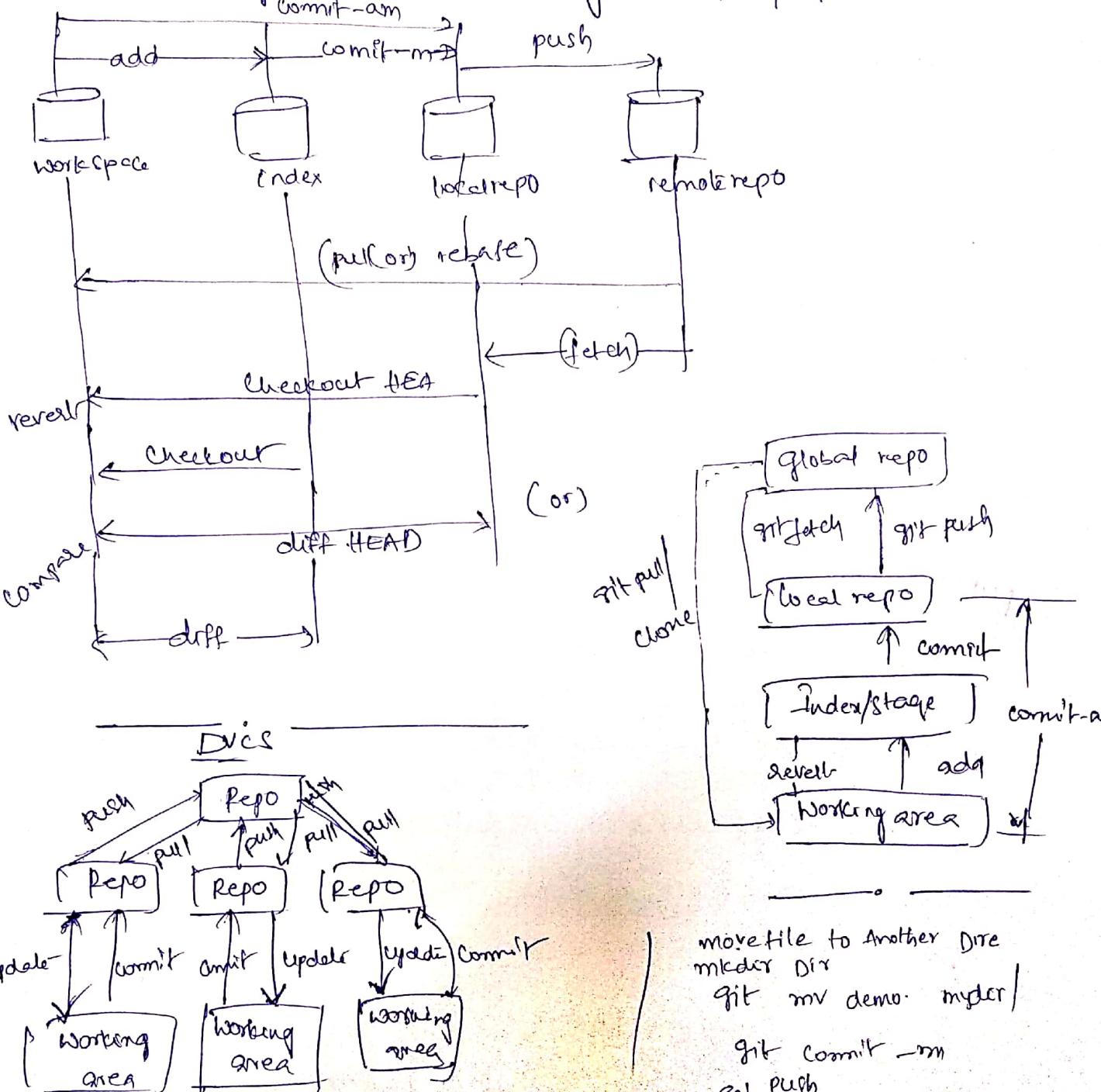


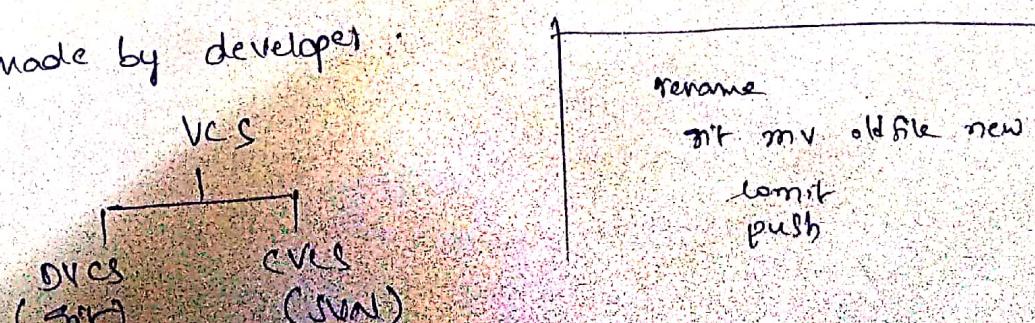
Java Version: 1.8
git: 2.7.4

Git: (Source Code Management Tool)

① Git: Git is a SCM tool it used to tracking the changes in Computer files and Coordinating work on those among multiple people.



VCS: Version Control System that records changes to a file (or) set of files made by developer.



② diff b/w git/SVN

git

- It is a DVCS
- Commit possible even it offline
- push/pull operation are faster
- Commit

SVN

- CVS
- not possible, possible only online
- Checkout/update operation are slow
- Revision

③ language used in git :→ C language

④ what is Bare repository on git?

- Bare repository in git just contains the Version Control info & no working files (no tree) and it doesn't contain .git

⑤ How to revert a commit that has already been pushed & made public?

- git revert <Commit id>

⑥ diff b/w gitpull & git fetch?

git fetch → git fetch is used to pull all new commits ~~and~~ from desired branch & stored in new branch in your local repo

gitpull → git fetch + git merge

pulls new changes from a particular branch from our central and update our target branch in our local repo

⑦ what is staging area/Index area?

It is an intermediate area, before completing the commit it can be formatted & reviewed.

⑧ git stash → stash is a temporary storage location

∴ if u have been working in one project, there is an urgency to complete other projects. the present working project will be kept in stash area.

git stash save -m "message"

—o—

⑨ How to find a list of files that had been changed in a particular Commit.

git diff-tree -r stash

—o—

⑩ Function of git config?

→ git uses ur username associate commits with an Identity

→ git config command is used to change ur git configuration include ur username

→ git config --global user.name Saichand. (add username)

→ git config --global user.email Saichand32@gmail.com. (add email)

—o—

⑪ Commit Object Content:

→ Set of files, representing state of project at a given point of time

→ Reference to parent Commit Object

→ SHA1 Code contains 40 character string

—o—

⑫ squash last N commits in to Single Commit

git rebase -i HEAD^{n}

—o—

⑬ How will u know in git if a branch has already been merged

git branch -merged → merged

⑭ git rebase: replay changes made in branch at the tip of the master
+ no commit generated.

⑮ Subgit :→ Subgit is a tool - for SVN to git migration,

⑯ repository in git?

- Repository contains a directory .git contains all metadata.
- .git is private to git

⑰ Advantage of using git:

- Data redundancy & replication
- High availability
- One .git directory per repo
- Collaboration friendly.

⑱ git push: → update remote refs along with associated objects.

⑲ git stash drop: When ur done with stashed items, if u want to remove stash file u use 'git stash drop'

⑳ git clone: → it creates a copy of an existing git repo

㉑ git config: → Set Configuration of ur repo, user info, preference etc, Behaviour of repo

㉒ How to create a repo in git → git init

㉓ How can conflict in git resolved?

- To resolve the conflict in git
- edit the file to fix the conflict changes
- Add the file git add filename
- and Commit. git commit -m "mess"

②4) Conflict in git?

→ A conflict arise when the commit that has to be merged has some changes in one place, and the current commit also has a change at the same place. Git will not be able to predict which changes should be take precedence.

②5) Another option for merging in git?

→ Rebasing.

②6) Syntax of rebase → `git rebase Branchname`

②7) git remote / git clone ?

git remote add → creates an entry in your git config that specifies a name for particular url

git clone: creates a copy of existing git repo.

②8) git diff function in git:

git diff shows changes b/w commits, commits & working tree etc.

②9) git status:

git status shows the the diff b/w working directory & index.

②10) git checkout: → git checkout command is used to update directories or specific file in your working tree with those from another branch without merging in the whole branch.

②11) git rm: → is used to remove the files from staging area.

②12) git log: → To find specific commit in your project history by author date, or history 'git log' used. (- to find the commit history)

③3) Git reset: → git reset is used to reset index as well as working directory to the state of our last commit.

③4) Git instaweb: Git instaweb automatically directs a web browser & runs web server with a interface in to your local repos.

③5) What does 'hook' consists in git?
This directory consists of shell scripts, which are activated after running the git commands; ex: git will try to execute post-commit script after you run a commit.

③6) How to fix broken commit

{ git commit --amend -m "update" }

git amend is used to change the commit message(only recent)

③7) git add :→ add files from working area to Index area

③8) Explain commit message!

Commit message is a feature of git which appears when u commit a change

③9) git fork: → fork is a remote, server side copy of a repo

⑩) git pull request :→ it gives the access to merge the other developer changes in to your branch

⑪) How to revert previous Commit

git reset --hard HEAD^{~1} → code & commit removed

git reset --soft HEAD^{~1} → code not removed only commit

(42) git cherry-pick → copy a commit made in one branch to another
git cherry-pick commitid

(43) How do cherry-pick a merge commit

git cherry-pick -m 1 Commit id
parent

(44) HEAD → recent commit / reference of last commit

(45) Working-tree → Working area where the developers see the file and
edit the file

(46) git bisection → is used to find a bad commit in git

git bisection start

git bisection bad // mark current version bad

git bisection good {hash or tag} mark given commit good

git bisection reset '

(47) git rebase conflict solve → The feature branch is created at the

tip of master. The development is done at both branches

the git rebase is used to replay the changes made in feature

branch added to tip of the master so this conflict raised then

verify the file and commit the file.

(48) How to recover deleted branch?

Checkout the lastest commit in reflog, and check it out as new
branch

(48) What is head in git and how many heads can be created in repo?

Head is a recent commit id/reference of last commit.

U can create number of Head in repo, Every branch having one head.

(49) Regular way of Branching in GIT?

Git has one main branch.

Create feature Branch development (or) Feature is done and added to master branch.

(50) Which command will u use to bring a new feature to the master branch.

Git merge (or) GIT pull

(51) Branching Strategies can u apply.

Main Branch.

Dev Branch.

Feature Branch.

Hotfix Branch.

(52) Git / Github:

Git helps manage ur source code history

Github! → is a hosting service for git.

53) How can we see n most recent commits in GIT?

git log --oneline / git log

54) How can we see the config setting of GIT installation.

git config --list

55) How do we use interactive rebase in place of Auto rebase

→ git rebase -i head^ (Squash last n commits into 1 commit)

56) git rerere: reuse recorded resolution →

→ If we have solved conflict before after same conflict occurs
we simply use git rerere

57) How can we see different b/w two commits in git

git diff commit--commit

58) When we run git branch <branchname> how does git know
the SHA-1 of the last commit.

59) How can we rename a remote repo.

github → Project/repo → Setting → rename repo

60) Is origin a special branch in git

Origin is the name of the remote repo

default remote repo/

61 How can we configure git to not ask for password every time?

git remote set-url origin git@github.com: name/repo.git

62 How can we work on a project where we do not have push access?

63 Git Blame: → is used to give information of line by line Committing a file.

git blame -L filename .

64 Replace Commit Id

git rebase --onto Commitid^ Commitid.

65 How to move a file from one branch to another branch

git diff --stat ~~branchname~~ {1,3} {Show all files in branch}

git checkout --merge branchname file .

66 git reflog → shows user logs.

67 How to clone particular file .

git clone --depth=1 'giturl' filename .

68 How to clone particular branch

git clone -b branchname giturl .

Q7) How to set user details globally in git hub

git config --global user.name Sarthak

git config --global user.email Sarthak321@gmail.com

Q8) How to see user details

git config --list

Q9) How to see remote url \rightarrow git remote -v

Q10) How to add repo to remote

git remote add remotename $\frac{\text{github}}{\text{(or)}}/\text{username}/\text{repo}/\text{git}$

Q11) How to revert Commit id origin

Soft reset

git reset --soft
HEAD⁽¹⁾

Code is remained
Commit id removed

Code in Stage
area

Hard reset

git reset --hard
HEAD⁽¹⁾

Code & Commit id
both removed

Mixed reset

git reset --mixed HEAD⁽¹⁾

Commit id removed

Code in working area

Q12) How to undo merge / Rebate process?

git merge --abort

git rebase --abort

Q13) git contains { branches, Config, descriptions, HEAD, hooks, info
objects, refs }.

Q14) How to make patches

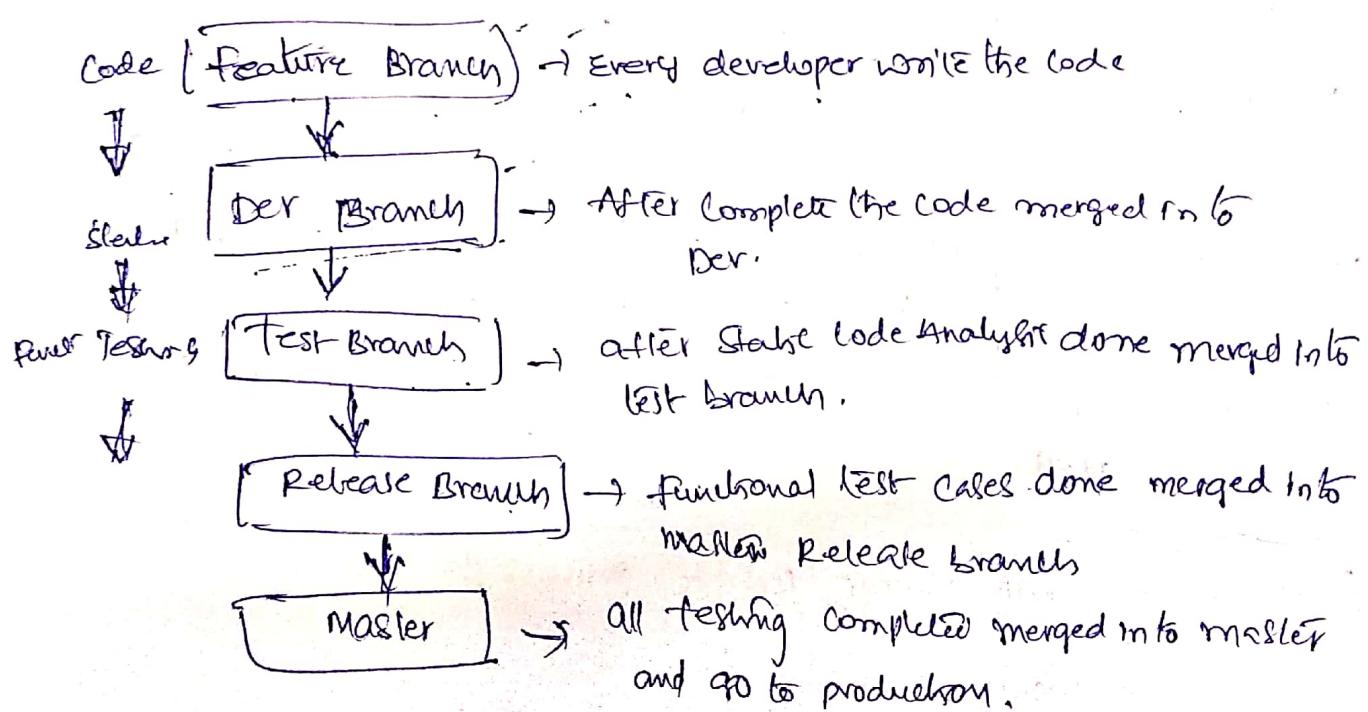
git format-patch CommitId (single commit)

78) git patch: git patch is a text file, whose contents are similar to git diff but along with code it also has metadata about commits: commit id, com date, Commit msg etc, we can create a patch from commit & other people apply them to their repository.

79) git tag uses?

tagging in git refers to creating specific point of history for ur repository.

80) What is branching strategy you are using, How many branches in ur current env what are those purpose.



81) How to push multiple branches at a time in git hub

git push origin <br1> <br2> <br3>

82) what is .gitignore file? what is the purpose of the file.

all w/ commit

git log

git log -p

git log --since=12-3-17 --until=13-3-2017

git log --oneline

→ To make changes to tracked files

→ git diff commitid -- Commitid.

git log -1

Review changes.

git show Commitid.

① diff b/w lastcommit / github
gitlab

Search by commit message → git log --grep "pattern"

Set number of records

git log -n 2

git log -- Commit message without date

Add users

① GitHub → Repo → Setting → (Onboard) → Add email address → Env file

② Organisation → [repositories | people | teams | projects] (Setting)

How to give access to the particular branch.

→ Organisation → Repository → Setting → ^{Branch} Branch protection Rule →

(1) Require

Request pull request reviews before merging.

(2) Require status checks to pass before merging.

(3) Require signed commits

(4) Include administrators

Let's have a look at the branches