In [127…
```
def f():
    return 3
    return 4
```

In [128…
```
f()
```

Out[128… 3

In [129…
```
def g():
    yield 3
    yield 4
```

In [130…
```
gen = g()
```

In [131…
```
next(gen)
```

Out[131… 3

In [132…
```
next(gen)
```

Out[132… 4

In [133…
```
next(gen)
```

```
---------------------------------------------------------------------------
StopIteration                             Traceback (most recent call last)
<ipython-input-133-6e72e47198db> in <module>
----> 1 next(gen)

StopIteration:
```

In [134…
```
def prime_gen():
    yield 1
    n=2
    while True:
        for d in range(2, n-1):
            if n%d==0:
                break
        else:
            yield n
        n+=1
```

In [135…
```
gen = prime_gen()
```

In [136…
```
next(gen)
```

Out[136… 1

In [12]:
```
next(gen)
```

Out[12]: 1

In [13]:
```
next(gen)
```

Out[13]: 2

In [14]:
```
next(gen)
```

Out[14]: 3

In [15]:
```
next(gen)
```

Out[15]: 5

In [16]:
```
next(gen)
```

Out[16]: 7

In [77]:
```
next(gen)
```

Out[77]: 313

In [78]:
```
gen = prime_gen()
sum(next(gen) for i in range(100))
```

Out[78]: 23071

In [108…
```
a = ['a', 'b', 'c']
g = enumerate(a)
```

In [109…
```
next(g)
```

Out[109… (0, 'a')

In [110…
```
next(g)
```

Out[110… (1, 'b')

In [111…
```
next(g)
```

Out[111… (2, 'c')

In [114…
```
a=[x for x in range(100_000_000)]
```

In [115…
```
a[10000]
```

Out[115… 10000

In [116…
```
del a
```

In [117…
```
a=(x for x in range(100_000_000))
```

In [118…
```
next(a)
```

Out[118… 0

In [119…
```
next(a)
```

Out[119… 1

In [120…
```
next(a)
```

Out[120… 2

In [121…
```
next(a)
```

Out[121… 3

In [122…
```
c = 0
for x in a:
    print (x)
    c += 1
    if c>5:
        break
```

```
4
5
6
7
8
9
```

In [123…
```
next(a)
```

Out[123… 10

In [124…
```
c = 0
for x in a:
    print (x)
    c += 1
    if c>5:
        break
```

```
11
12
13
14
15
16
```

In [125…
```
a=(x for x in range(100_000_000))
```

In [126…
```
next(a)
```

Out[126… 0

In [138…
```
print (a)
```

```
mitsos
```

In [137…
```
a='mitsos'
```

In [139…
```
import antigravity
```

In [140…
```
import this
```

```
The Zen of Python, by Tim Peters
```

```
Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
```

In [141…
```
import random
```

In [171…
```
random.randint(10,20)
```

Out[171… 17

In [196…
```
random.choice(["heraklion", "athens", "thessaloniki"])
```

Out[196… 'thessaloniki'

In [242…
```
random.sample([1,2,3,4,5,6,7,8], 2)
```

Out[242… [5, 8]

In [271…
```
random.random()
```

Out[271… 0.7247444535768666

In [1]:
```
import program
```

In [274…
```
program.my_fabulous_function(10)
```

Out[274… 20

In [3]:
```
program.name
```

Out[3]: 'mitsos'

In [1]:
```
from program import my_fabulous_function
```

In [2]:
```
my_fabulous_function(10)
```

Out[2]: 20

In [3]:
```
from random import randint
```

In [4]:
```
randint(10, 50)
```

Out[4]:
```
49
```

In [1]:
```
from program import *
```

In [2]:
```
name
```

Out[2]:
```
'mitsos'
```

In [3]:
```
my_fabulous_function(10)
```

Out[3]:
```
20
```

In [1]:
```
import program
```

```
hello world!
```

In [1]:
```
from program import print_numbers
```

```
hello world!
```

In [2]:
```
print_numbers(5)
```

```
1
2
3
4
5
```

Exceptions

In [3]:
```
qwertyu
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-3-bfe75041371e> in <module>
----> 1 qwertyu

NameError: name 'qwertyu' is not defined
```

In [4]:
```
1/0
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-4-9e1622b385b6> in <module>
----> 1 1/0

ZeroDivisionError: division by zero
```

In [5]:
```
a={
    "name": "mitsos"
}
a['age']
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-5-264634e8a94b> in <module>
      2     "name": "mitsos"
      3 }
----> 4 a['age']

KeyError: 'age'
```

In [6]:
```
int('mitsos')
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-6-24e8b5b4a1dd> in <module>
----> 1 int('mitsos')

ValueError: invalid literal for int() with base 10: 'mitsos'
```

In [7]:
```
open('dfasdf')
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-7-5d440f3b39dc> in <module>
----> 1 open('dfasdf')

FileNotFoundError: [Errno 2] No such file or directory: 'dfasdf'
```

In [8]:
```
try:
    a=1/0
except Exception:
    print ('oops')
```

```
oops
```

In [9]:
```
a=1/0
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-9-023a503edd86> in <module>
----> 1 a=1/0

ZeroDivisionError: division by zero
```

In [15]:
```
try:
    open('asdfasdf')
    a=1/0
except Exception:
    print ('oops')
```

```
oops
```

In [11]:
```
print (a)
```

```
0.5
```

In [17]:
```
try:
    open('dsfsdf')
    a=1/0
except ZeroDivisionError:
    print ('Daireses me to 0')
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-17-4dba8afe5ddf> in <module>
      1 try:
----> 2     open('dsfsdf')
      3     a=1/0
      4 except ZeroDivisionError:
      5     print ('Daireses me to 0')

FileNotFoundError: [Errno 2] No such file or directory: 'dsfsdf'
```

In [21]:
```
def g():
    #a=1/0
    open('sdfsd')

def f():
    g()

try:
    f()
except ZeroDivisionError:
    print ('Daireses me to 0')
except FileNotFoundError:
    print ('DEN YPARXEI TO ARXEIO!!!!!')
```

```
DEN YPARXEI TO ARXEIO!!!!!
```

In [23]:
```
try:
    #a=1/0
    #open('sdfsd')
    int('mitsos')
except ZeroDivisionError:
    print ('Daireses me to 0')
except FileNotFoundError:
    print ('DEN YPARXEI TO ARXEIO!!!!!')
except Exception:
    print ('Kati allo kako sunebei!')
```

```
Kati allo kako sunebei!
```

In [24]:
```
try:
    #a=1/0
    #open('sdfsd')
    int('mitsos')
except ZeroDivisionError:
    print ('Daireses me to 0')
except FileNotFoundError:
    print ('DEN YPARXEI TO ARXEIO!!!!!')
except:
    print ('Kati allo kako sunebei!')
```

```
Kati allo kako sunebei!
```

In [27]:
```
try:
    #a=1/0
    #open('sdfsd')
    int('mitsos')
except ZeroDivisionError:
    print ('Daireses me to 0')
except FileNotFoundError:
    print ('DEN YPARXEI TO ARXEIO!!!!!')
except Exception as e:
    print ('Kati allo kako sunebei: ', str(e))

print ('aaa')
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-27-4423f089ce6e> in <module>
      2     #a=1/0
      3     #open('sdfsd')
----> 4     int('mitsos')
```

```
      5 except ZeroDivisionError:
      6     print ('Daireses me to 0')

ValueError: invalid literal for int() with base 10: 'mitsos'
```

In [28]:
```
try:
    #a=1/0
    #open('sdfsd')
    int('mitsos')
except ZeroDivisionError:
    print ('Daireses me to 0')
except FileNotFoundError:
    print ('DEN YPARXEI TO ARXEIO!!!!!')
#except Exception as e:
#    print ('Kati allo kako sunebei: ', str(e))

print ('aaa')
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-28-4423f089ce6e> in <module>
      2     #a=1/0
      3     #open('sdfsd')
----> 4     int('mitsos')
      5 except ZeroDivisionError:
      6     print ('Daireses me to 0')

ValueError: invalid literal for int() with base 10: 'mitsos'
```

In [29]:
```
try:
    #a=1/0
    #open('sdfsd')
    int('mitsos')
except ZeroDivisionError:
    print ('Daireses me to 0')
except FileNotFoundError:
    print ('DEN YPARXEI TO ARXEIO!!!!!')
except Exception as e:
    pass # DON'T DO THIS. SHUT UP Exception

print ('aaa')
```

```
aaa
```

In [30]:
```
try:
    #a=1/0
    #open('sdfsd')
    int('mitsos')
except ZeroDivisionError:
    print ('Daireses me to 0')
except FileNotFoundError:
    print ('DEN YPARXEI TO ARXEIO!!!!!')
except Exception as e:
    print ('Kati allo kako sunebei: ', str(e))
finally:
    print ('FINALLY')

print ('aaa')
```

```
Kati allo kako sunebei:  invalid literal for int() with base 10: 'mitsos'
FINALLY
aaa
```

In [31]:
```
try:
    #a=1/0
    a=3
    #open('sdfsd')
    #int('mitsos')
except ZeroDivisionError:
    print ('Daireses me to 0')
except FileNotFoundError:
    print ('DEN YPARXEI TO ARXEIO!!!!!')
except Exception as e:
    print ('Kati allo kako sunebei: ', str(e))
finally:
    # Clean up code
    print ('FINALLY')

print ('aaa')
```

```
FINALLY
aaa
```

In [33]:
```
try:
    a=1/0
    #a=3
    #open('sdfsd')
    #int('mitsos')

finally:
    # Clean up code
    x
```

```
FINALLY
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-33-5c5c8601f536> in <module>
      1 try:
----> 2     a=1/0
      3     #a=3
      4     #open('sdfsd')
      5     #int('mitsos')

ZeroDivisionError: division by zero
```

In [36]:
```
try:
    a=1/0
    #a=3
    #open('sdfsd')
    #int('mitsos')
except ZeroDivisionError:
    print ('Daireses me to 0')
except FileNotFoundError:
    print ('DEN YPARXEI TO ARXEIO!!!!!')
except Exception as e:
    print ('Kati allo kako sunebei: ', str(e))
else:
    print ('Everything was ok!')
finally:
    # Clean up code
    print ('FINALLY')

print ('aaa')
```

```
Daireses me to 0
FINALLY
aaa
```

In [40]:
```
def f(x):

    if not type(x) is int:
        raise Exception('ONLY FLOATS PLEASE!')

    return x*2
```

In [41]:
```
f(3.2)
```

```
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
<ipython-input-41-8442a3e670e8> in <module>
----> 1 f(3.2)

<ipython-input-40-cb498a4e40ae> in f(x)
      2
      3     if not type(x) is int:
----> 4         raise Exception('ONLY FLOATS PLEASE!')
      5
      6     return x*2

Exception: ONLY FLOATS PLEASE!
```

In [44]:
```
def f(x):

    if not type(x) is int:
        raise NotImplementedError('ONLY FLOATS PLEASE!')

    return x*2
```

In [45]:
```
f(3.2)
```

```
---------------------------------------------------------------------------
NotImplementedError                       Traceback (most recent call last)
<ipython-input-45-8442a3e670e8> in <module>
----> 1 f(3.2)

<ipython-input-44-832e5e363355> in f(x)
      2
```

```
      3     if not type(x) is int:
----> 4         raise NotImplementedError('ONLY FLOATS PLEASE!')
      5
      6     return x*2

NotImplementedError: ONLY FLOATS PLEASE!
```

In [46]:
```
def g():
    yield 1
    yield 2
    yield 3


list(g())
```

Out[46]:
```
[1, 2, 3]
```

In [47]:
```
gen = g()

while True:
    try:
        a = next(gen)
    except StopIteration:
        break
    print(a)
```

```
1
2
3
```

In [48]:
```
from collections import Counter
```

In [49]:
```
Counter('asdlkfjhgasldjfasldjfhlakdsgfhlsdkjfghlsdkjfghlsdkjfhgsldkjfgh')
```

Out[49]:
```
Counter({'a': 4,
         's': 8,
         'd': 8,
         'l': 8,
         'k': 6,
         'f': 8,
         'j': 7,
         'h': 7,
         'g': 6})
```

In [50]:
```
Counter([1,2,2,3,4,5,4,5,6,7,8,7,6,5,4,2])
```

Out[50]:
```
Counter({1: 1, 2: 3, 3: 1, 4: 3, 5: 3, 6: 2, 7: 2, 8: 1})
```

In [51]:
```
a = Counter('Mitsos')
print (a)
```

```
Counter({'s': 2, 'M': 1, 'i': 1, 't': 1, 'o': 1})
```

In [52]:
```
b = Counter('Kostas')
print (b)
```

```
Counter({'s': 2, 'K': 1, 'o': 1, 't': 1, 'a': 1})
```

In [53]:
```
a+b
```

Out[53]:
```
Counter({'M': 1, 'i': 1, 't': 2, 's': 4, 'o': 2, 'K': 1, 'a': 1})
```

In [56]:
```
from collections import defaultdict
```

In [57]:
```
a = defaultdict(int)
b = {}
```

In [58]:
```
a['mitsos']
```

Out[58]:
```
0
```

In [59]:
```
b['mitsos']
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-59-4edd9cb0949e> in <module>
----> 1 b['mitsos']

KeyError: 'mitsos'
```

In [60]:
```
a['kostas'] += 1
```

In [61]:
```
a['kostas']
```

Out[61]:
```
1
```

In [62]:
```
a = defaultdict(list)
```

In [63]:
```
a['kostas']
```

Out[63]:
```
[]
```

In [64]:
```
a['mitsos'].append(3)
```

In [65]:
```
a['mitsos']
```

Out[65]:
```
[3]
```

In [66]:
```
Counter('kostas' + 'Mitsos')
```

Out[66]:
```
Counter({'k': 1, 'o': 2, 's': 4, 't': 2, 'a': 1, 'M': 1, 'i': 1})
```

In [67]:
```
!pwd
```

```
/Users/admin/Downloads
```

In [ ]: