

**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 рубежному контролю №1
Вариант 9А

Выполнил:

студент группы ИУ5-35Б
Лахин Никита

Подпись и дата:

Проверил:

преподаватель каф. ИУ5
Гапанюк Юрий Евгеньевич

Подпись и дата:

Постановка задачи

1. «Операционная система» и «компьютер» связаны соотношением один-ко-многим. Выведите список всех компьютеров и связанных с ними операционных систем (отсортированный по операционным системам).
2. «Операционная система» и «компьютер» связаны соотношением один-ко-многим. Выведите список операционных систем с указанием общего количества компьютеров, работающих под управлением каждой ОС (отсортированный по общему количеству компьютеров).
3. «Операционная система» и «компьютер» связаны соотношением многие-ко-многим. Выведите список операционных систем, в названии которых присутствует слово "Windows", и список компьютеров, на которых они установлены.

Текст программы

```
from operator import itemgetter

class Computer:
    def __init__(self, computer_id, brand, model, os_id):
        self.computer_id = computer_id
        self.brand = brand
        self.model = model
        self.os_id = os_id

class OperatingSystem:
    def __init__(self, os_id, name):
        self.os_id = os_id
        self.name = name

class ComputerOS:
    def __init__(self, computer_id, os_id):
        self.computer_id = computer_id
        self.os_id = os_id

computers = [
    Computer(1, 'Dell', 'XPS 13', 1),
    Computer(2, 'HP', 'EliteBook', 2),
    Computer(3, 'Apple', 'MacBook Pro', 3),
    Computer(4, 'Lenovo', 'ThinkPad', 1),
    Computer(5, 'Asus', 'ZenBook', 2),
]

operating_systems = [
    OperatingSystem(1, 'Windows 10'),
    OperatingSystem(2, 'Ubuntu 20.04'),
    OperatingSystem(3, 'macOS'),
]

computer_os_relationship = [
    ComputerOS(1, 1),
    ComputerOS(2, 2),
    ComputerOS(3, 3),
    ComputerOS(4, 1),
    ComputerOS(5, 2),
]
```

```

]

def main():
    one_to_many = [(c.brand, c.model, os.name)
                    for c in computers
                    for os in operating_systems
                    if c.os_id == os.os_id]

    many_to_many_temp = [(os.name, co.computer_id, co.os_id)
                          for os in operating_systems
                          for co in computer_os_relationship
                          if os.os_id == co.os_id]

    many_to_many = [(c.brand, c.model, os_name)
                    for os_name, computer_id, os_id in many_to_many_temp
                    for c in computers if c.computer_id == computer_id]

    print('Запрос 1: Список всех компьютеров и связанных с ними
операционных систем (отсортированный по операционным системам):')
    res_1 = sorted(one_to_many, key=itemgetter(2))
    print(res_1)

    print('\nЗапрос 2: Список операционных систем с указанием общего
количества компьютеров, работающих под управлением каждой ОС
(отсортированный по общему количеству компьютеров):')
    os_counts = {}
    for os in operating_systems:
        os_name = os.name
        os_computers = list(filter(lambda i: i[2] == os_name,
one_to_many))
        os_counts[os_name] = len(os_computers)

    res_2 = sorted(os_counts.items(), key=lambda x: x[1],
reverse=True)
    print(res_2)

    print('\nЗапрос 3: Список операционных систем, в названии
которых присутствует слово "Windows", и список компьютеров, на
которых они установлены:')
    res_3 = {}
    for os in operating_systems:
        if 'Windows' in os.name:
            os_name = os.name
            os_computers = list(filter(lambda i: i[2] == os_name,
many_to_many))
            computer_list = [f"{computer[0]} {computer[1]}" for
computer in os_computers]
            res_3[os_name] = computer_list

    print(res_3)

if __name__ == '__main__':
    main()

```