Building an Excel-like formula interpreter using ANTLR

http://tiny.cc/hsformula
http://tiny.cc/hsformulaslides

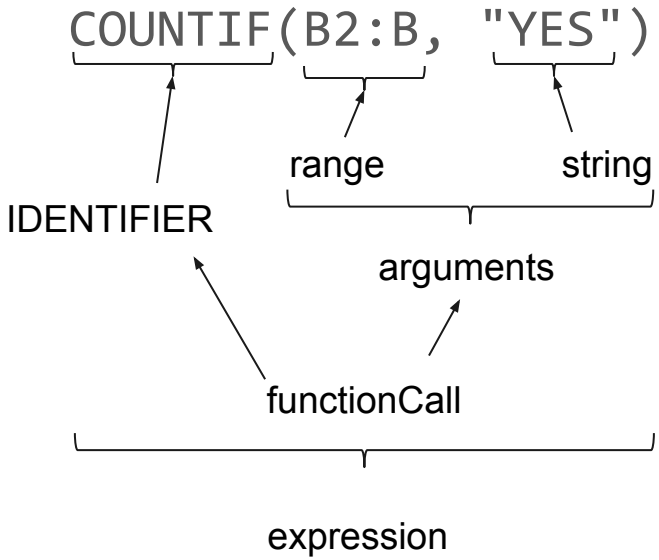
HSFormula steps

Parsing

```
COUNTIF(B2:B, "YES")
```

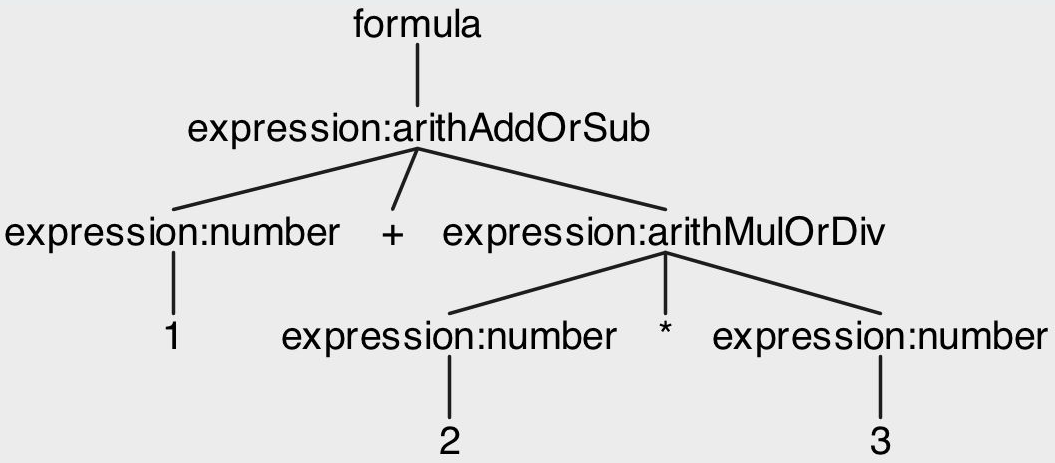
```
COUNTIF    '(' B2   ':' B    ',' "YES"  ')'
IDENTIFIER '(' CELL ':' CELL ',' STRING ')'
```

Parsing


COUNTIF(B2:B, "YES")
range
string
IDENTIFIER
arguments
functionCall
expression

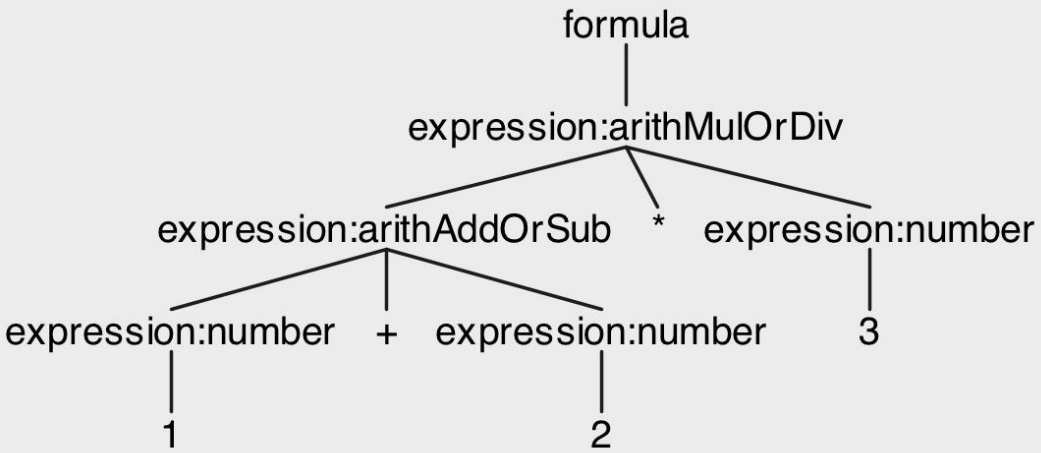
Parsing - precedence

```
expression
    : left=expression op=('*'|'/') right=expression # arithMulOrDiv
    | left=expression op=('+'|'-') right=expression # arithAddOrSub
```



Parsing - precedence

```
expression
    : left=expression op=('+'|'-') right=expression # arithAddOrSub
    | left=expression op=('*'|'/') right=expression # arithMulOrDiv
```



Lexing - precedence

```
CELL
    : [a-zA-Z][0-9]*
    ;

IDENTIFIER
    : [a-zA-Z]+
    ;
```

Lexing - precedence

```
IDENTIFIER
    : [a-zA-Z]+
    ;

CELL
    : [a-zA-Z][0-9]*
    ;
```

B1 : B

CELL ‘:’ IDENTIFIER

ANTLR supports context sensitive lexing[link]

Listener and Visitor

```
public interface HSFormulaListener extends ParseTreeListener {
    void enterFormula(HSFormulaParser.FormulaContext ctx);
    void exitFormula(HSFormulaParser.FormulaContext ctx);
    void enterNumber(HSFormulaParser.NumberContext ctx);
    void exitNumber(HSFormulaParser.NumberContext ctx);
    void enterString(HSFormulaParser.StringContext ctx);
    void exitString(HSFormulaParser.StringContext ctx);
   // ...
```

Listener and Visitor

```
public interface HSFormulaVisitor<T> extends ParseTreeVisitor<T> {
    T visitFormula(HSFormulaParser.FormulaContext ctx);
    T visitNumber(HSFormulaParser.NumberContext ctx);
    T visitString(HSFormulaParser.StringContext ctx);
    T visitFunctionCall(HSFormulaParser.FunctionCallContext ctx);
    // …
```

Analysis

- Function calls must be validated
  - Check if function exists
  - Check argument count
  - Check argument type
  - Propagate function return value
- Arithmetic operators must be validated
- We don’t validate ranges, let it blow up at runtime instead

Evaluation

HSFormula bails on out of range cells or cells that don’t have properly formatted values.

Pretty straightforward - visit and evaluate!

ANTLR resources

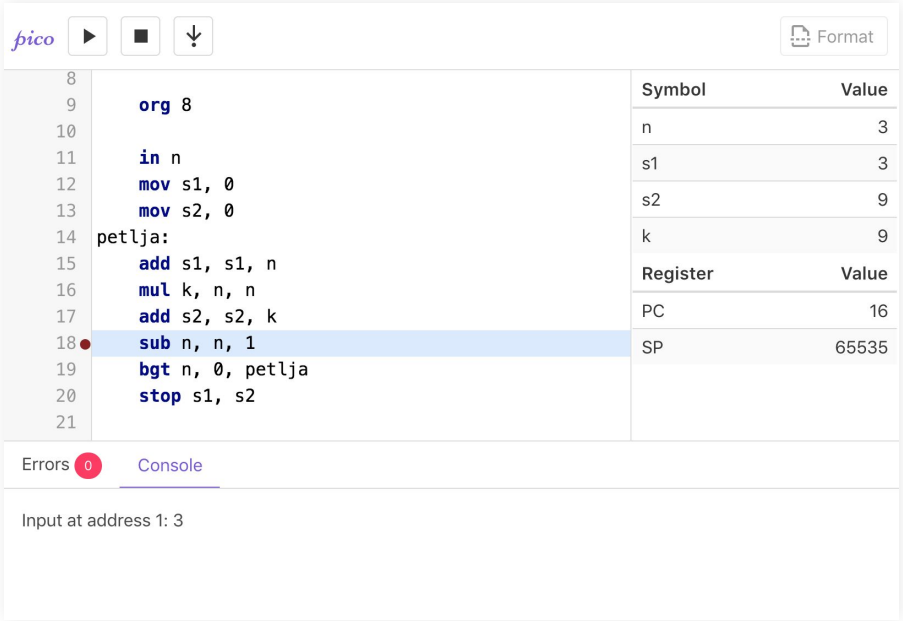
grammars-v4 is a repository of ANTLR4 grammars that you can use (check the license first) or let them serve as inspiration.

https://github.com/antlr/grammars-v4

The Definitive ANTLR 4 Reference

https://pragprog.com/book/tpantlr2/the-definitive-antlr-4-reference

picoComputer Web IDE



Demo
https://nemanjamiljkovic.me/pico/

Parsing, Analysis, Bytecode generation, VM
https://github.com/nmiljkovic/pico-asm

Vue Frontend, CodeMirror editor
https://github.com/nmiljkovic/pico-sim

Q&A

http://tiny.cc/hsformula
http://tiny.cc/hsformulaslides

@nmiljkovic
https://nemanjamiljkovic.me/