**Exercise 1**

Git commands are essential for developers to manage and track changes in their codebase. Here's a concise overview of the key commands you mentioned:

**git init**

The git init command initializes a new Git repository in the current directory. When you run this command, Git creates a hidden .git directory which contains all the necessary files to track your project's history. This is typically the first step when you want to start using Git on a new project.

**git status**

The git status command shows you the current state of your working directory and staging area. It lets you know which files have been modified, which files are staged for the next commit, and which files aren't being tracked by Git. It's a great way to get a quick overview of your project's status before making a commit.

**git add**

The git add command stages changes for the next commit. This means it takes changes from your working directory and adds them to the staging area. You can stage specific files by using git add <filename> or stage all changes with git add .. Staging allows you to group related changes into a single commit, giving you more control over your commit history.

**git commit**

The git commit command saves your staged changes to the repository's history. When you run git commit, Git creates a new commit object with a unique identifier. This commit captures a snapshot of your project at that moment. You should always include a clear and concise message with your commit using the -m flag, like git commit -m "Add new feature for user authentication".

**git push**

The git push command uploads your local commits to a remote repository. It's how you share your work with others and keep the remote repository up-to-date with your local changes. You need to specify the remote name (usually origin) and the branch name (e.g., main or master) like so: git push origin main.

**git pull**

The git pull command is used to download and integrate changes from a remote repository into your local branch. It's essentially a combination of git fetch (which downloads the changes) and git merge (which integrates them into your current branch). Running git pull origin main will update your local main branch with the latest changes from the remote origin repository.

ARVINTH@LAPTOP-KOK8JOQ1 MINGW64 ~/Downloads/GitDemo (master)

$ git version

git version 2.50.1.windows.1

ARVINTH@LAPTOP-KOK8JOQ1 MINGW64 ~/Downloads/GitDemo (master)

$ git config --global --list

filter.lfs.smudge=git-lfs smudge -- %f

filter.lfs.process=git-lfs filter-process

filter.lfs.required=true

filter.lfs.clean=git-lfs clean -- %f

user.name=Arvinth-M

user.email=122240192+Arvinth-M@users.noreply.github.com

core.editor="D:\Microsoft VS Code\bin\code" --wait



