

# Primeiros Passos no GitHub

Seu Guia Rápido para o Controle de  
Versão e Colaboração



# Bem-vindo ao Mundo do GitHub!

Este guia descomplica o GitHub, mostrando como ele pode simplificar sua vida como desenvolvedor.

Prepare-se para dar seus primeiros passos e transformar a maneira como você desenvolve.



# O Que é GitHub e Por Que Você Precisa Dele?

**GitHub** é uma rede social para desenvolvedores e um "super HD" na nuvem para seus projetos, com histórico de todas as alterações.

## Por que você precisa do GitHub?

**Controle de Versão:** Salve instantâneos do seu código.



**Colaboração:** Trabalhe em equipe sem conflitos.



**Portfólio:** Seus projetos como currículo online.



**Backup:** Código seguro na nuvem, acessível de qualquer lugar.



# Seu Primeiro Repositório

Um **repositório** é onde seu projeto e todo o histórico de alterações vivem.

1. Acesse **github.com**, faça login.
2. 2. Clique em "+" > "**New repository**".
3. 3. **Nome**: Escolha algo claro (ex: `meu-primeiro-projeto` ).
4. 4. **Descrição**: (Opcional) Breve explicação.
5. 5. **Visibilidade**: Público ou Privado.
6. 6. Marque "**Add a README.md**".
7. 7. Clique em "**Create repository**".



# Clonando o Repositório


## Trazendo o Código para Sua Máquina

Baixe o repositório online para trabalhar nele localmente.

1. No GitHub, clique “<> **Code**” e copie o URL.
2. No seu **Terminal** (ou Prompt de Comando):

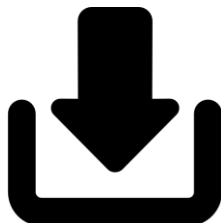


```
git clone [URL_DO_REPOSITORIO]
```



```
git clone https://github.com/GAvila99/prompts-recipe-to-create-a-ebook.git
```

Isso cria uma pasta com seu projeto!

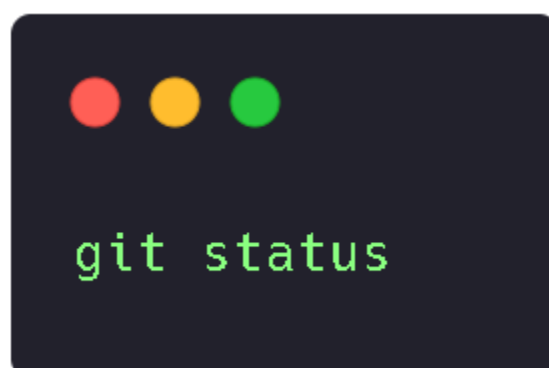


# Seu Primeiro Commit

## Salvando Suas Mudanças

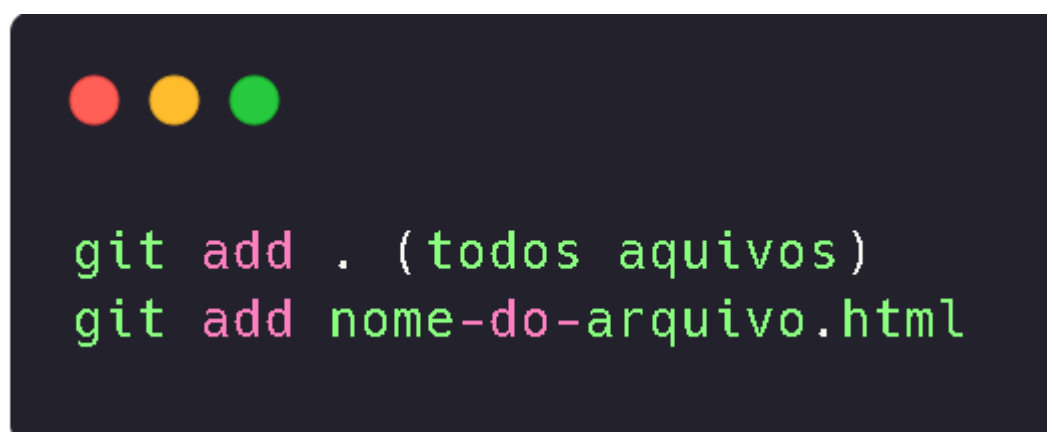
Um **commit** é um "ponto de salvamento" no histórico do seu projeto.

1. **Verifique o status:** Veja o que mudou.

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The text 'git status' is written in a light green monospace font.

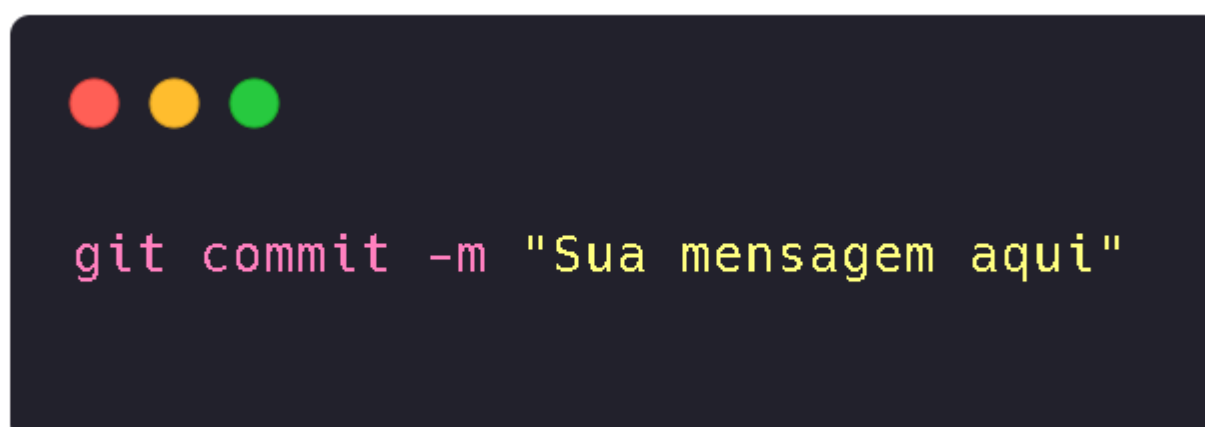
```
git status
```

2. **Prepare as mudanças:** Escolha o que incluir.

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. Two lines of text are shown in a light green monospace font: 'git add .' followed by '(todos arquivos)' in parentheses, and 'git add nome-do-arquivo.html'.

```
git add . (todos arquivos)
git add nome-do-arquivo.html
```

3. **Commite as mudanças:** Salve com uma mensagem clara.

A terminal window with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The text 'git commit -m "Sua mensagem aqui"' is written in a light green monospace font.

```
git commit -m "Sua mensagem aqui"
```

# Enviando Suas Mudanças

## O `git push` para o GitHub

Envie seus commits locais para o repositório online.  
ira como você desenvolve.



**origin:** Apelido para a URL do seu GitHub.

**main:** Nome da Branch principal do projeto.

Suas alterações agora estão visíveis online e seguras!

# Mantendo Seu Código Atualizado

## O git pull

Baixe as últimas atualizações do GitHub para o seu projeto local.

Use antes de começar a trabalhar para garantir que você tem a versão mais recente.

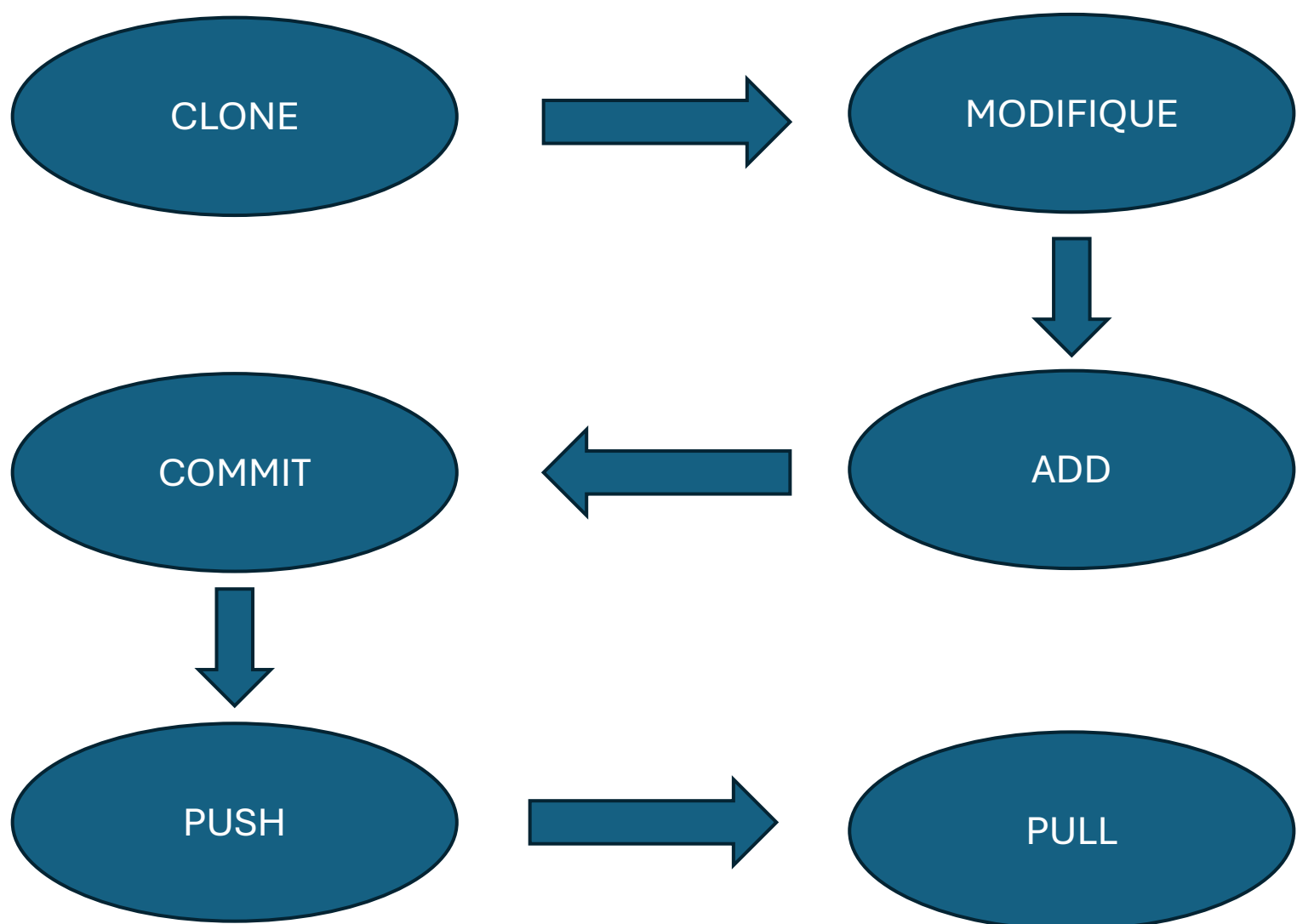
A dark-themed terminal window with three colored window control buttons (red, yellow, green) in the top-left corner. The text 'git pull origin main' is displayed in a light pink font.

```
git pull origin main
```

**Dica:** Sempre puxe antes de começar a codificar!



# Em Resumo: Seu Fluxo de Trabalho Básico



# Onde ir agora?

- Explore mais comandos Git.
- Participe de projetos Open Source.
- Crie mais projetos!