P1: DataFrame

```
Student_ID <- c()

for(i in 1001:1010) {

+ Student_ID <- append(Student_ID, i)

+ }

Student_ID



Name <- c("jim", "jacy", "ben", "lexi", "john", "suzan", "lee", "emma", "drax", "alice")

Name



Marks <- c(23,34,54,35,65,34,76,45,87,88)

Marks



 Gender <- c("male", "female", "male", "female", "male", "female", "male", "female", "male", "female")

Gender



stringAsFactors = FALSE



df <- data.frame(Student_ID, Name, Marks, Gender)

df


str(df)
summary(df)

extra <- c(1011, "jazz", 44, "male")
df[nrow(df) + 1,] <- extra
df

data.frame(df$Student_ID, df$Marks)
```


P2: Import Export

```
install.packages("writexl")
library("writexl")


Student_ID <- c()

for(i in 1001:1010) {

+ Student_ID <- append(Student_ID, i)

+ }

Name <- c("jim", "jacy", "ben", "lexi", "john", "suzan", "lee", "emma", "drax", "alice")

Marks <- c(23,34,54,35,65,34,76,45,87,88)

Gender <- c("male", "female", "male", "female", "male", "female", "male", "female", "male", "female")

stringAsFactors = FALSE

df <- data.frame(Student_ID, Name, Marks, Gender)



write.csv(df, file = "D:/University/Sem 4/MFDA/PR/pr2_try1.csv")

write.table(df, file = "D:/University/Sem 4/MFDA/PR/pr2_try2.txt")



df_read1 <- read.csv("D:/University/Sem 4/MFDA/PR/pr2_try1.csv")

df_read2 <- read.table(file = "D:/University/Sem 4/MFDA/PR/pr2_try2.txt")
```


P3: Central Tendencies

```
mean(mtcars$cyl)
mean(mtcars$mpg)
mean(mtcars$gear)
mean(mtcars$disp)

mode<-function(x){which.max(tabulate(x))}
mode(mtcars$cyl)

median(mtcars$cyl)

var(mtcars$cyl)

sd(mtcars$cyl)
```

```
boxplot(mpg ~ cyl, data = mtcars, xlab = "Number of Cylinders", ylab = "Miles Per Gallon", main =
"Mileage Data")

png(file = "C:/Users/HP/Documents/MFDATrash/line_chart1.jpg")
plot(mtcars$cyl, type = "o", xlab = "Index", ylab = "Cylinders", main = "Cylinders in mtcars DataSet")
dev.off()

png(file = "C:/Users/HP/Documents/MFDATrash/dotplot1.jpg")
dotchart(mtcars$cyl, labels = rownames(mtcars))
dev.off()

hist(mtcars$mpg, xlab = "Miles Per Gallon", ylab = "Number of Cars", main = "Cars Distribution")
hist(mtcars$cyl, xlab = "Cylinders", main = "Histogram for Cylinder in mtcars")

barplot(mtcars$cyl, xlab = "Cars", ylab = "Cylinders", main = "Barplot: Cars and Number of cyl")

barplot (table(mtcars$cyl), main = "Car Distribution", xlab = "Number of Cylinders", col = c("darkblue",
"green", "red"), names.arg = c("4 Cylinder", "6 Cylinder", "8 Cylinder"))

pie(table (mtcars$cyl), labels = c("4 Cylinder", "6 Cylinder", "8 Cylinder"), main="Car Distribution")

plot(x = input$cyl, y = input$mpg, xlab = "Number of Cylinders", ylab = "Miles Per Gallon", main =
"Scatterplot: Cylinders vs MPG")
```


P4: Advanced Visual

```
sapply(mtcars, function(mtcars) max(mtcars, na.rm = TRUE) - min(mtcars, na.rm = TRUE))

order(mtcars$mpg)
sort(mtcars$mpg)
IQR(mtcars$mpg)

quantile(mtcars$wt, probs = c(0,0.25,0.5,0.75,1))

quantile(mtcars$wt, probs = c(.75, .8))

boxplot(mpg ~ cyl, data = mtcars, xlab = "Number of Cylinders", ylab = "Miles Per Gallon", main =
"Mileage Data")
```

a) 8 cylinders are needed for lowest milage per gallon
b) Prediction with 6 number of cylinders gives maximum confidence

----

P5: Prob Dis

```
x = 0:5
```

```
barplot(height = dbinom(x,5,0.4), names.arg = c(0,1,2,3,4,5), xlab = "Number of Late Arrivals", ylab =
"Probability", main = "Problem 1")

x = 0:5
barplot(dbinom(x,5,0.1))
barplot(dbinom(x,5,0.3))
```

When prob is lower, i.e. 0.1: the data is right skewed. When prob increases, the data becomes left skewed. At 0.9, the data is left skewed.ype = "o") / barplot(dpois(x, 3))

```
x = 1:20
barplot(dbinom(x,20,0.4))
```

Keeping p = 0.4, and increasing n, the data follows normal distribution.

```
x <- c(0:4)
dpois(x, 5)


ppois(7, 5, lower.tail = TRUE)
ppois(7, 5, lower.tail = FALSE)
ppois(8,5)-ppois(4,5)


x <- c(1:50)
plot(x, dpois(x, 3), type = "o") / barplot(dpois(x, 3))
```


P6: Case Study of Prob Dis

```
x <- seq(0, 50, by = 1)
y <- dnorm(x, mean = 25.0, sd = 10)
plot(x,y, main = "Normal Distribution", col = "blue")

pnorm(27.5, mean = 22, sd = 29) - pnorm(16.2, mean = 22, sd = 29)

cat("The probability is less than 17: ", pnorm(17, mean = 22, sd = 29))

pnorm(17, mean = 22, sd = 29)

pnorm(15, mean = 22, sd = 29) + (1 - pnorm(25, mean = 22, sd = 29))

pnorm(850000, mean = 1000000, sd = 200000)
```


P7: Algebra Op and Dist Met

```
install.packages("pracma")
```

```
a <- c(1,2,3)
b <- c(5,6,7)
dot(a,b)
cross(a,b)

d = matrix(c(1,2,3), nrow=3, ncol=3, byrow=TRUE)
e = matrix(c(4,5,6), nrow=3, ncol=3, byrow=TRUE)
cross(d, e)

f = matrix(c(1,2,3), nrow=1, ncol=3, byrow=TRUE)
f
t(f)

install.packages("geometry")
install.packages("philentropy")
library(geometry)
library(philentropy)

  x <- c(50,5,2)
  y <- c(1,8,9)
  p <- rbind(x,y)
  distance(p, method = "euclidean")
  distance(p, method = "manhattan")
  distance(p, method = "jaccard")
  print('Hamming Distance')
  sum(x != y)

  d = matrix(c(0.68567, 0.12975, -0.71626, 0.14807, 0.93855, 0.31176, 0.71269, -0.31982, 0.62433),
  nrow=3, ncol=3, byrow=TRUE)
  det(d)

  install.packages("matlib")
```

```
library(matlib)

e = matrix(c(13, -4, 2, -4, 11, -2, 2, -2, 8), nrow=3, ncol=3, byrow=TRUE)
ev <- eigen(e)
(values <- ev$values)
(vectors <- ev$vectors)

sum(e^2)
sum(values^2)

det(e)
prod(values)

R(e)
sum(values != 0)
```

P8: Sampling Dis Techniques

```
#define number of samples
n = 10000

#create empty vector of length n
sample_means = rep(NA, n)

#fill empty vector with means
for(i in 1:n){
  sample_means[i] = mean(rnorm(20, mean=5.3, sd=9))
}

#view first six sample means
head(sample_means) [o/p: [1] 4.304143 6.058354 7.803126 4.667816 4.379790 8.193655]
```

```
#create histogram to visualize the sampling distribution
hist(sample_means, main = "", xlab = "Sample Means", col = "steelblue")

#mean of sampling distribution
mean(sample_means)

#standard deviation of sampling distribution
sd(sample_means)

#calculate probability that sample mean is less than or equal to 6
sum(sample_means <= 6) / length(sample_means)
```


P9: Estimated Value

```
install.packages(“MASS”)
library(MASS)
height.survey = survey$Height
mean(height.survey, na.rn = TRUE)
height.response = na.omit(survey$Height)
n = length(height.response)
s = sd(height.response)
SE = s/sqrt(n)
SE
E = qt(.975, df = n-1)*SE
xbar = mean(height.response)
xbar + c(-E, E)
```


P10: Hypothesis

```
xbar = 9900

mu0 = 10000

sigma = 120

n = 30

z = (xbar - mu0) / (sigma/sqrt(n))

z

alpha = 0.05

z.alpha = qnorm(1-alpha)

-z.alpha (reject)


xbar = 2.1

mu0 = 2

sigma = 0.25

n = 35

z = (xbar - mu0) / (sigma/sqrt(n))

z

alpha = .05

z.alpha = qnorm(1-alpha)

z.alpha (reject)
```