

# ■ Guia Rápido de Git – Commits

## ■ Índice

1. Criar um commit
2. Alterar mensagem do último commit (sem push)
3. Alterar mensagem do último commit (já com push)
4. Alterar mensagem do penúltimo commit
5. Alterar mensagem de commits mais antigos
6. Cancelar último commit (mas manter alterações)
7. Cancelar último commit (e também remover alterações)
8. Ver histórico de commits
9. Desfazer alterações de um arquivo específico (não commitado)
10. Criar commit vazio (só mensagem, sem alterações)
11. Boas práticas
12. Ver diferenças antes de commitar
13. Corrigir último commit (adicionando arquivos esquecidos)
14. Reverter um commit específico
15. Mudar commits de ordem ou juntar commits
16. Remover último commit mas guardar alterações
17. Guardar alterações temporariamente (stash)
18. Commit parcial (só parte de um arquivo)

---

## ■ 1. Criar um commit

```
git add arquivo.txt
```

```
git add .
```

```
git commit -m "Mensagem clara e objetiva"
```

*Sempre escreva mensagens que expliquem **\*\*o que foi feito\*\***.*

## ■ 2. Alterar mensagem do último commit (sem push)

```
git commit --amend -m "Nova mensagem"
```

### ■ 3. *Alterar mensagem do último commit (já com push)*

```
git commit --amend -m "Nova mensagem"
```

```
git push origin minha-branch --force
```

### ■ 4. *Alterar mensagem do penúltimo commit*

```
git rebase -i HEAD~2
```

### ■ 5. *Alterar mensagem de commits mais antigos*

```
git rebase -i HEAD~5
```

### ■ 6. *Cancelar último commit (mas manter alterações)*

```
git reset --soft HEAD~1
```

### ■ 7. *Cancelar último commit (e também remover alterações)*

```
git reset --hard HEAD~1
```

### ■ 8. *Ver histórico de commits*

```
git log --oneline
```

### ■ 9. *Desfazer alterações de um arquivo específico (não commitado)*

```
git checkout -- arquivo.txt
```

### ■ 10. *Criar commit vazio (só mensagem, sem alterações)*

```
git commit --allow-empty -m "Marca versão X"
```

### ■ 11. *Boas práticas*

- Mensagens curtas e objetivas (até 50 caracteres).
- Use imperativo: "Adiciona", "Corrige", "Remove".
- Se precisar, detalhe no corpo da mensagem.

### ■ 12. *Ver diferenças antes de commitar*

```
git diff
```

git diff --staged

### ■ 13. *Corrigir último commit (adicionando arquivos esquecidos)*

git add arquivo\_esquecido.txt

git commit --amend --no-edit

### ■ 14. *Reverter um commit específico*

git revert abc123

### ■ 15. *Mudar commits de ordem ou juntar commits*

git rebase -i HEAD~5

- pick → mantém
- reword → altera mensagem
- squash → junta commits
- drop → remove commit

### ■ 16. *Remover último commit mas guardar alterações*

git reset --mixed HEAD~1

### ■ 17. *Guardar alterações temporariamente (stash)*

git stash

git checkout outra-branch

git stash pop

### ■ 18. *Commit parcial (só parte de um arquivo)*

git add -p