



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 КАФЕДРА _____ Системы
обработки информации и управления

Рубежный контроль №1

По курсу «Разработка интернет-приложений»

Подготовил:
Студент группы ИУ5-55Б.
Кожиев Т.К.
24.10.2021

Проверил:
Гапанюк Ю.Е.

Вариант предметной области: 7

Класс 1: Микропроцессор

Класс 2: Компьютер

Вариант запросов: Д

1. «Микропроцессор» и «Компьютер» связаны соотношением один-ко-многим. Выведите список всех микропроцессоров, у которых название хранимой записи содержит "", и названия компьютеров с этими микропроцессорами.
2. «микропроцессоры» и «Компьютер» связаны соотношением один-ко-многим. Выведите список компьютеров со средней датой публикации микропроцессоров в каждом компьютере, отсортированный по средней дате публикации.
(отдельной функции вычисления среднего значения в Python нет, нужно использовать комбинацию функций вычисления суммы и количества значений).
3. «Микропроцессоры» и «Компьютер» связаны соотношением многие-ко-многим. Выведите список всех компьютеров, у которых название начинается с буквы «А», и список их микропроцессоров.

Модели:

Микропроцессор:

```
class Microprocessor:
    """microprocessor"""

    def __init__(self, id, name, publication_year, computer_id):
        self.id = id
        self.name = name
        self.publication_year = publication_year
        self.computer_id = computer_id
```

Компьютер

```
class Computer:
    """Компьютер"""

    def __init__(self, id, name):
        self.id = id
        self.name = name
```

Компьютеры с микропроцессорами:

```
class MicroprocessorComputer:
    """
    Микропроцессоры компьютеров
    для реализации связи многие-ко-многим
    """

    def __init__(self, computer_id, microprocessor_id):
        self.microprocessor_id = microprocessor_id
        self.computer_id = computer_id
```

База данных:

Микропроцессоры:

```
microprocessors = [
    Microprocessor(1, 'MCIMX515DJM8C', 2000, 7),
    Microprocessor(2, 'D9833', 2009, 5),
    Microprocessor(3, 'SP708SEN', 2015, 1),
    Microprocessor(11, 'Arduino', 1999, 10),
    Microprocessor(22, 'ADM1232ARNZ', 2005, 12)
```

Компьютеры:

```
computers = [  
    Computer(1, 'Lenovo IdeaCentre 3 07ADA05'),  
    Computer(2, 'Lenovo T460s'),  
    Computer(3, 'iRU Home 315'),  
    Computer(4, 'Intel NUC Frost Canyon'),  
    Computer(5, 'iRU Home 310 298386'),  
    Computer(6, 'Dell Vostro 3267-5090'),  
    Computer(7, 'Dell Optiplex MT 3070-7681'),  
    Computer(8, 'Acer Nitro N50-600'),  
    Computer(9, 'Acer Predator P03-600'),  
    Computer(10, 'Apple Mac mini i7'),  
    Computer(11, 'Apple Mac Pro Intel Xeon 8'),  
    Computer(12, 'HP Pavilion Gaming TG01-2011ur')  
]
```

Компьютеры с операционными системами:

```
computers_with_microprocessor = [
    MicroprocessorComputer(1, 3),
    MicroprocessorComputer(1, 2),
    MicroprocessorComputer(1, 1),
    MicroprocessorComputer(2, 3),
    MicroprocessorComputer(3, 1),
    MicroprocessorComputer(3, 22),
    MicroprocessorComputer(4, 3),
    MicroprocessorComputer(5, 2),
    MicroprocessorComputer(6, 2),
    MicroprocessorComputer(7, 1),
    MicroprocessorComputer(8, 1),
    MicroprocessorComputer(9, 1),
    MicroprocessorComputer(9, 2),
    MicroprocessorComputer(9, 3),
    MicroprocessorComputer(9, 11),
    MicroprocessorComputer(10, 11),
    MicroprocessorComputer(11, 11),
    MicroprocessorComputer(11, 12),
    MicroprocessorComputer(12, 1),
    MicroprocessorComputer(12, 2),
    MicroprocessorComputer(12, 3),
    MicroprocessorComputer(12, 11),
    MicroprocessorComputer(12, 22),
```

Выполнение задания:

```
print('Задание Д-1')
# Выведем id, name, publication_year таблицы "microprocessor"
# При этом name != Windows
# И выведем компьютеры этих микропроцессоров
D1 = [(x['microprocessor'].id, x['microprocessor'].name, x['microprocessor'].publication_year, x['computers'].name)
      for x in microprocessor_join_computers
      if x['microprocessor'].name.find('D9833') != - 1]
for x in D1:
    print(x)
```

Принтскрин с результатом:

Задание Д-1

```
(2, 'D9833', 2009, 'iRU Home 310 298386')
```

Выполнение задания Д-2

```
print('\nЗадание Д-2')
# Выведем имя компьютера, среднее по дате публикации микропроцессоры этого компьютера
# Сортируя по этому среднему

# Заведем таблицу с накапливаемой суммой дат и кол-вом ОС:
computer_sum_count_dict = {}
for os_computers_row in microprocessor_join_computers:
    computer_name = os_computers_row['computers'].name
    publication_year = os_computers_row['microprocessor'].publication_year

    if computer_name in computer_sum_count_dict:
        computer_sum_count_dict[computer_name]['sum'] = computer_sum_count_dict[computer_name]['sum'] + publication_year
        computer_sum_count_dict[computer_name]['count'] = computer_sum_count_dict[computer_name]['count'] + 1
    else:
        computer_sum_count_dict[computer_name] = {'sum': publication_year, 'count': 1}

D2 = sorted(
    [(computer_name, computer_sum_count_dict[computer_name]['sum'] / computer_sum_count_dict[computer_name]['count'])
     for computer_name in computer_sum_count_dict
     if computer_sum_count_dict[computer_name]['count'] != 0],
    key=itemgetter(1), reverse=True
)
for x in D2:
    print(x)
```

Принтскрин с результатом:

Задание Д-2

```
('Lenovo IdeaCentre 3 07ADA05', 2015.0)
('iRU Home 310 298386', 2009.0)
('HP Pavilion Gaming TG01-2011ur', 2005.0)
('Dell Optiplex MT 3070-7681', 2000.0)
('Apple Mac mini i7', 1999.0)
```

Выполнение задания Д-3

```

print('\nЗадание Д-3')

# Соединение данных многие-ко-многим
many_to_many = [(c.name, co.computer_id, co.microprocessor_id)
                 for c in computers
                 for co in computers_with_microprocessor
                 if c.id == co.computer_id]

computers_with_microprocessor_table = [(microprocessor.name, microprocessor.publication_year, computer_name)
                                       for computer_name, computer_id, microprocessor_id in many_to_many
                                       for microprocessor in microprocessors if microprocessor.id == microprocessor_id]

D3 = {}
for computer in computers:
    if computer.name.startswith('A'):
        microprocessor_of_computer = list(filter(lambda i: i[2] == computer.name, computers_with_microprocessor_table))
        microprocessor_names = [x for x, _, _ in microprocessor_of_computer]
        D3[computer.name] = microprocessor_names

print(D3)

if __name__ == '__main__':
    main()

```

Принтскрин с результатом:

```

Задание Д-3
{'Acer Nitro N50-600': ['MCINX515DJM8C'], 'Acer Predator P03-600': ['MCINX515DJM8C', 'D9833', 'SP708SEN', 'Arduino'], 'Apple Mac mini i7': ['Arduino'], 'Apple Mac Pro Intel Xeon

```