

TIP Feed this guide to your AI assistant to get started. Drag the PDF into Claude Code or ChatGPT's Codex mode and say **"Set up my Mac with this guide."** It reads the guide, checks what's missing and downloads the tools you need.

DOZA · FREE GUIDE FOR EDITORS

Automate your **video podcast** editing with AI

For Final Cut Pro and DaVinci Resolve editors. Turn a multicam podcast or interview into a finished episode (the cut, transcript, selects, assembly, brand graphics, social clips and deliverables) with Doza Cuts, Doza Assist and Claude or ChatGPT.



Doza Cuts



Doza Assist

doza.ai

Contents

You don't need to write code for any of this. Your AI assistant does the reading, picking and scripting; Doza Cuts and Doza Assist do the cutting and transcribing on your Mac; you keep every creative decision.

01	How it works	The idea, and what you get	3
02	What you need	For Final Cut Pro and for Resolve	5
03	Connect everything	Claude Code or ChatGPT, side by side	6
04	Level 1: your first AI-assisted episode	An afternoon, no code	7
05	Level 2: build your show kit	Once per show, then every episode is automatic	9
06	The full episode run	Eleven stages, Final Cut and Resolve	10
07	Prompts that work	Copy, paste, change the names	11
08	Rules, checks and troubleshooting	Keeping it safe and fixing what breaks	12
09	Checklist and templates	House rules file and a skill to copy	13
→	Doza Assist and Doza Cuts	The two apps this guide is built on	15

LEVEL 1 · AN AFTERNOON

Connect your AI to Doza Cuts and Doza Assist. Cut a multicam to whoever is talking, transcribe it, and have the AI find your cold opens, montage moments and social clips as selects in your edit.

LEVEL 2 · ONCE PER SHOW

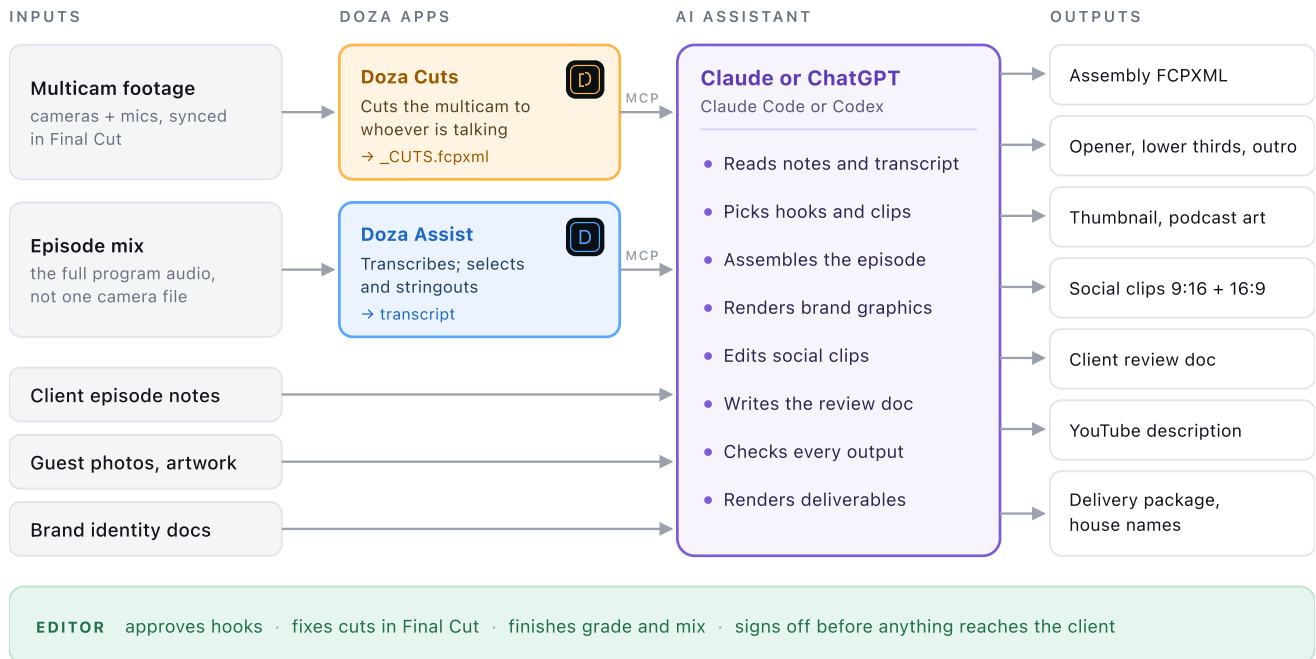
Turn your client's brand guide and one approved episode into a kit. From then on the AI builds every episode's assembly, graphics, social clips, client review doc and deliverables for you to approve.



New to Doza? Doza Assist has a free trial: every feature on the first two minutes of each file.
doza.ai/download

How it works

Most of the post on a video podcast or interview repeats every episode: cutting cameras, transcribing, finding the moments, building the open and close, cutting social clips, rendering deliverables. Doza's apps do the cutting and transcribing on your Mac. An AI assistant connects to them, reads what they produce, and does the rest of the repeated work. You approve.



Doza Cuts and Doza Assist run on your Mac and talk to your AI assistant through MCP, an open standard for connecting apps to AI. Footage and audio never leave your drives; the assistant only receives text, timings and file paths.

Three kinds of work

APPS & SCRIPTS

Exact work

Switching cameras, transcribing, frames, timecode, loudness, file names, renders. Measured, so it's the same every time.

AI ASSISTANT

Reading and judgment

Reading transcripts and notes, picking moments, writing copy, following brand rules, and writing the scripts it needs.

YOU

Taste and approval

Approve the picks, adjust the cut, finish the grade and mix, and sign off before anything goes out.

What one episode can produce

- A multicam cut, switched to whoever is talking.
- A transcript with speaker names, flagged wherever someone asked for an edit.
- Cold-open options and montage moments, each with a reason.
- An assembled episode with your open and close placed to the frame.
- Opener, lower thirds, outro credits and thumbnail from your templates.
- Social clips in 9:16 and 16:9.
- A transcript review document your client marks up, read back into timecodes.
- A YouTube description with chapters.
- Every deliverable rendered, named and filed.

WHY IT WORKS

AI is good at reading and deciding, and bad at counting frames. So every number that matters (a cut point, a graphic's position, a loudness, an A/V offset) is measured by an app or a script, and every judgment the AI makes comes with a written reason you can accept or reject.

What you need

One Apple-silicon Mac runs everything. Pick your editor and your AI assistant; the rest is the same.

ITEM	FINAL CUT PRO EDITORS	DAVINCI RESOLVE EDITORS
Mac	Apple silicon (M1 or later), macOS 15 or later. 16 GB of memory minimum; 32 GB or more recommended for large multicam libraries.	
Your editor	Final Cut Pro	DaVinci Resolve Studio 21.1 or later to let the AI work inside Resolve (its built-in MCP server is a Studio feature). The free edition works too, by importing files.
 Doza Cuts	Cuts the multicam to whoever is talking: live multicam cuts for Final Cut, a flat timeline for Resolve. Free trial for three exports, then \$79 once at launch (\$99 after): doza.ai/cuts	
 Doza Assist	Transcribes on your Mac, finds selects, builds stringouts. Free trial, then \$199 once, or Studio at \$29/month or \$299/year: doza.ai/download	
An AI assistant	Claude (a paid plan that includes Claude Code) or ChatGPT (a paid plan that includes Codex), using the Mac desktop app. Pick one; this guide shows both.	
Command-line tools	Python 3, FFmpeg and whisper.cpp. Your assistant installs them (Chapter 3). Level 2 adds Node.js and Remotion for brand graphics.	

WHAT LEAVES YOUR MAC, AND WHAT DOESN'T

Doza Cuts and Doza Assist talk to your assistant over the Mac's own loopback connection. Video and audio never leave your machine. The assistant sees transcript text and selects from the Doza Assist projects you switch on, and XML paths, angle names and cut statistics from Doza Cuts. Claude and ChatGPT are cloud services, so that text is processed by Anthropic or OpenAI under their terms. If you work for clients, check that your agreements allow it before you switch a project on.

FINAL CUT OR RESOLVE: WHICH IS FASTER?

Both work end to end. Resolve Studio 21.1 goes further, because its built-in MCP server lets the assistant work *inside* Resolve: organising media, adding markers, setting up renders and running the render queue. Those steps become a sentence instead of a script. Final Cut editors get the same results through files the assistant writes and you import.

Connect everything

About half an hour, once per Mac. Do the column for the assistant you use. Doza Cuts and Doza Assist each add themselves to your assistant with one button.

STEP	● CLAUDE CODE	● CHATGPT (CODEX)
1 • Open a folder	Install the Claude desktop app, sign in, open the Code tab and choose a folder for your show.	Install ChatGPT for Mac, sign in, open the ChatGPT menu at the top and choose Codex , then Local , and pick a folder for your show.
2 • Doza Cuts	Open Doza Cuts › Help › AI assistants › switch on AI assistant access › press Add to Claude Desktop or Add to ChatGPT . Keep Doza Cuts open while you work.	
3 • Doza Assist	On a project page (import any interview first), open the gear menu › AI assistant access . Add the connector for your assistant (Claude Desktop or ChatGPT), then switch on access for each project the AI may read. Keep Doza Assist open.	
4 • Restart and test	Ask: <i>"List my Doza projects, and what did Doza Cuts cut today?"</i>	Quit and reopen ChatGPT, type <code>/mcp</code> to see both connectors, then ask the same question.
5 • Tools	Ask: <i>"Check what's installed, then install Homebrew, Python 3, FFmpeg and whisper.cpp if they're missing, and download the ggml-medium.en Whisper model to ~/models. Tell me what you'll install before you do it."</i> If Homebrew isn't installed, its installer needs your Mac password, so the assistant gives you one command to paste into Terminal.	
6 • House rules	<code>CLAUDE.md</code> in the show folder. Claude reads it at the start of every session.	<code>AGENTS.md</code> in the show folder. ChatGPT reads it at the start of every session.
7 • Permissions	Claude asks before running commands. Ask it to add safe, repeated ones to <code>.claude/settings.json</code> .	Choose Ask for approval under the composer: ChatGPT works inside the show folder and asks before anything else.

Prefer the terminal? Run `claude` or `codex` in the show folder; the Codex CLI shares ChatGPT's connectors. Claude Code:

```
claude mcp add doza-cuts --scope user -- \
  "/Applications/Doza Cuts.app/Contents/Resources/python/bin/python3" \
  -B "/Applications/Doza Cuts.app/Contents/Resources/mcp/server.py"

claude mcp add doza-assist --scope user -- \
  "/Applications/Doza Assist.app/Contents/Resources/python/bin/python3" \
  -B "/Applications/Doza Assist.app/Contents/Resources/mcp/server.py"
```

Resolve Studio: let the assistant work inside Resolve

1 Turn on Resolve's AI assistant connection

In DaVinci Resolve Studio 21.1 or later, choose **File** › **Setup AI Assistants**. Resolve finds the assistants installed on your Mac (Claude Desktop, Claude Code, ChatGPT's Codex) and configures its MCP server for them. Restart your assistant afterwards. The AI assistants chapter of the Resolve 21.1 manual lists what it can do.

2 Allow local scripting

In **Preferences** › **System** › **General**, set **External scripting using** to **Local**. Doza Cuts needs this to send a finished cut straight into your open project. Then test on a duplicate project: *"What's in the timeline that's open in Resolve?"* The assistant acts with your permissions, so stay on duplicates until you trust a prompt. On free Resolve, skip this section; everything else works through file imports.

Level 1: your first AI-assisted episode

No code, no templates. You'll get a cut episode, a transcript, and AI-picked moments sitting in your timeline. Use any finished multicam podcast or interview to try it.

01 Export the multicam

YOU

Final Cut Pro Build the multicam clip with every camera and mic, put it in a project, then **File > Export XML**.

Resolve Build the multicam clip, put it on a timeline, then **File > Export > Timeline** and choose FCPXML.

02 Cut it to whoever is talking

DOZA CUTS

AI

ASK YOUR ASSISTANT

"Inspect ~/Desktop/Episode 12.fcpxml with Doza Cuts: who is on which camera and mic, and is any media offline? Then cut it in the interview style for Final Cut." (For Resolve: "...for Resolve.")

Doza Cuts listens to every mic and watches every camera, works out who is on which, and writes `Episode 12_CUTS.fcpxml` next to your export. It never overwrites anything. **Final Cut:** ask "open it in Final Cut" and every cut arrives as a live multicam cut you can change in the Angle Viewer. **Resolve:** press **Send to Resolve** in Doza Cuts (Studio, with local scripting on), or use **File > Import > Timeline**.

The Doza Cuts free trial covers three exports with every feature, with no account or card: doza.ai/cuts.

03 Transcribe it

DOZA ASSIST

Import the episode into Doza Assist and transcribe it. Doza Assist reads Final Cut XML, including multicam, or you can bring in the episode's audio. Transcription runs on your Mac. When it finishes, switch on **AI assistant access** for the project. On the free trial, transcripts stop at two minutes: try it on a short clip, or unlock the full length at doza.ai/buy.

04 Let the AI find the moments

AI

YOU

ASK YOUR ASSISTANT

"Read the whole transcript of Episode 12. Give me five cold opens under 15 seconds, six short montage moments, and three social clips of 30 to 90 seconds. For each: timecodes, what's said, and why it works. Then mark them as selects in Doza Assist, with the reason as the note."

The selects appear in your Doza Assist project within seconds, labelled "AI assistant", next to any you made yourself. Keep the ones you like; delete the rest.

05 Put them in your timeline

DOZA ASSIST

YOU

ASK YOUR ASSISTANT

"Build a selects stringout from Episode 12, in reviewer order, with one-second handles."

Doza Assist writes a stringout timeline to `~/Documents/Doza Assist/Exports`. **Final Cut:** File > Import > XML. **Resolve:** File > Import > Timeline. Every select arrives as a clip, with reviewer markers.

06 Resolve Studio: markers and renders from the chat

AI

RESOLVE MCP

ASK YOUR ASSISTANT

"In Resolve, add colour-coded markers to the episode timeline: blue for the cold opens, green for the social clips. Then add render jobs for a ProRes 422 HQ master, an H.264 review copy and a 9:16 version for social, and start the render queue."

This is where Resolve saves the most time: markers, render presets and the render queue are handled through Resolve's own MCP server. Check the first marker lands where you expect; transcript times match the timeline when the transcript was made from the same audio. Final Cut editors export as usual, or let the Level 2 kit render for them.



Cut a multicam in minutes. Doza Cuts cut a 67-minute, five-camera interview in two minutes on an M-series Mac. Free trial: doza.ai/cuts

Level 2: build your show kit

Do this once per show. Afterwards, a new episode is one request: the assistant builds the draft, you approve it. Test every piece on one approved pilot episode first.

1 Turn the brand guide into files

Put the client's brand guide, fonts and logos in `brand/source/` and ask: *"Compile brand/tokens.css, brand/fonts and brand/rules.md. Quote the guide for every rule, and list what it doesn't settle as questions for the client."* Check every value yourself; a wrong colour repeats in every render.

2 Make the graphics templates

Have the assistant build each piece (opener, lower thirds, transition, outro, thumbnail, social) as a **Remotion** template, which renders video from code, fed by one data file of names, titles, credits and photos. Ask the assistant to install Node.js and Remotion first. Remotion is free for individuals and small teams; larger companies need a company licence (remotion.dev/license). Masters work in both editors: ProRes 4444 with alpha for overlays, ProRes 422 HQ for full-frame. Ask for a contact sheet with every render.

3 Measure your approved pilot

Export the client-approved pilot's timeline and ask the assistant to measure the open and close frame by frame: every transition, graphic, montage, music edit and lower third, with its lane, scale and gain, counted from **C, the end of the cold open**, plus the dialogue loudness. The result, `template.json`, is the one file that holds your design.

4 Write the house rules

Your `CLAUDE.md` or `AGENTS.md` holds the non-negotiables: what's read-only, where outputs go, how names are spelled, which pieces the client approved. Start from the template in Chapter 9.

5 Have the assistant build the pipeline, one stage at a time

Give it the ground rules first, then build the stages in Chapter 6's order, testing each on the pilot.

ASK YOUR ASSISTANT, IN A NEW SESSION

"We're building a per-episode post pipeline for this show in `pipeline/` and `episodes/`. Plain Python scripts, one job each. Never modify source media or Doza Cuts' `_CUTS` files; only write inside `pipeline/` and `episodes/`. Version every output (`_v001`, `_v002`...) and never overwrite. Hash every source and stop if it changes. All timing math in whole frames. Keep a README that says who runs each stage. Plan first and show me the plan."

6 Prove it on the pilot, then save it as a skill

Rebuild the pilot from its raw cut and compare it with the approved final. When it matches frame for frame, every later episode starts from something the client already approved. Then ask the assistant to turn the run into a **skill** (a saved instruction file, see Chapter 9) so anyone on your team can start an episode with one request.

The full episode run

Each episode gets its own folder and a manifest: one file recording its sources, speakers, approvals and the status of every stage. Stages can be re-run and never overwrite. Where Resolve Studio's MCP server can do a stage directly, the Resolve column says so.

#	STAGE	WHO	FINAL CUT PRO	DAVINCI RESOLVE STUDIO
1	Cut the multicam	CUTS YOU	Doza Cuts > open in Final Cut; fix any cut you'd make differently	Doza Cuts for Resolve > Send to Resolve
2	Start the episode	SCRIPT	Folder and manifest; picks the newest cut and records its hash	
3	Read the client's notes	AI	Names and titles exactly as written; credits for the outro; a list of anything still to come ("TK")	
4	Transcript	ASSIST SCRIPT	Pulled from Doza Assist; checked against the audio for any offset; speaker names mapped; lines where someone asked for an edit flagged	
5	Hooks	AI YOU	Five cold opens, 6–10 montage moments and the interview's in and out, each with a reason; snapped to real silences; you approve	
6	Graphics	AI SCRIPT	Opener, lower thirds, outro credits and thumbnail rendered from your templates and checked on contact sheets	
7	Assembly	SCRIPT	Versioned FCPXML from the recipe, validated, imported with File > Import > XML	Built from the flat Resolve cut and imported with File > Import > Timeline; check the pilot import against your approved final first
8	Clip pulls	AI ASSIST	2–3 minute standalone moments and 30–90 second social picks, as Doza Assist selects and a stringout	
9	Social clips	AI SCRIPT	Tightened at real silences, faces kept in frame, captions re-timed; rendered 9:16 and 16:9 from your templates	Same edit; Resolve can also reframe the timeline to 9:16 and render it from the chat
10	Client review	SCRIPT CLIENT	A Word transcript, one numbered line per turn, timed to the draft; the client highlights cuts; their markup comes back as timecodes	
11	Finish and deliver	YOU AI	You finish; the AI renders and files deliverables with scripts	You finish; the AI sets up the render queue with your presets and runs it

START AN EPISODE WITH ONE REQUEST

"Start a new episode for Guest Name from the newest Doza Cuts cut and the Doza Assist project 'Guest Name'. Run every stage up to the hooks, then stop and show me the proposals." After you approve: "Carry on through the review document, then tell me where everything is and list every TK."

Prompts that work

These work the same in Claude Code and ChatGPT. Change the names and paths to yours.

WHEN YOU WANT TO...	SAY
Check the connections	"List my Doza projects, and what did Doza Cuts cut this week?"
Cut a multicam	"Cut ~/Desktop/Episode 12.fcpxml with Doza Cuts in the conversational style, no punch-ins, and open it in Final Cut."
Find a moment	"Search the Episode 12 transcript for every time the guest talks about their first job, with timecodes."
Get story options	"Give me five cold opens under 15 seconds, each with why it works and the risk."
Mark selects	"Mark the three strongest 30–90 second social moments as selects, with a title in the note."
Build a stringout	"Build a selects stringout from Episode 12 with one-second handles."
Resolve: organise	"In Resolve, rename every clip on V1 as Interview_A_001, 002, 003, and add a red marker wherever someone asked for an edit."
Resolve: deliver	"Render the current timeline as ProRes 422 HQ and as an H.264 review copy, plus a 9:16 version for social."
Fix a title everywhere	"The client changed the guest's title to '...'. Re-render everything that shows it and rebuild the assembly."
Read the client's markup	"Here's the returned review doc. Turn it into a cut list with timecodes and flag anything ambiguous."
Deliver	"Show me the file list and names first, then copy them to the delivery folder once I say yes."

Habits that make it reliable

One episode per session

Name the episode in the first message. Long sessions covering several episodes mix up context.

Plan before anything new

"Plan this; don't change any files yet." Approve the plan, then let it work.

Ask for proof

"Show me the contact sheet." "Run the validator."
Done means checked.

Correct it once

When you correct something, ask the assistant to add the rule to your house rules file so it sticks.

Rules, checks and troubleshooting

The rules that keep the pipeline safe, and fixes for every problem we hit on real episodes.

Rules the pipeline follows

- **Sources are read-only** and hash-checked. Doza's apps never edit or delete your media either.
- **Nothing is overwritten.** Every output gets the next version number.
- **Names and titles exactly as the client writes them.** Missing text is "TK", never a guess.
- **Proof, not promises:** a validation report for every assembly, contact sheets for every render, loudness and A/V sync measured on every social clip.
- **You sign off** before anything is sent or uploaded, every time.
- **Client data:** switch on Doza Assist access only where your agreements allow AI processing of transcript text.

Troubleshooting

PROBLEM	WHY	FIX
The assistant can't see Doza Cuts or a Doza Assist project	The app is closed, or AI assistant access is off	Open the app and switch access on. In ChatGPT, restart after adding a connector and check <code>/mcp</code> .
Transcript is only two minutes long	Doza Assist trial limit	Unlock full length with a licence at doza.ai/buy ; nothing needs redoing.
Transcript runs a few seconds early or late	It was made from a camera file with its own sync offset	Transcribe the episode mix. Have the assistant check the offset against the audio.
Wrong speaker names	Speaker labels differ between projects	Tell the assistant which label is which person; it records the map.
Doza Cuts put a face on the wrong camera	Unusual framing or a shared mic	Press Swap cameras in Doza Cuts and cut again; nothing is re-analysed.
Send to Resolve does nothing	Free Resolve, or local scripting is off	Resolve Studio > Preferences > System > General > External scripting using: Local. Or use File > Import > Timeline.
The assistant can't reach Resolve	Resolve isn't set up for that assistant	Open Resolve Studio 21.1+, run File > Setup AI Assistants again, restart the assistant.
Brand colour looks wrong after rendering	The render isn't tagged Rec. 709	Ask the assistant to render and tag every master as Rec. 709.

Checklist and templates

Tick through Level 1 in an afternoon. Copy the house rules below into `CLAUDE.md` or `AGENTS.md` when you start Level 2.

Level 1

- ☐ Doza Assist installed (free trial at doza.ai/download)
- ☐ Doza Cuts installed (free trial at doza.ai/cuts)
- ☐ Claude Code or ChatGPT (Codex) open on a show folder
- ☐ Both Doza connectors added and tested
- ☐ Resolve Studio only: File > Setup AI Assistants done; scripting set to Local
- ☐ A multicam cut with Doza Cuts and opened in your editor
- ☐ Episode transcribed; AI access switched on
- ☐ AI picks marked as selects and a stringout in your timeline

Level 2

- ☐ Brand guide compiled into tokens, fonts and rules; checked
- ☐ Graphics templates built and proofed
- ☐ Pilot measured into `template.json`
- ☐ House rules file written
- ☐ Pipeline built in order and the pilot rebuilt to match
- ☐ The run saved as a skill
- ☐ First new episode drafted, approved and delivered

```
# <Show> post-production    (save as CLAUDE.md or AGENTS.md)

## House rules
- Camera originals, the mix, the client's files and Doza Cuts' _CUTS files are read-only.
- Scripts write only inside pipeline/ and episodes/. Outputs are versioned, never overwritten.
- Names and titles exactly as the client writes them. Anything missing is "TK".
- Read brand/rules.md before making any graphic. Approved pieces are in brand/locked.md.
- After every render, look at its contact sheet. After every assembly, read the validation report.
- Never send, upload or share anything with the client without my OK.

## Start of every session
Read episodes/<slug>/manifest.json and say which stage is next.

## Where things are
- Footage: /Volumes/<SSD>/<Show>/          - Deliveries: <Dropbox>/<Show>/Episodes/<Guest>/
```

A skill: one request per episode

Save as `.claude/skills/new-episode/SKILL.md` (Claude Code) or `.agents/skills/new-episode/SKILL.md` (ChatGPT). Then say “start a new episode for Guest Name”, or in Claude Code type `/new-episode`.

```
---
name: new-episode
description: Build a draft episode of <Show> from a Doza Cuts cut and a Doza Assist
  project, through assembly, clip pulls and the client review doc.
---
1. Find the guest's Doza Assist project and newest Doza Cuts cut. If either is missing, stop.
2. Start the episode, read the client's notes, pull and check the transcript.
3. Propose hooks with reasons. STOP and wait for my approval.
4. Render the graphics, build and validate the assembly, pull clips, make the review doc.
5. Report where every file is, every TK and every warning.
```

Keep going

Tutorials

Eleven short videos, start to finish.

doza.ai/tutorials

How-to guides

One job at a time, with the exact clicks: stringouts, paper edits and more.

doza.ai/how-to

Doza for Final Cut Pro

Multicam transcription and sequences that open in Final Cut.

doza.ai/for/final-cut-pro

Doza for DaVinci Resolve

Transcription and selects timelines for Resolve, free edition included.

doza.ai/for/davinci-resolve

User guide

Every Doza Assist feature, from installation to export.

doza.ai/guide

Discord

Editors using Doza, and the maker. Bring your questions.

[Linked from doza.ai](https://doza.ai)

Built by an editor, for **editors**.



Doza Assist

Local AI for interview editors. Transcribe, find the story and get selects into your edit, all on your Mac.

- Transcription on your Mac in about 100 languages, with speaker ID
- AI analysis, Story Brief and Quote Sheets
- Selects, stringouts and paper edits to Final Cut Pro, Premiere Pro and DaVinci Resolve
- The AI assistant connector used in this guide
- Studio adds Share: send a link, get selects back

\$199 once

or Studio **\$29**/month or **\$299**/year

Free trial: every feature on the first two minutes of each file. 30-day money back.

doza.ai/download



Doza Cuts

Your multicam, cut in minutes. Doza Cuts switches a whole podcast or interview to whoever is talking.

- Listens to every mic and watches every camera; nothing to map
- Live multicam cuts in Final Cut Pro, a flat timeline in Resolve
- Conversational, Fast, Interview or Custom styles
- A 67-minute, five-camera interview cut in two minutes
- Runs on your Mac; Claude or ChatGPT can drive it

\$79 once at launch

then **\$99** once

Free trial: every feature for three exports. No account, no card. Two Macs.

doza.ai/cuts

doza.ai

Tutorials, the user guide and the free trial