Студент: Нестеренко Иван

Группа: ИУ5-32Б

Рубежный контроль №1 по дисциплине «Базовые компоненты интернет-технологий»

```
Вариант №17Б
```

Текст программы:

Файл Conductors.py

```
class conductor():
    #дирижер
    def __init__(self,id_value,name_value, rating_value,
id_orchestra_value):
    self.id = id_value
    self.name = name_value
    self.rating = rating_value
    self.id orchestra = id orchestra value
```

Файл Orchestras.py

```
class orchestra():
    #opkectpa
    def __init__(self, id_value, number_value):
        self.id = id_value
        self.number = number value
```

Файл CONORCH.py

```
class ConOrch():
    #дирижеры оркестра, дял реализации связи многие-ко-многим
    def __init__(self, id_con_val, id_orch_val):
        self.id_con = id_con_val
        self.id orch = id orch val
```

Файл таіп.ру

```
from Conductors import conductor
from Orchestras import orchestra
from CONORCH import ConOrch
from operator import itemgetter
#Вариант 17Б
#Оркестры
orchestras = [
```

```
orchestra(1, 'Opkectp №1'),
    orchestra(2, 'OpkecTp №2'),
    orchestra(3,'OpkecTp №3'),
    orchestra(11, 'He-OpkecTp №1'),
    orchestra(22, 'He-OpkecTp №2'),
    orchestra(33,'he-Opkec⊤p №3')
]
#Дирижеры
conductors = [
    conductor (1, 'Мурорский', 4.7,3),
    conductor (2, 'Глинка', 5.0, 3),
    conductor (3, 'Верди', 3.9, 2),
    conductor (4, 'Стравинский', 4.9,2),
    conductor (5, 'Крылов', 4.6, 2),
    conductor (6, 'Гончаров', 4.1, 1)
]
con orch = [
    ConOrch(1,1),
    ConOrch(2,2),
    ConOrch(3,2),
    ConOrch(4,3),
    ConOrch(5,3),
    ConOrch(6,3),
    ConOrch(1,33),
    ConOrch(2,22),
    ConOrch(3,33),
    ConOrch(4,11),
    ConOrch(5,11),
    ConOrch (6, 22)
]
def main():
    #соединение данных один-ко-многим
    one to many = [(c.name, c.rating, orc.number)
        for orc in orchestras
        for c in conductors
        if c.id orchestra == orc.id
    ]
    #соединение данных многие-ко-многим
    many to many temp = [(orc.id, orc.number, corc.id con)
        for orc in orchestras
        for corc in con orch
        if orc.id == corc.id orch
```

```
]
    many to many = [(c.name, c.rating, orch num)
         for c in conductors
         for orch id, orch num, c id in many to many temp
         if c.id == c id
    1
    #Task1
    res 1 = sorted(one to many, key = itemgetter(0))
    print('Задание №1:')
    print(res 1)
    #Task2
    res 2 = []
    for i in range(3):
         count con = list(filter(lambda j: j[2] ==
orchestras[i].number, one to many))
         res 2.append((orchestras[i].number, len(count con)))
    res 2 = sorted(res 2, key = itemgetter(1), reverse = True)
    print('Задание №2:')
    print(res 2)
    #Task3
    res 3 = []
    for i in many to many:
         if (i[0][-2:] == 'oB'):
              res 3.append(i)
    print('Задание №3:')
    print(res 3)
if name == " main ":
    main()
    Результаты работы:
Задание №1:
[('Верди', 3.9, 'Оркестр №2'), ('Глинка', 5.0, 'Оркестр №3'), ('Гончаров', 4.1, 'Оркестр
№1'), ('Крылов', 4.6, 'Оркестр №2'), ('Мурорский', 4.7, 'Оркестр №3'), ('Стравинский',
4.9, 'Opkectp №2')]
Задание №2:
[('Opkectp No2', 3), ('Opkectp No3', 2), ('Opkectp No1', 1)]
Задание №3:
[('Крылов', 4.6, 'Оркестр №3'), ('Крылов', 4.6, 'не-Оркестр №1'), ('Гончаров', 4.1,
'Оркестр №3'), ('Гончаров', 4.1, 'не-Оркестр №2')]
```