

<code>`git add .`</code>	Adiciona todos os arquivos modificados para serem rastreados pelo Git.
<code>`git commit -m "&lt;mensagem&gt;"`</code>	Cria um novo commit com as mudanças adicionadas, junto com uma mensagem descritiva.
<code>`git status`</code>	Exibe o estado atual do repositório, mostrando arquivos modificados, adicionados e não rastreados.
<code>`git push`</code>	Envia as mudanças locais para um repositório remoto.
<code>`git pull`</code>	Atualiza o repositório local com as últimas alterações do repositório remoto.
<code>`git branch`</code>	Lista todas as branches (ramificações) no repositório.
<code>`git checkout &lt;branch&gt;`</code>	Altera a branch atual para a especificada.
<code>`git merge &lt;branch&gt;`</code>	Combina a branch especificada com a branch atual.
<code>`git log`</code>	Mostra um histórico dos commits realizados no repositório.
<code>`git reset &lt;commit&gt;`</code>	Desfaz os commits posteriores ao commit especificado, mantendo as alterações nos arquivos.
<code>`git revert &lt;commit&gt;`</code>	Desfaz um commit específico criando um novo commit que reverte as alterações.
<code>`git stash`</code>	Salva as alterações atuais em uma pilha temporária para que você possa alternar para outra branch.
<code>`.gitignore`</code>	Arquivo no qual você pode especificar arquivos e diretórios que devem ser ignorados pelo Git.
<code>`git remote add &lt;nome&gt; &lt;URL&gt;`</code>	Adiciona um repositório remoto com um nome especificado.

Lembre-se de que esses são apenas alguns dos comandos e recursos essenciais do Git. Existem muitos outros recursos e opções disponíveis para gerenciar um repositório Git. É recomendado consultar a documentação oficial do Git ou tutoriais mais detalhados para aprender mais sobre esses recursos e como utilizá-los adequadamente.



Send a message

