



Top 20 Ansible Interview Questions and Answers

Introduction

DevOps roles often include questions on automation tools. Ansible appears in many of those discussions. Teams use Ansible for server setup, configuration tasks, and application deployment across multiple systems. Because many companies manage large infrastructure environments, knowledge of this tool receives strong attention during technical hiring.

Interview panels test both basic concepts and practical use. You face questions about playbooks, inventory files, modules, roles, and automation workflows. Clear understanding of these topics helps you explain how tasks run across servers in a controlled way.

This guide on **Ansible Interview Questions and Answers** presents topics often discussed in DevOps interviews. Review these questions before attending your next technical round.

List of Ansible Interview Questions for Freshers

1. What is Ansible?
2. How do you install Ansible?
3. Describe the Ansible architecture.
4. What are Ansible playbooks?
5. What characteristics does Ansible have?
6. Name Ansible tools
7. What is CI/CD?
8. What is configuration management?
9. What specifications does the Ansible server need to meet?
10. How do Ansible tasks work?
11. Describe Ansible-doc.

12. Describe the facts about Ansible.

Ansible Technical Interview Questions and Answers for Freshers

1. What is Ansible?

Ansible is a free and open-source automation tool for application deployment, configuration management, and software provisioning.

2. How do you install Ansible?

Package managers such as apt or yum on Linux, pip on macOS and Windows, or Ansible's pre-built packages can be used to install Ansible on various operating systems.

3. Describe the Ansible architecture.

Ansible has a client-server architecture in which managed nodes and the control node connect via SSH. The control node runs the Ansible installation and performs the actions indicated in the playbook.

4. What are Ansible playbooks?

YAML files known as Ansible playbooks specify a series of actions to be carried out on distant hosts. They make it possible to automate intricate setups and tasks.

5. What characteristics does Ansible have?

Features of Ansible include:

Agentless/SSH/Push: Unlike Puppet or Chef, there is no software or agent node administration.

Python: Python is an easy-to-learn programming language on which Ansible is built. It is very easy to write scripts on and quite sturdy.

Simple to set up and learn: Ansible is a simple program to use. Since it's open source, anybody can use it.

Take your Knowledge
Test Report

Check your Score



6. Name Ansible tools

A list of many tools you may bring up in an Ansible developer interview is provided below:

Ansible Galaxy: To help you save time, the Galaxy website lets you search for and share community roles.

Ansible Tower is a command-line utility that offers more features along with a graphical user interface.

Microsoft created **Visual Studio Code**, a code editor that can be extended to handle YAML and Ansible.

Atom is a free and open-source text editor useful for integrating with YAML files and tracking project modifications.

Check out: [Ansible Course Syllabus PDF](#)

7. What is CI/CD?

One tool used to streamline the development and deployment processes is continuous integration, or CI. Coherent software has developed quickly as a result. An automatic build verifies each integration to identify integration faults as soon as possible.

Your code can be pushed to a remote repository and then deployed to production at any moment with continuous delivery (CD). To put it another way, it's the process of creating software that can be released into production at any time.

Deep dive into DevOps culture with our [Nagios training in Chennai](#).

8. What is configuration management?

One procedure we should use to monitor any updates that are made to the system over time is configuration management (CM).

This is especially helpful when a significant bug that must be resolved with the least amount of downtime has been introduced into the system as a result of some recent changes.

System configuration management (CMM) maintains track of all necessary system upgrades and makes sure that the system's build and design are current and operating correctly.

9. What specifications does the Ansible server need to meet?

In case you operate Windows, you must have a virtual machine where Linux needs to be installed. Python 2.6 or later is required to use Ansible Server. If these conditions are met, you can move on without difficulty.

10. How do Ansible tasks work?

In Ansible, the task is a single action. It divides a configuration policy into more manageable files or code segments. These building elements can help automate certain tasks, such as updating software or installing packages:

1 Command: Install <package_name>

2 Command: update <software_name>

11. Describe Ansible-doc.

Ansible-doc shows details about installed modules in Ansible libraries. It generates a brief sample that can be put into a playbook, prints out the documentation strings for each plug-in, and presents an inventory of plug-ins together with a brief explanation.

12. Describe the facts about Ansible.

You may think of Ansible facts as a means for Ansible to gather host-related data and store it in variables for convenient access. The script can make use of this data, which is kept in predefined variables. Ansible runs the set-up module to produce facts.

Check out: [Ansible Salary in Chennai](#)

List of Ansible Interview Questions for Experienced

13. What do Ansible callback plug-ins mean?

14. Define Ansible inventory and its types.

15. What is an ansible vault?

16. How do we write an Ansible handler with multiple tasks?

17. Is it possible to copy files onto a destination host recursively? If so, how?

18. How are Ansible projects tested?

19. What distinguishes the Ansible `set_fact` module from `include_var`, `vars`, and `vars_file`?

20. How can content be made more redistributable and reusable?

Ansible Technical Interview Questions and Answers for Experienced

13. What do Ansible callback plug-ins mean?

Callback plug-ins manage the output that appears when we run CMD (Command Prompt) programs. In addition, it can be utilized to add more outputs, or even more than one.

For instance, the mail callback is used to send an email on playbook failures, and the log_plays callback records playbook events into a log file.

Additionally, you can add custom callback plug-ins by placing them inside a role, in a callback directory source that is set in *ansible.cfg*, or by dropping them into a `callback_plugins` directory next to play.

14. Define Ansible inventory and its types.

The hosts and host groups that the tasks, commands, and modules in a playbook will run on are defined in an Ansible inventory file.

Static and dynamic inventory files are the two types available in Ansible, and they are described as follows:

Static Inventory: A static inventory file is a plain text file that contains a list of managed hosts that have been defined under a host group using either hostnames or IP addresses.

In each line, the managed host entries are listed beneath the group name.

Dynamic Inventory: A script built in Python, any other programming language, or—ideally—with the help of plug-ins produces dynamic inventory.

Static inventory file settings in a cloud setup won't work since IP addresses change when virtual servers are stopped and restarted.

Train your brain for faster app development with our [Git Training in Chennai](#)

15. What is an ansible vault?

Passwords and other sensitive information are stored in the Ansible vault instead of being stored in playbooks or roles as plain text. Ansible may encrypt a single value within a YAML file or any structured data file.

The data is encrypted using the following command:

1 Command: ansible-vault encrypt foo.yml bar.yml baz.yml

The data must be decrypted using the following command:

1 Command: ansible-vault decrypt foo.yml bar.yml baz.yml

16. How do we write an Ansible handler with multiple tasks?

Assume you wish to write a handler that will only restart a service if it is already operational.

Tasks can notify those subjects as indicated below, and handlers can comprehend generic themes. Triggering numerous handlers is significantly simpler with this functionality.

Additionally, it separates handlers from their names, which facilitates handler sharing between roles and playbooks.

– name: Check if restarted

shell: check_is_started.sh

register: result

listen: Restart processes

– name: Restart conditionally step 2

service: name=service state=restarted

when: result

listen: Restart processes

17. Is it possible to copy files onto a destination host recursively? If so, how?

The copy module allows us to copy files recursively onto a target host. It can copy files from a directory using a recursive argument. This is particularly the purpose of another module named “Synchronize.” The directive for the same is provided below:

– *synchronize:*

src: /first/absolute/path

dest: /second/absolute/path

delegate_to: “{{ inventory_hostname }}”

**Take your Knowledge
Test Report**

Check your Score



18. How are Ansible projects tested?

Three testing approaches are available, and they are listed below:

Asserts: Asserts mimic the behavior of the test in languages other than Python. It checks to see if your system has arrived at its target state rather than merely operating in check mode as a simulation. Assertions indicate

if the task modified the relevant resources and completed the task at hand.

Check Mode: In check mode, you can see how things would function in the absence of the simulation. As a result, it is simple to determine whether the project performs as anticipated.

The scripts and commands used in the roles and playbooks are not executed in check mode, which is a limitation. To get past this, we must use the following command to disable check mode for particular tasks:

Command: check_mode: no

Manual Run: The play is just executed to see if the system is operating as intended. The simplest testing option may also be the riskiest since it creates a test environment that might not be identical to the production environment.

Get complete hands-on experience with DevOps practices through our [Jenkins training program](#).

19. What distinguishes the Ansible `set_fact` module from `include_var`, `vars`, and `vars_file`?

Ansible's `set_fact` function is used to set new variable values on a host-by-host basis, much like Ansible facts, which are located by the `setup` module.

A playbook's subsequent plays can use these variables. When using `set_fact`, we can save the value after preparing it on the fly using specific tasks such as applying filters or taking subparts of another variable.

In contrast, when using `vars`, `vars_file`, or `include_var`, we know the value ahead. A fact cache can also be set over it.

The `set_fact` variable is assigned using key-pair values, where the assignment to the variable is the value and the variable name is the key. Below is a basic example of what it looks like.

– *set_fact:*

one_fact: value1

second_fact:

value2

20. How can content be made more redistributable and reusable?

To enable the reuse and redistribution of content, you can utilize Ansible roles. Playbooks can be arranged essentially at an abstract level using Ansible roles.

For instance, setting down every activity that needs to be completed on five different systems in the playbook could result in errors and confusion. Rather, we develop ten roles and refer to them within the playbook.

Check out: [Ansible Training in OMR](#)

Conclusion

Ansible knowledge supports many DevOps roles. Companies run large numbers of servers. Manual setup wastes time and creates errors. Automation solves this problem. Ansible handles server configuration, software deployment, and routine infrastructure tasks. Recruiters check this knowledge during DevOps interviews.

Prepare with focus on real usage. Study playbooks, inventory files, roles, and modules. Practice automation tasks in a lab setup. Clear understanding helps you explain workflows during technical discussions.

If you want guided practice, join our [Ansible Training in Chennai](#). You learn automation through hands on exercises and project based sessions. This training builds job ready DevOps skills and supports better preparation for technical interviews.