1. [Use the String Formatting With the Modulo % Sign in Python](https://www.delftstack.com/howto/python/python-print-without-space/" \l "use-the-string-formatting-with-the-modulo-sign-in-python)
2. [Use the String Formatting With the str.format() Function in Python](https://www.delftstack.com/howto/python/python-print-without-space/#use-the-string-formatting-with-the-str.format-function-in-python)
3. [Use String Concatenation in Python](https://www.delftstack.com/howto/python/python-print-without-space/#use-string-concatenation-in-python)
4. [Use the f-string for String Formatting in Python](https://www.delftstack.com/howto/python/python-print-without-space/#use-the-f-string-for-string-formatting-in-python)
5. [Use the sep Parameter of the print Statement in Python](https://www.delftstack.com/howto/python/python-print-without-space/#use-the-sep-parameter-of-the-print-statement-in-python)

When we generally use the print statement, we sometimes use a comma (,) as a separator, which sometimes leads to unnecessary spaces between values. Fortunately, you can utilize some alternatives in Python that help you handle these spacing problems.

In this guide, we’ll teach you how to use the different methods to print without spaces between values in Python.

We will take a simple code for all the methods, which takes up a print statement and contains several arguments separated by commas. For example, the following program below uses the comma operator to print the values.

x = 10

print ('The number of mangoes I have are "', x, '"')

Output:

The number of mangoes I have are " 10 "

We should note that there are unnecessary spaces between the number 10 and the double quotes surrounding it. The aim is to prevent or remove this excessive or unnecessary spacing.

[Use the String Formatting With the Modulo % Sign in Python](https://www.delftstack.com/howto/python/python-print-without-space/#use-the-string-formatting-with-the-modulo-sign-in-python)

String formatting gives more customization options to the users to go with the classic print statement. The % sign is also known as an interpolation or a string formatting operator.

String formatting can be implemented in two ways, and using the % sign is one of those options.

The % sign, followed by a letter that represents the conversion type, works as a placeholder for the variable. The code below uses the % sign to print without spaces between values in Python.

x = 10

print ('The number of mangoes I have are "%d"' %x )

Output:

The number of mangoes I have are "10"

[Use the String Formatting With the str.format() Function in Python](https://www.delftstack.com/howto/python/python-print-without-space/#use-the-string-formatting-with-the-str-format-function-in-python)

When using the string formatting, braces {} are used to mark the spot in the statement where the variable would be substituted.

The str.format() has been introduced in Python 3 and is available to use in the latest versions of Python. This function is utilized for the efficient handling of complex string formatting.

The following code uses the str.format() function to print without spaces between values in Python.

x = 10

print ('The number of mangoes I have are "{}"'.format(x) )

Output:

The number of mangoes I have are "10"

It is recommended to use the format() function instead of the old % operator in the newer versions of Python.

[Use String Concatenation in Python](https://www.delftstack.com/howto/python/python-print-without-space/#use-string-concatenation-in-python)

The + operator, also known as the string concatenation operator, can be used in this case to prevent unnecessary spacing between the values. It’s a direct alternative to comma separation and can be used along with the print statement.

Here is an example code that shows the use of string concatenation in the print statement.

x = 10

print ('The number of mangoes I have are "' + str(x) + '"')

Output:

The number of mangoes I have are "10"

[Use the f-string for String Formatting in Python](https://www.delftstack.com/howto/python/python-print-without-space/#use-the-f-string-for-string-formatting-in-python)

Python 3.6 introduced the f-string, which is another method of achieving string formatting; however, it has the edge over the two other processes for string formatting mentioned above because it’s comparatively faster than its other two peers.

The following code uses the fstring formatting to print without spaces between values in Python.

x = 10

print (f'The number of mangoes I have are "{x}"')

Output:

The number of mangoes I have are "10"

[Use the sep Parameter of the print Statement in Python](https://www.delftstack.com/howto/python/python-print-without-space/#use-the-sep-parameter-of-the-print-statement-in-python)

You can modify the spacing between the arguments of the print statement by using the sep parameter. The sep parameter can only be found and used in Python 3 and later versions. It can also be utilized for the formatting of the output strings.

The following code uses the sep parameter to print without spaces between values in Python.

x = 10

print ('The number of mangoes I have are "', x, '"', sep='')

Output:

The number of mangoes I have are "10"