

Introduction to R Programming

Introduction to R Programming

Quick reference guide for beginners

1. What Is R?

Aspect	Description
Type	Open-source language & environment for statistical computing and graphics
Origin	Developed in 1993 by Ross Ihaka & Robert Gentleman (University of Auckland)
License	GNU GPL (free to use, modify, distribute)
Core Strengths	Data manipulation, statistical modeling, reproducible research, high-quality visualizations
Community	CRAN (Comprehensive R Archive Network) hosts >19,000 packages; active mailing lists, Stack Overflow, R Bloggers

2. Getting Started

Step	Action
Install R	Download from https://cran.r-project.org (Windows, macOS, Linux)
Install an IDE	RGui (basic), RStudio (most popular), VS Code with R extension, or Jupyter notebooks
Run a script	<code>Rscript myfile.R</code> (command line) or click Run in the IDE
Package management	<code>install.packages("pkg")</code> – install <code>library(pkg)</code> – load <code>remove.packages("pkg")</code> – uninstall

3. Basic Syntax & Data Types

Category	Example	Notes
Comments	<code># This is a comment</code>	Whole line or trailing comment

Introduction to R Programming

Category	Example	Notes
Assignment	<code>x <- 10</code> or <code>x = 10</code>	<code><-</code> is idiomatic; <code>=</code> works inside function calls
Vectors	<code>v <- c(1, 2, 3)</code>	Homogeneous; recycling rules apply
Sequences	<code>seq(1, 10, by = 2)</code> or <code>1:10</code>	Shortcut for integer sequences
Matrices	<code>m <- matrix(1:9, nrow = 3, byrow = TRUE)</code>	2D, column-major storage
Lists	<code>lst <- list(name="Alice", age=30, scores=c(90,85))</code>	Heterogeneous, can contain any object
Data frames	<code>df <- data.frame(id=1:3, gender=c("F","M","F"))</code>	Tabular data; each column is a vector
Factors	<code>f <- factor(c("yes","no","yes"))</code>	Categorical data with levels
Logical	<code>TRUE, FALSE</code>	Used for subsetting, conditionals
Missing values	<code>NA, NaN, NULL</code>	<code>is.na()</code> , <code>is.nan()</code> , <code>is.null()</code> to test

4. Essential Operations

4.1 Indexing & Subsetting

```
v[2]           # 2nd element
v[ c(1,3) ]    # elements 1 and 3
v[ v > 2 ]     # conditional subset
df[1:2, "age"]  # rows 1&2, column "age"
df$age         # shortcut for column
df[[ "age" ]]  # same as df$age
```

4.2 Vectorized Arithmetic

```
x <- 1:5
y <- x * 2      # element-wise multiplication
z <- x + y      # addition
```

Introduction to R Programming

All arithmetic is automatically vectorized; no explicit loops needed.