

Introduction to LangChain:



Introduction to LangChain



LangChain is a powerful framework for building applications that combine LLMs with external data, tools, and workflows. It simplifies the creation of intelligent agents by managing prompts, memory, retrieval, and chains in production-ready pipelines.

01

What is Framework



A framework is a **ready-made structure or set of tools** that helps you build something faster and in an organized way.

Example: Building a house

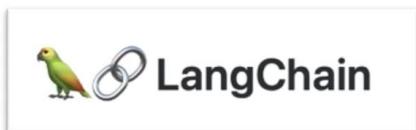
A framework is like a **pre-built scaffolding + toolkit** that helps you build faster (instead of starting from zero with bricks and cement).



02

What is LangChain

LangChain is a **framework** that helps developers build applications using **Large Language Models (LLMs)** like Chat GPT, Claude, and others



A screenshot of the LangChain website. The header features the LangChain logo, navigation links for Products, Resources, Docs, Company, and Pricing, and buttons for "Get a demo" and "Sign up". The main visual is a dark teal background with a large, stylized white wave graphic. The text "The platform for reliable agents." is prominently displayed in white. Below it, a subtext reads "Tools for every step of the agent development lifecycle - built to unlock powerful AI in production." At the bottom, there are two buttons: "Request a demo" and "See the docs".

03

Why do we need LangChain

Imagine:

You want to build a **chatbot** that can read PDF files.
Or an **AI app** that can **search Google + summarize + send an email**.

Doing this **from scratch** is **hard** because:

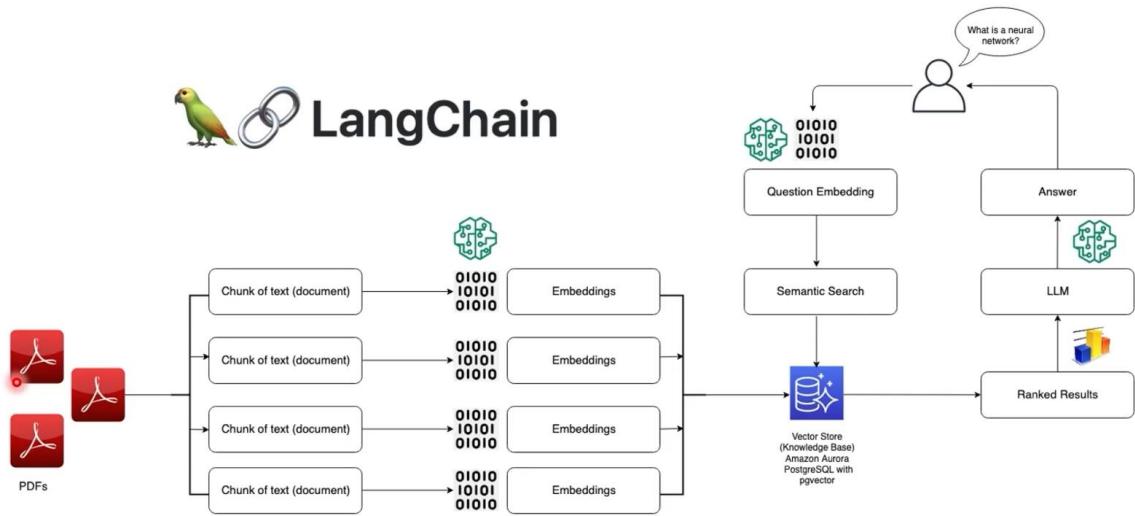
- You need to **connect many tools**,
- Handle **memory & context**,
- And **manage LLM conversations**.

LangChain makes all this easier!



04

Why do we need LangChain



Benefits

Benefit	Description	Use Case Example
Tool Integration	Connects LLMs with external tools like APIs, web search, calculators, etc.	Ask "What's the weather in Delhi?" → LLM calls weather API
Memory Support	Maintains short-term and long-term memory for continuity in conversation	Personal AI tutor remembers past questions
Document Q&A	Enables LLMs to read and answer questions from PDFs, Word files, or databases	Upload resume → Ask "What skills are missing?"
Modular Design	Reusable components (chains) for faster, structured development	Chain: Prompt → Tool → Output pipeline
Agent Framework	Creates autonomous agents that can reason and choose actions/tools dynamically	Agent plans: Search → Extract → Respond

06

Benefits

Benefit	Description	Use Case Example
Ecosystem & Integrations	Supports OpenAI, HuggingFace, Pinecone, Chroma, SQL, and many others	Build vector search over internal documents
Rapid Prototyping	Build apps quickly using pre-built chains and templates	Demo chatbot in under 10 minutes
Open-source & Community	Active community with examples, docs, and plugins	Contribute, customize, or learn from existing examples
Real-world Applications	Applicable in customer support, legal, education, reporting, and more	Automated business report summarizer

07

What can you build



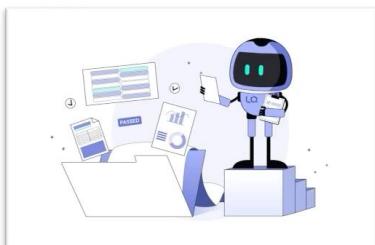
AI Chatbots



AI Agents



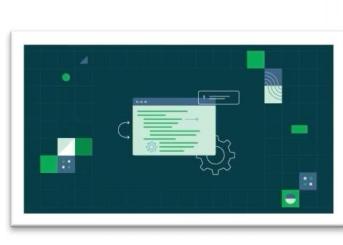
Personal AI Tutors



Document QA Systems



Voice-Activated Assistants



Custom LLM Apps

08

Alternatives



09