|  |  |  |
| --- | --- | --- |
| **Setup** | | |
| 1 | Command to check config options | git config --list git config --list --show-origin git config user.name |
| 2 | Command to check config options + file origin | git config –show-origin user.email |
| 3 | Command to configure config options  name and email | git config --global user.name "Akintunde Adegbayo" git config --global user.email akt231@yahoo.com |
| 4 | Command to configure config options  editor | git config --global core.editor emacs |
| 5 | Command to prepare a directory for git | #make sure you are in current directory |
| 6 | Command to open git config file in editor of choice | git config --global --edit |
| 7 |  |  |
| 8 |  |  |
| 9 |  |  |
| 10 |  |  |
|  |  |  |
| **Queries** | | |
| 1 | Command to Check Git Status;  ***Checks what branch you are on***  ***compares branch you are on to remote branch***  ***lists untracked files***  ***lists modified tracked files*** | git status  git status -s ***#concise version*** |
| 2 | What have you changed but not yet staged(changes after last add)? | git diff |
|  | And what have you staged that you are about to commit? | git diff --staged |
| 3 | Command for git help | git <verb> -h ***#concise version*** git <verb> help ***#verbose version*** git help <verb> ***#verbose version*** |
| 4 | Command to check content of a directory | ls -la ls |
| 5 | Command to check current file path | pwd |
| 6 | To view Commit History | git log |
| 7 | To view Commit history and show difference (patch output) | git log -p  git log --patch |
| 8 | To view Commit history and show difference (patch output) and show only last 2 entries | git log –patch -2 |
| 9 |  |  |
| 10 |  |  |
| **add new and changed items to staging area for tracking** | | |
| 1 | add new and changed items to staging area | git add <file\_name> |
| 2 | add all new and changed items to staging area | git add \* |
| 3 | undo a addition of new and changed items to  staging area | git reset HEAD <file\_name> |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |
| 7 |  |  |
| 8 |  |  |
| 9 |  |  |
| 10 |  |  |
| **commit item** | | |
| 1 | commit changes | git commit -m "comment\_description" |
| 2 | undo a commit | git commit --amend |
| 3 | commit every file and stage automatically | git commit -a |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |
| 7 |  |  |
| 8 |  |  |
| 9 |  |  |
| 10 |  |  |
| **push to github** | | |
| 1 | push to github | git push |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |
| 7 |  |  |
| 8 |  |  |
| 9 |  |  |
| 10 |  |  |
| **Remove a file from git** | | |
| 1 | Removed file from git  you have to remove it from your tracked files (more accurately, remove it from your staging area) and then commit. Deletes file physically too | git rm -f <file name> |
| 2 | Removed file from git staged area but lists file as to be tracked | git rm –cached <file name> |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Stopped @

https://git-scm.com/book/en/v2/Git-Basics-Viewing-the-Commit-History