datetime_module

November 14, 2019

```
[1]: import datetime
```

```
[2]: print(dir(datetime))
```

```
['MAXYEAR', 'MINYEAR', '__builtins__', '__cached__', '__doc__', '__file__',
'__loader__', '__name__', '__package__', '__spec__', 'date', 'datetime',
'datetime_CAPI', 'sys', 'time', 'timedelta', 'timezone', 'tzinfo']
```

```
[3]: print('datetime.MAXYEAR:', datetime.MAXYEAR)
     print('datetime.MINYEAR:', datetime.MINYEAR)
```

```
datetime.MAXYEAR: 9999
datetime.MINYEAR: 1
```

```
[4]: print('datetime.tzinfo:', datetime.tzinfo)
```

```
datetime.tzinfo: <class 'datetime.tzinfo'>
```

```
[5]: datetime.tzinfo()
```

```
[5]: <datetime.tzinfo at 0x4dc3320>
```

```
datetime  has three different kinds of objects:
    1. date object      :- stores the date
    2. time object      :- stores the time
    3. datetime object  :- stores both the date and time
```

```
[6]: now = datetime.datetime.now()
     print('now                    :', now)
     print('type(now)              :', type(now))
```

```
now                    : 2019-11-14 09:45:55.681539
type(now)              : <class 'datetime.datetime'>
```

```
[7]: today = now.date()
     print('today                        :', today)
     print('type(today)                  :', type(today))
```

```
today                        : 2019-11-14
type(today)                  : <class 'datetime.date'>
```

```
[8]: moment = now.time()
     print('moment                       :', moment)
     print('type(moment)                 :', type(moment))
```

```
moment                       : 09:45:55.681539
type(moment)                 : <class 'datetime.time'>
```

```
[9]: now_using_combine = datetime.datetime.combine(today, moment)
     print('now_using_combine        :', now_using_combine)
     print('type(now_using_combine)  :', type(now_using_combine))
```

```
now_using_combine        : 2019-11-14 09:45:55.681539
type(now_using_combine)  : <class 'datetime.datetime'>
```

```
[10]: yesterday = today - datetime.timedelta(days=1)
      print('yesterday                    :', yesterday)
      print('type(yesterday)              :', type(yesterday))
```

```
yesterday                    : 2019-11-13
type(yesterday)              : <class 'datetime.date'>
```

```
[11]: # Getting the difference between two datetime objects
      time_difference = yesterday - today
      print('time_difference              :', time_difference)
      print('type(time_difference)        :', type(time_difference))
```

```
time_difference              : -1 day, 0:00:00
type(time_difference)        : <class 'datetime.timedelta'>
```

```
[12]: # date object has three mandatory arguments
      my_date = datetime.date(1947, 8, 15)  # order YEAR, MONTH, DAY
      print('my_date                      :', my_date)
      print('type(my_date)                :', type(my_date))
```

```
my_date                      : 1947-08-15
type(my_date)                : <class 'datetime.date'>
```

```
[13]: my_date = datetime.date(day=15, year=1947, month=8)  # order can be changed␣
      ↪using keyword args
      print('my_date                  :', my_date)
      print('type(my_date)            :', type(my_date))
```

```
my_date                  : 1947-08-15
type(my_date)            : <class 'datetime.date'>
```

```
[14]: # time object don't have mandatory arguments
      my_time = datetime.time()
      print('my_time                  :', my_time)
      print('type(my_time)            :', type(my_time))
```

```
my_time                  : 00:00:00
type(my_time)            : <class 'datetime.time'>
```

```
[15]: my_time = datetime.time(0, 0)   # first argument hour, second minute
      print('my_time                  :', my_time)
      print('type(my_time)            :', type(my_time))
```

```
my_time                  : 00:00:00
type(my_time)            : <class 'datetime.time'>
```

```
[16]: my_time = datetime.time(hour=0, minute=0)
      print('my_time                  :', my_time)
      print('type(my_time)            :', type(my_time))
```

```
my_time                  : 00:00:00
type(my_time)            : <class 'datetime.time'>
```

```
[17]: # datetime objects have the same mandatory arguments as date object
      my_datetime = datetime.datetime(year=1984, month=6, day=24)  # Time is set to 0:
      ↪00
      print('my_datetime              :', my_datetime)
      print('type(my_datetime)        :', type(my_datetime))
```

```
my_datetime              : 1984-06-24 00:00:00
type(my_datetime)        : <class 'datetime.datetime'>
```

```
[18]: my_datetime = datetime.datetime(1984, 6, 24, 18, 30)
      print('my_datetime              :', my_datetime)
      print('type(my_datetime)        :', type(my_datetime))
```

```
my_datetime              : 1984-06-24 18:30:00
type(my_datetime)        : <class 'datetime.datetime'>
```

```
[19]: my_datetime = datetime.datetime(year=1984, month=6, day=24, hour=18, minute=30)
      print('my_datetime                :', my_datetime)
      print('type(my_datetime)          :', type(my_datetime))
```

```
my_datetime                : 1984-06-24 18:30:00
type(my_datetime)          : <class 'datetime.datetime'>
```

```
[20]: # Change one datetime object to obtain another using the replace method
      another_datetime = my_datetime.replace(year=2019, month=1)
      print('another_datetime           :', another_datetime)
      print('type(another_datetime)     :', type(another_datetime))
```

```
another_datetime           : 2019-01-24 18:30:00
type(another_datetime)     : <class 'datetime.datetime'>
```

```
[21]: # Obtain a datetime object representing the epoch: 1st Jan, 1970
      epoch = datetime.datetime.utcfromtimestamp(0)
      print('epoch                      :', epoch)
      print('type(epoch)                :', type(epoch))
```

```
epoch                      : 1970-01-01 00:00:00
type(epoch)                : <class 'datetime.datetime'>
```

```
[22]: # Obtain difference between epoch time and now
      delta = now - epoch
      print('now - epoch                :', delta)
```

```
now - epoch                : 18214 days, 9:45:55.681539
```

```
[23]: days = delta.days
      print('(now - epoch) diff days  :', days)
```

```
(now - epoch) diff days  : 18214
```

```
[24]: seconds = delta.seconds
      print('(now - epoch) diff secs  :', seconds)
```

```
(now - epoch) diff secs  : 35155
```

```
[25]: total_seconds = delta.total_seconds()
      print('(now - epoch)total_seconds:', total_seconds)
```

```
(now - epoch)total_seconds: 1573724755.681539
```

0.0.1 Time Information

```
[26]: local_now = datetime.datetime.now()
      print('local now: {}'.format(local_now))
```

```
local now: 2019-11-14 09:54:58.769246
```

```
[27]: utc_now = datetime.datetime.utcnow()
      print('utc now  : {}'.format(utc_now))
```

```
utc now  : 2019-11-14 04:25:15.902763
```

```
[28]: # You can access any value separately:
      print('{} {} {} {} {} {}'.format(local_now.year, local_now.month,
                                       local_now.day, local_now.hour,
                                       local_now.minute, local_now.second))

      print('date: {}'.format(local_now.date()))
      print('time: {}'.format(local_now.time()))
```

```
2019 11 14 9 54 58
date: 2019-11-14
time: 09:54:58.769246
```

```
datetime.strftime - For string formatting the datetime
For more datetime related string formatters, go to http://strftime.org/
```

```
[29]: local_now.strftime('%Y/%m/%d-%H:%M:%S')
```

```
[29]: '2019/11/14-09:54:58'
```

```
[30]: local_now.strftime('%y/%m/%d-%H:%M:%S')
```

```
[30]: '19/11/14-09:54:58'
```

```
[31]: local_now.strftime('%y %m %d-%H:%M:%S')
```

```
[31]: '19 11 14-09:54:58'
```

```
[32]: local_now.strftime('%y %m %d-%I:%M:%S %p')
```

```
[32]: '19 11 14-09:54:58 AM'
```

```
[35]: local_now.strftime('%Y %b %d %I:%M:%S %p')
```

```
[35]: '2019 Nov 14 09:54:58 AM'
```

```
[36]: formatted2 = local_now.strftime('date: %d %b,%Y time:%H:%M:%S')
      print('formatted2', formatted2)
```

```
formatted2 date: 14 Nov,2019 time:09:54:58
```

```
datetime.strptime() - For converting a datetime string into a datetime object
```

```
[41]: my_dt = datetime.datetime.strptime('2000-01-01 10:00:00', '%Y-%m-%d %H:%M:%S')
      print(f'my_dt        :{my_dt} \ntype(my_dt):{type(my_dt)}')
```

```
my_dt        :2000-01-01 10:00:00
type(my_dt):<class 'datetime.datetime'>
```

datetime.timedelta - For working with time difference.

```
[43]: tomorrow = local_now + datetime.timedelta(days=1)
      print('tomorrow this time: {}'.format(tomorrow))
```

```
tomorrow this time: 2019-11-15 09:54:58.769246
```

```
[44]: delta = tomorrow - local_now
      print(f'tomorrow - now : {delta}')
      print(f'days           : {delta.days}')
      print(f'seconds        : {delta.seconds}')
      print(f'total seconds  : {delta.total_seconds()}')
```

```
tomorrow - now : 1 day, 0:00:00
days           : 1
seconds        : 0
total seconds  : 86400.0
```

0.0.2 Working with time zones

```
[45]: import pytz
```

```
[47]: naive_utc_now = datetime.datetime.utcnow()
      print(f'naive utc now        : {naive_utc_now}')
      print(f'tzinfo               : {naive_utc_now.tzinfo}')
```

```
naive utc now        : 2019-11-14 04:36:36.895482
tzinfo               : None
```

```
[48]: # Localizing naive datetimes
      UTC_TZ = pytz.timezone('UTC')
```

```
[49]: utc_now = UTC_TZ.localize(naive_utc_now)
      print(f'utc now              : {utc_now}')
      print(f'tzinfo               : {utc_now.tzinfo}')
```

```
utc now              : 2019-11-14 04:36:36.895482+00:00
tzinfo               : UTC
```

```python
[50]: # Converting localized datetimes to different timezone
      PARIS_TZ = pytz.timezone('Europe/Paris')
```

```python
[52]: paris_now = PARIS_TZ.normalize(utc_now)
      print(f'Paris now              : {paris_now}')
      print(f'tzinfo                 : {paris_now.tzinfo}')
```

```
Paris now              : 2019-11-14 05:36:36.895482+01:00
tzinfo                 : Europe/Paris
```

```python
[53]: NEW_YORK_TZ = pytz.timezone('America/New_York')
      ny_now = NEW_YORK_TZ.normalize(utc_now)

      print(f'New York now           : {ny_now}')
      print(f'tzinfo                 : {ny_now.tzinfo}')
```

```
New York now           : 2019-11-13 23:36:36.895482-05:00
tzinfo                 : America/New_York
```

```python
[56]: for tz in pytz.all_timezones:
          print(tz)
```

```
Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
Africa/Asmara
Africa/Asmera
Africa/Bamako
Africa/Bangui
Africa/Banjul
Africa/Bissau
Africa/Blantyre
Africa/Brazzaville
Africa/Bujumbura
Africa/Cairo
Africa/Casablanca
Africa/Ceuta
Africa/Conakry
Africa/Dakar
Africa/Dar_es_Salaam
Africa/Djibouti
Africa/Douala
Africa/El_Aaiun
Africa/Freetown
Africa/Gaborone
Africa/Harare
```

```
Africa/Johannesburg
Africa/Juba
Africa/Kampala
Africa/Khartoum
Africa/Kigali
Africa/Kinshasa
Africa/Lagos
Africa/Libreville
Africa/Lome
Africa/Luanda
Africa/Lubumbashi
Africa/Lusaka
Africa/Malabo
Africa/Maputo
Africa/Maseru
Africa/Mbabane
Africa/Mogadishu
Africa/Monrovia
Africa/Nairobi
Africa/Ndjamena
Africa/Niamey
Africa/Nouakchott
Africa/Ouagadougou
Africa/Porto-Novo
Africa/Sao_Tome
Africa/Timbuktu
Africa/Tripoli
Africa/Tunis
Africa/Windhoek
America/Adak
America/Anchorage
America/Anguilla
America/Antigua
America/Araguaina
America/Argentina/Buenos_Aires
America/Argentina/Catamarca
America/Argentina/ComodRivadavia
America/Argentina/Cordoba
America/Argentina/Jujuy
America/Argentina/La_Rioja
America/Argentina/Mendoza
America/Argentina/Rio_Gallegos
America/Argentina/Salta
America/Argentina/San_Juan
America/Argentina/San_Luis
America/Argentina/Tucuman
America/Argentina/Ushuaia
America/Aruba
```

```
America/Asuncion
America/Atikokan
America/Atka
America/Bahia
America/Bahia_Banderas
America/Barbados
America/Belem
America/Belize
America/Blanc-Sablon
America/Boa_Vista
America/Bogota
America/Boise
America/Buenos_Aires
America/Cambridge_Bay
America/Campo_Grande
America/Cancun
America/Caracas
America/Catamarca
America/Cayenne
America/Cayman
America/Chicago
America/Chihuahua
America/Coral_Harbour
America/Cordoba
America/Costa_Rica
America/Creston
America/Cuiaba
America/Curacao
America/Danmarkshavn
America/Dawson
America/Dawson_Creek
America/Denver
America/Detroit
America/Dominica
America/Edmonton
America/Eirunepe
America/El_Salvador
America/Ensenada
America/Fort_Nelson
America/Fort_Wayne
America/Fortaleza
America/Glace_Bay
America/Godthab
America/Goose_Bay
America/Grand_Turk
America/Grenada
America/Guadeloupe
America/Guatemala
```

```
America/Guayaquil
America/Guyana
America/Halifax
America/Havana
America/Hermosillo
America/Indiana/Indianapolis
America/Indiana/Knox
America/Indiana/Marengo
America/Indiana/Petersburg
America/Indiana/Tell_City
America/Indiana/Vevay
America/Indiana/Vincennes
America/Indiana/Winamac
America/Indianapolis
America/Inuvik
America/Iqaluit
America/Jamaica
America/Jujuy
America/Juneau
America/Kentucky/Louisville
America/Kentucky/Monticello
America/Knox_IN
America/Kralendijk
America/La_Paz
America/Lima
America/Los_Angeles
America/Louisville
America/Lower_Princes
America/Maceio
America/Managua
America/Manaus
America/Marigot
America/Martinique
America/Matamoros
America/Mazatlan
America/Mendoza
America/Menominee
America/Merida
America/Metlakatla
America/Mexico_City
America/Miquelon
America/Moncton
America/Monterrey
America/Montevideo
America/Montreal
America/Montserrat
America/Nassau
America/New_York
```

```
America/Nipigon
America/Nome
America/Noronha
America/North_Dakota/Beulah
America/North_Dakota/Center
America/North_Dakota/New_Salem
America/Ojinaga
America/Panama
America/Pangnirtung
America/Paramaribo
America/Phoenix
America/Port-au-Prince
America/Port_of_Spain
America/Porto_Acre
America/Porto_Velho
America/Puerto_Rico
America/Punta_Arenas
America/Rainy_River
America/Rankin_Inlet
America/Recife
America/Regina
America/Resolute
America/Rio_Branco
America/Rosario
America/Santa_Isabel
America/Santarem
America/Santiago
America/Santo_Domingo
America/Sao_Paulo
America/Scoresbysund
America/Shiprock
America/Sitka
America/St_Barthelemy
America/St_Johns
America/St_Kitts
America/St_Lucia
America/St_Thomas
America/St_Vincent
America/Swift_Current
America/Tegucigalpa
America/Thule
America/Thunder_Bay
America/Tijuana
America/Toronto
America/Tortola
America/Vancouver
America/Virgin
America/Whitehorse
```

```
America/Winnipeg
America/Yakutat
America/Yellowknife
Antarctica/Casey
Antarctica/Davis
Antarctica/DumontDUrville
Antarctica/Macquarie
Antarctica/Mawson
Antarctica/McMurdo
Antarctica/Palmer
Antarctica/Rothera
Antarctica/South_Pole
Antarctica/Syowa
Antarctica/Troll
Antarctica/Vostok
Arctic/Longyearbyen
Asia/Aden
Asia/Almaty
Asia/Amman
Asia/Anadyr
Asia/Aqtau
Asia/Aqtobe
Asia/Ashgabat
Asia/Ashkhabad
Asia/Atyrau
Asia/Baghdad
Asia/Bahrain
Asia/Baku
Asia/Bangkok
Asia/Barnaul
Asia/Beirut
Asia/Bishkek
Asia/Brunei
Asia/Calcutta
Asia/Chita
Asia/Choibalsan
Asia/Chongqing
Asia/Chungking
Asia/Colombo
Asia/Dacca
Asia/Damascus
Asia/Dhaka
Asia/Dili
Asia/Dubai
Asia/Dushanbe
Asia/Famagusta
Asia/Gaza
Asia/Harbin
```

```
Asia/Hebron
Asia/Ho_Chi_Minh
Asia/Hong_Kong
Asia/Hovd
Asia/Irkutsk
Asia/Istanbul
Asia/Jakarta
Asia/Jayapura
Asia/Jerusalem
Asia/Kabul
Asia/Kamchatka
Asia/Karachi
Asia/Kashgar
Asia/Kathmandu
Asia/Katmandu
Asia/Khandyga
Asia/Kolkata
Asia/Krasnoyarsk
Asia/Kuala_Lumpur
Asia/Kuching
Asia/Kuwait
Asia/Macao
Asia/Macau
Asia/Magadan
Asia/Makassar
Asia/Manila
Asia/Muscat
Asia/Nicosia
Asia/Novokuznetsk
Asia/Novosibirsk
Asia/Omsk
Asia/Oral
Asia/Phnom_Penh
Asia/Pontianak
Asia/Pyongyang
Asia/Qatar
Asia/Qostanay
Asia/Qyzylorda
Asia/Rangoon
Asia/Riyadh
Asia/Saigon
Asia/Sakhalin
Asia/Samarkand
Asia/Seoul
Asia/Shanghai
Asia/Singapore
Asia/Srednekolymsk
Asia/Taipei
```

```
Asia/Tashkent
Asia/Tbilisi
Asia/Tehran
Asia/Tel_Aviv
Asia/Thimbu
Asia/Thimphu
Asia/Tokyo
Asia/Tomsk
Asia/Ujung_Pandang
Asia/Ulaanbaatar
Asia/Ulan_Bator
Asia/Urumqi
Asia/Ust-Nera
Asia/Vientiane
Asia/Vladivostok
Asia/Yakutsk
Asia/Yangon
Asia/Yekaterinburg
Asia/Yerevan
Atlantic/Azores
Atlantic/Bermuda
Atlantic/Canary
Atlantic/Cape_Verde
Atlantic/Faeroe
Atlantic/Faroe
Atlantic/Jan_Mayen
Atlantic/Madeira
Atlantic/Reykjavik
Atlantic/South_Georgia
Atlantic/St_Helena
Atlantic/Stanley
Australia/ACT
Australia/Adelaide
Australia/Brisbane
Australia/Broken_Hill
Australia/Canberra
Australia/Currie
Australia/Darwin
Australia/Eucla
Australia/Hobart
Australia/LHI
Australia/Lindeman
Australia/Lord_Howe
Australia/Melbourne
Australia/NSW
Australia/North
Australia/Perth
Australia/Queensland
```

```
Australia/South
Australia/Sydney
Australia/Tasmania
Australia/Victoria
Australia/West
Australia/Yancowinna
Brazil/Acre
Brazil/DeNoronha
Brazil/East
Brazil/West
CET
CST6CDT
Canada/Atlantic
Canada/Central
Canada/Eastern
Canada/Mountain
Canada/Newfoundland
Canada/Pacific
Canada/Saskatchewan
Canada/Yukon
Chile/Continental
Chile/EasterIsland
Cuba
EET
EST
EST5EDT
Egypt
Eire
Etc/GMT
Etc/GMT+0
Etc/GMT+1
Etc/GMT+10
Etc/GMT+11
Etc/GMT+12
Etc/GMT+2
Etc/GMT+3
Etc/GMT+4
Etc/GMT+5
Etc/GMT+6
Etc/GMT+7
Etc/GMT+8
Etc/GMT+9
Etc/GMT-0
Etc/GMT-1
Etc/GMT-10
Etc/GMT-11
Etc/GMT-12
Etc/GMT-13
```

```
Etc/GMT-14
Etc/GMT-2
Etc/GMT-3
Etc/GMT-4
Etc/GMT-5
Etc/GMT-6
Etc/GMT-7
Etc/GMT-8
Etc/GMT-9
Etc/GMT0
Etc/Greenwich
Etc/UCT
Etc/UTC
Etc/Universal
Etc/Zulu
Europe/Amsterdam
Europe/Andorra
Europe/Astrakhan
Europe/Athens
Europe/Belfast
Europe/Belgrade
Europe/Berlin
Europe/Bratislava
Europe/Brussels
Europe/Bucharest
Europe/Budapest
Europe/Busingen
Europe/Chisinau
Europe/Copenhagen
Europe/Dublin
Europe/Gibraltar
Europe/Guernsey
Europe/Helsinki
Europe/Isle_of_Man
Europe/Istanbul
Europe/Jersey
Europe/Kaliningrad
Europe/Kiev
Europe/Kirov
Europe/Lisbon
Europe/Ljubljana
Europe/London
Europe/Luxembourg
Europe/Madrid
Europe/Malta
Europe/Mariehamn
Europe/Minsk
Europe/Monaco
```

```
Europe/Moscow
Europe/Nicosia
Europe/Oslo
Europe/Paris
Europe/Podgorica
Europe/Prague
Europe/Riga
Europe/Rome
Europe/Samara
Europe/San_Marino
Europe/Sarajevo
Europe/Saratov
Europe/Simferopol
Europe/Skopje
Europe/Sofia
Europe/Stockholm
Europe/Tallinn
Europe/Tirane
Europe/Tiraspol
Europe/Ulyanovsk
Europe/Uzhgorod
Europe/Vaduz
Europe/Vatican
Europe/Vienna
Europe/Vilnius
Europe/Volgograd
Europe/Warsaw
Europe/Zagreb
Europe/Zaporozhye
Europe/Zurich
GB
GB-Eire
GMT
GMT+0
GMT-0
GMT0
Greenwich
HST
Hongkong
Iceland
Indian/Antananarivo
Indian/Chagos
Indian/Christmas
Indian/Cocos
Indian/Comoro
Indian/Kerguelen
Indian/Mahe
Indian/Maldives
```

```
Indian/Mauritius
Indian/Mayotte
Indian/Reunion
Iran
Israel
Jamaica
Japan
Kwajalein
Libya
MET
MST
MST7MDT
Mexico/BajaNorte
Mexico/BajaSur
Mexico/General
NZ
NZ-CHAT
Navajo
PRC
PST8PDT
Pacific/Apia
Pacific/Auckland
Pacific/Bougainville
Pacific/Chatham
Pacific/Chuuk
Pacific/Easter
Pacific/Efate
Pacific/Enderbury
Pacific/Fakaofo
Pacific/Fiji
Pacific/Funafuti
Pacific/Galapagos
Pacific/Gambier
Pacific/Guadalcanal
Pacific/Guam
Pacific/Honolulu
Pacific/Johnston
Pacific/Kiritimati
Pacific/Kosrae
Pacific/Kwajalein
Pacific/Majuro
Pacific/Marquesas
Pacific/Midway
Pacific/Nauru
Pacific/Niue
Pacific/Norfolk
Pacific/Noumea
Pacific/Pago_Pago
```

```
Pacific/Palau
Pacific/Pitcairn
Pacific/Pohnpei
Pacific/Ponape
Pacific/Port_Moresby
Pacific/Rarotonga
Pacific/Saipan
Pacific/Samoa
Pacific/Tahiti
Pacific/Tarawa
Pacific/Tongatapu
Pacific/Truk
Pacific/Wake
Pacific/Wallis
Pacific/Yap
Poland
Portugal
ROC
ROK
Singapore
Turkey
UCT
US/Alaska
US/Aleutian
US/Arizona
US/Central
US/East-Indiana
US/Eastern
US/Hawaii
US/Indiana-Starke
US/Michigan
US/Mountain
US/Pacific
US/Samoa
UTC
Universal
W-SU
WET
Zulu
```

[57]:
```python
IST_TZ = pytz.timezone('Asia/Kolkata')
ist_now = IST_TZ.normalize(utc_now)

print(f'India now            : {ist_now}')
print(f'tzinfo               : {ist_now.tzinfo}')
```

```
India now            : 2019-11-14 10:06:36.895482+05:30
tzinfo               : Asia/Kolkata
```

```
[58]: len(pytz.all_timezones)
```

[58]: 592

```
[59]: len(pytz.common_timezones)
```

[59]: 440

```
[64]: utc_now_1 = datetime.datetime.now(tz=pytz.UTC)
      # utc_now_2 = datetime.datetime.utcnow(tz=pytz.UTC)   # utcnow doesnt accept any␣
       ↪keywords

      UTC_TZ = pytz.timezone('UTC')
      naive_utc_now = datetime.datetime.utcnow()
      utc_now_2 = UTC_TZ.localize(naive_utc_now)
```

```
[65]: assert utc_now_1 == utc_now_2
```

```
[66]: utc_now_1
```

[66]: datetime.datetime(2019, 11, 14, 5, 7, 5, 356739, tzinfo=<UTC>)

Problem: For a given date, tell the week of the day

```
[71]: local_now = datetime.datetime.now()
      # weekday returns 0 (monday) through 6 (sunday)
      wd = datetime.date.weekday(today)
      # Days start at 0 for monday
      days = ('monday', 'tuesday', 'wednesday', 'thursday', 'friday', 'saturday',␣
       ↪'sunday')
      print(f'Today is day number {wd}' )
      print(f'which is a {days[wd]}')
```

```
Today is day number 3
which is a thursday
```

```
[72]: def get_week_of_day():
          import calendar

          given_input = '08 23 2019'  # "MM DD YYYY" # input()
          month, day, year = map(int, given_input.split(' '))

          weeks = ('MONDAY', 'TUESDAY', 'WEDNESDAY', 'THURSDAY',
                   'FRIDAY', 'SATURDAY', 'SUNDAY')
          print(calendar.weekday(year, month, day))

          print(weeks[calendar.weekday(year, month, day)])

      get_week_of_day()
```

```
4
FRIDAY
```