

// SECTION 01 • SKILL PACK

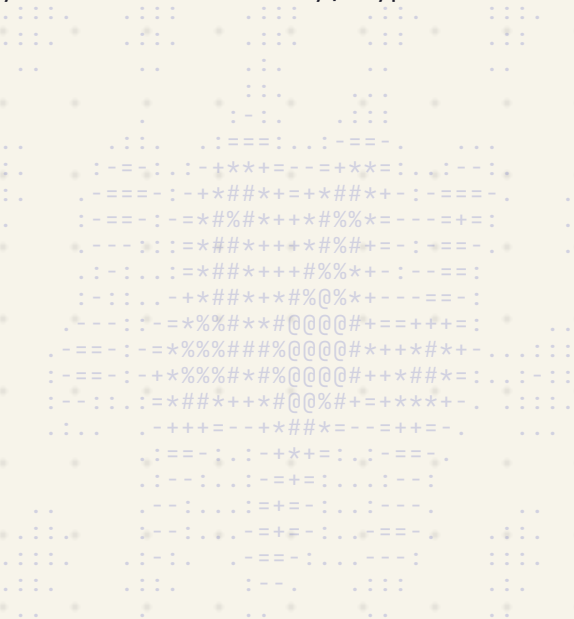
ILLEGAL CODES



Five Claude Code skills that feel illegal to know: Mandarin-compressed agent runs, an anti-slop design mode, a six-agent tournament, a self-attacking fact checker, and a startup analyst with Euro Summer energy. Free on GitHub.

```
> boot opusjake_os
> resource: illegal-codes
> version: v1.0 • 2026
> status: ready_
```

Five codes, five operating modes. The names sound like a group chat. Underneath each one is a workflow that saves real money, real hours, or a bad month. All five are installed the same way: clone the repo, drop a folder in your skills directory, type the slash command.



CHINA MAXING

Type `/china-maxing` and every prompt Claude sends to a sub-agent ships in Mandarin. Chinese carries more meaning per character, so the same brief costs fewer tokens on most tokenizers. Prose briefs compress 30 to 50 percent. Code, file paths, and identifiers stay in English, and every result comes back in English. Your instructions do not change. The bill does.

```
> /china-maxing
> mode: ON for this session
> dispatching agent 3 of 6...
> brief: 1,240 tokens (EN) → 690 tokens (ZH)
> savings: 44 percent • results return in English
```

Use it when you are fanning out sub-agents with long briefs and the token bill keeps climbing.

AURA FARMING

Type `/aura-farming` before any visual: a Meta ad, a slide deck, a landing page, a social graphic. Claude locks a real design direction first (one aesthetic point of view, two typefaces max, one committed palette), bans every AI-slop tell (purple gradients, glassmorphism, emoji bullets, six identical rounded cards), then audits the draft against a checklist and fixes every failure before it ships.

```
> /aura-farming
> direction: brutalist mono on bone white
> banned: gradients, glassmorphism, emoji bullets,
>         six identical cards, corporate memphis
> audit: 2 slop flags found → fixed → clean
> verdict: would a designer post this? yes_
```

Use it when the visual has to look like a designer made it. People smell AI slop in half a second.

67

Type `/67` and Claude spawns six agents that each solve your task a different way: minimalist, maximalist, first principles, prior art, adversarial, wildcard. A seventh agent judges all six, scores them 1 to 10, names the strongest element of each, and builds the final answer by stealing the best parts. You get a leaderboard and the strongest version that exists, not the first one that came out.

```
> /67
> agents 1-6: minimalist · maximalist · first principles
>           prior art · adversarial · wildcard
> agent 7: judging all six...
> ranked: #4 wins at 8.7 · best parts stolen from #1 and #5
> final: synthesized_
```

Use it when the deliverable is high-stakes and one attempt would leave quality on the table. It costs six runs. Spend them where quality pays.

CRASH OUT

Type `/crash-out` and Claude drafts its answer, then becomes obsessed with destroying it. Every load-bearing claim gets extracted and attacked with evidence: run the code, read the source, search for the counter-case. Claims that survive ship with their evidence cited. Claims it could not verify get flagged, never asserted. Claims that break get corrected in the open, plus a kill list of what died on the way.

```
> /crash-out
> claims extracted: 11
> attacking with: code runs, source reads, counter-search
> survived: 8 · shaky (flagged): 2 · busted: 1
> busted claim corrected in the open
> shipping survivors only_
```

Use it when a confident wrong answer costs more than a slow right one. Research, debugging, anything with numbers in it.

LARP

Type `/larp` and paste your startup idea. Claude becomes a 24-year-old founder posting from Positano who raised \$100M for a ChatGPT wrapper, and under the bit runs a real teardown: the wedge, the honest TAM math, the moat or the admission there is none, the pricing, the channel that gets the first 100 customers, and the one risk that actually kills it. Then it assembles the strongest business model the idea can support and stamps a verdict: LARP, FUNDABLE LARP, or ACTUAL COMPANY.

```
> /larp
> vibe check: "this is actually insane"
> tam larp: $80B · tam real: $140M, math shown
> moat: none yet · shortest path: workflow lock-in
> kill shot: distribution, not product
> verdict: FUNDABLE LARP_
```

Use it when you have an idea and want the truth before you spend a month on it.

INSTALL

All five are open source on GitHub: github.com/Jakeschincariol/claude-illegal-codes

```
git clone https://github.com/Jakeschincariol/claude-illegal-codes.git
cd claude-illegal-codes
cp -r china-maxing aura-farming 67 crash-out larp ~/.claude/skills/
```

Type any code as a slash command inside Claude Code. That is the whole install.

// NO SETUP VERSION

No terminal handy? Open any SKILL.md in the repo and paste its body at the top of a chat as a mode. You get most of the effect with zero setup.

RUN THE STACK

Five names, one idea: a label you will remember wrapped around a mode that earns its keep.

WHEN YOU ARE...	REACH FOR
Fanning out expensive agent runs	/china-maxing
Making an ad, a deck, or a page	/aura-farming
Shipping something high-stakes	/67
Checking facts that matter	/crash-out
Sizing up a startup idea	/larp

The name is the joke. The mode is real. Want a hundred more codes? [The Code Book](#) has them. And the next pack ships first to the newsletter at [opusjake.ai](#).