

## Условие

Вариант запросов - Г. Вариант ПО - 20.

1. «Поставщик» и «Деталь» связаны соотношением один-ко-многим. Выведите список всех поставщиков, у которых название начинается с буквы «А», и список деталей, которые они поставляют.
2. «Поставщик» и «Деталь» связаны соотношением один-ко-многим. Выведите список поставщиков с максимальной стоимостью деталей для каждого поставщика, отсортированный по максимальной стоимости.
3. «Поставщик» и «Деталь» связаны соотношением многие-ко-многим. Выведите список всех связанных деталей и поставщиков, отсортированный по поставщикам, сортировка по деталям произвольная.

## Исходный код программы

```
1  #!/usr/bin/env python
2
3  from operator import itemgetter
4
5  class Detail:
6      """Деталь"""
7      def __init__(self, id, name, price, supplier_id):
8          self.id = id
9          self.name = name
10         self.price = price
11         self.supplier_id = supplier_id
12
13  class Supplier:
14      """Поставщик"""
15      def __init__(self, id, name):
16          self.id = id
17          self.name = name
18
19  class DetailSupplier:
20      """Деталь к поставщику, для реализации многие-ко-многим"""
21      def __init__(self, detail_id, supplier_id):
22          self.detail_id = detail_id
23          self.supplier_id = supplier_id
24
25
26  if __name__ == "__main__":
27      suppliers = [
28          Supplier(0, "ООО ПОСТАВЩИК"),
29          Supplier(1, "ААО МЕТАЛЛСТРОЙ"),
30          Supplier(2, "МедСклад"),
31          Supplier(3, "Акладстрой"),
32          Supplier(4, "Труба завод"),
33          Supplier(5, "РусБытХим"),
34      ]
35
36      details = [
37          Detail(0, "Труба пластиковая", 150, 0),
38          Detail(1, "Труба железная", 1500, 1),
39          Detail(2, "Труба стекловолоконная", 280, 2),
40          Detail(3, "Труба титановая", 164, 3),
41          Detail(4, "Ключ", 2080, 3),
42          Detail(5, "Часть", 1047, 3),
```

```

43         Detail(6, "Круг", 1100, 3),
44         Detail(7, "Стеклянный кусок", 120, 2),
45         Detail(8, "Углеволокно", 130, 2),
46         Detail(9, "Кран", 260, 1),
47         Detail(10, "Кусочек лего", 320, 4),
48         Detail(11, "Доска", 480, 5),
49         Detail(12, "Металлическая пластина", 720, 5),
50         Detail(13, "Пластиковая пластина", 640, 4),
51         Detail(14, "Минеральное стекло", 770, 3),
52     ]
53
54     details_suppliers = [
55         DetailSupplier(0, 0),
56         DetailSupplier(1, 1),
57         DetailSupplier(2, 2),
58         DetailSupplier(3, 3),
59         DetailSupplier(3, 4),
60         DetailSupplier(3, 5),
61         DetailSupplier(5, 5),
62         DetailSupplier(4, 4),
63         DetailSupplier(8, 1),
64         DetailSupplier(11, 5),
65     ]
66
67
68     # Соединение данных один-ко-многим
69     one_to_many = [(d.name, d.price, s.name)
70         for s in suppliers
71         for d in details
72         if d.supplier_id==s.id]
73
74     # Соединение данных многие-ко-многим
75     many_to_many_temp = [(s.name, ds.supplier_id, ds.detail_id)
76         for s in suppliers
77         for ds in details_suppliers
78         if s.id==ds.supplier_id]
79     many_to_many = [(d.name, d.price, supplier_name)
80         for supplier_name, supplier_id, detail_id in many_to_many_temp
81         for d in details if d.id==detail_id]
82
83     print("Задание A1")
84     res_11 = {}
85     selected_suppliers = [one_suppl[2] for one_suppl in one_to_many if one_suppl[2].startswith('a')]
86     for supplier_name in selected_suppliers:
87         details_for_suppl = [(one_detail[0], one_detail[1]) for one_detail in one_to_many if one
88             res_11.update({supplier_name: details_for_suppl})
89     print(res_11)
90     print()
91
92     print("Задание A2")
93     res_12_unsorted = []
94     for s in suppliers:
95         s_details = list(filter(lambda i: i[2]==s.name, one_to_many))
96         if len(s_details) > 0:
97             s_prices = [price for _, price, _ in s_details]
98             s_price_max = max(s_prices)
99             res_12_unsorted.append((s.name, s_price_max))
100
101     res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)
102     print(res_12)

```

```

103     print()
104
105     print("Задание A3")
106     res_13 = {}
107     suppliers.sort(key=lambda one_supplier: one_supplier.name)
108     for s in suppliers:
109         s_details = list(filter(lambda i: i[2]==s.name, many_to_many))
110         s_details_names = [x for x,_,_ in s_details]
111         res_13[s.name] = s_details_names
112     print(res_13)

```

## Скриншот работы программы

```

[tujh@tujhNotebook rk1]$ ./rk1.py
Задание A1
{'ААО МЕТАЛЛСТРОЙ': [('Труба железная', 1500), ('Кран', 260)], 'Акладстрой': [('Труба титановая', 164), ('Ключ', 2080), ('Часть', 1047), ('Круг', 1100), ('Минеральное стекло', 770)]}

Задание A2
[('Акладстрой', 2080), ('ААО МЕТАЛЛСТРОЙ', 1500), ('РусБытХим', 720), ('Трубзавод', 640), ('МедСклад', 280), ('ООО ПОСТАВЩИК', 150)]

Задание A3
{'ААО МЕТАЛЛСТРОЙ': ['Труба железная', 'Углеволокно'], 'Акладстрой': ['Труба титановая'], 'МедСклад': ['Труба стекловолоконная'], 'ООО ПОСТАВЩИК': ['Труба пластиковая'], 'РусБытХим': ['Труба титановая', 'Часть', 'Доска'], 'Трубзавод': ['Труба титановая', 'Ключ']}
[tujh@tujhNotebook rk1]$ 

```