

IMPLEMENTASI PENGUJIAN PERANGKAT LUNAK

KELAS A



UTS

Penyusun :

Suhermanto

212410103022

PRODI INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS JEMBER

2023

Code :

```
def main():
    print('=' * 25)
    print('Menu Data')
    print('1. Sorting Data')
    print('2. Lenght Data')
    print('3. Maximum Data')
    print('4. Minimum Data')
    print('5. Average Data')
    print('=' * 25)
    hitung()
```

Dalam Fungsi Menu ini digunakan untuk membuat menu “Menu Data” seperti ada Sorting Data, Lenght Data, Maximum Data, Minimum Data, dan Average Data.

```
def hitung(operasi, data):
    if operasi == 1:
        return sorted(data)
    elif operasi == 2:
        return len(data)
    elif operasi == 3:
        return max(data)
    elif operasi == 4:
        return min(data)
    elif operasi == 5:
        if len(data) == 0:
            return None
        return sum(data) / len(data)
    else:
        return 'Tidak Valid'
```

Dalam fungsi hitung ini merupakan 6 perulangan yang digunakan untuk melakukan logika perhitungan dari menu sebelumnya, dalam fungsi ini menggunakan 2 parameter yaitu “operasi” yang digunakan untuk pilihan user, “data” ini digunakan untuk input angka user.

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

Digunakan untuk menjalankan program yang diatas dimulai dari fungsi/method main()

UnitTest :

```

import unittest
from pengolah_data import hitung

You, 1 second ago | 1 author (You)
class TestPengolahData(unittest.TestCase):
    def test_urutan_data(self):
        Result = [6,3,4,1,8,10,9]
        UrutanData = hitung(1, Result)
        self.assertEqual(UrutanData, [1, 3, 4, 6, 8, 9, 10])

    You, 4 days ago • push ...
    def test_panjang_data(self):
        Result = [6,3,4,1,8,10,9]
        PanjangData = hitung(2, Result)
        self.assertEqual(PanjangData, 7)

    def test_maksimum_data(self):
        Result = [6,3,4,1,8,10,9]
        MaksimumData = hitung(3, Result)
        self.assertEqual(MaksimumData, 10)

    def test_minimum_data(self):
        Result = [6,3,4,1,8,10,9]
        MinimumData = hitung(4, Result)
        self.assertEqual(MinimumData, 1)

    def test_ratarata_data(self):
        Result = [6,3,4,1,8,10,9]
        RataRataData = hitung(5, Result)
        self.assertAlmostEqual(RataRataData, 5.86, places=2)

    def test_invalid_input(self):
        Result = [6,3,4,1,8,10,9]
        InvalidInput = hitung(6, Result)
        self.assertEqual(InvalidInput, 'Tidak Valid')

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

```

Import Unittest ini digunakan untuk menjalankan modul unittest, from pengolah_data ini digunakan untuk mengimport fungsi hitung yang ada pada modul pengolah_data yang akan diuji, lalu class TestPengolahData digunakan untuk test data tersebut dengan menggunakan modul fungsi hitung. setiap metode ini menggunakan self.assertEqual ini untuk memeriksa apakah hasil akan sesuai yang diharapkan atau tidak. Dari code yang sudah dijelaskan itu

nantinya akan menguji apakah program atau code kalkulator tersebut akan berfungsi dengan benar dan dapat menghasilkan output sesuai dengan apa yang diharapkan.

```
.....
-----
Ran 6 tests in 0.001s

OK
```

Output yang akan dihasilkan jika semua test benar.

```
def test_urutan_data_2(self):
    Result = [6,3,4,1,8,10,9]
    UrutanData = hitung(1, Result)
    self.assertEqual(UrutanData, [3, 6, 9, 10, 8, 4, 1])

=====
FAIL: test_urutan_data_2 (__main__.TestPengolahData.test_urutan_data_2)
-----
Traceback (most recent call last):
  File "d:\Kuliah\Semester 5\IPPL\IPPL_UTS\unit-test.py", line 13, in test_urutan_data_2
    self.assertEqual(UrutanData, [3, 6, 9, 10, 8, 4, 1])
AssertionError: Lists differ: [1, 3, 4, 6, 8, 9, 10] != [3, 6, 9, 10, 8, 4, 1]

First differing element 0:
1
3

- [1, 3, 4, 6, 8, 9, 10]
+ [3, 6, 9, 10, 8, 4, 1]

-----
Ran 7 tests in 0.003s

FAILED (failures=1)
```

jika kita membuat code baru untuk mengetahui error atau tidak nya seperti yang sudah dibuat diatas, seperti kita memilih pilihan operasi 1 dan memasukan angka [6, 3, 4, 1, 8, 10, 9] harapan kita nilai yang akan keluar yaitu [1, 3, 4, 6, 8, 9, 10] tapi kita coba masukan pada code selanjutnya dengan hasil yang salah, maka output yang dihasilkan akan ada 1 test yang mengalami failed dan disitu sudah ada apa yang salah dari jawaban dan jawaban apa yang benar seperti output yang dihasilkan diatas.