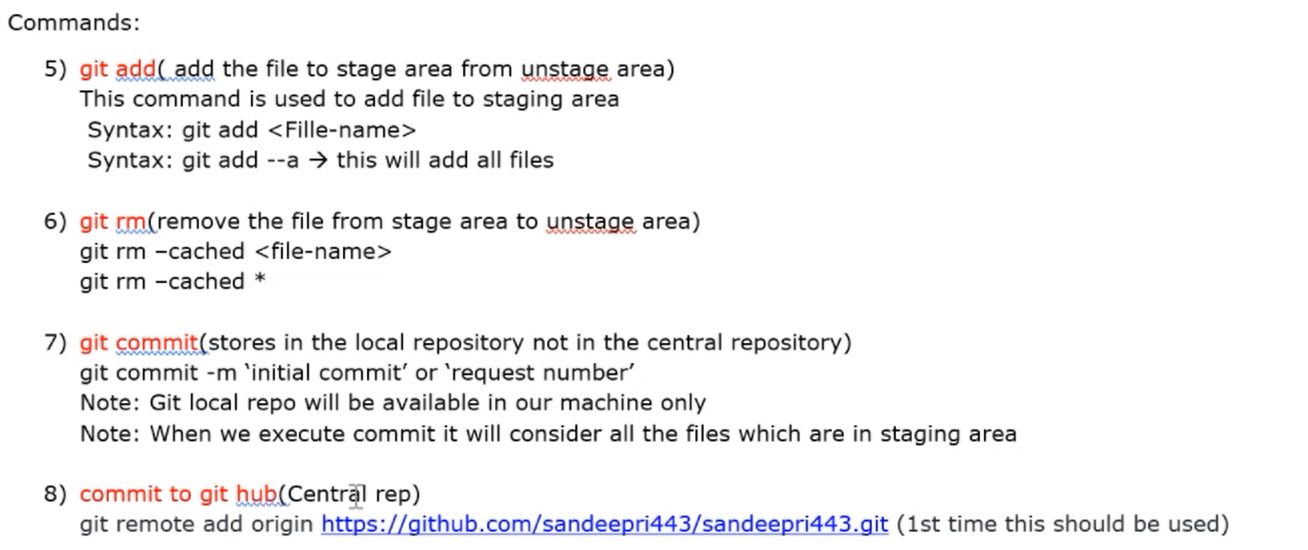
Commands

|  |  |  |
| --- | --- | --- |
| Task | Function | command |
| See all command in git bash | Shows list frequently used several git commands | Git help |
| Info about a command | It opens the documentation of that particular command | git help <command> |
| Initialise repo or create empty repo | It is used to create empty repository or re-initialize existing repo | git init |
| File status | display the status of current repository | git status |

Types of Files:

1) Stage files; Files which are added and they are **eligible to commit Displayed in green colour**

2) Un-staged files: I Modified files will be displayed here, we need to stage these **files to commit Displayed in Red colour.**



8) commit to git hub(Central rep) git remote add origin https://github.com/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.git (1st time this should be used)

9) git config --global user.email [sandeepri443@gmail.com](mailto:sandeepri443@gmail.com)

git config --global user name "your-github\_username" git config --global user email "your.github\_email" git config -1

10) Git Push (generate token) git push-u origin master(Used to move changes from local to central) git push(2nd time onwards)

In case of errors

11) git reset( it is used to unstage a file) git reset HEAD

12) git checkout (used to discard changes done in a file)

git checkout -filename commit-id: whenever we commit, git will generate commit-id(7 characters) commit ID contains 40 alphanumeric characters from 40 alphanumeric characters it will display first 7 characters to us.

Check History of the commit:

Syntax: Git log Git log -onetime

Git log -3

In logs it will display below details

* commit-id
* author
* timestamp

|  |  |  |
| --- | --- | --- |
| Task | Function | command |
| Check the current working branch: | Present working branch | git branch |
| Git Checkout(from master to main) | Switch one branch to another branch | git checkout main |
| Create new branch | New branch | git branch -b antony  Git checkout -b antony |
| Delete branch | GIT Branch Delete | Git branch -D antony |
| Git Merge  *Antony to mater* | Checkout to master and merge the modified data from branch (Antony) | Git Checkout branch master Git merge antony |