maketrans and translate

In [4]:
```
intab='aeiou'
outtab='12345'
my='this is string translate function..!!'
transtab=my.maketrans(intab,outtab)
print(my.translate(transtab))
```

```
th3s 3s str3ng tr1nsl1t2 f5nct34n..!!
```

In [5]:
```
print(transtab)
```

```
{97: 49, 101: 50, 105: 51, 111: 52, 117: 53}
```

removing punctuations

In [6]:
```
import string
```

In [7]:
```
string.punctuation
```

Out[7]:
```
'!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'
```

In [8]:
```
mystring='hello! W#@$or^$ld'
newstring=mystring.translate(str.maketrans("","",string.punctuation))# third
print(newstring)
```

```
hello World
```

write a program to create a function to find the length of the string without using len method

In [3]:
```
s=input("enter the value")
def length(s):
    count=0
    for i in s:
        count+=1
        print(count)

```

```
enter the value4158
```

In [ ]:
```
length(s)
```

write a program to swapcase

In [2]:
```
s=input("enter string")
ans=""
for i in s:
    if i.isupper():
        ans+=i.lower()
    elif i.islower():
        ans+=i.upper()
    else:
        ans+=i
print(ans)
```

```
enter stringsD
Sd
```

write a program to find num of digit in given string the sum of the digit and the avg

In [1]:
```
s=input("enter the string")
sum=0
count=0
for i in s:
    if i.isdigit():
        count+=1
        sum+=int(i)
print(f'sum is:{sum}')
print(f'avg is:{sum//count}')
print(f'digit is:{count}')
```

```
enter the stringsf 453648
sum is:30
avg is:5.0
digit is:6
```

write a program to shift the number of following condition

input - given string -12345

In [12]:
```
inp=int(input("how many type shift"))
inpn=int(input("enter the number"))

def shift_num(inp,inpn):
    num=str(inpn)
    digit=len(num)

    if inp>digit:
        result=num[::-1]
    else:
        result=num[inp:]+num[:inp]
    return int(result)
shift_num(inp,inpn)
```

```
how many type shift2
enter the number12345
```

Out[12]:
```
34512
```

In [ ]:
```

```

In [14]:
```
## ziping

a=1,2,3
b=4,5,6
print(tuple(zip(a,b)))
list(zip(a,b))
```

```
((1, 4), (2, 5), (3, 6))
```

Out[14]:
```
[(1, 4), (2, 5), (3, 6)]
```

In [16]:
```
## tuple allows different data structure as element
t=(None,1,[1,2],'hi')
print(t)
```

```
(None, 1, [1, 2], 'hi')
```

In [17]:
```
# index
t[3]
```

Out[17]:
```
'hi'
```

In [18]:
```
t[2][0]=777
```

In [19]:
```
print(t)
```

```
(None, 1, [777, 2], 'hi')
```

In [20]:
```
# type conversion
t=tuple('hello')
t
```

Out[20]:
```
('h', 'e', 'l', 'l', 'o')
```

In [21]:
```
t1=1,2,3
t2=4,5,6
print(t1+t2)
print(t1*3)
```

```
(1, 2, 3, 4, 5, 6)
(1, 2, 3, 1, 2, 3, 1, 2, 3)
```

In [22]:
```
t=1,2,3,4
print(4 in t)
```

```
True
```

In [23]:
```
for i in t :
    print(i)
```

```
1
2
3
4
```

In [24]:
```
for i in enumerate(t):
    print(i)
```

```
(0, 1)
(1, 2)
(2, 3)
(3, 4)
```

In [25]:
```
for i,j in enumerate(t):
    print(i,j)
```

```
0 1
1 2
2 3
3 4
```

In [26]:
```
t=78,5245,78,15,7458,541,10
print(len(t))
print(min(t))
print(max(t))
print(sorted(t))
print(sorted(t,reverse=True))
```

```
7
10
7458
[10, 15, 78, 78, 541, 5245, 7458]
[7458, 5245, 541, 78, 78, 15, 10]
```

In [31]:
```
t1=1,2,3
t2=4,5,6
t1==t2

```

Out[31]:
```
False
```

In [30]:
```
t1=1,2,3
t2=4,5,6
t1<t2

```

Out[30]:
```
True
```

In [33]:
```
t1=1,2,3
t2=1,2,3,4
t1==t2

```

Out[33]:
```
False
```

In [46]:
```
t=(1,2,3,3)

```

In [40]:
```
sum
```

Out[40]:
```
30
```

In [41]:
```
t
```

Out[41]:
```
(1, 2, 3)
```

In [47]:
```
t.count(3)
```

Out[47]:
```
2
```

In [ ]:
```

```