1. Will the following code return any error? State the reason behind your answer and explain the logic behind the code

```
val &lt;- numeric()
result &lt;- vector(&quot;list&quot;, length(val))
for (index in 1:length(val)) {
result[index] &lt;- val[index] ^ 2
}
```

2. . What is the value of equation1(3) for the following R code and explain the logic.

```
&gt; num &lt;- 4
&gt; equation1 &lt;- function (val)
+ {
+ num &lt;- 3
+ num^3 + g (val)
+ }
&gt; equation2 &lt;- function (val)
+ {
+ val*num
+ }
}
```

3. Write R function to find nth highest value of a vector in the R program

```
find_nth_highest <- function(x, n) {
  sorted_x <- sort(x, decreasing = TRUE)
if (n > length(sorted_x)) {
    return(NA)
  } else {
    return(sorted_x[n])
```

```
 }
}
x <- c(3, 1, 4, 1, 5, 9, 2, 6, 5, 3)
find_nth_highest(x, 3)
```

4. Explore the airquality dataset. It contains daily air quality measurements from New York during a period of five months:

• Ozone: mean ozone concentration (ppb),

• Solar.R: solar radiation (Langley),

• Wind: average wind speed (mph),

• Temp: maximum daily temperature in degrees Fahrenheit,

• Month: numeric month (May=5, June=6, and so on),

• Day: numeric day of the month (1-31).

i. Compute the mean temperature(don't use build in function)

ii. Extract the first five rows from airquality.

iii. Extract all columns from airquality except Temp and Wind

iv. Which was the coldest day during the period?

v. How many days was the wind speed greater than 17 mph?

```
data(airquality)

mean_temp <- sum(airquality$Temp) / length(airquality$Temp)
print(mean_temp)

first_five_rows <- airquality[1:5, ]
print(first_five_rows)
all_cols_except_temp_wind <- airquality[, !(names(airquality) %in% c("Temp", "Wind"))]
print(all_cols_except_temp_wind)
```

```
coldest_day <- airquality[which.min(airquality$Temp), "Day"]
print(coldest_day)

num_windy_days <- sum(airquality$Wind > 17, na.rm = TRUE)
print(num_windy_days)
```

5. Write R Program to find maximum and minimum value of a given vector using control statement.

```
vec <- c(2, 5, 7, 1, 8, 4)
max_val <- vec[1]
min_val <- vec[1]
for (i in 2:length(vec)) {
  if (vec[i] > max_val) {
    max_val <- vec[i]
  }
  if (vec[i] < min_val) {
    min_val <- vec[i]
  }
}
print(paste("Maximum value:", max_val))
print(paste("Minimum value:", min_val))
[1] "Maximum value: 8"
[1] "Minimum value: 1"
```

1. Create the following matrices (i) Square Matrix (ii) Identity Matrix (iii)diagonal matrix

```
mat <- matrix(c(1, 2, 3, 4, 5, 6, 7, 8, 9), nrow = 3, ncol = 3)
mat
```

```
     [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
mat <- diag(4)
mat
     [,1] [,2] [,3] [,4]
[1,]    1    0    0    0
[2,]    0    1    0    0
[3,]    0    0    1    0
[4,]    0    0    0    1
```

2. Using sapply, check that all elements of the list are vectors of the same length.
Also calculate the sum of each element.

```
my_list <- list(c(1, 2, 3), c(4, 5, 6), c(7, 8, 9))
sums <- sapply(my_list, sum)
print(sums)
[1]  6 15 24
```

3. We found out that the blood pressure instrument is under-recording each measure and all measurement incorrect by 0.1. How would you add 0.1 to all values in the blood vector?

```
blood <- c(120.3, 130.5, 115.2, 125.8, 118.6)
```

```
blood_corrected <- blood + 0.1
```

print(blood_corrected)

[1] 120.4 130.6 115.3 125.9 118.7

4. We found out that the first patient is 33 years old. How would you change the first element of the vector age to 33 years?

age <- c(25, 40, 33, 55, 30)

age[1] <- 33

print(age)

[1] 33 40 33 55 30

5. Suppose A = [1 1 3 5 2 6 −2 −1 −3] (a) Check that A 3 = 0 where 0 is a 3 × 3 matrix with every entry equal to 0. (b) Replace the third column of A by the sum of the second and third columns

A <- c(1, 1, 3, 5, 2, 6, -2, -1, -3)

A_mat <- matrix(A, nrow = 3)

zeros <- matrix(0, nrow = 3, ncol = 3)

```
result <- A_mat %*% zeros

print(result)
     [,1] [,2] [,3]
[1,]    0    0    0
[2,]    0    0    0
[3,]    0    0    0
```

1.a. The numbers below are the first ten days of rainfall amounts in 1996. Read them into a vector using

the c() function

1.1 0.6 33.8 1.9 9.6 4.3 33.7 0.3 0.0 0.1

```
rainfall <- c(1.1, 0.6, 33.8, 1.9, 9.6, 4.3, 33.7, 0.3, 0.0, 0.1)

print(rainfall)

[1]  1.1  0.6 33.8  1.9  9.6  4.3 33.7  0.3  0.0  0.1
```

b. What was the mean rainfall, how about the standard deviation?

```
rainfall <- c(1.1, 0.6, 33.8, 1.9, 9.6, 4.3, 33.7, 0.3, 0.0, 0.1)

mean_rainfall <- mean(rainfall)

sd_rainfall <- sd(rainfall)
```

```
cat("Mean rainfall: ", mean_rainfall, "\n")
cat("Standard deviation of rainfall: ", sd_rainfall, "\n")
```

```
 Mean rainfall:  7.54
Standard deviation of rainfall:  13.80578
```