

**Министерство науки и высшего образования Российской Федерации**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**



**«Московский государственный технический университет**  
**имени Н.Э. Баумана**  
**(национальный исследовательский университет)»**  
**(МГТУ им. Н.Э. Баумана)**

**Факультет «Информатика и системы управления»**

**Курс «Базовые компоненты интернет-технологий»**

**Рубежный контроль №1**

**Выполнил:**  
**студент группы ИУ5-33Б**  
**Кузнецов В. А.**

**Проверил:**  
**Преподаватель кафедры ИУ-5**  
**Гапанюк Ю. Е.**

**2022 г.**

## Текст программы

```
from operator import itemgetter
from math import inf

class HardDrive:
    """Жёсткий диск"""

    def __init__(self, id, model, size, computer_id):
        self.id = id
        self.model = model
        self.size = size
        self.computer_id = computer_id

class Computer:
    """Компьютер"""

    def __init__(self, id, processor):
        self.id = id
        self.processor = processor

class HDComp:
    """
    'Жёсткий диск компьютера' для реализации
    СВЯЗИ МНОГИЕ-КО-МНОГИМ
    """

    def __init__(self, computer_id, hard_drive_id):
        self.computer_id = computer_id
        self.hard_drive_id = hard_drive_id

# Компьютеры
computers = [
    Computer(1, 'Intel Core i3-6100'),
    Computer(2, 'Intel Core i3-7300'),
    Computer(3, 'Intel Core i5-7400'),

    Computer(11, 'Intel Core i3-7100'),
    Computer(22, 'AMD Ryzen 3 3100'),
    Computer(33, 'Intel Core i5-6500'),
]

# Жёсткие диски
hard_drives = [
    HardDrive(1, 'Western Digital Blue WD10EZEX', 1024, 1),
    HardDrive(2, 'Seagate Barracuda ST500LM030', 500, 2),
    HardDrive(3, 'Western Digital Purple WD40PURX', 4096, 3),
    HardDrive(4, 'Seagate SkyHawk ST2000VX008', 2048, 3),
    HardDrive(5, 'Western Digital Blue WD5000LPCX', 500, 3),
]

hds_comps = [
    HDComp(1, 1),
    HDComp(2, 2),
    HDComp(3, 3),
    HDComp(3, 4),
]
```

```

        HDComp(3, 5),

        HDComp(11, 1),
        HDComp(22, 2),
        HDComp(33, 3),
        HDComp(33, 4),
        HDComp(33, 5),
    ]

def print_by_line(data):
    for line in data:
        print(line)

def task_1(one_to_many):
    """
    «Компьютер» и «Жёсткий диск» связаны соотношением один-ко-многим.
    Выведите список всех жёстких дисков фирмы «Western Digital»,
    и названия процессоров компьютеров, в которые они установлены.
    """

    print('Задание A1')
    print_by_line([(record[0], record[2]) for record in one_to_many if
                    record[0].startswith('Western Digital')])

def task_2(one_to_many):
    """
    «Компьютер» и «Жёсткий диск» связаны соотношением один-ко-многим.
    Выведите список процессоров компьютеров с минимальным объёмом диска в
    каждом компьютере,
    отсортированный по минимальному объёму диска.
    """

    print('\nЗадание A2')
    mins = {}
    for model, size, processor in one_to_many:
        mins[processor] = min(mins.get(processor, inf), size)
    print_by_line(sorted(mins.items(), key=itemgetter(1)))

def task_3(many_to_many):
    """
    «Компьютер» и «Жёсткий диск» связаны соотношением многие-ко-многим.
    Выведите список всех связанных жёстких дисков и компьютеров,
    отсортированный по объёму жёстких дисков, сортировка по компьютерам
    произвольная.
    """

    print('\nЗадание A3')
    print_by_line(sorted(many_to_many, key=itemgetter(1)))

```

```

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

    # Соединение данных один-ко-многим
    one_to_many = [(hard_drive.model, hard_drive.size, computer.processor)
                    for hard_drive in hard_drives
                    for computer in computers
                    if hard_drive.computer_id == computer.id]

    # Соединение данных многие-ко-многим
    many_to_many_temp = [(computer.processor, hd_c.computer_id,
hd_c.hard_drive_id)
                           for computer in computers
                           for hd_c in hds_comps
                           if computer.id == hd_c.computer_id]

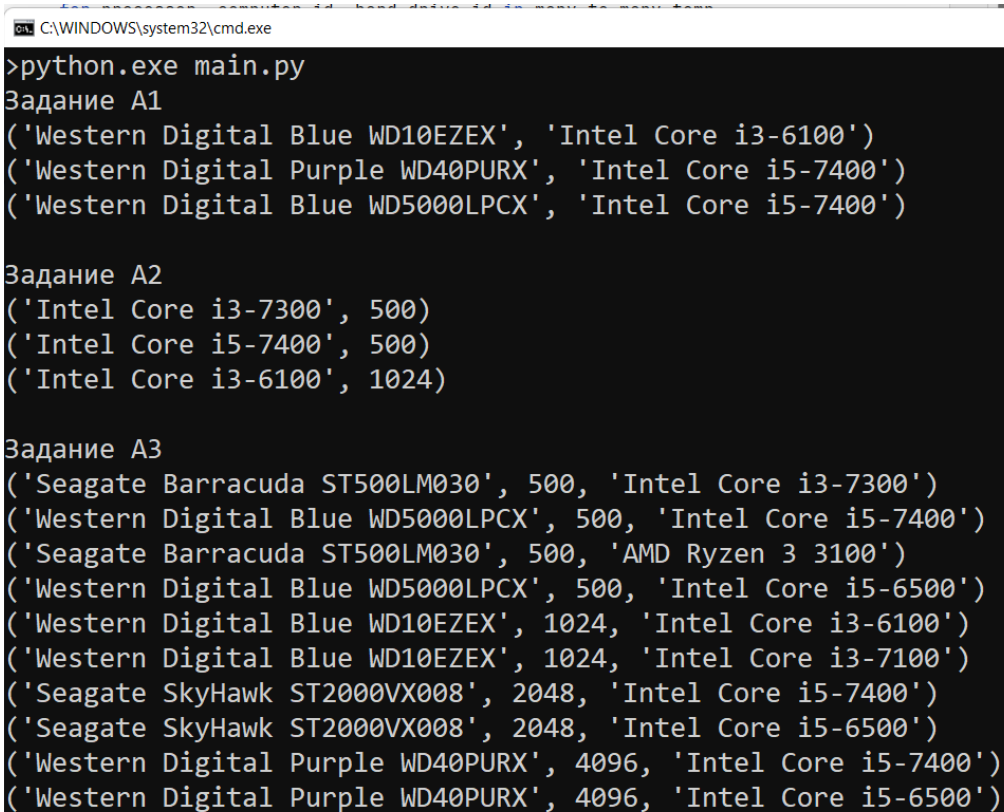
    many_to_many = [(hard_drive.model, hard_drive.size, processor)
                     for processor, computer_id, hard_drive_id in
many_to_many_temp
                     for hard_drive in hard_drives
                     if hard_drive.id == hard_drive_id]

    task_1(one_to_many)
    task_2(one_to_many)
    task_3(many_to_many)

if __name__ == '__main__':
    main()

```

## Результат выполнения



```

C:\WINDOWS\system32\cmd.exe
>python.exe main.py
Задание A1
('Western Digital Blue WD10EZEX', 'Intel Core i3-6100')
('Western Digital Purple WD40PURX', 'Intel Core i5-7400')
('Western Digital Blue WD5000LPCX', 'Intel Core i5-7400')

Задание A2
('Intel Core i3-7300', 500)
('Intel Core i5-7400', 500)
('Intel Core i3-6100', 1024)

Задание A3
('Seagate Barracuda ST500LM030', 500, 'Intel Core i3-7300')
('Western Digital Blue WD5000LPCX', 500, 'Intel Core i5-7400')
('Seagate Barracuda ST500LM030', 500, 'AMD Ryzen 3 3100')
('Western Digital Blue WD5000LPCX', 500, 'Intel Core i5-6500')
('Western Digital Blue WD10EZEX', 1024, 'Intel Core i3-6100')
('Western Digital Blue WD10EZEX', 1024, 'Intel Core i3-7100')
('Seagate SkyHawk ST2000VX008', 2048, 'Intel Core i5-7400')
('Seagate SkyHawk ST2000VX008', 2048, 'Intel Core i5-6500')
('Western Digital Purple WD40PURX', 4096, 'Intel Core i5-7400')
('Western Digital Purple WD40PURX', 4096, 'Intel Core i5-6500')

```