

# Guia Completo Git e GitHub no Terminal

Este documento apresenta os principais comandos do Git e GitHub utilizados no terminal, com explicações detalhadas, passo a passo e procedimentos de reversão de versões. Ideal para versionamento profissional de código.

## 1. Configuração Inicial do Git

```
git config --global user.name "Seu Nome"
git config --global user.email "seuemail@email.com"
```

Define o autor dos commits.

## 2. Inicializar Repositório

```
git init
```

Cria um repositório Git local.

## 3. Verificar Status

```
git status
```

Mostra arquivos modificados e preparados para commit.

## 4. Adicionar Arquivos

```
git add nome_arquivo
git add .
```

Prepara arquivos para commit.

## 5. Criar Commit

```
git commit -m "descrição"
```

Salva alterações no histórico.

## 6. Conectar ao GitHub

```
git remote add origin https://github.com/usuario/repositorio.git
git remote -v
```

Conecta ao repositório remoto.

## 7. Enviar para GitHub

```
git push -u origin main
git push
```

Envia commits para o GitHub.

## 8. Atualizar Projeto

```
git pull
```

Baixa atualizações do GitHub.

## 9. Clonar Projeto

```
git clone https://github.com/usuario/repositorio.git
```

Cria cópia local.

## 10. Branches

```
git branch nome
git checkout nome
git checkout -b nova-branch
```

Gerenciamento de branches.

## 11. Merge

```
git checkout main
git merge nome-branch
```

Une alterações.

## 12. Histórico

```
git log
git log --oneline
```

Visualiza histórico de commits.

## 13. REVERTER PARA VERSÃO ANTERIOR (MUITO IMPORTANTE)

Ver histórico de commits:

```
git log --oneline
```

Copie o código do commit desejado.

Voltar para versão anterior temporariamente:

```
git checkout ID_DO_COMMIT
```

Exemplo:

```
git checkout alb2c3d
```

Voltar definitivamente o projeto para versão antiga:

```
git reset --hard ID_DO_COMMIT
```

Exemplo:

```
git reset --hard alb2c3d
```

ATENÇÃO: Isso apaga alterações atuais.

Reverter commit específico (sem apagar histórico):

```
git revert ID_DO_COMMIT
```

Exemplo:

```
git revert alb2c3d
```

Esse comando cria um novo commit desfazendo o anterior.

Enviar reversão ao GitHub:

```
git push
```

## 14. Boas Práticas

- Sempre usar `git pull` antes de trabalhar
- Commits pequenos e organizados
- Mensagens claras
- Usar branches
- Evitar `git reset --hard` sem necessidade
- Sempre testar antes do push