



**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»
Вариант №15-В**

**Выполнил:
студент группы ИУ5-33Б
Рассказов Н.Д.**

**Проверил:
Преподаватель кафедры ИУ-5
Гапанюк Ю. Е.**

2022 г.

Выданное 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.

№ варианта	Класс 1	Класс 2
15	Файл	Каталог файлов

Сформированные запросы

1. «Каталог файлов» и «Файл» связаны соотношением один-ко-многим. Выведите список всех файлов, у которых название начинается с буквы «А», и пути их каталогов.
2. «Каталог файлов» и «Файл» связаны соотношением один-ко-многим. Выведите список каталогов файлов с минимальным размером файла в каждом каталоге, отсортированный по минимальному размеру файла
3. «Каталог файлов» и «Файл» связаны соотношением многие-ко-многим. Выведите список всех связанных файлов и каталогов файла, отсортированный по файлам, сортировка по каталогам файлов произвольная.

Текст программы

```
from operator import itemgetter

class File:
    """Файл"""
    def __init__(self, id, name, weight_mb, catalog_id):
        self.id = id
        self.name = name
        self.weight_mb = weight_mb
        self.catalog_id = catalog_id

class Catalog:
```

```

        """Каталог файлов"""
    def __init__(self, id, path):
        self.id = id
        self.path = path

class FileCatalog:
    """
        'Файл в каталоге файлов' для реализации
        связи многие-ко-многим
    """
    def __init__(self, catalog_id, file_id):
        self.catalog_id = catalog_id
        self.file_id = file_id

# Каталоги файлов
catalogs = [
    Catalog(1, 'C:\\'),
    Catalog(2, 'C:\\Users\\'),
    Catalog(3, 'C:\\Users\\User\\'),

    Catalog(11, 'D:\\'),
    Catalog(22, 'D:\\Users\\'),
    Catalog(33, 'D:\\Users\\User\\'),
]

# Файлы
files = [
    File(1, 'AllGuys.mp4', 14.9, 1),
    File(2, 'Valorant.mp4', 1699.84, 2),
    File(3, 'Atement.docx', 0.013, 3),
    File(4, 'Statement.pdf', 0.013, 3),
    File(5, 'Programm.zip', 100, 3),
]

# Файлы в каталогах файлов
files_catalogs = [
    FileCatalog(1,1),
    FileCatalog(2,2),
    FileCatalog(3,3),
    FileCatalog(3,4),

```

```

FileCatalog(3,5),

FileCatalog(11,1),
FileCatalog(22,2),
FileCatalog(33,3),
FileCatalog(33,4),
FileCatalog(33,5),
]

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

    # Соединение данных один-ко-многим
    one_to_many = [(f.name, f.weight_mb, c.path)
                    for c in catalogs
                    for f in files
                    if f.catalog_id==c.id]

    # Соединение данных многие-ко-многим
    many_to_many_temp = [(c.path, fc.catalog_id, fc.file_id)
                          for c in catalogs
                          for fc in files_catalogs
                          if c.id==fc.catalog_id]

    many_to_many = [(f.name, f.weight_mb, catalog_path)
                    for catalog_path, catalog_id, file_id in
many_to_many_temp
                    for f in files
                    if f.id==file_id]

    print('Задание B1')
    # res_11 = [for i in one_to_many if i[0][0]=="A"]
    res_11 = list(filter(lambda i: i[0][0]=="A", one_to_many))
    print(res_11)

    print('\nЗадание B2')
    res_12_unsorted = []
    # Перебираем все каталоги
    for c in catalogs:
        # Список файлов в каталоге

```

```

        c_files = list(filter(lambda i: i[2]==c.path,
one_to_many))
        # Если каталог не пустой
        if len(c_files) > 0:
            # Весы файлов в каталоге
            c_weights = [weight_mb for _,weight_mb,_ in
c_files]

            # Минимальный вес файла в каталоге
            c_weights_min = min(c_weights)
            res_12_unsorted.append((c.path, c_weights_min))

        # Сортировка по минимальному весу файла
        res_12 = sorted(res_12_unsorted, key=itemgetter(1),
reverse=True)
        print(res_12)

    print('\nЗадание B3')
    res_13 = {}
    # Перебираем все каталоги
    for c in catalogs:
        # Список файлов каталога
        c_files = list(filter(lambda i: i[2]==c.path,
many_to_many))
        # Только имена файлов
        c_files_names = [x for x,_,_ in c_files]
        # Добавляем результат в словарь
        res_13[c.path] = sorted(c_files_names)

    # Сортируем словарь по файлам
    res_13 = sorted(res_13.items(), key=itemgetter(1))
    print(res_13)

if __name__ == '__main__':
    main()

```

Результаты выполнения

Задание B1

[('AllGuys.mp4', 14.9, 'C:\\'), ('Atement.docx', 0.013, 'C:\\Users\\User\\')]

Задание B2

`[('C:\\Users\\', 1699.84), ('C:\\', 14.9), ('C:\\Users\\User\\', 0.013)]`

Задание B3

`[('C:\\', ['AllGuys.mp4']), ('D:\\', ['AllGuys.mp4']), ('C:\\Users\\User\\', ['Atement.docx', 'Programm.zip', 'Statement.pdf']), ('D:\\Users\\User\\', ['Atement.docx', 'Programm.zip', 'Statement.pdf']), ('C:\\Users\\', ['Valorant.mp4']), ('D:\\Users\\', ['Valorant.mp4'])]`

Скриншот с результатами выполнения

```
Задание B1
[('AllGuys.mp4', 14.9, 'C:\\'), ('Atement.docx', 0.013, 'C:\\Users\\User\\')]

Задание B2
[('C:\\Users\\', 1699.84), ('C:\\', 14.9), ('C:\\Users\\User\\', 0.013)]

Задание B3
[('C:\\', ['AllGuys.mp4']), ('D:\\', ['AllGuys.mp4']), ('C:\\Users\\User\\', ['Atement.docx', 'Programm.zip', 'Statement.pdf']), ('D:\\Users\\User\\', ['Atement.docx', 'Programm.zip', 'Statement.pdf']), ('C:\\Users\\', ['Valorant.mp4']), ('D:\\Users\\', ['Valorant.mp4'])]
```