

МГТУ им. Н.Э. Баумана

Отчёт по РК №1  
по курсу «Парадигмы и конструкции языков программирования»  
Вариант 19.

Руководитель  
Гапанюк Ю.Е.

Студент группы ИУ5-34Б  
Файзуллин К.Х.  
Подпись:

Задание:

**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.

19	Деталь	Производитель
----	--------	---------------

Текст программы:

```
from operator import itemgetter
```

```
class Detail:
```

```
    """Деталь"""
```

```
    def __init__(self, id, name, price, man_id):
```

```
        self.id = id
```

```
        self.name = name
```

```
        self.price = price
```

```
        self.man_id = man_id
```

```
class Manufacturer:
```

```
    """Производитель"""
```

```
    def __init__(self, id, title):
```

```
        self.id = id
```

```
        self.title = title
```

```
class DetMan:
```

```
    def __init__(self, man_id, det_id):
```

```
        self.man_id = man_id
```

```
        self.det_id = det_id
```

```
manufacturers = [
```

```
    Manufacturer(1, 'Дом в уюте'),
```

```
Manufacturer(2, 'Строй в удовольствие'),  
Manufacturer(3, 'Домашний помощник'),  
Manufacturer(4, 'Леруа мерлен'),  
]
```

```
details = [  
    Detail(1, 'Подшипник', 600, 1),  
    Detail(2, 'Колесо', 800, 1),  
    Detail(3, 'Винт', 400, 2),  
    Detail(4, 'Болт', 450, 3),  
    Detail(5, 'Втулка', 700, 3),  
]
```

```
det_man = [  
    DetMan(1, 1),  
    DetMan(1, 4),  
    DetMan(2, 3),  
    DetMan(3, 2),  
    DetMan(3, 3),  
    DetMan(3, 4),  
    DetMan(4, 1),  
    DetMan(4, 3),  
    DetMan(5, 3),  
    DetMan(5, 2),  
]
```

```
def main():  
    o_to_m = [(d.name, d.price, m.title)  
               for d in details  
               for m in manufacturers  
               if d.man_id == m.id]  
  
    m_to_m = [(m.title, dm.man_id, dm.det_id)  
               for m in manufacturers  
               for dm in det_man  
               if m.id == dm.man_id]  
  
    m_to_m = [(d.name, d.price, man_title)  
               for man_title, man_id, det_id in m_to_m]
```

```

    for d in details if d.id == det_id]

print('Задание Г1')
titles = []
for i in range(0, len(o_to_m)):
    if o_to_m[i][2][0] == "Д":
        titles.append(o_to_m[i][2])
res_1 = {title: [(otm[0], otm[1]) for otm in o_to_m if otm[2] == title] for title in
titles}
print(res_1)


print("\nЗадание Г2")
mans = [m.title for m in manufacturers]
res_2 = sorted([(title, max([otm[1] for otm in o_to_m if otm[2] == title])) for
title in titles], key=itemgetter(1), reverse=True)
print(res_2)


print("\nЗадание Г3")
res_3 = sorted(m_to_m, key=itemgetter(2))
for i in range(0, len(res_3)):
    print(res_3[i])


if __name__ == '__main__':
    main()

```

### Результат выполнения:

```

Задание Г1
{'Дом в уютe': [('Подшипник', 600), ('Колесо', 800)], 'Домашний помощник': [('Болт', 450), ('Втулка', 700)]}

Задание Г2
[('Дом в уютe', 800), ('Дом в уютe', 800), ('Домашний помощник', 700), ('Домашний помощник', 700)]

Задание Г3
('Подшипник', 600, 'Дом в уютe')
('Болт', 450, 'Дом в уютe')
('Колесо', 800, 'Домашний помощник')
('Винт', 400, 'Домашний помощник')
('Болт', 450, 'Домашний помощник')
('Подшипник', 600, 'Леруа мерлен')
('Винт', 400, 'Леруа мерлен')
('Винт', 400, 'Строй в удовольствие')

Process finished with exit code 0

```