**FSD Laboratory 01**

**Name:** Aayush Shah

**Panel:** H

**Roll No.:** 07

**TY. B.Tech CSE**

Aim: Version control with Git.

Objectives:

1. To introduce the concepts and software behind version control, using the example of Git.
2. To understand the use of 'version control' in the context of a coding project.
3. To learn Git version control with Clone, commit to, and push, pull from a git repository.

Theory:

**1. What is Git? What is Version Control?**

**Git** is a specific type of version control system, and it’s the most popular one out there. It’s free, open- source, and very powerful.

**Version Control** is a way to track changes in files over time. This is especially important for things like software development, where you might make a bunch of changes to code, realize it’s broken, and want to go back to an earlier version. Version control systems let you do this easily.

**2. How to use Git for version controlling?**

Using Git for version control involves a workflow of commands typically used in the terminal.

Git provides a powerful workflow for managing code versions. Here’s a professional summary:

**1. Initialization:** Establish a Git repository within your project directory using git init this creates a local database to track changes.

**2. Staging Changes:** Use git add to designate specific files or git add. to include all modified files for the next commit.

**3. Committing Changes:** Execute git commit to create a permanent snapshot of the staged changes. Include a descriptive message summarizing the modifications.

**4. Version History:** Utilize git status to view the current state of your files (modified, staged, committed). Leverage git log to explore the commit history and associated

messages.

**5. Remote Collaboration (Optional):** Integrate a remote repository service like GitHub for

collaborative development and project backups. Use git push to upload your local commits and git pull to retrieve changes from teammates.

FAQ:

**1. What is branching in Git?**

In Git, branching is a fundamental concept that allows you to create temporary isolated workspaces to develop new features or fix bugs without affecting the main codebase.

● **Divergence:** Imagine the main codebase as a highway. Branching lets you create a new road (branch) that diverges from the highway at a specific point. You can work on this new road without impacting traffic on the main highway.

● **Isolated Development:** On your branch, you can make changes, experiment with new ideas, or fix bugs without affecting the stable code in the main branch. This isolation ensures a safe

development environment.

● **Merging:** Once your work on the branch is complete and tested, you can merge it back into the main highway (main branch) to integrate your changes. This allows you to seamlessly incorporate your work into the main codebase.

**2. How to create and merge branches in Git? Write the commands used.**

Creating branches:

● git branch <name>: Creates a new branch.

● git checkout -b <name>: Creates and switches to a new branch.

Merging branches:

● Switch to the target branch (e.g., master).

● git merge <branch\_to\_merge>: Merges the specified branch into the current branch.

Output: Screenshots of the output to be attached.

**Problem Statement:**

Create a public git repository for your team and submit the repo URL as a solution to this assignment, Learn Git concept of Local and Remote Repository, Push, Pull, Merge and Branch.

**Output:**

Public git repository link (Github): <https://github.com/Aayush01055/FSD-Assignments.git>

**Git Commands:**

**1. Creating Local Repository:**

Forcreating local repositorygit init is used

A screenshot of a computer

Description automatically generated

**2. Adding into remote repository:**

To add a new remote, use the git remote add command on the terminal, in the directory your repository is stored at.

The git remote add command takes two arguments:

* A remote name, for example, origin
* A remote URL, for example, <https://github.com/OWNER/REPOSITORY.git>

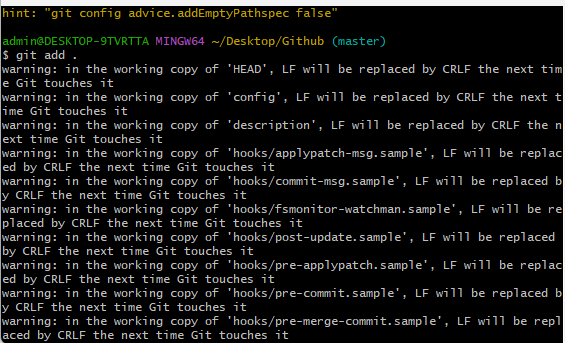
A computer screen with text

Description automatically generated

**3. Push files into remote repositories:**

Following commands are used

git add .



git commit -m "new commit"

A screenshot of a computer screen

Description automatically generated

git push origin main

A computer screen shot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

1. **Pull files from remote repository:**

git pull origin main

A computer screen with white text

Description automatically generated

1. **Merge branch**

git checkout master

git pull origin master

git merge test

git push origin master

A computer screen shot of text

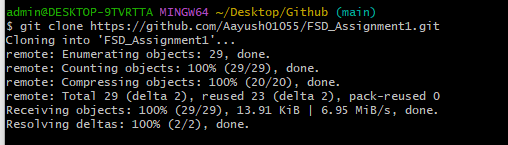
Description automatically generated

A screenshot of a computer program

Description automatically generated

1. **Cloning repository**

Git clone <repository url >



1. **Status of current working directory**

git status

A screenshot of a computer screen

Description automatically generated

**Github link:** [**https://github.com/Aayush01055/FSD-Assignments.git**](https://github.com/Aayush01055/FSD-Assignments.git)