**Git And GitHub**  
Interview Question and Answer

# Basic

**Q1 What is Git?**

Git is a version control system for tracking changes in computer files and is used to help coordinate work among several people on a project while tracking progress over time. In other words, it’s a tool that facilitates source code management in software development.

Git favors both programmers and non-technical users by keeping track of their project files. It enables multiple users to work together and handles large projects efficiently.

**Q2 What do you understand by the term ‘Version Control System?**

A version control system (VCS) records all the changes made to a file or set of data, so a specific version may be called later if needed.

This helps ensure that all team members are working on the latest version of the file.



**Q3 What is GitHub?**

To provide Internet hosting for version control and software development, GitHub makes use of Git.

### **Q4 Mention some popular Git hosting services.**

### GitHub

### SourceForge

### GitLab

### Bitbucket

**Q5 Different types of version control systems?**

* Local version control systems have a database that stores all file changes under revision control on disc in a special format.
* Centralized version control systems have a single repository, from which each user receives their working copy.
* Distributed version control systems contain multiple repositories, and different users can access each one with their working copy.

### **Q6 What benefits come with using GIT?**

**(i) Free & Open Source:**

Git is issued under GPL’s (General Public License) open source license. You need not pay anything to use Git.

**(ii) Speed:**

As you are not required to connect to any network for executing all the actions, it performs all the tasks quickly. Obtaining version history from a locally stored repository can be one hundred times speedier than obtaining it from the remote server.

Git is written in C, which is the underlying programming language that evades runtime overheads linked with other high-level languages.

**(iii) Scalable:**

Git is highly scalable. So, if the number of collaborators increases in the coming time, then Git can easily accommodate this change.

**(iv) Reliable:**

As every collaborator has its own local repository, on the instances of a system crash, the lost data can be recuperated from any of the local repositories. At all times, you will have a backup of all your files.

**(v) Secure:**

Git utilizes the SHA1 (Secure Hash Function) to name and identify objects inside its repository. Each artifact and commit are check-summed and recovered through its checksum during checkout.

**(vi) Economical:**

In the case of a centralized version control system, the central server must be strong enough to attend requests of the entire team. This is not a problem for smaller teams, however as the team expands, the hardware limitations of the server can be an impediment for performance.

**(vii) Supports Non-linear Development:**

Git provides rapid branching & merging and contains particular tools for envisaging and traversing a non-linear development history. A basic notion in Git is that a change will be merged more frequently than it is written as it is sent across different reviewers.

Git Branches are extremely lightweight. A branch in Git refers only to a single commit. The complete branch structure can be created, with the help of parent commits.

**(viii) Easy Branching:**

Branch management through Git is very straightforward and easy. It requires just a few jiffies to create, delete, and merge branches. Feature branches give an insulated environment to each change to your codebase.

When a developer requires to begin working on something, irrespective of the size of work, they create a new branch. This makes sure that the master branch constantly holds a production-quality code.

**(ix) Distributed Development:**

Git provides every developer a local copy of the whole development history, plus the changes get cloned from one such repository to another. These changes are introduced as added development branches and can be merged in the same manner as a locally developed branch.

**(x) Compatibility along with present Systems or Protocol:**

Repositories can be published through HTTP, FTP or a Git protocol on top of either a plain socket or ssh.

**Q7 What’s the difference between Git and GitHub?**

| **Git** | **GitHub** |
| --- | --- |
| Git is a software | GitHub is a service |
| Git can be installed locally on the system | GitHub is hosted on the web |
| Provides a desktop interface called git GUI | Provides a desktop interface called GitHub Desktop. |
| It does not support user management features | Provides built-in user management |

**Q8 What is a Git repository?**

Git repository refers to a place where all the Git files are stored. These files can either be stored on the local repository or on the remote repository.



**Q9 How can you initialize a repository in Git?**

If you want to initialize an empty repository to a directory in Git, you need to enter the git init command. After this command, a hidden .git folder will appear.

****

**Q10 How is Git different from Subversion (SVN)?**

| **GIT** | **SVN** |
| --- | --- |
| Git is a distributed decentralized version control system | SVN is a centralized version control system. |
| Git stores content in the form of metadata. | SVN stored data in the form of files. |
| The master contains the latest stable release. | In SVN, the trunk directory has the latest stable release |
| The contents of Git are hashed using the SHA-1 hash algorithm. | SVN doesn’t support hashed contents. |

**Q11. Name a few Git commands with their function.**

* Git config - Configure the username and email address
* Git add - Add one or more files to the staging area
* Git diff - View the changes made to the file
* Git init - Initialize an empty Git repository
* Git commit - Commit changes to head but not to the remote repository.

**Q12. What language is used in Git?**

Git is a fast and reliable version control system, and the language that makes this possible is ‘C.’

Using C language reduces the overhead of run times, which are common in high-level languages.

**Q13 What is the correct syntax to add a message to a commit?**

git commit -m "Your Message"

**Q14 Which command is used to create an empty Git repository?**

git init - This command helps to create an empty repository while working on a project.

### **Q15 What does git pull origin master do?**

### The git pull origin master fetches all the changes from the master branch onto the origin and integrates them into the local branch.

### git pull = git fetch + git merge origin/ master.

### **Q16 What are some of the drawbacks of Git?**

* One of the main drawbacks is that it can be difficult to learn and use, especially for those who are not familiar with version control systems.
* And Git is not always reliable and can sometimes be slow.

**Q17 What is a branch in Git?**

A branch is a way to isolate development work on a particular aspect of a project. When a branch is created, it diverges from the primary branch. It allows developers to work on a new feature or bug fix without affecting the main codebase.

**Q18 What is a commit in Git?**

A commit is a way to save changes to a branch. When a commit is made, a snapshot of the current state of the branch is created. This snapshot can be used to revert the branch to that state if necessary.

**Q19 What is conflict in Git?**

Conflict in Git occurs when two or more developers have made changes to the same part of a file, and those changes can't be automatically merged. When this happens, Git will mark the file as conflicted and leave it up to the developers to resolve the conflict.

Resolving a conflict can be done by manually editing the file to choose which changes should be kept, or by using a tool like Git's merge command to automatically merge the changes.

**Q20 What does the git status command do?**

The git status command is used to obtain the current state of a Git repository. This command can be used to determine whether the repository is clean or dirty, and to see which files have been modified. The git status command will also show which branch is currently checked out and whether there are any uncommitted changes.

**Q21 What differentiates between the commands git remote and git clone?**

The main difference between the git remote and git clone commands is that the git remote adds a remote repository as a shortcut to your current repository, while the git clone creates an entirely new copy of a remote repository.

**Q22 Tell me the difference between git pull and git fetch?**

Both of these commands will fetch any new commits from the remote repository, but they differ in how they handle these commits.

Git pull will merge the remote commits into the current branch, while git fetch will simply retrieve the commits and store them in the local repository. This means that if you have any uncommitted changes, git pull may result in merge conflicts, while git fetch will not.

**Q23 Is Git and GitHub the same thing?**

No, Git and GitHub are two different things.

* Git is a version control system that lets you track changes to your code.
* GitHub is a hosting service for Git repositories. You can use GitHub to store your code remotely, or you can use it to collaborate with other developers on a project.

**Q24 What does git add command do?**

* **This command adds files and changes to the index of the existing directory.**
* **You can add all changes at once using git add . command.**
* **You can add files one by one specifically using git add <file\_name> command.**
* **You can add contents of a particular folder by using git add /<folder\_name>/ command.**

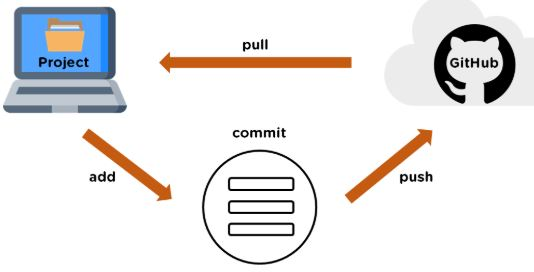
**Q25 What is the command used to delete a branch?**

* **To delete a branch we can simply use the command git branch –d [head].**
* **To delete a branch locally, we can simply run the command: git branch -d <local\_branch\_name>**
* **To delete a branch remotely, run the command: git push origin --delete <remote\_branch\_name>**
* **Deleting a branching scenario occurs for multiple reasons. One such reason is to get rid of the feature branches once it has been merged into the development branch**

intermediate GIT interview questions and answers.

**Q1 What does the git push command do?**

The Git push command is used to push the content in a local repository to a remote repository. After a local repository has been modified, a push is executed to share the modifications with remote team members.

****

**Q2 Difference between git fetch and git pull?**

| **Git Fetch** | **Git Pull** |
| --- | --- |
| **The Git fetch command only downloads new data from a remote repository.** | **Git pull updates the current HEAD branch with the latest changes from the remote server.** |
| **It does not integrate any of these new data into your working files.** | **Downloads new data and integrate it with the current working files.** |
| **Command - git fetch origin**  **git fetch --all** | **Tries to merge remote changes with your local ones.**  **Command - git pull origin master** |

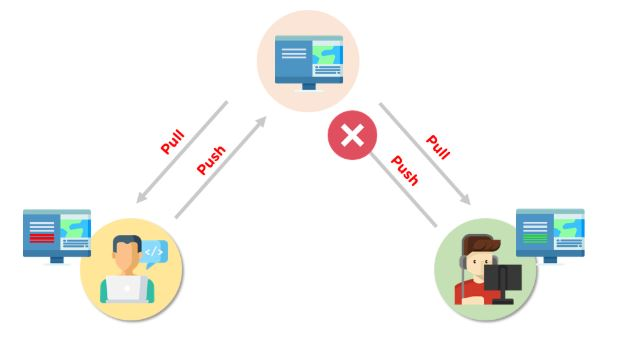
**Q3 GitHub, GitLab and Bitbucket are examples of git repository \_\_\_\_\_\_\_ function?**

hosting. All the three are services for hosting Git repositories.

**Q4 What do you understand about the Git merge conflict?**

**A Git merge conflict** is an event that occurs when Git is unable to resolve the differences in code between the two commits automatically.

Git is capable of automatically merging the changes only if the commits are on different lines or branches.

****

**Q5 How do you resolve conflicts in Git?**

Here are the steps that will help you resolve conflicts in Git:

* Identify the files responsible for the conflicts.
* Implement the desired changes to the files
* Add the files using the git add command.
* The last step is to commit the changes in the file with the help of the git commit command.

### **Q6 What is the functionality of git ls-tree?**

### The git ls-tree command is used to list the contents of a tree object.

### **Q7 What is the process to revert a commit that has already been pushed and made public?**

There are two processes through which you can revert a commit:

**1.** Remove or fix the bad file in a new commit and push it to the remote repository. Then commit it to the remote repository using:

git commit –m “commit message”

**2.** Create a new commit to undo all the changes that were made in the bad commit. Use the following command:

git revert <commit id>

### **Q8 How is a bare repository different from the standard way of initializing a Git repository?**

| **Standard way** | **Bare way** |
| --- | --- |
| You create a working directory with the git init command. | Does not contain any working or checked out copy of source files. |
| A .git subfolder is created with all the git-related change history. | Bare repositories store git revision history in the root folder of your repository instead of the .git subfolder. |

**Q9 What does git clone do?**

Git clone allows you to create a local copy of the remote GitHub repository. Once you clone a repo, you can make edits locally in your system rather than directly in the source files of the remote repo.

**Q10 What is Git stash?**

Let’s say you're a developer and you want to switch branches to work on something else. The issue is you don’t want to make commits in uncompleted work, so you just want to get back to this point later. The solution here is the Git stash.

Git stash takes your modified tracked files and saves it on a stack of unfinished changes that you can reapply at any time. To go back to the work you can use the stash pop.

**Q11 What does the git reset --mixed and git merge --abort commands do?**

* git reset --mixed command is used for undoing changes of the working directory and the git index.
* git merge --abort command is used for stopping the merge process and returning back to the state before the merging occurred.

**Q12 What do you understand about the Staging area in Git?**

The Staging Area in Git is when it starts to track and save the changes that occur in files. These saved changes reflect in the .git directory. Staging is an intermediate area that helps to format and review commits before their completion.

**Q13 What is Git Bisect and how do you use it?**

The Git Bisect command performs a binary search to detect the commit which introduced a bug or regression in the project’s history.

Syntax: git bisect <subcommand> <options>

**Q14 How do you find a list of files that has been changed in a particular commit?**

The command to get a list of files that has been changed in a particular commit is:

git diff-tree –r {commit hash}

* -r flag allows the command to list individual files
* commit hash lists all the files that were changed or added in the commit.

**Q15 What is the use of the git config command?**

The git config command is used to set git configuration values on a global or local level. It alters the configuration options in your git installation. It is generally used to set your Git email, editor, and any aliases you want to use with the git command.

### **Q16. What is the functionality of git clean command?**

### The git clean command removes the untracked files from the working directory.

**Q17. What is SubGit and why is it used?**

SubGit is a tool that is used to migrate SVN to Git. It transforms the SVN repositories to Git and allows you to work on both systems concurrently. It auto-syncs the SVN with Git.

**Q18. If you recover a deleted branch, what work is restored?**

The files that were stashed and saved in the stashed index can be recovered. The files that were untracked will be lost. Hence, it's always a good idea to stage and commit your work or stash them.

### **Q19 Explain these commands one by one– git status, git log, git diff, git revert <commit>, git reset <file>.**

### Git status - It shows the current status of the working directory and the staging area.

### Git revert<commit> - It is used for undoing changes to a repository's commit history.

### Git log- It is a key tool for reviewing and reading the history of everything that happens to a repository.

### Git diff- It is a multi-purpose Git command that performs a diff function on Git data sources when executed.

### Git reset<file>- it is used to unstage a file.

**Q20 What exactly is tagging in Git?**

Tagging enables developers to mark all significant checkpoints as their projects progress.

**Q21 What exactly is forking in Git?**

It is a repository duplicate and forking allows one to experiment with changes without being concerned about the original project.

**Q22 How to change any older commit messages?**

You can change the most recent commit message with the git commit —amend command.

**Q23 How to handle huge binary files in Git?**

Git LFS is a Git extension for dealing with large and binary files in a separate Git repository.

**Q24 Name a few GIT tools?**

Git comes with a few built-in tools like Git Bash and Git GUI.

**Q25 Will you make a new commit or amend an existing one?**

The git commit —amend command allows you to easily modify the most recent commit.

**Q26 What do you mean by branching strategy?**

It is employed by a software development team while writing and managing code with a version control system.

**Q27 Difference between head, working tree, and index?**

They are all names for various branches. Even Though a single git repository can track an arbitrary number of branches, the working tree is only associated with one of them, and HEAD points to that branch.

**Q28 Is there a git GUI client available for Linux?**

Git includes built-in GUI tools for committing (git-gui) and browsing (gitk), but there are a number of third-party tools available for users seeking platform-specific experience.

**Q29 What is the benefit of a version control system?**

Version control enables software teams to maintain efficiency and agility while the team grows by adding more developers.

**Q30 What do you mean by git instaweb?**

It is a script used to set up a temporary instance of Gitweb.

**Q31 What exactly is the forking workflow?**

Forking is a git clone operation that is performed on a server copy of a project's repository.

**Q32 Mention benefits of forking workflow.**

Contributions can be integrated without everyone trying to push to a single central repository.

**Q33 What is the Gitflow workflow?**

The Gitflow Workflow specifies a strict branching model centered on the project release.

**Q34 What does the commit object contain?**

The commit object contains a tree of blob objects and other tree objects that represent the project revision's directory structure.

* A set of files that represents the state of a project at a given point in time.
* Reference to parent commit objects.
* A 40 character string termed as SHA-1 name uniquely identifies the commit object.

**Q35 Write the syntax of rebasing in git.**

**Syntax is as follows: $git rebase <branch name>**

**Q36 What are Git Hooks?**

They are scripts that are executed automatically whenever a specific event occurs in a Git repository.

**Q37 What is Git stash vs Git stash pop?**

Git stash pop removes the (topmost, by default) stash when applied, whereas git stash apply keeps it in the stash list for future use.

**Q38 Explain git reflog**

### This command is used by Git to record changes made to the branches' tips.

**Q39 Role of the git annotate command.**

In git, it is used to track each line of the file based on the commit information.

**Q40 What is a git Directory?**

It is the storage place of the metadata and object database of the project.

**Q41 How can a conflict be settled in Git?**

Edit the files to resolve any incompatible changes first, then use "git add" to add the corrected files and "git commit" to save the repaired merge.

**Q42 What is the standard method for branching in GIT?**

In GIT, the best way to create a branch is to have one'main' branch and then another branch for implementing the changes that we want to make.

### **Q43 How do you set up a Git repository?**

### If you want to add an empty repository to a directory in Git, use the git init command.

### **Q44 What is the proper syntax for appending a message to a commit?**

### Git commit -m "x files created" is the syntax.

### **Q45 Use of git instaweb.**

### It is used to launch a web browser and a webserver with an interface into a local repository automatically.

**Q46 Describe git is-tree.**

It represents a tree object with each item's mode and name included.

**Q47 What exactly is git cherry-pick?**

A command typically used to move specific commits from one branch of a repository to another.

**Q48 State the difference between “git remote” and “got clone”?**

“Git remote” allows you to create an entry in the git configuration which specify a URL.

“Git clone” lets you create a new git repository by letting you copy it from the current URL.

**Q49 Difference between “pull request” and “branch”?**

“Pull request” is done when you feel like changing the developer’s change to another person's code branch. And “Branch” is just a separate version of code.

### **Q50 How might you recover a branch that has previously pushed changes in the main repository yet has been coincidentally erased from each team member's local machines?**

### We can easily recover this by seeing the latest commit of the branch in the reflog and then going through the new branch.

### **Q51 What is a detached head?**

### Detach head refers to that the currently checked repository is not in the local branch.

### **Q52 What command helps us to know the branches merged into master and which are not?**

### git branch - -merged lets us get the lost of the branches which are currently merged into the current branch

### git branch - - no- merged shows the branches which are not merged

### **Q53 Is LDAP Authentication Supported?**

### GitLab API only supports LDAP authentication since version 6.0 and higher.

**Q54 Explain the different points when a merge can enter a conflicted stage.**

There are two stages when a merge can enter a conflicted stage.

**1. Starting the merge process**

If there are changes in the working directory of the stage area in the current project, the merge will fail to start. In this case, conflicts happen due to pending changes that need to be stabilized using different Git commands.

**2. During the merge process**

The failure during the merge process indicates that there’s a conflict between the local branch and the branch being merged. In this case, Git resolves as much as possible, but some things have to be fixed manually in the conflicted files.

**Q55 What has to be run to squash the last N commits into a single commit?**

In Git, squashing commits means combining two or more commits into one.

Use the below command to write a new commit message from the beginning.

git reset -soft HEAD~N &&git commit

But, if you want to edit a new commit message and add the existing commit messages, then you must extract the messages and pass them to Git commit.

The below command will help you achieve this:

git reset -soft HEAD~N &&git commit -edit -m“$(git log -format=%B -reverse .HEAD@{N})”

**Q56 What is the command used to fix a broken commit?**

To fix a broken commit in Git, you may use the “git commit --amend” command, which helps you combine the staged changes with the previous commits instead of creating an entirely new commit.

**Q57 How can you discover if a branch has already been merged or not?**

There are two commands to determine these two different things.

* git branch --merged helps to get the list of the branches that have been merged into the current branch.
* Note: git branch --no-merged lists the branches that have not been merged to the current branch.

**Q58 How do you find a commit which broke something after a merge operation?**

* This can be a time-consuming process if we are not sure what to look at exactly. Fortunately, git provides a great search facility that works on the principle of binary search as git-bisect command.
* The initial set up is as follows:

git bisect start # initiates bisecting session

git bisect bad # marks current revision as bad

git bisect good revision # marks last known commit as good revision

* Upon running the above commands, git checks out a revision that is labeled as halfway between “good” and “bad” versions. This step can be run again by marking the commit as “good” or “bad” and the process continues until the commit which has a bug is found.

**Q59 What happens if the .git directory gets deleted?**

If the .git/ directory gets deleted, then you will lose track of your project’s history. The repository will no longer be under version control.

**Q60 How will you remove a file from Git without actually removing it from your local filesystem?**

You can use the ‘cached’ option for this:

git rm -rf –cached $FILES

This command will remove the files from your repository without deleting them from your disk.