Hello, Haskell!

Redexes are reducible expressions. (Also called "*normalizing*" or "executing" an expression)
Functions are a specific type of expression. Currying is applying a series of nested functions.

1 Evaluation

Evaluating an expression is reducing the terms until the expression reaaches its simplest form.

2 Associativity and precedence

Higher number, higher precedence. Associativity: left or right.
Use space, not tab.
Error: indent, not starting a declaration at the beggining (left) column of the line.
The first declaration in a module defines the remaining ones.
(quot x y) * y + (rem x y) == x (div x y) * y + (mod x y) == x
mod has the same sign as the divisor. rem: dividend
sectioning (+1)

3 Let and where

Let introduces an expression, where is a declaration.
Scope is the area of source code where a binding of a variable applies.
Write code in a source file: the order is unimportant, but when writing code directly into the REPL, it does matter.
Infix: place between.

Definition

Syntactic sugar is syntax within a programming language designed to make expressions easier to write or read.

Types are way of categorizing values.
:: has the type (type signature)

Definition : Typeclasses

Typeclasses provide definitions of operations, or functions, that can be shared across sets of types.

(:) cons

Definition : Scope

Scope is where a variable referred to by name is valid. Another word used with the same meaning is *visibility*, because if a variable isn't visible, it's not in scope

Definition : Local and Top level bindings

- Local bindings are bindings local to particular expressions. Cannot be imported by other programs or modules.
- Top level bindings are bindings that stand outside of any other declaration. Can be made available to other modules within your programs or to ther ppl.

Definition : Data Structures

Data structures are way of organizing data so that the data can be accessed conveniently or efficiently.