

Aqui está a tabela em Markdown contendo uma lista abrangente de comandos Git, organizados de acordo com o fluxo de trabalho típico de desenvolvimento de software:

Comando Git	Descrição	Exemplo prático
<code>git clone &lt;url&gt;</code>	Clona um repositório Git para o diretório local.	<code>git clone https://github.com/usuario/repositorio. git</code>
<code>git add &lt;arquivo&gt;</code>	Adiciona alterações no arquivo ao índice (staging area) para preparar para o commit.	<code>git add arquivo.txt</code>
<code>git status</code>	Exibe o estado do diretório de trabalho e do índice.	<code>git status</code>
<code>git commit -m "mensagem"</code>	Grava as alterações confirmadas no repositório.	<code>git commit -m "Adicionando funcionalidade de login"</code>
<code>git pull</code>	Atualiza o repositório local com as alterações do repositório remoto.	<code>git pull origin master</code>
<code>git push</code>	Envia os commits locais para o repositório remoto.	<code>git push origin master</code>
<code>git branch</code>	Lista todas as branches no repositório.	<code>git branch</code>
<code>git branch &lt;nome&gt;</code>	Cria uma nova branch com o nome especificado.	<code>git branch feature-novo-componente</code>
<code>git checkout &lt;branch&gt;</code>	Muda para a branch especificada.	<code>git checkout feature-novo-componente</code>
<code>git merge &lt;branch&gt;</code>	Combina a branch especificada com a branch atual.	<code>git merge feature-novo-componente</code>

<code>git diff &lt;arquivo&gt;</code>	Mostra as diferenças entre o arquivo no diretório de trabalho e a versão no índice (staging area).	<code>git diff arquivo.txt</code>
<code>git log</code>	Exibe o histórico de commits.	<code>git log</code>
<code>git remote -v</code>	Lista os repositórios remotos configurados.	<code>git remote -v</code>
<code>git fetch</code>	Busca todas as branches do repositório remoto.	<code>git fetch origin</code>
<code>git reset &lt;arquivo&gt;</code>	Remove o arquivo do índice (staging area), mas preserva as alterações no diretório de trabalho.	<code>git reset arquivo.txt</code>
<code>git rm &lt;arquivo&gt;</code>	Remove o arquivo do diretório de trabalho e do índice (staging area).	<code>git rm arquivo.txt</code>
<code>git stash</code>	Armazena temporariamente as mudanças não commitadas para uso posterior.	<code>git stash</code>
<code>git cherry-pick &lt;commit&gt;</code>	Aplica as alterações introduzidas por um commit específico.	<code>git cherry-pick abc123</code>
<code>git rebase &lt;branch&gt;</code>	Reorganiza o histórico de commits para que a branch atual se ramifique da branch especificada.	<code>git rebase master</code>
<code>git bisect</code>	Ajuda a encontrar o commit que introduziu um bug, usando uma busca binária entre commits conhecidos.	<code>git bisect start</code>

Esses comandos abrangem uma variedade de tarefas comuns no dia a dia de um desenvolvedor e devem fornecer uma base sólida para começar a trabalhar com Git.

Lembre-se de consultar a documentação oficial do Git ou usar `git --help` para obter mais informações sobre cada comando.