Haskell:

- Type System, Lazy Evaluation, Pure Functional

Y = 1 :: Int

- We can declare the type of symbol or expression

Type Sinature:

- NameFunction :: Integer -> Integer

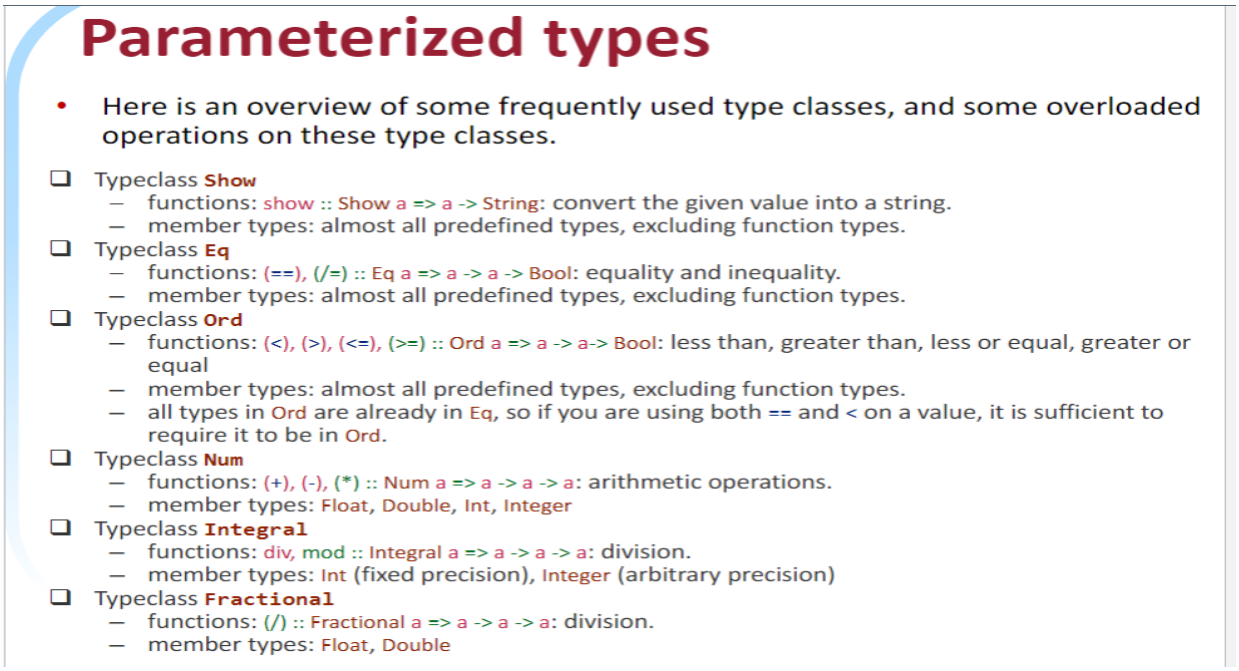
Functions of multiple arguments that can be applied to their arguments one at a time are called **curried functions**

- average :: Float-> Float-> Float
- average 3.0 4.0 is equivalent to (average 3.0) 4.0

Infix notation vs Prefix:

- 3.0 `average ` 4.0 = average 3.0 4.0
- (+) 3 4 = 3+4

Lazy – definition of symbols are evaluated when needed



Tuples can have many types

["red", "green", "blue"] : "yellow"⇒Error!

++ list append, !! get element at index, head [list] = get head