```csharp
1  using CounterTask;
2  using System.Diagnostics.CodeAnalysis;
3
4  namespace TestClock
5  {
6      public class Tests
7      {
8          private Clock myClock;
9          [SetUp]
10         public void Setup()
11         {
12             myClock = new Clock();
13         }
14
15         [Test]
16         public void TestClockTick()
17         {
18             // Test clock tick less than 1 minute
19             for (int i = 0; i < 59; i++)
20             {
21                 myClock.Tick();
22             }
23             Assert.That(myClock.Hours, Is.EqualTo(0));
24             Assert.That(myClock.Minutes, Is.EqualTo(0));
25             Assert.That(myClock.Seconds, Is.EqualTo(59));
26
27             // Test clock tick when reaching over 1 minute
28             for (int i = 0; i < 60; i++)
29             {
30                 myClock.Tick();
31             }
32             Assert.That(myClock.Hours, Is.EqualTo((0)));
33             Assert.That(myClock.Minutes, Is.EqualTo(1));
34             Assert.That(myClock.Seconds, Is.EqualTo(59));
35
36             // Test clock tick when reaching over 1 hour
37             for (int i = 0; i < 3600; i++)
38             {
39                 myClock.Tick();
40             }
41             Assert.That(myClock.Hours, Is.EqualTo(1));
42             Assert.That(myClock.Minutes, Is.EqualTo(1));
43             Assert.That(myClock.Seconds, Is.EqualTo(59));
44
45             // Test clock tick when reaching over 24 hours
46             for (int i = 0; i < 3600*24; i++)
47             {
48                 myClock.Tick();
49             }
```

```
50              Assert.That(myClock.Hours, Is.EqualTo(1));
51              Assert.That(myClock.Minutes, Is.EqualTo(1));
52              Assert.That(myClock.Seconds, Is.EqualTo(59));
53
54              Assert.Pass();
55          }
56
57          [Test]
58          public void TestReset()
59          {
60              for (int i = 0; i < 3676; i++)
61              {
62                  myClock.Tick();
63              }
64
65              myClock.Reset();
66              Assert.That(myClock.Hours, Is.EqualTo(0));
67              Assert.That(myClock.Minutes, Is.EqualTo(0));
68              Assert.That(myClock.Seconds, Is.EqualTo(0));
69
70              Assert.Pass();
71          }
72
73          [Test]
74          public void TestTimeFormat()
75          {
76              for (int i = 0; i < 3676; i++)
77              {
78                  myClock.Tick();
79              }
80              string a = myClock.PrintTime();
81              Assert.That(a, Is.EqualTo("01:01:16"));
82
83              Assert.Pass();
84          }
85      }
86  }
```