# Prompts

A prompt for a language model is a set of instructions or input provided by a user to guide the model's response, **helping it understand the context and generate relevant and coherent language-based outpu**t, such as answering questions, completing sentences, or engaging in a conversation.

LangChain provides several classes and functions to help construct and work with prompts:

Prompt templates: Parametrized model inputs

Example selectors: Dynamically select examples to include in prompts

**Prompt templates are pre-defined recipes for generating prompts for language models.**

A template may include instructions, few-shot examples, and specific context and questions appropriate for a given task.

# Models

LangChain provides interfaces and integrations for two types of models:

* [LLMs](https://python.langchain.com/docs/modules/model_io/models/llms/): Models that take a text string as input and return a text string

There are lots of LLM providers (OpenAI, Cohere, Hugging Face, etc) - the LLM class is designed to provide a standard interface for all of them.

* [Chat models](https://python.langchain.com/docs/modules/model_io/models/chat/): Models that are backed by a language model but take a list of Chat Messages as input and return a Chat Message

Chat models are a variation on language models. While chat models use language models under the hood, the interface they use is a bit different. Rather than using a "text in, text out" API, they use an interface where "chat messages" are the inputs and outputs.