lecture_19

October 16, 2022

1 Lecture 19

[1]:
```
0.1 + 0.2
```

[1]:
```
0.30000000000000004
```

[2]:
```
from decimal import Decimal
```

[5]:
```
x = Decimal("0.1") + Decimal("0.2")
```

[7]:
```
float(x)
```

[7]:
```
0.3
```

1.1 `map()` and `zip()` and `filter()`

[8]:
```
a = [12, 24, 42]
b = [1, 2, 3, 4]
```

[9]:
```
for x, y in zip(a, b):
    print(x, y)
```

```
12 1
24 2
42 3
```

[11]:
```
a = [12, 24, 42]
b = {1, 2, 3, 4}
```

[12]:
```
for x, y in zip(a, b):
    print(x, y)
```

```
12 1
24 2
42 3
```

[13]:
```
for i, x in zip(range(10), a):
    print(i, x)
```

```
0 12
1 24
2 42
```

[14]:
```
for x, y in zip("test", a):
    print(x, y)
```

```
t 12
e 24
s 42
```

[15]:
```
for x, y in zip(10, a):
    print(x, y)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [15], in <cell line: 1>()
----> 1 for x, y in zip(10, a):
      2     print(x, y)

TypeError: 'int' object is not iterable
```

[16]:
```
c = ("test", "this", "functionality")
```

[17]:
```
for x, y, z in zip(a, b, c):
    print(x, y, z)
```

```
12 1 test
24 2 this
42 3 functionality
```

[22]:
```
t = list(zip(a, b))
```

[24]:
```
x, y = t[0]
```

[25]:
```
x
```

[25]: 12

[26]:
```
y
```

[26]: 1

[27]:
```
for x, y in zip(a, b):
    print(x, y)
```

```
12 1
24 2
42 3
```

[28]:
```
a
```

[28]:
```
[12, 24, 42]
```

[29]:
```
def foo(x):
    return x**x
```

[30]:
```
for i in a:
    print(foo(i))
```

```
8916100448256
133373577685028412444908147284377 6
150130937545296572356771972164254457814047970568738777235893533016064
```

[31]:
```
[foo(i) for i in a]
```

[31]:
```
[8916100448256,
 133373577685028412444908147284377 6,
 150130937545296572356771972164254457814047970568738777235893533016064]
```

[33]:
```
for value in map(foo, a):
    print(value)
```

```
8916100448256
133373577685028412444908147284377 6
150130937545296572356771972164254457814047970568738777235893533016064
```

[34]:
```
def bar(x, y):
    return x**y
```

[35]:
```
a
```

[35]:
```
[12, 24, 42]
```

[36]:
```
b
```

[36]:
```
{1, 2, 3, 4}
```

[37]:
```
for value in map(bar, a, b):
    print(value)
```

```
12
576
74088
```

[40]:
```
a
```

[40]:
```
[12, 24, 42]
```

[41]:
```
b
```

[41]:
```
{1, 2, 3, 4}
```

[42]:
```
c
```

[42]:
```
('test', 'this', 'functionality')
```

[43]:
```
for i in b:
    if i % 2 == 0:
        print(i)
```

```
2
4
```

[44]:
```
def is_even(x):
    return x % 2 == 0
```

[45]:
```
is_even(10)
```

[45]:
```
True
```

[46]:
```
is_even(11)
```

[46]:
```
False
```

[47]:
```
for i in filter(is_even, b):
    print(i)
```

```
2
4
```

[48]:
```
def even_reminder(x):
    return x % 2
```

[49]:
```
even_reminder(10)
```

[49]:
```
0
```

[51]:
```
even_reminder(11)
```

[51]:
```
1
```

[52]:
```
for i in filter(even_reminder, b):
    print(i)
```

```
1
3
```

[55]:
```
d = [0, 1, "", "test"]
```

[56]:
```
list(filter(None, d))
```

[56]:
```
[1, 'test']
```

[57]:
```
a = [1, 2, 1, 3, 1, 4, 5, 6]
b = [0, 2, 1, 2, 3, 5, 5, 1]
```

[58]:
```
def count_pairs(d_1, d_2):
    pass
```

[60]:
```
def is_equal(x, y):
    return x == y
```

[61]:
```
sum(map(is_equal, a, b))
```

[61]:
```
3
```

[62]:
```
from operator import eq
```

[63]:
```
sum(map(eq, a, b))
```

[63]:
```
3
```

[66]:
```
sum(map(lambda x, y: x == y, a, b))
```

[66]:
```
3
```

[67]:
```
c = sum(x == y for x, y in zip(a, b))
```

[68]:
```
c
```

[68]:
```
3
```

[71]:
```
a = ["RaNdOM", "LettERS", "tEsT"]
```

[72]:
```
t = "RaNdOM"
```

[73]:
```
t.lower()
```

[73]: 'random'

[74]:
```
for t in map(str.lower, a):
    print(t)
```

```
random
letters
test
```

[76]:
```
for t in map(str, [1, 2, 3]):
    print(t, type(t))
```

```
1 <class 'str'>
2 <class 'str'>
3 <class 'str'>
```

[77]:
```
a = [1, 2, 0, 42]
```

[78]:
```
all(a)
```

[78]: False

[79]:
```
any(a)
```

[79]: True

[80]:
```
b = [1, 2, 3]
```

[81]:
```
all(b)
```

[81]: True

[82]:
```
any(b)
```

[82]: True

[86]:
```
c = [0, "", False, None, [], {}]
```

[87]:
```
all(c)
```

[87]: False

[88]:
```
any(c)
```

[88]: False

[89]:
```
all([])
```

[89]:
```
True
```

[90]:
```
any([])
```

[90]:
```
False
```

[94]:
```
a = [1, 2, 3, 4, 5, 6]
b = [6, 6, 6, 6]
```

[95]:
```
all(i == 6 for i in a)
```

[95]:
```
False
```

[96]:
```
all(i == 6 for i in b)
```

[96]:
```
True
```

[98]:
```
6 in a
```

[98]:
```
True
```

[101]:
```
def our_any(iterable):
    for element in iterable:
        if element:
            return True
    return False
```

[102]:
```
our_any(c)
```

[102]:
```
False
```

[103]:
```
def our_all(iterable):
    for element in iterable:
        if not element:
            return False
    return True
```

[104]:
```
our_all(c)
```

[104]:
```
False
```

[106]:
```
f = lambda x: x**2
```

[108]:
```
f(2)
```

[108]: 4

[109]:
```
def f(x):
    return x**2
```

[110]:
```
a = [1, 2, 3, 4, 5]
```

[111]:
```
for i in map(lambda x: x**2, a):
    print(i)
```

```
1
4
9
16
25
```

[112]:
```
a = [1, 2, 1, 3, 1, 4, 5, 6]
b = [0, 2, 1, 2, 3, 5, 5, 1]
```

[113]:
```
for i in map(lambda x, y: x*y, a, b):
    print(i)
```

```
0
4
1
6
3
20
25
6
```

[115]:
```
a = [1, 2, 1, 3, 1, 4, 5, 6]
```

[123]:
```
for i in filter(lambda x: not x % 2, a):
    print(i)
```

```
2
4
6
```

[125]:
```
x
```

[125]: 42

[126]:
```
x % 2
```

[126]: 0

[127]:
```
not 0
```

[127]:
```
True
```

[128]:
```
not 42
```

[128]:
```
False
```

[ ]:
```

```