Documentation: Key Concepts in Generative AI and RAG

# 1. LangChain

LangChain is a framework for building applications using large language models (LLMs). It provides components and abstractions for managing chains of LLM calls, incorporating memory, and integrating with external tools and data sources.  
  
Example:  
from langchain.llms import OpenAI  
llm = OpenAI()  
response = llm("What is LangChain?")

# 2. RAG (Retrieval-Augmented Generation)

RAG enhances LLM capabilities by integrating a retrieval mechanism. It fetches relevant context documents for a query, which are then passed to the LLM to improve accuracy and relevance.  
  
Workflow:  
User query -> Document retriever -> LLM with retrieved documents -> Final response

# 3. LLMs (Large Language Models)

LLMs are deep neural networks trained on large-scale text datasets. They can perform various NLP tasks such as summarization, translation, and question answering.  
  
Examples: OpenAI GPT, Google's PaLM, Meta’s LLaMA.

# 4. FAISS (Facebook AI Similarity Search)

FAISS is a library for efficient similarity search and clustering of dense vectors. It is commonly used in vector databases and semantic search.  
  
Example:  
import faiss  
index = faiss.IndexFlatL2(dimension)  
index.add(vectors)  
D, I = index.search(query\_vector, k)

# 5. VectorDB (Vector Database)

Vector databases are optimized for storing and retrieving high-dimensional vector embeddings, often used for semantic similarity search.  
  
Examples: Pinecone, Weaviate, Chroma.

# 6. Generative AI

Generative AI refers to AI systems that create new content, such as text, images, or audio. These models are trained on existing data and can produce creative outputs.  
  
Examples: GPT for text, DALL·E for images.

# 7. GANs (Generative Adversarial Networks)

GANs consist of two neural networks, a generator and a discriminator, that compete. The generator creates samples, and the discriminator evaluates their realism.  
  
Used for: Image synthesis, super-resolution, style transfer.