

Friendly little hint about junit

Paul Robertson

Lets make a simple class

```
class VerySimple
{
    int mySquare (int x) { return x*x; }
}
```

```
C:\mit16_410_13>javac VerySimple.java

C:\mit16_410_13>dir VerySimple.*
Volume in drive C is C
Volume Serial Number is 68D9-78CE

Directory of C:\mit16_410_13

09/14/2005  09:03 AM                256 VerySimple.class
09/14/2005  08:59 AM                 84 VerySimple.java
                2 File(s)                340 bytes
                0 Dir(s) 14,464,000,000 bytes free

C:\mit16_410_13>java VerySimple
Exception in thread "main" java.lang.NoSuchMethodError:
main
C:\mit16_410_13>
```

Writing a test

```
class TestVerySimple {
    public static void main (String[] args) {
        VerySimple foo=new VerySimple();
        System.out.println(foo.mySquare(0));
    }
}
```

```
C:\mit16_410_13>javac TestVerySimple.java

C:\mit16_410_13>java TestVerySimple
0
```

Using Junit

```
import junit.framework.*;

public class JunitVerySimpleTest extends TestCase {
    VerySimple foo;

    public JunitVerySimpleTest (String name) { super(name); }

    protected void setUp () {
        foo=new VerySimple();
    }

    public void testVerySimple1() { assertEquals(foo.mySquare(0), 0); }
    public void testVerySimple2() { assertEquals(foo.mySquare(1), 1); }
    public void testVerySimple3() { assertEquals(foo.mySquare(2), 4); }
    public void testVerySimple4() { assertEquals(foo.mySquare(-1), 1); }
}
```

Invoking Junit

- To invoke Junit on your test program do any of the following:
 - For text output that you can cut and paste your output into a report:

```
java junit.textui.TestRunner MyTest
```
 - For graphical output:

```
Java junit.swingui.TestRunner MyTest
```

```
C:\mit16_410_13>java junit.textui.TestRunner JunitVerySimpleTest
....
Time: 0.01

OK (4 tests)

C:\mit16_410_13>
```

Insert a bug

```
public class VerySimple
{
    public int mySquare (int x) { return x*(x-1); }
}
```



A bug!

```
C:\mit16_410_13>java junit.textui.TestRunner JunitVerySimpleTest
..F.F.F
Time: 0
There were 3 failures:
1) testVerySimple2(JunitVerySimpleTest)junit.framework.AssertionFailedError:
  expected:<0> but was:<1>
   at junit.VerySimpleTest.testVerySimple2(JunitVerySimpleTest.java:13)
   at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
   at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
   at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
2) testVerySimple3(JunitVerySimpleTest)junit.framework.AssertionFailedError:
  expected:<2> but was:<4>
   at junit.VerySimpleTest.testVerySimple3(JunitVerySimpleTest.java:14)
   at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
   at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
   at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
3) testVerySimple4(JunitVerySimpleTest)junit.framework.AssertionFailedError:
  expected:<2> but was:<1>
   at junit.VerySimpleTest.testVerySimple4(JunitVerySimpleTest.java:15)
   at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
   at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
   at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)

FAILURES!!!
Tests run: 4, Failures: 3, Errors: 0

C:\mit16_410_13>
```

Hints

- Don't add bugs to a working program
remove them from a non-working program
- Try it – it's easy