# Sample Project Git

## 1. Introduction

This document has git commands implemented for version control of sample project.

## 2. Project Setup and Initialization

### 2.1 Creating a New Local Repository

1. Create and initialize the project:

* mkdir sample-project  
  cd sample-project  
  git init

1. Create initial project files:

* touch index.html

1. Make the first commit:

* git add .  
  git commit -m "Initial commit"

### 2.2 Connecting to GitHub

After creating a repository on GitHub, linking local repository:

1. Add the remote repository:

* git remote add origin https://github.com/sanskruti019/MyGitRepo.git
* git branch -M main  
  git push -u origin main

## 3. Branch Management

### 3.1 Creating and Managing Branches

1. Create and switch to a new branch:

* git checkout -b feature/update-readme

1. List branches:

* git branch # Local branches  
  git branch -r # Remote branches  
  git branch -a # All branches

### 3.2 Working with Files

Managing files and changes in repository:

1. Create and modify files:

* echo "# Sample Project" > README.md

1. Stage and commit changes:

* git add README.md  
  git commit -m "Add README with project description"

1. Push changes to remote:

* git push -u origin feature/update-readme

## 4. Merging and Cleanup

### 4.1 Merging Changes

When feature development is complete, merging changes into the main branch:

1. Prepare for merge:

* git checkout main  
  git pull origin main

1. Perform the merge:

* git merge feature/update-readme  
  git push origin main

### 4.2 Branch Cleanup

After successful merges, clean up unnecessary branches:

1. Delete local branch:

* git branch -d feature/update-readme

1. Delete remote branch:

* git push origin --delete feature/update-readme

**Key Questions Answered**

1. How will you set up the project for version control and push it to GitHub?

-Initialize a Git repository locally using git init.

-Create a corresponding repository on GitHub.

-Link the local repository to GitHub using git remote add origin.

-Stage, commit, and push the initial changes to the main branch with git push -u origin main.

2. What steps will you take to create, push, and manage branches effectively?

-Use git checkout -b <branch-name> to create and switch to a new branch.

-After making changes, stage and commit them. Push the branch using git push -u origin <branch-name>.

-Manage changes and reviews using GitHub's pull request functionality.

3. How will you ensure the changes in the feature branch are merged into the main branch?

-Submit a pull request from the feature branch on GitHub.

-After review, merge the changes into the main branch through GitHub.

-Pull the updated main branch locally using git pull.

4. What commands will you use to clean up unused branches locally and remotely?

-To delete a branch locally, use git branch -d <branch-name>.

-To remove the branch remotely, use git push origin --delete <branch-name>.

Name: Sanskruti Kakad

Date of submission : 5/01/2025