

GitHub Learning:

git status

git add {file_name}

git add *

git commit -m "your_message"

git reset head {file_name}

git push -u origin master

git push origin master --force

git pull origin

config:

git config --global user.name "abdulaziz"

git config --global --unset user.name

git config -list

git config -l

git config --global --edit

A screenshot of a code editor showing a .gitconfig file. The file is divided into sections: [filter "lfs"], [user], [color "status"], and [color "branch"]. The [filter "lfs"] section contains settings for clean, smudge, process, and required. The [user] section contains email and name. The [color "status"] section contains settings for added, changed, and untracked. The [color "branch"] section contains a setting for remote. The code is color-coded: blue for section headers, green for variable names, and yellow for values. The 'bold' keyword is highlighted in the 'changed' and 'untracked' settings.

```
1 [filter "lfs"]
2   clean = git-lfs clean -- %f
3   smudge = git-lfs smudge -- %f
4   process = git-lfs filter-process
5   required = true
6 [user]
7   email = o@gmail.com
8   name = OsamaElzero
9
10 [color "status"]
11   added = yellow
12   changed = red bold
13   untracked = magenta bold
14
15 [color "branch"]
16   remote = yellow
17
```

ssh-keygen -t rsa -b 4096 -C "azoozsm2@gmail.com"

ssh -T git@github.com

create branch:
git branch Scroll

move to branch:
git checkout Scroll
git checkout master

delete branch (normal delete):
git branch -d Scroll

delete branch (Force Delete):
delete branch -D Scroll

List all exists branches:
git branch

Create Branch and move to it:
git checkout -b Dev-Feature

Rename Branch:
git branch -m Development

Merge Branch:
git merge Development

Stash:

git stash
git stash list
git stash pop (will pop last one)
git stash apply
git stash pop stash@{2}
git stash drop (will delete last one)
git stash drop stash@{1}
git stash show (will show last one)
git stash show stash@{0}
git stash clear

git restore --staged {file_name}

git clean -n (to list files that would be removed)

git clean -f (to remove files from (list from git clean -n))

git log

git reset --hard {commit_hash_that_before_hash_you_want_delete_it}

gitignore:

touch .gitignore

Into .gitignore file:

*.log

!vip.log

node_packs/

text.txt

git add -f text.txt (Force Add)

git ignore patterns

git tag

git tag v1.0

git push origin v1.0

git tag -a v2.0 -m "your_message"

For Delete tag (Local):

git tag -d v1.0

For Delete tag (Remote):

git push origin --delete v1.0