Git is source/version control management.

We always have access to previous versions

We need to configure our details so that we have a track how has uploaded any doc

pavan@pavani MINGW64 ~ (master)

$ git config --global user.name "Pavani Billapati"

pavan@pavani MINGW64 ~ (master)

$ git config --global user.email [pavanibillapati01@gmail.com](mailto:pavanibillapati01@gmail.com)

**Create a default branch**

pavan@pavani MINGW64 ~ (master)

$ git config --global init.default branch main

**Help**

Git config -h

Or

Git help config

Git init 🡪 initializes empty repository & creates hidden .git files in local file system

Git status 🡪 shows the

Untracked files – we wont be able to know changes made or retrieve these files

Git add . 🡪 we use add followed by filename or . to track all files

Git rm –cached <fn> 🡪to untrack a file

If we want to ignore any particular file, we can include them in .gitignore folder & specify regular expressions based on ur need

Git commit -m “message”🡪 we want this version

Git diff 🡪 shows difference between previous & current version

We can also see that file has been modified in status

**3 environments 🡪 working directory, staging, commit**

Git restore –stagged . 🡪 brings files back to working files.

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git commit -a -m "upload"

[master 9f1a923] upload

2 files changed, 0 insertions(+), 0 deletions(-)

create mode 100644 ~WRL1820.tmp

This skips staging environment & directly commits it

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git rm "doubts.docx"

rm 'doubts.docx'

deletes the mentioned file

git status 🡪 shows that file has been deleted

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git restore --source=HEAD --staged --worktree -- doubts.docx

Restores the file if the changes are not committed yet

**or**

If the file hasn't been committed yet, you can use **git checkout** to discard changes in the working directory

**git checkout -- path/to/your/file.txt**

If the changes have already been committed, you can use **git checkout** with the commit hash

**git checkout <commit\_hash> -- path/to/your/file.txt**

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git mv "doubts.docx" "AWS doubts.docx"

Rename file

Git log 🡪 to see all commits made so far with details

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git log --oneline

2c47490 (HEAD -> master) upload

9f1a923 upload

3830166 upload git

Displays one line of commit details

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git commit -m "changed commit message" --amend

[master dc98f77] changed commit message

Date: Thu Jan 25 12:54:45 2024 -0500

1 file changed, 0 insertions(+), 0 deletions(-)

rename doubts.docx => AWS doubts.docx (100%)

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git log --oneline

dc98f77 (HEAD -> master) changed commit message

9f1a923 upload

3830166 upload git

Changes the commit message

Git log -p 🡪 detailed changes would be listed

Branch – we create branches before making any changes, switch to it, commit them in that branch & switch back to main branch & merge changes

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git branch testbranch

Creates a new branch

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git branch

\* master

Testbranch

Displays all branches available & displays \* before the active branch

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git switch testbranch

Switched to branch 'testbranch'

M git.docx

To change to different branch

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (testbranch)

$ git branch

master

\* testbranch

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (testbranch)

$ git commit -m "updated git file"

On branch testbranch

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: git.docx

no changes added to commit (use "git add" and/or "git commit -a")

this didn’t commit file as it only includes changes that have been explicitly staged using **git add**

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (testbranch)

$ git commit -a -m "updated git file"

[testbranch 5983100] updated git file

1 file changed, 0 insertions(+), 0 deletions(-)

This commits changes in all tracked files

$ git switch master 🡪 switching back to main branch

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git merge -m "Merge" testbranch

Merge made by the 'ort' strategy.

* Merging content from testbranch to main

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git branch -d testbranch

Deleted branch testbranch (was 5983100).

Deletes the branch

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (master)

$ git switch -c UpdateText

Switched to a new branch 'UpdateText'

Creates & switches to that branch

pavan@pavani MINGW64 /C/Users/pavan/OneDrive/Desktop/prep/Notes (UpdateText)

$ git branch

\* UpdateText

master