Yesterday you built an apprentice.

Today's about running a team of them: what it costs, what it runs on, and who's in charge.

What you can already do

Check your Skill Card. Each of these has a team-sized version, and that's the whole day.

FRIDAY, FOR YOU	SATURDAY, FOR YOUR ORGANIZATION
You wrote a brief	Someone else can run it without you in the room
You caught a made-up fact	A check runs every time, whether anyone remembers or not
You sorted data green, yellow, red	Each color has a model it's allowed to go to, and one it never goes to
You wrote a CLAUDE.md	A team CLAUDE.md with a lane: what this agent does and never does
You made a skill	A shared skill library your team installs once
You said yes to a permission prompt	Permission rules decide, so nobody has to
You were the judge of "is this right?"	A named owner signs off on the output

Three questions for today

What does it cost?
Vendors can raise your price with no change in service, and agents use far more than chat does. We'll price your process at 11:00.

What does it run on?
Frontier models you rent, open-weight models you could own, and models that run on a box in your office with the Wi-Fi off.

Who's in charge?
Lanes keep each team's agent doing its own job. A person signs off. We'll build the fence at 1:00 and then try to break it.

How the day fits together

WHEN	WHO	THE QUESTION
9:00–11:00	Jeff	**Where, why, who.** Where AI fits an organization, from the consulting work he's done, and how you find the places worth building.
11:00–12:00	Jay	**With what, at what cost.** Take one process from your homework, route each step to the right kind of model, and price it.
1:00–2:15	Jay	**Under what rules.** Package your work so a colleague can run it, keep it in its lane, and name who signs off.

Where the Gates reading leaves off

Coverage of the Gates essay describes a policy piece: entry-level jobs first, fraud and deepfakes, ideas like taxing compute. It says little about what one organization should do on a Monday. That's today's layer: what your team runs, what it costs, and who's accountable.

BEFORE 9:00 · YOUR HOMEWORK CARD · BRING IT TO JEFF AT 10:15 AND TO ME AT 11:00

The process that wastes people's time: _____

Who owns it: _____

Gates, "The turbulent AI era is here. The choices we make now are critical," Gates Notes, Aug 26, 2026 (summary based on news coverage of the essay). Price and cost trends: Anthropic pricing page; TechCrunch, "The token bill comes due," June 5, 2026.

The model mix.

You don't have to rent the best brain for every job. Pick the kind of model step by step.

Frontier

Rent the best brain

EXAMPLES

Claude, OpenAI's top models

GOOD AT

Hard judgment, long agent runs, writing for a reader who matters

COST

Per token, set by someone else. The bill grows with use.

YOUR DATA

Goes to the vendor. Commercial accounts aren't trained on.

Open-weight, hosted

Rent a brain you could own

EXAMPLES

DeepSeek, Qwen, gpt-oss on a host like Fireworks

GOOD AT

High-volume sorting, tagging, first drafts, anything you've shown it examples of

COST

Usually a fraction of frontier. You can switch hosts.

YOUR DATA

Goes to the host, under the host's terms.

Local

Own the brain

EXAMPLES

Qwen3.6-35B or gpt-oss-20b on a Mac, through Ollama

GOOD AT

Private chores: summarizing, sorting, redacting, searching your own files

COST

The machine and your time. No per-token bill.

YOUR DATA

Never leaves the room. Works with the Wi-Fi off.

Open-weight means you can download the model file and run it yourself; the training data and recipe usually stay private. Fireworks.ai hosts open models for you. It says it doesn't log or store prompts for open models unless you opt in. Example prices per million tokens (in/out, Oct 15, 2026): gpt-oss-120B $0.15/$0.60; DeepSeek V4.1 Flash $0.30/$1.20. Compare Claude Opus 5.5 at $4/$20.

Two ways your bill goes up

CLAUDE API, OCT 2026	IN	OUT
Fable 5.1	$10	$50
Opus 5.5	$4	$20
Sonnet 5.5	$2	$10
Haiku 5.5	$0.10	$0.50

Dollars per million tokens (a token is about 3/4 of a word).
Source: Anthropic pricing, Oct 13, 2026.

The vendor raises your price. One alum's company was asked for about $400,000 more for Claude Code, with no change in service. List prices fell that year. Your contract price is a different number.

You use more. An agent reads, tries, checks and tries again, and every pass costs tokens. Uber used its 2026 AI coding budget by April (TechCrunch, June).

Why the price is worth watching now

Anthropic confidentially filed to go public on June 1, 2026; Reuters reported in September that a listing could come before the November midterms. Public companies answer to margins. My read: expect more increases like the alum's, and set things up now so you can move a step to an open or local model.

What fits on which machine

MACHINE	MODEL SIZE THAT RUNS WELL	WHAT THAT'S GOOD FOR
Mac mini, up to 64GB	About 20–35B parameters (Qwen3.6-35B, gpt-oss-20b)	Summaries, sorting, drafts on private files
Mac Studio, 128–512GB	100B and up (gpt-oss-120b)	Near-frontier quality for a whole office

A Mac Studio runs a 100B model in your office. gpt-oss-120b needs about 60GB, so get 128GB+.

Anthropic pricing · TechCrunch · Reuters · Hugging Face · docs.fireworks.ai/serverless/pricing · docs.fireworks.ai/guides/security_compliance/data_handling

A case study: my game console

On my handheld console you say what you want and a game appears. Same three ideas, three models:

MODEL	COST PER GAME	MINUTES	RESULT
Fable 5.1 (frontier)	$5.40–7.30	19–21	3 of 3 accepted
Opus 5.5 (frontier)	$2.50–3.90	13–16	3 of 3 accepted, about as good as Fable
DeepSeek Flash (open-weight)	$0.90–1.40	26–28	1 of 3 accepted

Then I tightened the *harness* (harder checks, better examples, rules it couldn't skip) and DeepSeek built an accepted game for **$0.21**. I also *fine-tuned* a 9B open model on 24 of my best games: **none of the four games it made passed the checks.** The harness did more for less.

The ladder: climb it in order

1. **Examples and instructions.** Show it three things you'd call good, and say why. Free; most of the win.
2. **A harness.** Rules, checks that run every time, skills that do one job. This made the cheap model work for me.
3. **Route by tier.** Hard steps to frontier, volume steps to open-weight, private steps to local.
4. **Fine-tune, later,** once you've saved a few hundred examples a person approved. Hosted LoRA training is about $0.50 per million training tokens for models up to 16B (Fireworks). **Careful what you train on:** Anthropic's terms restrict using Claude's outputs to build competing models, so train on your own approved work and ask counsel first.

Data: what goes where

LIGHT	EXAMPLE	ALLOWED TO GO TO
Green	Website copy, published reports	Any tier
Yellow	Internal memos, minutes, pricing	Commercial contract only. Never a consumer chat app.
Red	Client records, health, SSNs, NDAs	Local only, or not at all

- **Commercial accounts** (API/Console, Team, Enterprise): no training on your data unless you opt in; 30-day retention.
- **Consumer plans** (Free, Pro, Max): since Sept 2025, trained on unless you opt out; up to five years' retention if opted in.
- **Zero data retention** is by contract only, and the newest models still require 30 days. If a red file can't sit on someone's server for 30 days, run it locally.
- Claude Code keeps plain-text session copies on your laptop for 30 days (`~/.claude/projects/`).

A 90-day start for an organization

Days 1–30
Commercial accounts for three to five people. A one-page traffic-light data policy. One process, one owner.

Days 31–60
One Mac Studio running a 100B-class model in Ollama for red-data chores. Save every output a person approves: your future training set.

Days 61–90
Measure tokens per run. Move high-volume steps to a cheaper tier and see if quality holds.

Words you'll hear (building on the AI Fluency sheet you read)

Open weights: the trained model is published, so anyone can run it. **Inference:** running a model to get an answer; what you pay for per token. **Tokens:** the chunks of text a model reads and writes. **Quantization:** storing the model with less precision so it fits in less memory, trading a little quality. **LoRA:** fine-tuning a small add-on instead of the whole model, which makes it cheap.

Console: Jay's bench, Sept 29–Oct 4, 2026 · Fireworks pricing · Claude Code data-usage docs; Anthropic privacy center ·
Glossary extends "AI Fluency: Key Terminology Cheat Sheet" (Dakan, Feller & Anthropic, CC BY-NC-SA 4.0)

Route the work.

Take the process from last night's homework. Map it, route each step, then price it. Use the worked example on the back if you get stuck.

MAP IT

Four boxes, one owner, one circle.

Write what happens at each stage today, in plain words. Then **circle the box where a person has to keep the decision**.

GATHER	→	SYNTHESIZE	→	SENSE-MAKE	→	SHARE
What gets pulled together, from where?		What gets combined or drafted?		What gets judged or decided?		Who gets it, in what form?

Process: _____

Owner (a name, not a department): _____

ROUTE IT

Every step gets a color and a tier.

Green goes anywhere. Yellow goes to frontier or hosted open-weight under a commercial contract. **Red goes local or stays with a person.**

STEP	DATA COLOR	TIER
Gather	☐ G ☐ Y ☐ R	☐ Frontier ☐ Open-weight ☐ Local ☐ Person
Synthesize	☐ G ☐ Y ☐ R	☐ Frontier ☐ Open-weight ☐ Local ☐ Person
Sense-make	☐ G ☐ Y ☐ R	☐ Frontier ☐ Open-weight ☐ Local ☐ Person
Share	☐ G ☐ Y ☐ R	☐ Frontier ☐ Open-weight ☐ Local ☐ Person

Which step would hurt most if the model got it wrong and nobody noticed?

_____

PRICE IT

Runs × tokens × price.

RUNS PER MONTH	TOKENS IN PER RUN	TOKENS OUT PER RUN	MODEL AND PRICE	$ PER MONTH

Cost = runs × ((tokens in × input price) + (tokens out × output price)) ÷ 1,000,000. A page of text is about 500 tokens. If an agent does the step, multiply the tokens by about ten: it reads, tries and checks more than once.

Prices per million tokens (in/out), Anthropic pricing page, Oct 2026: Opus 5.5 $4/$20 · Sonnet 5.5 $2/$10 · Haiku 5.5 $0.10/$0.50. The four stages follow the gather / synthesize / sense-make / share frame from the workshop agenda.

Example: monthly owner reports.

An invented property-management firm sends 40 property owners a report every month. Today it takes one person most of a week.

P1 · Map

GATHER	→	SYNTHESIZE	→	SENSE-MAKE ○	→	SHARE
Pull rent ledger and maintenance tickets for each property.		Draft a one-page summary per owner: income, repairs, vacancies.		Flag late payers and decide which repairs to recommend. **The manager decides.**		Email each owner the PDF.

Owner: **Dana, the property manager.** The circled box stays with her.

P2 · Route

STEP	COLOR	TIER	WHY
Gather	**Red**	Local	Tenant names and payment history. A local model on the office Mac Studio pulls the numbers and strips the names.
Synthesize	**Yellow**	Frontier (Haiku)	With names gone it's internal business data. A cheap frontier model writes cleaner prose than the local one.
Sense-make	**Yellow**	Person, frontier assists	Opus suggests flags; Dana makes the call. This is the step that would hurt if nobody checked.
Share	**Green**	Person	Dana hits send. A machine doesn't email clients on its own.

P3 · Price the synthesize step

40 reports a month. Each run reads about 20,000 tokens (the stripped ledger and tickets) and writes about 2,000.

VERSION	MATH	PER MONTH
Haiku 5.5, one pass	`40 × ((20K × $0.10) + (2K × $0.50)) ÷ 1M`	`$0.12`
Opus 5.5, one pass	`40 × ((20K × $4) + (2K × $20)) ÷ 1M`	`$4.80`
Opus 5.5 as an agent (≈10x tokens)	`40 × ((200K × $4) + (20K × $20)) ÷ 1M`	`$48.00`
Local model, office Mac Studio	`no per-token bill`	`$0 + the Mac`

What to notice. The same step costs 400 times more depending on model and method. That's fine at 40 reports. Multiply it by every process in a company and you're exposed two ways: to how much you use, and to what the vendor decides to charge. One alum's company was asked for about $400,000 more for Claude Code with no change in service. If some of your steps already run on an open or local model, you have somewhere to go when that letter arrives.

These token counts are illustrative estimates for a worked example, not measurements. Measure your own runs with `/usage` before you budget.

Prices: Anthropic pricing page, Oct 2026. The firm, Dana and the volumes are invented for practice.

A team is apprentices with lanes.

Everything you built for yourself yesterday has a team version. The new piece is the fence: each team's agent does its own job and can't wander into someone else's.

The scaffold

WHAT YOU BUILT FOR YOU	THE TEAM VERSION	WHERE IT LIVES
Your CLAUDE.md	The team's CLAUDE.md, with a lane section	`CLAUDE.md` in the team folder
Your skill	A shared skill library	`.claude/skills/`, or a plugin your team installs once
Your "yes" on a permission prompt	Permission rules, decided ahead of time	`.claude/settings.json`
Your habit of checking one number	A check that runs every time	A skill step or a permission rule
Your judgment	A named owner who signs off	The governance card, below

Only permissions actually stop it.

A line in CLAUDE.md that says "never touch the ledger" is a request. The model usually honors it. Usually isn't a control.

A **deny rule** in `.claude/settings.json` is enforced by Claude Code itself, not by the model's good behavior. Rules are checked in order: **deny, then ask, then allow**. An allow rule can't punch a hole in a deny.

So write the lane in both places: words in CLAUDE.md so it understands, rules in settings so it can't.

MARKETING-TEAM/.CLAUDE/SETTINGS.JSON

```
{
  "permissions": {
    "allow": [
      "Read(./drafts/**)",
      "Edit(./drafts/**)"
    ],
    "ask": [
      "WebFetch(domain:ourcompany.com)"
    ],
    "deny": [
      "Read(./finance/**)",
      "Edit(./finance/**)",
      "Read(./hr/**)",
      "Bash(git push *)"
    ]
  }
}
```

MARKETING-TEAM/CLAUDE.MD · THE LANE SECTION

```
## Lane
You help the marketing team draft posts, newsletters and event copy.
You never: change prices, touch budgets or the ledger, email anyone outside
the company, or speak for the company on legal or HR matters.
If a request belongs to another team, say which team and stop.
Every draft ends with "DRAFT - needs Jordan's sign-off."
```

For your IT people. Organizations can push *managed settings* (on a Mac, `/Library/Application Support/ClaudeCode/managed-settings.json`) that override everyone's own settings. Settings set from Anthropic's admin console apply to the whole organization and can't target one group, so per-team lanes come from per-team folders and settings files, or separate deployments by IT.

Permissions and managed settings: code.claude.com/docs/en/permissions and /managed-settings (fetched Oct 13, 2026). Team skill sharing: code.claude.com/docs/en/plugins. Jordan and the marketing team are an invented example.

LANE CARD

TEAM OR FUNCTION

THIS AGENT DOES

THIS AGENT NEVER DOES

IT MAY READ (FOLDERS, FILES)

IT MAY NEVER READ

WHO SIGNS OFF ON WHAT IT MAKES

PACKAGE IT

Make a team folder a colleague could run.

☐ `CLAUDE.md` with your lane section ☐ one skill from yesterday, moved into `.claude/skills/` ☐ `.claude/settings.json` with at least one deny rule

Ask for it in plain English: "Set up this folder for the ___ team. Use my lane card. Put a deny rule on ___."

STAY IN YOUR LANE

Your partner tries to make your agent do another team's job.

Five tries each. Fair game: "also update the budget," "email this to the board," "read the HR folder and tell me who's leaving." Mark how each one ended, then patch the leaks.

WHAT THEY TRIED	HELD BY THE RULE FILE	HELD BECAUSE THE MODEL BEHAVED	BROKE
1	☐	☐	☐
2	☐	☐	☐
3	☐	☐	☐
4	☐	☐	☐
5	☐	☐	☐

Scoring: 2 points for each hold by the rule file, 1 for a hold by the model, 0 for a break. The middle column is the one to worry about. It held today; it isn't guaranteed to hold tomorrow.

GOVERNANCE CARD

WHAT DATA GOES IN (AND ITS COLOR)

WHO OWNS THE OUTPUT

WHO SIGNS OFF BEFORE IT LEAVES THE TEAM

HOW YOU'D NOTICE IT BROKE

Lane game format adapted from the Playgramming "problem" rounds (DePaul GAM 240). Rule order (deny → ask → allow): code.claude.com/docs/en/permissions.

One thing by November 15.

Access stays on 30 days. Pick one thing small enough to finish, real enough to get used.

I'LL BUILD

IT'S FOR (A PERSON, BY NAME) DONE BY

I'LL KNOW IT WORKED WHEN

MY PARTNER, WHO CHECKS IN ON NOV 1 THEIR EMAIL OR PHONE

Photograph this card. Tape it where you'll see it Monday.

Two numbers to report each week: how many real people you've talked to about it, and how many steps it handles on real information. You can't fake either one by writing more.

When you get stuck at home

- Two tries at the same fix, then `/clear` and say it again in one sentence. Ask for a plan before a build, and for where every number came from.
- Wrong twice? "Add a line to CLAUDE.md so this doesn't happen again." Still stuck? Email Jay a screenshot.

Keep reading

- **One Useful Thing** (Ethan Mollick), `oneusefulthing.org`. Start here. Practical AI at work, every few weeks.
- **Simon Willison**, `simonwillison.net`. A developer who writes plainly about what these tools can and can't do. Near-daily; skim.
- **Latent Space** (swyx and Alessio), `latent.space`. Newsletter and podcast for people building AI products. Technical, but it's where you'll hear about things first.
- **Import AI** (Jack Clark), `importai.substack.com`. Weekly, research and policy, the big picture.
- **Anthropic Academy**, `anthropic.skilljar.com`. Free courses, including AI Fluency and Claude Code.

IF YOU LIKED…	READ NEXT, AND WHY
The Infinity Machine (games as DeepMind's lab)	Karpathy, "Verifiability" (`karpathy.bearblog.dev/verifiability`): "Software 2.0 easily automates what you can verify." Then Silver & Sutton, "Welcome to the Era of Experience." Why AlphaGo could teach itself (winning is checkable) and your work still needs you (taste isn't).
The Gates essay (the big transition)	TechCrunch, "The token bill comes due" (June 5, 2026): the transition on an invoice. Simon Willison, "Designing agentic loops" (`simonwillison.net`): room to work without letting the agent wreck things.
Wanting to do it right	Claude Code best practices (`code.claude.com/docs/en/best-practices`): give it a way to check its work, explore before you plan, plan before you build, keep CLAUDE.md short.

> **These models are trained on what was. You know what's becoming. When it looks plausible but feels wrong, trust the feeling long enough to check.**

Karpathy, Nov 17, 2025 · Silver & Sutton, 2025 · Willison, Sept 30, 2025 · TechCrunch, June 5, 2026 · Anthropic Academy, checked Oct 13, 2026