IT424- Big Data Analytics

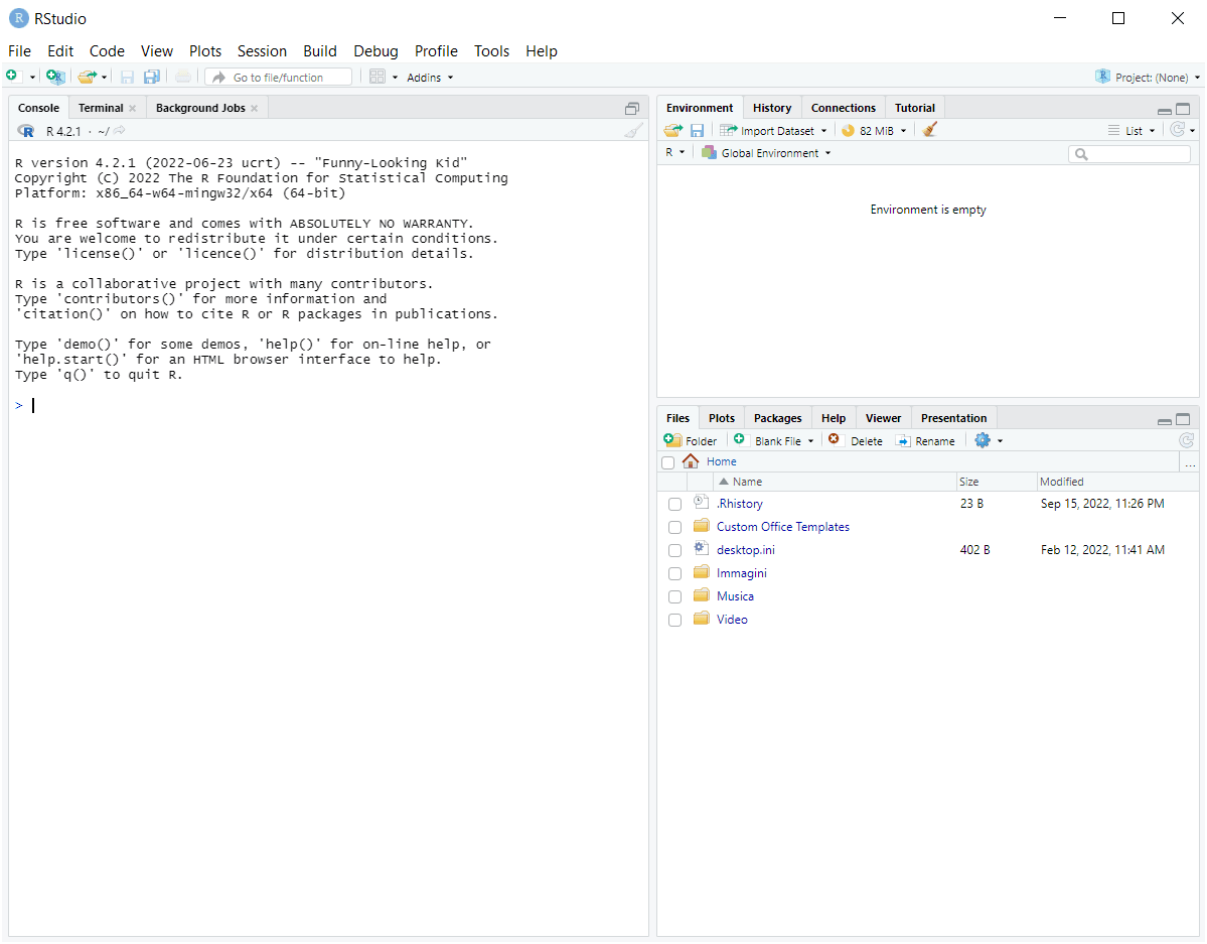
LAB 1- Write an R Program to solve roots of a quadratic equation

Perquisites-

Install R programming language and R studio

https://cran.r-project.org/

Once installed, open R studio



Roughly, we can divide the working window into three areas:

- **Left area**: includes the tabs Console, Terminal, and Background Jobs
- **Top-right area**: includes the tabs Environment, History, Connections, and Tutorial
- **Bottom-right area**: includes the tabs Files, Plots, Packages, Help, Viewer, and Presentation

Further R studio tutorial-

https://www.datacamp.com/tutorial/r-studio-tutorial

R basics-

R is a programming language and software environment for statistical analysis, graphics representation and reporting. R was created by ***Ross Ihaka and Robert Gentleman*** at the University of Auckland, New Zealand, and is currently developed by the R Development Core Team.
The core of R is an interpreted computer language which allows branching and looping as well as modular programming using functions.
R allows integration with the procedures written in ***the C, C++, .Net, Python or FORTRAN languages for efficiency.***
R is freely available under the GNU General Public License, and pre-compiled binary versions are provided for various operating systems like Linux, Windows and Mac.
R is free software distributed under a GNU-style copy left, and an official part of the GNU project called **GNU S**.

Features of R

As stated earlier, R is a programming language and software environment for statistical analysis, graphics representation and reporting. The following are the important features of R:

1. R is a well-developed, simple and effective programming language which includes conditionals, loops, user defined recursive functions and input and output facilities.

2. R has an effective data handling and storage facility.

3. R provides a suite of operators for calculations on arrays, lists, vectors and matrices.

4. R provides a large, coherent and integrated collection of tools for data analysis.

5. R provides graphical facilities for data analysis and display either directly at the computer or printing at the papers.

Basic Syntax-

```
> print("hello world")
[1] "hello world"

> 12.3+13.5
[1] 25.8

> a<-"Hello World"
> print(a)
[1] "Hello World"
```

R data types

Vectors
Lists

Matrices
Arrays
Factors
Data Frames

To find the data type, use class(variable)

Operators in R

Same (Arithmetic, Logical, Relational, Assignment, Miscellaneous)

Loop

Repeat Loop
While Loop
For Loop

Repeat Loop-

```
repeat {
commands
if(condition){
break
}
}
```

```
Ex: v <- c("Hello","loop")
cnt <- 2
repeat{
print(v)
cnt <- cnt+1
if(cnt > 5){
break
}
}
```

```
O/P- [1] "Hello" "loop"
[1] "Hello" "loop"
[1] "Hello" "loop"
[1] "Hello" "loop"
```

While Loop

```
while (test_expression) {
statement
}
```

```
v <- c("Hello","while loop")
cnt <- 2
while (cnt < 7){
```

```
print(v)
cnt = cnt + 1
}
```

For Loop

```
for (value in vector) {
statements
}
```

```
v <- LETTERS[1:4]
for ( i in v) {
print(i)
}
```

O/p- [1] "A"
[1] "B"
[1] "C"
[1] "D"

Loop break and Next (same as continue in C)

R- function

```
function_name <- function(arg_1, arg_2, ...) {
Function body
}
```

Built-in functions
User-Defined Functions

Built-in Functions

```
# Create a sequence of numbers from 32 to 44.
print(seq(32,44))
# Find mean of numbers from 25 to 82.
print(mean(25:82))
# Find sum of numbers frm 41 to 68.
print(sum(41:68))
```

User-Defined Function

```
# Create a function to print squares of numbers in sequence.
new.function <- function(a) {
for(i in 1:a) {
b <- i^2
print(b)
}
```

Call the function new.function supplying 6 as an argument.

```
new.function(6)

# Create a function with arguments.
new.function <- function(a,b,c) {
result <- a*b+c
print(result)
}
# Call the function by position of arguments.
new.function(5,3,11)
# Call the function by names of the arguments.
new.function(a=11,b=5,c=3)

# Create a function with arguments.
new.function <- function(a = 3,b =6) {
result <- a*b
print(result)
}
```

Program-

```
# Constructing Quadratic Formula
result <- function(a,b,c){
  if(delta(a,b,c) > 0){
    x_1 = (-b+sqrt(delta(a,b,c)))/(2*a)
    x_2 = (-b-sqrt(delta(a,b,c)))/(2*a)
    result = c(x_1,x_2)
  }
  else if(delta(a,b,c) == 0){
    x = -b/(2*a)
  }
  else {"There are no real roots."}
}

# Constructing delta
delta<-function(a,b,c){
  b^2-4*a*c
}
```

Examples

```
a <- result(1,-2,1); a
## [1] 1
b <- result(1,-4,1); b
## [1] 3.7320508 0.2679492
c <- result(4,-1,5); c
## [1] There are no real roots.
```