# APRENDA A USAR O GIT HUB

**Autor**: João Henrique Domingos

**I.A utilizada**: chatgpt



**O QUE É O GIT HUB:**

O GitHub é uma plataforma de hospedagem de código baseada em Git. Ele permite que desenvolvedores armazenem, gerenciem e colaborem em projetos de software. Com ele, é possível versionar código, trabalhar em equipe, revisar alterações e automatizar processos. Além disso, o GitHub facilita o desenvolvimento open source e a integração com diversas ferramentas de CI/CD.



**CRIANDO UMA CONTA NO GIT HUB:**

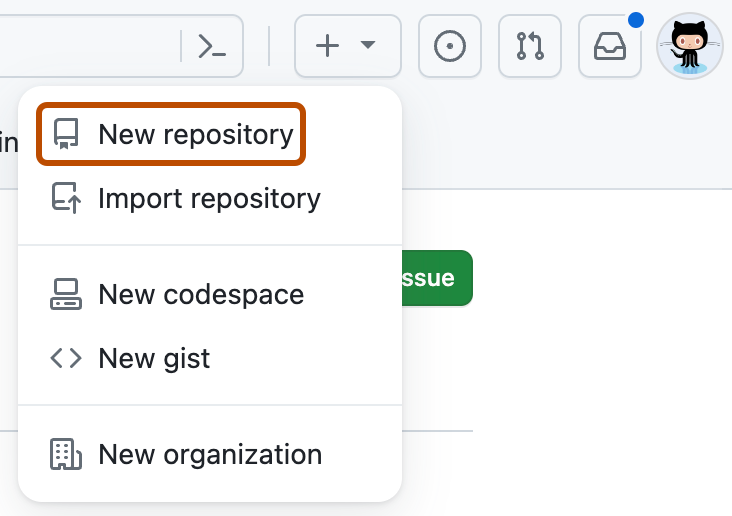
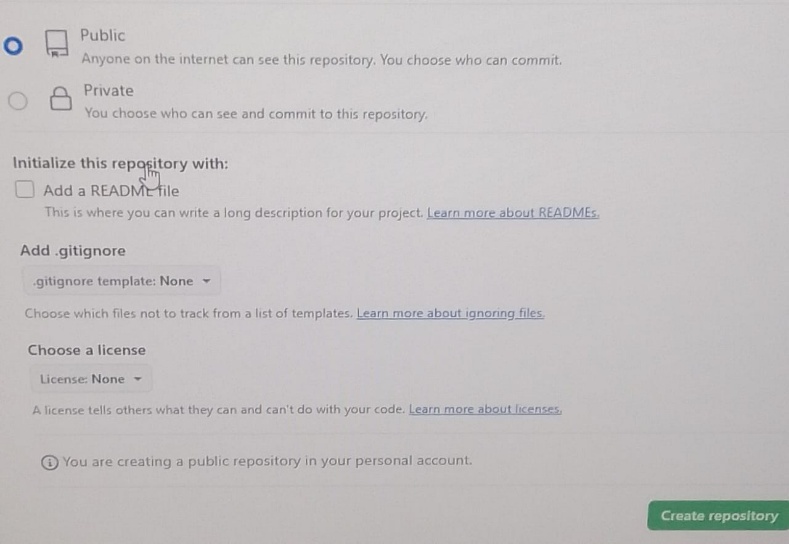
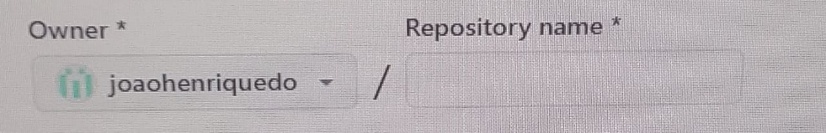
Acesse [GitHub](https://github.com/) e cadastre-se. Após isso, confirme seu e-mail e configure seu perfil.



**CRIANDO UM REPOSITORIO:**

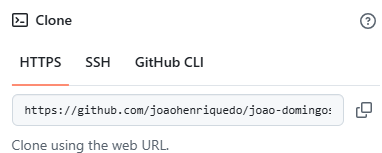
**No GitHub:**

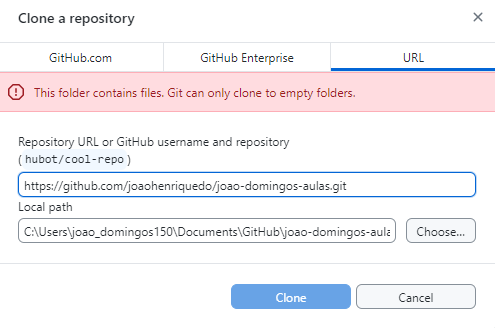
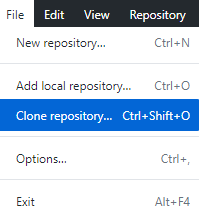
1. Clique no botão "New repository"
2. Dê um nome ao repositório
3. Escolha entre público ou privado
4. Clique em "Create repository"

**CLONANDO UM REPOSITORIO:**

1. Acesse o repositório no GitHub.
2. Clique no botão verde **"Code"**.
3. Copie a URL fornecida (HTTPS ou SSH).
4. Abra o GitHub Desktop.
5. Clique em "File" **>** "Clone Repository" ou pressione Ctrl + Shift + O (Cmd + Shift + O no Mac).
6. Escolha a pasta onde deseja salvar o projeto.
7. Clique em **"Clone"** e aguarde o download dos arquivos.





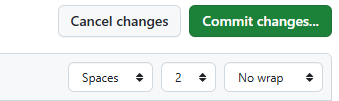
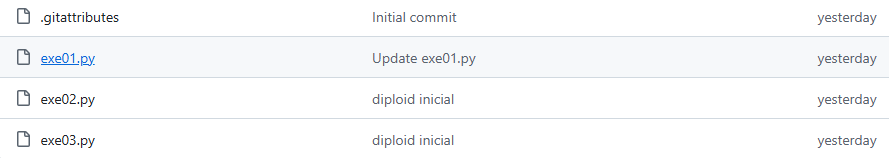
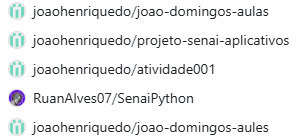
**EDITANDO UM ARQUIVO DIRETAMENTE NO GIT HUB:**

1 Acesse o repositório no [GitHub](https://github.com/).

2 Clique no arquivo que deseja editar.

3 Clique no ícone "Editar" (lápis) no canto superior direito.

4 Faça as alterações desejadas.

5 Role para cima e clique em "Commit changes" para salvar

**O QUE É BRANCH:**

Um branch (ou "ramo") no Git é como uma linha paralela de desenvolvimento dentro de um repositório. Ele permite que você trabalhe em novas funcionalidades sem afetar o código principal.

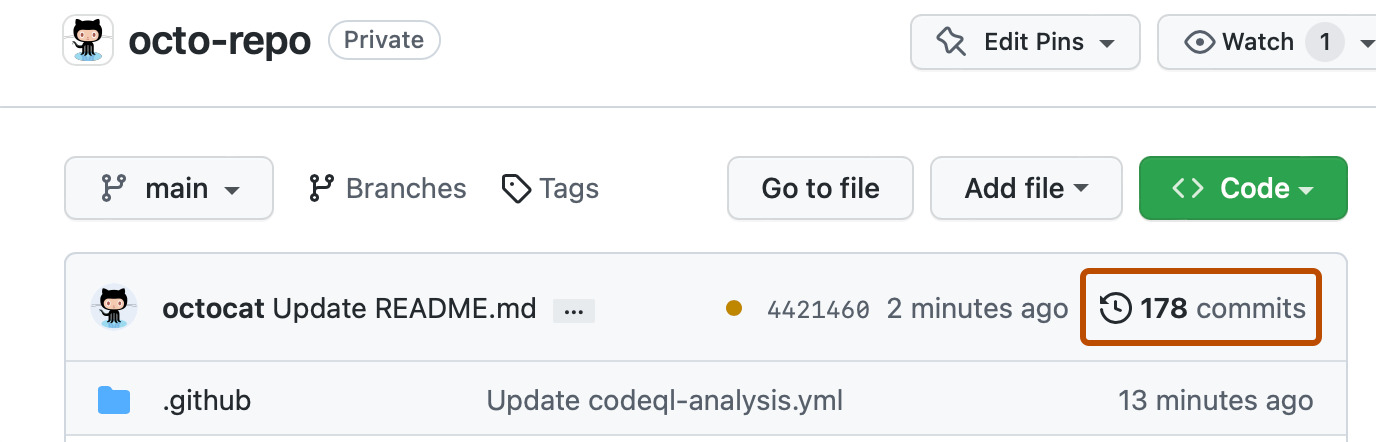
Exemplo: Imagine que você tem um projeto funcionando e quer adicionar uma nova funcionalidade. Em vez de modificar diretamente o código principal (geralmente chamado de main ou master), você cria um branch separado, faz suas alterações lá e só depois junta com o código principal.



**O QUE É UM COMMIT**:

Um **commit** no Git é como um "salvamento" no histórico do seu projeto. Ele registra as mudanças feitas nos arquivos e cria um **ponto de restauração**, permitindo que você volte para esse estado no futuro, se necessário.

**Pense no commit como uma foto do seu código naquele momento!**



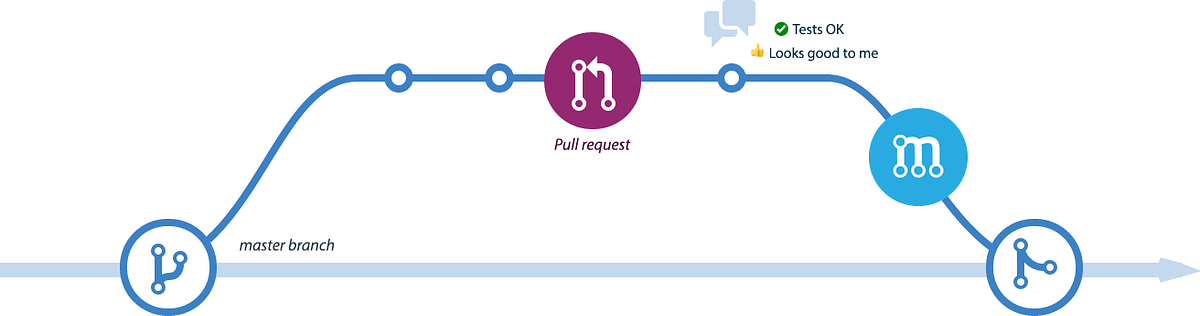
**O QUE É UM PULL REQUEST:**

Um Pull Request (PR) é um pedido para mesclar (juntar) as mudanças de um branch para outro no GitHub.

Geralmente, é usado para sugerir e revisar mudanças antes de adicioná-las ao código principal (main ou master).

**Como Funciona um Pull Request?**

1️ Você cria um branch e faz alterações no código.  
2️ Faz commits e envia (push) para o GitHub.  
3️ Abre um Pull Request para sugerir a mudança.  
4️ Outros desenvolvedores revisam o código e podem aprovar, pedir ajustes ou recusar.  
5️ Se aprovado, o PR é mesclado (merged) no branch principal.



**O QUE É UM MAIN:**

O main (ou master em versões mais antigas) é o branch principal de um repositório Git. Ele geralmente contém a versão mais estável e pronta para produção do projeto.

**CARACTERISTICAS DO MAIN**

É o ponto central do desenvolvimento.

1. Normalmente, todas as alterações passam por revisão antes de serem adicionadas a ele.

Branches secundários (como feature-x ou bugfix-y) são criados para testar novas funcionalidades sem afetar o main.

Após testes e revisões, mudanças de outros branches são mescladas (merged) no main por meio de um Pull Request.