

Requests: HTTP for Humans™

Release v2.32.3. ([Installation](#))

downloads/month 575M license Apache-2.0 wheel yes python 3.8 | 3.9 | 3.10 | 3.11 | 3.12

Fork me on GitHub

Requests is an elegant and simple HTTP library for Python, built for human beings.

Behold, the power of Requests:

```
>>> r = requests.get('https://api.github.com/user', auth=('user', 'pass'))
>>> r.status_code
200
>>> r.headers['content-type']
'application/json; charset=utf8'
>>> r.encoding
'utf-8'
>>> r.text
'{"type": "User"... '
>>> r.json()
{'private_gists': 419, 'total_private_repos': 77, ...}
```

See [similar code, sans Requests](#).

Requests allows you to send HTTP/1.1 requests extremely easily. There's no need to manually add query strings to your URLs, or to form-encode your POST data. Keep-alive and HTTP connection pooling are 100% automatic, thanks to [urllib3](#).

Beloved Features

 v: latest ▾

Requests is ready for today's web.

- Keep-Alive & Connection Pooling
- International Domains and URLs
- Sessions with Cookie Persistence
- Browser-style SSL Verification
- Automatic Content Decoding
- Basic/Digest Authentication
- Elegant Key/Value Cookies
- Automatic Decompression
- Unicode Response Bodies
- HTTP(S) Proxy Support
- Multipart File Uploads
- Streaming Downloads
- Connection Timeouts
- Chunked Requests
- .netrc Support

Requests officially supports Python 3.8+, and runs great on PyPy.

The User Guide

This part of the documentation, which is mostly prose, begins with some background information about Requests, then focuses on step-by-step instructions for getting the most out of Requests.

- [Installation of Requests](#)
 - [\\$ python -m pip install requests](#)
 - [Get the Source Code](#)
- [Quickstart](#)

- [Make a Request](#)
- [Passing Parameters In URLs](#)
- [Response Content](#)
- [Binary Response Content](#)
- [JSON Response Content](#)
- [Raw Response Content](#)
- [Custom Headers](#)
- [More complicated POST requests](#)
- [POST a Multipart-Encoded File](#)
- [Response Status Codes](#)
- [Response Headers](#)
- [Cookies](#)
- [Redirection and History](#)
- [Timeouts](#)
- [Errors and Exceptions](#)
- [Advanced Usage](#)
 - [Session Objects](#)
 - [Request and Response Objects](#)
 - [Prepared Requests](#)
 - [SSL Cert Verification](#)
 - [Client Side Certificates](#)
 - [CA Certificates](#)
 - [Body Content Workflow](#)
 - [Keep-Alive](#)
 - [Streaming Uploads](#)
 - [Chunk-Encoded Requests](#)
 - [POST Multiple Multipart-Encoded Files](#)
 - [Event Hooks](#)
 - [Custom Authentication](#)

- [Streaming Requests](#)
- [Proxies](#)
- [Compliance](#)
- [HTTP Verbs](#)
- [Custom Verbs](#)
- [Link Headers](#)
- [Transport Adapters](#)
- [Blocking Or Non-Blocking?](#)
- [Header Ordering](#)
- [Timeouts](#)
- [Authentication](#)
 - [Basic Authentication](#)
 - [Digest Authentication](#)
 - [OAuth 1 Authentication](#)
 - [OAuth 2 and OpenID Connect Authentication](#)
 - [Other Authentication](#)
 - [New Forms of Authentication](#)

The Community Guide

This part of the documentation, which is mostly prose, details the Requests ecosystem and community.

- [Recommended Packages and Extensions](#)
 - [Certifi CA Bundle](#)
 - [CacheControl](#)
 - [Requests-Toolbelt](#)
 - [Requests-Threads](#)
 - [Requests-OAuthlib](#)
 - [Betamax](#)

- [Frequently Asked Questions](#)
 - [Encoded Data?](#)
 - [Custom User-Agents?](#)
 - [Why not HttpLib2?](#)
 - [Python 3 Support?](#)
 - [Python 2 Support?](#)
 - [What are “hostname doesn’t match” errors?](#)
- [Integrations](#)
- [Articles & Talks](#)
- [Support](#)
 - [Stack Overflow](#)
 - [File an Issue](#)
 - [Send a Tweet](#)
- [Vulnerability Disclosure](#)
 - [Process](#)
 - [Previous CVEs](#)
- [Release Process and Rules](#)
 - [Major Releases](#)
 - [Minor Releases](#)
 - [Hotfix Releases](#)
 - [Reasoning](#)
- [Community Updates](#)
- [Release History](#)

If you are looking for information on a specific function, class, or method, this part of the documentation is for you.

- [Developer Interface](#)
 - [Main Interface](#)
 - [Exceptions](#)
 - [Request Sessions](#)
 - [Lower-Level Classes](#)
 - [Lower-Lower-Level Classes](#)
 - [Authentication](#)
 - [Encodings](#)
 - [Cookies](#)
 - [Status Code Lookup](#)
 - [Migrating to 1.x](#)
 - [Migrating to 2.x](#)

The Contributor Guide

If you want to contribute to the project, this part of the documentation is for you.

- [Contributor's Guide](#)
 - [Be Cordial](#)
 - [Get Early Feedback](#)
 - [Contribution Suitability](#)
 - [Code Contributions](#)
 - [Steps for Submitting Code](#)
 - [Code Review](#)
 - [Code Style](#)
 - [New Contributors](#)

- [Documentation Contributions](#)
- [Bug Reports](#)
- [Feature Requests](#)
- [Authors](#)
 - [Keepers of the Crystals](#)
 - [Previous Keepers of Crystals](#)
 - [Patches and Suggestions](#)

There are no more guides. You are now guideless. Good luck.

```
# .readthedocs.yaml
build.tools:
  python: "3.11"
sphinx:
  configuration: conf.py
python.install:
  - requirements: pip.in
```

Docs as Code hosting is easy on Read the Docs. Free for open source, paid for business.

Sign up today.

Ad by EthicalAds · 