**DAY-2 EXERCISE**

**ITA0443-STATISTICS WITH R PROGRAMMING**

**INDU SEKHAR**

**192124123**

**Exercise: 1**

**Construct the following data frame ‘country’.**

# create data frame

country<-data.frame(c("A","B","C"),c(100,200,120),c(2000,7000,15000))

colnames(country)<- c("countries","population\_in\_million","gdp\_percapita")

country

**a) Reshape in R from wide to long:**

## reshape in R from wide to long example

country\_w\_to\_L<- reshape(data=country, idvar="countries",

varying = c("population\_in\_million","gdp\_percapita"),

v.name=c("value"),

times=c("population\_in\_million","gdp\_percapita"),

new.row.names = 1:1000,

direction="long")

**b) Reshape in R from long to wide:**

## reshape in R from long to wide example

country\_L\_to\_w <- reshape(data=country\_w\_to\_L,idvar="countries",

v.names = "value",

timevar = "time",

direction="wide")

**7. MELTING AND CASTING IN R**

**Exercises :**

**1. Melt airquality data set and display as a long – format data ?**

**CODE:**

library(reshape2)

data("airquality")

airquality\_melted <- melt(airquality, id.vars = c("Month", "Day"))

head(airquality\_melted)

**2. Melt airquality data and specify month and day to be “ID variables” ?**

**CODE:**

airquality\_melted2 <- melt(airquality, id.vars = c("Month", "Day"), value.name = "value", variable.name = "variable")

head(airquality\_melted2)

**3. Cast the molten airquality data set .**

**CODE:**

airquality\_cast <- dcast(airquality\_melted2, Month + Day ~ variable, value.var = "value")

head(airquality\_cast)

**4. Use cast function appropriately and compute the average of Ozone, Solar.R , Wind and temperature per month ?**

**CODE:**

airquality\_avg <- dcast(airquality\_melted2, Month ~ variable, value.var = "value", fun.aggregate = mean)

airquality\_avg[,c("Ozone", "Solar.R", "Wind", "Temp")]

**8 FILE MANUPULATION IN R**

**Exercise**

**1. Consider the following data present. Create this file using windows notepad . Save the file as input.csv using the save As All files(\*.\*) option in notepad.**

**CODE:**

* Open notepad & type the below data

Id,name,salary,start\_date,dept

1,Rick,623.3,2012-01-01,IT

2,Dan,515.2,2013-09-23,operations

3,Michelle,611,2014-11-15,IT

4,Ryan,729,2014-05-11,HR

5,Gary,843.25,2015-03-27,Finance

6,Nina,578,2013-05-21,IT

7,Simon,632.8,2013-07-30,operation

8,Guru,722.5,2014-06-17,Finance

* Go to File > Save As.
* In the "Save As" dialog, enter the file name "input.csv".
* Change the "Save as type" to "All files (\*.\*)".
* Save the file in the desired location.

**2. Use appropriate R commands to read input.csv file.**

**CODE:**

data <- read.csv("input.csv")

**3. Analyze the CSV File and compute the following.**

**a. Get the maximum salary**

**CODE:**

max\_salary <- max(data$Salary)

**b. Get the details of the person with max salary**

**CODE:**

max\_salary\_person <- data[data$Salary == max\_salary, ]

**c. Get all the people working in IT department**

**CODE:**

it\_people <- data[data$Department == "IT", ]

**d. Get the persons in IT department whose salary is greater than 600**

**CODE:**

it\_people\_high\_salary <- it\_people[it\_people$Salary > 600, ]

**e. Get the people who joined on or after 2014**

**CODE:**

people\_after\_2014 <- data[as.Date(data$JoinDate, "%Y-%m-%d") >= as.Date("2014-01-01"), ]

**4. Get the people who joined on or after 2014 and write the output onto a file called output.csv**

**CODE:**

write.csv(people\_after\_2014, "output.csv")