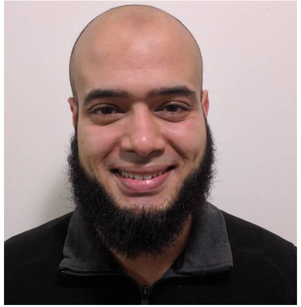
Python Programming

Time 1

Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

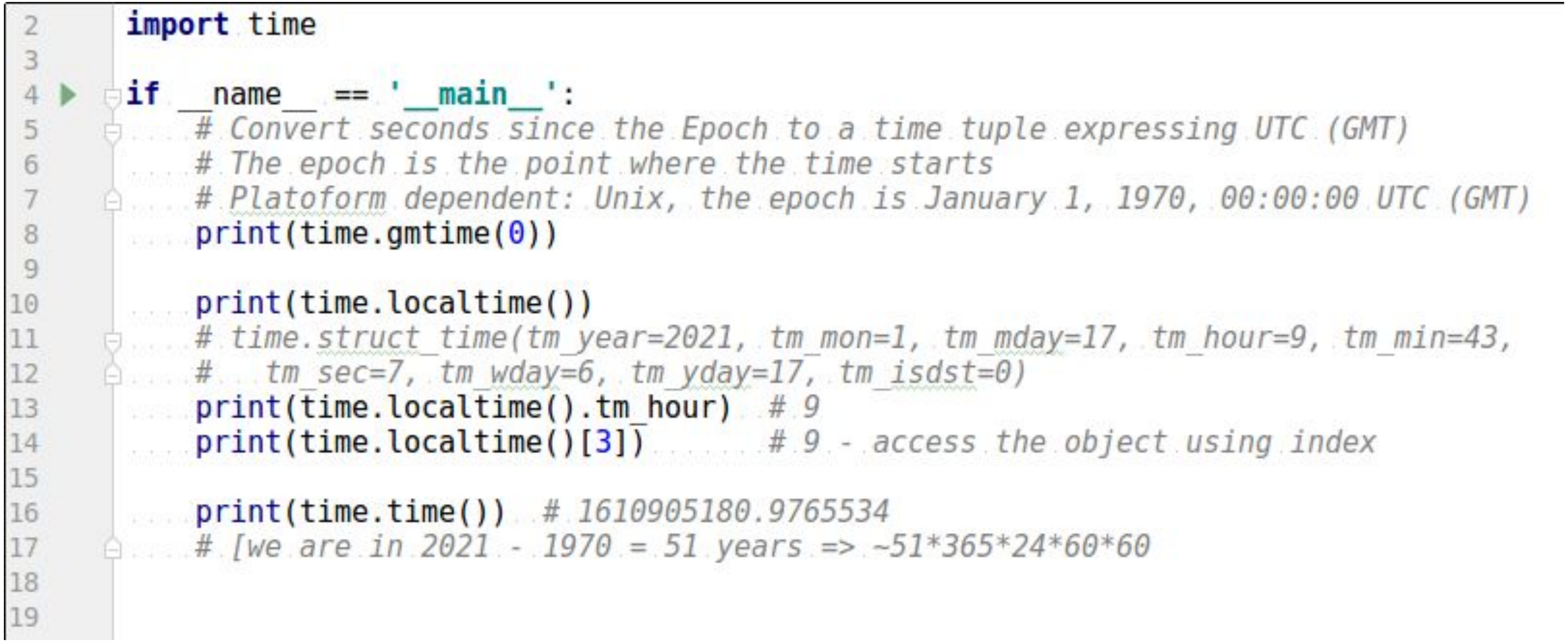
Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)



Time Module

- With time module, we can do several things
- Get localtime and its date/time components
- Compute time difference between points
  - Helps in **benchmarking**: assess the relative performance of a part of code / program
  - We have 3 ways to code a function. There is 2 3rd party packages to install. Which one?
- Printing functionalities for a flexible output
- The computed time is **relative**: from a starting point named **epoch**
  - Which is **platform dependent**

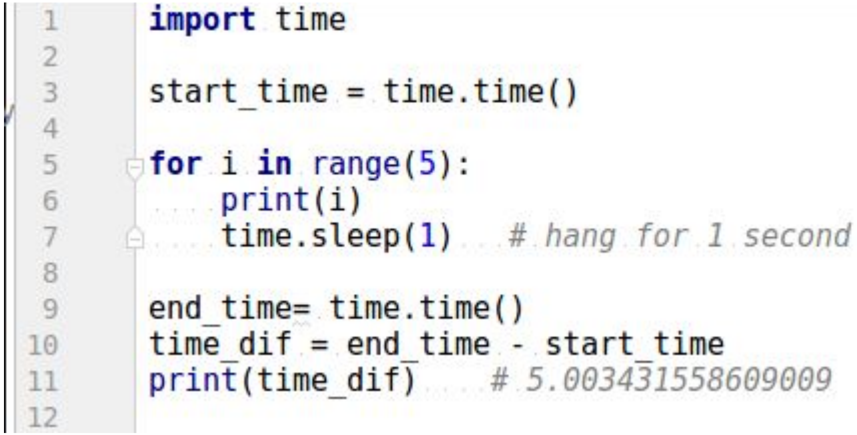
Time



```python
import time

if __name__ == '__main__':
    # Convert seconds since the Epoch to a time tuple expressing UTC (GMT)
    # The epoch is the point where the time starts
    # Platoform dependent: Unix, the epoch is January 1, 1970, 00:00:00 UTC (GMT)
    print(time.gmtime(0))

    print(time.localtime())
    # time.struct_time(tm_year=2021, tm_mon=1, tm_mday=17, tm_hour=9, tm_min=43,
    #   tm_sec=7, tm_wday=6, tm_yday=17, tm_isdst=0)
    print(time.localtime().tm_hour)  # 9
    print(time.localtime()[3])       # 9 - access the object using index

    print(time.time())  # 1610905180.9765534
    # [we are in 2021 - 1970 = 51 years => ~51*365*24*60*60


```

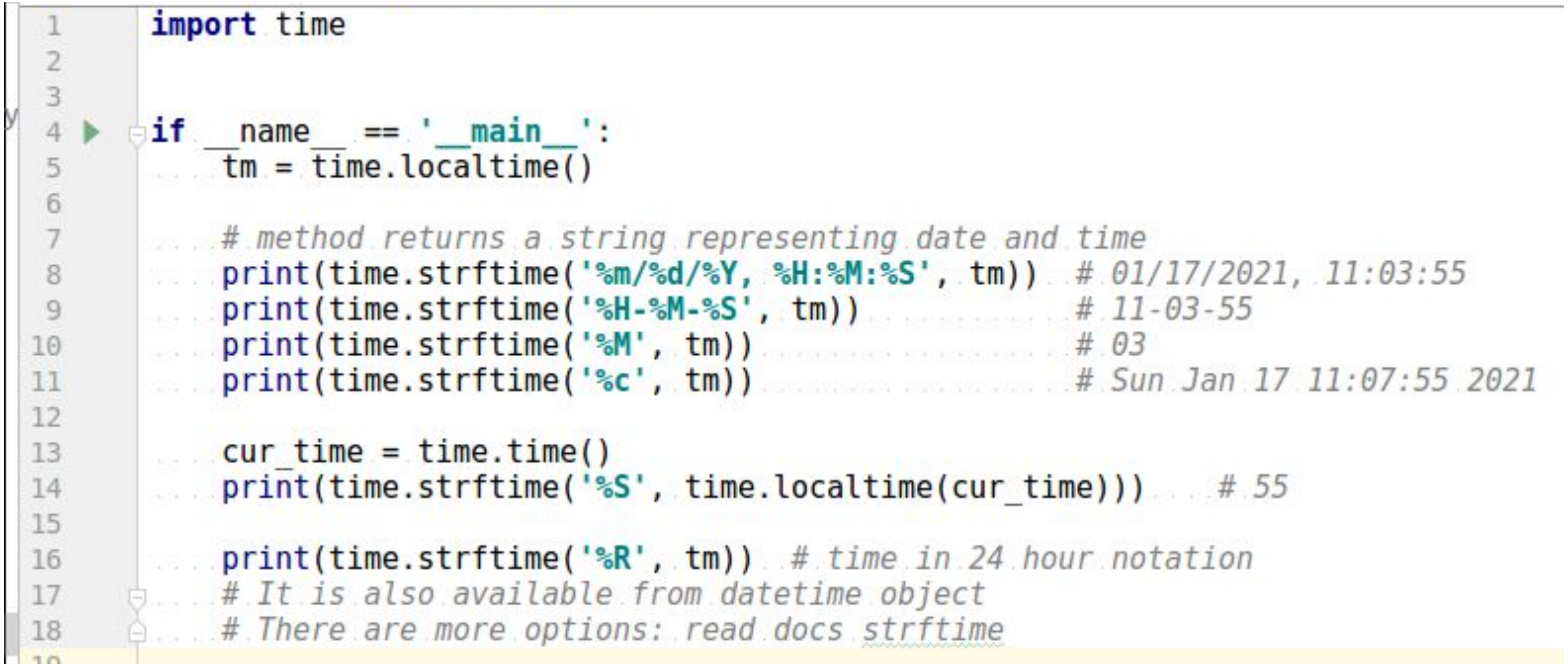
Sleep and time difference

- Sleeping is a common functionality in practice
- For example, each 5 minute check if there is an update in a specific web page



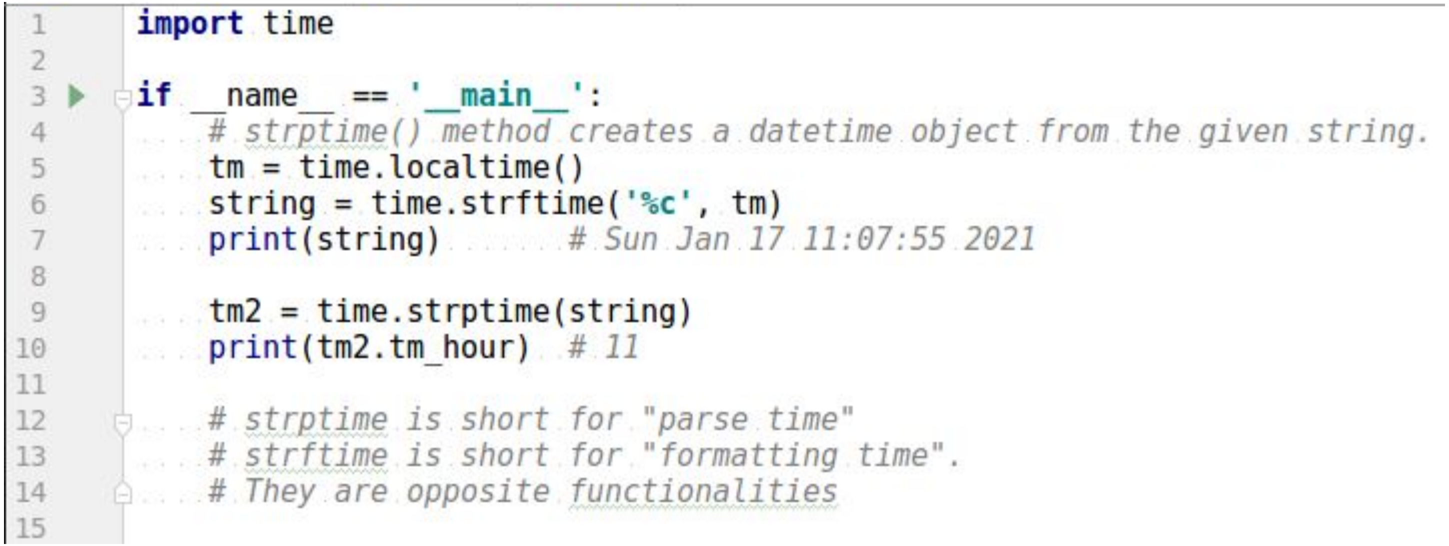
```python
import time

start_time = time.time()

for i in range(5):
    print(i)
    time.sleep(1)   # hang for 1 second

end_time= time.time()
time_dif = end_time - start_time
print(time_dif)    # 5.003431558609009

```

From date to a formatted string



```python
import time


if __name__ == '__main__':
    tm = time.localtime()

    # method returns a string representing date and time
    print(time.strftime('%m/%d/%Y, %H:%M:%S', tm))  # 01/17/2021, 11:03:55
    print(time.strftime('%H-%M-%S', tm))            # 11-03-55
    print(time.strftime('%M', tm))                  # 03
    print(time.strftime('%c', tm))                  # Sun Jan 17 11:07:55 2021

    cur_time = time.time()
    print(time.strftime('%S', time.localtime(cur_time)))    # 55

    print(time.strftime('%R', tm))  # time in 24 hour notation
    # It is also available from datetime object
    # There are more options: read docs strftime
```

From formatted string to a date



```python
import time

if __name__ == '__main__':
    # strptime() method creates a datetime object from the given string.
    tm = time.localtime()
    string = time.strftime('%c', tm)
    print(string)        # Sun Jan 17 11:07:55 2021

    tm2 = time.strptime(string)
    print(tm2.tm_hour)  # 11

    # strptime is short for "parse time"
    # strftime is short for "formatting time".
    # They are opposite functionalities
```

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."