

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

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

Курс «Парадигмы и конструкции языков программирования»

Отчет по рубежному контролю №1

Вариант 22, Б

**Выполнил:
студент группы ИУ5-33Б
Буров Роман**

**Проверил:
преподаватель каф. ИУ5
Гапанюк Ю. Е.**

Москва, 2025 г.

```
# Библиотеки программирования

class Library:

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

```
# Языки программирования

class Language:

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

```
# Для связи многие-ко-многим

class LibraryLanguage:

    def __init__(self, lib_id, lang_id):
        self.lib_id = lib_id
        self.lang_id = lang_id
```

```
# Данные

langs = [
    Language(1, "JavaScript"),
    Language(2, "Python"),
    Language(3, "TypeScript"),
]
```

```
# Библиотеки
```

```
libs = [
```

```
    Library(1, "React", 1),
```

```
    Library(2, "Vue", 1),
```

```
    Library(3, "NumPy", 2),
```

```
    Library(4, "Django", 2),
```

```
    Library(5, "NestJS", 3),
```

```
]
```

```
# Связи многие-ко-многим
```

```
libs_langs = [
```

```
    LibraryLanguage(1, 1),
```

```
    LibraryLanguage(1, 3),
```

```
    LibraryLanguage(2, 1),
```

```
    LibraryLanguage(3, 2),
```

```
    LibraryLanguage(4, 2),
```

```
    LibraryLanguage(5, 3),
```

```
    LibraryLanguage(5, 1),
```

```
]
```

```
def func1(one_to_many):
```

```
    return sorted(one_to_many, key=lambda x: x[1])
```

```
def func2(one_to_many, langs):
```

```
    res_b2_unsorted = []
```

```
    for lang in langs:
```

```
        libs_count = len([x for x in one_to_many if x[1] == lang.title])
```

```
if libs_count > 0:
    res_b2_unsorted.append((lang.title, libs_count))
res_b2 = sorted(res_b2_unsorted, key=lambda x: x[1], reverse=True)
return res_b2

def func3(many_to_many):
    return [(lib_name, lang_title) for lib_name, lang_title in many_to_many if
lib_name.endswith("JS")]

def main():
    one_to_many = [(lib.name, lang.title)
        for lang in langs
        for lib in libs
        if lib.lang_id == lang.id]

    many_to_many_temp = [(ll.lib_id, ll.lang_id) for ll in libs_langs]
    many_to_many = [(lib.name, lang.title)
        for lib_id, lang_id in many_to_many_temp
        for lib in libs if lib.id == lib_id
        for lang in langs if lang.id == lang_id]

    print("Задание Б1: Отсортировать библиотеки по названию языка программирования")
    for lib in func1(one_to_many):
        print(f" {lib[0]} - {lib[1]}")

    print("\nЗадание Б2: Отсортировать языки по количеству библиотек")
    for lang, count in func2(one_to_many, langs):
        print(f" {lang}: {count} библиотек")
```

```
print("\nЗадание Б3: Найти библиотеки, оканчивающиеся на 'JS'")  
  
js_libs = func3(many_to_many)  
  
for lib_name, lang_title in js_libs:  
    print(f" {lib_name} - {lang_title}")  
  
  
if __name__ == '__main__':  
    main()
```

Результат выполнения программы

```
● artorias@LAPTOP-2ESPT302:~/IU5/PCPL/RK/RK1$ python3 RK1.py
Задание Б1: Отсортировать библиотеки по названию языка программирования
    React - JavaScript
    Vue - JavaScript
    NumPy - Python
    Django - Python
    NestJS - TypeScript

Задание Б2: Отсортировать языки по количеству библиотек
    JavaScript: 2
    Python: 2
    TypeScript: 1

Задание Б3: Найти библиотеки, оканчивающиеся на 'JS'
    NestJS - TypeScript
    NestJS - JavaScript
```