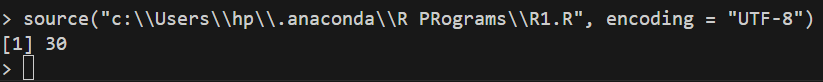
1. WAP to display any value in vector from location.

my\_vector <- c(10, 20, 30, 40, 50)

index <- 3

value\_at\_index <- my\_vector[index]

print(value\_at\_index)



1. WAP to add two numbers using functions.

twoNumAdder <- function(num1, num2){

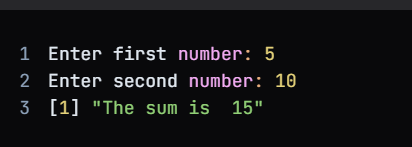
  return(num1 + num2)

}

number1 <- as.integer(readline(prompt = "Enter first number: "))

number2 <- as.integer(readline(prompt = "Enter second number: "))

paste("The sum is ",twoNumAdder(number1, number2))



1. WAP to check if a number is even or odd by using function.

is\_even <- function(num) {

  if (num %% 2 == 0) {

    return("Even")

  } else {

    return("Odd")

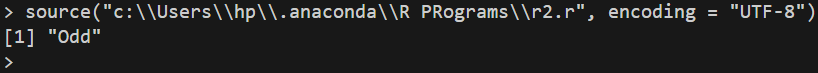
  }

}

number <- 25

result <- is\_even(number)

print(result)



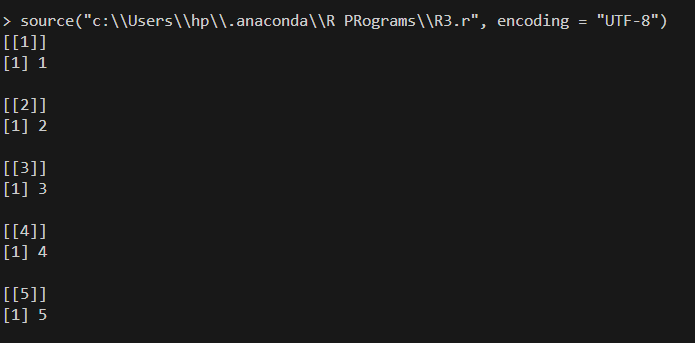
1. Create a list, add an element, and remove the last element.

my\_list <- list(1, 2, 3, 4, 5)

my\_list <- append(my\_list, 6)

my\_list <- my\_list[-length(my\_list)]

print(my\_list)



1. WAP to create data frame for 10 students containing rollno, name & marks of 3 subjects - sub1, sub2 & sub3.

df <- data.frame(

  Roll\_No = 1:10,

  Name = c("Agrim", "Bobby", "Prabjeet", "Mansi", "Himanshu", "Atul", "Etika", "Mona", "Raj", "Kapil"),

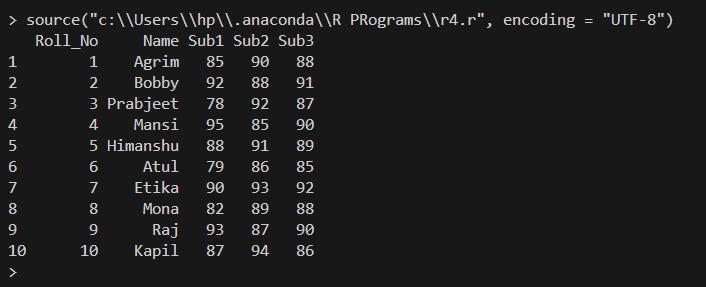
  Sub1 = c(85, 92, 78, 95, 88, 79, 90, 82, 93, 87),

  Sub2 = c(90, 88, 92, 85, 91, 86, 93, 89, 87, 94),

  Sub3 = c(88, 91, 87, 90, 89, 85, 92, 88, 90, 86)

)

print(df)



1. Generate dataset with 10 rows and 4 columns(ID, Name, Age, Score). Include atleast 5 NA values distributed across different columns. Print the original dataset.

df <- data.frame(

ID = 1:10,

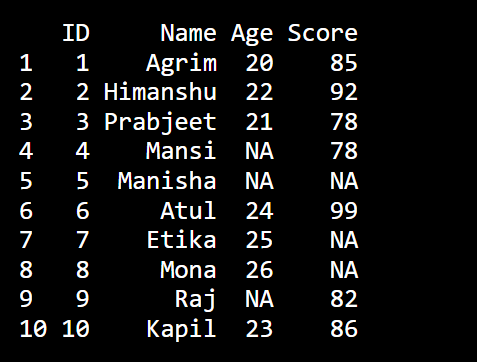
Name = c("Agrim", "Himanshu", "Prabjeet", "Mansi", "Manisha", "Atul", "Etika", "Mona", "Raj", "Kapil"),

Age = c(20, 22, 21, NA, NA, 24, 25, 26, NA, 23),

Score = c(85, 92, 78, 78, NA, 99, NA, NA, 82, 86)

)

print(df)

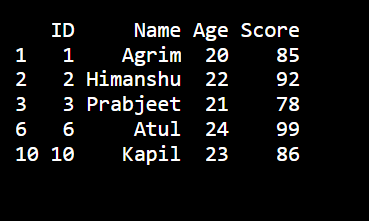


Write Code to:

1. Remove all rows that contain NA values from the dataset and print the cleaned dataset.

df\_cleaned <- na.omit(df)

print(df\_cleaned)



1. Replace NA values in numeric columns (Age & Score) with median of the respective column and print the dataset

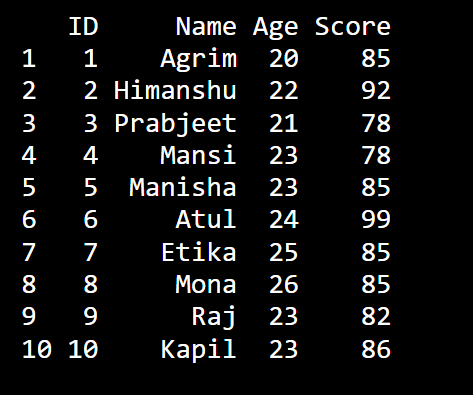
median\_age <- median(df$Age, na.rm = TRUE)

median\_score <- median(df$Score, na.rm = TRUE)

df$Age[is.na(df$Age)] <- median\_age

df$Score[is.na(df$Score)] <- median\_score

print(df)

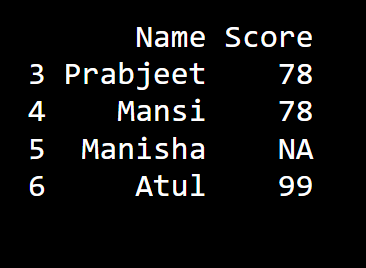


1. Perform slice and filter operation on dataset of program

# Slice Operation

sliced\_data <- df[3:6, c("Name", "Score")]

print(sliced\_data)



# Filter Operation

filtered\_data <- df[df$Score >= 85, ]

print(filtered\_data) 