

Countdown timer program in Python

Using concepts discussed in previous topics, we will write a “Countdown Timer” that asks the user for a time in seconds, then counts down until we hit zero.

We need to import the “time” module and can use a function within the “time” module called the “sleep” function, which effectively puts the program to sleep for some time.

```
import time
time.sleep(3)      # <-- sleep function pauses for 3 secs
print("Times Up!")
```

When we execute this code, nothing happens for 3 seconds, then it prints the output.

To add a visual indication that something is happening:

```
import time
duration = int(input("Timer duration (in seconds)? "))

for x in range(0, duration): # <-- increments
    print(x)
    time.sleep(1)

print("Times Up!")
```

but this is counting in the wrong direction.

Reverse the countdown by changing the “for loop” to decrement from the user inputted value:

```
import time
duration = int(input("Timer duration (in seconds)? "))

for x in range(duration, 0, -1): # <-- decrements
    print(x)
    time.sleep(1)

print("Times Up!")
```

To change the format to a digital clock countdown appearance: 00:05:23

```
import time
duration = int(input("Timer duration (in seconds)? "))

for x in range(duration, 0, -1):
    seconds = x
    print(f"00:00:{seconds}")
    time.sleep(1)

print("Times Up!")
```

If we enter 70 seconds for the duration, it will show as 70. We don't want to see a number bigger than 60 so we can use the "modulus" feature to show remainder after dividing by 60.

```
import time
duration = int(input("Timer duration (in seconds)? "))

for x in range(duration, 0, -1):
    seconds = x % 60 # <-- modulus: only shows the remainder
    print(f"00:00:{seconds:02}") # <-- 2 digits, 0 padded
    time.sleep(1)

print("Times Up!")
```

We should assign minutes, hours and possibly days if the program required it.

```
import time
duration = int(input("Timer duration (in seconds)? "))

for x in range(duration, 0, -1):
    seconds = x % 60 # <-- modulus: only shows the remainder
    minutes = int(x / 60)
    # minutes = x // 60
    hours = int(x / 3600)
    print(f"{hours}:{minutes:02}:{seconds:02}")
    time.sleep(1)

print("Times Up!")
```

Advanced

A quick introduction of Python `sys.stdin` and `sys.stdout`

`sys.stdin` and `sys.stdout` are file objects which are used by the interpreter for standard input and standard output. Data received by `raw_input` comes from `stdin`, which is normally connects to the keyboard. `stdin` can be redirected so that input instead comes from a file. This can be very useful for automating scripts.

About `sys.stdout.write()` and `print()`

When you call the `print(obj)` function, it actually calls the `sys.stdout.write(obj + '\n')` like:

```
import sys
sys.stdout.write("hello" + "\n")
print("hello")
```

```
import time
import sys
duration = int(input("Timer duration (in seconds)? "))

for x in range(duration, 0, -1):
    seconds = x % 60 # <-- modulus: only shows the remainder
    minutes = int(x / 60)
    # minutes = x // 60
    hours = int(x / 3600)
    sys.stdout.write(f"\r{hours}:{minutes:02}:{seconds:02}")
    # \r - carriage return - overwrites previous words
    time.sleep(1)

print("\nTimes Up!") # <-- \n new line
```

Higher End Example using a Function:

In this example:

- The **countdown_timer** function takes number of seconds as an argument.
- Inside the function, a **while** loop is used to countdown from the specified number of seconds to zero.
- The **divmod** function is used to get the minutes and seconds from the remaining time.
- The **timer_format** variable is used to format and print the remaining time in **MM:SS** format.
- The **time.sleep(1)** function is used to pause the program for one second in each iteration.
- Countdown is displayed by overwriting previous line using carriage return (**\r**).
- The loop continues until the countdown reaches zero, and then the "Time's up!" message is printed.

```
import time
import sys

def countdown_timer():
    try:
        # Get user input for countdown duration in seconds
        duration = int(input("Enter countdown duration in
seconds: "))

        # Countdown timer
        while duration:
            mins, secs = divmod(duration, 60)
            timer_format = "{:02d}:{:02d}".format(mins, secs)
            sys.stdout.write("\rTime remaining:
{}".format(timer_format))
            sys.stdout.flush()
            time.sleep(1)
            duration -= 1

        print("\nTime's up!")

    except ValueError:
        print("Invalid input. Please enter a valid number of
seconds.")

# Run the countdown timer
countdown_timer()
```