

*'''*

*UserInterface.py*

*Created on Jan 23, 2014*

**@author:** *mokter*

*'''*

import unittest

#from app.UserInterface.UserInterface import AddReference

from app.UserInterface.AddReference import \*

#import AddReference

class **UserInterface**(unittest.TestCase):

*''' AutoGenerated Ref# '''*

count = 0

author = *"Hossain"*

year = 2014

def **testForSuccess**(*self*):

*''' this is a proven testCase*

*calc = AddReference()*

*result = calc.add(op1=5, op2=5)*

*self.assertEqual(result, 10, "Oops!! Addition failed")*

*'''*

addref = AddReference()

newcount = addref.addRef(count=10)

*self*.assertEqual(newcount, 11, *"Addition of new reference is failed"*)

#if (newcount == 11)

#print "Addition of new reference is succeeded"

if \_\_name\_\_ == *"\_\_main\_\_"*:

#import sys;sys.argv = ['', 'Test.testName']

unittest.main()

*'''*

*AddReference.py*

*Created on Jan 23, 2014*

**@author:** *mokter*

*'''*

class **AddReference**(object):

*'''*

*classdocs*

*'''*

def **add**(*self*, op1, op2):

return op1 + op2

def **keyGenerator**(*self*, name, date):

return (name + date)

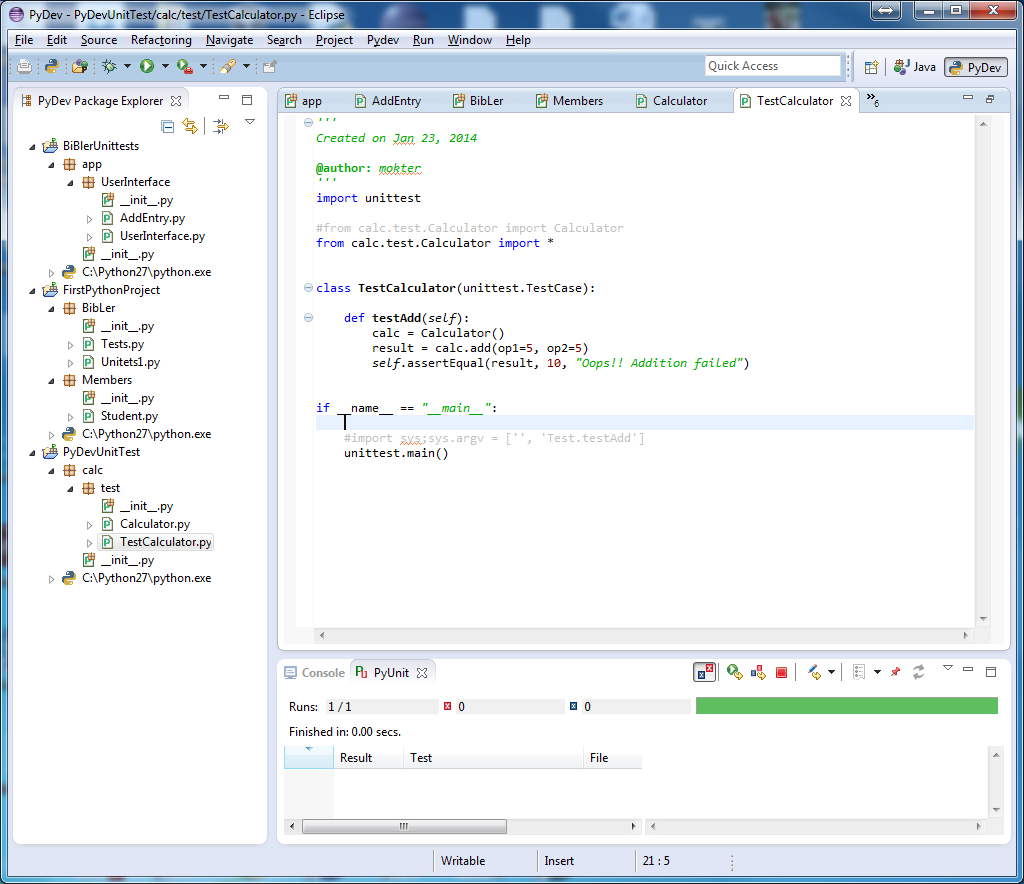
def **addRef**(*self*, count):

count = count + 1

return count

def **\_\_init\_\_**(*self*):

pass



*''' TestCalculator.py*

*Created on Jan 23, 2014*

**@author:** *mokter*

*'''*

import unittest

#from calc.test.Calculator import Calculator

from calc.test.Calculator import \*

class **TestCalculator**(unittest.TestCase):

def **testAdd**(*self*):

calc = Calculator()

result = calc.add(op1=5, op2=5)

*self*.assertEqual(result, 10, *"Oops!! Addition failed"*)

if \_\_name\_\_ == *"\_\_main\_\_"*:

#import sys;sys.argv = ['', 'Test.testAdd']

unittest.main()

*Calculator.py*

*'''*

*Created on Jan 23, 2014*

**@author:** *mokter*

*'''*

class **Calculator**(object):

*'''*

*classdocs*

*'''*

def **add**(*self*, op1, op2):

return op1 + op2

*'''*

*def \_\_init\_\_(self, params):*

*'''*

*'''*

*Constructor*

*'''*

*'''*

*Created on Jan 23, 2014*

**@author:** *mokter*

*'''*

import unittest

class **Test**(unittest.TestCase):

def **testAddition**(*self*):

*self*.assertEqual(1+1,2)

*self*.failUnlessEqual(1+1,2)

*self*.failIfEqual(1+1,1)

if \_\_name\_\_ == *"\_\_main\_\_"*:

#import sys;sys.argv = ['', 'Test.testAddition']

unittest.main()

*'''*

*Created on Jan 23, 2014*

**@author:** *mokter*

*'''*

class **Student**(object):

*'''*

*classdocs*

*'''*

def **\_\_init\_\_**(*self*, fname = *""*, lname = *""*):

*'''*

*Constructor*

*'''*

*self*.fname = fname

*self*.lname = lname

def **\_\_str\_\_**(*self*):

return *self*.fname + *" "* + *self*.lname

s1 = Student(*"Mokter"*, *"Hossain"*)

print s1

s2 = Student(*"Mahdi"*, *"Munim"*)

print s2

s3 = Student(*"Husain"*, *"Ergin"*)

print s3