

Counter.py

```
class Counter:  
    def __init__(self, name):  
        self._name = name  
        self._count = 0  
  
    def increment(self):  
        self._count += 1  
        return self._count  
  
    def reset(self):  
        self._count = 0  
        return self._count  
  
    @property  
    def tick(self):  
        return self._count  
  
    @property  
    def name(self):  
        return self._name  
  
    @name.setter  
    def name(self, value):  
        self._name = value
```

Program.py

```
from Clock import Clock

if __name__ == "__main__":
    clock = Clock()
    clock.start_clock(56)
```

Clock.py

```
import time
from Counter import Counter

class Clock:
    def __init__(self):
        self._hours = Counter("Hours")
        self._minutes = Counter("Minutes")
        self._seconds = Counter("Seconds")

    @property
    def hours(self):
        return self._hours.tick

    @property
    def minutes(self):
        return self._minutes.tick

    @property
    def seconds(self):
        return self._seconds.tick

    def tick(self):
        if self._seconds.tick < 59:
            self._seconds.increment()
        elif self._minutes.tick < 59:
            self._seconds.reset()
            self._minutes.increment()
        elif self._hours.tick < 23:
            self._seconds.reset()
            self._minutes.reset()
            self._hours.increment()
        else:
            self._hours.reset()
            self._minutes.reset()
            self._seconds.reset()

    def reset(self):
        self._hours.reset()
        self._minutes.reset()
        self._seconds.reset()

    def print_time(self):
        current_time =
f"{self._hours.tick:02}:{self._minutes.tick:02}:{self._seconds.tick:02}"
        print(current_time)
        return current_time

    def start_clock(self, seconds):
        for _ in range(seconds):
            time.sleep(1)
            self.tick()
            self.print_time()
```

Output Screenshot

The screenshot shows a code editor interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** Week_11.
- Explorer:** Shows a folder named WEEK_11 containing files: _pycache_, 11.1P - Answer, \$Clock.txt, 11.1P - Clock in Another Language.pdf, Clock.pdf, Clock.py, Clock.txt, Counter.pdf, Counter.py, Counter.txt, Program.pdf, Program.py (selected), and Program.txt.
- Code Editor:** The active tab is Program.py, displaying the following code:

```
from Clock import Clock
if __name__ == "__main__":
    clock = Clock()
    clock.start_clock(56)
```
- Terminal:** The terminal tab shows the following output:

```
00:00:18
00:00:19
00:00:20
00:00:21
00:00:22
00:00:23
00:00:24
00:00:25
00:00:26
00:00:27
00:00:28
```
- Status Bar:** Includes icons for file operations, a search bar, and status text: Ln 5, Col 25, Spaces: 4, UTF-8, CRLF, Python 3.12.3 64-bit.