

Отчет по РК1 по дисциплине

“Парадигмы и конструкции языков программирования”

Вариант запросов Г. Вариант предметной области 15 (Файл – Каталог файлов)

Запросы:

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

Текст программы:

```
from operator import itemgetter
```

```
class File:
```

```
    """Файл
```

```
    size - размер вайла в МБ"""
```

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

```
        self.id = id
```

```
        self.name = name
```

```
        self.size = size
```

```
        self.catalog_id = catalog_id
```

```
class Catalog:
```

```
    """Каталог файлов"""
```

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

```
        self.id = id
```

```
self.name = name
```

```
class Files_in_Catalog:
```

```
    """
```

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

```
связи многие-ко-многим
```

```
    """
```

```
    def __init__(self, catalog_id, file_id):
```

```
        self.catalog_id = catalog_id
```

```
        self.file_id = file_id
```

```
# Каталоги
```

```
catalogs = [
```

```
    Catalog(1, 'Аспирантские закупки'),
```

```
    Catalog(2, 'Закупки для университета'),
```

```
    Catalog(3, 'Закупки для офиса'),
```

```
    Catalog(4, 'Закупки для партнеров')
```

```
]
```

```
# Файлы
```

```
files = [
```

```
    File(1, 'Сумма закупки', 500, 1),
```

```
    File(2, 'Руководитель закупки', 320, 1),
```

```
    File(3, 'Номер контрактов', 490, 2),
```

```
    File(4, 'Поставщики', 550, 3),
```

```
    File(5, 'Юр. данные', 120, 4)
```

```
]
```

```
files_in_catalog = [
```

```
    Files_in_Catalog(1, 1),
```

```
    Files_in_Catalog(1, 2),
```

```
    Files_in_Catalog(2, 3),
```

```
    Files_in_Catalog(3, 4),
```

```
    Files_in_Catalog(4, 5)
```

```
]
```

```
def main():
```

```
    """Основная функция"""
```

```

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

# Соединение данных многие-ко-многим
many_to_many_temp = [(u.name, fu.catalog_id, fu.file_id)
                      for u in catalogs
                      for fu in files_in_catalog
                      if u.id == fu.catalog_id]

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

print('Задание Г1')
res_Г1 = [item for item in one_to_many if item[2].startswith('A')]
print(res_Г1)

print('\nЗадание Г2')
res_Г2_unsorted = []
# Перебираем все каталоги
for u in catalogs:
    # Список файлов каталога
    u_files = list(filter(lambda i: i[2] == u.name, one_to_many))
    # Если каталог не пустой
    if len(u_files) > 0:
        # Размеры файлов каталога
        u_sizes = [size for _, size, _ in u_files]
        # Максимальный размер файла каталога
        u_max_size = max(u_sizes)
        res_Г2_unsorted.append((u.name, u_max_size))
res_Г2 = sorted(res_Г2_unsorted, key=itemgetter(1), reverse=True)
print(res_Г2)

print('\nЗадание Г3')
res_Г3 = {}
for u in catalogs:
    u_files = list(filter(lambda i: i[2] == u.name, many_to_many))
    u_file_name = [name for name, _, _ in u_files]

```

```
res_Г3[u.name] = u_file_name  
print(res_Г3)
```

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

Результат выполнения программы:

```
"C:\Users\misha\PycharmProjects\lab 3-4\.venv\Scripts\python.exe" "C:\Users\misha\PycharmProjects\lab 3-4\main.py"  
Задание Г1  
[('Сумма закупки', 500, 'Аспирантские закупки'), ('Руководитель закупки', 320, 'Аспирантские закупки')]  
  
Задание Г2  
[('Закупки для офиса', 550), ('Аспирантские закупки', 500), ('Закупки для университета', 490), ('Закупки для партнеров',  
120)]  
  
Задание Г3  
{'Аспирантские закупки': ['Сумма закупки', 'Руководитель закупки'], 'Закупки для университета': ['Номер контрактов'],  
 'Закупки для офиса': ['Поставщики'], 'Закупки для партнеров': ['Юр. данные']}  
  
Process finished with exit code 0
```