which are measured as Resource Units (RUs). **1,000 tokens = 1 RU,** and each RU costs **$0.0001 USD.**
  Learn more about tokens and tokenization.
- If you are using **Jupyter Notebook editor on watsonx.ai,** consider selecting **a lower runtime environment** to avoid high resource consumption and quickly depleting your credits. Notebook runtimes are billed based on **Capacity Unit Hours (CUH)** at a rate of **$1.02 USD per CUH.**
  Learn more about capacity unit hours and watsonx.ai Studio pricing plans.

watsonx.ai Prompt Lab app templates:

- LangGraph LLM app template with function calling capabilities (base template)
- LlamaIndex Workflow LLM app template with function calling capabilities (base template)
- CrewAI LLM app template with function calling capabilities (base template)

- arXiv Research agent (community template)
- Agentic RAG LangGraph template (community template)

BeeAI Agent Framework:

- BeeAI framework examples

LangChain and LangGraph:

- Create a LangChain AI Agent in Python using watsonx
- Build a RAG agent using LangGraph to answer complex questions
- Build a LangChain agentic RAG system using the Granite model in watsonx.ai
- Use watsonx, and LangChain Agents to perform sequence of actions
- Use watsonx, and LangChain to make a series of calls to a language model
- arXiv Research agent
- Base LangGraph LLM app template with function calling capabilities

LlamaIndex:

- Use watsonx and LlamaIndex for Text-to-SQL task
- Use watsonx, and `llama-3-1-70b-instruct` and LlamaIndex to make simple chat conversation and tool calls
- LlamaIndex Workflow LLM app template with function calling capabilities

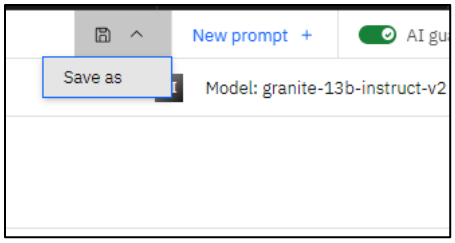
CrewAI:

- Leveraging CrewAI and IBM watsonx
- Build an agentic framework with CrewAI memory, i18n, and IBM watsonx.ai
- Base CrewAI LLM app template with function calling capabilities

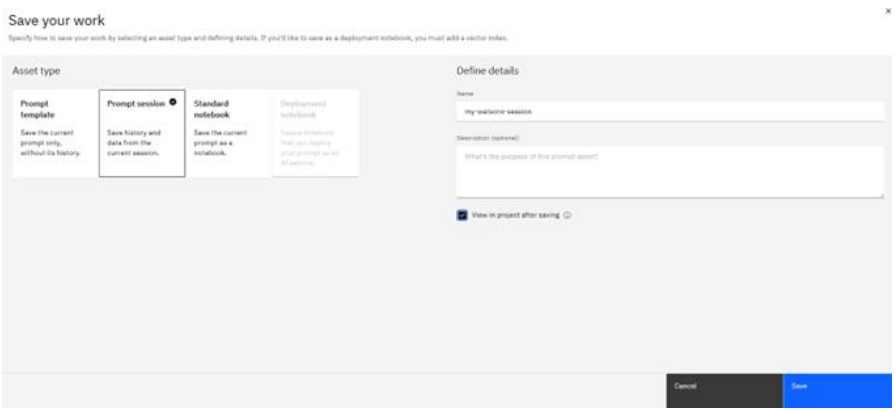
Save your Prompt Lab session

You can save your Prompt Lab editor session for later use.

1. At the top of the Prompt Lab screen, select the **Save work** dropdown button and then select the **Save as** option.



2. A **Save your work** widget will appear. Select **Prompt session** under the **Asset type** option.
3. Enter a **name** and check the **View in project after saving** option under the **Define details** section.
4. Finally, click the **Save** button. Once you save, you will see the saved work under the **Assets** tab



You can also save your work as:

- **Prompt template** to save only the current prompt without its history and selecting a **Task** suitable for your prompting.
- **Notebook** to continue prompting on a Jupyter Notebook environment. Prior knowledge of notebooks and Python programming language would be helpful to work with a Jupyter notebook. Read more about notebooks.

Save your work on watsonx.ai

Make sure to save any work you want to retain for your records. IBM Cloud accounts will be deactivated at the end of the hackathon. Follow the steps below to save your work:

1. Go to your project's 'Overview' tab.
2. Select the 'Export or import project' drop down below the Bell icon in the top menu bar.
3. Click the 'Export project' option. This will open 'Export project to desktop' screen.
4. Select all the assets shown in your project (Work saved as Project session cannot be exported) and click 'Export' on the bottom-right of the screen.

5. The next screen will ask for confirmation that all sensitive information has been removed.
6. Click on ‘Continue export’.
7. The download (zip) will be initiated, and the file will be saved on your computer.

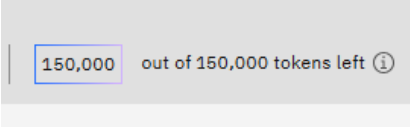
3. IBM watsonx Code Assistant (OPTIONAL)

After successfully joining the IBM Cloud account, you can now access the watsonx Code Assistant to build your solution.

Note on available watsonx Code Assistant services

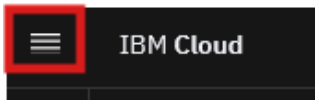
A **Trial** watsonx Code Assistant service is **pre-configured with only the services required** to complete the hackathon. If you notice a permission/access issue for any service or the cloud catalog, then they are not required/available for this hackathon.

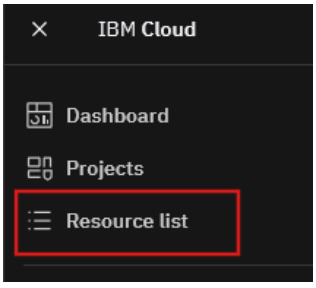
The Trial WCA service comes with a limit of 150,000 tokens, and additional tokens will not be provided for the hackathon. You can track your service token consumption on WCA service landing page or by selecting the **“Tokens and resource units”** tile on the home page**.**



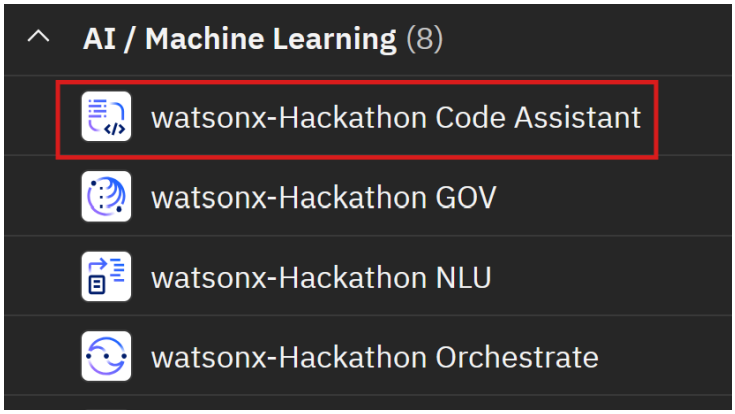
Access watsonx Code Assistant (WCA)

- In your IBM Cloud account dashboard, select the **Navigation menu** on the top left of the dashboard and select the **Resource list** option.

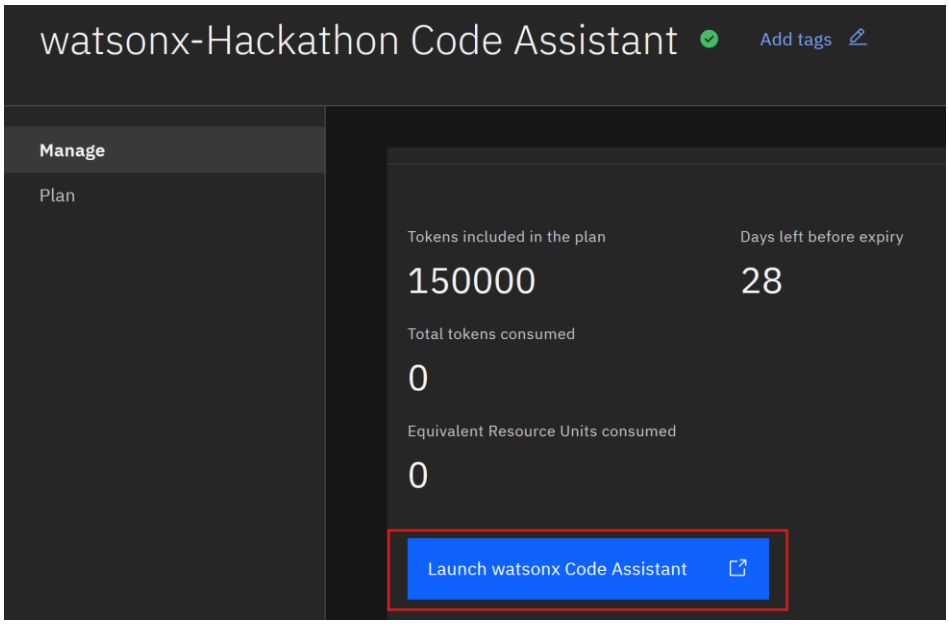




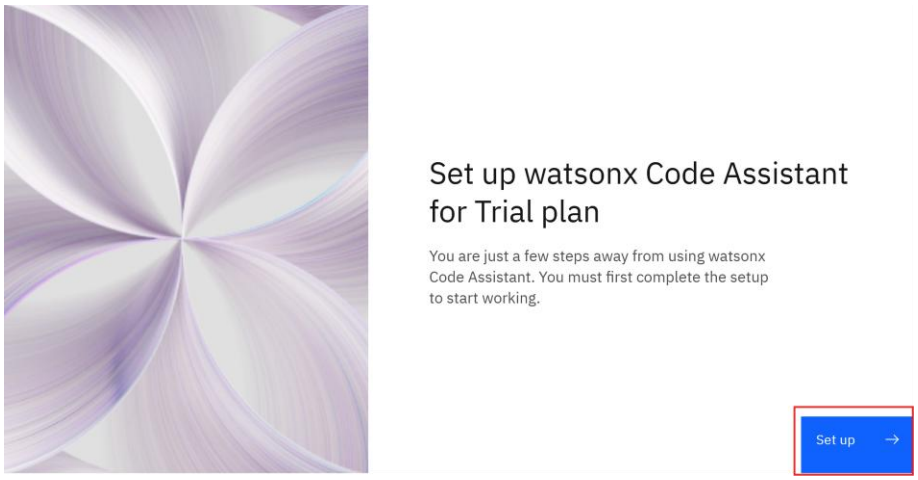
- Expand the “AI / Machine Learning” section and select watsonx-Challenge Orchestrate service.



- You will be navigated to **watsonx-Challenge Code Assistant** service instance dashboard. Select the **Launch watsonx Code Assistant** button.

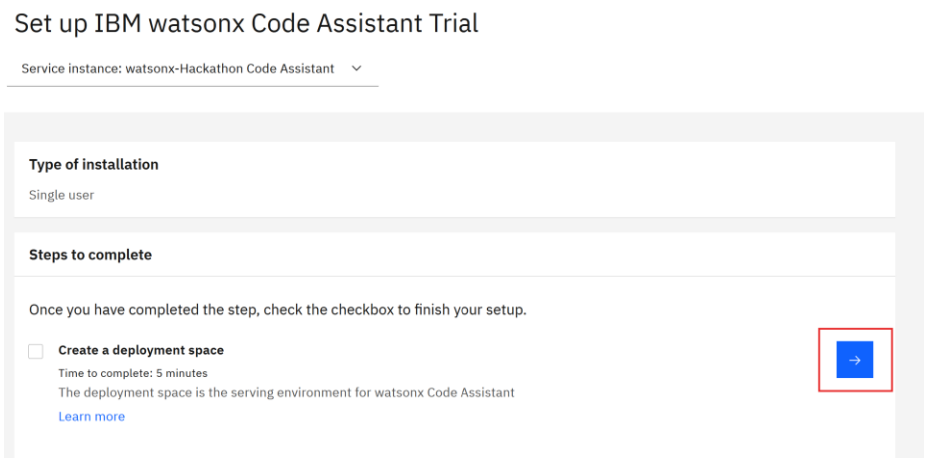


- You will be navigated to the watsonx Code Assistant dashboard, and a setup widget will be displayed. Select the **Set up** button.

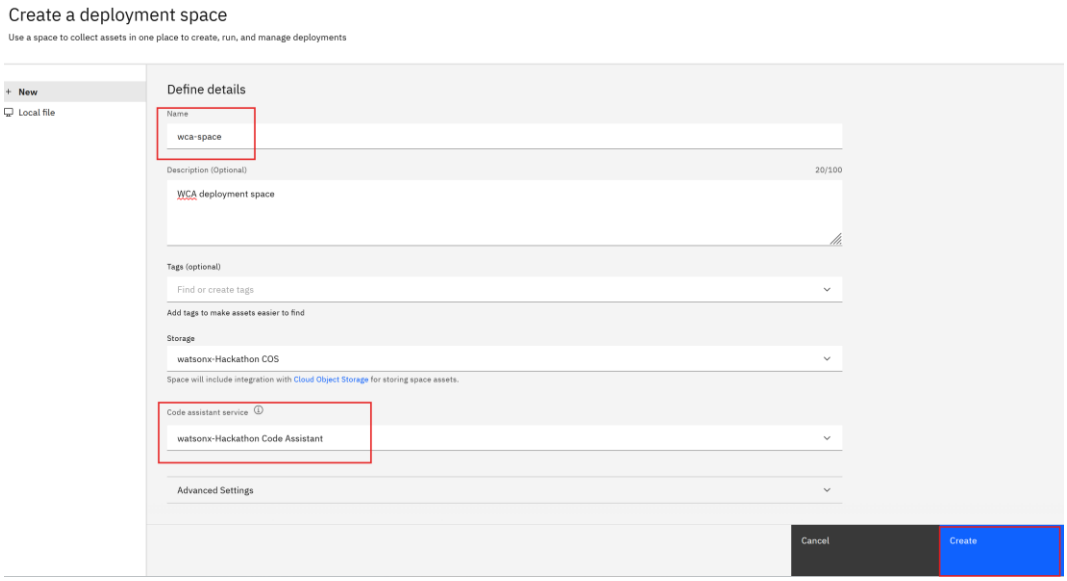


- A deployment space needs to be created to use the WCA tool. In most cases, a deployment space will be preconfigured. If it is not, you will need to manually create a deployment space to complete the

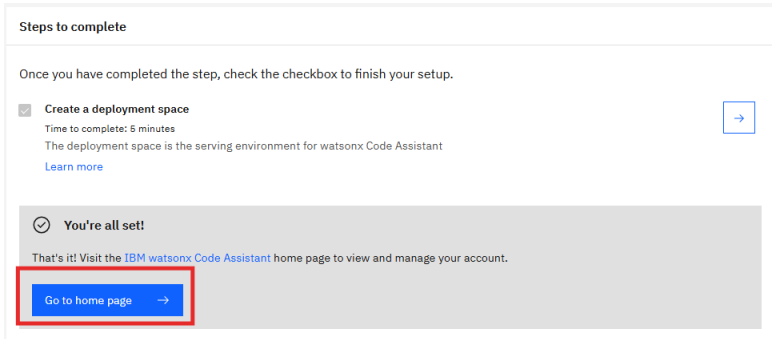
setup. Click the **Create a deployment space** button.



- Enter a name and optional description. Select **watsonx-Hackathon Code Assistant** for the "Code assistant service" drop-down. Click **Create** button.



- Once the setup is completed, you will see "**You're all set**" confirmation. Select the **Go to home page** button.



- You will be navigated to the home page, and a welcome message will be displayed. You can either take the tour or skip it. Now you are all set and ready to install the WCA extension in your preferred IDE.

Installing the WCA IDE extension

The watsonx Code Assistant (WCA) supports extension installation for Visual Studio Code and Eclipse IDEs. Select your preferred IDE to view the installation instructions.

- Visual Studio Code
- Eclipse

Note: An IBM Cloud API key would be required to login and use the extension. Refer to Create an API key instructions to generate one and use it.

Using the WCA IDE extension

Try the interactive demo.

Refer to the following instructions to learn and try different capabilities of the product:

- Generate code suggestions
- Explaining code
- Documenting code
- Generating unit tests
- Translating code from one language to another
- Prompting guide

Appendix

Example use cases

You are not limited to these ideas, but here are several examples of how you could apply IBM watsonx to solve a specific issue within your chosen theme:

Agent mode activated challenge

- **AI-powered market research analyst**: Build an AI agent that automates competitive and market research by aggregating data from reports and internal systems. Analyze trends, identify business opportunities, and generate concise insight summaries to support faster, data-driven decision-making.
- **Inclusive knowledge and innovation hub**: Demonstrate how agentic AI can democratize access to research knowledge and innovation opportunities, fostering inclusive participation. The solution might include agents that connect researchers with mentors, recommend relevant open-access papers or datasets, or automate collaboration workflows.
- **Smart research infrastructure assistant**: Explore how agentic AI can make scientific research

environments more resilient and accessible, reducing waste and downtime while supporting innovation. The solution might include agents that help universities and labs manage research infrastructure more efficiently, schedule lab equipment usage, monitor resource availability (energy, bandwidth, cloud compute), and orchestrate maintenance alerts.

- **Innovation pipeline accelerator**: Orchestrated agents track an idea's lifecycle: capturing and clustering ideas, scoring feasibility and testing it with technical experts, assigning tasks, validating with customer feedback, or aligning with compliance/legal requirements.
- **Contract intelligence orchestrator**: Build agents that review contracts such as Non-Disclosure Agreements (NDAs), Master Services Agreement (MSAs**)**, or Statements of Work (SOWs) across a portfolio, and highlight risks, compliance gaps, or negotiation opportunities.
- **Ecosystem matchmaker**: Identifies potential business-to-business partners, startups, or labs that align with enterprise strategies. Suggests co-innovation or joint go-to-market opportunities.
- **Patent and IP discovery agent**: Build an AI agent that searches global patent databases, identifies overlapping technologies, and flags potential collaboration or licensing opportunities. The agent summarizes findings to guide R&D and legal teams during innovation planning.
- **Collaborative proposal generator:** Design an AI agent that automates proposal and grant creation by gathering team inputs, aligning tone and structure, and generating coherent drafts. Streamline collaboration across departments, reducing turnaround time and improving proposal quality.

Industry, innovation, and infrastructure challenge, brought to you by Call for Code

- **AI for smarter systems**: Highlight how AI agents can make infrastructure smarter, safer, more resilient or efficient, or more adaptive to future challenges. Using IBM watsonx Orchestrate, agents might collect an analyze data from sensors, inspection reports and environmental systems to monitor infrastructure health, predict maintenance needs, prioritize repairs, coordinate resources automatically, or support planning for new sustainable transport, energy, or urban systems.
- **Green campus assistant**: Build multi-agent AI workflows that can encourage collective climate action, reduce environmental impact, or make sustainability an engaging part of student/campus life. The solution might track energy usage in dorms or labs (via simulated data feeds), recommend greener transportation options such as bike-sharing or bus routes, or gamify eco-friendly behaviors with leaderboards and points.
- **Green freight route optimizer**: Build AI agents that can optimize delivery routes in response to traffic and weather to balance cost, speed, and carbon impact.
- **Predictive equipment maintenance agent**: Monitor IoT data by agents that track industrial equipment usage, predict failures, auto-schedule repairs, or order parts before breakdowns occur.
- **Circular manufacturing planner**: Use AI agents to track raw materials, suggest recycling loops, or design closed-loop supply chains for industrial parks.
- **Construction project orchestration agent**: Design an AI agent that coordinates schedules, budgets, and material deliveries across construction projects. It uses data from digital twins to identify risks and suggest timeline optimizations. The agent improves project efficiency and reduces waste in infrastructure development.
- **Resilient port operations**: Build orchestrated agents that can analyze shipping schedules, weather risks, port congestion, or customs data to optimize throughput and avoid bottlenecks.
- **Smart waste-to-energy planner**: Build AI agents that identify industrial waste streams or recommend pathways to convert them into bioenergy or secondary materials.
- **Sustainable manufacturing optimization agent**: Design an AI agent that analyzes manufacturing

production data to identify inefficiencies in energy use, materials, and equipment performance. Recommend process adjustments, schedule maintenance, or track key sustainability performance indicators. Manufacturers can reduce emissions, lower costs, and improve operational resilience.

watsonx Orchestrate hands-on exercise

Overview

In this activity, you'll step into the role of an AI solution developer working within the IBM watsonx Orchestrate platform. The lab is designed to simulate a real-world healthcare use case by guiding you through the process of creating, configuring, and deploying intelligent agents from scratch. This activity demonstrates how intelligent agents can be configured and deployed in watsonx Orchestrate to address healthcare-specific use cases. Upon the completion you will:

- Understand the end-to-end agent creation process
- Learn how agentic AI uses context, memory, and tools to enable adaptive, natural multi-turn interactions
- Be familiar with OpenAPI-based tool integration
- Get practical experience with multi-agent solutions
- Gain experience testing and validating agent behaviour in practical scenarios

You'll be completing two core activity parts:

- **Build your own AskBenefits agent** – a digital agent that answers questions about medical procedures and health plan benefits. It detects if a procedure is past, future, or overdue and uses the right tools to respond. It can retrieve past procedure data, flag overdue care, guide upcoming procedures, help schedule/reschedule appointments and showcase HR policies regarding eligibility criteria, dependant coverage, claims process and more (from the uploaded knowledge).
- **Enable collaboration between AskBenefits and pre-configured AskDental agent** – an agent that supports questions about dental benefits by referencing a PDF document. It can answer a range of questions about coverage and plan details.

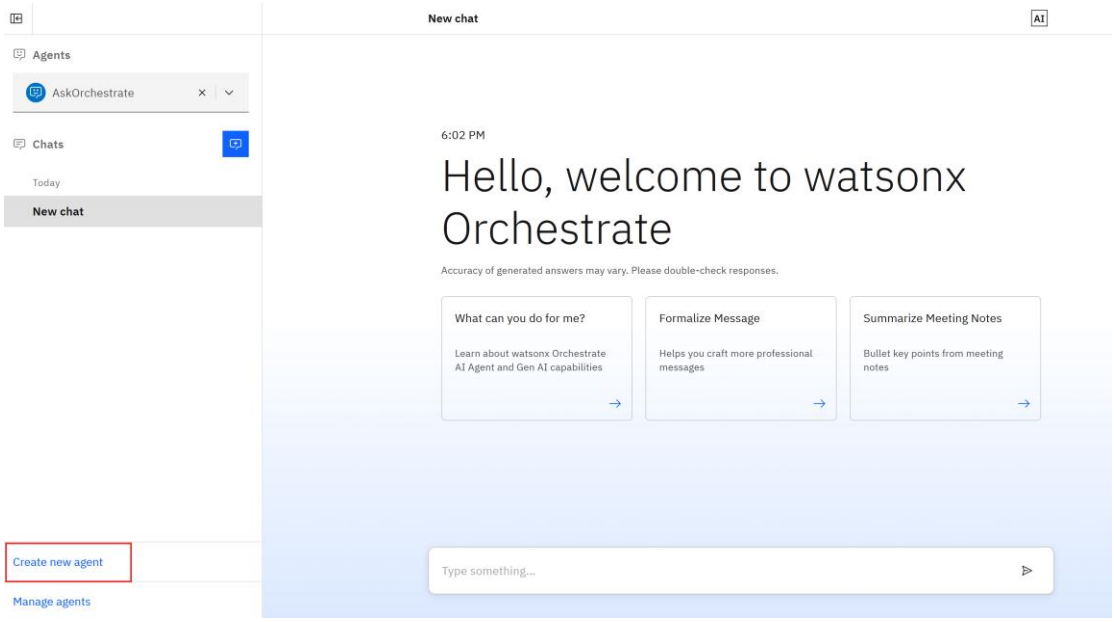
What you'll need

- The OpenAPI specification file (ask_benefits_tools.yaml) – Click here to download
- The HR policy document on medical benefits (HR_Policy_Medicalbenefits.pdf) – Click here to download
- The dental benefits file (dental_benefits_summary.pdf) – Click here to download

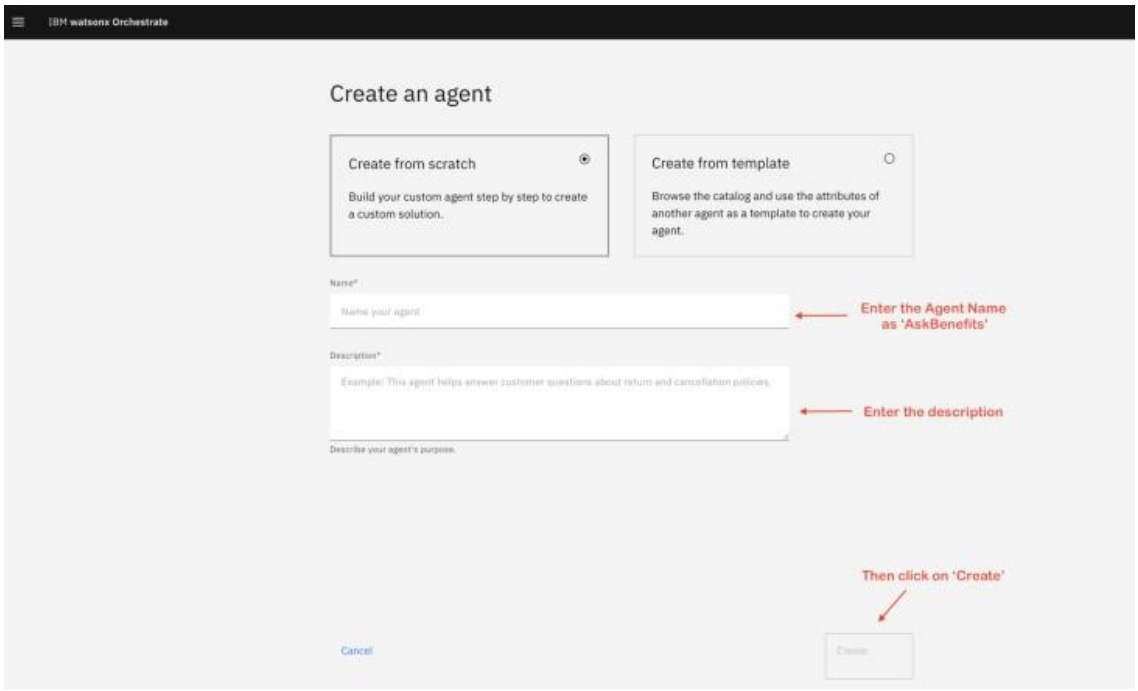
Part 1: Build and engage with a new AskBenefits agent

Step 1. Create a new AskBenefits agent

- Click the **Create new Agent** in the bottom left corner.

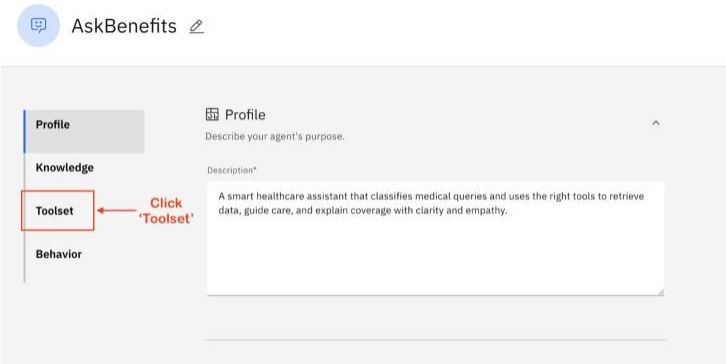


- Enter the Agent Name and description:
  - **Agent Name** – <Your_Name>_AskBenefits
  - **Agent Description** – A smart healthcare agent that classifies medical queries and uses the right tools to retrieve data, guide care, book appointments and explain coverage with clarity and empathy.
- Then click **Create** button.

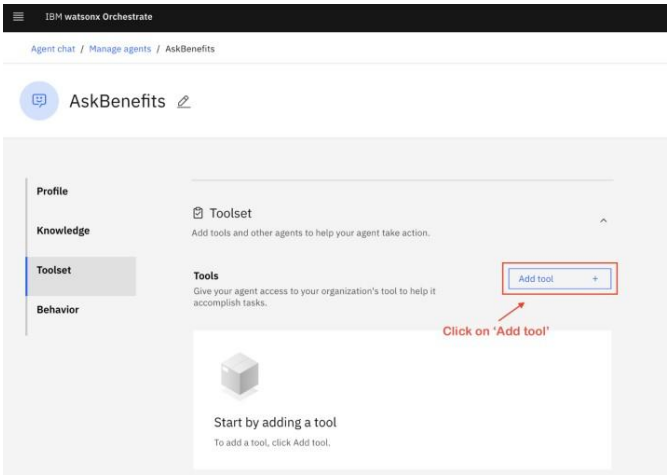


Step 2. Import and attach the tools

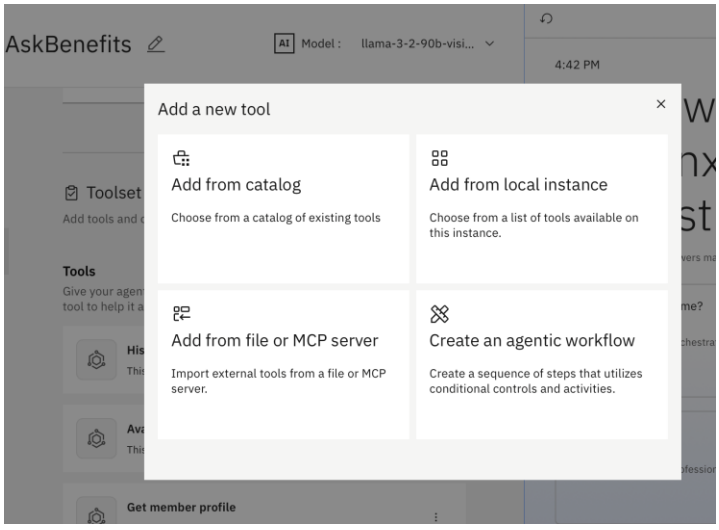
- Click the **Toolset** tab on the left-hand panel.



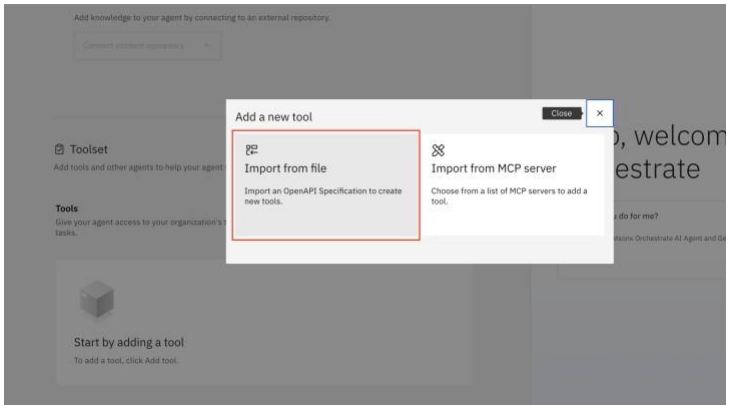
- Click **Add tool** button.



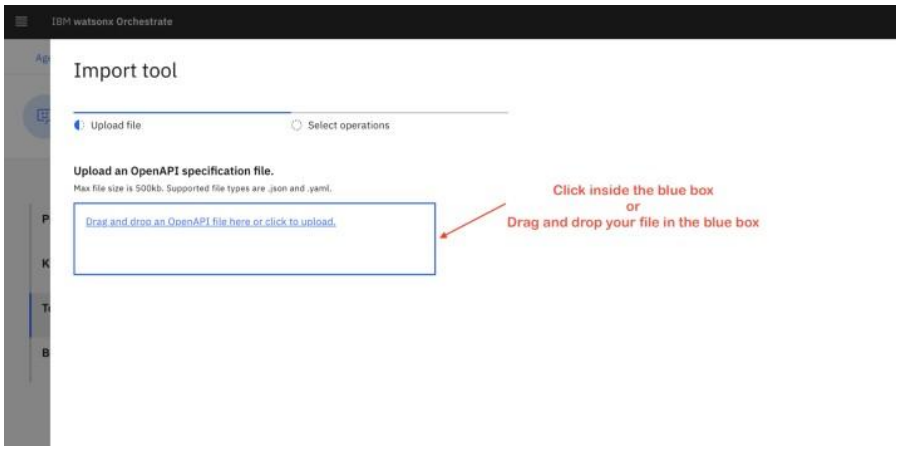
- Select the **Add from file or MCP server** option in the pop-up window.



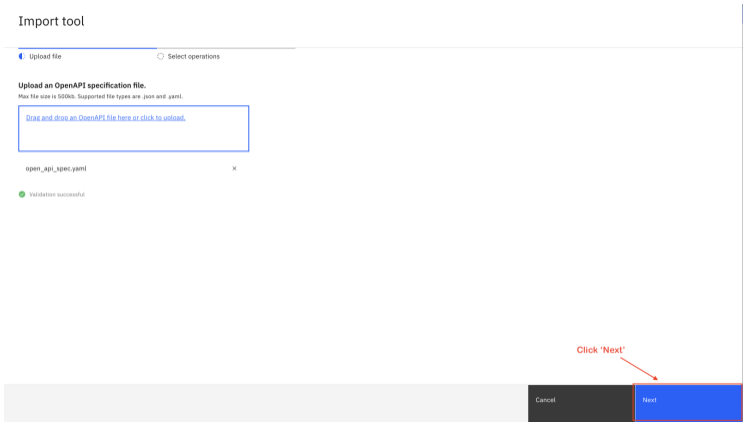
- Select '**Import from file**' option.



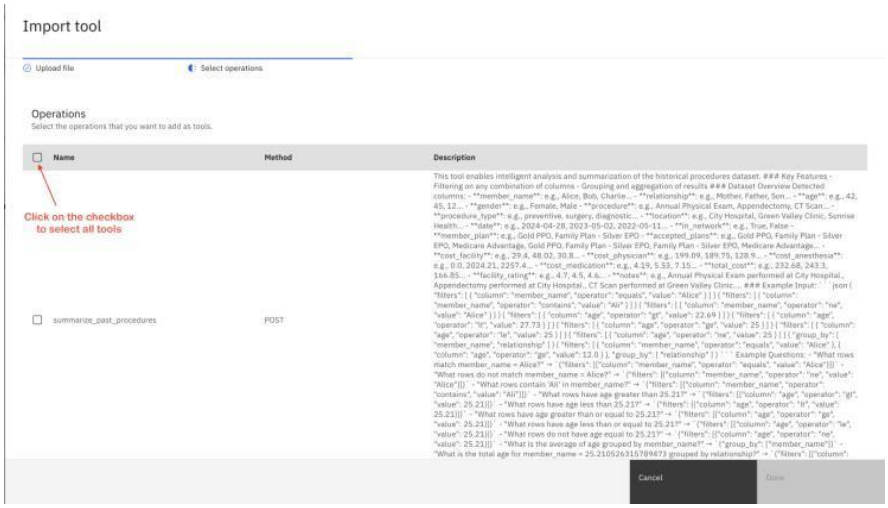
- Click inside the blue box to select the **ask_benefits_tools.yaml** file downloaded from the **What you'll need** section above.



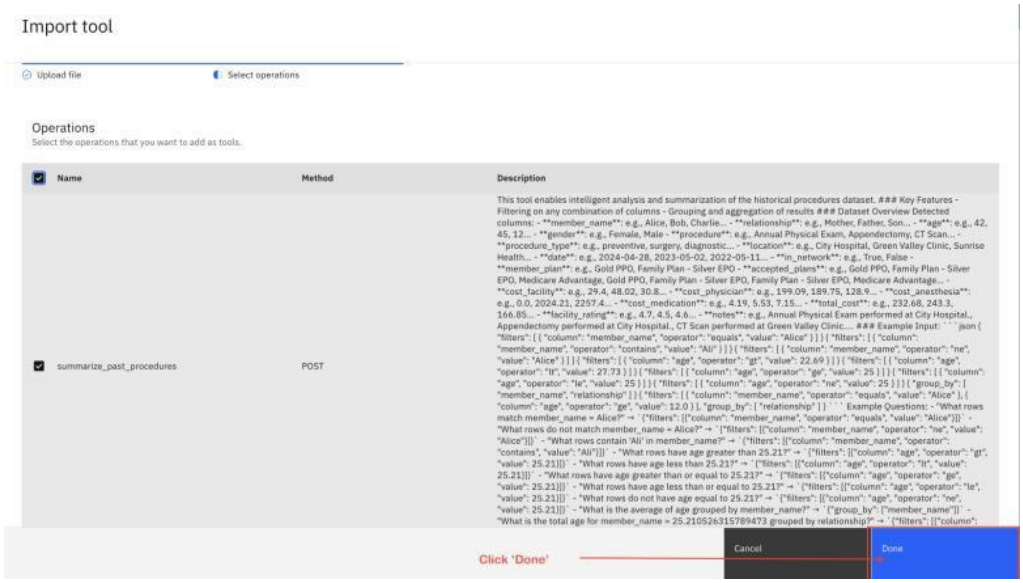
- Once the file is imported successfully, click **Next**.



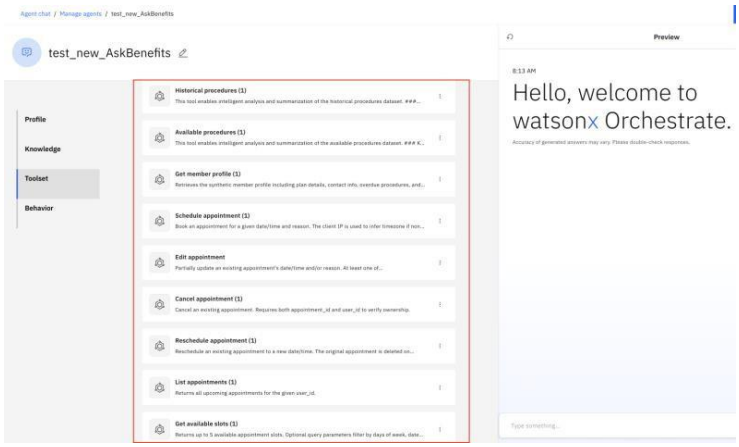
- Click **Select all rows in the table** checkbox to select all available tools.



- Then click on the blue **Done** button at the right bottom corner.

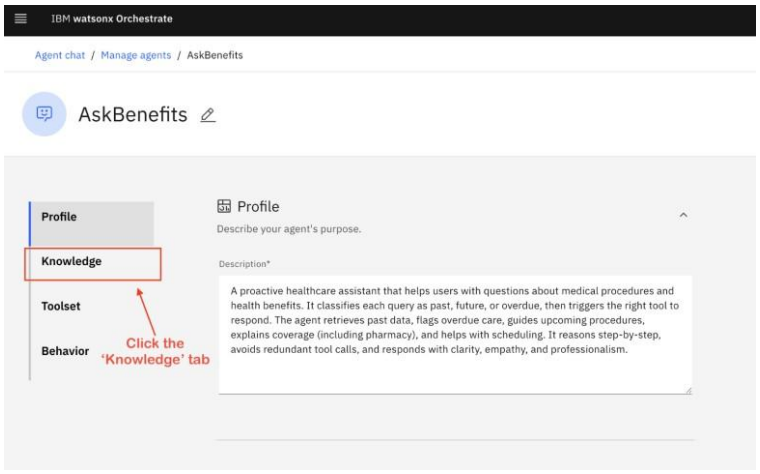


- All 9 tools will be added to your Agent. Sometimes it would take few seconds to reflect and show successfully adding the tools.

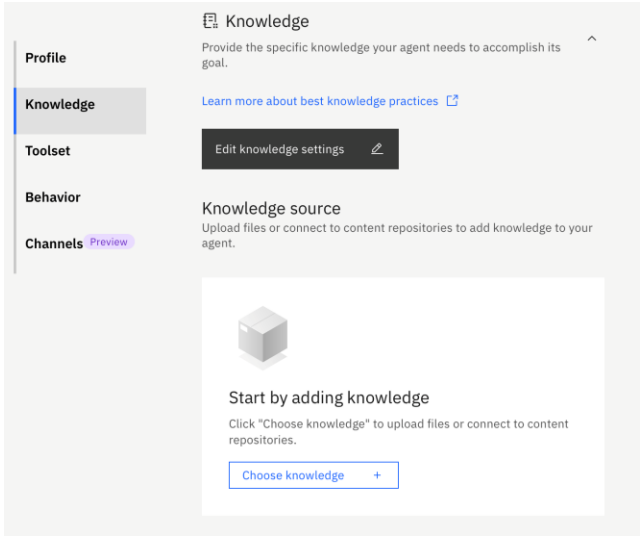


Step 3. Upload knowledge document

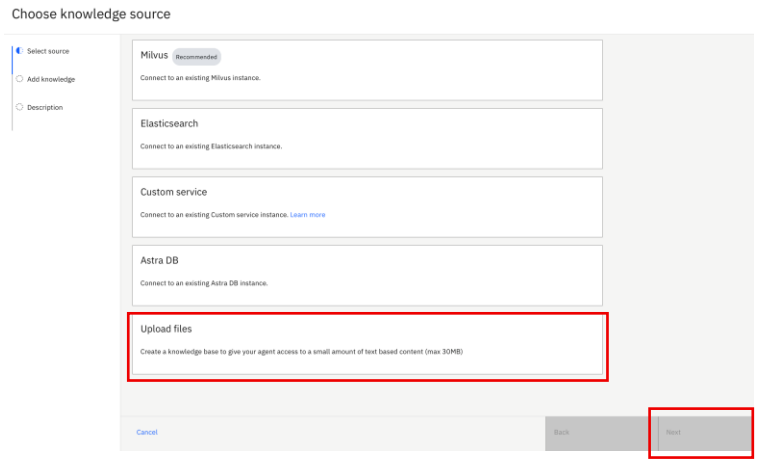
- Click the **Knowledge** tab in the left-hand panel.



- In the '**Knowledge**' section, click on **Choose Knowledge**.

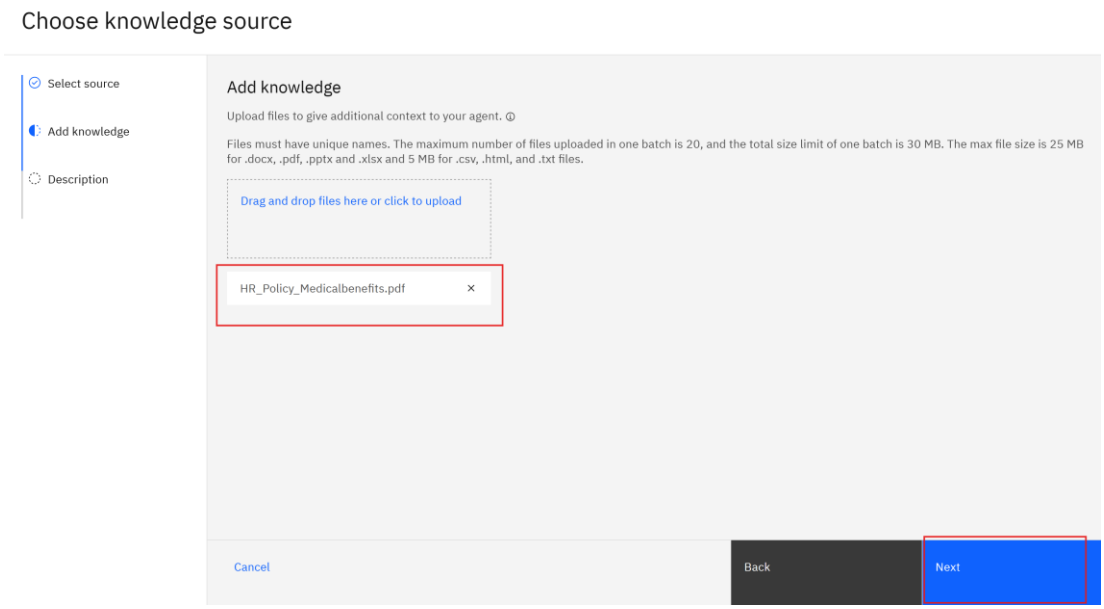


- Choose **Upload files** and then click **Next**.



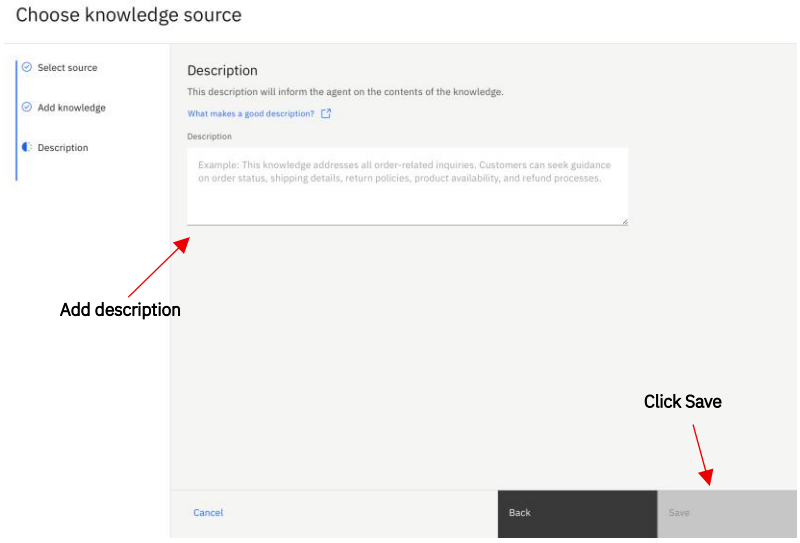
- Upload the **HR_Policy_Medicalbenefits.pdf** file downloaded from the '**What you'll need**' section and

select **Next**.



- As a best practice, in the '**Knowledge**' description field, add the below context to inform the agent on the contents of the knowledge and click **Save**.

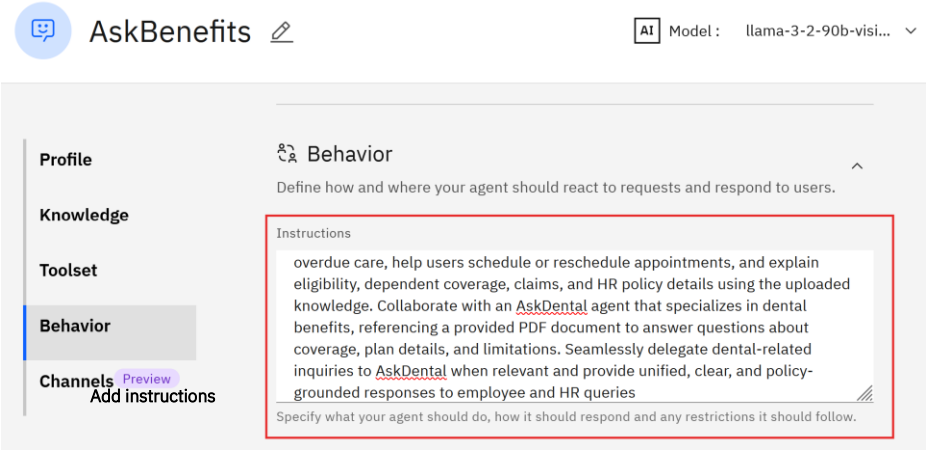
  "*This document covers key healthcare benefit policies—eligibility, coverage, claims, exclusions, and post-employment options. Use it to answer employee questions related to health benefits*"



- Select **Behavior** section on the left navigation pane and add the following instructions:

  "*Answer questions about medical procedures and health plan benefits. Detect whether a procedure is past, future, or overdue, retrieve relevant records, flag overdue care, help users schedule or reschedule appointments, and explain eligibility, dependent coverage, claims, and HR policy details using the uploaded knowledge. Collaborate with an AskDental agent that specializes in dental benefits, referencing a provided PDF document to answer questions about coverage, plan details, and limitations. Seamlessly delegate dental-related inquiries to AskDental when relevant and provide unified, clear, and policy-grounded responses to employee and HR queries.*"

Define behavioral rules and interaction logic to guide the agent's responses and actions to help it act consistently and effectively in various scenarios.



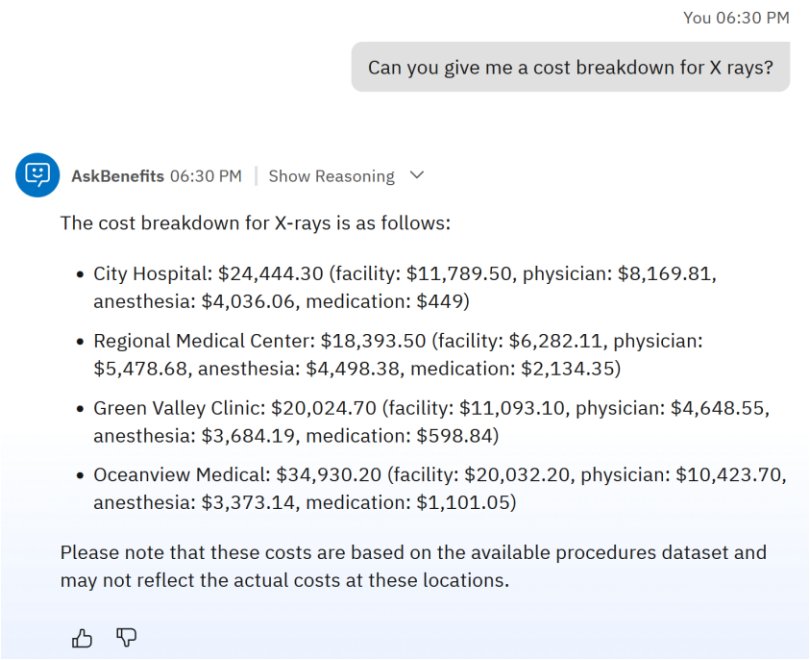
Step 4. Test the agent

In the **Preview** chat panel, enter the below prompts one-by-one in the "Type something..." field and press **Enter**.

- **Prompt 1**: Can you give me a cost breakdown for X rays?

  ***Expected outcome**:* The agent displays the X-ray cost breakdown across multiple hospitals, including facility, physician, anesthesia, and medication charges.

  What's happening behind the scenes? To understand that click **Show reasoning** and expand the steps.



  ***Key Observations**:* When you insert the prompt, the agent:

- Understands User Intent: The agent correctly interprets the request as requiring a filtered view of cost-related procedure data.

Dynamic Tool Invocation: It selects the correct tool '*Available_procedures'*, which is appropriate for analyzing procedure costs from a dataset.

- StructuredReasoningviaFilters:Itcreatesastructuredqueryusing:

```
"filters": "[{\"column\": \"procedure\", \"operator\": \"equals\", \"value\": \"X ray\"}]"
```

This shows the agent's ability to construct a parameterized request dynamically based on the user's natural language.

- **Prompt 2:** *Which among them is the cheapest?*

  Expected Outcome: The agent compares total costs and responds with the hospital offering the lowest price.

  Key Observation: This is a strong example of co-referencing, where the agent correctly interprets the user's use of "them" based on prior context. It also demonstrates the use of memory to retain and build on prior context—both key aspects of agentic behaviour.

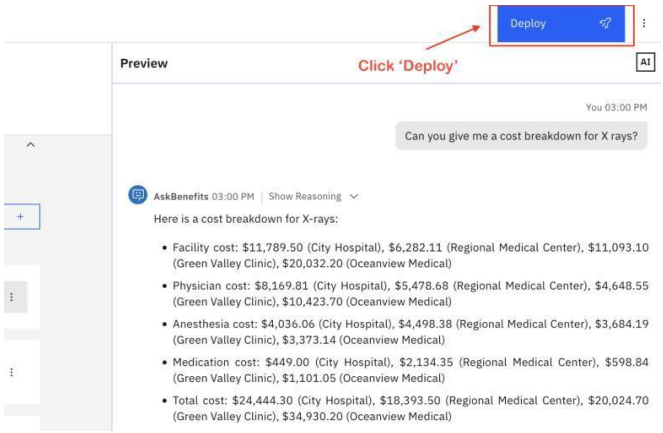
- **Prompt 3**: Am I eligible to add my spouse to my health insurance?

  Expected Outcome*:* The agent confirms that spouses are eligible dependents and advises the employee to provide supporting documents during enrolment.

  Key Observation*:* The agent intelligently processes the document, identifies the relevant section on dependent coverage, and retrieves a precise answer—demonstrating context-aware document parsing and retrieval.

Step 5. Testing the agent

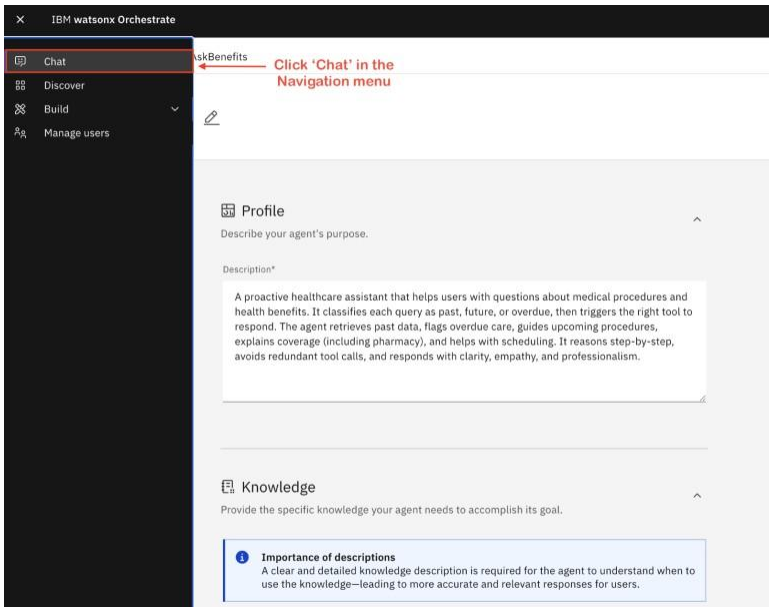
- Click **Deploy** to publish your AskBenefits agent.



- Congratulations! You successfully deployed the AskBenefits Agent.

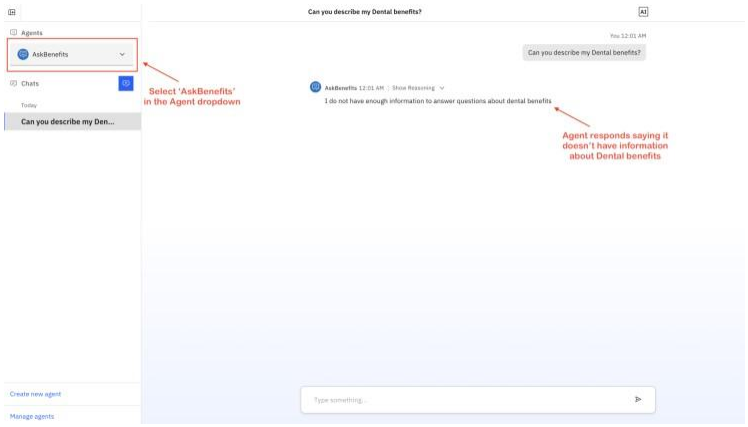
Step 6. Identify the knowledge gap

- Click **Chat** in the main Navigation menu.



- Select the **AskBenefits** agent in the Agent dropdown and enter the below prompt in the 'Type something…' field. Then press **Enter key**.

  Prompt: *Can you describe my Dental benefits?*



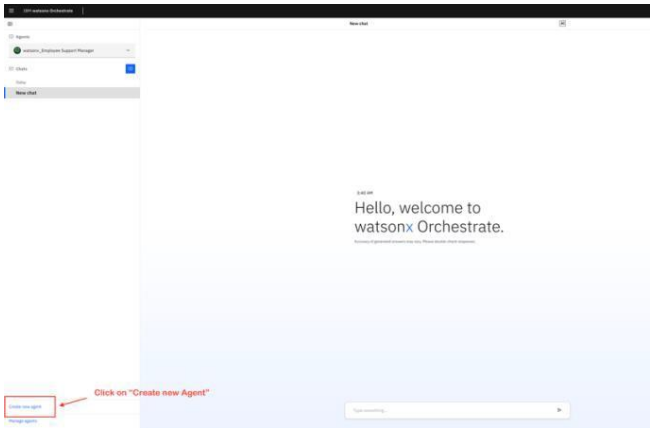
  Agent response: *I do not have enough information to answer questions about dental benefits.*

- Let's now look at how the above information can be retrieved from the AskBenefits agent. A collaborator agent is used by AskBenefits to retrieve answers related to Dental benefits.

Part 2: Create a new collaborator agent – AskDental

Step 1. Create a new AskDental agent

- Click on **Create new Agent** in bottom left corner.

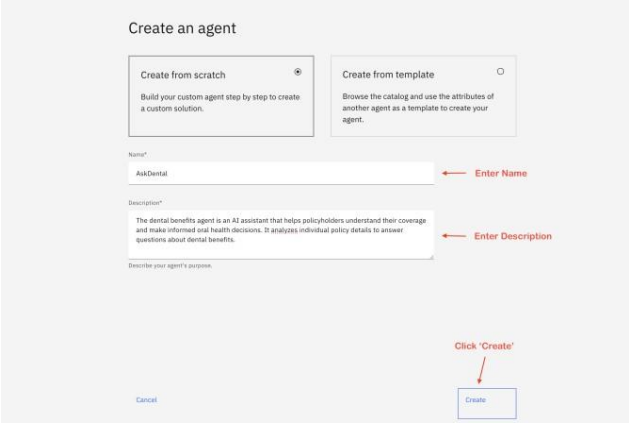


- Enter the Agent Name and description:

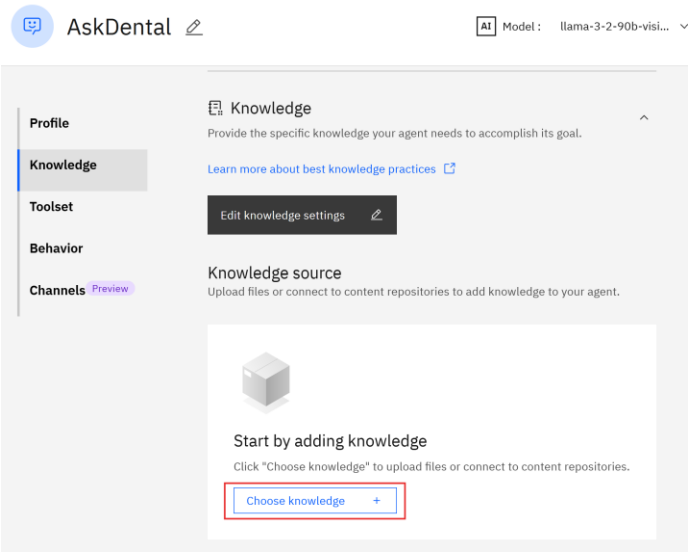
  Name – <Your_Name>_AskDental

  Agent Description – The dental benefits agent is an AI assistant that helps policyholders understand their coverage and make informed oral health decisions. It analyzes individual policy details to answer questions about dental benefits.

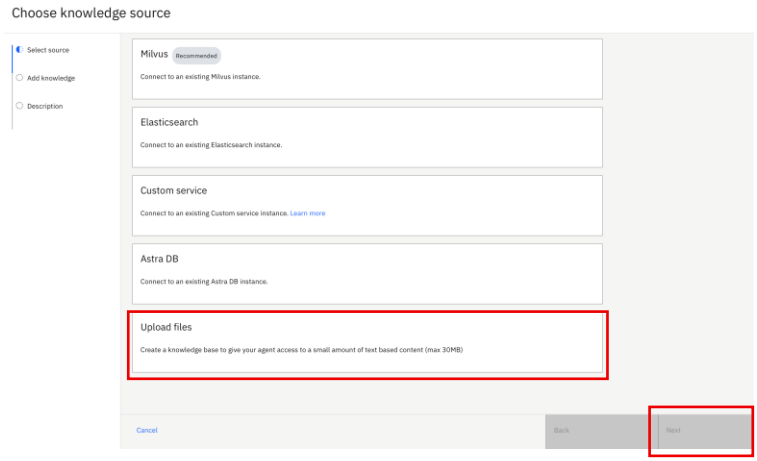
- Then click on '**Create**'.



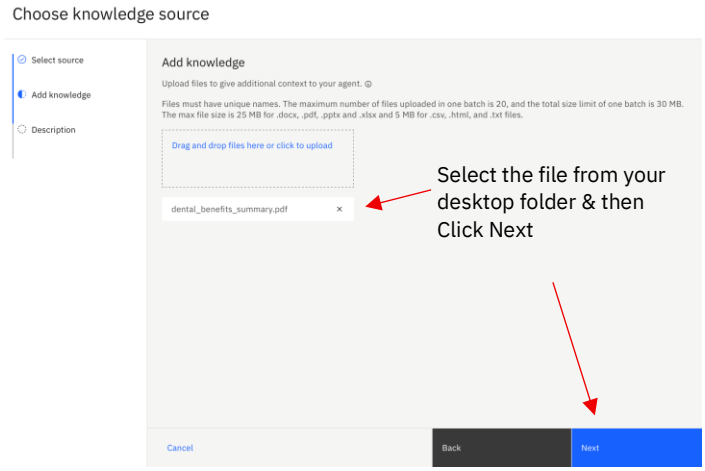
- In the **Knowledge Source** section under the '**Knowledge**' tab click **Choose Knowledge**.



- Select **Upload files** and click **Next**.

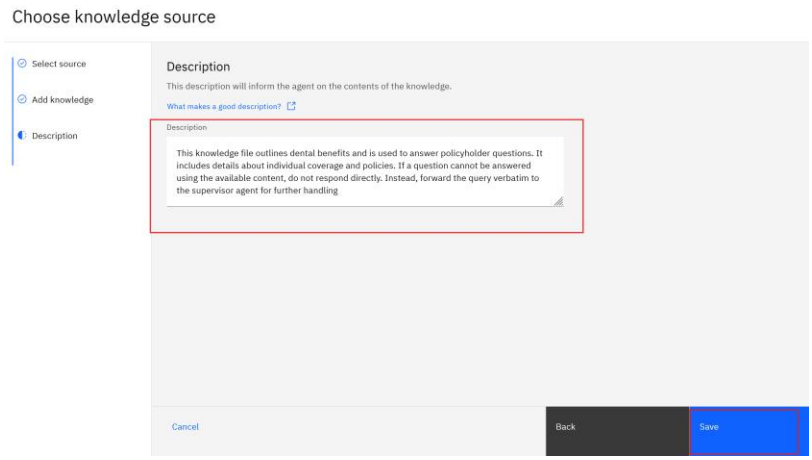


- Select and upload the file containing **Dental Benefit information *dental_benefits_summary.pdf*** you downloaded from What *you'll need* section.



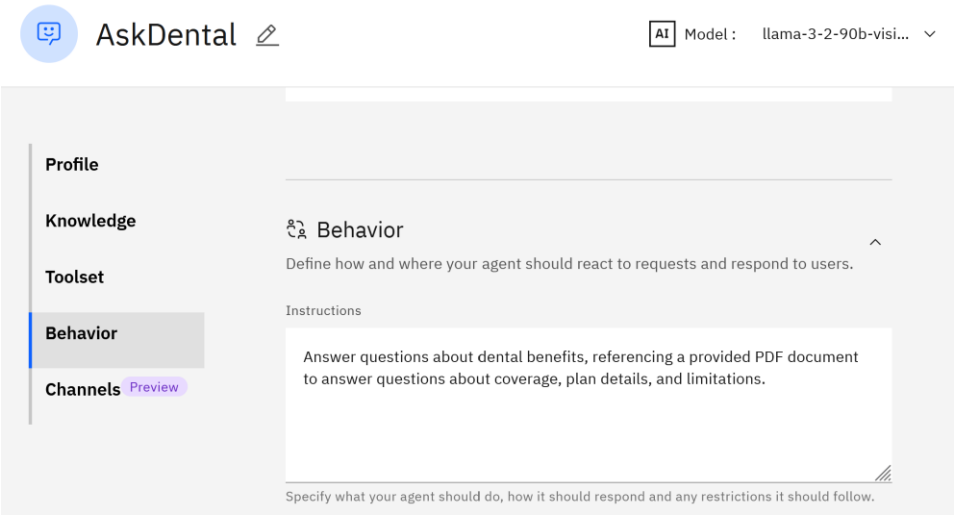
- In the Knowledge '**description**' field enter the following description:

  "This knowledge file outlines dental benefits and is used to answer policyholder questions. It includes details about individual coverage and policies. If a question cannot be answered using the available content, do not respond directly. Instead, forward the query verbatim to the supervisor agent for further handling."

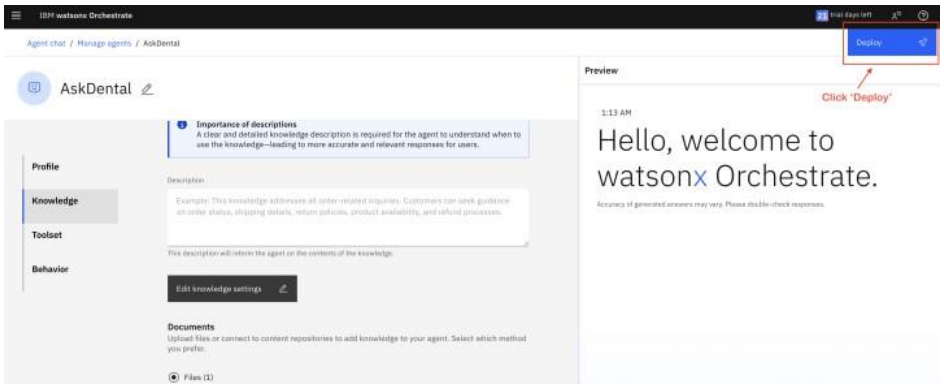


- Select **Behavior** section on the left navigation pane and add the following instructions:

  "*Answer questions about dental benefits, referencing a provided PDF document to answer questions about coverage, plan details, and limitations.*"

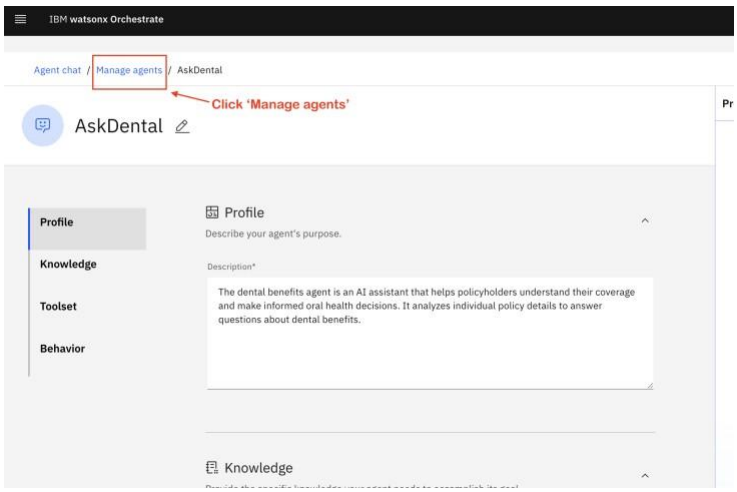


- Click **Deploy**.

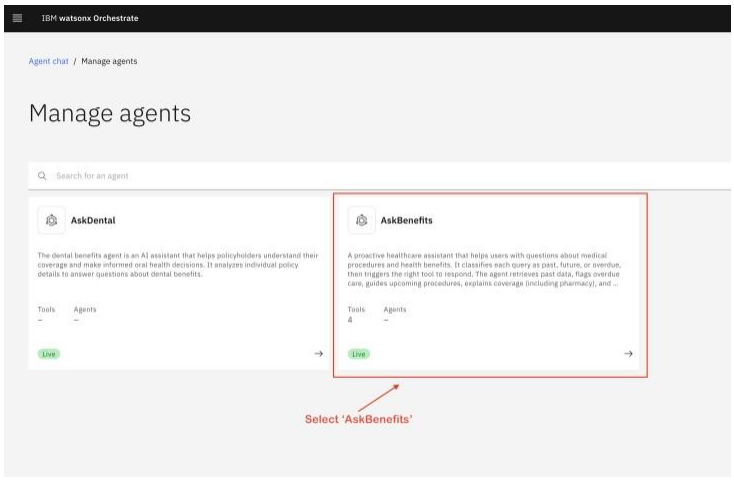


Step 2. Add AskDental agent as a collaborator

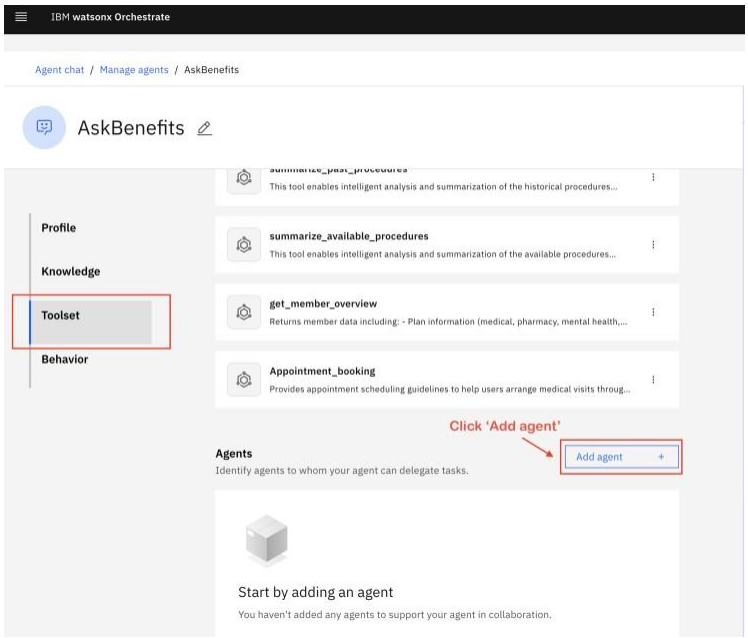
- Click **Manage agents**.



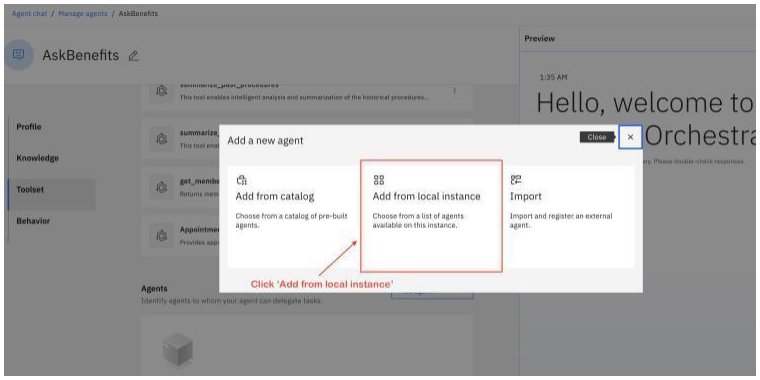
- Select **AskBenefits** agent.



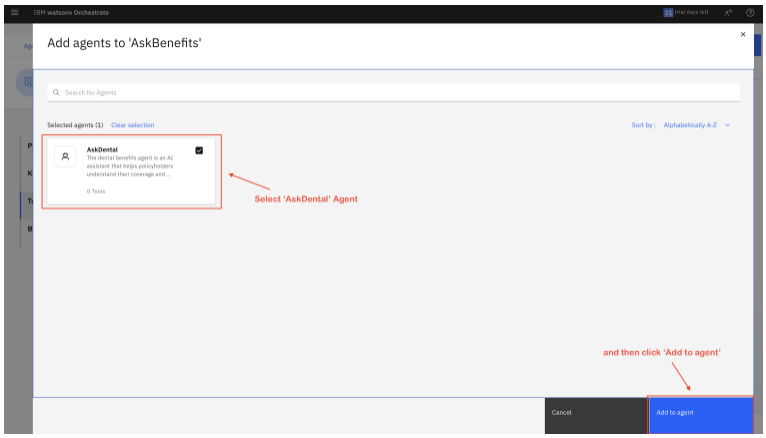
- Click **Add agent** under the '**Toolset**' section.



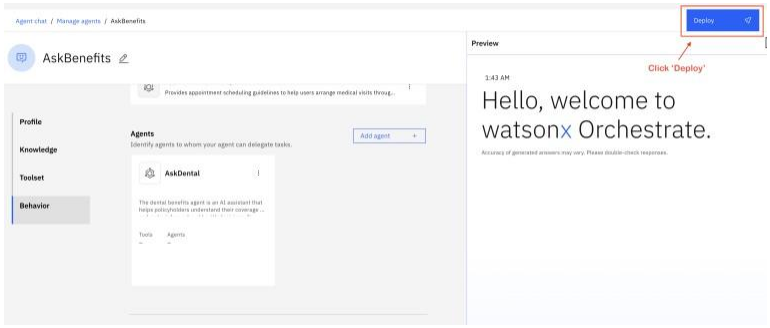
- Click **Add from local instance**.



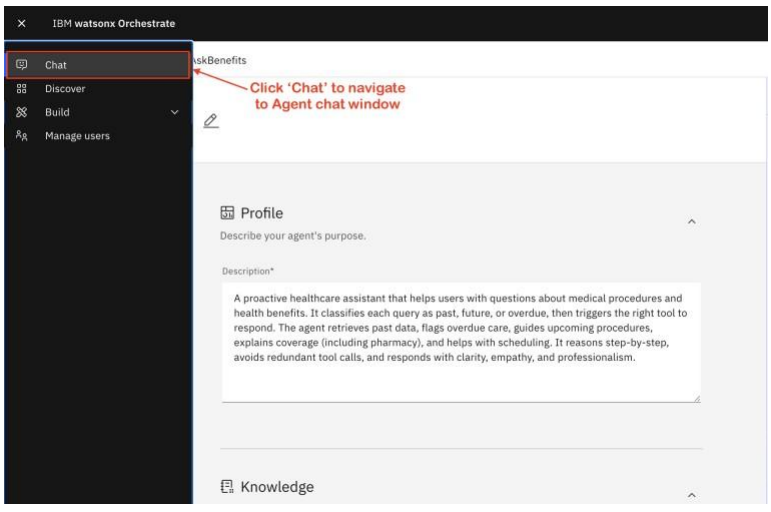
- Select **AskDental** agent and then click **Add to agent**.



- Click **Deploy**.



- Click **Chat** to navigate to the Agent chat screen.



Step 3. Testing scenarios

After successful deployment, select the **AskBenefits** agent in the Agent dropdown and test the same prompt to verify how the collaborator agent added to AskBenefits helps in answering Dental related queries.

- **Prompt**: *Can you describe my dental benefits?*

  Expected Outcome*:* The AskBenefits agent routes the query to the AskDental agent, which responds with a detailed summary of dental benefits including deductible, coverage, and plan-specific details—retrieved from the uploaded dental policy document.

  Key Observation*:* The AskBenefits agent correctly routes the dental benefits query to the AskDental agent, which retrieves a detailed response from its underlying document—demonstrating effective agent collaboration and document-grounded reasoning.

  Try some more prompts:

  1. What procedures have we had at City Hospital in the last year?
  2. Show some appointment slots available for Dental checkup?
  3. Can you show the available slots on weekends instead?
  4. Book an appointment for <Enter an available slot>
  5. List my upcoming appointments
  6. Give me a summary of my insurance plan