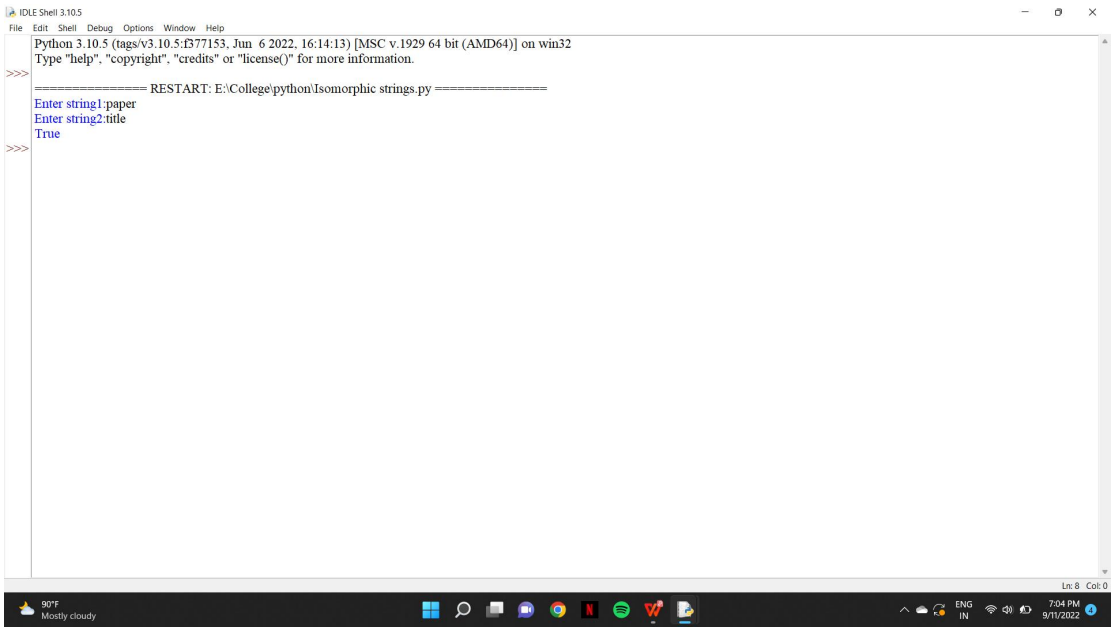
DAY-1 LAB EXPERIMENT

NAME: S.G.DEVSACHIN
SUBJECT CODE: CSA0836

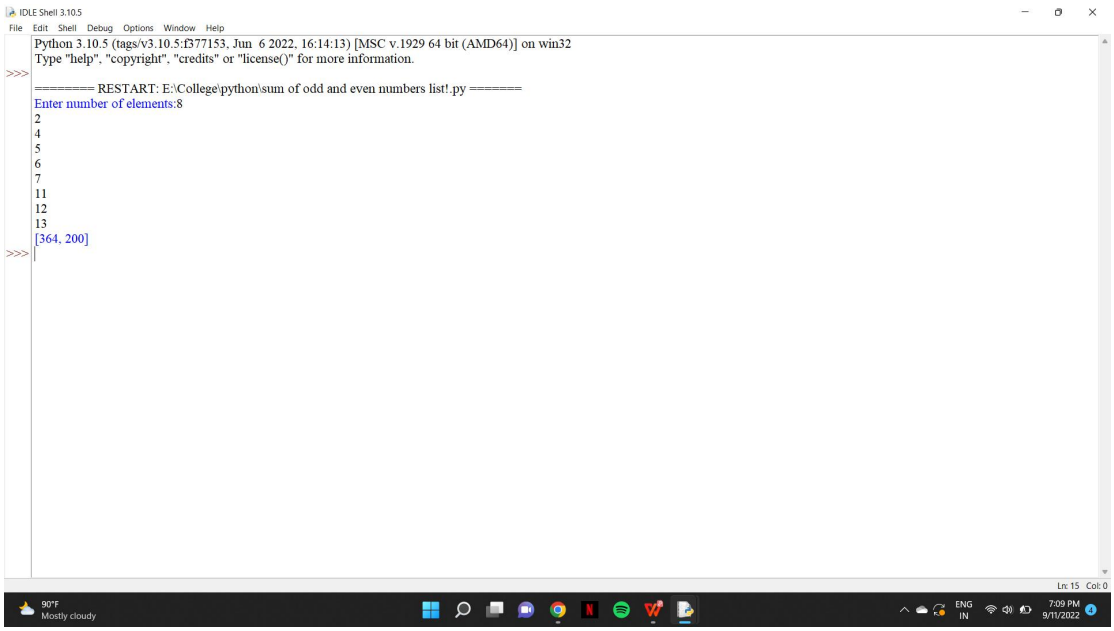
1)
```
def isomorphic(str1,str2):
    if len(str1) != len(str2):
        return False
    else:
        map1,map2={},{}
        for i in range(len(str1)):
            ch1,ch2=str1[i],str2[i]
            if ch1 not in map1:
                map1[ch1]=ch2
            if ch2 not in map2:
                map2[ch2]=ch1
            if  ((map1[ch1] != ch2) or (map2[ch2]!=ch1)):
                return False
    return True
str1=input("Enter string1:")
str2=input("Enter string2:")
print(isomorphic(str1,str2))
```

OUTPUT:



2) l=int(input("Enter number of elements:"))
lst=[]
sum1=0
sum2=0
for i in range(0,l):
    ele=int(input())
    lst.append(ele)
for i in range(0,l):
    if(lst[i]%2==0):
        sum2=sum2+lst[i]**2
    else:
        sum1=sum1+lst[i]**2
l1=[sum1,sum2]
print(l1)

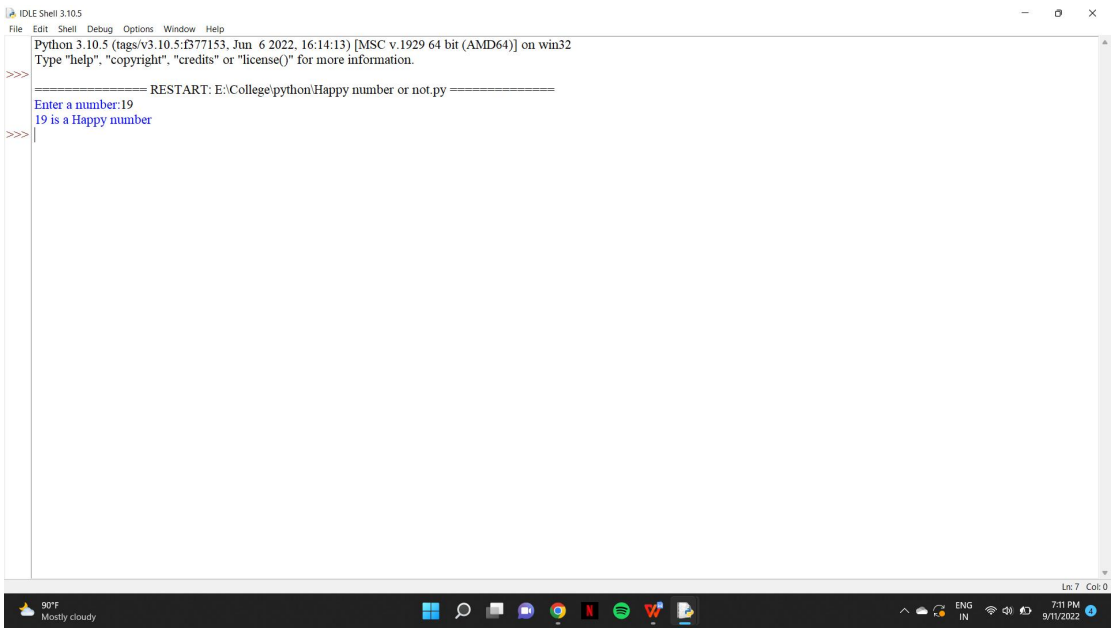
OUTPUT:



3) def numSquareSum(n):
    squareSum = 0
    while(n):
        squareSum += (n%10)*(n%10)
        n=int(n/10)
    return squareSum
def isHappynumber(n):
    slow=n
    fast=n
    while(True):

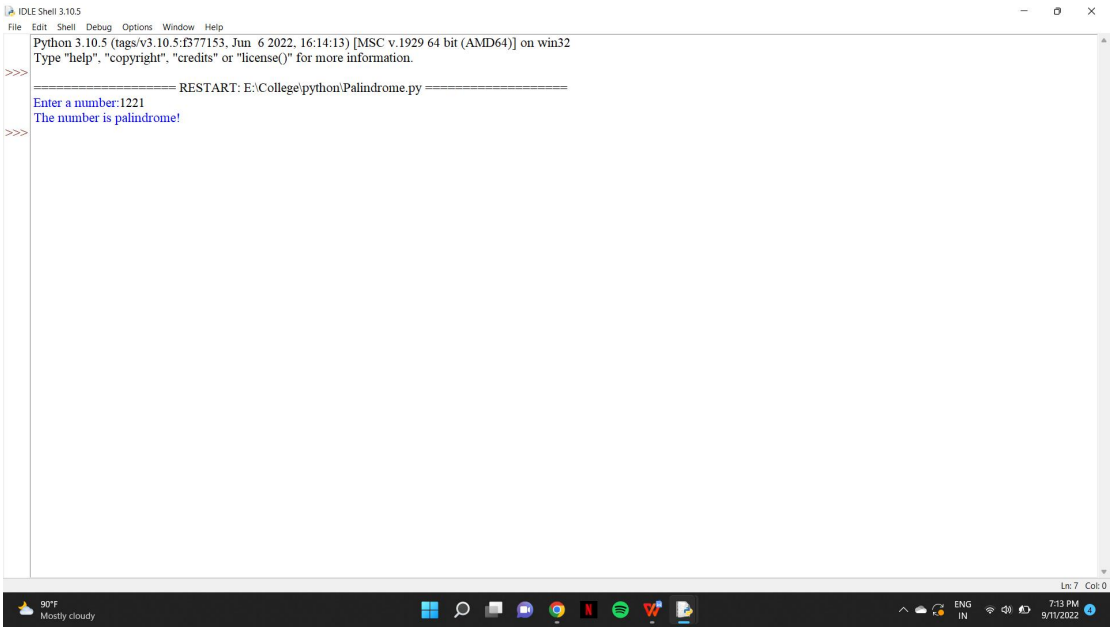
```
        slow=numSquareSum(slow)
        fast=numSquareSum(numSquareSum(fast))
        if(slow!=fast):
            continue
        else:
            break
    return (slow==1)
n =int(input("Enter a number:"))
if (isHappynumber(n)):
    print(n , "is a Happy number")
else:
    print(n , "is not a Happy number")
```

OUTPUT:



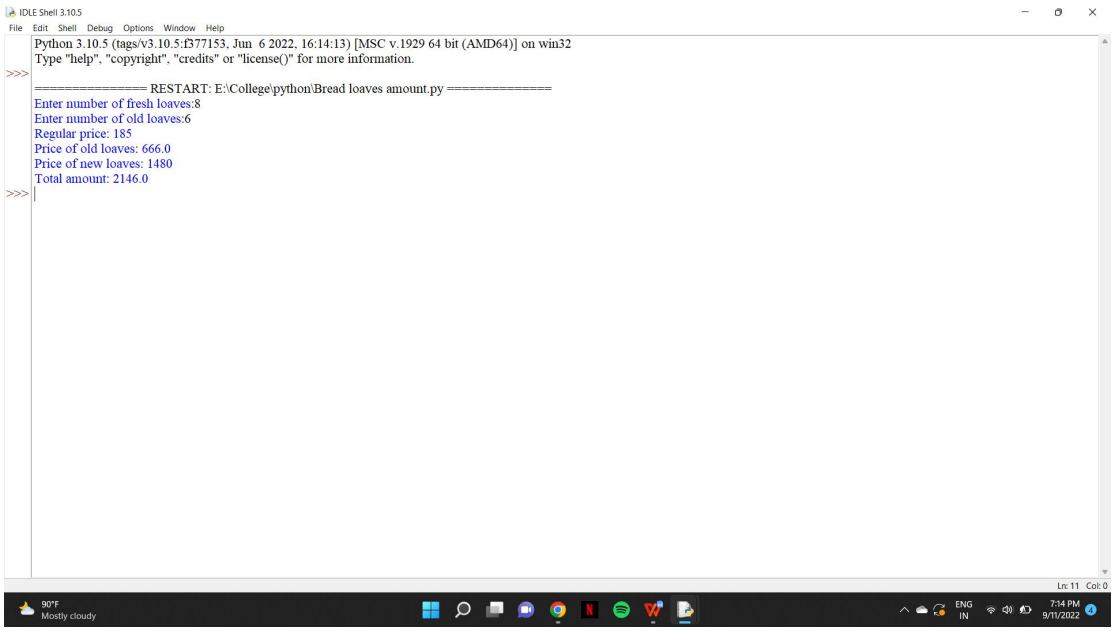

```
4) num=int(input("Enter a number:"))
temp=num
rev=0
while(num>0):
    dig=num%10
    rev=rev*10+dig
    num=num//10
if(temp==rev):
    print("The number is palindrome!")
else:
    print("Not a palindrome!")
```

OUTPUT:



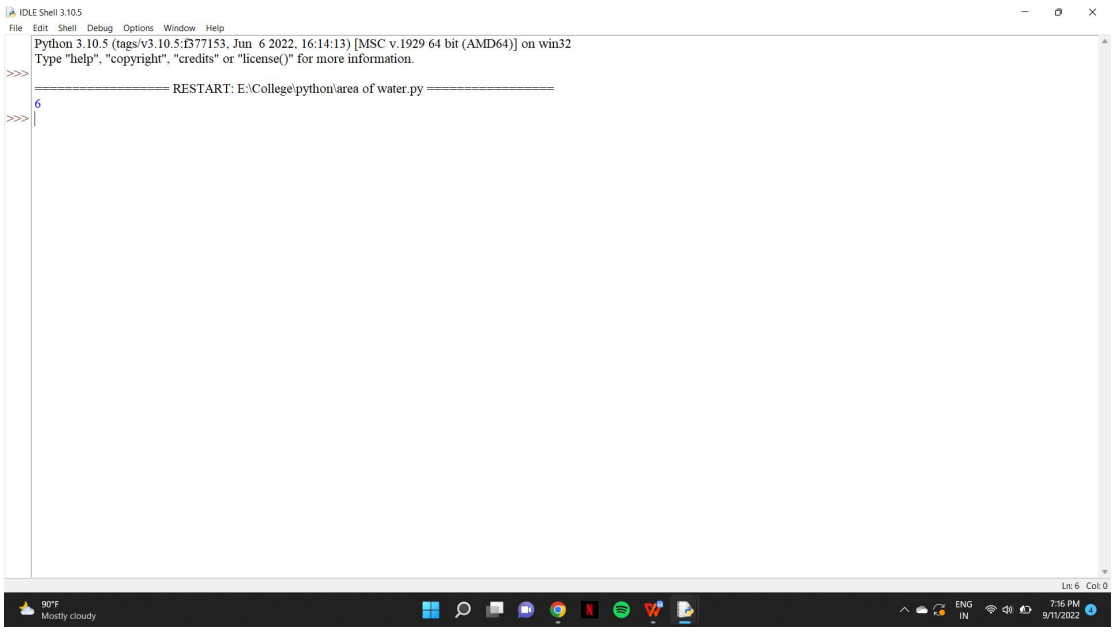
5) a=int(input("Enter number of fresh loaves:"))
b=int(input("Enter number of old loaves:"))
c=185*a
d=185*b*0.6
e=185
total=c+d
print("Regular price:",e)
print("Price of old loaves:",d)
print("Price of new loaves:",c)
print("Total amount:",total)

OUTPUT:



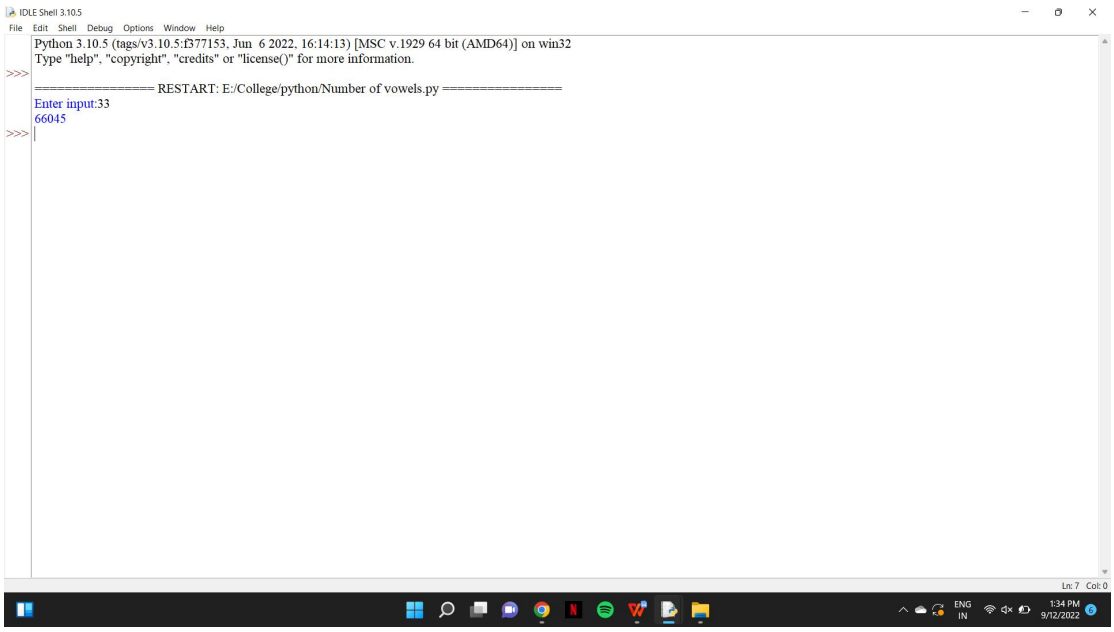
6)
```
def maxArea(A, Len) :
    area = 0
    for i in range(Len) :
        for j in range(i + 1, Len) :

            # Calculating the max area
            area = max(area, min(A[j], A[i]) * (j - i))
    return area

# Driver code
a = [1,5,4,3]

len1 = len(a)
print(maxArea(a, len1))
```

OUTPUT:




```
7) def countstrings(n, start):
        if n == 0:
                return 1
        cnt = 0
        for i in range(start, 5):
                cnt += countstrings(n - 1, i)
        return cnt

def countVowelStrings(n):
        return countstrings(n, 0)

n = int(input("Enter input:"))
print(countVowelStrings(n))
```

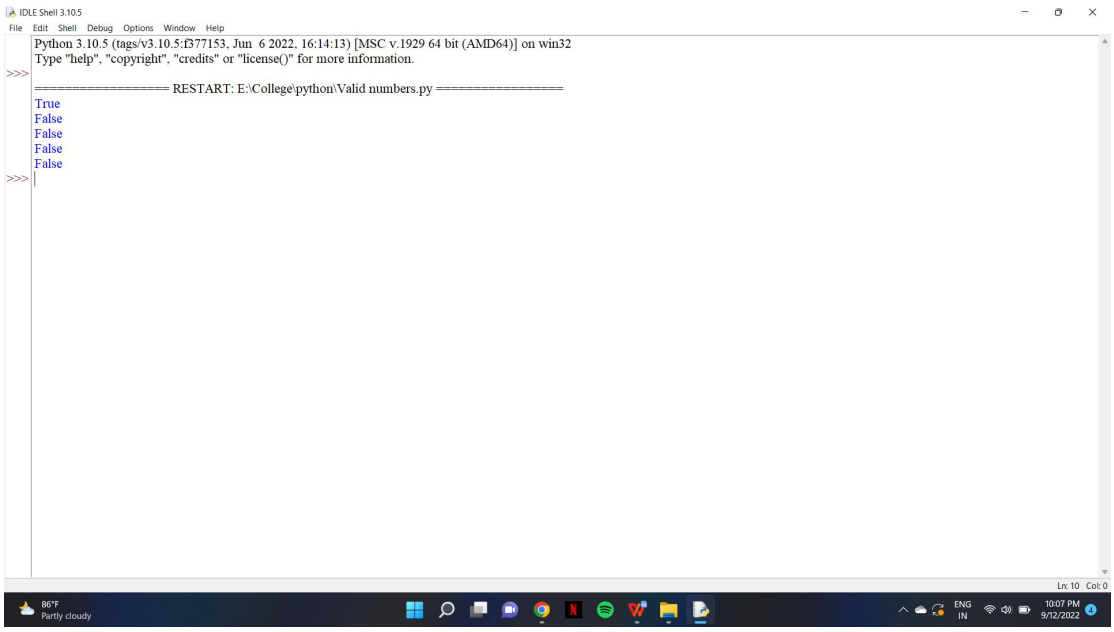
OUTPUT:



8) class Solution(object):
  def isNumber(self, s):
    s = s.strip()
    try:
      s = float(s)
      return True
    except:
      return False

ob = Solution()
print(ob.isNumber("0"))
print(ob.isNumber("e"))
print(ob.isNumber(" "))
print(ob.isNumber("."))
print(ob.isNumber("%"))

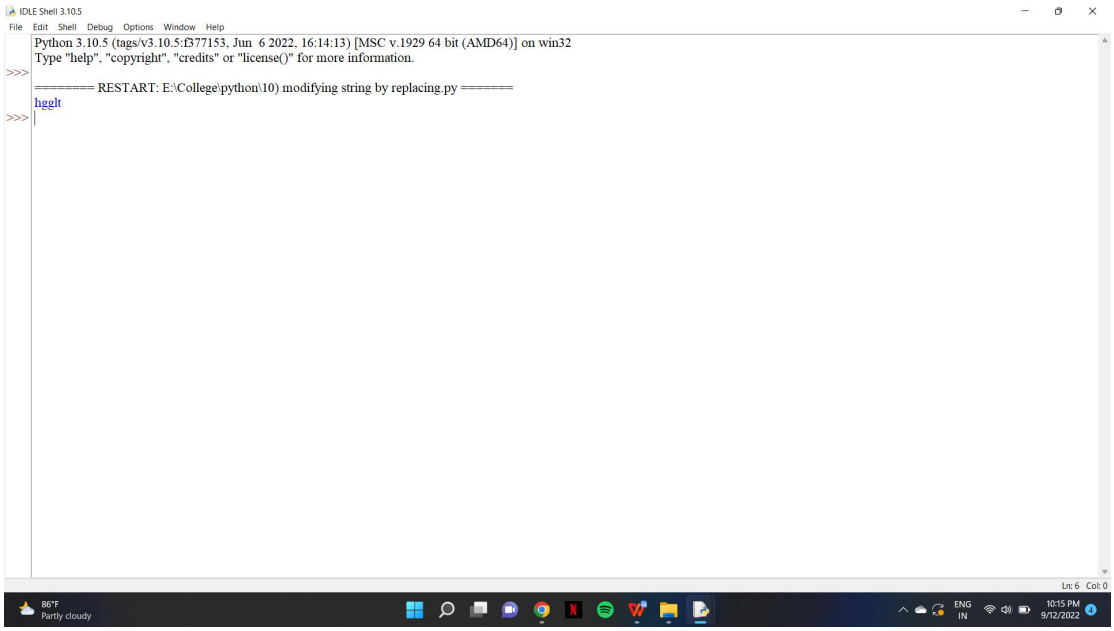
OUTPUT:



9)
10) def addFrequencyToCharacter(s):

```
    frequency = [0] * 26

    n = len(s)

    for i in range(n):

        frequency[ord(s[i]) - ord('a')] += 1

    for i in range(n):

        add = frequency[ord(s[i]) - ord('a')] % 26

        if (ord(s[i]) + add <= ord('z')):
            s[i] = chr(ord(s[i]) + add)

        else:
            add = (ord(s[i]) + add) - (ord('z'))
            s[i] = chr(ord('a') + add - 1)

    print("".join(s))

str = "geeks"
addFrequencyToCharacter([i for i in str])
```

OUTPUT:


IDLE Shell 3.10.5
File Edit Shell Debug Options Window Help
Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
======== RESTART: E:\College\python\10) modifying string by replacing.py ========
hggIt
>>>
Ln: 6 Col: 0
86°F
Partly cloudy
ENG
IN
10:15 PM
9/12/2022