

## Mid-1 Exercises:

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

Source Code:

```
a = c(1, 2, 5, 3, 4, 0, -1, -3)
b = c("Red", "Green", "White")
c = c(TRUE, TRUE, TRUE, FALSE, TRUE, FALSE)
print(a)
print(typeof(a))
print(b)
print(typeof(b))
print(c)
print(typeof(c))
```

2. Write a R program to get the first 10 Fibonacci numbers.

Source Code:

```
Fibonacci <- numeric(10)
Fibonacci[1] <- Fibonacci[2] <- 1
for (i in 3:10) Fibonacci[i] <- Fibonacci[i - 2] + Fibonacci[i - 1]
print("First 10 Fibonacci numbers:")
print(Fibonacci)
```

3. Write a R program to find the factors of a given number.

Source Code:

```
print_factors = function(n) {
  print(paste("The factors of", n, "are:"))
  for(i in 1:n) {
    if((n %% i) == 0) {
      print(i)
    }
  }
}
print_factors(4)
print_factors(7)
print_factors(12)
```

4. Write a R program to create a list of random numbers in normal distribution and count occurrences of each value.

Source Code:

```
n = floor(rnorm(1000, 50, 100))
print('List of random numbers in normal distribution:')
print(n)
t = table(n)
print("Count occurrences of each value:")
print(t)
```

5. Write a R program to read the .csv file and display the content.

Source Code:

```
source_data = read.csv(file="_____", header=TRUE, sep=",")
print("Content of the .csv file:")
print(source_data)
```

6. Write a R program to get all prime numbers up to a given number

Source Code:

```
prime_numbers <- function(n) {
if (n >= 2) {
  x = seq(2, n)
  prime_nums = c()
  for (i in seq(2, n)) {
    if (any(x == i)) {
      prime_nums = c(prime_nums, i)
      x = c(x[(x %% i) != 0], i)
    }
  }
  return(prime_nums)
}
else
{
  stop("Input number should be at least 2.")
}
}
prime_numbers(12)
```

7. Write a R program to create an array with three columns, three rows, and two "tables", taking two vectors as input to the array. Print the array.

Source Code:

```
v1 = c(1, 3, 5, 7)
v2 = c(2, 4, 6, 8, 10)
arra1 = array(c(v1, v2), dim = c(3,3,2))
print(arra1)
```

8. Write a R program to create a list of elements using vectors, matrices and a functions. Print the content of the list.

Source code:

```
l = list(  
  c(1, 2, 2, 5, 7, 12),  
  month.abb,  
  matrix(c(3, -8, 1, -3), nrow = 2),  
  asin  
)  
print("Content of the list:")  
print(l)
```

9. Write a R program to create a simple bar plot of five subjects marks.

Source Code:

```
marks = c(70, 95, 80, 74)  
barplot(marks,  
  main = "Comparing marks of 5 subjects",  
  xlab = "Marks",  
  ylab = "Subject",  
  names.arg = c("English", "Science", "Math.", "Hist."),  
  col = "darkred",  
  horiz = FALSE)
```

10. Write a R program to create bell curve of a random normal distribution.

Source Code:

```
n = floor(rnorm(10000, 500, 100))  
t = table(n)  
barplot(t)
```

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

Source Code:

```
nums = c(10, 20, 30)  
print('Original vector:')  
print(nums)  
print(paste("Sum of vector elements:", sum(nums)))  
print(paste("Mean of vector elements:", mean(nums)))  
print(paste("Product of vector elements:", prod(nums)))
```

12. Write a R program to create a Dataframes which contain details of 5 employees and display summary of the data.

Source Code:

```
Employees = data.frame(Name=c("Anastasia S","Dima R","Katherine S",  
"JAMES A","LAURA MARTIN"),  
                        Gender=c("M","M","F","F","M"),  
                        Age=c(23,22,25,26,32),  
  
                        Designation=c("Clerk","Manager","Exective","CEO","ASSISTANT"),  
                        SSN=c("123-34-2346","123-44-779","556-24-  
433","123-98-987","679-77-576")  
                        )  
print("Summary of the data:")  
print(summary(Employees))
```