

Swinburne University of Technology

Faculty of Science, Engineering and Technology

COS20007 OBJECT ORIENTED PROGRAMMING

Semester 1-2024



3.1P Clock Class

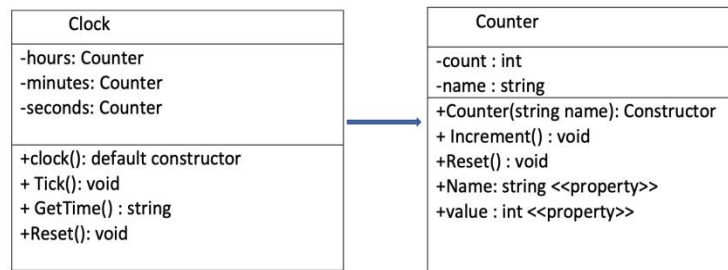
Program Class

```
using System;  
using ClockClass;
```

```
class Program
```

```
{  
    static void Main(string[] args)  
    {  
        Clock clock = new Clock();  
  
        for (int i = 0; i < 100; i++)  
        {  
            clock.Tick();  
            Console.WriteLine(clock.GetTime());  
        }  
  
        clock.Reset();  
        Console.WriteLine(clock.GetTime());  
  
        Console.ReadLine();  
    }  
}
```

Class UML diagram



Clock Class

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace ClockClass
{
    public class Clock
    {
        private Counter hoursCounter;
        private Counter minutesCounter;
        private Counter secondsCounter;

        public Clock()
        {
            hoursCounter = new Counter("hours");
            minutesCounter = new Counter("minutes");
            secondsCounter = new Counter("seconds");
        }

        public void Tick()
        {
            secondsCounter.Increment();
            if (secondsCounter.Ticks == 60)
            {
                secondsCounter.Reset();
                minutesCounter.Increment();
            }
            if (minutesCounter.Ticks == 60)
            {
                minutesCounter.Reset();
                hoursCounter.Increment();
            }
            if (hoursCounter.Ticks == 24)
            {
                hoursCounter.Reset();
            }
        }

        public void Reset()
        {
            hoursCounter.Reset();
            minutesCounter.Reset();
            secondsCounter.Reset();
        }
    }
}
```

```

        public string GetTime()
        {
            return $"{hoursCounter.Ticks:00}:{minutesCounter.Ticks:00}:{secondsCounter.Ticks:00}";
        }
    }
}

```

Counter Class

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

```

```

public class Counter
{
    private int _count;
    private string _name;
    public Counter(string name)
    {
        _count = 0;
        _name = name;
    }
    public int Increment()
    {
        _count++;
        return _count;
    }
    public int Reset()
    {
        _count = 0;
        return _count;
    }
    public string Name
    {
        get
        {
            return _name;
        }
        set
        {
            _name = value;
        }
    }
    public int Ticks
    {
        get
        {
            return _count;
        }
    }
}

```

Clock Test

```
using ClockClass;
```

```
namespace ClockClass;
```

```
public class ClockTests
```

```
{
```

```
    [Test]
```

```
    public void Clock_Initialises_At_Zero()
```

```
    {
```

```
        Clock clock = new Clock();
```

```
        string time = clock.GetTime();
```

```
        Assert.That(time, Is.EqualTo("00:00:00"));
```

```
    }
```

```
    [Test]
```

```
    public void Clock_Tick_Increases_Seconds_By_One()
```

```
    {
```

```
        Clock clock = new Clock();
```

```
        clock.Tick();
```

```
        string time = clock.GetTime();
```

```
        Assert.That(time, Is.EqualTo("00:00:01"));
```

```
    }
```

```
    [Test]
```

```
    public void Clock_Tick_60_Times_Increases_Minutes_By_One()
```

```
    {
```

```
        Clock clock = new Clock();
```

```
        for (int i = 0; i < 60; i++)
```

```
        {
```

```
            clock.Tick();
```

```
        }
```

```
        string time = clock.GetTime();
```

```
        Assert.That(time, Is.EqualTo("00:01:00"));
```

```
    }
```

[Test]

public void Clock_Tick_3600_Times_Increases_Hours_By_One()

{

 Clock clock = new Clock();

 for (int i = 0; i < 3600; i++)

 {

 clock.Tick();

 }

 string time = clock.GetTime();

 Assert.That(time, Is.EqualTo("01:00:00"));

}

[Test]

public void Clock_Reset_Sets_Time_To_Zero()

{

 Clock clock = new Clock();

 clock.Tick();

 clock.Tick();

 clock.Tick();

 clock.Reset();

 string time = clock.GetTime();

 Assert.That(time, Is.EqualTo("00:00:00"));

}

}

Counter Test

```
using NUnit.Framework;
using System.Diagnostics.Metrics;

namespace ClockClass;

public class CounterTests
{
    [Test]
    public void Counter_Initialises_At_Zero()
    {
        Counter counter = new Counter("testCounter");

        int count = counter.Ticks;

        Assert.That(count, Is.EqualTo(0));
    }

    [Test]
    public void Counter_Increment_Adds_One_To_Count()
    {
        Counter counter = new Counter("testCounter");
        counter.Increment();
        int count = counter.Ticks;

        Assert.That(count, Is.EqualTo(1));
    }

    [Test]
    public void Counter_Increment_Multiple_Times_Increases_Count_To_Match()
    {
        Counter counter = new Counter("testCounter");
        counter.Increment();
        counter.Increment();
        counter.Increment();
        int count = counter.Ticks;

        Assert.That(count, Is.EqualTo(3));
    }

    [Test]
    public void Counter_Reset_Sets_Count_To_Zero()
    {
        Counter counter = new Counter("testCounter");
        counter.Increment();
        counter.Increment();
        counter.Reset();
        int count = counter.Ticks;

        Assert.That(count, Is.EqualTo(0));
    }
}
```

Clock Output

FileEditViewGitProjectBuildDebugTestAnalyzeToolsExtensionsWindowHelp

Process: [16968] ClockClass.exe

Build, deploy, and develop faster by using an Re

Program.csClock.csCounter.cs

1234567891011121314151617

100 %

Autos

Search (Ctrl+E)

NameValue

AutosLocalsWatch1

C:\Users\shekh\3-T\OOP\3.1P

00:01:13
00:01:14
00:01:15
00:01:16
00:01:17
00:01:18
00:01:19
00:01:20
00:01:21
00:01:22
00:01:23
00:01:24
00:01:25
00:01:26
00:01:27
00:01:28
00:01:29
00:01:30
00:01:31
00:01:32
00:01:33
00:01:34
00:01:35
00:01:36
00:01:37
00:01:38
00:01:39
00:01:40
00:00:00

0 references
0 references

00:01:13
00:01:14
00:01:15
00:01:16
00:01:17
00:01:18
00:01:19
00:01:20
00:01:21
00:01:22
00:01:23
00:01:24
00:01:25
00:01:26
00:01:27
00:01:28
00:01:29
00:01:30
00:01:31
00:01:32
00:01:33
00:01:34
00:01:35
00:01:36
00:01:37
00:01:38
00:01:39
00:01:40
00:00:00

No issues found

Call StackBreakpointsError

Test Results

ClockClass

using System;

Program

Test Explorer

Ready

Test

Duration

Traits

Er...

Group Summary

1

using System;

Program

990

0

0 Warnings

0 Errors

Test

Duration

Traits

Er...

ClockTest (5)

14 ms

ClockClass (5)

14 ms

ClockTests (5)

14 ms

Clock_Initialises_At_Zero

14 ms

Clock_Reset_Sets_Time_To_Zero

< 1 ms

Clock_Tick_3600_Times_Increas...

< 1 ms

Clock_Tick_60_Times_Increases_...

< 1 ms

Clock_Tick_Increases_Seconds_...

< 1 ms

CounterTest (4)

23 ms

ClockClass (4)

23 ms

CounterTests (4)

23 ms

Counter_Increment_Adds_One_...

23 ms

Counter_Increment_Multiple_Ti...

< 1 ms

Counter_Initialises_At_Zero

< 1 ms

Counter_Reset_Sets_Count_To_...

< 1 ms

Group Summary

ClockTest

Tests in group: 5

Total Duration: 14 ms

Outcomes

5 Passed