In [ ]:
```
==========================================
          String Handling - Part 1
==========================================
- Indexing → Used to access a single character.
- Slicing → Used to extract a range of characters (5 syntaxes).

==========================================
          String Handling - Part 2
==========================================
Along with Indexing and Slicing, we can also perform
various operations using Predefined Functions of *str*.
==========================================
```

In [ ]:
```

```

Key Point :- Strings in Python are immutable. Any operation like .capitalize(), .upper(), .lower(), etc. always creates a new string object. If you want to keep the modified string, you must assign it back to a variable.

# 1. capitalize()

In [2]:
```
s="python" #In python, string are immutable
s.capitalize()
print(s)
```

```
python
```

In [6]:
```
s="python"
print(s,id(s))
```

```
python 1932004130032
```

In [15]:
```
s=s.capitalize()
print(s,id(s))
```

```
Python 1932143826176
```

In [ ]:
```
#Note:
#The method capitalize() does not change the original string.
#Instead, it creates a new string object ("Python") with a different memory address
#That's why the id(s) is different after reassignment.
```

In [23]:
```
s="python is an oop lang"
print(s,id(s))
s=s.capitalize()
print(s,id(s))
```

```
python is an oop lang 1932155902192
Python is an oop lang 1932155906032
```

In [48]:
```
s="python IS AN OOP LANG"
print(s)
```

```
python IS AN OOP LANG
```

In [52]:
```
s.capitalize()
print(s)
```

```
Python is an oop lang
```

# 2. title()

In [44]:
```
s="python"
print(s)
```

```
python
```

In [46]:
```
s.title()
```

Out[46]: 'Python'

In [9]:
```
s="MAHABOOB IS STUDENT OF MRIIRS"
print(s)
```

```
MAHABOOB IS STUDENT OF MRIIRS
```

In [17]:
```
s=s.title()
print(s)
```

```
Mahaboob Is Student Of Mriirs
```

In [25]:
```
s="MRIIRS IS THE BEST FOR CDOE"
print(s)
s=s.title()
print(s)
```

```
MRIIRS IS THE BEST FOR CDOE
Mriirs Is The Best For Cdoe
```

# 3. count()

In [64]:
```
s="MRIIRS"
print(s)
```

```
MRIIRS
```

In [66]:
```
s.count("M")
```

Out[66]: 1

In [68]:
```
s.count("R")
```

Out[68]: 2

In [70]:
```
s.count("I")
```

Out[70]: 2

In [72]:
```
s.count("S")
```

Out[72]: 1

# 4. swapcase()

In [76]:
```
s="MaHaBooB kHaN"
print(s)
s=s.swapcase()
print(s)
```

```
MaHaBooB kHaN
mAhAbOOb KhAn
```

In [82]:
```
s="mahaboob KHAN"
print(s)
s=s.swapcase()
print(s)
```

```
mahaboob KHAN
MAHABOOB khan
```

# 5. upper()

In [84]:
```
s="Mahaboob Khan"
print(s)
s=s.upper()
print(s)
```

```
Mahaboob Khan
MAHABOOB KHAN
```

In [86]:
```
s="Mahaboob Khan123$%"
print(s)
s=s.upper()
print(s)
```

```
Mahaboob Khan123$%
MAHABOOB KHAN123$%
```

# 6. lower()

In [92]:
```
s="MAHABOOB KHAN"
print(s)
s=s.lower()
print(s)
```

```
MAHABOOB KHAN
mahaboob khan
```

In [94]:
```
s="Mahaboob Khan123$%"
print(s)
s=s.lower()
print(s)
```

```
Mahaboob Khan123$%
mahaboob khan123$%
```

# 7. isupper()

In [99]:
```
s="MRIIRS"
s.isupper()
```

Out[99]: True

In [101…
```
s="mriirs"
s.isupper()
```

Out[101… False

In [103…
```
s="JAN24CDOEBCA001"
s.isupper()
```

Out[103… True

In [105…
```
s="12345M"
s.isupper()
```

Out[105… True

# 8. islower()

In [108…
```
s="mriirs"
s.islower()
```

Out[108… True

In [110…
```
s="MriIrs"
s.islower()
```

Out[110… False

In [112…
```
s="12345$%&#@"
s.islower()
```

Out[112… False

# 9. isalpha()

## True → if all characters are alphabetic and the string is not empty.

## False → if the string contains any non-alphabetic characters

(numbers, spaces, symbols, punctuation, etc.) or is empty.

## Write a python program given name is either valid or invalid?

In [123…
```
s="Mahaboobkhan"
s.isalpha()
```

Out[123… True

In [127…
```
s="Mahaboob Khan"
s.isalpha()
```

Out[127… False

In [129…
```
s=" "
s.isalpha()
```

Out[129… False

In [131…
```
s="123"
s.isalpha()
```

Out[131… False

# 10. isdigit()

In [147…
```
s="123 456"
s.isdigit()
```

Out[147… False

In [151…
```
s="123456"
```

```
s.isdigit()
```

Out[151… True

In [153…
```
s="Khan123"
s.isdigit()
```

Out[153… False

In [155…
```
s="12.35"
s.isdigit()
```

Out[155… False

# 11. isalnum()

In [163…
```
s="Mahaboob"
s.isalnum()
```

Out[163… True

In [165…
```
s="JAN24CDOEBCA001"
s.isalnum()
```

Out[165… True

In [167…
```
s="JAN24/CDOE/BCA/001"
s.isalnum()
```

Out[167… False

# 12. isspace()

In [171…
```
s="   "
s.isspace()
```

Out[171… True

In [173…
```
s="Khan MRIIRS CDOE"
s.isspace()
```

Out[173… False

In [175…
```
s="123456"
s.isspace()
```

Out[175… False

# 13. split(delimeter)

In [182…
```
s="MAHABOOB IS STUDENT OF MRIIRS"
len(s)
```

Out[182… 29

In [199…
```
x=s.split()
print(x,type(x),id(x))
```

```
['MAHABOOB', 'IS', 'STUDENT', 'OF', 'MRIIRS'] <class 'list'> 1861908892608
```

In [12]:
```
s="01-06-1982"
x=s.split("-")
print(x,type(x))
print("Day=",x[0])
```

```
['01', '06', '1982'] <class 'list'>
Day= 01
```

In [18]:
```
print("Day=",x[0])
```

```
Day= 01
```

In [20]:
```
print("Month=",x[1])
```

```
Month= 06
```

In [22]:
```
print("Year=",x[2])
```

```
Year= 1982
```

In [62]:
```
s="Mango#Apple#Sberry-kiwi#Gava"
s.split("#")
```

Out[62]: ['Mango', 'Apple', 'Sberry-kiwi', 'Gava']

# 14. Joint(Iterable-Object)

In [74]:
```
lst=["python","is","an","oop","Lang"]
print(lst)
```

```
['python', 'is', 'an', 'oop', 'Lang']
```

In [76]:
```
s=" "
```

In [82]:
```
s.join(lst)
```

Out[82]: 'python is an oop Lang'

# 15. lstrip()

In [97]:
```
s="     Python"
print(s)
```

```
    Python
```

In [99]:
```
len(s)
```

Out[99]: 11

In [114…
```
x=s.lstrip()
print(x)
```

```
Python
```

In [116…
```
len(x)
```

Out[116… 11

# 16. rstrip()

In [106…
```
s="Python     "
print(s)
```

```
Python
```

In [108…
```
len(s)
```

Out[108… 11

In [110…
```
x=s.rstrip()
print(x)
```

```
Python
```

In [112…
```
len(s)
```

Out[112… 11

# 17. strip()

In [9]:
```
s="     Python     "
s
```

Out[9]: '     Python     '

In [11]:
```
len(s)
```

Out[11]: 16

In [15]:
```
s=s.strip()
print(s)
```

```
Python
```

In [17]:
```
len(s)
```

Out[17]: 6

In [19]:
```
s="     Mahaboob Khan     "
s
```

Out[19]: '     Mahaboob Khan     '

In [21]:
```
len(s)
```

Out[21]: 23

In [23]:
```
s=s.strip()
print(s)
```

```
Mahaboob Khan
```

In [25]:
```
len(s)
```

Out[25]: 13

# 18. startswith()

In [31]:
```
s="MRIIRS IS THE BEST FOR CDOE"
s.startswith("MRIIRS")
```

Out[31]: True

In [33]:
```
s.startswith("IS")
```

Out[33]: False

In [35]:
```
s.startswith("THE")
```

Out[35]: False

In [37]:
```
s.startswith("BEST")
```

Out[37]: False

In [39]:
```
s.startswith("FOR")
```

Out[39]: False

In [41]:
```
s.startswith("MR")
```

Out[41]: True

In [43]:
```
s.startswith("M")
```

Out[43]: True

# 19. endswith()

In [46]:
```
s="MRIIRS IS THE BEST FOR CDOE"
s.endswith("CDOE")
```

Out[46]: True

In [48]:
```
s.endswith("DOE")
```

Out[48]: True

In [50]:
```
s.endswith("E")
```

Out[50]: True

In [52]:
```
s.endswith("CD")
```

Out[52]: False

In [54]:
```
s.endswith("MRIIRS")
```

Out[54]: False

In [ ]:
```
---- THE END ---
```