CIND830 - Python Programming for Data Science

Assignment 1 (15% of the final grade)

Due on Oct 24, 2022 11:59 PM

This is a Jupyter Notebook document that extends a simple formatting syntax for authoring HTML and PDF. Review this website for more details on using Jupyter Notebooks.

Consider using a Jupyter Notebook platform to complete this assignment. Ensure using **Python 3.7 release or higher** then complete the assignment by inserting your Python code wherever seeing the string `#INSERT YOUR ANSWER HERE`.

You are expected to submit the notebook file (in IPYNB format) and the exported version (either in PDF or HTML) in the same Assignment link in D2L. Use these guidelines to submit **both** the IPYNB and the exported file (HTML or PDF). Failing to submit both files will be subject to mark deduction.

Please be advised that you cannot get more than 100% in this assignment, and the **BONUS** question (if there is any) will only be graded if all other questions have been submitted.

Coverage:

1. Data Types and Expressions
2. Repetition Statements
3. Selection Statements
4. Strings and Text Files

Question 1 **[30 pts]**:

a) [10 pts] Write a Python program to find the factorial of an integer. For example, if the user enters `5`, the output should be `120`.

```
num = int(input(" Please enter any Number : "))
factorial = 1
if num < 0:
    print("Factorial does not exist for negative numbers")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    for i in range(1, num + 1):
        # multiply factorial by current number
        factorial = factorial * i
    print("The factorial of", num, "is", factorial)
```

```
 Please enter any Number : 5
The factorial of 5 is 120
```

b) [10 pts] Write a Python program to check whether an integer is a prime number or not. For example, if the user enters `8`, the output should be `The integer 8 is not a prime number.`

```
# To take input from the user
num = int(input("Enter a number: "))

# prime numbers are greater than 1
if num > 1:
   # check for factors
   for i in range(2,num):
       if (num % i) == 0:
           print("The integer",num,"is not a prime number")
           break
   else:
       print("The integer",num,"is a prime number")

# if input number is less than
# or equal to 1, it is not prime
else:
   print("The integer",num,"is not a prime number")
```

```
Enter a number: 8
The integer 8 is not a prime number
```

c) [10 pts] Write a Python program to accept an integer number and dispay the digits at the ones, tens and hundreds place of that number.
For example, if the user enters `86421` then the output should be:

```
The digit at the ones place of 86421 is 1
The digit at the tens place of 86421 is 2
The digit at the hundreds place of 86421 is 4
```

```
num = int(input("Enter a number: "))

hundreds = (num % 1000) // 100
tens = (num % 100) // 10
```

```
ones = (num % 10)
print("The digit at the ones place of %d  is %d" %(num, ones))
print("The digit at the tens place of %d  is %d" %(num, tens))
print("The digit at the hundreds place of %d  is %d" %(num, hundreds))
```

```
Enter a number: 86421
The digit at the ones place of 86421  is 1
The digit at the tens place of 86421  is 2
The digit at the hundreds place of 86421  is 4
```

BONUS [10 pts] Write a Python program to display the running total of a sequence of numbers. For example, if the user enters `2,3,4,6,10`, the output should be `2 5 9 15 25`

```
numbers·=·list(map(int,·input("\nEnter·the·numbers·:·").strip().split("·")))·#·numbers·split·
runningTotal·=·[]
total·=·0
for·number·in·numbers:
····total·+=·number
····runningTotal.append(total)

print(runningTotal)
```

```
Enter the numbers : 2 3 4 6 10
[2, 5, 9, 15, 25]
```

Question 2 **[30 pts]**:

a) [10 pts] Write a Python program that concatenates the first three and the last three characters from a given string and displays the result. The program returns an empty string if the given string length is less than 3.

```
Sample String: `DataScience`
Expected Result : `Datnce`

Sample String: `830`
Expected Result : `830830`

Sample String: `IT`
Expected Result : ''
```

```
my_string = input("Enter a string ")
if len(my_string) > 2:
    print(my_string[:3]+my_string[-3:])
else:
    print('') # 'empty string'
```

```
Enter a string DataScience
Datnce
```

b) [10 pts] Write a Python program to return the sum and average of the numbers that appear in a given string after rounding them up to the nearest thousandth. You might use the `isdigit()` string method to check if a character is a digit or not.

```
Sample String: TOronTo29!@#8496
Expected Result : Sum is: 38 Average is: 6.333
```

```
String = input("Enter the String:")
counter=0
sum =0
for character in String:
    if (character.isdigit()): # if character is digit, character.isdigit==True then enter in
        sum+=int(character) # change the data type of character from char to int for calculat
        counter+=1 # increment
        sum_round = round(sum,3)
average=round((sum/counter),3)
print("Sum is:",(sum_round), "Average is:",average)
```

```
Enter the String:TOronTo29!@#8496
Sum is: 38 Average is: 6.333
```

c) [10 pts] Write a Python program to find words with alphabets and numbers from a given string. You might use the built-in function `any()` with the combination of the `isalpha()` and `isdigit()` string methods.

```
Sample String: John35 is taking CIND830 in Data Analytics 2022Fall semester
Expected Result:  John35, CIND830, 2022Fall
```

```
# Words with both alphabets and numbers -Using isdigit() + isalpha() + any()
string = input("Enter the String:")
output = []
temp = string.split()
for idx in temp:
    if any(chr.isalpha() for chr in idx) and any(chr.isdigit() for chr in idx):
        output.append(idx)
# printing result
print("Words with alphabets and numbers : " + str(output))
```

```
Enter the String:John35 is taking CIND830 in Data Analytics 2022Fall semester
Words with alphabets and numbers : ['John35', 'CIND830', '2022Fall']
```

Question 3 **[40 pts]**:

The following text is an example of a fake payment receipt, scammers use to encourage recipients to engage.

```
From: robert lor <robert6388lor@gmail.com>
Subject: PAYMENT DONE______________NYR7755269R
Date: September 9, 2021 at 10:31:42 AM CDT
To: consumer233@gmail.com

___OUR valued consumer___

thank you for your purchase through NORTON . this email that is to inform you your annual subscripti

Here is an overview of your recent purchase :-

PRODUCT INFO
invoice no :- NYR7755269R
ORDER date :- 09/09/2021
PAYMENT method : - auto – debited

VALUE :- $321.67

to upgrade /cancel your subscription , PLEASE contact our customer service at
+1 (810)-(515)-(7954)

Thank you,
Norton billing support
```

a) [10 pts] Save the text to a file called `emailSpam.txt`, then compute and display the total number of lines: `23` and words: `91`.

```
textFile = open("/content/emailSpam.txt", "w") #Opening a file for writing
textFile.writelines("""From: robert lor <robert6388lor@gmail.com>
Subject: PAYMENT DONE______________NYR7755269R
Date: September 9, 2021 at 10:31:42 AM CDT
To: consumer233@gmail.com

___OUR valued consumer___

thank you for your purchase through NORTON . this email that is to inform you your annual sub

Here is an overview of your recent purchase :-

PRODUCT INFO
invoice no :- NYR7755269R
ORDER date :- 09/09/2021
PAYMENT method : - auto – debited

VALUE :- $321.67

to upgrade /cancel your subscription , PLEASE contact our customer service at
+1 (810)-(515)-(7954)

Thank you,
Norton billing support""")
textFile.close()

textFile = open("/content/emailSpam.txt", "r")
contentFile = textFile.readlines()
lineNumbers = 0
for line in contentFile:
  lineNumbers += 1
wordNumbers = 0
for line in contentFile:
  words = len(line.split())
  wordNumbers += words
print("The number of lines in the emailSpam.txt file is:",lineNumbers)
print("The number of words in the emailSpam.txt file is:",wordNumbers)
textFile.close()
```

```
     The number of lines in the emailSpam.txt file is: 23
     The number of words in the emailSpam.txt file is: 91
```

b) [10 pts] Find and display the email subject and the number of the upper and lower cases in the subject text.

```
textFile = open("/content/emailSpam.txt", "r")
contentFile = textFile.readlines()
```

```
subjectLine = ()
for line in contentFile:
  if (line.startswith("Subject:") and line.endswith("\n")):
    subjectLine = line.replace('Subject:','')
lower = 0
upper = 0
for i in subjectLine:
  if (i.islower()):
    lower += 1
  if (i.isupper()):
    upper += 1
print(subjectLine)
print("Number of lower cases characters in Subject line: ", lower)
print("Number of upper cases characters in Subject line: ", upper)
textFile.close()
```

```
      PAYMENT DONE______________NYR7755269R

     Number of lower cases characters in Subject line:  0
     Number of upper cases characters in Subject line:  15
```

c) [10 pts] Define a list that consists of 10 different email spam trigger words, such as `RENEWED` or `purchase`, then check whether any of these words exist in the body of the email. If the email contains a spam word, print it; otherwise, return a message that the email does not include any predefined spam words.

```
textFile = open("/content/emailSpam.txt", "r")
contentFile = textFile.readlines()
words = []
spam = ['Money','Earn', 'Cheap', 'Save', 'Free', 'Cash', 'Prize', 'Vacation', 'debited']
spam_words = []
for line in contentFile:
  if ((line.startswith("From:") and line.endswith("\n")) or line.startswith("Subject:") and l
    pass
  else:
    #print(line)
    word = line.split()
    #print(word)
    words.extend(word)
#print(words)
for word in words:
  if word in spam:
    spam_words.append(word)
if len(spam_words) == 0:
  print("The email does not include any predefined spam words")
else:
  print("The body of the email contains the predefined spam words are: \n", spam_words)
textFile.close()
```

```
The body of the email contains the predefined spam words are:
 ['debited']
```

d) [10 pts] Find and display the percentage of spam words found in the body of the email using the list of spam trigger words from part c.

```
textFile = open("/content/emailSpam.txt", "r")
contentFile = textFile.readlines()
words = []
spam = ['Money','Earn', 'Cheap', 'Save', 'Free', 'Cash', 'Prize', 'Vacation', 'debited']
spam_words = []
for line in contentFile:
  if ((line.startswith("From:") and line.endswith("\n")) or line.startswith("Subject:") and l
    pass
  else:
    #print(line)
    word = line.split()
    #print(word)
    words.extend(word)
#print(words)
for word in words:
  if word in spam:
    spam_words.append(word)
if len(spam_words) == 0:
  print("The email does not include any predefined spam words")
else:
 print("The percentage of spam words found in the body of the email:",round((len(spam_words)/
#print("Number of spam words in emailSpam.txt file are:", len(spam_words))
#print("Number of words in the body of emailSpam.txt file is:",len(words))
textFile.close()
```

```
The percentage of spam words found in the body of the email: 1.35 %
```

This is the end of assignment 1