1.Try to write a code for printing sequence of numbers from 1 to 50 with the differences of 3, 5, 10.

Ans: print (seq (1,50))

2. What are the different data objects in R? and write syntax and example for each and every object

Ans: R consists of a number of data objects to perform various functions. There are 6 types of objects in R Programming. They include vector, list, matrix, array, factor, and data frame. Vectors are one of the basic R programming data objects.

Syntaxes-

* **Vector**

Vector – c(elements)

* **List**

List – list(data)

* **Matrix**

M – matrix(data, nrow, ncol, byrow, dimnames)

* **Array**

A – array(data, dim, dimnames)

* **Factor**

Factor – factor(data)

* **Data Frame**

Df – data.frame(Key = value,

Key = value)

3. Create Data frame with 3 columns and 5 rows and write a code to fetch and delete row and a column using index and add new column and row to the existed data frame

emp\_id = c(1:5)

emp\_name = c('Rick','Dan','Michelle','Ryan','Gary')

salary = c(623.3,515.2,611.0,729.0,843.25)

emp.data <- data.frame(emp\_id,emp\_name, salary)

emp.data = emp.data[-c(3)] **#Delete Column**

emp.data = emp.data[-c(3),] **#Delete Row**

emp.data

**#Add New Row**

DataFrame.col <- data.frame(emp\_id, emp\_name, salary)

ep.data <- data.frame(6, "Jimmy", 458.59)

names(ep.data) <- c("emp\_id", "emp\_name", "salary")

new.data <- rbind(DataFrame.col, ep.data)

new.data

**#Add New Column**

new.data['designation'] = c("data scientist", "senior manager", "HR", "Admin", "Sysmter Engineer", "Executive")

new.data

4.Write nested if else statements to print whether the given number is negative, positive or Zero

Num = 45

If (num > 0) {

print('Positive')

} else if (num < 0) {

print('Negative')

} else {

print('Zero')

}

5.write a program to input any value and check whether it is character, numeric or special character

x = "Hello"

if (is.character(x)){

print('Character')

} else if (is.numeric(x)){

print('Numeric')

} else {

print('Special')

}

6.write difference between break and next also write examples for both

Ans: The next statement in R is useful when we need to skip the current iteration of a loop without eliminating it. Like with while and repeat loops, you can break the for loop completely by using a break statement.

Examples

**Break:**

no <- 1:10

for (val in no)

{

    if (val == 5)

    {

        Print (paste ("Coming out from for loop Where i = ", val))

        break

    }

    Print (paste ("Values are: ", val))

}

**Next:**

no <- 1:10

  for (val in no)

{

    if (val == 6)

    {

        Print (paste ("Skipping for loop Where i = ", val))

        next

    }

    Print (paste ("Values are:  ", val))

}

7.write a program to print a given vector in reverse format

x= c(1,5.6,3,10,3.5,5)

v = c (1,5.6,3,10,3.5,5)

rv = rev(v)

print (rv)

8.write a program to get the mode value of the given vector (‘a’,’b’,’c’,’t’,’a’,’c’,’r’,’a’,’c’,’t’,’z’,’r’,’v’,’t’,’u’,’e’,’t’)

vet = c("a","b", "c","t","a", "c","r","a","c", "t","z","r","v","t","u","e","t")

my\_mode <- function(x) { # Create mode function

unique\_x <- unique(x)

tabulate\_x <- tabulate(match(x, unique\_x))

unique\_x[tabulate\_x == max(tabulate\_x)]

}

my\_mode(vet)

9.Write a function to filter only data belongs to ‘setosa’ in species of Iris dataset.( using dplyr package)

iris <- datasets::iris

library(dplyr)

filter(iris,Species=="virginica")

10.Create a new column for iris dataset with the name of Means\_of\_obs, which contains mean value of each row.( using dplyr package)

iris <- datasets::iris

library(dplyr)

spc <- iris['Species']

spc

iris= iris[-c(5)]

iris['Means\_of\_obs'] = apply(iris, 1, mean)

iris['Species'] = spc

iris

11.Filter data for the “versicolor” and get only ‘sepel\_length’ and Sepel \_width’ columns.( using dplyr package)

iris <- datasets::iris

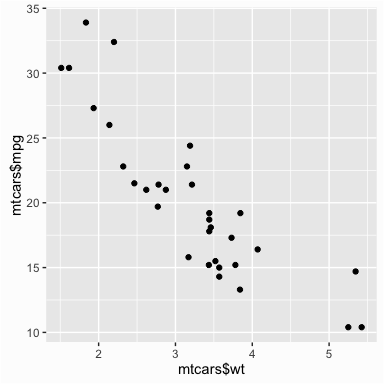
library(dplyr)

data = filter(iris,Species=="versicolor")

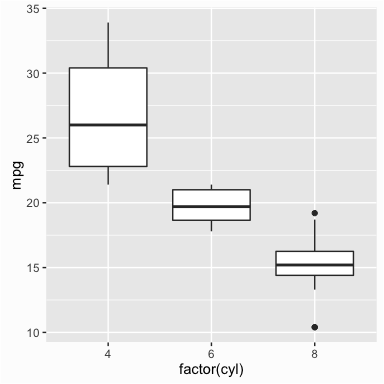
data [c('Sepal.Length', 'Sepal.Width')]

12.create below plots for the mtcars also write your inferences for each and every plot (use ggplot package) Use Different ( Size , Colour )

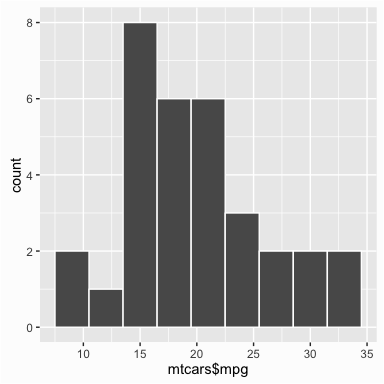
* scatter plot



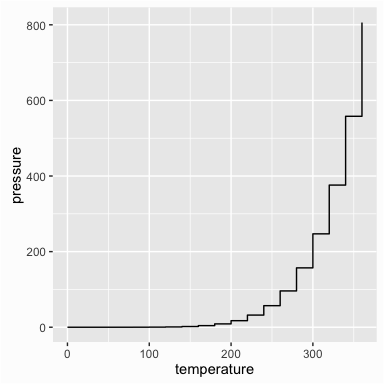
* boxplot



* histogram



* line graph



* bar graph

