(* The start of a nix expression *)
Expression →

[Preamble] (*ExpressionNoAssignment* | *Assignment*)

ExpressionNoAssignment →

(*Attrset* | *Function* | *Arithmetic*)

(* A let .. in or with *)
Preamble →
  { (*let-in* | *with*) }⁺,

(* Let in *)
(* (* `let a = 5; in ...` *) *)
let-in →
  `let` { (*Assignment* | *Inherit*) }⁺ `in`

with →
  `with`

(*Attrset* | ? Identifier of Attrset ?) `;`

Inherit →
  `inherit` [? Identifier of Attrset ? | *Attrset*] { ? Identifier ? }⁺ `;`

(* Expression of an attrset *)
Attrset →
  `{` { (*Assignment* | *Inherit*) }⁺ `}`

(* An assignment without the preamble *)
(* (* `a = 5;` *) *)
Assignment →
  ? someIdentifier ? `=` *ExpressionNoAssignment* `;`

(* A string, integer or float *)
Primary →
  (? string ? | ? integer ? | ? float ? | `true` | `false` | *List* | (`(` *ExpressionNoAssignment* `)`))

List →
  `[` { ExpressionNoAssignment } `]`

(* A function *)
(* (* `input: output` *), *)
Function →
  ? InputIdentifier ? `:` [Preamble] *ExpressionNoAssignment* `;`

Arithmetic →
  ArithmeticMul { (`-` | `+`) *ArithmeticMul* }

ArithmeticMul →
  PrimaryOrIdentifier { (`*` | `/`) *PrimaryOrIdentifier* }

PrimaryOrIdentifier →
  (*Primary* | *Identifier*)