

**Московский государственный технический университет
им. Н. Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5. Курс «Разработка интернет-приложений»

Рубежный контроль №1

Вариант А24.

Выполнил:

Студент группы ИУ5-71Б

Бабакулыев Ф.А.

Проверил:

Гапанюк Ю. Е.

Дата: 21.10.2021

Дата:

Подпись:

Подпись:

Москва, 2021 г.

Задание варианта:

Вариант запросов: А

Вариант предметной области: 17

Описание классов предметной области:

1. Класс «Дирижер», содержащий поля:
 - ID записи о дирижере
 - ФИО дирижера
 - Количество произведений, которые оркестр исполняет под руководством этого дирижера (количественный признак)
 - ID записи об оркестре
2. Класс «Оркестр», содержащий поля:
 - ID записи об оркестре
 - Название оркестра
3. Класс для реализации связи многие-ко-многим «Дирижеры оркестра», содержащий поля:
 - ID записи о дирижере
 - ID записи об оркестре

Запросы по варианту:

1. «Глава» и «Книга» связаны соотношением один-ко-многим. Вывести список всех связанных дирижеров и оркестров, отсортированный по оркестрам, сортировка по дирижерам произвольная.
2. «Глава» и «Книга» связаны соотношением один-ко-многим. Вывести список оркестров с суммарным количеством произведений, исполняемых под руководством каждого из дирижеров, отсортированный по суммарному количеству произведений.
3. «Глава» и «Книга» связаны соотношением многие-ко-многим. Вывести список всех оркестров, у которых в названии присутствует слово «оркестр», и список руководящих ими дирижеров.

Листинг программы:

используется для сортировки

from operator import itemgetter

class Chp:

"""Глава"""

def __init__(self, id, name, page, number):

self.id = id

self.name = name

self.page = page

self.number = number

class Book:

"""Книга"""

def __init__(self, name, emp_id):

self.id = id

self.emp_id = emp_id

class ChpofBook:

def __init__(self, id, chb_id):

#self.self = self

self.id = id

self.chb_id = chb_id

Глава

Chps = [

Chp(1, 'Начало', 2,1),

```
    Chr(2, 'Заключение', 50,6),  
    Chr(3, 'Начало', 2,1),  
    Chr(4, 'Начало', 2,1),  
    Chr(5, 'Начало', 2,1),  
    Chr(6, 'Начало', 2,1),  
    Chr(3, 'Местоположение', 18,2),  
    Chr(5, 'Ведение', 1,0),  
    Chr(1, 'История', 11,2),  
]
```

Книги

```
Books = [  
    Book(1, 'Пхеньян'),  
    Book(2, 'Программирование с нуля'),  
    Book(3, 'Америка'),  
    Book(4, 'Автоматизация'),  
    Book(5, 'C#'),  
    Book(6, 'Навальный'),  
]
```

```
Chps_Books = [  
    ChpofBook(1,1),  
    ChpofBook(2,2),  
    ChpofBook(3,3),  
    ChpofBook(3,4),  
    ChpofBook(3,5),  
    ChpofBook(11,1),
```

```

    ChpofBook(22,2),
    ChpofBook(33,3),
    ChpofBook(33,4),
    ChpofBook(33,5),
]

def main():
    """Основная функция"""

    # Соединение данных один-ко-многим
    one_to_many = [(e.name, e.page, d.name)
                    for d in Chps
                    for e in Books
                    if e.emp_id==d.id]

    # Соединение данных многие-ко-многим
    many_to_many_temp = [(d.name, ed.id, ed.chb_id)
                           for d in Chps
                           for ed in Chps_Books
                           if d.id==ed.id]

    many_to_many = [(e.name, e.page, Chp_name)
                     for dep_name, dep_id, Book_id in many_to_many_temp
                     for e in Books if e.id==Book_id]

    print('Задание A1')
    res_11 = sorted(one_to_many, key=itemgetter(2))
    print(res_11)

```

```
print('\nЗадание A2')

res_12_unsorted = []

# Перебираем все отделы

for d in Books:

    d_Books = list(filter(lambda i: i[2]==d.name, one_to_many))

    if len(d_Books) > 0:

        d_prices = [price for _,price,_ in d_Books]

        d_price_sum = sum(d_sals)

        res_12_unsorted.append((d.name, d_price_sum))

res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)

print(res_12)
```

```
print('\nЗадание A3')

res_13 = {}

#

for d in Chps:

    if 'отдел' in d.name:

        d_Books = list(filter(lambda i: i[2]==d.name, many_to_many))

        d_Books_names = [x for x,_,_ in d_Books]

        # Добавляем результат в словарь

        # ключ - отдел, значение - список фамилий

        res_13[d.name] = d_Books_names


print(res_13)

if __name__ == '__main__':

    main()
```

Результаты выполнения:

Задание A1

```
{ 'Америка': [ ('Начало', 2), ('История', 13), ('Местоположение', 18)] }  
{ 'Америка': [ ('Начало', 2), ('История', 13), ('Местоположение', 18)],  
  'Автоматизация': [ ('Начало', 2)] }
```

Задание A2

```
[ ('Начало', 2), ('Начало', 2)]  
[ ('Начало', 2), ('Начало', 2), ('Начало', 2), ('Начало', 2),  
  ('Начало', 2), ('Начало', 2)]  
[ ('История', 13), ('Начало', 2), ('Начало', 2), ('Начало', 2),  
  ('Начало', 2), ('Начало', 2), ('Начало', 2)]  
[ ('История', 13), ('История', 13), ('Начало', 2), ('Начало', 2),  
  [ ('Местоположение', 18), ('История', 13), ('История', 13), ('История',  
    13), ('Начало', 2), ('Начало', 2), ('Начало', 2), ('Начало', 2),  
    ('Начало', 2), ('Начало', 2)]  
[ ('Закключение', 50), ('Местоположение', 18), ('История', 13),  
  ('История', 13), ('История', 13), ('Начало', 2), ('Начало', 2),  
  ('Начало', 2), ('Начало', 2), ('Начало', 2), ('Начало', 2)]  
[ ('Закключение', 50), ('Местоположение', 18), ('История', 13),  
  ('История', 13), ('История', 13), ('Начало', 2), ('Начало', 2),
```

Задание A3

```
{ 'Ведение': ['Америка'] }  
{ 'Ведение': ['Америка'], 'Закключение': ['Программирование с нуля'] }  
{ 'Ведение': ['Америка'], 'Закключение': ['Программирование с нуля'],  
  'История': ['Пхеньян', 'Программирование с нуля', 'Америка'] }  
{ 'Ведение': ['Америка'], 'Закключение': ['Программирование с нуля'],  
  'История': ['Пхеньян', 'Программирование с нуля', 'Америка'] }  
{ 'Ведение': ['Америка'], 'Закключение': ['Программирование с нуля'],  
  'История': ['Пхеньян', 'Программирование с нуля', 'Америка'] }  
{ 'Ведение': ['Америка'], 'Закключение': ['Программирование с нуля'],  
  'История': ['Пхеньян', 'Программирование с нуля', 'Америка'],  
  'Местоположение': ['Америка'], 'Начало': ['Пхеньян', 'Программирование  
с нуля', 'Америка', 'Америка', 'Америка'] }  
{ 'Ведение': ['Америка'], 'Закключение': ['Программирование с нуля'],  
  'История': ['Пхеньян', 'Программирование с нуля', 'Америка'],
```