



**«Московский государственный технический университет  
имени Н.Э. Баумана»  
(МГТУ им. Н.Э. Баумана)**

---

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Системы обработки информации и управления (ИУ5)

**О т ч е т**

**по рубежному контролю №1**

**Дисциплина: Разработка Интернет-Приложений**

Студент гр. ИУ5-53Б

\_\_\_\_\_  
(Подпись, дата)

Перлин Л.В.

(И.О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

Гапанюк Ю.Е.

(И.О. Фамилия)

Москва, 2020

# Задание

## Вариант В

Вариант предметной области

18	Музыкальное произведение	Оркестр
----	-----------------------------	---------

1. «Оркестр» и «Музыкальное произведение» связаны соотношением один-ко-многим.

Вывод списка всех музыкальных произведений, у которых название начинается с буквы «Л», и названия Оркестров, которые его исполняют.

2. «Оркестр» и «Музыкальное произведение» связаны соотношением один-ко-многим.

Вывод списка оркестров с минимальной продолжительностью композиции, отсортированный по минимальной продолжительности композиции (в секундах).

3. «Оркестр» и «Музыкальное произведение» связаны соотношением многие-ко-многим.

Вывод списка всех связанных исполняемых композиций и оркестров, отсортированный по композициям, сортировка по композициям произвольная.

## Текст программы

```
from operator import itemgetter
```

```
class compositionorchestra:
```

```
    #Музыкальные произведения оркестров (многие-ко-многим)
```

```
    def __init__(self, orchestra_id, composition_id):
```

```
        self.orchestra_id = orchestra_id
```

```
        self.composition_id = composition_id
```

```
class composition:
```

```
    #Музыкальная композиция
```

```
    def __init__(self, id, title, autor, length, orchestra_id):
```

```
        self.id = id
```

```
        self.title = title
```

```
        self.autor = autor
```

```
        self.length = length
```

```
        self.orchestra_id = orchestra_id
```

```
class orchestra:
    #Оркестр
    def __init__(self, id, name):
        self.id = id
        self.name = name

# Оркестры
orchestras = [
    orchestra(1, 'Royal Concertgebouw Orchestra'),
    orchestra(2, 'Berlin Philharmonic Orchestra'),
    orchestra(3, 'Vienna Philharmonic Orchestra'),
]

# Композиции
compositions = [
    composition(1, 'К Элизе', 'Людвиг ван Бетховен', 170, 1),
    composition(2, 'Турецкое рондо', 'Вольфганг Амадей Моцарт', 258, 2),
    composition(3, 'Аве Мария', 'Франц Шуберт', 324, 3),
    composition(4, 'Утро', 'Эдвард Григ', 563, 1),
    composition(5, 'Лунный свет', 'Клод Дебюсси', 434, 2),
    composition(6, 'Лебедь', 'Камиль Сен-Санс', 453, 3),
]

# Оркестры - композиции
orchestra_composition = [
    compositionorchestra(1, 1),
    compositionorchestra(2, 1),
    compositionorchestra(3, 2),
    compositionorchestra(2, 4),
    compositionorchestra(1, 5),
    compositionorchestra(3, 6)
]

def main():
    """Основная функция"""

    #один ко многим
    one_to_many = [(b.title, b.length, s.name)
                    for s in orchestras
                    for b in compositions
                    if b.orchestra_id == s.id]

    #многие ко многим
    many_to_many_temp = [(s.name, bs.orchestra_id, bs.composition_id)
                          for s in orchestras
                          for bs in orchestra_composition
                          if s.id == bs.orchestra_id]
```

```

many_to_many = [(b.title, orchestra_name)
                 for orchestra_name, orchestra_id, composition_id in many_to_many_temp
                 for b in compositions if b.id == composition_id]

print('Задание A1')
res_11 = list(filter(lambda x: x[0].startswith('Л'), one_to_many))
print(res_11)

print('\nЗадание A2')
res_12_unsorted = []

for s in orchestras:
    s_compositions = list(filter(lambda i: i[2] == s.name, one_to_many))
    if len(s_compositions) > 0:
        s_length = [length for _, length, _ in s_compositions]
        s_length_min = min(s_length)
        res_12_unsorted.append((s.name, s_length_min))
res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=False)
print(res_12)

print('\nЗадание A3')
res_13 = sorted(many_to_many, key=itemgetter(0))
print(res_13)

if __name__ == '__main__':
    main()

```

## Результат работы программы

```

Задание A1
[('Лунный свет', 434, 'Berlin Philharmonic Orchestra'), ('Лебедь', 453, 'Vienna Philharmonic Orchestra')]

Задание A2
[('Royal Concertgebouw Orchestra', 170), ('Berlin Philharmonic Orchestra', 258), ('Vienna Philharmonic Orchestra', 324)]

Задание A3
[('К Элизе', 'Royal Concertgebouw Orchestra'), ('К Элизе', 'Berlin Philharmonic Orchestra'), ('Лебедь', 'Vienna Philharmonic Orchestra'), ('Лунный свет', 'Royal Concertgebouw Orchestra'), ('Турецкое рондо', 'Vienna Philharmonic Orchestra'), ('Утро', 'Berlin Philharmonic Orchestra')]

```