



Министерство образования и науки Российской
Федерации
Федеральное государственное бюджетное
образовательное учреждение высшего
образования
«Московский государственный технический
университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Отчёт по рубежному контролю №1 по курсу
«Разработка интернет-приложений»
Вариант № 5

Выполнил: Горбик Иван, РТ5-51Б

Проверил: _____

9 декабря 2020 г.

Москва, 2020г.

Задание:

Создать классы «Музыкант» и «Оркестр», а также класс для связи М-М между ними.

Запросы:

1. Классы «Музыкант» и «Оркестр» имеют связь 1-М. Вывести названия всех оркестров, в которых есть музыканты, играющие на виолончелях.
2. Классы «Музыкант» и «Оркестр» имеют связь 1-М. Вывести средний возраст музыканта в каждом оркестре
3. Классы «Музыкант» и «Оркестр» имеют связь М-М. Вывести названия оркестров, возраст всех музыкантов которых колеблется от 30 до 40 лет.

Листинг программы

```
# создадим класс "Музыкант" с полями:
# ID музыканта, ФИО, пол, возраст, инструмент, стаж, ID оркестра
class Musician:
    def __init__(self, id, fio, age, instrument, orch_id):
        self.id = id
        self.fio = fio
        self.age = age
        self.instrument = instrument
        self.orch_id = orch_id

# создадим класс "Оркестр" с полями: ID оркестра, название, год основания и
число музыкантов
class Orchestra:
    def __init__(self, id, name, year, staff):
        self.id = id
        self.name = name
        self.year = year
        self.staff = staff

# создадим класс для реализации связи М-М
class MusOrch:
    def __init__(self, mus_id, orch_id):
        self.mus_id = mus_id
        self.orch_id = orch_id

# данные по музыкантам
musicians = [
    Musician(1, "Загорецкий Михаил Павлович", 30, "фортепиано", 2),
    Musician(2, "Мусечкина Зоя Петровна", 28, "виолончель", 4),
    Musician(3, "Романов Григорий Ильич", 41, "туба", 1),
    Musician(4, "Жданова Людмила Константиновна", 37, "флейта", 3),
    Musician(5, "Богородский Даниил Владимирович", 32, "литавры", 1),
    Musician(6, "Цаплин Сергей Семёнович", 49, "большой барабан", 2),
    Musician(7, "Трубецкая Софья Андреевна", 29, "скрипка", 1),
```

```

    Musician(8, "Юрченко Константин Петрович", 42, "альт", 3),
    Musician(9, "Балайкин Алексей Николаевич", 47, "валторн", 4),
    Musician(10, "Харитоньева Светлана Юрьевна", 31, "кларнет", 2),
    Musician(11, "Соловецкая Екатерина Михайловна", 43, "виолончель", 2),
]

# данные по оркестрам
orchestras = [
    Orchestra(1, "Большой духовой оркестр", 2011, 70),
    Orchestra(2, "Большой симфонический оркестр", 2005, 120),
    Orchestra(3, "Оркестр Большого театра", 1912, 150),
    Orchestra(4, "Военный оркестр", 1999, 50),
]

mus_orchs = [
    MusOrch(1, 2),
    MusOrch(3, 1),
    MusOrch(2, 4),
    MusOrch(4, 3),
    MusOrch(5, 1),
    MusOrch(6, 2),
    MusOrch(7, 1),
    MusOrch(8, 3),
    MusOrch(9, 4),
    MusOrch(10, 2),
    MusOrch(11, 2),
]

one_to_many_list = list((mus, orch)
    for mus in musicians
    for orch in orchestras
    if (mus.orch_id == orch.id))

many_to_many_list = list((orch.name, mus.id)
    for mus in musicians
    for orch in orchestras
    for bk, sh in list((item.mus_id, item.orch_id) for
item in mus_orchs)
    if mus.id == sh and orch.id == bk)

# Задание 1: Вывести названия всех оркестров,
# в которых есть музыканты, играющие на виолончелях
print("Задание 1")
for i in one_to_many_list:
    if "виолончель" in i[0].instrument:
        print(i[1].name)

# Задание 2: Вывести средний возраст музыкантов в каждом оркестре
print("\nЗадание 2")
ageAvgList = []
for o in orchestras:
    musList = list(filter(lambda x: x[0].orch_id == o.id, one_to_many_list))
    ageAvg = 0
    for item in musList:
        musician = item[0]
        ageAvg += musician.age
    ageAvg = round(ageAvg/len(musList), 2)
    ageAvgList.append((o.name, ageAvg))
for item in sorted(ageAvgList, key=lambda x: x[0]):
    print("Средний возраст музыканта в оркестре", item[0], "составляет",
item[1])

```

```
# Задание 3: Вывести названия всех оркестров,  
# все музыканты в которых не моложе 30 лет и не старше 40  
print("\nЗадание 3")  
for musician in musicians:  
    if musician.age > 40 or musician.age < 30:  
        continue  
    orchList = list(filter(lambda x: musician.id == x[1], many_to_many_list))  
    for item in orchList:  
        print(item[0])
```

Скрин с примером выполнения:

Задание 1

Военный оркестр

Большой симфонический оркестр

Задание 2

Средний возраст музыканта в оркестре Большой духовой оркестр составляет 34.0

Средний возраст музыканта в оркестре Большой симфонический оркестр составляет 38.25

Средний возраст музыканта в оркестре Военный оркестр составляет 37.5

Средний возраст музыканта в оркестре Оркестр Большого театра составляет 39.5

Задание 3

Оркестр Большого театра

Большой симфонический оркестр