import bacon
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.
Import bacon
The Python standard library can do almost anything.
What's new in Python 3.0?

```python
>>> import bacon
...awesomeness.
```
import bacon  

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

import bacon
639

XXXs
XXXs

# XXX obviously wrong, see #3232
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: shouldn't this code be removed, not commented out?
cpython/Lib/importlib/_bootstrap_external, line 411

# 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 XXXs 80 hacks
# Can't think of a clean solution, so we hack away

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

import bacon
# 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

import bacon

@aerinthenniel
639 XXXs
80 hacks
74 TODOs

import bacon

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

cpython/Lib/enum.py, line 426
61

FIXMEs

import bacon
639 XXXs
80 hacks
74 TODOs

61 FIXMEs

import bacon
@aerintheniell
import bacon
import bacon
import bacon

639 XXXs

80 hacks

74 TODOs

61 FIXMEs

8 !!!s
the notable

(noteworthy)

(for all sorts of reasons)
turtle

Turtle graphics.
turtle

Turtle graphics.

```python
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

import bacon
import bacon

turtle

Turtle graphics.

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

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

cpython/Lib/turtle.py, line 3192

cpython/Lib/turtle.py, line 866
Turtle graphics.

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

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

if abs(self._position - new) > 0.5:
    print("undogoto: HALLO-DA-STIMMT-WAS-NICHT!")
```
$ python3 -m turtle
$ python3 -m turtle

import bacon
string

* ~~ just string things ~~ *

>>> import string
```
>>> import string

>>> string.ascii_lowercase
'abcdefghijklmnopqrstuvwxyz'

>>> string.printable
'0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ'"#$%\'()>*+,-./:;<=>?@[\]^_`{|}~ 	
\r\n\x0b\x0c'
```
import bacon

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
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.
Paragraph & indentation helpers.

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

Paragraph & indentation helpers.

```python
>>> 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!"
... ""

>>> 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!'
```

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

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

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

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

Your partner for all your browser-opening needs.
webbrowser

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

import bacon
>>> import antigravity
```python
>>> import antigravity

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

Working with URLs.

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

Working with URLs.

```python
>>> 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/
```
Working with URLs.

```python
>>> 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.

```python
>>> 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()
```
Working with URLs.

```python
>>> 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.

```python
>>> 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))
```
Working with URLs.

```python
>>> 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.

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

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

```python
>>> 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
import bacon

iterertools

Advanced iterators.
itertools

Advanced iterators.

```python
>>> import itertools

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

Advanced iterators.

```python
>>> 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.

```python
>>> 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
```python
>>> 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.

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

Additional useful data structures.

```python
>>> 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.

```python
>>> 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.

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

>>> occupations['Brandon']
'n/a'
Stack traces you *like* to see.
traceback

Stack traces you *like* to see.

```python
>>> def frobnicate():
...    foobarify()

>>> def foobarify():
...    bazificate()

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

*import bacon*
traceback

Stack traces you *like* to see.

```python
>>> 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
```
import bacon

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

import bacon
argparse

...and the list goes on.

colorsys

smtpd

import bacon
...and the list goes on.

import bacon
import bacon

argparse
colorsys
inspect
email
ast
smtpd
html

...and the list goes on.
...and the list goes on.
...and the list goes on.
...and the list goes on.
...and the list goes on.
The Python standard library can do almost anything.
What's new in Python 3.0?

```ruby
>>> import bacon  # ???
```

[Image: abstrusegoose.com/81]
For the rest, there is the Python Package Index.

```
import bacon
pip install baconify
```
Thank you!

@aerinthenniel