git config - -global (define global settings) - 1

git config - -list (check all the configuration settings)-2

mkdir (creates new directory) - 3

cd (changes the active directory) - 4 git

init (initiate git) - 5

git add (to track file) - 6 git rm (to

unstage or remove a file) - 7

git commit (to make commit) - 8

git diff (shows the changes in the tracked files) - 9

git status -10

git push (sending the package to the recipient) -11 git log

(review the history, show the list of commits) -12

git show + hash (lists information about a commit) -13

git remote (add a remote to our repository) -14

: Chapter 3

.git branch : To see the list of branches and the current branch you're working on - 1

. git branch -a : to see the remote branches - 2

. git branch (followed by a branch name) : to create a new branch - 3

. git checkout (followed by a branch name) : to change the active branch - 4

. git checkout -b : to create and checkout to a new branch in a single command - 5

.git branch -m (followed by the new name) : to rename the current branch - 6

. git branch —oneline : shows a compact form of the Git history - 7

. git branch -D (followed by an existing branch name) : to delete that branch - 8

git merge --no-ff : For no-fastforward merge - 9

: Chapter 4

. git clone (followed by the url of the project) : to clone a remote repository - 1

git clone (followed by the url of the project and the new name of the root) : to change - 2

. the root directory name of the repository while cloning

git remote -v : used to display the current remote repositories associated with your local - 3

.Git repository. The -v shows the URLs of the remotes along with their names

git config --global credential.helper cache : to enable credential storage - 4

git config --global credential.helper "cache -- timeout= number of seconds" : to set the - 5
timeout limit in seconds

git push : to pushe the code in the current branch to the origin remote branch of the - 6
.same name

git push remote_name : to pushe the code in the current branch to the remote_name - 7
.remote branch

git push remote_name local_branch:remote_branch : This command pushes the - 8
.local_branch from the local repository to the remote_branch of the remote repository

git push remote_name branch_name : This command pushes the code on the - 9
.branch_name branch to the remote branch of the same name

git fetch remote_name : To download the changes that have appeared in the remote . -10
11- git merge origin/master : To merge the branch origin/master with your current active
.branch

. git pull : to fetch and merge in one step -12

Forking workflow

Definition

Forking Workflow is a model for managing software development commonly used in opensource projects. This approach involves creating independent copies of a project (forks) that .developers can work on freely without affecting the original version

When to Use It

: Forking Workflow is used in the following scenarios

.When multiple developers are working on an open-source project •

When there's a need to develop new features or fix bugs without impacting the stable •
.version

.In large projects with numerous contributors •

Benefits

Flexibility: Allows developers to experiment with new ideas without risking the main •
.codebase

Collaboration: Facilitates contributions through Pull Requests after developers complete
•
.their modifications

Isolation: Each developer works in their own environment, reducing the likelihood of •
.conflicts

Features

Easy Tracking: Changes and commits can be tracked separately, making code reviews •
.simpler

.Encourages Creativity: Developers can work on new ideas freely •

Distributed Responsibilities: Many developers can contribute to the project without •
.complex coordination

This model is ideal for teams that embrace open development culture and collective
.contributions