ASSIGNMENT - GIT

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.

```
rps@rps-virtual-machine:~$ mkdir Assignment1
rps@rps-virtual-machine:~$ ls
Assignment1  Documents  eclipse-workspace  Music     Public  Templates  Videos  wiproAssignments
Desktop      Downloads  git                Pictures  snap    test2.sh   wipro
rps@rps-virtual-machine:~$ cd Assignment1
rps@rps-virtual-machine:~/Assignment1$ pwd
/home/rps/Assignment1
rps@rps-virtual-machine:~/Assignment1$ touch sample.txt
rps@rps-virtual-machine:~/Assignment1$ ls
sample.txt
```

1.mkdir Assignment1 - Directory created in local machine

2. cd Assignment1 - Change Directory

3. git init - initialized repository

4. vi sample.txt - created a file

```
Cloning into 'wiproAssignments'...
remote: Enumerating objects: 33, done.
remote: Counting objects: 100% (33/33), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 33 (delta 5), reused 30 (delta 5), pack-reused 0
Receiving objects: 100% (33/33), 4.06 KiB | 1.35 MiB/s, done.
Resolving deltas: 100% (5/5), done.
```

5. git add . - added a change in working directory to staging area

```
rps@rps-virtual-machine:~/Assignment1$ git commit -m "first commit"
On branch master

Initial commit

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

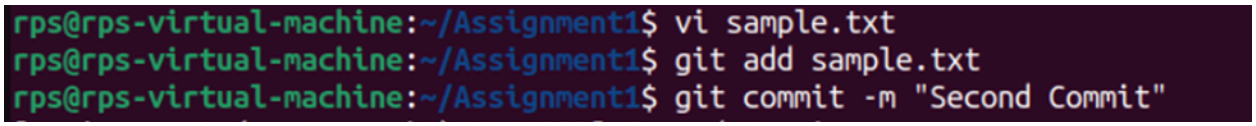
6. git commit -m “first commit” - created commit

7. git remote add origin https://github.com/ReshJain/WiproTraning - Added a new remote connection to the repository.
8. git push origin main - pushed changes to the remote repository from local repository.

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

The following commands are used for branch creation and switching in the above repository:

1. git branch - checked the current branch
2. git checkout -b feature - created a new branch and switched from main to feature



3. vi example.txt - created a file
4. git add example.txt - added a change to staging area
5. git commit -m “In feature after switching” - created commit
6. git push origin main - pushed the changes

ASSIGNMENT - GIT

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.

The issue was a typo issue in intro.txt(in main branch) which had text inside it “Hello” corrected to “Hello”(from feature branch). The following commands were used for hotfix:

```
rps@rps-virtual-machine:~/Assignment1$ git checkout -b hotfix
Switched to a new branch 'hotfix'
rps@rps-virtual-machine:~/Assignment1$
```

1. git branch - checked branch which was in feature branch
2. git checkout -b hotfix feature - created a branch and switched

```
rps@rps-virtual-machine:~/Assignment1$ vi sample.txt
rps@rps-virtual-machine:~/Assignment1$ git add sample.txt
```

3. git add sample.txt - corrected file of the main branch file
4. git commit -m “Fixing” - created commit

```
rps@rps-virtual-machine:~/Assignment1$ git status
On branch hotfix
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   sample.txt

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

rps@rps-virtual-machine:~/Assignment1$ git commit -m "Third Commit"
[hotfix 2f0227b] Third Commit
 1 file changed, 2 insertions(+)
rps@rps-virtual-machine:~/Assignment1$
```

5. git checkout main - switched from hotfix to main branch
6. git merge hotfix - merged hotfix branch into main branch
7. git push origin main - pushed the changes in the repository