Placement Empowerment Program

Cloud Computing and DevOps Centre

local repository in Git

Name: ERIC JEEVAN A   Department : ADS

Introduction and Overview

A **local Git repository** allows you to track changes to files and manage your codebase efficiently on your own computer before pushing changes to a remote repository (like GitHub or GitLab). It helps keep a record of your work and provides a history of changes, enabling easy collaboration, version control, and project management.

This guide will walk you through the process of creating a local Git repository, adding files, committing changes, and checking the status of your repository.

Objective

A **local Git repository** allows you to track changes to files and manage your codebase efficiently on your own computer before pushing changes to a remote repository (like GitHub or GitLab). It helps keep a record of your work and provides a history of changes, enabling easy collaboration, version control, and project management.

This guide will walk you through the process of creating a local Git repository, adding files, committing changes, and checking the status of your repository.

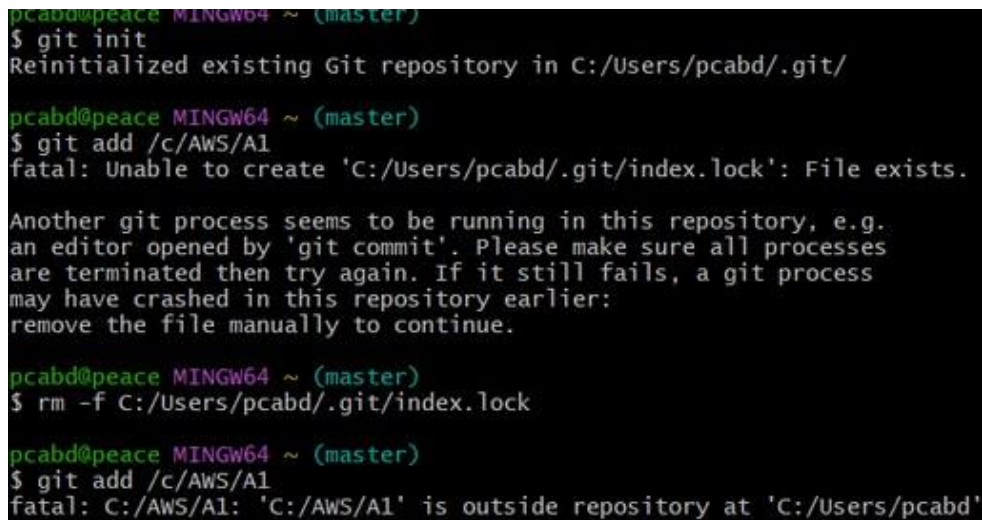# Step-by-Step Overview

Step 1: Initialize a New Git Repository

Open your terminal or **Git Bash**.

Navigate to the folder where you want to create the Git repository. You can do this with the cd command, like so:

cd path/to/your/project

- **2.Initialize a new Git repository**

  git init

```
pcabd@peace MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/pcabd/.git/

pcabd@peace MINGW64 ~ (master)
$ git add /c/AWS/A1
fatal: Unable to create 'C:/Users/pcabd/.git/index.lock': File exists.

Another git process seems to be running in this repository, e.g.
an editor opened by 'git commit'. Please make sure all processes
are terminated then try again. If it still fails, a git process
may have crashed in this repository earlier:
remove the file manually to continue.

pcabd@peace MINGW64 ~ (master)
$ rm -f C:/Users/pcabd/.git/index.lock

pcabd@peace MINGW64 ~ (master)
$ git add /c/AWS/A1
fatal: C:/AWS/A1: 'C:/AWS/A1' is outside repository at 'C:/Users/pcabd'
```

Step 2: Add Files to the Repository

- After initializing the repository, you need to tell Git which files to track. Use the git add command to add individual files

git add "file_name"

- To add all files in the directory, use:

git add .

Commit the Changes

Once your files are staged, commit them to the repository

git commit -m "Initial commit"

Check the Status

git status

Expected Outcome

After setting up the script and scheduling it, the following will happen:

1. Automated Copying: The script will automatically copy files from the source folder to the destination folder at the scheduled time.
2. Confirmation: A message will appear confirming that the files have been copied successfully.
3. No Manual Intervention: The task will run on its own without needing you to start it manually.
4. Error Handling: Any issues during copying will be logged, but the script will continue without stopping.