# Assignment 1

## Linux Git Repository Manager

## Format

    Individual take-home assignment

## Goal

To create a simplified command-line tool that combines Git repository management with Linux file system operations. This tool will allow users to create Git repositories, clone existing repositories, perform basic commit and push operations, and manage files and directories within the repositories.

## Due

    By the end of Week 3 (11:59 PM on Sep. 22)

## Grace Period

    48 hours.

    The drop-box of this assignment will stay open for 48 more hours after the due time.

    No late penalty will be applied if you (re)submit your work before the end of Tuesday.

    After the given Grace Period, the drop-box will be no longer available, and no more submission will be accepted.

## Requirements

1. User Interface: Implement a user-friendly command-line interface that blends Git and Linux commands for users to interact with the tool effectively.
2. Git Initialization: Create a command to initialize a new Git repository in a specified directory.
3. Cloning Repositories: Implement a command to clone existing Git repositories from remote URLs to local directories.
4. Committing Changes: Allow users to add and commit changes to the local repository.
5. Pushing Changes: Create a command to push committed changes to a remote repository (GitHub, GitLab, etc.).
6. Linux File Operations: Integrate basic Linux file operations such as creating directories, deleting files and directories, and listing directory contents within the Git repository.
7. Error Handling: Implement error handling to manage both Git-related and Linux-related issues gracefully. Provide informative error messages.
8. Documentation: Create a user guide or help message that explains how to use your tool, including all available Git and Linux commands.

## Sample Commands

Here's the list of commands for this assignment:

1. **Initialize a Git Repository:**

mygit-init <directory>

This command initializes a new Git repository in the specified directory.

1. **Clone a Git Repository:**

mygit-clone <remote\_url> <local\_directory>

This command clones an existing Git repository from a remote URL to a local directory.

1. **Commit Changes:**

mygit-commit -m "<commit\_message>"

This command allows users to add and commit changes to the local Git repository.

1. **Push Changes:**

mygit-push

This command pushes committed changes to a remote Git repository.

1. **Create a Directory:**

mygit-create-directory <directory\_name>

This command creates a directory within the Git repository.

1. **Delete a File:**

mygit-delete-file <file\_name>

This command deletes a file from the repository.

1. **Delete a Directory:**

mygit-delete-directory <directory\_name>

This command deletes a directory and its contents from the repository.

1. **List Directory Contents:**

mygit-list-contents <directory>

This command lists the contents of a directory within the Git repository.

## Deliverables

One executable file OR one zip file including your multiple executables.

Each executable file should be one shell file (with .sh file extension)

## Scope

This assignment artifact allows users work with Git repositories in a more integrated way within the Linux environment. Users can manage both their version control and file system operations efficiently from the command line.

## References

Below are some inputs for your reference. Please notice that none of them is mandatory part of your assignment. They are just for your understanding of the topics.

### Shell Scripting

1. **Bash Guide for Beginners:** A comprehensive guide for beginners to shell scripting with Bash. It covers the basics and provides practical examples. [Bash Guide for Beginners](http://tldp.org/LDP/Bash-Beginners-Guide/html/)
2. **Advanced Bash-Scripting Guide:** A more in-depth resource for learning advanced shell scripting concepts and techniques. [Advanced Bash-Scripting Guide](http://tldp.org/LDP/abs/html/)
3. **Bash Reference Manual:** The official reference manual for the Bash shell. It provides detailed information about Bash commands and features. [Bash Reference Manual](https://www.gnu.org/software/bash/manual/)

### Linux

1. **Linux Documentation Project:** An extensive collection of guides, how-tos, and documentation on various Linux topics, including shell scripting. [Linux Documentation Project](http://tldp.org/)

### Git

1. **Pro Git:** An excellent and comprehensive book by Scott Chacon and Ben Straub that covers Git fundamentals and advanced topics. [Pro Git](https://git-scm.com/book/en/v2)
2. **Atlassian Git Tutorials:** A series of Git tutorials provided by Atlassian, the company behind Bitbucket. [Atlassian Git Tutorials](https://www.atlassian.com/git/tutorials)
3. **Git Cheat Sheet:** A handy cheat sheet with common Git commands and their descriptions. [Git Cheat Sheet](https://education.github.com/git-cheat-sheet-education.pdf)