



`import bacon`

self.intro()

Ivana Kellyerova

 @aerinthenniel  gitlab.com/jenx

 backend developer at iMobility

 dabbling in Python for 5+ years

 tea drinker

 game player

 prog metal <3

The Python standard library
can do almost **anything**.



Accoutrements Yodelling Pickle

by Archie McPhee



708 customer reviews | 60 answered questions

List Price: \$14.99

Price: **\$11.92**

You Save: **\$3.07 (20%)**

- Hours of mindless entertainment
- Batteries included
- Great gift for the person who has everything except a yodeling pickle

[Used & new \(22\) from \\$11.92 Details](#)

<https://www.amazon.com/dp/B0010VS078/>

The Python standard library
can do almost **anything**.

What's new in Python 3.0?

```
>>>
```



```
>>> import bacon
```



...awesomeness.

CC BY-NC 3.0 US

abstrusegoose.com/81

`import bacon`

 @aerinthenniel

Python 3.7.3

<https://github.com/python/cpython/tree/3.7>

<https://docs.python.org/3/library/>



the standard library

800+

contributors

10 000+

commits

also
the standard library

639

XXX_s

639

XXX_s

XXX obviously wrong, see #3232

`cpython/Lib/encodings/idna.py, line 194`

639

XXX_s

```
# XXX obviously wrong, see #3232
```

```
cpython/Lib/encodings/idna.py, line 194
```

```
# XXX: shouldn't this code be removed, not commented out?
```

```
cpython/Lib/imaplib.py, line 423
```

639

XXXs

```
# XXX obviously wrong, see #3232
```

```
cpython/Lib/encodings/idna.py, line 194
```

```
# XXX: shouldn't this code be removed, not commented out?
```

```
cpython/Lib/imaplib.py, line 423
```

```
# XXX yuck
```

```
cpython/Lib/importlib/_bootstrap_external, line 411
```

639

XXX_s

80

hack_s

639

XXX_s

80

hack_s

Can't think of a clean solution, so we hack away

`cpython/Lib/idlelib/paragraph.py, line 164`

639

XXX_s

80

hack_s

```
# Can't think of a clean solution, so we hack away
```

```
cpython/Lib/idlelib/paragraph.py, line 164
```

```
# Now for some serious hackery
```

```
cpython/Lib/email/_header_value_parser.py, line 2283
```

639

XXXs

80

hack_s

74

TODO_s

639

XXX_s

80

hack_s

74

TODO_s

```
# TODO: replace the frame hack if a blessed way  
# to know the calling module is ever developed
```

`cpython/Lib/enum.py, line 426`

639

XXX_s

80

hack_s

74

TODO_s

61

FIXME_s


```
self.assertEqual(binop.col_offset, 3) # FIXME: this is wrong
self.assertEqual(binop.left.col_offset, 3) # FIXME: this is wrong
self.assertEqual(binop.right.col_offset, 7) # FIXME: this is wrong
self.assertEqual(binop.col_offset, 3) # FIXME: this is wrong
self.assertEqual(binop.left.col_offset, 3) # FIXME: this is wrong
self.assertEqual(binop.right.col_offset, 7) # FIXME: this is wrong
```

cpython/Lib/test/test_fstring.py, lines 250, 251, 252, 262, 263, 264

61

FIXME_s

639

XXX_s

80

hack_s

74

TODO_s

61

FIXME_s

639

XXX_s

80

hack_s

74

TODO_s

61

FIXME_s

8

!!!_s

```
##### vererbung!!!!!!!!!!!!!!!!!!!!!!
```

cpython/Lib/turtle.py, line 3188

61

FIXME_s

8

!!!_s

639

XXX_s

80

hack_s

74

TODO_s

61

FIXME_s

8

!!!_s

the noteworthy

(for all sorts of reasons)

turtle

Turtle graphics.

turtle

Turtle graphics.

```
if len(self.currentLine) > 42: # 42! answer to the ultimate question  
                                # of life, the universe and everything  
    self._newLine()
```

cpython/Lib/turtle.py, line 3192

turtle

Turtle graphics.

```
if len(self.currentLine) > 42: # 42! answer to the ultimate question
                               # of life, the universe and everything
    self._newLine()
```

cpython/Lib/turtle.py, line 3192

```
class TurtleGraphicsError(Exception):
    """Some TurtleGraphics Error
    """
```

cpython/Lib/turtle.py, line 866

turtle

Turtle graphics.

```
if len(self.currentLine) > 42: # 42! answer to the ultimate question  
                                # of life, the universe and everything  
    self._newLine()
```

cpython/Lib/turtle.py, line 3192

```
class TurtleGraphicsError(Exception):  
    """Some TurtleGraphics Error  
    """
```

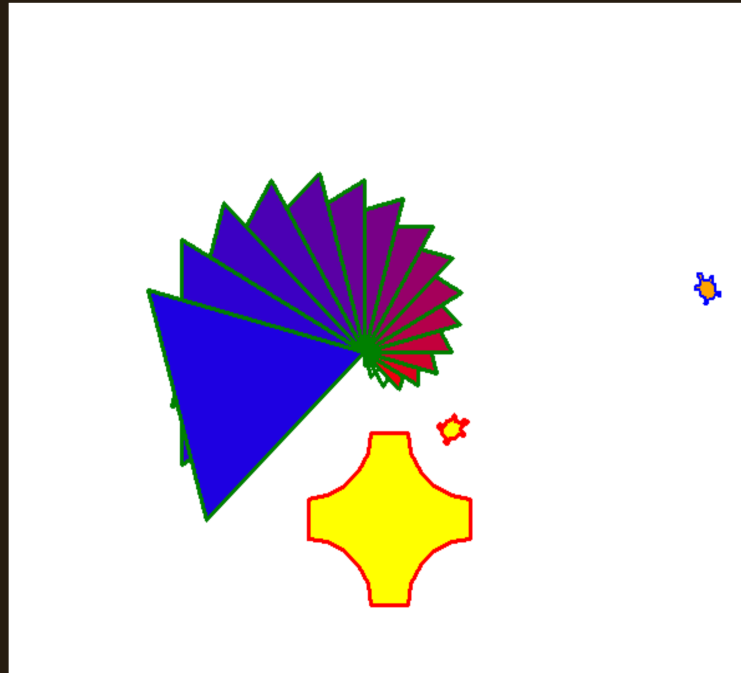
cpython/Lib/turtle.py, line 866

```
if abs(self._position - new) > 0.5:  
    print ("undogoto: HALLO-DA-STIMMT-WAS-NICHT!")
```

cpython/Lib/turtle.py, line 3204

```
$ python3 -m turtle
```

```
$ python3 -m turtle
```



string

** ~~ just string things ~~ **

```
>>> import string
```

string

** ~~ just string things ~~ **

```
>>> import string
```

```
>>> string.ascii_lowercase  
'abcdefghijklmnopqrstuvwxyz'
```

```
>>> string.printable  
'0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ  
LMNOPQRSTUVWXYZ!"#$%&\'()*+,-./:;<=>?  
@[\\]^_`{|}~ \t\n\r\x0b\x0c'
```

2to3

Automatic Python 2 to 3 code translation.

2to3

Automatic Python 2 to 3 code translation.

```
name = raw_input()
print "I am {} the Mighty, a knight of great renown.
My weapon, 'Life Ender' has struck down hundreds of
foes. Begone, or you'll be next!".format(name)
```


2to3

Automatic Python 2 to 3 code translation.

```
name = raw_input()
print "I am {} the Mighty, a knight of great renown.
My weapon, 'Life Ender' has struck down hundreds of
foes. Begone, or you'll be next!".format(name)
```

```
$ 2to3 example.py
```

2to3

Automatic Python 2 to 3 code translation.

```
name = raw_input()
print "I am {} the Mighty, a knight of great renown.
My weapon, 'Life Ender' has struck down hundreds of
foes. Begone, or you'll be next!".format(name)
```

```
$ 2to3 example.py
```

```
name = input()
print("I am {} the Mighty, a knight of great renown.
My weapon, 'Life Ender' has struck down hundreds of
foes. Begone, or you'll be next!".format(name))
```

textwrap

Paragraph & indentation helpers.

textwrap

Paragraph & indentation helpers.

```
>>> intro = """\
...     A Slug captain hails and invites himself aboard
...     your ship to present a flask of something slimy.
...     "Now, most gracioussss captain, you must join me
...     please in a drink to our alliance!"
...     """
```

textwrap

Paragraph & indentation helpers.

```
>>> intro = """\n...     A Slug captain hails and invites himself aboard\n...     your ship to present a flask of something slimy.\n...     "Now, most gracioussss captain, you must join me\n...     please in a drink to our alliance!"\n... """
```

```
>>> intro\n'A Slug captain hails and invites himself aboard\nyour ship to present a flask of something slimy.\n"Now, most gracioussss captain, you must join me\nplease in a drink to our alliance!"
```

```
>>> intro
```

```
██████████ A Slug captain hails and invites himself aboard\n██████████ your ship to present a flask of something slimy.\n██████████ "Now, most gracioussss captain, you must join me\n██████████ please in a drink to our alliance!"
```

```
>>> import textwrap
>>> # Remove leading whitespace
>>> intro = textwrap.dedent(intro)
>>> intro
```

```
A Slug captain hails and invites himself aboard\n
your ship to present a flask of something slimy.\n
"Now, most gracioussss captain, you must join me\n
please in a drink to our alliance!"\n
```

```
>>> # Wrap to a specific number of characters
>>> textwrap.fill(intro, width=40)
```

```
A Slug captain hails and invites himself\naboard your ship to present a flask of\nsomething slimy. Now, most graciousss\n captain, you must join me please in a\ndrink to our alliance!
```



```
>>> # Wrap to a specific number of characters  
>>> textwrap.fill(intro, width=40)
```

```
A Slug captain hails and invites himself\naboard your ship to present a flask of\nsomething slimy. Now, most graciousss\n captain, you must join me please in a\ndrink to our alliance!\n
```

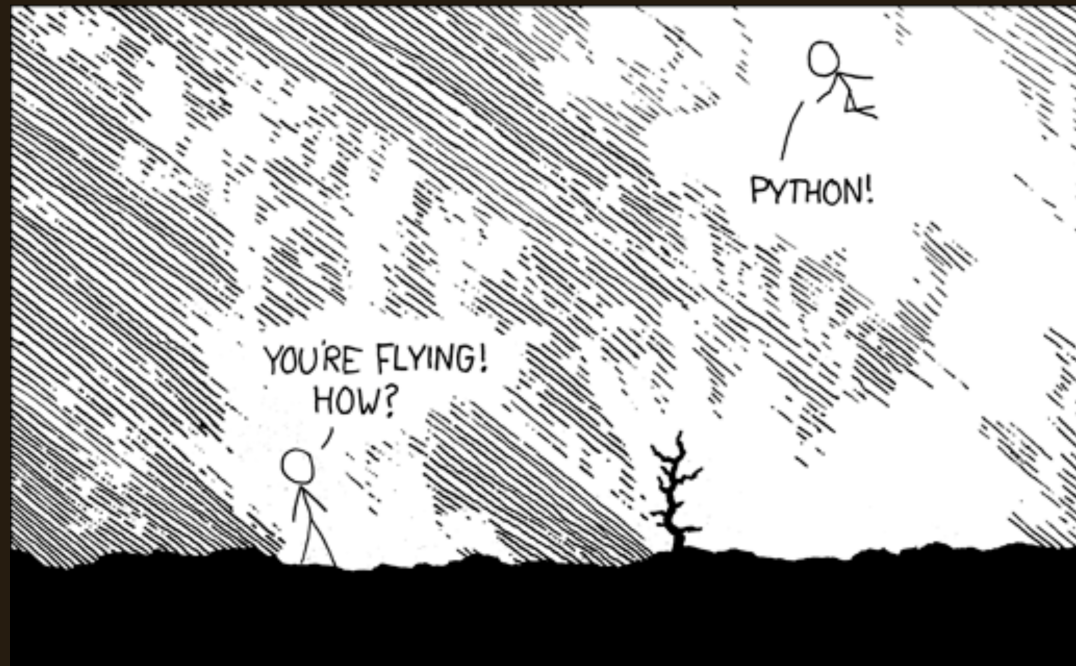


https://ftl.fandom.com/wiki/Races/Crew_Members

webbrowser

Your partner for all your browser-opening needs.

webbrowser



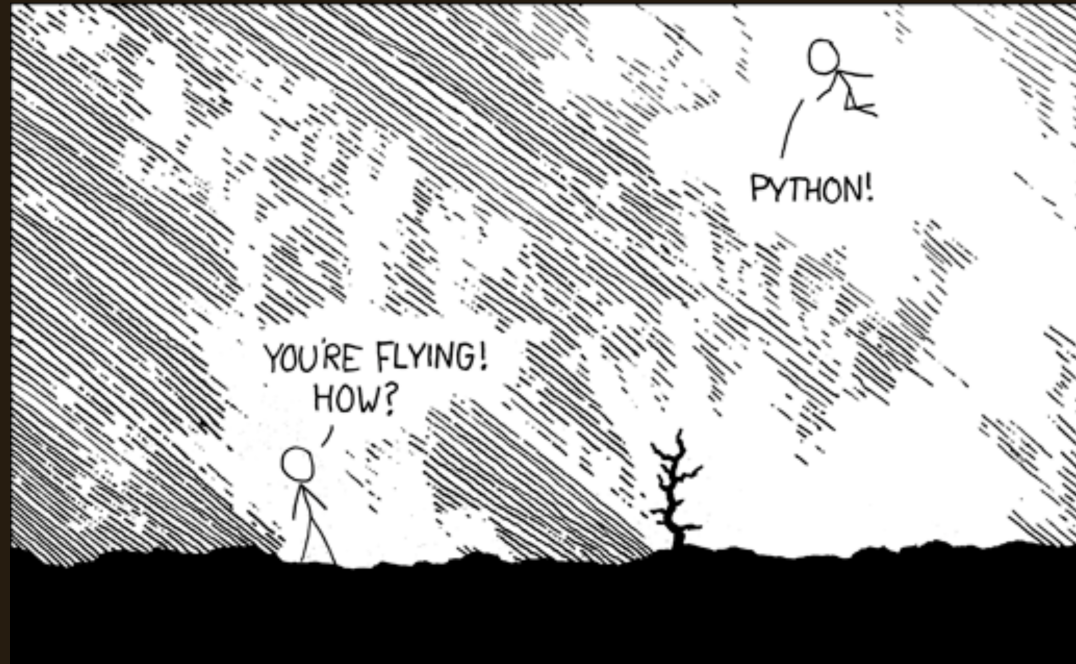
CC BY-NC 2.5

xkcd.com/353

webbrowser

```
webbrowser.open("https://xkcd.com/353/")
```

cpython/Lib/antigravity.py, line 5



CC BY-NC 2.5

xkcd.com/353

```
>>> import antigravity
```

```
>>> import antigravity
```

```
>>> from antigravity import ???
```

urllib

Working with URLs.

```
>>> from urllib.request import urlopen
```

urllib

Working with URLs.

```
>>> from urllib.request import urlopen  
  
>>> request = urlopen(  
...     'https://pokeapi.co/api/v2/pokemon/umbreon/'  
... )
```

Make sure to follow PokeApi's usage policy: <https://pokeapi.co/docs/v2.html/>

urllib

Working with URLs.

```
>>> from urllib.request import urlopen

>>> headers = {
...     'User-Agent': 'Python/urllib @ EuroPython'
... }
>>> request = urlopen(
...     'https://pokeapi.co/api/v2/pokemon/umbreon/'
... )
>>> # ...I can has sends headers?
```

urllib

Working with URLs.

```
>>> from urllib.request import Request, urlopen

>>> headers = {
...     'User-Agent': 'Python/urllib @ EuroPython'
... }
>>> request = Request(
...     'https://pokeapi.co/api/v2/pokemon/umbreon/',
...     headers=headers
... )
>>> umbreon = urlopen(request).read()
```

urllib

Working with URLs.

```
>>> from urllib.request import Request, urlopen

>>> headers = {
...     'User-Agent': 'Python/urllib @ EuroPython'
... }
>>> request = Request(
...     'https://pokeapi.co/api/v2/pokemon/umbreon/',
...     headers=headers
... )
>>> umbreon = urlopen(request).read()
>>> # ...I can has JSON?
```

urllib

Working with URLs.

```
>>> import json
>>> from urllib.request import Request, urlopen

>>> headers = {
...     'User-Agent': 'Python/urllib @ EuroPython'
... }
>>> request = Request(
...     'https://pokeapi.co/api/v2/pokemon/umbreon/',
...     headers=headers
... )
>>> umbreon = json.load(urlopen(request))
```

urllib

Working with URLs.

```
>>> import json
>>> from urllib.request import Request, urlopen

>>> headers = {
...     'User-Agent': 'Python/urllib @ EuroPython'
... }
>>> request = Request(
...     'https://pokeapi.co/api/v2/pokemon/umbreon/',
...     headers=headers
... )
>>> umbreon = json.load(urlopen(request))
```





<https://sonicpokemon.fandom.com/wiki/File:Glaceon.png>

urllib

Working with URLs.

```
>>> import json
>>> from urllib.request import Request, urlopen

>>> headers = {
...     'User-Agent': 'Python/urllib @ PyDays'
... }
>>> request = Request(
...     'https://pokeapi.co/api/v2/pokemon/umbreon/',
...     headers=headers
... )
>>> umbreon = json.load(urlopen(request))
```



Maybe, just this once...

```
$ pip install requests
```



```
>>> import requests    # pip install requests
>>> umbreon = requests.get(
...     'https://pokeapi.co/api/v2/pokemon/umbreon/',
...     headers={'User-Agent': 'Python/urllib @ PyDays'}
... ).json()
```

pprint

String representation for humans.

```
>>> import requests    # pip install requests
>>> umbreon = requests.get(
...     'https://pokeapi.co/api/v2/pokemon/umbreon/',
...     headers={'User-Agent': 'Python/urllib @ PyDays'}
... ).json()
```

pprint

String representation for humans.

```
>>> import requests    # pip install requests
>>> umbreon = requests.get(
...     'https://pokeapi.co/api/v2/pokemon/umbreon/',
...     headers={'User-Agent': 'Python/urllib @ PyDays'}
... ).json()
```

```
{'abilities': [{'ability': {'name': 'inner-focus', 'url': 'https://pokeapi.co/api/v2/ability/39/'}, 'is_hidden': True, 'slot': 3}, {'ability': {'name': 'synchronize', 'url': 'https://pokeapi.co/api/v2/ability/28/'}, 'is_hidden': False, 'slot': 1}], 'base_experience': 184, 'forms': [{'name': 'umbreon', 'url': 'https://pokeapi.co/api/v2/pokemon-form/197/'}], 'game_indices': [{'game_index': 197, 'version': {'name': 'white-2', 'url': 'https://pokeapi.co/api/v2/version/22/'}}], {'game_index': 197, 'version': {'name': 'black-2', 'url': 'https://pokeapi.co/api/v2/version/21/'}}], {'game_index': 197, 'version': {'name': 'white', 'url': 'https://pokeapi.co/api/v2/version/18/'}}], {'game_index': 197, 'version': {'name': 'black', 'url': 'https://pokeapi.co/api/v2/version/17/'}}], {'game_index': 197, 'version': {'name': 'soulsilver', 'url': 'https://pokeapi.co/api/v2/version/16/'}}],
```

```
# ...and much more (261+ kB)
```

```
>>> from pprint import pprint  
>>> pprint(umbreon)
```

```
>>> from pprint import pprint
>>> pprint(umbreon)
```

```
{'abilities': [{'ability': {'name': 'inner-focus',
                             'url': 'https://pokeapi.co/api/v2/ability/39/'},
                  'is_hidden': True,
                  'slot': 3},
                 {'ability': {'name': 'synchronize',
                             'url': 'https://pokeapi.co/api/v2/ability/28/'},
                  'is_hidden': False,
                  'slot': 1}],
 'base_experience': 184,
 'forms': [{'name': 'umbreon',
             'url': 'https://pokeapi.co/api/v2/pokemon-form/197/'}],
 'game_indices': [{'game_index': 197,
                    'version': {'name': 'white-2',
                                'url': 'https://pokeapi.co/api/v2/version/22/'}}],

# ... and much more
```

itertools

Advanced iterators.

itertools

Advanced iterators.

```
>>> import itertools
```

```
>>> seq = [[10], [25, 31, 5], [], [89]]
```

itertools

Advanced iterators.

```
>>> import itertools

>>> seq = [[10], [25, 31, 5], [], [89]]

>>> list(itertools.chain.from_iterable(seq))
[10, 25, 31, 5, 89]
```



```
>>> import itertools

>>> seq = [10, 25, 31, 5, 89]

>>> list(itertools.dropwhile(lambda x: x < 30,
...                           seq))
[31, 5, 89]
```

functools

Higher order functions.

functools

Higher order functions.

```
>>> import functools
>>> import math

>>> functools.reduce(
...     math.gcd,
...     [105, 21, 99, 10, 256]
... )
1
```

```
>>> def address(latitude, longitude):  
...     address = expensive_geocoder(latitude,  
...                                   longitude)  
...     return address
```

```
>>> import functools
>>> # Memoize all the things!
>>> @functools.lru_cache(maxsize=1024)
>>> def address(latitude, longitude):
...     address = expensive_geocoder(latitude,
...                                   longitude)
...     return address
```

collections

Additional useful data structures.

collections

Additional useful data structures.

```
>>> import collections
>>> inventory = ['aloe', 'salve', 'ginseng',
...              'ginseng', 'aloe', 'aloe']
```

collections

Additional useful data structures.

```
>>> import collections
>>> inventory = ['aloe', 'salve', 'ginseng',
...              'ginseng', 'aloe', 'aloe']

>>> counter = collections.Counter(inventory)
>>> counter
Counter({'aloe': 3, 'ginseng': 2, 'salve': 1})
```


collections

Additional useful data structures.

```
>>> import collections
>>> inventory = ['aloe', 'salve', 'ginseng',
...              'ginseng', 'aloe', 'aloe']

>>> counter = collections.Counter(inventory)
>>> counter
Counter({'aloe': 3, 'ginseng': 2, 'salve': 1})

>>> counter['ginseng']
2
```

collections

Additional useful data structures.

```
>>> import collections
>>> inventory = ['aloe', 'salve', 'ginseng',
...              'ginseng', 'aloe', 'aloe']

>>> counter = collections.Counter(inventory)
>>> counter
Counter({'aloe': 3, 'ginseng': 2, 'salve': 1})

>>> counter['ginseng']
2
>>> counter['bezel cup']
0
```

```
>>> occupations = {'Zanthia': 'mystic',  
...                 'Malcolm': 'jester'}  
>>> occupations['Brandon']
```

```
>>> occupations = {'Zanthia': 'mystic',  
...                'Malcolm': 'jester'}  
>>> occupations['Brandon']  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
KeyError: 'Brandon'
```

```
>>> occupations = {'Zanthia': 'mystic',  
...                 'Malcolm': 'jester'}  
>>> occupations['Brandon']  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
KeyError: 'Brandon'
```

```
>>> occupations = \  
...     collections.defaultdict(lambda: 'n/a')  
>>> occupations['Zanthia'] = 'mystic'  
>>> occupations['Malcolm'] = 'jester'
```

```
>>> occupations = {'Zanthia': 'mystic',  
...                 'Malcolm': 'jester'}  
>>> occupations['Brandon']  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
KeyError: 'Brandon'
```

```
>>> occupations = \  
...     collections.defaultdict(lambda: 'n/a')  
>>> occupations['Zanthia'] = 'mystic'  
>>> occupations['Malcolm'] = 'jester'  
  
>>> occupations['Brandon']  
'n/a'
```

His Royal Highness
BRANDON, King of
Kyrandia, Defender of
the Kyragem, Vanquisher
of Malcolm, Patron of
the Arts, Wearer of the
Sandal, and Majority
Stock Owner and CEO of
Timbermist Casual
Footwear, Inc.

https://lparchive.org/Legend-of-Kyrandia-2-Hand-of-Fate/Update%2016/72-kyrandia2_update15_037.png

```
>>> occupations = \
...     collections.defaultdict(lambda: 'n/a')
>>> occupations['Zanthia'] = 'mystic'
>>> occupations['Malcolm'] = 'jester'

>>> occupations['Brandon']
'n/a'
```

traceback

Stack traces you *like* to see.

traceback

Stack traces you *like* to see.

```
>>> def frobnicate():  
...     foobarify()  
  
>>> def foobarify():  
...     bazificate()  
  
>>> def bazificate():  
...     import traceback  
...     traceback.print_stack()
```

traceback

Stack traces you *like* to see.

```
>>> def frobnicate():  
...     foobarify()  
  
>>> def foobarify():  
...     bazificate()  
  
>>> def bazificate():  
...     import traceback  
...     traceback.print_stack()  
  
>>> frobnicate()  
File "<stdin>", line 1, in <module>  
File "<stdin>", line 2, in frobnicate  
File "<stdin>", line 3, in foobarify  
File "<stdin>", line 4, in bazificate
```



<https://images.chickadvisor.com/item/47673/original/285af65a5ddf144a4c59a224d69e9f6c.jpg>



<https://images.chickadvisor.com/item/47673/original/285af65a5ddf144a4c59a224d69e9f6c.jpg>

...and the list goes on.

argparse

...and the list goes on.

argparse

...and the list goes on.

smtpd

argparse

colorsys

...and the list goes on.

smtpd

argparse

colorsys

...and the list goes on.

ast

smtpd

argparse

email

inspect

colorsys

...and the list goes on.

ast

smtpd

html

asyncio

email

argparse

colorsys

inspect

logging

...and the list goes on.

curses

ast

smtpd

time

html

asyncio

email

argparse

array

os

colorsys

inspect

logging

...and the list goes on.

curses

sys

tempfile

ast

smtpd

time

html

platform

import bacon

 @aerinthenniel

difflib weakref asyncio email
argparse multiprocessing array
os colorsys csv
inspect re logging
threading ...and the list goes on.
curses sys copy tempfile
struct ast smtpd
html time signal
numbers platform

difflib weakref pickle enum
asyncio email
argparse wave
multiprocessing array
os ipaddress colorsys csv
inspect re logging
threading ..and the list goes on.
curses sys binhex
struct ast marshal copy tempfile
html dbm marshal smtpd base64
time subprocess signal
numbers platform
gettext
import bacon

The Python standard library
can do almost **anything**.

What's new in Python 3.0?

```
>>>
```



```
>>> import bacon
```



...awesomeness.

abstrusegoose.com/81

CC BY-NC 3.0 US

```
import bacon    # ???
```


For the rest, there is the
Python Package Index.

```
pip install baconify
```

Thank you!

 @aerinthenniel