

# 00-Introduction

*Adi Sarid / adi@sarid-ins.co.il*

*March 2019*

## Background

You are now viewing a document generated via R Markdown. R Markdown is a friendly format for writing R code along with documentation that surrounds it. It's a very powerful tool - a simple text document can then be compiled very easily to an html, a pdf, word and additional formats.

It is good for documenting and communicating your work to others, and we are going to rely on this format.

The aim of this exercise is to get you familiar with the RStudio interface, while writing your first R Markdown document along with some R code.

Ready? Let's get started!

## The console and some basic commands

You can notice that when you open RStudio, the window is divided into 4 segments. In the lower segment (on the left) you can see the console. The console can be used to run R commands. Try running some code in the console, i.e., copy and paste the following code (line by line).

Note that we are:

- Using `<-` which is the sign used to set a variable's value.
- We are using `a` and `b` as variable names.
- `cars`, which is an example dataset.
- `?` which is used to get help on commands.
- `plot()` which is used to generate a base-r plot.

```
1 + 1
a <- 1
b <- 2
a + b
cars
?cars
?plot
plot(cars)
```

## Answer/note these:

Decipher what each command did.

- a. Did you notice the help pop up when you used `?cars` and `?plot`? **Bottom right corner**
- b. Did you notice where the plot came up? **Bottom right corner again**

Look at the console and click Ctrl+L. What happend? **The console was cleared of all text and output**

Set the marker in the console window, click on Ctrl + up arrow. What happened? **You can view the history of latest commands used**

Finally, type the letter c in the console and click Ctrl + up arrow. What does this do? **You can view the history of latest commands used which start with a c**

---

## File types you can use in RStudio

Throughout the course we will rely heavily on writing in RMarkdown, however there are some more file types in RStudio. Use the file menu (File -> New -> R Script) to open up a new script, in the script type the code from the previous part above and click Ctrl + Shift + Enter. See what happens.

Now mark the last two lines and click Ctrl + Enter (without the shift). What is the difference? **Sourcing the entire file (running all the code in the script) versus just the marked lines.**

When do you think you would use a script file versus an R Markdown file? **If you ever need something “quick and dirty” use a script. If you need a thorough documentation, use an R Markdown.**

Another important option is using R Projects. When you open up a new project it will generate an environment file called .Rproj. This file will preserve the relative location of the directory, and this will make it slightly easier to load and save files from within R (it will be a huge benefit later on).

Let's start a new project: Click on File -> New Project Follow the wizard's instructions to open up a new project with project type “New Project”. Name it “My first R project”, and check the two checkboxes for “Create a .git repository” and for “Open in new session”.

In the new RStudio window that appeared, open up a new RMarkdown file, call it “My brand new RMarkdown”, keep the default output format as html, and click OK. Save the RMarkdown file inside the directory (click on the save button in the upper left corner or on File -> Save).

Now, click on “Knit” (Ctrl + Shift + K).

## Master additional shortcuts and RStudio panes.

Click on Alt+Shift+K. This is the shortcut window - it can help you if you ever forget shortcuts. Look at the RStudio window and try to answer the following short questions (please ask the one sitting next to you if you are lost!!!)

1. Where can you see all the environment variables currently loaded? **Environment pane on upper right corner**
2. Where can you access the history of all the commands you previously ran? **History pane also on the upper right corner**
3. Where would you look for the packages that are available to you? (RStudio has a pane for it, you just have to look for it) **Packages pane on the lower right corner**
4. If you are familiar with Git, how would you commit a file from within RStudio? **Git pane on the upper right, “Commit” button opens up a dialog for that.**
5. How would you find cheatsheets from within RStudio's menus? **Help->Cheatsheets**

---

Congratulations! you've completed your very first exercise in R. **Yay!**