

1-HOUR CRASH COURSE

AI Coding *Mastery.*

From zero → building real things with AI. No math. No fear. Just the bare essentials you actually need today.

FACILITATOR

Simon Covenant

FORMAT

60 Minutes

LEVEL

Absolute Beginner

THE PROMISE

By the end of this hour, you
stop *guessing* and start *directing*.

- ✗ Understand what AI really is
No jargon, real intuition
- ✗ Know which tool for which job
Stop tool-hopping
- ✗ Understand how apps work
Plain English, no fear
- ✗ Talk to AI so it actually delivers
Context beats clever prompts
- ✗ Use AI as a thinking partner
For work and daily life
- ✗ Build + ship your first AI tool
Idea → working product

AGENDA

Seven moves, one hour.

00:00	Mindset reset	what AI really is	8 MIN
00:08	Talk to AI right	context & cost	10 MIN
00:18	The toolbox	the AI landscape	8 MIN
00:26	Think with AI	your daily partner	8 MIN
00:34	How software works	plain English	6 MIN
00:40	Build your first app	live	14 MIN
00:54	Ship it & stay a builder		6 MIN

MENTAL RESET

What AI *really* is.

Kill the intimidation first. Then everything else gets easy.

NO MATH, REAL INTUITION

AI is a *prediction engine* — it guesses the next best word.

i

It pattern-matches

Trained on huge text. It doesn't "know" — it predicts what usually comes next.

ii

It mirrors you

Vague in → vague out. Clear, specific in → sharp, useful out.

iii

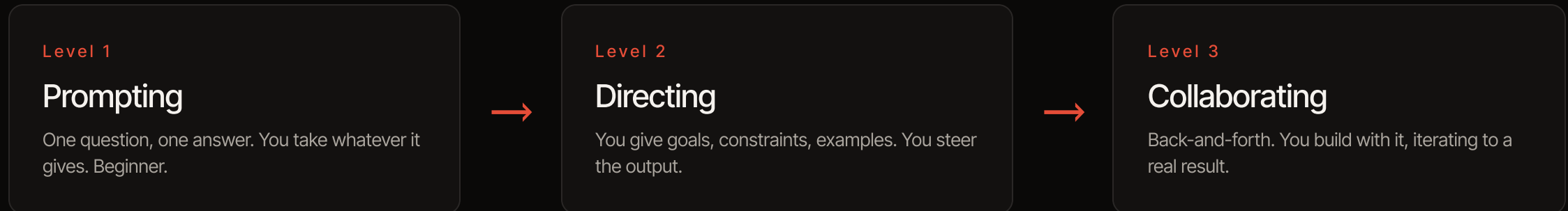
It needs direction

It won't read your mind. You supply the goal, the context, the standard.

AI doesn't replace builders. It replaces *guessers*.

LEVEL UP

Most people get stuck at level one.



Today we move you from *asking* → to *directing* → to *building*.

CORE AI LITERACY

Talk to AI the *right* way.

Context is the cheat code. Cost control keeps
you in the game.



THE #1 SKILL

Context is *everything*. Same question, different answer.

NO CONTEXT

"Write me an email."

→ Generic, robotic, useless. You rewrite it all anyway.

LAYERED CONTEXT

"You're my sales assistant. Write a warm 4-line follow-up to a buyer who toured a 3-bed in Lekki yesterday but went quiet. Friendly, no pressure."

→ Ready to send.

Give it a *role*, the *goal*, the *details*, and the *tone*. Stack those four every time.

CHEAP BUT POWERFUL

Tokens are the *fuel*. Don't burn money by accident.

i

Tokens = words-ish

AI reads & writes in tokens (~¾ of a word). You pay per token, in and out.

ii

Long ≠ better

Walls of text and giant pastes drain credits fast for little gain.

iii

Be lean

Clear instruction, only the context that matters, ask for the format you want.

Rule of thumb: *say more with less*. Tight prompts cost less and work better.

THE TOOL ECOSYSTEM

The right tool for the *job*.

Stop tool-hopping. There's a map — and
it's simple.

MATCH TASK → TOOL

Five families. Pick by the job, not the hype.

CHAT / REASONING

Claude, GPT, Gemini

Writing, thinking, planning, coding help.

YOUR DAILY DRIVER

CODE

Cursor, Claude Code

Build & edit real apps with AI.

WHEN YOU BUILD

IMAGE / VISION

Midjourney, DALL·E

Generate & read images.

VISUALS & DESIGN

VOICE / SPEECH

ElevenLabs, Whisper

Voiceover, transcripts.

AUDIO & CALLS

And know *when NOT to use AI* — anything needing certainty, legal/medical facts, or a human relationship.

PRIVACY & POWER

Cloud is *powerful*. Local is *private*.

CLOUD AI

Claude, GPT, Gemini. Most powerful, always updated, easiest to start.

Best for: almost everything you'll do today.

LOCAL AI · LM STUDIO

Runs on your own computer. Free, offline, fully private — your data never leaves the machine.

Best for: sensitive data, no internet, experiments.

Start in the cloud. Reach for local when *privacy* or *cost* demands it.

THINKING PARTNER

Think *with* AI.

Don't copy from it. Reason with it — for work
and for life.



YOUR DAILY PARTNER

Teach it your *goal* — not just your question.

1

Decompose

Break the problem into small steps before asking.

2

Give the goal

Say what success looks like, not just the task.

3

Loop

"Make it shorter / sharper / simpler." Refine.

4

Ask why

"Explain your choices." Catch the weak spots.

Daily uses: *writing · research · planning · learning faster · decisions*. Turn it into your personal assistant.

NO CODE FEAR

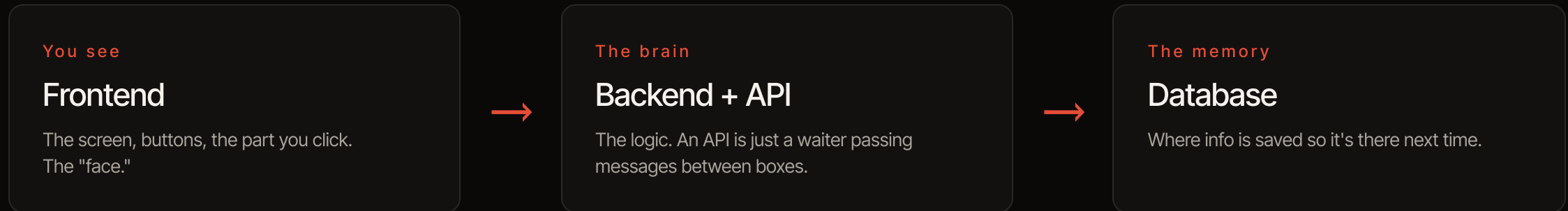
How software *really* works.

Boxes and arrows. That's the whole secret.



EVERY APP, SIMPLIFIED

An app is just three boxes talking.



AI plugs in as the *brain* — you send it text, it sends back smart answers.

BUILD IT LIVE

Scope *tiny*. Most people fail by overscoping.

1

One idea

Summarizer, idea generator, or chat helper. Pick one.

2

Scope small

One job, one screen. Nothing extra. Ship the smallest version.

3

Let AI code it

Describe it plainly. AI writes the code. You direct & review.

4

Run it

Open it locally. It works. You just built software.

Then improve: *better prompts · a cleaner screen · a little memory*. Working first, pretty second.

FROM STUDENT → BUILDER

You don't need permission to *build*.

✗ Ship it — demo & explain simply

Turn projects into products

✗ Keep learning as tools change

Avoid AI dependency — direct it

✗ Solve a real problem for a real person

Problem · user · AI solution

✗ Build responsibly

Ethics & honesty matter

Your challenge: build *one* tiny AI tool this week. Then tell us why it exists, how it works, and who it helps.

@iamccurves — everywhere.