1. Concepts of Git explaining various terms

Repository (Repo):

A repository is a central location where all the files and their version history are stored. It can be local (on your machine) or remote (on a server).

Clone:

Cloning is the process of creating a copy of a repository. When you clone a repository, you get all the files, commit history, and branches.

Commit:

A commit is a snapshot of the changes made to files in the repository at a specific point in time. Each commit has a unique identifier (hash) and includes a commit message describing the changes.

Branch:

A branch is an independent line of development. Creating branches allows you to work on new features or bug fixes without affecting the main codebase. Branches can be merged later.

Merge:

Merging is the process of combining changes from one branch into another. It brings the changes made in one branch into another, typically used to incorporate features or bug fixes.

Pull Request (PR):

A pull request is a way to propose changes to a repository. It allows collaborators to review the changes made in a branch and, if approved, merge those changes into the main branch.

2. Basic Commands of GIT

Initialize a Repository: git init

Clone a Repository: git clone <repository_url>

Add Changes: git add <file_name>

Commit Changes: git commit -m "Your message"

Check Status: git status

View Commit History: git log

Pull Changes: git pull origin <branch_name>

Push Changes: git push origin <branch_name>

3.Concepts on GITHUB, GitLab and BitBucket

GitHub:

GitHub is a platform for hosting and collaborating on Git repositories, with features like Issues and Pull Requests for efficient project management and code review.

GitLab:

GitLab is a web-based Git repository manager that provides a comprehensive DevOps platform, including features like Merge Requests, built-in CI/CD, and project organization through groups.

Bitbucket:

Bitbucket is a Git repository hosting service with features like Pull Requests and branch permissions, and it also offers integrated CI/CD pipelines for automated testing and deployment.

4. Industrial Practices of Using Git

Collaboration:

Git enables seamless collaboration, allowing multiple developers to work on a project simultaneously.

Branching Efficiency:

Effective branching strategies, like Gitflow, streamline development stages for managing features and releases.

Code Review Automation:

Pull requests automate code reviews, ensuring code quality through collaborative feedback.

Issue Management:

Built-in issue tracking in Git platforms helps teams efficiently manage tasks, bugs, and enhancements.

Versioned Documentation:

Git repositories serve as versioned documentation, ensuring organized and accessible project information.

5. Cloning a Repo to Local

To clone a repository to our local machine use code

git clone<repositary_url>

6. Resources

- GitLab Documentation
- BitBucket Documentation
- Open AI