

Задание

1. «Книга» и «Глава» связаны соотношением один-ко-многим. Выведите список всех связанных книг и глав, отсортированный по названиям книг, сортировка по главам произвольная.
2. «Книга» и «Глава» связаны соотношением один-ко-многим. Выведите список всех связанных книг и глав, отсортированный по количеству глав в каждой книге.
3. «Книга» и «Глава» связаны соотношением многие-ко-многим. Выведите список всех книг, у которых в названии присутствует слово «стих», и список глав этих книг.

Код программы

```
from operator import itemgetter

#Описания классов
class Book:
    def __init__(self, ID, title, genre, year):
        self.ID = ID
        self.title = title
        self.genre = genre
        self.year = year

class Chapter:
    def __init__(self, ID, title, page, book_ID):
        self.ID = ID
        self.title = title
        self.page = page
        self.book_ID = book_ID

class BookChapters:
    def __init__(self, book_ID, chapter_ID):
        self.book_ID = book_ID
        self.chapter_ID = chapter_ID

#Список книг
books = [
    Book(1, "Белая книга", "Психология", 2014),
    Book(2, "Полный сборник стихов Кукушкина", "Стихотворения", 2010),
    Book(3, "Точно полный сборник стихов Кукушкина и его легендарного внука Макушкина",
    "Стихотворения", 2012),
    Book(4, "Программирование для чайников", "Учебные пособия", 2006),
    Book(5, "Водородная бомба в домашних условиях", "Учебные пособия", 1977)
]

#Список глав
chapters = [
    Chapter(1, "Жить здорово", 3, 1),
    Chapter(2, "Работа убеждений на подсознательном уровне", 20, 1),
    Chapter(3, "Меняться можно без насилия над собой", 57, 1),
    Chapter(4, "Стихотворение 'Бетономешалка'", 2, 2),
    Chapter(5, "Стихотворение 'Самогон'", 3, 2),
    Chapter(6, "Стихотворение 'Про летающую корову'", 5, 2),
    Chapter(7, "Стихотворение 'Самогон 2.0'", 1, 3),
    Chapter(8, "Вы - чайник, если..", 1, 4),
    Chapter(9, "Самодельный процессор из куска кремния и проволоки", 10, 4),
    Chapter(10, "Программируем на калькуляторе", 25, 4),
```

```

Chapter(11, "Осваиваем Логомиры 3.0", 66, 4),
Chapter(12, "Применение водородных бомб на практике", 3, 5),
Chapter(13, "Добыча водорода", 9, 5),
Chapter(14, "Поэтапная сборка бомбы", 18, 5)
]

#Список глав книг
book_chapters = [
    BookChapters(1, 1),
    BookChapters(1, 2),
    BookChapters(1, 3),
    BookChapters(2, 4),
    BookChapters(2, 5),
    BookChapters(2, 6),
    BookChapters(3, 7),
    BookChapters(4, 8),
    BookChapters(4, 9),
    BookChapters(4, 10),
    BookChapters(4, 11),
    BookChapters(5, 12),
    BookChapters(5, 13),
    BookChapters(5, 14)
]

#Основная функция
def main():
    #Связь один-ко-многим
    one_to_many = [
        (book.title, chapter.title, chapter.page)
        for book in books
        for chapter in chapters
        if chapter.book_ID == book.ID
    ]

    #Связь многие-ко-многим
    many_to_many_temp = [
        (book.title, bc.book_ID, bc.chapter_ID)
        for book in books
        for bc in book_chapters
        if bc.book_ID == book.ID
    ]

    many_to_many = [
        (chapter.title, chapter.page, title)
        for title, book_ID, chapter_ID in many_to_many_temp
        for chapter in chapters
        if chapter.ID == chapter_ID
    ]

    #Задание A1 (1->M). Вывод всех связанных глав и книг, отсортированных по названиям
    книг.
    print(" ")
    print("Задание A1")
    res_1 = sorted(one_to_many, key = itemgetter(0))
    print(res_1)

    #Задание A2 (1->M). Вывод всех связанных глав и книг, отсортированных по количеству
    глав в книге.
    print(" ")
    print("Задание A2")
    res_2 = []
    for b in books:
        chapter_count = 0
        for c in chapters:

```

```

        if c.book_ID == b.ID:
            chapter_count += 1
        res_2.append((b.title, chapter_count))
    res_2 = sorted(res_2, key = itemgetter(1), reverse = True)
    print(res_2)

#Задание A3 (М->М). Вывод всех книг, в названии которых есть слово "стих", и глав
этих книг.
print(" ")
print("Задание A3")
res_3 = {}
for b in books:
    if 'стих' in b.title:
        books_res = list(filter(lambda i: i[2] == b.title, many_to_many))
        books_res_titles = [x for x, _, _ in books_res]
        res_3[b.title] = books_res_titles
print(res_3)

if __name__ == '__main__':
    main()

```

Вывод программы

```

C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python37_64\python.exe
Задание A1
[('Белая книга', 'Жить здорово', 3), ('Белая книга', 'Работа убеждений на подсознательном уровне', 20), ('Белая книга', 'Меняться мо
жно без насилия над собой', 57), ('Водородная бомба в домашних условиях', 'Применение водородных бомб на практике', 3), ('Водородная
бомба в домашних условиях', 'Добыча водорода', 9), ('Водородная бомба в домашних условиях', 'Поэтапная сборка бомбы', 18), ('Полный
сборник стихов Кукушкина', 'Стихотворение 'Бетономешалка'', 2), ('Полный сборник стихов Кукушкина', 'Стихотворение 'Самогон'', 3),
('Полный сборник стихов Кукушкина', 'Стихотворение 'Про летающую корову'', 5), ('Программирование для чайников', 'Вы - чайник, если.
.', 1), ('Программирование для чайников', 'Самодельный процессор из куска кремния и проволоки', 10), ('Программирование для чайников
', 'Программируем на калькуляторе', 25), ('Программирование для чайников', 'Осваиваем Логомиры 3.0', 66), ('Точно полный сборник сти
хов Кукушкина и его легендарного внука Макушкина', 'Стихотворение 'Самогон 2.0'', 1)]

Задание A2
[('Программирование для чайников', 4), ('Белая книга', 3), ('Полный сборник стихов Кукушкина', 3), ('Водородная бомба в домашних усл
овиях', 3), ('Точно полный сборник стихов Кукушкина и его легендарного внука Макушкина', 1)]

Задание A3
{'Полный сборник стихов Кукушкина': ["Стихотворение 'Бетономешалка'", "Стихотворение 'Самогон'", "Стихотворение 'Про летающую корову
'"], 'Точно полный сборник стихов Кукушкина и его легендарного внука Макушкина': ["Стихотворение 'Самогон 2.0'"]}
Press any key to continue . . .

```