

Ejercicios de funciones en R

David Criado Ramón

26/10/2019

1. Crea una función `impares` que dado un vector devuelva la cantidad de elementos impares que contiene.

```
impares <- function(v) {  
  length(which(v%%2==1))  
}  
prueba <- 1:10  
n_impares <- impares(prueba)  
n_impares
```

```
## [1] 5
```

2. Crea una función `cambio` que dada una matriz de números enteros reemplace todos los NA por el valor 0.

```
cambio <- function(m) {  
  m[which(is.na(m))] <- 0  
  m  
}  
  
matriz <- matrix(1:9, nrow=3, ncol=3)  
matriz[2,2] <- NA  
matriz[3,1] <- NA  
matriz <- cambio(matriz)  
matriz
```

```
##      [,1] [,2] [,3]  
## [1,]    1    4    7  
## [2,]    2    0    8  
## [3,]    0    6    9
```

3. Crea una función `unir` que dados dos vectores devuelva un nuevo vector con los elementos de ambos vectores sin repetidos.

```
unir <- function(v1, v2) {  
  union(v1, v2)  
}  
  
v1 <- c(1,2,3,4,5,9)  
v2 <- c(4,5,6,7,8,9,10)  
  
v_out <- unir(v1, v2)  
v_out
```

```
## [1] 1 2 3 4 5 9 6 7 8 10
```

4. Crea una función `vyc` que dado un string devuelva una lista de dos componentes que contenga las vocales y las consonantes.

```
vyc <- function(s) {  
  letras <- unlist(strsplit(s, ''))  
  pos_vocales <- grep("[aeiou]", letras)  
  vocales <- letras[pos_vocales]  
  consonantes <- letras[-pos_vocales]  
  list(consonantes, vocales)  
}  
  
my_vic <- vyc("Prueba")  
my_vic
```

```
## [[1]]  
## [1] "P" "r" "b"  
##  
## [[2]]  
## [1] "u" "e" "a"
```

5. Crea una función `partir` que dado un vector `v` y dos valores `x` e `y` (siendo `y` opcional), retorne un vector con los valores que aparecen luego del primer `x` y hasta el primer `y`. De no indicarse el valor de `y`, se devolverán todos los valores que aparecen luego del primer `x` hasta el final del vector.

```
partir <- function(v, x, y=F) {  
  pos_ini <- which(v==x)[1]  
  if (y) {  
    pos_fin <- which(v==y)[1]  
  }  
  else {  
    pos_fin <- length(v)  
  }  
  v[pos_ini:pos_fin]  
}  
  
v <- c(1,2,3,4,5,3)  
partir(v, 3, 5)
```

```
## [1] 3 4 5  
partir(v,2)
```

```
## [1] 2 3 4 5 3
```