Basic Commands | R programming language

To ensure better understanding and minimize mistakes related to spacing or formatting, the notes will be displayed directly from the interface. This approach allows readers to gain a first-hand experience of how the interface appears, even if they do not have the program installed, especially if the goal is to provide a read-only overview.

General guidelines

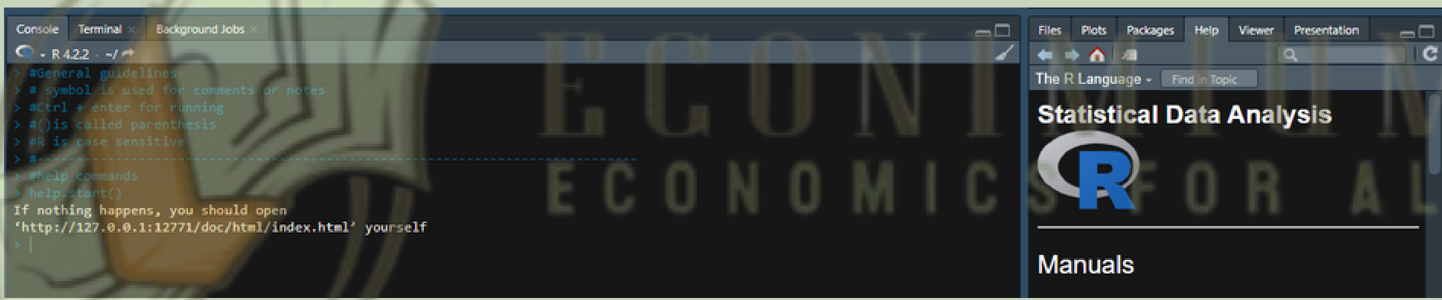
- # symbol is used for comments or notes
- Ctrl + enter for running
- ()is called parenthesis
- R is case sensitive

Help commands

Script Editor

```
help.start()
```

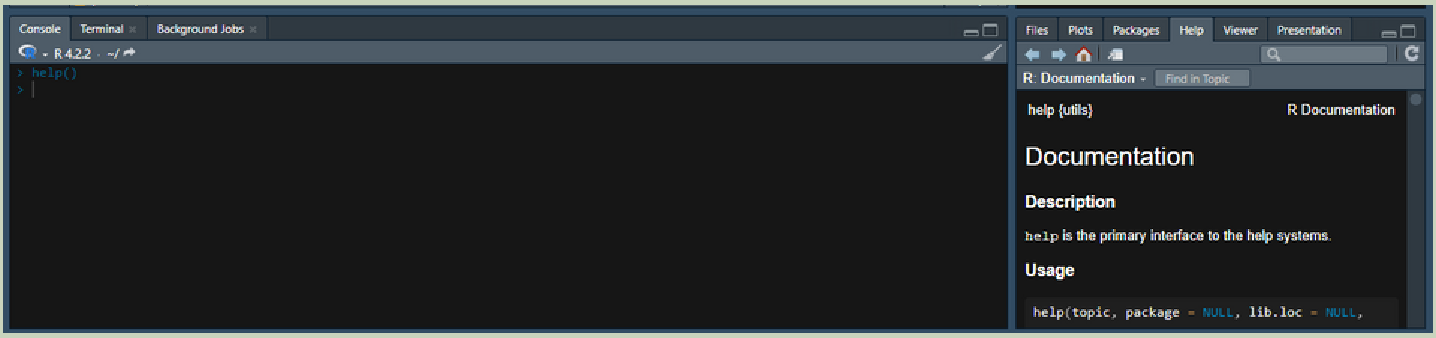
Console



Script Editor

```
help()
```

Console



To check the packages already installed

A package in R is a collection of pre-written functions, data, and documentation that extends R's capabilities for specific tasks.

Script Editor

```
installed.packages()
```

Console

```
> installed.packages()
         Package    LibPath                                        Version  Priority
abind    "abind"    "C:/Users/amalz/AppData/Local/R/win-library/4.2" "1.4-5"  NA
AER      "AER"      "C:/Users/amalz/AppData/Local/R/win-library/4.2" "1.2-14" NA
ape      "ape"      "C:/Users/amalz/AppData/Local/R/win-library/4.2" "5.8"    NA
askpass  "askpass"  "C:/Users/amalz/AppData/Local/R/win-library/4.2" "1.1"    NA
```

Basic Commands | R programming language

To install package *(eg: graDient)*

Script Editor

```
17  install.packages("graDiEnt")
```

Console

```
package 'graDiEnt' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\amalz\AppData\Local\Temp\RtmpsRmOr1\downloaded_packages
> install.packages("ggplot2")
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate ve
rsion of Rtools before proceeding:
```

To (un)load package

Script Editor

```
20  #to load a package
21  library(ggplot2)
22  #to unload a package
23  detach("package:ggplot2")
```

Console

```
> #to load a package
> library(ggplot2)
Warning message:
package 'ggplot2' was built under R version 4.2.3
> #to unload a package
> detach("package:ggplot2")
>
```


To show the list of installed packages available for use

Script Editor

```
25  search()
```

Console

```
> search()
 [1] ".GlobalEnv"        "tools:rstudio"     "package:stats"     "package:graphics"  "package:grDevices"
 [6] "package:utils"     "package:datasets"  "package:methods"   "Autoloads"         "package:base"
>
```

Basic Calculations

- +,-,/ used for basic algebra.
- put _ between_spaces whenever you type two words
- %%used to get the remainder after division
- #all the trig functions are in radians.

Basic Commands | R programming language

Script Editor

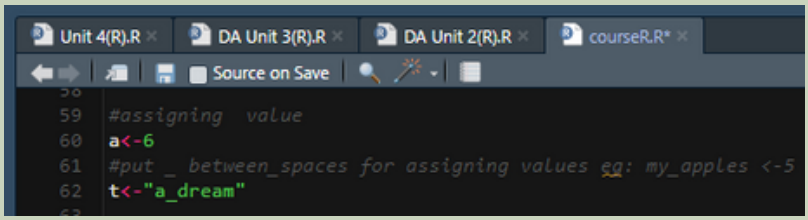
```r
2+5
2^6
abs(-58)
abs(659-985*100+587-98/87%%2)
factorial(5)
24%%5
log(32)
log(32,2)
exp(1)
sin(pi/2)
asin(1)
round(259/6,2)
```

Console

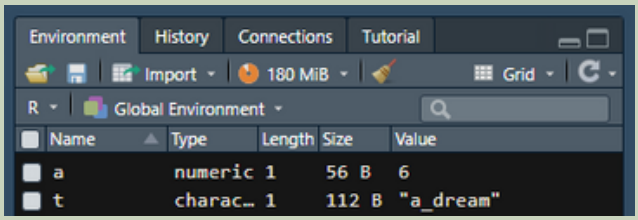
```
> 2+5
[1] 7
> 2^6
[1] 64
> abs(-58)
[1] 58
> abs(659-985*100+587-98/87%%2)
[1] 97352
> factorial(5)
[1] 120
> 24%%5
[1] 4
> log(32)
[1] 3.465736
> log(32,2)
[1] 5
> exp(1)
[1] 2.718282
> sin(pi/2)
[1] 1
> asin(1)
[1] 1.570796
> round(259/6,2)
[1] 43.17
>
```


Assigning value (using <-)

Script Editor



Environment



Console

```
> #assigning  value
> a<-6
> #put _ between_spaces for assigning values eg: my_apples <-5
> t<-"a_dream"
>
```

assigned values gets stored and is displayed in the environment panel

Basic Commands | R programming language

Creating Vectors to assign multiple values (using <- c())

Script Editor

```
70 #-----------------------------------------------------
71 #Vector-one dimensional array
72
73 #c-combine fn
74 x<-c(1,2,3)
75 y<-c("a","b","c")
76 e<- c(FALSE, TRUE, FALSE,FALSE, FALSE, TRUE)
77
```

Environment

Name	Type	Length	Size	Value
a	numeric	1	56 B	6
e	logical	6	80 B	logi [1:6] FALSE…
t	charac…	1	112 B	"a_dream"
x	numeric	3	80 B	num [1:3] 1 2 3
y	charac…	3	248 B	chr [1:3] "a" "b…

Console

```
> #c-combine fn
> x<-c(1,2,3)
> y<-c("a","b","c")
> e<- c(FALSE, TRUE, FALSE,FALSE, FALSE, TRUE)
>
```

Basic Calculations continued

Script Editor

```
49 #let us create a vector z
50 z<-c(-1,2,5,18,0,0,-98,-65,-69,0,54,-96,0,65,658,-987,0,-98,-98)
51 sign(z)
52 floor(14.668)
53 ceiling(13.854)
```

Console

```
> #let us create a vector z
> z<-c(-1,2,5,18,0,0,-98,-65,-69,0,54,-96,0,65,658,-987,0,-98,-98)
> sign(z)
 [1] -1  1  1  1  0  0 -1 -1 -1  0  1 -1  0  1  1 -1  0 -1 -1
> floor(14.668)
[1] 14
> ceiling(13.854)
[1] 14
>
```

To know class of a variable

Script Editor

```
91 class(x)
92 class(y)
93 class(e)
```

Console

```
> class(x)
[1] "numeric"
> class(y)
[1] "character"
> class(e)
[1] "logical"
>
```