Python DateTime, TimeDelta, Strftime(Format) with Examples

While working with real-time applications, you have a lot to do with data and time. For example, an inventory management system you have to keep track of when a specific product was received and sold. You have to print statements at specific intervals. In all these tasks data and time are used.

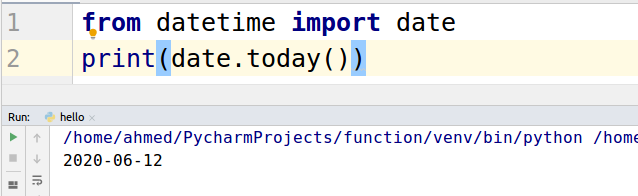
Python provides a ‘DateTime’ class under which 5 major classes are defined.

* **‘date’** – used to get just date ( Month, day, year)
* **‘time’** – Time independent of the day (Hour, minute, second, microsecond)
* **‘datetime’** – Both time and date (Month, day, year, hour, second, microsecond)
* **‘timedelta’**— A duration of time used for manipulating dates
* **‘tzinfo’**— An abstract class for dealing with time zones

# DateTime

Let’s take a look at how to get data and time. For using data and time we need to import the following.

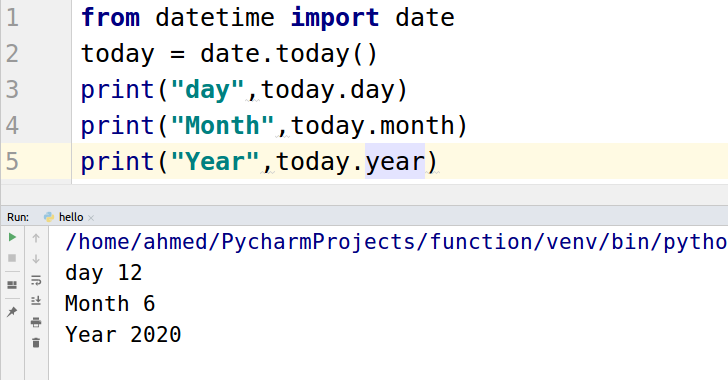
|  |
| --- |
| From datetime import date print(date.today()) |



We can access the year, month and the day separately. Let’s have a look at the example.

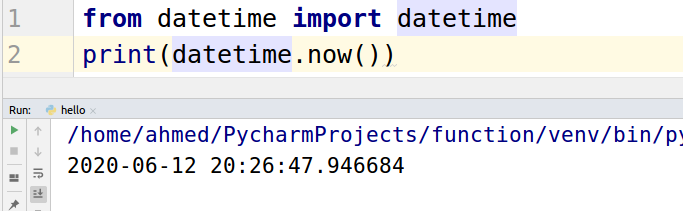
|  |
| --- |
| from datetime import date today = date.today() print("day",today.day) print("Month",today.month) print("Year",today.year) |

The following will be the output.



For the current date and time will use “datetime.now()” function.

|  |
| --- |
| from datetime import datetime print(datetime.now()) |



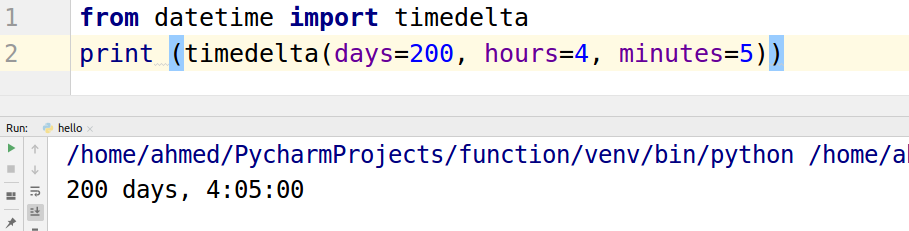
# TimeDelta

Python **timedelta()** function is present under *DateTime library* which is generally used for calculating differences in dates and also can be used for date manipulations in Python.

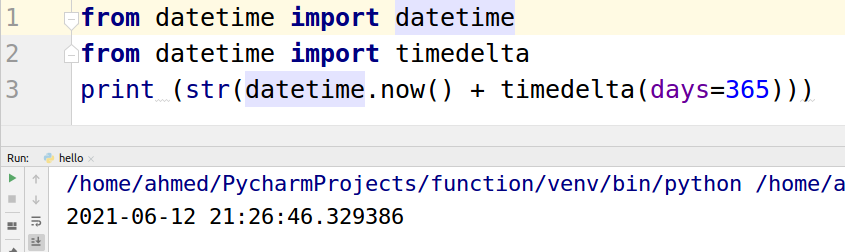
Let’s take a look at an example.

|  |
| --- |
| from datetime import timedelta print (timedelta(days=200, hours=4, minutes=5)) |

The output of the above code is shown below.



Let’s take a look at some practical example. Let’s print today date one year from now.

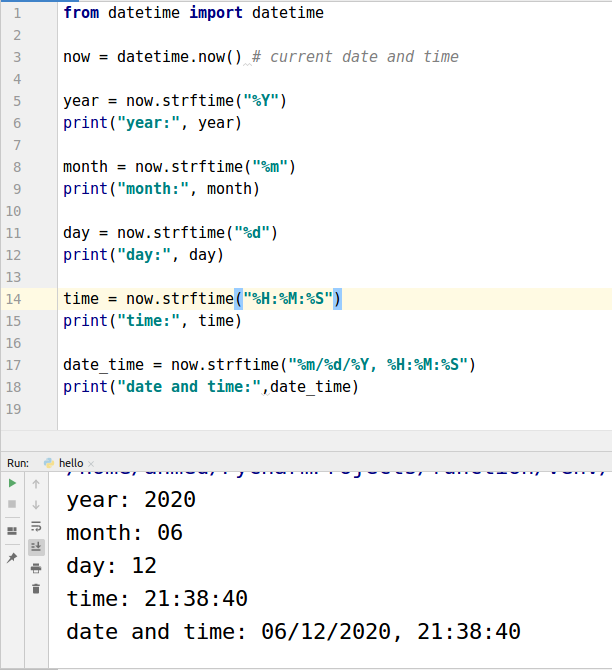


Strftime(Format)

The strftime() method is used to get a string representing date and time using date,time orDateTime object.

1. %C- indicates the local date and time
2. %x- indicates the local date
3. %X- indicates the local time
4. %Y- indicates the year
5. %m- indicates the month
6. %d- indicates the day

|  |
| --- |
| from datetime import datetime  now = datetime.now() # current date and time  year = now.strftime("%Y") print("year:", year)  month = now.strftime("%m") print("month:", month)  day = now.strftime("%d") print("day:", day)  time = now.strftime("%H:%M:%S") print("time:", time)  date\_time = now.strftime("%m/%d/%Y, %H:%M:%S") print("date and time:",date\_time) |



The "strftime function" can also be used to print time in any format 24 hours or 12 hours.

Let’s take a look at an example.

**12 hours** time is declared [print now.strftime("%I:%M”) ]

**24 hours** time is declared [print now.strftime("%H:%M")]

|  |
| --- |
| from datetime import datetime now = datetime.now() print("12 hours",now.strftime("%I:%M")) print("24 hours",now.strftime("%H:%M")) |

