

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

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

Курс «Базовые компоненты интернет-технологий»

Рубежный контроль №1

Выполнил:  
студент группы ИУ5-32Б:  
Ховен Ольги Александровны  
Подпись и дата:

Проверил:  
преподаватель каф. ИУ5  
Гапанюк Ю.Е.  
Подпись и дата:

Москва, 2022 г.

## Вариант 23. Б. (Синтаксическая конструкция и Язык программирования)

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список всех связанных сотрудников и отделов, отсортированный по сотрудникам, сортировка по отделам произвольная.
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список отделов с количеством сотрудников в каждом отделе, отсортированный по количеству сотрудников.
3. «Отдел» и «Сотрудник» связаны соотношением многие-ко-многим. Выведите список всех сотрудников, у которых фамилия заканчивается на «ов», и названия их отделов.

```
# используется для сортировки
from operator import itemgetter
class Language:
    """Языки программирования"""
    def __init__(self, id, name):
        self.id = id
        self.name = name
class Syntax:
    """Синтаксис"""
    def __init__(self, id, op, Language_id):
        self.id = id
        self.op = op
        self.Language_id = Language_id
class LanguageSyntax:
    """
    для реализации
    СВЯЗИ МНОГИЕ-КО-МНОГИМ
    """
    def __init__(self, Language_id, Syntax_id):
        self.Language_id = Language_id
        self.Syntax_id = Syntax_id

# Языки
Languages = [
    Language(1, 'C++'),
    Language(2, 'Python'),
    Language(3, 'C#')
]

# Синтаксис
Syntaxs = [
    Syntax(1, 'if', 1),
    Syntax(2, 'for', 3),
    Syntax(3, 'while', 2),
    Syntax(4, 'goto', 3),
    Syntax(5, 'print', 2)
]

Languages_Syntaxs = [
    LanguageSyntax(1, 1),
    LanguageSyntax(2, 5),
    LanguageSyntax(3, 4),
    LanguageSyntax(1, 3),
    LanguageSyntax(2, 1),
]
```

---

```

LanguagesSyntax(2,5),
LanguagesSyntax(3,4),
LanguagesSyntax(1,3),
LanguagesSyntax(2,1),
]

def main():
    """Основная функция"""
    # Соединение данных один-ко-многим
    one_to_many = [(s.op, l.name)
                   for s in Syntaxs
                   for l in Languages
                   if s.Language_id==l.id]
    # Соединение данных многие-ко-многим
    many_to_many_temp = [(l.name, ls.Language_id)
                          for l in Languages
                          for ls in Languages_Syntaxs
                          if l.id==ls.Language_id]
    many_to_many = [(l.name, Language name)
                    for Language_name, Language_id in many_to_many_temp
                    for l in Languages if l.id==Language_id]
    print('Задание B1')
    res_1 = sorted(one_to_many, key=itemgetter(1))
    print(res_1)
    print('\nЗадание B2')
    res_2_unsorted = []
    for l in Languages:
        l.Language = list(filter(lambda i: i[1]==l.name, one_to_many))
        Language_count=len(l.Language)
        res_2_unsorted.append((l.name, Language_count))
    res_2 = sorted(res_2_unsorted, key=itemgetter(1), reverse=True)
    print(res_2)
    print('\nЗадание B3')
    res_3 = {}
    for s in Syntaxs:
        if "f" in s.op:
            # Список операторов, в которых встречается данный символ
            l_langs = list(filter(lambda i: i[1]==s.Language_id, many_to_many))
            l_langs_names = [x for x, _ in l_langs]
            res_3[s.op] = l_langs_names

    print(res_3)
if __name__ == '__main__':
    main()

```

---

## Результат:

```

Задание B1
[('for', 'C#'), ('goto', 'C#'), ('if', 'C++'), ('while', 'Python'), ('print', 'Python')]

Задание B2
[('Python', 2), ('C#', 2), ('C++', 1)]

Задание B3
{'if': [], 'for': []}

```