1. \*\*What Is Git? What Are The Advantages Of Using Git?\*\*

- Git is a distributed version control system that helps manage and track changes in source code during software development. It allows multiple developers to collaborate on a project efficiently. Some advantages of Git include:

- \*\*Distributed Version Control:\*\* Every developer has their own local copy of the entire project history.

- \*\*Branching and Merging:\*\* Easy and efficient handling of branching and merging for parallel development.

- \*\*History Tracking:\*\* Detailed history of changes, making it easy to understand the evolution of the codebase.

- \*\*Staging Area:\*\* Allows selective commits by staging changes before committing.

2. \*\*What Do You Understand By The Term ‘Version Control System’?\*\*

- A Version Control System (VCS) is a software tool that helps manage changes to source code over time. It allows multiple contributors to work on a project simultaneously without conflicts. VCS keeps track of changes, facilitates collaboration, and provides mechanisms for branching, merging, and version history.

3. \*\*What’s The Difference Between Git And GitHub?\*\*

- Git is a version control system (VCS) that manages and tracks changes in source code. GitHub, on the other hand, is a web-based platform that provides hosting for Git repositories. GitHub adds a web-based interface, collaboration features, and additional tools like issue tracking. In essence, Git is the tool, while GitHub is a service built around the Git technology.

4. \*\*Name A Few Git Commands With Their Function.\*\*

- Here are some common Git commands:

- \*\*`git init`:\*\* Initializes a new Git repository.

- \*\*`git clone`:\*\* Creates a copy of a remote repository on the local machine.

- \*\*`git add`:\*\* Adds changes to the staging area.

- \*\*`git commit`:\*\* Records changes to the repository with a message.

- \*\*`git push`:\*\* Uploads local changes to a remote repository.

- \*\*`git pull`:\*\* Fetches changes from a remote repository and merges them into the current branch.

- \*\*`git branch`:\*\* Lists, creates, or deletes branches.

- \*\*`git merge`:\*\* Merges changes from one branch into another.

- \*\*`git status`:\*\* Shows the status of changes as untracked, modified, or staged.

- \*\*`git log`:\*\* Displays the commit history.

5. \*\*Difference Between Git Fetch And Git Pull.\*\*

- \*\*`git fetch`:\*\* Fetches changes from a remote repository to the local repository but does not automatically merge them into the working directory. It updates the remote tracking branches.

- \*\*`git pull`:\*\* Fetches changes from a remote repository and automatically merges them into the current branch. It is essentially a combination of `git fetch` and `git merge`.

In summary, `git fetch` retrieves changes and updates references, but it does not automatically update the working directory. `git pull` fetches changes and immediately integrates them into the working directory.

**Commands in GitHub**

* echo "# my-first-github-repo" >> README.md
* git init
* git add README.md
* git commit -m "first commit"
* git branch -M main
* git remote add origin https://github.com/Ra-ina/my-first-github-repo.git
* git push -u origin main