Clase de programación en R - $N^0.1$

Victor Augusto Lizcano Sandoval

October 13, 2021

¿Qué es R?

R es un lenguaje de programación comúnmente empleado para análisis estadístico y matemático.

"Hello World!"

```
> Hello World!
Error:  unexpected symbol in "Hello World"
> "Hello World!"
[1] "Hello World!"
> print("Hello World!")
[1] "Hello World!"
```

Función print()

```
> print(5+5)
[1] 10
> print("5+5")
[1] "5+5"
```

Función cat()

La función cat() se utiliza para imprimir en la pantalla o en un archivo.

```
> cat("Hola como estás?")
Hola como estás?
> cat("hola\ncómo\nestás?")
hola
cómo
estás?
```

Función cat()

```
> cat(5+5)
10
> cat("5+5")
5+5
> cat(1:5, sep="-")
1-2-3-4-5
```

Comentarios

```
>#  Este es un comentario
```

Variables

Las variables son contenedores para el almacenamiento de datos.

Variables

Las variables son contenedores para el almacenamiento de datos.

Variables

```
> #variables
> Nombre = "María"
> Apellido <- "Magdalena"
> Nombre
[1] "María"
> Apellido
[1] "Magdalena"
```

Variables

```
> x = 2
> y = 5
> x
[1] 2
> y
[1] 5
```

Notación de variables

1. Camel Case (contarElementos)
2. Pascal Case (ContarElementos)
3. Snake Case (contar_elementos)
4. Kebab Case (contar−elementos)

Reglas - declaración de variables

- Una variable, siempre debe iniciar con una letra (mayúscula o minúscula) ó un guión bajo (_).
- Una variable, puede contener números, solamente después de la primer letra (siguiendo la regla anterior).
- No es permitido dejar un espacio en blanco a lo largo de la variable.
- Aunque una variable puede ser del largo que tú desees, lo recomendable es que sea una variable corta (regularmente entre 20 y 30 caracteres como máximo).
- No puedes utilizar palabras reservadas para la declaración de una variable.
- El nombre de una variable en R es case sensitive (es decir, a lo largo de tu programa debe escribirse exactamente igual).
- Utiliza un nombre que exprese algo del contexto en el cual la estás declarando.

Múltiples variables

```
var1 <- var2 <- var3 <- "Naranja"
var1
[1] "Naranja"
var2
[1] "Naranja"
var3
[1] "Naranja"
```

Concatenación de elementos

```
#Concatenación
Texto = "Fantástica"
paste("Eres una persona",Texto, sep=" ")
[1] "Eres una persona Fantástica"
```

Tipos de datos

```
# numeric
x <- 10.5
class(x)
# integer
x <-1000L
class(x)
# complex
x <- 9i + 3
class(x)
# character/string
x <- "R is exciting"
class(x)
# logical/boolean
x <- TRUE
class(x)
```

Conversión de tipo

```
x = 1L # integer
y = 2 # numeric
# convert from integer to numeric:
a = as.numeric(x)
# convert from numeric to integer:
b <- as.integer(y)
# print values of x and y
x
y
# print the class name of a and b
class(a)
class(b)
```

Matemáticas

```
10 + 5 #Suma
10 - 5 #Resta
max(5, 10, 15) #Máximo
min(5, 10, 15) #Mínimo
sqrt(16) #Raíz cuadrada
sd(1:5) #Desviación estandar
abs(-4.7) #Valor absoluto
ceiling(1.4) #Redondear arriba
floor(1.4) #Redondear abajo
```

Cadenas de texto

```
str <- "We are the so-called \"Vikings\" , from the
north."
cat(str)
We are the so-called "Vikings", from the north.
str
[1] "We are the so-called \"Vikings\" , from the
north."
nchar(str) #Longitud de la cadena
```

Caracteres especiales

\\	Backslash
\n	Nueva linea
\r	Retorno
\t	tab
\b	Retroceso

Booleanos

```
10 > 9 # TRUE because 10 is greater than 9
10 == 9 # FALSE because 10 is not equal to 9
10 < 9 # FALSE because 10 is greater than 9
```

Operadores aritmeticos

Suma $(+)$
Resta $(-)$
Multiplicación $(*)$
División $(/)$
Exponente $(\hat{})$
Módulo $(\%\%)$
Divisón de enteros $(\%/\%)$

Operadores de asignación

$my\_var <- 3$
$my\_var = 3$
$my\_var <<- 3$
$3 -> my\_var$
$3 ->> my\_var$

Operadores de comparación

==	Igual
!=	No igual
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que

Operadores lógicos

&	Operador "y"
&&	Operador "y"
|	Operador "o"
||	Operador "o"
!	Operador "No"

Operadores miscelaneos

:	Crear secuencia de números
$\%in\%$	Buscar un elemento dentro un vector
$\% * \%$	Multiplicar matriz