

## **Molecular Approaches to Clinical Microbiology in Africa 2023**

2 - 8 September 2023

**Kenya Medical Research Institute (KEMRI)**

**Instructors: Leonard Ndwiga and Dr. Odile Harrison**

### **Topic: R programming with R Studio**

#### **Background**

R is a programming is an open-source language used for data manipulations, statistical analysis, and data visualization. Once R language is installed using an appropriate distributive from CRAN, the R language can be run using a console. However, many people prefer to use an Integrated Development Environment (IDE). There are many IDEs such as Visual Studio, Rattle e.t.c, but R Studio is the most popular. RStudio includes a free version as well as several enterprise products (e.g. RStudio Commercial Desktop and RStudio Server Pro) that require an annual license fee. The free version of RStudio makes the R language much easier to use and facilitates development of advanced R scripts.

#### **Learning Outcomes**

- Gain familiarity with R Studio
- Install and load add-in packages
- Import external data into R for data processing and statistical analysis
- compute basic summary statistics
- produce data visualisations with the ggplot package
- solve fundamental error problems

#### **Session 01: Introduction to R Studio**

In this session we will go straight into the R studio Graphical User Interface (GUI). In the top left corner of the screen one can see a script editor window. Within this pane one can edit their R script. The script can be run by selecting the lines that one wishes to be executed with a mouse and then pressing the “Run” button at the top of the script editing pane or use the “ctrl + enter” shortcut. The results of the script execution together with the script lines that generated these results will be displayed in the Console window located in the bottom left corner of the screen. The top right pane of the screen provides information about the variables and data structures used or generated by the script. This is the so-called “Environment” window. The window on the bottom right corner of the screen shows information about the files and packages used by the project and allows one to view plots (or visualizations) generated by R and also access help about various elements of R syntax.

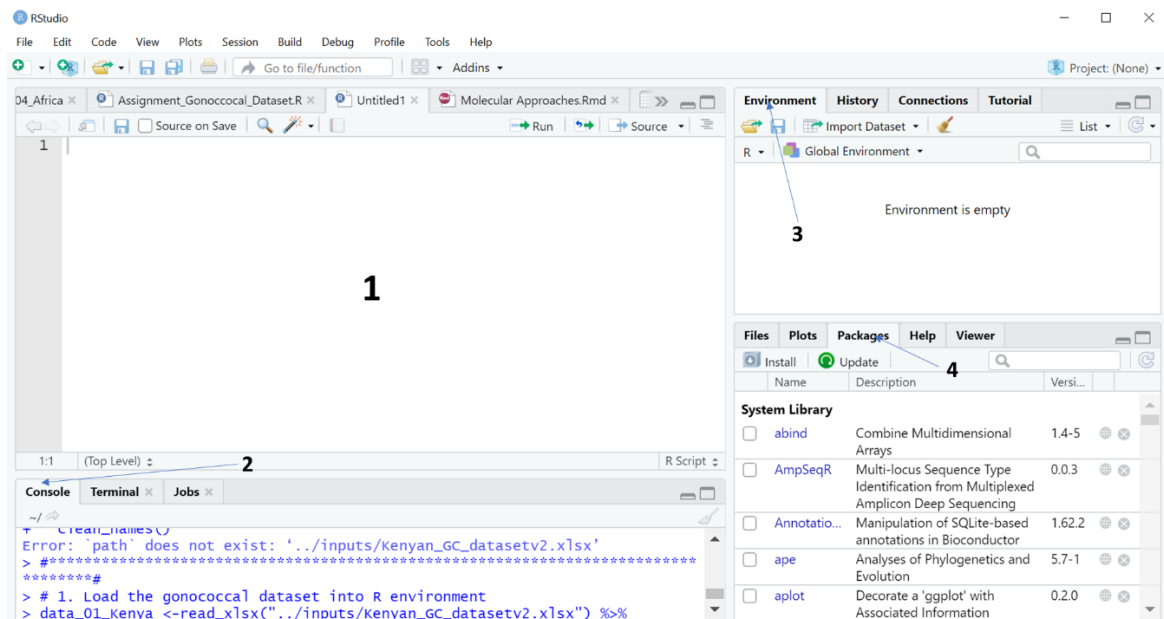
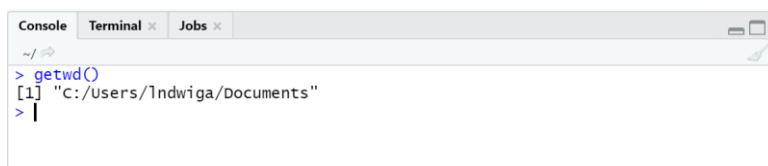


Figure 1: R Studio GUI: Among the many tabs on RStudio, it includes a script editor (1), console (2), Environment and (3) packages (4) tab.

## I. Getting your current working directory

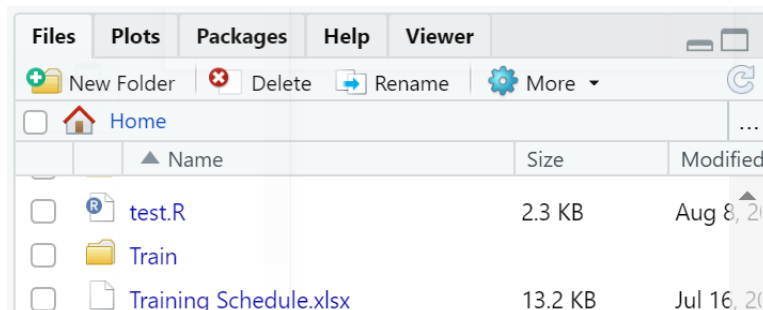
**Command:** `getwd()`



## II. Listing files in your current working directory

**Command:**

`list.files(path = ".")` OR `list.files(path = "./")` OR see the **Files** Tab on the bottom right panel

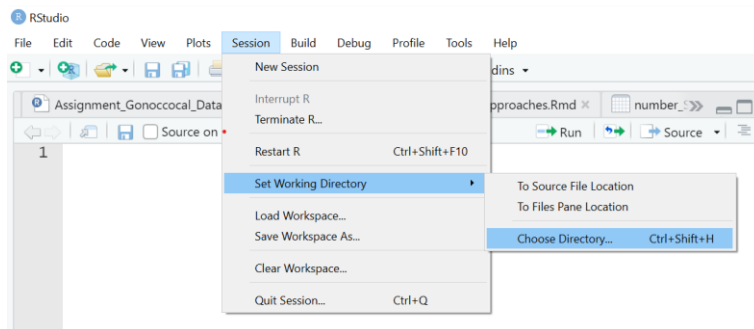


### III. Setting your working directory

**Command:**

```
setwd()
```

OR go to the **Sessions** tab -> **Set Working Directory** -> **Choose Directory**



### IV. Installing packages (require internet)

**Command:**

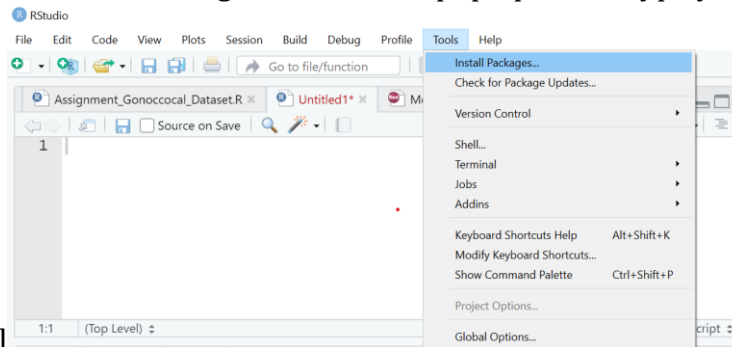
```
install.packages("packagename")
```

```
install.packages ("ggplot2") # install the package called ggplot2
```

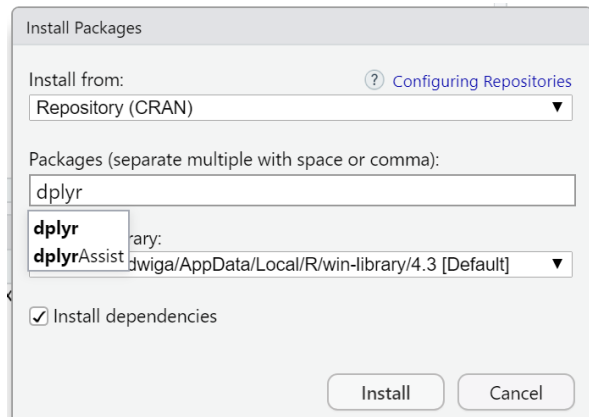
```
library ("ggplot2") # load the library ggplot2 help (package=ggplot2)
```

```
(package="package_name") #help to get help about a specific package
```

OR go to “Tools -> Install Packages” tab. On the pop up menu, type your desired package



and click install



## V. Comments on code

Use “#”. Comments can be used to annotate your code or separate different code chunks that have different functions



## VI. Assigning Variables

Values are assigned to named variables with an **<- (arrow)** or an **= (equal)** sign. In most cases they are interchangeable, however it is good practice to use the arrow since it is explicit about the direction of the assignment.

### Command:

# assign the number 7 to a variable named x

```
x <- 7
```

## **VII. Creating an R Studio Project**

File > New Project... > New Directory > New Project Directory name: MolAppAfrica\_2023  
Create project as a subdirectory of: ~/ Browse... (directory/folder to save the workshop data)

## **VIII. Loading Data into R Studio**

Most R users need to load their own datasets, usually saved as table files (e.g. Excel, or .csv files), to be able to analyse and manipulate them.

### **Command:**

```
# Load the file named "African_meningococci.xlsx"  
data_meningococci <- read_excel(path = "Africa_meningococci.xlsx")
```