Claude Code

Het spiekbriefje van je bouwdag, naast je toetsenbord

30 NOVEMBER 2026

OPEN DIT IN EEN TABBLAD OF PRINT HET · DE EERSTE WEKEN SCHEELT HET ZOEKEN

1 BEGINNEN

1. Open je **terminal**, maak een map voor je project (`mkdir bouwdag`) en ga erheen (`cd bouwdag`). Start nooit in je thuismap.
2. Typ `claude` en druk op enter.
3. Run één keer `/init` - Claude leest je project en maakt een instructiebestand aan. Noem het `AGENTS.md`, dan lezen andere AI-tools het ook.
4. Run één keer `/statusline` - daarna zie je onderin altijd hoe vol je context zit.
5. Stel gewoon je vraag in normaal Nederlands. Bouwen, uitleggen, fixen - alles mag.

> **AGENTS.md** = *het geheugen van je project. Zet hier afspraken, stijl en spelregels in; Claude leest dit elke sessie automatisch.*

3 TOETSEN UIT JE HOOFD

`Shift` + `Tab`	Wissel permissie-modus (zie blok 5)
`Esc`	Onderbreek Claude midden in een antwoord
`Esc` `Esc`	Open het terugspoel-menu
`@`	Verwijs naar een bestand of map
`↑` `↓`	Door je eerdere prompts bladeren
`Ctrl` + `R`	Zoek terug in je prompt-historie
`Ctrl` + `V`	Plak een afbeelding of screenshot (Mac: Ctrl, niet Cmd. Windows: Alt+V)
`Ctrl` + `C` `Ctrl` + `C`	Twee keer = Claude Code afsluiten
`?`	Alle sneltoetsen tonen

4 MCP & CLI - JE NIEUWE COLLEGA'S

MCP — *Een expertcollega die je installeert* - kent de documentatie, bedient de API van een dienst. Bijv. de MCP van ActiveCampaign voor MailBlue.

CLI — *Command line interface* - tools die met tekst-commando's werken; daar kan Claude razendsnel mee praten. Bijv. de GitHub CLI (`gh`) en de Cloudflare CLI (`cf`).

2 DE COMMANDO'S DIE JE ECHT GEBRUIKT

`/clear`	Schoon beginnen. *Nieuw onderwerp = even clearen.*
`/compact`	Gesprek inkorten als 't lang wordt, zonder de draad kwijt te raken.
`/usage`	Wat je verbruikt en hoeveel van je abonnement over is.
`/rewind`	Terug naar een eerder punt. *Stap verkeerd? Spoel terug.*
`/resume`	Een eerder gesprek weer oppakken.

5 DE 4 STANDEN (SHIFT+TAB)

- **Manual** - vraagt netjes toestemming voor elke actie. Veilig startpunt.
- **Accept Edits** - keurt wijzigingen automatisch goed. Gebruik dit zodra je 'm vertrouwt.
- **Plan** - leesmodus. Denkt mee, raakt niks aan. Ideaal vóór een grote stap.
- **Auto** - draait door zonder steeds te vragen. In nieuwe versies de startstand.

6 ZO HAAL JE ER HET MEESTE UIT

1. Geef **context**, niet alleen een opdracht: wat, waarvoor, voor wie.
2. Zet bij grote dingen eerst **planmodus** aan (`Shift` + `Tab`) - plan lezen, dan akkoord.
3. Werk in **kleine stappen**. Klaar met een onderdeel? `/clear` en door.
4. Twijfel je over een wijziging? `Esc Esc` en spoel terug. Niks gaat stuk.
5. Slop-alarm: vraag om **karakter**, geef voorbeelden. "Maak het niet generiek."

Verder dan de basis

7 SKILL OF AGENT?

	SKILL	AGENT
WAT	Kennis + instructie in een bestand (`SKILL.md`): wat AI ***weet***.	Een handelende entiteit die uitvoert, beslist, itereert: wat AI ***doet***.
GEDRAG	Passief - wordt ingeladen wanneer relevant.	Actief - keten van stappen: input → logica → output, met tool-toegang.
BIJV.	Je huisstijl, review-checklist, offerte-conventies.	Een hulpje dat een deeltaak apart uitzoekt terwijl jij doorgaat.

Vuistregel: kennis die je wilt **hergebruiken** *→ skill. Een klus die* **zelfstandig uitgevoerd** *moet worden → agent. Ze combineren prima: een agent laadt skills in.*

8 DE STACK OM ZELF TE BOUWEN

`Nuxt` ***Framework*** - site, app en API in één project. Claude kent het door en door.

`Tailwind` ***Styling*** - vormgeving als utility-classes. AI schrijft dit vloeiend en consistent.

`Cloudflare` ***Hosting*** - Pages & Workers, gratis om te starten. Met de ***Cloudflare CLI*** (`cf`) zet Claude je site in seconden live.

`GitHub` ***Versiebeheer*** - elke stap vastgelegd; Claude bedient 'm via de ***GitHub CLI*** (`gh`). Thuisbasis van je skills (blok 9).

Tip: *maak een* `design.md` *met je huisstijl en assets. Deel 'm net als je skills (blok 9) - alles wat je bouwt is meteen on-brand.*

9 SKILLS DELEN IN JE ORGANISATIE

1. Zet jullie skills in **één git-repo**: `skills/<naam>/SKILL.md`.
2. Iedereen clonet 'm, bijv. naar `~/git/skills`.
3. Koppel met een **symlink** vanuit de git-map: `ln -s ~/git/skills/<naam> ~/.claude/skills/` - zo werkt de skill in élk project en blijft de repo de bron.
4. Collega maakt iets nieuws? `git pull` en jij hebt 'm ook.

10 SUPERKRACHTEN

`/skills` ***Skills*** = kant-en-klare recepten. Eén commando, hele klus klaar (bijv. een GEO/SEO-check op je site).

`/skill-creator` ***Claude helpt je een skill maken.*** Eenmalig installeren: `/plugin install skill-creator@claude-plugins-official`

`/agents` ***Subagents*** - vraag Claude om een hulpje dat een deeltaak apart uitzoekt terwijl jij doorgaat.

`/mcp` Koppel externe bronnen (Stripe, GitHub, je analytics) zodat Claude er direct bij kan.

`/memory` Bewerk wat Claude over je project onthoudt.

`/model` Wissel van model (bijv. een zwaarder model voor lastige klussen).

`/voice` ***Praten i.p.v. typen*** - gedachtes hardop delen gaat sneller dan tikken. Ideaal voor brainstorms en context geven.

11 WELK MODEL WANNEER?

`Opus 5.5` ***De standaard*** - sterk op alledaags én complex werk. Hier start je.

`Sonnet 5.5` ***Het werkpaard*** - routinetaken, snel en nog steeds erg krachtig. Wisselen: `/model`.

12 HERDR: EEN PAAR AGENTS TEGELIJK

1. Open **Herdr** en start per klus een eigen agent, **elk in een eigen map** (site, teksten, event).
2. Herdr laat zien wie **werkt**, wie **op jou wacht** en wie **klaar** is. Begin bij wie wacht.
3. Venster dicht? De agents **werken door** op de achtergrond. Open Herdr en je ziet waar ze zijn.

Begin met **twee** *agents, niet vijf. Jij leest mee en keurt goed: meer agents is meer lezen.*

13 BOUW EEN SECOND BRAIN

1. **Transcribeer alles** - meetings, telefoontjes, losse gedachtes. Dicteren: Paraspeech (Mac), FluidVoice (gratis, open source) of Handy (Windows, gratis). Als team: **Wispr Flow** of **Fireflies** (betaald, cloud).
2. Bewaar de transcripten als tekstbestanden in een map in je project.
3. **Voed ze aan je agents** - elke meeting wordt context om mee te bouwen, elke gedachte doorzoekbaar.

Opname → transcript → context → bouwen. Je gesprekken worden **brandstof** *in plaats van herinnering.*