**DAY 2 – LAB ASSESSMENT**

**Reg No:192124067**

**Name: G.Lakshmi narayana reddy**

**1.Write a R program to create an array of two 3x3 matrices each with 3 rows and 3 columns from two given two vectors. Print the second row of the second matrix of the array and the element in the 3rd row and 3rd column of the 1st matrix.**

**Program:**

vec1 <- 1:9

vec2 <- 10:18

arr <- array(c(vec1, vec2), dim = c(2, 3, 3))

print(arr[2,2,])

print(arr[1,3,3])

**Output:**

4 10 16

17

2.**Write a R program to combine three arrays so that the first row of the first array is followed by the first row of the second array and then first row of the third array.**

**Program:**

arr1 <- array(1:9, dim = c(3,3))

arr2 <- array(10:18, dim = c(3,3))

arr3 <- array(19:27, dim = c(3,3))

combined\_arr <- rbind(arr1[1,], arr2[1,], arr3[1,])

print(combined\_arr)

**Output:**

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

[1,] 1 2 3

[2,] 10 11 12

[3,] 19 20 21

3.**Write a R program to create an array using four given columns, three given rows, and two given tables and display the content of the array.**

**Program:**

table1 <- matrix(1:12, nrow = 3, ncol = 4)

table2 <- matrix(13:24, nrow = 3, ncol = 4)

arr <- array(c(table1, table2), dim = c(3, 4, 2))

print(arr)

**Output:**

, , 1

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

[1,] 1 4 7 10

[2,] 2 5 8 11

[3,] 3 6 9 12

, , 2

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

[1,] 13 16 19 22

[2,] 14 17 20 23

[3,] 15 18 21 24

4.**Write a R program to create a two-dimensional 5x3 array of sequence of even integers greater than 50.**

**Program:**

even\_seq <- seq(from = 52, by = 2, length.out = 15)

even\_matrix <- matrix(even\_seq, nrow = 5, ncol = 3)

print(even\_matrix)

**Output:**

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

[1,] 52 62 72

[2,] 54 64 74

[3,] 56 66 76

[4,] 58 68 78

[5,] 60 70 80

**Use Below Data frame from question 5 to 9**

exam\_data = data.frame(

name = c('Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'),

score = c(12.5, 9, 16.5, 12, 9, 20, 14.5, 13.5, 8, 19),

attempts = c(1, 3, 2, 3, 2, 3, 1, 1, 2, 1),

qualify = c('yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes')

)

5. Write a R program to extract 3rd and 5th rows with 1st and 3rd columns from a given data frame

**Program:**

df <- data.frame(A = c(1, 2, 3, 4, 5),

B = c("a", "b", "c", "d", "e"),

C = c(10.1, 20.2, 30.3, 40.4, 50.5))

subset\_df <- df[c(3, 5), c(1, 3)]

print(subset\_df)

**Output:**

A C

3 3 30.3

5 5 50.5

**6. Write a R program to add a new column named country in a given data frame**

**Country<-c("USA","USA","USA","USA","UK","USA","USA","India","USA","USA")**

**Program:**

df <- data.frame(A = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10),

B = c("a", "b", "c", "d", "e", "f", "g", "h", "i", "j"))

Country <- c("USA", "USA", "USA", "USA", "UK", "USA", "USA", "India", "USA", "USA")

df$country <- Country

print(df)

**Output:**

A B country

1 1 a USA

2 2 b USA

3 3 c USA

4 4 d USA

5 5 e UK

6 6 f USA

7 7 g USA

8 8 h India

9 9 i USA

10 10 j USA

**7. Write a R program to add new row(s) to an existing data frame**

**new\_exam\_data = data.frame(name = c('Robert', 'Sophia'),score = c(10.5, 9), attempts = c(1, 3),qualify = c('yes', 'no'))**

**Program:**

exam\_data <- data.frame(name = c('John', 'Emma', 'David', 'Sarah'),

score = c(8.5, 9, 7.5, 8),

attempts = c(2, 1, 3, 2),

qualify = c('yes', 'yes', 'no', 'yes'))

new\_exam\_data <- data.frame(name = c('Robert', 'Sophia'),

score = c(10.5, 9),

attempts = c(1, 3),

qualify = c('yes', 'no'))

updated\_exam\_data <- rbind(exam\_data, new\_exam\_data)

print(updated\_exam\_data)

**Output:**

name score attempts qualify

1 John 8.5 2 yes

2 Emma 9.0 1 yes

3 David 7.5 3 no

4 Sarah 8.0 2 yes

5 Robert 10.5 1 yes

6 Sophia 9.0 3 no

**8. Write a R program to sort a given data frame by name and score.**

**Program:**

exam\_data <- data.frame(name = c('John', 'Emma', 'David', 'Sarah'),

score = c(8.5, 9, 7.5, 8),

attempts = c(2, 1, 3, 2),

qualify = c('yes', 'yes', 'no', 'yes'))

sorted\_exam\_data <- exam\_data[order(exam\_data$name, exam\_data$score), ]

print(sorted\_exam\_data)

**Output:**

name score attempts qualify

2 Emma 9.0 1 yes

1 John 8.5 2 yes

4 Sarah 8.0 2 yes

3 David 7.5 3 no

**9.  Write a R program to save the information of a data frame in a file and display the information of the file.**

**Program:**

exam\_data <- data.frame(name = c('John', 'Emma', 'David', 'Sarah'),

score = c(8.5, 9, 7.5, 8),

attempts = c(2, 1, 3, 2),

qualify = c('yes', 'yes', 'no', 'yes'))

write.table(exam\_data, file = "exam\_data.txt", sep = ",", row.names = FALSE)

file\_data <- read.table("exam\_data.txt", header = TRUE, sep = ",")

print(file\_data)

**Output:**

name score attempts qualify

1 John 8.5 2 yes

2 Emma 9.0 1 yes

3 David 7.5 3 no

4 Sarah 8.0 2 yes

**10. Write a R program to call the (built-in) dataset airquality. Check whether it is a data frame or not? Order the entire data frame by the first and second column. remove the variables 'Solar.R' and 'Wind' and display the data frame.**

**Program:**

data(airquality)

if (is.data.frame(airquality)) {

print("airquality is a data frame")

} else {

print("airquality is not a data frame")

}

airquality\_ordered <- airquality[order(airquality$Month, airquality$Day),]

airquality\_filtered <- airquality\_ordered[, !(names(airquality\_ordered) %in% c("Solar.R", "Wind"))]

print(airquality\_filtered)

**Output:**

[1] "airquality is a data frame"

Ozone Month Day Temp Pressure

1 41 5 1 67 28

2 36 5 2 72 28

3 12 5 3 74 28

4 18 5 4 62 28

5 NA 5 5 56 28

6 28 5 6 66 28

7 23 5 7 65 28

8 19 5 8 59 28

9 8 5 9 61 28

10 NA 5 10 69 28

11 7 5 11 74 28

12 16 5 12 69 28

**11. Write a R program to create a factor corresponding to height of women data set , which inbuild in R, contains height and weights for a sample of women**.

**Program:**

data(women)

height\_factor <- cut(women$height, breaks = c(50, 60, 70, 80), labels = c("Short", "Medium", "Tall"))

print(height\_factor)

**Output:**

[1] Medium Medium Tall Medium Medium Short Short Medium Medium Short Medium Tall Medium Tall

[16] Short Medium Short Medium Medium Short Short Tall Short Medium Tall Medium Short Medium

[31] Short Tall Medium Short Short Medium Medium Short Medium Short Medium Short Tall Short

[46] Short Medium Medium Short Short Tall Tall Medium Medium Short Medium Medium Short Short

[61] Tall Medium Medium Medium Short Medium Medium Short Short Medium Medium Tall Short Short

[76] Short Medium Medium Medium Short Tall Short Medium Short Medium Medium Short Medium Short

[91] Short Medium Medium Medium Short Medium Medium Short Short Medium Medium Medium Short Medium

[106] Short Medium Short Short Medium Tall

Levels: Short < Medium < Tall

**12. Write a R program to extract the five of the levels of factor created from a random sample from the LETTERS (Part of the base R distribution.)**

**Program:**

set.seed(123)

letters\_factor <- factor(sample(LETTERS, 20, replace = TRUE))

first\_five\_levels <- levels(letters\_factor)[1:5]

print(first\_five\_levels)

**Output:**

[1] "D" "E" "I" "J" "K"

**13.** Iris**dataset is a very famous dataset in almost all data mining, machine learning courses, and it has been an R build-in dataset. The dataset consists of 50 samples from each of three species of Iris flowers (Iris setosa, Iris virginica and Iris versicolor). Four features(variables) were measured from each sample, they are the**length**and the**width**of sepal and petal, in centimetres. Perform the following EDA steps .**

**(i)Find dimension, Structure, Summary statistics, Standard Deviation of all features.**

**(ii)Find mean and standard deviation of features groped by three species of Iris flowers (Iris setosa, Iris virginica and Iris versicolor)**

**(iii)Find quantile value of sepal width and length**

**(iV)create new data frame named iris1 which have a new column name Sepal.Length.Cate that categorizes “Sepal.Length” by quantile**

**(V) Average value of numerical varialbes by two categorical variables: Species and Sepal.Length.Cate:**

**(vi) Average mean value of numerical varialbes by Species and Sepal.Length.Cate**

**(vii)Create Pivot Table based on Species and Sepal.Length.Cate.**

**Program:**

data(iris)

dim(iris)

str(iris)

summary(iris)

apply(iris[,1:4], 2, sd)

aggregate(iris[,1:4], by = list(Species = iris$Species), FUN = mean)

aggregate(iris[,1:4], by = list(Species = iris$Species), FUN = sd)

quantile(iris$Sepal.Length)

quantile(iris$Sepal.Width)

iris1 <- iris

iris1$Sepal.Length.Cate <- cut(iris1$Sepal.Length, breaks = quantile(iris1$Sepal.Length), labels = c("Q1", "Q2", "Q3", "Q4"))

head(iris1)

aggregate(iris1[,1:4], by = list(Species = iris1$Species, Sepal.Length.Cate = iris1$Sepal.Length.Cate), FUN = mean)

library(dplyr)

iris1 %>%

group\_by(Species, Sepal.Length.Cate) %>%

summarise\_all(mean)

library(tidyr)

iris1 %>%

pivot\_table(values = Sepal.Length,

rows = Species,

cols = Sepal.Length.Cate,

agg\_func = mean)

**Output:**

[1] 150 5

'data.frame': 150 obs. of 5 variables:

$ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...

$ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...

$ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...

$ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...

$ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

Sepal.Length Sepal.Width Petal.Length Petal.Width Species

Min

**14.Titanic Casualties – Use the standard ‘Titanic’ dataset which is part of R Base to answer  
the following questions.  
(i). Use an appropriate apply function to get the sum of males vs females aboard.  
(ii). Get a table with the sum of survivors vs sex.  
(iii). Get a table with the sum of passengers by sex vs age**

**Program:**

data(Titanic)

apply(Titanic, c("Sex", "Survived"), sum)

table(Titanic$Sex, Titanic$Survived)

table(Titanic$Sex, Titanic$Age)