

```
--Definición de funciones propias
--Cabecera
--nom_fun::TipoDato1 -> TipoDato2 -> ... -> TipoDatoN
--Definición propiamente dicha
--nom_fun var1 var2 ... varM = implementación


--Tipos de datos
--Numéricos enteros: Int
--                      Integer

--Numéricos reales: Float
--                      Double

--Booleanos:          Bool

--Character:          Character
--Los caracteres literales se
--delimitan entre comillas simples.
--Ejemplos: 'a', '0', '?', '*', 'A', ' '

--Cadenas de caracteres:      String
--Las cadenas literales se delimitan entre comillas
--dobles.
--Ejemplos: "PPR", "012", "A", "0", "cadena", "", " "


--Implementar una función que permita sumar
--dos números enteros
suma::Int -> Int -> Int --cabecera
suma n1 n2 = n1 + n2


cuadrado::Int -> Int
cuadrado x = x^2


--definición parcializada
menor::Int -> Int -> Int
menor n1 n2 = if n1<n2 then n1 else n2


--definición no parcializada
mayor::(Int,Int) -> Int
mayor(n1,n2)=if n1>n2 then n1 else n2


es_par::Int -> Bool
es_par n = mod n 2 == 0


par_impar::Int -> String
par_impar n = if es_par n then "par" else "impar"


(++)::Int -> Int
(++ ) x = x + 1


--utilizando if-then-else ANIDADOS
evaluacion1::Int -> String
evaluacion1 x = if x>0 then "mayor que 0" else (if x<0 then "menor que cero" else "igual que cero")


evaluacion2::Int -> String
evaluacion2 x | (x^2 + 5*x + 0) > 0 = "mayor que cero"
              | (x^2 + 5*x + 0) < 0 = "menor que cero"
              | (x^2 + 5*x + 0) == 0 = "igual que cero"


evaluacion3::Int -> String
evaluacion3 x | (x^2 + 5*x + 0) > 0 = "mayor que cero"
              | (x^2 + 5*x + 0) < 0 = "menor que cero"
              | otherwise = "igual que cero"


evaluacion4::Int -> String
evaluacion4 x | f > 0 = "mayor que cero"
              | f < 0 = "menor que cero"
              | otherwise = "igual que cero"
              where f=x^2 + 5*x + 0


dia::Int -> String
dia n = case n of
    0 -> "domingo"
    1 -> "lunes"
    2 -> "martes"
    3 -> "miercoles"
    4 -> "jueves"
    5 -> "viernes"
    6 -> "sabado"
    otherwise -> "incorrecto"
```