Datetime

Dates

- 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.

Example

Import the datetime module and display the current date:

```python
import datetime

x = datetime.datetime.now()
print(x)
```

Output :

```
2021-04-29 10:42:06.734338
```

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

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

Example

```
import datetime

x = datetime.datetime.now()

print(x.year)
print(x.day)
print(x.month)
print(x.hour)
print(x.minute)
print(x.second)
print(x.microsecond)
```

Creating Date Objects

- To create a date, we can use the `datetime()` class (constructor) of the `datetime` module.
- The `datetime()` class requires three parameters to create a date: year, month, day.

Example

Create a date object:

```
import datetime
x = datetime.datetime(2020, 5, 17)
print(x)
```

The `datetime()` class also takes parameters for time and timezone (hour, minute, second, microsecond, tzone), but they are optional, and has a default value of `0`, (`None` for timezone).

The strftime() Method

- The `datetime` object has a method for formatting date objects into readable strings.
- The method is called `strftime()`, and takes one parameter, `format` , to specify the format of the returned string:

Example

Display the name of the month:

```
import datetime

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

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

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
%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

Creating Time Objects

- With a time object, we create a time, such as 3:45:12.
- The 3 is the hour. 45 is the minutes. And 12 is the seconds.
- So we'll show how to create a time object now in Python using the time class in the Python datetime module.
- To create a time object, we use the time class in the datetime module using the statement,

datetime.time(hour, minutes, seconds)

where hour is the hour, minutes is the minutes, and seconds is the seconds.

```
import datetime

time1= datetime.time(3,45,12)

print(time1)
03:45:12

time1.hour
3

time1.minute
45

time1.second
12
```