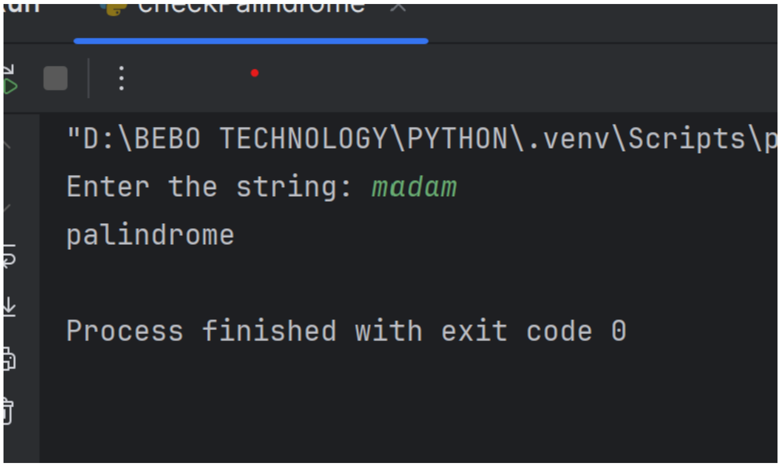
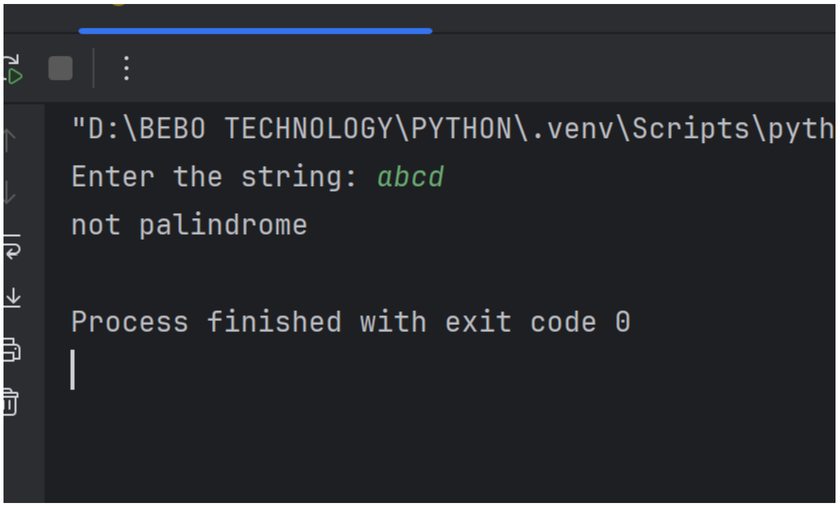
Assessment-3

1.Check if a string is a palindrome.

```
n = input("Enter the string: ")

rev_str = ""

for i in n:
    rev_str = i + rev_str

if rev_str == n:
    print("palindrome")
else:
    print("not palindrome")
```



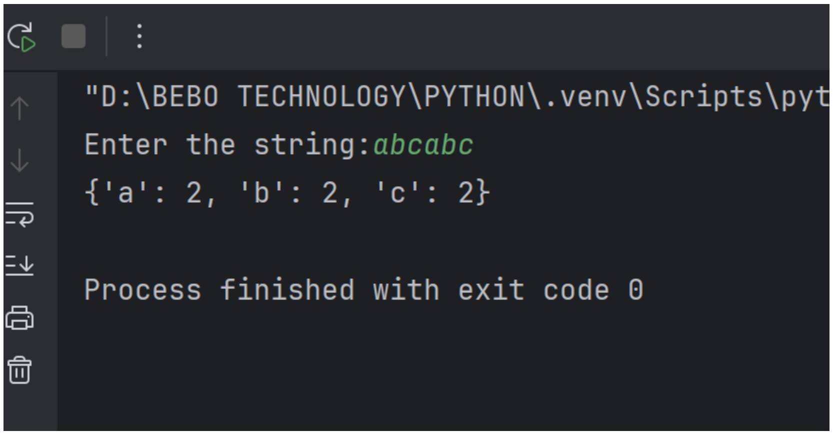
```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\p
Enter the string: madam
palindrome

Process finished with exit code 0
```



```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\pyth
Enter the string: abcd
not palindrome

Process finished with exit code 0
```

2.Count the frequency of characters in a string

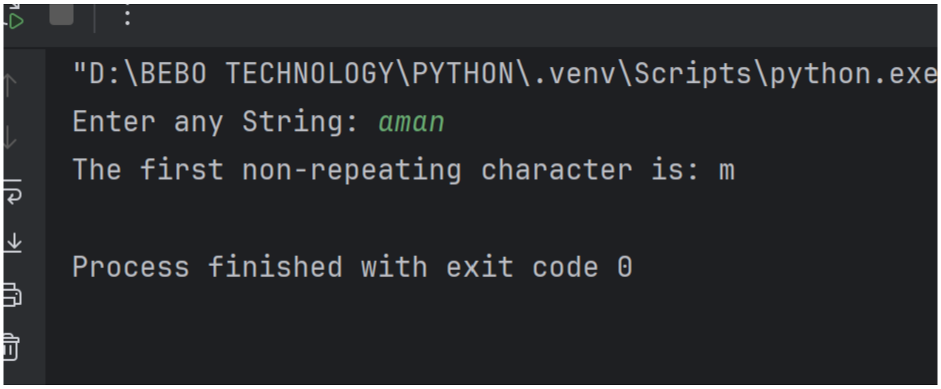
```
s1 = input("Enter any String: ")

f = {}

for i in s1:
    if i in f:
        f[i]+=1
    else:
        f[i] = 1
print(f)
```



```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\pyt
Enter the string:abcabc
{'a': 2, 'b': 2, 'c': 2}

Process finished with exit code 0
```

3.Find the first non-repeating character

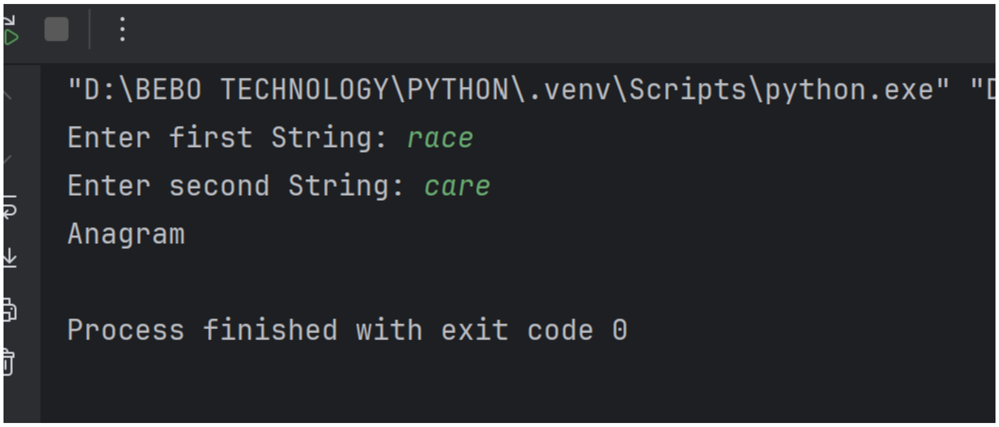
```
s1 = input("Enter any String: ")

d = {}

for i in s1:
    if i in d:
        d[i]+=1
    else:
        d[i] = 1

for i in s1:
    if d[i] == 1:
        print("The first non-repeating character is:", i)
        break
```



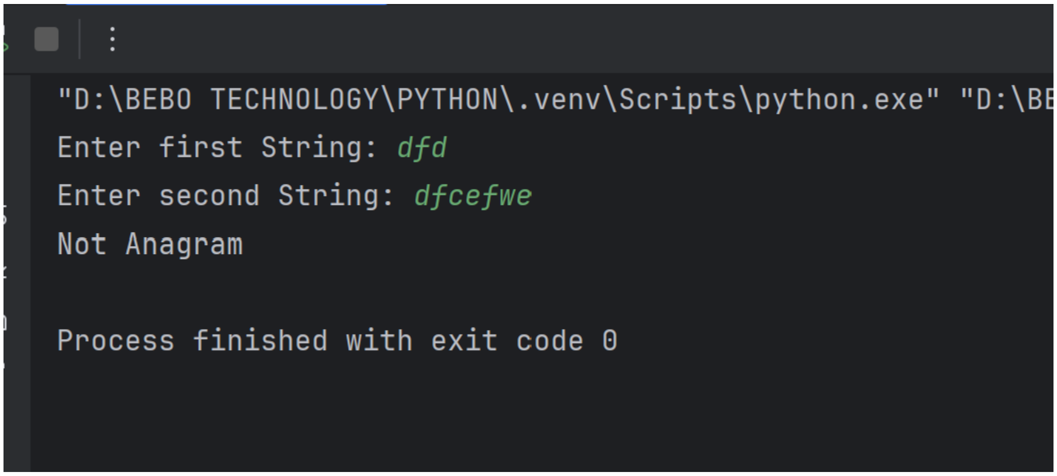
```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\python.exe
Enter any String: aman
The first non-repeating character is: m

Process finished with exit code 0
```

4.Check if two strings are anagrams

```
s1 = input("Enter first String: ")
s2 = input("Enter second String: ")

if len(s1)!=len(s2):
    print("Not Anagram")
else:
    a = sorted(s1)
    b = sorted(s2)
    if(a == b):
        print("Anagram")
    else:
        print("Not Anagram")
```



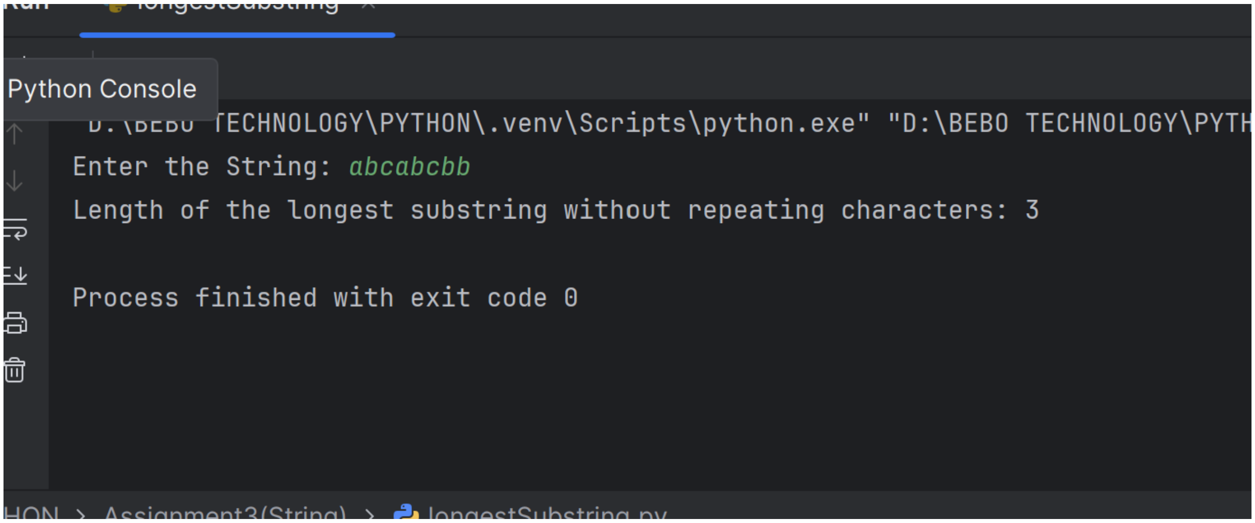
```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\python.exe" "D
Enter first String: race
Enter second String: care
Anagram

Process finished with exit code 0
```



```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\python.exe" "D:\BE
Enter first String: dfd
Enter second String: dfcefwe
Not Anagram

Process finished with exit code 0
```

5.Longest substring without repeating characters

```
n = input("Enter the String: ")
st = 0
m = 0
d = {}

for i in range(len(n)):
    if n[i] in d and d[n[i]] >= st:
        st = d[n[i]] + 1
    d[n[i]] = i
    m = max(m, i - st + 1)

print("Length of the longest substring without repeating characters:", m)
```

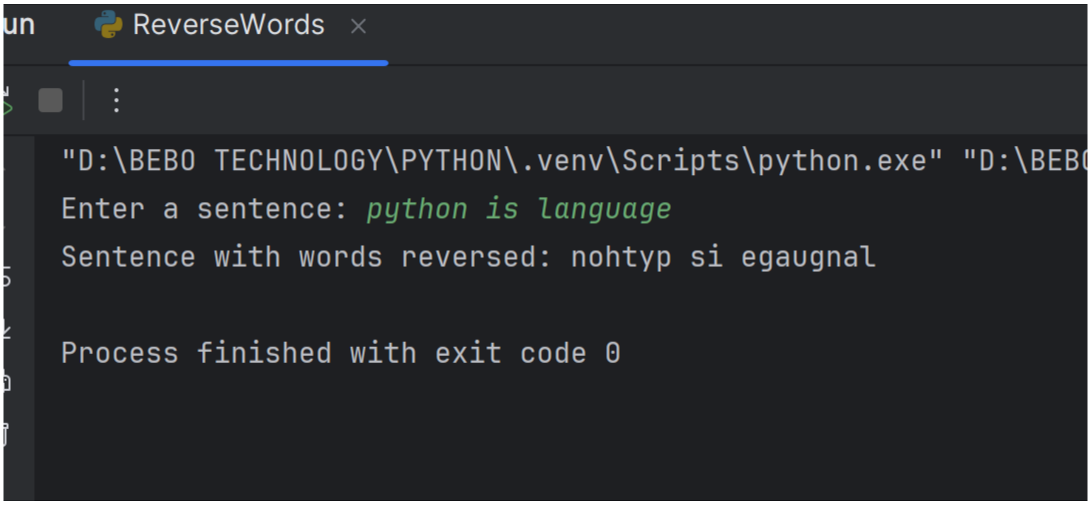


```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\python.exe" "D:\BEBO TECHNOLOGY\PYTH
Enter the String: abcabcbb
Length of the longest substring without repeating characters: 3

Process finished with exit code 0
```

6.Write a Python program that reverses the order of words in a given

sentence.

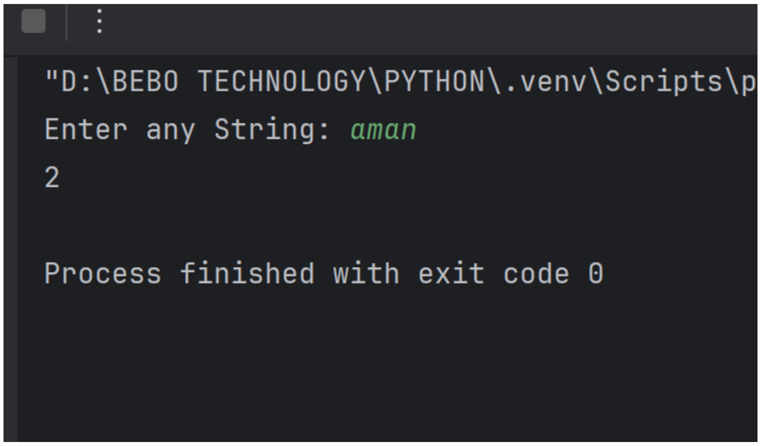
```
def reverse_characters_in_words(sentence):
    words = sentence.split()
    reversed_words = [word[::-1] for word in words]
    reversed_sentence = ' '.join(reversed_words)
    return reversed_sentence


sentence = input("Enter a sentence: ")
print("Sentence with words reversed:", reverse_characters_in_words(sentence))
```



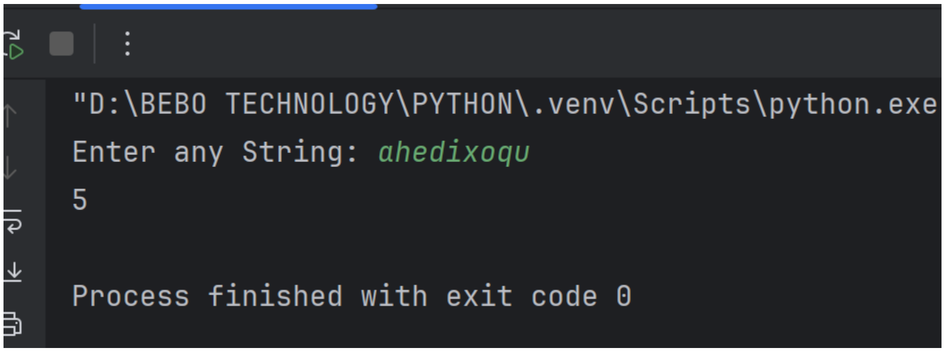
```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\python.exe" "D:\BEB
Enter a sentence: python is language
Sentence with words reversed: nohtyp si egaugnal

Process finished with exit code 0
```

7.Count Vowels in a String

```
n = input("Enter any String: ")
count = 0
for i in n:
    if i in "aeiou":
        count+=1
print(count)
```



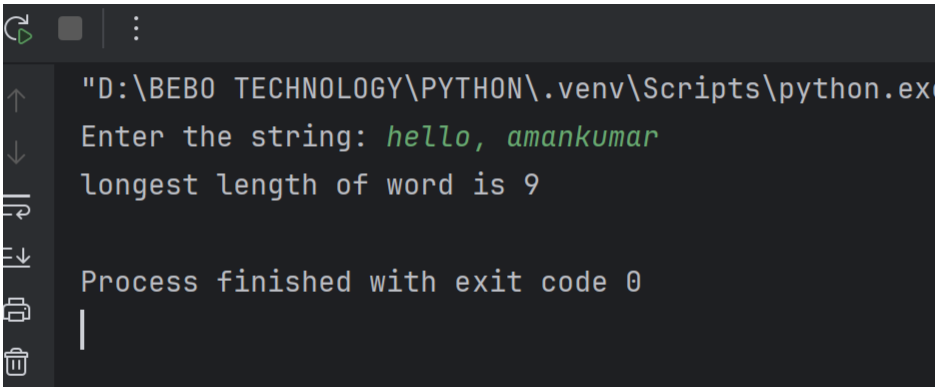
```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\p
Enter any String: aman
2

Process finished with exit code 0
```



```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\python.exe
Enter any String: ahedixoqu
5

Process finished with exit code 0
```

8.Write a Python function to find the longest word in a given sentence.

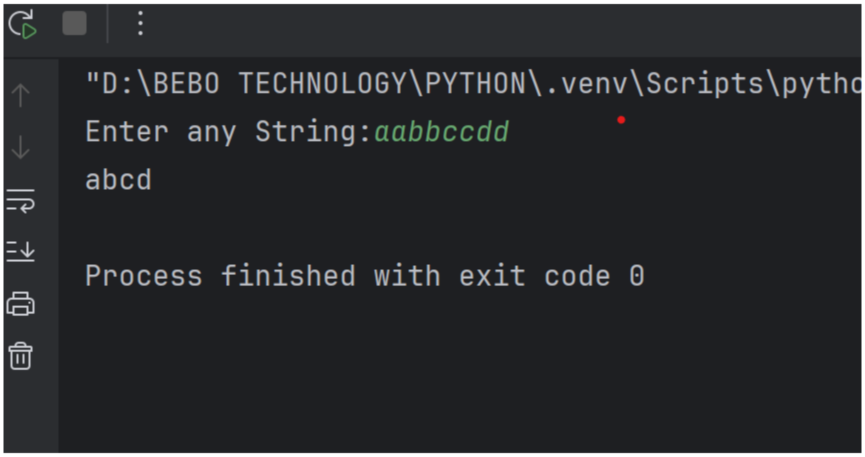
```
n = input("Enter the string: ")

longest_word = n.split()
length = 0

for word in longest_word:
    if len(word) > length:
        length = len(word)
print(length)
```



```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\python.ex
Enter the string: hello, amankumar
longest length of word is 9

Process finished with exit code 0
```

9.Write a Python program to remove duplicate characters from a string while preserving the order of characters.

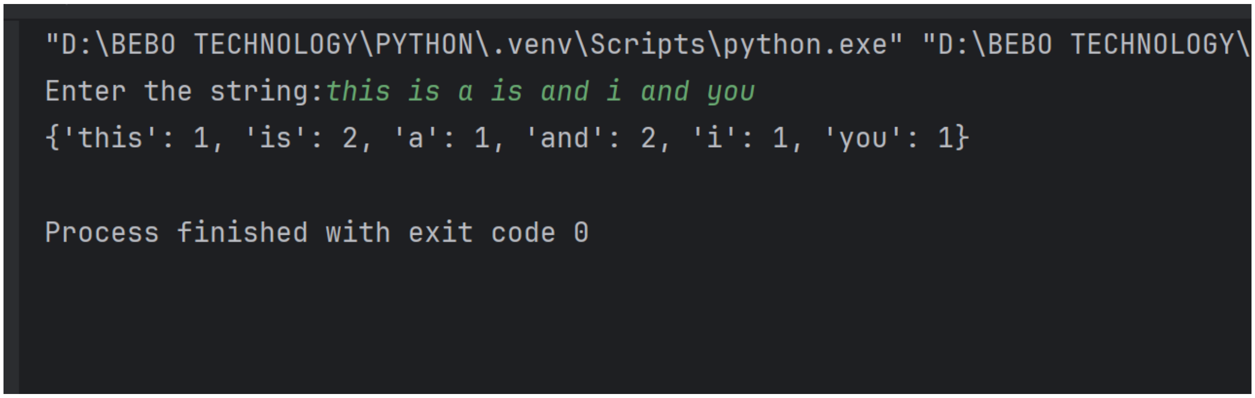
```
def remove_dup(n):
    list = []

    for i in n:
        if i not in list:
            list.append(i)
    st = "".join(list)
    return st

n = input("Enter any String:")
print(remove_dup(n))
```



```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\pytho
Enter any String:aabbccdd
abcd

Process finished with exit code 0
```

10.Write a Python program to count the occurrences of each word in a given sentence.

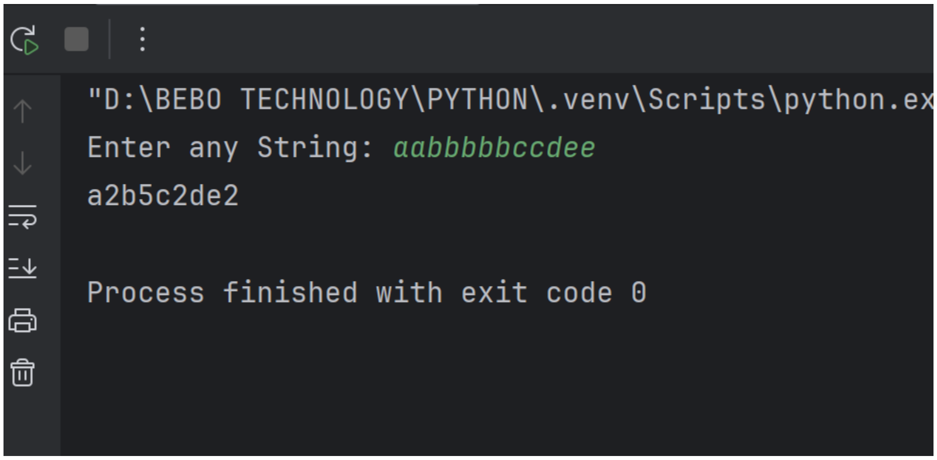
```
s1 = input("Enter the string:")
f = {}
words = s1.split()

for word in words:
    if word in f:
        f[word] += 1
    else:
        f[word] = 1
print(f)
```



```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\python.exe" "D:\BEBO TECHNOLOGY\
Enter the string:this is a is and i and you
{'this': 1, 'is': 2, 'a': 1, 'and': 2, 'i': 1, 'you': 1}

Process finished with exit code 0
```

11.Write a Python program to compress a string by replacing consecutive repeating characters with the character followed by its count.

```
def String_compression(st):
    i = 0
    compressed = ""
    while i < len(st):
        count = 1
        while i < len(st) - 1 and st[i] == st[i + 1]:
            i += 1
            count += 1
        compressed += st[i]
        if count > 1:
            compressed += str(count)
        i += 1
    return compressed

st = input("Enter any String: ")
result = String_compression(st)
print(result)
```



```
"D:\BEBO TECHNOLOGY\PYTHON\.venv\Scripts\python.ex
Enter any String: aabbbbbccdee
a2b5c2de2

Process finished with exit code 0
```