

Master LangChain Tools Deep Dive

A complete system-level guide to LangChain Tools: built-in tools, API tools, search tools, and how to design safe, production-grade custom tools for agents.

1. What Is a Tool in LangChain?

A Tool is an interface that allows an LLM agent to perform actions outside of text generation. The LLM decides which tool to call, but the system executes it and returns structured results.

2. Why Tools Exist

LLMs cannot browse the web, query databases, or call APIs reliably on their own. Tools provide controlled, deterministic access to external systems.

3. Built-in LangChain Tool Categories

- Search Tools (Wikipedia, Tavily, YouTube, SerpAPI)
- Retrieval Tools (VectorStore as Tool)
- Computation Tools (Math, Python REPL)
- Database Tools (SQL, Pandas)
- File & Document Tools
- API Wrapper Tools

4. YouTubeSearchTool (Why It Exists)

YouTubeSearchTool allows an agent to search video metadata (titles, descriptions). It is useful for tutorials, product demos, and real-time knowledge discovery.

```
from langchain.tools import YouTubeSearchTool

tool = YouTubeSearchTool()
results = tool.run("LangChain RAG tutorial")
```

5. Common Search Tools

- WikipediaSearchTool – public factual knowledge
- TavilySearchTool – web search optimized for LLMs
- SerpAPI – Google search results
- ArxivSearchTool – research papers

6. Creating Custom Tools

Custom tools allow you to expose business logic, APIs, or computations to an agent. They are the most powerful part of agentic systems.

```
from langchain.tools import tool

@tool
def get_order_status(order_id: str) -> str:
    return f"Order {order_id} is shipped"
```

7. Tool Design Best Practices

- Make tools deterministic and side-effect free when possible
- Use narrow, well-defined inputs
- Return structured outputs
- Never let tools execute arbitrary code
- Validate all inputs

8. Tools in RAG vs Agentic RAG

In classic RAG, retrieval happens automatically. In Agentic RAG, retrieval itself becomes a tool the agent can choose to call.

9. Hands-on Mini Labs

- Lab 1: Add WikipediaSearchTool to a chatbot
- Lab 2: Add YouTubeSearchTool for tutorials
- Lab 3: Expose a SQL query as a tool
- Lab 4: Convert a retriever into a tool
- Lab 5: Build a full agent with custom business tools

Final Takeaway: Tools turn LLMs from text generators into system controllers.