main.py

```python
 4      # Press Double Shift to search everywhere for classes, files, tool windows, actions, and s
 5
 6      from tkinter import *
 7
 8      currentline = 0.0
 9
10
11      class LexerGUI:  # class def
12
13          # init function
14          def __init__(self, root):
15              # master parent obj
16              self.master = root
17              # title of window
18              self.master.title("Lexer Tinypie")
19              # frame widget
20              self.mainframe = Frame(self.master, width=2000,height=2000, bg='#D4EEE3')
21              self.mainframe.pack()
22              self.leftframe = Frame(self.mainframe, width=500, height=500, bg='light blue')
23              self.leftframe.grid(row=1, column=0, padx=10, pady=5)
24
25              self.rightframe = Frame(self.mainframe, width=500, height=500, bg='light blue')
26              self.rightframe.grid(row=1, column=1, padx=10, pady=5, sticky=N)
27
```

LexerGUI › __init__()

Run: main

Lexer Tinypie

Source Code Input:

```
int a=4
int b=5
c=a+b
```

Lexical Analyzed Result

Current Processing Line: 0

Next Line

Quit

Version Control   Run   TODO   Problems   Terminal   Python Packages   Python Console   Services

22:85 (10 chars)   CRLF   UTF-8   4 spaces   Python 3.11 (hw4)

main.py

```
28
29        # variables
30        global currentline
31        self.linecounter = StringVar()
32        self.linecounter.set("0")
33
34        # lefthand side of gui
35        # label1 : source code
36        self.labelsrcin = Label(self.leftframe, text="Source Code Input:", bg='light blue')
37        self.labelsrcin.grid(row=0, column=0, sticky=W)
38
39        # input text box multiline for label1
40        self.textinput = Text(self.leftframe, width=40, height=5)
41        self.textinput.grid(row=1, column=0, pady=5, columnspan=2)
42
43        # label2: processing line
44        self.labelcurrentline = Label(self.leftframe, text="Current Processing Line:", bg='light blue
45        self.labelcurrentline.grid(row=2, column=0, sticky=W)
46
47        # entry text box for current line dispaly
48        self.entrylinecount = Entry(self.leftframe, width=5, textvariable=self.linecounter, relief='s
49        self.entrylinecount.grid(row=2, column=1, sticky=E, pady=5, padx=5)
50        self.entrylinecount['state'] = 'disabled'
51
52        # button1: next line button
```

LexerGUI > __init__()

Run: main

```
C:\                                                              s\hw4\main.py
1.
```

Lexer Tinypie

Source Code Input:

```
int a=4
int b=5
c=a+b
```

Lexical Analyzed Result

```
int a=4
```

Current Processing Line: 1

Next Line   Quit

```python
        self.buttonnext = Button(self.leftframe, text="Next Line", command=self.nextline)
        self.buttonnext.grid(row=3, column=1, sticky=E, pady=(0,10), padx=5)

        ##righthand side
        # label3 : result for
        self.labelresult = Label(self.rightframe, text="Lexical Analyzed Result", bg='light blue')
        self.labelresult.grid(row=0, column=0, sticky=W)
        # text box output multiline for label3
        self.outputbox = Text(self.rightframe, width=40, height=5)
        self.outputbox.grid(row=1, column=0, columnspan=2, sticky=W, pady=(5,32))
        #button2: quit button
        self.buttonnext = Button(self.rightframe, text="Quit", command=self.quit)
        self.buttonnext.grid(row=3, column=1, sticky=SE, pady=(0,10), padx=5)

    def highlight(self, currentline):
        self.textinput.tag_add("highlight", currentline+1, "{}+1lines".format(currentline+1))
        self.textinput.tag_config("highlight", background='#badfda')

    def removehighlight(self,):

        self.textinput.tag_delete("highlight")

    def quit(self):
        self.master.destroy()
```
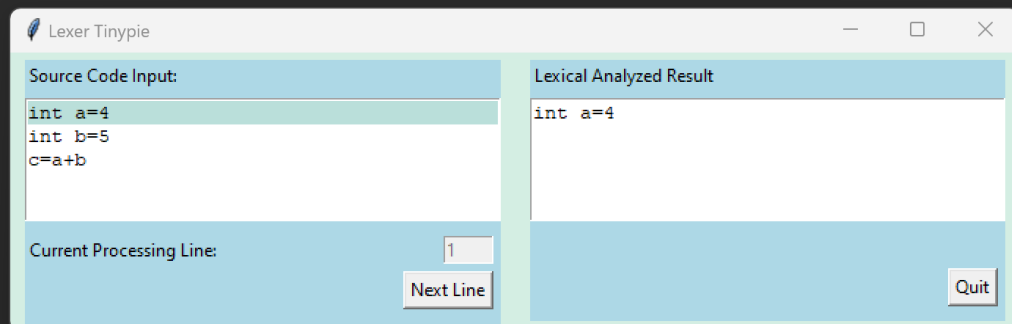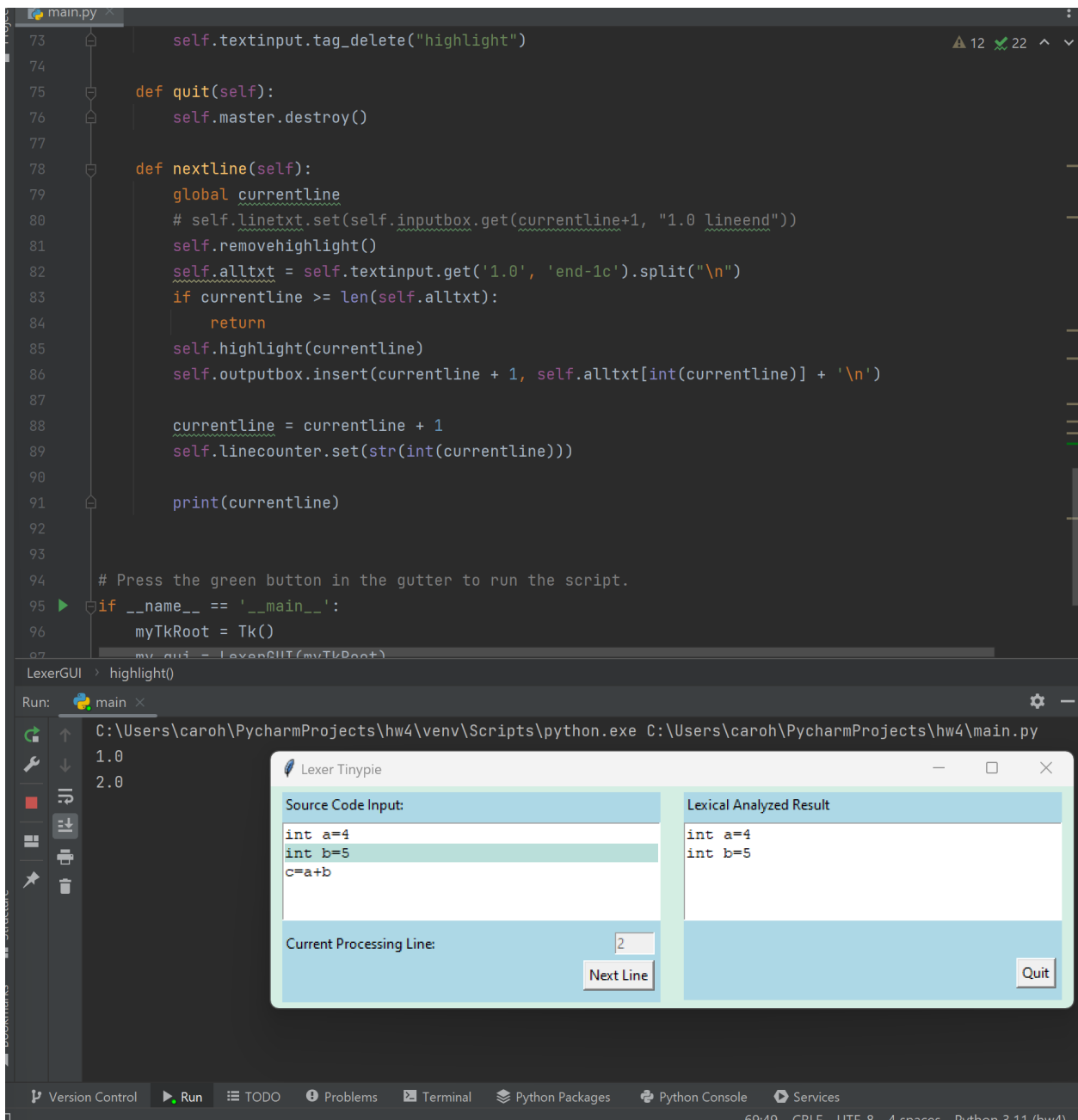
init()
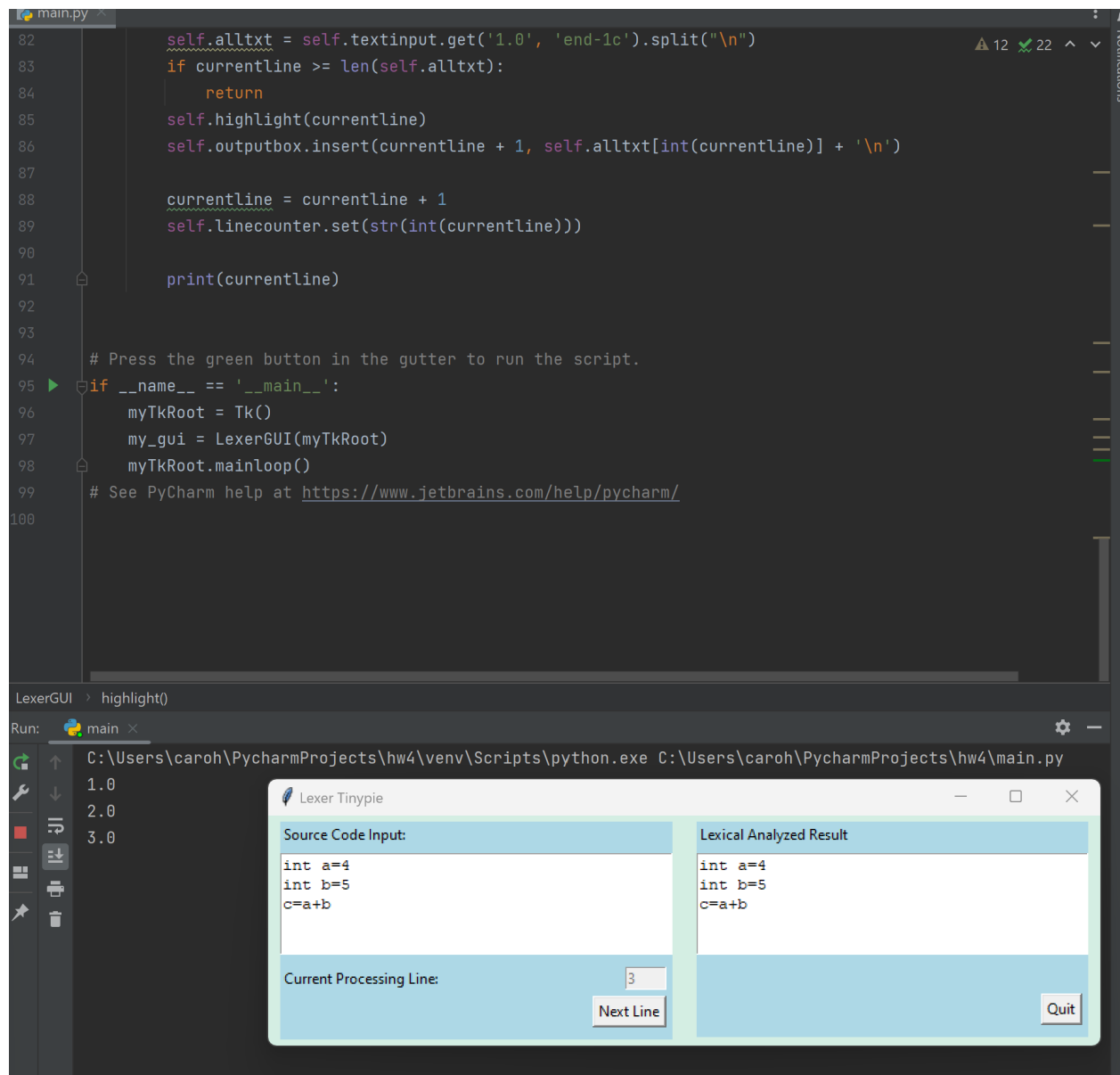
main ×

```
:\Users\caroh\PycharmProjects\hw4\venv\Scripts\python.exe C:\Users\caroh\PycharmProjects\hw4\main.py
.0
```

Lexer Tinypie

Source Code Input:
```
int a=4
int b=5
c=a+b
```

Lexical Analyzed Result
```
int a=4
```

Current Processing Line: | 1 |

Next Line

Quit

```
73              self.textinput.tag_delete("highlight")

74

75          def quit(self):
76              self.master.destroy()

77

78          def nextline(self):
79              global currentline
80              # self.linetxt.set(self.inputbox.get(currentline+1, "1.0 lineend"))
81              self.removehighlight()
82              self.alltxt = self.textinput.get('1.0', 'end-1c').split("\n")
83              if currentline >= len(self.alltxt):
84                  return
85              self.highlight(currentline)
86              self.outputbox.insert(currentline + 1, self.alltxt[int(currentline)] + '\n')

87

88              currentline = currentline + 1
89              self.linecounter.set(str(int(currentline)))

90

91              print(currentline)

92

93

94      # Press the green button in the gutter to run the script.
95      if __name__ == '__main__':
96          myTkRoot = Tk()
97          my_gui = LexerGUI(myTkRoot)
```

LexerGUI > highlight()

Run: main

```
C:\Users\caroh\PycharmProjects\hw4\venv\Scripts\python.exe C:\Users\caroh\PycharmProjects\hw4\main.py
1.0
2.0
```

Lexer Tinypie

Source Code Input:

```
int a=4
int b=5
c=a+b
```

Lexical Analyzed Result

```
int a=4
int b=5
```

Current Processing Line: 2

Next Line

Quit

Version Control   Run   TODO   Problems   Terminal   Python Packages   Python Console   Services

69:49   CRLF   UTF-8   4 spaces   Python 3.11 (hw4)

```python
            self.alltxt = self.textinput.get('1.0', 'end-1c').split("\n")
            if currentline >= len(self.alltxt):
                return
            self.highlight(currentline)
            self.outputbox.insert(currentline + 1, self.alltxt[int(currentline)] + '\n')

            currentline = currentline + 1
            self.linecounter.set(str(int(currentline)))

            print(currentline)


# Press the green button in the gutter to run the script.
if __name__ == '__main__':
    myTkRoot = Tk()
    my_gui = LexerGUI(myTkRoot)
    myTkRoot.mainloop()
# See PyCharm help at https://www.jetbrains.com/help/pycharm/
```

LexerGUI > highlight()

Run: main

```
C:\Users\caroh\PycharmProjects\hw4\venv\Scripts\python.exe C:\Users\caroh\PycharmProjects\hw4\main.py
1.0
2.0
3.0
```

Lexer Tinypie

Source Code Input:
```
int a=4
int b=5
c=a+b
```

Lexical Analyzed Result
```
int a=4
int b=5
c=a+b
```

Current Processing Line: 3

Next Line

Quit

After press quit

```python
        currentline = currentline + 1
        self.linecounter.set(str(int(currentline)))

        print(currentline)


# Press the green button in the gutter to run the script.
if __name__ == '__main__':
    myTkRoot = Tk()
    my_gui = LexerGUI(myTkRoot)
    myTkRoot.mainloop()
# See PyCharm help at https://www.jetbrains.com/help/pycharm/
```

Run:

main

```
C:\Users\caroh\PycharmProjects\hw4\venv\Scripts\python.exe C:\Users\caroh\PycharmProjects\hw4\main.py
1.0
2.0
3.0

Process finished with exit code 0
```

Version Control   ▶ Run   ☰ TODO   ❶ Problems   ⊡ Terminal   Python Packages   Python Console   Services