

1. Set your username and email in git config

```
ruchigautam@Practos-MacBook-Pro project % git config --list
credential.helper=osxkeychain
init.defaultbranch=main
user.name=ruchgautammuski
user.email=ruchi.gautam@practo.com
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
core.ignorecase=true
core.precomposeunicode=true
remote.origin.url=https://github.com/ruchgautammuski/test.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.master.remote=origin
branch.master.merge=refs/heads/master
```

2. Create a new branch named "feature-branch" and switch to it.

```
ruchigautam@Practos-MacBook-Pro project % git checkout -b feature-branch
Switched to a new branch 'feature-branch'
```

3. List all branches in the repository.

```
ruchigautam@Practos-MacBook-Pro project % git branch
  conflict-branch
* feature-branch
  feature1
  main
  master
```

4. Delete the branch "feature-branch"

```
ruchigautam@Practos-MacBook-Pro project % git branch -d feature-branch
Deleted branch feature-branch (was d572f51).
```

5. How do you undo the last commit

```
ruchigautam@Practos-MacBook-Pro project % git reset HEAD^
ruchigautam@Practos-MacBook-Pro project % git reset --hard HEAD~1
HEAD is now at 3ef2d10 first commit
```

6. Create a new branch names “conflict-branch”

```
ruchigautam@Practos-MacBook-Pro project % git checkout -b conflict-branch  
Switched to a new branch 'conflict-branch'
```

7. Create a another branch named “feature1”

```
ruchigautam@Practos-MacBook-Pro project % git checkout -b feature1  
Switched to a new branch 'feature1'  
ruchigautam@Practos-MacBook-Pro project % touch index.js  
ruchigautam@Practos-MacBook-Pro project % git add .  
ruchigautam@Practos-MacBook-Pro project % git commit -m"changes in feature 1 branch"
```

8. Make some changes in to feature1 branch

```
ruchigautam@Practos-MacBook-Pro project % git checkout -b feature1  
Switched to a new branch 'feature1'  
ruchigautam@Practos-MacBook-Pro project % touch index.js  
ruchigautam@Practos-MacBook-Pro project % git add .  
ruchigautam@Practos-MacBook-Pro project % git commit -m"changes in feature 1 branch"
```

9. Merge “feature1” branch into main branch

```
ruchigautam@Practos-MacBook-Pro project % git checkout -b conflict-branch  
Switched to a new branch 'conflict-branch'
```

10. Make changes in “conflict-branch”, in the same file and line that you had made changes in feature1

```

ruchigautam@Practos-MacBook-Pro project % git branch
  conflict-branch
  feature1
* main
  master
ruchigautam@Practos-MacBook-Pro project % git checkout feature1
Switched to branch 'feature1'
Your branch is up to date with 'origin/feature1'.
ruchigautam@Practos-MacBook-Pro project % ls
index.css      index.js

```

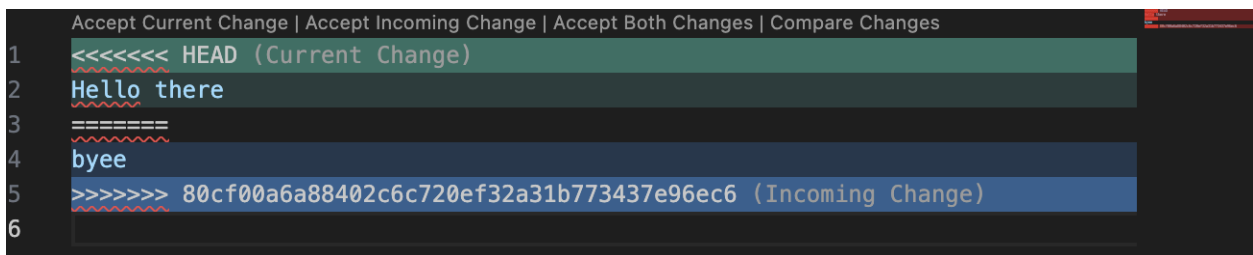
11. Merge master into conflict-branch [Attach screenshot of terminal & file]

```

❗ ruchigautam@Practos-MacBook-Pro project % git pull origin master
From https://github.com/ruchigautammuski/test
 * branch                master      -> FETCH_HEAD
Auto-merging Desktop/project/index.js
CONFLICT (add/add): Merge conflict in Desktop/project/index.js
Automatic merge failed; fix conflicts and then commit the result.

```

12. Resolve merge conflicts



```

Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
1 <<<<<<< HEAD (Current Change)
2 Hello there
3 =====
4 bye
5 >>>>>> 80cf00a6a88402c6c720ef32a31b773437e96ec6 (Incoming Change)
6

```

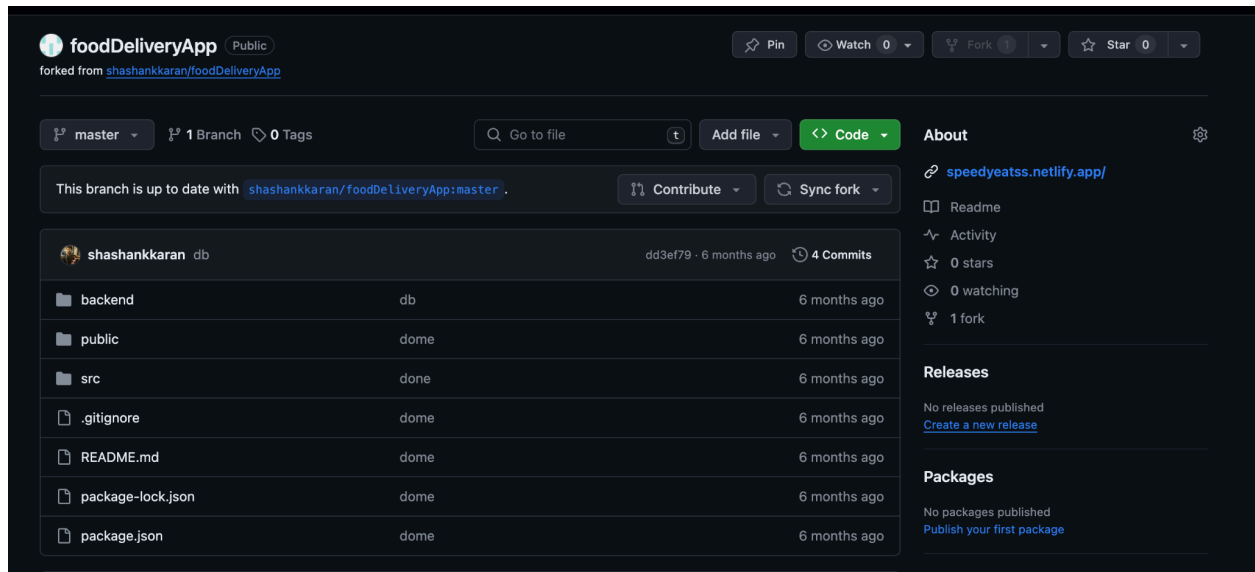
13. Add a remote named "origin" pointing to a GitHub repository.

```

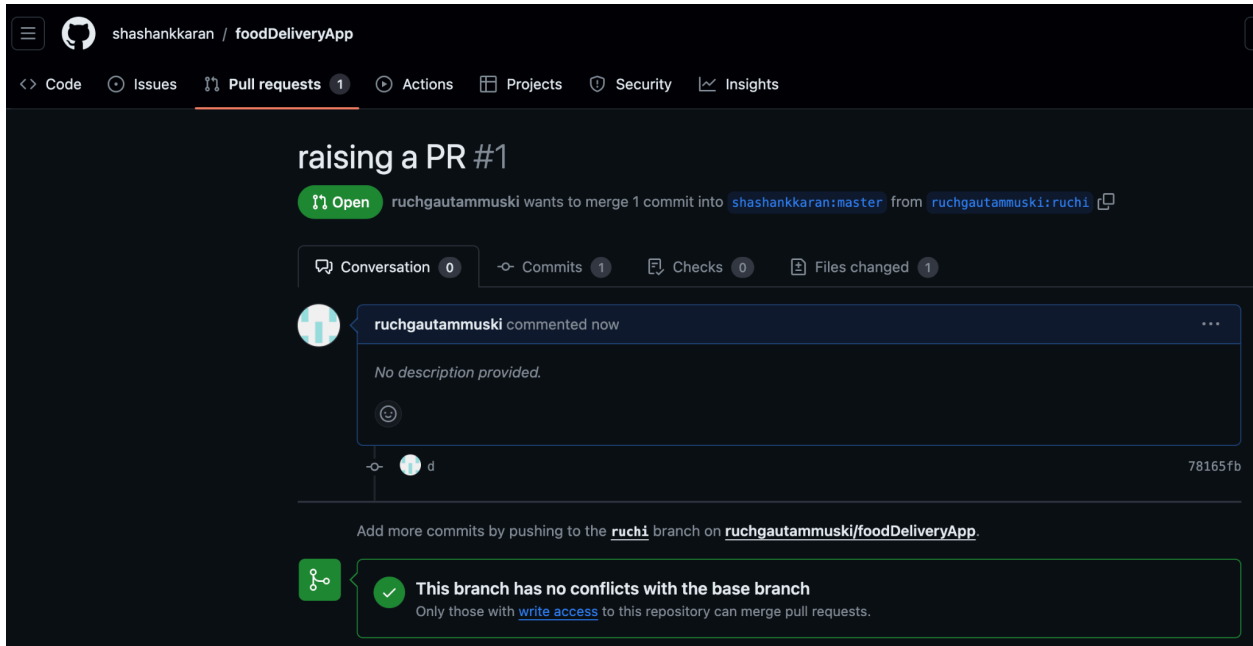
● ruchigautam@Practos-MacBook-Pro project % git remote add origin https://github.com/ruchigautammuski/test2.git
○ ruchigautam@Practos-MacBook-Pro project %

```

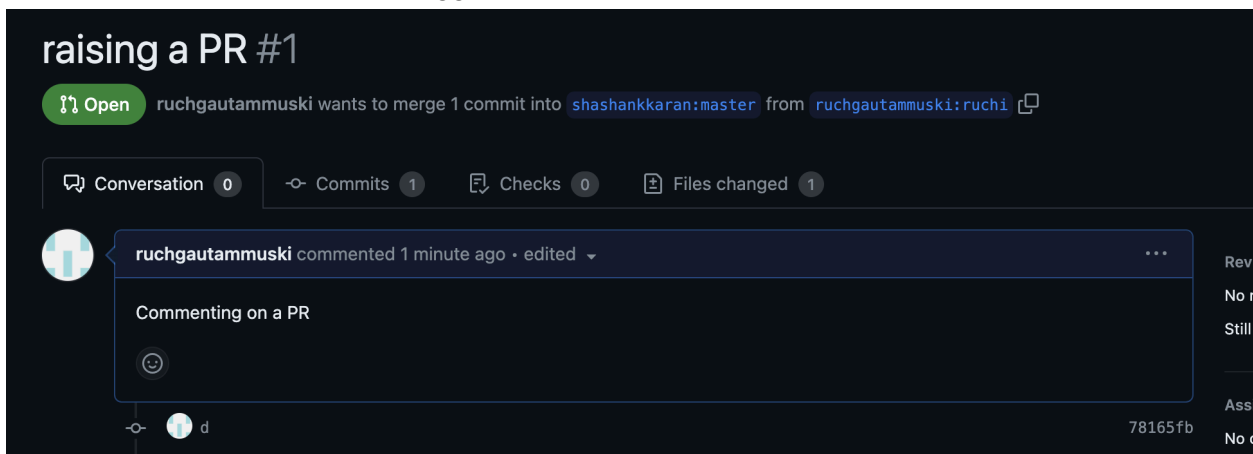
14. Fork a repository on GitHub and clone it to your local machine.



15. Create a new branch on your fork, make changes, and open a pull request to the original repository.



## 16. Comment on a PR and suggest improvements



## 17. Create a Git alias for the command `git log --oneline` named `gitlol`.

```
ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git config --global alias.gitlol "log --oneline"
```

```
"
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git gitlog
78165fb (HEAD -> ruchi, origin/ruchi) d
dd3ef79 (origin/master, origin/HEAD, master) db
18729a3 done
80da760 dome
35e27df dome
```

18. Create a pre-commit hook

```
bin
bin > No Selection

1 # Check if this is the initial commit
2 if git rev-parse --verify HEAD >/dev/null 2>&1
3 then
4     echo "pre-commit: About to create a new commit..."
5     against=HEAD
6 else
7     echo "pre-commit: About to create the first commit.."
8     against=4b825dc642cb6eb9a060e54bf8d69288fbee4904
9 fi
10
11 # Use git diff-index to check for whitespace errors
12 echo "pre-commit: Testing for whitespace errors..."
13 if ! git diff-index --check --cached $against
14 then
15     echo "pre-commit: Aborting commit due to whitespace"
16     exit 1
17 else
18     echo "pre-commit: No whitespace errors :)"
19     exit 0
20 fi
```

19. You have made local changes in your branch, but you need to switch to another branch urgently without committing. How would you handle this situation?

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git stash
Saved working directory and index state WIP on ruch: 78165fb d
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

20. You accidentally deleted a file in your local repository. How do you restore it using Git?

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git status
On branch ruch
Your branch is up to date with 'origin/ruchi'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    src/screens/Home.js

no changes added to commit (use "git add" and/or "git commit -a")
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git reset --hard origin/master
HEAD is now at dd3ef79 db
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

21. You have committed changes to your branch but forgot to include a file. How do you add the file to the last commit without creating a new commit?



```

Already up to date.
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git status
On branch ruch
Your branch is up to date with 'origin/ruchi'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    src/screens/Home.js

no changes added to commit (use "git add" and/or "git commit -a")
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git reset --hard origin/master
HEAD is now at dd3ef79 db
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git status
On branch ruch
Your branch is behind 'origin/ruchi' by 1 commit, and can be fast-forwarded.
  (use "git pull" to update your local branch)

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

nothing added to commit but untracked files present (use "git add" to track)
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git add src/,ecc
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git commit -m"file added"
[ruchi cf853a5] file added
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 src/,ecc
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git push'
quote>
⊗ ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git push
To https://github.com/ruchgautammuski/foodDeliveryApp.git
 ! [rejected]        ruch -> ruch (non-fast-forward)
error: failed to push some refs to 'https://github.com/ruchgautammuski/foodDeliveryApp.git'
hint: Updates were rejected because the tip of your current branch is behind
hint: its remote counterpart. Integrate the remote changes (e.g.
hint: 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
⊗ ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git pull
hint: You have divergent branches and need to specify how to reconcile them.
hint: You can do so by running one of the following commands sometime before
hint: your next pull:
hint:
hint:   git config pull.rebase false  # merge
hint:   git config pull.rebase true   # rebase
hint:   git config pull.ff only        # fast-forward only
hint:
hint: You can replace "git config" with "git config --global" to set a default
hint: preference for all repositories. You can also pass --rebase, --no-rebase,
hint: or --ff-only on the command line to override the configured default per
hint: invocation.
fatal: Need to specify how to reconcile divergent branches.
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git config pull.rebase false
⊗ ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git push
To https://github.com/ruchgautammuski/foodDeliveryApp.git
 ! [rejected]        ruch -> ruch (non-fast-forward)
error: failed to push some refs to 'https://github.com/ruchgautammuski/foodDeliveryApp.git'
hint: Updates were rejected because the tip of your current branch is behind
hint: its remote counterpart. Integrate the remote changes (e.g.
hint: 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git push -f
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 328 bytes | 328.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/ruchgautammuski/foodDeliveryApp.git
 + 78165fb...cf853a5 ruch -> ruch (forced update)
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp % █

```



22. You want to discard all changes in your working directory and revert to the last commit. What Git command would you use?

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git reset --hard origin/master
HEAD is now at dd3ef79 db
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

23. You need to view a specific commit's changes. What Git command can be used to show the changes introduced by a particular commit?

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git log
commit dd3ef797b5fdc30677874178b3dab88431a6f99c (HEAD -> ruchi, origin/master, origin/HEAD, master)
Author: Shashank Karan <shashankkaran999@gmail.com>
Date: Thu Jul 20 19:56:23 2023 +0530

    db

commit 18729a33f3587f9ef4b0e25c51cc5d89738eadb1
Author: Shashank Karan <shashankkaran999@gmail.com>
Date: Thu Jul 20 18:41:16 2023 +0530

    done

commit 80da7604aa04bbc792b83094812d927a73fa3ba9
Author: Shashank Karan <shashankkaran999@gmail.com>
Date: Thu Jul 20 18:37:27 2023 +0530

    dome

commit 35e27dfe26f87fb078c8b89544266c2cd70992cb
Author: Shashank Karan <shashankkaran999@gmail.com>
Date: Thu Jul 20 18:33:46 2023 +0530

    dome
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

24. You want to change a commit message, after you have already committed, how do you do so?

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git commit --amend -m "amending commit msg"
[ruchi 2fb52d6] amending commit msg
Author: Shashank Karan <shashankkaran999@gmail.com>
Date: Thu Jul 20 19:56:23 2023 +0530
2 files changed, 2 insertions(+), 2 deletions(-)
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

25. Your colleague has made changes in their branch, and you want to incorporate those changes into your branch without merging. How do you achieve this?

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git commit --amend -m "amending commit msg"
[ruchi 2fb52d6] amending commit msg
Author: Shashank Karan <shashankkaran999@gmail.com>
Date: Thu Jul 20 19:56:23 2023 +0530
2 files changed, 2 insertions(+), 2 deletions(-)
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

26. You've made several commits on a branch, but you want to club them into a single commit before pushing to the remote repository. How would you do that?

```
Ⓢ ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git cherry-pick 18729a33f3587f9ef4b0e25c51cc5d89738eadb1
Auto-merging backend/index.js
CONFLICT (content): Merge conflict in backend/index.js
error: could not apply 18729a3... done
hint: After resolving the conflicts, mark them with
hint: "git add/rm <pathspec>", then run
hint: "git cherry-pick --continue".
hint: You can instead skip this commit with "git cherry-pick --skip".
hint: To abort and get back to the state before "git cherry-pick",
hint: run "git cherry-pick --abort".
```

27. You accidentally staged a file that you don't want to commit. How do you unstage it?

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git status
On branch ruch1
Your branch and 'origin/ruchi' have diverged,
and have 1 and 2 different commits each, respectively.
(use "git pull" to merge the remote branch into yours)

You are currently cherry-picking commit 18729a3.
(all conflicts fixed: run "git cherry-pick --continue")
(use "git cherry-pick --skip" to skip this patch)
(use "git cherry-pick --abort" to cancel the cherry-pick operation)

Changes to be committed:
  modified:   backend/index.js
  modified:   src/screens/Home.js

● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git add src/screens/Home.js
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git restore --staged src/screens/Home.js
```

```
# See https://help.github.com/articles/ignoring-files/

# dependencies
node_modules
/.pnp
.pnp.js
/node_modules
# testing
/coverage

# production
/build

# misc
.DS_Store
.env.local
.env.development.local
.env.test.local
.env.production.local

npm-debug.log*
yarn-debug.log*
yarn-error.log*

*.yml
config/*
~
~
~
~
```

28. You don't want to commit files that have .yml in the end, and also files inside folder config. How do you do that?

```

modified:   src/screens/Home.js

ruchigautam@Practos-MacBook-Pro foodDeliveryApp % touch .gitignore
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % vim .gitignore
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp % █

```

29. You want to see a list of all the files changed in the last commit. What Git command would you use?

```

● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git diff --name-only HEAD^
.gitignore
backend/db.js
backend/index.js
src/screens/Home.js
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp % █

```

30. You realize that your local branch is outdated, and you want to fetch the latest changes from the remote repository. How do you do this without merging?

```

● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git remote add upstream https://github.com/shashankkaran/foodDeliveryApp.git
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git pull upstream master
From https://github.com/shashankkaran/foodDeliveryApp
 * branch      master      -> FETCH_HEAD
 * [new branch] master      -> upstream/master
fatal: You have not concluded your cherry-pick (CHERRY_PICK_HEAD exists).
Please, commit your changes before you merge.
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp % █

```

31. You accidentally deleted a branch. How do you recover it?

```

● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git branch -D temp
Deleted branch temp (was 2fb52d6).
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git reflog

dd3ef79 (HEAD -> master, upstream/master, origin/master, origin/HEAD) HEAD@{0}: checkout: moving from temp to master
2fb52d6 (ruchi) HEAD@{1}: checkout: moving from ruchi to temp
2fb52d6 (ruchi) HEAD@{2}: commit (amend): amending commit msg
dd3ef79 (HEAD -> master, upstream/master, origin/master, origin/HEAD) HEAD@{3}: reset: moving to origin/master
cf853a5 (origin/ruchi) HEAD@{4}: commit: file added
dd3ef79 (HEAD -> master, upstream/master, origin/master, origin/HEAD) HEAD@{5}: reset: moving to origin/master
78165fb HEAD@{6}: reset: moving to HEAD
78165fb HEAD@{7}: commit: d
dd3ef79 (HEAD -> master, upstream/master, origin/master, origin/HEAD) HEAD@{8}: checkout: moving from master to ruchi
dd3ef79 (HEAD -> master, upstream/master, origin/master, origin/HEAD) HEAD@{9}: clone: from https://github.com/ruchigautamusk/foodDeliveryApp.git
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git reset --hard HEAD(2)
zsh: unknown file attribute: 2
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git reset --hard HEAD
HEAD is now at dd3ef79 db

```

32. You want to remove untracked files and directories from your working directory. What Git command would you use?

```
HEAD is now at dd3ef79 db
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git clean -fdX
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

33. You have a commit from a feature branch that you want to apply to the main branch without merging the entire feature branch.

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git cherry-pick dd3ef79b5fdc30677874178b3dab88431a6f99c
On branch master
Your branch is up to date with 'origin/master'.

You are currently cherry-picking commit dd3ef79.
(all conflicts fixed: run "git cherry-pick --continue")
(use "git cherry-pick --skip" to skip this patch)
(use "git cherry-pick --abort" to cancel the cherry-pick operation)

nothing to commit, working tree clean
```

34. You mistakenly committed a change to the wrong branch and need to apply that commit to the correct branch.

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git branch new-branch 18729a33f3587f9ef4b0e25c51cc5d89738eadb1
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git checkout new-branch
Switched to branch 'new-branch'
warning: cancelling a cherry picking in progress
```

35. There is a series of commits on a feature branch, but you only want to cherry-pick a specific range of commits.

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git cherry-pick 35e27dfe26f87fb078c8b89544266c2cd70992cb .. 80da7604aa04bbc792b83094812d927a73fa3ba9
error: empty commit set passed
fatal: cherry-pick failed
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

36. You want to clone a GitHub repository onto your local machine, but you only need a specific branch. How can you achieve this?

```
fatal: cherry-pick failed
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % cd ../
● ruchigautam@Practos-MacBook-Pro project % git clone -b master https://github.com/shashankkaran/crypto-gyan.git
Cloning into 'crypto-gyan'...
remote: Enumerating objects: 42, done.
remote: Counting objects: 100% (42/42), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 42 (delta 8), reused 37 (delta 6), pack-reused 0
Receiving objects: 100% (42/42), 389.75 KiB | 493.00 KiB/s, done.
Resolving deltas: 100% (8/8), done.
● ruchigautam@Practos-MacBook-Pro project %
```

37. You've made changes to your local repository and want to push them to your fork on GitHub. What Git commands would you use?

```

● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git branch
  master
* new-branch
  ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git add .
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git commit -m"local changes"
[new-branch 9024a33] local changes
  1 file changed, 1 insertion(+)
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git push origin new-branch
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 438 bytes | 438.00 KiB/s, done.
Total 5 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
remote:
remote: Create a pull request for 'new-branch' on GitHub by visiting:
remote:   https://github.com/ruchgautammuski/foodDeliveryApp/pull/new/new-branch
remote:
To https://github.com/ruchgautammuski/foodDeliveryApp.git
 * [new branch]      new-branch -> new-branch
💡 ruchigautam@Practos-MacBook-Pro foodDeliveryApp %

```

38. You want to create a new branch both locally and on GitHub to work on a new feature. What commands would you use?

```

● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git checkout -b feature
Switched to a new branch 'feature'
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git add .
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git commit -m"new branch feature added"
On branch feature
nothing to commit, working tree clean
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git push
fatal: The current branch feature has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin feature

To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetupRemote' in 'git help config'.
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git push --set-upstream origin feature
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'feature' on GitHub by visiting:
remote:   https://github.com/ruchgautammuski/foodDeliveryApp/pull/new/feature
remote:
To https://github.com/ruchgautammuski/foodDeliveryApp.git
 * [new branch]      feature -> feature
branch 'feature' set up to track 'origin/feature'.
💡 ruchigautam@Practos-MacBook-Pro foodDeliveryApp %

```

39. You want to see the commit history of a GitHub repository. How can you do this using Git commands?

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git log
commit 9024a3311a70c97371da997c2a89f30c5f088064 (HEAD -> feature, origin/new-branch, origin/feature, new-branch)
Author: ruchigautammuski <ruchi.gautam@practo.com>
Date: Wed Jan 24 22:49:52 2024 +0530

    local changes

commit 18729a33f3587f9ef4b0e25c51cc5d89738eadb1
Author: Shashank Karan <shashankkaran999@gmail.com>
Date: Thu Jul 20 18:41:16 2023 +0530

    done

commit 80da7604aa04bbc792b83094812d927a73fa3ba9
Author: Shashank Karan <shashankkaran999@gmail.com>
Date: Thu Jul 20 18:37:27 2023 +0530

    dome

commit 35e27dfe26f87fb078c8b89544266c2cd70992cb
Author: Shashank Karan <shashankkaran999@gmail.com>
Date: Thu Jul 20 18:33:46 2023 +0530

    dome
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

40. You've accidentally committed sensitive information and want to remove the commit from both your local and remote repositories on GitHub. What commands would you use?

```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git reset HEAD^
Unstaged changes after reset:
M   src/screens/Cart.js
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git push -f
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/ruchigautammuski/foodDeliveryApp.git
+ 9024a33...18729a3 feature -> feature (forced update)
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

41. You want to delete a remote branch on GitHub. What Git command would you use?



```
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git branch
* feature
  master
  new-branch
  ruch
● ruchigautam@Practos-MacBook-Pro foodDeliveryApp % git push origin --delete ruch
To https://github.com/ruchgautammuski/foodDeliveryApp.git
- [deleted]      ruch
○ ruchigautam@Practos-MacBook-Pro foodDeliveryApp %
```

42. Create a git repository for all your assignments and upload them in it. Ask your peers to code review it, and you need to code review your peers assignments

<https://github.com/ruchgautammuski/assignment>

43. Create a pull request on any open source library on github, attach the pull request link to the readme file of this project's repository

<https://github.com/shashankkaran/foodDeliveryApp/pull/2>