



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

Факультет «ГУИМЦ»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»»

ОТЧЕТ

ПО РУБЕЖНОМУ КОНТРОЛЮ №1

Вариант предметной области 27

Вариант запросов: Б

Студент: Мефодьев И.Н., группа ИУ5Ц-52Б

Преподаватель: Гапанюк Ю. Е.

Москва, 2024 г.

Вариант запросов Б. Предметная область 27

1. «Преподаватель» и «Учебный курс» связаны соотношением один-ко-многим. Выведите список всех связанных преподавателей и учебных курсов, отсортированный по преподавателям, сортировка по курсам произвольная.
2. «Преподаватель» и «Учебный курс» связаны соотношением один-ко-многим. Выведите список преподавателей с количеством учебных курсов у каждого преподавателя, отсортированный по количеству ведомых им курсов.
3. «Преподаватель» и «Учебный курс» связаны соотношением многие-ко-многим. Выведите список всех преподавателей, у которых фамилия заканчивается на «ов», и список их курсов (для каждого отдельно).

Листинг программы

```
# Мефодьев ИУ5Ц-52Б вариант 27 задание Б
# классы Преподаватель, учебный курс

class Prepod:    #Преподаватель
    def __init__(self, id: int, FIO: str, salary: float):
        self.id = id
        self.FIO = FIO
        self.salary = salary

class Course:    # учебный курс
    def __init__(self, id: int, name: str, prepod_id: int):
        self.id = id
        self.name = name
        self.prepod_id = prepod_id

class Prep_course:    # курсы у преподавателя (для реализации
связи многие-ко-многим)
    def __init__(self, prep_id: int, course_id: int):
        self.prep_id = prep_id
        self.course_id = course_id

Prepods = [
    Prepod(1, "Большаков Сергей Алексеевич", 45000.00),
    Prepod(2, "Крылов Алексей Олегович", 55000.00),
    Prepod(3, "Яковишена Светлана Георгиевна", 40000.00),
    Prepod(4, "Маслеников Константин Юрьевич", 65000.00),
```

```

    Prepod(5, "Чепик Елена Чеславовна", 30000.00),
]

Courses = [
    Course(1, "Основы программирования", 1),
    Course(2, "Системное программирование", 1),
    Course(3, "История", 2),
    Course(4, "Английский язык", 3),
    Course(5, "Модели данных", 4),
    Course(6, "Базы данных", 4),
    Course(7, "Оперативный анализ данных", 4),
    Course(8, "Инженерная графика", 5)
]

Cour_preps = [
    Prep_course(1, 1),
    Prep_course(1, 2),
    Prep_course(2, 3),
    Prep_course(3, 4),
    Prep_course(4, 5),
    Prep_course(4, 6),
    Prep_course(4, 7),
    Prep_course(5, 8),
]

def task1(prepos: list[Prepod], courses: list[Course]):
    print("Запрос 1")
    result = [(p, c) # формируем связь один ко многим
               for p in prepos
               for c in courses
               if p.id == c.prepod_id
    ]

    # сортируем по преподавателям
    result.sort(key = lambda el: el[0].FIO)

    for (p, c) in result:
        print(p.FIO, "\t - \t", c.name)

```

```

def task2(prepos: list[Prepod], courses: list[Course]):
    print("Запрос 2")
    result = []
    for p in prepos:
        tmp_res = (p, [])
        for c in courses:
            if p.id == c.prepod_id:
                tmp_res[1].append(c)
        result.append(tmp_res)

    #сортировка по количеству курсов у преподавателей
    result.sort(key = lambda el: len(el[1]), reverse=True)

    for (p, c) in result: print(p.FIO, "\t - \t", len(c))

def task3(prepos: list[Prepod], courses: list[Course],
prep_courses: list[Prep_course]):
    print("Запрос 3")
    result = []
    curPrepID = 0
    index = -1
    for el in prep_courses:
        #print(prepos[el.prep_id - 1].FIO.split()[0][-2:])
        if(prepos[el.prep_id - 1].FIO.split()[0][-2:] ==
"ОВ" and curPrepID != el.prep_id):
            # второе условие - чтоб избежать повторов
            curPrepID = el.prep_id
            index+=1
            result.append((Prepos[curPrepID-1], []))

        if(curPrepID == el.prep_id):
            result[index][1].append(Courses[el.course_id])

    for el in result:
        print(el[0].FIO, end=":\n")

```

```

        for elC in el[1]:
            print("\t",elC.name, end=";\n")

        print("\n")

task1(Prepods, Courses)
print()
task2(Prepods, Courses)
print()
task3(Prepods, Courses, Cour_preps)

```

Результат работы программы

```

Запрос 1
Большаков Сергей Алексеевич - Основы программирования
Большаков Сергей Алексеевич - Системное программирование
Крылов Алексей Олегович - История
Маслеников Константин Юрьевич - Модели данных
Маслеников Константин Юрьевич - Базы данных
Маслеников Константин Юрьевич - Оперативный анализ данных
Чепик Елена Чеславовна - Инженерная графика
Яковишена Светлана Георгиевна - Английский язык

Запрос 2
Маслеников Константин Юрьевич - 3
Большаков Сергей Алексеевич - 2
Крылов Алексей Олегович - 1
Яковишена Светлана Георгиевна - 1
Чепик Елена Чеславовна - 1

Запрос 3
Большаков Сергей Алексеевич:
    Системное программирование;
    История;

Крылов Алексей Олегович:
    Английский язык;

Маслеников Константин Юрьевич:
    Базы данных;
    Оперативный анализ данных;
    Инженерная графика;

PS W:\BM57U\paradigmy\RK_1>

```