Python Datetime

Python Dates

In Python, the date is not a data type, but we can work with the date objects by importing the module named with **datetime, time, and calendar**.

- Import the datetime module and display the current date:
- import datetime

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

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

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:

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

Directive	Description	Example
%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

- import datetime

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

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

Performing Date Arithmetic:

You can perform arithmetic operations on dates using timedelta.

```
# Adding 3 days to the current date
three_days_later = current_datetime + datetime.timedelta(days=3)
 print("Date 3 days later:", three_days_later)
```

Comparison of two dates

To compare two dates in Python, you can simply use comparison operators (<, <=, ==, >=, >) on datetime objects. The datetime module allows for straightforward comparison of dates and times.

Calculating Time Difference:

- To calculate the difference between two dates or times, use the subtraction operator.

- You can extract different components of the time difference, such as days, seconds, or microseconds, and use them according to your specific needs.

- We can create two datetime objects (date1 and date2) representing different dates and times.
- We can print the time difference in days, seconds, and microseconds.

The calendar module

- Python provides a calendar object that contains various methods to work with the calendars.

- We can print the calendar of a particular month of a given year

```
import calendar
cal = calendar.month(2023, 4)
 print ("Here is the calendar of Month 4 of Year 2023:")
print (cal)
```

- The prcal() method of calendar module is used to print the calendar of the entire year. The year of which the calendar is to be printed must be passed into this method.