**What is GIT?**

GIT is a distributed version control system and source code management (SCM) system with an emphasis to handle small and large projects with speed and efficiency.

**What is a repository in GIT?**

A repository contains a directory named .git, where git keeps all of its metadata for the repository. The content of the .git directory are private to git.

**What is the command you can use to write a commit message?**

The command that is used to write a commit message is “git commit –a”.  The –a on the command line instructs git to commit the new content of all tracked files that have been modified. You can use “git add<file>” before git commit –a if new files need to be committed for the first time.

**What is the difference between GIT and SVN?**

The difference between GIT and SVN is

a)      Git is less preferred for handling extremely large files or frequently changing binary files while SVN can handle multiple projects stored in the same repository.

b)      GIT does not support ‘commits’ across multiple branches or tags.  Subversion allows the creation of folders at any location in the repository layout.

c)        Gits are unchangeable, while Subversion allows committers to treat a tag as a branch and to create multiple revisions under a tag root.

**What are the advantages of using GIT?**

a)      Data redundancy and replication

b)      High availability

c)       Only one.git directory per repository

d)      Superior disk utilization and network performance

e)      Collaboration friendly

f)       Any sort of projects can use GIT

**What language is used in GIT?**

GIT is fast, and ‘C’ language makes this possible by reducing the overhead of runtimes associated with higher languages.

**What is the function of ‘GIT PUSH’ in GIT?**

‘GIT PUSH’ updates remote refs along with associated objects.

**Why GIT better than Subversion?**

GIT is an open source version control system; it will allow you to run ‘versions’ of a project, which show the changes that were made to the code overtime also it allows you keep the backtrack if necessary and undo those changes.  Multiple developers can checkout, and upload changes and each change can then be attributed to a specific developer.

**What is “Staging Area” or “Index” in GIT?**

Before completing the commits, it can be formatted and reviewed in an intermediate area known as ‘Staging Area’ or ‘Index’.

**What is GIT stash?**

GIT stash takes the current state of the working directory and index and puts in on the stack for later and gives you back a clean working directory.  So in case if you are in the middle of something and need to jump over to the other job, and at the same time you don’t want to lose your current edits then you can use GIT stash.

**What is GIT stash drop?**

When you are done with the stashed item or want to remove it from the list, run the git ‘stash drop’ command.  It will remove the last added stash item by default, and it can also remove a specific item if you include as an argument.

**How will you know in GIT if a branch has been already merged into master?**

Git branch—merged lists the branches that have been merged into the current branch

Git branch—-no merged lists the branches that have not been merged

**What is the function of git clone?**

The git clone command creates a copy of an existing Git repository.  To get the copy of a central repository, ‘cloning’  is the most common way used by programmers.

**What is the function of ‘git config’?**

The ‘git config’ command is a convenient way to set configuration options for your Git installation.  Behaviour of a repository, user info, preferences etc. can be defined through this command.

**What does commit object contain?**

a)      A set of files, representing the state of a project at a given point of time

b)      Reference to parent commit objects

c)       An SHAI name, a 40 character string that uniquely identifies the commit object.

**How can you create a repository in Git?**

In Git, to create a repository, create a directory for the project if it does not exist, and then run command “git init”. By running this command .git directory will be created in the project directory, the directory does not need to be empty.

**What is ‘head’ in git and how many heads can be created in a repository?**

A ‘head’ is simply a reference to a commit object. In every repository, there is a default head referred as “Master”.  A repository can contain any number of heads.

**What is the purpose of branching in GIT?**

The purpose of branching in GIT is that you can create your own branch and jump between those branches. It will allow you to go to your previous work keeping your recent work intact.

**What is the common branching pattern in GIT?**

The common way of creating branch in GIT is to maintain one as “Main“

branch and create another branch to implement new features. This pattern is particularly useful when there are multiple developers working on a single project.

**How can you bring a new feature in the main branch?**

To bring a new feature in the main branch, you can use a command “git merge” or “git pull command”.

**What is a ‘conflict’ in git?**

A ‘conflict’ arises when the commit that has to be merged has some change in one place, and the current commit also has a change at the same place. Git will not be able to predict which change should take precedence.

**How can conflict in git resolved?**

To resolve the conflict in git, edit the files to fix the conflicting changes and then add the resolved files by running “git add” after that to commit the repaired merge,  run “git commit”.  Git remembers that you are in the middle of a merger, so it sets the parents of the commit correctly.

**To delete a branch what is the command that is used?**

Once your development branch is merged into the main branch, you don’t need

development branch.  To delete a branch use, the command “git branch –d [head]”.

**What is another option for merging in git?**

“Rebasing” is an alternative to merging in git.

**What is the syntax for “Rebasing” in Git?**

The syntax used for rebase is “git rebase [new-commit] “

**What is the difference between ‘git remote’ and ‘git clone’?**

‘git remote add’  just creates an entry in your git config that specifies a name for a particular URL.  While, ‘git clone’ creates a new git repository by copying and existing one located at the URI.

**What is GIT version control?**

With the help of GIT version control, you can track the history of a collection of files and includes the functionality to revert the collection of files to another version.  Each version captures a snapshot of the file system at a certain point of time. A collection of files and their complete history are stored in a repository.

**Mention some of the best graphical GIT client for LINUX?**

Some of the best GIT client for LINUX is

a)      Git Cola

b)      Git-g

c)       Smart git

d)      Giggle

e)      Git GUI

f)       qGit

**What is Subgit? Why to use Subgit?**

‘Subgit’ is a tool for a smooth, stress-free SVN to Git migration.  Subgit is a solution for a company -wide migration from SVN to Git that is:

a)      It is much better than git-svn

b)      No requirement to change the infrastructure that is already placed

c)       Allows to use all git and all sub-version features

d)      Provides genuine stress –free migration experience.

**What is the function of ‘git diff ’ in git?**

‘git diff ’ shows the changes between commits, commit and working tree etc.

**What is ‘git status’ is used for?**

As ‘Git Status’ shows you the difference between the working directory and the index, it is helpful in understanding a git more comprehensively.

**What is the difference between the ‘git diff ’and ‘git status’?**

‘git diff’ is similar to ‘git status’, but it shows the differences between various commits and also between the working directory and index.

**What is the function of ‘git checkout’ in git?**

A ‘git checkout’ command is used to update directories or specific files in your working tree with those from another branch without merging it in the whole branch

**What is the function of ‘git rm’?**

To remove the file from the staging area and also off your disk ‘git rm’ is used.

**What is the function of ‘git stash apply’?**

When you want to continue working where you have left your work, ‘git stash apply’ command is used to bring back the saved changes onto the working directory.

**What is the use of ‘git log’?**

To find specific commits in your project history- by author, date, content or history ‘git log’ is used.

**What is ‘git add’ is used for?**

‘git add’ adds file changes in your existing directory to your index.

**What is the function of ‘git reset’?**

The function of ‘Git Reset’ is to reset your index as well as the working directory to the state of your last commit.

**What is git Is-tree?**

‘git Is-tree’ represents a tree object including the mode and the name of each item and the SHA-1 value of the blob or the tree.

**Explain what is commit message?**

Commit message is a feature of git which appears when you commit a change. Git provides you a text editor where you can enter the modifications made in commits.

**How can you fix a broken commit?**

To fix any broken commit, you will use the command “git commit—amend”. By running this command, you can fix the broken commit message in the editor

**Why is it advisable to create an additional commit rather than amending an existing commit?**

There are couple of reason

a)      The amend operation will destroy the state that was previously saved in a commit.  If it’s just the commit message being changed then that’s not an issue.  But if the contents are being amended then chances of eliminating something important remains more.

b)      Abusing “git commit- amend” can cause a small commit to grow and acquire unrelated changes.

**What is ‘bare repository’ in GIT?**

To co-ordinate with the distributed development and developers team, especially when you are working on a project from multiple computers ‘Bare Repository’ is used. A bare repository comprises of a version history of your code.

**Name a few Git repository hosting services**

* Pikacode
* Visual Studio Online
* GitHub
* GitEnterprise
* SourceForge.net

**What Is Github?**

**Answer :**

GitHub is industry-standard version control and publishing platform for web developers. GitHub is the place where developers store the code. It is the best place to share code with friends, co-workers, classmates, and total strangers. For understanding GitHub, we should understand about “Git.” “Git” is version control system, which means that whenever the developer creates something and make changes to the code or release new versions, anyone can keep track of all the modifications in a central repository.

**How Does Github Help In Collaborating With Other Developers?**

**Answer :**

Whenever any programmer uploads their code to Git, other developers can access and edit it, isolated from each other and with all versions kept intact. Git takes away any confusion with having multiple people working on the same file at the same time. There are no chances of your hard work.

**How To Use Github?**

**Answer :**

**The use of GitHub can be explained in the following steps:**

1. Install git and create a GitHub account.
2. Create a local git repository.
3. Add a new file to the repo.
4. Add a file to the staging environment.
5. Create a commit.
6. Create a new branch.

**What Is Github Link?**

**Answer :**

GitHub is a web-based Git or version control repository on the Internet with a hosting service. It offers the distributed version control and source code management (SCM) functionality of Git as well as adding its own new features.

**Is Github Open Source Software?**

**Answer :**

GitHub is not open-source software because they have a commercial offer based known as "GitHub for Enterprise.” Nevertheless, anyone can apply their ideas just like open-source software on GitHub look alike, known as GitLab which is a ruby application with its source code here.

**What Is A Branch On Github?**

**Answer :**

Branch in Git means a lightweight movable pointer to one of these commits. The default branch name in Git is the master. As we initially make commits, we are given a master branch that points to the last commit being made. Every time we commit, it moves forward automatically.

**What Is A Git Pull?**

**Answer :**

Git pull is shorthand for Git fetch, followed by Git merge FETCH\_HEAD. More specifically, Git pull runs Git fetch with the given parameters and calls Git merge to merge the retrieved branch heads into the current branch. Should be the name of a remote repository as forwarded to Git-fetch.

**What Is A Git Pull Request?**

**Answer :**

Pull requests let us inform others about the changes that have been pushed to a GitHub repository. Once a pull request is sent, interested developers can review the set of changes, discuss potential modifications, and even push follow-up commits if necessary.

**What Is Github Handle?**

**Answer :**

GitHub handle is the username, which is displayed, on the top right corner in GitHub. This is the one we use to login to GitHub when we enter the site and commit to HTTPS and the one that appears in the URLs of our GitHub repositories.

**What Is Repo?**

**Answer :**

GitHub users shorten this to “repo.” It can be local to a folder on the computer, or it can be a storage space on GitHub or another online host. We can keep code files, text files, image files; our name it, inside a repository.

**How Can We Create A Branch In Github?**

**Answer :**

GitHub has a simple UI for creating branches. Once opened the branch drop-down and it prompts "Find or create a branch.” We need to Type the name of your new branch, then click the "create" button that appears. For retrieving the new branch from GitHub, we can use the standard Git fetch command.

**What Is A Github Fork?**

**Answer :**

Fork is a copy of a repository. Forking a repository allows us to freely experiment with the changes without affecting the original project. Most commonly, forks are used to propose changes to someone else's project or to use someone else's project as a starting point of our own idea.

**Can We Change Our Github Username?**

**Answer :**

We can change our GitHub account name at any time. For this, we need to click the profile picture > Settings > Account Settings > Change Username. Links to the repositories will redirect to the new URLs, but they should be updated on all other sites because someone who chooses the abandoned username can override the links.

**How Much Space Do We Get On Github?**

**Answer :**

We get a space of 1 GB but if it exceeds 1GB, we receive a polite email from GitHub Support requesting to reduce the size of the repository and scale it down. In addition, we place a strict limit of files exceeding 100 MB in size.

**Can We Delete The Github Account?**

**Answer :**

We can delete the GitHub user account any time. Before we do so, we should hand over the reins of any organizations we might own. Deleting the user account removes all repositories, forks of private repositories, wikis, issues, pull requests, and pages owned by the account.

**What Is The Difference Between Bit Bucket And Github?**

**Answer :**

Bit bucket and GitHub are very similar in terms of features. If the VCS is Mercurial, then Bit bucket is for us. Each has its own features not found in the other, such as GitHub pages for small web-hosting projects, or Bit bucket’s JIRA integration.

**What Is A Commit?**

**Answer :**

Commit is a record of the files that have been modified, since the last time we made a commit. We make changes to our repo (for example, adding a file or modifying one) and then tell Git to put those files into a commit. Commits make up the core of our project and allow us to go back to the state of a project at any point.

**What Is A Star In Github?**

**Answer :**

Star in GitHub is equivalent to “Like” button on Facebook. A voting system, which enables developers to vouch for projects they think are excellent.

**On this collaboration front, GitHub also lets team members set up the following:**

1. **Issues:** These keep track of what collaborators are doing and allow them to ask about bugs. Issues can be opened or closed once they have been rectified.
2. **Milestones:** These pre-determined set goals, which motivate the collaborators to work towards recognition.

**How Can We Create A Gist?**

**Answer :**

**Creating a gist requires a very simple process as depicted in the steps below: -**

1. Sign in to GitHub.
2. We should the navigate to the gist home page.
3. After this, we need to type an optional description and name for the gist.
4. Key in the text of your gist into the gist text box.
5. Following this we should select either to create a public gist or to create a secret gist.

**What Is A Gist Programming?**

**Answer :**

GitHub provides a hosting service that facilitates a web-based Git repository. It includes all the functionality of Git with additional features added in. The gist is an additional attribute added to GitHub, which facilitates the sharing of code snippets, notes, to do lists and more. We can save our Gists as secret or public in the repository.

**What Does It Mean To Star A Repo In Github?**

**Answer :**

Starring a repository allows us to keep track of projects that we find interesting, even if we aren't associated with the project. When we Star a repository, we are actually performing two distinct actions: Creating a bookmark for easier access, apart from motivating fellow developers and rate them.