

MyPL Language Compiler

Programming language built using a JVM-like interpreter. This language also has the ability to be transpiled to go to improve performance.

[View Demo](#) · [Report Bug](#) · [Request Feature](#)

► Table of Contents

Tests

This project was tested using both `GoVisitorTest.java` and using the files stored within `./examples`.

Built With

- Java
- [Bazel](#)

[\(back to top\)](#)

Usage

To transpile MyPL code into go code run the following:

```
$ bazel-bin/mypl --go [filename]
```

This creates a new file called `[filename].go` that can be then run using

```
$ go run [filename].go
```

You are also able to use the Makefile provided by running the following command for generating and executing MyPL-GO code (leave off the extension for filename):

```
$ make gorun FILE=[filename]
```

```
$ make run FILE=[filename]
```

Will run the source code file using the MyPL interpreter.

Roadmap

- ☒ Lexer

- ☒ Parser
- ☒ ASTParser
- ☒ VM
 - ☒ Code Runner
 - ☒ Code Generator
- ☒ Go Transpiler

([back to top](#))

Go Limitations

- Type inferencing is no longer allowed within type declarations when converting to Go Code. This is due to a limitation in the `GoVisitor`. It can be improved, but most likely will not be before the due date of this assignment.