

Idioms & What is *Pythonic*?

Scott Burns
@scottsburns

Idioms

How old are you?

English

I am 30 years old.

I have 30 years.

French

Je suis 30 ans.

J'ai 30 ans.

Idioms

Constructs of the
language that tend
not to translate
literally.

What does it mean to
be *pythonic*?

Code uses the language
idioms



Important Idioms

Iteration

Comprehensions

Exceptions

(not an exhaustive list)

Iteration, pt. 0

Do something for each item in a collection

```
data = ['a', 'b', 'c']  
for datum in data:  
    do_some_work(datum)
```

Iteration, pt. 1

```
from me import eat
cookies = [
    'peanut butter',
    'chocolate',
    'raisin',
    'oatmeal',
]
```

```
for i in range(cookies):
    eat(cookies[i])
```

PLEASE DON'T

```
for cookie in cookies:
    eat(cookie)
```

PYTHONIC

Iteration, pt. 2

```
from me import eat_the_whole_box
from cookie import CookieBox
cookies = {
    'oatmeal': CookieBox('oatmeal', N=30),
    'chocolate': CookieBox('chocolate', N=10),
}
```

```
for cookie_type in cookies:
    if cookie_type == 'chocolate':
        box = cookies[cookie_type]
        eat_the_whole_box(box)
```

PLEASE DON'T

```
for cookie_type, box in cookies.iteritems():
    if cookie_type == 'chocolate':
        eat_the_whole_box(box)
```

PYTHONIC

Iteration, pt. 3

```
from me import eat
cookies = [
    'peanut butter',
    'chocolate',
    'raisin',
    'oatmeal',
]
```

```
for cookie in cookies:
    ind = cookies.index(cookie)
    if ind % 2 == 0:
        eat(cookie)
```

PLEASE DON'T

```
for ind, cookie in enumerate(cookies):
    if ind % 2 == 0:
        eat(cookie)
```

PYTHONIC

Comprehensions, pt. 0

Very readable for-loops

```
[expr for item in iterable (if cond)]
```

```
{key: value for item in iterable (if cond)}
```

```
{item for item in iterable (if cond)}
```

Comprehensions, pt. 1

```
from me import eat
cookies = [
    'peanut butter',
    'chocolate',
    'raisin',
    'oatmeal',
]
```

```
eaten_cookies = []
for cookie in cookies:
    eaten_cookies.append(eat(cookie))
```

```
eaten_cookies = [eat(cookie) for cookie in cookies]
```

PLEASE DON'T

PYTHONIC

Comprehensions, pt. 2

```
from me import eat
cookies = [
    'peanut butter',
    'chocolate',
    'raisin',
    'oatmeal',
]
```

```
eaten_cookies = []
for cookie in cookies:
    if cookie != 'raisin':
        eaten_cookies.append(eat(cookie))
```

PLEASE DON'T

```
eaten_cookies = [eat(cookie) for cookie in cookies
                  if cookie != 'raisin']
```

PYTHONIC

Comprehensions, pt. 3

```
from me import eat
from cookie import CookieBox
cookies = [
    'peanut butter',
    'chocolate',
    'raisin',
    'oatmeal',
]
```

```
cookie_boxes = {}
for cookie in cookies:
    cookie_boxes[cookie] = CookieBox(cookie)
```

```
eaten_cookies = {cookie: CookieBox(cookie)
                  for cookie in cookies}
```

PLEASE DON'T

PYTHONIC

Comprehensions, pt. 4

```
from me import eat
cookies = [
    'peanut butter',
    'chocolate',
    'raisin',
    'oatmeal',
    'raisin',
]
```

```
eaten_cookies = []
for cookie in cookies:
    eaten_cookies.append(eat(cookie))
unique_cookies = set(eaten_cookies)
```

```
unique_cookies = {eat(cookie)
                  for cookie in cookies}
```

PLEASE DON'T

PYTHONIC

Exceptions, pt. 0

Flow-control when bad things
happen :(

```
try:  
    my_method(data)  
except ExceptionName:  
    do_something_else()  
except (OtherException, YetAnotherException):  
    do_something_for_either_exception()
```


Exceptions, pt. 1

```
from me import eat, StomachAche
cookies = [
    'peanut butter',
    'chocolate',
]
```

```
for cookie in cookies:
    eaten, ache = eat(cookie)
    if ache:
        break
```

PLEASE DON'T

```
for cookie in cookies:
    try:
        eaten = eat(cookie)
    except StomachAche:
        break
```

PYTHONIC

Exceptions, pt. 2

```
from me import eat, StomachAche, HeadAche
cookies = [
    'peanut butter',
    'chocolate',
]
```

```
for cookie in cookies:
    eaten, ache = eat(cookie)
    if ache == 'stomach':
        break
    if ache == 'head':
        take_advil()
```

PLEASE DON'T

```
for cookie in cookies:
    try:
        eaten = eat(cookie)
    except StomachAche:
        break
    except HeadAche:
        take_advil()
```

PYTHONIC

Exceptions, pt. 2

```
from me import eat, StomachAche, HeadAche
cookies = [
    'peanut butter',
    'chocolate',
]
```

```
for cookie in cookies:
    eaten, ache = eat(cookie)
    if ache == 'stomach':
        break
    if ache == 'head':
        take_advil()
```

PLEASE DON'T

```
for cookie in cookies:
    try:
        eaten = eat(cookie)
    except StomachAche:
        break
    except HeadAche:
        take_advil()
```

PYTHONIC

But seriously.....

Raymond Hettinger's Talk:
“Beyond PEP8: Best practices
for beautiful intelligible code