Retrieval-Augmented Large Language Model for AI Programming Question Answering

Project Description (Goals & Limitations):

Goal:

- Build a small-scale AI assistant that answers questions specifically about AI programming (Python, PyTorch, TensorFlow, ML concepts).
- Use retrieval-augmented generation (RAG) to ensure answers are grounded in a small set of technical documents (PDFs or text files).

Limitations: include a small dataset, limited model size, and no real-time web search.

What Will I Achieve (Learn & Produce) and Prior Knowledge:

I will learn core data science and AI engineering concepts including text preprocessing, embeddings, vector databases, prompt design, and evaluation of language model outputs.
The final product will be a working prototype of a document-based Q&A system.
My prior knowledge includes basic Python, machine learning concepts, and introductory experience with data analysis and AI models.

Why I Want to Do This Project:

This project is a strong foundation for a career in data science and AI engineering because RAG systems are widely used in industry for chatbots, enterprise search, and knowledge assistants. It is highly extensible, practical, and in strong market demand, making it an ideal first project.

Final Deliverable and Potential Users:

The final deliverable will be a small demo application (simple web interface) that answers user questions from programming PDFs/docs.
Potential users: Students learning AI programming, educators, developers need quick references, small teams managing AI codebases.

Project Boundaries (30-Hour Scope):

Included: (5 – 10) technical PDFs (Python, PyTorch, TensorFlow tutorials), Pre-trained LLM + basic vector store and Simple interface and qualitative evaluation.
Excluded: large-scale deployment, model training from scratch, real-time web search, advanced UI/UX, and enterprise-level security features