**Introducción a Ruby**

Código

|  |
| --- |
| def format\_number(number,dec)  precision = 0  if dec > 0  precision = dec  end  puts "NUMERO INICIAL #{number}"  if dec > 0 then  number = number.round(precision)  puts "truncado de numero en #{number}"  end  todo, decimal = number.to\_s.split(".")  strdec = decimal.to\_s  puts "ENTERO = #{todo} DECIMAL = #{strdec} PRECISION = #{dec}"  for i in(strdec.length..dec-1)  strdec += "0"  end  num\_groups = todo.chars.to\_a.reverse.each\_slice(3)  todo\_y\_coma = num\_groups.map(&:join).join(',').reverse  if precision > 0 then  return [todo\_y\_coma, strdec].compact.join(".")  else  return todo\_y\_coma    end  end  #numero = 246574.256654  numero = 32432  #strnum = numero.round(2)  #puts "el numero es " + strnum.to\_s + "- "  puts format\_number(numero,2) |

**Descripción**

Hoy me dediqué a hacer una función que le de formato a un número determinado. Me ha sido de mucha utilidad para trabajar más adelante mostrando los resultados calculados.

**Reflexión**

Se me está haciendo muy difícil escribir la evidencia, precisamente porque las cosas básicas del lenguaje ya están aprendidas, y el resto es práctica. No se me ocurre cómo expresar en 80 palabras un pequeño descubrimiento que impacta muy fuertemente mi forma de programar.