

**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»

Отчет по Лабораторной работе 5.

Выполнил:

студент группы ИУ5-33Б
Комаров Дмитрий

Проверил:

преподаватель каф. ИУ5
Юрий Евгеньевич Гапанюк

Подпись и дата:

Подпись и дата:

Москва, 2022 г.

Постановка задачи

- Выберите любой фрагмент кода из лабораторных работ 1 или 2 или 3-4.
- Модифицируйте код таким образом, чтобы он был пригоден для модульного тестирования.
- Разработайте модульные тесты. В модульных тестах необходимо применить следующие технологии:
 - о TDD - фреймворк (не менее 3 тестов).
 - о BDD - фреймворк (не менее 3 тестов).
 - о Создание Моск-объектов (необязательное дополнительное задание).

Код TDD теста:

```
import unittest
from lab_python_fp.field import field
from lab_python_fp.sort import sort_abs

class Test_field(unittest.TestCase):
    def setUp(self):
        self.goods = [
            {'title': 'Ковер', 'price': 2000, 'color': 'green'},
            {'title': 'Диван для отдыха', 'price': 5300, 'color': 'black'}
        ]
        self.data = [4, -30, 100, -100, 123, 1, 0, -1, -4]

    def test_field1(self):
        self.assertEqual(list(field(self.goods, 'title')), ['Ковер',
'Диван для отдыха'])

    def test_fied2(self):
        self.assertEqual(list(field(self.goods, 'title', 'price')),
[{'title': 'Ковер', 'price': 2000}, {'title': 'Диван для отдыха',
'price': 5300}])

    def test_sort(self):
        self.assertEqual(sort_abs(self.data), [123, 100, -100, -30, 4,
-4, 1, -1, 0])

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

Результат:

```

userdead@DESKTOP-439N993:/mnt/c/Users/dimka/BK11_Labs/lab5$ python3 unit_test_module.py
...
-----
Ran 3 tests in 0.001s

OK

```

Код BDD теста:

```

from behave import given, when, then
from lab_python_fp.sort import sort_abs, sort_abs1

@given("List is [{array}]")
def step_given(context, array):
    context.array = [int(i) for i in array.split(', ')]

@when("Sorted this list with Sort_abs")
def step_when(context):
    context.sorted_array = sort_abs(context.array)

@then("List is [{array}]")
def step_then(context, array):
    result = [int(i) for i in array.split(', ')]
    assert context.sorted_array == result

```

файл feature:

```

≡ example.feature X
features > ≡ example.feature
1  Feature: Sorting
2      Scenario: Sort_abs with lambda
3          Given List is [4, -30, 100, -100, 123, 1, 0, -1, -4]
4          When Sorted this list with Sort_abs
5          Then List is [123, 100, -100, -30, 4, -4, 1, -1, 0]
6

```

Результат:

```

userdead@DESKTOP-439N993:/mnt/c/Users/dimka/BK11_Labs/lab5$ behave
Feature: Sorting # features/example.feature:1

  Scenario: Sort_abs with lambda # features/example.feature:2
    Given List is [4, -30, 100, -100, 123, 1, 0, -1, -4] # features/steps/example_steps.py:4 0.000s
    When Sorted this list with Sort_abs # features/steps/example_steps.py:8 0.000s
    Then List is [123, 100, -100, -30, 4, -4, 1, -1, 0] # features/steps/example_steps.py:12 0.000s

1 feature passed, 0 failed, 0 skipped
1 scenario passed, 0 failed, 0 skipped
3 steps passed, 0 failed, 0 skipped, 0 undefined
Took 0m0.000s

```