Github Interview Questions

1. **Git** – Git is a distributed version control system used to track changes in source-code during software development. It allows multiple developers to work on a project simultaneously, track their changes and help in merging the work efficiently.

2. What is Version Control System?
→ A VCS is a s/w that keep track of changes in a file/set of files over time so that you can recall any specific version later.
→ It allows..... ↑ ... work efficiently.
→ Developer can compare the earlier version of the code with the old version to fix the mistake.

3. Git vs Github?
→ Git is a VCS used for tracking & managing changes in a ~~file~~ code.
→ Github is a cloud-based hosting service for Git-Repositories. It provide additional features such as collaboration tools, pull requests and more.

4. Repositary
→ Repositary is a directory or storage-space where your project resides. It contains all the project files & the history of changes make to those files.

5. Types of Repositary
→ 3 types →
1. Local Repo. → Exist on local machine, where you make changes.
2. Remote Repo. → A repo. hosted on a server like Github, enabling collaborations.
3. Bare Repo.

6. What are the stages in Git WorkFlow?
→ 1. Working Directory
2. Staging Area
3. Repositary.

Git Commands

1. Git Configuration
   (a) git config --global user.name "NR";
   (b) git config --global user.email "xyz@gmail.com";

2. Initialize git
   (a) git init → initializes new/existing directory as a git repo.
   (b) git clone URL → used to create a local copy of the remote repo.

3. Git status → shows current state of working directory and staging area
   It displays : ① Changes that are staged for next commits.
   ② Modifications that are not yet staged.
   ③ Untracked files.

4. Git Add → working directory to → staging Area
   (a) git add .
   (b) git add --all
   (c) git add -A

   • Unstage files while retaining the changes in working directory
   → git reset filename

5. Git commit - Creates a new snapshot of the ~~local~~ changes in local repo.
   (a) git commit -m "commit msg";
   or
   (b) git commit -am "commit msg";

6. Git log → view history of git commits.

7. .gitIgnore file → It is a feature provided by the git VCS to make git ignore such files which user do not want to be tracked by git.

8. Git Branch →
   (a) In git, a branch is the new/seperate version of the main repository.
   (b) It allows you to work on a different feature, independently from the main ~~repository~~ project.

(a) To create a new branch
→ git branch branchName;

(b) To delete a branch
→ git branch -d branchName;

(c) To switch to a branch
→ git checkout branchName;

(d) To merge branch
→ git merge branchName;

(e) To list all the branches
git branch;

9. Undo Commit

① git reset --soft sha-id;

② git push -f;

or

git revert sha-id;

10. Tagging →

11. Git stash →

(a) git stash

(b) git pull

(c) git stash apply

(d) git push