**Convertir un objeto date en string en Python**

## Cómo convertir un objeto datetime, date o time a string

Para convertir una fecha a string tienes que hacer uso del método strftime() de las clases datetime, date o time. Este método devuelve la representación de la fecha/hora como una cadena con un formato específico. El formato de la cadena de salida se pasa como argumento al método y se define a partir de unos [códigos de formato](https://j2logo.com/convertir-objeto-date-en-string-python/#format-codes) que te enseñaré más adelante. Antes, veamos unos ejemplos.

Ejemplo 1. Transformar un objeto datetime en string

En el ejemplo siguiente se crea un objeto de tipo datetime y a partir de él se obtienen distintas representaciones de tipo string con diferentes formatos. Recuerda importar el módulo datetime. La fecha de ejemplo es el 3 de enero de 2020 a las 8 horas.

import datetime

# Creamos un objeto datetime

una\_fecha = datetime.datetime(2020, 1, 3, 8, 0)

# Convertimos el objeto datetime a string con formato <día>/<mes>/<año>

f1\_str = una\_fecha.strftime('%d/%m/%Y')

print(f1\_str)

> 03/01/2020

# Convertimos el objeto datetime a string con formato <día>-<mes>-<año>

f2\_str = una\_fecha.strftime('%d-%m-%Y')

print(f2\_str)

> 03-01-2020

# Convertimos el objeto datetime a string

# con formato <hora>:<minutos>:<segundos>

hora\_str = una\_fecha.strftime('%H:%M:%S')

print(hora\_str)

> 08:00:00

# Convertimos el objeto datetime a string uniendo todo lo anterior

fecha\_hora\_str = una\_fecha.strftime('%d/%m/%Y %H:%M')

print(fecha\_hora\_str)

> 03/01/2020 08:00

## Ejemplo 2. Convertir un objeto date en string

Tomando como base el objeto una\_fecha del ejemplo anterior, vamos a pasar su objeto date a string.

# Convertimos el objeto date a string con formato <día> de <mes> de <año>

solo\_fecha = una\_fecha.date()

solo\_fecha\_str = solo\_fecha.strftime('%d de %m de %Y')

print(solo\_fecha\_str)

> 03 de 01 de 2020

Sé lo que estás pensando… ¿Qué ocurre si tratas de obtener la hora o minutos de un objeto date? Pues Python les asigna por defecto el valor 0.

solo\_hora\_str = solo\_fecha.strftime('%H:%M')

print(solo\_hora\_str)

> 00:00

## Ejemplo 3. Convertir un objeto time en string

De nuevo tomaremos como base el objeto una\_fecha para convertir su objeto time a string.

# Convertimos el objeto time a string con formato <hora>:<minutos>:<segundos>

solo\_hora = una\_fecha.time()

solo\_hora\_str = solo\_hora.strftime('%H:%M:%S')

print(solo\_hora\_str)

> 08:00:00

Si tratamos de obtener la fecha de un objeto time, veremos que Python le asigna por defecto el día 1 de enero de 1900.

solo\_fecha\_str = solo\_hora.strftime('%d/%m/%Y')

print(solo\_fecha\_str)

> 01/01/1900

## Códigos de formato

A continuación se muestran los códigos de formato válidos para aquellas plataformas con una implementación estándar de C.

| **Directiva** | **Significado** | **Ejemplo** |
| --- | --- | --- |
| %a | Nombre abreviado del día de la semana | Sun, Mon, … |
| %A | Nombre completo del día de la semana | Sunday, Monday, … |
| %w | Día de la semana como número decimal (domingo es 0) | 0, 1, …, 6 |
| %d | Día del mes como número decimal (añade cero delante) | 01, 02, …, 31 |
| %-d | Día del mes como número decimal | 1, 2, …, 30 |
| %b | Nombre del mes abreviado | Jan, Feb, …, Dec |
| %B | Nombre del mes completo. | January, February, … |
| %m | Mes como número decimal (añade cero delante) | 01, 02, …, 12 |
| %-m | Mes como número decimal | 1, 2, …, 12 |
| %y | Año sin siglo como número decimal (añade cero delante) | 00, 01, …, 99 |
| %-y | Año sin siglo como número decimal | 0, 1, …, 99 |
| %Y | Año con siglo como número decimal | 2013, 2019 etc. |
| %H | Hora (24 horas) como número decimal (añade cero delante) | 00, 01, …, 23 |
| %-H | Hora (24 horas) como número decimal | 0, 1, …, 23 |
| %I | Hora (12 horas) como número decimal (añade cero delante) | 01, 02, …, 12 |
| %-I | Hora (12 horas) como número decimal | 1, 2, … 12 |
| %p | AM o PM locales | AM, PM |
| %M | Minuto como número decimal (añade cero delante) | 00, 01, …, 59 |
| %-M | Minuto como número decimal | 0, 1, …, 59 |
| %S | Segundo como número decimal (añade cero delante) | 00, 01, …, 59 |
| %-S | Segundo como número decimal | 0, 1, …, 59 |
| %f | Microsegundos como número decimal (añade ceros delante) | 000000 – 999999 |
| %z | Compensación UTC con formato +HHMM o -HHMM |  |
| %Z | Nombre de la zona horaria |  |
| %j | Día del año como número decimal (añade cero delante) | 001, 002, …, 366 |
| %-j | Día del año como número decimal | 1, 2, …, 366 |
| %U | Número de semana del año (domingo como primer día de la semana). Los días de un nuevo año anteriores al primer domingo se consideran en la semana 0. | 00, 01, …, 53 |
| %W | Número de semana del año (lunes como primer día de la semana). Los días de un nuevo año anteriores al primer lunes se consideran en la semana 0. | 00, 01, …, 53 |
| %c | Representación apropiada de fecha y hora local | Mon Sep 30 07:06:05 2013 |
| %x | Representación apropiada de fecha local | 09/30/13 |
| %X | Representación apropiada de hora local | 07:06:05 |
| %% | El carácter ‘%’ literal | % |