Python idiomático

¿Para qué?

Aprovecha la expresividad de Python

Más legible

Menos errores

Recuerda: Mejor legible que eficiente

Intercambiar variables



a=1

b=2

c=a

a=b

b=c

Intercambiar variables



a=1

b=2

c=a

a=b

b=c



a,b = b,a



```
number_list = [4, 1, 9]
a, b, c = number_list
```



```
list = [1, 2, 3, 4]
first, last = list[0], list[-1]
```



```
list = [1, 2, 3, 4]
first, last = list[0], list[-1]
6
first, *__, last = list
```



```
test_list = [1, 2, 3, 4]
first, last = test_list[0], test_list[-1]

first, *__, last = test_list

header, *rows = open("filename").readlines()
```

¿Alguno es X?



```
if city == 'Troya' or city == 'Esparta' or city == '
found = True
```

¿Alguno es X?



```
if city == 'Troya' or city == 'Esparta' or city == '
found = True
```



```
found = city in {'Troya', 'Esparta', 'Atenas'}
```

¿Lista vacía?



```
a = [1, 2, 3]
if len(a) > 0:
```

¿Lista vacía?



```
a = [1, 2, 3]
if len(a) > 0:
...
```



```
a = [1, 2, 3] if a:
```

¿Entre dos valores?



```
if x < 10 and x > 5:
```

¿Entre dos valores?

```
if x < 10 and x > 5:
...

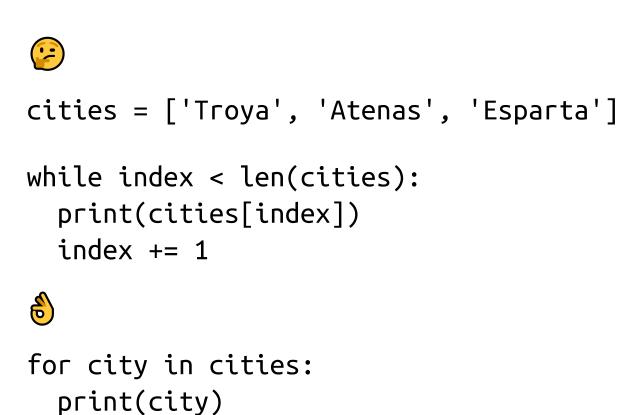
if 5 < x < 10:
```

Bucles



```
cities = ['Troya', 'Atenas', 'Esparta']
while index < len(cities):
   print(cities[index])
   index += 1</pre>
```

Bucles



Mas Bucles



```
while index < len(ls):
    print(index, ls[index])
    index += 1</pre>
```

Mas Bucles



```
while index < len(ls):</pre>
  print(index, ls[index])
  index += 1
6
for index, value in enumerate(ls):
  print(index, value)
```

Acceso a diccionarios



```
d = {'Aquiles': 'aquiles@notmyemail.com'}
if d.has_key('Aquiles'):
    a = d['Aquiles']
else:
    a = 'sin_email'
```

Acceso a diccionarios

```
d = {'Aquiles': 'aquiles@notmyemail.com'}

if d.has_key('Aquiles'):
    a = d['Aquiles']

else:
    a = 'sin_email'

a = d.get('Aquiles', 'sin_email')
```

Acceso a diccionarios

```
d = {'Aquiles': 'aquiles@notmyemail.com'}
if d.has_key('Aquiles'):
    a = d['Aquiles']
else:
    a = 'sin_email'
6
if 'Aquiles' in d:
    print( d['Aquiles'] )
```

Fichero de texto

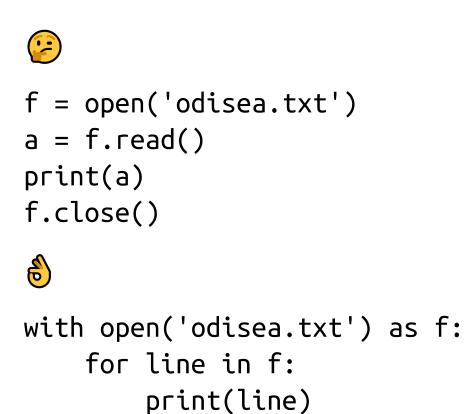


```
f = open('odisea.txt')
a = f.read()
print(a)
f.close()
```

Fichero de texto

```
f = open('odisea.txt')
a = f.read()
print(a)
f.close()
try:
    f = open('odisea.txt')
    a = f.read()
    print(a)
finally:
    f.close()
```

Fichero de texto



Pero recuerda: Explícito mejor que implícito



```
def make_complex(*args):
    x, y = args
    return dict(**locals())
```

Pero recuerda: Explícito mejor que implícito



```
def make_complex(*args):
    x, y = args
    return dict(**locals())

def make_complex(x, y):
    return {'x': x, 'y': y}
```