Git commands

Commands :

git init

git status

git add [file]

git log

git commit -m "[descriptive message]"

Setup command:

git config --global user.name "[firstname lastname]"

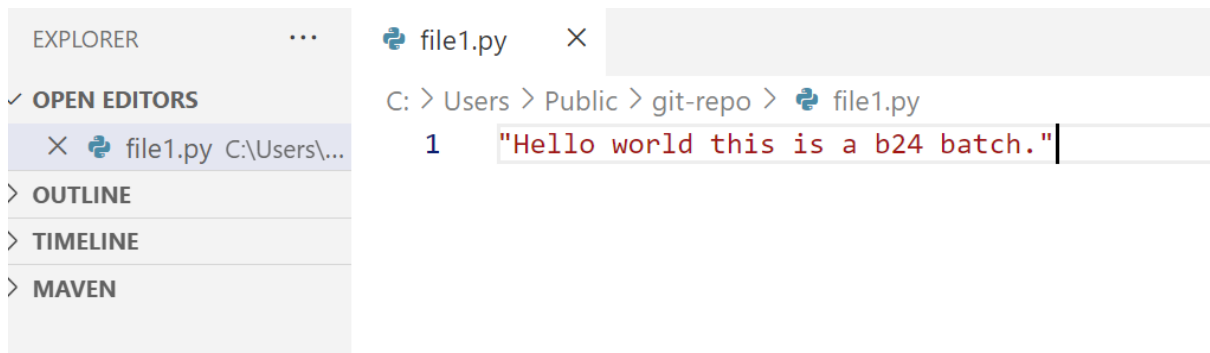git config --global user.email "[valid-email]"

1.Create folder git-repo:

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public
$ mkdir git-repo

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public
$ ls
AccountPictures/
Desktop/
Documents/
Downloads/
Libraries/
Music/
NTUSER.DAT
NTUSER.DAT.LOG1
NTUSER.DAT.LOG2
NTUSER.DAT{9f190aaf-f491-11ed-9f14-ac4171aa4dd4}.TM.blf
NTUSER.DAT{9f190aaf-f491-11ed-9f14-ac4171aa4dd4}.TMContainer00000000000000000001.regtrans-ms
NTUSER.DAT{9f190aaf-f491-11ed-9f14-ac4171aa4dd4}.TMContainer00000000000000000002.regtrans-ms
Pictures/
Videos/
desktop.ini
git-repo/

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public
$ cd git-repo

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo
$ ls
file1.py
```

2.Create file file1.py in git-repo directory

```
EXPLORER                    ...        🐍 file1.py        ✕
✓ OPEN EDITORS
    ✕  🐍 file1.py  C:\Users\...        C: › Users › Public › git-repo › 🐍 file1.py
> OUTLINE                                     1      "Hello world this is a b24 batch."
> TIMELINE
> MAVEN
```

3.Initialize git :

git init : initialize an existing directory as a Git repository

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo
$ git init
Initialized empty Git repository in C:/Users/Public/git-repo/.git/

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ ls -a
./  ../  .git/  file1.py

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ cd .git/
```

4. check git status :

Git status : show modified files in working directory, staged for your next commit

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        file1.py

nothing added to commit but untracked files present (use "git add" to track)
```

5.Add file in staging area and check git status again file is committed :

git add [filename] : add a file as it looks now to your next commit (stage)

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git add file1.py

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   file1.py
```

6 .commit git :

git commit -m "[descriptive message]" : commit your staged content as a new commit snapshot

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git commit -m "this is version 1"
[master (root-commit) 4923799] this is version 1
 1 file changed, 1 insertion(+)
 create mode 100644 file1.py
```

7. Update the file :

```
EXPLORER                    ...    🐍 file1.py    ✕

∨ OPEN EDITORS                     C: > Users > Public > git-repo > 🐍 file1.py
    ✕ 🐍 file1.py  C:\Users\...          1    "Hello world this is a b24 batch."
> OUTLINE                                2    "Hello cloudblitz"
> TIMELINE
> MAVEN
```

8.check status :

```
create mode 100644 file1.py

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   file1.py

no changes added to commit (use "git add" and/or "git commit -a")
```

9.Again git add file1.py and check status been modified :

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git add file1.py

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   file1.py
```

10.add new version file be changed :

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git commit -m "This is version 2"
[master 945baff] This is version 2
 1 file changed, 2 insertions(+), 1 deletion(-)
```

11.check git log :

git log : show all commits in the current branch's history

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git log
commit 945baff9e4ebc7ec54443924268f26e8ffd07d35 (HEAD -> master)
Author: Nehakolte <kolateneha401@gmail.com>
Date:    Wed Feb 14 23:07:13 2024 +0530

    This is version 2

commit 492379962d359f3866769aee1f6ea5bcb2929263
Author: Nehakolte <kolateneha401@gmail.com>
Date:    Wed Feb 14 22:59:34 2024 +0530

    this is version 1
```

12.git log –-oneline : display current directory

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git log --oneline
945baff (HEAD -> master) This is version 2
4923799 this is version 1
```

13.git reset –-soft commit id :

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git reset --hard 945baff
HEAD is now at 945baff This is version 2

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git log
commit 945baff9e4ebc7ec54443924268f26e8ffd07d35 (HEAD -> master)
Author: Nehakolte <kolateneha401@gmail.com>
Date:    Wed Feb 14 23:07:13 2024 +0530

    This is version 2

commit 492379962d359f3866769aee1f6ea5bcb2929263
Author: Nehakolte <kolateneha401@gmail.com>
Date:    Wed Feb 14 22:59:34 2024 +0530

    this is version 1
```

14.git reset –hard commit id:

```
Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git reset --hard 4923799
HEAD is now at 4923799 this is version 1

Neha@LAPTOP-44014GTB MINGW64 /c/Users/public/git-repo (master)
$ git log
commit 492379962d359f3866769aee1f6ea5bcb2929263 (HEAD -> master)
Author: Nehakolte <kolateneha401@gmail.com>
Date:   Wed Feb 14 22:59:34 2024 +0530

    this is version 1
```