SWINBURNE UNIVERSITY OF TECHNOLOGY

COS20007 OBJECT ORIENTED PROGRAMMING

Clock in Another Language

PDF generated at 18:55 on Thursday 26th October, 2023

```python
1   class Counter:
2       def __init__(self, name):
3           self.name = name
4           self.count = 0
5
6       def increment(self):
7           # Increment the counter by 1
8           self.count += 1
9
10      def reset(self):
11          # Reset the counter to 0
12          self.count = 0
13
14  class Clock:
15      def __init__(self):
16          self.hours = Counter("hours")
17          self.minutes = Counter("minutes")
18          self.seconds = Counter("seconds")
19
20      def tick(self):
21          # Increment the seconds counter
22          self.seconds.increment()
23
24          # Check if seconds have reached 60
25          if self.seconds.count == 60:
26              # Reset seconds and increment minutes
27              self.seconds.reset()
28              self.minutes.increment()
29
30          # Check if minutes have reached 60
31          if self.minutes.count == 60:
32              # Reset minutes and increment hours
33              self.minutes.reset()
34              self.hours.increment()
35
36          # Check if hours have reached 24
37          if self.hours.count == 24:
38              # Reset the entire clock
39              self.reset_clock()
40
41      def reset_clock(self):
42          # Resetting the counters when 24 hours have passed
43          self.seconds.reset()
44          self.minutes.reset()
45          self.hours.reset()
46
47      def show_time(self):
48          # Format the time as HH:MM:SS
49          return
            ↪   f"{self.hours.count:02}:{self.minutes.count:02}:{self.seconds.count:02}"
50
51  if __name__ == "__main__":
52      clock = Clock()
```

```
53
54    for i in range(86500):
55        clock.tick()
56        print(f"Time (Press Enter space to Tick the clock): {clock.show_time()}")
57        input()
```