PYTHON TIME MODULE

- Module name **time**
- **Import time**

Time Related Functions

1. **Python time.time() -** The time() function returns the number of seconds passed since epoch (the point where time begins).

2. **Python time.ctime() -** The time.ctime() function takes seconds passed since epoch as an argument and returns a string representing local time.

3. **Python time.sleep() -** The sleep() function suspends (delays) execution of the current thread for the given number of seconds.

Time.struct_time Class (Using the struct module)

- Gmtime(), asctime()

4. **Python time.localtime() -** The localtime() function takes the number of seconds passed since epoch as an argument and returns struct_time in local time.

5. **Python time.gmtime() -** The gmtime() function takes the number of seconds passed since epoch as an argument and returns struct_time in UTC.

6. **Python time.mktime() -** The mktime() function takes struct_time (or a tuple containing 9 elements corresponding to struct_time) as an argument and returns the seconds passed since epoch in local time. Basically, it's the inverse function of localtime().

7. **Python time.asctime() -** The asctime() function takes struct_time (or a tuple containing 9 elements corresponding to struct_time) as an argument and returns a string representing it.

8. **Python time.strftime() -** The strftime() function takes struct_time (or tuple corresponding to it) as an argument and returns a string representing it based on the format code used.

9. **Python time.strptime() -** The strptime() function parses a string representing time and returns struct_time.

#Create a digital clock in Python

```
import time
    while True:
        localtime=time.localtime()
        result=time.strftime("%I : %M : %S %p",localtime)
        print(result)
        time.sleep(1)
```