

Факультет «Радиотехнический»

Кафедра ИУ5 «Системы обработки информации и управления»

**Отчет по лабораторной работе №1 по курсу  
Базовые компоненты интернет-технологий**

5

(количество листов)

Исполнитель

Студент группы РТ5-316

\_\_\_\_\_

Усанкин А.А.

«26» октября 2021 г.

## Условия рубежного контроля №1 по курсу БКИТ

Рубежный контроль представляет собой разработку программы на языке Python, которая выполняет следующие действия:

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 Е.

1. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список всех отделов, у которых в названии присутствует слово «отдел», и список работающих в них сотрудников.
2. «Отдел» и «Сотрудник» связаны соотношением один-ко-многим. Выведите список отделов со средней зарплатой сотрудников в каждом отделе, отсортированный по средней зарплате. Средняя зарплата должна

быть округлена до 2 знака после запятой (отдельной функции вычисления среднего значения в Python нет, нужно использовать комбинацию функций вычисления суммы и количества значений; для округления необходимо использовать функцию <https://docs.python.org/3/library/functions.html#round>).

## Текст программы

```
from operator import itemgetter

class Operator:
    """Оператор"""
    def __init__(self, id, surname, salary, id_CompLang):
        self.id = id
        self.surname = surname
        self.salary = salary
        self.id_CompLang = id_CompLang

class CompLang:
    """Язык программирования"""
    def __init__(self, id, name):
        self.id = id
        self.name = name

class CompLang_Operator:
    """Операторы языка программирования"""
    def __init__(self, id_Operator, id_CompLang):
        self.id_Operator = id_Operator
        self.id_CompLang = id_CompLang

Operators=[
    Operator(0, 'Иванов', 1000, 1),
    Operator(1, 'Петров', 2000, 2),
    Operator(2, 'Александров', 3000, 4),
    Operator(3, 'Сидоров', 4000, 1),
    Operator(4, 'Абдулов', 2500, 3),
    Operator(5, 'Нурмагомедов', 1300, 2),
    Operator(6, 'Семенов', 2000, 3),
    Operator(7, 'Оппенгеймер', 4300, 4),
    Operator(8, 'Лоуренс', 2300, 1),
    Operator(9, 'Неймер', 1700, 0),
]

CompLangs=[
    CompLang(0, 'язык программирования C++'),
    CompLang(1, 'C'),
    CompLang(2, 'язык программирования Python'),
    CompLang(3, 'Pascal'),
    CompLang(4, 'Java'),
]

CompLang_Operators=[
    CompLang_Operator(0, 0),
    CompLang_Operator(0, 2),
    CompLang_Operator(0, 4),
    CompLang_Operator(1, 2),
    CompLang_Operator(2, 4),
```

```

CompLang_Operator(2,1),
CompLang_Operator(2,0),
CompLang_Operator(3,1),
CompLang_Operator(3,2),
CompLang_Operator(4,3),
CompLang_Operator(5,2),
CompLang_Operator(6,3),
CompLang_Operator(7,4),
CompLang_Operator(8,1),
CompLang_Operator(9,0),
]
def main():
    """Основная функция"""

    one_to_many = [(o.surname, o.salary, c.name)
                    for o in Operators
                    for c in CompLangs
                    if c.id == o.id_CompLang]

    many_to_many_temp = [(c.name, oc.id_CompLang, oc.id_Operator)
                          for c in CompLangs
                          for oc in CompLang_Operators
                          if c.id == oc.id_CompLang]

    many_to_many = [(o.surname, o.salary, CompLang_name)
                     for CompLang_name, CompLang_id, Operator_id in
many_to_many_temp
                     for o in Operators
                     if o.id == Operator_id]

    print('Задание E1')
    res = list()
    for surname, salary, nameCompLang in one_to_many:
        if 'язык программирования' in nameCompLang.lower():
            res.append((surname, salary, nameCompLang))
    print(res)
    print('Задание E2')
    res = []
    res1 = []
    for c in CompLangs:
        c_operators = list(filter(lambda i: i[2] == c.name, one_to_many))
        if len(c_operators)>0:
            c_complangs = [salary for _, salary, _ in c_operators]
            c_sals_midle = round(sum(c_complangs)/len(c_complangs),2)
            res1.append((c.name, c_sals_midle))
    res = sorted(res1, key=itemgetter(1), reverse=True)
    print(res)

    print('Задание E3')
    for surname, salary, name in many_to_many:
        if surname[0] == 'А': print('Язык:', name, 'Фамилия:', surname, ';')

if __name__ == "__main__":
    main()

```

## Пример выполнения программы:

```

[('Петров', 2000, 'язык программирования Python'), ('Нурмагомедов', 1300, 'язык программирования Python'), ('Неймер', 1700, 'язык программирования C++')]
Задание E2
[('Java', 3650.0), ('C', 2433.33), ('Pascal', 2250.0), ('язык программирования C++', 1700.0), ('язык программирования Python', 1650.0)]
Задание E3
Язык: язык программирования C++ Фамилия: Александров ;
Язык: C Фамилия: Александров ;
Язык: Pascal Фамилия: Абдулов ;
Язык: Java Фамилия: Александров ;

```