Exercise - 7

Functions

Question 1

1. Suppose the Ganga retail store management wants to provide discount for all bill amounts as mentioned below. Assume bill amount will be always greater than 0.

Regular Customer

Bill Amount	Discount %
>=1000	5
>=500 and <1000	2
>0 and <500	1

New Customer

Bill Amount	Discount %
>=1000	3
>=500 and <1000	1
>0 and <500	.5

Write a function which accept Bill amount and Type of customer and generate the Discount for the given customers.

```
def bill(amt,typ):
  if typ==1:
    if amt>=1000:
      disc = 0.05*amt
    elif (amt>=500 and amt<1000):
      disc=0.02*amt
    else :
      disc=0.01*amt
  elif typ==2:
    if amt>=1000:
      disc = 0.03*amt
    elif (amt>=500 and amt<1000):
      disc=0.01*amt
    else :
      disc=0.005*amt
  else : print("None")
```

```
  print('\nBill generated successfully......\n')
  print(f'Amount to be paid = Rs.{amt - disc}')

amt = float(input("Enter the amount : "))
typ = int(input("1.Regular \t 2.New : "))
bill(amt,typ)
#print("Bill amount = ",(amt - disc))
```

```
Enter the amount : 505
1.Regular        2.New : 1

Bill generated successfully......

Amount to be paid = Rs.494.9
```

Question 2

2. Write a function called calculator. It should take the following parameters: two numbers, an arithmetic operation (which can be addition, subtraction, multiplication or division and is addition by default), and an output format (which can be integer or floating point, and is floating point by default). Division should be floating-point division.
   The function should perform the requested operation on the two input numbers, and return a result.

```
def calculator(a,b):
  if typ==1:   return (a+b)
  elif typ==2: return (a-b)
  elif typ==3: return (a*b)
  elif typ==4: return (a/b)
  else : return 0
a,b=eval(input('Enter 2 numbers : '))
typ = int(input('Select operation to be performed : \n 1.Addition 2.Subtraction 3.Multiplicat
res = calculator(a,b)
print(f'Result = {res}')
```

```
Enter 2 numbers : 10,20
Select operation to be performed :
 1.Addition 2.Subtraction 3.Multiplication 4.Division3
Result = 200
```

Question 3

3. Create dictionary users and store N users' details such as user name and password. Write a function called acceptlogin (username, password) with two parameters. The function should

return true if the user exists and the password is correct and false otherwise.

Input and output:
Enter the number of users: 2
Enter user name: raja
Enter user password: 123
Enter user name: Babu
Enter user password: 123
Enter the user name to check raja
Enter the password123
Correct

```
def check(key_id,key_pwd):
  if ((key_id,key_pwd) in d.items()):
    print("Correct")
  else :
    print("Invalid")

id = [id for id in input("Enter ID : ").split(',')]
pwd = [pwd for pwd in input("Enter Password : ").split(',') ]
d = dict(zip(id,pwd))
key_id = input("Enter user name to check : ")
key_pwd = input("Enter password : ")
check(key_id,key_pwd)
```

```
Enter ID : raja,babu
Enter Password : 1234,5567
Enter user name to check : raja
Enter password : 1234
Correct
```

Question 4

5. If a five-digit number is input through the keyboard, write a function to print a new number by adding one to each of its digits. For example if the number that is input is 12391 then the output should be displayed as 23402.

```
def add(lst):
  for ele in lst :
    print(ele+1,end="")
lst=[int(lst) for lst in input("Enter the 5 digit number : ")]
add(lst)
```

```
Enter the 5 digit number : 12391
```

```
234102
```

Exercsie - 8
File Handling

Question 1

1. Write a python program to read a file of numbers, one in each line and perform the following task.

The count of numbers in the file.
The sum of the number in the file.
The smallest number in the file.
The largest numbers in the file.

```
file = open(r'ip.txt','r')
content=(file.read())
content=content.split('\n')
print("Count of numbers = ",len(content))
s=0
for x in content:
  s+=int(x)
print("Sum of numbers = ",s)
print("Smallest number = ",min(content))
print("Largest number = ",max(content))
```

```
Count of numbers =  10
Sum of numbers =  55
Smallest number =  1
Largest number =  9
```

Question 2

2. Consider the inventory management system includes following details stock number, quantity, and price. Store the inventory details into output.txt file and handle appropriate exceptions for the following:

I. An error message, if the stock number is negative or higher than 999
II. The quantity, if it is less than 0
III. The price, if it is over $100.00

```
file = open(r'output.txt','r')
```

```
file = open(r'op.text','w')
sno = [] ; quan=[] ; price = []
n = int(input("Enter no. of customers : "))
for i in range(n):
  try :
    x = int(input("Enter stock no. : "))
    if (x<0 or x>=999) :
      raise ValueError("Stock no. should be greater than 0")
    else : sno.append(x)
    file.write(f'{str(x)} \t')
    y = int(input("Enter quantity : "))
    if (y<0) :
      raise ValueError("Quantity should be greater than 0")
    else : quan.append(y)
    file.write(f'{str(y)}\t')
    z = int(input("Enter price in $ : "))
    if (z>100) :
      raise ValueError("Price should be greater than 100")
    else :
      price.append(z)
    file.write(f'{str(z)}\n')
  except ValueError as e :
    ("Invalid")
file.close()
```

```
Enter no. of customers : 1
Enter stock no. : 1
Enter quantity : 10
Enter price in $ : 111
```

Question 3

3. The Kannan Electronic store maintains the price details in the Price.txt file. Price.txt file contains three filed such as item name and company of item and price of the item.

```
Price.txt
SmartTV                    Samsung      30000
SmartTV                Sony            80000
Mobile                   Samsung      10000
Mobile                   Nokia           12000
Laptop                   HP                50000
Laptop                   DEL             60000
```

Write a python program to solve the following task.

1. Read the list of items and company name of the items bought by the customer.
2. Compute the bill for the customer and write the bill detail into the Bill.txt file.
3. Generate the exception when the item is invalid. (Item that is not present in price.txt) SAMPLE INPUT:
   Enter the customer name:
   Raja
   Enter the list of item and type item bought by customers:
   SmartTV Samsung, Mobile Nokia, Laptop HP
   SAMPLE OUTPUT:
   Bil.txt
   Name: Raja
   SmartTV Samsung: 30000
   Mobile Nokia: 12000
   Laptop HP: 50000
   Total amount to be paid: 92000

```
#class ItemNotFound(Exception):
#    pass
#if __name__ == '__main__':

    # If i use a function for loading price details then for every new bill
    # we have to do the below calculations. Better way will be to use a class
    # but this method is also good

price_dict = {}  # To create above dictionary
entries = []  # To store the test file entries

with open(r'Q3.txt', 'r') as p:
        temp_entries = [line.rstrip().split('\t') for line in p]
        for item in temp_entries:
            entries.append([word for words in item for word in words.split()])
for entry in entries:
        if entry[0] not in price_dict:
            price_dict.update({entry[0]: {entry[1]: entry[2]}})
        else:
            price_dict[entry[0]][entry[1]] = entry[2]

    # Proceed with tasking customer input , You can use a loop for this
n = int(input("Enter how many customers : ")) ; i=1
while (i<=n):

        cus_name = str(input("Enter The Customer Name: \n"))

        print("Enter the item and type seperated by commas")
        items = list(map(str, input().split(',')))
```

```
    # This will generate a list of lists where each individual
    # list will contain name and type
    # ['Nokia tv', ' Samsung tv'] => [['Nokia tv'],['Samsung tv']]
    items = list(map(lambda x: [x], items))

    # Now name and type is a single string and here they get
    # converted two different items
    # ['Samsung Tv'] => ['Samsung','Tv']
    final_items = []
    for item in items:
        final_items.append(
            [word for words in item for word in words.split()])

    # Calculate total amount and print the bill
    bill_amount = 0
    with open('bill.txt', 'a') as bill:
        bill.write(f"Name : {cus_name}\n")

        for item in final_items:
            price = 0

            try:
                price = int(price_dict[item[0]][item[1]])
            except KeyError:
                pass

            if price:
                bill_amount += int(price)
                bill.write(f"{item[0]} {item[1]} : {price}\n")
            else:
                raise ItemNotFound("Invalid Item entry")

        bill.write(f"Total Amount to be paid : {bill_amount}\n\n")

    print("\nBill Generated Successfully...\n")
    i+=1
```

```
Enter how many customers : 2
Enter The Customer Name:
Ram
Enter the item and type seperated by commas
SmartTV Samsung,Mobile Nokia

Bill Generated Successfully...

Enter The Customer Name:
Sai Ram
Enter the item and type seperated by commas
Laptop HP

Bill Generated Successfully...
```

Question 4

4. HDFC bank keeps the transaction details in the Transcation.txt file. Transcation.txt file contains transaction details of the HDFC bank customers of the month. Transactions.txt file contains four filed such as customer id, customer name, transaction type (D means deposit while W means withdrawal), and transaction amount. Assume the balance amount of all accounts initially is zero.

Transcation.txt

101	Raja	D	10000
102	Elango	D	20000
107	John	D	30000
109	Murugan	D	40000
101	Raja	W	2000
107	John	D	40000
109	Murugan	W	1000

Write a program that computes the balance amount of a bank account based on a transaction log from Transcation.txt and writes the balance amount of each customer into Balance.txt. The output should be written in the following format. Generate an exception when the transaction type is invalid.

Balance.txt

Accountno	name	balance amount
101	raja	8000
102	Elango	20000
107	John	70000
109	Murugan	39000

```
fp1=open("/content/transaction1.txt","r")
fp2=open("/content/balance1.txt","w")
D={}
P=[]
for i in fp1:
 temp=i.split(" ")

 amount=temp[3].replace("\n"," ")

 try:
   if temp[0]+temp[1] in D:
     if (temp[2]=="D"):
       D[temp[0]+temp[1]]=D[temp[0]+temp[1]]+(int(amount))
     elif temp[2]=="W":
       D[temp[0]+temp[1]]=D[temp[0]+temp[1]]-(int(amount))
     else:
       raise ValueError("Invalid Transaction type")
   else:
     D[temp[0]+temp[1]]=(int(amount))
 except ValueError as e1:
```

```
   print(e1)

for j in D:
  print(j,D[j],file=fp2)
fp1.close()
fp2.close()
```

Question 5

5. Write a python program to reverse the contents of an input.txt file and write it to the output.txt file and handle appropriate exceptions for the following.

```
        i) Generate the exception if the file is not found (File Not Found Exceptio
```

```
try:
  f1 = open("output1.txt", "w")
  with open("file.txt", "r") as myfile:
    data = myfile.read()
  data_1 = data[::-1]

  f1.write(data_1)

  f1.close()
except FileNotFoundError : print("No such file")
```

```
    No such file
```