

Let Expressions

Let expressions allows us to introduce the concept of local variables. Let expressions can be more efficient, its important to use them when they are needed.

Syntax: uses 3 keywords `let`, `in`, `end` Between the let and the in you can put any number of bindings.

- `let b1 b2 b3 ... bn in e end`
- each `bi` is any binding and `e` is any expression

Type-Checking: Type check each `bi` and `e` in a static environment that includes the previous binding the type of the `let` expression type is the type of `e`

Evaluation: Evaluate each (binding) `bi` in order and `e` in a dynamic environment, the bindings are available to each binding that follows it. All bindings are available in `e` the body. The result of the whole let expression is the result of evaluating `e`

Scoping

Let expressions introduces the idea of scopes, using let expressions allows you to define where a binding is in the environment. Anywhere an expression can be used a new scope aka a let expression can be used.