

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

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Рубежный контроль №1

Выполнил:		Проверил:
студент группы ИУ5-32Б		преподаватель каф. ИУ5
Панов Г. Д.		Гапанюк Ю. Е.
Подпись и дата		Подпись и дата

Москва, 2021 г.

Задание, Вариант Б:

1. «Библиотека CD-дисков» и «CD-диск» связаны соотношением один-ко-многим. Выведите список всех связанных CD-дисков и библиотек CD-дисков, отсортированный по CD-дискам, сортировка по библиотекам произвольная.
2. «Библиотека CD-дисков» и «CD-диск» связаны соотношением один-ко-многим. Выведите список библиотек CD-дисков с количеством CD-дисков, в каждой библиотеке, отсортированный по количеству CD-дисков.
3. «Библиотека CD-дисков» и «CD-диск» связаны соотношением многие-ко-многим. Выведите список всех CD-дисков, у которых название заканчивается на «ов», и названия библиотек, в которых есть эти CD-диски.

Листинг программы:

```
# используется для сортировки
from operator import itemgetter
# используется для разбиения названия диска по словам
import re

class Emp:
    """CD-диск"""

    def __init__(self, id, fio, sal, dep_id):
        self.id = id          # id записи о диске
        self.fio = fio        # название диска
        self.sal = sal        # цена диска
        self.dep_id = dep_id  # id записи о библиотеке

class Dep:
    """Библиотека CD-дисков"""

    def __init__(self, id, name):
        self.id = id          # id записи о библиотеке
        self.name = name      # наименование библиотеки

class EmpDep:
    """
    'Диски в библиотеке' для реализации
    связи многие-ко-многим
    """

    def __init__(self, dep_id, emp_id):
        self.dep_id = dep_id  # id записи о библиотеке
        self.emp_id = emp_id  # id записи о диске

# Библиотеки CD-дисков
deps = [
    Dep(1, 'библиотека фильмов'),
```

```

Dep(2, 'архивная библиотека документов'),
Dep(3, 'библиотека музыки'),

Dep(11, 'библиотека (другая) фильмов'),
Dep(22, 'архивная (другая) библиотека документов'),
Dep(33, 'библиотека (другая) музыки'),
]

# CD-диски
emps = [
    Emp(1, 'Земля вампиров', 2000, 1),
    Emp(2, 'Виндоус 7', 6000, 2),
    Emp(3, 'Пакет Office', 7000, 2),
    Emp(4, 'Альбом AC/DC', 2500, 3),
    Emp(5, 'Самоучитель по ловле кальмаров', 300, 2),
]

# CD-диски в библиотеках (связь многие-ко-многим)
emps_deps = [
    EmpDep(1, 1),
    EmpDep(2, 2),
    EmpDep(2, 3),
    EmpDep(3, 4),
    EmpDep(2, 5),

    EmpDep(11, 1),
    EmpDep(22, 2),
    EmpDep(22, 3),
    EmpDep(22, 5),
    EmpDep(33, 4),
]

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

    # Соединение данных один-ко-многим
    one_to_many = [(e.fio, e.sal, d.name)
                    for d in deps
                    for e in emps
                    if e.dep_id == d.id]

    # Соединение данных многие-ко-многим
    many_to_many_temp = [(d.name, ed.dep_id, ed.emp_id)
                           for d in deps
                           for ed in emps_deps
                           if d.id == ed.dep_id]

    many_to_many = [(e.fio, e.sal, dep_name)
                     for dep_name, dep_id, emp_id in many_to_many_temp
                     for e in emps if e.id == emp_id]

    print('Задание Б1')
    res_11 = sorted(one_to_many, key=itemgetter(0))
    print(res_11)

    print('\nЗадание Б2')
    res_12_unsorted = []
    # Перебираем все библиотеки
    for d in deps:
        # Список библиотек

```

```

    d_emps = list(filter(lambda i: i[2] == d.name, one_to_many))
    count = sum(isinstance(x, tuple) for x in d_emps)
    # Если библиотека не пустая
    if len(d_emps) > 0:
        res_12_unsorted.append((d.name, count))

# Сортировка по количеству дисков
res_12 = sorted(res_12_unsorted, key=itemgetter(1), reverse=True)
print(res_12)

print('\nЗадание Б3')
res_13 = {}
# Перебираем все диски
for e in emps:
    # проверяем оканчивается ли CD-диск на 'ов'
    if 'ов' in str(re.split('; |, |:|-', e.fio)):
        # Список библиотек данного диска
        d_emps = list(filter(lambda i: i[0] == e.fio, many_to_many))
        # Только названия библиотек
        demps_names = [x[2] for x in d_emps]
        # Добавляем результат в словарь
        # ключ - CD-диск, значение - список библиотек, в которых есть
этот CD-диск
        res_13[e.fio] = demps_names

print(res_13)

if __name__ == '__main__':
    main()

```

Пример работы:

Задание Б1

```
[('Альбом AC/DC', 2500, 'библиотека музыки'), ('Виндоус 7', 6000, 'архивная библиотека документов'), ('Земля вампиров', 2000, 'библиотека фильмов'), ('Пакет Office', 7000, 'архивная библиотека документов'), ('Самоучитель по ловле кальмаров', 300, 'архивная библиотека документов')]
```

Задание Б2

```
[('архивная библиотека документов', 3), ('библиотека фильмов', 1), ('библиотека музыки', 1)]
```

Задание Б3

```
{'Земля вампиров': ['библиотека фильмов', 'библиотека (другая) фильмов'], 'Самоучитель по ловле кальмаров': ['архивная библиотека документов', 'архивная (другая) библиотека документов']}
```