Write a program to generate tokens from given input string.

Intro:

- Lexical analysis is the first phase of a compiler.

- It reads character streams from the source code, checks for legal tokens, and passes the data to the syntax analyser when it demands.

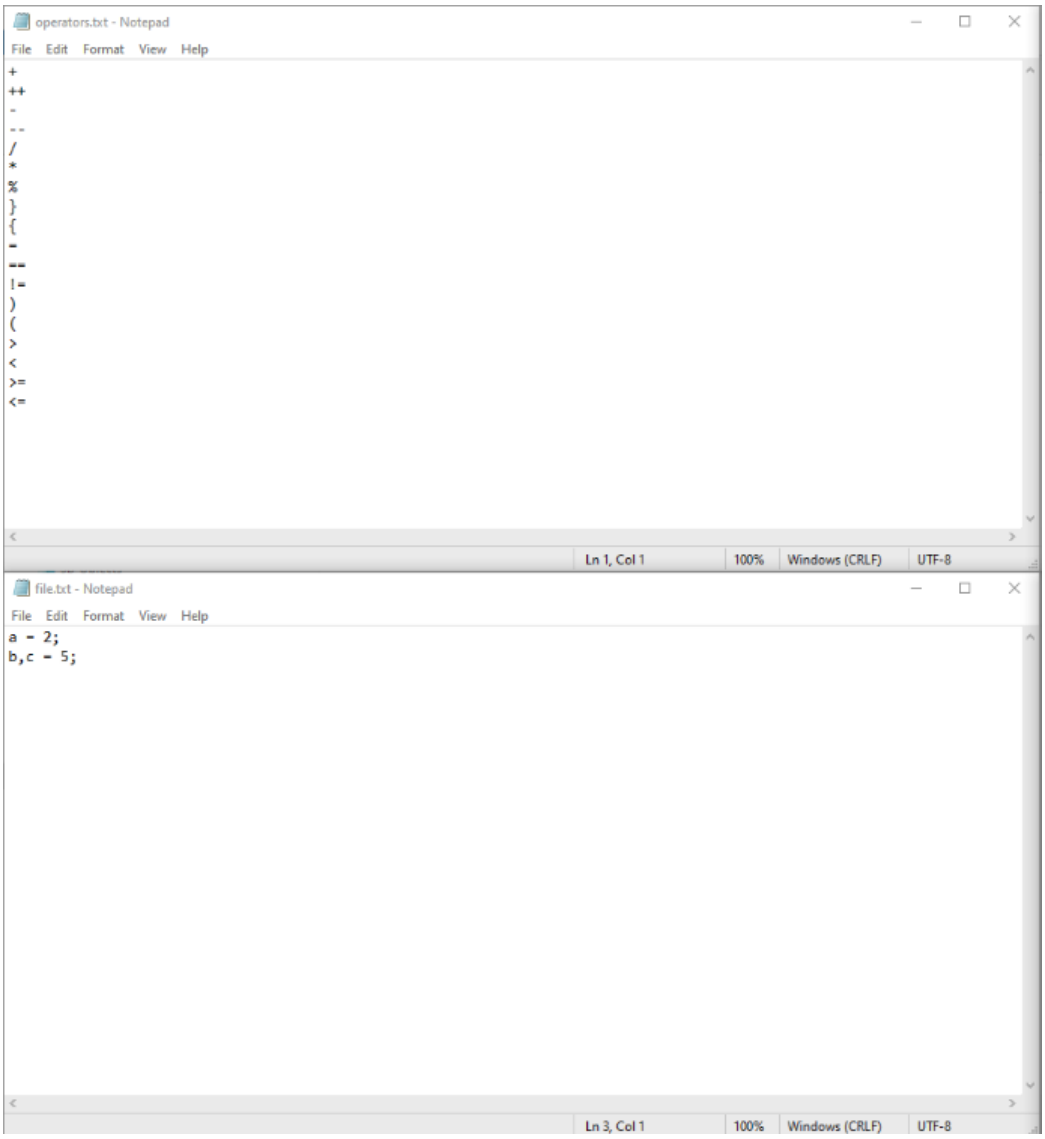
- this analysis generates tokens from given string.

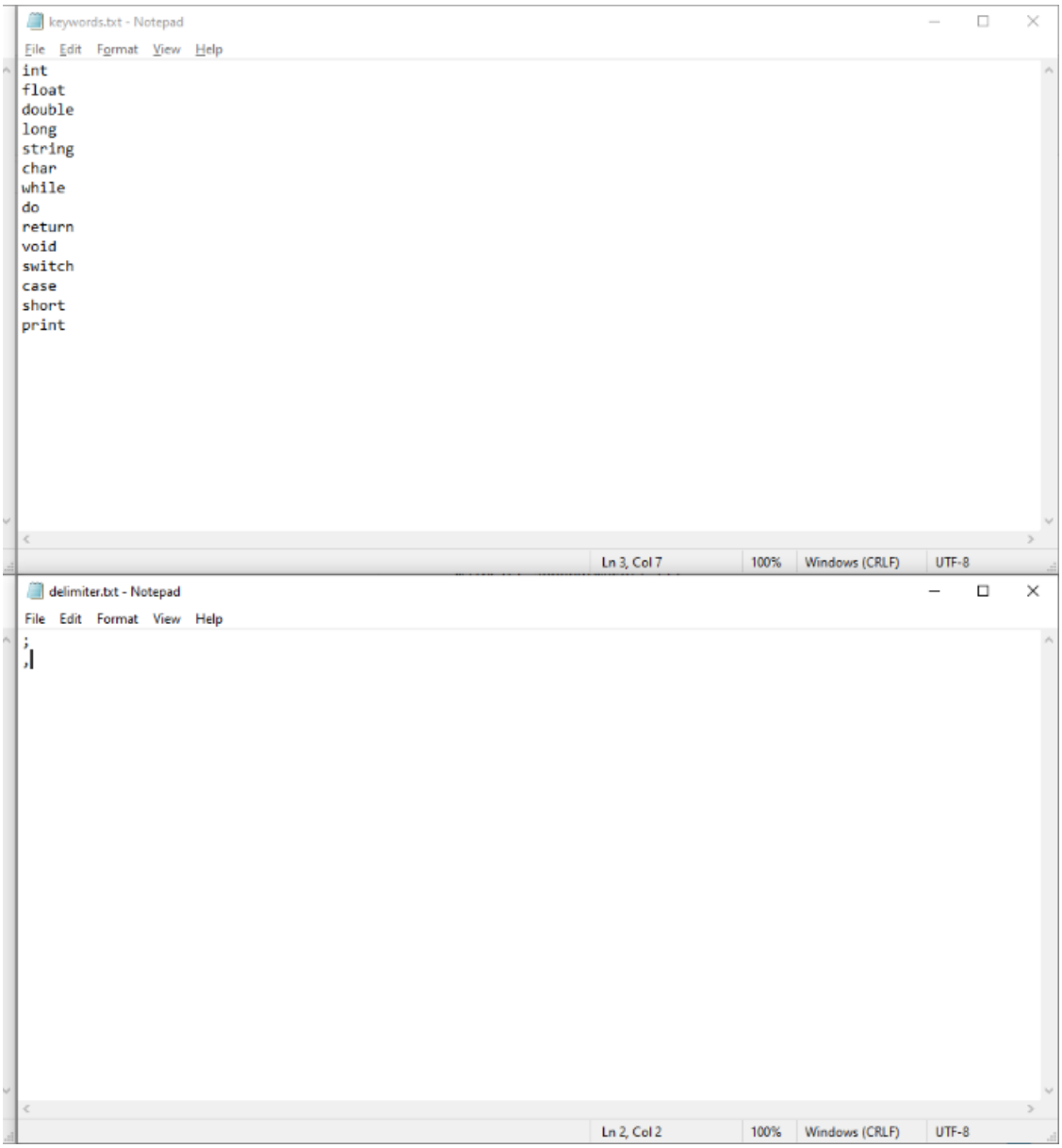
Code:

```
# string = input().split()

keyWords = open("keywords.txt","r")
operators = open("operators.txt","r")
delimiters = open("delimiter.txt","r")
#keywords,op from files
keyW = [char.strip() for char in keyWords]
op = [char.strip() for char in operators]
dl = [char.strip() for char in delimiters]

#input from file
mainFile = open("file.txt",'r')
fileList = mainFile.read().split()
mainList = []
#differenciate list
for word in fileList:
    if ',' in word:
        count = word.count(",")
        temp = word.split(",")
        # print(temp)
        for char in temp:
            mainList.append(char)
            if count > 0:
                mainList.append(",")
            count -= 1
    elif word.endswith(';'):
```

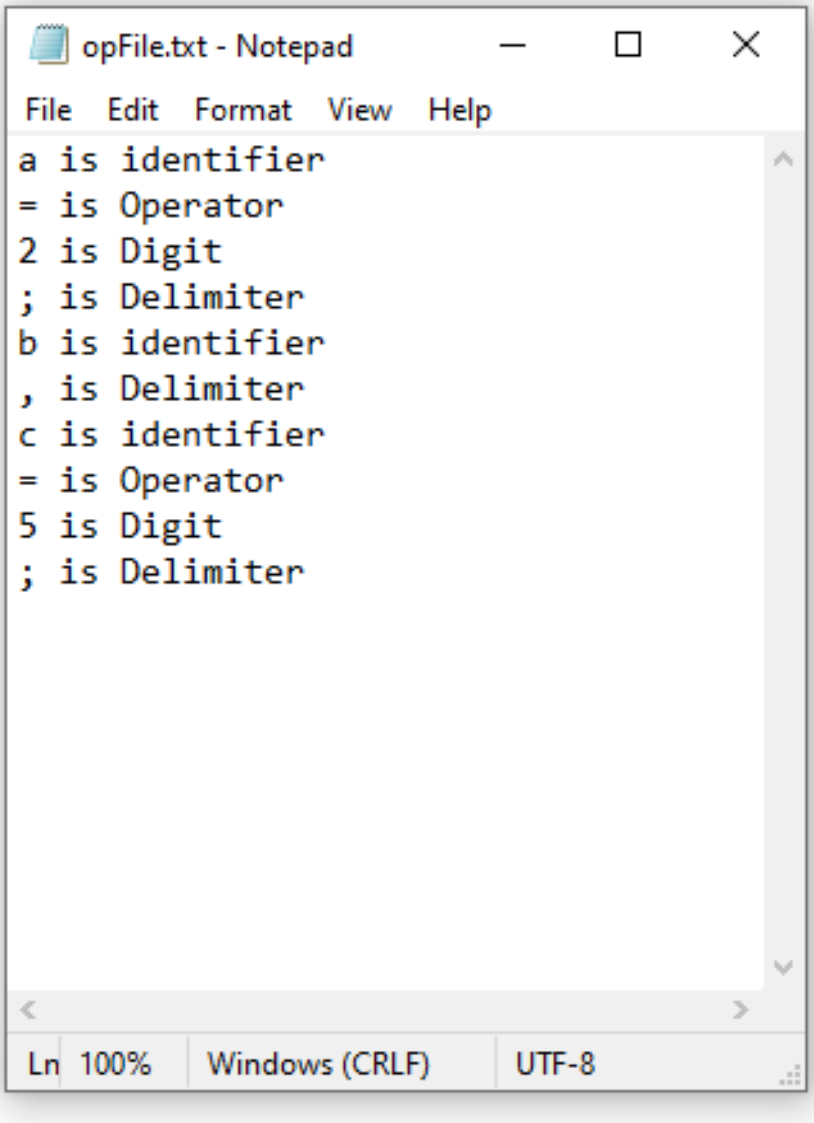
```
        mainList.append(word[:-1])
        mainList.append(";")
    else:
        mainList.append(word)

print(mainList)
opFile = open("opFile.txt",'w')
opFile.close()
opFile = open("opFile.txt",'a')
for token in mainList:
    if token in keyW:
        p = str(token + " is Keyword")
        opFile.write(p+'\n')
    elif token in op:
        p = str(token + " is Operator")
        opFile.write(p+'\n')
    elif token in dl:
        p = str(token + " is Delimiter")
        opFile.write(p+'\n')
    elif token.isdecimal():
        p = str(token + " is Digit")
        opFile.write(p+'\n')
    elif token.isalnum():
        p = str(token + " is identifier")
        opFile.write(p+'\n')
opFile.close()
keyWords.close()
operators.close()
mainFile.close()
```

Files:


operators.txt - Notepad
File Edit Format View Help
+
++
-
--
/
*
%
}
{
=
==
!=
)
(
>
<
>=
<=
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8
file.txt - Notepad
File Edit Format View Help
a = 2;
b,c = 5;
Ln 3, Col 1
100%
Windows (CRLF)
UTF-8


keywords.txt - Notepad
File Edit Format View Help
int
float
double
long
string
char
while
do
return
void
switch
case
short
print
Ln 3, Col 7
100%
Windows (CRLF)
UTF-8
delimiter.txt - Notepad
File Edit Format View Help
;
,
Ln 2, Col 2
100%
Windows (CRLF)
UTF-8

Output:


opFile.txt - Notepad
File Edit Format View Help
a is identifier
= is Operator
2 is Digit
; is Delimiter
b is identifier
, is Delimiter
c is identifier
= is Operator
5 is Digit
; is Delimiter
Ln 100% Windows (CRLF) UTF-8