

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

**Факультет «Информатика и управление»**

**Кафедра ИУ5. Курс «Программирование на основе классов и  
шаблонов»**

**Отчет по рубежному контролю №1**

Выполнил:

студентка группы ИУ5-31Б  
Покшубина С.С.

Подпись и дата: 21.10.2021

Проверил:

Гапанюк Ю.Е.

Подпись и дата:

г. Москва, 2021 г.

## Задание:

### Вариант 15А.

1. «Каталог файлов» и «Файл» связаны соотношением один-ко-многим. Выведите список всех связанных файлов и каталогов, отсортированный по каталогам, сортировка по файлам произвольная.
2. «Каталог файлов» и «Файл» связаны соотношением один-ко-многим. Выведите список каталогов с суммарным размером файлов в каждом каталоге, отсортированный по суммарному размеру.
3. «Каталог файлов» и «Файл» связаны соотношением многие-ко-многим. Выведите список всех каталогов, у которых в названии присутствует слово «Папка 2», и список файлов, находящихся в них.

## Текст программы:

```
# используется для сортировки
from operator import itemgetter

class File:
    """ Файл """
    def __init__(self, id, name, size, cat_id):
        self.id = id
        self.name = name
        self.size = size
        self.cat_id = cat_id

class Catalog:
    """Каталог файлов"""
    def __init__(self, id, cat_name):
        self.id = id
        self.cat_name = cat_name

class CatalogFiles:
    """ Файлы каталога для реализации связи многие-ко-многим """
    def __init__(self, cat_id, file_id):
        self.cat_id = cat_id
        self.file_id = file_id

# Каталоги
catalogs = [
    Catalog(1, 'Папка 1'),
    Catalog(2, 'Папка 2'),
    Catalog(3, 'Папка 3'),
    Catalog(11, 'Папка 4'),
    Catalog(22, 'Папка 5'),
    Catalog(33, 'Папка 6'),
]

# Файлы
files = [
    File(1, 'Курсовая работа', 25, 1),
    File(2, 'Отчет по лабораторной работе', 35, 2),
    File(3, 'Отчет по практике', 45, 3),
    File(4, 'Реферат', 35, 3),
]
```

```

        File(5, 'Аннотация по АСОИУ', 25, 3),
    ]
    catalogs_files = [
        CatalogFiles(1, 1),
        CatalogFiles(2, 2),
        CatalogFiles(3, 3),
        CatalogFiles(3, 4),
        CatalogFiles(3, 5),
        CatalogFiles(11, 1),
        CatalogFiles(22, 2),
        CatalogFiles(33, 3),
        CatalogFiles(33, 4),
        CatalogFiles(33, 5),
    ]

def main():
    """Основная функция"""
    # Соединение данных один-ко-многим
    one_to_many = [(f.name, f.size, c.cat_name)
                   for c in catalogs
                   for f in files
                   if f.cat_id == c.id]

    # Соединение данных многие-ко-многим
    many_to_many_temp = [(c.cat_name, fc.cat_id, fc.file_id)
                          for c in catalogs
                          for fc in catalogs_files
                          if c.id == fc.cat_id]
    many_to_many = [(f.name, f.size, cat_name)
                    for cat_name, cat_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=itemgetter(2))
    print(res_11)

    print('\nЗадание A2')
    res_12_unsorted = []
    # Перебираем все каталоги
    for c in catalogs:
        # Список файлов каталога
        c_files = list(filter(lambda i: i[2] == c.cat_name, one_to_many))
        # Если каталог не пустой
        if len(c_files) > 0:
            # Размеры файлов каталога
            c_sizes = [size for _, size, _ in c_files]
            # Суммарный размер файлов каталога
            c_sizes_sum = sum(c_sizes)
            res_12_unsorted.append((c.cat_name, c_sizes_sum))

    # Сортировка по суммарному размеру
    res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)
    print(res_12)

    print('\nЗадание A3')
    res_13 = {}
    # Перебираем все каталоги
    for c in catalogs:
        if 'Папка' in c.cat_name:
            # Список файлов каталога
            c_files = list(filter(lambda i: i[2] == c.cat_name,
many_to_many))
            # Только название файлов
            c_files_names = [x for x, _, _ in c_files]
            # Добавляем результат в словарь
            # ключ - каталог, значение - список названий

```

```
        res_13[c.cat_name] = c_files_names
    print(res_13)

if __name__ == '__main__':
    main()
```

## Экранные формы с примерами выполнения работ

```
Задание A1
[('Курсовая работа', 25, 'Папка 1'), ('Отчет по лабораторной работе', 35, 'Папка 2'), ('Отчет по практике', 45, 'Папка 3'), ('Реферат', 35, 'Папка 3'), ('Аннотация по АСОИУ', 25, 'Папка 3')]

Задание A2
[('Папка 3', 105), ('Папка 2', 35), ('Папка 1', 25)]

Задание A3
{'Папка 2': ['Отчет по лабораторной работе'], 'Папка 2 (другая)': ['Отчет по лабораторной работе']}

Process finished with exit code 0
```