

20	Деталь	Поставщик
----	--------	-----------

Вариант Г.

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список всех отделов, у которых название начинается с буквы «А», и список работающих в них сотрудников.
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список отделов с максимальной зарплатой сотрудников в каждом отделе, отсортированный по максимальной зарплате.
3. «Отдел» и «Сотрудник» связаны соотношением многие-ко-многим. Выведите список всех связанных сотрудников и отделов, отсортированный по отделам, сортировка по сотрудникам произвольная.

```

1
2 from operator import itemgetter
3 class Spare_part:
4     """Деталь"""
5     def __init__(self, id, title, cost, prov_id):
6         self.id = id
7         self.title = title
8         self.cost = cost
9         self.prov_id = prov_id
10
11 class Provider:
12     """Поставщики"""
13     def __init__(self, id, name):
14         self.id = id
15         self.name = name
16
17 class SparProv:
18     """
19     Детали от поставщика
20     """
21     def __init__(self, prov_id, spare_id):
22         self.prov_id = prov_id
23         self.spare_id = spare_id
24
25
26 # Поставщики
27 prov = [
28     Provider(1, 'Ремонт'),
29     Provider(2, 'Гранд'),
30     Provider(3, 'Автомат'),
31     Provider(11, 'Машины'),
32     Provider(22, 'Алтай'),
33     Provider(33, 'Всё для ремонта'),
34 ]
35
36 # Детали
37 spar = [
38     Spare_part(1, 'Дрель', 7000, 1),
39     Spare_part(2, 'Молоток', 1700, 2),
40     Spare_part(3, 'Лобзик', 7500, 3),
41     Spare_part(4, 'Цемент 500 1 кг', 350, 22),
42     Spare_part(5, 'Цемент 500 10кг', 2700, 3),
43 ]
44
45 spar_prov = []
46     SparProv(1,1),
47     SparProv(2,2),
48     SparProv(3,3),
49     SparProv(3,4),
50     SparProv(3,5),
51     SparProv(11,1),
52     SparProv(22,2),
53     SparProv(33,3),
54     SparProv(33,4),
55     SparProv(33,5),
56 ]

```

```

one_to_many = [(e.title, e.cost, d.name)
               for d in prov
               for e in spar
               if e.prov_id==d.id]

many_to_many_temp = [(d.name, ed.prov_id, ed.spare_id)
                     for d in prov
                     for ed in spar_prov
                     if d.id==ed.prov_id]

many_to_many = [(e.title, e.cost, prov_name)
                for prov_name, prov_id, spare_id in many_to_many_temp
                for e in spar if e.id==prov_id]

print("A1")
first_result=[(i[2], i[0]) for i in one_to_many if i[2][:1] == 'A']
print(first_result)

print('\nA2')
second_result=[]
for d in prov:
    d_spar = list(filter(lambda i: i[2]==d.name, one_to_many))
    if len(d_spar) > 0:
        d_cost = [cost for _,cost,_ in d_spar]
        d_cost_max = max(d_cost)
        second_result.append((d.name, d_cost_max))
print(sorted(second_result, key=itemgetter(1), reverse=True))

print('\nA3')
third_result = sorted(many_to_many, key=itemgetter(2))
print(third_result)

```

```

A1
[('Автомат', 'Лобзик'), ('Автомат', 'Цемент 500 10кг'), ('Алтай', 'Цемент 500 1 кг')]

A2
[('Автомат', 7500), ('Ремонт', 7000), ('Гранд', 1700), ('Алтай', 350)]

A3
[('Лобзик', 7500, 'Автомат'), ('Лобзик', 7500, 'Автомат'), ('Лобзик', 7500, 'Автомат'), ('Молоток', 1700, 'Гранд'), ('Дрель', 7000, 'Ремонт')]
PS C:\Users\CROK>

```