



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»**

Подготовила:

Студентка группы ИУ5-55Б.

Очеретная С.В.

24.10.2021

Проверил:

Гапанюк Ю.Е.

Москва, 2021 г

## Вариант предметной области: 9

Класс 1: Операционная система

Класс 2: Компьютер

## Вариант запросов: Д

1. «Операционная система» и «Компьютер» связаны соотношением один-ко-многим. Выведите список всех операционных систем, у которых название хранимой записи содержит "Windows", и названия компьютеров с этими ОС.
2. «Операционная система» и «Компьютер» связаны соотношением один-ко-многим. Выведите список компьютеров со средней датой публикации операционных систем 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. «Операционная система» и «Компьютер» связаны соотношением многие-ко-многим. Выведите список всех компьютеров, у которых название начинается с буквы «А», и список их операционных систем.

## Модели:

### Операционная система:

```
class OperatingSystem:
    """Операционная система"""

    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 операционными системами:

```
class OperatingSystemComputer:
    """
    Операционные системы компьютеров
    для реализации связи многие-ко-многим
    """

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

## База данных:

### Операционные системы:

```
operating_systems = [  
    OperatingSystem(1, 'Linux', 1991, 7),  
    OperatingSystem(2, 'Windows 7', 2009, 5),  
    OperatingSystem(3, 'Windows 10', 2015, 1),  
    OperatingSystem(11, 'MacOS', 1984, 10),  
    OperatingSystem(22, 'FreeDOS', 1994, 12)  
]
```

### Компьютеры:

```
computers = [  
    Computer(1, 'Lenovo IdeaCentre 3 07ADA05'),  
    Computer(2, 'HyperPC Lumen 3070Ti i5'),  
    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_operation_systems = [  
    OperatingSystemComputer(1, 3),  
    OperatingSystemComputer(1, 2),  
    OperatingSystemComputer(1, 1),  
    OperatingSystemComputer(2, 3),  
    OperatingSystemComputer(3, 1),  
    OperatingSystemComputer(3, 22),  
    OperatingSystemComputer(4, 3),  
    OperatingSystemComputer(5, 2),  
    OperatingSystemComputer(6, 2),  
    OperatingSystemComputer(7, 1),  
    OperatingSystemComputer(8, 1),  
    OperatingSystemComputer(9, 1),  
    OperatingSystemComputer(9, 2),  
    OperatingSystemComputer(9, 3),  
    OperatingSystemComputer(9, 11),  
    OperatingSystemComputer(10, 11),  
    OperatingSystemComputer(11, 11),  
    OperatingSystemComputer(11, 12),  
    OperatingSystemComputer(12, 1),  
    OperatingSystemComputer(12, 2),  
    OperatingSystemComputer(12, 3),  
    OperatingSystemComputer(12, 11),  
    OperatingSystemComputer(12, 22),  
]
```

## Выполнение задания:

Для заданий Д-1, Д-2:

# Соединение данных один-ко-многим

```
operating_systems_join_computers = [{'operating_systems': o, 'computers': c}
    for o in operating_systems
    for c in computers
    if o.computer_id == c.id
]
```

## Выполнение задания Д-1

Выведем id, name, publication\_year таблицы "Операционная система"

При этом name != Windows

И выведем компьютеры этих ОС

```
D1 = [(x['operating_systems'].id, x['operating_systems'].name,
x['operating_systems'].publication_year, x['computers'].name)
    for x in operating_systems_join_computers
    if x['operating_systems'].name.find('Windows') != - 1
]
for x in D1:
    print(x)
```

## Принтскрин с результатом:

Задание Д-1

(2, 'Windows 7', 2009, 'iRU Home 310 298386')

(3, 'Windows 10', 2015, 'Lenovo IdeaCentre 3 07ADA05')

## Выполнение задания Д-2

Выведем имя компьютера, среднее по дате публикации ОС 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 operating_systems_join_computers:
    computer_name = os_computers_row['computers'].name
    publication_year = os_computers_row['operating_systems'].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', 1994.0)
('Dell Optiplex MT 3070-7681', 1991.0)
('Apple Mac mini i7', 1984.0)

```

### Выполнение задания Д-3

Соединение данных многие-ко-многим

```

many_to_many = [(c.name, co.computer_id, co.operation_system_id)
    for c in computers
        for co in computers_with_operation_systems
            if c.id == co.computer_id]

computers_with_operation_systems_table = [(operating_system.name,
operating_system.publication_year, computer_name)
    for computer_name, computer_id, operating_system_id in many_to_many
        for operating_system in operating_systems if operating_system.id ==
operating_system_id]

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

print(D3)

```

### Принтскрин с результатом:

```

Задание Д-3
{'Acer Nitro N50-600': ['Linux'], 'Acer Predator
PO3-600': ['Linux', 'Windows 7', 'Windows 10',
'MacOS'], 'Apple Mac mini i7': ['MacOS'], 'Apple
Mac Pro Intel Xeon 8': ['MacOS']}

```

