LAB 5 FOUNDATIONS OF DATA ANALYTICS

MEHER SHRISHTI NIGAM

20BRS1193

CODE

```
# Lab 5
# L7+L8
# Meher Shrishti Nigam
# CSE AI + Robotics
# 20BRS1193

# FDA Lab Experiment-5(9/9/2022)

# Q1. Create a data frame for the sports day events as shown in Table 1. It records the points
# scored by three teams namely â�“ Orange, Yellow and Blue Teams in various sports day events.
sport <- c("Football", "Cricket", "Throw Ball", "Badminton", "Track and Field")
orange <- c(50, 25, 19, 30, 23)
yellow <- c(0, 20, 25, 43, 21)
blue <- c(20, 45, 26, 21, 0)
sports_teams <- data.frame(orange, yellow, blue)
row.names(sports_teams) <- sport
sports_teams

# a) Include a new team â��Greenâ�� with the score (10,10,30,30,40)

sports_teams$green <- c(10, 10, 30, 30, 40)

# b)  Display the scores of Football and Throw ball events.
```

```
selection <- c("Football", "Throw Ball")

sports_teams[selection,]


selection <- c(1, 3)

sports_teams[selection,]
```

# c) Find the total points scored by each team and bind the total score with the data set.

```
# t<-colSums(match[,2:5])

total_points <- colSums(sports_teams)

total <- c(total_points)

sports_teams <- rbind(sports_teams, total)

sport <- c(sport, "Total Points")

row.names(sports_teams) <- sport

View(sports_teams)

sports_teams
```

# d) Display the name of the winning team.

```
which.max(sports_teams[c("Total Points"),])
```

# e) Find the least points scored by Orange Team and Blue Team.

```
min(sports_teams$orange)

min(sports_teams$blue)
```

# f) Display the favourable game of Yellow Team and Green Team.

```
max(sports_teams$orange)
```

```r
max(sports_teams$blue)
```

# g) Display the average score of each event with the corresponding sports name.

```r
average_points <- rowMeans(sports_teams)
sports_teams$average <- average_points
sports_teams
```

# Q2. Assume that a Fall semester registration report contains various fields such as registration no.,

# name of the student, course code, course name, credits, faculty name and slot.

# A) Identify a suitable data structure to store the names of the courses offered in the Fall semester

# and justify your choice.

# We use a vector to store the names of courses

# B) Write an R code to do the following

# i) Create the identified data structure with a sample set of courses.

```r
course_name <- c("C++", "Python", "Java", "OOPS", "R", "HTML/CSS", "JavaScript")
```

# ii) Extract the course names in the data structure stored at locations 2, 5 and 7.

```r
course_name[c(2,5,7)]
```

# iii) Extract all the course names except at locations 1, 2 and 3.

```r
selection <- 4:7
course_name[selection]
```

# C) Suggest a suitable data structure to store the values of all fields for a

```r
# student and justify your choice.

# We use a data frame as our data set has varied datatypes and contains necessary fields

# D) Write an R code to do the following: [4 marks]

# I) Create the identified data structure with a sample values.
reg_no <- c("101","102","103","104","105","106","107")
names <- c("John", "Jake", "Jackson", "Jeff", "James", "Jessie", "Jack")
course_code <- c("CSE101","CSE102","CSE103","CSE104","CSE105","CSE106","CSE107")
course_name <- c("C++", "Python", "Java", "OOPS", "R", "HTML/CSS", "JavaScript")
credits <- c(4,2,3,3,4,3,2)
faculty_name <- c("Dr. Samuel", "Dr. Sandrah", "Dr. Sanjay", "Dr. Shashwat", "Dr. Sean", "Dr. Sebastien", "Dr. Saif")
slot <- c("L33 + 34", "L11 + 12", "A2 + TA2", "B1", "C2", "E1 + TE1", "F2")

registration_report <- data.frame(reg_no, names, course_code, course_name, credits, faculty_name, slot)
registration_report

# II) Assign names to all the values stored in the data structure.
rownames <- c("Student 1", "Student 2", "Student 3", "Student 4", "Student 5", "Student 6", "Student 7")
row.names(registration_report) <- rownames
# Columns are already named

# III) Extract the name of the student, course name, credits and slot in any 2 ways.

registration_report[c("names", "course_name", "credits", "slot")]

subset(registration_report, select = c("names", "course_name", "credits", "slot"))

# Q3) Write a R code snippet to do the following
# i) Create the above Matrix and display the matrix


stars <- matrix(c("Satchin", "Virat", "Rohit",
"Dhoni","Amitab","Amir","Akhya","Salman","Modi","Amit","Rahul","Neheru","Delhi","Chennai","Kolkata","Mumbai"
), nrow = 4)

stars


# ii) Define the column names as Players, Actors, Politicians, Metro city

column_names <- c("Players", "Actors", "Politicians", "Metro Cities")

colnames(stars) <- column_names

stars


# iii) Define the row names as Record1, Record2, Record3, Record4

row_names <- c("Record1", "Record2", "Record3", "Record4")

rownames(stars) <- row_names

stars


# iv) Display only the names of cricket players and Politicians

stars[,c("Players", "Politicians")]


# v) Display the records that contain Kolkata and Mumbai

stars[c(3,4),]


# Q4. As a data analyst, you have been asked to retrieve the following information from the given

#string “8/08/2022”. Write an R code snippet to do the following (5x1=5 Marks)


#i) Display the given string in date format

date <- "08/08/2022"

date <- as.Date(date, format ="%d/%m/%Y")

date
```

```r
#ii) As a data analyst extract from the date object, the day number in that year, for example,
#8/25/2022 is 237th day in this year.
day_no <- format(date, format = "%j")
day_no

#iii) As a data analyst extract from the date object, the month number in that year. For example,
#08/25/2022 has the month number 08.
month <- format(date, format = "%m")
month

#iv) Convert the following dates which are in string format to date format
#12/11/2010”, “13/12/1990”,”30/1/2001” and “15/08/2022” and display the dates on the
#console.
date1 <- "12/11/2010"
date1 <- as.Date(date1, format ="%d/%m/%Y")
date1

date2 <- "13/12/1990"
date2 <- as.Date(date2, format ="%d/%m/%Y")
date2

date3 <- "30/1/2001"
date3 <- as.Date(date3, format ="%d/%m/%Y")
date3

date4 <- "15/08/2022"
date4 <- as.Date(date3, format ="%d/%m/%Y")
date4
```

```
#v) Extract two dates “30/01/2001” and “12/11/2010” from the above created object and
#calculate the difference between the given dates in terms of days and months and display
#the results.

daysdiff <- difftime(date3, date1, units="days")
daysdiff

monthsdiff <- as.numeric(daysdiff/30, units="days")
monthsdiff
```

OUTPUT:

```
> # Q1. Create a data frame for the sports day events as shown in Table 1. It records the points
> # scored by three teams namely â“ Orange, Yellow and Blue Teams in various sports day events.
> sport <- c("Football", "Cricket", "Throw Ball", "Badminton", "Track and Field")
> orange <- c(50, 25, 19, 30, 23)
> yellow <- c(0, 20, 25, 43, 21)
> blue <- c(20, 45, 26, 21, 0)
> sports_teams <- data.frame(orange, yellow, blue)
> row.names(sports_teams) <- sport
> sports_teams
                orange yellow blue
Football            50      0   20
Cricket             25     20   45
Throw Ball          19     25   26
```

```
Badminton          30    43   21
Track and Field     23     21    0
> sports_teams$green <- c(10, 10, 30, 30, 40)
> selection <- c("Football", "Throw Ball")
> sports_teams[selection,]
           orange yellow blue green
Football       50      0   20    10
Throw Ball     19     25   26    30
> selection <- c(1, 3)
> sports_teams[selection,]
           orange yellow blue green
Football       50      0   20    10
Throw Ball     19     25   26    30
> # t<-colSums(match[,2:5])
> total_points <- colSums(sports_teams)
> total <- c(total_points)
> sports_teams <- rbind(sports_teams, total)
> sport <- c(sport, "Total Points")
> row.names(sports_teams) <- sport
> View(sports_teams)
> # View(sports_teams)
> sports_teams
                orange yellow blue green
Football            50      0   20    10
```

```
Cricket          25    20   45    10

Throw Ball        19    25   26    30

Badminton          30    43   21    30

Track and Field     23    21    0    40

Total Points      147    109  112   120

> which.max(sports_teams[c("Total Points"),])

orange

    1

> min(sports_teams$orange)

[1] 19

> min(sports_teams$blue)

[1] 0

> max(sports_teams$orange)

[1] 147

> max(sports_teams$blue)

[1] 112

> average_points <- rowMeans(sports_teams)

> sports_teams$average <- average_points

> sports_teams

               orange yellow blue green average

Football          50     0   20    10     20

Cricket          25    20   45    10     25

Throw Ball        19    25   26    30     25

Badminton          30    43   21    30     31
```

```
Track and Field     23     21    0    40      21

Total Points       147    109  112   120     122
```

Q2)

```
> # i) Create the identified data structure with a sample set of courses.

> course_name <- c("C++", "Python", "Java", "OOPS", "R", "HTML/CSS",
"JavaScript")

> # ii) Extract the course names in the data structure stored at locations 2, 5 and
7.

> course_name[c(2,5,7)]

[1] "Python"     "R"          "JavaScript"

> # iii) Extract all the course names except at locations 1, 2 and 3.

> selection <- 4:7

> course_name[selection]

[1] "OOPS"       "R"          "HTML/CSS"   "JavaScript"

> # I) Create the identified data structure with a sample values.

> reg_no <- c("101","102","103","104","105","106","107")

> names <- c("John", "Jake", "Jackson", "Jeff", "James", "Jessie", "Jack")

> course_code <-
c("CSE101","CSE102","CSE103","CSE104","CSE105","CSE106","CSE107")

> course_name <- c("C++", "Python", "Java", "OOPS", "R", "HTML/CSS",
"JavaScript")

> credits <- c(4,2,3,3,4,3,2)

> faculty_name <- c("Dr. Samuel", "Dr. Sandrah", "Dr. Sanjay", "Dr. Shashwat",
"Dr. Sean", "Dr. Sebastien", "Dr. Saif")

> slot <- c("L33 + 34", "L11 + 12", "A2 + TA2", "B1", "C2", "E1 + TE1", "F2")
> registration_report <- data.frame(reg_no, names, course_code, course_name, credits, faculty_name, slot)

> registration_report

  reg_no   names course_code course_name credits  faculty_name     slot

1    101    John      CSE101        C++       4    Dr. Samuel L33 + 34

2    102    Jake      CSE102      Python       2   Dr. Sandrah L11 + 12

3    103 Jackson      CSE103       Java       3    Dr. Sanjay A2 + TA2

4    104    Jeff      CSE104       OOPS       3  Dr. Shashwat       B1

5    105   James      CSE105          R       4      Dr. Sean       C2

6    106  Jessie      CSE106    HTML/CSS       3 Dr. Sebastien E1 + TE1

7    107    Jack      CSE107  JavaScript       2      Dr. Saif       F2

> # II) Assign names to all the values stored in the data structure.

> rownames <- c("Student 1", "Student 2", "Student 3", "Student 4", "Student 5", "Student 6", "Student 7")

> row.names(registration_report) <- rownames

> registration_report[c("names", "course_name", "credits", "slot")]

            names course_name credits     slot

Student 1    John         C++       4 L33 + 34

Student 2    Jake      Python       2 L11 + 12

Student 3 Jackson        Java       3 A2 + TA2

Student 4    Jeff        OOPS       3       B1

Student 5   James           R       4       C2

Student 6  Jessie    HTML/CSS       3 E1 + TE1

Student 7    Jack  JavaScript       2       F2
```

```
> subset(registration_report, select = c("names", "course_name", "credits",
"slot"))

          names course_name credits     slot

Student 1    John        C++       4 L33 + 34

Student 2    Jake      Python       2 L11 + 12

Student 3 Jackson       Java       3 A2 + TA2

Student 4    Jeff       OOPS       3       B1

Student 5   James          R       4       C2

Student 6  Jessie    HTML/CSS       3 E1 + TE1

Student 7    Jack  JavaScript       2       F2
```

Q3)

```
> stars <- matrix(c("Satchin", "Virat", "Rohit",
"Dhoni","Amitab","Amir","Akhya","Salman","Modi","Amit","Rahul","Neheru","D
elhi","Chennai","Kolkata","Mumbai"), nrow = 4)

> stars

     [,1]      [,2]     [,3]     [,4]

[1,] "Satchin" "Amitab" "Modi"   "Delhi"

[2,] "Virat"   "Amir"   "Amit"   "Chennai"

[3,] "Rohit"   "Akhya"  "Rahul"  "Kolkata"

[4,] "Dhoni"   "Salman" "Neheru" "Mumbai"

> # ii) Define the column names as Players, Actors, Politicians, Metro city

> column_names <- c("Players", "Actors", "Politicians", "Metro Cities")

> colnames(stars) <- column_names

> stars

     Players   Actors   Politicians Metro Cities
```

```
[1,] "Satchin" "Amitab" "Modi"      "Delhi"
[2,] "Virat"   "Amir"   "Amit"      "Chennai"
[3,] "Rohit"   "Akhya"  "Rahul"     "Kolkata"
[4,] "Dhoni"   "Salman" "Neheru"    "Mumbai"
> # iii) Define the row names as Record1, Record2, Record3, Record4
> row_names <- c("Record1", "Record2", "Record3", "Record4")
> rownames(stars) <- row_names
> stars
        Players   Actors   Politicians Metro Cities
Record1 "Satchin" "Amitab" "Modi"      "Delhi"
Record2 "Virat"   "Amir"   "Amit"      "Chennai"
Record3 "Rohit"   "Akhya"  "Rahul"     "Kolkata"
Record4 "Dhoni"   "Salman" "Neheru"    "Mumbai"
> # iv) Display only the names of cricket players and Politicians
> stars[,c("Players", "Politicians")]
        Players   Politicians
Record1 "Satchin" "Modi"
Record2 "Virat"   "Amit"
Record3 "Rohit"   "Rahul"
Record4 "Dhoni"   "Neheru"
> # v) Display the records that contain Kolkata and Mumbai
> stars[c(3,4),]
        Players Actors   Politicians Metro Cities
Record3 "Rohit" "Akhya"  "Rahul"     "Kolkata"
```

```
Record4 "Dhoni" "Salman" "Neheru"    "Mumbai"
```

Q4)

```
> #i) Display the given string in date format
> date <- "08/08/2022"
> date <- as.Date(date, format ="%d/%m/%Y")
> date
[1] "2022-08-08"
> #ii) As a data analyst extract from the date object, the day number in that year, for example,
> #8/25/2022 is 237th day in this year.
> day_no <- format(date, format = "%j")
> day_no
[1] "220"
> #iii) As a data analyst extract from the date object, the month number in that year. For example,
> #08/25/2022 has the month number 08.
> month <- format(date, format = "%m")
> month
[1] "08"
> #iv) Convert the following dates which are in string format to date format
> #12/11/2010", "13/12/1990","30/1/2001" and "15/08/2022" and display the dates on the
> #console.
> date1 <- "12/11/2010"
> date1 <- as.Date(date1, format ="%d/%m/%Y")
> date1
[1] "2010-11-12"
> date2 <- "13/12/1990"
> date2 <- as.Date(date2, format ="%d/%m/%Y")
> date2
[1] "1990-12-13"
> date3 <- "30/1/2001"
> date3 <- as.Date(date3, format ="%d/%m/%Y")
> date3
[1] "2001-01-30"
> date4 <- "15/08/2022"
> date4 <- as.Date(date3, format ="%d/%m/%Y")
> date4
[1] "2001-01-30"
> daysdiff <- difftime(date3, date1, units="days")
> daysdiff
Time difference of -3573 days
> monthsdiff <- as.numeric(daysdiff/30, units="days")
> monthsdiff
[1] -119.1
>
```