

Anthropic

This notebook covers how to get started with Anthropic chat models.

```
from langchain.chat_models import ChatAnthropic
from langchain.prompts.chat import (
    ChatPromptTemplate,
    SystemMessagePromptTemplate,
    AIMessagePromptTemplate,
    HumanMessagePromptTemplate,
)
from langchain.schema import AIMessage, HumanMessage, SystemMessage
```



```
chat = ChatAnthropic()
```

```
messages = [
    HumanMessage(
        content="Translate this sentence from English to French. I love programming."
    )
]
chat(messages)
```

```
AIMessage(content=" J'aime la programmation.", additional_kwargs={}, example=False)
```

ChatAnthropic also supports async and streaming functionality:

```
from langchain.callbacks.manager import CallbackManager
from langchain.callbacks.streaming_stdout import StreamingStdOutCallbackHandler
```

```
await chat.agenerate([messages])
```

```
LLMResult(generations=[[ChatGeneration(text=" J'aime programmer.", generation_info=None,
message=AIMessage(content=" J'aime programmer.", additional_kwargs={}, example=False))]], llm_output={}, run=
[RunInfo(run_id=UUID('8cc8fb68-1c35-439c-96a0-695036a93652'))])
```

```
chat = ChatAnthropic(
    streaming=True,
    verbose=True,
    callback_manager=CallbackManager([StreamingStdOutCallbackHandler()]),
)
chat(messages)
```

```
J'aime la programmation.
```

```
AIMessage(content=" J'aime la programmation.", additional_kwargs={}, example=False)
```