-indexing

In [ ]:
```
# make a string
```

In [1]:
```
a = 'samosa,pakora'
a
```

Out[1]:
```
'samosa,pakora'
```

In [2]:
```
a[0]
```

Out[2]:
```
's'
```

In [6]:
```
a[0:6]
```

Out[6]:
```
'samosa'
```

In [7]:
```
#last index is Exclusive
a[0:13]
```

Out[7]:
```
'samosa,pakora'
```

In [9]:
```
len (a)
```

Out[9]:
```
13
```

String methods

In [10]:
```
food='beryani'
food
```

Out[10]:
```
'beryani'
```

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

Out[11]:
```
7
```

In [ ]:
```
#change frist word capital
food.capitalize()
```

In [14]:
```
#change every word capital Upper case
food.upper()
```

Out[14]:
```
'BERYANI'
```

In [15]:
```
#Replace
food.replace("b","sh")
```

Out[15]:
```
'sheryani'
```

In [17]:
```
#counting a spesific alphabit in string
name = "umar_anwar"
name
```

Out[17]:
```
'umar_anwar'
```

In [22]:
```
name.count('a')
```

Out[22]:
```
3
```

finding an index number in string

In [23]:
```
name = "umar_anwar"
name
```

Out[23]:
```
'umar_anwar'
```

In [24]:
```
name.find('a')
```

Out[24]:
```
2
```

Basic data struture in Python

-1 Tuple

-2 List

-3 Dictionaries

-4 Set

Tuple

- Ordered Collection of elements
- Enclosed in () round braces/Paranthesis
- Different kind of element can be stored
- Once element can be stored you can not change them(unmutatable)

In [26]:
```
tup1 = (2,'umar',2.4,True)
tup1
```

Out[26]:
```
(2, 'umar', 2.4, True)
```

In [27]:
```
#type of Tuple
type(tup1)
```

Out[27]:
```
tuple
```

Indexing in Tuple

In [28]:
```
tup1[1]
```

Out[28]:
```
'umar'
```

In [29]:
```
tup1[0:3]
```

Out[29]:
```
(2, 'umar', 2.4)
```

In [30]:
```
#count of element in tuple
len(tup1)
```

Out[30]:
```
4
```

In [33]:
```
tup2=(3,"class",2.6, False)
tup2
```

Out[33]:
```
(3, 'class', 2.6, False)
```

In [34]:
```
tup1+tup2
```

Out[34]:
```
(2, 'umar', 2.4, True, 3, 'class', 2.6, False)
```

2-List

- orderd collection of elements
- inclosed in [ ] square braces/brack
- Mutateable you can change the value

In [39]:
```
list1=[2,"baba aamar",False]
list1
```

Out[39]:
```
[2, 'baba aamar', False]
```

In [40]:
```
len(list1)
```

Out[40]:
```
3
```

In [41]:
```
type(list1)
```

Out[41]:
```
list
```

In [42]:
```
list1[2]
```

Out[42]:
```
False
```

In [43]:
```
list2=[1,3,"umar_anwar", "umar",250,False]
list2
```

Out[43]:
```
[1, 3, 'umar_anwar', 250, False]
```

In [44]:
```
list1+list2
```

Out[44]:
```
[2, 'baba aamar', False, 1, 3, 'umar_anwar', 250, False]
```

In [45]:
```
list1*2
```

Out[45]:
```
[2, 'baba aamar', False, 2, 'baba aamar', False]
```

In [60]:
```
list1
```

Out[60]:
```
[2, 'baba aamar', False]
```

In [67]:
```
list1.count("umar")
```

Out[67]:
```
0
```

In [1]:
```
aList = [123, 'xyz', 'zara', 'abc', 123];

print ("Count for 123 : ", aList.count(123))
print ("Count for zara : ", aList.count('zara'))
```

```
Count for 123 :  2
Count for zara :  1
```

In [2]:
```
lista=[123,'abc','abc','umar',123]
lista.count(123)
```

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

In [4]:
```
list3=[30,32,34,54,75,20,21,23]
```

In [5]:
```
#List Sorting
list3.sort()
list3
```

Out[5]:
```
[20, 21, 23, 30, 32, 34, 54, 75]
```

In [8]:
```
lista+list3
```

Out[8]:
```
[123, 'abc', 'abc', 'umar', 123, 20, 21, 23, 30, 32, 34, 54, 75]
```

In [9]:
```
list = lista+list3
list
```

Out[9]:
```
[123, 'abc', 'abc', 'umar', 123, 20, 21, 23, 30, 32, 34, 54, 75]
```

3- Dictionaries

- Am unorderd collection of elements
- Key and values
- curly braces or bracket { }
- Mutateable/change the value

In [10]:
```
# food and their prices
food1={"samosa":35,"pakora":70,"raita":20,"chicken roll":40}
food1
```

Out[10]:
```
{'samosa': 35, 'pakora': 70, 'raita': 20, 'chicken roll': 40}
```

In [11]:
```
type(food1)
```

Out[11]:
```
dict
```

In [14]:
```
#extract data
keys1 = food1.keys()
keys1
```

Out[14]:
```
dict_keys(['samosa', 'pakora', 'raita', 'chicken roll'])
```

In [17]:
```
value1 = food1.values()
value1
```

Out[17]:
```
dict_values([35, 70, 20, 40])
```

In [18]:
```
#Adding new element
food1['Tikki']=15
food1
```

Out[18]:
```
{'samosa': 35, 'pakora': 70, 'raita': 20, 'chicken roll': 40, 'Tikki': 15}
```

4- Set

- Unorderd and Unindexed
- Curly braces is used { }
- no duplicate allowed

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: