

 KingSchlock Add files via upload

1 contributor

70 lines (61 sloc) | 1.71 KB

...

```
1  using NUnit;
2  using NUnit.Framework;
3  using ClockClass;
4
5  namespace ClockClass
6  {
7      [TestFixture()]
8      internal class ClockTests
9      {
10         Clock testClock;
11
12         [SetUp()]
13         public void Setup()
14         {
15             testClock = new Clock();
16         }
17
18         [Test()]
19         public void TestClockInitialize()
20         {
21             Assert.That(testClock.ReadClock(), Is.EqualTo("00:00:00"));
22         }
23
24         [Test()]
25         public void TestClockSecondIncrement()
26         {
27             testClock.IncrementClock();
28             Assert.That(testClock.ReadClock(), Is.EqualTo("00:00:01"));
29         }
30
31         [Test()]
32         public void TestClockMinuteIncrement() //Test for day and hour
```

```
33     {
34         for(int i = 0; i < 60; i++)
35         {
36             testClock.IncrementClock();
37         }
38         Assert.That(testClock.ReadClock(), Is.EqualTo("00:01:00"));
39     }
40
41     [Test()]
42     public void TestClockHourIncrement()
43     {
44         for(int i = 0; i < 3600; i++)
45         {
46             testClock.IncrementClock();
47         }
48         Assert.That(testClock.ReadClock(), Is.EqualTo("01:00:00"));
49     }
50
51     [Test()]
52     public void TestClockDayIncrement()
53     {
54         for (int i = 0; i < 86400; i++)
55         {
56             testClock.IncrementClock();
57         }
58         Assert.That(testClock.ReadClock(), Is.EqualTo("00:00:00"));
59     }
60
61     [Test()]
62     public void TestClockReset()
63     {
64         testClock.IncrementClock();
65         testClock.ResetClock();
66
67         Assert.That(testClock.ReadClock(), Is.EqualTo("00:00:00"));
68     }
69 }
70 }
```