

La Librería `Datetime`

Para manejar fechas en Python se suele utilizar la librería `datetime` que incorpora los tipos de datos `date`, `time` y `datetime` para representar fechas y funciones para manejarlas. Algunas de las operaciones más habituales que permite son:

- Acceder a los distintos componentes de una fecha (año, mes, día, hora, minutos, segundos y microsegundos).
- Convertir cadenas con formato de fecha en los tipos `date`, `time` o `datetime`.
- Convertir fechas de los tipos `date`, `time` o `datetime` en cadenas formateadas de acuerdo a diferentes formatos de fechas.
- Hacer aritmética de fechas (sumar o restar fechas).
- Comparar fechas.

Los tipos de datos `date`, `time` y `datetime`

- `date(año, mes, día)`: Devuelve un objeto de tipo `date` que representa la fecha con el `año`, `mes` y `día` indicados.
- `time(hora, minutos, segundos, microsegundos)`: Devuelve un objeto de tipo `time` que representa un tiempo la `hora`, `minutos`, `segundos` y `microsegundos` indicados.
- `datetime(año, mes, día, hora, minutos, segundos, microsegundos)`: Devuelve un objeto de tipo `datetime` que representa una `fecha` y `hora` con el `año`, `mes`, `día`, `hora`, `minutos`, `segundos` y `microsegundos` indicados.

Ejercicio de Inducción: Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
from datetime import date, time, datetime
>>> date(2020, 12, 25)
datetime.date(2020, 12, 25)
>>> time(13,30,5)
datetime.time(13, 30, 5)
>>> datetime(2020, 12, 25, 13, 30, 5)
datetime.datetime(2020, 12, 25, 13, 30, 5)
>>> print(datetime(2020, 12, 25, 13, 30, 5))
2020-12-25 13:30:05
```

—

1. Acceso a los componentes de una fecha

- `date.today()`: Devuelve un objeto del tipo `date` la fecha del sistema en el momento en el que se ejecuta.
- `datetime.now()`: Devuelve un objeto del tipo `datetime` con la fecha y la hora del sistema en el momento exacto en el que se ejecuta.
- `d.year`: Devuelve el año de la fecha `d`, puede ser del tipo `date` o `datetime`.
- `d.month`: Devuelve el mes de la fecha `d`, que puede ser del tipo `date` o `datetime`.
- `d.day`: Devuelve el día de la fecha `d`, que puede ser del tipo `date` o `datetime`.

- **d.weekday()**: Devuelve el día de la semana de la fecha **d**, que puede ser del tipo **date** o **datetime**.
- **t.hour**: Devuelve las horas del tiempo **t**, que puede ser del tipo **time** o **datetime**.
- **t.minute**: Devuelve los minutos del tiempo **t**, que puede ser del tipo **time** o **datetime**.
- **t.second**: Devuelve los segundos del tiempo **t**, que puede ser del tipo **time** o **datetime**.
- **t.microsecond**: Devuelve los microsegundos del tiempo **t**, que puede ser del tipo **time** o **datetime**.

Ejercicio de Inducción: Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
>>> from datetime import date, time, datetime
>>> print(date.today())
2020-04-11
>>> dt = datetime.now()
>>> dt.year
2020
>>> dt.month
4
>>> dt.day
11
>>> dt.hour
22
>>> dt.minute
5
>>> dt.second
45
>>> dt.microsecond
1338
```

2. Conversión de fechas en cadenas con diferentes formatos

- **d.strftime(formato)**: Devuelve la cadena que resulta de transformar la fecha **d** con el formato indicado en la cadena **formato**. La cadena **formato** puede contener los siguientes marcadores de posición: **%Y** (año completo), **%y** (últimos dos dígitos del año), **%m** (mes en número), **%B** (mes en palabra), **%d** (día), **%A** (día de la semana), **%a** (día de la semana abreviado), **%H** (hora en formato 24 horas), **%I** (hora en formato 12 horas), **%M** (minutos), **%S** (segundos), **%p** (AM o PM), **%C** (fecha y hora completas), **%x** (fecha completa), **%X** (hora completa).

Ejercicio de Inducción: Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
>>> from datetime import date, time, datetime
>>> d = datetime.now()
>>> print(d.strftime('%d- %m - %Y'))
13-04-2020
>>> print(d.strftime('%A, %d %B, %y'))
Monday, 13 April, 20
>>> print(d.strftime('%H : %M : %S'))
20:55:53
>>> print(d.strftime('%H horas, %M minutos y %S segundos'))
20 horas, 55 minutos y 53 segundos
```

3. Conversión de cadenas en fechas

- **strptime(s, formato)**: Devuelve el objeto de tipo **date**, **time** o **datetime** que resulta de convertir la cadena **s** de acuerdo al formato indicado en la cadena **formato**. La cadena formato puede contener los siguientes marcadores de posición: **%Y** (año completo), **%y** (últimos dos dígitos del año), **%m** (mes en número), **%B** (mes en palabra), **%d** (día), **%A** (día de la semana), **%a** (día de la semana abreviado), **%H** (hora en formato 24 horas), **%I** (hora en formato 12 horas), **%M** (minutos), **%S** (segundos), **%p** (AM o PM), **%C** (fecha y hora completas), **%x** (fecha completa), **%X** (hora completa).

Ejercicio de Inducción: Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
>>> from datetime import date, time, datetime
>>> datetime.strptime('15/4/2020', '%d/%m/%Y')
datetime.datetime(2020, 4, 15, 0, 0)
>>> datetime.strptime('2020-4-15 20:50:30', '%Y-%m-%d %H:%M:%S')
datetime.datetime(2020, 4, 15, 20, 50, 30)
```

4. Aritmética de fechas

Para representar el tiempo transcurrido entre dos fechas se utiliza el tipo **timedelta**.

- **timedelta(dias, segundos, microsegundos)**: Devuelve un objeto del tipo **timedelta** que representa un intervalo de tiempo con los **dias**, **segundos** y **microsegundos** indicados.
- **d1 - d2**: Devuelve un objeto del tipo **timedelta** que representa el tiempo transcurrido entre las fechas **d1** y **d2** del tipo **datetime**.
- **d + delta**: Devuelve la fecha del tipo **datetime** que resulta de sumar a la fecha **d** el intervalo de tiempo **delta**, donde **delta** es del tipo **timedelta**.

Ejercicio de Inducción: Pruebe las siguientes líneas de código y verifique los resultados presentados:

```
>>> from datetime import date, time, datetime, timedelta
>>> d1 = datetime(2020, 1, 1)
>>> d1 + timedelta(31, 3600)
datetime.datetime(2020, 2, 1, 1, 0)
>>> datetime.now() - d1
datetime.timedelta(days=132, seconds=1826, microseconds=895590)
```
