**Git Terminal commands**

1. Check if local system has Git tool. (run either in windows terminal / Git bash)

**git - -version**

1. Checks default editor that Git Bash uses.

**git config - -global core.editor**

1. If return “**code - -wait**” – then VS Code is set as the default editor.
2. If nothing is shown, Git is likely using the default **Vim editor** which comes built-in with Git Bash

**Test: vim test.txt** – if it opens a text editor inside Git Bash, then Git Bash is using Vim editor

1. To check location path of VS Code in local system.

**Where code**

1. To change Git Bash editor.

**git config - -global core.editor “notepad”** - making Notepad as default editor.

**git config - -global core.editor “code - -wait” -** making VS Code as default editor.

**NOTE:** ‘code’ should be in PATH

1. Create directory

**mkdir folder\_name**

1. Change directory

**cd folder\_name**

1. List files and folders inside directory

**ls**

1. Create a file form Git Bash.

**touch hello.py -** Creates new file and Update last modified time of

an existing file.

**NOTE: ‘*touch’*** is Linux/Unix command. Since Git Bash uses a Linux-like environment (vis MinGW64), **‘*touch’*** works just like in Linux**.**

1. Open file or folder in VS Code from Git Bash terminal

**code hello.py or code .**

1. To run python file from Git Bash terminal

**./hello.py**

1. Command to know where python interpreter located in local system

**where python**

1. Command to run python file form **Windows terminal/VS Code terminal**.

**Make sure** you have opened terminal from the folder where your python file reside.

**python hello.py or C:/Python313/python.exe hello.py**

Steps to clone GitHub repo to local system and Push our local project to cloned GitHub repo

$ git remote add origin <https://github.com/Indraja9090/IndrajaBrishRepo.git>

$ git branch -M main

$ git push -u origin main

$ git remote remove origin

$ git remote -v

git push or git push --set-upstream origin main

git checkout -b main

git branch

git pull origin main --rebase