

# Output Parsers

## Instructor

Dipanjan Sarkar

Head of Community & Principal AI Scientist at Analytics Vidhya

Google Developer Expert - ML & Cloud Champion Innovator

Published Author



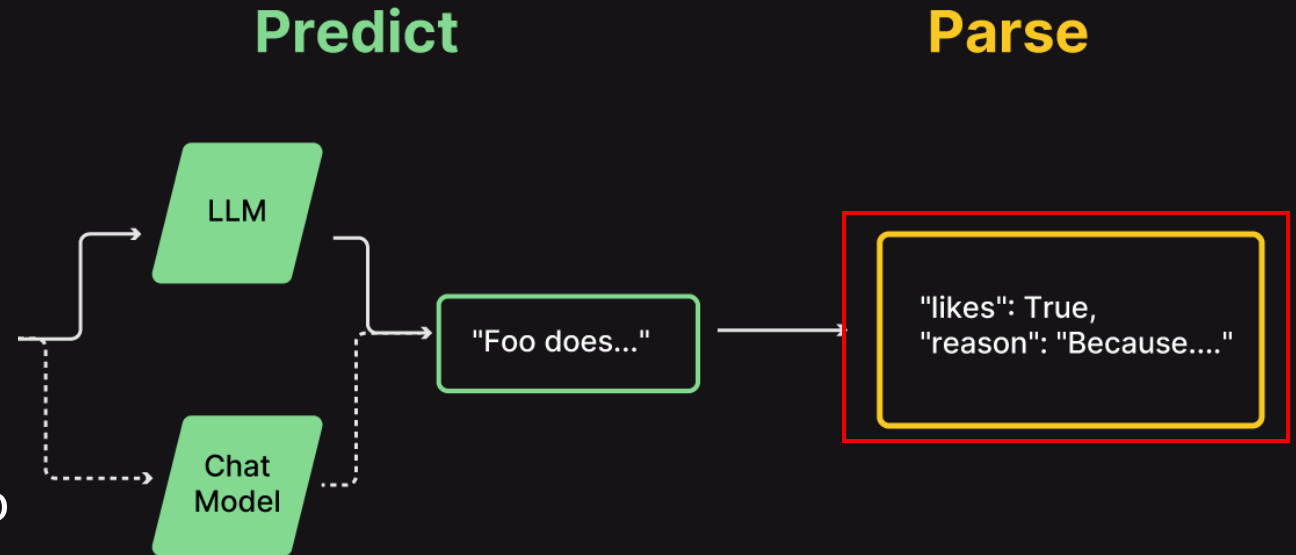
# Outline

- What is an Output Parser?
- Types of Output Parsers In LangChain
- Popular Built-in Output Parsers in LangChain

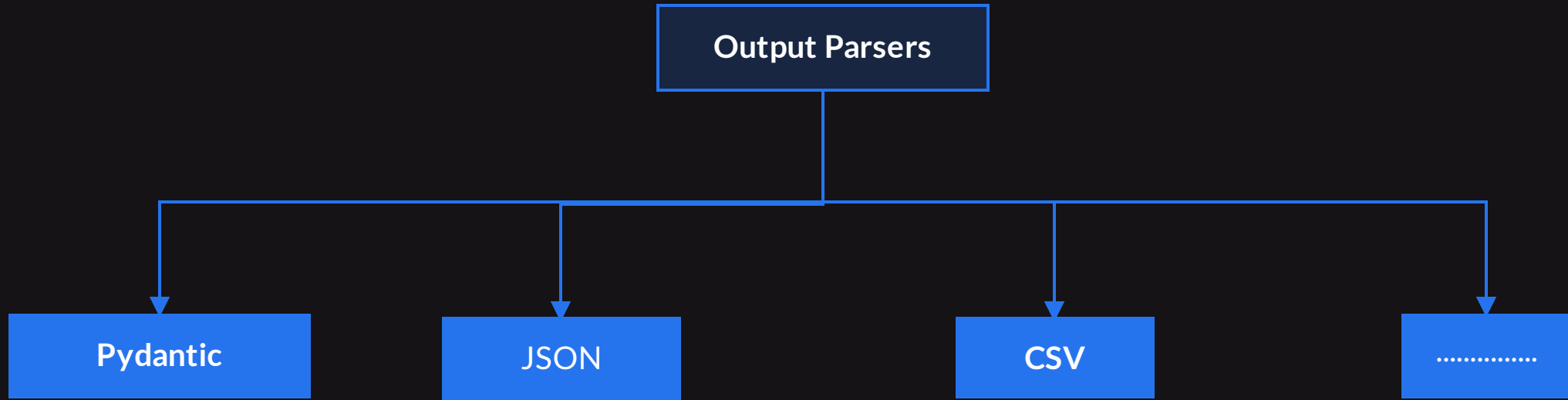


# What is an Output Parser?

- Output parsers are responsible for taking the output of an LLM and transforming it to a more consumable format
- Very useful when we want to get structured or semi-structured outputs to store them or use them in some workflow



# Types of Output Parsers In LangChain



- LangChain enables you to have access to a variety of output parsers to format LLM responses:
  - [PydanticOutputParser](#): Uses the Pydantic library to build a data class to format data into structured fields
  - [JsonOutputParser](#): Formats output into a well-defined JSON structure
  - [CommaSeparatedListOutputParser](#): Formats output to return a list of comma-separated items

# Popular Built-in Output Parsers in LangChain

Parser Type	Description
<a href="#"><u>JSON</u></a>	Returns a JSON object as specified. You specify a Pydantic model and it will return JSON for that model. Probably the most reliable output parser for getting structured data
<a href="#"><u>XML</u></a>	Returns a dictionary of tags. Use when XML output is needed. Use with models that are good at writing XML (like Anthropic Claude)
<a href="#"><u>CSV</u></a>	Returns a list of comma separated values
<a href="#"><u>Pydantic</u></a>	Takes a user defined Pydantic model and returns data in that format
<a href="#"><u>YAML</u></a>	Takes a user defined Pydantic model and returns data in that format. Uses YAML to encode it.

# Thank You

---