

A/ Clock.py

B/ Counter.py

C/ Program.py

A/ Clock.py

```
1  from Counter import Counter
2
3  class Clock:
4      def __init__(self):
5          self._seconds = Counter("seconds")
6          self._minutes = Counter("minutes")
7          self._hours = Counter("hours")
8
9      def Tick(self):
10         self._seconds.Increment()
11         if self._seconds.Ticks > 59:
12             self._minutes.Increment()
13             self._seconds.Reset()
14             if self._minutes.Ticks > 59:
15                 self._hours.Increment()
16                 self._minutes.Reset()
17                 if self._hours.Ticks > 23:
18                     self.Reset()
19
20     def Reset(self):
21         self._seconds.Reset()
22         self._minutes.Reset()
23         self._hours.Reset()
24
25     @property
26     def Time(self):
27         width = 2
28         return str(self._hours.Ticks).zfill(width) + ":" + str(self._minutes.Ticks).zfill(width) + ":" + str(self._seconds.Ticks).zfill(width)
29
```

B/ Counter.py

```
1  class Counter:
2      def __init__(self, name):
3          self._count = 0
4          self._name = name
5
6      def Increment(self):
7          self._count += 1
8
9      def Reset(self):
10         self._count = 0
11
12     @property
13     def Name(self):
14         return self._name
15
16     @Name.setter
17     def Name(self, value):
18         self._name = value
19
20     @property
21     def Ticks(self):
22         return self._count
```

C/ Program.py

```
1  import os
2  from time import sleep
3  from Clock import Clock
4
5  clock = Clock()
6
7  for x in range(86400):
8      clock.Tick()
9      print(clock.Time)
10     sleep(1)
11     os.system('clear')
```