Getting Started

-Free and Open Source
-Multiplatform
*Utilizarei o símbolo > para indicar o input do console e >>> para o output
-As funções básicas são o R base, o restante são adicionados pela comunidade com CRAN ou github.
-É possível salvar os scripts para serem usados depois.

Installing Packages

install.packages("namepackage")

Como instalar um pacote:
install.packages("dslabs") //instalando o pacote dslabs
após instalar:
library (dslabs) //carregando o pacote dslabs

Para instalar mais de um pacote:
install.packages(**c**("pacote1", "pacote2"))

Também é possível utilizar o botão **tools - > install packages**

*Portanto, é importante ter um script que instale todos os pacotes que precisa, pois caso seja necessário reinstalar ou instalar uma nova versão do R, será necessário instalar os pacotes novamente.

*You can add the option dependencies = TRUE, which tells R to install the other things that are necessary for the package or packages to run smoothly. Otherwise, you may need to install additional packages to unlock the full functionality of a package.

Running Commands While Editing Scripts

Color and identation are automatically added in R studio.
O editor Rstudio também nos ajuda a testar nosso código enquanto editamos os scripts.
As primeiras linha de código em R geralmente são dedicadas a carregar as bibliotecas que utilizaremos.

Exemplo. (mostrar um gráfico de assassinatos vs população total)

```
library(tidyverse)
library(dslabs)
data(murders)

murders %>%
        ggplot(aes(population, total, label = abb, color = region)) +
        geom_label()
```

*Para testar apenas uma linha por vez se utilizar Ctrl-Enter
*Para testar o código todo Ctrl-Shift-Enter

Objects

Assignar variáveis:
a <- -1
b <- 2 //ou utilizar o simbolo = mas não é recomendado fazer isto

Para ver o **valor de um variável**:
>*a*
>>>1

Print:
print()
ex.
> *print(a)*
>>> 1

Ver as **variáveis salvas no workspace**:
ls()
ex.
>*ls()*
>>>"a" "b" "c"

Code: solving the equation $x^2 + x - 1 = 0$

```
# assigning values to variables
a <-1
b <-1
c <--1

# solving the quadratic equation
(-b + sqrt(b^2 - 4*a*c))/(2*a)
(-b - sqrt(b^2 - 4*a*c))/(2*a)
```

Functions

Funções:
ex. *log()*
>log(8)
caso digite log sem parêntese, a IDE mostrará o código de como usar o log

>>> 2.079
>exp(1) // e^1
>>>2.718
>log(2.718)
>>>1
>log(exp(1))
>>>1

Nested Functions:
Usar a função como argumento para outra função.
Ex.
log(exp(1))

Help system:
ex.
help("log") ou *?log*
ex.
help("+")
?"+" //note que nesse caso é necessário aspas para o operador
>...shows the help file… //arquivo mostra como a função log funciona
>help("+")
Para ver os **argumentos da função**:
>args(log)
>function (x, base=exp(1)) //quando se tem o igual temos o valor de default do argumento
Null
Como descobrimos como os argumentos funcionam, se quisermos o log de 8 na base 2:
>log(8,base=2)
>>>3

Ver os **nomes de objetos**:
data()
ex.
> pi
>>>3.1415
Co2
>>>Console mostra a data de pre-built objetcs para Co2

Variable names in R

Star with a letter
Can't contain spaces
ex.
solution_1
solution_2

Comments:
now commenting

Data Types

class():
ajuda a determinar a classe de um objeto
ex.
> a <- 2
> class(a)
>>> numerical

Data Frame:
>libray(dslabs)
>data("murders")
>class(murders)
> "data.frame"
str(muders) //str vem de structure e mostrará as estruturas de murders,
//como nome de colunas
>head(murders) //mostra as primeiras 5 linhas de murders

Acessar nome das colunas:
>names(murders)

Acessar as variáveis das colunas:
>$murders$population //mostra a coluna population

Vetores:
Valores da tabela

lenght:
>pop <- murders$population
>lenght(pop)
>>>51 //tamanho de populations em murders

Characteres Vectors:
>a <- 1
>a
>>>1
Caso queira saber o **valor da string** a:
>*"a"*
>>>"a"
ex. class(murders$state)
>"character"

Logical Vectors:

TRUE or FALSE
ex.
> z <- 3 == 2
> z
>>> false
> class(z)
>>> logical

Factor: //useful for storing categorical data
class(murders$region) //cada região é uma categoria
>"factor"
Para ver as categorias:
levels(murders$region)
>"North" "South" "west" "North Central" //*** não confundir com characters