

**LANGGRAPH 101**

## LangGraph Crash Course: From Basics to...

by Harish Neel | AI

Playlist • 47 videos • 118,674 views

Play all

- 1 LangGraph Crash Course #1 - Introduction  
Harish Neel | AI • 23K views • 4 months ago
- 2 LangGraph Crash Course #2 - Levels of Autonomy in LLM applications  
Harish Neel | AI • 10K views • 4 months ago
- 3 LangGraph Crash Course #3 - Agents & Tools - Intro  
Harish Neel | AI • 7.1K views • 4 months ago
- 4 LangGraph Crash Course #4 - Agents & Tools - Implementation  
Harish Neel | AI • 9.6K views • 4 months ago
- 5 LangGraph Crash Course #5 - Drawbacks of React Agents  
Harish Neel | AI • 6.1K views • 4 months ago
- 6 LangGraph Crash Course #6 - Reflection Agent - Introduction  
Harish Neel | AI • 5.5K views • 4 months ago

### LangGraph Crash Course #1 - Introduction

Harish Neel | AI  
7.67K subscribers

Subscribe

440 Share Save

23,882 views Feb 10, 2025 #python #artificialintelligence #langgraph

In this beginner-friendly playlist, I'll show you how to use LangGraph to build powerful AI-agents from scratch

1. Levels of Autonomy in LLM applications (Code -> LLM Call -> Chain -> Router -> Agent)
2. Understanding Agents & Tools
3. Building Agents & Tools from Scratch
4. Building Agents From pre-defined LangChain classes

5. Graph Structure
  - Direct Acyclic Graph (DAG) vs Cyclic Graph
6. What is LangGraph?

7. Why LangGraph is required?
8. Creating LangGraph from scratch
9. Creating a LangGraph using in-built classes (Reflection, Reflexion ager

10. Key concepts and terms in LangGraph
  - Graph, state, nodes, edges, visualisation, checkpoints, breakpoints configuration



## 11. Creating a Chatbot with LangGraph

## 12. Common Agentic Patterns

- Human-in-the-loop
- ReAct Agent, and many more.

## 13. Multi-agent systems using LangGraph

## 14. RAGs with LangGraph: CRAG vs ARAG vs self-RAG

## 15. Persistence



## 16. LangGraph ecosystem

- LangGraph Studio
- LangGraph Cloud API, etc

## 17. Agents in Production

## Pre-requisites

1. You need to have Python 3.8 or higher installed

2. LangChain (as LangGraph builds on top of LangChain)

## LangGraph Crash Course #2 - Levels of Autonomy in LLM applications



Harish Neel | AI  
7.67K subscribers

Subscribe

209

1

Share

Clip

Save

...

10,454 views Feb 10, 2025 #python #artificialintelligence #langgraph

In this beginner-friendly playlist, I'll show you how to use LangGraph to build powerful AI-agents from scratch

## Levels of Autonomy in LLM applications

### 1. Code

Code has zero autonomy and is 100% deterministic

We all know that everything is hard-coded and it is not even really a cognitive architecture.

### Disadvantage:

The problem? You'd need to write rules for every possible scenario - making it impossible to handle real-world complexity.

## 2. LLM call

A single LLM call means your app basically does one main thing - you give it an input, it processes it, and gives you back an output.

Think of chatbots that just take your message and respond, or apps that translate text.

This was a huge leap from hard-coded rules, even though it's still pretty simple and is only in the 2nd stage of autonomy

## 2. LLM call



Example User Input: You are an expert LinkedIn post writer. Write me a post on "AI Agents Taking over Content Creation"



## 2. LLM call

A single LLM call means your app basically does one main thing - you give it an input, it processes it, and gives you back an output.

Think of chatbots that just take your message and respond, or apps that translate text.

This was a huge leap from hard-coded rules, even though it's still pretty simple and is only in the 2nd stage of autonomy

### Disadvantage:

Trying to get everything done in one shot often leads to confused or mixed-up responses - just like how a single person can't be an expert at everything.

## 3. Chains

Think of chains like having multiple specialists instead of one generalist. Instead of asking one AI to do everything, we break it down into steps where each AI is really good at one thing.

Imagine a customer service chatbot: The first AI reads your complaint and figures out exactly what product you're talking about

The second AI finds the right solution from the company's help docs, and the third AI turns that solution into a friendly response.

Each step is simple, but together they create a much smarter system than a single LLM call could.

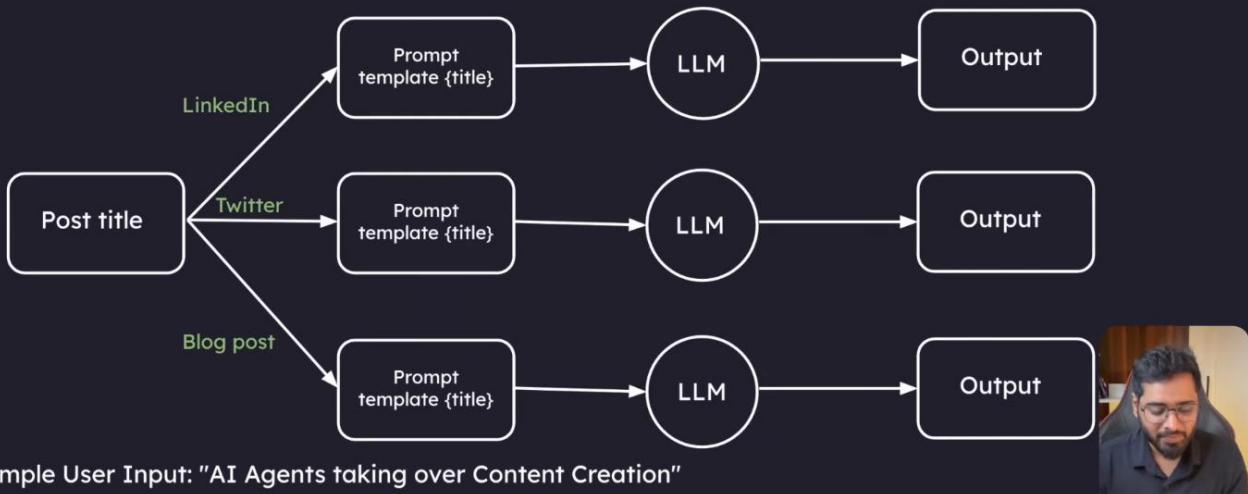
### 3. Chains (contd.)

This is where we first started seeing AI apps that could handle more complex tasks - not just by being smarter, but by breaking big problems into smaller, manageable pieces.

#### Disadvantage:

The downside? These fixed sequences are like a rigid assembly line - they always follow the same steps defined by the human.

### 3. Chains (contd.)



Example User Input: "AI Agents taking over Content Creation"

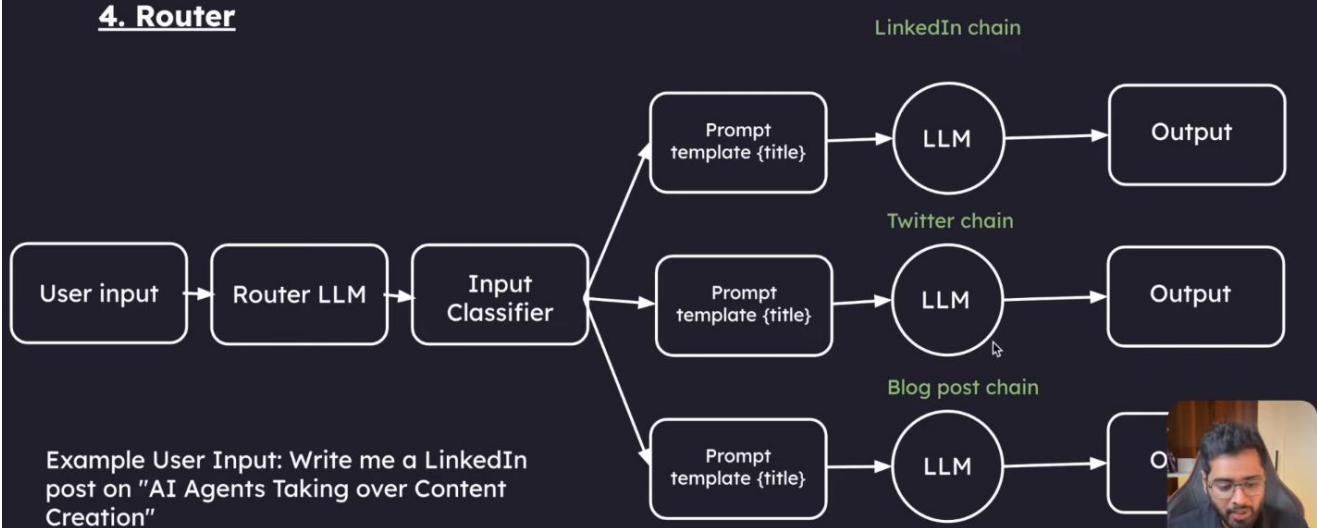


### 4. Router

Now this is where it gets interesting - routers are like smart traffic cops for your AI. Instead of having a fixed path like in chains, the AI itself decides what steps to take next.

Imagine a personal assistant bot: when you ask it something, it first figures out if you need help with scheduling, research, or calculations, then routes your request to the right tool or chain for the job.

### 4. Router



Example User Input: Write me a LinkedIn post on "AI Agents Taking over Content Creation"

## **4. Router**

Now this is where it gets interesting - routers are like smart traffic cops for your AI. Instead of having a fixed path like in chains, the AI itself decides what steps to take next.

Imagine a personal assistant bot: when you ask it something, it first figures out if you need help with scheduling, research, or calculations, then routes your request to the right tool or chain for the job.

### **Disadvantage:**

While it can choose different paths, it still can't remember previous conversations or learn from mistakes.



## **5. State Machine (Agent)**

This is combining the previous level (router) but with loops.

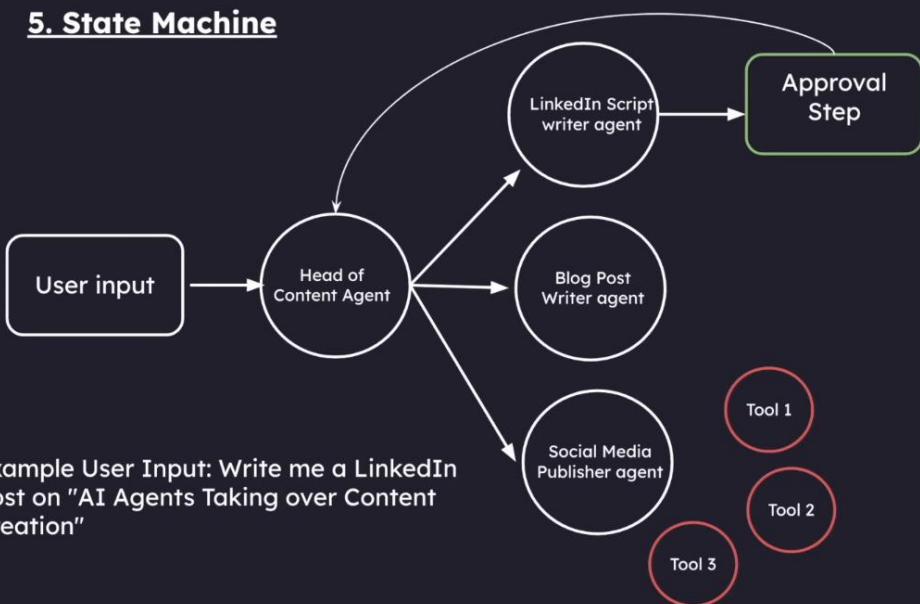
Agent ~= control flow controlled by an LLM

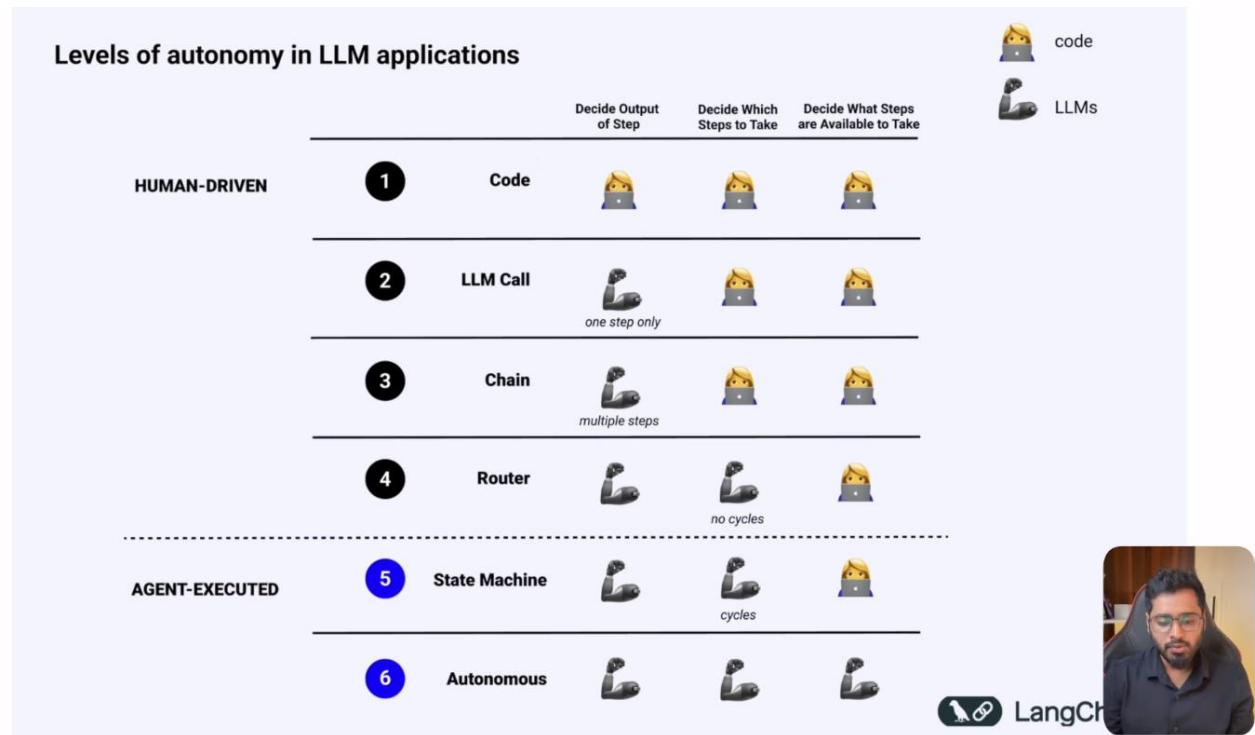
This involves features like:

1. Ability to have human-in-loop, ask for approval before moving on
2. Multi-agent systems
3. Advanced memory management
4. Go back in history and explore better alternate paths
5. Adaptive Learning

And many more, and THIS is where LangGraph comes into the picture

## **5. State Machine**





## Chain/Router vs Agent

A Chain or even a router is one directional. Hence, it is not an agent

Whereas in a state machine, we can go back in the chain, have cycles and the flow is controlled by the LLM, hence it is called an Agent

### LangGraph Crash Course #3 - Agents & Tools - Intro



Harish Neel | AI  
7.67K subscribers

Subscribe

112    Share    Clip    Save    ...

7,122 views Feb 10, 2025 #python #artificialintelligence #langgraph

In this beginner-friendly playlist, I'll show you how to use LangGraph to build powerful AI-agents from scratch

## Understanding AI Agents

## AI Agents & Tools

Think of Agents as the "problem-solvers" of the AI world. Agents are capable of thinking on their own.

In other words, it's AI that can make autonomous decisions.

In the case of Chains and Router, they follow our specific instruction.

But with agents, they actually take it a step further. They can decide for themselves what steps to take on their own.

## AI Agents & Tools

### What are tools then?

Tools are specific functions that Agents can use to complete tasks

Just like a chef's kitchen tools (knife for cutting, oven for baking, blender for mixing), tools are the special abilities we give to AI - like giving it a calculator tool, or a search engine tool, or a calendar tool

## AI Agents & Tools (Re-Act Agent Pattern)

This is one of the best known patterns in AI today to build agents. It stands for Reasoning + Acting

This is basically a concept that mimics how human beings think.

### ReACT pattern

**Think:** LLM first thinks about the user prompt/problem

**Action:** LLM decides if it can answer by itself or if it should use a tool

**Action Input:** LLM provides the input argument for the tool

*Here, langChain executes the tool and returns the output to the LLM*

**Observe:** LLM observe the result of the tool

⋮

**Final Answer:** "This is your final answer"



# ReACT pattern

Re-Act Agent



## Let's Jump Into The Code

1. Build a Re-act Agent Using LangChain ↗
2. What are its drawbacks and where does LangGraph come into the picture?

### LangGraph Crash Course #4 - Agents & Tools - Implementation

 Harish Neel | AI  
7.67K subscribers

Subscribe

214



Share

Clip

Save



9,695 views Feb 10, 2025 #python #artificialintelligence #langgraph

In this beginner-friendly playlist, I'll show you how to use LangGraph to build powerful AI-agents from scratch

EXPLORER

...  
LANGGRAPH  
  venv  
    bin  
      activate  
      activate.csh  
      activate.fish  
      Activate.ps1  
      pip  
      pip3  
      pip3.12  
      python  
      python3  
      python3.12  
    > include  
    > lib  
  pyvenv.cfg

PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

(base) harishb@Harishb-MacBook-Air LangGraph % python3 -m venv venv  
(base) harishb@Harishb-MacBook-Air LangGraph % source venv/bin/activate  
(venv) (base) harishb@Harishb-MacBook-Air LangGraph %

EXPLORER

...  
LANGGRAPH  
  1\_Introduction  
    react\_agent\_basic.py  
  venv

1\_Introduction > react\_agent\_basic.py

Google langchain google chat model

All Images Videos Shopping Forums Web News More Tools

**LangChain**  
https://python.langchain.com ... > Chat models

**ChatGoogleGenerativeAI**

13 May 2024 — This doc will help you get started with Google AI chat models. For detailed documentation of all ChatGoogleGenerativeAI features and ...

Google langchain openai chat model

All Videos Images News Shopping Forums Web More Tools

**LangChain**  
https://python.langchain.com ... > Chat models

**ChatOpenAI**

13 May 2024 — This notebook provides a quick overview for getting started with OpenAI chat models. For detailed documentation of all ChatOpenAI features and configurations ...

ChatGoogleGenerativeAI | langchain openai chat model

python.langchain.com/docs/integrations/chat/google\_generative\_ai/

**LangChain** Integrations API Reference More v0.3 Search

Providers

- Anthropic
- AWS
- Google
- Hugging Face
- Microsoft
- OpenAI
- More

Components

- Chat models
- Retrievers

To learn more about the key features of the two APIs see the [Google docs](#).

## Overview

### Integration details

Class	Package	Local	Serializable	JS support	Package downloads	Package latest
ChatGoogleGenerativeAI	langchain-google-genai	X	beta	✓	989k/month	v2.0.9

Overview

- Integration details
- Model features
- Setup
- Credentials
- Installation
- Instantiation
- Invocation
- Chaining
- Safety Settings
- API reference
- Related

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python

```
(base) harishb@Harishs-MacBook-Air LangGraph % python3 -m venv venv
(base) harishb@Harishs-MacBook-Air LangGraph % pip install langchain langchain_community langchain-google-genai python-dotenv
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS zsh

```
pyeyeballs, yarl, typing-inspect, SQLAlchemy, rsa, requests, pydantic-core, pyasn1-modules, proto-plus, marshmallow, jsonpatch, httplib2, httpcore, googleapis-common-protos, anyio, aiosignal, requests-toolbelt, pydantic, httpx, grpcio-status, google-auth, dataclasses-json, aiohttp, pydantic-settings, langsmith, google-auth-httplib2, google-api-core, langchain-core, google-api-python-client, langchain-text-splitters, google-ai-generative-language, langchain, google-generativeai, langchain-google-genai, langchain_community
Successfully installed PyYAML-6.0.2 SQLAlchemy-2.0.38 aiohappyeyeballs-2.4.6 aiohttp-3.11.12 aiosignal-2.0.0 anyio-4.8.0 attrs-25.1.0 cachetools-5.5.1 certifi-2025.1.31 charset-normalizer-3.4.1 dataclasses-1.2.0 frozenlist-1.5.0 google-auth-generative-language-0.6.15 google-api-core-2.24.1 google-api-python-protobuf-2.38.0 google-auth-httplib2-0.2.0 google-generativeai-0.8.4 googleapis-common-protos-1.6.0 grpcio-status-1.70.0 h11-0.14.0 httpcore-1.0.7 httplib2-0.22.0 httpx-0.28.1 https-sse-0.4.0 idna-3.10 onpointer-3.0.0 langchain-0.3.18 langchain-core-0.3.34 langchain-google-genai-2.0.9 langchain-text-splitter, langchain_community-0.3.17 langsmith-0.3.7 marshmallow-3.26.1 multidict-6.1.0 mypy-extensions-1.0.0 numpy-2.10.5 packaging-24.2 propcache-0.2.1 proto-plus-1.26.0 protobuf-5.29.3 pyasn1-0.6.1 pyasn1-modules-0.4.1 pydantic-core-2.27.2 pydantic-settings-2.7.1 pyparsing-3.2.1 python-dotenv-1.0.1 requests-2.32.3 requests-aiohttp-4.9 sniffio-1.3.1 tenacity-9.0.0 tqdm-4.67.1 typing-extensions-4.12.2 typing-inspect-0.9.0 uritemplate-3.0.0 yarl-1.18.3 zstandard-0.23.0
```



```

react_agent_basic.py
1 from langchain_google_genai import ChatGoogleGenerativeAI
2
3 llm = ChatGoogleGenerativeAI()
4
5 result = llm.invoke("Give me a fact about cats")
6
7 print(result)

```

**ChatGoogleGenerativeAI #**

```

class langchain_google_genai.chat_models.ChatGoogleGenerativeAI # [source]
Bases: BaseGoogleGenerativeAI, BaseChatModel

```

Google AI chat models integration.

**Instantiation:**

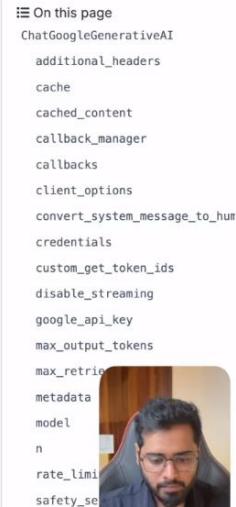
To use, you must have either:

1. The `GOOGLE_API_KEY` environment variable set with your API key, or
2. Pass your API key using the `google_api_key` kwarg to the `ChatGoogleGenerativeAI` constructor.

```

from langchain_google_genai import ChatGoogleGenerativeAI
llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")
llm.invoke("Write me a ballad about LangChain")

```



**Get API key**

**Quickly test the Gemini API**

```

curl "https://generativelanguage.googleapis.com/v1beta/models/gemini-1.5-flash:generateContent?"
key=GEMINI_API_KEY" \
-H 'Content-Type: application/json' \
-X POST \
-d '{
  "contents": [
    "parts": [{"text": "Explain how AI works"}]
  }
}'

```

**Create API key**

API key generated

Use your API keys securely. Do not share them or embed them in code the public can view.

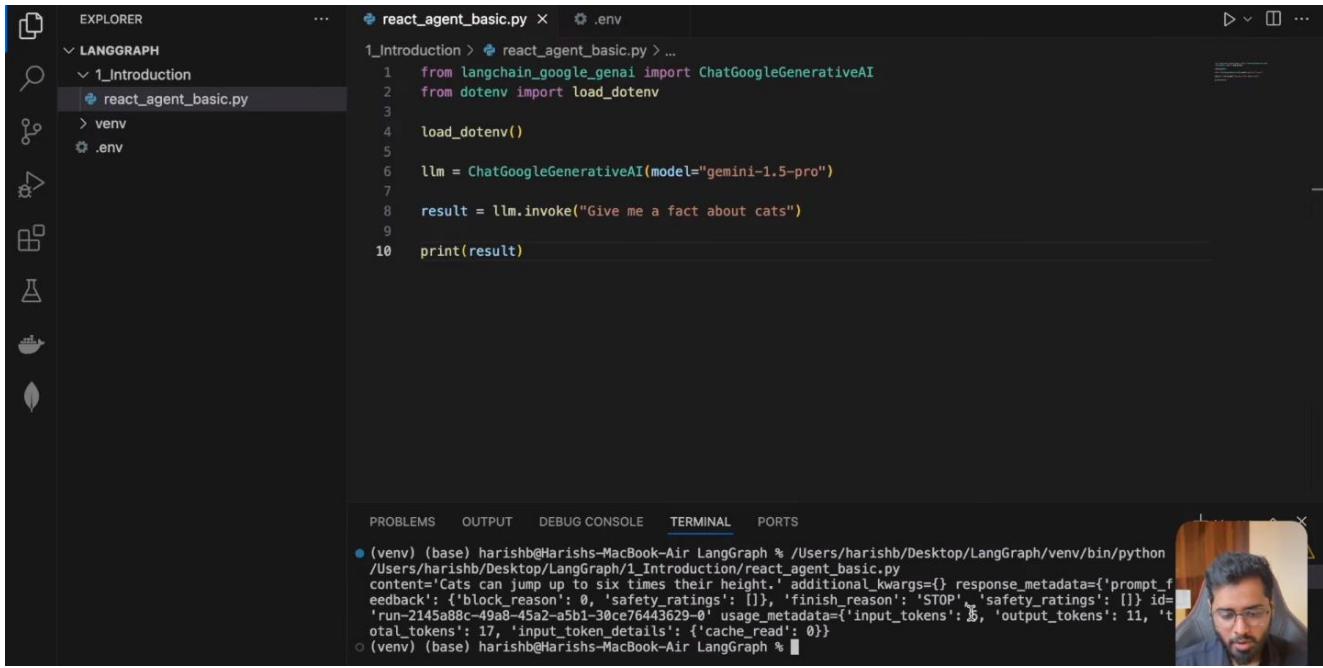
AlzaSyAG1YLD2BX-SB4vfnhNk5nb0oSm\_nbwXw

**Copy**

```

.env
1 GOOGLE_API_KEY=AlzaSyAG1YLD2BX-SB4vfnhNk5nb0oSm_nbwXw

```



```
from langchain_google_genai import ChatGoogleGenerativeAI
from dotenv import load_dotenv

load_dotenv()

llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")

result = llm.invoke("Give me a fact about cats")

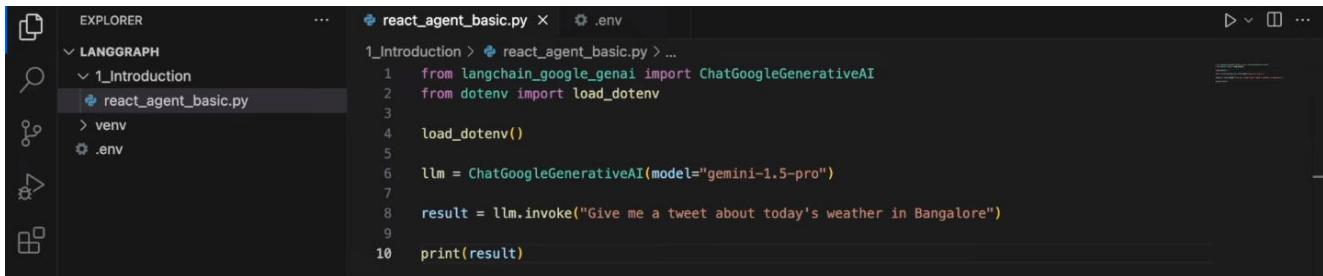
print(result)
```

(venv) (base) harishb@Harishb-MacBook-Air LangGraph % /Users/harishb/Desktop/LangGraph/1\_Introduction/react\_agent\_basic.py

content='Cats can jump up to six times their height' additional\_kwargs={} response\_metadata={'prompt\_feedback': {'block\_reason': 0, 'safety\_ratings': []}, 'finish\_reason': 'STOP', 'safety\_ratings': []} id='run-2145a88c-49a8-45a2-a5b1-30ce76443629-0' usage\_metadata={'input\_tokens': 5, 'output\_tokens': 11, 'total\_tokens': 17, 'input\_token\_details': {'cache\_read': 0}}

(venv) (base) harishb@Harishb-MacBook-Air LangGraph %

We can see that the model was correctly invoked and we got a response back. Next, let us give a complex prompt that the LLM can't answer as below.



```
from langchain_google_genai import ChatGoogleGenerativeAI
from dotenv import load_dotenv

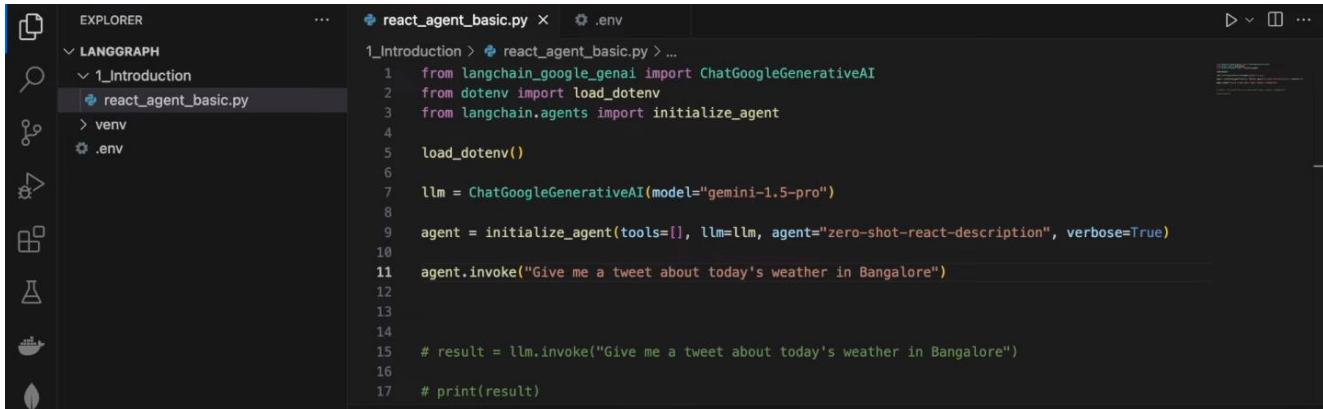
load_dotenv()

llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")

result = llm.invoke("Give me a tweet about today's weather in Bangalore")

print(result)
```

This is not accurate and the LLM is hallucinating, we need to give the LLM some tools.



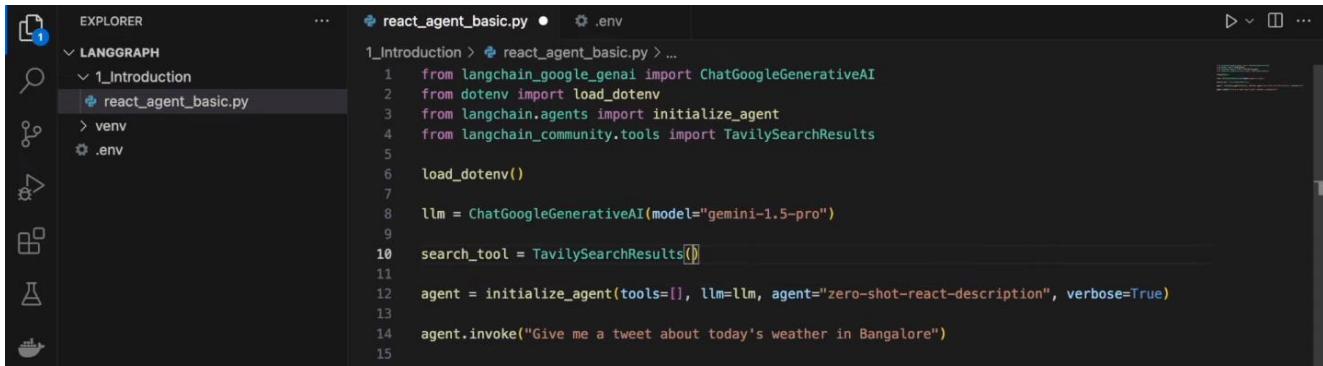
```
from langchain_google_genai import ChatGoogleGenerativeAI
from dotenv import load_dotenv
from langchain.agents import initialize_agent

load_dotenv()

llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")

agent = initialize_agent(tools=[], llm=llm, agent="zero-shot-react-description", verbose=True)
agent.invoke("Give me a tweet about today's weather in Bangalore")
```

We need to provide at least one tool for the agent to use, we will use the TavilySearchResults tool for making a Google search as below



```
from langchain_google_genai import ChatGoogleGenerativeAI
from dotenv import load_dotenv
from langchain.agents import initialize_agent
from langchain_community.tools import TavilySearchResults

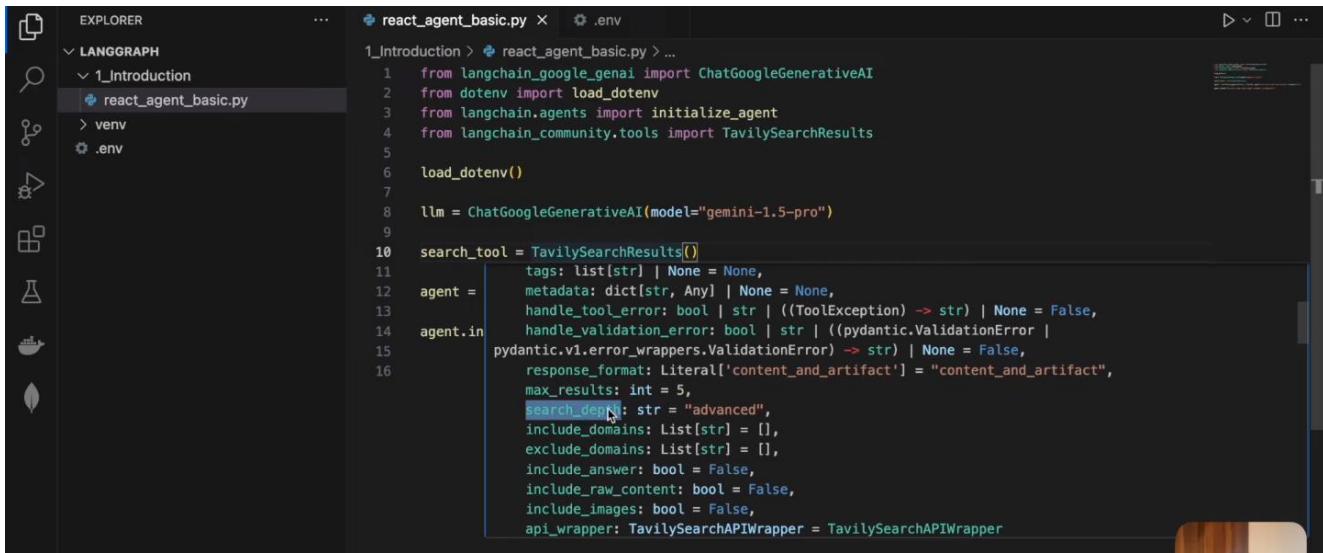
load_dotenv()

llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")

search_tool = TavilySearchResults()

agent = initialize_agent(tools=[], llm=llm, agent="zero-shot-react-description", verbose=True)

agent.invoke("Give me a tweet about today's weather in Bangalore")
```

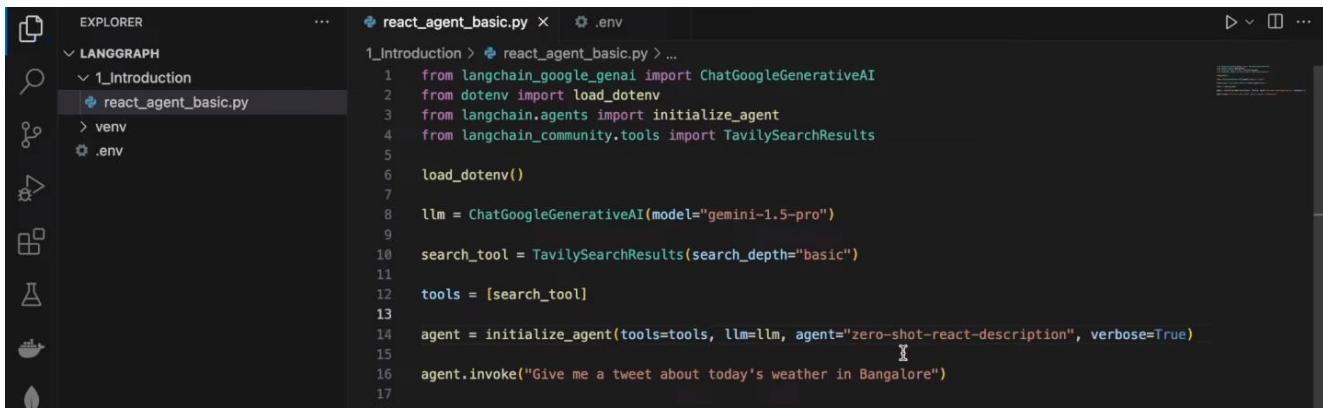


```
from langchain_google_genai import ChatGoogleGenerativeAI
from dotenv import load_dotenv
from langchain.agents import initialize_agent
from langchain_community.tools import TavilySearchResults

load_dotenv()

llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")

search_tool = TavilySearchResults()
    tags: list[str] | None = None,
    metadata: dict[str, Any] | None = None,
    handle_tool_error: bool | str | ((ToolException) -> str) | None = False,
    handle_validation_error: bool | str | ((pydantic.ValidationError |
    pydantic.v1.error_wrappers.ValidationError) -> str) | None = False,
    response_format: Literal['content_and_artifact'] = "content_and_artifact",
    max_results: int = 5,
    search_depth: str = "advanced",
    include_domains: List[str] = [],
    exclude_domains: List[str] = [],
    include_answer: bool = False,
    include_raw_content: bool = False,
    include_images: bool = False,
    api_wrapper: TavilySearchAPIWrapper = TavilySearchAPIWrapper
```



```
from langchain_google_genai import ChatGoogleGenerativeAI
from dotenv import load_dotenv
from langchain.agents import initialize_agent
from langchain_community.tools import TavilySearchResults

load_dotenv()

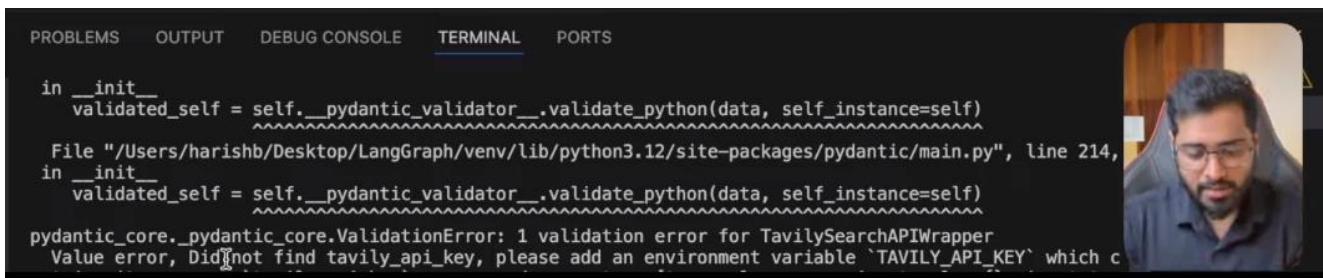
llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")

search_tool = TavilySearchResults(search_depth="basic")

tools = [search_tool]

agent = initialize_agent(tools=tools, llm=llm, agent="zero-shot-react-description", verbose=True)

agent.invoke("Give me a tweet about today's weather in Bangalore")
```



```
in __init__
    validated_self = self.__pydantic_validator__.validate_python(data, self_instance=self)
File "/Users/harishb/Desktop/LangGraph/venv/lib/python3.12/site-packages/pydantic/main.py", line 214,
in __init__
    validated_self = self.__pydantic_validator__.validate_python(data, self_instance=self)
pydantic_core._pydantic_core.ValidationException: 1 validation error for TavilySearchAPIWrapper
  Value error, DidNot find tavily_api_key, please add an environment variable `TAVILY_API_KEY` which c
```

We need the Tavily API KEY value from the Tavily free tier as below

The screenshot shows a Google search results page for "tavily api key". The first result is a link to the Tavily website, which displays an "Overview" page. The page includes sections for "API Usage" (5/1,000 Credits) and "API Keys". A table lists one key: "default" (dev type, usage 5, key: "tvly-\*\*\*\*\*"). Below the table is a note about using the key for Research API requests. The right side of the page features a placeholder for a user profile picture.

The screenshot shows a code editor interface with two tabs: "react\_agent\_basic.py" and ".env". The ".env" tab displays environment variables:

```
GOOGLE_API_KEY=AIzaSyAG1YLD2BX5-BS4vfNhNk5nb0oSm_nbwXw
TAVILY_API_KEY=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
```

The "react\_agent\_basic.py" tab contains the following Python code:

```
from langchain_google_genai import ChatGoogleGenerativeAI
from dotenv import load_dotenv
from langchain.agents import initialize_agent
from langchain_community.tools import TavilySearchResults
load_dotenv()
llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")
search_tool = TavilySearchResults(search_depth="basic")
tools = [search_tool]
agent = initialize_agent(tools=tools, llm=llm, agent="zero-shot-react-description", verbose=True)
agent.invoke("Give me a funny tweet about today's weather in Bangalore")
```

At the bottom of the screen, a terminal window shows the command: `(venv) harishb@Harishb-MacBook-Air LangGraph % /Users/harishb/Desktop/LangGraph/venv/bin/python /Users/harishb/Desktop/LangGraph/1_Introduction/react_agent_basic.py`.

PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

```
(venv) (base) harishb@Harishs-MacBook-Air LangGraph % /Users/harishb/Desktop/LangGraph/venv/bin/python /Users/harishb/Desktop/LangGraph/1_Introduction/react_agent_basic.py /Users/harishb/Desktop/LangGraph/1_Introduction/react_agent_basic.py:14: LangChainDeprecationWarning: LangChain agents will continue to be supported, but it is recommended for new use cases to be built with LangGraph. LangGraph offers a more flexible and full-featured framework for building agents, including support for tool-calling, persistence of state, and human-in-the-loop workflows. For details, refer to the `LangGraph documentation <https://langchain-ai.github.io/langgraph/>`_ as well as guides for `Migrating from AgentExecutor <https://python.langchain.com/docs/how\_to/migrate\_agent/>`_ and LangGraph's `P re-built ReAct agent <https://langchain-ai.github.io/langgraph/how-tos/create-react-agent/>`_. agent = initialize_agent(tools=tools, llm=llm, agent="zero-shot-react-description", verbose=True)
```

> Entering new AgentExecutor chain...

**Question:** Give me a funny tweet about today's weather in Bangalore

*Thought: I need to find out what the weather is like in Bangalore today. Then I can try to write a funny tweet about it.*

Action: `tavily_search_results_json`

Action Input: weather in Bangalore today

The ReAct pattern is THINK...ACTION...OBSERVE, so we expect to see this kind of result below

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
re-built ReAct agent <https://langchain-ai.github.io/langgraph/how-tos/create-react-agent/>_.  
    agent = initialize_agent(tools=tools, llm=llm, agent="zero-shot-react-description", verbose=True)
```

> Entering new AgentExecutor chain...

**Question:** Give me a funny tweet about today's weather in Bangalore

*Thought: I need to find out what the weather is like in Bangalore today. Then I can try to write a funny tweet about it.*

Action: *tavily\_search\_results\_json*

Action Input: weather in Bangalore today

**Observation:** [{"url": "https://www.weatherapi.com/", "content": {"location": {"name": "Bangalore", "region": "Karnataka", "country": "India", "lat": 12.9833, "lon": 77.5833, "tz\_id": "Asia/Kolkata", "local\_time\_epoch": 1739083165, "localtime": "2025-02-09 12:09"}, "current": {"last\_updated\_epoch": 1739082600, "last\_updated": "2025-02-09 12:00", "temp\_c": 27.2, "temp\_f": 81.0, "is\_day": 1, "condition": {"text": "Sunny", "icon": "//cdn.weatherapi.com/weather/64x64/day/113.png", "code": 1000}, "wind\_mph": 12.8, "wind\_kph": 20.5, "wind\_degree": 89, "wind\_dir": "E", "pressure\_mb": 1018.0, "pressure\_in": 30.06, "precip\_mm": 0.0, "precip\_in": 0.0, "humidity": 39, "cloud": 0, "feelslike\_c": 26.1, "feelslike\_f": 78.9, "windchill\_c": 28.8, "windchill\_f": 83.8, "heatindex\_c": 27.4, "heatindex\_f": 81.3, "dewpoint\_c": 8.1, "dewpoint\_f": 46.6, "vis\_km": 6.0, "vis\_miles": 3.0, "uv": 10.5, "gust\_mph": 14.7, "gust\_kph": 23.6}}, {"url": "https://www.msn.com/en-in/news/India/bangalore-weather-and-aqi-today-warm-start-at-1762-c-check-weather-forecast-for-february-9-2025/ar-AA1yFEQ7", "content": "The temperature in Bangalore today, on February 9, 2025, is 25.55 °C. The day's forecast indicates a minimum and maximum temperature of 17.62 °C and 29.33 °C, respectively. The relative humidity is 39%. The wind direction is East, with a speed of 20.5 km/h. The pressure is 1018 mb. The dew point is 8.1 °C. The heat index is 27.4 °C. The wind chill is 28.8 °C. The visibility is 6.0 km. The UV index is 10.5. The gust wind speed is 14.7 mph or 23.6 kph."}, {"url": "https://www.hindustantimes.com/cities/bengaluru-news/bangalore-weather-today-aqi-and-rain-forecast-updates-february-9-2025-101739064606425.html", "content": "Bangalore Weather and AQI Today: Warm start at 17.62 °C. Check weather forecast for February 9, 2025, in Bangalore. The current temperature is 25.55 °C. The forecast for the day indicates a minimum of 17.62 °C and a maximum of 29.33 °C. The relative humidity is 39%. The wind direction is East, with a speed of 20.5 km/h. The pressure is 1018 mb. The dew point is 8.1 °C. The heat index is 27.4 °C. The wind chill is 28.8 °C. The visibility is 6.0 km. The UV index is 10.5. The gust wind speed is 14.7 mph or 23.6 kph."}]}

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

*Thought: It seems the weather in Bangalore is sunny and warm, around 27°C. That's good information for a tweet.*

*Thought: I can now write a funny tweet.*

**Final Answer:** Bangalore weather is so perfect today, I almost forgot I have responsibilities. Almost. #BangaloreWeather #SunnySideUp

> Finished chain.

(venv) (base) harishb@Harishs-MacBook-Air LandGraph %

The agent is now using a tool to get the correct correct

smith.langchain.com/hub/hwchase17/react

hwchase17/react Public Try it

Prompt Commits 1 Updated a year ago • 117 • 135k • 5.55M

**PromptTemplate**

Answer the following questions as best you can. You have access to the following tools:

{tools}

Use the following format:

Question: the input question you must answer  
Thought: you should always think about what to do  
Action: the action to take, should be one of [{tool\_names}]  
Action Input: the input to the action  
Observation: the result of the action  
... (this Thought/Action/Action Input/Observation can repeat N times)  
Thought: I now know the final answer  
Final Answer: the final answer to the original input question

Begin!

Question: {input}  
Thought:{agent\_scratchpad}

DETAILS

TYPE StringPromptTemplate

COMMITS 1 commit

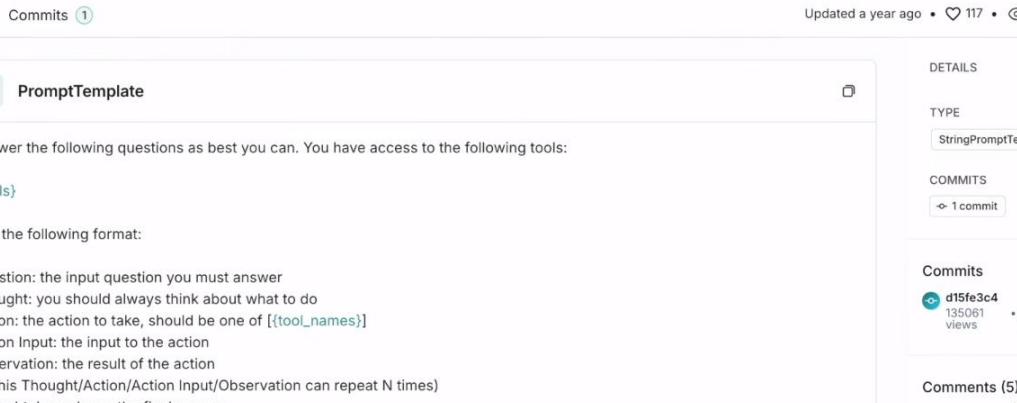
Commits

d15fe3c4 135061 views 5554172 downloads a year

Comments (5)

M Matteo No i've a problem with a multi-input only one input.

A Aditya Kar 3 replies • 0 likes



Above is the prompt being used by the Agent in the background for the LLM tool call in the ReAct pattern, the agent context is provided in the **agent scratchpad**.

```

1_Introduction > react_agent_basic.py > ...
1   from langchain_google_genai import ChatGoogleGenerativeAI
2   from dotenv import load_dotenv
3   from langchain.agents import initialize_agent
4   from langchain_community.tools import TavilySearchResults
5
6   load_dotenv()
7
8   llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")
9
10  search_tool = TavilySearchResults(search_depth="basic")
11
12  tools = [search_tool]
13
14  agent = initialize_agent(tools=tools, llm=llm, agent="zero-shot-react-description", verbose=True)
15
16  agent.invoke("When was SpaceX's last launch and how many days ago was that from this instant")
17
18

```

Next, let us allow the agent to be able to use a few more tools so it can answer more complex problems.

```

> Entering new AgentExecutor chain...
Question: When was SpaceX's last launch and how many days ago was that from this instant
Thought: I need to find the date of the most recent SpaceX launch and then calculate the difference between that date and today. I can use the tavily_search_results_json tool to find this information.
Action: tavily_search_results_json
Action Input: "SpaceX most recent launch date"
Observation: [{"url": "https://rocketlaunch.org/launch-schedule/spacex", "content": "The most comprehensive, up-to-date list of upcoming SpaceX rocket launches. Search the 113 scheduled launches, with filters to easily find specific SpaceX missions. Perfect for researchers and space enthusiasts. ... Last Update: Updated launch date and time. Transporter 13 (Dedicated SSO Rideshare) SpaceX TBD. Mar 2025."}, {"url": "https://www.spacex.com/news/live/spacex-starship-missions-updates", "content": "Some media reports have suggested SpaceX could attempt a Starship Flight 7 launch in early January, perhaps around Jan. 11, but SpaceX has not yet issued a targeted launch date. 2024-11-19T23:26:"}, {"url": "https://www.spacex.com/space-exploration/launches-spacecraft/spacex-launching-27-starlink-satellites-today-in-1st-flight-since-starship-explosion", "content": "SpaceX launching 27 Starlink satellites today in 1st flight since Starship explosion | Space Space Search Space SpaceX launching 27 Starlink satellites today in 1st flight since Starship explosion A SpaceX Falcon 9 rocket launches the NROL-167 mission for the U.S. National Reconnaissance Office from Vandenberg Space Force Base in California on Oct. 24, 2024. SpaceX is set to launch another batch of its Starlink internet satellites today (Jan. 18), just two days after a test flight of the company's Starship megarocket ended in an explosion. SpaceX launches 21 Starlink satellites from Florida, lands booster for 15th time (video) Breaking space news, the latest updates on rocket launches, skywatching events and more! MORE FROM SPACE..."}, {"url": "https://www.spacex.com/launches/", "content": "SpaceX designs, manufactures and launches advanced rockets and spacecraft. The company's goal is to reduce the cost of space transportation and make long-term space habitation possible."}]

```

```

o live on other planets. ... Starlink Mission: SpaceX's 100th Successful Flight. Learn More. OCTOBER 18 , 2020. STARLINK MISSION. Learn More. OCTOBER 06, 2020", "url": "https://spaceflightnow.com/launch-schedule/", "content": "Updated:\nOctober 26\nA United Launch Alliance Atlas 5 rocket, designated AV-085, will launch Boeing's CST-100 Starliner spacecraft on its first mission with astronauts, known as the Crew Test Flight, to the International Space Station. Updated:\nNovember 13\nA SpaceX Falcon Heavy will launch the fourth and final satellite of the next-generation series of geostationary weather satellites for NASA and NOAA. Updated:\nDecember 30\nA SpaceX Falcon 9 rocket will launch Northrop Grumman's 21st Cygnus cargo freighter on the 20th operational cargo delivery flight to the International Space Station. Updated:\nDecember 13\nA SpaceX Falcon 9 rocket will launch the IM-1 mission with the Nova-C lander built and owned by Intuitive Machines. Updated:\nDecember 17\nA SpaceX Falcon 9 will launch a mission for the U.S. Space Force and Missile Defense Agency.\n}
Thought: Thought: The search results indicate the most recent launch was a Starlink mission on January 8, 2025. I need to calculate how many days ago that was.

Action: calculate_days_since
Action Input: 2025-01-18
Observation: calculate_days_since is not a valid tool, try one of [tavily_search_results_json].
Thought: Thought: I can't directly calculate the days since a date using the available tools. I'll try searching for the number of days between January 8, 2025 and today's date. I'll need today's date first.

Action: tavily_search_results_json
Action Input: "what is today's date"
Observation: [{"url": "https://whatisthedatedtoday.com/index.html", "content": "Find Today's Date, today"}]

```

```

1_Introduction > react_agent_basic.py > ...
11
12     tools = [search_tool]
13
14     agent = initialize_agent(tools=tools, llm=llm, agent="zero-shot-react-description", verbose=True)
15
16     agent.invoke("When was SpaceX's last launch and how many days ago was that from this instant")
17
18

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

two dates calculator. Input start and end dates for precise results.'}, {'url': 'https://www.timeanddate.com/date/durationresult.html', 'content': 'Calculate Duration Between Two Dates - Result Calculators Date Calculator (add / subtract) Date Pattern Calculator Home |x@0 Calculators |x@0 Days Calculator: Days Between Two Dates Days Calculator: Days Between Two Dates How many days, months, and years are there between two dates? Include end date in calculation (1 day is added) Start again with a new calculation between two other dates New calculation, with both date and time included Some typical uses for the Date Calculators Date Calculators Duration Between Two Dates - Calculates number of days Time and Date Duration - Calculate duration, with both date and time included Date Calculator - Add or subtract days, months, years @ Time and Date AS 1995-2025 Calculators © Time and Date AS 1995-2025.'}
Thought:Retrying langchain_google_genai.chat_models._chat_with_retry.<locals>._chat_with_retry in 2.0 seconds as it raised ResourceExhausted: 429 Resource has been exhausted (e.g. check quota).. Traceback (most recent call last):
File "/Users/harishb/Desktop/LangGraph/1_Introduction/react_agent_basic.py", line 16, in <module>
    agent.invoke("When was SpaceX's last launch and how many days ago was that from this instant")
File "/Users/harishb/Desktop/LangGraph/venv/lib/python3.12/site-packages/langchain/chains/base.py", line 170, in invoke
    raise e

```

The LLM is hallucinating a date calculation tool which it doesn't have. We need to provide a Python function tool to use as below.

```

1_Introduction > react_agent_basic.py > .env
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

from langchain_google_genai import ChatGoogleGenerativeAI
from dotenv import load_dotenv
from langchain.agents import initialize_agent, tool
from langchain_community.tools import TavilySearchResults
import datetime

load_dotenv()

llm = ChatGoogleGenerativeAI(model="gemini-1.5-pro")

search_tool = TavilySearchResults(search_depth="basic")

@tool
def get_system_time(format: str = "%Y-%m-%d %H:%M:%S"):
    """ Returns the current date and time in the specified format """
    current_time = datetime.datetime.now()
    formatted_time = current_time.strftime(format)
    return formatted_time

tools = [search_tool, get_system_time]

agent = initialize_agent(tools=tools, llm=llm, agent="zero-shot-react-description", verbose=True)

agent.invoke("When was SpaceX's last launch and how many days ago was that from this instant")

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

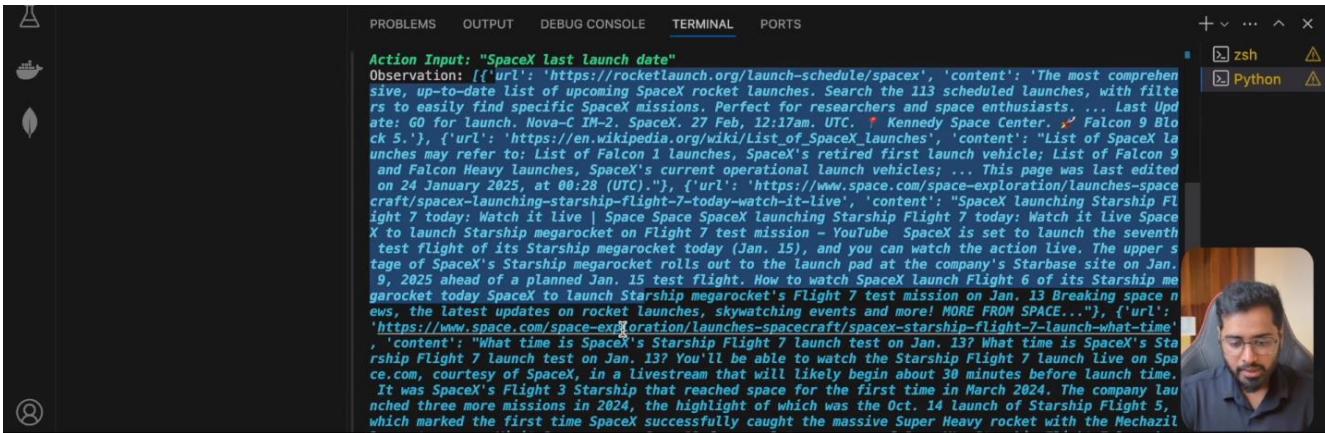
```

LangGraph. LangGraph offers a more flexible and full-featured framework for building agents, including support for tool-calling, persistence of state, and human-in-the-loop workflows. For details, refer to the LangGraph documentation <https://langchain-ai.github.io/langgraph/>_ as well as guides for Migrating from AgentExecutor <https://python.langchain.com/docs/how_to/migrate_agent/>_ and LangGraph's pre-built ReAct agent <https://langchain-ai.github.io/langgraph/how_tos/create-react-agent/>_.
agent = initialize_agent(tools=tools, llm=llm, agent="zero-shot-react-description", verbose=True)

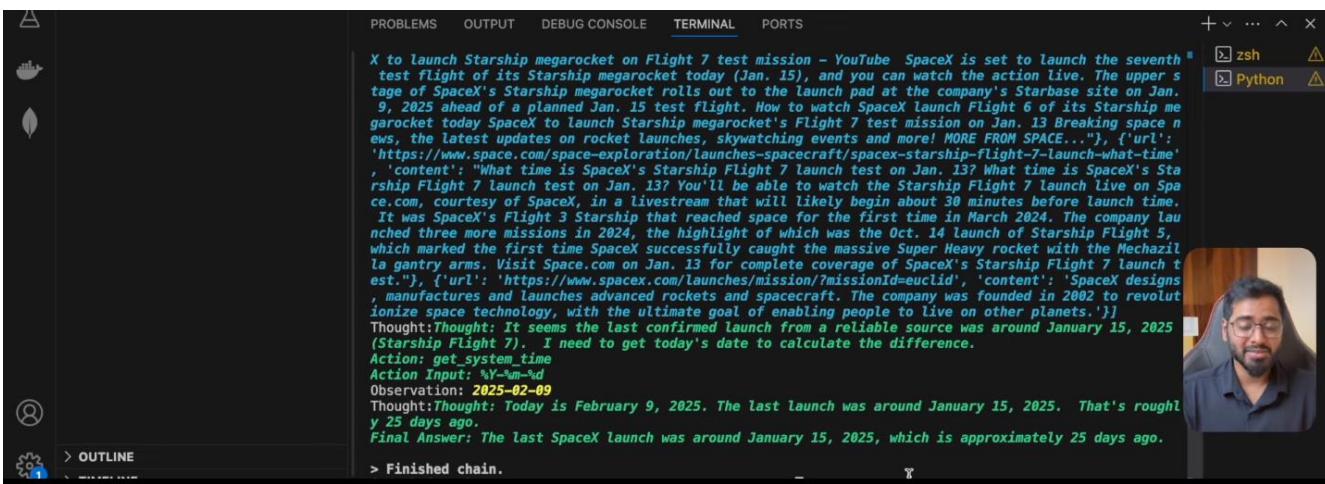
> Entering new AgentExecutor chain...
Question: When was SpaceX's last launch and how many days ago was that from this instant
Thought: I need to find the date of the last SpaceX launch and then calculate the difference between that date and today's date.
Action: tavily_search_results_json
Action Input: "SpaceX last launch date"
Observation: [{"url": "https://rocketlaunch.org/launch-schedule/spacex", "content": "The most comprehensive, up-to-date list of upcoming SpaceX rocket launches. Search the 113 scheduled launches, with filters to easily find specific SpaceX missions. Perfect for researchers and space enthusiasts. ... Last Updated: 60 for launch, Nova-C IN-2, SpaceX-27 Feb, 12:17am, UTC. ↗ Kennedy Space Center. ↗ Falcon 9 Block 5."}, {"url": "https://en.wikipedia.org/wiki/List_of_SpaceX_launches", "content": "List of SpaceX launches which may refer to: List of Falcon 1 launches, SpaceX's retired first launch vehicle; List of Falcon 9 and Falcon Heavy launches, SpaceX's current operational launch vehicles; ... This page was last edited on 24 January 2025, at 00:28 (UTC)."}, {"url": "https://www.space.com/space-exploration/launches-spacecraft/spacex-launching-starship-flight-7-today-watch-it-live", "content": "SpaceX launching Starship Flight 7 today: Watch it live | Space Space SpaceX launching Starship Flight 7 today: Watch it live Space"}]

```

Action Input: "SpaceX last launch date"  
Observation: [ {'url': 'https://rocketlaunch.org/launch-schedule/spacex', 'content': 'The most comprehensive, up-to-date list of upcoming SpaceX rocket launches. Search the 113 scheduled launches, with filters to easily find specific SpaceX missions. Perfect for researchers and space enthusiasts. ... Last Updated: 60 minutes ago. Nova-C IM-2. SpaceX. 27 Feb, 12:17am UTC. At Kennedy Space Center. Falcon 9 Block 5.'}, {'url': 'https://en.wikipedia.org/wiki/List\_of\_SpaceX\_launches', 'content': 'List of SpaceX launches may refer to: List of Falcon 1 launches, SpaceX's retired first launch vehicle; List of Falcon 9 and Falcon Heavy launches, SpaceX's current operational launch vehicles; ... This page was last edited on 24 January 2025, at 00:28 (UTC).'}, {'url': 'https://www.space.com/space-exploration/launches-spacecraft/spacex-launching-starship-flight-7-today-watch-it-Live', 'content': 'SpaceX launching Starship Flight 7 today: Watch it live | Space Space SpaceX launching Starship Flight 7 today: Watch it live SpaceX is set to launch the seventh test flight of its Starship megarocket today (Jan. 15), and you can watch the action live. The upper stage of SpaceX's Starship megarocket rolls out to the launch pad at the company's Starbase site on Jan. 9, 2025 ahead of a planned Jan. 15 test flight. How to watch SpaceX launch Flight 6 of its Starship megarocket today SpaceX to launch Starship megarocket's Flight 7 test mission on Jan. 13 Breaking space news, the latest updates on rocket launches, skywatching events and more! MORE FROM SPACE...'}, {'url': 'https://www.space.com/space-exploration/launches-spacecraft/spacex-starship-flight-7-launch-what-time', 'content': 'What time is SpaceX's Starship Flight 7 launch test on Jan. 13? What time is SpaceX's Starship Flight 7 launch test on Jan. 13? You'll be able to watch the Starship Flight 7 launch live on Space.com, courtesy of SpaceX, in a livestream that will likely begin about 30 minutes before launch time. It was SpaceX's Flight 3 Starship that reached space for the first time in March 2024. The company launched three more missions in 2024, the highlight of which was the Oct. 14 launch of Starship Flight 5, which marked the first time SpaceX successfully caught the massive Super Heavy rocket with the MechaZilla gantry arms. Visit Space.com on Jan. 13 for complete coverage of SpaceX's Starship Flight 7 launch test.'}, {'url': 'https://www.spacex.com/launches/mission/?missionId=euclid', 'content': 'SpaceX designs, manufactures and launches advanced rockets and spacecraft. The company was founded in 2002 to revolutionize space technology, with the ultimate goal of enabling people to live on other planets.'}]  
Thought: Thought: It seems the last confirmed launch from a reliable source was around January 15, 2025 (Starship Flight 7). I need to get today's date to calculate the difference.  
Action: get\_system\_time  
Action Input: %Y-%m-%d  
Observation: 2025-02-09  
Thought: Thought: Today is February 9, 2025. The last launch was around January 15, 2025. That's roughly 25 days ago.  
Final Answer: The last SpaceX launch was around January 15, 2025, which is approximately 25 days ago.



X to launch Starship megarocket on Flight 7 test mission - YouTube SpaceX is set to launch the seventh test flight of its Starship megarocket today (Jan. 15), and you can watch the action live. The upper stage of SpaceX's Starship megarocket rolls out to the launch pad at the company's Starbase site on Jan. 9, 2025 ahead of a planned Jan. 15 test flight. How to watch SpaceX launch Flight 6 of its Starship megarocket today SpaceX to launch Starship megarocket's Flight 7 test mission on Jan. 13 Breaking space news, the latest updates on rocket launches, skywatching events and more! MORE FROM SPACE...  
, {'url': 'https://www.space.com/space-exploration/launches-spacecraft/spacex-starship-flight-7-launch-what-time', 'content': 'What time is SpaceX's Starship Flight 7 launch test on Jan. 13? What time is SpaceX's Starship Flight 7 launch test on Jan. 13? You'll be able to watch the Starship Flight 7 launch live on Space.com, courtesy of SpaceX, in a livestream that will likely begin about 30 minutes before launch time. It was SpaceX's Flight 3 Starship that reached space for the first time in March 2024. The company launched three more missions in 2024, the highlight of which was the Oct. 14 launch of Starship Flight 5, which marked the first time SpaceX successfully caught the massive Super Heavy rocket with the MechaZilla gantry arms. Visit Space.com on Jan. 13 for complete coverage of SpaceX's Starship Flight 7 launch test.'}, {'url': 'https://www.spacex.com/launches/mission/?missionId=euclid', 'content': 'SpaceX designs, manufactures and launches advanced rockets and spacecraft. The company was founded in 2002 to revolutionize space technology, with the ultimate goal of enabling people to live on other planets.'}]  
Thought: Thought: It seems the last confirmed launch from a reliable source was around January 15, 2025 (Starship Flight 7). I need to get today's date to calculate the difference.  
Action: get\_system\_time  
Action Input: %Y-%m-%d  
Observation: 2025-02-09  
Thought: Thought: Today is February 9, 2025. The last launch was around January 15, 2025. That's roughly 25 days ago.  
Final Answer: The last SpaceX launch was around January 15, 2025, which is approximately 25 days ago.



This is an accurate answer that is grounded in reality and it was arrived at using tools in the ReAct framework. Next, we will see the issues with this Langchain ReAct approach and how using LangGraph will help us some those problems.