Московский государственный технический университет им. Н.Э. Баумана

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

Курс «Парадигмы и конструкции языков программирования»

Отчет по РК №1
Вариант запросов: Д
Вариант предметной области: 20

Выполнил:
студент группы ИУ5-36Б
Ордиянц Эрик

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю. Е.

Москва, 2024 г.

Вариант запросов Д. Предметная область 20.

1. «Поставщик» и «Деталь» связаны соотношением один-ко-многим. Выведите список всех деталей, у которых название заканчивается на «ор», и имена их поставщиков.

2. «Поставщик» и «Деталь» связаны соотношением один-ко-многим. Выведите список поставщиков со средней поставкой деталей, отсортированный по средней поставке деталей (*отдельной функции вычисления среднего значения в Python нет, нужно использовать комбинацию функций вычисления суммы и количества значений*).

3. «Поставщик» и «Деталь» связаны соотношением многие-ко-многим. Выведите список всех поставщиков, у которых имя начинается с буквы «П», и список деталей, которые они поставляют.

Листинг программы.

models.py

```
class Part:
    """Деталь"""
    def __init__(self, part_id, name, supplier_id):
        self.part_id = part_id  # ID детали
        self.name = name  # Наименование детали
        self.supplier_id = supplier_id  # ID поставщика

class Supplier:
    """Поставщик"""
    def __init__(self, supplier_id, name):
        self.supplier_id = supplier_id  # ID поставщика
        self.name = name  # Наименование поставщика

class PartSupply:
    """
    'Поставки деталей' для реализации
    связи многие-ко-многим
    """
    def __init__(self, part_id, supplier_id, quantity):
        self.part_id = part_id  # ID детали
        self.supplier_id = supplier_id  # ID поставщика
        self.quantity = quantity  # Количество поставленной детали
```

data.py

```
from models import Part, Supplier, PartSupply

# Поставщики
suppliers = [
    Supplier(1, 'Алексей'),
    Supplier(2, 'Николай'),
    Supplier(3, 'Пётр'),
]

# Детали
parts = [
    Part(1, "Резистор", 1),
    Part(2, "Конденсатор", 2),
    Part(3, "Транзистор", 1),
    Part(4, "Диод", 3),
]

# Поставки деталей
part_supplies = [
    PartSupply(1, 1, 100),
    PartSupply(2, 1, 200),
    PartSupply(3, 2, 150),
    PartSupply(4, 3, 50),
    PartSupply(1, 3, 300),
]
```

services.py

```
from operator import itemgetter
from data import suppliers, parts, part_supplies

def task_1():
    """Список всех деталей, у которых название заканчивается на 'op', и имена поставщиков"""
    res_1 = [(p.name, s.name)
             for p in parts
             for s in suppliers
             if p.supplier_id == s.supplier_id and p.name.endswith('op')]
    return res_1

def task_2():
    """Список поставщиков со средней поставкой деталей, отсортированный по средней поставке"""
    res_2_unsorted = []
    for s in suppliers:
        total_quantity = sum(ps.quantity for ps in part_supplies if ps.supplier_id == s.supplier_id)
        count_details = len([ps for ps in part_supplies if ps.supplier_id == s.supplier_id])

        if count_details > 0:
            avg_quantity = round(total_quantity / count_details)
            res_2_unsorted.append((s.name, avg_quantity))

    return sorted(res_2_unsorted, key=itemgetter(1), reverse=True)

def task_3():
    """Список всех поставщиков, у которых имя начинается с буквы 'П', и детали, которые они поставляют"""
    res_3 = {}
    for s in suppliers:
        if s.name.startswith('П'):
            s_parts = [p.name for p in parts if p.supplier_id == s.supplier_id]
            res_3[s.name] = s_parts
    return res_3
```

main.py

```
from services import task_1, task_2, task_3

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

    print('Задание Д1')
    res_1 = task_1()
    print(res_1)

    print('\nЗадание Д2')
    res_2 = task_2()
    print(res_2)

    print('\nЗадание Д3')
    res_3 = task_3()
    print(res_3)

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

Результат выполнения.

```
● →  PCPL git:(main) p RK1/main.py
  Задание Д1
  [('Резистор', 'Алексей'), ('Конденсатор', 'Николай'), ('Транзистор', 'Алексей')]

  Задание Д2
  [('Пётр', 175), ('Алексей', 150), ('Николай', 150)]

  Задание Д3
  {'Пётр': ['Диод']}
```