

Programación orientada a objetos con Python

Python permite varios paradigmas de programación, incluyendo la programación orientada a objetos (POO). La POO es una manera de estructurar el código que le hace especialmente efectivo organizando y reutilizando código, aunque su naturaleza abstracta hace que no sea muy intuitivo cuando se empieza.

La programación orientada a objetos en Python es opcional y de hecho hasta ahora no la hemos usado directamente, aunque indirectamente lo hemos hecho desde el principio. Aunque su mayor ventaja aparece con los programas largos y más complejos, es muy útil entender cómo funciona la POO, ya que es así como Python funciona internamente.

La idea básica es sencilla. Si tenemos un tipo de dato más complejo de los que hemos visto hasta ahora como listas o diccionarios y queremos crear un nuevo tipo de dato con propiedades particulares, podemos definirlo con una clase, algo parecido a una función def.

La clase tiene una función principal especial `__init__()` que construye el elemento de la clase (llamado objeto) y que se ejecuta cuando crea un nuevo objeto o instancia de esa clase; hemos puesto `name` como parámetro único obligatorio, pero no tiene por qué tener ninguno.

La misteriosa variable `self` con la que empieza cada función (llamadas métodos en los objetos), se refiere al objeto en concreto que estamos creando, esto se verá más claro con un ejemplo.