Learn how to use Red Hat Ansible Lightspeed with IBM watsonx Code Assistant.
Learn how to use Red Hat Ansible Lightspeed with IBM watsonx Code Assistant.
Abstract

This guide shows you how to use Red Hat Ansible Lightspeed with IBM watsonx Code Assistant.
# Table of Contents

MAKING OPEN SOURCE MORE INCLUSIVE ................................................................. 4

CHAPTER 1. INTRODUCTION TO RED HAT ANSIBLE LIGHTSPEED .......................... 5
  1.1. ABOUT RED HAT ANSIBLE LIGHTSPEED ................................................. 5
     1.1.1. Benefits of using Red Hat Ansible Lightspeed ................................. 5
  1.2. KEY FEATURES OF RED HAT ANSIBLE LIGHTSPEED ................................ 6
  1.3. USING RED HAT ANSIBLE LIGHTSPEED WITH IBM WATSONX CODE ASSISTANT
     1.3.1. Process for using Red Hat Ansible Lightspeed .................................. 6
  1.4. DATA GATHERED TO TRAIN THE IBM WATSONX CODE ASSISTANT MODELS
     1.4.1. Models .................................................................................. 7
     1.4.2. Data sources ........................................................................ 7
     1.4.3. Data telemetry .................................................................... 7

CHAPTER 2. LOGGING INTO THE ANSIBLE LIGHTSPEED ADMINISTRATOR PORTAL .... 9
  2.1. LOGGING IN TO THE ANSIBLE LIGHTSPEED ADMINISTRATOR PORTAL .......... 9

CHAPTER 3. CONFIGURING RED HAT ANSIBLE LIGHTSPEED TO CONNECT WITH IBM WATSONX CODE ASSISTANT ................................................................. 10
  3.1. ABOUT THE WATSONX CODE ASSISTANT KEY AND MODEL ID ................. 10
  3.2. CONNECTING RED HAT ANSIBLE LIGHTSPEED TO IBM WATSONX CODE ASSISTANT
     3.2.1. Additional resources .................................................................. 11

CHAPTER 4. ASSIGNING AND REMOVING SEAT LICENSES ...................................... 12
  4.1. ASSIGNING SEAT LICENSES ................................................................. 12
  4.2. REMOVING SEAT LICENSES ............................................................... 12

CHAPTER 5. LOGGING IN AND OUT OF THE ANSIBLE LIGHTSPEED PORTAL .......... 14
  5.1. LOGGING IN TO THE ANSIBLE LIGHTSPEED PORTAL ............................... 14
  5.2. LOGGING OUT OF THE ANSIBLE LIGHTSPEED PORTAL ............................. 14

CHAPTER 6. INSTALLING AND CONFIGURING THE ANSIBLE VS CODE EXTENSION .... 16
  6.1. INSTALLING THE ANSIBLE VS CODE EXTENSION ..................................... 16
  6.2. CONFIGURING THE ANSIBLE VS CODE EXTENSION ................................. 16

CHAPTER 7. REQUESTING TASK RECOMMENDATIONS ............................................ 19
  7.1. OVERVIEW .................................................................................. 19
     7.1.1. Best practices to improve the recommended guidance ......................... 19
  7.2. REQUESTING CODE RECOMMENDATIONS FOR A SINGLE TASK ................ 20
  7.3. REQUESTING CODE RECOMMENDATIONS FOR MULTIPLE TASKS ............. 24
  7.4. VIEWING ANSIBLE LIGHTSPEED TRAINING MATCHES ............................. 29
  7.5. PROVIDING FEEDBACK ON THE ANSIBLE LIGHTSPEED SERVICE .............. 30

CHAPTER 8. INSTALLING AND CONFIGURING ANSIBLE CODE BOT ......................... 33
  8.1. INSTALLING ANSIBLE CODE BOT ..................................................... 33
  8.2. CONFIGURING ANSIBLE CODE BOT TO SCAN YOUR REPOSITORY AT REGULAR INTERVALS 35
  8.3. MANUALLY SCANNING YOUR GIT REPOSITORIES .................................. 35
  8.4. HOW ANSIBLE CODE BOT HANDLES DUPLICATE PULL REQUESTS ............ 36

CHAPTER 9. TROUBLESHOOTING ........................................................................ 38
  9.1. TROUBLESHOOTING RED HAT ANSIBLE LIGHTSPEED CONFIGURATION ERRORS
     9.1.1. Cannot access the Ansible Lightspeed administrator portal .................. 38
     9.1.2. Cannot upload the WCA key .......................................................... 38
     9.1.3. Cannot configure the model ID due to authentication failure .................. 38
     9.1.4. Cannot configure the model ID due to inference failure ....................... 38
9.2. TROUBLESHOOTING ANSIBLE VISUAL STUDIO CODE EXTENSION ERRORS

9.2.1. Cannot view the generated code recommendations using the Ansible VS Code extension

9.3. TROUBLESHOOTING ANSIBLE CODE BOT ERRORS

9.3.1. Cannot access Ansible code bot

9.3.2. Cannot scan your Git repository using Ansible code bot
MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright’s message.
Learn about Red Hat Ansible Lightspeed with IBM watsonx Code Assistant, its benefits, key features, process, and data gathered to train the IBM watsonx Code Assistant models.

1.1. ABOUT RED HAT ANSIBLE LIGHTSPEED

Red Hat Ansible Lightspeed with IBM watsonx Code Assistant is a generative AI service that helps automation teams create, adopt, and maintain Ansible content more efficiently.

Red Hat Ansible Lightspeed uses natural language prompts to generate code recommendations for automation tasks based on Ansible best practices. Red Hat Ansible Lightspeed also proactively creates pull requests with code improvement recommendations for you to review, test, and apply in your Ansible content.

Red Hat Ansible Lightspeed uses IBM watsonx Code Assistant models trained on subject matter expertise across the Ansible ecosystem, which includes Ansible Galaxy, GitHub, and Ansible Certified and Validated content.

Red Hat Ansible Lightspeed is integrated with your existing Ansible developer workflows. You can use your existing Git repositories (both public and private) to train your single-tenant IBM watsonx Code Assistant model. You can access Red Hat Ansible Lightspeed content suggestions in Visual Studio Code (VS Code) by using the Ansible VS Code extension.

1.1.1. Benefits of using Red Hat Ansible Lightspeed

Red Hat Ansible Lightspeed with IBM watsonx Code Assistant offers the following benefits:

- **Reduces the onboarding learning period for Ansible developers**
  With just a basic understanding of YAML syntax, Ansible developers can use natural language prompts to describe the automation goal. Red Hat Ansible Lightspeed then offers Ansible code recommendations to help achieve the automation goal more efficiently. This combination of content and best practice suggestions reduces the learning curve and offers a smoother onboarding experience for new Ansible users.

  For example, to get a multitask code recommendation, you can enter the prompt `Install postgresql-server & run postgresql-setup command`. The Ansible Lightspeed service reads the text, interacts with IBM watsonx Code Assistant, and generates code recommendations to automate a multitask that installs a PostgreSQL server and sets up a PostgreSQL database. You can then view and accept the code recommendations to create tasks in an Ansible YAML file.

- **Increases productivity with quality content creation**
  Red Hat Ansible Lightspeed offers automation code recommendations that adhere to Ansible best practices, and IBM watsonx Code Assistant provides model fine-tuning features to improve the accuracy of suggested content based on your organization’s existing Ansible content. Therefore, the AI-generated code recommendations are more accurate, more reliable, and integrated with your existing automation development workflows.

- **Extends trust with AI-generated code recommendations**
  The AI-generated code recommendations enable you to extend trust, with an automation code base that adheres to accepted Ansible best practices and significant data safeguards.
1.2. KEY FEATURES OF RED HAT ANSIBLE LIGHTSPEED

Red Hat Ansible Lightspeed offers the following key features:

- **Ansible-specific IBM watsonx Code Assistant models**
  Red Hat Ansible Lightspeed with IBM watsonx Code Assistant uses Ansible-specific, single-tenant IBM watsonx Granite models unique to your organization, which are provided, managed, and maintained by IBM.

- **Single tasks and multitask generation**
  Using natural language prompts, you can generate single task or multiple task recommendations for Ansible task files and playbooks. To request multitask code recommendations, you can enter a sequence of natural language task prompts in a YAML file comment separated by ampersand (&) symbols.

- **Content source matching**
  For each generated code recommendation, Red Hat Ansible Lightspeed lists content source matches, including details such as potential source, content author, and relevant licenses. You can use this data to gain insight into potential training data sources used to generate the code recommendations.

- **Post-processing capabilities**
  Red Hat Ansible Lightspeed offers post-processing capabilities that augment IBM watsonx Code Assistant and improve the quality and accuracy of code recommendations.

- **Content modernization**
  The Ansible code bot scans existing content collections, roles, and playbooks through Git repositories, and proactively creates pull requests whenever best practices or quality improvement recommendations are available. The bot automatically submits pull requests to the repository, which proactively alerts the repository owner to a recommended change to their content.

1.3. USING RED HAT ANSIBLE LIGHTSPEED WITH IBM WATSONX CODE ASSISTANT

**Prerequisites**

To use Red Hat Ansible Lightspeed with IBM watsonx Code Assistant, ensure that you have the following components:

- Ansible Automation Platform subscription
- VS Code version 1.70.1 or later
- Ansible extension for VS Code version 2.8 or later
- IBM watsonx Code Assistant service

1.3.1. Process for using Red Hat Ansible Lightspeed

- Red Hat customer Portal organization administrators can use this guide to:
  - Connect Red Hat Ansible Lightspeed to IBM watsonx Code Assistant instance
  - Configure the IBM watsonx Code Assistant model that you want to use
Assign and remove Red Hat Ansible Lightspeed seat licenses

- Ansible users (content developers and automators) can use this guide to:
  - Install and configure Ansible Visual Studio Code extension
  - Request task recommendations
  - Provide feedback on the Ansible Lightspeed service

- Ansible repository administrators can use this guide to:
  - Install and configure Ansible code bot
  - Configure Ansible code bot to scan your repositories at regular intervals
  - Manually scan your git repositories

1.4. DATA GATHERED TO TRAIN THE IBM WATSONX CODE ASSISTANT MODELS

1.4.1. Models

Red Hat Ansible Lightspeed with IBM watsonx Code Assistant uses single-tenant Ansible-specific IBM watsonx Granite models unique to your organization. These models are provided, managed, and maintained by IBM.

You can use the Ansible-specific IBM watsonx Granite model as the default model for Red Hat Ansible Lightspeed, so that all Ansible Lightspeed users in your organization can use the fine-tuned model.

1.4.2. Data sources

IBM watsonx Code Assistant models are trained on Ansible content from Ansible Galaxy, data from public Git repositories, and Red Hat Ansible subject matter expert examples.

If you publish content to Ansible Galaxy and want to restrict your Ansible Galaxy content from being used to train the models, you can opt out of sharing your Ansible Galaxy data in the Ansible Galaxy namespace configuration.

1.4.3. Data telemetry

The Ansible VS Code extension, with Red Hat Ansible Lightspeed with IBM watsonx Code Assistant features enabled, automatically collects recommendations, usage telemetry, and playbook state through automated events. This data is eventually shared with Red Hat Ansible Lightspeed and IBM watsonx Code Assistant to generate code recommendations.

The following table lists the data that the Ansible VS Code extension collects.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommendation accepted</td>
<td>An indicator that a code recommendation was accepted by an Ansible user and included in the Ansible playbook.</td>
</tr>
<tr>
<td>Metric</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Recommendation rejected</td>
<td>An indicator that a code recommendation was rejected by an Ansible user and excluded from the Ansible playbook.</td>
</tr>
<tr>
<td>Anonymized document URI</td>
<td>The string that helps to determine the document type (playbook, role, or task).</td>
</tr>
<tr>
<td>Request and response timestamps</td>
<td>The request and response times captured in UTC format.</td>
</tr>
<tr>
<td>Anonymized user ID</td>
<td>A random anonymous user ID that is stored locally on ~/.redhat/anonymoid.</td>
</tr>
<tr>
<td>Anonymized suggestion ID</td>
<td>A random auto-generated session ID associated with the suggestion request.</td>
</tr>
<tr>
<td>IBM watsonx Code Assistant metadata</td>
<td>Contextual information about the suggestion returned from the IBM watsonx Code Assistant model such as the recommended Ansible module, recommended parameter keys, and length of the suggestion.</td>
</tr>
</tbody>
</table>
CHAPTER 2. LOGGING INTO THE ANSIBLE LIGHTSPEED ADMINISTRATOR PORTAL

As a Red Hat account organization administrator, you can use the Ansible Lightspeed administrator portal to configure settings to connect Red Hat Ansible Lightspeed to IBM watsonx Code Assistant.

2.1. LOGGING IN TO THE ANSIBLE LIGHTSPEED ADMINISTRATOR PORTAL

Prerequisites

- You have organization administrator privileges to a Red Hat Customer Portal organization with a valid Red Hat Ansible Automation Platform subscription.

Procedure

1. Log in to the Ansible Lightspeed portal as an organization administrator.

2. Click Log in → Log in with Red Hat

   As part of the authentication process, the Ansible Lightspeed Service checks whether you are part of an organization that has a valid Ansible Lightspeed subscription, and whether your organization has a valid Ansible Automation Platform subscription. On successful authentication, the login screen is displayed along with your username and your assigned user role.

4. From the login screen, click Admin Portal.
   You are redirected to the Red Hat Ansible Lightspeed with IBM watsonx Code Assistant administrator portal where you can connect Red Hat Ansible Lightspeed to your IBM watsonx Code Assistant instance.
CHAPTER 3. CONFIGURING RED HAT ANSIBLE LIGHTSPEED TO CONNECT WITH IBM WATSONX CODE ASSISTANT

As a Red Hat customer portal administrator, you must configure Red Hat Ansible Lightspeed to connect to your IBM Watsonx Code Assistant instance.

3.1. ABOUT THE WATSONX CODE ASSISTANT KEY AND MODEL ID

You need the following IBM Watsonx Code Assistant information to connect Red Hat Ansible Lightspeed to your IBM Watsonx Code Assistant:

- **Watsonx Code Assistant (WCA) API key**
  A WCA API key authenticates all requests made from Red Hat Ansible Lightspeed to IBM Watsonx Code Assistant. Each Red Hat organization with a valid Ansible Automation Platform subscription must have a configured WCA API key. When an authenticated RH-SSO user creates a task request in Red Hat Ansible Lightspeed, the WCA API key associated with the user’s Red Hat organization is used to authenticate the request to IBM Watsonx Code Assistant.

- **Model ID**
  A unique WCA model ID identifies an IBM Watsonx Code Assistant model in your IBM Cloud account. The model ID that you configure in the Ansible Lightspeed administrator portal is used as the default model, and can be accessed by all Ansible Lightspeed users within your organization.

**IMPORTANT**
You must configure both the WCA key and the model ID when you are initially configuring Red Hat Ansible Lightspeed.

3.2. CONNECTING RED HAT ANSIBLE LIGHTSPEED TO IBM WATSONX CODE ASSISTANT

**Prerequisites**

- You have obtained a WCA API key and a model ID from the IBM Watsonx Code Assistant that you want to use in Red Hat Ansible Lightspeed.
  For information about how to obtain a WCA API key and model ID from IBM Watsonx Code Assistant, see the IBM Watsonx Code Assistant documentation.

**Procedure**

1. Log in to the Ansible Lightspeed with IBM Watsonx Code Assistant Hybrid Cloud Console as an organization administrator.

2. Specify the WCA key of your IBM Watsonx Code Assistant instance:
   a. Under **IBM Cloud API Key**, click **Add API key**. A screen to enter the **API Key** is displayed.
   b. Enter the API Key.
   c. Optional: Click **Test** to validate the WCA key.
   d. Click **Save**.
3. Specify the model ID of the model that you want to use:
   a. Click **Model Settings**.
   b. Under **Model ID**, click **Add Model ID**. A screen to enter the **Model ID** is displayed.
   c. Enter the **Model ID** that you obtained in the previous procedure as the default model for your organization.
   d. Optional: Click **Test model ID** to validate the model ID.
   e. Click **Save**.
      When the WCA API key and model ID is successfully validated, Red Hat Ansible Lightspeed is connected to your IBM watsonx Code Assistant instance.

### 3.2.1. Additional resources

- [Troubleshooting Red Hat Ansible Lightspeed configuration errors](#)
CHAPTER 4. ASSIGNING AND REMOVING SEAT LICENSES

After connecting Red Hat Ansible Lightspeed to an IBM watsonx Code Assistant instance, Red Hat Customer Portal organization administrators can assign seat licenses to users in their organization. The seat assignment feature enables select Ansible users in your organization to access and use Red Hat Ansible Lightspeed.

Your Red Hat Ansible Automation Platform subscription includes a specific number of Ansible Lightspeed named seats depending on the number of Ansible Automation Platform subscriptions your organization has purchased. Red Hat Ansible Lightspeed authenticates users by using Red Hat Single Sign On (SSO) and verifies that they have an assigned seat license.

4.1. ASSIGNING SEAT LICENSES

You can grant Ansible Lightspeed seat licenses to select users in your organization.

Prerequisites

- You must be assigned as an organization administrator in the Red Hat Customer Portal.
- You have configured Red Hat Ansible Lightspeed to connect to IBM watsonx Code Assistant instance.

Procedure

1. Go to Ansible Automation Platform on the Red Hat Hybrid Cloud Console.
2. From the navigation panel, select Ansible Lightspeed with IBM watsonx Code Assistant→ Seat Management.
3. Click Assign users.
4. Specify the usernames of users that you want to assign a seat license to.
5. Click Assign, and on the verification message, click Ok to confirm.

4.2. REMOVING SEAT LICENSES

You can remove users who no longer need to access Red Hat Ansible Lightspeed, which makes licenses available for others to use.

NOTE

This task removes the Ansible user’s Red Hat Ansible Lightspeed seat license only; it does not remove the user from your Red Hat Customer Portal organization.

Procedure

1. Go to Ansible Automation Platform on the Red Hat Hybrid Cloud Console.
2. From the navigation panel, select Ansible Lightspeed with IBM watsonx Code Assistant→ Seat Management.
3. Click Remove users.
4. Select the username of the user whose license you want to remove.

5. Click **Remove**, and on the verification message, click **Ok** to confirm.
CHAPTER 5. LOGGING IN AND OUT OF THE ANSIBLE LIGHTSPEED PORTAL

Ansible users can use Red Hat Ansible Lightspeed after the organization administrator has assigned them seat licenses. You access Ansible Lightspeed through the Ansible Lightspeed portal. Red Hat Ansible Lightspeed uses Red Hat Single Sign-On (RH-SSO) for authentication, and verifies that you have an assigned seat license. After you log in to the Ansible Lightspeed portal, your assigned user role is displayed on the login screen.

Table 5.1. User login scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>You are a RH-SSO user with an assigned seat license. Note: This is the typical scenario for accessing as Red Hat Ansible Lightspeed an Ansible user.</td>
<td>You are routed to the Red Hat Ansible Lightspeed paid commercial offering.</td>
</tr>
<tr>
<td>You are a RH-SSO user with an assigned seat license, but your organization administrator has not configured Red Hat Ansible Lightspeed to connect with IBM watsonx Code Assistant.</td>
<td>You are routed to the Red Hat Ansible Lightspeed paid commercial offering with a message that your organization administrator has not configured a model for your organization.</td>
</tr>
<tr>
<td>You are a RH-SSO user without an assigned seat license.</td>
<td>You are identified as a community user and redirected to Red Hat Ansible Lightspeed Technical Preview, which is a free community model.</td>
</tr>
<tr>
<td>Your organization has an Ansible Lightspeed subscription but you do not have an assigned seat license.</td>
<td>You are identified as a community user and redirected to Red Hat Ansible Lightspeed Technical Preview, which is a free community model.</td>
</tr>
</tbody>
</table>

5.1. LOGGING IN TO THE ANSIBLE LIGHTSPEED PORTAL

Procedure

1. Go to the Ansible Lightspeed portal login page.
2. Click Log in → Log in with Red Hat
3. Enter your Red Hat account username and password.

Red Hat Ansible Lightspeed uses Red Hat Single Sign-On (RH-SSO) to authenticate you, and verifies that you have an assigned seat license. If you have an assigned seat license, you can access Ansible Lightspeed and the login portal displays your assigned role.

5.2. LOGGING OUT OF THE ANSIBLE LIGHTSPEED PORTAL

To log out of the Ansible Lightspeed Service, you must log out of both the Ansible Lightspeed VS Code extension and the Ansible Lightspeed portal.

Procedure
Log out of the Ansible Lightspeed VS Code extension:

- Click the Person icon. You will see a list of accounts that VS Code is logged into.
- Select Ansible Lightspeed → Sign Out.

Log out of the Ansible Lightspeed portal:

- Navigate to the Ansible Lightspeed portal login page.
- Click Log out.
CHAPTER 6. INSTALLING AND CONFIGURING THE ANSIBLE VS CODE EXTENSION

Red Hat Ansible Lightspeed with IBM watsonx Code Assistant is integrated with the Ansible Visual Studio (VS) Code extension in VS Code. The Ansible VS Code extension, with Red Hat Ansible Lightspeed with IBM watsonx Code Assistant features enabled, automatically collects recommendations, usage telemetry, and Ansible YAML file state through automated events.

To access Red Hat Ansible Lightspeed, all Ansible users must install and configure the Ansible VS Code extension in their VS Code. The Ansible VS Code extension uses the Ansible-specific IBM watsonx Granite model configured in the Ansible Lightspeed administrator portal as the default mode for all users that have been assigned an Ansible Lightspeed seat. However, Ansible users can use the model-override setting in the Ansible VS Code extension settings to specify a different model ID.

6.1. INSTALLING THE ANSIBLE VS CODE EXTENSION

Prerequisites
- VS Code version 1.70.1 or later.

NOTE
You can also install VScode derivatives, such as VScode Insider or VS Codium.

Procedure
1. Open the VS Code application.
2. From the navigation menu, click the Extensions icon.
3. In the Search field, enter Ansible.
4. Select Ansible to choose the Ansible language support extension published by Red Hat.
5. Click Install.
6. After installation is complete, verify your VSCode installation:
   a. Create a new YAML file using the .yml or .yaml file extension.
   b. From the Status toolbar, click the language indicator and select Ansible to associate the Ansible language type with the new YAML file.
   c. Start writing a test playbook. Contextual aids are displayed as you start creating your content.

6.2. CONFIGURING THE ANSIBLE VS CODE EXTENSION

You can configure the Ansible VS Code extension to enable Ansible Lightspeed, and to specify the Ansible Lightspeed portal URL and WCA model ID.

Prerequisites
Your organization administrator has configured an IBM watsonx Code Assistant model for your organization.

You are assigned a Red Hat Ansible Lightspeed seat license by your organization administrator.

**Procedure**

1. Open the VS Code application.

2. From the Activity bar, click the Extensions icon.

3. From the Installed Extensions list, select Ansible.

4. From the Ansible extension page, click the Settings icon and select Extension Settings.

5. Select Ansible Lightspeed settings, and specify the following information:
   a. Select the Enable Ansible Lightspeed check box.
   b. In the URL for Ansible Lightspeed field, verify that you have the following URL: https://c.ai.ansible.redhat.com/.

6. Optional: If you want to use a different IBM watsonx Code Assistant model, enter the model ID of the model you want to use in the Model ID Override field.
   Your settings are automatically saved in VS Code.

The following illustration displays the configured settings for the Ansible VS Code extension:

**Figure 6.1. Configured settings for the Ansible VS Code extension**
• Troubleshooting Ansible Visual Studio Code extension errors
CHAPTER 7. REQUESTING TASK RECOMMENDATIONS

Red Hat Ansible Lightspeed is integrated into Visual Studio (VS) Code through the Ansible VS Code extension. You can request code recommendations for your task intent by using Ansible VS Code extension.

7.1. OVERVIEW

You can perform the following tasks from the Ansible VS Code extension:

- **Create single task or multitask requests by using natural language prompts**
  - Create a single task prompt
    Write a description of your task in the **name**: key of a new task line in your Ansible file. For example, to automate a task of installing PostgreSQL server, you can enter the prompt **name: Install postgresql-server**.
  - Create a multitask prompt
    Place your cursor on a new line in your Ansible YAML file at the correct indentation, and start your prompt with a Pound key (#).
    Write the descriptions of your tasks, separating each prompt by using Ampersand symbols (&). For example, to automate a multitask of installing PostgreSQL server and running the initial PostgreSQL setup command, you can enter the prompt **# Install postgresql-server & run postgresql-setup command**.

The Ansible Lightspeed service reads the text, interacts with the IBM watsonx Code Assistant model, and generates Ansible task recommendations based on your natural language prompt.

- **View the content source matching results**
  For each generated code recommendation, Red Hat Ansible Lightspeed lists content source matches, including details such as potential source, content author, and relevant licenses. You can use this data to gain insight into potential training data sources used to generate the code recommendations.

- **Provide feedback on the Ansible Lightspeed service**
  The Ansible Lightspeed service learns your organizational patterns and improves the code recommendation experience over time. You can provide feedback on whether the generated code recommendations were suitable for your task intent. This feedback enables Red Hat Ansible Lightspeed with IBM watsonx Code Assistant to improve on the quality of its suggestions.

7.1.1. Best practices to improve the recommended guidance

Follow these guidelines to improve the likelihood of a quality code recommendation.

- Ensure that your YAML file is properly formatted. See the Ansible YAML syntax guidelines for details.
- Avoid context switching within a single playbook file.
  The Ansible Lightspeed service attempts to correlate earlier tasks to the active recommendation, and the entire contents of the file before the cursor position are used as context by the model. If the earlier task is not relevant to your prompt, VS code provides inline suggestions instead of code recommendations.
- Reword your natural language prompts to get code recommendations that match your task intent. If you get a recommendation that does not align with the intent of your task name, then rewording your prompt to provide more information about what is desired can lead to improved results.

- Use descriptive prompts and provide additional content to improve the code recommendations. Red Hat Ansible Lightspeed reads the full Ansible YAML file when generating a code recommendation. Using descriptive prompts and having additional YAML file content related to the desired task improves the code recommendation. For example, you can add the previous Ansible tasks and appropriate playbook and variable names to improve the code recommendations.

### 7.2. Requesting Code Recommendations for a Single Task

You can request code recommendations for a single task by entering natural language prompts in Ansible VS Code extension. For example, to automate a task of installing a PostgreSQL server, you can enter the prompt `- name: Install postgresql-server`. The Ansible Lightspeed service reads the text, interacts with the IBM watsonx Code Assistant model, and generates the code recommendations.

**Prerequisites**

- You have an assigned seat license to access Red Hat Ansible Lightspeed with IBM watsonx Code Assistant.
- You have installed and configured Ansible VS Code extension.

**Procedure**

1. Log in to VS Code with your Red Hat account.

2. Create a new YAML file or use an existing YAML file:
   - Create a YAML file:
     i. Select File → New Text File.
     ii. From the lower right of the screen, click Plain Text, and in the language mode, select Ansible.
     iii. Save the file as a YAML file format extension (.yml or .yaml).
   - Use an existing YAML file:
     i. On the bottom right of the screen, click the existing language mode, and in the language mode settings, select Ansible.

   **NOTE**
   
   If you do not see the language mode section in your VS Code editor, from the Command Palette, select **Configure Language Mode** → Ansible.

3. Verify that you see an entry for Lightspeed on the status bar at the lower right of VS Code.
If Ansible is already selected as the desired language but the Lightspeed entry is not displayed, re-select Ansible as the language mode. The following illustration shows Lightspeed and Ansible entries on the VS Code status bar.

Figure 7.1. Ansible and Lightspeed set as selected language mode

4. Optional: If you see an error message about missing Ansible lint, you can install the missing module or disable it. Perform any one of the following tasks:

- Install Ansible lint: For installation information, see the Installing section of the Ansible Lint documentation.

- Disable Ansible lint:
  
i. From the Activity bar, click the Extensions icon.
  
ii. From the Installed extensions list, select Ansible.
  
iii. From the Ansible extension page, click the Settings icon and select Extension Settings.
  
iv. Clear the Ansible › Validation › Lint: Enabled checkbox.

5. Create a playbook or use an existing playbook.
6. In the playbook, provide the following information to request code recommendations for a single task:
   
i. Add a new Ansible task by starting a new line with `- name:` at the correct indentation.

   ii. Add a detailed natural language prompt in the task description after `- name:` on the same line. For example, you can specify the following single task prompt:

   ```yaml
   - name: Install postgresql-server
   ```

   iii. Press `Enter` directly after the task description. Keep the cursor at the same location in your file, and wait for the code recommendation results to populate. The Ansible Lightspeed service is engaged, and it starts generating code recommendations for a single task.

   **IMPORTANT**

   Ansible Lightspeed service takes around 5 seconds per task to populate the code recommendations. If you are using a multitask prompt, the Ansible Lightspeed service takes a bit longer (number of tasks times 5 seconds) to populate the results. Do not move your cursor or press any key while the code recommendation is being generated. If you change the cursor location or press any key, Ansible VS Code extension cancels the request and the Ansible Lightspeed service does not process your request.

   When the Ansible Lightspeed service is engaged, a Lightspeed processing status indicator is displayed in the lower right of the screen to denote that your code recommendation is being generated.

   ![Lightspeed Status Indicator]

7. View your code recommendations and ensure that the recommendations match your task intent. The following illustration shows the code recommendations generated by the Ansible Lightspeed service for the single task `Install postgresql-server`:
8. Accept or reject the code recommendations:

- To accept a code recommendation, press **Tab**.
- To reject a code recommendation, press **Esc**.

**NOTE**

If you reject a recommendation, you can modify the prompt and review the generated code recommendations once again to match your task intent.

9. On the **ANSIBLE: LIGHTSPEED TRAINING MATCHES** tab, view the content source matching results.

The following illustration shows the training matches found in existing Ansible Galaxy content for the task prompt **Install postgresql-server**.
10. Click **Save** to save the code recommendation changes in your Ansible YAML file.

**Additional resources**

- Troubleshooting Ansible Visual Studio Code extension errors
- Troubleshooting Ansible code bot errors

### 7.3. REQUESTING CODE RECOMMENDATIONS FOR MULTIPLE TASKS

You can request multitask code recommendations by entering a sequence of natural language task prompts in Ansible VS Code extension. In a YAML file, start a comment by using a Pound key (``#``), and separate each prompt by using Ampersand (``&``) symbols.

For example, to automate a multitask of installing PostgreSQL server and running the initial PostgreSQL setup command, you can enter the prompt **# Install postgresql-server & run postgresql-setup command**. The Ansible Lightspeed service reads the text, interacts with the IBM watsonx Code Assistant models, and generates the code recommendations.

**Prerequisites**
You have an assigned seat license to access Red Hat Ansible Lightspeed with IBM watsonx Code Assistant.

You have installed and configured Ansible VS Code extension.

Procedure

1. Log in to VS Code with your Red Hat account.

2. Create a new YAML file or use an existing YAML file.
   - Create a YAML file:
     i. Select File → New Text File.
     ii. From the lower right of the screen, click Plain Text, and in the language mode, select Ansible.
     iii. Save the file as a YAML file format extension (.yml or .yaml).
   - Use an existing YAML file:
     i. On the bottom right of the screen, click the existing language mode, and in the language mode settings, select Ansible.

   **NOTE**
   If you do not see the language mode section in your VS Code editor, from the Command Palette, select Configure Language Mode → Ansible.

3. Verify that you see an entry for Lightspeed on the status bar at the lower right of VS Code.
   If Ansible is already selected as the desired language but the Lightspeed entry is not displayed, re-select Ansible as the language mode. The following illustration shows Lightspeed entry on the VS Code status bar.
4. Optional: If you see an error message about missing Ansible lint, you can install the missing module or disable it. Perform any one of the following tasks:
   
   - Install Ansible lint: For installation information, see the Installing section of the Ansible Lint documentation.
   
   - Disable Ansible lint:
     
     i. From the Activity bar, click the Extensions icon.
     
     ii. From the Installed extensions list, select Ansible.
     
     iii. From the Ansible extension page, click the Settings icon and select Extension Settings.
     
     iv. Clear the Ansible › Validation › Lint: Enabled checkbox.

5. Create a playbook or use an existing playbook.
   
   For more information, see Creating playbooks in the Ansible Automation Platform Creator Guide.

6. In the playbook, provide the following information to request multitask code recommendations:
i. Start a new YAML file comment by entering a Pound key (#) at the correct indentation.

ii. Add a detailed natural language prompt in a sequence, separating each task by using an Ampersand (&) symbol. For example, to automate the multitask of installing PostgreSQL server and running the PostgreSQL setup command, enter the following natural language prompt:

```
# Install postgresql-server & run postgresql-setup command.
```

iii. Press Enter directly after the task description. Keep the cursor at the same location in your file, and wait for the code recommendation results to populate. The Ansible Lightspeed service is engaged, and it starts generating code recommendations for multiple tasks.

**IMPORTANT**

Ansible Lightspeed service takes around 5 seconds per task to populate the code recommendations. If you are using a multitask prompt, the Ansible Lightspeed service takes a bit longer (number of tasks times 5 seconds) to populate the results. Do not move your cursor or press any key while the code recommendation is being generated. If you change the cursor location or press any key, Ansible VS Code extension cancels the request and the Ansible Lightspeed service does not process your request.

When the Ansible Lightspeed service is engaged, a Lightspeed processing status indicator is displayed in the lower right of the screen to denote that your code recommendation is being generated.

7. Optional: If multitask code recommendations are not being generated, log out of VS Code and log in again using your Red Hat account.

8. View your code recommendations and ensure that the recommendations match your task intent. The following illustration shows the code recommendations generated by the Ansible Lightspeed service for the multitask prompt:

```
Install postgresql-server & run postgresql-setup command.
```

: 

---

**CHAPTER 7. REQUESTING TASK RECOMMENDATIONS**
9. Accept or reject the code recommendations:
   
   - To accept a code recommendation, press Tab.
   
   - To reject a code recommendation, press Esc.

   **NOTE**

   If you reject a recommendation, you can modify the prompt and review the generated code recommendations once again to match your task intent.

10. On the **ANSIBLE: LIGHTSPEED TRAINING MATCHES** tab, view the content source matching results.
    The following illustration shows the training matches found in existing Ansible Galaxy content for the task prompt multitask prompt **Install postgresql-server & run postgresql-setup command**:
11. Click **Save** to save the code recommendation changes in your Ansible YAML file.

Additional resources

- Troubleshooting Ansible Visual Studio Code extension errors
- Troubleshooting Ansible code bot errors

### 7.4. VIEWING ANSIBLE LIGHTSPEED TRAINING MATCHES

The Red Hat Ansible Lightspeed with IBM watsonx Code Assistant machine learning model is trained on the following content:

- Existing public or private Git repositories
- Content from Ansible Galaxy

Owing to IBM watsonx Code Assistant’s generative AI technology, as well as the types of Ansible content that were used to train the model, it is not possible to identify the specific set of training data that contributed to the generated code recommendations. However, Ansible Lightspeed provides a capability that helps you to understand the possible origins of generated code recommendations.

For each generated code recommendation, Red Hat Ansible Lightspeed lists the content source matches, including details such as potential source, content author, and relevant licenses. You can use this data to gain insight into potential training data sources used to generate the code recommendations.
After you enter a natural language prompt in VS Code and see the generated code recommendations, you can view the content source matches on the **ANSIBLE: LIGHTSPEED TRAINING MATCHES** tab.

For example, the following illustration shows the training matches for the multitask recommendation **Install postgresql-server & run postgresql-setup command**

**Figure 7.3. Training matches for a multitask recommendation**

This capability enables you to find out the open source license terms that are associated with related training data. However, it is unlikely that either the training data used in fine-tuning the code or the output recommendations themselves are protected by copyright, or that the output reproduces training data that is controlled by copyright licensing terms.

**NOTE**

Red Hat does not claim any copyright or other intellectual property rights in the suggestions generated by Red Hat Ansible Lightspeed with IBM watsonx Code Assistant.

### 7.5. PROVIDING FEEDBACK ON THE ANSIBLE LIGHTSPEED SERVICE
Red Hat Ansible Lightspeed with IBM watsonx Code Assistant is designed to be improved through feedback on the quality of its suggestions. The technical details of user experiences with Lightspeed are useful in informing further improvements.

You can submit feedback through the following channels:

- From the Ansible VS Code extension: Use this method to provide feedback about the quality of the suggested code recommendations.

  **IMPORTANT**

  Red Hat Support cannot assist with the suggestion quality reports. Content quality issues are routed to IBM for resolution.

- From the Red Hat customer portal: Use this method to log bug reports and service disruption incidents, and feature requests.

  **NOTE**

  On the login screen of the Ansible Lightspeed Portal, there is a Chat link that redirects you to a Matrix channel. Use the Matrix channel to ask questions pertaining to your Ansible Lightspeed experience and request help to troubleshoot your issues. However, the Matrix channel is not an official Support channel, and issues raised in the Matrix chat would not be tracked through Red Hat Service Level Agreement (SLA). To raise a bug or a feature request, contact Red Hat Support and open a support ticket.

Prerequisites

- You have an assigned seat license to access Red Hat Ansible Lightspeed with IBM watsonx Code Assistant.

Procedure


2. Click the Lightspeed entry in your status bar to see options.

3. In the Tell us why field, provide your feedback. Here, provide feedback about what results you were expecting to receive, compared to what results were generated and the training match.

4. Select the issue type: Bug report, Feature request, or Suggestion feedback.

  **NOTE**

  To raise a bug or feature request, contact Red Hat Support and open a support ticket. Bug features and feature requests made through Ansible Lightspeed feedback are not tracked through the Red Hat Service Level Agreement (SLA).

5. Select the I understand that feedback is shared with Red Hat and IBM check box.

6. Click Send.

   The following image shows an example of providing suggestion feedback:
Figure 7.4. Providing feedback on Ansible Lightspeed
CHAPTER 8. INSTALLING AND CONFIGURING ANSIBLE CODE BOT

The Ansible code bot scans existing content collections, roles, and playbooks hosted in GitHub repositories, and proactively creates pull requests whenever best practices or quality improvement recommendations are available. The bot automatically submits pull requests to the repository, which proactively alerts the repository owner to a recommended change to their content. You can configure Ansible code bot to scan your existing Git repositories (both public and private). Your organization must have a valid Red Hat Ansible Lightspeed with IBM watsonx Code Assistant subscription to use Ansible code bot.

Ansible code bot uses Ansible lint to recommend code quality improvements. Ansible code bot is intended to promote proven practices, patterns, and behaviors while avoiding common errors that can easily lead to bugs or make code harder to maintain. Ansible code bot scans your content based on the configured rules, to ensure that your content adheres to Ansible best practices.

NOTE

- Red Hat Ansible Lightspeed provides Ansible code bot in Service Preview, that you can deploy to your GitHub account.
- Ansible code bot is supported on the following GitHub versions:
  - GitHub.com
  - GitHub Enterprise Cloud

The following examples are code recommendations that the Ansible code bot can suggest:

- Available alternatives for deprecated legacy syntax or implementation patterns
- Module version changes and updates, such as:
  - Adding any new required parameters
  - Flagging deprecated parameters
  - Removing unused parameters
- Applying YAML best practices
- Adding comment blocks
- Fixing casing issues in name fields

8.1. INSTALLING ANSIBLE CODE BOT

Install the Ansible code bot to get code recommendations.

Procedure

1. Log in to GitHub by using the account associated with your organization.
2. Install the GitHub app for the organization that you are a member of.
3. Go to the Ansible code bot GitHub app.

4. Select the repositories that you want the app to access:
   - **All repositories**: Provides access to read the metadata of all repositories.
   - **Only select repositories**: Provides access to read the metadata of only the repositories that you select.

5. Optional: If you selected **Only select repositories** in the previous step, select the repositories that you want the Ansible code bot to access from the **Select repositories** list.

6. Click **Install & Authorize**. A message is displayed that specifies the following permissions are granted automatically to the bot during installation:
   - Read access to metadata
   - Read and write access to code and pull requests

7. When prompted, log in to your {RH-SSO} account. You are logged in to Ansible code bot.

The following image shows the Ansible code bot login screen with a message stating **Your subscription is active**:

Figure 8.1. Ansible code bot login screen

![Welcome! Ansible code bot](image)

Your subscription is active

After the Ansible code bot is installed for the Git repositories you selected, you can configure a schedule to scan your Git repositories at regular intervals. You can also manually scan your Git repositories if you have not set up a scanning schedule for your Ansible code bot or if you do not want to wait for the next scheduled scan.

**Additional resources**

- Configuring Ansible code bot to scan your repository at regular intervals
- Manually scanning your Git repositories
Troubleshooting Ansible code bot errors

8.2. CONFIGURING ANSIBLE CODE BOT TO SCAN YOUR REPOSITORY AT REGULAR INTERVALS

You can schedule the Ansible code bot to scan your repositories at daily, weekly, or monthly intervals. To specify the scan schedule, use the configuration file `ansible-code-bot.yml` that was created automatically when Ansible code bot was installed. If the file is not created automatically, use the procedure below to create the `ansible-code-bot.yml` file and specify your scan schedule.

You can specify one of following interval cadence to scan your Git repositories:

- **Daily**: Runs every day from Monday to Sunday.
- **Weekly**: Runs once a week. By default, this is on Monday.
- **Monthly**: Runs on the first day of the month, once each month.

For each interval cadence, Ansible code bot starts scanning your Git repositories at 9 AM UTC.

Procedure

1. In GitHub, navigate to the configuration file in the repository that you want to scan.

2. Optional: If the `ansible-code-bot.yml` file was not created automatically, manually create a `.yml` configuration file named `ansible-code-bot.yml` in your repository `.github` folder. For example, `github/ansible-code-bot.yml`.

3. In the configuration file, specify the `interval` parameter. You can specify the `interval` parameter as **daily**, **weekly**, or **monthly**. For example:

   ```yaml
   schedule:
     interval: "<daily | weekly | monthly>"
   ```

4. Save your changes.

Ansible code bot starts scanning your repository per the schedule you configured at 9 AM UTC time.

8.3. MANUALLY SCANNING YOUR GIT REPOSITORIES

You can manually scan your Git repositories if you did not set up a scanning schedule for your Ansible code bot or if you do not want to wait for the next scheduled scan.

If the avoid duplicate PRs condition is not met, then a manual scan results in a new PR with all the necessary Ansible code bot recommendations.

Following are the avoid duplicate conditions:

- Verify that the Ansible code bot pull request (PR) already exists, and is on the latest commit main branch. If this condition is met, you can skip the scan.

- If there is an existing PR but it is not on the latest commit branch, the Ansible code bot takes a PR difference. If there is no difference, then the bot skips pushing the scan results as a new pull request.
Procedure

1. In GitHub, go to the main page of the repository that you want to scan.

2. Click the Settings icon.

3. In the Topics field, enter the keyword topic `ansible-code-bot-scan` to the repository.
   The following illustration shows the keyword topic for starting a manual scan:

   ![Edit repository details](image)

   - **Description**
     - Automate the deployment of Red Hat OpenShift Container Platform on IBM zSystems (s39)
   
   - **Website**
     - [https://ibm.github.io/Ansible-OpenShift-Provisioning/](https://ibm.github.io/Ansible-OpenShift-Provisioning/)

   - **Topics (separate with spaces)**
     - `ansible-code-bot-scan`

   - Include in the home page
     - Release
     - Packages
     - Deployments

4. Click Save changes.
   Based on the repository webhook event, Ansible code bot starts a manual scan of your repository. If the avoid duplicate pull requests condition is not met, then the manual scan results in a new pull request with all the necessary Ansible code bot recommendations.

Additional resources

- Troubleshooting Ansible code bot errors

8.4. HOW ANSIBLE CODE BOT HANDLES DUPLICATE PULL REQUESTS

- If Ansible code bot has created a pull request on the latest commit default branch, it does not scan the repository. The bot skips scanning the repository because the pull request was committed on the latest default branch, and no new commit was made after that pull request.

- If there is an existing pull request that is not on the latest commit default branch, Ansible code bot does a pull request difference to compare the changes in both the branches. The following scenarios are possible:
- **There is no difference in the existing and new scan results** Ansible code bot does not push the scan results as a new pull request.

- **There are differences found in the existing and the new scan results** Ansible code bot creates a new pull request. The newly-created pull request does not close the existing pull request, against which the pull request difference was noted. This behavior makes it easier for the repository administrator to review only the latest pull request created by the Ansible code bot, and the administrator can avoid reviewing the older pull requests created by the bot. If required, the administrator can close the older pull requests.
CHAPTER 9. TROUBLESHOOTING

This section contains information to help you diagnose and resolve issues with using Red Hat Ansible Lightspeed with IBM watsonx Code Assistant.

9.1. TROUBLESHOOTING RED HAT ANSIBLE LIGHTSPEED CONFIGURATION ERRORS

9.1.1. Cannot access the Ansible Lightspeed administrator portal

The Ansible Lightspeed administrator portal can be accessed by the Red Hat organization administrator only.

If you are the Red Hat organization administrator, before you access the Ansible Lightspeed administrator portal, ensure that:

- You have a valid Ansible Automation Platform subscription.

9.1.2. Cannot upload the WCA key

When you enter the Watsonx Code Assistant (WCA) API key, authentication fails and shows the following error message:

IBM Cloud API key is invalid

Red Hat Ansible Lightspeed verifies the WCA API key by generating an associated access token. To resolve the error, ensure that you have not accidentally included any extra spaces when obtaining the WCA API key from IBM watsonx Code Assistant. If you still cannot upload the WCA key, contact IBM support.

9.1.3. Cannot configure the model ID due to authentication failure

When you enter the model ID in the Red Hat Ansible Lightspeed administrator portal, the authentication fails.

To resolve the error, ensure that:

- You have configured a valid WCA API key before you upload the model ID.
- You have not accidentally included any extra spaces when entering the model ID.

9.1.4. Cannot configure the model ID due to inference failure

While validating the model ID, Red Hat Ansible Lightspeed performs a test inference. If Red Hat Ansible Lightspeed detects an error, the validation fails and an Inference failed message is displayed.

To resolve the error, ensure that:

- You have a valid WCA API key and model ID.
- You have not accidentally included any extra spaces when obtaining the WCA key and model ID from IBM watsonx Code Assistant.
9.2. TROUBLESHOOTING ANSIBLE VISUAL STUDIO CODE EXTENSION ERRORS

9.2.1. Cannot view the generated code recommendations using the Ansible VS Code extension

The following scenarios are possible:

- You receive a **403 error** message. To resolve this error, ensure that:
  - Your organization administrator has configured Red Hat Ansible Lightspeed for your organization.
  - You have a valid Red Hat Ansible Lightspeed seat license.
- You have not configured the required Ansible VS code extension settings. To resolve this error, ensure that you have enabled the **Lightspeed:Enabled** and **Lightspeed → Suggestions:Enabled** settings. For more information, see Configure the Ansible VS Code extension.
- You receive a **Failure on completion requests** error when you make inference requests in VS Code. If you have an assigned seat license for Red Hat Ansible Lightspeed but your organization administrator has not configured an IBM watsonx Code Assistant model for your organization, you will encounter a **Failure on completion requests** error when you make inference requests in VS Code.
- You are logged in using an account that does not have a valid Red Hat Ansible Lightspeed seat license. If you logged in using an account that does not have an assigned seat license, you must log out and log in again. In VS Code, click the Accounts icon and sign in using your Red Hat account with an assigned seat license.
- Your VS Code Workspace settings override user settings. If your Workspace settings are configured, they can override our user settings even if you have configured the Ansible VS Code extension correctly. The Workspace settings can disable your VS Code extension settings, and therefore you cannot access the Ansible Lightspeed service. To resolve this error, ensure that there are no Workspace settings configured in VS Code. For more information, see Workspace settings in the VS Code documentation.
- You entered a multitask prompt, but code recommendations were not generated. To resolve this error, log out of VS Code and log in again using your Red Hat account.
- You clicked a different location or switched to a different window; therefore, the populated code recommendations disappeared. Red Hat Ansible Lightspeed service could take multiple seconds per task to populate the code recommendations. If you are using a multitask prompt, the Red Hat Ansible Lightspeed service takes a bit longer to populate the results. Do not move your cursor or press any key while the code recommendation is being generated. If you change the cursor location or press any key, Ansible VS Code extension cancels the request and the Red Hat Ansible Lightspeed service does not process your request. In this scenario, you must get the cursor back to its original position and repopulate the results.
9.3. TROUBLESHOOTING ANSIBLE CODE BOT ERRORS

9.3.1. Cannot access Ansible code bot

After you install Ansible code bot and attempt to log in, you receive the following error message:

**Your organization does not have a valid Red Hat Ansible Lightspeed subscription**

After you install Ansible code bot, you are redirected to a page that shows an active subscription status, as shown in the following image:

Figure 9.1. Ansible code bot login screen with an active subscription

If the login screen displays an inactive subscription status, Ansible code bot does not scan your Git repositories. The error occurs because your organization does not have a valid Ansible Automation Platform subscription. To resolve this error, ensure that you are part of an organization that has a valid Red Hat Ansible Automation Platform subscription.

9.3.2. Cannot scan your Git repository using Ansible code bot

If the Ansible code bot is not configured correctly, it does not scan your Git repositories or does not create pull requests.

To resolve Ansible code bot errors, ensure that:

- You have selected all the Git repositories that you want to scan.
- You have a `.yml` configuration file named `ansible-code-bot.yml` in your repository `.github` folder. For example, `.github/ansible-code-bot.yml`.

Run a manual scan on your git repositories by adding the `ansible-code-bot-scan` topic to your repository. For more information, see Manually scan your Git repositories.

If the Ansible code bot still cannot scan your Git repository, the following scenarios are possible:
• The Ansible code bot did not identify any ansible-lint violations in the Git repository.

• The Ansible code bot does not have permission to scan the Git repository.

• Your organization does not have a valid Red Hat Ansible Automation Platform subscription.