



Type '/' to search projects



Help

Docs

Sponsors

Log in

Register

# facebook-scraper 0.2.59



Latest version

```
pip install facebook-scraper
```



Released: Aug 30, 2022

Scrape Facebook public pages without an API key

## Navigation

Project description

Release history

Download files

## Verified details



These details have been verified by PyPI

## Maintainers



kevinzg

## Project description

### Facebook Scraper

downloads 6.3k/month downloads 1.6k/week downloads 177/day

python 3.6 | 3.7 | 3.8 | 3.9 pypi v0.2.59 commits since v0.2.59 16

code style black

Scrape Facebook public pages without an API key. Inspired by [twitter-scraper](#).

## Install

To install the latest release from PyPI:

```
pip install facebook-scraper
```

## Unverified details

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

## Project links

[Homepage](#)

[Repository](#)

## Meta

- **License:** MIT License (MIT)
- **Author:** [Kevin Zúñiga](#) [✉](#)
- **Requires:** Python >=3.6, <4.0

## Classifiers

### License

- [OSI Approved :: MIT License](#)

### Programming Language

- [Python :: 3](#)
- [Python :: 3.6](#)
- [Python :: 3.7](#)
- [Python :: 3.8](#)
- [Python :: 3.9](#)

Report project as  
malware

```
pip install git+https://github.com/kevinzg/facebook-scraper
```

## Usage

Send the unique **page name**, **profile name**, or **ID** as the first parameter and you're good to go:

```
>>> from facebook_scraper import get_posts

>>> for post in get_posts('nintendo', pages=1):
...     print(post['text'][:50])
...
The final step on the road to the Super Smash Bros
We're headed to PAX East 3/28-3/31 with new games
```

## Optional parameters

(For the `get_posts` function).

- **group:** group id, to scrape groups instead of pages. Default is `None`.
- **pages:** how many pages of posts to request, the first 2 pages may have no results, so try with a number greater than 2. Default is 10.
- **timeout:** how many seconds to wait before timing out. Default is 30.
- **credentials:** tuple of user and password to login before requesting the posts. Default is `None`.
- **extra\_info:** bool, if true the function will try to do an extra request to get the post reactions. Default is False.
- **youtube\_dl:** bool, use Youtube-DL for (high-quality) video extraction. You need to have youtube-dl installed on your environment. Default is False.
- **post\_urls:** list, URLs or post IDs to extract posts from. Alternative to fetching based on username.
- **cookies:** One of:

Logging into Facebook with an extension like [Get Cookies](#) (Chrome) or [Cookie Quick Manager](#) (Firefox). Make sure that you include both the c\_user cookie and the xs cookie, you will get an InvalidCookies exception if you don't.

- A [CookieJar](#)
- A dictionary that can be converted to a CookieJar with [cookiejar from dict](#)
- The string "from\_browser" to try extract Facebook cookies from your browser
- **options:** Dictionary of options. Set `options={"comments": True}` to extract comments, set `options={"reactors": True}` to extract the people reacting to the post. Both `comments` and `reactors` can also be set to a number to set a limit for the amount of comments/reactors to retrieve. Set `options={"progress": True}` to get a `tqdm` progress bar while extracting comments and replies. Set `options={"allow_extra_requests": False}` to disable making extra requests when extracting post data (required for some things like full text and image links). Set `options={"posts_per_page": 200}` to request 200 posts per page. The default is 4.

## CLI usage

```
$ facebook-scraper --filename nintendo_page_posts.csv -
```



Run `facebook-scraper --help` for more details on CLI usage.

**Note:** If you get a `UnicodeEncodeError` try adding `--encoding utf-8`.

## Post example

```
{'available': True,
 'comments': 459,
 'comments_full': None,
 'factcheck': None,
```

```
'is_live': False,
'likes': 3509,
'link': 'https://www.nintendo.com/amiibo/line-up/',
'post_id': '2257188721032235',
'post_text': 'Don't let this diminutive version of the Hero
            'Young Link is just as heroic as his fully grown
            'Link joins the Super Smash Bros. series of amiibo
            '\n'
            'https://www.nintendo.com/amiibo/line-up/',
'post_url': 'https://facebook.com/story.php?story_fbic
'reactions': {'haha': 22, 'like': 2657, 'love': 706,
'reactors': None,
'shared_post_id': None,
'shared_post_url': None,
'shared_text': '',
'shared_time': None,
'shared_user_id': None,
'shared_username': None,
'shares': 441,
'text': 'Don't let this diminutive version of the Hero
        'Young Link is just as heroic as his fully grown
        'joins the Super Smash Bros. series of amiibo
        '\n'
        'https://www.nintendo.com/amiibo/line-up/',
'time': datetime.datetime(2019, 4, 30, 5, 0, 1),
'user_id': '119240841493711',
'username': 'Nintendo',
'video': None,
'video_id': None,
'video_thumbnail': None,
'w3_fb_url': 'https://www.facebook.com/Nintendo/posts,
```

## Notes

- There is no guarantee that every field will be extracted (they might be `None`).
- Group posts may be missing some fields like `time` and `post_url`.
- Group scraping may return only one page and not work on private groups.
- If you scrape too much, Facebook might temporarily ban your IP.
- The vast majority of unique IDs on facebook (post IDs, video IDs, photo IDs, comment IDs, profile IDs, etc) can be appended to

Some functions (such as extracting reactions) require you to be logged into Facebook (pass cookies). If something isn't working as expected, try pass cookies and see if that fixes it.

## Profiles

The `get_profile` function can extract information from a profile's about section. Pass in the account name or ID as the first parameter. Note that Facebook serves different information depending on whether you're logged in (cookies parameter), such as Date of birth and Gender. Usage:

```
from facebook_scraper import get_profile
get_profile("zuck") # Or get_profile("zuck", cookies="c
```

### Outputs:

```
{'About': "I'm trying to make the world a more open place",
 'Education': 'Harvard University\n'
              'Computer Science and Psychology\n'
              '30 August 2002 - 30 April 2004\n'
              'Phillips Exeter Academy\n'
              'Classics\n'
              'School year 2002\n'
              'Ardsley High School\n'
              'High School\n'
              'September 1998 - June 2000',
 'Favourite Quotes': '"Fortune favors the bold."\n'
                    '- Virgil, Aeneid X.284\n'
                    '\n'
                    '"All children are artists. The problem is\n'
                    'to remain an artist once you grow up."\n'
                    '- Pablo Picasso\n'
                    '\n'
                    '"Make things as simple as possible, but not\n'
                    'simpler." - Albert Einstein',
 'Name': 'Mark Zuckerberg',
 'Places lived': [{'link': '/profile.php?id=10402292630',
                   'text': 'Palo Alto, California',
                   'type': 'Current town/city'},
                  {'link': '/profile.php?id=10550639614',
                   'text': 'New York City, New York',
                   'type': 'Previous town/city'}]}
```

```
'1 December 2015 - Present\n'
'Facebook\n'
'Founder and CEO\n'
'4 February 2004 - Present\n'
'Palo Alto, California\n'
'Bringing the world closer together.'}
```

To extract friends, pass the argument `friends=True`, or to limit the amount of friends retrieved, set `friends` to the desired number.

## Group info

The `get_group_info` function can extract info about a group. Pass in the group name or ID as the first parameter.

Note that in order to see the list of admins, you need to be logged in (cookies parameter).

Usage:

```
from facebook_scraper import get_group_info
get_group_info("makeupartistsgroup") # or get_group_info("123456789")
```

Output:

```
{'admins': [{'link': '/africanstylemagazinecom/?refid=18', 'name': 'African Style Magazine'},
{'link': '/connectfluencer/?refid=18', 'name': 'Everythingbrightandbeautiful'},
{'link': '/Kaakakigroup/?refid=18', 'name': 'Kaakaki'},
{'link': '/opentoohelp/?refid=18', 'name': 'Opentoohelp'}],
'id': '579169815767106',
'members': 6814229,
'name': 'HAIRSTYLES',
'type': 'Public group'}
```

To-Do

- [Profiles or post authors](#) (`get_profile()`)
- [Comments](#) (with `options={'comments': True}`)

## Alternatives and related projects

- [facebook-post-scraper](#). Has comments. Uses Selenium.
- [facebook-scraper-selenium](#). "Scrape posts from any group or user into a .csv file without needing to register for any API access".
- [Ultimate Facebook Scraper](#). "Scrapes almost everything about a Facebook user's profile". Uses Selenium.
- [Unofficial APIs](#). List of unofficial APIs for various services, none for Facebook for now, but might be worth to check in the future.
- [major-scrapy-spiders](#). Has a profile spider for Scrapy.
- [facebook-page-post-scraper](#). Seems abandoned.
  - [FBLYZE](#). Fork (?).
- [RSSHub](#). Generates an RSS feed from Facebook pages.
- [RSS-Bridge](#). Also generates RSS feeds from Facebook pages.



## Help

- [Installing packages](#) [↗](#)
- [Uploading packages](#) [↗](#)
- [User guide](#) [↗](#)
- [Project name retention](#) [↗](#)
- [FAQs](#)

## Contributing to PyPI

- [Bugs and feedback](#)
- [Contribute on GitHub](#) [↗](#)

## About PyPI

- [PyPI Blog](#) [↗](#)
- [Infrastructure dashboard](#) [↗](#)
- [Statistics](#)
- [Logos & trademarks](#)
- [Our sponsors](#)

## Using PyPI

- [Terms of Service](#) [↗](#)
- [Report security issue](#)

🗨️ 2025 Python Packaging Survey is now live!

Take the survey now ↗

[Development credits](#) ↗

[Acceptable Use Policy](#) ↗

---

Status: [All Systems Operational](#) ↗

Developed and maintained by the Python community, for the Python community.

[Donate today!](#)

"PyPI", "Python Package Index", and the blocks logos are registered [trademarks](#) of the [Python Software Foundation](#) ↗.

© 2025 [Python Software Foundation](#) ↗

[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](#)   [한국어](#)