R Básico

Manoel Lucas Batista Vieira

2023-11-22

# o que é R?

É uma linguagem de programação multi-paradigma orientada a objetos, programação funcional, dinâmica, fracamente tipada, voltada à manipulação, análise e visualização de dados.



# O que é RStudio?

O RStudio é um ambiente de desenvolvimento integrado (IDE) para a linguagem de programação R. Ele fornece uma interface gráfica amigável e diversas ferramentas que facilitam o desenvolvimento, a execução e a visualização de análises estatísticas e científicas feitas com R.



# Por que devo aprende-lo?

Aumenta as oportunidades, é gratuito, etc.

# O que é um vetor?

Vetores no R são apenas conjuntos indexados de valores. Para criá-los, basta colocar os valores separados por vírgulas dentro de um c() . Os objetos vetor1 e vetor2 são vetores. Uma maneira fácil de criar um vetor com uma sequência de números é utilizar o operador :

## Console

No contexto do R Studio, o termo “console” refere-se à área interativa onde você pode digitar comandos diretamente e ver a saída imediata. O R Studio é uma interface de desenvolvimento integrada (IDE) para a linguagem de programação R, e o console desempenha um papel crucial na interação com o R.

## Script

O contexto do R Studio, um “script” refere-se a um arquivo que contém um conjunto de comandos em linguagem R. Esses comandos podem ser executados sequencialmente para realizar uma tarefa específica, como análise de dados, geração de gráficos, modelagem estatística, entre outros.

A principal finalidade de um script é automatizar tarefas repetitivas, documentar o processo de análise de dados e permitir a reprodução dos resultados. Em vez de digitar comandos diretamente no console toda vez que você precisa realizar uma análise, você pode escrever esses comandos em um script e executá-los em lote.

## Vetores

Em R, um vetor é uma estrutura de dados fundamental que armazena elementos do mesmo tipo. Vetores podem ser unidimensionais, ou seja, possuem apenas uma dimensão (uma lista ordenada de elementos). No R Studio, um ambiente de desenvolvimento integrado (IDE) para a linguagem R, você frequentemente trabalhará com vetores ao realizar análises de dados.

x <-c(1, 2, 3, 4, 5)  
x

## [1] 1 2 3 4 5

## Matrizes

Uma matriz em R é uma estrutura bidimensional que contém elementos do mesmo tipo organizados em linhas e colunas. Em R Studio, que é um ambiente de desenvolvimento integrado (IDE) para a linguagem R, você pode trabalhar com matrizes para armazenar e manipular dados de forma tabular.

y <-1:12  
mat<-matrix(y,ncol=3)  
mat

## [,1] [,2] [,3]  
## [1,] 1 5 9  
## [2,] 2 6 10  
## [3,] 3 7 11  
## [4,] 4 8 12

## Array

Em R, um array é uma estrutura de dados multidimensional que pode conter elementos do mesmo tipo. Ao contrário de matrizes, que são bidimensionais, arrays podem ter mais de duas dimensões. Em R Studio, que é um ambiente de desenvolvimento integrado (IDE) para a linguagem R, você pode trabalhar com arrays para lidar com dados multidimensionais.

c<-array(1:12, c(2, 3, 2))  
c

## , , 1  
##   
## [,1] [,2] [,3]  
## [1,] 1 3 5  
## [2,] 2 4 6  
##   
## , , 2  
##   
## [,1] [,2] [,3]  
## [1,] 7 9 11  
## [2,] 8 10 12

## Data.frame

Um data.frame é uma estrutura de dados tabular em R que é semelhante a uma tabela de banco de dados ou a uma planilha do Excel. No R Studio, um ambiente de desenvolvimento integrado (IDE) para a linguagem R, o data.frame é uma estrutura fundamental e muito utilizada para armazenar e manipular dados de forma organizada.

frutas<- data.frame(  
 Nome = c("maçã", "banana", "pêra", "uva"),  
 cor = c("vermelha", "amarela", "verde", "roxa"),  
 sabor = c("doce", "doce", "doce", "doce"))  
frutas

## Nome cor sabor  
## 1 maçã vermelha doce  
## 2 banana amarela doce  
## 3 pêra verde doce  
## 4 uva roxa doce

## Lista

Em R Studio, uma “lista” refere-se a uma estrutura de dados que pode conter elementos de diferentes tipos e até mesmo outras listas. As listas são usadas para organizar dados de maneira flexível, permitindo armazenar uma coleção heterogênea de objetos.

# Exemplos de lista  
ip <-list(  
 Pessoa1 = list(  
 Nome = "Alice",  
 Idade = 28,  
 Cidade = " São Paulo"),  
 Pessoa2 = list(  
 Nome = "Clarilene",  
 Idade = 50,  
 Cidade = " Monte Alegre"),  
 Pessoa3 = list(  
 Nome = "Diane",  
 Idade = 22,  
 Cidade = " Santarém"),  
 Pessoa4 = list(  
 Nome = " Gabriel",  
 Idade = 25,  
 Cidade = " Juruti"))  
ip

## $Pessoa1  
## $Pessoa1$Nome  
## [1] "Alice"  
##   
## $Pessoa1$Idade  
## [1] 28  
##   
## $Pessoa1$Cidade  
## [1] " São Paulo"  
##   
##   
## $Pessoa2  
## $Pessoa2$Nome  
## [1] "Clarilene"  
##   
## $Pessoa2$Idade  
## [1] 50  
##   
## $Pessoa2$Cidade  
## [1] " Monte Alegre"  
##   
##   
## $Pessoa3  
## $Pessoa3$Nome  
## [1] "Diane"  
##   
## $Pessoa3$Idade  
## [1] 22  
##   
## $Pessoa3$Cidade  
## [1] " Santarém"  
##   
##   
## $Pessoa4  
## $Pessoa4$Nome  
## [1] " Gabriel"  
##   
## $Pessoa4$Idade  
## [1] 25  
##   
## $Pessoa4$Cidade  
## [1] " Juruti"

## O que é if e else?

No R Studio, assim como em R em geral, if é uma estrutura de controle de fluxo que permite executar diferentes blocos de código com base em uma condição.

O contexto do R Studio, assim como na linguagem de programação R, else é uma palavra-chave que faz parte da estrutura condicional if. A estrutura else é usada para especificar um bloco de código que deve ser executado quando a condição no if não é verdadeira.

# Estrutura condicionais if e else  
g <- 8  
if (g > 7){  
 print(" o número é maior que 7.")  
} else{  
 print( " o número é menor ou igual a 7. ")  
}

## [1] " o número é maior que 7."

número <- -5  
  
if( número > 0){  
 print( " o número é positivo.")  
} else{  
 print( " o número é negativo ou zero.")  
}

## [1] " o número é negativo ou zero."

idades <-c(28, 30)  
nomes <-c("Joao", "Maria")  
df<- data.frame(nomes, idades)  
df

## nomes idades  
## 1 Joao 28  
## 2 Maria 30

if(df$idade[df$nomes == "Joao"] > df$idades[df$nomes == "Maria"]){  
 "mais velho: Joao"  
} else{  
 "mais velho: Maria"  
}

## [1] "mais velho: Maria"

## Loops (for, while)

## While

A palavra-chave while é usada em R, incluindo no R Studio, para criar loops enquanto uma determinada condição for verdadeira. Em outras palavras, o bloco de código dentro do while é repetido continuamente enquanto a condição especificada for verdadeira.

contador <- 5  
while (contador >= 1) {  
 cat(contador, "\n")  
 contador <- contador - 1  
}

## 5   
## 4   
## 3   
## 2   
## 1

cat("Contagem regressiva concluída!")

## Contagem regressiva concluída!

## For

No R, incluindo o R Studio, a estrutura de controle de fluxo é usada para criar loops que permitem que um bloco de código seja repetido um número específico de vezes.

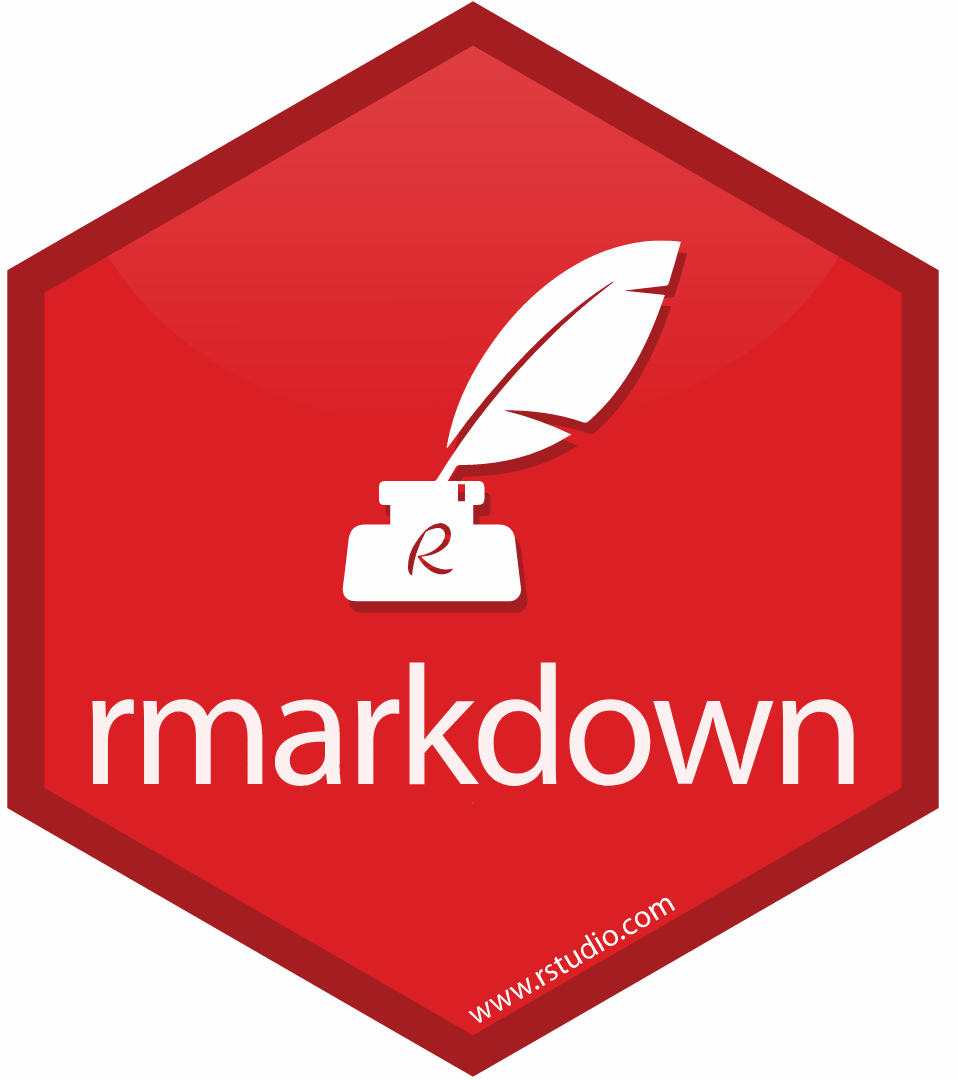
for(i in c(1, 2, 3, 4, 5)) {  
 print(i^2)  
}

## [1] 1  
## [1] 4  
## [1] 9  
## [1] 16  
## [1] 25

# EXTRAS

## Rmarkdown

O R Markdown é uma ferramenta poderosa no R Studio que permite combinar código R, texto e elementos gráficos em um único documento. Isso possibilita a criação de documentos dinâmicos, nos quais você pode incorporar análises, visualizações e narrativas em um formato que pode ser facilmente compartilhado e reproduzido por outros.



Manoel