

# Guide to My IDE

Steven O'Meara

CMPT 440

Professor Rivas

May 2nd, 2016

## Table of Contents

Part 1: Setting up/Running my Project

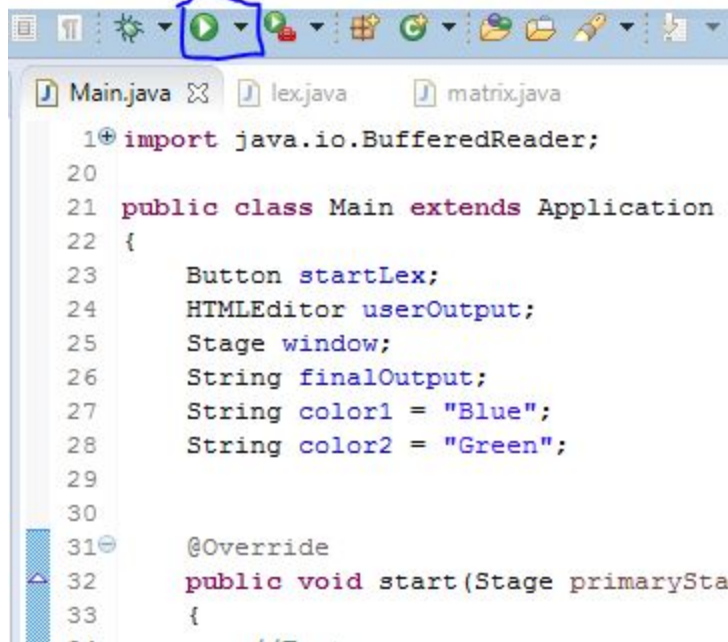
Part 2: A Guide to the IDE

# Part 1 - Setting up/Running my Project

-Clone the Milestone.2 folder from github,

[https://github.com/Steven-omeara/Formal\\_Languages/tree/master/Milestone.2](https://github.com/Steven-omeara/Formal_Languages/tree/master/Milestone.2)

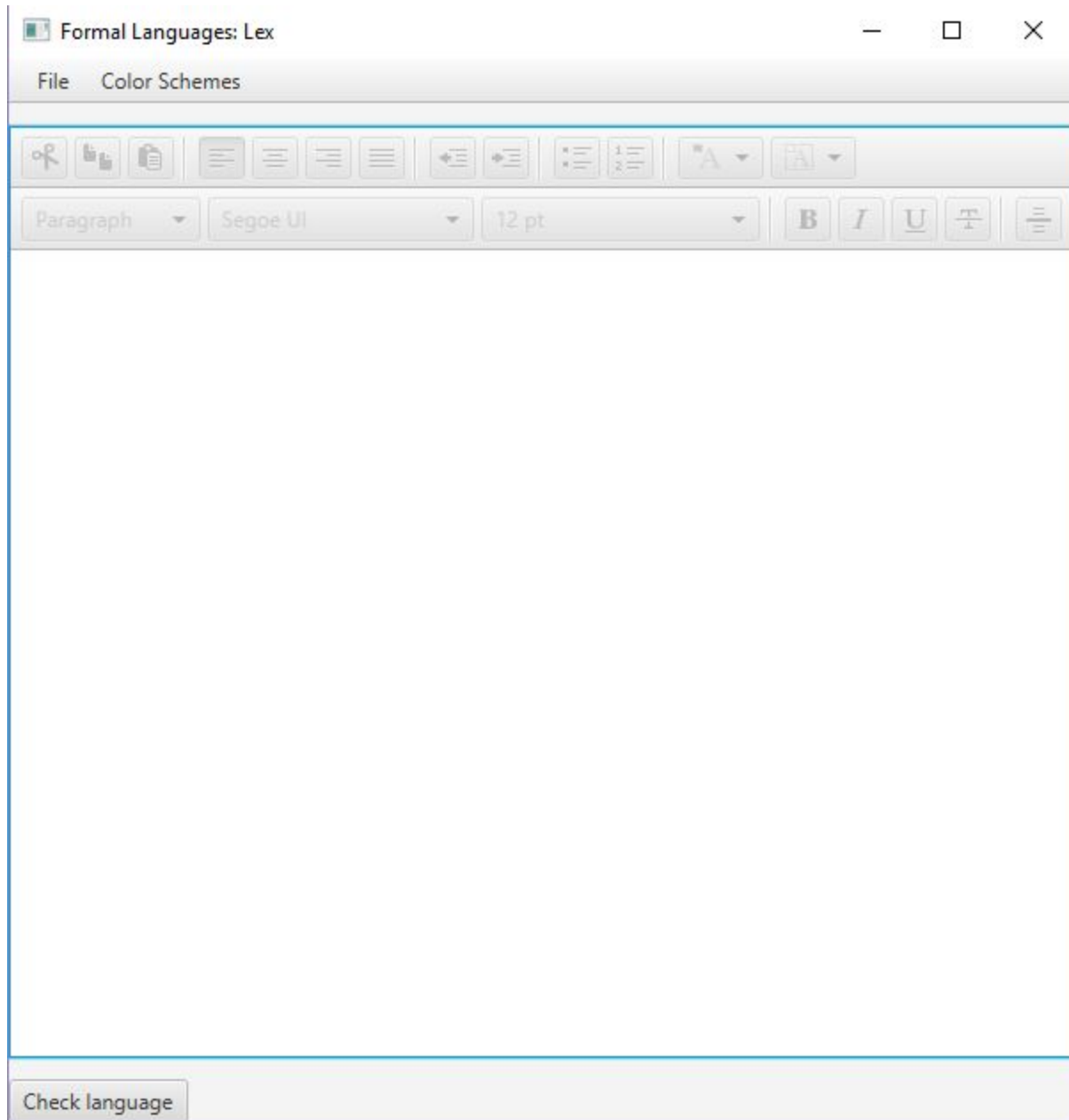
and extract the three .java files, make sure they stay in the same folder if you choose to move them to another location (Main.java, lex.java, matrix.java). Either compile the files through the command line then execute Main.java or run it through your favorite IDE, I used Eclipse. An example of how to run it with Eclipse is included below.



# Part 2 - A Guide to the IDE

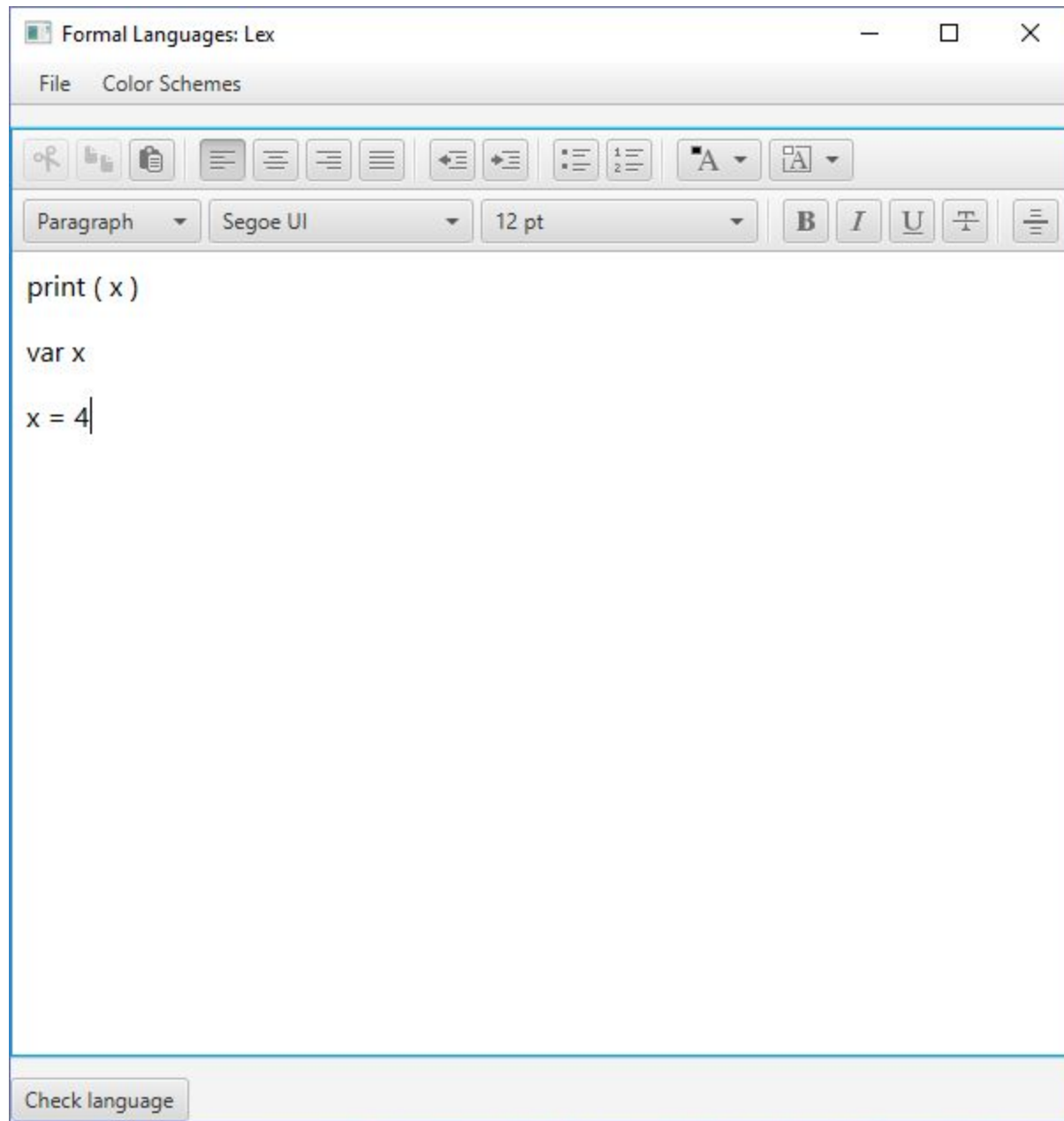
## An intro to the IDE

-The IDE is where all of the magic happens, I created it using JavaFx and its panel system. It includes a file system, a HTML editor and a button that runs the code through the DFA. The entire functionality of the IDE stems off the the main panel in the middle, which is where the user inputs their code. The basic layout is shown below:

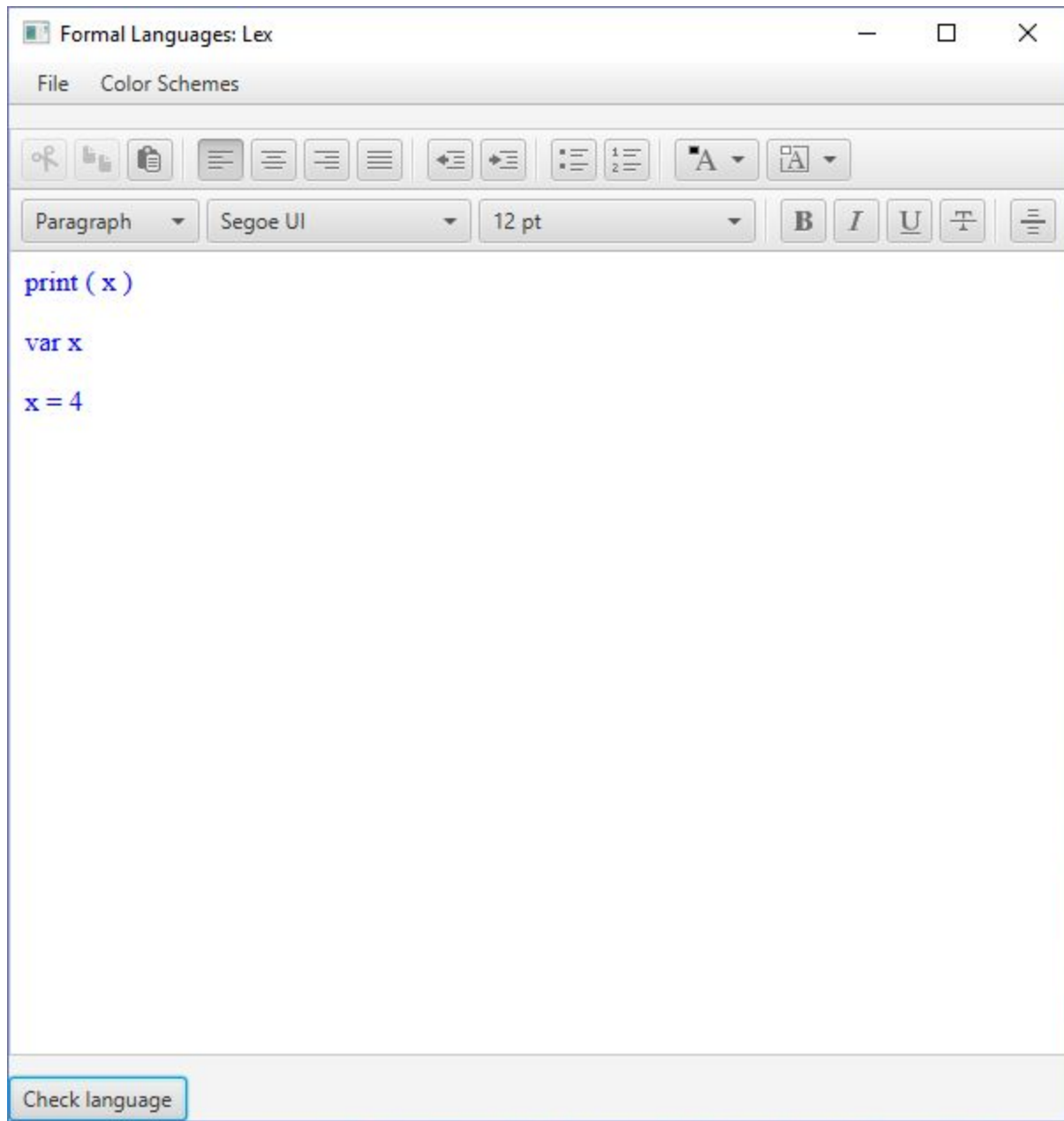


## The Check Language Button

-The check language button in the bottom right corner checks the string in the HTML editor box against the DFA. The first step of using the check language button is inputting code into the HTML editor box:

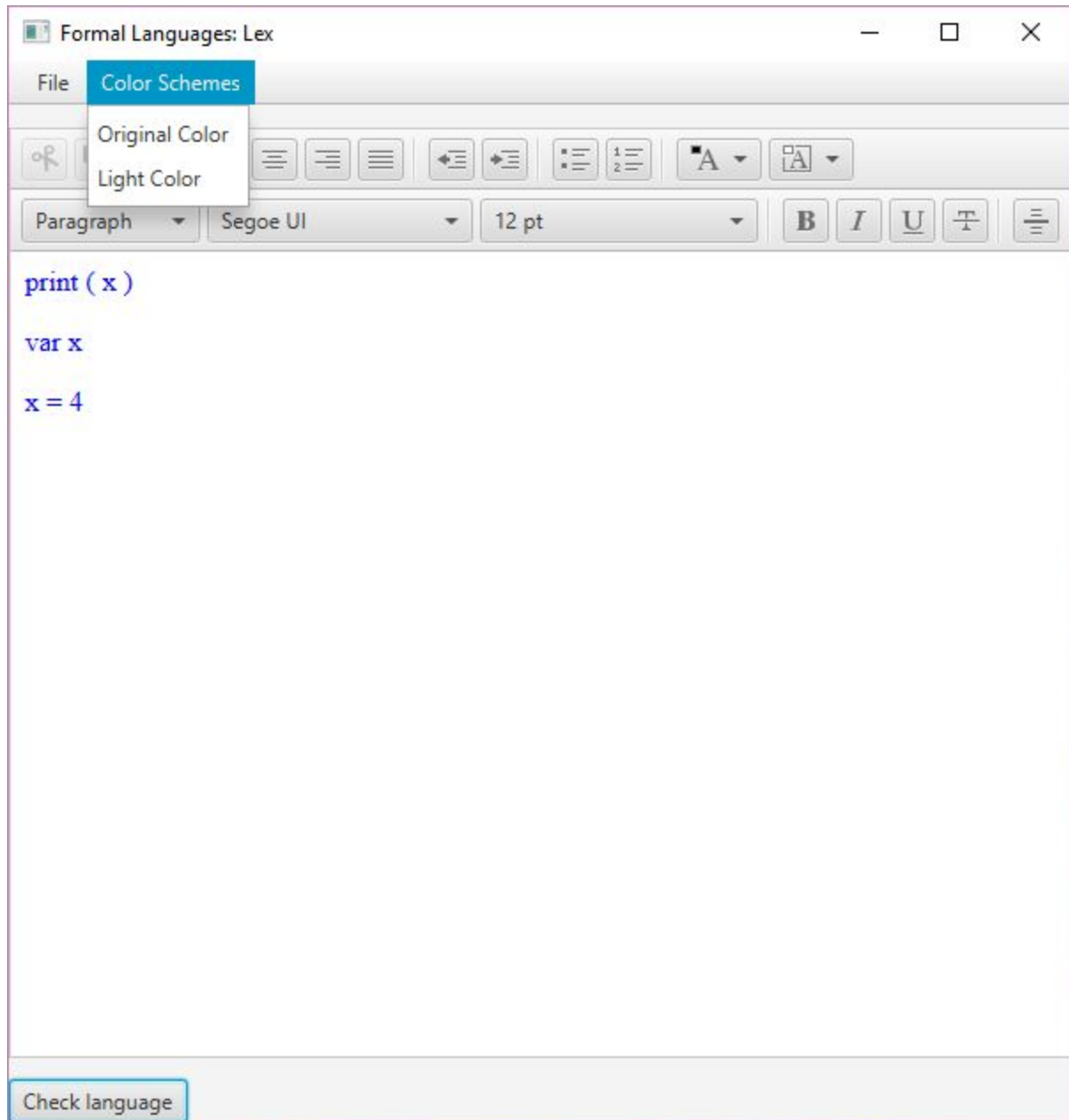


-You then press the check language button, which starts to check the inputted string against the DFA and returns the results. In the basic color scheme the correct statements will be blue, the comments will be green and the errors will be red. There will be more information about this later but if you push the button on the code above you will get the following:



## Color Schemes

-On the top of the IDE is the options to change the colors. When you click the Color Schemes tab it will drop two options, original or light. The IDE starts with the original option as it has to have something to start it off but to change the output just click on an option. After you choose an option you have to press the Check Language button again to run the string through the DFA and change the color values. The process is shown below with the light option but the original color option functions in the same way(The light color option changes accepts to purple, comments to pink and errors stay red to keep consistency):





```
print ( x )
```

```
var x
```

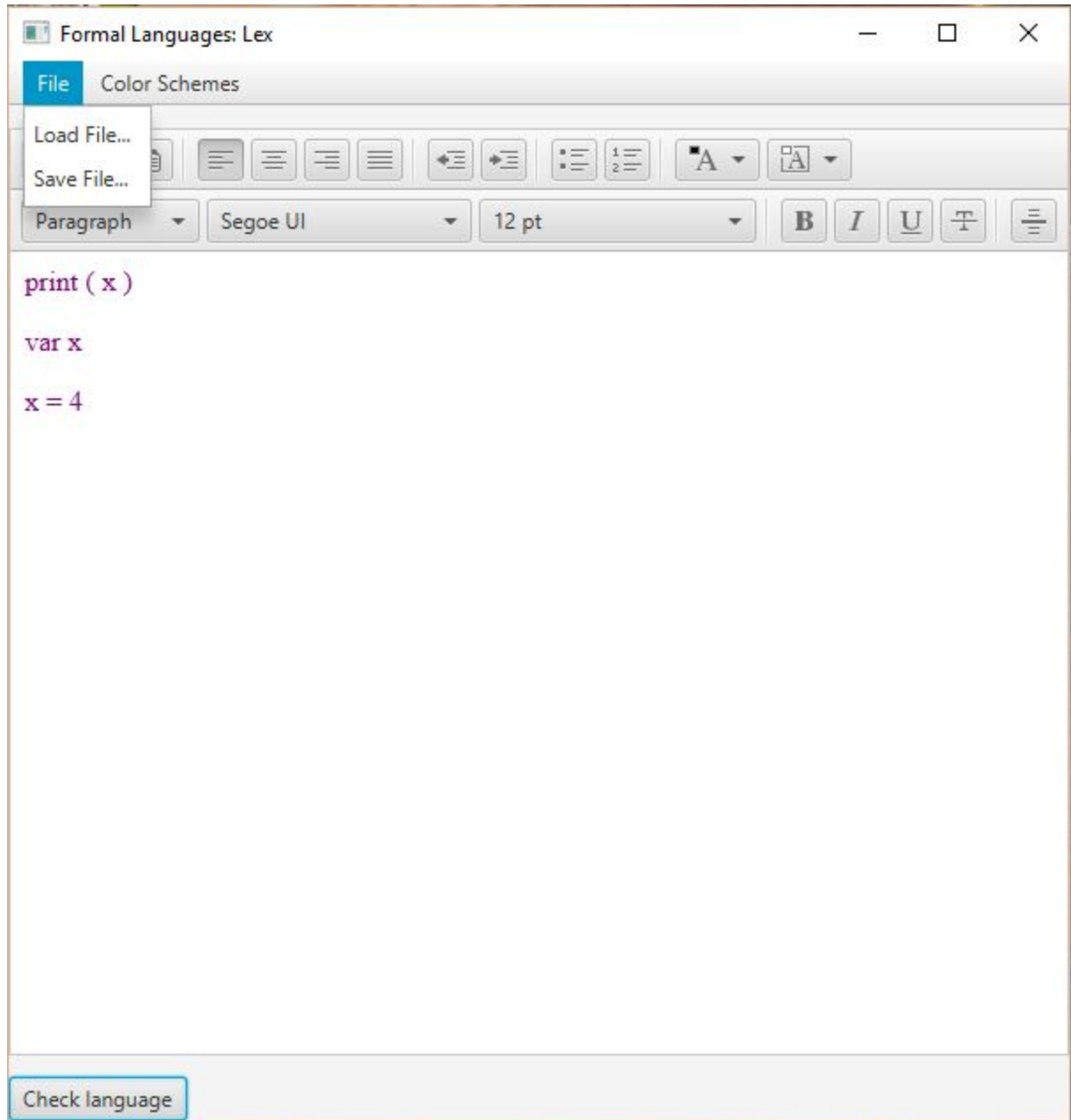
```
x = 4
```

[Check language](#)

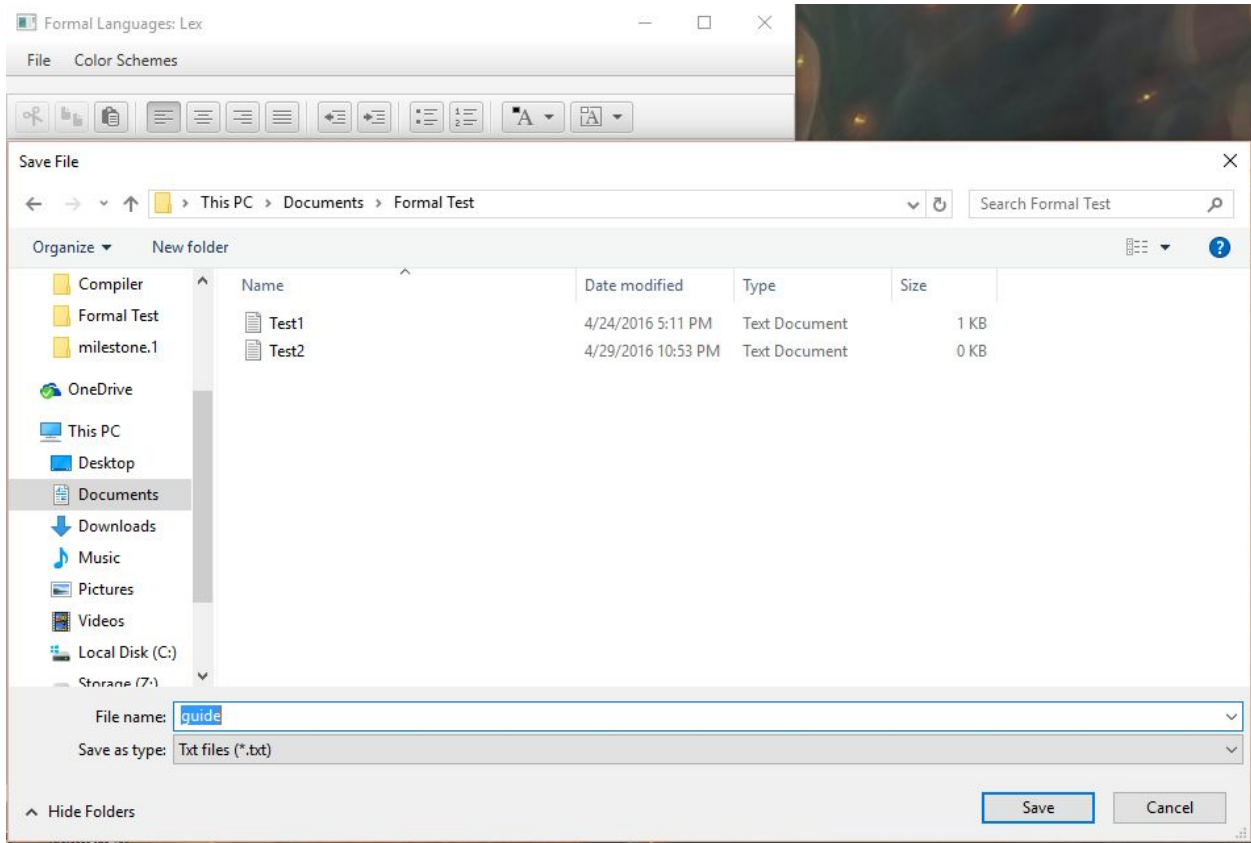


## Saving Files

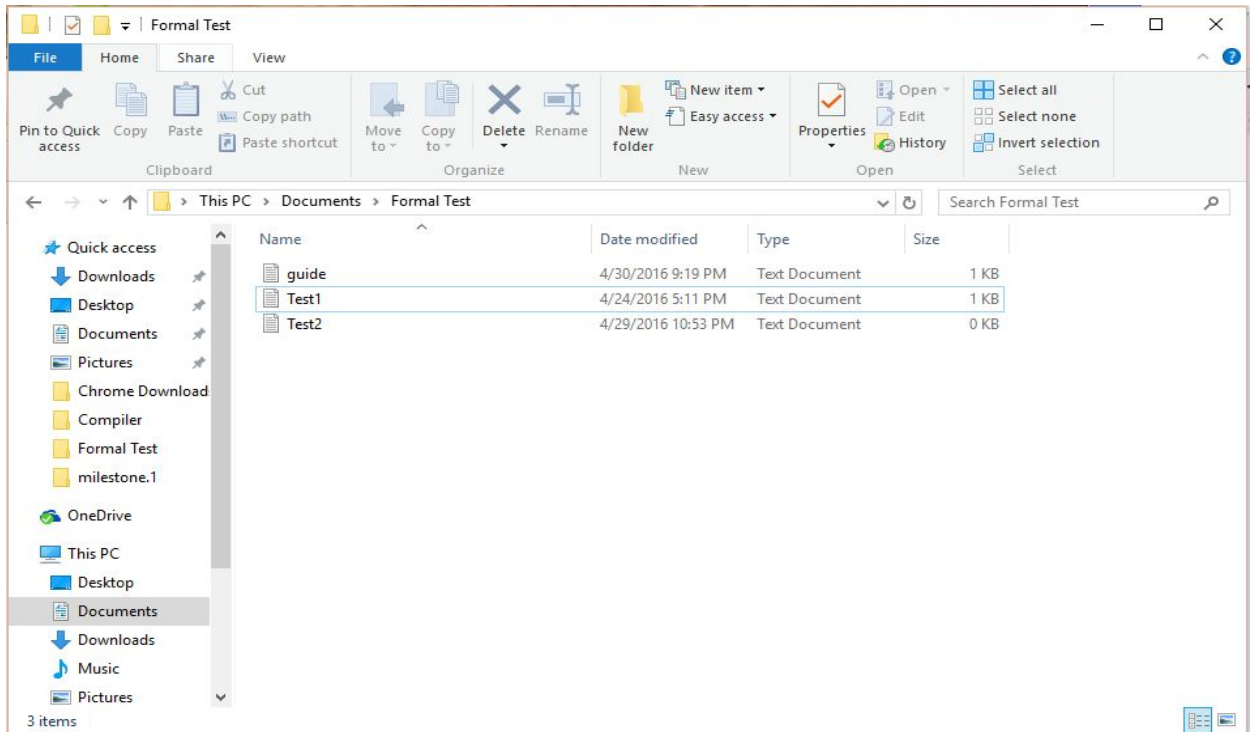
-One of the last options of the IDE is to save the text files that the user inputs. This can be done with the buttons in the top left corner of the IDE. If you click File you will be presented with two options, save file and load file. Save file will take the given string in the HTML field and save it to a .txt file (there is a filter so that it will only save to a .txt file). We will show this process with the same string as above, the process is illustrated below:



After clicking save file:

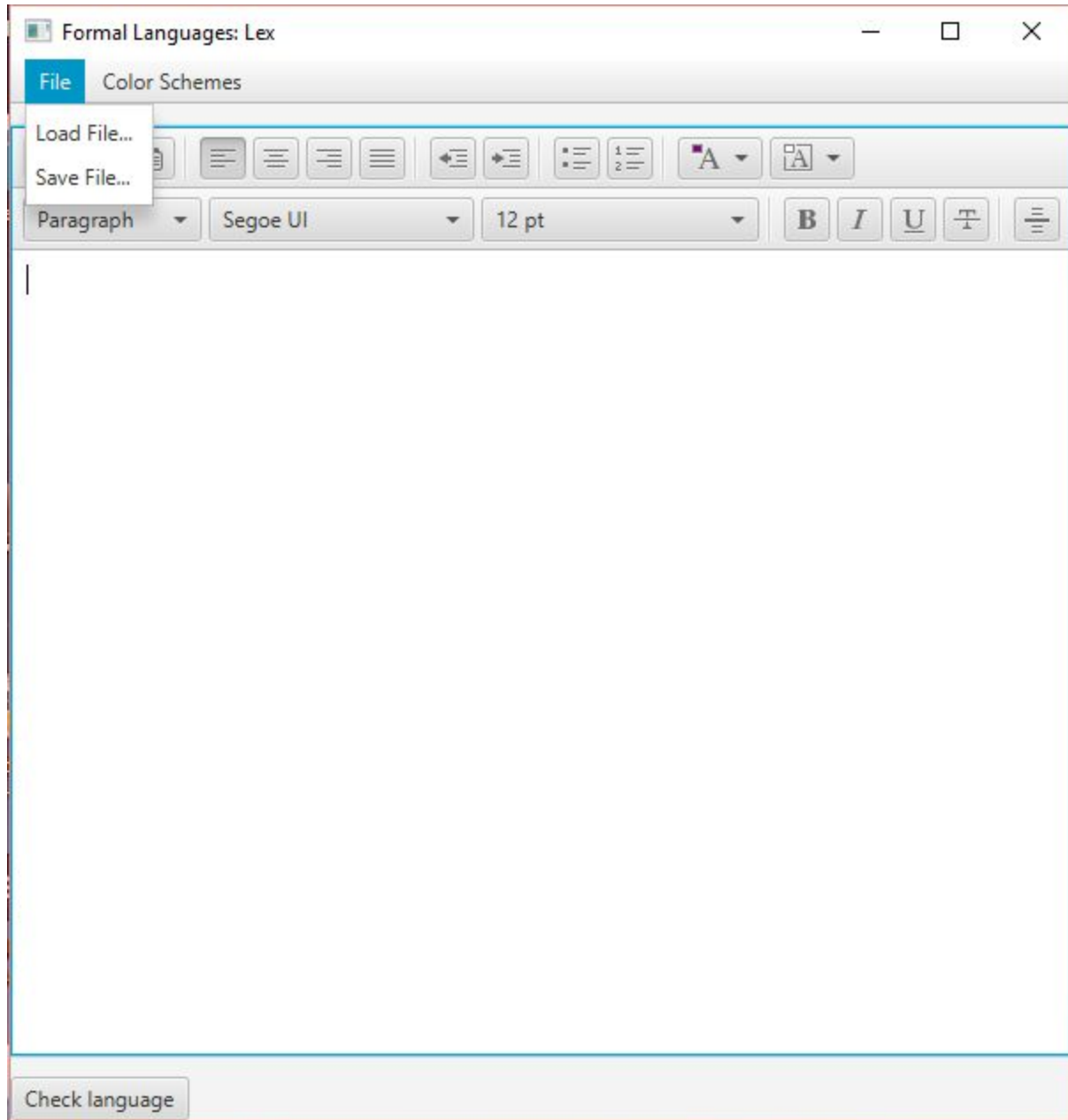


-Once you hit save file the file will be saved into the location that you have given it:

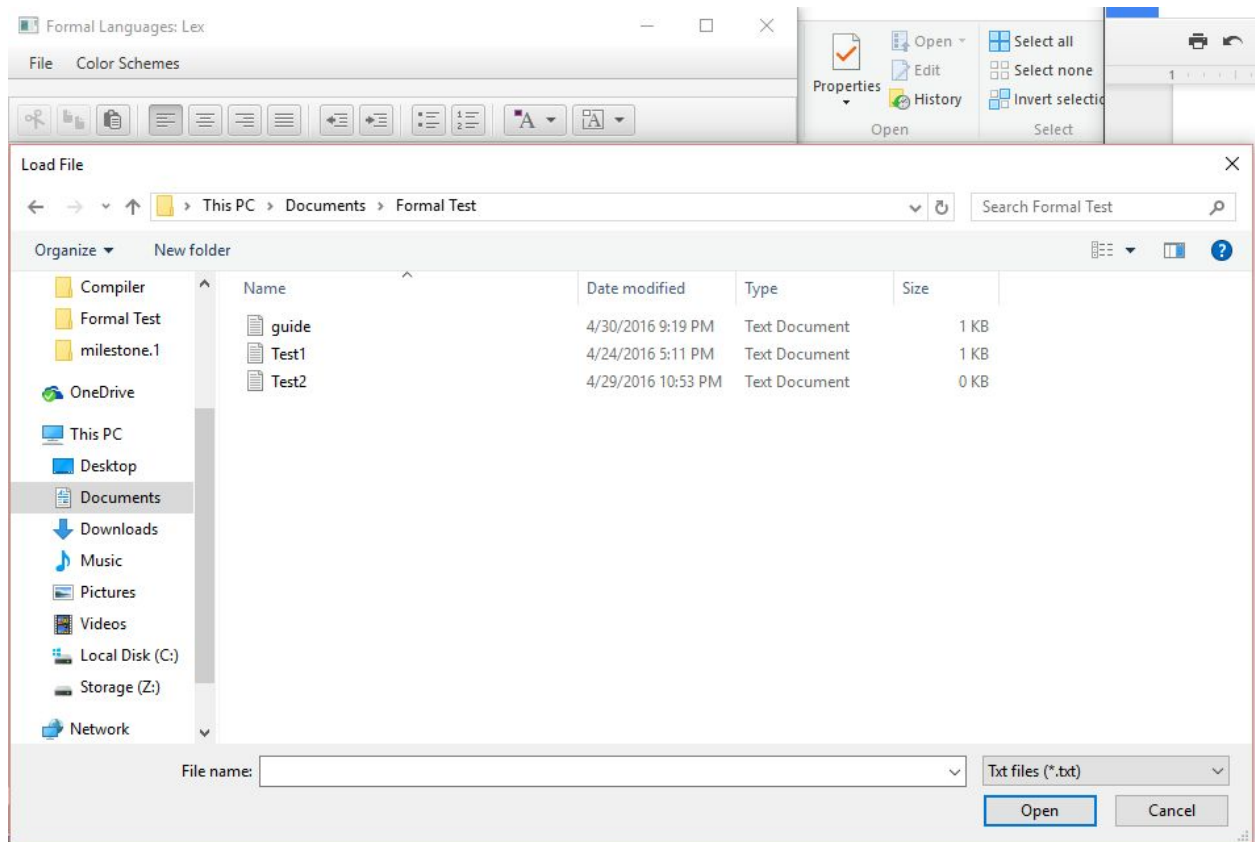


## Load File

-Right below save file is the load file option. This button is identical to save file but instead of saving a file it will take a .txt file and load the string inside into the HTML editor field. How this is done is illustrated below(we are going to load the file we saved in the previous step):



After you hit the load button, you will be given an file explorer prompt, just choose the .txt file that you want to load (it will only find .txt files):



When you load the string, it will be un-formatted as it hasn't been put through the DFA yet:

