

## DevOps Questions and Answers

**DevOps** is a set of practices, principles, and tools that combines software development (Dev) and IT operations (Ops). Its primary goal is to shorten the software development lifecycle and deliver high-quality software continuously. **DevOps** includes collaboration, communication, automation, continuous delivery, and integration between software developers and IT operations professionals to improve the speed, efficiency, and reliability of software delivery.

### What is DevOps?

**DevOps** is a transformative culture and practice that unites software development (Dev) and IT operations (Ops) teams. By fostering collaboration and leveraging automation technologies, DevOps enables faster, more reliable code deployment to production in an efficient and repeatable manner.

### What is a DevOps Engineer?

A **DevOps Engineer** is a professional who combines software development (Dev) and IT operations (Ops) skills to improve and streamline the process of developing, testing, and releasing software.

Their goal is to ensure that software is delivered quickly, efficiently, and reliably. They work to automate and integrate the processes between software development and IT teams, allowing for continuous delivery and continuous integration of software.

#### 1. What is the use of SSH?

**SSH(Secure Shell)** is an access credential used in the SSH Protocol. In other words, it is a cryptographic network protocol that transfers encrypted data over the network. It allows you to connect to a server, or multiple servers, without having to remember or enter your password for each system that is to log in remotely from one system to another.

#### 2. What is CI/CD?

**CI And CD** is the practice of automating the integration of code changes from multiple developers into a single codebase. It is a software development practice where the developers commit their work frequently to the central code repository (Github or Stash).

- **Continuous Integration:** With Continuous Integration, developers frequently commit to a shared common repository using a version control system such as Git. A continuous integration pipeline can automatically run builds, store the artifacts, run unit tests, and even conduct code reviews using tools like Sonar.
- **Continuous Delivery:** Continuous delivery helps developers test their code in a production-similar environment, hence preventing any last-moment or post-production surprises. These tests may include UI testing, load testing, integration testing, etc. It helps developers discover and resolve bugs preemptively.

#### 3. What is Hypertext Transfer Protocol Secure(HTTPS)?

**Hypertext Transfer Protocol Secure is a protocol** that is used to communicate between the user browser and the website. It also helps in the transfer of data. It is the secure variant of HTTP. To make the data transfer more secure, it is encrypted. Encryption is required to ensure security while transmitting sensitive information like passwords, contact information, etc.

#### 4. What is the Blue/Green Deployment Pattern?

Blue Green Deployment is just like we deploy two versions of our application, one is the stable version, and another is a new feature or bug fix let's say, forwarding a certain percentage of traffic to the second version as well in production to ensure that everything is working fine.

- **Blue Deployment:** It's the primary Deployment that is stable, and being used as production.

- **Green Deployment:** It's a kind of clone version, but it has additional changes in it, we can route the traffic to the Green deployment so that if any issues are there in the Deployment we can fix them and then promote it to Blue, so that reducing the chances of failures in production environment.

## 5. What's the difference between DevOps & Agile?

Agile	DevOps
Agile is a method for creating software.	It is not related to software development. Instead, the software that is used by DevOps is pre-built, dependable, and simple to deploy.
An advancement and administration approach.	Typically a conclusion of administration related to designing.
The agile handle centers on consistent changes.	DevOps centers on steady testing and conveyance.
Agile relates generally to the way advancement is carried out, any division of the company can be spry on its hones. This may be accomplished through preparation.	DevOps centers more on program arrangement choosing the foremost dependable and most secure course.

## 6. What is the continuous testing process?

Continuous testing is a process of automated testing done on software continuously as soon as a piece of code is delivered by the developers. This testing is done at every stage starting from the initial stages of development until the deployment of software.

## 7. What is 'Pair Programming'?

Pair programming is a development technique in which two programmers work together at a single workstation. A person who writes code is called a driver and a person who observes and navigates each line of the code is called a navigator. They may switch their role frequently.

## 8. What do you mean by Configuration Management?

The process of controlling and documenting change for the development system is called Configuration Management. Configuration Management is part of the overall change management approach. It allows large teams to work together in a stable environment while still providing the flexibility required for creative work.

## 9. What is Infrastructure as Code (IaC)?

[Infrastructure as Code \(IaC\)](#) is a method of managing and provisioning IT infrastructure using code, rather than manual configuration. It allows teams to automate the setup and management of their infrastructure, making it more efficient and consistent. This is particularly useful in the DevOps environment, where teams are constantly updating and deploying software.

## 10. Explain the concept of branching in Git.

Branching means diverging from the mainline and continuing to work separately without messing with the mainline. Nearly every VCS has some form of branch support. In Git, a branch is simply a reference to the commit, where the following commits will be attached.

### 11. What is Git stash?

The [Git stash](#) command can be used to accomplish this if a developer is working on a project and wants to preserve the changes without committing them. This will allow him to switch branches and work on other projects without affecting the existing modifications. You can roll back modifications whenever necessary, and it stores the current state and rolls back developers to a prior state.

### 12. What is a GIT Repository?

Repositories in GIT contain a collection of files of various versions of a Project. These files are imported from the repository into the local server of the user for further updations and modifications in the content of the file. A VCS or the [Version Control System](#) is used to create these versions and store them in a specific place termed a repository.

### 13. What is Dogpile effect?

Cache Stampede or Dogpile Problem in [System Design](#) is a phenomenon that can occur in systems that rely on caching to improve performance. As a result, the system experiences a sudden surge in demand, often overwhelming the backend resources and causing a performance degradation.

### 14. What Is Jenkins?

[Jenkins](#) is a tool that is used for automation, and it is an open-source server that allows all the developers to build, test and deploy software. It works or runs on java as it is written in java. By using Jenkins we can make a continuous integration of projects(jobs) or end-to-endpoint automation.

### 15. What is the use of the cherry-pick command in git?

[Git cherry-pick](#) in git means choosing a commit from one branch and applying it to another branch. This is in contrast with other ways such as merge and rebases which normally apply many commits into another branch.

**The command for Cherry-pick is as follows:**

```
git cherry-pick<commit-hash>
```

### 16. What is sudo command in Linux?

Sudo (Super User **DO**) command in Linux is generally used as a prefix for some commands that only superusers are allowed to run. If you prefix any command with “sudo”, it will run that command with elevated privileges or in other words allow a user with proper permissions to execute a command as another user, such as the superuser. This is the equivalent of the “run as administrator” option in Windows.

### 17. What's the Difference Between Git Fetch and Git Pull ?

Git Fetch	Git Pull
Used to fetch all changes from the remote repository to the local repository without merging into the current working directory	Brings the copy of all the changes from a remote repository and merges them into the current working directory
Repository data is updated in the .git directory	The working directory is updated directly
Review of commits and changes can be done	Updates the changes to the local repository immediately.
Command for <a href="#">Git fetch</a> is git fetch<remote>	Command for <a href="#">Git Pull</a> is git pull<remote><branch>

## 18. What are the components of Selenium?

Selenium is a powerful tool for controlling web browser through program. It is functional for all browsers, works on all major OS and its scripts are written in various languages i.e Python, Java, C#, etc, we will be working with Python.

Selenium has four major components :-

- Selenium IDE
- Selenium RC
- Selenium Web driver
- Selenium GRID

## 19. What is a Puppet in DevOps?

Puppet is an open-source configuration management automation tool. Puppet permits system administrators to type in infrastructure as code, using the Puppet Descriptive Language rather than utilizing any customized and individual scripts to do so. This means in case the system administrator erroneously alters the state of the machine, at that point puppet can uphold the change and guarantee that the framework returns to the required state.

## 20. What is Ansible?

Ansible is an open-source IT engine that automates application deployment, cloud provisioning, intra-service orchestration, and other IT tools. [Ansible](#) can be used to deploy the software on different servers at a time without human interaction. Ansible can also be used to configure the servers and create user accounts. Ansible is an agent-less software which means there is no need to install the software in the nodes which means you need to do the SSH to connect the nodes to perform the required operations on the servers.

## 21. What is Automation Testing?

Automated Testing is a technique where the Tester writes scripts on their own and uses suitable Software or Automation Tool to test the software. It is an Automation Process of a Manual Process. It allows for executing repetitive tasks without the intervention of a Manual Tester.

## 22. What is the importance of continuous feedback in DevOps?

Continuous Feedback in software testing is trying out an iterative process that involves presenting everyday comments, reviews, and critiques during the software program improvement lifecycle. It ensures that builders get an equal message approximately the quality and functionality of their code. Let's delve deeper into this concept little by little and discover the variations associated with it.

## 23. What is Git Bash?

[Git Bash is a command-line interface \(CLI\)](#) application for Windows that lets you communicate with Git, the version control system. Clone the repositories, commit changes, push and pull changes, and more are all possible using Git Bash. Git Bash can automate manual tasks with the scripts written by you. Git Bash helps you in a greater way to learn about Git and version control.

## 24. What is Git Squashing?

[Squashing](#) combines multiple commits into a single commit based on your commit history. With the help of squashing you can clean your branch history and can maintain an organized commit timeline. It is used before pulling requests or merging feature branches.

## 25. What is a merge conflict in Git?

[Merge Conflicts](#) are the conflicts that occur when a developer is editing a file in a particular branch and the other developer is also editing that same file or when developer A edits some line of code and that same line of code is being edited by another developer B that leads to conflicts while merging.

## 26. What is Git prune?

Git prune is a command that deletes all the files that are not reachable from the current branch. The prune command is useful when you have a lot of files in your working directory that you don't want to keep. The primary use of git prune is to clean up your working directory after you have finished working on a project. What actually git prune does is, suppose you have an object or commit that is no longer reachable from the current branch.

### Command:

```
git fetch --prune <remote>
```

## 27. What's the difference between HTTP and HTTPS ?

<a href="#">HTTP</a>	<a href="#">HTTPS</a>
HTTP does not use data hashtags to secure data.	While HTTPS will have the data before sending it and return it to its original state on the receiver side.
In HTTP Data is transfer in plaintext.	In HTTPS Data transfer in ciphertext.
HTTP does not require any certificates.	HTTPS needs SSL Certificates.
HTTP does not improve search ranking	HTTPS helps to improve search ranking

## 28. Explain the “Shift left to reduce failure” concept in DevOps?

In DevOps, "[shift left](#)" means bringing testing and security audits earlier in the development cycle. Problems are recognized and resolved early, which reduces the likelihood of errors and failures in subsequent phases, boosting the efficiency and dependability of the development pipeline.

## 29. What are Virtual machines (VMs) ?

Virtual Machine abstracts the hardware of our personal computer such as CPU, disk drives, memory, NIC (Network Interface Card) etc, into many different execution environments as per our requirements, hence giving us a feel that each execution environment is a single computer. For example, VirtualBox.

## 30. What is a bare repository?

A [bare repository](#) is the same as default, but no commits can be made in a bare repository. The changes made in projects cannot be tracked by a bare repository as it doesn't have a working tree. A working tree is a directory in which all the project files/sub-directories reside. Bare repository is essentially a .git folder with a specific folder where all the project files reside.

## 31. What is Component-Based Model (CBM) in DevOps?

The component-based assembly model uses object-oriented technologies. In object-oriented technologies, the emphasis is on the creation of classes. Classes are the entities that encapsulate data and algorithms. In component-based architecture, classes (i.e., components required to build application) can be uses as reusable components.

## 32. How to Make a CI-CD Pipeline in Jenkins?

DevOps professionals mostly work with pipelines because pipelines can automate processes like building, testing, and deploying the application. With the help of [Continuous Integration / Continuous](#)

[Deployment \(CI/CD\)](#) Pipeline scripts we can automate the whole process which will increase productivity save lots of time for the organization and deliver quality applications to the end users.

### 33. What's the difference between Chef and Puppet?

Chef	Puppet
Ruby programming knowledge is needed to handle the management of <a href="#">Chef</a> .	DSL programming knowledge is needed to handle the management of Puppet.
Chef is mostly used by small and medium-sized companies for management.	Large corporations and enterprises use Puppet for management.
There is no error visibility at installation time which results in difficulty.	Error visibility at installation time is provided to ease the installation process.
The transmission process to establish communication in this software is slower as compared to Puppet.	The transmission process to establish communication in this software is faster as compared to Chef.

### 34. What is Git Rebase?

[Rebasing in Git](#) is a process of integrating a series of commits on top of another base tip. It takes all the commits of a branch and appends them to the commits of a new branch. The main aim of rebasing is to maintain a progressively straight and cleaner project history. Rebasing gives rise to a perfectly linear project history that can follow the end commit of the feature all the way to the beginning of the project without even forking. This makes it easier to navigate your project.

The technical syntax of rebase command is:

```
git rebase [-i | --interactive] [ options ] [--exec cmd] [--onto newbase | --keep-base] [upstream [branch]]
```

### 35. What is Selenium Tool Suite?

Selenium is a very well-known open-source software suite, mainly used for testing web browsers and web applications by automating some processes. It comes with a set of tools and libraries that allow developers or testers to automate some functions related to web browsers and web applications. Selenium Tool suite consists of 4 major components:

- Selenium IDE (Integrated Development Environment)
- Selenium WebDriver
- Selenium Grid
- Selenium Remote Control (Deprecated)

### 36. What is Selenium IDE?

[Selenium IDE \(Integrated Development Environment\)](#) is an open-source web testing solution. Selenium IDE is like a tool that records what you do on a website. Subsequently, these recorded interactions can be replayed as automated tests. You don't need much programming skills to use it. Even if you're not great at programming, you can still make simple automated tests with it.

### 37. What is Banker's Algorithm in OS?

The banker's algorithm is a resource allocation and deadlock avoidance algorithm that tests for safety by simulating the allocation for the predetermined maximum possible amounts of all resources, then makes an



“s-state” check to test for possible activities, before deciding whether allocation should be allowed to continue.

### 38. How do you create a backup and copy files in Jenkins?

In [Jenkins](#), create a backup by copying the JENKINS\_HOME directory, which contains all configurations and job data. To copy files, use the `sh` or `bat` command in a pipeline script, such as `sh 'cp source_file destination'` for Unix or `bat 'copy source_file destination'` for Windows. Use plugins like "ThinBackup" for scheduled backups

### 39. Explain how you can set up a Jenkins job?

To set up a Jenkins job:

1. Open Jenkins and log in with your credentials.
2. Click "New Item" from the dashboard.
3. Enter a name for your job and select the job type (e.g., Freestyle project).
4. Click "OK" to create the job.
5. Configure your job by adding a description, source code management details (e.g., Git repository), and build triggers.
6. Add build steps, such as shell commands or invoking scripts.
7. Save the job and click "Build Now" to run it.

### 40. Explain the architecture of Docker.

[Docker architecture](#) consists of several key components:

1. **Docker Client:** Issues commands to the [Docker daemon](#) via a command-line interface (CLI).
2. **Docker Daemon (dockerd):** Runs on the host machine, managing Docker objects like images, containers, networks, and volumes.
3. **Docker Images:** Read-only templates used to create [Docker containers](#).
4. **Docker Containers:** Lightweight, portable, and executable instances created from [Docker images](#).
5. **Docker Registry:** Stores and distributes Docker images; Docker Hub is a popular public registry.
6. **Docker Compose:** A tool for defining and running multi-container Docker applications using a YAML file.
7. **Docker Networking:** Allows containers to communicate with each other and with non-Docker environments.

### 41. What is the DevOps life cycle?

DevOps Lifecycle is the set of phases that includes DevOps for taking part in Development and Operation group duties for quicker software program delivery. DevOps follows positive techniques that consist of code, building, testing, releasing, deploying, operating, displaying, and planning. [DevOps lifecycle](#) follows a range of phases such as non-stop development, non-stop integration, non-stop testing, non-stop monitoring, and non-stop feedback. 7 Cs of DevOps are:

- Continuous Development
- Continuous Integration
- Continuous Testing
- Continuous Deployment/Continuous Delivery
- Continuous Monitoring
- Continuous Feedback
- Continuous Operations

#### 42. What is the difference between Git Merge and Git Rebase?

Git Merge	Git Rebase
Git Merge merges two branches to create a “feature” branch.	Git Rebase rebases the feature branch to add the feature branch to the main branch.
Git Merge is comparatively easy.	Git Rebase is comparatively harder.
Git Merge safeguards history.	<a href="#">Git Rabse</a> doesn’t safeguard history.
Git Merge is more suitable for projects with the less active main branch.	Git Rebase is suitable for projects with frequently active main branches.

#### 43. What's the difference between DataOps and DevOps?

DataOps	DevOps
The DataOps ecosystem is made up of databases, data warehouses, schemas, tables, views, and integration logs from other significant systems.	This is where <a href="#">CI/CD pipelines</a> are built, where code automation is discussed, and where continual uptime and availability improvements happen.
Dataops focuses on lowering barriers between data producers and users to boost the dependability and utility of data.	Using the DevOps methodology, development and operations teams collaborate to create and deliver software more quickly.
Platforms are not a factor in <a href="#">DataOps</a> . It is a collection of ideas that you can use in situations when data is present.	DevOps is platform-independent, but cloud providers have simplified the playbook.
Continuous data delivery through automated modeling, integration, curation, and integration. Processes like data governance and curation are entirely automated.	Server and version configurations are continuously automated as the product is being delivered. Automation encompasses all aspects of testing, network configuration, release management, version control, machine and server configuration, and more.

#### 44. What are the 7 Cs of DevOps?

The 7 Cs of DevOps are:

1. **Continuous Integration:** Regularly merging code changes into a shared repository.
2. **Continuous Testing:** Automatically running tests to ensure code quality.
3. **Continuous Delivery:** Ensuring code is always in a deployable state.
4. **Continuous Deployment:** Automatically deploying code to production.
5. **Continuous Monitoring:** Tracking system performance and issues in real-time.
6. **Continuous Feedback:** Gathering and responding to user and system feedback.
7. **Continuous Operations:** Maintaining system stability and uptime through automated processes.



**45. Explain the concept of Infrastructure as Code (IaC) and discuss the benefits and challenges of implementing IaC in a large-scale production environment.**

[Infrastructure as Code \(IaC\)](#) is the practice of managing and provisioning computing infrastructure through machine-readable definition files, rather than physical hardware configuration. Its benefits include faster deployment, consistency, scalability, and easier management. Challenges may include initial learning curve, complexity in maintaining code, and ensuring security and compliance across diverse environments.

**46. What strategies can be employed to achieve zero-downtime deployments, and how does the Blue/Green Deployment pattern fit into these strategies?**

To achieve zero-downtime deployments, strategies like canary releases and rolling updates are used. [Blue/Green Deployment](#) is a method where you maintain two identical production environments, with only one active at a time. Updates are deployed to the inactive "blue" environment, then traffic is switched to it, ensuring seamless transitions and mitigating downtime.

**47. How do you ensure security and compliance in a CI/CD pipeline, particularly when integrating with multiple cloud providers and third-party services?**

To ensure security and compliance in a CI/CD pipeline with multiple cloud providers and third-party services, implement robust [authentication and authorization mechanisms](#). Utilize encryption for data in transit and at rest, and regularly audit access controls. Employ automated security scanning and testing throughout the pipeline to catch vulnerabilities early. Lastly, maintain clear documentation and communication channels to stay abreast of evolving compliance requirements.

**48. Discuss the importance of monitoring and logging in a DevOps environment. What tools and practices do you recommend for effective observability and incident management?**

[Monitoring and logging](#) in DevOps ensure system health and performance. Tools like Prometheus and Grafana offer real-time insights, while ELK stack provides robust logging. Adopting practices like centralized logging and automated alerting enhances observability and incident response efficiency.

**49. Explain the concept of immutable infrastructure and how it contrasts with traditional infrastructure management. What are the benefits and potential drawbacks of adopting immutable infrastructure in a DevOps workflow?**

Immutable infrastructure is a paradigm where servers and components are never modified after deployment, but instead replaced with updated versions. Unlike traditional methods, where systems are continually altered, immutable infrastructure ensures consistency and reliability.

Benefits include easier deployment, improved scalability, and better fault tolerance. Drawbacks may include initial setup complexity and challenges in managing [stateful applications](#).

**50. Explain the concept of serverless computing and its implications for DevOps practices.**

[Serverless computing](#) is a cloud computing model where the cloud provider dynamically manages the allocation and provisioning of servers. Users only pay for the actual resources consumed by their applications, without worrying about server management.

This model simplifies infrastructure management, allowing developers to focus solely on writing code. For DevOps, serverless reduces the overhead of managing servers, enabling faster development cycles and easier deployment, while emphasizing automation and monitoring for efficient resource utilization.