Basic Git commands

1. git init - adding & commiting files / directory
2. git add - $ git add <file or directory name>
   ex - git add css , git add index.html
3. git commit - Record the changes made to files
   each commit has unique ID.

ex $ git commit -m "commit my message hello"

4. git status - It returns the current state of repository
   or if there no changes it will (clean)

ex - $ git status

⑤ git config – it show configured present & we can if you apply --global it'll show all repositories in computer & if you don't apply it'll only show current repository.

→ $ git config <setting> <command>

& $ git config --global user.email "my@gmail.com"
$ git config --global user.name "AayushModi"

---

⑥ git branch – To determine what branch the local repository is on, add a new branch, or dlt branch.

→ $ git branch <branch_name>

to list all local or remote branches $ git branch -a

to delete branch $ git branch -d <branch_name>

---

⑦ git checkout – to start working in different branch (to switch branches)

→ $ git checkout <branch_name>

# create & checkout a new branch with that name
$ git checkout -b <new_branch>

---

⑧ git merge – Integrate branches together

$ git merge <branch_name>

---

⑨ git remote – to connect local repository with remote

# Add remote repository
$ git remote <command> <remote_name> <remote_URL>

# List named remote repo
$ git remote -V

origin

(10) git clone – to copy & download repo in comp

$ git clone <remote_URL>

(11) git pull – pulls the changes from the remote repo → to local comp.

→ $ git pull <branch_name> <remote_URL/remote-name>

(12) git push – sends local commits to remote repo

$ git push <remote_URL_name> <branch>

push all the local branches to repmote repo

$ git push --all

tracked files – files that were in the last snapshot.

Recursive add – git add -A or
git add --all

Renaming & moving files – git mv /command

deleting files – git rm

Glet alias – workflow tool that create short cuts to frequently used

Ex

$ git config --global alias.co checkout

ignore unwanted files

- .gitignore

To visually compare & merge files -

→ DiffMerge

git diff → multi use git command
when executed runs diff
funct^n on git data sources

1/2

Branching & Merging

$ git branch -a → lists existing branches
list local & remote branch

$ git branch mynewbranch → To create new branch
name

$ git branch -a

$ git log --oneline --decorate to see history

$ git checkout master → to go back.

$ git branch -m mynewbranch newbranch
rename

$ git branch -a

$ git branch -d newbranch
to delete

$ git branch -a

Ru el
cd
git status

fast forward Merge

$ git branch - shows all global branches

$ git branch -a -b will create a branch without checking it out

$ git checkout -b title-change

$ git status

$ git commit -am "changing title of HTML file"

$ git log --oneline

$ git checkout master this will take you to master branch

(master)—$ git diff master title-change
(shows differences b/w master & title-change)

$ git difftool (if we prefer to see diff. visually)
master title-change

$ git merge title-change to merge

To create new branch —

$ git checkout -b add-copyright

$ git branch

$ mate simple.html — to edit simple html file

$ git commit -am "adding copyright notice"

$ mate README.me

Rebase —

$ git rebase master
eg.

$ git log --online --decorate --all --graph

Pull with Rebase —

$ git pull origin master

$ git push origin master

Simple stash

$ git stash list

$ gitstash drop

$ git stash apply ← to modify.

$ git stash -u it will include any untracked file that are not been excluded by get ignore

$ git stash pop

$ git commit -am "_____" (express commit)

Managing multiple stashes

$ mate index.html

$ git stash save "index change"

$ mate README

git: - save

$ git stash list ← it show all the saved stash

$ git stash show stash@{1} ← see specific stash

$ git stash apply

$ git stash clear ← to get rid of all stash list

$ mate new.md ← this'll create new file

$ git stash branch newname

Tags (simple / lightweighted)

$ git tag mytag

$ git tag --list

$ git show mytag

$ git tag --delete mytag

$ git tag -a v-1.0

$ git tag --ammend

$ git tag -a v-0-8-alph -f bd35d4c

-force to update tag