

Выполнил: Романеев А.Ф.

Группа: РТ5-316

Рубежный контроль №2 по дисциплине «Базовые компоненты интернет-технологий»

Вариант запросов: Б.

Вариант предметной области: 24.

Условия задания

- 1) Проведите рефакторинг текста программы рубежного контроля №1 таким образом, чтобы он был пригоден для модульного тестирования.
- 2) Для текста программы рубежного контроля №1 создайте модульные тесты с применением TDD - фреймворка (3 теста).

Текст программы

Файл main.py (измененный под условия РК2)

```
from operator import itemgetter

'''Вариант Б.
1. «Книга» и «Глава» связаны соотношением один-ко-многим.
Выведите список всех связанных глав и книг, отсортированный по главам, сортировка
по книгам произвольная.
2. «Книга» и «Глава» связаны соотношением один-ко-многим.
Выведите список книг с количеством глав в каждой книге, отсортированный по
количеству глав.
3. «Книга» и «Глава» связаны соотношением многие-ко-многим.
Выведите список всех глав, у которых названия записываются на "ми", и названия их
книг.
'''

class book():
    def __init__(self, ID, name):
        self.id = ID # Номер книги
        self.name = name # Название книги
```

```

class chapter() :
    def __init__ (self, ID, chapter_name, chapter_page, book_ID):
        self.id = ID # Номер главы
        self.name = chapter_name # Название главы
        self.page = chapter_page # Номер страницы, где находится глава
        self.book_id = book_ID # Номер книги

class bookchapter() :
    def __init__ (self, book_ID, chapter_ID):
        self.book_id = book_ID # Номер книги
        self.chapter_id = chapter_ID # Номер главы

books = [
    book(1, 'Google Android. Программирование для мобильных устройств'),
    book(2, 'Java 2. Самоучитель'),
    book(3, '1С: предприятие 8.1. Конфигурирование и администрирование'),
    book(4, 'Думай и богатей'),
    book(5, 'Возрождение бренда. Шесть принципов. Вдохните в свой бренд новую жизнь вместе с McDonalds'),
    book(6, 'Горизонт событий'),
]

chapters = [
    chapter(1, 'Компоненты Android-приложения', 43,1),
    chapter(2, 'Управление деятельностью', 249,1),
    chapter(3, 'Ресурсы, активы и локализация приложений', 349,1),
    chapter(4, 'Все, что надо знать о программах', 14,2),
    chapter(5, 'Наследование', 203, 2),
    chapter(6, 'Объекты конфигурации', 80 ,3),
    chapter(7, 'Механизм Web-сервисов', 612, 3),
    chapter(8, 'Изумительная сила секрета доказывается фактами', 3,4),
    chapter(9, 'Шесть принципов возрождения бренда', 46,5),
    chapter(10, 'Принцип 2. Восстановление соответствия бренда запросам потребителей', 68,5),
    chapter(11, 'Альфа Цефея', 2048,6),
]

chapters_of_books = [
    bookchapter(1, 1),
    bookchapter(1, 2),
    bookchapter(1, 3),
    bookchapter(2, 4),
    bookchapter(2, 5),
    bookchapter(3, 6),
    bookchapter(3, 7),
    bookchapter(4, 8),
    bookchapter(5, 9),

```



```

        if book.id == chobs.book_id
    ]

    manyToMany = [(chap.name, chap.page, book_name)
                   for book_name, book_id, chapter_id in manyToMany_temp
                   for chap in chapters if chap.id == chapter_id]

    print(task1(oneToMany))
    print(task2(oneToMany))
    print(task3(manyToMany))

if __name__ == "__main__":
    main()

```

Файл modul_test.py (файл с модульными тестами с TDD)

```

import unittest
from unittest import main
from main import task1, task2, task3

class testiki(unittest.TestCase):
    def testik1(self):
        self.assertEqual(task1(
            [('Альфа Цефея', 2048, 'Горизонт событий'),
             ('Изумительная сила секрета доказывается фактами', 3, 'Думай и богатеи'),
             ('Объекты конфигурации', 80, '1С: предприятие 8.1. Конфигурирование и администрирование'),
             ('Управление деятельностью', 249, 'Google Android. Программирование для мобильных устройств'),
             ('Все, что надо знать о программах', 14, 'Java 2. Самоучитель'),
             ('Наследование', 203, 'Java 2. Самоучитель'),
             ('Механизм Web-сервисов', 612, '1С: предприятие 8.1. Конфигурирование и администрирование'),
             ('Ресурсы, активы и локализация приложений', 349, 'Google Android. Программирование для мобильных устройств'),
             ('Компоненты Android-приложения', 43, 'Google Android. Программирование для мобильных устройств')]),
            [
                ('Альфа Цефея', 2048, 'Горизонт событий'),
                ('Все, что надо знать о программах', 14, 'Java 2. Самоучитель'),
                ('Изумительная сила секрета доказывается фактами', 3, 'Думай и богатеи'),
            ]),

```

```

        ('Компоненты Android-приложения', 43, 'Google Android. Программирование
для мобильных устройств'),
        ('Механизм Web-сервисов', 612, '1С: предприятие 8.1. Конфигурирование и
администрирование'),
        ('Наследование', 203, 'Java 2. Самоучитель'),
        ('Объекты конфигурации', 80, '1С: предприятие 8.1. Конфигурирование и
администрирование'),
        ('Ресурсы, активы и локализация приложений', 349, 'Google Android.
Программирование для мобильных устройств'),
        ('Управление деятельностью', 249, 'Google Android. Программирование
для мобильных устройств'))])

def testik2(self):
    with self.assertRaises(TypeError) as err:
        task1([1, 'str', 1.0])
    self.assertEqual("Выражение должно быть типа list[tuple[str, int, str]]",
err.exception.args[0])

def testik3(self):
    self.assertEqual(task2(
        [('Альфа Цефея', 2048, 'Горизонт событий'),
        ('Все, что надо знать о программах', 14, 'Java 2. Самоучитель'),
        ('Изумительная сила секрета доказывается фактами', 3, 'Думай и
богатеи'),
        ('Компоненты Android-приложения', 43, 'Google Android. Программирование
для мобильных устройств'),
        ('Механизм Web-сервисов', 612, '1С: предприятие 8.1. Конфигурирование и
администрирование'),
        ('Наследование', 203, 'Java 2. Самоучитель'),
        ('Объекты конфигурации', 80, '1С: предприятие 8.1. Конфигурирование и
администрирование'),
        ('Принцип 2. Восстановление соответствия бренда запросам потребителей',
68, 'Возрождение бренда. Шесть принципов. Вдохните в свой бренд новую жизнь вместе
с McDonalds'),
        ('Ресурсы, активы и локализация приложений', 349, 'Google Android.
Программирование для мобильных устройств'),
        ('Управление деятельностью', 249, 'Google Android. Программирование
для мобильных устройств'),
        ('Шесть принципов возрождения бренда', 46, 'Возрождение бренда. Шесть
принципов. Вдохните в свой бренд новую жизнь вместе с McDonalds'))]),

        [('Google Android. Программирование для мобильных устройств', 3),
        ('Java 2. Самоучитель', 2),
        ('1С: предприятие 8.1. Конфигурирование и администрирование', 2),
        ('Возрождение бренда. Шесть принципов. Вдохните в свой бренд новую
жизнь вместе с McDonalds', 2),
        ('Думай и богатеи', 1),
        ('Горизонт событий', 1)])

def testik4(self):

```

```

self.assertEqual(task3(
    [('Альфа Цефея', 'Горизонт событий'),
     ('Управление деятельностью', 'Google Android. Программирование для
мобильных устройств'),
     ('Изумительная сила секрета доказывается фактами', 'Думай и богатей'),
     ('Все, что надо знать о программах', 14, 'Java 2. Самоучитель'),
     ('Компоненты Android-приложения', 43, 'Google Android. Программирование
для мобильных устройств'),
     ('Механизм Web-сервисов', 612, '1С: предприятие 8.1. Конфигурирование и
администрирование'),
     ('Наследование', 203, 'Java 2. Самоучитель'),
     ('Объекты конфигурации', 80, '1С: предприятие 8.1. Конфигурирование и
администрирование'),
     ('Принцип 2. Восстановление соответствия бренда запросам потребителей',
68, 'Возрождение бренда. Шесть принципов. Вдохните в свой бренд новую жизнь вместе
с McDonalds'),
     ('Ресурсы, активы и локализация приложений', 349, 'Google Android.
Программирование для мобильных устройств'),
     ('Шесть принципов возрождения бренда', 46, 'Возрождение бренда. Шесть
принципов. Вдохните в свой бренд новую жизнь вместе с McDonalds')]),

    [('Управление деятельностью', 'Google Android. Программирование для
мобильных устройств'),
     ('Изумительная сила секрета доказывается фактами', 'Думай и богатей')])
if __name__ == "__main__":
    main()

```

Результаты выполнения

```
Ran 4 tests in 0.001s
```