МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

«ОРЛОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИМЕНИ И. С. ТУРГЕНЕВА»

Кафедра программной инженерии

**ОТЧЕТ**

по лабораторной работе № 3

на тему: «Объектно-ориентированное программирование»

по дисциплине: «Программирование на языке Python»

Выполнил: Евдокимов Н.А.

Институт приборостроения, автоматизации и информационных технологий

Направление: 09.03.04 «Программная инженерия»

Группа: 71-ПГ

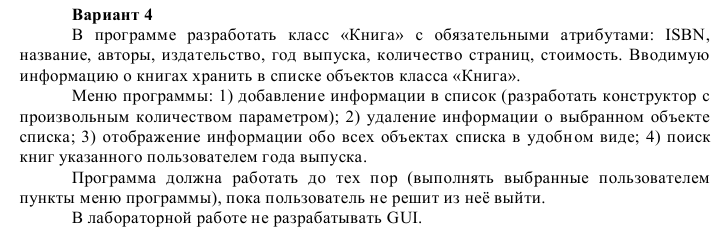
Проверила: Захарова О.В.

Отметка о зачете:

Дата: «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2019 г.

Орёл, 2019

Задание



Код

import textwrap as tw

from lab3.Book import Book

def print\_menu():

menu = """

1) Добавить элемент в список

2) Удалить элемент из списка

3) Вывести список на экран

4) Поиск по году

0) Выйти

"""

print(tw.dedent(menu))

def add\_item(target\_list):

if type(target\_list) != list:

raise ValueError

name = None

while name is None:

name = input('Введите название: ')

if len(name) <= 0:

name = None

publication\_year = None

while publication\_year is None:

try:

publication\_year = int(input('Введите год издания: '))

except ValueError:

publication\_year = None

n\_pages = None

while n\_pages is None:

try:

n\_pages = int(input('Введите кол-во страниц: '))

except ValueError:

n\_pages = None

isbn = input('Введите ISBN: ')

if len(isbn) <= 0:

isbn = None

authors = input('Введите авторов через запятую: ')

authors = authors.lstrip().rstrip().replace(' ', '').split(',')

authors = [author for author in authors if len(author) > 0]

if len(authors) <= 0:

authors = None

publisher = input('Введите издательство: ')

if len(publisher) <= 0:

publisher = None

try:

price = float(input('Введите цену: '))

except ValueError:

price = None

book = Book(name, publication\_year,

n\_pages, isbn, authors,

publisher, price)

target\_list.append(book)

def print\_book\_list(book\_list):

if type(book\_list) != list:

raise ValueError

if len(book\_list) <= 0:

return

if type(book\_list[0]) != Book:

raise ValueError

for index, book in enumerate(book\_list):

book\_str = '{})\n\tНазвание: {}\n\tГод издание: {}\n\tКол-во страниц: {}\n'\

.format(index, book.name, book.publication\_year, book.n\_pages)

if book.isbn is not None:

book\_str += '\tISBN: {}\n'.format(book.isbn)

if book.authors is not None and len(book.authors) > 0:

book\_str += '\tАвторы: {}\n'.format(book.authors)

if book.publisher is not None:

book\_str += '\tИздательство: {}\n'.format(book.publisher)

if book.price is not None:

book\_str += '\tЦена: {}'.format(book.price)

book\_str = tw.dedent(book\_str)

print(book\_str)

def remove\_item(target\_list):

if type(target\_list) != list:

raise ValueError

print\_book\_list(target\_list)

index = None

while index is None:

try:

index = int(input('Введите номер элемента: '))

except ValueError:

index = None

target\_list.pop(index)

def search\_by\_year(book\_list):

if type(book\_list) != list or type(book\_list[0]) != Book:

raise ValueError

menu = """

Год искомой книги должен быть:

1) Меньше

2) Меньше или равен

3) Равен

4) Больше или равен

5) Больше

"""

menu = tw.dedent(menu)

condition = None

while condition is None:

print(menu)

try:

condition = int(input())

except ValueError:

condition = None

year = None

while year is None:

print('Введите год')

try:

year = int(input())

except ValueError:

year = None

if year < 0:

year = None

books = []

for book in book\_list:

if condition == 1: # <

if book.publication\_year < year:

books.append(book)

elif condition == 2: # <=

if book.publication\_year <= year:

books.append(book)

elif condition == 3: # ==

if book.publication\_year == year:

books.append(book)

elif condition == 4: # >=

if book.publication\_year >= year:

books.append(book)

elif condition == 5: # >

if book.publication\_year > year:

books.append(book)

return books

def run():

should\_run = True

book\_list = []

while should\_run:

print\_menu()

try:

user\_choice = int(input())

except ValueError:

user\_choice = None

if user\_choice is None:

print("Некоректный ввод")

continue

if user\_choice == 0:

should\_run = False

elif user\_choice == 1:

add\_item(book\_list)

elif user\_choice == 2:

remove\_item(book\_list)

elif user\_choice == 3:

print\_book\_list(book\_list)

elif user\_choice == 4:

print\_book\_list(search\_by\_year(book\_list))

if \_\_name\_\_ == '\_\_main\_\_':

run()

Book.py

class Book:

def \_\_init\_\_(

self, name, publication\_year, n\_pages,

isbn=None, authors=None, publisher=None, price=None

):

self.price = price

self.publisher = publisher

self.isbn = isbn

self.n\_pages = n\_pages

self.publication\_year = publication\_year

self.name = name

if authors is None:

authors = []

self.authors = authors