Interpreter for executing a list of declarative commands

This generic interpreter understands three classes of commands, which are:

* Variable declaration

If we say “let x = 3” so it should create a variable “x” with a value of “3”. It also decides whether the new variable is integer, float or string. If null variable is created it considers it as a string.

* Operations

If after declaration we say “x = x + 3” then it should replace the value of x in the equation, evaluates it and then assigns the new result to the variable x.

* Printing

Finally after calculation if a user says “print x” then it should prints a value of x.

**Note:**

Since I have used templates therefore integer, float or string any operations can be performed.

# Interpreter Class

This class consists of all the basic logic of interpreter this class takes all the commands and then parses it depending on the nature of command. Following function of the class is used to parse the commands.

# Parse(arrayList file)

Then parse function is called with that array list as an argument. It reads each line from that array and parses it depending on the nature of the command.

Variable declaration, printing and operations are performed in this function.

When this function gets a “let” command it creates a new variable which is an object of a class and then stores it in a hash table as variable name and variable object pair. Variable object simply saves its name, value and variable type. Since we have used templates therefore any operation whether it is an integer, float or string, it works correctly.

# Variable Class

This class simply creates a variable object. In this object variable name , variable value and variable type is stored.

# readFile(String filename)

Initially read file function reads a whole file at once and saves it in array list line by line. It gets the filename as an argument, opens that file and stores it in an array list.

Unit Tests

# IntTest class

This class runs all the unit tests on the interpreter.

## IntegerTest()

It first reads the file for the integer test and runs it if all the instructions get executed correctly without any exception test case passes

## FloatTest()

It first reads the file for the float test and runs it if all the instructions get executed correctly without any exception test case passes

## StringTest()

It first reads the file for the string test and runs it if all the instructions get executed correctly without any exception test case passes

GITHUB

https://github.com/hunainarif/interpreter