Task1

Arijit Guchhait

2025-01-14

Overview

The main purpose of this task is to check that you can get it to load and run and knit it to html. Once that's working, have a go at the embedded exercises. Exercise 7 is just bonus. At the end, knit it to html and upload both files (Rmd and html) on blackboard (not zipped).

R and the RStudio IDE

R is a free software environment for statistical computing and graphics. We will use the RStudio IDE (Integrated Development Environment). This has a number of nice features which we'll cover as we go through - identify the following elements:

- Source Window
- Console
- Help
- Environment

Exercise 1a

Create variables for the following information: name (character): age (numeric): Your age is_local (logical): Do you live in Leicester? (TRUE/FALSE)

```
name <- 'Arijit Guchhait'
age <- 32
is_local <- TRUE
```

Exercise 1b

Is it possible to create a vector containing the values for your three variables? If not, create a named list containing the values for your three variables and print it.

```
#Vector only accept same datatypes
x <- list(a=name,b=age,c=is_local)
x
```

```
## $a
## [1] "Arijit Guchhait"
##
```

```
## $b
## [1] 32
##
## $c
## [1] TRUE
```

Exercise 2

Create a data frame called "students" with the following columns: Name (character): Names of students (include your name and at least three more names) Age (numeric): Ages of students Grade (factor): Grades of students (with levels: A, B, C) Is_Local (logical): Whether each student lives in Leicester (TRUE/FALSE)

```
students <- data.frame(
  Name=c('Arijit','Poulami','Rajita','Madhumita','Krittika'),
  Age= c(32,34,28,33,32),
  Grade = factor(c('A','A','B','B','C'),levels=c('A','B','C')),
  Is_Local =c(TRUE,FALSE,FALSE,FALSE,FALSE)
)
print(students)
```

```
##        Name Age Grade Is_Local
## 1    Arijit  32     A     TRUE
## 2   Poulami  34     A    FALSE
## 3    Rajita  28     B    FALSE
## 4 Madhumita  33     B    FALSE
## 5  Krittika  32     C    FALSE
```

Exercise 3

Let R show how many times the different levels appear in the grade-column

```
summary(students$Grade)
```

```
## A B C
## 2 2 1
```

Exercise 4

Extract and print the Age vector from the students data frame.

```
print(students$Age)
```

```
## [1] 32 34 28 33 32
```

Exercise 5

Look up the function "mean()" with the help function. Calculate the average age of the students from your dataframe and print it out. Use the pipe operator to do this calculation in one line of code.

```r
students$Age |> mean()
```

```
## [1] 31.8
```

Exercise 6

Why does the following code not run? Fix the Code (in the code chunk below named r "Exercise 6, solution"). Note, before you run the code, you need to get rid of the "#"s. You can do this by selecting everything and using the shortcut 'Ctrl Shift C'." (Commenting it out again works the same way).

```r
df <- data.frame(Name = c("John", "Jane", "Doe"), Age = c(25, 30,99))
print(df)
```

```
##   Name Age
## 1 John  25
## 2 Jane  30
## 3  Doe  99
```

```r
print("The problem was Age was not same dimension as name" )
```

```
## [1] "The problem was Age was not same dimension as name"
```

```r
age_in_10_years <- df$Age +10
mean(age_in_10_years)
```

```
## [1] 61.33333
```

```r
df <- data.frame(Name = c("John", "Jane", "Doe"), Age = c(25, 30,99))
print(df)
```

```
##   Name Age
## 1 John  25
## 2 Jane  30
## 3  Doe  99
```

```r
print("The problem was Age was not same dimension as name" )
```

```
## [1] "The problem was Age was not same dimension as name"
```

```r
age_in_10_years <- df$Age +10
mean(age_in_10_years)
```

```
## [1] 61.33333
```

Exercise 7* (bonus)

R contains some built-in data sets (that means they are included with the R installation and can be accessed without the need to load any external libraries.) One of them is "mtcars". Print it out and explore it using the functions head(), View(), summary(), glimpse(), str(). Try to understand what the functions tell you about the data set. Moreover, why should you not include the function "View()" when you try to knit your file?

```
head(mtcars)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
View(mtcars)
summary(mtcars)
```

```
##       mpg             cyl             disp             hp
##  Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0
##  1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5
##  Median :19.20   Median :6.000   Median :196.3   Median :123.0
##  Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7
##  3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0
##  Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0
##       drat             wt             qsec             vs
##  Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000
##  1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000
##  Median :3.695   Median :3.325   Median :17.71   Median :0.0000
##  Mean   :3.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375
##  3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000
##  Max.   :4.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000
##        am              gear            carb
##  Min.   :0.0000   Min.   :3.000   Min.   :1.000
##  1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000
##  Median :0.0000   Median :4.000   Median :2.000
##  Mean   :0.4062   Mean   :3.688   Mean   :2.812
##  3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :1.0000   Max.   :5.000   Max.   :8.000
```

```
dplyr::glimpse(mtcars)
```

```
## Rows: 32
## Columns: 11
## $ mpg  <dbl> 21.0, 21.0, 22.8, 21.4, 18.7, 18.1, 14.3, 24.4, 22.8, 19.2, 17.8,~
## $ cyl  <dbl> 6, 6, 4, 6, 8, 6, 8, 4, 4, 6, 6, 8, 8, 8, 8, 8, 8, 4, 4, 4, 4, 8,~
## $ disp <dbl> 160.0, 160.0, 108.0, 258.0, 360.0, 225.0, 360.0, 146.7, 140.8, 16~
## $ hp   <dbl> 110, 110, 93, 110, 175, 105, 245, 62, 95, 123, 123, 180, 180, 180~
## $ drat <dbl> 3.90, 3.90, 3.85, 3.08, 3.15, 2.76, 3.21, 3.69, 3.92, 3.92, 3.92,~
```

```
## $ wt   <dbl> 2.620, 2.875, 2.320, 3.215, 3.440, 3.460, 3.570, 3.190, 3.150, 3.~
## $ qsec <dbl> 16.46, 17.02, 18.61, 19.44, 17.02, 20.22, 15.84, 20.00, 22.90, 18~
## $ vs   <dbl> 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0,~
## $ am   <dbl> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0,~
## $ gear <dbl> 4, 4, 4, 3, 3, 3, 3, 4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 4, 4, 4, 3, 3,~
## $ carb <dbl> 4, 4, 1, 1, 2, 1, 4, 2, 2, 4, 4, 3, 3, 3, 4, 4, 4, 1, 2, 1, 1, 2,~
```

```
str(mtcars)
```

```
## 'data.frame':    32 obs. of  11 variables:
##  $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
##  $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
##  $ disp: num  160 160 108 258 360 ...
##  $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
##  $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
##  $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
##  $ qsec: num  16.5 17 18.6 19.4 17 ...
##  $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
##  $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
##  $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
##  $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

```
#Because it will open in another windows
```