**Arithmetic Operations and Mathematical Functions**

**o Example: Basic Arithmetic Operations**

# Addition

result\_add <- 5 + 3

cat("Addition:", result\_add, "\n")

# Subtraction

result\_sub <- 7 - 2

cat("Subtraction:", result\_sub, "\n")

# Multiplication

result\_mult <- 4 \* 6

cat("Multiplication:", result\_mult, "\n")

# Division

result\_div <- 10 / 2

cat("Division:", result\_div, "\n")

output:

> # Addition

> result\_add <- 5 + 3

> cat("Addition:", result\_add, "\n")

Addition: 8

> # Subtraction

> result\_sub <- 7 - 2

> cat("Subtraction:", result\_sub, "\n")

Subtraction: 5

> # Multiplication

> result\_mult <- 4 \* 6

> cat("Multiplication:", result\_mult, "\n")

Multiplication: 24

> # Division

> result\_div <- 10 / 2

> cat("Division:", result\_div, "\n")

Division: 5

**o Example: Mathematical Functions**

**# Square root**

sqrt\_result <- sqrt(25)

cat("Square root:", sqrt\_result, "\n")

# Exponential function

exp\_result <- exp(2)

cat("Exponential function:", exp\_result, "\n")

# Trigonometric functions

sin\_result <- sin(pi/2)

cat("Sine of pi/2:", sin\_result, "\n")

output:

sqrt\_result <- sqrt(25)

> cat("Square root:", sqrt\_result, "\n")

Square root: 5

> # Exponential function

> exp\_result <- exp(2)

> cat("Exponential function:", exp\_result, "\n")

Exponential function: 7.389056

> # Trigonometric functions

> sin\_result <- sin(pi/2)

> cat("Sine of pi/2:", sin\_result, "\n")

Sine of pi/2: 1

**Statistical Functions**

o Example: Using Statistical Functions

# Generate random data

set.seed(123)

data <- rnorm(100)

# Mean

mean\_result <- mean(data)

cat("Mean:", mean\_result, "\n")

# Standard deviation

sd\_result <- sd(data)

cat("Standard deviation:", sd\_result, "\n")

# Quantiles

quantiles\_result <- quantile(data, probs = c(0.25, 0.5, 0.75))

cat("Quantiles (25th, 50th, 75th percentiles):", quantiles\_result, "\n")

output:

**> set.seed(123)**

**> data <- rnorm(100)**

**> # Mean**

**> mean\_result <- mean(data)**

**> cat("Mean:", mean\_result, "\n")**

**Mean: 0.09040591**

**> # Standard deviation**

**> sd\_result <- sd(data)**

**> cat("Standard deviation:", sd\_result, "\n")**

**Standard deviation: 0.9128159**

**> # Quantiles**

**> quantiles\_result <- quantile(data, probs = c(0.25, 0.5, 0.75))**

**> cat("Quantiles (25th, 50th, 75th percentiles):", quantiles\_result, "\n")**

**Quantiles (25th, 50th, 75th percentiles): -0.4938542 0.06175631 0.6918192**

**Plotting Data**

o Example: Creating Basic Plots

# Generate data for plotting

x <- seq(0, 10, length.out = 100)

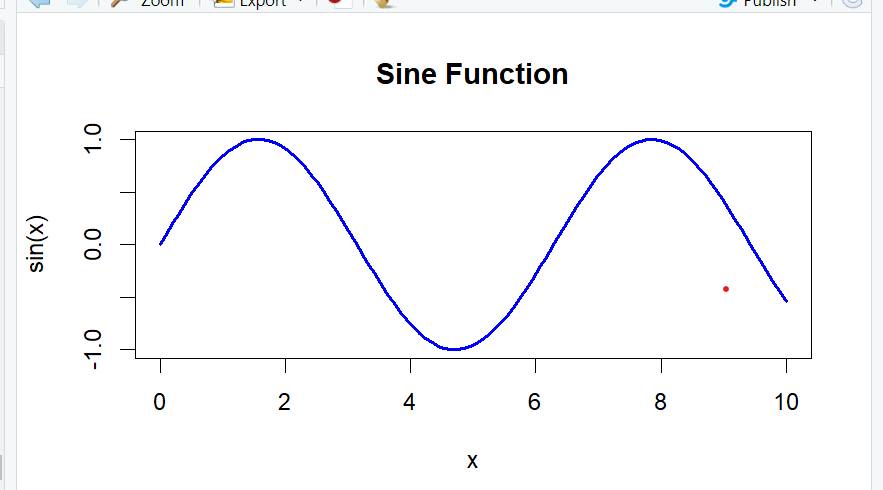
y <- sin(x)

# Plotting

plot(x, y, type = "l", col = "blue", lwd = 2, main = "Sine Function", xlab = "x",

ylab = "sin(x)")

output:



**Exercises**

**1. Exercise 1: Arithmetic Operations**

o Task: Perform arithmetic operations to calculate the area of a rectangle with

length 8 units and width 5 units. Print the result.

o Expected Output: Display the calculated area of the rectangle.

# Calculate area of rectangle

length <- 8

width <- 5

area <- length \* width

# Print result

cat("Area of rectangle:", area, "\n")

output:

> length <- 8

> width <- 5

> area <- length \* width

> # Print result

> cat("Area of rectangle:", area, "\n")

Area of rectangle: 40

**2. Exercise 2: Mathematical Functions**

o Task: Use R functions to calculate the factorial of 6 and the natural logarithm of

10. Print the results.

o Expected Output: Display the factorial and logarithm results.

# Calculate factorial and logarithm

factorial\_result <- factorial(6)

log\_result <- log(10)

# Print results

cat("Factorial of 6:", factorial\_result, "\n")

cat("Natural logarithm of 10:", log\_result, "\n")

output:

> factorial\_result <- factorial(6)

> log\_result <- log(10)

> cat("Factorial of 6:", factorial\_result, "\n")

Factorial of 6: 720

> cat("Natural logarithm of 10:", log\_result, "\n")

Natural logarithm of 10: 2.302585

**3. Exercise 3: Statistical Functions and Plotting**

o Task: Generate 100 random numbers from a normal distribution with mean 50

and standard deviation 10. Calculate the mean and standard deviation of the

generated data. Plot a histogram of the data.

o Expected Output: Display the calculated mean and standard deviation, and

visualize the histogram.

# Generate random data

set.seed(456)

data <- rnorm(100, mean = 50, sd = 10)

# Calculate mean and standard deviation

mean\_result <- mean(data)

sd\_result <- sd(data)

# Print results

cat("Mean:", mean\_result, "\n")

cat("Standard deviation:", sd\_result, "\n")

# Plot histogram

hist(data, breaks = 10, col = "skyblue", main = "Histogram of Random Data", xlab =

"Values", ylab = "Frequency")

Output:

> fact<-factorial(5)

> cat(fact,"\n")

120

> l<-log(20)

> cat("log",l,"\n")

log 2.995732

> set.seed(456)

> data <- rnorm(100, mean = 50, sd = 10)

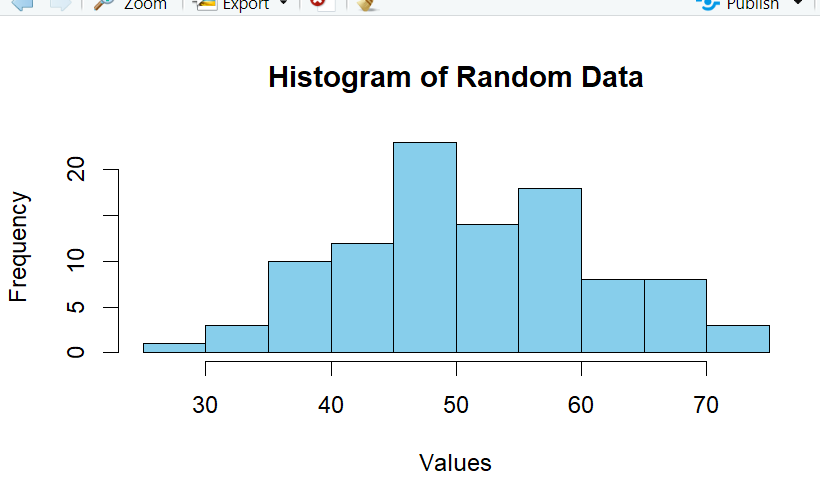
> mean\_result <- mean(data)

> sd\_result <- sd(data)

> cat("Mean:", mean\_result, "\n")

Mean: 51.20575

> cat("sd:", sd\_result

****