In [1]:

```
print(dir(str))
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc_
_', '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__
getnewargs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__ite
r__', '__le__', '__len__', '__lt__', '__mod__', '__mul__', '__ne__', '__new_
_', '__reduce__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__se
tattr__', '__sizeof__', '__str__', '__subclasshook__', 'capitalize', 'casefo
ld', 'center', 'count', 'encode', 'endswith', 'expandtabs', 'find', 'forma
t', 'format_map', 'index', 'isalnum', 'isalpha', 'isascii', 'isdecimal', 'is
digit', 'isidentifier', 'islower', 'isnumeric', 'isprintable', 'isspace', 'i
stitle', 'isupper', 'join', 'ljust', 'lower', 'lstrip', 'maketrans', 'partit
ion', 'replace', 'rfind', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstri
p', 'split', 'splitlines', 'startswith', 'strip', 'swapcase', 'title', 'tran
slate', 'upper', 'zfill']
```

In [3]:

```
s = "python programming"
s.capitalize()  #capitalize the first character
```

Out[3]:

```
'Python programming'
```

In [4]:

```
s1 = "PythonN"
s2 = "python"
s1.casefold()  #converts to lowercase alphabet
```

Out[4]:

```
'pythonn'
```

In [5]:

```
s1.lower()   #always converts to lowercase
```

Out[5]:

```
'pythonn'
```

In [6]:

```
s2.upper()   #converts to upper case
```

Out[6]:

```
'PYTHON'
```

In [7]:

```
s1.swapcase()  # converts uppercase alphabets to lower and lower to uppercase
```

Out[7]:

```
'pYTHONn'
```

In [14]:

```
s = "Python123"
print(s.isdigit())
s1 = "Python Programming 123 #$@"
print(s1.isdigit())
s2 = "12345"
print(s2.isdigit())
s2.isdigit()
```

```
False
False
True
```

Out[14]:

```
True
```

In [17]:

```
s3= "Python Programming"
print(s3.isalpha())
print(s2.isalpha())
print(s1.isalpha())
s4 = "PythonProgramming"
print(s4.isalpha())
```

```
False
False
False
True
```

In [15]:

```
s1 = "Python Programming 123 #$@"
print(s1.isalnum())
```

```
False
```

In [16]:

```
s = "Python123"
print(s.isalnum())
```

```
True
```

In [24]:

```
s3= "Python Programming"
print(s3.split())
```

```
['Python', 'Programming']
```

In [25]:

```
s3= "Python Programming"
s4= s3.split("P")
type(s4)
s3
s4
```

Out[25]:

```
['', 'ython ', 'rogramming']
```

In [27]:

```
s3 = "hdfgPython Programming"
s4 = s3.split("P")
type(s4)
s3
s4
```

Out[27]:

```
['hdfg ', 'ython ', 'rogramming']
```

In [28]:

```
s3 = "hdfgPython#$# Programming"
s4 = s3.split("#")
type(s4)
s3
s4
```

Out[28]:

```
['hdfgPython', '$', ' Programming']
```

In [31]:

```
s1 = "Python Programming 123 #$@"
for i in s1:
    print(i,end=" ")
    if i.isdigit():
        print(i)
```

```
P y t h o n   P r o g r a m m i n g   1 1
2 2
3 3
  # $ @
```

In [34]:

```
s1 = "Python Programming 123 #$@"
print(s1)
for i in s1:
    if i.isdigit():
        print(i)
```

```
Python Programming 123 #$@
1
2
3
```

In [33]:

```
s1 = "Python Programming 123 #$@"
for i in s1:
    if i.isdigit():
        print(i,end="")
```

```
123
```

In [38]:

```
s1 = "Python Programming 123 #$@".split()
print(s1)
for i in s1:
    if i.isdigit():
        print(i)
```

```
['Python', 'Programming', '123', '#$@']
123
```

In [39]:

```
s = "       Python programming"
```

In [40]:

```
print(s.lstrip())
print(s.rstrip())
print(s.strip())
```

```
Python programming
       Python programming
Python programming
```

In [42]:

```
print(s.replace(" ","@"))
print(s.replace("p","#"))
```

```
@@@@@@@Python@programming
       Python #rogramming
```

In [43]:

```
print(s.replace(" ","@"))
print(s.replace("p","#"))
s1 = s.strip()
s1.replace(" ","")
s.replace(" ","")
```

```
@@@@@@@Python@programming
       Python #rogramming
```

Out[43]:

```
'Pythonprogramming'
```

In [44]:

```
s = "Python Programming"
"@".join(s)
"CSE".join(s)
```

Out[44]:

```
'PCSEyCSEtCSEhCSEoCSEnCSE CSEPCSErCSEoCSEgCSErCSEaCSEmCSEmCSEiCSEnCSEg'
```

In [46]:

```
s.count("g")
print(s.count("m"))
print(s.count("mm"))
```

```
2
1
```

In [51]:

```
s.index("g")
```

Out[51]:

```
10
```

In [49]:

```
s = "Python programming"
s.istitle()
```

Out[49]:

```
False
```

In [50]:

```
s.title()
```

Out[50]:

```
'Python Programming'
```

In [52]:

```
s.index("g")
s[10]
```

Out[52]:

```
'g'
```

In [53]:

```
s.index("g")
s[15]
```

Out[53]:

```
'i'
```

In [57]:

```
s = "Python programming 123"
s.istitle()
```

Out[57]:

```
False
```

In [59]:

```
#s = "Python Programming 123"
s.title()
```

Out[59]:

```
'Python Programming 123'
```

In [63]:

```
s.startswith("P")
```

Out[63]:

```
True
```

In [61]:

```
s
```

Out[61]:

```
'Python Programming 123'
```

In [1]:

```
s = "Python"
s #APSSDCPythonAPSSDC # @@@@@@@@@Python@@@@@@@@
s.center()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-1-a37334f24b27> in <module>
      1 s = "Python"
      2 s #APSSDCPythonAPSSDC # @@@@@@@@@Python@@@@@@@@
----> 3 s.center()

TypeError: center() takes at least 1 argument (0 given)
```

In [2]:

```
s = "Python"
s #APSSDCPythonAPSSDC # @@@@@@@@@Python@@@@@@@@
s.center(8,"1")
```

Out[2]:

```
'1Python1'
```

In [3]:

```
s = "Python"
s #APSSDCPythonAPSSDC # @@@@@@@@@Python@@@@@@@@
s.center(5,"@")
```

Out[3]:

```
'Python'
```

In [4]:

```
s = "Python"
s #APSSDCPythonAPSSDC # @@@@@@@@@Python@@@@@@@@
s.center(10,"1")
```

Out[4]:

```
'11Python11'
```

In [2]:

```
s = "Python Python Python Programming"
for i in range(len(s)):
    #print(i) # 0,1,2,3
    #print(s[i]) #pyt
    if s[i] == "P":
         print(i)
```

```
0
7
14
21
```

In [3]:

```
s = "Python Python Python Programming"
c = 0
for i in range(len(s)):
    #print(i) # 0,1,2,3
    #print(s[i]) #pyt
    if s[i] == "P":
        c+=1
print(c)
```

```
4
```

In [4]:

```
s = "Python Python Python Programming"
for i in s:
    if i == "P":
        print(s.index(i))
```

```
0
0
0
0
```

In [ ]:

```

```