**Resumo**

|  |  |
| --- | --- |
| **Configuração**  # para a maquina toda  git config --global user.email "you@example.com"  git config --global user.name "Your Name"  # para um projeto especifico  git config --local user.email "you@example.com"  git config --local user.name "Your Name"  #verificar  git config user.name  git config user.email | **admin server**  # add server  git remote add <name> <new-URL>  # mostra endereço do servidor  git remote -v  # remove server  git remote remove origin  # servidor local: vá até a pasta do servidor e digite  git init --bare, cria servidor repositório puro  git remote add repository\_name diretorio (ex: git remote add local C:/Users/git-e-github/servidor) |

|  |  |
| --- | --- |
| **commit**  # adcionar para commit  git add .  #comitar  git commit -m “mensagem”  # pegar alterações feitas por outras pessoas  git pull  # enviar para o servidor  git push origin  # -u define que sempre que usarmos git push e estivermos na master, o envio será feito para origin  git push -u origin master | git status  #Resturar arquivos  git restore file\_name  **Ignorar arquivo depois do commit**  Caso não tenha feito o commit ainda, utilize:  git reset <arquivo-ou-diretório>  Caso já tenha realizado um commit, utilize: git rm --cached <arquivo> git rm -r --cached <diretório> |

|  |  |
| --- | --- |
| **Branch**  https://git-school.github.io/visualizing-git/  # mostra as branchs  git branch  # criar uma branch  git branch titulo  # trocar de branch  git chechout -b titulo  # muda e cria branch  git checkout -b titulo (-b cria a branch)  #deletar branch (-d ou –delete)  git branch –d  O checkout, Este comando também é utilizado para dispensar mudanças de um arquivo;  Alterando o branch podemos levar alterações que não foram commitadas junto, tome cuidado! | **Merge**  # fazer um merge da branch titulo com a main.  git checkout main git merge titulo  # juntar os arquivos sem perder a log dos commits  git rebase titulo |

|  |  |
| --- | --- |
| **Stash**  # Após o comando o branch será resetado para a sua versão de acordo com o repositório.  git stash  # lista  git stash list  # resgata o rascunho  git stash apply <number>  # mostrar a alteração de cada uma  git stash show –p <number>  # deletar rascunho  git stash drop <nome>  # deletar todos os rascunhos  git stash clear | **Merge**  # fazer um merge da branch titulo com a main.  git checkout main git merge titulo  # juntar os arquivos sem perder a log dos commits  git rebase titulo |

# Caso haja algum conflito, como o branch a ser deletado estar à frente do branch atual, ou seja, se novo-release tivesse alguns commits à frente do master

git branch -D

**gráfico**

git graph (vs code)

gitLens

https://git-cola.github.io/

<https://www.gitkraken.com/>

git rm para remover o arquivo

################################################# log

git log  
git log --oneline

# ver modificações  
git log -p

# mostra só os 2 ultimos comittes   
git log -n 2

https://devhints.io/git-log

https://git-school.github.io/visualizing-git/

git push origin

# juntar os arquivos sem perder a log dos commits

#remover branch

git branch -d

# Caso haja algum conflito, como o branch a ser deletado estar à frente do branch atual, ou seja, se novo-release tivesse alguns commits à frente do master

git branch -D

################################################## Desfazer

# para desfazer alteração que ainda não foram comitadas

git checkout --filename (ex: git checkout --index.html)

# para desfazer uma alteração que for marcada para comite com add

git reset HEAD filename ( ex. git reset HEAD index.html )

# defazer uma alteração já comitadas. Git log para pegar o hash

git log

git revert hash (ex: git revert 545i09239203-0232v0iodfd )

git log --graph

# confirmar alteração

:x + enter

#################### guardar alteração em outro arquivo para continuar depois

git stash

# lista os arquivos guardados

git stash list

# restaurar um arquivo que estava guardado

git stash apply numero\_arquivo (ex git stash apply 0) -stash@{0}

#é preciso deletar o arquivo guardado

git stash drop

#### voltar para um commit X

git log --oneline

git checkout 54727de

# mostrar alteração que ainda não foram comitadas

git diff

# Comparar comites

git log -p

git diff ea539b3..6ca12ac