Assignment 1: Initialize a new Git repository in a directory of your choice. Add a simple text file to the repository and make the first commit.

Git Initialization

```
Git Init
```

Adding a text file

➔ using touch command

```
touch gitFile.txt
```

➔ Or using cat command

```
cat >gitFile2.txt
```

Steps to commit files

➔ Firstly, we have to add a file into staging area:
  o This can be done with either using only current file name needed to commit:

```
git add gitFile.txt
```

  o Or we can add all files of directory to staging area:

```
git add .
```

➔ After adding to stage area we can commit our changes using:

```
git commit -m "Added: files into git initially"
```

Git Bash Process as follows:

```
hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git
$ git init
Initialized empty Git repository in F:/_wipro_711/Assigns/Git/.git/

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ touch gitFile.txt

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git status
On branch master

No commits yet

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

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

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git add .

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   gitFile.txt


hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git commit -m "Added: Files are added initially."
[master (root-commit) aa131c5] Added: Files are added initially.
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 gitFile.txt

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ |
```

Assignment 2: Branch Creation and Switching

Create a new branch named 'feature' and switch to it. Make changes in the 'feature' branch and commit them.

Create a New Branch and Switching on it:

➔ Using **checkout** only to switch and create branch at same time:

```
git checkout -b feature
```

➔ Using checkout and branch only to switch and create branch at same time:

- o **branch** command to create branch named feature

```
Git branch feature
```

- o checkout command to switch to feature branch

```
git checkout feature
```

- o **Alternatively, switch** command to switch to feature

```
git switch feature
```

➔ After switch and changing files, we use add and commit to save changes:

```
git add .

git commit -m "Changes: Files are Changed on feature branch"
```

Git Bash Process as follows:

```
hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns (master)
$ git branch feature

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns (master)
$ git branch
  feature
* master

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns (master)
$ git switch feature
Switched to branch 'feature'

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns (feature)
$ git branch
* feature
  master

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns (feature)
$ ls
Git/

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns (feature)
$ cd Git

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ ls
 assign1.PNG    gitFile.txt    git_assignments.docx    git_assignments.pdf    '~$t_assignments.docx'

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ cat >>gitFile.txt
Adding something new from feature branch
to make changes in text files

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ cat gitFile.txt
Make Changes In Files in git master branch
Adding something new from feature branch
to make changes in text files

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ git status
On branch feature
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:   gitFile.txt

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

```
hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ git status
On branch feature
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:   gitFile.txt

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

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ git add .
warning: in the working copy of 'Git/gitFile.txt', LF will be replaced by CRLF the next time Git touches it

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ git status
On branch feature
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   gitFile.txt

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ git commit -m "Changes: On Feature branch to gitFile"
[feature 010e0ad] Changes: On Feature branch to gitFile
 1 file changed, 2 insertions(+)

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ git status
On branch feature
nothing to commit, working tree clean

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ git push origin feature
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 408 bytes | 408.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'feature' on GitHub by visiting:
remote:      https://github.com/Sanjogharinkhede/my_private_repo/pull/new/feature
remote:
To https://github.com/Sanjogharinkhede/my_private_repo.git
 * [new branch]      feature -> feature

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (feature)
$ |
```

Assignment 3: Feature Branches and Hotfixes

Create a 'hotfix' branch to fix an issue in the main code. Merge the 'hotfix' branch into 'main' ensuring that the issue is resolved.

Fixing a bug on hotfix and merging to master(main):

➔ Hotfixes are usually created from master branch and switch to hotfix with latest version of it.

➔
```
git checkout master

git pull origin master

git checkout -b hotfix
```

➔ Make necessary changes on code and save changes locally then commit:

```
git add .

git commit -m "Hotfix: Bug on code with msg"
```

➔ After Fixing issues and testing, merge branch to main for that we have to checkout to main and there use merge request or if having permissions merge hotfix branch into master

```
git checkout main

git merge hotfix
```

➔ Usually in production we make tags for these hotfixes (say version 1.1.0), can be done using and push to master branch with tags

```
git tag -a v1.1.0 -m "Release: Fix critical issue"

git push origin main --tags
```

➔ Usually after fixes we delete the hotfix branch:

```
git branch -d hotfix

git push origin --delete hotfix
```

Git Bash Process as follows:

```
git checkout -b hotfix
error: pathspec 'main' did not match any file(s) known to git
fatal: couldn't find remote ref main
Switched to a new branch 'hotfix'

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ git status
On branch hotfix
nothing to commit, working tree clean

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ cat code.sh
x=1
y=2
echo $x+$y

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ vi code.sh

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ cat code.sh
x=1
y=2
echo $((x+y))

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ bash code.sh
3

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ git add .
warning: in the working copy of 'Git/code.sh', LF will be replaced by CRLF the next time Git touches it

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ git commit -m "Fix: concate issue fixed"
[hotfix f774773] Fix: concate issue fixed
 1 file changed, 1 insertion(+), 1 deletion(-)

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ git push origin hotfix
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 330 bytes | 330.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'hotfix' on GitHub by visiting:
remote:      https://github.com/Sanjogharinkhede/my_private_repo/pull/new/hotfix
remote:
To https://github.com/Sanjogharinkhede/my_private_repo.git
 * [new branch]      hotfix -> hotfix

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
```

```
hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (hotfix)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git merge hotfix
Updating dab1963..f774773
Fast-forward
 Git/code.sh | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git tag -a v1.1.0 -m "Release: Fix arithmetic concate issue"

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git push origin main --tags
error: src refspec main does not match any
error: failed to push some refs to 'https://github.com/Sanjogharinkhede/my_private_repo.git'

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git push origin master --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 188 bytes | 188.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Sanjogharinkhede/my_private_repo.git
   dab1963..f774773  master -> master
 * [new tag]         v1.1.0 -> v1.1.0

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git branch -d hotfix
Deleted branch hotfix (was f774773).

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git push origin --delete hotfix
To https://github.com/Sanjogharinkhede/my_private_repo.git
 - [deleted]         hotfix

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ git branch
  feature
* master

hp@DESKTOP-RILG1IE MINGW64 /f/_wipro_711/Assigns/Git (master)
$ |
```