

# Quickstart Guide

## Contents

- Installation
- Environment Setup
- Building a Language Model Application: LLMs
- Building a Language Model Application: Chat Models

This tutorial gives you a quick walkthrough about building an end-to-end language model application with LangChain.

## Installation

To get started, install LangChain with the following command:

```
pip install langchain
```

## Environment Setup

Using LangChain will usually require integrations with one or more model providers, data stores, apis, etc.

For this example, we will be using OpenAI's APIs, so we will first need to install their SDK:

```
pip install openai
```

We will then need to set the environment variable in the terminal.

[Skip to main content](#)

Alternatively, you could do this from inside the Jupyter notebook (or Python script):

```
import os
os.environ["OPENAI_API_KEY"] = "..."
```

## Building a Language Model Application: LLMs

Now that we have installed LangChain and set up our environment, we can start building our language model application.

LangChain provides many modules that can be used to build language model applications. Modules can be combined to create more complex applications, or be used individually for simple applications.

LLMs: Get predictions from a language model



Prompt Templates: Manage prompts for LLMs



Chains: Combine LLMs and prompts in multi-step workflows



Agents: Dynamically Call Chains Based on User Input



Memory: Add State to Chains and Agents



## Building a Language Model Application: Chat Models

Similarly, you can use chat models instead of LLMs. Chat models are a variation on language models. While chat models use language models under the hood, the interface they expose is a

[Skip to main content](#)

bit different: rather than expose a “text in, text out” API, they expose an interface where “chat messages” are the inputs and outputs.

Chat model APIs are fairly new, so we are still figuring out the correct abstractions.

Get Message Completions from a Chat Model



Chat Prompt Templates



Chains with Chat Models



Agents with Chat Models



Memory: Add State to Chains and Agents

