**Printing Output**

There are various methods to print the output. Some method are available to print output in R program. Also if the program of R is written over the **console** line by line then the output is printed normally, no need to use any function for print that output. To do this just select the output variable and press **run** button.

|  |
| --- |
| # select 'x' and then press 'run' button  x <- "Hello"  x |

[**print()**](https://www.geeksforgeeks.org/print-the-argument-to-the-screen-in-r-programming-print-function/) function to print output is the most common method in R. Implementation of this method is very simple. **Check ?print or help(print)**

***Syntax:****print(“any string”) or, print(variable)*

**Example:**

|  |
| --- |
| # R program to illustrate printing output of an R program  # print string  print("GFG")  # print variable  x <- "AIML"  print(x) |

**paste()** method in [R](https://www.geeksforgeeks.org/introduction-to-r-programming-language/) programming is used to concatenate the two string values by separating with delimiters. To print output with string and variable together. **Check ?paste or help(paste)**

***Syntax:****paste(string1, string2, sep=)*

***Return:****Returns the concatenate string.*

**Example 1:**

|  |
| --- |
| # R program to concatenate two strings  # Given Strings  string1 <**-** "AI"  string2 <**-** "ML"  # Using paste() method  answer <**-** paste(string1, string2, sep**=**"&")  print(answer) |

[**sprintf()**](https://www.geeksforgeeks.org/print-a-formatted-string-in-r-programming-sprintf-function/) is basically a **C library** function. This function is use to print string as **C language**. This is working as a wrapper function to print values and strings together like **C language.** This function returns a character vector containing a formatted combination of string and variable to be printed. **Check ?sprintf or help(sprintf)**

***Syntax:****sprintf(“any string %d”, variable) or, sprintf(“any string %s”, variable) or, sprintf(“any string %f”, variable)) etc.*

**Example:**

|  |
| --- |
| # R program to illustrate printing output of an R program  x = "AIML" # string  x1 = 255            # integer  x2 = 23.14          # float  # string print  sprintf("%s is best", x)  # integer print  sprintf("%d is integer", x1)  # float print  sprintf("%f is float", x2) |

**cat()** function in [R Language](https://www.geeksforgeeks.org/introduction-to-r-programming-language/) is used to print out to the screen or to a file. **Check ?cat or help(cat)**

***Syntax:****cat(…, file = “”, sep = ” “, fill = FALSE, labels = NULL, append = FALSE)****Parameters:******…:****atomic vectors, names, NULL and objects with no output****file:****the file in which printing will be done****sep:****specified separator****fill:****If fill=TRUE, a new line will be printed, otherwise not****labels:****specified labels*

# R program to illustrate printing output of an R program

# print string with variable "\n" for new line

x = "AIML"

cat(x, "is best\n")

# print normal string

cat("This is R language")

# print on a file

cat("This is R language", file=”demo.txt”)

This demo.txt is present in your current working directory which can be checked using getwd().

[**write()**](https://www.geeksforgeeks.org/writing-to-files-in-r-programming/) is used to print or write a file with a value of a variable. This function is used a option called **table** to write a file. **Check ?write or help(write)**

***Syntax:****write.table(variable, file = “file1.txt”) or, write.table(“any string”, file = “file1.txt”)*

**Example:**

|  |
| --- |
| # R program to illustrate printing output of an R program  x = "AIML"  # write variable  write.table(x, file = "my\_data1.txt")  # write normal string  write.table("AIML is best", file = "my\_data2.txt") |