

git --version
git --v
git config --global user.name "Dário da Silva Melo"
git config --global user.e-mail "dario.melo@ufu.br"
git config --global user.name
git config --global user.e-mail
git config --global core.editor "code --wait"
git config --global init.defaultBranch main
git config --list
git status
git add .
git add -A
git add --all
git add (nome_arquivo.extensão)
git commit -m "Primeira Versão"
git commit --amend -m "mensagem alterada"
git commit --amend --no-edit
git diff
git diff --cached
git log
git log --oneline
git log -1
git log -2
git log --oneline -1
git log --patch
git log -p
git log --stat
git log --shortstat
code .
git checkout numero_do_comiite
git checkout nome_do_arquivo.extensão
git checkout .
git restore --staged nome_do_arquivo.extensão
git restore --staged .
git rm --cached (nome_aquivo.extensão)
git rm --cached -r .
git reset --hard
git clean -f
git rm -fr ./git
touch .gitignore
git update-index --skip-worktree main.txt
git update-index --no-skip-worktree main.txt
git clone pasta_origem pasta_clone
git clone https://....
git remote -v
git remote add <alias> https://....
git remote set-url <alias> https://...
git push
git pull
ssh-keygen

cd ~/.ssh/
start .
muda o nome dos arquivos
eval \$(ssh-agent)
ssh-add ~/.ssh/nome_do_arquivo (não tem extensão)
git branch
git branch --list
git branch nome_da_branch
git checkout nome_da_branch
git checkout -b nome_da_branch
git switch nome_da_branch
git switch -
git switch -c
git checkout -f nome_da_branch
git push --set-upstream <alias_remoto> <nome_branch>
git push -u <alias_remoto> <nome_branch>
git branch -d <nome_branch>
git branch -D <nome_branch>
git push --delete <alias_remoto> <nome_branch>
git branch -m <novo_nome_branch>
git branch -m <nome_branch> <novo_nome_branch>
git log <nome_branch> --oneline -5
git merge <nome_branch_que_vai_ser_margiada>
git branch --no-merged
git branch --merged
git merge --abort
git reset --hard

mostra a versão do git
mostra a versão do git
seta o usuario no git
seta o email no git
mostra o usuario do git
mostra o e-mail que está setado
configura o git para usar o editor de texto VsCode
define que ao criar uma Branch ela iniciara como main
mostra a configuração do git
lista os arquivos não rastriados (untracked)
adiciona todos os arquivos no rastreamento
adiciona todos os arquivos no rastreamento
adiciona todos os arquivos no rastreamento
adicono um arquivo especifico no rastreamento
salva as alterações no repostioria e cria a primeira vesão
faz o comite novamente (acrescenta um arquivo esquecio no mesmo commit e alterara a mensagem
faz o comite novamente (acrescenta um arquivo esquecio no mesmo commit e não altera a mensagem
mostra a diferença da versão Modified e Unmodified
mostra a diferença da versão Staged e Unmodified
mostra a lista dos commit's realizados com informações
mostra de forma resumida a lista de commit's
lista o ultimo commit
lista o penultimo commit e assim vai git log -3...-n
lista o ultimo commit de forma resumida
mostra um historio de todas as modificações ocorridas
mostra um historio de todas as modificações ocorridas
mostra um historio dos arquivos modificados e commitados
mostra quantos aquivos forão alterados e que linha
abri o VsCode no diretorio corrente
mostra a versão dos arquivos referente ao commit que deseja
todas as alterações feitas no arquivo expecifico são perdidas e volta as informações do ultimo commit (não podem e
todas as alterações feitas nos arquivos são perdidas e volta as informações do ultimo commit (não podem estar na a
tira o arqqquivo da área de preparação (Staged - arquivo rastreado). Se já existe um commit
tira todos arqqquivo da área de preparação (Staged - arquivo rastreado). Se já existe um commit
tira o arqqquivo da área de preparação (Staged - arquivo rastreado). Se não existir um Commit
tira todos os arquivos da area de preparação (Staged- arquivo rastreado). Se não existiir um Commit
perde tudo que estiver na area de modificação (modified) e preparação (Staged). Volta para o Ultimo Commit
apaga todos os arquivos não rastreados. no estado: Unmodified
apaga o repositorio (não faça isso - não é uma opção valida)
cria o orquivo .gitignore (para não versionar arquivos). Exemplo *.exe, conf.doc, etc. A # é o comentário
faz com que o arquivo não seja mais ratreado
faz com que o arquivo volte a ser rastreado
faz o clone da pasta de origem
cria um clone do repositorio no github
mostra o servidor remoto associado para burcar e enviar informações
faz a ligação entre o repositorio e o servidor remoto. Um Repositorio pode apontar para varios servidores
atualiza o caminho do servidor remoto para aquele alias.
envia as informações do repositorio para o servidor remoto
traz a ultima versão no servidor remoto

lista as branch

lista as branch

cria uma nova branch

entra na branch desejada

cria uma nova branch e entra nela

entra na branch desejada

volta para ultima branch que eu tinha selecionado

cria uma nova branch e entra nela

entra na branch desejada apagando todas as modificações feitas nos arquivos rastros na branch que eu estava

seta a nova branch para o servidor remoto

seta a nova branch para o servidor remoto

apaga a branch

força a deleção da branch (caso o git pergunte se deseja realmente apagar)

apaga a branch remota

renomeando a branch

renomeando a branch (caso não esteja nela)

Variações do log: pode listar os commites e outras funcionalidades do log apenas passando o nome da branch com

git merge (mesclar) a branch desejada para a branch que está logado.

lista as branch que não foram mergeadas.

lista as branch que foram mergeadas.

no caso de conflito este comando serve para cancelar o mergeamento.

no caso de conflito este comando serve para cancelar o mergeamento.

Importante para identificar o Usuario (É preciso indenticar apenas uma vez)
é por meio do e-mail que vemos se a usuario tem permissão para acessar os arquivos

Estado = Untracked (não rastreado)
Estado = Staged (preparado)

Estado = Unmodified (não modificado). Quando for alterado o Estado = modified (modificado)

faz um comite e deixa um arquivo para traz. O arquivo modificado e colocado no mesmo commit
tracked (Unmodified, Modified, Staged)

HEAD mostra qual é commit mais recente. (qual é a cabeça dos commit's)

estar na area de preparação - Staged)
area de preparação - Staged)

o parametro.