

Вариант №22 «Библиотека – Язык программирования»

Вариант запросов Г.

1. «Язык программирования» и «Библиотека» связаны соотношением один-ко-многим. Выведите список всех языков программирования, у которых название начинается с буквы «J», и список их библиотек.
2. «Язык программирования» и «Библиотека» связаны соотношением один-ко-многим. Выведите список языков программирования с максимальным количеством скачиваний библиотеки данного языка, отсортированный по максимальному количеству скачиваний.
3. «Язык программирования» и «Библиотека» связаны соотношением многие-ко-многим. Выведите список всех связанных библиотек и языков программирования, отсортированный по языкам программирования, сортировка по библиотекам произвольная.

```
from operator import itemgetter

class Library:
    def __init__(self, id, name, downloads, language_id):
        self.id = id
        self.name = name
        self.downloads = downloads
        self.language_id = language_id

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

class LibraryLanguage:
    def __init__(self, library_id, language_id):
        self.library_id = library_id
        self.language_id = language_id

languages = [
    Language(1, "Python"),
```

```

    Language(2, "C/C++"),
    Language(3, "Javascript"),
    Language(4, "Java")
]

libraries = [
    Library(1, "NumPy", 16_000_000, 1),
    Library(2, "Pandas", 13_000_000, 1),
    Library(3, "SFML", 20_000, 2),
    Library(4, "React", 6_500_000, 3),
    Library(5, "Guava", 4_000_000, 4)
]

libraries_languages = [
    LibraryLanguage(1, 1),
    LibraryLanguage(1, 2),
    LibraryLanguage(2, 1),
    LibraryLanguage(2, 2),
    LibraryLanguage(3, 2),
    LibraryLanguage(4, 3),
    LibraryLanguage(5, 4)
]

def first_task(one_to_many):
    print("Задание Г1")

    result = dict()

    for language in languages:
        language_libraries = [library_name for library_name, _, _ in
list(filter(lambda x: x[2] == language.name, one_to_many))]

        if language.name.startswith("J"):
            result[language.name] = language_libraries

    print(result)

def second_task(one_to_many):
    print("Задание Г2")

    result = list()

    for language in languages:
        language_libraries = list(filter(lambda x: x[2] == language.name,
one_to_many))

        if len(language_libraries) > 0:
            result.append((language.name, max(library_downloads for _,
library_downloads, _ in language_libraries)))
        else:
            result.append((language.name, 0))

```

```

        print(sorted(result, key=itemgetter(1), reverse=True))

def third_task(many_to_many):
    print("Задание Г3")

    result = sorted(many_to_many, key=itemgetter(2))

    print(result)

def main():
    one_to_many = [(library.name, library.downloads, language.name)
                    for library in libraries
                    for language in languages
                    if library.language_id == language.id]

    many_to_many_temp = [(language.name, library_language.language_id,
                           library_language.library_id)
                          for language in languages
                          for library_language in libraries_languages
                          if language.id == library_language.language_id]

    many_to_many = [(library.name, library.downloads, language_name)
                     for language_name, _, library_id in many_to_many_temp
                     for library in libraries if library.id == library_id]

    first_task(one_to_many)
    second_task(one_to_many)
    third_task(many_to_many)

if __name__ == "__main__":
    main()

```

Результат выполнения

```

mnmnm@LAPTOPIC:~/projects/pcpl_2025$ python3 rk_1/main.py
Задание Г1
{'Javascript': ['React'], 'Java': ['Guava']}
Задание Г2
[('Python', 16000000), ('Javascript', 6500000), ('Java', 4000000), ('C/C++', 20000)]
Задание Г3
[('NumPy', 16000000, 'C/C++'), ('Pandas', 13000000, 'C/C++'), ('SFML', 20000, 'C/C++'), ('Guava', 4000000, 'Java'), ('React', 6500000, 'Java script'), ('NumPy', 16000000, 'Python'), ('Pandas', 13000000, 'Python')]

```