```
1  using ClockApplication;
2
3  namespace ClockTest
4  {
5      public class Tests
6      {
7          private Clock clock;
8
9          [SetUp]
10         public void Setup()
11         {
12             clock = new Clock();
13         }
14
15         [Test]
16         public void ClockInitializes()
17         {
18             Clock ini = new Clock();
19             Assert.IsNotNull(ini);
20             Assert.AreEqual(0, ini.Seconds);
21             Assert.AreEqual(0, ini.Minutes);
22             Assert.AreEqual(0, ini.Hours);
23         }
24
25         [Test]
26         public void TestTickShouldIncrementSecondsByOne()
27         {
28             int initialSeconds = clock.Seconds;
29
30             clock.Tick();
31
32             Assert.AreEqual(initialSeconds + 1, clock.Seconds);
33         }
34
35         [Test]
36         public void                                                    ↵
             TestTickShouldIncrementMinutesByOneWhenSecondsReachSixty()
37         {
38             int initialMinutes = clock.Minutes;
39             for (int i = 0; i <= 60; i++)
40             {
41                 clock.Tick();
42             }
43
44             Assert.AreEqual(initialMinutes + 1, clock.Minutes);
45         }
46
47         [Test]
48         public void TestTickShouldIncrementHoursByOneWhenMinutesReachSixty  ↵
```

```
            ()
        {
            int initialHours = clock.Hours;
            for (int i = 0; i <= 3600; i++)
            {
                clock.Tick();
            }

            Assert.AreEqual(initialHours + 1, clock.Hours);
        }
    }
}
```