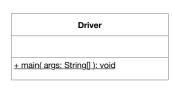Responsibilities:
1. Evaluation thing
    - order of operations/precedence is important
    - Needs to return an answer
2. ~~Input reader/processor~~
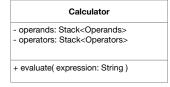   - ~~turn a string into numbers and operators~~
3. Output

Data:
- Operands (which start as strings)
- Operators

Driver
+ main(args: String[]): void

This is responsible for breaking the expression string up into individual useful units
(# means end of token stream)

ExpressionReader
- expression: String - index: int
+ ExpressionReader(expression: String) + nextToken(): String

" 1 + 3 + 2 * 5 "   => | 1 | + | ... |

Calculator
- operands: Stack<Operands> - operators: Stack<Operators>
+ evaluate(expression: String)

Operand
- value: int
+ Operand(token: String) + Operand(value: int) + getValue(): int

<> Operator
+ Map<String, Operator> validOperators
+ execute(operandOne: Operand, operandTwo: Operand): Operand + getPrecedence(): int

Add
+ execut... + getPre...

Multiply
+ execute(... + getPrece...

Divide
+ execute(ope... + getPreceden...

Subtract
+ execute(operandOne: Operand, operandTwo: Operand): Operand + getPrecedence(): int