Logo

Description automatically generated

**Assignment # 1**

**CSCS 351 (SQA)**

**Name: Joshua Naeem (22-10367)**

**Instructor: Dr. Saad Bin Saleem**

**TEST SUITE:**

My test suite named: (TestLibrary) consists of 5 test cases names are:

1. test\_availablebooks
2. test\_lendbook
3. test\_add\_book
4. test\_new\_book
5. test\_delete\_book

**Introduction**

This test suite will test 5 test cases mentioned above from a python file library.py using unittesting module of Python.

**Major functions/library used:**

* **import unittest**
* **from library import Library:**  this line tells the python interpreter to include the Library class from the library.py python file.
* **def setUp(self):** this function helps to initiate the class objects later being used in the test cases.

**The Tested Program (library.py):**

class Library:

    def \_\_init\_\_(self,listofbooks):

        self.listofbooks = listofbooks

    def get\_available\_books(self):

        return self.listofbooks

    def lendbook(self, requestedbook):

        if requestedbook in self.listofbooks:

            print("")

        else:

            print("Sorry the book you have requested is currently not in the library")

        return requestedbook

    def add\_book(self,returnedbook):

        return returnedbook

    def new\_book(self, new):

        return new

    def delete\_book(self, delete):

        return delete

**The Testing Program (test\_library.py):**

import unittest

from library import Library

class TestLibrary(unittest.TestCase):

    def setUp(self):

        self.c = Library(['Harry Potter','The Last Airbender', 'Multiverse of Madness'])

        self.c.lendbook('Harry Potter')

        self.c.add\_book('Garvin Adventures')

        self.c.new\_book('James Harold')

        self.c.delete\_book('Harry Potter')

    #test\_case\_1

    def test\_availablebooks(self):

        print('test\_availablebooks')

        self.assertEqual(self.c.get\_available\_books(), ['Harry Potter','The Last Airbender', 'Multiverse of Madness'])

    #test\_case\_2

    def test\_lendbook(self):

        print('test\_lendbook')

        self.assertEqual(self.c.lendbook('Harry Potter'), 'Harry Potter')

    #test\_case\_3

    def test\_add\_book(self):

        print('test\_add\_book')

        self.assertEqual(self.c.add\_book('Garvin Adventures'), 'Garvin Adventures')

    #test\_case\_4

    def test\_new\_book(self):

        print('test\_new\_book')

        self.assertEqual(self.c.new\_book('James Harold'), 'James Harold')

    #test\_case\_5

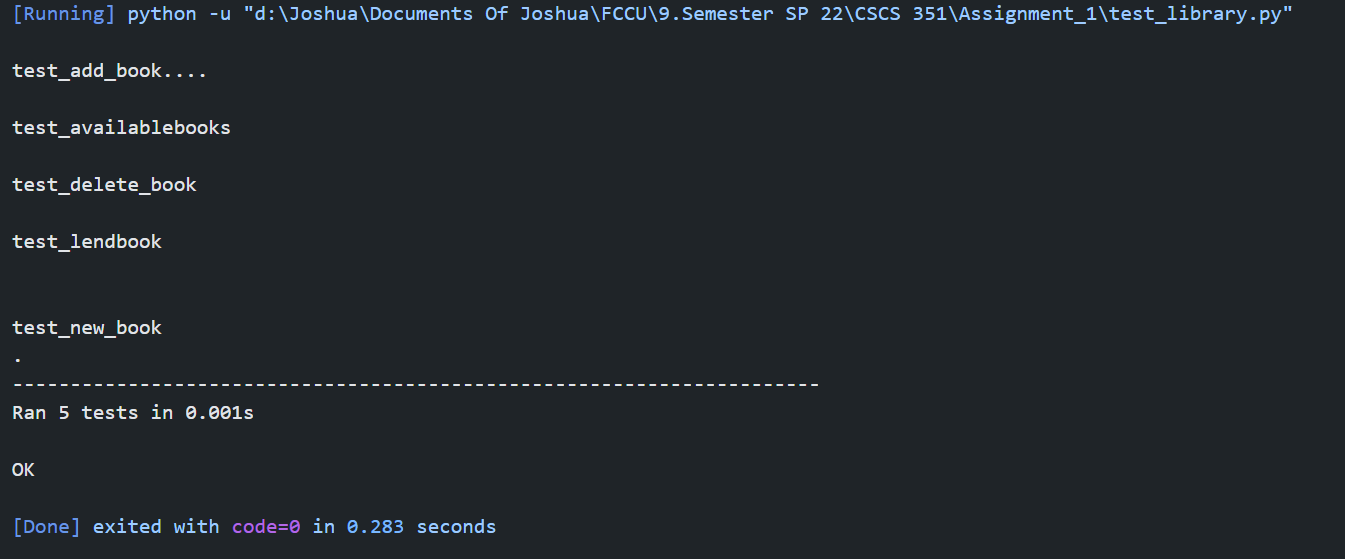
    def test\_delete\_book(self):

        print('test\_delete\_book')

        self.assertEqual(self.c.new\_book('Harry Potter'), 'Harry Potter')

if \_\_name\_\_ == '\_\_main\_\_':

    unittest.main()

**Output:**

**Results:**

5 out of 5 test cases are valid!