

Homework: 02

16. Technomakers: Anupam Kumar (11940160), Abdur Rahman Khan (11940020), Ruchit Prakash Saxena (11941040)

Solution of problem 1. .

(a) A git repository can support multiple working trees, allowing you to check out more than one branch at a time. With,

\$ git worktree add <path> <branch> a new working tree is associated with the repository.

(b) \$ git checkout <files(s)... or dir>

Updates the file or directory in the workspace. Does NOT switch branches.

\$ git restore <file> to discard changes in working directory.

\$ git rm <file(s)> Remove a file from the workspace and the index.

\$ git mv <file(s)> Move file in the workspace and the index.

(c) use \$ git add -patch ;filename; (or -p for short), and git will begin to break down your file into "hunks" (portions of the file).

then it will then prompt you with this question:

Stage this hunk [y,n,q,a,d,/j,J,g,s,e,?]?

select an option :

y- stage this hunk for the next commit

n- do not stage this hunk for the next commit

q- quit; do not stage this hunk or any of the remaining hunks

a- stage this hunk and all later hunks in the file

d- do not stage this hunk or any of the later hunks in the file

If the file is not in the repository yet, you can first do `git add -N <filename>`. Afterwards you can go on with `git add -p <filename>`.

and then you can commit a part(hunk) of a file by -

`$ git commit -v` to view your commit while you edit the commit message.

(d) To turn multiple commits in a feature branch into a single commit is to reset the feature branch changes in the master and commit everything again.

Switch to the master branch and make sure you are up to date.

`$ git checkout master`

`$ git fetch` this may be necessary (depending on your git config) to receive updates on origin/master

`$ git pull`

Merge the feature branch into the master branch.

`$ git merge feature branch`

Reset the master branch to origin's state.

`$ git reset origin/master`

Git now considers all changes as unstaged changes.

We can add these changes as one commit.

Adding `.` will also add untracked files.

`$ git add -all`

`$ git commit`