Assignment – 1 ' R Programming'

Q1. Write a simple R program to print "Hello, World!" to the console. (1 Mark | 50+ words)

In R, the print() function or cat() function is used to display output on the console. The following is a simple R program that prints **Hello, World!**. This is usually the first program written by beginners to understand basic syntax and execution.

```
print("Hello, World!")
```

When this command is executed, R sends the text to the console exactly as written. This program confirms that the R environment is working correctly.

Q2. In what areas is R primarily used, and why is it popular in data science? (1 Mark | 50+ words)

R is primarily used in **statistics, data analysis, data visualization, machine learning, and research**. It is very popular in data science because it has powerful built-in statistical functions, a large collection of packages like ggplot2 and dplyr, and strong visualization capabilities. R is open-source, flexible, and widely used by data analysts and researchers.

Q3. How would you assign a value to a variable in R? Write the syntax using both <- and =.

In R, variables can be assigned values using either the **assignment operator <-** or the **equal sign =**. Both methods store a value in a variable name. The <- operator is more commonly used in R programming.

```
x <- 10

y = 20
```

Both statements assign numeric values to variables, and the values can be used later in calculations.

Q4. Describe the difference between the == operator and the = operator in R. (1 Mark | 50+ words)

The = operator is used for **assigning values** to variables, while the == operator is used for **comparison**. The == operator checks whether two values are equal and returns TRUE or FALSE.

Example:

```
x = 5

x == 5   # TRUE
```

Thus, = assigns values, whereas == compares values.

Q5. What is the length of x? x <- 5:10 (1 Mark | 50+ words)

The statement x <- 5:10 creates a sequence of integers starting from 5 up to 10. The values are **5, 6, 7, 8, 9, and 10**. Since there are six numbers in this sequence, the **length of x is 6**. The length can be verified using the length() function in R.

Q6. Create a vector of integers in R. How can you check the type of the vector? (1 Mark | 50+ words)

A vector of integers in R can be created using the c() function. The data type of the vector can be checked using the typeof() or class() function.

```
v <- c(1, 2, 3, 4, 5)

typeof(v)
```

This will return "double" because numeric values in R are stored as doubles by default.

Q7. How would you handle missing data (NA) in R? Provide an example. (2 Marks | 80+ words)

Missing data in R is represented using NA. Handling missing values is important because many functions return errors if NA values are present. One common method is

using the na.rm = TRUE argument to ignore missing values. Another method is removing missing values using na.omit().

Example:

x <- c(10, 20, NA, 30)

mean(x, na.rm = TRUE)

This calculates the mean while ignoring NA. Proper handling ensures accurate data analysis.

Q8. What is subsetting in R? Provide an example using a vector. (2 Marks | 80+ words)

Subsetting in R means **extracting specific elements** from a data structure such as a vector, list, or data frame. It allows users to access required data efficiently using index positions or conditions.

Example:

v <- c(10, 20, 30, 40, 50)

v[2:4]

This code extracts elements from position 2 to 4. Subsetting is widely used in data manipulation and analysis.

Q9. List and explain the main data types in R. (4 Marks | 120+ words)

R supports several main data types.
Numeric is used for numbers with or without decimals.
Integer stores whole numbers and uses the L suffix.
Character stores text or strings.
Logical stores boolean values TRUE or FALSE.
Complex stores numbers with imaginary parts.

Each data type differs in storage and usage. Numeric and integer are used for calculations, character for text processing, logical for conditions, and complex for

mathematical computations. Understanding data types helps prevent errors and ensures efficient memory usage.

Q10. Best practices for naming variables in R and their importance. (4 Marks | 120+ words)

Best practices for naming variables in R include using **meaningful and descriptive names**, avoiding special characters, starting names with letters, and using consistent naming styles such as snake_case. Variable names should reflect their purpose clearly, for example student_marks instead of x1.

Good naming practices improve **readability, debugging, and maintenance** of code. They help other programmers understand the logic easily and reduce confusion. Clean variable naming also makes long data science projects manageable and professional.

Q11. Key features of R and why it is suitable for data science. (6 Marks | 150+ words)

R has many powerful features that make it ideal for data science. It provides **strong statistical computing capabilities**, built-in mathematical functions, and advanced data visualization tools. R has thousands of packages available through CRAN, enabling machine learning, data cleaning, and modeling.

It supports data frames, which are perfect for structured data analysis. R is open-source, platform-independent, and widely supported by the data science community. Its ability to integrate with databases, Python, and big-data tools further increases its usefulness. These features make R a preferred language for data analysis and research.

Q12. Operators in R, BODMAS rule & equation solving. (6 Marks | 150+ words)

Operators in R include **arithmetic, relational, logical, and assignment operators**. Arithmetic operators perform calculations, such as +, -, *, /, and %% (modulus). R follows the **BODMAS rule**, meaning Brackets, Orders, Division, Multiplication, Addition, and Subtraction.

Equation:

5 + 4 * 9 %% (3 + 1) / 6 - 1

Step-by-step:

- Brackets: (3 + 1) = 4
- Modulus: 9 %% 4 = 1
- Multiplication & division: 4 * 1 / 6 = 0.6667
- Addition & subtraction: 5 + 0.6667 - 1 = 4.6667

Final Answer ≈ 4.67

This demonstrates correct operator precedence in R.