

Exemplos R

Comandos úteis

1. Verificar qual é a working directory
2. Definir a working directory para o ambiente de trabalho
3. Listar as variáveis de ambiente
4. Criar uma variável (ex: x=1)
5. Listar as variáveis de ambiente
6. Eliminar a variável criada
7. Listar as variáveis de ambiente

Variáveis

1. Verificar que é possível criar as seguintes variáveis:
 - 1.1. division com o valor 3
 - 1.2. Quadrado com o valor "azul"
 - 1.3. .sub.multiplication com o valor "a"
 - 1.4. accumulative_sum com o valor 2
 - 1.5. Sum5 com o valor 10
2. Verificar que não é possível criar as seguintes variáveis, independentemente do valor
 - 2.1. tot@l
 - 2.2. 5um
 - 2.3. _fine
 - 2.4. FALSE
 - 2.5. .0three

Constantes

1. Verificar o tipo das seguintes constantes:
 - 1.1. 2
 - 1.2. 2L
 - 1.3. 2i
 - 1.4. 'example'
 - 1.5. '2'

Operadores

1. Atribuição: atribuir à variável x os seguintes valores
 - 1.1. Valor 20 (operador <-)
 - 1.2. Valor 30 (operador =)
 - 1.3. Valor 5 (operador ->)
2. Aritméticos
 - 2.1. Criar uma variável x com o valor 20 e uma variável y com o valor 15 e efetuar as seguintes operações entre as variáveis:
 - 2.1.1. Soma
 - 2.1.2. Subtração
 - 2.1.3. Multiplicação
 - 2.1.4. Divisão
 - 2.1.5. Parte inteira da divisão
 - 2.1.6. Resto da divisão
 - 2.1.7. Exponenciação
 - 2.2. Relacionais: Criar uma variável x com o valor 20 e uma variável y com o valor 15 e efetuar as seguintes operações:
 - 2.2.1. Verificar se x é menor que y
 - 2.2.2. Verificar se x é maior que y
 - 2.2.3. Verificar se x é menor ou igual a y
 - 2.3. Lógicos: Criar o vetor x com os elementos (TRUE, FALSE, 0, 3) e o vetor y com os elementos (FALSE, TRUE, FALSE, TRUE) e efetuar as seguintes operações:
 - 2.3.1. Não x
 - 2.3.2. x E y (elemento-wise)
 - 2.3.3. x OU y (lógico)

Precedência

Verificar a precedência nas seguintes operações

1. $3 + 4 / 2$
2. $(3 + 4) / 2$
3. $3 / 4 / 2$
4. $3 / (4 / 2)$

Vetores

1. Criação de vetores usando a função `c()`:
 - 1.1. Elementos: 1, 5, 4, 9, 0
 - 1.2. Elementos: 1, 5.4, TRUE, "hello"
2. Criação de vetores usando o operador `":"`
 - 2.1. Criar um vetor com os números inteiros de 1 a 7
 - 2.2. Criar um vetor com os números inteiros de 2 a -2
3. Criação de vetores usando a função `seq()`
 - 3.1. Criar um vetor com valores de 1 a 3 com intervalo de 0.2 (ex: 1, 1.2, 1.4, ...)
 - 3.2. Criar um vetor com valores de 1 a 5 com 4 elementos
4. Funções para vetores
 - 4.1. Criar o seguinte vetor: `a <- c(4, 3, 2, 5, 2, 4, 3, 1, 2, 4)`
 - 4.2. Obter o tamanho do vetor
 - 4.3. Ordenar o vetor de forma crescente
 - 4.4. Obter os valores únicos do vetor
 - 4.5. Obter a frequência dos elementos do vetor
5. Leitura usando vetor lógico como índice
 - 5.1. Criar o vetor `b <- seq(-3, 3, 2)`
 - 5.2. Mostrar apenas o primeiro e quarto elementos do vetor
 - 5.3. Mostrar os elementos negativos do vetor
 - 5.4. Mostrar os elementos positivos do vetor
6. Leitura usando vetor de caracteres como índice
 - 6.1. Criar o vetor `d <- c(first=3, second=0, third=9)`
 - 6.2. Visualizar os nomes dos elementos
 - 6.3. Obter o elemento cujo nome é "second"
 - 6.4. Obter os elementos de nomes "first" e "third"
7. Modificar um vetor
 - 7.1. Criar um vetor `e <- seq(-3, 9, 1)`
 - 7.2. Substituir o segundo elemento do vetor pelo valor 0
 - 7.3. Substituir os elementos negativos pelo valor 12
 - 7.4. Criar um vetor `f` com os 4 primeiros elementos do vetor `e`

Matrizes

1. Criar uma matriz usando a função `matrix()`
 - 1.1. Valores de 1 a 9, com 3 linhas e 3 colunas
 - 1.2. Valores de 1 a 9, com 3 linhas e 3 colunas, distribuídos por linhas
 - 1.3. Mudar os nomes das linhas para L1, L2, L3 e das colunas para C1, C2, C3
2. Criação de matriz usando as funções `cbind()` e `rbind()`
 - 2.1. Criar uma matriz pela junção por colunas dos vetores `c(1,2,3)` e `c(4,5,6)`
 - 2.2. Criar uma matriz pela junção por linhas dos vetores `c(7,8,9)` e `c(10,11,12)`
3. Leitura usando uma matriz de inteiros como índice (utilizando a matriz de 1.1)
 - 3.1. Mostrar apenas a primeira linha
 - 3.2. Mostrar apenas a primeira coluna
 - 3.3. Mostrar toda a matriz, exceto a primeira linha
 - 3.4. Mostrar a interseção das linhas 1 e 2 com as colunas 2 e 3
 - 3.5. Mostrar todas as colunas das linhas 1 e 2
4. Leitura usando uma matriz de valores lógicos como índice
 - 4.1. Mostrar os valores pares existentes na matriz (nota: o resto da divisão de um nº por 2 é 0)
5. Leitura usando uma matriz de caracteres como índice (utilizando a matriz de 1.3)
 - 5.1. Mostrar as colunas cujos nomes são C1 e C2
 - 5.2. Mostrar as linhas cujos nomes são L2 e L3
 - 5.3. Mostrar a interseção das segunda e terceira linhas com as colunas C1 e C2
6. Modificar uma matriz
 - 6.1. Substituir o elemento na segunda linha e segunda coluna por 10
 - 6.2. Substituir os elementos inferiores a 5 por 0
 - 6.3. Obter a transposta da matriz
 - 6.4. Adicionar à matriz a coluna a partir do vetor `c(20,30,40)`
 - 6.5. Adicionar à matriz a linha a partir do vetor `c(500,600,700,800)`
 - 6.6. Remover a terceira linha da matriz

Listas

1. Criar uma lista usando a função `list()`
 - 1.1. Criar uma lista com 3 elementos: elemento "a" com o valor 2.5; elemento "b" com o valor TRUE, e elemento "c" com os inteiros de 1 a 3
 - 1.2. Verificar a estrutura da lista
2. Ler elementos de uma lista
 - 2.1. Obter os elementos nas posições 1 a 2 da lista
 - 2.2. Obter todos os elementos da lista, exceto o segundo

- 2.3. Usando um vetor logico, obter o primeiro elemento da lista
- 2.4. Criar uma lista com os seguintes valores: “nome” com o valor “John”, “age” com o valor 19, “speaks” com os valores “English” e “French”
- 2.5. Obter o elemento “age”
- 2.6. Obter os elementos “age” e “speaks”
- 2.7. Obter o elemento “name” usando o operador \$
- 2.8. Obter o primeiro elemento de “speaks” usando o operador \$
3. Modificar uma lista
 - 3.1. Alterar o “name” da lista anterior para “Clair”
 - 3.2. Adicionar à lista o elemento “married” com o valor FALSE

Data frames

1. Criar um data frame usando a função data.frame()
 - 1.1. Criar um data frame com o seguinte conteúdo

SN	Age	Name
1	21	John
2	15	Dora
 - 1.2. Ver a estrutura do data frame
2. Ler data frames
 - 2.1. Ver os Names com três operadores: [], [[]] e \$
 - 2.2. Ver o Name do segundo registo
 - 2.3. Ver o dataset iris
 - 2.4. Ver as primeiras 3 linhas do dataset iris
 - 2.5. Ver os registos do dataset iris cuja Species seja “setosa”
3. Modificar um data frame
 - 3.1. No data frame do primeiro exercício alterar a idade do John para 20
 - 3.2. Adicionar ao data frame do primeiro exercício a coluna “State” com os valores “NY” e “FL” para os registos 1 e 2, respetivamente

Factors

1. Criar um factor usando a função factor()
 - 1.1. Criar um factor com os seguintes valores: “single”, “married”, “married”, “single”
 - 1.2. Criar um factor com os seguintes valores: “single”, “married”, “married”, “single”, definindo os níveis como: “single”, “married”, “divorced”
2. Modificar um fator
 - 2.1. Modificar o segundo elemento de 1.2 para “single”
 - 2.2. Modificar o segundo elemento de 1.2 para “widowed” (impossível)



Funções predefinidas

1. Funções matemáticas
 - 1.1. Calcular a raiz quadrada de 2
 - 1.2. Calcular o cosseno de pi
 - 1.3. Calcular o valor absoluto de -3
2. Funções de texto
 - 2.1. Criar a variável x com o valor "abcdef"
 - 2.2. Mostrar as letras nas posições 2 a 4
 - 2.3. Procurar "A" em c("B","A","C")
 - 2.4. Criar o vetor [x1 x2 x3] por concatenação da letra x com os números de 1 a 3
3. Funções estatísticas
 - 3.1. Criar o vetor x com os elementos 2, 5 e 7
 - 3.2. Calcular a média de x
 - 3.3. Calcular o valor máximo de x
4. Funções úteis
 - 4.1. Criar um vetor com 5 repetições do valor 20