strings

- string is a sequence of characters
- string are immutable(un changeable)
- represented with ' ' ,""

In [2]:
```
 s=12
type(s)
```

Out[2]:
```
int
```

In [3]:
```
s=input('enter a value')
print(s)
type(s)
```

```
enter a value2
2
```

Out[3]:
```
str
```

In [6]:
```
s='hello ece'
len(s)
```

Out[6]:
```
9
```

In [8]:
```
s=int(input('enter a value'))
print(s)
type(s)
```

```
enter a value4
4
```

Out[8]:
```
int
```

In [21]:
```
s="Hello ammu"
```

In [25]:
```
#string slicing
s[4:10]
```

Out[25]:
```
'o ammu'
```

In [19]:
```
print(dir(str),end=' ')
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__',
'__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__getnewa
rgs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__iter__', '__l
e__', '__len__', '__lt__', '__mod__', '__mul__', '__ne__', '__new__', '__reduce
__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__setattr__', '__siz
eof__', '__str__', '__subclasshook__', 'capitalize', 'casefold', 'center', 'cou
nt', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format_map', 'inde
x', 'isalnum', 'isalpha', 'isascii', 'isdecimal', 'isdigit', 'isidentifier', 'i
slower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join',
'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'replace', 'rfind', 'rind
ex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'split', 'splitlines', 'startsw
ith', 'strip', 'swapcase', 'title', 'translate', 'upper', 'zfill']
```

In [26]:
```
s='abc123'
s.isalpha()
```

Out[26]: False

In [27]:
```
s.isalnum()
```

Out[27]: True

In [28]:
```
s.isdigit()
```

Out[28]: False

In [29]:
```
s.isnumeric()
```

Out[29]: False

In [30]:
```
s.islower()
```

Out[30]: True

In [31]:
```
s.isspace()
```

Out[31]: False

In [33]:
```
s='hello ece'
s.capitalize()
```

Out[33]: 'Hello ece'

In [35]:
```
s.casefold()
```

Out[35]: 'hello ece'

In [36]:
```
s1='hello ammu'
s1.lower()
```

Out[36]:
```
'hello ammu'
```

In [37]:
```
s1.center(15)
```

Out[37]:
```
'   hello ammu  '
```

In [38]:
```
s1.count('b')
```

Out[38]:
```
0
```

In [39]:
```
s1.count('a')
```

Out[39]:
```
1
```

In [41]:
```
s1='problem sloving and programming python'
s1.count('pro')
```

Out[41]:
```
2
```

In [43]:
```
s1.find('p')
```

Out[43]:
```
0
```

In [44]:
```
s1.find('py')
```

Out[44]:
```
32
```

In [45]:
```
s1.find('a')
```

Out[45]:
```
16
```

In [46]:
```
s1.find('kw')
```

Out[46]:
```
-1
```

In [48]:
```
s1='apssdc'
s2='python'
s1.join(s2)
```

Out[48]:
```
'papssdcyapssdctapssdchapssdcoapssdcn'
```

In [51]:
```
# split method
s1.split('a')
```

Out[51]:
```
['', 'pssdc']
```

In [52]:
```
s1[0]
```

Out[52]:
```
'a'
```

In [53]:
```
s2='hello k'
s2[0]
```

Out[53]:
```
'h'
```

In [54]:
```
s2='hello k'
s2=s2.split(' ')
s2
```

Out[54]:
```
['hello', 'k']
```

In [55]:
```
s2[0]
```

Out[55]:
```
'hello'
```

In [58]:
```
s2[1]
```

Out[58]:
```
'k'
```

In [59]:
```
s2='python workshop'
s2=s2.split(' ')
s2
```

Out[59]:
```
['python', 'workshop']
```

In [64]:
```
# in= 'python workshop'
# o/p = w.python
st=input('enter a value')
print(len(st))
st=st.split()
print(st)
print(st[1][0])
```

```
enter a valuepython workshop
15
['python', 'workshop']
w
```

In [65]:
```
len(st)
```

Out[65]:
```
2
```

In [68]:
```
# in= 'python workshop'
# o/p = w.python
st=input('enter a value')
st=st.split()
print(st[1][0]+'.',st[0])
```

```
enter a valuePYTHON WORKSHOP
W. PYTHON
```

In [69]:
```
s='hai'
s[::-1]
```

Out[69]: 'iah'

In [70]:
```
s='hai'
s[::2]
```

Out[70]: 'hi'

In [71]:
```
s='hai'
s[::3]
```

Out[71]: 'h'

In [72]:
```
s='hai'
s[::-2]
```

Out[72]: 'ih'

In [73]:
```
s='hello kids'
s.startswith('k')
```

Out[73]: False

In [74]:
```
s='hello kids'
s.startswith('h')
```

Out[74]: True

In [75]:
```
s='hello kids'
s.endswith('k')
```

Out[75]: False

In [78]:
```
#strip - to remove unwanted spaces
s1='hello dude     '
s1.strip()
```

Out[78]: 'hello dude'

In [79]:
```
s2='krishna'
s2.lower()
```

Out[79]: 'krishna'

In [80]:
```
s2.title()
```

Out[80]: 'Krishna'

In [81]:
```
s2.swapcase()
```

Out[81]: 'KRISHNA'

In [82]:
```
s2.capitalize()
```

Out[82]:
```
'Krishna'
```

data structures in python

- lists
- tuples
- dictionaries
- sets

lists

- collection of data of different data types
- list are mutable
- represented in [] , comma seperated values
-

In [83]:
```
li=[]
type(li)
```

Out[83]:
```
list
```

In [90]:
```
li=[1,2,3,4,5,'a',',']
li[0]
```

Out[90]:
```
1
```

In [91]:
```
len(li)
```

Out[91]:
```
7
```

In [93]:
```
li[-1]
```

Out[93]:
```
','
```

In [95]:
```
li[::-1]
```

Out[95]:
```
[',', 'a', 5, 4, 3, 2, 1]
```

In [97]:
```
li1=[1,2,3,4,5,]
print(max(li1))
print(min(li1))
print(sum(li1))
```

```
5
1
15
```

In [98]:
```
del(li)
```

In [100]:
```
print(dir(list),end=' ')
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__delitem__', '__dir_
_', '__doc__', '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem
__', '__gt__', '__hash__', '__iadd__', '__imul__', '__init__', '__init_subclass
__', '__iter__', '__le__', '__len__', '__lt__', '__mul__', '__ne__', '__new__',
'__reduce__', '__reduce_ex__', '__repr__', '__reversed__', '__rmul__', '__setat
tr__', '__setitem__', '__sizeof__', '__str__', '__subclasshook__', 'append', 'c
lear', 'copy', 'count', 'extend', 'index', 'insert', 'pop', 'remove', 'revers
e', 'sort']
```

In [3]:
```
l1=[1,2,3,'a','b','c']
l1.append('10')
print(l1)
```

```
[1, 2, 3, 'a', 'b', 'c', '10']
```

In [5]:
```
l1=[1,2,3,'a','b','c']
l1.clear()
print(l1)
```

```
[]
```

In [6]:
```
l1=[1,2,3,'a','b','c']
l1.append('10')
l1
```

Out[6]:
```
[1, 2, 3, 'a', 'b', 'c', '10']
```

In [7]:
```
l2=[1,3,4,5,'a','d']
l2.append('6')
l2
```

Out[7]:
```
[1, 3, 4, 5, 'a', 'd', '6']
```

In [9]:
```
l1.extend(l2)
l1
```

Out[9]:
```
[1,
 2,
 3,
 'a',
 'b',
 'c',
 '10',
 1,
 3,
 4,
 5,
 'a',
 'd',
 '6',
 1,
 3,
 4,
 5,
 'a',
 'd',
 '6']
```

In [10]:
```
l2.copy()
```

Out[10]:
```
[1, 3, 4, 5, 'a', 'd', '6']
```

In [18]:
```
l2=[1,2,3,4,'a']
```

In [20]:
```
l3=l2.copy()
```

In [21]:
```
print(l3)
```

```
[1, 2, 3, 4, 'a']
```

In [22]:
```
print(l3)
```

```
[1, 2, 3, 4, 'a']
```

In [23]:
```
l3.count(19)
```

Out[23]:
```
0
```

In [25]:
```
l3.index(1)
```

Out[25]:
```
0
```

In [1]:
```
s='krishna'
s[0]
#s[0]='h'
s=s.split('k')
print(s)
print(s[0])
s[1]='h1'
print(s)
```

```
['', 'rishna']

['', 'h1']
```

In [26]:
```
l3
```

Out[26]:
```
[1, 2, 3, 4, 'a']
```

In [30]:
```
l3.insert(2,'kri')
print(l3)
```

```
[1, 2, 'kri', 'kri', 3, 4, 'a']
```

In [31]:
```
l3.pop()
```

Out[31]:
```
'a'
```

In [32]:
```
print(l3)
```

```
[1, 2, 'kri', 'kri', 3, 4]
```

In [34]:
```
l3.remove('kri')
print(l3)
```

```
[1, 2, 3, 4]
```

In [35]:
```
l3
```

Out[35]:
```
[1, 2, 3, 4]
```

In [ ]:
```

```