# Agents

An "agent" is an automated reasoning and decision engine. It takes in a user input/query and can make internal decisions for executing that query in order to return the correct result. The key agent components can include, but are not limited to:

- Breaking down a complex question into smaller ones
- Choosing an external Tool to use + coming up with parameters for calling the Tool
- Planning out a set of tasks
- Storing previously completed tasks in a memory module

Research developments in LLMs (e.g. [ChatGPT Plugins](https://openai.com/blog/chatgpt-plugins)), LLM research ([ReAct](https://arxiv.org/abs/2210.03629), [Toolformer](https://arxiv.org/abs/2302.04761)) and LLM tooling ([LangChain](https://python.langchain.com/en/latest/modules/agents.html), [Semantic Kernel](https://github.com/microsoft/semantic-kernel)) have popularized the concept of agents.

## Agents + LlamaIndex

LlamaIndex provides some amazing tools to manage and interact with your data within your LLM application. And it can be a core tool that you use while building an agent-based app.

- On one hand, some components within LlamaIndex are "agent-like" - these make automated decisions to help a particular use case over your data.
- On the other hand, LlamaIndex can be used as a core Tool within another agent framework.

In general, LlamaIndex components offer more explicit, constrained behavior for more specific use cases. Agent frameworks such as ReAct (implemented in LangChain) offer agents that are more unconstrained + capable of general reasoning.

There are tradeoffs for using both - less-capable LLMs typically do better with more constraints. Take a look at [our blog post on this](https://medium.com/llamaindex-blog/dumber-llm-agents-need-more-constraints-and-better-tools-17a524c59e12) for a more information + a detailed analysis.

## Learn more

Our Putting It All Together section has [more on agents](/understanding/putting_it_all_together/agents.md)