AetherGraph – Quick Notes
=========================

Core ideas
----------

- Python-first agentic runtime.
- Graphs are explicit: TaskGraph + Interpreter.
- graph_fn lets you write Python functions that can be turned into graphs on demand.

RAG integration
---------------

- context.rag() gives you a RAGFacade, wired to:
  - artifacts store
  - embedding client
  - LLM client
  - vector index backend

Typical workflow
----------------

1. Ingest documents into a named corpus.
2. Chunk + embed with your configured embed model.
3. Search or retrieve chunks for a query.
4. Call rag.answer() to generate a response with citations.

Example corpus IDs
------------------

- "demo-corpus-files"
- "demo-corpus-inline"
- "research-notes"

Useful labels
-------------

- format: txt, md, pdf
- topic: aethergraph, rag, scheduler
- source: local_files, inline