```r
####### COURSE: Basics of R programming language for statistical analysis #######

#Instructor: Marina FERENT-PIPAS
#email: marinaferent@gmail.com
#accredited by Multicultural Business Institute | mcb-institute.org

#_______ CHAPTER 2: CONTROL STRUCTURES AND FUNCTIONS | Statistical measures _____#
########## MEETING 03: FOR LOOPS| Challenge: Mean values| 25.08.2021 #############

#_____________________________________________________________________________#
#################### GOOD PROGRAMMING PRACTICES OF THE MEETING:
    ##1. DESIGN: Before coding a section - write down each step in order
    ##2. INDENTATION: Showcase the begining of a section, the body and the end of it
    through the right indentation.
            ##Anything that subordinates to a line is TABed once from that line.

#################### FOCUS OF THE MEETING: FOR LOOPS
##alow us to repeat the same instruction for a given number of times
    ###for aditional information on loops and usage in R, check:
    https://tinyurl.com/yy5mmxzc
#_____________________________________________________________________________#

1:10        #creates a sequence of integers from 1 to 10

1+5         #adds up 1 and 5
i=1         #i is 1
print (i+5) #prints i+5 = 1+5=6

2+5         #adds up 2 and 5
i=2         #i=2
print (i+5) #prints i+5 = 2+5=7

#EXERCISE 1: Create a for loop for i taking values from 1 to 10 and print the sum of i
and 5.

for(i in 1:10)      #i takes values from 1 to 10
{                   #begining of for loop
    print(i+5)      #what happens in the loop
}                   #end of for loop

#i=1 =>begin loop: 1+5 => 6 end loop
#i=2 =>begin loop: 2+5 => 7 end loop
#...
#i=10 =>begin loop: 10+5 =>15 end loop
#max value i is 10 so loop ends here

#END OF EXERCISE 1 HERE_________________

i=1                         #i is 1
print (sum(i-1,i,5))        #1-1+1+5=0+1+5=6

i=2                         #i is 2
print (sum(i-1,i,5))        #2-1+2+5=1+2+5=8


#EXERCISE 2: Reproduce the above for i taking values from 1 to 10.
for(i in 1:10)                  #i takes values from 1 to 10
{                               #begining of for loop
    print (sum(i-1,i,5))        #what happens in the loop: i-1+i+5
}                               #end of for loop

#i=1 => 1-1+1+5=0+1+5=6 etc.

#END OF EXERCISE 2 HERE_________________
```

```
67
68
69
70  #EXERCISE 3: What are the lines below doing?
71
72  j=0                              #j is 0
73  for(i in 1:10)                   #i takes values from 1 to 10
74  {
75      print (sum(j,i,5))           #j+i+5
76      j=j+1                        #j increases by 1, then i increases by 1, then j+i+5,
        then j increases by 1 etc.
77  }
78
79
80  #END OF EXERCISE 3 HERE_________________
81
82  #EXERCISE 4: What is the difference between EXERCISE 3 and the lines below? -> [Nested
    for loops]
83  for(i in 1:10)
84  {
85      for(j in 0:9)
86      {
87          print (sum(j,i,5))
88      }
89  }
90
91  #in exercise 3:
92  #j=0, i=1 =>0+1+5=6
93  #j=1, i=2 =>1+2+5=8
94  #....
95  #j=9, i=10 =>9+10+5=24
96
97  #in exercise 4:
98  #i=1
99  ##j=0=> 0+1+5=6
100 ##j=1=> 1+1+5=7
101 ##....
102 ##j=9 => 9+1+5=15
103 #only now i=2
104 ##j=0=> 0+2+5=7 etc
105
106
107 #END OF EXERCISE 4 HERE_________________
108
109
110
111 seq(1, 10, 1)        #creates a sequence of integers from 1 to 10 with step 1: 1,
    2,3,...,10
112 seq(1, 10, 2)        #creates a sequence of integers from 1 to 10 with step 2: 1,
    3,5,...,9
113 seq(1, 10, 3)        #creates a sequence of integers from 1 to 10 with step 3: 1,
    4,7,...,10
114
115 constant=5
116 i=1
117 print (i+constant)
118
119 i=3
120 print (i+constant)
121
122 #EXERCISE 5: Reproduce the above for i taking values 1, 3, 5, 7 and 9.
123 constant=5
124 for(i in seq(1, 10, 2))
125 {
126     print (i+constant)
127 }
128
129 #END OF EXERCISE 5 HERE_________________
130
```

```r
#EXERCISE 6: Create a vector containing all the values from 11 to 20. Name it <<vector>>. How many elements does the vector have?
vector=11:20

    #alternatively:
vector=c()
for(i in 11:20)
{
    vector=c(vector, i)
}

length(vector)
#END OF EXERCISE 6 HERE_________________

vector[1]
vector[2]
#EXERCISE 7: Reproduce the above for all the elements in vector <<vector>> using a for loop.
for(i in 1:length(vector))
{
    print(vector[i])
}

#END OF EXERCISE 7 HERE_________________


vector[1]+vector[2]
vector[2]+vector[3]

#EXERCISE 8: Reproduce the above for all the elements in vector <<vector>> using a for loop.

#vector[1]+vector[2] .... vector[length(vector)-1]+vector[length(vector)]
#for loop opt 1. i=1:9: vector[i]+vector[i+1]
#         opt 2. i=2:10: vector[i-1]+vector[i]
#daca i= 1:10, pentru i=10: vector[i]+vector[i+1]=vector[10]+vector[11]=20+NA=NA
#daca i= 1:10, pentru i=1: vector[i-1]+vector[i]=vector[0]+vector[1]=NA+11=NA



for(i in 1:9)
{
  print(vector[i]+vector[i+1])
}

  ##alternatively
for(i in 1:length(vector)-1)
{
  print(vector[i]+vector[i+1])
}
  ##or:
for(i in 2:10)
{
  print(vector[i-1]+vector[i])
}

  ##alternatively
for(i in 2:length(vector))
{
  print(vector[i-1]+vector[i])
}

#END OF EXERCISE 8 HERE_________________
```

```r
vector[1]+vector[3]
vector[3]+vector[5]

#EXERCISE 9: Reproduce the above for all the elements in vector <<vector>> using a for loop.

for(i in seq(1,8,2))
{
  print(vector[i]+vector[i+2])
}

  ##alternatively:

for(i in seq(1,length(vector)-2,2))
{
  print(vector[i]+vector[i+2])
}

#END OF EXERCISE 9 HERE_________________


#EXERCISE 10: Create an empty matrix called <<matrice>>, of 10 rows and 5 columns.
matrice=matrix(NA, nrow=10, ncol=5)

View(matrice)
dim(matrice)
matrice[,1]
matrice[1,]
matrice[1,2]

#END OF EXERCISE 10 HERE_________________


i=1
matrice[,i]

i=2
matrice[,i]

i=1
matrice[i,]

i=2
matrice[i,]

i=1
j=1
matrice[i,j]

i=1
j=2
matrice[i,j]

i=1
j=1
matrice[i,j]=i+j

i=1
j=2
matrice[i,j]=i+j

i=10
j=5
matrice[i,j]=i+j

#EXERCISE 11: Reproduce the above using 2 for loops (for all cells in matrix <<matrice>>) => NESTED FOR LOOPS
matrice=matrix(NA, nrow=10, ncol=5) #creating a matrix of 10 rows and 5 columns full of NAs
```

```r
for(i in 1:10)
{                           #beginning of for i loop
  for(j in 1:5)
  {                         #beginning of for j loop
    matrice[i,j]=i+j
  }                         #end of for j loop
}                           #end of for i loop

    #alternatively:
for(i in 1:nrow(matrice))
{                           #beginning of for i loop
  for(j in 1:ncol(matrice))
  {                         #beginning of for j loop
    matrice[i,j]=i+j
  }                         #end of for j loop
}                           #end of for i loop

View(matrice)

#END OF EXERCISE 11 HERE_________________




#__________ TODAY'S R CHALLENGE: Mean Values _____________#

#EXERCISE 1: Import the <<Campus crimes.csv>> into R.
setwd("E:/Work/Multicultural Business Institute/R/R_FBE2021/Meeting 3") #sets the working directory. all the files exported/imported following this line, are exported to/imported from this location
campusData=read.csv("Campus crimes.csv")                ##imports the campusData from a .csv file called Campus crimes

View(campusData)    #visualizes the campusData as a table
##We visualized together the campusData and observed that the names of the variables were not correctly imported, as such we changed them:
colnames(campusData)=c("enrollment", "privateCollege", "police", "crime") #changes campusData column names

#EXERCISE 2: Compute the average number of enrolled students per college.
mean(campusData[,1]) #computes the mean
    #alternatively:
sum(campusData[,1])/length(campusData[,1]) #insumeaza toate elementele vectorului si imparte suma la lungimea vectorului (nr colegii)
    ##where:
    ###sum(campusData[,1]) sums up all the values in column 1
    ###length(campusData[,1]) computes the number of values in column 1 (=number of colleges in dataset) = nrows(campusData)

    #we may store the value in vector meanValue for future use:
meanValue=sum(campusData[,1])/length(campusData[,1])
meanValue
#EXERCISE 3: Compute the mean values for each variable in campusData
means=colMeans(campusData) #computes the means of each column in campusData and stores them into a table:
                                ##rownames: variable names in campusData = colNames campusData
View(means)             #visualize the means table

    #alternatively:

for(i in 1:ncol(campusData))    #i takes values from 1 to number of columns of <<campusData>>
{                               #beggining of loop
    print(sum(campusData[,i])/length(campusData[,i])) ##we compute the mean of variable i as the sum of all elements in <<campusData>> column i divided to our sample size = number of colleges = length of each column=number of rows in <<campusData>>
}                               #ending of loop

```

```r
    #we may store the values into a matrix for future use:

#Step 1: We create an empty matrix of 4 rows and 1 column:
    ##each cell in the matrix has value NA
    ##the matrix has the same number of rows as many variables as campusData has
means=matrix(NA, nrow=ncol(campusData), ncol=1)
rownames(means)=colnames(campusData) #means will have the same rownames as the colnames
of campusData (variable name)

#Step 2: We write in each cell of our <<means>> matrix step by step:
##first: i=1, we write in <<means>> row 1 variable mean of <<campusData>> column 1
##then: i=2, we write in <<means>> row 2 variable mean of <<campusData>> column 2
##then: i=3, we write in <<means>> row 3 variable mean of <<campusData>> column 3
##then: i=4, we write in <<means>> row 4 variable mean of <<campusData>> column 4
##then: we finished, since i takes values from 1 to 4

for(i in 1:ncol(campusData)) #i takes values from 1 to number of columns of
<<campusData>>
{                            #beggining of loop
  #in matrix <<means>>, row i, column 1 we store the mean of the variable in
  <<campusData>> column i
  ##we compute the mean of variable i as the sum of all elements in <<campusData>>
  column i divided to our sample size = number of colleges = number of rows in
  <<campusData>>
  means[i,1]=sum(campusData[,i])/length(campusData[,i]
}                            #ending of loop

    #alternative way of writing the for loop:

for(i in seq(1, ncol(campusData),1)) #i takes values from 1 to number of columns of
<<campusData>>, by 1: 1, 2, 3, 4
                                        ##useful when taking each 2nd, 3rd etc element
                                        -eg: by 2: seq(1, 100, 2), by 3: seq(1, 100, 3 )
{                            #beggining of loop
  #in matrix <<means>>, row i, column 1 we store the mean of the variable in
  <<campusData>> column i
  ##we compute the mean of variable i as the sum of all elements in <<campusData>>
  column i divided to our sample size = number of colleges = number of rows in
  <<campusData>>
  means[i,1]=sum(campusData[,i])/nrow(campusData)
}                            #ending of loop
```