



**Министерство науки и высшего образования  
Российской Федерации Федеральное государственное  
бюджетное образовательное учреждение высшего  
образования «Московский государственный  
технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)**

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

Лабораторная работа №4  
по дисциплине «Базовые компоненты интернет-технологий»

Выполнил:  
студент группы ИУ5-32Б  
Долинский А.А.

Проверил:  
Канев А.И.

2021 г.

## Описание задания

1. Необходимо для произвольной предметной области реализовать от одного до трех шаблонов проектирования: один порождающий, один структурный и один поведенческий. В качестве справочника шаблонов можно использовать [следующий каталог](#). Для сдачи лабораторной работы в минимальном варианте достаточно реализовать один паттерн.
2. Вместо реализации паттерна Вы можете написать тесты для своей программы решения биквадратного уравнения. В этом случае, возможно, Вам потребуется доработать программу решения биквадратного уравнения, чтобы она была пригодна для модульного тестирования.
3. В модульных тестах необходимо применить следующие технологии:
  - о TDD - фреймворк.
  - о BDD - фреймворк.
  - о Создание Mock-объектов.

**Цель лабораторной работы:** изучение реализации шаблонов проектирования и возможностей модульного тестирования в языке Python.

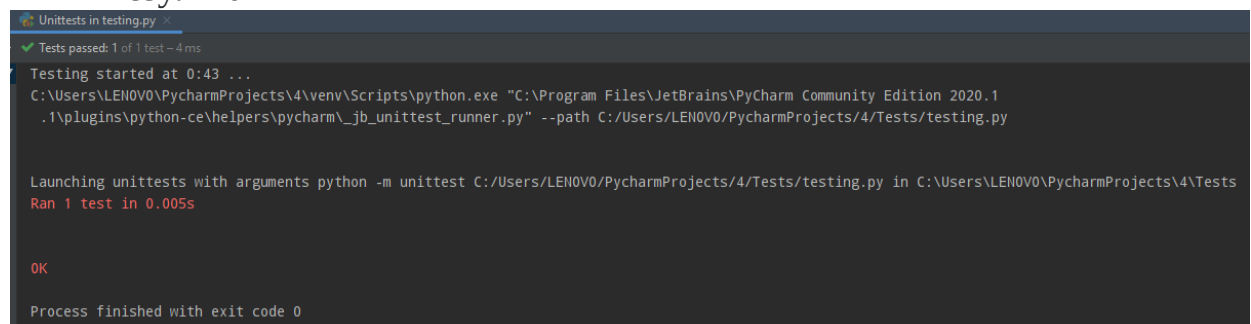
testing.py

```
import unittest
import sys, os
sys.path.append(os.getcwd())
from main import *

class Testget_roots(unittest.TestCase):
    def test_get_roots(self):
        self.assertEqual(get_roots(0, 0, 0), [])
        self.assertEqual(get_roots(5, 6, 7), [])
        self.assertEqual(get_roots(0, 4, 0), [0])
        self.assertEqual(get_roots(1, -18, 81), [3, -3])
        self.assertEqual(get_roots(1, -4, 0), [2, -2, 0])
        self.assertEqual(get_roots(1, -5, 6), [math.sqrt(3), -math.sqrt(3),
math.sqrt(2), -math.sqrt(2)])

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

## Результат



The screenshot shows the 'Unittests in testing.py' window in PyCharm. It displays the following information:

- Tests passed: 1 of 1 test - 4 ms
- Testing started at 0:43 ...
- Command line: `C:\Users\LENOVO\PycharmProjects\4\venv\Scripts\python.exe "C:\Program Files\JetBrains\PyCharm Community Edition 2020.1\plugins\python-ce\helpers\pycharm\jb_unittest_runner.py" --path C:/Users/LENOVO/PycharmProjects/4/Tests/testing.py`
- Launch command: `python -m unittest C:/Users/LENOVO/PycharmProjects/4/Tests/testing.py`
- Output: `Ran 1 test in 0.005s`
- Status: OK
- Process finished with exit code 0

## BDD.py

```
from behave import *
from main import *

@given('I have 3 coef')
def get_coef(context):
    context.n1 = 1
    context.n2 = -4
    context.n3 = 0

@when('I count the roots')
def roots(context):
    context.result = get_roots(context.n1, context.n2, context.n3)

@then('checking the result')
def c(context):
    assert context.result == [2, -2, 0]
```

## Feature

```
Feature: BDD Test
  Scenario: Test
    Given I have 3 coef
    When I count the roots
    Then checking the result
```

## Результат

```
(venv) C:\Users\LENOVO\PycharmProjects\4>behave
Feature: BDD Test # features/BDD.feature:1

  Scenario: Test # features/BDD.feature:2
    Given I have 3 coef # features/steps/BDD.py:5
    When I count the roots # features/steps/BDD.py:12
    Then checking the result # features/steps/BDD.py:17

1 feature passed, 0 failed, 0 skipped
1 scenario passed, 0 failed, 0 skipped
3 steps passed, 0 failed, 0 skipped, 0 undefined
Took 0m0.004s
```

## mock.py

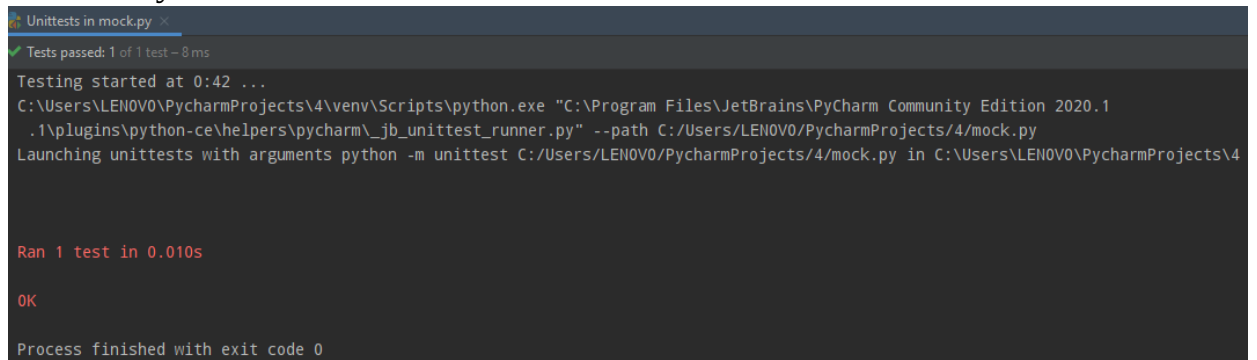
```
import unittest
from main import get_roots
from unittest.mock import patch, Mock

class Testget_roots(unittest.TestCase):
    @patch('main.get_roots', return_value = [])
    def test_get_roots(self, a):
        self.assertEqual(set(get_roots(5, 6, 7)), set([]))

    @patch('main.get_roots', return_value=[0])
    def test_get_roots(self, a):
        self.assertEqual(set(get_roots(0, 4, 0)), set([0]))

    @patch('main.get_roots', return_value=[3, -3])
    def test_get_roots(self, a):
        self.assertEqual(set(get_roots(1, -18, 81)), set([3, -3]))
```

## Результат



```
Unittests in mock.py x
✓ Tests passed: 1 of 1 test - 8 ms
Testing started at 0:42 ...
C:\Users\LEN0V0\PycharmProjects\4\venv\Scripts\python.exe "C:\Program Files\JetBrains\PyCharm Community Edition 2020.1
.1\plugins\python-ce\helpers\pycharm\_jb_unittest_runner.py" --path C:/Users/LEN0V0/PycharmProjects/4/mock.py
Launching unittests with arguments python -m unittest C:/Users/LEN0V0/PycharmProjects/4/mock.py in C:\Users\LEN0V0\PycharmProjects\4

Ran 1 test in 0.010s

OK

Process finished with exit code 0
```