# Git - Assignment 1

**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.**

**Steps :**

1. **Open Terminal/Command Prompt:** Navigate to the directory where you want to start your project.
2. **Initialize Git:** Type “git init” to set up a new Git repository in that directory.
3. **Add File to Git**: Type “git add .” to add the text file to Git.
4. **Make a Commit:** Type “git commit -m "Add newFile.txt" “ to save the changes you've made to the text file.
5. **Verify:** Type “git status” to see if everything is in order. It should show that there are no more changes to commit.

```
E:\Ep-Java-WIpro\New folder>git init
Initialized empty Git repository in E:/Ep-Java-WIpro/New folder/.git/

E:\Ep-Java-WIpro\New folder>git add .

E:\Ep-Java-WIpro\New folder>git status
On branch master

No commits yet

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


E:\Ep-Java-WIpro\New folder>git commit -m "Adding newFile"
[master (root-commit) 72d9873] Adding newFile
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 newFile.txt

E:\Ep-Java-WIpro\New folder>git status
On branch master
nothing to commit, working tree clean
```

**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.**

1. **Create a New Branch:** Use the command **'git checkout -b feature'** to create a new branch named 'feature' and switch to it. This command creates the branch and immediately switches to it.
2. **Make Changes:** Make the desired changes to your files. You can use any text editor or IDE to modify the files in your project directory.
3. **Stage Changes**: Use the command **'git add .'** to stage all the changes you've made.
4. **Commit Changes:** Commit the staged changes to the 'feature' branch using the '**git commit'** command. Provide a meaningful commit message to describe the changes.

```
E:\Ep-Java-WIpro\New folder>git branch
* master

E:\Ep-Java-WIpro\New folder>git checkout -b feature
Switched to a new branch 'feature'

E:\Ep-Java-WIpro\New folder>git branch
* feature
  master

E:\Ep-Java-WIpro\New folder>git add .

E:\Ep-Java-WIpro\New folder>git commit -m "New changes are made and adding file"
On branch feature
nothing to commit, working tree clean

E:\Ep-Java-WIpro\New folder>git status
On branch feature
nothing to commit, working tree clean
```

**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.**

**Steps:**

1. **Create a Hotfix Branch:** Use the command **'git checkout -b hotfix'** to create a new branch named 'hotfix' and switch to it. This branch will be used to fix the issue in the main code.
2. **Make Changes:** Make the necessary changes in the 'hotfix' branch to fix the issue in the main code.
3. **Stage Changes:** Use the command **'git add .'** to stage the changes you've made.
4. **Commit Changes:** Commit the staged changes to the 'hotfix' branch with a descriptive commit message.

5. **Switch to Main Branch**: Switch back to the 'main' branch using the command **‘git checkout main’**.
6. **Merge Hotfix Branch:** Merge the 'hotfix' branch into the 'main' branch to apply the fix to the main code. Use the command **‘git merge hotfix’**.
7. **Resolve Conflicts (if any):** If there are any conflicts during the merge, resolve them manually and then commit the changes.
8. **Push Changes:** Finally, push the changes to the remote repository to update the 'main' branch with the fix.

```
E:\Ep-Java-WIpro\New folder>git checkout -b hotfix
Switched to a new branch 'hotfix'

E:\Ep-Java-WIpro\New folder>git add .

E:\Ep-Java-WIpro\New folder>git commit -m "Changes are made and commiting in hotfix"
[hotfix ddcc11e] Changes are made and commiting in hotfix
 1 file changed, 2 insertions(+), 1 deletion(-)

E:\Ep-Java-WIpro\New folder>git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

E:\Ep-Java-WIpro\New folder>git merge hotfix
Updating dc91064..ddcc11e
Fast-forward
 newFile.txt | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)

E:\Ep-Java-WIpro\New folder>git push -u origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 285 bytes | 285.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/kpranay613/myrepo.git
   dc91064..ddcc11e  master -> master
branch 'master' set up to track 'origin/master'.
```