# Python Datetime

A date in Python is not a data type of its own, but we can import a module named datetime to work with dates as date objects.

Import the datetime module and display the current date:

```
import datetime

x = datetime.datetime.now()

print(x)
```

## Date Output

When we execute the code from the example above the result will be:

2022-10-15 15:23:19.902238

The date contains year, month, day, hour, minute, second, and microsecond.

The datetime module has many methods to return information about the date object.

```
import datetime

x = datetime.datetime.now()

print(x.year)

print(x.strftime("%A"))
```

O/P : 2022

Saturday

```
import datetime

x = datetime.datetime(2018, 6, 1)

print(x.strftime("%B"))
```

O/P : June

**A reference of all the legal format codes:**

| Directive | Description | Example |
|---|---|---|
| %a | Weekday, short version | Wed |
| %A | Weekday, full version | Wednesday |
| %w | Weekday as a number 0-6, 0 is Sunday | 3 |
| %d | Day of month 01-31 | 31 |
| %b | Month name, short version | Dec |
| %B | Month name, full version | December |
| %m | Month as a number 01-12 | 12 |
| %y | Year, short version, without century | 18 |
| %Y | Year, full version | 2018 |
| %H | Hour 00-23 | 17 |
| %I | Hour 00-12 | 05 |

| | | |
|---|---|---|
| %p | AM/PM | PM |
| %M | Minute 00-59 | 41 |
| %S | Second 00-59 | 08 |
| %f | Microsecond 000000-999999 | 548513 |
| %z | UTC offset | +0100 |
| %Z | Timezone | CST |
| %j | Day number of year 001-366 | 365 |
| %U | Week number of year, Sunday as the first day of week, 00-53 | 52 |
| %W | Week number of year, Monday as the first day of week, 00-53 | 52 |
| %c | Local version of date and time | Mon Dec 31 17:41:00 2018 |
| %C | Century | 20 |

| | | |
|---|---|---|
| %x | Local version of date | 12/31/18 |
| %X | Local version of time | 17:41:00 |
| %% | A % character | % |
| %G | ISO 8601 year | 2018 |
| %u | ISO 8601 weekday (1-7) | 1 |
| %V | ISO 8601 weeknumber (01-53) | 01 |