cauce

Guía gratis · Claude Code

Seis líneas de llegada: dejá de escribir seguí.

Con /goal, Claude sigue solo hasta cumplir una condición. Otro modelo la evalúa después de cada vuelta, pero no corre comandos: juzga lo que Claude mostró. Por eso la condición tiene que decir cómo se comprueba.

01

Tests en verde.

`código`

Con lo que no se puede tocar adentro.

```
/goal todos los tests de test/auth
pasan y el lint da limpio, sin
modificar ningún test, o pará a
las 20 vueltas
```

02

Que compile.

`build`

Un código de salida no deja lugar a dudas.

```
/goal npm run build termina con
código 0, mostrando la salida, o
pará a las 20 vueltas
```

03

Según el pedido.

`SPEC.md`

Ideal después de una entrevista.

```
/goal cada requisito de SPEC.md
está hecho y mostraste cómo probaste
cada uno, o pará a las 30 vueltas
```

04

Partir lo gigante.

`orden`

Con un número y un comando que lo confirma.

```
/goal ningún archivo de src/ pasa
las 300 líneas, wc -l lo muestra y
npm test termina en 0, o pará a
las 25 vueltas
```

05

Vaciar la lista.

`pendientes`

Una cola vacía es fácil de comprobar.

```
/goal no queda ningún TODO en src/,
grep lo confirma y npm test termina
en 0, o pará a las 25 vueltas
```

06

Completar un registro.

`documentación`

Para trabajo repetitivo que tiene un final claro.

```
/goal CHANGELOG.md tiene una entrada
por cada PR mergeado esta semana, o
pará a las 15 vueltas
```

+

Linda no se mide.

Nada de que quede linda: poné algo comprobable y un límite. Para que corra sin pedirte permisos, usá modo auto. /goal solo te muestra cómo va; /goal clear lo frena.

Seguime para más trucos **@felipetizonmedia**

Fuente: code.claude.com/docs/en/goal (consultado 2026-11-09).