

SWINBURNE UNIVERSITY OF TECHNOLOGY

COS20007 OBJECT ORIENTED PROGRAMMING

---

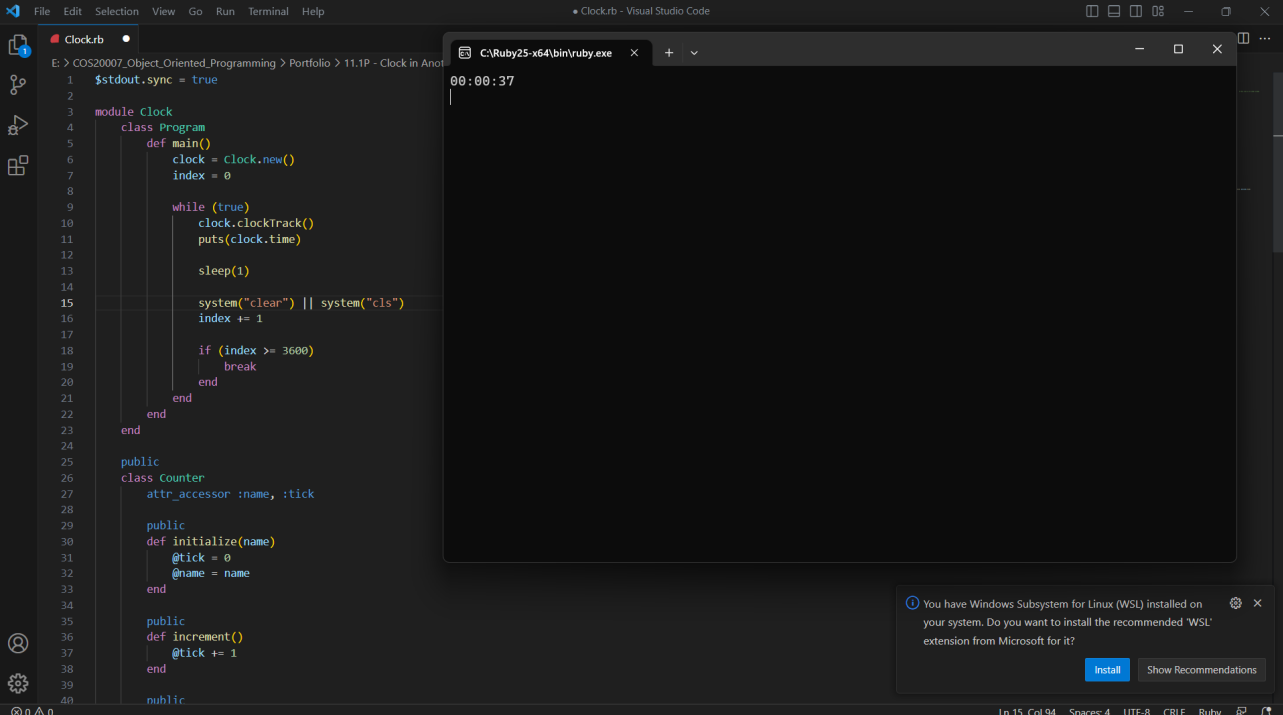
## 11.1P - Clock in Another Language

---

PDF generated at 00:07 on Monday 8<sup>th</sup> May, 2023

```
1 $stdout.sync = true
2
3 module Clock
4   class Program
5     def main()
6       clock = Clock.new()
7       index = 0
8
9       while (true)
10        clock.clockTrack()
11        puts(clock.time)
12
13        sleep(1)
14        system("clear") || system("cls")
15        index += 1
16
17        if (index >= 3600)
18          break
19        end
20      end
21    end
22  end
23
24  public
25  class Counter
26    attr_accessor :name, :tick
27
28    public
29    def initialize(name)
30      @tick = 0
31      @name = name
32    end
33
34    public
35    def increment()
36      @tick += 1
37    end
38
39    public
40    def reset()
41      @tick = 0
42    end
43  end
44
45  public
46  class Clock
47    def initialize()
48      @seconds = Counter.new("Seconds")
49      @minutes = Counter.new("Minutes")
50      @hours = Counter.new("Hours")
51    end
52
53    private
```

```
54     def trackSecondsValue()
55         if (@seconds.tick < 59)
56             @seconds.increment()
57         else
58             @seconds.reset()
59             @minutes.increment()
60         end
61     end
62
63     private
64     def trackMinutesValue()
65         if (@minutes.tick > 59)
66             @minutes.reset()
67             @hours.increment()
68         end
69     end
70
71     private
72     def trackHoursValue()
73         if (@hours.tick > 23)
74             @hours.reset()
75         end
76     end
77
78     public
79     def clockTrack()
80         trackSecondsValue()
81         trackMinutesValue()
82         trackHoursValue()
83     end
84
85     public
86     def time
87         format("%02d:%02d:%02d", @hours.tick, @minutes.tick, @seconds.tick)
88     end
89
90     public
91     def clockReset()
92         @seconds.reset()
93         @minutes.reset()
94         @hours.reset()
95     end
96 end
97 end
98
99 Clock::Program.new.main()
```



The screenshot shows the Visual Studio Code editor with a file named `Clock.rb` open. The code defines a `Clock` module with a `Program` class and a `Counter` class. The `Program` class has a `main` method that creates a `Clock` object, tracks time, and prints it every second. The `Counter` class has an `initialize` method and an `increment` method. The output window shows the time `00:00:37`. A notification at the bottom right suggests installing the WSL extension.

```
1 $stdout.sync = true
2
3 module Clock
4   class Program
5     def main()
6       clock = clock.new()
7       index = 0
8
9       while (true)
10        clock.clockTrack()
11        puts(clock.time)
12
13        sleep(1)
14
15        system("clear") || system("cls")
16        index += 1
17
18        if (index >= 3600)
19          break
20        end
21      end
22    end
23  end
24
25  public
26  class Counter
27    attr_accessor :name, :tick
28
29    public
30    def initialize(name)
31      @tick = 0
32      @name = name
33    end
34
35    public
36    def increment()
37      @tick += 1
38    end
39  end
40  public
```

00:00:37

You have Windows Subsystem for Linux (WSL) installed on your system. Do you want to install the recommended 'WSL' extension from Microsoft for it?

[Install](#) [Show Recommendations](#)

Ln 15, Col 94 Spaces: 4 UTF-8 CRLF Ruby