

**Московский государственный технический
университет им. Н.Э. Баумана**

**Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»**

Курс «Разработка интернет приложений»

Рубежный контроль №1

Вариант №18Б

Выполнил:
студент группы ИУ5-52Б
Перова Анна

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю.Е.

Москва, 2021 г.

Описание задания:

- 1) Необходимо создать два класса данных в соответствии с Вашим вариантом предметной области, которые связаны отношениями один-ко-многим и многие-ко-многим.
- 2) Необходимо создать списки объектов классов, содержащих тестовые данные (3-5 записей), таким образом, чтобы первичные и вторичные ключи соответствующих записей были связаны по идентификаторам.
- 3) Необходимо разработать запросы в соответствии с Вашим вариантом. Запросы сформулированы в терминах классов «Сотрудник» и «Отдел», которые используются в примере. Вам нужно перенести эти требования в Ваш вариант предметной области. При разработке запросов необходимо по возможности использовать функциональные возможности языка Python (list/dict comprehensions, функции высших порядков).

Вариант 18Б:

1. «Оркестр» и «Музыкальное произведение» связаны соотношением многие-ко-многим. Выведите список всех связанных оркестров и их музыкальных произведений, отсортированный по композициям, сортировка по оркестрам произвольная.
2. «Оркестр» и «Музыкальное произведение» связаны соотношением многие-ко-многим. Выведите список оркестров с количеством музыкальных произведений, играемых каждым из них, отсортированный по количеству композиций.
3. «Дирижер» и «Оркестр» связаны соотношением один-ко-многим. Выведите список всех дирижеров, у которых ФИО заканчивается на «вич», и названия их оркестров.

Текст программы:

```
class Orchestra:
```

```
    def __init__(self, id, type, name):
```

```
        self.id = id
```

```
        self.type = type
```

```
        self.name = name
```

```
class Melody:
```

```
    def __init__(self, id, name, id_orch):
```

```
        self.id = id
```

```
        self.name = name
```

```

        self.id_orch = id_orch
class Conductor:
    def __init__(self, id, fio):
        self.id = id
        self.fio = fio
class OrchCond:
    def __init__(self, id, id_orchestra, id_conductor):
        self.id = id
        self.id_orchestra = id_orchestra
        self.id_conductor = id_conductor
def main():
    orch = [
        Orchestra(0, 'Brass Band', 'Glenn Miller'),
        Orchestra(1, 'Symphony Orchestra', 'Paul Mauriat-Foccata'),
        Orchestra(2, 'Folk Orchestra', 'La Peregrinacion'),
        Orchestra(3, 'Jazz Orchestra', 'Nicolas Rieu'),
        Orchestra(4, 'Chamber Orchestra', 'Lalalend'),
    ]
    melody = [
        Melody(0, 'Melody 1', 1),
        Melody(1, 'Melody 2', 0),
        Melody(2, 'Melody 3', 3),
        Melody(3, 'Melody 4', 4),
        Melody(4, 'Melody 5', 4),
    ]

    conductors = [Conductor(0, 'Перова Анна Евгеньевна'),
                  Conductor(1, 'Бабин Артём Сергеевич'),
                  Conductor(2, 'Барышников Михаил Игоревич')]

    orch_con = [
        OrchCond(0, 0, 0),
        OrchCond(1, 1, 1),
        OrchCond(2, 2, 2),
        OrchCond(3, 3, 3),
        OrchCond(4, 4, 4),]

```

```

    res1 = sorted([(mel.name, orc.type, orc.name) for mel in melody for orc in orch if
mel.id_orch == orc.id], key=lambda x: x[0])
    res2 = sorted({orc.name: len(list(filter(lambda x: x.id_orch == orc.id, melody)))
for orc in orch}.items(), key=lambda x: x[0], reverse=True)
    res3 = {cond.fio: [orc.name for orc in orch if orc.id in [orc_cond.id_orchestra for
orc_cond in orch_con if orc_cond.id_conductor == cond.id]] for cond in conductors
if str(cond.fio).endswith('вич')}
    print(res1)
    print()
    print(res2)
    print()
    print(res3)
if __name__ == '__main__':
    main()

```

Экранная форма с результатом выполнения программы:

```

[('Melody 1', 'Symphony Orchestra', 'Paul Mauriat-Foccata'), ('Melody 2', 'Brass Band', 'Glenn Miller'), ('Melody 3', 'Jazz Orchestra', 'Nicolas Rieu'), ('Melody 4
< Chamber Orchestra', 'Lalalend'), ('Melody 5', 'Chamber Orchestra', 'Lalalend')]

[('Paul Mauriat-Foccata', 1), ('Nicolas Rieu', 1), ('Lalalend', 2), ('La Peregrinacion', 0), ('Glenn Miller', 1)]

{'Бабин Артём Сергеевич': ['Paul Mauriat-Foccata'], 'Барышников Михаил Игоревич': ['La Peregrinacion']}

```