Interpreter

Introduction:

The implemented interpreter has following features:

- Two variable types are supported: Boolean and Integer
- It is a dynamic typed language. There is no need to mention the type of variable.
- It supports following operations:
  - Logical: AND, OR, NOT
  - Arithmetic: +, - , *, /
  - Relational: <, >, ==, ><, <=, >=
- Negative Operator "-" is universal(works for both data types). It can be applied to Boolean or Integer. Depending on the type of the operand, corresponding value is returned.
- Variable names can contain alphanumeric characters. Additionally, they can contain '*' symbol.
- Can show the type of syntax errors.
- Can show the line number of error occurrence.

Novelity:

I referred to a book titled 'Practical Interpreter Construction' by '*Mehmet Emin Coskun*'.
It helped me in:

- Getting the notion and logic behind Compilers and Interpreters.
- Understanding the concept of Tokenizer, Parser and Interpreters
- Using Abstract Syntax Tree for storing and manipulating tokens.
- Understanding the chain of Processes: Tokenizer → Parser → Interpreters.
- Getting a bigger picture of Interpreter Construction.

By understanding the concept, I implemented the working interpreter.

I made certain additions to the concept:

- Accepting variables with names containing '*' (*asterisk*);
- Defining '-' operator for both Boolean and Integers and defining its behavior for both.
- Defining '><' (*Not Equal*) operator .

Running the Application:

There are two ways:

- Tests: There is a file named 'test' in the package, which can be opened and executed. It can verify the expected results with the outcome.
- MainClass: This java source file contains main method through which interpreter can be accessed. It can show the output on screen rather than verifying it.

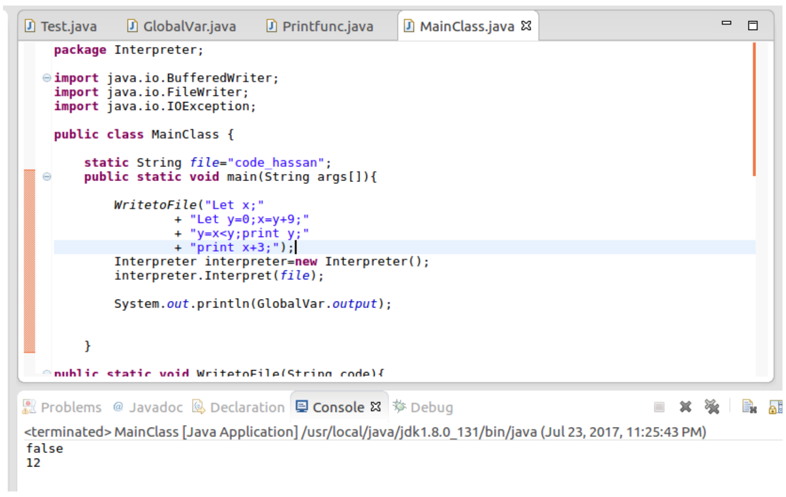
Note: To test the codes with errors, they are put in separate test file. The reason is because they would halt the process of execution if they find a syntax error. Halting the process will terminate the program. In short, Interpreter executes each source code one by one.

GitHub Link: https://github.com/hassan-mahmood/Interpreter

Example Programs:

Simple Program:

```
Let x;
Let y=0;
x=y+9;
y=x<y;
print y;
print x+3;
```

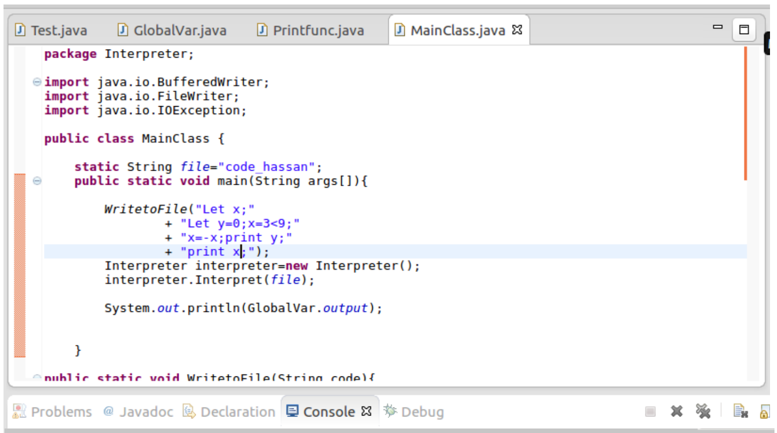


```
package Interpreter;

import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;

public class MainClass {

    static String file="code_hassan";
    public static void main(String args[]){

        WriteToFile("Let x;"
                + "Let y=0;x=y+9;"
                + "y=x<y;print y;"
                + "print x+3;");
        Interpreter interpreter=new Interpreter();
        interpreter.Interpret(file);

        System.out.println(GlobalVar.output);


    }

public static void WritetoFile(String code){
```

```
<terminated> MainClass [Java Application] /usr/local/java/jdk1.8.0_131/bin/java (Jul 23, 2017, 11:25:43 PM)
false
12
```

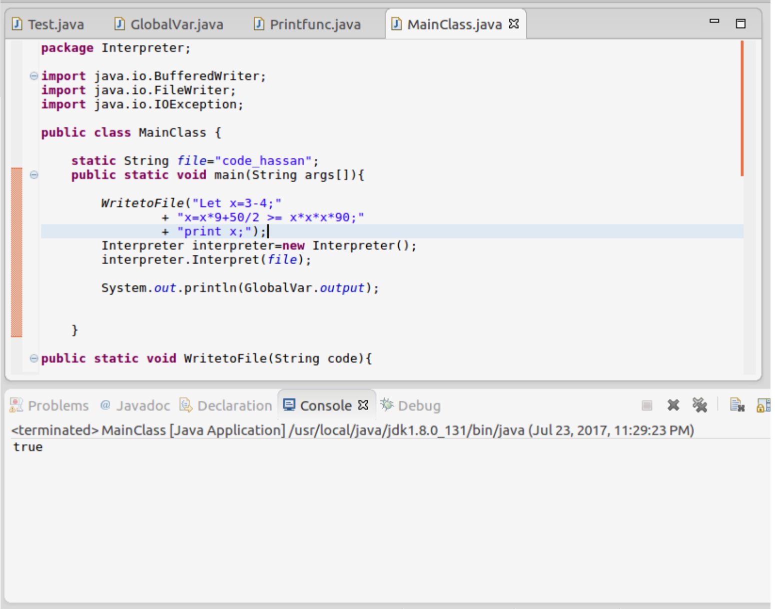
'-' Symbol:

```
Let x;
Let y=0;
x=3<9;
x=-x;
print y;
print x;
```



```
package Interpreter;

import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;

public class MainClass {

    static String file="code_hassan";
    public static void main(String args[]){

        WritetoFile("Let x;"
                + "Let y=0;x=3<9;"
                + "x=-x;print y;"
                + "print x;");
        Interpreter interpreter=new Interpreter();
        interpreter.Interpret(file);

        System.out.println(GlobalVar.output);


    }

public static void WritetoFile(String code){
```

Calculating Operators:



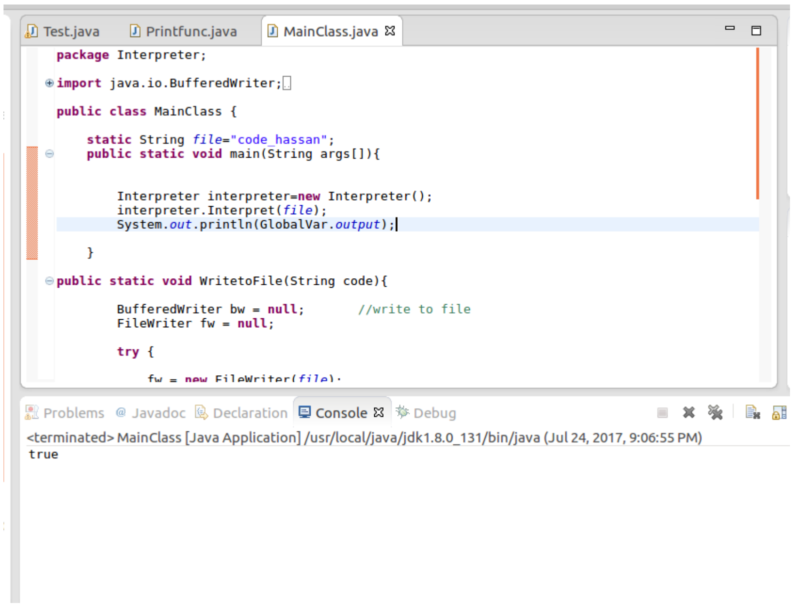
```
package Interpreter;

import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;

public class MainClass {

    static String file="code_hassan";
    public static void main(String args[]){

        WritetoFile("Let x=3-4;"
                + "x=x*9+50/2 >= x*x*x*90;"
                + "print x;");
        Interpreter interpreter=new Interpreter();
        interpreter.Interpret(file);

        System.out.println(GlobalVar.output);


    }

public static void WritetoFile(String code){
```

```
<terminated> MainClass [Java Application] /usr/local/java/jdk1.8.0_131/bin/java (Jul 23, 2017, 11:29:23 PM)
true
```

```
Let x=3-4;
x=x*9+50/2 >= x*x*x*90;
print x;
```

<u>Arithmetic and Relational Op:</u>

```
Let x=3-4;
x=x*9+50/2 >= x*x*x*90"
print x;
```

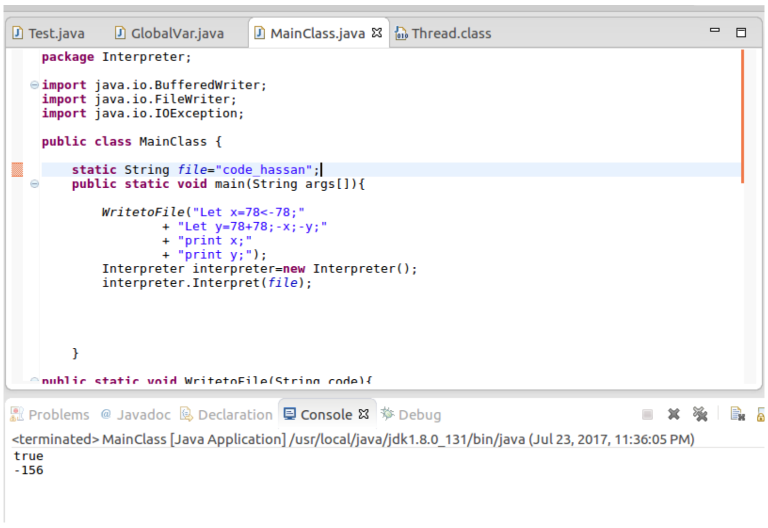


```
package Interpreter;

import java.io.BufferedWriter;

public class MainClass {

    static String file="code_hassan";
    public static void main(String args[]){


        Interpreter interpreter=new Interpreter();
        interpreter.Interpret(file);
        System.out.println(GlobalVar.output);

    }

public static void WritetoFile(String code){

        BufferedWriter bw = null;       //write to file
        FileWriter fw = null;

        try {

            fw = new FileWriter(file);
```

```
<terminated> MainClass [Java Application] /usr/local/java/jdk1.8.0_131/bin/java (Jul 24, 2017, 9:06:55 PM)
true
```

Bool and Int Negative Op:

```
Let x=78<-78;
Let y=78+78;-x;-y;
print x;
print y;
```

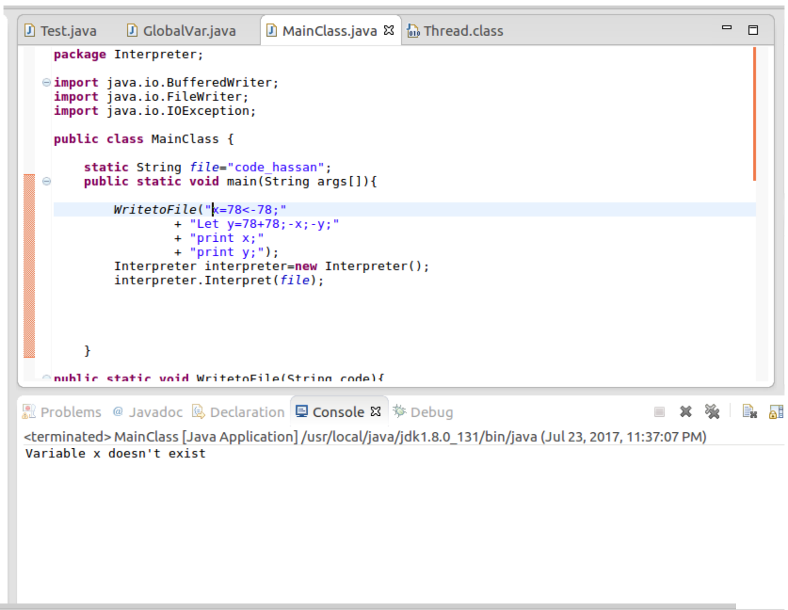


```
package Interpreter;

import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;

public class MainClass {

    static String file="code_hassan";
    public static void main(String args[]){

        WritetoFile("Let x=78<-78;"
                + "Let y=78+78;-x;-y;"
                + "print x;"
                + "print y;");
        Interpreter interpreter=new Interpreter();
        interpreter.Interpret(file);



    }
public static void WritetoFile(String code){
```

```
<terminated> MainClass [Java Application] /usr/local/java/jdk1.8.0_131/bin/java (Jul 23, 2017, 11:36:05 PM)
true
-156
```

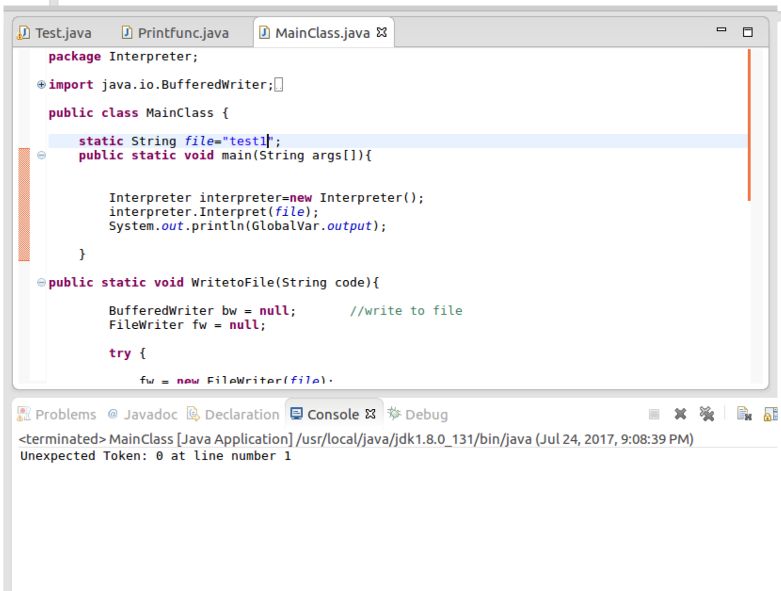
Unknown Variable:

```
x=78<-78;
Let y=78+78;-x;-y;
print x;
print y;
```



```
package Interpreter;

import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;

public class MainClass {

    static String file="code_hassan";
    public static void main(String args[]){

        WritetoFile("x=78<-78;"
                + "Let y=78+78;-x;-y;"
                + "print x;"
                + "print y;");
        Interpreter interpreter=new Interpreter();
        interpreter.Interpret(file);



    }
public static void WritetoFile(String code){
```

```
<terminated> MainClass [Java Application] /usr/local/java/jdk1.8.0_131/bin/java (Jul 23, 2017, 11:37:07 PM)
Variable x doesn't exist
```



```
package Interpreter;

import java.io.BufferedWriter;

public class MainClass {

    static String file="test1";
    public static void main(String args[]){


        Interpreter interpreter=new Interpreter();
        interpreter.Interpret(file);
        System.out.println(GlobalVar.output);

    }
public static void WritetoFile(String code){

        BufferedWriter bw = null;       //write to file
        FileWriter fw = null;

        try {

            fw = new FileWriter(file);
```

```
<terminated> MainClass [Java Application] /usr/local/java/jdk1.8.0_131/bin/java (Jul 24, 2017, 9:08:39 PM)
Unexpected Token: 0 at line number 1
```

Syntax Error:

```
Let 0=x;Y=x;

Let x = (x==y)

x = x+1
```