CSCA08 Fall 2016

Week 12 - TESTING(DOCTEST & UNITTEST)

Bo(Kenny) Zhao

University of Toronto Scarborough

November 30, 2016

LEARNING OBJECTIVES

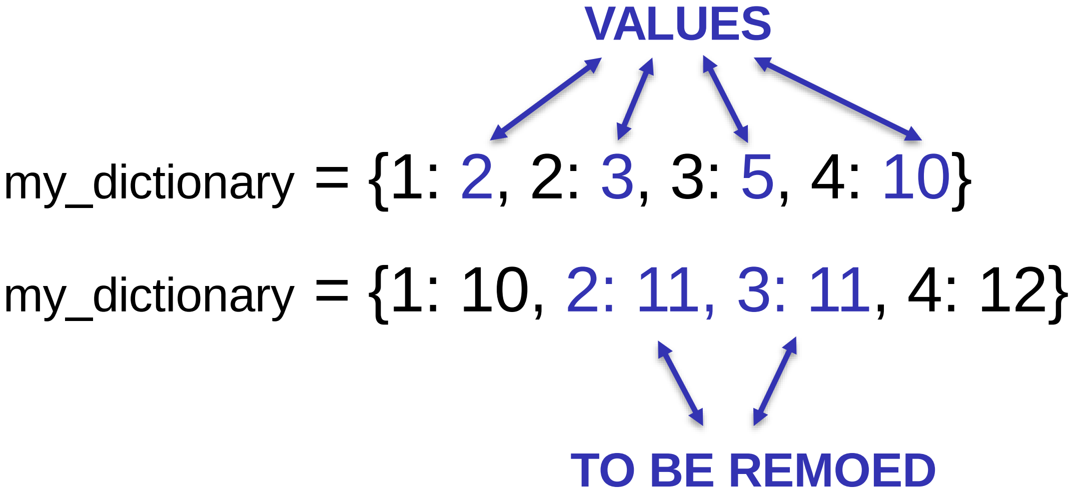
- At the end of the tutorial, you will be able to …

Test your functions thoroughly (~25 mins)

- How many test cases we need?
- What are they?
- How to write the them?
- How to test a function using the test cases?

EXAMPLE

```
def remove_dup_values(my_dictionary):
    ''' (dict of {int: int}) -> int
    Remove all entries of my dictionary whose values are not unique.
    Return the number of entries that were removed.
    '''
```



VALUES

my_dictionary = {1: 2, 2: 3, 3: 5, 4: 10}

my_dictionary = {1: 10, 2: 11, 3: 11, 4: 12}

TO BE REMOED

HOW MANY TEST CASES WE NEED & WHAT ARE THEY

length	num_dup_entries	examples
0	0	dict() or {}
1	0	{1: 2}
2	0	{1: 2, 3: 4}
2	2	{1: 3, 2: 3}
3	2	{1:3, 3:4, 5:4}
3	3	{1:3, 3:3, 5:3}
4	4(2, 2)	{1:4, 3:4, 2: 2, 0: 2}

INHERITANCE

- **To Wing**

SUMMARY

```
import unittest
import filename

class TestNameOfAFunction(unittest.TestCase):

    def test_name_of_test_case(self):
        # define input value
        # define expected value
        # get actual result
        # define a message
        self.assertEqual(expected_result, actual_result, message)


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