

```
In [16]: students = [  
    'student 1',  
    'student 2'  
]  
  
import random  
def pick():  
    return random.choice(students)
```

```
In [453]: def pick2():  
    while True:  
        for student in random.shuffle(students):  
            yield student
```

```
In [429]: a = [1,2,3,4]  
random.shuffle(a)  
print (a)  
  
[3, 1, 4, 2]
```

```
In [452]: a = [1,2,3,4]  
print (random.choice(a))  
  
2
```

```
In [ ]:
```

```
In [390]: gen = pick2()
```

```
In [391]: for i in range(100):  
           print (next(gen))
```

Πολίνα  
Ιπποκράτης  
Αιμίλιος  
Νίκος  
Δανάη  
Αθανασία  
Έλσα  
Γιάννης  
Πολίνα  
Ιπποκράτης  
Αιμίλιος  
Νίκος  
Δανάη  
Αθανασία  
Έλσα  
Γιάννης  
Πολίνα  
Ιπποκράτης  
Αιμίλιος  
Νίκος  
Δανάη  
Αθανασία  
Έλσα  
Γιάννης  
Πολίνα  
Ιπποκράτης  
Αιμίλιος  
Νίκος  
Δανάη  
Αθανασία  
Έλσα  
Γιάννης  
Πολίνα  
Ιπποκράτης  
Αιμίλιος  
Νίκος  
Δανάη  
Αθανασία  
Έλσα  
Γιάννης  
Πολίνα  
Ιπποκράτης  
Αιμίλιος  
Νίκος  
Δανάη  
Αθανασία  
Έλσα  
Γιάννης  
Πολίνα  
Ιπποκράτης  
Αιμίλιος  
Νίκος  
Δανάη  
Αθανασία  
Έλσα  
Γιάννης  
Πολίνα  
Ιπποκράτης  
Αιμίλιος  
Νίκος  
Δανάη  
Αθανασία  
Έλσα  
Γιάννης  
Πολίνα  
Ιπποκράτης  
Αιμίλιος  
Νίκος  
Δανάη  
Αθανασία  
Έλσα  
Γιάννης  
Πολίνα

```
In [ ]: 
```

```
In [380]: next(gen)
```

```
Out[380]: 'Πολίνα'
```

```
In [381]: next(gen)
```

```
Out[381]: 'Ιπποκράτης'
```

```
In [382]: next(gen)
```

```
Out[382]: 'Αιμίλιος'
```

```
In [383]: next(gen)
```

```
Out[383]: 'Νίκος'
```

```
In [384]: next(gen)
```

```
Out[384]: 'Δανάη'
```

```
In [385]: next(gen)
```

```
Out[385]: 'Αθανασία'
```

```
In [386]: next(gen)
```

```
Out[386]: 'Έλσα'
```

```
In [387]: next(gen)
```

```
Out[387]: 'Γιάννης'
```

```
In [388]: next(gen)
```

```
-----
StopIteration                                Traceback (most recent call last)
<ipython-input-388-6e72e47198db> in <module>()
----> 1 next(gen)

StopIteration:
```

```
In [ ]: 
```

```
In [2]: pick()
```

```
Out[2]: 'Αθανασία'
```

```
In [15]: a = [4,1,2]
```

```
In [4]: sorted(a)
```

```
Out[4]: [1, 2, 4]
```

```
In [5]: a
```

```
Out[5]: [4, 1, 2]
```

```
In [6]: a.sort()
```

```
In [7]: a
```

```
Out[7]: [1, 2, 4]
```

```
In [8]: _
```

```
Out[8]: [1, 2, 4]
```

```
In [9]: 42
```

```
Out[9]: 42
```

```
In [10]: _
```

```
Out[10]: 42
```

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

```
Out[11]: [1, 2, 4]
```

```
In [12]: _
```

```
Out[12]: [1, 2, 4]
```

```
In [17]: sorted(a)
```

```
Out[17]: [1, 2, 4]
```

```
In [18]: a
```

```
Out[18]: [4, 1, 2]
```

```
In [20]: _
```

```
Out[20]: [4, 1, 2]
```

```
In [21]: a = [  
    ('kosta', 15),  
    ('manoli', 13),  
    ('maria', 20),  
    ]
```

```
In [22]: min(a)
```

```
Out[22]: ('kosta', 15)
```

```
In [23]: pick()
```

```
Out[23]: 'Νίκος'
```

```
In [24]: [(grade, name) for name, grade in a]
```

```
Out[24]: [(15, 'kosta'), (13, 'manoli'), (20, 'maria')]
```

```
In [25]: min([(grade, name) for name, grade in a])
```

```
Out[25]: (13, 'manoli')
```

```
In [26]: a
```

```
Out[26]: [('kosta', 15), ('manoli', 13), ('maria', 20)]
```

```
In [32]: def f(item):  
         return item[1]
```

```
In [31]: pick()
```

```
Out[31]: 'Αιμίλιος'
```

```
In [30]: a[0]
```

```
Out[30]: ('kosta', 15)
```

```
In [33]: f(a[0])
```

```
Out[33]: 15
```

```
In [36]: min(a, key=f)
```

```
Out[36]: ('manoli', 13)
```

```
In [38]: pick()
```

```
Out[38]: 'Νίκος'
```

```
In [ ]:
```

```
In [35]: a
```

```
Out[35]: [('kosta', 15), ('manoli', 13), ('maria', 20)]
```

```
In [37]: sorted(a, key=f)
```

```
Out[37]: [('manoli', 13), ('kosta', 15), ('maria', 20)]
```

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

```
Out[41]: ('manoli', 13)
```

```
In [43]: pick()
```

```
Out[43]: 'Νίκος'
```

```
In [ ]:
```

```
In [42]: students
```

```
Out[42]: ['Πολίνα',  
          'Ιπποκράτης',  
          'Αιμίλιος',  
          'Νίκος',  
          'Δανάη',  
          'Αθανασία',  
          'Έλσα',  
          'Γιάννης']
```

```
In [44]: sorted(students)
```

```
Out[44]: ['Ελσα',  
          'Αθανασία',  
          'Αιμίλιος',  
          'Γιάννης',  
          'Δανάη',  
          'Ιπποκράτης',  
          'Νίκος',  
          'Πολίνα']
```

```
In [75]: pick()
```

```
Out[75]: 'Νίκος'
```

```
In [76]: sorted(students, key=len)
```

```
Out[76]: ['Ελσα',  
          'Νίκος',  
          'Δανάη',  
          'Πολίνα',  
          'Γιάννης',  
          'Αιμίλιος',  
          'Αθανασία',  
          'Ιπποκράτης']
```

```
In [78]: sorted(students, key=len, reverse=True)
```

```
Out[78]: ['Ιπποκράτης',  
          'Αιμίλιος',  
          'Αθανασία',  
          'Γιάννης',  
          'Πολίνα',  
          'Νίκος',  
          'Δανάη',  
          'Ελσα']
```

```
In [79]: a
```

```
Out[79]: [('kosta', 15), ('manoli', 13), ('maria', 20)]
```

```
In [80]: def f(item):  
         return item[1]
```

```
In [ ]: sorted(a, key=f)
```

```
In [81]: lambda item : item[1]
```

```
Out[81]: <function __main__.<lambda>(item)>
```

```
In [84]: (lambda item : item[1])([2,3])
```

```
Out[84]: 3
```

```
In [ ]:
```

```
In [82]: pick()
```

```
Out[82]: 'Πολίνα'
```

```
In [83]: f([2,3])
```

```
Out[83]: 3
```

```
In [85]: f = lambda item : item[1]
```

```
In [86]: min(a, key=lambda item : item[1])
```

```
Out[86]: ('manoli', 13)
```

```
In [93]: min(a, key=lambda item : item[1])
```

```
Out[93]: ('manoli', 13)
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [87]: students
```

```
Out[87]: ['Πολίνα',  
          'Ιπποκράτης',  
          'Αιμίλιος',  
          'Νίκος',  
          'Δανάη',  
          'Αθανασία',  
          'Έλσα',  
          'Γιάννης']
```

```
In [88]: pick()
```

```
Out[88]: 'Δανάη'
```

```
In [89]: sorted(students, key=lambda item:item[1])
```

```
Out[89]: ['Νίκος',  
          'Δανάη',  
          'Αθανασία',  
          'Αιμίλιος',  
          'Γιάννης',  
          'Έλσα',  
          'Πολίνα',  
          'Ιπποκράτης']
```

```
In [94]: a=[4,2,3]
```

```
In [95]: pick()
```

```
Out[95]: 'Έλσα'
```

```
In [100]: new_list = [i*2 for i in a]
```

```
In [ ]:
```

```
In [97]: new_list = []  
         for i in a:  
             new_list.append(i*2)
```

```
In [98]: new_list
```

```
Out[98]: [8, 4, 6]
```

```
In [101]: def g(x):  
          return x*2
```

```
In [102]: a
```

```
Out[102]: [4, 2, 3]
```

```
In [105]: map(g, a)
```

```
Out[105]: <map at 0x10da55748>
```

```
In [104]: list(map(g, a))
```

```
Out[104]: [8, 4, 6]
```

```
In [106]: list(map(lambda x :x*2 , a))
```

```
Out[106]: [8, 4, 6]
```

```
In [108]: what = lambda x: lambda y:y+x
```

```
In [110]: what(3)(4)
```

```
Out[110]: 7
```

## Functional programming

### Generators

```
In [111]: def f():  
          return 4
```

```
In [115]: pick()
```

```
Out[115]: 'Νίκος'
```

```
In [113]: f()
```

```
Out[113]: 4
```

```
In [114]: def f():  
          return 4  
          return 5
```

```
In [116]: f()
```

```
Out[116]: 4
```

```
In [117]: f()
```

```
Out[117]: 4
```



```
In [118]: def g():
           yield 4
           yield 5
```

```
In [119]: gen = g()
```

```
In [120]: type(gen)
```

```
Out[120]: generator
```

```
In [121]: next(gen)
```

```
Out[121]: 4
```

```
In [122]: next(gen)
```

```
Out[122]: 5
```

```
In [123]: next(gen)
```

```
-----
StopIteration                                Traceback (most recent call last)
<ipython-input-123-6e72e47198db> in <module>()
----> 1 next(gen)

StopIteration:
```

```
In [124]: gen
```

```
Out[124]: <generator object g at 0x10d992930>
```

```
In [285]: import math

def is_prime(x):
    if x in [1,2,3]:
        return True

    for i in range(2, int(math.sqrt(x))+1):
        if not x%i:
            return False
    return True

def prime_generator():
    start = 1
    #while start<100:
    while True:
        if is_prime(start):
            yield start
        start += 1
```

```
In [135]: is_prime(24)
```

```
Out[135]: False
```

```
In [256]: gen = prime_generator()
```

```
In [283]: next(gen)
```

```
-----  
StopIteration                                Traceback (most recent call last)  
<ipython-input-283-6e72e47198db> in <module>()  
----> 1 next(gen)  
  
StopIteration:
```

```
In [181]: next(gen)
```

```
Out[181]: 2
```

```
In [182]: next(gen)
```

```
Out[182]: 3
```

```
In [183]: next(gen)
```

```
Out[183]: 5
```

```
In [184]: next(gen)
```

```
Out[184]: 7
```

```
In [254]: next(gen)
```

```
Out[254]: 373
```

```
In [286]: gen = prime_generator()  
first_100_p = [next(gen) for x in range(100)]
```

```
In [288]: primers_lower_100 = []  
gen = prime_generator()  
while True:  
    p = next(gen)  
    if p>100:  
        break  
  
    primers_lower_100.append(p)
```

```
In [289]: primers_lower_100
```

```
Out[289]: [1,  
2,  
3,  
5,  
7,  
11,  
13,  
17,  
19,  
23,  
29,  
31,  
37,  
41,  
43,  
47,  
53,  
59,  
61,  
67,  
71,  
73,  
79,  
83,  
89,  
97]
```

```
In [296]: pick()
```

```
Out[296]: 'Γιάβνης'
```

```
In [302]: i=0  
while i<20:  
  
    if i>5:  
        break  
    i += 1  
    print (i)
```

```
1  
2  
3  
4  
5  
6
```

```
In [301]: i=0
while i<20:
    i += 1
    if i<5:
        continue

    print (i)
```

5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20

In [ ]:

In [299]: pick()

Out[299]: 'Νίκος'

In [ ]:

In [294]: 1 list(range(1,21))

Out[294]: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]

In [290]: pick()

Out[290]: 'Ιπποκράτης'

```
In [305]: i=0
while i<20:

    if i>10:
        break
    i += 1
    print (i)
else:
    print ('HIT BREAK')
```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11

```
In [306]: i=0
while i<20:

    if i>50:
        break
    i += 1
    print (i)
else:
    print ('HIT BREAK')
```

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
HIT BREAK
```

```
In [307]: first_100_p
```

```
Out[307]: [1,  
2,  
3,  
5,  
7,  
11,  
13,  
17,  
19,  
23,  
29,  
31,  
37,  
41,  
43,  
47,  
53,  
59,  
61,  
67,  
71,  
73,  
79,  
83,  
89,  
97,  
101,  
103,  
107,  
109,  
113,  
127,  
131,  
137,  
139,  
149,  
151,  
157,  
163,  
167,  
173,  
179,  
181,  
191,  
193,  
197,  
199,  
211,  
223,  
227,  
229,  
233,  
239,  
241,  
251,  
257,  
263,  
269,  
271,  
277,  
281,  
283,  
293,  
307,  
311,  
313,
```

```
In [308]: abs(-3)
```

```
Out[308]: 3
```

```
In [310]: abs(-1212.1232)
```

```
Out[310]: 1212.1232
```

```
In [311]: age=40
          if age>=18:
              status='adult'
          else:
              status='underage'
```

```
In [313]: status
```

```
Out[313]: 'adult'
```

```
In [318]: status = 'adult' if age>=18 else 'underage'
```

```
In [321]: status
```

```
Out[321]: 'adult'
```

```
In [315]: status = lambda age: 'adult' if age>=18 else 'underage'
```

```
In [317]: status(10)
```

```
Out[317]: 'underage'
```

```
In [326]: age = input('what is your age?  ')
```

```
what is your age?  1231234
```

```
In [324]: int(age)
```

```
Out[324]: 1234
```

```
In [327]: a=3
```

```
In [329]: a
```

```
Out[329]: 3
```

```
In [330]: del a
```

```
In [331]: a
```

```
-----
NameError                                Traceback (most recent call last)
<ipython-input-331-3f786850e387> in <module>()
----> 1 a

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

```
In [332]: a = [1,2,3,4]
```

```
In [333]: del a[-1]
```

```
In [334]: a
```

```
Out[334]: [1, 2, 3]
```

```
In [335]: a = {
            'name': 'mitsos',
            'age': 50
          }
```

```
In [336]: a
```

```
Out[336]: {'name': 'mitsos', 'age': 50}
```

```
In [337]: del a['name']
```

```
In [338]: a
```

```
Out[338]: {'age': 50}
```

```
In [344]: def f():
            a = list(range(100000000))
            return 42
```

```
In [343]: f()
```

```
Out[343]: 42
```

```
In [358]: def f(age):
            assert 0<=age<=100 # TI HLIKIA EINAI AUTH RE?
            assert type(age) is int # LABOS TYPO!
            return 'adult' if age>=18 else 'underage'
```

```
In [349]: f(50)
```

```
Out[349]: 'adult'
```

```
In [351]: f(14)
```

```
Out[351]: 'underage'
```

```
In [356]: f(200)
```

```
-----
AssertionError                                Traceback (most recent call last)
<ipython-input-356-a403249833e6> in <module>()
----> 1 f(200)

<ipython-input-355-93f5b89bce83> in f(age)
      1 def f(age):
----> 2     assert 0<=age<=100 # TI HLIKIA EINAI AUTH RE?
      3     assert type(age) is int
      4     return 'adult' if age>=18 else 'underage'

AssertionError:
```



In [359]: `f(14.3)`

```
-----
AssertionError                                Traceback (most recent call last)
<ipython-input-359-76580c14de57> in <module>()
----> 1 f(14.3)

<ipython-input-358-7f57e1af76ea> in f(age)
      1 def f(age):
      2     assert 0<=age<=100 # TI HLIKIA EINAI AUTH RE?
----> 3     assert type(age) is int # LA8OS TYPO!
      4     return 'adult' if age>=18 else 'underage'

AssertionError:
```

In [360]: `gen = prime_generator()`

In [361]: `next(gen)`

Out[361]: 1

In [362]: `next(gen)`

Out[362]: 2

In [363]: `def f():
 for i in range(10):
 yield i`

In [364]: `gen = f()`

In [366]: `next(gen)`

Out[366]: 0

In [368]: `next(gen)`

Out[368]: 1

In [369]: `list(gen)`

Out[369]: [2, 3, 4, 5, 6, 7, 8, 9]

In [372]: `next(gen)`

```
-----
StopIteration                                Traceback (most recent call last)
<ipython-input-372-6e72e47198db> in <module>()
----> 1 next(gen)

StopIteration:
```

In [373]: `gen = f()`

In [374]: `next(gen)`

Out[374]: 0

```
In [375]: next(gen)
```

```
Out[375]: 1
```

```
In [376]: next(gen)
```

```
Out[376]: 2
```

```
In [377]: list(gen)
```

```
Out[377]: [3, 4, 5, 6, 7, 8, 9]
```

```
In [ ]:
```

```
In [465]: pick()
```

```
Out[465]: 'Ελσα'
```

## Generator comprehension

```
In [466]: def f():  
           for i in range(10):  
               yield i
```

```
In [472]: gen = (i for i in range(10))
```

```
In [468]: next(gen)
```

```
Out[468]: 0
```

```
In [469]: next(gen)
```

```
Out[469]: 1
```

```
In [470]: next(gen)
```

```
Out[470]: 2
```

```
In [471]: list(gen)
```

```
Out[471]: [3, 4, 5, 6, 7, 8, 9]
```

```
In [480]: gen = (i for i in range(10,25,3))
```

```
In [474]: list(gen)
```

```
Out[474]: [10, 13, 16, 19, 22]
```

```
In [477]: list(enumerate(gen))
```

```
Out[477]: [(0, 10), (1, 13), (2, 16), (3, 19), (4, 22)]
```

```
In [481]: for i, item in enumerate(gen):
           if i%2==0:
               continue

           print (item)
```

13  
19

## Files

```
In [523]: with open('test.txt', 'w') as f:
           f.write('Mitsos\n')
```

```
In [524]: !cat test.txt
```

Mitsos

```
In [525]: !cat t.txt
```

hello  
my name is Alex

```
In [526]: with open('t.txt', 'r') as f:
           content = f.read()
           content
```

Out[526]: 'hello\nmy name is Alex\n'

```
In [527]: print (content)
```

hello  
my name is Alex

```
In [538]: with open('DMET_14GreekGenomes_chr4.vcf', 'r') as f:
           content=f.read()
```

```
In [539]: type(content)
```

Out[539]: str

```
In [532]: with open('DMET_14GreekGenomes_chr4.vcf', 'r') as f:
           for line in f:
               #print (line)
               pass
```

```
In [536]: with open('DMET_14GreekGenomes_chr4.vcf', 'r') as f:
           print (f.readline().strip())
           print (f.readline().strip())
```

```
##fileformat=VCFv4.1
#CHROM  POS      ID      REF      ALT      QUAL      FILTER  INFO      FORMAT  SAMPLE_
1       SAMPLE_2      SAMPLE_3      SAMPLE_4      SAMPLE_5      SAMPLE_
6       SAMPLE_7      SAMPLE_8      SAMPLE_9      SAMPLE_10     SAMPLE_
11      SAMPLE_12     SAMPLE_13     SAMPLE_14
```

```
In [540]: with open('DMET_14GreekGenomes_chr4.vcf', 'r') as f:
           content = f.readlines()
```

In [ ]:

```
In [529]: content
```

[illegible]

```
In [576]: f = open('DMET_14GreekGenomes_chr4.vcf', 'r')
```

```
In [549]: content = f.readlines()
```

```
In [550]: len(content)
```

Out[550]: 26565

```
In [551]: content = f.readlines()
```

```
In [552]: len(content)
```

Out[552]: 0

```
In [553]: f.read()
```

Out[553]: ''

```
In [554]: f.readline()
```

Out[554]: ''

```
In [555]: f.readlines()
```

```
Out[555]: []
```

```
In [556]: f.close()
```

```
In [578]: with open('test.txt', 'r') as f1, open('test2.txt', 'w') as f2:
           for l in f1:
               f2.write(l.strip() + ' asdasd\n')
```

```
In [579]: !cat test2.txt
```

Mitsos asdasD

```
In [580]: ! wget -O Wq2PpbZy.txt https://pastebin.com/raw/Wq2PpbZy
```

```
--2019-11-12 11:48:25-- https://pastebin.com/raw/Wq2PpbZy (https://pastebin.co
m/raw/Wq2PpbZy)
Resolving pastebin.com (pastebin.com)... 104.22.3.84, 104.22.2.84
Connecting to pastebin.com (pastebin.com)|104.22.3.84|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/plain]
Saving to: 'Wq2PpbZy.txt'

Wq2PpbZy.txt          [ <=>          ]  4.91K  --.-KB/s   in 0.001s

2019-11-12 11:48:26 (8.91 MB/s) - 'Wq2PpbZy.txt' saved [5027]
```

```
In [582]: with open('Wq2PpbZy.txt', 'r') as f:
          content = f.read()
```

```
In [584]: print (content)
```

Sepal length	Sepal width	Petal length	Petal width	Species
5.2	3.5	1.4	0.2	I. setosa
4.9	3.0	1.4	0.2	I. setosa
4.7	3.2	1.3	0.2	I. setosa
4.6	3.1	1.5	0.2	I. setosa
5.0	3.6	1.4	0.3	I. setosa
5.4	3.9	1.7	0.4	I. setosa
4.6	3.4	1.4	0.3	I. setosa
5.0	3.4	1.5	0.2	I. setosa
4.4	2.9	1.4	0.2	I. setosa
4.9	3.1	1.5	0.1	I. setosa
5.4	3.7	1.5	0.2	I. setosa
4.8	3.4	1.6	0.2	I. setosa
4.8	3.0	1.4	0.1	I. setosa
4.3	3.0	1.1	0.1	I. setosa
5.8	4.0	1.2	0.2	I. setosa
5.7	4.4	1.5	0.4	I. setosa
5.4	3.9	1.3	0.4	I. setosa
5.1	3.5	1.4	0.3	I. setosa
5.7	3.8	1.7	0.3	I. setosa

```
In [585]: c = content.split('\n')
```

```
In [588]: c = [line.split('\t') for line in content.split('\n')]
```

```
In [589]: c[30]
```

```
Out[589]: ['4.7 ', '3.2 ', '1.6 ', '0.2 ', 'I. setosa']
```

```
In [591]: header = c[0]
```

```
In [593]: header = [x.strip() for x in header]
          header
```

```
Out[593]: ['Sepal length', 'Sepal width', 'Petal length', 'Petal width', 'Species']
```

```
In [595]: c = c[1:]
```

```
In [597]: [float(line[0]) for line in c if line[-1] == 'I. setosa']
```

```
Out[597]: [5.2,  
4.9,  
4.7,  
4.6,  
5.0,  
5.4,  
4.6,  
5.0,  
4.4,  
4.9,  
5.4,  
4.8,  
4.8,  
4.3,  
5.8,  
5.7,  
5.4,  
5.1,  
5.7,  
5.1,  
5.4,  
5.1,  
4.6,  
5.1,  
4.8,  
5.0,  
5.0,  
5.2,  
5.2,  
4.7,  
4.8,  
5.4,  
5.2,  
5.5,  
4.9,  
5.0,  
5.5,  
4.9,  
4.4,  
5.1,  
5.0,  
4.5,  
4.4,  
5.0,  
5.1,  
4.8,  
5.1,  
4.6,  
5.3,  
5.0]
```

In [598]: c

```
Out[598]: [['5.2 ', '3.5 ', '1.4 ', '0.2 ', 'I. setosa'],
 ['4.9 ', '3.0 ', '1.4 ', '0.2 ', 'I. setosa'],
 ['4.7 ', '3.2 ', '1.3 ', '0.2 ', 'I. setosa'],
 ['4.6 ', '3.1 ', '1.5 ', '0.2 ', 'I. setosa'],
 ['5.0 ', '3.6 ', '1.4 ', '0.3 ', 'I. setosa'],
 ['5.4 ', '3.9 ', '1.7 ', '0.4 ', 'I. setosa'],
 ['4.6 ', '3.4 ', '1.4 ', '0.3 ', 'I. setosa'],
 ['5.0 ', '3.4 ', '1.5 ', '0.2 ', 'I. setosa'],
 ['4.4 ', '2.9 ', '1.4 ', '0.2 ', 'I. setosa'],
 ['4.9 ', '3.1 ', '1.5 ', '0.1 ', 'I. setosa'],
 ['5.4 ', '3.7 ', '1.5 ', '0.2 ', 'I. setosa'],
 ['4.8 ', '3.4 ', '1.6 ', '0.2 ', 'I. setosa'],
 ['4.8 ', '3.0 ', '1.4 ', '0.1 ', 'I. setosa'],
 ['4.3 ', '3.0 ', '1.1 ', '0.1 ', 'I. setosa'],
 ['5.8 ', '4.0 ', '1.2 ', '0.2 ', 'I. setosa'],
 ['5.7 ', '4.4 ', '1.5 ', '0.4 ', 'I. setosa'],
 ['5.4 ', '3.9 ', '1.3 ', '0.4 ', 'I. setosa'],
 ['5.1 ', '3.5 ', '1.4 ', '0.3 ', 'I. setosa'],
 ['5.7 ', '3.8 ', '1.7 ', '0.3 ', 'I. setosa'],
 ['5.1 ', '3.6 ', '1.5 ', '0.3 ', 'I. setosa']]
```

In [601]: d = {h: [float(line[i]) for line in c for i, h in enumerate(header[:-1])]}

In [602]: d.keys()

Out[602]: dict\_keys(['Sepal length', 'Sepal width', 'Petal length', 'Petal width'])

In [603]: d['Sepal width']

```
Out[603]: [3.5,
 3.0,
 3.2,
 3.1,
 3.6,
 3.9,
 3.4,
 3.4,
 2.9,
 3.1,
 3.7,
 3.4,
 3.0,
 3.0,
 4.0,
 4.4,
 3.9,
 3.5,
 3.8,
 3.6]
```

```
In [610]: l = '-++++++-'

while '+-' in l or '-+' in l:
    l = l.replace('+-', '')
    l = l.replace('-+', '')

print(l)

+++++
```



```
In [4]: def f(s):  
        for x in s:  
            if not x in list('ACGT'):  
                return False  
        return len(s) % 3 == 0
```

```
In [6]: f('ACGGGE')
```

```
Out[6]: False
```

```
In [15]: import sys  
         sys.version
```

```
Out[15]: '3.7.0 (default, Jun 28 2018, 07:39:16) \n[Clang 4.0.1 (tags/RELEASE_401/fina  
1)]'
```

In [10]: `dir(sys)`

```
Out[10]: ['__breakpointhook__',
          '__displayhook__',
          '__doc__',
          '__excepthook__',
          '__interactivehook__',
          '__loader__',
          '__name__',
          '__package__',
          '__spec__',
          '__stderr__',
          '__stdin__',
          '__stdout__',
          '_clear_type_cache',
          '_current_frames',
          '_debugmallocstats',
          '_framework',
          '_getframe',
          '_git',
          '_home',
          '_xoptions',
          'abiflags',
          'api_version',
          'argv',
          'base_exec_prefix',
          'base_prefix',
          'breakpointhook',
          'builtin_module_names',
          'byteorder',
          'call_tracing',
          'callstats',
          'copyright',
          'displayhook',
          'dont_write_bytecode',
          'exc_info',
          'excepthook',
          'exec_prefix',
          'executable',
          'exit',
          'flags',
          'float_info',
          'float_repr_style',
          'get_asyncgen_hooks',
          'get_coroutine_origin_tracking_depth',
          'get_coroutine_wrapper',
          'getallocatedblocks',
          'getcheckinterval',
          'getdefaultencoding',
          'getdlopenflags',
          'getfilesystemencodingerrors',
          'getfilesystemencoding',
          'getprofile',
          'getrecursionlimit',
          'getrefcount',
          'getsizeof',
          'getswitchinterval',
          'gettrace',
          'hash_info',
          'hexversion',
          'implementation',
          'int_info',
          'intern',
          'is_finalizing',
          'last_traceback',
          'last_type',
          'last_value',
          'maxsize',
```

