

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет «Радиотехнический»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Разработка интернет-приложений»
Рубежный контроль №1
Вариант Е7

Выполнил:
студент группы РТ5-51Б

Грызин Алексей

Подпись и дата:

Проверил:
преподаватель каф.
ИУ5

Гапанюк Ю.Е.

Подпись и дата:

Описание задания

1. «Микропроцессор» и «Компьютер» связаны соотношением один-ко-многим. Выведите список всех Микропроцессоров, у которых в названии присутствует слово «Микропроцессор», и список работающих в них Компьютеров.
2. «Микропроцессор» и «Компьютер» связаны соотношением один-ко-многим. Выведите список Микропроцессоров со средней зарплатой Компьютеров в каждом Микропроцессоре, отсортированный по средней зарплате. Средняя зарплата должна быть округлена до 2 знака после запятой (*Микропроцессорной функции вычисления среднего значения в Python нет, нужно использовать комбинацию функций вычисления суммы и количества значений; для округления необходимо использовать функцию <https://docs.python.org/3/library/functions.html#round>*).
3. «Микропроцессор» и «Компьютер» связаны соотношением многие-ко-многим. Выведите список всех Компьютеров, у которых фамилия начинается с буквы «А», и названия их Микропроцессоров.

Текст программы

models.py

```
class Processor:
    """Процессор"""
    def __init__(self, id, title, num_cores, freq, price):
        self.id = id
        self.title = title
        self.numb_cores = num_cores
        self.freq = freq
        self.price = price

class PC:
    """Компьютер"""
    def __init__(self, id, title, proc_id, price):
        self.id = id
        self.title = title
        self.proc_id = proc_id
        self.price = price

class ProcessorPC:
    """
    Процессоры компьютеров
    """
    def __init__(self, proc_id, pc_id):
        self.proc_id = proc_id
        self.pc_id = pc_id
```

mocks.py

```
from models import *

processors = [
    Processor(1, 'AMD FX 4300', 4, 3800, 4000),
    Processor(2, 'AMD FX 8300', 8, 3500, 8000),
    Processor(3, 'Микропроцессор AMD Ryzen 7 3800', 8, 3800, 16000),
    Processor(4, 'Intel Core i7-6700', 8, 3800, 16000),
    Processor(5, 'Микропроцессор Intel Core i7-6700K', 8, 4000, 20000),
]

pcs = [
    PC(1, 'BichMachina', 1, 20000),
    PC(2, 'Dota2PC', 5, 50000),
    PC(3, 'NaMinimalkah', 2, 25000),
    PC(4, 'PC4', 3, 35023),
    PC(5, 'PC5', 3, 40123),
    PC(6, 'PC6', 4, 37421),
    PC(7, 'TOP-PC', 5, 52421),
```

```

PC(8, 'BEST-PC', 4, 40000),
PC(9, 'Античный ПК', 1, 18000),
PC(10, 'Автобот', 3, 45000),
PC(11, 'Антивирус', 2, 40322)
]

proc_pcs = [
    ProcessorPC(1, 1),
    ProcessorPC(1, 9),
    ProcessorPC(2, 3),
    ProcessorPC(2, 11),
    ProcessorPC(3, 4),
    ProcessorPC(3, 5),
    ProcessorPC(3, 10),
    ProcessorPC(4, 6),
    ProcessorPC(4, 8),
    ProcessorPC(5, 2),
    ProcessorPC(5, 7),
]

```

main.py

```

from mocks import *

def main():
    one_to_many = [(proc.title, proc.id, pc.title, pc.price)
                    for proc in processors
                    for pc in pcs
                    if pc.proc_id == proc.id]

    many_to_many_temp = [(proc.title, proc_pc.proc_id, proc_pc.pc_id)
                          for proc in processors
                          for proc_pc in proc_pcs
                          if proc.id==proc_pc.proc_id]

    many_to_many = [(pc.title, title)
                    for title, proc_id, pc_id in many_to_many_temp
                    for pc in pcs if pc.id==pc_id]

    res_1 = list(filter(lambda i: i[0].lower().find('микропроцессор') != -
1, one_to_many))

    res_2 = []
    res_2_ids = []

    for i in one_to_many:
        if res_2_ids.count(i[1]) != 0:
            continue
        buf = list(filter(lambda x: x[1] == i[1], one_to_many))
        res_2_ids.append(i[1])

```

```

        sum = 0
        count = len(buf)
        for j in buf:
            sum += j[3]
        res_2.append((i[0], round(sum / count, 2)))

a = ['a', 'a']

res_3 = list(filter(lambda i: a.count(i[0][0].lower()) != 0, many_to_many))

print('Задание 1', res_1)
print('Задание 2', res_2)
print('Задание 3', res_3)

if __name__ == "__main__":
    main()

```

Результат работы программы

```

feelsbadmans@feelsbadmans-desktop:~/Univer/bmstu-4-sem-rip$ /bin/python3 /home/feelsbadmans/Univer/bmstu-4-sem-rip/rk1/main.py
Задание 1 [('Микропроцессор AMD Ryzen 7 3800', 3, 'PC4', 35023), ('Микропроцессор AMD Ryzen 7 3800', 3, 'PC5', 40123), ('Микропроцессор AM
D Ryzen 7 3800', 3, 'Автобот', 45000), ('Микропроцессор Intel Core i7-6700K', 5, 'Dota2PC', 50000), ('Микропроцессор Intel Core i7-6700K',
5, 'TOP-PC', 52421)]
Задание 2 [('AMD FX 4300', 19000.0), ('AMD FX 8300', 32661.0), ('Микропроцессор AMD Ryzen 7 3800', 40048.67), ('Intel Core i7-6700', 38710
.5), ('Микропроцессор Intel Core i7-6700K', 51210.5)]
Задание 3 [('Античный ПК', 'AMD FX 4300'), ('Антивирус', 'AMD FX 8300'), ('Автобот', 'Микропроцессор AMD Ryzen 7 3800')]
feelsbadmans@feelsbadmans-desktop:~/Univer/bmstu-4-sem-rip$ git add

```