

МГТУ им. Н.Э. Баумана

Отчёт по рубежному контролю №1
по курсу «Базовые компоненты и интернет-технологии»
Вариант Б-14.

Руководитель
Гапанюк Ю.Е.
29.10.2022

Студент группы ИУ5-32Б
Кузьмин Я.А.
29.10.2022

2022 г.

Полученное задание:

Необходимо создать два класса данных в соответствии с Вашим вариантом предметной области, которые связаны отношениями один-ко-многим и многие-ко-многим.

Необходимо создать списки объектов классов, содержащих тестовые данные (3-5 записей), таким образом, чтобы первичные и вторичные ключи соответствующих записей были связаны по идентификаторам.

Необходимо разработать запросы в соответствии с Вашим вариантом.

Предметная область: класс_1 – CD-диск, класс_2 – CD-библиотека, вариант запросов: Б.

Запросы:

1. «CD-диск» и «CD-библиотека» связаны соотношением один-ко-многим. Выведите список всех связанных дисков и библиотек, отсортированный по именам дисков, сортировка по библиотекам произвольная.

2. «CD-диск» и «CD-библиотека» связаны соотношением один-ко-многим. Выведите список библиотек со средним годом издания дисков в каждой библиотеке, отсортированный по среднему году издания дисков.

3. «CD-диск» и «CD-библиотека» связаны соотношением многие-ко-многим. Выведите список всех дисков, у которых заголовок заканчивается на «и», и названия их библиотек.

Листинг программы (также можно посмотреть в [GitHub-репозитории](#)):

```
14 # используется для сортировки
15 from operator import itemgetter
16
17 class Disk:
18     """CD-диск"""
19     def __init__(self, id, title, year, lib_id):
20         self.id = id
21         self.title = title
22         self.year = year
23         self.lib_id = lib_id
24
25 class Lib:
26     """CD-библиотека"""
27     def __init__(self, id, name):
28         self.id = id
29         self.name = name
30
31 class DiskLib:
32     """
33     'CD-диски в библиотеке' для реализации
34     связи многие-ко-многим
35     """
36     def __init__(self, disk_id, lib_id):
37         self.disk_id = disk_id
38         self.lib_id = lib_id
```

```

40 # CD-диски
41 disks = [
42     Disk(1, "Финал ЛЧ: Челси - Ман Сити", 2020, 1),
43     Disk(2, "Кузьмины свадьба", 2002, 2),
44     Disk(3, "Цой концерт Мск", 1986, 2),
45     Disk(4, "Смешарики новые серии", 2009, 2),
46     Disk(5, "Лекция 3 БКИТ МГТУ ИУ5", 2022, 3),
47 ]
48
49 # CD-библиотеки
50 libs = [
51     Lib(1, "Государственная"),
52     Lib(2, "Частная"),
53     Lib(3, "Бауманская"),
54 ]
55
56 disks_libs = [
57     DiskLib(1, 1),
58     DiskLib(2, 2),
59     DiskLib(3, 2),
60     DiskLib(4, 2),
61     DiskLib(5, 3),
62 ]

```

```

64 def main():
65     """Основная функция"""
66
67     # Соединение данных один-ко-многим
68     one_to_many = [(d.title, d.year, l.name)
69                    for d in disks
70                    for l in libs
71                    if d.lib_id == l.id]
72
73     # Соединение данных многие-ко-многим
74     many_to_many_temp = [(l.name, dl.lib_id, dl.disk_id)
75                           for l in libs
76                           for dl in disks_libs
77                           if l.id == dl.lib_id]
78
79     many_to_many = [(d.title, d.year, lib_name)
80                     for lib_name, lib_id, disk_id in many_to_many_temp
81                     for d in disks if d.id == disk_id]
82
83     print("Задание 51")
84     res_11 = sorted(one_to_many, key = itemgetter(0))
85     print(res_11)
86

```

```

87     print("\nЗадание 52")
88     res_12_unsorted = []
89
90     # Перебираем все библиотеки
91     for l in libs:
92         # Список дисков библиотеки
93         l_disks = list(filter(lambda i: i[2] == l.name, one_to_many))
94         # Если библиотека не пустая
95         if len(l_disks) > 0:
96             # Год диска библиотеки
97             l_years = [year for _, year, _ in l_disks]
98             # Сумма годов дисков библиотеки
99             l_years_sum = sum(l_years)
100             res_12_unsorted.append((l.name, int(l_years_sum / len(l_disks))))
101
102     # Сортировка по суммарному году
103     res_12 = sorted(res_12_unsorted, key = itemgetter(1), reverse = False)
104     print(res_12)
105
106     print("\nЗадание 53")
107     res_13 = []
108     # Перебираем все библиотеки
109     for d in disks:
110         if d.title[-1] != 'и':
111             continue
112         for l in libs:
113             if l.id == d.lib_id:
114                 res_13.append((d.title, l.name))
115                 break
116
117     print(res_13)

```

```

119 if __name__ == "__main__":
120     main()
121

```

Результаты выполнения:

```
yarik_tri@LAPTOP-MNOK6D6C:~/yarik/bmstu/bcit/BCIT-3-term$ python3 solution.py
Задание Б1
[('Кузьмины свадьба', 2002, 'Частная'), ('Лекция 3 БКИТ МГТУ ИУ5', 2022, 'Бауманская'), ('Смешарики новые серии', 2009, 'Частная'), ('Финал ЛЧ: Челси - Ман Сити', 2020, 'Государственная'), ('Цой концерт Мск', 1986, 'Частная')]

Задание Б2
[('Частная', 1999), ('Государственная', 2020), ('Бауманская', 2022)]

Задание Б3
[('Финал ЛЧ: Челси - Ман Сити', 'Государственная'), ('Смешарики новые серии', 'Частная')]
```