

Московский государственный технический  
университет им. Н.Э. Баумана

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

Курс «Разработка интернет приложений»  
Отчет по рубежному контролю №1

Выполнил :

студент группы ИУ5-53Б

Алёшин Александр

Вариант РК:

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.

Вариант запросов РК1	Вариант предметной области РК1
В	2

№ варианта	Класс 1	Класс 2
2	Школьник	Класс

Код программы:

```
from operator import itemgetter

class Pup:
    """Ученик"""

    def __init__(self, id, fio, age, cls_id):
        self.id = id
        self.fio = fio
        self.cls_id = cls_id
        self.age = age

class Cls:
    """Класс"""

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

class ClsPup:
```



```

sort = list(filter(lambda x: x[0].startswith('A'), one_to_many))
print('Задание B1')
res_11 = sorted(sort, key=itemgetter(1))
print(res_11)

print('\nЗадание B2')
res_12_unsorted = []
# Перебираем все отделы
for d in classes:
    # Список учеников класса
    d_pups = list(filter(lambda i: i[2] == d.name, one_to_many))
    # Если класс не пустой
    if len(d_pups) > 0:
        # Возраст учеников класса
        d_age = [age for _, age, _ in d_pups]
        # Минимальный возраст учеников класса
        d_age_min = min(d_age)
        res_12_unsorted.append((d.name, d_age_min))

# Сортировка по минимальному возрасту
res_12 = sorted(res_12_unsorted, key=itemgetter(1))
print(res_12)

print('\nЗадание B3')
res_13 = {}
# Перебираем все
print(sorted(many_to_many, key=itemgetter(0)))

if __name__ == '__main__':
    main()

```

## Результат работы программы :

```

Задание B1
[('Артамонов', 18, '11 A')]

Задание B2
[('10 A', 15), ('11 Б', 16), ('11 А', 17)]

Задание B3
[('Артамонов', '11 А'), ('Иваненко', '11 Б'), ('Иванин', '10 А'), ('Иванов', '11 Б'), ('Петров', '11 А'), ('Фёдоров', '10 А')]
{}

Process finished with exit code 0

```