Леонтьева Вера Анатольевна
ИБМ3-34Б

Парадигмы и конструкции языков программирования

Рубежный Контроль №2
Вариант 12

Задание:

Рубежный контроль представляет собой разработку тестов на языке Python.
1) Проведите рефакторинг текста программы рубежного контроля N°1 таким образом, чтобы он был пригоден для модульного тестирования.
2) Для текста программы рубежного контроля N°1 создайте модульные тесты с применением TDD - фреймворка (3 теста).

1. Проведем рефакторинг текста программы из РК1, чтобы он был пригоден для модульного тестирования.

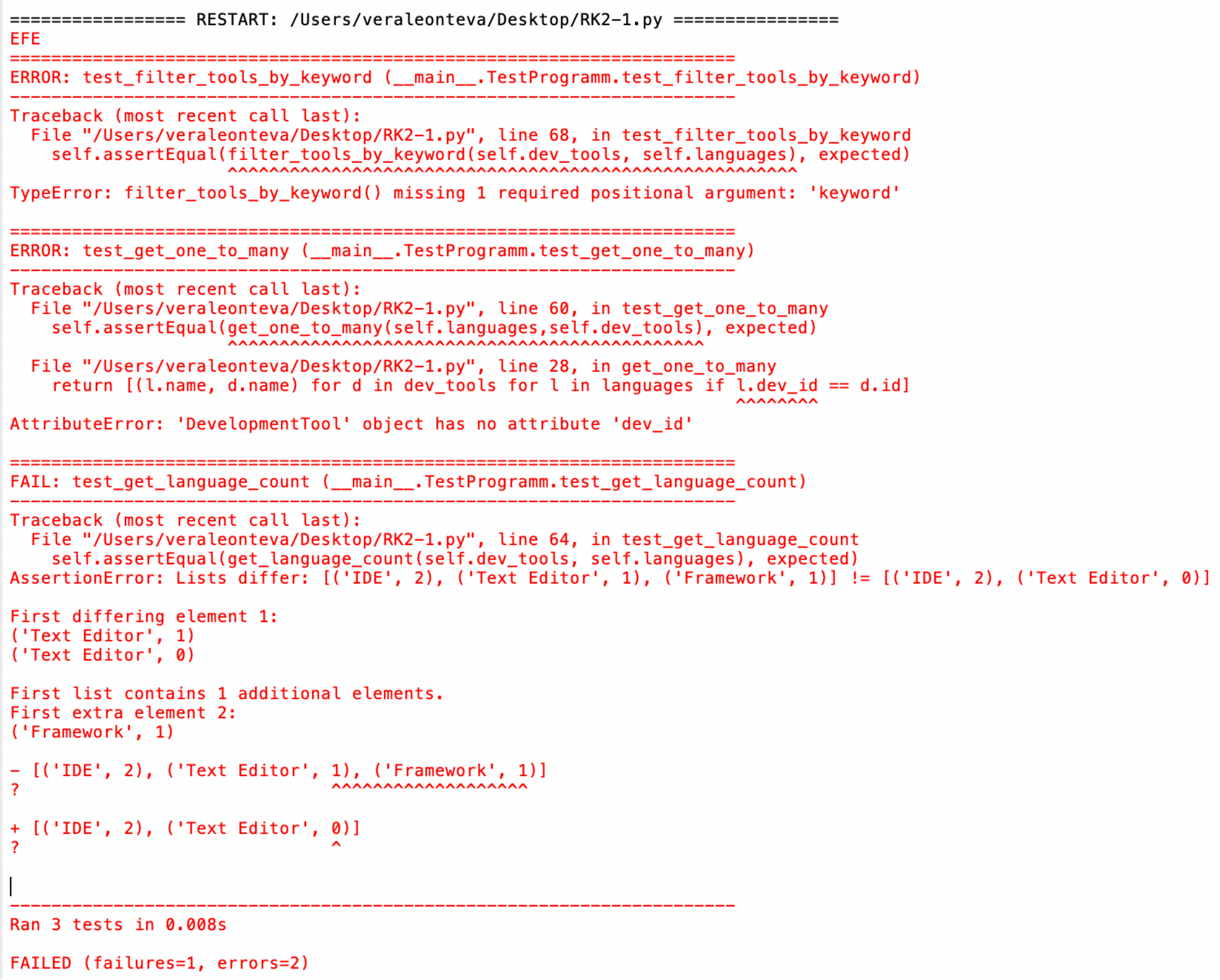
```
from operator import itemgetter
import unittest

class Language:
    """Язык программирования"""

    def __init__(self, id, name, dev_id):
        self.id = id
        self.name = name
        self.dev_id = dev_id

class DevelopmentTool:
    """Средство разработки"""

    def __init__(self, id, name):
        self.id = id
        self.name = name

class ToolLang:
    """Связь между средствами разработки и языками
программирования"""
```

```
    def __init__(self, dev_id, lang_id):
        self.dev_id = dev_id
        self.lang_id = lang_id


def get_one_to_many(dev_tools, languages):
    return [(l.name, d.name) for d in dev_tools for l in languages
if l.dev_id == d.id]

def get_language_count(dev_tools, languages):
    return [(d.name, len([l for l in languages if l.dev_id ==
d.id])) for d in dev_tools]

def filter_tools_by_keyword(dev_tools, languages, keyword):
    return {d.name: [l.name for l in languages if l.dev_id ==
d.id] for d in dev_tools if keyword in d.name}
```

2. Создаем модульные тесты

```
class TestProgramm(unittest.TestCase):

    def setUp(self):
        self.dev_tools = [
            DevelopmentTool(1, 'IDE'),
            DevelopmentTool(2, 'Text Editor'),
            DevelopmentTool(3, 'Framework'),
        ]
        self.languages = [
            Language(1, 'Python', 1),
            Language(2, 'JavaScript', 1),
            Language(3, 'Ruby', 2),
            Language(4, 'Java', 3),
        ]
        self.tool_langs = [
            ToolLang(1, 1),
            ToolLang(1, 2),
            ToolLang(2, 3),
            ToolLang(3, 4),
        ]

    def test_get_one_to_many(self):
        expected = [('Python', 'IDE'), ('JavaScript', 'IDE'),
('Ruby', 'Text Editor'), ('Java', 'Framework')]

self.assertEqual(get_one_to_many(self.languages,self.dev_tools),
expected)

    def test_get_language_count(self):
        expected = [('IDE', 2), ('Text Editor', 0)]
        self.assertEqual(get_language_count(self.dev_tools,
self.languages), expected)

    def test_filter_tools_by_keyword(self):
        result = {'Text Editor': ['Python', 'JavaScript']}
        self.assertEqual(filter_tools_by_keyword(self.dev_tools,
self.languages), expected)

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

Результат:



```
================= RESTART: /Users/veraleonteva/Desktop/RK2-1.py ================
EFE
======================================================================
ERROR: test_filter_tools_by_keyword (__main__.TestProgramm.test_filter_tools_by_keyword)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "/Users/veraleonteva/Desktop/RK2-1.py", line 68, in test_filter_tools_by_keyword
    self.assertEqual(filter_tools_by_keyword(self.dev_tools, self.languages), expected)
                     ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
TypeError: filter_tools_by_keyword() missing 1 required positional argument: 'keyword'

======================================================================
ERROR: test_get_one_to_many (__main__.TestProgramm.test_get_one_to_many)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "/Users/veraleonteva/Desktop/RK2-1.py", line 60, in test_get_one_to_many
    self.assertEqual(get_one_to_many(self.languages,self.dev_tools), expected)
                     ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/Users/veraleonteva/Desktop/RK2-1.py", line 28, in get_one_to_many
    return [(l.name, d.name) for d in dev_tools for l in languages if l.dev_id == d.id]
                                                                      ^^^^^^^^
AttributeError: 'DevelopmentTool' object has no attribute 'dev_id'

======================================================================
FAIL: test_get_language_count (__main__.TestProgramm.test_get_language_count)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "/Users/veraleonteva/Desktop/RK2-1.py", line 64, in test_get_language_count
    self.assertEqual(get_language_count(self.dev_tools, self.languages), expected)
AssertionError: Lists differ: [('IDE', 2), ('Text Editor', 1), ('Framework', 1)] != [('IDE', 2), ('Text Editor', 0)]

First differing element 1:
('Text Editor', 1)
('Text Editor', 0)

First list contains 1 additional elements.
First extra element 2:
('Framework', 1)

- [('IDE', 2), ('Text Editor', 1), ('Framework', 1)]
?                              ^^^^^^^^^^^^^^^^^^^

+ [('IDE', 2), ('Text Editor', 0)]
?                              ^

|
----------------------------------------------------------------------
Ran 3 tests in 0.008s

FAILED (failures=1, errors=2)
```