

¿Qué es recursión?

La recursión en Python (y en la programación en general) es un concepto en el que una función se llama a sí misma directa o indirectamente para resolver un problema. Es útil cuando un problema puede ser descompuesto en casos más pequeños y similares al problema original. La recursión tiene dos partes principales: el caso base y el caso recursivo.

Caso base

El caso base es la condición en la que la función recursiva deja de llamarse a sí misma y simplemente devuelve un valor sin hacer más llamadas recursivas. Esto es fundamental para evitar que la función se llame infinitamente.

Caso recursivo

El caso recursivo es la parte donde la función se llama a sí misma con un conjunto de parámetros ligeramente modificados para resolver una versión más pequeña del problema original.

Ejemplo

Para entender la recursión podemos pensar en las muñecas rusas, también conocidas como matrioskas.

Imagina una muñeca grande que contiene muchas muñecas más pequeñas dentro de ella, cada una similar a la siguiente pero de menor tamaño.

Para llegar a la muñeca más pequeña (el caso base), necesitas abrir cada una de las muñecas más grandes (caso recursivo) hasta que llegues a la más pequeña.

Cada muñeca más grande contiene otra muñeca similar dentro, así como cada llamada recursiva resuelve un subproblema más pequeño hasta alcanzar el caso base.