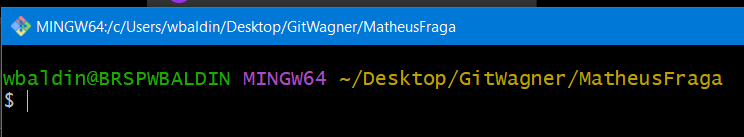
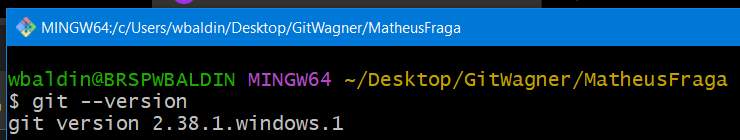
## First-Time Git Setup

<https://git-scm.com/book/en/v2/Getting-Started-First-Time-Git-Setup>



git --version

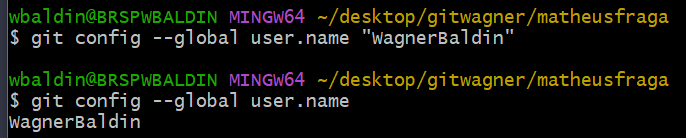


Clear

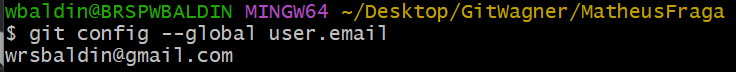
**Your Identity**

git config --global user.name "WagnerBaldin"

**Definindo o usuário e depois visualizando esse usuário.**



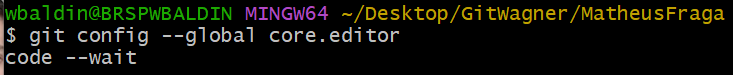
git config --global user.email [wrsbaldin@gmail.com](mailto:wrsbaldin@gmail.com)



### Your Editor

<https://git-scm.com/book/en/v2/Appendix-C%3A-Git-Commands-Setup-and-Config#ch_core_editor>

git config --global core.editor "code --wait"



O que é um repositório?

É o local onde os arquivos dos projetos ficam para serem gerenciados.

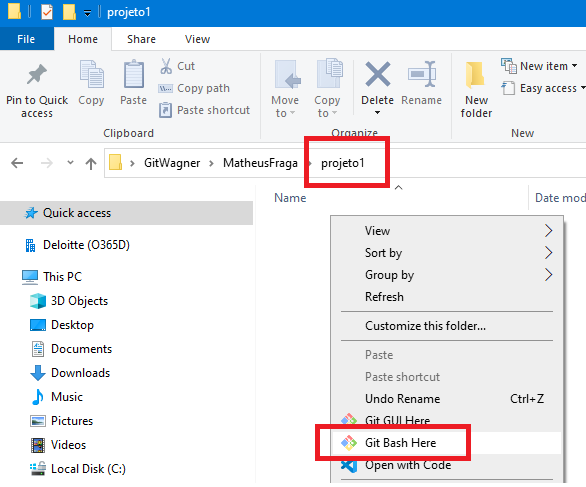
Para cada projeto nós criamos um repositório para ele.

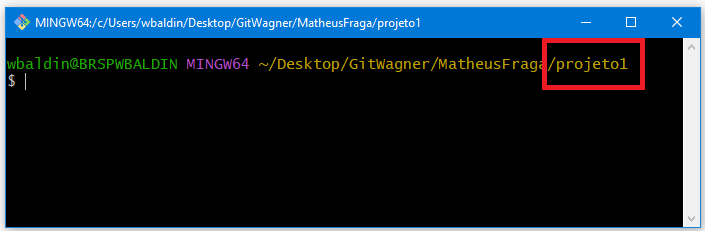
Os servidores de hospedagens mais famosos para gerenciarmos os repositórios são o Github e o Bitbucket.

Iniciando um repositório:

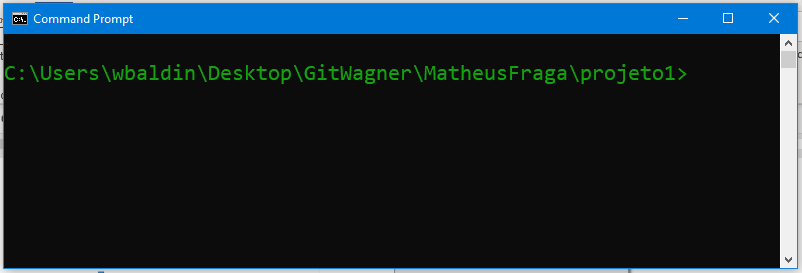
Criar uma pasta para o projeto e abrindo o Terminal do Git dentro dessa pasta:

Abrindo o Git dentro da pasta criada:

só clicar aqui



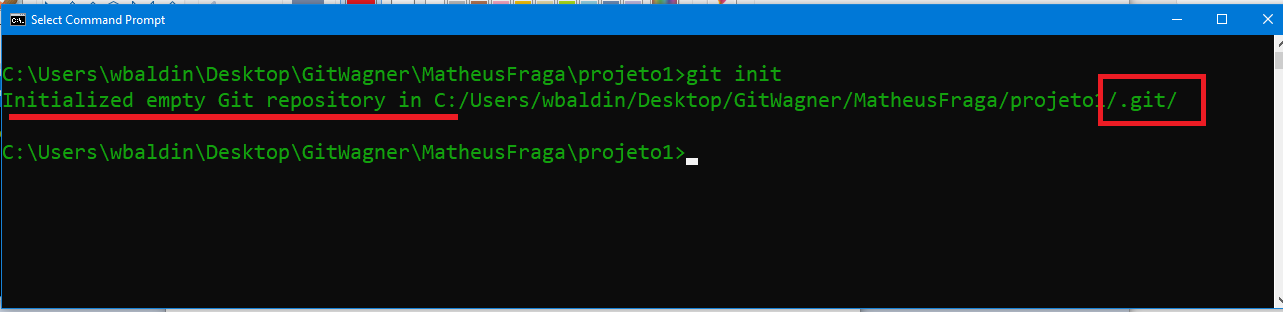
No CMD do Windows



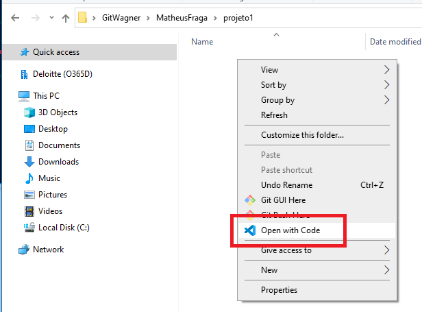
Git init

Iniciando o repositório no CMD e automaticamente ele cria a pasta oculta chamada /.git/ dentro da pasta do projeto.

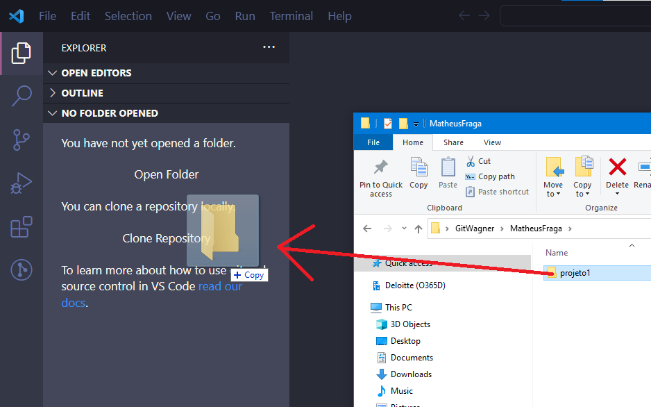
Essa pasta é onde fica nossas configurações e ela é oculta.



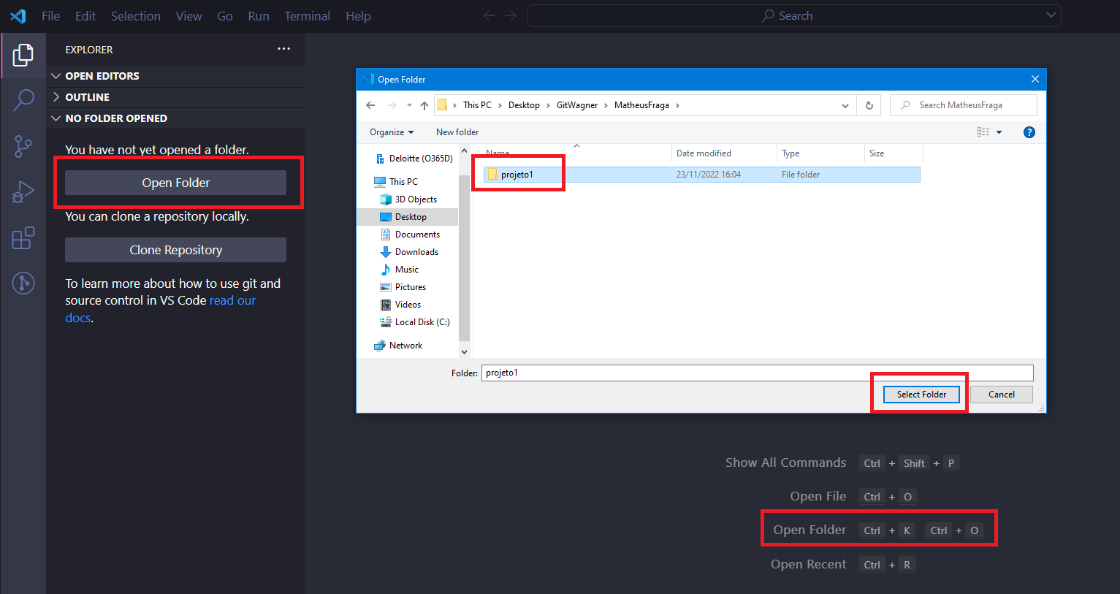
Abrindo essa pasta de projeto dentro do VS Code

Clicando dentro da pasta

Ou

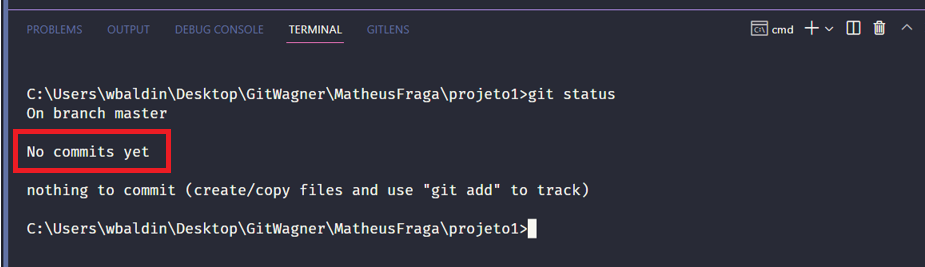
 Arrastanto e soltando a pasta na área de explorer do VSCode

Ou clicando em uma dessas opções.



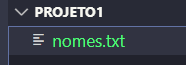
Vendo o Status no CMD dentro do VSCode:

Git status

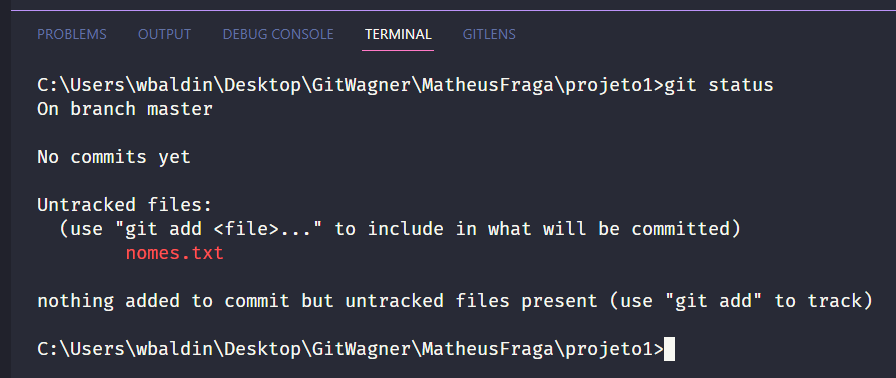


Como não tem arquivos e nada para commitar, ele mostra que não temos commit e a pasta está vazia

Criando um arquivo TXT

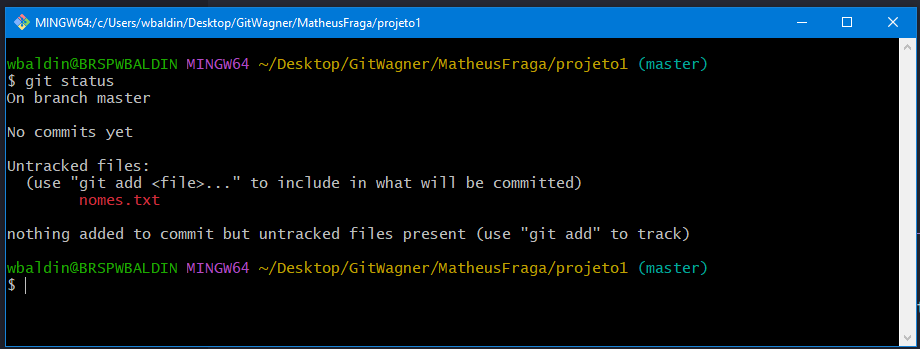


Git status novamente

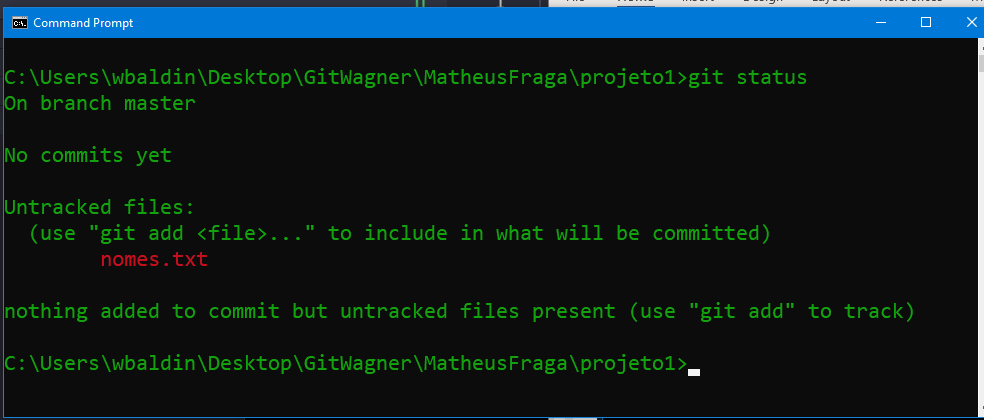


Ele mostra o arquivo não trackeado

Vendo a mesma tela no gitbash



No CMD no Windows



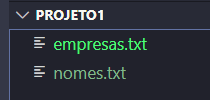
Basicamente a mesma coisa

Seguimos no VSCode por enquanto...

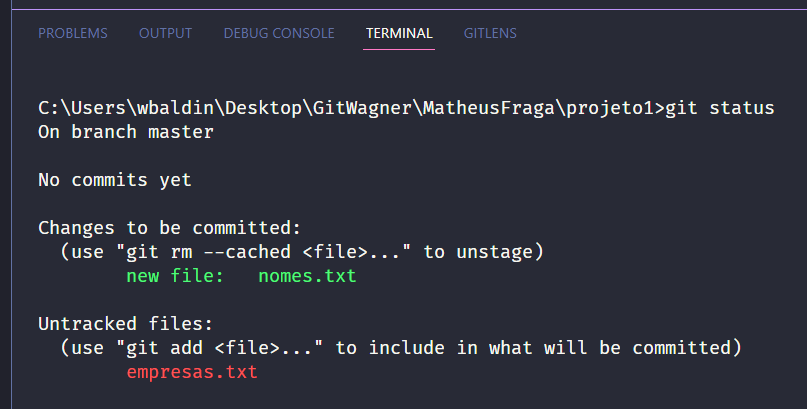
Criando mais um arquivo e adicionando

Git add NomeDoArquivo.ext faz o track do arquivo específico para ser commitado.

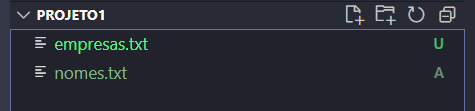
Git add nomes.txt



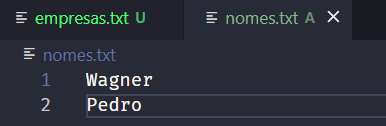
Agora tenho um arquivo trackeado pronto para ser commitado e um arquivo ainda não commitado

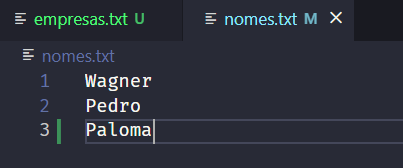


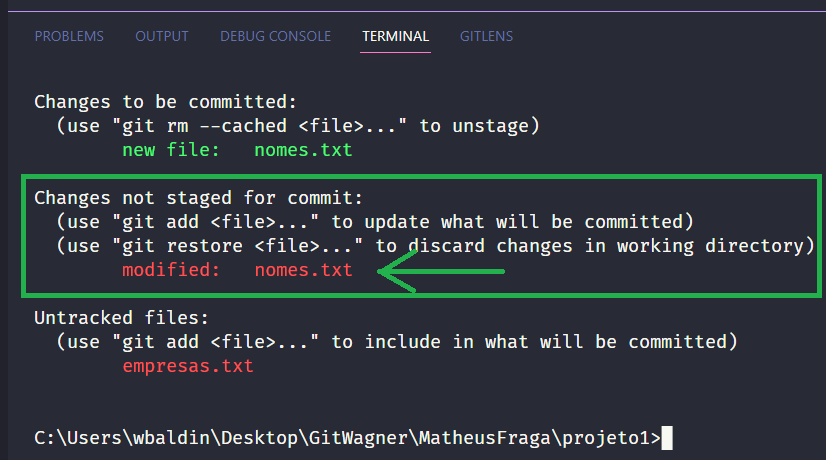
No VSCode vemos os arquivos com os sinais de U e A para mostrar o arquivo Adicionado e o arquivo Untracked



Fazendo alteração no arquivo nomes.txt

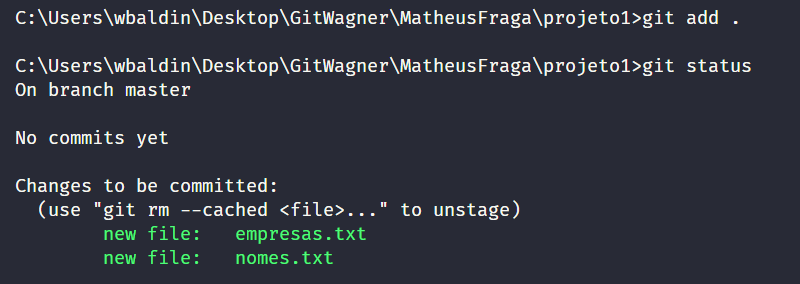
 Adicionando mais um nome:

 Git status novamente



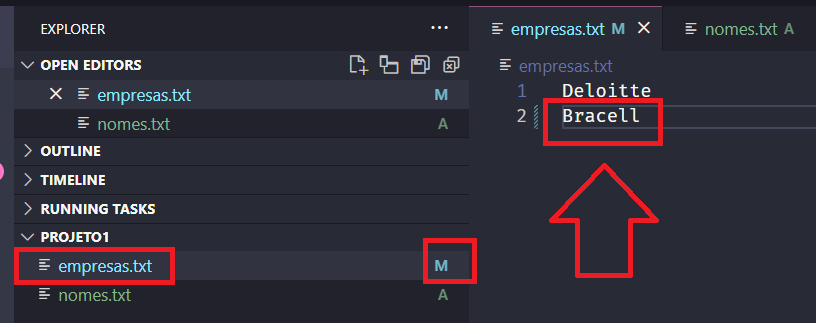
Adicionando TODOS os arquivos de vez para serem commitados de uma só vez e depois um novo Git Status para acompanhar as alterações.

Git add .

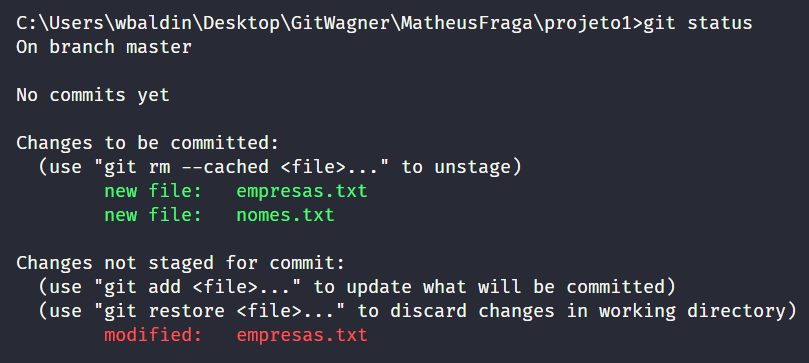


Alterando um desses arquivos.

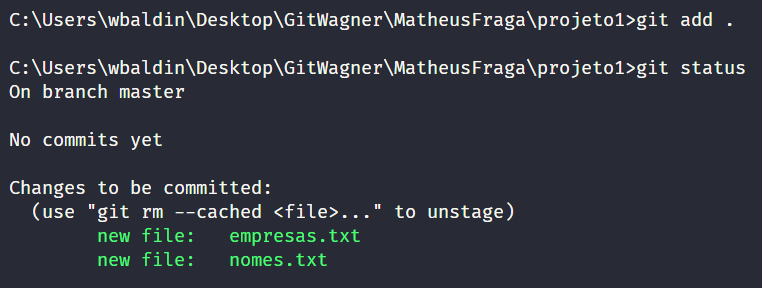
Adicionando uma informação qualquer em um dos arquivos o VSCode identifica automaticamente com um M os arquivos modificados



Ao verificar novamente com o git status, o Git mostra em que arquivo houve uma alteração



Basta dar o comando git add. E git status novamente que tudo normaliza, pronto para ser feito o commit



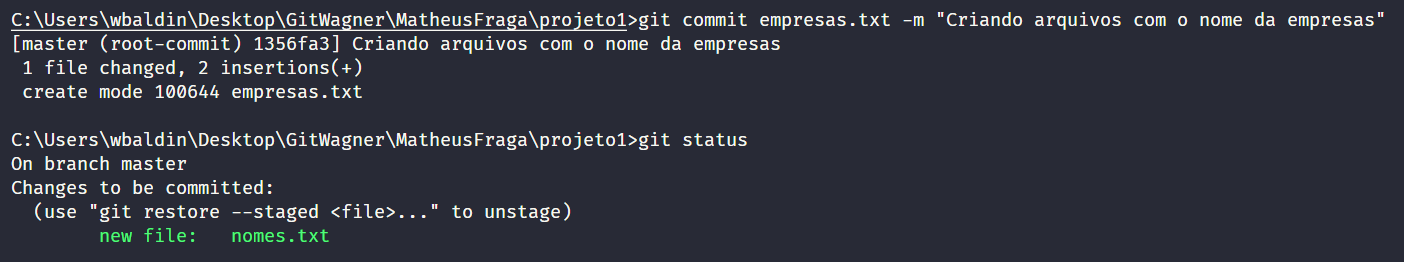
O que é um committ ?

É basicamente salvar os arquivos no nosso repositório criado.

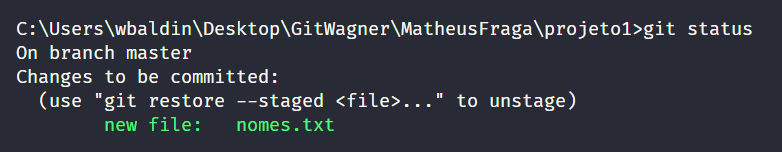
Salvando apenas um arquivo por vez e depois vendo o Status desse commit

git commit empresas.txt -m "**Criando arquivos com o nome da empresas**"

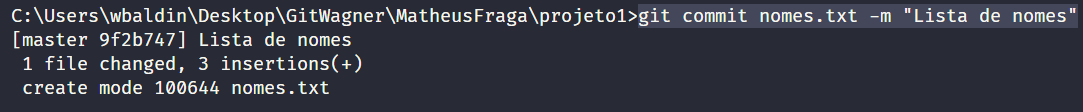
A mensagem deve ser intuitiva e dizer o que estou fazendo para que depois qualquer pessoa saiba o que foi feito e em que ponto se encontra o projeto.



Vemos aqui o arquivo ainda sem commit

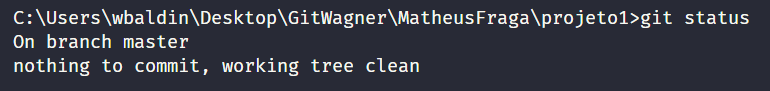


git commit nomes.txt -m "Lista de nomes"

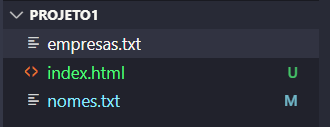


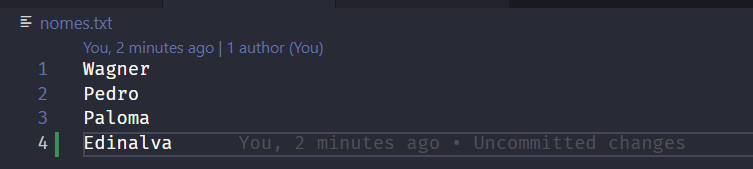
Novo git status

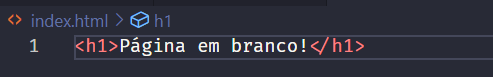
Nada para commitar e a árvore do diretório está limpa



Criando um novo arquivo e alterando um dos arquivos anteriores já criados.

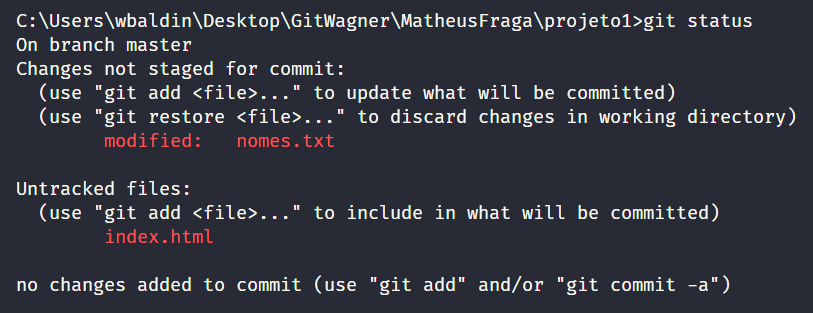






Git status

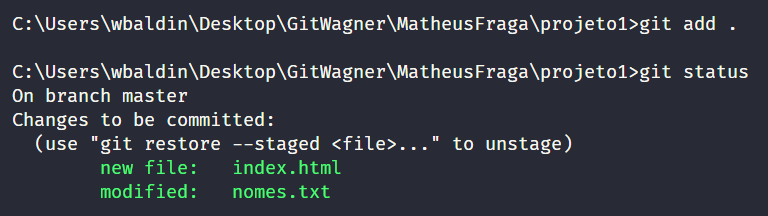
Vemos o arquivo modificado e o novo arquivo não trackeado



Git add.

Git status

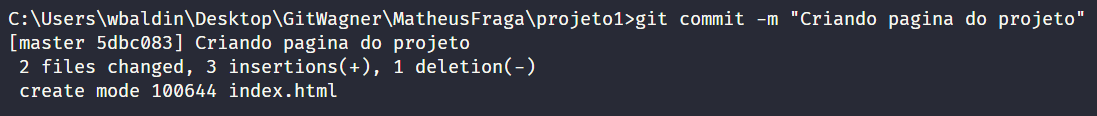
Deixando tudo pronto para o commit.



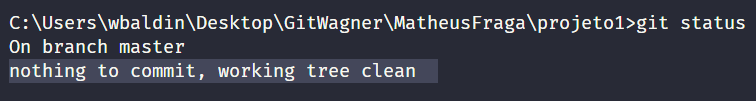
Fazendo o commit de todos os arquivos de uma só vez e inserindo a mesma descrição para todos

git commit -m ***"Descriçaõ da Feature"***

git commit -m "Criando pagina do projeto"



Git status



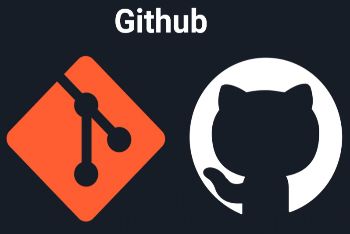
Nada para commitar e árvore limpa

Num projeto real, normalmente não se faz o commit de arquivo por arquivo, é feito um commit só para todos os arquivos de uma vez só.

Pergunta 1:

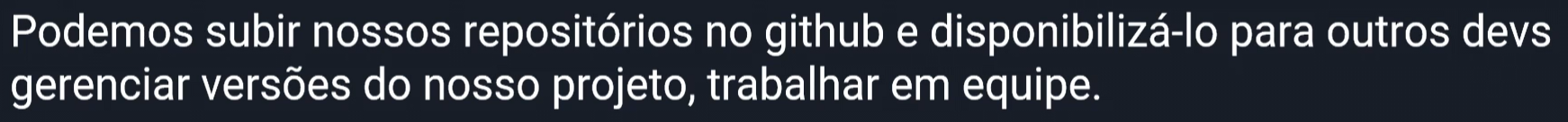
Qual comando usamos para verificar o status do repositório??







É como se fosse também uma “Rede Social” para desenvolvedores



É possível trabalhar em conjunto no github

12. Enviando repositório pro github

[git@github.com:wagnerbaldin/curso\_github.git](mailto:git@github.com:wagnerbaldin/curso_github.git)

Sugestão do github:

**…or create a new repository on the command line**

echo "# curso\_github" >> README.md

git init

git add README.md

git commit -m "first commit"

git branch -M main

git remote add origin git@github.com:wagnerbaldin/curso\_github.git

git push -u origin main

### …or push an existing repository from the command line

git remote add origin git@github.com:wagnerbaldin/curso\_github.git

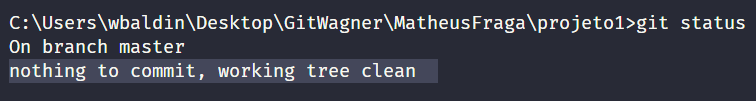
git branch -M main

git push -u origin main

### …or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Após iniciasr o repositório, adicionar os arquivos e dar o comit, ficamos com a árvore limpa



Vamos enviar os arquivos para o Github

Inserindo o comando da Brench principal, a Main

git branch -M main

Adicionando a Origin, ou seja, a origem de onde os arquivos vão sair.

git remote add origin [git@github.com:wagnerbaldin/curso\_github.git](mailto:git@github.com:wagnerbaldin/curso_github.git)

Agora fanzendo o Push dos arquivos no repositório

git push -u origin main

Atenção!!!!

Quando aparecer uma mensagem de erro não deixando

<https://www.youtube.com/watch?v=en1dycYtL48>

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Chave adicionada

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Palavra passe no git: Wrsb159753#

Criar o repositório primeiro no git e fazer o clone dele em seguida com a chave SSH

Assim não preciso fazer o Git Init, porque o relacionamento já foi feito com o git Clone

git clone [git@github.com:wagnerbaldin/curso\_git.git](mailto:git@github.com:wagnerbaldin/curso_git.git)

No Git Bash

Criando um arquivo

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Listando os arquivos do diretório: ls ou Dir

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Fazendo o primeiro commit nessa pasta nova

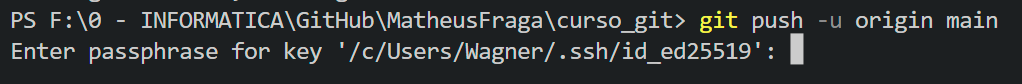
Obs: pode usar tanto aspas simples como duplas nos commits

‘’ ou “”

git commit -m 'primeiro commit de arquivos'

FAZENDO UM GIT PUSH – Empurrando os arquivos la para o Github

git push -u origin main

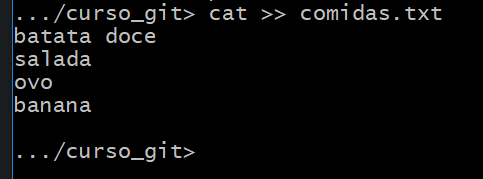
 senha

Agora foi!!!!

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Adicionando informações no arquivo txt direto do git bash

 Ctrl+c pra fechar

Depois que fiz o primeiro PUSH com o comando: git push -u origin main

Da segunda vez em diante não preciso mais escrever o comando todo, pois o caminho já foiestabelecido, basta apenas o:

git push

Texto

Descrição gerada automaticamente

E todos os arquivos estão no github

Uma imagem contendo Texto

Descrição gerada automaticamente