

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

Факультет ИУ  
Кафедра ИУ5

Курс «Основы информатики»

Отчет по Рубежному контролю №1  
Вариант В 13

Выполнил:  
студент группы ИУ5-23Б:

Селиханович Д.С.  
Подпись и дата:

Проверил:  
преподаватель каф.

\_\_\_\_\_  
Подпись и дата:

Москва, 2024 г.

## Постановка задачи

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

Пример классов данных для предметной области Сотрудник-Отдел:

1. Класс «Сотрудник», содержащий поля:

- ID записи о сотруднике;
- Фамилия сотрудника;
- Зарплата (количественный признак);

ID записи об отделе. (для реализации связи один-ко-многим)

2. Класс «Отдел», содержащий поля:

- ID записи об отделе;
- Наименование отдела.

3. (Для реализации связи многие-ко-многим) Класс «Сотрудники отдела», содержащий поля:

- ID записи о сотруднике;
- ID записи об отделе.

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

3) Необходимо разработать запросы в соответствии с Вашим вариантом. Запросы сформулированы в терминах классов «Сотрудник» и «Отдел», которые используются в примере. Вам нужно перенести эти требования в Ваш вариант предметной области. При разработке запросов необходимо по возможности использовать функциональные возможности языка Python (list/dict comprehensions, функции высших порядков).

Для реализации запроса №2 введите в класс, находящийся на стороне связи «много», произвольный количественный признак, например, «зарплата сотрудника».

Результатом рубежного контроля является документ в формате PDF, который содержит текст программы и результаты ее выполнения.

### Вариант В.

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список всех сотрудников, у которых фамилия начинается с буквы «А», и названия их отделов.
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список отделов с минимальной зарплатой сотрудников в каждом отделе, отсортированный по минимальной зарплате.
3. «Отдел» и «Сотрудник» связаны соотношением многие-ко-многим. Выведите список всех связанных сотрудников и отделов, отсортированный по сотрудникам, сортировка по отделам произвольная.

#### 4. Таблица 1. Варианты предметной области

№ варианта	Класс 1	Класс 2
13	Книга	Библиотека

## Текст программы

```
class Book:
    """Книга"""
    def __init__(self, id, title, author, year, lib_id):
        self.id = id
        self.title = title
        self.author = author
        self.year = year
        self.lib_id = lib_id

class Library:
    """Библиотека"""
    def __init__(self, id, name):
        self.id = id
        self.name = name

class BookLib:
    """
    'Книги в библиотеке' для реализации
    СВЯЗИ МНОГИЕ-КО-МНОГИМ
    """
    def __init__(self, lib_id, book_id, count):
        self.lib_id = lib_id
        self.book_id = book_id
        self.count = count

# Библиотеки
libs = [
    Library(1, 'Главная библиотека'),
    Library(2, 'Университетская библиотека'),
    Library(3, 'Городская библиотека'),
    Library(4, 'Детская библиотека'),
    Library(5, 'Научная библиотека'),
    Library(6, 'Историческая библиотека'),
]

# Книги
books = [
    Book(1, 'Война и мир', 'Лев Толстой', 1865, 1),
    Book(2, 'Анна Каренина', 'Лев Толстой', 1877, 1),
    Book(3, 'Преступление и наказание', 'Федор Достоевский', 1866, 2),
    Book(4, 'Братья Карамазовы', 'Федор Достоевский', 1880, 2),
    Book(5, '1984', 'Джордж Оруэлл', 1949, 3),
    Book(6, 'Скотный двор', 'Джордж Оруэлл', 1945, 3),
    Book(7, 'Мастер и Маргарита', 'Михаил Булгаков', 1939, 4),
    Book(8, 'Собачье сердце', 'Михаил Булгаков', 1925, 5),
    Book(9, 'Доктор Живаго', 'Борис Пастернак', 1957, 6),
    Book(10, 'Лолита', 'Владимир Набоков', 1955, 6),
]
```

```
]
```

```
# Связь книг и библиотек
```

```
books_libs = [  
    BookLib(1, 1, 5),  
    BookLib(1, 2, 3),  
    BookLib(2, 3, 2),  
    BookLib(2, 4, 1),  
    BookLib(3, 5, 1),  
    BookLib(3, 6, 2),  
    BookLib(4, 7, 1),  
    BookLib(5, 8, 3),  
    BookLib(6, 9, 2),  
    BookLib(6, 10, 1),  
    BookLib(2, 1, 9),  
    BookLib(3, 2, 8),  
    BookLib(4, 3, 7),  
    BookLib(5, 4, 6),  
    BookLib(6, 5, 5),  
    BookLib(6, 6, 4),  
    BookLib(1, 7, 9),  
    BookLib(3, 8, 3),  
    BookLib(2, 9, 3),  
    BookLib(1, 10, 10),  
]
```

```
def main():
```

```
    """Основная функция"""
```

```
    # Соединение данных один-ко-многим
```

```
    one_to_many = [(b.title, b.author, b.year, 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 = [(lib.name, bl.lib_id, bl.book_id)  
    for lib in libs  
    for bl in books_libs  
    if lib.id == bl.lib_id]
```

```
    many_to_many = [(book.title, book.author, book.year, lib_name)  
    for lib_name, lib_id, book_id in many_to_many_temp  
    for book in books if book.id == book_id]
```

```
    #B1
```

```
    #«Библиотека» и «Книга» связаны соотношением один-ко-многим.
```

```
    #Выведите список всех книг, у которых название начинается с буквы «А»,  
    и названия их библиотек.
```

```
print('Задание B1')
result = [item for item in one_to_many if item[0].startswith('C')]
for i in result:
    print(i[0],i[2])
```

#B2

#«Библиотека» и «Книга» связаны соотношением один-ко-многим.

#Выведите список отделов с минимальной годом создания книги в каждой библиотеке,

#отсортированный по минимальному году.

```
print('\nЗадание B2')
min_years = {}
for title, author, year, lib in one_to_many:
    if lib not in min_years or year < min_years[lib]:
        min_years[lib] = year
sorted_libs = sorted(min_years.items(), key=lambda x: x[1])
for lib, year in sorted_libs:
    print(f"Библиотека: {lib}, Год самой старой книги: {year}")
```

#B3

#«Библиотека» и «Книга» связаны соотношением многие-ко-многим.

#Выведите список всех связанных книг и библиотек, отсортированный по книгам,

#сортировка по библиотекам произвольная.

```
print('\nЗадание B3')
sorted_books = sorted(many_to_many, key=lambda x: (x[0], x[1]))
for book in sorted_books:
    print(book[0],book[3])
```

```
if __name__ == '__main__':
    main()
```

## Анализ результатов

Задание В1

Скотный двор 1945

Собачье сердце 1925

Задание В2

Библиотека: Главная библиотека, Год самой старой книги: 1865

Библиотека: Университетская библиотека, Год самой старой книги: 1866

Библиотека: Научная библиотека, Год самой старой книги: 1925

Библиотека: Детская библиотека, Год самой старой книги: 1939

Библиотека: Городская библиотека, Год самой старой книги: 1945

Библиотека: Историческая библиотека, Год самой старой книги: 1955

Задание В3

1984 Городская библиотека

1984 Историческая библиотека

Анна Каренина Главная библиотека

Анна Каренина Городская библиотека

Братья Карамазовы Университетская библиотека

Братья Карамазовы Научная библиотека

Война и мир Главная библиотека

Война и мир Университетская библиотека

Доктор Живаго Университетская библиотека

Доктор Живаго Историческая библиотека

Лолита Главная библиотека

Лолита Историческая библиотека

Мастер и Маргарита Главная библиотека

Мастер и Маргарита Детская библиотека

Преступление и наказание Университетская библиотека

Преступление и наказание Детская библиотека

Скотный двор Городская библиотека

Скотный двор Историческая библиотека

Собачье сердце Городская библиотека

Собачье сердце Научная библиотека