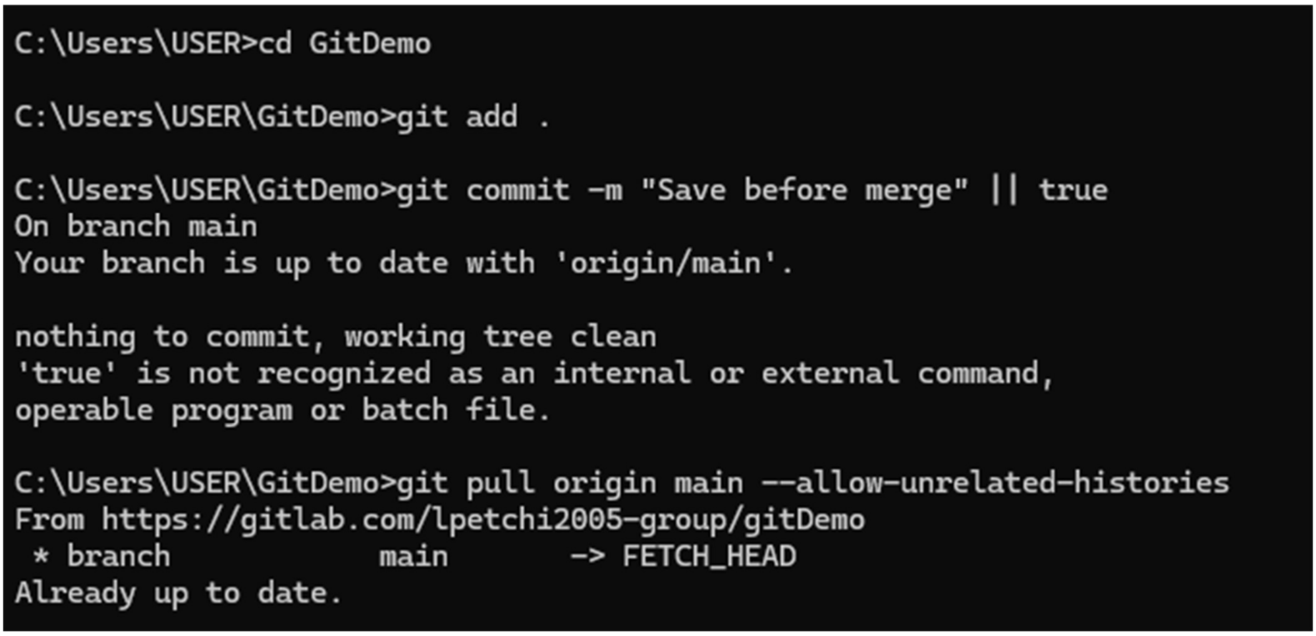
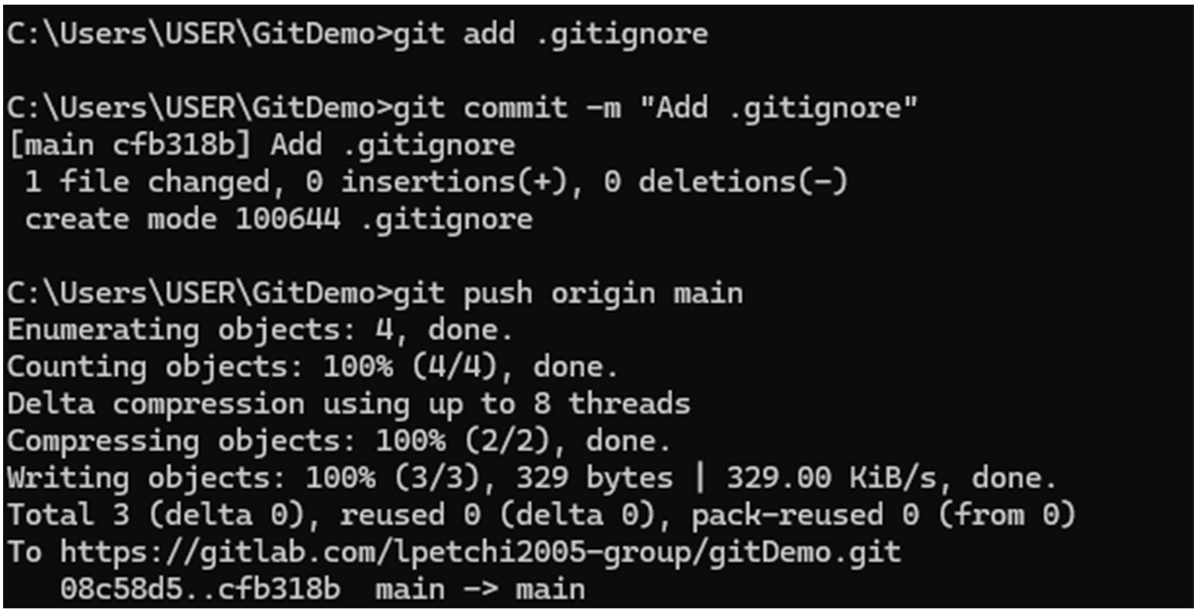
2. Git-HOL

Create a “.log” file and a log folder in the working directory of Git. Update the .gitignore file in such a way that on committing, these files (.log extensions and log folders) are ignored.

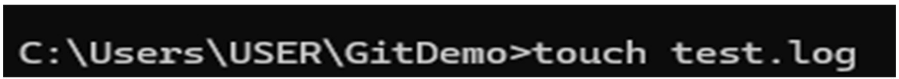
Verify if the git status reflects the same about working directory, local repository and git repository.



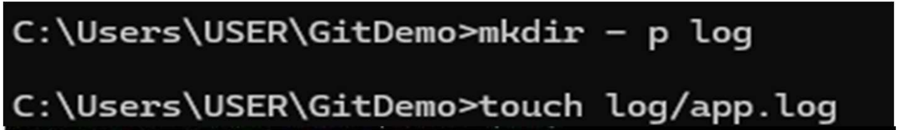
```
C:\Users\USER>cd GitDemo

C:\Users\USER\GitDemo>git add .

C:\Users\USER\GitDemo>git commit -m "Save before merge" || true
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
'true' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\USER\GitDemo>git pull origin main --allow-unrelated-histories
From https://gitlab.com/lpetchi2005-group/gitDemo
 * branch            main       -> FETCH_HEAD
Already up to date.
```



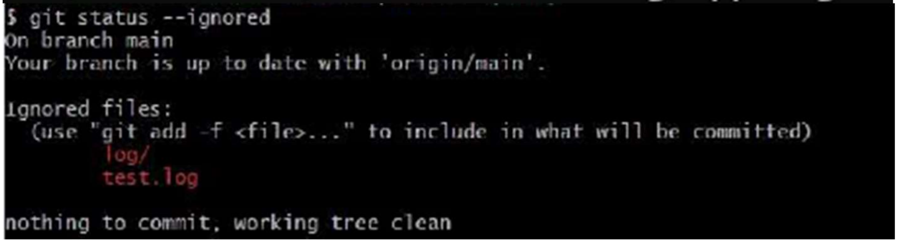
```
C:\Users\USER\GitDemo>git add .gitignore

C:\Users\USER\GitDemo>git commit -m "Add .gitignore"
[main cfb318b] Add .gitignore
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 .gitignore

C:\Users\USER\GitDemo>git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 329 bytes | 329.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://gitlab.com/lpetchi2005-group/gitDemo.git
   08c58d5..cfb318b  main -> main
```



```
C:\Users\USER\GitDemo>touch test.log
```



```
C:\Users\USER\GitDemo>mkdir - p log

C:\Users\USER\GitDemo>touch log/app.log
```



```
$ git status --ignored
On branch main
Your branch is up to date with 'origin/main'.

Ignored files:
  (use "git add -f <file>..." to include in what will be committed)
        log/
        test.log

nothing to commit, working tree clean
```