string

```
A String is a data structure in Python that represents a sequence of
characters.
It is an immutable data type, meaning that once you have created a
string, you cannot change it.
Strings are used widely in many different applications,such as storing
and manipulating text data.
for representing names, addresses
```

how to read string

In [2]:
```
string1='python' # single quote
string1
```

Out[2]: 'python'

In [3]:
```
string2="python" # double quotes
string2
```

Out[3]: 'python'

In [4]:
```
### Triple quotes
# Doc string is used to say some information about your python code
```

for type of string we use

In [5]:
```
string1='python'
type(string1)
```

Out[5]: str

for length of string we use

In [6]:
```
len(string1)
```

Out[6]: 6

for max and min ascii value we use

In [7]:
```
max(string1),min(string1)
```

Out[7]: ('y', 'h')

tip-this values are baased on ascii value of char

addition of two string

In [8]:
```
str1='hello'
str2='python'
str1+str2
```

Out[8]:
```
'hellopython'
```

docstrings

```
##documentation strings (or docstrings) provide a convenient way of
associating documentation with Python modules, functions, classes, and
methods. It’s specified in source code that is used, like a comment, to
document a specific segment of code. Unlike conventional source code
##it write i triple double qoutes.
```

str1-str2 # str1*str2 # str1/str2 not possible it gives type error

operator in python (for loop)

In [9]:
```
str1='python'
for i in str1:
    print(i)
```

```
p
y
t
h
o
n
```

In [11]:
```
str='python'
for i in range(len('python')):
    print(i)
```

```
0
1
2
3
4
5
```

##range() : you need to provide number inside the range ##in : is used only for strings ##if you want print the letters using for loop go for in operator

In [15]:
```
print(ord('p'))
print(ord('y'))
print(ord('t'))
print(ord('h'))
print(ord('o'))
print(ord('n'))
```

```
112
121
116
104
111
110
```

```
dir('')
```

string methods

In [18]:
```
str1='python'
str1.upper()
```

Out[18]: 'PYTHON'

In [19]:
```
##upper : ALL letters are in upper case
```

lower

In [21]:
```
str1.lower()
```

Out[21]: 'python'

##lower : All letters are in lower case

count

In [24]:
```
str1='welcome to python'
str1.count('o')
```

Out[24]: 3

casefold

In [25]:
```
string1='WelCome'
string1.casefold()
```

Out[25]: 'welcome'

##casefold : Case less comparision(lower case)

capitilize

In [26]:
```
string1='welcome'
string1.capitalize()
```

Out[26]: 'Welcome'

In [27]:
```
#capitalize : First letter as capital
```

replace

In [28]:
```
string1='welcome'
# replace 'l' with 'L'
string1.replace('l','@')
```

Out[28]: 'we@come'

index

In [30]:
```
str1='python'
str1.index('y')
```

Out[30]: 1

find

In [32]:
```
str1='python'
str1.find('y')
```

Out[32]: 1

strip

In [33]:
```
## strip is used for remove spaces
```

In [34]:
```
str1=' hello how are you '
print(str1.strip())
```

```
hello how are you
```

startswith- endswith

In [36]:
```
str1='hai how are you'
str1.startswith('hai how are you')
#str1.startswith('h')
```

Out[36]: True

In [37]:
```
str1.endswith('you')
```

Out[37]: True

isalpha/isnumeric/isalnum

In [38]:
```
str1='90hai hello 8 888how are you'
str1.isalnum()
```

Out[38]: False

In [39]:
```
str1='abc'
str1.isalpha()
```

Out[39]: True

In [40]:
```
str1='1234'
str1.isnumeric()
```

Out[40]: True

split

In [42]:
```
str1='hai howw are you'
str1.split()
```

Out[42]: ['hai', 'howw', 'are', 'you']

In [ ]:
```

```