**Using readline() method**

readline() method takes input in string format. If one inputs an integer then it is inputted as a string, lets say, one wants to input 255, then it will input as “255”, like a string. So one needs to convert that inputted value to the format that he needs. In this case, string “255” is converted to integer 255. To convert the inputted value to the desired data type, there are some functions in R, 

* [as.integer(n)](https://www.geeksforgeeks.org/convert-a-character-object-to-integer-in-r-programming-as-integer-function/); —> convert to integer
* [as.numeric(n)](https://www.geeksforgeeks.org/convert-factor-to-numeric-and-numeric-to-factor-in-r-programming/); —> convert to numeric type (float, double etc)
* as.complex(n); —> convert to complex number (i.e 3+2i)
* [as.Date(n)](https://www.geeksforgeeks.org/convert-a-string-into-date-format-in-r-programming-as-date-function/) —> convert to date …, etc

*Syntax:   
var = readline();   
var = as.integer(var);  
Note that one can use “<-“ instead of “=”*

Example: 

|  |
| --- |
| # R program to illustrate taking input from the user  # taking input using readline() this command will prompt you  # to input a desired value  var = readline();  # convert the inputted value to integer  var = as.integer(var);  # print the value  print(var) |

One can also show message in the console window to tell the user, what to input in the program. To do this one must use a argument named prompt inside the readline() function. Actually prompt argument facilitates other functions to constructing of files documenting. But prompt is not mandatory to use all the time.

*Syntax:   
var1 = readline(prompt = “Enter any number : “);   
or,   
var1 = readline(“Enter any number : “);*

Example: 

|  |
| --- |
| # R program to illustrate taking input from the user  # taking input with showing the message  var = readline(prompt = "Enter any number : ");  # convert the inputted value to an integer  var = as.integer(var);  # print the value  print(var) |

Taking multiple inputs in R

Taking multiple inputs in R language is same as taking single input, just need to define multiple readline() for inputs. One can use braces for define multiple readline() inside it.

*Syntax:   
var1 = readline(“Enter 1st number : “);   
var2 = readline(“Enter 2nd number : “);   
var3 = readline(“Enter 3rd number : “);   
var4 = readline(“Enter 4th number : “);  
or,  
{   
var1 = readline(“Enter 1st number : “);   
var2 = readline(“Enter 2nd number : “);   
var3 = readline(“Enter 3rd number : “);   
var4 = readline(“Enter 4th number : “);   
}*

Example: 

|  |
| --- |
| # R program to illustrate taking input from the user taking multiple inputs using braces  {      var1 = readline("Enter 1st number : ");      var2 = readline("Enter 2nd number : ");      var3 = readline("Enter 3rd number : ");      var4 = readline("Enter 4th number : ");  }  # converting each value  var1 = as.integer(var1);  var2 = as.integer(var2);  var3 = as.integer(var3);  var4 = as.integer(var4);  # print the sum of the 4 number  print(var1 + var2 + var3 + var4) |

Taking String and Character input in R

To take string input is the same as an integer. For “String” one doesn’t need to convert the inputted data into a string because R takes input as string always. And for “character”, it needs to be converted to ‘character’. Sometimes it may not cause any error. One can take character input as same as string also, but that inputted data is of type string for the entire program. So the best way to use that inputted data as ‘character’ is to convert the data to a character.

*Syntax:  
string:   
var1 = readline(prompt = “Enter your name : “);  
character:   
var1 = readline(prompt = “Enter any character : “);   
var1 = as.character(var1)*

Example: 

|  |
| --- |
| # R program to illustrate taking input from the user string input  var1 = readline(prompt = "Enter your name : ");  # character input  var2 = readline(prompt = "Enter any character : ");  # convert to character  var2 = as.character(var2)  # printing values  print(var1)  print(var2) |