

Operadores e Variáveis em linguagem R

Ivanildo Batista

19 de março de 2021

Operadores matemáticos em R

```
5+5 #soma
```

```
## [1] 10
```

```
5-5 #subtração
```

```
## [1] 0
```

```
5*5 #multiplicação
```

```
## [1] 25
```

```
5/5 #divisão
```

```
## [1] 1
```

```
5^2 #potência
```

```
## [1] 25
```

```
5**5 #potência
```

```
## [1] 3125
```

```
17%%5 #módulo ou resto de uma divisão
```

```
## [1] 2
```

Operadores relacionais

```
#atribuindo valores a duas variáveis
```

```
a=7
```

```
b=5
```

```
a>8 #maior
```

```
## [1] FALSE
```

```
a<8 #menor
```

```
## [1] TRUE
```

```
a<=8 #menor ou igual
```

```
## [1] TRUE
```

```
a>=8 #maior ou igual
```

```
## [1] FALSE
```

```
a==8 #igual (comparação)
```

```
## [1] FALSE
```

```
a!=8 #diferente
```

```
## [1] TRUE
```

```
(a==8)&(b==6) #se a é igual a 8 E b igual a 6 (as duas devem ser verdadeiras)
```

```
## [1] FALSE
```

```
(a==8)|(b>5)#se a é igual 8 OU b maior que 5 (pelo menos uma deve ser verdadeira)
```

```
## [1] FALSE
```

```
print(!a>8) #negando um resultado anterior
```

```
## [1] TRUE
```

Tipos de dados

```
#dados numéricos  
num=2.5 #variável com valor decimal  
num
```

```
## [1] 2.5
```

```
num2=4 #variável com valor inteiro  
num2
```

```
## [1] 4
```

```
x=-123 #variável com valor negativo  
x
```

```
## [1] -123
```

```
class(x)#verificando a classe da variável
```

```
## [1] "numeric"
```

```
is.integer(num2) #perguntando para o R se a variável é um inteiro
```

```
## [1] FALSE
```

```
y=as.integer(num2) #transformando a variável em inteiro  
class(y) #verificando a classe
```

```
## [1] "integer"
```

```
x2 = 3.14  
x2
```

```
## [1] 3.14
```

```
y2=as.integer(3.14) #arredondando a variável  
y2 #valor arredondado
```

```
## [1] 3
```

```
as.integer('4.27') #também é possível arredondar string, desde que seus componentes sejam número  
s
```

```
## [1] 4
```

```
#atribuindo strings a valores  
char1 = 'A'  
char1
```

```
## [1] "A"
```

```
char2='bolha'  
char2
```

```
## [1] "bolha"
```

```
#números complexos  
compl = 2.5+4i  
compl
```

```
## [1] 2.5+4i
```

```
class(compl)
```

```
## [1] "complex"
```

```
#trabalhando com raiz quadrada  
sqrt(-1) #não é possível tirar a raiz quadrada de um número negativo
```

```
## Warning in sqrt(-1): NaNs produzidos
```

```
## [1] NaN
```

```
sqrt(-1+0i) #mas é possível de um número complexo
```

```
## [1] 0+1i
```

```
sqrt(as.complex(-1))#é possível converter um número negativo para complexo e tirar sua raiz quadrada
```

```
## [1] 0+1i
```

Operadores lógicos

```
x=1;y=2  
z=x>y  
z #nessa variável lhe é atribuído um resultado lógico
```

```
## [1] FALSE
```

```
class(z)
```

```
## [1] "logical"
```

```
#também  
u=TRUE; v=TRUE  
class(u)
```

```
## [1] "logical"
```

```
class(v)
```

```
## [1] "logical"
```

```
u&v #usando operador and
```

```
## [1] TRUE
```

```
u|v #usando operador or
```

```
## [1] TRUE
```

```
!u #usando o operador not
```

```
## [1] FALSE
```

Variável

```
#atribuindo valor a uma variável  
var1=367  
var1
```

```
## [1] 367
```

```
mode(var1)
```

```
## [1] "numeric"
```

```
sqrt(var1)
```

```
## [1] 19.15724
```

```
var2 = as.integer(var1)#atribuindo a uma variável outra variável, mas modificando o seu tipo para inteiro  
var2
```

```
## [1] 367
```

```
mode(var2) #modo de armazenamento é numeric
```

```
## [1] "numeric"
```

```
typeof(var2)#mas seu tipo é inteiro
```

```
## [1] "integer"
```

```
var3 = c('seg','ter','qua') #atribuindo caracteres a uma variável  
var3
```

```
## [1] "seg" "ter" "qua"
```

```
mode(var3)
```

```
## [1] "character"
```

```
var4=function(x){x+3} #atribuindo uma função para uma variável  
var4
```

```
## function(x){x+3}
```

```
mode(var4)
```

```
## [1] "function"
```

```
var5 = as.character(var1) #atribuindo outra variável, mas modificando o seu tipo para caracter  
var5
```

```
## [1] "367"
```

```
mode(var5)
```

```
## [1] "character"
```

```
#atribuindo valores a objetos  
x=c(4,5,6)  
x
```

```
## [1] 4 5 6
```

```
#ou  
  
c(4,5,6) -> y  
y
```

```
## [1] 4 5 6
```

```
#ou  
  
assign('x',c(4.3,4,-2))  
x
```

```
## [1] 4.3 4.0 -2.0
```

```
x[1]#verificando valor na posição 1
```

```
## [1] 4.3
```

```
x[2]#verificando valor na posição 2
```

```
## [1] 4
```

```
x[3]#verificando valor na posição 3
```

```
## [1] -2
```

Listando todos os objetos*

```
ls()
```

```
## [1] "a"      "b"      "char1" "char2" "compl" "num"    "num2"  "u"     "v"  
## [10] "var1"    "var2"    "var3"   "var4"   "var5"   "x"      "x2"    "y"     "y2"  
## [19] "z"
```

```
objects()
```

```
## [1] "a"      "b"      "char1" "char2" "compl" "num"    "num2"  "u"     "v"  
## [10] "var1"    "var2"    "var3"   "var4"   "var5"   "x"      "x2"    "y"     "y2"  
## [19] "z"
```