**Eval:**

Eval is a data type for controlling synchronous evaluation and is stack-safe at all times (using a technique called trampolining).

There are two different factors that play into evaluation: memoization and laziness:  
 - Memoized evaluation evaluates an expression only once and then remembers (memoizes) that value.  
 - Lazy / eager evaluation refers to when the expression is evaluated. Eager evaluation is when expression  
 is immediately evaluated when defined and lazy evaluation is when the expression is evaluated when  
 it's first used (not when it is defined).

A normal val evaluates eagerly and also memoizes the result.  
A lazy val is both lazy and memorized  
A method/function def is lazy, but not memoized, since the body will be evaluated on every call.

Eval is able to express all of these evaluation strategies and allows us to chain computations using its Monad instance.