

O LIVRO MAIS SIMPLES DO MUNDO SOBRE

A BARRA MÁGICA

101 COMANDOS DO CLAUDE CODE

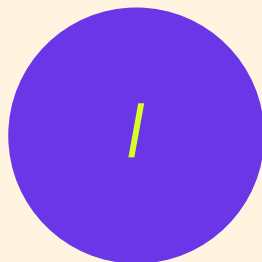


THYFANI XAVIER



DIGITE / E ABRA A CAIXA DE FERRAMENTAS.

Sem decorar. Sem medo.





ANTES DE COMEÇAR

Comandos de barra são atalhos digitados no começo da mensagem. Digite apenas / para ver o menu ou continue escrevendo para filtrar.

Tudo que vem depois do nome do comando é um argumento. Em `/plan corrigir login`, por exemplo, “corrigir login” explica o que será planejado.

Nem todo comando aparece para todo mundo. A disponibilidade muda conforme versão, sistema operacional, plano e ambiente.

Esta edição foi conferida em 5 de julho de 2026, usando a documentação oficial do Claude Code. Comandos podem mudar nas próximas versões.

A REGRA MAIS IMPORTANTE

O comando só funciona quando aparece no início da mensagem. Se estiver no meio de uma frase, vira apenas texto comum.



1. PRIMEIROS PASSOS

`/help`

Mostra a ajuda e os comandos disponíveis na sua instalação.

`/init`

Cria um CLAUDE.md inicial para ensinar ao Claude como o projeto funciona.

`/init`

`/status`

Mostra versão, modelo, conta e conectividade mesmo durante uma resposta.

`/login`

Entra na sua conta Anthropic.

`/logout`

Sai da sua conta Anthropic.

`/config [key=value ...]`

Abre ou altera configurações como tema, modelo, estilo de saída e pensamento.

`/config theme=dark model=sonnet`

`/theme`

Troca o tema visual, incluindo opções claras, escuras e acessíveis para daltonismo.

`/color [color|default]`

Troca a cor da barra de prompt para diferenciar sessões visualmente.

`/color pink`

`/terminal-setup`

Configura atalhos do terminal, como Shift+Enter para quebrar linha.

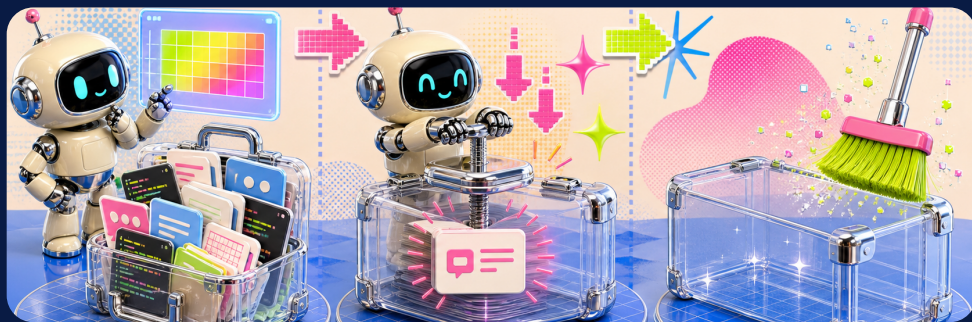
/keybindings

Abre o arquivo onde os atalhos de teclado são personalizados.

/powerup

Abre pequenas aulas interativas para descobrir recursos do Claude Code.

2. CONTEXTO E CONVERSA



```
/context [all]
```

Mostra o que está ocupando a janela de contexto e onde existe desperdício.

```
/context all
```

```
/compact [instructions]
```

Resume a conversa atual para liberar espaço na janela de contexto sem começar do zero.

```
/compact preserve decisões e próximos passos
```

```
/clear [name]
```

Começa uma conversa vazia. A sessão anterior continua guardada para ser retomada depois.

```
/clear tarefa-anterior
```

`/memory`

Edita memórias CLAUDE.md, controla a memória automática e mostra o que foi aprendido.

`/recap`

Gera um resumo de uma linha sobre o que aconteceu na sessão.

`/rename [name]`

Dá um nome à sessão para encontrá-la com facilidade depois.

```
/rename corrigir-login
```

`/resume [session]`

Retoma uma conversa anterior por nome ou ID; sem argumento, abre o seletor.

```
/resume corrigir-login
```

`/branch [name]`

Cria uma cópia da conversa neste ponto para você testar outro caminho sem perder o original.

`/btw <question>`

Faz uma pergunta lateral rápida sem colocar essa pergunta no histórico principal.

`/copy [N]`

Copia uma resposta do Claude ou um bloco de código para a área de transferência.

`/copy 2`

`/export [filename]`

Exporta a conversa atual como texto para copiar ou salvar em arquivo.

`/rewind`

Volta código e conversa a um checkpoint anterior ou resume uma parte do histórico.

3. MODELO E DESEMPENHO

```
/model [model]
```

Troca o modelo usado agora e pode definir o padrão das próximas sessões.

```
/model sonnet
```

```
/effort [level|auto]
```

Ajusta quanto esforço de raciocínio o modelo deve gastar antes de responder.

```
/effort high
```

```
/fast [on|off]
```

Liga ou desliga o modo rápido quando ele estiver disponível.

```
/advisor [model|off]
```

Liga um segundo modelo para aconselhar o Claude nas decisões mais difíceis.

`/focus`

Mostra uma interface mais limpa, escondendo detalhes intermediários e destacando o resultado.

`/tui [default|fullscreen]`

Alterna entre a interface padrão e o modo fullscreen.

`/scroll-speed`

Ajusta a velocidade da roda do mouse na interface fullscreen.

`/statusline`

Cria ou ajusta a linha de status com contexto, custos e informações do Git.

`/usage`

Mostra consumo, limites do plano e quais recursos estão gastando mais.

`/cost`

Atalho de `/usage` para consultar consumo, limites e atividade.

`/stats`

Atalho de `/usage` que abre diretamente a aba de estatísticas.

`/usage-credits`

Configura créditos extras para continuar depois de atingir o limite.



4. PASTAS, EDITOR E APLICATIVO

```
/add-dir <path>
```

Libera outra pasta para o Claude ler e usar nesta sessão, sem mudar a pasta principal.

```
/add-dir ../documentacao
```

```
/cd <path>
```

Move a sessão para outra pasta, preservando a conversa e o cache já construído.

```
/cd ../outro-projeto
```

```
/ide
```

Gerencia a integração com o editor ou IDE e mostra seu estado.

```
/desktop
```

Continua a sessão atual no aplicativo Claude Code Desktop.

`/chrome`

Abre as configurações da integração do Claude com o navegador Chrome.

`/mobile`

Exibe um QR code para baixar o aplicativo móvel do Claude.

`/voice [hold|tap|off]`

Liga o ditado por voz no modo segurar, tocar ou desliga o recurso.

```
/voice hold
```

`/remote-control`

Disponibiliza a sessão local para continuar no navegador, celular ou outro dispositivo.

`/teleport`

Traz para o terminal uma sessão iniciada no Claude Code para web.

/remote-env

Escolhe o ambiente padrão usado pelos agentes que rodam na nuvem.

5. PLANEJAR E TRABALHAR EM PARALELO



`/plan [description]`

Entra no modo de planejamento antes de uma mudança grande, sem sair editando imediatamente.

`/plan reorganize a tela de login`

`/agents`

Mostra como criar ou administrar subagentes especializados do projeto.

`/fork <directive>`

Entrega uma tarefa lateral a um subagente que herda a conversa e trabalha em paralelo.

`/fork investigue por que o teste está lento`

`/background [prompt]`

Solta a sessão inteira para continuar trabalhando em segundo plano e devolve seu terminal.

`/batch <instruction>`

Divide uma mudança enorme em várias partes e entrega cada parte a um agente em worktree própria.

`/batch migrar src/ para React`

`/tasks`

Mostra e administra tudo que está rodando em segundo plano.

`/stop`

Interrompe uma sessão em segundo plano sem apagar o histórico ou a worktree.

`/goal [condition|clear]`

Define uma condição de conclusão e mantém Claude trabalhando até alcançá-la ou você cancelar.

`/goal todos os testes passam`

`/loop [interval] [prompt]`

Repete um prompt em intervalos ou deixa o Claude decidir quando verificar novamente.

`/loop 5m verifique se o deploy terminou`

`/workflows`

Abre o painel para acompanhar, pausar, retomar ou salvar workflows.

`/ultraplan <prompt>`

Cria um plano aprofundado na nuvem para revisar no navegador antes de executar.

6. REVISAR, TESTAR E ENTREGAR

```
/diff
```

Abre um visor das mudanças ainda não commitadas e das alterações feitas em cada turno.

```
/code-review [low|medium|high|xhigh|max|ultra]  
[--fix] [--comment] [target]
```

Revisa alterações procurando bugs e melhorias, pode corrigir ou comentar no GitHub.

```
/code-review high --fix
```

```
/review [PR]
```

Revisa um pull request do GitHub sem alterar o código.

```
/review 42
```

```
/security-review
```

Examina as mudanças procurando vulnerabilidades, vazamento de dados e falhas de autenticação.

`/simplify [target]`

Chama agentes para limpar o código alterado, reduzir duplicação e melhorar eficiência.

`/run`

Inicia o aplicativo e navega nele para comprovar que a mudança funciona de verdade.

`/verify`

Constrói e usa o aplicativo para verificar a mudança além de testes automáticos.

`/run-skill-generator`

Ensina `/run` e `/verify` a construir e testar este projeto específico.

`/autofix-pr [prompt]`

Cria uma sessão na nuvem que acompanha o PR e tenta corrigir falhas de CI e comentários.

/install-github-app

Instala o aplicativo do Claude no GitHub e pode configurar Actions.

/web-setup

Conecta o GitHub ao Claude Code para web usando as credenciais locais do gh.



7. PERMISSÕES E SEGURANÇA

`/permissions`

Gerencia o que Claude pode fazer sem perguntar, o que exige aprovação e o que é proibido.

`/fewer-permission-prompts`

Analisa o histórico e cria regras para reduzir pedidos repetitivos de permissão.

`/sandbox`

Liga ou desliga o modo isolado para execução mais segura, quando suportado.

`/hooks`

Mostra os hooks configurados para reagir a eventos como edições e comandos.

/privacy-settings

Mostra e altera preferências de privacidade quando disponíveis no seu plano.



8. SKILLS, PLUGINS E CONEXÕES

```
/skills
```

Lista as skills disponíveis e permite filtrar ou controlar sua visibilidade.

```
/reload-skills
```

Procura novamente skills e comandos que foram criados ou alterados no disco.

```
/plugin [subcommand]
```

Lista, instala, ativa ou desativa plugins do Claude Code.

```
/reload-plugins [--force]
```

Recarrega plugins ativos sem reiniciar; --force aceita mudanças que afetam ferramentas MCP.

```
/mcp [reconnect <server>|enable|disable  
[<server>|all]]
```

Administra conexões MCP, autenticação e o estado de cada servidor externo.

```
/claude-api [migrate|managed-agents-onboard]
```

Carrega a referência da API da Anthropic ou ajuda a migrar código para modelos mais novos.

```
/dataviz [request]
```

Ativa orientação especializada para criar gráficos, dashboards e visualizações acessíveis.

```
/deep-research <question>
```

Espalha uma pesquisa pela web entre agentes e devolve um relatório com fontes.

```
/design-login
```

Autoriza o acesso necessário para sincronizar um design system.

`/design-sync [hint]`

Envia os componentes React do projeto para o Claude Design reutilizá-los.

`/install-slack-app`

Abre o fluxo para instalar e autorizar o aplicativo do Claude no Slack.



9. NUVEM E EMPRESAS

```
/schedule [description]
```

Cria e administra rotinas que rodam na infraestrutura em nuvem da Anthropic.

```
/team-onboarding
```

Cria um guia de entrada para a equipe com base nos seus últimos 30 dias de uso.

```
/setup-bedrock
```

Configura autenticação, região e modelos quando Claude Code usa Amazon Bedrock.

```
/setup-vertex
```

Configura projeto, região e modelos quando Claude Code usa a plataforma do Google Cloud.



10. DIAGNÓSTICO E MANUTENÇÃO

/doctor

Faz um check-up da instalação e das configurações; pode tentar corrigir os problemas encontrados.

/debug [description]

Liga os registros de depuração e investiga o que está dando errado na sessão.

/heapdump

Gera um retrato técnico da memória para investigar consumo excessivo.

/insights

Analisa suas sessões e mostra padrões de uso, áreas do projeto e pontos de atrito.

`/feedback [report]`

Envia feedback ou um relatório de bug com o contexto da sessão.

`/release-notes`

Mostra as novidades e mudanças das versões do Claude Code.



11. CONTA, EXTRAS E COMANDOS ESPECIAIS

`/exit`

Sai do Claude Code; numa sessão em segundo plano, apenas desconecta sem parar o trabalho.

`/upgrade`

Abre a página para mudar para um plano superior.

`/passes`

Permite compartilhar uma semana gratuita quando sua conta for elegível.

`/radio`

Abre a rádio lo-fi Claude FM no navegador. É o comando recreativo da turma.

/stickers

Abre a página para pedir adesivos do Claude Code.



12. REMOVIDOS OU SUBSTITUÍDOS

`/pr-comments [PR]`

Comando removido. Hoje, peça diretamente ao Claude para ler os comentários do PR.

`/ultrareview [PR]`

Alias de revisão profunda multiagente; a forma preferida hoje é `/code-review ultra`.

`/vim`

Comando removido. O modo Vim agora é escolhido em `/config`, na opção de editor.



A COLA DE BOLSO

/init

ensinar o projeto

/plan

pensar antes de editar

/context

ver o que ocupa memória

/compact

resumir sem recomeçar

/clear

começar outra tarefa

/model

trocar o cérebro

/permissions

controlar acessos

/diff

ver as mudanças

/code-review

caçar bugs

/doctor

diagnosticar problemas

/resume

voltar a uma sessão

/help

descobrir o resto



FONTES E AVISO DE ATUALIZAÇÃO

Fonte principal: documentação oficial “Commands” do Claude Code.

Índice oficial consultado: <https://code.claude.com/docs/llms.txt>

Referência de comandos: <https://code.claude.com/docs/en/commands>

A lista foi conferida em 5 de julho de 2026. Recursos podem depender da versão, plataforma, plano e ambiente.

Este é um guia editorial independente, escrito para iniciantes. Para decisões técnicas ou de segurança, confirme sempre a documentação oficial mais recente.

THYFANI XAVIER

Coleção: O Livro Mais Simples do Mundo sobre Inteligência Artificial



A BARRA ESTÁ ABERTA.

Agora é só escolher
a ferramenta certa.

DIGITE /