TRAP SILICON

CLIENT TERMINAL GUIDE

YOUR COMPLETE WALKTHROUGH TO OPENING, NAVIGATING, AND MASTERING THE TERMINAL — FROM ZERO TO BUILDING WITH AI IN YOUR OWN MACHINE.

WHAT IS THE TERMINAL?

The Terminal is a built-in app on your Mac that lets you control your computer by typing text commands instead of clicking around with your mouse. You type a command, hit Enter, and the computer does exactly what you told it to do.

Think of it like this: the regular screen you see every day (your Desktop, Finder, app icons) is the front door of a house. The Terminal is the control room behind the walls. Everything your computer does — opening apps, saving files, connecting to the internet — is actually a command running behind the scenes. The Terminal lets you tap into that power directly.

Every professional developer and engineer works in the Terminal. It is faster and more powerful than clicking through menus. With Trap Silicon, you get an AI partner named Claude that lives inside this Terminal and can build real software for you just by you telling it what you want in plain English.

KEY CONCEPT

The Terminal is not something you need to download or install. It is already on your Mac right now. It has been there since you first turned on your computer. You are just learning how to use something you already own.

HOW TO OPEN THE TERMINAL

There are a few different ways to open the Terminal on your Mac. Pick whichever one feels easiest to you.

METHOD 1 — SPOTLIGHT SEARCH (MAGNIFYING GLASS)

1

OPEN SPOTLIGHT

Press **Command + Space** on your keyboard (hold the Command key and tap the spacebar). A search bar will pop up in the middle of your screen. This is called Spotlight — it is your Mac's built-in search. You can also click the magnifying glass icon in the top-right corner of your screen to open the same thing.

2

TYPE "TERMINAL"

Just start typing the word Terminal. It will appear as the top result almost immediately.

3

PRESS ENTER

Hit the Return (Enter) key. The Terminal app will open up. You will see a dark or white window with a blinking cursor waiting for you to type something. That blinking cursor is your command line. You are in.

METHOD 2 — THE DOCK (BOTTOM OF SCREEN)

1

FIND TERMINAL IN LAUNCHPAD

Look at the bottom of your screen — that strip of applications is called the Dock. Click the Launchpad icon (it looks like a rocket ship). Once Launchpad opens, use the search bar at the top and type Terminal. Click on it to open.

2

PIN IT TO YOUR DOCK

Once Terminal is open, you will see the icon appear in the Dock at the bottom. Right-click that icon by holding Control and clicking it. Select **Options → Keep in Dock**. Now Terminal will stay in your Dock permanently so you can open it with one click anytime.

METHOD 3 — FINDER

1

NAVIGATE MANUALLY

Open Finder. Go to **Applications** → **Utilities** → Terminal. Double-click to open.

PRO TIP

The fastest way will always be **Command + Space**, then type "Terminal" and hit Enter. Once you do this a few times, it becomes second nature. Two seconds and you are in.

ACTIVATING CLAUDE & OTHER AI

Once the Trap Silicon dev pack is installed, you turn on your AI coding assistant by simply typing its name into the Terminal. That is it — just type the name and hit Enter.

LAUNCHING CLAUDE

With the Terminal open, type:

```
claude
```

Press Enter. Claude will start up right inside your Terminal. You will see a place where you can type regular sentences — just talk to it like you would talk to a real person. Tell it what you want to build, and it builds it.

On your very first session, type:

```
/init
```

This runs the first-time setup. It checks that all your tools are installed correctly, connects Claude to your files and folders, sets up your workspace, and makes sure everything is good to go.

OTHER AI TOOLS YOU CAN LAUNCH BY NAME

COMMAND	WHAT IT LAUNCHES	WHAT IT DOES
`claude`	Claude Code (Anthropic)	Your main AI coding partner. It can read your files, write code, run commands, and put apps online.

COMMAND	WHAT IT LAUNCHES	WHAT IT DOES
`python3`	Python Interpreter	Run Python programs — good for AI projects, data work, and automating repetitive tasks.
`node`	Node.js Runtime	Run JavaScript code on your computer — used to build servers and the behind-the-scenes parts of apps.
`bun`	Bun Runtime	A faster version of Node.js. Does the same job but runs quicker.
`gh`	GitHub CLI	Manage your code on GitHub (where developers store and share code) without leaving the Terminal.

REMEMBER

You do not need to memorize any of these. The only word you need to remember is Claude. Once Claude is running, you can ask it to do everything else for you using normal everyday language.

WHAT YOU CAN BUILD

With Trap Silicon installed and Claude running in your Terminal, you can build real, working software just by describing what you want in normal words. No coding experience needed.

SAY THIS TO CLAUDE	WHAT GETS BUILT
`new app MyApp`	A complete phone app (works on iPhone and Android) with sections, buttons, and navigation already set up.
`new web MySite`	A professional website that looks good on phones, tablets, and desktops — ready to go live.
`new api MyBackend`	A server (the behind-the-scenes engine that powers apps) with security and routing already built in.
`new saas MyProduct`	A complete software product with user sign-up, login, payment processing, a database, and an admin dashboard.
`add auth`	Add a login and sign-up system so users can create accounts in your app.
`add payments`	Add the ability to accept credit card payments, subscriptions, or one-time purchases.
`add database`	Set up a database — a place to store all your app's data (users, orders, messages, etc.)
`deploy`	Put your app online so real people can use it — websites go live, phone apps go to the App Store or Google Play.
`fix [error message]`	Paste any error message you see — Claude will figure out what went wrong and fix it for you.

SAY THIS TO CLAUDE	WHAT GETS BUILT
`explain [code]`	Claude will break down any piece of code and explain what it does in simple terms.

THE POWER

You are not learning to write code yourself. You are learning to **direct** an AI that writes code for you. Claude does the building. You call the shots. That is the difference between being a worker and being the boss.

CLAUDE COMMANDS & SHORTCUTS

Once Claude is running in your Terminal, you can type these special commands that start with a forward slash (/). They give you quick access to useful features. Think of them like cheat codes.

ESSENTIAL SLASH COMMANDS

COMMAND	WHAT IT DOES
/init	First-time setup. Makes sure everything is installed and connected properly. You only need to run this once.
/resume	Picks up where you left off. If you closed the Terminal and came back later, this brings back your last conversation so you do not have to start over.
/compact	Shrinks your conversation down to save memory. Use this if you have been chatting with Claude for a long time and it starts getting slow or forgetting things.
/clear	Wipes the conversation clean and gives you a fresh start. Your actual files and code stay safe — it only resets the chat.
/help	Shows you a list of every command and shortcut you can use. Your built-in cheat sheet.
/config	Opens Claude's settings so you can change things like which AI model it uses and what it is allowed to do.
/cost	Shows how much usage you have spent in this session so you can keep track of your costs.
/doctor	Checks if Claude is set up correctly. If something is broken, it tells you what is wrong and how to fix it.
/permissions	Lets you control what Claude can and cannot do — like whether it can read your files, create new files, or run commands on your computer.
/mcp	Manage the connections between Claude and other tools on your computer — like your file system, databases, and online services. Think of these as bridges that let Claude talk to other software.
/commit	Saves a snapshot of your current code changes (like a save point in a video game) with a description of what you changed.

KEYBOARD SHORTCUTS (WHILE INSIDE CLAUDE)

Escape

Cancel what Claude is doing or go back a step

Ctrl + C

Emergency stop — forces Claude to stop whatever it is doing right now

Up Arrow

Bring back things you typed before so you can reuse them

Tab

Auto-fills the rest of a file name so you do not have to type the whole thing

Ctrl + L

Cleans up the screen so it is not cluttered (your session stays open)

Ctrl + A

Moves your cursor to the very beginning of what you are typing

Ctrl + E

Moves your cursor to the very end of what you are typing

Shift + Tab

Scroll through different options that Claude suggests

TERMINAL BASICS

These are the basic Terminal commands that are good to know. You do not need to memorize them — Claude can run all of these for you — but knowing what they do will help you feel more comfortable.

COMMAND	WHAT IT DOES	EXAMPLE
`ls`	Shows you all the files in the folder you are currently in.	`ls -la` shows hidden files too.
`cd`	Move into a different folder (like double-clicking a folder in Finder).	`cd ~/Projects/MyApp`
`pwd`	Shows you exactly which folder you are in right now.	`/Users/you/Projects`
`mkdir`	Makes a new folder.	`mkdir MyNewProject`
`touch`	Makes a new blank file.	`touch index.html`
`open .`	Opens the folder you are in as a regular Finder window so you can see it visually.	Useful to see your files visually.
`open file.html`	Opens a file using whatever app normally opens it.	Opens HTML in browser, PDF in Preview.
`clear`	Cleans up the screen so it feels fresh.	Just cleans up visual clutter.
`exit`	Closes the Terminal window.	Or just close the window.

NAVIGATION

cd .. takes you back up one folder (like hitting the back button). **cd ~** takes you to your home folder. **cd ~/Projects** jumps straight to your Projects folder no matter where you are. These three commands will get you anywhere.

These three commands will get you anywhere.

YOUR COMPUTER IS A BRAIN

Here is something most people never realize: your computer is built the same way as a human brain. Every part of your computer has a matching part in your head. Once you see the connection, everything about the Terminal, AI, and building agents starts to make sense.

PROCESSING

Brain: The thinking part (Cerebral Cortex)

Computer: The processor (CPU) — does all the thinking

SHORT-TERM MEMORY

Brain: What you are thinking about right now

Computer: RAM — holds what the computer is working on right now

LONG-TERM MEMORY

Brain: Your memories stored over years

Computer: Hard Drive / SSD — stores everything permanently

VISUAL PROCESSING

Brain: The part that processes what you see

Computer: GPU (Graphics Card) — handles all the visuals

NERVOUS SYSTEM

Brain: Nerves that connect everything together

Computer: Motherboard — the highway that connects all the parts

SENSES (INPUT)

Brain: Your eyes, ears, and sense of touch

Computer: Keyboard, mouse, camera, microphone

COMMUNICATION

Brain: Your ability to talk and communicate

Computer: WiFi / Internet — how it talks to the outside world

AUTONOMIC SYSTEM

Brain: The part that keeps your heart beating without you thinking about it

Computer: The operating system (macOS) — runs everything in the background

EXECUTIVE FUNCTION	SUBCONSCIOUS
Brain: The part that makes big decisions on purpose	Brain: Thoughts running in the back of your mind
Computer: The Terminal — where you give direct, intentional commands	**Computer: Apps and services running behind the scenes that you do not see**

When you open the Terminal, you are tapping into the decision-making center of your machine. You are not just mindlessly clicking around — you are giving your computer clear, direct instructions. You are thinking with your computer, not just using it.

Using your Mac the normal way (clicking icons, dragging windows) is like your brain on autopilot — it handles easy, everyday stuff. The Terminal is like deliberate, focused thinking — it gives you complete control and precision over what your computer does.

BUILDING AI AGENTS WITH INDIVIDUAL BRAINS

Now take the brain idea one step further. If a computer is a brain, then an AI agent is a custom-built brain designed to do one specific job really well. It can think, remember, learn, and take action — all focused on the one thing it was built to do.

WHAT IS AN AI AGENT?

An AI agent is a program that can understand what you tell it, make smart decisions, take real action, and learn from the results — all without you having to hold its hand through every step. It is not just a chatbot that answers questions. It is not Google. It is a digital worker that actually gets things done for you.

When you give an agent a brain file (like the CLAUDE.md file in Trap Silicon), you are giving it a personality, a memory, a set of rules, and a specific skill set. You are literally building a mind.

WHY INDIVIDUAL BRAINS MATTER

- **Specialization:** No human can be the best at everything at the same time. Same with AI. An agent built just for financial work will crush a general-purpose AI at financial tasks every single time. When you focus a brain on one thing, it masters it.
- **Memory:** Each agent remembers. It knows your preferences, your past projects, and your code. It does not forget everything when you close the window — it builds up knowledge over time, just like a person who gets better with experience.
- **Autonomy:** A good agent does not need you babysitting it. You tell it "build me a landing page" and it figures out the plan, writes the code, tests it, and hands you the finished product. It works on its own.
- **Scalability:** You can only hire so many people. But you can run 10, 100, or even 1,000 AI agents at the same time — each one focused on a different task — working around the clock. They do not sleep, they do not take days off, and they never lose focus.

- **Getting Smarter Over Time:** Every task an agent finishes makes it better at the next one. Its brain grows. It recognizes more patterns. After a few weeks, an agent with its own dedicated brain becomes way more useful than a basic AI that starts fresh every time.

THE TRAP SILICON BRAIN FILE

The CLAUDE.md file that comes in your Trap Silicon dev pack is Claude's brain. Here is what is inside it:

- **Identity** — who Claude is and how it acts (straight to the point, fast, always gives you complete working code)
- **Standards** — rules it follows on every project so your code is always clean and secure
- **Protocols** — step-by-step instructions for how it sets up, builds, and launches projects
- **Memory** — knowledge that it keeps from one session to the next so it does not forget
- **Commands** — quick shortcuts so you can get things done faster

When you copy CLAUDE.md to your home folder, Claude reads it every time it starts up. It stops being a generic chatbot and becomes your personal, skilled engineering partner who already knows how you like things done. That is the power of giving an AI its own brain.

THE VISION

One person with the right AI agents can now do the work of an entire team. You are not replacing people — you are multiplying yourself. Every agent you build is like cloning your thinking and putting it to work at machine speed, 24/7, even while you sleep. That is what Trap Silicon unlocks.

FROM ZERO TO BUILDING

Here is the full step-by-step path from opening your laptop to actually building software with AI. Follow these in order.

1

OPEN TERMINAL

Press **Command + Space**, type Terminal, press Enter.

2

INSTALL PREREQUISITES

Run the three setup commands from the README file that comes in the pack. These install Xcode Tools (Apple's developer kit), Homebrew (a tool installer), and Node.js (the engine that runs JavaScript).

3

INSTALL CLAUDE CODE

```
npm install -g @anthropic-ai/claude-code
```

4

COPY BRAIN FILE

```
cp CLAUDE.md ~/CLAUDE.md
```

5

RUN INSTALL SCRIPT

```
bash install.sh
```

Wait 60 minutes. Restart Terminal when done.

6 LAUNCH CLAUDE

```
claude
```

7 INITIALIZE

```
/init
```

8 BUILD

Tell Claude what you want: **new app MyFirstApp** — and watch it build.

WHAT GETS INSTALLED

The Trap Silicon install script automatically sets up all the tools a professional developer uses. Here is everything it puts on your machine:

CATEGORY	TOOLS
Runtimes	Node.js LTS (via NVM), Python 3, Go, Rust, Bun
Mobile Development	Expo CLI, EAS CLI, React Native CLI, CocoaPods, Watchman
Web Development	Next.js scaffolder, Vite CLI, Tailwind CSS
Backend & Database	Railway CLI, Supabase CLI, Prisma, Drizzle ORM
AI & Machine Learning	Anthropic SDK, OpenAI SDK, LangChain, FastAPI
DevOps & Testing	PM2, Nodemon, serve, http-server, ngrok, Playwright
Code Quality	TypeScript, ts-node, ESLint, Prettier
System Utilities	Git, GitHub CLI, FFmpeg, tree
AI Terminal	Claude Code with MCP servers (filesystem, memory, GitHub)

TRAP SILICON

BUILD ANYTHING. SHIP TODAY.

C/O JOSHUA KALEB GRAHAM-BERRY

SIOS V101 | 2026