```
\dots7-working\11.1P - Clock in Another Language\Clock.cpp
```

```
1
```

```
1 #include <string>
 2 #include <sstream>
 3 #include <iomanip>
 5 using namespace std;
7 class Counter {
 8 private:
        int _count;
9
        string _name;
10
11
12 public:
        Counter(const string& name) : _count(0), _name(name) {}
13
14
       void Increment() {
15
16
            ++_count;
17
       }
18
19
       void Reset() {
20
            _{count} = 0;
        }
21
22
23
        string Name() const {
24
            return _name;
25
       }
26
       void SetName(const string& value) {
27
            _name = value;
29
       }
30
        int Ticks() const {
31
32
            return _count;
33
        }
34 };
35
36 class Clock {
37 private:
38
        Counter _seconds;
       Counter _minutes;
39
40
       Counter _hours;
41
42 public:
        Clock() : _seconds("seconds"), _minutes("minutes"), _hours("hours") {}
43
44
45
       void Tick() {
            _seconds.Increment();
46
            if (_seconds.Ticks() > 59) {
47
48
                _minutes.Increment();
49
                _seconds.Reset();
```

```
...7-working\11.1P - Clock in Another Language\Clock.cpp
```

```
if (_minutes.Ticks() > 59) {
50
                    _hours.Increment();
51
52
                    _minutes.Reset();
53
                    if (_hours.Ticks() > 23) {
                        Reset();
54
55
                    }
56
                }
            }
57
       }
58
59
60
       void Reset() {
            _seconds.Reset();
61
62
            _minutes.Reset();
63
            _hours.Reset();
64
       }
65
        string Time() const {
66
            ostringstream oss;
67
            oss << setw(2) << setfill('0') << _hours.Ticks() << ":"
68
                << setw(2) << setfill('0') << _minutes.Ticks() << ":"
69
                << setw(2) << setfill('0') << _seconds.Ticks();</pre>
70
71
            return oss.str();
72
       }
73 };
74
```

2