print(\*nums), is called **unpacking** in Python.

**1. The list:**

nums = [2, 4, 6, 7, 8]

This creates a list of integers.

**2. Using print(nums)**

This prints the list **as is**, meaning the entire list object:

[2, 4, 6, 7, 8]

**3. Using print(\*nums)**

Here’s where the magic happens with the \*.

The \* operator **unpacks** the list — it takes each individual element of the list and passes them as **separate arguments** to the print() function.

So internally, this:

print(\*nums)

Is equivalent to:

print(2, 4, 6, 7, 8)

Which results in:

2 4 6 7 8

Each element becomes its **own argument**, and print() separates arguments by default with a space.

**This works with any iterable**

You can use \* to unpack:

* lists
* tuples
* sets (unordered though)
* even strings (they unpack into characters)

Example:

chars = ['H', 'e', 'y']

print(\*chars) # Output: H e y