```
Data formats
# DATA TYPES #############################################
# Numeric
n1 <- 15  # Double precision by default
n1
typeof(n1)
n2 <- 1.5
n2
typeof(n2)
# Character
c1 <- "c"
c1
typeof(c1)
c2 <- "a string of text"
c2
typeof(c2)
# Logical
l1 <- TRUE
l1
typeof(l1)
l2 <- F
l2
typeof(l2)


###decision statements
# if...else statement

x <- c("what","is","truth")

if("Truth" %in% x) {
   print("Truth is found")
} else {
   print("Truth is not found")
}

# if...elseif statement
x <- c("what","is","truth")

if("Truth" %in% x) {
   print("Truth is found the first time")
} else if ("truth" %in% x) {
   print("truth is found the second time")
} else {
   print("No truth found")
}
#switch
x <- switch(
   3,
   "first",
   "second",
   "third",
   "fourth"
)

print(x)

##looping statements
```

```r
#for..loop
v <- LETTERS[1:4]
for ( i in v) {
   print(i)
}

#while..loop
v <- c("Hello","while loop")
cnt <- 2

while (cnt < 7) {
   print(v)
   cnt = cnt + 1
}

#repeat statement
v <- c("Hello","loop")
cnt <- 2

repeat {
   print(v)
   cnt <- cnt+1

   if(cnt > 5) {
      break
   }
}




######defining a function
# Create a function to print squares of numbers in sequence.
new.function <- function(a) {
   for(i in 1:a) {
      b <- i^2
      print(b)
   }
}

# Call the function new.function supplying 6 as an argument.
new.function(6)




# DATA STRUCTURES #####################################
## Vector ###########################################
v1 <- c(1, 2, 3, 4, 5)
v1
is.vector(v1)
v2 <- c("a", "b", "c")
v2
is.vector(v2)
v3 <- c(TRUE, TRUE, FALSE, FALSE, TRUE)
v3
is.vector(v3)
## Matrix ###########################################
```

```
m1 <- matrix(c(T, T, F, F, T, F), nrow = 2)
m1
m2 <- matrix(c("a", "b",
               "c", "d"),
nrow = 2,
byrow = T)
m2
## Array ############################################
# Give data, then dimemensions (rows, columns, tables)
a1 <- array(c( 1:24), c(4, 3, 2))
a1
## Data frame #######################################
# Can combine vectors of the same length
vNumeric<- c(1, 2, 3)
vCharacter<- c("a", "b", "c")
vLogical<- c(T, F, T)
dfa<- cbind(vNumeric, vCharacter, vLogical)
dfa  # Matrix of one data type
df<- as.data.frame(cbind(vNumeric, vCharacter, vLogical))
df  # Makes a data frame with three different data types
## List ############################################
o1 <- c(1, 2, 3)
o2 <- c("a", "b", "c", "d")
o3 <- c(T, F, T, T, F)
list1 <- list(o1, o2, o3)
list1
list2 <- list(o1, o2, o3, list1)  # Lists within lists!
list2

# CLEAN UP ##########################################
# Clear environment
rm(list = ls())
# Clear console
ctrl+L
```

Types of Operators

We have the following types of operators in R programming –

•   Arithmetic Operators

•   Relational Operators

•   Logical Operators

•   Assignment Operators

•   Miscellaneous Operators

```
#Colon
v <- 2:8
print(v)
#in
v1 <- 8
v2 <- 12
t <- 1:10
```

```
print(v1 %in% t)
print(v2 %in% t)
#Matrix multiplication
M = matrix( c(2,6,5,1,10,4), nrow = 2,ncol = 3,byrow = TRUE)
t = M %*% t(M)
print(t)
```

Exercise

Write a R program to create three vectors numeric data, character data and logical data. Display the content of the vectors and their type.

Write a R program to create a 4 x 5 matrix, 3 x 2 matrix with labels and fill the matrix by rows and 2 × 2 matrix with labels and fill the matrix by columns.

Write a R program to compute sum, mean and product of a given vector elements.

List all the observations of "iris" dataset.

Write a R program to compute addition and subtraction of two matrices of dimension n x (n+1).

Write a R program to create a list containing a vector, a matrix and a list; and give names to the elements in the list. Access the second element of the list.