SET - 3

1 .

```
fractions <- vector("character", 11)
# Create a vector to store the fractions
j <- 1

for (i in seq(1, 21, 2)) {
  fractions[j] <- paste("1/", i, sep = "")
  j <- j + 1
}

Fractions
```

2.

```
n <- 10
p <- 0.10
k <- 2

probability_binomial <- dbinom(k, n, p)
probability_binomial

lambda <- n * p

probability_poisson <- dpois(k, lambda)
probability_poisson
```

3.

```
# Load the mtcars dataset
data(mtcars)

# Fit the Poisson regression model
model <- glm(mpg ~ ., data = mtcars, family = poisson)

# Print the model summary
summary(model)
```

4.

```
# Install and load required packages
install.packages("plotrix")
library(plotrix)

# Data
x <- c(21, 62, 10, 53)
labels <- c("London", "New York", "Singapore", "Mumbai")

# Set random rainbow colors
colors <- rainbow(length(x))
# Create pie chart
pie(x, labels = labels, col = colors, main = "City Pie Chart")
```

SET - 4

1.
```
# Data
ages <- c(18, 19, 19, 19, 19, 20, 20, 20, 20, 20, 21, 21, 21, 21, 22, 23, 24, 27, 30, 36)

# (a) Median and Mean of all students
median_all <- median(ages)
mean_all <- mean(ages)

# (b) Median age of students under 25
median_under_25 <- median(ages[ages < 25])

# (c) Modal age of all students
modal_age <- names(table(ages))[table(ages) == max(table(ages))]

# Print the results
cat("Median of all students:", median_all, "\n")
cat("Mean of all students:", mean_all, "\n")
cat("Median age of students under 25:", median_under_25, "\n")
cat("Modal age of all students:", modal_age, "\n")
```

2.
```
# Create a 2x3 matrix with random values between 10 and 30
M <- matrix(sample(10:30, 6, replace = TRUE), nrow = 2, ncol = 3, byrow = TRUE)

# Create a vector with three values
V <- c(15, 20, 25)

# Add the vector as the third row of the matrix
M <- rbind(M, V)

# Print the matrix
M
```

3.
```
# Heights of father and son
father <- c(152, 155, 157, 160, 161, 164, 165, 150)
son <- c(156, 158, 159, 160, 162, 161, 164, 154)

# Create a data frame
data <- data.frame(father, son)

# Fit a linear regression model
model <- lm(son ~ father, data = data)

# Get the coefficients of the regression line
intercept <- coef(model)[1]
slope <- coef(model)[2]
```

```
# Print the coefficients
cat("Intercept:", intercept, "\n")
cat("Slope:", slope, "\n")

# Predict the son's height for a given father's height
father_height <- 158
son_height <- predict(model, newdata = data.frame(father = father_height))
cat("Predicted son's height for father's height", father_height, "is", son_height)
```

4.

```
# Install and load required packages
install.packages("plotrix")
library(plotrix)

# Data
x <- c(21, 62, 10, 53)
labels <- c("London", "New York", "Singapore", "Mumbai")

# Set random rainbow colors
colors <- rainbow(length(x))
# Create pie chart
pie(x, labels = labels, col = colors, main = "City Pie Chart")
```

SET - 5

1. a)

```
# Create the data frame
df <- data.frame(
  Item = c("Baby food", "Cereal", "Office supplies", "Fruits", "Office supplies",
"Household", "Household"),
  OrderPriority = c(1, 2, 3, 1, NA, 3, 3),
  UnitPrice = c(255.28, 205.7, NA, 9.33, 651.21, 668.27, 668.27),
  UnitsSold = c(9925, 2804, 1779, 8102, 5062, 8974, NA),
  stringsAsFactors = FALSE
)

# Print the data frame
df
```

b)

```
mean_order_priority <- mean(df$OrderPriority, na.rm = TRUE)
df$OrderPriority <- replace(df$OrderPriority, is.na(df$OrderPriority),
mean_order_priority)
```

c)

```
median_units_sold <- median(df$UnitsSold, na.rm = TRUE)
df$UnitsSold <- replace(df$UnitsSold, is.na(df$UnitsSold), median_units_sold)
```

d)

```
df <- na.omit(df)
```

e)

```
df$UnitPrice <- unique(df$UnitPrice)
```

f)

```
total_missing <- sum(is.na(df))
cat("Total number of missing values in df:", total_missing, "\n")
```

2.

```
# Load the mtcars dataset
data(mtcars)

# Fit the Poisson regression model
```

```
model <- glm(mpg ~ ., data = mtcars, family = poisson)

# Print the model summary
summary(model)
```

3.

```
# Character vector with white spaces
text <- c("Hello ", "   World", "    Open  AI   ")

# Remove white spaces
text <- gsub("\\s", "", text)

# Print the modified character vector
print(text)
```