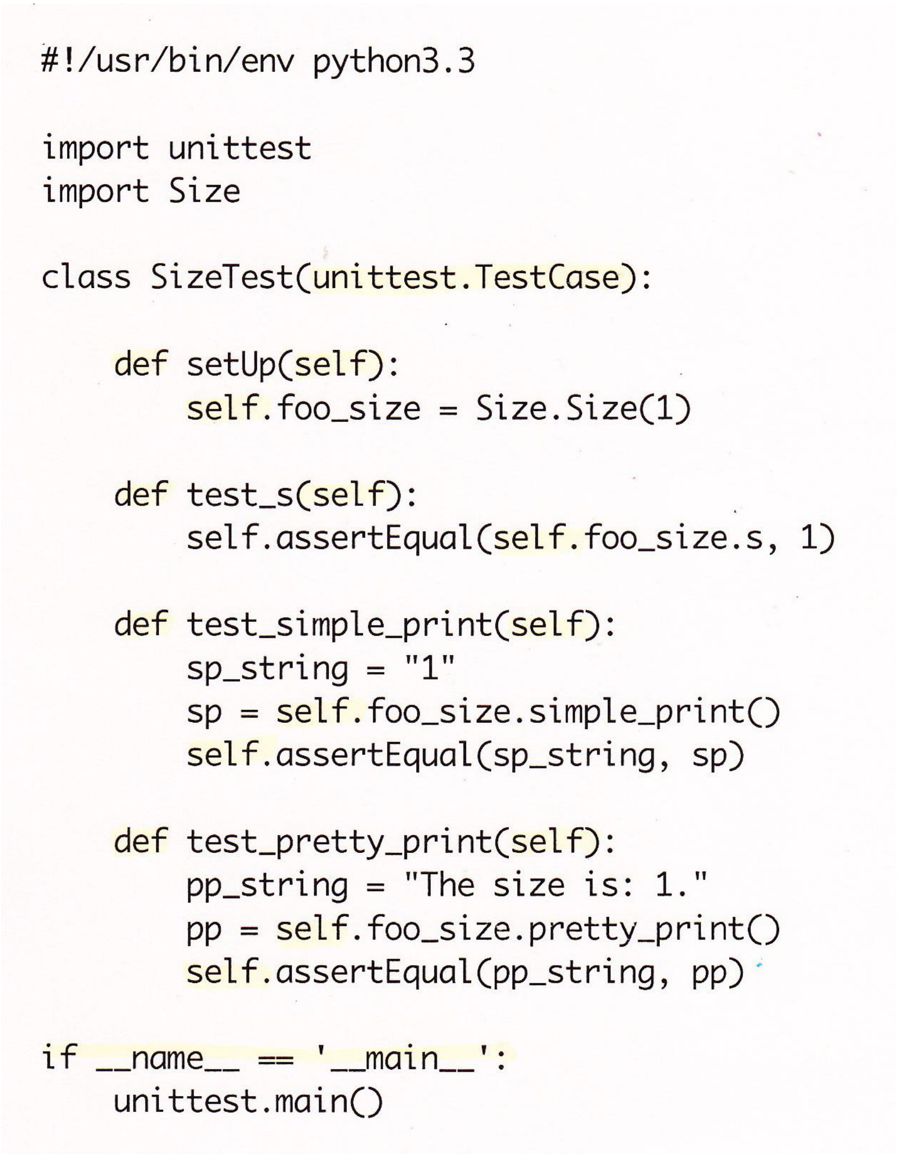
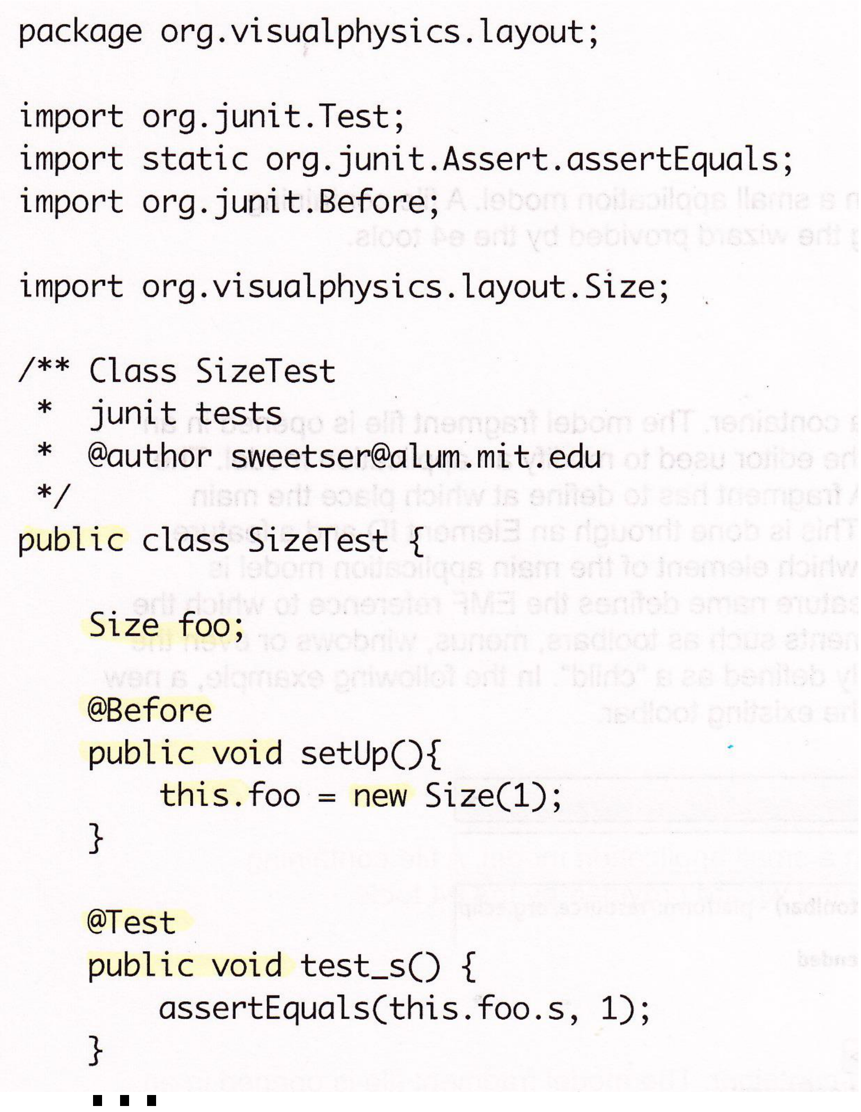
Python vs Java: Tests

def ...
test_x(self)
if '__name__' = '__main__'



```
#!/usr/bin/env python3.3

import unittest
import Size

class SizeTest(unittest.TestCase):

    def setUp(self):
        self.foo_size = Size.Size(1)

    def test_s(self):
        self.assertEqual(self.foo_size.s, 1)

    def test_simple_print(self):
        sp_string = "1"
        sp = self.foo_size.simple_print()
        self.assertEqual(sp_string, sp)

    def test_pretty_print(self):
        pp_string = "The size is: 1."
        pp = self.foo_size.pretty_print()
        self.assertEqual(pp_string, pp)

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

public void ...
@Test
Type variable
new Object()
this.method()
gradle test



```
package org.visualphysics.layout;

import org.junit.Test;
import static org.junit.Assert.assertEquals;
import org.junit.Before;

import org.visualphysics.layout.Size;

/** Class SizeTest
 *  junit tests
 *  @author sweetser@alum.mit.edu
 */
public class SizeTest {

    Size foo;

    @Before
    public void setUp(){
        this.foo = new Size(1);
    }

    @Test
    public void test_s() {
        assertEquals(this.foo.s, 1);
    }
    ...
```