

Step2 : Building The Parser



- We will use ANTLR to automatically obtain the code of the parser
- To complete step 2, you need to:
 1. Use the grammar file (Micro.g4) to write the full grammar of Micro (the language grammar is available on the course webpage)
 2. Use ANTLR to obtain the parser code ([MicroParser.java](#))
 3. Integrate MicroParser.java with your code project
 4. Modify [Driver.java](#) to invoke the parser and check for syntax errors



Micro.g4

grammar Micro;

KEYWORD:

.....

WS : [\t\r\n]+ -> skip ;



These are the regular
expressions that you wrote in
step 1

program: 'PROGRAM' id 'BEGIN' pgm_body 'END' ;

id: IDENTIFIER ;

....



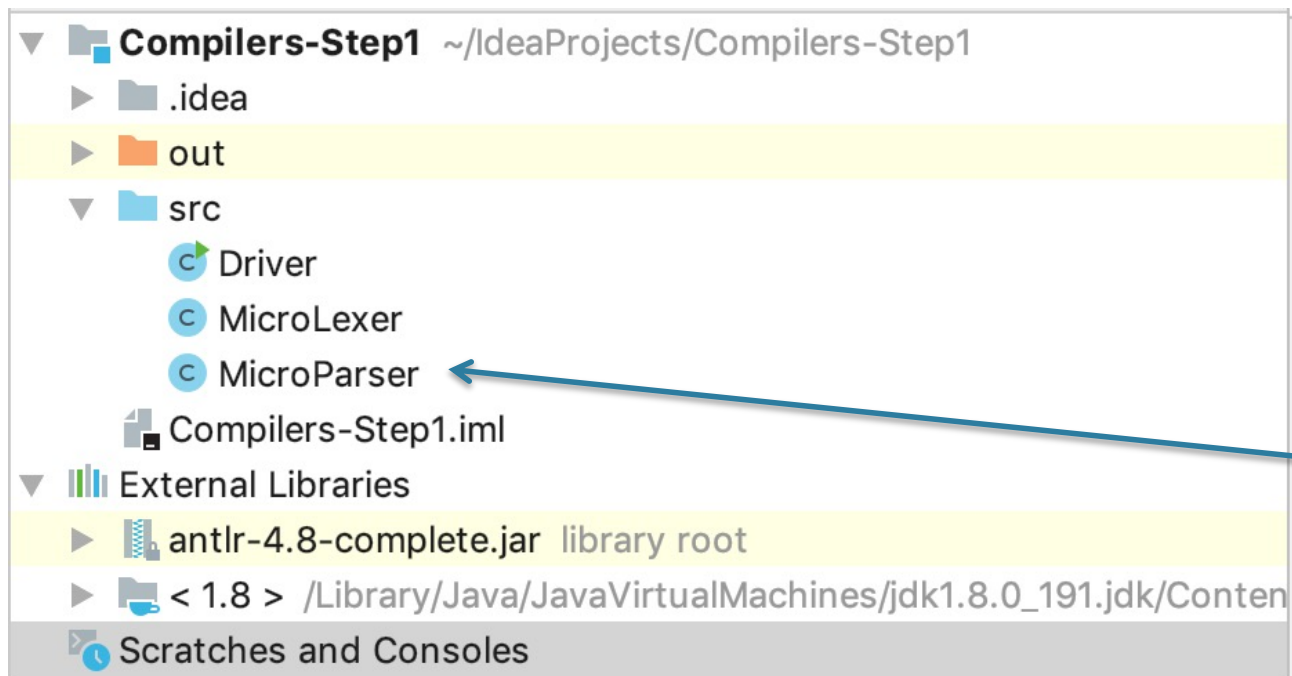
Now, write the
grammar rules



Using ANTLR Tool

- Use the following command
`java -jar antlr.jar -no-listener Micro.g4`
- The scanner code is found in [MicroLexer.java](#)
- The parser code is found in [MicroParser.java](#)

Integrating Parser Code



Add the parser code
to the project you
already created in
step1

Driver.java



```
import org.antlr.v4.runtime.CharStream;
import org.antlr.v4.runtime.CharStreams;
import org.antlr.v4.runtime.CommonTokenStream;
import java.io.IOException;
```

```
public class Driver {
```

```
    public static void main(String[] args) {
```

```
        MicroLexer lexer = null;
```

```
        try{
```

```
            String srcFile = args[0];
```

```
            CharStream charStream = CharStreams.fromFileName(srcFile);
```

```
            lexer = new MicroLexer(charStream);
```

```
        }
```

```
        catch(IOException e){
```

```
            System.out.println("Error: incorrect input file");
```

```
            System.exit( status: 0);
```

```
        }
```

```
        MicroParser parser = new MicroParser(new CommonTokenStream(lexer));
```

```
        parser.program();
```

```
        if(parser.getNumberOfSyntaxErrors() == 0)
```

```
            System.out.println("accepted");
```

```
        else
```

```
            System.out.println("not accepted");
```

```
    }
```

```
}
```

This code
invokes the
parser

This code will check
for syntax errors



Step 2 Output

- By the end of this step, your compiler should be able to take a source Micro code, parse it and produce an “accepted” message if no syntax error is found
 - Otherwise, the compiler produces a “not accepted” message
- I uploaded a set of input *MICRO* code examples on the course website, as well as the expected output for each code example
- Due: One week from today