

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

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

Курс «Парадигмы и конструкции языков программирования»

РК №2

Выполнил:
студент группы ИУ5-24Б:
Здобняков Фёдор
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю.
Подпись и дата:

Москва, 2023 г.

Условия рубежного контроля №2 по курсу ПиК ЯП

Рубежный контроль представляет собой разработку тестов на языке Python.

- 1) Проведите рефакторинг текста программы рубежного контроля №1 таким образом, чтобы он был пригоден для модульного тестирования.
- 2) Для текста программы рубежного контроля №1 создайте модульные тесты с применением TDD - фреймворка (3 теста).

Код программы:

дд.ру:

```
mus_ID = 1
orc_ID = 1

class Orchestra:
    def __init__(self, name):
        global orc_ID
        self.name = name
        self.orc_ID = orc_ID

        orc_ID += 1

class Musician:
    def __init__(self, name, salary, orc_ID):
        global mus_ID
        self.name = name
        self.salary = salary
        self.mus_ID = mus_ID
        self.orc_ID = orc_ID
        mus_ID += 1

class Mus_In_Orc:
    def __init__(self, mus_ID, orc_ID):
        self.mus_ID = mus_ID
        self.orc_ID = orc_ID

mus = []
orc = []
m_o = []

with open("musician.txt", "r", encoding='utf8') as file:
    for line in file:
        data = line.split()
        mus1 = Musician(data[0], int(data[1]), int(data[2]))
        mus.append(mus1)

with open("Orchestra.txt", "r", encoding='utf8') as file:
    for line in file:
        data = line.split()
        orc1 = Orchestra(data[0])
        orc.append(orc1)

for i in orc:
    for j in mus:
        if i.orc_ID == j.orc_ID:
            m_o1 = Mus_In_Orc(j.mus_ID, i.orc_ID)
            m_o.append(m_o1)

#Задание 1
print("Задание 1")
for i in orc:
    if i.name[0] == 'A':
```

```

        print(i.name)
        print("Сотрудники:")
        for j in mus:
            if i.orc_ID == j.orc_ID:
                print(j.name)

def task1(orc, mus):
    ans = []
    for i in orc:
        if i.name[0] == 'A':
            print(i.name)
            print("Сотрудники:")
            for j in mus:
                if i.orc_ID == j.orc_ID:
                    ans.append(j.name)

    return ans

print('\n', '\n')
print("Задание 2")
#Задание 2

sp = []
for i in orc:
    max_sal = 0
    for j in mus:
        if j.orc_ID == i.orc_ID and j.salary > max_sal:
            max_sal = j.salary

    sp.append([i.name, max_sal])

sp.sort(key=lambda arr: arr[1])
sp = sp
for i in sp:
    print(i[0], "максимальная зарплата:", i[1])

print("Отсортировано по возрастанию")

def task2(orc, mus):
    ans = []
    sp = []
    for i in orc:
        max_sal = 0
        for j in mus:
            if j.orc_ID == i.orc_ID and j.salary > max_sal:
                max_sal = j.salary

        sp.append([i.name, max_sal])

    sp.sort(key=lambda arr: arr[1])
    sp = sp
    for i in sp:
        ans.append([i[0], "максимальная зарплата:", i[1]])

    return ans

#Задание 3
print('\n', '\n')
print("Задание 3")

for i in orc:

```

```

print("Сотрудники, работающие в оркестре", i.name, ':')
for j in mus:
    if j.orc_ID == i.orc_ID:
        print(j.name)
print('')

def task3(orc, mus):
    ans = []
    for i in orc:
        elem = []
        for j in mus:
            if j.orc_ID == i.orc_ID:
                elem.append(j.name)
        if len(elem) != 0:
            ans.append(elem)
    return ans

```

rk2.py:

```

from unittest import *
import unittest

class Test(unittest.TestCase):
    def data(self):
        self.mus = [
            Musician("Василий", 30000, 2),
            Musician("Сергей", 40000, 1),
            Musician("Виктор", 20000, 3),
            Musician("Никита", 50000, 2),
            Musician("Дмитрий", 60000, 1)
        ]

        self.orc = [
            Orchestra("Aria_Orchestra"),
            Orchestra("Estrad_Orchestra"),
            Orchestra("Jazz_Orchestra")
        ]

    def test1(self):
        res = task1(orc, mus)
        expected_res = ["Сергей", "Дмитрий"]
        self.assertEqual(res, expected_res)

    def test2(self):
        res = task2(orc, mus)
        expected_res = [
            ["Jazz_Orchestra", "максимальная зарплата:", 20000],
            ["Estrad_Orchestra", "максимальная зарплата:", 50000],
            ["Aria_Orchestra", "максимальная зарплата:", 60000]
        ]
        self.assertEqual(res, expected_res)

    def test3(self):
        res = task3(orc, mus)
        expected_res = [
            ["Сергей", "Дмитрий"],
            ["Василий", "Никита"],
            ["Виктор"]
        ]
        self.assertEqual(res, expected_res)

if __name__ == '__main__':
    unittest.main()

```

Текстовые базы данных:

1	Aria_Orchestra	
2	Estrad_Orchestra	
3	Jazz_Orchestra	
4		
1	Василий 30000 2	
2	Сергей 40000 1	
3	Виктор 20000 3	
4	Никита 50000 2	
5	Дмитрий 60000 1	

Результаты:

```
Run Python tests in rk2.py x
Test Results 0ms
Tests passed: 3 of 3 tests - 0ms
C:\Users\Fzdob\PycharmProjects\Foophbot\venv\Scripts\python.exe "D:/progs/PyCharm Community Edition 2023.2.2/plugins/python-test-runner/runners/runner.py"
Testing started at 11:15 ...
Launching unittests with arguments python -m unittest C:\Users\Fzdob\PycharmProjects\Foophbot\rk2.py in C:\Users\Fzdob\PycharmProjects\Foophbot

Задание 1
Aria_Orchestra
Сотрудники:
Сергей
Дмитрий

Задание 2
Jazz_Orchestra максимальная зарплата: 20000
Estrad_Orchestra максимальная зарплата: 50000
Aria_Orchestra максимальная зарплата: 60000
Отсортировано по возрастанию

Задание 3
Сотрудники, работающие в оркестре Aria_Orchestra :
Сергей
Дмитрий

Сотрудники, работающие в оркестре Estrad_Orchestra :
Василий
Никита

Сотрудники, работающие в оркестре Jazz_Orchestra :
Виктор

Aria_Orchestra
Сотрудники:

Ran 3 tests in 0.001s
OK
Process finished with exit code 0
```