



**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»**

**Курс «Базовые компоненты интернет-технологий»
Отчёт по рубежному контролю №1**

**Выполнил:
студент группы ИУ5-32Б
Поддубный М.Н.**

**Проверил:
Гапанюк Ю.Е.**

2021 г.

Описание задания

(Вариант предметной области - 5, вариант запросов - Б)

1. «Каталог файлов» и «Файл» связаны соотношением один-ко-многим. Выведите список всех связанных музыкантов и оркестров, отсортированный по файлам, сортировка по каталогам произвольная.
2. «Каталог файлов» и «Файл» связаны соотношением один-ко-многим. Выведите список каталогов с количеством файлов в каждом каталоге, отсортированный по количеству файлов.
3. «Каталог файлов» и «Файл» связаны соотношением многие-ко-многим. Выведите список всех музыкантов, у которых фамилия заканчивается на «ия», и названия их оркестров.

Текст программы

```
from operator import itemgetter
```

```
class Filedir:
```

```
    """Каталог файлов"""
```

```
    def __init__(self, id, name):
```

```
        self.id = id
```

```
        self.name = name
```

```
class File:
```

```
    """Файл"""
```

```
    def __init__(self, id, name, size, dir_id):
```

```
        self.id = id
```

```
        self.name = name
```

```
        self.size = size
```

```
        self.dir_id = dir_id
```

```
class FiledirFile:
```

```
    """
```

```
    'Файлы в каталоге' для реализации
```

СВЯЗИ МНОГИЕ-КО-МНОГИМ

"""

```
def __init__(self, Filedir_id, File_id):  
    self.File_dir_id = Filedir_id  
    self.File_id = File_id
```

```
File_dirs = [  
    Filedir(1, 'отдых'),  
    Filedir(2, 'учёба'),  
    Filedir(3, 'телефон'),  
  
    Filedir(11, 'отдых (другой)'),  
    Filedir(22, 'учёба (другая)'),  
    Filedir(33, 'телефон (другой)'),  
]
```

```
Files = [  
    File(1, 'Крым', 4036, 1),  
    File(2, 'Резервная копия', 349803, 3),  
    File(3, 'Турция', 12089, 1),  
    File(4, 'конспект', 4036, 2),  
    File(5, 'программы', 9836, 3),  
    File(6, 'Демидович учебник', 59836, 2),  
    File(7, 'Лаба3', 2638, 2),  
]
```

```
File_dir_File = [  
    FiledirFile(1, 1),  
    FiledirFile(1, 3),  
    FiledirFile(2, 4),  
    FiledirFile(3, 2),  
    FiledirFile(3, 5),  
  
    FiledirFile(11, 1),  
    FiledirFile(11, 3),  
    FiledirFile(22, 4),  
    FiledirFile(33, 2),  
    FiledirFile(33, 5),
```

]

```
def main():
```

```
    """Основная функция"""
```

```
    # Соединение данных один-ко-многим
```

```
    one_to_many = [(f.name, f.size, d.name)
```

```
                    for d in File_dirs
```

```
                    for f in Files
```

```
                    if f.dir_id == d.id]
```

```
    many_to_many_temp = [(d.name, fd.Filedir_id, fd.File_id)
```

```
                        for d in File_dirs
```

```
                        for fd in File_dir_File
```

```
                        if d.id == fd.Filedir_id]
```

```
    many_to_many = [(f.name, f.size, dir_name)
```

```
                    for dir_name, dir_id, File_id in many_to_many_temp
```

```
                    for f in Files if f.id == File_id]
```

```
    print('Задание A1')
```

```
    res_11 = sorted(one_to_many, key=lambda x: x[0].lower())
```

```
    [print (e) for e in res_11]
```

```
    print("\nЗадание A2')
```

```
    res_12_unsorted = []
```

```
    # Перебираем все отделы
```

```
    for d in File_dirs:
```

```
        # Список сотрудников отдела
```

```
        d_Files = list(filter(lambda i: i[2] == d.name, one_to_many))
```

```
        # Если отдел не пустой
```

```
        if len(d_Files) > 0:
```

```
            # Зарплаты сотрудников отдела
```

```
            res_12_unsorted.append((d.name, len(d_Files)))
```

```
    # Сортировка по суммарной зарплате
```

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

```
    [print (e) for e in res_12]
```

```
    print("\nЗадание A3')
```

```

res_13 = {}
# Перебираем все отделы
for f in Files:
    if f.name.endswith('ия'):
        # Список сотрудников отдела
        f_dirs = list(filter(lambda i: i[0] == f.name, many_to_many))
        # Только ФИО сотрудников
        f_dirs_names = [x for _, _, x in f_dirs]
        # Добавляем результат в словарь
        # ключ - отдел, значение - список фамилий
        res_13[f.name] = f_dirs_names

[print(k, v) for k, v in res_13.items()]

if __name__ == '__main__':
    main()

```

Результат выполнения программы

```

Задание А1
('Демидович учебник', 59836, 'учёба')
('конспект', 4036, 'учёба')
('Крым', 4036, 'отдых')
('Лаба3', 2638, 'учёба')
('программы', 9836, 'телефон')
('Резервная копия', 349803, 'телефон')
('Турция', 12089, 'отдых')

Задание А2
('учёба', 3)
('отдых', 2)
('телефон', 2)

Задание А3
Резервная копия ['телефон', 'телефон (другой)']
Турция ['отдых', 'отдых (другой)']

Process finished with exit code 0

```