

Guia Rápido de Comandos Git

Projeto: Ionic Currency Converter Gerente de Configuração: Davi (você)

1. Introdução

Este documento é o tutorial oficial para os comandos Git que usaremos no projeto. Ele segue o fluxo de trabalho definido no `README.md`. O objetivo é ter um guia rápido para consulta.

2. Configuração Inicial (Fazer 1 vez)

Para baixar o projeto pela primeira vez.

```
git clone <URL_DO_REPOSITÓRIO>
```

- **O que faz:** Clona (baixa) o repositório do GitHub para o seu computador.

3. Fluxo de Trabalho Diário (O Loop Principal)

Este é o processo que você deve seguir toda vez que for começar uma nova tarefa.

Passo 1: Sincronizar com a `main`

(Sempre faça isso antes de criar uma branch nova)

```
git checkout main
```

- **O que faz:** Muda seu "ponteiro" para a branch principal.

```
git pull origin main
```

- **O que faz:** Baixa todas as atualizações mais recentes da branch `main` que estão no GitHub.

Passo 2: Criar sua Branch de Tarefa

```
git checkout -b prefixo/nome-da-tarefa
```

- **O que faz:** Cria uma nova branch *a partir da main* e já muda para ela.
- **Prefixos:**
 - `dev/` (Desenvolvimento, ex: `dev/criar-tela-login`)
 - `doc/` (Documentação, ex: `doc/escrever-casos-de-uso`)

- `sm/` (Scrum Master, ex: `sm/atualizar-trello`)
- `config/` (Configuração, ex: `config/ajustar-readme`)
- `fix/` (Correção de Bug, ex: `fix/corrigir-botao`)

Passo 3: Trabalhar e Salvar (Commits)

(Faça isso várias vezes durante o dia, enquanto trabalha)

```
git add .
```

- **O que faz:** "Prepara" todos os arquivos que você modificou para serem salvos.

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

- **O que faz:** Salva um "snapshot" (uma foto) do seu trabalho com uma mensagem.
- **Exemplo:** `git commit -m "docs: adiciona tutorial de comandos git"`

Passo 4: Enviar para o GitHub

(Faça isso quando sua tarefa estiver concluída)

```
git push origin prefixo/nome-da-tarefa
```

- **O que faz:** Envia a sua branch (e todos os seus commits) para o GitHub.
- Depois disso, você deve ir ao site do GitHub e "Abrir um Pull Request".

4. Comandos Úteis (Para Verificar)

```
git status
```

- **O que faz:** Mostra quais arquivos você modificou e em qual branch você está. (O comando mais útil!)

```
git branch
```

- **O que faz:** Lista todas as branches que existem no seu computador.

```
git pull
```

- **O que faz:** Baixa atualizações para a branch que você está no momento.