

Unit 6.2 Graded Assignment

Instructions:

Implement tests for tasks/6_writing_robust_readable_and_maintainable_code/da

y_2_testing/assignment/my_proj project (there is a tests/test_sum_func.py file ready, missing the tests only).

Aim for 100% code coverage, branch coverage and condition

coverage. Run the tests with pytest. Then run the tests with a flag

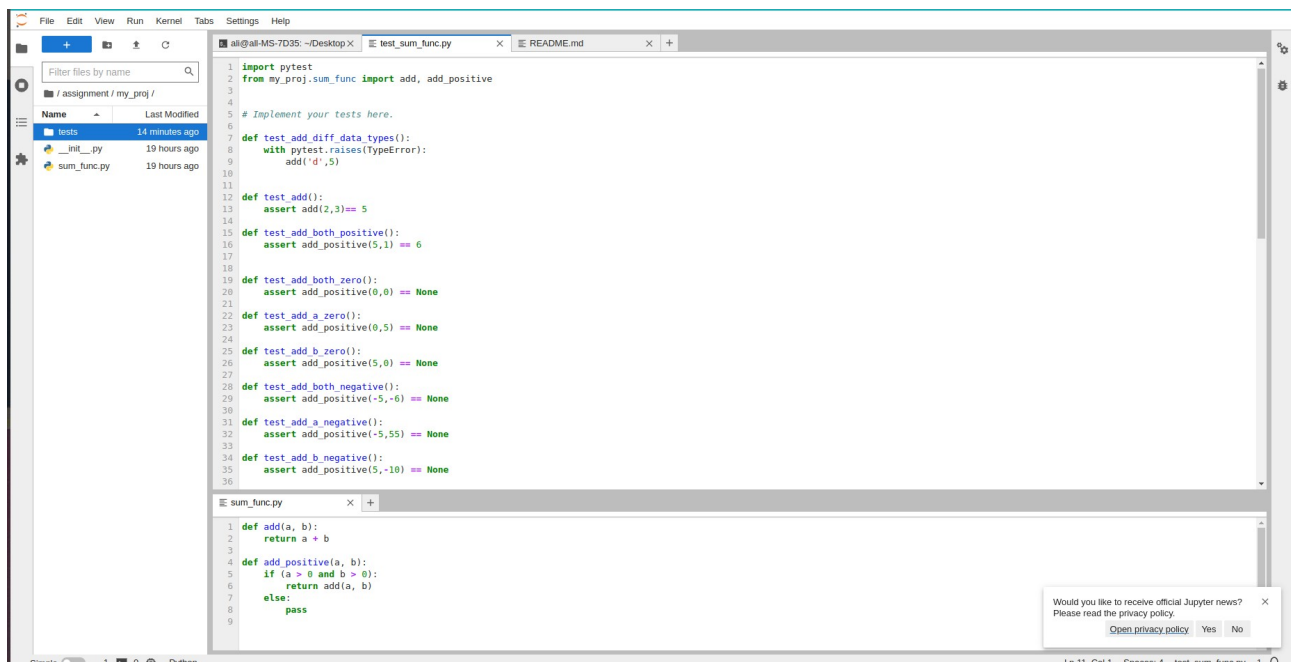
producing coverage reports (you'll need to install pytest-cov at

<https://pypi.org/project/pytest-cov/>).

Submitted by:

1. Ali Nasir (2303.KHI.DEG.012)
2. Saif ur Rehman (2303.KHI.DEG.007)

Solution:



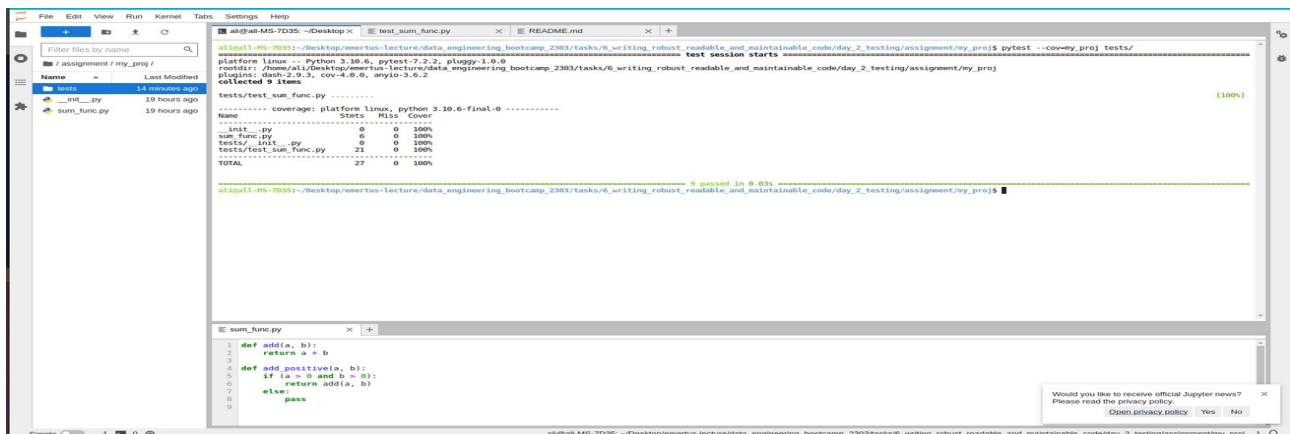
The screenshot shows a Jupyter Notebook with two open files: `test_sum_func.py` and `sum_func.py`. The `test_sum_func.py` file contains the following Python code:

```
1 import pytest
2 from my_proj.sum_func import add, add_positive
3
4 # Implement your tests here.
5
6
7 def test_add_diff_data_types():
8     with pytest.raises(TypeError):
9         add('d', 5)
10
11
12 def test_add():
13     assert add(2, 3) == 5
14
15 def test_add_both_positive():
16     assert add_positive(5, 1) == 6
17
18
19 def test_add_both_zero():
20     assert add_positive(0, 0) == None
21
22 def test_add_a_zero():
23     assert add_positive(0, 5) == None
24
25 def test_add_b_zero():
26     assert add_positive(5, 0) == None
27
28 def test_add_both_negative():
29     assert add_positive(-5, -6) == None
30
31 def test_add_a_negative():
32     assert add_positive(-5, 55) == None
33
34 def test_add_b_negative():
35     assert add_positive(5, -10) == None
36
```

The `sum_func.py` file contains the following Python code:

```
1 def add(a, b):
2     return a + b
3
4 def add_positive(a, b):
5     if (a > 0 and b > 0):
6         return add(a, b)
7     else:
8         pass
9
```

A small dialog box in the bottom right corner asks: "Would you like to receive official Jupyter news? Please read the privacy policy." with buttons for "Open privacy policy", "Yes", and "No".



1.