



Output:

```
D:\python_classes>py test.py
0 ... 1 ... a
2 ... 3 ... b
4 ... 5 ... c
6 ... 7 ... k
8 ... 9 ... z
```

6. sub():

sub means substitution or replacement

`re.sub(regex,replacement,targetstring)`

In the target string every matched pattern will be replaced with provided replacement.

Eg:

```
1) import re
2) s=re.sub("[a-z]","#","a7b9c5k8z")
3) print(s)
```

Output: #7#9#5#8#

Every alphabet symbol is replaced with # symbol

7. subn():

It is exactly same as sub except it can also returns the number of replacements.

This function returns a tuple where first element is result string and second element is number of replacements.

(resultstring, number of replacements)

Eg:

```
1) import re
2) t=re.subn("[a-z]","#","a7b9c5k8z")
3) print(t)
4) print("The Result String:",t[0])
5) print("The number of replacements:",t[1])
```

Output:

```
D:\python_classes>py test.py
('#7#9#5#8#', 5)
The Result String: #7#9#5#8#
The number of replacements: 5
```