

**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
Вариант В-21

Выполнил:
студент группы ИУ5-53Б
Сергеев М.Ю.

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю.Е.

Москва, 2021 г.

Описание предметной области:

Существуют две сущности: Оператор и Язык программирования.

Оператор имеет следующие атрибуты: номер (id), имя (name), частота употребления (frequency) и номер связанного языка программирования один-ко-многим (lan_id).

Язык программирования имеет следующие атрибуты: номер (id), название (name) и используемость(usability). Задание сущностей и список их значений в Python:

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

    def values(self):
        return (self.id, self.name)

class Operator:
    def __init__(self, id, name, frequency, lan_id):
        self.id = id
        self.name = name
        self.frequency = frequency
        self.lan_id = lan_id
```

Список всех операторов и языков программирования:

```
#Языки программирования
languages = [Language(1, 'JavaScript', 18.4),
             Language(2, 'Java', 15.4),
             Language(3, 'C#', 13.7),
             Language(4, 'Python', 13.2),
             Language(5, 'PHP', 10.8),
             Language(6, 'C++', 5.8)]

#Операторы
operators = [Operator(1, '=', 10.5, 5),
             Operator(2, '*=', 5.4, 2),
             Operator(3, '/=', 5.1, 4),
             Operator(4, '+=', 8.9, 5),
             Operator(5, '++', 7.7, 6),
             Operator(6, '--', 3.6, 3)]
```

Задание В1:

«Язык программирования» и «Оператор» связаны соотношением один-ко-многим. Выведите список всех операторов, название которых начинается «+», и названия языков программирования, в которых они используются.

Реализация на Python:

```
one_to_many = [(lan.id, lan.name, lan.usability, op.name, op.frequency)
               for lan in languages
               for op in operators
               if op.lan_id == lan.id]

#Задание В1
print('Задание В1')
a1 = list(filter(lambda x: (str)(x[3]).startswith('+'), one_to_many))
a1 = [(el[3], el[1]) for el in a1]
print(a1)
```

Результат:

```
Задание В1
[('+=', 'PHP'), ('++', 'C++')]
```

Задание В2:

«Язык программирования» и «Оператор» связаны соотношением один-ко-многим. Выведите список языков с минимальным количеством частоты использования из всех операторов в соответствующем языке программирования, отсортированный по минимальному использованию.

Реализация на Python:

```
#Задание В2
print('Задание В2')
a2 = []
for lan in languages:
    lan_oper = list(filter(lambda x: x.lan_id == lan.id, operators))
    if len(lan_oper) > 0:
        min_frequency = min([op.frequency for op in lan_oper])
        a2.append((*lan.values(), min_frequency))

a2 = sorted(a2, key= lambda x: x[2])
print(a2)
```

Результат:

```
Задание В2
[(3, 'C#', 3.6), (4, 'Python', 5.1), (2, 'Java', 5.4), (6, 'C++', 7.7), (5, 'PHP', 8.9)]
```

Задание В3:

«Язык программирования» и «Оператор» связаны соотношением многие-ко-многим. Выведите список всех связанных языков и операторов, отсортированный по операторам, сортировка по языкам произвольная.

Для реализации связи многие-ко-многим зададим промежуточную сущность:

```
class Oper_Lang:
    def __init__(self, lan_id, oper_id):
        self.lan_id = lan_id
        self.oper_id = oper_id
```

Список связей:

```
#Операторы в языках программирования (много-ко-многим)
oper_in_lang = [Oper_Lang(2, 3),
                 Oper_Lang(1, 3),
                 Oper_Lang(4, 1),
                 Oper_Lang(5, 1),
                 Oper_Lang(6, 1),
                 Oper_Lang(2, 6),
                 Oper_Lang(6, 2),
                 Oper_Lang(2, 4),
                 Oper_Lang(1, 5),
                 Oper_Lang(3, 5)]
```

Реализация задания:

```
#Задание В3
print('Задание В3')
a3 = {}
for op in operators:
    operlan = list(filter(lambda x: x.oper_id == op.id, oper_in_lang))
    a3[op.name] = [
        l.name for l in languages
        for bl in operlan
        if bl.lan_id == l.id]
a3 = {i[0]: i[1] for i in sorted(a3.items(), key=lambda x: x[0])}
print(a3)
```

Результат:

```
Задание В3
{'*=': ['C++'], '++': ['JavaScript', 'C#'], '+=': ['Java'], '--': ['Java'], '/=': ['JavaScript', 'Java'], '=': ['Python', 'PHP', 'C++']}
```