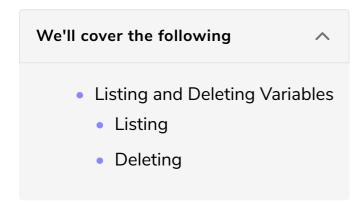
## Basic Methods for Handling Variables

In this lesson, we'll learn how to List and Delete variables.



## Listing and Deleting Variables #

## Listing #

We can check all the variables that have been created in the workspace using the keyword ls().

To import recent packages for execution of codes on our platform, we have declared a variable r at our backend. Under normal circumstances, you will not get this variable when using the ls() command.

```
myRealNumeric <- 10
myDecimalNumeric <- 10.0
myCharacter <- "10"
myBoolean <- TRUE
myInteger <- 0:10
myComplex <- 5i

cat("Variables in the current directory: \n")
ls() # returns all the variables created in the workspace alphabetically cat("\n")</pre>
```

Using Is() for finding all the variables created in the code

We can delete a specific variable from the workspace. The keyword rm() can help us permanently remove one or more objects from the workspace:

```
myRealNumeric <- 10
myDecimalNumeric <- 10.0
myCharacter <- "10"
myBoolean <- TRUE
myInteger <- 0:10
myComplex <- 5i
cat("Variables in the current directory: \n")
ls() # returns all the variables created in the workspace
cat("\n")
cat("Deleting myRealNumeric and myDecimalNumeric \n\n")
rm(myRealNumeric, myDecimalNumeric) # delete the two mentioned variables
cat("Variables in the current directory, now: \n")
ls() # returns all the variables created in the workspace
     # myRealNumeric, myDecimalNumeric are now deleted
cat("\n")
                                                                                  Using rm() to delete a variable.
```

In the next lesson, we have a small exercise for you to test your concepts on variables.