

Текст программы

```
from operator import itemgetter

class Prog:
    """Класс программы"""

    def __init__(self, id, name, mem, comp_id):
        self.id = id
        self.name = name
        self.mem = mem
        self.comp_id = comp_id

class Comp:
    """Класс компьютера"""

    def __init__(self, id, name):
        self.id = id
        self.name = name

class Prog_Comp:
    """Класс для реализации связи многие ко многим"""

    def __init__(self, prog_id, comp_id):
        self.prog_id = prog_id
        self.comp_id = comp_id

progs = [Prog(1, 'Photoshop', 789, 1),
          Prog(2, 'Euro Truck Simulator', 2300, 3),
          Prog(3, 'Euro Truck Simulator 2', 2566, 3),
          Prog(4, 'Internet Explorer', 2, 1),
          Prog(5, 'Microsoft To Do', 40, 2)]

comps = [Comp(1, 'Домашний компьютер'),
          Comp(2, 'Рабочий компьютер'),
          Comp(3, 'PlayStation'),
          Comp(4, 'Планшет'),
          Comp(5, 'Xbox'),
          Comp(6, 'Бабушкин пк')]
```

```

comps_progs = [Prog_Comp(1, 1),
                Prog_Comp(2, 3),
                Prog_Comp(3, 3),
                Prog_Comp(4, 1),
                Prog_Comp(5, 2),

                Prog_Comp(1, 4),
                Prog_Comp(2, 5),
                Prog_Comp(4, 6)]

def main():
    one_to_many = [(p.name, p.mem, c.name)
                   for c in comps
                   for p in progs
                   if p.comp_id == c.id]

    many_to_many_temp = [(c.name, cp.comp_id, cp.prog_id)
                         for c in comps
                         for cp in comps_progs
                         if c.id == cp.comp_id]

    many_to_many = [(p.name, p.mem, c_name)
                    for c_name, c_id, pr_id in
many_to_many_temp
                    for p in progs
                    if p.id == pr_id]

    answer_1 = {}
    for c in comps:
        if 'компьютер' in c.name:
            c_progs = list(filter(lambda i: i[2] == c.name,
one_to_many))
            only_name = [x for x, _, _ in c_progs]
            answer_1[c.name] = only_name

    print('Задание E1 (присутствует слово "компьютер"):\n',
answer_1)

    answer_2 = []
    for c in comps:
        c_progs = list(filter(lambda i: i[2] == c.name,
one_to_many))
        if len(c_progs) > 0:

```

```

        c_mem = [mem for _, mem, _ in c_progs]
        sr_mem = round(sum(c_mem) / len(c_progs), 2)
        answer_2.append((c.name, sr_mem))
    print('Задание E2 (средний размер программы):\n',
sorted(answer_2, key=itemgetter(1)))

answer_3 = {}
for p in progs:
    if p.name[0] == 'E':
        p_comps = list(filter(lambda i: i[0] == p.name,
many_to_many))
        only_comp = [x for _, _, x in p_comps]
        answer_3[p.name] = only_comp

    print('Задание E3 (название начинается с буквы "E"):\n',
answer_3)

if __name__ == '__main__':
    main()

```

Результат выполнения

Задание E1 (присутствует слово "компьютер"):

{'Домашний компьютер': ['Photoshop', 'Internet Explorer'], 'Рабочий компьютер': ['Microsoft To Do']}

Задание E2 (средний размер программы):

[('Рабочий компьютер', 40.0), ('Домашний компьютер', 395.5), ('PlayStation', 2433.0)]

Задание E3 (название начинается с буквы "E"):

{'Euro Truck Simulator': ['PlayStation', 'Xbox'], 'Euro Truck Simulator 2': ['PlayStation']}