```python
import time

t = time.ctime()
print(t) # Mon Aug 30 11:00:18 2021

localtime = time.localtime()

# Abbreviated weekday name
result = time.strftime("%a", localtime)
print(result) # Monday

# Full weekday name
result = time.strftime("%A", localtime)
print(result) # Mon

# Full month name
result = time.strftime("%B", localtime)
print(result) # August

# Abbreviated month name
result = time.strftime("%b", localtime)
print(result) # Aug
```

```python
import time

localtime = time.localtime()

# Hour (24hr) possible values 00 - 23
result = time.strftime("%H", localtime)
print(result) # 11

# Hour (12hr) possible values 01 - 12
result = time.strftime("%I", localtime)
print(result) # 11

# Day of the year possible values 001 -366
result = time.strftime("%j", localtime)
print(result) # 242

# Month of the year possible values 01 - 12
result = time.strftime("%m", localtime)
print(result) # 08
```

```python
import time

localtime = time.localtime()

# %p either AM or PM
result = time.strftime("%p", localtime)
print(result) # AM

# %w Weekday as decimal number , values from 0 to 6
result = time.strftime("%w", localtime)
print(result) # 2

# %x for date representation
result = time.strftime("%x", localtime)
print(result) # 08/31/21

# %X for time representation
result = time.strftime("%X", localtime)
print(result) # 11:00:02

# %c date and time representation
result = time.strftime("%c", localtime)
print(result) # Tue Aug 31 11:00:55 2021

# %y year without date representation, values 00 -- 99
result = time.strftime("%y", localtime)
print(result) # 21

# %Y year with century as decimal number
result = time.strftime("%Y", localtime)
print(result) # 2021

# %Z Time Zone Name
result = time.strftime("%Z", localtime)
print(result) # India Standard Time
```

```python
import datetime

# datetime.date(year, month, day)Creates a local date
d = datetime.date(2021, 8, 20)
print(d) # 2021-08-20
print(d.year, d.month, d.day) # 2021 8 20

# Today date -- returns the locale date
print(d.today()) # 2021-08-31
```

```python
import datetime

t = datetime.datetime.today()
print(t.day, t.month, t.year)  # 31 8 2021
print(t.hour, t.minute, t.second) # 14 7 36
print(t.__class__)  # <class 'datetime.datetime'>
```

```python
# from datetime import  *
import datetime


#Set Year, Month, Day
d = datetime.date(2021, 8, 30)

#Set Hour, Minute
t = datetime.time(12,45)
dt = datetime.datetime.combine(d, t)
print(dt) # 2021-08-30 12:45:00
```

```
from datetime import date
import time

d = []

d1 = date(2016, 8, 12)
d2 = date(2017, 7, 12)
d3 = date(2017, 6, 12)
d4 = date(2017, 5, 12)

d.append(d1)
d.append(d2)
d.append(d3)
d.append(d4)

d.sort() # Sort in sequence order

for i in d:
    print(i)

Output
2016-08-12
2017-05-12
2017-06-12
2017-07-12
```

```
import time

class Clock:

    def d1(self):
        try:
            print("Starting Clock")
            while True:
                localtime = time.localtime()
                result = time.strftime("%I:%M:%S %p",
localtime)
                # %M minutes, %S seconds, %I Hour(12hr), %p
AM or PM
                print(result)
                time.sleep(1)
        except KeyboardInterrupt:
            print("Stopping Clock")

c = Clock()
c.d1()

Output
Starting Clock
02:26:55 PM
02:26:56 PM
02:26:57 PM
02:26:58 PM
Stopping Clock
```

```python
# Not to explain

import datetime

#Orginal date
t = datetime.datetime.today()
print(t)

#Replace date
start = t.replace(year=1991, month=4, day=13)
print(start)

end = datetime.datetime(year=2021, month=8, day=31)
print(end)

d = end - start
print(d.days)  # days is a property , 11097
```