NAME - Patricia Stanley

GIT AND GITHUB TEST

1) What is Git and why is it used?

Ans: Git is a version control system that means it is a method to save the history of your files, and also the changes made in it. It is an open source software.

The uses of Git are:

- Git is open-source and freely available, making it accessible to developers and teams of all sizes.
- It helps to collaborate among multiple developers working on the same project.
- It is designed to be fast and efficient, making it suitable for both small and large-scale projects.

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

Ans: Both git pull and git fetch are essential commands for keeping your local repository up-to-date with a remote repository. However, they have differences:

Git pull

- git pull updates and merges the changes directly into the current branch.
- git pull is more automatic and straightforward but can lead to conflicts if there are incompatible changes.
- git pull can be risky if you are unaware of the changes, as it merges them immediately, potentially causing merge conflicts.

Git fetch

- git fetch only updates the local copy of the remote branch, allowing you to review changes before merging.
- git fetch gives you more control over when and how to merge the changes into your local branches.
- git fetch is safer for checking the status of the remote repository and understanding incoming changes before affecting your local branch.

3) How do you revert a commit in Git?

Ans: revert is the command we use when we want to take a previous commit and add it as a new commit, keeping the log intact.

- Use the Git log or reflog command to find the ID of the commit you want to revert.
- Enter the Git revert command , including the commit ID you want to work on.
- Provide an informative Git commit message to explain why you needed to perform the Git revert.

4) Describe the Git staging area.

Ans: The Git staging can be performed by the following:

- ★ When we make any modifications like add, delete or modify in the working directory, these are not tracked by Git initially until you stage them.
- ★ You add changes to the staging area using the git add command. This tells Git that you want to include these specific changes in your next commit.
- ★ We can check what is in the staging area using git status
- ★ Finally, you use git commit to move the changes from the staging area to the repository. This creates a new commit with the staged changes.

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

Ans: 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.

Steps to resolve merge conflict:

- ☐ The easiest way to resolve a conflicted file is to open it and make any necessary changes.
- ☐ After editing the file, we can use the git add a command to stage the new merged content.
- ☐ The final step is to create a new commit with the help of the git commit command.
- ☐ Git will create a new merge commit to finalize the merge.

6) How does Git branching contribute to collaboration?

Ans: Git branching can help with collaboration by allowing multiple developers to work on different parts of a project at the same time without interfering with each other. This can be especially useful in larger projects where multiple features are being developed simultaneously.

7) What is the purpose of Git rebase?

Ans: Git rebase is a powerful tool for maintaining a clean and organized commit history, but it should be used carefully, especially when working with shared branches, to avoid rewriting commit history that others rely on.

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

Ans: Git clone and Git fork are both used to create copies of a repository, but they serve different purposes and are used in different contexts.

Git Clone	Git Fork
1) Create a local copy of a repository.	1) Create a copy of a repository under your own Git terminal
2) Typically used when you want to work on an existing project without modifying the original repository.	2) Commonly used in open-source projects to propose changes or improvements without affecting the original repository.
3) Downloads the entire repository, including its history, to your local machine.	3) Creates a new repository on your that is a duplicate of the original. You can then clone this forked repository to your local machine.

9) How do you delete a branch in Git?

Ans: To delete both a local and remote Git branch, even if the Git branch has the same name locally and remotely, two commands can be used:

1. A git push origin delete command deletes the remote Git branch.
2. A git branch delete command deletes the local Git branch.

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

Ans: A Git hook is a script that runs automatically when a specific Git event occurs, such as a commit, push, or merge. These hooks are stored in the .git/hooks directory of your Git repository.

- ➢ Git hooks are stored in the .git/hooks directory of a repository.
- ➢ Each hook is a script with a specific name corresponding to the event it hooks into (e.g., pre-commit, post-commit).
- ➢ These scripts can be written in any scripting language.