**Отчет по РК № 1 по курсу**

**"Разработка Интернет-Приложений"**

Выполнила:

Студент группы

ИУ5-55Б

Смыслов М.А.

Москва, МГТУ – 2021

**Задание:**

1. «Книги» и «Библиотека» связаны соотношением один-ко-многим. Выведите список всех Книг, у которых цена больше 1000, и названия их Библиотек.
2. «Книги» и «Библиотека» связаны соотношением один-ко-многим. Выведите список Библиотек со средней ценой Книг в каждой Библиотеке, отсортированный по средней цене.
3. «Книги» и «Библиотека» связаны соотношением многие-ко-многим. Выведите список всех Библиотек, у которых название начинается с буквы «А», и список в них Книг.

|  |  |  |
| --- | --- | --- |
| **№ варианта** | **Класс 1** | **Класс 2** |
| **13** | **Книга** | **Библиотека** |

**Текст программы:**

# используется для сортировки

from operator import itemgetter

class Book:

    """Книга"""

    def \_\_init\_\_(self, id, name, cost, lib\_id):

        self.id = id

        self.name = name

        self.cost = cost

        self.lib\_id = lib\_id

class Library:

    """Библиотека"""

    def \_\_init\_\_(self, id, name):

        self.id = id

        self.name = name

class BookLibrary:

    """

    'Книги библиотеки' для реализации

    связи многие-ко-многим

    """

    def \_\_init\_\_(self, lib\_id, book\_id):

        self.lib\_id = lib\_id

        self.book\_id = book\_id

# Библиотеки

libs = [

    Library(1, 'Центральная библиотека'),

    Library(2, 'Пушкинская'),

    Library(3, 'Московская'),

    Library(11, 'Абрамовская'),

    Library(22, 'Аппель'),

    Library(33, 'Универсальная'),

]

# Книги

books = [

    Book(1, 'Белый бим черное ухо', 2500, 1),

    Book(2, 'Искра жизни', 1000, 2),

    Book(3, 'Общество потребления', 500, 3),

    Book(4, 'Искусство любить', 1100, 3),

    Book(5, 'О дивный новый мир', 900, 3),

]

libs\_books = [

    BookLibrary(1,1),

    BookLibrary(2,2),

    BookLibrary(3,3),

    BookLibrary(3,4),

    BookLibrary(3,5),

    BookLibrary(11,1),

    BookLibrary(22,2),

    BookLibrary(33,1),

    BookLibrary(33,2),

    BookLibrary(33,5),

]

def main():

    """Основная функция"""

    # Соединение данных один-ко-многим

    one\_to\_many = [(b.name, b.cost, l.name)

        for l in libs

        for b in books

        if b.lib\_id==l.id]

    # Соединение данных многие-ко-многим

    many\_to\_many\_temp = [(l.name, lb.lib\_id, lb.book\_id)

        for l in libs

        for lb in libs\_books

        if l.id==lb.lib\_id]

    many\_to\_many = [(b.name, b.cost, lib\_name)

        for lib\_name, lib\_id, book\_id in many\_to\_many\_temp

        for b in books if b.id==book\_id]

    print('Задание Д1')

    res\_11 = [(b.name, b.cost, l.name)

        for l in libs

        for b in books

        if b.lib\_id==l.id and b.cost>1000]

    print(res\_11)

    print('\nЗадание Д2')

    res\_12\_unsorted = []

    # Перебираем все библиотеки

    for l in libs:

        # Список книг библиотеки

        b\_lib = list(filter(lambda i: i[2]==l.name, one\_to\_many))

        # Если библиотека не пустая

        if len(b\_lib) > 0:

            # Цены книг

            b\_costs = [cost for \_,cost,\_ in b\_lib]

            # Средняя цена книг

            b\_costs\_mean = sum(b\_costs) / len(b\_costs)

            res\_12\_unsorted.append((l.name, b\_costs\_mean))

    # Сортировка по средней зарплате

    res\_12 = sorted(res\_12\_unsorted, key=itemgetter(1), reverse=True)

    print(res\_12)

    print('\nЗадание Д3')

    res\_13 = {}

    # Перебираем все библиотеки

    for l in libs:

        if l.name.lower().startswith('а'):

            # Список книг библиотеки

            books\_of\_l = list(filter(lambda i: i[2]==l.name, many\_to\_many))

            # Только названия книг

            books\_names = [x for x,\_,\_ in books\_of\_l]

            # Добавляем результат в словарь

            # ключ - библиотека, значение - список названий

            res\_13[l.name] = books\_names

    print(res\_13)

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

    main()

**Результат работы:**

