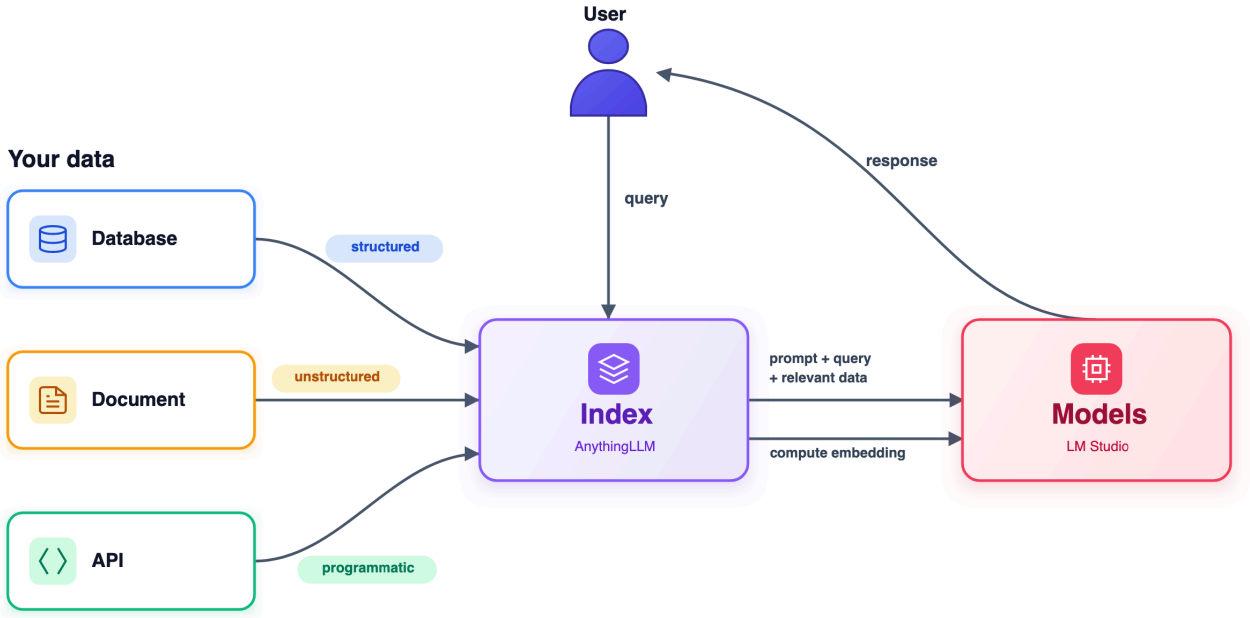
Local AI Cheat Sheet

Quick reference for the hands-on part. Keep this open while you work.

RAG pipeline at a glance

What you are building: your documents get embedded into a searchable index; your question retrieves the relevant chunks; the model answers from those chunks, with citations.



1. Decoding a model name

Today's models — three sizes, pick by your laptop's free RAM:

```
Qwen3-4B-Q4_K_M
└──┘ └┘ └────┘
family size quantization
 Qwen3  4B   Q4_K_M
```

- **Family + version**: who built it, which generation (`Qwen3` , `Llama-3.1` , `LFM2.5` , ...)
- **Size**: parameter count. `4B` = 4 billion weights. Bigger = smarter, slower, more RAM
- **Tuning**: `Instruct` / `Chat` = follows instructions (what you want). Qwen3 models are instruction-tuned by default (no suffix); a `-Thinking` suffix = reasoning-tuned. On other families, no suffix usually means a raw base model, skip those
- **Quantization**: `Q4_K_M` = compressed to ~4 bits per weight (see section 3)

Your three chat models: `Qwen3-1.7B` (fastest, lightest) · `Qwen3-4B` (balanced) · `Qwen3-14B` (best answers, needs ~10 GB free RAM). All share one embedder, `bge-m3` .

Another example: `Llama-3.1-8B-Instruct-Q4_K_M` = Meta's Llama, version 3.1, 8 billion params, instruction-tuned, 4-bit quant.

2. Reading a Hugging Face model card

Check these before downloading:

What	Where	Why
`Instruct` / `Chat` in name	title	you want instruction-tuned, not base
License	top of card	check before client work (MIT, Apache 2.0, Llama license, ...)
Context window	card text	max input + output size (32k tokens ≈ 50 pages)
File size	Files tab	must fit your RAM (see rule below)
GGUF format	Files tab	the format LM Studio runs

3. Quantization: which file to pick

Quant	Bits/weight	Quality	Size (8B model)
`FP16`	16	original	~16 GB
`Q8_0`	8	near-original	~8 GB
`Q5_K_M`	~5	very good	~5.5 GB
`Q4_K_M`	~4	good, **the laptop sweet spot**	~4.5 GB
`Q2_K`	~2	noticeably degraded	~3 GB

`_S` / `_M` / `_L` = small/medium/large: how many critical layers keep higher precision.

Will it fit? Rule of thumb: model file size + ~2 GB overhead ≤ your free RAM.

4. Model families

Family	What it does	Today's example
Chat / Instruct	answers questions, follows instructions	`Qwen3-1.7B` · `Qwen3-4B` · `Qwen3-14B`
Base	raw next-token engine, not instruction-tuned	(skip)
Embeddings	turns text into searchable "fingerprints", powers RAG	`bge-m3`
Vision (VLM)	also reads images and scanned PDFs	
Reasoning	"thinks" before answering; slower, better at multi-step	today's `Qwen3` can think

5. Prompt patterns

System prompt vs user prompt: system prompt = standing instructions applied to every question in the workspace. User prompt = your actual question. System wins on conflict.

Workspace system prompt (paste in AnythingLLM → workspace gear → Chat Settings):

```
You are a professional AI assistant for document question answering.

Use the provided context documents as your primary source. Quote, paraphrase, and combine
evidence from one or more retrieved chunks to answer the user's question.

Instructions:
- Give clear, concise, professional answers.
- Prefer structured formatting using headings and bullet points.
- Answer every part of the question you can from the context. If a specific sub-question has no
support, say so for that part only, then continue.
- Never invent facts, legal clauses, financial numbers, or references not present in the context.
- Cite the source document by file name for each fact you state. (AnythingLLM also lists the
sources it used under each answer.)

Entity attribution & reasoning:
- First identify the entity (investor / company / person) named in the question. Report ONLY
findings whose own source names that entity. Never attribute a finding from one investor's
document to a different investor, even when other investors dominate the retrieved context.
- If the question names an entity whose findings are not in the context, say so for that entity.
Do not substitute the nearest available entity.
- "Clean / no findings" is a valid answer. If a document says an entity has "no findings" or
"file complete", report that entity as clean. Do not conclude "no entity is clean" just because
others have findings.
- When asked whether a declaration matches a finding, state the declaration, state the finding,
then say "matches" or "contradicts". A declaration of legitimate funds alongside a sanctions or
money-laundering finding is a contradiction.
- When comparing across entities, give one block per entity, each drawn only from that entity's
own document. Never let one entity's verdict carry over to another.
- Every finding you list must be backed by a source. Do not output an uncited finding, and do not
invent finding-codes (e.g. KYC-GFO-001) that are not in the documents.

If none of the context is relevant, reply: "The provided documents do not contain enough
information to answer this question."
```

Per-question patterns:

Pattern	How	When
Role	"You are a fund accountant..."	sets tone + domain
Explicit format	"Answer as a table with columns X, Y"	structured output
Few-shot	give 2-5 worked examples	format and tone matter
Chain-of-thought	"Think step by step"	numbers, logic, comparisons
Grounding	"Answer only from the provided context, cite file + page, say 'not found' if absent"	every RAG question
Decompose	split compound questions into one question per item	model misses parts of multi-part asks

6. Ready-to-use questions for the case study

Warm-up (single-hop, reliable on small models):

- What is Giovanni Family Office's commitment in the Investor Register?
- When is the next capital call due and how much?
- Which LP's KYC file is clean? *(Archer Pension Fund)*

Findings (single-hop):

- Does any investor have a sanctions match? *(Giovanni Family Office)*
- Which LP has adverse media? *(Jessie Family Office)*
- Is there a whistleblower report, and who does it concern? *(Meowth, fingers Giovanni)*

Cross-document:

- Which LPs have open KYC findings, and what are they? (compound: often misses one pack; re-ask per LP to recover)
- Summarize the 2024 audit findings and which LPs they relate to.

Stress tests:

- Does the fund invest in German real estate? *(correct refusal expected)*
- Compare the side-letter terms across all LPs. *(3 docs at once: watch for missed LPs, decompose per LP)*

7. Glossary

Term	One-liner
Token	small text fragment (~¾ of a word); the unit models read and write
Weights	the millions of numbers inside a model; set during training
Inference	using a trained model to generate answers
Training	building the model: guess, compare, adjust weights, repeat billions of times
Context window	max amount of text the model can consider at once
Quantization	storing weights less precisely to shrink the model file
Embedding	a numeric "fingerprint" of a text's meaning; similar meaning = similar fingerprint
RAG	Retrieval Augmented Generation: find relevant document chunks, hand them to the model, answer grounded + cited
Hallucination	fluent, confident, wrong: the model predicts plausible text, it does not look things up
System prompt	standing instructions applied to every question in a workspace
GGUF	the model file format LM Studio runs

8. Links & references

Your stack:

Tool	Link	What
LM Studio	https://lmstudio.ai	runs models locally, serves the API
LM Studio docs	https://lmstudio.ai/docs	setup, server, model settings
AnythingLLM	https://anythingllm.com	RAG workspace on top of LM Studio
AnythingLLM docs	https://docs.anythingllm.com	workspaces, embedding, chat settings

Finding models:

Resource	Link	What
Hugging Face	https://huggingface.co/models	the model library; filter by "GGUF"
lmstudio-community	https://huggingface.co/lmstudio-community	curated GGUF builds, safe default source
Today's chat models	https://huggingface.co/lmstudio-community (Qwen3-1.7B / 4B / 14B GGUF)	what's on your laptop
Today's embedder	https://huggingface.co/ggml-org/bge-m3-Q8_0-GGUF	powers the document search
LM Arena	https://lmarena.ai	community model leaderboard, compare quality
OpenRouter	https://openrouter.ai	one API for many cloud models; good for browsing what exists and comparing prices. Cloud, not local: don't send confidential client or LP data

Editor:

Tool	Link	What
VS Code	https://code.visualstudio.com	free editor, handy for viewing markdown and config files

Appendix: quantization math (for the curious)

Per block of weights:

```
scale  = max_abs_weight_in_block / max_int_value
q_int  = round(weight / scale)          ← stored on disk
ŵ      = q_int × scale                  ← reconstructed at inference
```

Fewer bits → fewer integer levels → coarser rounding → bigger gap to original.

Worked example, one weight, block max-abs = `1.0`, original `w = 0.38472915`:

```
FP16   →  stores  0.38472915  directly                        (2 bytes)

INT8   →  scale = 1/127  ≈ 0.007874
          q = round(0.38472915 / 0.007874) = 49               (1 byte)
          ŵ = 49 × 0.007874 ≈ 0.3858

INT4   →  scale = 1/7    ≈ 0.1429
          q = round(0.38472915 / 0.1429)  = 3                 (½ byte)
          ŵ = 3 × 0.1429 ≈ 0.4286
```

`_S` / `_M` / `_L` mixed precision: a few critical tensors (`attention.wv` , `ffn.w2`) get upgraded to Q6_K, rest stays at base quant.