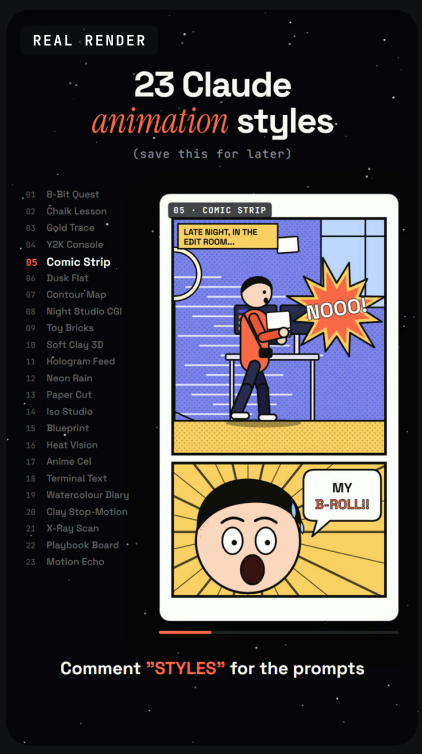
INSTAGRAM REELS WITH CLAUDE CODE

DIRECT, DON'T *edit.*

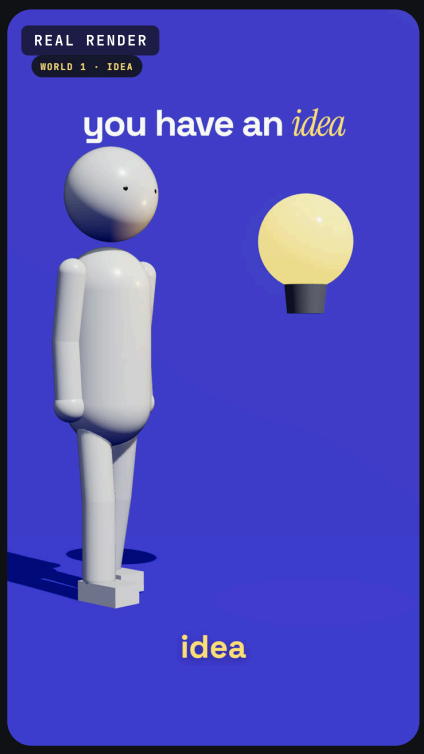
The complete system for turning ideas, raw clips and scripts into scroll-stopping Reels by describing them. Talking head, faceless motion and 3D explainers, from zero to a studio you can sell.



01 · TALKING HEAD + ANIMATION



02 · FACELESS MOTION



03 · 3D EXPLAINER

CONTENTS

Your *roadmap*

Seventeen Parts, from an empty computer to a studio you can sell. Each Part ends with something you can use.

01 **The new way** *to edit* — 009

Editing just changed · What Claude Code actually is · Why an AI can edit at all · Your timeline is data · The three roles · Great at, breaks at, needs you

02 **Setup** *from zero* — 019

Two ways in · The checklist · Install Claude Code · Runtimes + VERSIONS.md · FFmpeg + ffprobe · Transcription choice

03 **The toolkit in** *plain English* — 047

The pipeline map · FFmpeg and ffprobe · FFmpeg jobs in plain English · Whisper: speech becomes data · words.json, your real timeline · Cloud speech and AI voices

04 **Learn any style from** *references* — 068

Why references beat adjectives · Picking 3 to 5 references · Download and break them down · Measuring a frame · What to extract · What never to take

05 **Skills: teaching Claude** *your* **styles** — 081

What a skill is · Anatomy of a skill · Scope, discovery, naming · Build from evidence · Style skill template · The three shelves

06 **Prompting that gets it** *right* — 096

Quality starts before the edit · The five-part brief · Q1 Where is it going? · Q2 Length is a budget · Q3 Edit policy · Q4 The look as a system

07 **Directing every** *frame* — 113

Why every detail matters · Start here: five fields · The Frame Spec: 14 fields · Frame Spec template · The canvas and the safe zone · Layout zones

08 **Talking head** *+ animation* — 138

Anatomy of a great talking-head Reel · Film for the edit · Layout 1: Split tutorial · Layout 2: Effects menu · Layout 3: Command console · Layout 4: Explainer above the head

09 **Faceless** *motion* — 163

Faceless isn't "minus a face" · Order of operations · AI voice that sounds human · Kinetic type from one file · The command → result format · Six Samo motion styles

CONTENTS, CONTINUED

10 **3D** *explainers* 178
Three ways to do 3D · 3D basics in plain English · The 3D phone Reel · Mascot worlds · The Pill Body character sheet · Literal props library

11 **The Samo Animation** *Style Pack* 194
How the pack works · The Samo Base Scene · The Pill Body and the walk cycle · Your first style run · The style index · The 23 styles

12 **Nine brand** *looks* 231
How to use the looks · Clean Studio · Neon Terminal · Bold Pop · Cinematic Dark · Paper & Marker

13 **Sound makes the edit feel** *physical* 245
The four layers of sound · Voice: get the source right · Find: Openverse · Licences in plain English · Make: synthesise your own SFX · Place: the SFX map

14 **Hooks, scripts and** *retention* 260
The opening second · Four ways hooks fail · Six hook formats · Match the first frame · The lock-in: seconds 3 to 8 · Script structure

15 **Review, QA and** *export* 274
Verify, don't trust the log · Contact sheets and trigger frames · The Direct check · ffprobe: the file report · Loudness and the phone test · The 30-point release checklist

16 **Templates, batch and the** *prompt library* 286
Chat memory is not a spec · Template, skill, batch · Library: ingest and transcript · Library: cuts · Library: graphics and style · Library: captions

17 **Publish, test, grow** *and sell* 298
Before you post · Trial Reels: test one variable · Read results by timestamp · Build a series, not a post · Your next 10 Reels · Selling Reels to clients

Also inside

56
copy-paste prompts in the library (Part 16)

43
samo- skills mapped across three shelves (Part 05)

30
point release checklist before every post (Part 15)

56
glossary terms in plain English (end of Part 17)

HOW TO READ THE NUMBERS

Chapters are numbered by Part: **07.4** means Part 07, chapter 4. When a page says "see Part 13", jump with the contents above. The page number is always bottom right.

01

PART 01

The new way *to edit*

Before you install anything, you need the right picture in your head. This Part explains what Claude Code really does with video, why it can edit without a timeline, and where your taste still beats any machine.

BY THE END OF THIS PART YOU WILL

- Explain in one sentence what Claude Code is and what it isn't
- See a video as three clocks: words, frames and audio
- Know your job (director) versus Claude's job (operator)
- Know which tasks to hand over and which to keep
- Have a 30-day plan from zero to a repeatable studio

Editing just changed	01.1	The three roles	01.5
What Claude Code actually is	01.2	Great at, breaks at, needs you	01.6
Why an AI can edit at all	01.3	Short form vs long form	01.7
Your timeline is data	01.4	The 30-day plan	01.8

01.1 · EDITING JUST CHANGED

The skill moved from your hands *to your words*

For twenty years, a good video meant months of learning editing software: tracks, keyframes (marked points where a change starts or stops), cutting tools and export panels. That skill still exists, but it's no longer the only way in.

What used to happen

You dragged clips onto a timeline. You cut each "um" by hand. You typed every caption, nudged every title, and keyframed every zoom. A 40-second Reel with motion graphics could take a full day.

What happens now

You describe the result. Claude Code writes the instructions and runs free tools that do the cutting, captioning, animating and rendering. A ten-minute raw clip can become a captioned, animated Reel in minutes of machine time. Your time goes into the decisions.

What did not go away

Taste and clarity. A vague request still gives a vague video. "Make it engaging" gets you an average Reel, because Claude fills every gap with a safe default, and safe defaults don't stop a scroll.

The new core skill is **knowing exactly what to ask for**, then checking that you got it. That's what the rest of this book trains.

> *The skill isn't editing anymore. The skill is knowing what to ask for.*

TASK	OLD WAY (TIMELINE)	NEW WAY (DIRECTION)
Remove fillers	Drag through the clip, find each "um", cut it, close the gap	"Flag every filler and false start in an edit map. Wait for my approval."
Captions	Type them, time them, style each line	"Word-by-word captions from the transcript, one zone 1240 px from the top, current word indigo."
A title card	Keyframe position, scale and opacity	"Headline slides up 64 px in 0.26 s with a fast ease-out (quick start, soft stop) on the word 'three'."
Ten versions	Duplicate the project ten times	"Make 3 variants. Change only the hook text. Keep everything else locked."

Reading the right column: px means pixels, the tiny dots a screen is made of. A Reel is 1080 px wide and 1920 px tall. s means seconds.

DO THIS NOW

Think of one Reel you love. Write three sentences describing exactly what happens in its first two seconds: what's on screen, what text appears, and what you hear. You've just written your first direction. Keep it, because you'll use it in Part 04.

01.2 · WHAT CLAUDE CODE ACTUALLY IS

A producer that can *use tools*

In one sentence: Claude Code is an AI agent that can read your project folder, write files and run programs on your computer while you direct it in plain language.

It is not a hidden video editor

There is no secret editing engine inside Claude. It coordinates real tools that already exist and are free. That's good news: the same tools professionals use do the heavy work, and nothing is a black box.

What it can do for video

- Inspect your files: length, size in pixels, frame rate (images per second), audio
- Run FFmpeg (a free video tool) to cut, crop, join, mix and export
- Run Whisper (free speech-to-text) to turn speech into timed words
- Write the code for motion graphics, captions and 3D scenes
- Render the video and pull out still frames to check its own work
- Measure loudness and compare the file with your rules
- Remember your standing rules (CLAUDE.md) and load step-by-step procedures (skills) only when needed

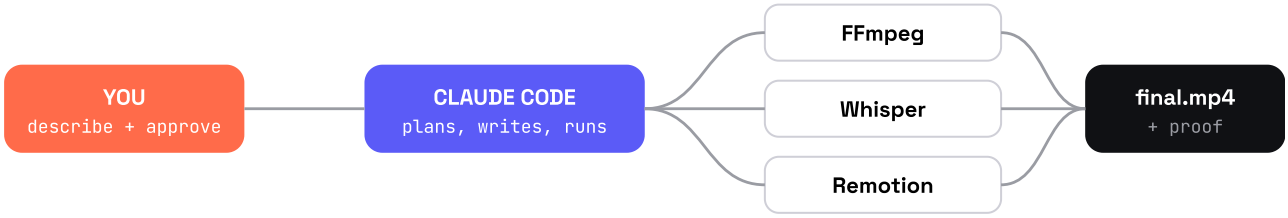
Where you can use it

SURFACE	BEST FOR
Claude desktop app, Code tab	Beginners. No terminal. Pick a folder and talk.
Terminal (CLI)	A text window where you type commands. Full control, fastest once you're comfortable
VS Code extension	A free code editor: files, each change and the chat side by side

All three run the same Claude Code. This book shows both the desktop-app route and the terminal route in Part 02.

ACCESS

Claude Code works with eligible paid Claude plans, a Claude Console (API) account, or supported enterprise cloud providers. Plans change, so check the official options before you buy.



CLAUDE COORDINATES. FREE TOOLS EXECUTE. YOU APPROVE.

PRO MOVE

Think "producer plus operator", not "AI editor". You set the acceptance criteria. Claude runs the tools and hands you evidence: a contact sheet (one image showing many frames in a grid), a frame, a measurement. If it can't show proof, it isn't done.

01.3 · WHY AN AI CAN EDIT AT ALL

Every editor is a friendly face on *three things*

Under the buttons, every video app does the same three things. Claude writes the instructions as text and runs the renderer directly, so it needs no timeline.

1

Files

Your clips, images, sounds, fonts and logos. They sit in folders. Claude can read a folder as easily as you read a list.

2

Instructions

"Keep 4.20 s to 6.85 s." "Show this title for 1.5 s." "Lower the music 8 dB (decibels, the unit of loudness) while I talk." A timeline is just a picture of instructions like these.

3

A renderer

The program that follows the instructions and writes a brand-new video file. FFmpeg, Remotion and HyperFrames are renderers.

The important consequence

Everything you want must be written as an instruction. Compare these two requests:

NOT AN INSTRUCTION

"Make the title pop."

Pop how? Bigger? Brighter? Faster? Claude has to guess, and a guess is a default.

AN INSTRUCTION

"Scale the headline from 90% to 100% over 0.25 s with a fast ease-out, and play a short whoosh on the same frame."

Measurable, repeatable, checkable.

Translating feelings into instructions

YOU FEEL	ASK FOR THIS INSTEAD
"It's boring"	A new visual event every 0.8 to 1.5 s; a slow zoom in (100% to 104%) on every shot that doesn't move
"It looks cheap"	One accent colour per frame; one caption zone; two fonts maximum; ease-out instead of bounce
"It feels slow"	Remove pauses over 0.25 s (keep a sliver of sound at each edge); first useful sentence by 1.5 s

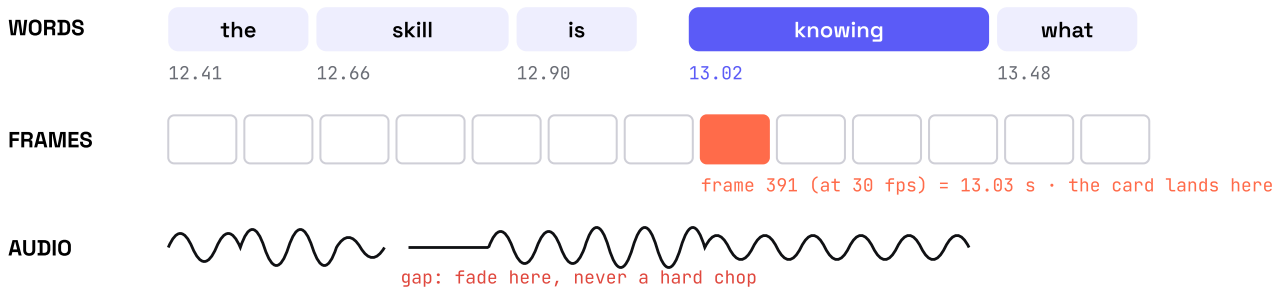
TIP

Can't put a feeling into numbers? Ask: `The opening feels weak. Give me three measurable fixes.` Then pick one.

01.4 · YOUR TIMELINE IS DATA

Three clocks run *every video*

A traditional editor shows the timeline as a picture. Claude needs the same information as data it can read. That data is made of three clocks, and almost every editing problem is one of them going wrong.



THE TRANSCRIPT ISN'T NOTES. IT'S MACHINE-READABLE TIMING DATA.

Clock 1 · Words

Each spoken word gets a start and end time. This drives cuts, captions and every graphic that appears "on the word".

Clock 2 · Frames

Video is a stack of still images. fps (frames per second) says how many per second. At 30 fps, frame 90 is exactly 3.00 s. At 60 fps, it's 1.50 s.

Clock 3 · Audio

Cuts need tiny fades or a little padding, or you hear clicks and lose the ends of words.

Frame maths you'll use every week

You never do this maths by hand: Claude does. Knowing it lets you read a plan and spot a wrong number. ms means milliseconds, thousandths of a second.

FRAME RATE	ONE FRAME LASTS	1 SECOND =	A 0.25 S MOVE =	WHEN TO USE
30 fps	33.3 ms	30 frames	7.5 frames (round to 8)	Talking heads, most Reels
60 fps	16.7 ms	60 frames	15 frames	Fast motion graphics, 3D, smooth camera moves

CHECK

Pick one sentence in a raw clip. Its transcript words should match what you hear, and the start time should line up when you play the clip at that timestamp. If they don't match, fix the transcription **before** touching captions or graphics, because everything else is built on top of it.

01.5 · THE THREE ROLES

You direct. Claude operates. *Machines render.*

Every Claude video setup has the same three roles underneath. Keeping them separate makes the work fast and repeatable.

ROLE	WHO	THEIR JOB	NEVER THEIR JOB
Director	You	Set the goal, write the brief, approve the plan, watch the real video, send timestamped notes, decide what becomes a rule	Editing code by hand, guessing at commands
Operator	Claude Code	Plan, write code and commands, run tools, check its own work with evidence, ask when something is unclear	Deciding taste, inventing facts or assets, changing approved work silently
Machines	FFmpeg, Whisper, Remotion, HyperFrames, voice tools	Do the actual cutting, transcribing, animating, mixing and rendering	Knowing what you meant

Why you shouldn't touch the code

If you catch yourself editing a file by hand, stop and describe the change instead. Three reasons:

1. It's faster. Claude edits in seconds and keeps related parts consistent.
2. Claude learns your preference for the rest of the session, and you can save it as a rule.
3. Hand edits are invisible to the system. Next time, the same mistake comes back.

What a good director actually does

- [ ] Decides before Claude builds (brief and plan)
- [ ] Asks for proof, not promises
- [ ] Gives notes with a time, an element, a problem and a fix
- [ ] Locks what's approved so revisions stay small
- [ ] Turns repeated corrections into saved rules

```
DIRECTOR · BRIEF.MD
DESTINATION Instagram Reel
LENGTH 32 s, max 38 s
CUT fillers, retakes
KEEP pause before "$9"
LOOK /reel-explainer v1.3
```

```
OPERATOR · PLAN.MD
beat 1 · "cheap" · card-in
beat 2 · "three" · counter
cuts: 14 · saves 9.6 s
sfx: 11 events mapped
open: logo file missing?
```

```
MACHINES · QA.MD
PASS 1080x1920 · 30 fps
PASS -14.1 LUFS · -1.2 dBTP
PASS captions in zone
PASS no clipped words
CHECK 0:21 by ear
```

THE TAKEAWAY

The cards show what each role writes: your order, Claude's plan, the machines' test results (LUFS and dBTP are loudness readings, Part 13). Every look starts with clear orders. Your edge is the quality of yours.

01.6 · GREAT AT, BREAKS AT, NEEDS YOU

Know what to hand over, *and what to keep*

This workflow is very good at repeatable, rule-based work. It's weak where the job needs human eyes and ears. Plan around both.

Hand it over

- Removing fillers, false starts, retakes and dead air
- Word-timed captions with a consistent style
- Title cards, counters, comparisons, charts, UI callouts
- The same brand rules across 50 videos
- Variants: new hooks, colours or first frames
- Technical checks: size, codec (compression format), loudness, safe zones
- 3D scenes and animation styles from a written spec

Keep for yourself

- Which take is funnier, warmer or more convincing
- Whether a pause feels intentional
- Whether a motion feels premium or annoying
- Whether a joke or claim is right for the audience
- Whether an asset is licensed or confidential
- The final watch, with sound, on a phone

Use another tool

- Rotoscoping (cutting a person out frame by frame)
- Visual effects that need hand-drawn art on every frame
- High-end colour grading (film-style colour work) on calibrated screens
- Rescuing badly damaged audio
- Live multi-camera switching

The honest rule about checking

Claude can look at still frames and measure files, but it can't watch motion or hear the way you do. So it gives you **evidence**, and you judge.

WHAT CLAUDE CAN PROVE	WHAT ONLY YOU CAN JUDGE	HOW TO COMBINE THEM
Resolution, fps, codec, duration	Whether the pacing feels right	Claude lists every cut; you watch the cuts it flags
Captions inside the zone on key frames	Whether the captions are pleasant to read in motion	Claude exports the trigger frames; you watch the full Reel once
Overall loudness (LUFS) and the loudest peak	Whether the music fights the voice	Claude measures; you listen on a phone speaker
Each sound effect (SFX) has a visual cause	Whether the sound design is too busy	Claude maps sounds; you mute or keep each one

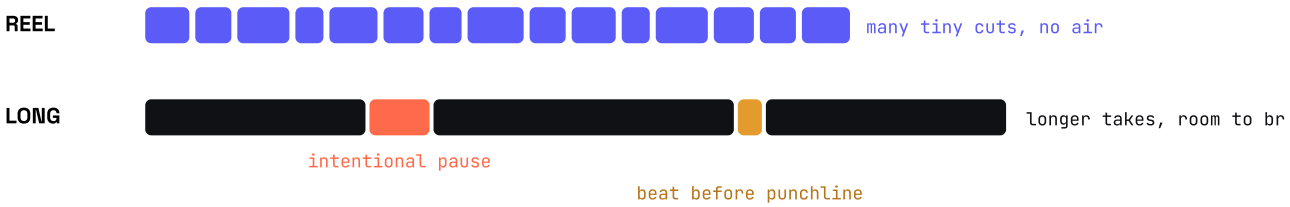
WARNING

Never accept "I checked it" as proof. Ask: *what* did you check, and where is the evidence? A contact sheet, a frame grab, an ffprobe report (ffprobe reads what's inside a video file) or a loudness number. If none exists, the check didn't happen.

01.7 · SHORT FORM VS LONG FORM

Short form is cut by rule. *Long form is cut by intention.*

One raw clip can become a fast Reel or a calm YouTube video. Each needs different instructions. Mixing them up makes AI edits feel "off".



SAME FOOTAGE, TWO JOBS

Reels (this book's focus)

- Cut almost everything that isn't a word: breaths, long pauses, restarts
- Something new on screen at least every 1.5 s
- A good style skill makes most decisions for you

Long form (and Reels over ~90 s)

- Some pauses are on purpose: a beat before a punchline, a stretched word, a stutter for effect
- Claude can't know which pauses are deliberate unless you say so
- Visual changes come at section changes, not every second
- You give more structure, and fewer rules

For long form, give Claude three extra things

01 **A script with sections**
So it knows which lines are intentional, where the story turns and what must stay word for word.

02 **What to keep**
"Keep pauses longer than 0.8 s in the story section; they're dramatic. Keep my laugh after 'it failed'."

03 **Structure by timestamp**
"1:00 title card. 2:50 lower third (a name strip near the bottom) with my name. 5:00 slow zoom in. 7:40 chapter card 'The fix'."

PROMPT · LONG FORM

```
This is a long-form video, not a Reel. Read script.md first: its [SECTION] markers are the
structure.
Cut only accidental fillers, false starts and repeated takes. KEEP every pause longer than 0.8 s
inside [STORY] sections.
Structure: 1:00 title card "[TITLE]"; 2:50 lower third "[NAME] · [ROLE]".
Write plan.md listing every proposed cut with source in/out and the words removed. Do not render
until I approve.
```

01.8 · THE 30-DAY PLAN

Zero to a repeatable studio *in 30 days*

Each block ends with something you can show, not just something you read. Follow it in order and you'll finish with a system that makes your 20th Reel as controlled as your first.

DAYS	DELIVERABLE	PARTS
1-2	Working studio, VERSIONS.md, and a 5-second test render that plays on your phone	02
3-5	One clip transcribed, cleaned and checked against the audio	03
6-7	A style sheet (your look, written down) built from 3 to 5 reference Reels	04
8-10	Your first style skill, tested on two different clips	05
11-13	Three complete briefs and approved plans	06
14-17	One Reel rebuilt from a Frame Spec instead of vague feedback	07
18-21	One finished Reel in each style: talking head, faceless, 3D	08, 09, 10
22-23	Four animation styles tested, each 5 seconds before any full render	11, 12
24-25	An SFX map (which sound plays on which moment) and a voice/music mix at the house loudness target	13
26-27	Three hook variants and a complete script system	14
28-29	QA checklist passed, verified export, a saved template and skill	15, 16
30	Publish a test, read retention (how long people kept watching), add one new rule	17

THE DAY-30 MASTERY TEST

Take a new 30 to 60 second talking-head clip and produce a branded Reel without dragging a single timeline clip, using:

- [ ] one written brief
- [ ] one plan approval
- [ ] one full render
- [ ] one batch of timestamped notes
- [ ] one verified export

DON'T CHASE A STOPWATCH

Render time depends on your computer, the source length and the graphics. The real win is fewer unclear decisions and fewer revision loops.

A DAILY RHYTHM (1 HOUR)

15 min read one chapter · **30 min** do its boxes on your own footage · **10 min** run the Check · **5 min** add one line to `LEARNINGS.md`.

PART 01 · WHAT YOU HAVE NOW

Your foundation is *in place*

You haven't installed anything yet, on purpose. You now see video as files, instructions and a renderer, driven by clear direction.

You can now

- [ ] Explain what Claude Code does and doesn't do with video
- [ ] Turn a feeling ("it's boring") into a measurable instruction
- [ ] Read a timeline as words, frames and audio
- [ ] Separate your job (director) from Claude's job (operator)
- [ ] Decide which tasks to hand over and which to keep
- [ ] Follow a 30-day plan with a clear test at the end

Before you move on

DO THIS NOW

Create a note called `my-first-direction.md` . Paste the three sentences you wrote in 01.1 about a Reel you love. Under them, rewrite each sentence as a measurable instruction, using the table in 01.3 as a guide.

CHECK

Could someone who has never seen the Reel build its first two seconds from your note alone, without asking you a question? If yes, you're ready for Part 02.

The five ideas to carry through the book

1 **Describe, don't drag.** Every change is an instruction.

2 **Words are the clock.** Timestamps decide when things happen.

3 **Plan before render.** A table is cheaper to fix than a video.

4 **Proof, not promises.** Every check leaves evidence.

5 **Save what works.** Approved rules become skills.

→ **Next:** Part 02 builds the studio on your computer.

Words you now own

Brief	Your written order: destination, length, cuts, look, placement	**Render**	The machine writing a new video file
Plan	Claude's proposal of every cut, graphic and sound, before rendering	**fps**	Frames per second: 30 or 60 in this book

That was Part 01 of 17.

The full book goes on from setup to finished Reels.

311 pages . 17 Parts . 43 Claude Code skills

The full book and the Skill Bundle are on the page where you opened this sample.