

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

Факультет «Информатика и управление»
Кафедра ИУ5. Курс «Вычислительные средства АСОИУ»

Курс «Разработка интернет-приложений»
Отчет по Рубежному контролю № 1

Выполнил:
студент группы ИУ5-51Б
Андреев А. В.
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю.Е.
Подпись и дата:

Москва, 2020 г.

Вариант 3 А

1. «Автопарк» и «Водитель» связаны соотношением один-ко-многим. Выведите список всех связанных Водителей и Автопарков, отсортированный по Автопаркам, сортировка по Водителям произвольная.
2. «Автопарк» и «Водитель» связаны соотношением один-ко-многим. Выведите список Автопарков с суммарной зарплатой Водителей в каждом Автопарке, отсортированный по суммарной зарплате.
3. «Автопарк» и «Водитель» связаны соотношением многие-ко-многим. Выведите список всех Автопарков, у которых в названии присутствует слово «Автопарк», и список работающих в них Водителей.

Текст программы

```
from operator import itemgetter

class Driver:
    """Водитель"""

    def __init__(self, id, fio, sal, Avtopark_id):
        self.id = id
        self.fio = fio
        self.sal = sal
        self.Avtopark_id = Avtopark_id

class Avtopark:
    """Автопарк"""

    def __init__(self, id, name):
        self.id = id
        self.name = name

class DriverAvtopark:
    """СВЯЗИ МНОГИЕ-КО-МНОГИМ"""

    def __init__(self, Avtopark_id, Driver_id):
        self.Avtopark_id = Avtopark_id
        self.Driver_id = Driver_id

# Автопарк
Avtoparks = [
    Avtopark(1, 'Автопарк Лада'),
    Avtopark(2, 'Автопарк Ford'),
    Avtopark(3, 'Автопарк BMW'),
]

# Водитель
Drivers = [
    Driver(1, 'Андреев', 60000, 3),
    Driver(2, 'Пахомкин', 35000, 1),
    Driver(3, 'Толпаров', 45000, 2),
    Driver(4, 'Шевчук', 35000, 1),
]
```

```

        Driver(5, 'Рогозин', 45000, 2),
    ]

Drivers_Avtoparks = [
    DriverAvtopark(1, 2),
    DriverAvtopark(2, 3),
    DriverAvtopark(1, 4),
    DriverAvtopark(2, 5),
    DriverAvtopark(3, 1),
]

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

    # Соединение данных один-ко-многим
    one_to_many = [(e.fio, e.sal, d.name)
                    for d in Avtoparks
                    for e in Drivers
                    if e.Avtopark_id == d.id]

    # Соединение данных многие-ко-многим
    many_to_many_tDriver = [(d.name, ed.Avtopark_id, ed.Driver_id)
                             for d in Avtoparks
                             for ed in Drivers_Avtoparks
                             if d.id == ed.Avtopark_id]

    many_to_many = [(e.fio, e.sal, Avtopark_name)
                    for Avtopark_name, Avtopark_id, Driver_id in
many_to_many_tDriver
                    for e in Drivers if e.id == Driver_id]

    print('Задание A1')
    res_11 = sorted(one_to_many, key=itemgetter(2))
    print(res_11)

    print('\nЗадание A2')
    res_12_unsorted = []
    for d in Avtoparks:
        d_Drivers = list(filter(lambda i: i[2] == d.name, one_to_many))
        if len(d_Drivers) > 0:
            d_sals = [sal for _, sal, _ in d_Drivers]
            d_sals_sum = sum(d_sals)
            res_12_unsorted.append((d.name, d_sals_sum))

    res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)
    print(res_12)

    print('\nЗадание A3')
    res_13 = {}
    for d in Avtoparks:
        if 'Автопарк' in d.name:
            d_Drivers = list(filter(lambda i: i[2] == d.name, many_to_many))
            d_Drivers_names = [x for x, _, _ in d_Drivers]
            res_13[d.name] = d_Drivers_names

    print(res_13)

if __name__ == '__main__':
    main()

```

Результат работы программы

Задание A1

[('Андреев', 60000, 'Автопарк BMW'), ('Толпаров', 45000, 'Автопарк Ford'), ('Рогозин', 45000, 'Автопарк Ford'), ('Пахомкин', 35000, 'Автопарк Лада'), ('Шевчук', 35000, 'Автопарк Лада')]

Задание A2

[('Автопарк Ford', 90000), ('Автопарк Лада', 70000), ('Автопарк BMW', 60000)]

Задание A3

{'Автопарк Лада': ['Пахомкин', 'Шевчук'], 'Автопарк Ford': ['Толпаров', 'Рогозин'], 'Автопарк BMW': ['Андреев']}