

## Егоров Дмитрий ИУ5-31Б Вариант 7

```
from operator import itemgetter

class Computer:
    """Класс для представления Компьютера"""
    def __init__(self, id, name, price, micro_id):
        self.id = id
        self.name = name
        self.price = price
        self.micro_id = micro_id

class Microprocessor:
    """Класс для представления Микропроцессора"""
    def __init__(self, id, model):
        self.id = id
        self.model = model

class CompMicro:
    """
    Связь многие-ко-многим между компьютерами и микропроцессорами
    """
    def __init__(self, comp_id, micro_id):
        self.comp_id = comp_id
        self.micro_id = micro_id

# Список микропроцессоров
microprocessors = [
    Microprocessor(1, 'Intel Core i7'),
    Microprocessor(2, 'Intel Core i6'),
    Microprocessor(3, 'AMD Ryzen 5'),
    Microprocessor(4, 'AMD Ryzen 4'),
    Microprocessor(5, 'Intel Xeon'),
]

# Список компьютеров
computers = [
    Computer(1, 'Персональный компьютер', 120000, 1),
    Computer(2, 'Персональный компьютер', 121000, 5),
    Computer(3, 'Персональный компьютер', 121000, 4),
    Computer(4, 'Серверный компьютер', 150000, 1),
    Computer(5, 'Серверный компьютер', 130000, 2),
    Computer(6, 'Серверный компьютер', 125252, 3),
    Computer(7, 'Рабочая станция', 52, 3),
]

# Связь многие-ко-многим между компьютерами и микропроцессорами
comp_micros = [
    CompMicro(1, 1),
    CompMicro(2, 1),
    CompMicro(3, 2),
    CompMicro(4, 3),
    CompMicro(5, 2),
    CompMicro(5, 1),
    CompMicro(5, 4),
    CompMicro(2, 5),
    CompMicro(1, 3),
]

def one_to_many_relationship():
    """Создает связь один-ко-многим: выводит все компьютеры для каждого процессора"""
```

```

        return [(m.model, c.name)
                for m in microprocessors
                for c in computers
                if c.micro_id == m.id]

def many_to_many_relationship():
    """Создает связь многие-ко-многим между компьютерами и
    микропроцессорами"""
    many_to_many_temp = [(m.model, c.name, c.price)
                        for m in microprocessors
                        for cm in comp_micros
                        for c in computers
                        if cm.micro_id == m.id and cm.comp_id == c.id]

    return many_to_many_temp

def task_A1():
    """Задание A1: Список всех компьютеров для каждого микропроцессора"""
    one_to_many = one_to_many_relationship()
    result = {}
    for model, comp_name in one_to_many:
        if model in result:
            result[model].append(comp_name)
        else:
            result[model] = [comp_name]

    for model, comps in result.items():
        print(f"{model}: {' '.join(comps)}")

def task_A2():
    """Задание A2: Суммарная стоимость всех компьютеров, использующих каждый
    микропроцессор"""
    many_to_many = many_to_many_relationship()
    result = []
    for m in microprocessors:
        m_computers = list(filter(lambda x: x[0] == m.model, many_to_many))
        total_price = sum(comp_price for _, _, comp_price in m_computers)
        result.append((m.model, total_price))

    result = sorted(result, key=itemgetter(1), reverse=True)
    for model, total in result:
        print(f"{model}: {total} Py6")

def task_A3(keyword):
    """Задание A3: Вывод всех процессоров с определенным словом и их
    компьютеров"""
    many_to_many = many_to_many_relationship()
    result = {}
    for model, comp_name, _ in many_to_many:
        if keyword.lower() in model.lower():
            if model in result:
                result[model].append(comp_name)
            else:
                result[model] = [comp_name]

    for model, comps in result.items():
        print(f"{model}: {' '.join(comps)}")

if __name__ == '__main__':
    print('Задание A1')
    task_A1()

    print('\nЗадание A2')

```

```
task_A2()  
  
print('\nЗадание A3')  
task_A3('Intel')
```

Задание A1

Intel Core i7: Персональный компьютер, Серверный компьютер

Intel Core i6: Серверный компьютер

AMD Ryzen 5: Серверный компьютер, Рабочая станция

AMD Ryzen 4: Персональный компьютер

Intel Xeon: Персональный компьютер

Задание A2

Intel Core i7: 371000 Руб

AMD Ryzen 5: 270000 Руб

Intel Core i6: 251000 Руб

AMD Ryzen 4: 130000 Руб

Intel Xeon: 121000 Руб

Задание A3

Intel Core i7: Персональный компьютер, Персональный компьютер, Серверный компьютер

Intel Core i6: Персональный компьютер, Серверный компьютер

Intel Xeon: Персональный компьютер