Share on your Social Media

Top 20 Git Interview Questions and Answers

Published On: June 7, 2024

Git Interview Questions and Answers

Git's significance in DevOps means that career opportunities will expand. To help you ace DevOps interviews, here are the **top 20 Git interview questions and answers** curated by SLA experts.

Download GIT Interview Questions PDF

Git Interview Questions and Answers for Freshers

1. What is Git?

Git is a popular distributed version control system (VCS) that is necessary for tracking changes made to source code as it is being developed for software.

Related Course at SLA

Git Training in OMR

Git Training in Chennai

Related Posts

MEAN Stack Interview Questions and Answers
Published On: June 19, 2024

2. Why Git for DevOps?

It was created by Linus Torvalds in 2005 to help develop the Linux kernel. Git is a fast and efficient project management tool that can handle any size project, whether big or small.

3. What is VCS?

A version control system, or VCS, tracks the contributions made by the developers working together on the projects. They keep track of any code modifications made, allowing developers to add new features, address issues, and conduct tests knowing that their previously functional copy can always be restored if something goes wrong.

4. How can one use git clone?

An existing git repository is copied (or cloned) with this command. Its usual purpose is to obtain a local repository copy from the remote repository.

5. What is a Git repository?

All of the project-based files that Git saves are organized into repositories. Git allows files to be stored in both local and remote repositories.

6. Explain the conflict in Git.

Although Git often manages feature merges automatically, there may occasionally be problems when working in a collaborative setting, such as:

- When two different branches have modified the same line in a file
- A file has been changed in one branch but erased in another.

Git cannot determine which changes should take precedence over others, so these conflicts must be resolved manually following team discussion.

GIT Course Syllabus

Introduction Since MEAN Stack combines several other applications as part of its functionality, it is...

Top 15 Struts Interview Questions and Answers

Published On: June 18, 2024

Struts Interview Questions and Answers When it comes to developing Java web applications, Struts is...

Top 20 C Sharp Interview Questions and Answers

Published On: June 17, 2024

C Sharp Interview Questions and Answers Microsoft created the general-purpose programming language C# together with...

Top 20 VB.Net Interview Questions and Answers

Published On: June 17, 2024

VB.Net Interview Questions and Answers A wide range of applications, including

desktop, web, and mobile...

7. How can you use the git status command?

The git status command is used to track tracked and non-tracked changes and to display the differences between the current directory and the index, which is useful for gaining a thorough grasp of git.

8. What can you do with git ls-tree?

This command yields a tree object representation of the current repository that includes the SHA-1 value of the blob, the mode, and the names of each item.

9. Explain what “index” means.

The developer can format, check, and innovate the files before committing the changes they’ve made. The “Index” or “Staging Area” is the common space where all these activities occur.

10. What makes GitHub and Git different from each other?

Git is a version control system that lets you keep track of and document how your source code changes over time. GitHub is an internet hosting platform that makes managing Git repositories easier.

In addition to features like task organization, permission control, problem tracking, and feature suggestion handling, GitHub also has an easy-to-use online interface.

11. Describe the commit in Git.

A commit in the Git workflow is a procedure that stores a version of the project’s currently prepared changes. This file contains information on the changes made, the name of the developer, the commit timestamp, and a unique identifier (a SHA-1 hash).

12. What does Git branching mean?

With Git, branching enables you to work on other tasks without affecting the main workflow and deviating from the main development path. With this strategy, each activity can be kept separate from the others when developing features, fixing bugs, or conducting experiments in a designated area of the repository.

13. Explain merging in Git.

Integrating changes from one branch into another is known as merging in Git. A three-way merge combines disparate branch histories into a single commit, or a fast-forward merge updates the target branch to the most recent commit of the source branch.

14. What distinguishes `git pull` from `git fetch`?

Without integrating them, `git fetch` retrieves updates to your local repository from a remote repository. `git pull`, on the other hand, pulls the updates into your current branch as well as fetches them.

15. What is a pull request in Git?

One way to contribute to a project is by using a pull request, which is commonly used for projects hosted on GitHub. A developer follows this procedure by making changes in their own branch, uploading them to a repository, and then starting a pull request.

By taking this step, the project's maintainers are prompted to review the suggested modifications, have conversations about potential alterations, and finally merge the pull request into the main branch.

GIT Salary

Git Interview Questions and Answers for Experienced

16. What does the git tag -a command accomplish?

To create an annotated tag in git, use the git tag -a command. Annotated tags have a tagging message and the tagger's name, email address, and date.

Syntax: *git tag -a <tagname> -m "<tagmessage>" [<commit>]*

The base command used to create tags in Git is git tag.

- **-a** denotes annotated, which tells git to make an annotated tag.
- **tagname:** The name you wish to assign to your tag is this.
- **-m** "message": You can describe the tag in a message that you attach to the tag using this optional parameter.

17. Explain Git Worktree.

The command *git-worktree* is used to handle several working trees that are connected to a single repository.

Command:

add <path> [<commit-ish>]: Create a worktree at <path> and checkout <commit-ish> into it.

List: List each worktree's specifics. All of the related worktrees are listed after the main worktree.

move: Transfer a worktree to a different place. This command cannot be used to transfer the main worktree or linked worktrees containing submodules.

remove: Take out a worktree. You can only remove clean worktrees, meaning they don't contain any untracked files or changes to tracked files.

repair: If it is feasible, fix worktree administrative files that have become out-of-date or corrupted as a result of outside influences.

18. Describe how to use git rebase – interactive. How can a commit history be cleaned using it?

With git rebase -i, you can interactively rewrite a set of commits by altering their order, squeezing them together, dividing them, changing the messages they contain, deleting them, and more. Grouping minor commits, rephrasing messages, and removing pointless changes, can tidy up history.

git rebase –interactive can be used in the following ways to tidy up a commit history:

- Compress several minor or recurring commits into one sizable commit to improve the readability and concision of the history.
- To make history easier to understand, edit or reword commit messages that are unclear, poorly phrased, or include spelling or grammar mistakes.

19. How is a git rebase reverted?

A git rebase can be reversed or undone in many different ways:

1. Go back to the original commit of the branch tip:

git reflog – identify the commit hash before the rebase

git reset –hard <commit-hash>

Your prior HEAD commits dating back a few months are displayed in the reflog. Reset back to the commit hash that was made before the rebase. All commits made after the rebase are thus totally removed.

2. New revert commit

git revert -m 1 <rebase-head-commit>

By doing this, all of the modifications made in the rebase head commit are reversed in a new commit. keeps all past commits intact.

20. In Git, what are submodules and how do they function? Give an instance of how you might put them to use.

You can embed external Git repositories at a given path inside a parent Git repository using Git submodules. Key characteristics of submodules include:

- Submodules clone a unique Git repository with its commit history, tags, etc. Like nested repositories, they function.
- The submodule’s repaired commit is recorded in the parent repository. A clear update in the parent is necessary for updating the commits in the submodule.
- helpful for modular project components with independent authors or release cycles.
- The parent repository that uses the submodule may not always automatically reflect changes made to it.

An excellent submodule example is:

- Many major projects use the shared library known as common-utils.
- A distinct development team oversees releases for the utils library at its discretion.
- Common-utils can be embedded into the main application as a submodule located in src/utils.
- The main app repo can regularly absorb new features and fixes from common-utils by upgrading the recorded submodule commit.
- Submodules, in this case, enable modularizing a big codebase into independent parts that can develop independently.

Conclusion

With these **Git interview questions and answers**, we hope to assist you in getting ready for your DevOps interviews. Elevate your DevOps career by enrolling in our **Git training in Chennai**.

GIT Training

Share on your Social Media

Navigation

- About Us
- Blog Posts
- Careers
- Contact
- Placement Training
- Corporate Training
- Hire With Us
- Job Seekers
- SLA’s Recently Placed Students
- Reviews
- Sitemap

Important Links

- Disclaimer
- Privacy Policy
- Terms and Conditions

Softlogic Academy

Softlogic Systems

KK Nagar [Corporate Office]

No.10, PT Rajan Salai, K.K. Nagar, Chennai – 600 078.
Landmark: Karnataka Bank Building
Phone: +91 86818 84318
Email: enquiry@softlogicsys.in
Map: Google Maps Link

OMR

No. E1-A10, RTS Food Street
92, Rajiv Gandhi Salai (OMR),
Navalur, Chennai - 600 130.
Landmark: Adj. to AGS Cinemas
Phone: +91 89256 88858
Email: info@softlogicsys.in
Map: Google Maps Link

Courses

- Python

Social Media Links

Software Testing

Full Stack Developer

Java

Power BI

Clinical SAS

Data Science

Embedded

Cloud Computing

Hardware and Networking

VBA Macros

Mobile App Development

DevOps

Review Sources

Google

Trustpilot

Glassdoor

Mouthshut

Sulekha

Justdial

Ambitionbox

Indeed

Software Suggest

Sitejabber