1. **List filtering.**

Дан объект типа list():

*l = [1, 2, '3', 4, None, 10, 33, 'Python', -37.5]*

Релизуйте функцию которая отфильтрует только integer (int) значения из этого списка.

Напишите несколько вариантов решения:

* используя цикл for
* используя list comprehensions
* используя filter() + lambda

Пример:

def filter\_list(l):

return # [1, 2, 4, 10, 33]

Check yourself:

filter\_list([1,2,'a','b']) == [1,2]

filter\_list([1,'a','b',0,15]) == [1,0,15]

filter\_list([1,2,'aasf','1','123',123]) == [1,2,123]

1. **IP validation.**

Напишите функцию для валидации IP-адреса.

Пример:

def check\_ip(ip\_address):

return True/False

Напишите несколько вариантов решения:

* используя библиотеку re
* используя socket.inet\_aton

Check yourself

assert check\_ip('') is False

assert check\_ip('192.168.0.1') is True

assert check\_ip('0.0.0.1') is True

assert check\_ip('10.100.500.32') is False

assert check\_ip(700) is False

assert check\_ip('127.0.1') is True

1. **Working with 'csv' and 'json' structures.**

Имеется файл cars.csv

Используйте библиотеку csv что бы прочитать содержимое файла.

Конвертируйте данные в формат json и сохраните в файл cars.json

Примечания:

* используйте csv.DictReader
* используйте json.dump с параметром indent=2
* используйте контекстный менеджер with для создания файла

Check yourself:

bash$ cat ../task\_23/cars.json

[

{

"Year": "1997",

"Make": "Ford",

"Model": "E350"

},

{

"Year": "2000",

"Make": "Mercury",

"Model": "Cougar"

},

{

"Year": "2006",

"Make": "Dacia",

"Model": "Logan"

}

]

1. **Unix `find' on Python.**

По аналогии с Unix утилитой find, реализуйте модуль на Python для поиска файлов и директорий на файловой системе.

Обратите внимание на скрипт find\_util

bash$ ./find\_util /usr/ -name "\*.pyc" -type f

Где "\*.pyc" есть паттерн имени файл(ов) в формате shell-pattern

Примечания:

* используйте os.walk
* используйте os.path.join
* для проверки имени файла на соответствие паттерну используйте fnmatch.

1. **Unix 'ls -lh' on Python.**

По аналогии с командой ls -lh для Unix систем, реализуйте модуль на Python который отображает содержимое директории

Примечания:

* Чтобы получить содержимое директории используйте os.listdir
* Используйте os.stat чтобы получить информацию о каждом файле
* Используйте библиотеку prettytable

1. pip install prettytable
2. Имена столбцов: Mode, Owner, Group, Size, File name

* Используйте библиотеки pwd и grp чтобы получить имя пользователя и группу