1. What is datetime module?

The datetime module supplies for manipulating dates and times. While date and time arithmetic is supported, the focus of the implementation is on efficient attribute extraction for output formatting and manipulation.

2. How to use datetime module

2.1 date

In [2]:
```
# Import datetime library
import datetime

# Create date
d = datetime.date(2016, 7, 24)
print(d)
```

```
2016-07-24
```

In [7]:
```
# Create date with today
tday = datetime.date.today()
print(tday)

# Get day of tday
print(tday.day)

# Get weekday of tday
# Monday : 0, Sunday : 6
print(tday.weekday())
```

```
2022-10-14
14
4
```

2.2 timedelta

We can use timedelta ojbect for calcuating dates. The way to create timedelta object is adding or subtracting another date from a date.

- date2 = date1 + timedelta
- timedelta = date1 + date2
- timedelta = date1 - date2

In [10]:
```
tdelta = datetime.timedelta(days=7)

# Check the date seven days from now
print(tday + tdelta)

# Get total seconds of timedelta object
print(tdelta.total_seconds())
```

```
2022-10-21
604800.0
```

2.3 time

With time object in datetime class, we can work with hours, minutes, seconds.

In [12]:
```
t = datetime.time(9, 30, 45, 100000)
print(t)

# Get hour attribute from t
print(t.hour)
```

```
09:30:45.100000
9
```

2.4 datetime

With datetime object in datetime class, we can work with all attributes from date and time.

In [15]:
```
t = datetime.datetime(2016, 7, 26, 12, 30, 45, 100000)
print(t)

# Get time from datetime object
print(t.time())

# Get date from datetime object
print(t.date())
```

```
2016-07-26 12:30:45.100000
12:30:45.100000
2016-07-26
```

In [27]:
```
# Convert datetime into string object : datetime.strftime()
print(t.strftime('%B %d, %Y'))
```

```
July 26, 2016
```

In [29]:
```
# Convert string into datetime object
dt_str = 'July 26, 2016'
dt = datetime.datetime.strptime(dt_str, '%B %d, %Y')
print(dt)
```

```
2016-07-26 00:00:00
```

3. Timezone set in UTC

In [22]:
```
import datetime
import pytz

dt = datetime.datetime(2022, 10, 14, 19, 44, tzinfo=pytz.UTC)
dt_now = datetime.datetime.now(tz=pytz.UTC)
dt_utcnow = datetime.datetime.utcnow().replace(tzinfo=pytz.UTC)
dt_mtn = dt_utcnow.astimezone(pytz.timezone('US/Mountain'))

print(dt)
print(dt_now)
print(dt_utcnow)
print(dt_mtn)
```

```
2022-10-14 19:44:00+00:00
2022-10-14 10:47:28.101290+00:00
2022-10-14 10:47:28.101479+00:00
2022-10-14 04:47:28.101479-06:00
```

In [25]:
```
for tz in pytz.all_timezones[:5]:
    print(tz)
```

```
Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
Africa/Asmara
```

In [ ]:
```

```