

In Python, specifically within an interactive interpreter or a Jupyter notebook, there is a distinct difference between simply typing a variable name and using the `print()` function to display its value.

### 1. Typing the Variable Name (Implicit Display):

When you type a variable name and press Enter in an interactive Python environment (like the standard Python shell or a Jupyter cell), the interpreter automatically evaluates the expression (which is just the variable in this case) and displays its representation. This representation is typically generated by the object's `__repr__` method. The `__repr__` method aims to produce a string that, if passed to `eval()`, would ideally recreate the object. This often includes details like the object's type and structure.

Python

```
>>> my_list = [1, 2, 3]
>>> my_list
[1, 2, 3]
>>> my_string = "hello"
>>> my_string
'hello'
```

### 2. Using the `print()` Function (Explicit Output):

The `print()` function is explicitly designed to produce human-readable output to the standard output stream (usually the console). It calls the object's `__str__` method to get a string representation, which is intended to be more user-friendly and less formal than `__repr__`. `print()` also adds a newline character by default after printing its arguments, and can handle multiple arguments, separating them with spaces by default.

Python

```
>>> my_list = [1, 2, 3]
>>> print(my_list)
[1, 2, 3]
>>> my_string = "hello"
>>> print(my_string)
hello
```

### Key Differences Summarized:

- Purpose: Typing the variable name is for interactive inspection/debugging, while `print()` is for explicit output in scripts and programs.
- Method Called: Variable name display typically uses `__repr__`, while `print()` uses `__str__`.
- Format: `__repr__` often provides a more technical, unambiguous representation, while `__str__` aims for a more readable, user-friendly string.
- Newline: `print()` adds a newline by default; typing a variable name does not inherently add one (the interpreter's output mechanism handles line breaks).
- Scope: Typing a variable name only works in interactive environments; `print()` works universally in scripts and interactive mode.