

Ejercicios r4ds

ronny hdez-mora

5/25/2019

Parte III

Capítulo 15

Funciones

##2 Hacer función propia para la varianza

```
secuencia <- c(10, 15, 30, 28, NA, 8)

varianza <- function(x) {
  media <- mean(secuencia, na.rm = TRUE)
  total <- length(secuencia) - 1
  sumatoria <- sum(cuadrado <- (x - media)^2)
  return(sumatoria / total)
}
```

##5 Hacer función para tomar dos vectores y

```
secuencia_a <- c(10, 15, 30, 28, NA, 8)
secuencia_b <- c(10, NA, 30, 28, 17, 8)

both_na <- function(x, y) {
  return(paste("La posición de x es:", which(is.na(x)), ", y",
              "la posición de y es:", which(is.na(y))))
}

both_na(secuencia_a, secuencia_b)
```

[1] "La posición de x es: 5 , y la posición de y es: 2"

##6 ¿Qué hacen las siguientes funciones?

```
is_directory <- function(x) file.info(x)$is.dir
is_readable <- function(x) file.access(x, 4) == 0
```

Functions are for humans and computers

##1 Leer el código de las funciones y hacer nombres

```
# Tomar en cuenta lo que hace doble o simple & /
a <- c(1:5)
b <- c(1:5)

c <- c(6:10)
d <- c(6:10)
```

```

a == b & c == d

## [1] TRUE TRUE TRUE TRUE TRUE
a == b && c == d

## [1] TRUE
5 > 4 & 6 > 3

## [1] TRUE
5 > 4 && 6 > 3

## [1] TRUE
5 < 4 & 6 > 3

## [1] FALSE
5 < 4 && 6 > 3

## [1] FALSE
5 < 4 | 6 > 3

## [1] TRUE
5 < 4 || 6 > 3

## [1] TRUE
a <- function(x, y, op) {
  switch (op,
    plus = x + y,
    minus = x - y,
    times = x * y,
    divide = x / y,
    stop("Unknown op!")
  )
}

```

Escribir funcion que diga buenos dias, buenas tardes o buenas noches de acuerdo a la hora del dia.

```

saludo <- function(hora = lubridate::now()){
  if (hour(hora) <= 12) {
    print(";Buenos días!")
  } else if (hour(hora) > 12) {
    print(";Buenas tardes!")
  } else {
    print(";Buenas noches!")
  }
}

```

##3 Escribir funcion que si numero es divisible por 3 imprime ## fizz, si es divisible por tres y cinco imprime fizzbuzz. Lo ## demas imprime el numero

```
fizzbuzz <- function(x) {
  if(x %% 3 == 0 & x %% 5 == 0) {
    print("fizzbuzz")
  } else if(x %% 3 == 0) {
    print("fizz")
  } else {
    return(x)
  }
}
```

Una funcion de ejemplo con dot-dot-dot

```
comas <- function(...) {
  stringr::str_c(..., collapse = ",")
}
```

```
comas(letters[1:5])
```

```
## [1] "a,b,c,d,e"
```

Capítulo 17

Uno de los primeros ejemplos de loops:

```
df <- tibble(
  a = rnorm(10),
  b = rnorm(10),
  c = rnorm(10),
  d = rnorm(10)
)

output <- vector("double", ncol(df))

for (i in seq_along(df)) {
  output[[i]] <- median(df[[i]])
}
```

Ejercicios

1 Generar loops para

Cada columna en `mtcars`

Determinar el tipo de cada columna en `flights13::flights`

Computar el número de valores únicos en cada columna de `iris`

Generar