1. Git
   1. Understand what is GIT

The Git is a tool used for the source code management or version control system, and it is the free source open platform -and the git is used to tracking the changes in the source code or project application code.

* 1. b. Why we need GIT

Git is the most commonly used version control system. Git tracks the changes you make to files, so you have a record of what has been done, and you can revert to specific versions should you ever need to. Git also makes collaboration easier, allowing changes by multiple people to all be merged into one source code.

* 1. How does the GIT work

Git is used to track the changes in the code and manage their project or product with the using of the Git commands like commit, pull, add, push ..etc.

The main part of the Git is the repository used to contain a project folder and it is stored to locally or GitHub.

* 1. What are alternatives to GIT

The alternatives of the Git are:

Free Service: Git, SVN, CVS

Paid service: Bit Bucket, P4, STASH.

* 1. How many companies are available for us to create repositories

Almost 50 companies are there to create repository like

Bitbucket, sub version , git, Project locker, GitLab…etc.

* 1. What is repository

Repository in git is considered as your project folder.

A repository has all the project related data

It contains the collection of the files and also history of changes made to those files.

* 1. What is a branch

A branch is a version of the repository that diverges from the main working project. A Git project can have more than one branch. These branches are a pointer to a snapshot of your changes. When you want to add a new feature or fix a bug, you spawn a new branch to summarize your changes. So, it is complex to merge the unstable code with the main code base and also facilitates you to clean up your future history before merging with the main branch.

The git branch command allows you to create, list, rename and delete branches.

* 1. How do we authenticate the user in GIT

Git provides multiple protocols for authenticating to and interacting with Git repositories. There are three main approaches you can take:

Using a personal authentication token or password

Using an SSH key

Using your GitHub password with 2-factor authentication

* 1. How do we add multiple developers to write code in my the repository

First of all we have to do the clone from the master branch

With the using of the command .Git branch branchName . so in that way different developers in the repository.

* 1. How to resolve merge conflicts

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. It will create a new merge commit to finalize the merge