

Primeiros comandos com o GIT

Iniciar o GIT

GIT INIT

Criando um repositório

1. Criamos uma pasta que usaremos ao decorrer do curso “livro-receitas”
2. Inicializamos o versionamento da pasta com o git init
 - .git: é uma pasta gerencial oculta por tanto ela não aparecer.
 - Para vê-la usamos uma flag especifica do ls que a ls -a
3. Configuração Inicial (se é a primeira vez usando)
 - Informa Autor (para o commit)
 - git config --global user.email “email”
 - git config --global [user.name](#) “nome”
4. Criar e Adicionando o primeiro arquivo do livro de receitas
 - Markdown
 - Um arquivo de uma forma mais humana de se escrever um arquivo HTML.
5. Basta cria um arquivo .md

Iniciar o versionamento

GIT ADD

- Git add *

```
Raylen Costa@DESKTOP-7P05DMR MINGW64 /c/workspace/livro-receitas (master)
$ git add *

Raylen Costa@DESKTOP-7P05DMR MINGW64 /c/workspace/livro-receitas (master)
$ git commit -m "commit inicial"
[master (root-commit) cdd673a] commit inicial
1 file changed, 5 insertions(+)
create mode 100644 strogonoff.md
```

Criar um commit

GIT COMMIT

- git commit -m ""
 - Mensagem que após temos acesso a lista de alterações, conseguiremos localizar e identificar facilmente qual e qual commit que fizemos e ter acesso a essa mensagem curta para entender o que aquele commit significa dentro das nossas alterações/dentro dos nossos projetos.

Ciclo de vida dos arquivos no GIT

Passo a passo no ciclo de vida

- **GIT INIT:** Comando, além de criar aquela pasta “.git/”, ela inicializa um conceito do git chamado “repositório”.
 - Quando usamos o “git init” estamos de fato criando um repositório dentro de um diretório, dentro de uma pasta.
- **Tracked ou Untracked:**
 - **Untracked:** São arquivos que o git ainda não tem conhecimento deles.
 - **Tracked:** São os arquivos que o git tem conhecimento deles.
 - Dentro de Tracked, dentro dos arquivos que são rastreáveis de fato pelo git, ele pode se subdividir em 3 estágios:
 - Unmodified: É o arquivo que ainda não foi modificado.
 - Modified: É o arquivo “Unmodified” que sofreu modificação.
 - Staged: É um conceito chave para entendermos, é os arquivos que estão se preparando para poder fazer parte de um outro tipo de agrupamento.
- **Repositórios:**

Nesse momento temos dois ambientes:

 1. **O ambiente de Desenvolvimento (tudo que está na nossa máquina)**

Ele tem uma versão de servidor na nossa máquina, então as alterações que você faz no seu código, na sua máquina, ele não repercute imediatamente no seu repositório remoto.

A não ser que seja executado um conjunto de códigos específicos que empurra a alteração do seu repositório local para o seu repositório remoto.
 2. **O servidor**

O Git tem sua versão de servidor, que é o GitHub.
- **GIT STATUS**

Monitora exatamente os conceitos apresentados os status dos arquivos (Untracked, Modified ou Staged).

Quando o GIT não sabe de onde vem a pasta ele coloca como “Untracked”.

 - Se quisemos mudar novamente os status dos arquivos para unstaged - git restore --staged <nome_arquivo>