Memory:

Memory

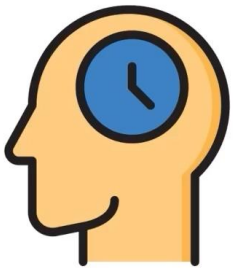
Memory in LangChain allows LLMs to remember previous interactions.This helps maintain context across multiple turns in a conversation, enabling more natural and coherent dialogue over time.

What is Memory in LangChain?

Definition

Memory in LangChain allows a language model to **remember previous interactions** across a conversation or a multi-step process.

Just like humans, memory helps an AI assistant carry forward context from earlier steps or messages — enabling more coherent and context-aware responses.



Why Memory is Important

- ❑ Maintains **conversation continuity**
- ❑ Enables **multi-turn conversations**
- ❑ Stores **intermediate outputs** in chains and agents
- ❑ Useful for personal assistants, chatbots, tutoring systems, and agents

Types of Memory in LangChain

Memory Type	Description	Use Case
ConversationBufferMemory	Stores the entire chat history as a string	Simple chat apps, summarization
ConversationSummaryMemory	Summarizes past messages instead of storing all of them	Long-running chats, memory-efficient bots
ConversationBufferWindowMemory	Only keeps the last *k* messages in memory	Lightweight contextual memory
VectorStoreRetrieverMemory	Stores memory in a vector DB, retrieves based on similarity	Long-term memory, semantic recall
CombinedMemory	Allows combining different memory strategies	Advanced AI agents

How Memory Fits in a LangChain Workflow

GUVI Skill Up. Level Up | HCL

User Input
↓
Memory (optional: loads prior context)
↓
Prompt (template + input + memory)
↓
LLM Response
↓
Memory (optional: updates with new input/response)