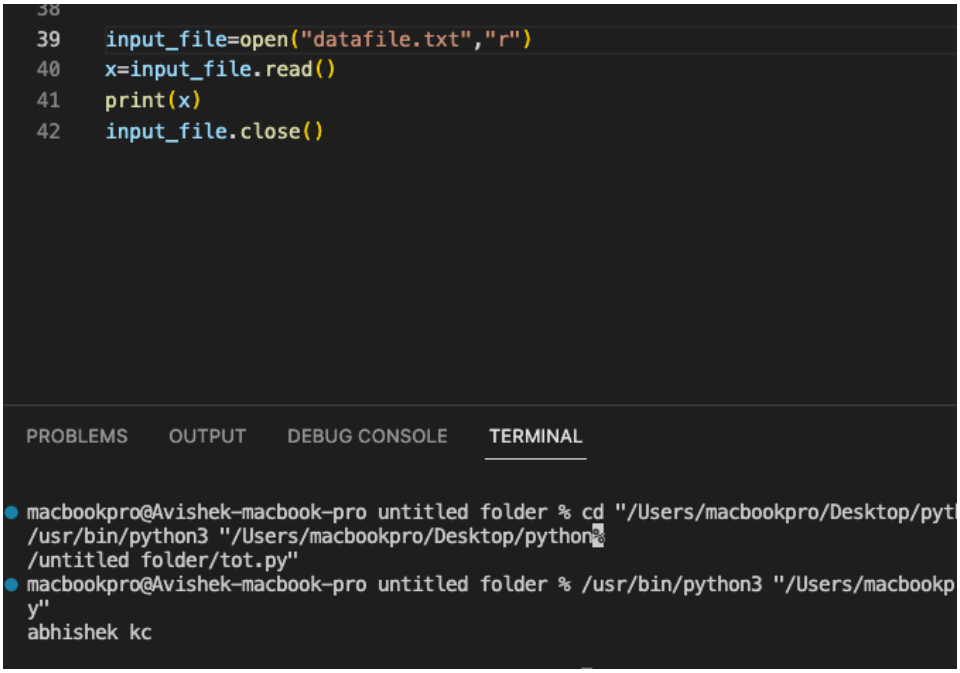
**4CS001**

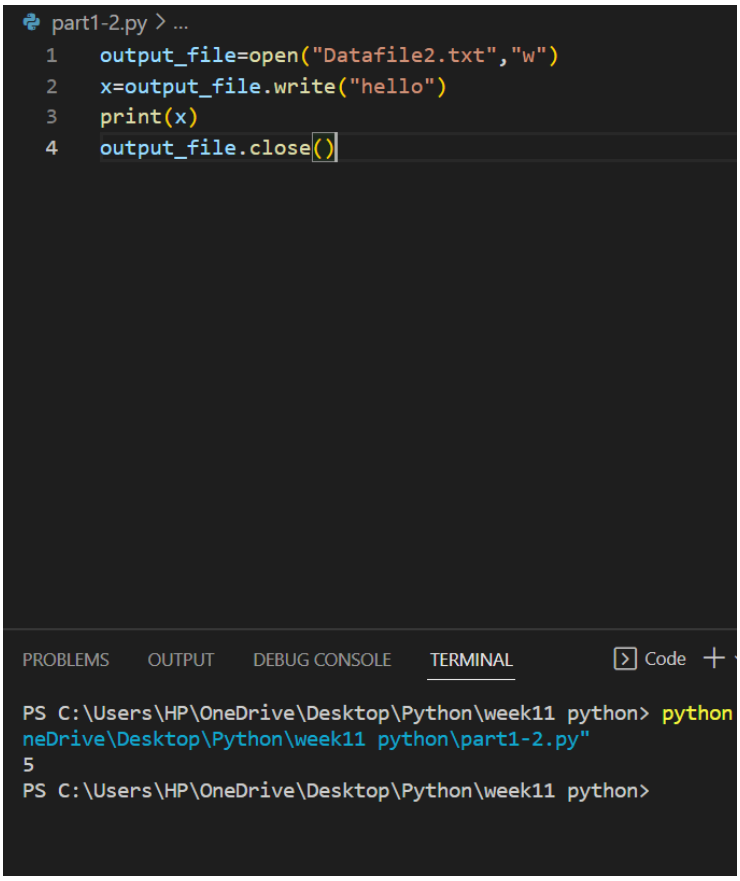
**Python Workshop 11: File Handling**

# Part 1

1. Create a program in Python that opens a file named `'datafile.txt'` for reading and assigns identifier `input_file` to the file object created.



2. Create a program in Python that opens a file named `'datafile2.txt'` for writing and assigns identifier `output_file` to the file object created.




3. Assume that `input_file` is a file object for a text file open for reading, and `output_file` is a file object for a text file open for writing. Explain the contents of the output after the following code terminates:

```
empty_str = ''
line = input_file.readline()
while line != empty_str:
    output_file.write(line + '\n'
        line = input file.readline()
```

```
t1-3.py > ...
empty_str=""
line=input_file.readline()
while line != empty_str:
    output_file.write(line+"\n")
```

4. Identify the error in the following code:

```
input_file_opened = False
while not input_file_opened:
try:
    file_name = input('Enter file name: ')
    input_file = open(file_name, 'r')
    input_file_opened = True
except: print('Input file not found')
```

=The given code does not close the opened file so here is the code without errors.

```
input_file_opened=False
while not input_file_opened:
    try:
        file_name=input("Enter file name: ")
        input_file=open(file_name,"r")
        print(input_file.read())
        input_file.close()
        input_file_opened=True
    except:
        print("File not found")
```

# Part 2

1. Write a Python function called `reduce_spaces` that is given a line read from a text file and returns the line with all extra space characters removed:

'This line has extra space characters' → 'This line has extra space characters'

```python
try:
    f=open("123.txt","r")
    x=f.read()
    y=x.split(" ")
    print(y)
    list_1=[]
    for i in y:
        list_1.append(i)

    while ("") in list_1:
        list_1.remove("")
    print(list_1)
    stri=""
    for j in list_1:
        stri=stri+j+" "

    print(stri)
except:
    print("Invalid name")
```

```
PROBLEMS 3  OUTPUT  DEBUG CONSOLE  TERMINAL  Code

PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "c:
neDrive\Desktop\Python\week11 python\part2-1.py"
['This', 'line', 'has', 'extra', 'space', 'characters\n\n']
['This', 'line', 'has', 'extra', 'space', 'characters\n\n']
This line has extra space characters
```

2. Write a Python function named `extract_temp` that is given a line read from a text file and displays the one number (integer) found in the string:

'The high today will be 75 degrees' → 75.

```python
try:
    f=open("123.txt","r")
    y=f.read()
    f.close()
except:
    print("Invalid filename")

def extract_temp(line):
    x=y.split()
    for i in x:
        if (i.isdigit()):
            print(i)
        else:
            pass

extract_temp(y)
```

```
PROBLEMS 3  OUTPUT  DEBUG CONSOLE  TERMINAL  Code

PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "
neDrive\Desktop\Python\week11 python\part2-2.py"
75
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```

3. Write a Python function named `check_quotes` that is given a line read from a text file and returns True if each quote characters in the line has a matching quote (of the same type), otherwise returns False.

'Today's high temperature will be 75 degrees' → False

```python
try:
    f=open("123.txt","r")
    x=f.read()
    f.close()
    print(x)
except:
    print("Invalid file name")

def check_quotes(line):
    quote="Today's high temperature will be 75 degrees"
    x1=quote.split(" ")
    list_1=[]
    for i in x1:
        list_1.append(i)
    list_2=[]
    x2=x.split(" ")
    for j in x2:
        list_2.append(j)
    set1=set(list_1)
    set2=set(list_2)
    z=set1.issubset(set2)
    return z
```

```
PROBLEMS 3  OUTPUT  DEBUG CONSOLE  TERMINAL  Code

neDrive\Desktop\Python\week11 python\part2-3.py"
Today's high temperature will be 75 degrees
```

4. Write a Python function named `count_letters` that is given a line read from a text file and returns a list containing every letter in the line and the number of times that each letter appears (with upper/lower case letters counted together)

'This is a line' → [ ('t', 1), ('h', 1), ('i', 3), ('s', 2), ('a', 1), ('l', 1), ('n', 1), ('e', 1) ]

```python
try:
    f=open("123.txt")
    x=f.read()
    f.close()
except:
    print("Invalid Filename")

def count_letters(lines):
    l1=[]
    line=(lines.lower()).split(" ")
    for i in line:
        for k in i:
            l1.append(k)
    l2=[]
    for j in l1:
        y1=str(l1.count(j))
        y2=str((j,y1)).strip("")    ##
        y3=j.replace(j,y2)  #or y3=j.replace(j,("("+"\'"+j+"\'"+", "+y1+")"))

        l2.append(y3)
    l3=[]
    l3=list(dict.fromkeys(l2))
    l4=str(l3).replace('"',"")
    print(l4)
count_letters(x)
```

```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "c:\Users\HP\OneDrive\Desktop\Pytho
n\week11 python\part2-4.py"
[('t', '1'), ('h', '1'), ('i', '3'), ('s', '2'), ('a', '1'), ('l', '1'), ('n', '1'), ('e', '1'), ('
\\n', '2')]
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```

5. Write a Python function named `interleave_chars` that is given two lines read from a text, and returns a single string containing the characters of each string interleaved: 'Hello', 'Goodbye' → 'HGeololdobye'

```
part2-5.py > ...
1   try:
2       f=open("123.txt")
3       a=f.readline()
4       b=f.readline()
5       f.close()
6   except:
7       print("Invalid filename")
8
9   def interleave_chars(line1,line2):
10      a1=[]
11      b1=[]
12      for i in line1.strip("\n"):
13          a1.append(i)
14      a1.append("")
15      a1.append("")
16      for j in line2.strip("\n"):
17          b1.append(j)
18      for k in range(0,len(b1)):
19          z=a1[k]+b1[k]
20          print(z,end="")
21  interleave_chars(a,b)
```

```
PROBLEMS 3   OUTPUT   TERMINAL   ...   Code

PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python
\Users\HP\OneDrive\Desktop\Python\week11 python\part2-5.py"
'H'e'llo
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```

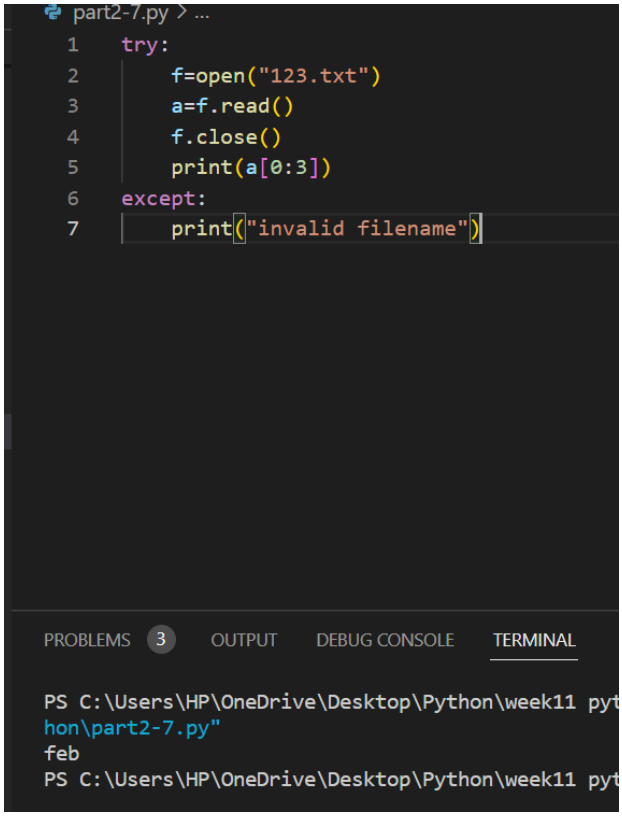
6. Give a for loop that counts all the characters in a string assigned to variable `line`, except blanks and the newline character.

```python
try:
    f=open("123.txt")
    x=[]
    for i in f:
        y1=i.strip("\n")
        for j in y1:
            x.append(j)
    while " " in x:
        x.remove(" ")
    f.close()
    l2=[]
    for i in x:
        count_1=(x.count(i))
        l2.append((i,count_1))
    l3=list(dict.fromkeys(l2))
    print(l3)
except:
    print("invalid filename")
```

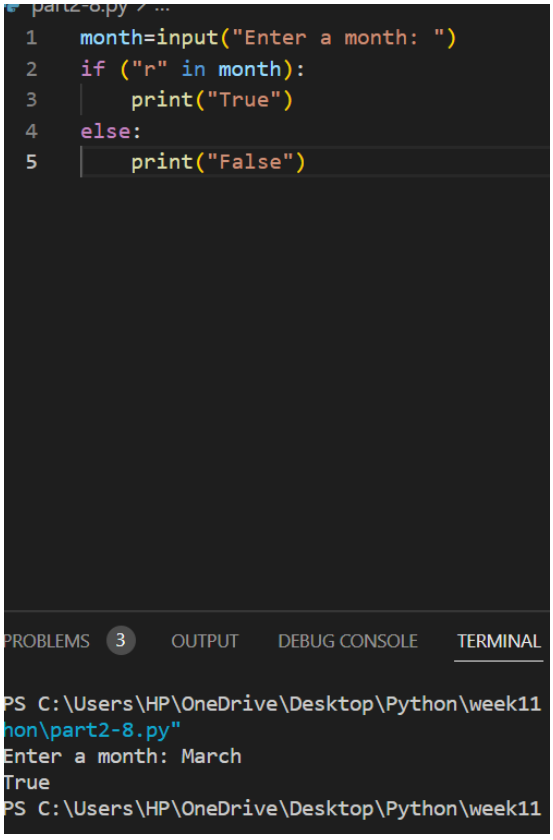
```
PROBLEMS 3  OUTPUT  DEBUG CONSOLE  TERMINAL                    Code

PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "c:\Users\HP\OneDrive\Desktop\Python\week
hon\part2-6.py"
[("'", 6), ('A', 1), ('k', 1), ('r', 1), ('i', 2), ('t', 1), ('D', 1), ('e', 1), ('v', 1)]
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```

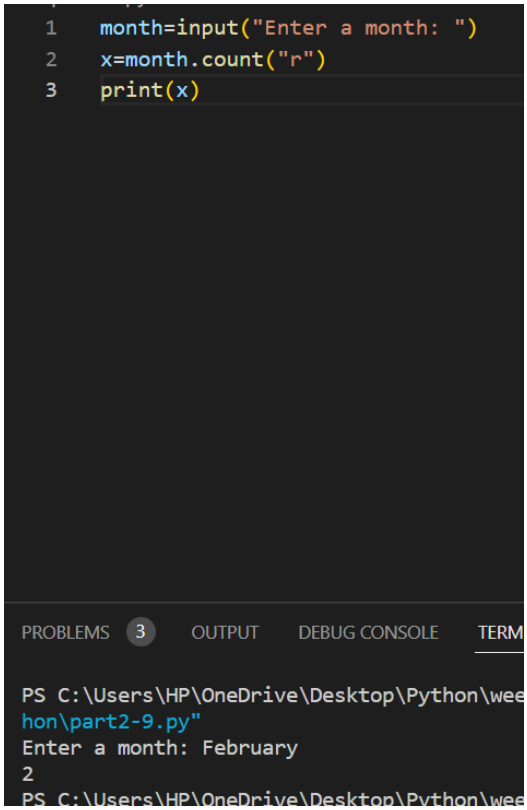
7. For variable `month` which contains the full name of any given month, give an expression to display just the first three letters of the month.



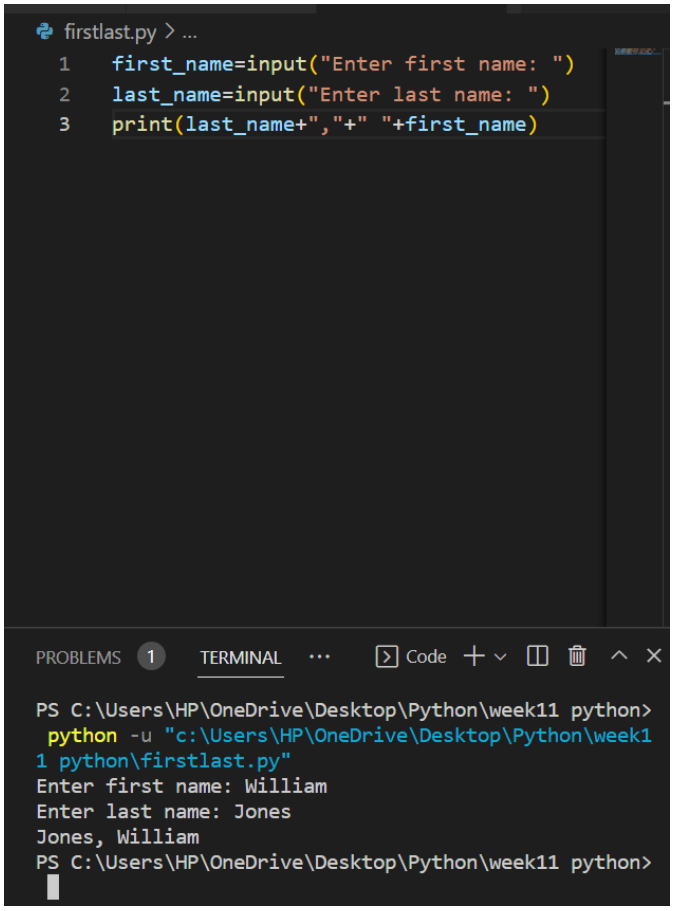
8. Give an expression that displays `True` if the letter ‘r’ appears in a given `month` name stored in variable month, otherwise displays `False`.



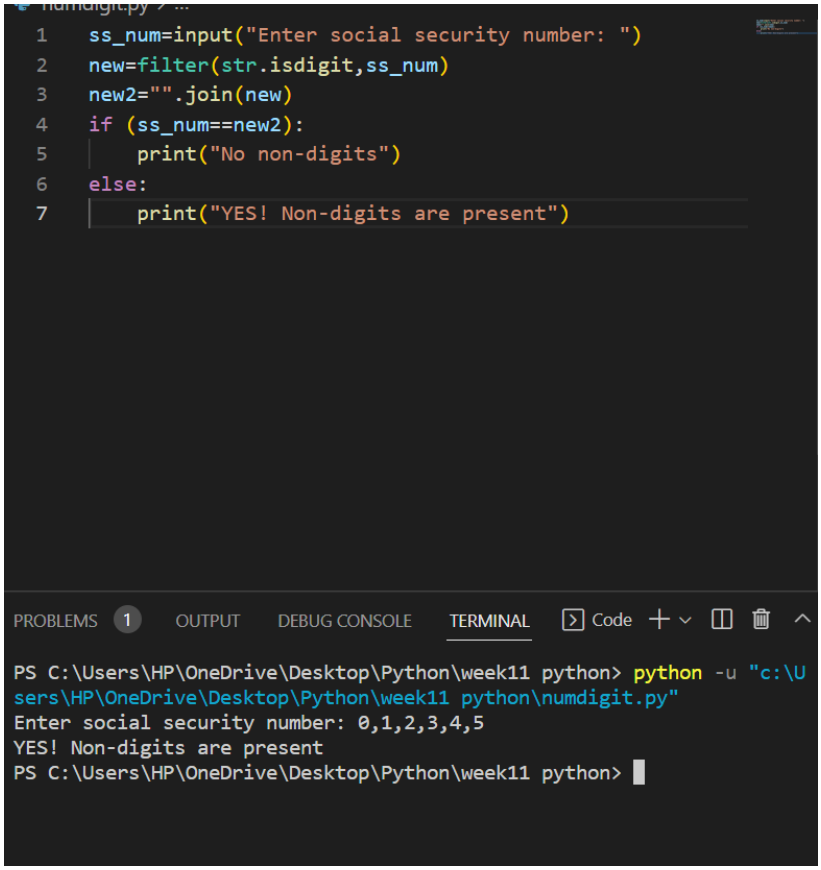
9. Give an expression for determining how many times the letter ‘r’ appears in a given month name stored in variable `month`.



10. For a person's first name stored in variable `first_name`, and last name stored in variable `last_name`, give an expression that displays the person's name formatted exactly as follows: `Jones, William`.



11. Give an instruction that determines if a given social security number represented as a string and stored in variable `ss_num`, contains any non- digits.

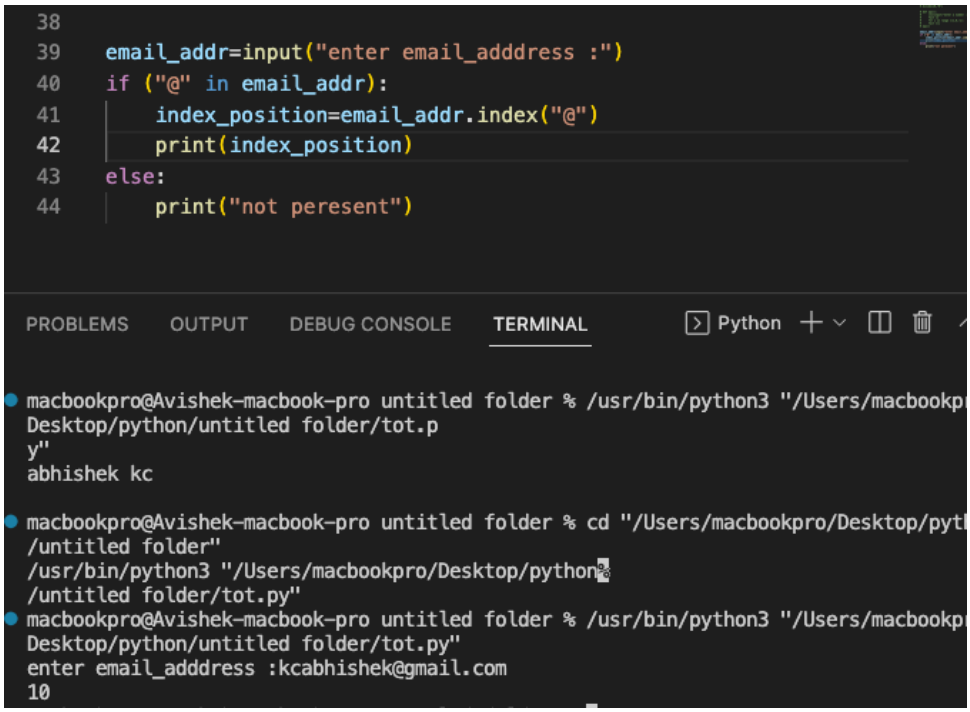


```
1  ss_num=input("Enter social security number: ")
2  new=filter(str.isdigit,ss_num)
3  new2="".join(new)
4  if (ss_num==new2):
5      print("No non-digits")
6  else:
7      print("YES! Non-digits are present")
```

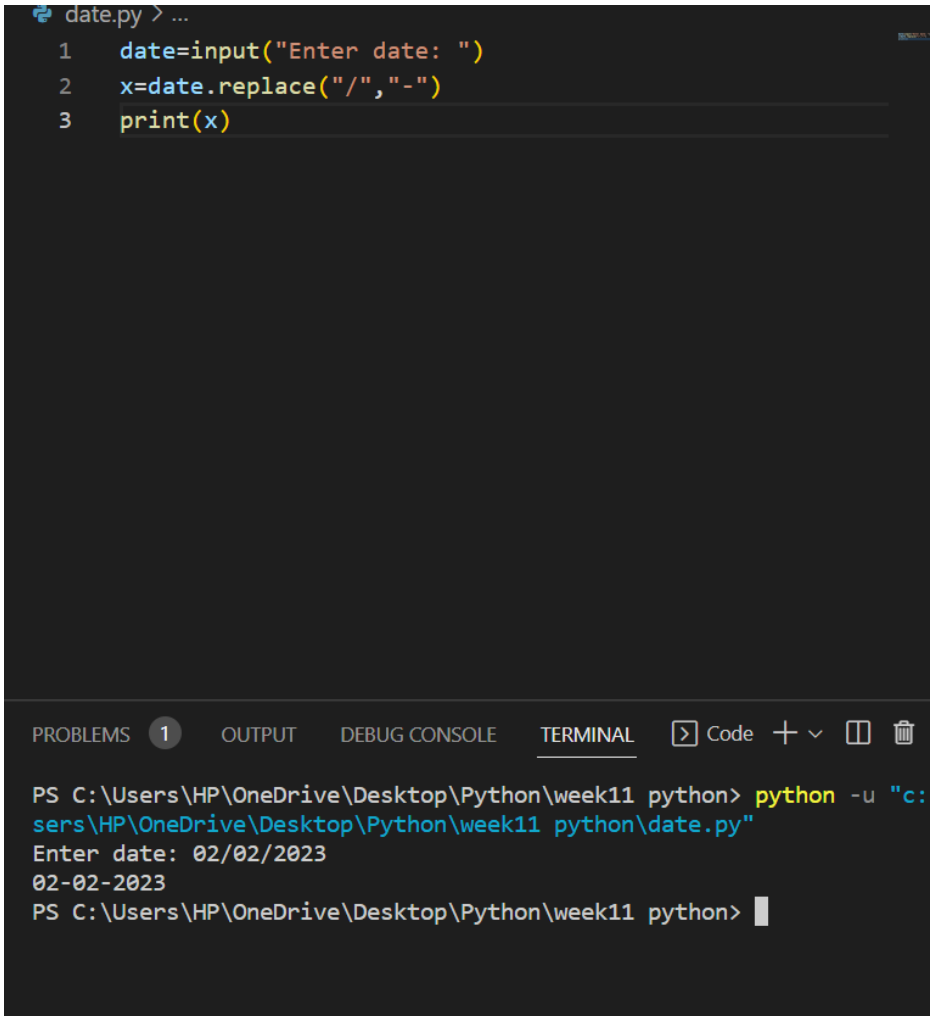
```
PROBLEMS 1  OUTPUT  DEBUG CONSOLE  TERMINAL  Code

PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "c:\U
sers\HP\OneDrive\Desktop\Python\week11 python\numdigit.py"
Enter social security number: 0,1,2,3,4,5
YES! Non-digits are present
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```

12. Give an instruction that determines the index of the '@' character in an email address stored in variable `email_addr`.



13. For a variable named `date` containing a date in the form 12/14/2012, give an expression that replaces all slashes characters with dashes.



14. For a variable named `err_mesg` that contains error messages in the form ** error message **, give an expression that produces a string containing the error message without the leading and trailing asterisks and blank characters.

```
err_mesg=input("Enter error message: ")
x=err_mesg.strip("*")
y=x.strip(" ")
print(y)
```

## Part 3

1. Write a program that opens and reads a text file and displays how many lines of text are in the file.

```python
try:
    f=open("123.txt")
    #x=f.read()
    count_lines=0
    l1=[]
    for i in f:
        l1.append(i)

    print(len(l1))

except:
    print("Invalid filename")
```

```
PROBLEMS 1  TERMINAL  ...  Code

PS C:\Users\HP\OneDrive\Desktop\Python\week11 pyt
hon> python -u "c:\Users\HP\OneDrive\Desktop\Pyth
on\week11 python\part3.py"
6
PS C:\Users\HP\OneDrive\Desktop\Python\week11 pyt
hon>
```

2. Write a program that reads a text file named `original_text`, and writes every other line, starting with the first line, to a new file named `new_text`.

```
try:
    f=open("original_text.txt")
    f1=open("new_text.txt","w")
    for i in f:
        f1.write(i)
    f.close()
    f1.close()
except:
    print("Invalid filename")
```

3. Write a program that reads a text file named `original_text`, and counts how many time the letter 'e' occurs (the most frequently occurring letter in English), and displays how many occurrences there are.

```python
try:
    f=open("original_text.txt")
    l=[]
    for i in f:
        l.append(i)
    l1=str(l).strip("\n")
    count=0
    for j in l1:
        for k in j:
            if (k=="e"):
                count=count+1
    f.close()
    print("The number of times e is present in the file is: "+str(count))
except:
    print("Invalid filename")
```

```
PROBLEMS 1  OUTPUT  DEBUG CONSOLE  TERMINAL

PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "c:\Users\HP\OneDrive\De
1 python\part3-3.py"
The number of times e is present in the file is: 2
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```

4. Write a program that reads a text file containing numerical expressions on each line and print them out along with the results. For example, for the numerical expression `4 + 2` in your file, your program should output: `4 + 2 = 6`.

part3-4.py > ...

```python
try:
    f=open("original_text.txt")
    l=[]
    for i in f:
        for j in i:
            l.append(j)
    summ=int(l[0])+int(l[2])
    print(summ)

except:
    print("Invalid filename")
```

PROBLEMS 1 OUTPUT TERMINAL ... Code

```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> pyt
hon -u "c:\Users\HP\OneDrive\Desktop\Python\week11 python
\part3-4.py"
6
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```

## Part 4 (Optional)

Write a Python program that encrypts and decrypts text files using a substitution cipher. Your program should ask the user for the name of a text file and whether they would like to encrypt or decrypt. Once the process is complete, you should write the output to a new text file with a modified name:

```
This program will encrypt and decrypt text files

Enter (e) to encrypt a password, and (d) to decrypt: e
Enter the name of a text file to encrypt: hello.txt

Output written to: encrypted_hello.txt
```

Your program should catch exceptions and print helpful error messages. You should use your solution to Coding Challenge 03 to help you.