Concrete Syntax of Hydra

Program -> **{** Statement **}**

Function -> Identifier : Parameters -> Type **(** Block **|** = Expression **[** LocalFs **])**
LocalFs -> where Function **{** Function **}**
Parameters -> (Declaration **{**, Declaration**}**)
Block -> { **{** Statement **}** }

Statement -> ; | Block | Declaration | Assignment | Conditional | Loop | Function
Assignment -> Identifier := Expression ;
Conditional -> if (Expression) Statement **[** else Statement **]**
Loop -> while (Expression) Statement
Expression -> Term **{** BinOp Term **}**
Op -> + **|** - **|** * **|** / **|** % **|** **
Term -> **[** UnOp **]** Primary
UnOp -> - **|** !
Primary -> (Expression) **|** Type (Expression) **|** Literal **|** Accessor **|** Identifier (Arguments)
Arguments -> **[** Expression **{**, Expression **}]**

Declaration -> Identifier : Type

Type -> Int **|** Bool **|** Float **|** Char
Identifier -> Letter **{** Letter **|** Digit **}**
Letter -> a **|** b **|** ... **|** z **|** A **|** B **|** ... **|** Z
Digit -> 0 **|** 1 **|** ... **|** 9
Literal -> Integer **|** Boolean **|** Floating **|** Character
Integer -> Digit **{** Digit **}**
Boolean -> True **|** False
Floating -> Integer . Integer
Character -> ' ASCIIChar '