

JavaCompiler

Prerequisites

- [Java](#)
- [JFlex](#)
- [VSCode](#)
- VSCode's [Java Extension Pack](#)

Run

If you have opened the project in vscode, you can run the project by pressing F5.

► [Example](#)

Structure

File Structure

```
.
├── README.md
├── src
│   ├── Main.java:   JavaCompiler
│   └── Lexer.java:  autoGenerated by JFlex
└── grammar.jflex:  main grammar file
```

Classes

- **Main**

```
java Main [--encoding ENCODING] <file(s)>
```

You have to enter file names manually if no file is specified.

- **Token**

A struct to store token information.

```
public static class Token {
    public String value;
    public TokenType type;
    public int line;
    public int column;

    ...
}
```

- **TokenType**

An enum to store token type.

```
public enum TokenType {
    NUMBER,
    STRING,
    RESERVED,
    IDENTIFIER,
    SYMBOL,
    KEYWORD,
    OPERATOR,
}
```

TokenType	Regex
-----------	-------

Examples

TokenType	Regex	Examples
Number	<code>\d+\.\d*\ \ \.\d+</code>	<code>12, 12.3, 2., .5</code>
String	<code>"((\\. [^\r\n])*)" '((\\. [^\r\n])*)'</code>	<code>"Hello World!", 'Hello World!'</code>
Reserved	<code>compiler\dxt\.init\(\) EOF</code>	<code>compiler, dxt.init(), EOF</code>
Identifier	<code>[a-zA-Z_][a-zA-Z0-9_]*</code>	<code>test</code>
Symbol	<code>[\[\]\(\)\{\}\,\;\:\.\@\#\\$\`]</code>	<code>() [] { } . , ; ? : : @ # \ \$ `</code>

► **Keywords**

► **Operators**