Accidentally popular:

Beautiful CLI interfaces with

rich-click

About Me

Phil Ewels

seqera

Product Manager for
Open Source Software

nextflow

multiqc

wave

fusion

Co-founder

nf-core

rich-click

Part 1: Origins

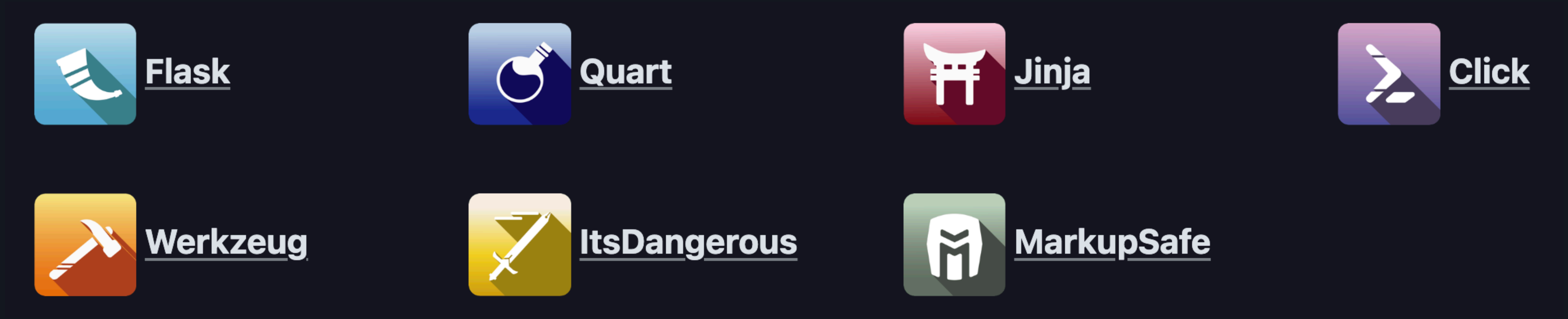
Part 2: Features

Part 3: Maintenance

Part 1: Origins

```
$ click_
```

pallets/click



Flask

Quart

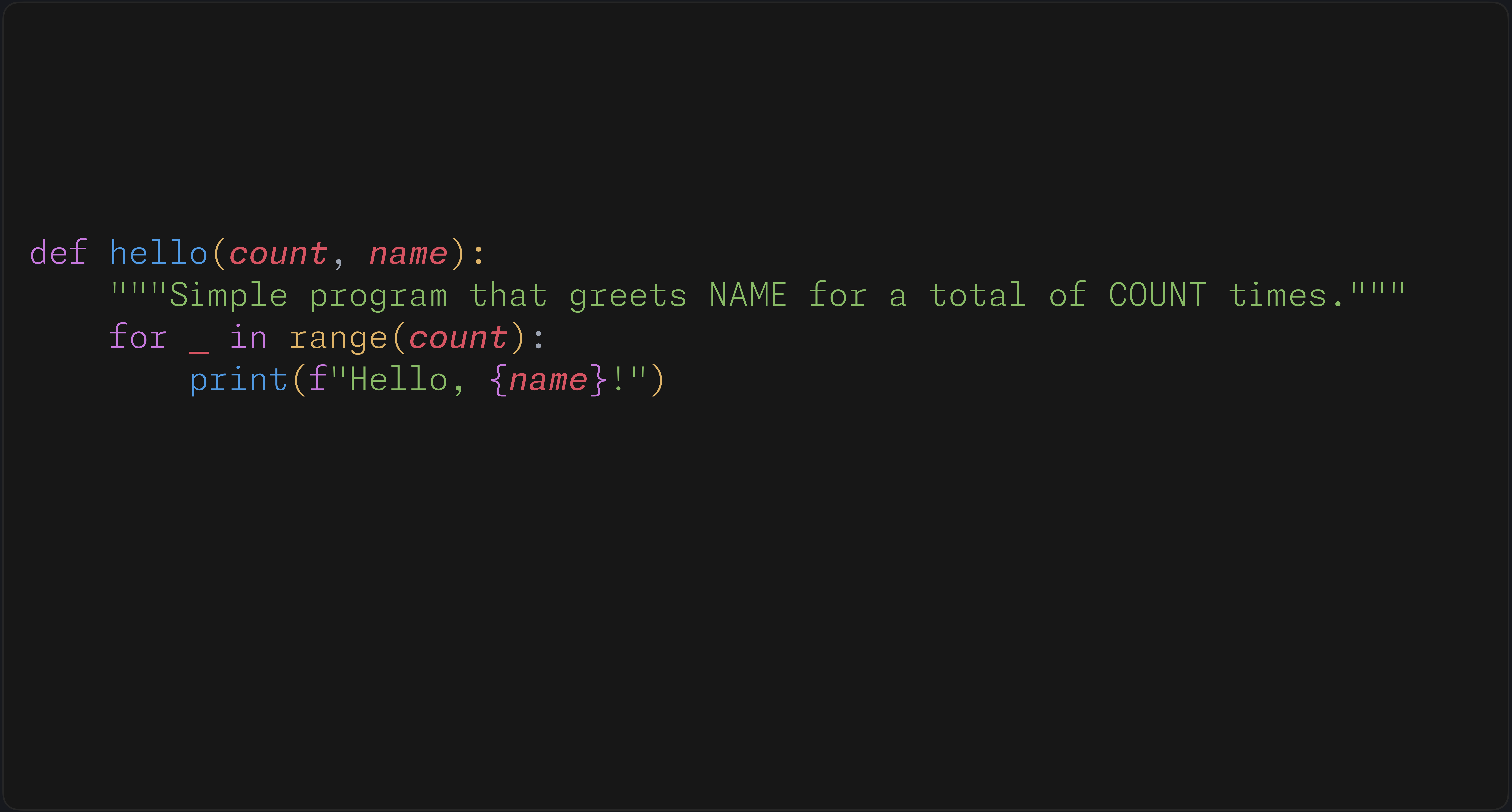
Jinja

Click

Werkzeug

ItsDangerous

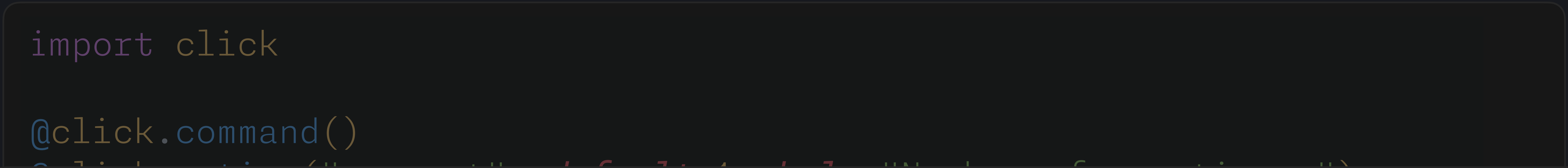
MarkupSafe



```python
def hello(count, name):
    """Simple program that greets NAME for a total of COUNT times."""
    for _ in range(count):
        print(f"Hello, {name}!")
```

```python
import click

@click.command()
@click.option("--count")
@click.option("--name")
def hello(count, name):
    """Simple program that greets NAME for a total of COUNT times."""
    for _ in range(count):
        print(f"Hello, {name}!")

if __name__ == "__main__":
    hello()
```

```python
import click

@click.command()
@click.option("--count", default=1, help="Number of greetings.")
@click.option("--name", required=True, help="The person to greet.")
def hello(count, name):
    """Simple program that greets NAME for a total of COUNT times."""
    for _ in range(count):
        print(f"Hello, {name}!")

if __name__ == "__main__":
    hello()
```



```python
import click

@click.command()
```

```
❯ python simple.py --help

Usage: simple.py [OPTIONS]

  Simple program that greets NAME for a total of COUNT times.

Options:
  --count INTEGER  Number of greetings.
  --name TEXT      The person to greet.  [required]
  --help           Show this message and exit.
```

```python
import click

@click.command()
```

```
❯ python simple.py

Usage: simple.py [OPTIONS]
Try 'simple.py --help' for help.

Error: Missing option '--name'.
```

```python
import click

@click.command()
```

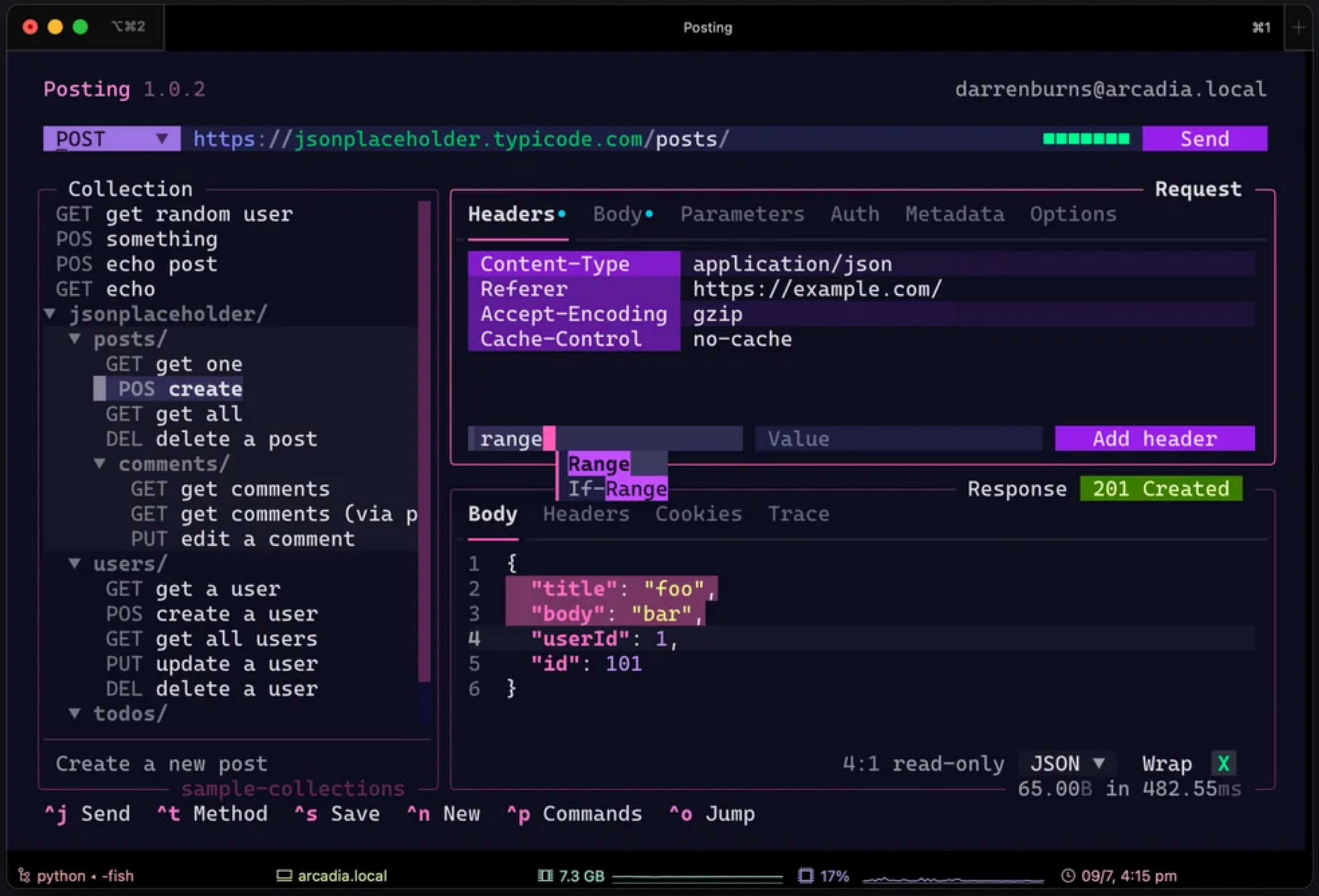
```
❯ python simple.py --name "PyCon Sweden" --count 3

Hello, PyCon Sweden!
Hello, PyCon Sweden!
Hello, PyCon Sweden!
```

rich_

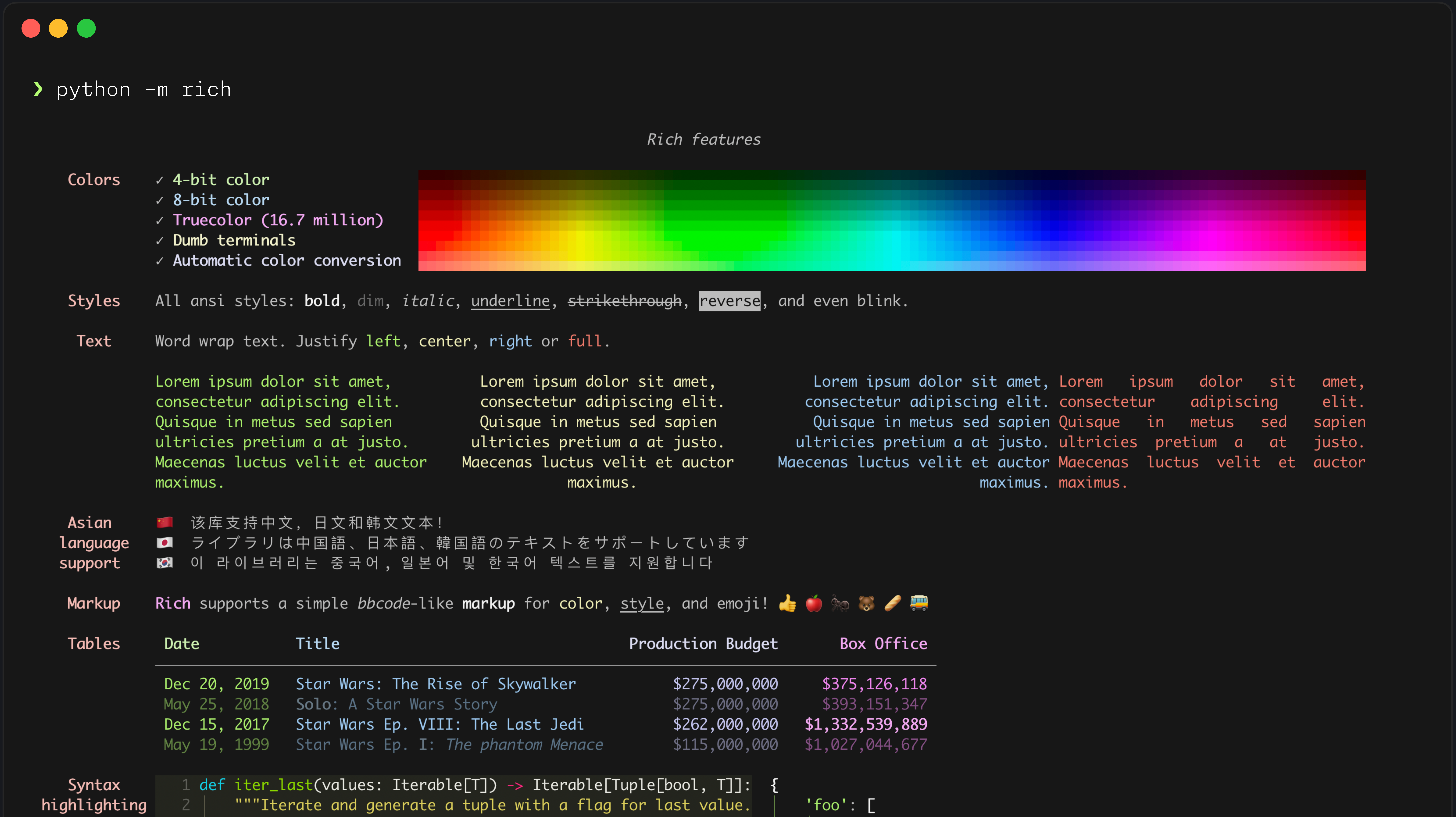
Will McGugan

Textualize

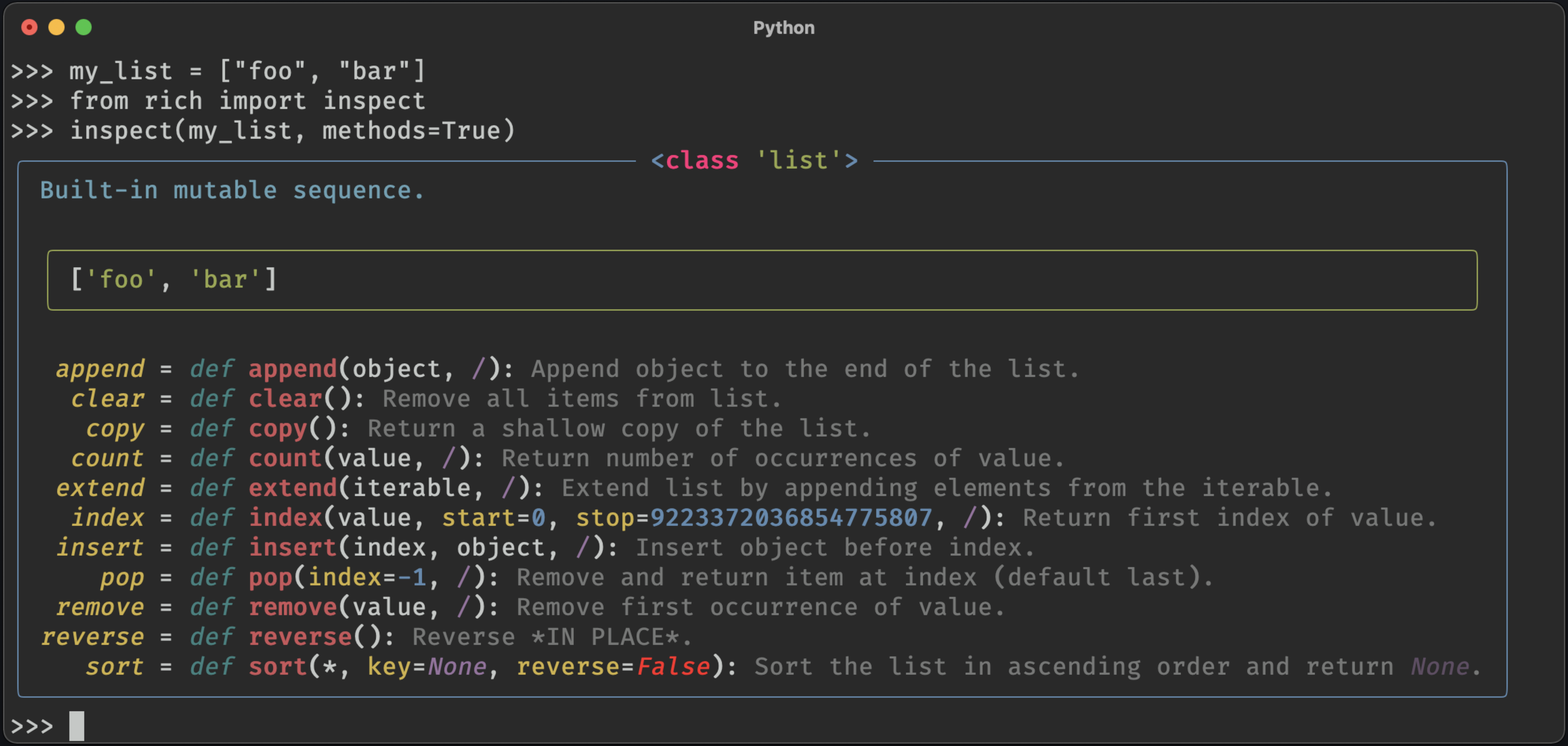
Textualize/rich



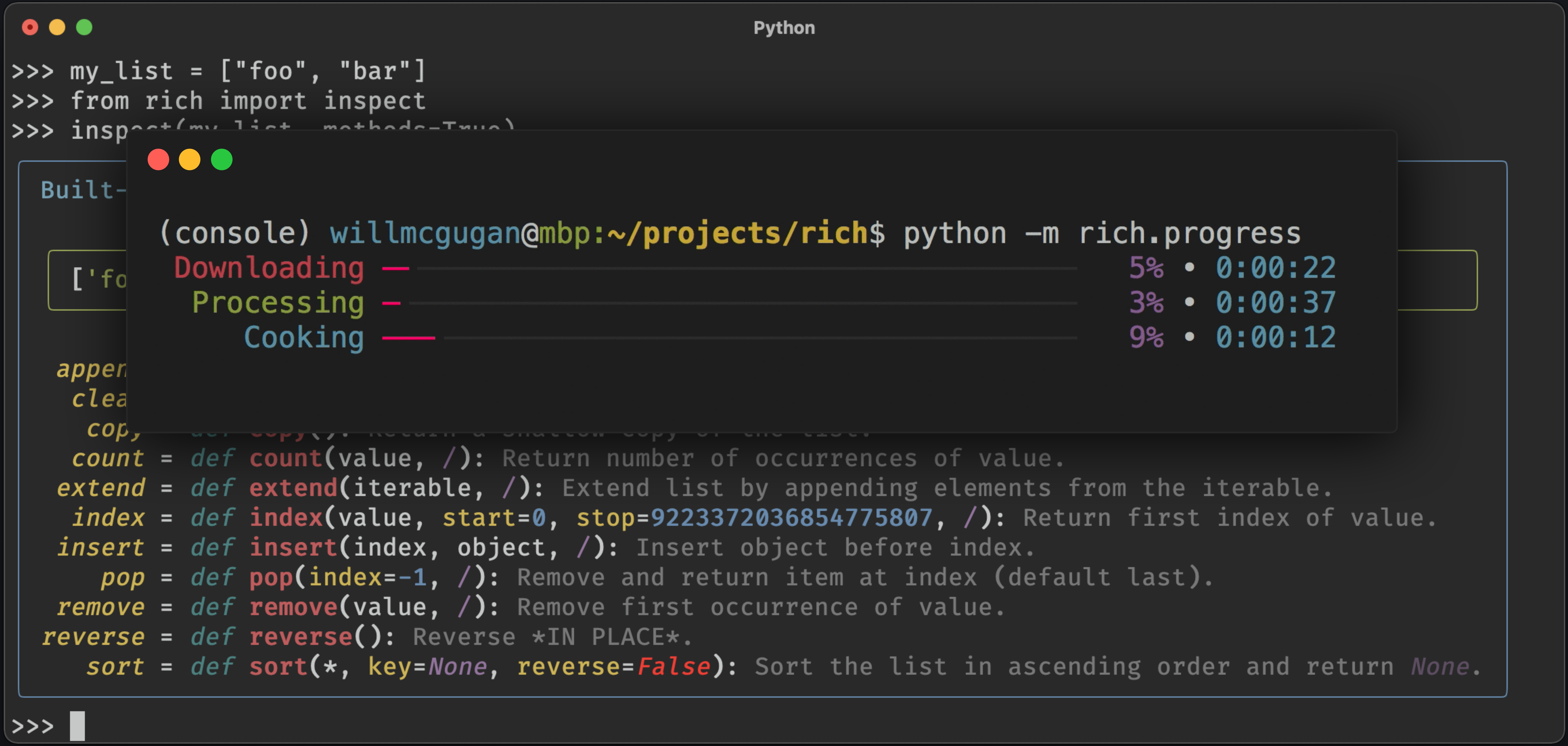
```
⌥⌘2                         Posting                              ⌘1

Posting 1.0.2                                       darrenburns@arcadia.local

POST  ▼ https://jsonplaceholder.typicode.com/posts/           ▮▮▮▮▮▮▮  Send

 Collection                        ┌ Headers•  Body•  Parameters  Auth  Metadata  Options   Request
 GET get random user               │
 POS something                     │  Content-Type     application/json
 POS echo post                     │  Referer          https://example.com/
 GET echo                          │  Accept-Encoding  gzip
▼ jsonplaceholder/                 │  Cache-Control    no-cache
  ▼ posts/                         │
    GET get one                    │
  ▌ POS create                     │  range            Value              Add header
    GET get all                    │                 Range
    DEL delete a post              │                 If-Range
   ▼ comments/                     │                                Response  201 Created
      GET get comments             │  Body  Headers  Cookies  Trace
      GET get comments (via p      │
      PUT edit a comment           │  {
  ▼ users/                         │    "title": "foo",
    GET get a user                 │    "body": "bar",
    POS create a user              │    "userId": 1,
    GET get all users              │    "id": 101
    PUT update a user              │  }
    DEL delete a user              │
  ▼ todos/                         │
                                   │
 Create a new post                 │            4:1 read-only  JSON ▼  Wrap X
        sample-collections         │                         65.00B in 482.55ms
^j Send  ^t Method  ^s Save  ^n New  ^p Commands  ^o Jump

python • -fish        arcadia.local        7.3 GB        17%        09/7, 4:15 pm
```



```
❯ python -m rich

                                   Rich features

  Colors    ✓ 4-bit color
            ✓ 8-bit color
            ✓ Truecolor (16.7 million)
            ✓ Dumb terminals
            ✓ Automatic color conversion

  Styles    All ansi styles: bold, dim, italic, underline, ~~strikethrough~~, reverse, and even blink.

   Text     Word wrap text. Justify left, center, right or full.

            Lorem ipsum dolor sit amet,         Lorem ipsum dolor sit amet,         Lorem ipsum dolor sit amet, Lorem   ipsum   dolor   sit   amet,
            consectetur adipiscing elit.        consectetur adipiscing elit.       consectetur adipiscing elit. consectetur    adipiscing     elit.
            Quisque in metus sed sapien         Quisque in metus sed sapien         Quisque in metus sed sapien Quisque   in   metus   sed   sapien
            ultricies pretium a at justo.      ultricies pretium a at justo.      ultricies pretium a at justo. ultricies  pretium  a  at  justo.
            Maecenas luctus velit et auctor   Maecenas luctus velit et auctor   Maecenas luctus velit et auctor Maecenas  luctus  velit  et  auctor
            maximus.                                    maximus.                                       maximus. maximus.

  Asian     🇨🇳  该库支持中文，日文和韩文文本！
 language   🇯🇵  ライブラリは中国語、日本語、韓国語のテキストをサポートしています
 support    🇰🇷  이 라이브러리는 중국어, 일본어 및 한국어 텍스트를 지원합니다

  Markup    Rich supports a simple bbcode-like markup for color, style, and emoji! 👍 🍎 🐜 🐻 🥖 🚌

  Tables     Date           Title                                Production Budget       Box Office
            ─────────────────────────────────────────────────────────────────────────────────────────
             Dec 20, 2019   Star Wars: The Rise of Skywalker          $275,000,000     $375,126,118
             May 25, 2018   Solo: A Star Wars Story                   $275,000,000     $393,151,347
             Dec 15, 2017   Star Wars Ep. VIII: The Last Jedi         $262,000,000   $1,332,539,889
             May 19, 1999   Star Wars Ep. I: The phantom Menace       $115,000,000   $1,027,044,677

  Syntax    def iter_last(values: Iterable[T]) -> Iterable[Tuple[bool, T]]:  │  {
highlighting    """Iterate and generate a tuple with a flag for last value.  │      'foo': [
```



```
>>> my_list = ["foo", "bar"]
>>> from rich import inspect
>>> inspect(my_list, methods=True)
                                        <class 'list'>
 Built-in mutable sequence.


  ['foo', 'bar']


  append = def append(object, /): Append object to the end of the list.
   clear = def clear(): Remove all items from list.
    copy = def copy(): Return a shallow copy of the list.
   count = def count(value, /): Return number of occurrences of value.
  extend = def extend(iterable, /): Extend list by appending elements from the iterable.
   index = def index(value, start=0, stop=9223372036854775807, /): Return first index of value.
  insert = def insert(index, object, /): Insert object before index.
     pop = def pop(index=-1, /): Remove and return item at index (default last).
  remove = def remove(value, /): Remove first occurrence of value.
 reverse = def reverse(): Reverse *IN PLACE*.
    sort = def sort(*, key=None, reverse=False): Sort the list in ascending order and return None.
>>> 
```



```
>>> my_list = ["foo", "bar"]
>>> from rich import inspect
>>> inspect(my_list, methods=True)

 Built-

  ['fo

  appen
   clea
    copy = def copy(): Return a shallow copy of the list.
   count = def count(value, /): Return number of occurrences of value.
  extend = def extend(iterable, /): Extend list by appending elements from the iterable.
   index = def index(value, start=0, stop=9223372036854775807, /): Return first index of value.
  insert = def insert(index, object, /): Insert object before index.
     pop = def pop(index=-1, /): Remove and return item at index (default last).
  remove = def remove(value, /): Remove first occurrence of value.
 reverse = def reverse(): Reverse *IN PLACE*.
    sort = def sort(*, key=None, reverse=False): Sort the list in ascending order and return None.

>>>
```

```
(console) willmcgugan@mbp:~/projects/rich$ python -m rich.progress
 Downloading ━━                                                   5% • 0:00:22
  Processing ━━                                                   3% • 0:00:37
     Cooking ━━━━                                                 9% • 0:00:12
```

Python

```
>>> my_list = ["foo", "bar"]
>>> from rich import inspect
>>> insp
Built-
['fo
appen
clea
copy
count = def
extend = def
index = def
insert = def
pop = def
remove = def
reverse = def
sort = def
>>> 
```

```
(console) willmcgugan@mbp:~/projects/rich$ python -m rich.progress
 Downloading ━                                         5% • 0:00:22
  Processing
```

```
[14:49:49] INFO      Server starting...                                          logging.py:62
           INFO      Listening on http://127.0.0.1:8080                          logging.py:63
[14:49:50] INFO      GET /index.html 200 1298                                    logging.py:66
           INFO      GET /imgs/backgrounds/back1.jpg 200 54386                   logging.py:67
           INFO      GET /css/styles.css 200 54386                               logging.py:68
           WARNING   GET /favicon.ico 404 242                                    logging.py:69
[14:49:51] DEBUG     JSONRPC request                                             logging.py:72
                     --> {'version': '1.1', 'method': 'confirmFruitPurchase', 'params': [['apple',
                     'orange', 'mangoes', 'pomelo'], 1.123], 'id': '194521489'}
                     <-- {'version': '1.1', 'result': True, 'error': None, 'id': '194521489'}
           ERROR     Unable to find 'pomelo' in database!                        logging.py:82
           INFO      POST /jsonrpc/ 200 65532                                    logging.py:83
           INFO      POST /admin/ 401 42234                                      logging.py:84
           WARNING   password was rejected for admin site.                       logging.py:85
           ERROR     An error of some kind occurred!                             logging.py:89
                     Traceback (most recent call last):
                       File "/Users/willmcgugan/projects/rich/rich/logging.py", line 87, in
                     <module>
                         1 / 0
                     ZeroDivisionError: division by zero
[14:49:52] CRITICAL  Out of memory!                                              logging.py:91
           INFO      Server exited with code=-1                                  logging.py:92
```

```
>>> my_list = ["foo", "bar"]
>>> from rich import inspect
>>> inspect(my_list, methods=T

  Built-

   ['fo

   appen
    clea
     cop
    count = def
   extend = def
    index = def
   insert = def
      pop = def
   remove = def
  reverse = def
     sort = def

>>> 
```

```
(console) willmcgu
 Downloading ━
  Processing
```

```
[14:49:49] INFO
           INFO
[14:49:50] INFO
           INFO
           INFO
           WARNI
[14:49:51] DEBUG



           ERROR
           INFO
           INFO
           WARNI
           ERROR



[14:49:52] CRITI
           INFO
```

-bash

```
Traceback (most recent call last)
exception.py:21 in divide_all


  def divide_all(divides: List[Tuple[float, float]]) -> None:
      """Do something impossible every day."""
      try:
          for number, divisor in divides:
❱             result = divide_by(number, divisor)
              console.print(f"{number} divided by {divisor} is {result}")
      except Exception:
          console.print_exception(extra_lines=5, show_locals=True)



 locals
 divides = [
               (1000, 200),
               (10000, 500),
               (0, 1000000),
               (3.1427, 2),
               (4294967296, 65536),
               (1, 0)
           ]
 divisor = 0
  number = 1
  result = 65536.0
```

```
>>> my_list = ["foo", "bar"]
>>> from rich import inspect
>>> insp
Built-
```

```
-bash
Traceback (most recent call last)
exception.py:21 in divide_all
```

```
❯ python simple.py --help

Usage: simple.py [OPTIONS]

  Simple program that greets NAME for a total of COUNT times.

Options:
  --count INTEGER  Number of greetings.
  --name TEXT      The person to greet.  [required]
  --help           Show this message and exit.
```

rich_ + $ click_

rich-click

Part 2: Features

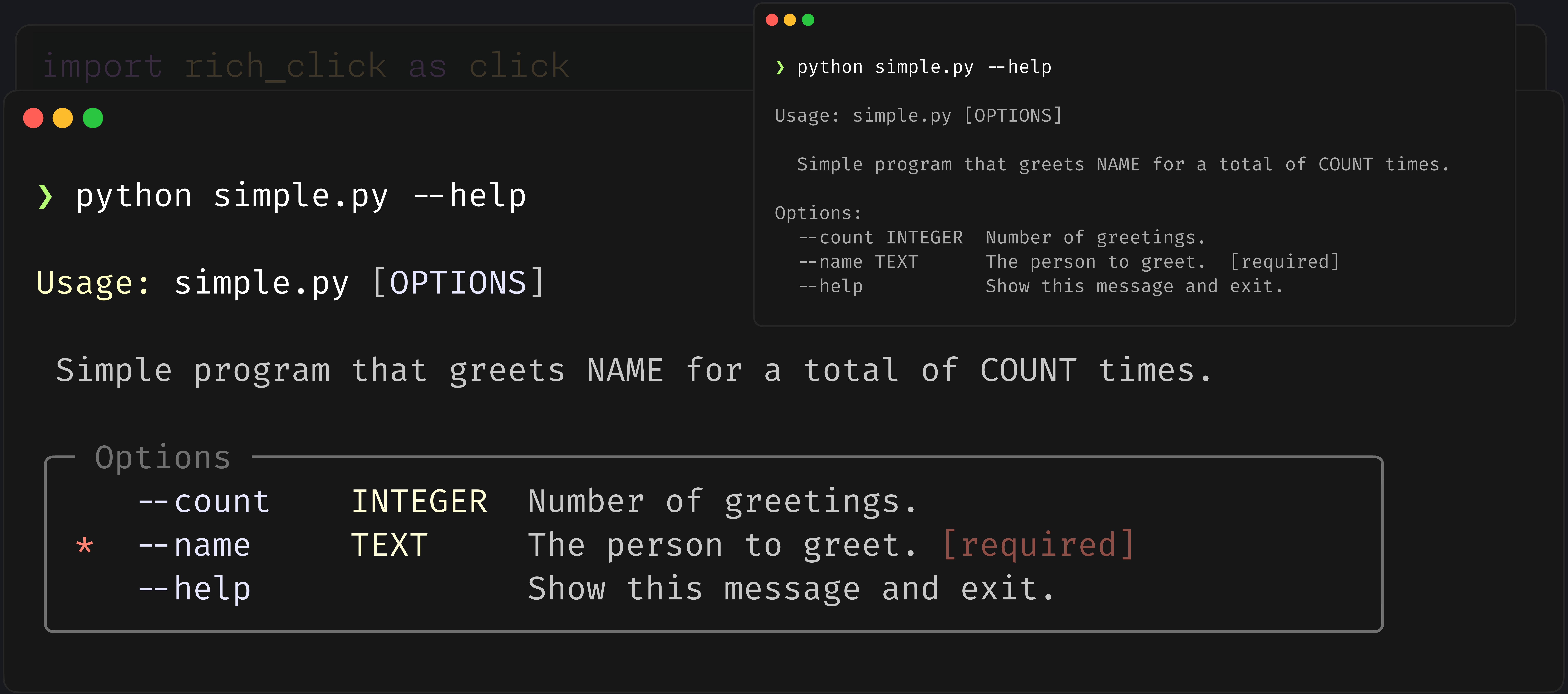
Usage

```python
import click

@click.command()
@click.option("--count", default=1, help="Number of greetings.")
@click.option("--name", required=True, help="The person to greet.")
def hello(count, name):
    """Simple program that greets NAME for a total of COUNT times."""
    for _ in range(count):
        print(f"Hello, {name}!")

if __name__ == "__main__":
    hello()
```

Usage

```python
import rich_click as click

@click.command()
@click.option("--count", default=1, help="Number of greetings.")
@click.option("--name", required=True, help="The person to greet.")
def hello(count, name):
    """Simple program that greets NAME for a total of COUNT times."""
    for _ in range(count):
        print(f"Hello, {name}!")

if __name__ == "__main__":
    hello()
```

Usage

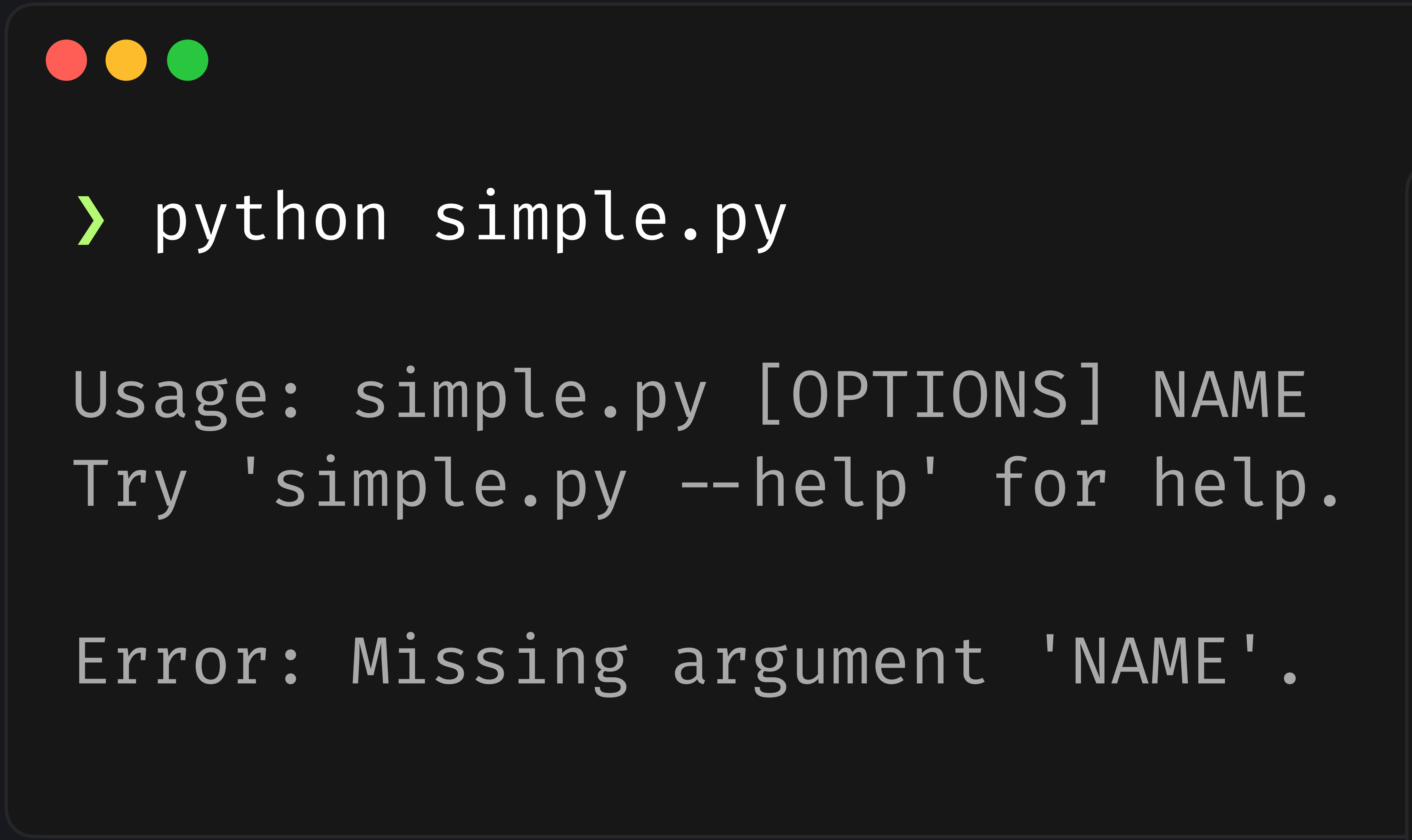


```
import rich_click as click
```

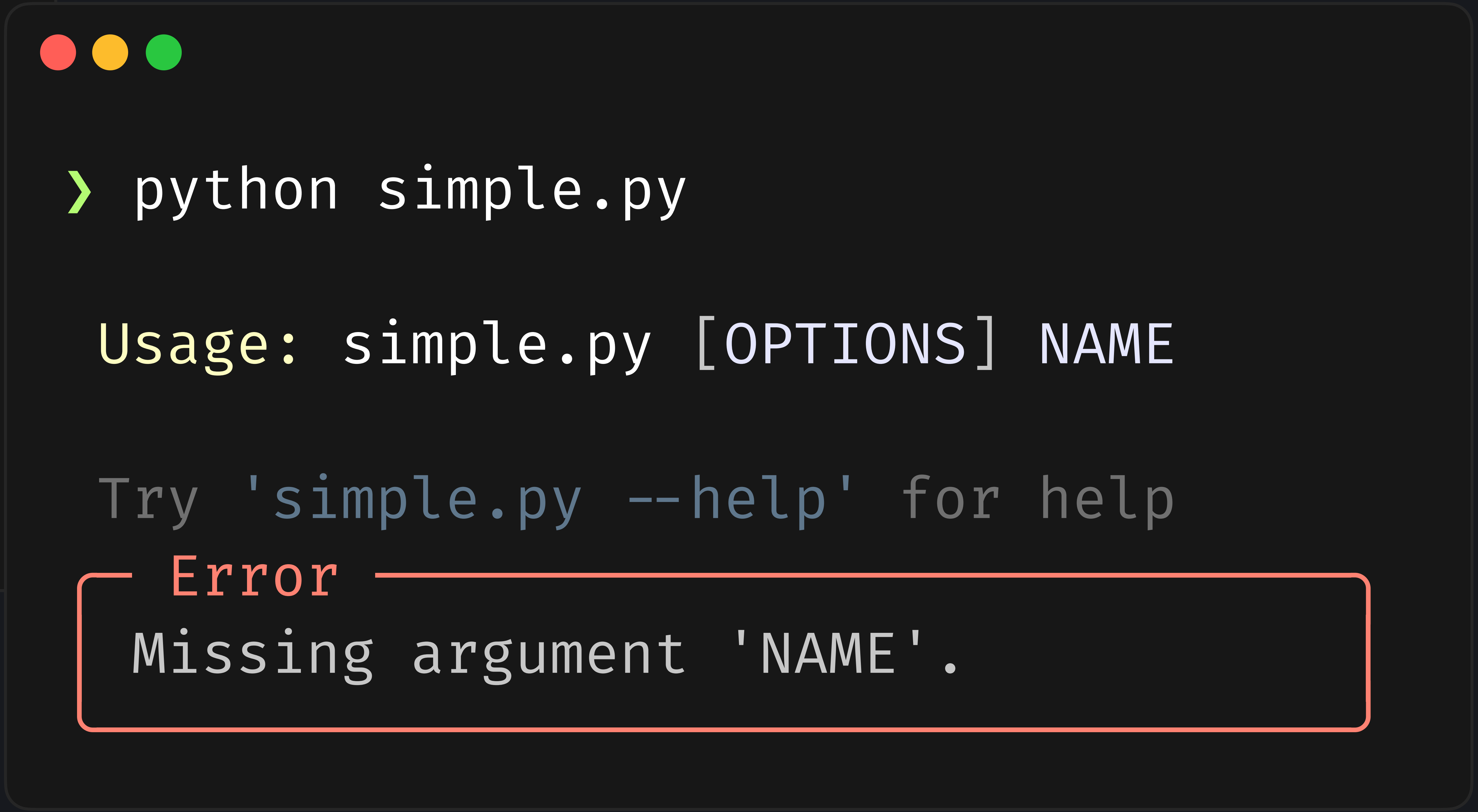
```
❯ python simple.py --help

Usage: simple.py [OPTIONS]

  Simple program that greets NAME for a total of COUNT times.

Options:
  --count INTEGER  Number of greetings.
  --name TEXT      The person to greet.  [required]
  --help           Show this message and exit.
```

```
❯ python simple.py --help

Usage: simple.py [OPTIONS]

 Simple program that greets NAME for a total of COUNT times.

╭─ Options ───────────────────────────────────────────────────────╮
│    --count    INTEGER  Number of greetings.                      │
│ *  --name     TEXT     The person to greet. [required]           │
│    --help              Show this message and exit.               │
╰──────────────────────────────────────────────────────────────────╯
```

Error Messages



```
❯ python simple.py

Usage: simple.py [OPTIONS] NAME
Try 'simple.py --help' for help.

Error: Missing argument 'NAME'.
```

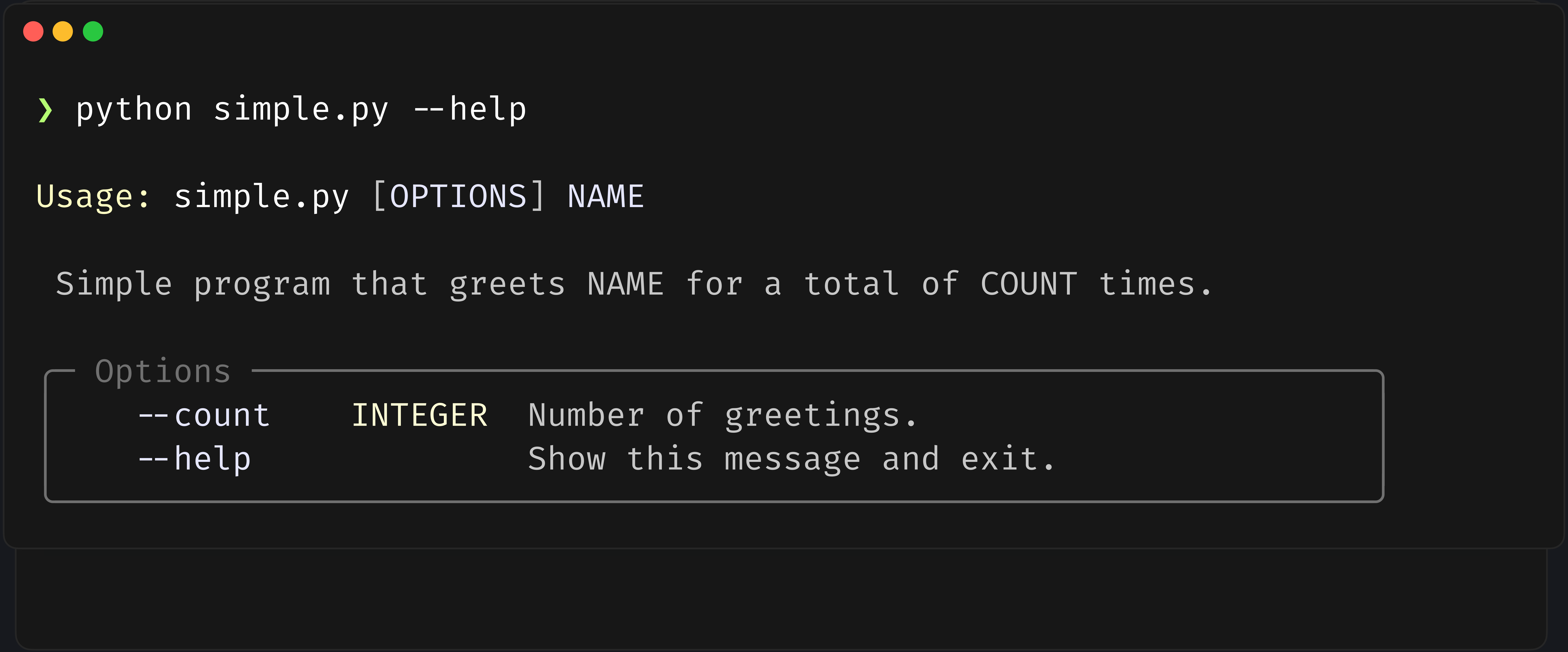


```
❯ python simple.py

 Usage: simple.py [OPTIONS] NAME

 Try 'simple.py --help' for help
╭─ Error ──────────────────────────────╮
│ Missing argument 'NAME'.             │
╰──────────────────────────────────────╯
```

Arguments

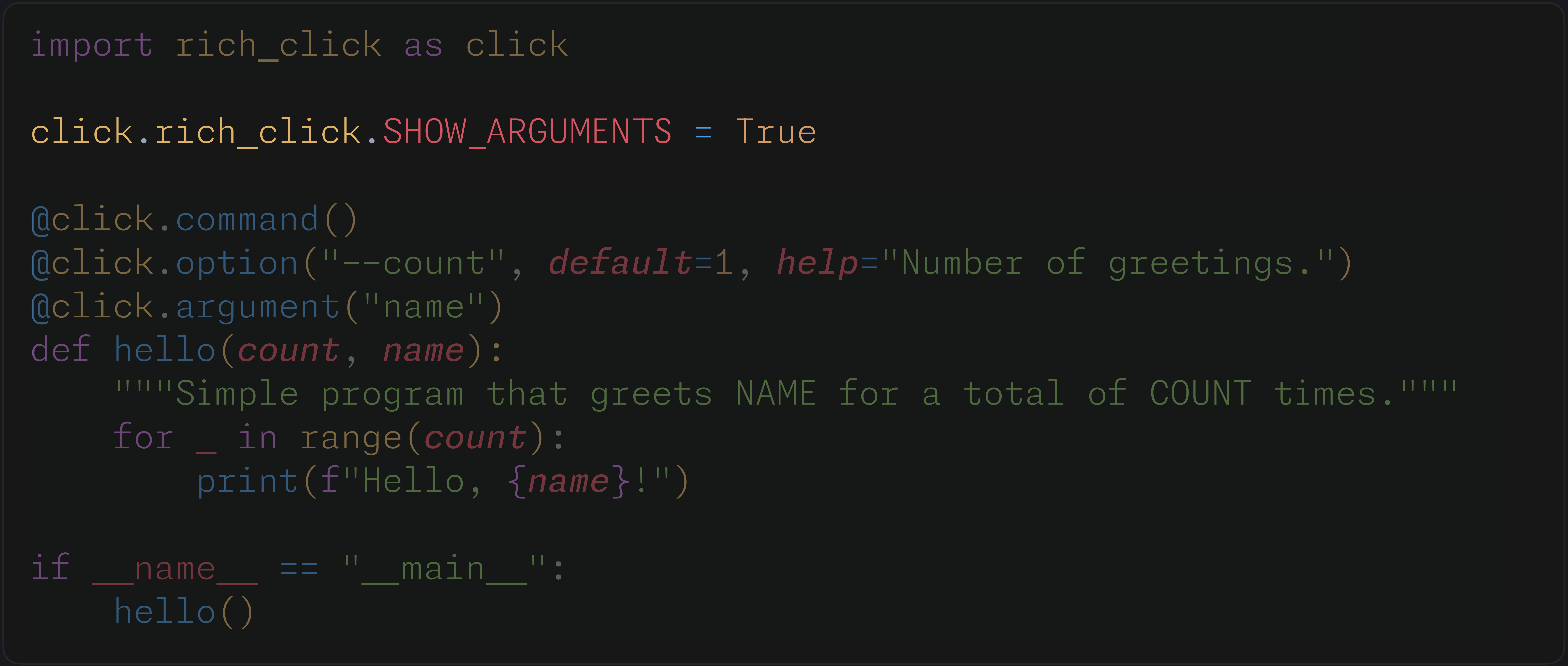
```python
import rich_click as click

@click.command()
@click.option("--count", default=1, help="Number of greetings.")
@click.argument("name")
def hello(count, name):
    """Simple program that greets NAME for a total of COUNT times."""
    for _ in range(count):
        print(f"Hello, {name}!")

if __name__ == "__main__":
    hello()
```

Arguments



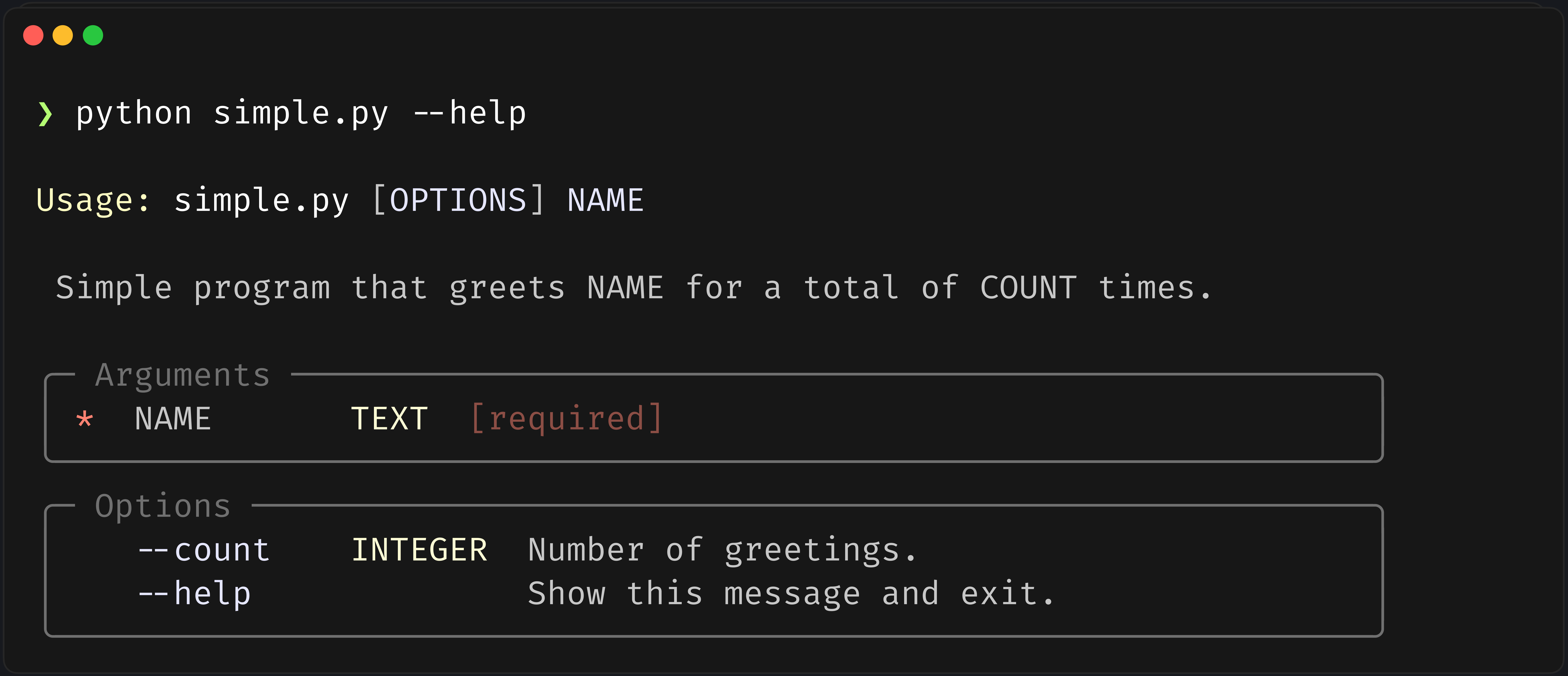
```
❯ python simple.py --help

Usage: simple.py [OPTIONS] NAME

 Simple program that greets NAME for a total of COUNT times.

╭─ Options ─────────────────────────────────────────────────────╮
│    --count     INTEGER  Number of greetings.                  │
│    --help               Show this message and exit.           │
╰───────────────────────────────────────────────────────────────╯
```

Arguments



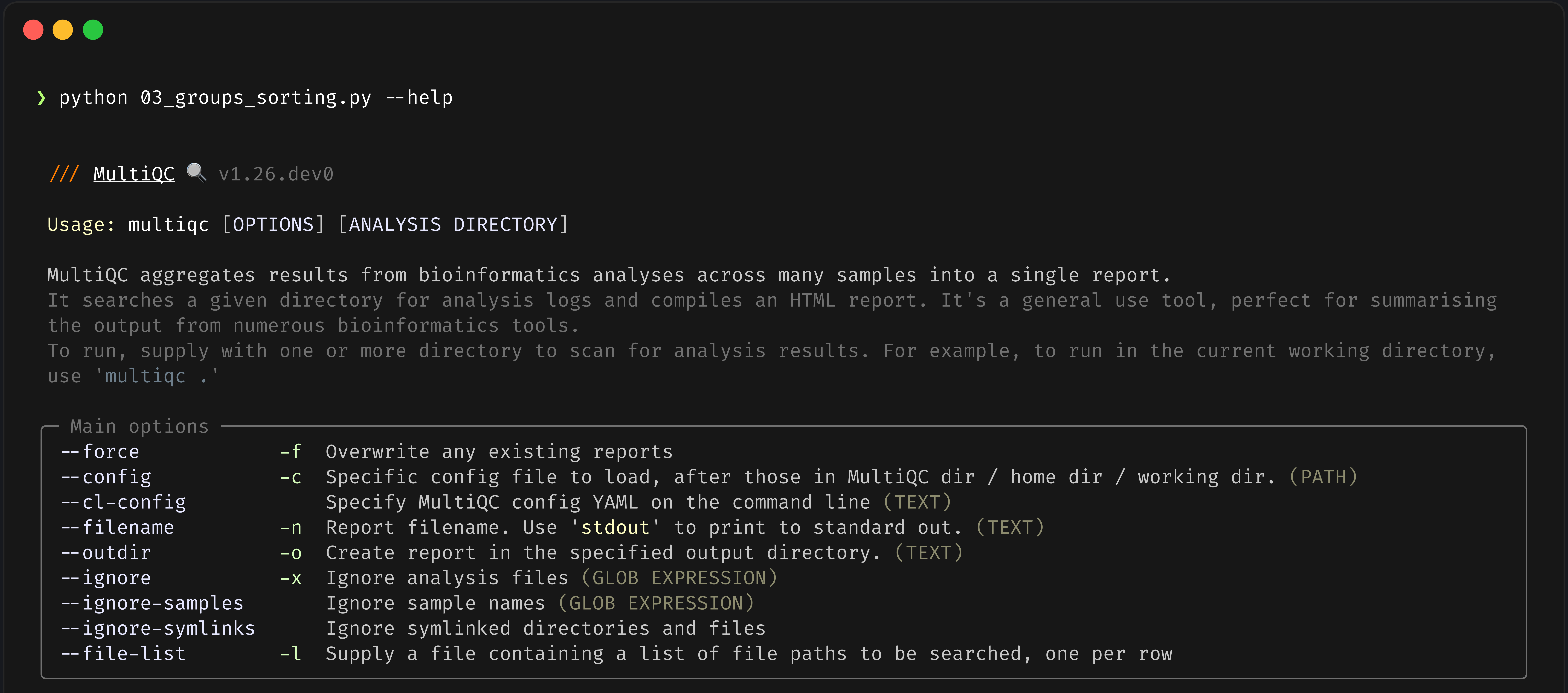
```python
import rich_click as click

click.rich_click.SHOW_ARGUMENTS = True

@click.command()
@click.option("--count", default=1, help="Number of greetings.")
@click.argument("name")
def hello(count, name):
    """Simple program that greets NAME for a total of COUNT times."""
    for _ in range(count):
        print(f"Hello, {name}!")

if __name__ == "__main__":
    hello()
```

Arguments



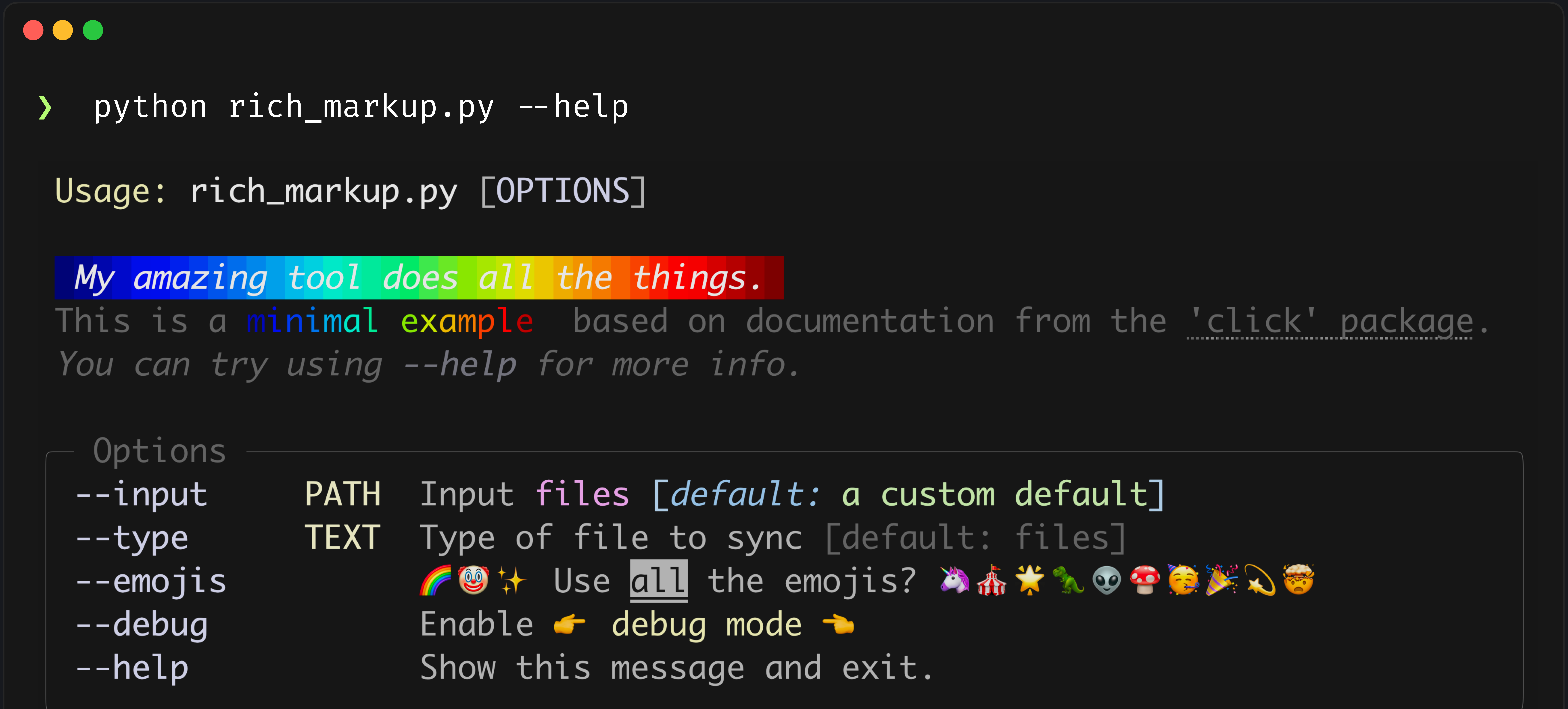
```
❯ python simple.py --help

Usage: simple.py [OPTIONS] NAME

 Simple program that greets NAME for a total of COUNT times.

╭─ Arguments ────────────────────────────────────────────────────╮
│ *   NAME        TEXT  [required]                               │
╰────────────────────────────────────────────────────────────────╯
╭─ Options ──────────────────────────────────────────────────────╮
│     --count     INTEGER  Number of greetings.                  │
│     --help               Show this message and exit.           │
╰────────────────────────────────────────────────────────────────╯
```

Groups + Sorting



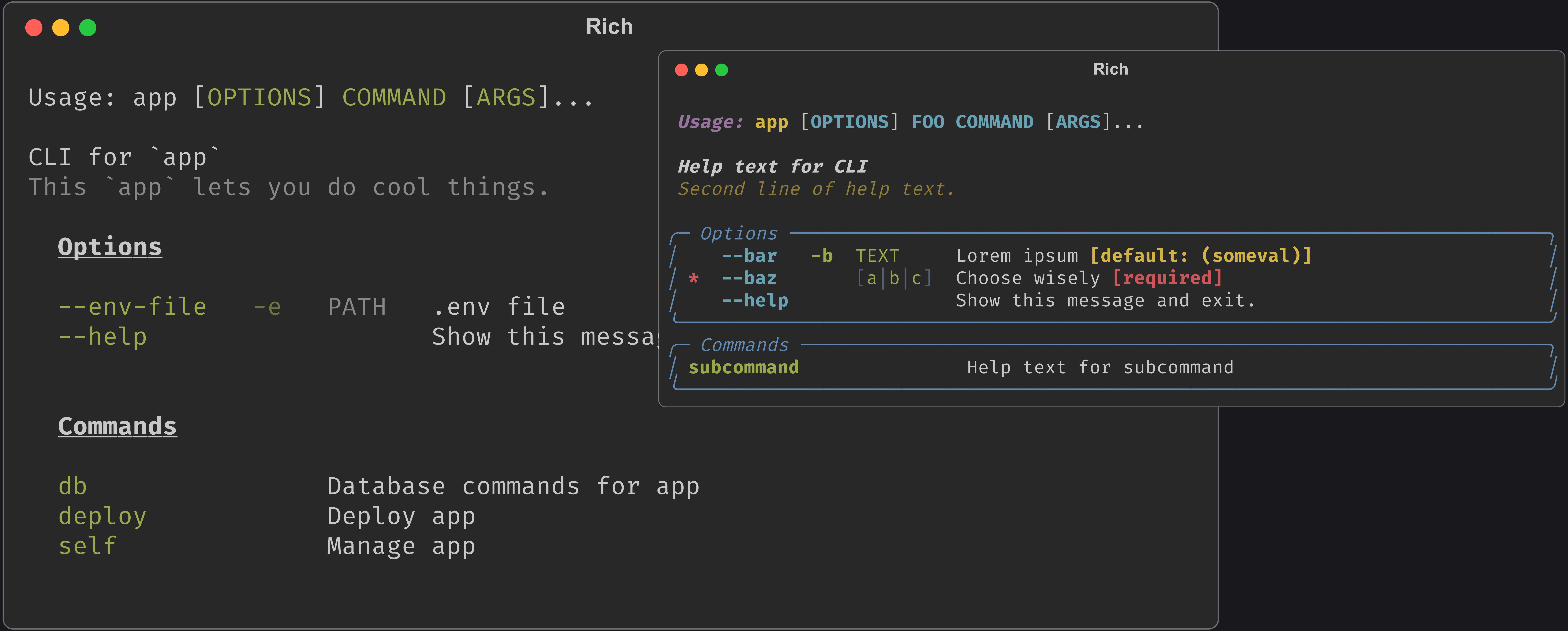
```
❯ python 03_groups_sorting.py --help


 /// MultiQC 🔍 v1.26.dev0

 Usage: multiqc [OPTIONS] [ANALYSIS DIRECTORY]

 MultiQC aggregates results from bioinformatics analyses across many samples into a single report.
 It searches a given directory for analysis logs and compiles an HTML report. It's a general use tool, perfect for summarising
 the output from numerous bioinformatics tools.
 To run, supply with one or more directory to scan for analysis results. For example, to run in the current working directory,
 use 'multiqc .'

╭─ Main options ──────────────────────────────────────────────────────────────────────────────────────────────────────────────╮
│ --force               -f   Overwrite any existing reports                                                                 │
│ --config              -c   Specific config file to load, after those in MultiQC dir / home dir / working dir. (PATH)      │
│ --cl-config                Specify MultiQC config YAML on the command line (TEXT)                                          │
│ --filename            -n   Report filename. Use 'stdout' to print to standard out. (TEXT)                                  │
│ --outdir              -o   Create report in the specified output directory. (TEXT)                                         │
│ --ignore              -x   Ignore analysis files (GLOB EXPRESSION)                                                         │
│ --ignore-samples           Ignore sample names (GLOB EXPRESSION)                                                           │
│ --ignore-symlinks          Ignore symlinked directories and files                                                          │
│ --file-list           -l   Supply a file containing a list of file paths to be searched, one per row                       │
╰─────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────╯
```

Text markup



```
❯  python rich_markup.py --help

 Usage: rich_markup.py [OPTIONS]

 My amazing tool does all the things.
 This is a minimal example  based on documentation from the 'click' package.
 You can try using --help for more info.

╭─ Options ────────────────────────────────────────────────────────────────────╮
│ --input      PATH  Input files [default: a custom default]                   │
│ --type       TEXT  Type of file to sync [default: files]                     │
│ --emojis           🌈🤡✨ Use all the emojis? 🦄🎪🌟🦖👽🍄🥳🎉💫🤯          │
│ --debug            Enable 👉 debug mode 👈                                   │
│ --help             Show this message and exit.                               │
```

Configurable



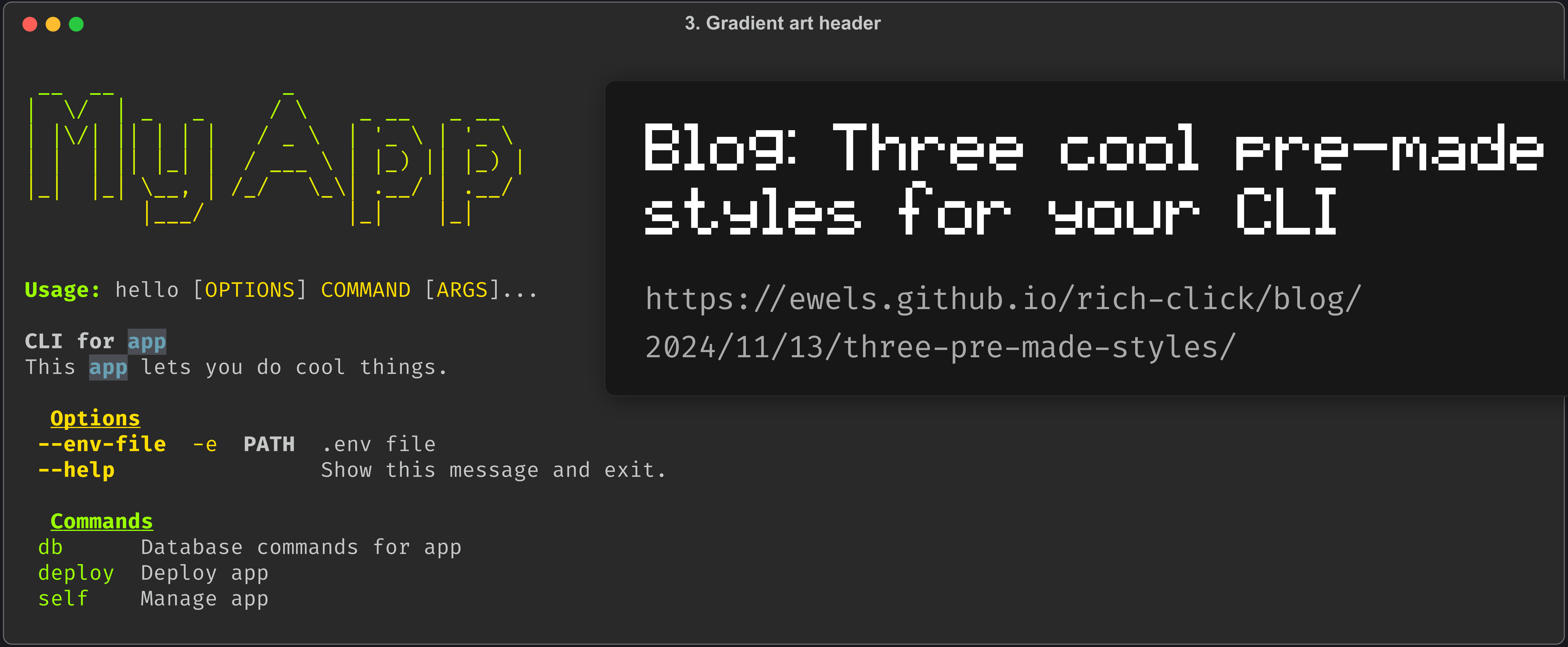
Rich

```
Usage: app [OPTIONS] COMMAND [ARGS]...

CLI for `app`
This `app` lets you do cool things.

  Options

  --env-file   -e   PATH   .env file
  --help                   Show this messa


  Commands

  db                 Database commands for app
  deploy             Deploy app
  self               Manage app
```

Rich

```
Usage: app [OPTIONS] FOO COMMAND [ARGS]...

Help text for CLI
Second line of help text.

╭─ Options ──────────────────────────────────────────────────────────────╮
│    --bar   -b  TEXT      Lorem ipsum [default: (someval)]               │
│ *  --baz       [a|b|c]   Choose wisely [required]                       │
│    --help                Show this message and exit.                    │
╰─────────────────────────────────────────────────────────────────────────╯
╭─ Commands ─────────────────────────────────────────────────────────────╮
│ subcommand                 Help text for subcommand                      │
╰─────────────────────────────────────────────────────────────────────────╯
```

Configurable



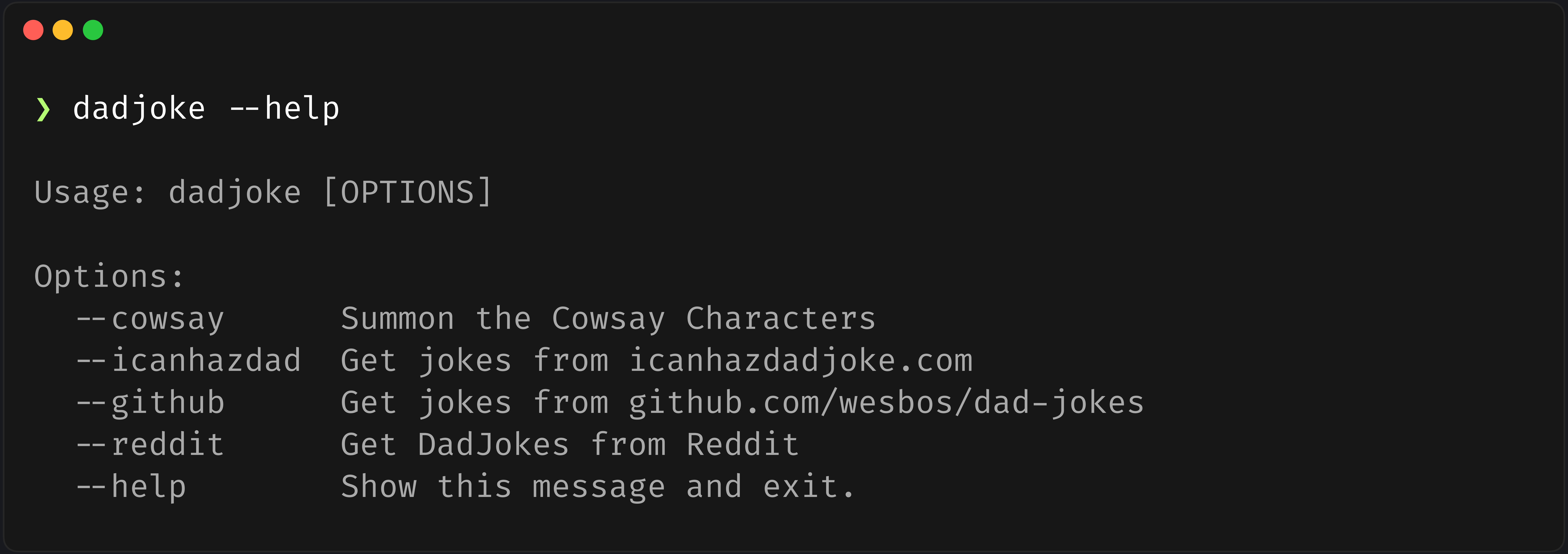
3. Gradient art header

```
 __  __             _
|  \/  | _   _     / \    _ __   _ __
| |\/| || | | |   / _ \  | '_ \ | '_ \
| |  | || |_| |  / ___ \ | |_) || |_) |
|_|  |_| \__, | /_/   \_\| .__/ | .__/
         |___/           |_|    |_|

Usage: hello [OPTIONS] COMMAND [ARGS]...

CLI for app
This app lets you do cool things.

  Options
 --env-file  -e  PATH  .env file
 --help                Show this message and exit.

  Commands
 db      Database commands for app
 deploy  Deploy app
 self    Manage app
```

Blog: Three cool pre-made styles for your CLI

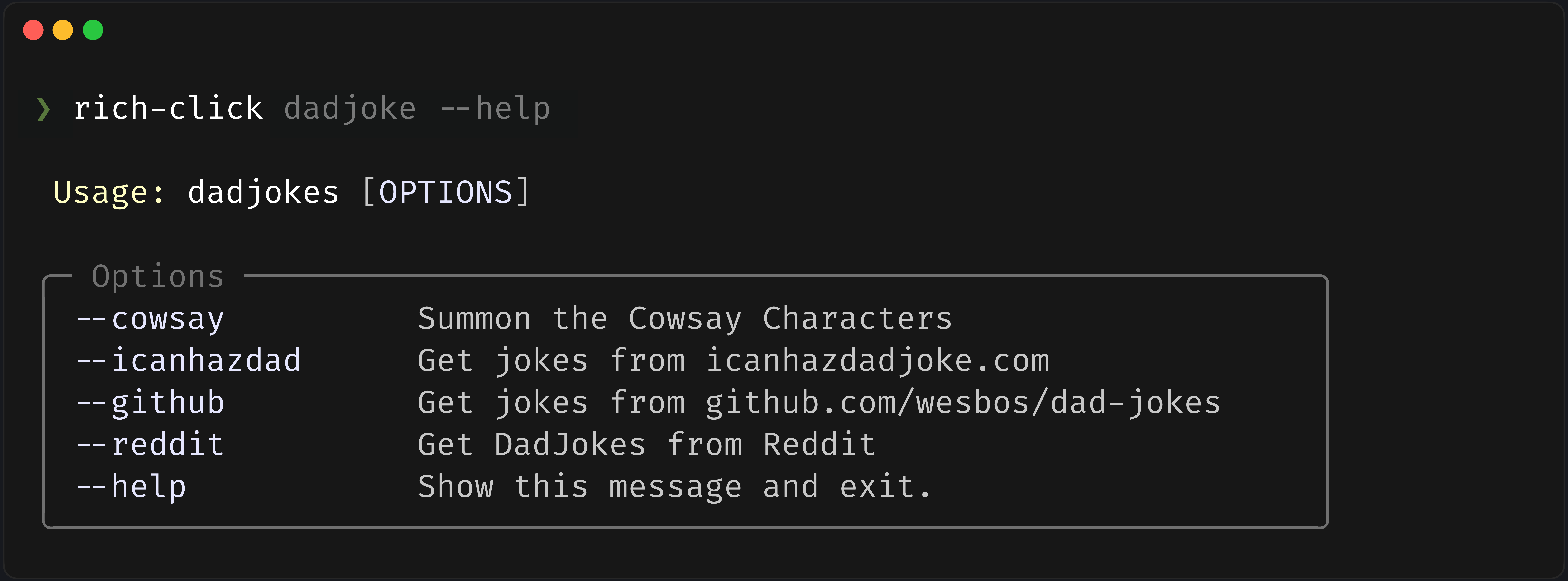
https://ewels.github.io/rich-click/blog/2024/11/13/three-pre-made-styles/

rich-click CLI



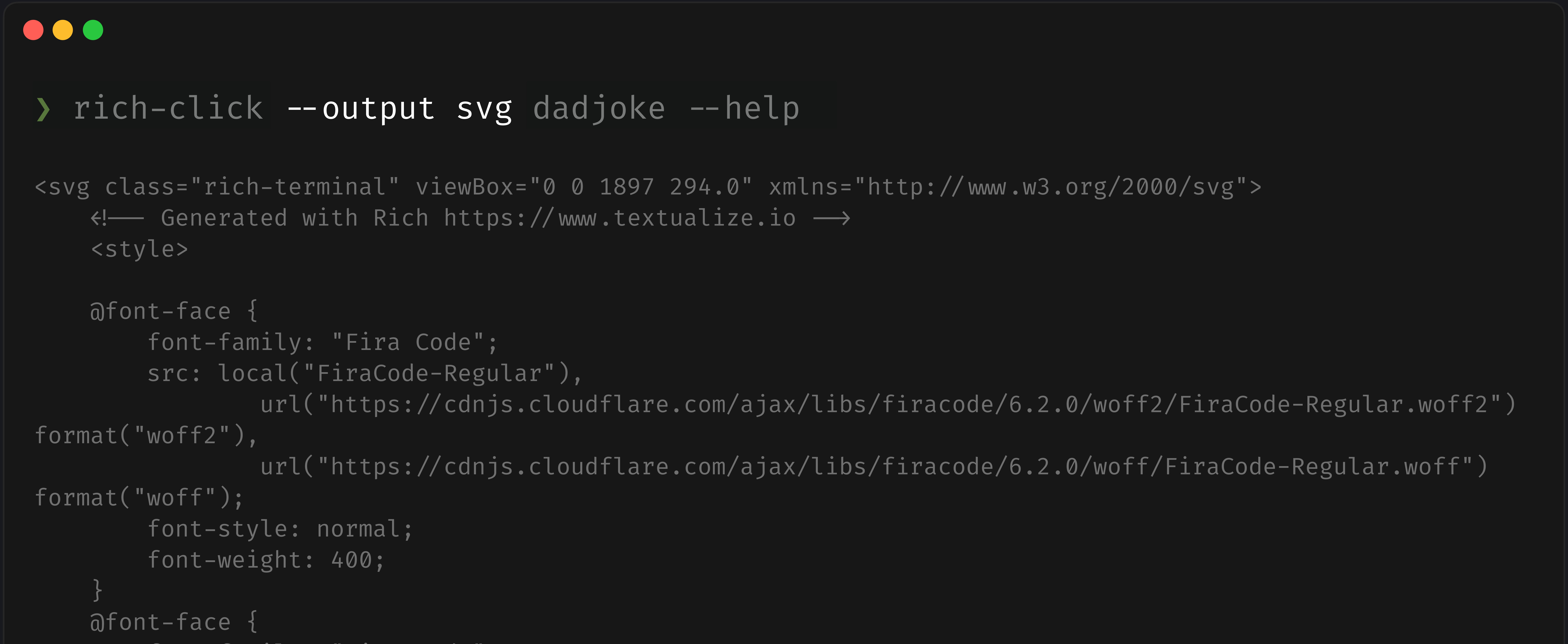
```
❯ dadjoke --help

Usage: dadjoke [OPTIONS]

Options:
  --cowsay       Summon the Cowsay Characters
  --icanhazdad   Get jokes from icanhazdadjoke.com
  --github       Get jokes from github.com/wesbos/dad-jokes
  --reddit       Get DadJokes from Reddit
  --help         Show this message and exit.
```

rich-click CLI



```
❯ rich-click dadjoke --help

 Usage: dadjokes [OPTIONS]

╭─ Options ──────────────────────────────────────────────────────────────╮
│ --cowsay           Summon the Cowsay Characters                         │
│ --icanhazdad       Get jokes from icanhazdadjoke.com                    │
│ --github           Get jokes from github.com/wesbos/dad-jokes           │
│ --reddit           Get DadJokes from Reddit                             │
│ --help             Show this message and exit.                          │
╰────────────────────────────────────────────────────────────────────────╯
```

Export SVGs



```
❯ rich-click --output svg dadjoke --help

<svg class="rich-terminal" viewBox="0 0 1897 294.0" xmlns="http://www.w3.org/2000/svg">
    <!-- Generated with Rich https://www.textualize.io -->
    <style>

    @font-face {
        font-family: "Fira Code";
        src: local("FiraCode-Regular"),
                url("https://cdnjs.cloudflare.com/ajax/libs/firacode/6.2.0/woff2/FiraCode-Regular.woff2")
format("woff2"),
                url("https://cdnjs.cloudflare.com/ajax/libs/firacode/6.2.0/woff/FiraCode-Regular.woff")
format("woff");
        font-style: normal;
        font-weight: 400;
    }
    @font-face {
```

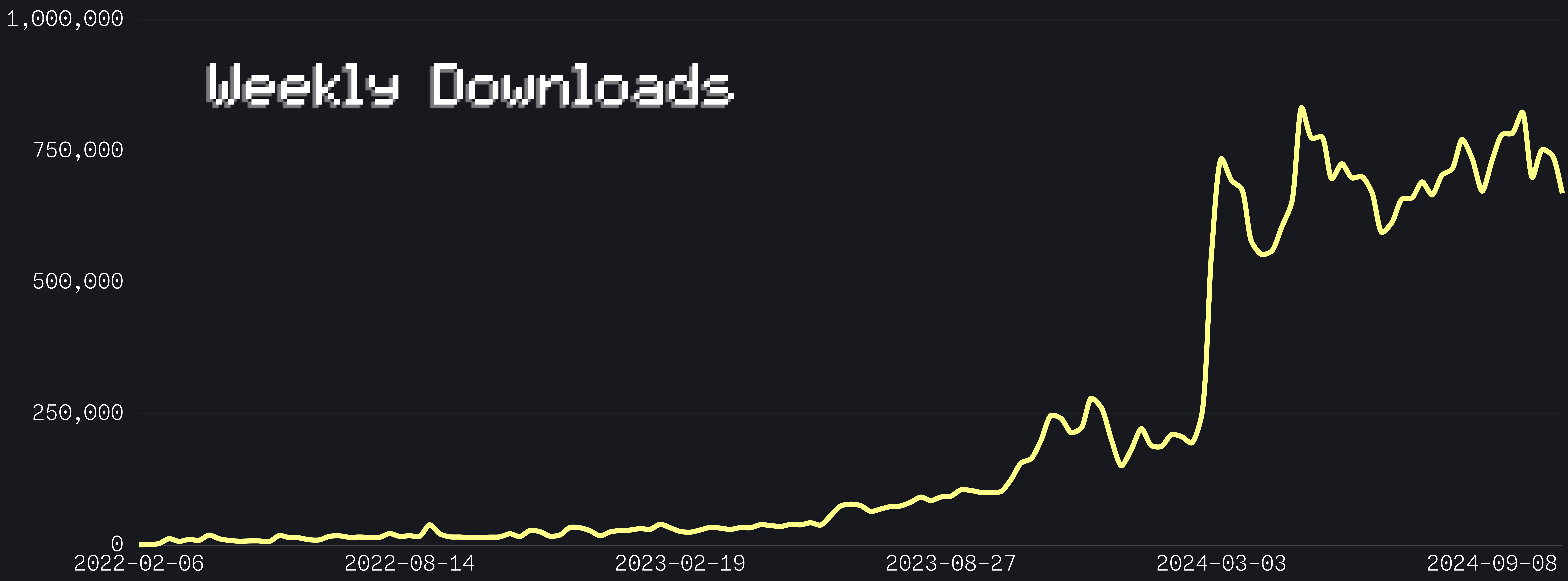
Export SVGs

```
❯ rich-click --output svg dadjoke --help
```

```
Usage: dadjokes [OPTIONS]

╭─ Options ──────────────────────────────────────────────────────────────────╮
│ --cowsay            Summon the Cowsay Characters                           │
│ --icanhazdad        Get jokes from icanhazdadjoke.com                      │
│ --github            Get jokes from github.com/wesbos/dad-jokes             │
│ --reddit            Get DadJokes from Reddit                               │
│ --help              Show this message and exit.                            │
╰────────────────────────────────────────────────────────────────────────────╯
```

```
    }
    @font-face {
```

Part 3: Maintenance



Weekly Downloads

Downloads

by period

181K	827K	3.3M	33M
last day	last week	last month	total

Typer

https://typer.tiangolo.com/

Sebastián Ramírez

Hey Phil! I hope all is good. I'm Sebastián, the author of Typer, based on Click. 😁

I want to find a way to have your work with rich-click directly in Typer, without having to import a different Typer class from rich-click. 🚀

I think it would make most sense to keep things separate from rich-click.

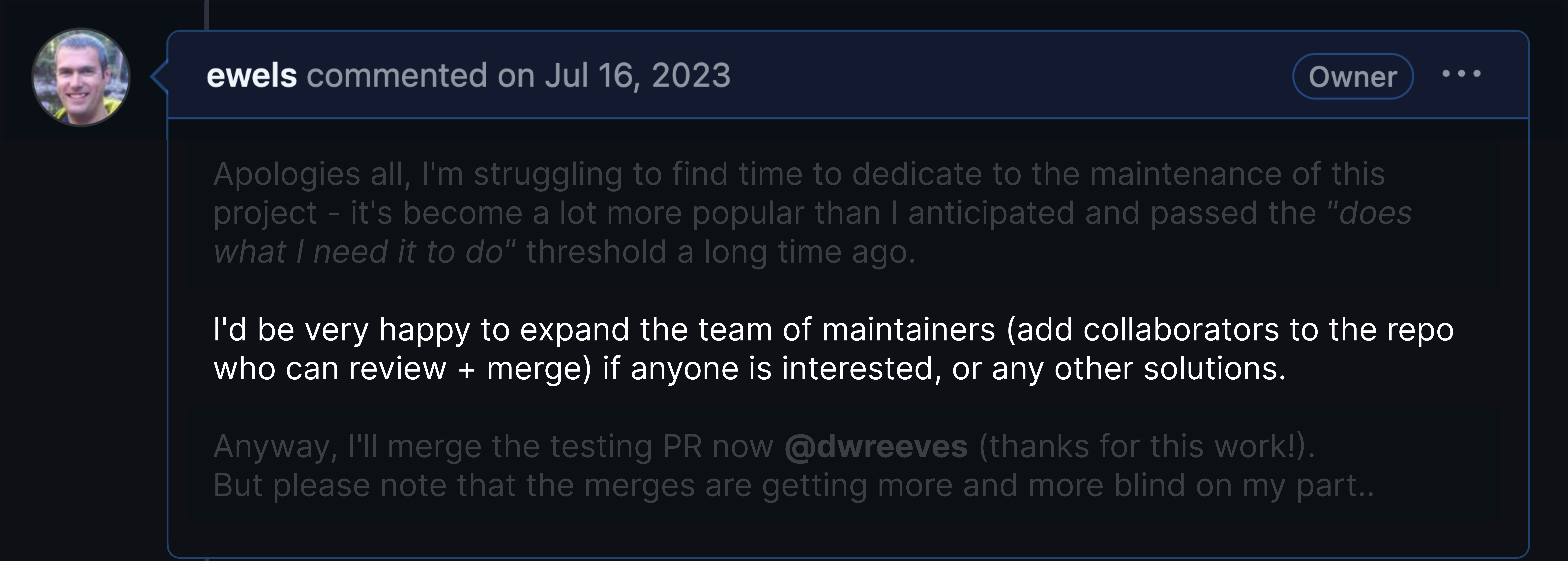
I can then redirect people your way instead, and it makes life easier for me for maintenance as well.

I want to find a way to have your work with rich-click directly in Typer, without having to import a different Typer class from rich-click. 🚀

I think it would make most sense to keep things separate from rich-click.

I can then redirect people your way instead, and it makes life easier for me for maintenance as well.

Wow, you were fast! Ok I'm doubly glad I said you should go ahead without waiting for me now 😂



ewels commented on Jul 16, 2023

Apologies all, I'm struggling to find time to dedicate to the maintenance of this project - it's become a lot more popular than I anticipated and passed the *"does what I need it to do"* threshold a long time ago.

I'd be very happy to expand the team of maintainers (add collaborators to the repo who can review + merge) if anyone is interested, or any other solutions.

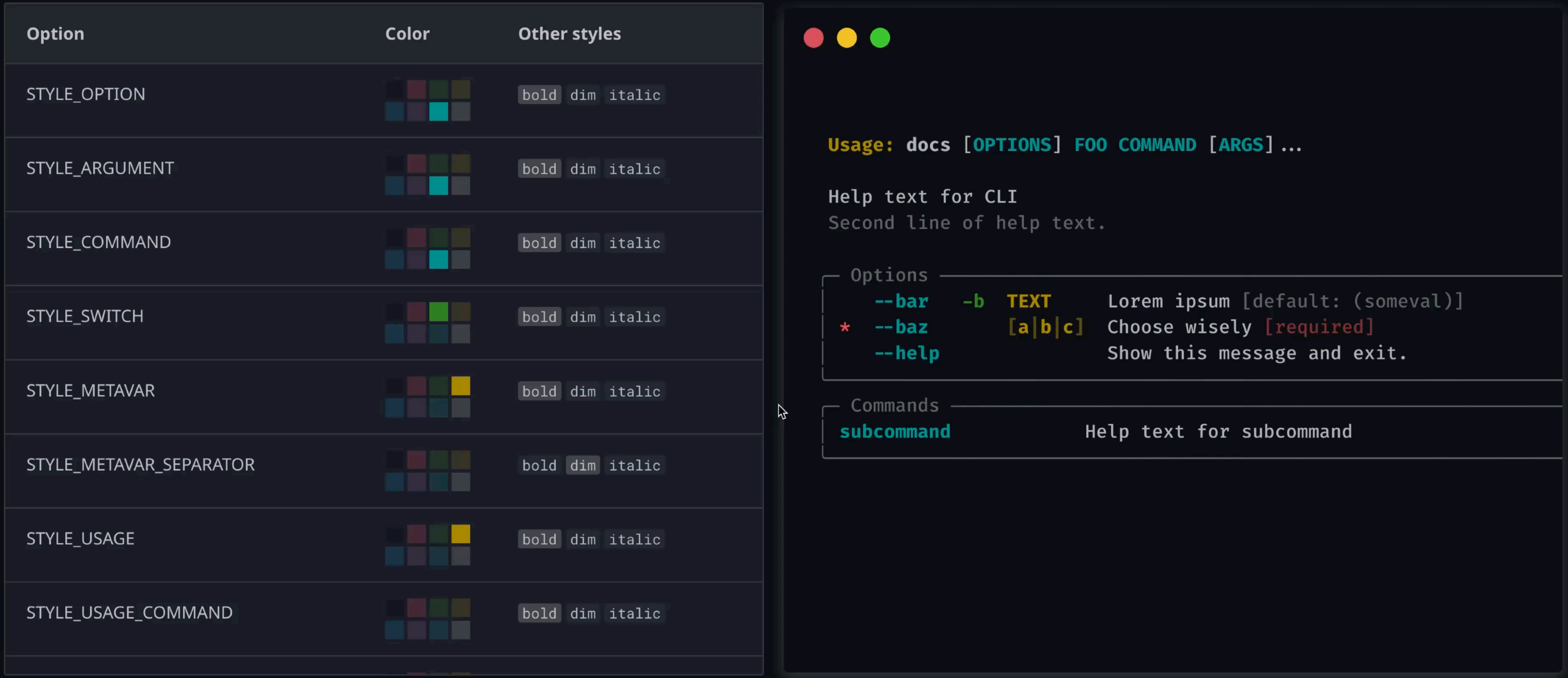
Anyway, I'll merge the testing PR now **@dwreeves** (thanks for this work!).
But please note that the merges are getting more and more blind on my part..

Daniel Reeves

Daniel Reeves

- Add tests!
- Add type hinting
- Performance improvements
- New docs website and blog
- New package logo
- Internal refactoring
- Easier configuration
- Live style editor

Live style editor



Option	Color	Other styles
STYLE_OPTION	🟦	**bold** dim italic
STYLE_ARGUMENT	🟦	**bold** dim italic
STYLE_COMMAND	🟦	**bold** dim italic
STYLE_SWITCH	🟩	**bold** dim italic
STYLE_METAVAR	🟨	**bold** dim italic
STYLE_METAVAR_SEPARATOR		bold **dim** italic
STYLE_USAGE	🟨	**bold** dim italic
STYLE_USAGE_COMMAND		**bold** dim italic

```
Usage: docs [OPTIONS] FOO COMMAND [ARGS]...

Help text for CLI
Second line of help text.

╭─ Options ──────────────────────────────────────────────────────
│   --bar    -b  TEXT     Lorem ipsum [default: (someval)]
│ * --baz        [a|b|c]  Choose wisely [required]
│   --help                Show this message and exit.
╰────────────────────────────────────────────────────────────────
╭─ Commands ─────────────────────────────────────────────────────
│ subcommand           Help text for subcommand
╰────────────────────────────────────────────────────────────────
```

Side note:

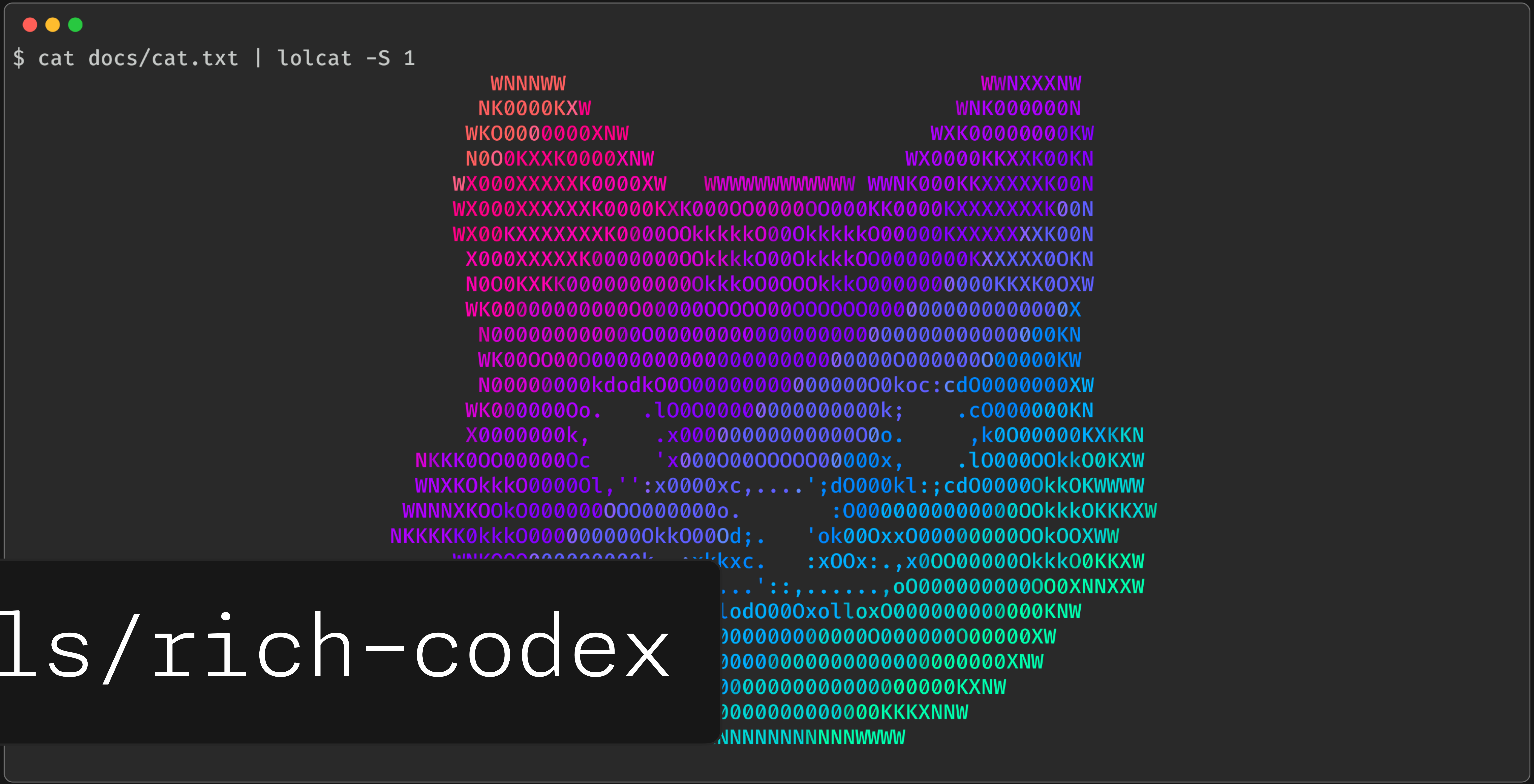
rich-codex

```markdown
# My markdown docs

Here's an example CLI output:

![`cat docs/cat.txt | lolcat -S 1`](docs/img/cat.png)
```

```
# My markdow

Here's an ex

![`cat docs/
```

# My markdown docs

Here's an example CLI output:



```
$ cat docs/cat.txt | lolcat -S 1
                                      WNNNWW                                 WWNXXXNW
                                     NK0000KXW                             WNK000000N
                                    WKO0000000XNW                        WXK00000000KW
                                    N0O0KXXK0000XNW                    WX0000KKXXK00KN
                                   WX000XXXXXK0000XW   WWWWWWWWWWWW WWNK000KKXXXXXK00N
                                   WX000XXXXXXK0000KXK0000000000O000KK0000KXXXXXXXK00N
                                   WX00KXXXXXXXXK00000OkkkkO000OkkkkO00000KXXXXXXXK00N
                                    X000XXXXXK00000000OkkkO000OkkkO000000000KXXXXX0OKN
                                    N0O0KXKK0000000000OkkkO000OkkkO00000000000KKXK0OXW
                                    WK0000000000000000000000000000000000000000000000X
                                     N000000000000000000000000000000000000000000000KN
                                     WK00000000000000000000000000000000000000000000KW
                                     N00000000kdodkO00000000000000000Okoc:cdO0000000XW
                                    WK0000000Oo.  .lO0000000000000000k;    .cO000000KN
                                    X0000000k,     .x000000000000000Oo.     ,k0O00000KXKKN
                                NKKK0O0000000Oc    'x0000000000000000x,    .lO0000OkkO0KXW
                                WNXKOkkkO00000Ol,'':x0000xc,....';dO000kl:;cdO00000OkkOKWWWW
                               WNNNXKOOkO000000O00000000o.       :O00000000000000OkkkOKKKXW
                              NKKKKK0kkkO000000000Okk0000d;.   'ok000xxO00000000OOkOOXWW
                                                        kxc.   :xOOx:.,x0O000000OkkkO0KKXW
                                                        ...'::,......,oO0000000000O0XNNXXW
                                                        lodO000xollox0000000000000KNW
                                                        0000000000000000000000000XW
                                                        00000000000000000000000XNW
                                                        0000000000000000000KXNW
                                                        0000000000000KKKXNNW
                                                        NNNNNNNNNNNWWWW
```

ewels/rich-codex

Advice

Generalise your solutions

Publish your tools

Ask for help if you need it

rich-click

```
❯ pip install rich-click
```

python™ Package Index

ewels/rich-click

ewels.github.io/rich-click

Thank you!

```
❯ pip install rich-click
```

Phil Ewels

seqera

phil.ewels@seqera.io

ewels

tallphil

@ewels.bsky.social

phil.ewels.co.uk

Will McGugan

Sebástian Ramírez

Daniel Reeves