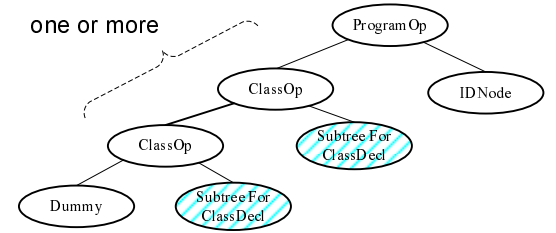
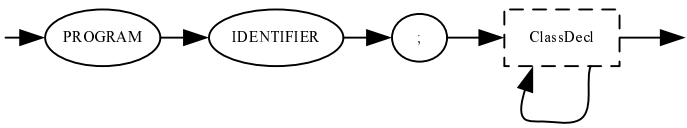
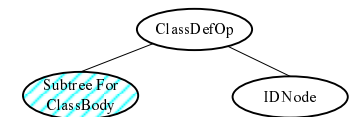
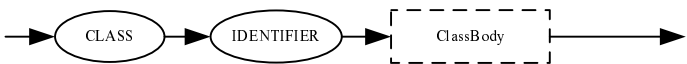


Legend:	dashed boxes → nonterminal symbols	eclipse → normal nodes
	solid ellipsis → terminal symbols (tokens)	shaded eclipse → subtree

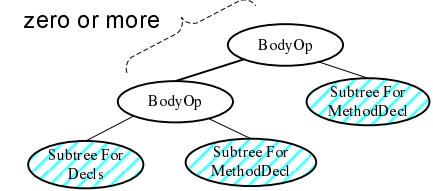
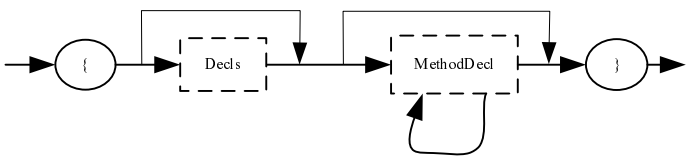
## Program



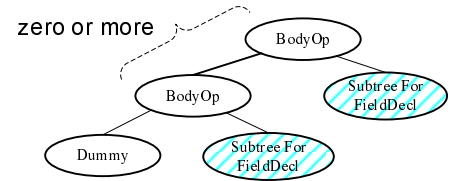
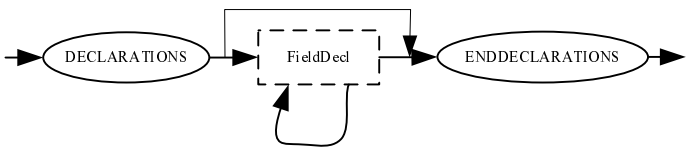
## ClassDecl



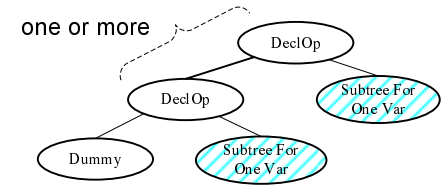
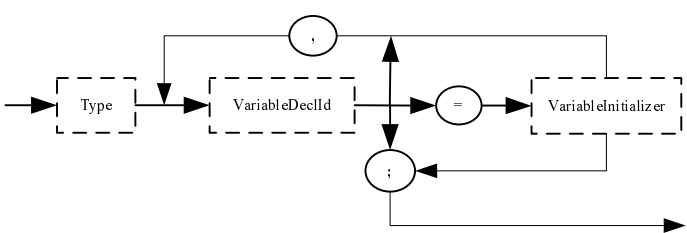
## ClassBody



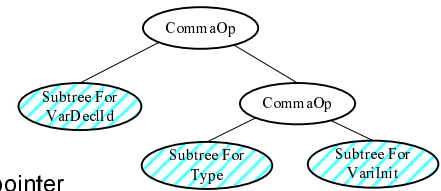
## Decls



## FieldDecl

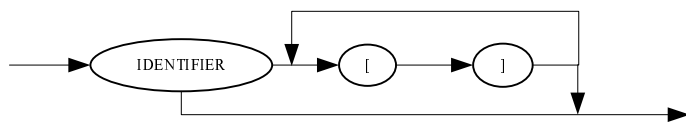


Each Var has the following subtree

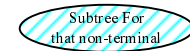
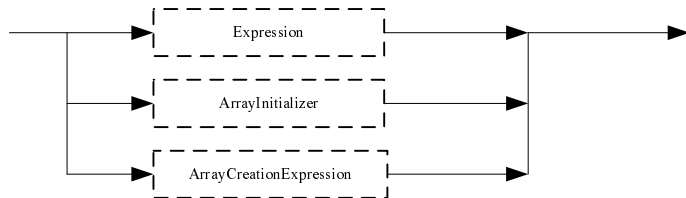


Type should be stored in a separate pointer (global variable) such that it may be used in building the VariableInitializer subtree.

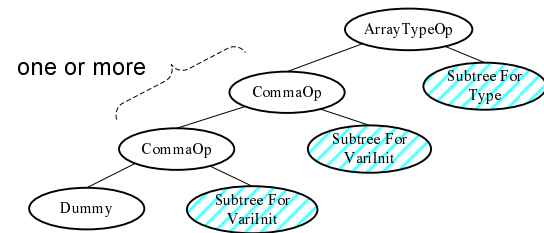
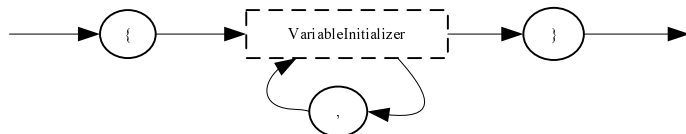
## VariableDeclId



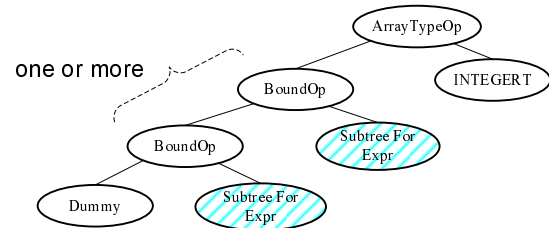
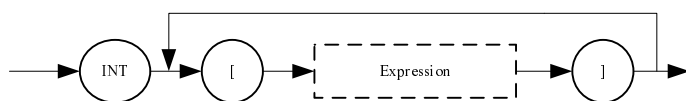
## VariableInitializer



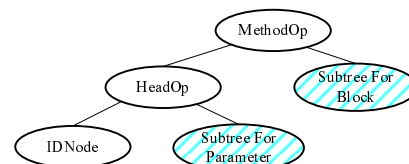
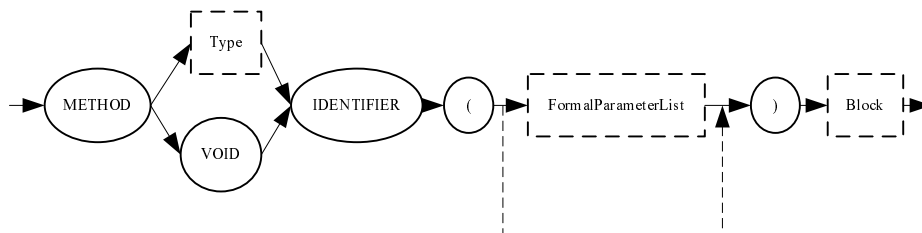
## ArrayInitializer



## ArrayCreationExpression

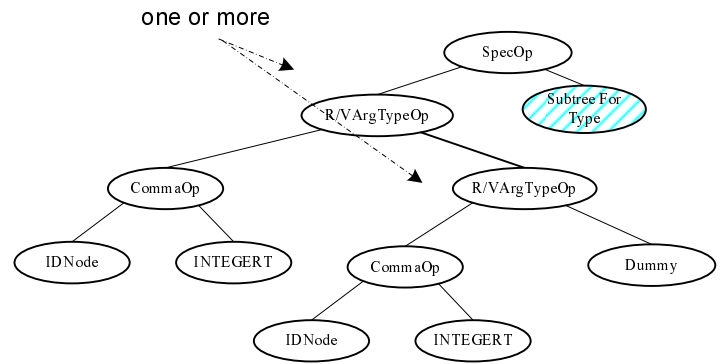
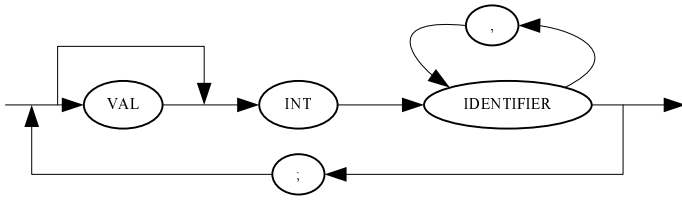


## MethodDecl

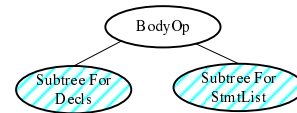
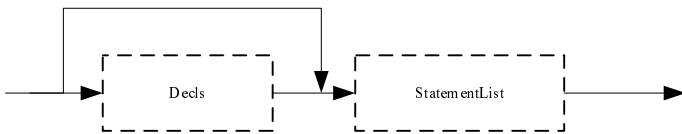


Type should be stored in a separate pointer (global variable) such that it may be used in building the *Parameter* and *Block* subtrees.

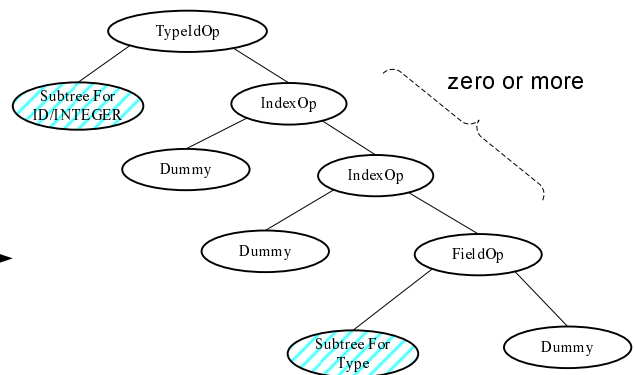
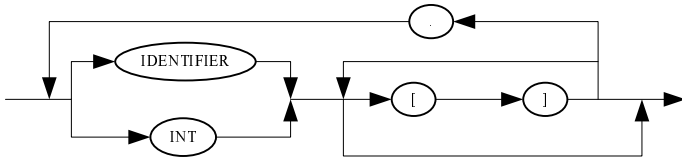
## FormalParameterList



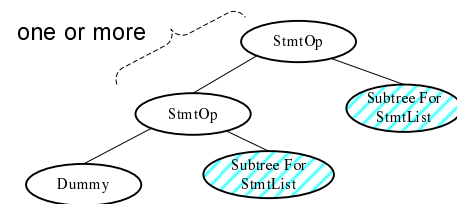
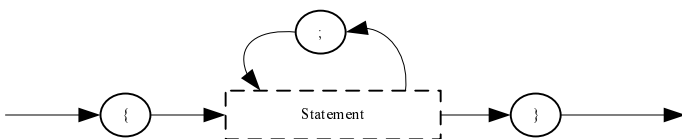
## Block



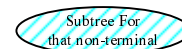
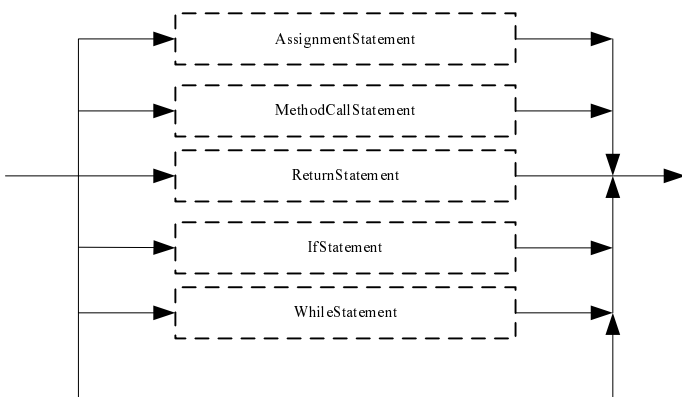
## Type



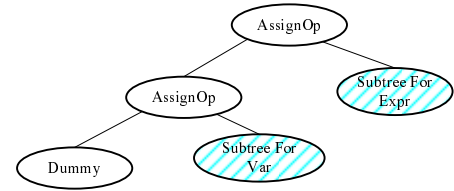
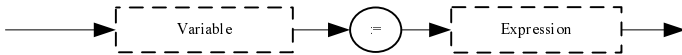
## StatementList



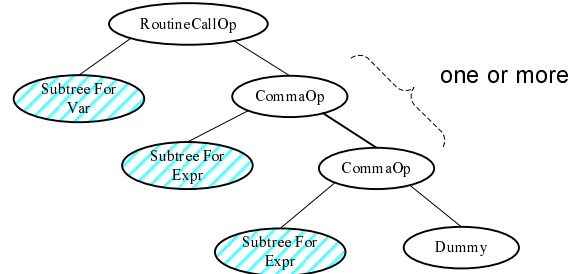
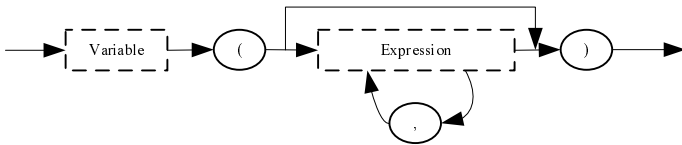
## Statement



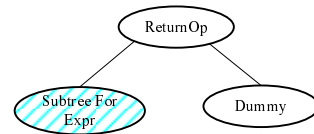
## AssignmentStatement



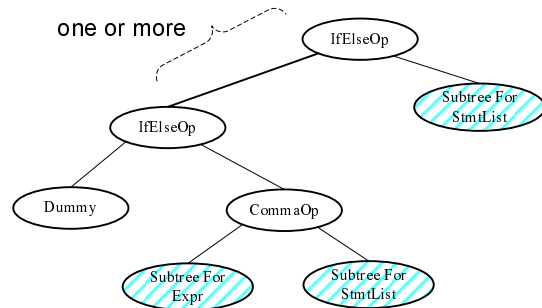
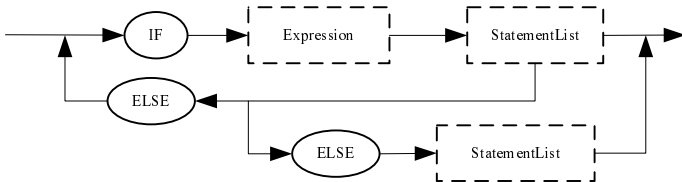
## MethodCallStatement



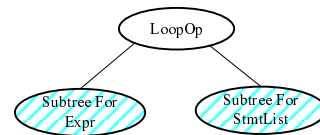
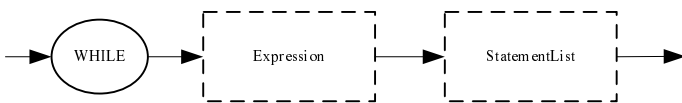
## ReturnStatement



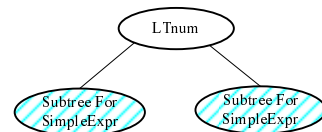
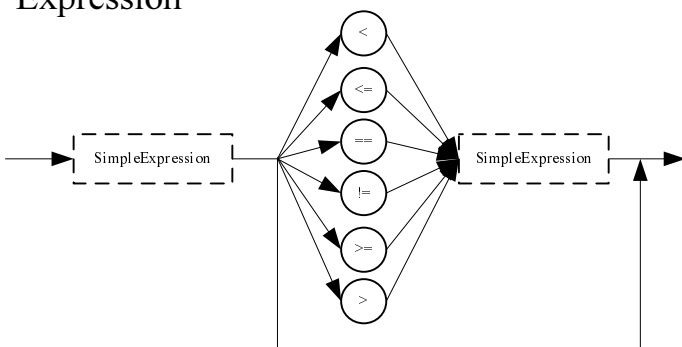
## IfStatement



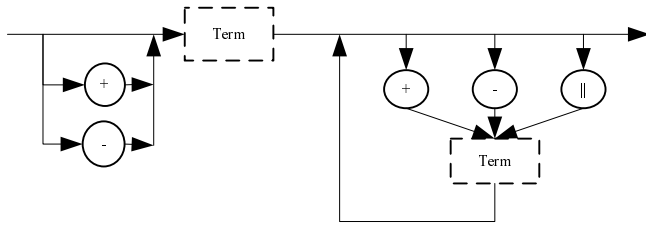
## WhileStatement



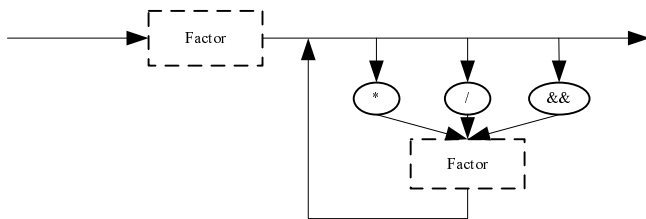
## Expression



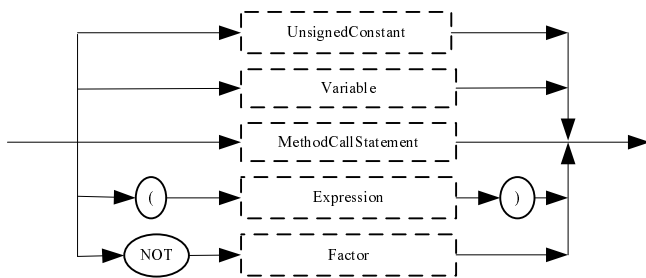
## SimpleExpression



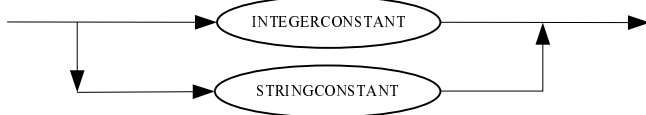
## Term



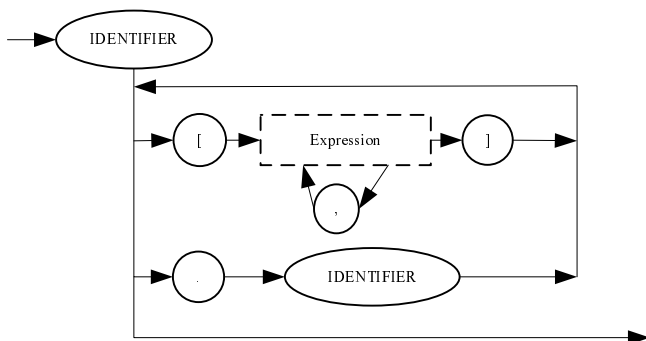
## Factor



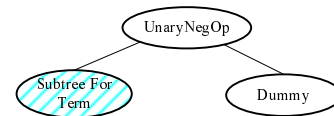
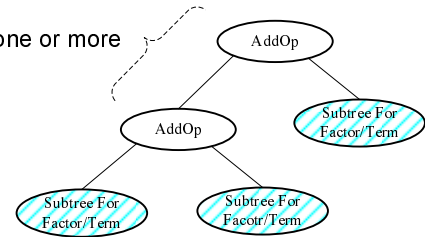
## UnsignedConstant



## Variable



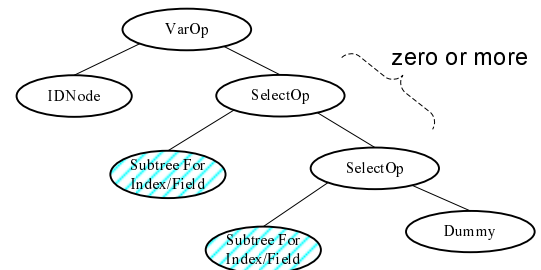
one or more



Subtree For that non-terminal

Subtree For that non-terminal

zero or more



one or more

