

# Claude Code para Líderes de Producto

Setup, comandos y conceptos esenciales para empezar a usar Claude Code en serio.



producthub.com.ar

## S01 · SETUP Instalación por sistema operativo

macOS

RECOMENDADO

```
# Native binary
curl -fsSL https://claude.ai/install.sh | bash

# Alternativa
brew install --cask claude-code
```

Linux / WSL

CROSS-PLATFORM

```
# Native binary
curl -fsSL https://claude.ai/install.sh | bash

# Vía npm (cualquier OS)
npm install -g @anthropic-ai/claude-code
```

Windows

VÍA WSL

```
# Habilitar WSL primero
wsl --install

# Luego, dentro de WSL:
curl -fsSL https://claude.ai/install.sh | bash
```

### PRE-REQUISITO · PLAN PRO USD 20/MES ALCANZA PARA USO INTENSIVO DE LÍDER DE PRODUCTO

```
# Verificar instalación + diagnóstico completo
claude --version # versión instalada
claude doctor # diagnóstico del setup
claude auth login # conectá tu cuenta Pro/Max
cd tu-proyecto && claude # arrancar sesión en el proyecto
```

## S02 · FUNDAMENTOS Los 5 conceptos que cambian todo

CLAUDE.md

./CLAUDE.md

El "cerebro" del proyecto. Markdown en la raíz que Claude lee al arrancar cada sesión. Contiene contexto durable: stack, convenciones, comandos, reglas.

CUÁNDO CREARLO

Primer comando en cualquier proyecto nuevo — `/init` lo genera.

Skills

~/.claude/skills/

Paquetes reusables de instrucciones, prompts y workflows que Claude invoca automáticamente cuando son relevantes. Encapsulá un proceso para que cualquier sesión lo use sin reexplicarlo.

CUÁNDO USARLO

Cuando notás que repetís el mismo prompt o workflow.

MCPs

Model Context Protocol

Protocolo estándar para conectar Claude a servicios externos: Supabase, GitHub, Slack, Notion, bases de datos. Cada MCP server expone herramientas que Claude usa como nativas.

CUÁNDO CONVIENE

Para que Claude lea o escriba en un servicio externo. Ver activos: `/mcp`

Sub-agentes

Task tool

Agentes especializados que Claude invoca para tareas paralelas o multi-perspectiva. Un agente principal coordina, los sub-agentes ejecutan en paralelo.

CUÁNDO CONVIENE

Tareas grandes paralelizables o contextos limpios y separados.

Agent Teams

orquestración a escala

Múltiples agentes colaborando con roles definidos: planner, executor, reviewer. Cada uno con su CLAUDE.md y permisos.

CUÁNDO CONVIENE

Proyectos con etapas claras: research → diseño → ejecución → review.

## S03 · MODELOS Opus, Sonnet, Haiku — cuándo usar cuál

Cambiar de modelo es la palanca #1 para controlar costo y velocidad sin perder calidad. Plan Pro incluye los tres con límites compartidos.

MODELO	CUÁNDO USAR	COMANDO
Opus 4.7	Razonamiento complejo, arquitectura, refactors grandes, decisiones de producto importantes.	<code>/model opus</code>
Sonnet 4.6	Caballo de batalla diario. Análisis, research, escritura, código común.	<code>/model sonnet</code>
Haiku 4.5	Tareas simples y rápidas. Resúmenes cortos, respuestas directas, formato.	<code>/model haiku</code>

EL TRUCO QUE AHORRA TOKENS

`/model opusplan`

Opus piensa y planifica, luego cambia automáticamente a Sonnet para ejecutar. Razonamiento de Opus al precio de Sonnet en la fase pesada.

PARA REFACTORS O TAREAS PLANIFICACIÓN + EJECUCIÓN

## S04 · COMANDOS Los 20+ comandos que vas a usar todos los días

A Setup y contexto

- `/init` Crea el CLAUDE.md inicial. Primer comando en cualquier repo.
- `/help` Lista todos los comandos disponibles, plugins y skills.
- `/config` Configuración global: notificaciones, permisos, defaults.
- `@archivo` Incluí archivos sin pegar texto. `@README.md`, `@src/`.

B Contexto y costo

- `/compact` ★ El más importante. Comprime la conversación. Cada 20-30 min.
- `/clear` Limpia el contexto al cambiar de tarea. Más agresivo que compact.
- `/cost` Cuántos tokens consumiste en la sesión actual.
- `/usage` Estado de tus límites del plan (Pro/Max).

C Modelos y razonamiento

- `/model opus` Switch a Opus 4.7 — razonamiento pesado.
- `/model sonnet` Switch a Sonnet 4.6 — caballo de batalla.
- `/model opusplan` ★ Opus planifica + Sonnet ejecuta. La jugada de eficiencia.
- `/effort xhigh` Máximo esfuerzo de razonamiento (Opus 4.7).

D Movilidad multi-device

- `/desktop` Pasa la sesión actual al desktop app de Claude.
- `/remote-control` Activa Remote Control. Continúa desde browser, mobile o iPad.
- `/rc` Alias corto de `/remote-control`.
- `/mobile` Muestra QR para descargar la app móvil.

## S05 · ECOSISTEMA Extensiones y movilidad

MCPs útiles para PMs

- `/mcp` Listar MCPs activos.
- Supabase Base de datos para landings.
- GitHub Issues, PRs, repos con eng.
- Slack / Notion Canales, docs, knowledge base.
- Linear / Jira Tickets, sprints, roadmap.

Skills bundled más usados

- pdf, docx Documentos sin salir de Claude.
- xlsx, pptx Spreadsheets y presentaciones.
- data-analysis CSVs, datasets, exports.
- frontend-design Mockups, componentes web.
- file-reading Archivos uploads grandes.

Tip de movilidad

Empezás en la laptop, la dejás corriendo, salís — y desde el celular seguís aprobando con la misma sesión. La compu tiene que estar prendida, pero tu setup local (MCPs, archivos, Skills, CLAUDE.md) sigue disponible en remoto via `/rc`.