Python fundamentals day 19

Today's Agenda

- Strings built-in functions (contd)
- Programs on strings
- String translation



Strings built-in functions (contd)

To continue with the built-in functions, let us see the below example where we shall be checking if the string starts with <a href="https://https.nif.google.com/https://html.nif.google.com/html.nif.google.c

Output:

```
In [35]: runfile('C:/Users/rooman/OneDrive/
Desktop/python/test.py', wdir='C:/Users/rooman/
OneDrive/Desktop/python')
https://www.google.com/
https://www.youtube.com/
```

In the above example we are using string slicing method, now let us see can we do the same using built-in functions

Output:

```
In [36]: runfile('C:/Users/rooman/OneDrive/
Desktop/python/test.py', wdir='C:/Users/rooman/
OneDrive/Desktop/python')
https://www.google.com/
https://www.youtube.com/
```



Now let us see a similar example but now we shall check how many amongst the url list ends with com

Output:

```
In [39]: runfile('C:/Users/rooman/OneDrive/
Desktop/python/test.py', wdir='C:/Users/rooman/
OneDrive/Desktop/python')
https://www.google.com/
https://www.youtube.com/
http://www.xyz.com
```



Let us see is there any built-in function which can do the same function with reduction of code complexity

Output:

```
In [40]: runfile('C:/Users/rooman/OneDrive/
Desktop/python/test.py', wdir='C:/Users/rooman/
OneDrive/Desktop/python')
https://www.google.com/
https://www.youtube.com/
http://www.xyz.com
```

We can simplify the above example like below

Note: If we are passing multiple inputs to endswith() then it should be passed as tuple.

Output:

```
In [41]: runfile('C:/Users/rooman/OneDrive/
Desktop/python/test.py', wdir='C:/Users/rooman/
OneDrive/Desktop/python')
https://www.google.com/
https://www.youtube.com/
http://www.xyz.com
```

Programs on strings

1. Program to count number of lower case, upper case, numbers and special characters present in a string.

```
s=input("Enter the string:")
low_count,up_count,sp_count,num_count=0,0,0

for i in s:
    if i.islower():
        low_count += 1
    elif i.isupper():
        up_count += 1
    elif i.isnumeric():
        num_count += 1
    else:
        sp_count += 1

print("Lower case=",low_count)
print("Upper case=",up_count)
print("Numeric=",num_count)
```



print("Special character=",sp_count)

Output:

```
In [42]: runfile('C:/Users/rooman/OneDrive/
Desktop/python/test.py', wdir='C:/Users/rooman/
OneDrive/Desktop/python')

Enter the string:PyThOn@123
Lower case= 3
Upper case= 3
Numeric= 3
Special character= 1

UPPER lower
lower

1234567

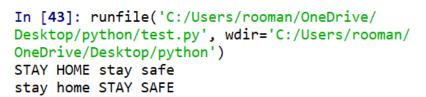
@_#$^&
```

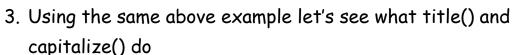
Now why don't you try the same example without using any built-in method.

2. Program to swap the case. That is, change upper case to lower and vice versa using swapcase().

```
s="STAY HOME stay safe"
s1=s.swapcase()
print(s)
print(s1)
```

Output:





```
s="STAY HOME stay safe"
s1=s.swapcase()
print(s)
print(s1)
s2=s.title() # makes the first letter of each word capital
print(s2)
s3=s.capitalize() # makes only the first letter capital
print(s3)
print("python".capitalize())
```

Output:

```
In [45]: runfile('C:/Users/rooman/OneDrive/
Desktop/python/test.py', wdir='C:/Users/rooman/
OneDrive/Desktop/python')
STAY HOME stay safe
stay home STAY SAFE
Stay Home Stay Safe
Stay home stay safe
Python
```

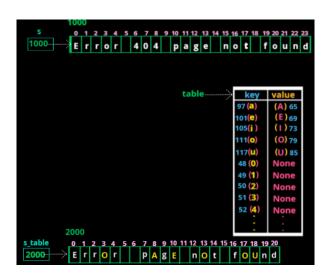


String translation

Python string method translate() returns a copy of the string in which all characters have been translated using table (constructed with the maketrans() function in the string module)

```
s="Error 404 not found"
table=s.maketrans("aeiou","AEIOU","0123456789")
s_table=s.translate(table)
print(s)
print(s_table)
```

Logic:



Output:

```
In [46]: runfile('C:/Users/rooman/OneDrive/
Desktop/python/test.py', wdir='C:/Users/rooman/
OneDrive/Desktop/python')
Error 404 not found
ErrOr nOt fOUnd
```