# Python

May 22, 2021

**Day 1 -** **{**Printing, Commenting, Debugging, String Manipulation and Variables**}**

**Printing**

**Def:** The print () function prints the specified message to the screen, or other standard output device.

The message can be a string, or any other object will be converted into a string before written to the screen.

**Syntax**

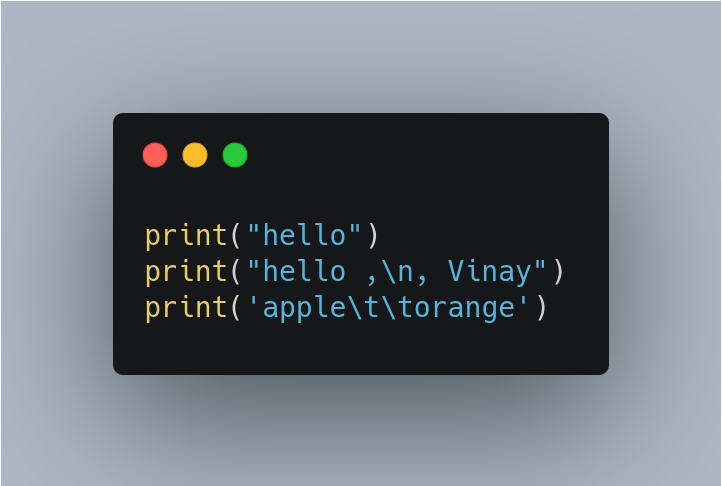
|  |
| --- |
| Print (Object(s), sep=separator, end=end, file=file, flush=flush) |

**Parameter Values**

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| Object(s) | Any object, and as many as you like. Will be converted to string before printed. |
| sep=’separator’ | Optional. Specify how to separate the objects, if there is more than one. Default is’ ’. |
| end=’end’ | Optional. Specify what to print at the end. Default is ‘/n’ (line feed). |
| file | Optional. An object with a write method. Default is sys. stdout. |
| flush | Optional. A Boolean, specifying if the object is flushed (True) or buffered (False). Default is False. |

For example, of the print input command

**Input**



**Output**

****

**Note:** Parentheses, you going to mention after the print **() function.** Inside the parentheses we also mentioned that **quotation mark** or **double quotation print (“”).** Furthermore, inside this one we are mentioning to print something, inside the text are what we calling **String print (“Hello World!”).**

**New Line using in print function:**

In above input images we witness the **\n** means that after this you want to print remaining string.

**ex:** print (“Hello World! \nHello World! \nHello World!”)

**Combine the two Strings:**

More about the string method we do want to add the output with the string in further main projects like below example.

**ex:** print (“Bunch of lists” +” Vinay”);

**Note:** While using the space in the editor we must follow some rules, so in further we mostly using the space a lot. Maybe in the if condition and other methods so that the space is commonly used in the Python.