

```
# используется для сортировки  
from operator import itemgetter
```

```
class Brw:
```

```
    """Браузер"""
```

```
    def __init__(self, id, memory, name, dvp, cmp_id):  
self.id = id  
    self.memory = memory #кол-во занимаемой памяти браузером(мб)  
self.name = name    self.dvp = dvp #разработчик    self.cmp_id =  
cmp_id
```

```
class Cmp:
```

```
    """Компьютер"""
```

```
    def __init__(self, id, name, type, os):  
self.id = id  
    self.name = name  
self.type = type  
    self.os = os
```

```
class CmpBrw:
```

```
    """
```

```
    'Браузеры компьютера' для реализации  
связи многие-ко-многим  
    """
```

```
    def __init__(self, brw_id, cmp_id):  
    self.cmp_id = cmp_id  
    self.brw_id= brw_id
```

```
cmps = [  
    Cmp(1, 'Alexei', 'ноутбук', 'Windows 11'),
```

```

    Cmp(2, 'PC Artem','ультрабук','Windows 10'),
    Cmp(3, 'Dmitriy','настольный','Linux'),
    Cmp(4, 'Anton PC','ноутбук','macOS Catalina'),
    Cmp(5, 'Anonim','сервер','Windows 11')
]

```

Компьютеры

```

brws = [
    Brw(1, 200,'Yandex', 'Яндекс',1),
    Brw(2, 350,'Chrome', 'Google',2),
    Brw(3, 300,'Firefox', 'Mozilla Foundation',4),
    Brw(4, 225,'Opera', 'Opera Software',5),
    Brw(5, 250,'Safari', 'Apple',3)
]

```

Браузеры

```

cmps_brws = [
    CmpBrw(1, 1),
    CmpBrw(1, 2),
    CmpBrw(3, 1),
    CmpBrw(3, 3),
    CmpBrw(3, 5),
    CmpBrw(2, 1),
    CmpBrw(2, 4),
    CmpBrw(5, 4),
    CmpBrw(4, 3),
    CmpBrw(5, 5),
]

```

def main():

"""Основная функция"""

Соединение данных один-ко-многим

one_to_many = [(b.name, b.memory, c.name)

for b **in** brws **for** c **in** cmps

if b.cmp_id == c.id]

```

    # Соединение данных многие-ко-многим
    many_to_many_temp = [(c.name, cb.cmp_id, cb.brw_id)
    for c in cmps          for cb in cmps_brws
                          if c.id == cb.cmp_id]

    many_to_many = [(b.name, b.memory, cmp_name)
    for cmp_name, cmp_id, brw_id in many_to_many_temp
    for b in brws if b.id == brw_id]

    print('Задание A1')
    res_11 = sorted(one_to_many, key=itemgetter(2))
    print(res_11)

    print('\nЗадание A2')
    res_12_unsorted = []
    # Перебираем все компьютеры
    for c in cmps:
        # Список браузеров компьютера
        c_brws = list(filter(lambda i: i[2] == c.name, one_to_many))
        # Если на компьютере установлен браузер
        if len(c_brws) > 0:
            # Памяти, занимаемые браузерами
            c_memories = [memory for _, memory, _ in c_brws]
            # Суммарная память, занимаемая всеми браузерами
            c_memories_sum = sum(c_memories)
            res_12_unsorted.append((c.name,
            c_memories_sum))

    # Сортировка по суммарной памяти
    res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)
    print(res_12)

    print('\nЗадание A3')
    res_13 = {}
    # Перебираем все компьютеры
    for c in cmps:
        # Список браузеров компьютера
        c_brws = list(filter(lambda i: i[2] == c.name, many_to_many))
        # Только название браузера

```

```
c_brws_names = [x for x, _, _ in c_brws]
# Добавляем результат в словарь
# ключ - компьютер, значение - список браузеров
res_13[c.name] = c_brws_names
```

```
print(res_13)
```

```
if __name__ == '__main__':
    main()
```

Результаты работы:

Задание A1

```
[('Yandex', 200, 'Alexei'), ('Opera', 225, 'Anonim'), ('Firefox', 300, 'Anton PC'),
('Safari', 250, 'Dmitriy'), ('Chrome', 350, 'PC Artem')]
```

Задание A2

```
[('PC Artem', 350), ('Anton PC', 300), ('Dmitriy', 250), ('Anonim', 225), ('Alexei',
200)]
```

Задание A3

```
{'Alexei': ['Yandex', 'Firefox', 'Chrome'], 'PC Artem': ['Yandex'], 'Dmitriy': ['Firefox',
'Opera'], 'Anton PC': ['Chrome', 'Safari'], 'Anonim': ['Firefox', 'Safari']}
```