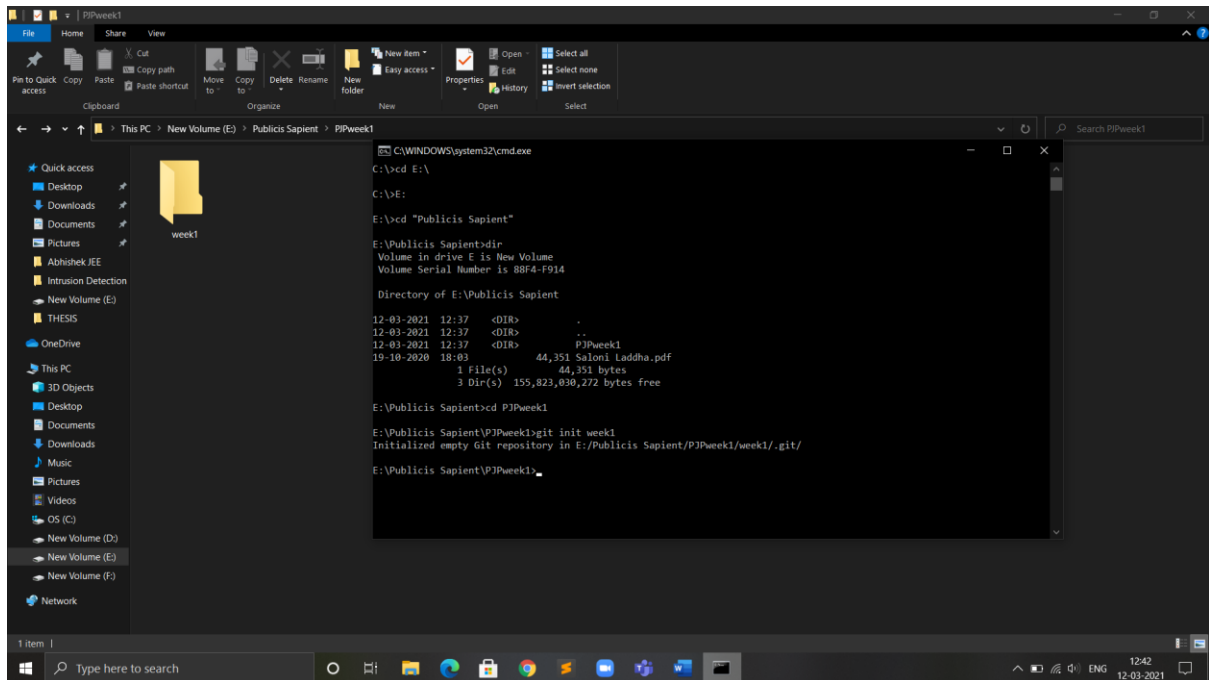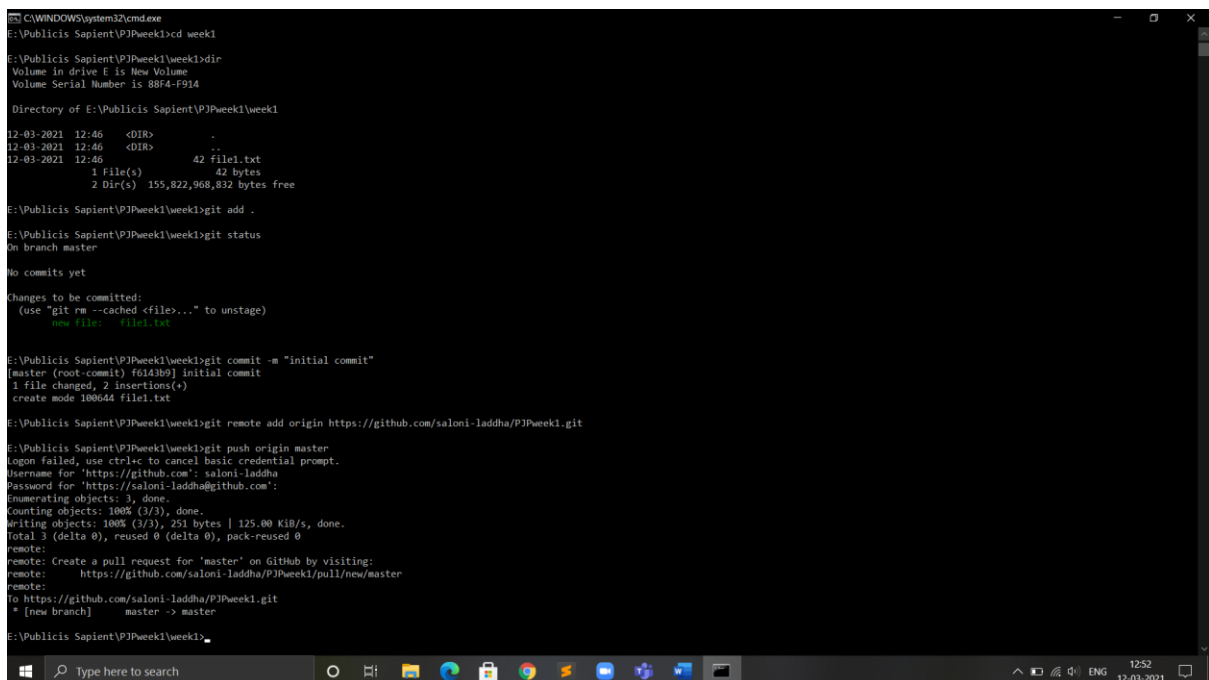# PJP Assignment-1

GIT

Submitted by: Saloni Laddha, NIT Hamirpur
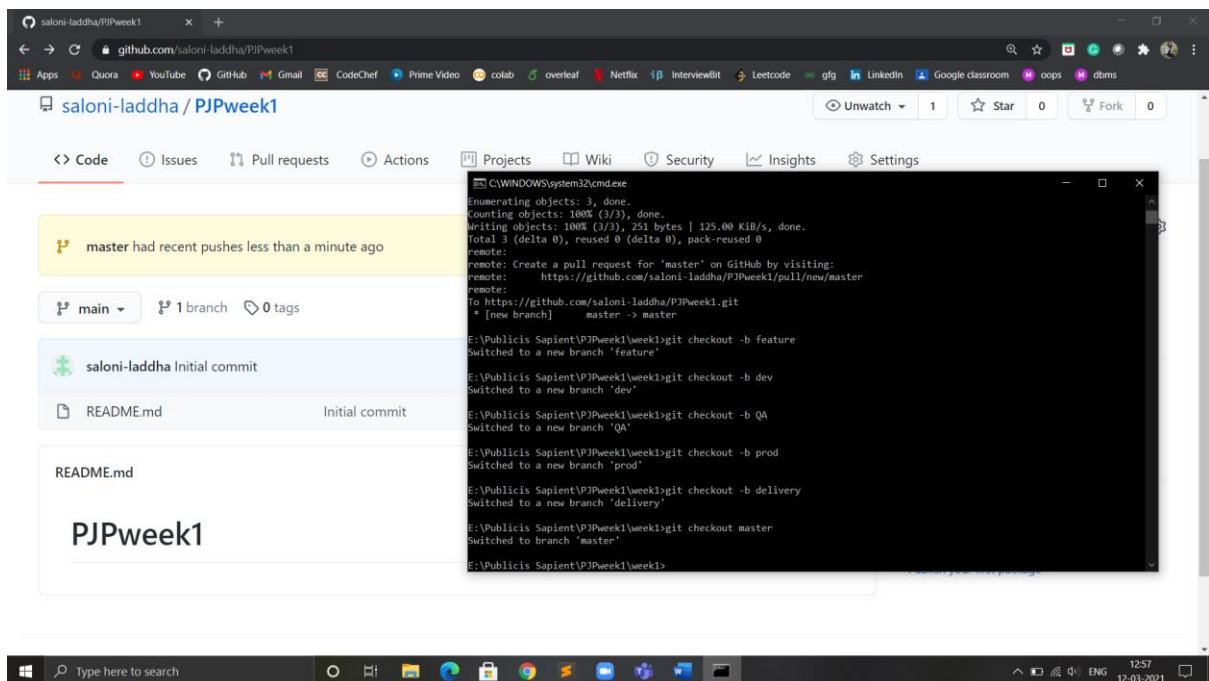
A) Basic usage of cli
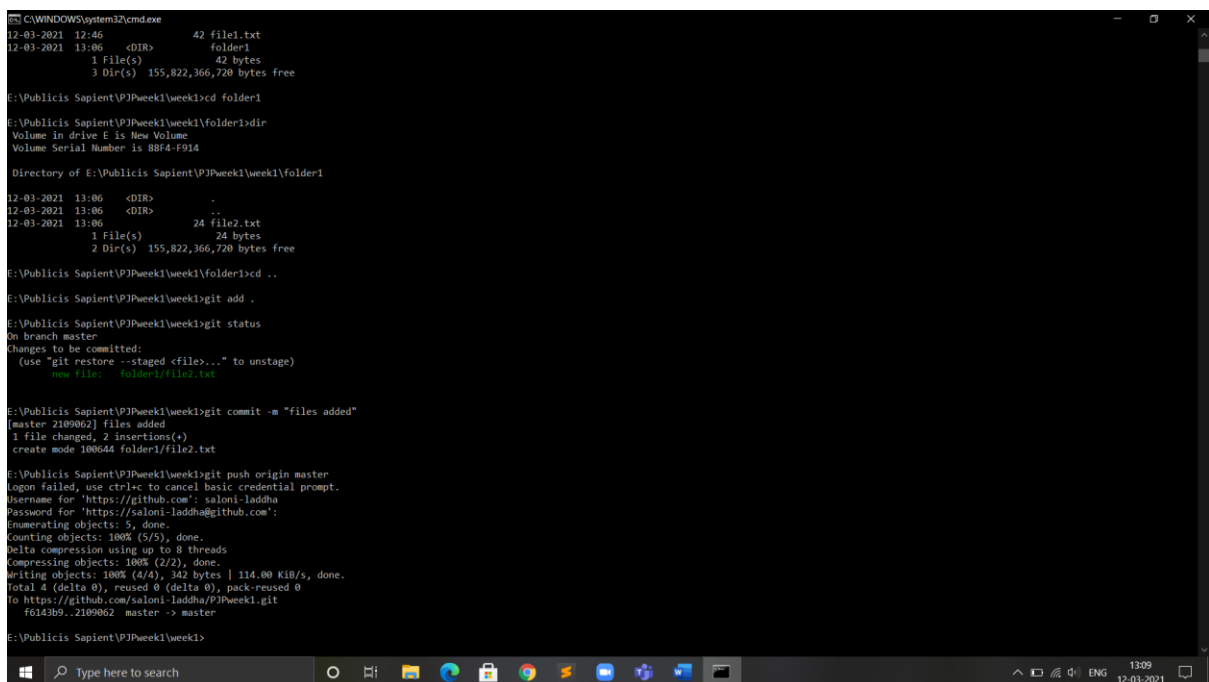
i. Setup a local repository
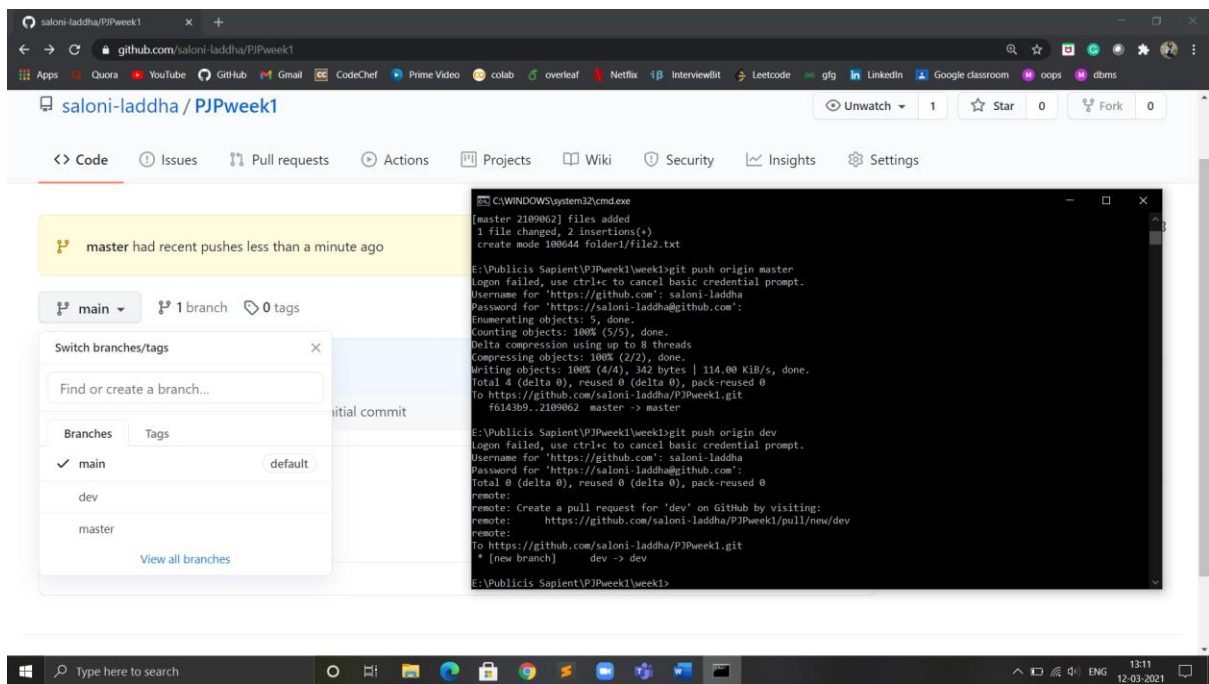

ii. Setup a remote repository

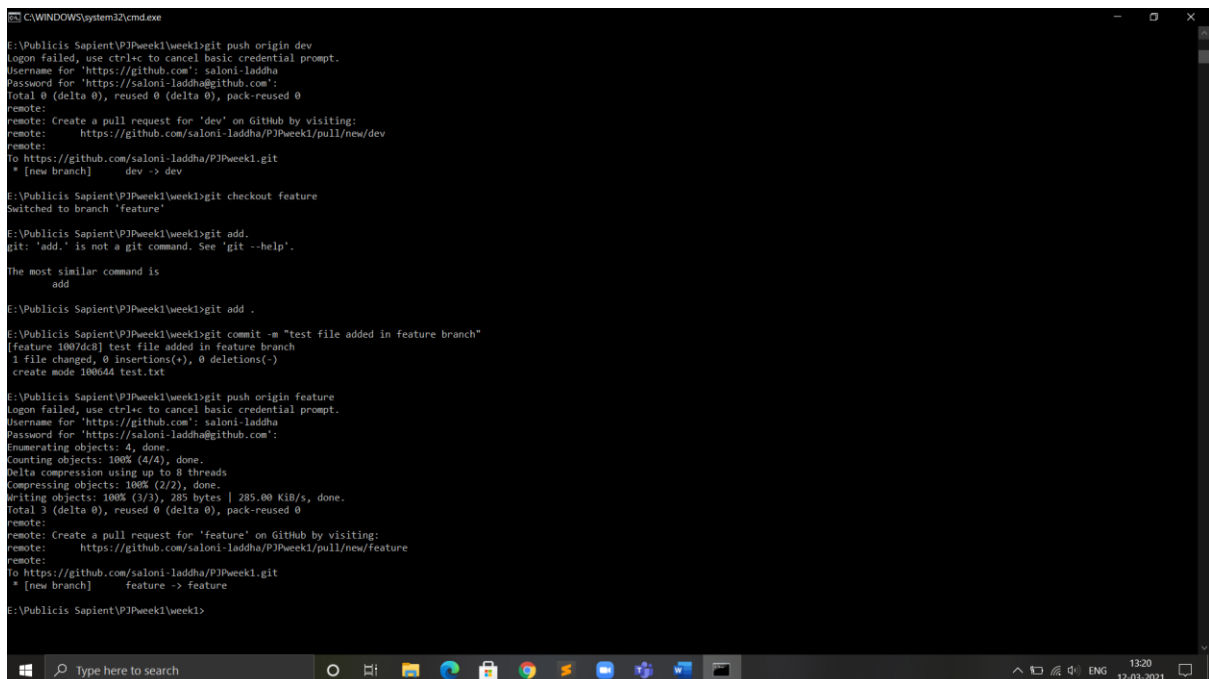## iii. Create local branches

iv. Add files/folders

v. Create remote branches

vi. Check in, stage, commit, push changes into feature branch

vii. Promote code from feature branch to dev branch via pull requests

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.

base: feature ← compare: master ✓ Able to merge. These branches can be automatically merged.

files added

Write | Preview

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Create pull request ▾

ⓘ Remember, contributions to this repository should follow our GitHub Community Guidelines.

Reviewers
No reviews

Assignees
No one—assign yourself

Labels
None yet

Projects
None yet

Milestone
No milestone

Linked issues

---

files added #2

⇄ Open saloni-laddha wants to merge 1 commit into feature from master

Conversation 0 | Commits 1 | Checks 0 | Files changed 1

+2 −0

saloni-laddha commented now   Owner ···

No description provided.

files added   2109062

Add more commits by pushing to the master branch on saloni-laddha/PJPweek1.

Continuous integration has not been set up
GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.

✓ This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request ▾ You can also open this in GitHub Desktop or view command line instructions.

Edit | Open with ▾

Reviewers
No reviews
Still in progress? Convert to draft

Assignees
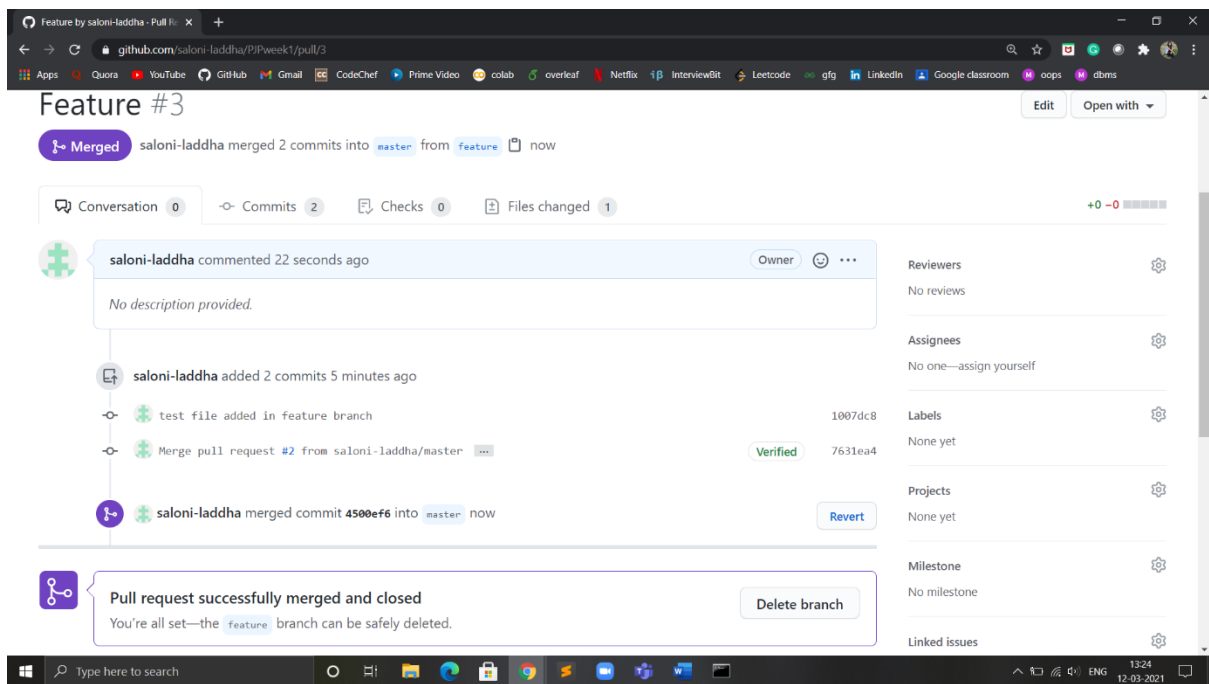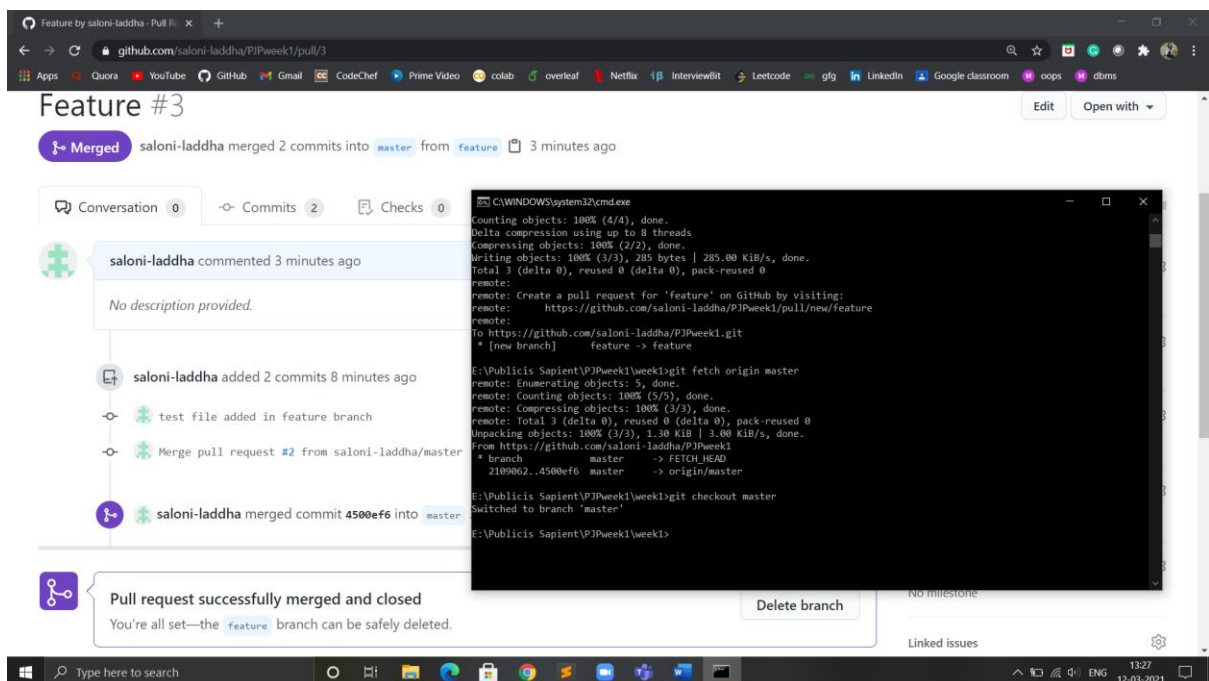No one—assign yourself

Labels
None yet

Projects
None yet

Milestone
No milestone

viii. Checkout the latest code from remote branch to local branch

ix. Explore difference between checkout and pull

Checkout moves the head to a specific branch or commit (if checkout is done by commit id).

Pull means adding the changes to the local folders / repo that are present in the remote repository.
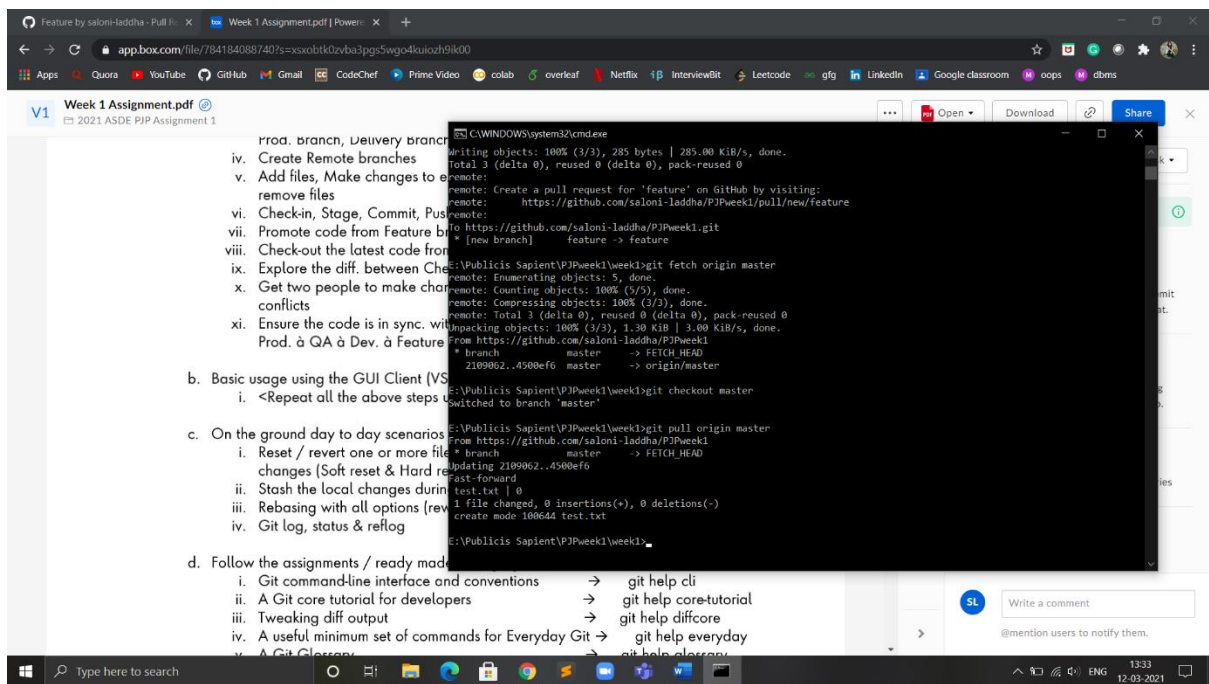
x. Get two people to make changes to the same file, check in and handle merge conflicts

Created 2 branches first: p1 and p2.

P1 and p2 both changed files and made commits which results in conflict.

```
E:\Publicis Sapient\PJPweek1\week1>git checkout feature
Switched to branch 'feature'

E:\Publicis Sapient\PJPweek1\week1>git merge p1
Updating 1007dc8..589c5ac
Fast-forward
 folder1/file2.txt | 2 ++
 test.txt          | 1 +
 2 files changed, 3 insertions(+)
 create mode 100644 folder1/file2.txt

E:\Publicis Sapient\PJPweek1\week1>git merge p2
Auto-merging test.txt
CONFLICT (content): Merge conflict in test.txt
Automatic merge failed; fix conflicts and then commit the result.

E:\Publicis Sapient\PJPweek1\week1>
```
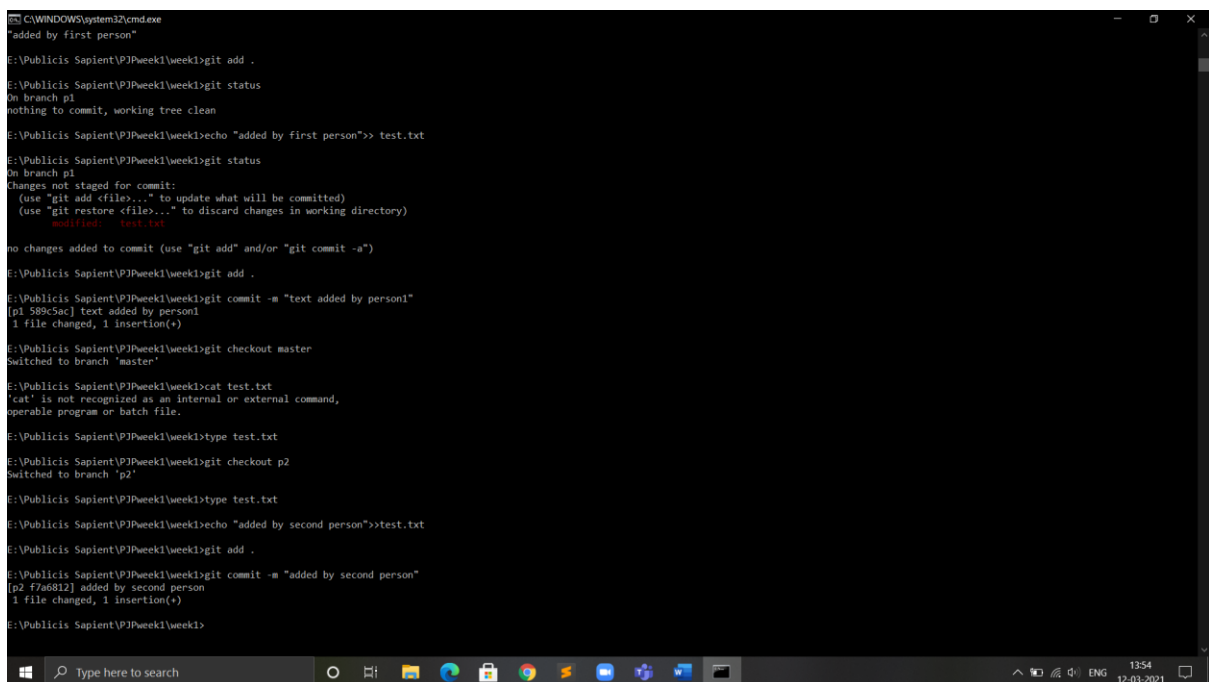
Conflict arises. Now merging them.

```
E:\Publicis Sapient\PJPweek1\week1>git status
On branch feature
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
        both modified:   test.txt

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

E:\Publicis Sapient\PJPweek1\week1>git add .

E:\Publicis Sapient\PJPweek1\week1>git commit -m "merging files"
[feature af55ad5] merging files

E:\Publicis Sapient\PJPweek1\week1>type test.txt
<<<<<<< HEAD
"added by first person"
=======
"added by second person"
>>>>>>> p2

E:\Publicis Sapient\PJPweek1\week1>
```

xi. Ensure code is in sync with the latest changes across all branches

```
E:\Publicis Sapient\PJPweek1\week1>git checkout master
Switched to branch 'master'

E:\Publicis Sapient\PJPweek1\week1>git merge feature
Updating 4500ef6..af55ad5
Fast-forward
 test.txt | 5 +++++
 1 file changed, 5 insertions(+)

E:\Publicis Sapient\PJPweek1\week1>git push origin master
Logon failed, use ctrl+c to cancel basic credential prompt.
Username for 'https://github.com': saloni-laddha
Password for 'https://saloni-laddha@github.com':
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (9/9), 778 bytes | 389.00 KiB/s, done.
Total 9 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
To https://github.com/saloni-laddha/PJPweek1.git
   4500ef6..af55ad5  master -> master

E:\Publicis Sapient\PJPweek1\week1>
```
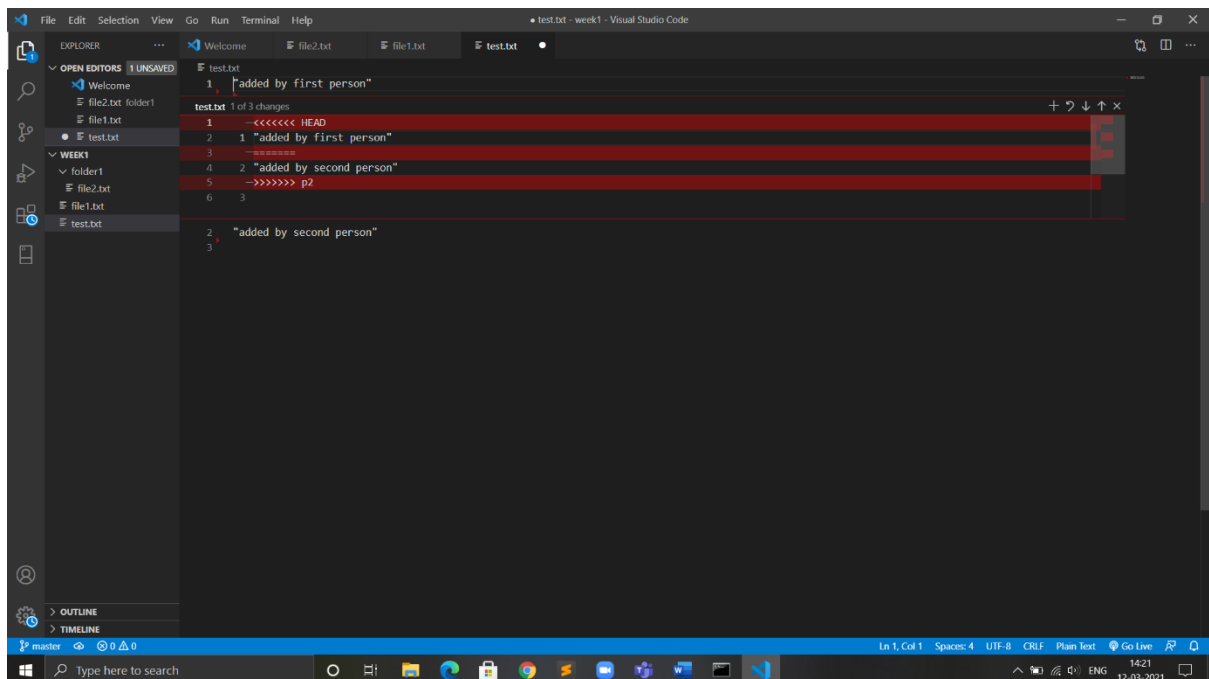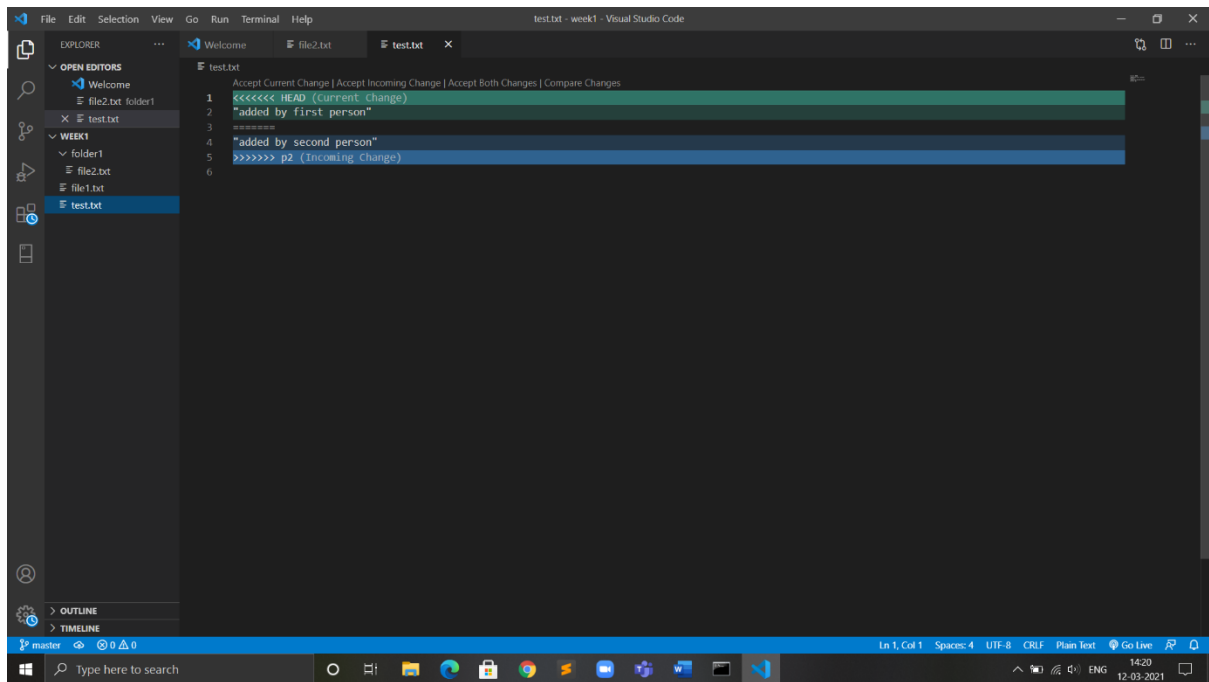
B) Basic usage using the GUI client

Repeat the above steps using a GUI (VS code/GitHub client)

VS code has different extensions for stage, commit, merging and pushing to see changes done by each person

C) On ground day-to-day scenarios

i. Soft reset and hard reset

Soft reset: Reset the head to specific state with putting the changes staged. Does not touch the index file or working tree.

Git log before and after soft reset:

```
E:\Publicis Sapient\PJPweek1\week1>git log --oneline
af55ad5 (HEAD -> master, origin/master, feature) merging files
f7a6812 (p2) added by second person
589c5ac (p1) text added by person1
4500ef6 Merge pull request #3 from saloni-laddha/feature
7631ea4 Merge pull request #2 from saloni-laddha/master
1007dc8 (origin/feature) test file added in feature branch
2109062 files added
f6143b9 (origin/dev, prod, dev, delivery, QA) initial commit

E:\Publicis Sapient\PJPweek1\week1>echo "another change">>test.txt

E:\Publicis Sapient\PJPweek1\week1>type test.txt
"added by first person"
"added by second person"
"another change"

E:\Publicis Sapient\PJPweek1\week1>git log --oneline
af55ad5 (HEAD -> master, origin/master, feature) merging files
f7a6812 (p2) added by second person
589c5ac (p1) text added by person1
4500ef6 Merge pull request #3 from saloni-laddha/feature
7631ea4 Merge pull request #2 from saloni-laddha/master
1007dc8 (origin/feature) test file added in feature branch
2109062 files added
f6143b9 (origin/dev, prod, dev, delivery, QA) initial commit
```

Hard reset: Reset the head to specific state while getting rid of changed done.

Changes in files are also gone in hard reset.

Git log before and after hard reset:

```
E:\Publicis Sapient\PJPweek1\week1>echo "hard reset done">>test.txt

E:\Publicis Sapient\PJPweek1\week1>type test.txt
"added by first person"
"added by second person"
"another change"
"hard reset done"

E:\Publicis Sapient\PJPweek1\week1>git log --oneline
af55ad5 (HEAD -> master, origin/master, feature) merging files
f7a6812 (p2) added by second person
589c5ac (p1) text added by person1
4500ef6 Merge pull request #3 from saloni-laddha/feature
7631ea4 Merge pull request #2 from saloni-laddha/master
1007dc8 (origin/feature) test file added in feature branch
2109062 files added
f6143b9 (origin/dev, prod, dev, delivery, QA) initial commit

E:\Publicis Sapient\PJPweek1\week1>git reset --hard 7631ea4
HEAD is now at 7631ea4 Merge pull request #2 from saloni-laddha/master

E:\Publicis Sapient\PJPweek1\week1>git log --oneline
7631ea4 (HEAD -> master) Merge pull request #2 from saloni-laddha/master
1007dc8 (origin/feature) test file added in feature branch
2109062 files added
f6143b9 (origin/dev, prod, dev, delivery, QA) initial commit

E:\Publicis Sapient\PJPweek1\week1>type test.txt

E:\Publicis Sapient\PJPweek1\week1>
```
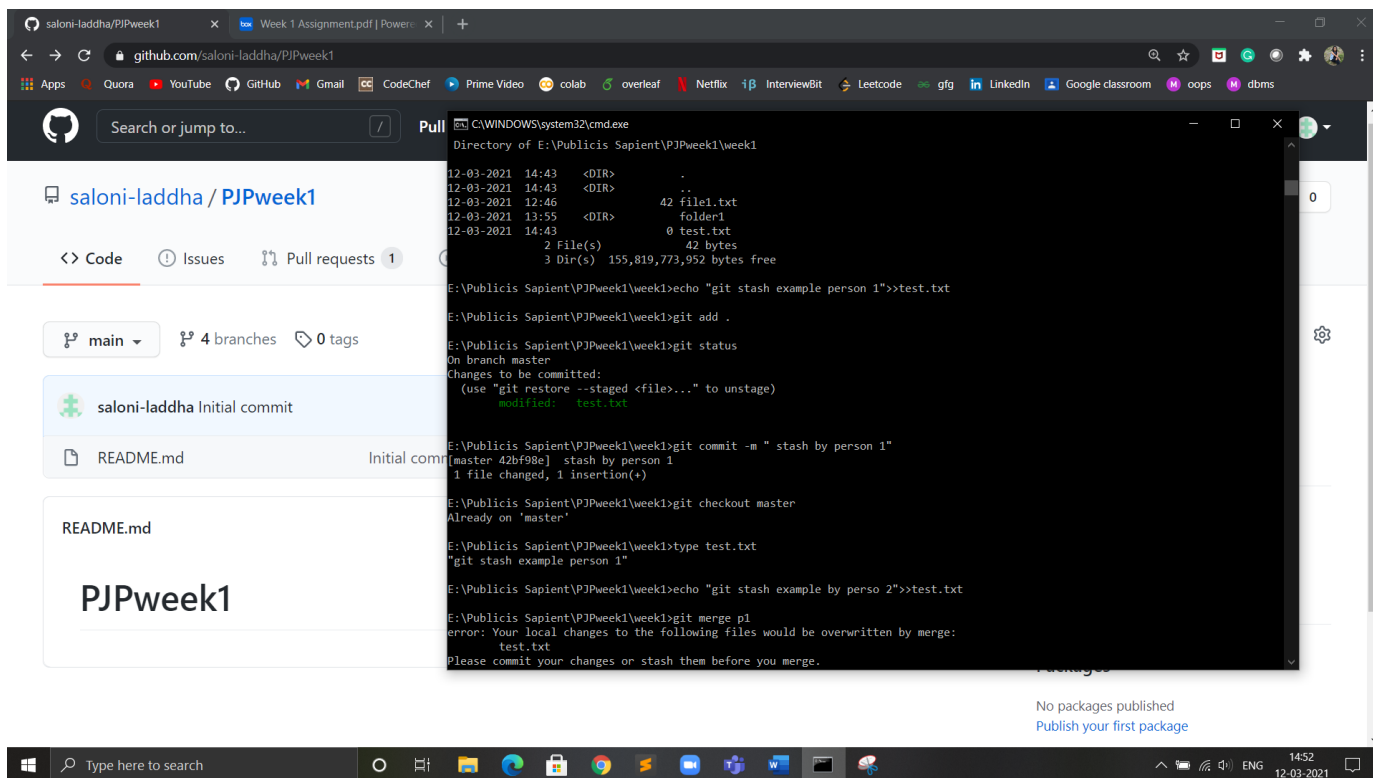
ii. Stash the local changes during merge conflicts

Let person 1 make the changes and commit it. Now master makes changes and tries to merge them.

Git stash:

iii. Rebasing with all options

Git log before and after rebase:

iv. Git log, status and reflog

Git log: shows the current head and its ancestors.

```
E:\Publicis Sapient\PJPweek1\week1>git log
commit e46859d90ab8dddd59a788b4460c75afede321fc (HEAD -> p1)
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:   Fri Mar 12 14:59:21 2021 +0530

    new text added in file2

commit 94c7286dc07535e92d7358c8c5f0c377cbc2dbe1 (master)
Merge: 42bf98e 589c5ac
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:   Fri Mar 12 14:57:56 2021 +0530

    new text added

commit 42bf98e8db3fa267915d7847ded7cb0941227b28
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:   Fri Mar 12 14:50:01 2021 +0530

     stash by person 1

commit 589c5acad47e33957ef86efabb1cd85b4589ce0c
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:   Fri Mar 12 13:46:22 2021 +0530

    text added by person1

commit 4500ef671c9e0adb2df4993e39d95106bf8ea5fc
Merge: 2109062 7631ea4
Author: Saloni Laddha <saloniladdha8@gmail.com>
Date:   Fri Mar 12 13:24:22 2021 +0530

    Merge pull request #3 from saloni-laddha/feature
:...skipping...
commit e46859d90ab8dddd59a788b4460c75afede321fc (HEAD -> p1)
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:   Fri Mar 12 14:59:21 2021 +0530

    new text added in file2

commit 94c7286dc07535e92d7358c8c5f0c377cbc2dbe1 (master)
Merge: 42bf98e 589c5ac
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:   Fri Mar 12 14:57:56 2021 +0530

    new text added

commit 42bf98e8db3fa267915d7847ded7cb0941227b28
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:   Fri Mar 12 14:50:01 2021 +0530
```

```
commit 42bf98e8db3fa267915d7847ded7cb0941227b28
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:    Fri Mar 12 14:50:01 2021 +0530

    stash by person 1

commit 589c5acad47e33957ef86efabb1cd85b4589ce0c
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:    Fri Mar 12 13:46:22 2021 +0530

    text added by person1

commit 4500ef671c9e0adb2df4993e39d95106bf8ea5fc
Merge: 2109062 7631ea4
Author: Saloni Laddha <saloniladdha8@gmail.com>
Date:    Fri Mar 12 13:24:22 2021 +0530

    Merge pull request #3 from saloni-laddha/feature

    Feature

commit 7631ea47da873af98fd977665a065c4c243e15ca
Merge: 1007dc8 2109062
Author: Saloni Laddha <saloniladdha8@gmail.com>
Date:    Fri Mar 12 13:22:11 2021 +0530

    Merge pull request #2 from saloni-laddha/master

    files added

commit 1007dc8fc4a117b2d45177acafbf0e7eb04647bc (origin/feature)
Author: saloni-laddha <saloniladdha8@gmail.com>
Date:    Fri Mar 12 13:18:52 2021 +0530

    test file added in feature branch

:
```

Git status: displays the state of the working directory and the staging area. It lets you see which changes have been staged and which files aren't being tracked by Git.

```
E:\Publicis Sapient\PJPweek1\week1>git status
On branch p1
nothing to commit, working tree clean

E:\Publicis Sapient\PJPweek1\week1>
```

Git reflog: This command is to manage the information recorded in it. Basically every action you perform inside of Git where data is stored, you can find it inside of the reflog.

```
commit 4500ef671c9e0adb2df4993e39d95106bf8ea5fc
Merge: 2109062 7631ea4
Author: Saloni Laddha <saloniladdha8@gmail.com>
Date:    Fri Mar 12 13:24:22 2021 +0530

    Merge pull request #3 from saloni-laddha/feature

E:\Publicis Sapient\PJPweek1\week1>git reflog
e46859d (HEAD -> p1) HEAD@{0}: rebase (finish): returning to refs/heads/p1
e46859d (HEAD -> p1) HEAD@{1}: rebase (pick): new text added in file2
94c7286 (master) HEAD@{2}: rebase (start): checkout master
4bb8aac HEAD@{3}: commit: new text added in file2
589c5ac HEAD@{4}: checkout: moving from master to p1
94c7286 (master) HEAD@{5}: commit (merge): new text added
42bf98e HEAD@{6}: reset: moving to HEAD
42bf98e HEAD@{7}: checkout: moving from master to master
42bf98e HEAD@{8}: commit: stash by person 1
7631ea4 HEAD@{9}: reset: moving to 7631ea4
af55ad5 (origin/master, feature) HEAD@{10}: reset: moving to HEAD
af55ad5 (origin/master, feature) HEAD@{11}: reset: moving to af55ad5
af55ad5 (origin/master, feature) HEAD@{12}: reset: moving to HEAD
af55ad5 (origin/master, feature) HEAD@{13}: merge feature: Fast-forward
4500ef6 HEAD@{14}: checkout: moving from feature to master
af55ad5 (origin/master, feature) HEAD@{15}: commit (merge): merging files
589c5ac HEAD@{16}: merge p1: Fast-forward
1007dc8 (origin/feature) HEAD@{17}: checkout: moving from p2 to feature
f7a6812 (p2) HEAD@{18}: commit: added by second person
4500ef6 HEAD@{19}: checkout: moving from master to p2
4500ef6 HEAD@{20}: checkout: moving from p1 to master
589c5ac HEAD@{21}: commit: text added by person1
4500ef6 HEAD@{22}: checkout: moving from master to p1
4500ef6 HEAD@{23}: pull origin master: Fast-forward
2109062 HEAD@{24}: checkout: moving from feature to master
1007dc8 (origin/feature) HEAD@{25}: commit: test file added in feature branch
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{26}: checkout: moving from master to feature
2109062 HEAD@{27}: commit: files added
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{28}: checkout: moving from delivery to master
:...skipping...
e46859d (HEAD -> p1) HEAD@{0}: rebase (finish): returning to refs/heads/p1
e46859d (HEAD -> p1) HEAD@{1}: rebase (pick): new text added in file2
94c7286 (master) HEAD@{2}: rebase (start): checkout master
4bb8aac HEAD@{3}: commit: new text added in file2
589c5ac HEAD@{4}: checkout: moving from master to p1
94c7286 (master) HEAD@{5}: commit (merge): new text added
42bf98e HEAD@{6}: reset: moving to HEAD
42bf98e HEAD@{7}: checkout: moving from master to master
42bf98e HEAD@{8}: commit: stash by person 1
7631ea4 HEAD@{9}: reset: moving to 7631ea4
```

```
1007dc8 (origin/feature) HEAD@{17}: checkout: moving from p2 to feature
f7a6812 (p2) HEAD@{18}: commit: added by second person
4500ef6 HEAD@{19}: checkout: moving from master to p2
4500ef6 HEAD@{20}: checkout: moving from p1 to master
589c5ac HEAD@{21}: commit: text added by person1
4500ef6 HEAD@{22}: checkout: moving from master to p1
4500ef6 HEAD@{23}: pull origin master: Fast-forward
2109062 HEAD@{24}: checkout: moving from feature to master
1007dc8 (origin/feature) HEAD@{25}: commit: test file added in feature branch
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{26}: checkout: moving from master to feature
2109062 HEAD@{27}: commit: files added
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{28}: checkout: moving from delivery to master
:...skipping...
e46859d (HEAD -> p1) HEAD@{0}: rebase (finish): returning to refs/heads/p1
e46859d (HEAD -> p1) HEAD@{1}: rebase (pick): new text added in file2
94c7286 (master) HEAD@{2}: rebase (start): checkout master
4bb8aac HEAD@{3}: commit: new text added in file2
589c5ac HEAD@{4}: checkout: moving from master to p1
94c7286 (master) HEAD@{5}: commit (merge): new text added
42bf98e HEAD@{6}: reset: moving to HEAD
42bf98e HEAD@{7}: checkout: moving from master to master
42bf98e HEAD@{8}: commit: stash by person 1
7631ea4 HEAD@{9}: reset: moving to 7631ea4
af55ad5 (origin/master, feature) HEAD@{10}: reset: moving to HEAD
af55ad5 (origin/master, feature) HEAD@{11}: reset: moving to af55ad5
af55ad5 (origin/master, feature) HEAD@{12}: reset: moving to HEAD
af55ad5 (origin/master, feature) HEAD@{13}: merge feature: Fast-forward
4500ef6 HEAD@{14}: checkout: moving from feature to master
af55ad5 (origin/master, feature) HEAD@{15}: commit (merge): merging files
589c5ac HEAD@{16}: merge p1: Fast-forward
1007dc8 (origin/feature) HEAD@{17}: checkout: moving from p2 to feature
f7a6812 (p2) HEAD@{18}: commit: added by second person
4500ef6 HEAD@{19}: checkout: moving from master to p2
4500ef6 HEAD@{20}: checkout: moving from p1 to master
589c5ac HEAD@{21}: commit: text added by person1
4500ef6 HEAD@{22}: checkout: moving from master to p1
4500ef6 HEAD@{23}: pull origin master: Fast-forward
2109062 HEAD@{24}: checkout: moving from feature to master
1007dc8 (origin/feature) HEAD@{25}: commit: test file added in feature branch
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{26}: checkout: moving from master to feature
2109062 HEAD@{27}: commit: files added
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{28}: checkout: moving from delivery to master
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{29}: checkout: moving from prod to delivery
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{30}: checkout: moving from QA to prod
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{31}: checkout: moving from dev to QA
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{32}: checkout: moving from feature to dev
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{33}: checkout: moving from master to feature
f6143b9 (origin/dev, prod, dev, delivery, QA) HEAD@{34}: commit (initial): initial commit
~
~
```