

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

Факультет ИУ
Кафедра ИУ5

Курс «Основы информатики»

Отчет по Рубежному контролю №1
Вариант В 11

Выполнил:
студент группы ИУ5-23Б:

Османов З. Ш.
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
15	Программа	Компьютер

Код программы:

```
from operator import itemgetter

class Program:
    def __init__(self, program_id, name, release_year, computer_id):
        self.id = program_id # ID записи программы
        self.name = name # Название программы
        self.release_year = release_year # Год выпуска программы
        self.computer_id = computer_id # ID компьютера, на котором установлена
        программа

class Computer:
    def __init__(self, computer_id, model, owner):
        self.id = computer_id # ID записи о компьютере
        self.model = model # Модель компьютера
        self.owner = owner # Владелец компьютера

class InstalledProgram:
    def __init__(self, program_id, computer_id):
        self.program_id = program_id # ID записи программы
        self.computer_id = computer_id # ID записи компьютера

computers = [
    Computer(1, "Asus", "Aleksey"),
    Computer(2, "MacBook Pro", "Sasha"),
    Computer(3, "Lenovo", "Michael"),
    Computer(4, "Dell XPS", "Alice"),
    Computer(5, "MacBook Air", "Bob"),
    Computer(6, "HP ", "Charlie"),
]

programs = [
    Program(1, "Photoshop", 1988, 1),
    Program(2, "Visual Studio", 1997, 2),
    Program(3, "Zoom", 2012, 3),
    Program(4, "Slack", 2013, 1),
    Program(5, "PyCharm", 2010, 2),
    Program(6, "Microsoft Word", 1983, 4),
    Program(7, "Blender", 1998, 5),
    Program(8, "AutoCAD", 1982, 6),
    Program(9, "Notepad++", 2003, 3),
    Program(10, "Firefox", 2002, 4)
]

installed_programs = [
    InstalledProgram(1, 1),
    InstalledProgram(2, 2),
    InstalledProgram(3, 3),
    InstalledProgram(4, 1),
    InstalledProgram(5, 2),
    InstalledProgram(2, 3),
    InstalledProgram(5, 1),
    InstalledProgram(6, 4),
    InstalledProgram(7, 5),
    InstalledProgram(8, 6),
    InstalledProgram(9, 3),
    InstalledProgram(10, 4),
```

```

        InstalledProgram(1, 5),
        InstalledProgram(6, 6)
    ]

# Соединение один-ко-многим
one_to_many = [(p.name, p.release_year, c.model, c.owner)
                for c in computers
                for p in programs
                if p.computer_id == c.id]
print(one_to_many)

# Соединение данных многие-ко-многим
many_to_many_temp = [(c.model, ip.computer_id, ip.program_id)
                      for c in computers
                      for ip in installed_programs
                      if c.id == ip.computer_id]

many_to_many = [(p.name, p.release_year, computer_model)
                 for computer_model, computer_id, program_id in many_to_many_temp
                 for p in programs if p.id == program_id]

# Задание 1: Компьютеры, владельцы которых начинаются с "А", и установленные программы
print('Задание 1')

result_1 = [(program_name, computer_model)
             for program_name, _, computer_model, owner in one_to_many
             if owner[0] == 'А']
for program_name, computer_model in result_1:
    print(f"Программа: {program_name}, Компьютер: {computer_model}")

# Задание 2: Найти программы с минимальным годом выпуска для каждого компьютера,
# отсортировать по году
print('\nЗадание 2')
result_2_unsorted = []

for c in computers:
    print(c.model)
    c_programs = [program_relise for _, program_relise, computer_model, _
                  in one_to_many
                  if c.model == computer_model]
    if len(c_programs) > 0:
        min_year = min(c_programs)
        result_2_unsorted.append((c.model, min_year))
result_2 = sorted(result_2_unsorted, key = itemgetter(1))
for computer_model, min_year in result_2:
    print(f"Компьютер: {computer_model}, Минимальный год выпуска программы: {min_year}")

# Задание 3: Все компьютеры и установленные на них программы (многие-ко-многим)
print('\nЗадание 3')
# Сортируем по названию программ
result_3 = sorted(many_to_many, key=itemgetter(0))
for program_name, release_year, computer_model in result_3:
    print(f"Программа: {program_name}, Год выпуска: {release_year}, Компьютер: {computer_model}")

```

Результаты выполнения программы:

Задание 1

Программа: Photoshop, Компьютер: Asus

Программа: Slack, Компьютер: Asus

Программа: Microsoft Word, Компьютер: Dell XPS

Программа: Firefox, Компьютер: Dell XPS

Задание 2

Asus

MacBook Pro

Lenovo

Dell XPS

MacBook Air

HP

Компьютер: HP , Минимальный год выпуска программы: 1982

Компьютер: Dell XPS, Минимальный год выпуска программы: 1983

Компьютер: Asus, Минимальный год выпуска программы: 1988

Компьютер: MacBook Pro, Минимальный год выпуска программы: 1997

Компьютер: MacBook Air, Минимальный год выпуска программы: 1998

Компьютер: Lenovo, Минимальный год выпуска программы: 2003

Задание 3

Программа: AutoCAD, Год выпуска: 1982, Компьютер: HP

Программа: Blender, Год выпуска: 1998, Компьютер: MacBook Air

Программа: Firefox, Год выпуска: 2002, Компьютер: Dell XPS

Программа: Microsoft Word, Год выпуска: 1983, Компьютер: Dell XPS

Программа: Microsoft Word, Год выпуска: 1983, Компьютер: HP

Программа: Notepad++, Год выпуска: 2003, Компьютер: Lenovo

Программа: Photoshop, Год выпуска: 1988, Компьютер: Asus

Программа: Photoshop, Год выпуска: 1988, Компьютер: MacBook Air

Программа: PyCharm, Год выпуска: 2010, Компьютер: Asus

Программа: PyCharm, Год выпуска: 2010, Компьютер: MacBook Pro

Программа: Slack, Год выпуска: 2013, Компьютер: Asus

Программа: Visual Studio, Год выпуска: 1997, Компьютер: MacBook Pro

Программа: Visual Studio, Год выпуска: 1997, Компьютер: Lenovo

Программа: Zoom, Год выпуска: 2012, Компьютер: Lenovo