A. Installation and Getting Started

1. Download and install the git:

- ✓ If you have not installed git, please use this link:
  https://git-scm.com/downloads

- ✓ If you already downloaded and installed git, send git bash on your desktop as a shortcut.

2. Verify Your git version

- ✓ Open a command line and verify that git is installed.

```
$ git --version
```

3. Explore git help

- ✓ View overall git help by typing git at the command line.

- ✓ Choose any git command that you see and view the help on it using git help <command>.

```
$ git help init
```

- ✓ View concise help on a command using the -h option.

```
$ git init -h .
```

4. Configure your user name, email address and default git editor

- ✓ View your current setting for your user name with:

```
$ git config user.name
```

- ✓ If you would like to change your user name, use:

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

- ✓ View your current setting for your email address with:

```
$ git config user.email
```

- ✓ If you would like to change your email address, use:

```
$ git config --global user.email "your_email@email.com"
```

5. Clean your git bash screen

- ✓ You can clean your git bash screen with:

```
$ clear
```

B. Create a git project

1. Create new directory or folder

- ✓ Go to desktop, right click on it and select New → Folder and give this folder a name, for example "Your name".
- ✓ If you would like to use git command in order to make new directory, write:

```
$ mkdir Folder-Name
```

2. Create new text file

- ✓ Go to your current directory, right click on it and select New → Text Document and give this file a name, for example "text.txt".
- ✓ If you would like to use git command in order to create new text file, write:

```
$ touch text.txt
```

C. Manage Your Path

1. Your current path

- ✓ If you would like to display your current path, use:

```
$ pwd
```

2. Change your path

- ✓ If you would like to go to your home directory for your computer, use:

```
$ cd ~
```

- ✓ If you would like to move backward only one directory, use:

```
$ cd ..
```

- ✓ If you would like to go to a specific directory, use:

```
$ cd <Folder-Name>
```

3. Display all folders and files

- ✓ If you would like to list all the folders and files inside the current directory, use:

```
$ ls
```

D. Create a Local Repository

1. Initialize an empty local git repository

- ✓ If you would like to turn your current project directory into .git directory, you need to initialize your first local repository inside your current directory.

```
$ git init
Initialized empty Git repository in projecta/.git/
```

- ✓ If you would like to verify that your project directory contains a .git directory, use:

```
$ ls -a
.
..
.git
```

E. Commit to a Local Repository

1. View file status

- ✓ If you would like to view your current directory status, use:

```
$ git status
```

(Note: for the first time you should see the message "No commit yet")

- ✓ Create an empty file named file.txt inside your current directory, execute git status command, you should see that git notices the file.txt identified as untracked (located inside the working tree).

2. Stage content

- ✓ If you would like to add file.txt to the staging area, use:

```
$ git add file.txt
```

- ✓ Execute git status again. You should see that file.txt status changed to be committed.

3. Commit content

- ✓ If you would like to create a commit, use:

```
$ git commit -m "Write-Commit-Message"
```

4. View the commit history

- ✓ If you would like to see your commit details (including your commit message), use:

```
$ git log
```

- ✓ If you would like to display a concise version of the history, use:

```
$ git log --oneline
```

- ✓ If you would like to display the history of specific author, use:

```
$ git log --auther=”Author-Name”
```

- ✓ If you would like to display the history of specific period of time, use:

```
$ git log --until=dd-mm-yy OR $ git log --since=dd-mm-yy
```

- ✓ If you will not use your current local repository in future labs. Feel free to delete it by simply deleting your current directory.

Congratulations .. You have completed this lab.