



Print Function

by [shashank21j](#)

Tutorial

Problem

Submissions

Leaderboard

Discussions

Editorial

In Python 2, the default print is a simple IO method that doesn't give many options to play around with.

The following two examples will summarize it.

Example 1:

```
var, var1, var2 = 1,2,3
print var
print var1, var2
```

Prints two lines and, in the second line, **var1** and **var2** are separated by a single space.

Example 2:

```
for i in xrange(10):
    print i,
```

Prints each element separated by space on a single line. Removing the comma at the end will print each element on a new line.

Let's import the advanced `print_function` from the `__future__` module.

Its method signature is below:

```
print(*values, sep=' ', end='\n', file=sys.stdout)
print(value1, value2, value3, sep=' ', end='\n', file=sys.stdout)
```

Here, `values` is an array and `*values` means array is unpacked, you can add values separated by a comma too. The arguments `sep`, `end`, and `file` are optional, but they can prove helpful in formatting output without taking help from a string module.

The argument definitions are below:

`sep` defines the delimiter between the values.

`end` defines what to print after the values.

`file` defines the output stream.

in Python 2 `print_function` is much faster than the default `print`

Join us on IRC at [#hackerrank](#) on freenode for hugs or bugs.

[Contest Calendar](#) | [Interview Prep](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)