**Compilers Phase2 Part1**

- Type::= "int" TypeDash  
| "float" TypeDash  
| "String" TypeDash  
| "char" TypeDash  
| "boolean" TypeDash

* TypeDash ::= "[" "]"| lamda

- Statement ::= "{" ( Statement )\* "}"  
| "if" "(" Expression ")" Statement StatementDash   
| "while" "(" Expression ")" Statement  
| "System.out.println" "(" Expression ")" ";"  
| Identifier StatementIdentifierDash

* StatementDash ::= "else" Statement | lamda
* StatementIdentifierDash ::= "=" Expression ";" | "[" Expression "]" "=" Expression ";"

- Expression ::=

<INTEGER\_LITERAL> ExpressionDash  
| <FLOAT\_LITERAL> ExpressionDash  
| "true" ExpressionDash

| "false" ExpressionDash  
| Identifier ExpressionDash  
| "this" ExpressionDash  
| "new" ExpressionNewDash  
| "!" Expression ExpressionDash  
| "(" Expression ")" ExpressionDash

* ExpressionDash ::=

( "&&" | "||" | "==" | "!=" | ">" | "<" |"<=" | ">="| "+" | "-" | "\*" | "/" ) Expression ExpressionDash

|"[" Expression "]" ExpressionDash

| "." ExpressionDotDash

| lamda

* ExpressionNewDash ::= ("int" | "float" | "String" | "char" | "boolean") "[" Expression "]" ExpressionDash | Identifier "(" (Expression ( "," Expression)\*)? ")" ExpressionDash
* ExpressionDotDash ::= "length" ExpressionDash | Identifier "(" ( Expression ( "," Expression )\* )? ")" ExpressionDash