**Date and Time**

In the Lives of Human beings dates and time are very important For e.g :

Employee's experience is counted based on his joining date and time.

His age is calulated based on birthday.

Bank people pay interest on depoist based on time elapsed.

Project release dates will disturb the sleep of most of the software developers.

So it is important to know how to work with date and time. Python provides 3 modules : 'datetime' , 'time' and 'calendar'

The datetime module consists of 4 important classess :

datetime - This class handles a combination of both date and time and it contains attributes like year,month,day,hour,minute,second ,microsecond and tzinfo

date - This class handles dates of Gregorian calender, without taking time zone into consideration . its attributes like year,month,and day

time - This class handles time assuming that every day has exactly 24 \* 60 \*60 seconds .its attributes :hour ,minute,second, microsecond and tzinfo

timedelta - The timedelta class is useful to handle the durations.the duration may be the difference between two date,time or datetime instances.

**The epoch**

The epoch is the point where the time starts .This point is taken as 0.0 hours of january 1st of the current year. it is possible to measure time in seconds since the epoch using time() function of time module.

​

*# A python program to measure the time in seconds since the epoch*

**import** time

epoch **=** time.time() *#since epoch*

print(epoch)

​

​

1644486199.6283157

The output shows that how much of time has gone since the beginning of the current year . since the output is much useful we can convert them into date and time using localtime() function of 'time' module.

localtime() function returns an object similar to C language structure by the name struct\_time , from this object it is possible to extract day,month,year, and all time details

epoch **=** time.localtime(time.time()) *#either pass the epoch value of previous code or call time.time() method*

print(epoch) *#c structure object of struct\_time*

​

day **=** epoch.tm\_mday

month **=** epoch.tm\_mon

year **=** epoch.tm\_year

​

hour **=** epoch.tm\_hour

mint **=** epoch.tm\_min

sec **=** epoch.tm\_sec

​

​

​

print("Current date is {} - {} - {}".format(day,month,year))

print("Current time is {} : {} : {} ".format(hour,mint,sec))

print("Time zone is ",epoch.tm\_zone)

time.struct\_time(tm\_year=2022, tm\_mon=2, tm\_mday=10, tm\_hour=15, tm\_min=14, tm\_sec=48, tm\_wday=3, tm\_yday=41, tm\_isdst=0)

Current date is 10 - 2 - 2022

Current time is 15 : 14 : 48

Time zone is India Standard Time

Here

index             Attributes                   Values

0               tm\_year                       4 digit year number 2016

1               tm\_mon                       range[1,12]

2               tm\_mday                       range[1,31]

3               tm\_hour                       range[0,23]

4               tm\_min                       range[0,59]

5               tm\_sec                       range[0,61] , including leap seconds

6               tm\_wday                       range[0,6] ,Monday is 0

7               tm\_yday                       range[1,366]

8               tm\_isdst                     [0,1,0r -1] , 0 = no DST , 1 = DST is in effect ,-1 = not know

                  tm\_zone                       timezone name

Another way of converting epoch time into date and time is by using ctime() function of 'time' module ,this function takes epoch time in seconds and returns corresponding date and time in string format

*#using ctime*

epoch **=** time.ctime(time.time())

print(epoch)

​

Thu Feb 10 15:17:15 2022

## Date and Time Now

​

The current date and time sown in our computer system can be known using the following functions

ctime() function of 'time' module

now() method of datetime class of datetime module

today() method of datetime class of datetime module

​

ctime() function returns corresponding date and time in string format.

​

now() - returns an object containing date and time information in any timezone, the time zone information should be provided to this

method , if the time zone is not provided then local time zone is considered like IST in india , GMT in UK etc

it returns datetime class object , we can access day,month etc from this object.

This method belongs to datetime module hence needs import

​

today() - This method belongs to datetime module and it returns the date and time information.

The same module contains another class by the name 'date' , this calss also contains

today method that return only date.

​

​

​

​

*#checking ctime*

**import** time

t **=** time.ctime()

print(t)

Mon Jun 21 11:40:06 2021

*#checking now*

*# from datetime import datetime*

**import** datetime

nowdate **=** datetime.datetime.now() *#datetime module has one class datetime itself and it has now method*

print(nowdate)

​

print('Date now is {}/{}/{}'.format(nowdate.day,nowdate.month,nowdate.year))

print('Time now is {}:{}:{}'.format(nowdate.hour,nowdate.minute,nowdate.second))

​

2021-11-30 11:49:53.046091

Date now is 30/11/2021

Time now is 11:49:53

*# Checking today*

**from** datetime **import** **\***

*#today of datetime class gives date and time*

tdm **=** datetime.today()

print("Todays date and time is ",tdm)

​

*#today of date class gives the date only*

td **=** date.today()

print('Todays date is',td)

Todays date and time is 2021-10-28 11:56:29.117789

Todays date is 2021-10-28

*# We can create datetime object and even replace its contents*

**from** datetime **import** **\***

d1 **=** datetime(year **=**2016,month**=**4,day**=**29,hour**=**15,minute **=**30,second**=**30)

print(d1)

​

*#change the year month*

dt2 **=** d1.replace(year **=** 2017,hour**=**20,month**=**10)

print(dt2)

2016-04-29 15:30:30

2017-10-29 20:30:30

## Fromatting Dates and Time

The contents of 'datetime' , 'date' and 'time' class objects can be formatted using strftime() method representing string format

of time. This method converts the objects into specified format and returns the formatted string.

​

​

Formatted code used by strftime method is

​

Format Code                                 Meaning                             Example

%a                                       weekday as abbrevated name             Sun,Mon ....

%A                                       weekday as full name                 Sunday,Monday...

%w                                       weekday as decimal, 0 - Sunday       0,1,...6

                                          and 6 is Saturday

%d                                       Day of the month with 0 padded       01,02....

%b                                       Month as abbrivated                   Jan, Feb...

%B                                       Month as full name                   January, febuary...

%m                                       Month with 0 padded                   01,02...12

%y                                       year with out century                 00,01,02......99

%Y                                       year with century as decimal no       0001,0002,........2002,2020,2021

%H                                       24 hour clock with 0 padded                 00,01.............23

%I                                       12 hour clock                         01,02............12

%p                                       either PM or AM                       AM ,PM

%M                                       minute as a zero padded               01,02..............59

%S                                       second as 0 padded                   01,02,03.........,59

%f                                       Microsecond , 0 padded on left       000000,000001,.......999999

%Z                                       Time Zone                             empty,UTC,EST,CST

%j                                       Day of year                           001,002,366

%U                                       Week number of the year             01,02,...........53

                                          (sunday as first day)

%W                                       Week number of the year

                                          (Monday as the first day of the week) 01,02............................53

%c                                         Appropriate date and time representation   Tue Aug 16 21:30:00 1988

%x                                       Appropriate date representation       08/16/88(None) , 08/16/1988(en\_US)

%X                                       Appropriate time representation         21:30:00 (en\_US)

%%                                         A single % character                 %

​

​

*#To display the date using strftime*

**from** datetime **import** **\***

​

*# get today's date*

td **=** date.today()

print(td)

​

s1 **=** td.strftime("%j")

print("Today is ",s1, 'th day of the current year')

​

*#find the week day name*

s2 **=** td.strftime("%a")

print('it is ',s2)

2022-02-10

Today is 041 th day of the current year

it is Thu

*# to display time using strftime*

**from** datetime **import** **\***

*#create time object*

dt **=** datetime.now()

print(dt)

*#display only time*

print('Current time: ',dt.strftime("%H:%M:%S"))

2022-02-10 15:26:54.554724

Current time: 15:26:54

*#Exapmple to read the date from the user and display using strftime*

​

**from** datetime **import** **\***

*#accept date,month and year through keyboard*

​

d,m,y **=** [int(x) **for** x **in** input("Enter the date:").split('/')]

​

*#store the date values into date object*

​

dt **=** date(y,m,d)

​

*#%w day number and %A full name*

print(dt.strftime('Day %w of the week.This is %A'))

​

​

Enter the date:10/02/2022

Day 4 of the week.This is Thursday

## Finding Durations using 'timedelta'

The 'timedelta' class of datetime module is useful to find the durations like difference between two dates or finding the date

after adding a perriod to an existing date. it is possible to know future dates or previous date using timedelta.

The timedelta class object is available in the following format

timedelta(days =0 , seconds =0,microseconds =0,milliseconds =0,minutes =0,hours =0,weeks =0)

​

All the arguments are optional and the default value is 0

​

Suppose we have date and time object as

​

d1 = datetime(2016,4,29,16,45,0)   #29th April 2016

​

and if wanted to know future date by adding 10days 12 hours ,20minutes and 10 seconds

​

period1 = timedelta(days =10,seconds =10,minutes=20,hours=12)

​

This duration should be added to d1 (d1+period1) to know the furture date

​

*#Finding future date and time*

**from** datetime **import** **\***

​

*#store the date values in date object*

d1 **=** datetime(2016,4,29,16,45,0)

​

*#define the duration using the timedelta object*

period1 **=** timedelta(days **=**10,seconds **=**19,minutes **=**20,hours **=**12)

​

*#add the duration to d1 and display*

​

print("The new date and time is",d1**+**period1)

The new date and time is 2016-05-10 05:05:19

*#program to display set of next 10 days*

**from** datetime **import** **\***

*#start with today*

d **=** date.today()

print(d)

​

*#add 1 to 9 days and get furture dates*

**for** x **in** range(1,10):

nextdate **=** d**+**timedelta(days **=**x)

print(nextdate)

2022-02-10

2022-02-11

2022-02-12

2022-02-13

2022-02-14

2022-02-15

2022-02-16

2022-02-17

2022-02-18

2022-02-19

#Comparing Two dates

It is possible to compare 2 date objects just like comparing 2 numbers

d1 == d2

d1 > d2

d1 < d2

*#program to accept the date and time of birth and finding out who is older*

​

**from** datetime **import** **\***

*#enter the date of birth and store it in date class object*

d1,m1,y1 **=** [int(x) **for** x **in** input('Enter the first person birth date(DD/MM/YY)').split('/')]

​

b1 **=** date(y1,m1,d1)

​

d2,m2,y2 **=** [int(x) **for** x **in** input('Enter the second person birth date(DD/MM/YY)').split('/')]

​

b2 **=** date(y2,m2,d2)

​

*#compare bot the date*

**if** b1 **==** b2:

print("Both are of equal age")

**elif** b1**>**b2:

print("Second person is older")

**else**:

print("The First person is older")

Enter the first person birth date(DD/MM/YY)10/1/1990

Enter the second person birth date(DD/MM/YY)10/2/1989

Second person is older

**Sorting Date**

We can sort the dates by adding them to a list and use sort method

**from** datetime **import** **\***

*#first take an empty lis*

​

group **=** []

​

*#add todays date first*

group.append(date.today())

​

*#create some more dates and add to the group*

d**=** date(2015,6,29)

group.append(d)

d **=** date(2014,6,30)

group.append(d)

​

*#add 25days to the date and add to the group*

group.append(d**+**timedelta(days **=** 25))

​

*#sort the dates*

group.sort()

​

*#display the sorted dates*

**for** d **in** group:

print(d)

2014-06-30

2014-07-25

2015-06-29

2021-08-25

**Stopping Execution temporily**

To stop execution of a program temporarily for a given amount of time we can use sleep() function of time module sleep(seconds)

**import** time,random

*#generate some random numbers*

**for** i **in** range(10):

*#generate the number in the range of 100 to 200*

num **=** random.randrange(100,200,5)

print(num)

*#suspend the execution for 3.5 seconds*

time.sleep(2.5)

110

120

140

175

185

100

100

125

145

185

## Calender Module

The calender module is useful to create calender of any month or year. This module is also useful to know whether the given

year is leap year or not.

The Month function of calender module can be used to display the calender of the specific month

Calender can be used to display entire year itself

​

calender(year,w = 2,l=1,c=6,m=3)

​

where w - width between 2 lines , default value = 2

l - represents blank lines between 2 rows , def value = 1

c - represents column wise space between 2 months def value = 6

m - represents number of months to be displayed in a row def value =3

​

All these values are optional , if not provided default values will be considered

*#Program to display month of an year*

**from** calendar **import** **\***

​

*#ask for month and year*

yy **=** int(input("Enter the year"))

mm **=** int(input("Enter the month"))

​

*#displya the month of the calendar*

​

str **=** month(yy,mm)

print(str)

Enter the year2021

Enter the month11

November 2021

Mo Tu We Th Fr Sa Su

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30

*# program to display the calendar for entire year*

**from** calendar **import** **\***

year **=** int(input("Enter the year"))

*# print(calendar(year,2,1,8,4))*

print(calendar(year))

Enter the year2022

2022

January February March

Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su

1 2 1 2 3 4 5 6 1 2 3 4 5 6

3 4 5 6 7 8 9 7 8 9 10 11 12 13 7 8 9 10 11 12 13

10 11 12 13 14 15 16 14 15 16 17 18 19 20 14 15 16 17 18 19 20

17 18 19 20 21 22 23 21 22 23 24 25 26 27 21 22 23 24 25 26 27

24 25 26 27 28 29 30 28 28 29 30 31

31

April May June

Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su

1 2 3 1 1 2 3 4 5

4 5 6 7 8 9 10 2 3 4 5 6 7 8 6 7 8 9 10 11 12

11 12 13 14 15 16 17 9 10 11 12 13 14 15 13 14 15 16 17 18 19

18 19 20 21 22 23 24 16 17 18 19 20 21 22 20 21 22 23 24 25 26

25 26 27 28 29 30 23 24 25 26 27 28 29 27 28 29 30

30 31

July August September

Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su

1 2 3 1 2 3 4 5 6 7 1 2 3 4

4 5 6 7 8 9 10 8 9 10 11 12 13 14 5 6 7 8 9 10 11

11 12 13 14 15 16 17 15 16 17 18 19 20 21 12 13 14 15 16 17 18

18 19 20 21 22 23 24 22 23 24 25 26 27 28 19 20 21 22 23 24 25

25 26 27 28 29 30 31 29 30 31 26 27 28 29 30

October November December

Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su

1 2 1 2 3 4 5 6 1 2 3 4

3 4 5 6 7 8 9 7 8 9 10 11 12 13 5 6 7 8 9 10 11

10 11 12 13 14 15 16 14 15 16 17 18 19 20 12 13 14 15 16 17 18

17 18 19 20 21 22 23 21 22 23 24 25 26 27 19 20 21 22 23 24 25

24 25 26 27 28 29 30 28 29 30 26 27 28 29 30 31

31

*## Knowing the time taken by a program*

​

Python provides 2 functions **from** time module ,to know the time difference between 2 differnt points of the program.hence can be used to know the

time taken to execute the programm

​

perf\_counter() : This function returns the time time duration **in** fractional seconds it measures the time taken by the program

to execute group of statements . It includes the time elapsed during sleep of the processor

process\_time(): This function returns the time duraition **in** fractional seconds.it measures the total time taken by the program

**and** CPU **in** executing a group of statements . But it will **not** count the time elapsed during sleep of the processor

​

**from** time **import** **\***

​

*#Note down the starting time of the program*

t1 **=** perf\_counter()

​

*#do some action*

i,sum **=**0,0

**while**(i**<**10000000):

sum**+=**i

i**+=**1

*#make the processor to sleep for 3 seconds*

*# sleep(3) #try to comment and check the time once again*

​

*#note the ending of the time*

t2 **=** perf\_counter()

​

*#print the total time taken by finfding their difference*

print(f"Execution time is = {t2**-**t1} seconds")

​

​

Execution time is = 10.387016900000162 seconds

​