



**Министерство науки и высшего образования 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)**

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

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

Выполнил:
студент группы ИУ5-33Б
Ефременко Д.С.

Проверил:
Гапанюк Ю.Е.

2021 г.

Полученное задание:

Вариант В.

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список всех сотрудников, у которых фамилия начинается с буквы «А», и названия их отделов.
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список отделов с минимальной зарплатой сотрудников в каждом отделе, отсортированный по минимальной зарплате.
3. «Отдел» и «Сотрудник» связаны соотношением многие-ко-многим. Выведите список всех связанных сотрудников и отделов, отсортированный по сотрудникам, сортировка по отделам произвольная.

3	Водитель	Автопарк
---	----------	----------

Текст программы:

```
# используется для сортировки
from operator import itemgetter

class Driver:
    """Водитель"""

    def __init__(self, id, fio, sal, exp_id):
        self.id = id
        self.fio = fio
        self.sal = sal
        self.exp_id = exp_id #водительский стаж

class Auto:
    """Автопарк"""

    def __init__(self, id, mode):
        self.id = id
        self.mode = mode #виды автобусов

class AutoDriver:
    """
    'Водитель автопарка' для реализации
    связи многие-ко-многим
    """

    def __init__(self, driv_id, auto_id):
        self.driv_id = driv_id
        self.auto_id = auto_id
```

```

# Автопарк
auto = [
    Auto(1, 'туристический'),
    Auto(2, 'городской'),
    Auto(3, 'школьный'),
    Auto(4, 'экскурсионный'),
    Auto(5, 'перонный'),
    Auto(6, 'пригородный'),
]

# Водители
driv = [
    Driver(1, 'Ефременко', 50000, 5),
    Driver(2, 'Семенов', 30000, 3),
    Driver(3, 'Стебунов', 45000, 4),
    Driver(4, 'Носкин', 44000, 6),
    Driver(5, 'Бегларов', 45000, 2),
    Driver(6, 'Алешин', 23000, 4),
    Driver(7, 'Ахтамбаев', 40000, 2),
    Driver(8, 'Андреев', 45000, 3)
]

auto_drivers = [
    AutoDriver(1, 1),
    AutoDriver(3, 2),
    AutoDriver(3, 4),
    AutoDriver(5, 1),
    AutoDriver(6, 3),
    AutoDriver(2, 1),
    AutoDriver(4, 2),
    AutoDriver(3, 3),
    AutoDriver(4, 5),
    AutoDriver(6, 2),
]

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

    # Соединение данных один-ко-многим
    one_to_many = [(d.fio, d.sal, a.mode)
                    for a in auto
                    for d in driv
                    if d.exp_id == a.id]

    # Соединение данных многие-ко-многим
    many_to_many_temp = [(a.mode, ad.auto_id, ad.driv_id)
                           for a in auto
                           for ad in auto_drivers
                           if a.id == ad.auto_id]

    many_to_many = [(d.fio, d.sal, auto_name)
                     for auto_name, auto_id, driv_id in many_to_many_temp
                     for d in driv if d.id == driv_id]

    print('Test') #Вывод списков со связями 1-м, м-м
    res_0 = (one_to_many)
    print(res_0)
    res_01 = (many_to_many)
    print(res_01)

    print('Задание B1')
    res_11 = []

```

```

for fio, sal, auto_name in one_to_many:
    if 'А' in fio[0]:
        res_11.append((fio, auto_name))
print(res_11)

print('Задание В2')
buff = []
for a in auto:
    #список видов транспорта
    a_modes = list(filter(lambda i: i[2] == a.mode, one_to_many))
    if len(a_modes) > 0:
        a_sal = [sal for , sal, in a_modes]
        min_sal = min(a_sal)
        buff.append((a.mode, min_sal))
res_12 = sorted(buff, key=itemgetter(1))
print(res_12)

print('Задание В3')
buff = []
for fio, sal, auto_name in many_to_many:
    buff.append((fio, auto_name))
res_13 = list(sorted(buff, key=itemgetter(0)))
print(res_13)

if __name__ == '__main__':
    main()

```

Результаты выполнения программы:

```

Задание В1
[('Ахтамбаев', 'городской'), ('Андреев', 'школьный'), ('Алешин', 'экскурсионный')]

```

```

Задание В2
[('экскурсионный', 23000), ('школьный', 30000), ('городской', 40000), ('пригородный', 44000), ('перонный', 50000)]

```

```

Задание В3
[('Алешин', 'городской'), ('Алешин', 'школьный'), ('Бегларов', 'туристический'), ('Ефременко', 'туристический'), ('Носкин', 'городской'),

```

```

('Носкин', 'перонный'), ('Семенов', 'туристический'), ('Стебунов', 'городской'), ('Стебунов', 'школьный'), ('Стебунов', 'экскурсионный')]

```