



GIT & GITHUB

GIT

iniciando repo:

```
git init
```

verificando status:

```
git status
```

mostrar ao git:

```
git add + nome do  
arquivo
```

selecionar todos os arquivos da pasta:

```
git add .
```

comitar 1 arquivo:

```
git commit + nome  
do arquivo -m "men-  
sagem para o  
commit"
```

comitar +1

arquivos:

git commit -m

“mensagem para os
commits”

GITHUB

adicionar projeto
ao

github:

```
new repository; dar  
nome  
e descrição ;  
create  
repository.
```

no git:

```
git branch -M main
```

```
git remote add  
origin link do  
repositório do  
github
```

primeiro push:

```
git push -u origin  
main
```

como sincronizar
os repositórios:

```
git pull
```

no github:

clonar
repositórios:

code, https e
copiar link

no git:

```
git clone link  
copiado do github
```

copiar tudo na
pasta diretamente

```
git clone link  
copiado do github .
```

detalhamento
cronológico:


```
git log
```

usar “enter”

para avançar a
visualização e
letra “q” para
sair

```
git log --
```

```
author="nome do  
author  
desejado"
```

mensurar

contribuições:

`git shortlog`

mensurar

contribuições

versão reduzida:

`git shortlog -sn`

o que foi feito

naquela

contribuição:

git show + hash do
commit

para visualizar
modificações
antes delas
serem
commitadas:

git diff

para ver
somente o nome
do arquivo
modificado:

```
git diff --name-  
only
```

retornar
arquivo para
antes da edição
recente:

git checkout +
nome do arquivo

tirar arquivo
de staged:

git reset HEAD +
nome do arquivo

retirar
modificações

```
git reset --soft  
ou --mixed ou --  
hard + hash do  
commit
```

que será mantido
mas o de cima será
modificado



altera o
histórico do
commit

criando branch:

```
git checkout -b +  
nome que quiser
```

mostrar os branch
existentes:

```
git branch
```

o “*” mostra
aonde você está

ir para um branch:

```
git checkout +  
nome do branch
```

deletar branch:

```
git branch -D +  
nome do branch
```

```
git merge: juntar  
branches
```



```
git merge +  
nome do  
branch
```

```
git log --graph
```

usando rebase:

```
git rebase +  
nome do branch
```

mostrar quais
arquivos tem
dentro da
pasta:

ls

como ignorar
arquivos no
git:

criar uma pasta:

`vi .gitignore`

`i` para inserir

ignorar todos d
euma extensão
específica com
asterisco

- `+ nome da
extensão`

apagando linha:

td

salvando:

esc + w + q

ignorar um
arquivo
específico:

entrar no git
ignore e colocar
o nome do arquivo

salvar um
arquivo para
commitar depois:

git stash

para aplicar as
mudanças do git
stash:

git stash apply

lista dos git
stash:

```
git stash list
```

limpar tudo que
estiveer no
stash:

```
git stash clear
```

criar atalhos:

```
git config --  
global alias.s  
status
```

o “s” é a tecla

o status é o que
a tecla vai fazer

criar tags:


```
git tag -a
```

```
1.0.0 -m
```

```
“header
```

```
finalizado”
```

```
subir tags no repo:
```

```
git push origin
```

```
master --tags
```

reverter
mudanças no
commit:

git revert +
hash do commit
que deseja
reverter

deletar tag
localmente:

```
git tag -d +  
versão da tag/  
nome
```

deletar tag
remoto:

```
git push  
origin :  
+versão da  
tag/ nome
```

```
deletar  
branch  
remoto:
```

```
git push origin  
:+ nome do  
branch
```


