

Ejercicios Tema 1. Primera Entrega.

Pau Vives, Harold Cruz, Samuel de Paúl

PARTE I

1. Utiliza el operador `$` para acceder a los datos del tamaño de la población y almacenarlos como el objeto `pop`. Luego, use la función `sort` para redefinir `pop` para que esté en orden alfabético. Finalmente, usa el operador `[` para indicar el tamaño de población más pequeño.

```
#Cargamos los datos:
#install.packages("dslabs")
library(dslabs)
data(murders)
pop <- murders$population

#Ordenamos Alfabéticamente: (Supondremos que se refiere a crecientemente)
pop2 <- sort(pop)

#Tamaño de población más pequeño:
min_pop <- pop2[1]
min_pop
```

```
## [1] 563626
```

2. Ahora, en lugar del tamaño de población más pequeño, encuentra el índice de la entrada con el tamaño de población más pequeño. Sugerencia: use `order` en lugar de `sort`.

```
min_ind_pop = order(pop)[1]
min_ind_pop
```

```
## [1] 51
```

3. Podemos realizar la misma operación que en el ejercicio anterior usando la función `which.min`. Escribe una línea de código que haga esto.

```
min_ind_pop2 = which.min(pop)
min_ind_pop2
```

```
## [1] 51
```

```
#Vemos que coincide con el apartado anterior
```

4. Ahora sabemos cuán pequeño es el estado más pequeño y qué fila lo representa. ¿Qué estado es? Define una variable `states` para que sea los nombres de los estados del data frame `murders`. Reporta el nombre del estado con la población más pequeña.

```
#Estado más pequeño:
murders$state[min_ind_pop]
```

```
## [1] "Wyoming"
```

```
#Definimos variable states
states = murders$state
```

5. Puedes crear un data frame utilizando la función `data.frame`. Utiliza la función `rank` para determinar el rango de población de cada estado desde el menos poblado hasta el más poblado. Guarda estos rangos en un objeto llamado `ranks`. Luego, crea un data frame con el nombre del estado y su rango. Nombra el data frame `my_df`.

```
ranks = rank(pop)
my_df <- data.frame(states = states , rank = ranks)
head(my_df)
```

```
##      states rank
## 1    Alabama   29
## 2     Alaska    5
## 3    Arizona   36
## 4   Arkansas   20
## 5 California   51
## 6   Colorado   30
```

6. Repite el ejercicio anterior, pero esta vez ordena `my_df` para que los estados estén en orden de menos poblado a más poblado. Sugerencia: cree un objeto `ind` que almacene los índices necesarios para poner en orden los valores de la población. Luego, use el operador de corchete `[` para reordenar cada columna en el data frame.

```
ind = order(pop)
my_df2 <- my_df[ind,]
head(my_df2)
```

```
##      states rank
## 51      Wyoming    1
## 9 District of Columbia  2
## 46      Vermont    3
## 35   North Dakota    4
## 2       Alaska    5
## 42   South Dakota    6
```

7. El vector `na_example` representa una serie de conteos. La función `is.na` devuelve un vector lógico que nos dice qué entradas son NA. Asigna este vector lógico a un objeto llamado `ind` y determina cuántos NAs tiene `na_example`.

```
data("na_example")
str(na_example)
```

```
## int [1:1000] 2 1 3 2 1 3 1 4 3 2 ...
```

```
ind = is.na(na_example)
#total = sum(ind)
length(which(ind == TRUE))
```

```
## [1] 145
```

8. Ahora calcula nuevamente el promedio, pero solo para las entradas que no son NA. Sugerencia: recuerde el operador `!`

```
ind_new = na_example[ind!=TRUE]
mean(ind_new)
```

```
## [1] 2.301754
```

PARTE II:

1. Anteriormente, creamos este data frame:

```
temp <- c(35, 88, 42, 84, 81, 30)
city <- c("Beijing", "Lagos", "Paris", "Rio de Janeiro", "San Juan", "Toronto")
city_temps <- data.frame(name = city, temperature = temp)
```

Vuelve a crear el data frame utilizando el código anterior, pero agrega una línea que convierta la temperatura de Fahrenheit a Celsius. La conversión es $C = 5/9(F - 32)$.

```
temp <- c(35, 88, 42, 84, 81, 30)
celsius <- 5/9 * (temp - 32)
city <- c("Beijing", "Lagos", "Paris", "Rio de Janeiro", "San Juan", "Toronto")
city_temps <- data.frame(name = city, temperature = celsius)
head(city_temps)
```

```
##           name temperature
## 1      Beijing    1.666667
## 2        Lagos   31.111111
## 3        Paris    5.555556
## 4 Rio de Janeiro 28.888889
## 5      San Juan   27.222222
## 6      Toronto  -1.111111
```

2. ¿Cuál es la siguiente suma $1 + 1/2^2 + 1/3^2 + \dots + 1/100^2$? Ayuda: gracias a Euler, sabemos que debería estar cerca de $\pi^2/6$.

#Planteamos dos formas diferentes de resolver el problema.

#Primera manera:

```
vec = c(1:100)
sum1 = sum(1/(vec^2))
sum1
```

```
## [1] 1.634984
```

```
(pi^2)/6
```

```
## [1] 1.644934
```

#Segunda manera:

```
sum2 = 0
for(i in 1:100){
  sum2 = sum2 + 1/(i^2)
}
sum2
```

```
## [1] 1.634984
```

```
(pi^2)/6
```

```
## [1] 1.644934
```

3. Calcula la tasa de asesinatos por cada 100000 habitantes para cada estado y almacénala en el objeto `murder_rate`. Luego, calcula la tasa promedio de asesinatos para EE.UU. con la función `mean`. ¿Cuánto vale la media?

```
murder_rate <- murders$total / murders$population * 100000
murder_mean <- mean(murder_rate)
murder_mean
```

```
## [1] 2.779125
```