*(how to merge working in different user name files)*

Install git bash

to change default directory from git bash cmd

cd /e/Onedrive/GENERAL/Books/Computer

Basic GIT commands

*git config* :- It is used to set the name of the author and the email address which you want your commitment to addressing.

git config --global user.name “Jain”

git config –global user.email [jainibrm@gmail.com](mailto:jainibrm@gmail.com)

git config – global – unset user.name // remover user name from config setting

git config –global –unset user.email // remove user email from config setting

The command below return a list of information about your git configuration

git config –list // to list all git config setting :q = quit

git config user.name // to display config user name

git config user.email // to display config user.email

*git init* := It is used to start a new git repository. This is generally used at the beginning.

git init <repo name> or git init if your in that subfolder

git init d:\new/ // start new directory

*git add*:= It is used to add a file to the staging area. Instead of choosing single file name,

you can also choose to give all filenames with an \*.

git add <file name> // touch text.txt - to creat .txt file

git add \* or git add.

git status :- to know the status of the working tree.

*git commit -m* := It is used to snapshot or record a file in its version history permanently.

git commit –m < ”type Remark Message”>

git comit –m “First Commit”

*git commit –a* :- This commit command is used to commit any such file which has been

added as a result of the git add command. It is also responsible for

committing any other files to which you have brought a change to since

then

git commit –a

*git log* :- This is used for listing down the version history for the current working branch

git log or git log --all

git checkout < Branch Name or ref.No.>

*git branch* : command is used to list down all the branches that are locally present in the

respository

git branch <branch name>

GIT Part 2

The git merge command lets you take the independent lines of development created by git branch and integrate them into a single branch.

*Creating Branches :* The git branch command can be used to create a new branch. When you want to start a new future, you create a new branch off main using.

git branch <new branch>

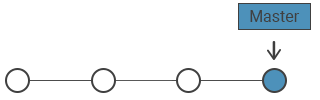
Once created you can then use ‘git checkout <new branch>’ to switch to that branch.

git merge :- to merge branch locally use git checkout to switch to the branch you want to merge into. This branch is typically the main branch. Next, use git merge and specify the name of the other branch into this branch.

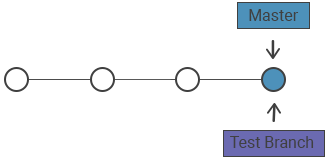
Install : Gitlab or Github or gitbucket

git remote add <variable name> <copy link for git site>

git push <variable name> // to push github into github



git branch < test\_branch>



The newly created branch should be selected with “git checkout”

*Deleting Branches* : After finishing the work on a branch and merging it into the main master, ou can delete it

Git branch –d test\_branch

*git stash* :-

*Git pull :* The git pull command is used to fetch and download content from a remote respository and immediately updte the local respository to match the content.

*REMEMBER :*

*after any changes in filenames*

*1) git add <file name> or all*

*2) git commit <filename> or git commit // to commit all files in the folder*

*3) git remote*

*4) git push*

*any changes goto 1*

Can I add other GitHub users to my organization?

You can add other GitHub users to your organization, and even transfer the organization's ownership to them if you even need to. Each user will have their own account, so you don't need to share a professional account's password with other people

How do I add a collaborator to my GitHub account?

Ensure the collaborator has a GitHub account, otherwise the person will not be added as a collaborator. Click Finish. If you already have an organization, click the organization name under the profile. Click Invite someone. Enter their GitHub username or email.

Admin : [jainbrm@gmail.com](mailto:jainbrm@gmail.com) Github Id = jainjosephk( old Jain2000), ibrm2000

Collaborator : [jainibrm@gmail.com](mailto:jainibrm@gmail.com) Github id = jainkjoseph, ibrm2000

*Part # 3*

Create id at Gitlab or Github

Creat New repository name (subfolder) for ech projects in public or private

Copy HTTP <url address > and past to git terminal (git CMD) along with

git remote add origin <copy http url>

How to Add a remote in Git

Cloning a respository (Subfolder) from a remote server downloads the project to your local computer and leaves you with a local Git repository.

We're going to add a new remote connection to our local repository using the git remote command and need to pieces of information for this:

1. The name we'd like for this new remote.
2. The URL of the remote repository. You can find this after creating a new remote repo on your hosting service of choice (e.g. GitHub, GitLab, Bitbucket...).

Copy HTTP <url address > and past to git terminal (git CMD) along with

*git remote add* origin <copy http url>

( git remote add origin https://github.com/gittower/example.git)

In Git, "origin" is a shorthand name for the remote repository that a project was originally cloned from. More precisely, it is used instead of that original repository's URL - and thereby makes referencing much easier.

git remote // display remote name (ie. origin)

git remote –v // check if command has worked

*Git Push Command:*  The git push command is used to upload local respository content to a remote repository.

git push <origin> master // git push –u origin master

How to copy from github to our system

*git clone :*- command downloads an existing GIT repository to your local computer

git clone https://github.com/Jainjosephk/Git.git

This will download the project to a folder name after the Git repository(<https://github.com/Jainjosephk>) (*“Git” in the case*). If you want a different folder name, simply specify it as the last parameter:

git clone <https://github.com/Jainjosephk/Git.git> < New foldername>

for ubundo OS

before push command : Open Github/Setting/Developer Setting/Personal Access Token

Generate New Personal Access Token

Note: Demo for Access , 30 days, Select All options & Confirm

ghp\_Xt6VDMUdtYEMdqFRTjXat7Kn4j00Vf0tgGBN

part#5

## Git Ignore

When sharing your code with others, there are often files or parts of your project, you do not want to share.

Examples

* log files
* temporary files
* hidden files
* personal files
* etc.

Git can specify which files or parts of your project should be ignored by Git using a .gitignore file.

Git will not track files and folders specified in .gitignore. However, the .gitignore file itself IS tracked by Git.

## Create .gitignore

To create a .gitignore file, go to the root of your local Git, and create it:

### Example

touch .gitignore

Now open the file using a text editor.

We are just going to add two simple rules:

* Ignore any files with the .log extension
* Ignore everything in any directory named temp

### Example

# ignore ALL .log files  
\*.log  
  
# ignore ALL files in ANY directory named temp  
temp/

Now all .log files and anything in temp folders will be ignored by Git.

Note: In this case, we use a single .gitignore which applies to the entire repository.

It is also possible to have additional .gitignore files in subdirectories. These only apply to files or folders within that directory.

ISSUES :

Use issues to collaborate on ideas, solve problems, and plan work. Share and discuss proposals with your team and with outside collaborators.

SPONSORSHIP : (Add from Git Settings)

GitHub Sponsors allows the developer community to financially support the people and organizations who design, build, and maintain the open source projects they depend on, directly on GitHub.

To link bank web etc

gist : Creating gists (for copy paste reusable programes links for future use.

You can create two kinds of gists: public and secret. Create a public gist if you're ready to share your ideas with the world or a secret gist if you're not

Action : (Yaml Tutorial)

Project Management:

git diffferent workflow: