Search projects

Help Sponsors Log in Register

ollama 0.3.3

Latest version

```
pip install ollama
```

Released: Sep 9, 2024

The official Python client for Ollama.

Navigation

- Project description
- Release history
- Download files

Project description

Ollama Python Library

The Ollama Python library provides the easiest way to integrate Python 3.8+ projects with Ollama.

Install

```
pip install ollama
```

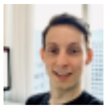
Usage

```
import ollama
response = ollama.chat(model='llama3.1', messages=[
```

Verified details ✓

These details have been [verified by PyPI](#)

Maintainers



jmorganca

Unverified details

*These details have **not** been verified by PyPI*

Project links

- Homepage
- Repository

Meta

- **License:** MIT License (MIT)
- **Author:** Ollama ✉
- **Requires:** Python <4.0, >=3.8

Classifiers

License

- OSI Approved :: MIT License

Programming Language

- Python :: 3
- Python :: 3.8
- Python :: 3.9
- Python :: 3.10
- Python :: 3.11
- Python :: 3.12

```
  {
    'role': 'user',
    'content': 'Why is the sky blue?',
  },
])
print(response['message']['content'])
```

Streaming responses

Response streaming can be enabled by setting `stream=True`, modifying function calls to return a Python generator where each part is an object in the stream.

```
import ollama

stream = ollama.chat(
    model='llama3.1',
    messages=[{'role': 'user', 'content': 'Why is the s
    stream=True,
)

for chunk in stream:
  print(chunk['message']['content'], end='', flush=True
```

API

The Ollama Python library's API is designed around the [Ollama REST API](#)

Chat

```
ollama.chat(model='llama3.1', messages=[{'role': 'user'
```

Generate

```
ollama.generate(model='llama3.1', prompt='Why is the sk
```

List

```
ollama.list()
```

Show

```
ollama.show('llama3.1')
```

Create

```
modelfile='''
FROM llama3.1
SYSTEM You are mario from super mario bros.
'''

ollama.create(model='example', modelfile=modelfile)
```

Copy

```
ollama.copy('llama3.1', 'user/llama3.1')
```

Delete

```
ollama.delete('llama3.1')
```

Pull

```
ollama.pull('llama3.1')
```

Push

```
ollama.push('user/llama3.1')
```

Embeddings

```
ollama.embeddings(model='llama3.1', prompt='The sky is
```

Ps

```
ollama.ps()
```

Custom client

A custom client can be created with the following fields:

- `host`: The Ollama host to connect to
- `timeout`: The timeout for requests

```
from ollama import Client
client = Client(host='http://localhost:11434')
response = client.chat(model='llama3.1', messages=[
  {
    'role': 'user',
    'content': 'Why is the sky blue?',
  },
])
```

Async client

```
import asyncio
from ollama import AsyncClient

async def chat():
  message = {'role': 'user', 'content': 'Why is the sky
  response = await AsyncClient().chat(model='llama3.1',

asyncio.run(chat())
```

Setting `stream=True` modifies functions to return a Python asynchronous generator:

```
import asyncio
from ollama import AsyncClient

async def chat():
  message = {'role': 'user', 'content': 'Why is the sky
  async for part in await AsyncClient().chat(model='lla
    print(part['message']['content'], end='', flush=Tru

asyncio.run(chat())
```

Errors

Errors are raised if requests return an error status or if an error is detected while streaming.

```
model = 'does-not-yet-exist'

try:
  ollama.chat(model)
except ollama.ResponseError as e:
  print('Error:', e.error)
  if e.status_code == 404:
    ollama.pull(model)
```

Help

Installing packages

Uploading packages

User guide

Project name retention

FAQs

About PyPI

PyPI Blog

Infrastructure dashboard

Statistics

Logos & trademarks

Our sponsors

Contributing to PyPI

Bugs and feedback

Contribute on GitHub

Translate PyPI

Sponsor PyPI

Development credits

Using PyPI

Code of conduct

Report security issue

Privacy policy

Terms of Use

Acceptable Use Policy

Status: All Systems Operational

Developed and maintained by the Python community, for the Python community.
Donate today!

Site map

Switch to desktop version

› English español français 日本語 português (Brasil) українська Ελληνικά Deutsch 中文 (简体)
中文 (繁體) русский עברית Esperanto