Z-SCOORE NORMALIZATION:

AIM:

To write the program for Z-scoore normalization using R-tool.

PROGRAM:

```
library(readxl)  # Ensure readxl package is loaded

# Read the Excel file
file_path <- "C:/Users/harik/Downloads/Data Excel sheet for study on DM.xlsx"
diabetest1 <- read_excel(file_path)

# Clean column names (remove extra spaces)
colnames(diabetest1) <- trimws(colnames(diabetest1))

# Function to compute Z-score for a given column
compute_zscore <- function(data, column_name) {
  if (column_name %in% colnames(data)) {
    # Convert column to numeric
    data[[column_name]] <- as.numeric(as.character(data[[column_name]]))

    # Remove NA values
    values <- na.omit(data[[column_name]])

    # Compute mean and standard deviation
    Mean <- mean(values, na.rm = TRUE)
    Std <- sd(values, na.rm = TRUE)

    # Compute Z-score
    Zscore <- (values - Mean) / Std

    # Print results
```

```
    print(paste("Z-score for column:", column_name))
    print(Zscore)
  } else {
    print(paste("Column", column_name, "not found in the dataset."))
  }
}

compute_zscore(diabetest1, "AGE")
compute_zscore(diabetest1, "Glucose")
```

Output:

```
R 4.4.2 · ~/
> source("~/.active-rstudio-document", echo=TRUE)

> library(readxl)  # Ensure readxl package is loaded

> # Read the Excel file
> file_path <- "C:/Users/harik/Downloads/Data Excel sheet for study on DM.xlsx"

> diabetest1 <- read_excel(file_path)

> # Clean column names (remove extra spaces)
> colnames(diabetest1) <- trimws(colnames(diabetest1))

> # Function to compute Z-score for a given column
> compute_zscore <- function(data, column_name) {
+   if (column_name %in% colnames(data)) {
+     .... [TRUNCATED]

> # Example: Compute Z-score for "AGE" column
> compute_zscore(diabetest1, "AGE")
[1] "Z-score for column: AGE"
 [1]  0.06433163  1.44993594 -0.16660242 -0.39753647 -0.51300350 -0.16660242  0.98806783  1.79633701 -0.62847053
[10] -1.20580565 -1.09033863  0.64166676 -0.05113540  0.75713378 -0.39753647  1.10353486 -0.74393755  0.41073271
[19]  1.33446891  1.21900189  1.21900189  0.52619973 -0.97487160 -1.09033863 -0.97487160 -0.85940458 -1.09033863
[28]  0.29526568  1.10353486 -0.97487160 -1.20580565  1.10353486 -1.55220673  1.44993594 -1.66767376  0.98806783
[37] -2.12954186 -0.51300350 -0.28206945 -0.85940458 -0.39753647  1.44993594 -1.66767376  0.75713378 -0.28206945
[46]  0.52619973  0.52619973  0.17979865  0.52619973  0.98806783 -0.51300350  1.10353486  1.44993594 -0.39753647
[55]  0.87260081 -1.55220673 -0.62847053 -0.28206945 -0.28206945 -0.74393755 -0.05113540  1.33446891  1.44993594
[64] -1.55220673 -0.74393755  0.52619973 -0.85940458 -0.97487160  0.87260081  1.44993594

> # Example: Compute Z-score for another column (e.g., "Glucose")
> compute_zscore(diabetest1, "Glucose")
[1] "Column Glucose not found in the dataset."
>
```