

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

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

Курс «Разработка интернет-приложений»

Отчет по рубежному контролю №1

Выполнил:

студент группы ИУ5-53Б

Ветошкин Артём

Подпись и дата:

Проверил:

преподаватель каф. ИУ5

Гапанюк Ю.Е.

Подпись и дата:

Москва, 2021 г.

Ветошкин Артём Алексеевич ИУ5-53Б

Задание:

Вариант В.

«Микропроцессор» и «Компьютер» связаны соотношением один-ко-многим. Выведите список всех ПК, у которых название начинается со слова "Apple", и названия их процессора.

«Микропроцессор» и «Компьютер» связаны соотношением один-ко-многим. Выведите список процессоров с минимальной стоимостью ПК использующих этот процессор, отсортированный по минимальной стоимости.

«Микропроцессор» и «Компьютер» связаны соотношением многие-ко-многим. Выведите список всех связанных процессоров и ПК, отсортированный по процессорам, сортировка по ПК произвольная.

```
from model import PC, Processor, PC_Processor
import utils
```

Создадим таблицу Микропроцессорами с полями id, name, frequency, memory-cash:

```
list_of_processors = [Processor(1, "Intel i5-9600", 3600, 128),
                      Processor(2, "Intel i7-10600", 3700, 128),
                      Processor(3, "Intel i5-6500", 3600, 12),
                      Processor(4, "Intel i5-6600", 3700, 32),
                      Processor(5, "Intel i7-8600", 3500, 64),
                      Processor(6, "Intel i7-10600", 3800, 256),
                      Processor(7, "Intel i3-5600", 3500, 128),
                      Processor(8, "Intel i7-4790", 3200, 128),
                      Processor(9, "Ryzen 5 5600X", 3800, 32),
                      Processor(10, "Ryzen 7 5800X", 3700, 32),
                      Processor(11, "Ryzen 9 5900X", 3700, 32),
                      Processor(12, "Ryzen 9 5950X", 3400, 64)]
```

Создадим таблицу Компьютеры с полями id, name, price

```
list_of_pc = [PC(1, "Lenovo IdeaCentre G5 14IMB05", 88000, 8),
              PC(2, "HP Pavilion Gaming TG01", 110000, 1),
              PC(3, "ZOTAC MAGNUS ONE", 187000, 7),
              PC(4, "Apple Mac mini 2020 M1", 173000, 6),
              PC(5, "Acer Aspire TC-895", 34000, 7),
              PC(6, "Apple iMac 24\" 2021", 248000, 11),
              PC(7, "Gigabyte GB-BR", 93000, 4),
              PC(8, "HP M01", 61000, 3)]
```

Создадим таблицу связи много ко многу этих двух таблиц

```
list_of_pc_processor = [PC_Processor(5, 2),
                        PC_Processor(10, 1),
                        PC_Processor(4, 1),
                        PC_Processor(4, 2),
                        PC_Processor(7, 5),
                        PC_Processor(2, 1),
                        PC_Processor(10, 3),
                        PC_Processor(4, 2),
                        PC_Processor(10, 1),
                        PC_Processor(2, 5)]
```

1 запрос

«Микропроцессор» и «Компьютер» связаны соотношением один-ко-многим. Выведите список всех ПК, у которых название начинается со слова "Apple", и названия их процессора.

```
prefix = "Apple"
correct_pc = list(filter(lambda x: x.name.startswith(prefix), list_of_pc))
id_processors = list(map(lambda x: x.proc_id, correct_pc))
correct_processors = list(map(lambda x: (x.name, x.id), filter(lambda x: x.id in
id_processors, list_of_processors)))

processor_by_id = {}
for (name, proc_id) in correct_processors:
    processor_by_id[proc_id] = name

results = [["name", "price", "pc names"]]
for pc in correct_pc:
    tmp_arr = []
    tmp_arr.extend(pc.values())
    if pc.proc_id in processor_by_id:
        tmp_arr.append(processor_by_id[pc.proc_id])
    else:
        tmp_arr.append("-----")
    results.append(tmp_arr)

utils.print_pretty_table(results)
```

```
name | price | pc names
-----+-----+-----
```

```
Apple Mac mini 2020 M1 | 173000 | Intel i7-10600
Apple iMac 24" 2021 | 248000 | Ryzen 9 5900X
```

2 запрос

«Микропроцессор» и «Компьютер» связаны соотношением один-ко-многим. Выведите список процессоров с минимальной стоимостью ПК использующих этот процессор, отсортированный по минимальной стоимости.

```
id_processors = list(map(lambda x: x.id, list_of_processors))
prices = list(map(lambda x: (x.price, x.proc_id), filter(lambda x: x.proc_id in
id_processors, list_of_pc)))

correct_price_by_id = {}
for (price, proc_id) in prices:
    if proc_id not in correct_price_by_id or correct_price_by_id[proc_id] > price:
        correct_price_by_id[proc_id] = price

results = []
for processor in list_of_processors:
    tmp_arr = []
    tmp_arr.extend(processor.values())
    if processor.id in correct_price_by_id:
        tmp_arr.append(correct_price_by_id[processor.id])
    else:
        tmp_arr.append(utils.max_int)
    results.append(tmp_arr)

results = [{"name", "frequency", "cash memory", "price"}] + list(map(lambda x:
utils.change_elem(x, "-----", 3) if x[3] == utils.max_int else x,

sorted(results, key=lambda x: x[3])))

utils.print_pretty_table(results)
```

name	frequency	cash memory	price
Intel i3-5600	3500	128	34000
Intel i5-6500	3600	12	61000
Intel i7-4790	3200	128	88000
Intel i5-6600	3700	32	93000
Intel i5-9600	3600	128	110000
Intel i7-10600	3800	256	173000
Ryzen 9 5900X	3700	32	248000
Intel i7-10600	3700	128	-----
Intel i7-8600	3500	64	-----
Ryzen 5 5600X	3800	32	-----

Ryzen 7 5800X	3700	32	-----
Ryzen 9 5950X	3400	64	-----

3 запрос

«Микропроцессор» и «Компьютер» связаны соотношением многие-ко-многим. Выведите список всех связанных процессоров и ПК, отсортированный по ПК, сортировка по процессорам произвольная.

```
id_processors = list(map(lambda x: x.id, list_of_processors))
pcs = list(filter(lambda x: x.proc_id in id_processors, list_of_pc))

pcs_by_id = {}
for pc in pcs:
    if pc.proc_id in pcs_by_id:
        pcs_by_id[pc.proc_id].append(pc)
    else:
        pcs_by_id[pc.proc_id] = [pc]

for key in pcs_by_id.keys():
    pcs_by_id[key] = sorted(pcs_by_id[key], key=lambda x: x.name)

results = [["name", "frequency", "cash memory", "pc"]]
for processor in list_of_processors:
    tmp_arr = []
    tmp_arr.extend(processor.values())
    if processor.id in pcs_by_id:
        tmp_arr.append(pcs_by_id[processor.id])
    else:
        tmp_arr.append("-----")
    results.append(tmp_arr)

utilits.print_pretty_table(results)
```

```
name | frequency | cash memory |
pc
-----+-----+-----+-----
-----
Intel i5-9600 | 3600 | 128 |
[PC with id: 2; name: HP Pavilion Gaming TG01; price: 110000; proc_id: 1]
Intel i7-10600 | 3700 | 128 |
-----
Intel i5-6500 | 3600 | 12 |
[PC with id: 8; name: HP M01; price: 61000; proc_id: 3]
Intel i5-6600 | 3700 | 32 |
[PC with id: 7; name: Gigabyte GB-BR; price: 93000; proc_id: 4]
Intel i7-8600 | 3500 | 64 |
-----
```

```
Intel i7-10600 |      3800 |      256 |  
[PC with id: 4; name: Apple Mac mini 2020 M1; price: 173000; proc_id: 6]  
Intel i3-5600 |      3500 |      128 | [PC with id: 5; name: Acer Aspire TC-  
895; price: 34000; proc_id: 7, PC with id: 3; name: ZOTAC MAGNUS ONE; price:  
187000; proc_id: 7]  
Intel i7-4790 |      3200 |      128 |  
[PC with id: 1; name: Lenovo IdeaCentre G5 14IMB05; price: 88000; proc_id: 8]  
Ryzen 5 5600X |      3800 |      32 |  
-----  
Ryzen 7 5800X |      3700 |      32 |  
-----  
Ryzen 9 5900X |      3700 |      32 |  
[PC with id: 6; name: Apple iMac 24" 2021; price: 248000; proc_id: 11]  
Ryzen 9 5950X |      3400 |      64 |  
-----
```