LangChain Roadmap

1. Fundamentals

Start with the core building blocks of LangChain:

- **What is LangChain** → Understand its purpose as an LLM application framework.
- **LangChain Components** → Learn about modular parts used to build pipelines.
- **Models** → Work with LLMs (open-source & closed-source) and their APIs.
- **Prompts** → Techniques for creating effective instructions for LLMs.
- **Parsing Output** → Extracting structured and usable results from model outputs.
- **Runnables & LCEL (LangChain Expression Language)** → Writing workflows with reusable components.
- **Chains** → Linking multiple steps/models/tools together for complex tasks.
- **Memory** → Adding short-term and long-term memory for conversational agents.

---

2. RAG (Retrieval-Augmented Generation)

Move into knowledge integration and retrieval-based AI:

- **Document Loaders** → Ingest documents from different formats (PDF, Word, Web).
- **Text Splitters** → Break documents into chunks for better context retrieval.
- **Embeddings** → Represent text as vectors for similarity search.
- **Vector Stores** → Store embeddings (e.g., Pinecone, FAISS, Chroma).
- **Retrievers** → Mechanisms to fetch the most relevant data for queries.
- **Building a RAG Application** → Combine LLM + retriever for Q&A and knowledge-augmented responses.

---

3. Agents

Learn how to make AI systems autonomous and interactive:

- **Tools & Toolkits** → Extend LLMs with external APIs, calculators, search engines, etc.
- **Tool Calling** → Let LLMs decide when and how to use tools.

- **Building an AI Agent** → Create agents that can reason, plan, and act across multiple steps.

✅ **Progression Tip**:

- **Start with Fundamentals** → Build simple LLM-powered apps.
- **Advance to RAG** → Make apps knowledge-aware.
- **Move to Agents** → Add autonomy and complex decision-making.