

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

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

Курс «Базовые компоненты интернет-технологий»

Рубежный контроль №2  
Вариант 12

Выполнил:

студент группы ИУ5-34Б  
Ромашко Дарья

Подпись и дата:

Проверил:

Гапанюк Ю. Е.

Подпись и дата:

Москва, 2021 г.

### **Постановка задачи.**

- 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).

### **Текст программы:**

#### **Файл language.py**

```
class language:
    def __init__(self, id, title, year_app, tool_id):
        self.id = id
        self.title = title
        self.year_app = year_app
        self.tool_id = tool_id
```

#### **Файл mytool.py**

```
class tool:
    def __init__(self, id, name):
        self.id = id
        self.name = name
```

#### **Файл LangTool.py**

```
class LangTool:
    def __init__(self, tool_id, lang_id):
        self.tool_id = tool_id
        self.lang_id = lang_id
```

#### **Файл data.py**

```
from language import language
from mytool import tool
from LangTool import LangTool
#список средств разработки
tools = [tool(1, "Visual Studio"),
          tool(2, "Visual Studio Code"),
          tool(3, 'Android Studio')]
#список языков программирования
languages = [ language(1, 'Python', 1991, 1),
              language(2, 'C++', 1983, 2),
              language(3, 'Assembly', 1949, 3),
              language(4, 'Pascal', 1970, 2),
              language(5, 'Java', 1995, 1)]
#список связей многие-ко-многим
lang_tool = [
    LangTool(1, 1),
    LangTool(1, 2),
    LangTool(1, 3),
    LangTool(2, 2),
    LangTool(3, 4),
    LangTool(3, 5),
```

```

]
#связь один-ко-многим
one_to_many = [(l.title, l.year_app, t.name)
                for t in tools
                for l in languages
                if l.tool_id == t.id]
#связь многие-ко-многим
many_to_many = [(l.title, l.year_app, t.name)
                 for t in tools
                 for l in languages
                 for relation in lang_tool
                 if t.id == relation.tool_id and l.id == relation.lang_id]

```

#### **Файл tests.py**

```

from data import *
from main import task1, task2, task3
import unittest

class Tests(unittest.TestCase):
    def test_task1(self):
        self.assertEqual(task1(), [('Assembly', 1949, 'Android Studio')])

    def test_task2(self):
        self.assertEqual(task2(), [('Visual Studio', 1995), ('Visual Studio Code', 1983),
                                    ('Android Studio', 1949)])

    def test_task3(self):
        self.assertEqual(task3(), [('Pascal', 'Android Studio'), ('Java', 'Android
Studio'), ('Python', 'Visual Studio'), ('C++', 'Visual Studio'), ('Assembly', 'Visual
Studio'), ('C++', 'Visual Studio Code')])

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

```

#### **Файл main.py**

```

from operator import itemgetter
from data import*

def task1():
    task_1=[]
    for i in range(len(one_to_many)):
        if ((one_to_many[i][2])[0] == 'A'):
            task_1.append(one_to_many[i])
    return task_1

```

```

def task2():
    res_2_unsorted = []
    # Перебираем все среды программирования
    for t in tools:
        # Список языков сред программирования
        t_lang = list(filter(lambda x: x[2] == t.name, one_to_many))
        # Если среда программирования не пуста
        if len(t_lang) > 0:
            res_2_unsorted.append((t.name, max(t_lang, key=lambda x: x[1])[1]))

    # Сортировка по новизне
    res_2 = sorted(res_2_unsorted, key=itemgetter(1), reverse=True)
    return(res_2)

def task3():
    res_3 = []
    # Перебираем все среды программирования
    for lang, _, tool in many_to_many:
        res_3.append((lang, tool))
    res_3 = sorted(res_3, key=itemgetter(1))
    return(res_3)

def main():

    print('Задание Г1')
    print(task1())
    print("\nЗадание Г2")
    print(task2())
    print("\nЗадание Г3")
    print(task3())

if __name__ == '__main__':
    main()

```

## Результат выполнения

```

Задание Г1
('Assembly', 1949, 'Android Studio')

Задание Г2
[('Visual Studio', 1995), ('Visual Studio Code', 1983), ('Android Studio', 1949)]

Задание Г3
[('Pascal', 'Android Studio'), ('Java', 'Android Studio'), ('Python', 'Visual Studio'), ('C++', 'Visual Studio'), ('Assembly', 'Visual Studio'), ('C++', 'Visual Studio Code')]

Process finished with exit code 0

```

## Результат тестирования

Testing started at 14:16 ...

Launching unittests with arguments python -m unittest C:/Users/ASUS/PycharmProjects/RK/tests.py in C:\Users\ASUS\PycharmProjects\RK

Ran 3 tests in 0.004s

OK

Process finished with exit code 0