

HH:MM:SS und String Formatierung

1. Schreiben Sie eine Python Funktion `secToHHMMSS(sec)`, die Sekunden `sec` (Typ `integer`) in eine Darstellung der Form HH:MM:SS umwandelt und auf die Konsole druckt. Nutzen Sie dazu die Modulooperation und die Ganzzahldivision. Verwenden Sie *String Formatierung*, um die Ausgabe von `print()` in die Form HH:MM:SS zu bringen.
2. Testen Sie Ihre Funktion an mehreren Beispielen.

Beispiel

```
secToHHMMSS(70)
```

```
>>> 00:01:10
```

Solution:

```
1  #!/usr/bin/env python
2  # coding: utf-8
3  # <h1>Table of Contents<span class="tocSkip"></span></h1>
4  # <div class="toc"><ul class="toc-item"><li><span><a href="#Sekunden-zu-HH:MM:SS" data-toc-
   modified-id="Sekunden-zu-HH:MM:SS-1"><span class="toc-item-num">1&nbsp;&nbsp;&nbsp;</span>
   Sekunden zu HH:MM:SS</a></span></li></ul></div>
5  # ### Sekunden zu HH:MM:SS
6  def secToHHMMSS(seconds):
7      """
8      takes seconds and prints them in HH:MM:SS format.
9
10     :param seconds: int
11     """
12     seconds = int(seconds)
13     H = seconds // (60*60)
14     seconds = seconds % (60*60)
15     M = seconds // 60
16     seconds = seconds % 60
17     S = seconds
18     print("{:02d}:{:02d}:{:02d}".format(H,M,S))
19 # Alternatively:
20 def secToHHMMSS(seconds):
21     """
22     takes seconds and prints them in HH:MM:SS format.
23
24     :param seconds: int
25     """
26     seconds = int(seconds)
27     H = seconds // (60*60)
28     M = seconds // 60 - H*60
29     seconds = seconds % 60
30     S = seconds
31     print("{:02d}:{:02d}:{:02d}".format(H,M,S))
32 secToHHMMSS(1)
33 secToHHMMSS(70)
34 secToHHMMSS(60)
35 secToHHMMSS(60*60)
36 secToHHMMSS(60*60*24)
```