1.What is Git and why is it used?

Git is a DevOps tool used for source code management. It is a free and open-source version control system used to handle small to very large projects efficiently. Git is used to tracking changes in the source code, enabling multiple developers to work together on non-linear development.

2.Explain the difference between Git pull and Git fetch.

Git Pull command is used to fetch all changes from the remote repository to the current working directory. It automatically try to merge or rebase them into our current working directory. It is the combination of git fetch and git merge or git rebase. It can generate merge conflicts if there are conflict changes between our local and remote branches.

Git Fetch command is used to fetch all changes from the remote repository to the local repository. It doesn't make any changes to the current working directory. It stores all the changes in a separate branch called the remote-tracking branch.

3.How do you revert a commit in Git?

To revert a commit in Git, you have a few options depending on your needs. The two common methods are using `git revert` or `git reset`

4.Describe the Git staging area.

The staging area is a file, generally contained in your Git directory, that stores information about what will go into your next commit. Its technical name in Git parlance is the "index", but the phrase "staging area" works just as well.

5.What is a merge conflict, and how can it be resolved?

A git merge conflict is an event that takes place when Git is unable to automatically resolve differences in code between two commits. Git can merge the changes automatically only if the commits are on different lines or branches.

There are a few steps that could reduce the steps needed to resolve merge conflicts in Git.

Step 1: The easiest way to resolve a conflicted file is to open it and make any necessary changes.

Step 2: After editing the file, we can use the git add a command to stage the new merged content.

Step 3: The final step is to create a new commit with the help of the git commit command.
Step 4: Git will create a new merge commit to finalize the merge.

6.How does Git branching contribute to collaboration?

Git branching is a powerful feature that significantly contributes to collaboration in software development. It enables multiple developers to work on different features, bug fixes, or improvements simultaneously without interfering with each other's work.

7. What is the purpose of Git rebase?

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.

8. Explain the difference between Git clone and Git fork.

Any public Git repository can be forked or cloned. A fork creates a completely independent copy of Git repository. In contrast to a fork, a Git clone creates a linked copy that will continue to synchronize with the target repository.

9.How do you delete a branch in Git?

To issue the command to delete a local Git branch, follow these steps:

1. Open a Git BASH or a command prompt in the root of your Git repository.
2. If necessary, use the git switch or checkout command to move off the branch you wish to delete.
3. Issue the following command:
   git branch --delete <branchname>
4. Run the git branch -a command to verify the local Git branch is deleted.

10. What is a Git hook, and how can it be used?

Git hooks are scripts that run automatically every time a particular event occurs in a Git repository. They let you customize Git's internal behavior and trigger customizable actions at key points in the development life cycle.