Hello world

main = putStrLn "Hello World"

data.char in Haskell

Predicates over characters :

Here are some of the exported predicates over characters (functions of type Char -> Bool):

isSpace isSpace checks whether a character is a white-space character; that includes spaces, tab characters, newlines etc.

isLower isLower checks whether a character is a lower-case character

isUpper isUpper checks whether a character is an upper-case character

isAlpha isAlpha checks whether a character is a letter

isAlphaNum isAlphaNum checks whether a character is a letter or a number

isDigit isDigit checks whether a character is a digit

isLetter isLetter checks whether a character is a letter

isNumber isNumber checks whether a character is numeric

isPunctuation isPunctuation checks whether a character is a punctuation

isSymbol isSymbol checks whether a character is a fancy mathematical or currency symbol

Basic operators

main = do

let var1 = 2

let var2 = 3

putStrLn "The addition of the two numbers is:"

print(var1 + var2)

main = do

let var1 = 10

let var2 = 6

putStrLn "The Subtraction of the two numbers is:"

print(var1 - var2)

main = do

let var1 = 2

let var2 = 3

putStrLn "The Multiplication of the Two Numbers is:"

print(var1 \* var2)

main = do

let var1 = 12

let var2 = 3

putStrLn "The Division of the Two Numbers is:"

print(var1/var2)

Sequence / Range Operator

main :: IO()

main = do

print [1..10]

Control structures

main = do

let var = 23

if var `rem` 2 == 0

then putStrLn "Number is Even"

else putStrLn "Number is Odd"

head function

main = do

let x = [1..10]

putStrLn "Our list is:"

print (x)

putStrLn "The first element of the list is:"

print (head x)

tail function

main = do

let x = [1..10]

putStrLn "Our list is:"

print (x)

putStrLn "The tail of our list is:"

print (tail x)