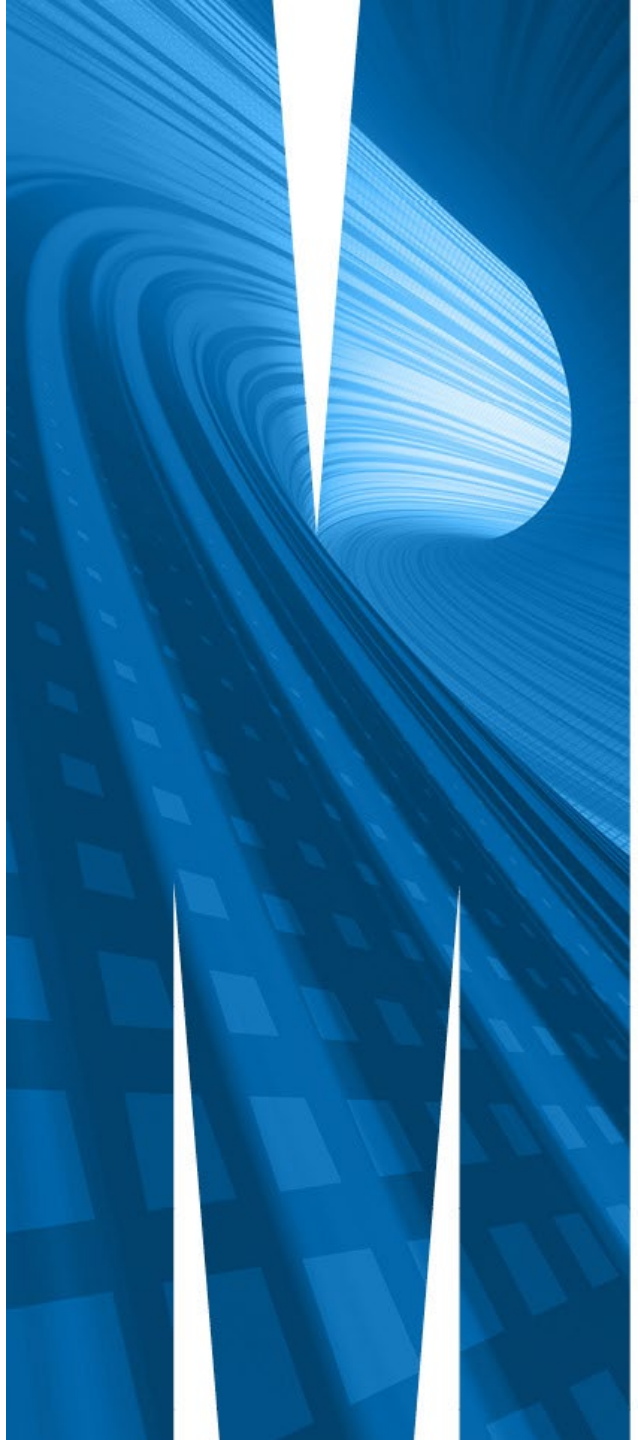
FIT2107-Software Quality & Testing

Lecture 7 – Unit Testing

15th September 2020

Dr Najam Nazar



Outline

- Part A
  - Introduction to Unit Testing
  - Automated Unit Testing using Python
- Part B
  - How to write unit tests in PyUnit
- Part C
  - Code Coverage in PyUnit.

PART A:
INTRODUCTION TO UNIT TESTING

Unit Test

- A unit test is a way to verify expected functionality in a *small*, *independent* bit of code.
- A test might focus on one method (function) in a class and extend it.
- What is unit?
  - OOP -> Class
- Combined and tested (Integration Testing)
- Companies expects coder to deliver production and Test code.
  - Ensure code you write is properly working
  - Students are wise to prepare for this

Automated Unit Testing

- Write code to test code.
- There are some common frameworks for automated united testing
  - JUnit for Java
  - PyUnit for Python
- IDE support is available.
  - Eclipse
  - PyCharm
  - VSCode
  - IDLE
- Write test code separate to the file in a production code.
  - Easy to ship code with out test code.
  - Only provide test code if it is in a contract.

Why automate tests?

- Why use PyUnit and not write our own framework?
  - We'd have to test harness that loads/runs tests,
  - Compares expected with actual output,
  - Shows messages for any failing tests, and
  - Summarises test results.
- This all would be time consuming and tedious.
- Why reinvent the wheel?

PART B:

HOW TO WRITE UNIT TESTS IN PYUNIT

REFER TO MY VIDEO ON HOW TO WRITE UNIT TESTS ON MOODLE.

Writing Test Cases

Sample Code: Production Code

- Let's say we have a python file called MyProgram.py
- Assume we have method called

```
def square(num):
  return num ** 2
```

- To be a good unit tester, we want to write some test cases for the square method.

Writing Test Cases

Sample Code: Test Code: Imports

- The first thing we need to do is import the unit test module
- We also need to import our production code to test

```
import unittest
Import MyProgram
```

- Now we need to setup the file creating test cases.
  - Some part of the code is confusing, it's just an object-oriented way of writing in Python
  - GoodNews: You can follow the same pattern every time; just fill in the blanks.

Writing Test Cases

Sample Code: Test Code: Setting up

- We need to set up a code “nest” in which the test cases will live
- Just under the import statements write this

```
class TestMyProgram(unittest.TestCase):
```

- You can also use other styles such as MyPyUnitTests; name should be meaningful.
- No reason to get fancy here; you use the same convention every time
  - With Test in the start of the name of class or at the end

Writing Test Cases

Sample Code: Test Code: Test Cases

- Test case should test one thing or few closely related thing
  - Don’t do too much
  - We want a failure to mean something; If a test fails you have a good idea where to look.
- A test for `square(num)` may look like this:

```
def test_square_int(self):
        self.assertEqual(9,MyProgram.square(3))
```

Orders of parameters isn’t important, but I use expected output and actual output order

- The method name should be meaningful and start with the test keyword in the start with a leading underscore.
- When the test runs
  - You will either get an ok message
  - or an error indicating the failure.

Writing Test Cases

Sample Code: Test Code: Test Cases

- After test definition we need to load the tests and run them

```
def main():
    suite = unittest.TestLoader().loadTestsFromTestCase(TestMyProgram)
    unittest.TextTestRunner(verbosity=2).run(suite)
```

- Type this exactly and replace the `TestMyProgram` with the name of your class.
- When we run main(), it will run all tests.

FOR MORE DETAILS WATCH THE VIDEOS

PART C:
CODE COVERAGE IN PYUNIT

Coverage

- Coverage.py is a third-party tool for measuring code coverages in Python programmes.
- It provides very nice command line and HTML output along with advanced features such as branch coverage.
- pip install coverage
- coverage run [Python File Name].py
- coverage report
- coverage html
- /html_cov
- https://coverage.readthedocs.io/en/coverage-5.2.1/

QUESTIONS???