
La librería datetime

La librería datetime provee distintas clases que definen los tipos fechas (date), fecha y hora (datetime), hora (time) y delta de tiempos (timedelta) entre otros. A continuación presentamos una breve referencia de cada uno.

Diferencia de tiempo (timedelta)

Un objeto *timedelta* representa una duración, la diferencia entre dos fechas u horas.

Para crear un objeto de tipo *timedelta* se le puede indicar, opcionalmente, la cantidad de días, la cantidad de segundos, la cantidad de microsegundos, la cantidad de milisegundos, la cantidad de minutos, la cantidad de horas, la cantidad de semanas. Todos estos argumentos tienen como valor por defecto el cero. Y pueden ser positivos o negativos. Solo los días, los segundos y los microsegundos son guardados internamente, el resto de los valores se calculan.

A continuación se listan los atributos del objeto *timedelta*:

Atributos de clase		
Atributo	Descripción	Valor
min	El objeto timedelta más negativo	timedelta(-999999999)
max	El objeto timedelta más positivo	timedelta(days=999999999, hours=23, minutes=59, seconds=59, microseconds=999999)
resolution	La diferencia más pequeña entre dos timedeltas distintos	timedelta(microseconds=1)
Atributos de instancia		
Atributo	Descripción	Valor
days	Días	Entre -999999999 y 999999999 inclusive
seconds	Segundos	Entre 0 y 86399 inclusive
microseconds	Microsegundos	Entre 0 y 999999 inclusive

Adicionalmente, soporta las siguientes operaciones: suma (+), resta (-), multiplicación por un número decimal, división con otro timedelta y con un número decimal, división entera por un número decimal o por un timedelta, el módulo (resto de la división) entre timedeltas.

Fechas (date)

Un objeto *date* representa una fecha (año, mes y día).

Para crear un objeto de tipo fecha hay que indicarle el año, el mes y el día. El año debe estar entre 1 y 9999. El mes debe estar entre 1 y 12. Y el día debe estar entre 1 y el máximo número de días para ese mes y año. Si alguno de estos argumentos están fuera de rango se lanza una excepción del tipo *ValueError*.

A continuación se listan los atributos y métodos más relevantes del objeto *date*.

Atributos de clase		
Atributo	Descripción	Valor
min	La fecha más chica representable	date(1, 1, 1)
max	La fecha más grande representable	date(9999, 12, 31)
resolution	La diferencia más pequeña entre dos fechas distintas	timedelta(days=1)
Atributos de instancia		
Atributo	Descripción	Valor
year	Año	Entre 1 y 9999 inclusive
month	Mes	Entre 1 y 12 inclusive
day	Día	Entre 1 y el máximo número de días para ese mes y año
Métodos de instancia		
Método	Descripción	
replace	Reemplaza los atributos de instancia indicados.	
weekday	Devuelve el día de la semana que corresponde. Lunes es 0 y Domingo es 6.	

isoweekday	Devuelve el día de la semana que corresponde. Lunes es 1 y Domingo es 7.
isocalendar	Devuelve una tupla con (año ISO, número de semana ISO, día de la semana ISO).
isoformat	Devuelve un string con el formato 'YYYY-MM-DD'
strftime	Convierte la fecha a un string con el formato que se especifica.
Métodos de clase	
Método	Descripción
date	Constructor de fecha
today	día actual local

Fechas y horas (datetime)

Un objeto *datetime* es un único objeto que contiene toda la información de un objeto *date* y un objeto *time*.

Para crear un objeto de tipo fecha y hora hay que indicarle el año, el mes y el día. Opcionalmente se le pueden indicar la hora, los minutos, los segundos y los microsegundos. Las horas deben estar entre 0 y 23. Los minutos entre 0 y 59. Los segundos entre 0 y 59. Y los microsegundos entre 0 y 999999. Si alguno de estos argumentos están fuera de rango se lanza una excepción del tipo *ValueError*.

A continuación se listan los atributos y métodos más relevantes del objeto *datetime*.

Atributos de clase		
Atributo	Descripción	Valor
min	La fecha y hora más chica representable	date(1, 1, 1)
max	La fecha y hora más grande representable	date(9999, 12, 31, 23, 59, 59, 999999)
resolution	La diferencia más pequeña entre dos fechas y horas distintas	timedelta(microseconds=1)
Atributos de instancia		
Atributo	Descripción	Valor
year	Año	Entre 1 y 9999 inclusive
month	Mes	Entre 1 y 12 inclusive
day	Día	Entre 1 y el máximo número de días para ese mes y año
hour	Hora	Entre 0 y 23 inclusive

minute	Minuto	Entre 0 y 59 inclusive
second	Segundo	Entre 0 y 59 inclusive
microsecond	Microsegundo	Entre 0 y 999999 inclusive
Métodos de instancia		
Método	Descripción	
date	Devuelve un objeto fecha correspondiente	
time	Devuelve un objeto hora correspondiente	
replace	Reemplaza los atributos de instancia indicados.	
weekday	Devuelve el día de la semana que corresponde. Lunes es 0 y Domingo es 6.	
isoweekday	Devuelve el día de la semana que corresponde. Lunes es 1 y Domingo es 7.	
isocalendar	Devuelve una tupla con (año ISO, número de semana ISO, día de la semana ISO).	
isoformat	Devuelve un string con el formato 'YYYY-MM-DDTHH:MM:SS.ffffff'	
strftime	Convierte la fecha y hora a un string con el formato que se especifica.	
Métodos de clase		
Método	Descripción	
datetime	Constructor de fecha y hora	
today	Día y hora actual local	
now	Día y hora actual local	
utcnow	Día y hora actual UTC	
strptime	Devuelve un datetime que corresponde a una fecha en string parseada de acuerdo con un formato	

Horas (time)

Un objeto time representa una del día (local), independiente de cualquier día.

Para crear un objeto de tipo hora se le puede indicar, opcionalmente, la hora, los minutos, los segundo y los microsegundo. Las horas deben estar entre 0 y 23. Los minutos entre 0 y 59. Los segundo entre 0 y 59. Y los microsegundos entre 0 y 999999. Si alguno de estos argumentos están fuera de rango se lanza una excepción del tipo ValueError.

Do Not Copy or Post

A continuación se listan los atributos y métodos más relevantes del objeto *time*.

Atributos de clase		
Atributo	Descripción	Valor
min	La hora más chica representable	time(0, 0, 0, 0)
max	La hora más grande representable	time(23, 59, 59, 999999)
resolution	La diferencia más pequeña entre dos horas distintas	timedelta(microseconds=1)
Atributos de instancia		
Atributo	Descripción	Valor
hour	Hora	Entre 0 y 23 inclusive
minute	Minuto	Entre 0 y 59 inclusive
second	Segundo	Entre 0 y 59 inclusive
microsecond	Microsegundo	Entre 0 y 999999 inclusive
Métodos de instancia		
Método	Descripción	
replace	Reemplaza los atributos de instancia indicados.	
isoformat	Devuelve un string con el formato 'HH:MM:SS.ffffff'	
strftime	Convierte la hora a un string con el formato que se especifica.	
Métodos de clase		
Método	Descripción	
time	Constructor de hora	

Conversión de fechas

La librería `datetime` nos provee algunas funciones para convertir fechas y horas en strings y viceversa. Esta funcionalidad es muy importante porque al transmitir datos a otros programas ya sea exportando datos a un archivo o enviando datos a través de un webservice, etc. es necesario poder serializar y deserializar las fechas.

La función `strftime` convierte un objeto de tipo fecha (`date`), fecha y hora (`datetime`) u hora (`time`) en un string y la función `strptime` convierte un string en un objeto de tipo fecha y hora (`datetime`).

A continuación veremos los ejemplos más relevantes de los usos de las funciones `strftime` y `strptime`.

```
>>> import datetime

>>> date_and_time = datetime.datetime(2019, 4, 30, 11, 25, 30)

# La función strftime crea un string a partir de un objeto datetime

>>> date_and_time.strftime('%Y-%m-%d')
'2019-04-30'

>>> date_and_time.strftime('%Y-%m-%d %H:%M:%S')
'2019-04-30 11:25:30'

>>> date_and_time.strftime('%Y/%m/%d')
'2019/04/30'
```

```
>>> date_and_time.strftime('%Y-%m-%d T%H:%M:%S')
'2019-04-30 T11:25:30'

>>> date_and_time.strftime('%d/%m/%y %H:%M:%S')
'30/04/19 11:25:30'

# La función strftime crea un objeto datetime a partir de un string

>>> datetime.datetime.strptime('2019-01-10', '%Y-%m-%d')
datetime.datetime(2019, 1, 10, 0, 0)

>>> datetime.datetime.strptime('2019-01-10 11:30:25', '%Y-%m-%d %H:%M:%S')
datetime.datetime(2019, 1, 10, 11, 30, 25)

>>> datetime.datetime.strptime('2019-01-10 T11:30:25', '%Y-%m-%d T%H:%M:%S')
datetime.datetime(2019, 1, 10, 11, 30, 25)

>>> datetime.datetime.strptime('2019-01-10 T11:30:25.95', '%Y-%m-%d T%H:%M:%S.%f')
datetime.datetime(2019, 1, 10, 11, 30, 25, 950000)

>>> datetime.datetime.strptime('10-01-19 11:30:25', '%d-%m-%y %H:%M:%S')
datetime.datetime(2019, 1, 10, 11, 30, 25)
```

Para ver todos los formatos que aceptan las funciones *strftime* y *strptime* ver la documentación de Python¹.

¹ <https://docs.python.org/3/library/datetime.html#strftime-and-strptime-behavior>