**R ASSIGNMENT**

**NAME: AYUSH NANDI**

**ROLL NO: 22MCF1R13**

1. **Write R function to initialize a data frame of 10 people with columns as name and height (in cm). Use for loop to convert the height in to meter.**

# q1

initialize\_df = function(x, y)

{

df = data.frame(names=x, height=y)

}

to\_meter = function(df)

{

m = numeric(0)

for(h in df$height){ m = c(m, h/100) }

new\_df = data.frame(names=df$names, height\_in\_m=m)

}

n = scan(what='')

h = scan()

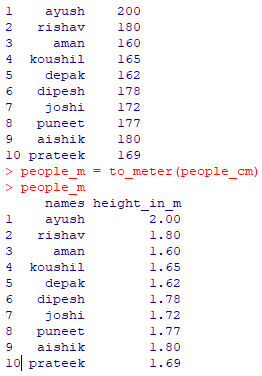
people\_cm = initialize\_df(n, h)

people\_cm

people\_m = to\_meter(people\_cm)

people\_m

**OUTPUT**

****

1. **The numbers of 10 days of rainfall amounts are given. Read them into a vector and calculate the mean and standard deviation, along with the days of highest and lowest rainfall.**

# q2

rainfall\_amt = scan()

print(paste("mean is ", mean(rainfall\_amt)))

print(paste('standard deviation is ', sd(rainfall\_amt)))

highest = max(rainfall\_amt)

lowest = min(rainfall\_amt)

for(i in 1:length(rainfall\_amt))

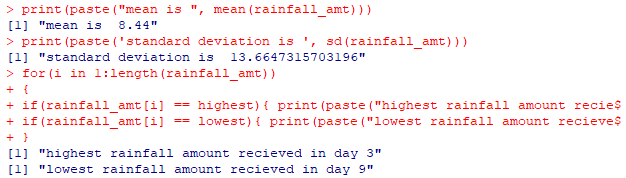
{

if(rainfall\_amt[i] == highest){ print(paste("highest rainfall amount recieved in day", (i)))}

if(rainfall\_amt[i] == lowest){ print(paste("lowest rainfall amount recieved in day", (i)))}

}

**OUTPUT**

****

1. **Consider a matrix filled with 10 rows and 10 columns all filled with random numbers between 0 and 1. Calculate row means and column means, sum of all diagonal elements and standard deviation across both**

# q3

mat = matrix(runif(100, 0, 1), 10, 10)

mat

# (i)

rmean = rowMeans(mat)

print('row means are')

print(rmean)

print('column means are')

cmean = colMeans(mat)

print(cmean)

# (ii)

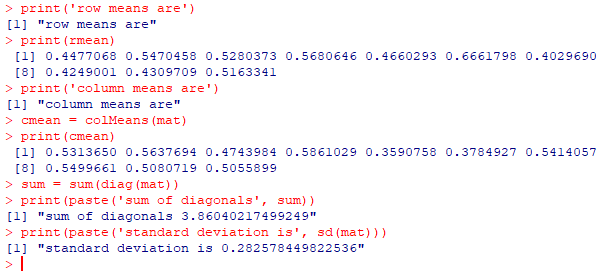
sum = sum(diag(mat))

print(paste('sum of diagonals', sum))

# (iii)

print(paste('standard deviation is', sd(mat)))

**OUTPUT**

****

1. **Consider a problem where a user needs to group a set of people P who wants to apply for a loan at a bank. The main criterion for applying is the income of the person. Given I, which corresponds to the respective incomes of people in P. Group the people into a low-risk customer based on whether the customer’s income is above 30000 (low risk) or not (High risk)**

P = scan(what='')

I = c(10000, 14000, 24000, 43000, 12323, 13414, 43212, 36000)

low\_risk = character(0)

high\_risk = character(0)

for(i in 1:length(I))

{

if(I[i] < 30000){ high\_risk = c(high\_risk, P[i])}

else { low\_risk = c(low\_risk, P[i]) }

}

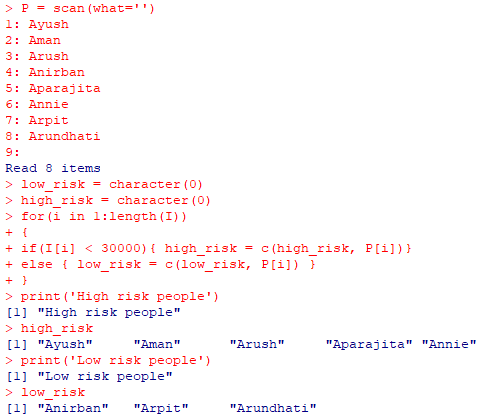
print('High risk people')

high\_risk

print('Low risk people')

low\_risk

**OUTPUT**

****

1. **Assign the values TRUE, FALSE, FALSE, TRUE to a logical vector X and the values FALSE, TRUE, FALSE, TRUE to the logical vector Y. Perform element wise AND and OR, also find logical AND and logical OR of X and Y**

# q5

X = c(TRUE, FALSE, FALSE, TRUE)

Y = c(FALSE, TRUE, FALSE, TRUE)

# (i)

print('element wise and of x and y')

X & Y

print('element wise or of x and y')

X | Y

# (ii)

a = X[0]

for(i in X) a = a && i

print(paste('Logical AND of X: ', a))

a = X[0]

for(i in X) a = a || i

print(paste('Logical OR of X: ', a))

a = Y[0]

for(i in Y) a = a && i

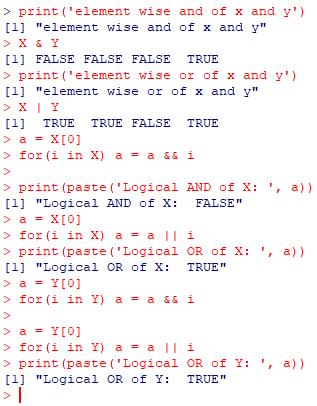
print(paste('Logical AND of Y: ', a))

a = Y[0]

for(i in Y) a = a || i

print(paste('Logical OR of Y: ', a))

**OUTPUT**

****

1. **Use the inbuilt iris and write r code for displaying the first few rows, displaying the structure of iris dataset, display the value that lies at the intersection of row 3 and column 4 and display the value that lies at the intersection of row 3 and columns 1 to 4**

# q6

data(iris)

# (a)

head(iris, 5)

# (b)

dim(iris)

summary(iris)

names(iris)

# (c)

x = iris

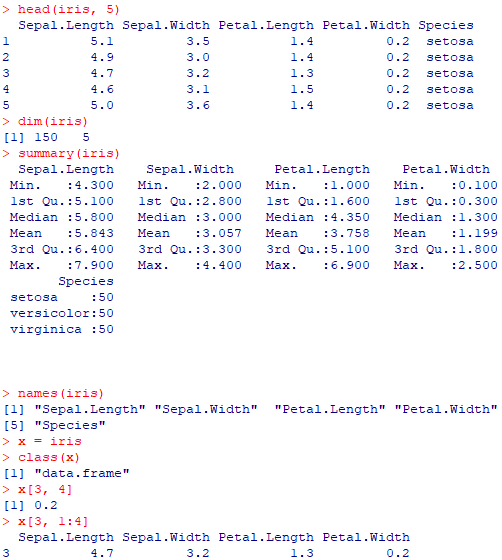
class(x)

x[3, 4]

# (d)

x[3, 1:4]

**OUTPUT**

****

1. **Given the name of chemical (c1, c2, …, c5) and amount produced, plot a pie chart with the initial angle 180 degree**

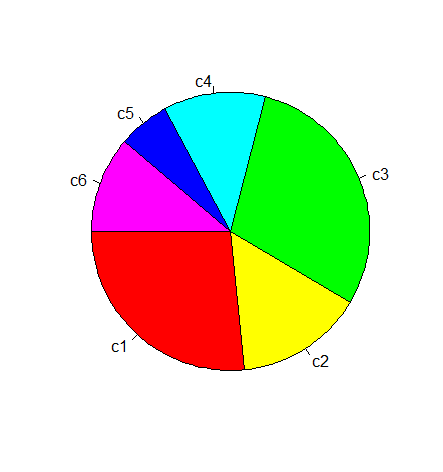
**# q7**

amt = c(90, 50, 100, 40, 20, 38)

chem = c('c1', 'c2', 'c3', 'c4', 'c5', 'c6')

pie(amt, label=chem, col=rainbow(length(amt)), init.angle=180)

**OUTPUT**

****

1. **Write R code using a loop to print the required output**

# q8

x = 1

while(x <= 7)

{

if(x <= 3)

{

print('Four is greater than 3')

}

else if(x == 4)

{

print('Next')

}

else

{

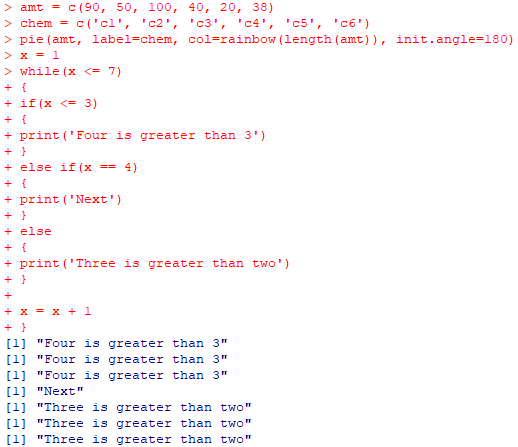
print('Three is greater than two')

}

x = x + 1

}

**OUTPUT**



1. **Write a program in R to print the harmonic series and its sum**

# q9

n = scan()

s = 0.0

for(i in 1:n)

{

if(i < n)

{

cat('1/', i, ' + ')

s = s + 1/i

}

if(i == n)

{

cat('1/', i)

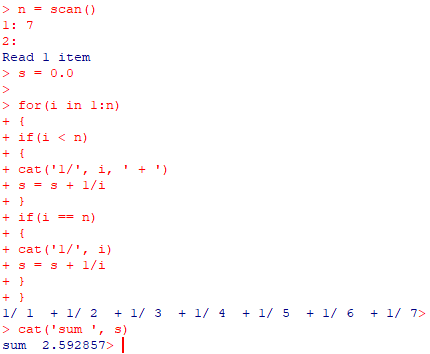
s = s + 1/i

}

}

cat('sum ', s)

**OUTPUT**

****

1. **Write a nested loop where the outer for() loop increments ‘a’ 3 times and ‘b’ 4 times. The break statement exits the loop after 2 increments. The nested loop prints the value of elements a and b**

# q10

a = 0

b = 0

for(i in 1:3)

{

a = a + 1

for(j in 1:4)

{

b = b + 1

if(j == 2)

{

break

}

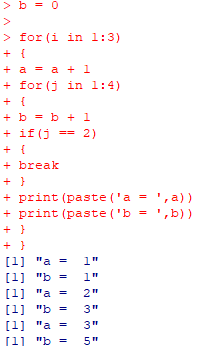
print(paste('a = ',a))

print(paste('b = ',b))

}

}

**OUTPUT**

****

1. **Write a R program to create a student data frame, replace a value across the entire data frame and replace multiple values across the dataframe**

# q11

sid = c(1,2, 3, 4, 5, 6)

age = c(22, 23, 22, 21, 21, 24)

pointer = c(8, 9, 8, 7, 6, 8)

dept = c('mca', 'mca', 'mca', 'mca', 'mca', 'mca')

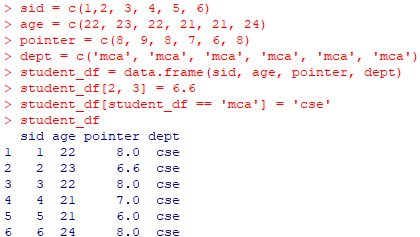
student\_df = data.frame(sid, age, pointer, dept)

student\_df[2, 3] = 6.6

student\_df[student\_df == 'mca'] = 'cse'

student\_df

**OUTPUT**

****

1. **Use inbuilt cars data frame to create a box plot as well as histogram**

# q12

data(cars)

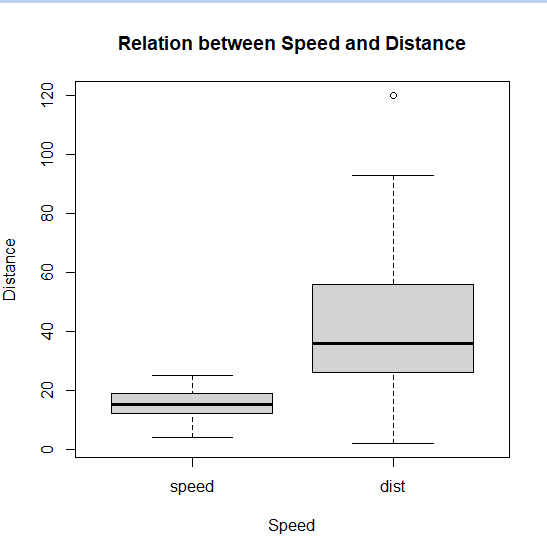
head(cars, 10)

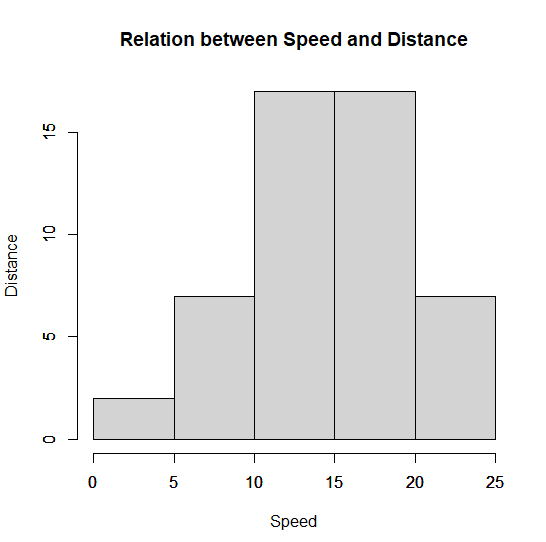
a = cars

boxplot(a, main='Relation between Speed and Distance', xlab='Speed', ylab='Distance')

hist(a$speed, main='Relation between Speed and Distance', xlab='Speed', ylab='Distance')

**OUTPUT**

****

****