

-----COMANDOS INICIAIS-----

git -l -> mostra as configs

git status -> mostrar os status das suas branches

git add -> adicionar na staging area

git rm <file> -> remoção de arquivos, parecido com o add, se der commit iremos deletar do repositório do git

git add --all -> faz todas as ações de deleção e adição

git rm -rf <file> --cached -> retira os arquivos da staging area (depois do add)

git commit -a -m ou (-am) "mensagem" -> adiciona tudo que já existe no repositório e commita ao mesmo tempo

git commit -m "" --amend -> substituir a última msg

git reset --soft (Hash do commit anterior) -> “volta” o commit com os arquivos em staged (pronto para commita)

git reset --mixed (Hash do commit anterior) -> “volta” o commit com os arquivos em modified

git reset --hard (Hash do commit anterior) -> “volta” tudo que foi feito no último commit

git log -> mostra os commit feitos

git diff -> mostra o que mudou dos arquivos

-----BRANCHS-----

git branch --orphan <nome> -> criar branch totalmente em branco

git branch -> mostrar a branch

git branch <nome> -> criar uma branch

git checkout -b <nome> -> adicionar uma nova branch e mudar para ela

git checkout <nome> -> mudar de branch

git merge <nome da branch> -> uni a branch em que vc está com a branch a qual você quer juntar

git branch -m <nome> -> renomear branch

git branch -D <nome> <nome2> -> deletar branches

deletar branch no github: deletar branch local e depois git push origin <branch> --Delete

-----REPOSITÓRIOS-----

git init -> inicia o repositório local

git remote add <nome do repo> <link> -> adiciona um repositório

git remote -v -> mostra todos os repositórios existentes

git push <repo name> <branch> -> empurrar do staging area para o repositório

git clone <link do repo> <pasta>-> clonar repositório

-----COMANDOS ADICIONAIS-----

git stash -> guarda modificações de arquivos

git stash apply -> retorna os arquivos modificados

git stash list

git stash clean

git diff -> mostrar as mudanças

git diff <file> -> apenas no arquivo

git diff --stage -> mostrar as mudanças quando add

git log --name-status -> as mudanças feitas, dentro do log

git diff <hash(7)commit> <hash(7)commit> -> mostrar diferença entre commits

TEM COMO ADICIONAR UM .gitignore global, é só criar o arquivo e adicionar ele no:

git config --global core.excludesfile <file>

COMMITTS ELABORADOS -> git commit, vai abrir o terminal do unix, e lá pode escrever a vontade... Para sair ctrl+c e :wq, para cancelar (:qa!)

git config --global core.editor "nome do editor"

git log --pretty=oneline -> dá uma linha no log

git log --abbrev-commit

git log --stats

git tag -> lista de tags

git tag -a <versão.tag> -m "<mensag.>" -> cria um marcador de versão

git push <nome.repo> <nome.branch> --tags -> dá um push nas tags

git revert <hash.commit> -> desfaz o que o commit selecionado fez

git push <nome.repo> :<nome.tag ou nome.branch> -> remove tag ou branch do repositório remoto

Quando algum arquivo é deletado ou perdido há como recuperar: