



**EDUCACIÓN**  
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO®

TECNOLÓGICO NACIONAL DE MÉXICO

INSTITUTO TECNOLÓGICO DE TIJUANA

SUBDIRECCIÓN ACADÉMICA

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN

NOMBRE DE LOS ALUMNOS:

RAYMUNDO HIRALES LAZARENO (N. CONTROL: 17212339)

GALAVIZ LONA OSCAR EDUARDO (N.CONTROL: 17212993)

Carrera: Ingeniería Informática

Semestre: 9no

MATERIA: Minería de datos

PROFESOR: JOSE CHRISTIAN ROMERO HERNANDEZ

TRABAJOS: Practica 2

FECHA: 21/09/21

```
# Hiraless Lazareno Raymundo - 17212339
# Galaviz Lona Oscar Eduardo - 17212993
#
#1 suma
suma <- function(x, y) { #funcion de la suma
  resultado <- x + y      #operacion de la suma
  return(resultado)
}
suma (x=4, y=8) #se asignan valores dentro de la formula para funcion
#
#2 Suma con while
fun1 <- function() {
  num <- runif(1) #asignacion de valores
  veces <- 1
  while (sum(num) < 3) {#condicion del while mientras el valor de la suma sea
menor a 3
    veces <- veces + 1 # realiza una suma de veces + 1
    num[veces] <- runif(1) #se le asigna el valor a num
  }
  return(veces) #regresa veces
}

fun1() # primera prueba
#
#3 punto medio
medio <- function(a, b) {
  medio <- (a + b) / 2 # operacion para calcular el punto medio
  cat("El punto medio de los valores", a, "y", b, #se expresa en mensaje de
resultado
    "ingresados es", medio)
}

medio(a=-3, b=-1) # Probando la función
#
#4 temperatura
fahr_to_kelvin<- function(temp){
  kelvin <-((temp-32)*(5/9)) +273.15 #Formula de la temperatura en kelvin
  return(kelvin)
}
fahr_to_kelvin(temp = 15) # se realiza la funcion asignadole un valor
#
#5 stop if
fahr_to_kelvin <- function(temp) { #funcion de temperatura
  if (!is.numeric(temp)) { #se condiciona si la temperatura es un numero
    stop("temp must be a numeric vector.") #si se cumple se detendra
  }
  kelvin <- ((temp - 32) * (5 / 9)) + 273.15 #Formula para calcular la
temperatura
  return(kelvin) #regresa el de la respuesta
}
fahr_to_kelvin(temp =120) # sele asigna un valor para poder usar la funcion
#
```

```

#6 Operacion de formula
an <- function(a1, r, n) { #se crea la funcion
  a1 * r ** (n - 1)      #se realiza la operacion
}
#se asignan los valores
an(a1 = 1, r = 2, n = 5) # 16
an(a1 = 4, r = -2, n = 6) # -128
#
#7 funcion return
asn <- function(a1 = 1, r = 2, n = 5) {
  A <- an(a1, r, n)
  S <- sn(a1, r, n)
  ii <- 1:n
  AA <- an(a1, r, ii)
  SS <- sn(a1, r, ii)
  return(list(an = A, sn = S,
              salida = data.frame(valores = AA,
                                   sum = SS)))
}
asn(2)
#
#8 variables locales y globales
fun <- function() { #funcion que solo imprime el valor de la variable
  print(x)
}

x <- 1 # se asigna un valor a la variable

fun() # 1

x <- 1 #asignacion de valores
y <- 3
fun3 <- function() { #funcion que permite cambiar las variables
  x <- 2 #se asignan nuevo valores
  y <- 5
  print(paste(x, y)) #se imprimen
}

fun3() # 2 5
x # 1 (el valor no cambi6)
y # 5 (el valor cambi6)
#
#9 Porcentaje
DNI <- function(numero) { #se crea una lista del DNI
  letras <- c("T", "R", "W", "A", "G", "M", "Y", "F", "P", "D", "X", "B",
             "N", "J", "Z", "S", "Q", "V", "H", "L", "C", "K", "E")
  letras <- letras[numero %% 23 + 1] # se realiza la operacion para buscar el
  porcentaje
  return(letras) #regrasa una letra
}
DNI(50247828) # G
#
#10 probabilidad con graficas
dado <- function(n = 100){ #funcion de lanzar dados

```

```

lanzamientos <- sample(1:6, n, rep = T) #valor del lanzamiento del dado
frecuencias <- table(lanzamientos)/n #frecuencia de lanzamientos
barplot(frecuencias, main = "")      #lo representa graficamente
abline(h = 1/6, col = "red", lwd = 2) #representacion de la linea roja
return(frecuencias) #regresa porcentaje de veces del numero
}
par(mfcol = c(1, 3))

dado(100) #cantidad de dados a tirar
dado(500)
dado(100000)
#
#11 rnorm numero aleatorios
rnorm(3:15, 10) #los primeros numeros son el rango y el segundo es la cantidad
#
#12 slappy
cuadrado.raro <- function(x) if(x < 5) x^2 else -x^2 #funcion de un cuadrado
sapply(1:10, cuadrado.raro) # interacion con una lista o vector
#
#13 raiz cuadrada
sqrt(1:10) #se saca raiz cuadrada a un numero
#
#14 reduce
Reduce(function(a, b) a + b, 1:10) #reduce la lista a un unico valor
#
#15 filtro
x <- 1:20 #asignacion del valor de la lista
x[x %% 3 == 0] # nos regresa los valores acorde a la condicion en este de 3 en 3
Filter(function(i) i %% 3 == 0,) #lo mismo que el anterior pero con la funcion
filter
#
#16 concatenacion
x <- c(1,2,3) #se asignan valores
t <- c("uno","dos","tres")
x; t #se combinan la informacion de las dos variables
#
#17 CAT
x <- 2 #asignacion de valores
y <- 4
cat(x,"elevado a",y,"es",x ^ y,"\n") #concatena e imprime el resultado
#
#18 Invisible
#funcion que vuelve invisible a la variable pero que los datos
#se encuentran dentro
suma <- function(a, b) {
  s <- a + b
  return(invisible(s)) #regresa la variable pero invisible para el usuario
}

suma(5,4)
x <- suma(5,4)
x
#
#19 Trunc elimina decimales

```

```
#Funcion que elimina decimales de una cantidad de una variable
x <- 56.13 #se asigna la cantidad a una variable
trunc(x) #elimina los decimales de esa variable
#
#20 Round redondeo
#Funcion que redondea los decimales
#el primero numero es la cantidad mientras que el segundo es la cantidad que se
quiere redondear
round(6.78,1)
```