- indexing

In [4]:
```
# make a string
a = "samosa pakora"
a
```

Out[4]:
```
'samosa pakora'
```

In [5]:
```
a[3]
```

Out[5]:
```
'o'
```

In [6]:
```
a[4]
```

Out[6]:
```
's'
```

In [7]:
```
# dafined length of string
len(a)
```

Out[7]:
```
13
```

In [12]:
```
# ratio
a[0:5]
```

Out[12]:
```
'samos'
```

In [13]:
```
a[-2]
```

Out[13]:
```
'r'
```

In [15]:
```
a[-6:-1]
```

Out[15]:
```
'pakor'
```

In [23]:
```
food = "baryani"
food
```

Out[23]:
```
'baryani'
```

string methods

In [19]:
```
len(food)
```

Out[19]:
```
7
```

In [24]:
```
# capitalize case
```

```
food.capitalize()
```

Out[24]: 'Baryani'

In [25]:
```
food.upper()
```

Out[25]: 'BARYANI'

In [26]:
```
# replace words
food.replace("b","sh")
```

Out[26]: 'sharyani'

In [27]:
```
# count a specific alphabet in a string
name = "hamad with Engg hamad"
name
```

Out[27]: 'hamad with Engg hamad'

In [28]:
```
name.count("a")
```

Out[28]: 4

finding an index number in string

In [29]:
```
name.find("t")
```

Out[29]: 8

In [30]:
```
### how to split a string
food =" i love samosa, pakora, raita, briyani and karahi"
food
```

Out[30]: ' i love samosa, pakora, raita, briyani and karahi'

In [31]:
```
food.split(",")
```

Out[31]: [' i love samosa', ' pakora', ' raita', ' briyani and karahi']

Basic data structure in pythan

1-tuples 2- List 3- Dicionaries 4-set

1- Tuples

- ordered collection of element -enclosed in () round braces
- differnt king of element can be stored -once element are stored you can not change them

In [15]:

```
tup1 = (1,"python",True, 2.5)
tup1
```

Out[15]: (1, 'python', True, 2.5)

In [16]:
```
# type of a tuple
type(tup1)
```

Out[16]: tuple

- indexing tuple

In [17]:
```
tup1[1]
```

Out[17]: 'python'

In [18]:
```
tup1[2]
```

Out[18]: True

In [20]:
```
#last element is exciustive
tup1[0:3]
```

Out[20]: (1, 'python', True)

In [21]:
```
# count of element in tuple
len(tup1)
```

Out[21]: 4

In [22]:
```
tup2 = (2,"ali",3.5,False)
tup2
```

Out[22]: (2, 'ali', 3.5, False)

In [23]:
```
tup1+tup2
```

Out[23]: (1, 'python', True, 2.5, 2, 'ali', 3.5, False)

In [24]:
```
#concatinate adding
tup1*2+tup2
```

Out[24]: (1, 'python', True, 2.5, 1, 'python', True, 2.5, 2, 'ali', 3.5, False)

In [25]:
```
tup3=(20,30,40,50,85)
tup3
```

Out[25]: (20, 30, 40, 50, 85)

In [26]:
```
max(tup3)
```

Out[26]:
```
85
```

In [27]:
```
min(tup3)
```

Out[27]:
```
20
```

---

2 -list

- orders collection of element
- enclosed in [] sqaure braces/bracket
- mutateble you can change the values

In [29]:
```
list1 = [2,"hamad ur rehman",False]
list1
```

Out[29]:
```
[2, 'hamad ur rehman', False]
```

In [30]:
```
type(list1)
```

Out[30]:
```
list
```

In [31]:
```
len(list1)
```

Out[31]:
```
3
```

In [36]:
```
list1[2]
```

Out[36]:
```
False
```

In [37]:
```
list2=[3.5,"khan","ahmed",478,53.2,False]
list2
```

Out[37]:
```
[3.5, 'khan', 'ahmed', 478, 53.2, False]
```

In [38]:
```
list1+list2
```

Out[38]:
```
[2, 'hamad ur rehman', False, 3.5, 'khan', 'ahmed', 478, 53.2, False]
```

In [39]:
```
list1*2
```

Out[39]:
```
[2, 'hamad ur rehman', False, 2, 'hamad ur rehman', False]
```

In [42]:
```
list1.reverse()
list1
```

Out[42]:
```
[False, 'hamad ur rehman', 2]
```

In [43]:
```
# adding sum items
list1.append("youtube chanal")
list1
```

Out[43]:
```
[False, 'hamad ur rehman', 2, 'youtube chanal']
```

In [47]:
```
list3=[20,30,35,50,40,12,15,10]
list3
```

Out[47]:
```
[20, 30, 35, 50, 40, 12, 15, 10]
```

In [48]:
```
# sorting a list
list3.sort()
list3
```

Out[48]:
```
[10, 12, 15, 20, 30, 35, 40, 50]
```

In [49]:
```
list3*3
```

Out[49]:
```
[10,
 12,
 15,
 20,
 30,
 35,
 40,
 50,
 10,
 12,
 15,
 20,
 30,
 35,
 40,
 50,
 10,
 12,
 15,
 20,
 30,
 35,
 40,
 50]
```

In [52]:
```
lists=list1+list2
lists
# also list1+list2
```

Out[52]:
```
[False,
 'hamad ur rehman',
 2,
 'youtube chanal',
 3.5,
 'khan',
 'ahmed',
 478,
```

```
53.2,
False]
```

3 - Dictioaies

-An unorderd collection of elements

- key and value
- curly braces pr bracket{}
- mutatetable change the values

In [56]:
```
#food and their price
food1 = {"samosa":30,"pakora":100,"Ratia":20,"Salad":50,"Chicken Rolls":30}
food1
```

Out[56]:
```
{'samosa': 30, 'pakora': 100, 'Ratia': 20, 'Salad': 50, 'Chicken Rolls': 30}
```

In [57]:
```
type(food1)
```

Out[57]:
```
dict
```

In [61]:
```
#extrect data
keys1 = food1.keys()
keys1
```

Out[61]:
```
dict_keys(['samosa', 'pakora', 'Ratia', 'Salad', 'Chicken Rolls'])
```

In [65]:
```
values1 = food1.values()
values1
```

Out[65]:
```
dict_values([30, 100, 20, 50, 30])
```

In [69]:
```
food1["tikki"]=10
food1
```

Out[69]:
```
{'samosa': 30,
 'pakora': 100,
 'Ratia': 20,
 'Salad': 50,
 'Chicken Rolls': 30,
 'tikki': 10}
```

In [70]:
```
# updates the values
food1["tikki"]=15
food1
```

Out[70]:
```
{'samosa': 30,
 'pakora': 100,
 'Ratia': 20,
 'Salad': 50,
 'Chicken Rolls': 30,
 'tikki': 15}
```

In [71]:
```
food2 = {"dates":50,"choclate":200,"swayan":1000}
```

In [ ]:
```
food2
```

Out[71]:
```
{'dates': 50, 'choclate': 200, 'swayan': 1000}
```

In [72]:
```
# cancatinate
food1.update(food2)
```

In [73]:
```
food1
```

Out[73]:
```
{'samosa': 30,
 'pakora': 100,
 'Ratia': 20,
 'Salad': 50,
 'Chicken Rolls': 30,
 'tikki': 15,
 'dates': 50,
 'choclate': 200,
 'swayan': 1000}
```

4- sets

- unorderd and unindexed
- curly braces are used()
- No duplicates allowed

In [5]:
```
s1 = {1, 2.2, 5.2, "Hamad", "swabi",True}
s1
```

Out[5]:
```
{1, 2.2, 5.2, 'Hamad', 'swabi'}
```

In [6]:
```
s1.add("ali")
s1
```

Out[6]:
```
{1, 2.2, 5.2, 'Hamad', 'ali', 'swabi'}
```

In [10]:
```
s1.remove("Hamad")
s1
```

Out[10]:
```
{1, 2.2, 5.2, 'swabi'}
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: