**Exercise - 1**

**Objectives**

Familiarity with Git commands such as:  
git init, git status, git add, git commit, git push, and git pull.

**In this lab, we will learn to:**

* Setup Git configuration on a local machine
* Integrate notepad++.exe with Git and set it as the default editor
* Add a file to a source code repository

**Prerequisites**

* Git Bash client must be installed on the machine.
* Do **not** use corporate credentials (e.g., Cognizant email ID) when signing up for GitHub/GitLab.
* Estimated time to complete: **30 minutes**

**Step 1: Git Configuration Setup**

1. **Check Git Installation**
   * Open **Git Bash** and execute:

git --version

* + Output like git version 2.XX.X indicates that Git is installed properly.

1. **Configure User Details**
   * Execute:

git config --global user.name "Your Name"

git config --global user.email "yourname@example.com"

1. **Verify Git Configuration**
   * Execute:

git config --list

* + Confirms the user name and email have been configured correctly.

**Step 2: Integrate Notepad++ with Git**

1. **Check if Notepad++ Runs from Git Bash**
   * In Git Bash, type:

notepad++

* + If it's not recognized, it means the path is not set.

1. **Add Notepad++ to Environment Variable**
   * Navigate:  
     Control Panel → System → Advanced System Settings → Environment Variables
   * Under *User Variables*, edit the Path and add the path to notepad++.exe (e.g., C:\Program Files\Notepad++).
2. **Restart Git Bash and Test Again**
   * Run:

notepad++

1. **Create Alias for Notepad++**
   * Run:

alias notepad='"/c/Program Files/Notepad++/notepad++.exe"'

1. **Configure Git to Use Notepad++**
   * Execute:

git config --global core.editor "'/c/Program Files/Notepad++/notepad++.exe' -multiInst -notabbar -nosession -noPlugin"

1. **Verify Default Editor**
   * Run:

git config --global --edit

* + It should open Git config file in Notepad++.

**Step 3: Add a File to Git Repository**

1. **Create New Project Directory**
   * Run:

mkdir GitDemo

cd GitDemo

git init

1. **Verify Initialization**
   * Run:

ls -a

* + You should see a .git folder.

1. **Create and Add File**
   * Run:

echo "Welcome to Git Lab!" > welcome.txt

1. **Check File Creation**
   * Run:

ls

1. **View File Content**
   * Run:

cat welcome.txt

1. **Check Git Status**
   * Run:

git status

1. **Add File to Staging Area**
   * Run:

git add welcome.txt

1. **Commit File with Multi-Line Comment**
   * Run:

git commit

* + This opens Notepad++ (default editor) to enter a multi-line commit message.

1. **Check Final Status**
   * Run:

git status

* + Output: No changes, as the file has been committed.

1. **Create Remote Repository**

* Login to GitLab
* Create a repository named GitDemo

1. **Pull Remote Repository**

* Run:

git pull origin master

1. **Push Local to Remote**

* Run:

git push origin master