Git is a popular version control system that allows developers to manage and track changes made to their codebase over time. It was developed by Linus Torvalds in 2005 and has since become widely used in the software development industry.

With Git, developers can create a local repository of their codebase, which can be used to keep track of changes made to the code over time. Git also allows multiple developers to collaborate on a project by merging their changes into a single codebase.

Git uses a branching model, which means that developers can create multiple branches of their codebase to work on different features or bug fixes without affecting the main codebase. Once the changes are complete, they can be merged back into the main codebase.

Git also provides a range of features for tracking changes, comparing different versions of code, and reverting to previous versions if needed.

Overall, Git is a powerful tool that is essential for modern software development, as it enables efficient collaboration, version control, and management of code changes over time.

Difference between git and github for freshers

- ➔ **Git is a version control system that allows developers to manage and track changes made to their codebase over time.**
- ➔ **It is a command-line tool that can be used locally on your computer.**

- ➔ **GitHub, on the other hand, is a web-based hosting service for Git repositories.**
- ➔ **It provides a centralized location where developers can store, collaborate on, and share their code with others.**

➔ **GitHub offers a user-friendly interface for managing Git repositories, as well as additional features such as issue tracking, pull requests, and project management tools.**

In summary, Git is the tool used for version control and managing changes to code locally, while GitHub is a web-based platform that provides additional features for collaborating and sharing code with others.

Step-by-step guide on how to create a Git repository and upload your project: [Demo]

1.**Sign up for a Git account: If you don't have a Git account already, sign up for one at git.com**.

2.Install Git: If Git isn't already installed on your machine, download it and install it.

3.Create a new repository: Log in to Git and create a new repository by clicking on the "New repository" button.

4.Name the repository: Give your repository a name that is descriptive and relevant to your project.

5.Choose the repository's visibility: Decide whether you want your repository to be public or private. Public repositories are visible to everyone, while private repositories require permission to access.

6.Initialize a Git repository on your local machine: Navigate to the directory on your local machine where you want to keep your project files and run the following command:

git init
This will create a new Git repository in the current directory.

Create your project: Create your project files and add them to the directory.

Add your files to the repository: Use the following command to add your files to the repository:
git add .

This will add all files in the current directory to the repository.

Commit your changes: Use the following command to commit your changes to the repository:
git commit -m "Initial commit"
This will commit your changes to the repository and give them a descriptive message.

Connect your local repository to the remote repository:
Use the following command to connect your local repository to the remote repository:
git remote add origin [repository URL]

Replace [repository URL] with the URL of the remote repository you created in step 3.

Push your changes to the remote repository: Use the following command to push your changes to the remote repository:
git push -u origin master
This will push your changes to the remote repository and set the upstream branch to master.

References:
https://www.youtube.com/watch?v=JrCC66R_EhQ

https://youtu.be/crlrz-Ks8DE