

começando Iniciando as operações**git config** Configura as credenciais do usuário.

comando

```
/ git config --global user.name "João Silva"
/ git config --global user.email "joao.silva@gmail.com"
```

git clone Utilizado para criar uma cópia de um repositório remoto.

comando

```
/ git clone https://github.com/awescolar/dwai-2023.git
```

```
Cloning into 'dwai-2023'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
```

git init Utilizado para iniciar o Git em um diretório.

comando

```
/dwai-2023/ git init
```

```
Initialized empty Git repository in c:/Users/{matricula}/dwai/.git/
```

produzindo trabalhando com o git**git status** Verifica se há arquivos para serem incluídos.

comando

```
/ git status
```

```
On branch master
No commits yet
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    atv1_{seunome}.html
```

git add Adicionando arquivos.

git add *

comando

```
/ git add <atv1_nome_aluno.html>
```

git commit Preparando arquivos para envio.

comando

```
/ git commit -m "Eu sou {nome_aluno} e estou enviando
minha primeira atividade."
```

```
[master a8f7582] Eu sou {nome_aluno} e estou enviando minha primeira
atividade
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 atv1_{nome_aluno}.html
```

git push Enviar os dados para o repositório.

comando

```
/ git push
```

```
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Writing objects: 100% (3/3), 279 bytes | 279.00 KiB/s, done.
Total 3 (delta 0), reused 1 (delta 0), pack-reused 0
remote: - Changes must be made through a pull request.
remote:
To https://github.com/awescolar/dwai.git
7f36686..a8f7582 master -> master
```

colaborando trabalhando em equipes**git branch** Trabalhando com *branches*

comando

```
/ git branch lista todas as branches
/ git branch <nome> cria uma branch
/ git branch -d <nome-branch> exclui a branch
```

```
cesimar
* cesimar-xavier
master
```

git checkout Navegando entre *branch*.

comando

```
/ git checkout -b <nome-branch> <nome-referencia> cria e
seleciona uma branch a partir de uma referência
/ git checkout <nome-branch> seleciona uma branch
/ git push origin <nome-branch> replica a branch no repo
```

```
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
```

git pull Atualiza o código para a versão mais recente do repo.

comando

```
/ git pull
```

```
Already up to date.
```

git merge Mescla arquivos de *branches* diferentes. É preciso mudar para *branch* que irá receber a mesclagem antes de executar.

comando

```
/ git checkout <branch-destino>
/ git merge <branch-origem>
```

```
[master a8f7582] Eu sou {nome_aluno} e estou enviando minha primeira
atividade
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 atv1_{nome_aluno}.html
```

gerenciando lidando com reestruturações.**git reflog** Mostra o histórico de arquivos trabalhados.

comando

```
/ git reflog
```

```
2bcf3db (HEAD -> cesimar-xavier, origin/cesimar-xavier) HEAD@{0}: commit:
cde0a99 HEAD@{1}: commit: Adicionei o arquivo de Footer
0f0b916 HEAD@{2}: commit: Inclusão do menu
de8d526 HEAD@{3}: checkout: moving from master to cesimar-xavier
bf6738d (origin/master, origin/HEAD, master) HEAD@{4}: commit: Arquivo com
a8f7582 (cesimar) HEAD@{5}: checkout: moving from cesimar-xavier to master
de8d526 HEAD@{6}: commit: Estutura HTML foi criada
```

git reset Para quando quiser restaurar para uma versão específica.

comando

```
/ git reset --hard <numero-id>
```

```
HEAD is now at de8d526 Estutura HTML foi criada
```

outros comandos úteis

```
/ git alias / git diff / git stash / git blame / git clean
/ git revert / git rm / git rebase
```

Todos estes comandos podem ser consultados em:
https://training.github.com/downloads/pt_BR/github-git-cheat-sheet/