

Tarea 10

Código:

#Alumno: Freya Melissa López Flores

#No. de cuenta: A225952-0

#Profesor: Lázaro Bustio Martínez

#Fecha de entrega: 08/02/2023

#Objetivo: El objetivo de este programa es contestar las siguientes preguntas

```
print("INCISO A")
```

#a) a) Crea un vector con estas notas y ponédle un nombre adecuado.

```
calf<-c(7.9, 4.3, 5.5, 7.9, 9.8, 2.7, 4.7, 2.4, 8.3, 7.3, 6.8, 6.3, 4.8, 5.7,  
3.8, 6.3, 5.4, 5.4, 80, 4.2, 8.3, 4.7, 6.0, 6.8, 5.7, 6.5, 4.6, 5.4, 3.7,  
7.1, 5.5, 6.0, 6.7, 7.0, 7.3, 3.0, 6.6, 6.1, 2.4, 7.1, 9.4, 3.7, 4.5, 5.1,  
5.9, 4.7, 5.5, 8.9, 8.1, 8.3, 4.3, 7.1, 9.3, 5.1, 6.1, 3.0, 5.7, 6.8, 3.1,  
7.7, 7.3 , 7.0, 6.2, 8.8, 5.3, 4.0)
```

#b) ¿Cuántas notas contiene este vector? ¿Cuál es su valor medio?

```
print("INCISO B")
```

```
print(paste("Este vector contiene: ", length(calf), " calificaciones."))
```

```
print(paste("El promedio (media) es: ", mean(calf)))
```

#usamos la función mean para sacar el promedio del vector

```
print(paste("El valor medio (mediana) del vector es: ", median(calf)))
```

```
print("INCISO C")
```

#c) Cambiad el 80 del vector anterior por un 8.0, sin volver a entrar el resto de notas.

Volved a calcular la media de las notas tras haber corregido este error

#replace(x, list, values), x es el vector y list s el índice

```
calf<-replace(calf,which(calf == 80),8.0)
```

#con replace cambiamos los valores iguales a 80 a 8.0

```
print(paste("El promedio (media) es: ", mean(calf)))
```

```
print(paste("El valor medio (mediana) del vector es: ", median(calf)))

print("INCISO D")

#d) ¿Cuál es la nota mínima obtenida por estos estudiantes? ¿Cuántos
estudiantes la han sacado?

print(paste("La nota mínima obtenida es: ", min(calf)))

#Usamos la función min para saber el valor mínimo del vector

print(paste(length(which(calf==min(calf))), "estudiantes obtuvieron la nota
mínima"))

#Which nos regresa los índices donde se encuentran los valores mínimos del
vector y con length podemos saber cuántos hay

print("INCISO E")

#¿Cuántos estudiantes han logrado un notable (entre 7 y 8.9)?
#¿Qué porcentaje del total de estudiantes representan?

notable<-length(which(calf>=7 & calf<=8.9))

#notable tiene cuantas calificaciones estan entre el rango deseado

print(paste(notable, "estudiantes obtuvieron una calificación entre 7 y
8.9"))

print(paste("Esto representa el ", notable*100/length(calf), "% de los
estudiantes"))

print("INCISO F")

#¿Qué grupo es más numeroso: el de los estudiantes que han sacado entre 4 y
4.9,
#o el de los que han sacado entre 5 y 5.9?

cuatros<-length(which(calf>=4 & calf<=4.9))
cincos<-length(which(calf>=5 & calf<=5.9))

#cuatros y cincos tiene cuantas calificaciones estan entre el rango deseado
if(cuatros>cincos){
  print(paste("Hay más estudiantes que han sacado entre 4 y 4.9 con ",
cuatros, "estudiantes"))
}
```

```
} else {  
  print(paste("Hay más estudiantes que han sacado entre 5 y 5.9 con ",  
cincos, " estudiantes"))  
}  
  
#El if nos ayuda a ver que grupo es más numeroso  
  
print("INCISO G: Sin usar la funcion median")  
  
#Ordena en orden creciente estas notas y obten su mediana: una vez ordenado  
el vector,  
  
#si tiene un número impar de entradas, su mediana es el valor central, y si  
tiene un número  
  
#par de entradas, su mediana es la media aritmética de los dos valores  
centrales.  
  
calf<-sort(calf)  
mitad<-length(calf)/2  
  
#mitad nos dira el primer valor central  
  
print(paste("La mediana del vector es: ", (calf[mitad]+calf[mitad+1])/2))  
  
  
print("INCISO H: usando median")  
  
# La mediana de un vector se puede calcular directamente con la función  
median. Calcula  
  
#la del vector anterior con esta función. ¿Da lo mismo que el valor obtenido  
en el punto  
  
#anterior?  
  
print(paste("La mediana del vector es: ", median(calf)))  
print("En ambos casos da la misma mediana")  
  
  
print("INCISO I: ")  
  
#¿Cuántos notas diferentes hay en esta muestra? (Podéis emplear astutamente  
#algunas funciones explicadas en esta lección, o podéis consultar  
help.search("duplicated")
```

#a ver si encontráis una función que elimine las entradas duplicadas de un vector.)

```
calf2<-!duplicated(calf)
```

#duplicate nos da false si se repite el dato

```
print(paste("Hay ", length(which(calf2==TRUE)), "calificaciones diferentes"))
```

#Contamos cuantos elementos son true para saber cuantas calificaciones diferentes hay

Ejecución del programa:

```
12:54 | (Top Level) | R Script |
Console | Terminal | Jobs |
R 4.2.2 · ~/ |
> source("~/Documents/Ciencia de Datos/Act_11.R")
[1] "INCISO A"
[1] "Este es el vector: "
[1] "INCISO B"
[1] "Este vector contiene: 66 calificaciones."
[1] "El promedio (media) es: 7.07424242424242"
[1] "El valor medio (mediana) del vector es: 6"
[1] "INCISO C"
[1] "El promedio (media) es: 5.98333333333333"
[1] "El valor medio (mediana) del vector es: 6"
[1] "INCISO D"
[1] "La nota mínima obtenida es: 2.4"
[1] "2 estudiantes obtuvieron la nota mínima"
[1] "INCISO E"
[1] "18 estudiantes obtuvieron una calificación entre 7 y 8.9"
[1] "Esto representa el 27.2727272727273 % de los estudiantes"
[1] "INCISO F"
[1] "Hay más estudiantes que han sacado entre 5 y 5.9 con 13 estudiantes"
[1] "INCISO G: Sin usar la función median"
[1] "La mediana del vector es: 6"
[1] "INCISO H: usando median"
[1] "La mediana del vector es: 6"
[1] "En ambos casos da la misma mediana"
[1] "INCISO I: "
[1] "Hay 40 calificaciones diferentes"
>

> print("INCISO A")
[1] "INCISO A"
> #a) a) Cread un vector con estas notas y ponedle un nombre adecuado.
> calf<-c(7.9, 4.3, 5.5, 7.9, 9.8, 2.7, 4.7, 2.4, 8.3, 7.3, 6.8, 6.3, 4.8, 5.7, 3.8, 6.3, 5.4, 5.4, 80,
+ 4.2, 8.3, 4.7, 6.0, 6.8, 5.7, 6.5, 4.6, 5.4, 3.7, 7.1, 5.5, 6.0, 6.7, 7.0, 7.3, 3.0, 6.6, 6.1,
+ 2.4, 7.1, 9.4, 3.7, 4.5, 5.1, 5.9, 4.7, 5.5, 8.9, 8.1, 8.3, 4.3, 7.1, 9.3, 5.1, 6.1, 3.0, 5.7,
+ 6.8, 3.1, 7.7, 7.3, 7.0, 6.2, 8.8, 5.3, 4.0)
> calf
[1] 7.9 4.3 5.5 7.9 9.8 2.7 4.7 2.4 8.3 7.3 6.8 6.3 4.8 5.7 3.8 6.3 5.4 5.4 80.0
[20] 4.2 8.3 4.7 6.0 6.8 5.7 6.5 4.6 5.4 3.7 7.1 5.5 6.0 6.7 7.0 7.3 3.0 6.6 6.1
[39] 2.4 7.1 9.4 3.7 4.5 5.1 5.9 4.7 5.5 8.9 8.1 8.3 4.3 7.1 9.3 5.1 6.1 3.0 5.7
[58] 6.8 3.1 7.7 7.3 7.0 6.2 8.8 5.3 4.0
> |
```