**GIT REPO. WITH & W/O README FILE**

1. **BASIC:**

* ***git init***  🡪 Initializes an empty git repo (turn an Existing Directory to GIT Repo)
* ***git add <file\_name>*** 🡪 adds files from directory to local repo
* ***git commit -m “commit message”*** 🡪 adds changes to local repo (in STAGING area; ready for the next stage)
* ***git branch -M <branch\_name>***  🡪 to which branch we are heading to; main(default)/any other branch
* ***git remote add origin <ssh>*** 🡪 connect local repo to the remote
* ***git push -u origin <branch\_name>*** 🡪 push changes to the remote repo

1. **BRANCHING STRATEGY:**

* ***git branch*** 🡪 displays the available branches in the repo
* ***git branch <branch\_name>*** 🡪 creates a new branch
* ***git checkout <branch>*** 🡪 heads to the particular branch (switching b/w branches)

1. **VIEWING & PULLING CHANGES made to REMOTE REPO**

* ***git fetch origin <branch>*** 🡪 displays any changes made in remote repo (by you or by anyone)
* ***git pull origin <branch>*** 🡪 pulls the made changes from remote to local repo

1. **MERGE BRANCHES (by ‘CLI’ (or) by ‘REMOTE’)**

* **By REMOTE**

1. Repo 🡪 **‘pull requests’** 🡪 **‘compare and pull request’**

***[we can rise ‘new request’]***

1. Admin gets the ‘pull request’ ; **‘can add any no.of persons for reviewing before merging’**
2. **Can ‘DELETE’ branch (if not required) after merging**
3. ***git pull origin main***
4. **RELEASING (make it Ready for BUILD)**

* ***git tag*** 🡪 checks the available tags
* ***git tag <tag\_name>*** 🡪 used to manage the release; after making required changes/adding new files can name (tag) it as new release
* ***git push origin <tag\_name>***

1. **DIFFERENCE B/W BRANCHES**

* ***git diff <branch1> <branch2>*** 🡪 shows the difference b/w any two branches

**INTERVIEW QUESTIONS**

* **Difference between ‘fetch’ and ‘pull’ ?**
* **What is ‘GIT Cherrypick’ ?**
* **What is “PR Rise”?** 🡪 when you made any changes in the code and intended to merge the code to the ‘main’ branch, can rise pull request to the ADMIN

**DOUBTS**

1. **Can we change the repo VISIBILITY** 🡪 YES
2. **Changing the name of the folder** 🡪 1. In repo 2. With CLI (mv)
3. **‘HYBRID Model’ 🡪 in SBI App we’re logging in ‘Public portal’, need our balance 🡪 whose database is private**
4. **What is SCM?**

* To add code to the repo’s, need SCM tools
* There are many SCM tools
* Doing work continuously w/o disturbing the piece of code in master branch 🡪 is the ‘main’ concept of SCM

1. **Why GIT required as we have already CLI in OS?**

GIT is the CLI required to perform GIT Actions

1. **What is GIT SSH command?**

* SSH, also known as ‘Secure Shell’ or ‘Secure Socket Shell’, is a **network protocol** that gives users, particularly system administrators, a secure way to access a computer over an unsecured network.
* Github **doesn’t need** your **username** and **password** each time when using SSH keys
* You can revoke specific keys in SSH 🡪 since you setup a key for each computer or machine you use by provoking a specific key
* Github automatically removes inactive key after an year

1. **Generating a new SSH key 🡪** refer[**https://www.youtube.com/watch?v=8X4u9sca3Io**](https://www.youtube.com/watch?v=8X4u9sca3Io)
2. ssh-keygen -t rsa -b 4096 -C "<mail\_id>" (to genearate the public-private key)
3. cat ~/.ssh/id\_rsa.pub
4. add the key to github
5. **Adding an existing key**
6. Enter ‘***ls -al ~/.ssh’*** to see if existing SSH keys are present
7. Check the directory listing to see if you already have a public SSH key. By default, the filenames of supported public keys for GitHub are one of the following. id\_rsa.pub; id\_ecdsa.pub and id\_ed25519.pub
8. ***eval "$(ssh-agent -s)"***
9. ***ssh-add ~/.ssh/id\_rsa***
10. ***cat ~/.ssh/id\_rsa.pub***
11. add the key to github

**ISSUES in GIT**

**ISSUE1: Connecting local repo to remote repo in LINUX**

* There are two alternatives to connect local repo to remote repo.

1. **Generating a new ssh key / adding an existing one to GITHUB (SSH)**

* Effective way to avoid entering username and password each time of connecting local repo to remote repo
* Instead of http, can use ssh url
* **\*\*Error prone:** better to add ‘rsa’ instead of ‘ed25519’ to avoid errors
* Check 🡪**(**[**https://windowsreport.com/bad-owner-or-permissions-on-ssh-config/#:~:text=While%20there%20is%20no%20official,web%20and%20in%20the%20forums**](https://windowsreport.com/bad-owner-or-permissions-on-ssh-config/#:~:text=While%20there%20is%20no%20official,web%20and%20in%20the%20forums)**.) 🡪 using ‘commands’ worked for me**

1. **Generating a token with a certain period(https)**

* Instead of entering ‘password’ in ‘https’ generate ‘token’ and save it (somewhere in notepad) and enter the token instead of ‘password’
* Github 🡪 settings 🡪 developer settings (lower left) 🡪 personal access tokens 🡪 tokens (classic) 🡪 generate new token (classic)
* \*\*Instead of ‘no expiration’ better to give ‘certain period’ ; “have to create new token after expiration of existing one”

**ISSUE2: Connecting local repo to remote repo in WINDOWS (GIT BASH)**

* Only “way2-through https” is working.