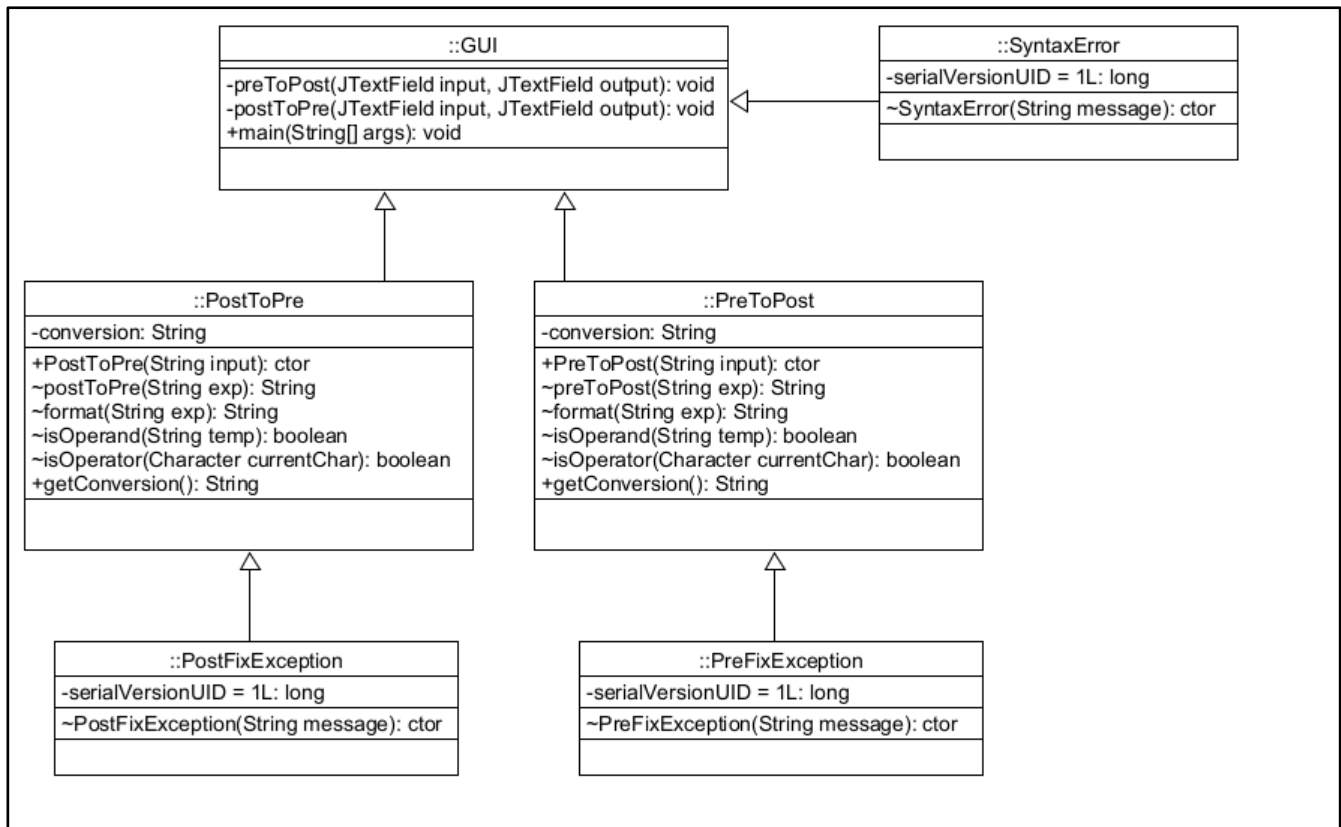


Allen Taylor
CMIS 350 6382
Project 1
1/16/2022

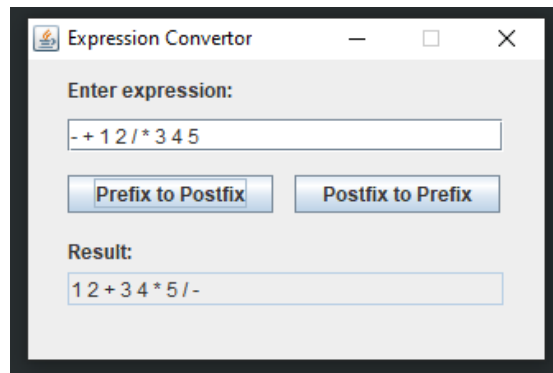
UML Diagrams



Test Case 1

Requirement: All operators included in test cases.

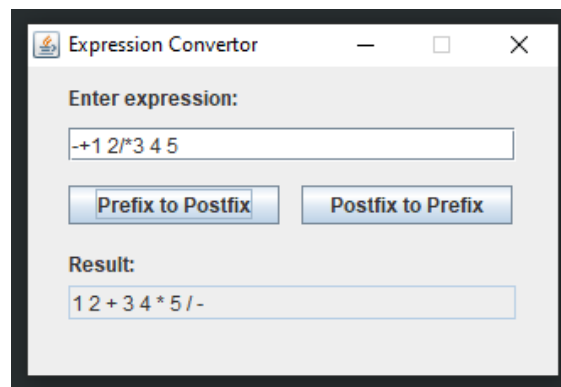
Input:	- + 1 2 / * 3 4 5
Expected Output:	1 2 + 3 4 * 5 / -
Actual Output:	1 2 + 3 4 * 5 / -



Test Case 2

Requirement: Test cases include expressions without spaces.

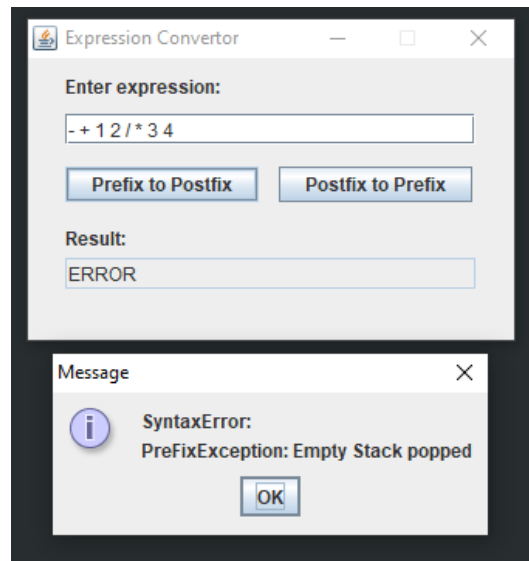
Input:	-+1 2/* 3 4 5
Expected Output:	1 2 + 3 4 * 5 / -
Actual Output:	1 2 + 3 4 * 5 / -



Test Case 3

Requirement: Test cases include cases that pop an empty stack.

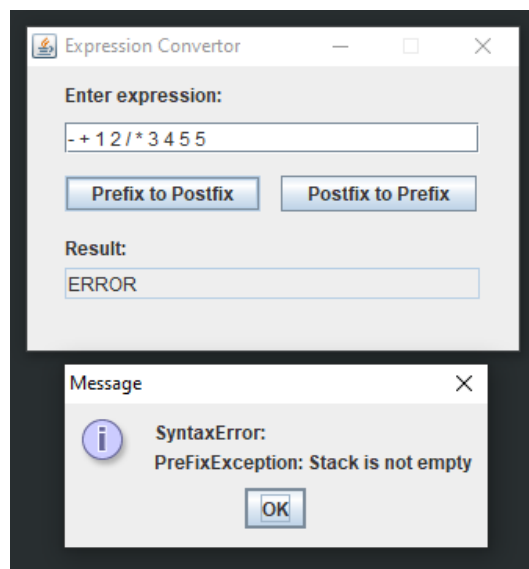
Input:	- + 1 2 / * 3 4
Expected Output:	Syntax Error with JOptionPane is displayed.
Actual Output:	Syntax Error with JOptionPane is displayed.



Test Case 4

Requirement: Test cases include cases which result in a non-empty stack.

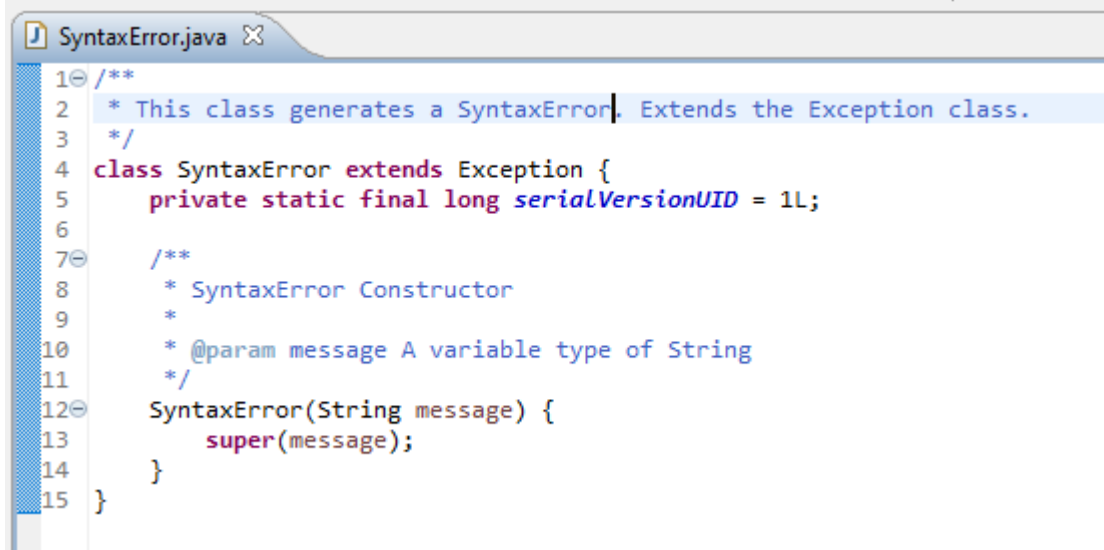
Input:	- + 1 2 / * 3 4 5 5
Expected Output:	Syntax Error with JOptionPane is displayed.
Actual Output:	Syntax Error with JOptionPane is displayed.



Lessons Learned

I had difficulties with generating a custom exception, but I used the following process to resolve the issue.

1. First, I created the **Syntax Error Class** that extends **Exception**, then used **super** within the constructor.



```
1- /**
2-  * This class generates a SyntaxError. Extends the Exception class.
3-  */
4- class SyntaxError extends Exception {
5-     private static final long serialVersionUID = 1L;
6-
7-     /**
8-      * SyntaxError Constructor
9-      *
10-      * @param message A variable type of String
11-      */
12-     SyntaxError(String message) {
13-         super(message);
14-     }
15- }
```

2. Next, I used **throws new SyntaxError()** on both the **preToPost()** and **postToPre()** methods.

```
/**
 * This method is used to convert an expression from Prefix to Postfix.
 *
 * @param input A variable type of JTextField
 * @param output A variable type of JTextField
 * @exception SyntaxError On syntax error
 */
private static void preToPost(JTextField input, JTextField output) throws SyntaxError {

    String exp = input.getText();
    if (input.getText().isEmpty() == false) {

        try {
            PreToPost p2p = new PreToPost(exp);
            output.setText(p2p.getConversion());
        } catch (Exception e) {
            output.setText("ERROR");
            throw new SyntaxError("\n" + e.toString());
        }

    } else {
        output.setText("Please enter an expression.");
    }

}
```

3. Last, I had to surround the method inside a **try/catch block** and display the **error string** with a **JOptionPane**.

```
preToPostButton.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent e) {  
        try {  
            preToPost(input, output);  
        } catch (Exception e1) {  
            JOptionPane.showMessageDialog(frame, e1);  
        }  
    }  
});
```